



Faculty of Computer Science and Information Technology

Hospital Patient Appointment Portal

Muhammad Zahrawi Alghazali Bin Muhammad Zahrudin

Faculty of Computer Science and Information Technology

Bachelor of Software Engineering with Honours

2024/2025

Hospital Patient Appointment Portal

Muhammad Zahrawi Alghazali Bin Muhammad Zahrudin

A thesis submitted In fulfillment of the requirements for the degree of bachelor of Software
Engineering with Honours

Faculty of Computer Science and Information Technology

UNIVERSITI MALAYSIA SARAWAK

2024/2025

DECLARATION

I declare that the work in this thesis was carried out in accordance with the regulations of Universiti Malaysia Sarawak. Except where due acknowledgements have been made, the work is that of the author alone. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.



Signature

Name: Muhammad Zahrawi Alghazali Bin Muhammad Zahrudin

Matric No : 78275

Faculty of Computer Science and Information Technology

Universiti Malaysia Sarawak

Date : 23/6/2025

Acknowledgement

First and foremost, I would like to express my heartfelt gratitude to my Final Year Project supervisor, Dr. Muhammad Asyraf bin Khairuddin, for his unwavering support, insightful guidance, and continuous encouragement throughout the course of this project. His expertise and mentorship have been instrumental in shaping the direction and success of this work, and I am truly grateful for his dedication.

I would also like to extend my sincere appreciation to my examiner, Dr. Dayang Nurfatimah binti Awang Iskandar, for her valuable feedback, constructive critiques, and thoughtful evaluation. Her insights have greatly contributed to the refinement and improvement of this project.

A special thank you goes to myself for the perseverance and commitment shown during this journey. Completing this project has been a challenging yet rewarding experience, and I am proud of the growth it has brought.

To my family and friends, your unwavering support, motivation, and belief in me have been a constant source of strength. Your encouragement has helped me stay focused and determined, and I am deeply thankful for your presence throughout this journey.

Lastly, I would like to acknowledge all individuals and organizations who have contributed directly or indirectly to the completion of this project. Your assistance and insights have been greatly appreciated.

Abstract

The Hospital Patient Appointment Portal (HPAP) is a web-based healthcare management system developed to improve the efficiency of patient appointment scheduling at UNIMAS Medical Center. The system is designed to allow patients with referral letters to bypass the triage process and directly access specific services such as radiology, pharmacy, or X-ray. Built using the MERN stack (MongoDB, Express.js, React.js, Node.js), HPAP integrates real-time appointment booking, centralized patient data access, and role-based dashboards for patients, doctors, and administrators. The system enhances collaboration among departments, reduces administrative workload, and improves patient satisfaction. Usability testing using SUS and UEQ instruments confirmed the system's effectiveness, ease of use, and positive user experience. HPAP represents a scalable solution for modernizing hospital operations and improving healthcare delivery.

Keywords: Hospital Appointment System, MERN Stack, Referral Letter, Healthcare Portal, Patient Management, UNIMAS, MongoDB, React.js, Node.js, Usability Testing

Abstrak

Portal Temujanji Pesakit Hospital (HPAP) ialah sistem pengurusan kesihatan berasaskan web yang dibangunkan untuk meningkatkan kecekapan penjadualan temujanji pesakit di Pusat Kesihatan UNIMAS. Sistem ini direka bentuk untuk membolehkan pesakit yang mempunyai surat rujukan terus mengakses perkhidmatan khusus seperti radiologi, farmasi atau X-ray tanpa perlu melalui proses triage. Dibangunkan menggunakan rangka kerja MERN (MongoDB, Express.js, React.js, Node.js), HPAP menggabungkan tempahan temujanji masa nyata, akses data pesakit berpusat dan papan pemuka berasaskan peranan untuk pesakit, doktor dan pentadbir. Sistem ini meningkatkan kerjasama antara jabatan, mengurangkan beban pentadbiran dan meningkatkan kepuasan pesakit. Ujian kebolegunaan menggunakan instrumen SUS dan UEQ mengesahkan keberkesanan sistem, kemudahan penggunaan dan pengalaman pengguna yang positif. HPAP mewakili penyelesaian berskala untuk memodenkan operasi hospital dan meningkatkan penyampaian perkhidmatan kesihatan.

Kata Kunci: Sistem Temujanji Hospital, MERN Stack, Surat Rujukan, Portal Kesihatan, Pengurusan Pesakit, UNIMAS, MongoDB, React.js, Node.js, Ujian Kebolegunaan

Contents

Chapter 1 : Introduction	1
1.1 Introduction.....	1
1.2 Problem Statement	2
1.3 Scope.....	2
1.4 Aims and objectives	3
1.5 Brief Methodology	3
1.5.1 Iterative Waterfall model Phase.....	4
1.6 Significance of Project	6
1.6 Project Schedule.....	6
1.7 Expected Outcome	7
Chapter 2 : Literature Review	9
2.1 Introduction.....	9
2.2 Overview Objectives.....	9
2.3 Review on Similar Existing System	10
2.3.1 MySejahtera	10
2.3.2 BookDoc	12
2.3.3 MyHealth360	15
2.4 Table comparison between the existing system and proposed system	17
2.5 Review on Software Tools.....	18
2.5.1 Visual Studio Code	18
2.5.2 Atlas MongoDB.....	19
2.6 Chapter Summary	20
Chapter 3 : Methodology	21
3.1 Introduction.....	21

3.2 Methodology	21
3.2.1 Phases in Iterative Waterfall Model.....	22
3.3 Requirement Analysis.....	26
3.3.1 Interview	26
3.4 Software Requirement	27
3.5 Hardware requirement	28
3.6 Use Case Diagram.....	29
3.6.1 Use Case Description.....	30
3.7 Entity Relationship Diagram.....	38
3.8 Wireframe	39
Chapter 4 : Implementation	46
4.1 Overview.....	46
4.2 Installation and Configuration	46
4.2.1 Hardware.....	46
4.2.2 Software	47
4.3 HPAP System Component.....	48
4.3.1 Dashboard	48
4.3.2 Appointment Booking Page.....	49
4.3.3 My appointment.....	52
4.3.4 Add Health Record	53
4.3.5 Health Record	55
4.3.6 My Notifications	56
4.3.7 My Profile.....	57
4.3.8 Admin Dashboard	59
4.3.9 Admin Login and Staff Login.....	60
4.3.10 All Appoinment Admin	62

4.3.11 Add Doctor Admin	63
4.3.12 All Doctor List for Admin	64
4.3.13 Dashboard for Doctor/Staff.....	64
4.3.14 All Appointment for doctor	65
4.3.15 Health Track Record for Doctor	66
4.4 Summary of the chapter	67
Chapter 5 Result and Discussion	69
5.1 Overview	69
5.2 Setting	69
5.3 Participants.....	70
5.4 Data Collection	70
5.5 Questionare	71
5.6 QuantitativeData	71
5.6.1 System Usability Scale (SUS)	72
5.6.2 User Experience Questionnaire (UEQ).....	74
5.7 Discussion.....	75
5.7.1 What Features Do You Like Most in HPAP?	75
5.7.2 Do you have any suggestions for improving the systems?	75
5.7.3 Did You Face Any Challenges While Using HPAP?	76
5.8 Implications.....	77
5.9 Conclusion	78
Chapter 6 : Conclusion and Future Work	79
6.1 Overview	79
6.2 Conclusions.....	79
6.2.1 Conclusion Based on Objective #1	79
6.2.2 Conclusion Based on Objective #2	79

6.2.3 Conclusion Based on Objective #3	80
6.3 Future Work	80
6.3.1 Integration with National Health Systems	80
6.3.2 Expansion of Patient-Centered Features	80
6.3.3 Development of a Mobile Application	81
6.4 Chapter Summary	81
References	82
Appendices	83
Appendix A : USABILITY TESTING METHOD	83
Appendix B : USER MANUAL INSTALLATION KIT	86
General Information	86
System Overview	86
2. System Requirements	86
Hardware Requirements	86
Software Requirements	86
3. Installation Steps	86
Quick Access (Deployed Version)	86
Full Local Installation	87
4. Using the Application	88
5. Troubleshooting	88

List Of Figures

Figure 1-1 Iterative waterfall model (GeeksforGeeks,2024)	4
Figure 2-1 dashboard for MySejahtera	10
Figure 2-2 Health Record page for Mysejahtera application.....	11
Figure 2-3 Appointment page for MySejahtera	12
Figure 2-4 Dashboard for BookDoc Application.....	13
Figure 2-5 Booking Appointment page for BookDoc application.....	13
Figure 2-6 track health record page	14
Figure 2-7 Dashboard for Myhealth application.....	15
Figure 2-8 appointment request page for MyHealth application	16
Figure 2-9 microsoft visual studio code.....	19
Figure 2-10 Atlas MongoDB.....	20
Figure 3-1 Iterative Waterfall model phases.....	22
Figure 3-2 Use Case Diagram for HPAP	30
Figure 3-3 UML Diagram.....	39
Figure 3-4 Wireframe diagram for login the system	40
Figure 3-5 Wireframe for book appointment for patient	40
Figure 3-6 Wireframe for Appointment History for patient	41
Figure 3-7 Wireframe for register account for patient.....	41
Figure 3-8 wireframe for medical record for patient	42
Figure 3-9 wireframe for review appointment for doc/staff	43
Figure 3-10 wireframe for manage schedule for Doc/ staff.....	43
Figure 3-11 Wireframe for View Patient Detail for Doc / Staff.....	44
Figure 3-12 Wireframe for register page for admin.....	45
Figure 4-1 dashboard for patient.....	49
Figure 4-2 appointment page form-1	49

Figure 4-3 appointment page -2	50
Figure 4-4 My Appointment page.....	52
Figure 4-5 Add Health Record.....	53
Figure 4-6 Health Record.....	55
Figure 4-7 my notifications.....	56
Figure 4-8 My Profile	57
Figure 4-9 Admin Dashboard	59
Figure 4-10 Admin Login	60
Figure 4-11 Staff Login	61
Figure 4-12 All Appointment Admin	62
Figure 4-13 Add Doctor Admin Page.....	63
Figure 4-14 Doctor List	64
Figure 4-15 dashboard for staff and doctor.....	64
Figure 4-16 All Appointment for doctors	65
Figure 4-17Health Track Record for Doctor	66
Figure 5-1UEQ Dimension.....	74

List of table

Table 1 Gantt chart form.....	6
Table 2 Comparison between Existing System and Proposed System	17
Table 3 Software Requirement	28
Table 4 Hardware Requirement	28
Table 5 Use Case Description for Register Account	30
Table 6 Use Case Description for Book Appointment	31
Table 7 use case description for Book Appointment	33
Table 8 use case description for View Medical Record	34
Table 9 use case description for staff / doctor manage schedules	35
Table 10 use case description for review appointment.....	36
Table 11 use case description for manage user.....	37
Table 12 SUS Question.....	72
Table 13 SUS Score.....	73

Chapter 1 : Introduction

1.1 Introduction

In the healthcare industry, efficient patient management is critical to providing high-quality care and ensuring patient satisfaction. However, traditional hospital workflows often require all patients, regardless of their needs, to go through the triage process and register all the information detail. Some of the patient they already go through with this process on the other hospital and their just need some specific treatment. This can result negatively impact the patient experience by wasting their time to do the registration process. Patient satisfaction with appointment booking depends on their ability to schedule appointments at convenient times with the appropriate healthcare providers (Zhao et al., 2017).

Hospital Patient Appointment Portal is designed specifically for UNIMAS Medical Center to simplify and optimize the performance for this process. It allows patients with reference letters from other hospitals to skip the triage process and directly access the services they need. The system ensures faster access to the required services while maintaining proper documentation and follow to hospital policies. Most patients and healthcare providers supported the introduction of an appointment system, even though they were satisfied with the existing walk-in system (Al-Haqwi & Al-Shehri, 2024).

This project introduces an integrated scheduling platform that not only optimizes appointment bookings but also allows hospital staff, including doctors and specialists, to access and review patient information seamlessly. The system is designed to enhance collaboration between departments, improve resource allocation, and minimize administrative workload, all while focusing on providing a better patient experience.

With features such as real-time slot availability, centralized patient data access, and robust reporting capabilities, this project aims to modernize hospital operations and create a scalable solution for efficient healthcare delivery. Through innovation and integration, the project aspires to redefine how hospitals manage patient appointments and services.

1.2 Problem Statement

In many hospitals, patients with reference letters from other medical institutions seeking specific treatment, such as radiology, pharmacy, or X-ray, are often required to go through the standard triage and registration process. This creates unnecessary delays, increases administrative workload, and causes inconvenience for both patients and hospital staff. Despite having clear documentation of their medical needs, these patients are unable to access targeted services directly, leading to inefficiencies in resource utilization and patient dissatisfaction.

An optimized process is required to allow such patients to bypass triage and access the needed services efficiently while ensuring proper documentation and follow the hospital protocols. Despite having walk-in system, the patient will save their time to choose their treatment by skip the beginning process through register the information, do the triage and check with the doctor. It will enhance the overall user experience.

1.3 Scope

The project scope for the **Hospital Patient Appointment Portal (HPAP)** is aimed at modernizing and simplifying the management of patient appointments and service access at UNIMAS Medical Center. This project involves the design, development, and implementation of a system tailored to meet the specific needs of the medical community.

The main objective is to improve the appointment booking process and allow patients with reference letters to bypass the triage stage for specific services such as radiology, pharmacy, or X-ray. Additionally, the system will integrate patient information to enable healthcare providers to access and manage data seamlessly, ultimately improving efficiency and the overall patient experience.

A core feature of the system is its appointment booking platform, designed to provide a seamless experience for patients. The system provides real-time updates on available time slots, allowing patients to book appointments at times that work best for them. This helps prevent overbooking and ensures that hospital resources, such as staff and equipment, are used effectively. With its simple and intuitive interface, the system aims to improve patient satisfaction by reducing the complexity of the booking process and making the experience more

efficient. This approach can save time for both patients and healthcare providers, contributing to a smoother healthcare experience overall.

Another significant aspect of the project is the integration of patient information. Doctors, radiologists, pharmacists, and other healthcare providers can access a centralized database containing essential details, including reference letters, previous medical records, and appointment histories. This fosters better collaboration among departments and enables informed decision-making, ultimately improving the quality of care.

1.4 Aims and objectives

There are three objectives that are highlighted in order to achieve this particular goal which is:

- a) To design a process that allow patients with reference letters from other hospitals to directly access specific services, enhancing patient workflow and reducing wait times.
- b) To develop an integrated system that allows doctors and other healthcare providers to easily access and review relevant patient information, including reference letters and prior medical records, to support informed decision-making.
- c) To evaluate and implement a solution that provides patients with real-time visibility of available time slots, enabling convenient appointment booking while optimizing hospital resource utilization.

1.5 Brief Methodology

This project adopts an iterative Waterfall methodology to develop a reliable and effective Healthcare Patient Appointment Platform (HPAP) using the MERN stack—which includes MongoDB, Express.js, React.js, and Node.js. The Waterfall model guides the development through sequential phases such as requirement analysis, system design, implementation, testing, and deployment. Although traditionally linear, iterative feedback loops are incorporated at each stage to ensure the system remains aligned with user requirements and project objectives.

To ensure accessibility, scalability, and ease of use across multiple healthcare facilities, the system will be developed as a web-based application. This model supports multi-platform compatibility and facilitates real-time access for both patients and healthcare providers.

The MERN stack is a widely adopted full-stack JavaScript framework that enables unified development using a single language across both client-side and server-side. It consists of MongoDB for flexible and scalable data storage, Express.js for building robust RESTful APIs, React.js for developing responsive and dynamic user interfaces, and Node.js as the server runtime environment (MongoDB Inc., 2024). This project places particular emphasis on React.js for front-end development and Node.js with Express.js for the back-end, allowing for efficient, modern web application development..

1.5.1 Iterative Waterfall model Phase

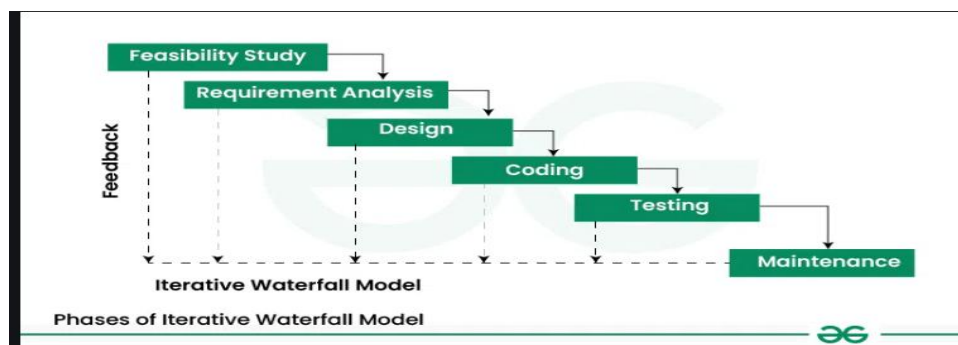


Figure 1-1 Iterative waterfall model (GeeksforGeeks,2024)

The Iterative Waterfall Model is a software development methodology that combines the sequential processes of the traditional Waterfall Model with the flexibility of iterative development. It enables improvement and adjustments to be made at any stage of the development process, rather than waiting until the end of the project (GeeksforGeeks, 2024).

a) Requirement Analysis

This is the step in which needs of stakeholders are identified and information about those needs is accumulated. These comprise both the essential and the extra features, and this would afford the team an opportunity to understand the specific aims and objectives of the project. Feedback from the stakeholders is crucial in this stage since any uncertainty in the requirements is corrected.

b) Design

The System Design phase which seeks to provide a blueprint of the system based on the requirements gathered. This entails the overall architectural planning, data base planning, and planning for the end users' interfaces. There are various models and diagrams that are developed in order to facilitate the whole development process. At this stage overall, reviews and validations are performed in order to make sure that the design is feasible and suitable regarding the requirements.

c) Implementation

The system is developed according to the designs made at the prior phase. It is modular and each of the modules is coded as well as tested afresh the output of the previous one. Unit testing guarantees that each component under tests performs the way it should perform. Implementation challenges may force the design to go back to the drawing board to address the changes that may have arisen.

d) Testing

Integration testing makes sure that the different parts of the system run fall together while system testing checks whether the system meets the initial specifications. If a construction or a notion needs correction, bugs or inconsistencies one circles back to the prior phases in the procedure involving design or implementation. This merely means that the methods used in the implementation of the system will be repeated as an acknowledgment to eliminating all possible errors that may have been made during the entire course of development of the system.

e) Maintenance

This involves actions on the system that include watching out for how well it performs, its stability, the correcting of bugs that arise and /or the addition of new version, improvements on the existing one. In addition, the model's flexibility lets the system be Updating even after it has been implemented to undergo further optimization with user feedback and other fluctuations that might occur.

1.6 Significance of Project

The **Hospital Patient Appointment Portal (HPAP)** is an important step to improving the quality of healthcare delivery at UNIMAS Medical Center. By addressing inefficiencies in traditional processes, this system the various improvement that work to increase the level of satisfaction for both customers and service providers, as well as healthcare productivity.

One of the biggest benefits of this system is how it improves the patient experience. Patients can book their appointments quickly and easily from their house, and those with reference letters can skip the triage process to directly access services like radiology, pharmacy, or X-ray. This saves time and makes the process less stressful for patients.

The system also helps the hospital manage its resources better. By showing available time slots and preventing overbooking, it ensures that staff, equipment, and facilities are used effectively. This helps the hospital provide better care with fewer delays.

Another important feature is the integration of patient information. Doctors and staff can access all the details they need, like reference letters and medical histories, in one place. This makes it easier for different departments to work together and provide faster, more accurate care.

1.6 Project Schedule

This section includes the project schedule for the Final Year Project. Table 1.1 presents the schedule in the form of a Gantt chart. This chart will serve as a roadmap for the development of the proposed system.

Table 1 Gantt chart form

Task Name	Start Date	Finish Date	Duration (Days)
Phase 1: Initiation Phase	07/10/2024	14 /11/2024	40
Brief proposal	07/10/2024	20/10/2024	14
Final proposal	20/10/2024	14 /11/2024	26
Phase 2: Planning Phase	14 /11/2024	13/12/2024	29
Create a detailed project plan	14 /11/2024	21/11/2024	7

Research on existing system	21/11/2024	28/11/2024	7
Conduct a detailed requirements analysis	28/11/2024	08/12/2024	10
Develop a comprehensive project plan	08/12/2024	13/12/2024	5
Phase 3: Design and Architecture Phase	13/12/2024	12/01/2025	25
Design the UIS system architecture	13/12/2024	18/12/2024	5
Develop prototypes	18/12/2024	25/12/2024	7
Define data model	25/12/2024	30/12/2024	5
Conduct a design review and obtain approvals	30/12/2024	02/01/2025	3
Final report and paper	02/01/2025	12/01/2025	10
Phase 4: Development and Testing Phase	03/04/2025	05/05/2025	32
Start coding and development of UIS modules	03/04/2025	11/04/2025	8
Conduct regular code reviews and testing	11/04/2025	19/04/2025	8
Develop user interfaces and system functionality	19/04/2025	27/04/2025	8
Identify and resolves defects and issues	27/04/2025	05/05/2025	8
Phase 5: Optimization and Feedback Phase	05/05/2025	06/06/2025	32
FYP 2 full report and paper	05/05/2025	13/05/2025	8
Preparation for symposium (presentation)	13/05/2025	21/05/2025	8
Present the system to user	21/05/2025	29/05/2025	8
Make necessary adjustments	29/05/2025	06/06/2025	8
Phase 6: Project Closure	06/06/2025	14/06/2025	16
Document project outcomes and lessons learned	06/06/2025	14/06/2025	8
Conduct final project review and evaluation	14/06/2025	22/06/2025	8

1.7 Expected Outcome

Hospital Patient Appointment Portal for UNIMAS Medical Center is expected to bring several positive changes to the hospital's operations and patient experience. Firstly, the system will improve patient satisfaction by making it easier for patients to book appointments at their preferred times. This convenience will reduce waiting times and ensure that patients, especially those with reference letters, can access services like radiology or pharmacy without delays. By skipping the triage process, patients will experience a smoother visit. Furthermore, the system will help the hospital use its resources more efficiently. By providing real-time information

about available slots, it will reduce overbooking and make sure that staff, equipment, and facilities are being used effectively. This will lead to better service delivery and avoid any unnecessary delays. Moreover, key outcome is better communication and access to data. The system will allow doctors and staff to easily access patient information, such as appointment details and medical history. This will ensure that everyone involved in patient care has the information they need, improving the overall quality of care. In conclusion, the expected outcome of this project is a more efficient, convenient, and patient-friendly appointment scheduling process that will benefit both the patients and the hospital staff. It will lead to better healthcare delivery, improved patient experiences, and more effective hospital operations.

Chapter 2 : Literature Review

2.1 Introduction

This chapter focuses on reviewing similar systems already in use and comparing them with the proposed system. The review highlights the features, advantages, and limitations of these existing systems. It also provides an overview of the features and compares the existing systems with the proposed one. Additionally, this chapter examines various software tools to determine which is best suited for the proposed system.

2.2 Overview Objectives

The **Hospital Patient Appointment Portal** is being developed for the **UNIMAS Medical Center** implemented to improve the method of booking and accessing health appointments for patients. This system will be helpful to both patients and the hospital staff to overcome various problems, which they encounter within the vicinity of its applicable area, by making the system easy and comfortable to use. Currently, patients often experience delays and inefficiencies due to manual booking processes and the need to go through unnecessary steps, such as triage, even when they only require specific services like radiology or pharmacy. This project focuses on resolving these issues by offering a centralized and automated platform for appointment scheduling. According to the set system, patients will be able to go to the portal and see availability of services such as doctors and equipment and times and be able to book in at their own convenient time. To enhance the management of patients who present reference letters from other hospitals, experience will be enhanced in terms of the patients by-passing the triage stations to access the particular services. In addition, the project will have smart patient data sharing within different departments and make information available to doctors and staff easily. This integration will enhance communication, reduce errors, and improve the overall quality of care.

In summary, the **Hospital Patient Appointment Portal** is designed to improve the patient experience while enhancing operational efficiency at UNIMAS Medical Center.

2.3 Review on Similar Existing System

This section reviews existing systems similar to the proposed system, focusing on their features. Three systems MySejahtera, BookDoc, and MyHealth360 are evaluated and compared. While these systems share similar features with the proposed system, the key difference lies in their scope. These systems cater to a wide network of panel hospitals, whereas the proposed system is specifically designed to focus exclusively on UNIMAS Medical Center as its primary panel.

2.3.1 MySejahtera

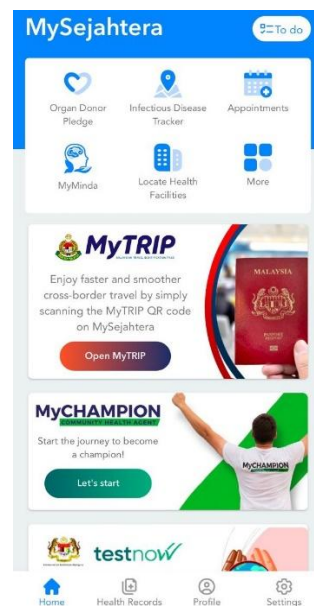


Figure 2-1 dashboard for MySejahtera

MySejahtera is a mobile application developed by the Government of Malaysia to assist in managing the COVID-19 pandemic. It was initially designed for contact tracing, vaccination appointments, and health status reporting. Over time, its scope expanded to include broader health-related functionalities. MySejahtera, while effective in managing pandemic-related needs, faces significant challenges in appointment scheduling and patient record integration. These issues highlight the limitations of a system not initially designed for comprehensive healthcare management.

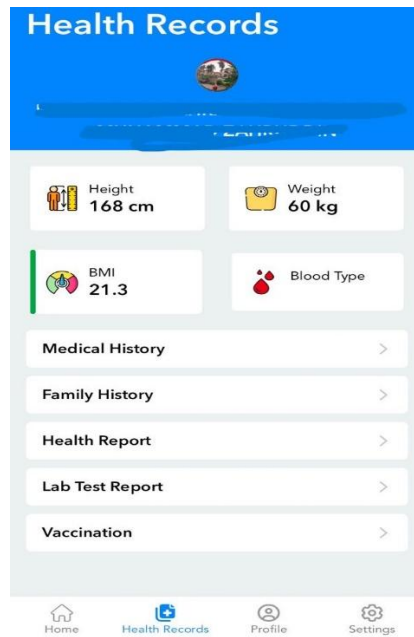


Figure 2-2 Health Record page for Mysejahtera application

Based on the figure 2.2 show that one of key problem lies in the absence of patient record integration. MySejahtera does not support the centralization of medical history across healthcare facilities. Patients cannot access or manage a comprehensive view of their health records, making it challenging to track ongoing treatments or prior medical interactions. Furthermore, the system is not equipped to handle other types of health appointments, such as consultations with general practitioners or specialists. This gap restricts its functionality to pandemic-specific tasks, leaving users without a holistic healthcare management solution.

Figure 2.3 show that detailed of Appointment page, which its inability to facilitate appointments for specific hospital departments, such as radiology, pharmacy, or X-ray services. While the app serves as a convenient tool for managing COVID-19-related tasks like vaccinations and test bookings, it does not cater to broader healthcare needs that require department-specific scheduling. For instance, patients who need to book an appointment for an X-ray or consult a pharmacist must rely on external systems or traditional methods. MySejahtera lacks a system to directly connect users with specialized hospital departments. This gap prevents efficient resource allocation for hospital equipment and staff in these areas. This system does not support on referral letter from other hospital if patient seek to specialize treatment. Furthermore, MySejahtera does not offer functionality to upload or manage reference letters from other healthcare facilities when patients require specific treatments. This

gap in functionality can lead to inefficiencies in patient care, as healthcare providers may not have access to critical medical information, such as past diagnoses or recommendations.

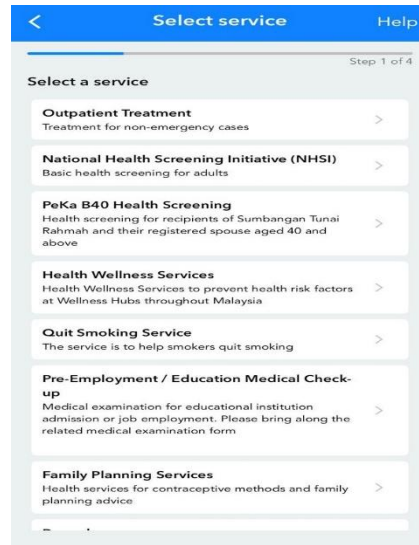


Figure 2-3 Appointment page for MySejahtera

After reviewing MySejahtera, it is a reliable system, and some of its features can serve as references when developing the **HPAP**. However, there are areas where improvements can be made to enhance its functionality. This is understandable, as every developer has their own design and approach. For instance, real-time slot visibility could be introduced to improve appointment scheduling, making it more transparent and efficient for users. Additionally, the integration of patient medical records across healthcare facilities would significantly enhance its usability for broader healthcare management. Since MySejahtera is designed primarily for pandemic-related tasks, its scope is limited and does not fully address general healthcare needs.

2.3.2 BookDoc

BookDoc is a Malaysian-based digital healthcare platform that aims to bridge the gap between patients and healthcare providers through a comprehensive suite of services. It is a user-friendly application designed to simplify healthcare access by enabling users to search for medical professionals, book appointments, and maintain their health through integrated wellness programs. However, **BookDoc**, faces significant challenges in patient record integration. These issues highlight the limitations of a system not initially designed for comprehensive healthcare management.

potential delays to the process. By integrating this functionality, the platform could ensure a more cohesive and efficient treatment process, especially for patients seeking specialized medical services.

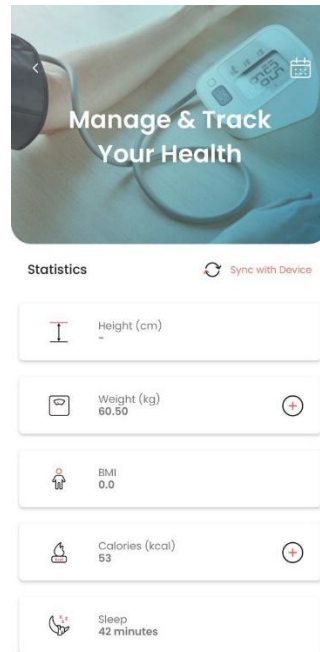


Figure 2-6 track health record page

Based on figure 2.6 show that inability to seamlessly integrate with existing medical records systems. When a patient books an appointment through BookDoc, their medical history, prior treatments, or any critical health information is often not automatically shared with the healthcare provider. This creates a gap in continuity of care, forcing patients to manually provide their medical records or repeat their medical history during consultations. For example, a patient with chronic conditions such as diabetes or hypertension may need to provide updates on their medication history or recent lab results. Without direct integration, this information may not be readily available, increasing the risk of miscommunication or errors in treatment. Additionally, this fragmentation can impact the efficiency of healthcare delivery. Healthcare providers may need to spend additional time collecting and verifying information, which could otherwise be used for patient care.

After reviewing BookDoc, it is a versatile healthcare platform, and some of its features can serve as valuable references when developing the HPAP. However, there are areas where improvements could enhance its functionality and user experience. This is understandable, as

every developer has their own design philosophy and approach. For instance, adding a feature to manage reference letters for patients seeking specific treatments, such as radiology or specialized consultations, would improve its utility for targeted healthcare services. Furthermore, addressing the challenges of integrating patient medical records across different healthcare facilities would significantly enhance its value for comprehensive healthcare management. While BookDoc excels in connecting users with healthcare providers and promoting wellness, its current scope does not fully address the needs of patients requiring more integrated and specialized care.

2.3.3 MyHealth360

MyHealth360 is a comprehensive healthcare platform developed by IHH Healthcare, it designed to meet the modern demands of healthcare management. Designed for patients in Malaysia and other regions served by IHH, the app simplifies access to medical services while promoting a proactive approach to personal health. It integrates seamlessly with IHH's network of hospitals, specialists, and wellness centers. MyHealth360 serves as an excellent example of how technology can bridge the gap between patients and healthcare providers. With its focus on real-time appointment booking and health tracking, it is a valuable tool for managing personal health efficiently. By addressing its current limitations, the app has the potential to set a new benchmark for healthcare platforms in Malaysia.



Figure 2-7 Dashboard for Myhealth application

The screenshot shows a mobile application interface for an appointment request. At the top, there is a header bar with a dropdown arrow and the text "Appointment Request Form". Below this is a card for the doctor: "DR ZULKEFLEE BIN MUHAMMAD", "CARDIOLOGY", "GLENEAGLES HOSPITAL", "KUALA LUMPUR". Below the card, it says "Send your appointment request to book an appointment". There are three input fields: "Preferred Date" with a date picker icon, "Preferred Time" with a time picker showing "08 : 00", and a "Note" text area. At the bottom is a purple button labeled "Send Request".

Figure 2-8 appointment request page for MyHealth application

Based on figure 2.8 show that appointment page does not provide essential features, such as referral letters section. This feature is essential for patients requiring advanced or targeted medical care, as referral letters often serve as the gateway to accessing these services. In its current form, MyHealth360 focuses primarily on facilitating appointment bookings, tracking health records, and providing wellness tools. While these are valuable features, the absence of referral letter management creates a gap in the system, particularly for patients who need optimized access to specialized departments like radiology, cardiology, or oncology. Patients typically rely on referral letters issued by general practitioners or previous healthcare providers to ensure continuity of care and efficient service delivery. Without a feature to handle these referrals, patients may face delays in scheduling appointments or confusion in communicating their medical needs to the receiving specialists.

2.4 Table comparison between the existing system and proposed system

Table 2 Comparison between Existing System and Proposed System

System Features	Existing System			Proposed System
	MySejahtera	BookDoc	MyHealth 360	HPAP
The system able to integrate patient information with panel hospital.	✗	✗	✓	✓
The patient able to access to medical record.	✗	✗	✓	✓
The system able to appointment booking page.	✓	✓	✓	✓
The system able to provide referral letters form to patient upload it.	✗	✗	✗	✓
The system able to provide real-time booking and resource management	✗	✓	✓	✓
Staff or Doctor can review and approve referral letter from patients.	✗	✗	✗	✓
The system able to booking certain treatment or services.	✗	✗	✓	✓

Following a review of three existing systems—MySejahtera, BookDoc, and MyHealth360—Table 1 serves as a reference for developing the proposed system, HPAP (Hospital Patient Appointment Portal). Based on the comparison, it is evident that these existing systems lack certain advanced features that HPAP offers. Unlike the other systems, HPAP is designed to integrate patient information with the Unimas Medical Center, providing seamless access to medical records and enabling staff or doctors to review and approve referral letters uploaded by patients.

The proposed system also addresses gaps found in the existing systems, such as real-time booking and resource management, which ensures that appointment scheduling is more

efficient and transparent. Furthermore, HPAP supports specific treatment or service bookings, such as radiology or pharmacy, which the other systems do not provide.

The first objective is achieved since HPAP integrates patient information with the Unimas Medical Center, allowing doctors to access patient records in real time. Additionally, HPAP meets the second objective by introducing a feature for uploading and managing referral letters, ensuring that patients requiring specialized treatments can seamlessly communicate their needs.

Moreover, HPAP's real-time booking system not only enhances appointment scheduling but also ensures proper resource allocation, such as staff and equipment availability. The system achieves the third objective by offering a user-friendly interface for both patients and healthcare providers, improving the overall patient experience. These enhancements collectively address the limitations of existing systems and position HPAP as a more advanced and specialized healthcare solution.

2.5 Review on Software Tools

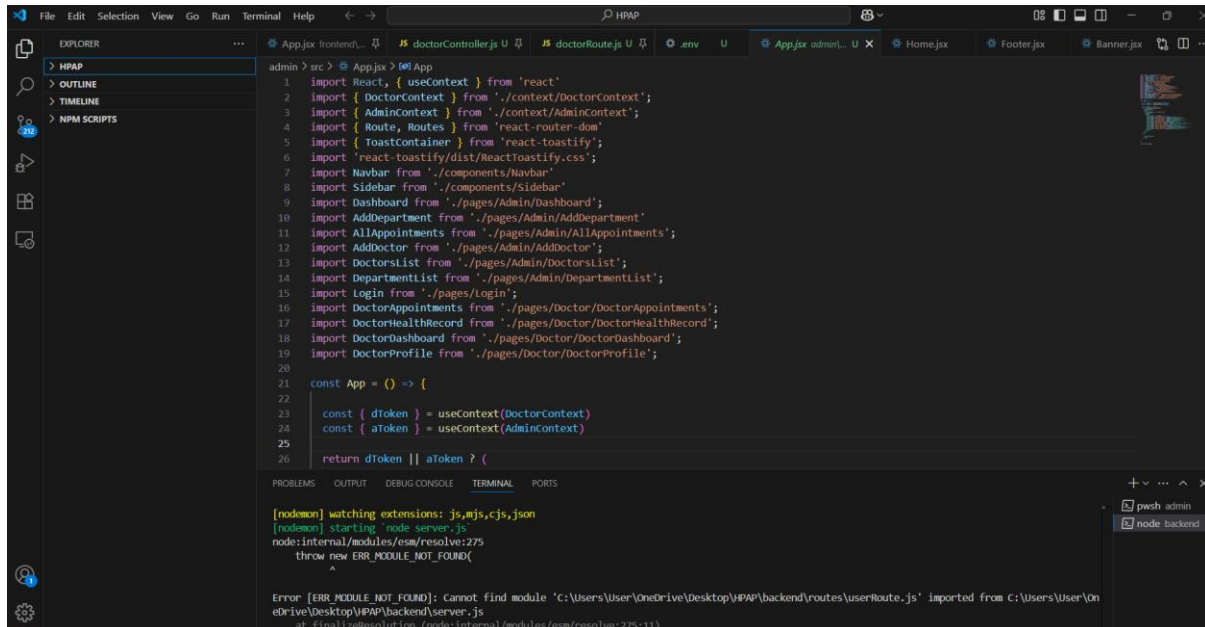
This section will discuss the software requirements for the proposed system. It will cover a few essential software tools, as outlined below.

2.5.1 Visual Studio Code

Microsoft Visual Studio Code (VS Code) is an open-source, efficient, and highly customizable source code editor developed by Microsoft. It supports a wide range of programming languages and frameworks, making it a versatile tool for developers across various fields. VS Code combines essential features such as editing, debugging, and source code management into a single, user-friendly application. With a built-in terminal, syntax highlighting, and IntelliSense—an intelligent code completion and suggestion tool—developers can write, test, and debug code efficiently. VS Code also supports a wide variety of extensions, enabling advanced functionalities such as Git integration, language-specific tools, and enhanced debugging capabilities.

Modern features like Live Share allow developers to collaborate in real time, while built-in Git tools support version control directly within the editor. It also offers support for Docker and

Kubernetes, enabling containerized development. Moreover, developers can tailor their environment with custom themes, extensions, and keybindings to fit their individual workflow preferences.



Security and integration are additional strengths of MongoDB. It includes built-in security features such as authentication, authorization, and encryption (at rest and in transit). Moreover, it integrates well with cloud environments via MongoDB Atlas, a fully managed database-as-a-service (DBaaS) available on AWS, Google Cloud, and Microsoft Azure, offering automatic backups, monitoring, and scaling (Kleppmann, 2017).

“MongoDB is designed for ease of development and scaling. Its document data model maps naturally to objects in application code, making development faster and more intuitive.” (MongoDB, 2024)

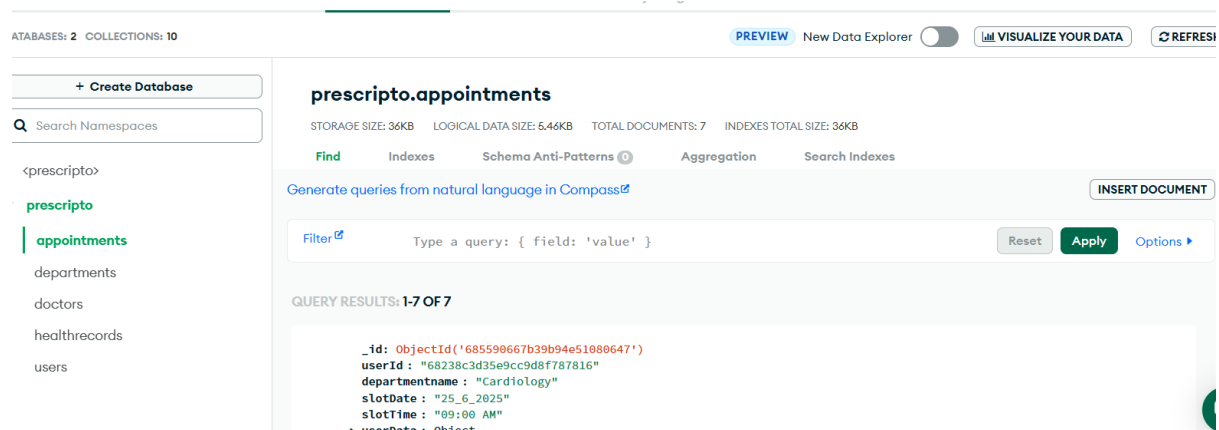


Figure 2-10 Atlas MongoDB

2.6 Chapter Summary

Overall, the purpose of this chapter is to explain the objectives of the Healthcare Management System. Existing systems that are comparable to the proposed system include MySejahtera, BookDoc, and MyHealth360. This chapter summarises a review of these current systems and a comparison of these similar existing systems and the proposed system. As a result, the proposed system may be a superior web-based platform and an upgrade over existing systems.

Chapter 3 : Methodology

3.1 Introduction

This chapter discusses the methodology used for the proposed system, detailing each phase and the corresponding actions required to complete them. It also elaborates on the analysis and design requirements based on the collected data. Additionally, this chapter provides a comprehensive explanation of the process for gathering the necessary information. It includes thorough documentation of the system architecture and design, context diagrams, data flow diagrams, and interface design.

3.2 Methodology

This section will discuss into the methodology introduced in Chapter 1. The system development methodology chosen for this project is the **Iterative Waterfall Model**. The Iterative Waterfall Model is a software development approach that emphasizes a phased process with opportunities for revisiting and refining each phase. This methodology consists of several stages, including requirements analysis, system design, implementation, testing, deployment, and maintenance. Unlike the traditional waterfall model, the iterative approach allows revisiting previous stages based on feedback, ensuring continuous improvement until the final system meets the desired goals.

This iterative process was selected because it facilitates structured development while enabling refinements at each phase. User feedback plays a critical role throughout the process, ensuring that the evolving system aligns with the project goals. Testing is integral to this approach, as it ensures that the system meets the requirements and satisfies both the developers and the end users. Any changes identified during user reviews can be incorporated into subsequent iterations, resulting in a system that is both robust and adaptable to future enhancements.

3.2.1 Phases in Iterative Waterfall Model

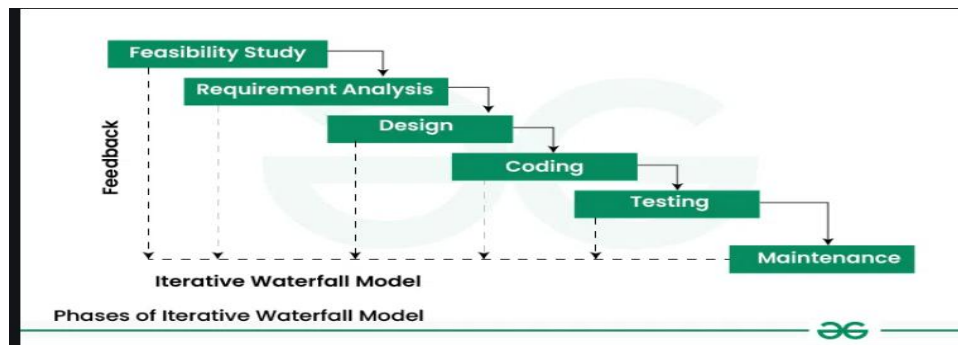


Figure 3-1 Iterative Waterfall model phases

3.2.1.1 Requirement Analysis

In this phase, we identify the current problem and clarify as well as finalize the requirements for the project. The analysis and design phase of the Iterative Waterfall Model involves recognizing the existing issue and thoroughly documenting the project's requirements. In the initial stages, the problem statement for this project is defined, followed by an evaluation of existing systems that are comparable to the proposed solution. A detailed comparison is conducted between the current system and the proposed system, focusing on their features and capabilities. This analysis ensures a clear understanding of the specific requirements, which are then finalized before moving to the design phase. Iteratively, feedback is incorporated to refine and adjust the requirements as necessary throughout the project lifecycle.

3.2.1.2 Design

After completing the analytical requirements in the first phase, the focus shifts to the **Design Phase**, which forms the foundation for developing the proposed system. This phase utilises the finalized requirements into a structured plan that outlines how the system will function and interact with users. The design phase includes detailing the **system requirements**, which cover both software and hardware components essential for the project. This step ensures that all necessary resources and tools are identified and aligned with the project objectives.

Key design elements are described to demonstrate the structure of the proposed system. These include:

1. **System Architecture:** A high-level design that outlines the overall structure of the system, its components, and their interactions, ensuring scalability and performance.

2. **Use Case Diagram:** A graphical representation of the system's functional requirements, showing how various users (actors) interact with the system to achieve specific goals.
3. **Activity Diagram:** A visual workflow of activities within the system, illustrating the sequence of processes and decision points.
4. **Database Design:** A blueprint for the organization of data, detailing tables, relationships, and constraints to ensure efficient data storage and retrieval.

This chapter, **Chapter 3**, elaborates on the design phase in detail, providing a comprehensive explanation of the system's architecture, diagrams, and user interface prototype. It serves as a critical reference for ensuring that the proposed design aligns with the project's goals and requirements. The iterative nature of the waterfall model ensures that the design phase incorporates feedback and revisions as needed, refining the system design to align closely with user expectations and project objectives.

3.2.1.3 Coding

After finalizing the design, the project transitions into the Coding Phase, where the detailed design is transformed into functional software. This phase is critical in the Iterative Waterfall Model as it involves the actual development of the system based on the previously defined and approved requirements and design specifications.

Key Aspects of the Coding Phase (MERN Stack):

1. **Implementation of Modules:**

The system is divided into modular components to facilitate maintainability and scalability. Using the MERN stack—MongoDB, Express.js, React.js, and Node.js—each layer of the application is implemented independently, with React handling the front-end UI, Node.js and Express.js managing the back-end logic, and MongoDB serving as the NoSQL database. This separation of concerns ensures modular development and efficient collaboration between front-end and back-end teams (MongoDB, 2024).

2. **Framework and Language:**

The MERN stack is chosen for its full-stack JavaScript environment, enabling seamless development using a single language across the entire application. React.js provides a

dynamic and efficient front-end interface, while Node.js and Express.js allow for scalable server-side logic. MongoDB offers flexible schema design suited for modern web applications (Kumar, 2022).

3. Adherence to Standards:

Standard coding conventions and best practices are followed across all stack components. This includes consistent naming conventions, modular folder structures, use of middleware in Express, React component organization, and RESTful API design. Clean code and thorough inline documentation improve readability and maintainability.

4. Integration of Components:

React components communicate with the backend through HTTP requests (using tools like Axios or Fetch API). The Express server acts as a bridge between the front-end and MongoDB database. Integration testing ensures that data flows correctly from client to server and is stored or retrieved accurately from MongoDB (Kleppmann, 2017).

5. Iterative Feedback:

Though based on the waterfall model, iterative feedback is incorporated during unit testing, user acceptance testing (UAT), or client review. Adjustments are made in both front-end components and back-end logic to ensure alignment with user needs and design specifications.

This phase brings the design to life through the implementation of full-stack code using the MERN architecture. The consistent use of JavaScript across the stack simplifies development and enables faster, more cohesive application building.

Details of implementation strategies, tools used (e.g., VS Code, npm, MongoDB Atlas), and technical challenges are further discussed in Chapter 4.

3.2.1.4 Maintenance Phase

After the system has been successfully deployed, the **Maintenance Phase** ensures that the system continues to operate efficiently, meets user needs, and adapts to evolving requirements.

This phase is vital for the long-term success of the project and involves making updates, resolving issues, and improving system performance over time.

Key Aspects of the Maintenance Phase:

1. **Bug Fixing:** After deployment, users may encounter unforeseen issues or bugs. These issues are addressed promptly to ensure the system remains functional and reliable.
2. **Performance Optimization:** As the system is used in real-world scenarios, opportunities for optimizing performance are identified. This includes improving response times, reducing memory usage, and enhancing scalability.
3. **Adapting to Changes:** Over time, user needs, business processes, or external factors may change. The system is updated to align with these changes. This may include adding new features, modifying existing functionality, or integrating with other systems.
4. **System Updates:** Regular updates are carried out to maintain system compatibility with the latest versions of frameworks, libraries, and development tools. For example, keeping the application up to date with the latest stable releases of React.js, Node.js, Express.js, and MongoDB ensures improved performance, enhanced security, and access to new features. These updates also help the system remain compatible with modern browsers and server environments (MongoDB, 2024; Node.js Foundation, 2023).
5. **User Support and Feedback:** Ongoing user feedback is collected to identify areas for improvement. Providing technical support ensures that users can effectively utilize the system and resolve any operational challenges they face.
6. **Iterative Improvements:** In the Iterative Waterfall Model, feedback loops allow the maintenance phase to revisit earlier stages if significant changes are required. For instance, if user feedback suggests a major redesign, the project can re-enter the requirements or design phase to accommodate the changes.

This phase is described in **Chapter 5**, where the maintenance process is detailed, including the methodologies, tools, and techniques used to ensure the system remains efficient and reliable over time.

3.3 Requirement Analysis

To gather the requirements for the Hospital Patient Appointment Portal, an interview session was conducted with several staff members and administrators from UNIMAS Medical Centre.

3.3.1 Interview

Mr. Hazman from the IT Department of UNIMAS Medical Centre shared important insights regarding the current state and future direction of their management system. He emphasized that, at present, there have been no specific requests to enhance the existing system, as the current modules implemented at the medical centre are deemed complete and functional. These modules effectively address the needs of the medical centre's operations, ensuring smooth management processes.

However, Mr. Hazman highlighted that the team's focus is now shifting toward the development of a new system aimed at improving patient convenience and accessibility. This new initiative involves creating a platform that allows patients to book medical appointments directly from their homes. By enabling online booking, the system aims to efficiently the scheduling process and eliminate the need for patients to visit the medical centre solely for appointment arrangements. This is particularly beneficial for enhancing patient satisfaction and reducing unnecessary physical visits.

Additionally, Mr. Hazman underscored the importance of designing a system that minimizes patient waiting time and enhances the overall treatment journey. The goal is to enable patients to proceed directly to the required treatment upon their arrival at the medical centre, using referral letters as an entry point. This approach seeks to bypass traditional processes such as triage and registration, which can often be time-consuming. By skipping these steps, the system aims to optimize efficiency and ensure that patients receive the care they need without unnecessary delays. This vision reflects a commitment to leveraging technology to improve the patient experience. By focusing on patient-centered solutions, UNIMAS Medical Centre aims to create a seamless healthcare journey that prioritizes both convenience and timely treatment.

An interview session with Dr. Dessmon from UNIMAS Medical Center was conducted to gather insights into the current appointment management process and identify requirements for the proposed Hospital Patient Appointment Portal. Dr. Dessmon explained that follow-up appointments are usually scheduled immediately after consultations, with the patient's availability being taken into account. However, the current system faces challenges when referring patients to specialists or external facilities, as the availability of these resources is often unclear. This leads to the need for follow-up calls, which can be complicated by difficulties in reaching some patients.

For patients with referral letters, Dr. Dessmon highlighted that the letters are first reviewed to assess whether further action is required. In most cases, even with a referral letter, patients must still go through consultation and diagnosis before proceeding to treatment. To

address these issues, the proposed system should include features that allow real-time appointment booking with specialists and external facilities, minimizing delays and reducing the need for follow-up calls. Additionally, the system should enable patients with referral letters to directly proceed to their specific treatment, bypassing unnecessary steps such as repeated consultations or triage, thus saving time for both patients and staff.

An interview session was conducted with Madam Rogayah, a staff nurse at PKP UNIMAS, to gather insights into the current appointment workflow and identify requirements for the proposed Hospital Patient Appointment Portal. Madam Rogayah explained that nurses play a key role in booking appointments for patients based on doctors' instructions. Currently, appointment details such as the section, date, and time are manually recorded on the patient's medical history card or a piece of paper. When handling patients with referral letters, the letters are reviewed to ensure all necessary details are clear and complete. However, in most cases, doctors still perform a diagnosis before determining the next steps for the referral. If sections or facilities are unavailable, patients are informed, and nurses follow up with phone calls once availability is confirmed.

To address these challenges, Madam Rogayah emphasized the need for a system that provides real-time updates on section or facility availability. This feature would enable nurses to make appointments more efficiently and reduce delays. She also highlighted the importance of automated reminders for upcoming appointments, which would help minimize missed appointments by notifying patients of their scheduled dates and times. The proposed system should efficiently the appointment management process by enabling nurses to book appointments directly through a centralized platform with real-time availability tracking. Additionally, it should facilitate the handling of referral letters, allowing patients to proceed directly to the required treatment without unnecessary steps. These insights reinforce the importance of an efficient and automated system to enhance the workflow and improve the overall patient experience.

3.4 Software Requirement

The software requirements for the development of the HPAP (Hospital Patient Appointment Portal) are outlined in Table 3.1. The development of HPAP requires Windows 10 as the operating system, Visual Studio as the integrated development environment (IDE), and SQL Server for database management. C# serves as the primary programming language for backend development, while Razor syntax is used for dynamic content rendering. HTML and CSS are employed for structuring and styling web content, and JavaScript is used to create interactive elements. These tools are utilized to ensure efficient development and maintenance of the HPAP.

Table 3 Software Requirement

Software Requirement	Details
Operating System	Windows 10
Development Software	Visual Studio
Scripting Languages	HTML, CSS, JavaScript,Razor Syntax
Programming Language	C#
Database	SQL Server

3.5 Hardware requirement

Table 3.2 shows the hardware requirements that are necessary for the development of HPAP. The hardware requirements for HPAP development include a laptop capable of running Visual Studio and SQL Server efficiently. It should be equipped with an Intel Core i7-9750H CPU @ 2.60GHz processor, 24 GB of RAM, and 1 TB of storage. These specifications ensure smooth performance and sample storage capacity for development and testing tasks.

Table 4 Hardware Requirement

Hardware Component	Specification
Processor	Intel Core i7-9750H @ 2.60GHz
RAM	24 GB
Storage	1TB

3.6 Use Case Diagram

Figure 3.2 illustrates the use case diagram for the HPAP (Hospital Patient Appointment Portal). Patients are able to register an account, book appointments, and view their medical records. The system allows patients to schedule appointments with their preferred doctors or for specific treatments, ensuring a smooth booking process. Patients can also upload referral letters when necessary, which can be reviewed by healthcare staff for further actions. Additionally, patients are provided with notifications to remind them of upcoming appointments or any changes in their schedules.

The system offers healthcare staff the ability to review and approve patient appointments, ensuring efficient management of their schedules. Doctors and staff can also view patients' details, including their medical history and appointment records, which helps them provide better and more personalized care. The portal also enables staff to manage their daily schedules, ensuring proper allocation of time and resources, and avoiding overbooking.

Administrative users can manage user accounts, including patients and staff, to ensure appropriate access levels and system security. Administrators can also generate detailed reports on various aspects of the system, such as appointment statistics, patient demographics, and resource utilization, which aid in performance evaluation and decision-making. Furthermore, the system includes features for monitoring its performance and activities, ensuring smooth operation, security, and reliability.

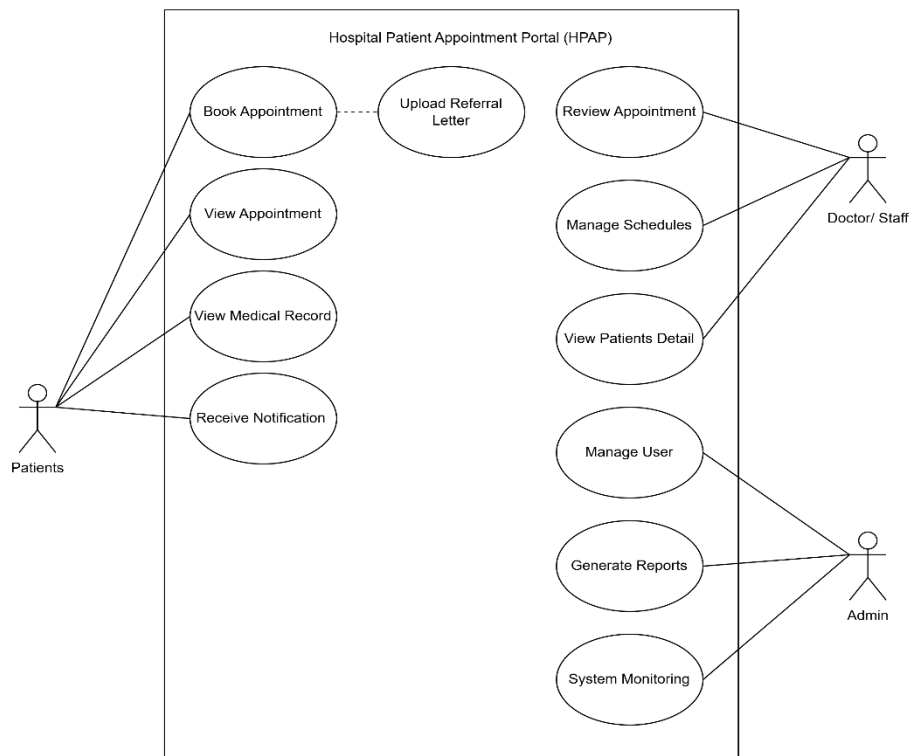


Figure 3-2 Use Case Diagram for HPAP

3.6.1 Use Case Description

The use case description details the interactions and steps for achieving specific functionalities within the system for patients, staff/doctors, and administrators, as shown in Table 5 – Table 11.

Table 5 Use Case Description for Register Account

No	UC1
Use Case Name	Register Account
Actor	Patients
Description	Patients need to register for an account to access the system features
Precondition(s)	1. Patient has accessed the system via a web browser with a stable internet connection. 2. Patient must not already have an account.
Postcondition(s)	A new account is created, allowing the patient to log in and access the system.

Basic Path	1. Patient clicks the “Register” button on the homepage. 2. The system navigates the patient to the registration form page. 3. Patient fills out personal information such as name, IC/passport number, email, phone number, and address. 4. Patient creates a secure password and confirms it. 5. Patient clicks the “Submit” button. 6. The system validates the information, creates an account, and displays a confirmation message.	
Alternative Path	N/A	
Exceptions	1a	Patient loses internet connection during the registration process.
		1a1 The system displays an error message indicating loss of connection.
		1a2 Patient resolves the connection issue and reloads the page to continue.
	2a	Patient already has an account.
		2a1 The system displays an error message indicating the account already exists.
		2a2 Patient navigates to the login page and logs into their existing account.
	2b	Patient submits an incomplete form or invalid data.
		2b1 The system displays an error message highlighting the missing or invalid fields.
		2b2 Patient corrects the form and submits it again.

Table 6 Use Case Description for Book Appointment

No	UC2
Use Case Name	Book Appointment
Actor	Patients

Description	Patients can book appointments with doctors or for specific treatments through the portal. The system provides available time slots and appointment details.		
Precondition(s)	1. Patient has registered and logged into the system. 2. The system has the doctor's schedule and available appointment slots configured.		
Postcondition(s)	The appointment is successfully booked, and the patient receives confirmation with the appointment details.		
Basic Path	1. Patient logs into the system. 2. Patient navigates to the "Book Appointment" section. 3. Patient selects the desired doctor or treatment category. 4. The system displays the available time slots for the selected option. 5. Patient chooses a date and time slot from the list of available options. 6. Patient provides any additional information required (e.g., reason for the appointment, symptoms). 7. Patient clicks the "Book Appointment" button. 8. The system validates the information and reserves the slot.		
Alternative Path	5a	No time slots are available for the selected doctor or treatment.	
		5a1	The system informs the patient that no slots are available and suggests checking another doctor or treatment.
	7a	Patient cancels the process before completing the booking.	
		7a1	The system returns the patient to the previous screen without saving the appointment.
Exceptions	3a	Patient is not logged into the system.	
		3a1	The system redirects the patient to the login page.
	5b	System encounters an error retrieving the schedule.	
		5b1	The system displays an error message and asks the patient to try again later.

	7b	System encounters an issue while reserving the appointment slot (e.g., server error).	
	7b1		The system displays an error message and asks the patient to retry.

Table 7 use case description for Book Appointment

No	UC3		
Use Case Name	View Appointment		
Actor	Patients		
Description	Patients can view the details of their upcoming and past appointments through the system. This includes appointment date, time, doctor, and location.		
Precondition(s)	1. Patient has registered and logged into the system. 2. Patient has at least one scheduled or past appointment in the system.		
Postcondition(s)	The system successfully displays the list of appointments and their details.		
Basic Path	1. Patient logs into the system. 2. Patient navigates to the "View Appointment" section. 3. The system retrieves the patient's appointment history and upcoming appointments. 4. The system displays the list of appointments, including date, time, doctor, location, and status (e.g., upcoming, completed). 5. Patient clicks on a specific appointment to view detailed information, such as notes or instructions. 6. Patient exits the section or navigates to another page.		
Alternative Path	6a	No appointments are found for the patient.	
		6a1	The system displays a message stating "No appointments found."
		6a2	Patient is provided with an option to book a new appointment.

Exceptions	3a	Patient is not logged into the system.	
		3a1	The system redirects the patient to the login page.
	3b	System encounters an error retrieving appointment data.	
		3b1	The system displays an error message and suggests the patient try again later.

Table 8 use case description for View Medical Record

No	UC4		
Use Case Name	View Medical Record		
Actor	Patients		
Description	Patients can view their medical records, including treatment history, test results, prescriptions, and other related information.		
Precondition(s)	1. Patient has registered and logged into the system. 2. Patient's medical records are available in the system.		
Postcondition(s)	The system successfully displays the patient's medical records.		
Basic Path	1. Patient logs into the system. 2. Patient navigates to the "View Medical Record" section. 3. The system retrieves the patient's medical records from the database. 4. The system displays a list of medical records categorized by date, type (e.g., lab reports, prescriptions), or doctor. 5. Patient selects a specific record to view detailed information. 6. Patient exits the section or navigates to another page.		
Alternative Path	2a	No medical records are found for the patient.	
		2a1	The system displays a message stating "No medical records found."
		2a2	Patient is informed to contact the hospital administrator for further assistance.

Exceptions	3a	Patient is not logged into the system.	
		3a1	The system redirects the patient to the login page.
	3b	System encounters an error retrieving appointment data.	
		3b1	The system displays an error message and suggests the patient try again later.

Table 9 use case description for staff / doctor manage schedules

No	UC5		
Use Case Name	Manage Schedules		
Actor	Doctor/Staff		
Description	Doctor or staff can manage schedules by creating, updating, or deleting appointment slots for patients.		
Precondition(s)	1. The doctor or staff has logged into the system. 2. The system has access to the database of existing schedules.		
Postcondition(s)	The updated schedule is saved in the system and becomes available for patients to book.		
Basic Path	1. Doctor/Staff logs into the system. 2. Doctor/Staff navigates to the "Manage Schedules" section. 3. The system displays the current schedule with available time slots. 4. Doctor/Staff chooses to create a new schedule, update an existing one, or delete an existing schedule. 5. For new schedules, the doctor/staff specifies the date, time, duration, and type of appointment (e.g., consultation, follow-up). 6. Doctor/Staff confirms the changes. 7. The system saves the changes and updates the schedule database. 8. Doctor/Staff exits the section or navigates to another page.		
Alternative Path	4a	Doctor/Staff decides to cancel changes.	

		4a1	The system discards any unsaved changes and retains the previous schedule.
	8a	The selected time slot conflicts with an existing appointment.	
		8a1	The system notifies the doctor/staff about the conflict and suggests alternative time slots.
	Exceptions	4a	System fails to retrieve the current schedule due to a database error.
			4a1 The system displays an error message and suggests trying again later.
		4b	System fails to save the updated schedule due to technical issues.
			4b1 The system displays an error message and retains the previously saved schedule.

Table 10 use case description for review appointment

No	UC6
Use Case Name	Review Appointment
Actor	Doctor/Staff
Description	Doctor or staff can review the details of patient appointments, including date, time, type of appointment, and patient details.
Precondition(s)	1. The doctor or staff has logged into the system. 2. The system has access to the database of scheduled appointments.
Postcondition(s)	The appointment details are successfully retrieved and reviewed by the doctor or staff.
Basic Path	1. Doctor/Staff logs into the system. 2. Doctor/Staff navigates to the "Appointments" section. 3. The system displays a list of all upcoming and past appointments for the doctor or staff. 4. Doctor/Staff selects a specific appointment to review.

	5. The system displays detailed information about the selected appointment, such as patient name, appointment type, and notes. 6. Doctor/Staff reviews the information and optionally adds or edits notes regarding the appointment. 7. Doctor/Staff exits the page or navigates to another section.	
Alternative Path	3a	Doctor/Staff searches for a specific appointment using a search or filter feature (e.g., by patient name or date).
	3a1	The system filters and displays matching results based on the search criteria.
Exceptions	4a	System fails to retrieve the current schedule due to a database error.
	4a1	The system displays an error message and suggests trying again later.
	5a	Appointment details fail to load due to a system error.
	5a1	The system displays an error message and prompts the doctor/staff to retry or contact technical support.

Table 11 use case description for manage user

No	UC7
Use Case Name	Manage User
Actor	Admin
Description	Admin can manage user accounts, including registering new staff or doctors, updating user details, and deactivating accounts.
Precondition(s)	1. Admin has logged into the system. 2. Admin has the necessary permissions to manage users.
Postcondition(s)	New staff or doctor accounts are successfully registered, updated, or deactivated as per the admin's action.
Basic Path	1. Admin logs into the system. 2. Admin navigates to the "Manage Users" section. 3. The system displays a list of existing users (staff or doctors).

		4. Admin clicks on "Register New User." 5. The system displays a registration form. 6. Admin fills in the required details, such as name, email, role (staff/doctor), and contact information. 7. Admin clicks the "Submit" button to create the account. 8. The system confirms the successful creation of the new user account. 9. Admin can also edit or deactivate existing users from the list if needed.
Alternative Path	4a	Admin searches for an existing user by name or role to edit their details.
		4a1 The system filters and displays matching results based on the search criteria.
		4a2 Admin selects a user and updates their information.
		4a3 The system saves the updated details and confirms the changes.
	4b	Admin clicks on "Deactivate User" to remove a user's access.
		4b1 The system asks for confirmation before deactivating the account.
Exceptions	6a	Admin enters incomplete or invalid details in the registration form.
		6a1 The system displays an error message, prompting the admin to correct the information before resubmitting.
	7a	System fails to save the new user account due to a database error.
		7a1 The system displays an error message and suggests trying again later or contacting technical support.

3.7 Entity Relationship Diagram

The Entity Relationship Diagram (ERD) shown below outlines the database structure for the Appointment Management System. It features primary entities like Patient, Appointment, and

Staff, along with their respective attributes and relationships. The ERD offers a comprehensive overview of how the data will be structured and linked within the system.

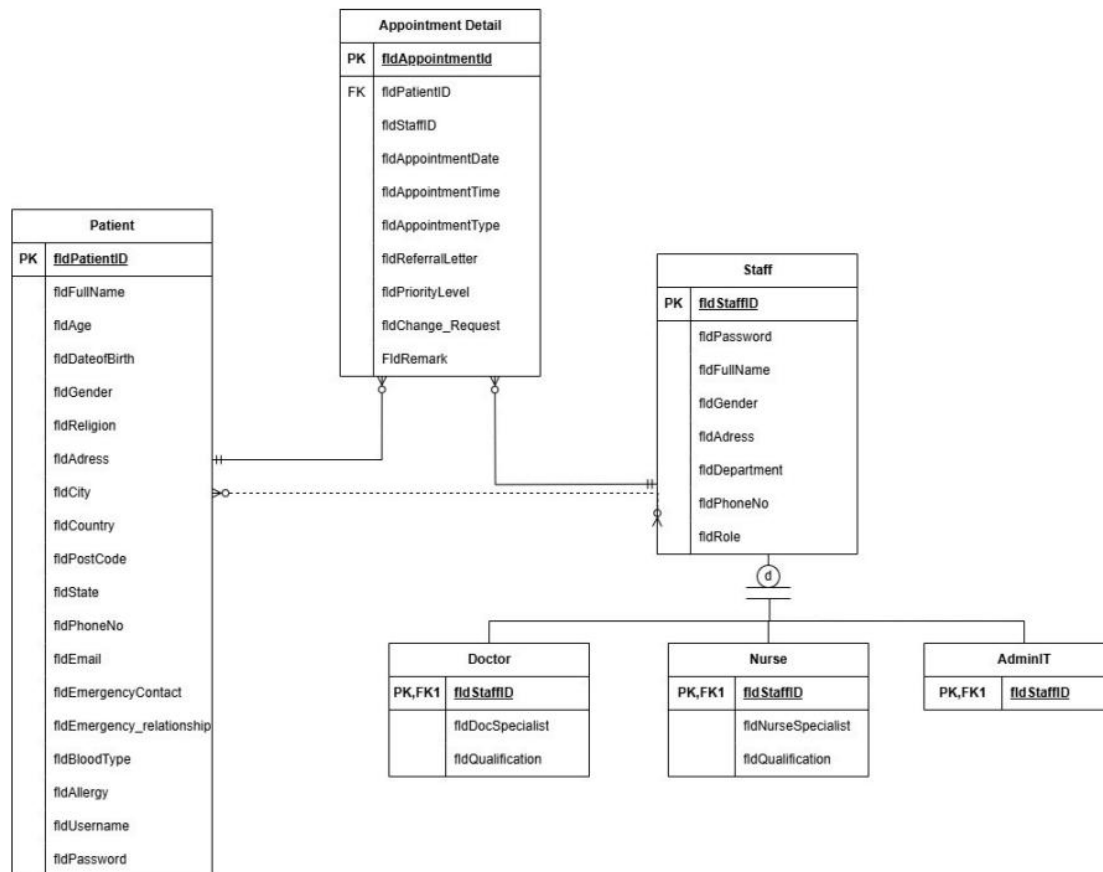


Figure 3-3 UML Diagram

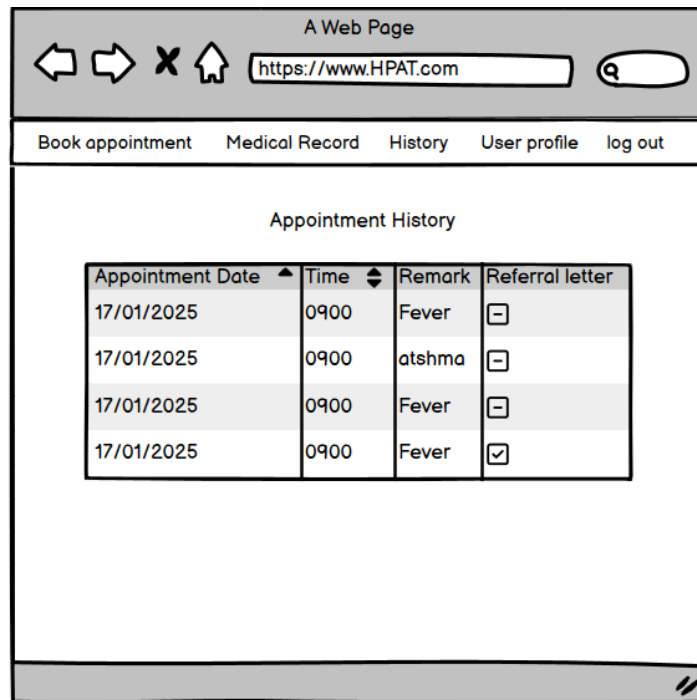
3.8 Wireframe

This section highlights the creation of a mockup interface for the Hospital Patient Appointment System using the Balsamiq wireframe design tool. The interface is designed to cater to both system administrators and hospital staff. Figure 3.4. illustrates the login screen for the Hospital Patient Appointment System, where administrators and staff must enter their username and password before clicking the Login button to access the system.

Figure 3-4 Wireframe diagram for login the system

Figure 3-5 Wireframe for book appointment for patient

Figure 3.5 displays the "Book Appointment" interface for the Hospital Patient Appointment System (HPAT). This page is designed for patients to schedule appointments with ease. It includes fields for selecting the appointment date and time, adding remarks for the appointment (such as symptoms), and uploading a referral letter if needed. Once all details are filled in, the user can click the "Submit" button to confirm the appointment. This simple and user-friendly design ensures patients can book their appointments efficiently.



A Web Page

https://www.HPAT.com

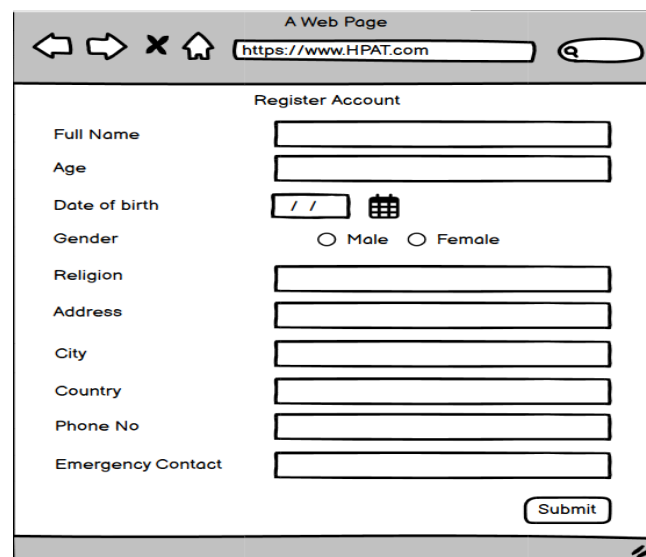
Book appointment Medical Record History User profile log out

Appointment History

Appointment Date ▲	Time ◆	Remark	Referral letter
17/01/2025	0900	Fever	☐
17/01/2025	0900	atshma	☐
17/01/2025	0900	Fever	☐
17/01/2025	0900	Fever	☒

Figure 3-6 Wireframe for Appointment History for patient

Figure 3.6 displays the "Appointment History" page of the Hospital Patient Appointment Portal (HPAP). This page provides patients with a detailed view of their past appointments. The information is organized into a table format with columns for the appointment date, time, remarks, and the presence of a referral letter. Users can easily review their appointment details, such as reasons for the visit (e.g., fever, asthma) and whether a referral letter was submitted.



A Web Page

https://www.HPAT.com

Register Account

Full Name

Age

Date of birth / / 

Gender ☐ Male ☐ Female

Religion

Address

City

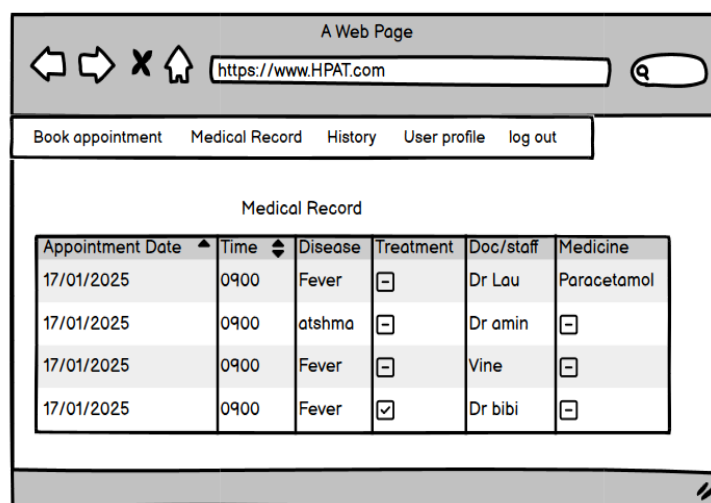
Country

Phone No

Emergency Contact

Figure 3-7 Wireframe for register account for patient

Figure 3.7 illustrates the "Register Account" interface designed for patients in the Hospital Patient Appointment Portal (HPAP). This page facilitates the patient registration process by capturing essential personal details. The form includes fields for entering the patient's full name, age, date of birth, gender (with radio buttons for male and female), religion, address, city, country, phone number, and emergency contact information. A clear and structured layout ensures that patients can input their details easily. The inclusion of a date picker for the date of birth enhances usability, while the "Submit" button at the bottom allows users to finalize their registration. This form is crucial for building patient profiles and ensuring accurate data collection for appointment management.



The wireframe shows a web browser window titled "A Web Page" with a URL bar containing "https://www.HPAT.com". Below the browser window is a navigation bar with links: "Book appointment", "Medical Record", "History", "User profile", and "log out". The main content area is titled "Medical Record" and contains a table with the following data:

Appointment Date	Time	Disease	Treatment	Doc/staff	Medicine
17/01/2025	0900	Fever	☐	Dr Lau	Paracetamol
17/01/2025	0900	asthma	☐	Dr amin	☐
17/01/2025	0900	Fever	☐	Vine	☐
17/01/2025	0900	Fever	☒	Dr bibi	☐

Figure 3-8 wireframe for medical record for patient

Figure 3.8 displays the " Medical Record " page of the Hospital Patient Appointment Portal (HPAP). This page provides patients with a detailed view of their past medical record. The information is organized into a table format with columns for the appointment date, time, disease, treatment, Doc/staff and medicine.

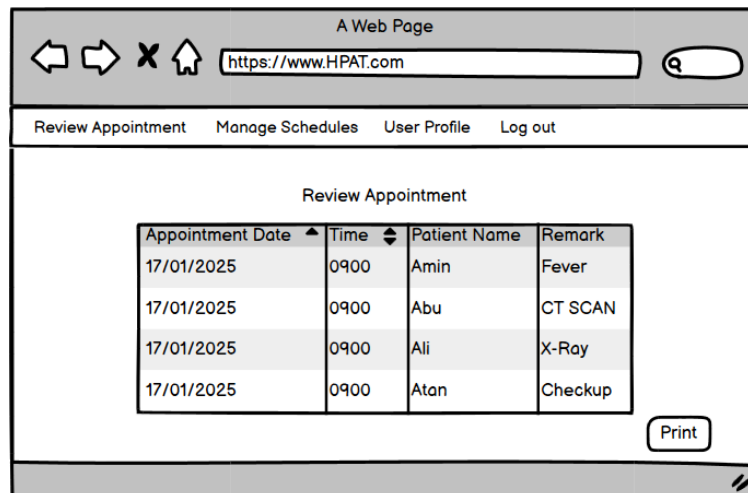


Figure 3-9 wireframe for review appointment for doc/staff

Figure 3.9 displays the "Review Appointment" page of the Hospital Patient Appointment Portal (HPAP). This page is designed for doctors and staff to view a list of upcoming appointments. The information is organized into a table format with columns for the appointment date, time, patient name, and remarks. The "Review Appointment" page helps healthcare staff manage their schedules efficiently. It allows them to quickly review patient details and remarks, such as reasons for the appointment (e.g., fever, CT scan, or checkup). At the bottom of the page, there is a "Print" button. This feature allows staff to generate a printed version of the schedule for offline use, ensuring they are prepared for the day's appointments.

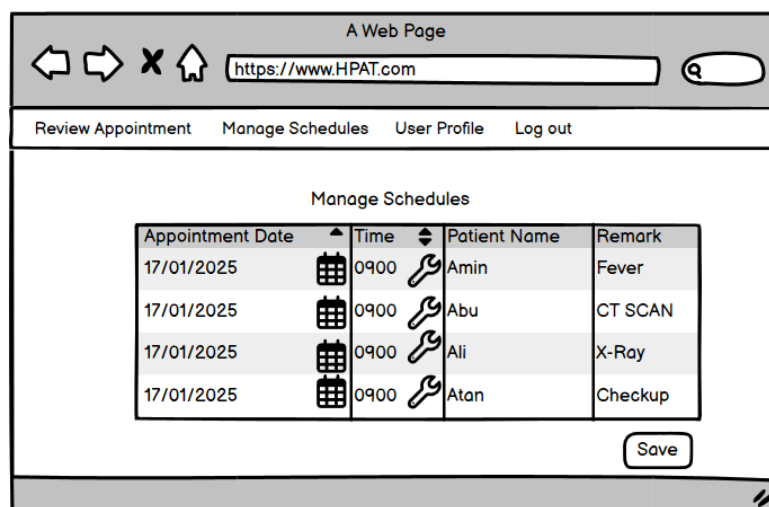


Figure 3-10 wireframe for manage schedule for Doc/ staff

Figure 3.10 displays the "Manage Schedules" page of the Hospital Patient Appointment Portal (HPAP). This page is designed for doctors and staff to manage and adjust appointment details. The information is displayed in a table format with columns for the appointment date, time, patient name, and remarks. The "Manage Schedules" page allows healthcare staff to edit the date and time of appointments as needed. Each row includes interactive icons next to the date and time fields for making changes. This feature ensures flexibility in rescheduling appointments based on staff availability or patient needs. At the bottom of the page, there is a "Save" button that enables staff to save any changes made to the schedule. This ensures all updates are securely stored and reflected in the system.

The wireframe shows a web browser window titled "A Web Page" with the URL "https://www.HPAT.com". The navigation bar includes links for "Review Appointment", "Manage Schedules", "User Profile", and "Log out". The main content area is titled "View Patient Detail" and contains the following fields:

Patient Name	Amin Bin Abu
Age	20
Gender	Male
Allergy	Seafood
Appointment Date & Time	17/01/2025 ,1903
Referral Letter	

A "Print" button is located at the bottom right of the form area.

Figure 3-11 Wireframe for View Patient Detail for Doc / Staff

Figure 3.11 displays the "View Patient Detail" page of the Hospital Patient Appointment Portal (HPAP). This page is designed for doctors and staff to view the patient details. The information is displayed with columns for the appointment date, time, patient name, gender, age, referral letter and allergy. At the bottom of the page, there is a "print" button that enables staff to print the detail of the patient for offline purpose.

A Web Page

https://www.HPAT.com

Register Account

Name

Password

Date of birth /

Gender ☐ Male ☐ Female

Address

Department

Phone No

Position

Figure 3-12 Wireframe for register page for admin

Figure 3.12 displays the "register page" page of the Hospital Patient Appointment Portal (HPAP). This page is designed for admin to register the doctor and staff to create an account. The information is contained with columns for the name, password, date of birth, gender, address, department, phone no and position. At the bottom of the page, there is a "submit" button that enables admin to save the detail and create an account for staff or doctor.

Chapter 4 : Implementation

4.1 Overview

This chapter describes the implementation and testing of the Hospital Patient Appointment Portal (HPAP) system. It provides an overview of the development process, including the tools, technologies, and development environment used to build the system. The chapter also explains how each module—such as user authentication, appointment scheduling, department management, and notification handling—was developed and integrated to form a fully functional application.

Additionally, it outlines the testing and evaluation strategies employed to verify the system's performance, reliability, and usability. Challenges encountered during the development phase are discussed, along with the practical solutions and adjustments made to address them. Overall, this chapter offers a detailed account of how the HPAP system was successfully implemented and validated.

4.2 Installation and Configuration

4.2.1 Hardware

The **Hospital Patient Appointment Portal (HPAP)** system was developed and tested on a laptop with sufficient processing capabilities to ensure smooth performance and system reliability. The application utilizes **React.js** for the front-end and **Node.js** for the back-end, technologies that benefit from responsive hardware—particularly when handling API calls, database operations, and rendering dynamic user interfaces.

To facilitate efficient development, testing, and real-time integration with external services, the system was built on a machine with the following specifications: **Intel Core i7-9750H @ 2.60GHz processor, 24GB of RAM, and 1TB of storage**. This configuration provided a stable and responsive development environment, enabling fast code compilation, quick server responses, and minimal latency during testing phases.

Cross-browser compatibility was ensured by testing the application on widely used web browsers, including **Google Chrome** and **Microsoft Edge**. These hardware and browser environments offered a robust foundation for both development and usability testing of the HPAP system.

4.2.2 Software

The development of the HPAP (Hospital Patient Appointment Portal) system involved the use of various software tools, frameworks, and technologies to support front-end design, back-end integration, API communication, and database operations. The frontend was developed using React.js, a widely-used JavaScript library for building fast, responsive, and component-based user interfaces. For styling and layout, Tailwind CSS was implemented to create a clean, mobile-responsive design that enhances the user experience for both patients and healthcare staff.

On the backend, the system was powered by Node.js and Express.js, which handled server-side logic, routing, API processing, and integration with the database. MongoDB, a NoSQL database, was used to store and manage patient records, appointment schedules, and department data efficiently. Communication between the frontend and backend was achieved through RESTful APIs, ensuring smooth data flow and system functionality.

Visual Studio Code (VS Code) was utilized as the main development environment due to its extensive support for JavaScript development, integrated terminal, and extension ecosystem. Version control was maintained using Git, with GitHub as the remote repository for collaboration, tracking changes, and ensuring backup throughout the development cycle.

For testing and debugging, the system was run on modern web browsers such as Google Chrome and Microsoft Edge. These browsers' developer tools were actively used to inspect UI elements, monitor API calls, verify network activity, and ensure cross-browser responsiveness and compatibility.

Development and testing were performed on a machine with the following specifications: Intel Core i7-9750H @ 2.60GHz, 24 GB RAM, and 1TB storage, which provided more than sufficient performance for running local servers, testing components, and managing databases concurrently.

This combination of modern tools and technologies provided a robust and efficient development workflow, facilitating the smooth creation, deployment, and testing of the HPAP system.

4.3 HPAP System Component

4.3.1 Dashboard

The dashboard serves as the central interface for the **UNIMAS Patient Appointment System**, offering users a streamlined and intuitive platform to manage healthcare appointments and access medical records. Designed with a responsive layout, the dashboard ensures a consistent user experience across both desktop and mobile devices, making healthcare services more accessible to students, staff, and the broader UNIMAS community.

At the top of the dashboard, users are welcomed by the **UNIMAS logo** and a clean navigation bar that provides quick access to key sections: **Home**, **Book Appointment**, **Add Health Records**, and **Health Records**. This layout promotes ease of use and allows users to navigate between booking services and managing their health data effortlessly.

The main display area features a prominent red banner that highlights the core functionality of the system:

"Book Appointment With Pusat Kesehatan UNIMAS".

This section encourages users to browse through the available health departments and schedule appointments with minimal friction. A clearly visible **"Book Appointment"** button, accompanied by an arrow icon, directs users to the booking interface, enhancing usability and guiding user interaction.

The visual design includes background imagery of medical professionals, such as a doctor holding a clipboard and other healthcare staff, reinforcing the system's focus on professional and reliable healthcare services. These visuals help build trust and convey the system's purpose effectively.

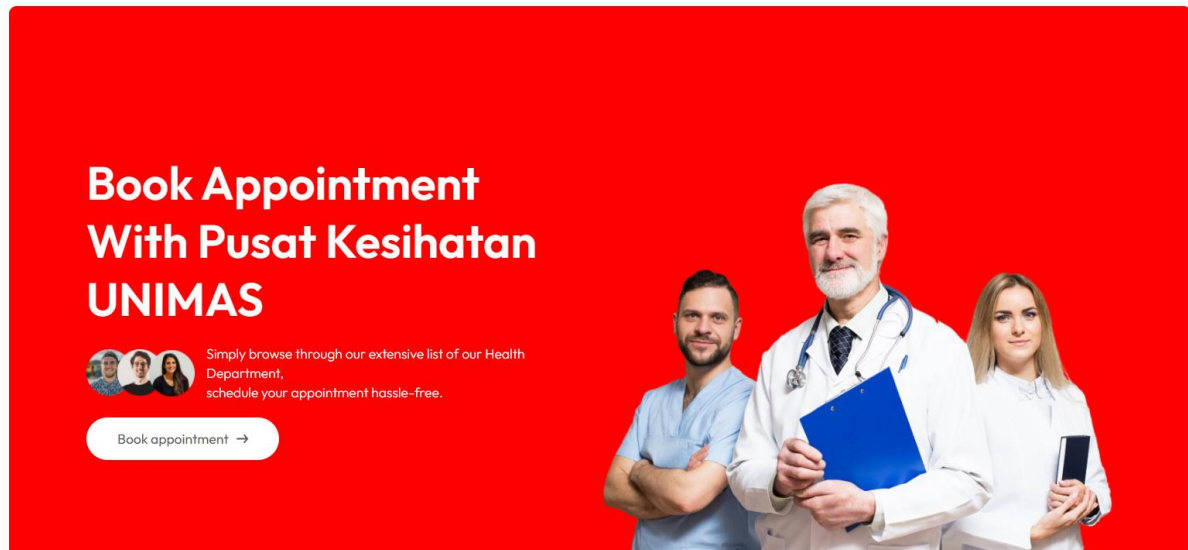


Figure 4-1 dashboard for patient

4.3.2 Appointment Booking Page

Choose Department

Select Department

 Cardiology	 Radiology
 Orthopedics	 General Medicine
 Psychiatry	 Pharmacy

Next

Figure 4-2 appointment page form-1

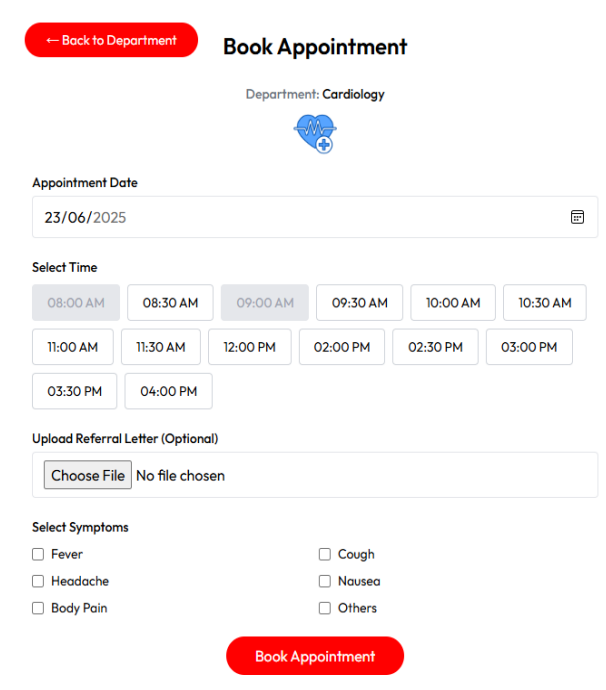
Upon clicking the "**Book Appointment**" button on the main dashboard, users are seamlessly navigated to the **Department Selection Form**. This transition marks the first step in the

appointment scheduling process, guiding users to choose the appropriate medical department based on their healthcare needs.

The new page is titled "**Choose Department**" and presents a visually organized layout of available departments at **Pusat Kesehatan UNIMAS**. Each department is represented with a distinct icon and label, enhancing clarity and accessibility. The departments listed include Cardiology, Radiology, Orthopedics, General Physique and Pharmacy. This interface is designed to be intuitive, allowing users to quickly identify and select the department relevant to their condition or concern. However, this page for department can be added from admin.

Navigation and User Flow

Once a department is selected, users can proceed by clicking the "**Next**" button located at the bottom of the form. This action advances the user to the next stage of the appointment process, which typically involves selecting a date, time slot, and confirming personal details.



The screenshot displays the 'Book Appointment' form for the Cardiology Department. At the top left, there is a red button labeled '← Back to Department'. The title 'Book Appointment' is centered at the top. Below the title, the department is specified as 'Department: Cardiology' with a heart icon. The 'Appointment Date' is set to '23/06/2025'. The 'Select Time' section shows a grid of time slots: 08:00 AM, 08:30 AM, 09:00 AM (highlighted), 09:30 AM, 10:00 AM, 10:30 AM, 11:00 AM, 11:30 AM, 12:00 PM, 02:00 PM, 02:30 PM, 03:00 PM, 03:30 PM, and 04:00 PM. Below this, there is an 'Upload Referral Letter (Optional)' section with a 'Choose File' button and the text 'No file chosen'. The 'Select Symptoms' section includes checkboxes for Fever, Headache, Body Pain, Cough, Nausea, and Others. At the bottom right, there is a red button labeled 'Book Appointment'.

Figure 4-3 appointment page -2

Following the selection of a department from the Department Selection Form, users are directed to a specialized Appointment Booking Form tailored to the chosen department. In this instance, the form is customized for the **Cardiology Department**, ensuring that the

information collected is relevant and specific to the nature of the consultation. This form is designed to streamline the appointment scheduling process by gathering all necessary details in a structured and user-friendly manner.

At the top of the page, users are provided with a **"Back to Department"** button, allowing them to return to the previous screen and modify their department selection if needed. This feature enhances navigation flexibility and ensures users can easily correct any selection errors. Just below this, the page is clearly titled **"Book Appointment"**, with the selected department name—**Cardiology**—displayed prominently. A medical icon accompanies the department name, serving as a visual cue that reinforces the department's identity and helps orient the user.

The form itself comprises several interactive components designed to capture essential appointment details. The first element is the **Appointment Date Selection**, which utilizes a calendar input. This calendar is intelligently configured to **disable weekends (Saturdays and Sundays)** and **prevent the selection of backdated dates**, ensuring that users can only choose valid and available appointment slots. This constraint reflects the operational schedule of the health center and helps maintain the integrity of the booking system.

Next, users are presented with a **Time Slot Selection** grid, offering appointment times ranging from **08:00 AM to 04:00 PM**. This allows users to select a time that best fits their personal schedule, providing flexibility and convenience. The time slots are clearly displayed and easy to interact with, contributing to a smooth user experience.

An optional feature of the form is the **Referral Letter Upload** section. Here, users can upload a referral letter from a general practitioner or another medical professional. This functionality supports better coordination between healthcare providers and ensures that specialists have access to relevant patient information prior to the consultation.

To further assist healthcare professionals in preparing for the appointment, the form includes a **Symptom Checklist**. Users can indicate their symptoms by selecting from a list that includes **Fever, Cough, Headache, Nausea, Body Pain, and Others**. This information helps prioritize cases based on urgency and allows medical staff to anticipate patient needs more effectively.

Finally, the form concludes with a prominently displayed **red "Book Appointment"** button. Once all required fields are completed, users can submit their appointment request. This action triggers the backend logic to store the appointment details and potentially send a confirmation to the user

4.3.3 My appointment

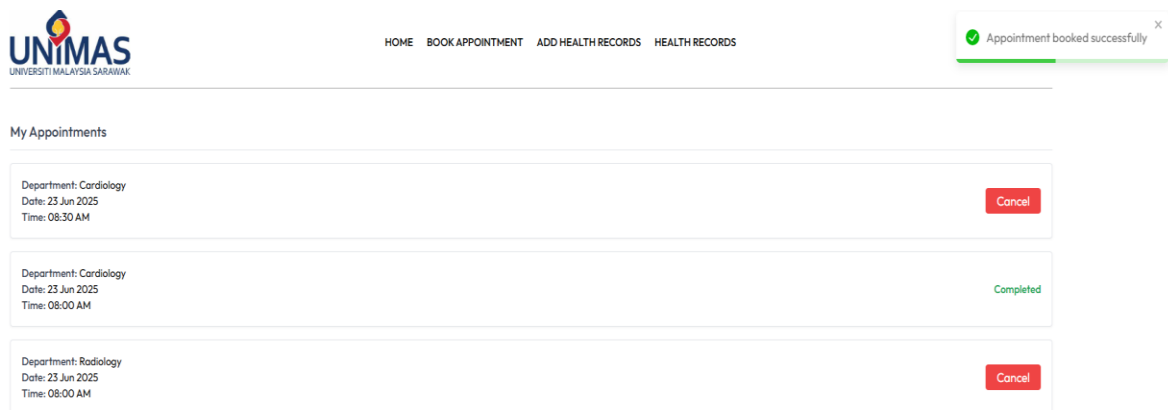


Figure 4-4 My Appointment page

Upon successfully booking an appointment through the Patient Appointment System, users are automatically redirected to the **"My Appointments"** page. This page serves as a centralized dashboard for users to view, manage, and track their upcoming and past medical appointments at **Pusat Kesihatan UNIMAS**.

A notification banner appears at the top right corner of the screen, confirming the successful booking with a message:

"Appointment booked successfully"

This message is accompanied by a green checkmark icon, providing immediate visual feedback and reassurance to the user.

The main section of the page is titled **"My Appointments"** and displays a list of all appointments associated with the user. Each appointment entry includes the following details:

- I. **Department Name** (e.g., Cardiology, Radiology)

II. **Appointment Date** (e.g., 23 Jun 2025)

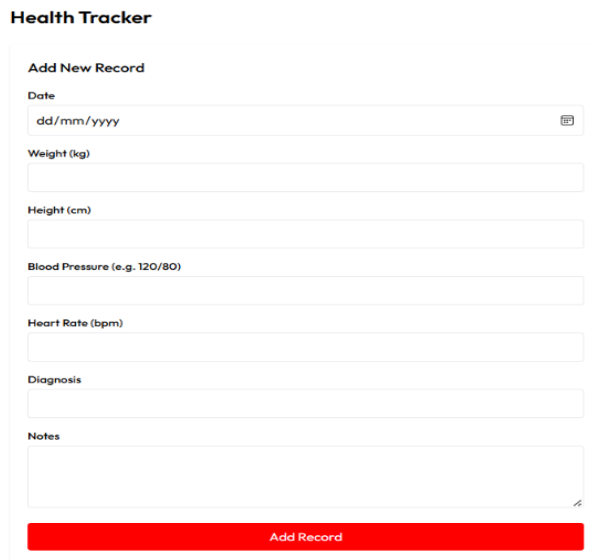
III. **Appointment Time** (e.g., 08:30 AM)

Each appointment is accompanied by a status indicator or action button:

- I. **Cancel Button:** Available for upcoming appointments, allowing users to cancel if needed.
- II. **Completed Status:** Displayed for past appointments that have already occurred, indicating that no further action is required.

This page not only confirms the user's booking but also empowers them to manage their healthcare schedule proactively. By providing clear visibility into both upcoming and completed appointments, the system enhances transparency and user control.

4.3.4 Add Health Record



The screenshot shows a web form titled "Health Tracker" with a sub-header "Add New Record". The form contains several input fields: "Date" with a placeholder "dd/mm/yyyy" and a calendar icon; "Weight (kg)", "Height (cm)", "Blood Pressure (e.g. 120/80)", "Heart Rate (bpm)", "Diagnosis", and "Notes" (a larger text area). At the bottom is a prominent red button labeled "Add Record".

Figure 4-5 Add Health Record

The **Health Tracker** module within the Patient Appointment System provides users with the ability to log and monitor their personal health metrics over time. This feature is designed to support continuous health tracking, enabling users to maintain a digital record of key physiological indicators and medical observations. The **Add Health Record** page is a central

component of this module, offering a structured and user-friendly form for entering new health data.

At the top of the form, users are prompted to input the **Date** of the health record. This field includes a calendar icon and follows the placeholder format **dd/mm/yyyy**, allowing users to select the appropriate date with ease. The calendar component is likely configured to prevent the selection of invalid or future dates, ensuring accurate historical tracking.

The form then guides users through a series of fields to capture essential health metrics:

- I. **Weight (kg):** Users enter their current body weight, which can be used to monitor changes over time.
- II. **Height (cm):** This static metric supports BMI calculations and general health assessments.
- III. **Blood Pressure:** Input is typically in the format **systolic/diastolic** (e.g., 120/80), providing insight into cardiovascular health.
- IV. **Heart Rate (bpm):** Users record their pulse rate, which is a key indicator of fitness and stress levels.

In addition to these quantitative metrics, the form includes fields for:

- I. **Diagnosis:** Users can enter any medical diagnosis received during consultations or self-monitoring.
- II. **Notes:** This open-ended field allows users to add contextual information, such as symptoms, lifestyle changes, or medication updates.

At the bottom of the form, a clearly visible **red "Add Record"** button enables users to submit their health data. Upon submission, the record is stored in the system's database and becomes part of the user's personal health history, accessible through the **Health Records** section.

4.3.5 Health Record

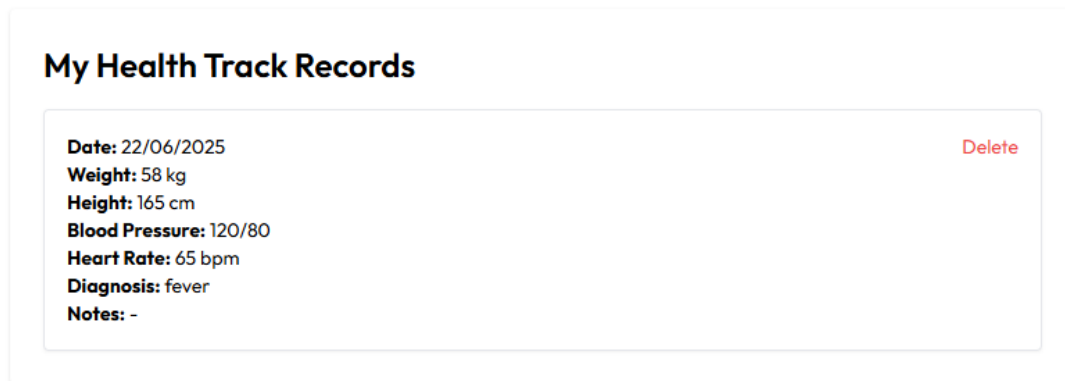


Figure 4-6 Health Record

The **"My Health Track Records"** page is a vital component of the Patient Appointment System, designed to provide users with a clear and organized view of their personal health history. This feature supports long-term health monitoring and enables users to track changes in their physiological metrics over time, contributing to better self-awareness and more informed medical consultations.

Upon accessing this page, users are presented with a list of previously submitted health records. Each record is displayed in a structured format, showing key health indicators and medical notes. For example, a typical entry includes date, weight, height, blood pressure, heart rate, notes and diagnosis.

Each record also includes a **"Delete"** option, displayed in red text on the right side of the entry. This allows users to remove outdated or incorrect records, giving them control over their health data and ensuring the accuracy of their personal health history.

4.3.6 My Notifications

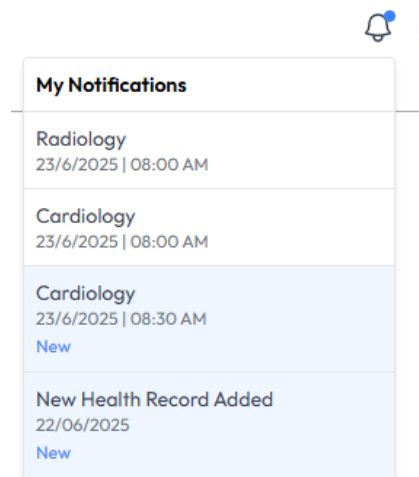


Figure 4-7 my notifications

The **"My Notifications"** page is an essential feature of the Patient Appointment System, designed to keep users informed about their recent activities and updates within the platform. This panel provides a centralized view of all relevant notifications, ensuring that users remain aware of upcoming appointments, newly added health records, and other system-generated alerts.

At the top of the page, the section is clearly titled **"My Notifications"**, establishing its purpose and guiding users to check for updates. The notifications are listed in a chronological and categorized format, making it easy for users to scan and identify important information.

Each notification entry includes:

- I. **Department Name** (e.g., Radiology, Cardiology)
- II. **Date and Time** of the appointment (e.g., 23/6/2025 | 08:00 AM)
- III. **Status Indicator** such as **"New"**, which highlights recent updates that the user has not yet viewed.

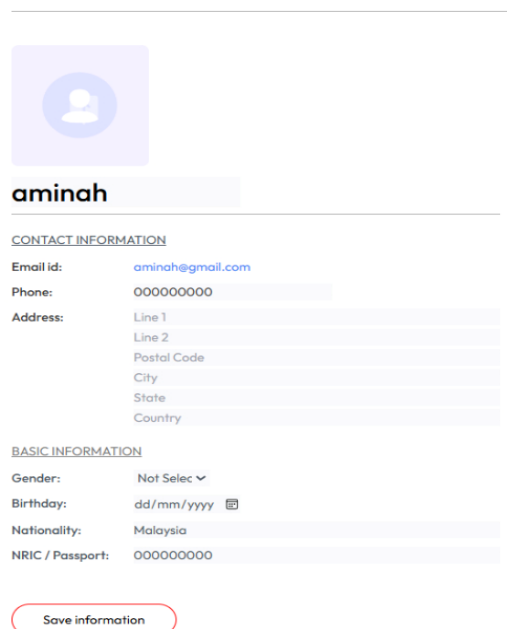
For example, the panel displays:

1. An appointment in **Radiology** scheduled for **23/6/2025 at 08:00 AM**.
2. Two appointments in **Cardiology**, one at **08:00 AM** and another at **08:30 AM**, with the latter marked as **"New"**.
3. A system-generated notification stating **"New Health Record Added"** on **22/06/2025**, also marked as **"New"**.

These notifications serve multiple purposes:

- I. **Reminders** for upcoming appointments to help users manage their schedules.
- II. **Alerts** for newly added health records, ensuring users are aware of updates to their medical history.
- III. **Status tracking**, allowing users to distinguish between viewed and unviewed items.

4.3.7 My Profile



The screenshot displays a user profile interface. At the top, there is a placeholder for a profile picture (a light purple square with a white person icon) and the name 'aminah'. Below this, the page is divided into two main sections: 'CONTACT INFORMATION' and 'BASIC INFORMATION'. The 'CONTACT INFORMATION' section includes fields for 'Email id:' (aminah@gmail.com), 'Phone:' (00000000), and 'Address:' (with sub-fields for Line 1, Line 2, Postal Code, City, State, and Country). The 'BASIC INFORMATION' section includes fields for 'Gender:' (Not Selec), 'Birthday:' (dd/mm/yyyy), 'Nationality:' (Malaysia), and 'NRIC / Passport:' (00000000). At the bottom of the form, there is a 'Save information' button.

Figure 4-8 My Profile

The **Profile Page** in the Patient Appointment System serves as a centralized interface for users to manage and update their personal information. This feature is essential for maintaining

accurate user records, which are critical for appointment scheduling, health tracking, and communication with healthcare providers.

At the top of the page, a **profile picture placeholder** is displayed, represented by a generic user icon. This space can be enhanced in future iterations to allow users to upload a personal photo, adding a layer of personalization to the system.

The profile section begins with the user's **name**, in this case, "aminah," followed by a set of **contact information fields**:

- I. **Email ID:** aminah@gmail.com
- II. **Phone Number:** 0000000000
- III. **Address:** Structured into multiple fields including Line 1, Line 2, Postal Code, City, State, and Country

These fields ensure that the system can store complete and structured contact details, which are essential for appointment confirmations, reminders, and potential follow-ups.

The **Basic Information** section captures key demographic and identification details:

- **Gender:** Provided as a dropdown menu, allowing users to select their gender.
- **Birthday:** Entered via a calendar input field in the format **dd/mm/yyyy**, ensuring consistency and ease of use.
- **Nationality:** Pre-filled as **Malaysia**, reflecting the system's primary user base.
- **NRIC / Passport Number:** A unique identifier field, which can be used for verification and record linkage.

At the bottom of the form, a "**Save Information**" button allows users to submit any updates made to their profile. This ensures that all changes are stored securely and reflected across the system.

4.3.8 Admin Dashboard

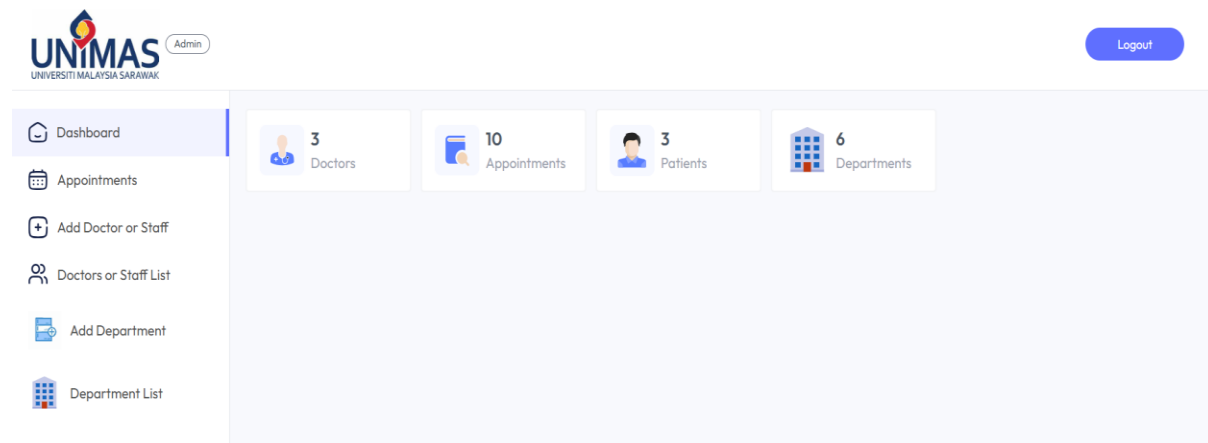


Figure 4-9 Admin Dashboard

The **Admin Dashboard** is a core component of the Patient Appointment System, designed to provide administrative users with a centralized interface for managing healthcare operations at **Pusat Kesihatan UNIMAS**. This dashboard empowers administrators to oversee appointments, manage staff and departments, and maintain system integrity through a streamlined and intuitive layout.

The left-hand sidebar serves as the primary navigation panel, offering quick access to essential administrative functions:

- I. **Dashboard** – Returns to the main overview screen.
- II. **Appointments** – Allows viewing and managing all scheduled patient appointments.
- III. **Add Doctor or Staff** – Opens a form to register new healthcare personnel into the system.
- IV. **Doctors or Staff List** – Displays a searchable list of all registered medical staff.
- V. **Add Department** – Enables the creation of new medical departments.
- VI. **Department List** – Shows all existing departments with options to edit or delete entries.

This structured navigation ensures that administrators can efficiently perform their duties without unnecessary complexity. The central area of the dashboard presents a set of **summary**

cards, each displaying key system metrics include Doctors, Appointment ,Patients and Departments. Each card includes an icon and a numerical value, offering a quick visual snapshot of the system's current status. This overview helps administrators monitor activity levels and identify areas that may require attention or updates. A **"Logout"** button is located at the top right corner, allowing administrators to securely exit the system once their tasks are completed. This ensures session security and protects sensitive data from unauthorized access.

4.3.9 Admin Login and Staff Login

The Patient Appointment System includes two distinct login interfaces to support **role-based access control**: one for **administrators** and another for **medical staff**. These interfaces ensure that users are directed to the appropriate dashboard based on their role, maintaining system security and operational clarity.

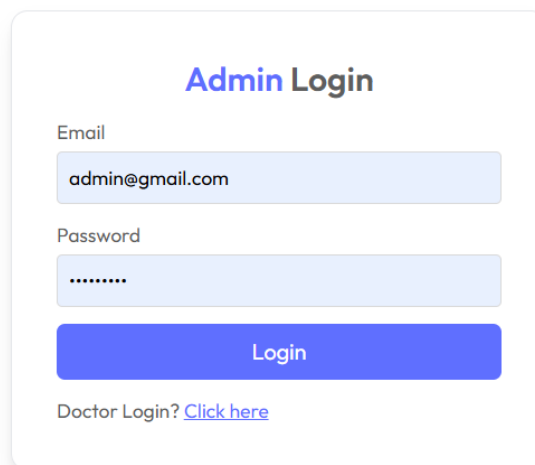
The image shows a web form titled "Admin Login" in blue text. Below the title, there are two input fields: "Email" with the value "admin@gmail.com" and "Password" with masked characters ".....". A blue "Login" button is positioned below the password field. At the bottom, there is a link that says "Doctor Login? Click here".

Figure 4-10 Admin Login

The **Admin Login** page is designed for system administrators who manage appointments, departments, and healthcare personnel. It includes:

- I. **Email and Password fields** for credential input.
- II. A **Login button** to initiate authentication.
- III. A link labeled **"Doctor Login? Click here"** to redirect users who may have selected the wrong login type.

Upon successful login, administrators are directed to the **Admin Dashboard**, where they can oversee system operations and manage resources.

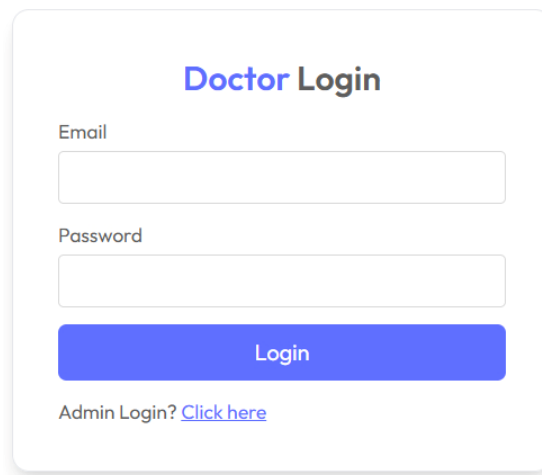
A login form titled "Doctor Login" in blue text. It contains two input fields: "Email" and "Password", both with light gray borders. Below the password field is a solid blue button with the text "Login" in white. At the bottom, there is a link that says "Admin Login? [Click here](#)".

Figure 4-11 Staff Login

The **Doctor Login** page is tailored for healthcare professionals. It features:

- I. **Email and Password fields** for secure login.
- II. A **blue Login button** for submission.
- III. A link labeled "**Admin Login? Click here**" for role correction.

Once authenticated, doctors gain access to their dedicated dashboard, where they can view appointments, patient records, and other relevant clinical data.

4.3.10 All Appoinment Admin

All Appointments				
#	Patient	Department	Date & Time	Action
1	aminah	Cardiology	23 Jul 2025, 08:30 AM	Cancel
2	aminah	Cardiology	23 Jul 2025, 08:00 AM	Completed
3	aminah	Radiology	23 Jul 2025, 08:00 AM	Cancel
4	Muhammad Zahravi Alghazali Bin Muhammad Zahrudin	Cardiology	26 Jul 2025, 09:00 AM	Completed
5	Muhammad Zahravi Alghazali Bin Muhammad Zahrudin	Cardiology	24 Jul 2025, 09:30 AM	Completed
6	Muhammad Zahravi Alghazali Bin Muhammad Zahrudin	Cardiology	26 Jul 2025, 08:30 AM	Completed

Figure 4-12 All Appointment Admin

The "All Appointments" page within the Admin Dashboard provides administrators with a comprehensive overview of all scheduled patient appointments across various departments. This interface is designed to facilitate administrative oversight and ensure that appointments are properly tracked and managed. The page features a structured table displaying key appointment details, including Patient Name, Department, Date & time and Action.

Each row in the table represents an individual appointment, allowing admins to monitor scheduling activity in real time. Importantly, the system enforces role-based permissions. Administrators can cancel appointments but cannot mark them as completed.

This restriction ensures that only medical staff—who conduct the consultations—can update the status to "Completed," preserving the accuracy and integrity of medical records.

4.3.11 Add Doctor Admin

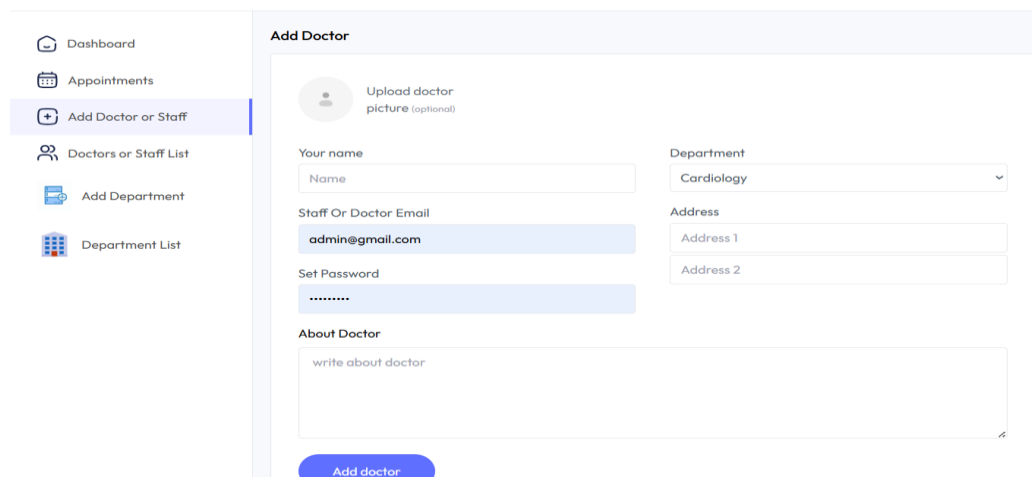


Figure 4-13 Add Doctor Admin Page

As part of the **Admin Dashboard**, the **Add Doctor or Staff** page enables administrators to register new medical personnel into the Patient Appointment System. This feature is essential for maintaining an up-to-date directory of healthcare providers and ensuring that departments are staffed appropriately.

The interface includes a structured form with the following fields:

- I. **Profile Picture Upload:** Allows the admin to upload a photo of the doctor or staff member.
- II. **Name:** Input field for the full name of the individual.
- III. **Department Selection:** A dropdown menu to assign the individual to a specific department (e.g., Cardiology) .It will take data from department list
- IV. **Email and Password:** Credentials for system login, ensuring secure access for the staff member.
- V. **Address:** Two fields for entering detailed location information.
- VI. **About Doctor:** A text area for entering a brief biography or professional background.

Once all fields are completed, the admin can click the **"Add Doctor"** button to submit the form. This action stores the new staff member's information in the system database and enables them to access their respective dashboard.

4.3.12 All Doctor List for Admin



Figure 4-14 Doctor List

The **Doctors or Staff List** page is a key feature within the Admin Dashboard of the Patient Appointment System. It provides administrators with a centralized view of all registered medical personnel, allowing for efficient management and oversight of healthcare staff.

The interface displays a list of doctor profiles under the heading "**All Doctors**", each showing name and available.

4.3.13 Dashboard for Doctor/Staff

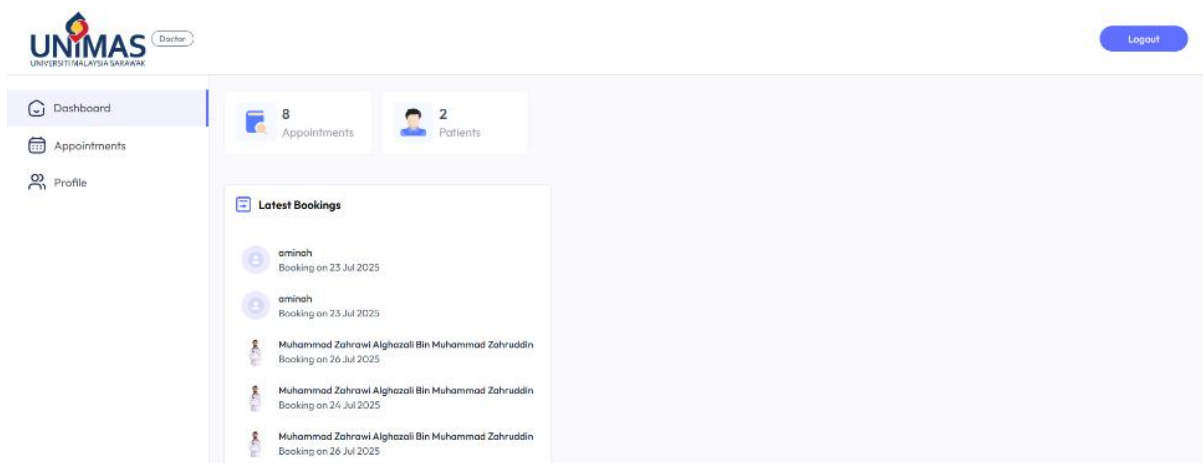


Figure 4-15 dashboard for staff and doctor

This dashboard provides doctors with a streamlined view of their clinical responsibilities, including upcoming appointments and patient interactions. The left-hand sidebar includes essential navigation options:

- I. Dashboard – Returns to the main overview screen.

- II. Appointments – Directs the doctor to a detailed list of their scheduled consultations.
- III. Profile – Allows the doctor to view and update their personal and professional information.
- IV. A Logout button is positioned at the top right corner, enabling secure session termination.

At the top center of the main content area, two key indicators provide a quick summary such as :

- I. Appointments – Displays the total number of upcoming or active appointments.
- II. Patients – Indicates the number of unique patients under the doctor’s care.

These metrics offer a snapshot of the doctor’s current workload and patient engagement. Beneath the summary indicators is the "Latest Bookings" section, which lists recent appointment entries. Each entry includes the patient name and booking date.

4.3.14 All Appointment for doctor

Dashboard

Appointments

Profile

All Appointments

#	Patient	Age	Date	Time	Symptoms	Referral	Department	Action
1	aminah	NaN yrs	23 Jul 2025	08:30 AM	Fever, Headache	View	Cardiology	
2	aminah	NaN yrs	23 Jul 2025	08:00 AM	Fever, Headache	None	Cardiology	Completed
3	Muhammad Zahrawi Alghazali Bin Muhammad Zahrudin	23 yrs	26 Jul 2025	09:00 AM	Nausea, Cough	View	Cardiology	Completed
4	Muhammad Zahrawi Alghazali Bin Muhammad Zahrudin	23 yrs	24 Jul 2025	09:30 AM	Cough, Nausea	None	Cardiology	Completed

Figure 4-16 All Appointment for doctors

The **Appointment Management Page** within the Doctor Dashboard allows medical professionals to view and manage their scheduled consultations. Doctors can perform two key actions on each appointment:

- I. Mark as Completed
- II. Cancel Appointment
- III. View patient detail

IV. Review referral letter

When a doctor marks an appointment as **Completed**, the system automatically redirects them to the **Health Tracker Record Form** for the respective patient. This form allows the doctor to log key health metrics and consultation details, including Date of record, Weight, height, blood pressure, heart rate, Diagnosis and notes .This integration ensures that every completed appointment is followed by a structured health record entry, supporting continuity of care and accurate medical documentation.

4.3.15 Health Track Record for Doctor

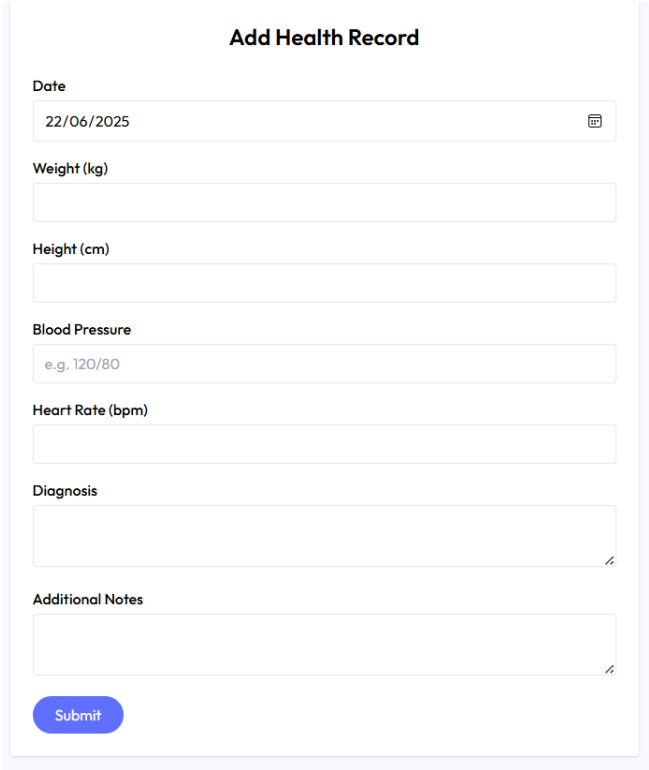


Figure 4-17Health Track Record for Doctor

Within the Doctor Dashboard, doctors can manage their appointments by either marking them as completed or cancelling them. However, this system includes a structured workflow to ensure that every completed appointment is properly documented.

In the Patient Appointment System, the process of completing a medical consultation is designed to ensure that every appointment is properly documented and linked to the patient's health history. This is achieved through a structured **completion workflow** that connects the appointment status with the health record entry process.

The workflow begins when a doctor clicks the "**Complete**" button for a scheduled appointment. At this stage, the system does not immediately update the appointment status. Instead, it redirects the doctor to the **Add Health Record** form associated with the specific patient. This redirection ensures that the doctor is prompted to document the consultation details before the appointment is officially marked as completed.

The **Add Health Record** form is a structured interface that captures essential clinical data. The doctor is required to fill in the following fields:

- I. **Date of consultation**
- II. **Vital signs** such as weight, height, blood pressure, and heart rate
- III. **Diagnosis** made during the consultation
- IV. **Additional notes** that may include observations, treatment plans, or follow-up instructions

Once the form is completed and submitted, the system performs two critical actions:

1. The health record is securely saved and linked to the patient's profile.
2. The appointment status is automatically updated to "**Completed**".

This integrated process ensures that no appointment is marked as completed without a corresponding health record. It promotes accurate documentation, supports continuity of care, and enhances the reliability of patient data within the system. By enforcing this workflow, the system not only improves clinical accountability but also contributes to better long-term health tracking and decision-making.

4.4 Summary of the chapter

This chapter described the implementation and testing of the HPAP System, including the development tools, environment, and core system components. It explained the functionality of key modules such as the User Dashboard, Doctor Dashboard, Admin Dashboard, and Health

Tracker, all designed to improve healthcare access and record-keeping at Pusat Kesehatan UNIMAS.

The system allows patients to book appointments, view health records, and receive notifications. Doctors can manage appointments within their department, complete consultations, and submit health records. Administrators can oversee appointments, departments, and staff, with permissions limited to cancellation and management tasks.

A key feature of the system is the completion workflow, where doctors must fill out a health record form before an appointment is marked as completed. This ensures accurate documentation and supports continuity of care. The system also enforces role-based access control, ensuring that users only interact with data relevant to their role and department.

Chapter 5 Result and Discussion

5.1 Overview

This chapter discusses the findings and analysis from testing the Patient Appointment and Health Management System, a web-based platform developed to improve healthcare services at Pusat Kesehatan UNIMAS. Testing was essential to evaluate the system's performance, usability, and effectiveness in helping users manage appointments and health records.

Usability testing was conducted to assess how well the system meets its goals. The evaluation focused on key areas such as the clarity of the user interface, the accuracy of appointment handling, the relevance of health tracking features, and overall user satisfaction. Tools like the System Usability Scale (SUS) and the User Experience Questionnaire (UEQ) were used to gather feedback from users.

The goal was to understand how patients, doctors, and administrators interact with the system, identify usability issues, and determine whether the platform supports efficient healthcare management. The insights gained from this testing are important for improving the system and ensuring it delivers practical value to its users.

5.2 Setting

Individual user sessions were conducted to evaluate the Hospital Patient Appointment Portal (HPAP), ensuring a controlled and supportive environment for testing. At the beginning of each session, participants received personalized guidance on how to use the portal's core features, including booking appointments, viewing doctor availability, and accessing medical record.

Following the instructional phase, users were encouraged to explore the portal independently, simulating real-world scenarios that typical patients might encounter when managing their healthcare appointments. This hands-on interaction allowed participants to engage with the system in a manner reflective of their everyday healthcare decision-making processes.

The combination of initial instruction and autonomous exploration provided a comprehensive assessment of both the portal's usability and the intuitiveness of its design. This

approach ensured that feedback captured the full spectrum of user experience, from first-time use to routine navigation.

5.3 Participants

To evaluate the Hospital Patient Appointment Portal (HPAP), a total of 15 volunteers were recruited, comprising final-year students from Universiti Malaysia Sarawak (UNIMAS) and medical staff from a local healthcare facility. The student participants, aged between 21 and 25, were selected from the Faculty of Computer Science and Information Technology due to their proficiency in digital systems and familiarity with user interface design and software usability principles.

In addition to the students, medical staff were included to represent actual end-users of the portal. Their practical experience in managing patient appointments and interacting with healthcare systems provided valuable insights into the portal's real-world applicability and effectiveness.

While HPAP is primarily designed for patients, involving both technically skilled students and healthcare professionals allowed for a comprehensive initial evaluation. The students contributed informed feedback on usability and design, while the medical staff assessed the system's relevance and functionality in a clinical context. This dual-group approach ensured that the system was thoroughly tested before broader deployment to the general patient population.

5.4 Data Collection

This study employed two standardized instruments to collect quantitative data: the System Usability Scale (SUS) and the User Experience Questionnaire (UEQ). The SUS was utilized to evaluate the overall usability of the Hospital Patient Appointment Portal (HPAP), offering a reliable and widely accepted benchmark for assessing user satisfaction, ease of use, and system effectiveness. After interacting with the portal, participants completed the SUS questionnaire, providing valuable insights into their perceptions of the system's usability.

To complement the SUS, the UEQ was administered to capture a broader perspective on the user experience. The UEQ evaluates multiple dimensions, including attractiveness,

efficiency, dependability, and stimulation, allowing for a deeper understanding of both the functional and emotional aspects of user interaction with HPAP.

This dual-instrument approach enabled a comprehensive assessment of the portal, combining objective usability metrics with subjective user experience feedback. The resulting data formed a robust foundation for analyzing how both UNIMAS students and medical staff perceived the system, guiding further improvements and refinements.

5.5 Questionnaire

Participants provided feedback via a Google Form after testing the Hospital Patient Appointment Portal (HPAP). In addition to standardized usability instruments, a set of custom questions was included to gather user perceptions regarding the usefulness, clarity, and relevance of the system's features. These questions were designed to better understand how users evaluated the effectiveness and practicality of HPAP from a real-world perspective.

Examples of the custom questions included:

- i. What features do you like the most in HPAP?
- ii. What features would you like to see improved or added?
- iii. Did you face any challenges while using HPAP? If yes, please explain.

5.6 Quantitative Data

To assess the usability and user experience of the Hospital Patient Appointment Portal (HPAP), two standardized instruments were employed: the System Usability Scale (SUS) and the User Experience Questionnaire (UEQ). These tools provided structured and quantifiable insights into how users rated the system in terms of usability, satisfaction, and emotional engagement. The combination of SUS and UEQ enabled a comprehensive evaluation of both the functional effectiveness and the overall user experience of the platform.

5.6.1 System Usability Scale (SUS)

The System Usability Scale (SUS) is a widely recognized and validated tool used to assess perceived usability. It consists of 10 items rated on a 5-point Likert scale, where 1 indicates "strongly disagree" and 5 indicates "strongly agree." After interacting with the HPAP system, participants were asked to complete the SUS questionnaire based on their experience.

The 10 SUS questions used in this study were as follows:

No.	Question
Q1	I think that I would like to use this system frequently.
Q2	I found the system unnecessarily complex.
Q3	I thought the system was easy to use.
Q4	I thought there was too much inconsistency in this system.
Q5	I would imagine that most people would learn to use this system very quickly.
Q6	I found the system very cumbersome to use.
Q7	I felt very confident using the system
Q8	I needed to learn a lot of things before I could get going with this system.
Q9	I think that I would need the support of a technical person to use this system.
Q10	I found the various functions in this system were well integrated.

Table 12 SUS Question

The SUS evaluation for the HPAP platform revealed a high level of perceived usability among users. The SUS scores from 15 respondents ranged from 77.5 to 100, with the majority scoring above 85. This indicates that users found the platform to be simple to use, efficient, and well-designed.

According to established SUS benchmarks, a score above 68 is considered above average, while scores in the 80s and 90s reflect excellent usability. The consistently high scores suggest

that users had a positive interaction experience, with minimal confusion or frustration. These results are particularly significant given the platform's role in managing healthcare appointments, where clarity, speed, and ease of use are critical.

The findings confirm that the HPAP system is not only functionally effective but also highly user-friendly. It is suitable for a wide range of users, including those with limited technical knowledge. The strong usability ratings also imply that the system requires little to no training or onboarding, which is advantageous for encouraging adoption among patients and healthcare staff alike.

Respondent	SUS Score
1	95.0
2	100.0
3	95.0
4	85.0
5	80.0
6	85.0
7	77.5
8	100.0
9	100.0
10	92.5
11	90.0
12	82.5
13	85.0
14	90.0
15	92.5

Table 13 SUS Score

5.6.2 User Experience Questionnaire (UEQ)

The User Experience Questionnaire (UEQ) was used to assess six key dimensions of user experience: Attractiveness, Perspicuity, Efficiency, Dependability, Stimulation, and Novelty. Each item was rated on a 7-point Likert scale, where 1 indicated a strong preference for the negative attribute and 7 indicated a strong preference for the positive attribute.

After interacting with the HPAP system, participants provided ratings that were consistently skewed toward the positive end of the scale. This indicates that users found the platform not only functional and efficient, but also engaging, clear, and visually appealing. These results align with the high SUS scores and further validate the system's overall usability and design quality.

Such findings are particularly encouraging for a platform intended to serve a diverse user base, including patients and healthcare staff who may have varying levels of digital literacy. The positive UEQ responses suggest that HPAP successfully balances practical utility with emotional appeal, making it both effective and enjoyable to use.

The chart below illustrates the average scores across the UEQ dimensions:

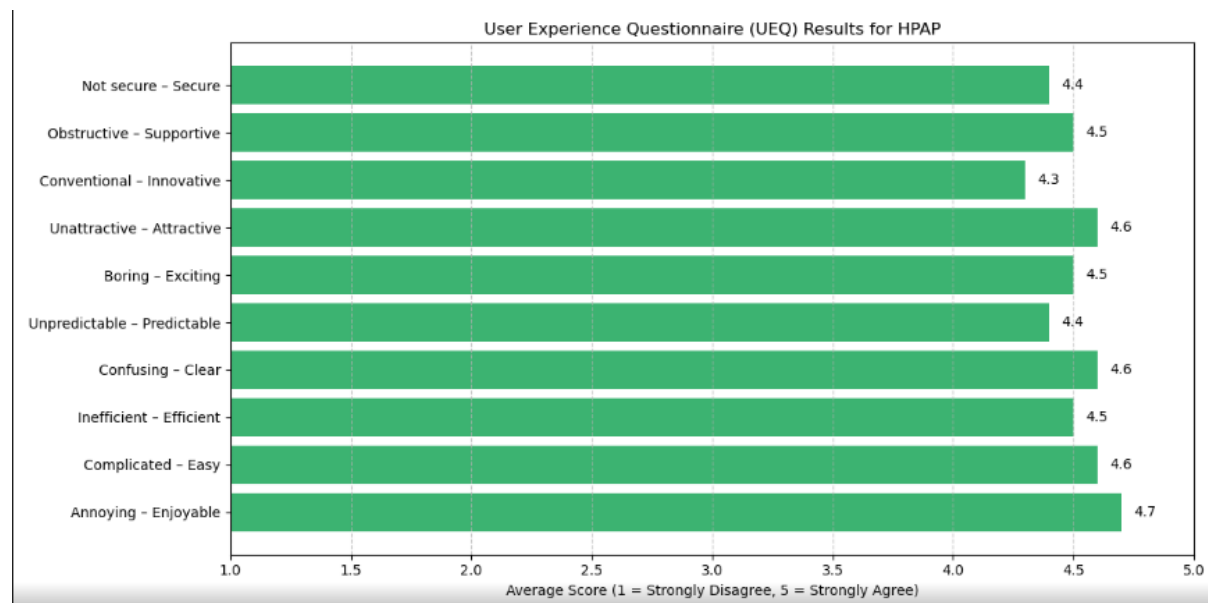


Figure 5-1UEQ Dimension

As shown, all dimensions received scores above 4.3 on a 5-point scale, with particularly strong ratings in Attractiveness, Ease of Use, and Efficiency. These results confirm that the HPAP system delivers a high-quality user experience, supporting its suitability for real-world healthcare environments.

5.7 Discussion

The user feedback collected from the questionnaire was analyzed to determine the overall usability and user satisfaction with the Hospital Patient Appointment Portal (HPAP). The following subsections summarize the key findings from the open-ended responses.

5.7.1 What Features Do You Like Most in HPAP?

Participants highlighted several features they appreciated most about the HPAP system. The appointment booking interface was frequently mentioned, with users noting that it was intuitive, easy to navigate, and efficient for scheduling appointments. Many respondents also praised the real-time doctor availability display, which they found helpful for making quick and informed decisions.

The medical history access feature received positive feedback for its clarity and convenience, allowing users to view past appointments and related information without hassle. Additionally, the system's responsiveness and speed were commended, indicating a smooth and seamless user experience.

Overall, the feedback suggests that users valued both the practical functionality and the user-friendly design of the platform. The combination of clear navigation, responsive performance, and relevant features contributed to a positive perception of the system's usability and effectiveness in a healthcare setting.

5.7.2 Do you have any suggestions for improving the systems?

User responses to the question regarding potential improvements for HPAP were diverse, though most suggestions were relatively minor, indicating overall satisfaction with the system. The feedback was collected from 15 respondents, and each suggestion was mentioned by one or two individuals, reflecting a low frequency of critical issues.

One recurring suggestion was to replace icons with character labels, which appeared in two similar responses. This indicates that some users found icon-based navigation less intuitive and preferred more descriptive, text-based elements for clarity. Another user recommended email integration with pop-up notifications, suggesting a desire for enhanced communication features, particularly for appointment confirmations or reminders.

Other responses included remarks such as “None,” “Zero,” and “No need,” which imply that several users did not perceive any immediate need for changes or enhancements. This reinforces the notion that the system is already meeting user expectations in terms of functionality and usability.

Overall, the suggestions were minimal and specific, pointing to refinements in interface clarity and communication features rather than fundamental changes. These insights are valuable for guiding future iterations of HPAP, ensuring that enhancements align with actual user needs while maintaining the system’s simplicity and effectiveness.

5.7.3 Did You Face Any Challenges While Using HPAP?

User responses regarding challenges or frustrations encountered while using HPAP were generally minimal, indicating a smooth and positive experience for most participants. Out of 15 respondents, a significant portion (20%) provided neutral or non-specific feedback (e.g., “-”), suggesting no major issues were encountered.

However, a few specific challenges were noted. One user mentioned downtime, indicating that the system may have experienced temporary unavailability or slow performance during their session. This highlights the importance of ensuring consistent uptime and server reliability, especially for healthcare-related platforms.

Another recurring theme involved notification handling, with responses such as “closed notifications” and “the UI to closed notifications.” These comments suggest that users may have found the notification system unclear or difficult to manage, pointing to a need for improved design or clearer feedback mechanisms when interacting with alerts.

Other responses such as “Nil,” “Not at the moment,” and “Not yet” further reinforce the overall positive sentiment, as these users did not encounter any significant issues. A couple of entries were partially cut off (e.g., “To complete the...”), but they may indicate minor usability concerns that could be clarified in future feedback rounds.

In summary, while the majority of users did not report substantial challenges, the feedback that was provided offers useful insights into areas for refinement—particularly in system reliability and notification interface clarity. Addressing these points will help enhance the overall user experience and ensure the platform remains intuitive and dependable for all users.

5.8 Implications

The findings from the usability and user experience evaluations of the Hospital Patient Appointment Portal (HPAP) offer several important implications for the development of digital healthcare systems, particularly in the context of improving patient engagement and administrative efficiency in Malaysian healthcare settings.

Users responded positively to the system’s core features, such as appointment booking, department availability tracking, and access to medical history, indicating strong potential for real-world application. These features empower patients to manage their healthcare more proactively and allow medical staff to streamline administrative tasks, ultimately enhancing the overall quality of care.

The system’s high usability scores and positive user experience feedback suggest that HPAP is accessible even to users with limited technical expertise. This is especially significant in healthcare environments where users may vary widely in digital literacy. The intuitive design and responsive interface reduce the need for training or onboarding, making the system suitable for rapid deployment across diverse user groups.

Moreover, the platform’s ability to minimize scheduling errors, improve communication, and reduce waiting times contributes to more efficient resource utilization and better patient satisfaction. These outcomes align with broader goals of digital transformation in healthcare, including improved service delivery, reduced operational costs, and enhanced patient outcomes.

In summary, HPAP demonstrates how well-designed, user-centered digital tools can have a transformative impact on healthcare management. The system’s success in usability and user experience evaluations supports its scalability and adaptability for wider implementation in clinics, hospitals, and community health centers.

5.9 Conclusion

This chapter presented the findings and discussions from the evaluation of the Hospital Patient Appointment Portal (HPAP). The usability and user experience assessments, conducted using the System Usability Scale (SUS) and the User Experience Questionnaire (UEQ), revealed that the system was well-received by users, demonstrating strong usability, effectiveness, and overall satisfaction.

In addition to standardized instruments, custom questionnaire responses provided deeper insights into the system's functionality, relevance, and areas for improvement. Users highlighted the platform's intuitive appointment booking, real-time doctor availability, and access to medical history as particularly valuable features. Suggestions for improvement were minimal and focused primarily on interface clarity and notification handling.

The findings indicate that HPAP has strong potential to enhance healthcare service delivery by offering a user-friendly, efficient, and accessible digital solution for managing patient appointments. However, limitations such as the small sample size, controlled testing environment, and limited demographic diversity should be addressed in future studies to validate these results further.

Overall, the evaluation supports the continued development and deployment of HPAP in real-world healthcare settings. The system contributes to the broader goal of improving digital healthcare infrastructure through user-centered design and technology-driven solutions.

Chapter 6 : Conclusion and Future Work

6.1 Overview

This chapter presents a comprehensive summary of the development and evaluation of the Hospital Patient Appointment Portal (HPAP). It highlights the key conclusions drawn from the research objectives and discusses the broader implications of the findings. Additionally, it outlines potential future enhancements aimed at improving the system's functionality, usability, and real-world applicability. The goal is to encapsulate the project's contributions while laying the foundation for future research and development in digital healthcare systems.

6.2 Conclusions

6.2.1 Conclusion Based on Objective #1

The first objective was to design a process that allows patients with reference letters from other hospitals to directly access specific services, thereby enhancing workflow and reducing wait times. This objective was successfully achieved through the implementation of a streamlined referral system within HPAP. The system enables patients to upload reference documents and bypass redundant registration steps, allowing for faster access to medical services. This feature significantly improves patient flow and reduces administrative bottlenecks, contributing to a more efficient healthcare experience.

6.2.2 Conclusion Based on Objective #2

The second objective focused on developing an integrated system that allows doctors and healthcare providers to easily access and review relevant patient information, including reference letters and prior medical records. This was accomplished through the creation of a centralized dashboard that consolidates patient data in an accessible format. Medical staff can now retrieve and review patient histories, referrals, and appointment details in real time, supporting informed decision-making and improving continuity of care. The integration of

these features demonstrates the system's potential to enhance clinical workflows and reduce information silos.

6.2.3 Conclusion Based on Objective #3

The third objective aimed to implement a solution that provides patients with real-time visibility of available time slots, enabling convenient appointment booking while optimizing hospital resource utilization. This was realized through a dynamic scheduling interface that displays up-to-date availability for doctors and services. Patients can easily select preferred time slots, while the system automatically updates availability to prevent double bookings. This feature not only improves user convenience but also helps hospitals manage resources more effectively.

6.3 Future Work

6.3.1 Integration with National Health Systems

A key direction for future development is the integration of HPAP with national health databases and electronic medical record (EMR) systems. This would allow for seamless data exchange between hospitals, clinics, and government health services, enabling a more unified healthcare experience. Such integration would also support automated verification of reference letters and medical histories, further reducing manual processing.

6.3.2 Expansion of Patient-Centered Features

Future enhancements may include the addition of personalized health dashboards, medication reminders, and follow-up appointment tracking. These features would empower patients to take a more active role in managing their health. By analyzing user behavior and medical history, the system could also offer tailored health tips and alerts, improving patient engagement and outcomes.

6.3.3 Development of a Mobile Application

While the current platform is web-based and mobile-responsive, the development of a dedicated mobile application would significantly improve accessibility. A mobile app could offer push notifications for appointment reminders, real-time updates, and offline access to essential information. Features such as voice-based navigation, multilingual support, and location-based services would further enhance usability, especially for users in rural or underserved areas.

6.4 Chapter Summary

This chapter provided a detailed review of the HPAP system's development, evaluation, and future prospects. The project successfully addressed its core objectives by streamlining patient referrals, enhancing data accessibility for healthcare providers, and improving appointment scheduling. The proposed future enhancements—including integration with national health systems, expansion of patient-centered features, and mobile app development—aim to elevate HPAP into a comprehensive digital health solution. These advancements will support broader adoption and contribute to the ongoing digital transformation of healthcare services in Malaysia.

References

1. Al-Haqwi, A. I., & Al-Shehri, A. M. (2024). Appointment system in primary care: Opinion of consumers and providers. *Journal of Family & Community Medicine*, 14(3), 99. <https://pmc.ncbi.nlm.nih.gov/articles/PMC3410158/>
2. Zhao, P., Yoo, I., Lavoie, J., Lavoie, B. J., & Simoes, E. (2017). Web-based medical appointment systems: A systematic review. *Journal of Medical Internet Research*, 19(4), e134. <https://doi.org/10.2196/jmir.6747>
3. Chodorow, K. (2013). *MongoDB: The definitive guide* (2nd ed.). O'Reilly Media.
4. Kleppmann, M. (2017). *Designing data-intensive applications*. O'Reilly Media.
5. MongoDB Inc. (2024). What is MongoDB? <https://www.mongodb.com/what-is-mongodb>
6. MongoDB Inc. (2024). What is the MERN stack? <https://www.mongodb.com/languages/mern-stack>
7. MongoDB Inc. (2024). MongoDB release notes <https://www.mongodb.com/docs/manual/release-notes/>
8. Kumar, N. (2022). *Full-stack web development with MongoDB, Express, React, and Node*. Packt Publishing.
9. Node.js Foundation. (2023). Node.js releases. <https://nodejs.org/en/about/releases>
10. React.js (2024). Introduction to React. Retrieved from <https://react.dev/>

Appendices

Appendix A : USABILITY TESTING METHOD

Instructions:

On a scale from 1 (Strongly Disagree) to 5 (Strongly Agree), how much do you agree or disagree with the following statements based on your experience with the system?

1. I think that I would like to use this system frequently.

☐ 1 – Strongly Disagree ☐ 2 – Disagree ☐ 3 – Neutral ☐ 4 – Agree ☐ 5 – Strongly Agree

2. I found the system unnecessarily complex. (Reverse-scored item)

☐ 1 – Strongly Disagree ☐ 2 – Disagree ☐ 3 – Neutral ☐ 4 – Agree ☐ 5 – Strongly Agree

3. I thought the system was easy to use.

☐ 1 – Strongly Disagree ☐ 2 – Disagree ☐ 3 – Neutral ☐ 4 – Agree ☐ 5 – Strongly Agree

4. I thought there was too much inconsistency in this system. (Reverse-scored item)

☐ 1 – Strongly Disagree ☐ 2 – Disagree ☐ 3 – Neutral ☐ 4 – Agree ☐ 5 – Strongly Agree

5. I would imagine that most people would learn to use this system very quickly.

☐ 1 – Strongly Disagree ☐ 2 – Disagree ☐ 3 – Neutral ☐ 4 – Agree ☐ 5 – Strongly Agree

6. I found the system very cumbersome to use. (Reverse-scored item)

☐ 1 – Strongly Disagree ☐ 2 – Disagree ☐ 3 – Neutral ☐ 4 – Agree ☐ 5 – Strongly Agree

7. I felt very confident using the system.

☐ 1 – Strongly Disagree ☐ 2 – Disagree ☐ 3 – Neutral ☐ 4 – Agree ☐ 5 – Strongly Agree

8. I needed to learn a lot of things before I could get going with this system. (Reverse-scored item)

☐ 1 – Strongly Disagree ☐ 2 – Disagree ☐ 3 – Neutral ☐ 4 – Agree ☐ 5 – Strongly Agree

9. I think that I would need the support of a technical person to use this system. (Reverse-scored item)

☐ 1 – Strongly Disagree ☐ 2 – Disagree ☐ 3 – Neutral ☐ 4 – Agree ☐ 5 – Strongly Agree

10. I found the various functions in this system were well integrated.

☐ 1 – Strongly Disagree ☐ 2 – Disagree ☐ 3 – Neutral ☐ 4 – Agree ☐ 5 – Strongly Agree

User Experience Questionnaire (UEQ) Items

Instructions:

On a scale from 1 (Strongly Disagree) to 5 (Strongly Agree), how much do you agree with the following paired adjectives based on your experience with the system.

1. Annoying [1 2 3 4 5 6] Enjoyable
2. Complicated [1 2 3 4 5 6] Easy
3. Inefficient [1 2 3 4 5 6] Efficient
4. Confusing [1 2 3 4 5 6] Clear
5. Unpredictable [1 2 3 4 5 6] Predictable
6. Boring [1 2 3 4 5 6] Exciting
7. Unattractive [1 2 3 4 5 6] Attractive
8. Conventional [1 2 3 4 5 6] Innovative
9. Obstructive [1 2 3 4 5 6] Supportive
10. Not secure [1 2 3 4 5 6] Secure

Appendix B : USER MANUAL INSTALLATION KIT

General Information

System Overview

The Hospital Patient Appointment Portal (HPAP) is a full-stack web application designed to streamline healthcare appointment management at UNIMAS Medical Center. Built using the MERN stack (MongoDB, Express.js, React.js, Node.js), the system includes:

- I. Admin Panel – for managing departments, staff, and appointments
- II. Backend API – for handling business logic and database operations
- III. Patient Frontend – for booking appointments, viewing health records, and uploading referral letters

HPAP improves operational efficiency, reduces patient wait times, and enhances the overall healthcare experience.

2. System Requirements

Hardware Requirements

- I. Minimum 4 GB RAM
- II. At least 5 GB free storage
- III. Stable internet connection for API communication and database access

Software Requirements

- I. Node.js (v18 or above) – for backend and frontend servers
- II. MongoDB Atlas – cloud-based NoSQL database
- III. Git – for cloning the repository
- IV. VS Code – or any preferred code editor
- V. Package Managers:
 - npm – for installing Node.js dependencies
 - MongoDB Compass – optional GUI for database inspection

3. Installation Steps

Quick Access (Deployed Version)

HPAP is deployed online for testing and demonstration.

- I. Patient Frontend: <https://hpap-frontend.onrender.com/>
- II. Admin Panel: <https://hpap-admin.onrender.com/>
- III. MongoDB Atlas: Connected via environment variables (not publicly accessible)

Full Local Installation

Step 1: Download and Extract

Extract the .zip file containing the following folders:

- I. project_frontend – React-based patient interface
- II. project_admin – React-based admin dashboard
- III. project_backend – Node.js + Express backend server

Step 2: Setup the Backend

1. Open terminal
2. Navigate to backend folder:

```
cd project_backend  
npm install
```

3. Start backend server:

```
node server.js  
npm run server
```

Step 3: Run the Patient Frontend

1. Open terminal
2. Navigate to frontend folder:

```
cd project_frontend  
  
npm install
```

3. Start frontend server:

```
npm run dev
```

4. Open browser and go to:

```
http://localhost:5173
```

Step 4: Run the Admin Panel

1. Open terminal
2. Navigate to admin folder:

```
cd project_admin
```

```
npm install
```

3. Start admin server:

```
npm run dev
```

4. Open browser and go to:

```
http://localhost:5174
```

4. Using the Application

- I. Patients can register, book appointments, upload referral letters, and view health records
- II. Doctors/Staff can manage schedules, review appointments, and update patient records
- III. Admins can add departments, register staff, and monitor system activity

5. Troubleshooting

- I. Port conflicts: Ensure ports 5173 (frontend), 5174 (admin), and 3000 (backend) are free
- II. MongoDB connection issues: Check .env file for correct MongoDB URI
- III. CORS errors: Verify backend allows requests from frontend and admin origins
- IV. Deployment issues: Ensure environment variables are correctly set on hosting platforms