Editor: Dana Simian

Proceedings of The International Conference on Applied Informatics

ICDD

IMAGINATION, CREATIVITY, DESIGN, DEVELOPMENT

Romania, Sibiu, May 17 - 19, 2018

Editor DANA SIMIAN

IMAGINATION, CREATIVITY, DESIGN, DEVELOPMENT

Proceedings of the International Conference on Applied Informatics
ICDD

May 17th – 19th, 2018
Sibiu, Romania

"Lucian Blaga" University of Sibiu

"Lucian Blaga" University, Sibiu, 2018

Editor Dana Simian

All papers in this volume were peer review by two independent reviewers

ISSN 2069-964X

Associated Editor Laura Florentina Stoica

Proceedings of the International Conference on Applied Informatics ICDD
May 17th – 19th, 2018, Sibiu, Romania

Preface

This volume contains refereed papers presented within the International Conference on Applied Informatics – ICDD – 2018, which was held between May 17th – 19th, at the Faculty of Sciences, University ” Lucian Blaga” of Sibiu, Romania.

The conference is mainly addressed to bachelor and master level students, PhD students and young researchers from all over the world. The conference gives the participants the opportunity to discuss and present their research on informatics and related fields (like computational algebra, numerical calculus, bioinformatics, etc.). The conference welcomes submissions of original papers on all aspects of informatics and related fields ranging from new concepts and theoretical developments to advanced technologies and innovative applications. Specific topics of the conference included but were not restricted to: Algorithms and data structures, Graph theory and applications, Formal languages and compilers, Cryptography, Modelling and simulation, Computer programming, Computer vision, Computer graphics, Game design, Data mining, Distributed computing, Artificial Intelligence, Service oriented applications, Networking, Grid computing, Mobile operating systems, Scientific computing, Software engineering, Bioinformatics, Robotics, Computer Architecture, Evolutionary Computing, Multimedia Systems, Internet Communication and Technologies, Web Applications.

The conference has brought together participants from 8 countries (Bulgaria, Germany, Italy, Romania, Russia, Serbia, Spain and Ukraine).

We thank all the participants for their interesting talks and discussions. We also thank the members of the scientific committee for their help in reviewing the submitted papers and for their contributions to the scientific success of the conference.

The conference was organized with the support of the following sponsors (in alphabetical order): AUSY, CODEXWORKS, Academia Ardeleană Foundation, GSD, IQUEST, KEEP CALLING, Romanian Ministry of National Education, NTT DATA, Pan Food, OMERON, PROIT, ROPARDO, TOP TECH, VISMA.

September 2018

Dana Simian
Conference chairman

Motto:

"There are no limits, only your imagination"

Proceedings of the International Conference on Applied Informatics
IMAGINATION, CREATIVITY, DESIGN, DEVELOPMENT
Sibiu, Romania

Editor *Prof. PhD Dana Simian*

Scientific committee

Contents:

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Application of component-oriented programming and MVC pattern in development of orthopedic salon information system

Andrey Kabanov Alexandrovich, Alexey Churkin Alexandrovich

Abstract

This article discusses the use of component-oriented programming and mvc pattern in the framework of .NET technology in the development of the information system of the orthopedic salon. Features of the component approach and mvc pattern in solving authorization tasks, organizing access to databases, managing screen forms and displaying information on screen forms and web application pages are analyzed.

1 Introduction

Sooner or later, the management of any company faces the problem of information processes systematization. At the initial stage of activity, a small company uses standard office applications, such as Microsoft Office for data accounting, but the growth of data sets the task of creating special software.
Our goal was to develop orthopedic software and develop an appointment web application. This article describes the process of creating and running software, the features of using .NET technology components and the basic principles of applying MVC pattern to software development.

2 Project implementation

2.1 Implementation of the Windows application for orthopedists

To implement Windows application for orthopedists .NET Framework [1] was chosen.
The user of the application enters the system with his login and password issued by the system administrator. Authorization component encryption part does not differ from most projects and basically uses md5 encryption. The authorization class is inherited from the form class. It contains content items and several private functions that are called during authorization. The function 'VerifyMd5Hash' contains validation for the password, 'GetMd5Hash' returns the already protected password. The form download function prepares the components for authorization and checks the database availability, the form close function completely closes the program if the user is not authorized. After the authentication process, a stream in the form of a "pipe" opens in the background in parallel to communicate with the database. Then this "pipe" is closed for security.

The main menu has a simple design and extensive selection of menus and submenus. It includes sections: Directories, Visits, Journals of Visits, Settings, Help and Exit. Figure 1 shows the "Visit" window. It contains information about the the patient's visit reason, the visit date, visit type and the doctor.

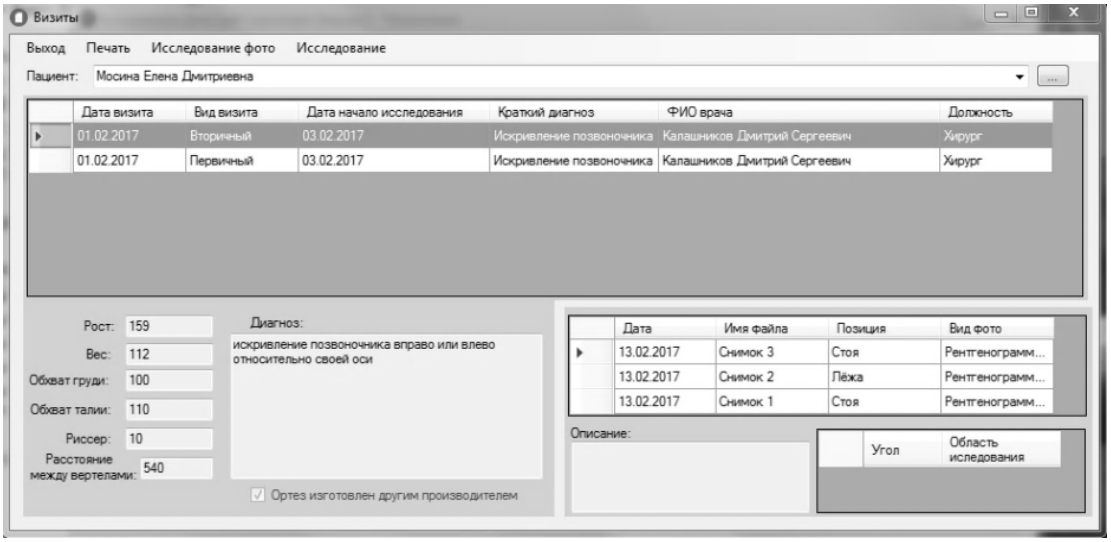


Fig. 1: Form «Visits»

When creating forms, menu components, tables, labels, text windows, lists, database access components were used. When the user selects a patient in the list, the form automatically loads all patient data.

Several variants of databases were considered. At the beginning of the development, the functionality of the Access database was sufficient for use in a simple application, but with the growth of software tasks, it was decided to switch to SQL Server. Database changing gave such advantages as fast connection to database, reduction of response time, storage of objects in different sections, convenient editor and many other important advantages. The database was accessed through standard .NET components: a data set, data tables, adapters, visual and non-visual components.

As a result of the program, Cobb angles and other information are calculated so the doctor can correctly write the conclusion and print it. The menu item "Print" prints the orthosis correction data (Figure 2) and the medical report.

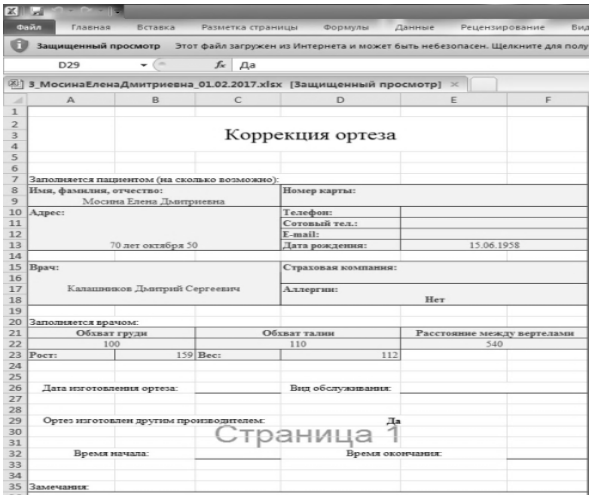


Fig. 2: Report and output to document

The output process uses an object that can work in a thread in parallel with other objects. This object is created on the basis of the library from Microsoft Office. Interop.Excel in which all the necessary functions are implemented.
In the process of programming the "Photos of the Study" form (Figure 3) objects were created to display data, calculate Cobb's angles and implement a slider with the ability to rotate photos. The slider class uses a list of objects with the photo code, patient name, date and photo description obtained from the database.

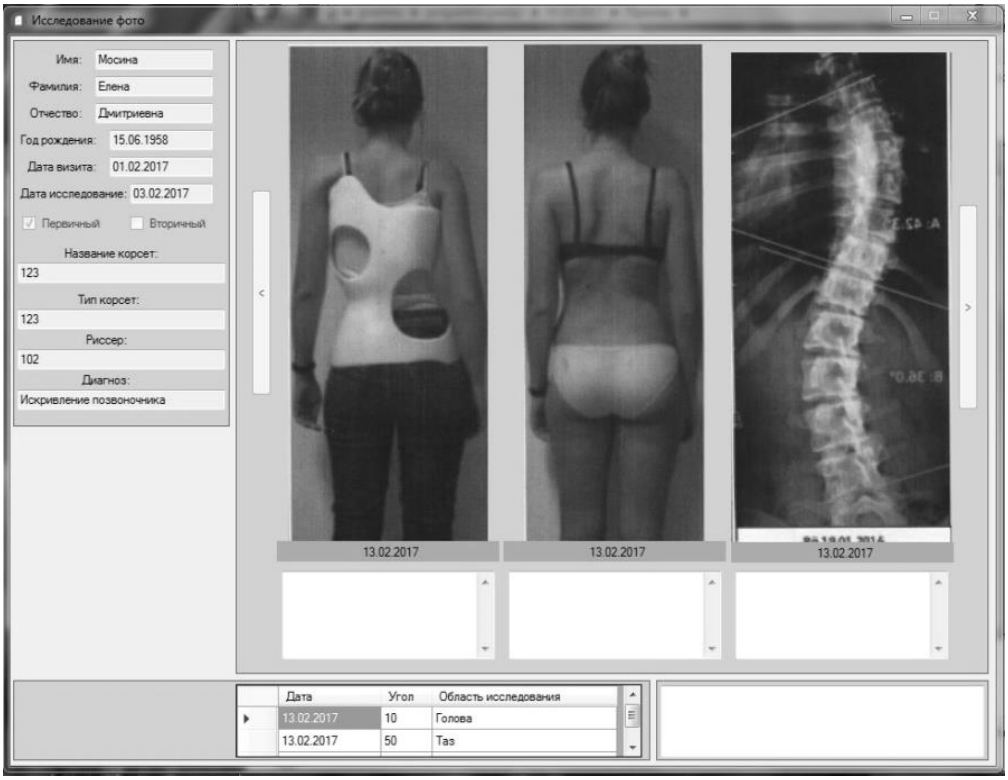


Fig. 3: Window «Research photos»

2.2 MVC pattern in appointment web application

Considering the goal of reducing the labor costs for the development of complex software, it is necessary to use ready-made unified solutions. The use of patterns facilitates communication between developers, allows you to refer to known designs, reduces the number of errors.
The appointment web application was implemented using mvc pattern. Model-View-Controller or MVC is a fundamental pattern that has found application in many technologies, has given development to new technologies and every day makes easier developers life. MVC consists of three parts, which gave it the name. To implement the Web application, the ASP.NET MVC Framework [2], [3], [4] was chosen. It uses the following software components: controllers, models (business logic objects), and views. Web application uses the same database as the Windows application. The data model (Figure 4) was created using the .NET tools.
The classes and objects of the model correspond to the database tables and they provide access to SQL Server database information. .NET tools created the necessary controllers and views for the correct output of data in users' windows.
The created classes need further development, especially for those tasks in which additional data processing and presentation of information in an unconventional form is necessary - for example, the task of displaying the patient recording time. CSS styles and the JavaScript script were also added.

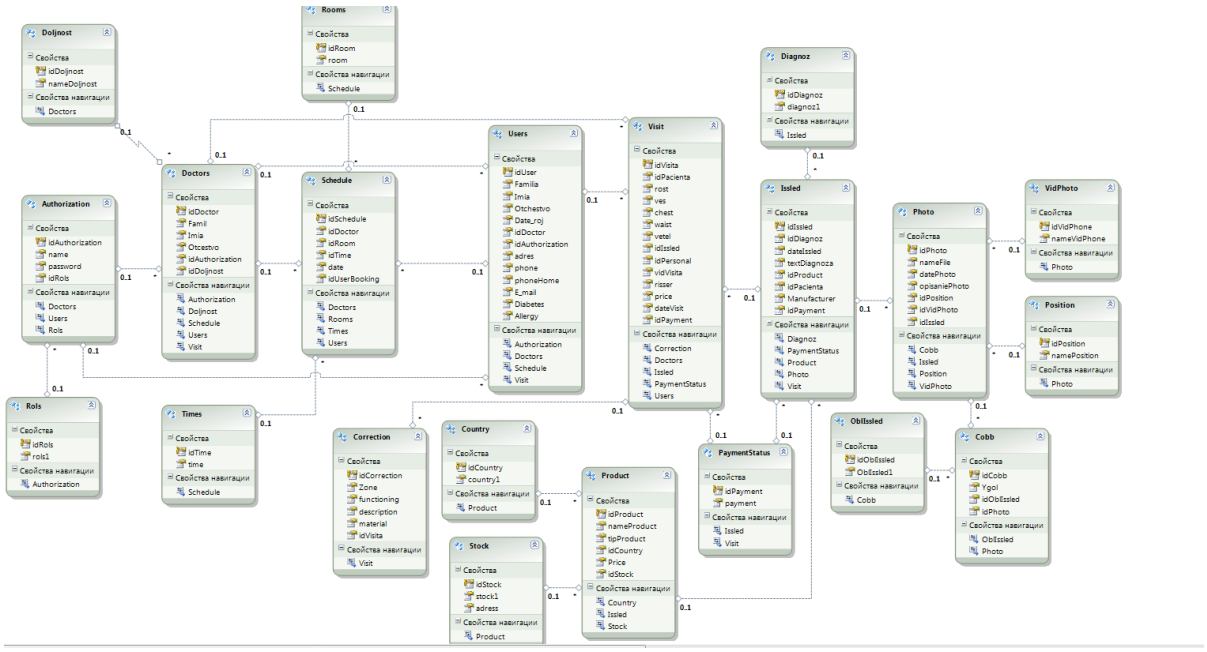


Fig. 4: Data Model

Below is a series of application windows. Figure 5 shows the " Choice of doctor" window.

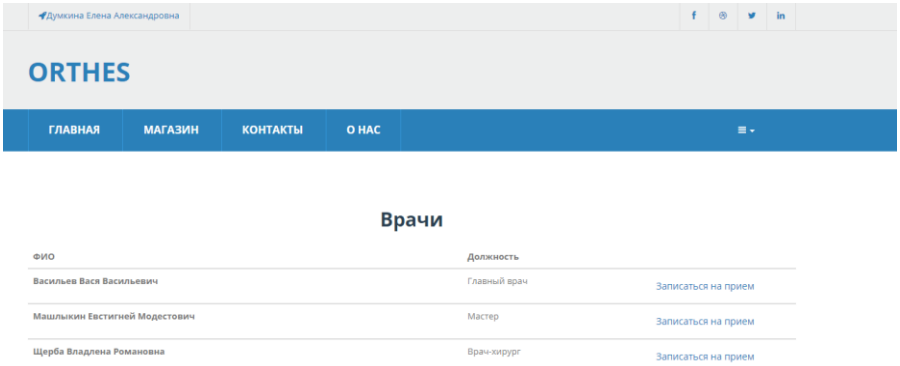


Fig. 5: Window «Choice of doctor »

Figures 6 and 7 demonstrate the process of visit date choosing and visit time selecting. User clicks on the date (Figures 6) and the time selection window (Figures 7) opens.

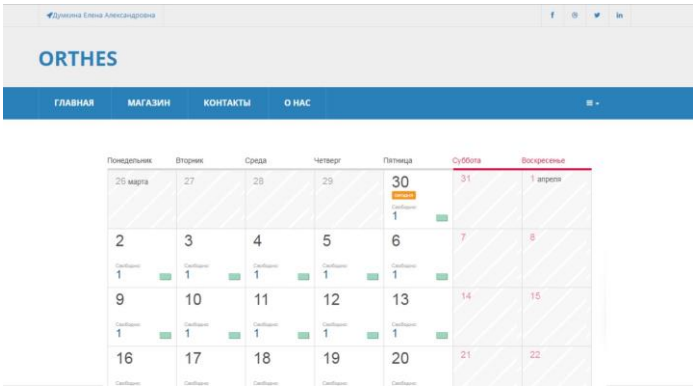


Fig. 6: Window «Choose the date of your visit»

If some other user has already been booked, then our smart system understands this fact and does not allow other user to register for this time. Moreover, it shows other users the choice of this cell in a different color and cancels the processing of the click event.

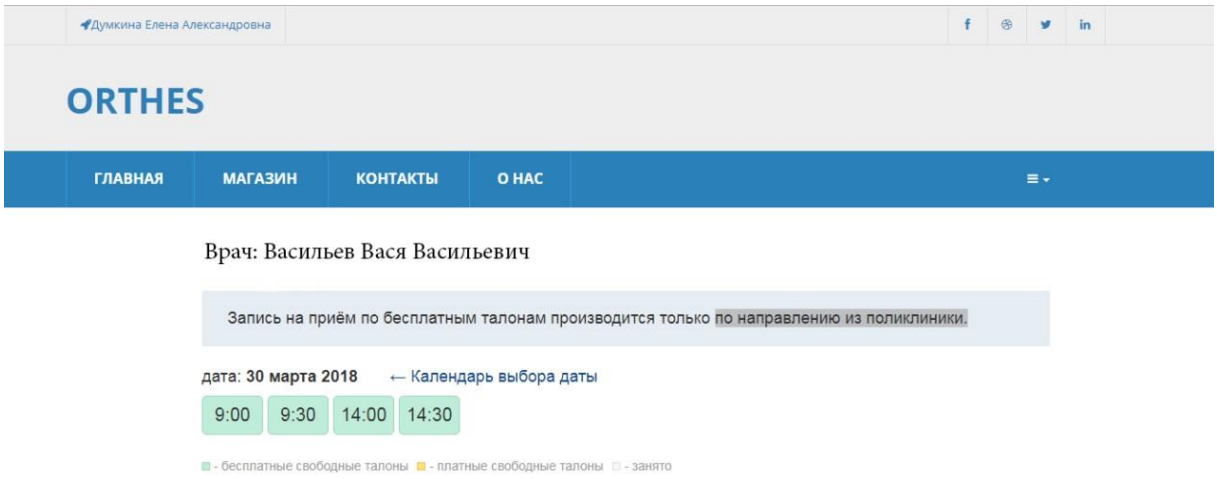


Fig. 7: Window «Select the time by the corresponding date»

The program uses 3 types of accounts (administrator, doctor and user), in which the duties are divided, and each user sees only their own functions. Also in the software implementation there is a personal window for each account type. Figure 8 shows how the doctor's window looks.

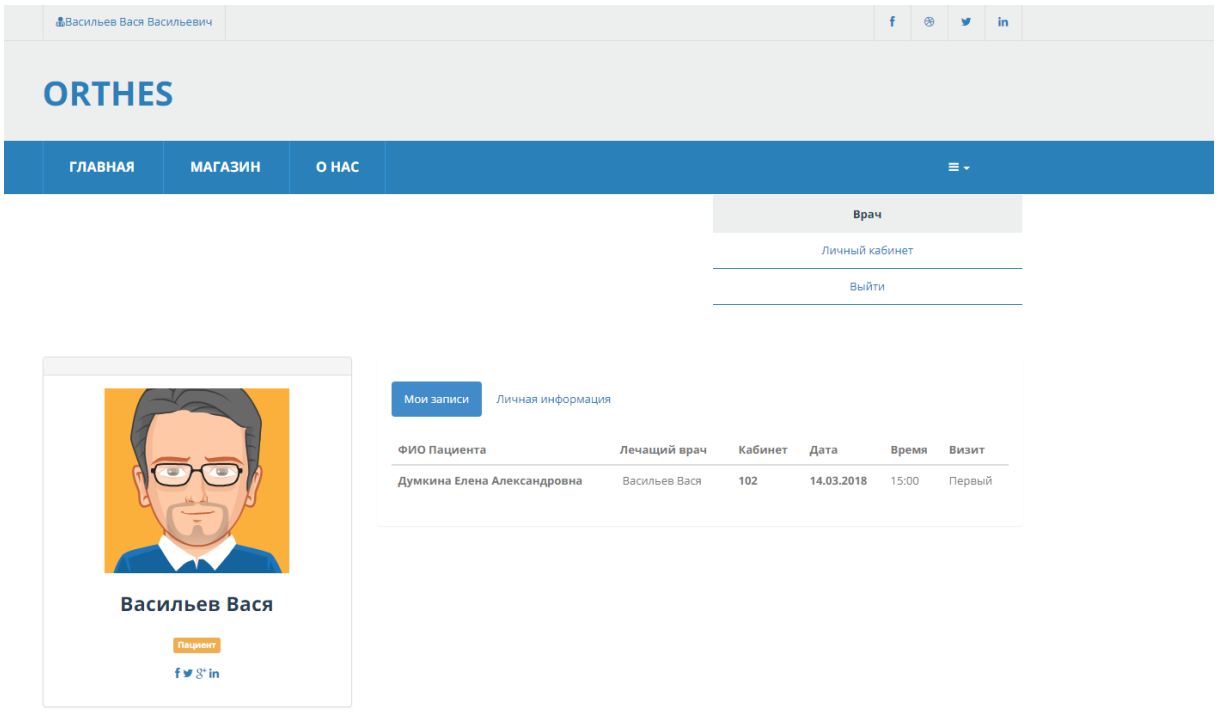


Fig. 8: Doctor window

Figure 9 is a user interface that includes future and past visits and some personal information.

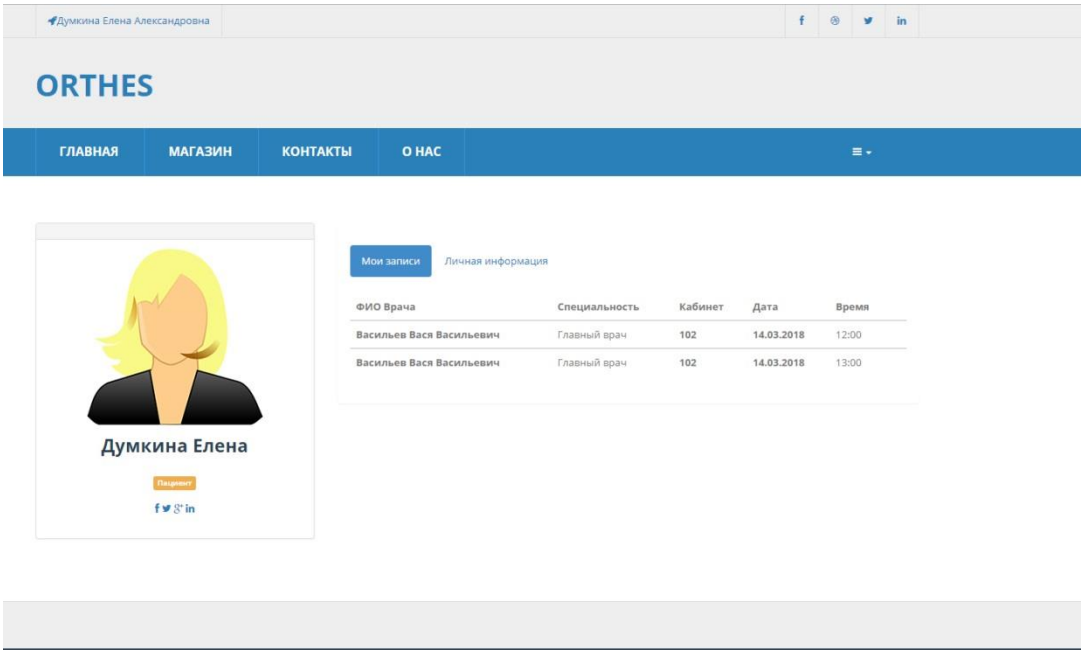


Fig. 9: Patient Window

Thus, we can say that the use of component-oriented programming and the MVC pattern in .NET technology allows reducing the information system development time, improving the development quality, its security and reliability. It should be noted that these technologies do not free the developer from the need to write code.

3 Conclusion

The software for the orthopedic salon was developed in accordance with the customer requirements. It was concluded that the use of component-oriented programming and the MVC pattern in the development of a medical information system reduces development time, improves its reliability and quality.

Acknowledgement: This work was supervised by Prof. PhD *Elena Malysheva* from *Department of Applied Informatics in Economics, Volga State University of Service.*

References

[1] *.NET Framework Guide* https://docs.microsoft.com/en-us/dotnet/framework/
[2] Jon Galloway, Brad Wilson, K. Scott Allen, David Matson *ASP.NET MVC 5*. John Wiley & Sons, Inc., Indianapolis, Indiana, 2014
[3] Dino Esposito *Programming based on Microsoft ASP.NET MVC*. Microsoft Press, 2014.
[4] Adam Freeman, *Pro ASP.NET MVC 5*. Apress, 2013.

Kabanov Andrey ALEXANDROVICH
Volga State University of Service,
Department of Applied Informatics in Economics
4 Gagarin st., Togliatti, Samara region, 445677
RUSSIA
E-mail: kabanovst@yandex.ru

Churkin Alexey ALEXANDROVICH
Volga State University of Service,
Department of Applied Informatics in Economics
4 Gagarin st., Togliatti, Samara region, 445677
RUSSIA
E-mail: Kaf_pive@tolgas.ru

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Automated Car Parking System

Andrei-Timotei Ardelean, Marius Ciocan

Abstract

In order to explore the problem of creating a vehicle that handles parking in a fully autonomous manner a virtual simulation was created. This virtual environment will be used to generate the parking context on which an agent will act upon. In this paper is described a model for an intelligent agent that successfully learns to control a car given realistic sensory input by using a state-of-the-art reinforcement learning algorithm. Considering real car sensory inputs can be easily mapped to simulation values and the environment mechanics can be adapted to match real car controls the results can be reproduced and used on a physical object. In our approach, a neural network is used to compute the preferred action at each step, the actions that will safely get the car to the parking spot. By performing a large number of parking attempts the network trains and learns how to act. Eventually, the agent accurately controls the car and can even exceed human performance in the environment.

1 Introduction

After the article from DeepMind which introduces the DQN agent[1] and its large scale use as presented in Nature (Mnih et al. 2015)[2], the reinforcement learning branch got a new lead for further research and lately attracted a lot of attention. This branch, originally less exploited than the other branches of machine learning (supervised or unsupervised) proves to be of great use by successfully surpassing human results on a variety of tasks and games. This type of algorithms present the advantage that it's not required for the training data to be processed and labeled with the correct answer. The agent learns from his own experience and interaction with the environment based on a feedback reward given by the environment after each step / executed action. Autonomous driving of vehicles is an important field of research, and a challenging task to master. There were and still are several attempts to solving it using different methods. In this paper the problem was approached by using reinforcement learning, moreover, autonomous driving wasn't tackled as a whole, instead autonomous parking is being solved specifically. Starting by presenting previous work in this field we then present our own solution regarding the issue. For that, the environment in which the agent will train is described first, followed by the details of the reinforced learning setup used. We conclude by showing the effectiveness of the approach and directions for further development.

2 Related Work

In attempts based on supervised learning, the agent may learn from the actions taken by a human in a given state. By analyzing professional human behaviour in a great number of scenarios the agent learns a policy to mimic that behaviour. This kind of approach has the disadvantage that acquiring that labeled data can be difficult. Also, there is the possibility that the trained network overfits to the vehicle on which data gathering took place. That would mean it must be trained again for a different car. The

method used in Alvinn (Autonomous Land Vehicle in a Neural Network)[3] creates the training data artificially. Multiple images are generated from actual road images and fed to the algorithm. Remarkable results obtained by using reinforced learning were presented in the paper Deep Reinforcement Learning for Simulated Autonomous Driving[4]. The learning environment used here is the TORCS simulator. The goal is to drive upon a certain route and finish it as fast as possible. As explained in the mentioned paper the agent successfully learns to navigate a variety of test tracks of highly variable shape in TORCS. However, unlike free driving, parking usually requires a strict sequence of maneuvers which may not give a high immediate reward.

3 Context

The algorithm and methods used in our approach was inspired by the remarkable results obtained on several environments by DQN agents + extensions (dueling[6] and double DQN[5]) and, more recently, A3C[7]. Comparing the efficiency of different algorithms is easier since the release of the OpenAi Gym. Gym is a toolkit for developing and comparing reinforcement learning algorithms, a platform which offers several environments such as Atari games, classic control and MuJoCo[8]. The environment created for training car autonomous parking was designed to be compatible with the infrastructure of the existing environments as exposed by the OpenAi Gym. It follows the same API so it can be easily be used to test several existing algorithms and agents. Thus, results presented in this paper can be considered a baseline for further research into self parking cars in the given scenarios.

4 Environment Definition

We created different setups for the virtual simulations in order to study the conditions in which the agent successfully converges to a good policy. The environment consists of the main actor (the car), which has to reach the parking spot (the goal) and obstacles (represented by other parked cars). In the scenarios we built both perpendicular and parallel parking were considered. Also, it can be specified how many *degrees of freedom* are desired, that is, which of the environment actors are randomly placed. Having the parking vehicle always start in the same position helps the algorithm converge much faster although at the cost of some generality. By varying the location of the parking spot and the number of obstacles from one episode to another we ensure that the network doesn't overfit by *learning* a certain path.

4.1 State

The states in which the car is in at a certain moment (frame) is defined by 12 values normalized between [0, 1], or [-1, 1] respectively on values which refer to orientation. The 12 values represent: car speed, car angle, data returned by the 4 proximity sensors situated in the corners of the main vehicle (4 pair of values), positional difference to the goal parking spot (another pair). The environment allows the representation of positional differences either as a vector or in polar coordinates. The polar representation system gave better results in our tests. Since the environment exposes the full problem to the agent and the state transitions satisfy the Markov property it is a discrete-time Markov Decision Process (MDP).

4.2 Action

In a given state the agent can choose between 6 discrete actions: NOTHING, STEER_LEFT, STEER_RIGHT, ACCELERATE, REVERSE(reverse acceleration) and BREAK. Each of these gradually affects the car movement. For example 10 acceleration frames are needed in order to reach full speed. The choice of discrete action space offers a neater control over the vehicle and it's closer to the way a human interacts with it.

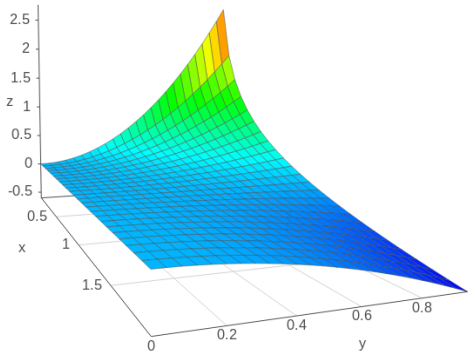


Figure 1: Reward function: -log(x)*y^2, where $x = \frac{distance\,to\,goal}{initial\,distance}$ and $y = \frac{\Delta angle}{\pi}$

4.3 Reward

When an obstacle is touched or the main actor moves out of the simulation active area it is rewarded with a negative value and the episode is finished by setting the 'done' flag. Otherwise, the car receives a reward as a function of the distance to the goal and the relative angle to the perfect parking angle (in order for a lateral parking maneuver to be possible). In order to stimulate the learning progress, several reward functions were considered. The most effective of the functions tested is one that grows exponentially when getting closer to the goal and decreases quadratic to the orientation deficiency (figure 1). Using a linear function with these parameters as a reward function almost always led to a local optima.

5 Strategy

One of the problems when learning from the agent own interaction with the environment arises from the delay in rewards. That is, the reward received when a certain state is reached due to an action being selected, is not based only on that last action. It could be a decision took maybe 10, 50 or 100 frames before that heavily influenced the current state and reward. Thus, an algorithm that learns long-term expected rewards is needed. Q-Learning algorithm indirectly (unlike policy gradient methods) tries to find the optimal policy of taking actions by learning the value of being in a given state, and taking a specific action there *Q(s,a)*. Eventually, the value of that function is an estimate of the efficiency (expected sum of future rewards) of taking an action a given a state s to reach a state s' and following an optimal policy afterwards. Given a fixed policy π, the value associated with an action a in state s is:

$$Q_\pi(s,a) \equiv E[R_1 + \gamma R_2 + \gamma^2 R_3 + ...|S_0 = s, A_0 = a, \pi] \tag{1}$$

R_k is the reward received at step k and $\gamma \in [0,1]$ represents the discount factor which decides how important the possible future rewards are compared to the present reward. "Most interesting problems are too large to learn all action values in all states separately. Instead, we can learn a parameterized value function $Q(s,a;\theta)$ " [5]. So in order to find the optimal policy we will use a neural network. By acting as a function approximator, we can take any number of possible states that can be represented as a vector and learn to map them to Q-values. [10] The weights of the neural network are updated as in:

$$Q(S_t, A_t) \to R_{t+1} + \gamma max_a Q(S_{t+1}, a) \tag{2}$$

With the use of recursion, the approximate dynamic programming relation is defined. In the practical implementation a target network is used, moreover, the selection and evaluation are decoupled as in Double Q-learning [11] in order to prevent overoptimistic estimates [5].

Figure 2: Performance

6 Agent

The main agent implemented is a type of double dueling DQN. The advantage module equation is used for the union between the value and advantage streams as recommended in the original paper[6]

$$Q(s,a;\theta) = V(s;\theta) + A(s,a;\theta) - Average_a A(s,a;\theta) \tag{3}$$

$V(s;\theta)$ indicates how good a state s is to be in whereas $A(s,a;\theta) - Average_a A(s,a;\theta)$ computes how good is action a compared to other actions in this state.

The model used consists of three hidden dense layers, fully connected of 312, 156, 128 neurons respectively. For the experience replay a sequential memory of 100000 experience traces is used. The optimization algorithm is Adam with a fixed learning rate of 0.0003

For choosing an action, considering the *exploration vs exploitation* trade-off problem we created a policy which combines *Max Boltzmann Q Policy*[9] and the classic *Epsilon Greedy Policy*. Therefore, usually the most efficient action is selected. However, with a chance of ψ, an action is chosen with respect to the Boltzmann model and with a chance of τ a completely random action is selected. In the final implementation $\psi = \tau = \frac{\epsilon}{2}$ was stable. ϵ is annealed from 1.0 to 0.25 within the first 20.000 steps and keeps a fixed value of $\epsilon = 0.25$ for the rest of the training. For the test phase ϵ is set to 0.1

7 Results

In our tests the agents successfully learns a good policy for parking the vehicle. In the most generic scenario the parking spot is chosen at random and the number of the obstacles (parked cars) and the distance between them varies between episodes. In this case the car properly parks over 90% of the time. The algorithm converges (figure 2) after about 2500 episodes $\approx$ 1.000.000 steps $\approx$ 2 hours of training on a mainstream architecture (i5 2.2Ghz).

8 Conclusions

Considering the results mentioned above, the proposed task was successfully solved proving the effectiveness of reinforcement learning algorithms in autonomous driving / parking. As for future development, the environment could be extended to support angle parking and combinations of the existent scenarios; that would increase the complexity of the policy required to solve the environment. In order to illustrate the effectiveness of the algorithm it should be seen in perspective. We are working on benchmarking: testing popular agents like Sarsa and A3C and also monitoring human performance.

Figure 3: Preview

Acknowledgement: This work was supervised by Professor *Alexandra Baicoianu*, Lecturer, PhD at Transilvania University of Brasov, Faculty of Mathematics and Informatics, Department of Mathematics and Informatics.

References

[1] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, Martin Riedmiller, *Playing Atari with Deep Reinforcement Learning* arXiv:1312.5602

[2] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, Demis Hassabis *Human-level control through deep reinforcement learning* Nature, 2015/02/25

[3] Dean A. Pomerleau, lvinn: An autonomous land vehicle in a neural network *Advances in Neural Information Processing Systems 1*, 1989

[4] Adithya Ganesh, Joe Charalel, Matthew Das Sarma, Nancy Xu, *Deep Reinforcement Learning for Simulated Autonomous Driving, http://cs229.stanford.edu/proj2016/report/Ganesh-Charalel-DasSarma-Xu-DeepReinforcementLearningForSimulatedAutonomousDriving-report.pdf*

[5] Hado van Hasselt, Arthur Guez, David Silver, *Deep Reinforcement Learning with Double Q-learning* arXiv:1509.06461

[6] Ziyu Wang, Tom Schaul, Matteo Hessel, Hado van Hasselt, Marc Lanctot, Nando de Freitas, *Dueling Network Architectures for Deep Reinforcement Learning* arXiv:1511.06581

[7] Volodymyr Mnih, Adri Puigdomnech Badia, Mehdi Mirza, Alex Graves, Timothy P. Lillicrap, Tim Harley, David Silver, Koray Kavukcuoglu, *Asynchronous Methods for Deep Reinforcement Learning* arXiv:1602.01783

[8] Todorov, E., *MuJoCo: Modeling, Simulation and Visualization of Multi-Joint Dynamics with Contact(ed 1.0).* Roboti Publishing, 2015

[9] Wiering, M. *Explorations in Efficient Reinforcement Learning* PhD thesis, University of Amserdam, Amsterdam, 1999

[10] Arthur Juliani, *Simple Reinforcement Learning with Tensorflow https://medium.com/emergent-future/simple-reinforcement-learning-with-tensorflow-part-0-q-learning-with-tables-and-neural-networks-d195264329d0*

[11] H. van Hasselt, Double Q-learning. *Advances in Neural Information Processing Systems*, 2010

Andrei-Timotei ARDELEAN
Transilvania University of Brasov
Faculty of Mathematics and Informatics
Iuliu Maniu, no 50
ROMANIA
E-mail: *timitoc@yahoo.com*

Marius CIOCAN
Transilvania University of Brasov
Faculty of Mathematics and Informatics
Iuliu Maniu, no 50
ROMANIA
E-mail: *marius.p.ciocan@gmail.com*

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Smart app for your smart home

Maria-Alexandra Băilă

Abstract

Living today in a world where technologies evolves fast, Internet of Things applications established as an easily accessible way to control house environments in order to ease up living, reduce costs and to offer access from anywhere in world. This paper presents a method of acquiring environmental date by several remote sensors and brings data visualization as web service together with strategies for taking action in specific situations. The target is to be able to automatically interpret occurred events and dynamically extrapolate possible events to be happening. Consequently, the application described here is intended to be used as a smart interface for a smart home.

1 Introduction

Why Internet of Things? Today, our daily tasks are more and more dependent of technologies considering that some of us already don't go anywhere before they check their smart watch, sleep monitoring or heart monitoring, don't leave home before they check the weather applications, they don't drive a car without a parking sensor or use seism application where they are informed about possible earthquakes and examples can easily continue. The interesting world of Internet of Things is wide spread due to the numerous application purposes, the interconnection of fields: software, hardware, statistic, understanding of human behaviour and the advantage that almost everyone with the basic programming knowledge can implement an Internet of Things project.

The application has come from a personal need, due to numerous travelling and the thought of being able to come back to a worm and safety place. Therefore the smart home application could give the possibility to measure the temperature, the light, the humidity of the room and also to register and announce if any movement was detected during the owner's absence.

The idea behind was to be able to set the desired temperature inside the room and benefit of costs reduction of the bills due to the higher attention of the energy resources that is being consumed. For safety, the implementation the module consists in movement detection sensor together with a raise alert e-mail to owner of the device, where it specifies the date and time when possible burglar intrusion could have happened.

For a modern wider applicability the smart house application was built as dynamic as possible by the using html responsive, “a responsive web design that makes the application look good by resizing, hiding, shrinking on all devices, no matter the size (tablets, phones, desktops)”.[1]

What is Internet of Things? According to IBM, Internet of Things “is a concept of connecting any device (so long as it has an on/off switch) to the Internet and other connected devices. The IoT is a giant network of connected things and people – all of which collect and share data about the way they are used and about the environment around them”[2]. Or in book *Analytics of Internet of Things* the definition the mentioned subject goes as follows “IoT refers to the ever growing network of physical objects that feature an IP address for the internet connectivity and the communication that occurs between these objects and other Internet enabled devices and systems”[3].

Basically IoT is a interaction of hardware parts such as: microcontroller, sensors, home appliances, where sensors reads a certain state and the information or let’s say data are collected and stored, because one of the major influence in developing this kind of application has been the possibility to store, analyse and interpret a large number of information and use it in order to take a certain decision. Never the less what makes this application an Internet of this is actual the Internet since it offers us the possibility to store all this collection of inputs through internet by using different protocols such as: “HTTP it is also used in machine to machine (M2M) communication, automation […] because it is easily accessible and easy to relate to”[4].

2 Hardware structure and software technologies used for the application

The concept of smart home goes very much in the wireless control zone, multimedia control, high speed data transmission, and its key technologies for the home network compatibility of family controllers and meet information transfer needs. [5] The first step in creating the application is ordering the hardware parts since it is the main core in what means controlling and monitoring the house environment. In order to make the correct decision since a lot of unseen compatibilities can appear, the following receipt needs to be followed:

- decide over the microcontroller,
- take into consideration the number of digital and analogical pins,
- choose according to this input the type of sensors(Digital/Analog),
- take into consideration the frequency that is necessary to monitor the data sent by sensors,
- last but least how accurate the sensor really is.

The microcontroller that was used in the application is an ESP8266 NodeMCU, build by the Chinese Espressif company. It consists of a Tesilica L106 32 bit micro controller unit (MCU) and a Wi-Fi transceiver. It has 11 GPIO pins (General purpose Input/Output pins), and analog input as well. This means that it can be programmed like a normal Arduino or other microcontroller.[6]

Node MCU is on open source IoT platform. It includes firmware which runs on ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on ESP-12 module. The therm „NodeMCU” by default refers to the firmware rather than the development kits. The firmware usus the Lua scripting language. It is based on eLua project and build on Espressif Non-OS SDK for ESP8266. It uses many open source projects, such as lua-cjson and spiffs. [7]

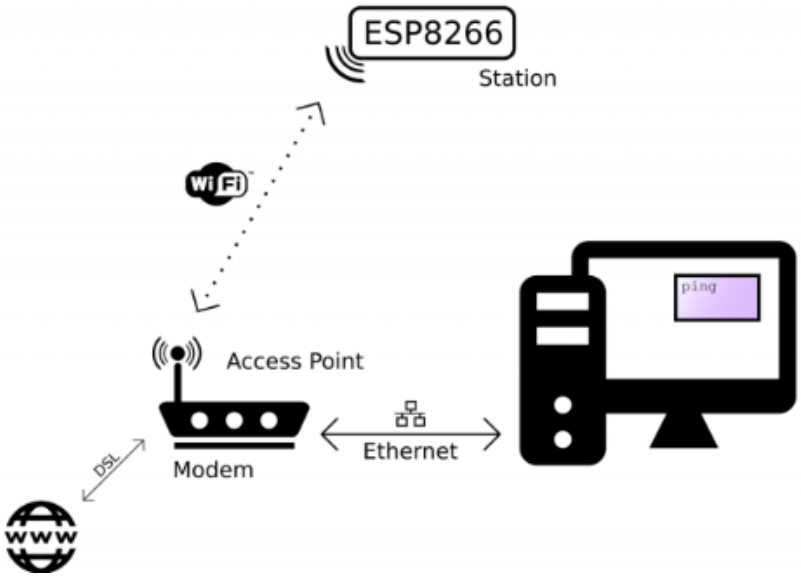


Fig. 1. ESP8266 Wi-Fi Connections[8]

In the image above, the ping command sends packages at the IP address of the ESP8266 micro controller. When ESP receives the package, it sends it back at the one that has sent it. Ping is part of the second TCP/IP layer, more specific the ICMP, Internet Control Message Protocol. As it can be seen in the image, the device with the antenna serves multiple purposes:

- Wi-Fi : other WiFi devices can connect to it be part of the local network
- Router: it routs the IP packages at the corresponding ESP sub-networks , under the Ethernet sub-networks and the router sends back the package from the WiFi sub-network to locate the ESP
- Modem: if the router does not find the address to the local network, the package will be sent to be integrated in modem and will be sent to Internet Supply Service through DSL line.[8]

2.1 Humidity and temperature sensor

The SNS-DHT11 is a digital sensor dedicated collecting temperature and humidity values. The sensor uses a simplified communication: single-bus. The single bus communication it is a single line that transmits the information to micro controller. The device is built also on a master-slave system, where the master is asking and the slave answers, data transmission is performed if high can be sent.[9]

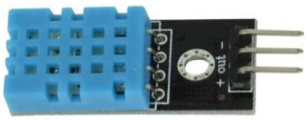


Fig. 2. Temperature and humidity sensor [9]

2.2 Luminosity sensor

Brick luminosity sensor detects the light inside the room, the value can be taken from 0 to 1024, which means that it cannot be calibrated, which means that the value of the light from the room

it cannot be taken directly, it can only be obtained a numerical value directly proportionate with the level of light, no unity value - luxeni).[10]

Fig. 3. Luminosity sensor [12]

2.3 Motion sensor

PIR (Passive Infrared) sensors allows to sense motion, almost always used to detect whether a human has moved in or out of the sensor range. PIRs are basically made of a pyroelectric sensor (which can be see below as the round metal can with a rectangular crystal centre), which can detect levels of infrared radiation. Everything emits some low level radiation, and the reason for that is that we are looking to detect motion (change) not average IR levels. The two halves are wired up so that they cancel each other out. If one half sees more or less IR radiation than other, the output will swing high or low. [11]

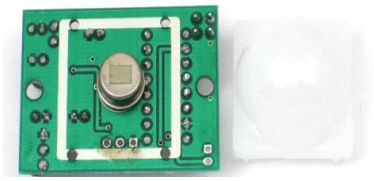


Fig. 4. PIR motion sensor [11]

2.4 Software technologies used for the application

The implementation of the software is performed with Arduino Software (IDE), which is an open source company that not only micro-controllers but also the software which is meant to for flashing the MCUs. The scope of the Integration Development Environment (IDE) was to introduce the programming also the world of the artists or to those people who are not very familiar with coding. [12]

Also for storing data sensors XAMPP was used, which stands for a Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Pearls (P). Basically it is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes. [13]

In order to create, connect or for the possibility to create queries for selecting, updating or inserting information from database tables, MySQL together with PHP was an easier choice for me to make since it is one of the most popular systems and easy-to-work scripts. MySQL is the de-facto standard database system for web sites with huge volumes of both data and end-users (like Facebook, Twitter and Wikipedia). Another great thing about MySQL is that it can be scaled down to support embedded database applications. [14]

3 Presentation of the application

Smart application for smart home starts with the an introduction page where hints of its purpose appears more or less like an introspection of what possibilities the app could offer, such as checking the status of your house and never the less to take smart decisions for your benefit.

Moreover, one of the main reason for implementing the application was to offer the possibility of changing the room temperature especially in the freezing days of winter, when everybody on the way home wants to find the cosiness of a warming and pleasant home. Therefore based on the room temperature there is the possibility to set the desired room temperature and nevertheless the central station to take the decision for the owner in order to start the central station and heat the place up.

Fig. 5. Start page of the application

3.1 Sensor reading page

The first page of the application shows the status of the information read by sensors and sent to microcontroller, than added in the database with the help of PHP scripts. Basically the status of movement is referred in the first section, with the date and hour of movement detection, informing also the user via e-mail of the potential intrusion with an e-mail alert .

Fig. 6. Sensors status page- movement information

The second section of the application is dedicate to the sensor data recordings, where the originality of it comes from the three gauges that have the scope to show the current information and also to indicate the normal values that must be inside a room for the human body comfort, marked also in the green areas. The yellow is an indicator, indicating that the home environment should be changed in order to avoid the possible discomfort. And the last but not least, considering that the health could be endangered there also with red marks, if the values are really abnormal. This off course were used only for testing purposes by stressing the sensors with high heat or high light from flashlight etc.

When the temperature entered in the web page form is higher that the sensor measurement, the central station is basically turned off. This acts like a regular thermostat but a much more efficient one due to its accessibility from anywhere in the world, no matter the time and place.

Off course the scope of the application so for was purely demonstrative and this is why the temperature station is just simulated by a led which is connected through wires to the ESP8266 micro-controller. The led will turn on and off exactly like the heat station, as well as it provides flexibility and safety during development phase, controlling the actual station while testing, debugging and implementation would be a bad approach.

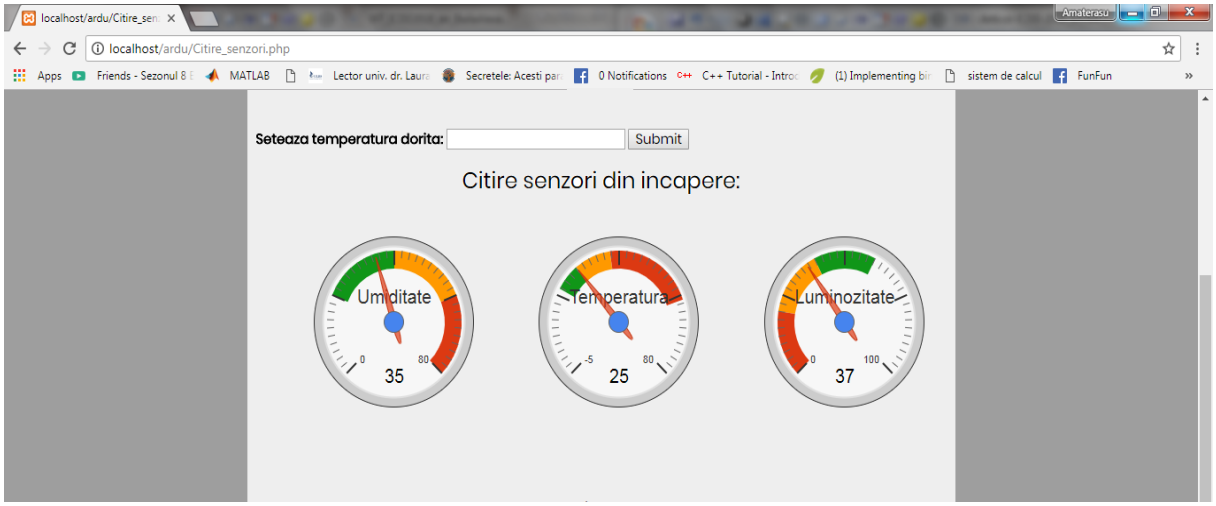


Fig. 7. Sensors status page – Form for setting the desired temperature and gauges for humidity, temperature and luminosity

In the last section of the page the scope was to be able to reduce the cost of the bills by having an overall image of how much time the temperature station (led) stays on and hoe much time is stays off. In this manner, it is possible for the owner to make a better estimation of consumed methane gas during autumn, winter and spring time.

Fig. 8. Sensors status page – Temperature station on/off graph

3.2 Sensor status and history page

Since an image is more easily assimilated to human than an amalgam of numbers and information, the next page of the application is dedicated only to the graphics. As indicated, the reason is untaught: because they are easily understood by everyone, simpler to be evaluated as a status and they are very suggestive due to the visualisation of data.

The first graph is used to indicate the status of the last reading of the sensors joined with the last requested temperature by the user. It is a fully dynamic chart that comes as an glance overview of the last entered values.

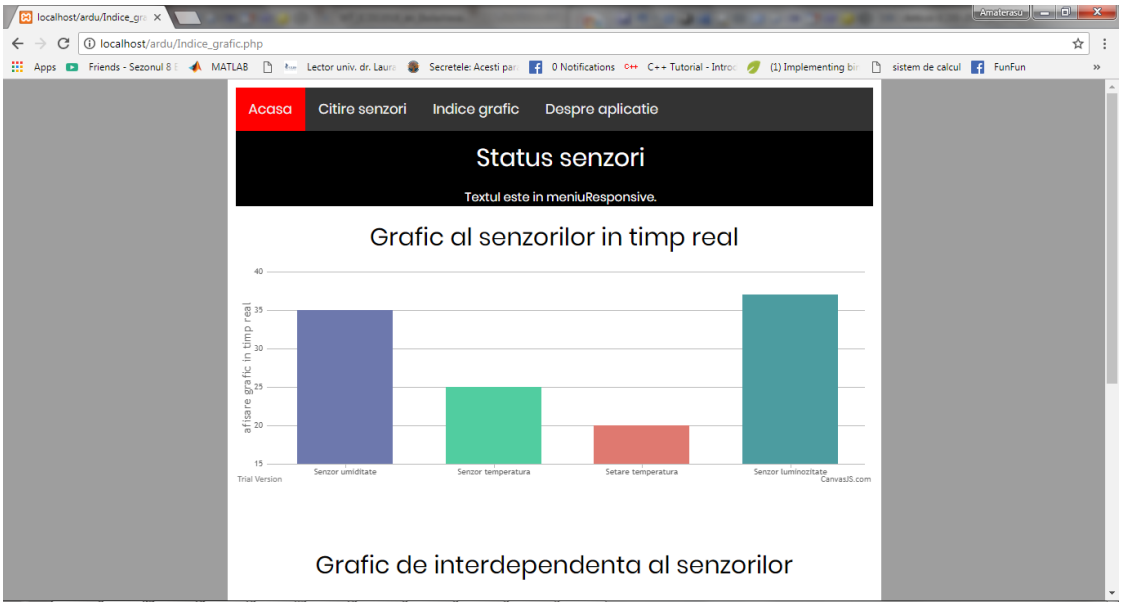


Fig. 9. Graphic view page – Live reading of sensors and of the set temperature

Next chart is a much more complex one, because it shows the information read by sensors among time and indicates how deeply the interfere with each other. In this way, the interdependency of the values can be easily observed and conclusions can be taken. As a small interpretation, it is possible to visualize that most of the time when the light registers higher values also the room temperature is higher and when the humidity shows lower numbers the temperature is on the peak. Analysis of temperature, humidity and day time dependency is thus available for the user.

Fig. 10. Graphic view page – Interdependency sensors chart

3.3 About application page

The last page of the Smart home application is dedicated to additional explanations on what makes this project work and how it can really an Internet of things application due to the decision making without any help of the user

Fig. 11. About application page – explained interconnection of the information via WiFi

Also a table with all Data Base recordings for sensor readings can be accessed by the user.

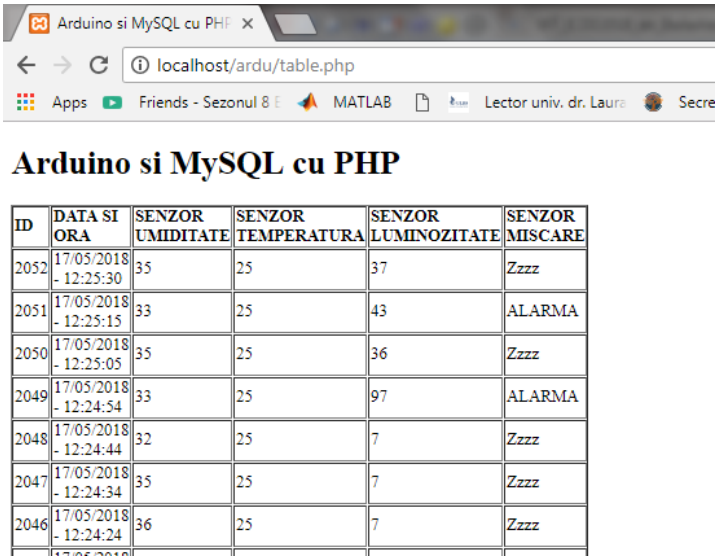


Fig 12. About application page – data base recordings table

4 Conclusions

Smart app for smart home came as a need and a challenge to see how something that a decade or two ago would have seem impossible without high costs and development time and knowledge. The availability of Development Kits allows you to focus on the application scope rather than technical difficulties, is not Plug and Play but at least you do not have to cope with "reinventing the wheel". Considering the information, hardware and software available today, the trend for automation and autonomous systems is becoming more transparent and accessible for students and developers, only our imagination is hold back for what can be created.

In this journey of implementing the application it was very clear at first because there was only the idea of controlling the temperature - a simple Html file that was stored in Arduino; without a DataBase, PHP, Json, gauges, graphs or a dedicated web page. Still, seeing what other sensors are available, the intentions started to change and it was just a question of „how" and not „if". It is understandable that improvement can be made. For starter, the next step would be to make the concept of home much more autonomous, just like implementation of the temperature station, based on the temperature sensor by connecting it to the air conditioner, also based the humidity to create the possibility of having a reaction to the a room humidification, by spraying the room with water. As far as it goes for the movement detection, next step would be to a install a camera and send images of the place where high infrared is picked.

More or less, no matter how large the applicability this project offers, by this presentation of the application implementation, it can be observed that improvements can be made in our lives in order to upgrade our living, to have a closer look on our daily consumption and also to protect the resources that our planet offers us with more and more sacrifice in our benefit and it should be also our scope to pay more attention in this direction by being more responsible.

Acknowledgment: The results presented here are part of the research project of my Bachelor Thesis of an on-going research and it has been supported by University Lecturer PhD Ralf Fabian.

References

[1] HTML Responsive Web Design. https://www.w3schools.com/html/html_responsive.asp (retrieved 12.05.2018).
[2] What is Internet of Things? https://www.ibm.com/blogs/internet-of-things/what-is-the-iot/ (retrieved 12.05.2018).
[3] A. Mineer *Analytics for the Internet of Things (IoT).* Birmingham: UK, p. 69, July 2017.
[4] P. WASHER, P. SENEVIRATNE, B. RUSSEL, D. VAN DUREN. *IoT: Building Arduino-Based Projects.* Birmingham: UK, p. 27, 2016.
[5] ZHEN-YA LIU.ASRI, *Article: Hardware Design of Smart Home System Based on zigBee Wireless Sensor Network,* Conference of Sports Engineering and Computer Science, p. 1, SECS 2014.
[6] A Beginner's Guide to the ESP8266. https://tttapa.github.io/ESP8266/Chap01%20-%20ESP8266.html (retrieved 12.05.2018).
[7] NodeMCU. https://en.wikipedia.org/wiki/NodeMCU (retrieved 11.06.2018).
[8] Establishing a Wi-Fi connection. https://tttapa.github.io/ESP8266/Chap07%20-%20Wi-Fi%20Connections.html (retrieved 12.05.2018).
[9] Temperature and humidity sensor. https://www.robofun.ro/senzor-temperatura-umiditate-sns-dh11 (retrieved 11.05.2018).
[10] Luminosity sensor. https://www.robofun.ro/senzor-temperatura-umiditate-sns-dh11 (retrieved 10.05.2018).
[11] Adafruit, PIR passive infrared proximity motion sensor. https://cdn-learn.adafruit.com/downloads/pdf/pir-passive-infrared-proximity-motion-sensor.pdf (retrieved 12.05.2018).
[12] Wikipedia: Arduino. https://ro.wikipedia.org/wiki/Arduino (retrieved 10.05.2018).
[13] Wikipedia: XAMPP. https://en.wikipedia.org/wiki/XAMPP (retrieved 10.05.2018).
[14] Php MySQL. https://www.w3schools.com/php/php_mysql_intro.asp (retrieved 0.05.2018).

Maria-Alexandra BĂILĂ
Univ. "Lucian Blaga" Sibiu, Faculty of Sciences
Informatics
Street Dr. Ion Ratiu no 5-7, 550012 Sibiu
ROMANIA
E-mail: baila_alexandra29@yahoo.com

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Prediction of project task execution time by clustering and classification methods

Nataliya Bobrovskaya

Abstract

This article examines execution time prediction and Story Points evaluation on the basis of numerical and text attributes. To assess the significance of words in the set , the metric Tf-idf was used. Clusters of tasks in terms of task execution time and Story Points are determined using the T-SNE algorithm. To classify Story Points the logistic regression algorithm was used.

1 Introduction

When creating software using flexible methodologies developers are constantly faced with incomplete and changing situation: the sets of tasks are changing, tasks can appear or disappear, there may be no task executor data or links with other tasks, some parameters appear during the execution of the task, some information is strictly confidential and access to it is strictly limited.
One of the main parameters is the task execution time. Typically, the estimated task execution time is determined by the size of the task. The task scope is determined at the evaluation stage, which is carried out either by the whole team or experienced developers.
The purpose of this study is to estimate the new project task execution time on the basis of previous projects data. Task data is taken in the financial company IT-division from the jira task tracking system. The sample size was about 2000 tasks. To solve the problem, the software was developed in JavaScript (including NPM jira-client, moment, Az libraries) and Python (numpy, pandas, matplotlib, sklearn, scipy libraries). Clustering, Tf-idf metric, t - SNE machine learning algorithm, logistic regression algorithm were used.

2 Prediction of project task execution time

2.1 Project life cycle

The project is a set of tasks that must be performed to implement some functionality. Figure 1 shows the typical life cycle of IT project.

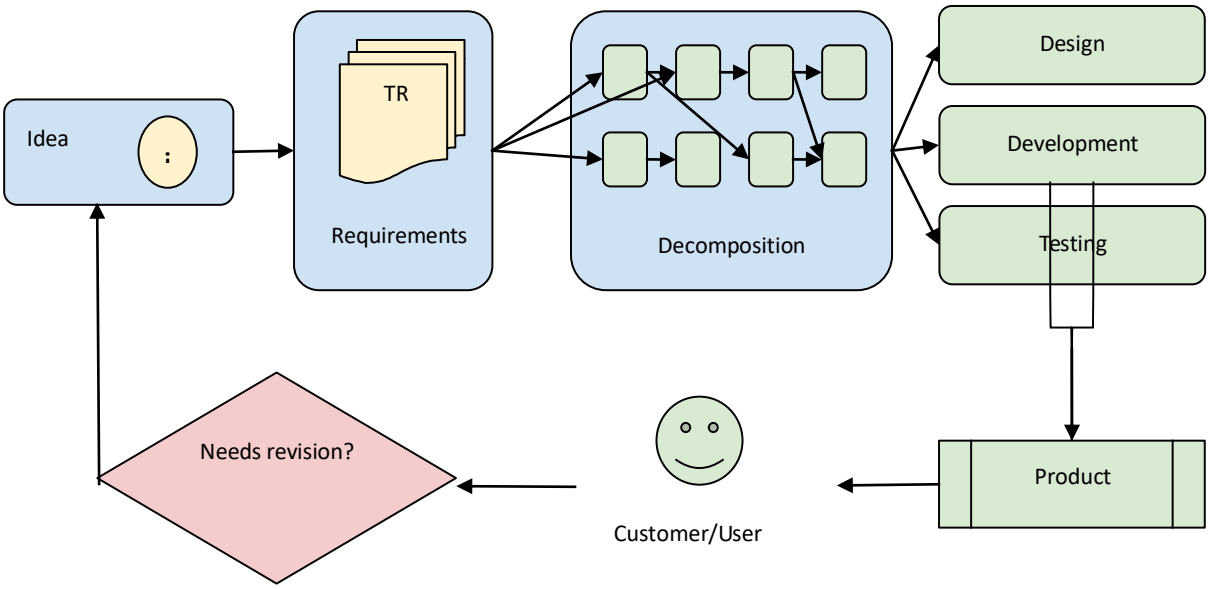


Fig. 1: Project life cycle

At the beginning of development, technical requirements TR based on the functionality realization idea are formed. This is followed by the decomposition of requirements into tasks. The project execution time is the time starting from the project first task start and ending with the last task completion (including its testing). The finished product is transferred for use to the customer/user. If a project needs revision (new functionality), a new project with new technical requirements is created.

2.2 Data composition and preparation of the dataset

The task includes information such as the title, description, start time, end time, sprint name, epic title, related tasks, task state changes, task type and the name of the team.
Sprint is a time- limited iteration of the project, which includes a set of tasks that must be performed in a fixed time. Epic is a business problem statement (User Story) that requires decomposition into technical problems. Related tasks information includes the names of the tasks that the current task is associated with and the relationship type. Task execution time starts from the moment of “in progress” state transition and ends with the time of “done” state transition without taking into account holidays and weekends. The size of the problem can be estimated in Story Point. Story point (SP) [2,3] is a sequence of numbers, most often Fibonacci numbers, which allows for a comparative evaluation.
According to the available information about the tasks, a dataset of attributes (metrics) was formed Data must be converted to a set of numeric attributes. The sprints number, the epics number, the related tasks number are numeric attributes, the type task and the team name are treated as categorical attributes with a value of 0 or 1 in the appropriate column in the dataset. For

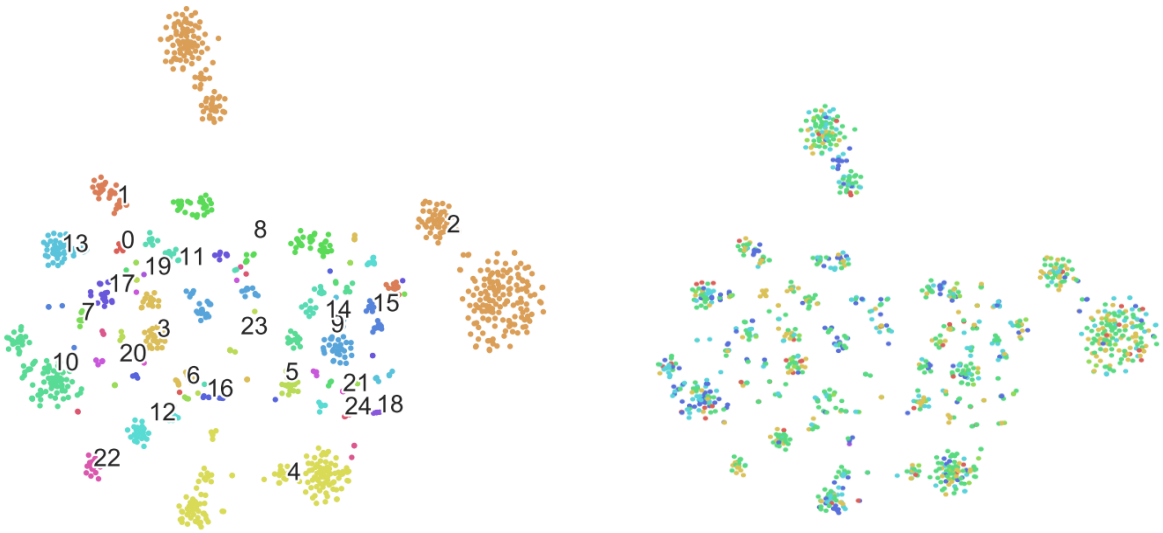
text attributes text processing was carried out, the dictionary of significant words was built [1] with the help of the dictionary was formed datasets. Tfidf feature [5] was chosen as the metric.

2.3 Clustering

The next stage of the study was to separate clusters with similar characteristics from the existing set of tasks. T-SN (t-distributed stochastic neighbor embedding) algorithm [6] is used for clustering. The Python library sklearn was used for experiments.

Experiment 1. Clustering with categorical attributes

Clustering was carried out taking into account all the attributes, including categorical attributes of the team name and task type. The results of the experiment are shown in Figure 2, each task corresponds to one point.



(a) tasks of the different teams are painted different colors

(b) tasks with different execution times are colored in different colors

Fig. 2: Clustering with categorical attributes

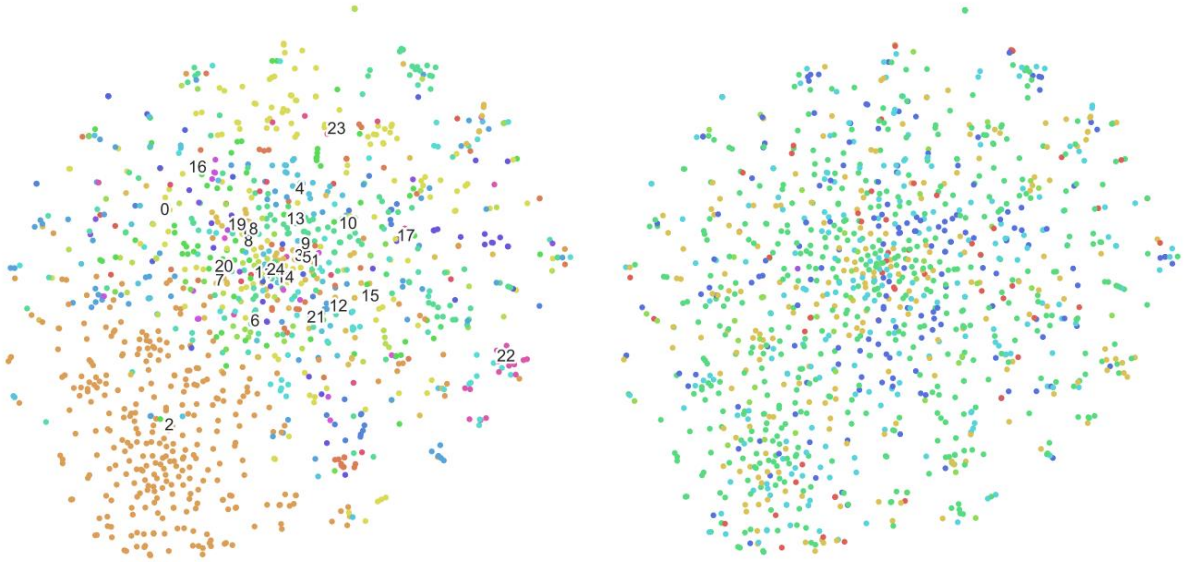
Figure 2(a) shows different teams tasks in different colors. It can be seen that clustering is carried out mainly on the basis of the team names and the task type. Team numbers are placed in the center of the selected clusters, splitting within teams corresponds to different tasks types. Figure 2(b) shows tasks with different execution times in different colors. The colors change according to the colors of the rainbow. Tasks with the shortest time (several hours) are shown in red. A task that has been done for a long time (about a month) is shown in purple. When comparing Fig. 2 (a) and Fig. 2 (b), it can be seen that the tasks of the # 10 team take a lot of time, and the # 22, # 12, # 6 teams have mostly short duration tasks. The speed of tasks can also be different because of the different expertise of each team.

Experiment 2. Clustering without regard to categorical attributes

In the sample only the quantitative characteristics and the data formed from the text were left. The experiment results are shown in Figure 3.

In Figure 3 (a), different team tasks are colored in different colors. Some team task sets, for example, #2, #22, #23, #16, #0, #17, #15 are noticeable apart from the rest sets.

Although the team attribute were not added in the sample, the sample is localized by team. So, the remaining attributes implicitly connect the tasks performed by one team. For example, team #2 can use the words in the description of the task that are not found in tasks of other teams. Figure 3(b) is similar to Figure 3 (a), only the run-time color is used instead of the team color. As in Figure 2 (b), the red color indicates the ones that spent the least time.



(a) tasks of the different teams are painted different colors

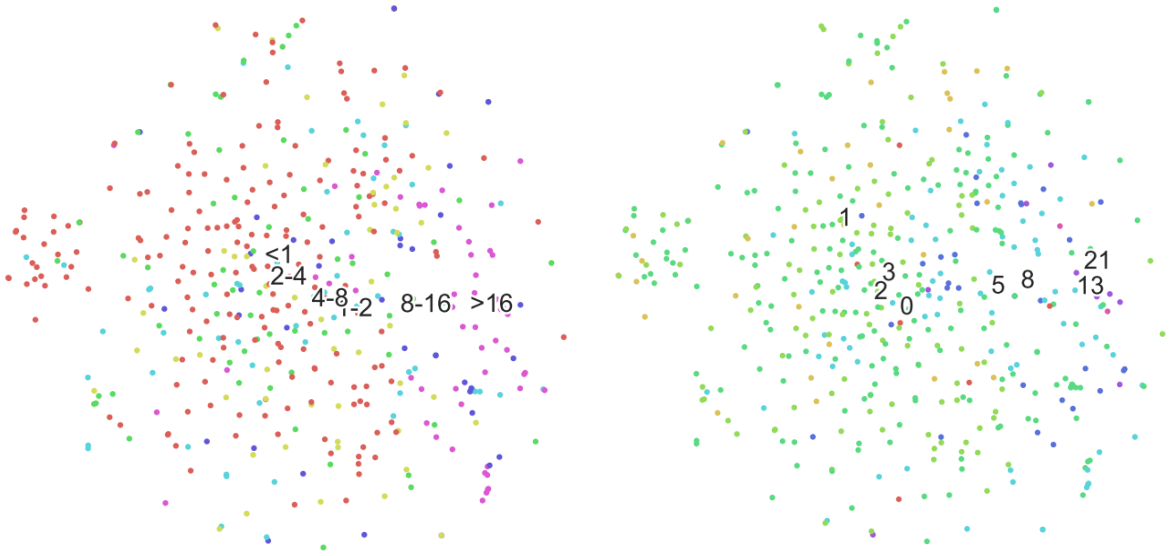
(b) tasks with different execution times are colored in different colors

Fig. 3: Clustering without regard to categorical features

In this case it is difficult to draw any conclusions on time. Perhaps, clusters are small and there are a lot of them.

Experiment 3. Analysis of tasks of one team taking into account Story Point

The Story Point (SP) metric was not used in the sample for all teams, because it is not used by all teams. This metric is used in #2 team set, so #2 team set was considered additionally. #2 team set size is 467 rows. The results of #2 team tasks clustering are shown in Figure 4.



(a) task execution time is highlighted in different color and the number of days signed

(b) SP is highlighted in different color and the number of SP signed

Fig. 4: Story Point-based clustering

In Figure 4(a), the task execution time is highlighted and the number of days is signed. In Figure 4 (b), Story Point (SP) is highlighted and the number of SP is signed. When comparing two pictures, you will notice that

- Story Points with close values are located near each other, and large Story Points are separated from small,
- time is due to SP, but not linearly,
- there is no explicit division into clusters.

To explore the relationship of SP and the execution time of tasks were built radar chart (Figure 5).

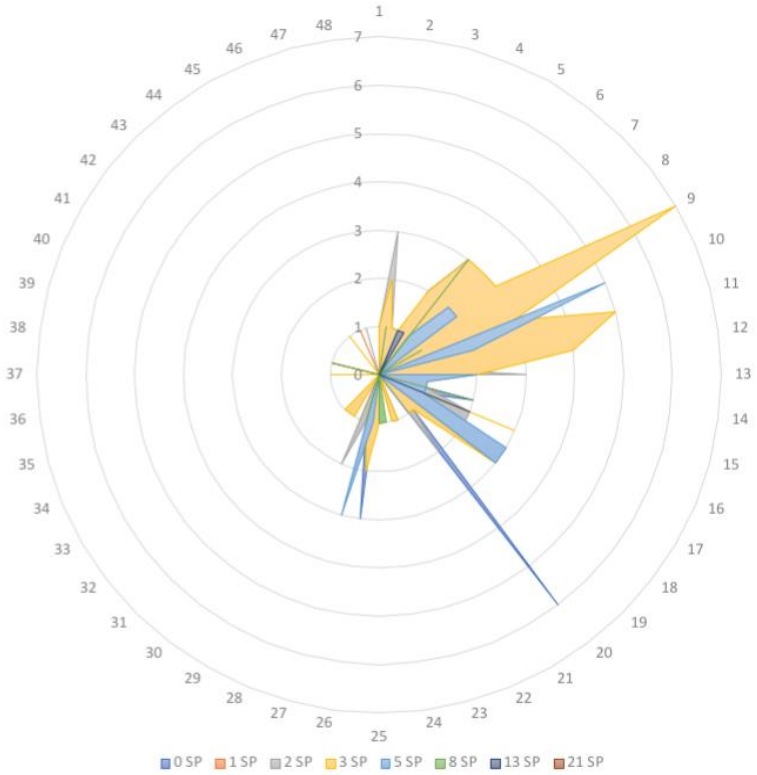


Fig. 5: The amount of SP relative to the execution time of a task

From the radar chart, you can see that the tasks evaluated in 1 SP can be performed both 9 days and 45. This difference in the estimate of task size (SP) and execution time may be due to the fact that:
- one task depends on another, maybe even being developed by another team,
- SP is incorrect,
- many resources were used to successfully solve the problem, so task ended faster,
- task execution time may depend on different risks that are not taken into account when task starting.

It was suggested that tasks volume can be predicted from the task description and other metrics. To confirm this hypothesis, SP task classification algorithm was determined. This algorithm estimated the volume of new tasks, which made it possible to predict task execution time.

2.4 Classification

Classification by SP within #2 team task dataset was performed using logistic regression algorithm [7]. The Python Logistic Regression algorithm from sklearn library was used.

Experiment 4. Classification using logistic regression

The existing dataset was divided several times into training (90%) and test sample (10%). The model was trained and predictions accuracy was determined. For different test samples predictions accuracy was from 30 to 60%. The results of one of the test samples are shown in Figure 6.

real	pred	days	prob:[0	1	2	3	5	8	13	21]
3	3	3	[0.02,	0.02,	0.17,	0.48,	0.25,	0.04,	0.01,	0.01]
3	5	25	[0.05,	0.01,	0.05,	0.37,	0.37,	0.11,	0.02,	0.01]
13	5	45	[0.01,	0.01,	0.02,	0.27,	0.46,	0.19,	0.03,	0.02]
13	3	9	[0.01,	0.02,	0.02,	0.44,	0.27,	0.17,	0.04,	0.02]
5	3	7	[0.03,	0.03,	0.1,	0.34,	0.32,	0.13,	0.03,	0.02]
8	3	1	[0.04,	0.03,	0.24,	0.48,	0.1,	0.08,	0.01,	0.01]
3	3	0	[0.01,	0.08,	0.28,	0.49,	0.09,	0.03,	0.01,	0.01]

Fig. 6: The results of logistic regression classification

The first column of the table is the actual number of SP, the second column is SP number predicted by the algorithm, the third column is the number of days spent on the task, and the fourth column is probability for each SP predicted by the logistic regression algorithm. It can be seen that 13 SP tasks are performed 9 and 45 days, and the algorithm distinguishes these two tasks into different SP: 5 SP for 45 days and 3 SP for 9 days. You can also see examples when a real 5 SP task runs for 7 days and an 8 SP task runs for 1 day. This is due to the fact that SP does not have absolute value, it is a comparative measure - tasks with similar volume have the same SP
Figure 7 shows the relationship between the task execution time and its SP. The x axis is task execution time, the y axis is SP. The real task SP predicted by the algorithm is marked with green dots. We can see that the red points are located more linearly in time, and the green points are characterized by emissions. For example, the task estimated at 13 SP was performed less than 10 days.

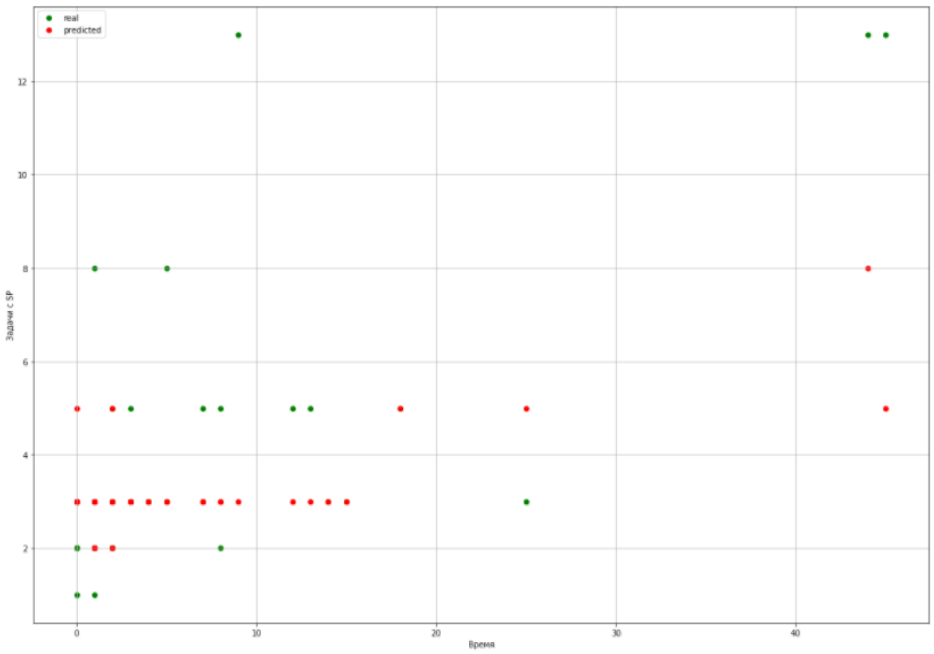


Fig. 7: The relationship between task execution time and its SP

The classification algorithm predicts SP based on existing tasks comparison. It was suggested that removing emissions from the sample would improve algorithm prediction

Experiment 5. The classification of the samples without emissions

For each SP, a graph of the distribution of the task execution time (in hours) was built, based on the assumption that the distribution is normal [4]. The result for some SP is shown in Figure 8.

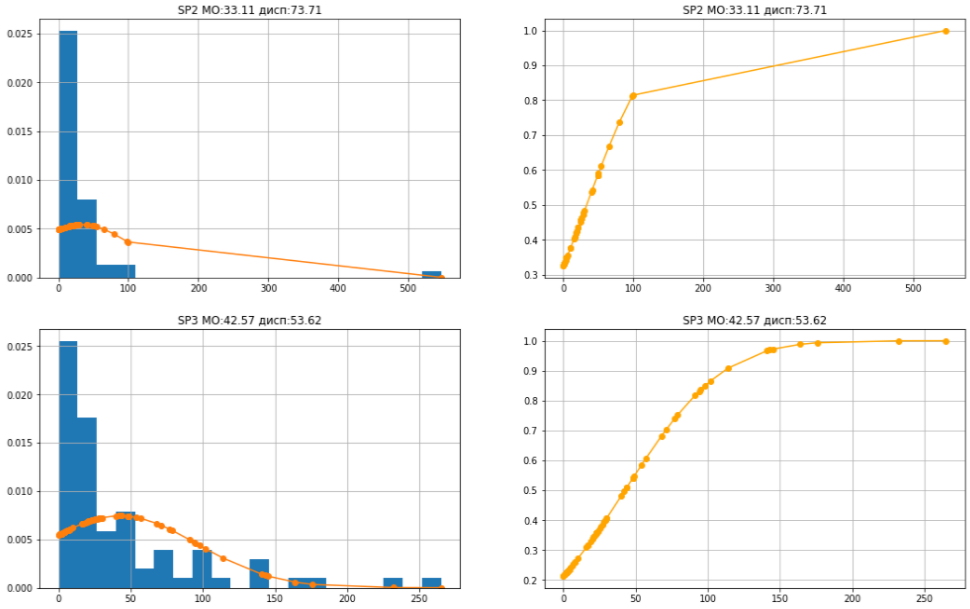


Fig. 8: Tasks execution time distribution

For each SP the interval of reference tasks was set, and only such tasks were taken into account in the classification. The diagram of task execution time distribution for some SP filtered samples is shown in Figure 9.

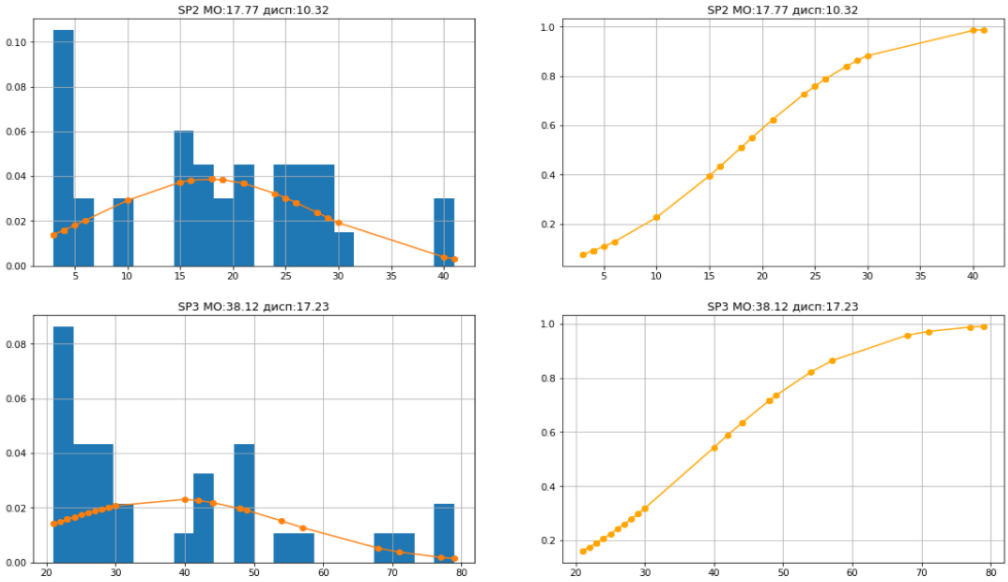


Fig. 9: Tasks execution time distribution by filtered data

A logistic regression classification algorithm using filtered tasks was constructed. Figure 10 shows the results of a test sample classification.

predicted	prob	real
3.0	[0.1, 0.33, 0.46, 0.07, 0.04]	5.0
3.0	[0.03, 0.19, 0.48, 0.19, 0.11]	2.0
3.0	[0.1, 0.37, 0.41, 0.09, 0.03]	3.0
3.0	[0.06, 0.38, 0.47, 0.06, 0.03]	1.0
2.0	[0.06, 0.64, 0.26, 0.02, 0.03]	2.0
2.0	[0.05, 0.53, 0.17, 0.16, 0.1]	1.0
2.0	[0.11, 0.49, 0.31, 0.06, 0.03]	2.0
3.0	[0.29, 0.27, 0.4, 0.03, 0.02]	3.0
2.0	[0.12, 0.68, 0.14, 0.03, 0.02]	2.0
2.0	[0.11, 0.38, 0.25, 0.2, 0.06]	2.0

Fig. 10: Results of a test sample classification according to the modified logistic regression algorithm

The first column shows the SP predicted by the algorithm, the second column shows probability predicted by the logistic regression algorithm for each SP (1SP, 2SP, 3SP, 5SP, 8SP), and the third column shows the actual number of SP. Classification accuracy is calculated as the ratio of the number of correctly classified objects to the number of all objects. In the test sample shown in Figure 10, classification accuracy was 60%, the rest of the test samples classification accuracy varies from 40% to 80%. The assumption of the normal distribution of the task execution time and the SP volume make it possible to estimate task execution time and the interval in which it can be located.

3 Implementation

Javascript libraries were used for preliminary information processing and statistical characteristics obtaining. The Az Javascript library was used to process text attributes and build a dictionary. Az library is based on the pymorphy2 Python library and serves for natural language processing (NLP) in Russian. The text was split into words (tokens) using the Az.Tokens module. The Az.Morph module methods were used for morphological analysis.
Using the ramda library, functions pipeline was created. In the pipeline, task descriptions were processed, a vocabulary was obtained, and task descriptions were translated into vectors.

```
const tokenizationTfIdfTest = (params_df, prefix = '') => {
  return R.pipe(
    R.map(getTokens),
    R.map(filterByTypes(needTypes)),
    R.map(filterByMorph(noNeedGrammems)),
    readDictionaryFromFile(`./dictionary-${prefix}.json`),
    createDocsDictionaries,
    getTfIdf(prefix)
  );
};
```

Using the ramda library, the pipeline was also created for the entire processing - data reading, data processing, filtering, normalizing, writing data sets to a file.
The stats-lite library was used to obtain a mathematical expectation and variance, and the distributions-normal library was used to obtain Story points normal distribution

```
function getSPDistribution(data) {
  const mean = stats.mean(data);
  const variance = stats.variance(data);
  return distributions-normal()
     .mean(mean)
     .variance(variance);
}
```

Python and its libraries were used to prepare data and conduct experiments.
The pandas library was used to work with data sets, it conveniently presents them in the form of tables, provides methods for columns adding (dataset ['newColumn'] = newColumn), columns deleting (dataset.drop ('oldColumn', 1)), filtering (dataset [(dataset.column == 'value')]) and others.
For the clustering and classification, the classes of scikit-learn (SKlearn) machine learning algorithms library were used. Data processing using the t-SNE library algorithm is as follows

```
tsne = TSNE(random_state=RS).fit_transform(sample)
```

Below is an example of using the logistic regression algorithm

```
model = LogisticRegression()
model.fit(Xtrn, Ytrn);
predicted = model.predict(Xtest)
```

To build the graphs we used matplotlib.pyplot, pylab and seaborn libraries. The scipy library was used to construct a normal distribution from a sample.
The following is an example of "Clustering with categorical attributes" experiment code

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.manifold import TSNE
import seaborn as sns

# function for draw plot
# x is result if tsne clustering
# colors is an array with numbers of classes by which the sample is colored
def scatter(x, colors):
    # We choose a color palette with seaborn.
    palette = np.array(sns.color_palette("hls", 25))

    # We create a scatter plot.
    f = plt.figure(figsize=(12, 12))
    ax = plt.subplot(aspect='equal')
    sc = ax.scatter(x[:,0], x[:,1], lw=0, s=40,
                    c=palette[colors.astype(np.int)])
    plt.xlim(-25, 25)
    plt.ylim(-25, 25)
    ax.axis('off')
    ax.axis('tight')

    # We add the labels for each digit.
    txts = []
    for i in range(25):
        # Position of each label.
        xtext, ytext = np.median(x[colors == i, :], axis=0)
        txt = ax.text(xtext, ytext, str(i), fontsize=28)
        txt.set_path_effects([
            PathEffects.Stroke(linewidth=5, foreground="w"),
            PathEffects.Normal()])
        txts.append(txt)

    return f, ax, sc, txts
```

```
# reading sample where each raw is a point for clustering
sample = pd.read_csv('sample.csv', sep=',')

# run tsne clustering
tsne = TSNE(random_state=RS).fit_transform(sample)

# draw plot
# components is an array with numbers of teams for each raw in sample
scatter(tsne, components)
```

4 Conclusion

According to the results of the study, the following conclusions can be drawn:
- teams differ in their experience and the development process, so the time of tasks with the same volume may vary,
- clustering with categorical attributes gives an opportunity to estimate the task execution time,
- classification using logistic regression allows you to estimate task volume and task execution time,
- remove outliers from the sample improves the task volume forecast and task implementation time on the basis of the classification using logistic regression.

Acknowledgement: This work was supervised by Pavel Akhmetchanov, Chief Programmer of the Web Interface Department, Tinkoff Bank.

References

[1] Mueller Andreas C., Guido Sarah. Introduction to Machine Learning with Python. O'Really Media, 2016
[2] Kniberg Henrik, Skarin Matthias. Kanban and Scrum - Making the Most of Both. Stockholm, published on site http://www.infoq.com/minibooks/kanban-scrum-minibook - 2009-12-21- 41p., ISBN: 978-0-557-13832-6.
[3] Jeff Sutherland and Scrum, Inc. Scrum. Revolutionary project management techniques, 2014.
[4] Cohn Mike. How Do Story Points Relate to Hours? 2009 - http://www.mountaingoatsoftware.com/blog/how-do-story-points-relate-to-hours
[5] Henrik Brink, Joseph W. Richards, Mark-Fetherolf. Real-World Machine Learning. ISBN 10: 1617291927 ISBN 13: 9781617291920.
[6] van der Maaten, L.J.P.; Hinton, G.E. Visualizing High-Dimensional Data Using t-SNE. Journal of Machine Learning Research 9:2579-2605, 2008
[7].Hastie, T., Tibshirani, R., Friedman, J. The Elements of Statistical Learning, 2nd edition. — Springer, 2009. — 533 p.

Nataliya BOBROVSKAYA
Moscow Institute of Physics and Technology (MIPT)
Department of Innovation and High Technology
9 Institutskiy per., Dolgoprudny, Moscow Region, 141701, Russian Federation
nataliya.bobrovskaya@phystech.edu

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Medical Image Classification Based on Pixel Intensity Features

Bogdan Dumitru Mateis Boruz

Abstract

The current work outlines a computer assisted diagnosis for cancer by analyzing histopathological images. The procedure identifies light components in each slide and creates basic morphological features and statistics related to them. The obtained numerical dataset is given to machine learning classifiers in order to find the relation between the values for the attributes and their classes. Three different such classifiers are tried and two of them reach accuracy values similar to more complex methodologies, showing potential for further study.

1 Introduction

Histological images are captured from slices of tissues and examined under a microscope for the purpose of providing an accurate diagnosis, like if the tissue is healthy or cancerous. In the latter case, the degree of the cancer can be established based on the number and shapes of the components in the image. The main focus is to achieve a good accuracy, fact that implies:

- To identify which of the tested algorithms performs best.
- To quantify the importance of the features involved.
- And to potentially fine tune the algorithms to improve the results.

Cancer diagnosis using microscopical imagery is well sustained by research efforts [8], [9], [11]. Currently there is a traditional methodology that follows a set of stages [2], [3], namely a *preprocessing* stage which adjusts the images by sampling and illuminating colors, allowing a more precise key component identification, a *segmentation* stage which identifies key components like glands or nuclei, the third stage is *feature extraction and selection* where a large number of attributes such as morphological, topological, textural or intensity based, the selection itself is used to produce and then select the number of attributes.

The final stage is *diagnosis*, where a classifier is used on the produced, usually numerical, dataset. Most typically a supervised learning algorithm is used at this stage but it is not uncommon to see unsupervised or semi-supervised methods applied as well [9].

Deep learning algorithms have also been researched recently for this field and their analysis begins during feature extraction [9]. A difference here is the features are obtain directly from data compared to the one produced based on human knowledge. While this can be a valuable advantage, this class of methods requires large amounts of data for training and also needs the image resolution kept low. Their current computational cost requirements are very high as concerns both time and memory usage, even for low resolution images, at the time of writing.

Some of the advantages of the methodology described in the current paper are its relatively low computational cost, compared to the traditionally applied methodology and respectable, albeit somewhat lower, accuracy compared to previous contributions. However the accuracy is lower also because of the focus on grading the stages right (four classes), not just differentiating between healthy and unhealthy tissue (as in binary classification).

2 Dataset and Feature Extraction

The initial images are a set of colorectal histopathological slides in digital image format, specifically 357 images with a resolution of 800x600 [12].

They are separated in 4 categories: healthy tissue and 3 grades representing degree of progression of the disease. Therefore healthy tissue will be labeled as G0, while the other three will be labeled G1, G2 and G3. The healthy tissue samples are lower in number (62 images), while the other sets contain close to 100 each. The aim is to accurately classify by grade instead of only separating healthy and unhealthy tissue, which resulted in a 90% accuracy in preliminary testing.

The image set in Figure 1 shows one sample from each class in the image set along with a visualization of the contours extracted from the image for processing.

Data is collected from the image set using OpenCV library and the QT IDE by obtaining attributes about the contours in each image. Each image is first transformed to the greyscale format then goes through a thresholding process before the contours are found which helps by highlighting valuable features in each image. The right column in Figure 1illustrates the identified components using similar thresholding for samples in all different classes.

The parameters for the thresholding function ranges between 0 and 255 and the classification accuracy depends at a high extent on the value chosen for it. Therefore fine tuning this parameter proved to be more than useful in optimizing the data collection phase of this process.

After thresholding, the program finds all the available contours using the *findContours* function. A valuable parameter for it was the retrieval mode, with the *list* mode providing the best results while the others lagged behind it, the external mode performing better than the others. After the contours are collected, the area and perimeter is calculated and stored for each of the contours in a sample.

The main attributes used in this dataset are 8 in number, the minimum, maximum, mean and standard deviation for both the area and perimeter for the contours in an image. In order to provide some statistics about the obtained numerical dataset, Tables 1 and 2 show the mean and standard deviation for each numerical attribute and for each class in turn.

Class	Min Area	Max Area	StDev Area	Mean Area
G0	33907 ± 1.13e+05	2.98e+05 ± 1.80e+05	9623 ± 7045	430± 426
G1	74212 ± 1.59e+05	4.15e+05 ±57001	13679 ± 6891	597 ± 559
G2	74886 ± 1.58e+05	3.66e+05± 88711	7840 ± 2822	226 ± 110
G3	93062 ± 1.87e+05	4.14e+05± 1.37e+05	11105 ± 4794	374± 234

Table 1: Mean and standard deviation values for area attributes of each class.

Class	Contour Count	Minim Perim	Max Perim	StDev Perim	Mean Perim
G0	1457± 652	551 ± 2244	7665 ± 4529	235 ± 87.81	37.41 ± 8.33
G1	1389± 702	458 ± 1938	7370 ± 2227	230 ± 67.45	33.18 ± 10.29
G2	2536±934	1128 ± 3780	10730 ± 5045	235 ± 112	26.25 ± 9.43
G3	1857 ± 929	445 ± 1457	4762 ± 1794	131 ± 51.90	28.58 ± 10.64

Table 2: Mean and standard deviation values for the perimeter attributes and contour count.

3 Classifier methodology

The learning and predicting process is repeated 30 times and outputs an average accuracy obtained across those iterations, each repetition randomly samples two thirds of the data for training and keeps the last third for the training set. Doing this prevents forming a biased model for a specific dataset.

The first classifier tested was support vector machine (SVM), offering various accuracies depending on the thresholding value used during collection. Values between 100 and 230 were tested in increments of 10 until reaching the 200-230 range, where accuracy was found to be the highest. In this range increments

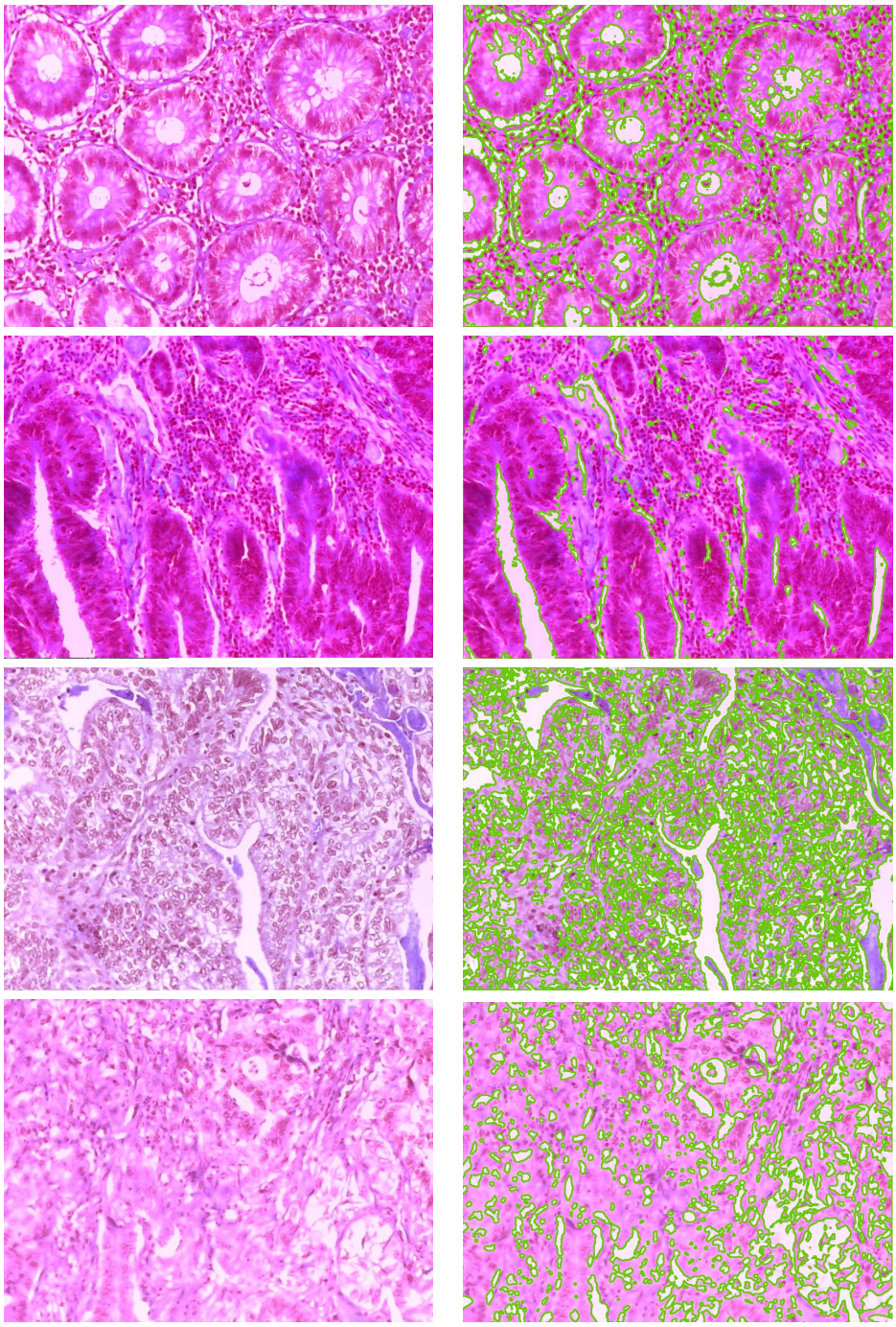


Figure 1: Image samples for each image class with contour detection visualized on the right column.

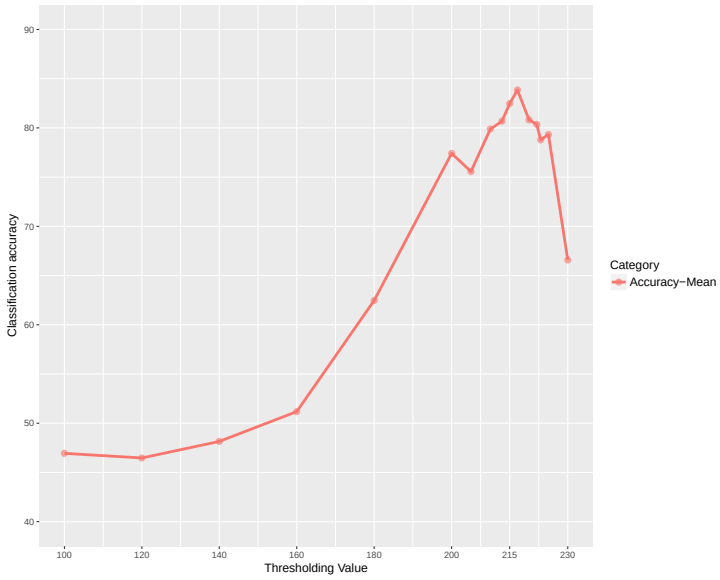


Figure 2: Classification accuracy as obtained by SVM for various threshold parameter values.

were lowered until pinpointing 217 as the best value for this parameter with an accuracy of 83%, after tuning the SVM classifier as well. There is a visual representation of this in Figure 2.

Before tuning the SVM parameters the classification accuracy by lower by about 10% on average than those displayed in Figure 2.

Two other classifiers were tested for this dataset were decision trees and bagging. Without tuning the former showed an accuracy of 76% while the latter came very close to the tuned SVM at just over 80%.

4 Results, visualization and tuning

For the Bagging and Decision Trees classifiers, the importance of attributes was calculated by making use of the caret package [4]. An interesting finding was the significant differences in how each classifier values the attributes used. The importance of the attributes shows some variety depending on the used classifier. However a level of consistency is notable in some, such as the area standard deviation. Their importance of the features, as discovered by the two classifiers is displayed in Table 3.

Feature	Decision Trees	Bagging
Contour Count	52.29	81.55
Area StDev	**70.67**	**103.14**
Perimeter StDev	43.87	86.53
Max Area	68.04	98.09
Max Perimeter	61.58	93.98
Mean Area	39.35	75.59
Mean Perimeter	67.15	99.04
Min Area	5.72	27.29
Min Perimeter	0.0	22.48

Table 3: Variable importance as found by decision trees and bagging.

The SVM classifier initially showed accuracies around 74% and after changing some of the parameters in the learning function, a noticeable change in the accuracy occurred which prompted further investigation. After using a trial and error approach the best values for the gamma and cost of the SVM function

were found to be 0.1 and 9 respectively for this dataset.

The tune function from the e1071 [5] package that uses a brute force approach to find the best parameters in a given interval provided different values that unfortunately did not provide any observable difference in the classifier results.

Finally the multiple kernel types were also tested but the default radial kernel proved to be the most effective, while the sigmoid kernel had very poor results, with less than 50% accuracy even on the best dataset.

The linear and polynomial kernels were somewhat better but even with some attempts at parameter tuning they ended up not even getting close to even the radial results before tuning.

4.1 Additional Features

In addition to the 8 attributes mentioned earlier, there were attempts to increase the dimensionality of the data, initially by adding another set of the same 8 attributes for each image using the same method of extracting the features with different values for the thresholding.

Another attempt to find improvement was storing and calculating those statistics only for contours at a relatively shallow level in the image and their immediate descendant in the contour hierarchy, however this did not affect any of the classifiers in a visible manner. This may indicate the additional data collected was redundant.

An attribute that proved to be surprising in how it affected the classifiers was the total number of contours found in an image. This attribute had no effect on the SVM classifier precision while showing a minor negative impact on the decision trees and a significant improvement for the bagging classifier, bringing the latter up to 84% accuracy.

4.2 Results

The table below displays a confusion matrix obtained from a single run from each classifier. It represents the predicted class versus the actual class for each observation.

	Bagging				SVM				Decision Trees			
	G0	G1	G2	G3	G0	G1	G2	G3	G0	G1	G2	G3
G0	15	3	1	3	16	2	0	1	14	1	1	0
G1	2	27	4	0	0	25	13	0	3	20	7	0
G2	2	4	25	0	4	0	27	1	3	7	26	2
G3	0	0	1	34	2	1	0	29	3	2	1	31

Table 4: Confusion matrix resulted from a distinct run of each classifier.

An important point to state is that for false negatives, which are arguably the worst mistake to make when dealing with disease, the highest values are obtained for the decision trees classifier. It has a respectable accuracy but lags noticeably behind in this regard. The bagging classifier is close in accuracy to the SVM but it has 33% less false negatives, making it arguably the best classifier among the three tested here.

Various statistics for each grade and classifier are listed in the Tables 5 and 6.

A noteworthy observation when looking at Table 5 is the overall lower accuracy for the decision tree classifier that showed high precision for the first class. However it showed poor performance in many other aspects (like of F1 measure) and fell noticeably behind the other two classifiers.

In addition, the classes G1, G2 and G3 are more accurately classified than the healthy one, for two of the classifiers, visible in the balanced accuracy column, making this methodology relevant at least for a secondary opinion from a medical expert needing to distinguish between slides.

Finally it is important to look at similar approaches applied to the same image dataset. Such methods are introduced in [10] and [11]. They make use of structure segmentation in the histological images and feature extraction. A key difference is the procedures assuming specific gland and nuclei identification, extracting roughly 80 features from them, as compared to 9 in the current work. Through feature

Class	Sensitivity	Specificity	Precision	F1	Balanced Acc.
Bagging					
G0	0.788	0.966	0.897	0.839	0.877
G1	0.958	0.918	0.742	0.836	0.938
G2	0.838	0.976	0.939	0.886	0.907
G3	0.889	0.957	0.857	0.873	0.923
SVM					
G0	0.737	0.980	0.875	0.800	0.859
G1	0.941	0.920	0.821	0.877	0.930
G2	0.806	0.967	0.893	0.847	0.887
G3	0.946	0.964	0.921	0.933	0.955
Decision Trees					
G0	0.435	1.000	1.000	0.606	0.717
G1	0.697	0.875	0.676	0.687	0.786
G2	0.852	0.883	0.676	0.754	0.867
G3	0.921	0.904	0.814	0.864	0.912

Table 5: Statistics for each class in turn and for each class for the three classifiers.

Class	Neg Pred Value	Prevalence	Detection Rate	Detection Prev.
Bagging				
G0	0.883	0.190	0.083	0.083
G1	0.885	0.273	0.190	0.281
G2	0.954	0.223	0.190	0.281
G3	0.962	0.314	0.289	0.355
SVM				
G0	0.952	0.157	0.116	0.132
G1	0.976	0.281	0.264	0.322
G2	0.935	0.256	0.207	0.231
G3	0.976	0.306	0.289	0.314
Decision Trees				
G0	0.924	0.273	0.215	0.240
G1	0.989	0.198	0.190	0.256
G2	0.932	0.306	0.256	0.273
G3	0.968	0.223	0.198	0.231

Table 6: Secondary statistical table for the classifiers used.

selection they reach accuracies of around 80% [10] and 84% [11]. It is fair to specify that in these works default parameter settings were kept for the classifiers. Regardless, the results are very close to those achieved in the current work but with less computational effort herein. A convolutional neural network [8] obtained up to 91.44% and a combination of classifiers [13] applied directly to the images (without crafted features) achieved the best performance up to now at an impressive 95.65%. However neither of the last two mentioned approaches does feature extraction and their computational costs are high.

4.3 Application

An application was developed in the C++ language using the QT 4.2.0 IDE. The QT framework is used for creating the user interface as well as for exporting the extracted data from the image files. The application outputs a preview for the contour detected live when modifying the threshold parameter via a slider. This feature helped in eventually narrowing down the optimal parameters.

Additionally, the application provides a visual aid in understanding the contour detection process and the importance of the retrieval methods, as well as the option of selecting the contour retrieval

type. Various functions from the OpenCV library are used. The components are retrieved using the *findContours* function for each image, while the perimeter and area are subsequently calculated using the *contourArea* and *arcLength* functions.

After a file is created with the extracted values, the file is read in R software where the machine learning models described earlier in the manuscript are created and used in the classification process.

5 Conclusions

The current paper puts forward a methodology making use of a simple, rather small amount of morphological features. They are not directly connected to components such as nuclei or glands in the images, but instead make use of light components in them through intensity-based thresholding. With only basic statistics extracted from perimeters and areas of the contours and fine tuning the SVM classifiers to perform close to methodologies involving very large amounts of attributes and a complex feature selection process, as well as reaching the same performance with a bagging classifier.

Acknowledgement: This work was supervised by Associated Professor *Catalin Stoean*, from *Department of Computer Science, University of Craiova*, Romania.
The author acknowledges the financial support of the Society for Computer Technologies, Craiova, Romania.

References

[1] Carl de Boor, *A practical guide to splines*, Springer-Verlag, New York Heidelberg Berlin, 1978.

[2] M. N. Gurcan, L. E. Boucheron, A. Can, A. Madabhushi, N. M. Rajpoot and B. Yener, Histopathological Image Analysis: A Review, *IEEE Reviews in Biomedical Engineering*, **2** (2009), 147–171.

[3] D. Komura and S. Ishikawa, Machine Learning Methods for Histopathological Image Analysis, *Computational and Structural Biotechnology Journal*, **16**, (2018), 34–42.

[4] Max Kuhn. Contributions from Jed Wing, Steve Weston, Andre Williams, Chris Keefer, Allan Engelhardt, Tony Cooper, Zachary Mayer, Brenton Kenkel, the R Core Team, Michael Benesty, Reynald Lescarbeau, Andrew Ziem, Luca Scrucca, Yuan Tang, Can Candan and Tyler Hunt. (2017). caret: Classification and Regression Training. R package version 6.0-78.

[5] David Meyer, Evgenia Dimitriadou, Kurt Hornik, Andreas Weingessel and Friedrich Leisch (2017). e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien. R package version 1.6-8.

[6] Andrea Peters and Torsten Hothorn (2017). ipred: Improved Predictors. R package version 0.9-6.

[7] T. Popoviciu. Introduction à la théorie des différences divisées. *Bull. Math. de la Soc. Roumaine des Sci.*, XLII: 65–78, 1940.

[8] S. Postavaru, R. Stoean, C. Stoean, G. Joya Caparros, Adaptation of Deep Convolutional Neural Networks for Cancer Grading from Histopathological Images. In *I. Rojas, G. Joya, A. Catala (eds) Advances in Computational Intelligence. IWANN 2017*, Springer, Cham, (2017), 38-49.

[9] D. Shen, G. Wu and H.-I. Suk, Deep Learning in Medical Image Analysis, *Annual Review of Biomedical Engineering*, **19** (2017), 221–248.

[10] C. Stoean, R. Stoean, A. Sandita, D. Ciobanu, C. Mesina, C. L. Gruia, SVM-Based Cancer Grading from Histopathological Images using Morphological and Topological Features of Glands and Nuclei. In *G. Pietro, L. Gallo, R. Howlett, L. Jain (eds) Intelligent Interactive Multimedia Systems and Services 2016. Smart Innovation, Systems and Technologies*, vol. 55, Cham, Springer (2016), 145-155.

[11] C. Stoean, In Search of the Optimal Set of Indicators when Classifying Histopathological Images, *IEEE Post-Proceedings of the 18th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, Timisoara (2016), 449-455.

[12] C. Stoean, R. Stoean, A. Victor Sandita, D. Ciobanu and C. Mesina, Colorectal cancer histopathological image data set

[13] C. Stoean, D. Lichtblau, Classifier Result Aggregation for Automatically Grading Histopathological Images, *19th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, IEEE Computer Society, Timisoara, in press.

[14] Terry Therneau, Beth Atkinson and Brian Ripley (2017). rpart: Recursive Partitioning and Regression Trees. R package version 4.1-11.

Dumitru Bogdan Mateis
BORUZ
Facultaty of Sciences
Department of Computer Science
13 A.I. Cuza Street, Craiova, 200585
Romania
E-mail:*bogdan.mateis@yahoo.com*

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

An efficient FPGA implementation of a deep learning approach for the classification of saccadic movements in clinical electro-oculography

Carlos Cano-Domingo

Abstract

This work presents a hardware implementation of a Convolutional Neural Network (CNN) on a Field Programmable Gate Array (FPGA) device. The implemented CNN is devoted to classifying in Healthy, Presymptomatic and Sick subjects concerning the Spino Cerebellar Ataxia type 2 (SCA 2) disease by using saccadic patterns as inputs. Our main objective is to study the viability of implementing this kind of deep neural networks on a low-cost FPGA device and its competitiveness versus classical personal computers. To this end, we have first implemented our CNN model in Keras, a high-level language oriented to software implementation on PC or GPU; then we have implemented it again in Matlab, in order to directly evaluate the different mathematical operations involved in it; finally, it has been implemented on a low-cost Zynq 7000 FPGA model using a High-Level Synthesis Language (HLS), which drastically reduces the design time and optimizes the use of components in the FPGA. The result report about the hardware implementation offers a reduction in processing times between 8 and 12 times regarding software solutions on PC. Respecting to the used resources, it has required a 78% of the total available Digital Signal Processors (DSP), the memory is mostly implemented in Look-up Table (LUT) and the total usage of LUT resources is around a 60% of the total. These indicators make promising the FPGA based implementation of CNN models versus traditional software-based implementations running on a current Hospital's computer.

1 Introduction

Electrooculography is a diagnostic technique that allows us to register the weak electrical potentials generated by the eye movement when following the trajectory of an object. Because of its non-invasive nature, it is frequently used for the diagnosis and monitoring of neurodegenerative diseases. Many of these diseases are characterised by having no cure and by producing a gradual and increasing alteration of the nervous system, making any physical activity result in a challenging task. These conditions justify the effort of investigating the development of a monitoring and diagnosis system aimed at two main objectives: the first one consists in allowing an early detection of the disease, thus allowing rehabilitation planning to improve the quality of life; the second one is the implementation of a system as portable as possible, therefore minimizing the need for patient displacements.

In this sense a team formed by researchers from the University of Malaga (Spain), Technical University of Manabí (Ecuador) and University of Craiova (Romania) is focused on the design of a system that classifies the possible patients into Healthy, Presymptomatic and Sick with regard to the Spino Cerebellar Ataxia type 2 (SCA 2) disease. The patterns to be classified consist of saccadic records corresponding to the tracking of an object in a well-defined displacement. This task would allow the early detection of people in the primary stages of the disease, where symptoms are so weak that they have no external manifestation (presymptomatic) [1]. However, it has been shown to be an incredibly complex objective so far, due to the difficulty of distinguishing between the patterns of a healthy person and those of a presymptomatic one. This limitation affects both classical computational methods and the best-known machine

learning techniques [2]. This team, which includes the author of the present work, is currently exploring the applicability of the relatively new Convolutional Neural Networks (CNN) for this classification.

From both objectives above mentioned, the present work focuses on the second one: the search for a device as portable as possible. In this sense, the paper presents a first approach to the implementation in a Field Programmable Gate Array (FPGA) of a primary CNN, which has been previously proposed by another member of our team. The software implementation on the PC of this model will serve as a counterpoint to the hardware implementation described here.

FPGA offers a flexible and high-performance platform for implementing a neural network model. Two other devices were considered. Mini-PCs were rejected due to the high power consumption. Microcontrollers are devices for low-performance applications which are not suitable for a Convolutional Neural network algorithm.

It is important to note that our interest in this study does not derive exclusively from the portability of an FPGA-based system. The latest advances in FPGA design are leading many researchers to think about the competitiveness of these (Graphics Processing Unit (GPU)-based implementations from a computational point of view [3]. Thus, among other aspects, we can point out that FPGAs are primarily designed to handle irregular parallelism and custom data types, which is a particularly hard task for GPUs [4].

The rest of this document is organised as follows. In the *Problem Description* section, the object of this work is briefly indicated, specifying the nature of input and output data, as well as the specific device used for the FPGA implementation. In *Model Description*, the architecture of the implemented CNN is described both graphically and from a computational point of view. *FPGA Description* briefly describes the essential characteristics of the FPGA devices, in particular, the Zynq 7000 model (used in our case), and introduces the functionality of a High Level Synthesis Language (HLS). *Implementation* tries to give a detailed and affordable description of the process followed by the data in the proposed hardware implementation. The results of the tests carried out are detailed in *Experimental Results*. Finally, *Conclusions* summarises the main ideas and conclusions of the work.

2 Problem Description

In essence, our work is focused on the hardware implementation of a pattern classification system. In our case, the patterns to be classified correspond to saccadic movements (the electro-oculographic record of an eye movement during the tracking of an object in a well-determined ocular angle). These patterns have to be classified into three categories: healthy or control, presymptomatic and sick (see figure 1). The classification system used consists of a CNN, and for its hardware implementation, an FPGA Zynq 7000 has been used. The study of the feasibility of this hardware implementation constitutes the real nucleus of this work.

The patterns to be classified, that is, the saccades, constitute a set of 6159 vectors obtained from real electro-oculographic tests to healthy or control subjects (2280 patterns), presymptomatic subjects who have started the SCA2 disease but without visible symptoms still (2317 patterns), and sick subjects with visible symptoms (1562 patterns). Initially, the obtained saccades are vectors of 192 components, resulting from measuring in microVolts the potential generated at a point in the ocular environment during each saccadic movement. The saccades correspond to the tracking of a luminous point that moves 10, 20, 30 and 60 degrees concerning the ocular axis in the tests performed on each subject. This vector of potentials is translated by calibration to a vector of angular positions, and later all the vectors are scaled to a 30 degree path providing the definitive input patterns for our system.

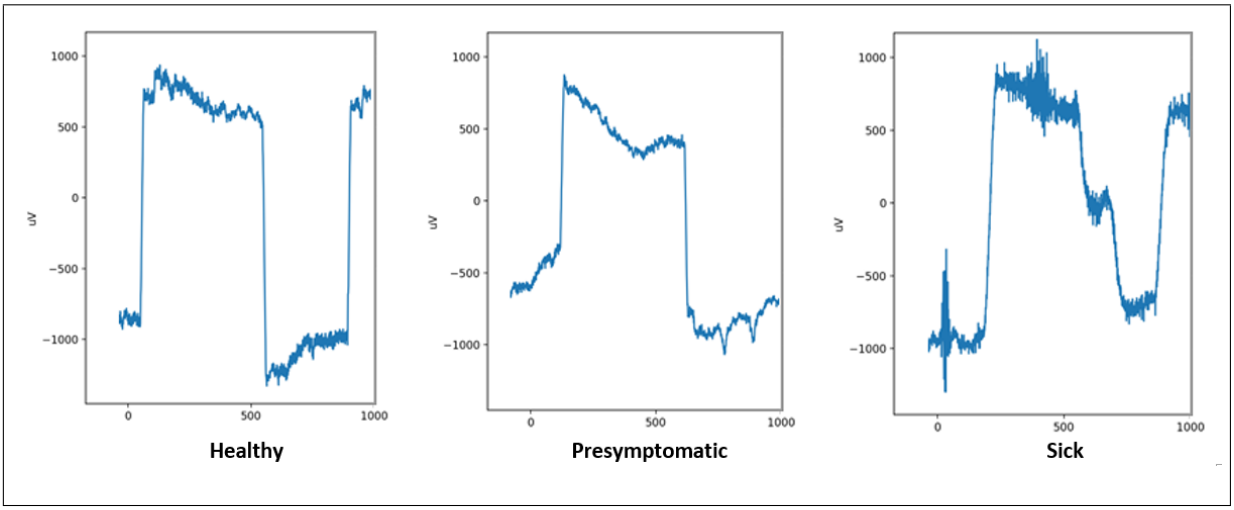


Figure 1: Examples of Healty, Presyntomatic and Sick saccades

3 Convolutional Neural Network

In this section, we will describe the used CNN model from a structural point of view, describing the layers and the data flow. Additionally, we will mention the different tools that we have used to program it: KERAS and MATLAB. Thinking of a possible reader who is not very familiar with the CNN terminology, we provide a brief description of terms such as convolutional filter, ReLu, max-pool or softmax the first time they appear. In any case, we recommend the consultation of [5] for a complete update regarding this terminology.

3.1 The Convolutional Neural Network architecture

The model used in this design is a four layer CNN (Figure 2). The inputs of the network are vectors with 192 components. The first layer carries out an 1-Dimensional convolutional of 19 component kernel size (i.e. we apply a filter (convolution) of length 19, which moves along the input vector advancing one unit at a time). At the same time, the learning depth is expanded to 10 filters (channels). The result of this convolution is subject to a ReLu activation function neuron (which can be considered a neuronal rectifier described by f(x)= max(0,x)). Finally, we apply a max-pooling of size 2 (i.e. a max filter to each non-overlapping pair of neurons), thus reducing the size of the layer by half. The second layer carries out a 1-Dimensional convolutional with 9 components and 10 filters in the kernel, a ReLu activation function and a max-pool of size 2. The output of this second layer is flattened and becomes the input of the third layer. This one is a fully connected layer with 60 ReLu neurons. The fourth layer is another fully connected one with 3 SoftMax neurons (neurons with a normalized exponential function), which constitute the Neural Network output. Each one of these output neurons is associated with a class (Healthy, Presymptomatic and Sick) of our classification problem (Figure 1).

3.2 Implementation. Keras vs MATLAB

Before physically implementing the CNN proposal in the FPGA, we have to define its architecture clearly. This involves not only the high-level architecture, but we have to delineate which operations are implicated. Besides, in this first stage of this project, we aim to implement the neural network that has already been trained. Thus, we need to define our network model for obtaining the final weights.

In order to train the network, we have to make a computational implementation of our model. The Keras API offers us a tool for making in a friendly way both the description of the network and the training, in a very high-level. Due to its easy programming, we have been able to study different architectures with distinct numbers of layers and diverse types of training. However, precisely because Keras

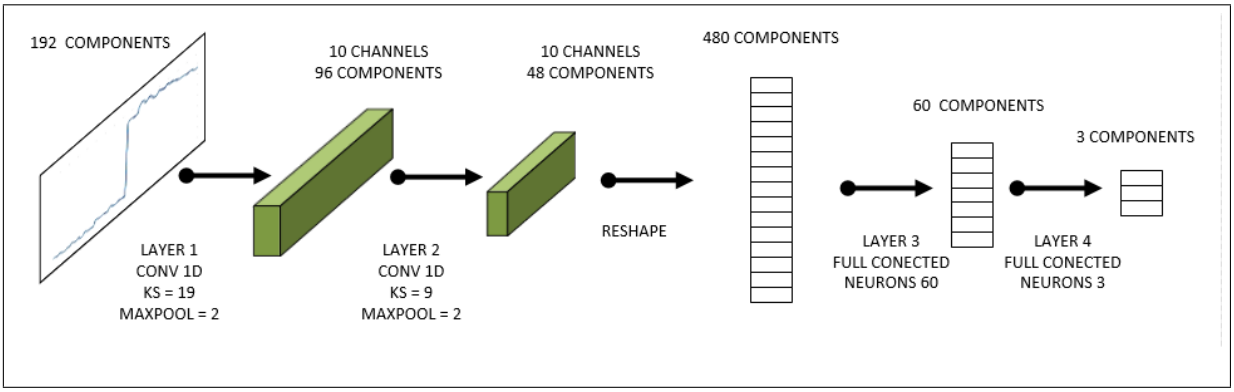


Figure 2: CNN Saccade Architecture

is a high-level API for programming, this tool does not allow us to test our network stage by stage and to know the operations involved. Additionally, with Keras it is not possible to know the intermediate output of the model, which is especially needed in our FPGA implementation.

One tool that does allow us to have a notion in detail of all operations involved in our model is the programing language MATLAB because this is a language that enables us to decompose all functions into matrix operation. Thus, when a definitive model is obtained with Keras, its similar implementation is performed in MATLAB. Finally, the HLS language allows us to describe the operations implemented in FPGA employing a medium-level programming process. Due to this implementation, we are able to test the model stage by stage in the FPGA, comparing the results with the same network in MATLAB design.

To summarize, our work methodology has consisted in having achieved the first design in a very high-level tool, i.e. Keras, for describing and obtaining the network with a friendly, quick and flexible behavior. The second lower-level stage consisted of decomposing all the operations from the Keras CNN model into mathematical operations within MATLAB. This step has allowed us to evaluate the functioning of the FPGA design for the last stage.

4 Field programmable gate array

The FPGA related concepts, the particular platform Zynq-7000 and the HLS language, are next outlined.

4.1 Background

Field Programmable Gate Arrays (FPGAs) are programmable electronics devices composed of a high number of logic blocks. Both connectivity and functionality of this block can be reconfigured by using a High Level Description language. The manufacturers propose distinct types of microarchitectures, each one of them adapted to specific purposes. These microarchitectures include types of logic blocks, such a simple set of logic gates or more complex RAM and DSP blocks. Different models have been implemented for each particular architecture to fit price and use resources to the application difficulty. Currently, new models have been introduced accordingly to join benefits from FPGA, GPUs and Microprocessors known as Heterogenous devices [6], for instance, Zynq-7000 in Xilinx or Cyclone in Intel. These allow the task to be shared in different hardware, thus maximizing advantages of each component.

4.2 Zynq 7000

Zynq-7000, the platform developed by Xilinx, was chosen for this work. It is composed of two different parts: one for programmable logic (PL) and the other for the microprocessor (PS). The microprocessor part is made of two peripheral ARM Cortex-A9 and other resources for interconnecting to programmable logic. The main resources of the logic area are the System Gates, which are sets of Look-up tables and

Registers, but it also contains 18Kb RAM Blocks and basic 48bit-depth DSPs with only one multiplier and one adder. CNN is formed by highly parallelizable part with many simple operations and by another part with a strong sequential behavior, which is very suitable for the Zynq platform. The device used in this work was the xc7z020, its logic area was formed by 280 blocks RAM, 220 DSP48E, 100.000 LUTS and 50.000 FF, and the microprocessor part being the same in all Zynq-7000 model.

4.3 High Level Synthesis

The programming of FPGA has always been an important problem as far as development time, and the difficulty is concerned. Therefore new paradigms of hardware programming have been released lately. Vivado High Level Synthesis is the language proposed by Xilinx. In it, Xilinx offers to programme in C++, C or SystemC, and focus in system level in opposition to register level used in traditional languages like Verilog and VHDL. HLS is a set of technologies which allows the transformation of the code written in the medium-level language to registers. This transformation is called synthesis by Xilinx. The same functionality could be written in many different codes, and these codes may be synthesized in several ways. Code Styling and indication for synthesis are paramount because each result of synthesis produces implementation recourse and performance. This paper proposes one code and set of indications to obtain the best performance and lower usage of resources. The terms related to HLS may be consulted in [7].

5 Implementation

The architecture designed in FPGA is described in this section.

The data type utilised is 18 bits fixed point with 8 bits for the integer part and 10 bits for the decimal part. All blocks in this section represent this data type.

Figure 3 shows the operation involved in the implementation of preparing inputs to the first convolutional layer. Observing the figure from left to right, one vector with 192 components can be seen, which represents a saccade. This vector is submitted to a padding operation, adding 9 components at the beginning of the vector and 9 components at the end, in order to facilitate the convolutional operation. This vector is replicated 10 times, for easing the parallelism in the convolutional layer 1. The last part of the figure represents the weights and bias structures of the first layer, which are distributed along the FPGA, so it is not possible to know where they are located.

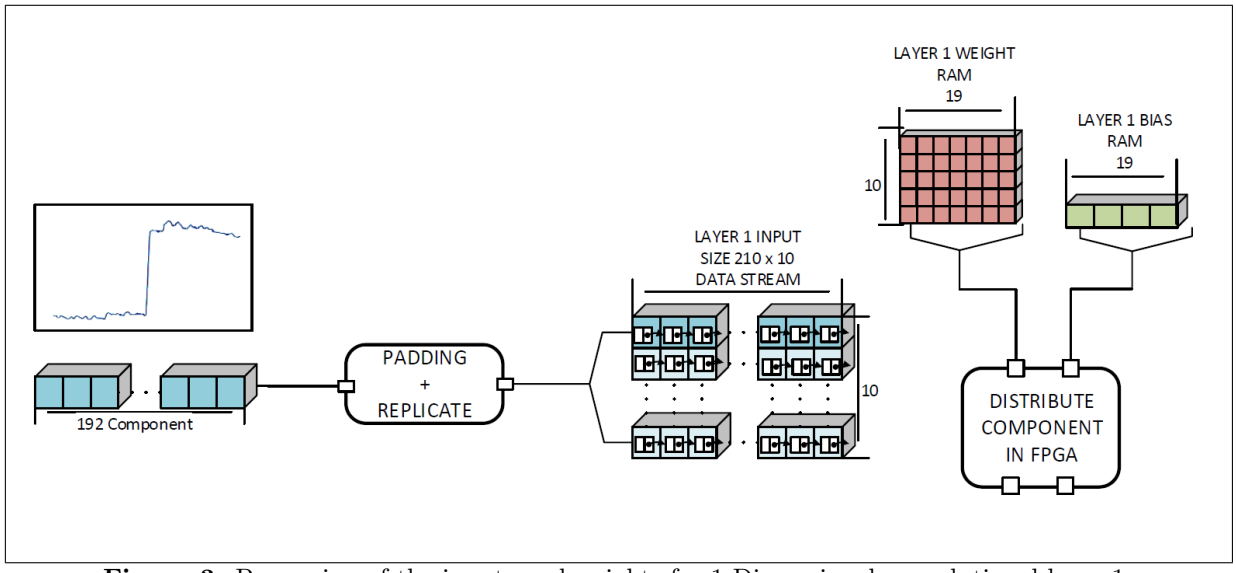


Figure 3: Processing of the inputs and weights for 1-Dimensional convolutional layer 1

Figure 4 shows the data flow in the stage of convolutional layer 1. The inputs, weights and bias that we have in the previous stage are the first elements in this figure, looking from left to right. In the first

loop, one element is taken of the 10 inputs replicated vector. With this vector, the second loop with 192 iterations is realized. In the first iteration, 19 components of this vector are taken into a buffer and in each new iteration one value from the vector is introduced in the buffer and all of the rest are shifted, losing the last value. The convolutional operation is done with the previous vector and the corresponding filter weight vector and bias component. The value obtained is processed by a ReLu activation function. The result of every 192 iterations is a vector with 192 components, and one vector is produced for each filter, so the output of the convolutional layer 1 is a set of 10 vectors with 192 components in each.

The hardware convolutional layer 1 optimization consists in using 19 DSP for doing all operations inside the product function in only 7 clock cycles, besides, pipeline directive in the loop allows to obtain all the sequence in 254 cycles, instead of a 192 multiplier for 7 cycles in product function. The compiler of HLS can optimize the data dependence by the usage of a data-flow directive and stream connection.

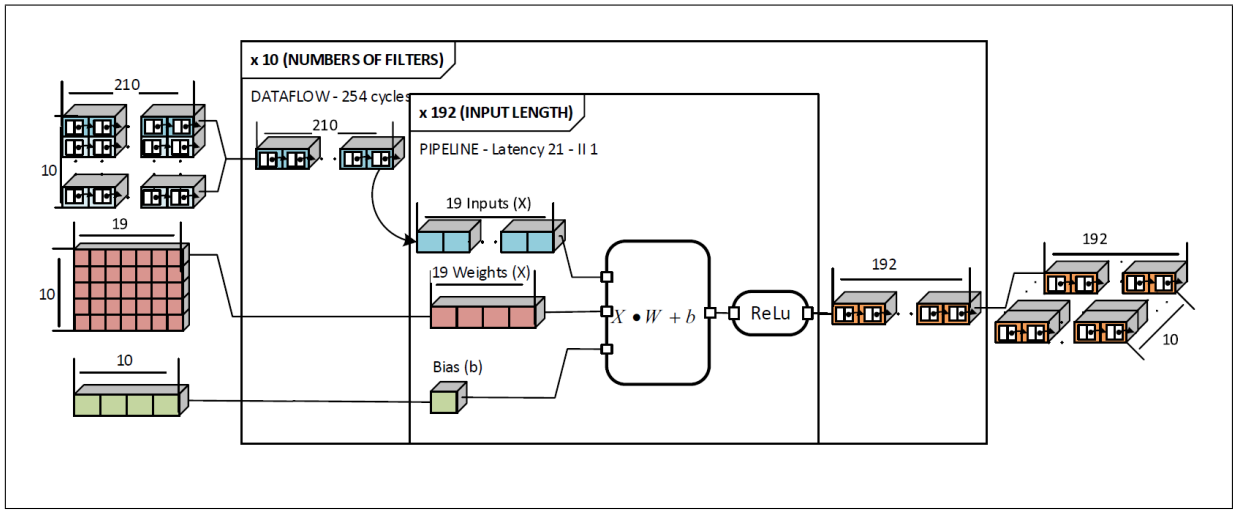


Figure 4: 1-Dimensional convolutional layer 1 of the network

Figure 5 represents the operations done before entering into the convolutional layer 2. In the top part, the max-pooling of size 2 and padding operations are performed. Also, the matrix is replicated 10 times, for the same reason as in the previous layer. The bottom part shows the weights and bias structures which are distributed along the FPGA.

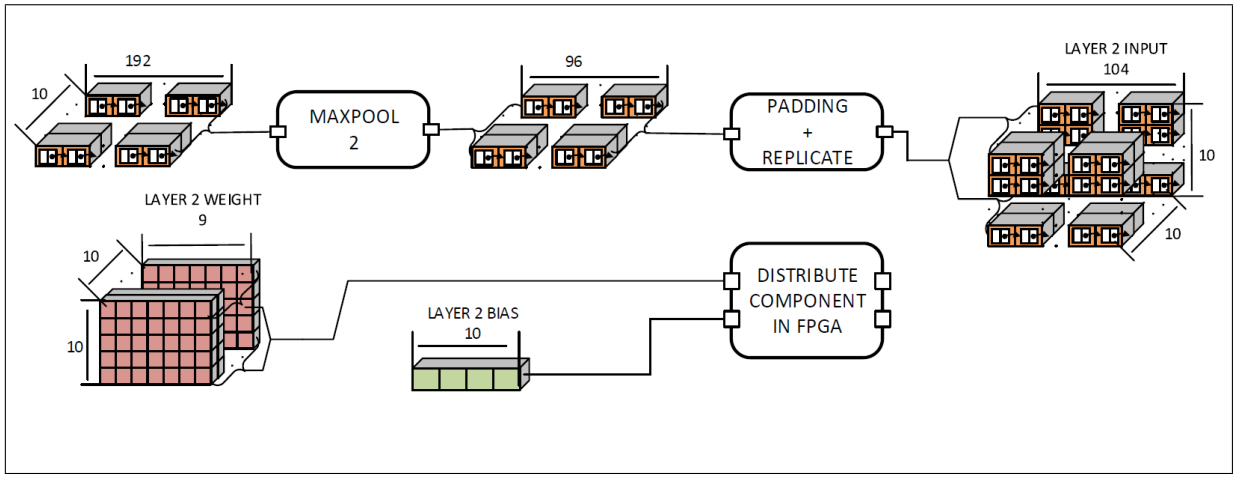


Figure 5: Processing of the inputs and weights for 1-Dimensional convolutional layer 2

Figure 6 shows the data flow in convolutional layer 2. The 3D matrix input, 3D matrix weights and bias vector are the same as in the previous stage. In the first loop, one horizontal matrix inputs of the 10

horizontal matrices are taken, which represents the one entire output of convolutional layer 2. A second loop with 96 iterations is performed with this matrix. In this filter, there are 10 buffers, one for each input vector. In the first iteration, 9 values are taken form each input vector, and in every new iteration one value of each filter is introduced in each buffer, the rest components of the buffer are shifted, and the last is discarded. For all of these buffers, the corresponding filter weight matrix and bias are processed by a function. The output of this function is calculated by a ReLu activation function, and this is done for all iterations, so the output of this loop is a vector with 96 components. Because the convolutional layer 2 has 10 filters, the output of the layer is a set of 10 vectors with 96 components each one.

The hardware convolutional Layer 2 optimization consists in doing loop filter considering data dependence, using stream data input and output and data-flows directives. In the second loop, pipeline optimization has been used, with 11 cycles for each iteration, but only one cycle between iterations. The function is performing with 90 DSP and only takes 11 clock cycles for doing all operations.

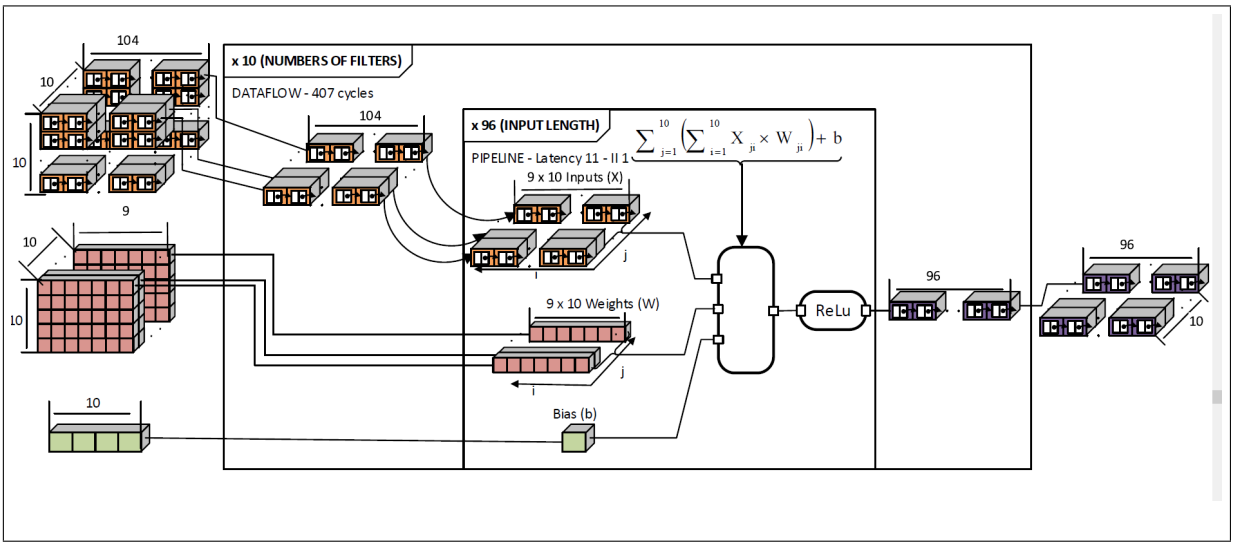


Figure 6: 1-Dimensional convolutional layer 2 of the network

Figure 7 represents the operations done before entering into the fully connected layer 3. The output of convolutional layer 2 is processed by a max-pool function. Fully connected layers need that all the input is as a 1D vector. Thus, the next step is to transform the matrix into a vector with the same total components, and this is done by a flatten function.

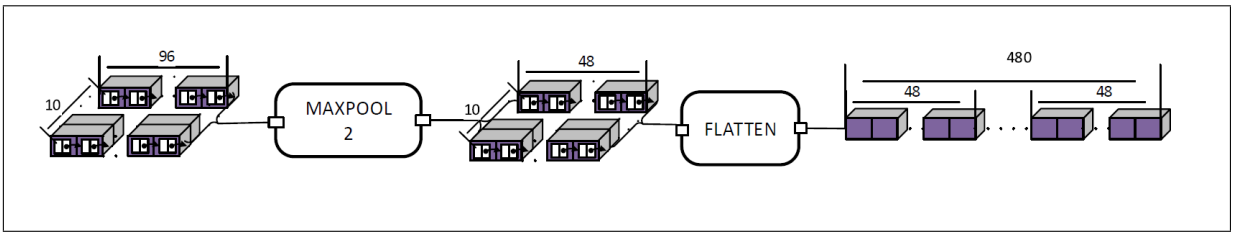


Figure 7: Processing of the inputs for the fully connected layer 3

The hardware optimization for this stage consists in applying a data-flow directive and parallelism in each output. Due to the fact that access to the data is done many times, it is not possible to achieve useful parallelism.

6 Experimental Results

The important results in FPGA design are the usage of the hardware resources and running time result. Vivado HLS estimates the usage of the hardware and offers reports for each part of the implementation.

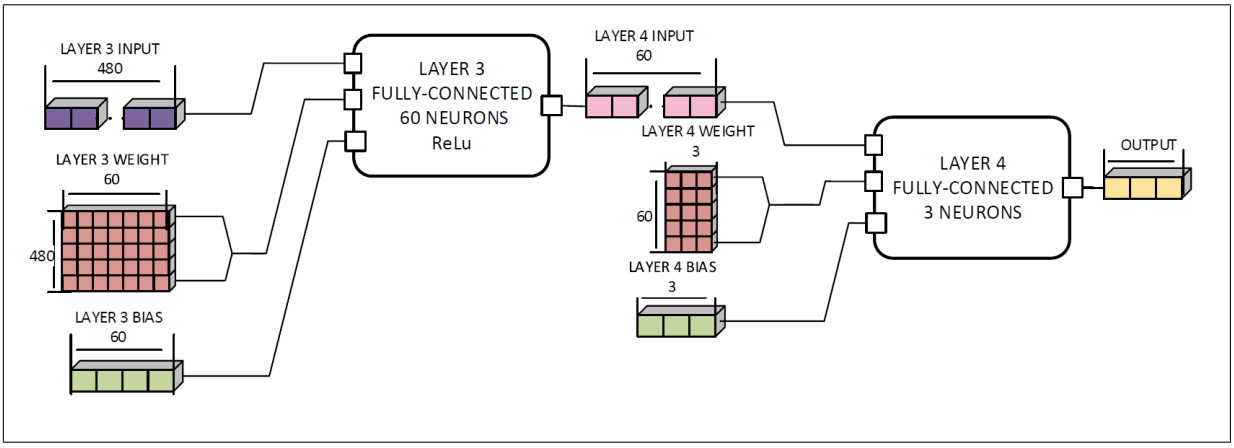


Figure 8: Fully Connected layers 3 and 4 of the network

The main report is the block utilization, as it can be seen in Table 1.

We have used a 78% of the total DSP available. This mark is a satisfactory result in order to optimize the operation along with the usage of a significant number of DSP. In the convolutional layer 1, we use 19 DSP, one for each component, for doing the multiplier operation, this optimization allowing us to do the operation in only 1 cycle. In the second convolutional layer, we use 90 DSP, one for each component of the matrix, and we complete the function in 7 cycles, while the rest of the 60 DSP used to correspond to the output neuron parallelization in the first dense layer.

Another critical thing to be careful of is the usage of the Block RAM (BRAM), because a significant usage of this resource may add latency time to the design. In our implementation, the primary structure used for memory is FIFO. For this, we only use 110 on BRAM of the total implemented, and all of the rest is done in LUT. This is not a problem because we have deferents memory for each vector, so we have access to the data in an individual way. Instead of the LUT resources, the 61% has been used, and this is a proper result. The FPGA needs other blocks such as memory controller, a clock controller, the connection with the processor part and another logic. Thus, we have practically a 40% of LUT resources to implement all these blocks in the final and complete implementation.

The running time results of the complete blocks can be seen in Table 2, and of each block in Table 3. It can be noted that the total time interval is near to layer 3 latency time. This is acceptable because the complete design uses dataflow optimization. In this, the data dependence is analyzed for doing each operation individually, without waiting for the entire block. HLS allows the next layers to use the data derived from the previous layers continuously, without needing to wait for the previous layers to finish generating all the data.

In order to compare our proposed implementation performance with that of a standard PC one, two laptops have been chosen that can be used in hospital environments. The compared running time results can be seen in Table 4. The last column shows the ratio between the reference computers and FPGA. The PC results exhibit between 8 to 12 times more needed time than the suggested FPGA design.

Name	BRAM_18K	DSP48E	FF	LUT
DSP	-	-	-	-
Expression	-	-	0	2108
FIFO	110	-	5950	12210
Instance	0	172	24886	17100
Memory	1	-	36	135
Multiplexer	-	-	-	1188
Register	-	-	138	-
Total	111	172	31010	32741
Available	280	220	106400	53200
Utilization (%)	39	78	29	61

Table 1: Utilization estimates

	Latency(Clock cycles)	Interval(Clock cycles)	Clock cycle(ns)
CNN	7894	7204	10

Table 2: Designed FPGA running time

	Latency (Clock cycles)	Interval (Clock cycles)
Layer 1: Padding and replicated	214	214
Layer 1: Convolution	2541	2541
Max-pool, Padding and replicated	2034	2034
Layer 2: Convolution	4071	4071
Max-pool and Flatten	484	484
Layer 3: Fully Connected	7203	7203
Layer 4: Fully Connected	484	484

Table 3: Running time report of each block

Ref	Processor	RAM	Time	$\frac{Computer_{time}}{FPGA_{time}}$
Computer 1	Intel® Core™ i5-5257U	8 GB 1866MHz DDR3	937μs	11.86
Computer 2	Intel® Core™ i7-8550U	16 GB 2400MHz DDR4	612μs	7.74
FPGA	-	-	79 μs	1

Table 4: Comparison of running time results between the FPGA implementation and those conducted on two standard PCs

7 Conclusions

A new paradigm for FPGA programming has been introduced in the recent years. Thus, the time spent for developing an FPGA design has been decreased from months or years to few weeks. Implementing a hardware design easily and quickly is now possible with FPGA. New architectures of Artificial Intelligence are released even faster, and the time required for traditional FPGA implementations would make their use impossible.

In this work, we have proposed a CNN design on FPGA. Our architecture is formed by a convolutional layer with a ReLu activation function, followed by one max-pool function, connected to another convolutional layer, with a ReLu activation function, followed by a max-pool function. The output stage consists of two fully connected layers, the first with 60 neurons and a ReLu activation function and the last layer with only 3 neurons and a SoftMax activation function.

The indicator of the results of our implementation offers us an essential reduction of the operating time between 8 to 12 times regarding conventional PC solutions. Respecting the usage of resources, we have obtained 78% of the total DSP available, the majority of the memory is implemented in LUT, and the total LUT used is around 60%, which are excellent marks concerning the viability, considering that our device is a low-cost FPGA. We consider that these results are auspicious regarding the competitiveness of other options as, for instance, the solutions based on GPU.

The second aspect of interest for this work is based on the task that we have wanted to solve with our particular CNN : the classification of the available patients into Healthy, Presymptomatic and Sick with regard to the Spino Cerebellar Ataxia type 2 (SCA 2) disease. One has to be considered that this task is complicated, and so far no other reported classification method has obtained a satisfactory result on this problem (see [8], [9]). In our case, the CNN architecture proposed here admits a margin of optimization for achieving a more efficient differentiation. Otherwise, it should be said that we have implemented the CNN with its parameters already calculated, i.e. these have been obtained from an external off-line training process. As a consequence, a future line will consist in analyzing the possibility of implementing in our device ZYNQ-7000 both the training and the operational tasks.

8 Acknowledgements

The original version of the CNN architecture was proposed by my teammate Roberto A. Becerra García. My tutors Ruxandra Stoean and Catalin Stoean, University of Craiova, and Gonzalo Joya, University of Málaga, have discussed my proposal and they have oriented me about the organization and the writing of my first congress paper. I acknowledge to the authority of the Faculty of Sciences of the University of Craiova for the grateful welcome and the facilitation of my work during my stay.

References

[1] Becerra-García, R,A.; García-Bermúdez, R. V.; Joya-Caparrós, G.; Fernández-Higuera, A.; Velázquez-Rodríguez, C.; Velázquez-Mariño, M. ; Cuevas-Beltrán, F. R.; García-Lagos, F. and Rodráguez-Labrada, R. Data mining process for identification of non-spontaneous saccadic movements in clinical electrooculography. *Neurocomputing*, 250, 28-36 2016. doi = 10.1016/j.neucom.2016.10.077

[2] García-Bermúdez, R. V.; Rojas Ruiz, F.; González Peñalver, J.; Valenzuela Cansino, O.; Velázquez-Pérez, L. and Becerra García, R. A. Evaluation of Electro-oculography Data for Ataxia SCA-2 Classification. *In 10th International Conference on Intelligent Systems Design and Applications (ISDA)*, 50–56, 2010. http://ieeexplore.ieee.org/document/5687258/

[3] Ortega-Zamorano, F.;Jerez, J. M.;Gómez, I. and Franco, L. Layer multiplexing FPGA implementation for deep back-propagation learning. *Integrated Computer-Aided Engineering*, 24, 171-185, 2016. doi=10.3233/ICA-170538

[4] Nurvitadhi, E; Venkatesh, G.; Sim, J.; Marr, D.; Huang, R.; Ong, J. G. H.;Liew, Y. T.,Srivatsan, T.; Moss, D.; Subhaschandra, S. and Boudoukh, G. Can FPGAs Beat GPUs in Accelerating Next-Generation Deep Neural Networks? *In FPGA'17*, 2016. doi=10.1145/3020078.3021740

[5] Nielsen, M. Neural Networks and Deep Learning Dec 2017 Available at `www.neuralnetworksanddeeplearning.com`

[6] Li, Y; Zhao, X; Cheng, T. Heterogeneous Computing Platform Based on CPU+FPGA and Working Modes *2016 12th International Conference on Computational Intelligence and Security (CIS)* doi=10.1109/CIS.2016.0161

[7] Xilinx; Vivado Design Suite User Guide High-Level Synthesis UG902(v2018.1) April 2018. Available at `https://www.xilinx.com/support/documentation/sw_manuals/xilinx2018_1/ug902-vivado-high-level-synthesis.pdf`

[8] Becerra-García, R,A.; Joya-Caparrós, G.; García-Bermúdez, R. Velázquez-Pérez, L; Rodráguez-Labrada, R and Pino-Avila, C Saccadic Points Classification Using Multilayer Perceptron and Random Forest Classifiers in EOG Recordings of Patients with Ataxia SCA2 *Advances in Computational Intelligence (I. Rojas, G. Joya, and J. Cabestany, eds.)*, pp. 115–123, Springer Berlin Heidelberg, 2013.

[9] Becerra-García, R,A.; García-Bermúdez, R. V.; Joya-Caparrós, G.; Fernández-Higuera, A.; Velázquez-Rodríguez, C.; Velázquez-Mariño, M. ; Cuevas-Beltrán, F. R.; García-Lagos, F. and Rodráguez-Labrada, R. Non Spontaneous Saccadic Movements Identification in Clinical Electrooculography Using Machine Learning *in Advances in Computational Intelligence (I. Rojas, G. Joya, and A. Catala, eds.)*, vol. 9095, , pp. 56–68, Springer International Publishing 2015

Carlos CANO-DOMINGO
University of Málaga, SPAIN
ETSI Telecomunicación
University of Craiova, Romania
Faculty of Sciences
E-mail:*carloscanodomingo@gmail.com*

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Autonomous navigation system based on data acquisition

Constantin-Robert Ciolompu

Abstract

Our modern society faces an increasing need for dealing with automation of the surrounding environment, due to a variety of benefits brought to human's professional and personal domain. This article contributes to such effort by presenting a control system of an autonomously exploring robot based on several data acquisition techniques. The data acquired from different hardware components is processed and interpreted in order to obtain an intelligent autonomous working machine with features building upon environment recognition or fault detection. The targeted control system is intended to prove useful in various situations, from industrial purposes to daily needs of the common man.

1 Introduction

The automation phenomenon has been watched with great interest even from the beginning. The most important element that has contributed to the automation process was and continues to be the need of replacing the manual work with the mechanical one, much more efficient and in some cases even cheaper and productive. The automation term itself goes back to 1946-1947 when Ford Motors has developed an automation department in cars production.

The target that people in the industry of automation want to reach represents creating control systems that are capable of operating without or with minimum human intervention. We can all agree that nowadays the automation process has reached a huge level and it impacts people in every aspect of their life. Even more, we all find it normal and in some cases necessary to use these automatic control systems in our daily activities For example, we drink coffee that is prepared by a coffee machine, we use the GPS to get from one point to another, we communicate with other people by using a keyboard and so on. The main idea is that humanity has reached a superior intelligence level that has been used in creating these machines that make everyone's life easier.

2 Description of the project

This article is based on the automated control system concept that works on receiving information and was applied to a rover robot. I decided to use this idea shortly after I realised how big the need of using these automated systems is in the real life. I imagined multiple scenarios in which automated machines

are being used in our society and the project that will be described in this article is one of those scenarios. It can also be interpreted as a response to the need of automated technology in everyone's life and it can be used as an inspiration for future control systems.

Everything around us can be seen as information that is received directly or indirectly. All this information can be received relatively easy from a technological point of view by using sensors. By using these hardware utensils, the control system of the robot will be capable of realising certain calculations and analysing certain situations in order to make the rover move independently in a certain environment/room.

2.1 Used technologies

Given the fact that the Hardware part of a machine needs a software that "makes it alive", the Software part also needs a Hardware environment. For applying the idea presented in this article these two great parts of technology (hardware and software) need to work together. A robot needs it's physical components to ensure its mobility just as much as it needs a system to help it operate in the environment that it's in.

2.1.1 Hardware technologies

The name of the robot that functions by using the control system presented in this article is Rover 5 (displayed in figure 1). This type of robot has been specially designed for students and hobbyist. The addition brought to this machinery is that the clearance can be adjusted by rotating the gearboxes in 5-degree increments. The rubber treads help in equally distributing tension in the motors as the clearance is raised.

- **Robot platform with integrated the quadrature encoders and DC motors.**

Figure 1. Rover 5 robot[1]

The gearboxes have an 87:1 ratio and include an optical quadrature encoders that generates 1000 pulses per 3 complete revolutions of the output shaft.

Specifications of the robot:

- Motor rated voltage: 7.2V
- Motor stall current: 2.5A
- Output shaft stall torque: 10 kg/cm
- Speed : 1 km/hr (max)

[1] Rover 5 robot https://www.generationrobots.com/media/Rover-5-chassis.pdf

➢ **Motor Driver**

The communication between the board which memory is flashed with the control system and the motors and encoders is made using a **4 Channel Motor Control Unit**, which is designed originally for the Rover 5 chassis. Current sensing for each motor allows the processor to determine if a motor has encountered any irregularity on its route. If such event happens, the affected motor will require a higher current as input.

The features of the motor driver are:

- 4x low resistance FET "H" bridge.
- Each channel rated for 4A stall current.
- Easy to use control logic.
- Current monitoring for each channel.
- Quadrature encoder mixing circuitry.

The 4 channels are composed from 3 types of pins:

- **4 x GPIO** - Driving the direction pin high or low will cause the motor to run forward or reverse.
- **4 x PWM -** The PWM pin is used to control the motor speed. When this pin is low, the motor is off. When this pin is high the motor is at full power. To vary the speed of the motor this pin must be Pulse Width Modulated.
- **4 x ADC** – Each channel has a current sensing circuit. The output of these circuits is approximately 1V (sent as digital voltage and is represented by the value 1053) for each amp the motor draws (5V max).

The encoder mixing circuit includes 4 mixing circuits, each one being connected to one of the 4 quadrature encoders. The mixing circuit takes the two inputs from an encoder and mixes them into a single output just as the figure 2 displays. This output represents the number of interrupts the encoder generates and is used to calculate the distance elapsed by the robot.

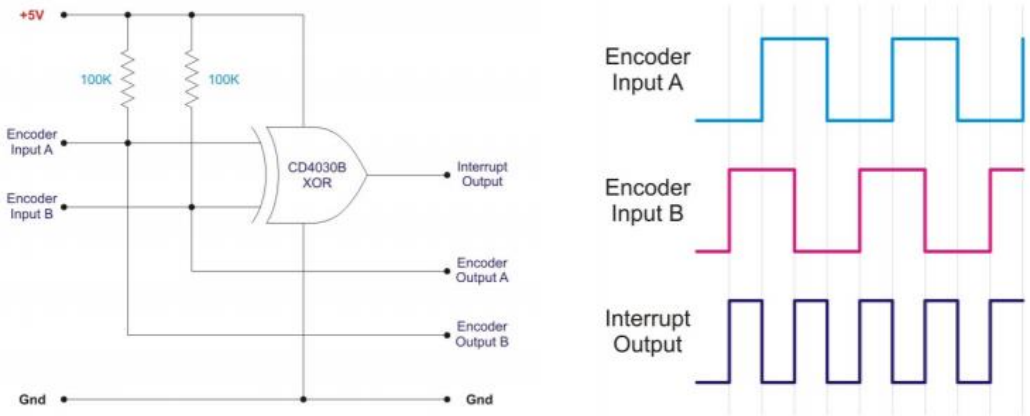


Figure 2. Representation of the input-output process in an encoder mixing circuit[2]

[2] https://cdn.sparkfun.com/datasheets/Robotics/4%20Channel%20instruction%20manual.pdf

Because the interrupt output is twice the frequency of either input it also allows speed and distance to be measured with twice the resolution.

The board has two **power connectors**, one is +5V (Vcc) which will supply the development board and one for the motor power supply. The second one should not be connected to the source without first connecting the Vcc. This device is rated for a maximum motor supply voltage of 12V.

- **Arduino Mega 2560**

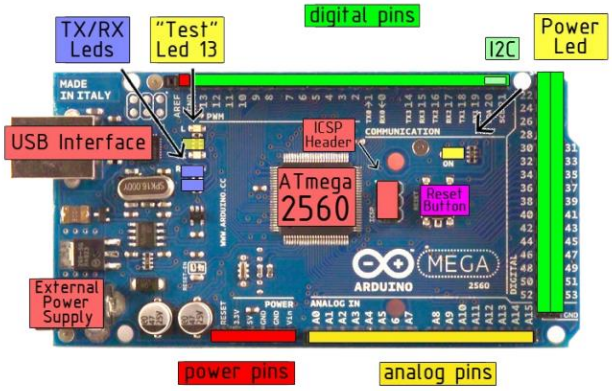


Figure 3. Arduino Mega 2560[3]

The control system of the robot is deployed in the memory of an Arduino Mega 2560 development board. As the figure 3 can say, Arduino Mega 2560 is a perfect environment for embedded developers because of the large variety of ports, the good processing power, the low price and many others.

Specifications of the Arduino Mega 2560 board:

- Microcontroller: ATmega2560
- Operating Voltage: 5V
- Input Voltage (recommended): 7-12V
- Input Voltage (limits): 6-20V
- Digital I/O Pins: 54 (of which 14 provide PWM output)
- Analog Input Pins: 16
- Hardware serial ports (UARTs): 4
- DC Current per I/O: Pin 40 mA
- DC Current for 3.3V: Pin 50 mA
- Flash Memory: 256 KB of which 8 KB used by bootloader
- SRAM: 8 KB
- EEPROM: 4 KB
- Clock Speed: 16 MHz

[3] http://www.mantech.co.za/datasheets/products/A000047.pdf

➢ **HC-SR04 Ultrasonic Sensor**

Figure 4. HC-SR04 ultrasonic sensor[4]

The most important data collector mounted on the robot is the HC-SR04 ultrasonic ranging module. The signals transmitted and received by the sensor are processed in the next manner: when a 10us high level signal is sent to the IO trigger (event displayed on the first signal line in figure 5) the module will automatically send eight 40 kHz (as the second signal line from figure 5) and detect if there is a pulse signal back; if the pulse signal is detected, through high level, time of high output IO duration (displayed on the third signal line in figure 5) is the time from sending ultrasonic to return.

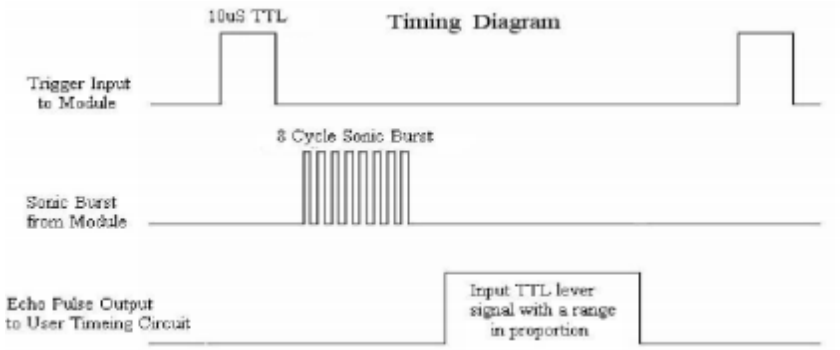


Figure 5. Timing Diagram for trigger and echo signals[5]

In order to calculate the distance to the detected object, one of the next formulas is to be applied: high level time (in uS!) * velocity of sound (340M/S) / 2 or highlevel time (in uS!) / 58 = centimetres (high level time (in uS!) / 148 = Inches)

Specifications of the HC-SR04 ultrasonic sensor:

- Working Voltage: 5V
- Working Frequency: 40 Hz
- Operating distance: 2 cm – 4 m
- Measuring Angle: 30 degrees

The advantages of using the HC-SR04 ultrasonic module: easy to use, 3 mm precision, low energy consumption, usage of only 4 pins (5V pin, Ground pin, 2 digital pins for trigger and echo).

[4] https://www.electroschematics.com/wp-content/uploads/2013/07/HCSR04-datasheet-version-1.pdf
[5] Ibid.

2.1.2 Software technologies

The control system is fully implemented in **MATLAB Simulink**. Also, in order to access the special Arduino Simulink blocks the **Arduino Libraries** are required. The main advantage of using model-based development in creating the functionality is the ease of manipulating the functions. Because MATLAB is a high level interpreted language, programming with it is easier than with C/C++ and other compiled languages, and you can see results from I/O instructions immediately – no compiling. MATLAB includes thousands of built-in math, engineering, and plotting functions that you can use to quickly analyse and visualize data collected from your Arduino.

2.2 Describing the logic of the control system

The control system is created using the following working principle: the rover tries to follow a route with the direction of going forward. If it encounters certain obstacles it will try to go around them by executing 3 specific procedures.

The 3 specific procedures are:

1. Going around to the right side of the object: the robot will take a right turn by changing the direction of going of the small engines (the direction of going of the small engines on the left side will be forward and the one of the small engines on the right side will be the other way around). For a 90 degrees rotation the robot will have to perform a 25 cm arc of circle, value obtained by reporting to the robot's size. After turning right, the movement will be forward for 30 cm in order to find a big enough space that allows the outrunning of the obstacle. When it gets to 30 cm, the robot turns left to detect if the initial road is clear now; if the initial object or a different one is detected with the help of the sonar in less than cm, then this procedure will be repeated. The robot is able to repeat this procedure maximum 5 times. After that, it will consider it reached a dead end.
2. Outrunning the object: if after the left turning executed through the procedure presented above, the route is clear, than the robot will run a distance of 65 cm which is considered enough for outrunning the obstacle initially detected. The number of times that the robot performed the first procedure will be remembered in order to calculate the distance that needed to be covered to go back to the initial route which is done through the 3rd procedure.
3. Re-alignment to the initial route: after finishing the outrunning process, the robot will perform a left turn to verify if it is possible to re-align to the initial route. If the obstacle hasn't been outrun yet a return to the 1st procedure will be allowed and after the execution the robot will go back to procedure number 3 directly. This routine will be performed until the re-alignment to the initial route is possible. In order to get to the point in which the robot has already started avoiding the obstacle, the distance that need to be covered through procedure number 2 will be calculated. Once this distance will be covered, the robot turns right again and will continue straight.

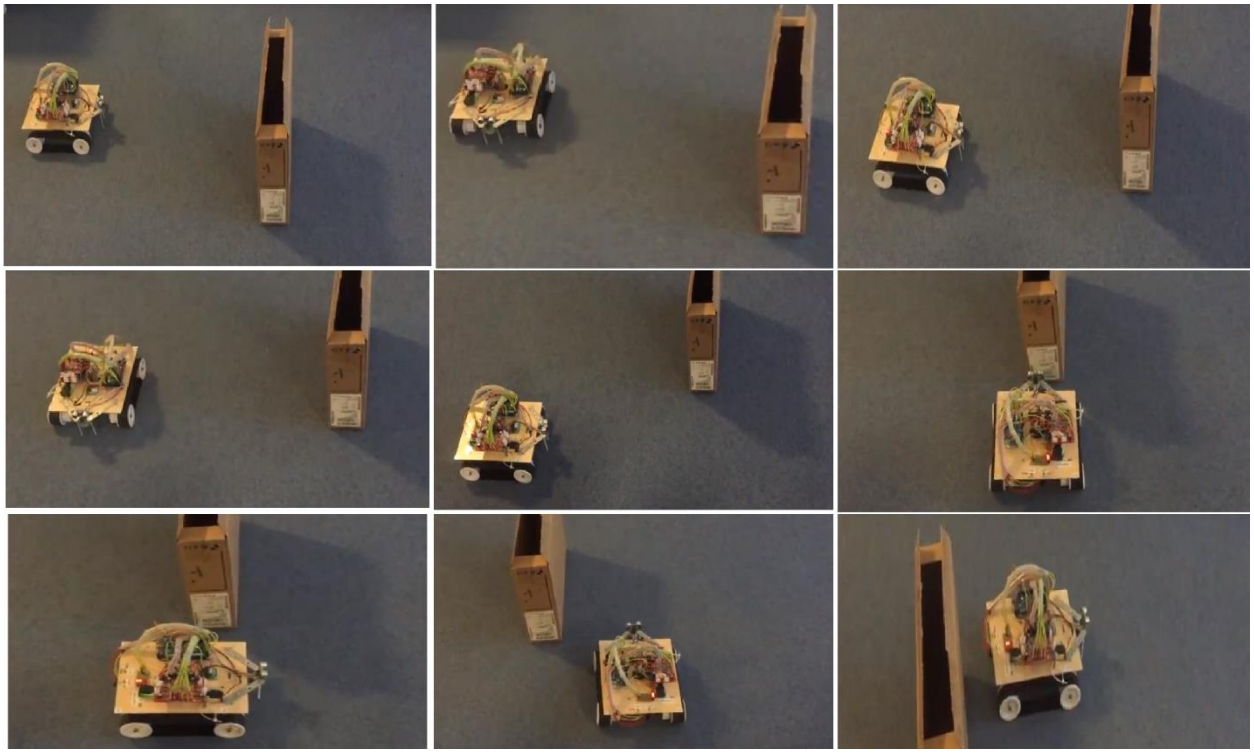


Figure 6. Full avoiding process

It is needed to mention the fact that in all situations in which the robot is moving forward in the process of outrunning the obstacle or obstacles encountered on its route, the ultrasonic sensor will continue to record values, but they will not be used. Instead, it will be verified if the power used by the engines goes over the settled limit (1A). This fact brings both advantages and disadvantages. The benefits will be as it follows: collision availability (instead of keep executing the object avoiding procedures it is better, in some cases, that the robot goes in collision), permitting the robot to continue trying to avoid the obstacle on the initial route (the robot will continue the execution of the avoiding procedures even if after a left or right turn it detects an object that is situated 45 cm from it or less, but not less than the distance that needed to be covered forward, the robot doesn't need to stop after the 5 executions of the procedure 1. The disadvantage is that the power in the small engines may not go over the established level, even if a certain object has been hit or ran over. This can cause the precision of the actions executed in avoiding an object.

The power measuring is done by reading the digital voltage transmitted by the motor driver. Each motor returns the quantity of power used, fact that allows seeing if a collision has occurred. The motor output comes as a digital voltage; for 5V the value of 1023 will be registered and the quantity of power that has been used will be determined by this report. The system has as established level 1A (the scale between volt and ampere is 1:1). In the case of exceeding this level, a counter that ensures the fact that motor has had some issues on its way will be activated. Considering the value that the counter registers, 2 scenarios will be possible: the issue was minor (the counter did not exceed the value 100 and it has been reinitialised with the value 0; it's functionality hasn't been affected) or the issue persists, the counter exceeds the value 100. This will determine the appropriate motor let the

system know that a problem has been detected causing a collision. The collision state will be kept even though the motor power will go under 100 again in order for the motor to go in recovery mode. The counter will be decremented and when the value 0 will be reached, it will let the system know that it can go off collision mode.

The travelled distance will be calculated with the help of the encoders. Knowing that for 3 complete revolutions of the wheel 1000 pulses will be registered, it will be calculated what distance is travelled with each pulse. Using the wheel diameter (6 cm), the circumference will be determined by the formula 2*π*r. The result is that the distance that a wheel has travelled in a complete revolution is approx. 18.84 cm, and after applying The Rule of Three it is determined that for each pulse a distance of 0.05652 cm (5.652 mm) is travelled. Applying these calculations on the number of pulses sent to the system by each encoder, the distance travelled by each wheel will be determined. In order to estimate the distance travelled by the robot the arithmetic mean will be calculated after applying the calculations above, firstly on each side (left, right) and then on the new results.

Detecting obstacles on the route: by measuring the period in which the signal sent by the sonar on the echo pin, the distance until the closest object on the robot's route will be determined. When implementing the control system presented in this article it was settled that the maximum distance towards the closest obstacle that the robot reacts to is 45 cm. The logic behind this value is: in order to be able to go forward without being influenced by the surrounding environment, the robot needs to be able to detect a wide enough space with no obstacles while it's moving. Knowing that the rover is 22.5 cm wide, the angle in which the data are transmitted through the ultrasonic sensor (30 degrees) will be used. It can be said that this particular angle creates an isosceles triangle with a 24 cm base (the robot's width + 1.5 cm margin of error). The median from the top that coincides with the spot from where the ultrasonic signals are actually being sent is the maximum distance that an object that may prevent the robot from going forward can be spotted from. Knowing that the median that starts from the opposite angle of the base is also the bisector in an isosceles triangle, the triangle will be divided in 2 right triangles in which we know a cathetus (24/2=12 cm) and it's opposite angle (15 degrees). The length of the side that represents the maximum distance until the obstacle and the longer cathetus from one of the right triangles previously created will be deduced by using the formula

$$\tan \alpha = \frac{a}{b}$$

where α represents the known angle, a is the opposite cathetus and b *is* the side about the right angle.

In the current case

$$0{,}268 = \frac{12}{b} => b = \frac{12}{0{,}268} = 44{,}77611... \cong 45.$$

The Simulink model created for the implementation of the presented control system runs at a fixed time step of 0,002 seconds in order to obtain a near-real time functionality. The whole control system is dependent of a mechanic switch with 2 states: on and off.

3. Conclusion and future work

The control system presented above can represent an alternative solution for the issue that comes with the constant need of automation of the environment that we live in. The benefits brought by this process of bridging tasks from the human beings to the machines can be countless, no matter if we are talking about personal life or professional life of a normal individual. Given the fact that the information became so much easier to reach using technology, the control system described in this article it's a good example of using data as distance, measuring results and many others.

Future work that has already been planned:

- improving the obstacles avoiding process by adding more ultrasonic sensors on the sides and on the edges of the robot.
- Attaching a video camera to the robot in order to be able to distinguish the obstacles on the route. However, the camera should not obstruct the robot's movement.
- Attaching a Bluetooth module and creating an android app through which the robot would be controlled from the distance.

2. Bibliography

[1] Rover 5 datasheet, https://www.generationrobots.com/media/Rover-5-chassis.pdf (retrieved 05/03/2018).
[2] Channel Motor Driver datasheet, https://cdn.sparkfun.com/datasheets/Robotics/4%20Channel%20instruction%20manual.pdf (retrieved 05/07/2018).
[3] Arduino Mega 2560 datasheet, http://www.mantech.co.za/datasheets/products/A000047.pdf (retrieved 04/27/2018).
[4] HC-SR04 ultrasonic ranging module datasheet, https://www.electroschematics.com/wp-content/uploads/2013/07/HCSR04-datasheet-version-1.pdf (retrieved05/11/2018).
[5] Trigonometry formulas, https://www.mathopenref.com/trigtangent.html (retrieved 05/14/2018).

Constantin-Robert CIOLOMPU
"Lucian Blaga" University of Sibiu
Faculty of Sciences
Str. Dr. Ion Ratiu nr.5-7 550012 Sibiu
ROMANIA
E-mail: robert.ciolompu@yahoo.com

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Decompression Algorithms – RGBM and VPM, a comparative approach

Alexandru Crăciun

Abstract

Modern hyperbaric diving theory has evolved much since the invention of high performing computation machines and the rapid miniaturization of technology. The last decades of the 20th century proved to be a turning point in the understanding of hyperbaric physiology and the mechanisms of organic decompression, this field of study providing a great opportunity for physicians, mathematicians and engineers alike to develop mathematical models for decompression prediction following standardized algorithms. This current document aims to delve into the fascinating field of diving theory, by allowing a concise comparison between two of the most acknowledged and preferred dive-prediction and planning algorithms in usage by commercial, military and recreational divers: *Suunto Fused RGBM* and *HHS VPM-B*. By the use of 5 theoretical diving scenarios (further expanded into 2 sub-scenarios, by considering different equipment parameters), the algorithm's results are being compared and contrasted, allowing the reader to form a rather broad image on the algorithm's capacities and possible shortcomings, emphasizing on the different approaches towards diving practice that the models suggest when using them as a sole mean to predict a dive's outcome.

1. Introduction

Since the very dawn of civilization, mankind manifested an outright uncanny interest to explore and conquer the lightless depths of the oceans. The very origins of the first breathhold divers are lost in the aeons, tough to this day their legacy survives through the practical appliance of this ancient occupation, as both a profession and – lately – as a past-time.

History speaks of divers as early as the 5th millennium BC, in Mesopotamia, presumably freediving for the sake of providing their communities food, in forms of fish, mollusks and crustaceans; they performed dives without the aid of any equipment. During Greek and Roman times, the first use of a diving bell was recorded. As early as the 13th century AD, the first forms of "modern" diving appliances (goggles) were used by Persians. [1]

The first real rebreather – akin to today's diving paraphernalia – was created around the year 1620 for the use in the first prototype military submarine employed by the British Royal Navy; built by Dutch inventor Cornelis Drebbel. The principle behind it was a chemical reaction involving the heating of potassium nitrate in order to reduce carbon dioxide levels in the breathable air [2] allowing

the crew of the rudimentary submarine to survive longer underwater and not suffer from carbon dioxide intoxication.

In 1878, Henry Fleuss, an English engineer, created the first commercially available breathing apparatus, starting the era of today's closed-circuit diving (CC). His invention consisted of a rubber mask directly connected to a breathing bag containing pressurized copper tanks with a mixture of pure oxygen and nitrogen. As in Drebbel's solution to the carbon dioxide problem, Fleuss used a similar chemical principle to clear off the toxic gas from the recirculated mixture; soaking a hemp rope in a chemical mixture that could absorb carbon dioxide from the gas [3].

Since the time of Fleuss, another approach towards the nature of the equipment was developed - in contrast to the rebreather - the so called open-circuit apparatus. Open-circuit diving (OC) differs from closed-circuit equipment by not recycling the gas mixture from the suspension tanks, instead exhausting all the exhaled gases into the environment. Whereas CC uses a pressurized tank that remains at the given pressure, the OC constantly drops in gas pressure inside the tank. As they are, both methods provide risks and advantages over each other in regards to the hazards the diver using them has to face.

It is of importance to know the backgrounds of those two diving techniques, as the article will make use of them as variable parameters in diving predictions.

Despite the technological advancements of this field's forbearers, one issue still remains unresolved: combating decompression sickness (DCS).

Based on empirical observations and statistical inference, various mathematical models of predicting this physiological state were proposed, one more different from the other. However, as with any mathematical generalization applied to the human organism, the reliability and methods of approach differ drastically.

The intention of this paper is to explain and underline the differences and similarities of two from the most used prediction algorithms in this field.

A similar study was conducted in 2003 by Eric B. Maiken [4], comparing VPM-B with RGBM, trough the usage of the HHS V-Planner software (the same software used for this paper) for VPM-B predictions, and GAP RGBM for the RGBM algorithm plotting. However, despite the large bank of data used by Maiken, no conclusive statement could be drawn as to which algorithm is better suited for the task. Despite this, Maiken has managed to define a rather comprehensive list of differences he has observed in regards to the model's predicting patterns.

With the history part of this article being thus laid out, the following chapter will immerse the reader further into the subject by presenting the main hazard of diving (decompression sickness), so as to lay the foundation for the main subject it wishes to tackle, decompression algorithms.

2. Decompression Sickness

As with any activity dealing with alien environments to the human organism, diving also comes at great risk. Since antiquity, so called diseases such as: "the diver's plays", "burst lungs" or "the sponge fisherman's disease" were known to affect all those working as free divers. For long, no methods of avoiding such inconveniences (sometimes even life-threatening situations) were known. The use of diving bells or pressurized suits were the only way to combat those dangers, however they required expensive equipment.

Modern free-divers are as prone to developing decompression sickness (DCS) today, as they were thousands of years ago.

First of all, what is decompression sickness (DCS)? During and after any hyperbaric exposure, the human body reacts by allowing molecules of inert gases to be transported inside tissue so as to equal partial pressures with the environment (according to Dalton's Law of Partial Pressure) [5]. When one is transferred out of the hyperbaric conditions those gases have the tendency to reach back their energetical equilibrium; that means escaping the organic tissue into the bloodstream by fracturing it, the mechanism being put forward in **figure 1**.

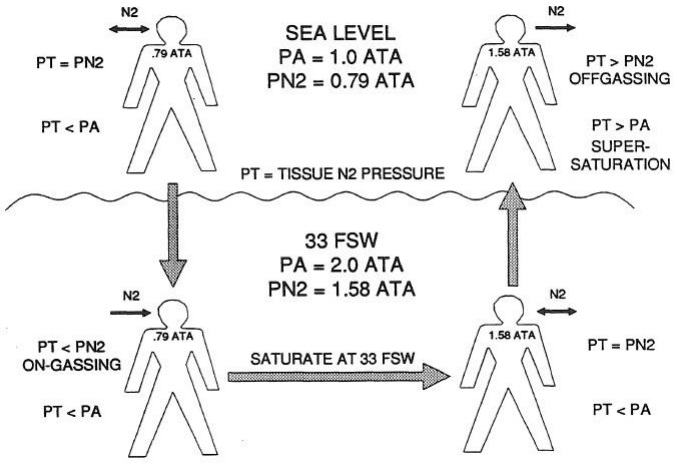


Figure 1 – The Dynamics of DCS [6]

Where **PT** represents tissular partial pressure of nitrogen at any of the 4 given moments, **PN2** represents the normal nitrogen partial pressure in the given environments, **PA** represents the total atmospheric pressure of the environments. The graph has its values adjusted accordingly for sea level altitude (0 m) and -10.06 m below sea level.

For the better understanding of this phenomenon, a pertinent comparison can be drawn to opening a soda can: gas is stored in the solution – the analogue for human tissue; as long as the can remains unopened the pressure remains constant – same goes with human cells; when the can is opened pressures equalize and gas escapes.

DCS is characterized as having the following symptoms, results for the sudden release of gas in the circulatory system:

- Type I DCS: limb and or joint pain, sometimes itching, usually presenting no threat to the diver beside being inconvenient [6] [7]
- Type II DCS: visual disturbances and quadriplegia (as nervous tissue is affected), skin rashes, choking sensation (as lung tissue is struggling to desaturate itself from excess nitrogen) [6] [7]

In common diving practice, as well as in the implementation of the various prediction algorithms the imminent danger considered is in the Type II DCS. As it is the case, the comparisons which will be made in this article will underline when the algorithms compared will predict such possible outcomes of a dive.

3. Diving Algorithms

For long years the diving community has tried to create a prediction model of DCS for general usage, with various degrees of success. Based on empirical observation Swiss physician Albert A. Bühlmann has created a mathematical model of decompression prediction based on his research in the field at the Laboratory of Hyperbaric Physiology at the University Hospital in Zürich, a research conducted since 1959 [8]. Based on the Bühlmann algorithm various decompression tables were developed and used since in diving computers.

The *Bühlmann* – also known as the *Parallel Tissue - model* was based on the assumption that different human tissues will react differently to the formation of saturated gas bubbles in hyperbaric conditions, defining 16 different tissue compartments with their own affinities to Hydrogen and Helium [6] [9], specifically the way gas manages to propagate itself through organic structures.

In contrast to Bühlmann' s parallel tissue model (tissue characterized by capacity to allow gas to diffuse trough cell membranes), there is also a different theoretical decompression model applied in diving computers and simulators, specifically: *The Bubble Nuclei model*.

This approach towards the dynamics of decompression and formation of gas bubbles inside human tissue assumes that inside all human cells there are "centers" of dissolved gas, thus by exposure to a hyperbaric environment, those gas nuclei are capable to amass more substance, forming bubbles and eventually reaching critical mass and rupturing the tissue.

In comparison to the *Parallel Tissue model*, the *Bubble model* assumes that all cells are equally prone to suffer from decompression; the particularities of each tissue being more distinct in regard to the speed of forming gas bubbles than to allowing gas to diffuse trough tissular membranes.

The *Bubble model* allowed mathematical prediction algorithms such as The Varying Permeability Model (VPM) or the Reduced Gradient Bubble Model (RGBM) to be developed as means for simulating decompression risks.

The purpose of this article is to compare the two afore mentioned; for doing such, both of them will be presented.

3.1. The Varying Permeability Model (VPM)

VPM represents the work of D.E. Yount in the field of decompression theory; an algorithm used by commercial and military divers to calculate decompression stops in theoretical dive profiles. The algorithm was developed from 1984 onwards at the University of Hawaii [10] [11].

The algorithm David Yount elaborated aims to create a prediction, based on a theoretically minimized volume of tissue bubbles. It does this by adjusting predicted external pressure to be as large as possible (for the diver), and the inspired inert gas partial pressures to be as low as they can during decompression.

VPM depends on the assumptions that different sizes of bubbles exist inside the human body while in pressurized conditions; the larger the bubbles, the less reduction in pressure they require to begin to grow. The algorithm aims to provide decompression schedules allowing all larger bubbles to be eliminated before tissular rupture can occur [11]. The mathematical fundaments of the algorithm were explained by Corrado Bonuccelli in his work "Calculating Deco Schedule with VPM" [12], as well as by Bruce Wienke in his paper: "Dive Computer Profile Data and Risk Estimators" [13].

The following assumptions and formulae are defined by VMP:

$$\tau_{N_2} = (1,2,5,10,20,40,80,120,160,240,320,400,480,560,720)\ min$$

This array representing the assumed half times (number of minutes in which the number of gas bubbles is halved inside a tissue) for nitrogen in the Yount compartmentation model. For calculating helium half times, the following relationship is used:

$$\tau_{He} = \frac{\tau_{N_2}}{3}$$

For the VPM model, it is assumed that any gas bubble is determined to grow upon reaching a certain critical radius (the minimal growth radius), defined as (ε) [13]. Thus, VPM expounds a radial distribution of bubbles in gel:

$$n = n_0 e^{-\frac{r}{\varepsilon}}$$

Where (n_0) represents an experimental normalization factor and (r) is the actual bubble radius.

The minimal growth radius is directly dependent on the absolute permissible pressure of the environment (P). The critical gas gradient (G) on which VPM calculates decompression stops is defined as following:

$$G = \Pi - P \leq \frac{\gamma}{\gamma_c}\left[\frac{2\gamma_c}{\varepsilon} - \frac{2\gamma}{\varepsilon_0}\right]$$

Where (Π) is the total tissue tension of the inert gas, (γ) is the bubble surface tension, (γ_c) is the surface tension at which bubbles are crushed; (ε_0) being again an experimental constant defined as a gas bubble radius [13].

When applied dynamically with changing variables, the VPM algorithm is defining various stages of decompression stops based on the value of (G), so as to limit (r)'s values, not to exceed those of (ε); 0,5 μm [13].

For the sake of this paper, the implementation of this algorithm will be achieved through the usage of the shareware program: *V-Planner – Deco Planning for Technical Divers*, developed by HHS Software, available from their website [14].

3.2. The Reduced Gradient Bubble Model (RGBM)

RGBM represents a different take on decompression theory in regard to the Bubble modeling. Basing its core on VPM, whilst differentiating itself on a key factor: rejecting the gel model proposed by Yount in his paper: "Gelatin, Bubbles, and the Bends" [10]. RGBM is a quite more complex algorithm, requiring more computer power to be adequately implemented than VPM [15].

RGBM was developed by Bruce R. Wienke and is constantly being improved since 2002, various dive computer manufacturers modifying the algorithm to better fit their own hardware specifications.

The basis on which RGBM is structured can be defined as following: capillary blood flow defines a limit for tissue gas penetration (by diffusion); inside any given tissue there is always an exponential distribution of sizes regarding bubble formation nuclei, being assumed that more smaller bubbles exist than large ones; gas bubbles transfer between any tissular surface boundary regardless of pressure; the tissue compartments have half times ranging from 2 to 300 minutes [15].

Regarding nitrogen compartment half times, they are defined by the following array:

$$\tau_{N_2} = (2,5,10,20,40,80,120,160,200,240,300)\ min$$

Whereas the helium compartments are defined by:

$$\tau_{He} = \frac{\tau_{N_2}}{2.65}$$

Assuming the same premise of keeping the critical gas gradient bellow bubble expansion levels, the following formulas were devised:

$$G\int_{\varepsilon}^{\infty} ndr = (\Pi - P)\int_{\varepsilon}^{\infty}\left[\frac{2\gamma}{r}\right]ndr$$

Where, once again, (Π) represents the total inert gas tension inside the tissue, (n) represents the assumed value of the bubble radial distribution, (ε) represents the critical bubble radius, (r) represents the real radius of the bubble, and (γ) represents the bubble surface tension [13].

By simplifying the equation, the following relationship follows:

$$G = (\Pi - P) \leq \beta e^{\beta\varepsilon}\int_{\varepsilon}^{\infty} e^{-\beta r}\left[\frac{2\gamma}{r}\right]dr$$

Where (β) is a constant value equaling: 0.6 μm^{-1}, as defined for both nitrogen and helium [13].

This paper will be using the Suunto Fused RGBM approach for calculating decompression stops and dive plots, by usage of the program *Suunto DM5*, developed by Suunto Oy (a Finnish company specializing in the production and distribution of specialized diving software and equipment), available for free on their website [16]. The currently used version of the *DM5* software was 1.2.57, a patch released in 26 March 2018.

4. Case Study: HHS VPM-B vs. Suunto Fused RGBM

Following the example laid out by Maiken in 2002 [4], one of the current working hypotheses for the simulations will be presumed as universal: the use of a specific gas mixture (21% oxygen, 0% helium, 79% nitrogen) similar to common breathable air.

The simulations will have different depth and time parameters so as to provide a sufficiently large data variation. Furthermore, for each simulation 2 radically different approaches will be implemented, distinguishing between CC and OC equipment. For each simulation, it will be assumed that the equipment is a standard one, implying the usage of 12-liter gas canisters pressurized at 2,32 technical atmospheres.

As different simulation software was used to implement the diving predictions, for each preferred algorithm a different plotting area will be constructed. Generally speaking, both graphs will have the X axis assigned to a temporal variable, specifically the time elapsed from the start of the dive; whereas the Y axis will track a variable representative of depth, sea-level being conventionally oriented towards the top of the graph. For the RGBM graph, partial pressures of oxygen and nitrogen are also represented in the graph, again the X axis being time dependent, and the Y axis being pressure dependent, the origin value of 0 atmospheres being however oriented towards the top of the graph instead of the standard notation, similar to the depth variable.

The first scenario will consist of a progressive dive to 30 meters, with stops every 15 minutes. The second scenario will consist of a sudden descent to 60 meters and a dive time of 30 minutes. The third

scenario will consist of a 30 meter dive and an ascent to 5 meters followed by another dive to 10 meters. The fourth scenario will consist of a progressive ascent to 20 meters, with a stop at 10 meters, followed by an ascent to 5, finalized with the descent at 20. The final scenario will consist of 2 successive descents to 20 meter depths, with an intermediate ascent to 10 meters between the two.

4.1. First Scenario

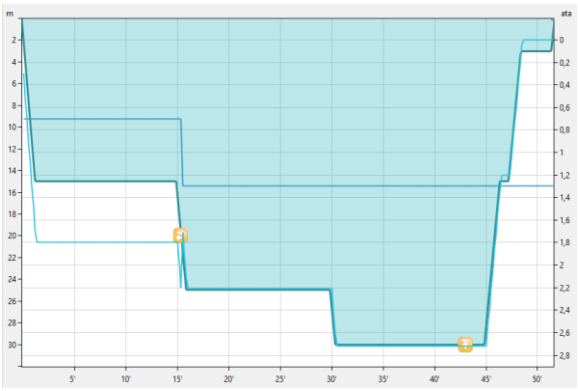
As noted in the paragraph before, this current scenario will include a simple progressive dive towards a depth of 30 meters, with a two accommodation stops at 15 and 25 meters respectively.

At all the three stops, the dive will consider an accommodation time of 15 minutes.

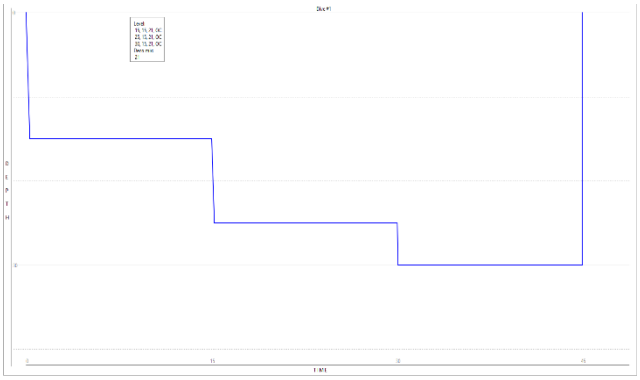
Gas mixture in the tanks will consist of 21% oxygen and 79% nitrogen, for the 12-liter canister pressurized at 2,32 atm.

As it will be the case in all scenarios, the simulation will include 2 different approaches, the usage of both CC and OC diving apparatus implicitly producing different results.

4.1.1. Closed-Circuit



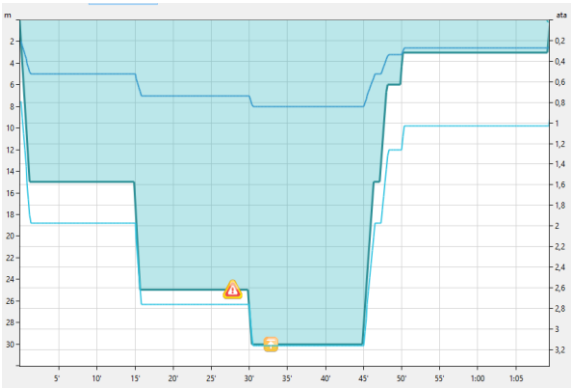
RGBM plot



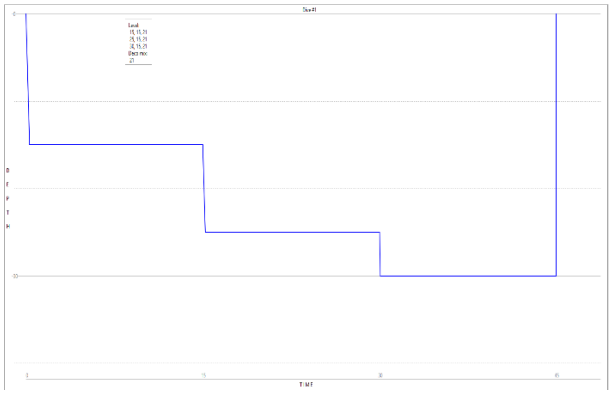
VPM-B plot

The results obtained in this diving instance are somewhat similar, RGBM predicting a longer dive time, at 55 minutes, whereas VMP, suggests a mere 45-minute dive. Another difference that could be noted from the 2 plotted graphs is the steep ascent that VPM predicts to be safe, suggesting that the dive shows no danger of developing DCS risk. RGBM, suggests a 10-minute-long ascent with short stops at 15 and 3 meters respectively.

4.1.2. Open-Circuit



RGBM plot



VPM-B plot

In this case, a very different RGBM plot is observable from the one suggested by VPM, with significant differences in both dive times (110 minutes for RGBM and 45 minutes for VMP) as well as in actual ascent approach; VPM being exactly the same as it was in the CC case.

For RGBM a progressive ascent is noted, with decompression stops at 15, 6 and 3 meters.

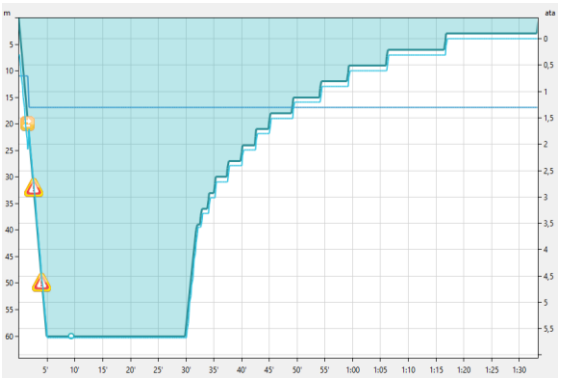
4.2. Second Scenario

This current scenario is mathematically less complex than the one presented just before, consisting of a simple dive towards a depth of 60 meters, without accommodation stops. At the bottom stop, the dive will consider a stagnation time of 30 minutes.

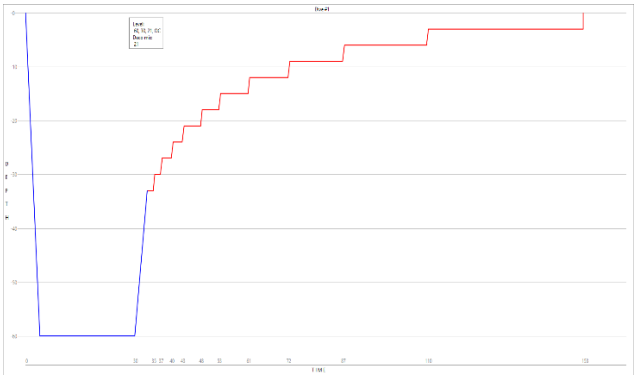
As before, gas mixture in the tanks will consist of 21% oxygen and 79% nitrogen, for the 12-liter canister pressurized at 2,32 atm.

As it will be the case in all scenarios, the simulation will include 2 different approaches, the usage of both CC and OC diving apparatus implicitly producing different results.

4.2.1. Closed-Circuit



RGBM plot

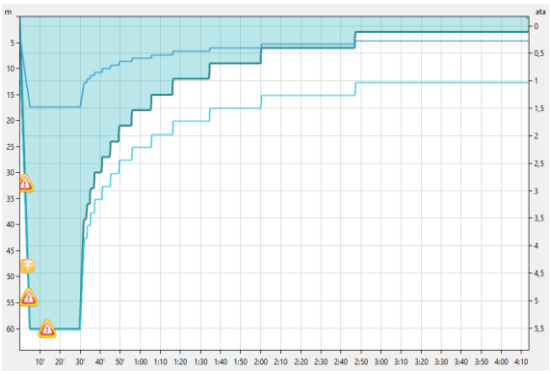


VPM-B plot

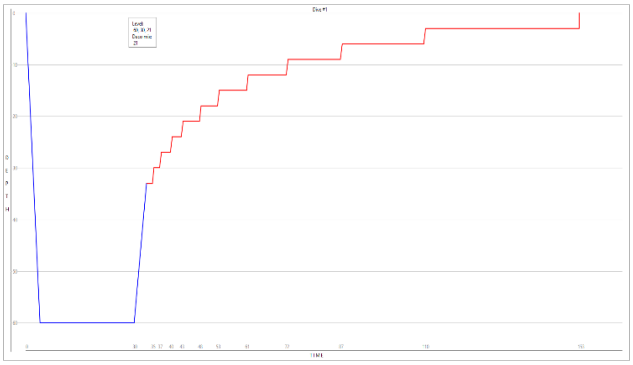
The results obtained in this diving simulation are somewhat similar to each other, RGBM predicting a shorter dive time on this instance, at 95 minutes, whereas VMP, suggests a longer dive at 153 minutes long.

Both VPM and RGBM suggest a progressive ascent, RGBM aiming for an ideal of 12 decompression stops whereas VPM suggests only 11 decompression stops. From a depth perspective, those stops are overlapping, however, depending on the algorithm they are in the case of VPM longer lasting (in minutes) than for RGBM.

4.2.2. Open-Circuit



RGBM plot



VPM-B plot

In this case, as in the first scenario, the RGBM plot is observably different from the one suggested by VPM; mostly for the suggested dive lasting much longer. There are significant differences in both dive times: 245 minutes for RGBM and a mere 153 minutes for VMP. For the actual ascent approach; VPM being once again exactly the same as it was in the CC case. RGBM suggests again a 13 decompression stops approach, however, the accommodation times being this time significantly longer for each stop in comparison to the CC model.

4.3. Third Scenario

This current scenario is somewhat mathematically more complex than the two that preceded it, consisting of a progressive dive towards a depth of 30 meters, with two 10-miunte long accommodation stops at 10 and 20 meters respectively. At all the lowest depth, the dive will consider a 10-minute stop, followed by an ascent towards a depth of 5 meters, where 10 minutes will be spent. The final descent of the dive will be done at 10 meters, where the dive stops once again for 10 minutes.

As in the scenarios before, gas mixture in the tanks will consist of 21% oxygen and 79% nitrogen, for the 12-liter canister pressurized at 2,32 atm.

As it will be the case in all scenarios, the simulation will include 2 different approaches, the usage of both CC and OC diving apparatus implicitly producing different results.

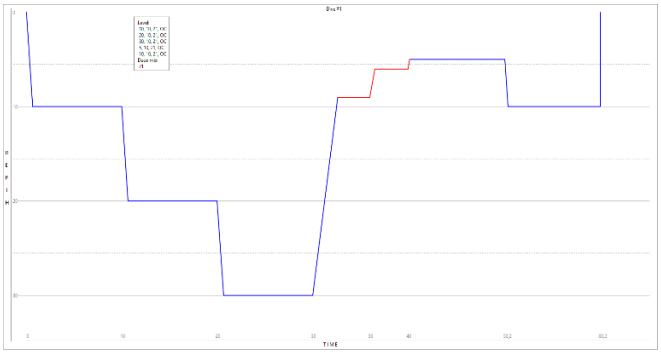
4.3.1. Closed-Circuit

The results obtained in this diving instance are once again somewhat similar, RGBM predicting a slightly longer dive time, at 70 minutes, whereas VMP, suggests a 60-minute dive. Another difference that could be noted from the 2 plotted graphs is the steep ascent that RGBM predicts from 30 to 5

meters in depth. In comparison, VPM suggests a progressive ascent towards the 5-meter depth stop, with 2 accommodation plateaus.

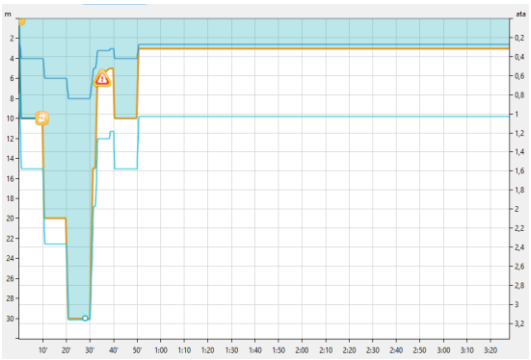
However, a difference of planning can be noted in regard to the final ascent, after the 10-meter dive; RGBM suggesting a few minutes of decompression at 3 meters, whereas VPM assumes that the dive can be suddenly ended after the last descent.

RGBM plot

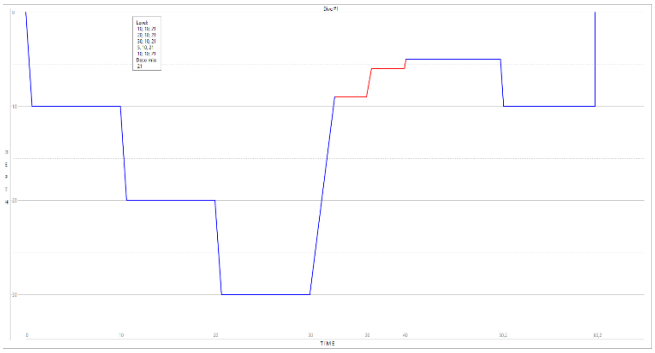


VPM-B plot

4.3.2. Open-Circuit



RGBM plot



VPM-B plot

In this case, as it happened in the last scenario, the VPM predicted dive simulation is similar in the OC case to the one in the CC case.

The RGBM plot is considerably different from the VPM one, the main difference observed being the total length of this simulation (spanning 210 minutes), compared to a much shorter VPM dive (at 60 minutes long). RGBM assumes after the final descent a very long decompression stop at three meters, whereas VPM would advise the diver to ascend directly from 10 meters back to the surface, without any decompression stop.

A particularity manifested in this specific simulation is the warning sign that RGBM issues, at the ascent towards 5 meters, the diver being in an imminent danger of suffocation as the pressure in the rebreather tank becomes smaller than the one of the environment.

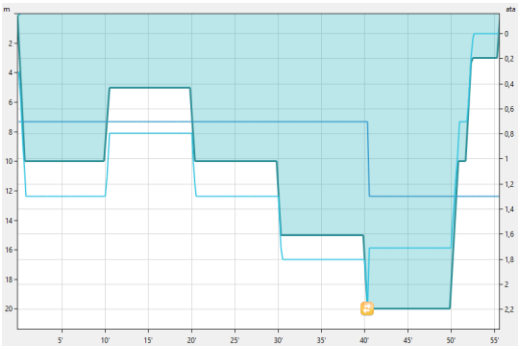
4.4. Fourth Scenario

This current scenario is similar to the previous scenario, as again it presents a more complex approach, consisting of a direct dive towards a depth of 10 meters (with a 10-minute stop), followed by a renascent towards 5 meters (with another 10-minute stop). From 5 meters onwards, the dive plans a progressive dive towards 20 meters, with two 10-miunte long accommodation stops at 10 and 15 meters respectively. At all the lowest depth, the dive will consider a 10-minute stop.

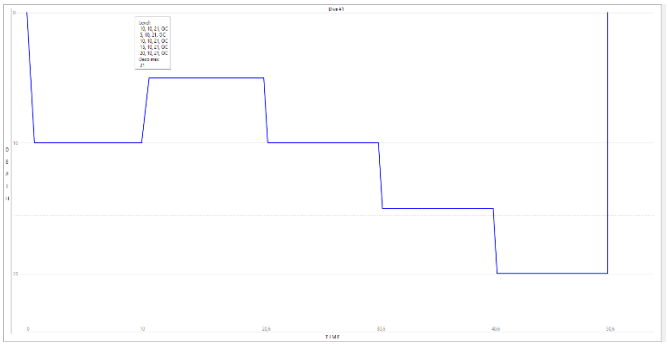
As in the scenarios before, gas mixture in the tanks will consist of 21% oxygen and 79% nitrogen, for the 12-liter canister pressurized at 2,32 atm.

As it will be the case in all scenarios, the simulation will include 2 different approaches, the usage of both CC and OC diving apparatus implicitly producing different results.

4.4.1. Closed-Circuit



RGBM plot

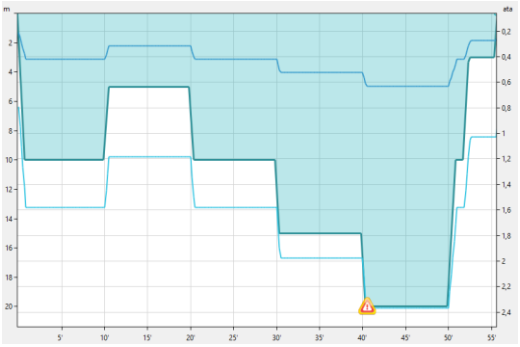


VPM-B plot

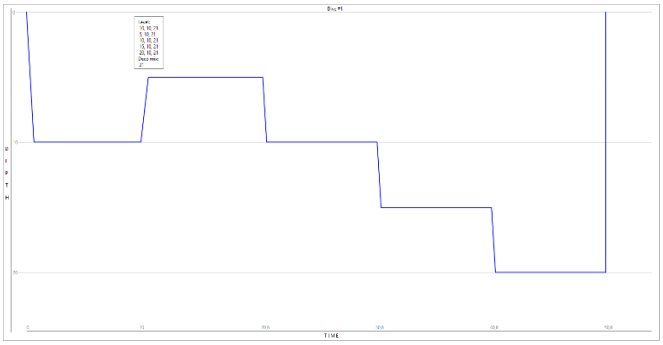
The results obtained in this diving instance are somewhat similar - in the same way they were in the first scenario - RGBM predicting a longer dive time, at 58 minutes, whereas VMP, suggests a closer 50-minute dive.

One difference that could be noted from the 2 plotted graphs is the steep ascent that VPM predicts for the last ascent, suggesting that the dive proves no danger of developing DCS if the will decide to follow through a direct return to the surface after the completion of the scenario. RGBM, suggests a 10-minute-long ascent with short stops at 10 and 3 meters respectively.

4.4.2. Open-Circuit



RGBM plot



VPM-B plot

In this case, RGBM's plot somewhat similar to one suggested by VPM, with no significant differences in both dive times (55 minutes for RGBM and 50 minutes for VMP) as well as in actual ascent approach; VPM being exactly the same as it was in the CC case.

For RGBM a somewhat more progressive ascent is noted, with very short decompression stops at 10 and 3 meters.

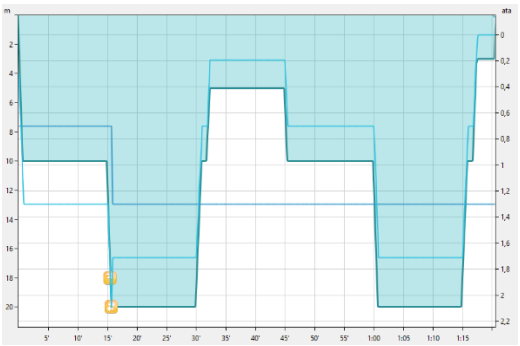
4.5. Fifth Scenario

As it was the case with the two previous scenarios, this current one being once again mathematically more complex, this time consisting of two progressive dives towards a depth of 20 meters (with stop times of 10 minutes), with 10-miunte long accommodation stops at 10. Between the two descents there will be a renascent to 5 meters (with a 10 minute accommodation).

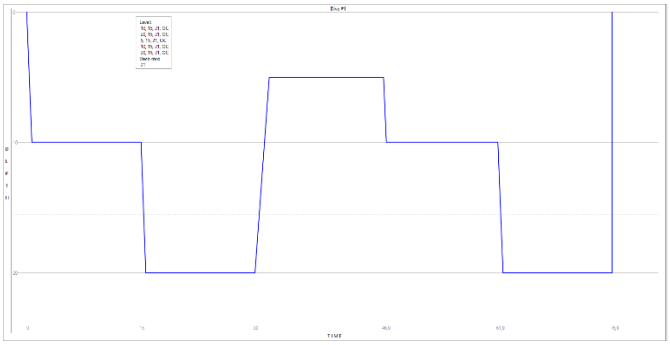
As in the scenarios before, gas mixture in the tanks will consist of 21% oxygen and 79% nitrogen, for the 12-liter canister pressurized at 2,32 atm.

As it will be the case in all scenarios, the simulation will include 2 different approaches, the usage of both CC and OC diving apparatus implicitly producing different results.

4.5.1. Closed-Circuit



RGBM plot



VPM-B plot

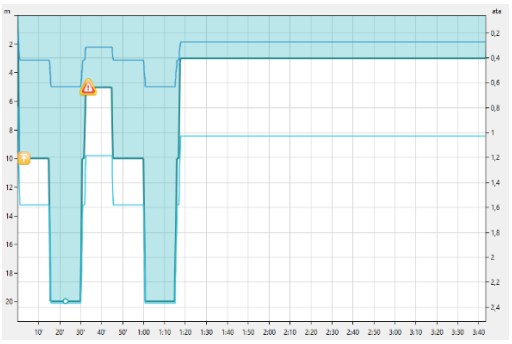
Once again, the results obtained in this diving instance are somewhat similar - in the same way they were in the first scenario and the fourth one - RGBM predicting a longer dive time, at 80 minutes, whereas VMP, suggests a very close 77-minute dive.

One difference that could be noted from the 2 plotted graphs is the steep ascent that VPM once again predicts for the last ascent. As in the previous examples RGBM, suggests a 10-minute-long ascent with short stops at 10 and 3 meters respectively.

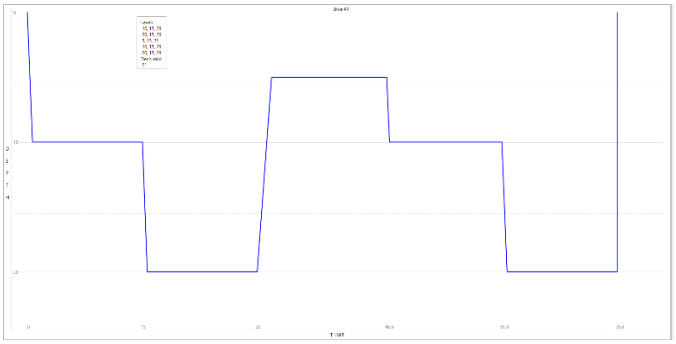
4.5.2. Open-Circuit

In this case, RGBM's is radically different to one suggested by VPM, with differences in both dive times (225 minutes for RGBM and 77 minutes for VMP) as well as in actual ascent approach; VPM being exactly the same as it was in the CC case, as it happened in all scenarios until now.

For RGBM a sudden ascent is noted, a very long decompression stop at 3 meters. At 5 meters, the algorithm once again signalizes a suffocation risk due to low tank pressures.



RGBM plot



VPM-B plot

5. Conclusions and future work

The previous section represented the core of this paper, by making true the comparison between the two diving algorithms: *Suunto Fused RGBM* and *HHS VPM-B* it wanted to compare. The case study's aim was accomplished, the algorithm's results being different from one another, despite sharing a few similarities in some of the discussed simulations.

The following conclusions could thus be drawn:

- VPM manifests same surface ascent times for both CC and OC dives, regardless of other parameters;
- The RGBM algorithm surface ascent times are always longer in cases of OC type dives in comparison to CC dives.
- Suunto Fused RGBM is a more sensible algorithm than VPM, providing more data in regards to the risks involved during the simulations, warning the user when danger was imminent.

As was noted in Maiken's work comparing the RGBM and VPM algorithms [4], the same conclusions were drawn in regards to the ascent times. However, since the time his work was published (2003), software implementation solutions were further developed, Suunto's DM5 proving to be a more sensible software in regards to risk prevention than the old HHS program.

For further development of this study, future works can be initiated by developing more diving scenarios and using different gas mixes in the simulations than the standardized air concoction.

From a programming perspective a long-term development plan that was taken into consideration will be implementing the two decompression algorithms into an independent software, linking the two used software into a newly developed one, for the purpose of providing any would-be divers with a convenient solution to calculating their decompression stops.

References

[1] R. F. Marx, The History of Underwater Exploration, Courier Corporation, 1990.

[2] B. McLaughlin, „Special - Cornelius Drebbel, inventor of the submarine," 30 05 2012. [Interactiv]. Available: https://archive.is/20120530025638/http://www.dutchsubmarines.com/specials/special_drebbel.htm.

[3] D. Quick, A history Of Closed Circuit Oxygen Underwater Breathing Apparatus, Royal Australian Navy, School of Underwater Medicine, 1970.

[4] E. B. Maiken, „VPM Technical Publications,” 2003. [Interactiv]. Available: http://decompression.org/maiken/VPM/Correlation/VPM-Bv3.2_vs_GAP_RGBM_and_GF_100ft_Air_Dives.pdf. [Accesat 12 April 2018].

[5] J. Dalton, „Essay IV. On the expansion of elastic fluids by heat,” *Memoirs of the Literary and Philosophical Society of Manchester, vol. 5, pt. 2,* p. 595–602, 1802.

[6] K. E. Huggins, The Dynamics Of Decompression - Workbook, Ann Arbor, Michigan: The University of Michigan, 1992.

[7] S. A. Pulley, „Decompression Sickness,” 12 July 2016. [Interactiv]. Available: https://emedicine.medscape.com/article/769717-overview.

[8] A. A. Bühlmann, Decompression - Decompression Sickness, Berlin: New York Springer-Verlag, 1984.

[9] A. A. Bühlmann, Tauchmedizin: Barotrauma Gasembolie Dekompression Dekompressionskrankheit, Berlin: Berlin Springer-Verlag, 1992.

[10] D. E. Yount, „Gelatin, Bubbles, and the Bends,” în *American Academy of Underwater Sciences Eleventh Annual Scientific Diving Symposium*, Honolulu, 1991.

[11] D. E. Yount și D. C. Hoffman, „On the use of a bubble formation model to calculate diving tables,” *Aviat Space Environ Med.; no.57,* p. 149–156, 1986.

[12] C. Bonuccelli, „Calculating Deco Schedule with VPM,” 2004.

[13] B. R. Wienke și T. R. Leary, „Dive Computer Profile Data and Risk Estimators,” ResearchGate, Los Alamos, 2017.

[14] HHS Software, „V-Planner - Deco Planning for Techincal Divers,” 2018. [Interactiv]. Available: http://www.hhssoftware.com/v-planner/download.php.

[15] B. Wienke, „Reduced Gradient Bubble Model,” 05 February 2002. [Interactiv]. Available: http://www.scuba-doc.com/rgbm.html.

[16] Suunto Oy, „Suunto DM5 - Download,” 2018. [Interactiv]. Available: http://www.suunto.com/dm5/. [Accesat 12 April 2018].

Alexandru CRĂCIUN
UMF „Iuliu Haţieganu” Cluj-Napoca
Facultatea de Medicină
400349 Cluj-Napoca, Strada Louis Pasteur nr. 4, et. 1
ROMÂNIA
E-mail: craciunalex98@yahoo.com

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Services marketplace web app

Crișan Gheorghe-Cătălin

Abstract

The goal of this article is to propose and present a web application that aims to help peoples to find local professionals or craftsmen to solve their day-by-day problems. Based on two-sided marketplace concept, the application brings together customers and service providers making easier and faster for peoples to search experts near their location and match the best one comparing their rating and reviews. The app also, helps freelancers and local businesses (large and small) to grow and find new customers. By using two-sided marketplace concept, communities get self-balancing by lowering the unemployment rate and growing their economy. Adopting freemium pricing strategy, service providers can increase customer retention rate by promoting their services on top of search results. Another goal of this paper is to present the architecture and functionalities of this application based on MEAN stack: a powerful open-source JavaScript software stack that help to build scalable soutions by using single language development for back end and front end – flexible, faster and reusable.

1 Introduction

Nowadays the internet has become the fastest and most effective way to share information. Thus, through the Internet, we can bring professionals from different fields together with people who need their services. Usually one of the most difficult problems a person encounters when needs an expert for solving quick and well a problem is the missing of information about local service providers: their names, addresses, field of services they provide, which one of them perform a more efficient, more professional and more quality work, which is the best value / price ratio. This was the main motivation for this application. With the application we propose in this article the user no longer has to waste time or spend even days to look for a suitable person with experience and confidence to solve the problem. Problems that people have to solve are very different: a broken pipe to the bathroom or kitchen, commanding a furniture at demand or assembling it, carrying objects too large for his car or even finding a housekeeper or a teacher for his child. In the rest of the paper we will use the terms “professionals” or “experts” to denote the providers of all kind of services people could need and require.

By using the application, the professionals who are in the proximity of the user are notified via e-mail about new requests and the available experts have the opportunity to offer a personalized offer to each person. Thus, service providers will find new customers having the opportunity to expand their business.

Service providers are required by the application to add a copy of the ID card and optional, copies of the certifications they own. This feature of the application is very useful, be cause by verifying the database, the application can mark a registered service provider as being verified and

trustworthy. By adding certificates, the application gives to customers a double check when hiring a professional: community reviews and previously-verified application certificates for increased confidence.

The application is highlighted compared to the two most popular applications in Romania: OLX (www.olx.ro/servicii-afaceri-colaborari/) and Publi24 (www.publi24.ro/anunturi/servicii/) through the problem-handling mode. Thus, the two listed applications allow companies and individuals to add ads and images in which they specify their services. However, they fail to provide the user with a positive service experience due to the fact that they can not give the user the degree of security and trust they need. The problem is due to the fact that the person providing services can not be verified by the application as concern the identity and the certifications for the services offered, and the users who have interacted with the person can not leave a feed-back. This results in "signing a blank check" between the client and the professional, all based on blind trust.

Another similar application is BlaBlaCar (www.blablacar.ro/) which is intended for the transport of persons. It works somewhat similarly: the professionals are the people who offer transport services and the customers are the ones who use the services offered. The application uses a similar review model with a score of 1 to 5 but without providing a separate rating for the most important aspects such as quality of service and professionalism. However, based solely on transport, BlaBlaCar does not allow the user to find experts in all areas of interest (template, electrician, housekeeper, etc.) in one place.

The main advantage of our application consists in a complex and well-organized structure of services intermediation where each person can provide services or hire someone for certain services. The application supports business development having the effect of increasing the local economy.

Concluding, our application is based on the two-sided marketplace concept [1] and proposes a complex structure for assembling all the services in one place. In this way, it is assured a better interaction between customers and services providers and brings the ease of finding the professional according to the needs of each customer to a new level in Romania.

The rest of the article is organized as follows: the second section presents the application and related information about app. It contains details about structure and functionality alongside with a light introduction to NoSQL. The third section contains information about frameworks integration and how to configure server to communicate with the database and client for layer. In the last section are formulated conclusions and further research directions.

2 Functionality of the application

The web platform's operating principle is the following: the application takes over the user's request, notifies all available professionals in the area about the problem the customer has, and the professionals interested in solving the problem send an offer including also pictures of similarly solved cases. The approach presented above allows each customer to solve any problem in a simple, fast and efficient way.

Service providers could not be notified about some requests in their field of interest due to the settings they have set. For example, the professional travel area does not coincide with the location where the user needs help, the professional has not activated the notification service when new requests appear whitin its area of activity or the professional wants to expand its range of services by receiving notifications from other or additional categories. Therefore, the application also allows service providers to search themselves for requests in different domains. Service providers can use this feature and look for new clients when they are not crowded with other work to do. Thus, depending on their time disponibility, service providers can choose between e-mail notification or manual search for customers.

When a professional accesses the application, the application attempts to locate him through the browser's geolocation service, and if the access to the location is not allowed, his account's default location is used to display active requests in its neighborhood (Fig. 1). Through this utility, the expert is always up to date with all requests. Furthermore, requests can be filtered so that the experts can only see requests that are relevant for his services.

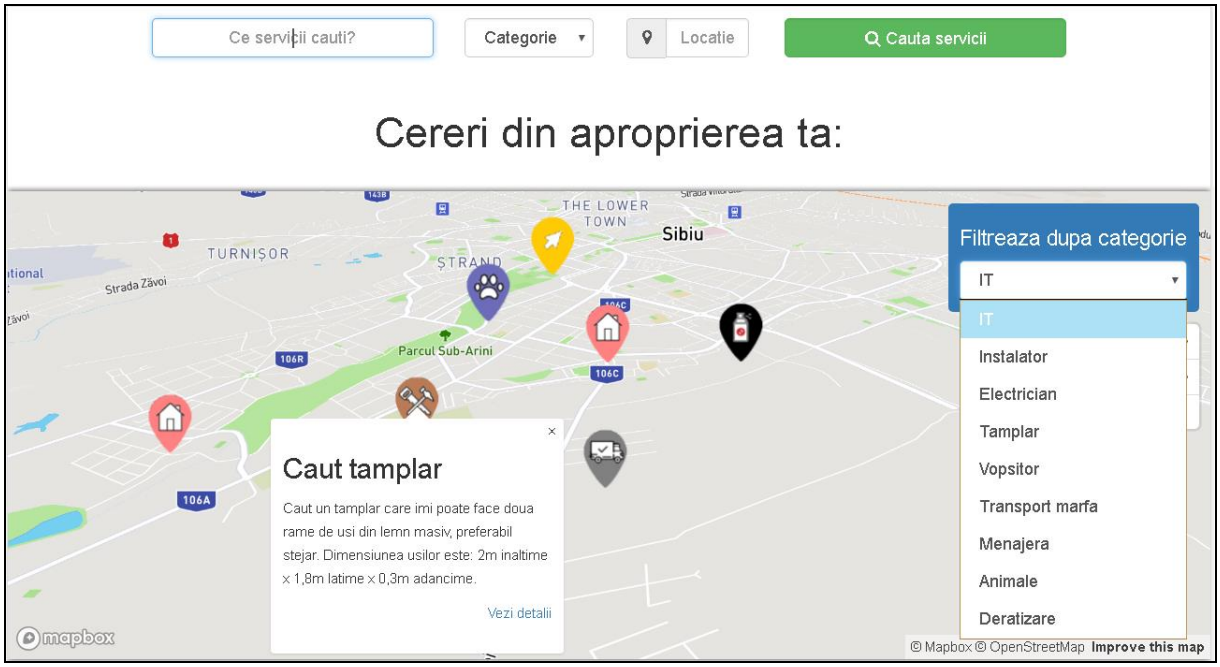


Fig.1 - Interactive map for filtering published applications

By using markers, experts can easily distinguish between types of active requests in a particular area. Filtering requests by category can be done easily. A brief description of the problem can be seen by simply clicking. In this way, experts can very easy check the requests in one area and make the decision. If experts are interested about request, they can send the offer to a client. More, professionals can have an image on the most professionally direct themselves to services that are most sought after in some regions.

Service providers also have a classic search model in which they specify a set of keywords, and optionally the category of services and the location where the service is intended to be provided. In this way, service providers may see a list of requests based on the search criteria they specify (Fig. 2), where each announcement specifies the length of validity and the distance in kilometers from the current location of the professional to the location of the application that has been added. Optionally, the expert can sort the requests after their distance from his location and thus he can optimize his scheduling reducing the travel time from one customer to another.

After chosing the requests he is interested in, the provider can visualize them. Any request includes a set of images meant to clarify the problem to be solved along with a detailed description of the problem. At this point, the experts could send a detailed personalized offer or might recommend other expert using the sharing feature. In Figure 3 the buttons for these two alternatives are displayed (Send offer and Recommend an expert). Taking into account the immediate purpose of the application, for now the language of the application is Romanian, but in the future, translations will be also provided in different languages.

Fig. 2 – List of requests resulting from distance sorting

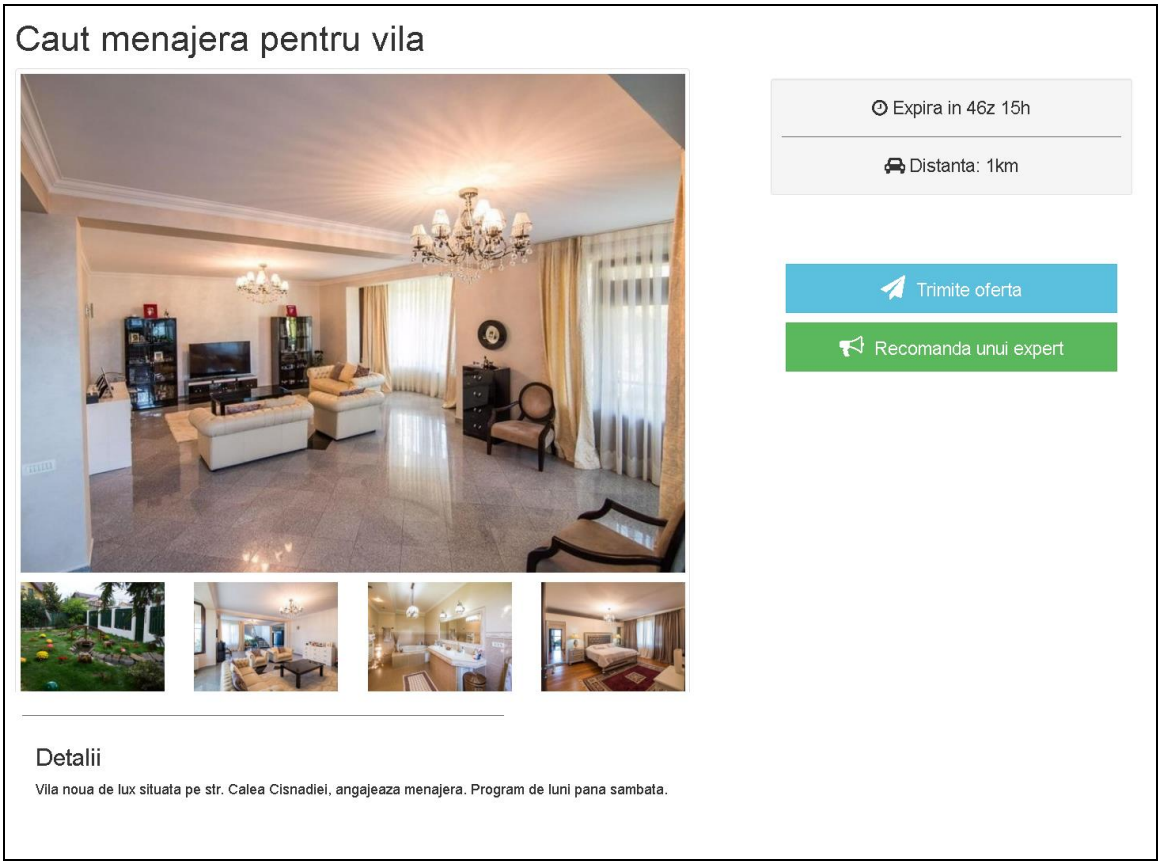


Fig. 3 – Active application details page

When proposing the offer, the professional has the obligation to specify details such as the duration of the work and the cost estimate, as can be seen in Fig. 4.

After sending the offer, the customer receives the notification in his / her account. Here he has access to see the proposed offer together with the price. In addition, in order to help the client to decide, in the client control panel details (Fig. 5) is display expert validation and certification for the specified category of service. In this way the client is encouraged to accept offers only from verified and certified people to get the most enjoyable experience. On the other hand, the

expert is also encouraged to become certified in order to become competitive with other experts in the field and to provide high quality services.

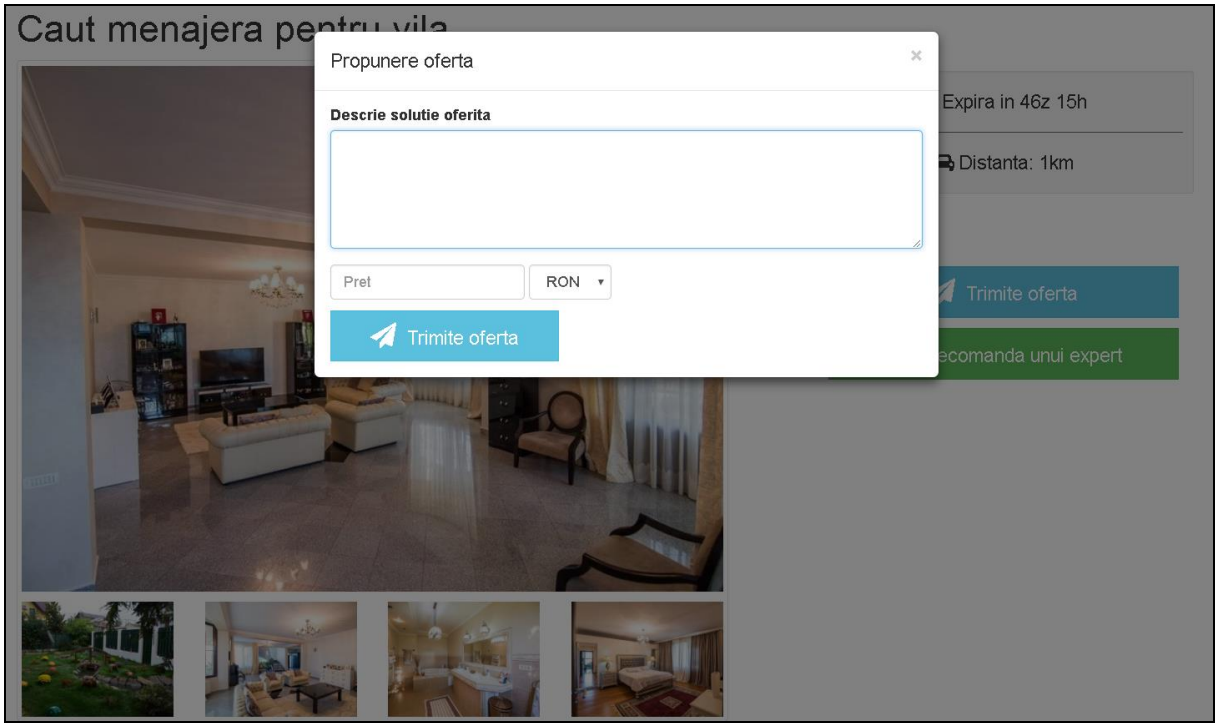


Fig. 4 – Modal window for proposing the client offer

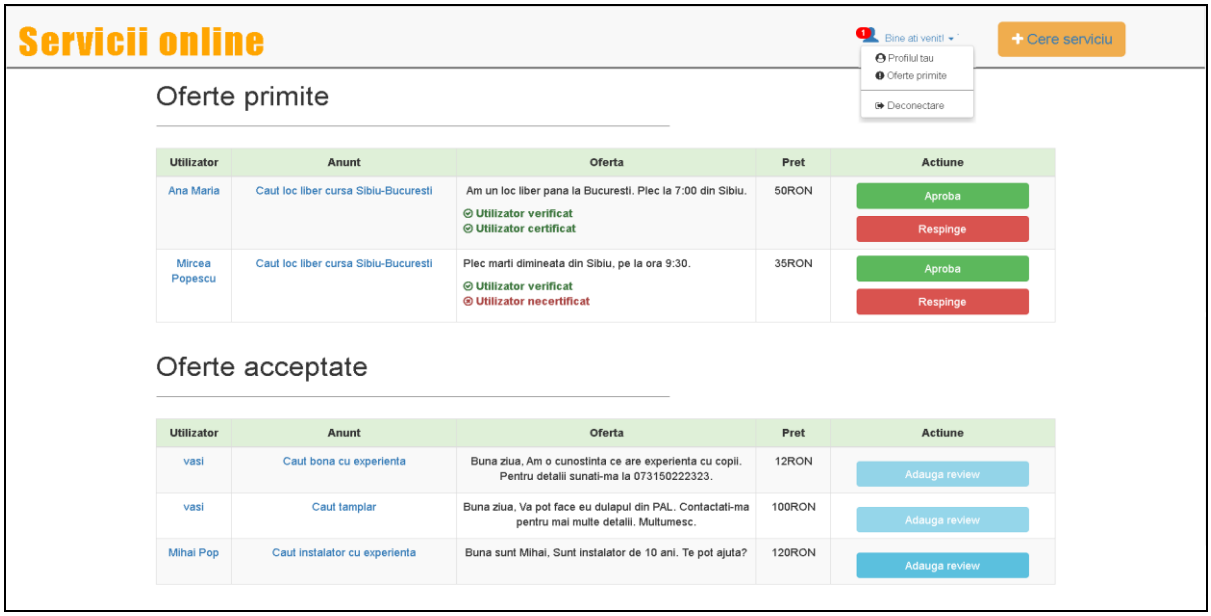


Fig. 5 – Client control panel for incoming requests

If the customer accepts the offer, the expert is notified by e-mail and can see the history of all accepted or pending offers in his / her own account (Fig. 6).

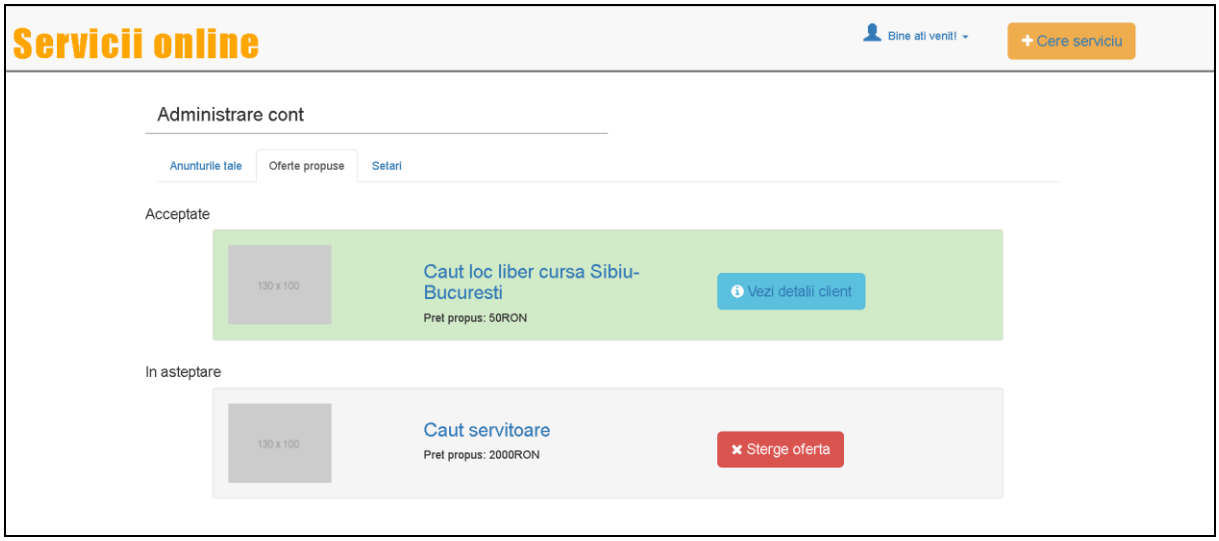


Fig. 6 – Expert control panel for accepted and pending requests

Upon completion of the service by the professional, the client can add a review for this expert to assess the punctuality, quality and professionalism that has been demonstrated until the completion of the execution (Fig. 7). Thus, the application develops with the help of the community a proper ecosystem through which clients can compare different service providers.

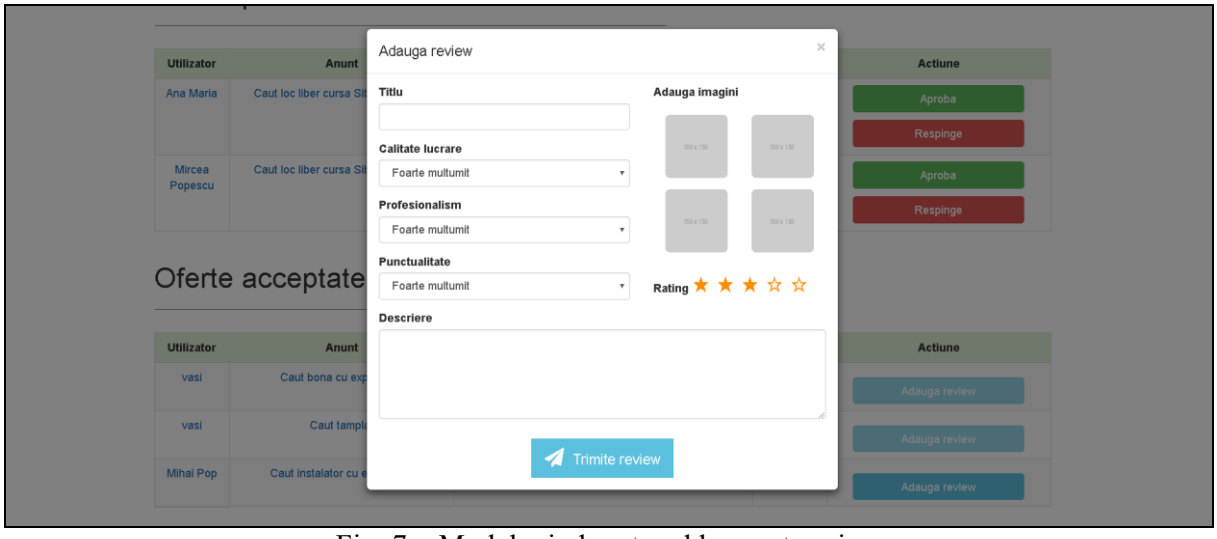


Fig. 7 – Modal window to add expert review

Adding the review is simple and does not take longer than five minutes. The clients provide the feed-back, using a rating score from 1 to 5, and convincing pictures that refer to what the completed service looks like. The reviews goals tohelp the potential clients to accept or reject an offer from an expert. Thus, even if the price is lower, rating and images are considered a primary factor in choosing an expert.

Figure 8 illustrates an expert profile page. Here, potential customers can see the description of skills, the area where the expert agrees to provide services and the reviews given by the other clients who used his services.

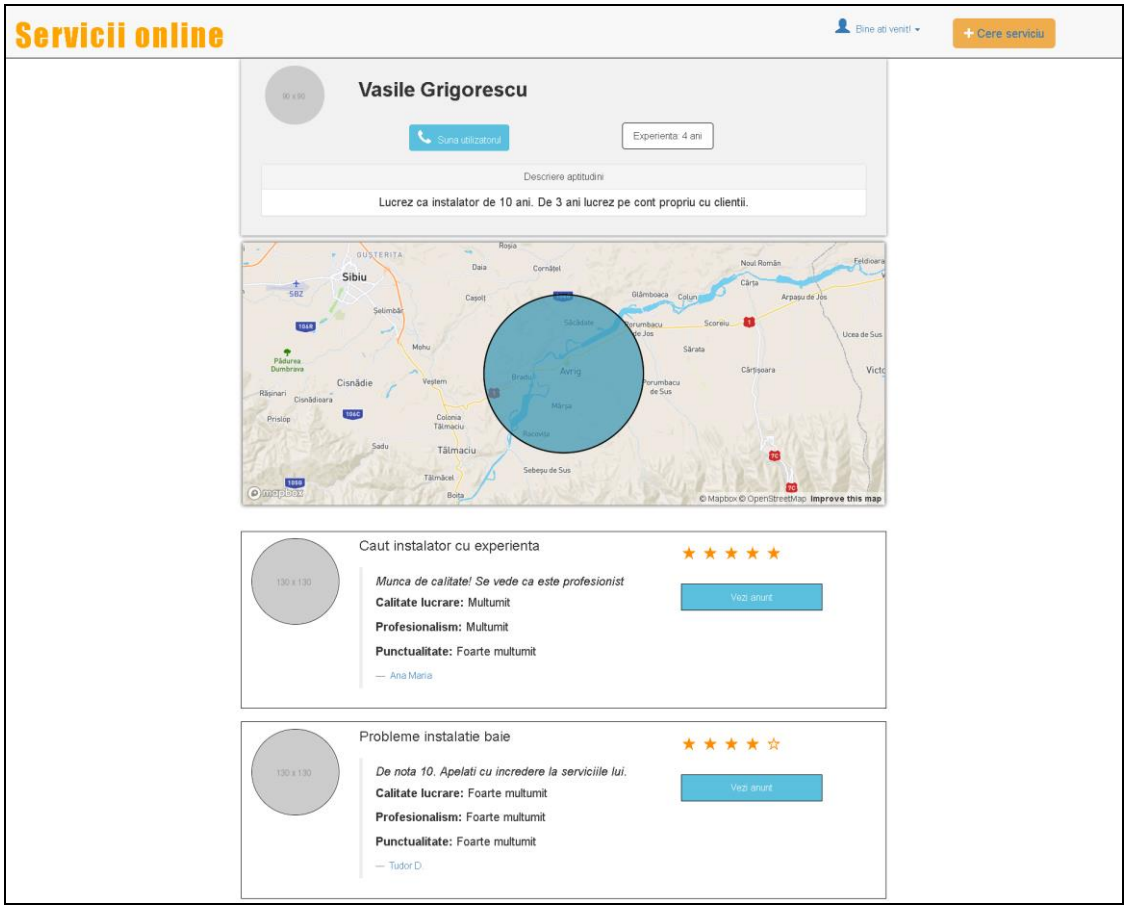


Fig. 8 – Expert's reviews profile page

A client can add a request in very simple way. He has to add a title as descriptive as possible with a detailed description of the problem. If the client adds only a brief descriptionof the problem then the likelihood of being contacted by an expert is small. In addition to these two fields, the client must add the category to which the request belongs, the location of the problem, images that help the experts to identify the problem and the validity period of the request.
In Figure 9 there presented the fields that a customer must complete to request a service.

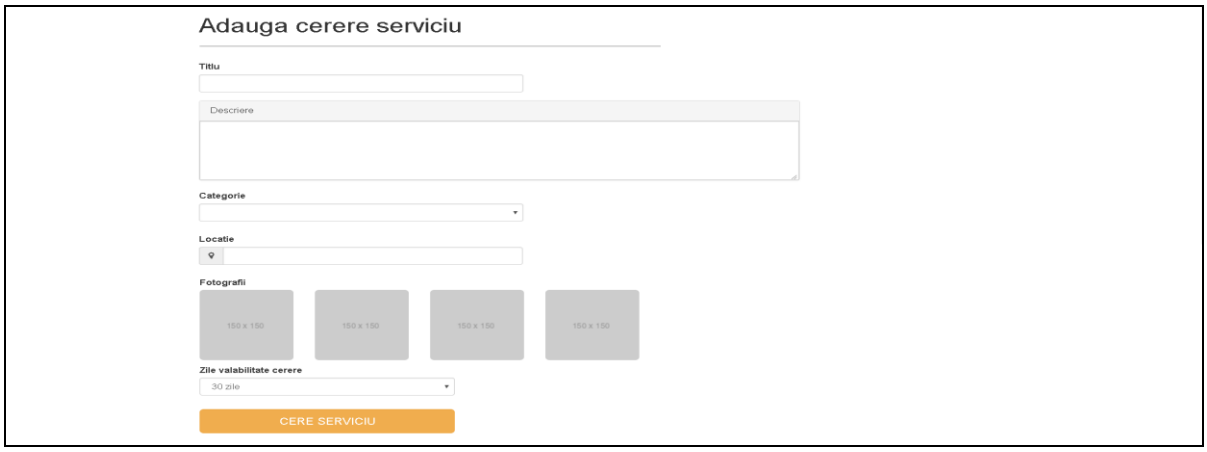


Fig. 9 – Add service request page

An important part of the application is the dashboard (Fig. 10) that allows service providers to activate the e-mail notification service to keep up-to-date with the latest requests from the service categories for which they want to receive notifications. Professionals are targeted for new requests based on the specified headquarters location and the clients' maximum distance accepted from this location. In the dashboard, the experts can add one or more certifications for existing service categories to provide greater credibility for the client and implicitly a greater acceptance rate of their offers.

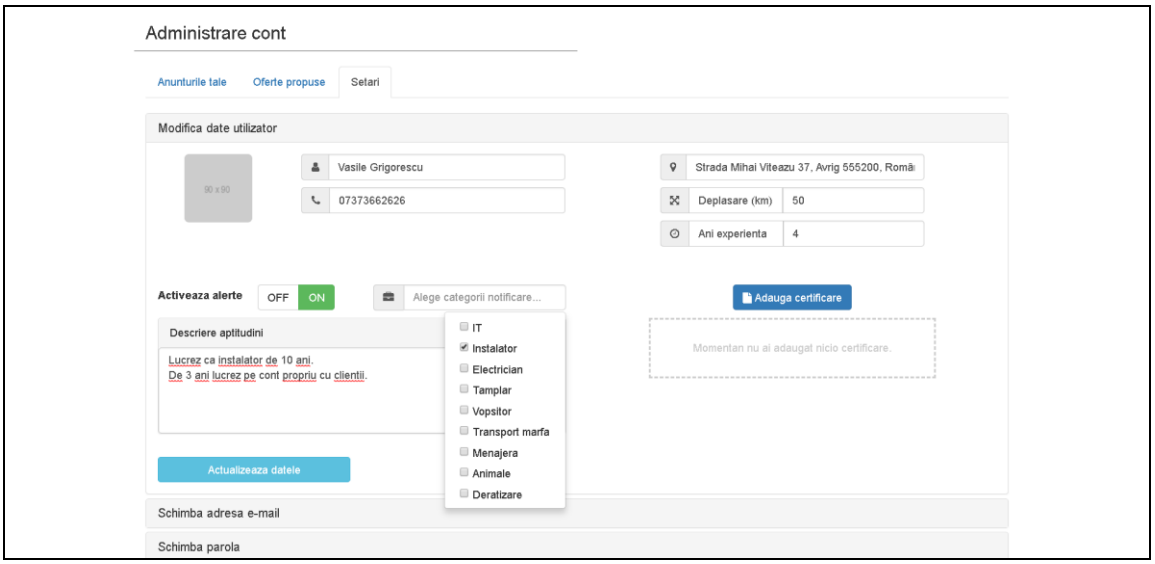


Fig. 10 – Expert dashboard

For ease of communication with experts and clients, the application uses the e-mail service for all types of notifications. Thus, experts are notified when new requests appear, when the offer is accepted or when they receive a new message from the person they are discussing with (Fig.11).

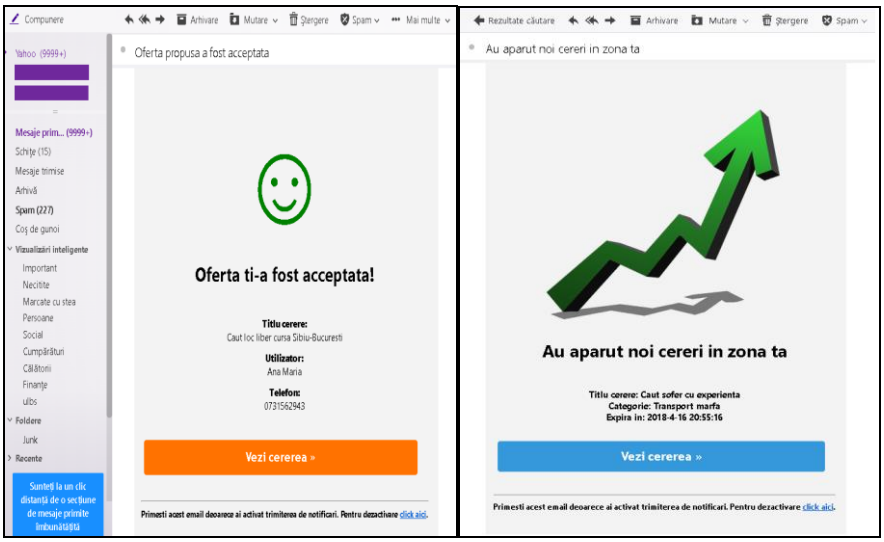


Fig. 11 – Template e-mail for users

3 Application structure

The application is based on MEAN Stack [2] - one of the web solutions that allow you to make web applications fast, robust and easy to maintain. The service package consists of MongoDB, Express.js, Angular and Node.js and is based on JavaScript technology. It is one of the full-stack development packages, that is, it is used on the back end and front end. It has grown a lot in popularity over the past years [2] because it uses one programming language. So, it does not need other languages like PHP or Java for back end and MYSQL for the database.

MEAN Stack is easy to use and to implement, which is due to the communication that exists between the three main layers of the application.

The first layer is represented by the UI or the client browser using Angular technology. Using Angular the written code becomes reusable by writing it in the form of components. Each component has its unique role that we can control and that allows us to integrate it together with other components resulting in a basic component that incorporates all the other components. This makes AJAX requests to the server that is the second layer. The server is based on Node.js, a server that allows the implementation of back end applications using the perfect JavaScript for applications that use real-time data intensifier. In order to provide developers with a simpler and more efficient development environment, the open-source Express.js framework, which complements Node.js, is a running module in Node.js. Requests sent by Angular are retrieved and analyzed by Node.js and processed by Express. With Express, it is easy to handle Angular request routing, giving us the architectural style based on REST API. Mainly, this processing is accomplished by communicating with the third layer, the NoSQL MongoDB database that is document oriented instead of the classical SQL specific tables. Using Express, the application has access to the database by using query requests with which we search, filter and return the desired information from MongoDB in a specific JavaScript format called JSON. Figure 12 shows how the MEAN Stack [3] works: the client makes a request via Angular to the Node.js. This request is taken over by Express.js and the request is made to the database. The result received from the database is processed and sent to the client as a reverse send response.

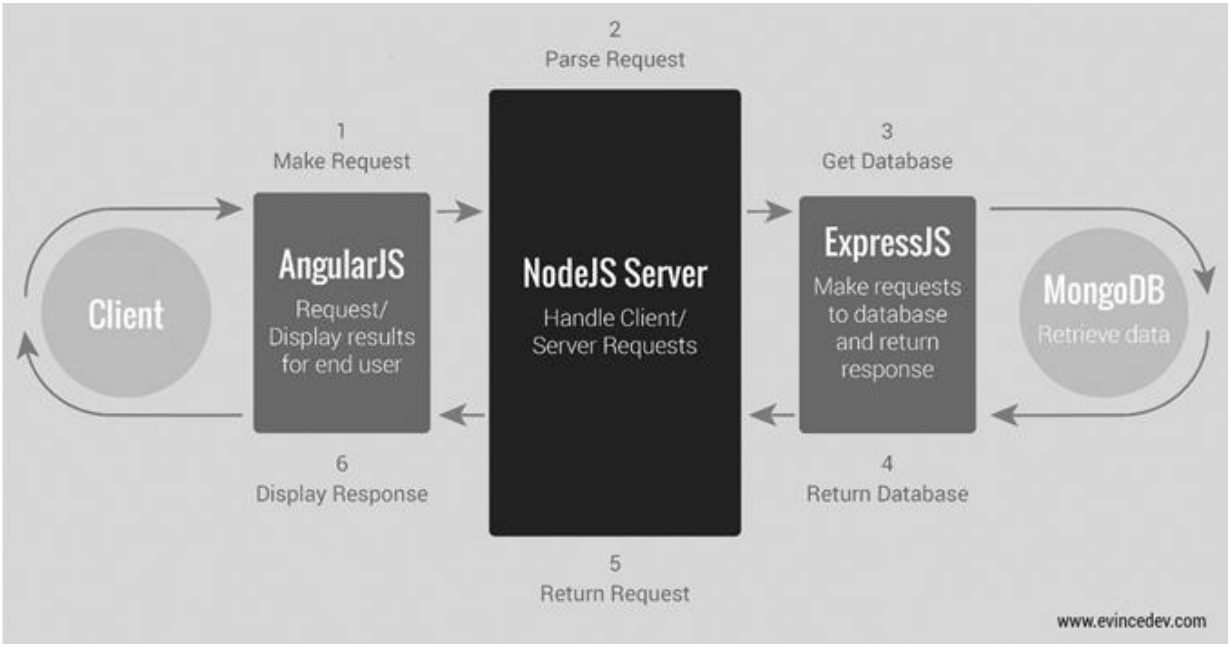


Fig. 12 – How MEAN Stack works

The main advantage that NoSQL offers is that we are not constrained to a fixed structure as in the classic SQL tables. This means that we can easily modify the documents of a collection very easy (Fig. 13) without wasting time to migrate the database from one version to another and without affecting the existing functionality.
The developed application folds well on NoSQL due to the fact that it is based on a flexible structure so the new changes to the application do not affect the database. Another reason is that the application does not use complex JOIN operations that affect the main advantage of NoSQL, allowing intensive work with data such as sending, adding requests, bids, and reviews. MongoDB has become extremely popular due to the speed, ease of operation and scalability it offers [4].

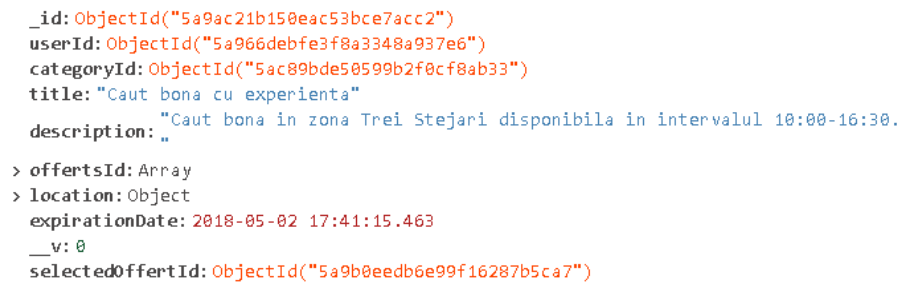


Fig 13 – Example of NoDQL document

Definition 1[5] *The NoSQL document represents a storage unit in the database being the equivalent of a record in SQL. It uses the JSON (Object Notation) format for data storage and it is the most widely used format that allows applications to exchange data between them.*

Definition 2[5] *NoSQL collections represent the equivalent of classic SQL tables. A collection has the ability to store documents that do not respect a particular structure; each document can have different key-value properties that can not be found in other documents in the same collection.*

A comparison between the relational data model and the document (Fig. 14) highlights the main difference between these models. In the relational model the organization of the tables is highly structured while in the document model the table is replaced by the collection of complex documents. This rigidly-defined format for data and record structures is replaced by arbitrary, nested data format.

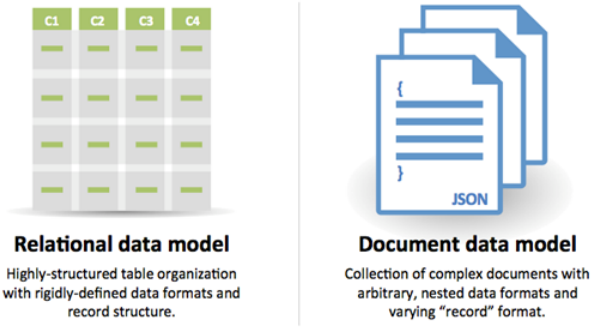


Fig. 14 – Comparation between SQL and NoSQL [6]

3 Integration framework. Program Code

In any web application, it is good to have a separation between the client and the server side, possibly even between the server and the database. This grants the application a high degree of independence, which is useful when we want to change one of these application layers. So, if you want to use another UI framework instead of Angular after a while, this should happen without the changes that affect the server side. Similarly to the database, if you want to move the localhost base to a dedicated server or if you want to switch between two databases on different servers, this must happen very simply without affecting the functionality of other layers.

At the beginning, the dependencies that the server uses for its operation must be added. Mainly, these are basic dependencies used by Angular to manipulate the client side and to be able to send the requests, the dependence for Express.js and other dependencies that have the role of parsing requests, encrypting information, and even communicating with the database (Fig. 15).

```
"dependencies": {
   "@angular/common": "^4.0.0",          "@angular/compiler": "^4.0.0",
   "@angular/compiler-cli": "^4.0.0",    "@angular/core": "^4.0.0",
   "@angular/forms": "^4.0.0",           "@angular/http": "^4.0.0",
   "@angular/router": "^4.0.0",
   // .....
   "body-parser": "~1.15.2",             "cookie-parser": "~1.4.3",
   "express": "~4.14.0",                 "jsonwebtoken": "^5.7.0",
   "hbs": "~3.1.0",                      "bcryptjs": "^2.4.3",
   "mongoose": "^4.4.12",          "mongoose-unique-validator": "^1.0.2"
   // .....
 }
```

Fig. 15 – Dependency packages for Node.js

The next step is to configure the Node.js server using the Express.js framework for communicating with the database, specifying the directory containing the view that the server sends as a response to the client to display it with the view engine used for the injection data in the view template (Fig 16).

```
var express = require('express');
var mongoose = require('mongoose');                    //dependinta pentru comunicarea cu MongoDB

var app = express();
app.set('views', path.join(__dirname, 'views');        //use views directory to render views
app.set('view engine', 'hbs');                         //set handlebars as view engine for templates

mongoose.connect('localhost:27017/database_name');     //set URI for database connection
```

Fig. 16 – Code sample for server configuration

Next, you need to configure the middleware part (Fig. 17) of the application that is used for:

- extracting information from your incoming requests
- specifying directories that are visible and accessible from outside the server
- receiving Cross-Origin Resource Sharing (CORS) requests coming from outside the server
- manipulating REST request paths to return the desired response

```
app.use(bodyParser.json());
app.use(express.static(path.join(__dirname, 'public')));

app.use(function (req, res, next) {
   res.setHeader('Access-Control-Allow-Origin', '*');
   res.setHeader('Access-Control-Allow-Headers', 'Origin, X-Requested-With, Content-Type, Accept');
   res.setHeader('Access-Control-Allow-Methods', 'POST, GET, PATCH, DELETE, OPTIONS');
   next();
});

app.use('/api/my-account', settingsRoutes);
app.use('/api/oferte', offersRoutes);
app.use('/api/review', reviewRoutes);
app.use('/api/search', searchRoutes);
app.use('/api/anunturi', adRoutes);
app.use('/api/categorii', categoryRoutes);
app.use('/api/auth', authRoutes);
app.use('/api/user', userRoutes);
app.use('/', appRoutes);
```

Fig. 17 – Code sample for request, CORS and routes setup

The final step is the configuration of the Angular Module (Fig. 18) specifying which modules and services use Angular, which the components are and finally which the basic component by which the application is started and used is.

```
@NgModule({
   declarations: [              //which are components used by application
      AppComponent,
      AuthenticationComponent,
      LogoutComponent,
      SignupComponent,
      SigninComponent,
      ForgotPasswordComponent,
//.............................
      HeaderComponent,
      FooterComponent
   ],
   imports: [                   //which are modules used by application for handling requests and forms
      BrowserModule,
      FormsModule,
      routing,
      ReactiveFormsModule,
      HttpModule
   ],
   providers: [                 //which are providers used for response to user requests
      AuthService,
      UserService,
      AdService
//.................
   ],
   bootstrap: [AppComponent]          //which is main component of application
})
export class AppModule { }
```

Fig. 18 – Sample code for Angular setup

4 Conclusions and further research directions

This article aims to present an application concept that uses the two-sided marketplace business model. It proposes a new approach to how service-customers and service providers can interact. This application system notifies the experts about area inquiries and provides information in the form of reviews for the clients. In this way the time spent by customers for finding a competent person is improved. Service providers are also helped to use more efficient their time, finding easier new clients and optimizing their scheduling.

On our opinion, the proposed application could be an important factor in developing local economies and lowering the unemployment rate. Through the possibilities offered by the application, even a person without a job, who learns certain technical skills and abilities, can earn an extra income by offering his services through the platform.

The application at the level that is achieved so far is functional and allows us to achieve the main objectives that we have created to design it. The application can be developed by adding extensions to improve its functionality. The main extensions we want to implement in the future are presented below.

One useful extension consists in improving the selection process of the experts by adding additional criteria to select service providers (e.g a rating situated in a specified interval, or presence of certifications).

Another possible extension of application functionality could be to deploy a service that allows the client to select time frames from a structured week to days in which the user can specify when he is available. In this way, the expert can make a realistic scheduling of his activity.

An important development of the application is the affiliation of training centers aiming to get certifications through the application. Thus, even if a person does not have certain technical abilities, he can acquire them from accredited partners through the application. This additional functionality, could lead to a radical change in the job market. The application would be of major importance not only for training people but also for supporting them to find new clients.

Acknowledgement: This work was supervised by Professor *Dana Simian*, from *Department of Mathematics and Informatics, Faculty of Sciences, "Lucian Blaga" University of Sibiu.*

References

[1] Reason Street, *Two-Sided Marketplace*, https://reasonstreet.co/business-model-two-sided-marketplace/ (accessed 01 May 2018)
[2] HelloMeets, *Why learn MEAN Stack?*, https://hackernoon.com/why-learn-mean-stack-9181506ec62 (accessed 01 May 2018)
[3] Evince Development, *MEAN Stack Architecture: AngularJS, NodeJS, ExpressJS and MongoDB*, https://evincedev.com/blog/mean-stack-architecture/. (accessed 01 May 2018)
[4] MongoDB, *MongoDB and MySQL Compared*, https://www.mongodb.com/compare/mongodb-mysql (accessed 01 May 2018)
[5] w3resource, *Databases, documents and collections in MongoDB, https://www.w3resource.com/mongodb/databases-documents-collections.php* (accessed 01 May 2018)
[6] Harshit Anand, *SQL vs NoSQL where to use that!*, https://medium.com/@wizzard_harshit/sql-vs-nosql-where-to-use-what-8b70d84ec7c8 (accessed 01 May 2018)

Gheorghe-Cătălin CRIȘAN
Lucian Blaga University of Sibiu
Faculty of Science
Str. Dr. I. Ratiu, No.5-7, Sibiu, 550012
ROMANIA
E-mail: crisan.gheorghecatalin@gmail.com

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

A generic Scratch interpreter for embedded systems

Alexander M. Frühwald

Abstract

This paper describes the design and development of a generic Scratch interpreter for use with embedded systems. As an example, the usage of the interpreter with controllers made by Fischertechnik is shown. This software can be used to easily extend existing software to allow any user to manipulate and control it using the visual programming language Scratch. The particularity of this piece of software is that it is easy to be integrated and optimized for the usage on embedded systems and it has a minimal footprint and a very low memory usage.

1 Introduction

The ability to code computer programs is an important part of literacy in today's society. Learning how to code computer programs is basically not simple, it requires a lot of time and experience. One has to deal with the syntax of a programming language, which might get quite complex depending on the language. It might get even more difficult when writing so called low level programs for embedded systems with sensors and actuators.

To deal with this problem, the Lifelong Kindergarden group at the MIT Media Lab developed the visual programming language Scratch [1]. Scratch consists of simple blocks which can be stacked together to create programs. Programmers do not have to bother with a complex syntax. The development environment (project editor) works with any modern browser of Windows, Linux, and MacOS systems, so one can start coding without prior installation of a development environment. Adobe Flash is the runtime environment. Scratch is useful to program interactive stories, games, and animations but it lacks the possibility to create proprietary commands to control components that do not belong to Scratch itself.

To overcome this drawback ScratchX [2] was invented. ScratchX is a platform that enables people to test experimental functionality built by developers for Scratch. This experimental functionality, called Experimental Extensions, makes it possible for Scratch to interface with external hardware, such as robots or programmable devices, and information outside the Scratch website through new blocks. Extensions are written in JavaScript [2].

Members of the Faculty of Computer Science at the University of Applied Sciences Wuerzburg-Schweinfurt created an extension for controllers made by Fischertechnik [4]. The extension for the TXT Controller is called FTScratchTXT and it is listed on the official ScratchX webpage. It extends ScratchX by blocks which allow one to read TXT inputs (sensors) and control outputs (motors).

Scratch and ScratchX have been designed to run in the online mode. It means that an active interconnection between a browser and external hardware is required to run programs, which in Scratch are called projects. Short after FTScratchTXT was released, the users' community came up with the demand for a download functionality to store programs persistently on the TXT Controller and run programs online. This fact led to the development of the software described in this paper.

The rest of the paper is organized as follows. Section 2 introduces the TXT Controller which is used in this paper. Section 3 targets the objectives of the software created for this project. The fourth section gives a deeper understanding of Scratch and the contents of project files. Section 5 introduces the implementation of the ScratchX-Interpreter and section 6 evaluates the results of various tests.

2 A brief introduction of the TXT Controller

The TXT Controller made by Fischertechnik was used to develop and test the software presented in this paper. The TXT Controller is the most feature-rich controller made by Fischertechnik (Fig. 1). It comes with various compatible construction kits [2]. It can be used in any Fischertechnik compatible project, since the controller's dimensions are only 90 mm x 90 mm x 25 mm. Eight inputs allow one the use of many sensors and eight outputs can be used to control eight devices or four motors instead.

Fig. 1: Fischertechnik TXT-Controller: Dual processor ARM Cortex A8 + Cortex M3, 8 universal inputs (digital/analog) I1 ... I8, 4 fast counting inputs, 4 motor outputs, BT 2.1 EDR+ 4.0, WLAN 802.11, Infrared receiver diode, USB 2.0 Client, colour touch display [5].

The controller runs a stripped down Linux variant on its ARM Cortex A8, so it is not difficult to write new software to run on it. Connecting the controller to a computer by WiFi, Bluetooth or USB results in an active network connection between a computer and the controller [5]. The controller may be accessed via SSH [10] then.

3 Objectives of the developed software

The software presented in this paper has the following objectives to achieve. First of all it has to be able to parse and execute ScratchX projects and associated extensions. It has to be platform independent, which means the software shall be compilable for the most common operating systems like, Windows and Linux. Since the solution should be as generic as possible, it should have a public interface to allow the development of new extensions. Also, any developer who wants to use features of the software should have the possibility to do so in the simplest way possible.

Because of these objectives, the final version of the software is a shared library with public interfaces. This library handles everything related to the execution of ScratchX projects so the developer does not need to bother it. To include it into a project, a developer just has to add the library, add some extensions and can instantly load and execute ScratchX projects. Fig. 2 shows a representation of the described model.

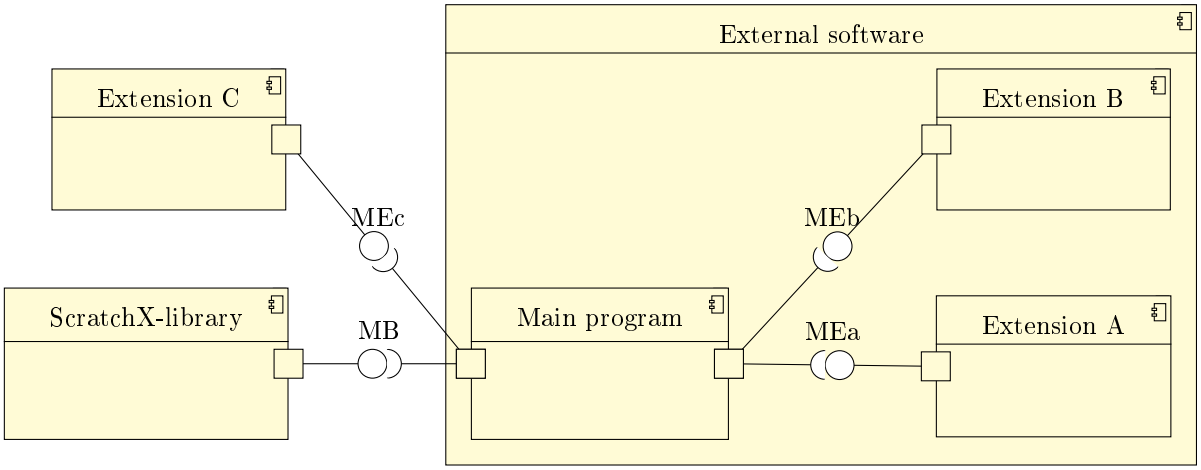


Fig. 2: Example sketch of software components when using multiple extensions

4 A more detailed view on ScratchX

Since the Scratch interpreter has to parse and execute ScratchX projects, it was necessary to deeply understand Scratch and its underlying structure. It is important to note that every citation of Scratch in this paper refers to Scratch Version 2.0, as Scratch 3.0 is still in the beta state and might be released officially only in late 2018.

Scratch is event-based, which means that every action depends on predeclared events. So called scripts are executed when specific criteria are met, e.g., when a specific button is pressed. This allows to create easily readable and abstract programs.

Any ScratchX project created in the browser environmant can be saved into a *.sbx project file to be shared with other people. The *.sbx file is a ZIP-compressed archive that contains all necessary files needed to execute the project. By default the file contains some images and one audio file but the most important one is *project.json*. This file contains every aspect of the project, including basic information about the creator, the language, variable definitions, and the definition of the script itself.

4.1 Blocktypes of Scratch

The first step is to understand the block structure of Scratch and to create a simplified model of it. Therefore it is reasonable to list up all types of blocks available and to compare the content of the JSON file in the *.sbx project archive with the corresponding Scratch program. A detailed list of blocks can be found at the official Scratch wiki [8]. In Scratch blocks are grouped into categories like Events, Control, Sensing, Operators.

Fig. 4 shows an example of a simple Scratch script as part of a *.sbx project file. This quite simple script adds the number ten to the variable *test* ten times and then evaluates the result by playing a sound on success. This example shows all six different types of blocks existing in the current version of Scratch.

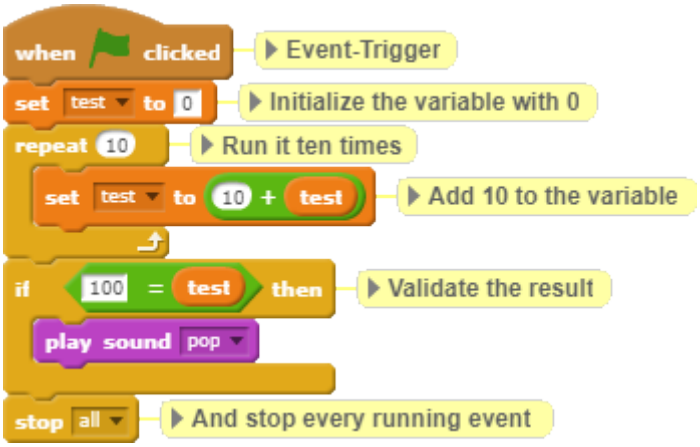


Fig. 3: A basic Scratch script to show the block types of Scratch

The six block-types are:

- Hat-Blocks, which start an event.
- Stack-Blocks, which can be stacked together to create a script.
- Boolean- and Reporter-blocks, which return values to be used in different scenarios.
- C-Block which give the possibility to create code branches.
- Cap-Blocks, which end the current script.

4.2 Blocktypes of Scratch

The first step is to understand the block structure of Scratch and to create a simplified model of it. Therefore it is reasonable list up all types of blocks available and to compare the JSON output in the *.sbx project file with the actual Scratch program. A detailed list of blocks can be found at [8]. In Scratch blocks are grouped into categories like Events, Control, Sensing, Operators.

Fig. 4 shows an example for a simple Scratch script as part of a *.sbx project file. The quite simple script adds the number of ten ten times to the variable *test* and then evuluates the result by playing a sound on success. This example contains all six different types of blocks existing in the current version of Scratch.

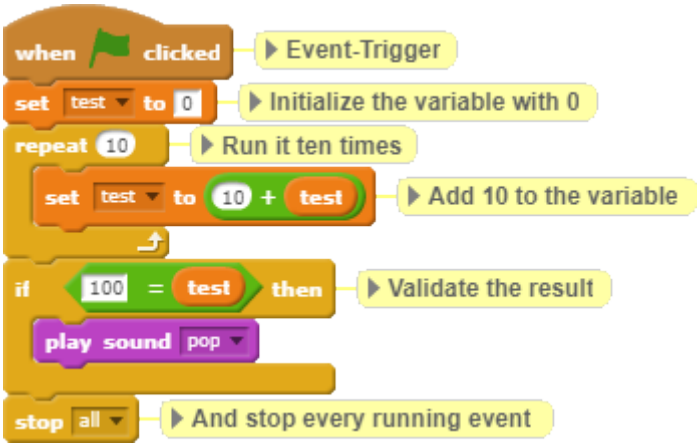


Fig. 4: A basic Scratch script to show the block types of Scratch

Procedure definitions

Scratch allows users to create own procedures. They can be used to reduce the overall size of the script by reducing the repetition of a code. A procedure de nition creates a new event that can be called like any other block. Arguments can be used, like other Reporter and Boolean blocks, but only in the scope of the current procedure. Fig. 5 shows such a procedure definition. This procedure accepts three arguments, a number, a string, and a boolean. Values for them are given to the respective method call on the left side.

Fig. 5: Example of a procedure definition in Scratch

Blocks added by extensions

An extension can add new blocks to the existing catalogue. They can be found in section "More Blocks". Every block added by extensions has the same pre-defined color and it is not possible to add extension blocks into one of the existing block categories.

The extension FTScratchTXT currently adds 21 new blocks: three Hat blocks, two Reporter blocks, one Boolean block and 15 Stack blocks. Each of those blocks controls a specific function of the controller, for example it can set a motor output or read a sensor value. Figure 6 shows some of the blocks from the extension.

Fig. 6: A script using blocks from the FTScratchTXT extension

When the switch on input I1 opens, sound no. 1 is played, a lamp on output O3 is set to 50 % (= 4) and motors on M1 and M2 move forward with maximum speed (=8). All blocks are intentionally self-speaking and documented on the official website [3].

4.3 Creating an abstract model of the Scratch script structure

Each block has its representation in the JSON file. Fig. 7 shows the JSON representation the program in Fig. 4, which is only a part of the JSON file. The file may hold multiple of the shown arrays. The example file shows all important features. The investigation gave the following results:

The first two values of the array represent the position of the script in the editor. As this position is not needed to parse and execute the script, these values are ignored. Only the third value is of interest: This array contains elements related to the current script which must be parsed. From now on, when *the array* is mentioned, it references to this argument.

```
[10,10, [
  ["whenGreenFlag"],
  ["setVar:to:", "test", "0"],
  ["doRepeat",
    10,
    [["setVar:to:",
      "test",
      ["+",
        10,
        ["readVariable", "test"]
      ]
    ]]
  ],
  ["doIf",
    ["=",
      "100",
      ["readVariable", "test"]
    ],
    [["playSound:", "pop"]]
    ],
  ["stopScripts", "all"]
]]
```

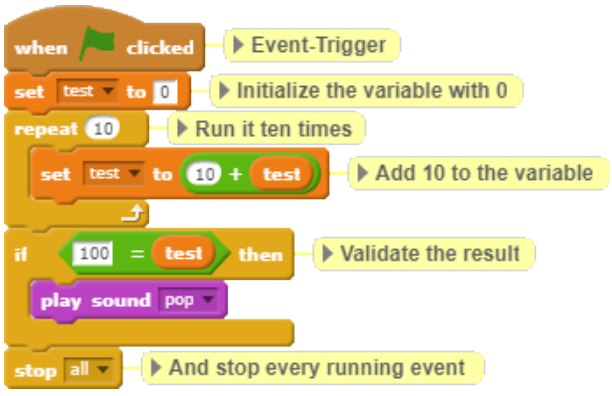


Fig. 7: An example Scratch script (right) and its JSON representation (left)

In general, this array is an array of commands, where each command is described in the reverse polish notation, which means that the name of the command is at the first place, followed by each of its arguments. The first element of the array describes the event on which the script should be executed. If the element at this position is no valid event, the whole script can be ignored and will not be parsed, because the script will never be executed.

Line 3 of the example is the representation of the first stack block in Fig. 4. As already noted, the first element of this sub-array is the name of the method, followed by its arguments. Comparing it to the Cap block at line 21 one can see that there is no structural difference at all, both blocks return no value and thereby both of them may be parsed with the same method.

Lines 4 to 13 show the "repeat block". This C block also receives two arguments: the first one tells how often the code is to be repeated and the second one, beginning from line 6, is the script that is to be repeated. The calculation starting from line 8 follows the same pattern.

To sum up the JSON structure, it can be described as follows. An element that is not an array, is just a native data type - a number, a text or a boolean. An element that forms a two-dimensional array describes a subscript, e.g., in Fig. 7-left the array at the lines 1, 6 and 19. Additionally, any element that is just a normal array describes a command, where the first element is the name of the command and the other elements form the arguments handed over to the command. The arguments themselves can be any of the three types.

Procedure definitions

Procedure definitions are just a special kind of events. The JSON representation of these acts therefore like an event. Listing 2 shows an example of a procedure definition. It clearly shows that the first element describes the whole procedure: the first value is always *procDef* which simplifies the detection in an algorithm. The second value contains the name of the procedure and describes the types of the procedure. The third value describes names of the arguments and the fourth one contains the respective default values. The last argument is always false for ScratchX - it is only used for the cloud feature of Scratch that is not available in ScratchX and therefore is not needed for this project.

Listing 1: Example of a procedure definition in JSON format

```
1  [264.2, 11.2, [
2    [
3      "procDef",
4      "Sample Procedure %n %s %b",
5      ["number1", "string1", "boolean1"],
6      [1, "", false],
7      false
8    ],
9    ["wait:elapsed:from:", 1],
10   ["wait:elapsed:from:", 3]
11 ]]
```

5 Implementation of the ScratchX-interpreter

After presenting the structure of Scratch project files, in this Section the implementation of the interpreter is discussed. The main assumption of the implementation was the ease of expandability, implying the definition of available commands had to be as simple as possible - the whole software has been built around this concept. Before the interpreter parses any project,

it registers available commands at a central place so that they can be accessed easily when the interpreter parses and executes ScratchX projects. The interpreter also registers any available event, as those are needed to parse any project, which is the next step.

5.1 Parsing the ScratchX project

With the knowledge gained, the development of an algorithm to parse the structure of Scratch project files was the next challenge. The result is a deeply recursive algorithm, as Scratch scripts can be nested nearly endlessly and method calls also can accept other methods as their arguments.

The algorithm first checks for every element in the *scripts*-array of the JSON file whether or not the first element targets a valid event or is a procedure definition. If this is not the case, the whole script is considered invalid and thereby is ignored (not executed). Otherwise, if there is a valid event, the algorithm creates a new event instance and fills it with the existing information. Afterwards, it loops through the existing method calls, attaching each one of those to the event so they can be executed in order after the parsing process is terminated. If there is a valid procedure definition, the algorithm will also create a valid representation of it and register it to the class that comprises and manages all available commands.

Parsing the method arguments is the point where the parsing process gets its depth. An argument can be either a native data type, another method call or another script (just without an event header on its top). Depending on the type, the algorithm calls the respective method recursively and attaches the result as an argument to the method call. The algorithm terminates when only native data types are left and it can not get more nested.

Internal structure of parsed ScratchX projects

While it is not difficult to develop such an algorithm, the decision how to store the parsed projects is more difficult. The final model describes blocks, scripts and events as trees. Blocks can receive arguments which are saved as leaves in the tree, whereas the block itself is the root of this tree. Let us take the construct in Fig. 8 as an example.

Fig. 8: Simple Scratch structure to show the tree concept

Its structure is not very complex, but a tree that resembles it can become quite deep. To convert it into a tree, it is first necessary to have a look at its JSON representation because the interpreter will only know this representation in Listing 2.

Listing 2: Example of a simple method chain in JSON structure

```
[
  "wait:elapsed:from:",
    [
        "+",
        [
            "readVariable",
            "someVar"
        ],
        10
    ]
]
```

The targeted tree structure is already partly visible in this form. Every argument a method call receives is a new leaf (new child) of it. If we follow this simple concept, we will get a tree like the one shown in Fig. 9. When the block is executed, the tree will be resolved from bottom to top until the arguments are just native types and thus, usable by the block itself.

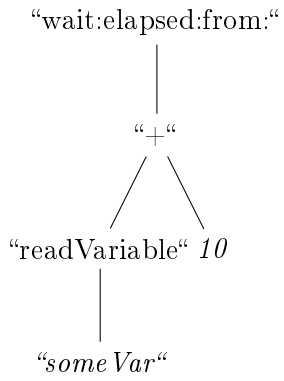


Fig. 9: A sample tree for the given method chain

Tree form of scripts

We can also follow this simple principle with scripts and events. An event always contains one script, a script contains one or more blocks, and a block can receive multiple arguments. These arguments can be native types, other blocks or whole scripts. A complete project can be represented as a tree when the root is the project itself, followed by every event that is available. The event contains its arguments and otherwise just a script and the script contains all blocks, in the order they shall be executed. Fig. 10 shows such a tree, reduced to the minimum of information needed.

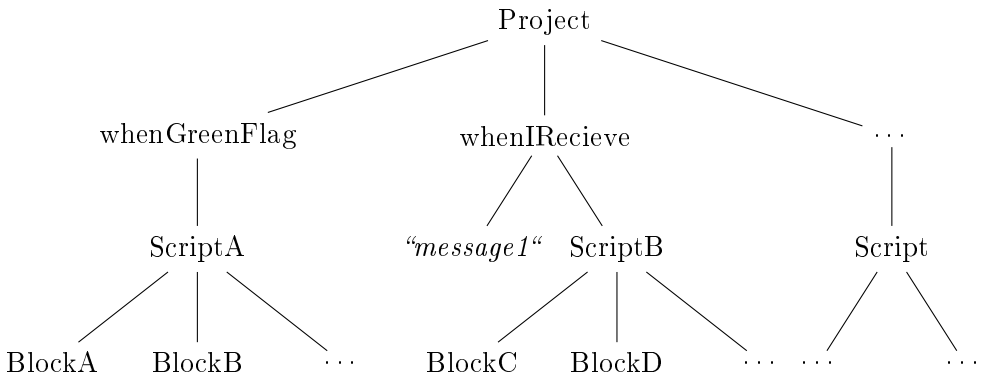


Fig. 10: Example of a project tree

5.2 Executing the parsed Scratch project

When a script is completely parsed, it can be executed. As Scratch is event-based, the software must check for every event to see whether it should be executed or not. These events can be very time critical, which means that missing an event should never happen. Because of that the

library checks 100 times a second whether a event has occurred or not. The frequency is high enough to catch all events while its impact on the CPU remains minimal. Each time an event is caught a new thread is started because the events must be able to run simultaneously.

A script is executed by iterating over all commands it contains and running each of those commands one after another. That means that the commands are blocked and will run in order. Commands unknown to the interpreter are simply ignored and skipped.

5.3 Registering extensions to the interpreter

The interpreter is not able to execute unknown blocks. This also applies to extension commands: they are not part of the native ScratchX environment and therefore there is no way the interpreter can know how to execute them.

In the browser environment, an extension is just a JavaScript file which contains the whole definition of the extension. This file can not be used in the local interpreter as JavaScript is unknown to the interpreter and additionally, the JavaScript implementation of an extension may differ from the intended behaviour when executed on an external hardware. Thereby, the developer must create a compatible version of her/his extension in a way the interpreter may use it. This version has to implement specific public interfaces that are determined by the interpreter. All available extensions must be registered to the interpreter instance before it starts to parse a project.

6 Practical application of the software using FTScratchTXT

The described library is used to develop a software which is able to run ScratchX projects on the Fischertechnik TXT Controller. It allows to read sensors and control the motor just as the existing browser-based solution does. The big advantage, however, is the fact that the network connection is no longer needed. The program runs directly on the controller and thus, it can react much faster to changed input values. The implemented extension has been tested thoroughly. Some tests were used to compare the existing (online) version with the new native (offline) version.

First comparison: Timing and detection of events

With Scratch being an event-driven language it is very important that the software can handle events as expected. For this purpose a test script that uses a special feature of the Fischertechnik motors was created. The TXT-controller increments an internal counter each time the motor moved a specific distance. Using these counters the script can measure the time it takes to detect a specific event: For this a motor is started at its maximum speed and an event waits for the related counter to reach the value of 100. After that the motor is stopped and the new counter value is written into a variable so it can be used for further calculations.

The diagram in Fig. 11 shows the arithmetic mean of the counter values the motor moved over the desired value of 100. For each of the speed values (25%, 50%, 75% and 100%) the motor has been started 100 times. It is clear from the figure that the local interpreter can detect events more effective and faster than the browser based solution.

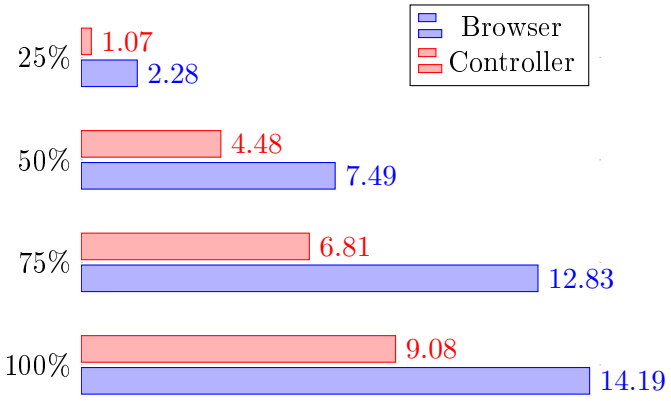


Fig. 11: Comparison of the arithmetic mean of the test runs. The black number on the left determines the speed of the motor. All other values are counter values that the motor moved over the desired value.

Second comparison: Comparing sensor values

The second comparison also checks for event compatibility. In this test a vehicle from the Fischertechnik *ROBOTICS TXT Discovery Set* [6] has been put in front of a wall at a distance of one meter. The used script let the "Discoverer Robot" go at full speed into the direction of the wall and it should stop as soon as an obstacle is detected in less than 20 cm distance using an ultrasonic sensor. This was repeated ten times each for the browser version and the local interpreter.

Table 1 shows the results of the test runs. The closer the distance comes to 20, the better. The precision is satisfying for both versions, but as shown, the local interpreter is much faster in checking for the distance and stopping the motor.

Table 1: Distances to the wall in the sensor test
All values in centimeter.

Test Nr.	1	2	3	4	5	6	7	8	9	10	Average
Browser	16	15	16	15	16	15	17	16	15	14	15.50
Local interpreter	17	18	17	17	20	19	17	18	18	18	18.00

Summary of the tests

The test cases show clearly that there is a huge performance gain when using the local interpreter. The main cause is that the browser environment needs an active network connection which slows down the data transfer rate. Additionally the local interpreter is able to use the full processing power of the controller, using optimized C++ code whilst the browser environment has to use limited browser resources. This shows that the (offline) ScratchX interpreter works as expected and is very efficient.

7 Conclusion and future development

The developed software is able to run ScratchX projects on most modern operating systems. The software can easily be integrated into any related project because it offers public interfaces that allow the registration of extensions to give ScratchX the ability to control the behaviour of the software or the platform it is running on. The software gives developers the possibility to allow external users, for example users of controllers made by Fischertechnik or other vendors, to modify the behaviour of the software with an easy to learn, visual programming language, making script languages like Lua or Python obsolete for event driven software.

But the development does not end here. The software can be still extended with some useful debugging features and interfaces. It could have a public interface that allows to read the currently running events and blocks. This will vastly improve the debugging process and thus, finding errors in ScratchX projects will be much easier. It could be improved even more by creating a visual graph that resembles the called event tree. Another nice feature could be the live display of variables and other runtime variables, as they are currently hidden from the user and could be very useful in debugging purposes. But the core of the future development will be the implementation of the described library in other software projects and therefore, proving its capabilities and usefulness.

Acknowledgement: This work was supervised by Professor Dr. *Arndt Balzer* and M. Sc. *Frank Ebner* from the *University of Applied Sciences Wuerzburg-Schweinfurt.*

8 References

References

[1] Lifelong Kindergarden research group, Scratch `https://scratch.mit.edu/`
[2] Lifelong Kindergarden research group, ScratchX FAQ `https://scratchx.org/`
[3] FH|W-S, FTScratchTXT `https://ftscratch.github.io/ROBO-TXT/www/en/index.html`
[4] Fischertechnik, Fischertechnik `https://www.fischertechnik.de/en`
[5] Fischertechnik, ROBOTICS TXT Controller `https://www.fischertechnik.de/de-de/produkte/spielen/robotics/522429-robotics-txt-controller`
[6] Fischertechnik, ROBOTICS Discovery Set `https://www.fischertechnik.de/de-de/service/elearning/spielen/txt-discovery-set`
[7] JSON, Introducing JSON `https://www.json.org/`
[8] Scratch Wiki, Available blocks in Scratch `https://en.scratch-wiki.info/wiki/Blocks`
[9] Scratch Wiki, Block Selectors `https://wiki.scratch.mit.edu/wiki/Scratch\_File\_Format\_(2.0)/\-Block\_Selectors`
[10] OpenSSH, OpenSSH `https://www.openssh.com`

Alexander M. FRÜHWALD
University of Applied Sciences Wuerzburg-Schweinfurt
Faculty of Computer Science
Sanderheinrichsleitenweg 20
97074 Würzburg
GERMANY
E-mail: *alexander.fruehwald@student.fhws.de*

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Bypassing Antiviruses in 2018

Svetoslav Hadziivanov

Abstract

Many popular Antivirus (AV) solutions claim to be effective against unknown malware which is wrong. This paper presents various techniques used by malware developers to evade detections as like as a protection from it. It comes that some AV solutions have better methods to detect x86 malware than others but when it comes to x64 malware, they lack code implementation for it.

1 Introduction

The AV solutions can hardly compare each other. Generally, there are 3 stages in the detection process: Scantime detection (Static detection), Code Emulation (before execution) and Runtime detection (during execution). The first stage is used by all popular AV vendors for a long time. It's based on signatures from known malware files. The AVs are analyzing every file's icon, assembly information, Portable Executable [1] (PE) headers, sections, etc. If there is anything wrong with any of these, the file entropy coefficient increases and when it reaches a certain value, the file is being detected. Any executable can be inspected by using a PE Viewer, for example – CFF Explorer. It is an open source which is completely free. In fig. 1 and fig. 2, a part of the analyzed structures is presented for every newly scanned file on http://virustotal.com.

Portable Executable Info

Header

Target Machine	Intel 386 or later processors and compatible processors
Compilation Timestamp	2013-07-29 13:53:11
Entry Point	63550
Contained Sections	4

Sections

Name	Virtual Address	Virtual Size	Raw Size	Entropy	MD5
.text	8192	55364	55808	5.7	a11fa451ebecab3a2fd166a34cec44e5
.sdata	65536	189	512	2.63	e19829aa0459a4f8fc65191000ed9ea7
.rsrc	73728	69248	69632	3.48	f14f084a81da6ad777d6d936dc0c5034
.reloc	147456	12	512	0.08	391becfb5cb3c92ca849dca0e02d025c

Imports

mscoree.dll

Fig. 1 http://virustotal.com

Contained Resources By Type

RT VERSION	1
RT MANIFEST	1
RT ICON	1
RT GROUP ICON	1

Contained Resources By Language

NEUTRAL	4

Contained Resources

SHA-256	File Type	Type	Language
55e86c2fd9084587181adf9ea05270a4242ee82042ca4509fc590c5dad2de382	data	RT_ICON	NEUTRAL
12598188b44d76a8828aa7a8211c4c1bfa8093f617928f5c8f3da9cd81a42d64	ASCII text	RT_GROUP_ICON	NEUTRAL
031626eb3ccec9981626b96f4fb74d21aad652f46af021aae86fde0a79ba1d02	data	RT_VERSION	NEUTRAL
c0085eb467d2fc9c9f395047e057183b3cd1503a4087d0db565161c13527a76f	data	RT_MANIFEST	NEUTRAL

Fig. 2 http://virustotal.com

The first phase is easily passed by obfuscating/packing the file (making the file less readable). Although, there are non-malicious files that often get detected by a mistake. Through the second phase, the AV is running executables in an emulated environment which is monitoring for suspicious behavior. At the last phase, some AVs were found to implement dynamic techniques to identify suspicious behavior while the malware is executing on the system.

2 Portable Executable Overview

Windows PE Loader requires a specific format so the executables can run. Since the AVs also know that obfuscators/packers are often used to evade static detections, they also attempt to detect anomalies in PE's file format. The executables contain two high-level parts: header and sections. The header contains metadata about the executable and it describes the sections. The sections are collections of bytes put in a certain order by the compiler.

2.1 Executable Header

The executable header consists of a few structures as shown in fig. 3. Each of them will be discussed in turn in the sections below.

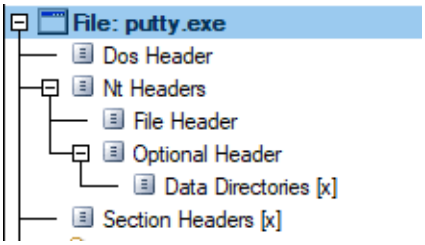


Fig. 3 Executable Header's structure

2.1.1 DOS Header

The DOS header is remaining from the old DOS MZ executable format. The first two bytes of the DOS header correspond to e_magic field that must be equal to 0x54AD – „MZ". The default PE DOS stub should contain „This program cannot be run in DOS mode" text which is ignored by

the modern Windows Operating Systems (OS) and they only verify the „MZ" and the offset to the PE header. If the file doesn't contain that, the file will be flagged as suspicious. This is shown in fig. 4.

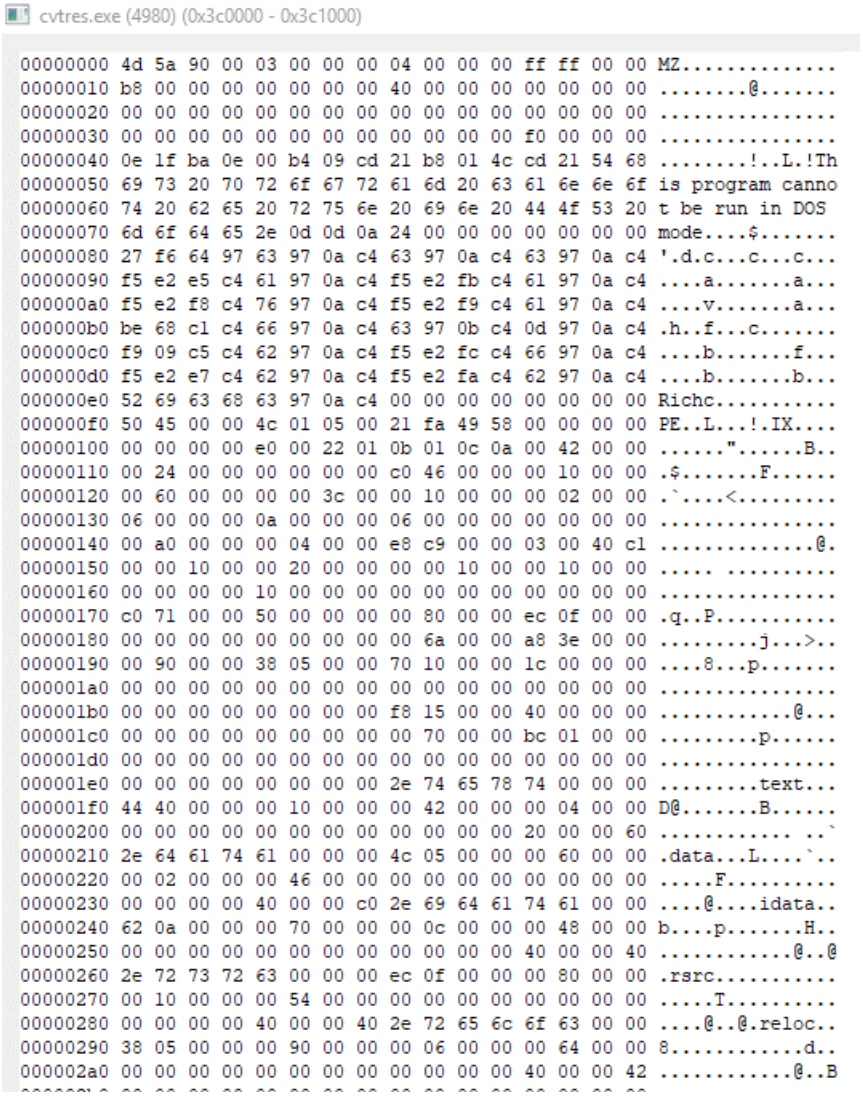


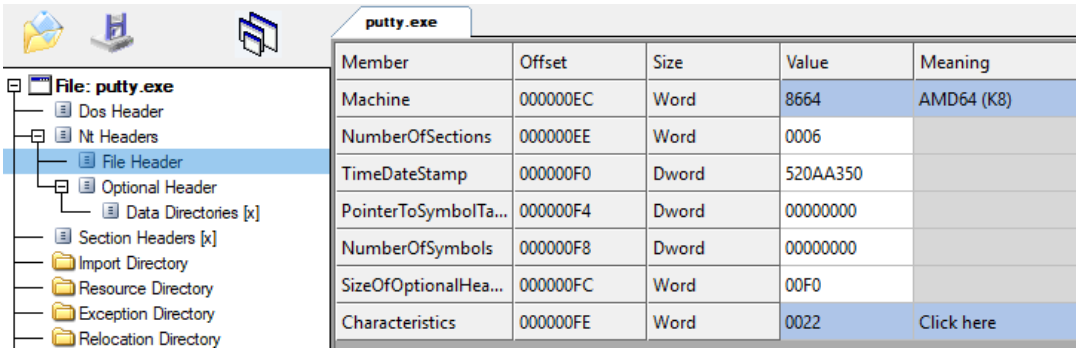
Fig. 4 PE in memory for every executable file (.exe, .dll)

2.1.2 NT Header (often referred as PE header)

It starts with a static signature value equal to 0x4550 – „PE". It is also verified by Windows PE Loader before loading the executable in memory. It consists of two subsections: File Header and Optional Header.

2.1.2.1 File Header

The only fields of that structure that were found to be checked by the AV vendors are TimeDateStamp, PointerToSymbolTable and NumberOfSymbols. If the TimeDateStamp field has unrealistic value or the PointerToSymbolTable and the NumberOfSymbols fields haven't zero values, the file entropy coefficient increases. It is shown in fig. 5 for another analyzed executable:



Member	Offset	Size	Value	Meaning
Machine	000000EC	Word	8664	AMD64 (K8)
NumberOfSections	000000EE	Word	0006	
TimeDateStamp	000000F0	Dword	520AA350	
PointerToSymbolTa...	000000F4	Dword	00000000	
NumberOfSymbols	000000F8	Dword	00000000	
SizeOfOptionalHea...	000000FC	Word	00F0	
Characteristics	000000FE	Word	0022	Click here

Fig. 5 File Header fields

2.1.2.2 Optional Header

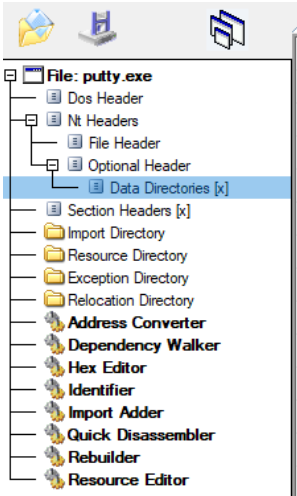
The Optional Header contains a lot of information from the Section Headers (fig. 6) for the same analyzed file (*putty.exe*).

Member	Offset	Size	Value	Meaning
Magic	00000100	Word	020B	PE64
MajorLinkerVersion	00000102	Byte	0A	
MinorLinkerVersion	00000103	Byte	00	
SizeOfCode	00000104	Dword	00076400	
SizeOfInitializedData	00000108	Dword	0002F600	
SizeOfUninitializedData	0000010C	Dword	00000000	
AddressOfEntryPoint	00000110	Dword	000691B8	.text
BaseOfCode	00000114	Dword	00001000	
ImageBase	00000118	Qword	0000000140000000	
SectionAlignment	00000120	Dword	00001000	
FileAlignment	00000124	Dword	00000200	
MajorOperatingSystemVers...	00000128	Word	0005	
MinorOperatingSystemVers...	0000012A	Word	0002	
MajorImageVersion	0000012C	Word	0000	
MinorImageVersion	0000012E	Word	0000	
MajorSubsystemVersion	00000130	Word	0005	
MinorSubsystemVersion	00000132	Word	0002	
Win32VersionValue	00000134	Dword	00000000	
SizeOfImage	00000138	Dword	000AE000	
SizeOfHeaders	0000013C	Dword	00000400	
CheckSum	00000140	Dword	000B09F8	
Subsystem	00000144	Word	0002	Windows GUI
DllCharacteristics	00000146	Word	8140	Click here
SizeOfStackReserve	00000148	Qword	0000000000100000	
SizeOfStackCommit	00000150	Qword	0000000000001000	

Fig. 6 Optional Header fields

The SizeOfCode, SizeOfInitializedDate, SizeOfUnitiliaizedData, BaseOfCode and BaseOfData fields are directly related to information found in the Section Headers at the very end

of the executable's header. If there is any inconsistency between them, several AV solutions raise detections. The most important fields in this section are SizeOfHeaders, SizeOfImage and AddressOfEntryPoint. They are used when we want to inject an executable in memory. The AddressOfEntryPoint contains the offset to the entry point where the PE loader will start executing from. It is often modified in order to hijack execution which also flags a detection.



putty.exe

Member	Offset	Size	Value	Section
Export Directory RVA	00000170	Dword	00000000	
Export Directory Size	00000174	Dword	00000000	
Import Directory RVA	00000178	Dword	00098E9C	.rdata
Import Directory Size	0000017C	Dword	000000F0	
Resource Directory RVA	00000180	Dword	000A9000	.rsrc
Resource Directory Size	00000184	Dword	00003904	
Exception Directory RVA	00000188	Dword	000A3000	.pdata
Exception Directory Size	0000018C	Dword	00005850	
Security Directory RVA	00000190	Dword	00000000	
Security Directory Size	00000194	Dword	00000000	
Relocation Directory RVA	00000198	Dword	000AD000	.reloc
Relocation Directory Size	0000019C	Dword	00000890	
Debug Directory RVA	000001A0	Dword	00000000	
Debug Directory Size	000001A4	Dword	00000000	
Architecture Directory RVA	000001A8	Dword	00000000	
Architecture Directory Size	000001AC	Dword	00000000	
Reserved	000001B0	Dword	00000000	
Reserved	000001B4	Dword	00000000	
TLS Directory RVA	000001B8	Dword	00000000	
TLS Directory Size	000001BC	Dword	00000000	
Configuration Directory RVA	000001C0	Dword	00000000	
Configuration Directory Size	000001C4	Dword	00000000	
Bound Import Directory RVA	000001C8	Dword	00000000	
Bound Import Directory Size	000001CC	Dword	00000000	
Import Address Table Directory ...	000001D0	Dword	00078000	.rdata

Fig. 7 Data Directories fields

2.1.2.3 Data Directories

The data directories are a subsection of Optional Header. This is a table of offsets to the specific directories in an executable. Import directory and relocations directory must be resolved in order to run an executable in memory. The IAT (Import Address Table) references a structure that lists the functions of external DLLs on which the executable relies (for example Kernel32.dll, NTDLL.DLL, etc.). The executable will have relocations directory only if ASLR (Address space layout randomization) is enabled while compiling the file. It contains all global variable addresses and whenever the executable is re-opened, it puts those variables on random places (offsets) in memory. The most Delphi applications require a TLS (Thread-Local Storage [2]) directory which should be resolved as well. The TLS callbacks are executed before the entry point (Main function). There are static TLS slots that should be resolved for each new thread. Data Directories fields are shown in fig. 7.

2.1.3 Section Headers

Fig. 8 shows the Section Headers for the same analyzed file (*putty.exe*). It contains 6 sections: „.text" section contains the program's code; „.rdata" contains specific metadata (in our case: Import Directory, Import Address Table Directory); „.data" – static data from the code (e.g. hard-coded static strings); „.pdata" – Exception Directory; „.rsrc" contains additional resources like version information and bitmap icons; „.reloc" has the Relocation Directory which means that ASLR for the executable is enabled. Characteristics field is also important because it contains the protection flags of each section (read, write and execute), as well as information about the content of the section which is a duplicate from Optional Header. If all sections are RWX (read, write and execute), the AVs are raising suspicion.

Name	Virtual Size	Virtual Address	Raw Size	Raw Address	Reloc Address	Linenumbers	Relocations N...	Linenumbers ...	Characteristics
Byte[8]	Dword	Dword	Dword	Dword	Dword	Dword	Word	Word	Dword
.text	00076281	00001000	00076400	00000400	00000000	00000000	0000	0000	60000020
.rdata	00022F22	00078000	00023000	00076800	00000000	00000000	0000	0000	40000040
.data	000077C4	0009B000	00002200	00099800	00000000	00000000	0000	0000	C0000040
.pdata	00005850	000A3000	00005A00	0009BA00	00000000	00000000	0000	0000	40000040
.rsrc	00003904	000A9000	00003A00	000A1400	00000000	00000000	0000	0000	40000040
.reloc	00000F70	000AD000	00001000	000A4E00	00000000	00000000	0000	0000	42000040

Fig. 8 Section Headers Overview

3 Stub Design

The most malicious files are composed of a loader and a stub. The loader has the stub bytes encrypted and saved in Resources or for example in Section Headers, byte array, string variable, etc. It decrypts those bytes at runtime (during execution) and it loads them in memory. The stub's construction is exactly like loader's but instead of the stub bytes it has payload's bytes (the real malware) stored in the Resources.

For the purpose of this research, the author created an example crypter. The most of the malware files are coded in C/ASM so they won't have a dependency and they can run on a fresh installed Windows OS from XP to 10. These examples were made in C# which requires .NET Framework to be installed on targetting system. By default, Windows 8, 8.1 and 10 has .NET Framework 4.0, but Windows XP, Vista and 7 have only .NET Framework 2.0. The examples are using dnSpy for source code view. In fig. 9, 10 and 11, an example stub and loader are presented.

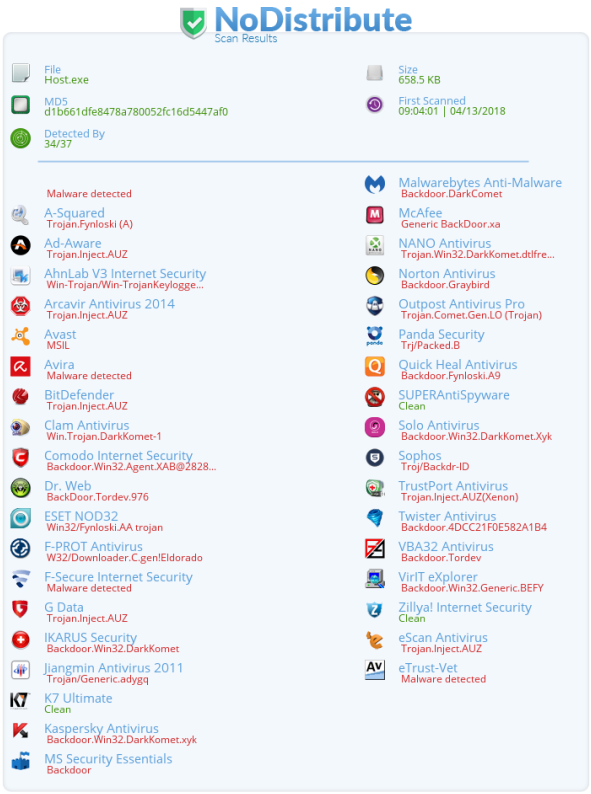


Fig. 9 Non-obfuscated/packed malware

Fig. 10 Obfuscated malware

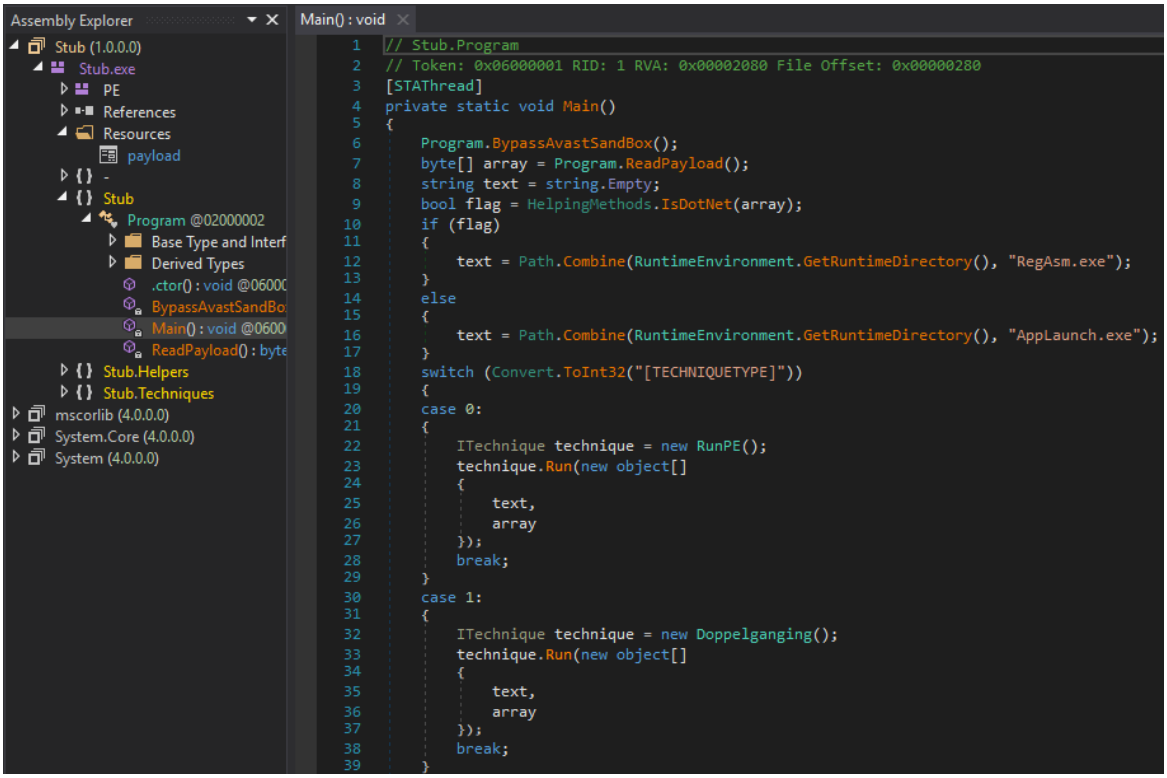


Fig. 11 Stub content (unobfuscated)

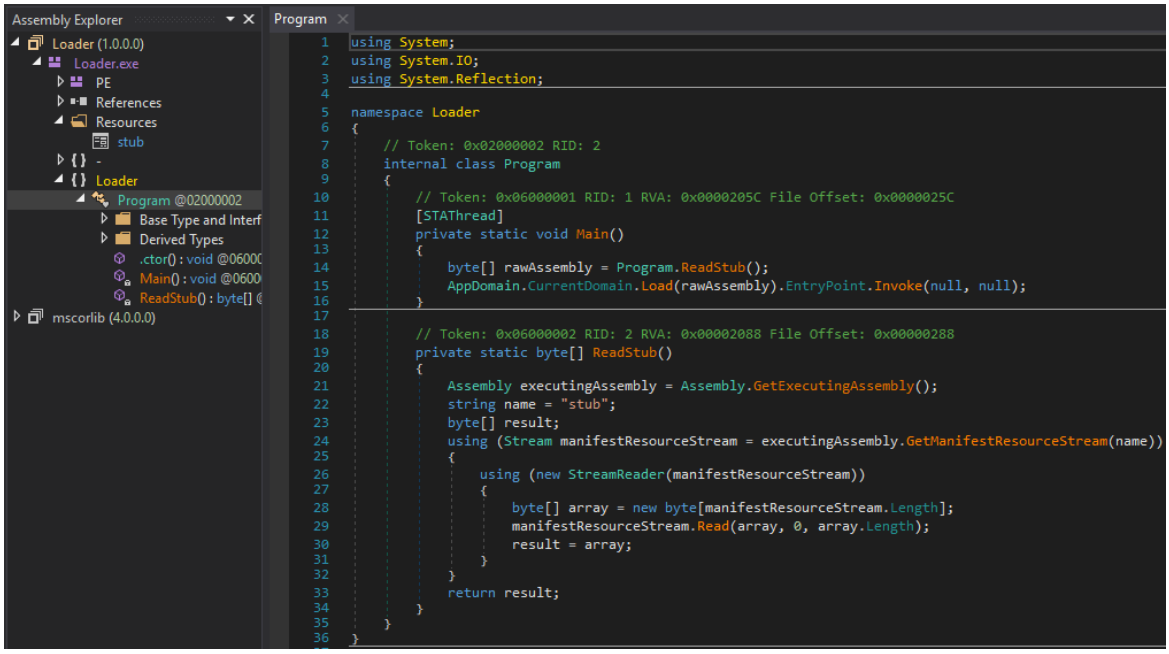


Fig. 12 Loader content (unobfuscated)

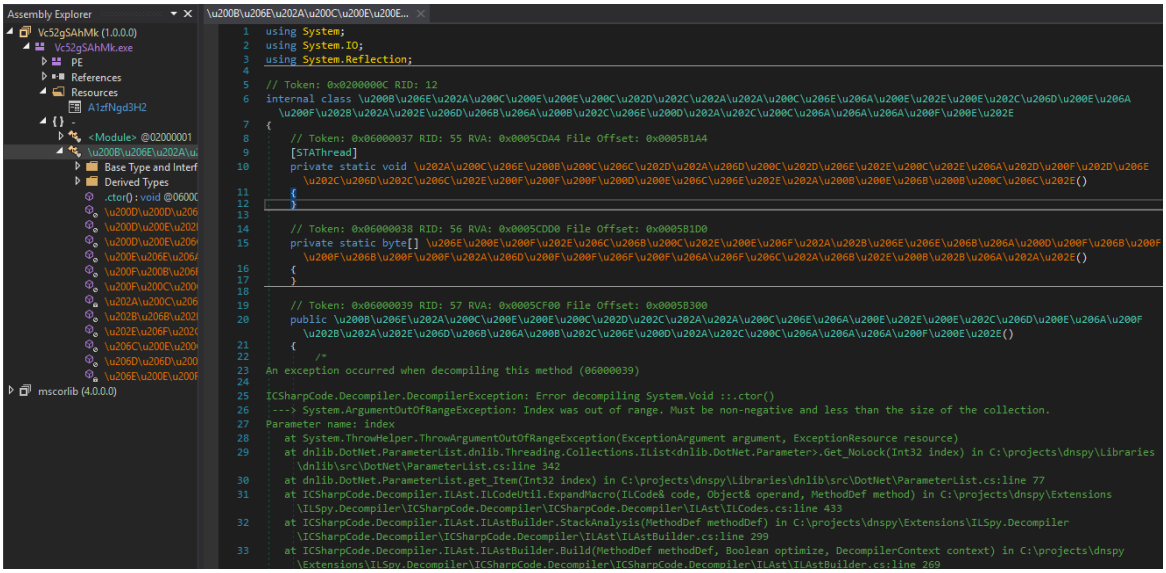


Fig. 13 Loader content (obfuscated)

To test whether the obfuscated file passes phase 1 (scantime) or not, the file can be scanned on http://nodistribute.com, http://viruscheckmate.com or by using installed on PC AVs. Note that http://nodistribute.com, http://virustotal.com, https://www.hybrid-analysis.com and a few others are distributing files to the AV labs for detailed analysis. So, it's not the best idea to scan there while doing a research like this.

Fig. 12 and 13 are showing the differences before obfuscating and after obfuscasting a malicious file. Fig. 13 shows that there are 0/37 detections which means that the file is completely FUD (Fully Undetectable) at scantime. The analyzed file is 32-bit which matters. The most AVs doesn't have implemented same engines for 64-bit files as like as they do have for 32-bit files. The reason is the most of the malicious files are made to target 32-bit systems. Anyway, scantime is the beginning of bypassing. For phase 2 (Code Emulation) and phase 3 (Dynamic techniques to detect malware at runtime), there should be some code to prevent each AV strong feature so we will be able to bypass them.

The following AVs were part of the research:

- „Windows Defender" (Windows 10)
- „AVAST/AVG" (They use same engine)
- „Avira"
- „ESET NOD32 Antivirus"
- „BitDefender Antivirus"
- „Kaspersky Antivirus"
- „Norton Security"
- „Malwarebytes Anti-Malware"
- „360 Total Security"

Not going to discuss AVs like McAfee because they failed on simple tests.

3.1 Bypassing Emulation (phase 2)

A few AVs were found to implement emulation technique. The emulation helps the AV to identify the name of the malware (if it's already known). A code from the most used anti-emulation technique by malware developers is presented below. The following check uses two timers to detect any sleep emulation. If that code is ran on a personal computer, the timers would not have a difference between them. This is not the best way to do this but it is the most used. The other (better) ways will not be discussed in this paper because they could be exploited in bad ways.

```
private static void AntiEmulation()
{
   DateTime then = DateTime.Now;
   int timeout = new Random(Guid.NewGuid().GetHashCode()).Next(800, 1200);
   typeof(Thread).GetMethod("Sleep", BindingFlags.Public | BindingFlags.Static, null, new Type[] { typeof(int) },
null).Invoke(null, new object[] { timeout });
   DateTime now = DateTime.Now;

   if (((now - then).TotalMilliseconds) < timeout)
   {
      GC.Collect(0, GCCollectionMode.Forced);
      GC.WaitForFullGCComplete();
      GC.WaitForPendingFinalizers();
      throw new BadImageFormatException(null, default(Exception));
   }
}
```

3.2 Bypassing AVs at runtime (phase 3)

3.2.1 Hooks

AVAST/AVG has implemented Ring3 (userland) inline hooks. They are hooking a few WinAPIs from ntdll.dll which means that kernel32.dll APIs are also affected by it because they refer to ntdll ones at some point. The inline hooks are replacing the very first 5 bytes with a jmp instruction which jumps to a trampoline function used to find suspicious activity. In fig. 14, a hook on NtWriteVirtualMemory is installed. There is also shown that there are a few more API's whose first 5 bytes are normal (not hooked). This can be easily bypassed by using syscalls (C/C++) or manually mapping our own copy of ntdll in the remote process and using the WinAPIs from it (the same way game hacks are bypassing anti cheats).

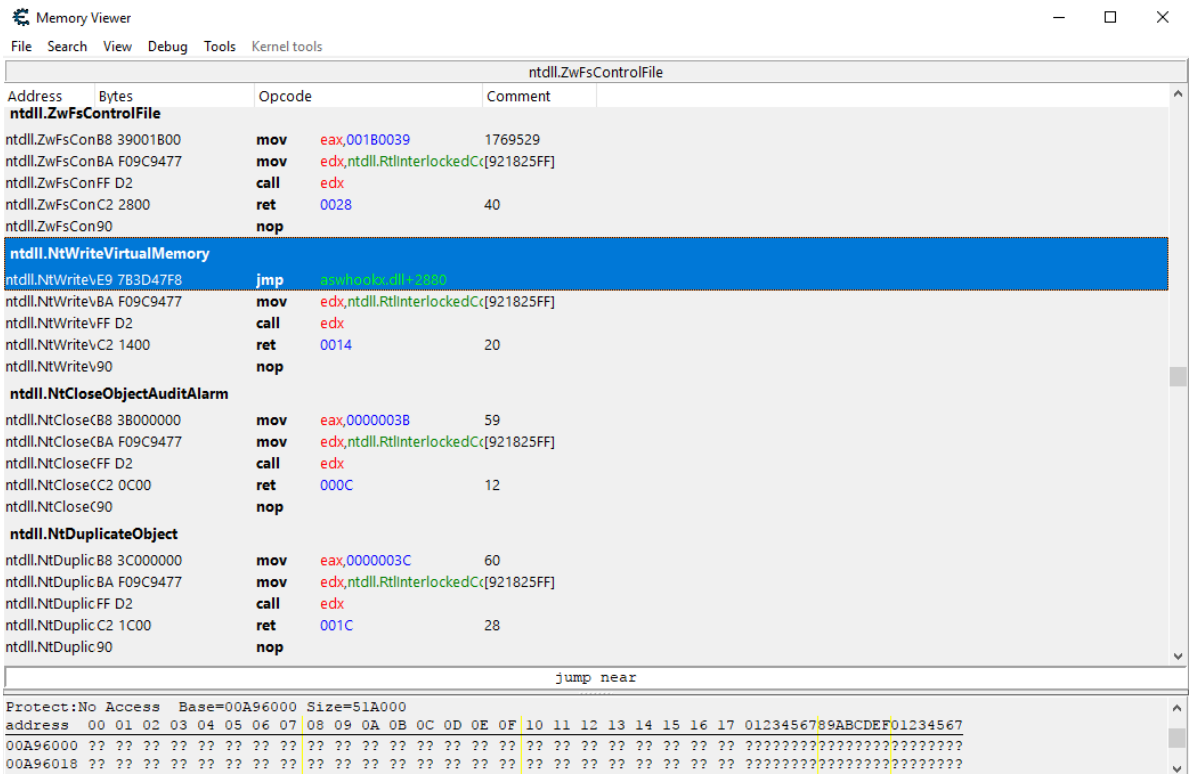


Fig. 14 AVAST/AVG NtWriteVirtualMemory Inline Hook

On the following code, all hooked WinAPIs by AVAST/AVG are presented:

```
[4392] output.exe!ntdll.dll->RtlQueryEnvironmentVariable || [0x7790B6B0] => aswhookx.dll [0x6FDA35DB] || Inline - Detour [5 Bytes] || push 20h || jmp 6FDA35E0h
[4392] output.exe!ntdll.dll->NtQueryInformationProcess || [0x7792E8D0] => aswhookx.dll [0x6FDA366B] || Inline - Detour [5 Bytes] || mov eax , 00000019h || jmp 6FDA3670h
[4392] output.exe!ntdll.dll->NtMapViewOfSection || [0x7792E9E0] => aswhookx.dll [0x6FDA29EB] || Inline - Detour [5 Bytes] || mov eax , 00000028h || jmp 6FDA29F0h
[4392] output.exe!ntdll.dll->NtWriteVirtualMemory || [0x7792EB00] => aswhookx.dll [0x6FDA287B] || Inline - Detour [5 Bytes] || mov eax , 0000003Ah || jmp 6FDA2880h
[4392] output.exe!ntdll.dll->NtOpenEvent || [0x7792EB60] => aswhookx.dll [0x6FDA2DEB] || Inline - Detour [5 Bytes] || mov eax , 00000040h || jmp 6FDA2DF0h
[4392] output.exe!ntdll.dll->NtCreateEvent || [0x7792EBE0] => aswhookx.dll [0x6FDA2C9B] || Inline - Detour [5 Bytes] || mov eax , 00000048h || jmp 6FDA2CA0h
[4392] output.exe!ntdll.dll->NtResumeThread || [0x7792EC80] => aswhookx.dll [0x6FDA2B8B] || Inline - Detour [5 Bytes] || mov eax , 00070052h || jmp 6FDA2B90h
[4392] output.exe!ntdll.dll->NtCreateMutant || [0x7792F230] => aswhookx.dll [0x6FDA2F2B] || Inline - Detour [5 Bytes] || mov eax , 000000ADh || jmp 6FDA2F30h
[4392] output.exe!ntdll.dll->NtCreateSemaphore || [0x7792F2E0] => aswhookx.dll [0x6FDA31BB] || Inline - Detour [5 Bytes] || mov eax , 000000B8h || jmp 6FDA31C0h
[4392] output.exe!ntdll.dll->NtCreateUserProcess || [0x7792F370] => aswhookx.dll [0x6FDA344B] || Inline - Detour [5 Bytes] || mov eax , 000000C1h || jmp 6FDA3450h
[4392] output.exe!ntdll.dll->NtOpenMutant || [0x7792F910] => aswhookx.dll [0x6FDA307B] || Inline - Detour [5 Bytes] || mov eax , 0000011Bh || jmp 6FDA3080h
[4392] output.exe!ntdll.dll->NtOpenSemaphore || [0x7792F980] => aswhookx.dll [0x6FDA330B] || Inline - Detour [5 Bytes] || mov eax , 00000122h || jmp 6FDA3310h
[4392] output.exe!ntdll.dll->RtlDecompressBuffer || [0x77990EB0] => aswhookx.dll [0x6FDA34FB] || Inline - Detour [5 Bytes] || mov edi , edi || jmp 6FDA3500h
```

Kaspersky and Norton have implemented SSDT Hooks [3] (kernel mode). This means that the bypass methods from AVAST/AVG will not work here because our file has only user permissions. Anyway, It doesn't mean that they cannot be bypassed. That can be done by preventing kernel drivers (for the SSDT hooks) from being loaded which will not be discussed in this paper.

3.2.1 Cloud Protection

Avira and Norton have implement a Cloud Protection which sends the file to their labs for a review. It only happens if the injection method is detected like non-modified Process Hollowing/Dynamic Forking. It can also be bypassed by pumping the file with 3 MBs. My tests found that the 3 MBs are the minimum size for the glitch. This means that if our file is 358 KB, it should be filled up with junk bytes until its total size gets at least 3 MBs. Usually those junk bytes are put at the end of the file but it doesn't really matter where exactly they will be put at.

3.2.2 Memory Scan

ESET's Memory Scanner (HIPS) is scanning all memory regions in each executable for anomalies/execution patterns of known malware. It can be bypassed by two ways. The first one is to set proper section protections by getting the values from Characteristics field and not only RWX (read, write and execute) permissions everywhere and making our newly allocated memory for the injected image (payload) either Image or Mapped (fig. 15). The second one is to spam PAGE_GUARD (protects memory region) on the headers which don't break the performance in any way. These methods will not be discussed further because they could be an object of another paper and they are not a part from any existing paper.

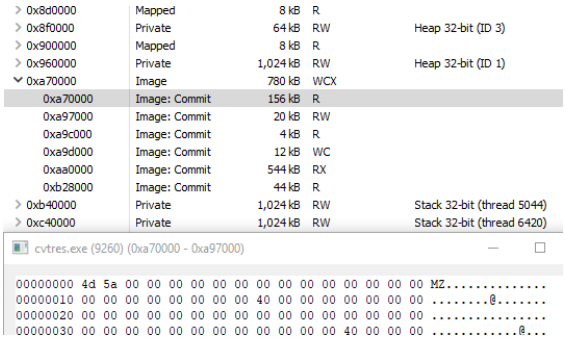


Fig. 15 Proper memory protections

4 Injection Techniques

4.1 Process Hollowing/Dynamic Forking (RunPE)

Process Hollowing/Dynamic Forking is the most used technique by malware developers. It spawns a suspended legit child process (like svchost.exe), allocates space inside it for the malware, copies PE headers and sections to that space, sets proper protection flags to the sections, patches Process Environment Block (PEB) [4] (modifying the base address to the new one) and then releases the thread. The AV solutions could have easily implemented a method which scans for mismatching fields from the PE on disk and PE in memory.

4.2 Process Doppelgänging

Process Doppelgänging is a new technique which uses NTFS cache system. It creates a transaction and then a transacted file on the disk which is not visible though because of the NTFS's cache. It then writes payload bytes into it and it creates a new section which we later map into the chosen process.

4.3 PE Loader (LoadPE)

PE Loader is also often used technique but it's more complicated to code because instead of injecting into a child process, we are injecting in the parent process' memory space. It means that we have resolve IAT, Relocations, TLS Directory, Static TLS (eventually if we aren't calling the entry point by *CreateThread WinAPI*).

5 Protection

After this research, the author of the paper finds these AVs for the most capable to use:

- Windows Defender (Windows 10) – currently the best existing scantime engine
- Kaspersky – the most effective Emulation and runtime detection methods
- Norton Security – same as Kaspersky
- ESET Antivirus – they are the fastest company in finding malware signatures

Better than nothing:

- Avira – only their Cloud service is awesome but it has glitches which makes Avira go down in the top list

The listed AVs have all of the capabilities of the not listed ones and they have shown a better performance than them. Having an Antivirus is not enough but it helps. Human's brain is the strongest protection against malware. Don't open suspicious files unless they are from a trusted source. Scan each downloaded file on http://virustotal.com before executing it because they distribute files to AV labs for further analysis and the results appear quickly. Don't look on the detection ratio from infamous AVs because they often mislead to false positive detections. Note that the executables are not the only dangerous files. There are exploits for putting an executable inside a ".doc(x)", ".mp3", ".mp4", ".pdf", etc.

6 Conclusion

This paper presents how easy the AVs can be bypassed. Every injection method could be prevented as like as it was described for Process Hollowing. The author of this paper is thinking of starting an Antivirus Software project that would implement better methods than the existing ones against malware.

Acknowledgment: This work was supervised by Professor *Katalina Grigorova* from *University of Ruse*.

References

[1] Portable Executable, https://msdn.microsoft.com/en-us/library/ms809762.aspx

[2] Thread-Local Storage, https://en.wikipedia.org/wiki/Thread-local_storage
[3] SSDT Hooks, https://resources.infosecinstitute.com/hooking-system-service-dispatch-table-ssdt/
[4] Process Environment Block, https://en.wikipedia.org/wiki/Process_Environment_Block

Svetoslav HADZIIVANOV
University of Ruse
Department of Informatics and
Information Technologies
8, Studentska str., Ruse 7017
BULGARIA
warrolen@gmail.com

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Food Ordering System

Nicolae Darius Hațegan

Abstract

Applications used to order food exist on the market, but most of them are made to help customers, rarely to help the provider to improve the order management system. The aim of this paper is to study many applications used for food ordering, to design and implement a new android app which helps customers to order food from a specific brand and to optimize providers order management. We name our application Food Ordering System. Our application is focused on provider needs but also on helping user to create complex customized orders in an optimized way. We have made research on 12 apps and we have combined features from each one in order to create a complex, useful application for both user and provider. A lot of existing applications are oriented only to the user, and not enough to the business. One example of application is "McDonald's app" which offers information about food menu or nutrition. We have selected a good feature from this application, the location one, which helps the user to find the location of a specific restaurant or to navigate to the location. Another application was "EuCeMananc" which provides the possibility to order from many restaurants from different cities. This application do not offer an option for order customization and usually requires a phone call from the provider for order validation. To the best of our knowledge, there are no applications which promotes the providers and allow the optimization of the order management system. In the first release, we will have one application for customers, another one for providers (restaurants) and one server which is the bridge between clients. Our proposed application allows customers to personalize in detail the food products by choosing al the ingredients and also offer the possibility of tracking in real-time status of the order. The user will know when the food ordered is ready through notifications. The notifications are another feature not found in others applications. The provider web app has two level of users, admin and employee. Our application is already in the development stage for a national company which has 21 fast-foods in Romania and Hungary. Our apps, Food Ordering System, has a great future ahead, being implemented by a fast-food chain.

1 Introduction

The idea of this project started with the necessity of ordering food quickly, for a big number of people, for which the menu should be customized. Using the classic method, calling the restaurant for ordering food, the time spent was big, so we came with the idea of creating an application which helps users to order customized food quick, easy and with a live status of the order.

On the market, can be found a lot of applications which satisfy the need of ordering, but also have drawbacks. These applications are oriented to a large number of restaurants, so the menu is hard to be customized. The project aims to create an application through which the user can order food from a chain of restaurants, (e.g. "McDonald's Romania"), and an application through which the provider can process the orders.

The goal also includes an easier way of processing orders, eliminating phone calls and misunderstood orders. With a fidelity system, the confidence in processing large orders increases and can be generated reports periodically, offers and vouchers for fidel customers.

Another goal is to help consumers to verify the order before is sent, after that will be able to see the status of the order in real time, all secured. The user has access to a chat system on each command for being informed in case that something goes wrong.

An advantage of this application is the ability to see other users ratings about a specific product or to leave reviews and rating for the products.

The existing solutions in food ordering and processing domain, are defined by a market analysis which helps on creating one complex product, original and which covers the people and businesses necessity.

In the research phase, we have checked multiple Android applications which came with different features and drawbacks. One tested application was "McDonald's Romania" which has nice features as the possibility to find the closest restaurant near the user with the exact location on the map, which it was considered a necessity for our application. Another feature found was the calories calculator which is a nice to have feature but will be not implemented in the first version of the app. Another good feature was the possibility to obtain vouchers based on how often the user uses the application, which we have changed the way in which the client receives the vouchers. The vouchers are received if the client has multiple orders in a short period of time, made using the application. A drawback for this application was the necessity to order food, drawback which has been transformed for our application into a feature. Another tested application was "UberEats" [6] which offers the possibility to pay or to customize products. One drawback is that it is implemented in Romania only in Bucharest, so it would be impossible to use it in other city. Although, in the research phage we have found a nice to have feature on "UberEats" [6] which estimates the order processing time.

2 Application structure

Food ordering system is splitted in three applications, one of them is Android application, created for ordering food, the second is the application server, which is the core, and the last one is a web application, created for processing orders and managing menu or to generate reports (Fig 1).

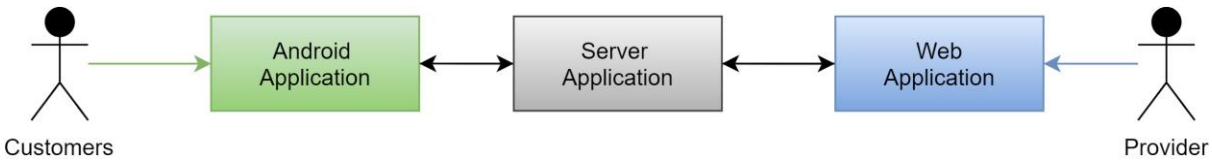


Fig 1 Applications and how customers and provider interact with them

In the Android application, the code is structured in two layers (Fig 2): the data layer that was in charge of retrieving/saving data from REST APIs and persistent data stores; and the view layer, whose responsibility is to handle and display the data on the UI.

The APIProvider provides methods to enable Activities and Fragments to easily interact with the REST API. These methods use URLConnection and AsyncTasks to perform network calls in a separate thread and return the result to the Activities via callbacks.

In a similar way, the CacheProvider contains methods that retrieve and store data from SharedPreferences. It also uses callbacks to pass the result back to the Activities.

In few words, the application use Activities and Fragments to display Information via XML. The information came through services as a response to call from the application server.

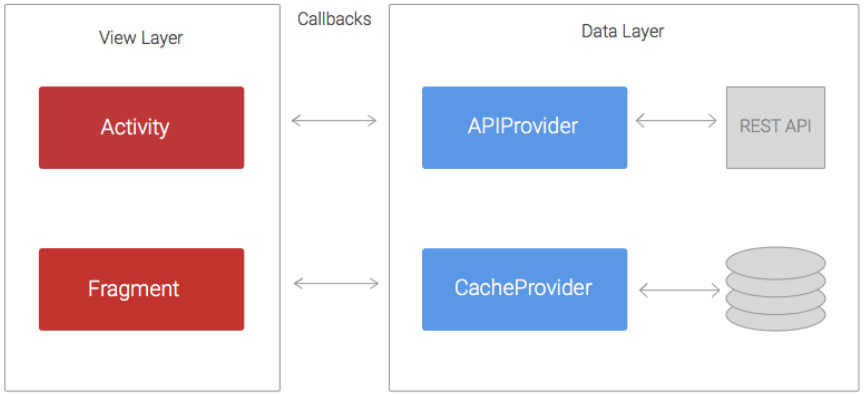


Fig 2 Android application structure based on layers

The Android application creates calls to the server which has the role to manage the call and return a response and/or push information to the web application. The server being the core of the system, he needs to be well structured, and it is, to be easy to adapt to new features, to change the frameworks quick and easy.

The server architecture is splitted in layers which also have a common part (Fig 3). First layer is “Services Layer” which has the role to expose services and to create calls or push notifications for clients (Android and Web application). The service layer also includes the security section which is defined by authentication and authorization. Authentication is used for creating a session between client and server while authorization it means that the client has access to a service or another component.

After the client (the application) is authenticated with the server, we assume that he has full rights for the web services and he is ready to make any call to the server. When the user made a call, the server catch in service layer the request, it validates the request and the next step is to tell to the “Business Layer” to execute an action.

The middle layer, the business one, it is designed to contain the logic, to process the request details and based on them to perform some operations. For example, when an order is sent from the Android application, the server catch the request in service layer, then the service layer send the details to business manager and he should validate the order, save to the database and push the order to the web application as well, so the provider can see the order in real time in his list.

The bottom layer, also named “Data Layer” is designed to handle data. This layer works with the database and/or external services (when it needs information from a third party service). For the database here are defined “Entities” which represents tables from the database with relations between them, “Data Access Objects” are used to interrogate the database, basically to create queries for receiving information. In the bottom layer we also have “Data Helpers”, usually used for creating translators for data types.

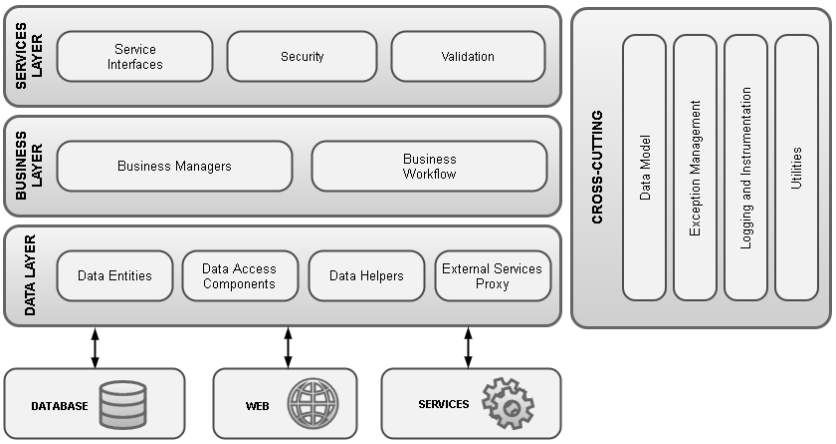


Fig 3 Server architecture based on layers

The last layer of the server is “Cross-Cutting”, created to define the common need between all layers. All cross cutting objects are important to the entire server because it starts with transfer objects, which are used to transfer data between layers, and proceed by exceptions which should be used to send messages from the lowest layer to the clients, and finished with instruments for logging and utilities classes which are used to track the actions and to define helpers for all layers.

The web application is under development and the architecture will be under MVC concept (Fig 4), which means “Model-View-Controller”. This architecture type is used to split the view from controller and the model as conceptual objects, and to create links between them. The view it is actually the HTML page which is used to display data, while controller is where the logic is implemented and the model is the place where the information is.

The web application use TypeScript for controllers and use web services to get information from the server.

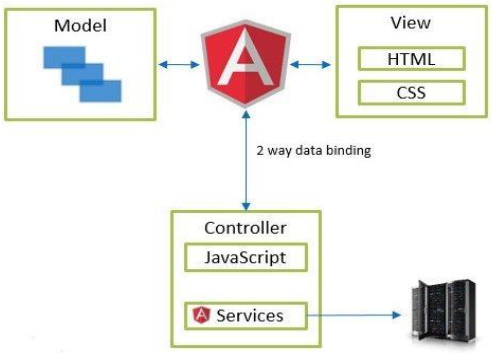


Fig 4 AngularJS MVC architecture

Based on the research, we have created a list of features which are nice to have, followed by prioritization. A first version was created using the ones with the highest priority (Fig 5).

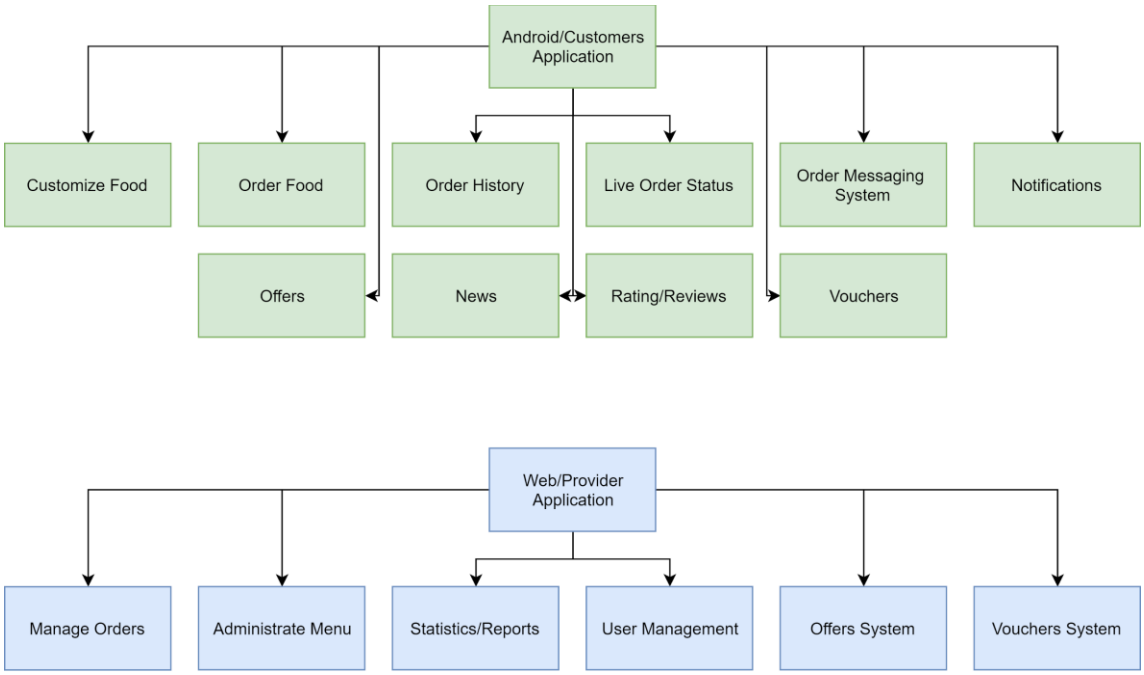


Fig 5 - Features fixed for Android and Web application

3 Functionality of the applications system

3.1 Android/Customer application

The application is developed exclusiv on Android operating system and it is created for the people who wants to create orders without the calling of provider and (in some cases) waiting in a queue.

The scope of the Android app is to help workers, students or other kind of people for which the time for ordering/buying food, books, electronics or all other things is small. Currently, the system is prepared for a food chain restaurant.

The application starts with a splash screen which is used to display a logo until the data is prepared for the client. Further, with the second screen, the interaction with the restaurant chain starts. The current screen offers the possibility to choose the wanted restaurant from the list (Fig 6), or from the map (Fig 7). The user can also search for restaurants by city name using the search field.

The list of restaurants shows for each restaurant the distance between customer and restaurant and for how much time it is still open or if it is closed.

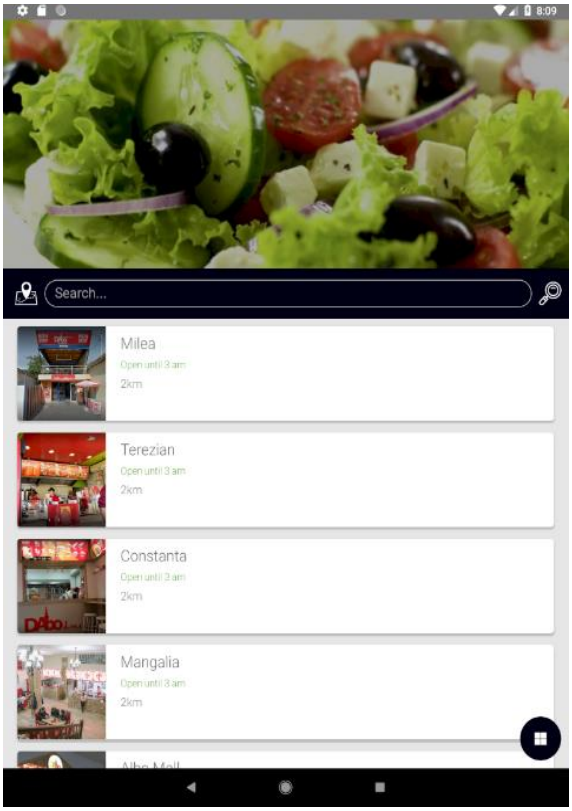


Fig 6 Restaurant chain displayed as a list

Fig 7 Restaurant chain displayed as a map

After restaurant is selected, then the customer is ready for adding the products, food in our case, to the cart. This page (Fig 8) contains three sections, first one being the menu section which is used to see the products (food) with name, price, rating, or to access product details by clicking on the image, to customize the product with ingredients and add it to the cart. The selected restaurant has in the common (top) side for all sections, a button which creates the route between the customer and the restaurant, and open the navigation to the selected restaurant. Also, in the common side exist two buttons which redirects users to the Facebook or Twitter page of the restaurant.

The second section of the screen is the preorder (Fig 9) where the products from the cart can be seen/edited/modified and the preorder can be sent or deleted. The products details can be seen separately or for the entire cart. When the user tries to edit a product, then he can add extras, modify the quantity or add a comment for the edited one (Fig 10).

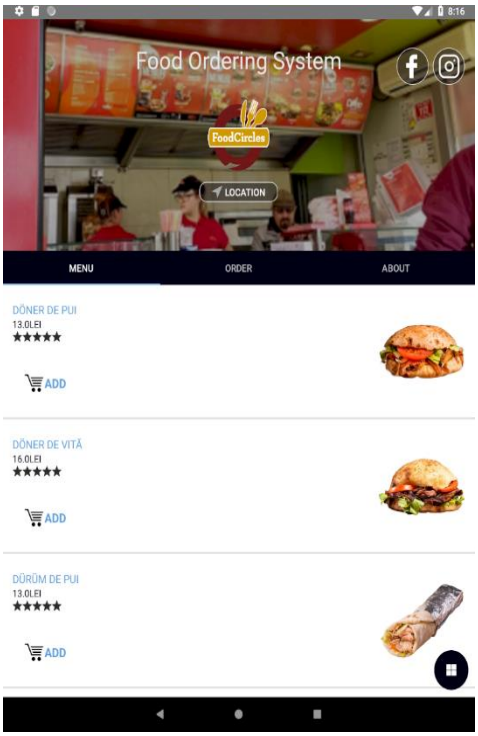


Fig 8 The restaurant screen, menu section

Common to the all pages of the application is the button from the bottom-right side, which when is clipping, then the customer is notified because he has new messages or one of his orders has a different state (e.g the order was processed). If the button is pressed, then a contextual menu is opened and the customer has the possibility to open notifications, to see the orders history, to open settings or to register/log in if he is not authenticated. Is not necessary for the user to be authenticated to see the menu and to add products to the cart but to place the order, he needs to be authenticated.

Once the order is sent, it goes through five states: “Pending”, until the provider confirms that it can be done (there are situations when the stock is empty for some products), “Confirmed”, until de provider starts to work on it, “In progress” until the order is done, “done” and the last one is “Delivered” when the order was picked up. The last state is important because based on

it, the last section of the restaurant screen, contains vouchers and offers which are generated based on number of delivered orders.

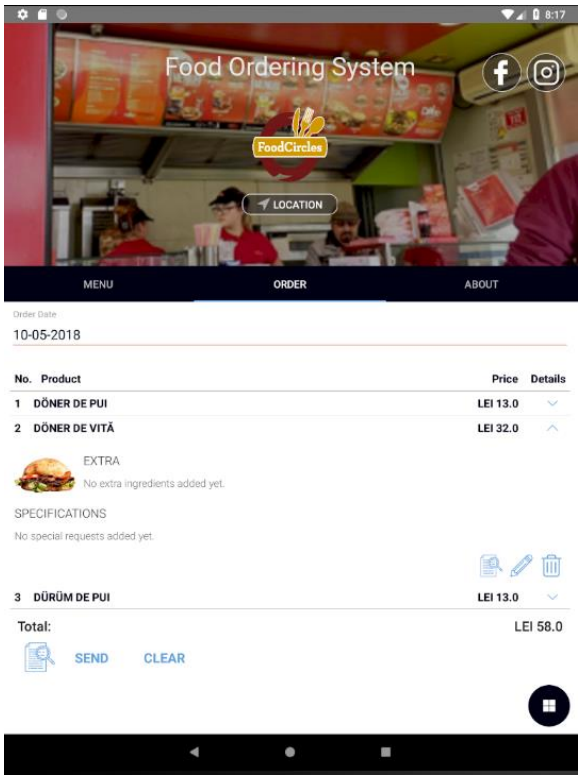


Fig 9 The restaurant screen, order section

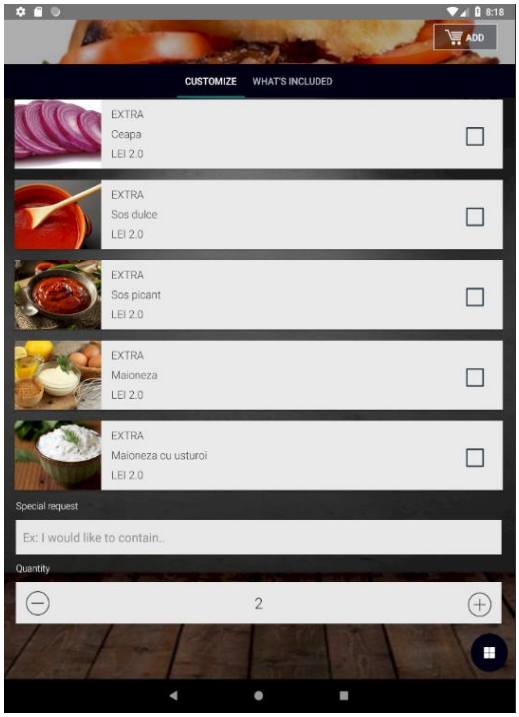


Fig 10 Edit product details and add to cart page

3.2 Web/Provider application

The customer or provider does not access directly the server, but through applications (Web and Android). Because of that, what we need to know for the moment about the server is that it is the brain of the system, where all information is kept. Because of that, we will discuss about the server only to the application structure because there can be displayed how it actually works.

The provider application has three main features in the first version, followed by new features in next versions. One important point is that every restaurant has at least one user account (Fig 11). The authentication is required in the managing orders process because orders are linked to a restaurant and a web application user is linked to a restaurant. The user can request a new password.

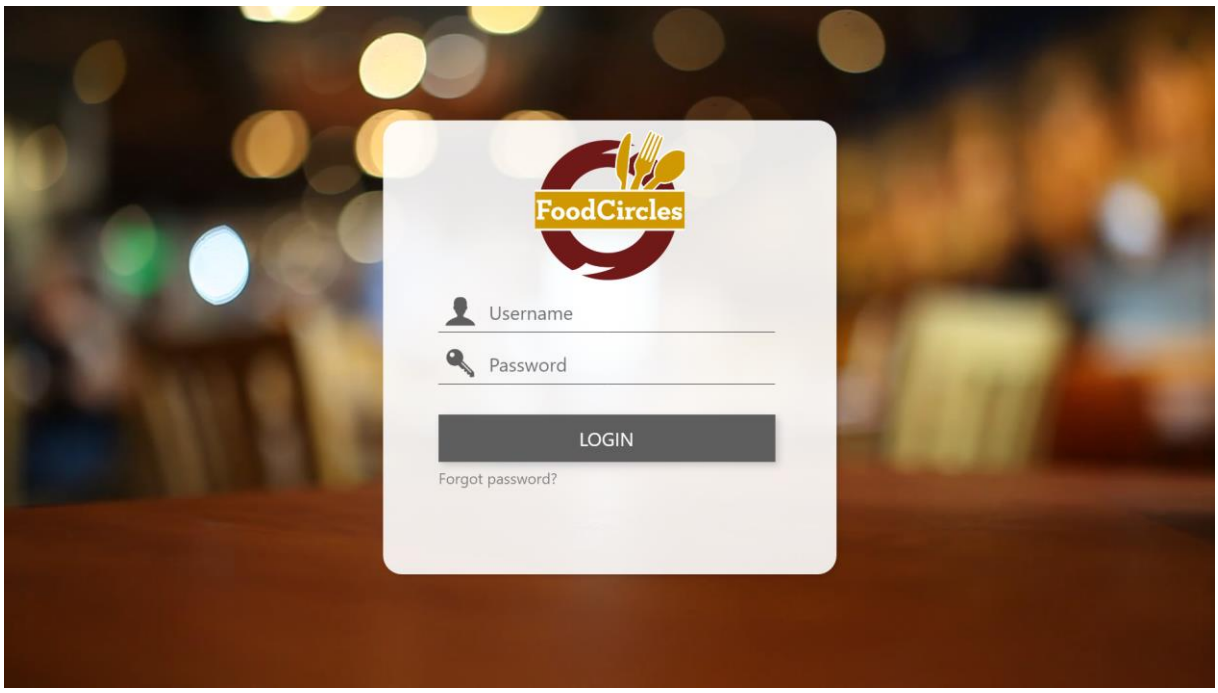


Fig 11 The login screen for the provider app

Once the user (employee) is logged in, the dashboard displays the main feature, the one of managing orders. Now he can start to process orders. In the left side of the screen (Fig 12), a list of orders is displayed, in chronological order. First orders are the older, which needs to be processed. Every order in the list has small details as the timestamp of the order, street delivery, the customer name which made the order. If the employee wants to see the prepared orders, then he can change easily the list of orders by choosing between “TO-DO” and “DONE”. The right ride of the screen contains the selected order details and the state of order. The employee should press “Confirm that can be done” if the order can be done. The next step would be to start to work on the order and after the order is done, to press the done button. The last step on processing orders is to press delivered button after the order was picked up or delivered, to update the user fidelity state. The fidelity status can generate vouchers for users or can raise questions for provider when an user refused to pick up orders.

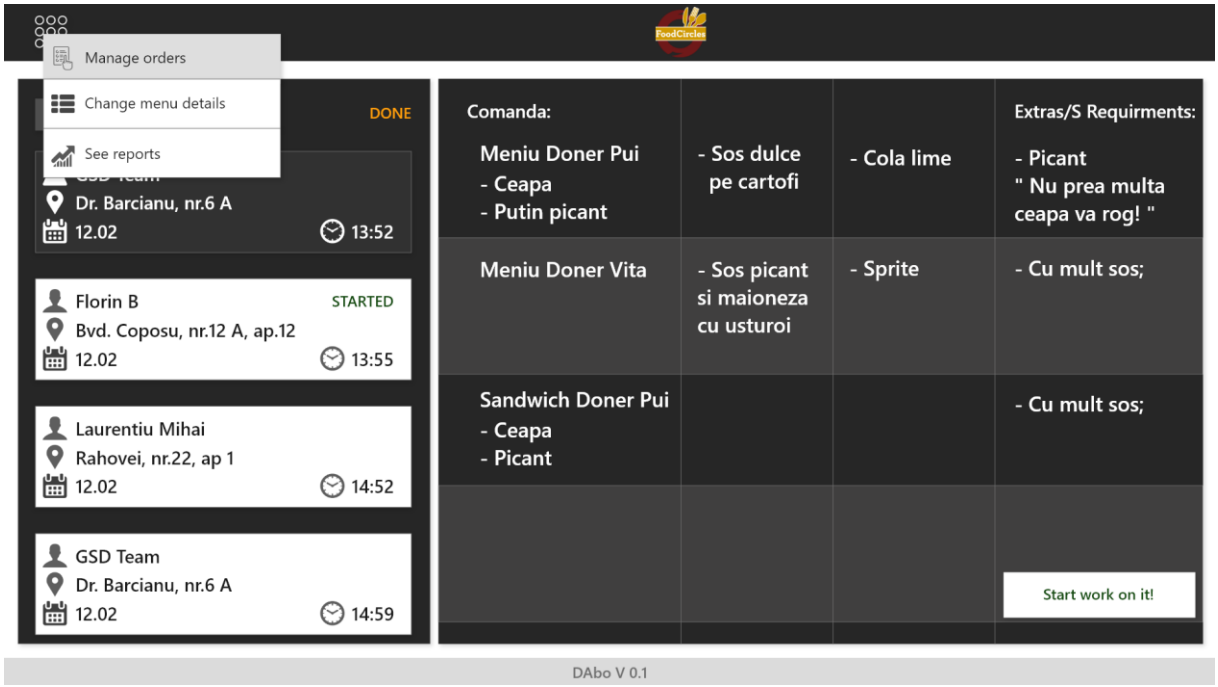


Fig 12 Orders dashboard

The second feature is to administrate menu details which affects the Android application. Also, here it can affect a presentation site if it is linked to our server for receiving the menu. The menu is splitted in categories, so the provider can easily change the category and to add or edit the product. The operations offered by the application will be to create a new product in any category, to edit products details or to delete products.

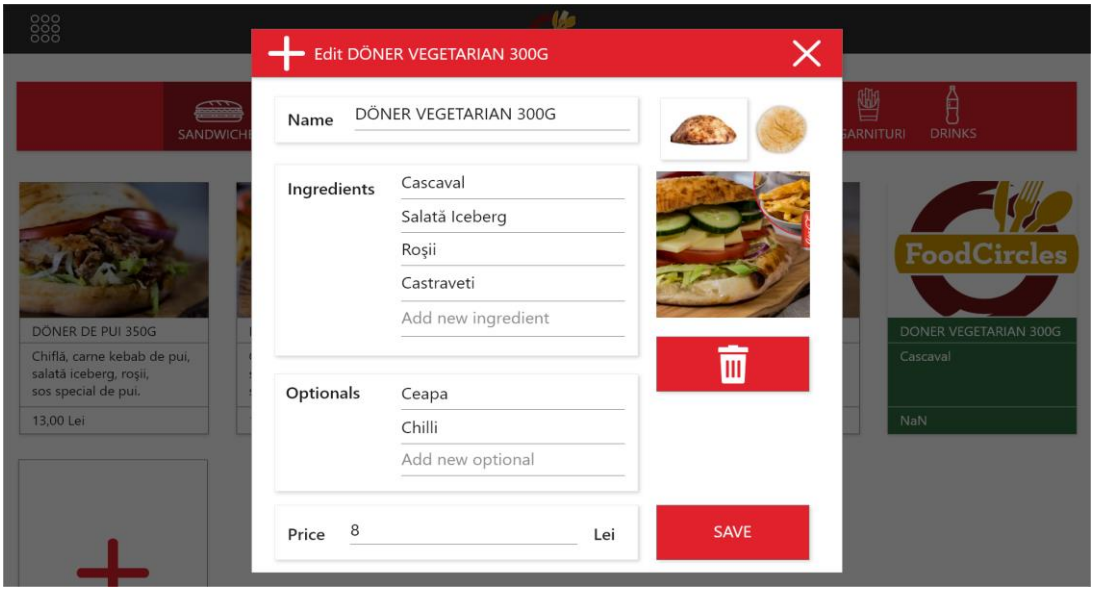


Fig 13 Edit product details

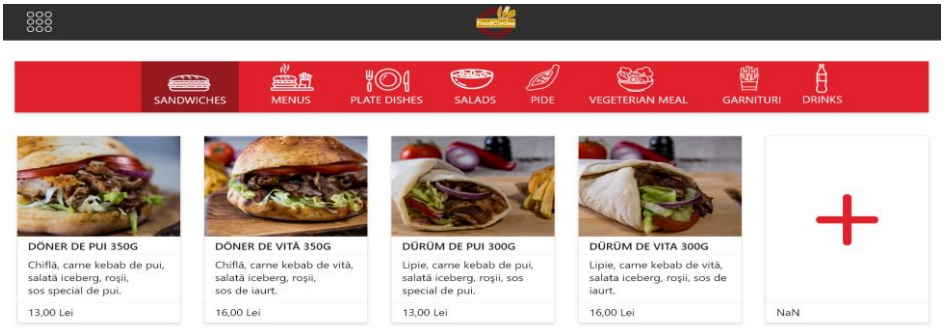


Fig 14 Screen for menu administration

4 Integration framework. Program Code

The system starts with the customer application which is developed on Android operating system using Java as programming language and a few libraries. As it was mentioned earlier, the architecture is based on MVC concept. Views are created using Activities and Fragments , as we can see Fragment defined in XML file (Fig 15) with the reference to the class which extends the Fragment.

Definition 1[3] *A Fragment represents a behavior or a portion of user interface in a FragmentActivity. You can combine multiple fragments in a single activity to build a multi-pane UI and reuse a fragment in multiple activities. You can think of a fragment as a modular section of an activity, which has its own lifecycle, receives its own input events, and which you can add or remove while the activity is running (sort of like a "sub activity" that you can reuse in different activities).*

Definition 1[4] *An activity represents a single screen with a user interface just like window or frame of Java.Android activity is the subclass of ContextThemeWrapper class.*

```
<RelativeLayout android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/res/android" >

    <fragment xmlns:android="http://schemas.android.com/apk/res/android"
        android:id="@+id/map"
        android:name="mobile.gsdgroup.net.foodtruckapp.utils.StoreListMapFragment"
        android:layout_width="match_parent"
        android:layout_height="match_parent">

    </fragment>
</RelativeLayout>
```

Fig 15 Fragment defined in XML

Further I will exemplify the communication between the application system through the code, but just the steps which include the use of new frameworks or libraries.

In the Android application, after the customer will add to the cart the food he wanted, he creates an account through services. After that, he must authenticate to place the order. In the Fig 16 is presented how a service is called using the Volley library [5] from Google which helps to create a queue of calls to the service. Volley has the following advantages:

- Automatic scheduling of network requests.
- Multiple concurrent network connections.
- Transparent disk and memory response caching with standard HTTP cache coherence.
- Support for request prioritization.

- Cancellation request API. You can cancel a single request, or you can set blocks or scopes of requests to cancel.
- Ease of customization, for example, for retry and backoff.
- Strong ordering that makes it easy to correctly populate your UI with data fetched asynchronously from the network.
- Debugging and tracing tools.

To create a call to the authentication service, it needs a GET request, in which the username is encoded and attached to the header (line 158 Fig 16). On the response, if it is a valid one, we receive a json in the body which should be mapped to a transfer object, for these we use Jackson library which offer a mapper to automatically parse the Json and set the values to an new object.

Further, the request arrives on the server, in the exposed service. The services are created using Jersey which is an implementation of RESTful web services in Java (Fig 17). Besides Jersey, we use Spring for dependency injection or inversion of control. In the Fig 17 we see that the class is annotated with "@Component" which tells to Spring that this class will be a generic stereotype. Also we have defined some variables annotated "@Autowired", which tells spring that somewhere is an implementation for that interface and when the variable is needed, then an object should be created and linked to the reference (that is inversion of control and dependency injection). The "getUser()" method is annotated with Jersey's annotations, "GET" the type of request for which the method responds, "Path(..)" used to define the url section for which the method responds, and the last one, "@Produces(MediaType.APPLICATION_JSON)" which is used to tell that the response body is of type JSON.

```
116     @Override
117     public void callGetUserService(final User user) {
118         //create json from user object.
119         final ObjectMapper objectMapper = new ObjectMapper();
120
121         StringRequest getRequest = new StringRequest(Request.Method.GET, GET_REQUEST_URL,
122                 new Response.Listener<String>()
123                 {
124                     @Override
125                     public void onResponse(String response) {
126                         try {
127                             User result = objectMapper.readValue(response.toString(), User.class);
128                             userAccountCallBack.getUserCallBack(result);
129                         } catch (IOException e) {
130                             Log.d( tag: "Parser-Error", e.toString());
131                         }
132                     }
133                 },
134                 new Response.ErrorListener()
135                 {
136                     @Override
137                     public void onErrorResponse(VolleyError error) {
138                         NetworkResponse networkResponse = error.networkResponse;
139                         if (networkResponse != null && networkResponse.data != null) {
140                             try {
141                                 String newData = new String(networkResponse.data);
142                                 ServiceError inputServiceError = objectMapper.readValue(newData,ServiceError.class);
143                                 userAccountCallBack.getUserCallBack(inputServiceError);
144                             } catch (IOException e) {
145                                 e.printStackTrace();
146                             }
147                         }
148                     }
149                 }
150         ) {
151
152             /**
153              * Passing some request headers
154              * */
155             @Override
156             public Map<String, String> getHeaders() throws AuthFailureError {
157                 HashMap<String, String> headers = new HashMap<>();
158                 headers.put(AUTHENTICATION_INFO, String.copyValueOf(SecureData.encode(user.getUsername().toCharArray())));
159                 return headers;
160             }
161         };
162         AppServiceController.getInstance().addToRequestQueue(getRequest);
```

Fig 16 Authentication service call

```
@Component
@Path("/users")
public class UserWebService implements IUserWebService {


    @Autowired
    private IUserManager userManager;

    @Autowired
    private ISecureData secure;

    @GET
    @Path("/user")
    @Produces(MediaType.APPLICATION_JSON)
    @Override
    public Response getUser(@Context HttpHeaders headers) throws TechnicalException {
        String decodedLoginInfo = String.copyValueOf(
        secure.decode(headers.getRequestHeader("Authentication-Info").toString().toCharArray()));

        UserTO user = userManager.getUserByUsername(decodedLoginInfo);
        if (user != null && user.getId() != null) {
            user.setPassword(null);
            return Response.status(Status.OK).entity(user).build();
        }

        GenericError error = new GenericError();
        error.setCode(GenericErrorCodes.CODE_1005);
        error.setMessage(GenericErrorCodes.MESSAGE_1005);
        return Response.status(Status.BAD_REQUEST).entity(error).build();
    }
```

Fig 17 Expose web service on server side

The request it is now on web service exposed method and what needs to be done is to tell to the "UserManager" (which has all the logic), to return the user which has the defined name. The manager calls an object named "DAO" which can be compared with a class which defines operations on the database (only on the user table or what implies user). In the Fig 18 it is exemplified a method for getting user by username from the database. The relation with the database is created by Hibernate, another strong framework which help developers to have a transparency on using the database. Hibernate allows developer to create generic queries using HQL or using Criteria, and on runtime, Hibernate translates the query for the specific database. Basically, the database has no relevance if the native sql is not used. In our example, Criteria it is used to get the results because it's logic is based on restrictions and we want the user which has a specific username.

```
@Repository(value = "userDao")
public class UserDaoImpl implements IUserDao {

    @Autowired
    private SessionFactory sessionFactory;

        @Override
    public UserTO findByUsername(final String username) {
        Criteria cr = sessionFactory.getCurrentSession().createCriteria(User.class);
        cr.add(Restrictions.eq("username", username));
        User user = (User) cr.uniqueResult();

        if (user == null) {
            return null;
        }
        return UserTranslator.getInstance().entityToTransferObject((User) cr.uniqueResult());

    }
```

Fig 18 Data access object implementation to find user

Once the result is created, then it will be transformed from entity to transfer object and sent back to the business which will perform some validations and send it back to the service. The service

will send the user in the body with the response code “200-OK” if there is an user, otherwise, a bad request error will be sent to the Android client.

Because the web application is under development, in the stage of design and preparing for implementation, we do not have what to present related to the implementation.

5 Conclusions and further research directions

This article aims to present a system of applications which work together to help people to order products and to help merchants to manage orders. Using two applications, one for customers and the other for providers, the system can be understood as a tool which eliminate the straight connection between the customer and the provider, and creates an online bridge which helps to eliminate the synchronization between them.

The goal of this system is to optimize two-side operations, the first being the product ordering (in our case food ordering) by performing orders in a easy and quick way, and the second being the order processing, menu administration and reports which create a straight system in order to process orders.

The current system is developed to satisfy the needs of provider, to eliminate ordering by phone, to have orders reports, giving in return to the customers for using the application vouchers and offers.

In the future we would like to add online payment for the orders, a delivery module which allows the user to track the deliver live.

Also, a new feature is to chronometer the order processing time and to create estimations based on the calculations, helping the user to estimate order duration.

Lately, the data sent through the internet became more and more sensitive and because of that, we want before we publish or sell the application to create a strong encrypting system for data.

In our opinion, the application can it is very useful because based on experience, we consumed a lot of time to make orders and we observed that the restaurant had a employee only to take orders by the phone.

Acknowledgement: This work was supervised by Professor *Dana Simian*, from *Department of Mathematics and Informatics, Faculty of Sciences, “Lucian Blaga” University of Sibiu.*

References

[1] Research object, *McDonald’s Romania*, https://play.google.com/store/apps/details?id=ro.mcdonalds.android&hl=ro (accessed 02 May 2018)
[2] AngularJS MVC Architecture, *Inkoniq?*, https://inkoniq.com/blog/googles-baby-angularjs-is-now-the-big-daddy-of-javascript-frameworks/ (accessed 04 May 2018)
[3] About Fragments, *Google Documentation,* https://developer.android.com/guide/components/fragments. (accessed 05 May 2018)
[4] About Activities, *Tutorials Point*, https://www.tutorialspoint.com/android/android_acitivities.htm (accessed 03 May 2018)
[5] The advantage of using Volley library, Volley Overview, https://developer.android.com/training/volley/ (accessed 03 May 2018)
[6] Research on Uber Eats, *Uber Eats Romania*, https://play.google.com/store/apps/details?id=com.ubercab.eats&referrer=mat_click_id%3Dd0d1b05202051

46518c81248a5a3255c-20180510-7336%26utm_campaign%3DInternal%26utm_content%3DInternal%26utm_medium%3DInternal%26utm_source%3DInternal%2B-%2BWebsite%26utm_term%3DInternal (accessed 07 May 2018)

Nicolae Darius HAȚEGAN
Lucian Blaga University of Sibiu
Faculty of Science
Str. Dr. I. Ratiu, No.5-7, Sibiu, 550012
ROMANIA
E-mail: hategandarius71@gmail.com

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Secret synergy effects from physics and predictive analytics - The future of mobility?

Wolfgang Heinz

Abstract

All kinds of driving assistance systems are used by present-day cars to support drivers. Innovations in engineering and a rising environmental awareness led to safer, more efficient and more economic cars in the field of individual traffic. Information technology developed besides the classical automotive engineering and now, cars finally consist of engineering as well as information systems. This paper aims to discover the potential of physics and predictive analytics to reduce fuel consumption and emissions. It tries to answer the question if it is possible to increase the efficiency of traffic through the interconnection between classical physics and modern information technology. First, due to the complexity of the subject, the mathematical basics have to be designed and documented with a special focus on driving dynamics and resistance. A formula for driving economics has been setup and the algorithm was tested and transferred into a data model. The model was built of individually collected and digitalised data. The gradient data of the track was raised in cooperation with the local freeway department. Finally different model based tests were performed and documented. To support the findings, an expert was asked for his professional and personal evaluation Findings. Different simulations on the said model have shown a decrease in fuel consumption of 1.4% on average. The model is of course unable to contain every aspect of a real scenario. Traffic congestion of construction sites at the freeway were not considered.

1 Introduction

Modern cars do have all kinds of electric assistance systems to support drivers. Technological improvement and a growing awareness for the environment in the field of individual traffic led to safer, more efficient and more economic vehicles. Besides the improvement of automobile engineering, information systems also evolved. Therefore, modern cars are pieces of engineering art, as well as information systems. The combination between physics and integrated IT in manufacturing led to aerodynamic cars, for example. The question is, can an interconnection between classical physics and modern IT-Systems help to make the passenger- and goods logistic more efficient? To answer this question, in this study different methods were used to calculate the fuel consumption in theory and a digital model was designed to run different tests on. During the study, the focus was always to generate reliable and comparable data.

2 Theoretical background

In this section, the fuel consumption is calculated in theory and transferred into a digital model.

2.1 Basics

To calculate the fuel consumption, the driving resistance (F_{DR}) is needed. It is the total of all forces the vehicle must overcome to move on a track.[1]

Those forces are:

- drag (F_{drag}),
- rolling resistance (R_{roll}),
- climbing resistance (F_{climb}),
- acceleration resistance (F_A).

The formula to calculate the driving resistance (F_{DR}) in Newton [N] is basically the sum of the different driving resistance forces:

$$F_{DR} = F_{drag} + F_{roll} + F_{climb} + F_A \quad (1)$$

The formula can be used to gain knowledge about the required driving force to accelerate on tilt or flat tracks, or even to keep speed.

2.1.1 Drag

Air resistance is one of the main factors to influence the fuel consumption. The more aerodynamic a car is, the less force it has to build up to gain or maintain speed. The drag is calculated through the following formula[2]:

$$F_{drag} = \frac{\rho_{drag}}{2} \cdot cw \cdot A \cdot V_{rel}^2 \quad (2)$$

where:

F_{drag} represents the drag in [N],

p_{drag} is the air density in [kg/m³],

cw stand for the drag coefficient and depends on the shape of the vehicle,

A represents the endface of the vehicle in [m²],

V_{rel} is the relative speed of the vehicle in [m/s].

[1] Alfred Böge.Physik – Grundlagen, Versuche, Aufgaben, Lösungen. Braunschweig/Wiesbaden. Page 49, 1991. ISBN 3-528-54046-X

[2] Herbert B. Callen. Thermodynamics and an Introduction to Thermostatistics. New York 1985. ISBN 0-471-86256

2.1.2 Rolling resistance

Due to the elastic deformation of tires and the track on the contact, the rolling resistance appears in dependence of the mass of the vehicle. This means, the heavier a car is, the more rolling resistance it has to overcome.[3] Eventhough it is not considered in this study, a possible payload will also influence the fuel consumption. The value *mFzg* is assumed as 0 in the following calculation[3]:

$$Froll = (mFzg + mzu) \cdot g \cdot f\text{roll} \cdot \cos(\alpha), \tag{3}$$

where:
F_{roll} is the rolling resistance in [N],

mFzg stands for the mass of the vehicle, in [kg],

mZu is possible payload in [kg],

g represents the acceleration of gravity and is used as, $g = 9.81$ m/s^2,

f_{roll} means the rolling resistance coefficient (depends on speed),

α represents the lead angle in Rad. On decelerating gradients, the value is negative.

2.1.3 Gradient resistance

The gradient resistance appears on every slope and gradient. It is the reason for a higher fuel consumption on positive slopes. Therefore, the formula is influenced by the car weight and the shape of the track. The following formula shows how the gradient resistance is calculated[4]:

$$F_{climb} = (mFzg + mZu) * g * sin(\alpha), \tag{4}$$

where:
F_{climb} represents the gradient resistance, in [N],

α stands for the gradient angle. It depends on the gradient and can be positive or negative.

Slopes and gradients are usually given in percent. Therefore p is needed:

$$p = \frac{s\%}{100\%} \Rightarrow F_{climb} = m_{Fzg} \cdot g \cdot \frac{p}{\sqrt{1+p^2}} \tag{5}$$

[3] Dobrinski/Krakau/Vogel. Physik für Ingenieure. Stuttgart. Page 274, 1993. 3-519-26501-X

[4] Dobrinski/Krakau/Vogel. Physik für Ingenieure. Stuttgart. Page 248, 1993. 3-519-26501-X

2.1.4 Acceleration resistance

The acceleration resistance appears when a vehicle is quickening. Technically it is the force to overcome in order to speed up. Decelerating a car, by breaking for example, does negate the value of a. This formula is used to calculate the acceleration resistance[5]:

$$F_A = (ei \cdot mFzg + mZu) \cdot a, \quad (6)$$

where:
the acceleration resistance [N] is described by F_A.

ei represents the mass factor. The moment of inertia in the powertrain, depending on the transmission is also considered.

a stands for the acceleration of the veficle and is measured in [m/s^2].

2.1.5 Applying the formula

Formula (1) can be used to calculate the total driving resistance in Newton [N]. It is needed to know how much force the car needs to develop because the fuel consumption depends on it.

For the first example, the force is measured on a specific point on the track. For the calculation of the drag and air density the value 1.2[kg/m^3] is used. The CW value is 0.27 and the front end of the car is 2 [m^2]. The values descended from a VW-Golf 7, which is used as a test object, value wise.

At a speed (V^2rel) of 100 km/h, or 27.7 [m/s], the drag, calculated according to (2), equals 2487[N] = 1.22 * 0.27 * 2 * 27.7^2.

To calculate the rolling resistance, the formula (3) is used. Again, the VW Golf 7 is used as a test object: the weight, without payload is 1400 kg. Gravitational acceleration is 9.81 and the lead angle is assumed by 0.8.

Therefore the rolling resistance is:
$F_{roll} = 1400 * 9.81 * 0.013 = 178.5$ [N].
The formula to calculate the gradient resistance is as shown in point 2.1.2 very similar:
$F_{climb} = 1400 * 9.81 * 0.019 = 260.95$ [N].

The following table shows mass factors at different gears, to calculate an approximate value, the average massfaktor value of 1.14 is used in the following.

	1.gear	2.gear	3.gear	4.gear	5.gear	average
car	1.32	1.15	1.10	1.07	1.06	1.14

The mass factor e depends on the transmition. If a car travels at 100 km/h or 0,27 [m/s], the acceleration resistance calculates:
$F_A = (a14 * 1400)*3 = 4788$ [N].

[5] Hans-Hermann Braess, Ulrich Seiffert. Handbuch Kraftfahrzeugtechnik. Braunschweig/Wiesbaden. Page 11-12, 2013. ISBN 978-3658016906

As all the resistances were correctly calculated, the driving resistance can be summed up:
F_{DR} = 248.6[N] + 178.5 [N] + 260.95[N] + 4788 [N] = 5476.06 [N].

The F_{DR} is the resistance the vehicle has to overcome. It does not have interconnection to time or how far the car has traveled yet, but to calculate the fuel consumption those values are needed. Fig.1 shows an acceleration process from 0 – 100 km/h and an acceleration of 3 m/s:

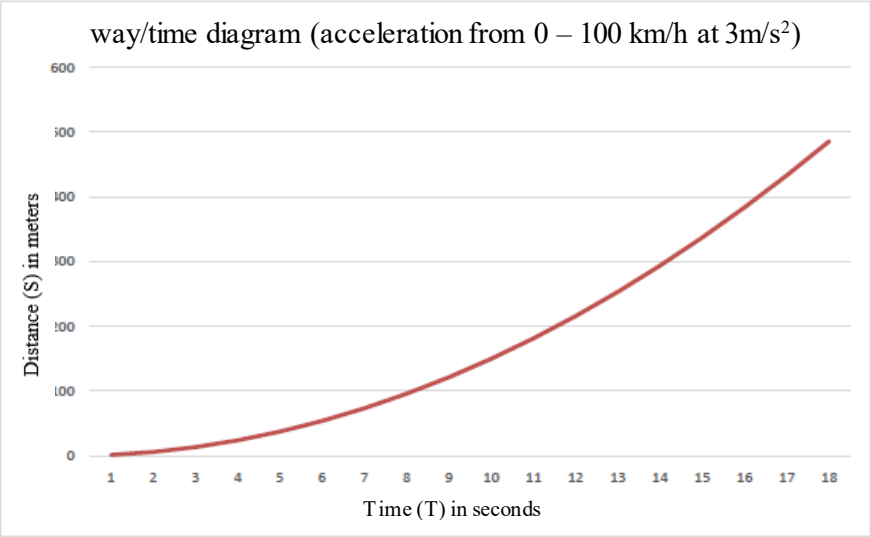


Fig. 1 way/time diagram

2.3 Simulation

A very important requirement was a realistic mapping of the track. The gradient values were crucial. Because the exact gradient data is not public, they needed to be found in cooperation with the north bavarian motorway authority, department Würzburg. As a testtrack the operational kilometer 207663 to 258292 of the german Autobahn 3 was selected, because these 50 km are very well documented and have a pretty balanced relation between ascendening and descending gradients.

Fig. 2 shows a gradient cross section from the analysed data.

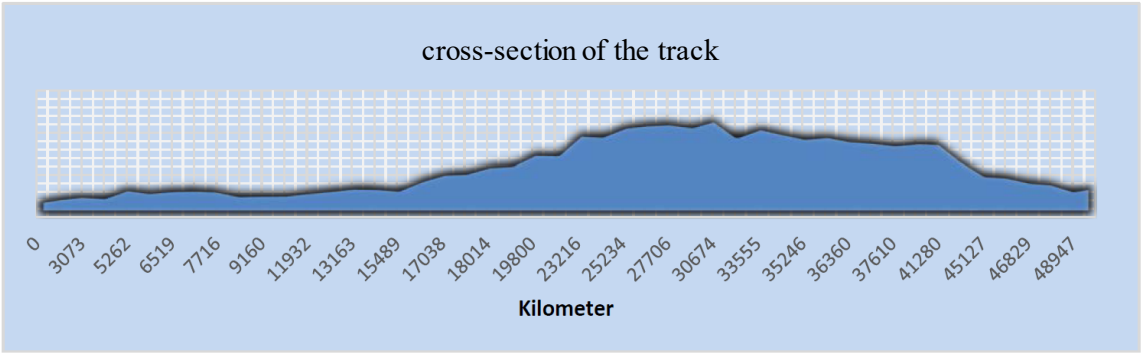


Fig. 2 cross-section oft he track

The testtrack was needed to calculate the fuel consumption for different speeds. Therefore, distance points were used to build sections with either ascending or descending gradients. The calculation of

the driving resistance is used using formulas (1) – (6) The differences in fuel efficiency between constant and adaptive speed were in focus.

First description oft he tests:

Four simulation scenarios were considered. Results are shown in Table 2.

Experiment 1

At first it was observed how much fuel a vehicle would need to drive the track at a constant speed of 100 km/h. The following table, contains the time, sum of the different resistances in (KJ) and fuel consumtion in litre per 100 kilometre.

Traveltime in minutes	Resistance in (KJ)	fuelconsumption in litre
60	≈ 4116	≈ 6,046

The fuel consumption is close to the manufacturer's instructions.

Experiment 2

In the next simulation the vehicle was moving at adaptive speed (between 110 km/h at descending gradients and 90 km/h at accending gradients).

Traveltime in minutes	Resistance in (KJ)	fuelconsumption in litre
≈ 61	≈ 4065	≈ 5,978

Even the small change in speed already influenced the fuel consumption in a positive way. 1.125% less fuel consumed.

Experiment 3

Now the simulation took place at higher pace and constant speed of 150 km/h.

Traveltime in minutes	Resistance in (KJ)	fuelconsumption in litre
40	≈7518	≈ 11,06

As expected, it took less time to arrive but also the fuel consumption increased.

Experiment 4

As a final test, the basic speed of 150km/h was varied by 10% (15 km/h) adapted on the gradient of the track. The result was the most significant during this study.

Traveltime in minutes	Resistance in (KJ)	fuelconsumption in litre
41	≈7413	≈ 10,9

Compared to the simulation with constant speed of 150km/h, 1.4% of fuel was saved and the time only increased by one minute.

3 Results

The documented test series was carried out to explore the differences between driving in constant speed and adjusted to the tracks gradient in focus of the fuel consumption. This experiment was part of the authors bachelor thesis. The detailed results and the documented walkthrough were published by the University of Applied Science (Würzburg).

The simulations should show how synergies between physics and predictive analytics influnce the efficieny in public and freight traffic. The theory that an adaptive speed adjusture can lead to less fuel consumption was supported by the simulations, in theory.

Acknowlegements
I wish to thank thank Prof.Dr. Isabel John who assisted me with this thesis by words and deeds.

References

[1] Böge, Alfred. 1991. *Physik - Grundlagen, Versuche, Aufgaben, Lösungen.* Braunschweig/Wiesbaden : s.n., 1991. ISBN 3-528-54046-X.

[2] Callen, Herbert B. 1985. *Thermodynamics and an Introduction to Thermostatistics.* New York : s.n., 1985. ISBN 0-471-86256.

[3,4] Dobrinski/Krakau/Vogel. 1993. *Physik für Ingenieure.* Stuttgart : s.n., 1993. ISB 3-519-26501-X.

[5] Hans-Hermann Braess, Ulrich Seiffert. 2013. *Handbuch Kraftfahrzeugtechnik.* Braunschweig/Wiesbaden : s.n., 2013.

Wolfgang HEINZ
FHWS
Sanderheinrichsleitenweg 20
97074, Würzburg
GERMANY
E-mail: wolfgang.heinz@student.fhws.de

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Affinity Propagation as an algorithm for segmentation problems in business intelligence

Moritz Heusinger

Abstract

Affinity Propagation (AP) is a modern clustering algorithm proposed by Frey & Dueck with a wide application range due to often superior results, compared to classical approaches like k-means. AP is based on a probabilistic graph modeling, with an underlying message passing approach, where the number of clusters evolves during the propagation procedure and has not to be defined in advance. The paper provides a comparative analysis of AP in the field of Business Intelligence (BI), with respect to standard methods like k-means and hierarchical clustering.

1 Introduction

In [9] it is claimed that the existing amount of data which increase by a factor of ten by 2025. Beside of the volume, also the complexity of the data is growing such that automatic and flexible data managment techniques are needed. Machine learning provides flexible and effective methods where the parametrization can be optimized from training data such that manual interaction is minimized. The two major categories are supervised and unsupervised learning. In supervised learning the data $x_i \in \mathbf{R}^D$ are povided with additional label information e.g. $y_i \in \mathbf{N}$, where D is the input dimensionality of the data. The learning consists in optimizing a decision function $f(x_i, \alpha) = y_i \forall i \in \{1, \ldots, N\}$, with parameters α and N being the number of samples. In general the problem is relaxed by looking for an $f(x_i, \alpha)$ which is only minimizing a loss function, e.g. the number of errors $f(x_i, \alpha) \neq y_i$ is small. The other domain is unsupervised learning where no y_i is given and the modeling aims on finding structure in the data. A typical application of unsupervised learning is clustering, or in the context of BI segmentation, of the data. In clustering the data are summarized in so called clusters, such that the within cluster similarity is maximized, while the between clustering similarity is minimized [1]. A large number of clustering algorithms has been proposed which can are roughly divided into partioning, hierarchical and exemplar / prototype based clustering approaches. The presented work is focused on prototype based clustering, particular affinity propagation (AP) as proposed by Frey & Dueck in [3]. AP is based on a probabilistic graph model, where the parameter optimization (identifying the cluster centers) is done in a message passing framework. As an extra benefit, the number of cluster has not to be specified, but evolves in relation to the self-similarities as detailed below. An initial introduction into the function of the considered clustering algorithms, including AP is provided in Section 2. Further the consider datasets and evaluation measures are discussed. Section 3 shows the results of the various algorithms on a BI-scenario which is derived from the well known AdventureWorks database. In section 4 a summary of the results is provided and concluding remarks and future work is given.

2 Preliminary

Subsequently we provide a short introduction in AP, k-means and into agglomerative hierarchical clustering. Further the AdventureWorks database and the associated scenario is provided, which is used as a

benchmark dataset. Further a few cluster evaluation measures are provided as used in the experimental section.

2.1 Affinity Propagation

AP [3] constitutes an exemplar based clustering method, i.e. prototype locations are restricted to data points of the original dataset. Further, it deals with similarities $s(x_i, x_j)$ rather than dissimilarities, such as the Euclidean dot product, for example. Obviously, the quantization error can be formulated accordingly. Since prototypes are located at discrete positions, the quantization error can no longer be optimized by means of gradient techniques. Therefore, the quantization error is first rephrased as

$$S(I) = -\frac{1}{2}\sum_i s(x_i, x_{I(i)}) + \sum_i \delta_i(I) \qquad I(i) \,\epsilon\, \{1, \ldots, N\} \tag{1}$$

Note that there is no longer a fixed number of clusters given. Rather, cluster assignments are defined by means of an indicator function $I : \{1, \ldots, n\} \to \{1, \ldots, n\}$. Every data point picks a prototype by means of this function. Since prototypes are exemplars contained in the data set itself, this can be written as $x_I(i)$ for data point x_i . $\delta_i(I)$ punishes invalid assignments, i.e.

$$\delta_i(I) = \begin{cases} -\infty & \text{if } \exists\, i, j \text{ and } I(i) \neq i \text{ and } I(j) = i \\ 0 & \text{otherwise} \end{cases}$$

During training, these assignments have to be adapted; basically the data points have to negotiate to arrive at a valid assignment function which optimizes the cost function. The number of clusters is no longer specified a priori. A trivial valid solution of the cost function would be given by the identity $I(i) = i$, i.e. every data point forms an exemplar. To avoid this trivial solution, costs have to be introduced as soon as a data point becomes an exemplar. This can be achieved by adjusting the self-similarities $s(x_i, x_i)$, indicating the preferences of data point x_i becoming an exemplar. In the limit, for low preferences, only one cluster will be found, and for high preferences, every data point will be an exemplar. In between, different numbers of clusters can be reached. Typically, there are two different strategies to set the self-similarities appropriately: either they are set to a reasonable fixed value such as the median of the given similarities. Alternatively, binary search can take place until a desired number of clusters is reached.

In [3] the Equation (1) is modeled as a factor graph. This can be optimized by means of the max-sum algorithm. In turn, two kinds of messages between the data points are exchanged during the training, namely responsibilities.

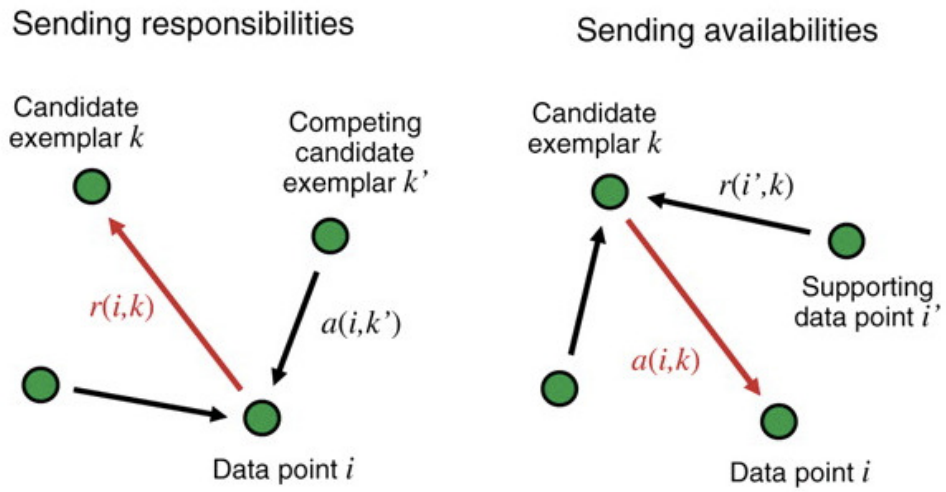


Fig. 1: Message exchange of responsibilities and availabilities (adapted image from [3])

Initially all availabilities are set to 0: $a_{ik} = 0$, subsequently responsibilities are calculated by:
Responsibilities-Update:

$$r_{ik} = s(x_i, x_k) - \max_{\substack{k'=1\dots N, \\ k' \neq k}} \{a_{ik'} + s(x_i, x'_k)\} \tag{2}$$

Availabilites-Update:

$$a_{ik} = \min\{0, r_{kk} + \sum_{\substack{i'=1\dots N, \\ i' \notin \{i,k\}}} \max\{0, r_{i'k}\}\} \tag{3}$$

The self-availabilities are updated by:

$$a_{kk} = \sum_{\substack{i'=1\dots N, \\ i' \neq k}} \max\{0, r_{i'k}\}$$

which can be combined at any stage of the training to decide the assignments

$$I(i) = \arg\max_k \{r_{ik} + a_{ik}\}$$

Since the factor graph is cyclic, there is no guarantee to obtain the global optimum or even convergence. For this reason, small random values are added to the similarities in every run to avoid cycles. This way, usually, convergence to a fixed point is observed. The algorithm stops the optimization if one of the following conditions is met [13]:

- a finite number of iterations (e.g. 500)
- the variation of the messages between two consequtive iteration is below some threshold (e.g difference is below $1e-4$)
- or the solution does not change between two iterations

A hierarchical formulation of AP was provided in [4] and additional optimization are suggested in [10].

2.2 k–means Clustering

K-mean is a partitioning and exemplar based clustering algorithm, where the number of cluster has to be specified in advance. K-mean aims on minimizing the quantization error [6]. Initially, the cluster centers are initialized randomly in the given data. Subsequently an alternating scheme of cluster assignment and cluster-center update is done until convergence is reached. In the first step (cluster assignment), each point x_j is assigned to its nearest cluster center μ_i:

$$j^* = arg\ min_{i=1}^{k} \{||x_j - \mu_i||^2\}$$

where j^* is the index of the closest cluster for a given point x_j. In the next step (center update), the cluster centers μ_i are updated by calculating the mean over the points x_j which are closest to the respective cluster center μ_i. Both steps alternate until convergence, which is guaranteed as long as the used dissimilarity measure is metric. The algorithms stops if: $\sum_{i=1}^{k} ||\mu_i^t - \mu_i^{t-1}||^2 \leq \epsilon$, where $\epsilon > 0$ and t is an iterator variable. The vector μ_i^t is a shortcut for the mean of cluster C_i in iteration t[12].

2.3 Agglomerative Hierarchical Clustering

Agglomerative, hierarchical clustering starts with having all x_i as individual clusters. The main step is to combine the most similar two clusters to form a new combined cluster. To evaluate the similarity between two clusters a variety of approaches have been proposed, like single linkage clustering, complete linkage

clustering and group average. They use the distance between two clusters by analyzing the (euclidean) distance between two points only:

$$d(x,y) = ||x-y||_2 = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

In the following we use complete linkage, where the maximum distance between a point in cluster C_i and a point in cluster C_j is calculated:

$$d(C_i, C_j) = \max\{d(x,y)|x \epsilon C_i, y \epsilon C_j\}$$

2.4 Cluster Evaluation

To access the quality of a clustering one typically aims on a high intra–cluster similarity and a small inter–cluster similarity [7]. This can be measured by a number of statistical and heuristical scores.

The R–Squared index is a measure which is frequently used for hierarchical clusterings, but is also applicable for other clustering algorithms. The R–squared index measures the dissimilarity between the obtained k clusters, or more precisely the homogenity between two groups [5]. The obtained quality estimates have values within 0 and 1, where 0 means that the clusters have maximum overlap and 1 indicates significant differences.

The silhouette coefficient measures compactness and separation of the clusters. Its based on the average difference to points in the next neighbored cluster. For each point x_i the silhouette coefficient s_i is calculated with values in $[-1, +1]$. A value of $+1$ indicates that x_i is much closer to points of his own cluster, than to points of other clusters. A value close to 0 indicates that x_i is on the border between two clusters. A value of -1 indicates that a point is closer to some other cluster than to its own cluster (wrong assignment).

C–Index evaluates the compactness of the clustering, by considering the N_{in} points which are closest to the k clusters, where N_{in} is the number of all intra cluster point pairings. The C–Index has values between $[0, 1]$. As smaller the C–index ist as better the clustering. A small value indicates compact clusters with rather small distances in the clusters and large distances between different clusters[12].

The **Runtime** of the clustering algorithm is also important, particular if a large number of samples is expected. In the field of business intelligence this is crucial because ad-hoc queries are common and fast calculations are needed.

2.5 AdventureWorks Data

In the experiment we use the well known Adventure Works Cycles database (version 2008) as provided by Microsoft. The adventure works data cover a fictional application scenario of a lare, multinational manufacturer of bicycles and related consumables. The company manufactures and sells various types of bikes on the north-american, european and asian market. The head quarter is located in Bothell, Washington with 290 employees. The company is organized in multiple regional sales agencies [2].

3 Affinity Propagation in business intelligence

To demonstrate the efficiency of AP in the context of BI and to compare it with the performance of k–means and the hierarchical clustering, a customer segmentation task is defined. By applying the various cluster algorithms we obtain different cluster solutions (clusterings) which are later on evaluated by the clustering evaluation measures, mentioned before.

3.1 Customer segmentation problem

Customer segmentation plays a key role in business intelligence projects and can be solved by a cluster analysis. The customer segmentation helps the company to get to know their potential customers and to

provide a better, targeted marketing. Using the results of the customer segmentation marketing events and offers can be optimized with respect to individual customer groups and to optimize sales performance indicators. Target groups can be differentiated by demographic, socio-economic and psychological properties as well as the shopping behaviour [8]. This scenario is also represented in the AdventureWorks database. To setup the experimental data the AdventureWorks database is filtered by a few relevant information provided in the associated datawarehouse table *ProspectiveBuyer*, indicating potential customers.

In this example we analyze 200 customers of california. Major points which are used to segment the data are provided in Table 1:

Dimension	Datatype	Description
YearlyIncome	Money	yearly customer income
BirthDate	Date	customer birthday
MaritalStatus	Char	customer married (M), or single (S)
Gender	Char	customer is female (F),customer is male (M)
TotalChildren	Integer	number of children of the customer
NumberChildrenAtHome	Integer	number of children at home
HouseOwnerFlag	Integer	customer owns a house (1), customer has no house (0)
NumberCarsOwned	Integer	number of cars owned by the customer

Table 1: Description of the prospective buyer data

3.2 Data preparation

Data are extracted using the SQL Server Management Studio and the AdventureWorks Datawarehouse-Schema. Within the extraction step elementary transformations and selections are done, such that only the first 200 results for california are used. The dimension YearlyIncome was converted from the datatype money to decimal. The age of the customer was calculated by analyzing the variable BirthDate. The extracted data are stored as CSV and imported into *RapidMiner* a data mining tool. Nominal attributes (marital status, gender) are encoded as numerical, binary attributes. Subsequently all data are normalized using the Z-transformation. The data are regularized using an outlier detection (provided in RapidMiner), eliminating 10/200 samples. The cleaned data are stored as CSV and further analyzed within *Matlab*.

3.3 Usage of the clustering algorithms

In the following the discussed clustering algorithms are applied on the pre-processed data and visualized using Principal Component Analysis (PCA).

The k-means algorithm generate in our scenario, two well separated clusters as shown in Figure 2. One can also see, that the clusters are particular clear to separate by considering the y-axis (principal component 2). Considering the loading of the PC-2 compnent we are able to identify the important variables such as MaritalStatus and HouseOwnerFlag.

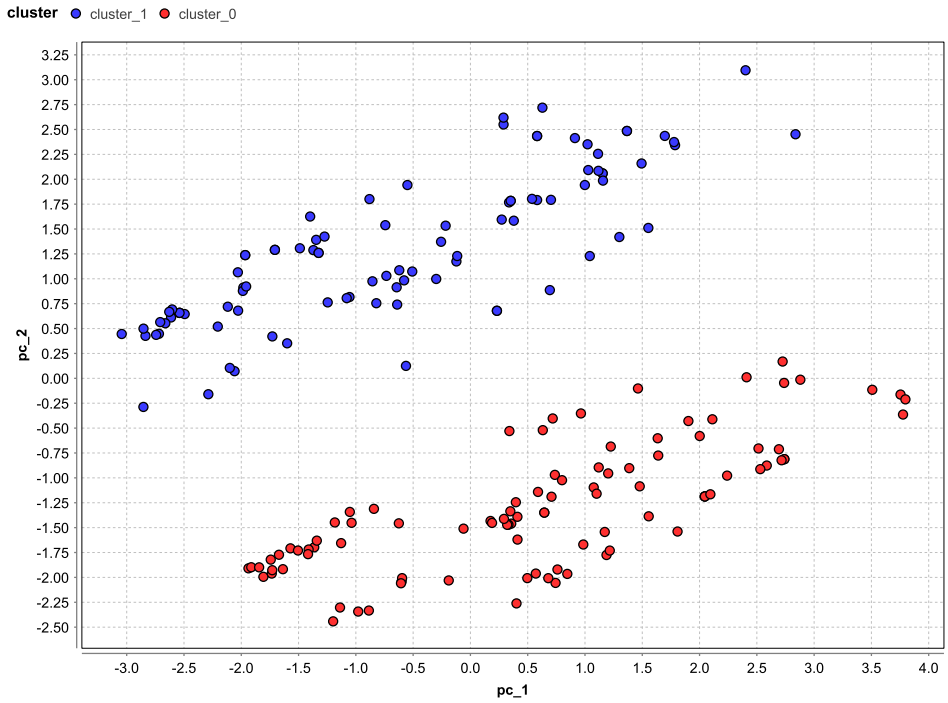


Fig. 2: Visualization of the k-means results using PCA

A detailed semantic analysis of the respective clusters is shown in Table 2. The results show that the attributes Marriage and HouseOwner are most disciminative. Both clusters are roughly balanced in the number of points.

	∅–Marriage	∅–# Woman	∅–yearly income	∅–age	∅–# children	∅–children at home	∅–# house owner	∅–# cars	cluster size n
Cluster-0	0%	56.25%	56,771$	50.64	1.71	0.79	53.13%	1.47	96
Cluster-1	100%	47.87%	45,319$	50.38	2.00	0.94	81.91%	1.45	94

Table 2: K-means results

In the next step we consider the obtained cluster in the context of a target group marketing task:

Cluster-0: Unmarried males and females at medium age and average income of $\approx$ 50.000$. The target group has in average two children, whereby one children is not living any longer at home, in general. House-owners and non house-owners are present equally. The clustered persons own at least one car.

Cluster-1: Married males and female at medium age, with an average income of $\approx$ 50.000$. The target group has in average two children, whereby one children is living at home. House-owners are dominating additionally owning at least one car.

The hierarchical agglomerative clustering generates clusters as depicted in Figure 3 using a PCA visualization. Cluster assignments are indicated by colors, clearly showing a separation based on the PC1 - axis. Following the PCA projection (which is not used in the clustering but to visualize the data) - we may assume that at least in parts some points are assigned to the wrong cluster.

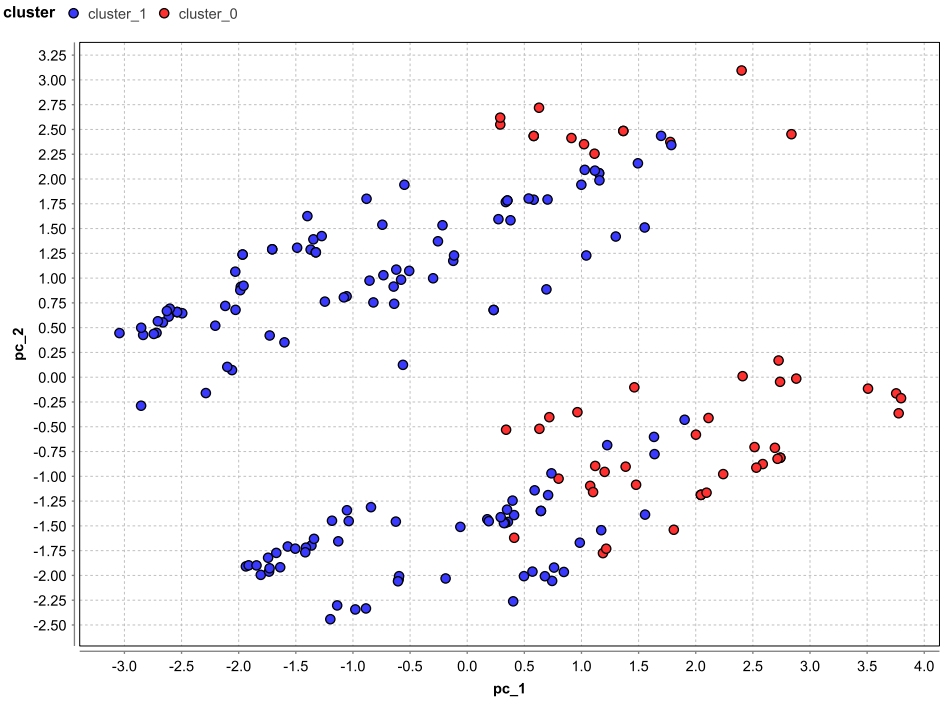


Fig. 3: Visualization of the hierarchical clustering using a PCA

The data statistic of the hierarchical clustering is shown in Table 3. The clustering is clearly determined by MaritalStatus, Gender, YearlyIncome, TotalChildren, NumberChildrenAtHome as well as HouseOwnerFlag. It is also obvious that the clusters are very imbalanced. Cluster-0 contains 39 samples, whereas Cluster-1 containts three-times more data (151 samples).

	∅–Married	∅–# Woman	∅–yearly income	∅–age	∅–# children	∅–children at home	∅–# house owner	∅–# cars	cluster size n
Cluster-0	25.49%	70.59%	77,059$	55.23	3.37	2.07	0.51%	2.47	39
Cluster-1	58.27%	45.32%	41,583$	48.78	1.30	0.42	73.38%	1.09	151

Table 3: Results of the hierarchical clustering

From the table we can draw two target groups which are very different from the target groups shown in Table 2:

Cluster-0: Mostly unmarried woman, with an income of over 75.000$ and an age of 55 years. In

general the persons have more than three children, with two living at home. House owners and persons without a house are equally present, each having more than two cars.

Cluster-1: Married and unmarried male and woman with an average income of 40.000$. The target group is barely 49 years old, with one, in general grown up, children. The target group members own a house and a car.

To apply Affinity Propagation on the given data a similarity matrix was calculated. To obtain comparable results the preferences of this similarity matrix are set such that two clusters are generated by AP. Each cluster is represented by a single exemplar.

The properties of the two exemplars are given in Table 4. Both clusters have roughly the same size.

	Married	Sex	yearly income	age	# children	children at home	house owner	# cars	cluster size n
Exemplar-0	Yes	M	50,000$	51	1	0	Yes	1	106
Exemplar-1	No	F	50,000$	61	2	1	No	2	84

Table 4: Results of AP

Using again the PCA we obtain a visualiztion of the cluster as shown in Figure 4). The clustering makes use of both principal components. Table 4 shows the properties of exemplar 1 and exemplar 2. The cluster differ in all variables, but YearlyIncome.

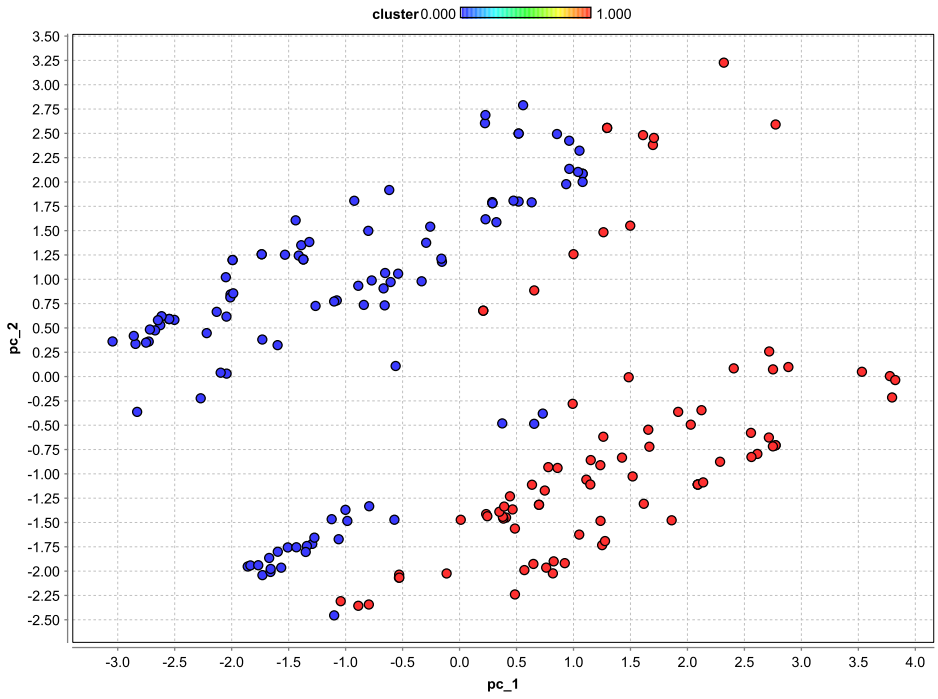


Fig. 4: Visualization of AP results using PCA

Using AP we can identify the two following prototypical customers within our target group marketing:

Exemplar-0: The prototypical customer is a 51 year old married man with an income of 50.000$. He has one children, which is not living at home. He owns a house and a car.

Exemplar-1: The prototypical customer is an unmarried 61 years old woman with an income of 50.000$. She has two children, with one living at home. She has no house but owns two cars.

3.4 Evaluation of clustering results

The clustering results are evaluated using the evaluation measures presented in Section 2.4. We make use of the Matlab Cluster Validity Analysis Platform (CVAP) Toolbox [11].

	Silhouette coefficient	R–Squared Index	C-Index	runtime
k-means	0,1073	0,5760	0,2809	20 ms
Hierarchical clustering	0,4258	0,8193	0,1768	44 ms
AffinityPropagation	0,1136	0,3361	0,2703	634 ms

Table 5: Evaluation of the data

Runtimes where calculated using a Lenovo Thinkpad X230T with an Intel Core i5-3320M (Dual Core 2,60 GHz) and 16 GB RAM. Table 5 shows the evaluation results of the three cluster algorithms. With respect to the tree measures, hierarchical clustering performed best. But hierarchical clustering provides very imbalanced clusterings with respect to AP and k-means. Considering AP and k-means, the evaluation measures show similar results. K-means is the fasted algorithm, followed by the hierarchical clustering and finally AP with $634ms$. The long runtime of AP is caused due to the quadratic complexity [10]. The target groups identified by the different clusterings differ to each other. AP and k-means cluster primary using MaritalStatus, HouseOwnerFlag and AgeIntYears, the income is of minor importance. The hierarchical clustering is focused on the yearly income. All three clustering algorithms define target groups which could be meaningful used by the marketing departments.

4 Conclusion

In this work we analyzed Affinity Propagation (AP) with respect to its efficiency in the context of business intelligence. AP has been compared to k-means and the agglomerative, hierarchical complete linkage clustering. In both considered application scenarios AP has shown competitive results to k-means. The hierarchical clustering on the other hand has shown to generate very imbalanced clusters. AP has the largest runtime complexity.
Finally we can concluded that AP shows beneficial results also in the BI context in comparison to k-means. For larger data sets AP is currently still a bit unattractive due the quadratic runtime complexity in the number of samples. Further k-means is very well known with many high efficient implementations in standard software whereas AP is not yet widely spread in data mining tools. While k-means generates centroids in the models, which may not be directly covered by the training data, AP is based on exemplars (e.g. a prototypical customer) such that the model becomes easier to interpret.
Future work should take runtime optimized approchaes of AP into account, e. g. like represented in [10]. By that the algorithm can be compared on an BI big data problem, which is very common in the business environment.

Acknowledgements

This work has been supported by Prof. Dr. Frank-Michael Schleif from the University of Applied Sciences Würzburg–Schweinfurt (FHWS).

References

[1] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.

[2] Microsoft Corporation. *Business cases of Adventure Works Cycles*. Available at: https://technet.microsoft.com/library/ms124825(v=sql.100).aspx, 2018. [Accessed 21-April-2018].

[3] B. J. Frey and D. Dueck. *Clustering by passing messages between data points. Science (New York, N.Y.)*, 315:972–976, 2007.

[4] Inmar E. Givoni, Clement Chung, and Brendan J. Frey. *Hierarchical Affinity Propagation. CoRR*, abs/1202.3722, 2012.

[5] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. *Clustering Validity Checking Methods: Part II. SIGMOD Rec.*, 31(3):19–27, September 2002.

[6] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer, New York, 2nd edition, 2008.

[7] C. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval.* Cambridge University Press, Cambridge, 2008.

[8] Sebastian Raschka. *Python Machine Learning.* Packt Publishing, Birmingham, 1st edition, 2015.

[9] D. Reinsel, J. Gantz, and J. Rydning. *Data Age 2025: The Evolution of Data to Life-Critical.* 2017.

[10] Leilei Sun, Chonghui Guo, Chuanren Liu, and Hui Xiong. *Fast affinity propagation clustering based on incomplete similarity matrix. Knowledge and Information Systems*, 51(3):941–963, Jun 2017.

[11] K. Wang, B. Wang, and L. Peng. *CVAP: Validation for Cluster Analyses. Data Science Journal*, 8:88–93, 2009.

[12] M. J. Zaki and W. Meira. *Data mining and analysis. Fundamental concepts and algorithms.* Cambridge Univ. Press, New York, 2014.

[13] Xibin Zhu. *Adaptive prototype-based dissimilarity learning.* PhD thesis, Bielefeld University, 2015.

Moritz HEUSINGER
Faculty of Computer Science and Business Information Systems
University of Applied Sciences Würzburg–Schweinfurt
Sanderheinrichsleitenweg 20, 97074 Würzburg
GERMANY
E-mail: *moritz.heusinger@gmail.com*

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Implementing a Local Grid-based System applied on E-Commerce web crawling

Elena Hinoveanu

Abstract

This paper presents the configuration process and challenges encountered in setting up a local low powered grid system for the purpose of crawling E-commerce websites. We will configure a local LAN network to support the communication between nodes and the main grid control centre. Multiple computing nodes with low hardware specifications are used collectively via low spec open source applications. Management engines provide a wide variety of alternatives to manual system monitoring. Sacrifice of automatic control leads to functionality deprivation and eventual inefficiency. Therefore, we will be using Puppet for secure information flow inside the structure. The cost of less control over system components as well as less functional options is balanced with the ability of using all resources simultaneously. Our efforts will show noteworthy results in the pursuit of a web crawler implementation designed to scrape products on distinct E-commerce websites. The experiment yielded an apparatus with the ability to receive several exclusive commercial websites, to parse their content in accordance to well-determined patterns and lastly notify users over available sale offers and their prices within the system. The overall structure demonstrates extensibility towards future demands and indicates an initiation in further similar operations. A future prospect for this project is the utilization of Hadoop within the cluster for big volumes of information to be downloaded, stored and processed accordingly.

1 Introduction

The 21st century has already diminished most of the widely accepted notions concerning the term scale. Current concepts surrounding proportions, size, metrics, numbers and ideas are intuitively shaped to offer a “narrative told in the context of business activities” [1]. Despite the diversity of human interest fields, the concern given to data collection and manipulation has been a never-ending process: with large numbers of individuals comes even greater responsibility in comprehending changes. Most importantly, “with the luxury of time and the ability to conduct data collection in an integrated and inductive manner, ethnographic research has become more fluid” [2].

Modern virtualization techniques stand as proof for the increasing need for “improving a large network's or data center's efficiency”[3] with the purpose of connecting the theoretical aspect of data handling to the its most practical usage. “The high demand of fully operational and efficient systems which focus on data mining”[4] maintains the connection between the two

previously highlighted demands. Consequently, there has been a considerable emphasis on building scalable software[1][5]. We will be presenting some of the widely mentioned options before describing our direct experience with the integration stage of the project.

The purpose of this project consists not only of exploiting the already present potential found in a cluster architecture but also to offer insight over the challenges faced during this task.

Our first objective was the implementation of a grid architecture consisting of four individual machines operated through Ubuntu 16.04 LTS[13], interconnected using four Local Area Network cables bound to a switch with the purpose of supplying the data transfer between the master and its subordinated entities. The logic behind the previously described structure assigns consistent roles to similar entities.

Our second objective was integrating one or more distribution technologies into the software architecture of the system and acquiring insight over the limitations along with possibilities of the industry regarding open source software. Research brought to our attention a wide array of options ranging from Metal as a Service (MAAS)[14], Canonical Openstack[15], Canonical Kubernets[16], Puppet[17], Hadoop[18], to more uncommon solutions as phpIPAM[19], Netbox[20] or Racktables[21].

Ultimately the purpose of the local grid system was the utilization of the network for a variety of practical assignments. We decided addressing an isolated issue concerning the applicability of web crawling techniques within the granted system. As previously stated, the leading focus of the paper is becoming accustomed to the educational potency of data collection.

The rest of this paper is structured as follows: section 2 presents the objectives of the experiment, section 3 presents our experience with various tools, section 4 briefly presents the development environment, section 5 involves our case study, section 6 concludes our experiment.

2 Related work

Metal as a Service (MAAS) intuitively captures the two most basic platforms on which founding a grid system is made possible at the time being: virtual and physical machines.[6] The convenient aspect of Maas is the number of services, included under one impressively thought software: a web UI, full API/CLI support, high availability, IPv6 support, open source IP address management (IPAM), Ubuntu, CentOS, Windows, RHEL and SUSE installation support, inventory of components, DHCP and DNS for other devices on the network, VLAN and fabric support , NTP for the entire infrastructure. [22] MAAS is also a flexible environment for fusing DevOps tools alike conjure-up, Juju, Puppet [7].

Holding a similar approach, Puppet grants the ability to manage a system infrastructure. Latter versions of Puppet include a considerable majority of the services required by the thriving industry: network management, version control, code distribution in conjunction with an ability of connecting systems regardless of the area. The exquisite aspect of Puppet stands in the swift configuration of the interdependent entities over a master-slave architecture. Given a business environment, multiple such architectures could potentially be linked to create an organization of clusters into the direction of cloud computing.

Separately, Hadoop is a software library developed by Apache with the purpose of distributing data among the components of a computing structure. Hadoop individually consists of four modules accomplishing different tasks: Hadoop Common, the primary provided utilities, Hadoop Distributed File System, the component monitoring the file access granted to client nodes,

Hadoop YARN, the task scheduler of the application, Hadoop MapReduce, the library which provides the parallelisation to the structural entity.

We will proceed to briefly describe the concept of a web crawler and how it harmonizes with the global purpose of the paper. The concept of a web crawler is overall focused on web page indexing. Furthemore, the web crawler receives several URLs known as seeds serving as starting points to the indexing process. Each seed is to be considered a nest of hyperlinks that provide continuity to the crawling task. Pages and the contained HTML code constitute the data to be stored and subsequently interpreted. They are commonly found in the structure of search engines, "systems that assemble a corpus of web pages, index them, and allow users to issue queries"[11]. Crawlers currently are in a ceaseless development process where the central target is eliminating possible visit redundancy.

We have taken into consideration the advantages of implementing an web crawler for both educational and experimental purposes. Firstly, utilizing a cluster structure ensures crawling speed achieved in real-time. Considering a highly concurrent environment, multiple crawlers provide the suitable context for timely data collection. Secondly,parallelization is ensured using a management tool which additionally secures the execution of code sections throughout runtime. A parallel environment in the case of a web crawler requires that each unique client node executes the code on one website address allotted before runtime. Conceptually, our project also associates a multi thread approach to the individual client-side application following the structure of the crawler4j[23] library. However, scarce system resources led to an increase in time consumption thus we preferred a single thread approach in favor of efficiency. The benefits of a single task assignation system consist in "maximination of download rate while minimizing the overhead from parallelization"[12]. Therefore, a list containing website addresses can be efficiently atomized into shorter tasks which are run concurrently with respect to high velocity rate.

"The distributed aspect of the application is advantageous to a centralized approach on counts of scalability, efficiency and throughput. A fundamental pseudocode version concerning crawling algorithms is:

```
Crawler ( )
 Do Forever
 Begin
 Read a URL from the set of seed URL's
 Determine the IP-address for the Host name
 Download the Robot.txt file, which carries download information and also includes the files to be excluded by the crawler
 Determine the protocol of underlying Host like HTTP, FTP, GOPHER
 Based on this protocol, download the document
 Check whether the document has already been downloaded or not
 If the document is a fresh one,
 Then store it and extract the links and references to other sides from that document
 Else
 Abandon the document
 End" [10]
```

3 Setup and configuration

The main activities of the project include grid configuration alike the installation of the Operating System on subordinated machines through network boot, configuring the Local Area Network Internet Protocol addresses accordingly, selecting the proper cluster management tool, implementing the web crawling system. Our initial choice for a cluster management tool was Metal as a Service provided by Canonical Ltd.[24] since it integrated a visualization platform, machine testing, automatic detection, integration of detected nodes. Moreover, it currently holds

an outstanding record for the professional quality of provided services amongst other available options.

The first activity incorporated installing Ubuntu Server 16.04 on the main computer along with configuring netboot on the remaining machines. While the task appeared to be achievable at first, it was soon brought to our attention that performance had been negatively affected by unchecked system requirements. Therefore the next step evolved into adding more RAM to the head computer for superior and reliable performance. Each computer holds a network card with two ports for Internet and LAN connection however, due to lack of resources we concluded that using only one port and setting a shared connection on the head computer would be sufficient to achieve the target of the project.

Following the hardware requirements stage, we were able to install the Operating System as planned and adequately modify the BIOS settings to PXE boot for MAAS detection. The next step involved installing MAAS on the head computer: adding the MAAS repository to the Ubuntu repositories, installing the MAAS application, configuring the network addresses for proper MAC address detection when enlisting nodes. The MAAS web interface was made accessible following the installation, at the address <<ipaddress>>/MAAS where we were given the opportunity to interact with the variety of services provided through MAAS.

The upcoming step involved configuring the DHCP server of MAAS to support immediate node identification as well as providing valid IP addresses within a specific range. For fast node login, MAAS provides a SSH login option which later grants the ability of logging in without the need of an already-set password.

Once restarted, every node instance was detected and furthermore enlisted under a default name in the web Graphical User Interface. The next challenge we confronted was the lack of a functional power management system. Earlier versions of MAAS supported manual powering since absolute control over nodes had not been considered of critical importance. However, following MAAS version 2.0, manual powering has been included under the condition of a temporary option rendering the cluster unmanageable according to latter specifications. Moreover, research provided that following the structure of the provided cluster nodes which included an outdated motherboard, the ability to automatically power components would not be an option regardless of any further modifications. Further attempts at integrating Wake-On-LAN or Etherwake to MAAS 2.3 proved to be inefficient since both options had been removed with the release of MAAS 2.0. we had attempted to resort to a community tool however the new web API failed to respond to the undertaken adjustments.

Inquest provided that with earlier versions of MAAS, vital functionalities the environment had been lacking could be restored. Rolling back to a previous release ought to have provided the missing functionalities and downsize requirements. However, the reality of the situation soon revealed the absence of any installation kit previous to MAAS 2.0 rendering the effort unsuccessful. Following the progression of business requirements, virtualization has only been made available to local, single-machine architectures. The project ought to have been composed of seven devices, four operating on the following requirements:

- A region controller (including PostgreSQL) is installed on one host: 4.5 GB memory, 4.5 GHz CPU, and 45 GB of disk space.
- A region controller (including PostgreSQL) is duplicated on a second host: 4.5 GB memory, 4.5 GHz CPU, and 45 GB of disk space.
- A rack controller is installed on a third host: 2.5 GB memory, 2.5 GHz CPU, and 40 GB of disk space.
- A rack controller is duplicated on a fourth host: 2.5 GB memory, 2.5 GHz CPU, and 40 GB of disk space.

The logic behind the above infrastructure constructs a web of individual controllers with complementary functionalities designated for directing any data flux between points, provide local and cluster management, collect and accumulate data with reference to node activities. A region controller is responsible with providing high-bandwidth services to other components, it ensures the proper communication between other upper-level components. A rack controller ensures the availability of data among region controllers, providing the version management functionality of stored files. A node describes a broader concept introducing either sequence of devices, machines and controllers.

Consequently, configuring a single rack controller in the absence of a secondary VLAN managed through an additional rack controller does not meet the minimum requirements for a consistent implementation. Starter projects involving limited resources have been devalued in favor of production environments with a certain funding opening.

Another peculiar perspective of the aforementioned tool consisted of the deployment facility offered through Juju, an application with the purpose of scaling and managing stacks. Juju provides the ability to deploy applications on public and private clouds, create and shape cluster environments. However, in order to efficiently integrate Juju[25] into a MAAS environment, the execution of spell-based scripts is mandatory during three phases of the deployment process: after Juju bootstrap, just before a Juju deploy, after a Juju deploy.

As a consequence of the difficulties encountered in previous endeavours, we faced the challenge of adjusting the management tool and consequently our previous thinking style. Therefore we were required to address the hardware requirements problem at hand. After prolonged research, we encountered the second tool which concerns itself with the integration of both monolithic and large environments: Puppet 4.0[26].

When initially discovering Puppet, we attempted at experimenting with the newest release, a version which focuses on integrating the demands of the field and those of the development teams, Puppet Enterprise. Software - user communication is easily achieved through Ruby DSL which ensures the efficient description of systems through a client - server architecture. “Each of the different components in an infrastructure: SSL CA, Reporting, Storeconfig or Catalog compilation can be split into its own server ”.[9] Nonetheless, Puppet Enterprise 2017 represents an ongrowing vision of the provider company and its members, being aimed at solving the issues of previous tools and tackling the requirements of modern devices. Figure 1 presents the hardware expectation threshold which the development team recommends for any server device. Our server machine lacked the prerequisites necessary for proper uninterrupted performance.

Node volume	Cores	RAM	/opt/	/var/	EC2
10 or fewer	2	6 GB	20 GB	—	m3.large instance

Fig. 1 - Table containing Puppet requirements

The obstacle continued to be the amount of requested RAM. As a result of time pressure and limitations, the alternative of choice remained to downgrade tool versions. We settled for an open source release provided under a low resource cost prospect. Following the installation procedure alongside additional configuration routines, we were able to purposefully use the previous network outline for the attainment of a entirely functional web of services. Alike the vast majority of similar software, Puppet 4.0 operates with two basic concepts which strongly follow the client - server distributed application model so to achieve low coupling as well as to properly separate functional tasks within the managed system: one or more devices running the Puppet master application whose route-request interconnected services include node identification through signature-based certificates, control of client-level requests, implementation of subsequent structural revisions, also at least one device supporting the Puppet Agent application which

combine informational exchange between distinct application layers with comprehending the response through properly defined protocols. In order to achieve our goal without further environmental limitations we have decided not to obscure the structural perspective through the addition of further data storage services such as Puppet DB.

The greater and more time-consuming task in our project was learning how to utilize the crawler4j Java [27] library. After all, a web crawler only made use of two classes: a controller which guides the activities throughout the application and the crawler which had a specific set of rules to keep track of once crawling web pages. The main target of our application is to make use of every functional node, assign it an E-commerce web page and in return receive an email with a link towards a product of a specific type and its price. During the implementation phase, we also made use of the JSOUP[3] library for the parsing requirement of the project. After crawling a web page, the HTML code will be stored and parsed for the price element with its content further sent through mail.jar[4] to an email address. Thus far, the aforementioned task has been properly completed.

The next task involved compiling the source code, creating a .JAR file and further attaching the program to the Puppet architecture. After successfully accomplishing all of the previously mentioned subtasks, we transferred the Java executable on every node at a similar location each and further delved into gaining knowledge over Puppet environment coding.

Besides its own services and capabilities, Puppet provides the ingenuity necessary to program a wide variety of tasks. To properly complete the task at hand, we included and interacted with the following notions: Puppet manifests, commands, privileges. Therefore, we wrote a manifest instructing every node to utilize the standard Java execution command "java MyProgram.jar [parameters]". The parameters section, in the considered case, involved a crawlable E-commerce web site. The mentioned task has been thoroughly completed including setting user privileges, ensuring the presence of a Java JRE installation.

In order to add a wider perspective to the project, we have settled on the implementation of a Hadoop multi node setup purposefully for the exploration of "map and reduce functions found within the information retrieval programing paradigm" [8] which in our case is provided through MapReduce. The future prospects of the project involve creating a functional Hadoop environment, the implementation of a folder tracking task and possibly utilize the functionalities at hand for data manipulation purposes.

We have succeeded in enabling SSH authentication between each one of the nodes but more importantly between the master node and the agent nodes, creating a separate administrator account for Hadoop. The currently undergone task represents experimenting with Hadoop and efficiently programming a folder tracking job which monitors the changes applied to a folder and reports it to the master device. We have already researched a few methods of completing the mentioned goal and the most interesting solution is using the *DFSInotifyEventInputStream* class which notifies the HDFS administrator over the indicated future events.

4 Development Environment

The cluster has been deployed using Ubuntu 16.04 LTS, a Debian architecture. Given its large-scale use and the benefits of a Linux kernel including multi user management, Ubuntu has been the main option for a multitude of tasks to be performed under distinct users, therefore, the most suitable choice for the evolution of the project.

3 JSOUP official download website https://jsoup.org/
4 Javax mailing API https://mvnrepository.com/artifact/javax.mail

We have used the Java development environment due to the type of undergone activities as well as the availability of public libraries designated for crawling and content parsing purposes. The project has well benefited off the object oriented language, separating responsibilities within classes for a consistent application. We have decided to integrate the JSOUP parser libraries for rapid HTML crossing and element selection. Java has also been proven useful in both program compilation and execution phases.
Puppet occupies a great portion of the project providing the declarative Unix-like system management. Puppet follows the agent-server architecture and introduces its own configuration language which requires every resource to present a title, a type, a list of configurable attributes.
Hadoop represents a framework used for distributed storage and processing sets of big data through a MapReduce programming model. Hadoop offers an alternative to the most common management tools and it was built on the assumption that hardware failure should be handled through the framework itself, therefore it offers an already incorporated error handling feature.

5 Case study

The proposed case scenario revolves around the Emag glitch where, during a sales campaign, more products which had been subjected to substantial price reductions were listed as having a 1 euro price tag. The most expensive item, earlier listed under a 2401 € tag, had been sold at a 1.2 € threshold and the Emag representatives announced that in spite of the erroneous system, every order would be honored. Consequently, such glitches fall into two contrasting perspectives: one in which the provider company experiences substantial losses both in trustworthiness and profit, another where such incidents provide a foundation for future ideas. Our case study proves that supervision of related cost incidents is attainable through theoretical and practical integration of feasible distributed concepts.

In order to completely utilize the cluster processing power we have assigned one E-commerce website to each individual node. The main program has been written in such manner that URLs are passed in an argument form to the Java executable therefore providing reusability and flexibility. Since user - client communication is achievable through the server node, we have created a Puppet manifest including the exec resource which customizes tasks for distinct nodes. We sent Emag, Amazon and Media Galaxy addresses as arguments to the java - MyCrawler.jar command. In the parsing section of the source code, JSOUP identifies the HTML area containing the price and sale percentage tags, gathers the containing integer numbers. These values are directly exploited during email composition as the containing message for the desired notifications.

On the part of better output clarity, we have assigned one thread to distinct crawling tasks in spite of its multithread support. A notable aspect of Puppet resources revolves around their illustrative nature, they can freely picture the responsibilities within an architecture and furthermore respect the structural model. The following phase resided in the application of the formerly created manifest to client nodes that we achieved through the puppet apply - crawler.pp command. Each node downloaded the indicated file, evaluated its content and lastly interpreted the assigned instructions. Following the execution of the .jar file, we were announced about the entire sales offer related to laptops related to different websites. Establishing which offer is noteworthy or outrageously advantageous is beyond the scope of this paper. For the time being, all offers are considered note-worthy and therefore a notification is issued. The results consisted of subsequent emails regarding product sales, sent at a 0.00001 ms interval which may vary depending on the crawling rate, the products which actually contained a promotion price and their frequency rate

Fig. 2 - Figure containing an example email

The structure of an email is presented in Figure 2, where an user can see the previous price and the subsequent current price to which the reduction has been applied. Therefore, we can see the parsing process attached to to the experiment through the usage of HTML labels extracted from the crawling process.

.

6 Conclusion

The encounter with numerous obstacles during the initial phases of the project demonstrated the system requirements barrier imposed under the aegis of enterprise influence. Moreover, local grid arrangements represent a possibility, but only in the context of proper resources within imposed limits. The demands on the server devices greatly vary between deployments considering size, complexity, responsibility and data structure. As seen in the case of MAAS where a single rack controller instance lacked the ability to coordinate activities within the structure, certain applications demand specific arrangement patterns. Other more recent distributions require distinct specifications suited for associations consequently sacrificing the educational potential in the experimental proportion.

Furthermore, the balancing of low hardware specifications has been achieved at the cost of less control over system components as well as less functional options. Management engines provide a wide variety of alternatives to manual system monitoring. Sacrifice of automatic control leads to functionality deprivation and eventual inefficiency. This is detrimental to an already resource scarce grid system. Substitution continues to represent a viable preference to the aforementioned issue nonetheless it creates deficient entities with lack of official support. On the other hand, a part of available software is supported through populous communities which have confronted as well as consequently created solutions to previously mentioned interferences.

Our efforts showed noteworthy results in the pursuit of a web crawler establishment designed to crawl products on distinct E-commerce websites. The experiment yielded an apparatus with the ability to receive several exclusive commercial websites, to parse their content in accordance to well-determined patterns and lastly notify users over available sale offers and their prices within the system. The overall structure demonstrates extensibility towards future demands and indicates an initiation in further similar operations.

Acknowledgement. This work was mentored by Sorin Ilie, Ph.D. Eng. from the University of Craiova, the Computers and Information Technology Department.

References

[1] G. Hardy, D. Everett, *Shaping the Future of Business Education: Relevance, Rigor, and Life Preparation*, 2013, ISBN: 978-1-3494-4148-8
[2] Greg Guest, Emily E. Namey, Marilyn L. Mitchell, *Collecting Qualitative Data*, 2012, ISBN: 978-1-4833-0714-5
[3] Fernando Rodríguez-Haro, Felix Freitag, *A summary of virtualization techniques, The 2012 Iberoamerican Conference on Electronics Engineering and Computer Science*, 2012, ISBN: 978-1-6274-8523-4
[4] Chowdhury, N.M. Mosharaf Kabir, Boutaba, Raouf, *A survey of network virtualization*, 2010 ISBN: 978-1-4503-0199-2

[5] Sourabh Sharma, David Gonzalez *Microservices: Building Scalable Software*,2016, ISBN: 978-1-7872-8098-4
[6] Das, Prashanta Kumar, Deka, Ganesh Chandra,*Design and Use of Virtualization Technology in Cloud Computing*, 2017, ISBN: 978-1-5225-3602-4
[7] Tom Haddon, "*Managing service orchestration with Juju and Mojo*", 2015, ISBN: 978-1-3494-4148-8
[8] Srinath Perera, *Hadoop MapReduce Cookbook*, 2013, ISBN:978-1-8495-1729-4
[9] Thomas Uphill, *Mastering Puppet*, 2014, ISBN: 978-1-7858-8810-6
[10] Krishna Kant Singh, Akansha Singh, *Faster and Efficient Web Crawling with Parallel Migrating Web Crawler*, 2010, ISSN: 1694-0784
[11] Christopher Olston, Marc Najork *Foundations and Trends: Web Crawling*", 2010, ISBN: 978-1- 6019-8322-0
[12] A.K.Sharma, Shruti Sharma, J.P.Gupta, *A Novel Architecture of a Parallel Web Crawler*, 2011, ISSN: 0975 – 8887
[3] Ubuntu 16.04 LTS download address http://www.ubuntu.com/download/desktop
[4] Metal as a Service official website http://MAAS.io/
[5] Openstack documentation providing an overview of the software http://docs.openstack.org/queens/
[6] Canonical Kubernets download and details website http://www.ubuntu.com/kubernetes
[7] Puppet description and download location http://puppet.com/download-open-source-puppet
[18] Hadoop official location providing information, releases and further documents http://hadoop.apache.org/
[19] phpIPAM official website http://phpipam.net/
[20] NetBox official website https://netbox.readthedocs.io/en/latest/installation/netbox/
[21] Racktables information and download website http://www.racktables.org/
[22] A descriptive version of MAAS functionalities and instructions, https://docs.maas.io/2.1/en/
[23] Crawler4j download location and instructions https://github.com/yasserg/crawler4j
[24] Canonical official website and documentation http://www.canonical.com/
[25] Juju official website containing bundles, charms and the newest releases https://jujucharms.com/
[26] Puppet installation tutorial and download website http://puppet.com/docs/puppetserver/2.7/install_from_packages.html
[27] Oracle Java official website http://www.java.com/en/

Elena HINOVEANU
Faculty of Automation, Computer Science and Electronics, University of Craiova
Blvd. Decebal nr. 107, Craiova
ROMANIA
E-mail: elenamhinoveanu@gmail.com

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Treasureland – 2D platform game

Iskren Ivanov

Abstract

Treasureland is a single player 2D platform game. It is based on Unity Engine using C#. Everyone likes playing games in the free time. The games are a good way to relieve stress or just to play them for fun. Treasureland is generally made for children but adults can also play it as well.

1 Introduction

There are a lot of benefits from playing games. The games are made for multiple purposes. They improve player's skills: concentration, reflexes, thoughts, adapting skills, etc. Treasureland is a small game which doesn't take so much performance and doesn't require a good computer. It is a simple game and it is very easy to be played. In Figure 1, the main menu of the game is presented. The final goal of the author of Treasureland was to create a 2D platform game based on Unity which includes character controlling and interacting with objects. These kind of games are favourite to many people. There is a large amount of 2D platform games and people still want more because they love them. These type of games are one of the oldest types but they are still up to date. Some people play them because of nostalgia, others are looking for new features and so on. There are lots of examples for games like Treasureland and Super Mario is the most famous one. After a Unity game is being created, the most of its objects can be re-used in future game projects. Treasureland was created for learning purposes – the game engines' concept, computer 2D graphics and computer programming.

Creating games is a good idea to get more into computer graphics and by the time to earn professional skills. This is not only because of learning skills but it is also fun doing it. It motivates the developer to continue to learn and discover new techniques, mechanics and technologies for game development.

2 How the game works

2.1 Main functionality

Treasureland's advantage consists in its simplicity. There is one character that the player can control. The goal of the game is collecting treasures, overcoming obstacles and reaching the final checkpoint. There are only three levels but the engine allows easily creating additional levels by just dragging objects into the scene. The player is able to move around as like as he can jump or shoot. The game can be played either by a keyboard and a mouse or by a joystick.

Figure 1 Main menu

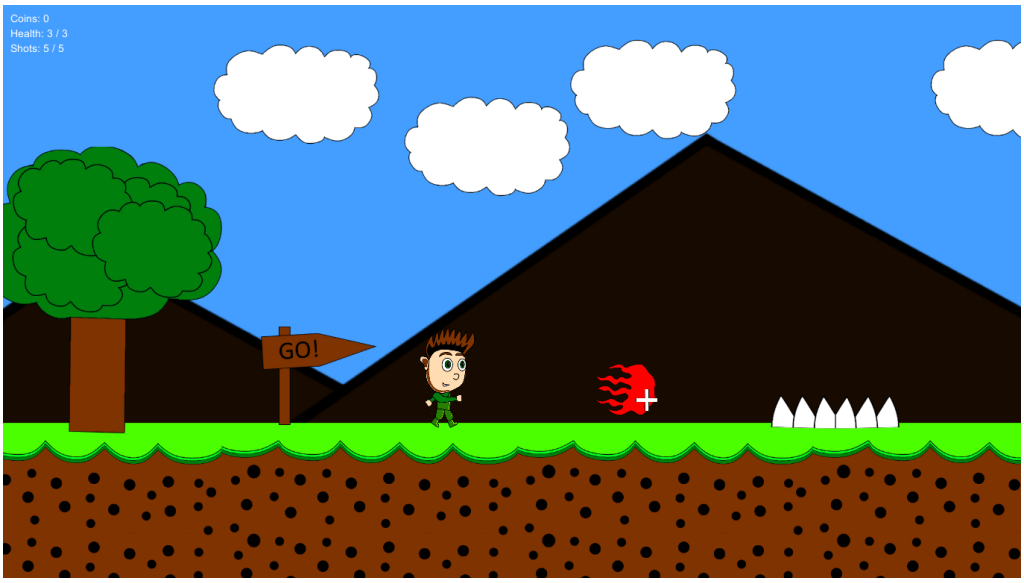


Figure 2 Level 1

2.1.1 Starting the game

Once the game starts, there are three messages on the left corner showing the collected coins/treasures (starting with zero), player's health and the remaining shots. There are 3 health points by default. Each time the player collects a treasure the message is being updated to the new amount. There are health objects which can be collected for additional health points. There is also another message for the shots. The player has 5 shots by the game start. He can destroy some obstacles by shooting at them. When the number of the shots end up at 0, the player will not be able to shoot anymore until he collects more shots. The shots can be found on the terrain. The player cannot have more than 3 health points and more than 5 shots but there are items which are increasing the maximum count.

Figure 3 Level 2

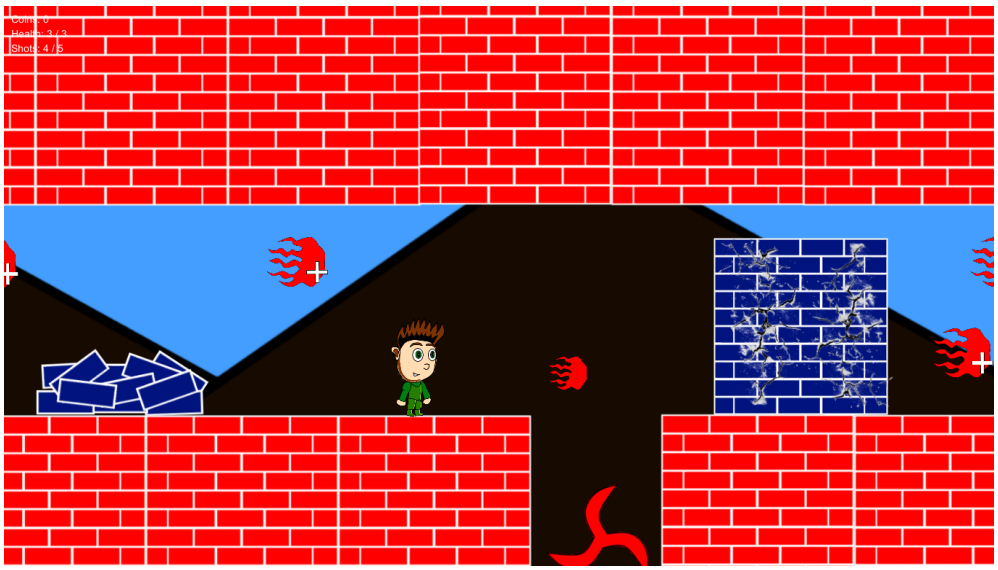


Figure 4 Level 3

2.1.2 Levels

There are three images from the game in Figure 2, Figure 3 and Figure 4. Level 1 is an introduction level. At level 2, the atmosphere is more different. It is night and it is harder to see the treasures and the obstacles. This makes the game harder and more interesting. At level 3, the character is placed somewhere else and he has to make a way himself through the walls by breaking them using shots.

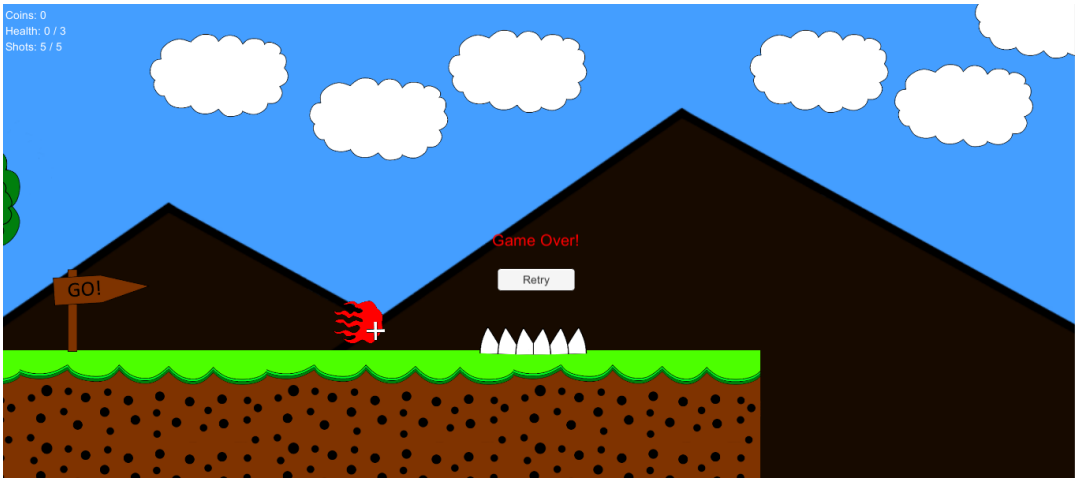


Figure 5 Game Over Dialog

2.1.3 Game Over

When the player interacts with an obstacle, he loses one health point. When the health points are zero, the player dies and a dialog about restarting the game appears (Figure 5). Another way, leading to game over is by jumping off from an edge.

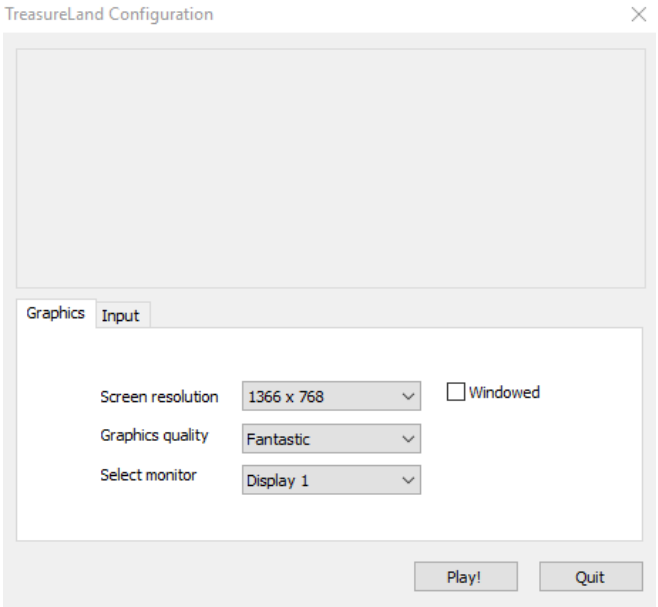


Figure 6 Configuration's window

2.1.4 Main menu and Pause menu

By starting the game, the player will be located at the main menu where he can choose to start a new game from a certain level or to quit the game. If he chooses to play, he starts from the starting checkpoint. By pressing Esc key he will be led to the Pause menu (Figure 7) where the game is in pause mode. Furthermore, he can choose to resume the game, go on the main menu or to quit the game.

2.1.5 Configurations before the game start

Unity's framework provides a configuration window where the player can manipulate with, as shown in Figure 6. The player is able to configure the graphics quality, screen resolution, output devices, controls, etc.

Figure 7 Pause menu

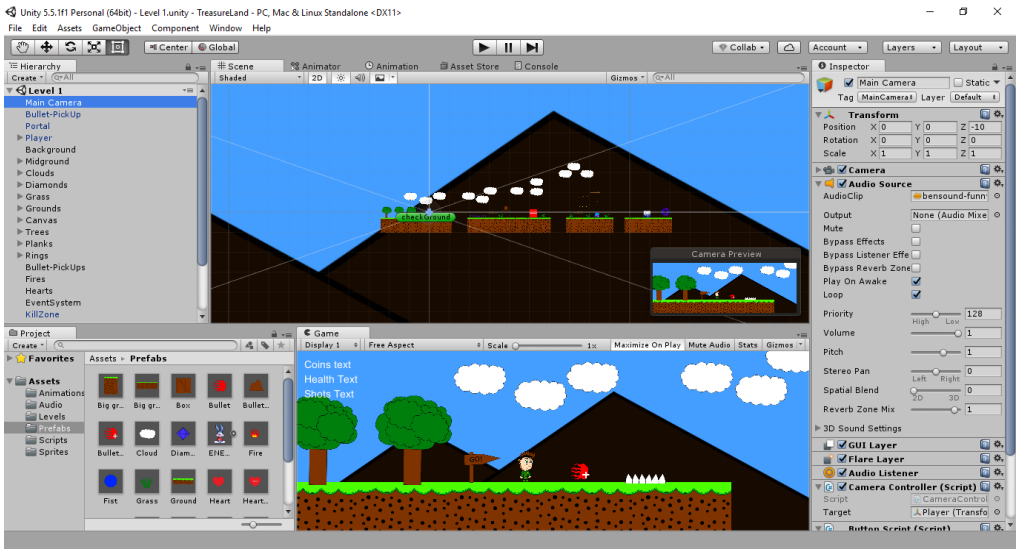


Figure 8 Unity's framework

2.2 Usage of Unity and C#

Building a game on Unity [1] is quite easy but the ideas for it are taking the most time. Creating a game requires drawing textures on any chosen graphic editor like Paint.NET, Photoshop or anything alike, dragging the textures to the scene, configuring their settings and writing a certain script for them by using C# or Javascript programming language. When a texture is added, it will not have functionality by default. It will be part of the terrain, but it will not have gravity or interaction properties. Therefore, components should be added to the texture (Collider [2], Rigidbody [3], etc.). After that, it is possible to save the configured texture as an object that could be placed many times on the scene (prefab). This makes adding new levels very easy. To add additional functionality which is not provided by Unity, it has to be arranged in a script. After that, the script is attached to an object which is going to have its functionality. An object could have several scripts. Images from Unity's Engine are presented in Figure 8 and Figure 9. In Figure 8, the game engine's view can be seen. This is the place where the game is being created. On the left side, there are all objects which were added to the scene. On the right side, there are all the configurations for the selected object (in this case – Main Camera). In the center is the scene. At the bottom the game view is shown. In Figure 9, the game mode is presented on full screen. The play button above is pressed and the game is running. By pressing these buttons the game could start, stop or pause to configure something.

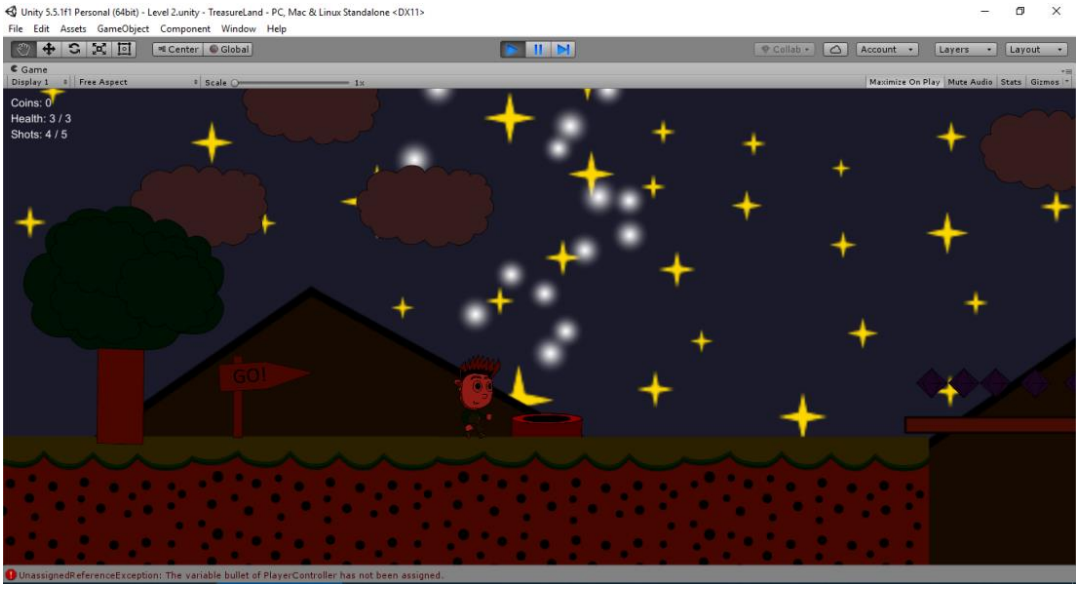


Figure 9 Play mode in Unity

3 Conclusion

There are a lot of technologies for creating games and Unity is one of the best. It provides everything needed for making any kind of game. Some features are more difficult to implement compared to other game engines, but there is always a way to do anything in Unity.

The author of Treasureland plans to expand the capabilities by: adding new levels, adding new functionality, adding new items that can be picked up, adding new character skins, adding new NPCs [4] and more.

Acknowledgement: This work was supervised by Professor *Katalina Grigorova* from *University of Ruse*.

References

[1] Unity Technologies (2018). Unity3D. [online]. available at https://unity3d.com/ [accessed 1 June 2018]
[2] Collider. [online]. available at https://docs.unity3d.com/Manual/CollidersOverview.html [accessed 1 June 2018]
[3] Rigidbody. [online]. available at https://docs.unity3d.com/Manual/RigidbodiesOverview.html [accessed 1 June 2018]
[4] NPC. [online]. available at https://en.wikipedia.org/wiki/Non-player_character [accessed 1 June 2018]

Iskren IVANOV
University of Ruse
Department of Informatics and Information Technologies
8, Studentska str., Ruse 7017
BULGARIA
iskren_damqnov@abv.bg

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

2D Videogame, turn based strategy in Unity

Mihail Juravlea

Abstract

Different types of computer games, like action, strategy or adventure are consider really challenging for the human brain. The aim of the present article is to propose and describe a new strategy game named “Climate war”. The game involves strategically decisions, attention, resources management, scheduling skills, objects management and learning skills about geography and fauna of the world. As "Climate War" is a strategy turn based game, it offers a real map of the world, the possibility of creating populations and to manage their resources. Among the important technical features of this game are: saving progress capabilities and low memory and CPU overhead, thanks to Unity game engine.

1 Introduction

Psychology in Video Games. The reasons why people are playing video games are numerous: to escape from reality, to learn some English, to make new friends and socialize with them, to enjoy the story from a game, but the aim is the same, to enjoy a game.

Nick Yee, a scientist in Cyber-Psychology and Behaviour, discover 3 motivations for people to play video-games:

1) Achievement - includes the want to advance in the game, the interest in the rules and system of the game, and the want to compete with others
2) Social - is the want to form connections with others, liking to chat and help other players, and the want to be a part of a group effort
3) Immersion - is creating and customizing a character to play in the unique story of the game as well as wanting to an escape from real-life

Another motivational factor from Richard Ryan and Edward Deci is self-determination, is a theory which states that human behaviour is driven by the need for competence, autonomy, and relatedness.

“If those needs are met, we enjoy activities more, even in gaming. Competence is fulfilled in games through flow, which is a completely focused mental state, and is used to explain the enjoyment of games. Flow is met in games when a game challenges the player enough so that it is still interesting and makes gamer want to continue to play, but not too hard where it is not fun anymore or too easy where it becomes boring. Autonomy is met because we have control in games and we voluntarily play games. Lastly, relatedness is fulfilled because we are able to interact and compete with others in a way and world that’s different from real life.

A study done at National Dong Hwa University, found that the satisfaction and flow experience plays a major role in whether one intends to continue to play a game. As mentioned before, factors that can relate to flow and enjoyment include the game story, graphics, and control. Those all contribute to continuance of playing. The social part of gaming is also very important. Especially in multiplayer and online games, if one has achieved a lot, gained recognition, or has made many connections with players, those play a significant roles in if one will continue to play a game

Other studies have researched the features of games that are attractive to players. One study found many features such as the enjoyment and relaxation gained from gaming, having no constraints in games like in one may have in real life, the artistry of the game, and the interactivity and competitiveness of a game.

Finally, let's point out some positive effects of gaming. Research has shown that as opposed to non-gamers, gamers had better visuo-motor coordination, spatial representation, iconic skill, and visual attention. Furthermore, recently there's been a change in focus in the gaming world, where people have had an interest in using games for a purpose other than just entertainment. They are called serious games, and the purposes are related to helping with problems in fields such as health care, education, and more."[1]

Turn Based Strategy. "A turn-based strategy (TBS) game is a strategy game (usually some type of war-game, especially a strategic-level war-game) where players take turns when playing. This is distinguished from real time strategy, in which all players play simultaneously."[2]

Examples of Turn Based Strategy are: Board Games, Turn Based Tactics, or Mainstream computer games. For my game is more like to be a Turn Based Tactics.

Turn Based Tactics. "Turn-based tactical game-play is characterized by the expectation of players to complete their tasks by using the combat forces provided to them, and usually by the provision of a realistic (or at least believable) representation of military tactics and operations. Tactical role-playing games are a part of this genre. Examples include Fire Emblem, The Battle for Wesnoth, Poxnora, Silent Storm, Steel Panthers: World at War, Lord of the rings battle for middle earth, Great Big War Game, Nintendo Wars, UniWar, XCOM 2 and uTanks."[2]

In the next chapter will be presented: about game (aim of the game, functionality with images and how to play), unity (introduction, the reason of using this game engine) and conclusion: things to be implemented later and bibliography.

2 "Climate War" Game Background

The aim of this game is to create an original game by crossing two areas: the turn based game types and a small part of the geography (the fauna and the map of the world), and from this crossing to get something original, fun and educative.

2.1 Story

The story of the game begins when, in a distant future, mankind is collapsing about air pollution, and the amount of carbon dioxide in the air has reached at alarming levels. The scientists have developed a substance that in contact with air change carbon dioxide in pure oxygen, only in time they found out that it is very harmful to man, but it was already too late for humanity. The only way to escape was to make a transplant between the human mind and an animal, only for part of the population, the process being very expensive and difficult, and the rest died in torment. After a long period of time in which humanity has turned into animals, natural hazards begin to emerge. They find out from the Maya's writings that the end of time is coming, and their only rescue is to get some runes across Europe, the role is requiring the gods to save them, but with the condition

of purifying the earth through fire or ice, for a limited period of time. In this sense, a dispute was created, and then it transformed to war between the animals in the hot zone of the world and those in the cold area.

2.2 About Gameplay

"Climate War" is a turn based strategy in which there are two camps: the animals in the cold zone of the world (the polar bear, the Arctic hare, the reindeer etc.) and those in the hot zone (monkeys, lions, elephants etc.), the fight will be between these two camps, and you will have the opportunity to play with which camp you want against the computer. The game will not focus too much on animations, but on the complex side, making the player choose the best moves.

The map of the game will be represented by a 2D copy of the earth, and the conflict area will be in Europe where will be conquest of territories, if you take all enemy territories you will win

Europe will be segmented in three parts: in the north the cold animals will have an advantage in the direct battle, in the southern part of Europe the others will have the advantage, and in the middle zone there will be no bonus. The seasons will benefit one camp or another: winter for animals in the north zone and summer: for other animals, and Spring and Autumn will benefit nobody.

The animals will be on the ranks, the 1st ones will cost very little, and will aim at feeding carnivorous animals, for example: deer, rabbits, pigs, etc. and the last rank will be the animals that will cost a lot will be hard to maintain.

Time will be in the months of the year, in turns more precisely, a turn representing a set of moves that you will make.

3 Implementation

3.1 Main Menu

In the Main Menu (figure 1) you will have a few buttons: Play-It will take you to another menu (Figure 2) with other buttons: the "New" button to start the game from scratch, and it will take you to another menu (Figure 3), where you will choose which part you want to be: Cold Animals (Hot Zone) and Hot Buttons (Return to Play Menu), where you will return to the Play Menu area, "Load" to enter the game the last time you save and "Return To Main Menu "where it will take you to the main menu, Options: to adjust volume of music and sounds in the game (Figure 4), About Game: information about game, how to play and tips for this game and a Quit button to exit the game.

Figure 1. Game Menu

Figure 2. Play Menu

Figure 3. New Game Menu

Figure 4. Options Menu

3.2 In Game

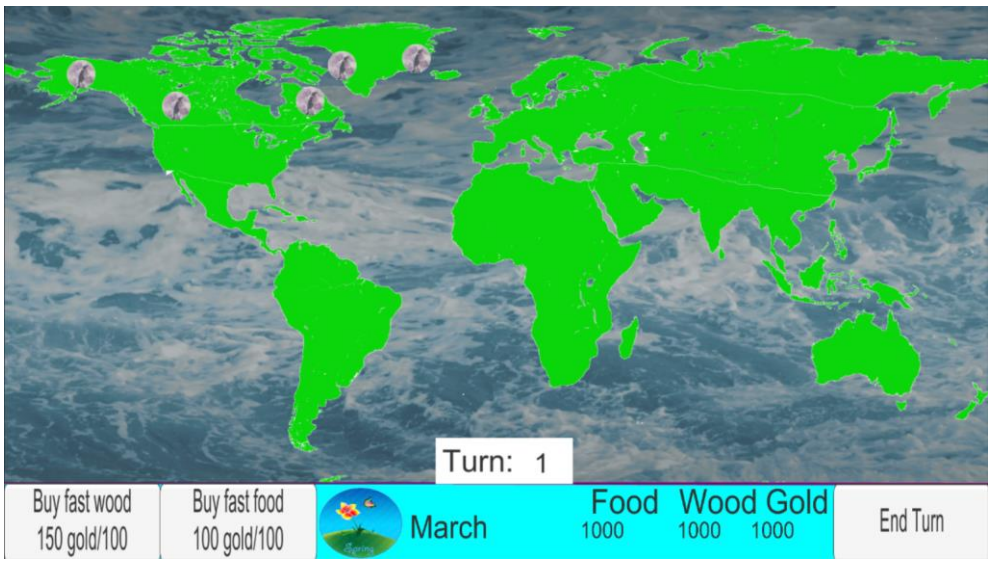


Figure 5. In Game

In Figure 5 there are some buttons and views: Bar UI (figure 5) represented by: 2 buttons (in the left) will help you to buy fast resources like Wood, you pay 150 gold for 100 wood quantity and Food: 100 gold for 100 food, these are not so efficiency than to build a house for them, like Mine Gold or Lumber Mill, in the middle a month text who represent the current month, one turn represent 1 month and an icon with Image Season that will change at every 3 turns with other season picture, in the right it show your resources: Food, Wood, Gold, in start you will have 1000 quantity for each other, and End Turn Button that is to end your set of moves and in top you will have a small view with current turn, at an ending turn the number will increase and a button that you can create buildings (Figure 6): Lumber Mill (for wood resources), Food House (food resources), Gold Mine(gold resources) and Army House, at every building you will have information (Figure 7): hint for building them, cost of building and number of turns to wait for created them.

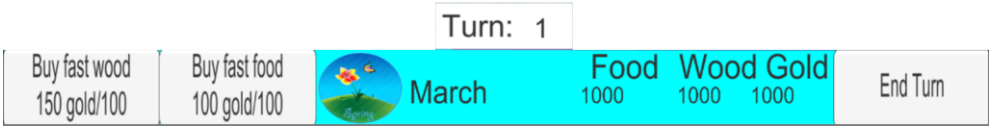


Figure 6. Bar UI

Figure 7. Building Creator

Figure 8. Building information

3.3 Buildings

In Figure 9 there are the all the buildings in the game, from left to right: Lumber mill(1), Mine of Gold(2), Army House(3) and Food House(4). For each building you will wait a number of turns to finish them.

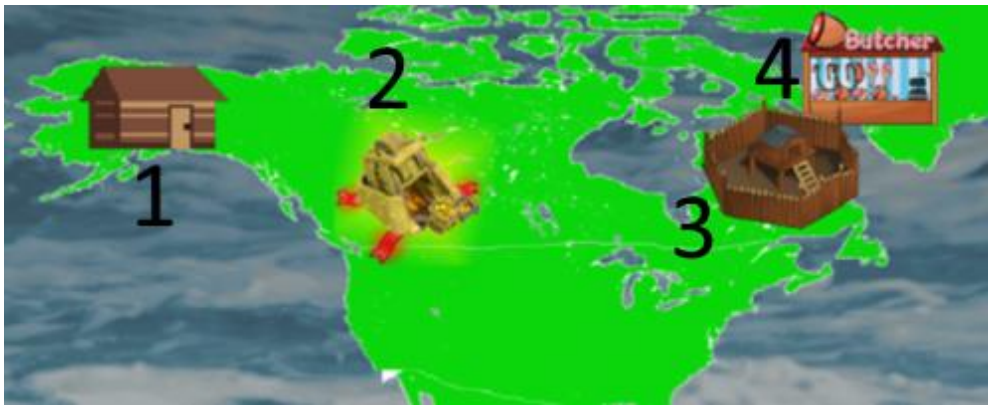


Figure 9. Buildings

The most important building is Army house, in figure 10. You can create an army as how large you want, if you have enough food. You will have different animals from zone to zone like below:

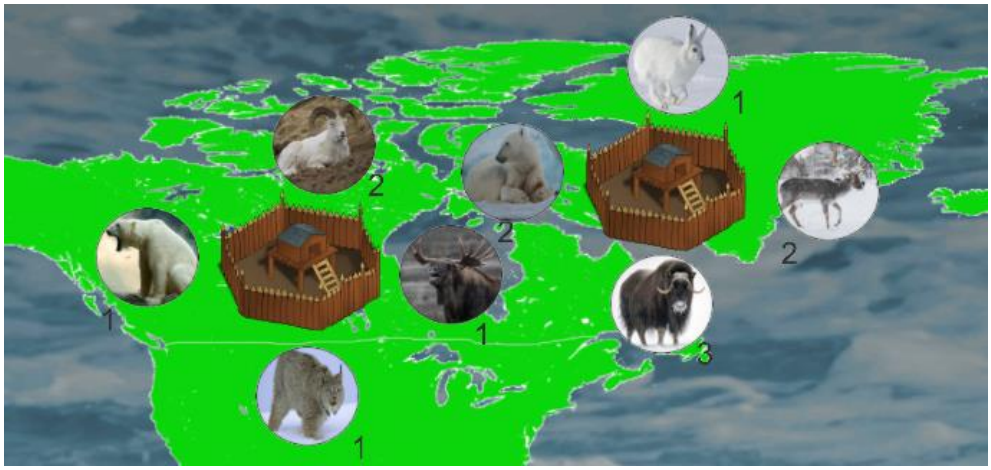


Figure 10. Army House

You have a count number for every type of animal in waiting, and every animal have a number of turn to add in your army. When a troupe of a type of an animal is done, it creates automatically a flag that represent all army in this territory like Figure 11.

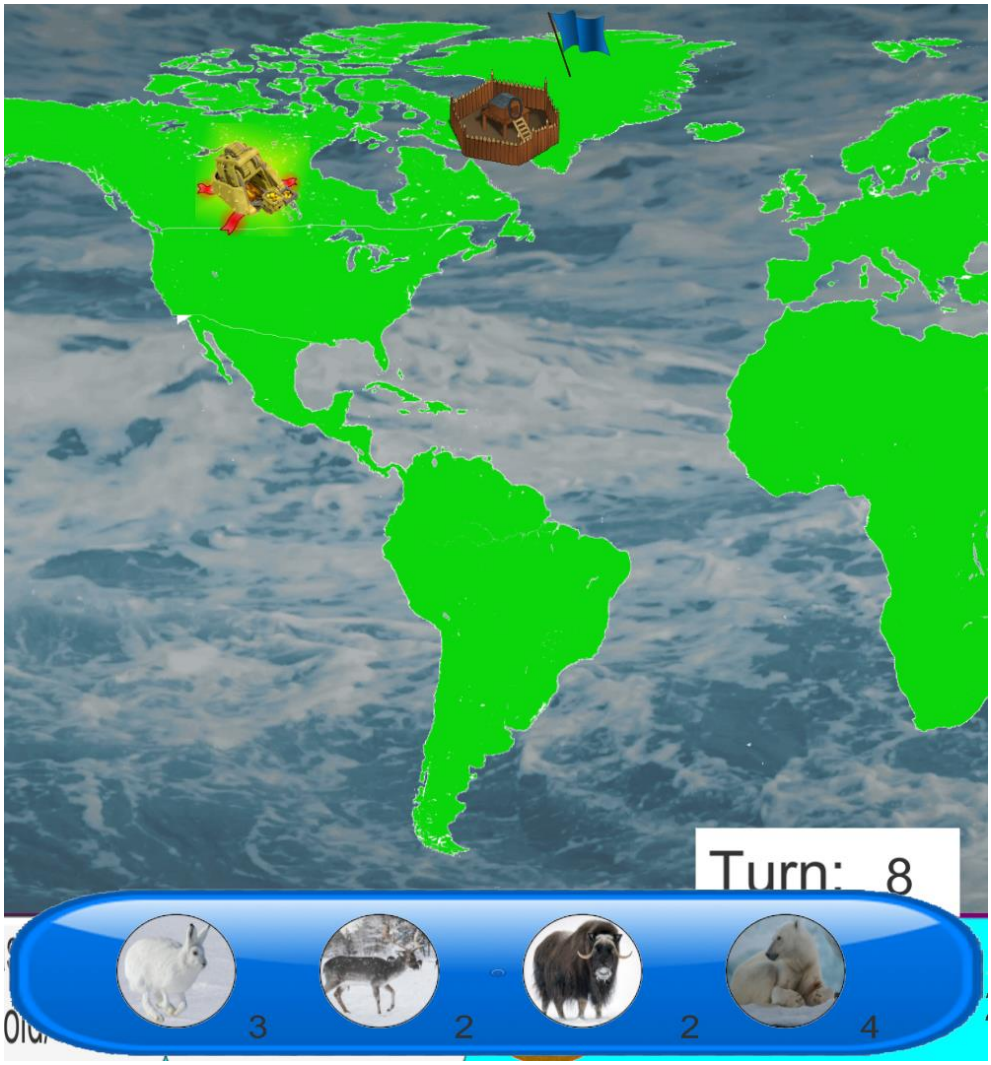


Figure 11. Flags and Troupes

Every animal will have stats about him, before buying them (Figure 12): attack, cost, wait to finish and if you press button "I" will appear information about the animal. (Figure 13)

Figure 12. Animal Stats

Figure 13. Animal Information

3.4 Unity

The reason of using this game engine is because is pretty easy for beginners and a good startup if you want to create games, for my project I used Unity 2D. "Unity is a cross-platform game engine developed by Unity Technologies, which is primarily used to develop both three-dimensional and two-dimensional video games and simulations for computers, consoles, and mobile devices."[3]

The most useful things at unity what I used are: *GameObject* and *Tags*.

"GameObjects are the fundamental objects in Unity that represent characters, props and scenery. They do not accomplish much in themselves but they act as containers for Components, which implement the real functionality."[4] For example, in Figure 14, a Light object is created by attaching a Light component to a GameObject.

In the right image represents information about GameObject, in this case the Light: position, rotation, scale, and you can add more components like "Rigidbody" component (to add physics to this object), Scripts (in C# or javascript), SpriteRenderer component (to add an image to this) and many others.

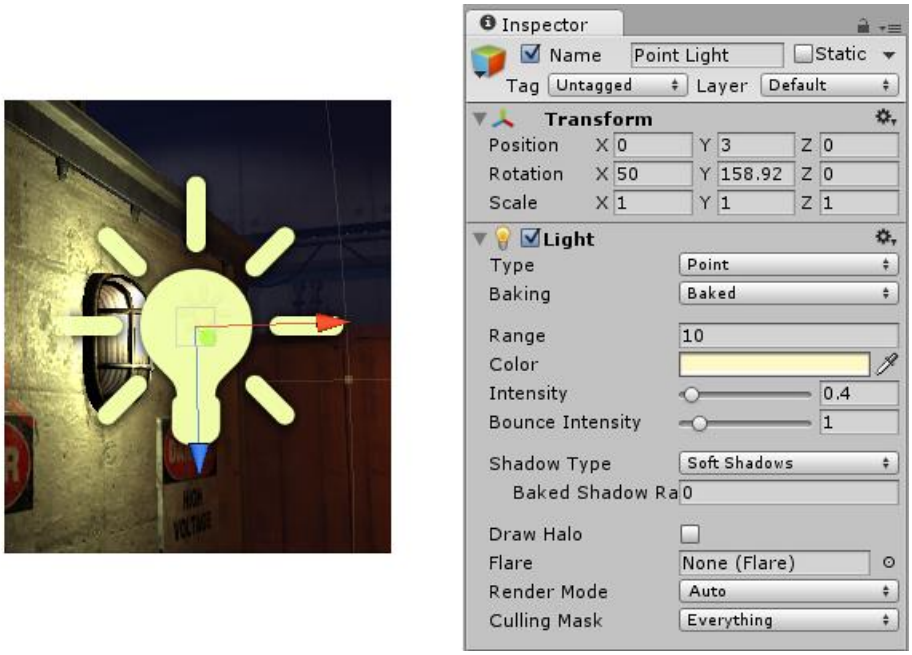


Figure 14. GameObjects

A **Tag** is a reference word which you can assign to one or more GameObjects. For example, you might define "Player" Tags for player-controlled characters and an "Enemy" Tag for non-player-controlled characters. You might define items the player can collect in a Scene with a "Collectable" Tag.

Tags help you identify GameObjects for scripting purposes. They ensure you don't need to manually add GameObjects to a script's exposed properties using drag and drop, thereby saving time when you are using the same script code in multiple GameObjects.

Tags are useful for triggers in Collider control scripts; they need to work out whether the player is interacting with an enemy, a prop, or a collectable, for example.

You can use the GameObject.FindWithTag() function to find a GameObject by setting it to look for any object that contains the Tag you want. In the Figure 15, the following example uses GameObject.FindWithTag(). It instantiates respawnPrefab at the location of GameObjects with the Tag "Respawn".

```
using UnityEngine;
using System.Collections;

public class Example : MonoBehaviour {
    public GameObject respawnPrefab;
    public GameObject respawn;
    void Start() {
        if (respawn == null)
            respawn = GameObject.FindWithTag("Respawn");

        Instantiate(respawnPrefab, respawn.transform.position, respawn.transform.rotation) as GameObject;
    }
}
```

Figure 15. Example in C# for Tags

4 Conclusions

In conclusion, the next features for this game will be: scouting enemy army with birds (you can spy your enemy territories with birds and you can get information about his army), battle fight with computer (implement AI to enemy and conquer territories from him), marine animals(for now are just land animals and will be animals like: whales, seals, octopus), ambush when animals attacks you (you can send "special animals" like giant octopus when enemy attacks you, for an example in real life are mines from war), neutral animals (animals not from north zone or south but in the middle), where you will have the chance to recruit them (if you gave them enough money, more than enemy), integrate music and sounds (sounds like clicking on buttons, buying animals and hear specific sounds for them and from different events, the music from this game will be a playlist that plays in background), better graphics (new sprites and new images) and simulate battles (you can testing with a number of soldiers that you introduce as input).

References

[1] Psychology in Video-Games, https://www.sekg.net/gamer-psychology-people-play-games/
[2] Turn Based Strategy, https://en.wikipedia.org/wiki/Turn-based_strategy
[3] Unity, https://en.wikipedia.org/wiki/Unity_(game_engine)
[4] Unity-GameObject component, https://docs.unity3d.com/560/Documentation/Manual/class-GameObject.html
[5] Unity-Tags component, https://docs.unity3d.com/Manual/Tags.html

Mihail JURAVLEA
"Lucian Blaga" University of Sibiu
Faculty of Science - Informatics
Str. Dr. I. Ratiu, No.5-7, Sibiu, 550012
ROMANIA
E-mail: the_mihail@yahoo.com

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Modeling of a life cycle of an innovation project

Asaf Ali Khan

Abstract

In this article a life cycle of an innovation project was described and studied. Types of innovations were classified and discussed. Modeling of a life cycle of an innovation project was made with the help of the Business Studio program in IDEF0 (Integration Definition for Function Modeling) notation. The paper illustrates advantages and disadvantages of a life cycle of an innovation project modeling.

1 Introduction

The main source of a good idea is experience. You may have passion in something and also you may have problems with it. Make notes all the time you have any ideas of how to solve the problems. Later you could sell these ideas. Unfortunately **most of the start-ups are not successful** because they really don't solve any problems at all [1]. Great start-ups solve old problem a little bit better then others (for example - Google compare to Yahoo) or they can solve old problem another way (for example – to sell books not in the stores but in the Internet like Amazon did) (Fig. 1).

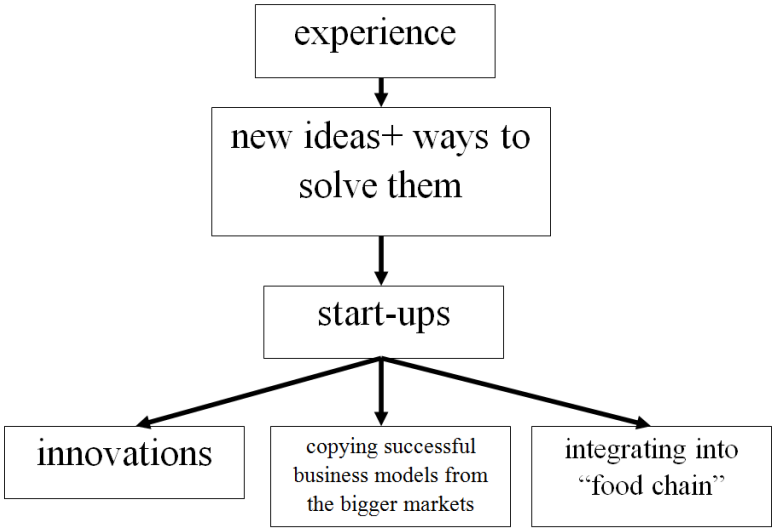


Figure 1 – Types of start-ups

Innovations become a good start-up very seldom.

The second way to create a start-up is **copying successful business models from the bigger markets**. This is the most fast and effective method in business. The most important components for start-up are market niche, product and team [2].

The third way to start a business is **integrating into "food chain"**. For example there is notebook market and there is notebook cases market. Also there are mouses and applications and so on. Usually if client pays big money for something, it is easy for him to pay a little more for extra service. Thereby there are many more little markets around big one. Successful businessmen integrate into "food chains".

At the heart of every start-up there is a certain original idea or development which lets it break into the market by offering something brand new and unique also to outstrip competitors and become monopolist itself. Such idea is called **basic innovation of a project**.

It is equally important to determine on which stage your project is. The methods you would use to get into a market, to outstrip competitors and eventually to earn money should match with these stages. (It is called "to commercialize an idea"). Before commercializing an idea you must protect your rights on it. In this case your idea becomes your intellectual property.

The problem of the formalized graphical representation of a life cycle of an innovation project stages for simplification of its understanding is actual. A main goal of this work is modeling of a life cycle of an innovation project.

This work consists of the description of types of innovation ideas and their classifications, the description and modeling of a life cycle of an innovation project stages by means of the Business Studio program in IDEF0 (Integration Definition for Function Modeling) notation.

2 Classifiction of innovative ideas

There are several types of innovative ideas that underlie the project (Fig. 2):

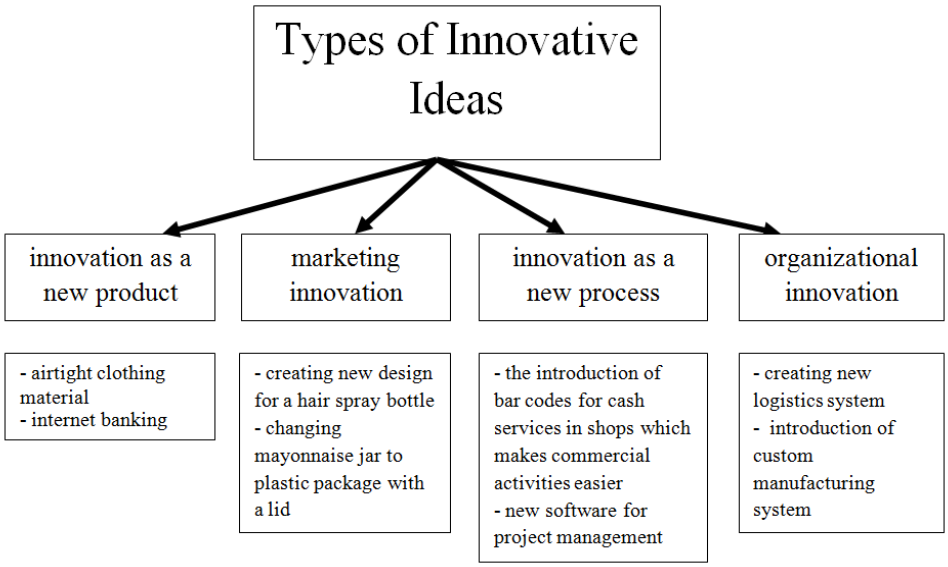


Figure 2 – Types of innovative ideas

1) **innovation as a new product**: this is an introduction of improved products and services. This includes also development of new methods of using this product. Examples of new products are - airtight clothing material, internet banking;

2) **innovation as a new process**: this is an introduction of a new or improved production method or a new delivery method of products and services. This includes changes in production technology, production equipment or software. Examples of innovation as a new process - installation of new automated equipment for the production line, the introduction of bar codes for cash services in shops which makes commercial activities easier, new software for project management;

3) **marketing innovation**: this is an introduction of a new method of promotion and selling already existed products (services). This includes changes in design, package, price and advertising methods of promotion. Examples of marketing innovations - creating discount or privileged cards for regular costumers, creating new design for a hair spray bottle, changing mayonnaise jar to plastic package with a lid. Distinctive feature of marketing innovation is it cannot change product or service but it can change consumer's attitude;

4) **organizational innovation**: this is an introduction of a new method of business practice in a company through cost reduction. Examples of organizational innovations – introduction of quality management system, creating new logistics system, introduction of custom manufacturing system. Distinctive feature of organizational innovations is affecting staff and work organization [3].

There is another classification of innovations. This classification is based on the idea of the usage for innovation. By this we mean areas or fields where innovations are used. It is possible to differentiate three principal applications for innovation: products, services and processes:

1) **Product Innovation**: products, especially consumer products are probably the most obvious innovation application. From a commercial perspective the attraction of product innovations is that the novelty of a new product will persuade consumers to make a purchase. Of course product innovations don't have to be consumer products; they can just as easily be industrial products such as machinery and equipment;

2) **Service Innovations**: often overlooked but equally important are service innovations that take the form of new service applications. One reason why service innovations don't attract as much attention as product innovations is that they are often less spectacular and less eyecatching. Service innovations typically take the form of a new way of providing a service, often with a novel and very different business model. Occasionally they even take the form of an entirely new service;

3) **Process Innovations**: if service innovations come second behind product innovations, then process innovations come the third. And yet process innovations often have an even bigger impact on society than either product or service innovations. Process innovations often have a big impact on the economics of production [4].

3 The main stages of a life cycle of an innovation project

There is a description of five key stages of a life cycle of an innovation project [2] but in practice there can be much more then five (Table 1).

Table 1
Stages of a life cycle of an innovation project

Name of a stage	Stage description	Results of a stage
Stage 1. Business as an idea, statement	An idea of creating a new business has appeared.	You already have a preliminary technical and economic justification.
Intermediate stage 1→2 Step 1. Search for investment for Research and Development (R&D)	Verification of scientific validity	Getting an external financing // Using your own funds
Step 2. Laboratory research or R&D// Market research		Patents or other types of intellectual property// Marketing research results

Name of a stage	Stage description	Results of a stage
		which confirm existence of demand
Stage 2. Business as a project, its appearance	Stage of design work for a new business project has begun.	Technical and economic justification.of an idea (Applications for contests or presentations)
Intermediate stage 2→3 Step 1. Search for investment for Research and Development (R&D)	Creation of a prototype – material model of an idea in the form of a product for consumer market	Getting an external financing // Using your own funds
Step 2. Research and Development (R&D)		Creation of a material prototype for a future product // A detailed list of services
Step 3. Choosing of development strategy for a new business		Business plan for a project
Step 4. Product typification		Technical documentation on a prototype // Detailed description of service delivery process
Stage 3. Business that is ready for selling a product, opening day.	Stage of a new business organization has begun.	
Intermediate stage 3→4 Step 1. Search for investment for business organization	Creating business according to plan	Getting an external financing // Using your own funds
Step 2. Opening of a company, starting accounting, hiring and training of staff		Registration certificate of a company, getting necessary licenses and permits for conducting activities
Step 3. Realization of business plan, making capital investments		Premises are equipped // there are renting premises, hired stuff// advertisement is placed
Stage 4. First sellings	Realization has begun	
Intermediate stage 4→5 Step 1. Earning revenues, getting contact information from first clients	Expansion of business	Getting revenue // Tax reporting with first sums
Step 2. Searching for investment for expantion of product/ service		Getting an external financing // Using your own funds
Step 3. Expansion of		Getting technological documentation

Name of a stage	Stage description	Results of a stage
product/service // or preparing a company for selling it.		for mass production // franchising // expansion of services
Stage 5. Growth // Sell of a business	Exiting an incubator	

4 Modeling of a life cycle of an innovation project in the IDEF0 notation

A hierarchical diagram of processes and subprocesses is represented in Fig. 3. A process hierarchy diagram (or functional decomposition diagram) provides a graphical view of the processes and helps to decompose them into a tree of sub-processes. Figure 3 gives the general represantation of the life cycle of the innovation project. The main process A0 consists of the 5 subprocesses A1-A5, which are also consists of subtasks.

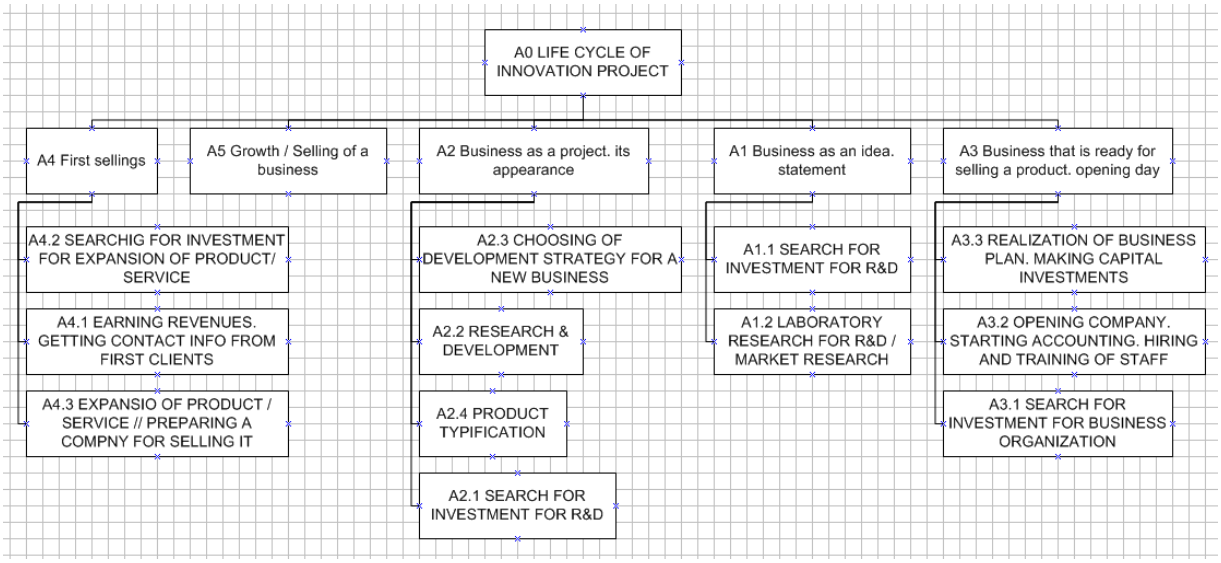


Figure 3 – A hierarchical diagram of processes

Modeling process of a life cycle of an innovation project with the use of the IDEF0 (Integration Definition for Function Modeling) notation is visually presented in Fig. 4-9. **IDEF0** is a function modeling methodology for describing manufacturing functions, which offers a functional modeling language for the analysis, development, reengineering, and integration of information systems and business processes. In this paper IDEF0 otation is used to better understanding of the life cycle of innovation pproject [5].

There is a **contextual chart** of the process "a life cycle of an innovation project" in Fig. 4 [1]. The chart consists of the **rectangular boxs** (containing a name and number) used to represent a function (process) and arrows, which connect boxes with each other and with the environment. There are four **main arrows** on the diagram: Inputs (left arrows), Controls (upper arrows), Outputs (right arrows) and Mechanisms (down arrows).

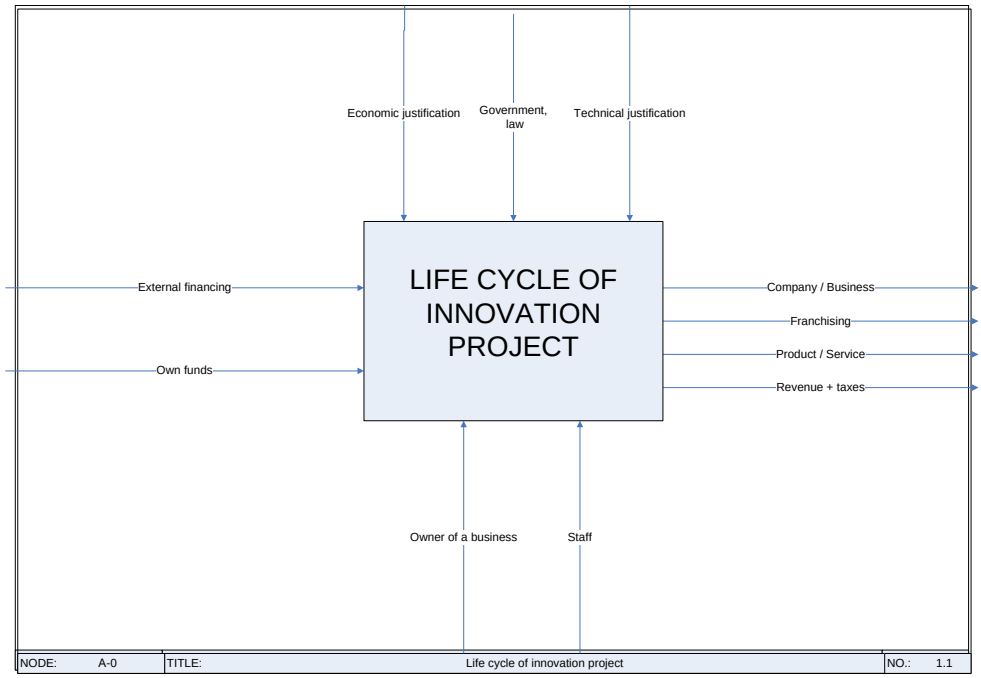


Figure 4 – Contextual chart of the process

The IDEF0 diagram was made in Business Studio program [6]. This diagram is based on the principal of decomposition, which means the partitioning of a modeled function into its component functions. In Fig. 5 you can see the main six stages of the life cycle of an innovation project (decomposition of the main function A0).

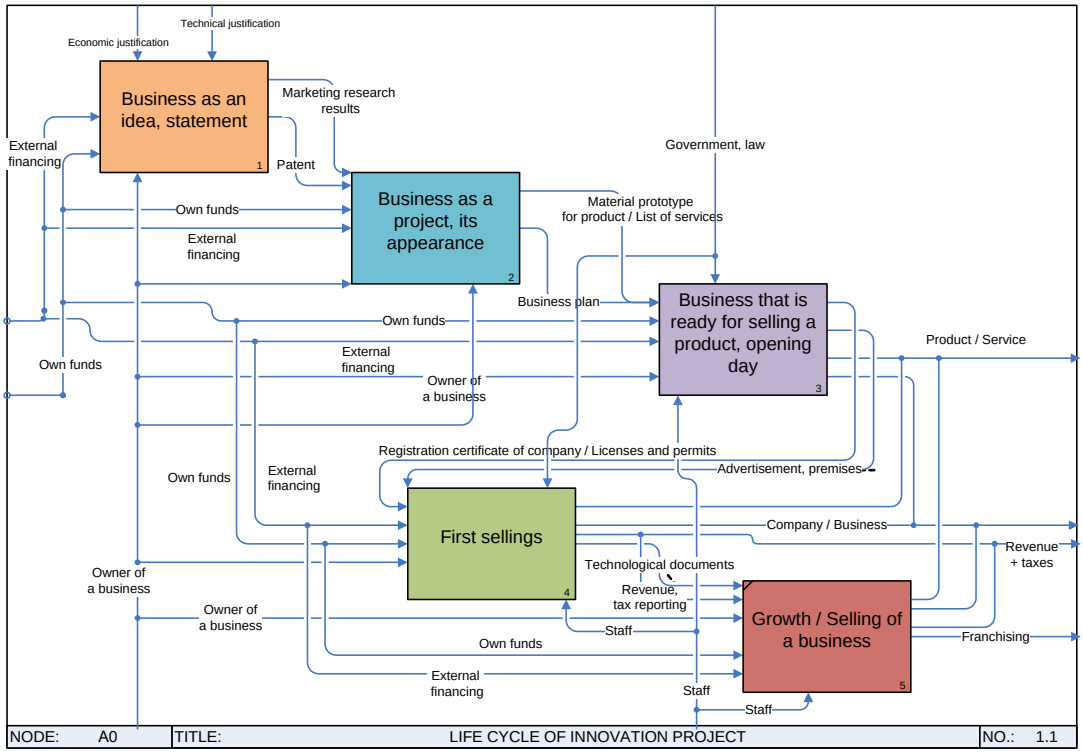


Figure 5 – A0 Life cycle of an innovation project

In Fig. 6 there are the main steps of the first stage of the life cycle of an innovation project (process A1 – Business as an idea, statement).

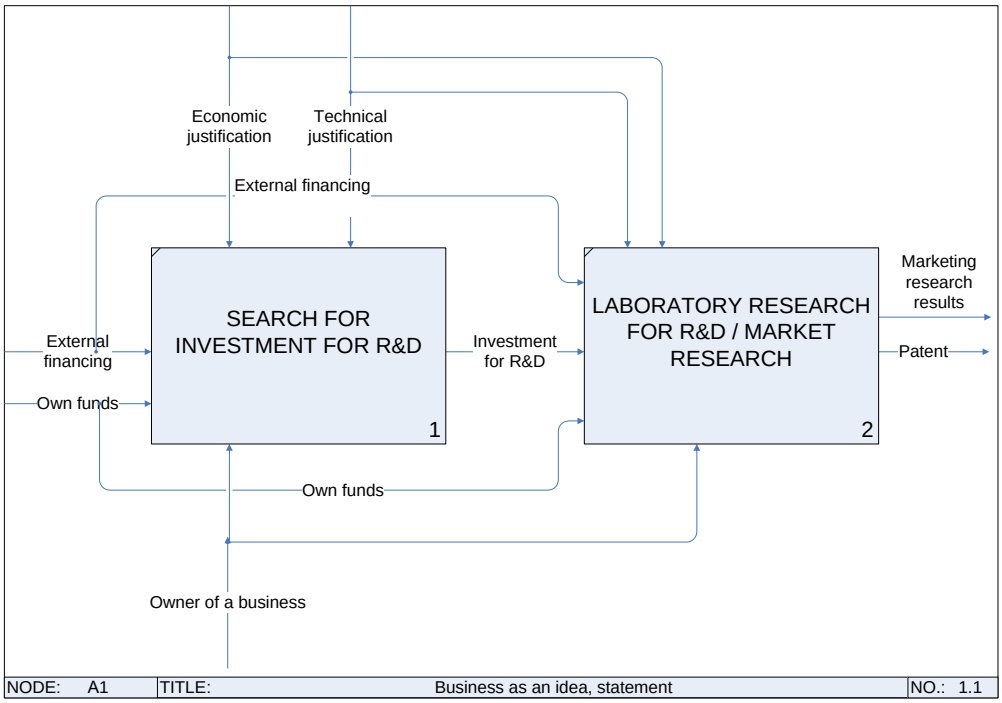


Figure 6 – A1 Business as an idea, statement

In Fig. 7 there are the main steps of the second stage of the life cycle of an innovation project (process A2 – Business as a project, its appearance).

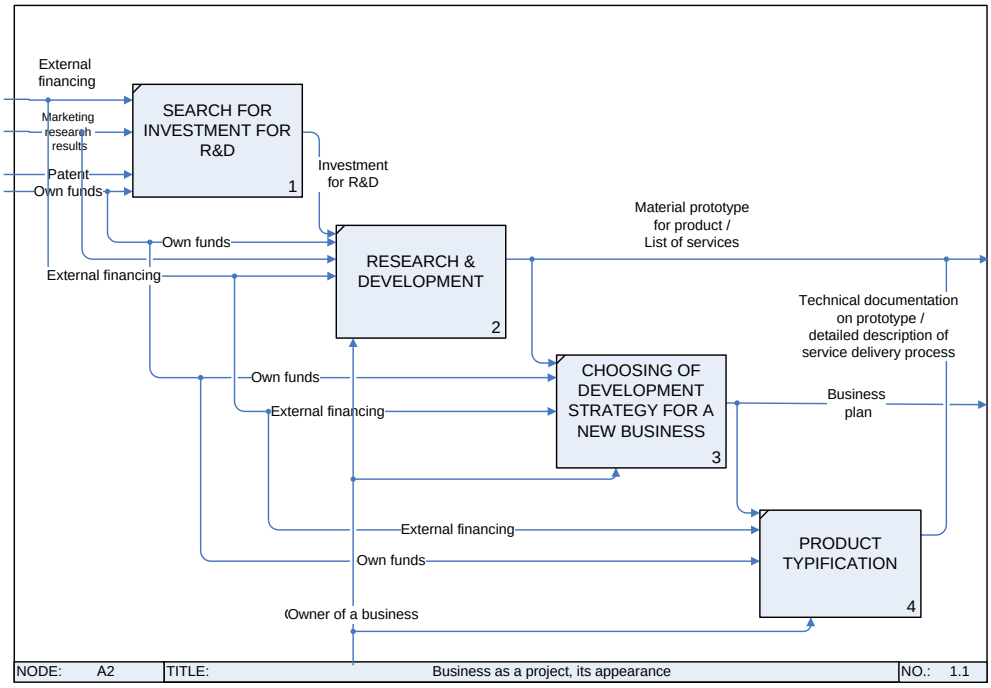


Figure 7 – A2 Business as a product, its appearance

In Fig. 8 there are the main steps of the third stage of the life cycle of an innovation project (process A3 – Business that is ready for selling a product).

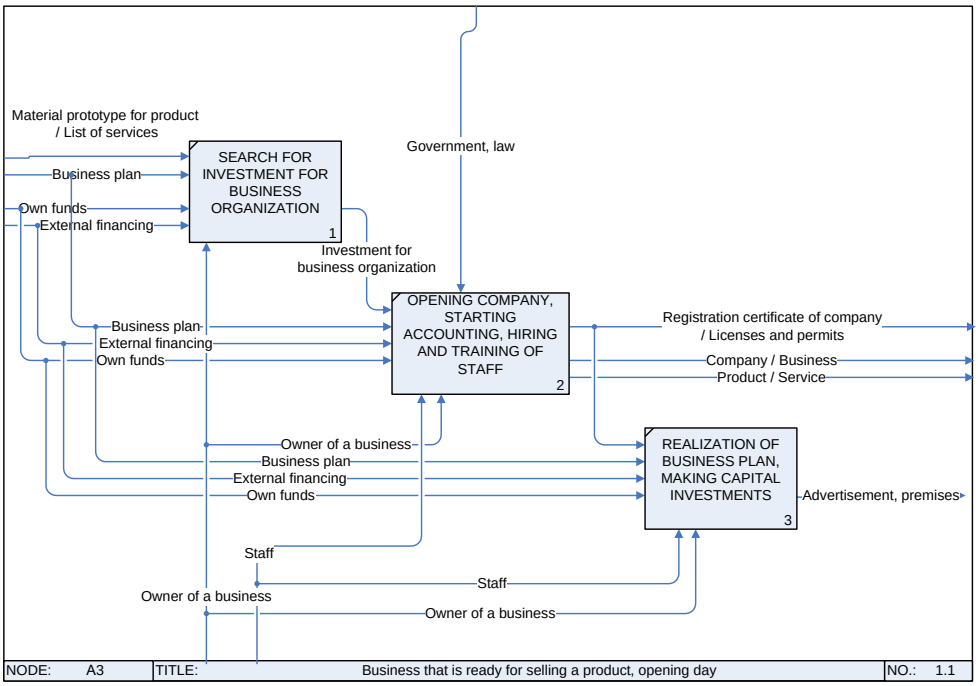


Figure 8 – A3 Business that is ready for selling a product, opening day

In Fig. 9 there are the main steps of the fourth stage of the life cycle of an innovation project (process A4 – First sellings).

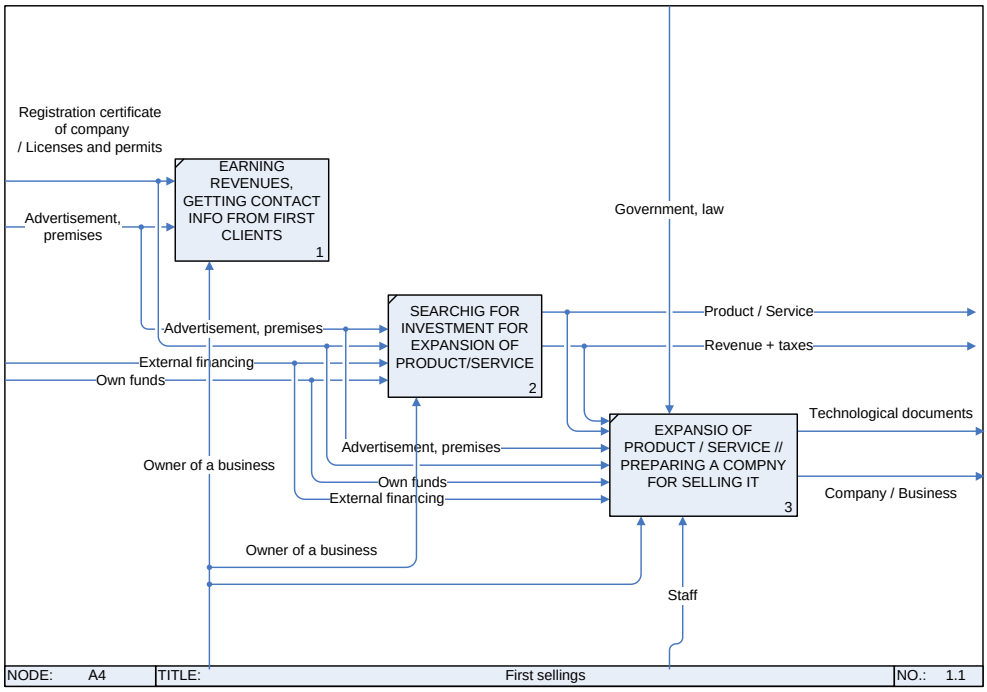


Figure 9 – A4 First sellings

5 Conclusions

Modeling of a life cycle of an innovation project was made with the **IDEF0** notation (Table 2).

Table 2
Advantages and disadvantages of IDEF0 notation

	Advantages	**Disadvantages**
IDEF0 notation	- use of functional approach, - visibility of charts, - accurate syntax and semantics of charts, - possibility of decomposition of charts and modeling of hierarchy of processes	- not expressed temporary sequence, - absence of specifications of processes (including restrictions on an input and output)

The main advantage of the IDEF0 notation is in decomposition principle which is applied when partitioning difficult process into its components [1]. Thus the level of specification of the process is defined directly by the developer of the model. The paper gives the grounds to speak about an opportunity of practical application of the innovation life cycle model.

References

[1] Sokolova M.L., Chernov V.G. *Modeling of decision-making process by means of a method of a multicriteria alternative choice ELECTRA,* Romania: ICDD, 2015.
[2] Howells, J. *The Management of Innovation and Technology*, London: Sage Publications, 2005.
[3] Henry, J. and Walker, D. *Managing Innovation*, London: Sage Publications, 1991.
[4] Basalla, G. *The Evolution of Technology*, Cambridge: Cambridge University Press, 1988.
[5] Green, A. *Systems Engineering Fundamentals,* Defense Acquisition University Press, 2001.
[6] Kareem, G. *Business Studio and Directum*, Moscow, 2010.

Asaf Ali KHAN
The University of Siena
Faculty of Economics, Law and Political Sciences
School of Economics and Management
Banchi de sotto, 55, Siena
ITALY
E-mail: asafali.khan@student.unisi.it

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Termohub – IoT web service

Borislav Kosharov

Abstract

Termohub is a web application for managing internet of things devices, storing data from their sensors and displaying charts with the collected data.

1 Introduction

The release of ESP8266 [1] device a few years ago enabled internet connectivity of devices and a new area of IT has been created – Internet of Things (IoT). The price of that development board is approximately $2 and it has WiFi and a full network stack. Termohub is a web application developed in C# and ASP .NET Core [2]. It was first created only for temperature sensors (hence the name), but now any kind of sensor can be used. For the database I used Microsoft SQL Server [3] since it integrates well with ASP .NET Core and Entity Framework Core. Entity Framework is a ORM (object-relational mapper) framework for .NET Core and was used to design the database using code-first approach. Since the whole application is done with .NET Core it means it is cross-platform and can be run in Windows, Mac OS X and GNU/Lunix. The firmware for the IoT devices was written in C++ in the Arduino [4] environment.

A lot of other IoT-related applications exist, but most of them are paid and offered as a cloud service. Termohub is free and open-source [5] and offers a DIY (Do it yourself) approach. It can either be started on your own machine or just use it on my server [6].

2 Architecture

2.1 System design

The whole system can be separated in these modules: Firmware, Web API, Database, Web app. Figure 1 shows how all of the modules interact together.

The IoT device sends the data it collects from sensors every N seconds (can be changed from the web interface) in JSON format in a POST request to the server. The server handles those requests and stores them in SQL Server. Later an user can access that data trough the web site in the form of charts and can also configure each device and sensor. Next I will explain each module in more detail.

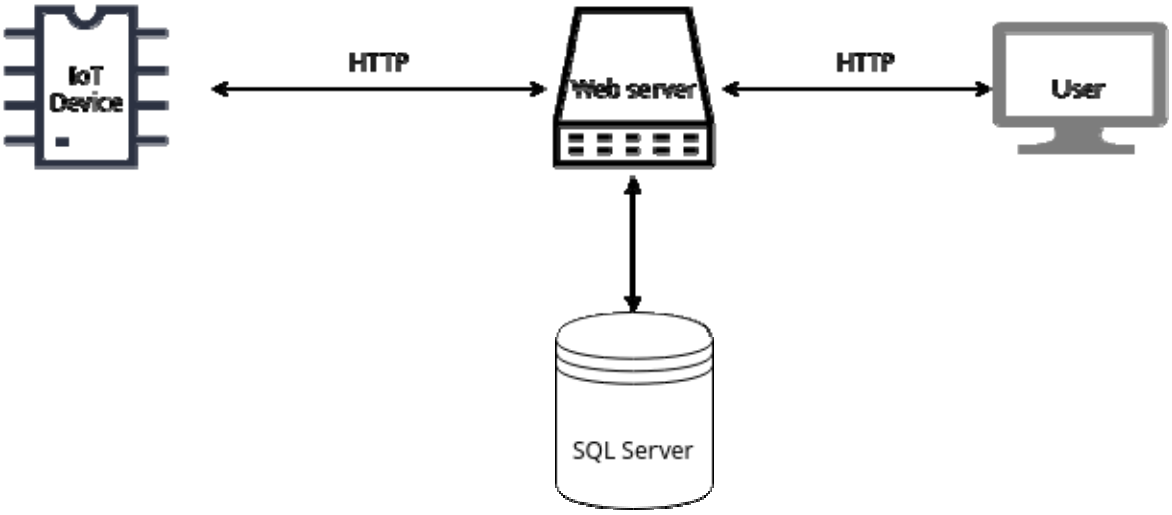


Figure 1: Architecture

2.2 Database

The database was designed in code-first approach which means no SQL code was written. The tables were designed as C# classes where each field is a column in the database. All the classes are mapped to tables with Entity Framework. It also translates LINQ (Language INtegrated Query) to SQL queries. This is very beneficial since SQL has different dialects and if you want to change the database engine you will have to change the SQL queries too, but with entity framework you write the table models and queries in C# and it translates them to SQL of the database you use. Any other Entity Framework supported database can be used. The only change you have to make is change the connection string.

Entity Framework also supports migrations. This means that the database model can have versions and we can change the design of the database iteratively without loosing any data. Entity Framework generates these migrations for us based on our model C# classes. So to add a column to a table is as easy as defining a new field in a class and tell it to generate a new migration.

The actual database model is designed like this. There is the User table that contains a list of Devices that the user owns. Each Device has columns for Name, Delay (this is the delay between device requests), Owner (an user) and a list of Sensors that belong to this Device. Each Sensor has a Name, Unit (unit of measure °C/°F used for display only), an optional Alert and a list of Readings. A Reading has Time and Value and represents a single data point in time. The Sensor's Alert stores Sign (representing > or <), Limit (a threshold number), Email (to who to send the alert) and IsNotified (if an e-mail was already sent). The database diagram is shown on Figure 2.

2.3 Firmware

The device firmware is written in C++ using the Arduino environment. It uses several open-source libraries for Arduino and ESP8266. It runs in two modes: configuration and execution.

When the device first starts it checks if it has a WiFi configuration stored in it's EEPROM. If it doesn't have then it turns itself into AP (Access Point) with SSID "Termohub" and password "12345678". It also starts a DNS server and a web server. When the user connects to the AP the device redirects all requests to it's web server. An interface opens and lets the user configure WiFi

by choosing SSID and entering a password for his own router. The user also has to enter his credentials for his account in Termohub's web service (you have to register beforehand). When the device connects successfully it stores the configuration in it's EEPROM. The next time it boots it would not turn itself into AP, but it will connect to the configured router as a client.

In the execution mode the device first sends a request to the web API with the stored user credentials and if successful gets an access token. Then the device initializes all sensors using the OneWire protocol. This protocol can be used to connect many sensors on a single bus and only use 3 pins for the whole bus. Each sensor has an address built-in. The protocol also enables connecting a different number of sensors without changing the code. It automatically finds all sensors on the bus. After initialization the device enters the main loop. It reads each sensor's value and builds a JSON payload. It sends a DeviceID (formed by taking a part of the device's MAC address), SensorID (formed by taking two bytes of the OneWire address) and Value (the reading from that sensor). Note that no Time value is sent, because it is the time the server writes the value in the database. As a response the server returns a single number that represents the delay the device should sleep before it's next request. With each request an Authorization header with the access token is also sent. This means that all unauthenticated requests will be ignored by the server.

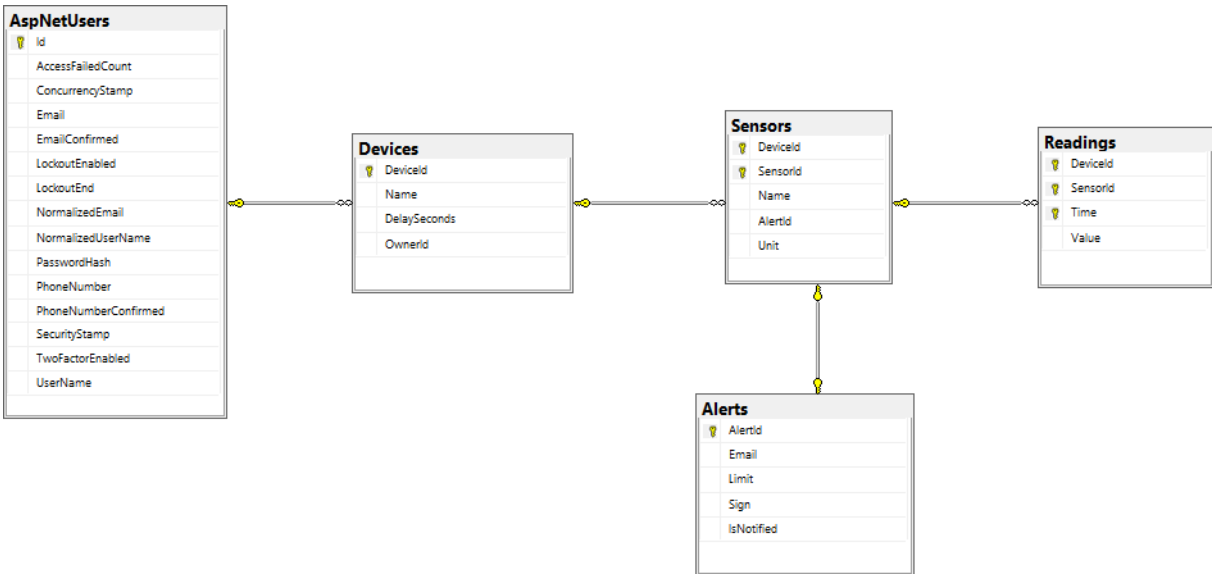


Figure 2: Database relations

2.4 Web application

ASP .NET Core is organized in MVC (Model View Controller) architecture and has DI (Dependency Injection) built-in. DI allows any service a controller needs to be injected into the constructor of that class. The models are the classes mentioned in the Database section used to design the tables. The controllers handle HTTP requests and serve the views to the user. The views contain the HTML content, but may contain some C# logic.

Users can be in Admin or User role. The Admins can view all the devices and sensors in the system while normal Users can view only those who they own. There are two types of authentication schemes: Cookie and JWT. Cookie authentication is used when using a browser while the JWT is for the API for the devices. The Account controller handles user registration and login. The first user that is registered in the system becomes an Admin. Each next user becomes a normal User, but Admins can promote normal users to Admins.

The Device controller shows a list of devices (Figure 3). If the current user is an Admin it shows all devices of all users. If the user is a normal User it only shows the devices that user owns.

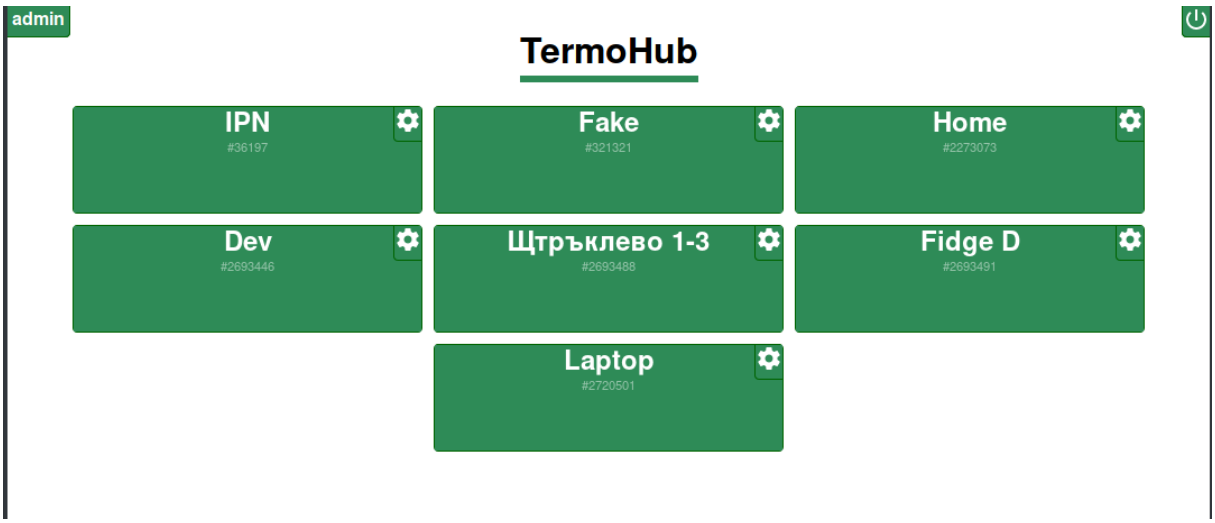


Figure 3: List of devices

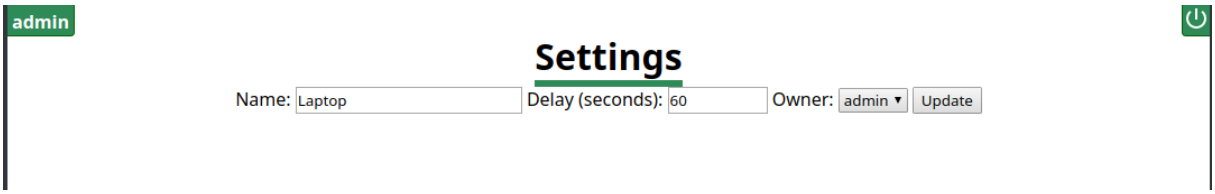


Figure 4: Device settings

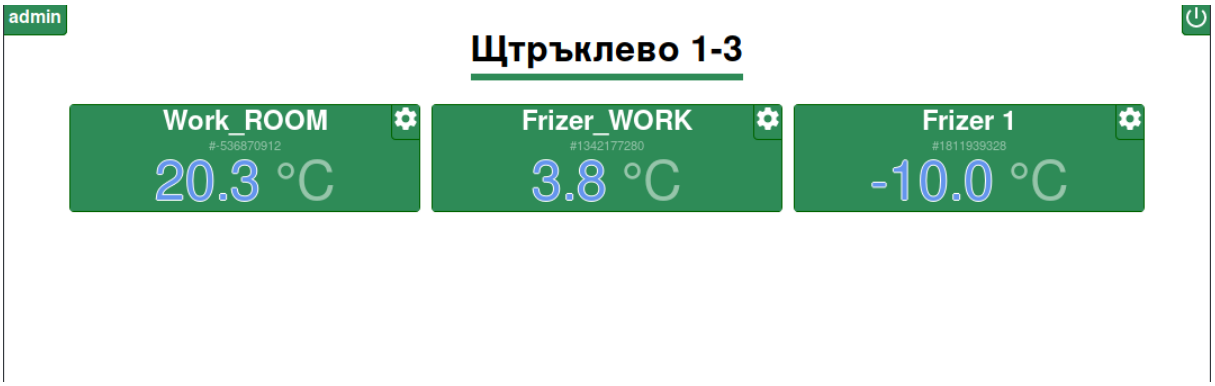


Figure 5: List of sensors

Each device can be configured (by pressing on the cog). When configuring a device it's name and delay can be changed. If the current user is an admin then also the device's owner can be changed (Figure 4).

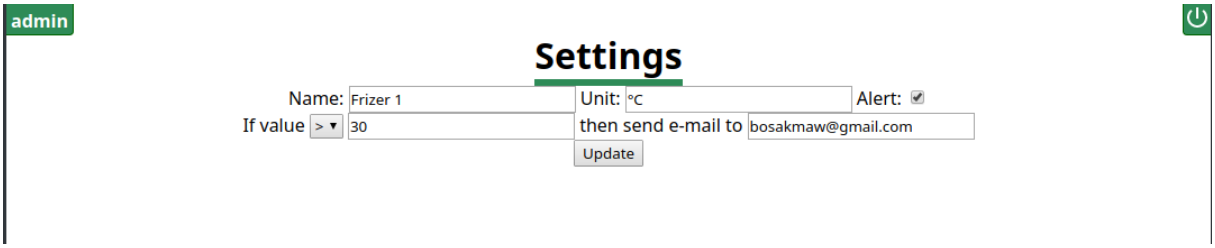


Figure 6: Sensor settings and alert

When you click on a device a list of that device's sensors is shown. The user interface is the same as the list of devices. For each sensor the most recently recorded temperature is shown (Figure 5).

Each sensor can be configured to set it's name, unit of measure and set an alert (Figure 6). When the alert checkbox is checked the additional fields bellow appear. When there is a reading with value above/below the given limit value an e-mail will be sent with an alert and a link to that sensor. An e-mail is sent only once when the limit value is exceeded. Another e-mail is sent when the temperature goes back to normal confirming that the problem is fixed. When you click on a sensor a page with a chart with the most recent temperatures is drawn. The same page can be used to show information for an older date given you provide from and to dates interval. The Live link changes the view to a similar chart that updates itself live as data comes in. The report link downloads a .csv format file with the data in the from-to date interval. It can later be used in a spreadsheet program.

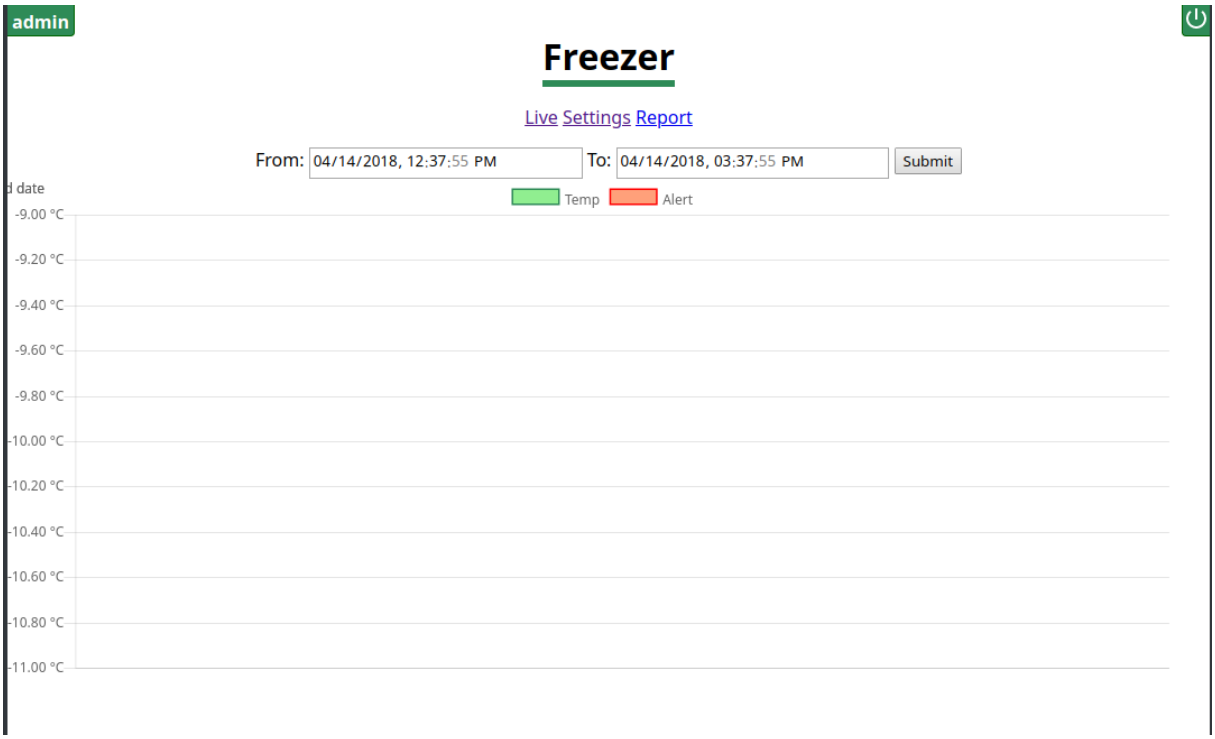


Figure 7: Sensor details and chart

The front-end part of the application that draws the charts is done in JavaScript using the Chart.js library. For the live chart every few seconds the script requests for newer data and appends it to the chart points.

3 Conclusion

Developing for the internet of things devices allows us to use existing technologies for web development to do exciting projects. Devices are now connected to the internet and we can think about them as web services. Termohub involved full stack programming from doing the hardware, writing the firmware, doing the back-end web API and then doing the front-end charts. I hope this paper has given some insight in this new field of IT.

Acknowledgement: This work was supervised by Prof. *Katalina Grigorova* from *University of Ruse*.

References

[1] *ESP8266 device*
https://en.wikipedia.org/wiki/ESP8266
[2] *ASP .NET Core framework*
https://www.asp.net/
[3] *Microsoft SQL Server*
https://www.microsoft.com/en-us/sql-server/sql-server-2017
[4] *Arduino*
https://www.arduino.cc/
[5] *Termohub on Github*
https://github.com/nikibobi/termo-hub
[6] *Termohub deployed*
http://asp.ru-se.com:5000
[7] *Chart.js library*
https://www.chartjs.org/

Borislav KOSHAROV
University of Ruse
Department of Informatics and Information Technologies
8, Studentska str., Ruse 7017
BULGARIA
bosakmaw@gmail.com

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Emergency Situations

Nicolae Lungu

Abstract

This Every time a patient goes to a doctor he needs to carry with him all his medical records, or at least a big part of them, but he does not have all of them somewhere centralized, and in case of need he or a specialized doctor cannot access them all together fast. “Emergency Situations” web application is designed to help patients and doctors to gain a stronger collaboration in case of emergency situations, and the main purpose of this application is to identify a patient based on his fingerprint. So in case of an emergency situation, when the life of the patient depends on those few minutes and the doctor has no idea what is happening with the patient, by scanning his fingerprint the doctor will gain access to the patient personal medical data and will be able to see what kind of allergies, diseases, surgeries, implants the patient has and will be able to take fast and precise measures avoiding the routine checks which can cost the patient life.

1 Introduction

1.1 Topic

Our society as a nation has the great luck to live and work in such special moments. It is a great time to take full advantage of the technology available nowadays and not just for commercial purposes but also for social purposes for medical purposes, industrial, and any other type of purpose that relates to the era we live in. In everyday work, whether we are students, employees, entrepreneurs, we are already combined with everything that means technology and internet speed. Of course there is plenty of services available in the form of web applications that help us and simplify our life. Starting from simple and banal landing pages that promote and inform us about the different realms of today and their work. Products that are created for solving some artificial needs, something that we normally do not need and without which we could lead a normal way of life.

Subscriptions and web services, monitoring of private and state institutions. Airports, train stations, buses, warehouses. Web services to collect data and give citizens a safer way to live in a particular place. In other words, everything that means web services is everywhere in our lives whether we want it or not.

1.2 Intention

But, unfortunately, many of them are necessities created to be monetized. The vision about the idea conceived tends to be about something else, tends to be about how to help people in critical situations

where a few actions can make a difference in a person's life, can prolong the life of a man. So the developed project is a web service in the medical area.

Based on an article from Wikipedia „Medicine is a branch of biological sciences that aims, on the one hand, in the study of the human body and its functioning, on the other, based on this knowledge, the preservation and restoration of health. Medicine operates with newer or older concepts in most sciences, from fundamental human anatomy, to complex mathematical models and even borrowed notions in the field of philosophy and art."[1]

Nowadays, medicine has reached a very high position with regard to our daily lives. Over the years, so many improvements have been made in this area, people have advanced a lot along with them and learned how to take care of the body, of our health. However, people are living less and less increasingly ill-tempered.

As far as Romania is concerned, we still cannot benefit from what digitized medicine means. In order not to create confusion, medicine has a lot of elements that were only possible with the help of information technologies, but here, the discussion is about digitized medical history that can be accessed anywhere in any situation that requires a context for taking an important decision for a patient. A digitized medical history that gives the patient the possibility of knowing exactly what his medical condition is, and a system that allows doctors to evaluate and solve medical problems, to provide treatments after understanding the entire medical history of a patient. And last but most importantly, accessing the medical profile based on the fingerprint.

1.3 Approach

In Romania, there is a system offered by the National Health Insurance House, whereby each citizen has a health card where his medical history is kept. However, accessing that card involves the introduction of a pine known only by the patient.

So important information about a patient will be stored in this system only if the patient gives his/her consent, this being very important and correct. But analyzing the problem that is being proposed to solve in an emergency when the patient can not cooperate to solve his medical problems, that card is useless, especially if it is not known who the patient is.

Here comes the novelty of the concept that has been developed. Namely, to associate each patient with his fingerprints. In this way, in emergency situations when the patient can not cooperate with the doctors to solve his medical problems, urgent ones, doctors will have access to his medical profile and will be able to act quickly based on the medical history of the patient.

2 Current status

The web application was made using the "Java Spring Boot" technology, based on Model View Controller (MVC).

The part of the models is a simple one, namely the database that is reflected by the models. This is due to the fact that the attention was on an architecture open to modifications and subsequent developments that tends to respect some principles of object-oriented programming such as "SOLID". So the database and the models are represented by three entities: "Doctor.java", "Patient.java", and "MedicalHistoryItem.java". For the relationships and generation of these entities "Hibernate" technology was used. By means of the compilation of the web application, using annotations at runtime "Hibernate" is generating the structure of the database and its relationships. The part of the views was built using HTML, CSS and Thymeleaf technologies. Thymeleaf is a template engine that allows working with dynamic objects in built web pages. The

design is a simplistic one designed to highlight the data and their rapid distinction that was made with HTML and CSS.

The controllers part is composed of “MainController.java", "DoctorController.java", and "atientController.java”. The "MainController.java" handles uploading the profile of a doctor or a patient to the web application with all the relevant data about it. "DoctorController.java" deals with the authentication of a doctor, the creation of a new account for a particular patient, and the management of patients' medical records. And "PatientController.java" deals with authenticating a patient and identifying a patient based on the id generated by fingerprint reader.

In addition to this Model View Controller architecture, an abstraction layer was added to ensure that data from the database is not altered, as well as the modularization of the web application is assured. This was possible by using a set of Repositories, Services and Dto’s. With the help of the "Repository" components, the data from the database was brought through dependency injections in the "Service" components. And through the "Dto" components the data were brought, processed, and displayed using a copy instead of the original data in the front end, and not just that, just the strictly necessary data.

As we can see in *Figure 1*., the purpose of this web application is that following authentication based on a patient's fingerprint to synthesize and digitize its medical history. Moreover, to quickly present as seen on the left side of the image the status of diseases, fractures, surgical operations, successfully performed with green colour, and with red colour, those that were only detected and unhealed.

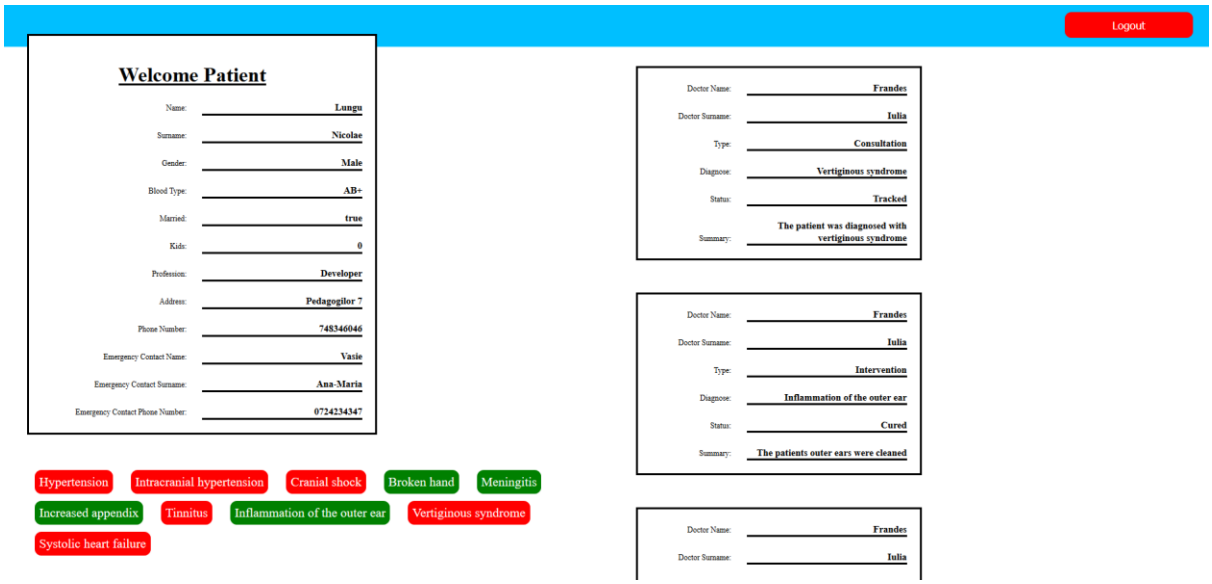


Figure 1. Main application view

3 Projects components

In the presented paper the project is a combination of a web application in which is integrated a part for indentifying the patient based on his/her fingerprints.

As Amos Ndegwa explains, “A web application is a computer program that utilizes web browsers and web technology to perform tasks over the Internet. Web applications are usually coded in browser-supported language such as JavaScript, Java, and HTML as these languages rely on the browser to render the program executable. Some of the applications are dynamic, requiring server-side processing. Others are completely static with no processing required at the server. The web application requires a web server to manage requests from the client, an application server to perform

the tasks requested, and, sometimes, a database to store the information. Application server technology ranges from ASP.NET, ASP, Java and ColdFusion, to PHP and JS."[2]

In her article Stephanie Watson explains that, "Fingerprints are the tiny ridges, whorls and valley patterns on the tip of each finger. They form from pressure on a baby's tiny, developing fingers in the womb. No two people have been found to have the same fingerprints, they are totally unique. There's a one in 64 billion chance that your fingerprint will match up exactly with someone else's. Fingerprints are even more unique than DNA, the genetic material in each of our cells. Although identical twins can share the same DNA, or at least most of it, they can't have the same fingerprints. Fingerprinting is one form of biometrics, a science that uses people's physical characteristics to identify them. Fingerprints are ideal for this purpose because they are inexpensive to collect and analyze, and they never change, even as people age. Fingerprints are made of an arrangement of ridges, called friction ridges. Each ridge contains pores, which are attached to sweat glands under the skin. Everyone leaves fingerprints on glasses, tables and just about anything else because of this sweat."[3]

4 Technologies

Nowadays there are a lot of software editors that can help us develop something to solve a problem, or to improve an already existing domain. We have a very wide range of editors for the information storage part , for the information processing part, as well as the representation part.

For data storage there are editors which are widely known and widely used like MYSQL, MariaDB, MongoDB, PostgreSQL and many more.

For processing information and their representation are known and widely used programming languages such as C, C ++, C #, JAVA, Ruby, Phyton, HTML, CSS, JS, PHP and many more.

For developing this project both software components and hardware components were used.

Software components:

- JAVA SPRING BOOT APPLICATION based on MVC model
- Maven
- Hibernate
- XAMPP
- MYSQL
- HTML5
- CSS3
- Thymeleaf
- Git for storing project on BitBucket

Hardware components:

- ZK4500 fingerprint reader which has a builted in .sln GUI

5 Conclusions

5.1 Downsides

Following the articles published by different institutions and academic centers, there is a question of how to validate fingerprints following accidents or skin diseases. These problems arise from the fact that no matter how well the patient is rehabilitated after one of the above cases if the fingerprints were

affected, the percentage of matching of his current fingerprint with the fingerprint stored in the database may be unsatisfactory to be able to be validated.

Another issue is to distribute information and access it anywhere in the country no matter where it is needed. Each point of the system needs to be synchronized, and any information can be accessed very quickly.

Another issue is the security of information. At present biometric information is very difficult to obtain from a legal point of view, if not impossible, and a simple alteration of the information can cause very big damage.

In spite of these very important aspects that tend to stop work for the development of this system, based on information technologies a demo version of this system was developed.

5.2 Future developments

For the future, to integrate the fingerprint reader with the web application to be a whole would be the next step, because at this point the fingerprint id generated by the fingerprint reader must be manually entered into the creation and identification of a patient.

Another part that could be a great addition could be to implement a mobile part where the data stored in the database could be used via an API.

And last but not least real life implementation of this web system meant to save lives in emergencies.

Acknowledgement: This work was supervised by *Ralf Fabian*, from *Department of Mathematics and Informatics Faculty of Sciences, "Lucian Blaga" University of Sibiu*

References

[1] Wikipedia, *Medicine*, https://ro.wikipedia.org/wiki/Medicin%C4%83, (retrieved 05.2018)
[2] Amos Ndegwa, *What is a Web Application?*, https://www.maxcdn.com/one/visual-glossary/web-application/, (retrieved 05.2018)
[3] Stephanie Watson, *What are fingerprints?*, https://science.howstuffworks.com/fingerprinting1.htm, (retrieved 05.2018)
[4] *Fingerprint Reader*, http://en.zkteco.com/product/ZK4500_238.html, (retrieved 05.2018)

Nicolae LUNGU
"Lucian Blaga" University of Sibiu
Faculty of Scrience - Informatics
Str. Dr. I. Ratiu, No.5-7, Sibiu, 550012
ROMANIA
E-mail: nicolaelungu8@gmail.com

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Online Rendering Tools and Design in Blender

Alexandru Pintea

Abstract

The paper presents a 3D project made in Blender with the technical, industrial design facility. The created object is included in a newly created website that allows interfacing with the rendering, viewing and online collection of 3D models. 3D modeling is achieved today with many software technologies, including Blender. Online rendering allows images to be of high quality regardless of the graphics performance of the user's computer. Users could interact in real-time with the 3D model.

1 Introduction

The purpose of this paper is to build a project in Blender and include 3D models on a new web page with online rendering and viewing interface. Blender-based technical drawing applications and online applications for modelling, visualizing 3D objects and scenes are presented.

Online rendering is a case of transforming wireframe models into images or videos; the process required to transform 3D data into 2D. Online rendering can refer to two main processes. The first process uses an online service to download data after online rendering; the other process displays web content that includes rendering of an online 3D model that is displayed on a webpage in a browser.

The main current technologies that can be used are: WebGL (intensively used and developed), Web Player (minimalist), VRML (Virtual Reality scripting, VR, out-dated), X3D, ThreeJS and Blend4Web engine for Blender, a 3D development environment, For these, basic JavaScript, Python, C ++ and C knowledge is required, and a basic understanding of 3D graphics. There are many other languages that can be used for 3D graphics on the web. The minimum configuration for computers running 3D web elements includes: 2Ghz dual-core processor with SSE2 support, 2GB RAM, 1280x768 24-bit display, mouse or graphic track pad compatible with OpenGL 2.1 with RAM of 512 MB [1]. In the near future, most computer users will see 3D content on webpages, even virtual reality.

The paper is organized as follows. Section 2 describes software technologies used in the technical drawing. Section 3 includes the 3D project made in Blender and a new web page used as an interface dedicated to rendering and viewing 3D models. The paper concludes with the main facilities observed during the working process about technical design and online rendering.

2 Technologies

The most widely used software applications for modelling and viewing, especially online 3D objects are described in the current section.

WebGL is a programming interface for browsers using the Canvas HTML5 element. 2D canvas is the equivalent of a 3D scene with 3D elements to be rendered. WebGL uses JavaScript and OpenGL for interpretation and rendering. The factor that influences the operation of 3D elements is the real device on which they are displayed. Devices like Oculus Rift or Google Cardboard use WebVR.

WebVR is used for online and game activities as well as C++ and Javascript/C#. The WebVR interaction will become the next technological advance in terms of mass use. There is no barrier in the programming language because WebGL is written in JavaScript and is similar to Bootstrap. Real-time online render of 2D element objects is easy. 3D models implemented as STL or OBJ models used with Blend4Web can be successfully embedded in WebGL.

VRML is the acronym for Virtual Reality Modeling Language. Similar to HTML, VRML is used to describe 3D objects that the browser can interpret. The user can interact with VRML to view, move, and rotate 3D objects from the scene. Most of the time to view and work with a VRML file, you will need to have a viewer and install a browser plug-in. Files are saved as .WRL and can be used in web applications [10].

ThreeJS is a JavaScript library for rendering and 3D Web interaction. It can be embedded in the web project [13]. The main elements introduced by ThreeJS are "camera" and renderer, which are declared by the ThreeJS functions [13]. ThreeJS is compatible with most 3D models, even if they can not use canvases created with the program itself. The official website [13] is learning, downloading platform and a functional database for ThreeJS.

Unlike the Blend4Web Web Viewer, the ***Online 3D Viewer*** [4] application is not intended to be deployed on web pages but can only be used to view 3D models. It is based on ThreeJS and JSModeler (a JavaScript frame) and can be used to display 3DS, OBJ and STL formats. 3DS and OBJ are the most used 3D file formats and STL are typically used for 3D printing.

P3D.in is a utility [11] for uploading, viewing and sharing 3D models. It can be used online with .OBJ and can generate unique URLs. The most useful part is the HTML code that can be used on Web pages.

Sketchfab is online software [12] used to view 3D, VR, AR (Augmented Reality) content, or 3D content for mobile devices. You can also embed 3D code on any website, as well as the Sketchfab. Web downloaded models work on browsers and use 3D libraries to make real-time online rendering

GrabCAD is a Web-based industrial design tool used to display a 3D model made in AutoCAD or other 3D modelling tools [9]. GrabCAD can be used to create 3D models on the web and print them with a 3D printer. GrabCAD is used by professional designers, engineers, manufacturers and students.

XVL Webmaster is the AutoCAD online version used to view 3D models. It can also be used with the XVL player to view 3D models, but the player is offline, made for Windows. So, to include a Windows application on a web page, this is an option made with AJAX. You can also use Vaadin.com [6] to easily embed applications into webpages. In the Vaadin framework, the code is included in the user interface of the HTML or Java web page. It is a solution to create a web page with 3D objects embedded in the Vaadin application and then into the desired web page.

Blend4Web is a rendering engine that can be used in Blender to create a web-compatible file format such as .HTML or .XML. Blend4Web is used with WebGL because some file formats often fail to be recognized by browsers or cause errors. A good alternative to saving files in X3D format and then converting to software-recognized format is an online converter.

Any X3D file [7] can be successfully transformed into XML, HTML5 or XHTML5. Even if you can edit the XML or HTML5 code after rendering, you can not make complex animations with Blender. X3D files are formatted as 3D models and not as scenes. Although this is the least professional method to use, it significantly reduces the complexity of the online 3D scene. Blend4Web is also used today by professionals and large companies, including the PRO version of Blend4Web.

The latest Blend4Web version allows developers to include Blend4Web devices VR and AR [8]. For these devices, the professional method is JSON. This is actually the direct output of Blend4Web [5]. In the future, it will be possible to produce a direct HTML output, a more convenient way to use it. Another method that can be used to turn the web page into an interactive 3D application uses Blend4Web by directly accessing the online renderer through JavaScript. Currently, this is a good implementation of 3D scene rendering, but the content is not displayed directly at the page load. But there is also code that allows WebGL based rendering and uploads the JSON file.

Blend4Web Web Player is a way to view 3D models in a web application. It is easy to use in handling HTML and JSON files from Blend4Web renditions. The Web Player is part of Blend4Web, but the interaction options are limited.

3 Applications

Below is a detailed and detailed description of the Bell project with the technical drawing option in Blender. In the second part, we present the new 3DShark website, a 3D rendering and viewing interface for 3D objects, including the Bell project. The paperwork is based on [17] and continue the author work from [15,16,18].

The Bell app. 3D modelling in Blender is 2D thought process and transposed into the 3D world. The first step is to draw the project outline from all perspectives: front, back, right, left, top and bottom. For most models considered, the right and left are symmetrical and only a projective drawing for the side is required. Projects are drawn on paper or using the computer. Sometimes, only a static image that can be rotated to get a 3D model is useful (Fig.1).

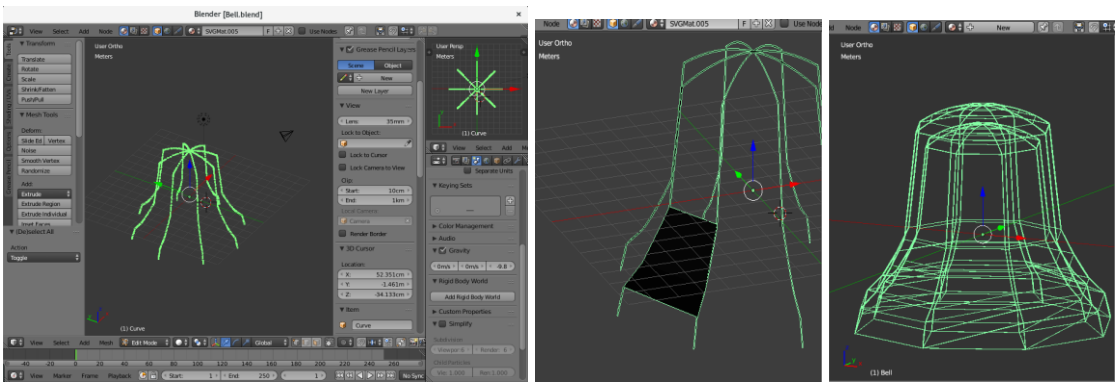


Fig. 1. Making the initial scenes of the Bell demo project in Blender
(a) Design screenshot (b) Detail view (c) Magnified view

The 3D bell model uses a 2D drawing in SVG format, for example with Inkscape. In Blender the final design of the product will be similar to Fig. 2. For greater accuracy the Screw function with a certain number of segments could be used. For 3D printing, an STL file is saved with the base Blender add-ons.

For the First Tech Challenge 2018 robotics project, the author was member of the TRIX team [14]. The logo and several components for the robot were made by the author in Blender and printed in plastic with a 3D printer using Cura software (Fig. 3, 4).

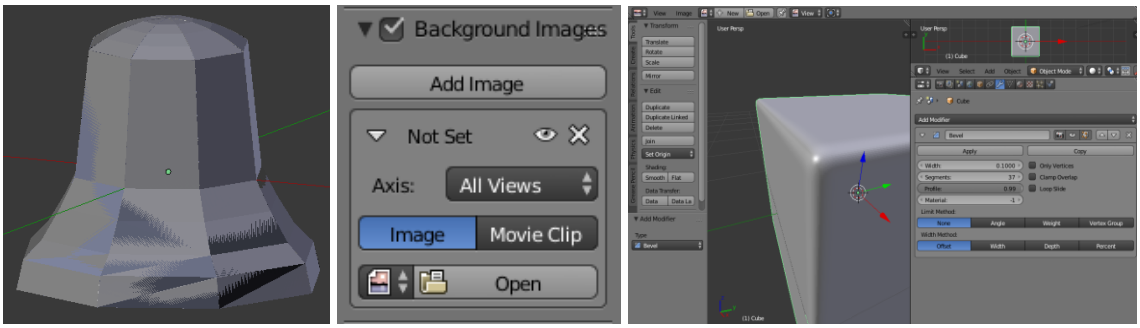


Fig.2: Bell project in Blender (a) Details of faces (b) Axel panel (c) Bevel function.

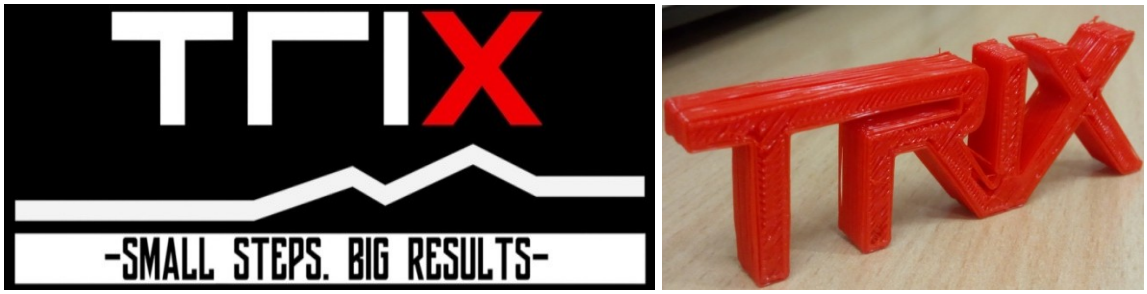


Fig.3: First Tech Challenge 2018 robotics project of the TRIX team in Blender and 3D model (a)Logo design (b) 3D printed Logo

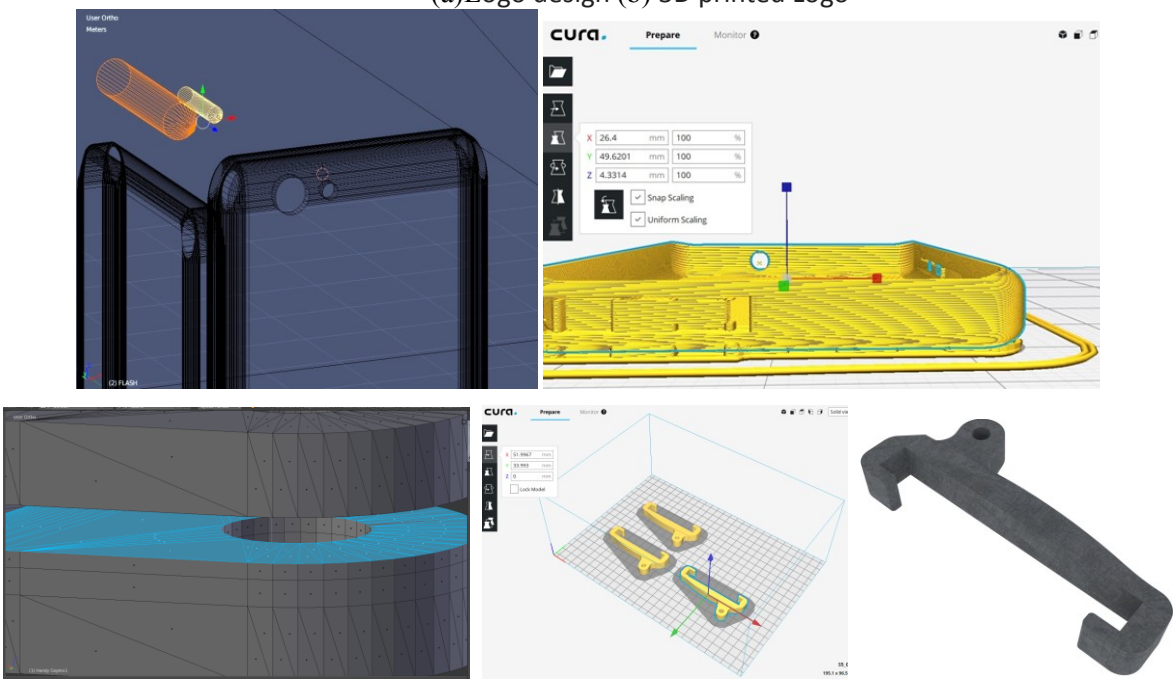


Fig.4: First Tech Challenge 2018: TRIX team project in Blender and 3D Printing using Cura Software (a) Design in Blender of a case to handle the device controlling the robot and (b) its model in Cura software, ready to be printed. (c) Blender Design of a component to handle the controller device of the robot used furthermore by (d) the Cura software to model the final (e) 3D real component.

For complex 3D models, several processes are involved.

- *Realization of projective drawings*. A projective drawing is a 2D drawing that represents the final appearance of the model from a certain perspective. It can be done in Blender by drawing the images at the exact scale. Upon completion, the Cycles Render rendering is performed in a superior quality with contrasting contours. Save with specific perspective names.
- *Create a new Blender project***.** Deleting default objects and applying the above settings. With Background Picture and Add image button (Fig.1), you select the perspective, the switch between the image, the video and the source file.
  The 3D model starts from a perspective. In Edit mode, the plan is extruded to draw another perspective. Blender has a Sculpt mode designed for artistic purposes, but to design Industrial Design models. Only accurate means of manipulating the 3D model will be used. Extrusion option is selected. The same applies for peaks and faces; proportional editing will be disabled. The process is repeated until the final 3D model is obtained.
- *Boolean Function* When creating perforations in the 3D model, create the object to be punched and select the object to be punched. In the Boolean function, select the object to be perforated and choose Operation: Difference. With Apply, perforate and remove the punch.
- *The Bevel function* is used to smooth out the edges of the object and include a piece, a 3D model, in a difficult position (Fig. 2).
- *The Array function* is used to multiply the created object and simultaneously print a set of small 3D models. Verify the constant offset or offset relative to the distance between objects on the axes. The file size allowed by most 3D printers is default of 128 MB.
- *The Solidify function* is essential for the object to become a real solid object. The thickness (thickness) of the object to be printed is set. Save the 3D model as a STL file for 3D printing. Plastic, metal and other materials can be made.

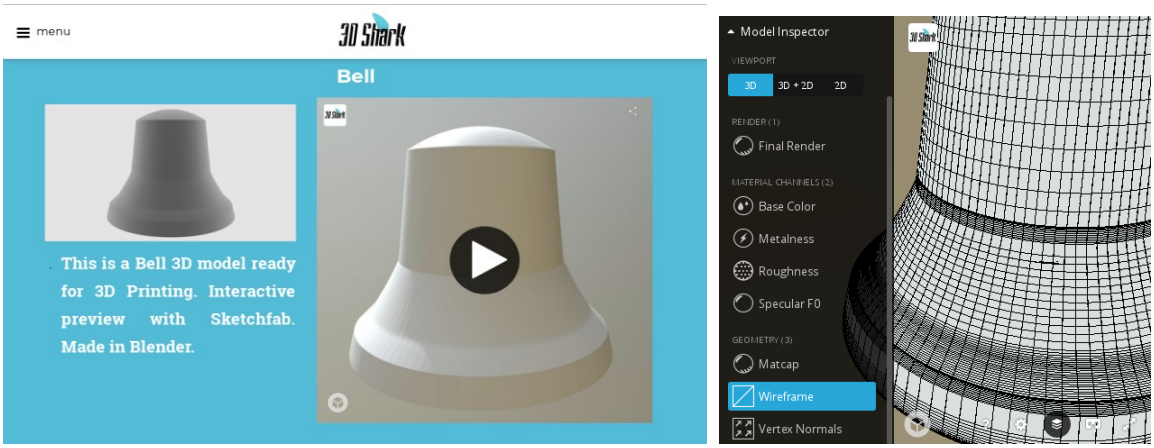


Fig. 5: 3D Shark, a new web rendering interface and collection of 3D models. Bell demonstration project in 3DShark [3]. (a) A page of the website [3] (b) Close view of the Bell project online

3D Shark is a new website [3] designed to include a collection of 3D models that can be viewed and support real-time rendering (Fig. 5). All models were made using the Blender 3D software.

Using the 3D Shark Website: The user chooses the 3D model to view from his or her own booklet or via the SketchFab website, where there are also stocked rendering templates. From here you can also download 3D models and embed HTML code into a web page.

Analog loads the Bell application on the website so users interact with it. For the 3D model, from the File menu, choose a format and save it. To include the 3D model, SketchFab is used, the model is loaded and the embed code becomes embedded and implemented on the website [3].

For a *professional solution* at Blend4web [5], download a free or paid package. In Blender, select the downloaded folder and restart Blender. Add-ons could be selected and the Blend4Web add-on checked. After modeling, the 3D model is saved as HTML or JSON. For SketchFab, check the specific add-on from the list. To store on other platforms, a folder is created including the textures. As the Blend4Web Web Player is embedded in HTML or JSON code, the files will be large, approximately 3500 code lines for a simple model. Rendering is done in real time, online.

4 Conclusions

The article describes the most used rendering online tools. A new website with interfaces of some specific tools to collect and render 3D images is also presented. The website shows also the new Blender projects based on the technical design specific features. The project also includes some components made for a robotics project. Among the possible improvements in the quality of 3D technical objects; we mention the use of Blender specific libraries in Python. The website will include interfaces with large libraries of objects and interfaces with various 3D printing options.

References

[1] Blender official website: *https://www.blender.org/*
[2] Learning WebGL online at: *http://learningwebgl.com*
[3] A. Pintea, 3DShark website: *http://3D-Shark.weebly.com*
[4] 3DViewer Online website presentation: *http://3Dvieweronline.com*
[5] Blend4Web website: *https://www.blend4web.com/en/*
[6] Vaadin framework website: *http://vaadin.com*
[7] Instant reality website. X3D Converter: *http://doc.instantreality.org/tools/x3d_encoding_converter/*
[8] Blend4web Adds Augmented Reality Support: *http://www.cgchannel.com/2017/07/blend4web-17-06-adds-augmented-reality-support/*
[9] GrabCAD website: *https://grabcad.com/*
[10] Embedding VRML *https://people.apache.org/~jim/NewArchitect/webtech/1998/02/vrml/index.html*
[11] P3d.in website: *https://p3d.in/*
[12] Sketchfab website: *https://sketchfab.com*
[13] ThreeJS website-Source codes: *https://threejs.org*
[14] First Tech Challenge Romania, Bucharest, 23-25 March 2018, RO028-TRIX, *https://natieprineducatie.ro*
[15] A. Pintea. *Making Animations in Blender*, Int. Conf. Applied Informatics ICDD 2017, 127-132 2017
[16] A.Pintea, *Model 2D Cartoons in Blender*, PCID 2017 Conference, 32-36 2017
[17] A.Pintea, *Industrial Design in Blender & 3D Web View*, Conference Computer Programming, Communication, Imagination, Design, PCID 2018, in press
[18] A.Pintea, *Spinning CD Video-Audio Sync: Blender&Kdenlive*, Conference Computer Programming, Communication, Imagination, Design, PCID 2018, in press

Alexandru PINTEA
National College E.Racovita
Mathematics & Informatics
9-11 Kogalniceanu, Cluj-Napoca
ROMANIA
E-mail: alecpintea@yahoo.com

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Digital interactions based on brainwaves signals

Robert Săndică, Petrică Bota

Abstract

Ever since their invention computer processing power has increased dramatically with each passing year. The processing power of a regular computer is much greater than it was 20 years ago but the capabilities of the person that interacts with the computer are relatively the same as they were 20 years ago. We as users are still limited to pushing buttons, scrolling or touching screens in order to communicate with the computer. The aim of this article is to propose a new solution to the issue of data input. We consider that by using a neural reader that directly reads the user brainwaves we are able to drastically reduce the time reqiured to input commands into the computer. This could be of help especially to users with limited range of movement. As proof of concept we created an application that reads and interprets the user‟s brainwaves in order to complete certain actions. We have found that it is possible to transform the user‟s brainwaves into digital signals that can operate real devices.

Keywords: data input, brainwaves, neural reader

1. Introduction

Moore‟s Law [6] states that the processor speeds and the overall processing power of computers doubles every two years. This phenomenon is happening every two years since 1970 when Moore first observed this trend and published his findings. .

Not much has changed for the typical user since 1970. In order to communicate commands to the computer we still push buttons or move cursors. The only recent advance in data input is represented by the popularity and usage of touchscreens [7]. The touchscreens have improved the number of widgets that we can insert on a limited surface and the general speed of inputting data. This improvement is welcomed but we consider that with the help of the neural reader we can further improve the user‟s capabilities of communication.

Our goal is to interpret the user‟s brainwave activity as a type of data input. We consider that if the brainwave signals are interpreted correctly and reliably it can be the fastest method of inputting data into a computer.

We chose to study the viability of using a neural reader as an input device for the following reasons:

- The technology required to read neural activity has improved greatly in the recent years
- The devices used for reading brainwaves are becoming more accessible
- We desire to improve the speed of communication between users and computers

- We want to provide an alternative input device for people with motor disabilities

The rest of the article is organized as follows. In Section 2 we present the principle of brainwave interpretation and the method that is used to study brainwaves. Section 3 presents the structure of an application that is able to transform brains signals into digital signals that the computer is able to process. In Section 4 we detail how our application is able to receive, transfer and interpret the brainwaves. Lastly, in Section 5 we present the findings and the conclusion of our research.

2. Background

For this article we used the Electroencephalogram principle [8]. Electroencephalography (EEG) is a noninvasive monitoring method used to collect information on the brain‟s electrical activity [1]. The brain‟s signals are recorded by electrodes placed on the scalp. The electrodes are coupled to a device called electroencephalograph. EEG measures the electrical fluctuations resulting from the brain‟s neurons. The first human EEG was recorded in 1924 by physiologist and psychiatrist Hans Berger. Berger also invented the electroencephalogram (giving the device its name) [1].

A normal medical EEG typically lasts around 30 minutes and it involves recording from scalp electrodes. EEG is used by doctors to identify epileptic seizures, psychogenic non-epileptic seizures, fainting, movement disorders, migraines and to determine whether to prescribe anti-epileptic medications. EEG is also used to monitor how certain sedatives or anesthesia affect patients that are in an induced coma. There are cases where EEG is used to monitor brain hemorrhages (this is still being researched).

EEG is used considerable in cognitive science, cognitive psychology, neuroscience, psychophysiological research and neurolinguistics. As EEGs are becoming more widely known they are being used in research and treatment for mental disabilities such as attention deficit hyperactivity disorder (ADHD) or auditory processing disorder (APD) [2].

Reading the brain activity means, getting the human brain in a certain state and read the low current signals emitted at the surface of the scalp. The brainwaves emitted by the human brain are: Infra-Low (<0.5HZ), Delta waves (0.5 to 3 Hz), Theta waves (3 to 8 Hz), Alpha waves (8 to 12 Hz), Beta waves (12 to 38 Hz) and Gamma waves (38 to 42 Hz).

Infra-Low brainwaves, are the basic cortical rhythms that are accountable for our higher brain functions. Not much is known about infra-low brainwaves. Due to their slow nature they are difficult to detect and accurately measure, therefore not many studies have been done. They appear to take a major role in brain timing and network function.

Delta brainwaves (Fig 1.) are slow, loud brainwaves (low frequency and deeply penetrating, like a drum beat). Delta brainwaves are usually produced when a person has a dreamless sleep. Delta waves dampen external stimulus and are the source of empathy. In this state healing and regeneration are accelerated, and that is why deep sleep is necessary for the brain‟s healing process.

Theta brainwaves (Fig 1.) mostly appear during deep meditation or during deep sleep. Theta brainwaves play a huge role for intuition, learning and memory. In theta, our senses are withdrawn from the external world and focused on signals originating from within.

Alpha brainwaves (Fig 1.) usually appear during slow, quiet thoughts and in light meditative states. Alpha is „the power of now‟, being here, in the present. Alpha is the default state for the brain. Alpha waves aid mental coordination, calmness, alertness, mind/body integration and learning.

Beta brainwaves (Fig 1.) are most present during our normal awake state of consciousness when attention is directed towards cognitive tasks and the outside world. Beta is a „fast‟ activity, present when we are alert, attentive, engaged in problem solving, judgment, decision making, or focused mental activity.

Gamma brainwaves have the highest frequency (similar to a flute). The purpose of gamma brainwaves is to process information from different brain areas. Gamma brainwaves pass information quietly and rapidly. The most subtle of the brainwave frequencies, the mind has to be quiet to access gamma.

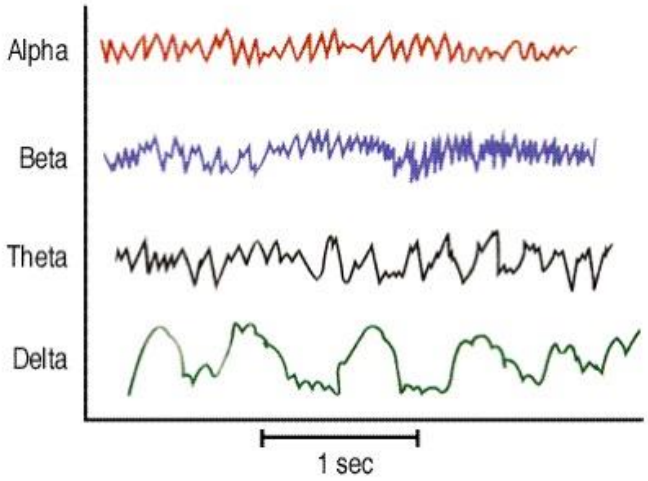


Fig 1. Brainwave lengths

Our brainwave profile and our daily experience of the world are inseparable. When our brainwaves are out of balance, there will be corresponding problems in our emotional or neuro - physical health. Research has identified brainwave patterns associated with most types of emotional and neurological conditions [5]. Over-arousal in certain brain areas is linked with anxiety disorders, sleep problems, nightmares, hyper-vigilance, impulsive behavior, anger/aggression, agitated depression, chronic nerve pain and spasticity.

3. Design of an application using brainwaves as data input

We consider that the user‟s capabilities of data entry are far smaller than the computer‟s processing power. The purpose of this article is to research an alternative data input method. We desire to limit the time delay between the user thinking of an action that he wants to transmit to the computer and when the computer interprets the command. As a result we created and application that directly reads and interprets the user brainwaves as commands for the computer. We consider that interpreting brainwaves as data input is the fastest type of user to computer communication that we are able to implement. Our intention is to provide a faster type of data entry and another option for people with reduced motor activity.

We designed an application to demonstrate the possibility of using brain waves as a form of input to obtain a new way of interacting with the computer. The project proves the possibility of using a brain wave analyzer to get the raw data and transform it in some sort of information based on the user‟s state of mind. The main objective of the application is to prove that everyone can benefit from this technology mostly the disabled persons. Paralyzed from neck to bottom persons can gain some control over electronic devices by blinking or changing their mind state from relaxed to focused.

The main idea of the project is to detect the state of mind emanated from the user‟s brain. We focus on two primary states of mind, meditation and attention. To achieve the meditation state the user

must try to keep his mind clear of any thoughts. Breathing exercises can help achieving this state of mind. The main characteristic of the meditation state happens when Delta and Alpha waves are predominant but the frequencies of the waves are low. This state stops when the user loses the clear of mind.

The attention state of mind comes when the user has it mind occupied with problem solving, or bringing back intense memories. The main characteristic of the attention state happens when the Beta and Gamma waves are predominant the user is in an alert state of mind ready to solve problems and at sharp cognitive functions.

Our solution of reading and analyzing the brain wave activity is presenting in the following way: In order to read the raw brainwave data we used a EEG portable device called MindWave Mobile (Fig. 2) that is developed by NeuroSky [3]. The neural reader measures the output of alpha waves, beta waves, etc. It also measures attention, meditation and eye blinks. MindWave Mobile is made up out of a headset, an ear clip, and a sensor arm [Fig. 2]. The headset‟s reference and ground electrodes are on the ear clip and the EEG electrode is on the sensor arm that will be places on the forehead above the eye.

MinWave reader uses a single AAA battery with 8 hours of battery life. For a more technical analysis the neural reader outputs a 12 bit Raw-Brainwaves (3 – 100 Hz) with a sampling rate of 512Hz. After the MindWave headset read the brainwaves signals it transmits the data through a bluetooth emitter. We received the raw data with the help of the laptop‟s built in Bluetooth receptor. Once we have received the raw data we interpret it by using a C# library. After that data has been interpreted we are able to use it inside the Unity Engine.

We decided to use Unity Engine for the interpretation of the brain signals due to the fact that it provides an array of tools that we can use. The Unity Framework provides easy to animate 2D and 3D objects, meshes and built-in colliders. Also, you can create different environments quickly and effortlessly with the help of the Unity Engine.

Unity was used as a development environment because it accepts C# scripts for compiling. As a result it is convenient to read that interpreted data from the C# library inside the Unity Engine. To be able to interpret the data you need to have ThinkGear connector installed on your PC. It provides a "server", and sends data bytes from Mindwave to 127.0.0.1:13854 with a TCP protocol [4]. This means the data are accessible like a web server. So we can use a TcpClient in C# to access the data.

Fig 2. MindWave Mobile neural reader

4. Application’s interface and functionalities

The application proposed in this article uses the brain waves and interprets them in two categories: attention and meditation. The application demo uses the attention and the meditation factors to float a sphere as the user mantains one of the two states the sphere goes higher and stais

there until the user loses his focus. To connect to the device a bluetooth connection must be enstablished, in Fig 3 the picture prezents the button to connect to the device after the connection has been enstablished in Fig 4 can be observed the date gathered from the device

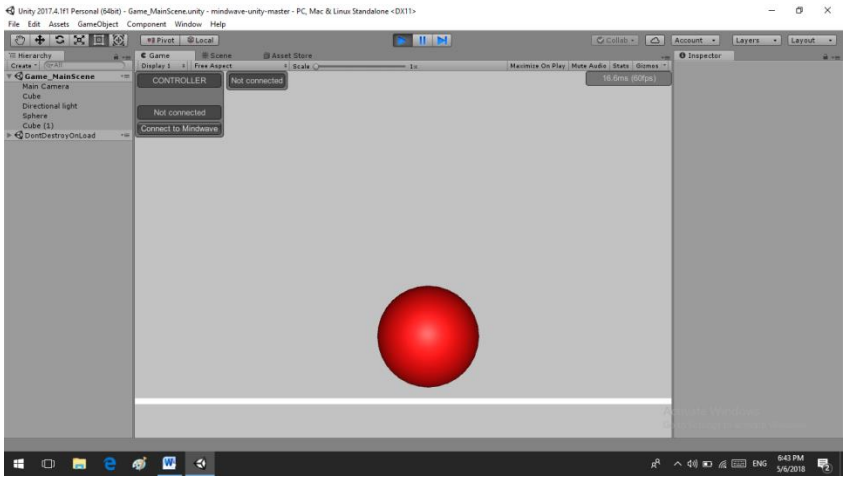


Fig 3 Connecting to the device menu

In Fig 4 the the menu presents from top to bottom the folowing :

- Senses section, displaying the attention and meditations factors
- Brain waves sections resposable with the displaying of the brain waves in the raw states unprocesed
- Others section showing the blinking intesiti and the eeg intensity allong with the disconect button

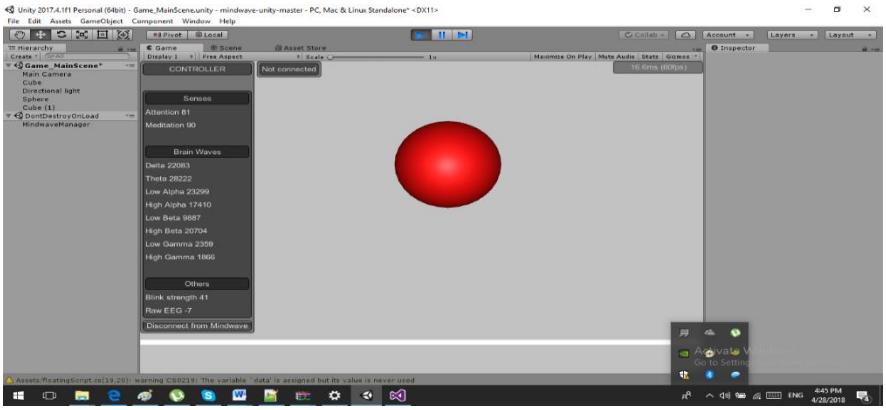


Fig 4. Sphere moved by brainwaves (in Unity)

The user need to have a level of atention over 70 to raise the sphere, if the atention level is under 70 the ball falls down. This functionality is realized using the following code C#:

```
atention = data.atention;
if (atention > 70)
        { transform.Translate(Vector3.up * 0.005f); }
else
        { transform.Translate(Vector3.down * 0.004f); }
```

Another deomnstrative functionality is the blinking intensity. For the brain blinking is like a shortcut of power it needs to adapt, and for a moment the brain has a shock, this shock is distinct everytime the user blinks, because of this we can count the number of blinks (Fig.5).

Fig 5. Blink counter function and result

By blinking the user changes the color of the ball from red to white and back. The following code realize the changing of the sphere colour:

```
if (blinkCounter % 2 == 0)
    { sphere.GetComponent<sRenderer>().material = material1; }
else
    { sphere.GetComponent<Renderer>().material = material2; }
```

To be able to enjoy the demo you need to follow these steps:

I. Relax your mind state
II. Put on the Headset and turn it on
III. Run the demo
IV. Connect to the Device from the demo
V. Wait 30 second to calibrate the device to your mind state
VI. Do some action like blinking or focus your mind on something and see the sphere rising

5. Conclusion

In conclusion, our application proves that it is indeed possible to interpret brainwaves into a digital format. From the transformation of brainwaves into digital actions could also benefit people with disabilities and limited motor actions.
It can also improve the range of input for a normal user.Furthermore, this technology can be improved upon so that the range of actions intepreted from the user‟s brainwaves can be expanded. For example it is theoretically possible to intepret the brainwaves of the user as data input for electronical devices or even projections of color.
Futher directions of study are oriented towards increasing the range of actions that can be intepreted. To this end we plan to further explore the existing technology and its capabilities.

Acknowledgement: This work was supervised by Professor Dr. Dana Simian, from “Lucian Blaga” University of Sibiu, Romania.

References

[1] Haas, L F (2003). "Hans Berger (1873-1941), Richard Caton (1842-1926), and electroencephalography". Journal of Neurology, Neurosurgery & Psychiatry
[2] Use of EEG to Diagnose (2015) ADHD Agatha Lenartowicz, Ph.D1,2 and Sandra K. Loo, Ph.D1,2
[3] NeuroSky homepage (accessed 29.04.2018) - https://store.neurosky.com/pages/mindwave
[4] Library required for brainwave interpretation (accessed 21.04. 2018) - http://developer.neurosky.com/docs/doku.php?id=thinkgear_connector_development_guide
[5] Brainwave types and correlated emotions (accessed 21.04. 2018) - http://harptherapycampus.com/wp-content/uploads/2011/01/EpsilonGamma.pdf
[6] Moore‟s Law (accessed 19.04.2018) - https://en.wikipedia.org/wiki/Moore%27s_law
[7] Touchscreen‟s history (accessed 19.04.2018) - https://en.wikipedia.org/wiki/Touchscreen
[8] Electroencephalogram principle - (accessed 20.04.2018) http://www.alpha-active.com/EEG%20info.html

Robert SĂNDICĂ
“Lucian Blaga” University of Sibiu
Faculty of Sciences
Dr. Ratiu Str 5-7
ROMANIA
e-mail: sandica_robert@yahoo.ro

Petrică BOTA
“Lucian Blaga” University of Sibiu
Faculty of Sciences
Dr. Ratiu Str 5-7
ROMANIA
e-mail: botapetre1@gmail.com

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

FRELaw Dictionary - a Trilingual Glossary of Law Terms

Mihai Sandu

Abstract

This article describes the elaboration of the trilingual French-Romanian-English dictionary of legal terms (source language: French). The dictionary aims at filling a gap in the landscape of available specialised dictionaries, targeting this three-language combination for the very first time, to the best of our knowledge. The electronic resource we intend to build will be used for a wide range of target groups:

(a) translators - who will use it as a working tool;

(b) linguists - disambiguation studies, development of automated translation software, linguistic studies for French, Romanian and English language;

(c) legal experts - terms are carefully classified according to their various meanings;

(d) students from different study areas: law, translation and philology - as a study device.

The dictionary was created in accordance with the best practices regarding the construction of electronic databases and the suitable structuring of terminological records. To this purpose, a framework was deployed so that the information required for the terminological sheets would be collected (electronic dictionaries, specialised e-catalogues, etc.) by qualified individuals. The goal is to provide content with a high degree of expertise, as well as ensure the required dimensions so that the dictionary may become a benchmark in the legal field.

1 Introduction

We are aware that terminology is part of our socio-professional communication. As Maria Tereza Cabre states, "it is practice, the most lively part of a language"s vocabulary, because it is confused with learning and the creativity of sciences and techniques; with learning when it comes to absorbing the past combined with a terminological tradition and with creativity when their dynamism leads to creating new notions and thus new terms". For the development of transversal skills, the students of the first year of the MA programme *Legal translation and terminology* through the academic years 2014-2017 at the University of Craiova, were given the task to contribute to the creation of a dictionary of legal terms. We have gathered all the resulted legal terms in a multilingual dictionary in electronic format. At the moment, the dictionary includes a parallel corpus of terms in Romanian, French and English.

The main goal of this dictionary is to complete the offer of specialized dictionaries, being a digital resource of legal terms in at least three languages. We point out that the elaboration of this

legal dictionary is aimed at creating a linguistic resource which emphasizes not only the lexical content, but also and the semantic information.
Its usefulness can be proven by enumerating the target groups to which the dictionary is addressed:

- Translators – the dictionary can be used as support in translation tasks;
- Linguists – it may serve as research and documentation material for compared linguistic studies of Romanian, French and English, including for the analysis of word semantics or for the development of automated translation programmes;
- Legal advisers, lawyers, public notaries, etc. - who can check the classification of the legal terms according to their different meaning (the contexts in which the terms are used);
- Students in philology (especially, but not only) - as a learning device.

Working at this project is itself beneficial since it qualifies the students for all the domains in which legal text comprehension is necessary, as well as for contrastive studies of these texts for the languages represented in this dictionary.

The specialists in terminology and translation are aware of the difficulty of correctly defining certain terms that are part of the basic vocabulary, especially when they investigate languages which are linguistically related, such as French and English. These languages have indeed a significant number of similar terms due to numerous cultural exchanges, and not only, which might seem to facilitate the translation at first, even without using a dictionary (for example, the English words “government” and “administration” and their French equivalents “gouvernement” and “administration”).

2 FRELaw Dictionary

2.1 The Working Plan

With this dictionary of legal terms, we propose the investigation of various aspects regarding the terminological comparative study of the three languages included here, and we equally allow specialists to extend their studies, since this resource is intended to be made available for the public the moment it included a significant number of words. More precisely, we want to investigate terms with various meanings from each of the three languages and, at the same time, the semantic overlapping or the lack of perfect semantic overlapping between the translations.

So far, one thousand terminological entries have been included in this dictionary 1000, all drawn up by the students of the MA programme *Legal translation and terminology* at the University of Craiova, during the courses of Legal terminology and Data bases.

As previously mentioned, the lexicographic resource presented in this paper targets three languages, two used internationally, English and French, as reference languages of the European area, and the third, our mother tongue, Romanian. As far as we know, this combination of languages is a novelty for the specialized resources available at the moment.

The format of the terminological entries was established in accordance with the approaches used in academic research, namely in the works of renowned specialists, such as Maria Teresa Cabré, Dqniel Goudec and Gérard Cornu.

The working plan drawn up and implemented for the creation of lexical resources consists in five stages, which are described as follows.

2.1.1 Compiling a trilingual corpus of compatible texts: establish the annotation method and the structuring of the texts

Any terminological study has to be based on relevant information from a specialized domain and in our case, this meant creating a collection of comparable texts in the three languages. These texts were compiled in three collections of authentic texts (in all the three languages) linked through *tertia comparationis*: the target domain (the legal domain), the text types (scientific presentations, terminological course handouts etc.), the degree of the specialized meaning (relevant texts of professional discourse) thus obtaining the reference corpus of the dictionary.

The next stage was the annotation of the texts according to the content level and syntactic tokens, in order to assure a common standard that would subsequently allow for a semi-automated comparison of the texts in the three languages. Moreover, these annotations could facilitate the identification of the legal terms which appear in texts as noun phrases. The following syntactic parsers were used:

- for English texts: Stanford Parser http://nlp.stanford.edu/software/lex-parser.shtml
- for French texts: TreeTagger for French, http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/
- for Romanian texts: NP Chunker for Romanian, http://nlptools.info.uaic.ro/WebNpChunkerRo/

2.1.2 Establishing the extraction parameters of the terms: term recognition criteria and interlingual correspondence

The corpus thus created passed through a common phase of exploitation, at the of end of which a set of parallel legal terms was obtained for the three languages taken into consideration. The terms were accurately extracted by using monolingual legal dictionaries, such as:

- for English - A Dictionary of Low, edited by Jonathan Low, Oxford University Press, 2015,
- for French - Dictionnaire du droit privé, Serge Braudo, (https://www.dictionnaire-juridique.com/lexique-juridique.php) and
- for Romanian - Drept online (http://www.dreptonline.ro/dictionar_juridic/dictionar_juridic.php).

Thus, we search for a way to duplicate the value of the dictionary: it represents an attested resource for the extraction of neologisms, and, additionally, by including information regarding collocations and contextual occurrences in the terminological files, the dictionary provides the background for cognitive and discourse analyses between the three languages.

2.1.3 Structure of the terminological files

As already stated, the organisation of the dictionary is in the form of terminological files, one file for each word of the three languages included in the dictionary.

The purpose of this stage is to determine the structure of the terminological files based on which the students introduce the data in the dictionary. This structure was rigorously analysed by specialists in the linguistic/terminological/translation and by specialists in the legal domain (the nature of the courses at the MA programme in *Legal terminology and translation* provided the experts in these domains, namely the professors teaching at this specialization).

This stage is compliant with the current standards which stipulate that beyond the issue of definition, a terminological analysis has to take into consideration all the aspects regarding the discourse use of these terms (valuable for legal translations and analyses). The advantage of this data structuring in the terminological files is due to the power resulted from the combination of

information referring to the same term in the three languages, but also to the creation of a framework for the identification of collocations owing to the contextual information that accompany the meaning/meanings of each term included in the dictionary.

2.1.4 Editing and validating the terminological files

The aim of this stage is to edit the content of the terminological files. At the moment, the dictionary has 1417 terminological files in French, English and Romanian, so 4251 terminological entries were recorded.

2.2 Examples and Statistics

Below you shall find an example of a terminological sheet for each language.

Table 1. Example of a French terminological record

ID	1
Limba [Language]	francais
Termen [Term]	litige
Categorie sintactică [Part of Speech]	n.f.
Neologism	TRUE
Sigla [Abbreviation]	-
Antonim [Antonym]	accord
Familie derivațională [Derivation Family]	litigieux
Sinonim [Synonym]	conflit
Zona geografică [Geographical Area]	France
Cod [Code]	J.
Domeniu [Domain]	juridique
Subdomeniu [Subdomain]	droit penal
Definiție [Definition]	Contestation donnant lieu à procès ou à arbitrage.
Sursa definiției [The source of the definition]	http://www.larousse.fr
Context	Le requérant fait valoir à cet égard que la condamnation d'une partie à un litige à rembourser au Tribunal de la fonction publique des frais exposés par celui-ci, au titre de l'article 94 du règlement de

	procédure, ne saurait être fondée que sur des faits en rapport étroit avec l‟affaire en cause, et non sur des comportements prétendus de la même partie dans d‟autres affaires.
Sursa contextului [The source of the context]	http://ro.linguee.com
Notă tehnică [Technical note]	Contestation d'un genre quelconque : Le litige est en voie de règlement.
Notă lingvistică [Linguistics note]	-
Fraza [Phrase]	-
Autor [Author]	Jianu Geanina

Table 2. Example of a Romanian terminological record

ID	1
Termen [Term]	litigiu
Categorie sintactică [Part of Speech]	s.n.
Neologism	TRUE
Sigla [Abbreviation]	-
Definiție [Definition]	Conflict între persoane, instituții, state etc. care poate forma obiectul nui proces, unui arbitraj etc.
Sursa definiției [The source of the definition]	http://dexonline.ro
Context	Sentința irevocabilă, dată în soluționarea unui litigiu între un profesionist și un simplu particular.
Sursa contextului [The source of the context]	http://www.juridice.ro
Latin	litigium
Autor [Author]	Jianu Geanina

Table 3. Example of an English terminological record

ID	1
Termen [Term]	litigation

Categorie sintactică [Part of Speech]	n.
Neologism	TRUE
Sigla [Abbreviation]	-
Definiție [Definition]	1. The taking of legal action by a "litigant. 2. The field of law that is concerned with all contentious matters.
Sursa definiției [The source of the definition]	Oxford Dictionary of Law
Context	Is the proposed agreement creating a unified patent litigation system (currently called the „European and Community Patents Court") compatible with the provisions of the Treaty establishing the European Community?
Sursa contextului [The source of the context]	http://www.linguee.com
Latin	litigium
Autor [Author]	Jianu Geanina

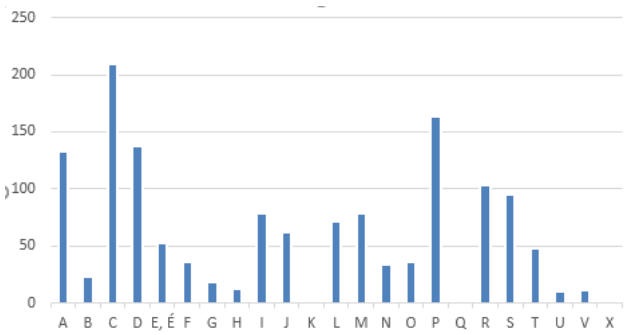


Figure 1. The French terms grouped upon the first letter

As far as the French language is concerned, the quantitative analysis of glossary terms grouped upon the first letter is given in Figure 1.

3 Conclusions

Our experience acquired during the elaboration of the **FRELaw Dictionary - a Trilingual Glossary of Law Terms** have enabled us to draw the following conclusions:

- the *actional perspective* brings forward the idea of *tasks to fulfil*, the student being seen as a social actor who is able to make use of all the skills and resources s/he has, thus becoming equally able to meet all the requirements related to transversal skills (working

autonomy, time management skills, ability to work in teams, cooperative learning, self-assessment skills);

- the theoretical knowledge in the field of legal terminology alongside with practical computer skills clearly lead to the creation of a pre-vocational climate and, thus, helps prepare students to meet labour market requirements;
- the linguistic inquiry to which the terms included in the dictionary are subjected to contributes to our understanding of the specificity of the legal language, both in terms of their origin and our awareness of the similarities and differences between the three linguistic systems

Acknowledgement: This work was supervised by Professor *Cristiana-Nicola Teodorescu*, from *University of Craiova, Romania* and Associate Professor *Mihaela Colhon*, from *University of Craiova, Romania*.

References

[1] Bercea, Raluca, Chermeleu, Adia-Mihaela, Français juridique, Bucuresti, Lumina Lex, 2010
[2] Cabré, Maria Tereza, *La terminologie. Théorie, méthode et application,* Armand Colin, Les Presses Universitaires d'Ottawa, 1992.
[3] Colhon, Mihaela, A Contrastive Study of Syntactic Constituents in English and Romanian Texts. In: Proceedings of the Workshop "Language Resources and Tools with Industrial Applications", Adrian Iftene, Diana-Maria Trandabas (eds.), ISSN 1843-911X, pp. 11-20, Editura Universitatii Alexandru Ioan Cuza din Iasi, 2011
[4] Cornu, Gérard, (dir.), *Vocabulaire juridique*, Paris, P.U.F. 2007
[5] Gémar, Jean-Claude, Les enjeux de la traduction juridique. Principes et nuances, Université de Genève, Ecole de traduction et d'interprétation, 2000
[6] Irimia, Elena, EBMT Experiments for the English-Romanian Language Pair. In: Recent Advances in Intelligent Information Systems, pp. 91-102, 2009
[7] Schmidt-König, Christine, *Introduction à la langue juridique française*, Munich, Nomos, 2006.
[8] Teodorescu, Cristiana, Colhon, Mihaela, Creation of Electronic Linguistic Resources. The Acquisition of Transversal Skills by MA Students in Letters. In Monica Tilea, Oana-Adriana Duță, Jón Freyr Johanssón, Patrick Murphy (Editors) Transversal Skills Development in Modern Teaching Practice. A Good Practice Guide, 171-181, Pro Universitaria București Publisher ISBN 978-606-26-0371-7, 2015

Mihai SANDU
University of Craiova
Department of Computer Science
Craiova, A.I.Cuza Street, no. 13
ROMANIA
E-mail: mihai1996sandu@gmail.com

Cristiana-Nicola TEODORESCU
University of Craiova
Department of Romance and Classical Languages
Craiova, A.I.Cuza Street, no. 13
ROMANIA
E-mail: cteodorescu05@yahoo.fr

Mihaela COLHON
University of Craiova
Department of Computer Science
Craiova, A.I.Cuza Street, no. 13
ROMANIA
E-mail: mcolhon@inf.ucv.ro

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

Information Security Awareness: It's Time to Change Minds!

Andreas E. Schütz

Abstract

In the age of digitization, the importance of information and information technology and thus of information security is increasing. Employees get more and more targeted by attackers and therefore need to be fully informed about their role for securing a company's information assets. Measures for raising the security awareness are often limited to informing employees about compliant behavior towards information security. However, employees must also be motivated to apply this knowledge to their situation and must be able to do so. The Integrated Behavior Model from the field of social psychology explains which factors cause a person to show a certain behavior. In addition to knowledge, these factors are, for example, intention, habit and environmental constraints. This paper applies the Integrated Behavior Model to the context of information security. It shows the complex interplay of several factors that determines an employee's compliant behavior and how companies can specifically target individual factors. With that, companies are empowered to actually sensitize their employees to information security rather than just conveying knowledge.

1 Motivation

Digitalized processes and digital business models are becoming increasingly important for companies to maintain and expand their competitive advantage. Digitization offers opportunities, but also entails new risks. With the increasing use of information technology in companies, the demands on information security are rapidly increasing. Companies cannot rely on technical security measures only to ensure the three key characteristics of security, known as confidentiality, integrity and availability. Most attacks nowadays target the user or so called "human factor" [1, 2, 3]. The user is exploited as a "vulnerability" by phishing, malware or social engineering attacks [4]. In the case of phishing, the majority of companies even face attacks on a daily basis [4]. Therefore, the effectiveness of security measures is highly dependent on the involvement of employees [5]. Nevertheless, only 10% of all scientific articles in the field of cyber security address the human factor [6].

To support security within companies, a holistic approach is necessary, actively involving the whole organization [5]. Employees should become a pillar in the information security concept through tailor-made sensitization [1] and thus support the company's digital strategy. Security awareness activities increase the employees' awareness towards information security and motivate them to securely behave in their workplaces while working with sensitive company information. This will be particularly important, in the light of the General Data Protection Regulation

(GDPR), which became enforceable on 25 May 2018, as the GDPR places particular emphasis on the protection of personal data.
Currently, transferring knowledge about information security and training employees are the prevalent security awareness measures in companies [7]. Research findings from social and health psychology, however, show that a complex network of various factors determines the behavior of a person, beside his knowledge. For example, an employee may know that using a third-party USB flash drive may result in the unintended installation of malware on a company computer. But this does not mean that she refrains from this behavior. Reasons for that include: The employee estimates the risk smaller than it actually is, she has no other opportunity to exchange data or she observes that other employees also use USB flash drives.
This paper examines how companies can actually make their employees aware of compliant behavior regarding information security. We explore research findings from social and health psychology with an exploratory approach. The Integrated Behavioral Model (IBM) [8] provides insights into the factors that influence a person's security-compliant behavior. This paper proposes techniques and measures that companies can use to address these factors.
The rest of the work is structured as follows. The next chapter discusses the topics security awareness and social psychology. Chapter 3 introduces the IBM and the factors that are able to influence human behavior. We then describe how knowing these factors can help to create tailor-made security awareness campaigns. We present techniques that can influence the behavior of employees and illustrate their effective application. The final chapter discusses the results and gives an outlook on future work.

2 Related Work

2.1 Security Awareness

Security awareness has established itself as a separate research area within information security. It targets the "human factor" and how IT users can be brought to an information security-compliant behavior. IT users should be motivated to use their theoretical knowledge about information security in practice [9, p. 121] and should be convinced of the importance of their actions [10, p. 193]. In practice, information security awareness campaigns mainly do one thing [7, p. 23]: In lectures, employees receive theoretical knowledge about information security. However, the actual behavior of an employee is hardly influenced by classical training [11].
Hänsch & Benenson [12] describe three possible perspectives of the term security awareness:

1. Employees know, which threats exist and recognize them ("perception").
2. Employees further know, how to protect themselves against threats ("protection").
3. Employees know what a threat is, what they can do about it and that they behave accordingly ("behavior").

In the end, only the third perspective – employees that behave compliantly – promises an actual increase in information security within the company. Awareness-raising means that employees know how to behave in compliance with information security (e.g., choosing a secure password), what consequences they and the company may face in the event of non-compliant behavior (e.g., loss of image and financial loss due to loss of customer data) and that they actually apply this knowledge in critical situations.
Helisch & Pokoyski [13] name organization as an additional aspect of security awareness. The organization ensures that employees in the company are able to behave in compliance with information security, i.e., no barriers exist, which are in conflict with compliant behavior. For example, a barrier is the password change link which is hidden in the depths of the company intranet. At the same time, organizational measures, such as increasing usability of applications, can support information security. Information security awareness is thus an interaction of

cognition (understanding of the problem and the knowledge to solve it), intention to act (will of the employee to behave in accordance with information security) and the organization [13].

2.2 Social Psychology

Research in the field of information security often focuses on technical issues such as backups, intrusion detection systems, firewalls, and secure software development. The focus on humans requires a different perspective. The field of social psychology can help to understand human behavior in terms of information security [14]. In particular, in the intention to act ("I want to act compliant with information security guidelines."), psychological processes in the minds of employees play a major role. Kabay [14] gives an overview of various socio-psychological issues, such as beliefs, attitudes or social norms that can be applied to information security in companies.
As a research field within social psychology, health psychology has been trying to explain the behavior of people for many years. It investigates how unwanted behavior can be changed. For example, researchers investigate smokers or people who eat unhealthy food. Over time, several theories and models have emerged. These behavioral models of health psychology are used by several papers in the context of security awareness [15; 16; 17]. The authors apply individual findings from those behavioral models in order to justify hypotheses in their own research model. Most frequently, the Theory of Reasoned Action (TRA) [18] and the Theory of Planned Behavior (TPB) [19] are used for this purpose [20]. The big advantage of the TPB is its universality [21].
The IBM (cf. Figure 1) described by Montaño and Kasprzyk [8] interprets insights from different behavioral models and theories and extends TRA and TPB by important factors, such as knowledge or habits of a person. We use the IBM in this work to explain information security-compliant behavior of employees.

3 It's Time to Change Minds!

3.1 Behavioral Factors

In order to increase information security awareness and to change employee behavior in the long term, companies need to create tailor-made security awareness campaigns. That is, campaigns that address specifically the actual beliefs of the employees and the individual general conditions within the company. According to the IBM, human behavior is directly influenced by five factors (see Figure 1): knowledge and skills, salience, habit, intention, and environmental constraints. Thus, the model covers all elements of the definition of security awareness: cognition is represented by knowledge and skills, salience and habit; intention is an own factor and the environmental constraints are responsibility of the organization.
The factor *knowledge and skills* is important for the execution of behavior: A strong will alone is not sufficient if the required knowledge for implementation is missing [8]. A user who does not know how to choose a secure password cannot behave in a compliant way, though she may want to do so.
Salience is present when something stands out from the immediate environment [22]. A salient behavior must be prominent for a person, so the person implements her behavioral intent and acts. If an employee gets an USB flash drive from a supplier, the employee must still remember the correct behavior for dealing with third-party devices.
A *habit* is intensified by the repeated execution of a behavior [23]. Locking the screen when leaving the workplace is an example of a frequently performed activity that can quickly become a habit. While classical training often only focuses on knowledge and skills, the IBM shows that the other two cognitive factors, salience and habit, also affect the employee's behavior.
The *intention* is much more complex and in turn influenced by several factors. It is also the strongest factor for influencing behavior [18]. Whether an employee forms a behavioral intention depends on her attitude to the specific behavior, her perception of the norms in her social working

environment and her assessment of her personal capacity to act. The attitude results from the experiential attitude ("What have I experienced while performing the behavior in the past?"), which is influenced by feelings, and the instrumental attitude ("What are the consequences of the execution of the behavior?"), which is affected by beliefs regarding the effects of the behavior. The perceived norm is also subdivided into two areas: the injunctive norm reflects the person's beliefs about what behavior his social environment expects of her, while the descriptive norm describes the beliefs of how the environment itself behaves. If a manager expects her staff to lock the screen when leaving the workplace, but does not follow the rule herself, the employee's injunctive and descriptive norm will diverge. The personal agency is formed by the perceived control ("Is the execution of the behavior simple or difficult in view of the circumstances?") and the self-efficacy ("Do I dare to perform the behavior with my abilities?"). These, too, are created by the beliefs of a person. To indirectly influence behavioral intent, security awareness campaigns should therefore address the employee's feelings and beliefs [8].

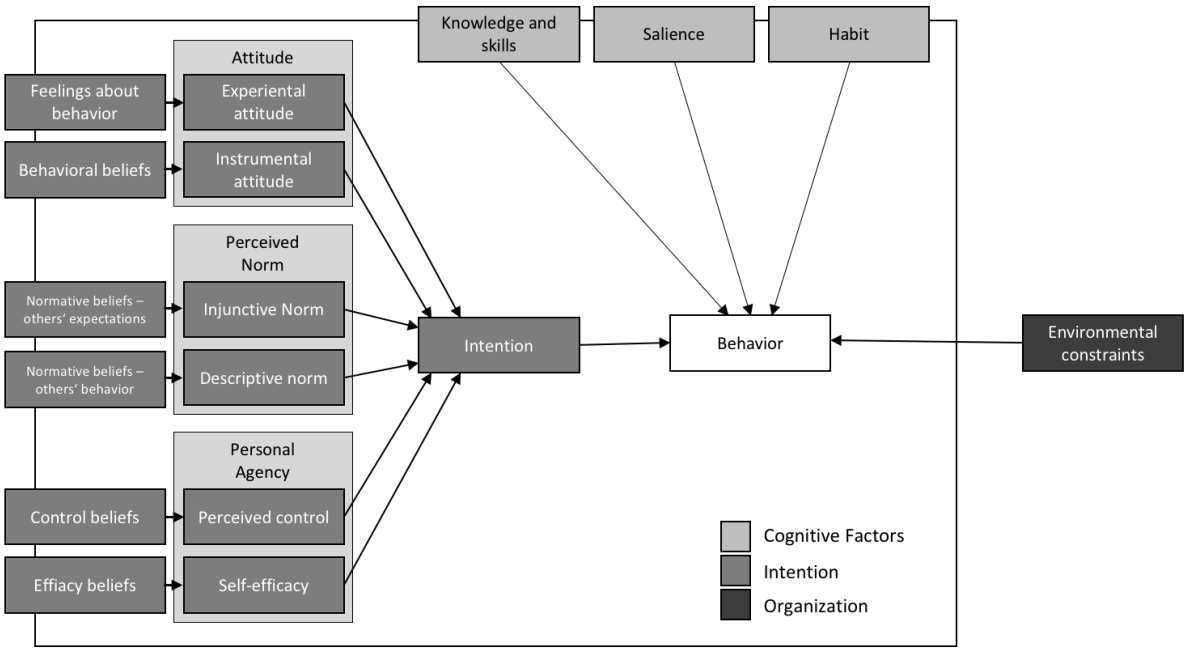


Fig. 1: Factors of the Integrated Behavioral Model (based on [8, p. 77])

The *environmental constraints* represented in the IBM directly affect a person's behavior. Constraints exist when, for example, an application prevents the execution of a compliant behavior by not offering employees the opportunity to choose a secure password. Constraints should be taken into account, and eliminated if possible, when an information security concept is created.

Factors such as attitude or behavioral purpose correlate more with actual behavior when they are at the same generalization level [24, p. 45; 21, p. 30]. Moreover, the more specific the factors, the better they correlate [24, p. 45]. Instead of propagating a general goal, such as increasing information security [21, p. 31], awareness-raising actions should influence specific attitudes and beliefs of employees. Therefore, in order to use the IBM in information security, it has to be instanced to a specific information security-compliant behavior, such as "compliance with password rules". Then, the factor "knowledge" refers to the knowledge of password rules.

3.2 How Convinced are the Employees?

Before companies can start influencing the behavioral factors of their employees within the context of an information security awareness campaign, they must carry out an as-is analysis. The aim of this analysis is to find out how strong the individual factors are and which emotions or beliefs prevail in the company regarding information security-compliant behavior. Montaño and Kasprzyk [8] recommend carrying out a qualitative study in form of interviews. The interviews identify the most salient problems. 15-20 employees of each target group, such as administrative or sales staff, should be interviewed to obtain the following information [8]:

- Experiential attitude: What positive or negative emotions exist regarding the behavior?
- Instrumental attitude: Which positive or negative attributes or outcomes result from the behavior?
- Normative influencers: Which influential individuals or groups support the behavior or are against it?
- Control beliefs and self-efficacy expectations: What situational barriers or supportive factors hinder or support the behavior.

From the findings gathered in the interviews, a questionnaire for a quantitative evaluation can be prepared [25]. A large portion of the employees within the company should answer it. To compose the questionnaire, the beliefs collected in the first step are generalized, transformed into questions and assigned to the four factors knowledge and skills, salience, habit and intention [25, pp. 150f]. A question about the normative beliefs could be: "My manager influences me in my working with USB flash drives". A behavioral belief might look like this: "Locking my screen is unnecessary because I have no important data on my computer." Employees respond to a five-point scale ranging from "I disagree" to "I agree" or similar [26, p. 113]. It is also possible to query the factors knowledge and skills as well as habit (e.g., through the "Self-Report Habit Index" [27]). In order to be efficient, the survey should be computer-aided, and employees should handle it within a reasonable time (e.g., 15 to 20 min). The computer-aided analysis of the survey results provides information about the manifestation of the respective behavioral factors in the company. The results serve as the basis for planning a security awareness campaign with individual, tailor-made content.

3.3 Create Convincing Campaigns

In order to persuade employees to adopt information security-compliant behaviors, a security awareness campaign should focus on the individual factors, taking the survey results into account. This chapter examines the factors that can be affected by "classical" security awareness techniques and measures and how they can be used most effectively.

In its Baseline Protection Catalogues, the German Federal Office for Information Security recommends using techniques for sensitization, training, reinforcement and public relations [28, p. 1652-1655]. We divided these techniques into the two categories intensive classes (e.g., classroom training, lectures) and social marketing (e.g., flyers, posters, printed cups). The Information Security Management System (ISMS) ISIS12 [29] describes techniques that address the organizational aspects of information security awareness. These are the information security policy and the information security team. In addition to these, we consider supporting factors, i.e., factors that facilitate the performance of the behavior, such as usability optimization. Table 1 gives an overview of the various techniques as well as their factors influenced by them.

Techniques	Influenced factors
Classes	Knowledge and skills, habit, feelings, behavioral beliefs, efficacy beliefs, control beliefs, normative beliefs (expectations)
Social Marketing	Knowledge, salience, efficacy beliefs, feelings, normative beliefs (expectations)

Information security guidelines	Knowledge, efficacy beliefs, normative beliefs (expectations)
Information security team	Efficacy beliefs
Supporting factors	Control beliefs, efficacy beliefs

Table 1: Security awareness techniques and the factors influenced by them (Source: Own Illustration).

Classes, due to their flexibility, have the potential to influence almost all behavioral factors. Typically, classes aim at increasing knowledge and skills. With exercises and training, habit can also be influenced: The more often an employee repeats a behavior, the sooner a habit can establish [23]. For example, it can be trained that an employee habitually locks the screen when leaving the workplace. The behavior should be trained as close to reality as possible, for example with the help of role-playing games [30]. By practicing the behavior, even emotions such as fear, can be changed and positive beliefs are formed. In order to influence beliefs in classroom sessions, it is important to present tailor-made content to the attendees. The content must strengthen, weaken or change the beliefs in accordance with the desired behavior [31, p. 4]. For example, a person's belief may change if contradicting information is presented to her [14]. An employee who believes that the use of third-party USB flash drives is harmless could be confronted with counter-examples and the consequences of this action. The trainer can also initiate the formation of new beliefs by presenting new information [31, p. 5]. Declaring to an employee that she is awaiting sanctions for the consequences of negligent use of a USB flash drive will make her internalize this behavioral belief. This can be further underlined by the fact that the participants of the measure receive a small reward for the correct execution [32, p. 172]. This also creates a positive feeling in terms of behavior. The emergence of negative feelings through failure is prevented by providing support in practicing the behavior. The control beliefs can be changed or reformed in dialogue by defusing expected barriers or presenting supporting factors that the employee did not previously know. Encouraging and supporting the employee can influence efficacy beliefs. Through the mediation of the guidelines normative beliefs are formed.
Social marketing can be used to promote the modification of behaviors [33, p. 33]. The salience of the behavior is influenced by highlighting the behavior and making it permanently visible, e.g., by the arousal of attention or by visual accents [22, p. 776]. When a poster instructs employees to lock the screen when they enter the coffee kitchen, the behavior gets salient and employees might go back to their computer to lock its screen. As a side effect, social marketing also creates knowledge [34, p. 437], although not to the same extent as classes. By demonstrating the benefits and the actual effort required to change a behavior, beliefs about the consequences of the behavior can also be changed. Depending on the behavior that has to be influenced, the message of social marketing can be designed. If normative beliefs need to be influenced, an e-mail from the manager or a poster with an employee could be used to deliver the message.
In general, it is advisable for companies to integrate the sensitization of their employees into a holistic security concept, such as an ISMS. An ISMS includes organizational security measures that also impact employees' security awareness by eliminating constraints ("I'm not able to lock my screen") and barriers ("It is difficult to lock my screen") for security-relevant behaviors. If, for example, the IT department places the link to the password changing function more prominently on the intranet, it will be easier for users to actually change their password. An information security team is set up that serves as a point of contact for employees. The existence of such a team can positively influence the beliefs of the employees' effectiveness and thus their self-efficacy expectations [16, p. 8], because they can rely on their support for security-related matters. The creation of an information security policy is often recommended as the basis for increasing information security in companies [29, p. 12; 14, p. 11]. It also influences the employee's security awareness by increasing knowledge through its informative nature. As a set of rules, it also provides the basis for normative beliefs about the company's expectations. The consequences, i.e., rewards or sanctions for compliance / non-compliance with security guidelines should also be

specified in this document [29, p. 18]. Rewards for normative and punishments for non-normative behavior play an important role as extrinsic motivation in influencing the beliefs regarding the injunctive norms [35]. However, the sanctions also have an effect on the behavioral beliefs by showing employees what consequences their behavior may have.
Supportive factors, such as increasing the usability of application systems, can simplify the performance of each behavior and impact a person's control beliefs and efficacy beliefs. The impact increases with the probability that supporting factors occur and their strength [8, p. 80]. Existing supportive factors can be identified through the survey results and, if possible, strengthened. By a critical examination of the applications and information systems required for security-related behavior, additional supporting factors can be identified and implemented.

4 Conclusion and Outlook

This paper offers a new approach to information security awareness. It transfers research results from social psychologies into the information security context in a holistic way. The social psychological insights regarding the complex interrelations within human behavior are consistently applied to information security-compliant behavior of employees. The Integrated Behavioral Model offers a new explanation why, in addition to the knowledge of information security-compliant behavior, other factors such as habits, salience and intention must be taken into account when sensitizing employees for information security. Knowing the as-is state of influencing factors prevalent among employees enables companies to compile individual and tailor-made security awareness campaigns. The goal of those tailor-made campaigns is to not only train employees, but to persuade them to actually change their behavior in a favored way. The paper proposes techniques and measures for security awareness campaigns, highlighting what factors these techniques may affect. Understanding the complex interrelations described in the IBM further allows for selecting additional techniques for increasing security awareness.
In order to use the techniques successfully, the current manifestation of all factors must be carefully analyzed among the employees in the company. The as-is analysis with interviews and questionnaires is initially expensive. Due to the complexity of the factor intention, Montaño and Kasprzyk [8] studied people's beliefs and emotions intensively. In further research, their results have to be taken into account in the information security environment as well. It remains to be explored whether, after examining several companies, a general catalogue of beliefs and feelings can be compiled. That catalogue would lead to an equally valid questionnaire that would shorten the analysis phase to the quantitative study.
In order to start verifying the assumptions laid down in this work, we started a research project at our university. As part of an ISMS introduction project, we analyzed the information security awareness of university members (students, lecturers and employees). Other research aims at developing practical recommendations for planning security awareness campaigns for certain information security-compliant behaviors, such as choosing a secure password.
Jäger et al. [36] describe findings about the influence of normative factors, such as role models within the company. In addition, their paper underlines the importance of an information security team that addresses the issue of information security and specific security incidents, which creates a salience for employees. These findings will be considered in our future research, as they provide concrete insights for the normative beliefs and salience that are neglected especially by common training classes about information security.

Acknowledgment: This work was supervised by Prof. Dr. Kristin Weber from the University of Applied Sciences Würzburg-Schweinfurt, Faculty of Computer Science and Business Information Systems, Sanderheinrichsleitenweg 20, 97074 Würzburg, Germany.

References

[1] S. Schäfer, C. Pinnow: *Industrie 4.0 – Grundlagen und Anwendungen. Branchentreff der Berliner Wissenschaft und Industrie (1.Aufl.)*, DIN e.V. (2015), 129.
[2] Semba, B., Eymann, T.: *Developing a Model to Analyze the Influence of Personal Values on IT Security Behavior*. In: Nissen, V. et al. (Hrsg.): *Multikonferenz Wirtschaftsinformatik (MKWI) 2016*, S. 1083-1091. TU Ilmenau, Ilmenau (2016).
[3] L. Hirshfield, P. Bobko, A. J. Barelka, M. R. Costa, G. J. Funke, V. F. Mancuso, V. Finomore, and B. A. Knott: *The Role of Human Operators' Suspicion in the Detection of Cyber Attacks*. In: *International Journal of Cyber Warfare and Terrorism*, 5(3), S. 28-44. IGI Global, Hershey (2015)
[4] ISACA: *State of Cybersecurity 2017. Part 2: Current Trends in the Threat Land-scape*, ISACA (2016).
[5] G. Widdershoven: *Energy sector cyber security 2.0: Pro-active approach needed to tackle insider threats*. In: KPN (Hrsg.): *European Cyber Security Perspectives 2017*. KPN (2017).
[6] D. van Hemert, C. Broekman, H. van den Berg & T. van Vliet: *What you need to know about research in 2016 on human factor in cyber security*. In: KPN (Hrsg.): *European Cyber Security Perspectives 2017*. KPN (2017).
[7] Allianz für Cyber-Sicherheit: *Awareness-Umfrage 2015*. Bundesamt für Sicherheit in der Informationstechnik (2016).
[8] D. E. Montaño, D. Kasprzyk: *Theory of Reasoned Action, Theory of Planned Behavior, and the Integrated Behavioral Model*. In: K. Glanz, B. Rimer, K. & K. Viswanath, *Health Behavior and Health Education. Theory, Research, and Practice*, 4th Edition, John Wiley & Sons (2008), 67-96.
[9] M. Bada, A. M. Sasse, J. R. Nurse: *Cyber Security Awareness Campaigns: Why do they fail to change behaviour*? In: *Global Cyber Security Capacity Centre: Draft Working Paper*, Global Cyber Security Capacity Centre (2014), 118-131.
[10] T. W. Harich: *IT-Sicherheit im Unternehmen* (1. Auflage), mitp-Verlag (2015).
[11] M. Wolf: *Von Security Awareness zum Secure Behaviour*. In: *Hakin9 Extra – IT-Forensik*, 5, Software Wydawnictwo (2012), 18 – 19.
[12] Hänsch, N., Benenson, Z.: *Specifying IT security awareness*. In: Morvan, F., Wagner, R., Tjoa, A. (Hrsg.): *25th International Workshop on Database and Expert Systems Applications 2014*, S. 326-330. IEEE Computer Society, Los Alamitos, CA (2014).
[13] M. Helisch, D. Pokoyski: *Security Awareness. Neue Wege zur erfolgreichen Mitarbeiter-Sensibilisierung*. Vieweg+Teubner Verlag / GWV Fachverlage GmbH Wiesbaden (2009).
[14] M. E. Kabay: *Using Social Psychology to Implement Security Policies*. In: S. Bosworth & M. E. Kabay: *Computer Security Handbook*, Wiley (2002), 1-22.
[15] S. R. Boss: *Control, Perceived Risk and Information Security Precautions. External and Internal Motivations for Security Behavior*, University of Pittsburgh (2007).
[16] S. M. Galvez, I. R. Guzman: *Identifying Factors that Influence Corporate Information Security Behavior*, AMCIS (2009).
[17] F. Häußinger: *Studies on Employees' Information Security Awareness*, University Göttingen (2015).
[18] M. Fishbein, I. Ajzen: *Belief, attitude, intention, and behavior. An introduction to theory and research (Addison-Wesley series in social psychology)*, Addison-Wesley Pub. Co. (1975).
[19] I. Ajzen: *The theory of planned behavior*. In: *Organizational Behavior and Human Decision Processes* (50), Academic Press (1991), 179–211.
[20] B. Lebek, J. Uffen, M. Neumann, B. Hohler, M. Breitner: *Employees' Information Security Awareness and Behavior: A Literature Review*. In: *Hawaii International Conference on System Sciences, HICSS (2013)*, 2978-2987.
[21] W. Stroebe, M. Stroebe, S. Niedernhuber: *Lehrbuch der Gesundheitspsychologie. Ein sozialpsychologischer Ansatz (1. Aufl.)*, Klotz (1998).
[22] R. F. F. Baumeister, K. D. D. Vohs: *Encyclopedia of Social Psychology*, SAGE Publications (2007), 775.
[23] H. C. Triandis: *Interpersonal behavior*, Brooks/Cole (1977).
[24] R. Schwarzer: *Psychologie des Gesundheitsverhaltens. Einführung in die Gesundheitspsychologie* (3., überarb. Aufl.), Hogrefe (2004).

[25] D. Kasprzyk, D. E. Montaño: *Application of an integrated behavioral model to understand HIV prevention behavior of high-risk men in rural Zimbabwe*. In: I. Ajzen, D. Albarracin, *Prediction and Change of Health Behavior: Applying the Reasoned Action Approach*, Psychology Press (2007), 145–168.
[26] D. E. Montaño, D. Kasprzyk: *Theory of Reasoned Action, Theory of Planned Behavior, and the Integrated Behavioral Model*. In: K. Glanz, Rimer, Barbara, K. & K. Viswanath, Health Behavior. Theory, Research, and Practice, 5th Edition John Wiley & Sons (2008), 95–124.
[27] B. Verplanken, H. Aarts: *Habit, Attitude, and Planned Behaviour. Is Habit an Empty Construct or an Interesting Case of Goal-directed Automaticity?* In: *European Review of Social Psychology*, 10 (1), European Association of Social Psychology (1999), 101–134.
[28] Bundesamt für Sicherheit in der Informationstechnik: *IT-Grundschutz-Catalogues*. 13. Version, BSI (2013).
[29] Bayerisches IT-Sicherheitscluster: *Handbuch zur effizienten Gestaltung von Informationssicherheit im Mittelstand*, Version 1.7., Bayrischer IT-Sicherheitscluster e.V. (2017).
[30] A. Rieb, M. Hofmann, A. Laux, S. Rudel, U. Lechner: *Wie IT-Security Match-plays als Awarenessmaßnahme die IT-Sicherheit verbessern können*. In Leimeister, J.M.; Brenner, W. (Hrsg.): *Proceedings der 13. Internationalen Tagung Wirtschaftsinformatik (WI 2017)* (2017), 867-881.
[31] I. Ajzen: *Behavioral Interventions Based on the Theory of Planned Behavior. Brief Description of the Theory of Planned Behavior* (2006). Zugriff am 20.01.2017. Verfügbar unter https://people.umass.edu/aizen/pdf/tpb.intervention.pdf
[32] M. Thomson, R. von Solms (1998): *Information security awareness: educating your users effectively*. In: *Information Management & Computer Security*, 4 (6), Emerald (1998), 167–173.
[33] T. Baranowski, K. W. Cullen, T. Nicklas, D. Thompson, J. Baranowski: *Are Current Health Behavioral Change Models Helpful in Guiding Prevention of Weight Gain Efforts?* In: Obesity, 11 (10), The Obesity Society (2003), 23–43.
[34] J. D. Storey, G. B. Saffitz, J. G. Rimón (2008): *Social Marketing*. In: K. Glanz, Rimer, Barbara, K. & K. Viswanath (Hrsg.), *Health Behavior and Health Education. Theory, Research and Practice*. 4th Edition, John Wiley & Sons (2008), 435–461.
[35] E. Aronson, T. D. Wilson, R. M. Akert: *Sozialpsychologie*. In: Pearson Studium - Psychologie, 6., aktualisierte Aufl., [Nachdr.]. Pearson Studium (2010).
[36] L. Jaeger, C. Ament, A. Eckhardt: *The Closer You Get the More Aware You Become – A Case Study about Psychological Distance to Information Security Incidents*. ICIS 2017 Proceedings (2017). 11.

Andreas E. SCHÜTZ
University of Applied Sciences Würzburg-Schweinfurt
Faculty of Computer Science and Business Information Systems
Sanderheinrichsleitenweg 20, 97074 Würzburg
GERMANY
E-mail: andreas.schuetz@fhws.de

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

The Pictures of Chaos

Eduard-Traian Ştefănescu, Anastasia Tica, Pruteanu-Popescu Marek

Abstract

The goal of this project is to present an application, designed for practical illustration of the concept of fractals. Considering this application is being created as an educational one, this project aims to ease the process of introduction in fractals, as much for pupils, as even for children of preschool age. We choose to focus on the fractals, considering that it is a relatively new field in mathematics (born in the 1970's) so we could call it an up-to-date subject. Despite this, most sources offer explanations and applications only for mathematically-prepared students or scientists, neglecting the other groups of less experienced learners. The challenge of this application is to give users the opportunity to learn how fractals can be created in real time, by showing the progress of its construction. As an educational impact, we expect this application to encourage children and pupils to venture into the wonderful world of fractals without fear that the subject goes beyond their limits.

1 Introduction

From the mid-80s to the present day the concepts of fractal and fractal geometry has become an extremely inviting direction for researches among mathematicians and programmers.

The word fractal, formed from the latin *fractus*, meaning "consisting of fragments", was proposed as the name for irregular, yet self-similar structures by Benoit Mandelbrot in 1975 in his publications regarding the study of those structures.

The same Mandelbrot is considered the pioneer of fractal geometry, thanks to his work and publication, in 1977, of the book `The Fractal Geometry of Nature ', in which Mandelbrot would also use the scientific results of other scientists who worked earlier in the same field (Henri Poincare, Pierre Fatou, Gaston Julia, Georg Cantor, Felix Hausdorff).

The fractal geometry itself became popular as soon as scientists predicted the prospects of its development side by side with computer graphics.

From the point of view of computer graphics, fractal geometry is indispensable in the generation of an imitation of both the earthly and extraterrestrial shapes that occur in our Universe.

In fact, fractal geometry allows the reproduction of objects whose shapes do not fit within the limits of Euclid's geometry.

Given the relevance of this topic, our project is based on an attempt to open this mathematical field for the study of an inexperienced user, who may not be so mathematically-prepared, but is willing to understand the nature of fractals, what it is and how it is formed.

As we consider that visualization is the best and more effective way of getting to know the concept of a fractal, this is what our application will do: the user will face a kind of dialog box in which he can select the desired form of the fractal and observe how this form will be constructed.

While designing this application, we came to the conclusion that visualization will be the best assistant for the first steps of familiarizing with the fractals: an explicit picture would emphasize the essence much better than some dry number and equations.

We should notice that the developed application aims to introduce very young children, pupils in junior classes, who do not yet have a solid baggage of mathematical knowledge into the world of fractals.

That being said we decided to use the visualization of mathematical concepts as a tool that will significantly facilitate the acquaintance with them, whether it is a solved example or, as in our case, a direct observation of the construction of a geometric form [1].

1.1 What is a fractal?

Although there is no widely accepted formal definition for fractals, the definition given by Mandelbrot tells that "a fractal is a geometric shape that can be separated into parts, each of which is a reduced-scale version of the whole."

Fractals are said to possess infinite detail, and they may actually have a self-similar structure that occurs at different levels of magnification. The concept of self-similarity consists in the repetition of the image of the same structure during the continuous approaching of the original image.

Among all the characteristics of this type of pattern we should notice that fractals forms are form is extremely irregular or fragmented, and remains so, independent of the scale of examination, they exist in fractal dimension and the whole structure is formed by iterations.

Resuming that, we could consider the fractal being a door into a new mathematical field – Fractal Geometry.

1.2 Why studying fractals?

Years ago, when fractals were a kind of novelty in mathematics, often they were blessed with attention due their amazing shapes. However, there are many more reasons to study fractals.

According to Mandelbrot himself, “geometry is often described as "cold" and "dry". One reason lies in its inability to describe the shape of a cloud, a mountain, a coastline, or a tree. Clouds are not spheres, mountains are not cones, coastlines are not circles, and bark is not smooth, nor does lightning travel in a straight line”.

The primary reason to work on fractals and fractal geometry was the challenge of describing the nature in a mathematically-approached context, to define and “to study those forms that Euclid leaves aside as being "formless," to investigate the morphology of the "amorphous”””.

Thus, we can define fractals as a bridge between living worlds, natural, so irregular and fragmented forms and mathematics.

If “Nature exhibits not simply a higher degree but an altogether different level of complexity”, then fractals and fractal geometry are the ladder that leads to its analytical understanding.

Fractals describe many of the irregular and fragmented patterns around us: coastlines, snowflakes, sea shells, tree leaves, lightning and others.

Besides, fractals are always much closer to us than we think. They are in us. In our chromatin. Literally. MIT researches declare that the chromatin has the shape of a fractal, preventing DNA from getting tangled [2].

The bronchial tubes in the human lung also have one fractal dimension for the first seven generations of branching, and a different fractal dimension from there on in. The kidneys, the liver, the pancreas and even the human brain, are all organs constructed along self-similar fractal rules.

Of course, in addition to empirical interest, there is also a practical reason to consider fractals.

The fractal implication in biology was revealed by the digital complex representation of a DNA sequence and the analysis of existing correlations by wavelets. The symbolic DNA sequences were mapped into a nonlinear time series. By studying these time series, the existence of fractal shapes and symmetries was shown.

Probably one of the most relevant examples of fractal applications in engineering is the fractal-shaped antenna, used for the first time in 1990, that is responsive to a much wider range of frequencies of signals (as the fractal repeats itself more and more, the fractal antenna can pick more and more signals, using less space). Later, the experience of fractal-shaped antenna inspired its creator (Nathan Cohen) to design a new antenna, using a fractal called the Menger Sponge. This type of antenna is sometimes used in cell phone antennas.

Fractals help us study and understand important concepts from different fields. Economics is not an exception: Mandelbrot tried using fractal mathematics to describe the market - in terms of profits and losses traders made over time, and found it worked well [3].

In the end, fractals are used in creating some computer graphics, so widely exploited nowadays in cinema and gaming industries, aiming to create realistic landscapes for the best experiences.

Highlighting the main idea of this point, the conclusion is that fractals are worthy of universal attention because they are the embodiment of Today. They are among us and they help us understand and analyse important things and “reveal that some of the most austerely formal chapters of mathematics had a hidden face: a world of pure plastic beauty unsuspected till now” [4].

1.3 Why should pupils be encouraged to study fractals?

While planning the developed application, we have chosen the pupils of elementary grades as the target audience.

We find it important to encourage small classes of children to get acquainted with the fractal concept. And that is why:

First of all, the particularly attractive shape of a fractal will capture children's attention and interest in studying a yet unknown element. Using this application, users will be able to understand how to construct a fractal step by step, in an illustrated way.

The long-term effect of this idea is that this splitting in simple steps and modular understanding of such a complex mathematical concept could make children feel more confident about their own potential and interest in such *a fearsome* field as mathematics.

Success in understanding fractals could draw children into mathematics and fractal geometry in particular, giving them courage and enthusiasm for new successes.

2 How 3D printing could help?

3D printing is a new popular technology for prototyping and manufacturing. The point of 3D printing is the process of making a solid 3D object from a digital computer aided design file. The printer adds successive layers of material together until the final object has been created.

Since we expect this application to be an efficient teaching tool, we would also like to notice the role of 3D printing in the learning process.

A 3D printer is a visual thing. That is why it is very important to use its potential in studying such a visual concept as fractals are.

Seeing how a 3D printer creates, for example, a tangible Sierpinski triangle, students could realize the complexity of its construction and analyse the figure from different angles.

Learning about fractals using 3D printing would kill two birds with one stone: curiosity, caused by an incredibly beautiful form, created by modern technology, which can create objects of unique shapes that cannot be cut or formed by other machines, would be enough to stir up the interest for a new actual discipline about fractals.

When moving to fractals with a more complex form, students will not have problems with their understanding once they have the possibility to observe the gradual growth of the figure, visualizing each step, but also the 3D printer could create several parts of a fractal, which the students would have to assemble, reinforcing the knowledge gained about the new subject and feeling themselves as creators.

Fractals and 3D printing, used side by side, could generate the favourite discipline to many pupils.

Allowing students to use the latest technology when studying will make them understand that the future is here, and they are part of it.

2.1 Our printed fractals

Our team, with the help of Professor Ralf Fabian, has printed several fractals made of polylactic acid. These fractals are saved from the C# application and transformed into a Vector Graphics Scalable (SVG) with Inkskape. The transformation process from a SVG into a stereolithographic file (STL) was to add a height to the initial shape using the Blender program.

3 Fractals represented with L-Systems

3.1 What are L-Systems?

The Lindenmayer Systems were introduced by the biologist Aristid Lindenmayer in 1968 for simulating the development of multicellular organisms. The main purpose of L-Systems was to create different types of graphical representations for plants and fractals, in a manner that is closely related to abstract automata and formal languages. They became known and used after 1984, when A. R. Smith introduced state-of-the art computer graphics techniques that allow the visualization and processing of complex structures [5].

3.2 L-System format and characteristics

L-Systems are using the recursion that leads to self-similar shapes relating to fractals and some species of plants. They are also used in the generation of artificial life.

L-Systems use parametric systems defined as a tuple G=(V, ω, P), where V represents the alphabet, which is a set of symbols containing both elements that can be replaced called variables and those which cannot be replaced called constants. ω represents the axiom, which is a string of symbols from V describing the introductory state of the system and P represents a set of rules characterizing the way in which variables will be replaced with combinations of constants and other variables [6].

L-Systems operate with parametric words, they are strings of modules composed of letters and symbols associated with parameters. The parameters are composed by expressions or symbols using the arithmetic operators +, -, *, /, the exponentiation operator ^, the relational operators <, >, <=, >=, ==, the logical operators *!*, &&, || and the parentheses *()*.

The symbols : and → are used to separate the predecessor, the condition and the successor, these three are called the components of a production.

A production in L-Systems is valid only if it respects the following conditions:

1. the letter in the module and the letter in the production predecessor are the same
2. the number of actual parameters in the module and the number of formal parameters in the production predecessor are the same
3. the condition is true only if the actual parameter values can be replaced with the formal parameters [5]

3.3 D0L-Systems

The D0L-Systems are a simple subclass of the L-Systems, which are a deterministic and context-free (D0L). For this system consider strings built of two letters *a* and *b*, which are repeated many times.

The rule $a \rightarrow ab$ represents that the letter *a* will be replaced by the string *ab*, and the rule $b \rightarrow a$ represents that the letter *b* is to be replaced by *a*.

The rewriting process is called the axiom. For the first step the axiom *b* is replaced by *a* using the $b \rightarrow a$ rule and in the second step *a* is replaced by *ab* using the $a \rightarrow ab$ rule. Thus, using these rules and the axiom described before, from the first iteration will results the string *aba*, from the second iteration *abaab*, for the third iteration *abaababa* and so on [7].

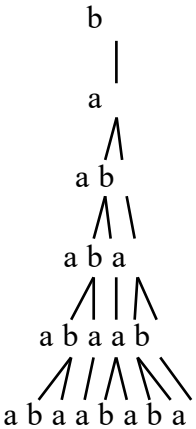


Fig. 1 Example of D0L-Systems

3.4 L-Systems rules

In programming languages fractals created with L-Systems are obtained by introducing character sets that describe the representation logic of a specific figure. Consider the following rule, $F \rightarrow F+F-F-FF+F+F-F$ and the axiom $F+F+F+F$ [8]. The F represents that the graphics need to draw a line forward and the + changes the angle from the initial angle to 90˚.

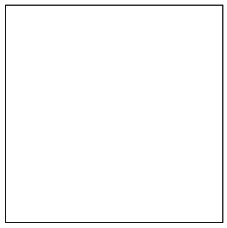


Fig. 2 The initial string F+F+F+F (∠90˚), generate a rectangle

Assuming the replacement rule $F \rightarrow F+F-F-FF+F+F-F$

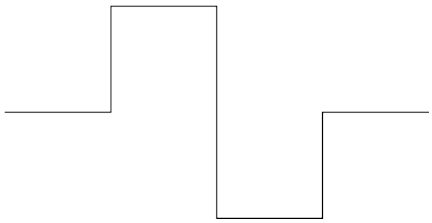


Fig. 3 The first iteration

In every iteration, the generated string produces the instructions for the next figure.

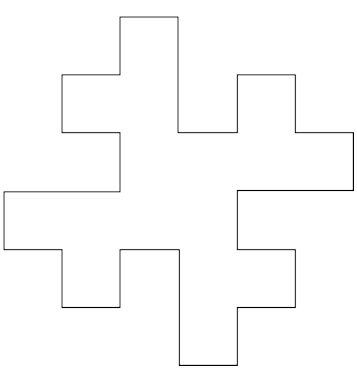


Fig. 4.1

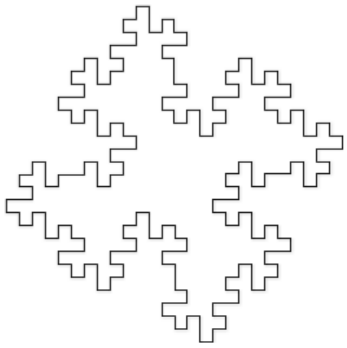


Fig. 4.2

Represents the second and the third iteration

Another example for a more complex fractal: [9]
Axiom: $F-F-F-F$ and angle: 90°
Rules:
$F \rightarrow F-b+FF-F-FF-Fb-FF+b-FF+F+FF+Fb+FFF$
$b \rightarrow bbbbbb$

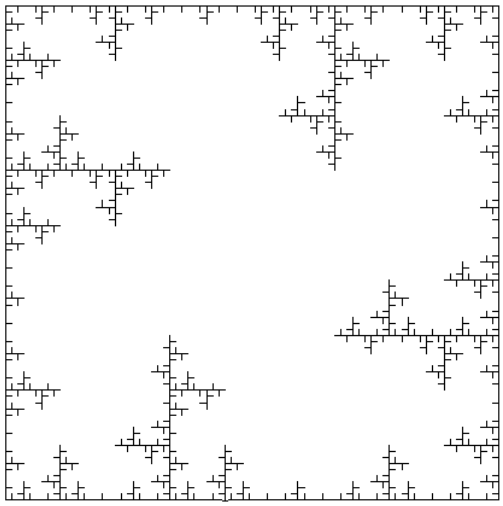


Fig. 5 Example of Ice Factal

3.5 Design models

3.5.1 Root growth

Lindenmayer Systems implement complex instructions for branching geometric forms, having a parameter t that represent a rule at a certain step. The delay rule $D^{\Delta t}$ is obtained for a character that is replaced after a predetermined time t_{final} by a successive string N_s.

$$D^{\Delta t}(t) \rightarrow \begin{cases} t + \Delta t \geq t_{end} : N_s^{t+\Delta t - t_{end}} \\ otherwise : D(t + \Delta t), \end{cases} \quad (1)$$

where the parameter t is the local age, t_{end} is the time at which the character is replaced and Δt is the time which passes during the application of the rule. The time t_{end} is replaced by its successor N_s only if the local time is longer than it. The time $t + \Delta t - t_{end}$ is applied to the product rule of N_s, that corresponding to the overlapping time, otherwise the character D have a new local time corresponding to $t + \Delta t$ [10].

3.5.2 Axial growth

The first subsection described a production rule of a single root without branching given to a continuous growth function $\lambda(t)$; this function is chosen for every root i. The root elongation is a negative exponential function [11].

$$\lambda_i(t) = k_i \left(1 - e^{-\frac{r_i}{k_i}t}\right), \quad (2)$$

where k_i is the maximal length of the root i, and the initial speed growth is represented by r_i. $G^{\Delta t}$ represent the production rule of a single root given by

$$G^{\Delta t}(t, l) \rightarrow \begin{cases} l + \Delta x < \lambda(t + \Delta t) : RF_{\Delta x}(t, l + \Delta x) \\ otherwise : G(t + \Delta t, l), \end{cases} \quad (3)$$

where t is the local age of the root tip, l is an approximation of the actual root length $\lambda(t)$, Δt and Δx are the time step and the spatial discretization. The character R indicates a rotation that describes root deflection and the character $F_{\Delta x}$ indicates a segment of the root with length Δx. In equation 3, the first expression represents the recursivity that produces segments $RF_{\Delta x}$ until the predetermined length $\lambda(t)$ is approximated, otherwise the local time is increased to $t + \Delta t$ and the root axis is described by segments with length Δx [12].

3.5.3 Lateral branching

In a system that is composed by roots, every root must produce lateral branches and a root is therefore divided into three sections: the first and the second section are composed by the basal and apical zones that are found near the base and the tip of the root and the third section is the branching zone where new roots of successive order are created.

The growth of basal and apical zones are as well as the growth between the branches, described by the section growth rule $S^{\Delta t}$, which is a generalisation of the axial growth rule give in equation 3. Thus, the equation 3 must be modified so it can produce segments of length Δx or less, obtained with the following system:

$$S^{\Delta t}(t,l) \rightarrow \begin{cases} A\ (\&)\ (l + \Delta x < l_s) : RF_{\Delta x} S^{\Delta t}(t, l + \Delta x) \\ A\ (\&)\ (l + \Delta x \geq l_s) : RF_{l_s - 1} N_s^{t + \Delta t - (t_{end} - t_0)} \\ otherwise : S(t + \Delta t, l), \end{cases} \tag{4}$$

where $A \coloneqq \lambda(t_0) + l + \Delta x \leq \lambda(t_0 + t + \Delta t)$, t_0 and t_{end} are the times at which the growth starts and ends, respectively the $\lambda(t_0)$ and $\lambda(t_{end})$ are the positions on the root axis (the first and the final), $l_s \coloneqq \lambda(t_{end}) - \lambda(t_0)$ is the length of the section and N_s represent the next section [12].

In the equation 3 the first expression coincides with the first expression in equation 4, with additional constraint, that growth does not exceed length l_s. The second expression represents the case if length exceeds l_s, and if that statement is true, a segment with the remaining length $F_{l_s - l}$ is produced and the overlap time is corrected by the successor $N_s = t + \Delta t - (t_{end} - t_0)$, otherwise the time is increased [12].

A fixed number of branches are created in the branching zone with a spacing between them that is determined by the section growth rule $S^{\Delta t}$, described in equation 4. The branching zone generate the L-Systems strings N_b, that stand for the branches of next topological order and are followed by a successive L-Systems string N_s, that stand for the second section of a branch represented by the apical zone. The next production rule for branching is $B^{\Delta t}$:

$$B^{\Delta t}(c) \rightarrow \begin{cases} c < n : D^{\Delta t}(0; d_c, N_b) \\ c < n : S^{\Delta t}(0, 0; t_{0,c}, t_{end,c}, \mathrm{B(c+1)}, \lambda, \Delta \mathrm{x}) \\ c = n : D^{\Delta t}(0; d_c, N_b) \\ c = n : S^{\Delta t}\big(0{,}0; t_{0,c}, t_{end,c}, N_8, \lambda, \Delta x\big), \end{cases} \tag{5}$$

where c represents the numbers of branches that have been already produced and n stand for the maximal number of branches. The first expression denotes that if the number of branches are less than the maximal number of branches, then a new lateral branch N_b is produced with a delay given by d_c and D is described in equation 1. The time delay is mandatory because the when a new branch is created it can only start growing when the apical zones has reached the required length and is significantly important that the length of the space between the branches to be created in times $t_{o,c}$ and $t_{end,c}$, that represent the initial and final time of a given number of branches. The S is explained in equation 4, but now is followed by the branching rule with increased counter c. And the second expression specify that after the last branch is created, the successive string N_s produces the final section, that represents the apical zone.

Therefore, the following structure

$$N_b = [R_\beta R_\alpha b] \tag{6}$$

describe the zone where the branch will be drawn, b describes a new branch, R_β and R_α produce the rotation at give angle β and α [12].

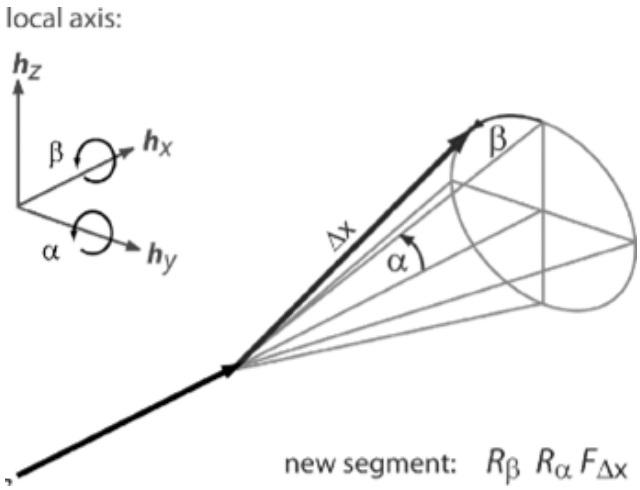


Fig. 6 The transformation of the root tip

4 The Pictures of Chaos application written in C#

4.1 A brief portrayal of the application

Considering the rules described by the Lindenmayer Systems, in this section will be presented the application created by our team written in C# programming language using Windows Forms. The design of the application keeps the modern standards, having a list of fractals from which the user can choose, he operates with different parameters, thus the drawn fractal will be an individual creation.

4.2 Program presentation

The business of the application is to generate a set of letters that will be the rules for the drawing framework. The application is structured in classes, one of these classes has the responsibility to generate a string of rules. The algorithm of that method that generate the rules respects the order described in D0L-Systems [13].

In order to draw a specific fractal, another the class is used to create a fractal with given information, related to the coordinates, line length, colour, angle and number of iterations, all of this informations are passed in a constructor.

The algorithm that follow the L-Systems rules is described in a method called DrawFractal, for each character that is in the axiom generated by the Generate method, the program produce a new step of drawing a line if the character is F, change to rotation sign if the characters are + or -, save the current state if the character is [and put it back if the character is].

In the principal window when the program is started, a welcome message is displayed. For saving the fractal, a pop-up window will appear and let the user to save the creation locally.

The last button represents all the fractals that were implemented in the application. In future implementations our team wants to add more fractals, and a more complex algorithm, that permits the user to enter his own sets of rules and axioms [14].

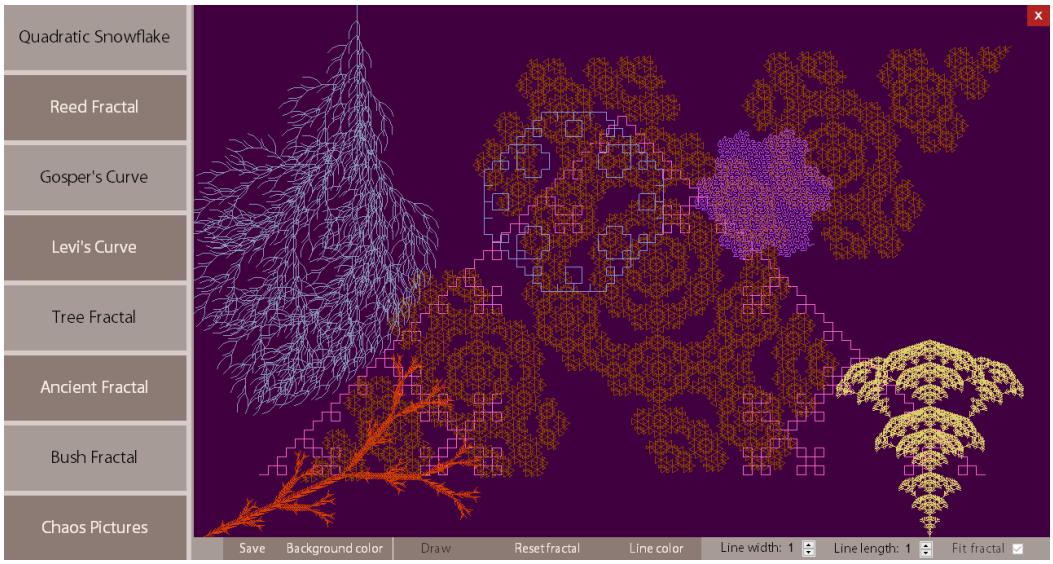


Fig. 7 Chaos Pictures

4.3 Alternative implementation

The same result can be achieved by using the Java programming language. In lieu of polluting the code with details for each and every L-System, one can use a JSON file which provides all of the needed details for generating one (axiom, angle, rules and much more). The core of the application is the same. It uses a turtle graphics-like engine to draw the shapes as the current sentence dictates. That same sentence is generated through a method that abides by the rules of L-Systems.

In order to produce a certain shape, the user needs to change a single line in the code, which is the name of the fractal that we want generated. That, however, can be difficult for the user who is not acquainted to programming, so with a few minor adjustments, the whole project can be

transformed into a jar file, which in turn can be run from the command line with a single argument between a set of quotes (e.g. java –jar lsystems.jar “Koch Snowflake”).

In the end, this can be taken a step further. Should we want more designs, appending to the JSON file is the desired solution. Another name, another axiom, another set of rules, and whatever more the user wishes to take advantage of. Surely, the code would need to be tinkered with, and that would fall under the expertise of a programmer.

The window itself is quite simple, presenting nothing more than a drawing space and a couple of buttons, one that computes the next generation and draws the lines, and the other, that clears the drawing space. Minimalism at its finest [15].

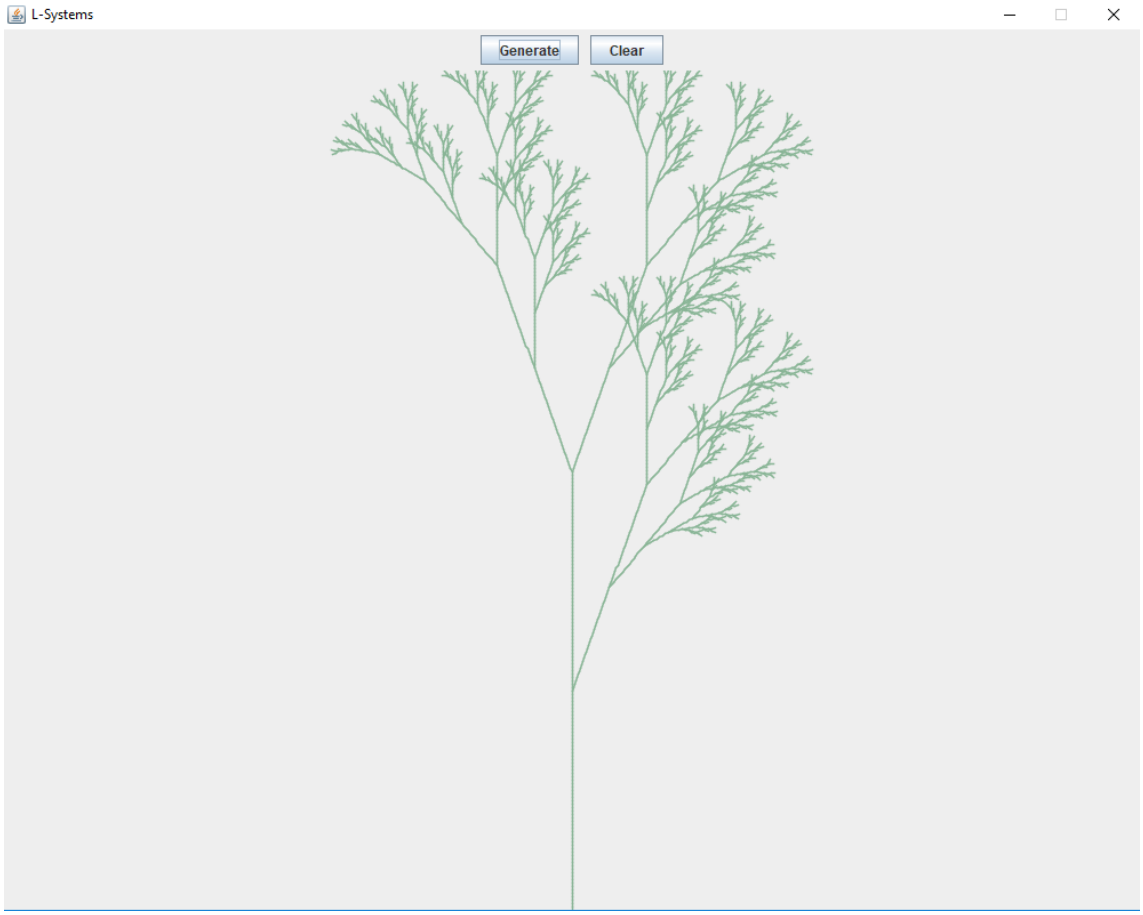


Fig. 8 Tree-like structure

5 Conclusion

In conclusion fractals represents an interesting field to explore, applicable in many domains. The forms obtained by many iterations are wonderful, being able to describes from the anatomic structures such as the lungs of our species to the form of a simple shell or can express an measurement unity for uneven forms such as the dimension of a mountain or the coastline of a country.

We consider that fractals can be used in education to improve the understanding of irregular shapes, such as an adult know the properties of a triangle, in the same way a student from primary classes can learn the attributes of fractals.

Our work in comparison with the other works, aims to produce a major interest in the study of fractals by people of all ages and wants to present them in an easy and formal way, close to the reality that surrounding us. But besides all of that, we wanted to have a palpable object, with that we can demonstrate the presented theories, that object is a 3D printed fractal.

We also have other ideas for developing the application, such as creating a puzzle game for phone in a fun and educational way, where the user need to put enough mental strength to solve a certain level. Another future implementation would be, that the desktop application users can save the created fractal in STL format for 3D printing.

Thus, with the above in mind, fractals represent an important domain for the future of our planet and universe.

Acknowledgement: This work was supervised by Professor Dana Simian from the Department of Mathematics and Informatics, Faculty of Sciences and by Professor Ralf Fabian from the Department of Mathematics and Informatics.

References

[1] Ruis Jules, "*Fractal*" 25 04 2018. [Online]. Available: http://www.fractal.org. [Accessed 05 01 2018].

[2] Carlo Cattani, *Fractals and Hidden Symmetries in DNA*, 2010.

[3] University Montana State, "*Phys*" 13 10 2013. [Online]. Available: https://phys.org/news/2011-10-beautiful-math-fractals.html#jCp. [Accessed 02 05 2018].

[4] Benoit B. Mandelbrot, *Fractal geometry of nature*, United States of America: W.H. Freeman and Company New York, 1997.

[5] J. H. M. H. R. M. Przemyslaw Prusinkiewicz, "*L-systems: from the Theory to Visual Models of Plants*" [Online]. Available: https://pdfs.semanticscholar.org/1fea/f3f7507805a6d8311fa7061a786270867110.pdf. [Accessed 04 05 2018].

[6] "*L-Systems structure*" Wikiepdia, [Online]. Available: https://en.wikipedia.org/wiki/L-system#Examples_of_L-systems. [Accessed 06 05 2018].

[7] Springer-Verlag, "*Graphical modeling using*" 1990. [Online]. Available: http://algorithmicbotany.org/papers/abop/abop-ch1.pdf. [Accessed 04 05 2018].

[8] Paul Bourke, "*L-System User Notes*" 07 1991. [Online]. Available: http://paulbourke.net/fractals/lsys/. [Accessed 04 05 2018].

[9] Dmytro Fedoriaka, "*L-systems list*" [Online]. Available: https://fedimser.github.io/l-systems.html. [Accessed 04 05 2018].

[10] A. F. P. D. T. Roose, "*A mathematical model of plant nutrient uptak*," SpringerLink, 04 2001. [Online]. Available: https://link.springer.com/article/10.1007%2Fs002850000075. [Accessed 05 05 2018].

[11] L. P. V.-L. D. L. C. L. Bot, "*Root Typ: a generic model to depict and analyse the root system architecture*" SprinkerLink, 01 2004. [Online]. Available: https://link.springer.com/article/10.1023%2FB%3APLSO.0000016540.47134.03. [Accessed 05 05 2018].

[12] S. K. G. B. A. S. Daniel Leitner, "*A dynamic root system growth model based on L-Systems*" SpringerLink, 07 2010. [Online]. Available: https://link.springer.com/article/10.1007/s11104-010-0284-7#Equ4. [Accessed 05 05 2018].

[13] D. Shiffman, "*Coding Challenge #16: Fractal Trees - L-System*" 31 05 2016. [Online]. Available: https://www.youtube.com/watch?v=E1B4UoSQMFw. [Accessed 05 05 2018].

[14] Eduard-Traian Stefanescu, GitHub, 11 04 2018. [Online]. Available: https://github.com/StefanescuEduard/ThePicturesOfChaos. [Accessed 10 05 2018].

[15] Pruteanu-Popescu Marek, GitHub, 29 04 2018. [Online]. Available: https://github.com/marek231/lsystems. [Accessed 11 05 2018].

Eduard-Traian STEFANESCU
University Lucian Blaga of Sibiu
Faculty of Sciences
Ion-Ratiu 5-7, Sibiu 550012
ROMANIA
E-mail: eduardstefanescu22@gmail.com

Anastasia TICA
University Lucian Blaga of Sibiu
Faculty of Sciences
Ion-Ratiu 5-7, Sibiu 550012
ROMANIA
E-mail: tica.anastasia@gmail.com

Marek PRUTEANU-POPESCU
University Lucian Blaga of Sibiu
Faculty of Sciences
Ion-Ratiu 5-7, Sibiu 550012
ROMANIA
E-mail: pruteanupopescumarek@gmail.com

International Conference on Applied Informatics
Imagination, Creativity, Design, Development
ICDD 2018, May 17-19
Sibiu, Romania

AI application using Mindstorms robot

Andrei Vass, Stefan Bereghici

Abstract

Using a Mindstorms robot and AI libraries, we teach the robot to answer question, and respond properly, also following the beacon he learns not to trespass the lines draw by us. We approach robotics, to become familiar with the nature of robotics AI using the Lego Mindstorms robot. We integrated in our project several AI libraries and also train the robot to not pass the lines we draw.

1 Voice and text-based conversational in the Mindstorms robot

Our aim was to create an application that plays with different AI APIs. For this purposes we use a robot – Lego Mindstorms EV3 that has a color sensor, touch sensor, control center and power station, remote infrared beacon, IR sensor, large motor and medium motor.

The language we choose to build our application was Python because of the lot of libraries that exists for AI, because is an open source language with a great community, also it provides ease of testing, and of course because of the flexibility which is the core advantages of Python. With the option to choose between OOPs approach and scripting Python is suitable for every purpose.

To interact with the robot we search through several libraries developed for this purposes. We found several APIs that can be used to interact with the robot by voice and text-based conversational interfaces powered by AI. We tested which one is better and gives better results so that we integrate in our application the most quality one. We tested the dialogflow, CMU Sphinx, Google Cloud Speech API, Wit.ai, Houndify API.

The test we run was very simple: we integrated the above APIs and run the same test on all of them. First we start by saying to the robot 20 simple words, then 20 simple sentences, then 10 phrases. To have a better accuracy we recorded this question and play them while testing the voice APIs.

We took the json responses, and comparing the results we noticed that the dialogflow was providing the most accurate results. We were surprised by the accuracy in which the robot responded to our questions, as well as how good it understood the text we were saying. Also the performance of the APIs tested was similar through all of them. Based on our test results, we choosed the DialoFlow to be the best voice and text-conversational API for our application.

Looking at all AI platforms, you will see that they are very similar. You have a user who speaks commands/questions to a device. This device will record the audio and stream it to an intermediate service.

The intermediary will recognize this as an initial request and send the audio to the speech-to-text service. The speech-to-text service converts the audio to text and returns text to the intermediary service. The intermediary then sends the text to the text-to-Intent/Action component.

This component is responsible for figuring out what the user wants to do. Usually AI Assistants will have phrases which trigger named intents.

For example, an application can look for the phrase "What's the weather in {Boston} {Massachusetts}" to trigger a get_weather intent.

Think of them as variables for voice commands. Once the intent/action name is figured out then depending on the platform, more interactions can take place gathering needed information by the platform or a webhook is invoked. Some platforms require you to make a web call for every intent.

Others, allow you to gather all of the data in their platform before invoking your webhook (business logic).

When your webhook is invoked the intent name, slot names, and slot values are passed to your business logic. This business logic could be housed in an AWS Lambda function or Heroku server. The business logic identifies which internal function needs to be called based on the intent name and then reads the required values from slot values using the slot names. The business logic can then invoke REST APIs on the internet to gather information which will be returned to the device and spoken to the end user.

Although, the AI assistant platforms are architecturally the same there are important differences you need to be aware of if you are developing for each platform. Each platform is special and provides unique advantages and disadvantages.[5]

Dialogflow incorporates Google's machine learning expertise and products such as Google Cloud Speech-to-Text.Dialogflow is backed by Google and runs on Google Cloud Platform, letting you scale to hundreds of millions of users.

The Dialogflow API is also suitable for machine learning. Machine Learning is the tool that allows understanding a user's interactions as natural language and convert them into structured data.

In Dialogflow terminology, we use machine learning algorithms to match user requests to specific intents and uses entities to extract relevant data from them. An agent "learns" both from the examples you provide in the Training Phrases section and the language models developed by Dialogflow.

Based on this data, it builds a model (algorithm) for making decisions on which intent should be triggered by a user input and what data needs to be extracted. This algorithm is unique.

The algorithm adjusts dynamically according to the changes made in your agent and in the Dialogflow platform. To make sure that the algorithm is improving, we need to constantly be trained using real conversation logs. The machine learning algorithm is updated every time we save changes in intents and entities, approve changes in Training.

To achieve good classification accuracy, it's important to provide the robot with enough data. The greater the number of natural language examples in the Training Phrases section of Intents, the better the classification accuracy. [1]

Figure 1: The Mindstorms robot

2 Reinforcement learning using Lego Mindstorms

We train the robot to follow a line and not trespass it using the reinforcement learning. To interact with the robot we search through several libraries developed for this purposes Reinforcement learning is one of the most active research areas in Artificial Intelligence. Reinforcement learning is training by rewards and punishments.

We do the training similar as if we train a dog. If the dog obeys and acts according to our instructions we encourage it by giving biscuits or we punish it by beating or by scolding. Similarly, if the system works well then the teacher gives positive value (i.e. reward) or the teacher gives negative value (i.e. punishment). The learning system which gets the punishment has to improve itself. Thus it is a trial and error process.

The reinforcement learning algorithms selectively retain the outputs that maximize the received reward over time. To accumulate a lot of rewards, the learning system must prefer the best experienced actions; however, it has to try new actions in order to discover better action selections for the future. [2]

In our application we use reinforcement learning by punishing the robot if he trespasses the line, and rewarding him in case he stays on the line.

Typically, a RL setup is composed of two components, an agent and an environment.

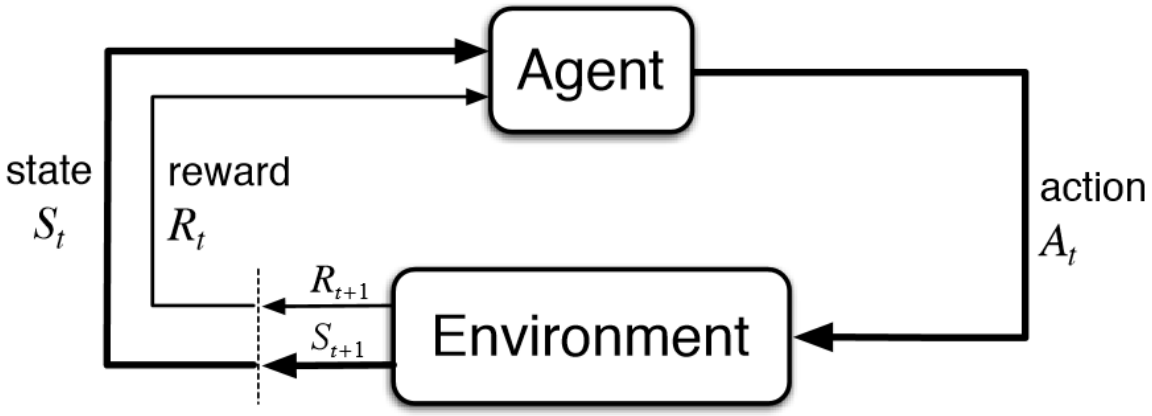


Figure 2: Reinforcement Learning Illustration

Then environment refers to the object that the agent is acting on while the agent represents the RL algorithm. The environment starts by sending a state to the agent, which then based on its knowledge to take an action in response to that state.

After that, the environment send a pair of next state and reward back to the agent. The agent will update its knowledge with the reward returned by the environment to evaluate its last action. The loop keeps going on until the environment sends a terminal state, which ends to episode.

Most of the RL algorithms follow this pattern:

1. Action (A): All the possible moves that the agent can take
2. State (S): Current situation returned by the environment.
3. Reward (R): An immediate return send back from the environment to evaluate the last action.
4. Policy (π): The strategy that the agent employs to determine next action based on the current state.
5. Value (V): The expected long-term return with discount, as opposed to the short-term reward R. $V\pi(s)$ is defined as the expected long-term return of the current state sunder policy π.
6. Q-value or action-value (Q): Q-value is similar to Value, except that it takes an extra parameter, the current action *a*. $Q\pi(s, a)$ refers to the long-term return of the current state *s*, taking action *a* under policy π. [4]

We also use in our application a scientific computing framework named Torch, that has a wide support for machine learning algorithms that is very easy to use and efficient.

Some core features of the framework we used are:

- a powerful N-dimensional array
- lots of routines for indexing, slicing, transposing, …
- amazing interface to C, via LuaJIT
- linear algebra routines
- neural network, and energy-based models
- numeric optimization routines
- Fast and efficient GPU support
- Embeddable, with ports to iOS and Android backend

The goal of Torch is to have maximum flexibility and speed in building scientific algorithms while making the process extremely simple. Torch comes with a large ecosystem of community-driven packages in machine learning, computer vision, signal processing, parallel processing, image, video, audio and networking among others. [3]

3 Program Code (Python language)

The main function that starts the robot:

```
def update(self, reward, new_signal):
 new_state = torch.Tensor(new_signal).float().unsqueeze(0)
 self.memory.push(
    (self.last_state, new_state, torch.LongTensor([int(self.last_action)]), torch.Tensor([self.last_reward])))
 action = self.select_action(new_state)
 if len(self.memory.memory) > 100:
    batch_state, batch_next_state, batch_action, batch_reward = self.memory.sample(100)
    self.learn(batch_state, batch_next_state, batch_reward, batch_action)
 self.last_action = action
 self.last_state = new_state
 self.last_reward = reward
 self.reward_window.append(reward)
 if len(self.reward_window) > 1000:
    del self.reward_window[0]
 return action
```

Audio and voice code sequence:

```
def listen(project_id, session_id, audio_file_path,
  language_code, noiseControl=0):
session_client = dialogflow.SessionsClient()

# Note: hard coding audio_encoding and sample_rate_hertz for simplicity.
audio_encoding = dialogflow.enums.AudioEncoding.AUDIO_ENCODING_LINEAR_16
sample_rate_hertz = 16000
 session = session_client.session_path(project_id, session_id)
print('Session path: {}\n'.format(session))
 r = sr.Recognizer()
with sr.Microphone() as source:
   if noiseControl != 0:
      print("Please wait. Calibrating microphone...")
      # listen for 3 seconds and create the ambient noise energy level
      r.adjust_for_ambient_noise(source, duration=noiseControl)
   print("You can now talk..")
   audio = r.listen(source)
   input_audio = audio.get_wav_data()
 audio_config = dialogflow.types.InputAudioConfig(
   audio_encoding=audio_encoding, language_code=language_code)
query_input = dialogflow.types.QueryInput(audio_config=audio_config)
 response = session_client.detect_intent(
   session=session, query_input=query_input,
   input_audio=input_audio)
```

Reinforcement code sequence:

```
class eyes():
   def __init__(self):
      self.cl = ev3.ColorSensor()
      self.ir = ev3.InfraredSensor()
      self.ir.mode = 'IR-SEEK'
      self.cl.mode = 'COL-REFLECT'
      self.isLeft = True
      self.isRight = False
     def scanSteps(self):
      result = []
      signal = []
```

```
        if self.isLeft:
            for x in range(0, 15):
                signal.append(self.scannerLeft())
            self.isRight = True
            self.isLeft = False
            result = np.array_split(signal, 3)
            return result
        if self.isRight:
            for x in range(0, 15):
                signal.append(self.scannerRight())
            self.isRight = False
            self.isLeft = True
            result = np.array_split(signal, 3)
            return result

    def getDistance(self):
        distance = self.ir.value(5)
        return distance

    def getHeading(self):
        heading = self.ir.value(4)
        return heading
```

4 Conclusion

We created a python application for the Lego Mindstorms EV3 robot, in which we integrated and play with main important AI APIs for voice and text based conversational based on AI, also implement the movement of the robot, on a line that we draw, and trained the robot using reinforcement learning.

References

[1] https://dialogflow.com/
[2] http://intelligence.worldofcomputing.net/machine-learning/reinforcement-learning.html#.Wuopg4huaUl
[3] http://torch.ch/
[4] https://towardsdatascience.com/introduction-to-various-reinforcement-learning-algorithms-i-q-learning-sarsa-dqn-ddpg-72a5e0cb6287
[5] https://medium.com/@abraham.kang/understanding-the-differences-between-alexa-api-ai-wit-ai-and-luis-cortana-2404ece0977c
[6] https://www.newgenapps.com/blog/python-for-ai-artificial-intelligence-ml

Andrei VASS
UTCLUJ- CUNBM
Computer Science
Victoriei 76, Baia Mare
ROMANIA
E-mail: vass.andrei.bm@gmail.com

Ştefan BEREGHICI
UTCLUJ-CUNBM
Computer Science
Victoriei 76, Baia Mare
ROMANIA
E-mail: sbereghici@gmail.com

LIST OF AUTHORS

Andrei-Timotei ARDELEAN
Transilvania University of Brasov
Faculty of Mathematics and Informatics
Iuliu Maniu, no 50
ROMANIA
E-mail: timitoc@yahoo.com

Alexandra Maria BĂILA
"Lucian Blaga" University of Sibiu
Faculty of Science
Department of Mathematics and Informatics
Dr. I. Ratiu St., No. 5-7, Sibiu, 550012
ROMANIA
Email: baila_alexandra29@yahoo.com

Ștefan BEREGHICI
Technical University of Cluj-Napoca
North University Center of Baia Mare
Computer Science
Victoriei 76, Baia Mare
ROMANIA
E-mail: sbereghici@gmail.com

Nataliya BOBROVSKAYA
Moscow Institute of Physics and Technology (MIPT)
Department of Innovation and High Technology
9 Institutskiy per., Dolgoprudny, Moscow Region, 141701
Russian Federation
E-mail: nataliya.bobrovskaya@phystech.edu

Dumitru Bogdan Mateis BORUZ
University of Craiova
Faculty of Sciences
Department of Computer Science
13 A.I. Cuza Street, Craiova,200585
ROMANIA
E-mail: bogdan.mateis@yahoo.com

Petrică BOTA
"Lucian Blaga" University of Sibiu
Faculty of Science
Department of Mathematics and Informatics
Dr. I. Ratiu St., No. 5-7, Sibiu, 550012
ROMANIA
E-mail: botapetre1@gmail.com

Carlos CANO-DOMINGO
University of Málaga
ETSI Telecomunicación
SPAIN
E-mail: carloscanodomingo@gmail.com

Marius CIOCAN
Transilvania University of Brasov
Faculty of Mathematics and Informatics
Iuliu Maniu, no 50
ROMANIA
E-mail: marius.p.ciocan@gmail.com

Alexey Alexandrovich CHURKIN
Volga State University of Service,
Department of Applied Informatics in Economics
4 Gagarin st., Togliatti, Samara region, 445677
RUSSIA
E-mail: Kaf_pive@tolgas.ru

Constantin Robert CIOLOMPU
"Lucian Blaga" University of Sibiu
Faculty of Science
Department of Mathematics and Informatics
Dr. I. Ratiu St., No. 5-7, Sibiu, 550012
ROMANIA
E-mail: robert.ciolompu@yahoo.com

Alexandru CRĂCIUN
Iuliu Haţeganu University of Medicine and Pharmacy of Cluj-Napoca
400349 Cluj-Napoca, Louis Pasteur no. 4
ROMANIA
E-mail: craciunalex98@yahoo.com

Gheorghe-Cătălin CRIȘAN
"Lucian Blaga" University of Sibiu
Faculty of Science
Department of Mathematics and Informatics
Dr. I. Ratiu St., No. 5-7, Sibiu, 550012
ROMANIA
E-mail: crisan.gheorghecatalin@gmail.com

Alexander M. FRÜHWALD
University of Applied Sciences Wuerzburg-Schweinfurt
Faculty of Computer Science
Sanderheinrichsleitenweg 20, 97074 Würzburg
GERMANY
E-mail: alexander.fruehwald@student.fhws.de

Svetoslav HADZIIVANOV
University of Ruse
Department of Informatics and Information Technologies
8, Studentska str., Ruse 7017
BULGARIA
E-mail: warrolen@gmail.com

Darius HAȚEGAN
"Lucian Blaga" University of Sibiu
Faculty of Science
Department of Mathematics and Informatics
Dr. I. Ratiu St., No. 5-7, Sibiu, 550012
ROMANIA
E-mail: hategandarius71@gmail.com

Wolfgang HEINZ
University of Applied Sciences Wuerzburg-Schweinfurt
Faculty of Computer Science
Sanderheinrichsleitenweg 20, 97074 Würzburg
GERMANY
E-mail: wolfgang.heinz@student.fhws.de

Moritz HEUSINGER
University of Applied Sciences Wuerzburg-Schweinfurt
Faculty of Computer Science and Business Information Systems
Sanderheinrichsleitenweg 20, 97074 Würzburg
GERMANY
E-mail: moritz.heusinger@gmail.com

Elena HINOVEANU
University of Craiova
Faculty of Automation, Computer Science and Electronics
Decebal Bd. no. 107, Craiova
ROMANIA
E-mail: elenamhinoveanu@gmail.com

Iskren IVANOV
University of Ruse
Department of Informatics and Information Technologies
8, Studentska str., Ruse 7017
BULGARIA
E-mail: iskren_damqnov@abv.bg

Mihail JURAVLEA
"Lucian Blaga" University of Sibiu
Faculty of Science
Department of Mathematics and Informatics
Dr. I. Ratiu St., No. 5-7, Sibiu, 550012
ROMANIA
E-mail: the_mihail@yahoo.com

Andrey Alexandrovich KABANOV
Volga State University of Service,
Department of Applied Informatics in Economics
4 Gagarin st., Togliatti, Samara region, 445677
RUSSIA
E-mail: kabanovst@yandex.ru

Asaf Ali KHAN
Siena State University
Faculty of Economics, Law and Political Sciences
Department of Business and Law
Piazza San Francesco, 8, Siena
ITALY
E-mail: asafalikhan8@gmail.com

Borislav KOSHAROV
University of Ruse
Department of Informatics and Information Technologies
8, Studentska str., Ruse 7017
BULGARIA
E-mail: bosakmaw@gmail.com

Nicolae LUNGU
"Lucian Blaga" University of Sibiu
Faculty of Science
Department of Mathematics and Informatics
Dr. I. Ratiu St., No. 5-7, Sibiu, 550012
ROMANIA
E-mail: nicolaelungu8@gmail.com

Alexandru PINTEA
National College E.Racovita
Mathematics & informatics
9-11 Kogalniceanu, Cluj-Napoca
ROMANIA
E-mail: al_pintea@yahoo.com

Marek PRUTEANU-POPESCU
"Lucian Blaga" University of Sibiu
Faculty of Science
Department of Mathematics and Informatics
Dr. I. Ratiu St., No. 5-7, Sibiu, 550012
ROMANIA

E-mail: pruteanupopescumarek@gmail.com

Mihai SANDU
University of Craiova
Department of Computer Science
Craiova, A.I.Cuza Street, no. 13
ROMANIA
E-mail: mihai1996sandu@gmail.com

Robert SĂNDICA
"Lucian Blaga" University of Sibiu
Faculty of Science
Department of Mathematics and Informatics
Dr. I. Ratiu St., No. 5-7, Sibiu, 550012
ROMANIA
E-mail: sandica.robert@gmail.com

Andreas SCHÜTZ
University of Applied Sciences Würzburg-Schweinfurt
Faculty of Computer Science and Business Information Systems
Sanderheinrichsleitenweg 20, 97074 Würzburg
GERMANY
E-mail: andreas.schuetz@fhws.de

Eduard Traian ȘTEFĂNESCU
"Lucian Blaga" University of Sibiu
Faculty of Science
Department of Mathematics and Informatics
Dr. I. Ratiu St., No. 5-7, Sibiu, 550012
ROMANIA
E-mail: eduardstefanescu22@gmail.com

Anastasia TICA
"Lucian Blaga" University of Sibiu
Faculty of Science
Department of Mathematics and Informatics
Dr. I. Ratiu St., No. 5-7, Sibiu, 550012
ROMANIA
E-mail: tica.anastasia@gmail.com

Andrei VASS
Technical University of Cluj-Napoca
North University Center of Baia Mare
Computer Science
Victoriei 76, Baia Mare
ROMANIA
E-mail: vass.andrei.bm@gmail.com

LIST OF PAPERS' SUPERVISING PROFESSORS

Name, Affiliation and List of papers

Pavel Akhmetchanov, Chief Programmer of the Web Interface Department, Tinkoff Bank, Russian Federation Paper: *Prediction of project task execution time by clustering and classification methods*
Arndt Balzer, University of Applied Sciences Würzburg-Schweinfurt, Germany Paper: *A generic Scratch interpreter for embedded systems*
Alexandra Băicoianu, Transilvania University of Brasov, Romania Papers: *Automated Car Parking System* *Smart Parking Solution*
Mihaela Colhon, University of Craiova, Romania Paper: *FRELaw Dictionary – a Trilingual Glossary of Law Terms*
Frank Ebner, University of Applied Sciences Würzburg-Schweinfurt, Germany Paper: *A generic Scratch interpreter for embedded systems*
Ralf Fabian, "Lucian Blaga" University of Sibiu, Romania Papers: *Smart app for your smart home* *2D Videogame, turn based strategy in Unity* *The Pictures of Chaos* *Emergency Situations* *Autonomous navigation system based on data acquisition*
Katalina Grigorova, University of Ruse, Bulgaria Papers: *Bypassing Antiviruses in 2018* *Termohub – IoT web service* *Treasureland – 2D platform game*
Mara Hajdu-Măcelaru, Technical University of Cluj-Napoca, North University Center of Baia Mare, Romania Paper: *AI application using Mindstorm robot*
Sorin Ilie, University of Craiova, Romania Paper: *Implementing a Local Grid-based System applied on E-Commerce web crawling*
Gonzalo Joya, University of Malaga Paper: *An ecient FPGA implementation of a deep learning approach for the classication of saccadic movements in clinical electro-oculography*
Isabel John, University of Applied Sciences Würzburg-Schweinfurt, Germany Paper: *Secret synergy effects from physics and predictive analytics - The future of mobility?*
Elena Malysheva, Volga State University of Service, Russia Paper: *Application of component-oriented programming and MVC pattern in development of orthopedic salon information system*
Frank-Michael Schleif, University of Applied Sciences Würzburg-Schweinfurt, Germany Paper: *Affinity Propagation as an algorithm for segmentation problems in business intelligence*

Dana Simian, "Lucian Blaga" University of Sibiu, Romania Papers: *Digital interactions based on brainwaves signals* *Food Ordering System* *The Pictures of Chaos* *Services marketplace web app*
Cătălin Stoean, from Department of Computer Science, University of Craiova, Romania Papers: *Medical Image Classication Based on Pixel Intensity Features* *An ecient FPGA implementation of a deep learning approach for the classication of saccadic movements in clinical electro-oculography*
Ruxandra Stoean, from Department of Computer Science, University of Craiova, Romania Paper: *An ecient FPGA implementation of a deep learning approach for the classication of saccadic movements in clinical electro-oculography*
Cristiana Nicola Teodorescu, University of Craiova, Romania Paper: *FRELaw Dictionary – a Trilingual Glossary of Law Terms*
Kristin Weber, University of Applied Sciences Würzburg-Schweinfurt, Germany Paper: *Information Security Awareness: It's Time to Change Minds!*

SPONSORS (in alphabetical order)

AUSY
Technologies Romania

ROPARDO

CodexWorks
technologies

TOPTECH
INTRODUCING TOP TECHNOLOGY

Academia Ardeleana Foundation

VISMA

GSD

IQUEST

KeepCalling

NTT DATA
Global IT Innovator

PAN FOOD

OMERON
TECHNOLOGIES

PROIT
AUDI STRATEGIC PARTNER