New York University Abu Dhabi
CS-UH 2012: Software Engineering

Ali Jafri
Maimuna Zaheer
Raiyan Reza
Shamma Alkhoori
Saoud Almansoori

Software Requirements Specification

< Qalb+ >

Version: 1 **Date: (10/14/2022)**

Table of Contents

5. Document Approvals

6. Supporting Information

1. Introduction

For the medical practitioners working in the United Arab Emirate's health sector, a web-based system for managing doctor appointments is to be engineered. With the help of this system, doctors, and the employees that assist them can effortlessly plan and manage the appointment calendar for their patients. This appointment scheduling technique is effective in facilitating doctors' productive work. Therefore, the platform enables users to easily schedule Zoom meetings with doctors, find nearby healthcare professionals, ascertain health insurance compatibility, and share medical data. This will also be put into practice by integrating the platform with already-available tools and services like Zoom, Google Maps, Uber, What's App, and others.

1.1 Purpose

The key objective is to provide a real-time overview of available time slots and online booking options to make scheduling doctor appointments more accessible. The Qalb+ system needs to respond to the wishes of patients and doctors.

Optional goals include:

- To make it possible for clients to see and modify their appointment times based on doctors availability, ratings, and suggestions. The system allows users to create new appointments, edit and delete existing appointments, and view a weekly schedule
- The system must make it simple to add, delete, and update schedules. It should also be simple to use and grasp.
- Providing patients with the option to prepay visit costs (primarily for those without health insurance but even for services not covered by insurance);
- To give patients a view of their medical history as it appears in the system from doctor and based on referral letters
- To make a calendar with reminders for actions related to an individual's health. For instance, make appointments for follow-up care, health certificate renewals, etc.

Currently, the main way to get information about open times for doctor's appointments is by phone, but Qalb+ will make it easier to arrange your schedule and identify good times because the data will be available on the web. Qalb+ portal would be created with both patients and medical professionals in mind (doctors and nurses). Individuals holding a license to practice medicine and provide treatment in Abu Dhabi are welcome to use the system. Thus, in principle our system can provide access to all physicians, nurses, counselors and other treatment providers who can practice in Abu Dhabi, whereas in reality not everyone will be on our system. Therefore, the system's user-friendly interface should result in a high level of usability.

1.2 Scope

The project's scope can be summarized in the following manner.

- Avoid losing patient and appointment information. The patient's medical history will also be included in the portal, along with information the doctor wants the patient to know about test results, vaccines, and prior visits to the doctor. Having this information online and accessible to patients would assist in alleviating the problem of doctors and nurses being overworked.
- A proper weekly schedule for the doctors needs to be prepared. It will make it possible for doctors to notify patients about vaccination deadlines and other visit dates. Both doctors and patients will benefit from the increased quality of healthcare and time savings.
- To offer a user-friendly interface for adding, moving, canceling, amending, and updating appointments, patients will be able to schedule or modify appointments with doctors online. Long call queue issues will be resolved, which is a significant problem for most.
- To assist patients in finding the nearest doctor, all patients should be able to use it because it will include currently practicing doctors in Abu Dhabi.
- Enabling physicians to post their profiles and promote their practices.
- The product will not be a client application, therefore, no mobile or desktop application. Our product will exclusively be hosted on the web server and accessed therein by web browser.
- The United Arab Emirates is the project's target country geographically, in particular Abu Dhabi ; if more demand comes, additional nations might be taken into consideration.

1.3 Definitions, Acronyms, and Abbreviations.

The following terms and acronyms are used in this document:

A Software Requirements Specification (SRS)

Qalb+: The name of the new patient/doctor portal

System: refers to the Qalb+ portal in this context

Appointment: a scheduled patient visit to a doctor's office that is made through the Qalb+ system. As soon as a patient books an appointment, it is recorded in the system and remains there until it is canceled or realized in a timely manner.

User : it refers to all portal users, including patients, doctors, nurses, and representatives. The administrators are not used here.

Doctor: an individual who is a medical professional

Treatment Provider: plays the roles of a doctor, nurse, psychologist e.t.c.

SQLite: The database, which will be used to permanently store all system data.

Python: The high-level, object-oriented programming language used to create the software

Django: a web framework which promotes quick development and simple, practical design. A method for creating web applications that emphasizes user interaction with the system.

1.4 References

Django. (2022). Django documentation | Django. Docs.djangoproject.com. https://docs.djangoproject.com/en/4.1/

SQLite. (2021). About SQLite. Sqlite.org. https://sqlite.org/about.html

Included Health. (2022). About Us. Doctor on Demand. https://doctorondemand.com/about-us/

1.5 Overview

The following report includes a collection of both functional and non-functional requirements along with an overview of the system's expectations.

Part 2 contains a general description of the product. A brief summary of the product is provided in Part 2.1. The primary tasks that the software will carry out are described in Part 2.2. User characteristics and the connections between the main actors are covered in Part 2.3. Part 2.4 discusses time restrictions, and Part 2.5 discusses how some changes may impact the requirements gathering process and alter system expectations.

Part 3 provides a brief technical breakdown of the project. From specifying user interface to some comments on standards compliances.

By the end of the fall 2022 semester, ARMS will prepare a web-based platform to connect doctors and patients. All of the elements listed in the purpose, scope, and objective section of this document will be included in the application. On November 9, 2022, ARMS will deliver different modeling, and on December 5, 2022, it will deliver a construction and testing report. On December 9, 2022, the ARMS team will present an overview of the web system to the business and other audience members as a part of the deliverables.

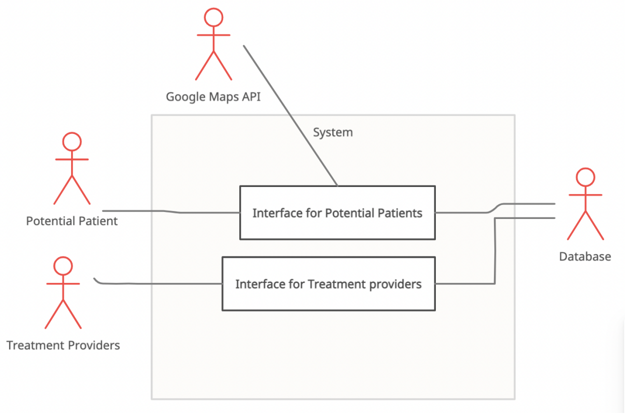
All the necessary framework activities needed— from communication, planning, construction, and deployment— will be done according to the AGILE framework. Agile promises flexibility and speed to complete demanding projects in a short notice.

The SRUM system that falls under the purview of AGILE in particular will be the organizational lynchpin holding together the different framework activities required to bring our plans to fruition.

2. The Overall Description

2.1 Product Perspective

The end product is intended to have certain integrations with other systems, such as Google Maps, Zoom, and Whatsapp. However, these integrations will only be one-way, in the sense that the system might utilize their services but not vice versa. There are similar systems to the one proposed, such as “Doctor on Demand”, which essentially places its focus on virtual healthcare. The system which is proposed will have a similar purpose, but mainly be focused on booking Zoom appointments with a physician, locate nearby treatment providers, determine health insurance compatibility, and sharing information with the treatment providers.


Google Maps API
System
Potential Patient
Interface for Potential Patients
Database
Interface for Treatment providers
Treatment Providers

2.1.1 System Interfaces

The first interface will be on the patient’s side, which will include a google maps API to show the treatment providers that will be closest to the patient.

2.1.2 Interfaces

There will be two different user interfaces in the system. One will be with the patients, and the other will be for the treatment providers.

On the patient’s interface, the patient will be able to find the closest doctor according to their current location, and also see which ones match with their insurance. Once matched, the patient will be able to contact the doctor through a means of their choice, and even share medical information with them.

On the treatment provider’s interface, they will be able to see a potential patient who wants to contact them, accept their request, view the patient’s details and contact them.

2.1.3 Hardware Interfaces

The system must be run on a device that supports internet connectivity, and can run web-browsing applications such as Google Chrome or Safari. The device can be a laptop, tablet or smartphone, which supports the previous web-browsing applications.

2.1.4 Software Interface

There are no customer requirements regarding interfaces with other systems. The main API that will be used is Google's Google Maps API, but this is a design choice.

2.1.5 Memory Constraints

Realistically, there are no explicit memory constraints, as the system will run on a browser that would already be installed on the user's devices (whether they be smartphones, tablets, or laptops).

2.1.6 Operations

There are two modes of operation, one is if the user is a patient, and the other is if the user is a treatment provider. Interactive operations include when the user inputs any data such as insurance details, medical information. Unattended operations are when the system processes the input and then stores it in the database.

2.2 Product Functions

The system will be an online web-based application. The application will have two different types of users: patients and treatment providers. Patients will be defined as people seeking medical services whereas treatment providers are defined as people licensed under a jurisdiction to provide medical services. On the server side, there will be a database that hosts some necessary information such as the following:

- Doctors, physicians, psychologists, and other people affiliated with health care facilities who are licensed to provide treatment.
- Health insurance and their compatibility with different treatment providers. For example, if GeoBlue health insurance is accepted by a physician affiliated with Cleveland Clinic in Abu Dhabi.
- Patient reviews detailing and rating their experiences with different treatment providers they had sessions with.
- The database has to be integrated with other preexisting services. Thus, once a patient creates an individual account they should be able to do the following:
- Locate the nearest treatment provider. Therefore, integration with Google Maps or a similar application is necessary.
- Schedule online appointments through means of their choice: Zoom, Skype, What'sApp, phone call etc.
- Users by uploading information of their health insurance plan should be able to determine if a given treatment provider can accept their plan.
- Provide safe and secure means to upload medical documentation that they would be able to seamlessly share with others.
- On the other hand, licensed treatment providers can create their own account
- They will have access to the following functionalities: Upload their license and other necessary credentials needed to register as physicians, doctors, or a professional person who is permitted to provide treatment.
- View patient details when a patient requests to meet them.
- Share their working hours, the health centers they are affiliated with, and addresses of their chambers.
- Provide medical documentation when a patient recommends them.

2.3 User Characteristics

There are two different types of intended users. The first type is potential patients, and since they have to do a lot of interaction with the web application (such as trying to find a treatment provider, inputting information such as insurance e.t.c.) they should be able to navigate and use the services provided by the system. Hence this means that they should be proficient with using web services on either a smartphone, tablet, or laptop. The second type of users are treatment providers, who also need a similar level of technical expertise as the patients, and be able to navigate a web application. The users should be above 16 years of age, as for the patient's side they would either be the guardian of the potential patient or the potential patient themself and to be a treatment provider it is given that you need to be an adult.

2.4 Constraints

The constraints regarding the system at hand are mainly concerned with security and legalities. Due to the fact that the system will be involved in dealing with personal medical records and information, the privacy of such information should be respected. Hence each class of user should have access to information that is appropriate to their context and position. For example, without the consent of the patient, a treatment provider should not be able to access or share the health information of a certain patient. The system must also be in compliance with local healthcare rules and regulations. For example, some places have stringent laws protecting patient privacy, enforcing doctor-patient confidentiality, and other measures in place to prevent abuse of medical information such as the Health Insurance Portability and Accountability Act (HIPAA) in the United States of America. As the system is going to be based in the UAE, it must follow the laws and regulations that are present within the country, specifically within the emirate of Abu Dhabi.

2.5 Assumptions and Dependencies

One assumption is that users will be using a device which would enable them to utilize all features of the web-application. As the 'access', per se, to the application is not restricted to anyone, the team in charge of the system cannot make sure that every user that arrives on the website is using a device which is fit to run the system.
One factor that the system depends on is the integration with the Google Maps API, as this would be vital to show the patients the closest treatment providers available to them.

2.6 Apportioning of Requirements.

If time permits, we would like to have the following requirements:

1. A subscription feature for Treatment Providers.
2. Adverts for treatment providers
3. Connecting with Zoom API
4. A payment system within the application
5. A chatting system between patients and doctors within the application

3. Specific Requirements

Requirements:

1. Authentication system: It should include login/signup feature along with a process to change/regain access to details such as password when required.
2. Store all the information in a well-protected and easy to access database.
3. Authorisation access should be given to the Patient user to connect with the Treatment Provider and share/request details.
4. Different accesses to different users:
    a. 4.1 Patient user should be able to look up the address and insurance requirements
    b. Patient has access to all the treatment providers but not vice versa.
5. Authorisation access should be given to the Treatment Provider to connect with the Patient and share/request details
6. The app should be connected with Google Maps API to help with directions.
7. The app should have a good search feature to look up for Treatment provider
8. Patients should be able to search Treatment provider names and should have a search filter including features like distance, insurance coverage, availability, speciality, and opening times.

Desired Features:

1. Subscription model. Treatment providers must pay a subscription fee to stay on the platform.

Optional Features:

1. Treatment Providers can promote themselves on the platform through advertisement.
2. Chat feature between Treatment Providers and Patient users.
3. Ability to manage medical bills.

3.1 External Interfaces

3.1.1 User interfaces

a. Login (Webpage)
    i. This user interface lets users login in to their accounts on the system.
    ii. It accepts text based format for username and passwords provided by the user. There will be some restrictions on the characters and symbols that are valid for setting up a username or password. For example: all symbols must be part of the English alphabet.
    iii. There will be a character limit for username and password.
    iv. If a user has no account, Login interface will send them to a Sign-Up page.
    v. If a user types valid credentials, login lets a user enter their accounts page. Patients and Treatment Providers will have separate accounts upon logging in.

b. Sign-Up (Webpage)
    i. This will accept input for valid email ids from the users.
    ii. We should verify that the standard ‘@<email provider>’ is included.

iii. Sign-Up should differentiate between Patients and Treatment Provider,thus asking someone what type of users they will be in.
iv. This difference will influence the user interface a user will use when they go to Login: Patient Account or User Account.
v. Input will ask for a user's name in the format, Full Name, Last Name & Middle Name (optional). All characters must be valid alphabets in the English language.
vi. Inputs will include information regarding birthday in the dd/mm/yyyy format.
vii. Input will ask for a user's sex: male, female, or other.

c. Patient Account (Webpage)
  i. Create a profile based on input from the Sign-Up interface. This will include information on name, date of birth, and sex.
  ii. Accepts input from the user to upload their medical documents in the pdd or jpeg.
  iii. There will be a size limit to each individual document. No individual pdf file of jpeg image will be allowed to exceed 5 MB in size.
  iv. Give users the ability to change information for name, date of birth, sex, and password. Username, however, cannot be changed.
  v. Upload information regarding health insurance.
  vi. Have a space for data fields to state diagnosis, health conditions, and other general information related to health such as height, weight, etc.
  vii. The data fields above should be private unless a user gives permission to share it with a treatment provider.
  viii. A space to see booked appointments and location of booked appointments if the appointment is in-person. This will provide information in a format suitable to tell time, date, and location.

d. Treatment Provider Account (Webpage)
  i. Create a profile based on input from the Sign-Up interface. This will include information on name, date of birth, and sex.
  ii. Have a field where a Treatment Provider can upload their credentials.
  iii. This credential has to be verified and if and when verified this will be shown to a treatment provider. E.g: a green mark appears next to the credentials field.
  iv. A field to let Treatment Providers pay a monthly subscription fee to remain on the platform.
  v. Maintain two states: active and inactive. An account will be active if and only if a Treatment Provider has verified credentials and paid the monthly subscription fees.
  vi. An active account lets a Treatment Provider manage requests by patents.

e. Patient-Treatment Provider Connect (Web application):
  i. This connects a Patient to a Treatment Provider.
  ii. Provides two different modes depending on the user type accessing it.
  iii. If accessed from a Patient Account, it can take input of the user's health insurance information and location to locate nearby treatment providers.
  iv. If viewed from a Treatment Provider, they can see incoming requests from patients.
  v. Must be able to interface with Google API.

3.1.2 Hardware interfaces

For this project hardware interfaces do not apply. Our web application should be compatible with modern browsers and these browsers handle hardware interfaces and compatibility.

3.1.3 Software interfaces

a. Google API:
The Patient-Treatment Connect interface should be able to use Google API.

3.1.4 Communications interfaces

a. Web Browsers:
Our entire application should be fully compatible with modern browsers such Google Chrome, Firefox, Opera, etc. But for ease of development we will confine ourselves to Google Chrome.

3.2 Functions

Signup:

- Check if someone is signing up as a patient or a treatment provider.
- make sure the username is not taken before
- make sure the password is within the given criteria (i.e. more than 8 characters and includes at least one number and one special character)
- make sure the all the required inputs are entered else show error message
- make the email address input in the proper format. The "@<email provider>" must be there or else the system shall raise an error message.

Login:

- During login in: makes sure the user and password matches else show an error message "Incorrect password/username"
- In case of Incorrect password/ wrong inputs:
  - Upon resetting the password: The old password is replaced with a new one based on the password criteria.

-Give a prompt to sign-up if someone does not have an account.

Treatment Provider Search Bar (used by patients):

- Entered characters should match with names in database else no results found
- Limit of 30 characters should be made.
- Characters should be valid alpha-numeric characters.

Check Subscription Fees:

- Check if a treatment provider has paid their subscription fee. If they have, then keep their account active if it is still, turn their account from inactive to action, and if no subscription fee is paid, keep an inactive account inactive or turn an active account to inactive.

Upload Documents:

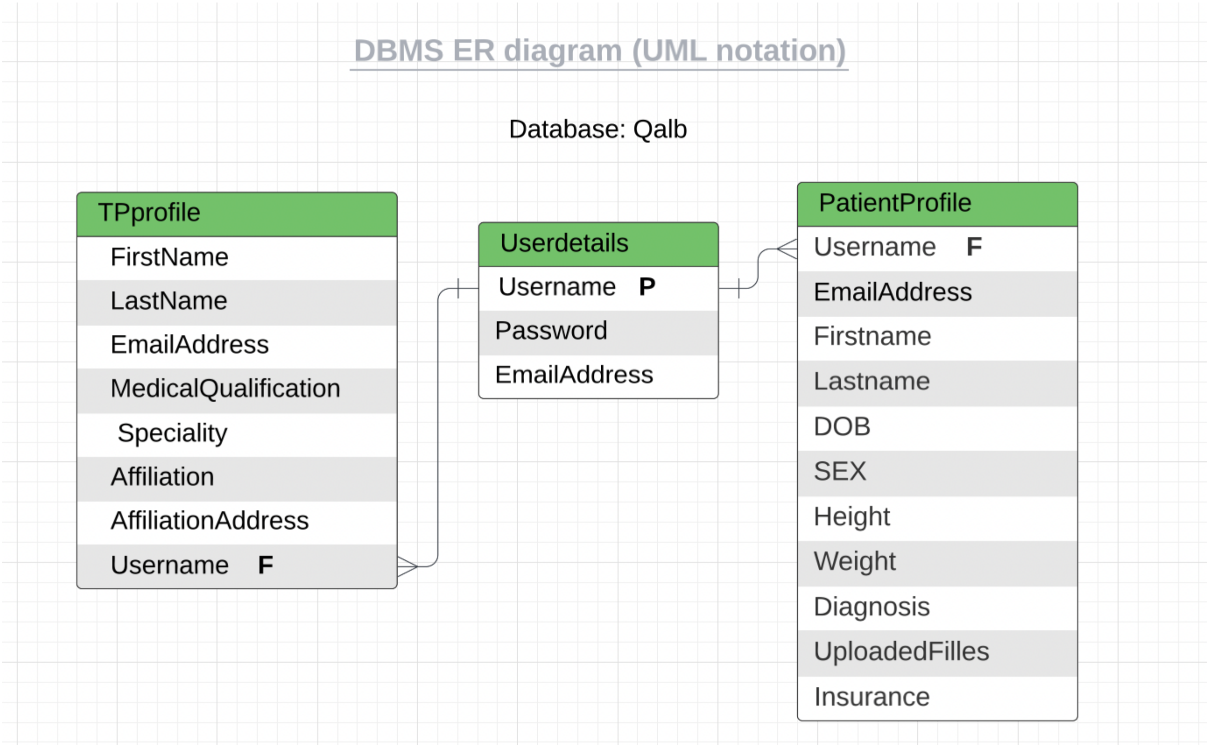
- A generic function that can be used to upload documents in pdf/jpeg format.
- Should have a specified file limit size.
- Must check the format of documents being uploaded is either in pdf or jpeg format.

3.3 Performance Requirements

- 100 simultaneous users to be supported.
- The ability to upload and download documents of around 5 MBs.
- Communication between the system and users should have acceptably latency.

3.4 Logical Database Requirements

Database Name: “qalb”
Tables: TPprofile, Patientprofile, userdetails



Every Time a user sign ups or logins into the system, the database is accessed. In cases of password changes, the database is updated. Upon addition of new Treatment Providers, the relevant table in the database is updated.

3.5 Design Constraints

The system doesn’t have any design constraints.

3.5.1 Standards Compliance

The system must comply with health rules and regulations of Abu Dhabi.

3.6 Software System Attributes

3.6.1 Security

1. Security is ensured by providing limited access to Treatment Providers so that they can access only the details/files shared or given acces by the Patients.
2. The Patient data is private unless a user gives permission to share it with a treatment provider.
3. Hashing Technique is used: Usernames and Passwords are hashed and stored in the database to ensure security.

3.6.2 Maintainability

1. Make sure the code/program follows the proper OOP guidelines
2. Proper comments are made throughout the code.

3.6.3 Portability

1. Since the program works on the web, we have to make sure that it works both for mobile and PC and has a responsive layout to support various devices.
2. The website should load fast and any delays should be prevented.

4. Change Management Process

Team consensus facilitates changes. These changes may be viewed at the updated SRS document that is kept in the git repository, projectARMS.

5. Document Approvals

Ali Jafri 14/10/22

Ajafri

Saoud Almansoori 05/12/2022

Syed Raiyan Nuri Reza: 14/10/22

Maimuna Zaheer 14/10/22

Shamma Alkhoori 14/10/22
ShammMK