ASHESI

ASHESI UNIVERSITY COLLEGE

AN ELECTRONIC HEALTH RECORD SYSTEM WITH INFORMATION SHARING AND DATA RE-USE CAPABILITIES FOR DIABETIC PATIENTS

APPLIED PROJECT

B.Sc. Management Information Systems

Audrey Marie Colombe Croley

2020

ASHESI UNIVERSITY

AN ELECTRONIC HEALTH RECORD SYSTEM WITH INFORMATION SHARING AND DATA RE-USE CAPABILITIES FOR DIABETIC PATIENTS

APPLIED PROJECT

Applied Project report submitted to the Department of Computer Science, Ashesi University, in partial fulfillment of the requirements for the award of Bachelor of Science degree in Management Information Systems.

Audrey Marie Colombe Croley

2020

DECLARATION

I hereby declare that this Applied Project is the result of my original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature:

………………………………………………………………………………………………

Candidate's Name:

………………………………………………………………………………………………

Date:

………………………………………………………………………………………………

ACKNOWLEDGEMENTS

I would like to thank God Almighty for his Guidance and for giving me the strength and intelligence to complete this project. I am also grateful to all people that assisted me, particularly my supervisor Stephane Nwolley for his Guidance, his productive criticism, and the overall support toward the completion of my project.

ABSTRACT

Diabetes is a significant cause of adult disability and death in Ghana. People living with diabetes experience complex psychosocial challenges, including psychological and emotional insecurities, limited social support, and inadvertent HIV/AIDS stigma. Management and self-care are weak, and healer-shopping between medical systems is standard. Significant limitations exist with diabetes care, including poor diabetes education, a lack of guidelines for diabetes care, erratic supply of essential diabetes drugs at health facilities, and poorly trained diabetes health care professionals, including doctors, nurses, and dieticians. We consider challenges for research, healthcare, and policy. This project focuses on how an electronic health record can facilitate healthcare tasks and, based on the findings, develop a technological platform to help people with diabetes in Ghana to get quick access to healthcare, and constant medical assistance while facilitating the work of healthcare actors. After interviewing 50 Ghanaians with diabetes, the significant findings were that people with diabetes in Ghana do not have access to convenient healthcare. The interviews also revealed that the healthcare routines, including regularly visiting the hospital, and checking what they eat, overwhelms the people with diabetes. This research showed that an electronic health record could allow self-management for the diabetics; it could be used as a medium for permanent medical consultation. Relying on my findings, this project developed a platform that will help people with diabetes to get medical assistance wherever they are and whenever they need it by keeping track of their medical record, reminding them of their daily routines and also allowing permanent communication with their medical doctors. The platform facilitates medical appointment booking and check-ups. The user of the platform is able to track their calories and sugar consumption. Looking at the long run, the platform will serve as an electronic medical record that will not only help people with diabetes but non-diabetics as well. Further updates will be made to adjust the platform to technological advancement.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1. Background and Motivation

Healthcare is one of the key drivers of human life; as a car needs servicing, humans also need to take care of their health. This is the reason why health information, as one of the key factors that contribute to strengthen healthcare delivery, must be managed for more efficiency. However, with people having access to their health records, it will create more awareness of their medical conditions. The potential of the electronic health record (EHR)system to transform medical care practice has been recognized over the past decades to enhance healthcare delivery and facilitate the decision-making process. Subsequently, EHR and other clinical decision support system tools are currently used in both primary and secondary healthcare facilities in most developed nations[1].

In 2010, the first policies on electronic Health in Ghana came into being, but the implementation was challenged by the commitment from the staff and users of the systems. Later on, with the upcoming of the National Health Insurance Scheme (NHIS), health institutions adopted the use of computers, and this favored the use of electronic health systems. Contributors, investors, and active members of the health center are putting measures in place for the effectiveness of electronic Health in Ghana. Prior research results indicate sufficient preparations made by the hospital before the system's introduction through thorough understudying of other implemented systems in other hospitals, the establishment of an Information Technology (IT) department, provision of infrastructure and network connections, and the training of staff[2].

In 2012, a study by Patrick and Laar proved that within this new healthcare system, citizens and patients are instead given more responsibility in the management of their Health and chronic diseases, leading to the gradual creation of a healthier nation. It was further pointed out that through e-health, patient records can be made shareable between both the patient and the doctor, compensating for the shortage of medical experts in the field [3].

E-Health (electronic Health) makes it possible to generate, capture, transmit, store, and retrieve digital data for clinical, educational, and administrative purposes. With an electronic health record, such data can be shared among health professionals and even patients who might not be found in the same geographical location. Communication is bi-directional, as users have the opportunity to monitor their Health at home and store their health information to the e-health provider's database as well as being able to access health information from the e-health provider [6].

The benefit of an electronic health record (EHR) is that it allows the medical records of patients to be viewed by physicians, doctors, pharmacists, and actors in the health center. Most importantly, it reduces the cost and the time spent by patients when visiting the hospital. As healthcare can be provided remotely, it provides self-care, which is very helpful in the case of chronic diseases such as diabetes. The adoption of EHR can help to better the Ghanaian healthcare system.

Upon the study of the Ghanaian health sector, one major challenge of the sectors was the geographical position of the health institutions. Several districts have multiple hospitals, whereas other districts have none. Shortages exist in terms of both lower-level health facilities as well as equipment at the subdistrict level [9].This situation creates unequal access to healthcare services around the country. Recruitment of health workers (nurses, physicians, medical doctors), especially physicians, remains a challenge, although the present situation represents a reversal of an earlier emigration trend [9]. In light of this problem, the implementation of EHRs will reduce the number of patients per health worker since patients, especially diabetics, will be able to self-manage their Health and access some medical care from their smartphones.

Furthermore, the case of diabetics is so delicate that they are in constant need of medical healthcare and check-ups. The advancement of technologies favored them with tools like test strips, lancets, and a lancet device to check patients' blood sugar levels. However, the problems come when they have to interpret the results; they need to refer to the health professional, which brings them to visit hospitals more often. With an electronic health records system, they will be able to interpret and receive their medical consultation and keep electronic journals on their smartphones.

As attendance in the health facilities increases with time, the volume of medical records becomes a big challenge to health facility management. This is no different in Ghana. The medical record of a patient is the clinical representation of the patient that is built over some time by various clinicians with the consent, trust, privacy, and confidence of the patient. It enables continuity of care, and again, over time, it becomes a comprehensive, clinical database from which various and salient clinical information is gathered through research [10].

This project seeks to solve the problem that Ghanaian health institutions have by creating a mobile that will help people with diabetes to track their sugars, get access to consultations on their phones for the technologically inclined among them. Doctors, physicians, and pharmacies can also use the mobile application. These features were created, bearing in mind that the number of patients in Ghanaian hospitals is excessive for the health workers. According to a Jumia Mobile Ghana report in 2016, given this technological advancement affecting the socio-economic state of Ghana; the buying behaviors of Ghanaians in mobile phones have also changed. In this era, the Ghanaian consumer seeks for the preference of internet browsing, social media integration, and mobile app applications as a defining feature in purchasing mobile phones [12].

This application will allow patients to enter their daily medicals constantly, which will be analyzed by a medical doctor, and the patients will get an alarm when something is abnormal. This will allow patients to self-manage their Health while being monitored by their medical doctor. Additionally, patients and doctors can schedule an interview if necessary.

The application will keep the patient's record, which can be retrieved and referred to at any moment of need. It will create a familiarity between doctors and patients and allow effective delivery of medical care. This will reduce the number of people with diabetes who visit hospitals daily and allow people with diabetes to save more time to do other things.

Diabetes imposes a significant economic burden on the global healthcare system and the broader global economy. This burden can be measured through direct medical costs, indirect costs associated with productivity loss, premature mortality, and the negative impact of diabetes on nations' gross domestic product (GDP). Also, diabetes is recognized as an essential cause of untimely death and disability. GDP worldwide from 2011 to 2030, including both the direct and indirect costs of diabetes, will total US$ 1.7 trillion, comprising US$ 900 billion for high-income countries and US$ 800 billion for low- and middle-income countries [8]. This finding can justify the fact that some middle-income and low-income countries remain at that level due to some factors such as diseases and epidemics.

2. Related works

Chiron Health is a US telemedicine platform that allows physicians to connect with their patients for routine follow-up appointments over a secure video conference. The app seamlessly integrates with existing practice workflows (e.g. Scheduling) and management systems (e.g. EHR / EMR, RIS, LIS). Doctors can offer patients remote care, and replace unpaid phone calls with reimbursable video visits, improving outcomes by 3%-4%. This platform facilitates communication between patients and doctors; such patients and doctors can have a conversation at the comfort of their homes. this system allows only appointment booking but omits the integration of a way of making sure that the patient keeps track of his health records. And also, the patients can use this software to book appointments and get video consultations in the following cases: routine follow-ups, general health questions, cold, flu, allergy, cough, fever concerns, discussion of lab results, medication adjustment needs, prescription refill requests, and more

Ii) **MySugr**

MySugr is app-based, all-around care for people with diabetes made by people with diabetes. The application, services, and diabetes coaching all work together to ease the life with diabetes. With the app, a person with diabetes can record his meals, and blood sugar; the app also estimates the hba1c. Persons with diabetes can also get their monthly record printed in a pdf format and get a detailed analysis of documents. However, such systems are not readily applicable to Ghana because of the variety of gastronomy and many other factors.

Iii) **Dr diabetes**

Dr diabetes was the first e-health application built for people with diabetes in Ghana. It was developed by Raindorf Owusu, the CEO of Oasiswebsoft when he was a student of Methodist University College in 2011. This app asks the users some questions to determine their tendency to have diabetes and advise them on healthy choices to make. Moreover, users will be able to interact online with medical practitioners. Patients who fear to go to the hospital because, to them, it carries a stigma, or those who want to avoid long lines, will be able to consult doctors privately through the app [13]. This app was limited since it did not allow people with diabetes to know their sugar level, which mostly determines their health level. The app instead suggests regular visits of users to the hospital. This situation promotes massive crowd in hospitals, another issue, since the number of health workers does not satisfy the need for healthcare.

Iv) Afyapap

Afyapap is an application developed in the United Kingdom which helps the Kenyan population to manage diabetes and hypertension. The app is designed to provide a variety of tips that are essential for a healthy lifestyle, including diet, physical activity, sexual Health, and mental wellness. It is also intended to track vital statistics like blood glucose [14]. This application allows users to check their vitals but will promote self-medication. The missing features are the involvement of medical doctors and other health workers.

3. Project Summary

This project identifies the lack of mobile assistance for individuals with some medical conditions, especially diabetics. It utilizes already established algorithms and techniques to develop a new one that is better suited to the Ghanaian community. It will combine the features of the different apps stated earlier to make options available for people, especially diabetics. It focuses mainly on assisting people with diabetes, as well as people who experience special medical conditions and need assistance. The primary goal of this project is to deliver a web-based application called SugarDose. This application:

1. Assists people, especially diabetics, with their healthcare;
2. Keeps track of the human vitals and alerts users and doctors in charge of the user when vitals are abnormal;
3. Checks the calorie and sugar consumption

This standalone platform will help people with diabetes to be live healthy lives so that they can live happily and be at the peak of their performance.

CHAPTER 2: REQUIREMENTS

2.1 Overview

This chapter focuses on the requirements of the system based on the international standard for software requirements. Using the agile method because of its flexibility for change, the different steps involved in the required engineering process were: first, identifying the users and stakeholders, and then, afterward, discovering the requirements and analyzing them. Moreover, this chapter touches on the application's functional and non-functional requirements.

2.2 Application users

The application will help people with diabetes to get their diet, and health status recorded and monitored by doctors. Therefore, the primary user types are patients, that is, diabetics, and health workers, which encompasses doctors, nurses, and pharmacists.

2.3 Requirement gathering and analysis

2.3.1 Requirement gathering

The requirement of the application was gathered by interviewing the various identified users. The purpose of the interviews was to gather more information about the routine of doctors with diabetics and the needs of diabetics. The following questions were asked to get the information needed:

1. How often do people with diabetes visit the hospital?
2. What are the most common reasons why people with diabetes visit the hospital?
3. How can health workers be helped to perform efficiently and save more lives?
4. What is the primary need of people with diabetes?
5. What is the user's level of education and technological competence?
6. What interface and level of technicality are desirable?

7. What are the ways and means software designers will adopt to test the usability of the software?

2.3.2 Requirement analysis

The interviews reveal that different classes of users have different priorities and needs.

2.3.2.1 Diabetics' perspectives

People with diabetes estimated that they visit the hospital a least once a week and pharmacies very often. They also mentioned that they are more concerned about the food they eat and the implication of the sugar level in their systems. Most of the people with diabetes who were interviewed had the basic knowledge of technology and were familiar with other apps, such as Facebook and WhatsApp. However, they insisted that for an application to be more useable, the application should be easy to manipulate. People with diabetes should be able to get their information secured and given feedback.

2.3.2.2 Health workers' perspectives

Doctors and nurses want an application that will reduce the number of diabetics that will visit the hospital just do some basic things such as constant checking, and interpretation. They want an application that will organize their daily schedules. Nurses and doctors are mostly university graduates. They are familiar with new technologies. The applications should provide a way for them to be in touch with people with diabetes. They should be able to give feedback on the application.

2.4 Purpose of the software

The application for people with diabetes, which was developed through the diabetics' "SugarDose" project is primarily intended to run on multiple mobile devices as well as on web platforms. The client-side of the application handles a wide array of activities. These activities range from allowing people with diabetes to control their Health to getting their Health supervised by a doctor.

2.5 Product Features

The product has the following features available within the application; it allows for:

1. Medical appointment booking
2. Record of vitals and sugar level
3. Message communication within the platform

All the users should be able to visualize previous vitals and sugar level records and check their improvements; these records will also be accessible to the doctor.

2.6 System features

The system has two categories of features, the additional features and the core one.

2.6.1 Core features

2.6.1.1.Health checking

This feature allows users to enter their vitals and allow their doctors to monitor their Health:

1. User (diabetic) signs in
2. User’s doctor signs in’
3. The user chooses an emoji describing how he is feeling
4. The user enters his vitals for the day.
5. The vitals are sent to doctor
6. Doctor checks the records
7. A doctor can message the user at any time.

2.6.1.2 Medication intake checks

This feature controls the daily intake of medication by users and reminds them of when to take their medicines. It also helps the doctor to monitor and know what his patients receive.

1. The user enters the name of all medicine at his disposal.

2. Users or doctors enter the prescription for each medicine.

3. The user sets a reminder for medicine.

4. Daily reports will be generated for the intake of medicine.

5. Doctors can view their patient's daily intake of medication and suggest a cure for impromptu sickness.

2.6.1.3 Diet and sugar level balancing

This feature allows the user to check his daily calorie consumption and sugar and know how to regulate his sugar level.

2.6.2 Additional features

2.6.2.1 Appointment and examination booking

For the hospital, this system creates a platform that allows patients to book appointments and book for examination, scanning, and other laboratory work.

2.6.2.2 Patient demographics

This feature of the application is fully customizable. It allows for tracking the diabetic’s demographics, such as their primary information (age, name, sex, identification). The patient's marital status, insurance coverage, and previous medical history are all available on the platform

2.7 Use case scenario

The use case diagram displays the different interactions between pharmacists, doctors, and diabetics.

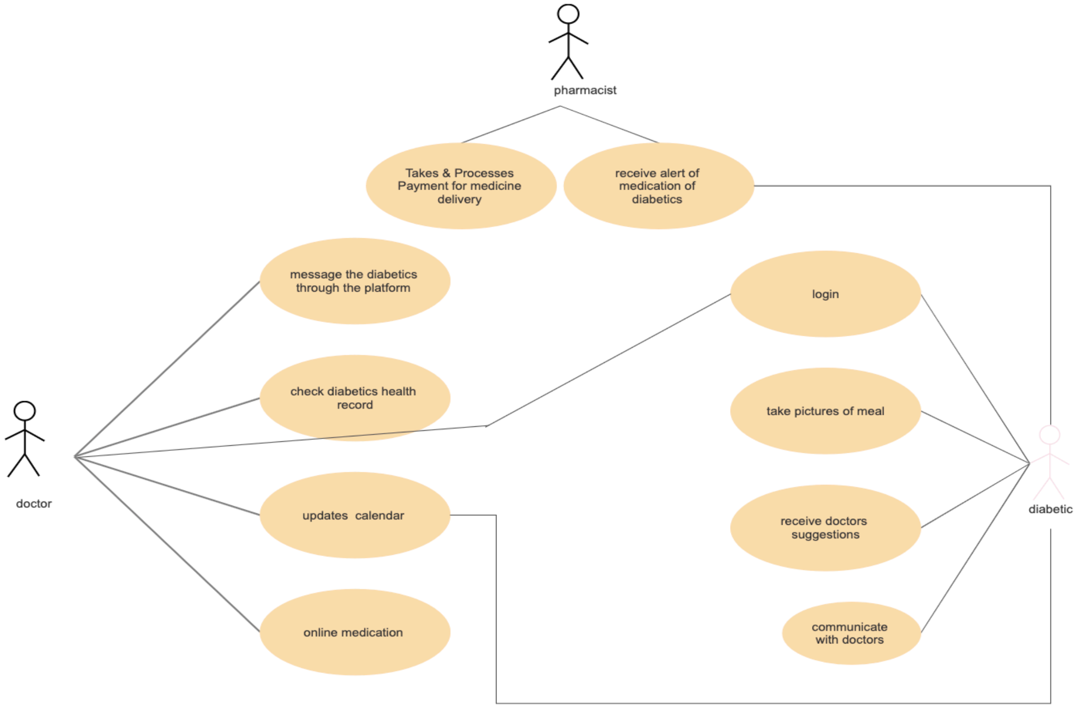

pharmacist
Takes & Processes Payment for medicine delivery
receive alert of medication of diabetics
message the diabetics through the platform
login
check diabetics health record
take pictures of meal
doctor
updates calendar
receive doctors suggestions
diabetic
online medication
communicate with doctors

Figure 1: Use case diagram

2.7.1 The login processes

To login to the application, patients (diabetics) and doctors will use their password and email. The users should have registered first after their first download of the app. The registration process requires the users to enter their surnames, other name(s), phone number, email, and password. Once they register successfully, an SMS or an email will prompt them to log in. All their registration information is saved in the database, and when they login in, the email and password submitted will be compared to the one in the database. The user will be directed to the dashboard only if the email and the password match. Once granted access to the panel, the user can perform any task and features that the application offers.

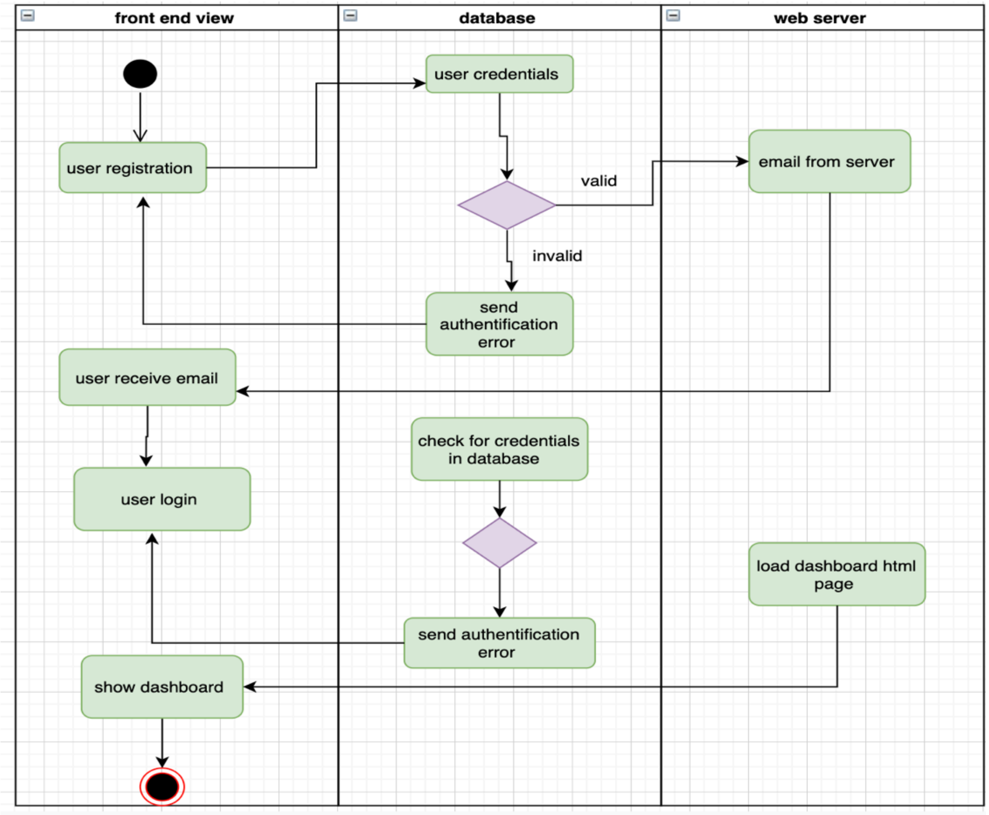


Figure 2:The diagram of the login activity process

2.7.2 The process of scheduling appointment

The application allows users to schedule an appointment. The process of scheduling appointments differs per user. For the patients (diabetics), after they log in, they will be redirected to the dashboard where they will have to select the button with the option 'schedule an appointment'. The user will be redirected to the doctor's calendar, where they will able to see the doctor’s free consultation slot. They will then select the time and date of the appointment and wait for the doctor to confirm through SMS or phone call.

On the doctor's side, after a successful login, they are redirected to a page with their patient's list. With the button ‘calendar’ at the top of the page, the doctors will be able to view their schedule. On the calendar page, they have the option to update, delete, confirm, or reject the appointment.

Afterward, they can save the changes made. The doctors finalize the appointment scheduling by sending an SMS or calling the patient to confirm the date.

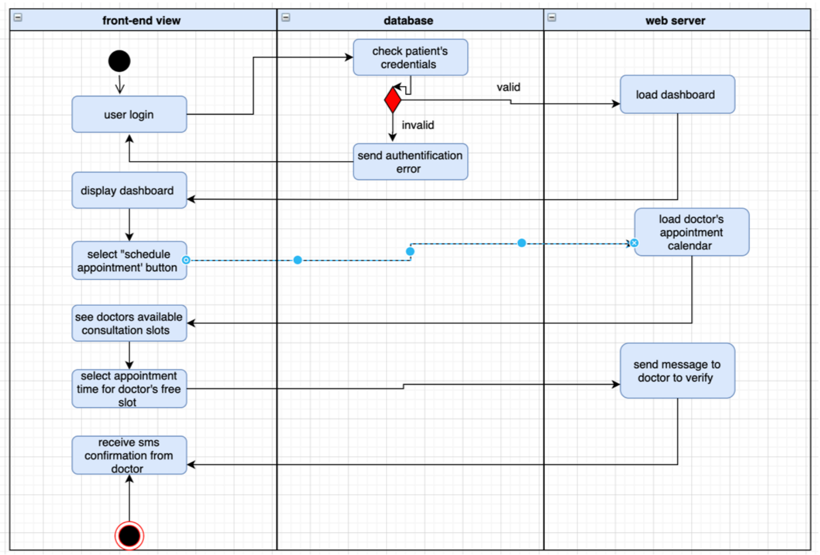

front-end view
database
web server
check patient's credentials
valid
load dashboard
user login
invalid
send authentification error
display dashboard
load doctor's appointment calendar
select "schedule appointment' button
see doctors available consultation slots
send message to doctor to verify
select appointment time for doctor's free slot
receive sms confirmation from doctor

Figure 3 : Activity diagram of scheduling an appointment process from the patient's side.

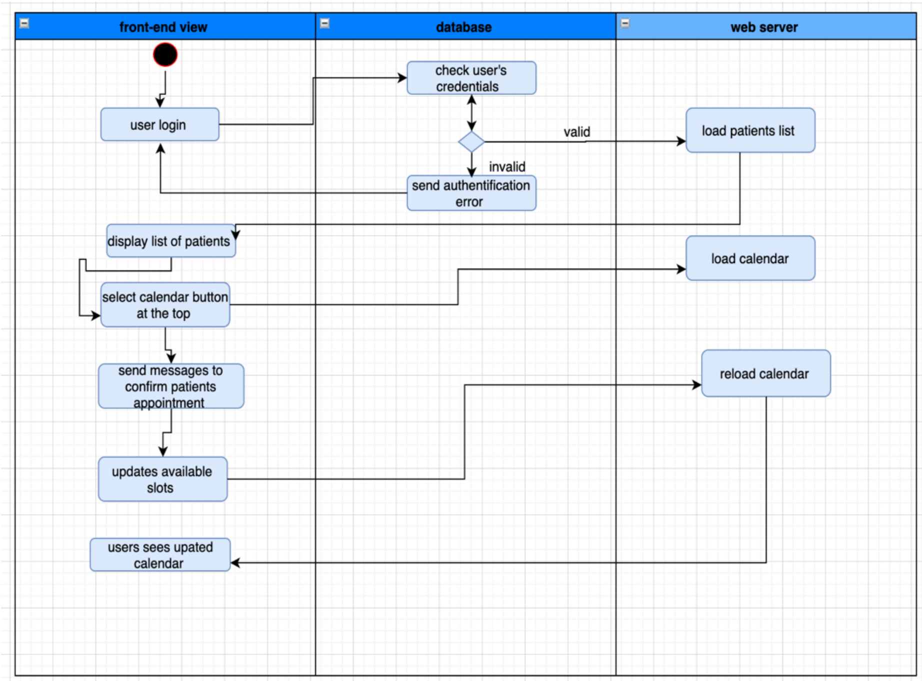


Figure 4:Activity diagram of scheduling an appointment process from the doctor's side.

2.8 Data requirement

2.8.1 Data requirement of user 1: patients

The patients interact with the system in several ways; the tables below show the data requirements of the different interactions of diabetic patients with the system 'SugarDose '.

Table 1: Patient's user requirement during registration

Type	This is the step required to register on the system of SugarDose as a user
Input	Type of user, Name, Surname, Email, password
Source	A filled online form by the user
Output	Send an email to the user to confirm and use the app when all criteria of registration are met, and all the fields are filled with the correct format. In case of an invalid record, the user will be asked to fill the form again.
Initiate	User clicks on the register button
Pre-conditions	None
Post-conditions	The user's information will be added to the Database.

Table 2 : Patient's user requirement during login

Type	This is the step required to log into the application
Input	User's email and password
Source	A filled online form by the user
Output	Directs users to the dashboard page when the user's credentials match the one in the database. When fields are empty, the user

	will be asked to enter correct credentials until it matches the one in the database. Form submission can achieve success or failure.
Initiate	User clicks on the login button
Pre-conditions	User must be a registered user of SugarDose
Post-conditions	The user's information will be redirected to the dashboard, and a session will be opened for them.

Table 3 : Patient’s user requirement during appointment booking

Type	This step allows the user to book a medical appointment.
Input	Date and time of the appointment
Source	The doctor’s calendar
Output	View an empty slot on the doctor's schedule and book an appointment. The user will receive a confirmation SMS from the doctor.
Initiate	User clicks on the ‘schedule medical appointment ’ button
Pre-conditions	User must log in to the application, SugarDose
Post-conditions	Reminder when the date and time of the appointment will approach

2.8.2 Data requirement of user 2: doctors

Table 4 : Doctor’s user requirement during registration

Type	This is the step required to register on the system of SugarDose as a user
Input	Type of user, Name, Surname, Email, Password
Source	A filled online form by the user
Output	Send an email to the user to confirm and use the app when all criteria of registration are met, and all the fields are filled with the correct format. In case of an invalid record, the user will be asked to fill the form again.
Initiate	User clicks on the register button
Pre-conditions	None
Post-conditions	The user's information will be added to the Database.

Table 5: Doctor's user requirement during login

Type	This is the step required to log into the application
Input	User's email and password
Source	A filled online form by the user
Output	Direct users to the dashboard page when the user's credentials match the one in the database. When fields are empty, the user will be asked to enter correct credentials until it matches the one in the database. Form submission can achieve success or failure.
Initiate	User clicks on the login button
Pre-conditions	User must be a registered user of SugarDose

Post-conditions	The user's information will be redirected to the dashboard page with the list of all his/her patients, and a session will be opened for the user.

Table 6 : doctor's data requirement during appointment scheduling

Type	This is the step required to confirm the medical appointment.
Input	User's free and busy dates, appointment confirmation.
Source	A calendar with a different arrangement A form of the patient's request for an appointment
Output	The user will confirm or reject the nomination. user will see his schedule
Initiate	The user clicks on the 'calendar' button.
Pre-conditions	User must be a registered user of SugarDose.
Post-conditions	The user will able to plan his consultations.

2.9 Non-functional requirements

2.9.1 Security

Since the users will be using the system on a personal device, the security will be at the level of their sign-up details and login features such as their email address, their phone number, and location. These security requirements will be in the form of encryption of user passwords and sites with a combination of MD5, VPN, and PHP session verification processes. The application allows us to add and remove the user's permission. Active directory support will be implemented; the database connection will be encrypted. Moreover, the application will be remotely accessible from any modern web browser with a suitable security certificate installed.

2.9.2 Availability

With reliable internet, the user should able to access all aspects of the application. The application should be available 24/7.

2.9.3 Usability

The users should be able to easily navigate and use the application to obtain the desired results. The user interface should be interactive and make use of icons that the users can easily recognize and understand. The design takes into consideration color-blindness and other standards.

2.9.4 Reliability

The users should be able to perform tasks without errors. They should also be able to trust the feedback acquired from this application to implement lifestyle and health decisions.

2.10 Systems requirements

System requirements are intended to communicate the functions that the system should provide [15]. To achieve the core features of the system, the system requires a smartphone, tablet, or laptop screen for the graphics; the application server uses Apache and MySQL for the database. Web socket was used the chat feature. Finally, the browser required is Safari, Chrome, Opera, or Firefox.

2.11 Summary of requirement analysis

In this chapter, the requirements specifications have been discussed. This contained an overview of the user requirements and system requirements for implementing the system. This requirement analysis was initially made based on assumptions and research, and later on, questionnaires were administered, and interviews were conducted to add more features and requirements to the system. The users can get help from their doctor, check their vitals, and have emergency features that allow them to get medical assistance everywhere at anything.

CHAPTER 3: ARCHITECTURE

3.1 Chapter overview

This chapter presents the design and the high-level architecture that was used for the implementation of the system and the components of the overall organization of the system. The design explored in this chapter is the database design, and the system architecture that the researcher opted for is a three-tier client-server architecture. 3-tier architectures provide many benefits for production and development environments by modularizing the user interface, business logic, and data storage layers [16]. The client-side server has a client-side, a database server and an application server. The application server is what we are more concerned about.

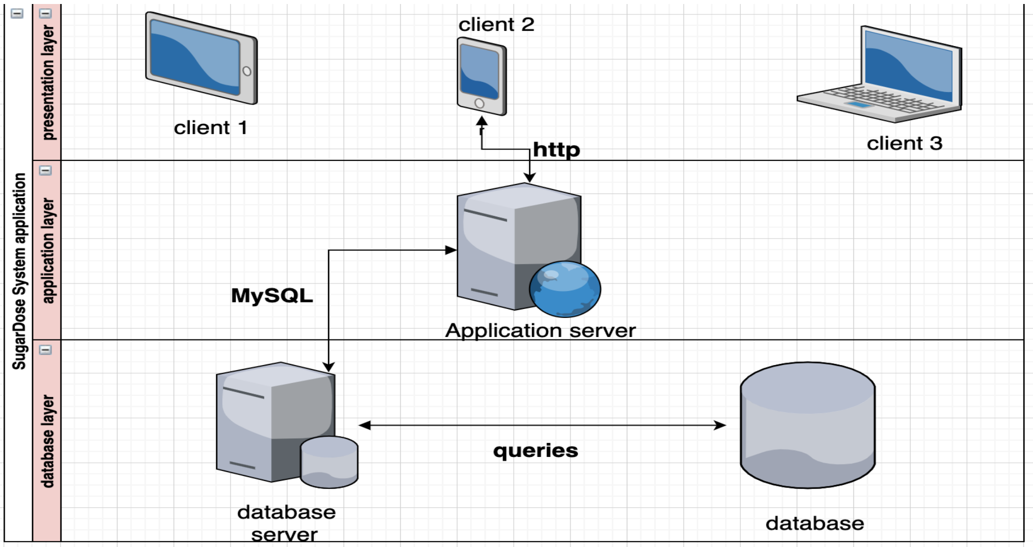


Figure 5: SugarDose's architecture

3.2 The client-server architecture

3.2.1 Presentation layer

The presentation layer focuses on what the client sees, the user interface. It responsible for the front view of the application, also referred to as front-end design. The presentation presents the application features to the user on its browser. The presentation layer makes the interaction possible between the end-user and the application. In the case of the application SugarDose, the presentation layer is responsible for the data entry, communication, and response to the tasks. The elaboration of the structure of the presentation layer was done by analyzing the user interface, navigation, and interaction.

Concerning the user interface, it refers to the layout of the page, such as layout, button, and positions that allow users to interact with the application easily. An excellent user interface must be consistent. With SugarDose, the user interface focuses on consistency by linking all the pages from both patients' and doctors' side. The design of forms and other components was consistently executed with the same model, colors, placeholders, and modal. This consistency in design will enhance the input collection strategy of the application.

Moreover, the form should be used for users. Users should be able to perform the intended task without any obstacle. The health domain has a lot of theorist terms; therefore, the slightest mistake can have a lot of consequences. SugarDose was built in such a way that the users can reverse their actions because a return button has been inserted on each page. Also, at the top left corner of each page, there is a button that will redirect the user to the dashboard. Communication was one critical feature of SugarDose; all users needed to communicate for an effective result. Therefore, the presentation layer is responsible for encrypting and decrypting messages sent or received. This process was done efficiently in such a way that messages sent can be received instantly, and the users will receive a sent receipt.

3.2.2 Application layer

The application layer defines the commands, responses, data types, and status reporting supported by the protocol [17]. The application layer from the application server binds and processes data frames, IP addresses, and sessions considered as an application. In the context of this project, the application layer interventions were needed to retrieve and display data from the database. Data were sent to the database using the application layer.

3.2.3 Data layer

The data layer is responsible for the processing of data. It deals with importing and exporting the required information needed to access the database to service user requests. SugarDose database includes six tables, namely: registration, patients, doctors, meals, report, and appointments.

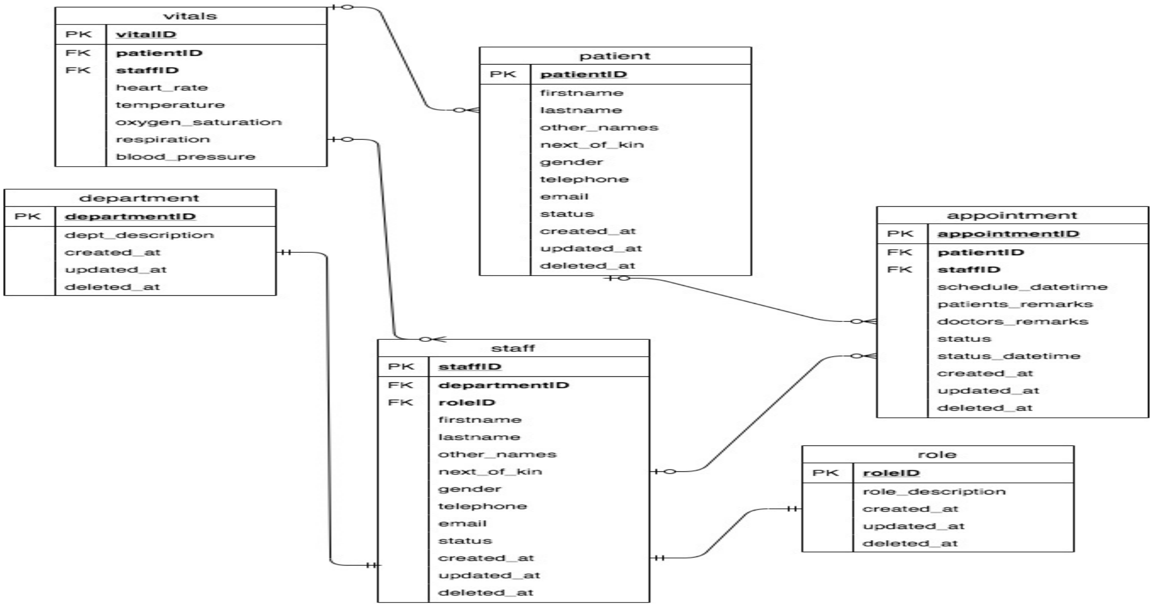
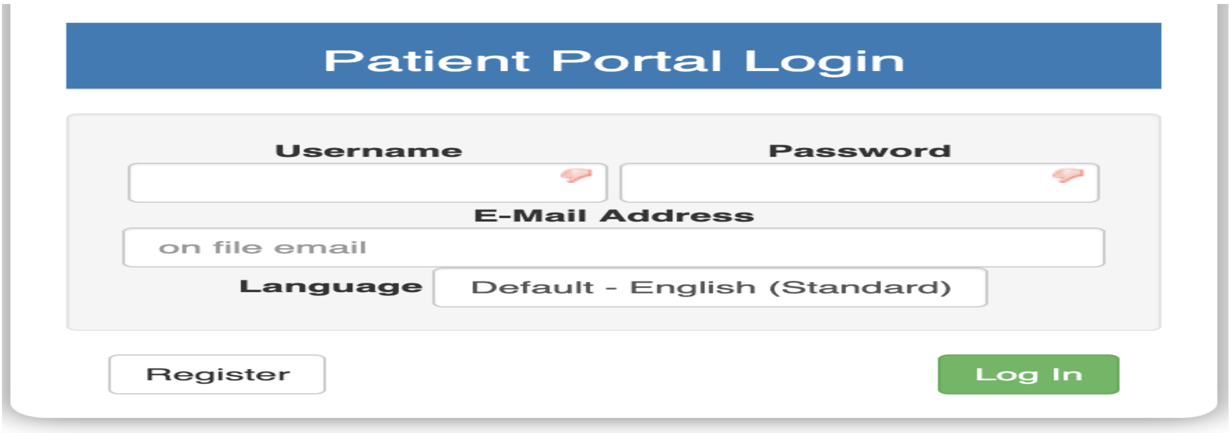


Figure 6 : Entity Relation Diagram of SugarDose

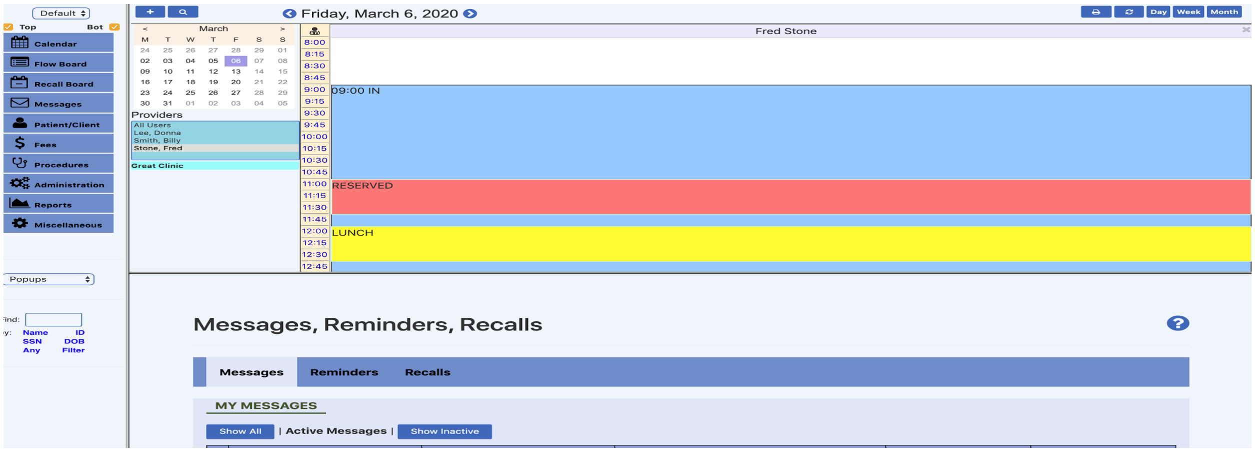
3.3 Prototype

This section shows the prototypes of the selected application. Figure 3.3 below shows the page where the API was able to calculate the number of calories in the food by just an upload of the meal picture. The picture can be uploaded by taking an instant snap or by going through your picture's file and choosing a picture of your meal. Moreover, the analysis of food data allowed us to generate the Graph in appendix 1 and back up with research; it demonstrated that sugar and the number of calories are related; the higher the number of calories, the higher the amount of sugar. Ultimately, these findings indicate that there may be limited availability of lower free sugar alternatives that have relatively lower calorie contents in many food and beverage subcategories [4].



Language: Default - English (Standard)

Figure 7: Prototype of the patient's login page



Friday, March 6, 2020

Fred Stone

09:00 IN

11:00 RESERVED

12:00 LUNCH

Messages, Reminders, Recalls

Figure 8: Prototype of doctor's page after login

3.4 Technology used

3.4.1 The integrated development environments (IDE)

The IDE used to create, edit, modify, and debug the code was visual studio code. Visual Studio Code is a lightweight but powerful source code editor that runs on your desktop and is available for

Windows, macOS, and Linux. It comes with a built-in support for JavaScript, TypeScript, and Node.js. It has a vibrant ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, Go) and runtimes (such as .NET and Unity) [18]. The choice of that IDE was influenced by some of its features such as the IntelliSense, a feature that allows highlighting and autocompleting of the code, the git command built-in, and the extensibility and customizability that would enable it to deploy and code with ease and confidence.

3.4.2 Programming languages

PHP: This server-side scripting language was used to develop the back-end of our web application. PHP stands for hypertext processor, and it is executed on the server, and the result is displayed on a web page as HTML. The use of PHP in this project is required for the database and other server-side interactions, and it stood out of other server-side scripting languages because of its fastness, simplicity, and its broad range of professional features. PHP is a language for the computer. It is a scripting language that programmers use to make computer communication if you are an online marketer or someone who has any need to use computers in greater detail. It is a language that is very useful and helpful to learn [11].

HTML: Hypertext Markup Language was used at the front end. In this application, HTML was used to manage the user interactions, create the web pages, and allow the synchronization between the various web pages. One of HTML's key advantages is its ease of use. HTML provides only a straightforward code structure, consisting of relatively few tags and attribute names; therefore, you can easily create HTML documents and not get bogged down with complex code or overwhelming tag choices. Hundreds, if not thousands, of inexpensive or free tools exist to help publish HTML documents from scratch or to export records from word processing programs to

HTML form [7]. For these previous reasons, HTML was the language chosen to implement the front end development of this project.

JavaScript: JavaScript is an object-oriented programming language; it is used to monitor the interactivity of web pages. In this project, JavaScript helped in the Function of different buttons on the various pages , which helped with the form validations and other features that have to do with the web application's interactivity.

SQL: SQL stands for structured query language; in the scope of this project, this tool helped to manipulate the database, as in creating, modifying, and managing the database.

3.4.3 Development platforms

This project made use of XAMPP, a cross-platform web server for Apache, MySQL, PHP, and Perl. XAMPP is used for development testing by creating a local HTTP server. HTTP stands for HyperText Transfer Protocol.

3.4.4 APIs

Web socket API was used to favorize the communication between the different categories of users within the web application.

CHAPTER 4: IMPLEMENTATION

4.1 Chapter overview

SugarDose is a web-based application that helps people with diabetes to keep in touch with their medical care professionals and to track the sugar in their daily meals. This chapter focuses on unfolding the implementation of the application, and outlines the techniques used in developing this application. It highlights the tools and technologies used in deploying the software package.

4.2 Approach and Implementation Limitations and Techniques

4.2.1 Approach

SugarDose is a web-based application; this means that with a device with a browser and functional internet connectivity, the application should be accessible. The type of application was decided after considering the cost of development, updates and customization, and platform independence.

Compared to a different kind of application, a web-based application is very flexible; it is implemented for a single operating system and can be accessed and tested on different browsers such as Firefox, Google Chrome, etc. Also, given a stable internet environment, the cost-effectiveness of the web-based application is advantageous compared to that of other forms. Moreover, a web application can be easily customizable. The web-based application gives room for changes that allow the developer to suit different users' needs. The application users will have different levels of expertise and knowledge of technology; therefore, a flexible and customizable application will help to bridge the gap. Thus, the web-based applications’ option was most favorable compared to other types of applications, such as desktop applications and native apps.

Considering the space that the application will take and the platform, web-based was judged to be more suitable since it does not take a lot of memory because everything is on the internet. With web-based applications, the user will not need to install the app of his devices.

Finally, the application aims to provide people with full diabetes control of their healthcare; the choice of a web-based application allows user independence and opportunity in such a way that the user can access the web application anywhere and at any time on any device connected to the internet and with a browser.

4.2.2 Limitations of implementations

The significant constraints of this application were with the software design process. The agile method was the software design method used for the implementation of this application. The agile method necessitates user involvement, which makes sure that the user's needs are met. This method allows for change; therefore, the requirement was subject to modifications and alterations. Furthermore, the need for the software to be useful and solve user problems influenced the choice of the agile method of software development, which focuses on simplicity in both the development processes and the software.

4.2.3 Implementation techniques

The application was broken into various sub-components and developed into stages to make the implementation more comfortable and practical. The first stage was to make sure that the primary user's diabetics add all their interfaces working, next to that was the validation of forms in the application.

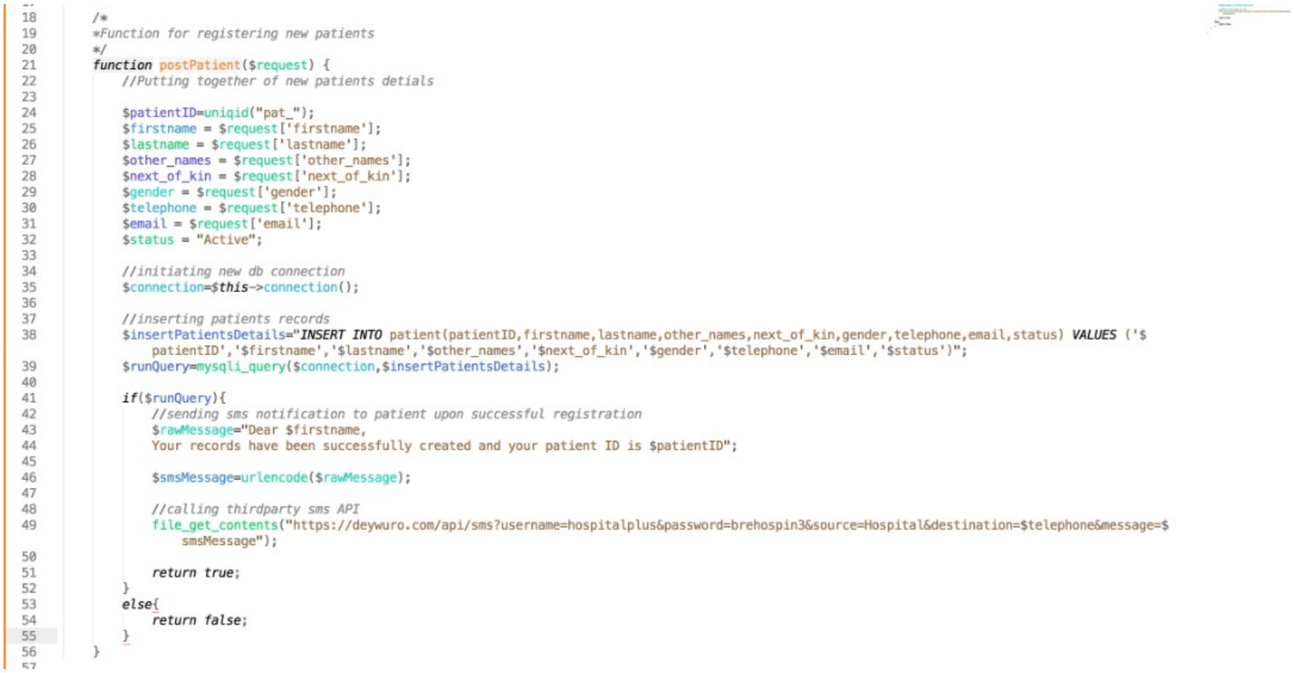
Guided by the agile method of development, the involvement of users was necessary for the success of the application development. The most important part was the transmission of the user 1 (people with diabetes) data to the user 2 (health workers). Also, feedback from the user was integrated into the development process. New requests from users were taken into consideration and analyzed through the requirements engineering and specifications processes, and modifications were made to meet these new requirements.

4.3 Description of components

The implementation of the web-based application was broken into subsections for flexible development. The critical functions of the app are illustrated below:

1. Appointment booking: this major component deals with all the steps the users must complete to be able to schedule an appointment. This superior functionality requires different stages:

  1. Registration: all users must register before they can perform any activities on this application. This component ensures the full log of the user whose information will be saved in the database. The figure below is the snippet of code for registering patients.

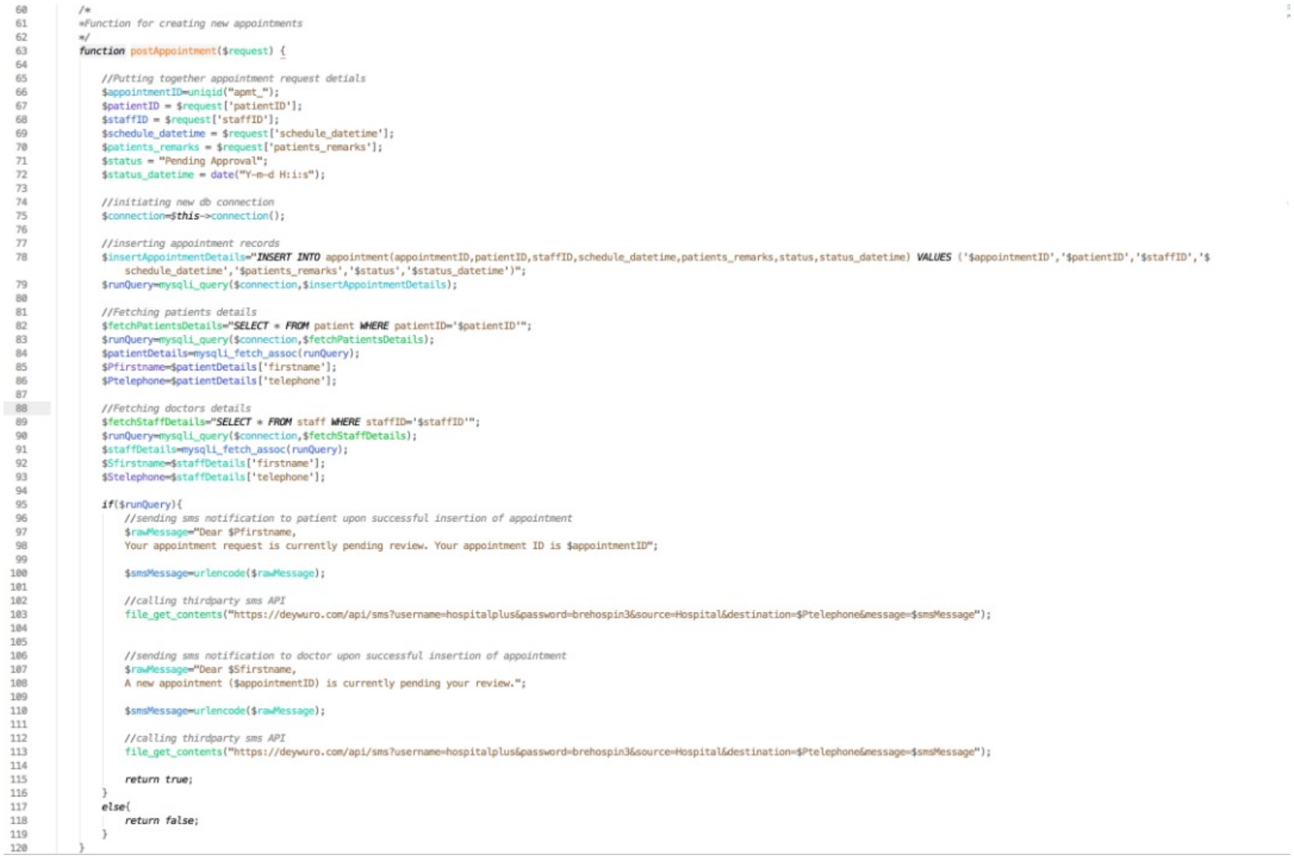


```
/*
*Function for registering new patients
*/
function postPatient($request) {
    //Putting together of new patients detials

    $patientID=uniqid("pat_");
    $firstname = $request['firstname'];
    $lastname = $request['lastname'];
    $other_names = $request['other_names'];
    $next_of_kin = $request['next_of_kin'];
    $gender = $request['gender'];
    $telephone = $request['telephone'];
    $email = $request['email'];
    $status = "Active";

    //initiating new db connection
    $connection=$this->connection();

    //inserting patients records
    $insertPatientsDetails="INSERT INTO patient(patientID,firstname,lastname,other_names,next_of_kin,gender,telephone,email,status) VALUES ('$
        patientID','$firstname','$lastname','$other_names','$next_of_kin','$gender','$telephone','$email','$status')";
    $runQuery=mysqli_query($connection,$insertPatientsDetails);

    if($runQuery){
        //sending sms notification to patient upon successful registration
        $rawMessage="Dear $firstname,
        Your records have been successfully created and your patient ID is $patientID";

        $smsMessage=urlencode($rawMessage);

        //calling thirdparty sms API
        file_get_contents("https://deywuro.com/api/sms?username=hospitalplus&password=brehospin3&source=Hospital&destination=$telephone&message=$
            smsMessage");

        return true;
    }
    else{
        return false;
    }
}
```

Figure 9:Snippet of code for registration

2. Login: After registering, the patient should have the ability to login to access other functionalities of the applications. An unsuccessful login will require the users to reset their password.

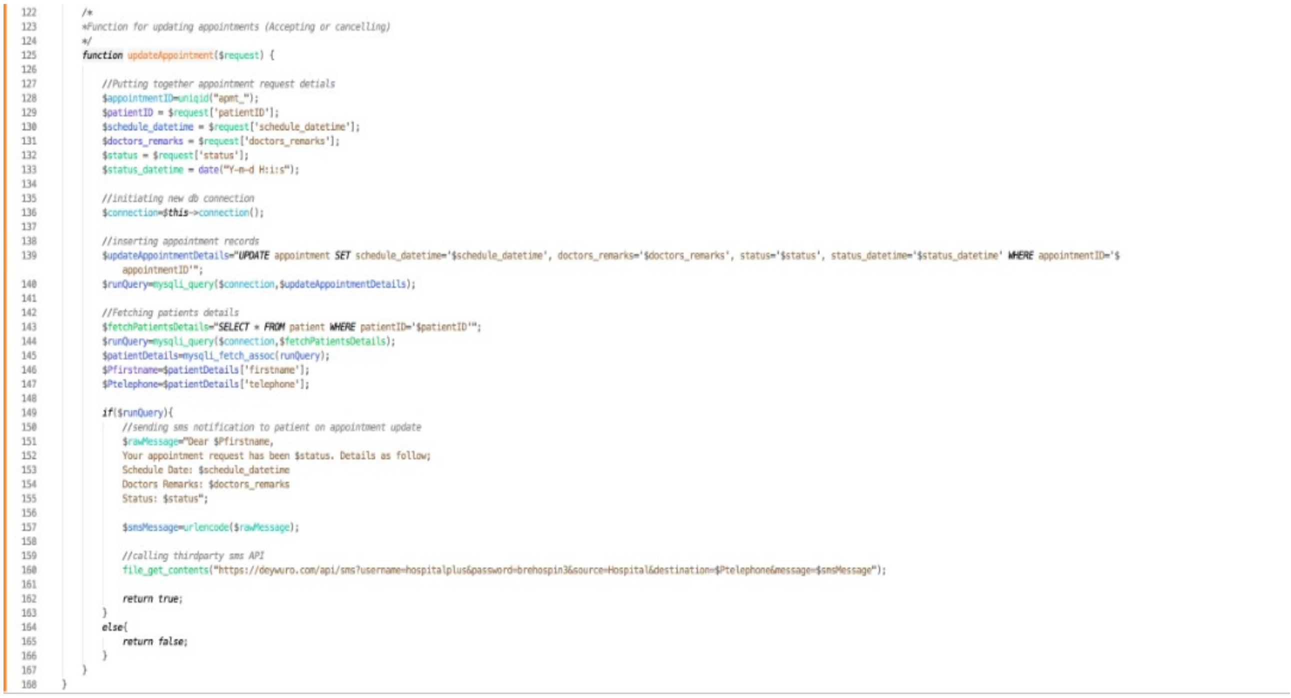
3. Creating a new a new appointment: while the user 1 (patient) is logged in, he/she can access the user 2's (doctor) appointment calendar so that they can fill the empty slot of the schedule and request their appointment. After entering the required information, a calendar of user two will be filled with the new arrangement. Below is the snippet of code for creating new methods.



```
/*
*Function for creating new appointments
*/
function postAppointment($request) {

    //Putting together appointment request detials
    $appointmentID=uniqid("apmt_");
    $patientID = $request['patientID'];
    $staffID = $request['staffID'];
    $schedule_datetime = $request['schedule_datetime'];
    $patients_remarks = $request['patients_remarks'];
    $status = "Pending Approval";
    $status_datetime = date("Y-m-d H:i:s");

    //initiating new db connection
    $connection=$this->connection();

    //inserting appointment records
    $insertAppointmentDetails="INSERT INTO appointment(appointmentID,patientID,staffID,schedule_datetime,patients_remarks,status,status_datetime) VALUES ('$appointmentID','$patientID','$staffID','$
        schedule_datetime','$patients_remarks','$status','$status_datetime')";
    $runQuery=mysqli_query($connection,$insertAppointmentDetails);

    //Fetching patients details
    $fetchPatientsDetails="SELECT * FROM patient WHERE patientID='$patientID'";
    $runQuery=mysqli_query($connection,$fetchPatientsDetails);
    $patientDetails=mysqli_fetch_assoc(runQuery);
    $Pfirstname=$patientDetails['firstname'];
    $Ptelephone=$patientDetails['telephone'];

    //Fetching doctors details
    $fetchStaffDetails="SELECT * FROM staff WHERE staffID='$staffID'";
    $runQuery=mysqli_query($connection,$fetchStaffDetails);
    $staffDetails=mysqli_fetch_assoc(runQuery);
    $Sfirstname=$staffDetails['firstname'];
    $Stelephone=$staffDetails['telephone'];

    if($runQuery){
        //sending sms notification to patient upon successful insertion of appointment
        $rawMessage="Dear $Pfirstname,
        Your appointment request is currently pending review. Your appointment ID is $appointmentID";

        $smsMessage=urlencode($rawMessage);

        //calling thirdparty sms API
        file_get_contents("https://deywuro.com/api/sms?username=hospitalplus&password=brehospin3&source=Hospital&destination=$Ptelephone&message=$smsMessage");


        //sending sms notification to doctor upon successful insertion of appointment
        $rawMessage="Dear $Sfirstname,
        A new appointment ($appointmentID) is currently pending your review.";

        $smsMessage=urlencode($rawMessage);

        //calling thirdparty sms API
        file_get_contents("https://deywuro.com/api/sms?username=hospitalplus&password=brehospin3&source=Hospital&destination=$Ptelephone&message=$smsMessage");

        return true;
    }
    else{
        return false;
    }
}
```

Figure 10:Snippet of code of appointment creation

1. Accepting or canceling appointment: After the user 1 (diabetics) fills the slot, the user 2 (health worker) has the flexibility to accept, cancel, or ask the user for rescheduling.



```
    /*
    *Function for updating appointments (Accepting or cancelling)
    */
    function updateAppointment($request) {

        //Putting together appointment request detials
        $appointmentID=uniqid("apmt_");
        $patientID = $request['patientID'];
        $schedule_datetime = $request['schedule_datetime'];
        $doctors_remarks = $request['doctors_remarks'];
        $status = $request['status'];
        $status_datetime = date("Y-m-d H:i:s");

        //initiating new db connection
        $connection=$this->connection();

        //inserting appointment records
        $updateAppointmentDetails="UPDATE appointment SET schedule_datetime='$schedule_datetime', doctors_remarks='$doctors_remarks', status='$status', status_datetime='$status_datetime' WHERE appointmentID='$
            appointmentID'";
        $runQuery=mysqli_query($connection,$updateAppointmentDetails);

        //Fetching patients details
        $fetchPatientsDetails="SELECT * FROM patient WHERE patientID='$patientID'";
        $runQuery=mysqli_query($connection,$fetchPatientsDetails);
        $patientDetails=mysqli_fetch_assoc(runQuery);
        $Pfirstname=$patientDetails['firstname'];
        $Ptelephone=$patientDetails['telephone'];

        if($runQuery){
            //sending sms notification to patient on appointment update
            $rawMessage="Dear $Pfirstname,
            Your appointment request has been $status. Details as follow;
            Schedule Date: $schedule_datetime
            Doctors Remarks: $doctors_remarks
            Status: $status";

            $smsMessage=urlencode($rawMessage);

            //calling thirdparty sms API
            file_get_contents("https://deywuro.com/api/sms?username=hospitalplus&password=brehospin3&source=Hospital&destination=$Ptelephone&message=$smsMessage");

            return true;
        }
        else{
            return false;
        }
    }
}
```

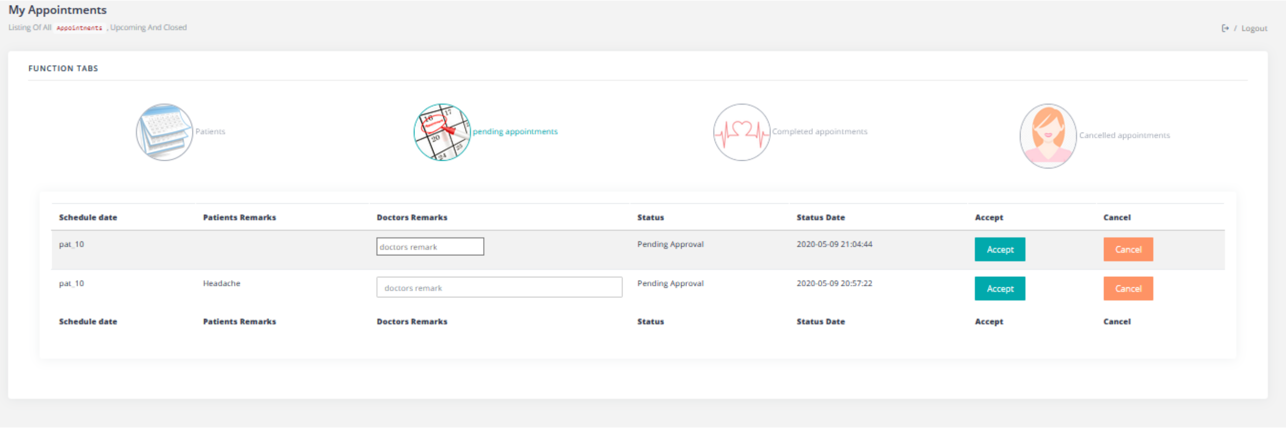
Figure 11: Snippet of the code for canceling and accepting the appointment

2. Calories checking: the diabetics has the flexibility of entering the consumed calories manually.

4.4 Evidence of implementation

This part of the work displays various screenshots of pages implemented according to multiple users . the Diabetics patients and the health workers.

4.4.1 The doctor view



Schedule date	Patients Remarks	Doctors Remarks	Status	Status Date	Accept	Cancel
pat_10		doctors remark	Pending Approval	2020-05-09 21:04:44	Accept	Cancel
pat_10	Headache	doctors remark	Pending Approval	2020-05-09 20:57:22	Accept	Cancel
Schedule date	Patients Remarks	Doctors Remarks	Status	Status Date	Accept	Cancel

Figure 12: screenshot of page listing all the pending appointment for the doctors

4.4.2 The diabetic patient views

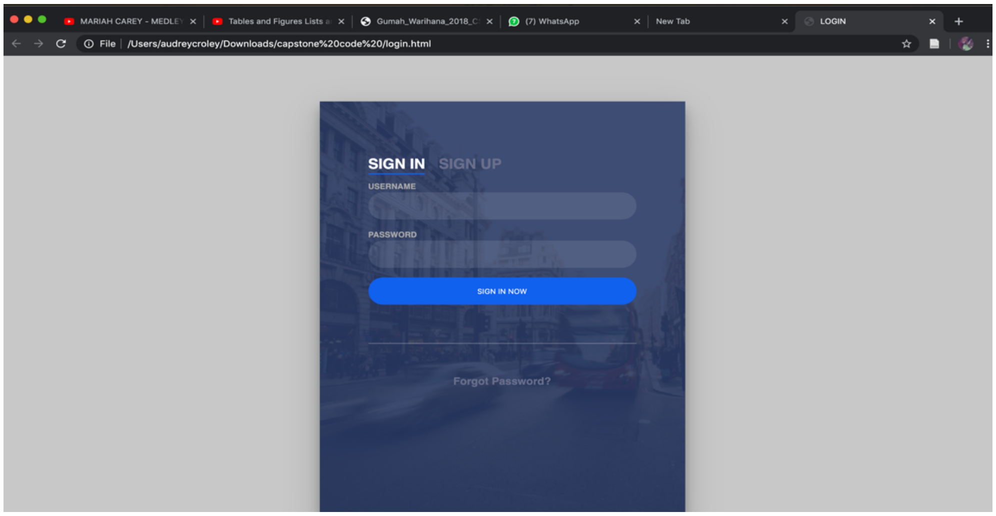
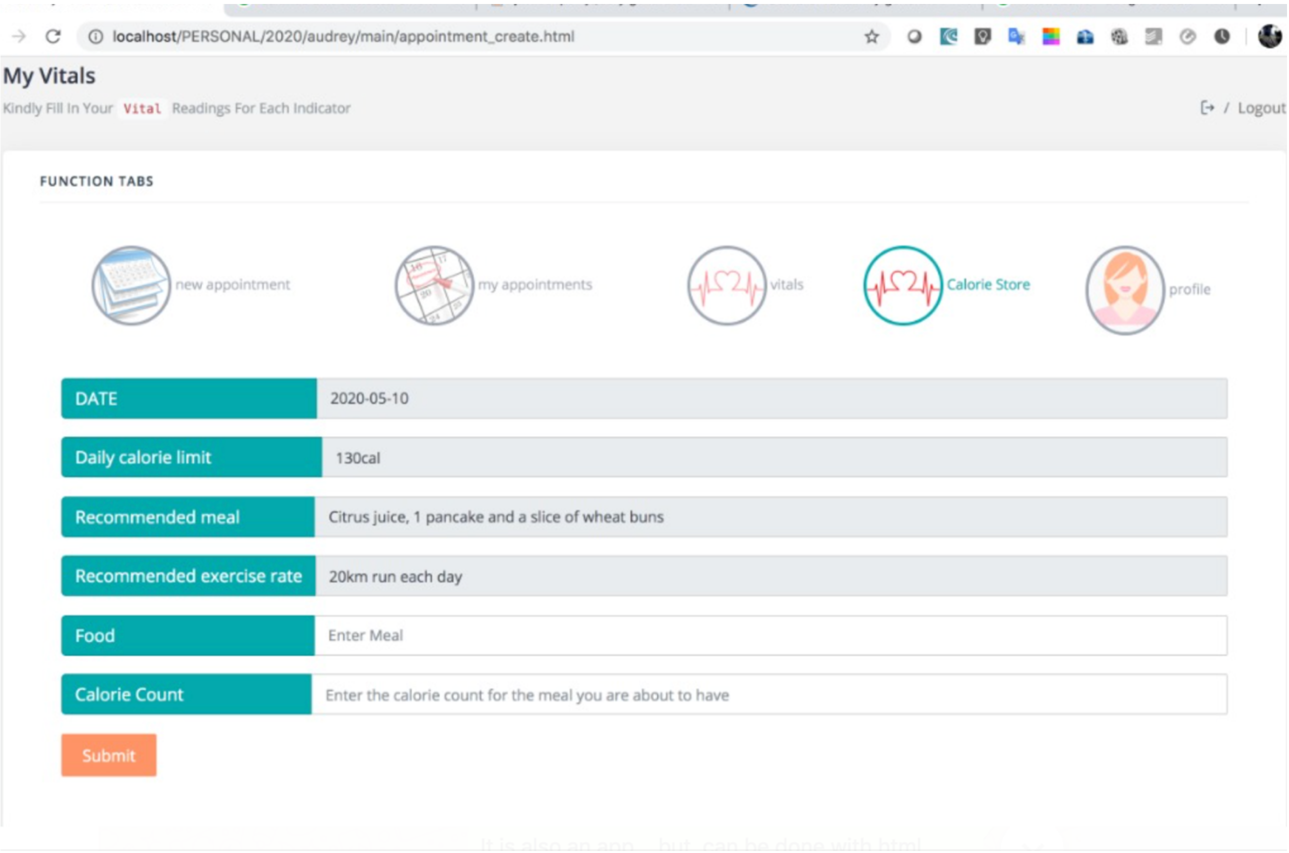
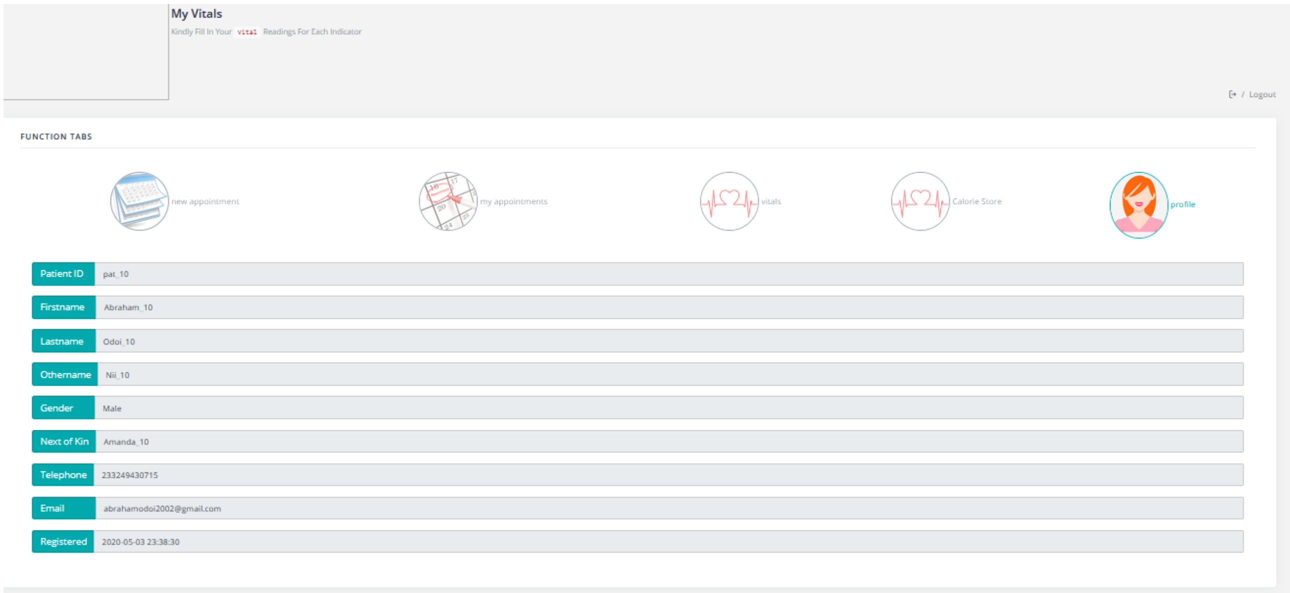


Figure 13 : the login page for the diabetic patients (User 1)


localhost/PERSONAL/2020/audrey/main/appointment_create.html
My Vitals
Kindly Fill In Your Vital Readings For Each Indicator
Logout
FUNCTION TABS
new appointment
my appointments
vitals
Calorie Store
profile
Submit

DATE	2020-05-10
Daily calorie limit	130cal
Recommended meal	Citrus juice, 1 pancake and a slice of wheat buns
Recommended exercise rate	20km run each day
Food	Enter Meal
Calorie Count	Enter the calorie count for the meal you are about to have

Figure 14: the calorie and meal checking page



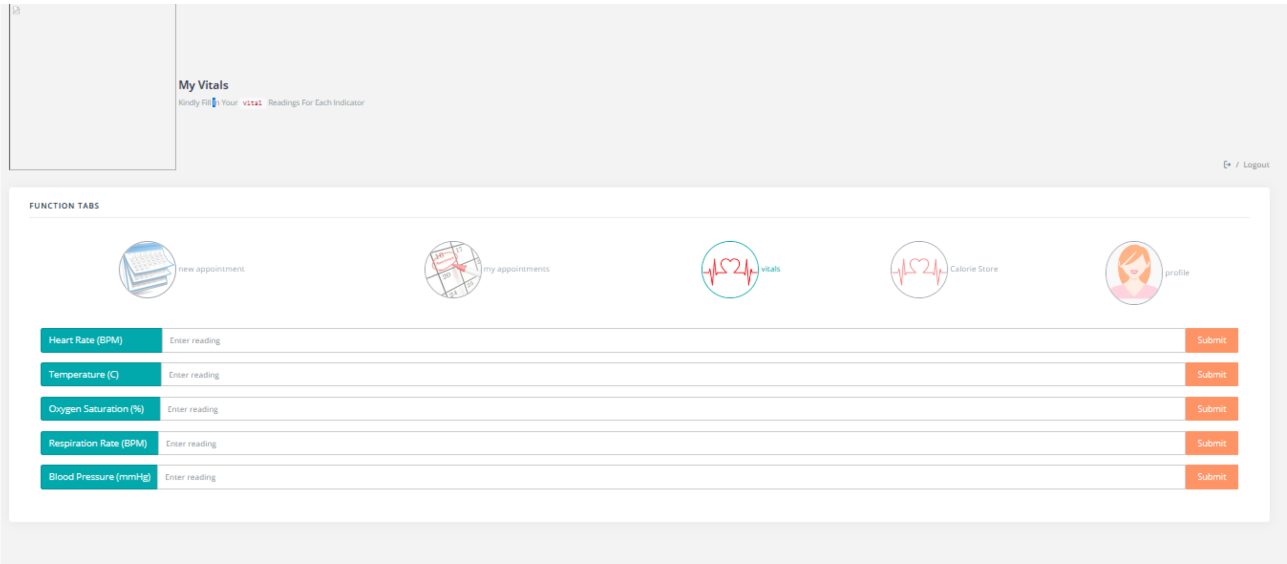
My Vitals

Kindly Fill In Your `vital` Readings For Each Indicator

FUNCTION TABS

Patient ID	pat_10
Firstname	Abraham_10
Lastname	Odoi_10
Othername	Nii_10
Gender	Male
Next of Kin	Amanda_10
Telephone	233249430715
Email	abrahamodoi2002@gmail.com
Registered	2020-05-03 23:38:30

Figure 15: screenshot of user's profile page



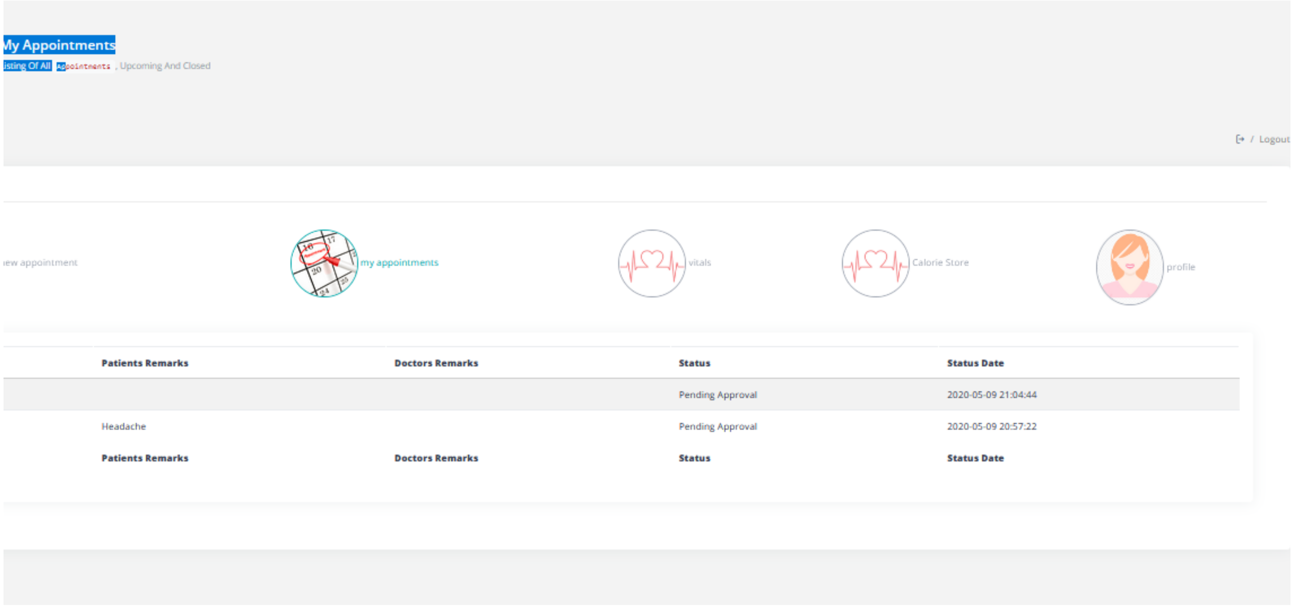
My Vitals

Kindly Fill In Your `vital` Readings For Each Indicator

FUNCTION TABS

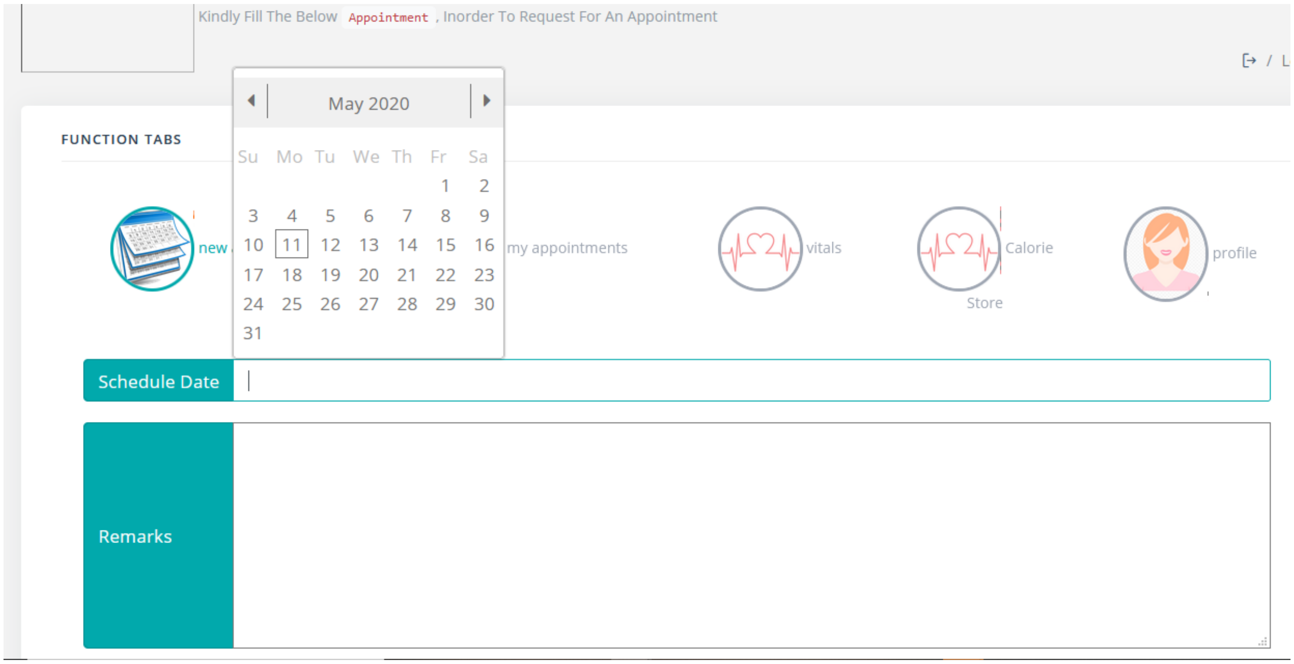
Heart Rate (BPM)	Enter reading
Temperature (C)	Enter reading
Oxygen Saturation (%)	Enter reading
Respiration Rate (BPM)	Enter reading
Blood Pressure (mmHg)	Enter reading

Figure 16: screenshot of vital entry page



Patients Remarks	Doctors Remarks	Status	Status Date
		Pending Approval	2020-05-09 21:04:44
Headache		Pending Approval	2020-05-09 20:57:22
Patients Remarks	**Doctors Remarks**	**Status**	**Status Date**

Figure 17: screenshot of appointment status checking


Kindly Fill The Below Appointment , Inorder To Request For An Appointment
/ L
FUNCTION TABS
new
my appointments
vitals
Calorie
Store
profile
May 2020

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Schedule Date

Remarks

Figure 18: screenshot of Appointment scheduling page

CHAPTER 5:TESTS AND RESULTS

5.1 Chapter overview

Good, well-tested examples are critical to user's reliance on documentation. Conversely, cases that fail when executed, will cause customers to lose faith in a product quickly. Therefore, a formal test plan should be written and maintained, and the examples themselves should be grouped into test cases for present and future test purposes [5]. This chapter focuses on the testing methods that were used to ensure that this application meets all the requirements specified in Chapter 2. The testing techniques adopted were unit testing, system testing, and component testing.

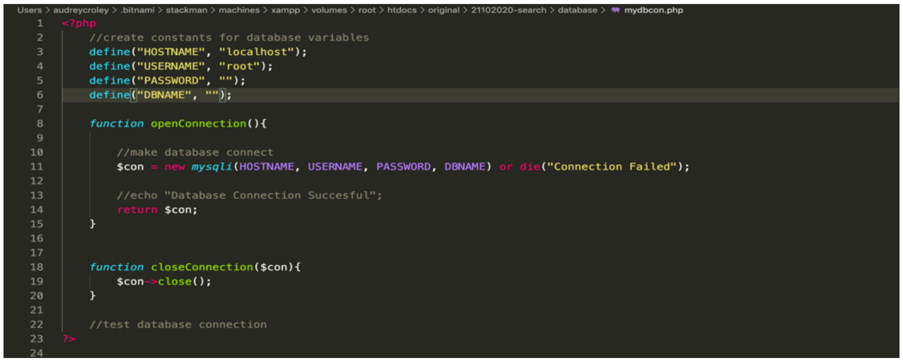
5.2 Unit testing

The Function of various PHP classes was tested to ensure a smooth operation. The PHP unit testing framework was used to test the object of types.

5.3 Components testing

Since this application is more concerned about the data, the significant testing was done at the database side. The DAO (Database Access Object), the connection to the database, the queries and schemas were tested.

DAO testing: In the PHP document, a code to connect was written, and the response was recorded. the connection is only successful when a positive output is obtained



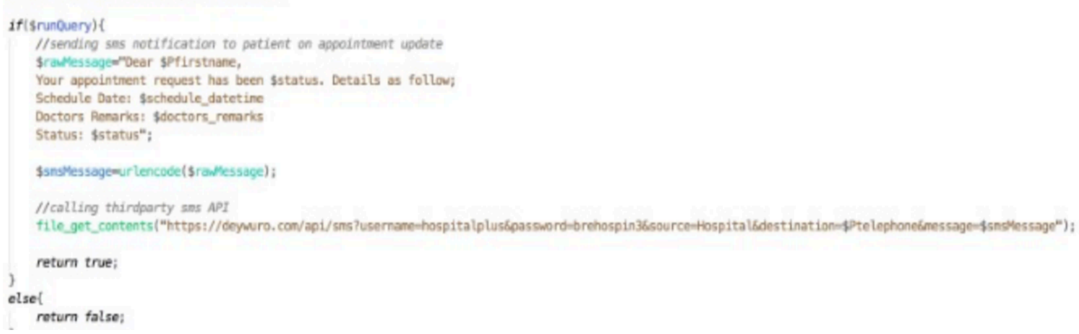
Users > audreycroley > .bitnami > stackman > machines > xampp > volumes > root > htdocs > original > 21102020-search > database > mydbcon.php

```php
<?php
    //create constants for database variables
    define("HOSTNAME", "localhost");
    define("USERNAME", "root");
    define("PASSWORD", "");
    define("DBNAME", "");

    function openConnection(){

        //make database connect
        $con = new mysqli(HOSTNAME, USERNAME, PASSWORD, DBNAME) or die("Connection Failed");

        //echo "Database Connection Succesful";
        return $con;
    }


    function closeConnection($con){
        $con->close();
    }

    //test database connection
?>

```

Figure 19: DAO testing

3. Schema testing: when data is sent to the database, tables, and columns existence is checked, their structure was also tested to accommodate all incoming data. This step ensures the correct collection and retrieval of data.

4. The queries testing was based on how the information was successfully recovered and passed to the query.



```php
if($runQuery){
    //sending sms notification to patient on appointment update
    $rawMessage="Dear $Pfirstname,
    Your appointment request has been $status. Details as follow;
    Schedule Date: $schedule_datetime
    Doctors Remarks: $doctors_remarks
    Status: $status";

    $smsMessage=urlencode($rawMessage);

    //calling thirdparty sms API
    file_get_contents("https://deywuro.com/api/sms?username=hospitalplus&password=brehospin3&source=Hospital&destination=$Ptelephone&message=$smsMessage");

    return true;
}
else{
    return false;
```

Figure 20:Code for Query Testing

5.4 Systems testing

This application was hosted with 000webhost. To test the systems, three techniques were used: load testing, usability testing, and functional testing.

1. Usability testing: this focuses on the ease of using the application. The application was tested by different users to ensure its redundancy from the early implementation; users were involved in providing feedback and determining the user's expectations. Approximately nine users have tested this application in 3 months to assess its usability. The overall feedback indicated that the application was easy to use.

2. Load testing: This technique focuses on how the application will perform when the community of users increases, how the app will perform under real-time and pressure. At the final stages, 20 users were made to use the system simultaneously. The test revealed that there is 95% efficiency based on the response time.

3. Functional testing: this technique was used to make sure that all the functionalities mentioned in chapter 2 were implemented and worked effectively. This test generated the idea of the SMS notification two days and 12 hours before the appointment time.

CHAPTER 6:CONCLUSION AND RECOMMENDATIONS

6.1 Chapter overview

This chapter works as a summary of all that was done in my project work. It also illustrates the challenges faced during the execution of the project work and how this work can be improved.

6.2 Recommendations

This project aimed to provide a system that will help people with diabetes to limit their visits to the hospital and develop their ability to self-manage their Health. In solution, the system's significant functions were to keep the health workers and the diabetic patients in constant touch and have the people with diabetes having their essential health information regularly at their availability for better monitoring. The systems were successful in delivering functions that will work towards the achievements of these goals; however, the systems lack few functionalities that could have also helped people with diabetes. Therefore, I recommend:

1. A mobile application platform: Currently, there is only a web version of this application. Thus, the decision to develop a web-based application was its flexibility. An area for further research that could improve usability and efficiency is a mobile application platform. This platform could work hand in hand with other health applications such as Health in apple devices.

2. To develop the app, future development can include the Connection to Lancet Device to read the patient's blood sugar directly. The connection to a fit bit to understand the patient's heart rate, and to use the pedometers installed in the patients' phone to track steps taken in a day.

6.3 Challenges

1. One major challenge was with getting the calorie tracking features. Finally, to get this functionality working even though the initiative was to get it through machine learning, the available option was to let the client enter is calorie consumption by himself.

2. Another challenging area was getting users to test the application. Due to the COVID-19 pandemic, most health workers were busy, and few had time to participate in the testing of the app; most testing was done by the first category of users (diabetics).

6.4 CONCLUSION

This project introduced a way of managing chronic diabetes through web application SugarDose. This application facilitates remote consultations and provides a reliable medium for storing patient data. Though there is still room for improvement, this application will bring a plus in the lives of people living with diabetes and facilitate the work of the people providing Health for them. The relevance of this project could be measured by how it reduces the frequent visits of diabetics to the hospitals as well as by how it reduces the workload on the health workers.

REFERENCES

[1] Emmanuel Kusi Achampong. 2012. Electronic Health Record System: A Survey in Ghanaian Hospitals. J Health Med Inform 03, 02 (2012). DOI:https://doi.org/10.4172/scientificreports.164

[2] Martha Acquah-Swanzy. Evaluating Electronic Health Record Systems in Ghana: the case of Effia Nkwanta Regional Hospital. 159.

[3] Rudolph Bedeley and Prashant Palvia. 2014. A Study of the Issues of E-Health Care in Developing Countries: The Case of Ghana. (2014), 12.

[4] Jodi T. Bernstein, Wendy Lou, and Mary R. L'Abbe. 2017. Examining the Relationship between Free Sugars and Calorie Contents in Canadian Prepacked Foods and Beverages. Foods 6, 9 (September 2017). DOI:https://doi.org/10.3390/foods6090075

[5] Gary Bist. 1995. Testing Examples: A Guide. Technical Communication 42, 1 (1995), 84–93.

[6] Mensah Kwabena Patrick and David Sanka Laar. 2012. E-Health Dedicated Hybrid Cloud: a Solution to Ghana's Health Delivery Problems. 4, 9 (2012), 10.

[7] Deborah S. Ray and Eric J. Ray. 1998. Bridging the Gap Between SGML and HTML: The Potential of XML for Technical Communicators. Technical Communication 45, 3 (1998), 427–432.

[8] Gojka Roglic and World Health Organization (Eds.). 2016. Global report on diabetes. World Health Organization, Geneva, Switzerland.

[9] Karima Saleh. 2013. The Health Sector in Ghana. Retrieved February 28, 2020 from https://elibrary.worldbank.org/doi/pdf/10.1596/978-0-8213-9599-8

[10] Emmanuel Akwoulo AGYIGEWE Teviu, Edwin Afari, Samuel Oko Sackey, P Boni, F Wurapa, Moses Aikins, and T I Abdulai. (PDF) Improving Medical Records Filing in a Municipal Hospital in Ghana. Retrieved February 28, 2020 from

https://www.researchgate.net/publication/236675906_Improving_Medical_Records_Filing_in_a_Municipal_Hospital_in_Ghana

[11] Steve F. Tyson. 2011. Decode The PHP Codes: A Simple And Easy PHP Tutorial For Beginners With Clear-Cut Details On HTML Basics, PHP Coding And Other PHP Basics So You Can ... PHP Scripts Like An Expert PHP Programmer. CreateSpace, Scotts Valley, CA.

[12] Smartphone and Internet Penetration in Ghana on the Rise - Think Expand. Retrieved February 28, 2020 from https://www.expandgh.com/smartphone-and-internet-penetration-in-ghana-on-the-rise/

[13] Young Ghanaian Tech Enthusiast Creates App to Ward off Diabetes. Retrieved February 28, 2020 from https://globalpressjournal.com/africa/ghana/young-ghanaian-tech-enthusiast-creates-app-to-ward-off-diabetes/

[14] ‘AfyaPap’ app introduced to help manage diabetes | Value Added Services News in Kenya. Retrieved February 28, 2020 from https://www.biztechafrica.com/article/afyapap-app-introduced-help-manage-diabetes/12260/

[15] CS2Ah0405-SoftwareRequirements.pdf. Retrieved March 3, 2020 from http://www.inf.ed.ac.uk/teaching/courses/ip/CS2Ah0405-SoftwareRequirements.pdf

[16] 3-Tier Architecture: A Complete Overview - JReport. Retrieved March 5, 2020 from https://www.jinfonet.com/resources/bi-defined/3-tier-architecture-complete-overview/

[17] Application Layer - an overview | ScienceDirect Topics. Retrieved March 5, 2020 from https://www.sciencedirect.com/topics/engineering/application-layer

[18] Documentation for Visual Studio Code. Retrieved April 15, 2020 from https://code.visualstudio.com/docs

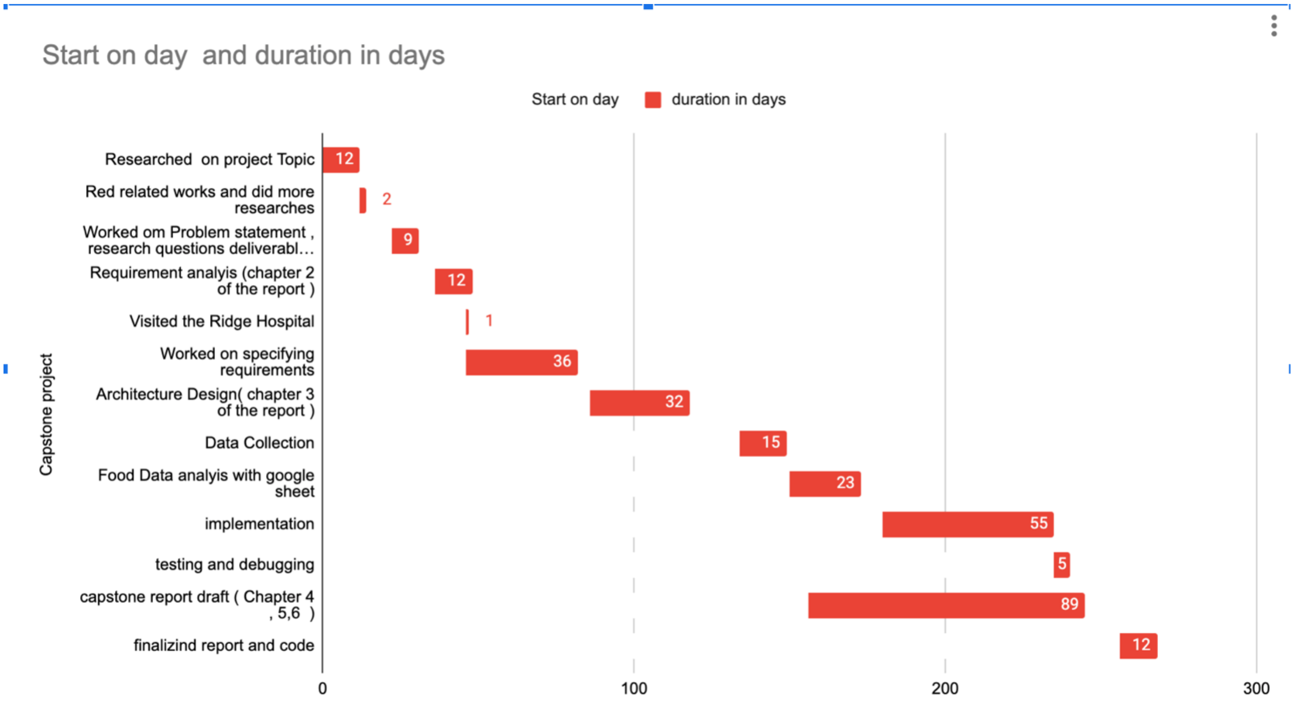
APPENDIX

Appendix A: Test Cases and Results

Test case	**Input Data**	**Expected results**	**Actual results**	**Pass /Fail**
User's successful login test	Email, password	See dashboard for patients (if patient) or dashboard for doctors (if doctor)	See dashboard for patients or doctor	Pass
Test login for unregistered user	Unregistered email and password	Login failed error message	Login failed error message	Pass
Test user login for valid email and empty password	Valid username	Error: "Password cannot be empty"	Error: "Password cannot be empty"	Pass
Test user login for valid password and empty email	Valid password	Error : "invalid credentials"	Error: "Invalid credential"	Pass
Check all links in patient dashboard page		Links lead to pages required	Links lead to expected pages	Pass
Check all links in doctor dashboard page		Links lead to required pages	Links lead to expected pages	Pass
Test if input meals form	Filled form with meals and quantities	Successful submission of data	Successful submission of data	Pass

submits actual data				
Test if doctor can contact patient	On the doctor’s dashboard, click on contact patient button beside patient’s information	SMS box is opened for doctor’s message to be typed and sent.	Patient receives message	Pass
Test if patient can view doctor’s calendar	Click on schedule appointment button on patient dashboard page	Doctor’s calendar is shown	Doctor’s calendar is shown	Pass
Test if patient can select date for appointment	Click on date on doctor’s calendar	Date is blocked out. Success message received. SMS sent to doctor	Date is blocked out, success message received, doctor receives SMS	Pass

Appendix B : Project Management Gantt Chart


Start on day and duration in days
Start on day
🟥 duration in days
Capstone project
Researched on project Topic
Red related works and did more researches
Worked om Problem statement , research questions deliverabl…
Requirement analyis (chapter 2 of the report)
Visited the Ridge Hospital
Worked on specifying requirements
Architecture Design(chapter 3 of the report)
Data Collection
Food Data analyis with google sheet
implementation
testing and debugging
capstone report draft (Chapter 4 , 5,6)
finalizind report and code