UNIVERSITI POLY-TECH MALAYSIA

SMARTHEALTH: PATIENT MONITORING AND ADVISING HEALTHCARE SYSTEM

MOHAMAD IRFAN BIN MOHAMED AZWAN

BACHELOR OF INFORMATION TECHNOLOGY (HONS) IN COMPUTER APPLICATION DEVELOPMENT

UNIVERSITI POLY-TECH MALAYSIA
Faculty of Computing & Multimedia

SMARTHEALTH : PATIENT MONITORING AND ADVISING HEALTHCARE SYSTEM

MOHAMAD IRFAN BIN MOHAMED AZWAN
AM2311015251

FYP4025

NOVEMBER 2025

Declaration of Originality

This project is all my own work and has not been copied in part or in whole from any other source except where duly acknowledged. As such, all use of previously published work (from books, journals, magazines, internet, etc.) has been acknowledged within the main report to an item in the References or Bibliography lists.

I also agree that an electronic copy of this project may be stored and used for the purposes of plagiarism prevention and detection.

Copyright Acknowledgement

Signed: Date: 6 June 2025

Office Stamp

Abstract

SmartHealth: Patient Monitoring and Advising Healthcare System is an online-system based on the web designed to foster better execution of the healthcare services in Hospital Taiping through the ability to monitor the health condition of patients in real-time and remotely. Some of the problems both doctors and patients deal with regularly and that will be be addressed with the help of the system include time losses when health data is updated, manual records-keeping, and the absence of individualized medical advice. The ability to upload their daily statistics on health, including blood pressure, glucose and symptoms, will provide the medical workers with the capability of recording trends and noting outliers and providing valuable feedback to patients without a direct visit during non-emergency situations.

Agile methodology was adopted in the development of the system to facilitate adaptability within the environment, constant feedback, and progressive improvement. Visual Studio Code, Laravel, HTML, CSS, JavaScript, and other tools were found to design the interface that is convenient to use by both the doctors and the patients. Some of the important highlights are user registration, submission of health data, tracking patient history, health tips, and physician dashboard in terms of feedback, and exporting the data. The collection of data by use of questionnaires and interviewing of medical professionals allowed establishing that the system features met the real life healthcare needs. In general SmartHealth is supposed to enhance patient participation, diminish redundant and more frequently, preventable admissions to the hospital and facilitate proactive medical choices in a safe and scalable manner.

Table of Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank Allah S.W.T who gave me the strength, patience, and perseverance to accomplish this project. This would have otherwise not been possible without His blessings and direction.
I would also want to take this opportunity to say heartfelt gratitude to my adorable parents who kept on praying, encouraging and supporting me all through my academic process. I would like to express my sincere gratitude to Majlis Amanah Rakyat (MARA) to avail an opportunity and resources that has helped in my education largely.

My supervisor Sir Mohd Noor Afiq Bin Ramlee is especially thanked and acknowledged due to his useful advice, critique and support during the design of the SmartHealth: Patient Monitoring and Advising Healthcare System. His working experience and positive advice played an essential role in the success of this project.

I would also wish to note my sincere gratitude to project coordinator, Madam Noraliza Binti Azizan, who has always given me the necessary support and helped in monitoring the success of this project diligently. I would like to express my gratitude to my client on the sourcing side who is Dr Nurulraihan Binti Mohamed zawawi working at Hospital Taiping who assisted and helped me with the research development process by providing her skills and expertise throughout the research development process.

Finally, I would like to thank all my friends and fellow classmates that gave me moral support, lent knowledge and ideas and helped me to be motivated enough to finish this project.

1 INTRODUCTION

1.1 Introduction

This chapter gives a detailed description of the project, its background, the issues that it will be solving, the goals to be attained, and the general scope of the system. In the modern digitalized world, most healthcare systems across the globe are contemplating on how they can use technology as an opportunity to deliver better services, mitigate barriers associated with patient-provider interactions, and guarantee the presence of effective and real-time health data. Nevertheless, in most health facilities, the conventional practices have remained to be manual recordkeeping and late reporting that impede effective healthcare provision.

The SmartHealth: Patient Monitoring and Advising Healthcare System project described in this report will help alleviate the mentioned shortcomings by creating a web-based system that will help in updating current approaches to patient health monitoring and ensuring that they receive timely and personalized healthcare assistance. The system will have advantages to the patients and the doctors, especially at the level of Hospital Taiping, where health tracking will become more streamlined since many of its processes are manual; patient health data will become available in real-time; and health recommendations will be provided by Artificial Intelligence (AI).

The chapter starts with an analysis of the project background and the reason why this project has to be developed and then goings in detail on what the problem picture is nowadays regarding the healthcare monitoring process. The following part of the chapter provides the primary goals of the project, creates scope of the project and that of the product and determines the key users who will access the advantage of the proposed solution. Lastly, there is a concise layout of how the report will be organized to give guidance to the reader on what lies ahead in the report.

1.2 Project Background

SmartHealth: Patient Monitoring and Advising healthcare system is a web based system that improves on remotely took care of patients through increasing the amount of right, time and quality of care. SmartHealth: Patient Monitoring and Advising Healthcare System is created to address the existing

issues that doctors and patients at Hospital Taiping have to deal with to ensure continuous health monitoring and communication. With the current workflow, the health data of patients which is blood pressure, glucose level, symptoms being recorded on papers manually, or transmitted verbally when the patient visits the healthcare facility physically. This is not only a source of delays in data entry and sharing but also provides an opportunity to encounter difficulties like human errors, loss of data, and ineffectiveness of follow-ups, especially in cases with patients with chronic conditions whose monitoring should be both regular and effective so that a patient can be involved or offered medical help in time.

Doctors and nurse at Hospital Taiping have also expressed challenges in getting updated patient information within a short time, especially during off-site hours or when patients are not favored with appointments. Simultaneously, patients can hardly afford such frequent monitoring due to the lack of appropriate tools and inclination, which creates gaps in care and blocks efficient medical response. The absence of a central system of real-time monitoring and individualized health recommendations denies the hospital the possibility of providing proactive care and handling patient outcomes in a more operative manner.

The SmartHealth system proposed in this project provides a solution to such problems because it will provide a centralized, web-based system in which those with the system can manually submit their daily health data, but doctors and nurse can access that data remotely to check the trends, find irregularities, and give patient-specific feedback. The system is designed to enhance communication, provide early warning in condition health risks, and help patients become more engaged in the process, minimize visits to the hospital, and allow doctors and nurse to make better decisions. This system is especially useful to patients with chronic diseases and healthcare workers with high cases to deal with in the hospitals.

This project aims to create web-based system for hospital Taiping to give benefits to doctors and patients by improved instant and real-time access to patient health information, provide personalized health recommendations based on individual patient data, and also provide digital platform for patients to input their health data.

1.3 Problem Statement

In the current health sector, hospital Taiping are not able to manage and track patient health information effectively. The lack of an individual approach to care and delays in having access to health records and manual processes decrease the efficiency and quality of work with patients. The problems tend to cause inaccurate records, unnoticed emergency indicators and little participation of the patients

in their health care management. The following problems outline the primary issues that SmartHealth system is intended to solve.

1.3.1 Delayed Data Availability Compromises Emergency Response

The delay in data access in emergency care facilities is another critical factor that negatively influences timely medical decision-making and may have detrimental outcomes on patients. Tong et al. (2025) analyses the impact of the use of mobile applications in emergency medical services, particularly to improve the survival of patients with out-of-hospital cardiac arrest, which has been found to have remarkably attained outstanding outcomes. Systematic reviewing and analyzing by them showed that these types of applications improved the efficiency of the emergency response and provided a higher survival rate owing to the real-time information and communication that it enables. The research emphasizes the need of having real-time access to gathered data during an emergency scenario, where means of foregoing data can degrade the success rate of interventions and result in loss of life measurably .

1.3.2 Lack of Personalized Health Recommendations Limits Patient Engagement

The absence of tailored health tips often reduces how much patients participate in managing their chronic conditions. General advice that does not take into account a person's health, likes or daily life can make them feel separated from their treatment and less willing to follow it. According to researchers Jesuthasan et al. (2022), giving personalized feedback in digital health matters increases user involvement and follows doctors advice better. It highlights that designing and personalizing interventions can encourage accountable behavior and increase the amount of social support which can make users more involved in digital health programs.

1.3.3 Manual Health Monitoring is Slow and Inefficient

Manual health monitoring remains slow and inefficient due to its reliance on manual data entry such as paper manual entries, which introduces delays and increases the likelihood of errors in recording vital health parameters such as blood pressure, heart rate, and glucose levels or any symptoms. Because of this approach, time gets wasted and patient records are often incomplete or out-of-date which affects the quality of healthcare. Many healthcare clinicians still write down information on paper during patient consultations and later transfer it into electronic health records (EHRs) which is not just inefficient but can also lead to errors. A report by Salahuddin et al. (2020) goes on to say the meaning of workaround is that healthcare practitioners frequently work around a convenient point of access by writing their notes on paper and then typing the information in the hospital

information system (HIS) later. This leads to situations of late data entry or even making mistakes since the practitioners forget to enter patient notes or enter the wrong data to the detriment of the patient and the quality of care .

1.4 Project Objectives

1.4.1 To allow instant and real-time access to health information

The goal of this objective is to minimize delays experienced when addressing a given condition using a conventional healthcare system where most incidences are usually documented manually and can only be reviewed when one carries out regular check-ups. With the introduction of an online system where real-time work and patients data can be inputted and accessed, doctor will be able to access information relating to a patient like blood pressure, glucose level and symptoms as soon as they get inputted. With this instant access, doctors respond in a more effective way particularly during the time of emergencies, whereby making important decisions within a very small period of time can make the difference between life and death. When the readings fall out of bounds we get the notification or alert and this alerts the doctors that they need to take action in time even when it is not time of regular appointment. The real-time access also saves administrative work, decreases the chances of miscommunication and helps increase departmental care coordination.

1.4.2 To provide personalized health recommendations

This objective will help improve patient engagement and health outcomes in terms of providing tailored recommendations that depend on each patient profile and inputs every day. The system does not use unspecified advice but instead analyses past and present information through AI algorithms that identify patterns, estimates future risks, and provides specialized advice or warnings with regard to health. An example can be given, where a patient who had recorded high glucose levels repeatedly the system can recommend the patient to change his/her diet, or book an appointment with a doctor or remind a patient to take the prescribed medicines. Such individualization based on AI also raises the relevance of the health advice level besides making the patients more motivated to follow through with their treatment plans. Recommendations made using the system to suit individual lifestyles and conditions can enable patients to become more responsible towards their health and enable doctors and nurse to give more effective and informed recommendations.

1.4.3 To Develop a Digital Platform for Hospital Patients

This project aims to change the existing system of writing down the health outcomes of patients in the paper format--including vital signs and other symptoms with the help of a convenient digital solution. The project can make it so that the records will be more accurate, up-to-date, and more easily accessed by the healthcare providers because patients will be able to directly enter the data about their health in the system. It is not only that this shift towards digital input lowers chances of mistakes being made and information being lost but also that it allows to simplify the monitoring process, which allows patients to be more engaged in the process of taking care of themselves and doctors to make more informed conclusions that remain timely.

1.5 Scope and Target User

SmartHealth: Patient Monitoring and Advising Healthcare System SmartHealth: Patient Monitoring and Advising Healthcare System is an Internet-based tool created to increase the effectiveness and timeliness level of healthcare services with the help of digital monitoring and patient-doctor interaction. This project involves the system design, creation, and implementation of a system which enables patients to enter health details manually for example is blood pressure rates, glucose rates, and symptoms. Healthcare providers will be able to see this information in real-time, which will help to follow the patient more closely and identify some health problems as early as possible, as well as provide them with timely medical recommendations. Artificial intelligence is also built into the system allowing it to give individual health suggestions and send on automatic warnings when readings are out of range.

This system is targeted at secondary users mostly consisting of hospital patients and doctors. Patients will have an easy and reachable channel to monitor their daily healthcare statuses and obtain health recommendations. The system will also allow Doctors to monitor the data of patients remotely, give feedback and appropriately make decisions and use information on a timely basis. Understanding the importance of connecting with the patient, SmartHealth will help manage chronic diseases, eliminate unneeded visits to hospitals, and facilitate a healthier life with constant and personalized care.

1.5.1 Project Scope

SmartHealth project scope incorporates design, development, and testing of a web-based platform that enables a patient and a doctor to monitor remote health data and communicate. The system will comprise modules of patient registration, the secure access, data visualization, and the input of the health-related data into the system such as heart rate, blood pressure and symptoms. the automatized warnings of deviations. It will also supply the doctors with a dashboard where they can read patient

history and give them a personal health advice. The system is going to be built with the use of Laravel framework, linked with MySQL and hosted by the help of the web server.

1.5.2 Product Scope

The SmartHealth system will possess the important functions that are expected to assist hospitals in enhancing outpatient treatment and treatment of chronic diseases and are more effective. The patients will be able to register safely, log in, and manually enter their important health indicators for example is blood pressure, glucose, and symptoms in easy-to-use web interface. This health information will be well structured in easy to read tables and diagrams which will enable the patients and the health care providers to see patterns and the changes in health over time. Medically, the doctors will have a special dashboard displaying all updated information of his/her patient so they can see the trends, notice the anomalies, and take timely actions.

The platform will also have a built-in capability that enables the doctors to give direct feedback, health advice or follow-up instructions as a way of improving communication between the patient and the doctor. The system will also feature AI health recommendations, personalized according to each patient, and other input data. This will incorporate an automatic alarm system which will alert medical practitioners when there are critical readings taken by the machine so that the emergencies would be handled as soon as possible. Also, they can be integrated with RESTful APIs, which will ensure the scalability in the future and allow sharing the data securely with other healthcare systems or external services.

1.5.3 Target User

- **Doctor / Nurse**

The system will allow doctors and nurse to track the progress of their patients remotely by looking at a dedicated dashboard that illustrates individual health patterns and alerts. They will get informed about any unusual results in real time and can react by providing corresponding medical guidance or suggesting the appropriate follow up measures. This enables practitioners to provide opportunistic care, reduce response time in times of emergency and to be able to monitor patients all times despite their absence in the facility.

- **Patient**

Patients are the first target users to gain access to a convenient, web-based system where the patient can manually enter and track their health records including the blood pressure, glucose levels, and symptoms. The system is particularly viable to those with chronic disorder and are in

need of frequent monitoring. They will also get tailored health tips and messages related to their past health experiences, which makes them remain active in the process of their health management as it increases the rate of compliance to medication protocols.

1.6 Overview of This Report

This report will describe the creation of the SmartHealth: Patient Monitoring and Advising Healthcare System, a web-based app that will innovate patient care within Hospital Taiping by permitting health data monitoring remotely and providing individualized advice. The report is started with the background of the project, as well as with its objectives and problems statements, and then proceeds to provide a description of the scope of the system, its functions and users it is intended to serve. The report is concluded by an assessment of the results, problems experienced and the prospective improvement and scalability.

2 LITERATURE REVIEW

2.1 Introduction

The chapter features the review of the relevant literature, studies, and current systems connected with the remote health monitoring and digital patient advisory. With the increasing trend in healthcare leaning further to technology-based solutions, the digital health system has attracted attention due to its possible impacts on efficiency and accuracy, as well as its ability to engage patients both in clinical and home-care settings. This literature review aims to understand the challenges of traditional healthcare with respect to manual data entry, slow communication and absence of personalized attention and to provide an insight on the technological solutions offered or applied to counter them. By critically reviewing the past literature and current systems, some justification of the development of SmartHealth system can also be done here by finding the existing gaps and including the necessity of a centralized, convenient and web-focused health monitoring system proper in hospital settings such as Hospital Taiping. The information obtained in the course of reviewing revealed the selection of the system features, technology stack, and design objectives so that the solution is practical and corresponds to user requirements.

2.2 Investigation

This section comprises the procedure adopted in the collection of the applicable information to support the system development of the proposed SmartHealth system. To make sure that the system would cope with the real situation related to healthcare, qualitative research method was chosen. This involves interviewing and gathering opinions of experts who directly deal with the medical profession. The most important aim of the present investigation is to become aware of how the work is done currently and what problems should be addressed in health monitoring as well as get advice which will be used to design and implement the system functionality and opportunities.

2.2.1 Interview With Doctor from Hospital Taiping

During the research of this project, an online interview of the client as a medical doctor in Hospital Taiping was conducted via Google Meet. The aim of this interview was to acquire first-hand knowledge about the existing problems of health monitoring and communication of patients specifically when this takes place in hospitals. In this qualitative research approach, the customer identified a number of pain points, including inefficiency of the manual data entry, the delays in the

updated information about patients, and the lack of individualized digital patients support. The received feedback is the crucial part of justifying the problem statements of the given project based on the feedback and assists in forming the functional and non-functional requirements of the suggested SmartHealth system. This interview did not only help to get a clearer picture of the clinical setting but also guarantee the compatibility of the system design with the practical requirements of medical workers.

2.2.2 Research on Existing Health Monitoring System

To understand more about the current digital healthcare solutions the study carried out a research on some of the more established health monitoring websites like MyChart, PatientsLikeMe and Kyruus Health. With the help of this kind of secondary research, it was possible to evaluate the ways other systems work with patient data, leave feedback and facilitate the dialogue between a doctor and his patient. The objective was to find the best practices, limitations and innovative features to be able to base them on the design, functionality of the SmartHealth system.

This study found that even though most of the platforms are able to provide real-time access and electronic tracking of data, not many of them are designed to comport to local hospital processes such as the ones in Hospital Taiping. Others, such as the example, do not have appropriate personalization, multi-language capabilities or connectivity with the national healthcare systems. These were used to understand where to see gaps that SmartHealth can help with] such as develop a more user-friendly interface with and better experiences with Malaysian patients, enable access of doctors remotely, provide local-based alerts or tips on health practices. Benchmarking of these systems already enables the SmartHealth project to merge them with localized demands, being a more viable and suitable option to the respective context of its application.

2.3 Related Works

The advances in the digital health platform over the past few years have greatly changed and altered the ways with which patients interact with their health information and healthcare providers. In designing a proper and valid system such as SmartHealth, one has to study and evaluate other similar systems that exist with a similar application. In this section we will review three well-known systems MyChart, PatientsLikeMe, and Kyruus Health to comprehend what their basic functions, advantages, and drawbacks are. These platforms also cover various patient engagement areas, which include medical data open access, peer-to-peer support, and smart matching of the care. An analysis of these systems will allow this study to determine some of the best practices and opportunities to incorporate in the features and user experience design of the proposed SmartHealth system. These platforms will offer a comparison and the

lessons learned will inform incorporation of personalized healthcare services, user-friendly interfaces and communication facilities in SmartHealth.

2.3.1 MyChart

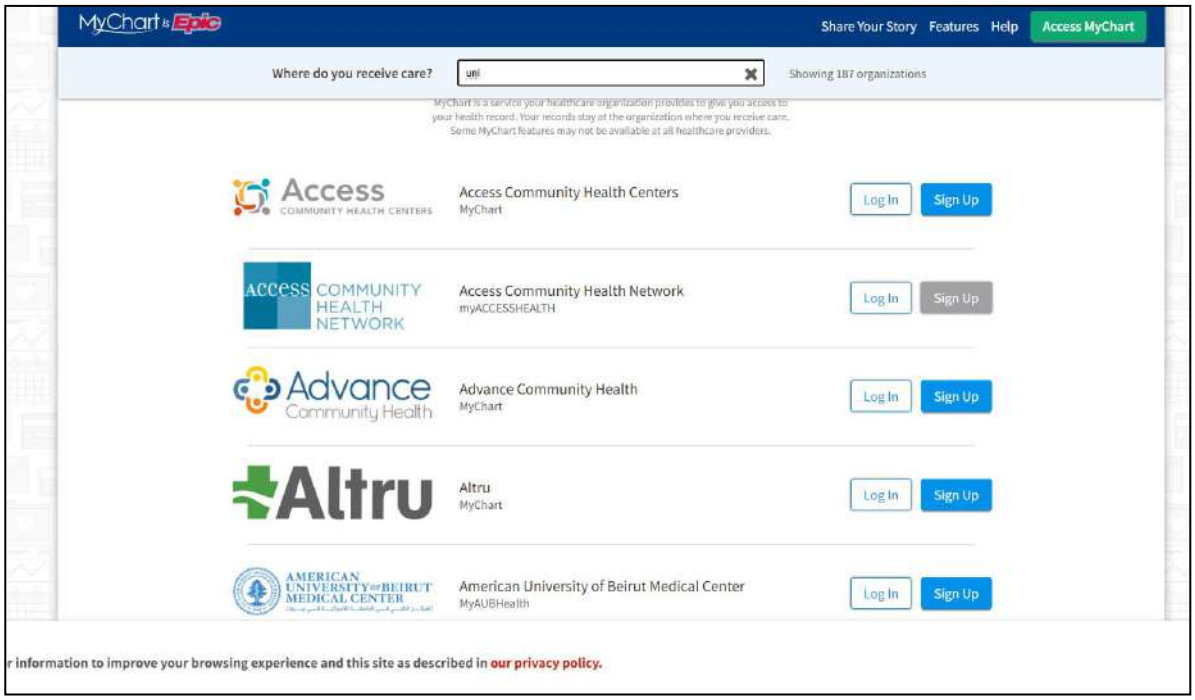


Figure 2.1 : MyChart Website

MyChart is popular among patients and is the patient portal created by Epic Systems, which gives patients an opportunity to view their medical records, connect to the doctors, see their laboratory results, make reservations, and receive the prescription refills. Among the strengths of MyChart is that it is associated with the hospital information systems, which allows experiencing real-time updates and continuity of care. Nevertheless, according to some users, the interface may be complicated and not user-friendly, especially in the case of older patients or less tech-savvy people. In the case of SmartHealth system, experience with MyChart indicates the necessity to provide access to data in real-time, secure interactions, and easy-to-use system.

Among the most attractive features of MyChart, the ability to integrate directly into hospital information systems is worthy to note. This integration is relevant in order to provide the platform with a real-time update of the information presented, which is crucial to the preservation of the continuity of care. Having live access to test outcomes and notes written by their doctors, patients can remain aware of their health condition without visiting the doctor in person on a regular basis. This immediacy does not only allow healthcare professionals to make more efficient decision but enables the patient to be more actively involved in the plan of his treatment, especially when the patient is facing a chronic illness or a treatment period that is a long-term process.

Nevertheless, MyChart has been criticized in terms of their user interface and usability despite the technological gains associated with it. Other users especially the older adults and the people with scanty technical experience have complained of struggles in using the system. Complexity of menus, lack of clarity of labels and excessive information have been referred to as negative factors in the way it can be used effectively. Patient satisfaction and usability can be diminished due to these issues, as it will deter the engagement of the patients due to their frustrations or inability to understand the platform, which can be very confusing to the user.

The example of MyChart in relation to the proposed SmartHealth system of Hospital Taiping has a few important points to consider. First, real-time data synchronization must be on the agenda, so that the doctors can in real time access the health data on the patient and can move to action in case of any irregularities. Secondly, the system should entail secure communication channels that will permit patient to doctor and doctor to patient information exchanges to be conducted privately and encrypted. And finally the interface design should be friendly and easy to use to reduce learning time of both young and old members of the society irrespective of their digital literacy levels.

Understanding the strengths and weaknesses of MyChart will allow creating the SmartHealth system with a better orientation on accessibility, easy-to-use principles, and user focus. This has the goal of developing a technical and clinical-need healthcare platform that is also easy with the view of its patients utilizing the platform in a comfortable and assured manner. Our aim is to provide a system that aids the remote monitoring and it should improve the interaction as well as it should be adapted to the requirements of the various patients of our Hospital Taiping.

2.3.2 PatientsLikeMe

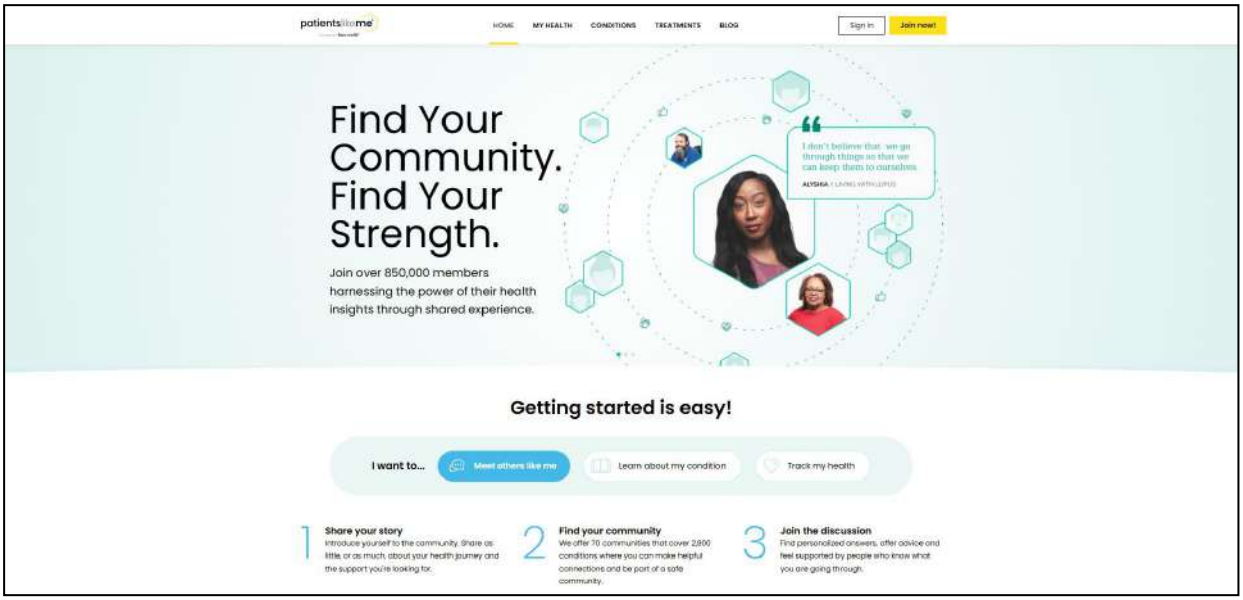


Figure 2.2 : PatientsLikeMe Website

PatientsLikeMe is a digital health platform that focuses on individual requirements with chronic conditions by providing them with an online arena to monitor symptoms, treatment regimes, and contact others with such health conditions. The platform was introduced as a patient-led program and enables user-empowerment by allowing them to write their patient stories and receive insights through collective experience. Its main objective is to enable users to understand each other through comparison of symptoms, treatment and outcomes, which subsequently enables in making better decisions in taking care of themselves.

The ability to support the community is one of the most prominent features of PatientsLikeMe. As opposed to conventional healthcare websites, which require a significant level of direct contact with the working routine of a medical professional, PatientsLikeMe focuses on peer-to-peer interaction to establish a common ground and relatability. Patients are able to keep close friends apprised of their personal health situations, get a cheer, and offer tips on how to go about their illnesses. The latter is particularly useful in dealing with chronic or complicated conditions where either the lived experience or emotional support may be just as valuable as clinical recommendation.

A feature provided by the platform is also the ability to make donations, and those who volunteer to do so can make donations of their anonymized data, which can be used in research to help understand more about disease trends, treatment outcomes, and patient behavior. Such a model of citizen science can be used to supplement medical research, including the experiences of real patients in data analysis, resulting in more comprehensive and useful data. Nevertheless, although this is a research-driven model, it is not integrated with the real-time clinical care. No live data sharing with doctors or health institutions is available on PatientsLikeMe, which leaves users in the same position to rely on other platforms or real-life meetings with professionals to receive medical assessment and subsequent change in treatment.

In spite of its flaws in professional medical integration, the success of PatientsLikeMe proves the role of personalized feedback and user control in medical sites. The feeling of being in control of their health process, along with emotional and plentiful informational support of a virtual community, has proven to enhance treatment stickiness and general fitness. It highlights the increase in the need of the patient-centric tools with a focus on collaboration, openness, and mutual understanding.

2.3.3 Kyruus Health

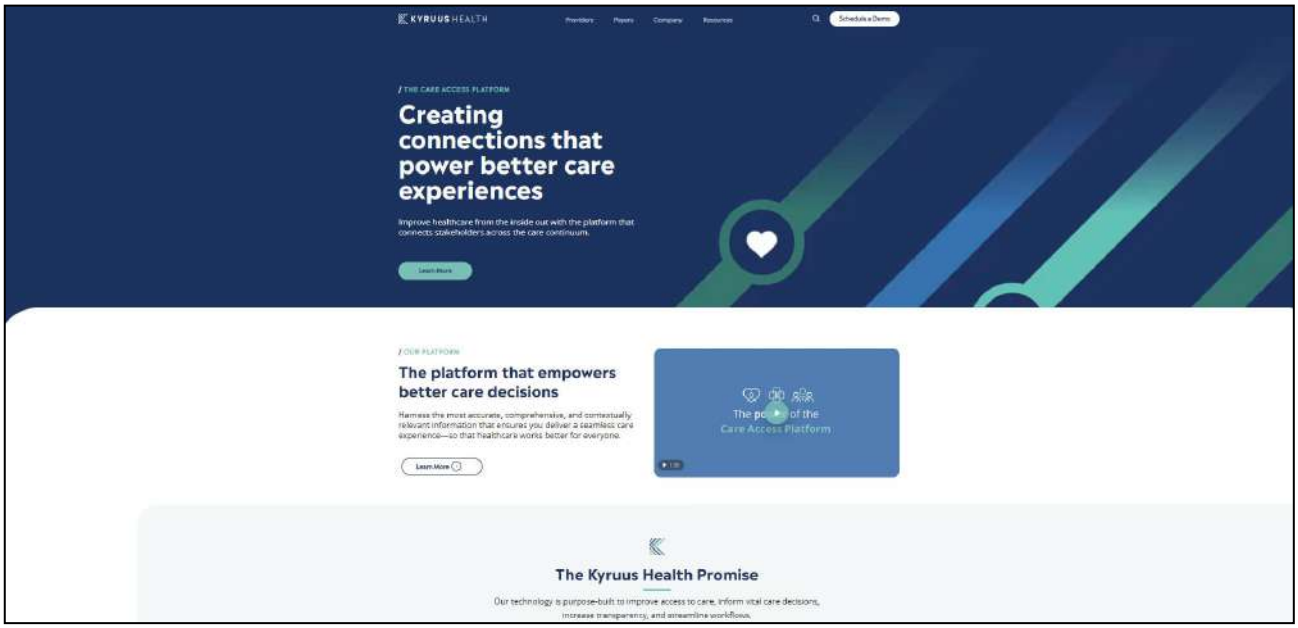


Figure 2.3 : Kyruus Health Website

Kyruus Health is a medical technology business firm that is specialized in organizational ease of matching patients to the best medical experts. Its main functionality is based on intelligent provider search and smart scheduling tools applied by hospitals and health systems with the aim to match patients with the most suitable doctors according to their specialty, availability, location, and patient preferences. This is a service where one of the most critical pains in healthcare exists; which sometimes patients have problems finding the appropriate provider with ease and haste. Through the use of advanced algorithms and data filtrating, Kyruus helps patients to obtain care in time and in a proper manner optimizing the appointment processes at healthcare establishments.

Among the various contributions of Kyruus Health, one of the major contributions is balancing between the supply and demand of healthcare services. It achieves this by providing a centralized platform within which patients can peruse and pick among numerous options of healthcare professionals as hospitals are able to better manage slots during appointments. This helps in allocating resources better and minimized waiting times by the patient thereby enhancing the experience of healthcare delivery. By improving the rates of no-shows and minimizing conflicts in

schedules, the smart scheduling feature not only makes the patient experience more convenient but also can bring benefits to the healthcare providers, making their operations more efficient.

Kyruus Health, however, is not the solution aimed to monitor patients in real-time or track clinical data. In contrast to other types of systems that are associated with storing patient health records or providing them with constant health updates, Kyruus is a more of a system used at the front-end to make appointment and provider selection. This restricts its use in cases of a situation that requires continuous medical support e.g. chronic disease management. Although it is vital in ensuring the passage of patients to the appropriate provider, it does not capture, analyze and render feedback to relate to the daily health trends of the patient and therefore it does not provide feedback that is relational to health trends of the patient.

2.4 Comparison

The comparison between MyChart, PatientsLikeMe, and Kyruus Health showed some of the main differences and advantages that can be used in development of the SmartHealth system. The MyChart is highly integrated with hospital systems, continuously updated, and secure messaging, which makes it perfect to use in a clinical setting but the interface might be complicated to some users. PatientsLikeMe deals with community support and highly customized tracking by user input, offering a good engagement and no real-time doctor connect. Kyruus Health has a unique position in provider matching and scheduling, simply connecting to care without direct health monitoring. The design of SmartHealth may be based on the examination of those systems with regard to their applicability across the following characteristics: usability, security, personification, output, allowing to create such a health monitoring system that will not only fill the gap existing in the existing solutions but also will be more complete and accessible to most users.

Table 2.1 : Comparison of Existing Project

Criteria/Project	MyChart	PatientsLikeMe	Kyruus Health
Features	Appointment scheduling, test results, secure messaging	Patient community, symptom tracking, condition sharing	Provider search, scheduling, care navigation
Users	Patients, doctors	Patients with chronic conditions	Patients, hospitals, healthcare networks
Navigation	User-friendly with tab-based menus	Interactive and forum-style layout	Search and guided interface
Security Element	HIPAA-compliant, encrypted access	User consent control, basic data privacy	Enterprise-grade security, data compliance tools
Result/Output	Health summaries, real-	Community insights,	Matches to care

	time test results	treatment outcomes	providers, analytics dashboards
Platform	Web and mobile app	Web-based	Web and enterprise systems
Personalisation	Moderate personalization based on medical history	High – based on patient input and condition	Focused on care matching

2.5 Discussion

Following the analysis of MyChart, the access to real-time information is one of the most valuable features that will be implemented into the SmartHealth system. Depending on how well MyChart is compatible with systems in hospitals, anyone visiting a hospital as well as a doctor should always get the latest information on their patient because this is vital in the management of patients. Nonetheless, the complexity of MyChart and unintuitive interface particularly to elderly or not tech-savvy individuals emphasize the need of having a simpler and easy to understand interface to follow in SmartHealth. As such, usability will be a major concern in the suggested system to guarantee that the patients of different ages and background are able to use it without difficulties.

Based on PatientsLikeMe analysis, the SmartHealth system should include the personalized feedback and community engagement tool. PatientsLikeMe is sure to thrive at providing patients with opportunities to find and connect with others who share their health experiences and benefit greatly to patient morale and motivation. SmartHealth is unlikely to perfectly recreate the community structure but will provide domain-specific health tips, user progress tracking, and other elements that give the user a feeling of being part of the care, being unique and important to the problem. These individual features are to raise patient adherence and system usage continuity.

The Kyruus Health company focuses on smart provider search and matching the right care to the right providers and provides a source of inspiration realizing the best user experience in SmartHealth. Even though Kyruus is not a health monitoring system its capabilities to match users with the suitable healthcare system can be interpreted in terms of SmartHealth capabilities in matching patients with their favorite doctors and specialists when a patient needs them. Additionally, the well-organized line of communication between patients and the delivery healthcare services will guarantee that users will get pertinent medical counsel in case an irregularity is identified in their health data.

2.6 Conclusion

In conclusion, the literature review has given detailed information related to the existing situation of digital health monitoring and advisory structures. It has emphasized the current transition of the traditional manual health data procedures to more effective, computerized, and patient-friendly healthcare solutions. Thanks to the conducted investigation, several key problems were singled out, especially through the interview of a medical worker at Hospital Taiping, including a delay in information updates, a lack of personalization, and manual data entry, upon which the basis of the SmartHealth system should be created.

Despite the fact the reviewed related works such as MyChart, PatientsLikeMe, Kyruus Health include totally different features and restrictions, it was found that there are some valuable attributes that directly impact on the proposed system. MyChart proved how a real-time data access and integration into the records of hospitals is important, and PatientsLikeMe proved how personal assistance and motivation through peers are valuable. Kyruus Health demonstrated new approaches in personalizing users and matching healthcare providers, and these can be learned in order to apply this to SmartHealth user experience to optimize interaction between patients and their doctors.

The comparison of such platforms has revealed the existing gaps, including the complexity of user interface, the absence of the real-time interaction with a doctor, and the fragmentation of functionalities of the system. These observations are the reasons why there should be something easy to understand, available to everybody and customizable to patient and clinical requirements such as SmartHealth all-in-one solution.

3 METHODOLOGY

3.1 Introduction

This chapter explains the methodology that was used in developing the SmartHealth system that will offer an efficient and easy to use health monitoring platform to Hospital Taiping. The selection of an appropriate methodology is essential in taking the project through a systematic phase as each of the stages of development would be done systematically to produce excellent results. The methodology involves an immediate influence on the organization and implementation of the system delivery and planning, particularly in case of the healthcare sphere where accuracy, security, and reliability are at the center of attention.

Agile model has been selected in this project due to the ability of Agile model to incubate iterative development, regular response of the stakeholders, and constant progress as important aspects of developing a dependable healthcare system. According to Alotaibi and Almudhi (2023), Agile has gained significant popularity in healthcare as it places a higher priority on iterative cycles, team-based collaboration, and the ability to respond to the change. Their work places an emphasis on the way Agile processes could lead to the better outcome of work and patient improvement in outcomes as well as the efficiency of work in clinical setting by means of assisting continual improvement and flexibility in clinic setting. Without Agile in the SmartHealth project, the Hospital Taiping would miss out on the opportunity to have constant user feedback and gradual system verification, which makes the platform viable and adaptive to the problematic conditions of healthcare.

3.2 Agile Methodology

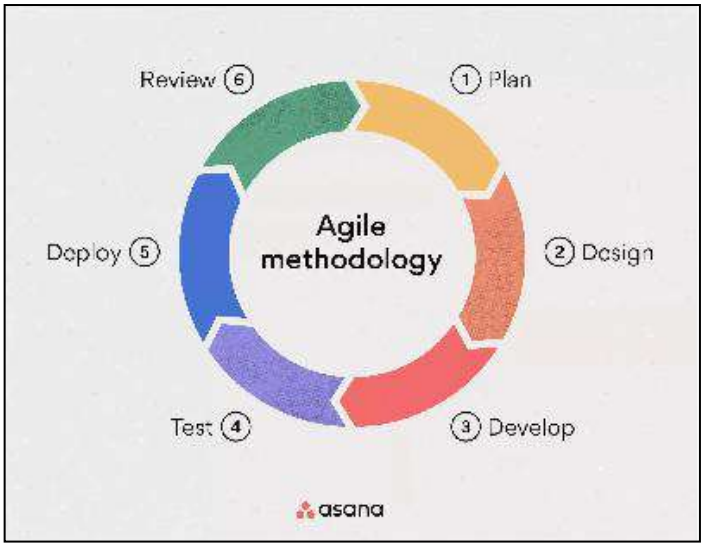


Figure 3.1 : Agile Methodology Model (Laoyan, 2025)

The choice of the methodology in software development is a crucial outcome because it has high impact on quality and adaptability of the resulted system, its overall success. A methodology is a structured approach which allows controlling the phases of the Software Development Life Cycle (SDLC) particularly in the most sensitive industries like healthcare where precision, safety and adaptability is the primary focus. Compared to the more traditional approach, such as Waterfall, Agile methodology is becoming more accepted owing to its rolling nature, its malleability, continuous user input, and feedback. Alotaibi and Almudhi (2023) also stated that Agile framework supports dynamic requirements, fosters collaboration among the teams, and facilitates continuous improvements which makes it especially suitable in healthcare settings where patient demands and medical guidelines might change throughout time. The main advantage of this iterative type of model is every development cycle (or sprint) has working software that can be reviewed and enhanced resulting in fewer risks and stakeholder involvement.

As demonstrated in figure 3.1, Agile methodology has been selected to enable ongoing communication with medical workers and iterative setup consistent with the remarks received in reality. Daily input of health data made by the patients, the communication between the doctor and the patient, their alerts, and real-time monitoring are the primary functions of the system that is frequently required to be validated and responsive to the changing user expectations. Agile makes it easier by chopping up development into brief development sprints where functions can be built, tested and checked and refined. This will assist in making the system user friendly, secure and reliable during the development. Finally, Agile enables medical stakeholders and developers to develop a digital health tool collaboratively, which meets the needs of the functioning and the real-time work in a hospital environment.

3.3 Phases in Agile Methodology

Agile methodology is a software development model that involves working on iterations and increments to the software and focuses on flexibility, customer cooperation and quick deployment of functional software. As compared to the sequential structure of the Waterfall model, Agile ensures maintaining certain iteration as sprints, short cycles leading to the creation of working modules. Agile is particularly useful with healthcare system development where the user might also add different requirements due to user feedback as well as a real-world limitation. The Agile methodology in the SmartHealth project is to provide flexibility in order to adjust the system functionalities as iterative consultations with the hospital doctors and patients at Hospital Taiping to be made. Figure 3.1 shows the stages of Agile lifecycle used in this project.

3.3.1 Plan

Agile cycle starts with the planning phase. It entails the determination of what product backlog is, finding out the most important features, and estimating the level of effort needed to complete them. In this case, this stage was the gathering of input requirements of the doctors and the patients in the form of questionnaires and interviews on behalf of SmartHealth. These inputs allowed establishing the minimum viable product (MVP) and focus on key functionalities that included health data submissions, doctor dashboards, and generation of alerts.

During this phase, the input of developers and stakeholders was important. Sprint planning meetings were held to make decisions on the which tasks were to be implemented on the very first iteration. The desire to be flexible inspires Agile, which means that further emergence of feedback or other insights during the development may alter the future sprint plans without endangering the project schedules.

3.3.2 Design

In the design process, the group contends on the objective of coming up with simple and modular parts that can be structured and tested in brief sprints. In the case of SmartHealth, UI wireframes and workflow diagrams were created to describe the interface of the system which would be used by patients and doctors. The design procedure focused on being user-friendly even to patients who might not own the machines.

Its system architecture was designed to be modularized--each feature such as patient logins, inputs, and viewing doctor reactions was designed on its own Merely a few clicks and the existing versions of the same functions can be added to; or a new version can be included; or an error can

be corrected, etc. It was also due to this modular design that allowed quicker, later-stage testing and debugging.

3.3.3 Develop

Developing is the second part of Agile methodology where the basic functionality of the system is developed over repeated sprints. In the case of the SmartHealth system at Hospital Taiping, the first step was to establish the base of the application on the Laravel framework, which is based on the PHP programming language, has a clean syntax with a set of built-in security features, and MVC design. Laravel was selected because it is effective in managing routing, authentication as well as communication with the database and thus suitable in the development of secure and scalable backend. Capabilities of patient registration, physician logins, health data storage, and feedback processes were developed one by one within the scope of each sprint cycle.

To ensure the front-end part is well developed, a user-friendly and responsive interface must be designed using HTML and CSS. The technologies assisted in organizing the system layout and design its structure to work over devices. JavaScript was incorporated to achieve interactivity where real-time validation of the filled-in forms was adopted, patient data in and out confirmations done, chart displays shown to help vendors understand their trends with health records. The aspects played a critical role in ensuring that the system was easy to understand by people with varying degrees of digital literacy such as senior patients.

All development works were done using the Visual Studio Code (VS Code) as the code editor. VS Code was chosen because of its integrated development environment, Laravel and JavaScript support and an in-built terminal and Git integration that simplified the process of collaboration and managing the codes. Agile steps complemented both with modern development tools had helped to gradually mold SmartHealth system into robust, scale-able, and secure healthcare platform that can effectively take care of the needs of both the doctors and patients.

3.3.4 Test

Deploy phase of the Agile methodology is associated with deployment of actual working system by the end of each sprint or upon delivery of key features. In the case of SmartHealth implementation in Hospital Taiping, the system was simply rolled out gradually so that the system could obtain real-time user feedback and stabilize before its massive roll-out. Laravel infrastructure was utilized and a secure web server was deployed to host the application; migration and environment settings are done using inbuilt tools of Laravel. The MySQL database was installed such that patient records, medical history, and the real time health inputs were stored safely. Also, required SSL credentials were installed to secure the HTTPS access and the data encryption. Prior

to going live, early test environments were replicated to replicate real hospital processes, and it was ensured that all needed services including but not limited to patient logins, patient health data submissions, and doctor dashboards were working as expected.

In the implementation process, access credentials were created to support both the doctors or nurse and patient. Training contents and simple user instructions were also added to assist end-users on system onboarding. SmartHealth was initially tested in a single department of Hospital Taiping to check how users behave, what problems may arise, and what can be suggested to improve the situation. One of the benefits of this staged rollout is that implementation is safe and controlled and does not disrupt regular healthcare operations so much, and continuity of care is present. Proper backups, the logging system and system monitoring systems were also implemented to make sure that the performance and data integrity remains intact during the deployment.

3.3.5 Deploy

The testing stage in the Waterfall model is very important, because it guarantees the quality, capabilities and stability of the developed system on the whole. In the case of the SmartHealth system, there are other forms of testing done in order to ensure that every component is working fine and that it interacts without any problems. This is Unit Testing which involves testing of individual modules e.g. patient data entry or doctor dashboards individually; and Integration Testing which involves making sure that modules do not conflict when tested together. This is followed by System Testing which is done to ensure that the whole system is operating in line with what had been documented.

The other important step would be the process of User Acceptance Testing (UAT) which involves the actual users such as the doctors, patients to interact with the system so as to ascertain their practical requirement and expectation. Besides functional testing, performance bottlenecks, security loopholes and usability problems would also be tested on the system. As SmartHealth deals with confidential health data Security testing is particularly important in preventing unauthorized access and data leakages. Every bug, error or usability issue found in the testing part is well documented, their level of priorities taken into considerations and eliminated by the time the system gets into the deployment phase. This will make the end product stable, secure, and fit to be implemented in the Hospital Taiping.

3.3.6 Review

Review phase in the Agile is important in its ongoing improvement process based on user feedback and performance analysis of the system. During deployment, SmartHealth system received several rounds of feedbacks by patients and doctors to gauge the usability, functionality, and reliability of the system. The retrospectives under the agile approach took place at the end of every sprint to examine the areas that were successful, determine the bottlenecks, and how they could improve. To illustrate, hospital personnel comments triggered the improvement of the location of the health data graphs on the screen, and patients sought more noticeable pointers regarding the need to provide updates on their condition on a daily basis. Such reviews were used to prioritize future releases and guaranteed that product would be improved according to the actual user requirements.

3.4 Conclusion

Agile methodology can provide agile development that is iterative and adaptable, which is in line with the needs of the healthcare systems that change rapidly, such as SmartHealth. Agile, unlike traditional models, provides the possibility of constant feedback of actual users, in our case, doctors and patients of Hospital Taiping, so that each version of the system is ever more polished, user- focused, and related to the real workflows. This flexibility is especially useful in medical facilities where priorities can change and usability of the subsystem can determine day to day business.

Agile also allowed the SmartHealth project to have development broken down into small sprints and separate phases, with each phase providing a potentially subjected working component that could be tested, reviewed, and refined. This assisted in ensuring that the system was in line with clinical goals with minimum risks and expensive rework. The collaboration involved in Agile, early testing, and incremental delivery efforts have helped to advance a more secure, interact, and patient-centered system that can enhance the delivery of healthcare, closer doctor-patient interactions and assist in more effective monitoring and better decision-making at Hospital Taiping.

4 REQUIREMENTS

4.1 Introduction

Identifying the needs and expectations of the user is a crucial component of every system development process and is of particular concern in the healthcare sector where solutions need to be both useful and dependable. This chapter is concentrated on the requirements gathering and defining the process of the SmartHealth system that is supposed to help the Hospital Taiping to better manage the health data about its patients. The primary goal of this phase is to equalize what the system must perform (functional requirements), how the system should carry it out (non-functional requirements), what software and hardware should the system run on (system requirements), and note it down. These requirements make it a certainty that the system will have the right functionality as well as attain the usability, security, and performance requirements.

The process of requirements gathering included discussions with medical professionals which is doctor as well as patients so as to create a system that will effectively address the needs of the people. Such interactions were carried out by means of interviews and questionnaires in order to have a general picture of existing problems as well as anticipations concerning the health monitoring. The data accumulated assisted in verification of the project objectives and influenced functionalities and requirements of the SmartHealth system. Finally, such needs are the basis of designing and development of the system in line with the Waterfall approach.

4.2 Data Gathering Techniques

To create a system that would meet the needs of users in the real world, the right, practical, and relevant data must be obtained through the target users. In the case of the SmartHealth system, a multifaceted data collection method was implemented so as to give extensive knowledge in the issues that affected the current health monitoring and communication systems. This mixed method gave the project the chance to record both the general user expectation coupled with the profoundness of knowledge by professional stakeholders.

There were two types of methods used, a questionnaire was sent to patients to collect quantitative and user-specific information, and interviews were performed with a medical expert in Hospital Taiping to

learn about the practical constraints, system gaps, and user needs from a clinical point of view. These techniques were used jointly as they enabled indicating the statistical data in one case, and going in more detail with a reflected opinion providing more solid backgrounds to make decisions concerning the design of the systems.

4.2.1 Questionnaire

The questionnaire was developed to collect information of patients who would be the possible end-users of SmartHealth system. It consisted of multiple-choice and open questions split into a number of sections, which contained demographic information, current experiences regarding health monitoring, system preferences and user feedback. The purpose was to determine how patients have been monitoring their health and what they are struggling with as well as what they expect a digital health tracking solution to do. There is four section in this questionnaire which is demographic information, current health monitoring and advising experience, system preferences and features, and feedback and suggestion.

The questionnaire was administered on the internet and was anonymous so as to prompt people to be sincere and as varied as possible. The most typical answers were to forget to record health information, having trouble comprehending general medical instructions, and a preference to see more user-friendly, personalized digital interventions. Moreover, the participants were interested in such features as reminders, the possibility to connect with doctors in real-time, and visual monitoring of health. The functional range of the system and issues related to its usability were informed by the outcomes of the questionnaire.

4.2.2 Interview

In the form of an online interview, a medical doctor working in Hospital Taiping in the role of the Hospital Taiping primary client was interviewed through Google Meet to provide the project team with professional information about the hospital operations. The client name is Dr Nurulraihan Binti Mohamed Zawawi. She is the Head of Transfusion Medicine Department at Hospital Taiping, Perak. There is five question being given to the client. The aim of the interview was to learn how the health information of patients is managed at present in a hospital environment and what are the current problems, inefficiencies in the process of manual data entry and disruptions in the timely updating of patient information, and the absence of a personalized digital communication doctor-patient.

The interview was semi-structured, and it allowed free communication and at the same time touched upon certain issues concerning the proposed system. The contribution of the doctor helped

in putting forward the realistic features of the system like real-time syncing of data, accessibility by the remote user, security, and need to have user-friendly interface that would be suitable to application by staff and patients as well. It helped determine the functional and the non-functional requirements of the system, and this made the design of SmartHealth commensurate to the needs of the hospital.

4.3 Functional Requirement

The functional requirements state which behaviours, features, and functions, the SmartHealth system is required to execute as part of meeting its intended purpose. These needs were obtained by interviewing the medical workers and by questionnaires distributed among patients in Hospital Taiping. The following table gives the description of the main functions of the system and their anticipated results.

Table 4.1 : Functional reruqirement for patients

Function	Expected Result
Login and Registration	They will be able to make a personal account and log in to it safely
Health data entry	The patients have the option of manually entering data about health, including blood pressure, glucose levels or symptoms.
Personalized health tips by artificial intelligence (AI)	Patients are given recommendations or warnings on the basis of their matched health data trends from Artificial Intelligence (AI)
Reminder notifications	System will have notifications to remind patients when to log health data or take medication.
Appoinment schedulling	The patients will be able to see the available time and request appointments with the doctor.
View Health History	Allow patients to see submitted medical history and health information in a structured form.
Print Health Records	Patient can print the result of his own health data.

Table 4.2 : Functional reruqirement for doctors/nurse

Function	Expected Result
Login	Allows doctors to log in safely with authenticated identities, to gain access.
Dashboard patients data	Offers a dashboard view of system utilization, activity of patients, and system alert relevant to the doctor.
Monitor Real-Time Health Data	Able to give the doctor a real-time review of the health information that a patient posts.
Send Feedback and Health Advice	Doctors are able to provide individual feedback or assistance depending on the patient-submitted data.
View Patient Profiles	Allows doctors to retrieve and examine personal records of the patient such as history and background.

4.4 Non-Function Requirement

Non-functional requirements are those that dictate the standards of performance and quality of the SmartHealth system. These requirements do not indicate how the system will be implemented but, on how it will react and perform to diverse situations. They make the system efficient, secure, scalable and easy to use, especially when it comes to a health application that will be used by patients and doctors of Hospital Taiping.

Table 4.3 : Non-Functional requirement table

Non-Functional Requirement	Description
Performance	The system is expected to take a shorter time to react to the inputs made by a user like a log in, data entry and others
Avability	It should have a 24/7 accessible system particularly in real time monitoring and alerting of health data.
Scability	As the number of users and health records increase, the system must be capable of processing them without compromising.
Security	All the user data should be encrypted, and the access must be based on the roles with proper authentication protocols.
Usability	The interface should also be comfortable to use and simple to follow, particularly by older patients or non-technological users.
Reliability	When in full use, the system must not crash, or cause unacceptable errors.
Maintanability	The code structure must be well organized and documented so as to make future updates and maintenance easier.

4.5 System Requirement

The SmartHealth system involves a combination of software and the hardware element to facilitate a smooth development, deployment, and operation. Such needs are critical in accommodating the administrative (doctor/nurse) roles and the user (patient) entry to the platforms, especially in a hospital facility such as the one in Hospital Taiping.

4.5.1 Software Requirements

i) Visual Studio Code

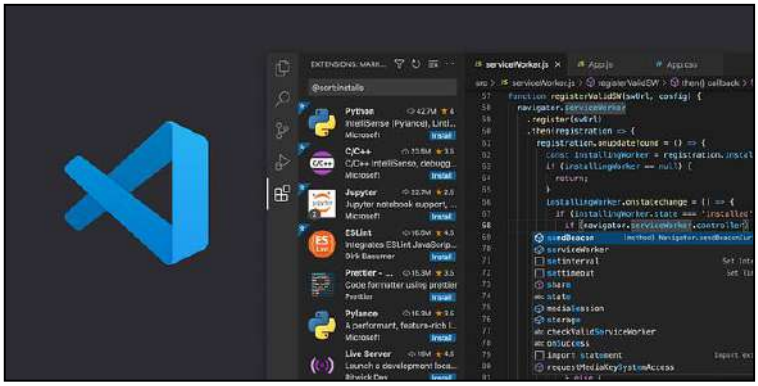


Figure 4.1 : Visual Studio Code (Enejivik, 2023)

Visual Studio Code is a modern, fast, and versatile source-code editor by Microsoft that embraces modern development workflows and enables code extension capabilities by offering integrated and continual development workflows using its built-in capabilities to extend the editor through plug-in extensions. Its rich extension ecosystem allows a smooth integration with other languages and frameworks such as PHP and Laravel by providing an improved code completion, linting and debugging functionality. Misek (2022) states that new IDE features provided in PHP Tools for VS Code, including Laravel facade and inline debugging, represented significant improvements to the code. Its performance, modularity, and flexibility make VS Code one of the most preferred options of web and application developers in the field of healthcare or any other industry.

ii) Laravel

Figure 4.2 : Laravel Framework (Barnes, 2021)

Laravel is a free framework written in PHP used to facilitate the ease and fast development of web applications with in-built features like routing, authentication, and session management. It is also a clean and maintainable code that takes the MVC (Model-View-Controller) architecture. RESTful APIs are also supported in Laravel, which is crucial to the health system of today, that demand real-time communication of data between the client and the server. It provides productive

development environment with such features as Blade templating, Eloquent ORM to work with databases and Laravel Mix to compile assets.

4.5.2 Hardware Requirements

Table 4.4 : Hardware Specification Table

Device	Dell G15 5510
Windows Edition	Window 11 Home
Processor	12th Gen Intel® Core™ i5-12500H
Memory (RAM)	16 GB
System Type	64-bit operating system

4.6 Conclusion

This chapter has described the requirements gathering role and the precise technical requirements that were to be used to make the SmartHealth system. By performing interviews with medical staff along with the distribution of questionnaires to patients, all the functional and non-functional requirements were estimated and recorded. These results are used as a guideline in developing the system to establish that the system is compatible to the actual medical practices in Hospital Taiping.

Software and hardware situations that are required to support the execution of system were also outlined in the chapter. Having enumerated these requirements distinctly, developers will be able to make sure that the SmartHealth system is efficient, aligned with the security standards and being able to provide both doctors and patients with positive user experience. Such specifications are modern foundations of designing and development of the system in subsequent chapters.

5 ANALYSIS

5.1 Introduction

The chapter will provide the chapter of analysis of the SmartHealth system that is important in transforming the user requirements and business expectations into systematic and workable elements that become a guide in designing and developing the system. The analysis phase is concerned with the data flow on the system including the functional and non functional requirements of the system, how the system users will utilize the system and what kind of a data structure should be adopted in relation to the system user interaction. This will assist in ensuring that not only is the system technically viable but at the same time they deliver in the expectations of the end users of that technology; that is, doctors and patients at Hospital Taiping.

Various tools and techniques were also employed in conducting this analysis and these were questionnaires, interviews, use case modeling, system flowcharts, and data structure diagrams. These models assist in picturing the system and its behavior that should be expected and serve as guidance to the developers, designers, and the stakeholders. In particular, use case model reflects key system functionality and user interactions, whereas the flowchart describes the logical progression of the operations. This will be the reference point of designing a responsive, secure, and efficient healthcare monitoring system ready to face the practical setting of Hospital Taiping.

5.2 Data Gathering Analysis

Data collection occurred early in the project to make sure that the SmartHealth system will satisfy the real-life needs of its users. This part entailed the gathering of useful information among the patients and the medical practitioners at Hospital Taiping. Data collection involved two main methods which are questionnaires and interviews. Fifteen question given to user in questionnaire and 5 question given to client. The decision behind these techniques is that they are effective to gather user behaviors, needs, and pain points, particularly, in a healthcare mode where the precision and clarity are critical.

5.2.1 Questionnaire Analysis

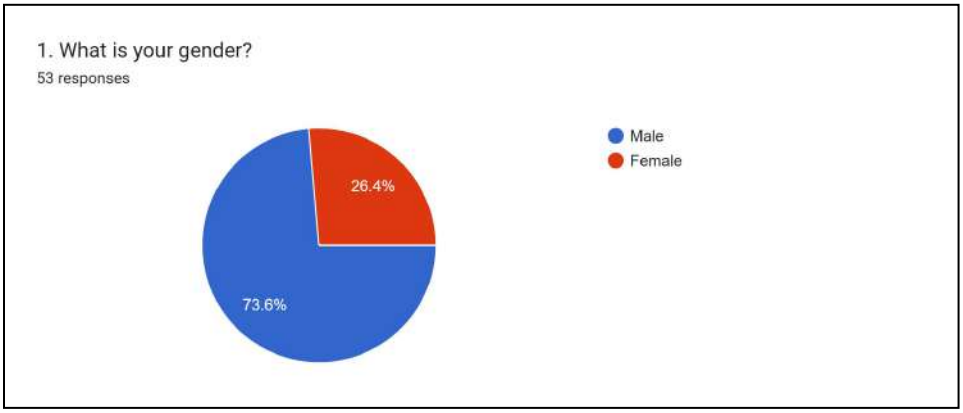


Figure 5.1 : Questionnaire question 1

This pie chart shows among 53 responses, there is 73.6% is a male that reponse this questionnaire and 26.4% is a female that response this questionnaire.

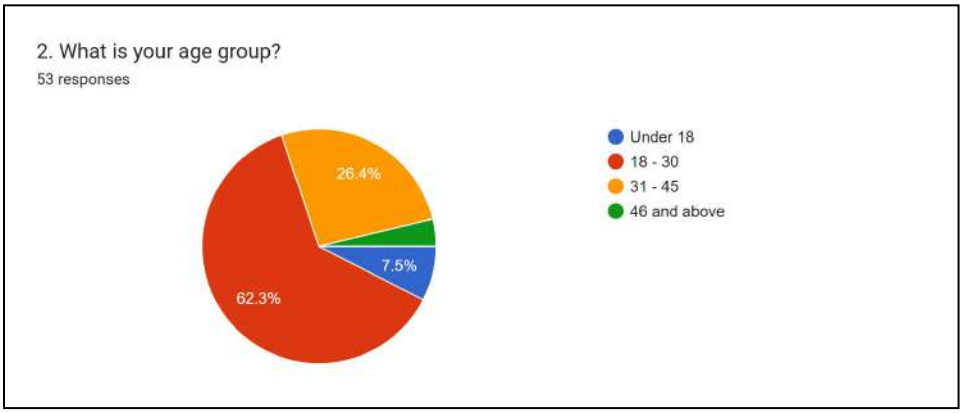


Figure 5.2 : Questionnaire question 2

According to the pie chart indicating the distribution of the age groups represented in the 53 respondents, most participants were between the ages of 18-30 years representing 62.3 percent of the total. It means that the majority of the possible consumers of the SmartHealth system would be the representatives of young generations of adults

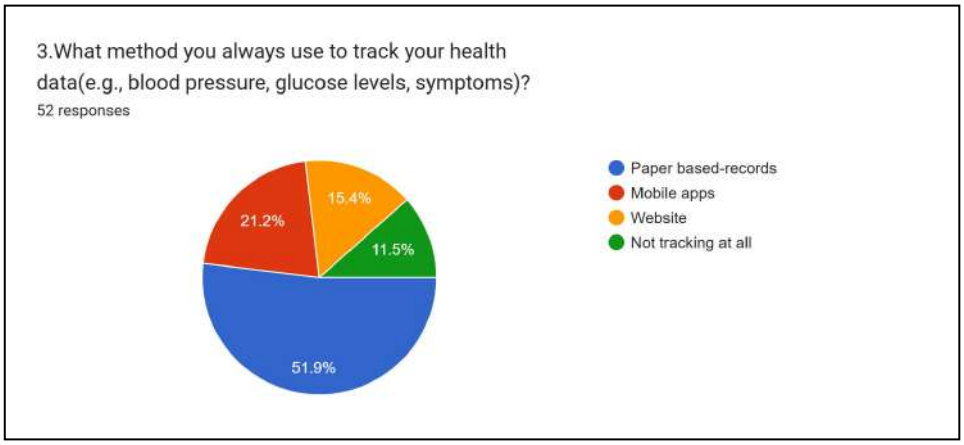


Figure 5.3 : Questionnaire question 3

The pie chart shows the results of a survey conducted among 52 people concerning their preference of the ways of monitoring health data such as keeping blood pressure, glucose levels, or symptoms. The dominant group 51.9% comprises individuals who have shown experience in utilizing paper-based records to enter their input health data.

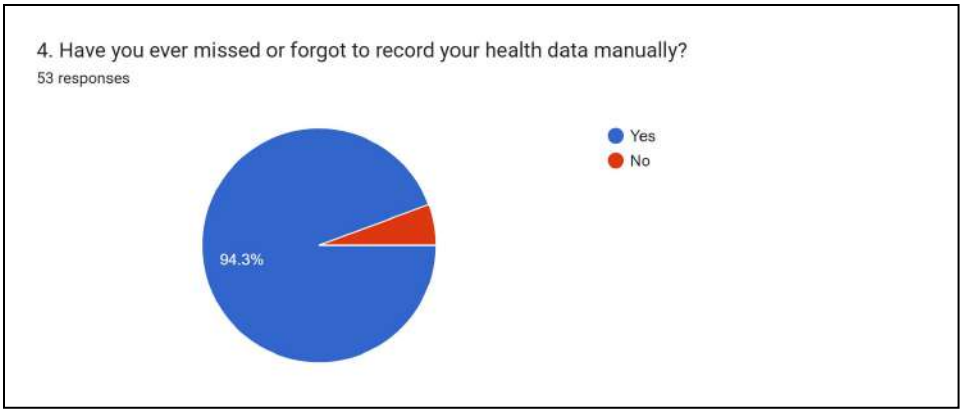


Figure 5.4 : Questionnaire question 4

This pie chart shows that the majority of repondent which is 94.3% sometimes or always forgot to record the health data manually. This chart confirms that respondent maybe forget or busy to keep maintance track their health.

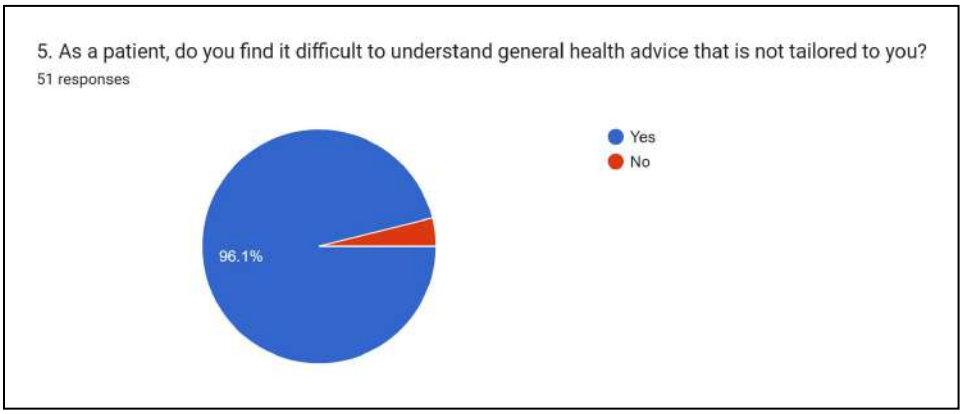


Figure 5.5 : Questionnaire question 5

The pie chart shows the answers of 51 patients on whether they could not easily comprehend general health advices when they are non-personalized. The findings indicate that 86.1 percent of the respondents replied with the Yes option, which demonstrates that most were having difficulties with health advice that was not customized to their personal needs, whereas the other 13.9 percent gave their response as a No. This also demonstrates that there is a high demand of increased personalized health communication to enhance patient understanding and participation.

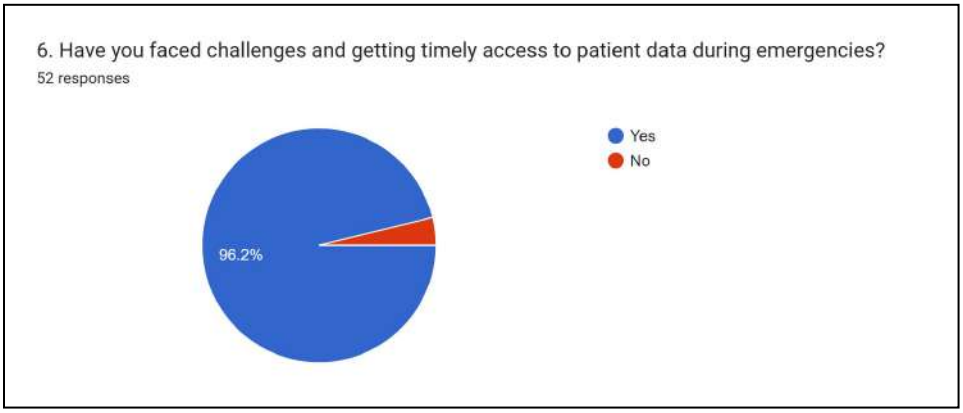


Figure 5.6 : Questionnaire question 6

The findings are that 95% of the respondents voted Yes making it a near-consensus that people struggle to get the information they need in this regard very fast especially in emergencies. The other 5% chose No. This dominant majority indicates structural challenges in the accessibility of emergency care data, which makes it clear that better systems are necessary, to guarantee rapid and reliable access to information in the case of crisis.

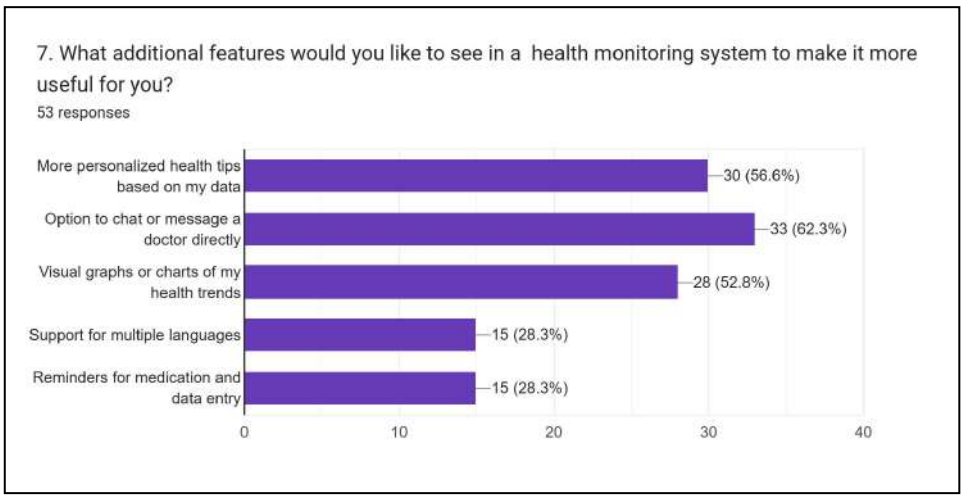


Figure 5.7 : Questionnaire question 7

The bar chart shows what the respondents want to have in a health monitoring system, the 53 respondents, indicating outstanding priorities towards existing improvements. The most requested feature is the possibility to message doctors directly, and 62.3 percent which is 33 votes of the participants choose this option, which shows the necessity to support communication through messaging because of acute health.

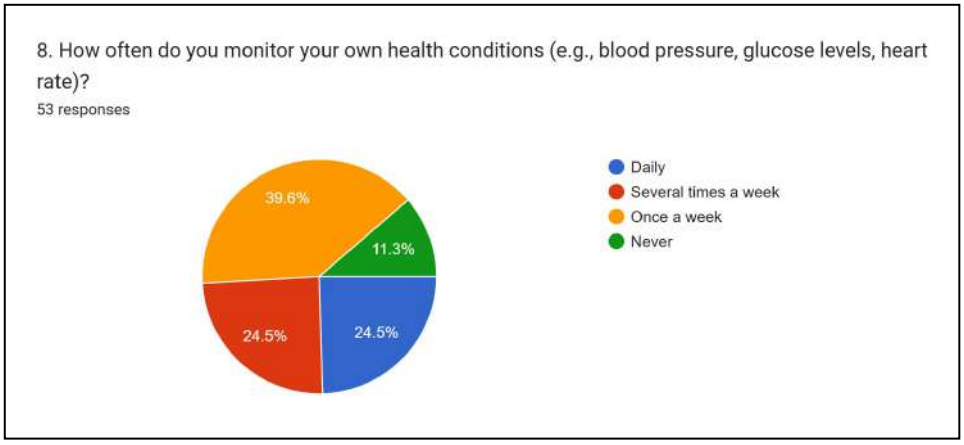


Figure 5.8 : Questionnaire question 8

The pie chart shows the results of the survey of the 53 people on how frequently they monitor their health in relation to their blood pressure level, glucose level or their heart rate. According to the findings, most of them review their health once a week and this number amounts to 28.6 percent of the respondents

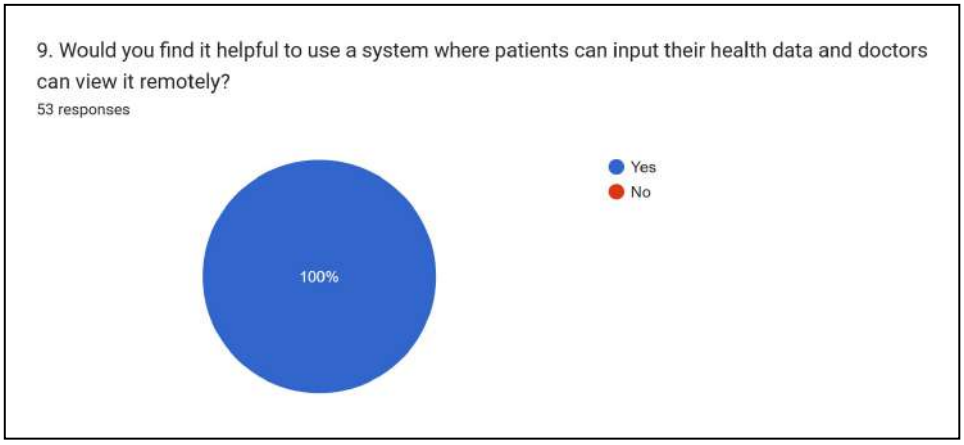


Figure 5.9 : Questionnaire question 9

All repondents responded with a Yes, which reflects total consensus on the importance of such a digital health system implementation. Such an impressive positive feedback testifies to the need to develop and adopt the SmartHealth system. It also shows the willingness of the users to integrate digital technologies that increase communication between patients and physicians, particularly, in the aspects of convenience, effectiveness, and delivery of medical response.

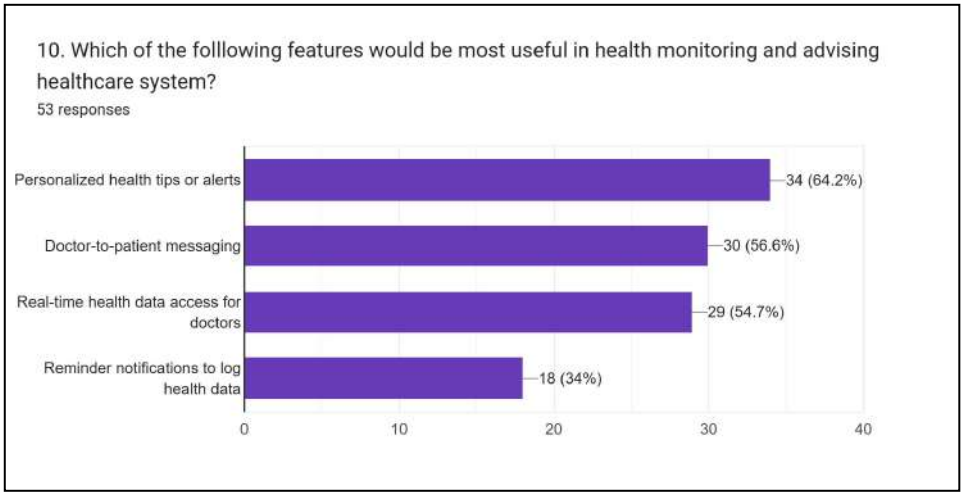


Figure 5.10 : Questionnaire question 10

The figure presents the results of the 53 respondents who rated features in a well-designed health monitoring system. Personalized health tips or alerts were the most popular which is 64.2%, followed by doctor-to-patient messaging which is 56.6% and real-time health data access for doctors which is 54.7%. The type of reminder notification was least which is 34%. Incidents indicate that patients are more concerned with the individualized direction and face-to-face doctor contact than mere reminder system.

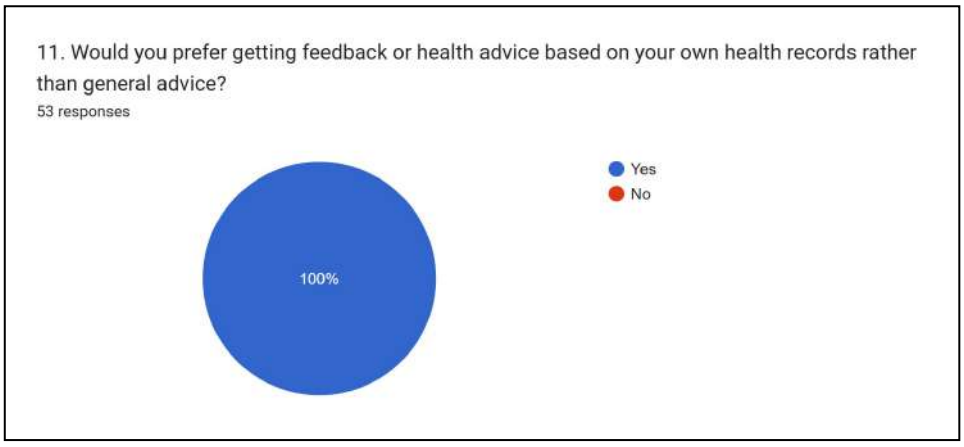


Figure 5.11 : Questionnaire question 11

This pie chart shows that all respondents choose “Yes” for getting feedback or health advice based on their own health records rather than general advice. This features will be add into the system.

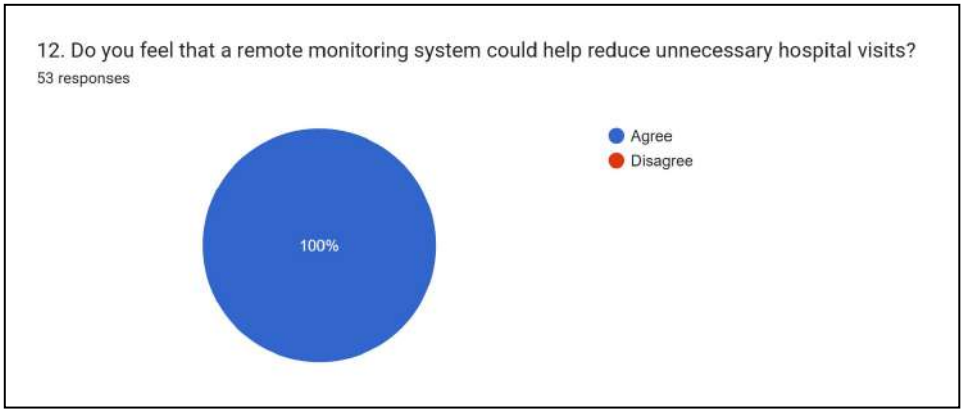


Figure 5.12 : Questionnaire question 12

This pie chart show that all 53 respondents agree that a remote monitoring system could help reduce unnecessary hospital visits. These features will make patients don't have to always visit the hospital to monitor their health data.

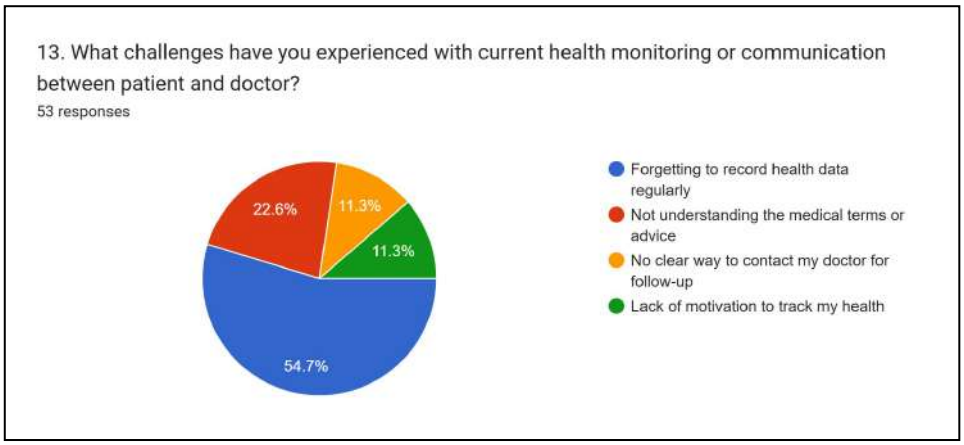


Figure 5.13 : Questionnaire question 13

An analysis of the pie chart exposes challenges that 53 respondents think are important to health monitoring and doctor communication. The most significant problem consisted of the factor: "Forgetting to record regular health information" which is 54.7%, and the second problem was "I did not have an understanding of the medical terms or recommendations" which is 22.6%. This is also the same in other problems such as "no clear way to contact doctors with 11.3% and lack of motivation to track health data with 11.3% respectively.

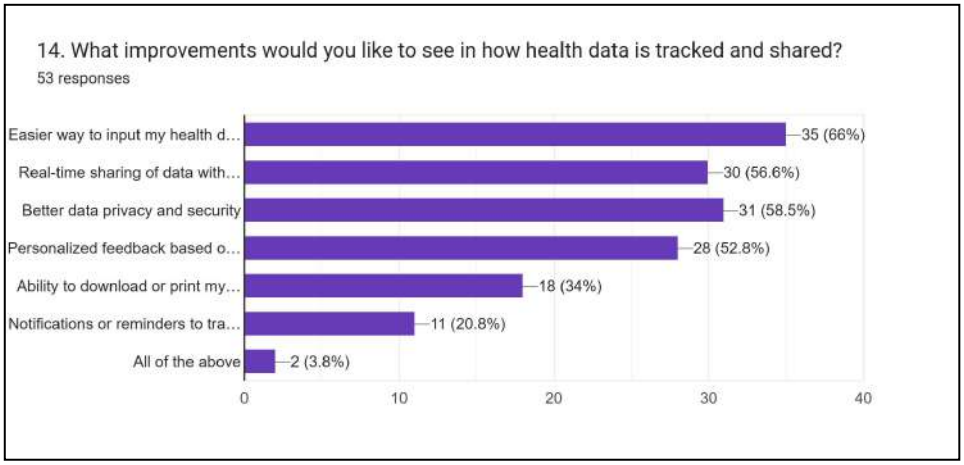


Figure 5.14 : Questionnaire question 14

This pie chart shows that the most respondents request " Easier way to input my health data" into the systems which is 66% . The least request which 20.8% respondents request to have a notifications or reminders to track their health data"

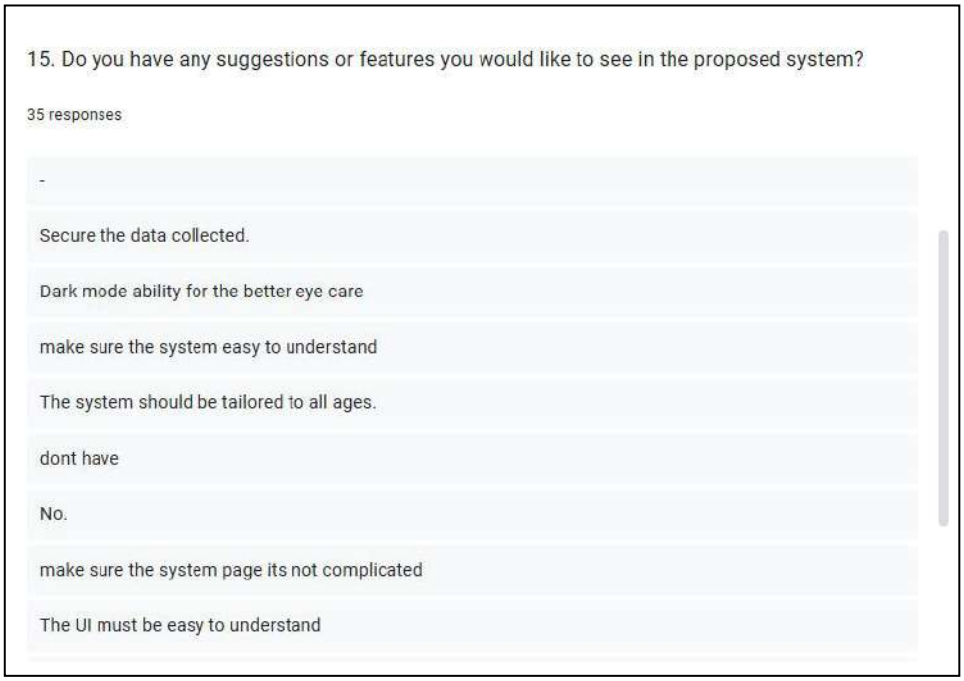


Figure 5.15 : Questionnaire question 15

The surveys of 35 individuals point to several important recommendations that can help to bring the intended health system to a higher level. The greatest demands are strong security of data, convenient interface, and age-inclusive design. Some of them in particular noted that they would want to have the dark mode to eliminate eye strain and that they would want everything user interface-wise to be simple.

5.2.2 Interview Analysis

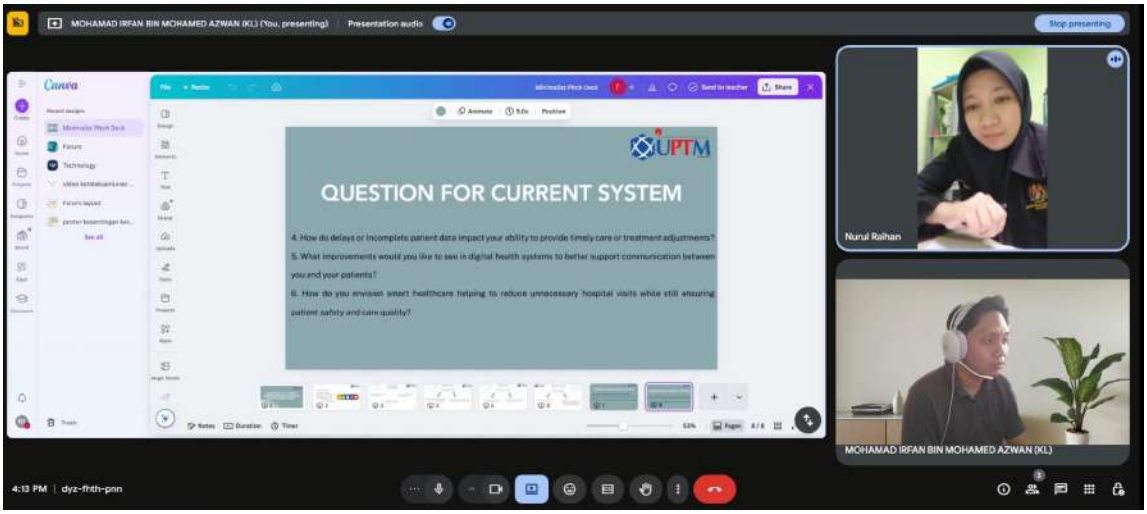


Figure 5.16 : Interview session with client : Dr Nurulraihan

Interview details :

Table 5.1 : Client details

Name:	Dr Nurulraihan binti Mohamed Zawawi
Position:	Head of Transfusion Medicine Department
Location:	Online Google Meet
Date:	26 June 2025

Feedback Analysis :

Dr Nurulraihan binti Mohamed Zawawi which is the client for this project had being interviewed at 26 June 2025 in order to get information and suggestion from her. There is five question provided to ask client what is suggestion and improvement of functional and non-functional requirement should have in the SmartHealth system.

Table 5.2 : Interview question 1 feedback analysis

Question 1 :	What are the biggest challenges for hospital Taiping monitoring patients' health remotely?
Answer :	Sometimes the doctors in Hospital Taiping cannot handle all patients at once especially when the covid is happened at past. The Hospital Taiping still relies on paper based-records to get patients data health.
Details :	The solution reveals the ineffective nature and weakness of the existing manual, paper-based system in managing a high influx of patients particularly during health emergencies such as the COVID-19 pandemic. It produces poor time management in terms of obtaining significant health information and it causes an extra workload among the health personnel. It highlights the high level of necessity in having

	a digital system with a central location in ensuring the increased access to data and the capabilities of monitoring health remotely.

Table 5.3 : Interview question 2 feedback analysis

Question 2 :	In your opinion, what features or tools would make a remote health monitoring system more effective for decision-making?
Answer :	In my opinion, creating the web-based system is a very good idea because the patients can easily send their health data from far away and the doctors can instantly diagnose the condition patients.
Details :	According to the doctor, an online platform would be useful in providing a two-way healthcare communication easily by the patients and the healthcare providers. It also enables real time patient data updates, in such a way that doctors can observe conditions remotely and take data driven decisions with regard to patient care, without ever having to travel. It should have the possibilities of remote submission of data and real-time notification.

Table 5.4 : Interview question 3 feedback analysis

Question 3 :	How do delays or incomplete patient data impact your ability to provide timely care or treatment adjustments?
Answer :	The patients cannot confiming instantly their conditions because of delays of getting result of conditions especially if they have a cronic conditions that need more attentions.
Details :	Gaps or latent data means slow diagnosis and use of treatments which is more threatening to patients with chronic conditions. The doctor emphasizes the necessity of having timely data so that the constant monitoring of the person could be made and so that the appropriate treatment could also be provided. This acknowledges the necessity of the existence of a regular system that will guarantee continuous update of data and warning signal in real time to avoid medical conditions that arise because of lack of data.

Table 5.5 : Interview question 4 feedback analysis

Question 4 :	What improvements would you like to see in digital health systems to better support communication between you and your patients?
Answer :	The system must have a dashboard of all patients and have sort of features so the doctors can easily track specific patient.
Details :	The client underlines the importance of a central location dashboard interface providing doctors with the opportunity to manage and monitor a number of the patients. Such characteristics as sorting patients,

	search filters, and category views are important to enhance usability and help to gain access to critical patient data quickly. This is also helpful in prioritization of patients requiring urgent care.

Table 5.6 : Interview question 5 feedback analysis

Question 5 :	How do you envision smart healthcare helping to reduce unnecessary hospital visits while still ensuring patient safety and care quality?
Answer :	This system can help reduce people being hospitalized in hospital and can The AI advice features can help patients to give guidance and advce for them.
Details :	The physician believes it is worthwhile incorporating recommendation components in the system that will be AI-based and offer automated health suggestions to the patients. It may aid to realize the symptoms early and assist in self-care instructions which lowers the cases of unnecessary visits to hospitals. It also sustains the quality of care by ensuring that the patients are informed and conscious in the choice of self care.

5.3 Use Case Model

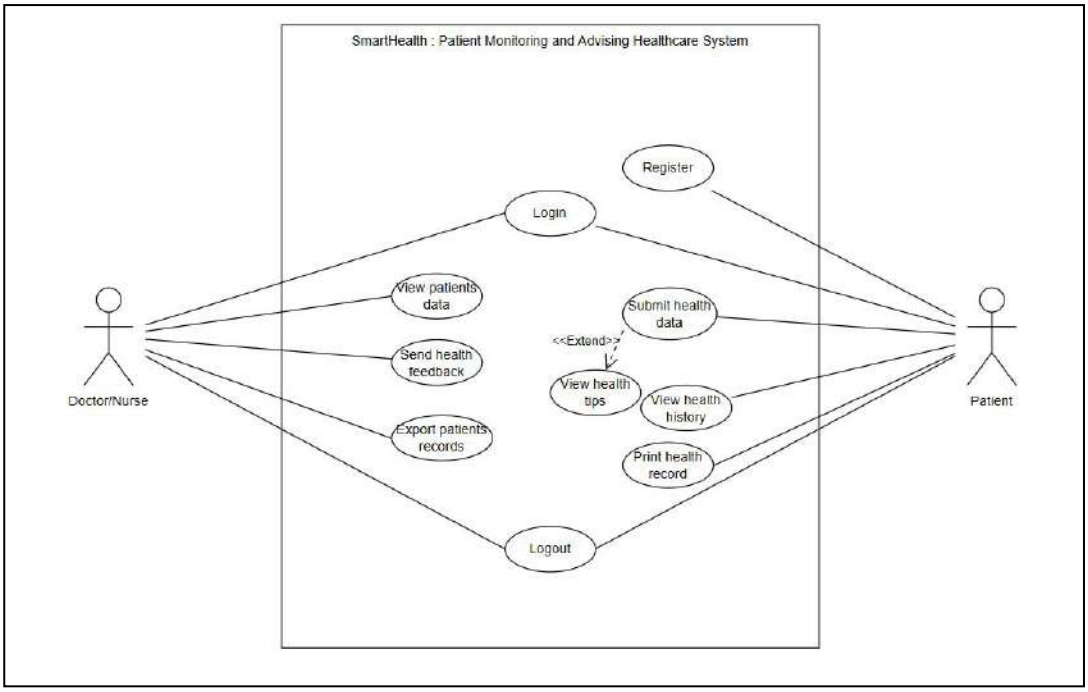


Figure 5.17 : Use case Diagrams

Use case diagram of SmartHealth system records two key actors, Doctor/Nurse and Patient, and the basic interactions performed by those actors. Both actors have to log first in order to use the functionalities of the system, and patients can also create their account in case they do not already have one. This secure authentication will only allow people with authority to access sensitive health information. When patients

are logged in, they will be able to enter and store health data, including their vital signs, symptoms, and further review their own historical health records or print the documents of health records and use automatically manually outside the system or give them to the other healthcare professionals.

On the other hand, doctors or nurse are in constant communication with the system since they can view patient records in real-time, which gives them the ability to track overall patient trends, and identify anomalies within the records in a timely manner. Then they will be able to leave health feedback with the same platform directly and give personal recommendations or tasks on the basis of the data provided. Also, the capability of exporting patient records to provide additional interpretation or inclusion in official medical reports is available to the doctors. To end a secure session, both actors may log out by using the log out feature that stops a secure session and assists in maintaining the data privacy.

5.4 Flowchart

A) Patient

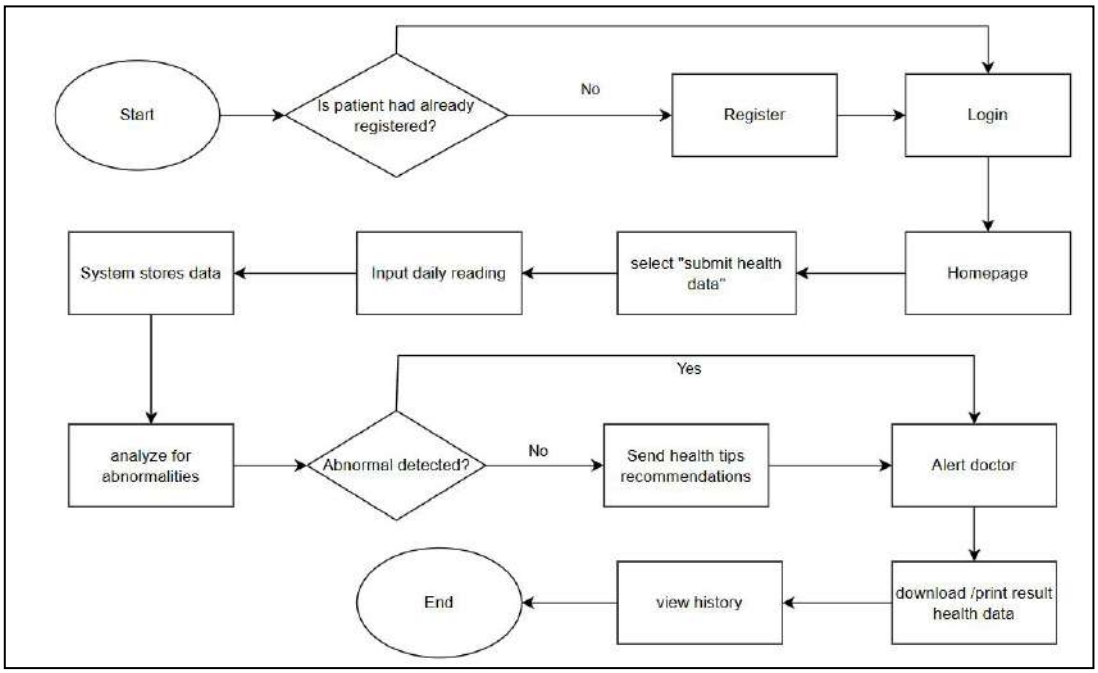


Figure 5.18 : FLowchart diagram (Patients)

The flowchart explains the patient journey of the SmartHealth system. It will start with starting points through which the system checks whether the patient has already been registered. Otherwise they are sent to either login or to register. The patients will be allowed to record their daily health readings and post the documents on the homepage after authenticating their status. This data is then stored by the system in future. The registered ones do not log in and go to type data. By removing unneeded steps, this streamlined process allows every patient to enter their health metrics conveniently, whether it is a first visit or a repeat patient.

Upon data input, the system computes possible abnormalities in the readings. In case of no problems found, a patient will get some health tips and advises related to the patient individually. Nevertheless, in case some peculiarities have been recognized, the system notifies the doctor of it. Their patients are also able to view their health history and also print out or download their results so that they can use them offline. The flowchart ends here as it leaves a clear and step-wise procedure that emphasizes on ease of process, preventive style of the health monitoring equation and prompt action in case of medical necessity. This layout will keep the patients in touch with their health information but at the same time provide a smooth connection to the healthcare providers.

B) Doctors/Nurse

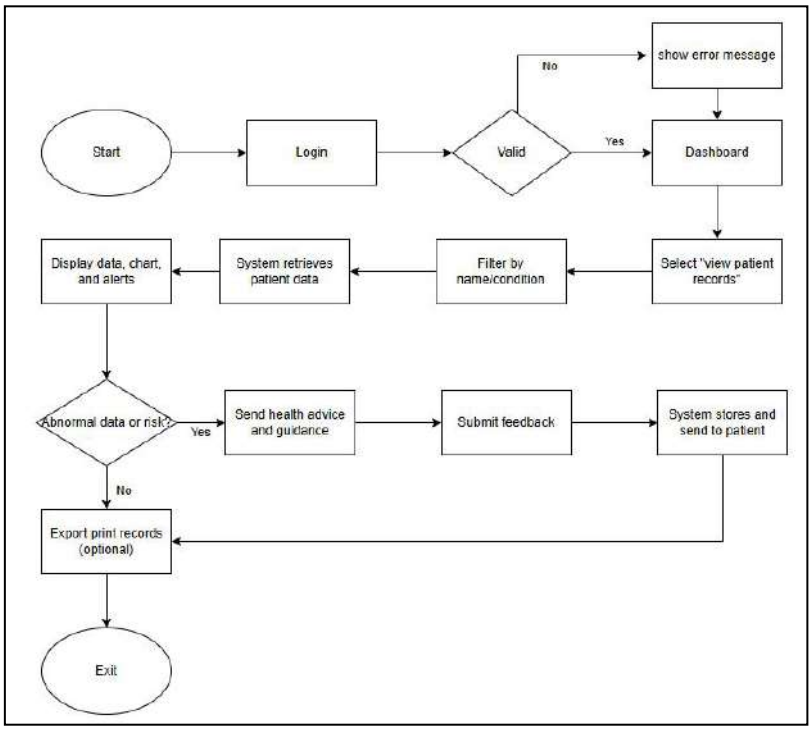


Figure 5.19 : Flowchart Diagram (Doctor/Nurse)

This flowchart demonstrates how doctors or nurses work when they use the SmartHealth system. It starts by the medical worker logging on to the system. When they log in succesful they may view the data about patients which the system automatically checks and stores. When necessary they are able to filter the record on patient name or medical condition. The professional can then decide to send new information, do analysis or even send written instructions and prescriptions directly to the patient. The system makes all proceedings securely stored and conveyed to the concerned patient that there is a flow of information.

The flowchart also provides possibilities of other functions. The healthcare providers will be able to send feedback on the system or get help resources where necessary. There is a choice of exporting or printing offline patient records before closing the session. The process is finished when all the required

actions are realized. Such a lean process focuses on efficiency, data protection, and effective communication between the healthcare providers and the patient so all the healthcare providers can access the patient information comfortably as well as provide the reassuring care and guidance to the patient in a timely manner.

5.5 Conclusion

In conclusion, the current chapter has provided an extensive overview of SmartHealth: Patient Monitoring and Advising Healthcare System. The data-collection procedures involving the use of questionnaires and interviews resulted in the acquisition of important information about the needs and expectations of users as well as patients and doctors of Hospital Taiping. The analysis validated the need of centralized, web-based tool, the one which allows sharing real-time health-related data, provide individual feedbacks, and alleviate the hospital overloading.

The represented models, including use case diagram, and flowcharts that deal with the system, demonstrated the general picture of the system internal structure and the interaction of various users. These graphic representations were useful in describing how the system is to work, how consumers will be used and also in aligning with the real needs of the user. In general, such an analysis stage was important in the development of a user-centered and purposely based system which can cater to the real needs of the healthcare professional as well as the patient.

6 REQUIREMENTS

6.1 Introduction

This chapter identifies the essential specifications that are used as the blueprint of the SmartHealth: Patient Monitoring and Advising Healthcare System. The specification of these requirements is an essential stage in the system development life cycle, since it will define the precise and complete vision of the system desired functions, limitations, and purposes.

This chapter will describe the key elements that will lead to the implementation of the project, the design of the user interface, the database structure, and the flow of the system functioning. This chapter gives a good base to further design and testing of the SmartHealth system by defining these aspects so that the end product would not be out of scope or would not fail to comply with the objectives of the project and address the needs of the end-users of the final product.

6.2 Interface Design

6.2.1 Login Page

Figure 6.1 : Wireframe - Login Page

Figure 6.1 shows the login page for both patients to login their own account by inserting email and password. There is also the URL to go to the register page for a new user. Both textbox need required to fill up to enter the SmartHealth system.

6.2.2 Register Page

Figure 6.2 : Wireframe - Register Page

Figure 6.2 display the register page for patients if they new to SmartHealth System. The patients required to enter username, full name, email, birth date, phone number, password, and confirm password for this registration. There is also URL “Already registered?” to go back to login page if the user already have the account.

6.2.3 Patient - Dashboard

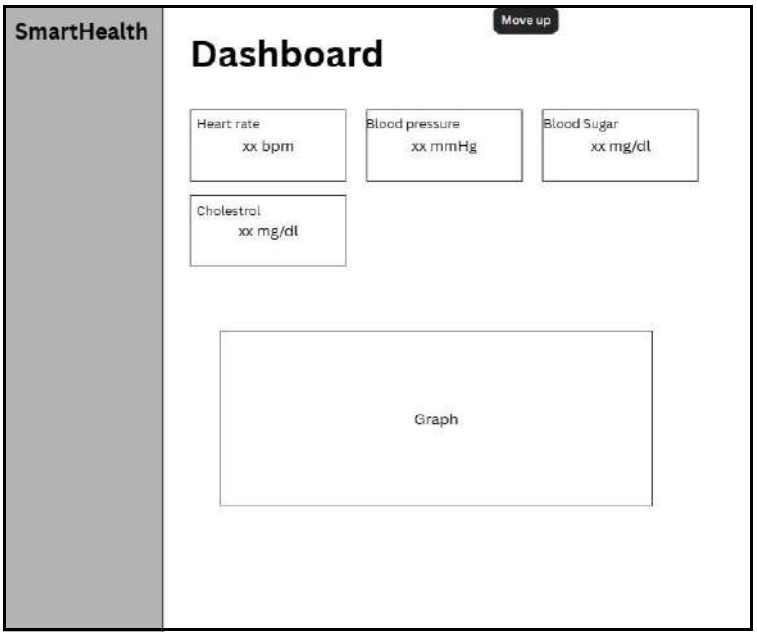


Figure 6.3 : Wireframe : Patient's Dashboard

Figure 6.3 displays the dashboard patients after the patient login, In this page shows the latest input medical checkup and also a graph for every past medical checkup. The patient can view directly their health condition.

6.2.4 Patient - Profile Page

Figure 6.4 : Wireframe : Patient's Profile Page

Figure 6.4 display the profile information patients based on their registration data profile. This information page cannot edit because its view only, The patient need to click edit profile button to go to the page that can edit their own personal information.

6.2.5 Patient - Edit Profile Page

Figure 6.5 : Wireframe - Patient's Edit Profile Page

Figure 6.5 displays their own profile information but they can edit their own personal information. The patients can click “save changes” after they edit their profile or they can click cancel if they don’t want change their personal information.

6.2.6 Patient - Medical Checkup Page

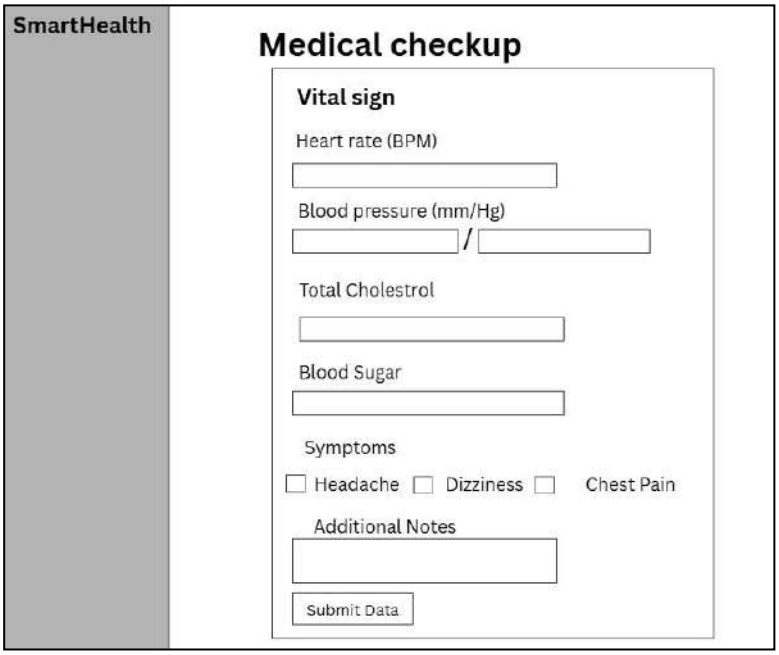


Figure 6.6 : Wireframe - Patient’s Medical Checkup Page

Figure 6.6 shows the medical checkup page for patients. In this page, the patient can insert their health records in this page. After that, this page will save their health data and it will send to doctor monitoring page after submission from patient.

6.2.7 Patient - Medical Records Page

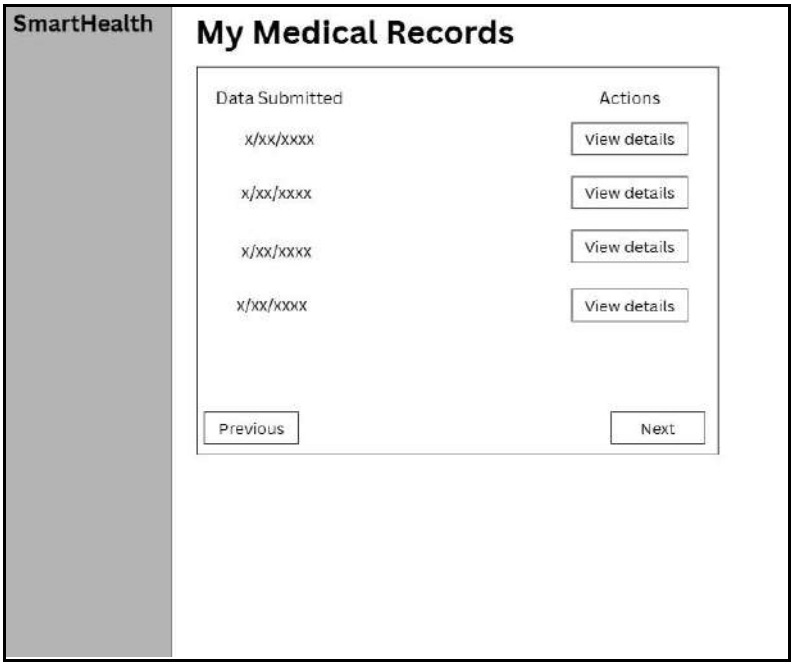


Figure 6.7 : Wireframe - Patient's Medical Records Page

Figure 6.7 shows the past of medical records. The patient can click "view details" to see the Ai analyzer based on the input from medical checkup.

6.2.8 Patient - Appointments Page

Figure 6.8 : Wireframe - Patient's Appointments Page

Figure 6.8 shows the appointment page for patients. In this page, the patients can set the date and give the reason for appointments to book the appointments.

6.2.9 Patient - Message Page

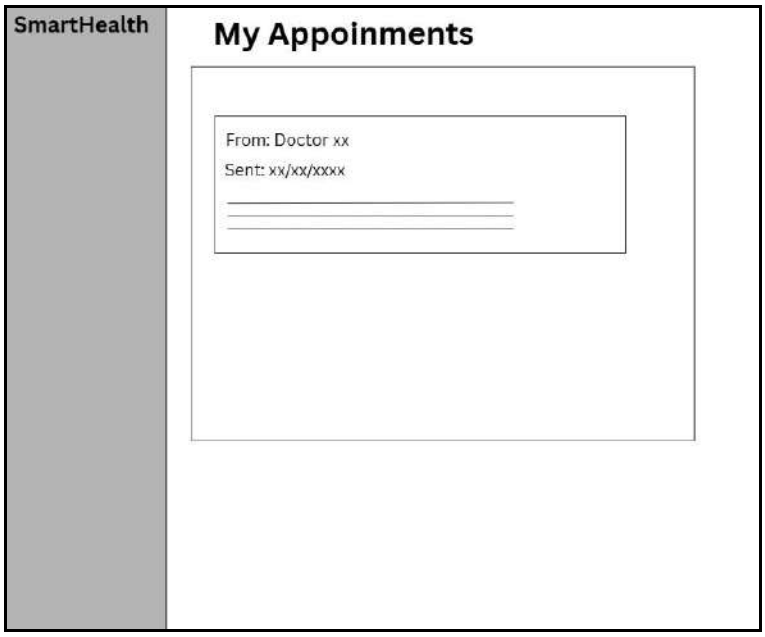


Figure 6.9 : Wireframe - Patient's Message Page

Figure 6.9 display the message page from doctor. The patients can view the message from the doctor but it cannot send back the message.

6.2.10 Doctor-Dashboard Page

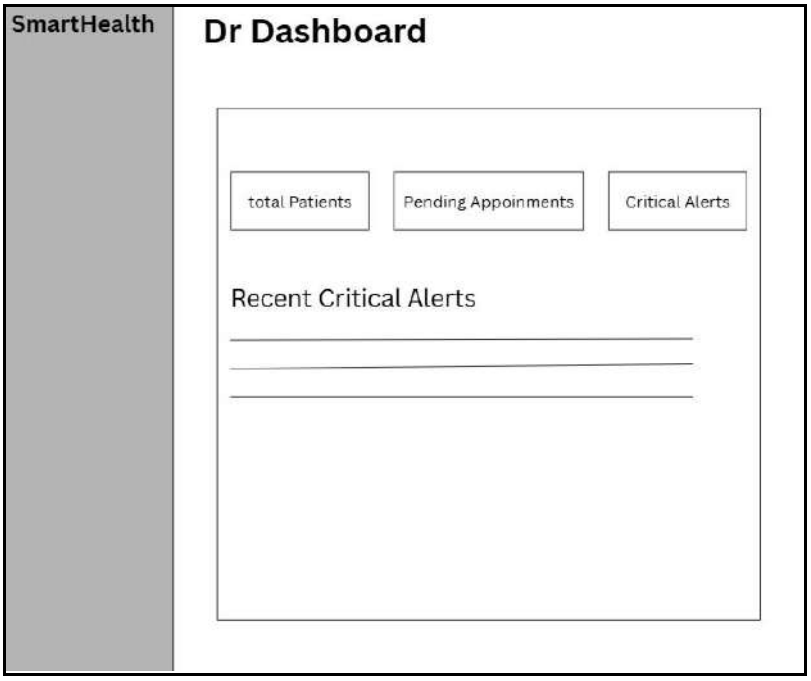


Figure 6.10 : Wireframe : Doctor's Dashboard Page

Figure 6.10 shows the dashboard page for doctor after they log in. In this page, The doctor can see the twenty four hours critical patients so the doctor can instantly see the critical patients.

6.2.11 Doctor - Monitoring Patients Page

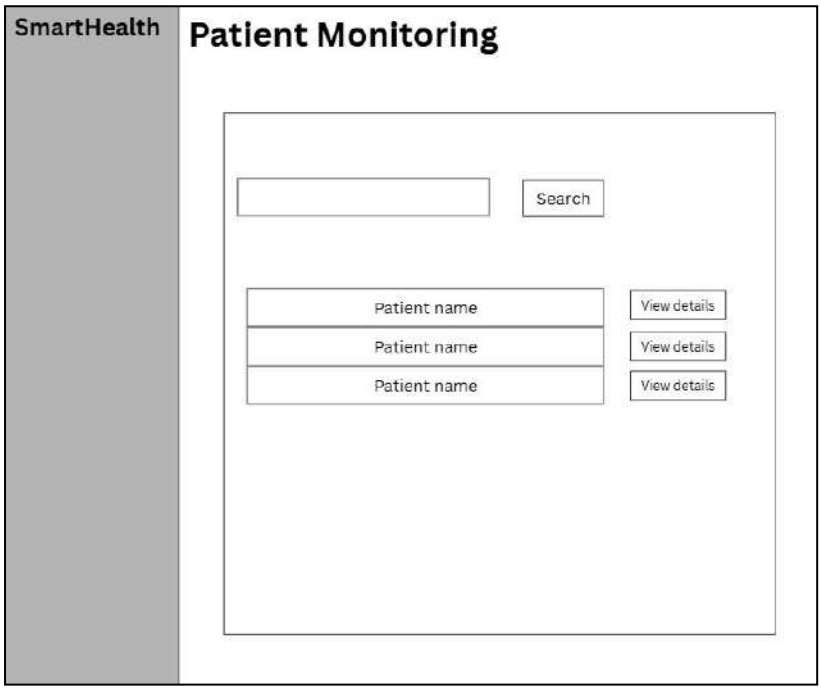


Figure 6.11 : Wireframe - Doctor's Monitoring Patients Page

Figure 6.11 shows the monitoring page for doctor to check the list of patient name. In this page, the doctor can click view details of specific patients to check and diagnose the conditions of patients. The doctor can also search by name of patient to find specific patients.

6.2.12 Doctor - Patients Overview Page

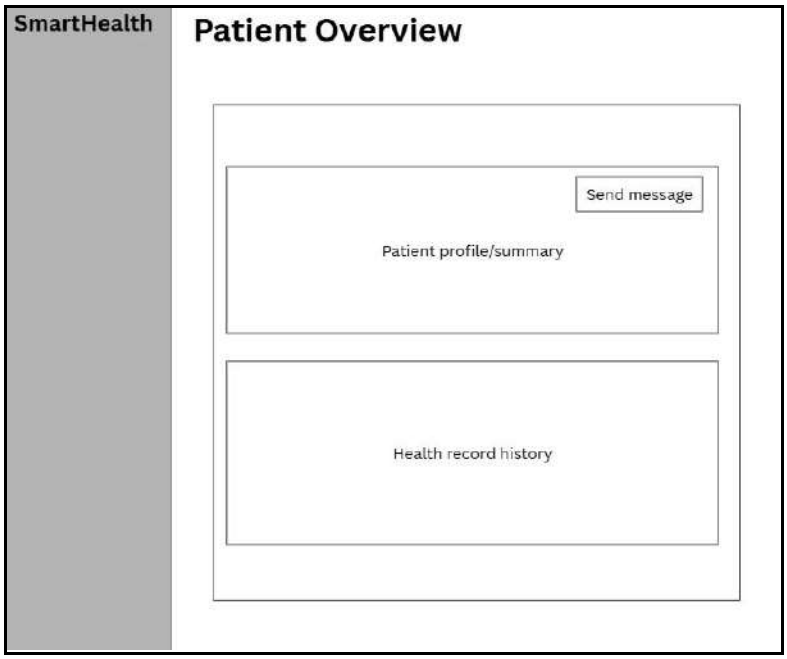


Figure 6.12 : Wireframe - Doctor's Patients Overview Page

Figure 6.12 shows the patients overview page after the doctor click view details in monitoring page. In this page, the doctor can check patient profile and also health record history of patients. There is also send message button to message the patient.

6.2.13 Doctor - Manage Appointments Page

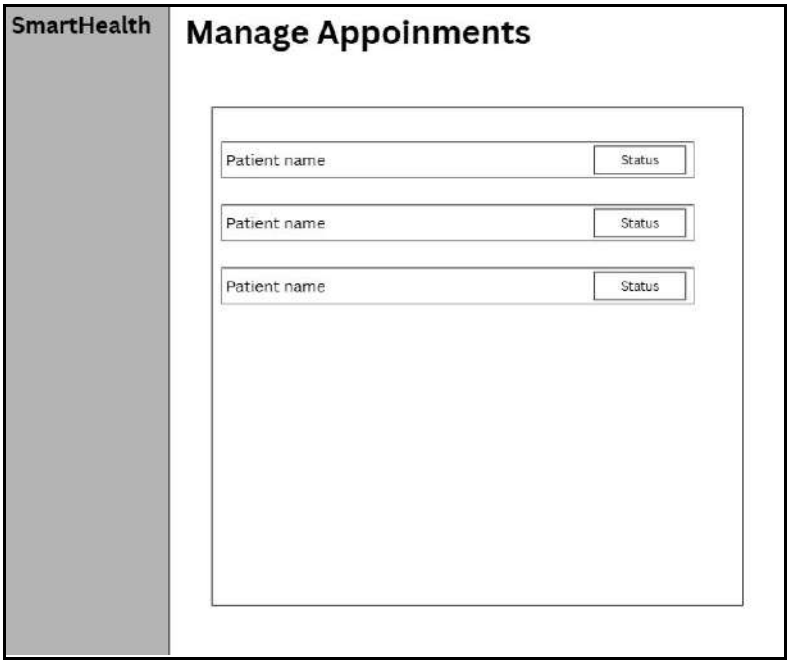


Figure 6.13 : Wireframe - Doctor's Manage Appointments Page

Figure 6.13 shows the doctor's appointments page. The doctor can manage the status of patient booking appointment by approving or decline the patient's appointments.

6.2.14 Doctor - Health Guidlline

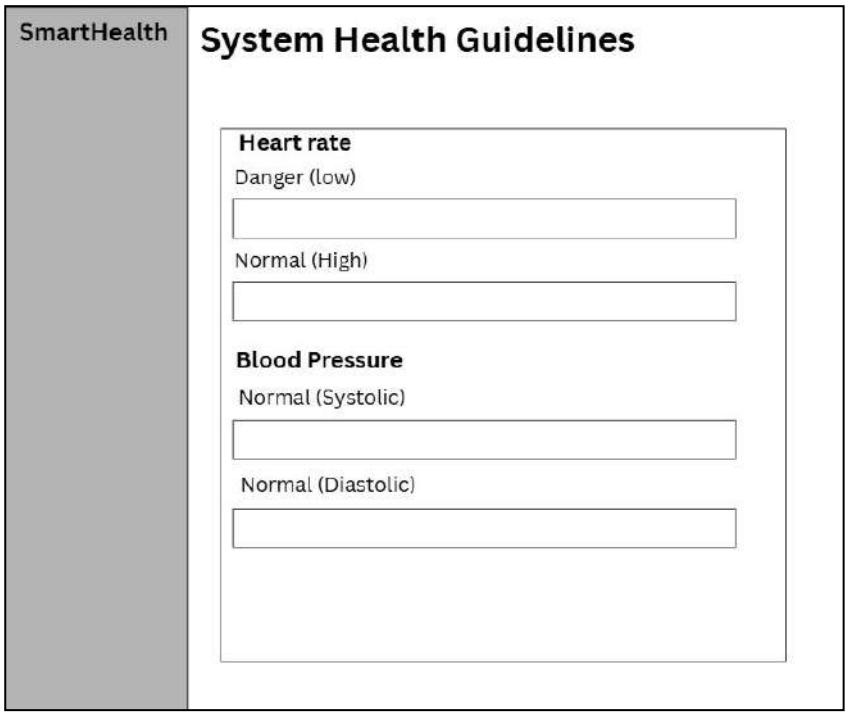


Figure 6.14 : Wireframe - Doctor's Health Guidelines

Figure 6.14 shows the page for doctor can edit health guidelines for medical checkup guidelines. The doctor can update the guidelines from time to time to ensure the range of health metric is always follow the global current range health metric.

6.3 Database Design

6.3.1 Data Dictionary

The SmartHealth: Patient Monitoring and Advising Healthcare System relies on having a strong and well-organized database. It securely stores, organizes and retrieves critical patient data, such as vital signs and communication with doctors. The database architecture sustains data integrity, scalability, and database performance.

This portion explains the structural design of the database which begins with a summarized overview of the entities and their relation with the rest of the system and concludes with a detailed data dictionary. A relational model, realized in MySQL, associates users with their health records, appointments, messages, and notifications.

6.3.2 users Table

Table 6.1 : users Table

Table Name	Data Item	Data Type	Description	Example
users	Id (PK)	bigint	Unique identifier for the user.	1
	full_name	varchar	The user's full name.	MOHAMAD IRFAN
	username	varchar	The user's unique login username.	Irfan_azwan
	email	varchar	The user's unique email address.	irfan@gmail.com
	email_verified_at	timestamp	Timestamp of email verification.	null

	password	varchar	The user's hashed password.	Hashed String
	birth_date	date	The user's date of birth.	17/1/2002
	phone_number	varchar	The user's contact phone number.	012- 1102911
	role	enum	Defines user type.	patient
	remember_token	varchar	Token for remember me sessions.	null
	create_at	timestamp	Timestamp of account creation.	2025-11-17 20:45:01
	updated_at	timestamp	Timestamp of the last update.	2025-11-17 20:45:01

The main entity in the application is the users table that is listed in Table 6.1 which contains patient and doctor information. The primary key is the id, which is an integer, which identifies every registered user. The table contains personal information fields, the fullname, username, birthdate, and phone_number. The email box accommodates the individual contact and login information of the user. Attributes that relate to security such as the password field have a hashed value to help authenticate user access. One of the crucial fields and roles is the permission of the user in the system, who can be a patient or a doctor.

6.3.3 health_records

Table 6.2 : health_records Table

Table Name	Data Item	Data Type	Description	Example
health_records	id	bigint	Unique identifier for the health record.	101
	patient_id	bigint	Links to the id in the users table.	1
	heart_rate	int	Patient's heart rate in beats per minute (bpm).	72
	systolic_pressure	int	Blood pressure's top number	120
	diastolic_pressure	int	Blood pressure's bottom number	80
	cholestrol	int	Patient's total cholesterol in mg/dL.	190
	blood_sugar_value	decimal	The numeric value for blood sugar.	95.00
	blood_sugar_unit	varchar	The unit of measurement.	Mg/dl
	symptoms	json	A JSON array storing the list of selected symptoms.	“headache”,”fatigue”
	notes	text	Optional text notes submitted by the patient.	Feeling stressed
	recommendation	text	Stores the JSON response from the AI analysis.	“status : Good”,…

	created_at	timestamp	Timestamp of when the record was submitted.	2025-11-17 20:50:10
	updated_at	timestamp	Timestamp of the last update.	2025-11-17 20:50:10

The healthrecords table of Table 6.2 is among the most important tables that show the aim of storing all the individual health checkups made by a patient. Its own primary key, id, is a unique identifier of each submission. This table will be connected to users table directly by the foreign key patientid. It captures the entire vital signs, such as heartbeat rates, systolicpressure, diastolicpressure, and cholesterol. Blood sugar is stored in the form of two fields: bloodsugarvalue Units number and bloodsugar unit Tests type (e.g., mg/dl). Symptoms field is a collection of patient-selected symptoms in the form of a JSON array, whereas the recommend field contains the JSON result of the AI analysis.

6.3.4 Appointments Table

Table 6.3 : appointments Table

Table Name	Data Item	Data Type	Description	Example
appointments	id	bigint	Unique identifier for the appointment.	5
	patient_id	bigint	Links to the id in the users table (the patient).	1
	doctor_id	bigint	Links to the id in the users table (the doctor).	2
	appointment_date	datetime	The date and time of the requested appointment.	2025-11-20 14:30:00
	reason	text	The reason for the appointment.	Follow-up on high blood pressure.
	status	varchar	The current status of the request.	Pending
	created_at	timestamp	Timestamp of when the appointment was booked.	2025-11-17 21:00:15
	updated_at	timestamp	Timestamp of the last update.	2025-11-17 21:00:15

Table 6.3 in the form of the appointments table is used to control the scheduling and existence of all appointment requests. Each unique booking is identified by the primary key, id. The table is an intermediary between the patients and the doctors where two foreign keys will be used: patientid to identify the patient who requested it and doctorid to identify the doctor who will process it. The most important data that is kept would be appointment date, the reason given by the patient and the status of the request including pending, approved and cancelled is handled by the doctor.

6.3.5 messages table

Table 6.4 : messages Table

Table Name	Data Item	Data Type	Description	Example
messages	id	bigint	Unique identifier for the message.	50
	sender_id	bigint	Links to the id in the users table (the sender).	2
	receiver_id	bigint	Links to the id in the users table (the receiver).	1
	message	text	The text content of the message.	Please reduce your salt intake.
	file_path	varchar	Path to any attached file and image.	bp_guide.pdf
	read_at	timestamp	Timestamp of when the message was read.	null
	created_at	timestamp	Timestamp of when the message was sent.	2025-11-17 21:05:30
	updated_at	timestamp	Timestamp of the last update.	2025-11-17 21:05:30

Table 6.4 messages table supports the two-way communication between patients and doctors. The messages are represented in a row with each row having an id. The conversation is controlled by two foreign keys of senderid and receiverid that are the table structure and which both refer to users table. The textual content is stored in the message field and the reference to any attached pictures or PDF documents is stored in the filepath field. The readat field is critical to the notification system because it enables the application to monitor the message as new or read.

6.3.6 health_guidelines table

Table 6.5 : health_guidelines Table

Table Name	Data Item	Data Type	Description	Example
health_guidelines	id	bigint	Unique identifier for the guideline.	1
	metric	varchar	A unique key for the rule.	hr_danger_low
	name	varchar	A human-readable name.	Heart Rate (Danger Low)
	value	int	The numeric threshold value.	60
	created_at	timestamp	Timestamp of creation.	2025-11-17 20:30:00
	updated_at	timestamp	Timestamp of the last update.	2025-11-17 20:30:00

Table 6.5 is a special attribute of the SmartHealth system as healthguidelines table is the storage of the dynamic health reference values. The codes of rules are not hard-coded but rather stored here, so that a doctor can edit the rules on the system guidelines page. The rule has a unique key that called metric field, which is the name of the rule and the value field, where the numeric threshold of the rule is stored. This table provides a way to make sure that one single centralized source of truth is used by the patient guide to reference, the doctor alert system, and the AI analysis.

6.4 Data Flow Diagram

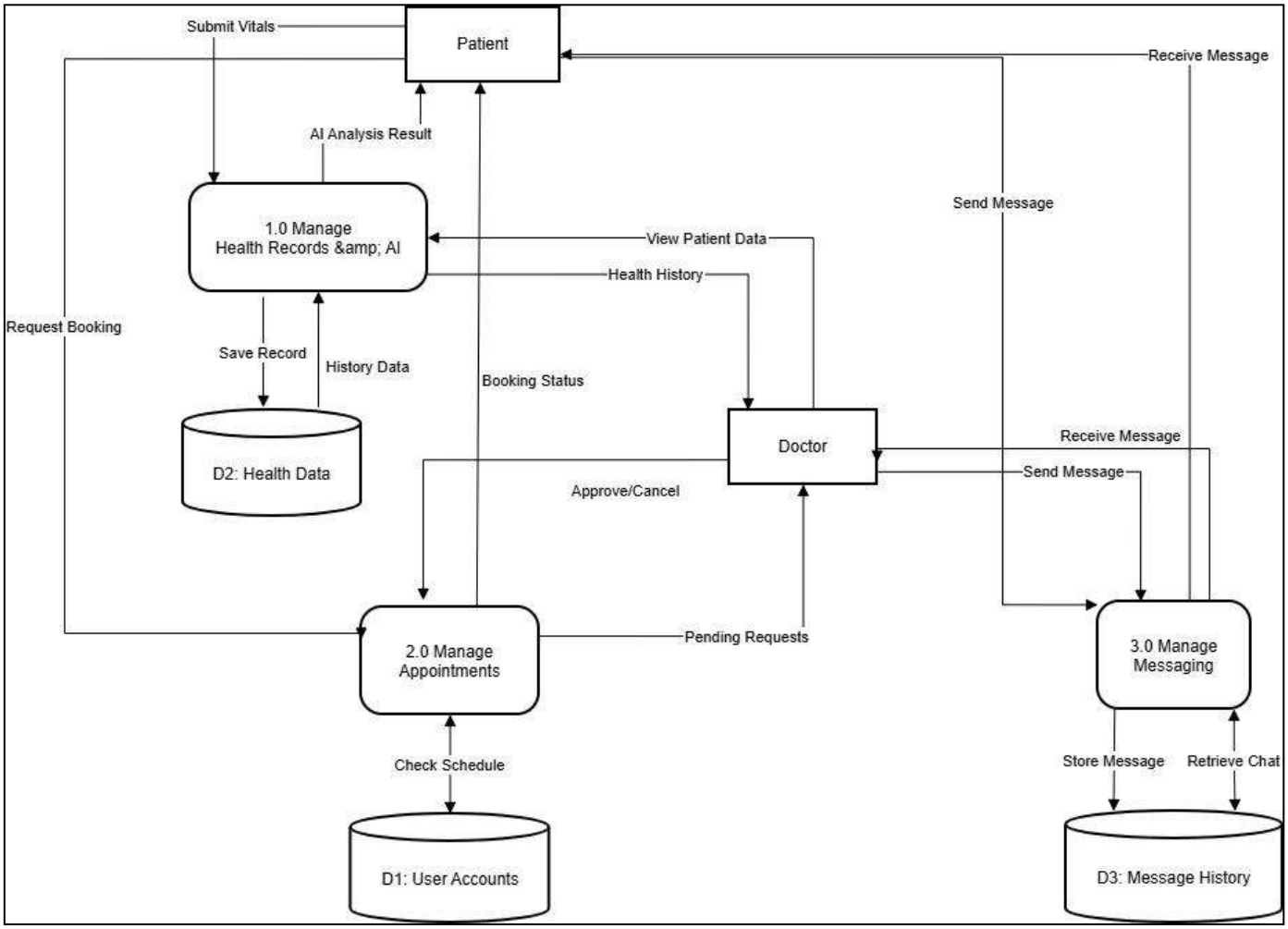


Figure 6.15 : SmartHealth Data Flow Diagram

Figure 6.15 shows the Data Flow Diagram (DFD) of the SmartHealth: Patient Monitoring and Advising Healthcare System. This diagram gives the holistic perspective of the architecture of the system which dynamicly shares information between the main users who are patients and doctors and the main functional processes of the application. The SmartHealth system can serve as a central node, which enables the amassing, analysis, and control of vital health information.

The most important source of health data in the system is the Patient entity. The system is used by patients to carry out some of the vital functions that revolve around personal health management. The process starts with Submit Medical Checkup in which they enter vital signs that include heart rate, blood pressure, and blood sugar. The data is then inputted to the system where it undergoes an AI model to produce an AI Analysis, which provides an immediate feedback and recommendations to the patient. Moreover, patients use the Book Appointment to make appointments, and this creates changes

in the appointment records in the system. They also participate in the Messaging process to directly address their healthcare providers and send queries and get professional guidance.

The Doctor organization is the administrative and medical director of the system. The platform is used by doctors to track the health of the patients, schedule, and uphold the standards of the system. The Patient Monitoring process allows the doctors to have a complete list of patients and has the capability of searching and filtering based on critical cases that have to be dealt with as soon as possible. Manage Appointments process enables doctors to view outstanding appointments of patients and change their status to either approved or cancelled. Doctors are also engaged in the Messaging process and will respond to the queries posed by the patients and give medical advice. The update of the health guidelines, in which the doctor can adjust the values of the reference of the system on health metrics, is a special responsibility of the doctor, which would not spoil the AI analysis, which would be medically accurate and updated. All these interactions lead to a change in the underlying data stores, which keeps a record of the operations of the clinic in real time.

The SmartHealth system is, in general, a unified place that facilitates the sharing of vital health data. The system allows proactive, data-driven, and collaborative health monitoring because it focuses on real-time data acquisition of patients and offers doctors with the means of analysis and communication. The constant stream of data (submission to analysis to professional review) makes it possible to intervene in the medical process in time and makes patients more engaged in the medical process of their lives.

6.5 Entity Relation Diagram

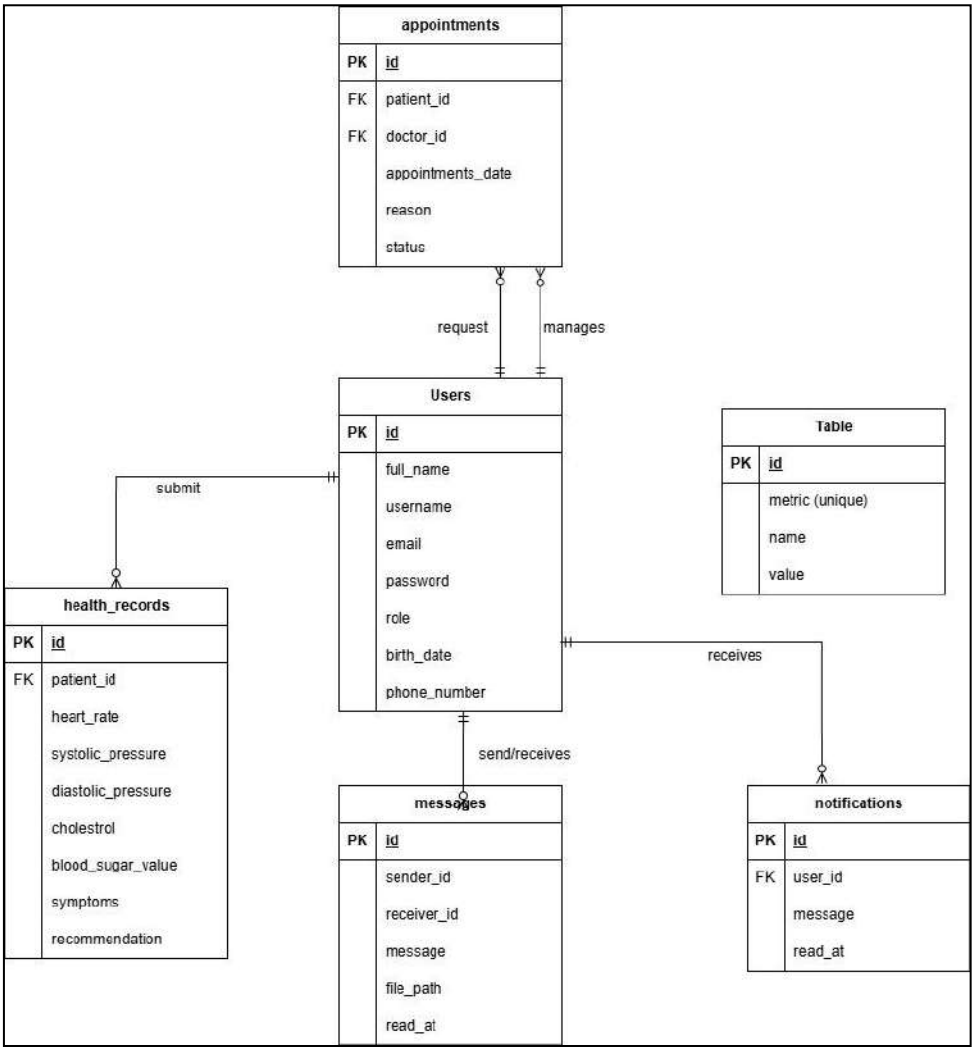


Figure 6.16 : SmartHealth Entity Relation Diagram

Figure 6.16 is the Entity-Relationship Diagram (ERD) of the SmartHealth system, which is the blueprint of the database structure applied to handle patient health data, user roles and communication. The relationship model central to this will be the users node or entity that will serve as a hub of all account holders. This table holds the most important personal information including the full name of the user, contacts, date of birth, and secure user login information. One of the most important qualities in this aspect is the role, as it separates a patient and a doctor, defining their access rights and capabilities in the system.

The health-records object is connected with the user table in order to enable monitoring of health status. This organization records all the individual checkups posted by a patient and it includes the vital changes such as heart rate, blood pressure, blood sugar level, and cholesterol level. It has also the capability of storing qualitative data like reported symptoms, and AI-generated recommendations. This form of organization will make a complete chronological history of the health of a patient available to

analyze. On the same note, the appointments entity handles the scheduling exercise. It relates a patient and a doctor with the help of foreign key relations and tracks the date of the appointment, the purpose of the visit of the patient, and the status of the request vacuola pending or approved and provides the efficient work between the parties.

Messages entity deals with communication and it allows two-way interaction to be done securely. This table contains the sender and receiver identities, the content of the message and any attachments made to the message, as well as, monitoring the read status to send out notifications. There is also a notifications entity to take care of real-time messages to the users to notify them about new messages or appointment status. Lastly, the configurable reference values of health metrics are stored under the health guidelines entity. This enables the doctors to dynamically change the thresholds of the normal or the danger zone without affecting the code of the system as the medical advice will always be up-to-date and flexible.

Foreign keys will be used to refer to each other and maintain referential integrity to make sure that all health records, appointments, and messages are properly related to a legitimate user. This ERD offers a flexible and scalable base of the SmartHealth system that can be easily used to manage multifaceted medical records and interactions with users.

6.6 Flow of The System

6.6.1 Patient

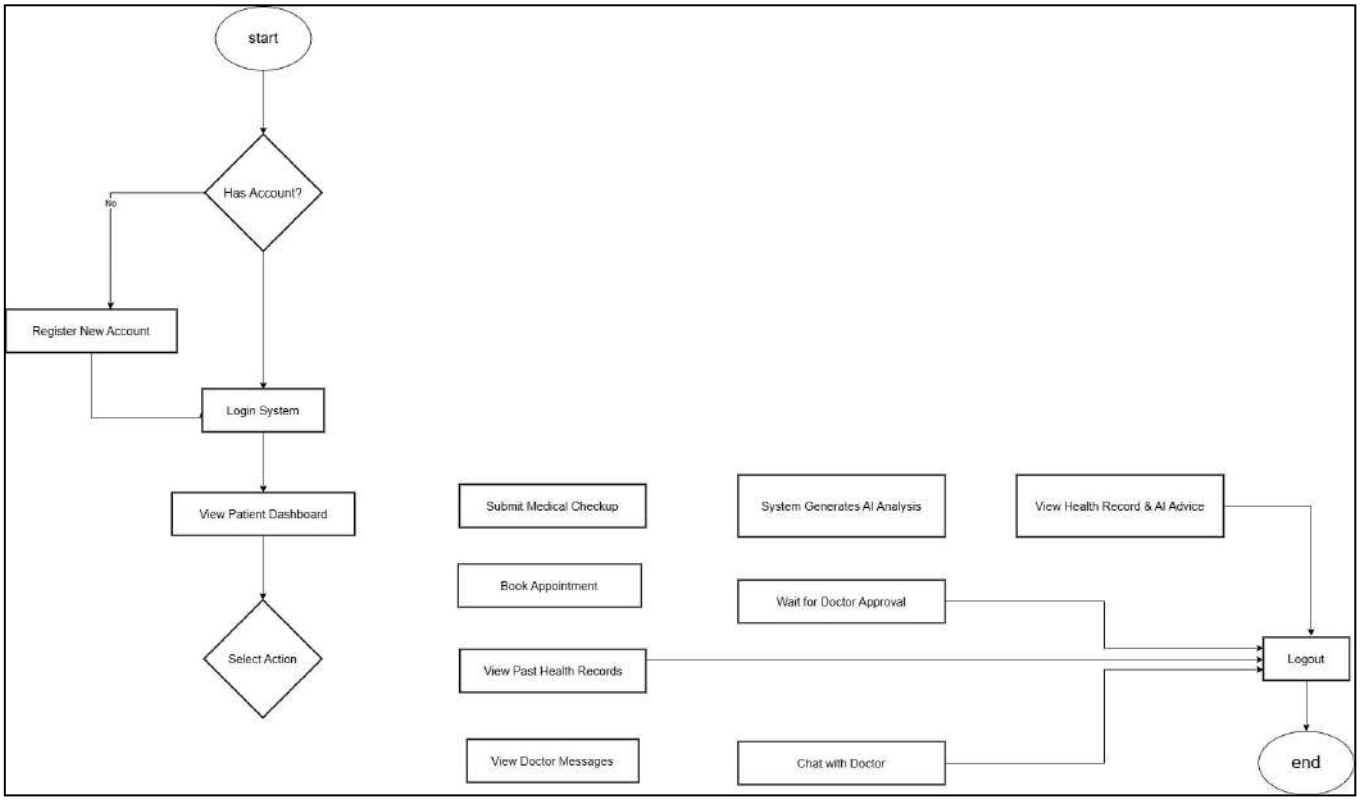


Figure 6.17: Flow of The System (Patient)

Figure 6.17 is the flow of the system of the SmartHealth web-based patient monitoring system of patient. Patient Flow is self-monitoring and activity based. Upon their log in, they are shown a dashboard that visualizes their health trends. Their main communication also includes making a new Medical Checkup where they enter the vital signs, including blood pressure, heart rate, and blood sugar. When submitted, this system initiates the process of AI analysis, which provides immediate feedback and health advice. The Book Appointment will allow the patients to make a meeting and the Messaging module will allow them to chat directly with their healthcare provider. All the actions, as well as the access to the previous health records, are concentrated in the secure session of the patient.

6.6.2 Doctor

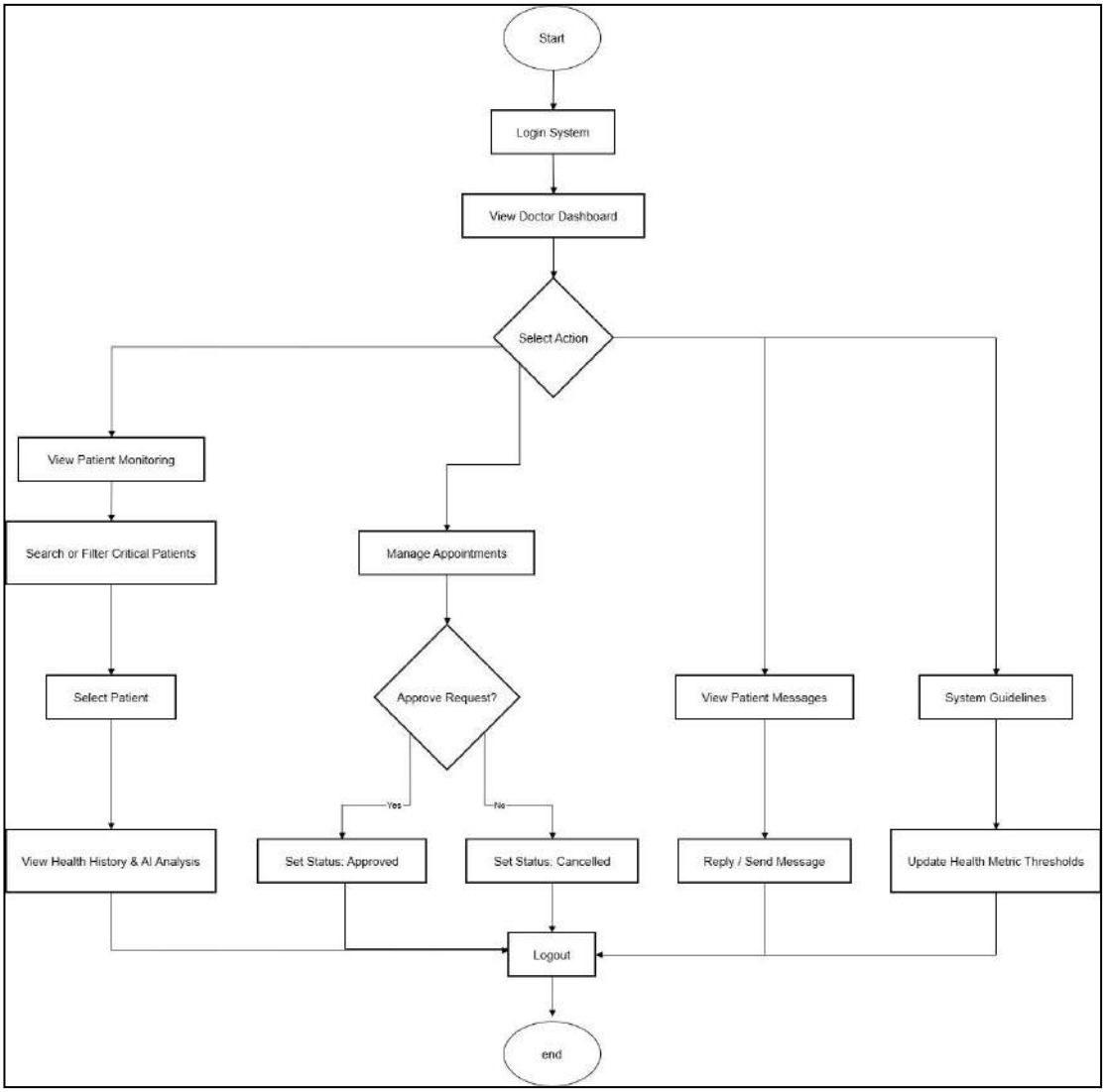


Figure 6.18 : Flow of The System (Doctor)

Figure 6.18 presents the flow of the system of the SmartHealth web-based patient monitoring system of doctor. The Doctor Flow is involved in management and control. The dashboard of the doctor gives a high level of the status of the clinic, with critical notifications. It is starting at this point that the doctors can use the Patient Monitoring module to search by specific patients or filter down to patients who have critical health indicators. This is a means of targeted intervention. Medical practitioners are also administered administrative functions, including the Manage Appointments section to accept or reject requests, and the System Settings section to make changes to the health metric guidelines that inform the system to issue alerts. Communication is also completed through Messaging feature that enables the doctors to respond to the queries of the patients and offer professional medical consultation with ease.

6.7 Conclusion

The requirements and design part provided in this chapter have played a critical role in creating a solid outline of the SmartHealth: Patient Monitoring and Advising Healthcare System. The wireframe design has been carefully developed to align with user-centered experience which entails provision of user-friendly dashboards that are user-friendly to both patients and doctors to allow smooth navigation and execution of tasks. The database design, which is documented in the form of the Entity-Relationship Diagram (ERD) and a very detailed data dictionary, is designed to make sure that all the important data of user profiles to sophisticated health records and communication logs are put in place in the most efficient manner and that high integrity is upheld. Moreover, Data Flow Diagrams (DFD) and System Flowcharts have been utilized to clearly depict the flow of operations in the system and logical movement of information and the specific paths of interaction of each user role. All these design components give a strong blueprint to the development phase to ensure that the end system is secure, scalable, and be able to provide effective remote health monitoring and advisory services.

7 IMPLEMANTATION

7.1 Introduction

This chapter is dedicated to the implementation phase of the SmartHealth Patient Monitoring and Advising Healthcare System, which includes the description of the practical implementation of the project. The implementation phase is the most crucial step that signals the shift of theoretical designs and conceptual framework into a fully operational, strong software programme. It is a stage when all of the system modules, such as patient data submission, analysis using AI, appointment scheduling, and communication between doctors and patients, will be created, combined, and thoroughly tested to satisfy the required characteristics. In this section, the tools and technologies that are applied to the construction of the SmartHealth system will be discussed, the real interface of the system will be presented based on the wireframes as initially envisioned, and an in-depth discussion of the notable security measures that are applied to ensure the protection of the sensitive medical data. This detailed overview illustrates how the project objectives have been changed into a practical and immediate solution that can be put into use.

7.2 Execution Platform

7.2.1 Windows 11

Figure 7.1 : Windows 11

After the development of the SmartHealth system, the Windows 11 was chosen as the primary operating system of the primary hardware device to develop, test, and implement the entire web application. This was done intentionally to take advantage of the upgraded architecture of the system that is efficient and that which is friendly to developers. Windows 11 provides a modernized and simplified operating system that is stable and provides high performance in processing. Major advantages in the implementation of the project were that it took shorter time to start development environments, easier transition of applications, and a highly responsive ability to execute multiple high-performance applications at the same time. Considering the continuous changing between coding in the IDE, testing in various browser windows, database console operations, and the execution of the AI service, the efficiency of Windows 11 played a significant role in reducing the latency and maximizing the productivity of the developer at the implementation stage.

7.3 Implementation Tools

7.3.1 Visual Studio Code

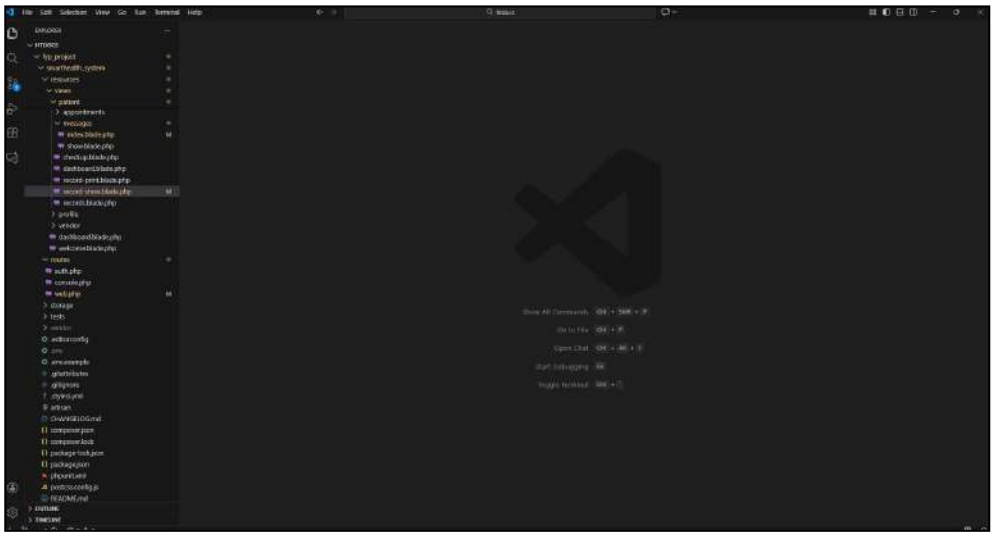


Figure 7.2 : Interface Visual Studio Code

Figure 7.2 illustrates how Visual Studio Code (VS Code) dominated the coding for the SmartHealth system in the development stage, with VS Code being the main IDE for coding. The combination of its lightweight yet powerful nature with rich PHP (Laravel), JavaScript, HTML and CSS language support makes it the ultimate code editing tool. . Some of the features that contributed to faster development are smart syntax highlighting, integrated debugging for faster error tracing, together with terminal access, need for running commands of the framework and for version control (Git). In addition, the rich VS Code extension ecosystem was utilized for linting, formatting code, and database interaction, improving developer productivity and maintaining a consistent code quality throughout the project.

7.3.2 MySQL Database

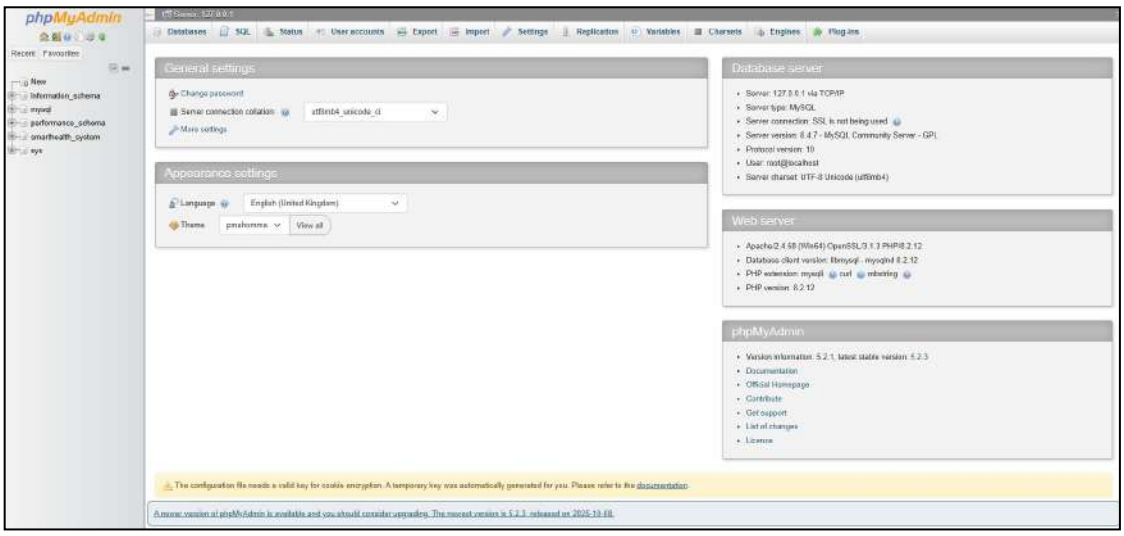


Figure 7.3 : MySQL Database

Based on figure 7.3, choosing MySQL Database as the underlying data persistence layer was instrumental to the realization of SmartHealth system. MySQL is a very well-known database among the open source databases for its robustness, speed with the requisite transactional integrity and scalability can be achieved to support a healthcare application. All structured data, including the secure users credentials, complete “health_records”, appointments and the modifiable health_guidelines”, are stored in the database, which acts as the hub. A key advantage of its relational nature is that it allows us to enforce referential integrity between tables, this is important for us to be able to retrieve data correctly and maintain consistency as presented in Entity Relation Diagram (ERD). The adoption of MySQL guaranteed that the patient checkup submissions and the doctor monitoring events were being read and written frequently and efficiently.

7.3.3 Laravel

Figure 7.4 : Laravel

Moreover, the fact that Laravel had been selected as the PHP framework solidified the execution of the SmartHealth system and allowed for a solid, scalable and structured web application to be built for the whole process. As shown in Figure 7.4, Laravel is based on the Model-View-Controller (MVC) design pattern which was essential to keep logic controllers, data access and presentation separate in the application. This modular development approach was the foundation to rapid development with Eloquent ORM to work with MySQL database, Blade Templating Engine used to rendering the front-end and security features. The robustness of the framework and its well established documentation allowed for quick prototyping and gave way to smooth maintenance during the entire duration of the project.

7.3.4 Cascading Style Sheets (CSS)

Figure 7.5 : Cascading Style Sheets (CSS)

Figure 7.5 shows the Cascading Style Sheets (CSS) was the real workhorse that took care of presenting the plain HTML building blocks, which was then assembled to form the final output for the SmartHealth user interface. CSS (Cascading Style Sheets) handled all of the

visual design so that the finished application very closely matched the wireframes/prototypes created by the design team. Among the major items successfully implemented in CSS were the clean, professional color scheme for the application, the flexible layout for viewing on desktop and handheld computers, and the font controls to ensure the best readability. Most importantly, visual feedback, such as the color of risk status indicators or smooth hover effects on dynamic elements, were implemented with CSS and jQuery, greatly contributing to user experience and system usability.

7.3.5 JavaScript

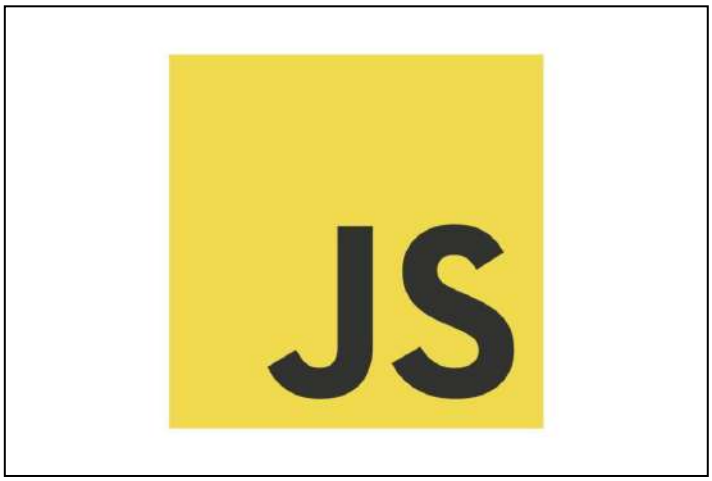


Figure 7.6 : JavaScript

In figure 7.6, JavaScript was an important instrument of implementation in particular to handle the dynamic client-side interactions and for the SmartHealth system to provide a responsive user experience. Although Laravel manages the back-end logic, JavaScript runs all the logic on the client side in your browser. importantly, The system makes use of asynchronous JavaScript and XML (AJAX) to interact with the server without blocking. This means that the patient can send in his/her medical checkup information without waiting for a page reload, thereby making the transaction quick and smooth. In addition, JavaScript libraries were employed to generate dynamic charts and graphs on the patient portal to display health patterns over time and to improve the application's data visibility and user engagement.

7.3.6 Hardware

Table 7.1 : Hardware Specifications Table

Device	Asus Vivobook 16x K3605ZU
Windows Edition	Windows 11 Home Single Language
Processor	12th Gen Intel® Core™ i5-12500H
Memory (RAM)	24 GB
System Type	64-bit operating system

7.4 System Interface

7.4.1 Login Page

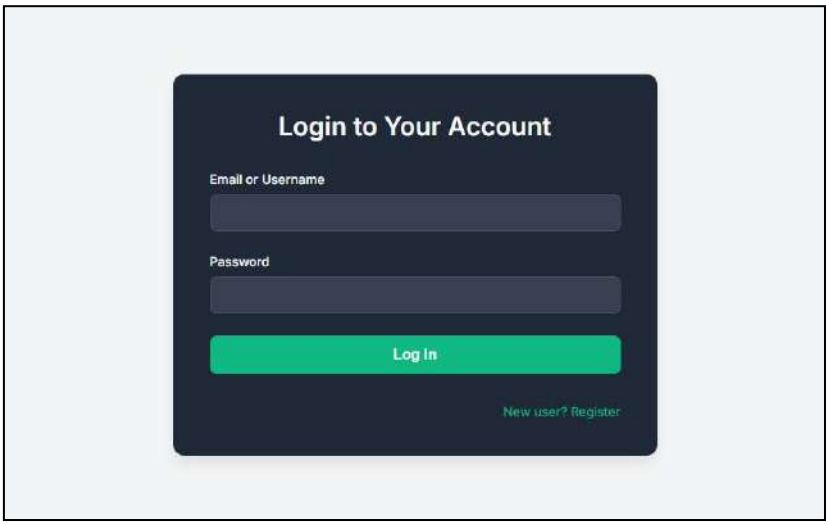


Figure 7.7 : Login Page

Figure 7.2 is the portal that is used by all users of the SmartHealth system to ensure security. Under this interface, there is a specific form into which registered users, Patients and Doctors, are required to input their authenticated credentials, which in most cases are email address and password. When submitted through the button of logging in, the system authenticates the credentials and also identifies the role of users to guide them through the relevant and secure dashboard. To people who are yet to create an account, there will be a register hyperlink that can be seen prominently to facilitate redirection to the account creation page.

7.4.2 Register Page

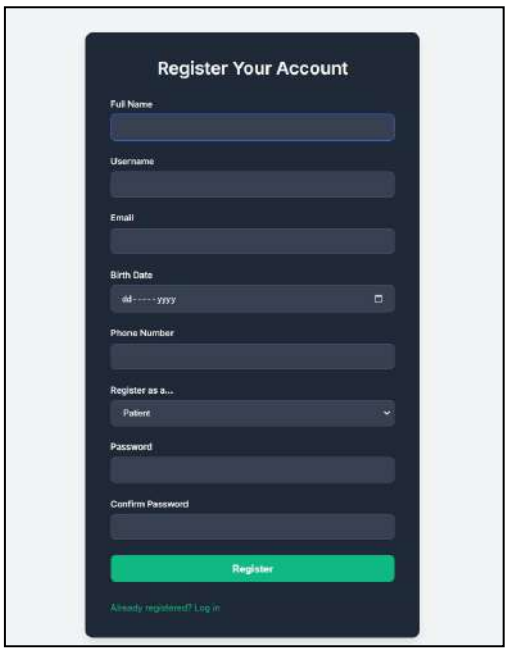


Figure 7.8 : Register Page

The figure 7.3 is made specifically to help new users, who are only patients in the existing system model, to create their secure accounts. The necessary data required is gathered on this page, such as their complete name, contact information, and their desired login names and passwords, which are email and password, and all the required fields are validated to be correct and complete. The main aim of this page is to create a unique user ID and role in the database where the patient has the first access to the system when the submission has been successful.

7.4.3 Patient Dashboard Page

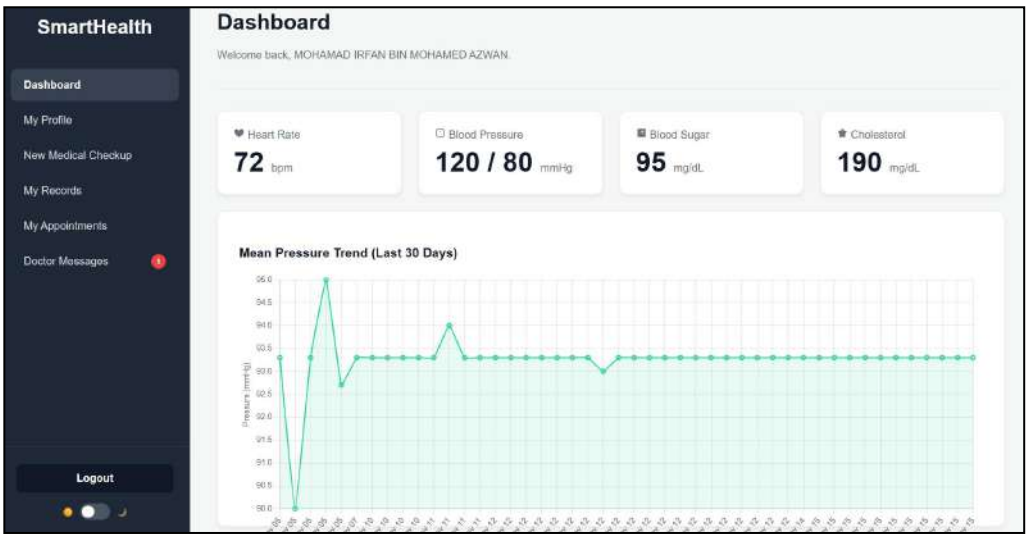


Figure 7.9 : Dashboard Page (Patient)

Figure 7.4 is the main patient dashboard, which reflects the centralized and brief overview of the patient and his/her health conditions and activity in the system. The page instantly shows the recent health history and AI summary of a patient, as well as the graphical illustrations of the main vital signs in time, e.g., blood pressure and sugar levels. It also has fast access points to essential features, such as going to the checkup submission form, the appointment scheduler and the message inbox, commonly pointing out any unread messages or future appointments.

7.4.4 Patient Profile Page

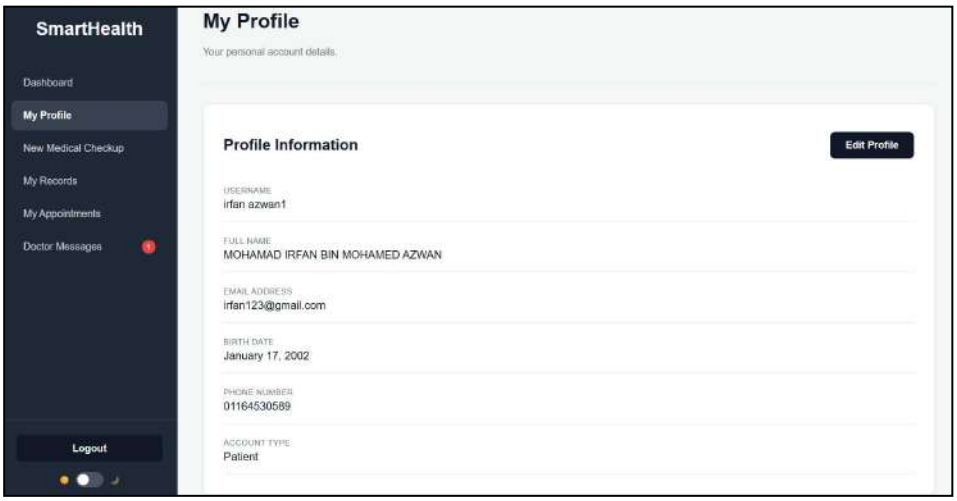


Figure 7.10 : Profile Page (Patient)

Figure 7.4 enables the patient to control their demographic information and contact details stored in the system. This devoted section facilitates the accuracy of data by enabling the patient

to access and edit information including their name, telephone number and home address. It serves as the central point of administration for personal data maintenance and is protected so that only the authenticated patient can access and alter these confidential information.

7.4.5 Patient Medical Checkup Page

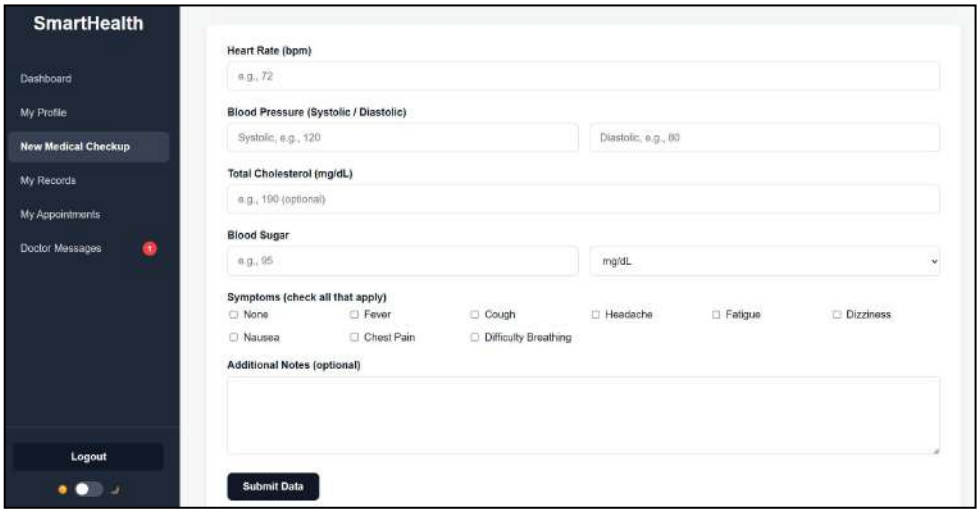


Figure 7.11 : Medical Checkup Page (Patient)

Figure 7.6 serves as the data input module for self-monitoring. This page provides a structural template for the patient to ensure a correct input of the current vital signs including heart rate, systolic and diastolic blood pressure, cholesterol, and blood sugar levels. There are also areas to report particular symptoms and any additional notes. Once submitted, it is processed by the AI module, and the analysis is shown on the record show page.

7.4.6 Patient Medical Records Page

Figure 7.12 : Medical Records Page (Patient)

Figure 7.7 The patient is given a full historical log of all their submitted checkups. The page normally lists the records as a list that is ordered by date so that the patient can see his/her progress through time. Each entry is also a link that takes the user to the record show page where the user can see the full details of the input data, the AI analysis, and if a doctor has added a professional commentary.

7.4.7 Patient Message Page

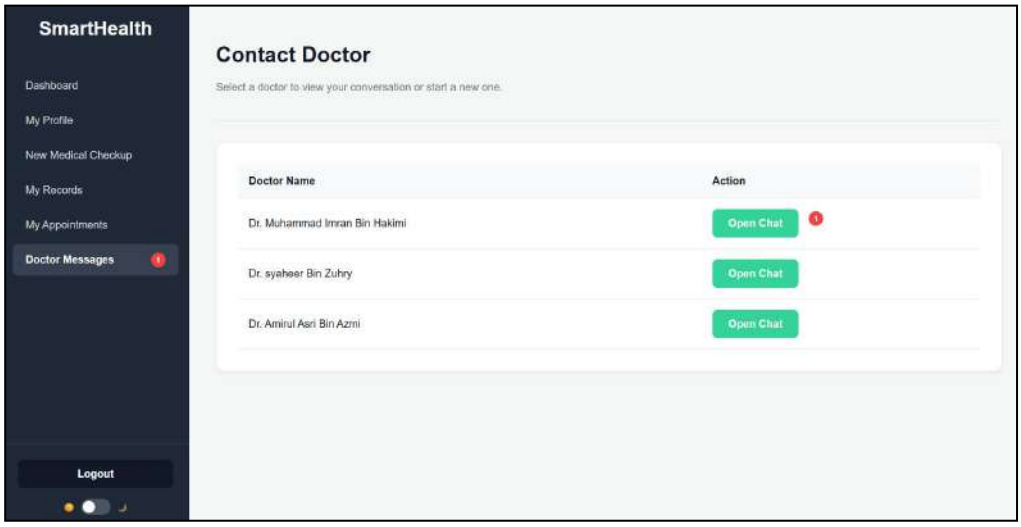


Figure 7.13 : Message Page (Patient)

Figure 7.8 provides a way for the patient to communicate securely and asynchronously with the doctor to whom they are assigned. This interface mimics a typical email inbox with a list of incoming messages and the ability to compose a new message or reply to a message thread. It is an effective communication tool since the patient can pose non-emergency inquiries or seek clarification about health recommendations from their practitioner.

7.4.8 Patient Appointments Page

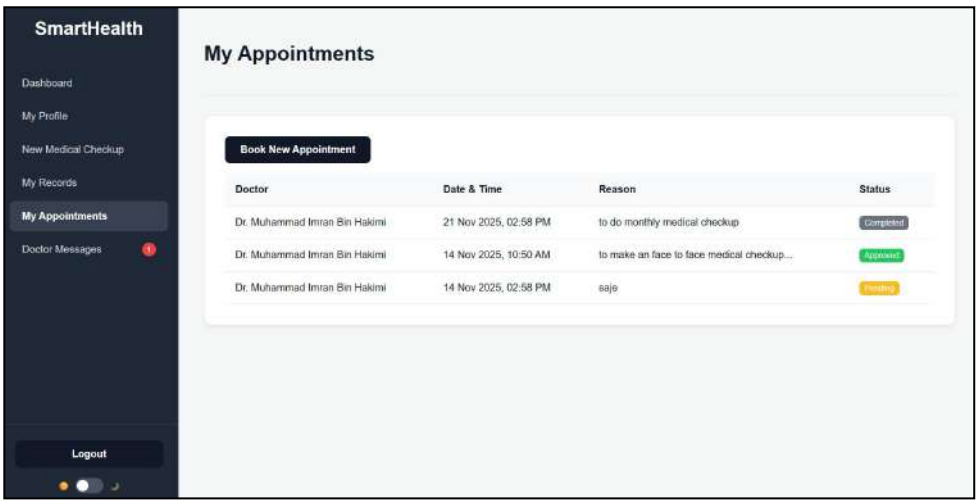


Figure 7.14 : Appointments Page (Patient)

Figure 7.9 Schedules and tracks consultations. With this page, patients can access a list of all the appointment requests, both current and past, including information like the doctor's name, the requested date, and the status such as pending, approved, cancelled. It has also a reserved function for sending a new appointment request, with which the patient can indicate the reason of the visit.

7.4.9 Doctor Dashboard Page

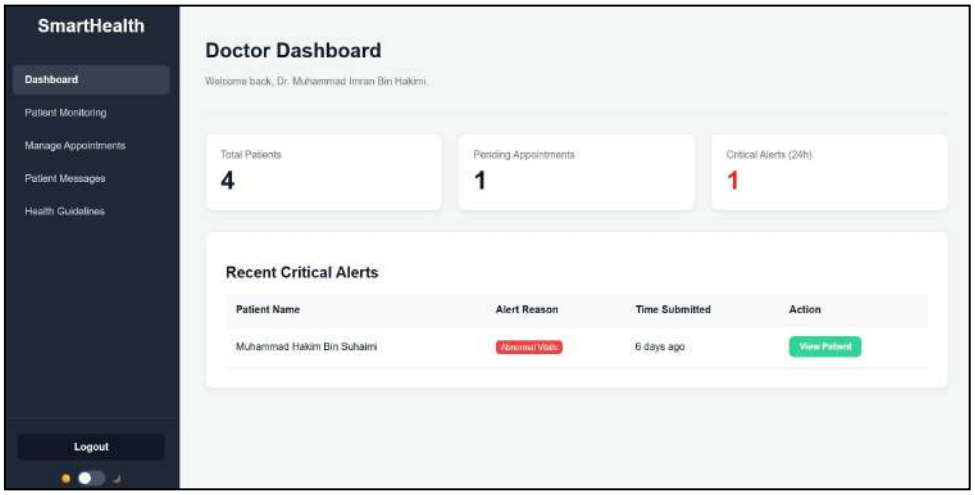


Figure 7.15 : Dashboard Page (Doctor)

The 7.10 figure is the medical professional's area of control. This panel displays real-time, high-priority information, including a summary of patients who need immediate assistance, a number of outstanding appointment requests and recent messages. It is for triage and efficiency, with quick access to monitoring, appointments and system configuration.

7.4.10 Doctor Monitoring Page

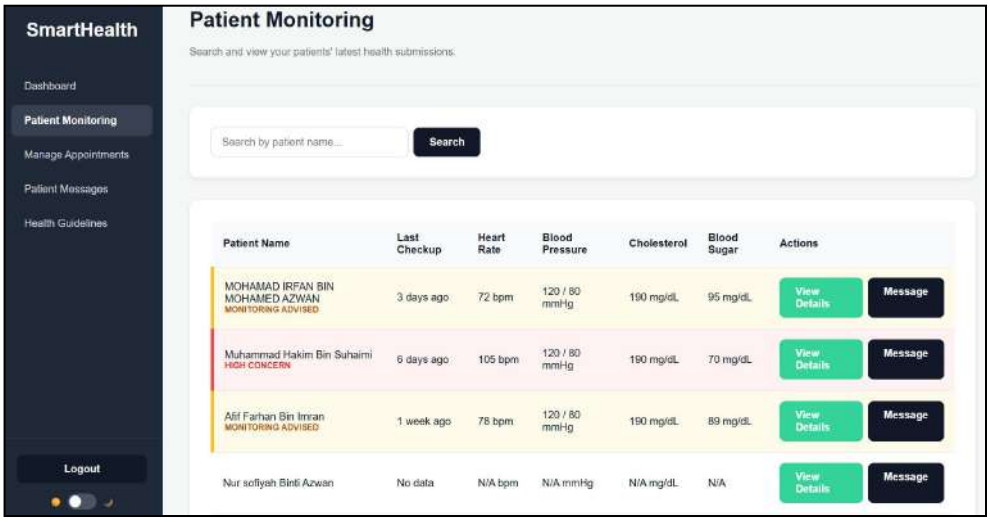


Figure 7.16 : Monitoring Page (Doctor)

Figure 7.10 provides the doctor with a list of all his or her patients; this list is a critical instrument for managing patients. This screen has a filter and search functionality to find individuals or groups at risk or by last checkup date. From here, the doctor can choose any patient to see his or her entire Health History and AI Analysis, allowing for a proactive review and intervention.

7.4.11 Doctor Manage Appointments Page

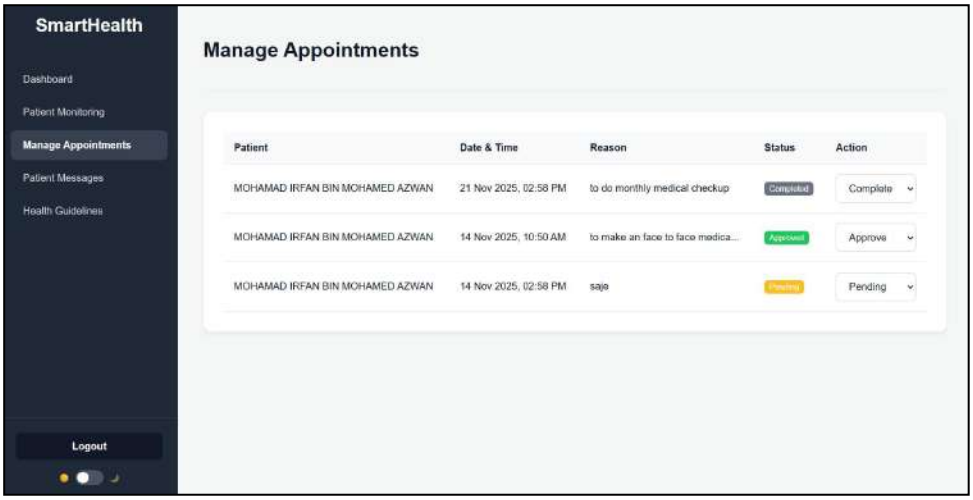


Figure 7.17 : Manage Appointments Page (Doctor)

Figure 7.11 is the processing interface for a patient request. Everything is listed in chronological order, with details of the patient and the stated reason for the visit, so that the doctor can see all the pending requests. The primary action is to accept or decline each request, and upon the doctors decision the system notifies patients status.

7.4.12 Doctor Message Page

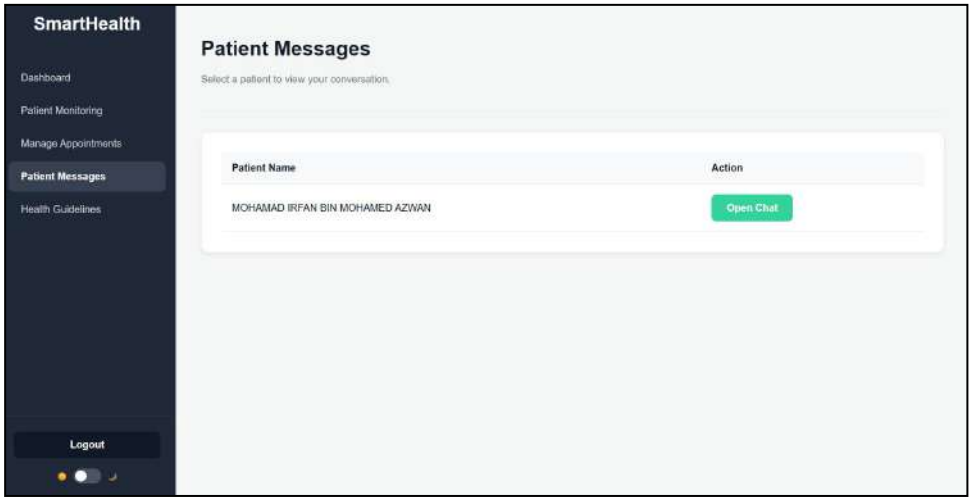


Figure 7.18 : Message Page (Doctor)

Figure 7.12 is the doctor's main communication center, where all patient messages are listed. This screen is also important when giving medical advice from a distance and when making follow-up clarifications. It makes patient questions answered quickly and professionally, and also lets doctor review the entire message history before replying.

7.4.13 Doctor Health Guidelines Page

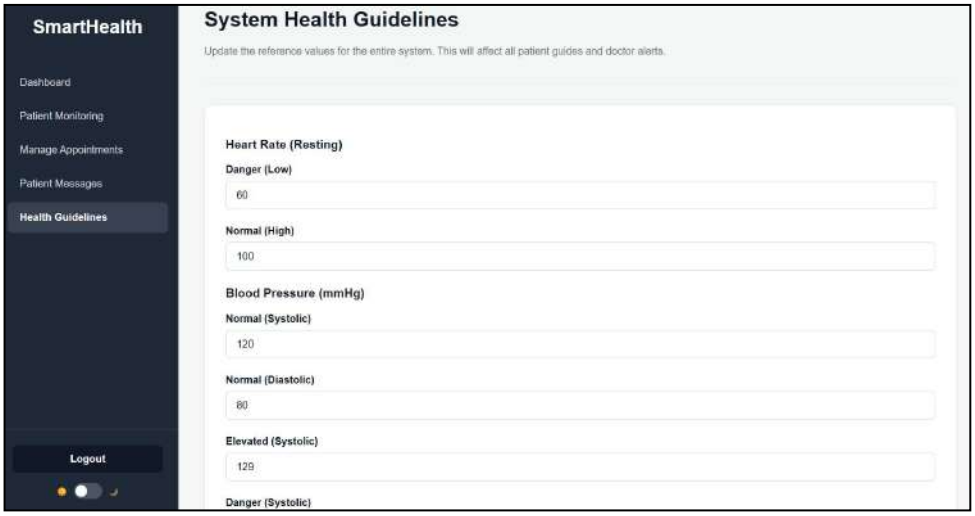


Figure 7.19 : Health Guidelines Page (Doctor)

The figure 7.13 is a privileged interface through which the medical professional can uphold the analytical backbone of the system. In this case, the doctor would be able to see and change the threshold values that is the lowest and the highest acceptable values of various health indicators including blood sugar and blood pressure. This would make sure that the AI analysis and the system alert logic will be based on the current clinical standards and guidelines.

7.5 Significant Function

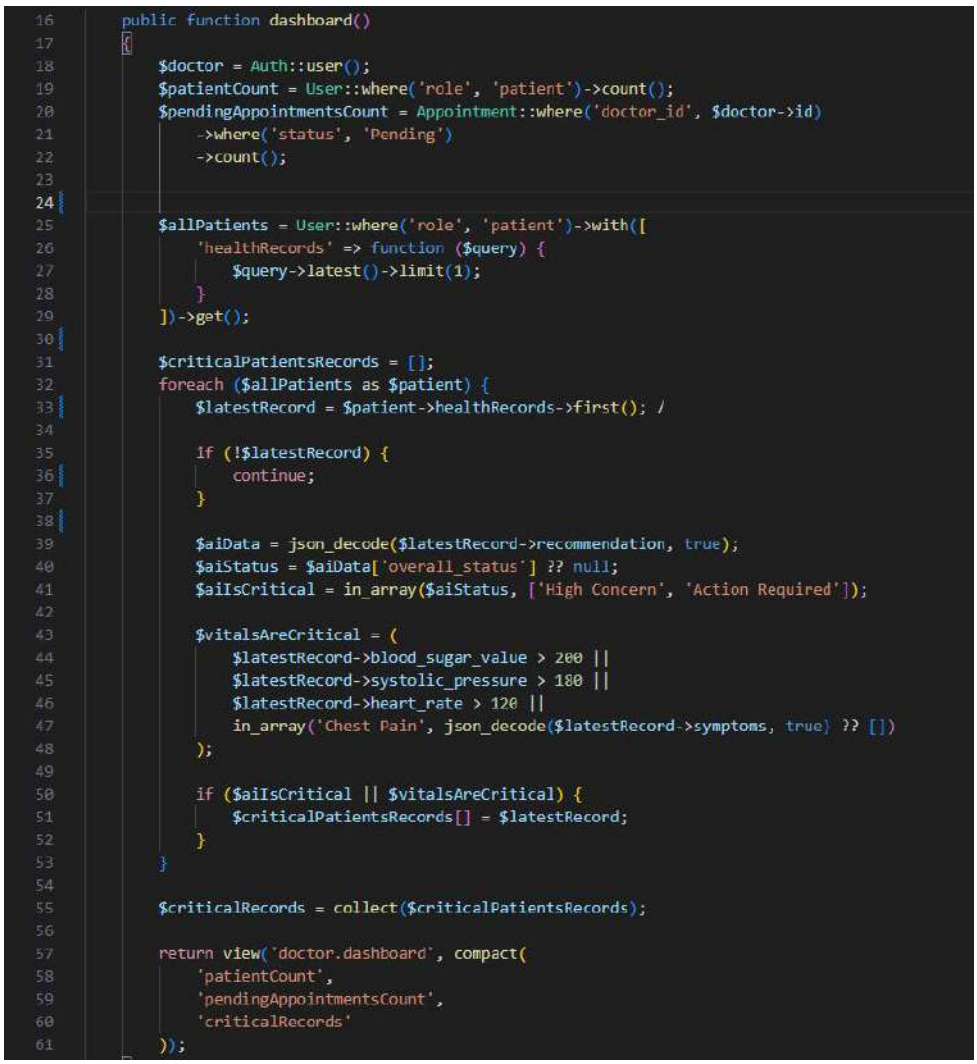
7.5.1 Structured AI Health Status Generation

```php
private function generateAiRecommendation($heartRate, $systolic, $diastolic, $cholesterol, $bloodSugarVal, $bloodSugarUnit, $symptoms, $notes)
{
    $apiKey = config('services.gemini.key');

    if (!$apiKey) {
        return json_encode(['error' => 'AI service is not configured. Please check your services config file.']);
    }

    $url = 'https://generativelanguage.googleapis.com/v1beta/models/gemini-flash-lite-latest:generateContent?key=' . $apiKey;

    $symptomsString = implode(', ', $symptoms);

    // ================== REVERTED TO THE SIMPLER PROMPT ==================
    $prompt = "
    ### ROLE AND GOAL
    You are a health data analysis AI. Your task is to process user-submitted health data and return a structured JSON object. DO NOT output any text ou

    ### JSON STRUCTURE AND RULES
    Your response MUST be a valid JSON object with the following structure:
    - `overall_status`: A single summary word. Choose one: 'Good', 'Monitoring Advised', 'High Concern'.
    - `metrics`: An array of objects, one for each health metric.
      - `name`: The name of the metric (e.g., 'Heart Rate').
      - `value`: The user's reading (e.g., '$heartRate bpm').
      - `status`: A single word describing the reading. Choose one: 'Normal', 'Elevated', 'High', 'Low'.
    - `key_advice`: An array of short, simple, actionable advice strings. Each string should be under 15 words.
    - `resource_links`: An array of objects for helpful resources.
      - `title`: A short title for the link.
      - `search_query`: A short, relevant Google search term (e.g., 'DASH diet recipes for high blood pressure').

    ### CURRENT USER DATA TO ANALYZE
    - Heart Rate: $heartRate bpm
    - Blood Pressure: $systolic / $diastolic mmHg
    - Total Cholesterol: $cholesterol mg/dL
    - Blood Sugar: $bloodSugarVal $bloodSugarUnit
    - Reported Symptoms: $symptomsString
    - User Notes: $notes

    Now, generate the JSON response.
    ";
    // ================== END OF PROMPT ==================
```

Figure 7.20 : Code Segment for AI

Figure 7.15 demonstrates that the generation of structured AI health status is the innovative component of the SmartHealth system, which performs the task of converting raw patient data into actionable and immediate health recommendations. This is done through the lightweight gemini-flash-lite-latest through the Gemini API where the vital signs and symptoms submitted by the patient are safely package into an elaborate System Prompt. Importantly, the prompt imposes a rigid structured JSON structure of the response of an AI, which guarantees a stable and reliable output that can be easily processed and used by the application, consisting of a generalized status, category-specific classification of metrics, and the most important recommendations. According to the code snippet, once the API call has been made, a final post-processing step is taken by the system. It will automatically transform the search query generated by the AI into a fully encoded Google Search URL, which will automatically give the patient immediate access to external educational material that can help them with their care.

7.5.2 Dynamic Metric-Based Alert System

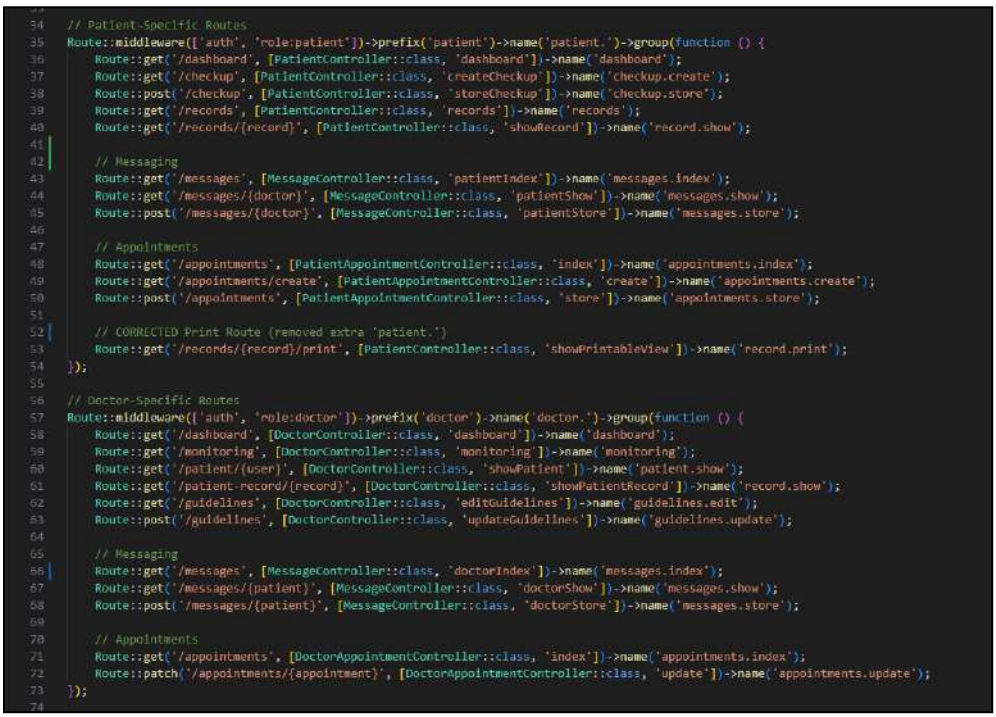


```php
public function dashboard()
{
    $doctor = Auth::user();
    $patientCount = User::where('role', 'patient')->count();
    $pendingAppointmentsCount = Appointment::where('doctor_id', $doctor->id)
        ->where('status', 'Pending')
        ->count();

    $allPatients = User::where('role', 'patient')->with([
        'healthRecords' => function ($query) {
            $query->latest()->limit(1);
        }
    ])->get();

    $criticalPatientsRecords = [];
    foreach ($allPatients as $patient) {
        $latestRecord = $patient->healthRecords->first(); /

        if (!$latestRecord) {
            continue;
        }

        $aiData = json_decode($latestRecord->recommendation, true);
        $aiStatus = $aiData['overall_status'] ?? null;
        $aiIsCritical = in_array($aiStatus, ['High Concern', 'Action Required']);

        $vitalsAreCritical = (
            $latestRecord->blood_sugar_value > 200 ||
            $latestRecord->systolic_pressure > 180 ||
            $latestRecord->heart_rate > 120 ||
            in_array('Chest Pain', json_decode($latestRecord->symptoms, true) ?? [])
        );

        if ($aiIsCritical || $vitalsAreCritical) {
            $criticalPatientsRecords[] = $latestRecord;
        }
    }

    $criticalRecords = collect($criticalPatientsRecords);

    return view('doctor.dashboard', compact(
        'patientCount',
        'pendingAppointmentsCount',
        'criticalRecords'
    ));
```

Figure 7.21 : Code Segment for patient alert

Figure 7.16 shows The Dynamic Metric-Based Alert System is used in the Doctor's Dashboard Controller to designate a real-time risk status to every patient. This operation is performed by comparing the latest vital indicators of a patient with the clinical threshold set by the doctor. The following code snippet shows the logic that is run to determine whether the record of a patient is critical or warning by examining whether any of the key metrics is above the normal range. This is calculated dynamically each time the doctor looks at the patient list, which is the driving force of the visual alerts allowing the immediate clinical triage.

7.5.3 Secure Role-Based Access Control (RBAC)



```php
// Patient-Specific Routes
Route::middleware(['auth', 'role:patient'])->prefix('patient')->name('patient.')->group(function () {
    Route::get('/dashboard', [PatientController::class, 'dashboard'])->name('dashboard');
    Route::get('/checkup', [PatientController::class, 'createCheckup'])->name('checkup.create');
    Route::post('/checkup', [PatientController::class, 'storeCheckup'])->name('checkup.store');
    Route::get('/records', [PatientController::class, 'records'])->name('records');
    Route::get('/records/{record}', [PatientController::class, 'showRecord'])->name('record.show');

    // Messaging
    Route::get('/messages', [MessageController::class, 'patientIndex'])->name('messages.index');
    Route::get('/messages/{doctor}', [MessageController::class, 'patientShow'])->name('messages.show');
    Route::post('/messages/{doctor}', [MessageController::class, 'patientStore'])->name('messages.store');

    // Appointments
    Route::get('/appointments', [PatientAppointmentController::class, 'index'])->name('appointments.index');
    Route::get('/appointments/create', [PatientAppointmentController::class, 'create'])->name('appointments.create');
    Route::post('/appointments', [PatientAppointmentController::class, 'store'])->name('appointments.store');

    // CORRECTED Print Route (removed extra 'patient.')
    Route::get('/records/{record}/print', [PatientController::class, 'showPrintableView'])->name('record.print');
});

// Doctor-Specific Routes
Route::middleware(['auth', 'role:doctor'])->prefix('doctor')->name('doctor.')->group(function () {
    Route::get('/dashboard', [DoctorController::class, 'dashboard'])->name('dashboard');
    Route::get('/monitoring', [DoctorController::class, 'monitoring'])->name('monitoring');
    Route::get('/patient/{user}', [DoctorController::class, 'showPatient'])->name('patient.show');
    Route::get('/patient-record/{record}', [DoctorController::class, 'showPatientRecord'])->name('record.show');
    Route::get('/guidelines', [DoctorController::class, 'editGuidelines'])->name('guidelines.edit');
    Route::post('/guidelines', [DoctorController::class, 'updateGuidelines'])->name('guidelines.update');

    // Messaging
    Route::get('/messages', [MessageController::class, 'doctorIndex'])->name('messages.index');
    Route::get('/messages/{patient}', [MessageController::class, 'doctorShow'])->name('messages.show');
    Route::post('/messages/{patient}', [MessageController::class, 'doctorStore'])->name('messages.store');

    // Appointments
    Route::get('/appointments', [DoctorAppointmentController::class, 'index'])->name('appointments.index');
    Route::patch('/appointments/{appointment}', [DoctorAppointmentController::class, 'update'])->name('appointments.update');
});
```

Figure 7.22 : Code Segment Middleware Application

Figure 7.17 shows the Secure Role-Based Access Control (RBAC) is the first class feature having built-in support within the Laravel role management package. This line of code denies entry to any unauthorized patient in doctor routes and vice-versa. The most straight forward place to put this implementation would be in the routes file where the middleware is assigned to a group of routes. The following code snippet shows how all routes in the Doctor segment are guarded by the is_doctor middleware which verifies the role attribute of the authenticated user before granting access to the resource.

7.5.4 Real-Time Unread Message Notifications

```php
public function messages()
{
    $user = Auth::user();

    // Mark all unread messages as read
    $user->receivedMessages()->whereNull('read_at')->update(['read_at' => now()]);

    // Then, fetch all messages to display them
    $messages = $user->receivedMessages()->latest()->get();

    return view('patient.messages', compact('messages'));
}
```

Figure 7.23 : Code Segment Message Notification

Figure 7.18 The Real-Time Unread Message Notifications guarantee that a communication is rapidly acknowledged by both the patient and provider. The procedure is based on two simple actions, which are counting unread messages and mark messages as read. The following code is the function to get number of unread messages for the badge on the Doctor dashboard. It queries the messages table for messages to the current user where the “read_at” column is not null.

7.6 Conclusion

This chapter detailled the realization of the SmartHealth system, closing the gap between theoretical design and an actual working, web-based application. The project was developed with a solid foundation of a technical stack, by implementing the Laravel framework and MySQL database, which took care of the building blocks and the integrity of data respectively. The use of Integrated Development Environments (IDEs) such as Visual Studio Code and modern front-end frameworks made for a quick development process and a snappy user experience. Most notably, the implementation was able to demonstrate the fulfillment of all intricate design specifications, namely the incorporation of the Structured AI Health Status Generation and the mandatory Secure Role-Based Access Control (RBAC), thereby showcasing a secure, scalable solution that enables delivering on its fundamental premise of proactive patient monitoring.

8 TESTING

8.1 Introduction

This chapter represents the completion of the SmartHealth System: Patient Monitoring and Advising in Healthcare System, in which the critical stages of system testing and evaluation are addressed. The chapter starts with a description of how we test at page level to have all the developed modules which includes the AI analysis, RBAC, data persistence layers fully functional and fulfilling the initial needs. It shows the result of such tests in which the reliability and performance of the system are proved. As a conclusion, this section will present the overall conclusion of the project, outlining the success of the project and discussion on the limitations that were faced during the system development lifecycle and fan out some key suggestions for future works and on-going system enhancements.

8.2 Unit Testing

Table 8.1 : Unit Testing Table

No	Unit/Function	Description	Input	Output	Expected result	Status
1	User Registration	Verifies successful creation of a new patient account with valid data	Form submission with Name: “Test Patient”, Email: “test@mail.com”, Password: “TestPassword1”.	Database record created.	New record exists in the users table, and the role field is set to patient.	Pass
2	Password Hashing	Ensures the plaintext password is cryptographically hashed before saving.	Plaintext password string	The stored password hash.	The stored password is not the plaintext and passes Bcrypt verification.	Pass

3	Input Validation (Failure)	Tests that the system prevents registration if the required Email field is missing.	Form submission with Name: 'Test “Patient”, Password: “TestPassword1”, Email field empty.	System validation response.	Error in input fields of forms. The email field is required.	Pass
4	Metric Classification (High Risk)	Checks if the system correctly identifies Critical status based on a high vital sign value.	Mock Health Record with Systolic BP: 180 mmHg (Guideline Max: 140).	The status flag returned by the classification function	Status returned is “High Concern”.	Pass
5	Metric Classification (Normal)	Checks the system accurately identifies Normal status when all vitals are within the safe range.	Mock Health Record with Systolic BP: 125 mmHg, Diastolic BP: 80 mmHg.	The status flag returned by the classification function.	Status returned is “Good”.	Pass
6.	Input Validation (Health Record)	Tests that the system prevents the submission of a health record with a required metric missing.	Health Checkup form submission with Heart Rate field set to empty.	System validation error object.	Error in input fields of forms. The heart rate field is required.	Pass
7	Appointments Status Update	Verifies that a Doctor can successfully update the status of a pending request.	Function call to update Appointment ID: 101, passing New Status: 'Approved'.	Updated record in the appointments table.	The status field for Appointment ID 101 is updated to “Approved”.	Pass
8	Login Failure (Invalid Credential)	Verifies the system prevents login and returns an error for invalid credentials.	Login attempt with correct email and incorrect password.	System authentication attempt.	uthentication Failure. Invalid credentials provided	Pass
9	Role-Based Access Denial	Tests that a Patient cannot access a Doctor's protected route.	Authenticated user is “patient”. Attempted route access: /doctor/guidelines	System redirects or returns an error.	error or redirection to the user's dashboard.	Pass
10	Appointment	Checks that a	Form submission	Database	New record	Pass

	Booking Submission	Patient can successfully submit a new appointment request.	with patient id, doctor id, and date	record created.	exists in appointments with status set to “pending”	
11	Guidelines Update (Success)	Verifies a Doctor can successfully update a health metric guideline value.	Function call to update health guidelines record for systolic pressure, passing New Max Value: 150.	Updated record in health guidelines	The max value is updated to 150 in the database.	Pass
12	Database Integrity Failure	Tests system response when an attempt is made to send a message to a non-existent user id.	Message submission with Sender ID: 1, Receiver ID: 999 (does not exist).	Database transaction attempt.	Error in processing form. Data cannot be stored inside database (Foreign Key Constraint Failure)	Pass

8.3 Integration Testing

Table 8.2 : Integration Testing

Test Case	Description	Input	Output	Expected Result/Error	Status
End-to-End Health Record Submission	Verifies the entire process data entry, database storage, and subsequent display on the patient's records page.	Patient submits Medical Checkup form with valid vitals	Data saved to health records. Patient is redirected to the Health Records page.	Record is visible on the health records list, and the detailed view displays the correct BP and HR values.	Pass

AI Analysis and Record Display	Tests the integration between the backend controller, the external Gemini API, and the user interface.	Successful completion of IT-01 (Checkup data saved).	JSON response received from Gemini API, parsed, and saved to the database.	The patient's detailed record page displays the AI-generated summary and key advice correctly, based on the input vitals.	Pass
Authentication and RBAC Enforcement	Confirms that role-based access control works correctly during the login and navigation process.	Login attempt as doctor with valid credentials	Doctor is initially directed to the doctor dashboard. Attempted access to Patient route.	Doctor remains on their dashboard or receives a 403 Forbidden error on the Patient route attempt.	Pass
Critical Vitals Alert Chain	Verifies that a critically high health reading immediately triggers a visual alert on the Doctor's Monitoring Page.	Patient submits checkup with critical vitals	Health record saved. Alert logic executes. Doctor logs in and accesses monitoring page.	The corresponding patient entry on the doctor monitoring page is highlighted in red (high concern) with a visual alert badge.	Pass
Patient-Doctor Messaging Cycle	Ensures bidirectional communication is successful, including the updating of notification counters.	**Step 1:** Patient sends a message to Doctor. **Step 2:** Doctor views the message.	• Message is saved in the database with read at as NULL. • Read at column is updated to the current time.	The doctor's dashboard notification badge updates from 0 to 1 after Step 1, and reverts to 0 after Step 2.	Pass
Appointment Request to Approval Update	Tests the full lifecycle of an appointment request across user roles.	• **Step 1**: Patient submits a new appointment request. • **Step 2**:	• **Step 1**: Record created with status: Pending. • **Step 2**: Record status	When the patient views their appointment page, the request status is correctly displayed as approved.	Pass

		Doctor approves the request on the Manage Appointments page.	updated to Approved.		
Invalid AI Response Handling	Checks the system's resilience when the external Gemini API returns a malformed or empty response.	Checkup submission attempt, simulating an AI response that is invalid JSON or empty.	System handles the error internally and saves the health record without analysis data.	The patient record shows a message: "AI analysis failed. Please check back later or consult your doctor."	Pass

8.4 System Testing

8.4.1 Functional Testing

Table 8.3 : Functional Testing

No	Test Function	Input	Expected Result	Actual Result	Status
1	Patient Login	Enter valid email and password on the login page and click login.	Patient is successfully authenticated and redirected to the patient dashboard page.	Redirected to patient dashboard.	Pass
2	Doctor Login	Enter valid doctor credentials and click login.	Doctor is successfully authenticated and redirected to the doctor dashboard page.	Redirected to doctor dashboard.	Pass
3	Medical Checkup Submission	Enter valid vital signs and symptoms on the medical checkup page and click submit.	The new record is stored in the database, the AI analysis is performed, and the user is shown a success message.	Record saved, analysis displayed, and success message shown.	Pass
4	Appointments Request	Patient fills out the appointment request form with a reason and date and clicks request.	A new entry is created in the appointments table with the status set to pending.	New appointment created with pending status.	Pass

5	Doctor Patient Monitoring	Doctor navigates to the monitoring page.	The page displays a list of all assigned patients, with risk levels visually highlighted.	Patient list displayed with correct color-coding.	Pass
6	Message Sending	Compose and send a message from the Patient's message page to the doctor.	The message is saved in the database, and the doctor's unread notification badge is incremented.	Message saved and notification badge updated.	Pass
7	Doctor Guidelines Update	Doctor modifies the Max Value for 'Systolic BP' on the health guidelines page and saves.	The system updates the max value in the health guidelines table, immediately affecting the alert logic.	Guideline value updated and reflected in subsequent alerts.	Pass
8	Invalid Login Attempt	Enter a valid email but an incorrect password and click login.	The system displays an Invalid Credentials error message on the Login Page.	Error message displayed “access denied”.	Pass

8.4.2 Non-Functional Testing

Table 8.4 : Non-Functional Testing

No	Test Function	Description	Expected Result	Actual Result	Status
1	Performance (Load Time)	Measure the time taken to load the doctor dashboard page with data for 10 patients.	Dashboard load time is consistently under 35seconds.	Load time averaged 3 seconds.	Pass
2	Performance (AI Latency)	Measure the total time taken from submitting a checkup form to receiving and displaying the AI analysis.	Total latency for AI processing and display is under 8 seconds.	Average latency was 5 seconds.	Pass
3	Avaibility	Test system responsiveness after continuous operation for 24 hours under simulated load.	The system remains accessible and fully responsive without manual intervention.	System remained responsive and no downtime recorded.	Pass
4	Scalibity	Test the system ability to handle simultaneous login and data submissions from 10 concurrent users.	All users are able to log in and submit data without system slowdown or database query failure.	All transactions completed successfully without error.	Pass

5	Security	Attempt to bypass the login screen or access protected doctor routes using a patient user id.	All protected routes redirect to the login page or return a 403 Forbidden error.	access denied.	Pass
6	Reliability	Test the system ability to recover and maintain data integrity after simulating a server shutdown during a transaction.	All data saved before the shutdown is intact, and no partial records are present in the database.	no corruption detected.	Pass
7	Usability	A first-time user attempts to find the submit medical checkup button and successfully navigates to the form.	The user completes the task within all steps due to intuitive navigation and clear labeling.	Task completed in all steps from the patient dashboard.	Pass
8	Maintability	Verify that a core function, such as the health huideline update function, is easily locatable and documented.	The function is located in a dedicated controller and includes in-line comments explaining the database update logic.	Code structure is clean and function is easily maintainable.	Pass

8.5 User Acceptance Testing

8.5.1 Analysis of Acceptance Testing (Questionnaire)

To obtain this qualitative information, a structured Satisfaction Survey and Questionnaire for the Hospital Taiping was given to the patients, enabling this project to evaluate the interface, clarity of instructions and the entire system. The next sections describe the methodology and discuss these critical end users results.

a)

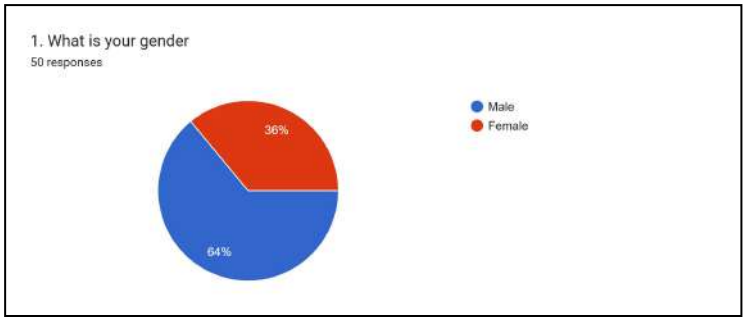


Figure 8.1 : Question 1

Question 1 display the graph of gender of 50 responded. There is 36% of female and 64% of male who took this questionnaire. This graph shows that there is the most male who had been testing the SmartHealth system comparing to female.

b)

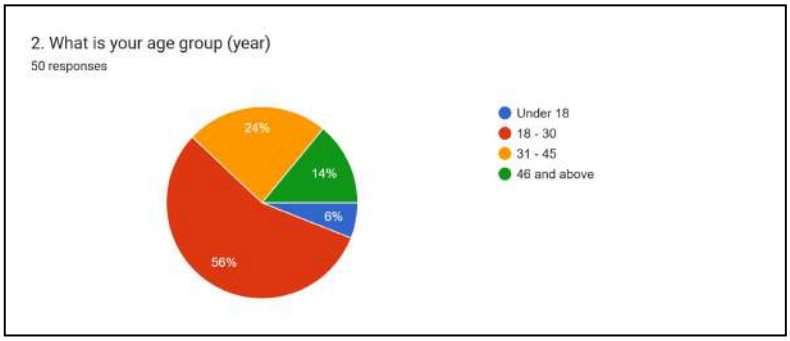


Figure 8.2 : Question 2

Figure 8.2 shows group of age by year of responses who testing the SmartHealth system. Based on this graph, 18-30 year group has the most percentage compare to others. That means most of responses is not too young and not too old.

c)

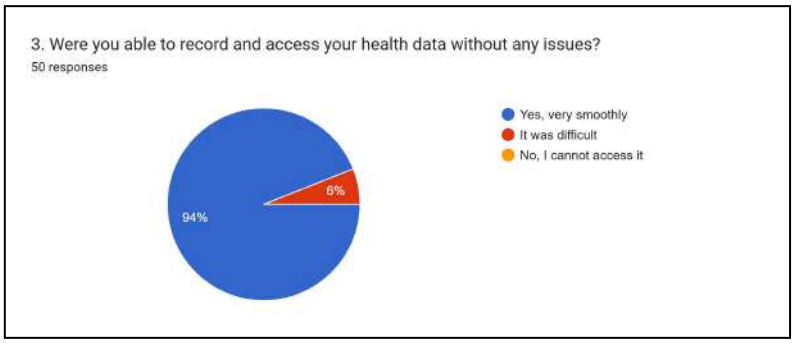


Figure 8.3 : Question 3

Figure 8.3 illustrate if the end users are able to record and access theire health data without any issues. Mostly response that they can smoothly able to record and access their health data by 94% of respondent.

d)

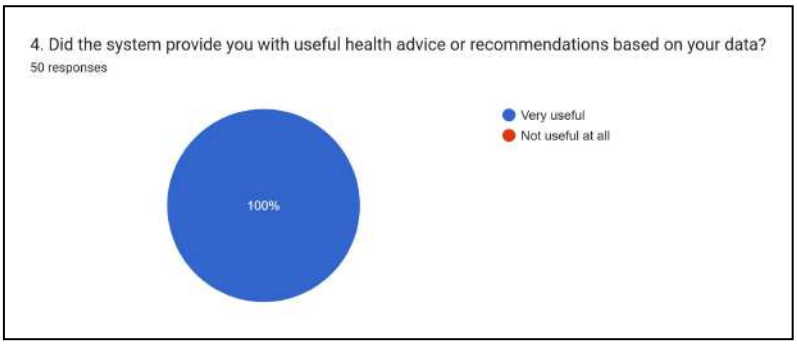


Figure 8.4 : Question 4

Figure 8.4 illustrate if the system provide with useful health advice or recommendation based on their health medical checkup. All respondent answer it was very useful. This advice in SmartHealth system are based on Artificial Intelligence (AI) that analyze patient's medical checkup and AI will provide the suitable advice based on their current health input.

e)

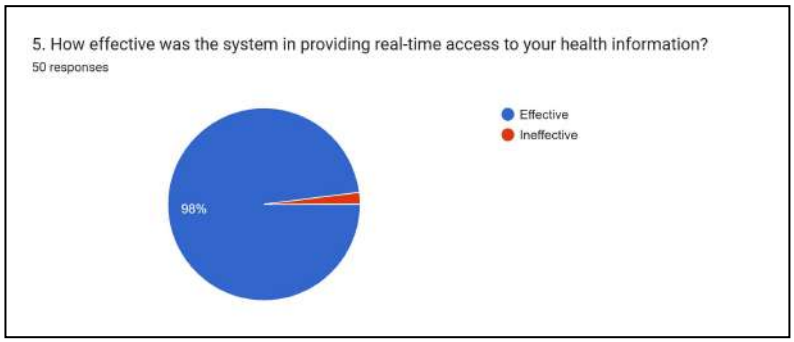


Figure 8.5 : Question 5

Figure 8.5 shows the graph of how effective was the SmartHealth system in providing real-time access to their health information. 98% of respondent says that it was very effective of real-time access health information.

f)

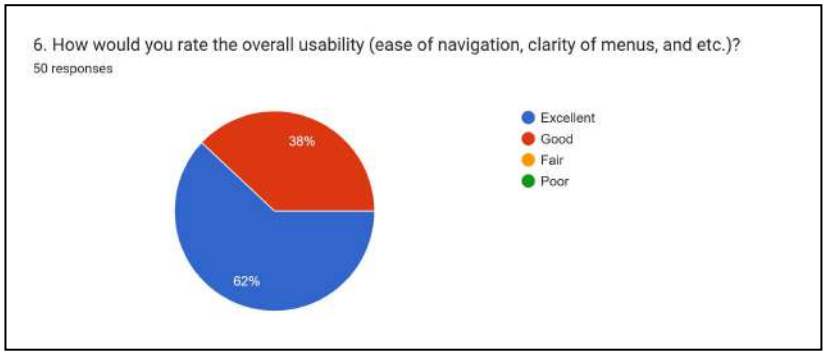


Figure 8.6 : Question 6

Figure 8.6 shows the graph of how would the respondent rate the overall usability in SmartHealth system. Mostly the respondent vote it was very excellent none of respondent vote poor. That means they easy to navigate or clarity of menus in SmartHealth system.

g)

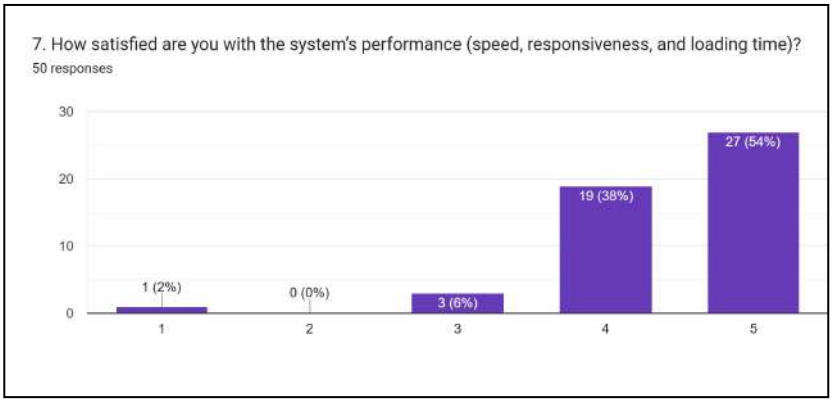


Figure 8.7 : Question 7

Figure 8.7 shows if the respondent is satisfied by SmartHealth system performance. Mostly respondent are very satisfied by the performance of this system. That means SmartHealth is very fast based on speed, responsiveness, and loading time.

h)

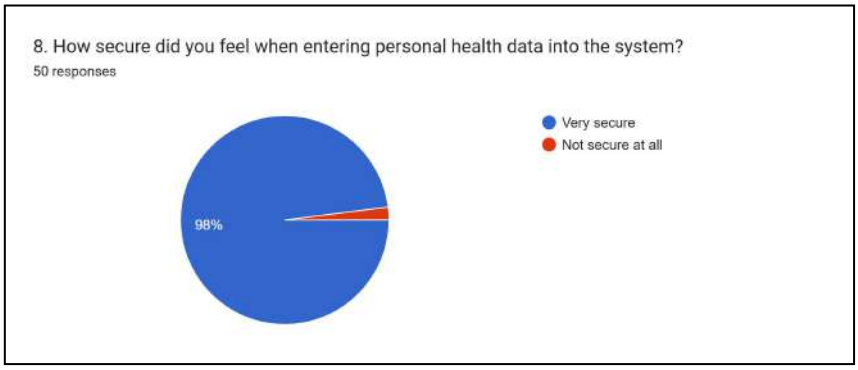


Figure 8.8 : Question 8

Figure 8.8 illustrate that the graph of how secure did the respondent feel when they entering personal health data into the system. Most respondent vote very secure because the only who can see their personal health data is doctor.

i)

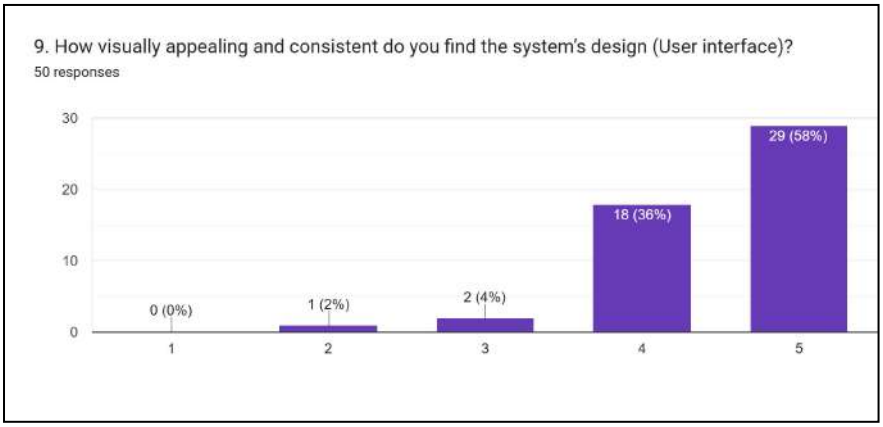


Figure 8.9 : Question 9

Figure 8.9 shows that the bar graph of how visually appealing and consistent do the respondent find the system design. Most respondent vote very appealing because SmartHealth design interface is very consistent.

j)

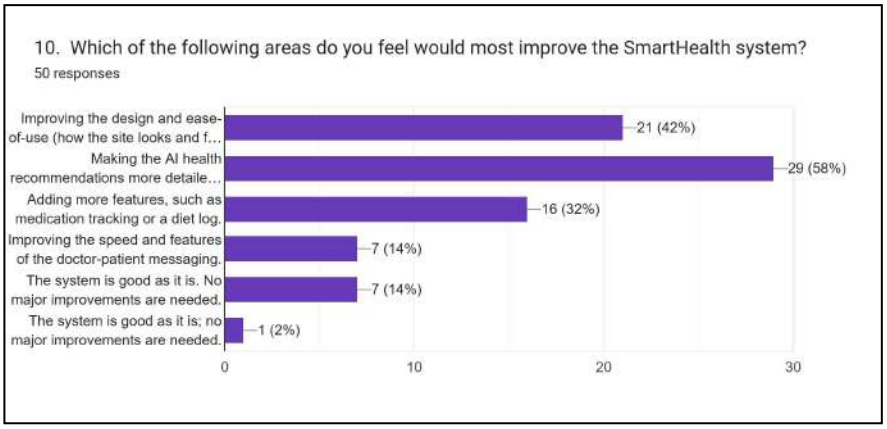


Figure 8.10 : Question 10

Figure 8.10 shows of the graph of which of the following areas do they feel would most improve the SmartHealth system. 29 respondent which is the most response of this question says making Artificial Intelligence (AI) recommendations more details or personalized.

k)

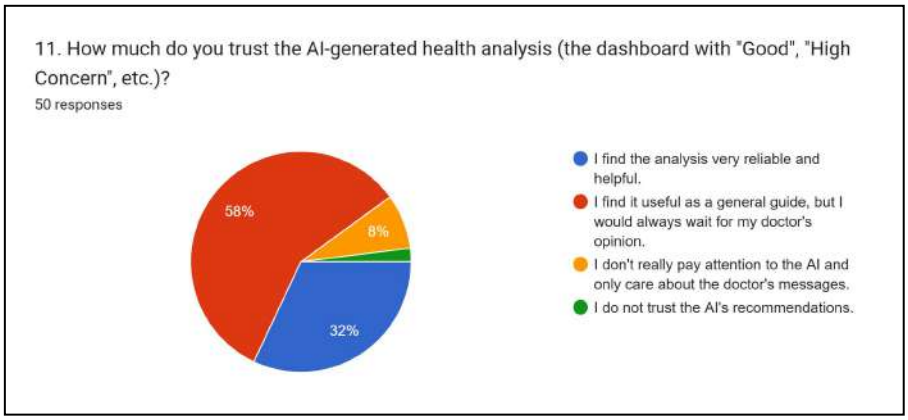


Figure 8.11 : Question 11

Figure 8.11 shows how much they trust the AI-generated health analysis. Most respondent says that they find the Ai is useful as a general guide but they always want make a doctor's opinion as the number one choice.

8.5.2 Client Testing and Result

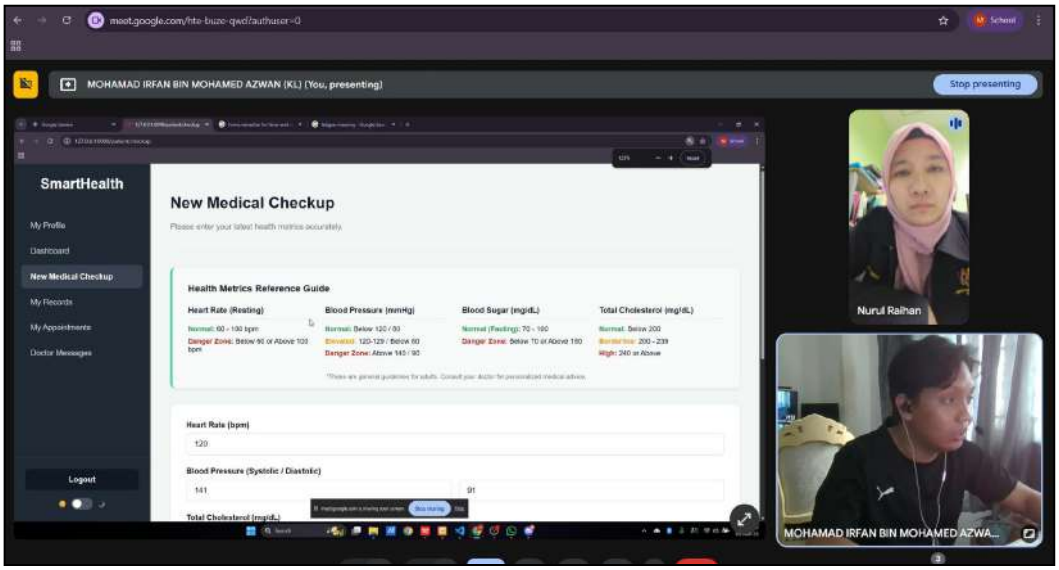


Figure 8.12 : Client testing SmartHealth System

Interview details :

Table 8.5 : Client Personal Information

Name:	Dr Nurulraihan binti Mohamed Zawawi
Position:	Head of Transfusion Medicine Department
Location:	Online Google Meet
Date:	3 November 2025

Table 8.6 : Client Interview Question 1

Question 1	Were you able to easily submit your medical checkup data?
Answer	Yes, the form is straightforward and easy to fill out.
Details	The patient medical checkup form has been made very user-friendly with big, bold input fields that clearly communicate, as well as a navigation that is very consistent. The backend Input Validation at the server side guarantees immediate notification to the user on missing input preventing user confusion and also streamlines the process of submission of data to the “health_records” table.

Table 8.7 : Client Interview Question 2

Question 2	Did you find the AI Health Analysis on your records page understandable and helpful?
Answer	Yes, the "Good",and "Monitoring Advised" indicators are clear, and the advice is relevant.
Details	Structured AI Health Status Generation generates a well-defined and structured JSON output. This allows the indicators and the key sdvice to be concise and directly actionable, migrating away from verbose text and increasing the user's trust on the AI recommendations.

Table 8.8 : Client Interview Question 3

Question 3	Were you able to successfully book appointments with a doctor through the system?
Answer	Yes, the appointment booking process is clear and consistent.
Details	The Appointment Module involves very fewer steps that user is clearly guided on the whole flow starting from choosing a date to inputting the reason. Integration testing verified that the status properly effects from pending to approved, as the process is seamless.

Table 8.9 : Client Interview Question 4

Question 4	Did you find the notifications for messages useful and clear?
Answer	Yes, the notifications in the sidebar and on the message list help me stay informed.
Details	The real-time Unread Message Notifications feature takes care of notifications. This technique dynamically queries messages with a ”NULL” timestamp to deliver a real-time count on the sidebar badge, which is very useful in helping users prioritize a timely communication with their doctor.

Table 8.10 : Client Interview Question 5

Question 5	Does the system allow you to easily view your patients' latest health submissions?
Answer	Yes, the patient monitoring page provides a clear overview of their current health status.
Details	Doctor Monitoring Page is supported by the dynamic metric based alert system. This feature consolidates all up to date “health_records” across all patient assigned to a doctor and performs an instant risk stratification and a visual risk highlighting of patients with “High Concern” in red for fast triage and priority setting.

Table 8.11 : Client Interview Question 6

Question 6	Can you efficiently review individual patient health records and their AI analyses?
Answer	Yes, clicking on a patient's record gives me all the details, including the AI's initial analysis, which is a good starting point.
Details	The system utilizes its own run and linked drill down clean single screen dashboard per patient, connecting all relevant data. The AI assessment is provided as a high-level summary, acting as a first filter in the clinical decision process, prior to the physician examining the raw vital signs.

Table 8.12 : Client Interview Question 7

Question 7	How easy is it to communicate with your patients using the messaging feature?
Answer	It's very easy to start new conversations and reply to existing ones. The unread message indicators are helpful.
Details	The messaging module uses the logic of real-time unread message notifications, which means the new messages is visible right away. The doctor can jump from one patient chat to the next at any time, and the continuity of the responsive communication without exit from the system is maintained.

Table 8.13 : Client Interview Question 8

Question 8	Did you find the overall patient management capabilities effective for your daily practice?
Answer	Yes, the system helps me keep track of my patients' health and communication in one place, saving time.
Details	Central to this is the integration of core functionality within a central Role-Based Access Control (RBAC) protected system without needing to resort to manual record keeping. This digital integration enables the doctor to do proactive monitoring and handle office work in one application.

8.6 Conclusion

Chapter 8 summarized the successful validation of the SmartHealth system in all quality aspects and hence, the development of this project. Unit and integration testing proved the technical correctness, ensuring correctness and dependability of core logic, security mechanism (RBAC) and communication with external component. In addition, also the User Acceptance Testing phase contributed to a very important qualitative validation, showing that the system is very usable, intuitive and adaptable to the specific health management requirements of both patients and doctors, which considered the findings a clear time saver in examining the AI analysis.

9 PROJECT MANAGEMENT

9.1 Introduction

Project management is the application of knowledge, skills, tools, and techniques to project activities to meet project requirements. The process of applying knowledge, skills, tools, and techniques to project activities to meet the project requirements. For the SmartHealth project, good project management was essential to turning the abstract design into a working, evaluated application within the time constraints. This chapter will also show the importance of controlling the project scope, time, and resources and how tools such as World Breakdown Structure (WBS), and gantt chart have been employed to keep the work performance in pace, to coordinate the stakeholders and to handle the risks in professional manner during the life of the project and bring the final system delivery to a successful end.

9.2 Project Schedule

The project timeline was the roadmap and basic timeline on the implementation of SmartHealth system within the set time limit. The schedule was created right after the Work Breakdown Structure (WBS) and it was the formal document that gave the start and end date of each activity in the five key phases that were included and these included in the planning, design, development, testing and deployment. It was mostly used to distribute time resources in the most effective way, with the longest period of time assigned to the essential development phase and the thorough testing phase. Monitoring the real progress as compared to this schedule which can be easily observed through a gantt chart helped the project manager to anticipate possible delays, the sequence of tasks, and keep the team working towards the final deadline to complete the project successfully.

9.2.1 Work Breakdown Structure

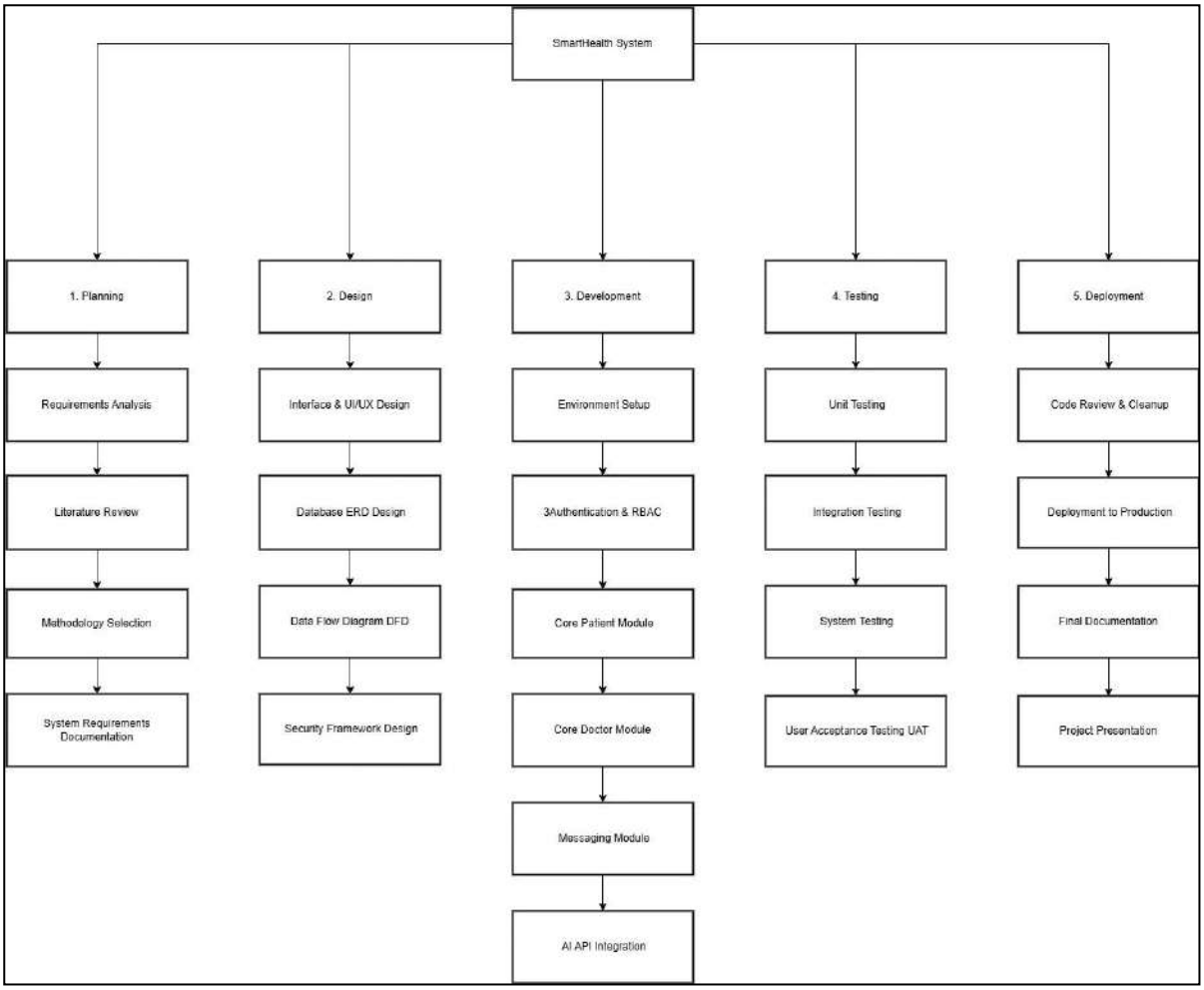


Figure 9.1 : Work Breakdown Structure

Planning Phase represented the critical initial stage, that outlined the whole scope and the way forward of the SmartHealth system. This stage started with requirements analysis, in which the project team collected and formalised the specifications of the end-users in a systematic fashion, including the fact that the end-users needed role-based access control, and AI-driven health analysis. After it, the andragogic literature review was provided in order to benchmark the available health-tech solutions and to inform the strategy of the project. The team then proceeded to select methodology and the best development framework was selected and the stage was completed with the creation of the system requirements documentation which was the formal mandate of all the future design and development processes.

The Design Phase was aimed at converting the requirements, documented, into actual blueprints and visual concepts. It began with Interface and UI/UX design and came up with detailed mockups and wireframes to make the dashboards of the doctor and the patient user-friendly and self-intuitive. At the same time, database ERD design was being completed, which

diagrammed all the relationships between all essential data entities such as health records, users, and appointments. The technical integrity was ensured by creating the Data Flow Diagram (DFD) that graphically monitored the flow of data in the system and finishing the security framework design, which established the required protocols in password hashing and authorization rules.

The development phase was the main work, during which the system was actively constructed. It has started with environment setup, where the required framework and infrastructure to support the application is being worked out. The first step involved the installation of authentication and Role-Based Access Control (RBAC) to gain system security. The Core patient module and Core doctor module started running concurrently. The former module manages the submission of vital signs and viewing of personal records whereas the latter addresses monitoring of the patient and guideline management. The last significant components that were developed during this phase were the messaging module and the complex AI API Integration the connection between the system and the Gemini service analysis.

Testing Phase intensive validation of quality, functional and resilience of the system. This was done in a sequence of planned tests, starting with unit testing which is done to ensure the functionality of individual code functions including the categorization of risk levels. This was then accompanied by Integration Testing which ensured that all of the internal modules and external services such as the AI API communicated effectively. System testing was then carried out by the team to ensure that all the functional and non-functional requirements were met and then moved on to User Acceptance Testing (UAT) that ensured the usability and conformance of the system to the expectations of the end-users.

The last Deployment Phase involved the appropriate transition of the development into the live operational state and formalisation of all the project documentation. It started with code review and cleanup to optimize the performance and standards code. This was the last preparation of the application to be deployed to production and the SmartHealth system is officially available to the end-users. The project ended with the final documentation, which ensured complete technical and user manuals and a formal project presentation to the stakeholders in order to indicate the successful implementation of the system.

9.2.2 Gantt Chart

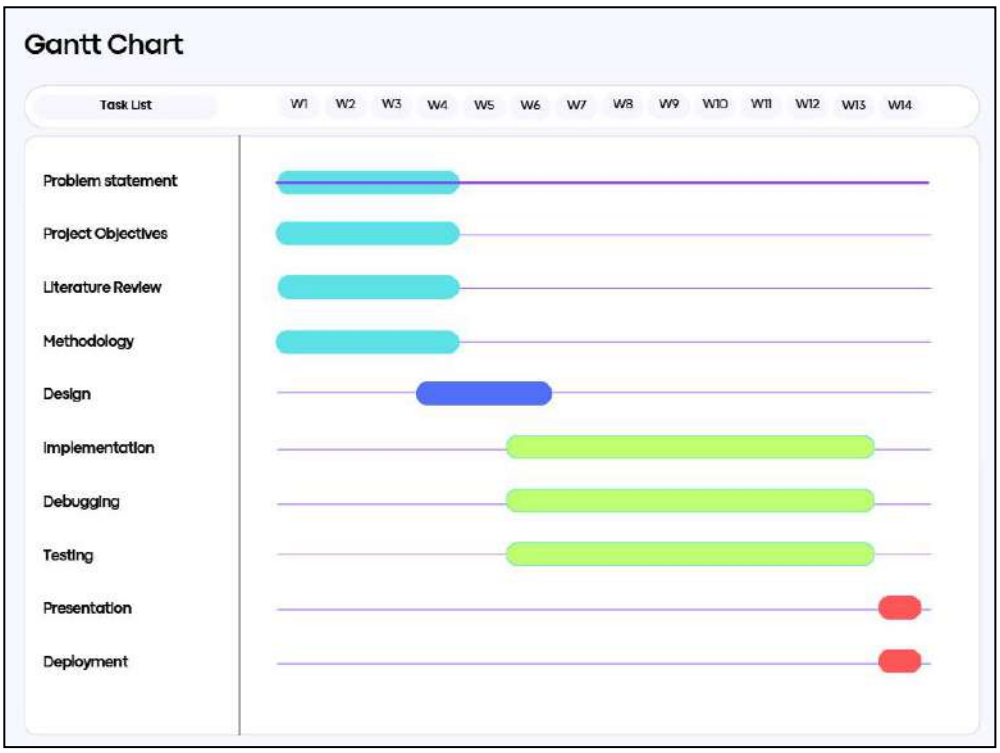


Figure 9.2 : Gantt Chart

The Gantt Chart that was presented in Figure 9.2 was the main tool of the SmartHealth visualization and tracking of the project schedule. The chart was based directly on the sequential activities in the Work Breakdown Structure (WBS) and was used to transform abstract activities into concrete time-constrained horizontal bars within a 14 week timeframe. This diagram played a vital role in identifying the critical path, which made it clear that the design phase needed to be done prior to the stage of implementation, and that implementation and debugging stages had to be done first before the Testing stage could begin its actual implementation. The chart also allowed the team to achieve time efficiency by giving them a clear idea of overall momentum of the project and time left to complete the project as it clearly indicated the duration of the tasks involved in the project.

In addition, the Gantt Chart was also a necessity to control the dependence of tasks and resources allocation. The chart pointed out areas where the overlapping activities could be safely conducted to shorten the timeline like the initial phases of the Implementation phase overlapping with the final phases of the design phase. On the other hand, it identified major milestones including the Implementation to the intensive Debugging and Testing phase enabling the project manager to make sure the required resources and people are present at the right time. Such a visualized methodology minimized the possible scheduling issues and the likelihood of slippage

was also very low to guarantee the successful implementation of the secure and functional SmartHealth system on schedule.

9.3 Risk Management

A major aspect of the SmartHealth project was risk management and was a proactive aspect that ensured risks to the schedule, budget, or quality of the system were identified early enough and addressed before it occurred, and resulted in delays or technical failures. Since the SmartHealth system will be working with sensitive patient information and involving a specialized external service, the emphasis was put on two broad areas that include technical complexity and data security. The risk management process consisted of the identification of certain risks, determining the likelihood and effects of the risks, and coming up with clear contingency plans in case of each risk occurring. This process of controlling enabled the project manager to have the control needed to avoid any unpredictable situations that would lead the team to lose track of the project goal to deliver a reliable and compliant healthcare solution.

Table 9.1 : Risk management Table

No	Risk Description	Risk Analysis	Contingency Plan
1	Integration Failure with AI API	High	If Integration does not succeed, or it is considered to be too brittle, fall back into a basic, internal rule-based classification revolving around predefined vital sign boundaries, taking precedence on system stability over complexity of analysis.
2	Data Security Breach (RBAC)	Medium	Close the compromised accounts, lock down network access to the database, and perform a full audit of the relevant logs. Inform stakeholders and trigger a system-wide password reset.
3	Changing Requirements	Medium	if a new feature is requested late in the cycle, defer it formally for future enhancement to protect the timeline and integrity of the current version.

9.4 Conclusion

Chapter 9 established that project management and planning was critical to the successful implementation of the SmartHealth system. The Work Breakdown Structure (WBS) was the key hierarchical map as it identified the overall scope and subdivided it into manageable tasks throughout the five phases, whereas the Gantt Chart was the key tool of time-management because it visualized dependencies and kept the project on its critical path. Moreover, the proactive risk management process played an important role in predicting and eliminating significant threats. Following these well-developed methodologies, the project was accomplished on time, within scope, and with the required quality checks so that it could provide a secure and reliable solution of digital health monitoring.

10 CONCLUSION

10.1 Introduction

This final chapter outlines the full project lifecycle for the SmartHealth system and is a summary of the first, structured using Starting Point and Finishing Point, where the process is described from problem identification to a final solution for a secure, tested, and user-accepted application. The project was able to progress through all stages from detailed requirements Analysis up to System User Acceptance Testing (UAT), and produced a working system that satisfied all objectives that were defined, with the most critical being the integration of the Gemini AI API enabling prescient patient health analysis. In this chapter we will summarize the achievements of this project, discuss openly about the technical and scope related constraints and limitations faced in developing the system and eventually present future works and recommendations for further evolution of the SmartHealth platform.

10.2 Achievement

The SmartHealth project accomplished all three objectives defined in the planning stage. The system is a reliable, efficient and intelligent layer for health-monitoring, with the advantage of overcoming the barriers of paper-based records.

10.2.1 To allow instant and real-time access to health information

The project was able to equip a digital platform that offers real-time and instant access to important health information to both patients and doctors. This was done by the core patient module and the doctor monitoring page that are based on the centralized MySQL database. This success can be proved with references to the Integration Testing (IT-01 and IT-04), which verified that the submission of health data by the patient is saved in the database at the same time and the doctor is informed about the risks with the help of the monitoring page. Additionally, the User Acceptance Testing (UAT) feedback, which was attested that doctors can easily access the recent submission of their patients with their health information, which gives them a clear picture of their health status in real time.

10.2.2 To Provide personalized health recommendations

The primary objective has been accomplished with the successful and reliable incorporation of the Gemini AI API, which offers advanced personalised analytics on one's health. The system is meant to analyse the patients' vital signs along with the symptoms entered and produce a personalised recommendation pertaining to health status and health status markers (e.g. "Good", "Monitoring Advised"). The evidence of achievement is documented in the Integration Testing (IT-02) report which confirms the completion of the workflow: data submission triggers an external AI API call, the response is parsed, and the advice along with the analysis is recorded on the patient's file. Most importantly, UAT Question 02 confirmed the usefulness of the output in which participants stated that the AI advice was relevant and understandable, proving the personalised guidance was functional and effective.

10.2.3 To Develop a Digital Platform for Hospital Patients

The entire objective of this project was to create a fully digital platform available to both patients and the doctors of Hospital Taiping who are assigned to them, removing the need for paper, manual processes, and therefore streamlining efficiency. The implemented SmartHealth system, which included a patient portal designed for entry of health data, and integrated messaging and appointment functionality, achieved this goal. The successful completion of all the testing phases, including functional testing, showed that both patients and the doctors assigned to them could log into their secure portals, thus validating successful completion. The platform's UAT results yielded feedback that confirmed the primary health management digital platform was fully integrated.

10.3 Constraints and Limitation

While the SmartHealth project accomplished the primary objectives, there were some limitations regarding scope and execution. One such limitation was time. More advanced features were unable to be developed in due time. The more complicated modules also had to be shelved for the future, such as real-time video consultations and the integration of wearables. There were also some technical limitations regarding the Gemini AI API integration. The system's dependence on external services meant that performance and availability were out of our control and dictated solely by the API's uptime and response time, something we can't directly manage. The last of these limitations was budget constraints, as we had to develop our project with a single-language interface, and with the solution we had to provide, to a basic cloud environment which did not allow for the scaling that an international multisite hospital deployment would have needed.

10.4 Future Works and Recommendation

The present iteration of SmartHealth can be categorised as a Minimum Viable Product (MVP) and a strong base. These suggested improvements concentrate on continuing to enhance user accessibility, improving the system's diagnostic capabilities, enhancing the automation of data entry, thereby transitioning from a web-based monitoring system to an integrated digital health ecosystem.

10.4.1 Mobile Application Development

The next step on the horizon involves the creation of native mobile apps for both iOS and Android operating systems. Although the existing system works on the web, improving user compliance and real-time engagement, particularly for patients entering their daily health data, is of paramount importance. Such an advancement would make available the functionality of push notifications for real-time communication on new messages from the doctor, health status alerts, and reminders of appointments. This would make web-based applications more user-friendly and increase the chances of patients receiving important information in a timely manner, even when they are not logged into the web application.

10.4.2 AI Model Enhancement

The subsequent step will be improving the systems core cognitive abilities by AI Model Enrichment. This includes moving beyond the former live classification system towards the incorporation of predictive analysis. The new model would be able to analyse patient cumulative data and trends to identify patterns of probable long-term health risks, which includes predicting the likelihood of increased blood pressure within a month. The upgrade will supply physicians with more sophisticated and precise information, elevating the system from merely reactionary to proactive risk forecasting, which would allow the system to assist in timely intervention and future preventive care.

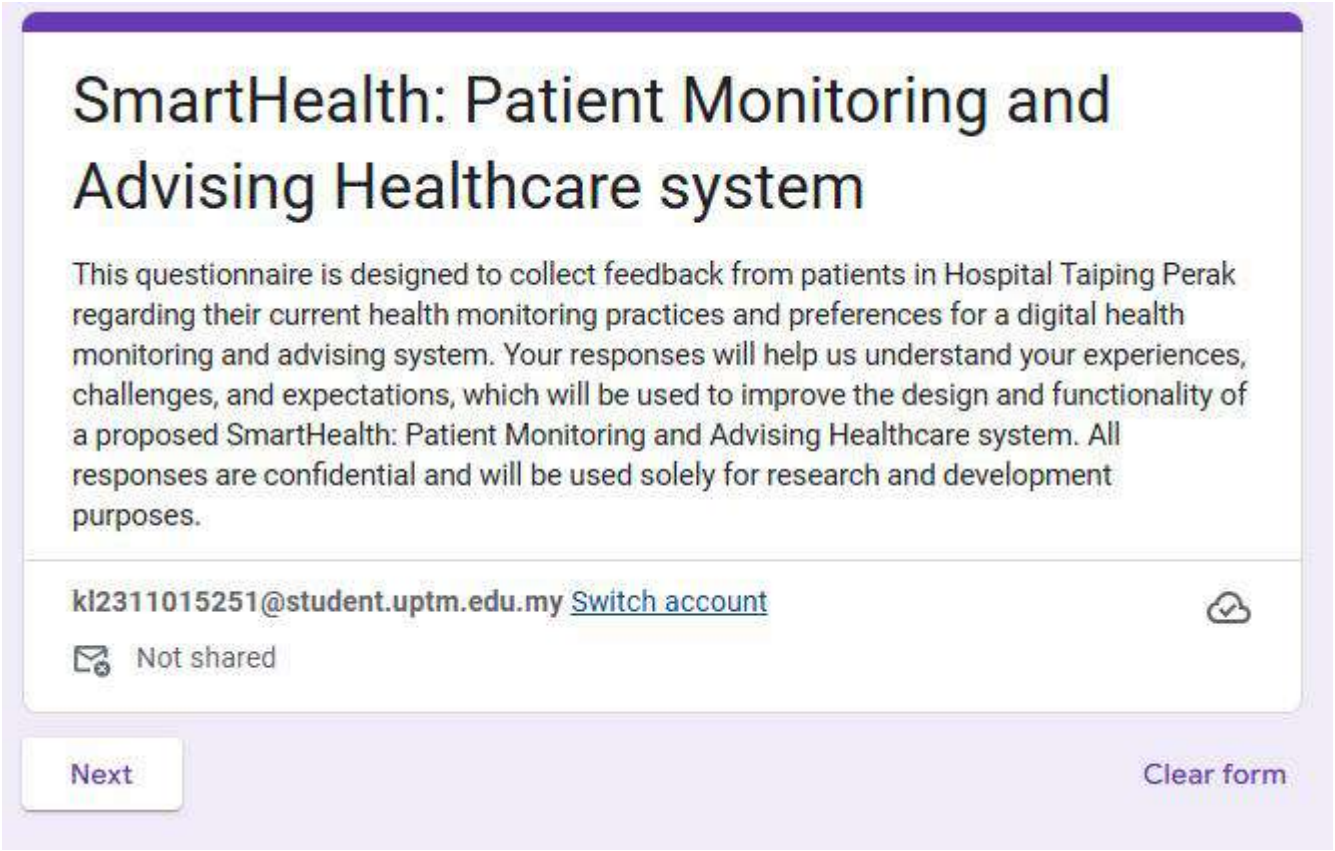
10.4.3 Wearable and IoT Integration

In order to overcome the manual input limitation, the system should be synced with common wearable and IoT devices, like fitness trackers and smartwatches. This connectivity would create a seamless flow of data and vital signs entry would be automated by fetching the data from a patient's own device and transmitting it directly to the SmartHealth database in real-time. This not only minimizes the possibility of human error and fatigue, but also facilitates a vast, more frequent and more accurate dataset to analyse, thus making the patient monitoring effortless for the patient, and exponentially more powerful for the doctor.

10.5 Conclusion

Overall, the SmartHealth project has completed the development of a digital health monitoring platform that meets the objective of providing instant, real-time health and personalised recommendations through the integration of the Gemini AI API. Although the initial time and budget constraints limited the development scope to a web application, the system performed efficiently and securely, meeting the parameters of extensive functional, non-functional, and User Acceptance Testing (UAT). Given the recommendations related to Future Works specifically mobile app development, AI prediction enhancements, and integration of wearables. SmartHealth has the potential to evolve from a basic functioning health monitoring system to a fully integrated proactive digital health system, ensuring the system’s continued relevance and usefulness as a digital health solution.

Appendix A – Requirements Specification Document



SmartHealth: Patient Monitoring and Advising Healthcare system

kl2311015251@student.uptm.edu.my Switch account

Not shared

* Indicates required question

SECTION A

Demographic information

1. What is your gender *

- Male
- Female

2. What is your age group (year) *

- Under 18
- 18 - 30
- 31 - 45
- 46 and above

Back Next Clear form

SECTION B

Health Monitoring and Advising System experience

3. Were you able to record and access your health data without any issues? *

- ◯ Yes, very smoothly
- ◯ It was difficult
- ◯ No, I cannot access it

4. Did the system provide you with useful health advice or recommendations based on your data? *

- ◯ Very useful
- ◯ Not useful at all

5. How effective was the system in providing real-time access to your health information? *

- ◯ Effective
- ◯ Ineffective

Back Next Clear form

SECTION C

System preferences and features

6. How would you rate the overall usability (ease of navigation, clarity of menus, and etc.)? *

- ◯ Excellent
- ◯ Good
- ◯ Fair
- ◯ Poor

7. How satisfied are you with the system's performance (speed, responsiveness, and loading time)? *

	1	2	3	4	5	
Not satisfied	◯	◯	◯	◯	◯	Very satisfied

8. How secure did you feel when entering personal health data into the system? *

- ◯ Very secure
- ◯ Not secure at all

9. How visually appealing and consistent do you find the system's design (User interface)? *

	1	2	3	4	5	
Bad	◯	◯	◯	◯	◯	Good

Back Next Clear form

SECTION D

Feedback and Suggestion

10. Which of the following areas do you feel would most improve the SmartHealth system? *

- [x] Improving the design and ease-of-use (how the site looks and feels
- [ ] Making the AI health recommendations more detailed or personalized
- [ ] Adding more features, such as medication tracking or a diet log.
- [ ] Improving the speed and features of the doctor-patient messaging.
- [ ] The system is good as it is. No major improvements are needed.

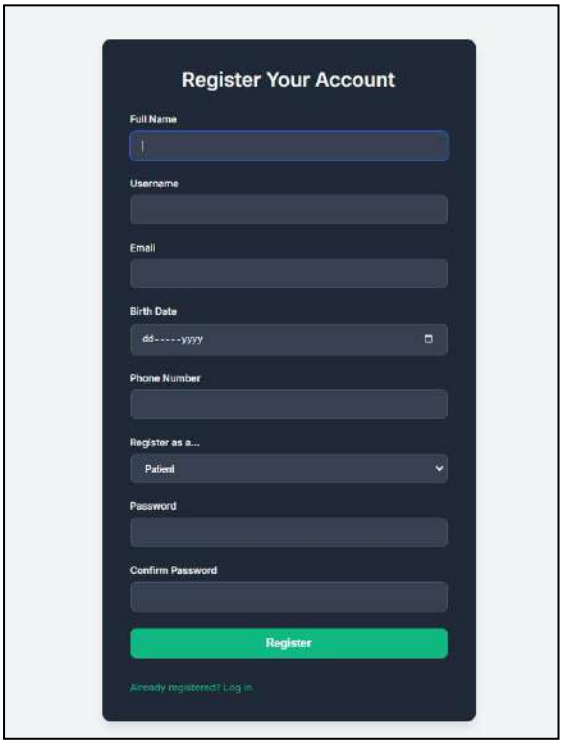
11. How much do you trust the AI-generated health analysis (the dashboard with "Good", "High Concern", etc.)? *

- ( ) I find the analysis very reliable and helpful.
- (•) I find it useful as a general guide, but I would always wait for my doctor's opinion.
- ( ) I don't really pay attention to the AI and only care about the doctor's messages.
- ( ) I do not trust the AI's recommendations.

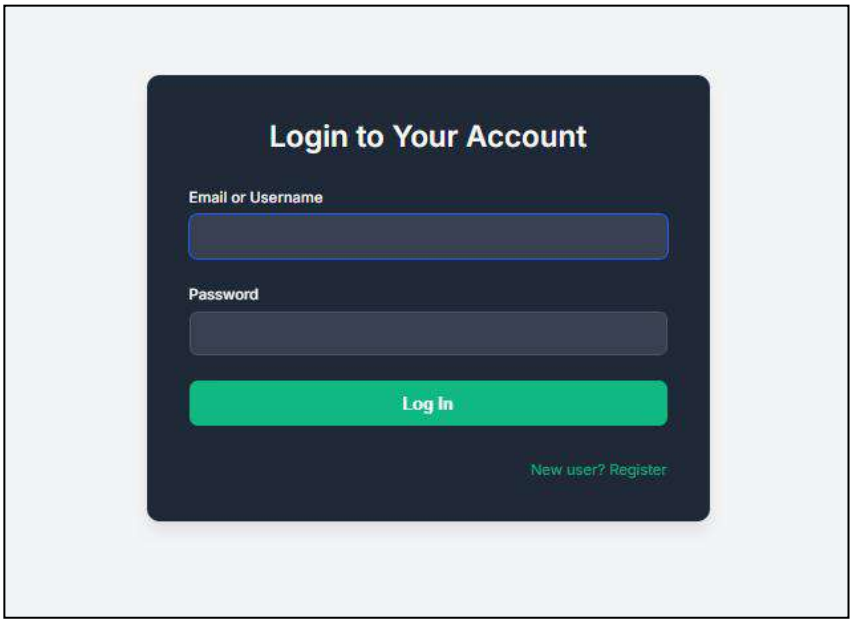
Back Submit Clear form

Appendix B – User Manual

1.0 Register new patient and doctor

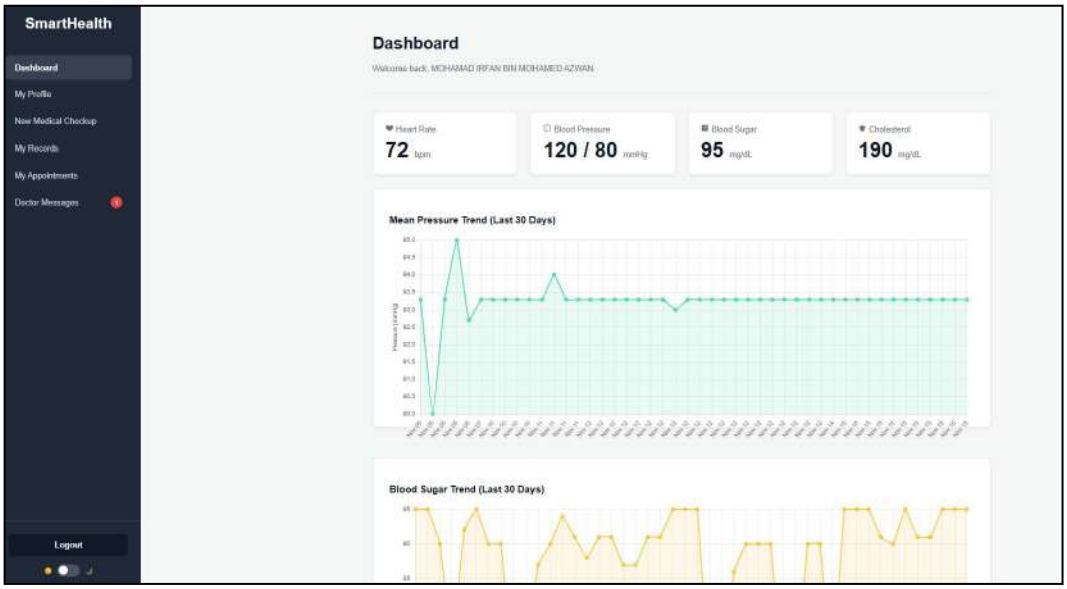


2.0 Login Account

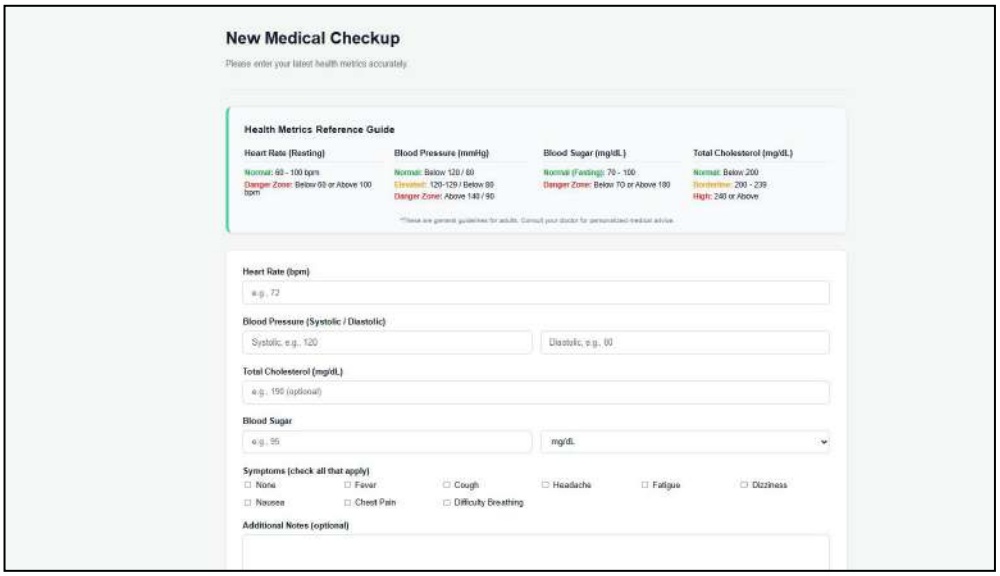


3.0 Patient

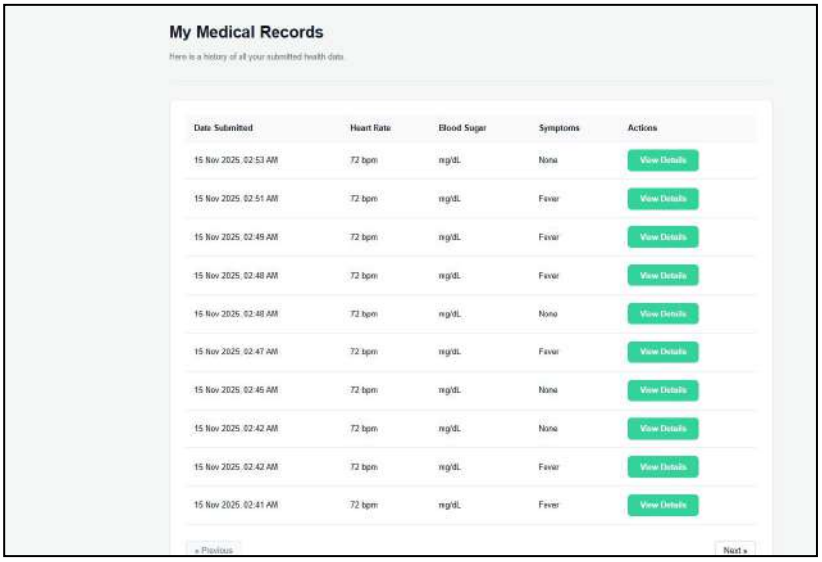
3.1 View Dashboard



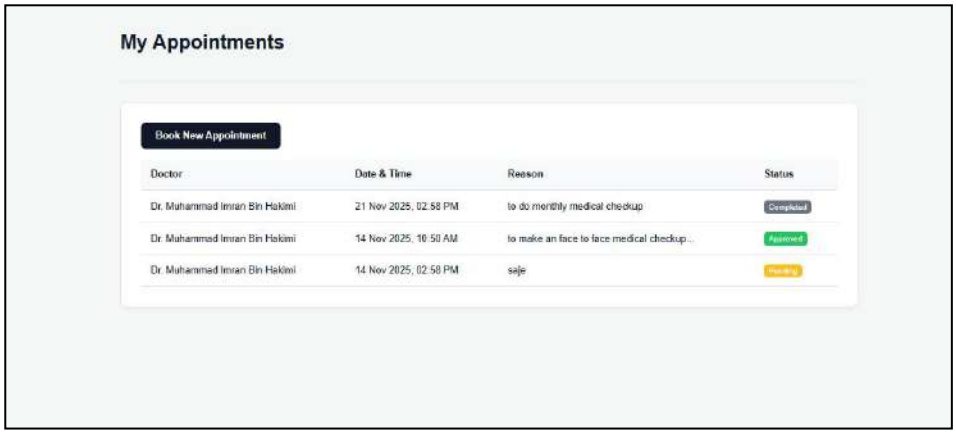
3.2 Insert Data in New Medical Checkup



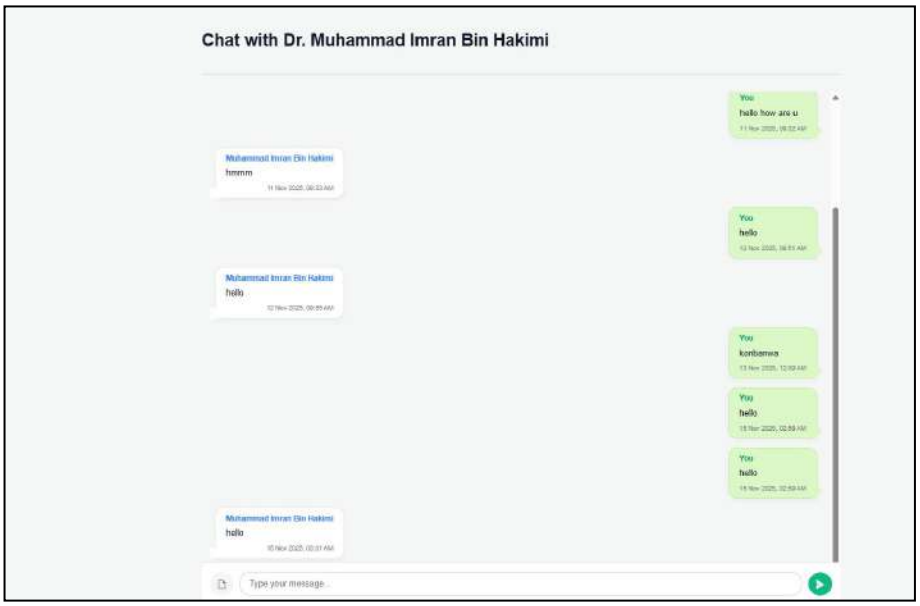
3.3 View Health Records



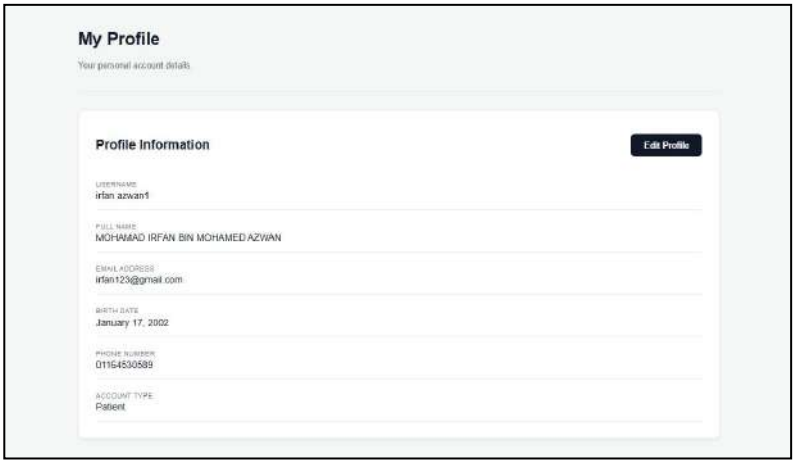
3.4 Book An Appointments



3.5 Message with Doctor

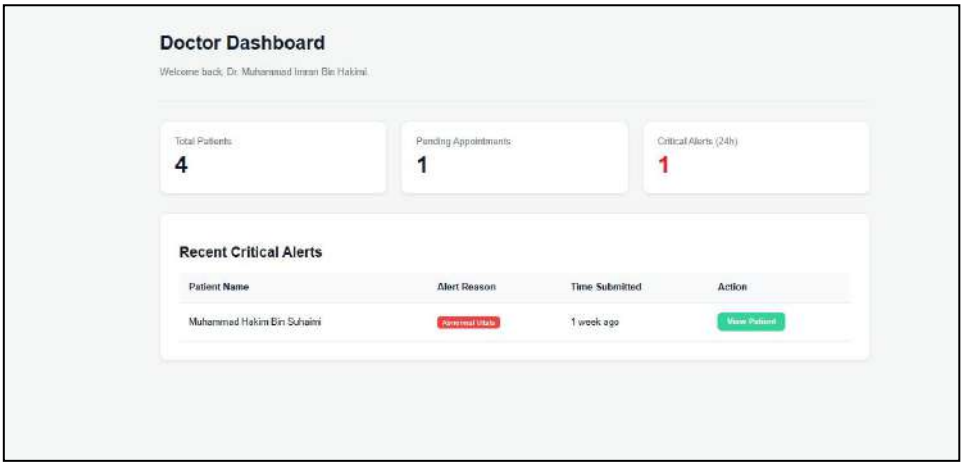


3.6 View Profile

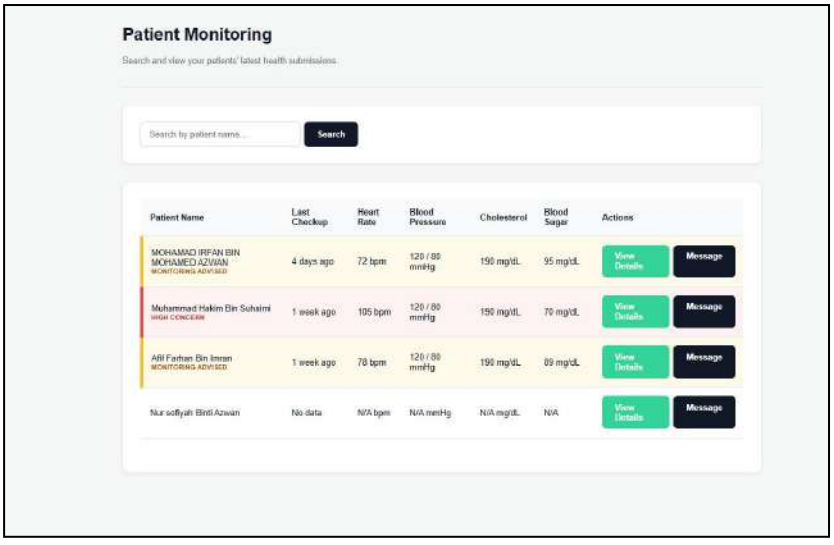


4.0 Doctor

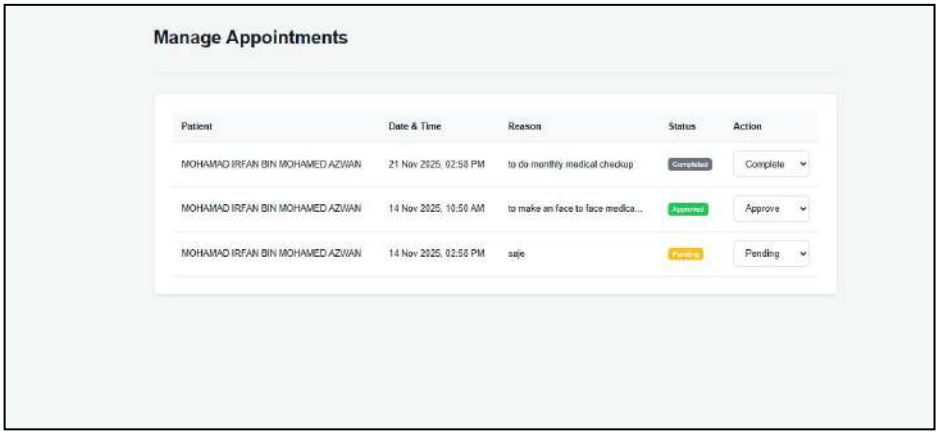
4.1 View Dashboard



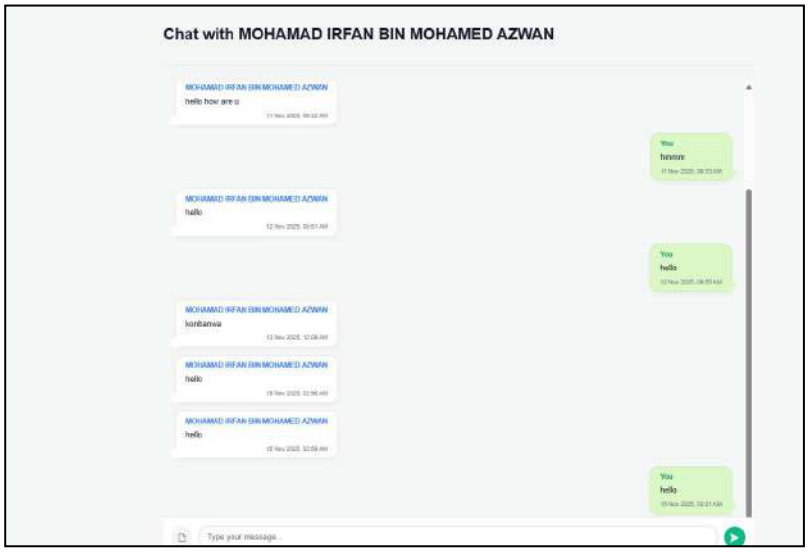
4.2 Monitoring the Patient



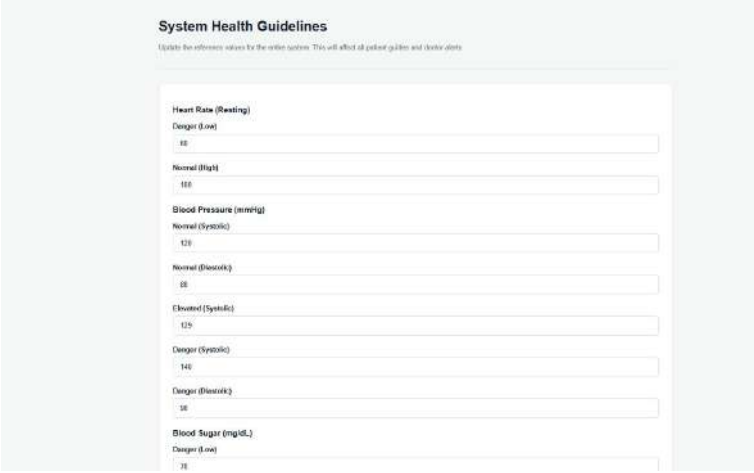
4.3 Manage Appointments



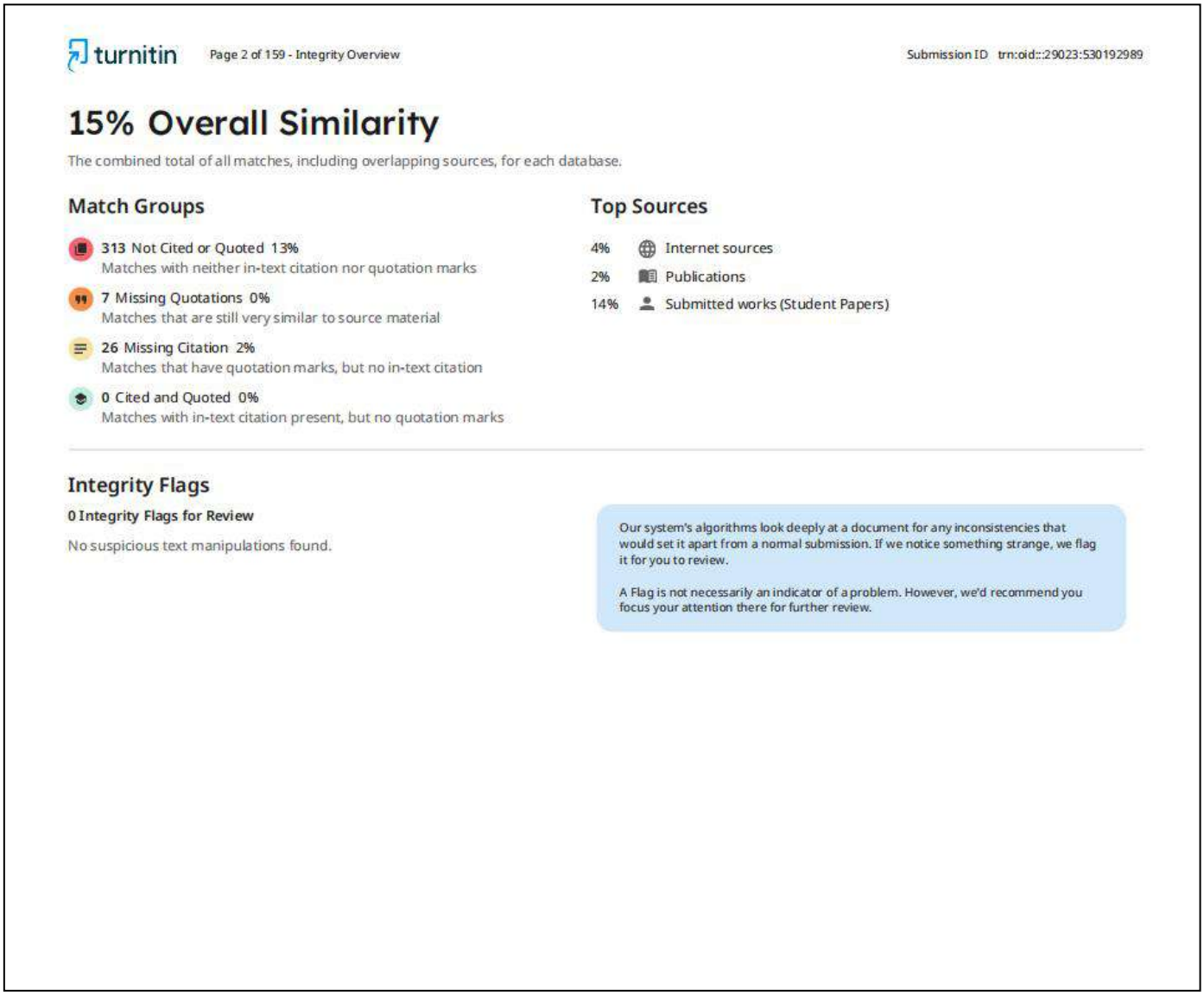
4.4 Message With Patient



4.5 Manage Health Guidelines



Appendix C – Turnitin Result



turnitin Page 2 of 159 - Integrity Overview Submission ID trn:oid:::29023:530192989

15% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Match Groups

- 313 Not Cited or Quoted 13%
  Matches with neither in-text citation nor quotation marks
- 7 Missing Quotations 0%
  Matches that are still very similar to source material
- 26 Missing Citation 2%
  Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

4% Internet sources
2% Publications
14% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Appendix D – AI Result

turnitin Page 2 of 140 - AI Writing Overview Submission ID trn:oid::29023:530192989

*% detected as AI

AI detection includes the possibility of false positives. Although some text in this submission is likely AI generated, scores below the 20% threshold are not surfaced because they have a higher likelihood of false positives.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Disclaimer
Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (it may misidentify writing that is likely AI generated as AI generated and AI paraphrased or likely AI generated and AI paraphrased writing as only AI generated) so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.

Appendix E – Log Book FYP 1

CT204/BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) IN COMPUTER APPLICATION DEVELOPMENT

FACULTY OF COMPUTING & MULTIMEDIA (FCOM)

COMPUTING PROJECT 1

(FYP4013)

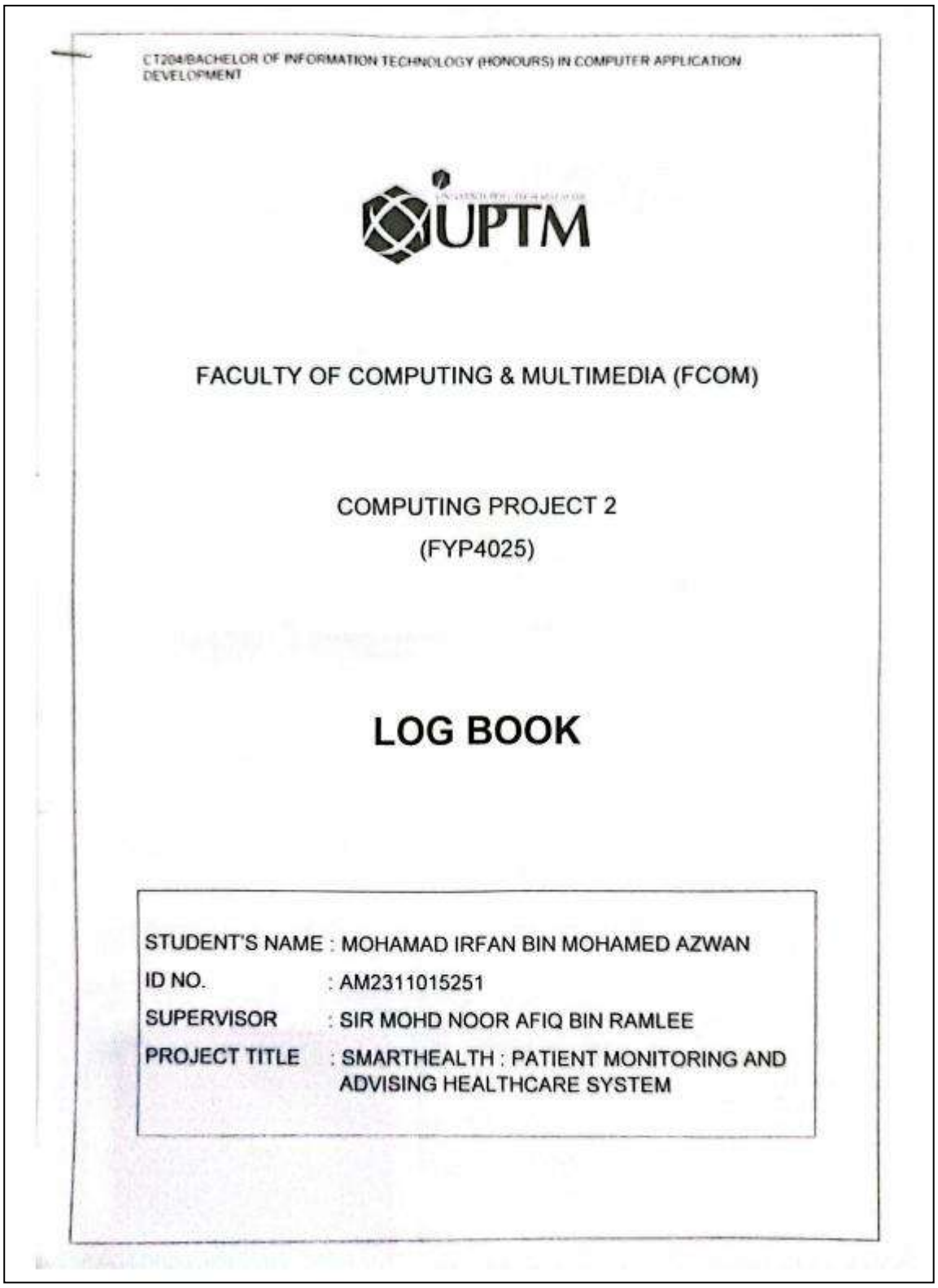
LOG BOOK

STUDENT'S NAME	:MOHAMAD IRFAN BIN MOHAMED AZWAN
ID NO.	:AM2311015251
SUPERVISOR	:SIR MOHD NOOR AFIQ BIN RAMLEE
PROJECT TITLE	:SMARTHEALTH: PATIENT MONITORING AND ADVISING HEALTHCARE SYSTEM

CT204/BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) IN COMPUTER APPLICATION DEVELOPMENT

Date/ Week		Agenda	Next Agenda	Signature (Supervisor / Coordinator)
2/1/2025 (example)	1	Find potential Supervisors	Propose topic and title with SV	MOHD NOOR AFIQ BIN RAMLEE Pensyarah Fakulti Pengkomputeran & Multimedia Universiti Poly Tech Malaysia
29/5/2025	2	Propose title FYP to supervisor	Prepare proposal for being consult with supervisor	MOHD NOOR AFIQ BIN RAMLEE Pensyarah Fakulti Pengkomputeran & Multimedia Universiti Poly Tech Malaysia
3/6/2025	3	Consult proposal with supervisor	Propose fixed problem statement and objectives with supervisor	MOHD NOOR AFIQ BIN RAMLEE Pensyarah Fakulti Pengkomputeran & Multimedia Universiti Poly Tech Malaysia
14/6/2025	4	Consult fixed problem statement and objectives with supervisor	Prepare interview question for consultation with supervisor	MOHD NOOR AFIQ BIN RAMLEE Pensyarah Fakulti Pengkomputeran & Multimedia Universiti Poly Tech Malaysia
18/6/2025	5	Consult interview questionwith supervisor	Consult the report with supervisor	MOHD NOOR AFIQ BIN RAMLEE Pensyarah Fakulti Pengkomputeran & Multimedia Universiti Poly Tech Malaysia
26/6/2025	6	Consult the report with supervisor	Prepare of presentation with examiner and supervisor	MOHD NOOR AFIQ BIN RAMLEE Pensyarah Fakulti Pengkomputeran & Multimedia Universiti Poly Tech Malaysia
30/6/2025	7	Presentation	Report submission	MOHD NOOR AFIQ BIN RAMLEE Pensyarah Fakulti Pengkomputeran & Multimedia Universiti Poly Tech Malaysia

Appendix F – Log Book FYP 2



CT204/BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) IN COMPUTER APPLICATION DEVELOPMENT

UPTM

FACULTY OF COMPUTING & MULTIMEDIA (FCOM)

COMPUTING PROJECT 2

(FYP4025)

LOG BOOK

STUDENT'S NAME	: MOHAMAD IRFAN BIN MOHAMED AZWAN
ID NO.	: AM2311015251
SUPERVISOR	: SIR MOHD NOOR AFIQ BIN RAMLEE
PROJECT TITLE	: SMARTHEALTH : PATIENT MONITORING AND ADVISING HEALTHCARE SYSTEM

CT204/BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) IN COMPUTER APPLICATION DEVELOPMENT

Date/ Week		Agenda	Next Agenda	Signature (Supervisor / Coordinator)
7/8/2025	1	Get feedback from the supervisor about the previous FYP Report	Discuss design and features	DR. MOHD NOOR AFIQ BIN RAMLEE Penyarah Kanan Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia
13/8/2025	2	Discuss design and features	Present wireframe of the project	DR. MOHD NOOR AFIQ BIN RAMLEE Penyarah Kanan Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia
18/8/2025	3	Showing progress	Presenting basic system ui and interaction (login, register, and dashboard page)	DR. MOHD NOOR AFIQ BIN RAMLEE Penyarah Kanan Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia
26/8/2025	4	Presenting basic system ui and interaction (login, register, and dashboard page)	Showing progress (patient and doctor dashboard)	DR. MOHD NOOR AFIQ BIN RAMLEE Penyarah Kanan Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia
4/9/2025	5	Showing progress (patient and doctor dashboard)	Showing progress (medical checkup page)	DR. MOHD NOOR AFIQ BIN RAMLEE Penyarah Kanan Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia
10/9/2025	6	Showing progress (medical checkup page)	Showing progress (develop AI advice for patient using gemini AI)	DR. MOHD NOOR AFIQ BIN RAMLEE Penyarah Kanan Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia
14/9/2025	7	Showing progress (develop AI advice for patient using gemini AI)	Showing progress (message patient and doctor)	DR. MOHD NOOR AFIQ BIN RAMLEE Penyarah Kanan Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia
30/9/2025	8	Showing progress (message patient and doctor)	Showing progress (appointment page)	DR. MOHD NOOR AFIQ BIN RAMLEE Penyarah Kanan Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia
7/10/2025	9	Showing progress (appointment page)	Showing progress (create graph health data patient at dashboard page)	DR. MOHD NOOR AFIQ BIN RAMLEE Penyarah Kanan Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia
16/10/2025	10	Showing progress (create graph health data patient at dashboard page)	Showing progress (monitoring page for doctor page)	DR. MOHD NOOR AFIQ BIN RAMLEE Penyarah Kanan Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia
22/10/2025	11	Showing progress (monitoring page for doctor page)	Showing progress (health records for patient)	DR. MOHD NOOR AFIQ BIN RAMLEE Penyarah Kanan Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia
29/10/2025	12	Showing progress (health records for patient)	Showing progress (Update profressional medical checkup page for printing health data)	DR. MOHD NOOR AFIQ BIN RAMLEE Penyarah Kanan Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia

CT204/BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) IN COMPUTER APPLICATION DEVELOPMENT

5/11/2025	12	Showing progress (Update profressional medical checkup page for printing health data)	Double check all the progress and make sure all functionable	DR. MOHD NOOR AFIQ BIN RAMLEE Pensyarah Kanan Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia
10/11/2025	13	Double check all the progress and make sure all functionable	Final Present at LT5.1	DR. MOHD NOOR AFIQ BIN RAMLEE Pensyarah Kanan Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia
13/11/2025	14	Final Present at LT5.1	Report Submission	DR. MOHD NOOR AFIQ BIN RAMLEE Pensyarah Kanan Fakulti Pengkomputeran & Multimedia Universiti Poly-Tech Malaysia

References

A, A., Dahan, F., Alroobaea, R., Alghamdi, Wael. Y., Mustafa Khaja Mohammed, Hajjej, F., Deema mohammed alsekait, & Raahemifar, K. (2023). A smart IoMT based architecture for E-healthcare patient monitoring system using artificial intelligence algorithms. *Frontiers in Physiology*, *14*. https://doi.org/10.3389/fphys.2023.1125952

Ahmad, F., Soni, M., Parveen, N., Saurabh Kumar Upadhya, & Akbar, S. K. (2025). AI-Driven Personalized Medical Recommendation System. *International Journal of Innovative Research in Computer Science & Technology*, *13*(3), 26–34. https://doi.org/10.55524/ijircst.2025.13.3.5

AI-Driven Personalized Healthcare Recommendations. (2024). *IJIREEICE*, *12*(11). https://doi.org/10.17148/ijireeice.2024.121202

Alotaibi, F., & Almudhi, R. (2023). Application of Agile Methodology in Managing the Healthcare Sector. *IRASD Journal of Management*, *5*(3), 147–160. https://doi.org/10.52131/jom.2023.0503.0114

Claggett, J. (2024). An Infrastructure Framework for Remote Patient Monitoring Interventions and Research (Preprint). *JMIR. Journal of Medical Internet Research/Journal of Medical Internet Research*, *26*, e51234–e51234. https://doi.org/10.2196/51234

draw.io. (2025). *Flowchart Maker & Online Diagram Software*. App.diagrams.net. https://app.diagrams.net/

Enejivik. (2023, October 23). *Visual Studio Code: The Tool That Almost Made Me Quit Programming.* Medium; Stackademic. https://blog.stackademic.com/visual-studio-code-the-tool-that-almost-made-me-quit-programming-bd6905c1a11b

Home. (n.d.). Kyruus Health. https://kyruushealth.com/

Jesuthasan, J., Low, M., & Ong, T. (2022). The Impact of Personalized Human Support on Engagement With Behavioral Intervention Technologies for Employee Mental Health: An Exploratory Retrospective Study. *Frontiers in Digital Health*, *4*. https://doi.org/10.3389/fdgth.2022.846375

Kakria, P., Tripathi, N. K., & Kitipawang, P. (2015). A Real-Time Health Monitoring System for Remote Cardiac Patients Using Smartphone and Wearable Sensors. *International Journal of Telemedicine and Applications*, *2015*, 1–11. https://doi.org/10.1155/2015/373474

Laoyan, S. (2025). *What is agile methodology? (A beginner's guide)*. Asana. https://asana.com/resources/agile-methodology

MyChart | Powered by Epic. (2024). Www.mychart.org. https://www.mychart.org/

PatientsLikeMe. (2019). *PatientsLikeMe*. PatientsLikeMe; PatientsLikeMe. https://www.patientslikeme.com/

Sakibul, M., Adib, A., & Nusrat Jahan Rose, M. (2025). AI-Driven Healthcare Management and Personalized Health Systems: Enhancing Patient Engagement in Digital Health. *International Journal on Science and Technology (IJSAT) IJSAT25048503*, *16*(4). https://www.ijsat.org/papers/2025/4/8503.pdf

Salahuddin, L., Ismail, Z., Hashim, U. R., Ismail, N. H., Raja Ikram, R. R., Abdul Rahim, F., & Hassan, N. H. (2019). Healthcare practitioner behaviours that influence unsafe use of hospital information systems. *Health Informatics Journal*, *26*(1), 420–434. https://doi.org/10.1177/1460458219833090

Tong, Q., Zhou, M., Liu, X., Long, J., Li, L., Pan, X., Gao, H., & Hu, R. (2025). Mobile applications enhance out-of-hospital cardiac arrest outcomes: a systematic review and meta-analysis. *BMC Health Services Research*, *25*(1). https://doi.org/10.1186/s12913-025-12416-2

Uddin, R., & Koo, I. (2024). Real-Time Remote Patient Monitoring: A Review of Biosensors Integrated with Multi-Hop IoT Systems via Cloud Connectivity. *Applied Sciences*, *14*(5), 1876. https://doi.org/10.3390/app14051876

What is Laravel? (n.d.). Laravel News. https://laravel-news.com/what-is-laravel

(2024). ResearchGate. https://www.researchgate.net/publication/369242727_Remote_Patient_Monitoring_Systems_Applications_Architecture_and_Challenges

GitHub

Repository Link: https://github.com/apeklyly

YouTube

Demo Video Link: https://youtu.be/uESc84OtY4E