

EPOKA UNIVERSITY

Medical Management System Requirements Specification

Version 1.0

April 2, 2018

Group members:

Jaser Sokoli

Alvi Lika

Neritan Hoxha

Ibrahim Pisha

Kris Ferko

Rigels Hita

**Table of Contents**

[1.Executive Summary 3](#_Toc510459586)

[1.1 Project Overview 3](#_Toc510459587)

[1.2 Purpose and Scope of this Specification 3](#_Toc510459588)

[2. Product/Service Description 4](#_Toc510459589)

[2.1 Product Context 4](#_Toc510459590)

[2.2 User Characteristics 4](#_Toc510459591)

[2.3 Assumptions 4](#_Toc510459592)

[2.4 Constraints 4](#_Toc510459593)

[2.5 Dependencies 5](#_Toc510459594)

[3. Requirements 5](#_Toc510459595)

[3.1 Functional Requirements 5](#_Toc510459596)

[3.2 Non-Functional Requirements 7](#_Toc510459597)

[3.2.1 User Interface Requirements 7](#_Toc510459598)

[3.2.2 Usability 7](#_Toc510459599)

[3.2.3 Performance 7](#_Toc510459600)

[Capacity 7](#_Toc510459601)

[Availability 8](#_Toc510459602)

[Latency 8](#_Toc510459603)

[3.2.4 Manageability/Maintainability 8](#_Toc510459604)

[3.2.4.1 Monitoring 8](#_Toc510459605)

[3.2.4.2 Maintenance 8](#_Toc510459606)

[3.2.4.3 Operations 8](#_Toc510459607)

[3.2.5 System Interface/Integration 8](#_Toc510459608)

[Network and Hardware Interfaces 9](#_Toc510459609)

[Systems Interfaces 9](#_Toc510459610)

[3.2.6 Security 9](#_Toc510459611)

[Protection 9](#_Toc510459612)

[Authorization and Authentication 9](#_Toc510459613)

[3.2.7 Data Management 9](#_Toc510459614)

[3.2.8 Standards Compliance 10](#_Toc510459615)

[3.2.9 Portability 10](#_Toc510459616)

[3.3 Domain Requirements 10](#_Toc510459617)

[4. User Scenarios/Use Cases 10](#_Toc510459618)

# 1.Executive Summary

## Project Overview

This project conforms with a simple application which is a Web-Based and consists as a simplifier management system for doctor visits . One of the main usage of our project is to leave an appointment online and keep track of records for each patient in a online form for a period of time and stored and connected to your health card instead of actual use , on paper.

In order to finish the idea we need to cowork with actual city clinics so we can't miss any detail. Family doctors will be able to leave appointmens for their patients to forward medication if their state is more serios , for example to the 'Specialist Doctor' or even 'the hospital'. They will be able to leave notes for each patient and their medical recors or their specific details.

Appointments can be cancelt by the patiens through the email with a automated response . An display of his timetable will we offered besides the one of Hospitals' doctors or Specialist doctor which he uses to book his patients visits or further treatment.

All this data can be also accesed for urgent cases because his personal and medical datas are displayed there . We intend to wipe of queue so the public can have simpler ,faster and better healthcare .

## Purpose and Scope of this Specification

The purpose of this specification is to assess the current state of the product design and to document the entire process based on design issues and the audience.

This specification encompasses several aspects of the process being discussed in an as broad scope as possible. Thus in this scope we address the following:

* In depth documentation of the features of the product
* Technical overview of the system processes and views
  + This is discussed in Part 2.1 and throughout the document
* User and System Requirements
* Components & Functional/non-functional requirements
  + These are discussed in Part 3 in some detail
* Definition of users’ means of using and accessing the product
  + Use cases/scenarios discussed in Part 4
* Dependencies and Constraints
  + These are discussed in Part 2.4/5 of the Document

Aspects not included in the scope are as follows:

* Legislative requirements for the product
* Auditing and financial considerations of the product

# Product/Service Description

Long queues of people in medical institutions in Elbasan is a prevalent problem which requires an immediate solution. The Compentent Public bodies for giving this services in Elbasan are:

”The house doctor”, ”Ambulance” and the ”Hospital”.

It operates on a hierarchical structure starting from “House doctor” until the hospital.

“MMS”(Medical Management System) will aim to reduce the obstacles faced by the patients of this institution and improve the data-maintenance process. It will be used by "Bashkia Elbasan" to efficiently avoid the queues on providing the health service for the patients.

It should help in the operations within the institution like: keeping the track of the patients examinations history; dynamic calendar of doctors and easily accesable from the patients.

The “house doctor”will be able to schedue an appoiment online using this facility by watching on the doctors calendar for empty places In his calencar. No log in needed for the clients but all the data must be stored on their unique health insurance number for example(id:1234).

All the doctors must have their log ins where they can see their calendar and in the end of the examination they will be able to leave a feedback or a recipt wich can also be printable.

## Product Context

Our software is directly related to the "Elbasan Hospital". It is concepted to be an independent system that will make the direct connection between the client and the patient.

## User Characteristics

There are four types of users that will interact with the system :

* patients ,
* doctors ,
* admin
* secretary

1. Patients will able to book a free time slot with a doctor without login, but just by using his unique insurance number. He can use this health card number to check all his examinations he has ever done in the medical system.
2. Doctors will have their own accounts given by the admin also they will have a time Slot Management using Calendar, Writing Client Report on occupied time slot.
3. Secretary will have her own account given by admin and Can forcefully occupy and cancel doctor time slot.
4. Administrator manages the staff management system. He can add a new staff member or remove one. He randomly generates a password and sends it confidentially to worker. The worker itself can reset the password.

## Assumptions

## It is assumed that some actions performed behind the scenes are performed regularly according to law. Therefore users under Drejtoria e Informacionit, who confirm the assistance, according to “Vendimit te Keshillit te Ministrave Nr.787 date 14.12.2005”. Therefore a client must have a health insurance card. For security issues, it is taken for granted that after administrator assigns a worker to a directory, he send confidentially by mail the random generated password.

## Constraints

The project is constrained by the Internet connection. Since the application fetches data from the database over the Internet, it is crucial that there is stable Internet connection for the application to function.For this project since it is a web based application we have decited to use Pure PHP .This application will be apple to work on different Operating Systems for example on Windows 7,8,10 , Windows XP, Linux,Max etc.

## Dependencies

In order for this web application to work we must frequently gain access on the health insurance card’s database.

# Requirements

## Functional Requirements

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Req#** | **Requirement** | **Comments** | **Priority** | **Date Rvwd** | **SME Reviewed / Approved** |
| R\_0\_1 | Different level users according to their position in the administrate | According to their position, each will have their own profile | 1 | 02/04/2018 | Jaser Sokoli |
| R\_0\_2 | Privileges according to their position. | - | 1 | 02/04/2018 | Jaser Sokoli |
| R\_0\_3 | Exchange of information | Information between levels of hierarchy  . | 2 | 02/04/2018 | Jaser Sokoli |
| R\_0\_4 | Limited access of information | The information will be visible only to the profile of intented user.  . | 1 | 02/04/2018 | Jaser Sokoli |
| R\_0\_5 | Interactivity between the unit and the doctors on duty. | Giving tasks and reporting in real time | 1 | 02/04/2018 | Jaser Sokoli |
| R\_0\_6 | Interactivity with the patients. | The patients can report with the help of the android app | 1 | 02/04/2018 | Jaser Sokoli |
| R\_0\_7 | Reporting to a higher hierarchical level. | - | 1 | 02/04/2018 | Jaser Sokoli |
| R\_0\_8 | Sharing the information restrictively. | The information shared will be higly protective. | 1 | 02/04/2018 | Jaser Sokoli |
| R\_0\_9 | Task problems | If a visit is not completed or is facing difficulties(problems) they should report to the higher up organization and give reason. | 1 | 02/04/2018 | Jaser Sokoli |
| R\_1\_0 | Flow of information | Flow of information between different levels of the hierarchy, directly or indirectly. | 1 | 02/04/2018 | Jaser Sokoli |
| R\_1\_1 | Tasks between levels of hierarchy | Assigning tasks to different level of hierarchy to the lower tier level of this hierarchy. | 1 | 02/04/2018 | Jaser Sokoli |
| R\_1\_2 | Detailed report after terminating tasks | Doctor is required to provide a detailed report to the visit, in order to fill a detailed recipt | 2 | 02/04/2018 | Jaser Sokoli |

## Non-Functional Requirements

### User Interface Requirements

1. Different screen resolutions and apperance based on devices
2. Receive native push notifications in real time
3. Sliding navigation drawer for the product
4. Static navigation drawer for the web page
5. Responsive simple design

### *Usability*

* Accessibility
  + The software shall be easy to access remotely and at all times, since both patients and doctors will use the application on their devices.
  + This software shall be easy to access with only your health card number as an indentification protocoll .

* Responsiveness
  + The software shall be responsive both in data transactions, especially because of the reliance on the servers .

* Flexibility
  + The software shall be easy to update in order to accommodate new requirements like new doctors or secretaries or administrators.
  + The software shall be designed in such a way that allows us to make changes without starting from the beginning.

* Effectiveness
  + The software shall provide both staff and clients with practical tools of managing their data and also with a convenient way of communicating their needs across a platform that avoids wasted time on ques of people.

* Efficiency

* Performance Requirements: The performance of the software is going to depend on the server used. JavaScript and PHP will be used which means that it will be light weighted and the browser won’t crash.
* Space Requirements: We are striving to build a good, strong server. It will be able to support a number of simultaneous users.
* As the software is supposed to perform a country size scale, the servers that will be used at least 1 Terabyte of memory, fast, preferable with SSD, and the operating system is thought to be Linux Ubuntu.
* The central server, which is located to the Central General Directory, which will operate and access the information from all subdirectories will require at least 100 Terabyte of memory.
* As the software is planned to be fast, there should an internet connection of at least 20 dedicated Megabytes of internet, both in download and upload .

### *Performance*

#### Capacity

The backend is built on top of Google’s infrastructure and thus scales very well horizontally.

* Database writes are limited to 2300 per second
* Software starting time will be in no time , other procedures will take time accoring to user internet strength
* Maximum concurrent connections for mobile/web clients are limited to 100000 per database.
* The project is web base so everythings is connected to web server
* Maximum API request size is 8-10 MB
* Maximum number of documents that can be passed to a Commit operation in a transaction is 500
* Maximum number of composite indexes for a database is 200
* Maximum function call depth is 15-20

#### Availability

* The app will be live 24/7
* Is has a very low probability of downtime,
* It will be region independent, but available in Albanian only , and in English the Home page.
* Impact of downtime will be very minimal around 0.05%, considering the high reliability of the Google infrastructure.
* Its availability is measured by its performance when a subsystem fails, its ability to resume service in a state close to the state of the system at the time of the original failure, and its ability to perform other service-affecting tasks (such as software upgrade or configuration changes) is a manner that eliminates or minimizes down time.

### *Manageability/Maintainability*

#### Monitoring

The system will be subject to periodic evaluation by our Tester and our Supervisor and This evaluation will be performed by measuring the error logs generated by our product and sent to use automatically.

For us to be able to fix those errors , we shall be able to make some procedures with prompts and validations of our data .

#### Maintenance

We are going to design in a Modular way our product so the administrator or we can perform and evident operations for maintenance . So the system is isolatet and easy maintained.

#### Operations

* Reporting in real time.
* Sharing info restrictively.
* Backup and recovery information.
* Assigning tasks

### System Interface/Integration

Specify the use of other required products (e.g., a database or operating system), and interfaces with other systems (e.g., UWHires package interfaces with PubCookie and ODS, HEPPS system interfaces with Budget system). For each interface, define the interface in terms of message format and content. For well-documented interfaces, simply provide a reference to the documentation.

Outline each interface between the product and the hardware or network components of the system. This includes configuration characteristics (e.g., number of ports, instruction sets), what devices are to be supported, and protocols (e.g., signal handshake protocols).

#### Network and Hardware Interfaces

The app will use either Wi-Fi, or mobile data to connect to the internet. Other network related issues are automatically handled by Firebase Infrastructure, including connection monitoring, operation queueing during offline periods, etc.

#### Systems Interfaces

The users and doctors will be able to authenticate using the following methods, but the app uses a unique identifier which is not affected by the possibly different sign in method.

* classic username and password
* email and password
* Gmail account
* Facebook
* Twitter
* Phone number

The signing of the consent will be done electronically, complying with all legislative regulations, according the specified template by the clinic.

### *Security*

* Specify the factors that will protect the system from malicious or accidental access, modification, disclosure, destruction, or misuse. For example:
* encryption
* activity logging, historical data sets
* restrictions on intermodal communications
* data integrity checks
* Security provided to the access of user private information.
* Security provided to access of data.
* Restriction provided to access of data.
* There should also be provided a term of agreement to the user, in order to access their information and the data posted to the server.
* The android application should also include a list of modules and components that will be required to be accessed from the application, as known as permissions

### *Data Management*

The datas in this product are managed only by the administrator and they include user personal datas and their records in medical files.

We can use the " Cloud Firestore" products,in which differently from the Realtime Database, data is stored in Documents and Collections, thus offering better query support, and a offline first approach.

### *Standards Compliance*

Only Legislative Requirements:

* Accounting Requirements: Verification with Id card no.
* Safety/Security Requirements: System encrypted.

### *Portability*

* Use of Firebase products, which offer flexibility in different platforms, such as: Android, iOS, Node.js, Java, Python and GO
* Real time updates

## Domain Requirements

*Everything related to the domain that might be needed in the project shall be mentioned in here. Sometimes the domain Requirements might be thought as part of either functional or non-functional requirements.*