EUROPEAN PROJECT SEMESTER

FINAL REPORT

Smart Healthcare for Smart Cities

Aneta Andrzejewska

Cristian Pérez Llasera

Jonatan González Jiménez

Alexander Likhterman

Mentor: Przemysław Ignaciuk, Professor

SALU+EM

19/06/2018

SALU+EM

Table of Contents

Abstract

The growing population around the world and increase of seniors, as well as a demand for high quality services create new problems and challenges, especially in a healthcare sector. Patient treatment is very often a difficult and time consuming process, however, in today's society people lack time in every aspect of their lives. Shortage of medical staff is only deepening the problem of long waiting time needed to acquire medical assistance. To fulfil the requirements of patients, governments must invest into finding innovative, efficient and yet affordable solutions that will increase efficiency of healthcare systems.

The aim of this paper is to describe how smart solutions can be implemented in order to cope with challenges of the healthcare sector. The research was focused on organisational and technological problems that are present in this sector. Though there are various existing solutions to these problems, which will be presented in this report, the main part of it will be devoted to description of the proprietary system that was developed by the team named Salutem, throughout the course of European Project Semester. The solution is a distributed system consisting of two main components, a Web Application and a Mobile Application that communicate with each other. The Web Application is aimed to provide a management tool for health centre employees, as well as an online reservation system for patients. It copes with a problem of long waiting time in queues to doctor's office by finding an optimal time of a visit for each patient registered for a given doctor. This is done with use of self-learning Artificial Intelligence (AI) algorithm that is embedded in the system. The Mobile Application is using Near Field Communication(NFC) technology to monitor patients that come for a visit, which includes gathering time data that will be used for the process of AI learning. NFC is also used to instantly provide patient data to doctors once a patient is detected in a doctor's office. The system was created to ensure security and privacy of patients' data.

The report will conclude with a summary of the system, gained experience and possible future development of the solution.

Keywords: smart healthcare, web application, mobile application, Artificial Intelligence, Near Field Communication

1. Team Introduction

In this chapter, an overview of a team is presented. It covers team's symbols followed by short characterization of each member as well as work organization, schedule and communication.

1.1. Team symbols

The team name Salutem, was taken from Latin meaning welfare or health and had direct connection to Smart Healthcare in Smart Cities. In the following Figure 1.1 the logo is presented.

Figure 1.1: Team logo

The letter 'T' was replaced with a red cross, which associated with human health and life. The Red Cross movement also ensures respect for all human being and prevents human suffering.

Above the name, there is a city skyline, which represents the second part of the project – Smart City. It can be clearly seen that the skyline is of a modern and developed city with high skyscrapers.

Another symbol is team's motto, "WE BRING THE DOCTOR TO YOU" (Figure 1.2).

WE BRING THE DOCTOR TO YOU

Figure 1.2: Team motto

The idea behind the motto is to make patients to have easier access for doctors and other healthcare services. The roman font that is used is very simple yet narrow and elegant. The same font was used for team's name and have a connection to Latin language.

1.2. Team presentation and responsibilities

Salutem team consists of four members, Aneta Andrzejewska, Cristian Pérez Llasera, Alexander Likhterman and Jonatan González Jiménez. Each of the team members comes from a different background and has different personality, which creates diversity in the group. Together with Belbin test, it was possible to determine the responsibilities for each member.

Aneta Andrzejewska - Team Leader, Completer - Finisher

Figure 1.3: Aneta Andrzejewska

Aneta studies Information Technology at Technical University of Lodz. Aneta is a passionate programmer who works professionally as a Web developer. Aneta also has high interest in Machine Learning. As a Team Leader, Aneta knows the work of each team member and how to delegate tasks in an efficient way. Apart from that, as a Completer – Finisher, Aneta has the ability to bring the work to perfection and makes it free from errors.

Cristian Pérez Llasera - Shaper, Implementer

Figure 1.4: Cristian Pérez Llasera

Cristian studies networks and systems in Universitat Politecnica de Catalunya. Cristian's specialty is cyber security and as a hobby, Cristian enjoys coding. As a Shaper, Cristian keeps the team moving forward and makes sure it will not lose focus. As an Implementer, Cristian makes good strategic plans that increase team's work efficiency.

Alexander Likhterman - Plant, Team worker

Figure 1.5: Alexander Likhterman

Alexander studies Telecommunication and Computer Science at Lodz University of Technology. Alexander interested in optoelectronics and volunteers in one of the student organizations. As a Plant, Alexander can come up with good solutions and be very creative. As a Team worker, Alexander keeps the team bonded and provides pleasant environment for work.

Jonatan González Jiménez - Monitor Evaluator, Resource Investigator

Figure 1.6: Jonatan González Jiménez

Jonatan studies software engineering at Universidad Politecnica de Madrid. During the studies, developing a software which answers client's needs and design, became Jonathan's highest interest. As a Monitor Evaluator Jonatan has a non-biased way to identify and analyse team's options. As Resource Investigator, Jonatan explores new ideas and opportunities and brings them to the team.

1.3. Organization

At the very beginning, it was very hard for each member of the team to find a specific role. It was mainly due to the diversity of the team, different backgrounds and different work methodologies. As the group started to get to know each other better and have a better understanding about the skills, the group has made a decision that Aneta is going to be the team leader. From there each group member found his place and responsibilities, easily.

One of the team's goal is to let every member to experience EPS. This means that during each meeting, a different Chairperson and Minutes Taker were assigned, depending on the topic.

This approach enabled even distribution of work and let the team members gain more experience from the project.

Through the whole semester, the group tried to have at least one meeting with the supervisor in order to show the progress, get feedback for improvement, and decide about future steps.

A stakeholder analysis was made in order to see by whom the group is mostly influenced and what its potential impact on the project as presented in Table 1.

Table 1: Stakeholders analysis

STAKEHOLDER	**POTENTIAL IMPACT ON THE PROJECT**	**STAKEHOLDER MANAGEMENT ACTIONS**
Project Tutor	Medium	Be active at the Team Building classes to avoid unnecessary individual problems. Present all the important things in an interesting and memorable way.
Mentor	High	Inform about project progress every week. Work hard and on a constant pace.
Team	High	Train ourselves in conflict management Obey the rules presented in a contract. Be ready to sacrifice some activities for the common goal. Respect each other.
Other teams	Low	Keep the project and presentations interesting for non-specialists.
Prospective customers	Low	Contact interested people to adjust the project to their needs. Spend as much time as possible on detailed problem analysis to avoid mistakes during product testing.
Country authorities	Low	Think of legal issues connected with product launch. Prepare cost analysis of the final product.

From Table 1 it can be seen that the two stakeholders that had the highest impact on the project were the team's mentor and the team itself.

1.4. Cooperation and communication

One of the most important elements for a successful project is good communication in the team. The communication includes applications and services that help the team to stay in contact. However, good communication does not end there, the team also has to be well bonded, open minded, and respectful towards each other.

The most used applications and services are Whatsapp, Messenger, Facebook, Google Hangout, Google Drive and Google Docs. For instant and urgent calls, it is possible to use Whatsapp or Messenger, because not all of the team members have Polish telephone number and the calls or messages are made via internet connection. For online meetings, when it was not possible to have a meeting face to face, the used application was Hangouts as its free and requires only Google account, which all the group members have. Google Drive was used in order to store and upload files whereas, Google Docs enabled the team to work on the same document at the same time simultaneously, which was very useful during online meetings.

However, the communication in the group was not good from the beginning and some of the group members had problems with understanding each other due to cultural differences. Nevertheless, as part of team bonding, each of the team members eventually got open and spoke its mind.

1.5. Work methodology

For this project, the team has decided to use Kanban methodology. Kanban works in four steps: Visualize work, Limit work in process, Focus on flow, and continues improvement.

Visualize work – visual model of the work and the workflow, makes it easier to see how well the tasks are going through, in the project. It also makes it easier to find blockages and instantly leads to increased communication and collaboration.

Limit work in Process – In order to reduce the time it takes for a task to go through the Kanban system (To do, Doing, Done) one needs to limit how much unfinished work is in progress.

Focus on flow – by developing what tasks are more important for the team and limiting the work in process, it is possible to optimize Kanban system. Implementation of it will enable work flow improvement and obtain work patterns.

Continuous improvement – It is possible to measure the effectiveness of the team by tracking throughput, flow lead times and eventually improve it.

One application that resembles this kind of work methodology is Trello, which was used in this project. After each meeting, tasks were gathered and divided between the group members. If the task has been finished, it is possible to continue to the next one. This enables transparency of work and real-time communication.

1.6. Schedule

In order to know if the group is in the correct phase of work a Gantt Chart has been made. The Gantt Chart shows the work that has to be focused on during each week as presented in Figure 1.7.

The Gantt Chart was divided into two parts, before the midterm evaluation and after. At the first part, one can see that most of the work was focused on the research, problem definition, and project idea. As for the second part, more focus was put into the technical research, implementation and testing. One element of the project, team bonding, continues through the whole time of the project.

The established rules and working methods helped the group to achieve the goals and without them, it would be much more difficult to manage the project with success.

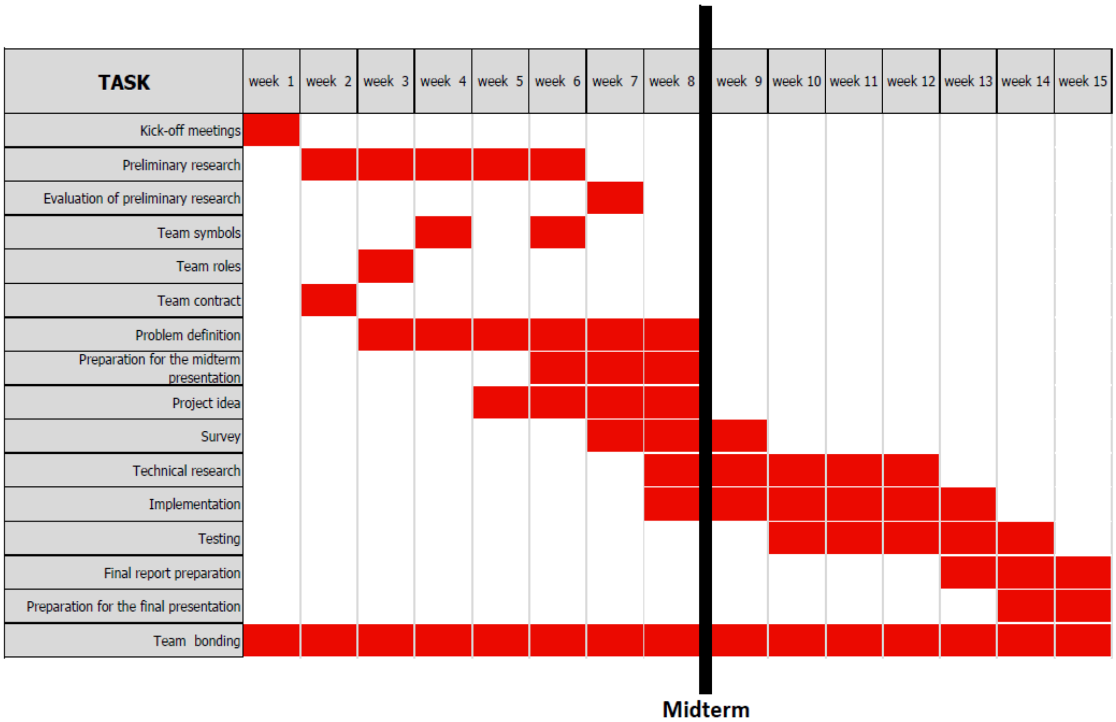

Figure 1.7: Gantt Chart

2. Smart Health Care in Smart Cities

The purpose of this chapter is to introduce the reader about basic concepts of Smart City and Smart Healthcare, in order to ease the reading and have more understanding about the project.

2.1. Smart City

A Smart City vision is mainly to integrate Information and Communication Technologies (ICT) and Internet of Things (IoT) in a secure manner in order to answer the growing urbanization challenges such as transportation, energy, and healthcare [1].

The core of a Smart City is in its intelligent network of connected objects and machines that transmit information between each other and the information and data which is stored on physical or virtual servers (cloud). The data that is received to the cloud from cloud-based applications which is analysed in real time and helps to make better decisions that improve quality of life [2].

In order for a Smart City to successfully address all the challenges, it has to have four main elements. The first is reliable and widely spread wireless connectivity. Low Power Wide Area Network (LPWAN) is applicable to most of Smart City needs as it is widely spread and cost efficient. LPWAN [3] is a wireless communication protocol and its goal is to optimize two main factors. Low Power – Sensors and devices that are used in IoT need to transmit data constantly however, they are not always reachable. This means that their battery life has to be prolonged as much as possible. Wide area coverage – In order for sensors and devices in IoT to be practical, they must be reachable remotely from anywhere. The emerging 5G networks will help Smart Cities and give rise to IoT by providing the infrastructure needed for data transmission.

The second element includes open data. It means that governments, enterprises and individuals must share information and merge it with relevant data that is analysed in real time, which enables better real time decisions. However, it also means that it is more prone to cyber-attacks.

Which brings the discussion to the third element, good security. On the one hand, the connectivity of different systems in Smart City can aid citizens in emergency. On the other, the city itself is vulnerable. In order to avoid breaches in security, strong policies of ID verification and management must be implemented in the ecosystem, to ensure that data is shared only with authorized parties.

The fourth element is flexible monetization scheme. In order to ensure the development of Smart Cities and other players in the ecosystem (such as government or individuals), commerce models has to be introduced. The intellectual property of each member such as government, developers and, integrators, has to be valued and rewarded. Subscription software capabilities, which will be part of IoT solutions, will enable to extract the contribution from each contributor to the ecosystem and hence, create new business models. Subscription based models give the opportunity to monetize software and hardware that were implemented into smart infrastructure as well as spreading capital's expenses [4].

2.1.1. ICT

The term ICT or Information Communication and Technologies is mainly accepted to include all the devices, network components, applications and systems that work together to allow people and organizations to communicate and interact in the digital world. Few of ICT's components include internet-enabled sphere, wireless networks, artificial intelligence and robotics [5].

In traditional approaches to urban development all the infrastructure systems are managed in silos (an isolated point in a system where data is kept and segregated from other parts of the architecture [6]), with limited communication and information sharing among and across government departments and civil society. Integration can be achieved with ICT tools working as bridges between the different physical infrastructures.

ICT has a huge impact on how people communicate, work and live and continues to revolutionize in all the parts of human experience. However, ICT is not developed evenly among all countries, richer countries and individuals have easier access to the internet whereas countries without the proper ICT infrastructure, it remains very expensive or without access at all [7].

The advantages of ICT, brought to companies and businesses a lot of savings, new opportunities and conveniences that vary from automated processes, to big data revolution with which the organizations create insights that drive new services and products

However, with the new abilities and services ICT also creates new challenges and problems to the world. Those may include new levels of crime, displacement of workers without the ability to find other positions, and allowing people to limit their interaction with others.

2.1.2. IoT

The IoT is a combination of ubiquitous communication, connectivity, and computing along with ambient intelligence. It refers to a cyber-physical paradigm where all of the real-world components can stay connected. The IoT gives users the ability to plan each day, and it integrates real physical world elements such as electronic devices, smartphones, and tablets that can communicate both physically and wirelessly.

The IoT helps in managing virtually any number of devices. It aims to extend the benefits of the Internet such as remote access, data sharing, and connectivity to various other application domains including health care, transportation, parking activities, agriculture, and surveillance [8].

Organizations are starting to realize that with the current evolving pace of network, applications and, data organization, it is required to develop a system that can adapt and expand accordingly to those demands which are not only from the inside but also form its surrounding environment.

It is estimated that implementation of smart devices in the industry will be around 50 milliards by 2020 [9]. The pace of this adaptation is five times faster than telephony or electricity. Therefore, the amount of traffic generated and the growing amount of network connections used by IoT devices and applications will go beyond the amount of traffic generated by the internet today.

There are no clear differences between IoT systems and non-IoT systems and therefore, there is no clear model for IoT, but a few. It is most probable that when data is being transmitted to the network from a source such as a machine or other control equipment it is an IoT system. However, using this definition may be inappropriate in all cases and not every network should be considered as an IoT one.

In the following Figure 2.1, a proposed model of an IoT system architecture can be seen.

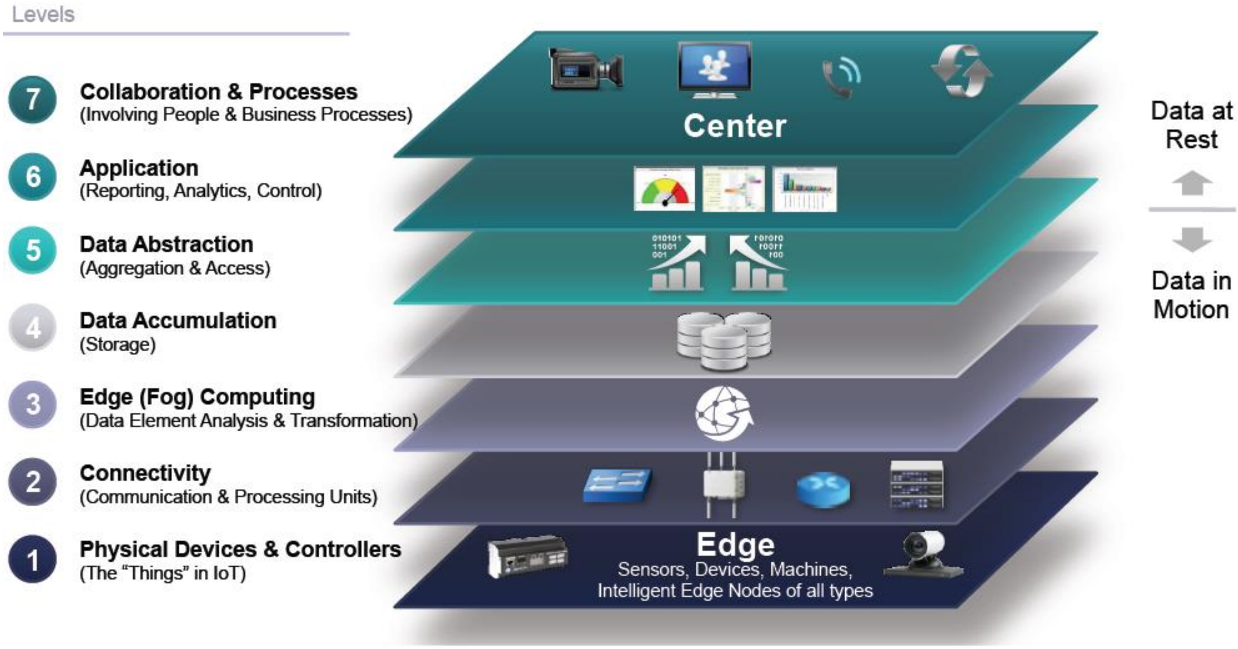

Figure 2.1: IoT reference Architecture src: IoT reference Model White paper 2014

Each layer is described in more details below starting from the bottom and going up.

The first layer consists of physical devices and controllers that are able to manipulate few devices. Those devices include various endpoints which send and receive information. There are no defined rules regarding the shape, size or the physical location of the device as some of them can be as small as computer chip whereas others as big as vehicles however, IoT must support all of them. The amount of the available devices in the market is vast and will multiple itself overtime.

The next layer, Connectivity, is responsible for reliable, timely information transmission. In other words, to distribute the information is such way that it will be relevant for public use. The transmission of the information happens, between the devise in the physical layer, across different networks and, between the low level information processing in layer three and the network. Referring to International Organization of Standardization, the reference model of data communication networks, have multiple functions. In operational IoT model, those functions are distributed among the additional levels it has upon the reference model of data communication network [10].

The Edge (Fog) Computing layer is responsible for converting the network data from layer two, into information that can be stored and processed further in Data Accumulation layer. The principle of IoT reference model is to start processing the information as close and early as possible to the edge of the network, in other words fog computing. All the activities in this layer are focusing on large amount of data analysis and transformation.

After the data has been converted in the third layer, the fourth layer – Data Accumulation is transferring the event-based data to query-based processing. Before the Data Accumulation layer, the data was organized by the time it was sent form the device which mean that the model is event-driven. This process is important as it connects the real time networking world and the non-real time application world.

In the fifth layer, Data Abstraction, the focus is on rendering data and its storage in such a way that will enable to developed simpler and efficient applications. In this layer, the data is also being protected using appropriate authentication and authorization. Moreover, there are additional processes that are happening in the layer, such as gathering data from multiple formats and different sources, assure consistence semantics across them and, confirm that the data is ready for higher-level application [11].

In the sixth layer, Application, is where the information interpretation happens. The software collects data from the layer below and data at rest in order not to operate with the pace of the network.

The seventh layer, Collaboration and Processes, makes sure that all the collected data brings meaning otherwise it does not yield any value. People are using different application in order to help them to make better decisions and for their specific needs. Therefore, the focus is on empowering people and not the application itself.

2.2. Smart Healthcare

In Smart Cities, there is a need to understand how advanced tools and technologies are being leveraged by the medical sector to improve healthcare services. The infrastructure and technology of Smart Cities reconstruct the thinking behind existing healthcare systems (mobile-health, electronic-health for example) and telemedicine to create a new and comfortable ubiquitous concept that is called Smart Health [12].

Moreover, Smart Healthcare integrates ideas from ubiquitous computing and ambient intelligence applied to predictive, personalized, preventive and participatory healthcare. Smart Healthcare is strongly connected to the concepts of wellness and wellbeing and includes a large volume of data, collected by large amounts of biomedical sensors, genomic driven big data, payer-provider big data and social media data actuators, to observe and predict patient's physical and mental conditions [13].

Smart Healthcare is a new but promising field of study at the intersection of medical informatics, public health and also business, alluding to intelligent healthcare services or enhanced cognitive capabilities through the IoT. In an elaborate sense, Smart Healthcare defines not only ICT development, but also a state-of-thinking, a way of lifestyle and approach, and a vow for connected entities to improve healthcare facilities at home, city, country and globe with the aid of Artificial Intelligence.

From the last decade, health-related data has been exponentially rising due to the application of smart devices and social media in healthcare services. The big data in healthcare mainly includes 3D imaging, medical records, radiology images, genomics and biometric sensor readings. Moreover, these generated raw sequencing health data are equal to approximately four terabyte each [14].

Although the health data volume is excessively large, advancements in data management and processing techniques, particularly machine learning, and AI are enabling innovative platforms to be developed for more effective and efficient management and manipulation.

In Smart Healthcare systems, data collection and modelling processes are being conducted at high velocity, almost in real time, which means that there is a rising prospect for big data analytics in healthcare to give immediate feedback on a patient's surrounding environment.

3. Related Work

During the research part and the solution implementation, the team consulted with few experts in order to understand better the problem that is investigated in the project and improve the solution. Those experts have direct connection to healthcare. Among them are doctors, nurses and, other medical staff.

In order to understand the problem from medical staff point of view, Anna and Mikhael Likhterman were interviewed. Both work in two different governmental hospitals in Israel for more than twenty years. It is important to state that healthcare in Israel is considered to be well developed among similar countries.

Both presented that the quality of treatment for patients gradually reduced through the years, although medical equipment and the technology were improved. The main reason behind is the lack of medical personnel. The amount of medical staff per working shift is decreasing from year to year and consequently, the work per one doctor or nurse is increased. This creates huge amount of pressure and brings exhaustion on the medical staff, which increases the probability of human errors and reduces the time spent on each patient.

During this short time, the medical staff needs to realize patient's problem, fill patient's documents and perform other observations in order to give a proper analysis and treatment. Consequently, in some cases, the prescribed treatment is not helping and during this time, patient's health is getting worse.

For a solution, they would be very interested in an application or a system that helps with the bureaucratic work. This will increase the time which is focused on patient's problem and not on filling up documentation.

From the patient point of view regarding the waiting time for the doctor, Mikhael stated that in most of the cases waiting in a queue for the appointment can take in average half an hour. However, some doctors spend significantly more time on the patient, which makes the waiting time as long as few hours. Patients that are not aware of it may spend their whole day in a queue if theirs' visit is booked for late hours. As someone that has a busy schedule every day, Mikhael books his appointment only at the begging of the day however, it is not always possible.

Another expert that the team has consulted with is the director of ZOZ Polesie in Łódź, Doctor Budzinski. The main part of the discussion included presentation of the solution and improvements that can be done in order to make it more efficient.

Firstly, Dr Budzinski pointed out that every doctor is different, and works differently even if they are form the same specialization. Moreover, from the operational point of view, healthcare centres and hospitals may work differently even though they are located at the same region.

Dr Budzinski also introduced the healthcare system in ZOZ Polesie and the challenges it faces in order to give the patients high quality treatment. A company called Medica implements the system in ZOZ Polesie. The system allows medical personnel to view medical history of a patient, prescribed drugs and sick leaves. On the other hand, patients can view available time slots for doctor's appointments.

From the discussion, the team could also understand the way ZOZ Polesie receives the money from Narodowy Fundusz Zdrowia (NFZ) in order to pay the doctors. The system works in that way, that each citizen in Poland is paying taxes to NFZ. The collected taxes are then converted to points with which different medical procedures are evaluated. Consequently, the more points there are the more procedures are available for the community. However, as those points are limited so does the different medical procedures. This results in long waiting times for special doctor appointments.

Another problem that Dr Budzinski introduced is that some of the patients are showing up to the appointment without any major problem. On the other hand, others have urgent cases that cannot wait and the doctor has to accept them. Although there is a system for registration, people, especially elderly, prefer to book a visit on the same day.

Finally, Dr Budzinski claimed that the problem of time management is global and connected to lack of medical personnel and other issues regarding contracts with NFZ.

From the meeting, a few conclusions have been made. Firstly, the work has to be focused on General Practitioner (GP) rather than special doctors as for GP, the time is repeatable for given age groups and disease units. The diseases are also repeatable for a given patient hence it may be predicted. Secondly, the implementation of the solution has to start from microscale for one healthcare centre and possible to interpolate it to macroscale later.

The discussion with the experts gave a better understanding of the problems healthcare system is facing each day. The lack of medical personal is a problem all over the world and there is high need to develop smart solution that will help with the organizational issues.

4. Problem Setting

In this chapter, research and findings concerning healthcare and Smart Healthcare were introduced. Later the problem statement is described and, Salutem's goals and vision statement. Finally, few of the existing solutions were mentioned.

4.1. Current problems in Healthcare

In Healthcare and Smart Healthcare there are four main problems all over the world. The chapter starts with financial problems, continues to technological, logistical and finally organizational.

4.1.1. Financial Problems

The financial crisis that has been all over Europe in the late 2000's, has effected different sectors in overall living standards and especially healthcare and some governments have even worsen the already difficult economic situation with their actions. From the healthcare point of view, those actions include closing down hospitals and clinics and cutting off the payment for medical personal. Consequently, it led to low healthcare services and spread of infectious diseases. Moreover, citizens reacted to the austerity with increased suicide rate, lower living standards and, higher difficulties treating chronic diseases such as cancer. However, countries, which had strong protective policies regarding healthcare did not experienced major changes during the period of crisis [15].

For this period, the government did not cover enough of the healthcare expenses. Only 65% of each paid dollar was directed for the purpose it was supposed to be. One explanation is that the rest was wasted on bureaucracy, the other is that there are indirect costs such as transportation, food or accommodation which had to be paid by the patient [16].

Another situation that could be observed is that having an insurance from a private company which is pre-paid, is more efficient than waiting for a solution from governmental insurance. In other cases government does not cover specific healthcare expenses such as psychiatrist whereas, private insurance does. Moreover, it could be seen that as the time gets closer and closer to the economic crisis, the healthcare costs are being covered more and more by the citizens themselves.

In another continent, India has some of the world's best healthcare however, it is not delivered through the public sector therefore, nearly 600 million poor citizens cannot afford high standard treatment. This lack of fairness, puts India close to the bottom of the UN Human Development

Index (HDI) [17] .India's HDI rank is 131st out of 188 countries whereas, China's rank is 90th. A recent report by the World Healthcare Organization reveals that nearly 70 percent of India's population spend most of their income on healthcare therefore, each year 40 million citizens are becoming extremely poor. India's spending on public funding of healthcare is around one percent of its GDP which is less than almost every other country in the world. Compared to China, which has similar lack of fairness, China publicly funding over 40 percent of the medicine cost whereas, India funds ten times less for pharmaceuticals [18].

Consequently, India has the highest population of stunted children due to malnutrition. In the rural areas, the healthcare facilities lack of basic needs such as clean water and some of the hospitals are not registered [19].

Opposed to India and other poor countries, United States is one of the countries that have high income however, the medical costs that are paid by the citizens themselves are extremely high and put many people into poverty. In United States, hospitals charge as much as they can in order to maximize revenue and the main reason is not medical but fiscal. Moreover, hospitals are the most powerful players in healthcare and the pricing system has almost no regulations in the private market [20].

After each treatment in hospital, computer programs and billers use the Chargemaster price list to translate the services of the patient into a price. A day spent as a patient at American hospital, costs on average more than four thousand dollars, which is five times more the charge in many other developed countries. In many hospitals, emergency room visits typically include separate charges for doctor's services and for supplies, as well as the charge for entering the hospital [21].

Although the reasons for financial issues in healthcare vary from country to country, they are all over the world, even for developed countries such as United States.

4.1.2. Logistical Problems

Financial problems in healthcare also has an impact on the logistics and the availability of medical equipment. In Poland, the availability of certain medical equipment in hospitals and clinics depends on the type of the device [22]. For instance, the availability of Computed Tomography, in which X-rays are being used in order to create detailed pictures of areas inside the body, was relatively good compared with Magnetic Resonance Imaging or Position Emission Tomograph. The number of made scans using MRI machines or CT scanners per

1000 inhabitants is significantly lower than in Czech Republic, Hungary or Spain as can be seen on Figure 4.1.

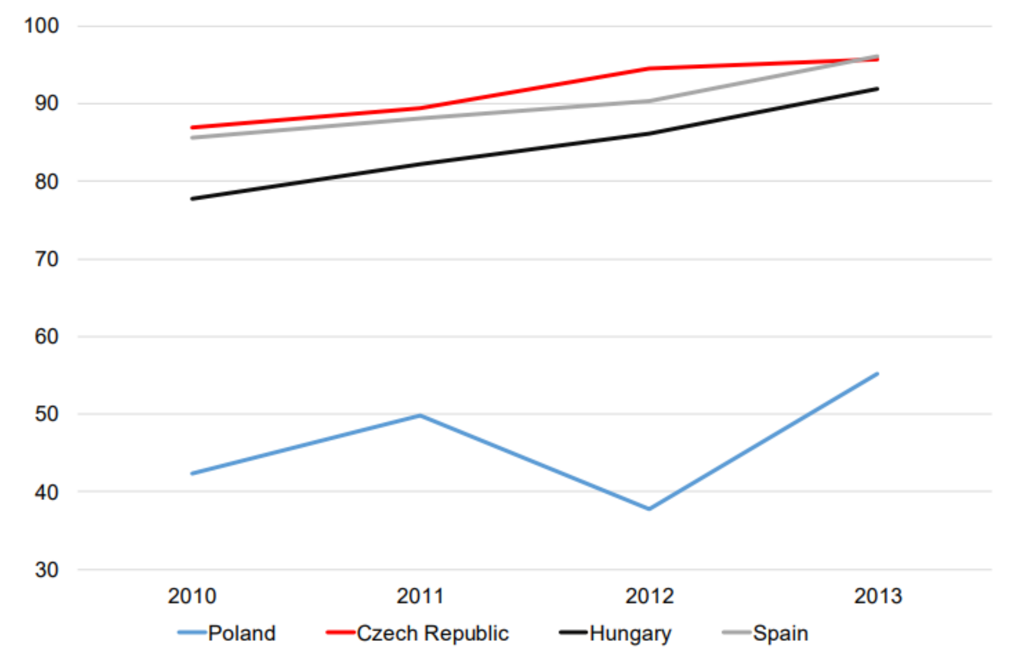

Figure 4.1: CT Exams per 1000 Inhabitats (Central Statistical Office, Ministry of Health, OECD, Blackpartners)

From Figure 4.1 it can be seen that in the year 2012, Poland had the lowest number of CT exams per one thousands inhabitants. However, in the following year the number of scans increased by fifty percent. Impressive as it is, Poland is still lagging behind other analysed countries almost by half.

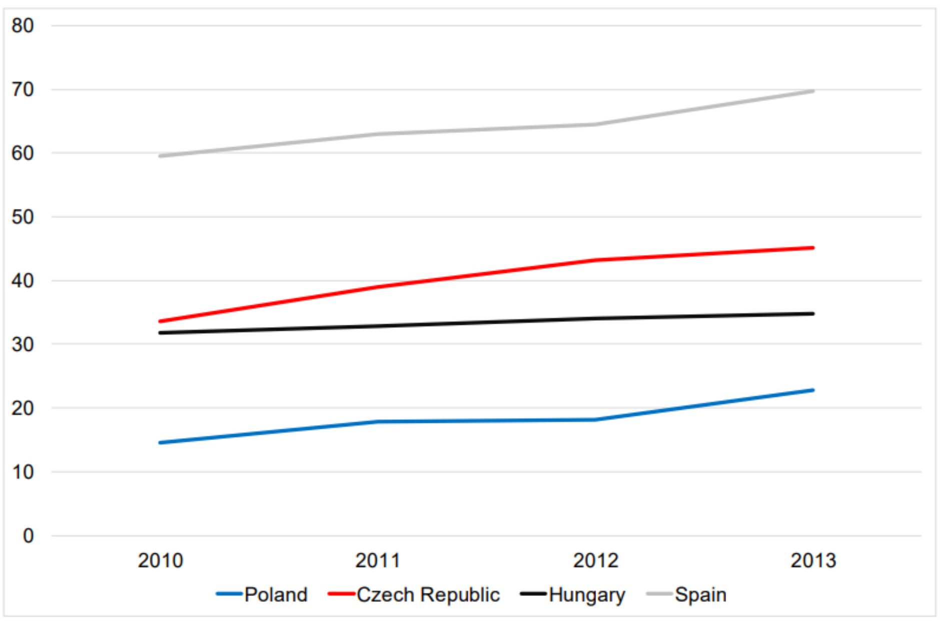

Figure 4.2: MRI Exams per 1000 Inhabitants (Central Statistical Office, Ministry of Health, OECD, Blackpartners)

Figure 4.2 shows no better results concerning MRI scans per one thousand inhabitants. Poland has the lowest number of scans whereas Spain has the highest, just about 75 scans per one thousand inhabitant.

The reasons behind this situation include improper use or maintenance of equipment or low allocation of budget towards medical imaging. In other situations, there is not enough space for new equipment or qualified personnel.

4.1.3. Organizational Problems

Healthcare system is also facing organizational problems such as medical staff shortage, long patient waiting time for the doctor and, time spent on bureaucracy by medical staff rather than on patient treatment.

Healthcare professionals are one of the pillars in the entire healthcare system. In 2014, Poland had an increase in the number of medical professions from previous years [23]. Among the entitled persons who practice medicine are doctors, dentists, nurses, midwives and, pharmacist and in total just below half a million of medical personal. However, the saturation of medical professionals in Poland is very low and there is high probability it will get even lower. According to OECD (Organization for Economic Co-operation and Development), Poland is below the average when it comes to the indicator of physicians per one thousands inhabitants with the average stands at 3.3 and Poland's indicator stands around 2.25. This problem will worsen with the fact that Poles are the fastest ageing population in the European Union.

The reason for this situation can be divided into two parts. The first is associated with retiring older generations and migration of medical professionals to Western Europe. The other is associated with the fact that new graduates in Poland mainly driven by the influx of new medical staff. In the academic year 2015/16 [24] there were just above 20 thousands new graduates with medical degree in Poland compared with Germany, which had four times more. Taking into account the population differences between Germany and Poland, Poland has to double its medical students to get even with Germany. The demographic structure of medical professionals in Poland is in very bad shape, as just more than a half of the total personal staff are above fifty years old, and just about eighth are over 70 years old. In 2016, the average age of doctor with specialization reached 54. Another big threat to the stability of medical staff in Poland is that around 40 percent of graduated doctors are thinking moving abroad due to financial matters, bureaucracy at work and difficulties in obtaining specialization.

Another issue is long waiting times whether it is waiting for a specialist appointment, which can take months, or time spend in the clinic, which sometimes can be even more than an hour. According to data analysis, the more specialists there are the shorter waiting time. Moreover, according to Watch Health Care Foundation average waiting period to visit a general surgeon, paediatrician or midwife is less than two weeks, while queues in endocrinology or orthodontics are more than nine months. The waiting time for the doctor in the clinic or hospitals can take up to several hours. According to Professor Klassen from Brock University in Canada, medical appointments run late almost everywhere due to a simple fact that the doctors have only a small idea of what patient's case is going to involve [25].

From the discussion with director of ZOZ Polesie, it could be understood that there are more patients that are treated by the doctor than on its list. This is due to the fact that some patients have emergency and their appointment is urgent and cannot wait.

Another organizational problem in healthcare is bureaucracy. Bureaucracy causing slow problem solving, discouraging form innovation and developments on new solutions as well as, concentrating huge amounts of time on issues which include politics or trying to understand how the whole system works.

Unfortunately, most of the managers do not see how bureaucratic issues influence on the work of the frontline medical staff. Due to the paper work that Electronic Medical Records require, the time that is spent on the patient itself is roughly one third of the total time with the patient, those requirements also include authorization from insurance companies which takes great time. Cutting the time spent of those bureaucratic tasks by two can also reduce the costs, and then the doctor will have more time for professional and patient centred work [26].

Although patient charts are very important for communication between the medical personal, some of them require filling information that is not about patient care. Moreover, those charts are usually the same although patients have different problems and instead of improving patient's situation, it makes it worse. Other cases where the background of the patient has to be checked before prescribing new medication, require doctors to keep track of their patients, such as dosage, side effects they had and for how long they were taking it. Some of the records are done on paper and require day-to-day communication with the patient and can get lost or forgotten because of the human factor [27].

4.2. Current problems in Smart Healthcare

Healthcare is one of the economic sectors in which implementation of effective IT solutions is very challenging. Implementation of IT systems and solutions in healthcare is lagging by ten to fifteen years compared to other major industries [28] . Among of those challenges are security and privacy of patient's data and compatibility between different devices. Smart healthcare is a very complex environment. Healthcare information comes from a variety of different sources and technologies, starting from administrative and financial technologies, biomedical technologies and up to electronic records and hospitals or clinics databases. Therefore, proper management of the system is needed in order to have a positive impact on the patient.

Technical challenges arise from lack of standardization and it might take years until data systems in healthcare will be able to communicate and transfer data between each other smoothly. This also permits the ability to share freely information between medical facilities and healthcare personal. It is estimated that, a healthcare system that is fully standardized and implemented can produce a net gain value of around 80 milliards US dollars a year [29]. The file formats and structure need to be consistent enough so they could be read by different software at different facilities.

However, due to security issues, companies have started to develop their own file formats in order to ensure security but this makes the exchange of information significantly more difficult. Federal Health Insurance Portability and Accountability Act (HIPAA) provide one of the guidelines for patient's data privacy and security. It requires harsh procedures in order to safeguard patient's data in any case including recovering the data in emergency or disaster situations. Moreover, Health Information Technology for Economic and Clinical Health act (HITECH) supported HIPAA and even increased the penalties and requirements at the event of security failures [30].

Another challenge for Smart Healthcare systems is the increasing amount of patient's data and storage of it. Existing data infrastructures are struggling to hold and manage properly patient's data as it comes from variety of different sources, digital images, electronic records and, data from telemedicine technologies. As a result, healthcare data doubles itself every 18 months. Few of the medical organizations have started to implement new storage tactics that include on-site, off-site and Cloud storage. However, without efficient management of those huge amounts of growing data, increasing the capacity will not solve the problem but it will only delay it [31].

There are two more factors that are delaying the implementation of IT solutions and project in healthcare. First of all, it is very hard to predict the financial return from implementation of IT systems. Usually businesses are expecting quick return from investments however, the return from investment in healthcare is very slow. That is why managers and decisions makers will not approve projects and changes that will not recover quickly from input costs. Moreover, due to government regulations and growing energy needs, costs of IT solutions are getting even higher [32]. The other reason for the delay is the traditional way of work and it is well known that making changes in your habits is very difficult. The usual belief about healthcare services is the importance of the relationship and interaction between the doctor and the patient and as a result, the work was done through a basic filling form. This requires form the doctor to spend essential time on storing and managing patient's data.

In many ways, healthcare is one of the most rapidly changing sectors. With pressures from a growing influx of patient data, legal requirements for strict privacy and security, rapidly advancing clinical technology, increasing energy costs, and other factors, healthcare IT personnel are looking for innovative management and storage solutions. Cloud services, including infrastructure as a service and data as a service, are among the solutions meeting the growing IT needs of the healthcare industry.

4.3. Problem Statement

According to Population Ageing, Labour Market and Public Finance in Poland [33], by 2050 half of the population in Poland will be over fifty years old therefore, as the working age population is decreasing the improvement in life standards is much more difficult [34]. Moreover, increase of the elderly population also gives a rise to chronic diseases and illnesses. That means that the healthcare system will have to face organizational and technological changes in order to answer the aging population needs.

As stated by the OECD, Poland is below the average when it comes to the indicator of physicians per one thousands inhabitants with the average stands at 3.3 and Poland's indicator stands around 2.25. This problem will worsen with the fact that Poles are the fastest ageing population in European Union. This means that the healthcare system in Poland has to make changes in its organization and increase the effective time spent on the patient.

Currently not all hospitals and clinics in Poland are able to share patient's data between each other and most of the data is in paper form. There is a high need for digitization of the data and the overall healthcare system. This means that, artificial intelligence and data mining can provide help with diagnostics, minimize errors done by the human factor and, speed up organizational processes.

4.4. Goals

In order to answer the problem successfully the team established few main goals. One of the goals is to develop a system that can be easily implemented in the current healthcare system. As the healthcare system in Poland and around the world is dealing with financial difficulties as well as various regulations from the government regarding data privacy, it is important for the solution to be implemented in such manner that will answer those needs. The system has to work in such a way that the patient's data is stored in a secure manner. Data privacy is a very important, especially when it comes to health issues. Moreover, limiting specific data between healthcare personnel is also part of the security as not every medical staff needs to have an access to the whole database of the patient.

Another important goal of the team is to ease the administrative tasks and reduce the bureaucracy work of the doctor in order to give more of the total spent time during the appointment on patient's problems. This objective also includes reducing patient's waiting time for the appointment and increasing the quality of medical service.

Finally, the objective of the project was also to develop new skills and enhance existing knowledge. To use the diversity of the team and the knowledge of each other in order to work together for a common goal while attending team building and management activities.

4.5. Motivation

As young generation that is just starting to plan its way around the world there are high concerns about health and future especially after the research that was done through this project. Moreover, those issues are not only concerning the group or relatives around the world, but also millions of people that are seeking for a better medical care.

As health is essential part of life, there is a high need to reduce time spent on queues for a doctor and receive a proper treatment.

4.6. Vision Statement

The group's vision is to deliver better patient's experience and treatment in medical centres and optimize the time spent in queues and on organizational work using affordable methods.

4.7. Existing Solutions

There are three widely known technical solutions in regard to the problem. The first solution that was introduced by Poland's ministry of health includes introducing new healthcare system followed by name "eZdrowie". Although there are still no concrete details, it will include new solutions that haven't been on the polish market before such as e-Prescriptions or e-Appointments ('e' stands for electronic). Future implementation of the system include online booking and the option to check your waiting time.

One of the private companies in Poland, Comarch, has developed an application called medNote which provides the medical professional with patients data and minimize the time spend on paperwork. The application includes easy prescription writing, referrals and, storing essential information of the patient such as personal data, records of visit and even test results. Another ability of the application is to book an appointment directly by the doctor or the registering person. The application also gives access to medNote Scanner which can scan paper documents and store them with the rest of patient's data and making it available to be viewed via smartphone [35].

A solution regarding long waiting times includes no change in the system or additional technologies. Professor Klassen, came up with a formula that could reduce the time patients spent in the clinic waiting for their appointment. His research included numerous clinic sessions and he finds out that if the appointments are closer to each other at the beginning of the appointment session and then spread the appointments a little apart from each other, eventually one can bunch the appointments again and let the doctor to go home on time.

5. User documentation

Throughout the EPS not only the problem was defined but also a prototype for a solution for this problem was created. This and the next chapter of this report will describe in details what is the solution of the already defined problem. Before presenting the system architecture and implementation details it is important to know what are the functionalities of the system. One of the methods of describing what the system does is connected with use cases, which are actions that can be performed by a user, defining the interactions between a user type and a system. This chapter will include overview of the use cases of the system together with figures presenting how those use cases can be reviewed from a user perspective. The whole system is composed of two main subsystems, which are interconnected to fulfil the requirements of the project. Those subsystems are Web Application and Mobile Application. Following subchapters will describe user documentation of each of these subsystems separately.

5.1. Web Application

As it has already been presented, the Web Application is one of the two main components of the created system. This application was created for three roles, which are Patient, Doctor and Administrator. Figure 5.1 presents the use case diagram of the Web Application, that contains main actions that each user can perform in the system. Some of the actions of one actor are dependent on actions of the other actors.

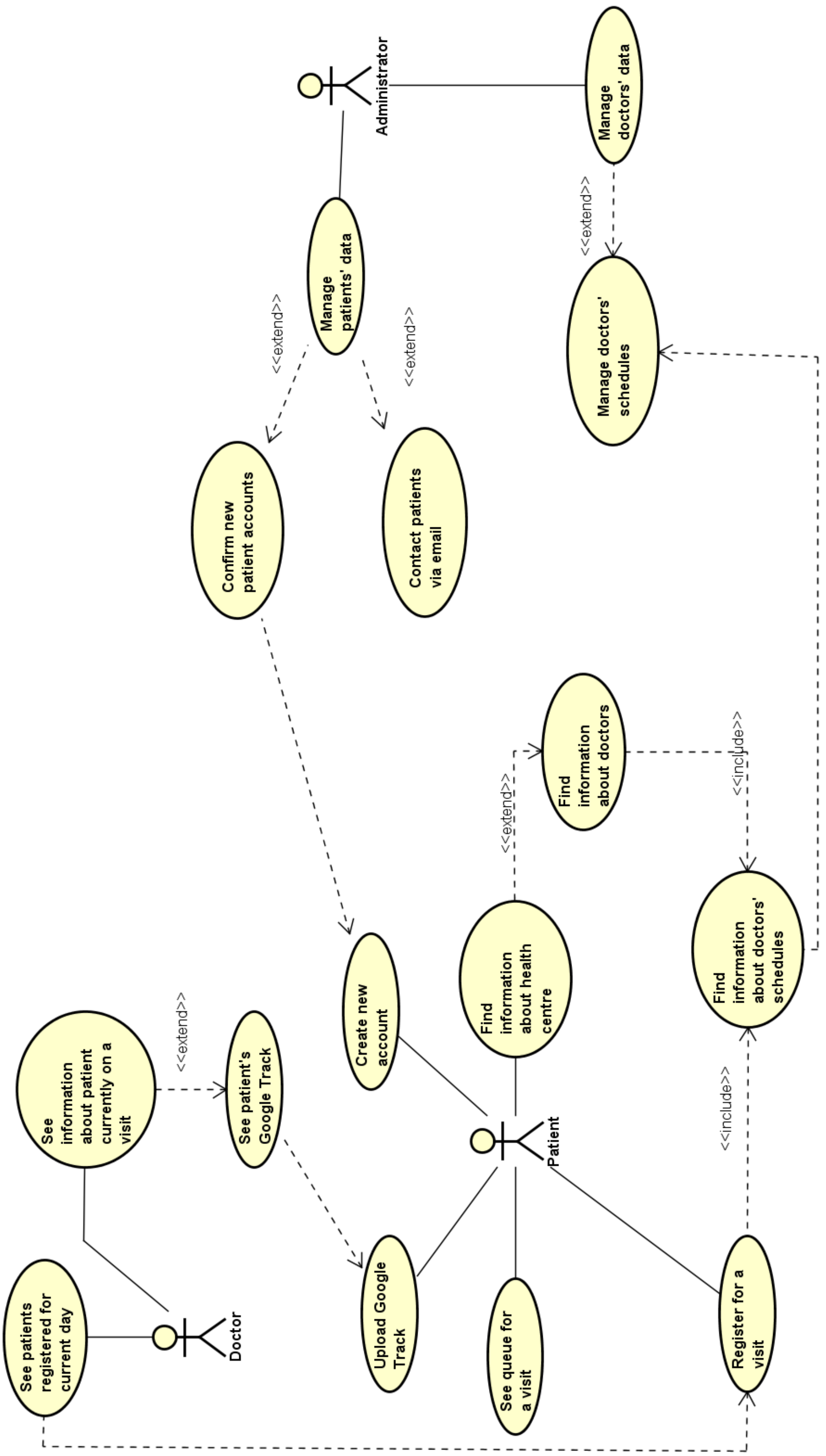

Figure 5.1: Use case diagram of the Web Application

Following subchapters will describe each role separately.

5.1.1. Administrator

The Administrator is one of the three actors that are present in the system. As it was shown on the use case diagram this role has two main responsibilities, which are doctors' and patients' management. Figure 5.2 presents a view of the application of managing patients. Only a user with role Administrator is able to see this view.

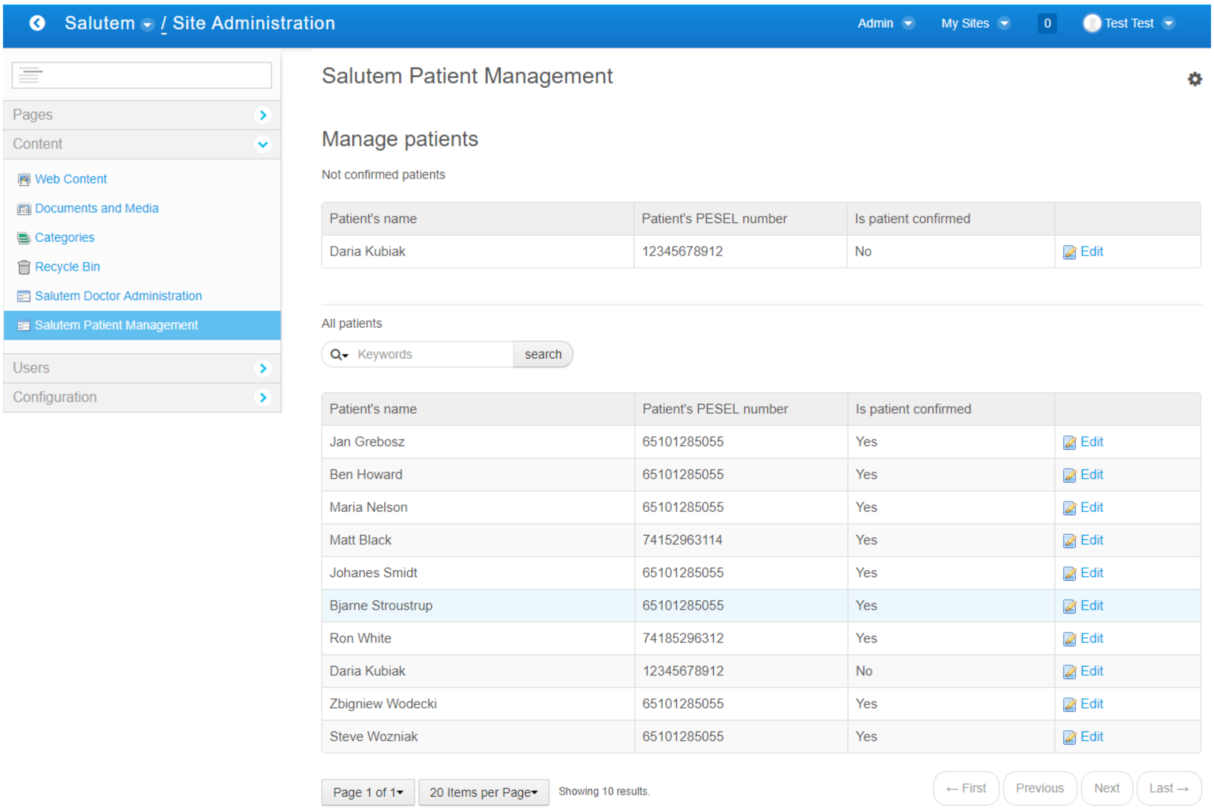

Figure 5.2: Patient Management – main view

The main view of this application is composed of two lists. A list at the bottom of the page contains all patients registered for this particular clinic. Apart from patient's name there is also information about PESEL number and whether a patient account has been confirmed. For better user experience of the system a possibility of searching was implemented. By default searching via keywords is shown. This means searching by name, surname and PESEL number at the same time. If there is a need of more advanced search, a user can click a dropdown with a search icon and more options of searching will appear, as presented in Figure 5.3.

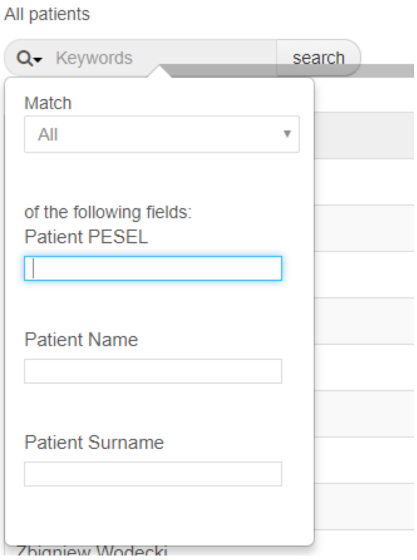

Figure 5.3: Patient Management – searching

The second list on the main view is composed of only not confirmed patients. It was done that way so that the administration can immediately see whether new patient accounts have been created. If an edit button is clicked the user gets redirected to the second page of the Patient Management application. This page differs slightly depending on whether the patient account was confirmed or not. Figures 5.4 and 5.5 present both views.

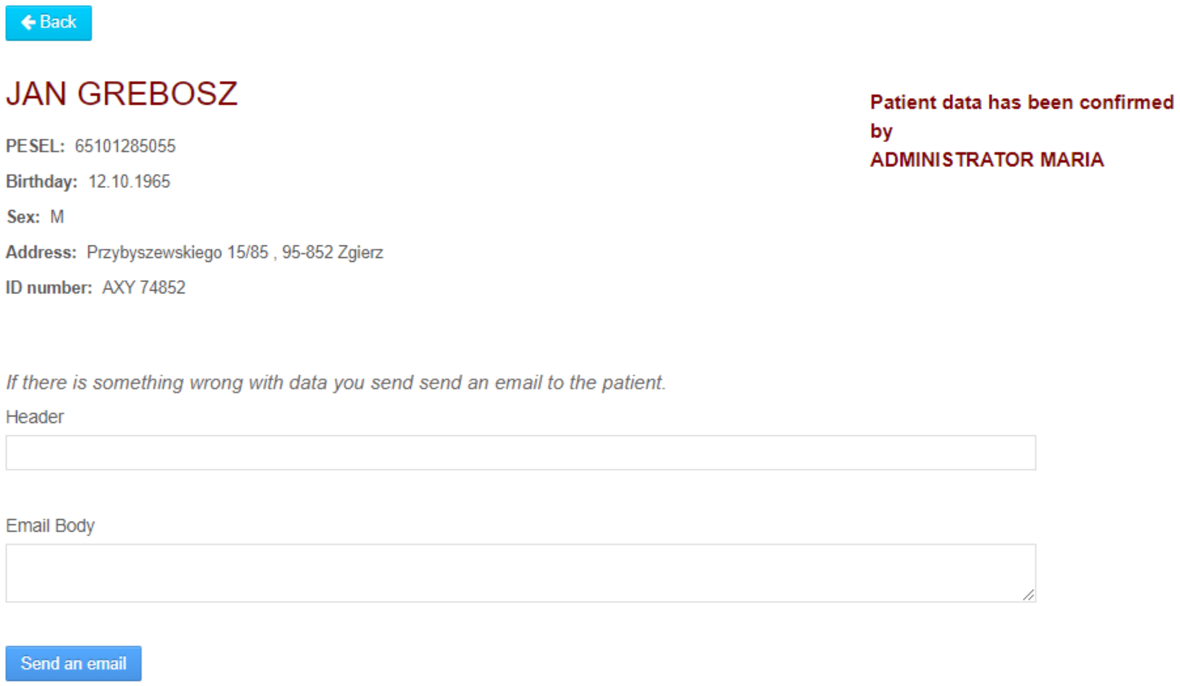

Figure 5.4: Patient Management – confirmed patient view

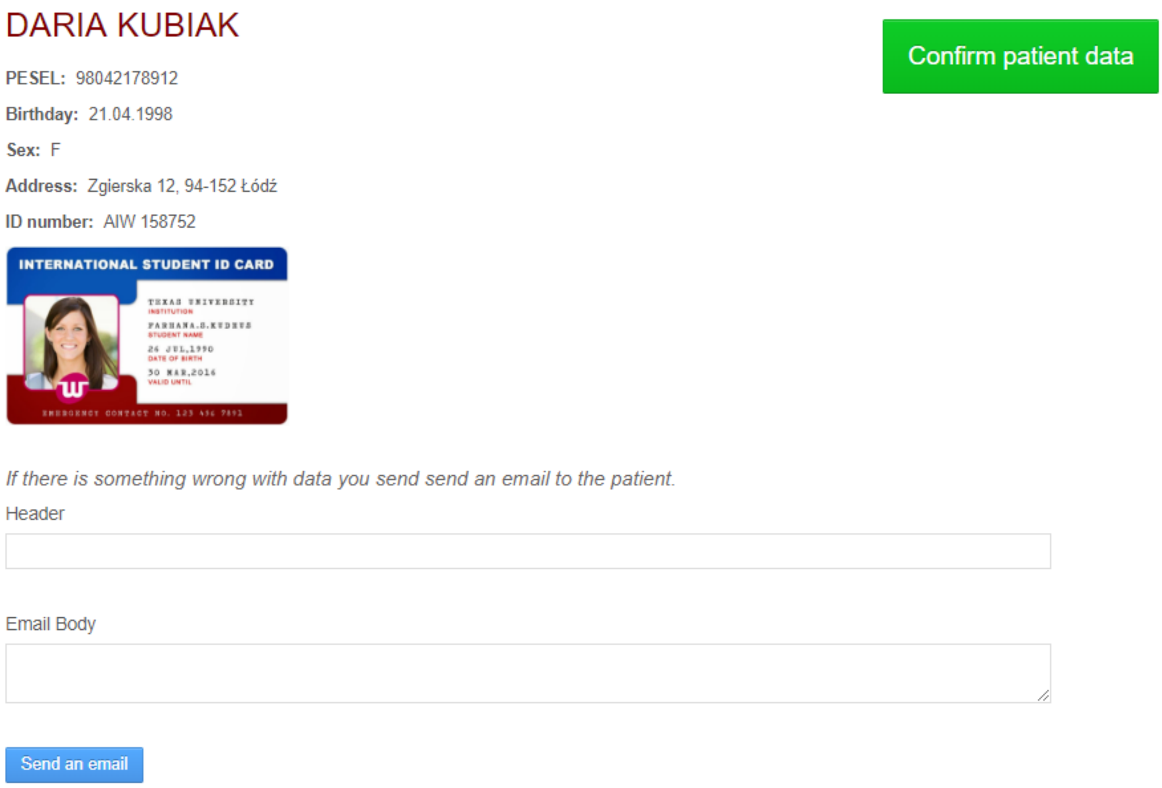

Figure 5.5: Patient Management – not confirmed patient view

As depicted, the patient view contains basic information about a patient such as full name, PESEL, birthday, sex, address and an ID number. If the patient account has not been confirmed yet there is also an image of an ID presented. Once clicked, this image is shown in a full page. When an administrator is certain that given data are correct and fit what is presented on the ID they can confirm such patient account by clicking a green button. When the account is confirmed, instead of the button there is information about who confirmed the account. This is how use case "Confirm Patient Accounts", which is one of the use cases extending the main use case of managing patient data, was realised. The next extending use case is "Contact patients via email". In the edit view of each patient there is a section for email sending. An administrator can fill in the header and the body of an email, and the rest will be done automatically after clicking a button send. Figures 5.6 and 5.7 present a part of the view responsible for email sending that is filled with example data, and a received email respectively.

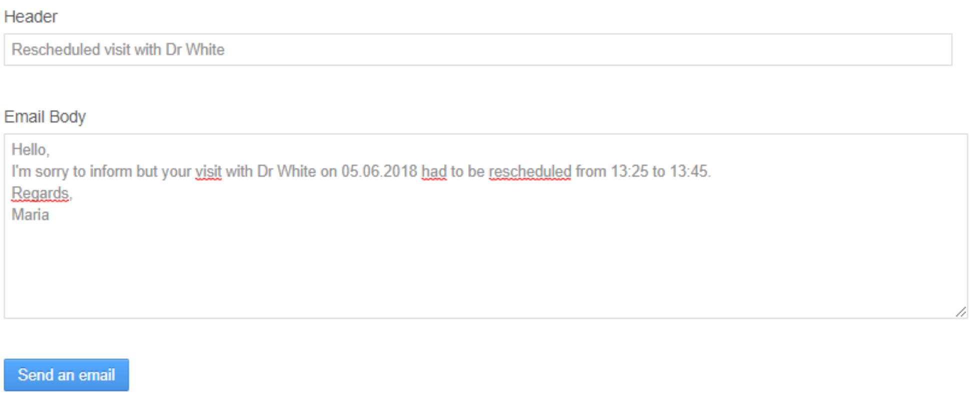

Figure 5.6: Patient Management – sending email

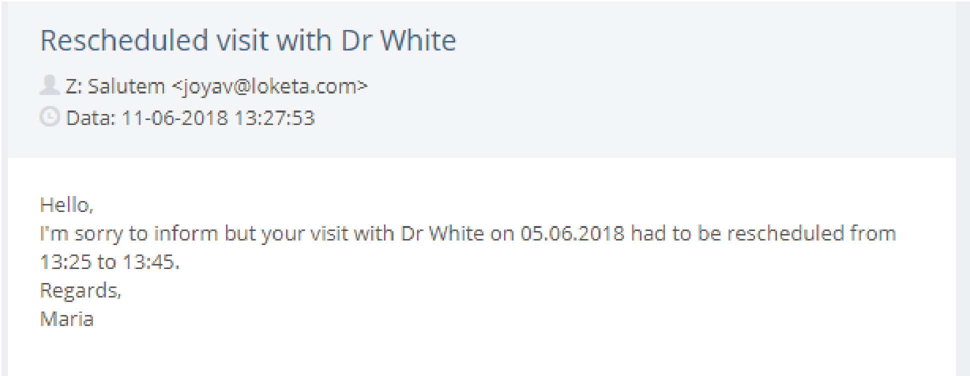

Figure 5.7: Patient Management – received email

This is how the first use case for an Administrator was realised. Another use case concerns managing doctors. For this functionality another plugin was created, named Salutem Doctor Administration, which is similar to the previously described one. Again a list of doctors is available together with possibility of searching.

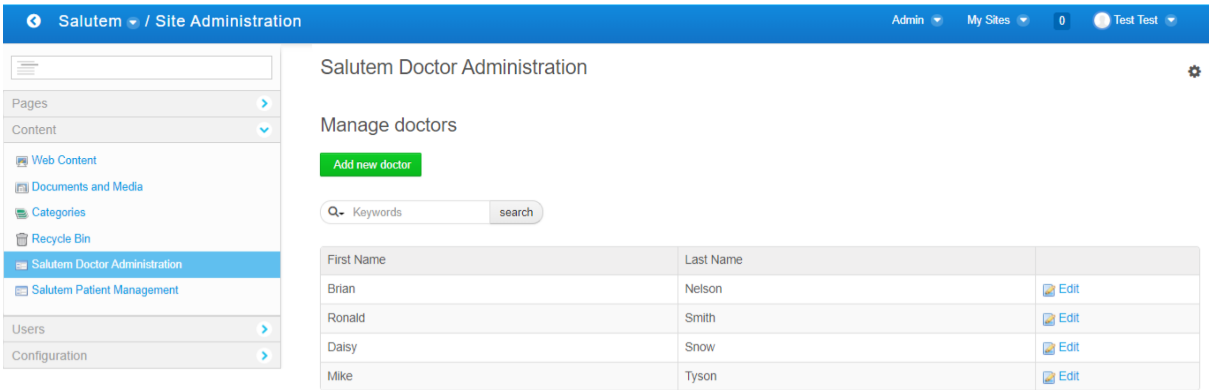

Figure 5.8: Doctors Management – main view

What differentiates this component from the previous one is the view of an individual doctor. Apart from an image depicting a doctor, which is then shown for patients visiting the website of the clinic, there is a possibility of defining a schedule of a doctor. Figures 5.9 and 5.10 show the edit doctor view with the latter one being devoted to the schedule management.

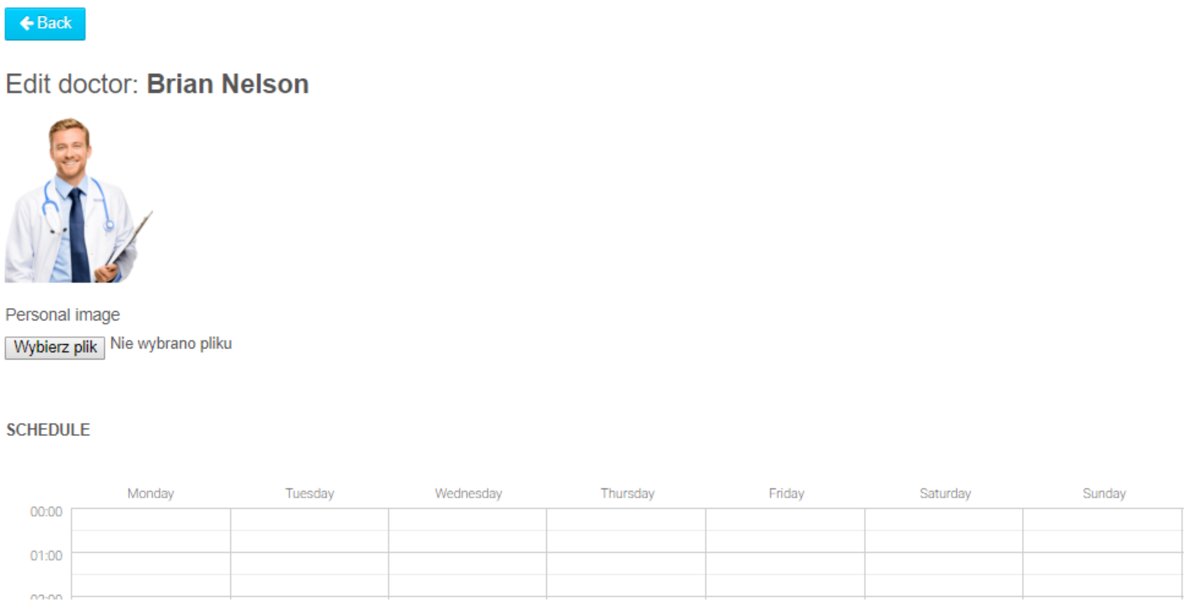

Figure 5.9: Doctors Management – doctor's view

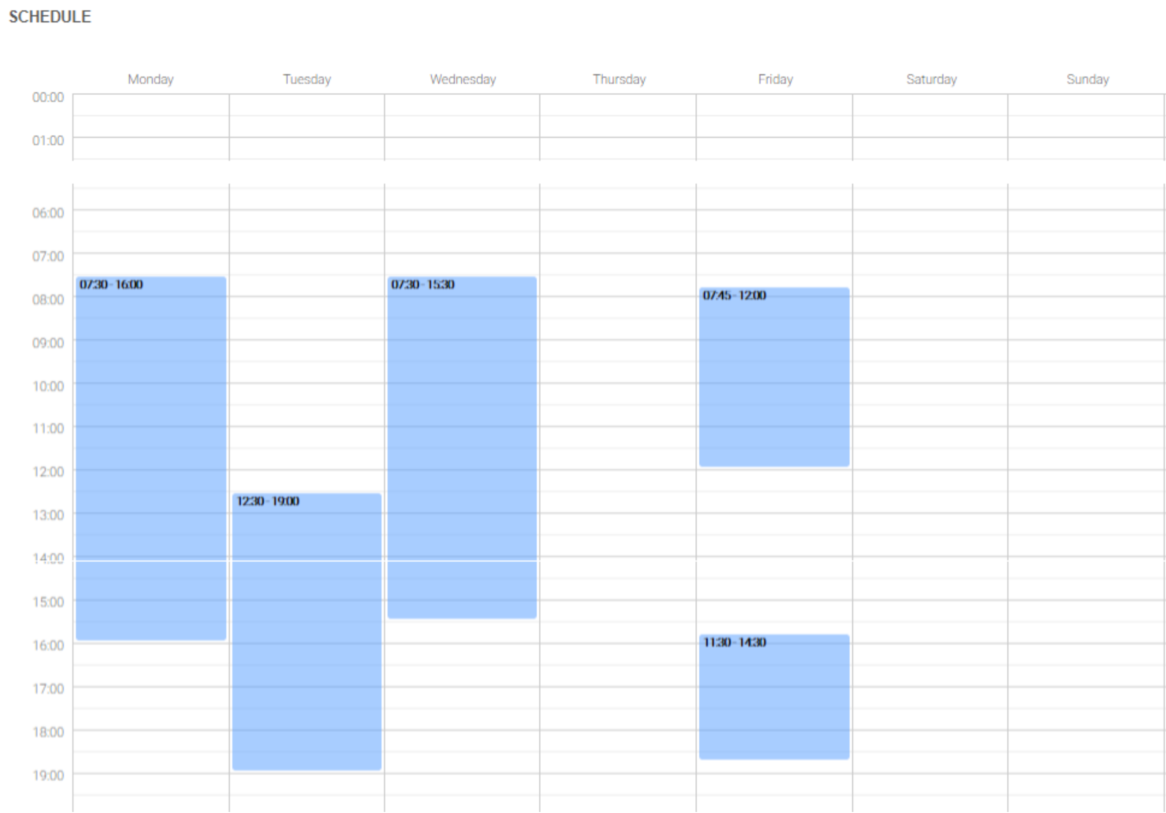

Figure 5.10: Doctors Management – doctor's schedule

This module was implemented to allow good user experience, hence, creation of schedules is done intuitively and in a visually attractive way. For each day, one or more blocks representing schedules can be created. Each schedule can be copied, deleted and changed by moving the block and extending or shrinking it. When save button is clicked, automatically the schedules of the doctor are updated and made available for patients.

These are the functionalities of the administration that were implemented in the project.

5.1.2. Patient

The next actor identified in the Web Application is a patient. This role according to the use case diagram has 5 main use cases, which are creation of a new account, finding information about the health centre which is extended by finding information about doctors, registration for a visit, seeing a queue of particular visit and uploading a Google Track. Each of these use cases will be described in this chapter, together with views from the application representing how those use cases were fulfilled.

First of all, a potential patient has to register in the application, as only registered users have access to information and functionalities of the system. Account creation is a twostep process. First, an ordinary account for the system is created, which allows seeing some restricted information about the clinic e.g. contact to employees or doctor schedules. When a user decides that he or she wants to become a patient of the clinic, then a patient account can be created. This twostep process was designed in such a way to make it scalable to more clinics which want to share patient data. Then, each person would have individual accounts on each clinic website but one common patient account with shared private information. There are three statuses of a patient account: not a patient yet, not confirmed patient and confirmed patient. When a user logs in to the system the status of the account is the first information available on the home page, as presented in Figure 5.11.

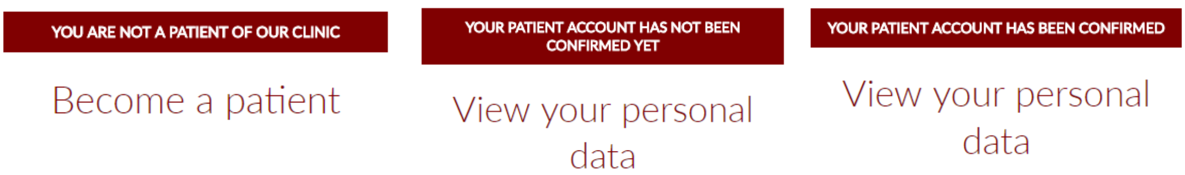

Figure 5.11: Patient statuses

Each of the statuses is a link redirecting to the page of adding new patient account, editing it, or viewing the data respectively. Figure 5.12 shows the view allowing creation of a new patient account and updating it. Fields Name, Middle Name and Surname are taken from a user account for the portal that has already been created. Other required data that potential patient has to fill in are date of birth, PESEL number, ID number, a photocopy of the ID (for the confirmation process by administrative part) and full address. Only when all data is provided an account can be created.

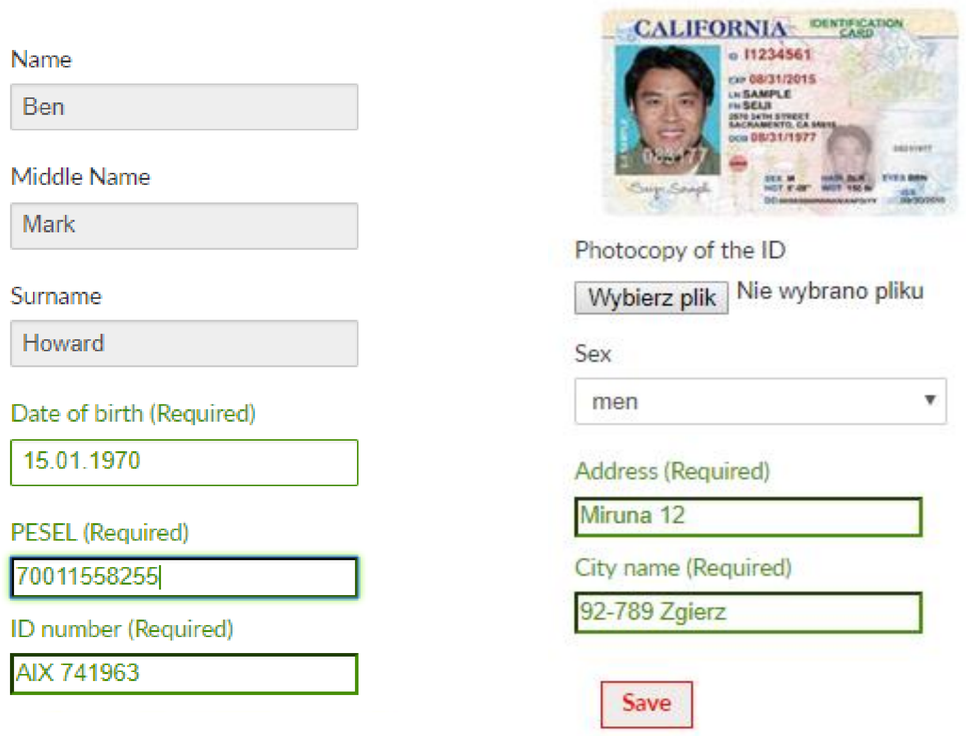

Figure 5.12: Patient account creation and editing.

In the same way view for editing patient data is constructed. When the patient account is confirmed there is an non editable list of data presented as in Figure 5.13.

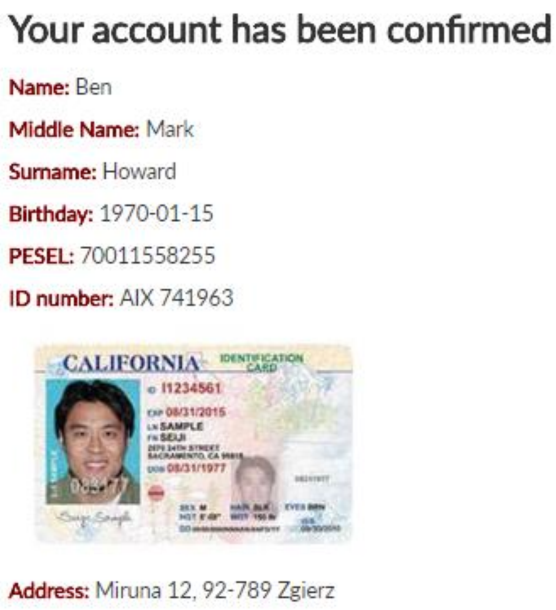

Figure 5.13: Confirmed patient data

The next use case is connected with finding information about the health clinic, which includes data of employees and their schedules. On the home page of the Web Application presented in Figure 5.14, apart from the aforementioned patient status, there is also a short description of a given clinic. Furthermore, whole layout was designed to reflect the team and EPS, hence there is the team’s logo and slogan in the header, as well as logos of the university. However, in the production environment those visuals would be exchanged with emblems of the clinic for which the Web Application would be created.

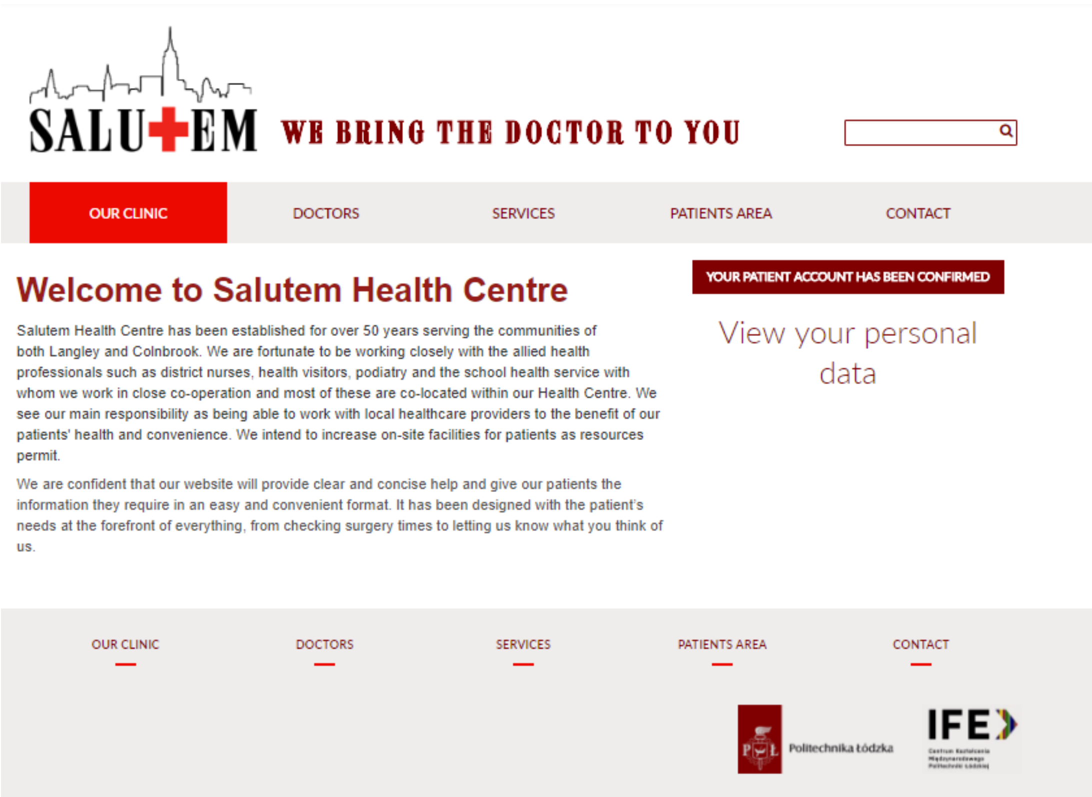

Figure 5.14: Home Page of the application

The next part of the described use case is connected with acquiring information about doctors and their schedules. When Doctors link in the menu is clicked, user is redirected to the subpage devoted to clinic employees. On this subpage every doctor is presented together with basic information such as name, specialisation and an image. Furthermore, the doctor schedule is also shown on this subpage. Figure 5.15 depicts information about one doctor.

Figure 5.15: Doctors view

When a registered patient knows schedules of the doctor there is a possibility of visit registration. This can be done on the Services subpage. Figure 5.16 depicts how view for this functionality was created.

Select date of the visit

11.06.2018

Available visits

Available doctor	Mean visit duration [min]	Visit time
BRIAN NELSON	15	08:45 - 09:55 10:10 - 12:00
DAISY SNOW	12	06:00 - 13:15
RONALD SMITH	15	10:15 - 12:55 13:10 - 14:30

Figure 5.16: Services view – visit doctor and date choosing

On this subpage user can see which doctors are available on which date. Date choosing was done using AJAX [36], which performs an asynchronous call to the backend of the Web Application, hence, list of the doctors gets automatically updated each time a visit date is changed. The list of doctors contains the name, mean visit duration for a given patient with a given doctor and possible visit time slots. Those time slots are connected with two aspects of visit registration. First of all, they are dependent on doctors schedules, as one doctor can have visits for example both in the morning and in the evening of the same day. Secondly, available time slots are defined basing on registered visits of other users. When a patient has decided on which time slot to choose, after clicking on it, there is a redirection to the registration page. This page is shown in Figure 5.17.

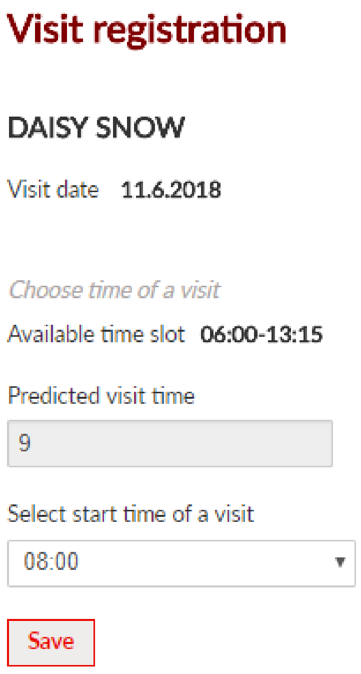

Figure 5.17: Registration page

While registering a visit, the patient has information about doctors name, visit date and available time slot. Then, patient has to choose the exact start time of a visit. Predicted visit time shows the effect of the implemented AI algorithm for visit time prediction, which will be described in more details in next chapter. After registration the patient is getting a confirming message on the screen:

Visit has been registered

In order to avoid registering more than once for the same doctor in the same day a method of checking the possibility of registration was implemented. In such situation patient would get a message:

You have already registered a visit for this day

The next two use cases, which are seeing the queue for the doctor and uploading Google Track are available on the same page as seeing confirmed patient user data. When a patient has any registered visit for a current day, on this page information about this visit will be shown. Apart from doctors name, visit time, expected visit duration and visit status there is also information about the queue. The queue is constructed in real time based on the information from the doctor's office. Each entrance and exit of a patient is saved in the database, not only for registered patients but also for emergencies. Thanks to these data system is able to present how many patients are still in the queue before the patient who sees the data, and what is the delay. Figure 5.18 shows an example of such view.

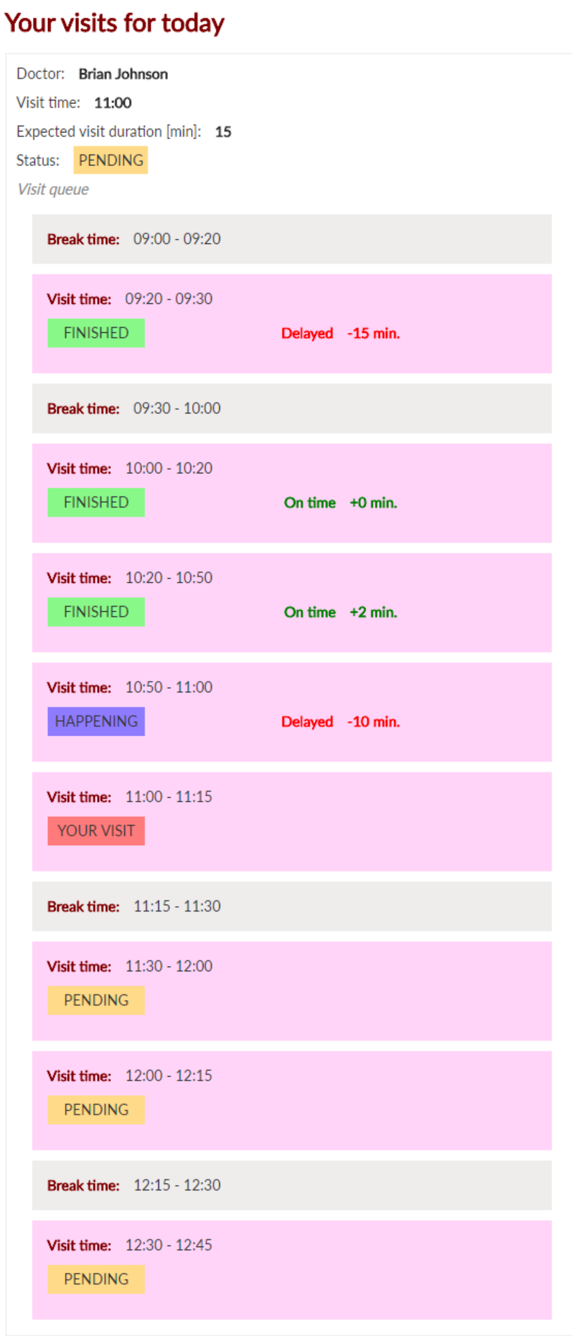

Figure 5.18: Queue of patients

As presented, something urgent happened twice in that day and there are two delays, one for 15 minutes and one for 10 minutes. Though the first delay does not affect the main patient, as after the delayed visit there was a break planned. However, from the second delay patient can deduce that his visit will be delayed as well. Not only this queue is important to show delays but also break periods, which are times in which doctor is in the office without any patient being registered for that time. With this information patient can decide to for example go to the doctor some time earlier than the planned visit.

Last use case for a patient is to upload Google Track to the server. Google Track can be acquired via Google Timeline, which is a feature of Google that allows seeing the travelled distance, visited places, pace of travelling defining walking, biking, driving etc [37]. As this information is for private use only the patient is required to upload their track manually to ensure that only people who want to share these data will share it. The reason why Google Tracking was implemented in the application is to allow monitoring of patients lifestyle habits, which can be useful for diagnosis. As the created software is just a prototype, geolocation is the only possibility of accessing patient specific real-time data, however, with use of IoT a number of different data can be gathered in real time, for example blood pressure, heart rate or temperature. All such data may help doctor to make a better diagnosis. Figure 5.19 depicts part of the view responsible for uploading Google Track on the server. Accepted format is JSON, as this is the format in which Google export its data.

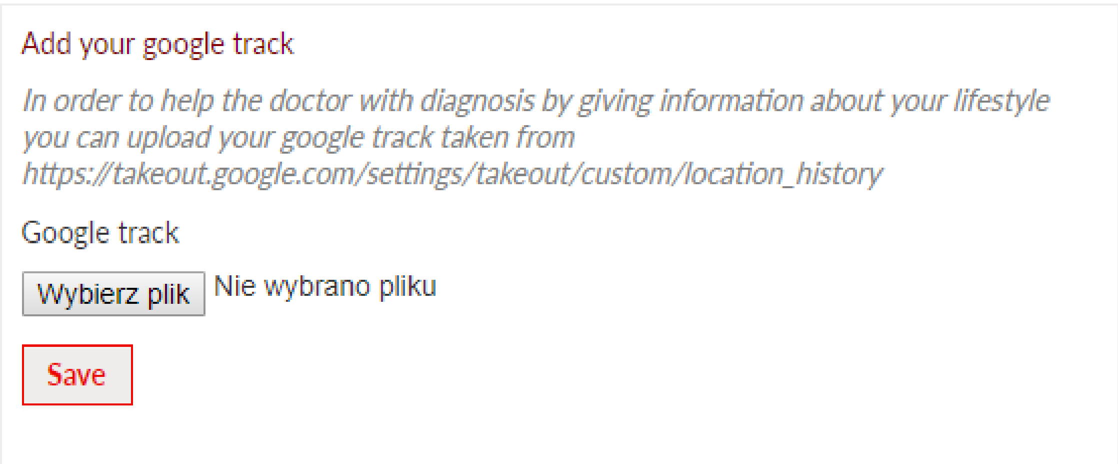

Figure 5.19: Uploading of Google Track

5.1.3. Doctor

The last actor that has been defined in the Web App is a doctor. This role has only two use cases, which are to see patient registered for a current day and to see data of a patient currently being on a visit, including seeing the Google Track. First use case is aimed for the doctor to see how many patients are registered for the current day and on which hour they are to come. However, this does not mean that the doctor see all the patient that may come, as the created system also allows some emergency visits. In such case visit is not registered, however, entrance and exit actions of the urgent patient are captured so that doctor can still see information on the patient that is currently in his or her office. Figure 5.20 depicts view with a list of planned visits for one doctor. It shows patients name, visit time, expected duration of the visit and a status – whether visit has already been conducted or still not.

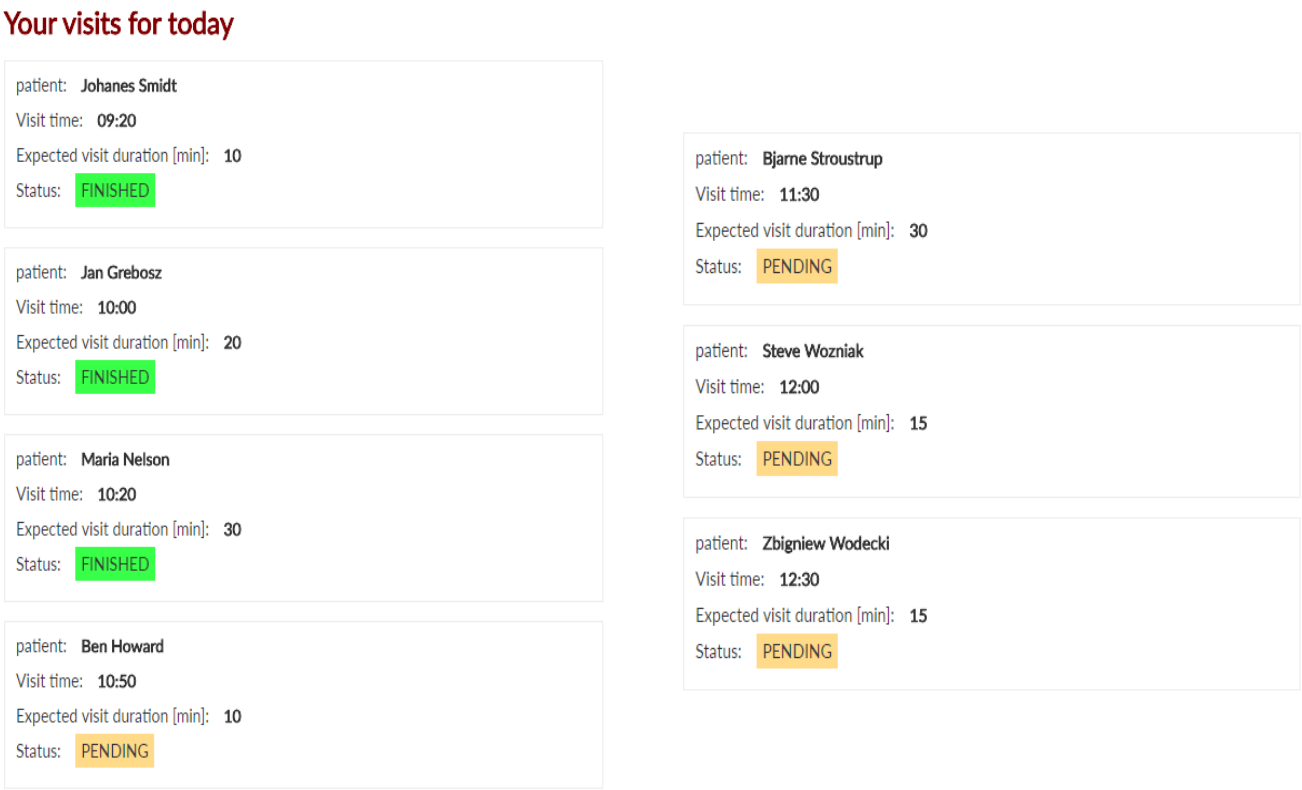

Figure 5.20: Visits for current day

Next to the list of visits there is information about the current visit. The current visit is visible in the moment when entrance of the patient in the office is detected and lasts until the exit is detected.

Current visit

SCHEDULED VISIT

Patient information

Name: Ben

Surname: Howard

Birthday: 1965-10-12

PESEL: 65101285055

ID number: AXY 74852

Address: Przybyszewskiego 15/85 95-852 Zgierz

Google track

Figure 5.21: Current visit together with Google Track

A visit can be both registered (scheduled) and urgent (not scheduled). A doctor can see information about the patient that is currently undergoing a visit. As the created software is just a prototype and not a full application only very basic information about the patient are shown in this view. In the future versions of the system, a doctor should be able to see all the medical history of a patient, including previous symptoms, diagnosis, prescribed medicine etc. Below basic information about the patient there is a map presented, which shows information from the Google Track uploaded by a patient. As shown, there are markers on the map, which suggest where the patient was spending some time.

Each of the functionalities for each actor are by default presented in English, however, the created portal supports multi-nationality of the system. All that has to be done is to append the language code to the URL, then automatically each element on the side will be translated, as shown in Figures 5.22 and 5.23.

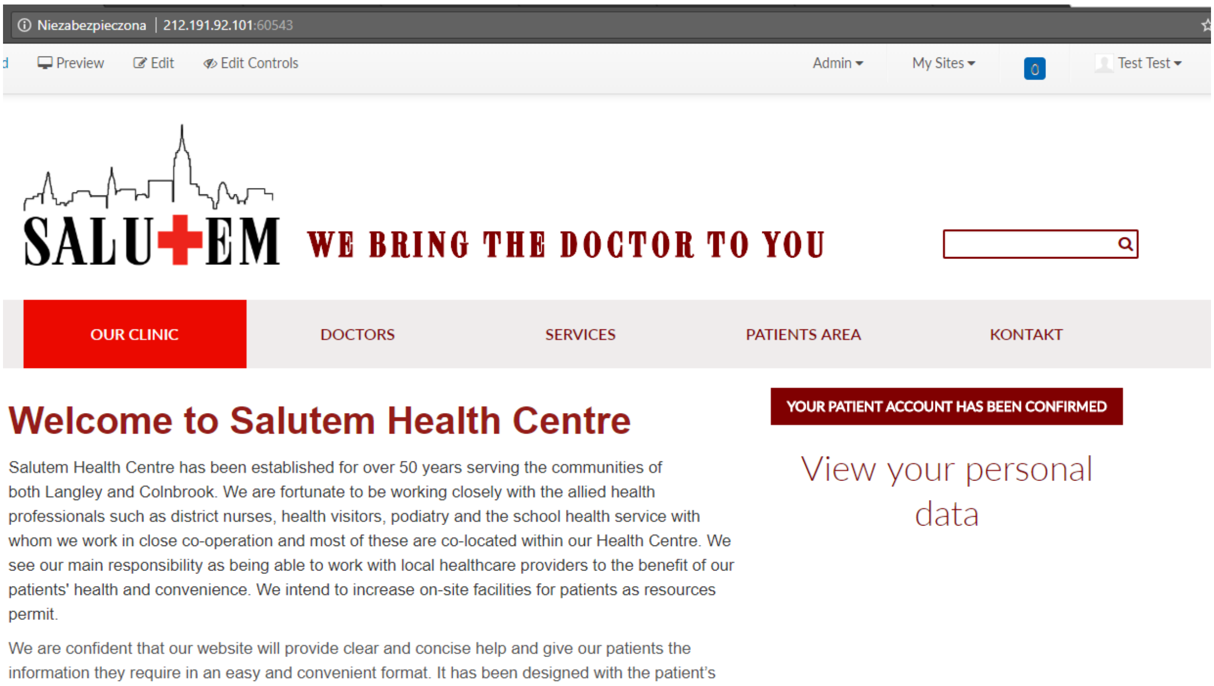

Figure 5.22: Front page in the default language - English

Figure 5.23: Front page in Polish

This chapter contained information on the functionalities of the Web Application created as a part of the team's solution for the problem. Those functionalities were described basing on the use case diagram. In the application three main actors were identified which are administrator, patient and a doctor. Each role was described separately.

5.2. Mobile Application

The second subsystem of the solution is a Mobile Application. The Mobile Application was aimed to communicate with Web Application and NFC tags. Its main role was to provide functionalities needed for having a visit. Tough this part of the system also has three actors, as the Web Application, only two actors – doctor and administrator are using the Mobile Application. The only action patient performs is connected with NFC tags. The purpose of the Mobile Application is to control entrance and exit of a given patient and to aid in presenting patient's data to a doctor in a fast and automatic way. Figure 5.24 presents two use cases of this application.

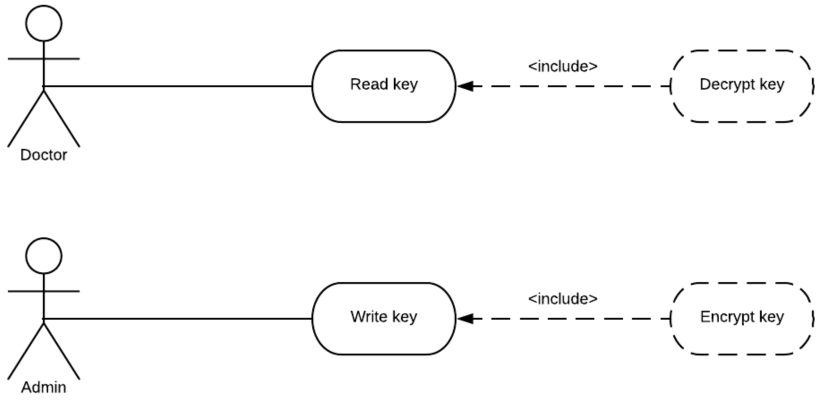

Figure 5.24: Mobile Application use cases

Following subchapters will go in details about the design of the application and they will present user manual needed to understand what are the functionalities of this part of the system.

5.2.1. Initial design of the application

Any application has to be not only functional but also attractive and simple from the user perspective. To make the Mobile Application appealing for the user, before starting the implementation part, the design of the Mobile Application was thought over and discussed in the team. As it was a result of brainstorming the first design was done on paper. This chapter will provide a brief explanation on how the views were planned to look.

The first view of the application that will be described is a login screen. Figure 5.25 depicts the initial idea of its design.

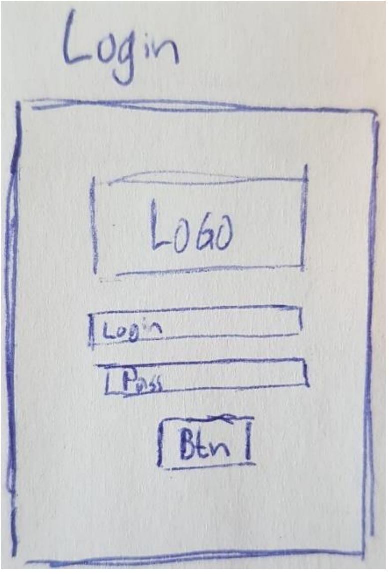

Figure 5.25: Design of the login screen

The presented login screen is planned to have an image on the top with the logo of the Salutem team. It was decided in that way as multimedia aspects of any application is what makes it attractive for a user. Also the decision of placing it on the top of the view was made according to the recommendations of experts in a design field. Users expect to have some elements of the layout in the same places in each app. Placing a logo on the top is one of such recommendations. Below the logo there is a place for two text fields required for the login. The first one is to provide a username, and the second one for password input. To make it safe, the latter input should be constructed in a way allowing hiding the characters for example under an asterisk sign. The last element of this view would be a button used to login and go to next views.

The second designed view is a main page of the doctors' part of the application. Figure 5.26 depicts the initial design of it.

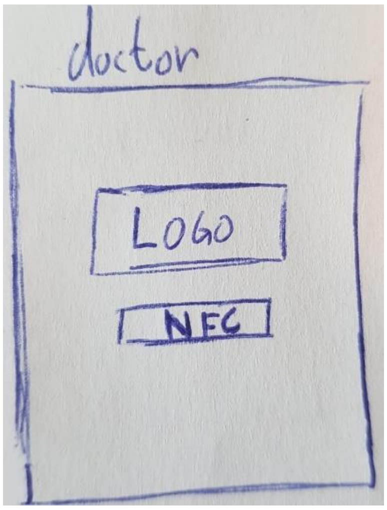

Figure 5.26: Design of the doctor's view

As in the previous screen, there is a logo of the company presented on the top. This time there are no inputs but only a button. This button should invoke an action of reading from the NFC tag that is going to be placed in the close proximity of a smartphone.

When the tag is detected there should be a message available for the user to confirm that. Figure 5.27 presents how such message should be designed.

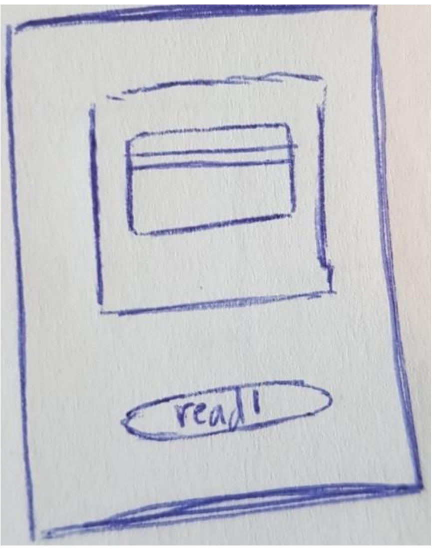

Figure 5.27: Design of the screen with message confirmation

After the detection of an NFC tag, in the initial design, there was an idea to place on the screen an icon with a card, and a popup message with a confirmation. The icon of a card was chosen as the NFC tag was to be placed in the card for the sake of better visual perception by users. It is also more convenient to hold a card than just a tag, and certainly it is less probable to lose it. Once the operation of reading from an NFC tag is finished the application should automatically come back to the main screen. Then, it should be in an idle state waiting for a new operation to appear.

The next highly important view that had to be designed is connected with a second actor of the system, namely the administrator. The login page is the same, regardless of the role of the user, however, a screen to which the user is redirected after log in should be different. Figure 5.28 depicts such a screen.

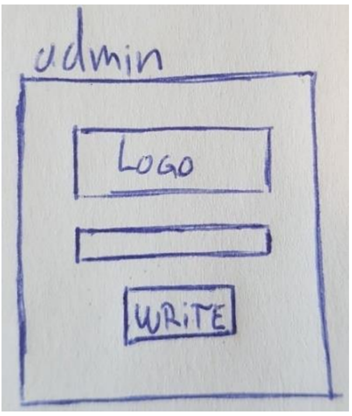

Figure 5.28: Design of the main view of an administrator

The view presented above should appear only after a successful login to an account with administration role. Again, at the very top there is the Salutem logo, which acts like a constant element visually connecting each view. Below the logo there should be a text field that will be used to input a message that is then used to write on the NFC tag. Like previously, below there will be a button placed that submits the data. After this button is clicked an action of writing data to an NFC tag should be triggered. When this action is detected a new screen should appear, as presented in Figure 5.29.

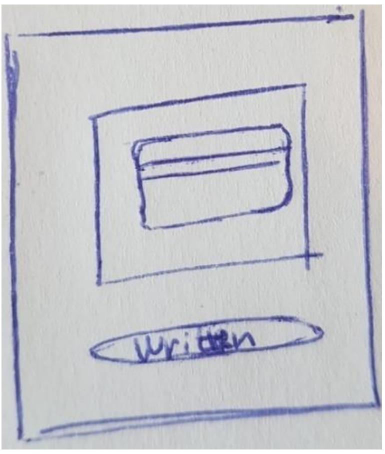

Figure 5.29: Design of the administrator confirmation screen

Similarly as in case of the doctor's, confirmation screen of the administrative part should appear after the action of NFC connection is detected. It contains a new icon, which again as an image of the card and a popup message confirming correct workflow of actions connected to writing to NFC tag. Also, after completion of this action user should get redirected to the main screen.

In such a way the system was designed, and according to presented sketches each of the functionalities was modelled and implemented.

5.2.2. Final design of the application

This chapter will contain figures depicting the final views of the Mobile Application together with a user manual on how the application is working. Colours chosen for the application are consistent with colours used in the Web Application and in the team emblems. To make it visually attractive every colour was chosen according to Material design palettes, which contain guidelines on which colours should be matched together.

Before starting of the design, the team made a research on some solutions that are already implemented in different countries of the world. The meeting with students from Denmark brought especially useful information, as Danish healthcare system is far more advanced than Polish.

After the research, Graphical User Interface (GUI) was designed, in accordance to guidelines of experts in the field of marketing. In this aspect an specialist in design and development of mobile applications David Gimenez Masvidal has shared his knowledge with the team. This expert is a professional engineer currently working in Telefonica Spain, who gave the team very important information about the first design of the system and GUI. The application was created in the minimalistic fashion, as the simpler is the design the easier it is to use for ordinary users and this application is to be used by people who do not need to have a deep knowledge of IT.

The first screen that was created was a login screen. This view is common for all the roles. Figure 5.30 depicts how it looks.

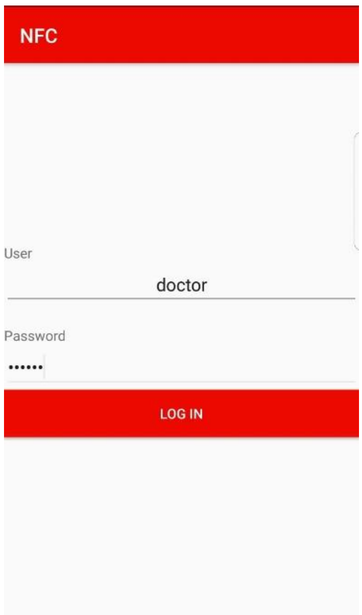

Figure 5.30: Login screen

As it has already been mentioned in a section devoted to initial design, this view contains logo and two input fields. One with plain text, which is used for username, and the second one for password, which is shown in a hidden way. This is a necessary safety precaution needed to ensure minimum privacy of the user. Once credentials are supplied and a login button is pressed the Mobile Application connects to the Web Server via the already described API, which then checks the credentials in the database. If the data is correct, the API call returns user data, which include a role, or roles. If not, then the API returns an authorisation error and user is not logged in. Depending on the user’s role, after login action, user gets redirected to a different page.

If the user has a doctor role, then the next screen is a main screen of the doctor’s view, as in Figure 5.31.

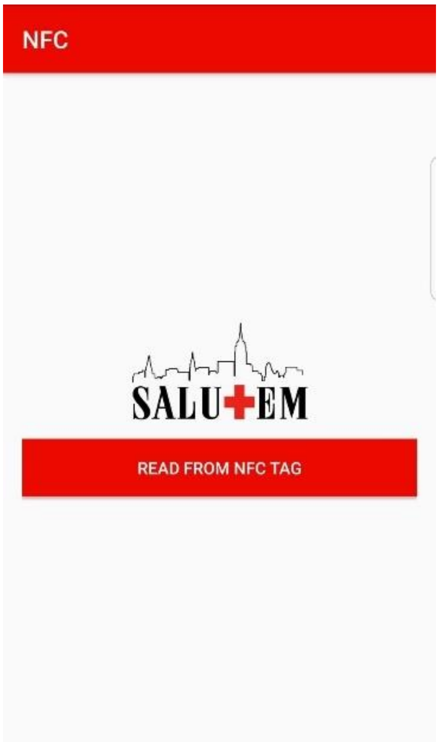

Figure 5.31: Doctors main view

As planned, the design of this screen is simple, according to the current fashion of minimalism. The main logo is presented in the middle of the screen and just below it there is a button with the functionality of reading from an NFC tag. When the button is pressed, the application performs an action of establishing the connection with a tag via NFC. Until this connection is fulfilled there is a second screen presented for a user as shown in Figure 5.32.

Figure 5.32: Waiting screen for NFC reading

A waiting view was designed in a way to be visually attractive and informative. Blurred screen and a popup clearly suggest that there is some action in the background. Also there is a caption and an image that together form a solid explanation for a user on what is currently happening.

When a tag is detected then a doctor gets redirected to the next page, which is presented in Figure 5.33.

Figure 5.33: Reading from NFC tag – success message

Once an NFC tag is in the close proximity of the reading device, which is a smartphone, the Mobile Application detects it. Again, a popup message appears with a confirmation that the process was successful. Apart from this message, there is also an information presented about what is written on the tag. This information is an encrypted patient key, which is then decrypted in the Mobile Application and used to connect to the API in other to obtain patients data, or to mark visit start or ending, which is then used in the Web Application for example for AI algorithm. When the user touches the screen, the application is redirecting to the main page.

Second actor of the system, with an administrative role has its own screen which appears after successful login. Figure 5.34 shows this view.

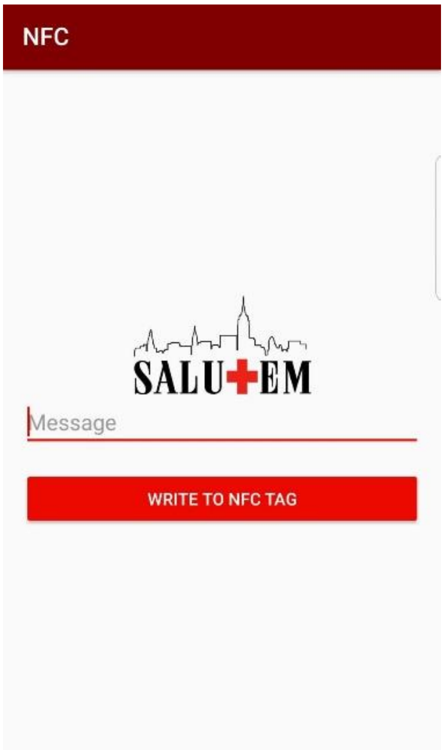

Figure 5.34: Administration main view

As it was planned, this screen is designed in a similar manner to the doctors view, with the only difference being an input field between the logo and a button. This input field is used to provide PESEL number of a patient whose ID is to be written on a tag. From the API there is a function that can be called to obtain all patients signed for the clinic together with their PESEL number. Hence, once a patient comes to a clinic it is enough to check if his or her PESEL number appears on the list of all PESEL numbers and write it to the NFC tag. Once the input field is filled with data and the button is pressed, the Mobile Application connects to the API in order to retrieve the key that is associated with a patient account with given PESEL number. Every functionality of the application that will require some processing of patient data will use this key instead of real data, for security reasons, which will be described in the next chapter.

Once a button is clicked user is redirected to the next screen, which is a loading screen as presented in Figure 5.35.

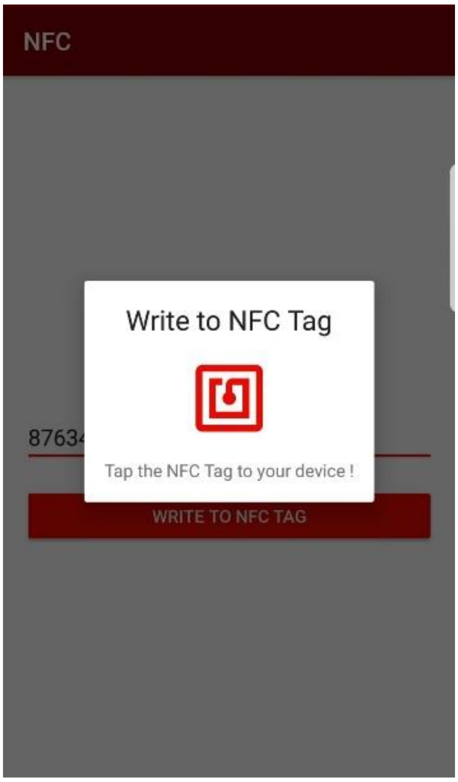

Figure 5.35: Writing to NFC loading screen

The principle of work for this screen is exactly the same as for the loading screen for reading from NFC tags. When a tag is detected, once again a new screen is shown, as in Figure 5.36.

Figure 5.36: Writing to the NFC tag

When the NFC tag is detected there is an action of message encryption called. Once the key is encrypted, the application connects with an NFC tag and writes an encrypted key to it. Application is redirected to the main page when a user touches the screen.

6. Solution description

This chapter will present in details the system architecture, structure and implementation details of the proposed solution. Firstly, a general overview will be presented and in the next chapters each component of the software will be described.

The effect of the team's work throughout the semester is an information system for one health centre, though it was implemented in a way that it can be easily scaled to more clinics. As it has already been mentioned the system is composed of two main parts, which are Web Application and Mobile Application. Those parts communicate with each other via created API (Application Programming Interface) [38]. It was constructed in a way allowing communication between components via Web Services. The API is communicating with the Database via the Web Application. Apart from the connection with the Database and the API, the Web Application uses also an Artificial Intelligence Algorithm, further abbreviated as AI. On the other hand, the Mobile Application uses a connection with NFC tags. For better visualisation of the system architecture the component diagram is presented in Figure 6.1.

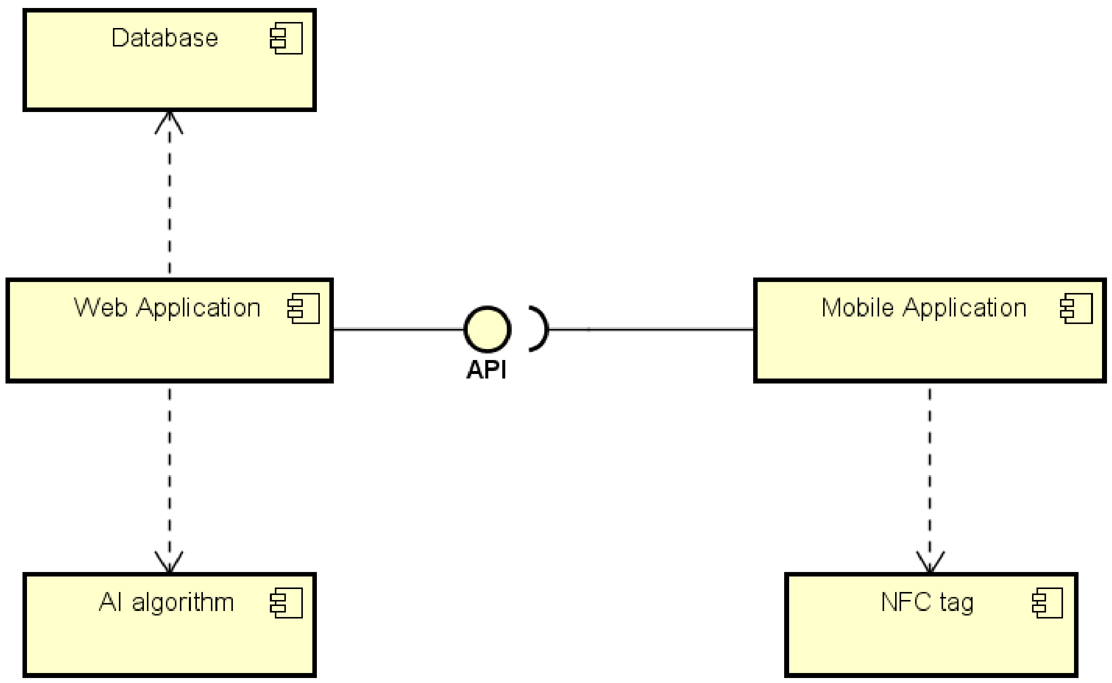

Figure 6.1: Component diagram of the solution.

Each of the system components will be described in following chapters. What is important to be noted is that the system developed is a distributed system. This means that each component, with an exception of AI, which is embedded in the Web Application, can be placed in a totally different location, in different networks. The database, the Web Application, the Mobile

Application and NFC tags are totally different components which are connected via communication channels independent on the location. For the database, the Web and the Mobile Applications to communicate the only requirement is Internet connection. On the other hand NFC tags have their own communication channel which requires only close proximity of sending and receiving devices.

6.1. Web Application

As one of the two main components of the system, implementation details of the Web Application need to be described. This chapter will contain information on which technologies were used for the solution implementation. Next, each of the subcomponents of the Web Application will be presented together with important technical details about them.

6.1.1. Used technology

Web Application being a part of the solution is a Content Management System (CMS) [39] created on Liferay [40], which is an open-source platform written in Java that allows building websites and portals in a customizable way. Thanks to Liferay Software Development Kit a default portal can be changed and enriched with extra plugins that can tailor the portal for future users' needs. For the project Liferay in version 6.2.5 GA6 Community Edition was used. Portal was deployed on Tomcat server in version 7.

Liferay Portal is a platform offering great possibilities to developers as it allows creation of responsive web applications and web services in an easy and clear way. There are five ways of modifying Liferay Portal named themes, portlets, hooks, layouttpls and extensions and in the project the first three were used. Extensions are aimed only for advanced modifications of the source code of Liferay Portal and should be avoided if possible. Layouttpls are a method to modify a basic template of the page, for example to add elements which are repeatable on some pages or to divide a page in columns. As default layout templates were satisfactory for the created portal no extra Layouttpl plugins were developed. Next plugin type, which are Hooks are aimed to modify Liferay default actions, properties or JavaServer Pages (JSPs) used to create views of default Liferay plugins. In the project there were two hook plugins created, one to override default language packages in order to provide multilanguage properties of the system, which has already been presented in previous chapter, and a second one to make some files available at the portal (global) level, for example library JQuery, which was used in the whole CMS. Themes are made to change the visual aspects of the portal, which are styles, scripts, templates defining page elements like header, banners, navigation etc. The language chosen for processing instructions referencing objects defined in backend (Java code) is Velocity. With the use of this language to create portal templates, extra CSS stylesheets and JS code portal can be adjusted to any need users of the portal can have. Last, but most important type of plugins are portlets. These are separate applications, which process requests and generate responses returning content in HTML for display in a browser. Portlets can also create custom database entities and provide local and external services used then as part of API. Portlets are created with MVC design pattern [41] which is a highly popular and recommended way of software creation that allows separation of logic and views. In the created CMS there were three portlets developed. First one is to manage patients, second to manage doctors and the last one for visit management. All the logic behind use cases described in the previous chapter was implemented in those three portlets. With all of the presented plugin types whole application which is described in this report was created.

In order to make communication between components possible, Web Application had to be deployed on a publicly available server. Thanks to the Institute of Informatics of the Faculty of Technical Physics, Information Technology and Applied Mathematics the server was made available for the portal to be deployed. It is available under address 212.191.92.101 on port 60543. The server was given to the team with a fresh installation of Linux of Ubuntu distribution, hence, before deployment of the application all necessary software had to be installed and configured.

6.1.2. Database

Database used for the project was created on MySQL server and bound to the Web Application via MySQL JDBC connector. Because of the fact that for the creation of the application Liferay platform was chosen there is a number of tables related directly to the portal itself. In this report only the necessary tables will be described, which are mainly custom entities created for the sake of the project. For best visualisation of the database and ER diagram was constructed and it is presented in Figure 6.2.

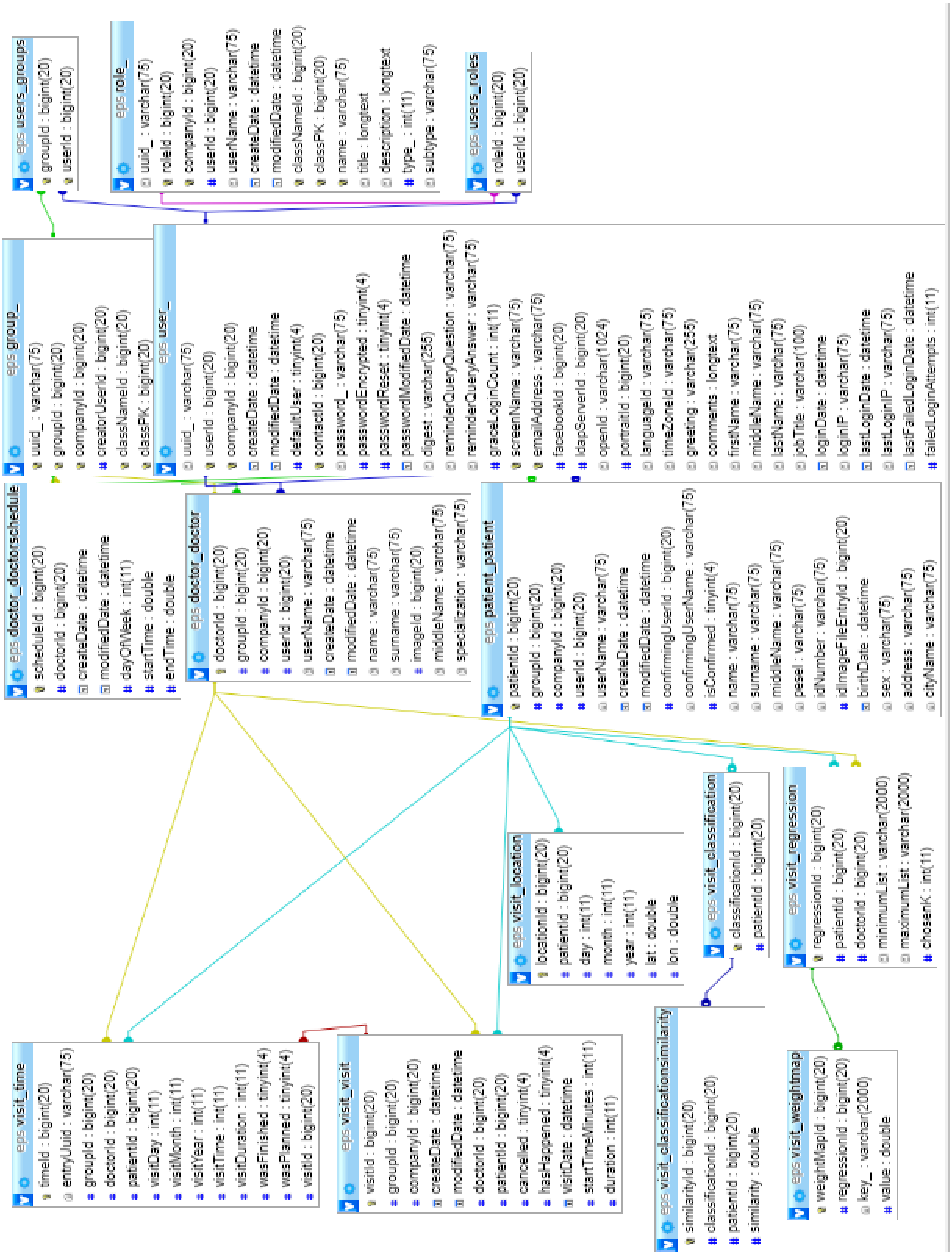

Figure 6.2: ER diagram (essential part).

As shown in the figure above there are 15 tables necessary for understanding of the project. Three tables: User_, Group_, Role_ are tables generated by Liferay. The first two were cut in the diagram for sake of better readability. Table User_ represents user accounts for the portal. It is connected to table Role_ with relation many to many. In the system there are three roles: patient, doctor, and administrator. Group_ is the table which stores information about specific health centre. Though in the prototype of the solution there is only one website for one clinic, the whole solution was implemented in a way allowing scaling to more clinics. Hence, Group_ table is connected to tables Doctor_Doctor and Patient_Patient, as those entities should be bound to one specific clinic. Those two tables are also related to table User_, as the portal allows twostep registration for patient and doctors, as it has been already presented in the previous chapter. Furthermore, table Doctor_Doctor is related to table Doctor_DoctorSchedule, which stores schedules of each doctor for each day. Then, there is a set of tables connected to visit management. Table Visit_Visit being related to Doctor_Doctor and Patient_Patient store information about registered visits. Table Visit_Time stores information about visits that happened, which are defined by entrance and exit time registration. It is also related to table Visit_Visit, though not every entry from Visit_Time satisfies this relation, as not every visit is scheduled. Tables Visit_Regression, Visit_WeightMap, Visit_Classification and Visit_ClassificationSimilarity are used to store data needed for AI algorithm of visit time prediction. Furthermore, each POI (Point of Interest) [42] of Google Track for patients is stored in a table Visit_Location.

These tables are used to store all the necessary data for the created CMS.

6.1.3. API

To enable communication between the Web and the Mobile Applications there was a need to establish a communication channel. This channel was created with use of Liferay JSON Web Service API. This API can be enriched with extra methods via Liferay portlets and can be invoked in many languages, not only Java but also Python, Ruby, JS or Groovy. Furthermore, it can be invoked via URL or Curl call [43].

In the portal there are five functionalities which required connection of the Mobile and the Web Applications via the API. First of all, to provide authentication for the Mobile Application there was a need to implement a method in the API to check whether the given credentials on the smart phone reflect a user account that was created in the CMS. Next functionality was connected with writing patient identity key on a NFC tag by administration so that it can then be used to access information about the patient in the doctor’s office. To achieve that, two API methods were needed: one to list all patients, so that, if necessary, administration can have user data of every patient in the given clinic, and the second method is to get patient's identity key basing on the PESEL number. With patient's key being saved on the NFC tag there is a possibility of final functionality of the API which is registration of entrance and exits of a patient. To present the general idea of how the API works, an example action of calling the last two methods will be described. Liferay provides user friendly interface for calling web services, though in the project service were called automatically from the mobile application. Figure 6.3 presents how Liferay JSON Web Service API looks.

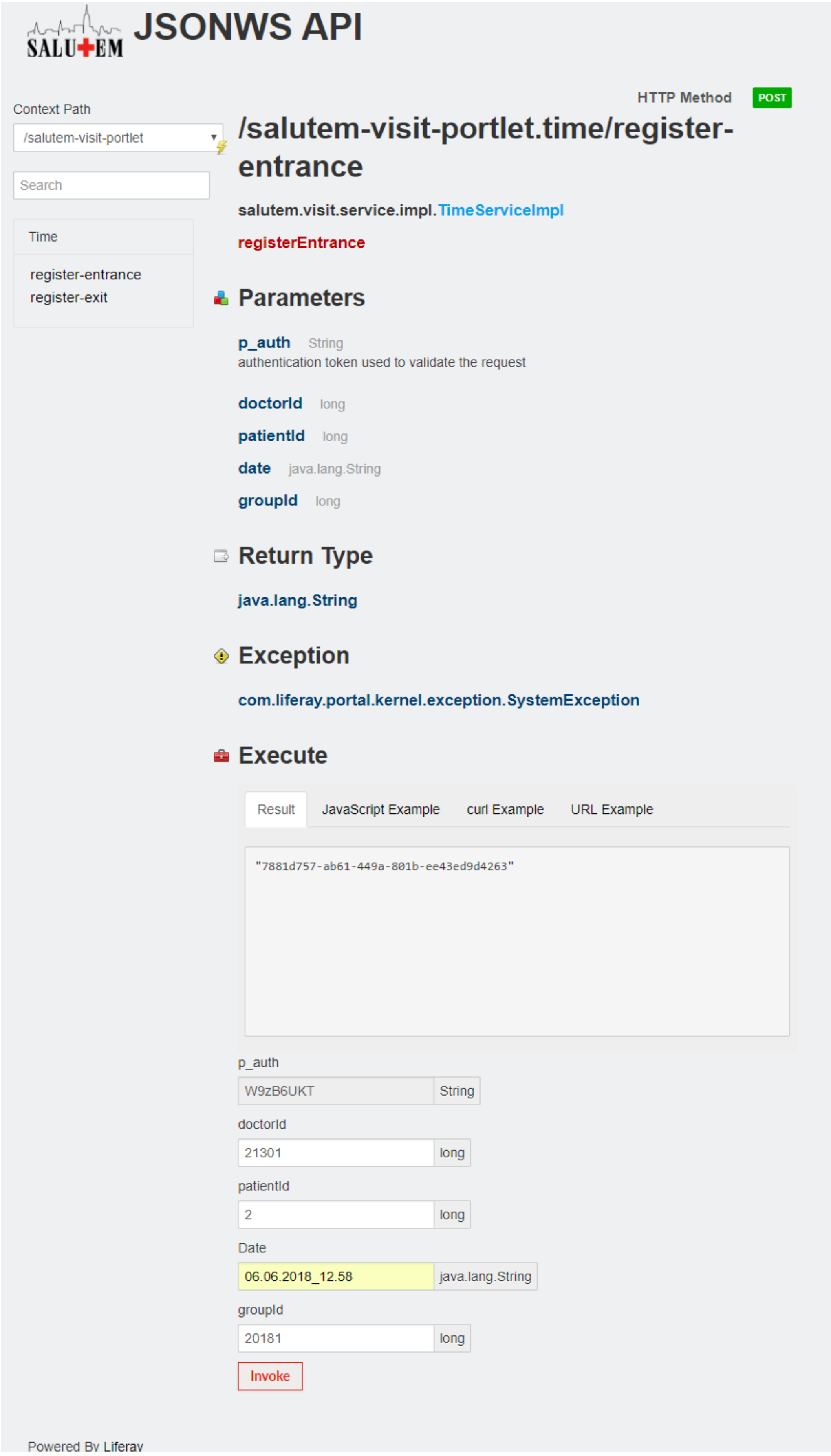

Figure 6.3: API call – entrance registration

On the left-hand side on the presented figure a method from one of the available context paths can be chosen. A context path gives information from which portlet a given method is coming. At the top there is a method name presented, then parameters of the call, return values and exception types are described. Under execution tab there are fields to be filled with call data. Hence, for example to register a visit of a patient with ID 2 to a doctor with ID 21301 for a visit in clinic with ID 20181 on 6th of June 2018 at 12:58 one has to put all these information in appropriate input fields. After an invoke button is clicked a Java method being a controller of this service is invoked. This method registers an entry of type Visit_Time and if the visit was planned makes a relation with appropriate Visit_Visit object. Then, a new entity is saved to the database. Returned value, presented in the execution box, is UUID, being a unique, generated identifier of this particular entry, which is then to be used to mark exit of a patient. Figure 6.4+ shows how calling of the method register-exit looks. In this method only two parameters are needed – UUID that has been generated from the previous call and the date of visit end. Once invoked the referenced Visit_Time entry is found and marked as finished, and the visit duration is calculated.

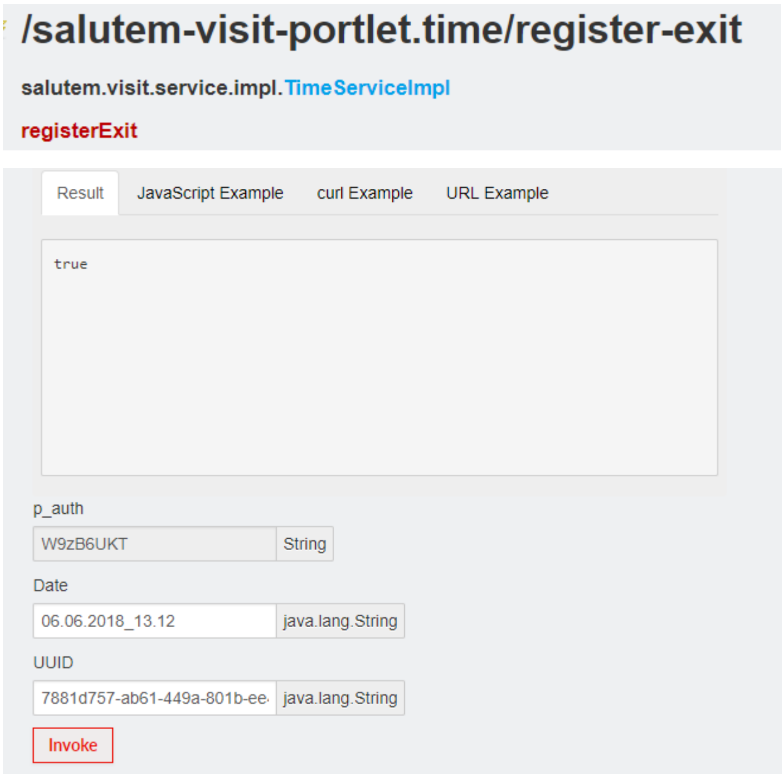

Figure 6.4: API – exit registration

In a similar way other API methods are called.

6.1.4. AI algorithm

A highly important part of the created system is an AI algorithm that is used to predict time spend by a particular user on a particular date for a visit to particular doctor. As the team's goal is to optimize time management of a health centre the algorithm is aimed to reduce waiting time for registered patients and to fit an optimum number of visits in one day.

Artificial Intelligence is a very popular and fairly new concept [44]. It is a broad term defining a branch of software engineering devoted to creation of algorithms that are able to make reasoning by mimicking human behaviour. One of the methods of creating intelligent algorithms is machine learning [45]. This is a way of constructing algorithms that are able to act in a desired way without being explicitly programmed. In simplified words, for the algorithm to work it has to be given a large number of training examples, with known inputs and outputs. Then the algorithm learns from those known examples. When it has learned all the known data, it should be able to predict the output for not known cases.

In the project created for EPS an algorithm should learn to predict visit time basing on previously gathered time entries in Visit_Time database entity. For each patient the system knows exactly how much time was spend with each doctor on certain dates. The developed algorithm is composed of two parts: regression algorithm and kNN classification algorithm. This chapter will be subdivided into three sections. The first two will be devoted to description of both algorithms, and the last one to presentation of the method of inferring the final result from sub-results of those two algorithms.

6.1.4.1. Regression

Linear regression [46] is a tool widely used for predictive analysis. In most general words linear regression is to construct such a function that will estimate the whole input space, which is composed of both training samples and future unknown inputs. Every function with one output can be defined as a polynomial of finite order and degree. The goal of regression is to learn parameters being coefficients in this polynomial formula. To better visualise the concept a very simple example will be shown. Table below presents two training samples.

Table 2: Example training data for regression

Input	Output
5	14
3	10

The idea is to give those two training examples to the algorithm and let it construct a polynomial that will approximate the data. Knowing that it is linear regression, hence a polynomial of degree one, and having only one input variable, hence, a polynomial of order 1, a function is simple:

$$y = ax + b$$

y is the expected output, x is an input, and a and b are the parameters describing the polynomial. Having two unknown parameters and two training inputs the resulting function can be easily calculated as:

$$y = 2x + 4$$

Knowing that formula any other test sample can be approximated by it and a correct result would be returned.

Though this is a very simple example it shows the idea of the goal of regression. However, in real life data is not linearly dependent. Moreover, from raw data human is not able to judge what is the optimal function estimating the data. However, computers are able to do that by learning and reasoning from a large number of training examples.

Regression algorithm implemented for this project use Gradient Descent Method [47]. It is one of the most popular methods of optimisation used in machine learning. The algorithm works by constructing a polynomial of degree 1 and with order being equal to the number of inputs, and firstly initialising it with random values. Then, iteratively it corrects parameters of the polynomial in order to obtain the final, good and optimised solution. This correction of parameters is done by use of Gradient Descent method. Gradient is a measure of how the calculated output of the function is different from the expected output of training data. Hence, the algorithm find such parameters that minimise the gradient. When gradient is stabilised and on an acceptably low level the result function is established, which contains all parameter values, hence, the final function would look like below:

$$y = p_n x_n + p_{n-1} x_{n-1} + \dots + p_2 x_2 + p_1 x_1 + p_0$$

With all p from p_0 to p_n with n being equal to the number of inputs.

Having a large number of training samples an algorithm can be taught to make good predictions about unknown cases. There is one main problem that had to be solved and it is that not every data can be estimated with use of polynomial of degree 1. Fortunately, each polynomial of higher order can be transformed with use of simple mathematical operations into polynomial of order 1 just by substituting each input of higher order by some new, introduced variable. For example equation of a polynomial of the degree 2 as presented below

$$y = 2x^2 + 4x + 3$$

Can be also presented as

$$y = 2z_1 + 4z_2 + 3$$

Where x^2 is substituted as z_1 and x as z_2. In such a way a polynomial of degree 1 is obtained and can be learned by linear regression. Having the substitutions remembered, after calculations one can go back from substitution to original input space.

Hence, knowing the expected order of the polynomial the algorithm can be taught to approximate unknow test cases. Finding an optimal value of polynomial order is done be separately training the algorithm for each order from some range and checking which predictions give the best results.

With use of such reasoning, the algorithm for this project was created. It takes into account three inputs, which are day of month, month and time of a visit. Those inputs were chosen as an example to show that regression can be applied for such prediction tasks. Created algorithm takes data from Visit_Time table and learns parameters defining a polynomial for each pair patient-doctor. With parameters learned, it can predict the visit time for a new visit. However, learning process can be computationally costly and time-consuming, hence, it would be unwise to train the algorithm each time a patient would like to register a visit. Instead, there table Visit_Regression and Visit_WeightMap were constructed which store data about calculated parameters of the polynomial for each pair doctor-patient. Every week, on Sunday a scheduler created in the WebApplication runs all the calculations required for training and saves them in the database. Because of tha,t from the user perspective visit time prediction result is obtained almost immediately. To visualise how it works Figure 6.5 presents a diagram of regression algorithm and Figure 6.6 a diagram of obtaining estimated visit time using calculated regression parameters.

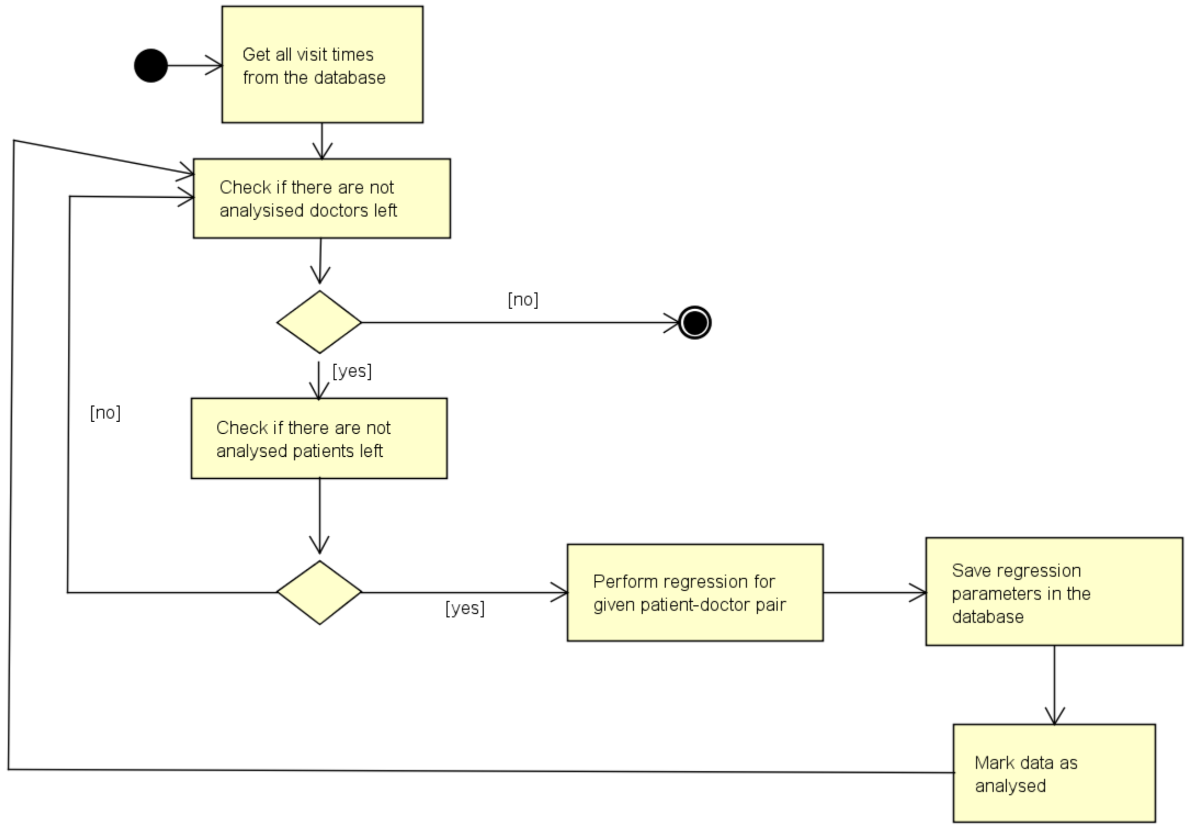

Figure 6.5: Flowchart of performing regression

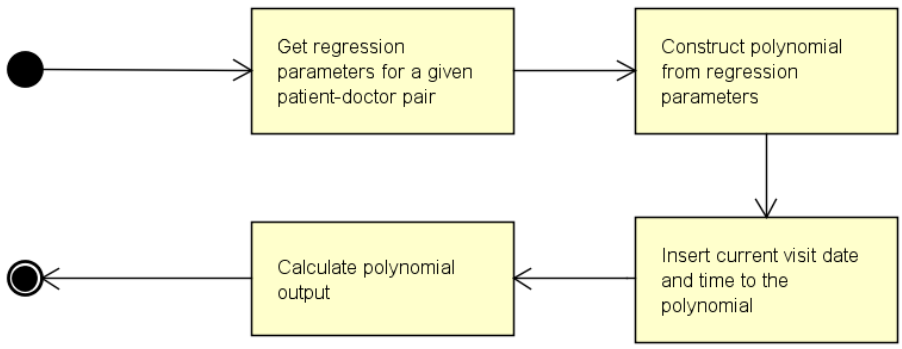

Figure 6.6: Visit time estimation based on regression

To show that the algorithm is working correctly there was a test conducted. Created profile of a patient is a men in middle twenties with high, but controlled pressure. This person is very talkative, and furthermore knows the doctor in person, hence, he likes to spend more time in the doctor's office then needed. However, this patient has a really strict boss, who does not tolerate being late. As a result there is a correlation between time of a visit and its duration. If a visit is in the morning then time should be shorter, as a patient needs to quickly go to work in order not to be late. Though in the afternoon, after work, this patient can spend more time in the doctor's office. Figure 6.7 depicts results of this test. The same patient tried to register for the same doctor and for the same date, however, for different times – one in the morning and one in the afternoon. As visible the expected time is different, 9 and 16 minutes respectively. Hence, regression was successful.

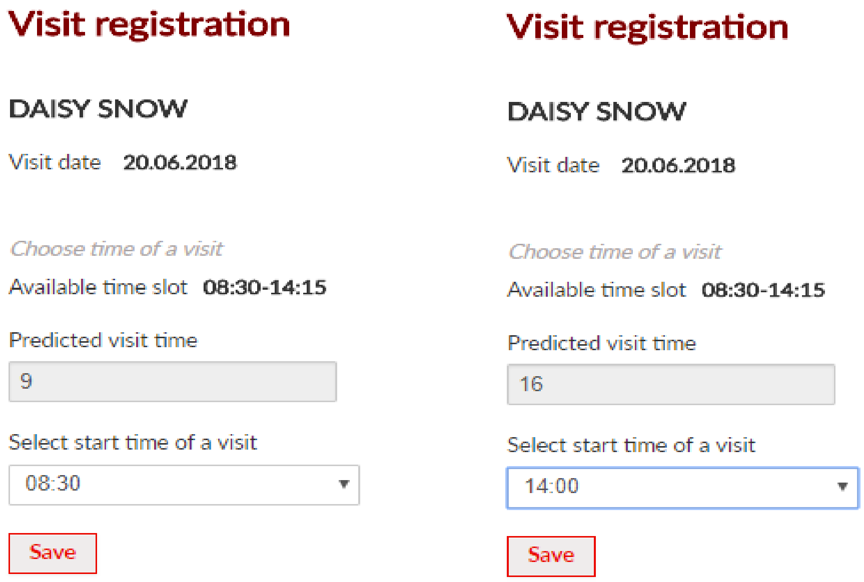

Figure 6.7: Result of regression

6.1.4.2. kNN Classification

The second algorithm that was taken into account while predicting visit time of patient is a method name k-nearest neighbours algorithm (kNN). This algorithm, instead of learning from the training data to predict the result tries to classify an object basing on k most similar other objects. Once again, to explain the basics of the algorithm a simple example will be proposed. Figure 6.8 presents input space with training samples and unknown test inputs marked.

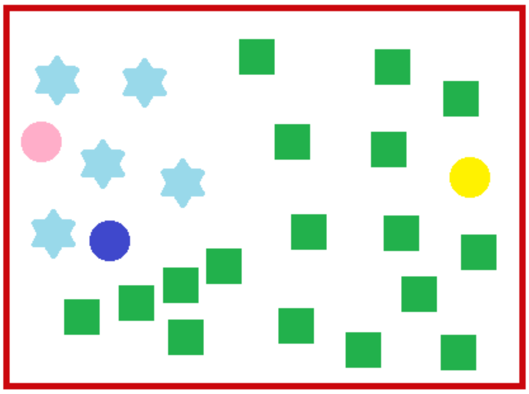

Figure 6.8: Example of kNN classification – input space

Circles represent three data points which are to be classified. The task of this example is to find classes of those circles. There only two possible ways of classification – either an object is a star or it is a square. kNN works by choosing a certain number of objects which are closest to the input sample and checking which class is the most popular in the neighbourhood. Then, this class is assigned to the unknown object. K is a constant number and has to be adjusted to the needs of a given problem. To show how important is right selection of this constant classification will be presented for different values of this constant namely : 1, 4 and 11. Figure 6.9 presents visualisation of neighbourhoods of each unclassified point.

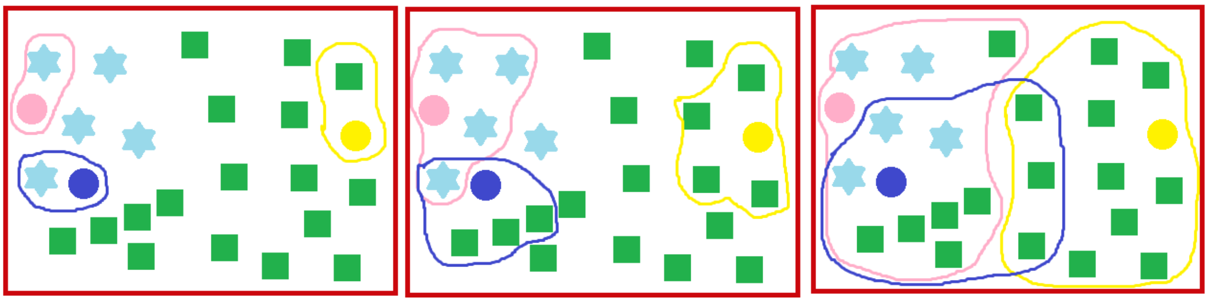

Figure 6.9: Example of kNN classification – marked neighbourhoods for k = 1, 4 and 11 respectively

The first picture, on the left, presents classification for one neighbour only, thus, classification is clear. Pink and blue circles would be classified as stars and yellow circle as a square. In case of k equal to 4 (middle picture) classification is different. As visible situation is the same for yellow circle – all neighbours are squares, hence, yellow circle would be a square too, and for pink circle, which would be a star. However, for blue circle three neighbours are squares and only one is a star. Hence, according to rules of kNN, as majority of neighbours share class square then blue circle should be a square too, though in previous classification it was marked as a star. Last example presented in the rightmost picture for k being 11 change the classification completely. In this example all three circles are classified as squares, and this is because k is more than twice larger than the number of objects with the least populous class. There are only 5 stars, hence if 11 objects are taken into account there is no such a situation in which stars would be in a majority.

Hence, knowing the value of k constant kNN algorithm can be used to predict the class of an unknown object. As in case of regression, finding an optimal magnitude of k requires performing evaluation on known samples for different values and choosing one giving lowest error.

Using kNN the second part of AI developed for this project was implemented. The expected result is a visit time of patient based on mean visit times of most similar patients. There were three factors taken into account while finding neighbours, which are age, sex and number of total visits. Last parameter was chosen as for example somebody may go to doctor only when something bad really happens, and then the visit will be longer and other person can go regularly only for prescription, thus much shorter visits. Similarity measure is calculated with use of mean square error from all of those three factors between a given patient and all others. Using this measure most similar patients to a given patients are found and saved, together with similarity level, in the database in tables Visit_Classification and Visit_ClassificationSimilarity. Again, every week, on Sunday a scheduler in the WebApplication runs that finds the most similar neighbours for each patients and saves it in a database.

Then, when a patient goes to the doctor calculations based of nearest neighbours are conducted. As the expected result is not a class but time, which is a continuous number, the algorithm presented in the example had to be changed slightly. Instead of taking the majority vote of neighbours there is a possibility to calculate the result with use of weighted average. Then the result is calculated according to formula

$$y = \frac{\sum_{i=0}^{N} s_i * \tilde{y_i}}{\sum_{i=0}^{N} s_i}$$

where y is the output, s represents similarity factor, $\tilde{y_i}$ is a mean time of individual patient doctor pair, and N is the total number of neighbours.

To visualise how estimated visit time from kNN is calculated, flowcharts on Figures 6.10 and 6.11 are presented.

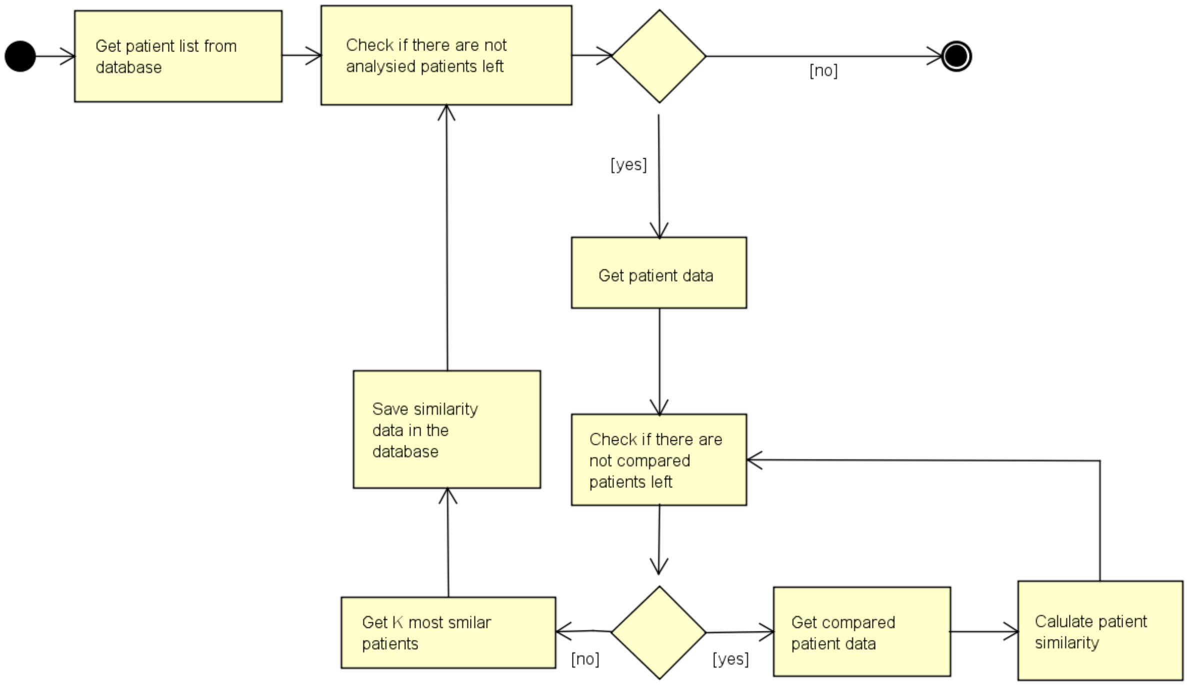

Figure 6.10: Flowchart of finding most k similar patients

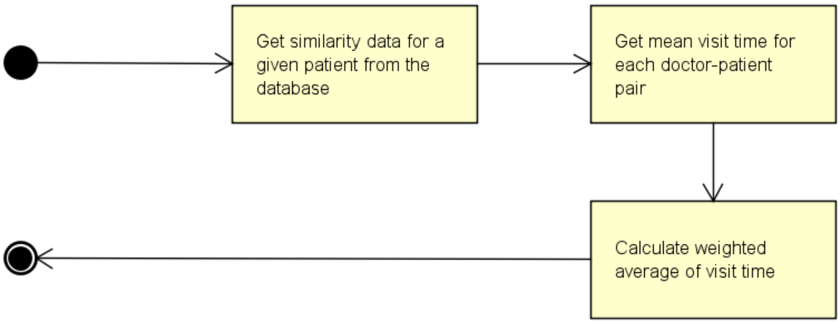

Figure 6.11: Visit time estimation based on kNN

Again to test working of the algorithm a test case was created. This time there are two different patients for whom the expected time is not known. First of all, there is an elderly lady, and second patient is a young student. Generally speaking older people tend to spend more time in the doctor's office not only because they have more health problems, but also because their mobility, hearing and understanding is poorer, which may have an effect on the visit time. On the other hand, young people, if they have to go to the doctor, they want to finish it as soon as possible. For each of these patients nearest neighbourhoods were found and the prediction was done based on those neighbours' mean visit times. Figure 6.12 depicts the result. Both patients tried to register a visit for exactly the same time for the same doctor on the same date. As shown, an elderly lady (left picture) has a predicted time of 32 minutes, and a student 7 minutes. This proves that kNN algorithm worked.

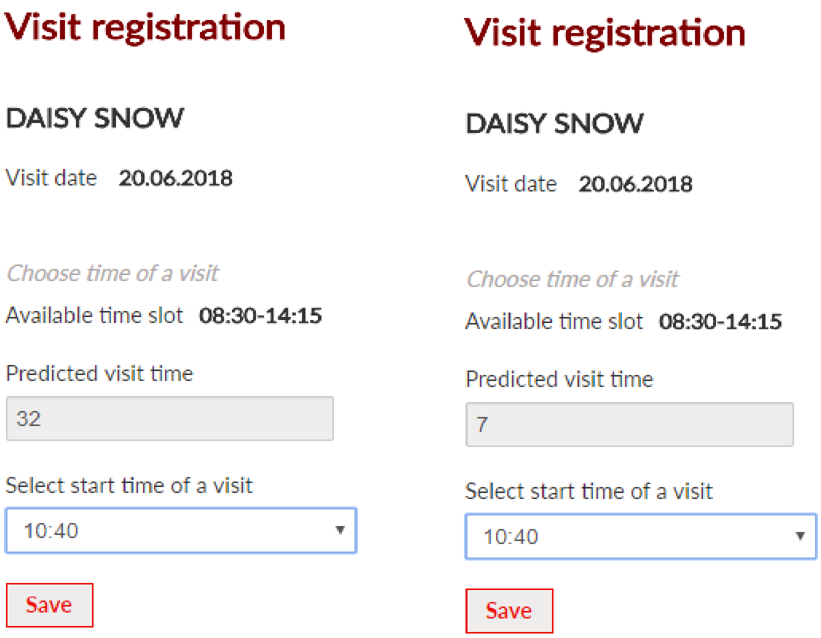

Figure 6.12: kNN classification result

6.1.4.3. Final prediction

Having the results of time prediction from both methods, linear regression and kNN there is a need to infer the result from both predictions. Unfortunately, data available in the system are not informative enough to propose an effective prediction algorithm that could be used in real world system, as there are a lot of factors that can influence the result. Hence, the created algorithm is just a very simplified version used in order to show that the concept of AI and Machine Learning can be used for automatic visit time prediction.

The simplest way of selecting the final prediction would be a weighted average of both predictions. However, weights of those factors should also be learned from training examples and adjusted in the same way as choosing right degree of polynomial in regression. Not having enough data it was decided that the final prediction would depend on the number of visits for a given patient in order to show that the more data is accessible for the given pair doctor-patient, the more algorithm should rely on regression. If not many visits have been recorded, then kNN classification result should have more impact of the final result. Figure 6.13 presents a flowchart illustrating how final prediction is constructed.

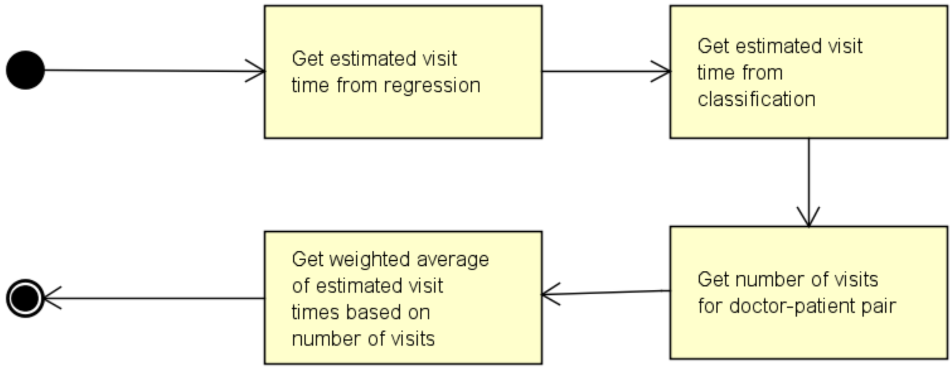

Figure 6.13: Calculation of final visit time estimation.

6.1.5. Future development

As the created Web Application is just a prototype of the real software there are many things that could be improved or embedded in the system. In this chapter few of such cases will be described.

First of all, creation of doctors schedule does not take into account multiple issues. There is only one schedule, which is applied immediately, which means that administration cannot change a schedule few weeks before, which would allow informing about the changes in advance. In real world scenario schedules should be bound with an entry possessing data on when such schedules are applicable. Furthermore, if a schedule is changed then something has to be done with patients that have already registered for visits according to the previous schedule. This prototype is not coping with such problems.

The next issue is connected with registration for visits. Tough it is possible to do so, the prototype does not allow cancellation or rescheduling. Furthermore, there is no reminders of visits. What could be implemented in future versions of the system would be some SMSs or emails send whenever a visit is approaching.

When it comes to new features, there are plenty of functionalities which would be useful in such a system. The most important one not only from patients and doctors point of view but also from AI perspective is collecting medical history of a patient. Created prototype is focused only on administrative part of health centre management, which includes account creation, visit registration or patient and doctors management, however, it does not have many functionalities in the aspect of conducting a visit. The only thing that is shown during a visit are patient data and Google Track. Though Google Track was implemented to show that some extra information can be presented in this application, however, there are other data which would be much more useful for the doctor. Furthermore, after having a meeting with experts from ZOZ

Polesie one of the conclusions was that visit times can be repeatable for people with similar disease units, hence gathering data about medical history would make the AI much more accurate. Furthermore, other functionality which could make the system more appealing and useful to doctors and patients would be introducing a mechanism of e-prescriptions. This is a way of electronically generating, transmitting and filling medical prescriptions instead on placing them on paper. It would allow accessing of error-free, accurate, and understandable prescriptions by a patient and pharmacies. However, it would raise a lot of security issues connected with data privacy.

As it comes to security issues, the security of the API could be improved. The API is a component that allows connection between the Web and the Mobile Applications, hence it is vulnerable to potential attacks. There are plenty of different ways of securing the connection. First of all, the API should be available with SSL, however, it requires funding to buy an SSL, hence, for the prototype it was decided not to use it. Also there could be a method of tokens implemented in order to allow given API functions only for authorised users. This would be extremely important when storing medical history of patient would be implemented, as only authorised doctors should be able to see these vulnerable data.

These are only examples of how the Web Application could be improved, however, it shows a potential of creation of the real life software from the implemented prototype.

6.2. Mobile Application

This chapter will present more technical details of the subsystem of Mobile Application. It will contain the description of the implementation that needed to be done in order to fulfil the functionalities that were presented in the previous chapter.

The proposed solution requires the Mobile Application to control the entrance and leaving of patients in order to provide data for teaching of the AI algorithm. This is the primary goal of the Mobile Application and all functionalities that were implemented were directed to achieve this goal. Furthermore, this application is also aimed to save time of doctors, as all the patients data are visible automatically on the doctor's screen once a patient enters a room. Also patients do not need to wait in the reception for a long time, as the only thing they need to provide is the PESEL number and they will be given a identifying NFC tag.

6.2.1. Chosen Operating System

It was decided that to create the Mobile Application Android operation system would be used. Android is a mobile operating system developed by Google, based on a modified version of the Linux kernel and other open source software and designed primarily for touchscreen mobile devices such as smartphones and tablets.

Before choosing of the technology there was a research conducted on which operating system is more popular. During the past years Android had the superiority in users all around the world against iOS, which is shown on Figure 6.14.

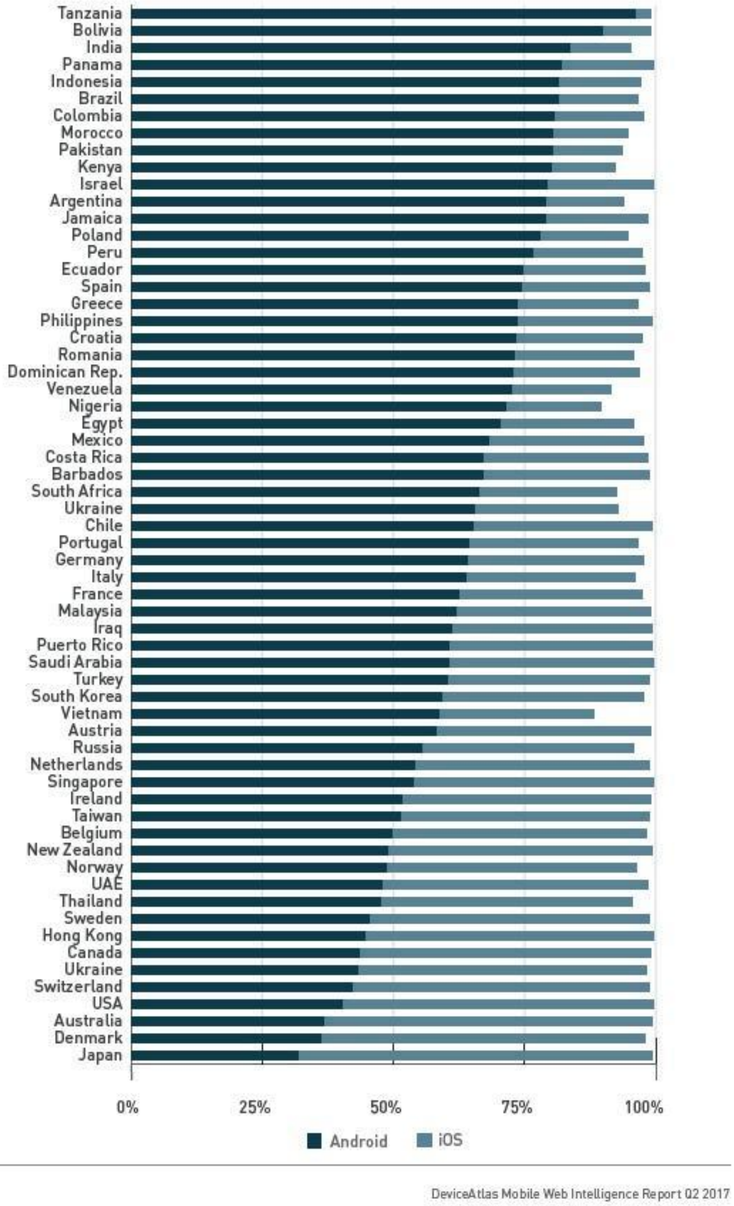

Figure 6.14: Mobile device operating system popularity

However, this was not the main reason behind the choice of an Operating System. To develop an application for iOS Apple requires possessing an Apple computer. Furthermore, for Apple products writing to NFC has not been implemented. Because of that the choice was clear, application had to be made in Android.

6.2.2. Applied technology

To develop the application in Android, Android Studio was chosen. Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development.

It was decided that the minimal version of Android compatibility is 4.0 as according to the conducted research, a vast majority of users are using Android of this or higher version. Because of this decision some external libraries for the implementation could be used.

When it comes to the NFC part of the system, five tags were purchased by the team and used for the process of implementation and testing.

For the software creation and testing following technologies and tools were used:

- NFC tags
- Windows and Mac OS
- Samsung S7 Edge with Android 7.0

6.2.3. Main modules of the system

The Mobile Application is composed on four main modules, which are Login, NFC reader, NFC writer, and Encryption and decryption algorithm. Each of these modules will be described in this chapter.

Because of the fact that the Mobile and the Web Applications should form a one, interconnected system there was need to have only one database of users. A situation in which different accounts would be needed for logging into both applications would be unacceptable. Hence, there was a method in the API prepared to allow logging into Mobile Application with credentials of the accounts created in the Web Application.

A highly important aspect of the Mobile Application creation was enabling of the communication with NFC. There are two functionalities of the system connected to NFC: reading and writing. Reading from NFC is used when a patient enters a doctor’s office. As a tag is bound to the patient account, once it is detected by a reader device, all the patient information can be accessed by a doctor. Due to security reasons, patients data are not stored on the tag, instead all data are stored in the database of the Web Application and can be accessed only via the API. Once the device detects the tag Mobile Application reads the key written on the tag. This key is in an encrypted form, hence, it has to be decrypted before further processing. Such a decrypted key is used to communicate with the API. This communication is used first of all to mark entrance and exits of a patient in the doctor's visit, which is saved in Visit_Time table and used for queue generation and for better accuracy of an algorithm. Also this information is used to show to the doctor information about the patient currently being on a visit.

NFC writer is a part of application that is used to write a patient's key to the tag in an encrypted form. This key is obtained from the API basing on the PESEL number of the patient. If this number fits one of the patient accounts created in the Web Application the API returns a patient ID, which is to be saved on the tag.

Last, but definitely not the least part of the functionalities of the Mobile App is encryption and decryption algorithm. This is a method to protect vulnerable information about patients of the clinic according to the laws of security and data protection. More information about it will be provided in the next subchapter about security.

All of these modules were created in order to provide functionality described in previous chapter.

6.3. Security of the system

As the created solution is connected to healthcare system, which as known, contain private and highly vulnerable data, security is a major issue that has to be taken into account while designing and implementing such a system.

Cybersecurity, computer security or IT security consist on the protection of computer systems that ranges from the possible damage to their hardware, software or electronic data, to disruption or misdirection of the services they provide. The main concern of cybersecurity include controlling the physical access to the systems' hardware, and protecting it against harm that may be done via network access, malicious data or code injection. Also, due to malpractice by operators, whether intentional or accidental, IT security personnel are susceptible to being tricked into deviating from secure procedures through various methods of social engineering.

The field of cybersecurity is of growing importance due to the increasing reliance on computer systems, the Internet and wireless networks such as Bluetooth and Wi-Fi, and due to the growth of "smart" devices, including smartphones, televisions and the enormous quantity small devices that constitute the Internet of Things, which is a trending paradigm.

Each computer system should contain security measures not only because of the awareness of possible dangers but also due to laws and regulations. Unfortunately, the laws conflict in cyberspace and this has become a major cause of concern for the computer security community. Some of the main challenges and complains about the antivirus industry consist on the lack of global web regulations, a global base of common rules to judge, and eventually punish, cyber-crimes and cyber criminals. There is no global law or cyber security treaty that can be invoked to enforce global cyber security issues.

Nowadays, the EU General Data Protection Law replaces the Data Protection Directive 95/46/EC and is designed to harmonize data privacy laws across Europe, to protect and empower all EU citizens' data privacy and to reshape the way organizations across the region approach data privacy. According to this law, its violation can be fined for up to the 4% of the annual global turnover for breaching GDPR or €20 Million. When referring to violation it is understood as consisting of revealing some personal data of costumers or, in some cases, patients of the entire system.

The definition of "personal data" can be described as any sort of information related to an identifiable person who can be directly or indirectly identified in particular by reference to an identifier. This definition encapsulates a wide range of personal identifiers that refer to personal data, including name, identification number, location. There's also a digital stamp identifier, that reflect changes in technology and the way organizations collect information about people. In case of the existence of users that are under 16 years old, parental consent is required to process the personal data.

The GDPR law states data protection must be appointed in the case of: (a) public authorities, (b) organizations that engage in large scale systematic monitoring, or (c) organizations that engage in large scale processing of sensitive personal data (Art. 37). In case of the system presented in this report, there are sensitive personal data stored. Consequently, there is a need of right protection against possible attacks.

6.3.1. Encryption

Since the security aspect is so important nowadays, especially when managing high sensitive data, one part of the project consisted on writing the encryption program. When building an encrypting program an important part is to decide which encryption method suits the best for what is to be protected.

Among the main algorithms, AES (Advanced Encryption Standard) is one of the bests avoiding all sorts of attacks, being only vulnerable to brute forcing. The advantage is that a 256 bit key can be used, needing approximately 8 octoecillion dollars (10^44 times the gross world product) consumed in energy in order to achieve it with the actual technology.

RSA is another important algorithm, but it has a major flaw which has to be taken into account. It's asymmetric and it works with the use of public and private keys. Being almost impossible to decrypt the message, if a hacker manages to access the device that sends the information, it is possible to obtain the private key, making the deciphering of the message an easy task.

Since another weakness of encryption is the obtention of the key used to encrypt the data the team decided to find a way to obtain it that proves to be as random as possible. The method used is to gather in the different data of the device that is decrypting the message, choose one of them randomly and extract the first 256 bytes. This method can make it extremely difficult for an attacker to follow a pattern, especially if he is not tracking only one device.

As created solution is just a prototype security protection measures were directed towards NFC, as it was identified as a most probable point of failure. Tags do not have any protection from access, when a person steals a tag then data can be accessed as well. There is a lot of problems in NFC security that can be solved depending on how you secure are the data. Some examples of them are:

- Eaves Dropping

Signal captures, air interface and communication channels decoding are main issues which we will avoid by placing and already encrypted key inside the NFC.

- Man in the middle attacks

Altering information of one device and sending it to the other is a huge risk, since critical information are given away. In order to avoid this, when a key is received it should not be stored in the database. Sending of a signal to delete a key stored inside the NFC could be one of the possible ways of protection from this type of attacks.

- Theft

This can be avoided by changing the key of the patient from the database after some time and sending a delete signal in case that key inquiry has been received.

In order to avoid all of this attacks the system included encryption of the actual content of the NFC with program created by the team. It works by encrypting an ID of a patient that is accessed via API and writing it to a tag. Then, even it the tag would be lost, there is no possibility of accessing the data inside it. When reading from the NFC decryption takes place.

Field of security measures is such a vast topic that there are always things that can be made in a better way. The most important part of future development would be to secure connection between the Mobile and Web Applications. Key should be encrypted on the Mobile Application and decrypted on the server side. With these measures, an attacker should decrypt the packet that contains the key that is being send to the server side and, once decrypted another decryption for the actual key should be done. For better visualisation of the system communication process Figure 6.15 has been prepared.

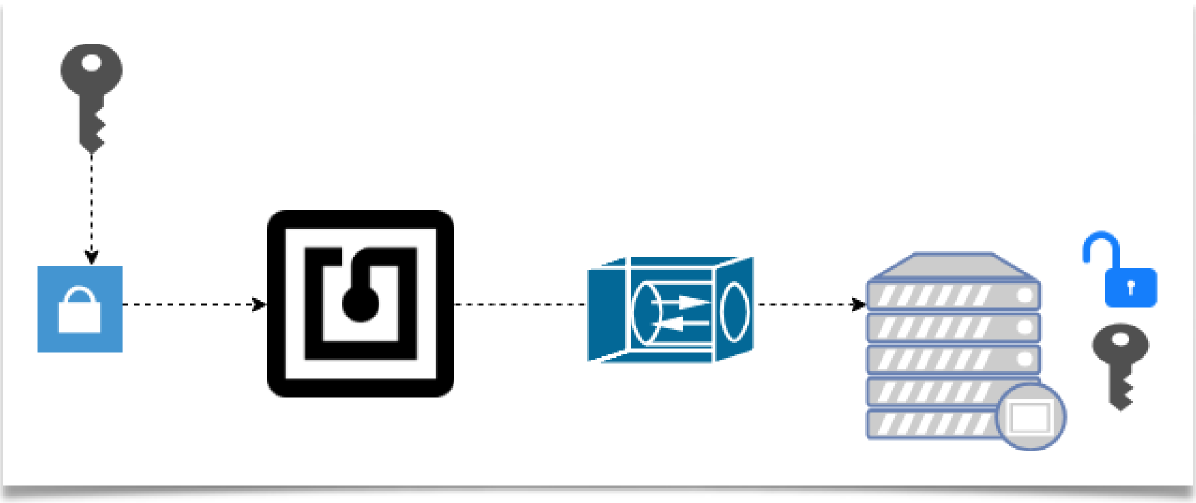

Figure 6.15: Diagram showing shows the main communication process in the system

As shown in the Figure above the key is encrypted before being stored in an NFC tag and when communicating with the server, it remains encrypted. Decryption is a task done by the server side.

Adding another security layer in the communication channel between the NFC tag and the server adds up to a highly complex scenario for an attacker to be successful.

6.3.2. Used encryption algorithm

This chapter will present the details on how the AES encryption algorithm works.

AES is based on a design principle known as a substitution–permutation network, a combination of both substitution and permutation, and is fast in both software and hardware. AES is a variant of Rijndael which has a fixed block size of 128 bits, and a key size of 128, 192, or 256 bits. By contrast, the Rijndael specification per se is specified with block and key sizes that may be any multiple of 32 bits, with a minimum of 128 and a maximum of 256 bits.

AES operates on a 4 × 4 column-major order matrix of bytes, termed the state, although some versions of Rijndael have a larger block size and have additional columns in the state. Most AES calculations are done in a particular finite field.

The principle of work of AES algorithm is as follows. AES algorithm combines each byte of the state with a block of the round key using bitwise xor. After that, SubBytes round starts. In this round each byte of the state matrix is replaced by with a subByte using a subByte substitution box, the Rijndael S-box. This operation provides the non-linearity in the cipher. This operation is presented in Figure 6.16.

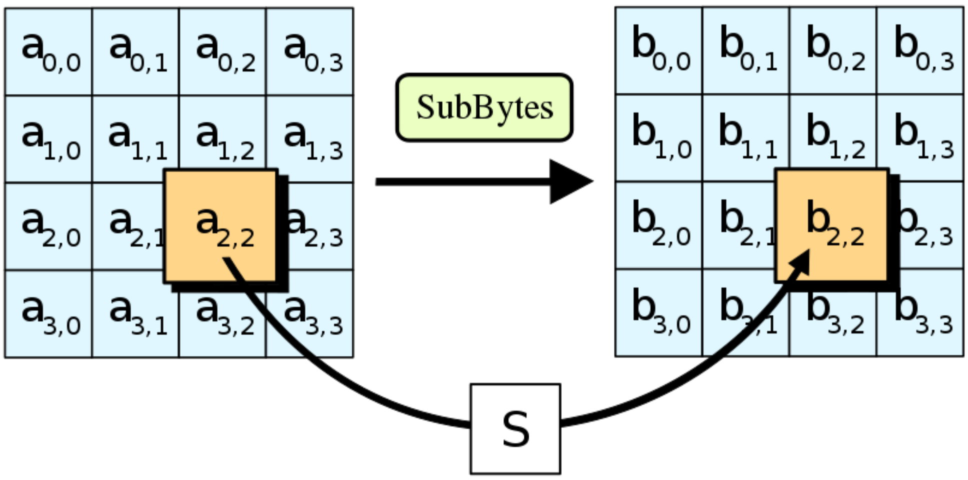

Figure 6.16: AES algorithm – SubBytes step

The next step is the ShiftRows step. The ShiftRows step operates on the rows of the state. It cyclically shifts the bytes in each row by a certain offset. For AES, the first row is left unchanged. Each byte of the second row is shifted one to the left. Similarly, the third and fourth rows are shifted by offsets of two and three respectively. For blocks of sizes 128 bits and 192 bits, the shifting pattern is the same. In this way, each column of the output state of the ShiftRows step is composed of bytes from each column of the input state. This step is depicted on Figure 6.17.

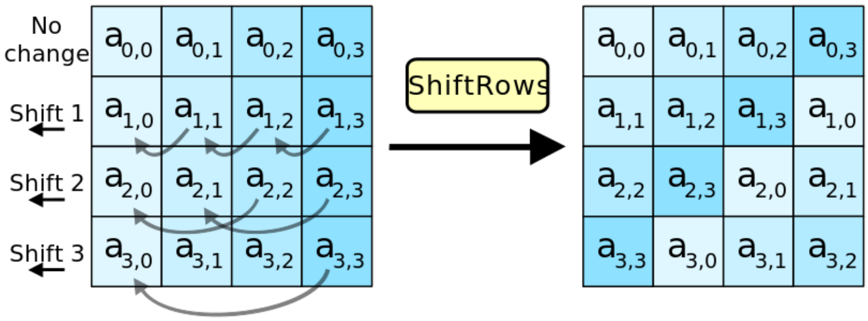

Figure 6.17: AES algorithm – ShiftRows step

Once ShiftRowes is finished, the MixColumns step is started, in which the four bytes of each column of the state are combined using an invertible linear transformation. The MixColumns function takes four bytes as input and outputs four bytes, where each input byte affects all four output bytes. Together with ShiftRows, MixColumns provides diffusion in the cipher.

During this operation, each column is transformed using a fixed matrix (matrix left-multiplied by column gives new value of column in the state). Again presented in a graphical form on Figure 6.18.

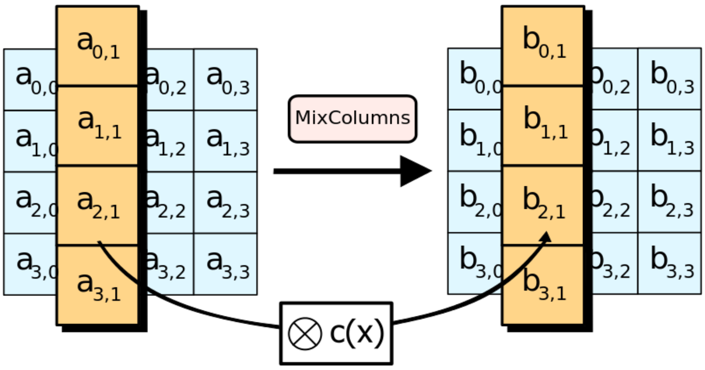

Figure 6.18: AES algorithm – MixColumns step

The last step is AddRoundKey step, the subkey is combined with the state. For each round, a subkey is derived from the main key using Rijndael's key schedule. Each subkey is the same size as the state. The subkey is added by combining each byte of the state with the corresponding byte of the subkey using bitwise XOR.

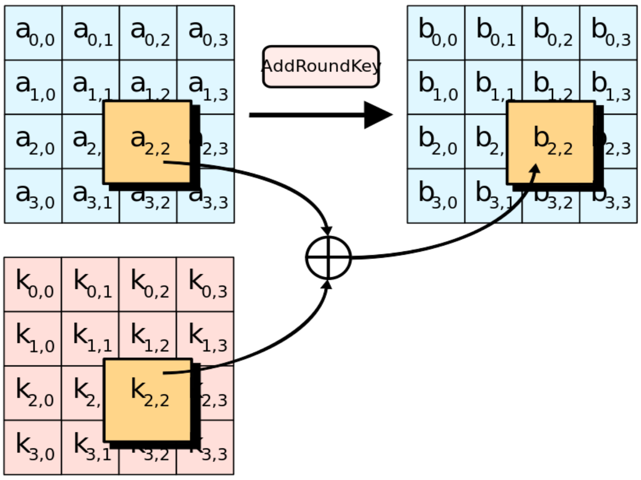

Figure 6.19: AES algorithm – AddRoundKey step

6.3.3. Testing

Since in the project's prototype there is no SSL certificate, a test can be done to measure the security of the system. A concrete scenario will be shown, were a potential attacker is sniffing data from the network. This is a common situation were an individual can use one of plenty of tools that exist to get information from a particular network. A healthcare centre is an open space that uses a network that can be vulnerable or even open to anyone, hence, it can be assumed that the data that travels is exposed.

Since mostly the internet is based on packet-switched networks, the communication between devices is made through data packets with a specific header and payload that are send and received by communications protocols. This packets travel throughout different networks to reach their destination, and, during that process they're highly exposed.

In the following scenario we will use a network sniffer called Wireshark to analyse the different network packets send by the system. This is the first approach an attacker would follow.

Figure 6.20 shows the result of sniffing packets that connects the NFC content to the system.

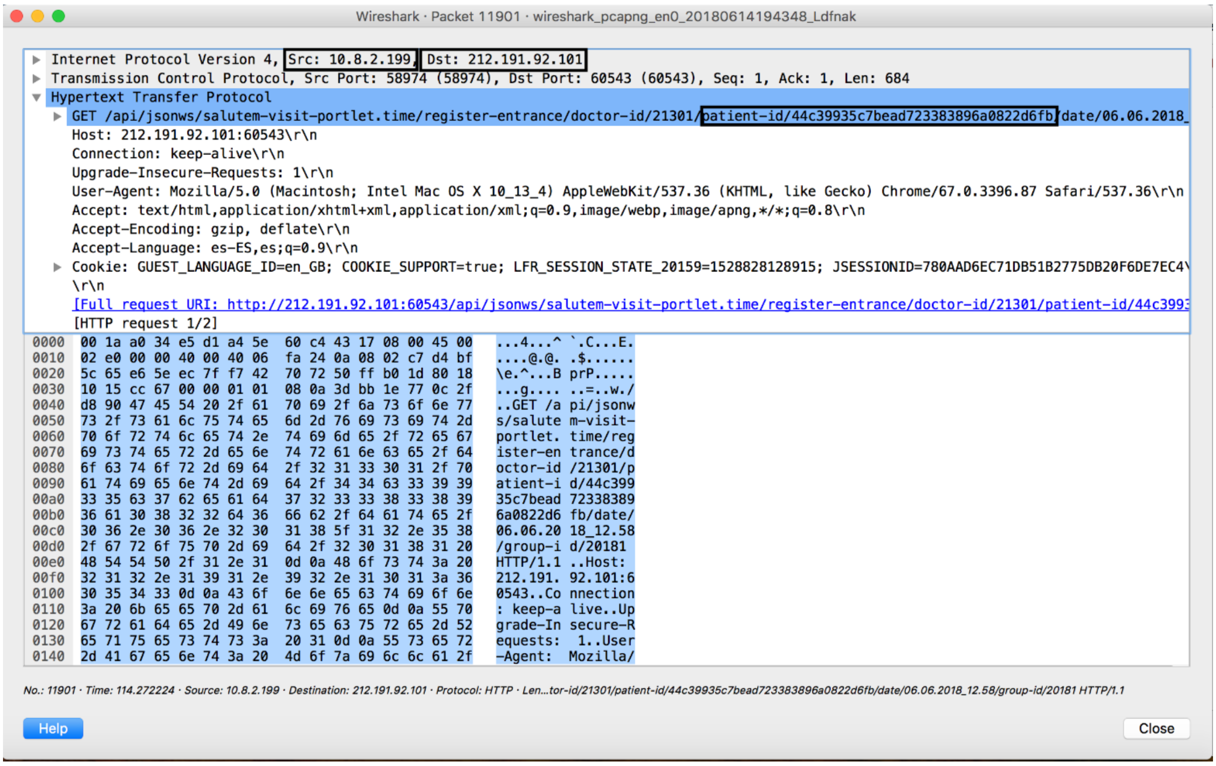

Figure 6.20: Result of sniffing the packet that connects the NFC content to the system

As shown in figure above, there is an encrypted id when trying to get the needed information. Since the encryption is done in a different process, in order to gain access to the sensitive information, a brute force attack is needed, which, as mentioned before, is hardly viable nowadays. With the needed structure and an SSL certificate to enable https, an additional encryption layer would exist making the system more secure.

7. Conclusions

The project conducted on the European Project Semester by the team named Salutem was connected to Smart Healthcare in Smart Cities. The team made a research that allowed gaining knowledge about the field and fining current problems in healthcare that can be solved in a smart way. The identified problems are in such areas as financial, organizational, logistical, and technological. Financial problems are present in every country, however, governments that do not implement strong policies in theirs healthcare systems, require the citizens to pay high out of pocket expenses. This can be seen in India and United States. Regardless of the development of the country, both populations give high percentage of their salaries to medical expenses. From the organizational point of view, it can be seen that the waiting times in queues for the doctor are highly dependent on lack of medical personnel and the time the doctor spends on bureaucratic work. The logistical problems in Poland's healthcare system are mostly due to improper use or maintenance and low budget allocation towards medical equipment and lack of qualified personnel. The last set of identified problems, which are technological problems include security issues and data privacy, lack of standardization in the used equipment as well as various regulations form the government that are delaying the implementation of new technologies in healthcare. Unfortunately not all problems can be solved during one semester, hence, the team had to limit the scope of the project. It was decided that the project will be focused on organisational and technological problems. The goal was to create a solution that will optimise work of employees of health centres with use of smart technology.

Throughout the semester there has been a lot of obstacles and problems to be solved not only in terms of the solution implementation but also in communication between team members. Fortunately, the team has put a lot of effort to overcome those obstacles and as a result, a prototype of the solution for the defined problem was created. It is a distributed system composed of two main components, The Mobile Application and The Web Application, that communicate with each other via the API.

The Web Application was developed to be used by patients, doctors and administrative staff. The patient can register in the system and create a patient account without a need of going to the healthcare facility. This is beneficial not only for patients, as they save time, but also for administration, because it limits the amount of paperwork required to register new patients. Then, there is a possibility of booking a visit for a specific doctor. All doctor schedules are updated automatically, hence, a patient can always see how many free time slots are available for a specific date. Moreover, in a day of a visit, a patient can see the queue for the doctor online. This queue contains information on finished and in queue visits. Furthermore, there is an information about time delays, if some urgent situation happen. Thanks to that, patients do not need to waste their time waiting next to the doctor's office because there was some emergency. The last feature made available for patients is uploading of Google Track, which represent location information gathered via GPS. As a doctor, a user can see which patients have booked visit for a given day. Furthermore, as the whole system allows urgent and unexpected visits, a doctor automatically see the data of a patient who entered the office. Administrative part is responsible for patient management, which includes confirmation of accounts, and for doctors management. It is also a role of the administration to fix the doctor schedules. The Web Application has also been enriched with an AI algorithm used to predict visit times. This prediction has been prepared with use of linear regression and kNN classification. Those two algorithms together estimate the time of a visit for a given patient doctor pair and for a given date and time.

The Mobile Application was created to gather data from patients' visits and send it to the Web Application. A highly important part of this subsystem is connected with reading and writing to NFC tags. Communication with NFC tags is used to shorten the time needed for doctors to access patients data, as once a patient enters the room then a doctor can see all their data immediately. Because of the fact that the created system contains vulnerable and private data there was a need of introducing security measures that will ensure data privacy and safety. For this reason there has been an encryption and decryption algorithm proposed.

The developed system fulfils all the requirements and expected functionalities that were planned in the beginning of the semester. However, still some functionalities could be added. A major improvement for the system would be gathering medical data of patients, as it would give a doctor a possibility to view the whole history of patients' diseases, prescribed medications, allergies etc. Furthermore, such data would be beneficial for the AI algorithm, as visit times can be repeatable for the same disease units. However, storing such vulnerable data would mean that the system needs to comply with the highest standards of security. The very first point to improve the security would be the connection between the Mobile and the Web Applications, which is the API. First of all, it should be made available via secured protocol https. Also the system could be adjusted to use the mechanism of automatically generated tokens, which would be necessary for each call to API. In such a way only authorised personnel would have access to patient's data. Moreover, a new functionality of e-prescriptions could be added to make the visit even more automated. These are only few examples on how to improve the system, however, it shows that though being only a prototype, the developed system shows a lot of potential and possibilities for improvement.

Finding a problem and developing the solution for it gave the team a lot of experience and new opportunities. The topic was highly interesting and important. It gave the team a possibility of creating innovative solutions with use of previously unknown technologies and paradigms. What is more, for each member of the team the project gave an opportunity to work in a fully international and intercultural team. Though it created some problems with communication, it certainly was a great experience that will be beneficial in the future.

References

[1] Margaret Rouse – What is Smart City? URL: https://internetofthingsagenda.techtarget.com/definition/smart-city (visited on 10/05/18)

[2] Gemalto - Secure, sustainable smart cities and the IoT URL: https://www.gemalto.com/iot/inspired/smart-cities (visited on 12/05/18)

[3] Michael Alba – The guide to Low-Power Wide Area Networks URL: https://www.engineering.com/IOT/ArticleID/16869/The-Guide-to-Low-Power-Wide-Area-Networks.aspx (visited on 14/05/18)

[4] Gemalto - Secure, sustainable smart cities and the IoT URL: https://www.gemalto.com/iot/inspired/smart-cities (visited on 10/05/18)

[5] Margaret Rouse – ICT URL: https://searchcio.techtarget.com/definition/ICT-information-and-communications-technology-or-technologies (visited on 15/05/18)

[6] Techopedia - Definition - What does Silo mean? URL: https://www.techopedia.com/definition/25939/silo (visited on 25/05/18)

[7] Sihou Zhang - the role of information and communication technology for smart city development in China URL: https://digi.lib.ttu.ee/i/file.php?DLID=8694&t=1 (visited on 22/05/18)

[8] Rodger Lea – Smart Cities: An Overview of the Technology Trends Driving Smart Cities URL: https://www.ieee.org/content/dam/ieee-org/ieee-web/pdf/ieee-smart-cities-trend-paper-2017.pdf (visited on 22/05/18)

[9] CISCO - The Internet of Things Reference Model URL: http://cdn.iotwf.com/resources/71/IoT_Reference_Model_White_Paper_June_4_2014.pdf (visited on 22/05/18)

[10] CISCO - The Internet of Things Reference Model (Pages 4-5) URL: http://cdn.iotwf.com/resources/71/IoT_Reference_Model_White_Paper_June_4_2014.pdf (visited on 22/05/18)

[11] CISCO - The Internet of Things Reference Model (Pages 6-8) URL: http://cdn.iotwf.com/resources/71/IoT_Reference_Model_White_Paper_June_4_2014.pdf (visited on 29/05/18)

[12] Andrzej Bielecki, Ryszard Stocki - Systems Theory Approach to the Health Care Organization on National Level. URL: https://www.researchgate.net/publication/220231784_Systems_Theory_Approach_to_the_Health_Care_Organization_on_National_Level DOI: 10.1080/01969722.2010.511533 (visited on 29/05/18)

[13] Ileas Pramanik, Raymond Y. K. Lau, Haluk Demirkan, Md Abul Kalam Azad - Smart Health: Big Data Enabled Health Paradigm within Smart Cities (Article 2.2) URL: https://www.researchgate.net/publication/317721544_Smart_Health_Big_Data_Enabled_Health_Paradigm_within_Smart_Cities (visited on 5/06/18)

[14] Ileas Pramanik, Raymond Y. K. Lau, Haluk Demirkan, Md Abul Kalam Azad - Smart Health: Big Data Enabled Health Paradigm within Smart Cities (Article 2.3) URL: https://www.researchgate.net/publication/317721544_Smart_Health_Big_Data_Enabled_Health_Paradigm_within_Smart_Cities (visited on 5/06/18)

[15] Silvia Palasca, Elisabeta Jaba - Economic Crisis' Repercussions on European Healthcare Systems (Intro) URL: https://www.sciencedirect.com/science/article/pii/S2212567115005687 (visited on 6/06/18)

[16] Silvia Palascaa, Elisabeta Jabab - Economic Crisis' Repercussions on European Healthcare Systems (4.2 Interpretations) URL: https://ac.els-cdn.com/S2212567115005687/1-s2.0-S2212567115005687-main.pdf?_tid=0da919d2-6179-48d6-9b8a-4530678d72be&acdnat=1529385566_fbdfc75d445a169fbe4e8e8cd2fe8e93 DOI: 10.1016/S2212-5671(15)00568-7 (visited on 6/06/18)

[17] UNDP – Human Development Report 2016 URL: http://hdr.undp.org/sites/default/files/2016_human_development_report.pdf (visited on 7/06/18)

[18] Steven Lott - India's growing healthcare crisis - challenges of equity, capacity and funding URL: http://www.wardhealth.com/access-healthcare/indias-growing-healthcare-crisis-challenges-equity-capacity-and-funding (visited on 7/06/18)

[19] Swagata Yadavar - Budget 2018: India's Healthcare Crisis Is Holding back National Potential URL: http://www.indiaspend.com/cover-story/budget-2018-indias-healthcare-crisis-is-holding-back-national-potential-29517 (visited on 7/06/18)

[20] Susan Brink - What Country Spends The Most (And Least) On Health Care Per Person? URL: https://www.npr.org/sections/goatsandsoda/2017/04/20/524774195/what-country-spends-the-most-and-least-on-health-care-per-person?t=1529472328265 (visited on 7/06/18)

[21] Elisabeth Rosenthal - As Hospital Prices Soar, a Stitch Tops $500 URL: https://www.nytimes.com/2013/12/03/health/as-hospital-costs-soar-single-stitch-tops-500.html?pagewanted=all (visited on 12/06/18)

[22] BLACKPANTERS - Polish Healthcare Industry Report URL: https://blackpartners.pl/wp-content/uploads/2016/08/Rynek-medyczny-raport-Blackpartners.pdf (visited on 12/06/18)

[23] BLACKPANTERS - Polish Healthcare Industry Report (Health Professionals) URL: https://blackpartners.pl/wp-content/uploads/2016/08/Rynek-medyczny-raport-Blackpartners.pdf (visited on 12/06/18)

[24] BLACKPANTERS - Polish Healthcare Industry Report (page 19) URL: https://blackpartners.pl/wp-content/uploads/2016/08/Rynek-medyczny-raport-Blackpartners.pdf (visited on 12/06/18)

[25] Kenneth J. Klassen, Reena Yoogalingam - Improving Performance in Outpatient Appointment Services with a Simulation Optimization Approach URL: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1937-5956.2009.01021.x DOI: https://doi.org/10.1111/j.1937-5956.2009.01021.x (visited on 12/06/18)

[26] Marion Mass, Kenneth Fisher - Bureaucrats, not doctors, are making healthcare so expensive URL: https://www.washingtonexaminer.com/bureaucrats-not-doctors-are-making-healthcare-so-expensive (visited on 13/06/18)

[27] Heidi Moawad - Top 8 worst administrative hassles according to physicians URL: http://www.medicaleconomics.com/medical-economics/news/top-8-worst-administrative-hassles-according-physicians/page/0/2 (visited on 13/06/18)

[28] Madison Ngafeeson - Healthcare Information Systems: Opportunities and Challenges URL: https://commons.nmu.edu/cgi/viewcontent.cgi?article=1012&context=facwork_bookchapters (visited on 13/06/18)

[29] Madison Ngafeeson - Healthcare Information Systems: Opportunities and Challenges (page 7) URL: https://commons.nmu.edu/cgi/viewcontent.cgi?article=1012&context=facwork_bookchapters (visited on 13/06/18)

[30] Steven Saslow - 5 Information Technology Challenges Faced by Healthcare Organizations URL: https://itgcloud.com/5-information-technology-challenges-faced-by-healthcare-organizations/ (visited on 14/06/18)

[31] Agusti Solanas, Fran Casino - Trends and challenges in smart healthcare research: A journey from data to wisdom URL: https://ieeexplore.ieee.org/document/8065986/ (visited on 14/06/18)

[32] Mohd Vaseem Ismail, Sakhi john - Information technology challenges and issues in Health and Hospital Sector- critical Evaluation URL: http://www.ijetee.org/Docs/Volume%2011/Issue%205/15.pdf (visited on 14/06/18)

[33] IBS – Population Ageing, Labour Market and Finance in Poland URL: http://ibs.org.pl/en/events/population-ageing-labour-market-and-public-finance-in-poland/ (visited on 19/06/18)

[34] Commission Staff Working Document - Country Report Poland 2017 URL: http://ibs.org.pl/app/uploads/2017/04/2017-european-semester-country-report-poland-en.pdf (visited on 19/06/18)

[35] COMARCH – COMARCH medNote URL: https://www.comarch.com/healthcare/products/comarch-mednote/ (visited on 19/06/18)

[36] W3Schools - AJAX Introduction URL: https://www.w3schools.com/xml/ajax_intro.asp (visited on 10/06/18)

[37] Google – Google Maps Help URL: https://support.google.com/maps/answer/6258979?co=GENIE.Platform%3DDesktop&hl=en (visited on 10/06/18)

[38] MuleSoft - What is an API? (Application Programming Interface) URL: https://www.mulesoft.com/resources/api/what-is-an-api (visited on 10/06/18)

[39] Comentum - What is a Content Management System (CMS)? URL: https://www.comentum.com/what-is-cms-content-management-system.html (visited on 10/06/18)

[40] LIFERAY URL: https://www.liferay.com/ (visited on 10/06/18)

[41] leon-earl et all. MVC Architecture URL: https://developer.mozilla.org/en-US/Apps/Fundamentals/Modern_web_app_architecture/MVC_architecture (visited on 10/06/18)

[42] Creative Commons Attribution - Points of interest URL: https://wiki.openstreetmap.org/wiki/Points_of_interest (visited on 10/06/18)

[43] LIFERAY - invoking json web services URL: https://dev.liferay.com/develop/tutorials/-/knowledge_base/6-2/invoking-json-web-services (visited on 11/06/18)

[44] Techopedia – Artificial Intelligence (AI) URL: https://www.techopedia.com/definition/190/artificial-intelligence-ai (visited on 12/06/18)

[45] Andrew Ng – Machine Learning URL: https://www.coursera.org/learn/machine-learning (visited on 12/06/18)

[46] Statistics Solutions - What is Linear Regression URL: http://www.statisticssolutions.com/what-is-linear-regression/ (visited on 12/06/18)

[47] Niklas Donges - Gradient Descent in a Nutshell URL: https://towardsdatascience.com/gradient-descent-in-a-nutshell-eaf8c18212f0 (visited on 12/06/18)

List of Figures

List of tables