ASHESI UNIVERSITY

A WEB-BASED REMOTE MEDICAL ASSISTANT FOR THE MOBILITY IMPAIRED AND AGED IN ACCRA – RHEMA RAPHA MEDICAL CENTER

APPLIED PROJECT

BSc. Management Information Systems

Pamela Anang

2021

ASHESI UNIVERSITY

A WEB-BASED REMOTE MEDICAL ASSISTANT FOR THE MOBILITY IMPAIRED AND AGED IN ACCRA

APPLIED PROJECT

Applied Capstone Project 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.

Pamela Anang

2021

Declaration

I hereby declare that this applied project is the result of my own 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:

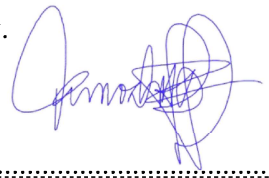
Pamela Anang.....

Date:

27th April 2021.....

I hereby declare that the preparation and presentation of this applied project were supervised in accordance with the guidelines on the supervision of the applied project laid down by Ashesi University.

Supervisor's Signature:



.....

Supervisor's Name:

Dr David Ebo Adjepon-Yamoah.....

Date:

27th April 2021.....

Acknowledgments

I want to honor the Almighty God for His wonderful mercies and abundant grace throughout this academic journey and the completion of this project. His watchful eye has brought this first major league of my life to an end, and for that, I want to say, "Thank You, My God."

I would also like to extend gratitude to my supervisor, Dr. David Ebo Adjepon-Yamoah, who saw this project as one worth exploring. His words of encouragement, such as "Don't worry about it. You'll do great," amongst others, kept this project running. His support and advice helped in the full implementation of this project precisely when I encountered roadblocks. I would like to say "Thank you" to the Rhema Rapha Medical Center physicians and patients who participated in this project's research and testing phases.

Finally, I would like to show appreciation for my family and the friends God brought my way through this experience. With their constant love, knowledge, and feedback, this project was successfully completed. Hence, I dedicate this project to all the participants as mentioned above. Thank you, and God Bless you all.

Abstract

This 'MediCare' project is centered on discovering telemedicine devices in Ghana and their implementation and use within the Ghanaian setting. The user needs assessment conducted at the beginning of this project was used to uncover system functional and non-functional requirements and benchmark the implementation of a system that contributes to the fulfillment of SDG 3; good health and well-being by promoting remote medical care. Since the COVID 19 pandemic placed a strict restriction on movement to and from medical facilities, the implementation of MediCare sought to offer patients, precisely the mobility impaired and aged who are most vulnerable in society, a way to receive medical care from the comfort of their homes. This system was designed for Rhema Rapha Medical center located in the Tse Addo community of Accra due to their desire for a telemedicine device that facilitates health consultation virtually and their patient demographic ranging from the middle to upper-class population of Ghana. This patient demographic is likely to have a more significant percentage consisting of the aged and mobility impaired. The results of this survey indicated a 100% desire for the deployment of this system for Rhema Rapha Medical Center and 88.2% achievement of functional requirements identified at the beginning of user needs assessment.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

1.1 Background and Motivation

Telemedicine has gradually become a relevant part of medical aid in the world [1]. It incorporates information and communication technology tools to provide health consultation to people worldwide [2]. Tailoring to the African diaspora in developing countries, telemedicine has received significant attention [3]. However, it rarely made substantial progress in its implementation due to a lack of governmental support in promotion, especially private sector initiatives and the low funding for large-scale distribution [3].

The primary purpose of telemedicine is to deliver faster medical attention to patients remotely using technology. This aids in the achievement of Sustainable Developmental Goal (SDG) Three. SDG Three deals with promoting good and healthy lives while encouraging all people's well-being [4]. Research shows that telemedicine improves the quality of health patients receive. This is done by limiting the possibility of unwanted referrals due to qualified personnel's unavailability [5]. In this situation, transportation costs and the time required to receive healthcare are reduced [6]. In so doing, it addresses health care issues in developing countries that are unable to equip medical facilities with experienced staff, while their population, specifically the aged and physically challenged, are unable to commute to medical facilities [5]. The use of ICT tools provides 24-hour access to medical information and teleconsultation for both doctors and patients, respectively [5], which increases the response to patient issues such as suicidal tendencies [8].

The attention to the adoption of home telemedicine services is limited [7] due to technical illiteracy. However, it is highly recommended that systems be built to improve patients' quality of life by providing medical assistance from the comfort of their homes. This reduces the need to commute to and from a hospital due to less demanding health issues

[7]. Conversely, transporting to a medical facility to receive medical care can be tiresome and hectic for mobility-impaired patients and the aged. Hence there is the need to develop a telemedicine system.

Telemedicine applications typically comprise of a real-time consultation section for patients. This includes direct communication with a medical specialist either through messaging or virtual video consultation [37]. It also consists of a repository containing information readily available to medical staff for diagnosis and a remote health monitoring functionality that allows medical advisors to monitor their patients' progress and health, which is the other central aspect of the telemedicine application [37]. *MDLive* [9] is a telemedicine provider that partners with other health operators to offer telemedical assistance to patients of a hospital. The organization partners with hospitals to provide secure platforms and interfaces that conform to the health insurance portability and accountability act to offer healthcare access in the American community. The system includes an AI chatbot functionality for system assistance to both patients and doctors. The organization consults with health operators and uses AI technology to develop telemedicine devices while training hospitals on telehealth systems and increasing brand awareness and revenue in the market [9].

As a result of the COVID-19 pandemic, physical access to medical facilities received strict restrictions unless necessary. This drew attention to the massive need for telemedical systems that are solely designed to support remote consultation between patients and doctors without interruptions [37]. In Ghana, *Nyaho* medical center [10] partnered with Africa eHealth Solutions International to release a telemedicine device that provided patients with ailments such as diabetes and hypertension remote direct contact to their medical practitioners to monitor patient vital signs. The system allows patients to manage their health while keeping them informed on their condition and progress. The purpose of the application is to manage patients' chronic ailments daily while away from a physical

consulting room. This has been discovered to reduce the cost for both patients and the hospital by giving doctors access to monitor patient health remotely to alleviate complications that would result in high costs and improve patient outcomes. This application's significance informs the provision of a self-monitoring or reminder feature in the proposed system to encourage patients to control their own health [10].

On the other hand, this project's target is to present a useful telemedicine system that provides medical assistance to patients in a hospital, earmarking the middle to upper-class population, specifically the aged and mobility impaired.

1.2 Problem Statement

The health sector took a significant hit this year as a result of the COVID-19 pandemic. Physical access to medical health facilities was greatly restricted due to social distancing protocols and the prevention of contracting the virus upon visiting a medical facility. In response to this, many were advised to contact their medical advisors for health-related issues before attending health facilities in person. Furthermore, the aged and mobility-impaired have long endured the struggle of commuting to and from a hospital to receive medical attention for minor health issues that could be addressed from the comfort of their homes. In situations where doctors or pharmacists are physically unavailable, they fend for themselves through online searches and self-diagnosis, which could be more harmful than good. Though research in telemedicine has been conducted to develop a solution to provide access to medical consultation remotely, there is a need to implement a functional system that addresses the public's needs, specifically the aged and mobility impaired, in receiving medical attention remotely.

1.3 Related Works

Technology has aided in improving health in many ways. Telemedicine has been discovered to impact people's lives by providing remote access to medical aid, especially in rural areas [3]. This section highlights some related technologies implemented to improve remote medical care.

In the Alberta province of Canada, a software called *Connect Care* [11] was developed in 2019 as the central access point for patient health information and delivery. The system incorporates the Alberta Health Record Information System (AHRIS) [11] that comprises a data asset directory and health service data asset inventory summary. The system's main domain includes a repository that stores patient treatment and diagnostic information, an access tool that enables browsing and viewing health information, a registry that stores patient data on location or events, and other identification details and infrastructure that connects and transfers health data across the system. The Connect Care system was implemented to target the health needs of individuals in the Alberta province on a one-on-one basis [11]. It replaced over 1,300 health systems across the region by incorporating customized, relevant functionalities that facilitate remote medical care [11]. The self-assessment tool within the system allows the people of Alberta to identify what appropriate tests to carry out, which alerts and redirects them to test centers. The tool also helps health workers prioritize high-risk populations and determine whether self-isolation is necessary regarding the COVID-19 pandemic [11]. In addition to Connect Care, Alberta's government introduced a contact tracing mobile application that uses Bluetooth-enabled securely encrypted tokens to detect nearby devices with the application activated. This system's relevance to the proposed project is the inclusion and orientation of an information deck or repository that stores patient treatment data and provides a browsing option to locate patient health information.

Moving forward, a study was carried out to test the effectiveness of individual real-time video counseling on the health risks of patients [12]. Patients were examined based on their experience and honest opinions on remote counseling compared to in-person counseling. The sample population included patients with weight, alcohol, smoking, or drug issues [12]. Considering the condition and sort of health attention they needed, patients preferred remote counseling. It made them more comfortable sharing their struggles and promoting individual growth since discrete attention was provided to patients instead of collective group counseling [12]. The use of individual real-time video counseling is vital to this project as it sheds insight into the comfort virtual counseling offers patients.

Bima Ghana [13] is the leading provider of telehealth and insurance delivery in Ghana. The Bima organization has developed a health service software that offers affordable health advice and care to low-income earners, transforming healthcare access in Ghana [13]. The software provides unlimited mobile service calls to medical practitioners at an affordable price to gain medical advice. It includes short messaging services (SMS) and interactive voice response (IVR) health coaching programs that deliver a set of targeted health-related topics to educate patients. It also consists of an electronic prescription via SMS, based on remote consultation received [13]. This system proves relevant to the project to be explored since it uses SMS and over-the-phone consultation. The exploration of SMS consultation would bring insight into its possible adaptation in this project as a mobile application.

Comarch [14] provides a remote medical care system that encourages health monitoring and remote diagnostics. The system uses mobile devices to measure patient vital signs and automatically analyzes patient data. Abnormalities detected are addressed by staff directly contacting patients. The system poses benefits such as relieving staff by tracking vital information of patients [14]. It also provides access to patient medical data using a repository that stores all patient data. The use of monitoring devices and features can

increase patient devotion to maintaining a healthy lifestyle by tracking patient vital signs and offering alerts of valuable ways to alleviate minimal abnormalities [14]. Incorporating the alert and tracking system would be beneficial to explore in the proposed project to monitor a patient's health.

1.4 Objective and Benefits of the Proposed System

The main aim for tackling this problem is to aid in quick response health consultation to patients at Rhema Rapha Medical Center. However, the objectives include encouraging remote medical care, which would reduce physical contact, thereby improving social distancing concerning COVID-19 protocols. The system would also provide direct connectivity to medical experts via real-time video conferencing or chat room features on medical consultation and diagnosis while improving medication delivery to patients. The system would also include a pharmacist section that offers doorstep delivery of medication to patients upon drug availability and patient/doctor request. After working hours, the system would serve as a health consult by providing information on what to do and how to address minor ailments based on previous patient data or symptoms gathered.

The internet is a frequently used resource for gathering information on absolutely anything. However, the acquisition of accurate information over the internet is not guaranteed, given information available is susceptible to public manipulation [15]. Research conducted on the impact of googling health conditions rather than one-on-one conversations with health experts showed that some people feel anxious and frightened at the gravity of information they came across during self-diagnosis [15] [16]. There is also the possibility of discovering incomplete information and outdated information, which may be an oversight on the part of patients or may not be clearly stated by publishers [16]. The proposed web-based system seeks to provide a first-hand direct connection between patients and experts or skilled doctors and pharmacists who are readily available to provide accurate and recent

information based on patient requirements. These professionals are trained to assist patients and are mandated to cater to their patients' needs.

A business is considered profitable when it can maximize its opportunities by utilizing its resources to the fullest [17]. The inclusion of innovation and creativity potentially increases the advantage a business has over others. Therefore, adopting and utilizing this proposed web system would give Rhema Rapha a competitive advantage in the health industry, accelerating its popularity.

1.5 End Product

The resulting solution or product is 'MediCare,' a web-based remote medical assistant that provides round-the-clock remote medical consultation to patients of the Rhema Rapha Medical Center located in the Tse Addo community. It specifically targets the aged and mobility impaired. The system is built for Rhema Rapha as the patient population ranges from the middle to upper-class income earners who are more likely to have internet access and primary vital recording devices such as temperature and a sphygmomanometer blood pressure monitor at their disposal. The system would include real-time video conferencing for consultation and a chatroom feature if preferred. It would consist of a data repository to collect patient vitals data fed to their doctor and a connection between doctor and pharmacist for swift medication delivery and inquiry at Rhema Rapha Medical center located in the Tse Addo community.

Chapter 2: Requirement Analysis

2.1 Introduction

This chapter of the project is to introduce the requirements of the project by stakeholders. It offers a description of the user's projected interaction with the system and a summary of how the system's functionalities and data requirements were gathered. This phase would include the developmental process of the project and the proposed time frame for completion.

2.1.1 Software Development Process

This project would adopt the agile software development process as the methodology for the projects' development life cycle. Agile originally means the ability to adapt to change easily. Agile software development [18], on the other hand, is an umbrella term for a set of frameworks and principles described in a guide called the agile manifesto that aid in the successful development of a project that satisfies customers due to heavy customer involvement and iterations. SCRUM [19] is an agile framework that deals with iterations and incremental product delivery with frequent feedback and collaboration with users to deliver a sound functioning system. In this project, the SCRUM framework is utilized under the agile development process to increase the final product's usability and desirability. This framework would be best suited for this project as the iterations and user interactions give insight into the system's feasibility. Given that new research is carried out in the telemedicine industry regularly, iterations would improve the information available to embed the proposed system's development and implementation.

2.1.2 Stakeholders and Description

The main stakeholders of this project are identified as those that affect and are affected by the system. They are categorized into two main groups.

- The system administrators are Rhema Rapha Medical center's health professionals in Tse Addo and the pharmaceutical professionals. These officials are efficient in the use of technology and have access to internet connectivity.
- The end-users of the system, the community of health patients that attend Rhema Rapha Medical center for medical attention. This group of people contains the middle to upper-class population and has adequate knowledge in technology use.

2.2 System Requirements

In gathering data requirements for this project, qualitative research was conducted to understand preferred system requirements. In determining the sample population for this project, the convenience sampling technique was adopted. The convenience sampling technique [20] is a non-probability sampling technique with its target population made up of people who meet criteria based on convenience such as geographical location, easy accessibility, availability with time, or general willingness to participate. This technique is best suited for the project, given the medical center's geographical proximity to the researcher's residence.

2.2.1 Requirements Gathering

The two main data collection methods were utilized in this project include the observation and immersion data collection technique and Survey questionnaires.

- Observation and Immersion data collection technique - Observation of the hospital's daily operations while studying the patient process of receiving medical attention

through immersion proved vital in generating a detailed understanding of system requirements that would reduce patient waiting time.

- Survey questionnaires data collection technique - Conducting a survey with electronic questionnaires using google forms allowed to maintain reduced physical contact due to social distancing protocols while gaining validated responses on the benefits of a telemedicine system and its usability and desirability among stakeholders.

2.2.2 Requirements Specification

The requirements of the system are categorized into two major sections. These are the functional requirements and non-functional requirements gathered from the results of the requirements survey conducted. This survey contained a total of 20 participants as the sample size with 15 complete responses. Amongst these responses, three were doctors and two pharmacists, with the remaining being patients.

2.2.2.1 Functional Requirements

A system's functional requirements include a system's functionalities that determine what the system must do [21]. The following are some functional requirements of the system.

- **[FR01]** Sign Up and Sign In - Patients are required to sign up and later log in to gain access to further system functionalities.
- **[FR02]** Health consultation - The system displays available physicians. It offers consultation to patients via live chat room or video conferencing with diagnosis and recommendations provided afterward as suggested by doctors and patients.
- **[FR03]** Dispensary Unit - The system integrates a dispensary unit as suggested by pharmacists that provides a quick response for medication delivery via dispatch based on the patient's diagnosis and keeps a medication log.

- **[FR04]** Administrator Management - The administrator management feature offers extra control or back-end management to physicians and pharmacists, such as adding and editing medication and availability and viewing patients' primary vitals data by physicians.
- **[FR05]** Chat Bot - The functioning chatbot offers assistance to patients in the absence of a doctor.
- **[FR06]** Cart and Payment API - Enables drug allocation and purchasing by individual patients.

2.2.2.2 Non-Functional Requirements

The non-functional requirements of a system determine how a system would perform its functional requirements. They include all the remaining functionalities that are not covered by the functional requirements [21]. Below are a few non-functional requirements identified for the system.

- **[NFR01]** Usability Requirements - The system should be easy to use and navigate for both administrators and patients.
- **[NFR02]** Reliability Requirements - The system should perform as expected whenever users access it provided there is stable internet connectivity.
- **[NFR03]** Security Requirements - The system must be secure and make user data available to authorized parties only. The system requires registration before accessing functionalities.
- **[NFR04]** Performance Requirements - The system should provide a quick and accurate response to users efficiently.

2.3 Use Cases and Diagrams

Use case diagrams describe the expected relationship between a user and a system [22]. It demonstrates how a user is expected to interact with a system. Fig 2.0 illustrates the use case for patients and who represent the external stakeholders or end-users. Fig 2.1 shows the system's interaction with the medical center administrators, who are the health physicians and pharmacists of Rhema Rapha.

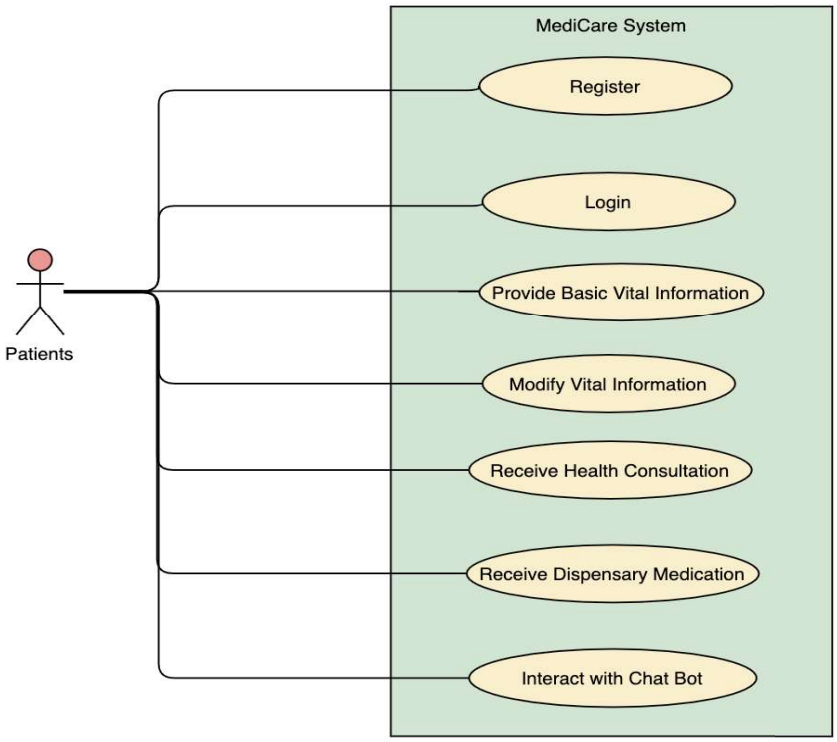


Fig 2.0 Use Case Diagram for Patients

Fig 2.0 shows patient interaction with MediCare System. Patients are required to register, then Log in to the system. Once both processes are successful, they would provide basic vital signs information then are redirected to a platform to receive health consultation with an available physician of their choice. Depending on the diagnosis, they receive medication from the dispensary unit. If the drug is available, it is dispatched to them, and payment is made. If not, they are offered digital prescriptions with a recommended pharmacy nearby.

In the absence of a physician, they are assisted with the chatbot. Patients are allowed to edit vital information for the next consultation.

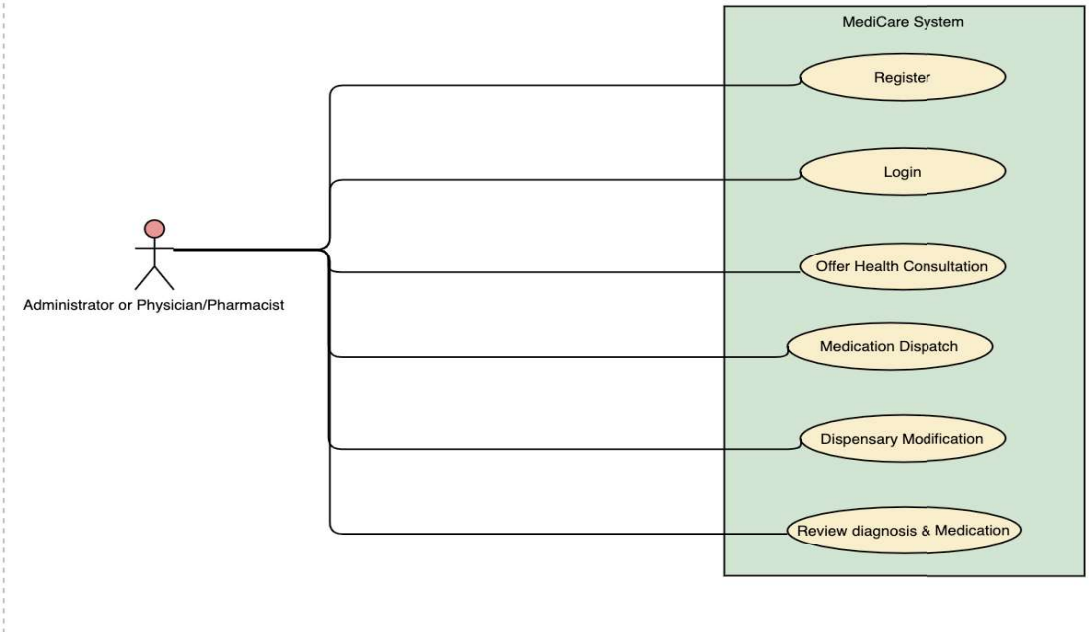


Fig 2.1 Use Case Diagram for Administrators

Fig 2.1 shows the administrator's interaction with the MediCare System. The administrator must register as well into the system and log in as a physician or pharmacist. The physician then interacts with a patient via the health consultation unit with either video conferencing or chat room feature. They then make a diagnosis that redirects them to the dispensary unit, where the pharmacist assesses a drug's availability and dispatches the drug. The physician can view patient data, including diagnosis and medication, and pharmacists can modify the dispensary unit.

Chapter 3: Architecture and Design

3.1 Introduction to System Architecture

This chapter outlines the system architecture and design for MediCare, a web-based stay home medical assistant. It gives a detailed description of the various components brought together to develop the system and the different elements that make up this system. The MediCare system adopts the use of a three-tier architecture in its development. The system architecture components include a front-end or user interface view, also known as the client presentation tier, a functional and logical tier known as the back-end operational interface, including a web server and the database tier.

3.1.1 Assumptions and Constraints

- Users of the system are not limited to the aged and mobility impaired but are the main focus of the system's use.
- Users have access to internet connectivity.
- Set number of physicians at Rhema Rapha Medical Center to be registered by the systems' developer.
- Users are middle to upper-class populations with basic vital recording equipment such as a thermometer, weighing scale, electronic body mass index monitor, and or sphygmomanometer, commonly known as a blood pressure monitoring device.
- A standard delivery fee of GHC 40.00 to patients beyond Tse Addo and surrounding neighborhoods and a standard consultation fee of GHC 50.00 using the Medicare system.

3.2 System Overview

The system overview illustrates the pictorial interaction of the different interfaces of the system. Once a user logs into the system, their details are stored in the MySQL database

in phpMyAdmin. In addition to this information, all other data collected from the patient, such as vital information, is stored in the database with basic encryption for user passwords and payment information. User requests are made from the client presentation tier and transferred to the logical tier that processes the request and collects information from the database. It then transmits this information through the logical tier and responds to the user's request on the client presentation tier.

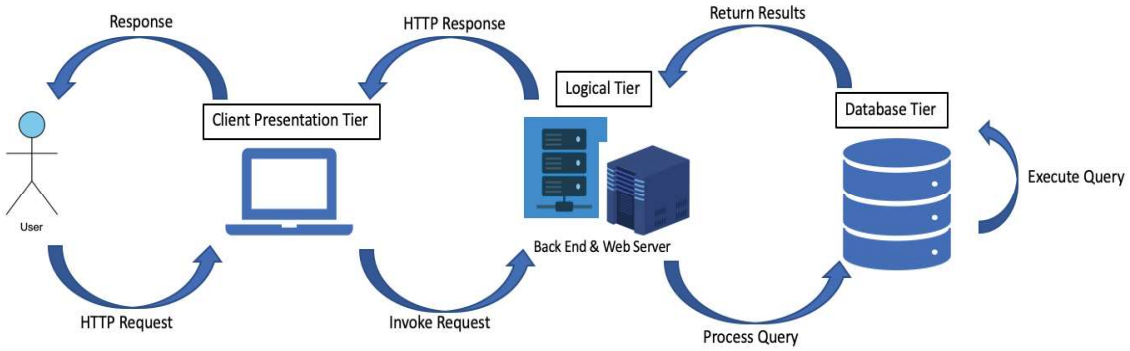


Fig 3.0 3-Tier System Architecture Relation

3.2.1 Client Presentation Tier

The client presentation tier for MediCare serves as the interactive forum for all stakeholders. It poses as the point of contact for users to interact with the system and view information while performing all relevant actions that affect the database. The user places a request using this interface and receives responses retrieved from the logical and database tiers.

3.2.2 Logical Tier

The logical tier or component contains all back-end operations that carry out processes and actions based on the user's request. It connects to the database and performs

logical operations to produce information for users. It also enacts the reasoning behind each process on the client presentation tier.

3.2.3 Database Tier

The database tier is the storage facility of the system. It contains all necessary and relevant information needed by stakeholders of the system. It stores user credentials and vital information for physicians to use during diagnosis. It also includes an inventory of available medication for delivery to patients.

3.3 Design

MediCare application adopts the Model View Controller (MVC) design pattern [23]. The MVC pattern is an architectural design pattern that splits a system's components into three major phases. The model phase is recognized as the form of the data. A class is typically considered an example of a model object. This phase stores data retrieved from the database. The view phase acts as the user interaction point. The view displays information from the model and creates room for modification to this information. It also serves as a means for the user to communicate with the model and controller. The controller handles or controls all user requests. The user makes an HTTP request on the view end, and the controller processes and returns a response. The controller operates between the view and model as a logical business end. This project utilizes the MVC design pattern to increase the developmental speed by separating interface development. Since the project is a web-based application, MVC and client-server architecture are natural architectural patterns to employ. However, due to the precision of the MVC design pattern, it is exclusively adopted for architectural design in this project.

3.3.1 Sequence Diagrams

The sequence diagrams are interaction diagrams that display how operations by the user are carried out. They visually explain the communication between the components of a system.

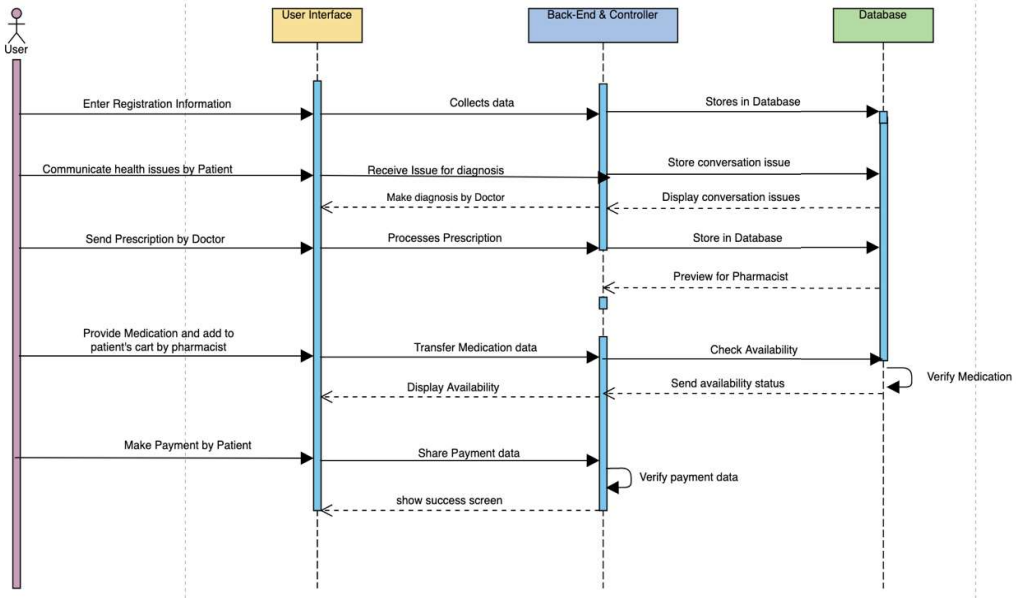


Fig 3.1 Overview Sequence Diagram for MediCare Users.

3.3.2 Activity Diagram

The activity diagram is an advanced version of a flow chart that displays the flow of interactions and information on the user's activities to achieve a specific goal.

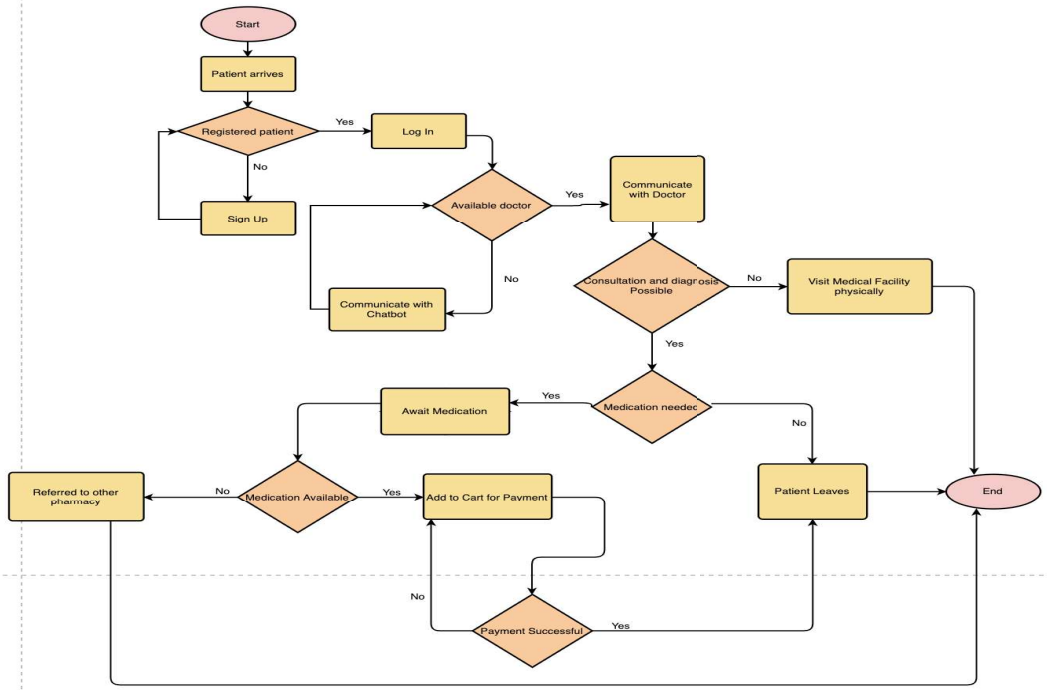


Fig 3.2 General Flowchart Activity Diagram for MediCare patient.

3.3.2.1 Login and Sign Up

The login and Sign Up process requires that first-timer users register or sign up with pertinent details necessary to utilize the MediCare system's functionalities. after signing up, they must log in to identify correct credentials to enforce security. All users of the system interact with the landing page, which offers a brief introduction to the system before being asked to register or log in to explore more. In the registration process, the username and email provided are verified against existing ones in the database to avoid duplicates and redundancy while enforcing unique patients' or physicians' experience with the system. If the details provided are unique, they are appended to the database. If a duplicate is detected, they are alerted to either log in as that user since the details already exist or to change the email address and or username.

Once a user logs in, their credentials are compared with those already stored in the database. Once there is a match, login is approved, and users can utilize other functionalities of

MediCare. If a match does not exist, they are alerted of the error in logging in and can either try again, register again, or utilize the 'forgot password or email' functionality.

3.3.3 Entity Relationship Diagram

The entity-relationship diagram gives a detailed and general overview of the entities and their relationship as stored in a database. The structure shows a visual representation of the tables used in storing information and operating the MediCare system. It also includes the relationship between these tables and further details of each table.

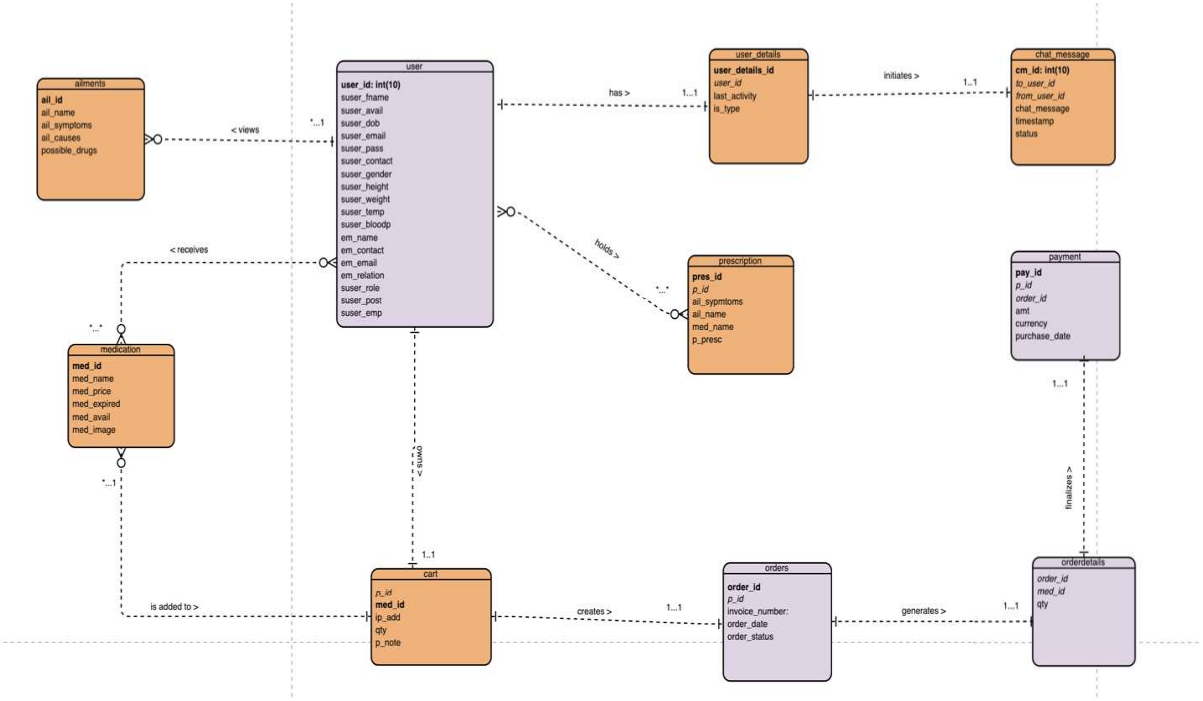


Fig 3.3 Entity-relationship Diagram & database structure for MediCare system.

Chapter 4: Implementation

4.1 Tools and Technology

The tools and technology used in developing this application are listed below in two parts: The Front-end tools and the Back-end tools.

4.1.1 Front-End

Front end in web development is the use of HTML, CSS, and JavaScript to design an interactive and easy-to-use interface for users of a web system [24].

- **Bootstrap framework -** The bootstrap framework is a free front-end framework that consists of hypertext markup language and cascading style sheet designs used to create responsive and presentable interfaces [25].
- **HTML -** Hypertext Markup Language is a basic language used in defining content on web pages [26].
- **CSS -** Cascading Style Sheet is a language used in conjunction with HTML to determine the styles and appearance of content on a webpage [27].
- **JS -** JavaScript is a scripting language that enhances the appearance of web pages by including interactivity to the webpage while describing how content should function and behave [28].

4.1.2 Back-End

Back-end web development is centered on server-side implementation, determining how exactly the system functions in its communication between the front end and the database.

- **PHP & cURL -** PHP is a programming language used to carry out back-end interactions between the database and the system. cURL (client URL) [29], on the other hand, is a command-line tool for receiving and retrieving files using URL syntax. In this project,

it is used in conjunction with PHP to return an external webpage through a 'GET' request by the user.

- **jQuery -** jQuery [30] is a lightweight JavaScript library that enhances and simplifies the use of JavaScript through reduced semantics. In this project, jQuery is used to call multiple JavaScript lines of code in one method singly.
- **AJAX and JSON -** Ajax [31] allow quick incremental updates to a webpage without reloading the webpage entirely. This increases the speed of an application and its responsivity to user requests. JSON [32] is a lightweight data-interchange format that sends data from a client-server to an application server and vice versa.
- **Regular Expression -** RegExp [33] is a pattern arrangement used to match character combinations in a string. In this project, Regular Expression is used to verify specific data input of users.
- **API's -** An application programming interface is a bridge between two software interfaces that enable communication between them [34]. In this project, the Pay Stack API is used to implement external payment within the Medicare system. The WebRTC API utilizes the scotch.io interface to enable video conferencing.
- **HubSpot Interface -** Hubspot is a developer and marketing software kit that allows developers to implement already built bot interfaces to minimize workload and personalize conversations with system users [35].
- **MySQL -** MySQL is the most popular open-source relational SQL database management system [36]. In this project, MySQL is used as the database for storing all relevant system data through the administration unit of phpMyAdmin.

4.2 Project File Structure

The file structure below is an overview of the Medicare system built-in mainly PHP and other basic web development languages such as CSS, JS, and HTML. This system uses the Model View Controller layout to facilitate easy iterations and description of system flow.

- ***Medicare***
  - *Admin*
  - *Assets*
    - *Img*
    - *Medi_img*
    - *Pro_img*
  - *Assets*
  - *Classes*
  - *Controllers*
  - *CSS*
  - *Function*
  - *JS*
  - *Login*
  - *Settings*
  - *View*
  - *Index.php*

The model aspect of the project is necessary for handling queries that affect the database. If a user should register into the system, the 'add_new_sys_user' SQL query in the database is implemented in the model aspect, represented by the 'classes' layout below. These models are grouped into four files, each affecting different parts of the database to improve organization.

- ***Model***
  - *Classes*
    - *Cart_class.php*
    - *Chat_class.php*
    - *Consultation_class.php*
    - *User_class.php*

The 'Cart class' handles all queries related to the cart table for users. The chat class handles all queries related to the chat table and functions, including sending a chat message from one user to another, reading or opening a chat message, and many more. The consultation class contains queries that add prescriptions, create a diagnosis, add medications, and other

essential executions to the database tables used by doctors and pharmacists only. Finally, the user class executes all queries related to system users such as register, login, display user details, and others.

- ***View***
  - ***Admin***
    - *Add_ail.php*
    - *Add_med.php*
    - *contact.php*
    - *Edit_med.php*
  - ***View***
    - *About.php*
    - *Cart.php*
    - *Checkout.php*
    - *Disease.php*
    - *Doc.php*
    - *Med_det.php*
    - *Med.php*
    - *Pay_success.php*
    - *Pres_det.php*
    - *Pres.php*
    - *Profle.php*
    - *Terms.php*
    - *V_conf.php*
    - *Index.php*

The view displays the user interface and HTML pages users interact with. The admin view allows only administrators such as pharmacists and doctors to add and edit medication logs and the ailment data that gives users access to potential ailment information. The main view unit displays pages all users may access except 'cart.php,' which is accessible by patients only. The view also determines the organization and layout of data from the model.

- ***Controllers***
  - *Cart_controller*
  - *Chat_controller.php*
  - *Consultation_controller.php*
  - *User_controller.php*

The controllers represent the request handlers of the user. The controllers act as the connection between the model containing database queries and the view for displaying data. For example, suppose a user places a request from the view end. In that case, this request is passed through the controller to carry out necessary checks, such as if a user is logged in

and if the user has permission to access the page before communicating with the model to produce relevant data.

Other necessary components of this project include the function folder that handles back-end implementation for controllers,

and the settings folder that contains database connection details such as the classes used in establishing the database connection and running queries, the credentials used to create the connection between the external database administration tool (phpMyAdmin) and the project file, and 'core.php', which contains functions that determine what session variables may be used for.

4.3 User Interface and Functionality

This section displays and highlights the implementation process of each interface and its functionality. The user interface is the point of interaction with the system by users. The main user of this system is a predefined number of doctors and pharmacists and a vast number of patients. Each user has a different set of functionalities. However, there are similarities in the design to encourage uniformity and familiarity with the system.

4.3.1 Landing Page

The landing page is the first interface a user interacts with upon accessing the MediCare web system. On this page, a user is introduced to the services offered by Medicare and the hospital's contact information and a preview of the external chatbot.

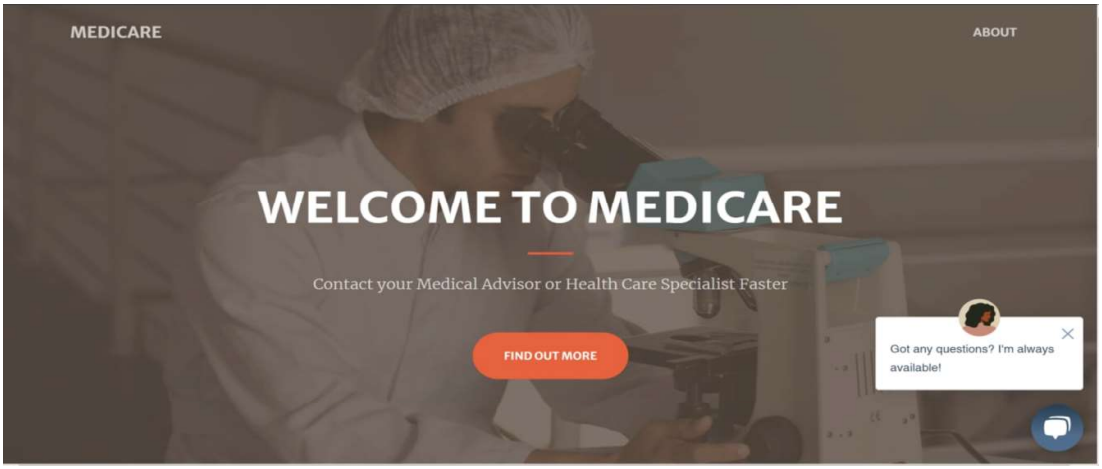


Fig 4.0 Landing Page for MediCare user.

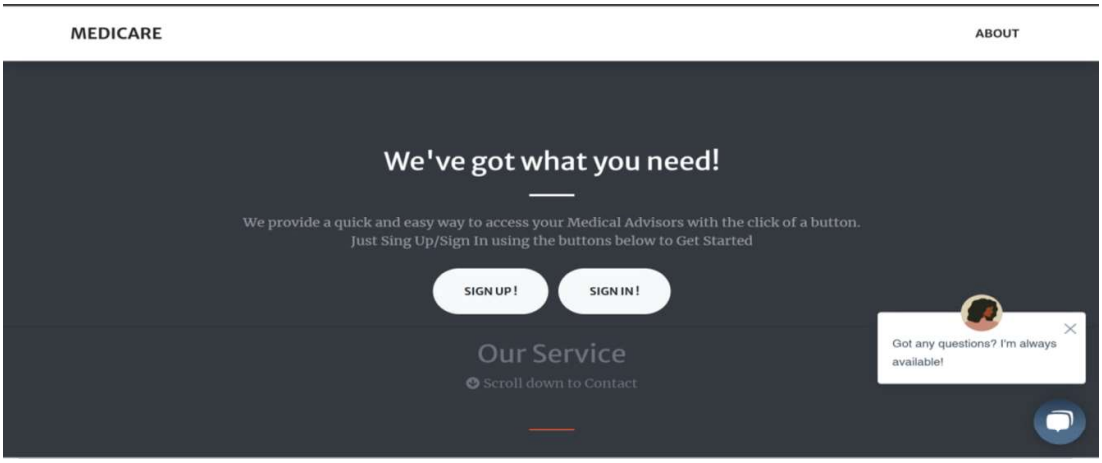


Fig 4.1 Landing Page for MediCare user.

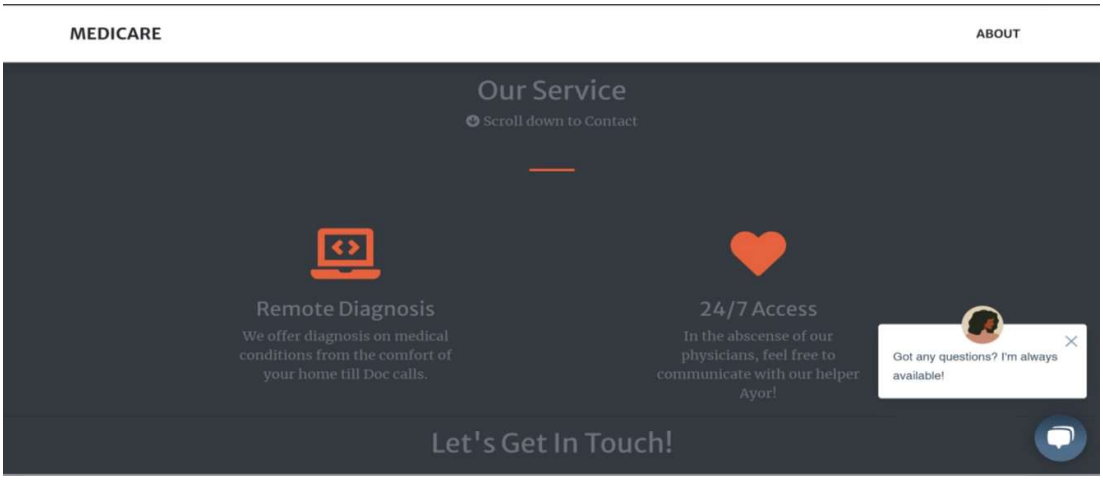


Fig 4.2 Services on landing page for MediCare user.

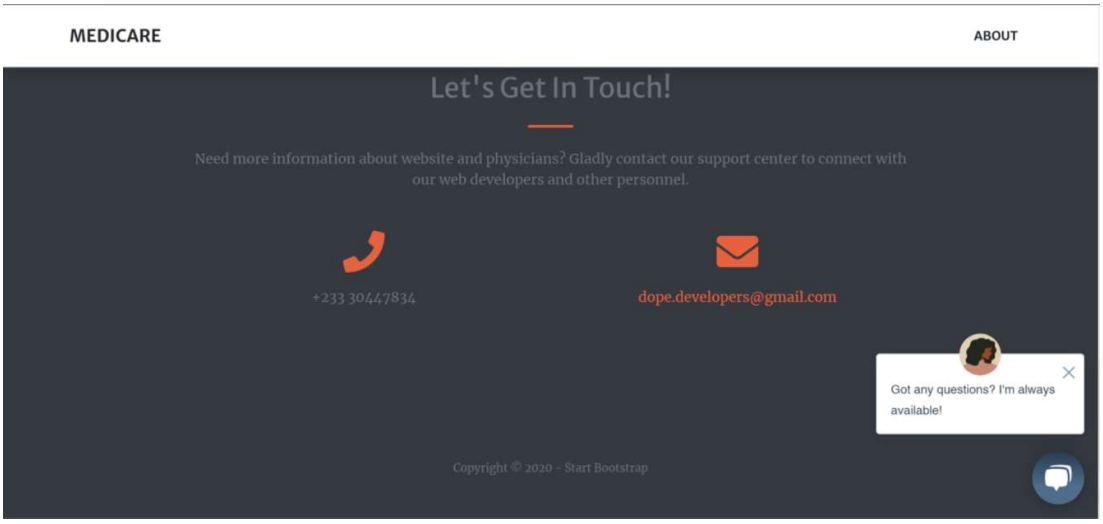


Fig 4.3 Contact on landing page for MediCare user.

4.3.2 Register

In the registration section, a user fills in the required data before registration is successful. This accomplishing system functional requirement FR01 (Sign in and Sign Up).

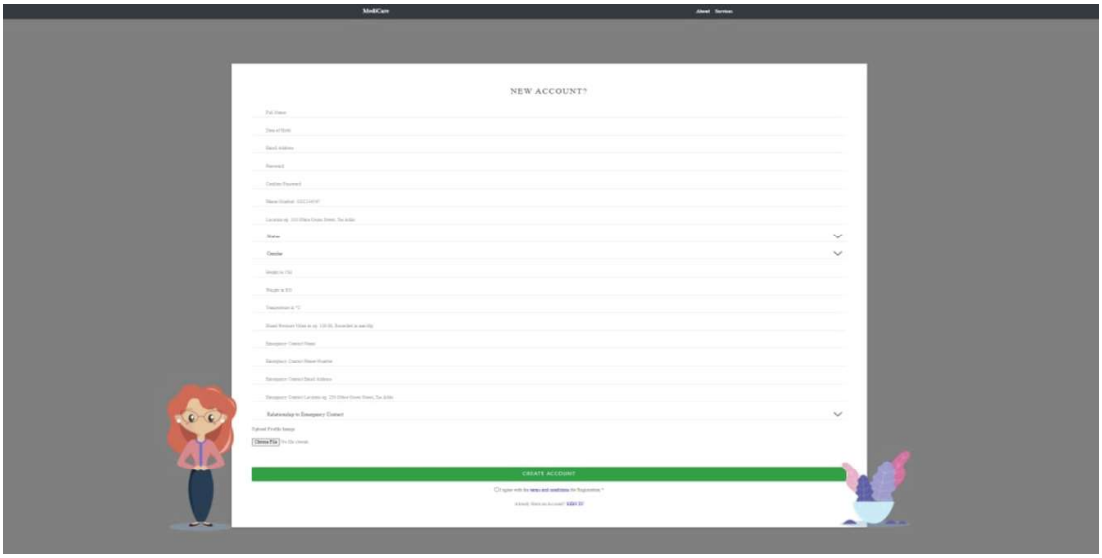


Fig 4.4 Register Page for MediCare user.

Since there is a predetermined number of doctors and pharmacists, Doctors and pharmacists register through the same registration form. However, a 'user_role' is set in the database to provide access to certain functionalities by different users, and a session variable is created

in the loginprocess.php to store the user's role throughout their interaction from when they sign in to when they sign out to monitor access and function is created to check this access. The user_role also distinguishes users with 1 representing doctors, 2 representing patients, and 3 representing pharmacists. However, every new user is automatically set to 2 in the database till the developer makes the change.

```php
//function to check for permission
function check_user_perm(){

    //get session role
    if (isset($_SESSION['user_role'])) {

        //assign session to an array
        $uperm = $_SESSION['user_role'];
        if ($uperm == 3) {
            //return role number
            return 3;
        } elseif ($uperm == 2) {
            return 2;
        } else {
            return 1;
        }
    }
}
```

Fig 4.5 Code snippet of user authentication per role.

For registration to be processed as successful, the user is required to enter valid details. In this section, front-end validation is performed using basic HTML validation types for fields such as "type='email' or 'password,'" a minimum and maximum date limit for date of birth registration, amongst others.

```html
<div class="form-group">
    <input type="text" class="form-control" id="udob" placeholder="Date of Birth" name="udob" min="1940-12-31" max="2005-12-31" onfocus="(this.type='date')" required>
</div>
<div class="form-group">
    <input type="email" class="form-control" id="umail" placeholder="Email Address" name="umail" onchange="validatemail(this.value);" required>
</div>
<div class="form-group">
    <input type="password" class="form-control" id="p_pass1" minlength="6" placeholder="Password" name="u_pass1" required>
</div>
<div class="form-group">
    <input type="password" class="form-control" id="p_pass2" minlength="6" placeholder="Confirm Password" name="u_pass2" onkeyup="checkPass(); return false;" required>
</div>
```

Fig 4.6 Code snippet of front-end input data validation.

Regular expression using JavaScript is used to validate field data such as phone number, name, email address. JavaScript is also used to check if the user input value for the password matches the input value to confirm the password to ensure the user is aware of the password they are setting and reduces the likelihood of a forgotten password.

```
function checkPass() {
    //Store the password field objects into variables ...
    var pass1 = document.getElementById('p_pass1');
    var pass2 = document.getElementById('p_pass2');
    //Store the Confimation Message Object ...
    var message = document.getElementById('confirmMessage');
    //Set the colors we will be using ...
    var success = "#66cc66";
    var fail = "#ff6666";

    //Compare the values in the password field
    //and the confirmation field
    if (pass1.value == pass2.value) {
        //passwords match

        pass2.style.backgroundColor = success;
        message.style.color = sucess;
        message.innerHTML = "Successful Match"
    } else {
        //passwords do not match.

        //notify the user.
        pass2.style.backgroundColor = fail;
        message.style.color = fail;
        message.innerHTML = "Error! Not a Match!"
    }
}

function validatephone(ucontact) {
    var num = '';
    var numval = ucontact.value
    if (numval.charAt(0) == '+') {
        var num = '';
    }
    curphonevar = numval.replace(/[\\A-Za-z!"£$%^&\,*+_={};:'@#~,.Ś\/<>?|`¬\]\[]/g, ' ');
    phone.value = maintainplus + curphonevar;
    var maintainplus = '';
    phone.focus;
}
// validates text only
function validatefname(ufname) {
    ufname.value = ufname.value.replace(/[^a-zA-Z-'\n\r.]+/g, ' ');
}
```

Fig 4.7 Code snippet of regular expression data validation.

To check if a user exists, a query within a function in the model (classes) that affects the database is created to get user data that must be unique, precisely email address.

```
public function check_existing_sys_user($uemail){
    //sql query
    $sql = "SELECT `suser_email` FROM `user` WHERE `suser_email` = '$uemail'";

    //return the executed query
    return $this->db_query($sql);
}
```

Fig 4.8 Code snippet of query to verify existing user email.

An instance of the class is then created in the controller that implements or invokes the query and function to be used on other pages.

```php
//takes username
function check_existing_sys_user_fxn($uemail){
    //create an instance
    $newuserObject = new user_class();


    //run the return method
    $return_usern = $newuserObject->check_existing_sys_user($uemail);

    if ($return_usern){
        $existing_usern = $newuserObject->db_fetch();

        return $existing_usern;
    }else{
        return false;
    }
}
```

Fig 4.9 Code snippet of controller calling 'check_existing_sys_user' query function.

In the registerprocess.php, the usable controller function is called on the user's data by requiring the controller's file. After the requirement is made and the user data is gathered from the form, the function to check if a user exists by their 'email' is called on the email data input. Once this data matches, the user is prompted that 'User Already Exits.' As long as the match is not successful, the password is encrypted using the 'md5()' built-in function in PHP, and the user profile image and other data are uploaded to the database. If the image already exists in the database, an alert indicating 'File already Exists' is activated, requiring the user to change the image. No field data can be left empty during registration. This check is made using the required term in HTML. Images that are not of type 'JPG, JPEG, PNG & GIF' are not allowed into the system.

```php
$existing_mail = check_existing_sys_user_fxn($umail);


if (!empty($existing_mail)) {

    <script type="text/javascript">
        alert("User Already Exist");
    </script>
<?php
}

$p_pass = md5($upass1);

$aim_dir = "../assets/pro_img/";
$aim_file = $aim_dir . basename($_FILES['p_img']["name"]);
// $aim_file = "../p_imgs/".basename($_FILES['p_img']["name"]);
$image_type = strtolower(pathinfo($aim_file, PATHINFO_EXTENSION));
```

Fig 4.10 Code snippet to check user data, encrypt password and add user image.

```
    //limit file type
    if ($image_type != "jpg" && $image_type != "png" && $image_type != "jpeg" && $image_type != "gif") {
    ?>
        <script type="text/javascript">
            alert("Apologies, only JPG, JPEG, PNG & GIF Files are Allowed.");
        </script>
    <?php
    }
}
$add_image = move_uploaded_file($_FILES["p_img"]["tmp_name"], $aim_file);
```

Fig 4.11 Code snippet to check image upload type.

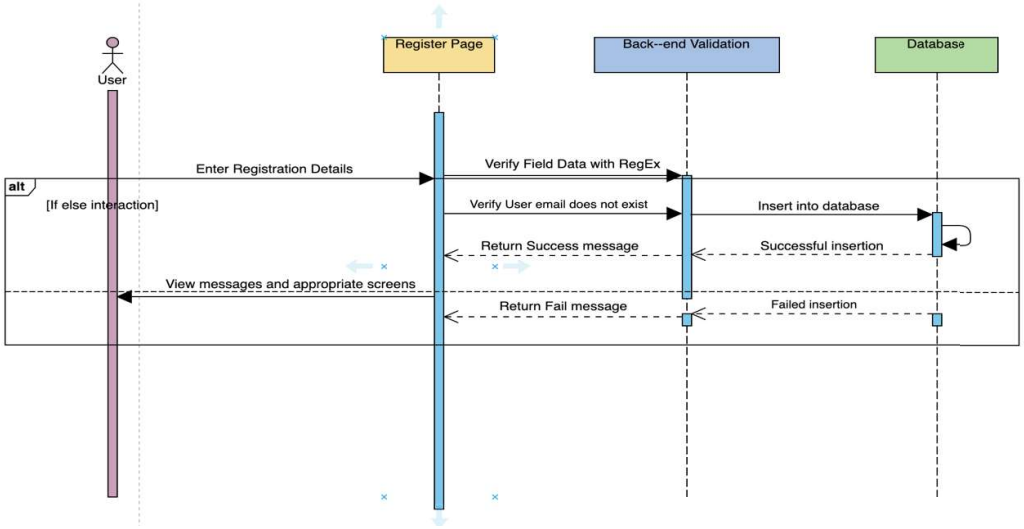


Fig 4.12 User Registration Sequence Diagram

4.3.3 Login

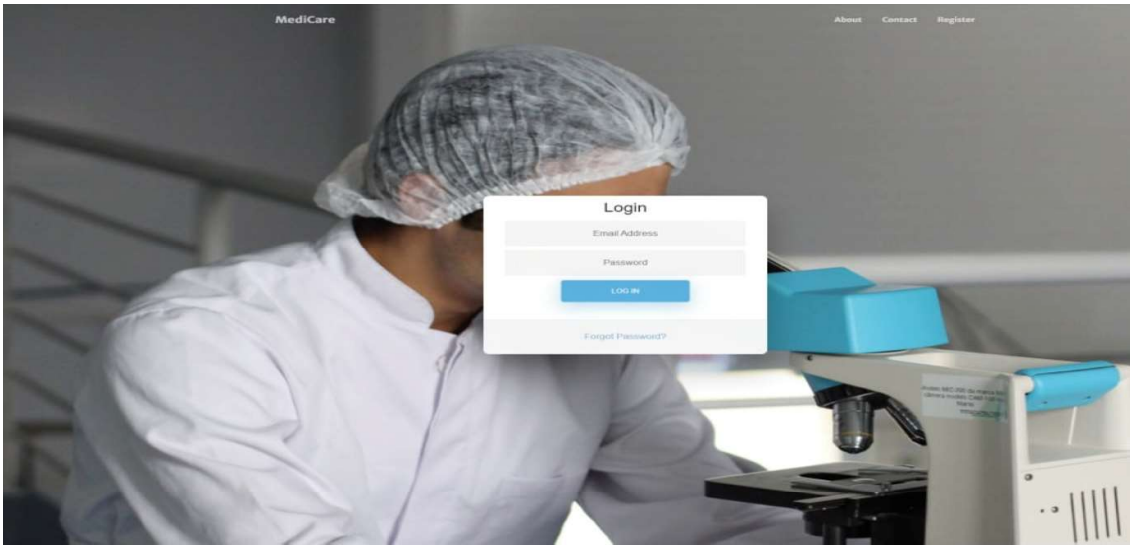
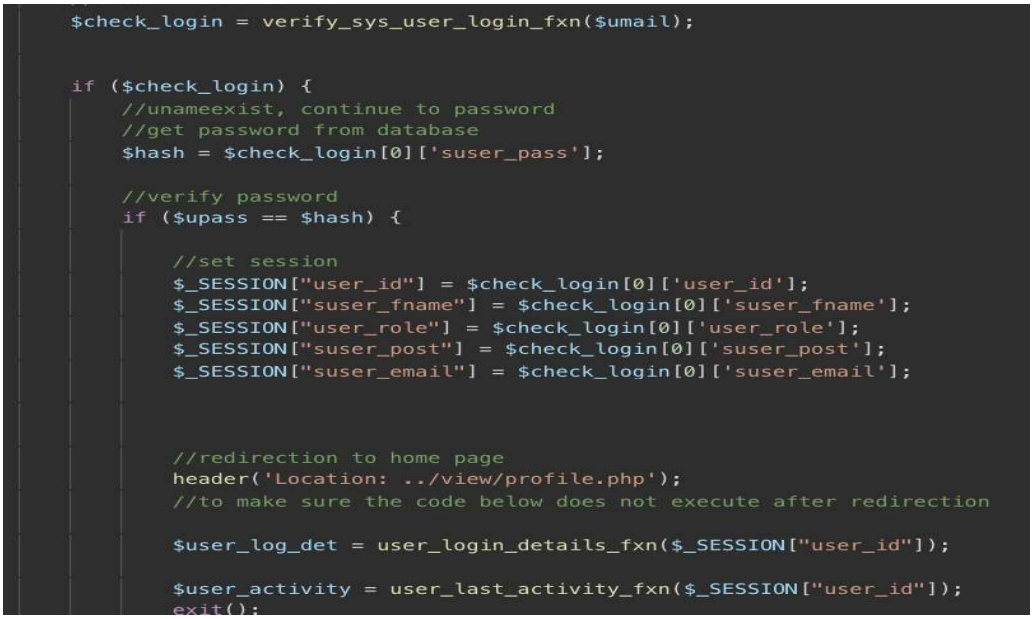


Fig 4.13 User Login Page.

To log in to Medicare, a user must provide a valid and existing email address that matches the password. Once user login is successful, they are redirected to their profile page.

The password entered by the user on the login page is encrypted both at the front-end using HTML ‘password’ type for the input field and in the back-end using ‘md5()’ to achieve the predetermined nonfunctional system requirement NFR03 (security requirements). Verification of password is done where the ‘user_email’ entered matches a ‘user_email’ in the database. In this light, verifying user email is performed first. A function containing the database query to select ‘user_email’ is created and called in the controller to be used in the loginprocess.php for the verification.



```
$check_login = verify_sys_user_login_fxn($umail);

if ($check_login) {
    //unameexist, continue to password
    //get password from database
    $hash = $check_login[0]['suser_pass'];

    //verify password
    if ($upass == $hash) {

        //set session
        $_SESSION["user_id"] = $check_login[0]['user_id'];
        $_SESSION["suser_fname"] = $check_login[0]['suser_fname'];
        $_SESSION["user_role"] = $check_login[0]['user_role'];
        $_SESSION["suser_post"] = $check_login[0]['suser_post'];
        $_SESSION["suser_email"] = $check_login[0]['suser_email'];



        //redirection to home page
        header('Location: ../view/profile.php');
        //to make sure the code below does not execute after redirection

        $user_log_det = user_login_details_fxn($_SESSION["user_id"]);

        $user_activity = user_last_activity_fxn($_SESSION["user_id"]);
        exit();
```

Fig 4.14 Code snippet of login process comparing passwords and setting SESSION variables.

After requiring the necessary files, the match for the email address is carried out by calling the controller function to verify the email address on the user's input data. Once the emails check out, a variable is created to call the encrypted password in the database that corresponds with the verified user email address. Then the comparison is made between the encrypted password in the database and the encrypted password provided by the user trying to log in. If the passwords match, the session variables are created for the user's id, full

name, role number, position (doctor, patient, pharmacist, which is set automatically as patient and later changed in the database by the system developer for respective users), and email address to be utilized later on through system interaction. Finally, the user is redirected to their profile page. If the user's email matches, yet the password is wrong, the user is prompted by the error with their password. If none match, the user is alerted that their data is incorrect, implying they do not exist or have entered inaccurate data.

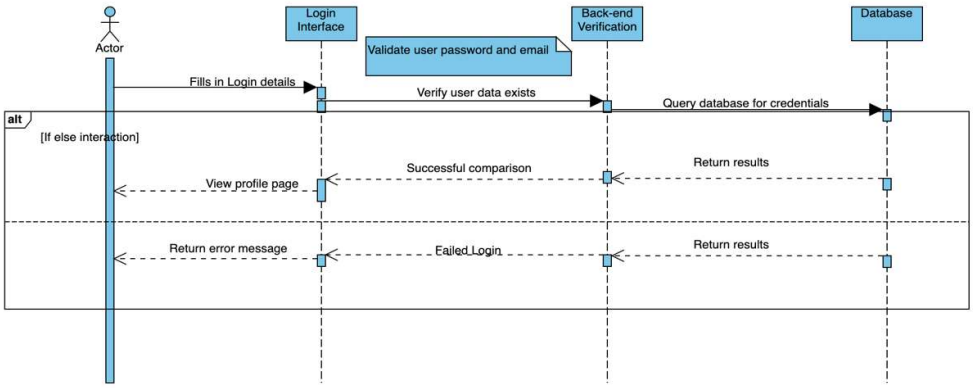


Fig 4.15 User Login Sequence diagram

4.3.4 Profile

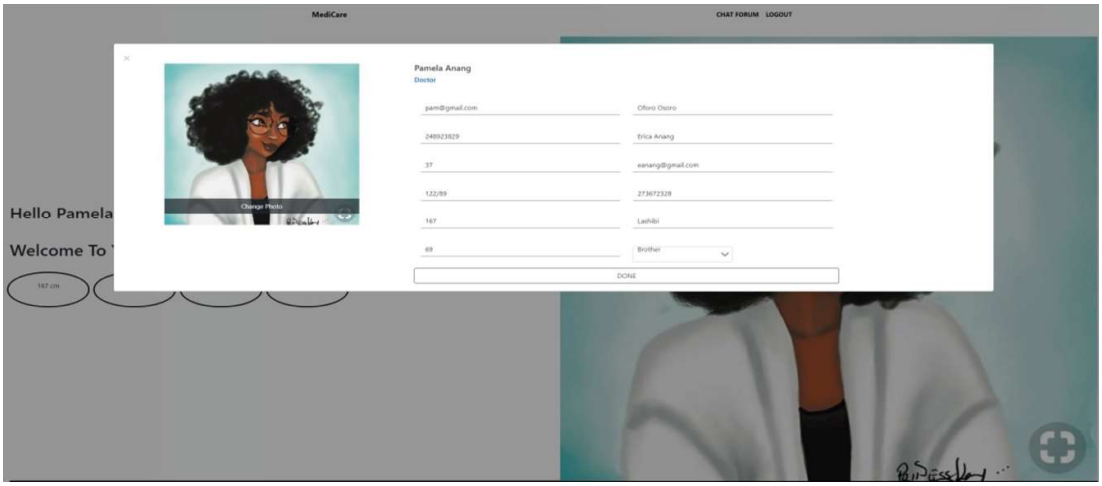


Fig 4.16 User Profile Page with Edit Profile Modal opened.

The profile page of each user displays their image, a welcome message, and essential vitals data. This is done by using a select query from the database in the model to select user data and display it on the front-end.

```php
<?php
$user_info = view_one_sys_user_fxn($user_id);

?>

<div class="container-fluid">
  <div class="row">
    <div class="col-sm-12 col-md-6 mx-auto my-auto position-relative">
      <p class="position-absolute top-50 start-50 translate-middle">
      <h1>
        Hello <?= $user_info['suser_fname'] ?> <br><br> Welcome To Your Profile
      </h1>
<br>
```

Fig 4.17 Code Snippet to view user profile data

On the profile page, users can edit their details by selecting the 'Edit Profile' button at the bottom of the page. The edit profile screen is implemented using a CSS/JavaScript modal element. CSS is used to specify the appearance of the modal, while JavaScript determines the animated dynamics and open/close features of the modal.

4.3.5 Chat Forum

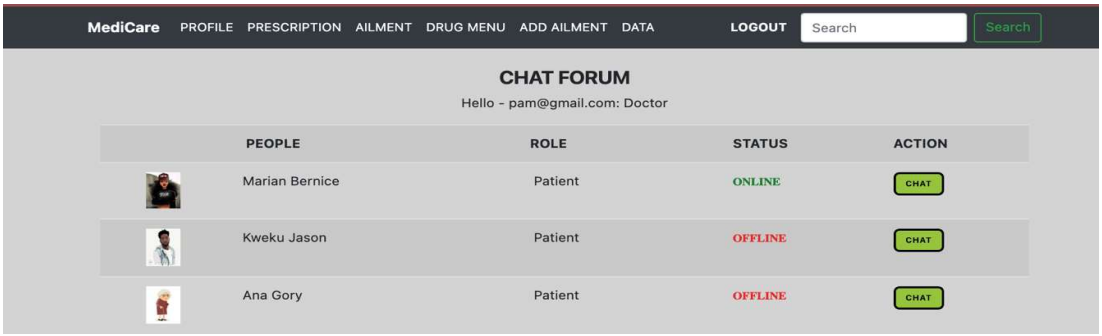


Fig 4.18 Chat Forum display for Doctors.

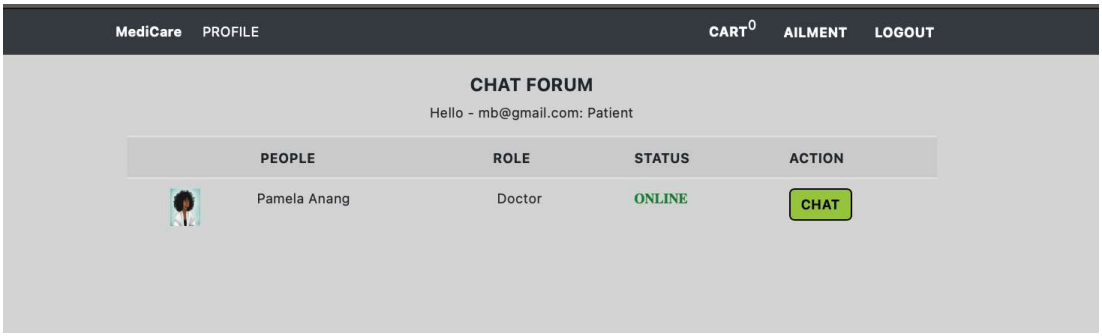


Fig 4.19 Chat Forum display for Patients similar to Pharmacists.

The chat forum is the most interactive view for all users. This page displays all system users to the doctor while pharmacists and patients are restricted to communicating with a doctor. The patient initiates a conversation with their doctor on their current situation, and the doctor requests vital data on this page. This page achieves functional requirement FR02 (health consultation). A patient provides their vital data to their doctor by recording the data from home using a basic home thermometer, weighing scale, and other basic data, then texts results to their doctor through the system.

Once the doctor is confident to make a diagnosis from home, an automatic prescription form pops up on his end, requesting he fills in the details to make the necessary diagnosis. The doctor then alerts the pharmacist from his end of the new prescription prompting the pharmacist to approve and offer essential medication. Once the medication is available and approved, the pharmacist alerts the doctor of the progress, with which he then informs the patient to make necessary payments through their cart.

In developing the chat room feature, jQuery is used to call the 'fetch_user()' function by sending an ajax request to the page that executes the SQL query to retrieve all system users in a table format. This also allows for frequent page information transfer without interfering with the web display. At various stages of chat room implementation, ajax calls are made to multiple pages. However, the user role is checked to select which users can communicate with each other.

```
<script>
    $(document).ready(function() {

        fetch_user();

        setInterval(function() {
            update_last_activity();
            fetch_user();
            update_chat_history_data();
        }, 1000);

        function fetch_user() {
            $.ajax({
                url: "../function/fetch_user.php",
                method: "POST",
                success: function(data) {
                    $('#user_details').html(data);
                }
            })
        }

        function update_last_activity() {
            $.ajax({
                url: "../function/update_last_activity.php",
                success: function() {

                }
            })
        }
```

Fig 4.20 Code snippet of jQuery 'fetch_user()' function and ajax call.

To detect the 'Online' or 'Offline' status of a user, an interval is set to update the users' last activity every second. The fetch user function is called to refresh the user data on the chat room page. To check the status of a user, the get user last activity function (which checks user login last activity inserted in the database using the datetime datatype) is called and compared with the most recent timestamp value in the date and time format of another user's activity. In Fig 4.21, if a user's activity happens to be less than the timestamp, it implies the user is 'OFFLINE' because their last login activity is less than any current login activity. If otherwise, the user is 'ONLINE.'

```
$status = '';

$current_timestamp = strtotime(date('Y-m-d H:i:s') . '-5000 second');
$current_timestamp = date('Y-m-d H:i:s', $current_timestamp);
$user_last_activity = get_user_last_activity_fxn($row['user_id']);

if ($user_last_activity < $current_timestamp) {

    $status = '<span style="color:red;font-family:sans;font-size:14px;" class = "label label-danger"><b>OFFLINE</b></span>';
} else {
    $status = '<span style="color:green;font-family:sans;font-size:14px;" class = "label label-success"><b>ONLINE</b></span>';
}
```

Fig 4.21 Code snippet implementing Login/Online status of a user.

To send chat data back and forth between users, JavaScript is used to design the chat box feature as a function which is called upon 'onclicking' the chat button next to a user's name. This function two user ids. That of the sender and that of the recipient. To facilitate communication, jQuery is used to get the first user's (sender's) input data. Then an ajax call is made to the insert chat function that adds to the 'chat_message' table in the database by collecting the sender's id and the recipient's id. The fetch chat history function displays all available messages to both users, allowing the recipient to view what has just been sent to them. The process repeats for an entire conversation between users.

```
$(document).on('click', '.send_chat', function() {
    var to_uid = $(this).attr('id');
    var chat_message = $('#chat_message_' + to_uid).val();
    $.ajax({
        url: "insert_chat.php",
        method: "POST",
        data: {
            to_uid: to_uid,
            chat_message_: chat_message
        },
        success: function(data) {
            $('#chat_message_' + to_uid).val('');
            $('#chat_history_' + to_uid).html(data);
        }
    });
});

function fetch_user_chat_history(to_uid) {
    $.ajax({
        url: "../function/fetch_chat_history.php",
        method: "POST",
        data: {
            to_uid: to_uid
        },
        success: function(data) {
            $('#chat_history_' + to_uid).html(data);
        }
    });
}
```

Fig 4.22 Code snippet to send and receive chat messages.

The read and unread status of a chat message is executed by counting the number of unseen messages in the database of status 1. Once a message is read, the status changes to 0.

4.3.6 Other Pages

This section discusses the implementation of other pages of the system that work alike. The use of a well-defined navigation bar on all page's aids in accomplishing NFR01 (usability requirements) ensuring easy use and navigation through the system.

4.3.6.1 Medication

The medication page is a catalog page that displays new medication added and keeps a record of drugs available. This page addresses the dispensary unit system requirement (FR03).

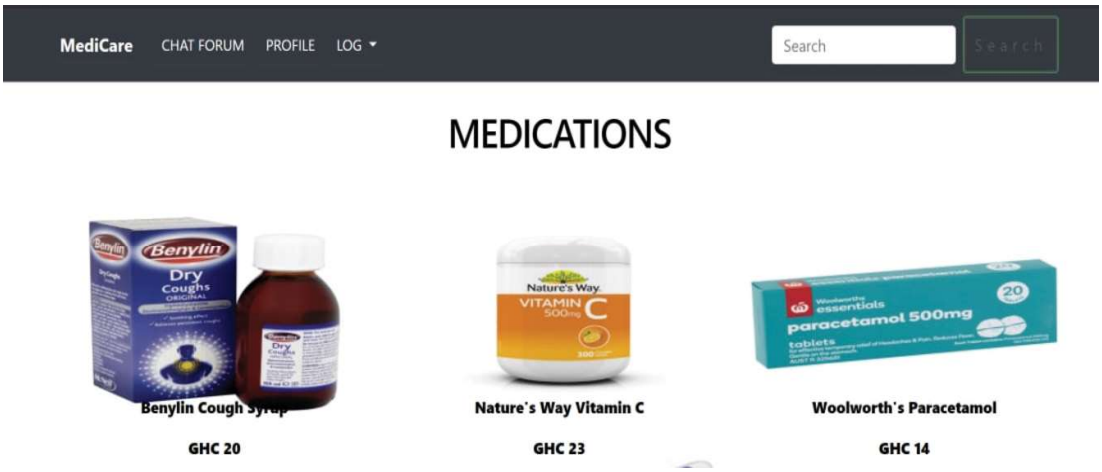


Fig 4.23 Medication Page.

The add and edit medication platform is accessible by pharmacists only, who are allowed to add medication and monitor the availability of each drug. On the other hand, doctors can view the drugs available only, while patients may not view any medication data. The editable features of a drug are its availability and price. The availability status of a drug is set as 2 automatically in the database upon addition. If the medicine becomes unavailable at any point, the pharmacist edits its status to 1, meaning 'out of stock.' Below is a code snippet showing how the user permission and availability status is checked before displaying a response to the appropriate user.

```
if ($user_perm == 3) {
    echo "<button type='button'  class='open-modal' id='myBtn' style='color:black' class='btn-dark'>EDIT</button>";
        if($status == 1){
            echo'<br>
            <br>
            <b> MEDICATION IS UNAVAILABLE</b>';
        }
} else if ($status == 2 && $user_perm == 1) {
    echo "";
} else {
    echo "<td><b>  SORRY MEDICATION NOT AVAILABLE</b></td>";
}
}
```

Fig 4.24 Code to check user's role and availability status of medication.

Physicians are allowed to search medications as well from the search feature in the navbar. A 'LIKE' query is created in the model class that selects all terms in the database with a similar character combination as the term entered in the search bar.

```
public function search_a_med($sterm){
    //a query to search product matching term
    $sql = "SELECT * FROM medication WHERE med_name LIKE '%$sterm%'";

    //execute the query and return boolean
    return $this->db_query($sql);
}
```

Fig 4.25 Code snippet of 'LIKE' query for search term.

```
$med_list;

 //check for search
 if (isset($_GET['search_t'])) {

    //get search term and store in variable
    $sterm = $_GET['term_search'];

    //run seach med function
    global $med_list;
    $med_list = search_med_function($sterm);
    }
```

Fig 4.26 Code snippet of search term processing.

4.3.6.2 Ailment

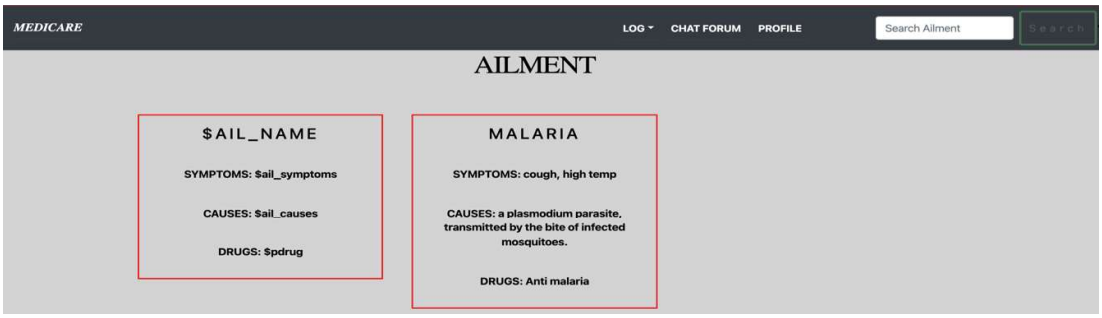


Fig 4.27 View Ailment Page.

The ailment page titled 'disease.php' displays information on diseases, symptoms, causes, and possible medication for patients. This page mainly offers information from first-hand expert physicians on possible ailments. It is, however, not a self-diagnosis unit for patients and provides more details on whatever diagnosis they may have received from their doctors.

Information on the ailment page is added and updated from the admin folder containing 'add_ail.php' and 'edit_ail.php.' These pages are accessible by all doctors only. This carries out FR04 by creating an administrator end for the physician. However, all system users are allowed to search for ailments at their convenience. If expert information has been provided on the specific disease, they may view further details using the search ailment field. The add ailment page view is similar to edit ailment and add drug (only accessible by pharmacists) page views.

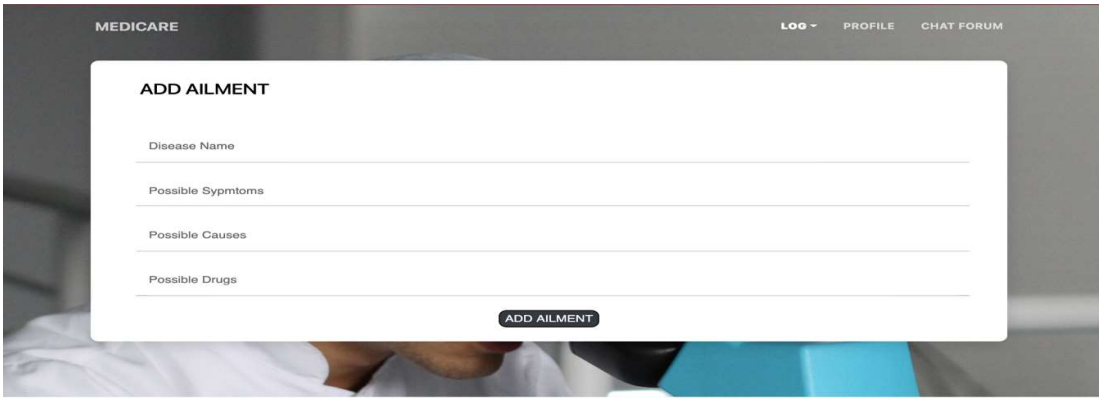


Fig 4.28 Add Ailment Page.

4.3.6.3 Prescription

The prescription page shows a view of all prescriptions ever written by the doctor and is only accessible by doctors and pharmacists. However, only a pharmacist can confirm a prescription and add medication to a patient's cart. Both doctors and pharmacists are allowed to search for an older prescription for reference purposes.

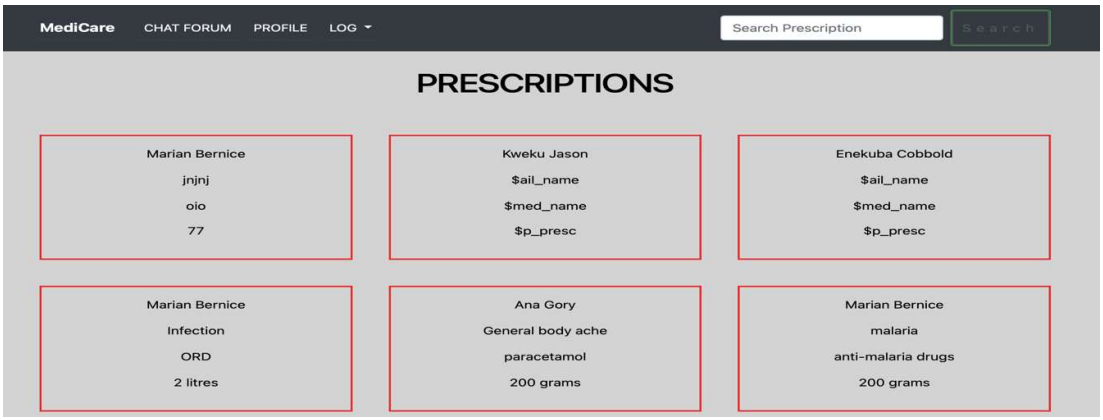


Fig 4.29 Prescription Log.

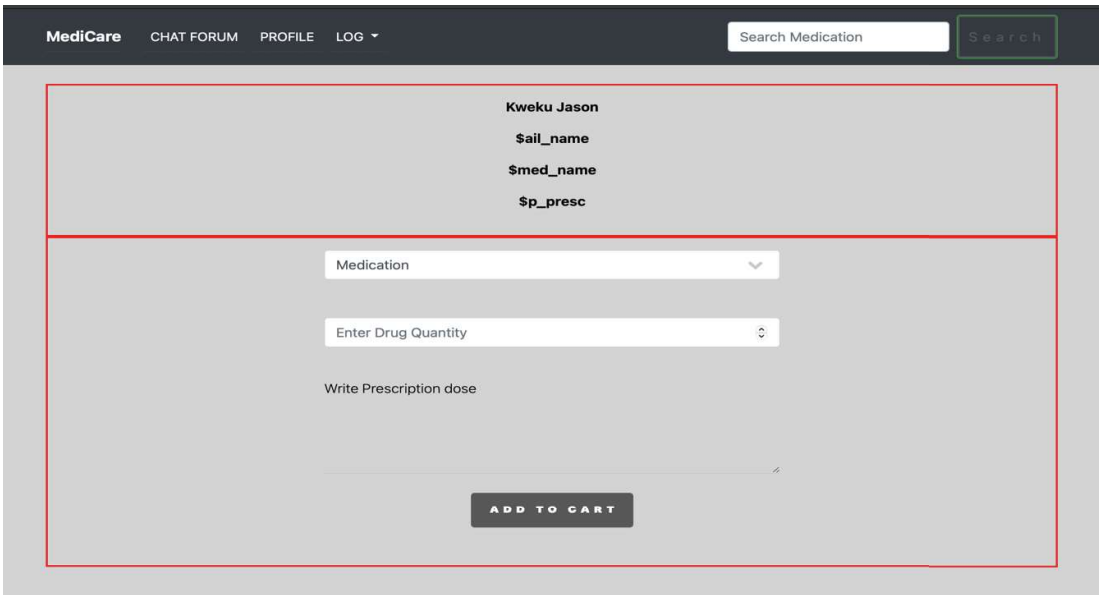


Fig 4.30 Add to cart prescription view

4.3.7 Cart & Payment

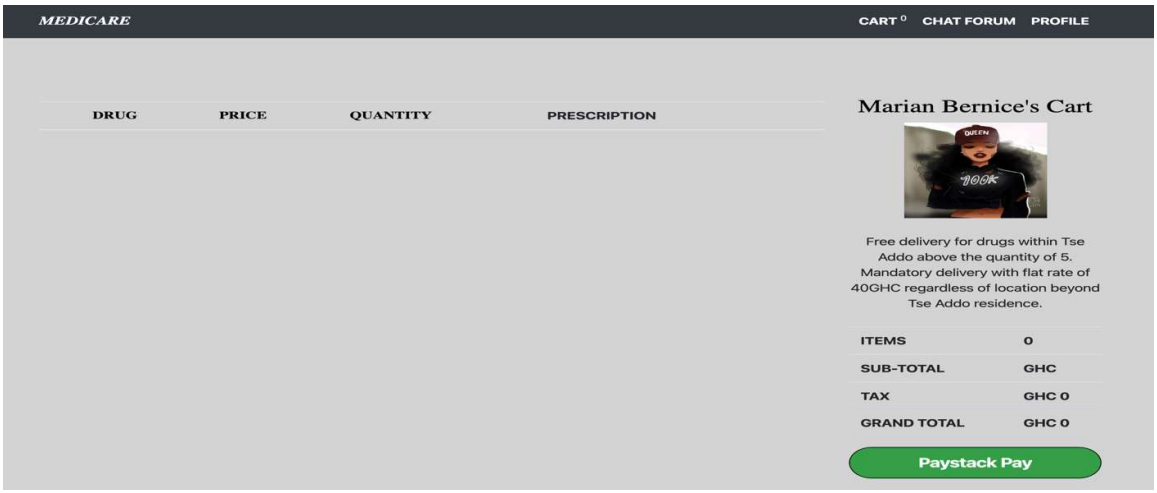


Fig 4.31 Empty cart/checkout deck of patients.

Adding to the cart is done by the pharmacist only. This fulfills system requirement FR06 (cart and payment). However, patients alone are allowed to view carts and make necessary payments. They are not permitted to adjust quantity but can remove items from the cart. Once the pharmacist confirms and supplies the drug, it is moved to the patient's cart. The 'id' of the patient is stored along with the prescription written by a doctor. When the pharmacist recommends the drug prescribed from a drop-down list of medications, the system collects the id stored with the prescription. It then executes the add-to-cart function using that user's id from the prescription form.

```
<form method="post" action="../function/add_to_cart.php" enctype="multipart/form-data">
    <div class="dropdown">
        <div class="md-form">
            <select style="width: 500px;" class="form-control" name="med_id" id="med_id" required>
                <option disabled selected>Medication</option>
                <?php

                if (!empty($med_ID)) {
                    foreach ($med_ID as $mid) {
                        $key = array_search($mid, $med_ID);
                        echo "<option value='" . $med_ID[$key] . "'>" . $med_name[$key] . "</option>";
                    }
                }

                ?>
            </select>
        </div>

        <input type='hidden' name="presid" value='<?php echo $pres_list['pres_id'] ?>'>
        <br><br>
        <input type='hidden' name="p_id" value='<?php echo $pres_list['p_id'] ?>'>

        <div class="form-group">
            <input type="number" class="form-control" style="width: 500px;" type='number' name="qty" placeholder='Enter Drug Quantity' required>
        </div><br>
        <textarea style="color:white; width: 500px;" name="p_note" id="p_note" rows="5" placeholder="Write Prescription dose" required></textarea><br>
        <input type="submit" name="add_cart" value="ADD TO CART" class="primary" />
        </strong>
</form>
```

Fig 4.32 Code snippet of add to cart front end implementation

The medication list is created by looping through medication id's stored in an array from data in the database. Then searching through this array where the medication id matches a key item and returns the key-value id and key-value name in the drop-down list.

```php
<?php
include_once("../controllers/consultation_controller.php");

if (isset($_POST['add_cart'])) {
    // grad user form data
    $pid = $_POST['p_id'];
    $ipadd = '::1';
    $med_id = $_POST['med_id'];
    $qty = $_POST['qty'];
    $p_note = $_POST['p_note'];

    //run duplicate function
    $double_cart = check_duplicate_function($med_id, $pid);

    // //check if there is a duplicate
    if ($double_cart) {
        //echo duplicate

?>
        <script type="text/javascript">
            alert("Oops! Already In Patiest's Cart");
            window.location.href = "../view/doc.php";
        </script>
        <?php
    } else {
        //run add to cart function
        $add_to_cart = add_cart_function($pid, $med_id,  $ipadd,  $qty, $p_note);

        //check if add to cart worked
        if ($add_to_cart) {
        ?>
            <script type="text/javascript">
                alert("New Item Added To Patient's Cart");
                window.location.href = "../view/doc.php";
            </script>
        <?php
```

Fig 4.33 Code snippet showing add to cart back end implementation.

Payment of medication is made by patients only. The pay stack API is used to implement this feature. The patient initializes payment. Once a patient hits the paystack button. A call is made to the JavaScript function 'paywithPaystack()' that processes the email address, currency, and amount displayed in the cart using jQuery. After the data is processed, the API call is made to the paystack server that collects this data and displays a payment section for the user to choose payment details and confirm payment. If a user closes this section, they are informed they canceled the payment process. However, if they confirm payment and payment is successful, the 'callback' function displays the transactions reference id and passes this to the verify_payment.php page. In the verify.php page, the reference id is retrieved from the URL using the global 'GET' variable. If the reference id is empty, the user is redirected to the previous page using the javascript syntax below.

```php
<?php
  $ref = $_GET['reference'];
  $curl = curl_init();

  if(!$ref){
    header('Location: javascript://history.go(-1)');
  }

  curl_setopt_array($curl, array(
    CURLOPT_URL => "https://api.paystack.co/transaction/verify/" . rawurlencode($ref),
```

Fig 4.34 Code snippet showing 'payWithPaystack' function processing user data

The reference is then passed to the paystack API using the rawurlencode() function to prevent any additional error information to the paystack API. The secret key is also passed to the API to send an encrypted confirmation of the user. To access response results of the verify_payment process, jsondecode() is used to pass the response variable and stored in another variable. This converts the JSON encoded 'response' variable to a PHP variable.

```php
    "Authorization: Bearer sk_test_88f636aa442339079681d799a017eb56b92648db",
    "Cache-Control: no-cache",
  ),
));

$response = curl_exec($curl);
$err = curl_error($curl);
curl_close($curl);

if ($err) {
  echo "cURL Error #:" . $err;
} else {
  //echo $response;
  $trans = json_decode($response);
}

if($trans->data->status == 'success'){
  // there was an error from the API
  header('Location: ../function/process_payment.php?status=completed');
}else{
  die('API returned error: ' . $trans->message);
}
```

Fig 4.35 Code snippet of API success response processing

Later an if statement is used to assess the success of the transaction and redirects to the process payment page. In the process payment page, the order is placed given the details from the cart then order detail are created and appended to the order_details table finally adding payment to the database with a successful status. Once all this is accomplished the user cart is emptied.

4.3.8 Chat Bot

The chatbot feature is a basic externally built chatbot that offers basic assistance to system users. This feature was implemented on the front-end using hubspot's already built chatbot and required at the end of the body tag of pages requiring the bot, such as 'index.php' the landing page, and 'doc.php' the chat forum. This feature attains functionality FR05 (chatbot).

```
<!-- Start of HubSpot Embed Code -->
<script type="text/javascript" id="hs-script-loader" async defer src="//js.hs-scripts.com/9415149.js"></script>
<!-- End of HubSpot Embed Code -->
```

Fig 4.36 Code snippet to include external HubSpot bot

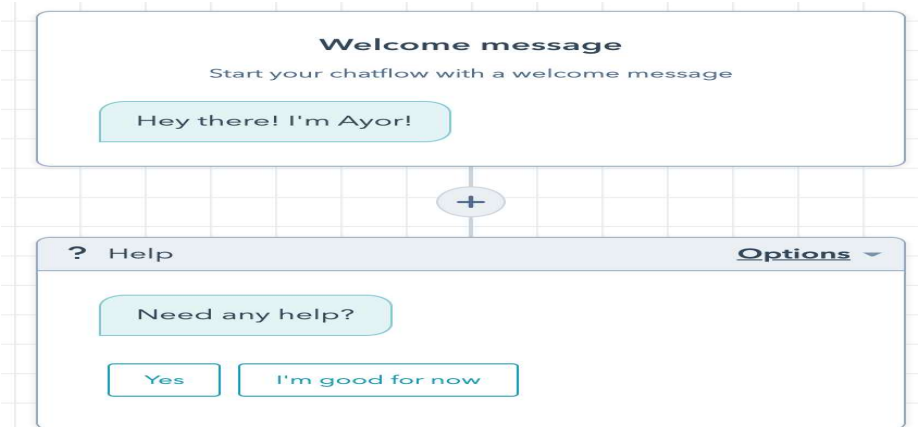


Fig 4.37 View of Bot Chat Flow.

4.3.9 Video Chat

The video chat feature utilizes WebRTC, an open-source project that provides real-time communication using application programming interfaces. In the implementation, the vidyio.io interface is used by including the external software development kit (SDK) script file. A guide to implementing this feature is retrieved from the WebRTC vidyio.io documentation kit available on the vidyio.io website.

```
<script src="https://static.vidyo.io/latest/javascript/VidyoClient/VidyoClient.js?onload=onVidyoClientLoaded"></script>
```

Fig 4.38 Vidyio.io SDK inclusion.

```
<script>
    var vidyoConnector;
        function onVidyoClientLoaded(status) {
            console.log("Client load state - " + status.state);
            if (status.state == "READY") {
                VC.CreateVidyoConnector({
                        viewId: "renderer", // Div ID where the composited video will be rendered, see VidyoConnector.html
                        viewStyle: "VIDYO_CONNECTORVIEWSTYLE_Default", // Visual style of the composited renderer
                        remoteParticipants: 3, // Maximum number of participants
                        logFileFilter: "error",
                        logFileName: "",
                        userData: ""
                    }).then(function(vc) {
                        console.log("Success");
                        vidyoConnector = vc;
                    }).catch(function(error) {
                        console.error("Fail " + error);
                    });
            }
```

Fig 4.39 Code Snippet of video connector object.

In setting out to implement this feature, first, check if the external script loaded as expected. A video connector object is then created that takes parameters such as 'viewId'; where the video should be displayed on the HTML page, the default style appearance of the video, the maximum number of remote participants per room, a log filter for errors. A major issue encountered in attempting to implement this feature included the user authentication and accessibility issue. Since users need to be registered and logged in before using the Medicare system to communicate using the video chat, a token needs to be generated and passed to the API. This process posed a challenge as the token generation algorithm was a new process to learn, and time constraints placed a strain on its implementation. In addition, the active response from the user's microphone and camera failed to connect after several successful tries.

Chapter 5: Testing

A system is tested to ensure it performs as expected. This chapter discusses the various testing techniques carried out on the Medicare system to assess its feasibility and desirability by system users. The main types of testing carried out in this phase are development tests, browser capability tests, and user acceptance tests.

5.1 Development Testing

Development test is carried out during the developmental process of a system to identify and correct bugs and other defects within the system. This system employed the unit and system test methods.

5.1.1 Unit Testing

In unit testing, the system's components are tested to ensure all unit's function as expected. In conducting unit testing, only the various classes used in implementing the operations of the system were tested, as well as the front-end execution of certain classes.

Table 1: Unit Test Results

CLASSES	FUNCTION	PASS/FAIL RESULTS
CART	Add_cart()	Pass
	Check_duplicate()	Pass
	Display_cart()	Pass
	Deleste_single_cart()	Pass
	Cart_value()	Pass
	View_all_medicaiton_in_cart()	Pass
	Add_payment()	Pass
	Add_order()	Pass

	Add_order_details()	Pass
	Recent_order()	Pass
	Get_order()	Pass
	Get_order_details()	Pass
CHAT	Get_user_last_activity()	Pass
	Get_user_chat_history()	Pass
	Get_ufname()	Pass
	View_all_users()	Pass
	Add_chat()	Pass
	Count_unseen_messages()	Pass
	Change_message()	Pass
	Istype()	Fail
	Istype_query()	Fail
	View_one_sys_user()	Pass
CONSULTATION	Add_ailments()	Pass
	Display_ail()	Pass
	Count_ail()	Pass
	Check_existing_ailments()	Fail
	Edit_one_ailment()	Pass
	View_one_possible_ailment()	Pass
	Add_med()	Pass
	Edit_one_med()	Pass
	Check_existing_med()	Fail
	Count_med()	Pass
	Display_med()	Pass

	One_med_info	Pass
	Search_med()	Pass
	Add_pres()	Pass
	Display_pres()	Pass
	View_one_possible_pres()	Pass
	Count_pres()	Pass
USER	Add_new_sys_user()	Pass
	View_all_sys_user()	Pass
	View_one_sys_user()	Pass
	Edit_one_sys_user()	Pass
	Verify_sys_user_login()	Pass
	Check_existing_sys_user()	Pass
	User_login_details()	Pass
	User_last_activity()	Pass
	View_all_patients()	Pass

5.1.2 System Testing

System testing is a level of software testing or mechanism that assesses the integrated and complete test functionality of a system. The purpose of the system test in this project is to each integrated component works as expected to ensure smooth interaction with users. In testing this application, a scenario is used to describe each component's interactive ability with users.

5.1.2.1 Scenario 1.0

A 70-year-old woman wakes up from bed one day with a general tummy ache. As a Rhema Rapha Medical center patient, she comes across a mail introducing their new telemedicine device called MediCare. She immediately decides to register into the system for the first time to contact her doctor about her runny tummy. The doctor receives all necessary details and recommends ORS for his patient. The pharmacist then approves the prescription and adds the medication to her cart. She then proceeds to make payment and awaits delivery at home. After successful payment and redirection to the receipt page, the cart is emptied, and the process is repeated for another user.

This scenario passed each test level upon execution on Windows Azure live server. This addressed nonfunctional requirement NFR04 (performance requirements).

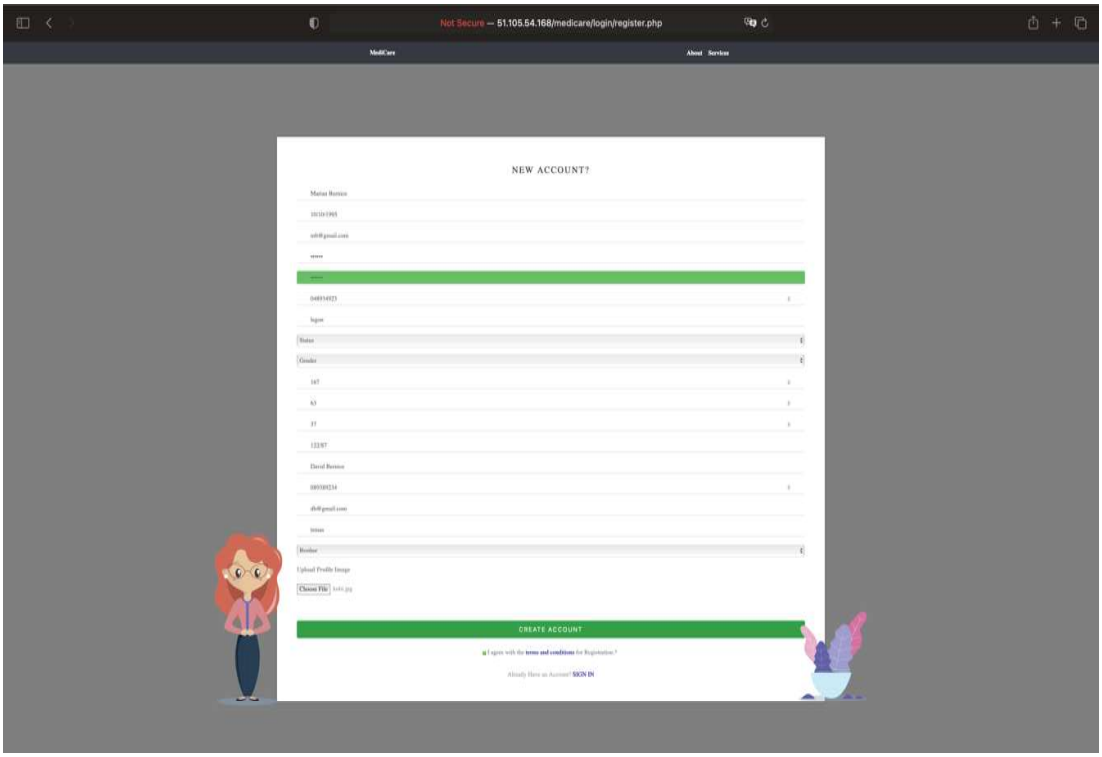


Fig 5.0 User Registering.

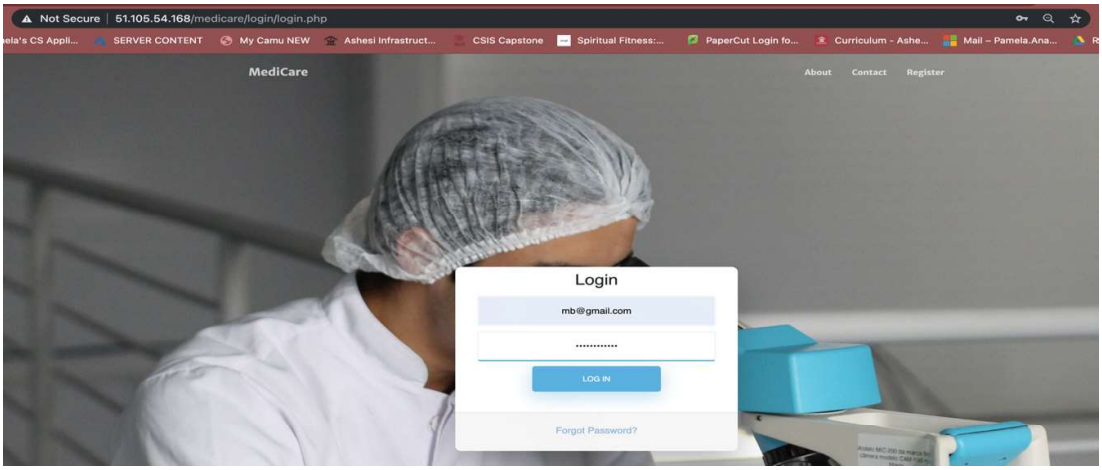


Fig 5.1 User Logging in.

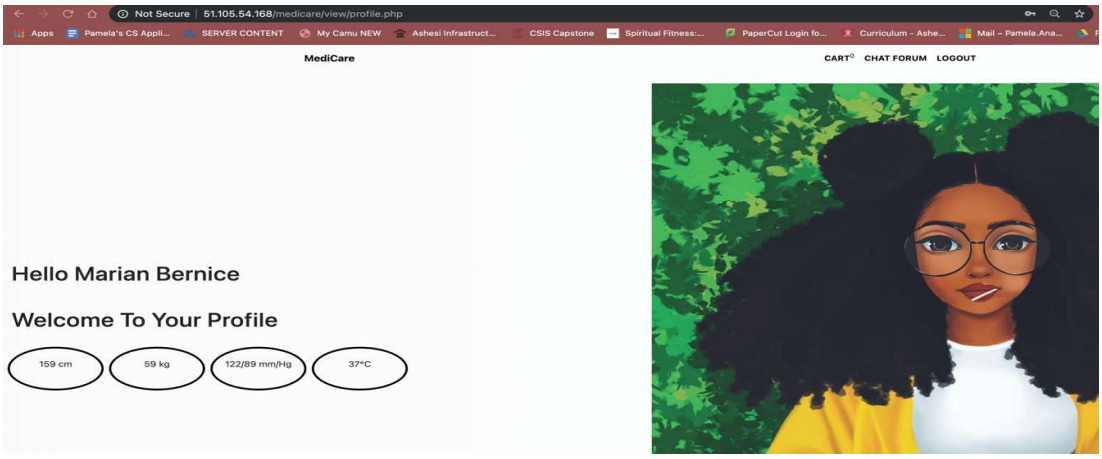


Fig 5.2 Logged in Sample User Profile (Patient Logged in).

Fig 5.3 Logged in Sample User Profile (Doctor Logged in).

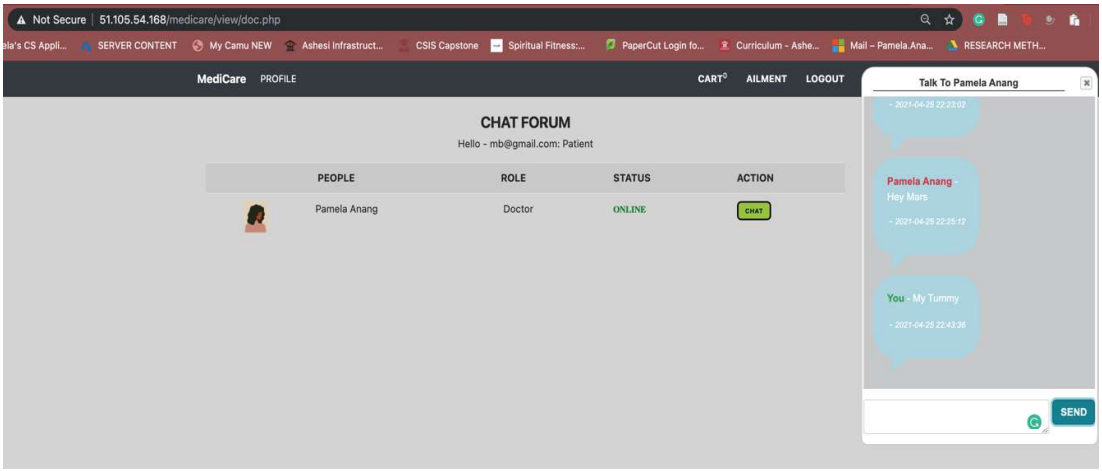


Fig 5.4 Logged in Patient Chat Forum of Similar User List to Pharmacist.

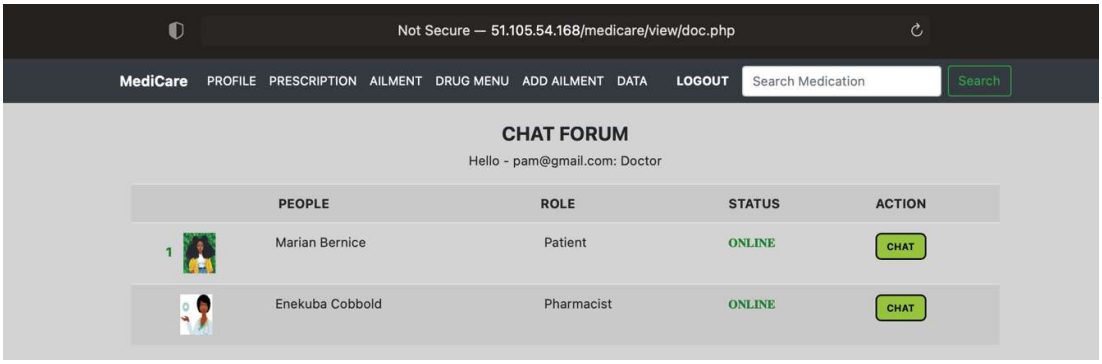


Fig 5.5 Logged in Doctor Chat Forum with message received from Patient.

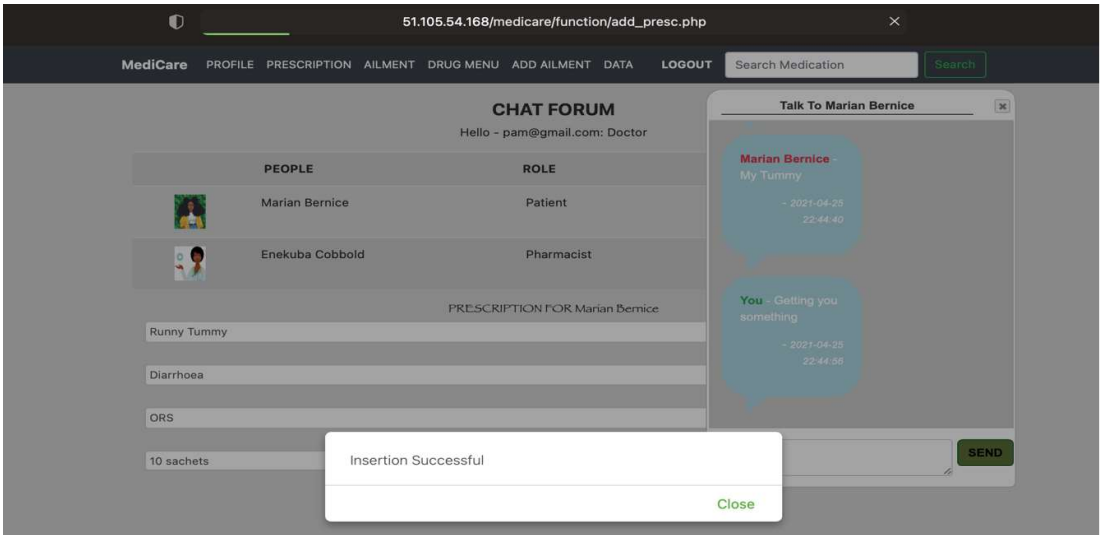


Fig 5.6 Doctor – patient interaction with prescription successfully written and stored.

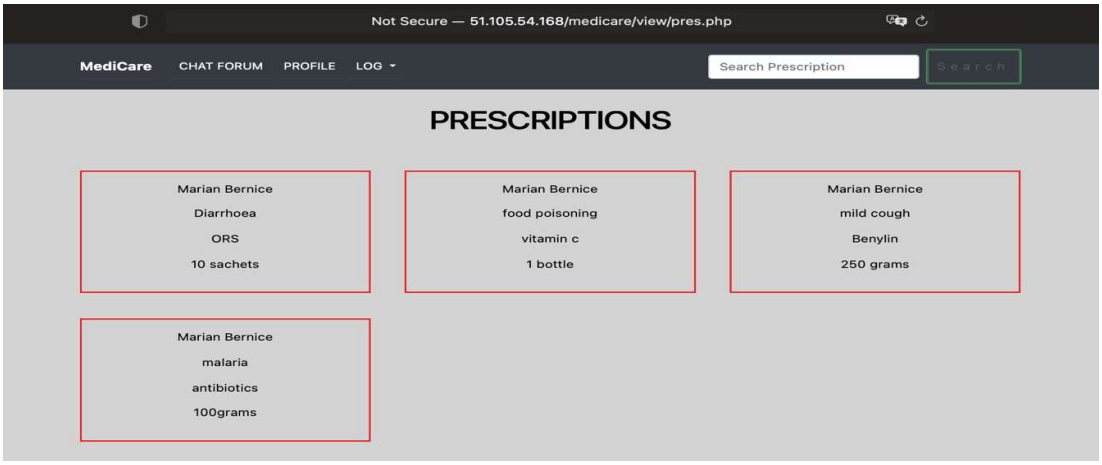


Fig 5.7 Pharmacist prescription log with newest prescription as first.

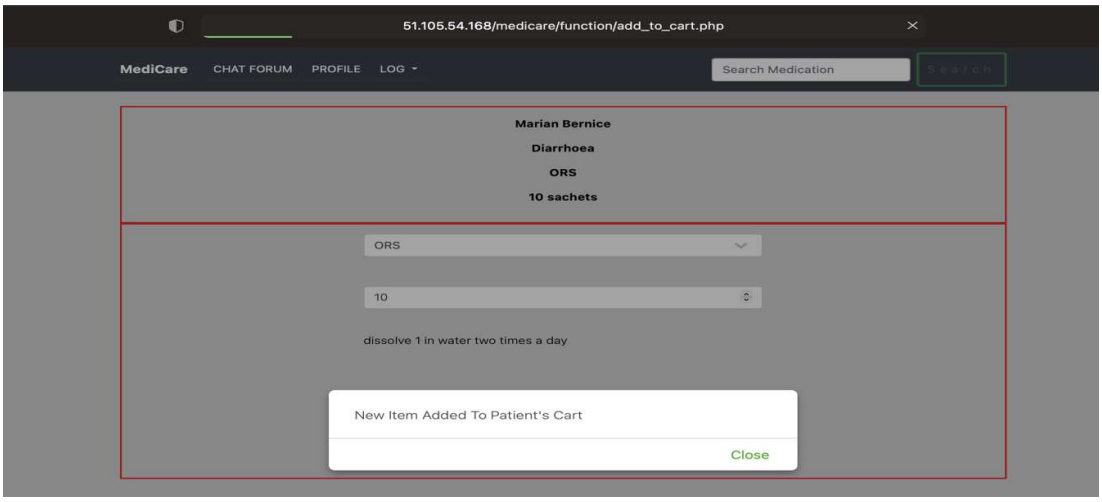


Fig 5.8 Prescription from Pharmacist Added to Patient's cart with Success Message

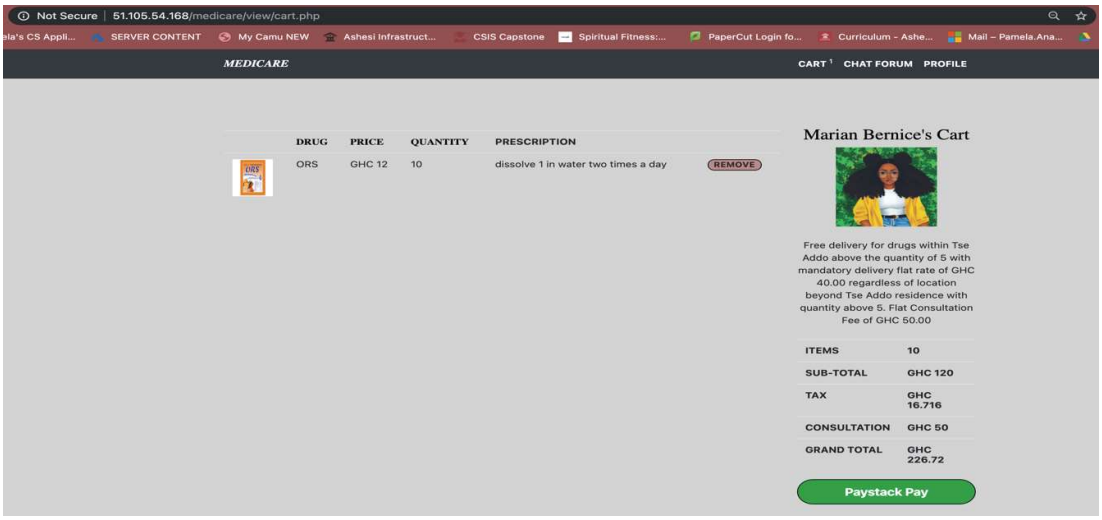


Fig 5.9 Patient Cart and Cart Summary with prescription provided.

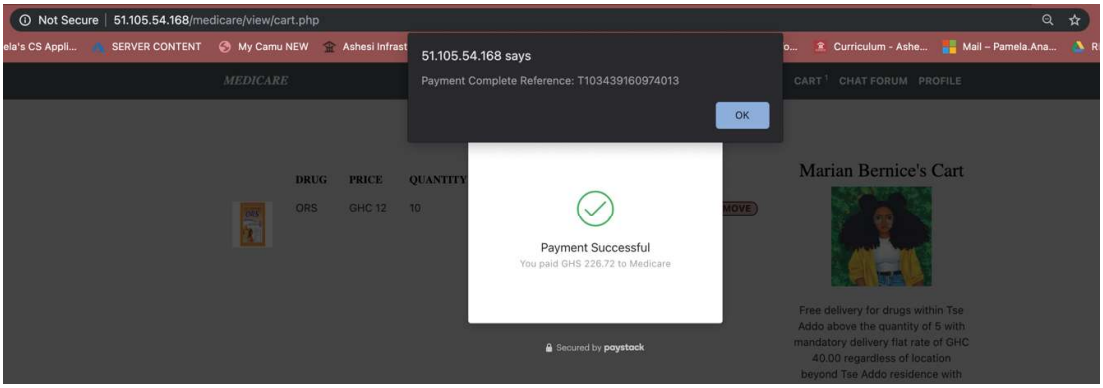


Fig 5.10 Successful Payment Completed by Patient.

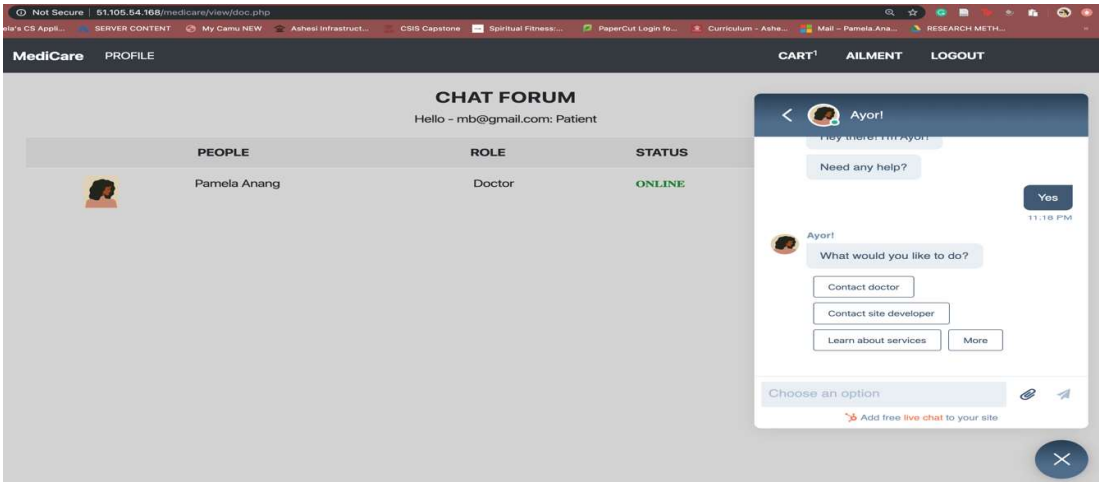


Fig 5.11 User Communicating with Chat bot

5.2 Browser Capability Testing

A capability test is a nonfunctional test that ensures a system can run on different browsers, different devices, different operating systems with different resolutions, and different networks when need. This application was used on Google Chrome, Safari, Mozilla Firefox to observe its operability. The system executed as expected; however, Safari and Mozilla Firefox failed to display the HTML date/time feature in the registration field.

Fig 5.12 Date of birth field using Safari or Mozilla.

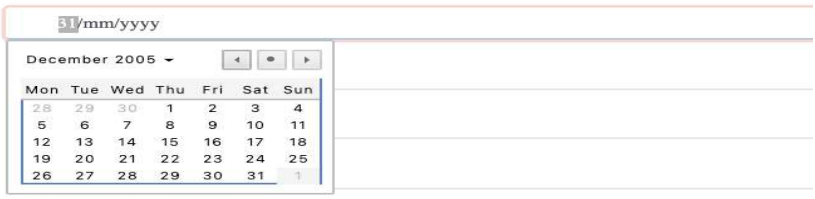


Fig 5.13 Date of Birth field using Google Chrome

5.3 User Acceptance Testing

User testing offers actual system users the platform to test the interface and functionalities of a system designed in a realistic setting.

In carrying out user acceptance testing, a survey was provided to Rhema Rapha patients and physicians through google form questionnaires. Users were given the URL link to the MediCare system hosted on the Microsoft Azure server. After being taken through the process of health consultation remotely. A few concerns were raised from the results of the survey conducted. A total of 20 users were contacted for the test. Three of these users were doctors, and two were pharmacists. A total of 17 responses were gathered at the end of the acceptability survey.

The features evaluated in the survey include

1) User interface acceptance

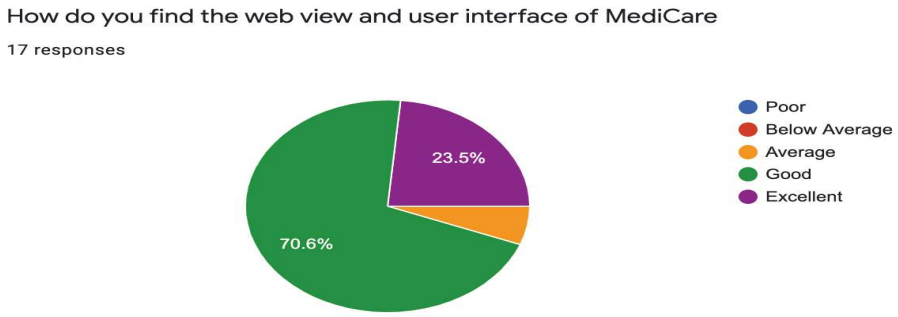


Fig 5.14 Results of interface acceptability question

2) System user convenience

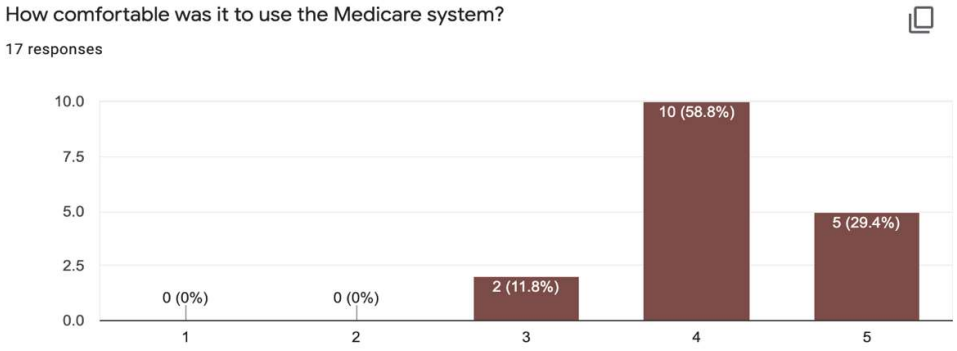


Fig 5.15 Results on convenience of system use

3) System functionality accomplishment

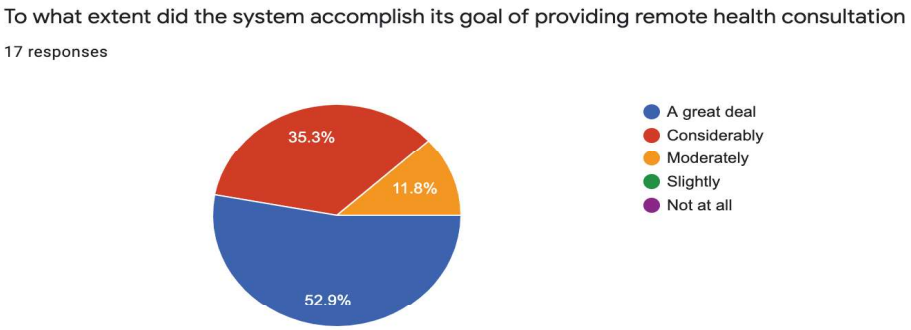


Fig 5.16 Results on system functionality accomplishment

4) System desirability and usefulness

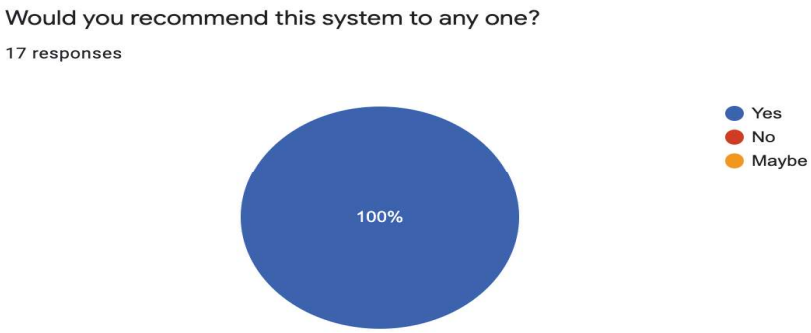


Fig 5.17 Results on system use and desirability

5.4 Analysis of Test Results

The test results above prove that the system successfully performs most functional and non-functional requirements as expected. The system, however, indicates an issue with the full function of the video conferencing feature. It activates the camera and microphones of both parties but does not process and display image (virtual) or audio communication. However, details from the user acceptability test carried out indicate a general acceptance of the system and desire to utilize the system. Furthermore, users suggested a few modifications to improve the system's appearance and inclusion of a mobile application version.

Chapter 6: Conclusion and Recommendations

This chapter concludes the processes involved in carrying out this capstone project. In addition, it discusses a summary of the project from the problem identification phase to final prototype testing. It also gives insight into the limitations or challenges faced during development and future works to be considered to improve the system's functionalities.

6.1 Project Summary

Medicare system is a web-based telemedicine system that seeks to connect patients of Rhema Rapha medical center to their physicians via chat room or video conferencing remotely. This is accomplished by creating an online platform called the 'chat forum' for all users upon registration and logging in. The chat forum allows patients and pharmacists to view the set number of doctors in Rhema Rapha registered on the system. In contrast, doctors may view all users, such as pharmacists, patients, and other doctors. With this telemedicine device, patients may receive a diagnosis from the comfort of their homes depending on the severity of their condition. They can receive medication at home as well using the payment and cart features. This implementation has aided in achieving the SDG 3 goal that improves health conditions by providing home health care during the COVID 19 pandemic.

6.2 Recommendations

Recommendations to this system and its implementation are based on the limitations and challenges faced during implementation and testing.

6.2.1 Limitations & Challenges

After conducting user acceptability testing, particular limitations to the project were identified. These limitations were also encountered as challenges during the developmental phase of the project and are as follows:

- The rotation between doctor to inform pharmacist via message on a new prescription added is a little tedious.
- The video chat does not display or allow communication between parties due to authentication and access issues.
- Patient vital data on a whole other page makes the transition between chat forum and the data inspection a lengthy process.
- The system only works on widened screens affecting its use on mobile phone web platforms.

A significant challenge faced in the development of the system was the successful implementation of the video conferencing feature using JavaScript's jQuery library. This feature utilized an API known as WebRTC through the vidyio.io interface that did not function as expected due to authentication and access and token generation issues.

6.2.2 Future Works

Future works of this project would focus on thoroughly addressing issues raised as limitations or challenges by both users and the researcher. Future works should also ideally include:

- A mobile application version of the system to improve remote use.
- Further improvements to the user interface, such as using a modal to display each user's vitals data.
- Other functionalities of the system could adopt the AJAX technique to enhance system responsivity through lightweight requests that do not affect page reloading.

- A functioning delivery feature that collects orders placed and creates a delivery unit to reduce physical interaction needed in facilitating drug delivery.
- Full implementation of a functioning video conferencing feature by addressing accessibility and authentication issues.

6.3 Conclusion

In conclusion, the MediCare system developed for Rhema Rapha Medical Center offers a substantial contribution to the implementation of telemedicine devices in Ghana that aid in remote medical assistance. Considering the current COVID 19 pandemic, this telemedicine device would help enforce social distancing protocols, thereby reducing active cases. This system also plays a role in achieving the SDG 3 target of universal health coverage by providing remote consultation during the ongoing COVID 19 pandemic.

References

[1] World Health Organization (Ed.). 2010. *Telemedicine: opportunities and developments in member states: report on the second Global survey on eHealth.* World Health Organization, Geneva, Switzerland.

[2] Pan American Health Organization. 2016.Fremawork for the Implementation of a Telemedicine Service. Washington DC.

[3] E. T. Tchao, Isaac Acquah, S. D., C. S., and J. J. 2019. On Telemedicine Implementations in Ghana. *ijacsa* 10, 3 (2019). DOI:https://doi.org/10.14569/IJACSA.2019.0100325

[4] Goal 3. Department of Economic and Social Affairs. Retrieved October 13, 2020 from https://sdgs.un.org/goals/goal3

[5] Ghana telemedicine. Novartis Foundation. Retrieved October 12, 2020 from https://www.novartisfoundation.org/our-work/reimagining-healthcare-through-digital-technology/ghana-telemedicine

[6] Jackie Sturt, Caroline Huxley, Btihaj Ajana, Caitjan Gainty, Chris Gibbons, Tanya Graham, Zarnie Khadjesari, Federica Lucivero, Rebecca Rogers, Annie Smol, JocelynA Watkins, and Frances Griffiths. 2020. How does the use of digital consulting change the meaning of being a patient and/or a health professional? Lessons from the Long-term Conditions Young People Networked Communication study. DIGITAL HEALTH 6, (January 2020), 2055207620942359. DOI:https://doi.org/10.1177/2055207620942359

[7] Miha Cimperman, Maja Makovec Brenčič, Peter Trkman, and Mateja de Leonni Stanonik. 2013. Older Adults' Perceptions of Home Telehealth Services. *Telemed J E Health* 19, 10 (October 2013), 786–790. DOI:https://doi.org/10.1089/tmj.2012.0272

[8] Roshini Peiris-John, Lovely Dizon, Kylie Sutcliffe, Kristy Kang, and Theresa Fleming. Co-creating a large-scale adolescent health survey integrated with access to digital health interventions. DIGITAL HEALTH, 13.

[9] Telemedicine Software Platform for Health Systems & Hospitals. *MDLIVE*. Retrieved October 13, 2020 from https://www.mdlive.com/health-systems/health-systems-contact/

[10] Nyaho Medical Centre. 2020. Nyaho Medical Centre and Africa eHealth. Nyaho Medical Centre. Retrieved October 14, 2020 from https://www.nyahomedical.com/about/updates/nyaho-medical-centre-and-africa-e-health-solutions-partner-for-innovative-healthcare-in-ghana

[11] Daniel C. Baumgart. 2020. Digital advantage in the COVID-19 response: perspective from Canada's largest integrated digitalized healthcare system. npj Digit. Med. 3, 1 (December 2020), 114. DOI:https://doi.org/10.1038/s41746-020-00326-y

[12] Judith Byaruhanga, Prince Atorkey, Matthew McLaughlin, Alison Brown, Emma Byrnes, Christine Paul, John Wiggers, and Flora Tzelepis. 2020. Effectiveness of Individual Real-Time Video Counseling on Smoking, Nutrition, Alcohol, Physical Activity, and Obesity Health Risks: Systematic Review. J Med Internet Res 22, 9 (September 2020), e18621. DOI:https://doi.org/10.2196/18621

[13] BIMA Doctor receives first ever telemedicine license in Ghana. Bima Mobile. Retrieved October 13, 2020 from https://bimamobile.com/archives/news/bima-doctor-receives-first-ever-telemedicine-licence-in-ghana

[14] Comarch - Global IT Business Products Provider. Retrieved September 14, 2020 from https://www.comarch.com/healthcare/products/remote-medical-care/remote-medical-center/

[15] Tan, S. S.-L., & Goonawardene, N. (2017). Internet health information seeking and the patient-physician relationship: a systematic review. Journal of Medical Internet Research, 19(1), e9. http://doi.org/10.2196/jmir.5729

[16] Ryen W. White and Eric Horvitz. 2009. Experiences with Web Search on Medical Concerns and Self Diagnosis. *AMIA Annu Symp Proc* 2009, (2009), 696–700.

[17] Ailsa Colquhoun. 2014. Think outside the box so you can drive your pharmacy business forward. The Pharmaceutical Journal. (February 2020). DOI: https://doi.org/10.1211/PJ.2014.11134176

[18] What is Agile Software Development? Agile Alliance. 2015. Retrieved November 10, 2020 from https://www.agilealliance.org/agile101/

[19] Agile project management with Scrum. Retrieved October 14, 2020 from https://www.pmi.org/learning/library/agile-project-management-scrum-6269

[20] Ilker Etikan. 2016. Comparison of Convenience Sampling and Purposive Sampling. AJTAS 5, 1 (2016), 1. DOI:https://doi.org/10.11648/j.ajtas.20160501.11

[21] Ulf Eriksson. 2012. Functional vs Non-Functional Requirements - Understand the Difference. ReQtest. Retrieved November 12, 2020 from https://reqtest.com/requirements-blog/functional-vs-non-functional-requirements/

[22] What is Use Case Diagram? Retrieved November 12, 2020 from https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/

[23] MVC Pattern. The Model-View-Controller (MVC) is an… | by Anshul Vyas | Medium. Retrieved April 26, 2021 from https://medium.com/@anshul.vyas380/mvc-pattern-3b5366e60ce4

[24] What Is a Front-End Developer? Retrieved April 26, 2021 from https://frontendmasters.com/books/front-end-handbook/2018/what-is-a-FD.html

[25] Bootstrap 4 Tutorial - An Ultimate Guide for Beginners. Retrieved April 26, 2021 from https://www.tutorialrepublic.com/twitter-bootstrap-tutorial/

[26] Introduction to HTML. Retrieved March 4, 2020, from https://www.w3schools.com/ html/html_intro.asp

[27] CSS Tutorial. Retrieved March 4, 2020, from https://www.w3schools.com/css/

[28] JavaScript basics. MDN Web Docs. Retrieved March 4, 2020, from https:// developer.mozilla.org/en-US/docs/Learn/ Getting_started_with_the_web /JavaScript_basics

[29] PHP | cURL. GeeksforGeeks. (2018). Retrieved April 26, 2021 from https://www.geeksforgeeks.org/php-curl/

[30] jQuery Introduction. Retrieved April 8, 2020 from https://www.w3schools.com/jquery/jquery_intro.asp

[31] Ajax - Developer guides | MDN. Retrieved March 30, 2021 from https://developer.mozilla.org/en-US/docs/Web/Guide/AJAX

[32] Working with JSON - Learn web development | MDN. Retrieved March 30, 2021 from https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Objects/JSON

[33] Regular expressions - JavaScript | MDN. Retrieved April 26, 2021 from https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular_Expressions

[34] What is an API? (Application Programming Interface). MuleSoft. Retrieved April 26, 2021 from https://www.mulesoft.com/resources/api/what-is-an-api

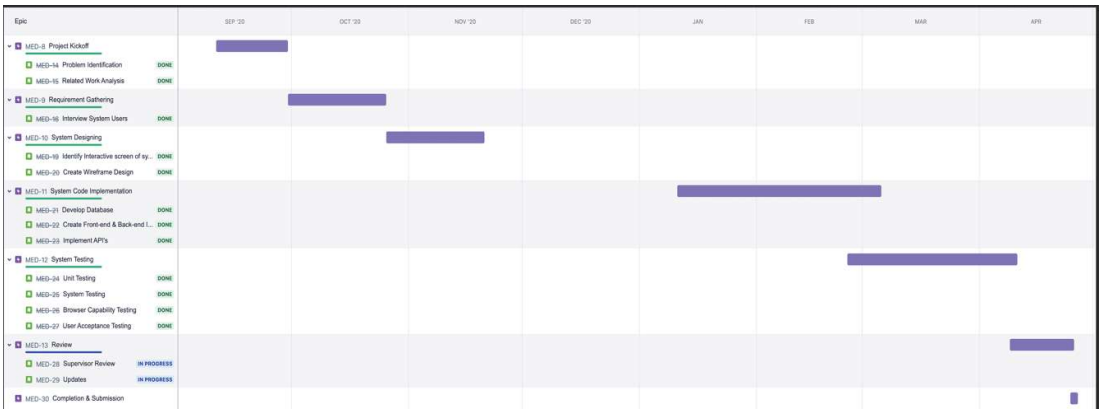
[35] HubSpot. Why Go HubSpot? Retrieved April 26, 2021 from https://www.hubspot.com/why-go-hubspot

[36] MySQL Tutorial - Tutorialspoint. Retrieved March 4, 2020, from https://www.tutorialspoint.com/mysql/index.htm

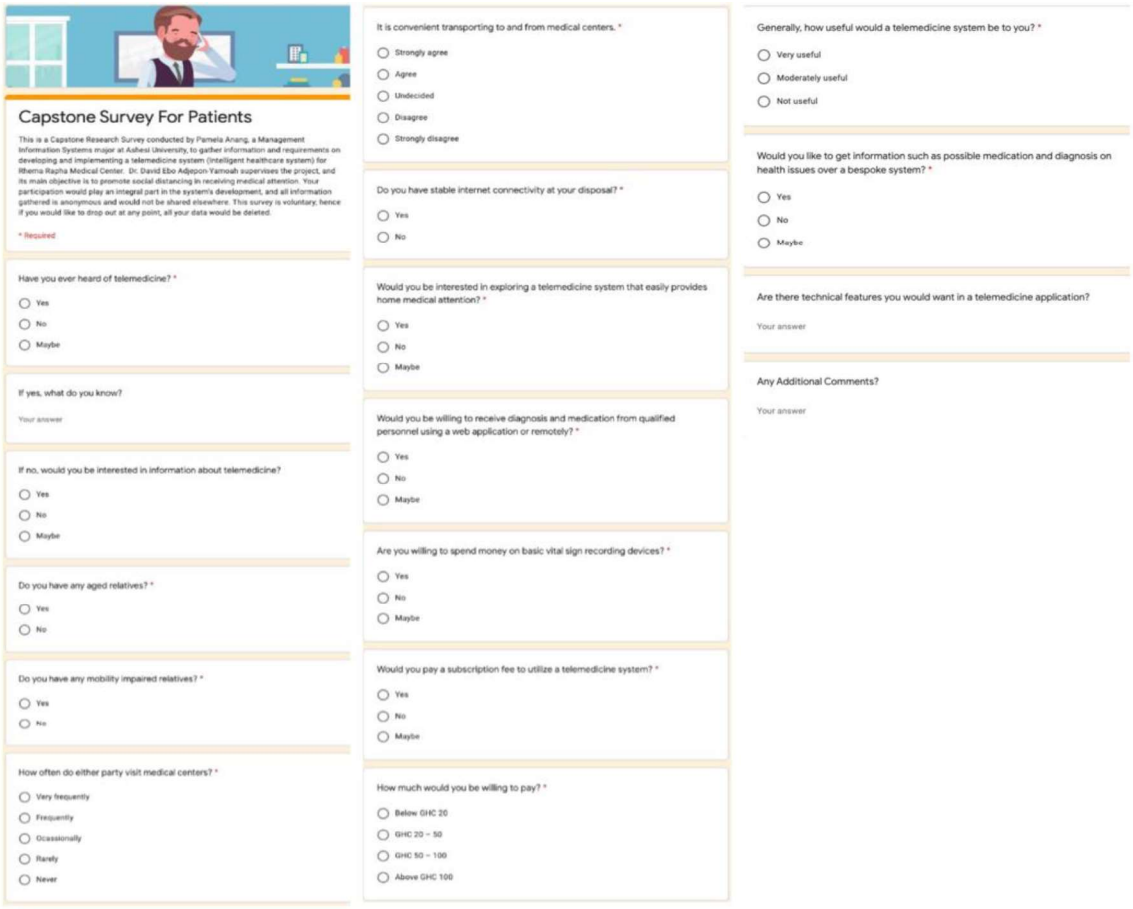
[37] Rowena Luk, Melissa Ho, and Paul M Aoki. Asynchronous Remote Medical Consultation for Ghana. 10.

Appendices

Appendix A: Project Gantt Chart



Appendix B: Interview Questionnaires



Capstone Survey For Patients

This is a Capstone Research Survey conducted by Pamela Anang, a Management Information Systems major at Ashesi University, to gather information and requirements on developing and implementing a telemedicine system (intelligent healthcare system) for Rhema Rapha Medical Center. Dr. David Ebo Adjepon-Yamoah supervises the project, and its main objective is to promote social distancing in receiving medical attention. Your participation would play an integral part in the system's development, and all information gathered is anonymous and would not be shared elsewhere. This survey is voluntary, hence if you would like to drop out at any point, all your data would be deleted.

* Required

Have you ever heard of telemedicine? *
Yes
No
Maybe

If yes, what do you know?
Your answer

If no, would you be interested in information about telemedicine?
Yes
No
Maybe

Do you have any aged relatives? *
Yes
No

Do you have any mobility impaired relatives? *
Yes
No

How often do either party visit medical centers? *
Very frequently
Frequently
Occasionally
Rarely
Never

It is convenient transporting to and from medical centers. *
Strongly agree
Agree
Undecided
Disagree
Strongly disagree

Do you have stable internet connectivity at your disposal? *
Yes
No

Would you be interested in exploring a telemedicine system that easily provides home medical attention? *
Yes
No
Maybe

Would you be willing to receive diagnosis and medication from qualified personnel using a web application or remotely? *
Yes
No
Maybe

Are you willing to spend money on basic vital sign recording devices? *
Yes
No
Maybe

Would you pay a subscription fee to utilize a telemedicine system? *
Yes
No
Maybe

How much would you be willing to pay? *
Below GHC 20
GHC 20 – 50
GHC 50 – 100
Above GHC 100

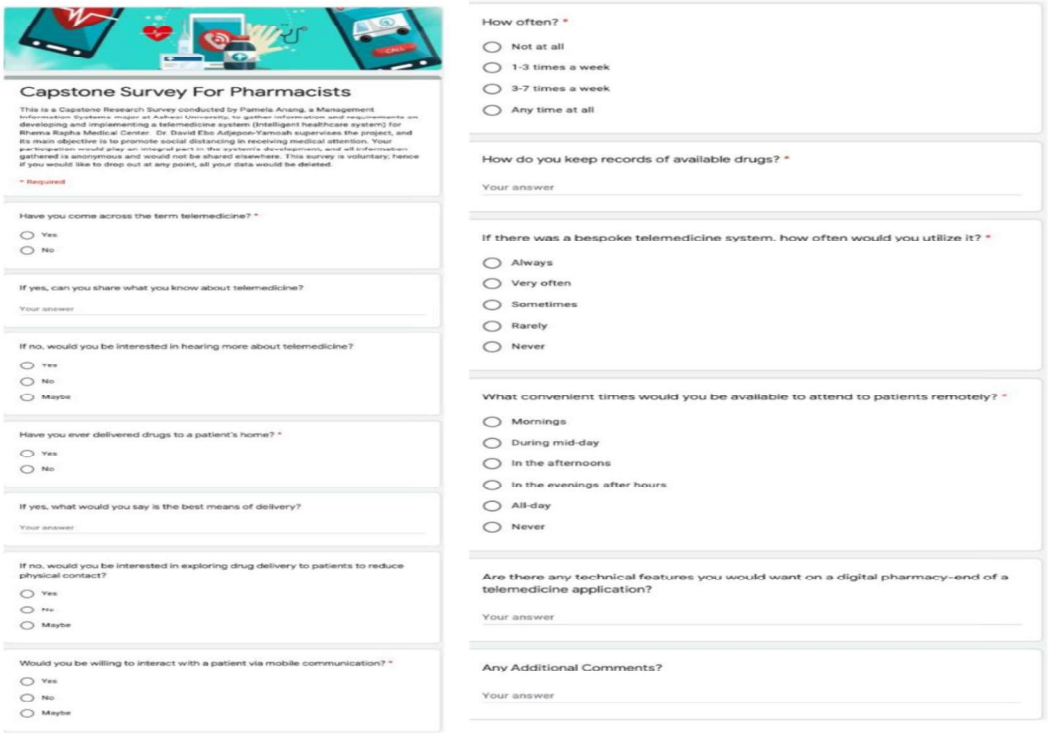
Generally, how useful would a telemedicine system be to you? *
Very useful
Moderately useful
Not useful

Would you like to get information such as possible medication and diagnosis on health issues over a bespoke system? *
Yes
No
Maybe

Are there technical features you would want in a telemedicine application?
Your answer

Any Additional Comments?
Your answer

Interview Questionnaire for Patients



Capstone Survey For Pharmacists

This is a Capstone Research Survey conducted by Pamela Anang, a Management Information Systems major at Ashesi University, to gather information and requirements on developing and implementing a telemedicine system (Intelligent healthcare system) for Rhema Rapha Medical Center. Dr. David Ebo Adjepon-Yamoah supervises the project, and its main objective is to promote social distancing in receiving medical attention. Your participation would play an integral part in the system's development, and all information gathered is anonymous and would not be shared elsewhere. This survey is voluntary; hence if you would like to drop out at any point, all your data would be deleted.

* Required

Have you come across the term telemedicine? *
- Yes
- No

If yes, can you share what you know about telemedicine?
Your answer

If no, would you be interested in hearing more about telemedicine?
- Yes
- No
- Maybe

Have you ever delivered drugs to a patient's home? *
- Yes
- No

If yes, what would you say is the best means of delivery?
Your answer

If no, would you be interested in exploring drug delivery to patients to reduce physical contact?
- Yes
- No
- Maybe

Would you be willing to interact with a patient via mobile communication? *
- Yes
- No
- Maybe

How often? *
- Not at all
- 1-3 times a week
- 3-7 times a week
- Any time at all

How do you keep records of available drugs? *
Your answer

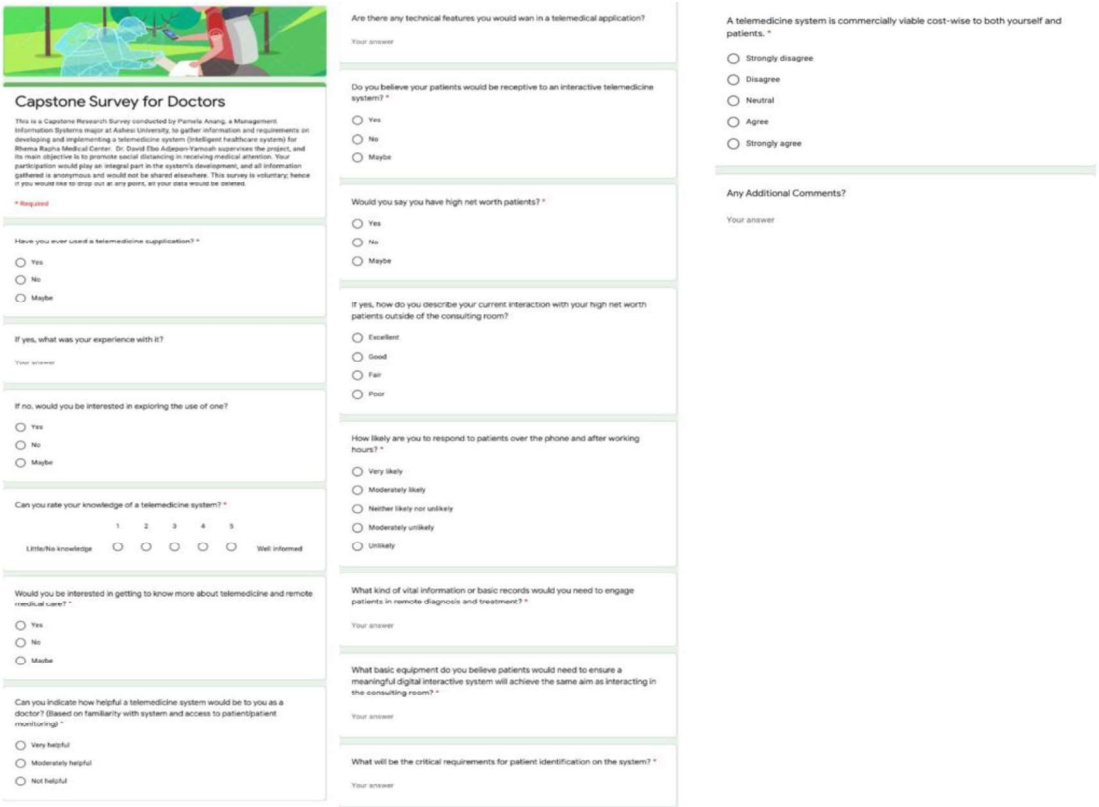
If there was a bespoke telemedicine system, how often would you utilize it? *
- Always
- Very often
- Sometimes
- Rarely
- Never

What convenient times would you be available to attend to patients remotely? *
- Mornings
- During mid-day
- In the afternoons
- In the evenings after hours
- All-day
- Never

Are there any technical features you would want on a digital pharmacy-end of a telemedicine application?
Your answer

Any Additional Comments?
Your answer

Interview Questionnaire for Pharmacists



Capstone Survey for Doctors

This is a Capstone Research Survey conducted by Pamela Anang, a Management Information Systems major at Ashesi University, to gather information and requirements on developing and implementing a telemedicine system (Intelligent healthcare system) for Rhema Rapha Medical Center. Dr. David Ebo Adjepon-Yamoah supervises the project, and its main objective is to promote social distancing in receiving medical attention. Your participation would play an integral part in the system's development, and all information gathered is anonymous and would not be shared elsewhere. This survey is voluntary; hence if you would like to drop out at any point, all your data would be deleted.

* Required

Have you ever used a telemedicine supplication? *
- Yes
- No
- Maybe

If yes, what was your experience with it?
Your answer

If no, would you be interested in exploring the use of one?
- Yes
- No
- Maybe

Can you rate your knowledge of a telemedicine system? *

	1	2	3	4	5	
Little/No knowledge	○	○	○	○	○	Well informed

Would you be interested in getting to know more about telemedicine and remote medical care? *
- Yes
- No
- Maybe

Can you indicate how helpful a telemedicine system would be to you as a doctor? (Based on familiarity with system and access to patient/patient monitoring) *
- Very helpful
- Moderately helpful
- Not helpful

Are there any technical features you would wan in a telemedical application?
Your answer

Do you believe your patients would be receptive to an interactive telemedicine system? *
- Yes
- No
- Maybe

Would you say you have high net worth patients? *
- Yes
- No
- Maybe

If yes, how do you describe your current interaction with your high net worth patients outside of the consulting room?
- Excellent
- Good
- Fair
- Poor

How likely are you to respond to patients over the phone and after working hours? *
- Very likely
- Moderately likely
- Neither likely nor unlikely
- Moderately unlikely
- Unlikely

What kind of vital information or basic records would you need to engage patients in remote diagnosis and treatment? *
Your answer

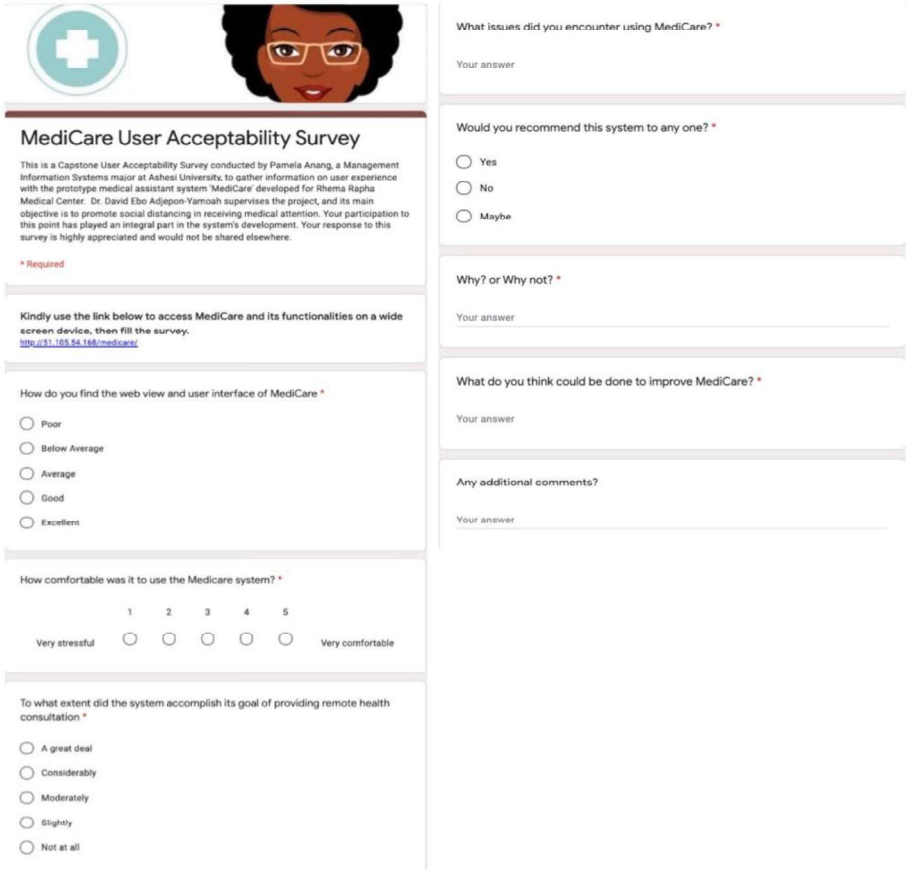
What basic equipment do you believe patients would need to ensure a meaningful digital interactive system will achieve the same aim as interacting in the consulting room? *
Your answer

What will be the critical requirements for patient identification on the system? *
Your answer

A telemedicine system is commercially viable cost-wise to both yourself and patients. *
- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree

Any Additional Comments?
Your answer

Interview Questionnaire for Doctors



MediCare User Acceptability Survey

This is a Capstone User Acceptability Survey conducted by Pamela Anang, a Management Information Systems major at Ashesi University, to gather information on user experience with the prototype medical assistant system 'MediCare' developed for Rhema Rapha Medical Center. Dr. David Ebo Adjepon-Yamoah supervises the project, and its main objective is to promote social distancing in receiving medical attention. Your participation to this point has played an integral part in the system's development. Your response to this survey is highly appreciated and would not be shared elsewhere.

* Required

Kindly use the link below to access MediCare and its functionalities on a wide screen device, then fill the survey.
http://51.105.54.168/medicare/

How do you find the web view and user interface of MediCare *

- Poor
- Below Average
- Average
- Good
- Excellent

How comfortable was it to use the Medicare system? *

	1	2	3	4	5	
Very stressful	○	○	○	○	○	Very comfortable

To what extent did the system accomplish its goal of providing remote health consultation *

- A great deal
- Considerably
- Moderately
- Slightly
- Not at all

What issues did you encounter using MediCare? *

Your answer

Would you recommend this system to any one? *

- Yes
- No
- Maybe

Why? or Why not? *

Your answer

What do you think could be done to improve MediCare? *

Your answer

Any additional comments?

Your answer

User Acceptability Questionnaire