WEB-BASED CLINIC MANAGEMENT SYSTEM

BY

LIM WEI QI

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION SYSTEMS (HONOURS) BUSINESS INFORMATION SYSTEMS

Faculty of Information and Communication Technology

(Kampar Campus)

JAN 2025

COPYRIGHT STATEMENT

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisors, Dr Zanariah Binti Zainudin who has given me this bright opportunity to engage in a Clinic Management System project. It is my first step to develop a web-based clinic management system. A million thanks to you.

To a very special person in my life, Su Ho Yee Hui, for her patience, unconditional support, and love, and for standing by my side during hard times. Finally, I must say thanks to my parents and my family for their love, support, and continuous encouragement throughout the course.

ABSTRACT

This project focus on the development of web-based clinic management system. The system which are currently used by others clinic, contain a few problems and limitations. Nowadays, the clinic management system may contain some lack of feature such as BMI calculation and record storing system, an effective clinic management system and a well-structured and managed billing and health report system. Therefore, this project and development of system is carried out to resolve the limitation and problem of the clinic management system others clinic currently used. In this project, a real-time appointment system has been carried out to allow patient to make their appointment faster and get confirmation immediately. Additional tools of the clinic management system website also have been carried out such as BMI measuring tools which can always track the user BMI result, make comparison and give the appropriate feedback. Also, the website allow patient to pay their clinic bill online and they can choose their preferred method to pay their bill and ensure safety to pay through online. On the other hand, the system included using incremental methodology to divide the project accordingly. There are a few technologies and tools will be used in this project to carry out a well-structured and managed clinic management system.

Area of Study (Minimum 1 and Maximum 2): Web-based development, Real time system

Keywords (Minimum 5 and Maximum 10): BMI measuring tools, Appointment website, Well-structured system, IoT Security, Web-based, Billing system

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

CMS	Clinic Management System
SDLC	System Development Life Cycle
HTML	Hyper Text Markup Language

Chapter 1

In this chapter, we present the background, introduction, problem statement and contribution to the clinic management system. Through the background of research, users can have an understanding on the advantage of implementing and developing the clinic management system. Through defining the problem statement of this project, user can understand what are the main feature that are going to advance in this project and the limitations of current clinic management systems. While through the contributions, user can know more about those feature and module included in this clinic management system.

Introduction

In the era of technology, where everything needs to be done efficiently and effectively, so there is a need on a system that can managed all user's data and help users to settle their things digitally. Therefore, the existences of Clinic Management System (CMS) become necessary.

A Clinic Management System (CMS) is a comprehensive website designed to streamline and automate the administrative, financial, and clinical aspects of healthcare facilities, such as clinics, medical practices, and outpatient centres. An CMS also includes the network that links these systems, databases, interfaces, physician' order entry, electronic communication systems, and the clinical workstations. [1] The primary purpose of a Clinic Management System is to enhance the overall efficiency and effectiveness of healthcare operations.

However, although many traditional management operations have been digitalized, there are still a lot of public and private sector clinics in Malaysia still using manual operation to manage their user data and operation. This is due to the lack of population of clinic management system in Malaysia. Most of the clinic are not willing to purchase a system for their clinic because they think that it is not a need for a clinic to digitalize their system.

By implementing the online CMS, users can perform their clinic activities and service through online. For example, patient can get access with their medical records

online instead of paper documents, so that they have more understanding on their disease and medical history and encouraging them to use the system. [2] Other than that, with the online CMS, patient can make appointment anytime, so that they can continue their work and no need to visit the clinic just for taking a number and keep waiting for their turn.

Besides, e-payment has been implemented few years ago, it has become a part of our lives. Many systems have enabled e-payment method due to security protection and it takes short time to process the transaction.[3] An online CMS system also support online payment so that no human error is made during processing the transaction, and more secure to protect patient from robbed due to all transaction is make online.[4]

In conclusion, an online Clinical Management System developed in this project is a web-based software designed for registration and management of patient's records and easy access of the records. The system will be used to assist the register, doctors, and admin to store and manage patient records in a hospital or clinic for easier access and reference.

1.1 Problem Statement and Motivation

Nowadays, in appointment scheduling function provided by clinic management system, majority of the clinic management system allow patient to make their appointment by requesting for appointment and then waiting for the clinic respond by contacting them after a period. This feature is not effective enough because the appointment booking is not process in real-time. [5] The patient must wait for respond rather than direct know whether their appointment has been approved. Sometimes this may also cause appointment conflicts because all record is arranged by the admin manually so it may easily cause human-error.

Furthermore, after some research, many of the clinic management website and software has only user for booking and payment purpose. It is just for the patient to visit the clinic to check their health problem, the function of the website has been limited. Other than user who willing to visit the clinic, the website should be develop for another user even they do not plan to visit the clinic physically, which is develop a multi-functional website. For example, BMI calculation feature can be added in the website, patient and other user can always engage with the website to calculate their BMI and follow up their weight and height condition anytime. Sometimes customer

body condition testing in system can provide motivation and interested to user to engage with the system and the user that engage with the system may be increased.

Apart from that, in the financial management, most of the clinic system has lack of online billing system, this may lead to some human errors on the billing data entry manually. Also, a physical billing system may take a lot of time to process the transaction because all process needs to be done manually. For example, key in the customer data one by one to create a new transaction history, manually counting of money and so on. Other than that, there are limited options of billing for the patient if they pay in the clinic. Through online, there are more option of payment, and the patient can select their prefer method to pay their bill.

Research Objective

1.2.1 To provide a real-time appointment system that allow patient to book their appointment immediately

To make this project success, similar system that contain similar appointment will be studied and compared to provide enhanced appointment function for the patient to make an appointment. Other than that, the project will be developed and implement in an online so that patient can check for available appointment time slot and book their appointment real-time without waiting for the confirmation from the clinic admin. Patients can make appointment even outside of their working hours. They no need to apply a leave and travel far to visit the clinic just for making an appointment.

1.2.2 To allow multiple used of the online clinic management system

Other than just for making appointment, pay their bill and view their health report, the online clinic management system is also developed for the user other clinic member, which they can visit the website to measure their health condition and tracking, with some calculation on the website to generate their BMI condition and give some comments and advice. The website also provides some knowledge of the disease for the user to learn and do some prevention, so

that their target customers not just on the clinic patient, also the normal website visitor.

1.2.3 To provide an efficient online billing system

In this project, an online billing system feature will be developed and implemented. The online billing system is developed to speed up the transaction processing process and reduced the human error that may be made by the clinic admin. Besides, with this billing system, patient billing record detailed can be stored in a database and patient can view their transaction history anytime to view their payment details and generate digital receipt. Other than that, the clinic can track patient billing record through the system to determine the income of the clinic monthly or yearly for financial purposes.

1.3 Project Scope and Direction

This project aims to develop and implement a web-based online clinic management system for all the users. Apart from that, this system will be developed and implemented with different function for the three types of users, such as patients, doctors and admin. The main functions in this system include real-time appointment booking, online users BMI calculation, online billing system, and patient health report management. In addition, the system will also develop and implement some additional features to provide users with more comprehensive services, such as a simple chatbot and appointment reminders.

1.3.1 Real-time Appointment Booking

The appointment system allows the patient to view the available time slot to make an appointment. If the time slots are full, it will automatically close the time slot so that the patient cannot select that time slot for booking. After booking successfully, the system will record the patient appointment and display to the admin and doctor at the same time.

1.3.2 BMI Calculation and Follow-up

The BMI calculation system allows all users to measure their BMI and have an understanding on their health condition. The system will store the record and time, allow user to view their pass record to see whether their health condition is improving.

1.3.3 Online Billing System

The online billing system allows patient to choose their preferred payment method to pay their bill. The invoice and record will automatically generate when the patient complete the payment.

1.3.4 Appointment Reminder for patient

Before few days from patient appointment day, a reminder will be sent through patient email to remind them regarding their appointment

1.4 Contributions

In this project, we aim to develop a user-friendly, efficient, and well managed Clinic Management System to help user make their appointment online, know more knowledge about some disease, measure their BMI and pay their medical bill online. There are a few features need to be included in this project. First and foremost, login feature is needed to be included in this clinic management system. There are a few logins type, which are doctor login, customer login and admin login.

For the clinic patient, they can view their health report anytime, print it out for storing purpose, and view their billing record anytime. Real-time appointment features also make the appointment process faster and easier. Patient only need to fill in their information and choose the available time slot and confirm the booking time. They no need to worry that they will forget that date and time because a reminder may send to their email to remind them that their appointment date is almost near to reach so that they may not forget their appointment due to busy work or others factor.

Furthermore, BMI measuring page. In this page, user that visit the website can calculate their BMI result by entering their weight and height so that they can get their result and save it to be view in next time. Under this feature, user can track their BMI record anytime in anyplace to monitor their health condition. Some comment will be

generated to the user based on their BMI result so that they can understand how to improve their body health and prevent obesity. They can always follow up their health conditions to identify whether their health conditions are increasing or decreasing.

From admin point of view, the system will provide the patient detail personal information for the recording purpose and display the total appointment record with the selected period. Apart from that, the admin can generate patient bill for them to pay online and view patient billing record. Admin also will be responsible to manage the doctor information and approve the doctor sign up account request to make sure the account is a doctor, not others.

1.5 Report Organization

The details of this research are shown in the following chapters. In Chapter 2, some similar systems are reviewed. The features, strength and weaknesses from different system will be listed. Also, there is a comparison made between the system and proposed system. Methodology, hardware and software used to implement the proposed project are described in Chapter 3. Then, a system design of the proposed system was presented in Chapter 3. In system design included ERD Diagram, use case diagram, use case description, activity diagram and sequence diagram. And then, Chapter 4 describes preliminary work of the proposed system. Finally, Chapter 5 will show the conclusion of the entire project.

Chapter 2 Literature Review

2.1 Previous works on Web Based Clinic Management System

In this chapter 2, a relevant literature will be conduct on existing Clinic Management System will be reviewed. During the literature review, many useful functions will be discovered in existing system. For example, their appointment system, billing system and some additional function for the website visitor. Other than that, in this chapter, the functions and characteristics between these similar systems also will be discovered.

In order to determine the limitations and useful function of those existing system used by other clinic, the similar online clinic system such as Mayo Clinic, UR Klinik, and Care Clinics. The purpose of conducting system review is to obtain the strength and limitations of each clinic management system mentioned above. In addition, through literature review, this project is able to purpose more comprehensive function in the system.

2.1.1 UR Klinik

Figure 2.1.1.1 UR Klinik

UR Klinik is a clinic that specialized in skin pigmentation treatment using the latest, advanced technologies in the aesthetic industry. URSKIN has inspired 5,000 customers nationwide to be beautiful on the inside and out, as part of their result-proven journey of becoming the No. 1 Skin Pigmentation Expert brand in Southeast Asia. For appointment scheduling function, UR Klinik has resolved this problem by adding a

feature for their customer to make appointment. Customer are required to enter their personal information, they preferred appointment date and time, preferred location and submit it. After that, the customer will receive a call or message after few days. To confirm their appointment date. Appointment System UR Klinik develop can save customer time since they can make appointment online without going to the clinic just for making an appointment. [5]

2.1.1.1 UR Klinik Function

There are a few functions provided by the UR Klinik to their user. First, they provide user to register for a user account, and also provide user ability to search for clinic by displaying clinic location and their contact information. Oher than that, they provide users ability to find for treatment and make an appointment.

2.1.1.2 UR Klinik Features

UR Klinik support a Waze feature that directs customers to their nearest branch, allow users to choose their desired treatment, and provide an option to direct them to WhatsApp for further inquiries. Additionally, it enables users to view detailed information about each treatment, helping them identify which option is most suitable for their specific skin condition. They also have the features to display google review and customer feedback

2.1.2 Strengths and Weakness

2.1.2.1 UR Klinik Strength

The website is designed to be simple and informative, providing customers with essential details about the clinic, including comprehensive descriptions of each treatment offered. It features a variety of customer reviews with diverse comments to build credibility and include different policies to enhance customer trust. Additionally, the website has integrated Waze, allowing customers to be directly navigated to the clinic via the Waze app when they click on the address.

2.1.2.2 UR Klinik Weaknesses

The UR Klinik system currently lacks several key features that could enhance the customer experience. It does not offer a skin report function, preventing customers from viewing their records. The appointment functionality is limited, relying on a

basic form submission process with no efficient booking or confirmation mechanism. Additionally, the system lacks advanced features such as online skin condition testing and does not provide an effective chat function for real-time user support. These shortcomings reduce the overall usability and efficiency of the system, potentially affecting customer satisfaction.

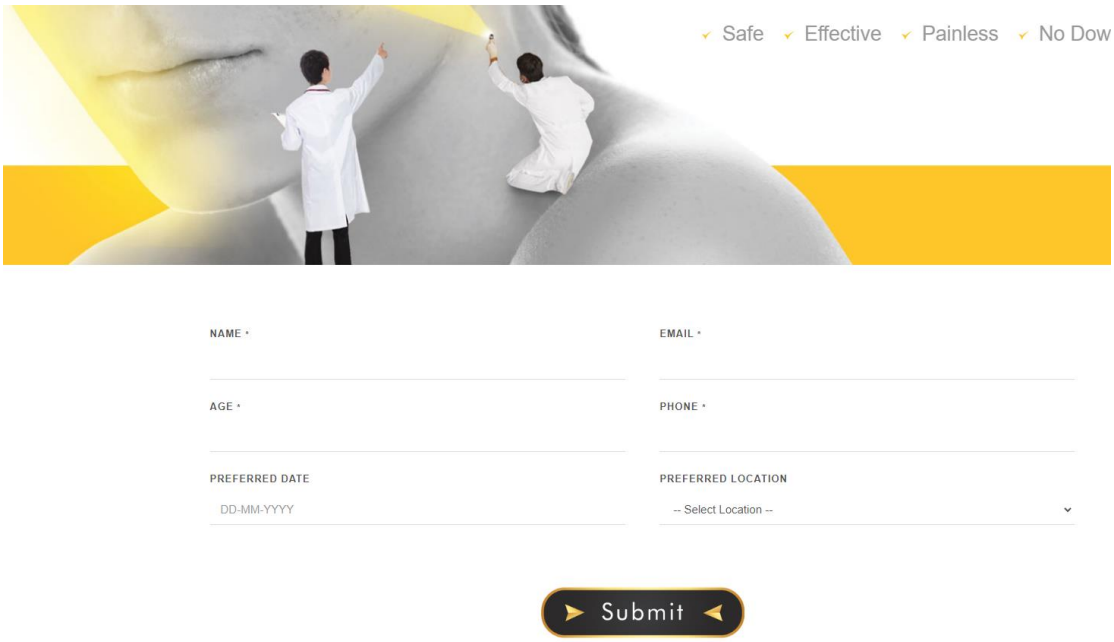


Figure 2.1.2.2.1 UR Klinik Appointment System

2.1.2 Mayo Clinic

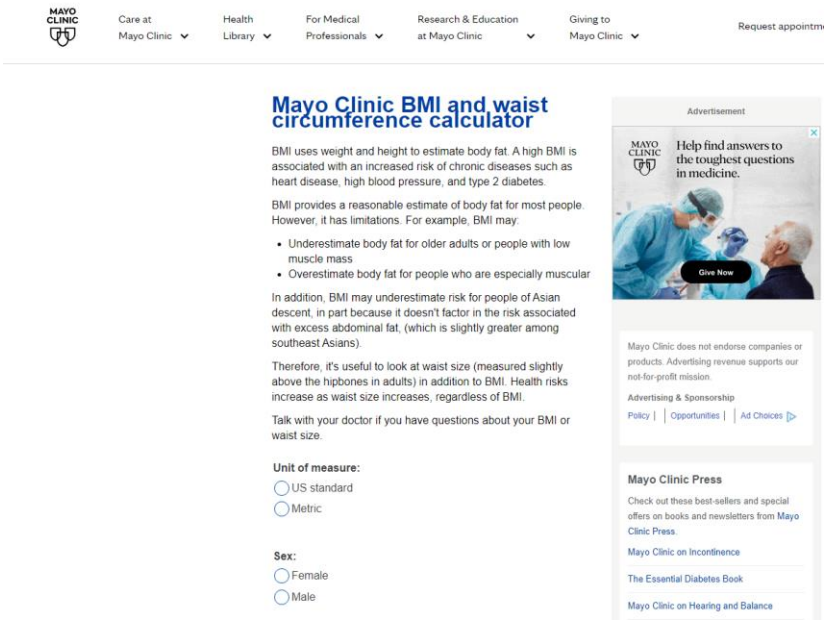


Figure 2.1.2 Mayo Clinic

The Mayo Clinic is a world-renowned nonprofit medical practice and research group that operates in multiple locations across the United States. Founded in 1864 in Rochester, Minnesota, the Mayo Clinic has grown to become a leader in patient care, medical research, and education. The Mayo Clinic is known for its patient-centred approach, providing high-quality medical care across a wide range of specialties. Patients from around the world visit Mayo Clinic for its expert consultations and advanced treatment options. [6]

2.1.2.1 Mayo Clinic Function

The system has included appointment scheduling features to allow users to book and manage their appointments efficiently. It also provides billing and payment services for seamless transactions. Additionally, integrating tools such as BMI and waist circumference calculators would offer users valuable health insights. The platform also is also enriched with extensive health information details, making it a comprehensive resource for users seeking guidance on various health topics.

2.1.2.2 Mayo Clinic Features

Mayo Clinic offers a comprehensive range of features designed to enhance patient care and streamline health management. The appointment request process is detailed, requiring patients to submit their medical documents before booking an appointment, ensuring that the clinic has all necessary information upfront. Patients can view their bills, pay online, and manage their payment history through a secure portal, providing convenience and transparency in billing. The system also allows patients to securely access their electronic health records (EHR), which include lab results, imaging, prescriptions, and visit summaries, enabling them to stay informed about their health. Additionally, Mayo Clinic offers an easy-to-use online BMI calculator on their website, where users can input their height and weight to calculate their BMI, providing a quick and accessible tool for health assessment.

2.1.2.2 Mayo Clinic Strength

The Mayo Clinic website is designed with a user-friendly interface that makes it easy for users to search and find the information they need. Patients can conveniently request, reschedule, or cancel appointments directly through the website, streamlining

the process and offering flexibility. The site also offers a vast array of educational materials, including detailed guides on specific diseases, treatment options, and preventive care, helping users stay informed about their health. Furthermore, the website features various interactive tools such as BMI calculators, heart disease risk assessments, and pregnancy trackers, allowing users to actively engage with and manage their health information.

2.1.2.3 Mayo Clinic Weaknesses

The extensive and detailed content of Mayo Clinic systems can lead to information overload, especially for users who are looking for quick, straightforward answers. With such a large repository of medical content, there is a possibility that some articles or resources may not be regularly updated. Other than that Website offers a vast amount of content, which can be overwhelming for new users or those who are not familiar with medical terminology.

2.1.3 Care Clinic

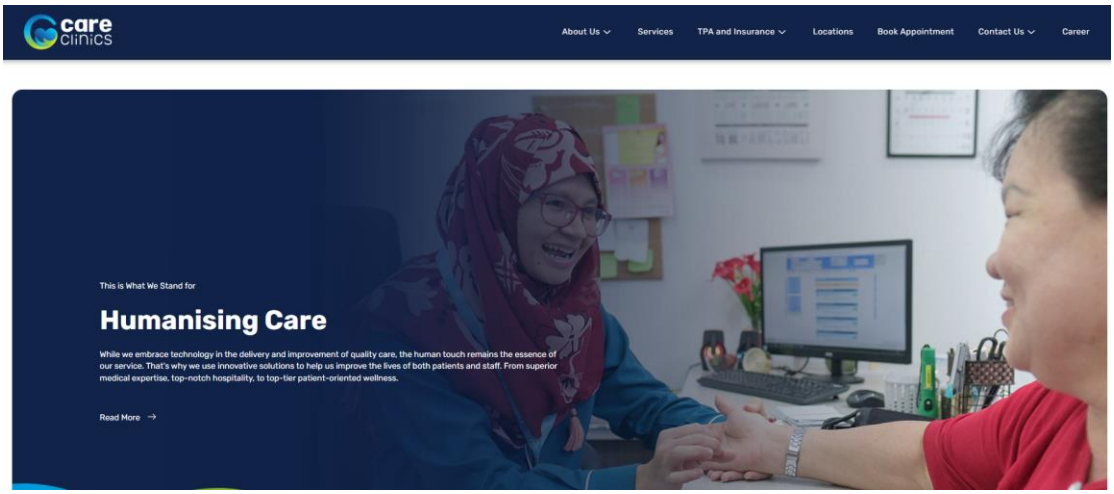


Figure 2.1.3.1 Care Clinic

Care Clinics Malaysia is a network of healthcare facilities offering a wide range of medical services across Malaysia. These clinics are designed to provide accessible, high-quality primary care to individuals and families, serving as the first point of contact for patients seeking medical attention. With multiple locations across the country, Care Clinics Malaysia aims to make healthcare accessible to as many people as possible. The clinics often offer extended hours and walk-in services, making it easier for patients to receive care when needed.[7]

2.1.3.1 Care Clinics Function

Care Clinics website offers a comprehensive directory of clinic locations, complete with addresses and operating hours, making it convenient for patients to locate and visit the nearest clinic. Patients can easily book appointments by filling out and submitting a form directly to the clinic, streamlining the scheduling process. The site also provides easy access to contact details for each clinic, including phone numbers, email addresses, and contact forms, ensuring that patients can reach out with any questions or concerns. Additionally, the website is integrated with the clinic's social media profiles, allowing users to follow updates and interact with the clinic on platforms like Facebook, Instagram, and Twitter. This integration helps maintain engagement and keeps patients informed about the latest news and events.

2.1.3.1 Care Clinics Features

The website has featured a dedicated space where patients can leave reviews and read feedback from others about their experiences, fostering transparency and trust. It also includes forms for general inquiries, appointment requests, or specific issues, making it easy for users to communicate with the clinic. Additionally, an interactive map or search tool would be beneficial, allowing users to find the nearest Care Clinic locations based on their address or current location, enhancing convenience and accessibility.

2.1.3.1 Care Clinics Strength

Care Clinics offer several strengths that enhance the patient experience and build a reliable healthcare environment. The ability for patients to leave reviews and feedback fosters trust and transparency, allowing future patients to make informed decisions based on others' experiences. The clinic provides multiple pages detailing different insurance options, ensuring that patients can find coverage that suits their needs. A well-organized library of articles, guides, and resources is available, offering valuable information on various health topics to help patients make educated decisions about their care. Additionally, Care Clinics offer various contact methods, including phone, email, and live chat, ensuring that users can easily reach out for support or inquiries, enhancing accessibility and patient satisfaction.

2.1.3.1 Care Clinics Weaknesses

There are few weaknesses contain in Care clinics website. The lack of Billing feature will be forcing all patient must pay their bill physically. Also, there are lack of searching function and patient report generator. Other than that, they have appointment booking feature. However, inefficient appointment system without real-time booking will lead to time consuming.

2.2 Comparison Between Similar System

Functions	UR Klinik	BMI Clinic	Care Clinics
Login and Logout	Yes	No	No
Appointment Booking	Yes	Yes	Yes
Online Billing	No	Yes	No
Chatbot or Live Chat	No	No	No
User Feedback	No	Yes	Yes
Additional tools	No	Yes	No
Patient Health Report Generator	No	Yes	No
Additional Information and Health Knowledge	Yes	Yes	Yes

Table 2.2.1 Comparison between similar system

Chapter 3 System Methodology/Approach

The processes of the project were categorized into different phases in the development, which were project pre-development, data pre-processing, model training architecture building and data training, and prediction on test dataset.

3.1 System Requirement

3.1.1 Hardware

The hardware involved in this project is computer and mobile devices. A computer issued for the process of coding and implementing the result of the coding in website. Other than that, a mobile device is used for testing and implementing the website in mobile form to make sure the page is well fixed in the mobile phone website.

Table 3.1.1.1Specifications of laptop

Description	Specifications
Model	Huawei MateBook D15
Processor	AMD Ryzen 5 3500U with Radeon Vega Mobile Gfx 2.10 GHz
Operating System	Windows 10
Memory	8GB RAM
Storage	168GB

3.1.2 Software

Software	Specification	Description
Operating System	Window 10	Used to provide services for laptop computer programs.
Web Browser	Google Chrome	Used to display system output
Visual Studio Code	Version 2019	Used to develop the project system
Front-End	HTML, CSS, Bootstrap, JavaScript, and PHP	Used as a development tool for the website's graphical user interface
Database	MySQL Database	Used to store users and appointment data

Table 3.1.2.1 Software Use

The operating system implemented in this laptop is Window 10, which is used to provide services for laptop programs, such as website programs. Besides, the web browser used to display system output is Google Chrome. Next, this project will use Visual Studio Code 2019 version for development and implementation. This project will use HTML, CSS, Bootstrap, JavaScript, and PHP as the development tools for the website's graphical user interface. In addition, MySQL will be used to store user and inventory data.

3.2 System Design Diagram

3.2.1 ERD Diagram

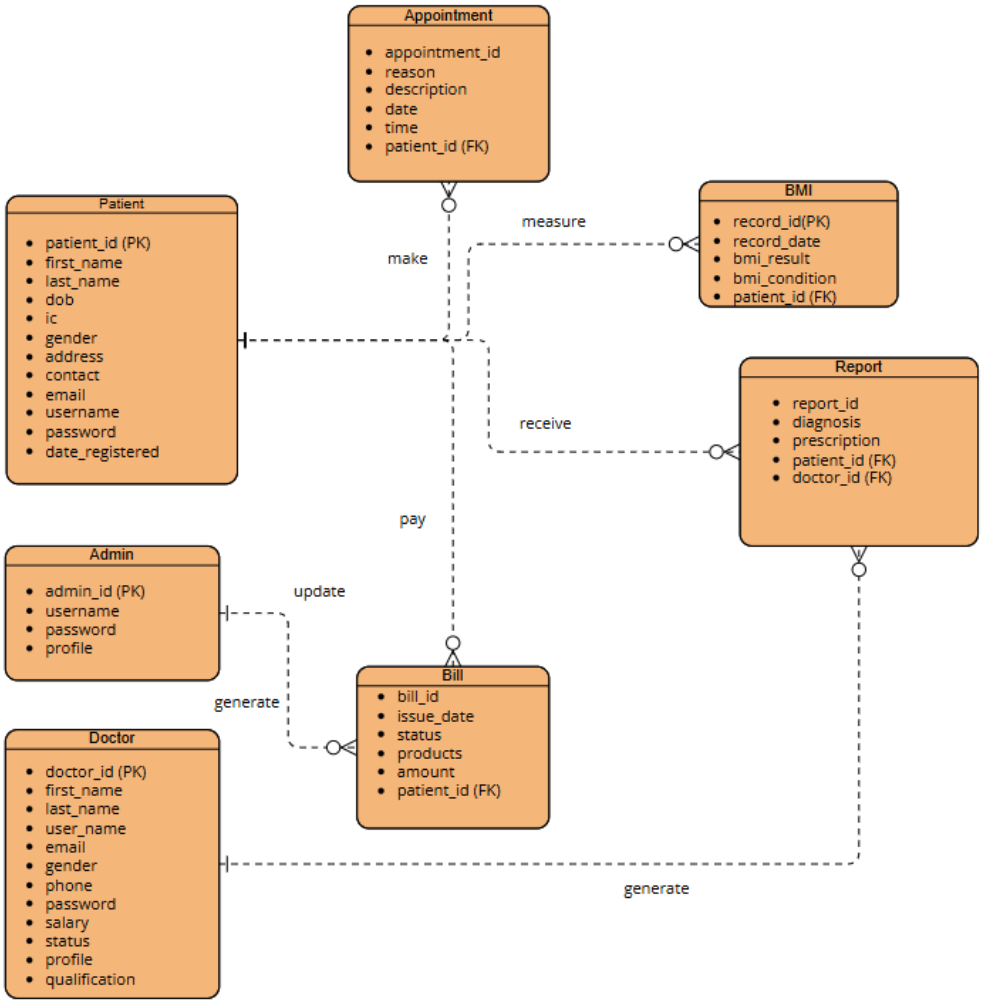


Figure 3.2.1.1 ERD Diagram

3.2.2 Use Case diagram

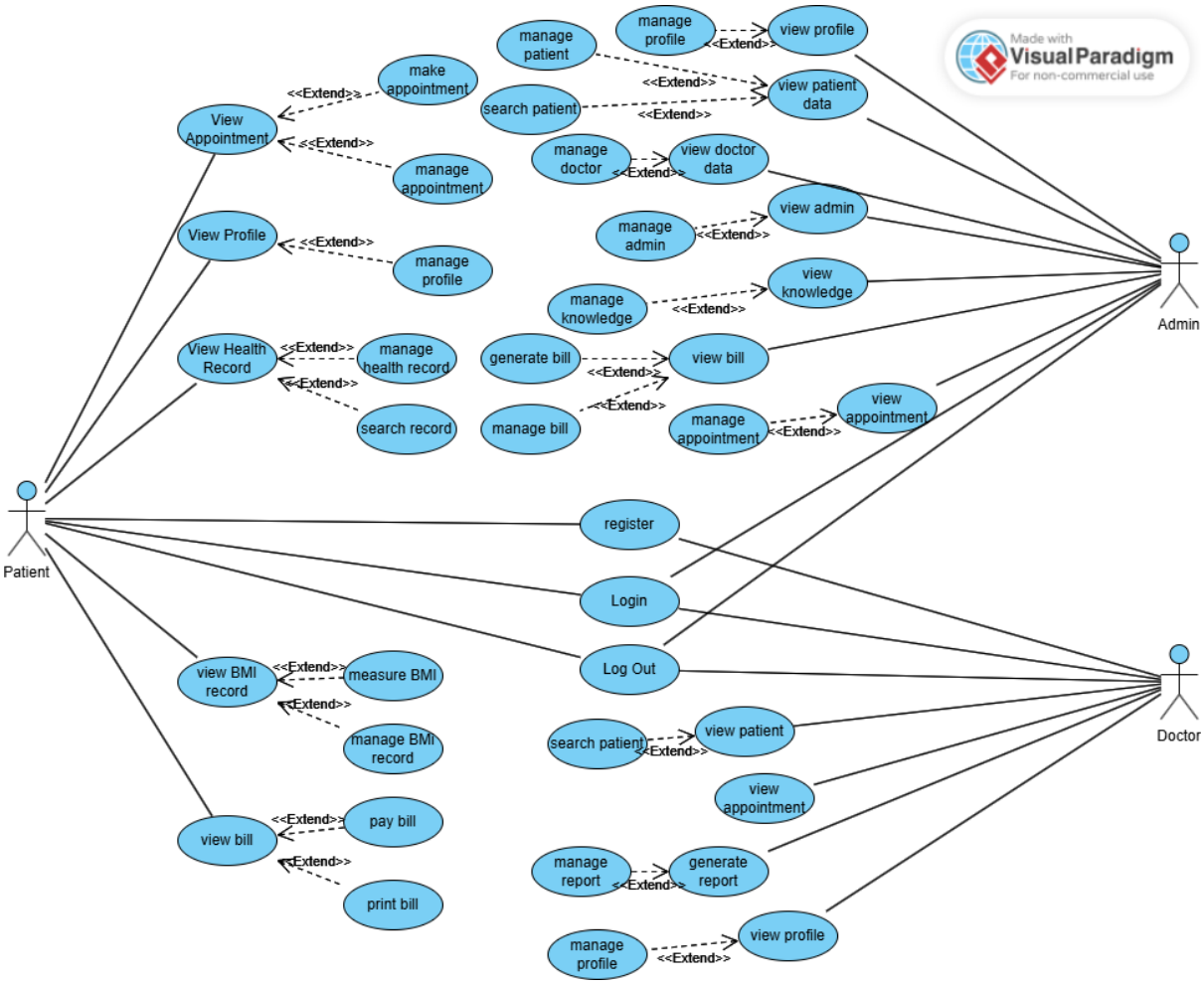


Figure 3.2.2.1 Use Case Diagram

3.2.3 Use Case Description

3.2.3.1 View Appointment – Patient Perspective

USE CASE DESCRIPTION
Name – View Appointment
Brief Description – To Allow Patient View Appointment Detail and Manage his/ her Appointment
Actors – Patient
Relationships – **Association:** **Include:** **Extends:** Manage Appointment **Generalization:**
Preconditions – 1. Patients had login into his/her account 2. Patients had performed appointment booking
Basic Flow – 1. Patients click on Appointment button in main page 2. System will display patient appointment 3. Patients click on the appointment button to view appointment detail 4. System will display the appointment detail
Alternate Flows – If patients edit their appointment 1. Patients click on the edit button 2. System will change the input field become editable 3. Patients edit the field 4. Patients click on the submit button 5. System will receive the user input and validate user input 6. System will check on the appointment schedule 7. System will update the database record 8. System will prompt out an appointment detail changed successfully message If patients delete their appointment

1. Patients click on the delete button 2. System will display delete confirmation message 3. Patients click the yes button 4. System will delete the database record
Exception Flows – Incorrect user input
Post Conditions – 1. Patient able to view appointment detail 2. System able to update/ delete the database record and display success message

Table 3.2.3.1.1 View Appointment Use Case Description

3.2.3.2 View Health Report – Patient Perspective

USE CASE DESCRIPTION
Name - View Health Report
Brief Description – To Allow Patient View Health Report and Print his/her report
Actors - Patient
Relationships – **Association:** **Include:** **Extends:** Print Report **Generalization:**
Preconditions – 1. Patients had login into his/her account 2. Doctor has generated report
Basic Flow – 1. Patients click on Health Report button in main page 2. System will display patient health report
Alternate Flows – If user search for report 1. Patients click on the search button 2. Patients select the appropriate date 3. Patients click on the report 4. system display patients report If user print their health report 1. Patients click on the print button 2. System will display print window 3. Patients print the report
Exception Flows –
Post Conditions – Report will print successfully

Table 3.2.3.2.1 View Health Report Use Case Description

3.2.3.3 View Bill – Patient Perspective

USE CASE DESCRIPTION
Name – View Bill
Brief Description – To Allow Patient View Their Bill
Actors - Patient
Relationships – **Association:** **Include:** **Extends:** Pay Bill, Print Bill **Generalization:**
Preconditions – 1. Patient has login into his/her account 2. Admin has generated bill
Basic Flow – 1. Patients click on the bill button 2. Patients look for the unpaid bill 3. Patients click on the bill 4. System displays bill information
Alternate Flows – If user pay the bill 1. Patients click on the pay button 2. Patients select payment method 3. Patients click on the pay button 4. system direct to the payment page If user print their bill 1. Patients click on the print button 2. System will display print window 3. Patients print the bill
Exception Flows –
Post Conditions – **1.** Bill will successfully pay 2. Bill will print successfully

Table 3.2.3.3.1 View Bill Use Case Description

3.2.3.4 View Personal Information – Patient Perspective

USE CASE DESCRIPTION
Name – View Personal Information
Brief Description – To Allow Patient View Their Profile
Actors - Patient
Relationships – **Association:** **Include:** **Extends:** Manage Information **Generalization:**
Preconditions – 1. Patient has login into his/her account
Basic Flow – 1. Patients click on Profile button in main page 2. System will display patient profile detail 3. Patients click on the appointment button to view profile information
Alternate Flows – If patients edit her profile 1. Patients click on the edit button 2. The system will prompt out editable input field 3. Patients edit their information 4. Patients click save button 5. System update the database
Exception Flows –
Post Conditions – 1. Patient Detail changed successfully

Table 3.2.3.4.1 View Personal Information Use Case Description

3.2.3.5 View BMI Record– Patient Perspective

USE CASE DESCRIPTION
Name - View BMI Record
Brief Description – Allow Users to View their BMI record
Actors - Patient
Relationships – **Association:** **Include:** **Extends:** Add BMI measure, Edit BMI Record **Generalization:**
Preconditions – Users has log in into his / her account
Basic Flow – 1. Users click into their BMI button in main page **2.** System will display the historical BMI record
Alternate Flows – If users add BMI measure 1. Users click on add new record button 2. Users key in their body height and weight 3. Users click generate result button 4. System will display result with advice and recommendation 5. Record will be automatically saved to database
Exception Flows – Lack of compulsory input
Post Conditions -

Table 3.2.3.5.1 View BMI record Use Case Description

3.2.3.6 Book Appointment – Patient Perspective

USE CASE DESCRIPTION
Name – Book Appointment
Brief Description – Allow Patient to make an appointment
Actors - Patient
Relationships – **Association:** **Include:** **Extends:** - **Generalization:**
Preconditions – Users has log in into his / her account
Basic Flow – 1. Patients click the appointment option in the main page 2. Patients select the book appointment button 3. System will display a calendar for patients to select the appointment date and time 4. After select, patients need to fill in their appointment information 5. System prompt out booking confirmation 6. Patients click on confirm 7. An appointment will be made, and database will be updated
Alternate Flows –
Exception Flows – Lack of compulsory input
Post Conditions -

Table 3.2.3.6.1 Book Appointment Use Case Description

3.2.3.7 User Login

USE CASE DESCRIPTION
Name - Login
Brief Description - Log in Users to their page
Actors – Patients, Doctors**,** Admin
Relationships – **Association:** **Include:** **Extends:** **Generalization:**
Preconditions – Users had registered their account
Basic Flow - 1. Users click on log in button 2. Users enter his/ her username and password 3. Users click the log in button **4.** System direct users to their main page
Alternate Flows -
Exception Flows – Incorrect user input, username not found
Post Conditions – Users able to login their page successfully

Table 3.2.3.7.1 User Login Use Case Description

3.2.3.8 User Logout

USE CASE DESCRIPTION
Name - Logout
Brief Description – Log out users from their session
Actors - Patients, Doctors, Admin
Relationships – **Association:** **Include:** **Extends:** **Generalization:**
Preconditions – users had log in to their session
Basic Flow – 1. Users click on log out button 2. System will log user out from their session **3.** System direct user to the home page
Alternate Flows -
Exception Flows -
Post Conditions – System able to log user out from their session

Table 3.2.3.8.1 User Logout Use Case Description

3.2.3.9 View profile – Admin Perspective

USE CASE DESCRIPTION
Name – View profile
Brief Description – Allow admin to view and edit their profile
Actors - Admin
Relationships – **Association:** **Include:** **Extends:** Edit Profile **Generalization:**
Preconditions – Admin has log into his/her account
Basic Flow – 1. Admin click the profile button in main page **2.** System displays admin profile to the admin
Alternate Flows – If admin click the edit button 1. Admin click the edit button 2. System displays editable input field for admin to edit their information 3. After admin edit, click the save button 4. System display save successfully 5. Database will be updated
Exception Flows – Incorrect user input
Post Conditions – Information can display, and database will be updated after admin update

Table 3.2.3.9.1 View Profile Use Case Description

3.2.3.10 View Patient Data – Admin Perspective

USE CASE DESCRIPTION
Name – View Patient Data
Brief Description – allow admin to view patient data and manage them
Actors - Admin
Relationships – **Association:** **Include:** **Extends: M**anage patient **Generalization:**
Preconditions – Admin are login into their account
Basic Flow – 1. Admin click on the patient option in the main page 2. A list of patient data will be displayed
Alternate Flows – If admin remove the patient 1. Admin click on the remove button beside patient data 2.. Patient Data will be removed 3. System databased will be updated
Exception Flows -
Post Conditions – Patient Information can display and removed from database

Table 3.2.3.10.1 View Patient data Use Case Description

3.2.3.11 View Admin – Admin Perspective

USE CASE DESCRIPTION
Name – View Admin
Brief Description – Admin can view others admin information and manage
Actors - Admin
Relationships – **Association:** **Include:** **Extends:** manage admin **Generalization:**
Preconditions – Admin are login into his/ her account
Basic Flow – 1. Admin click on the admin button in the main page 2. System will display all admin information to admin
Alternate Flows – If admin add new admin 1. select add admin button 2. Fill in admin information 3.. click the add button 4. A new admin will be added 5. database will be updated If admin remove admin 1. select remove button 2. admin will be removed 3. database will be updated
Exception Flows -
Post Conditions – Admin will be displayed, and admin are able to add new admin and remove existing admin

Table 3.2.3.11.1 View admin Use Case Description

3.2.3.12 View Bill – Admin Perspective

USE CASE DESCRIPTION
Name – View Bill
Brief Description – Allow admin to view patient bill
Actors - Admin
Relationships – **Association:** **Include:** **Extends:** Manage patient bill **Generalization:**
Preconditions – Admin is login into his/her account
Basic Flow – 1. Admin click on the patient bill button in the main page 2. A list of patient id with their bill will be displayed 3. Admin click on the patient bill 4. The bill information will be displayed to admin
Alternate Flows – If admin add patient bill 1.click on the add patient bill 2. key in patient id and the bill detail 3. click on add bill button 4. a bill will be added in the database
Exception Flows – Lack of necessary information
Post Conditions – 1. Billing able to display to admin successfully 2. Admin able to add new bill for the patient

Table 3.2.3.12.1 View Bill Use Case Description

3.2.3.13 View Appointment – Admin Perspective

USE CASE DESCRIPTION
Name – View Appointment
Brief Description – Allow Admin to view those appointment and manage
Actors - Admin
Relationships – **Association:** **Include:** **Extends:** Manage appointment **Generalization:**
Preconditions – Admin is log in into his/her account
Basic Flow – 1. Admin click the button of appointment in the appointment page 2. System will prompt out a list appointment 3. Admin click the appointment to view details 4. System display the appointment
Alternate Flows – If admin edit appointment 1. Admin click on edit appointment button 2. Appointment details will be showed in adjustable field 3. Admin edit the details 4. Click the save button 5. Appointment will be edit successfully and save to database
Exception Flows – Incorrect Input
Post Conditions – Appointment can be viewed and edited

Table 3.2.3.13.1 View Appointment Use Case Description

3.2.3.14 View Patient – Doctor Perspective

USE CASE DESCRIPTION
Name – View Patient
Brief Description – allow doctor to view patient and his/her details
Actors - Doctor
Relationships – **Association:** **Include:** **Extends:** search patient **Generalization:**
Preconditions – Doctors are log in into his/her account
Basic Flow – 1. Doctor click on patient button in the main page 2. System will show all patient details 3. Doctor click on patient list to view details
Alternate Flows -
Exception Flows -
Post Conditions – Patient details displayed successfully

Table 3.2.3.14.1 View Patient Use Case Description

3.2.3.15 View Appointment – Doctor Perspective

USE CASE DESCRIPTION
Name – View Appointment
Brief Description – Allow Doctor to view his/ her appointment
Actors - Doctors
Relationships – **Association:** **Include:** **Extends:** **Generalization:**
Preconditions - Doctors are log in into his/her account
Basic Flow – 1. Doctor click on appointment in the main page 2. System display doctor appointment 3. Doctor click on appointment to view details
Alternate Flows -
Exception Flows – Doctors has log into his/her account
Post Conditions – All appointment can display

Table 3.2.3.15.1 View Appointment Use Case Description

3.2.3.16 Generate Report – Doctor Perspective

USE CASE DESCRIPTION
Name – Generate Report
Brief Description – Allow Doctor to generate patient health report
Actors - Doctor
Relationships – **Association:** **Include:** **Extends:** Edit Report **Generalization:**
Preconditions – Doctor is log into his/her account
Basic Flow – 1. Doctor click on Report button in the main page 2. Doctor click on generate new report 3. Doctor fill in report details 4. Doctor click submit button 5. A new report will be generated and update to patient and admin
Alternate Flows – If Doctor edit report 1. Click edit button beside the report 2. Edit the content 3. Click save report 4. Report will be updated, and database will be updated
Exception Flows – invalid patient id
Post Conditions – All report can be generated and can be edited by doctor

Table 3.2.3.16.1 Generate report Use Case Description

3.2.3.17 View Profile – Doctor Perspective

USE CASE DESCRIPTION
Name – View profile
Brief Description – Allow doctor to view his/ her profile
Actors - Doctor
Relationships – **Association:** **Include:** **Extends:** Manage profile **Generalization:**
Preconditions – Doctor has log into his/her account
Basic Flow – 1. Doctor click on profile button in main page 2. System display doctor profile
Alternate Flows – If doctor edit profile 1. Doctor click on edit button 2. System display editable input field 3. Doctor edit his/ her profile 4. Doctor click save button 5. Profile successfully edited and updated
Exception Flows – incorrect input format
Post Conditions – profile can be display and edit successfully

Table 3.2.3.17.1 View Profile Use Case Description

3.2.4 Activity Diagram

3.2.4.1 View Appointment – Patient Perspective

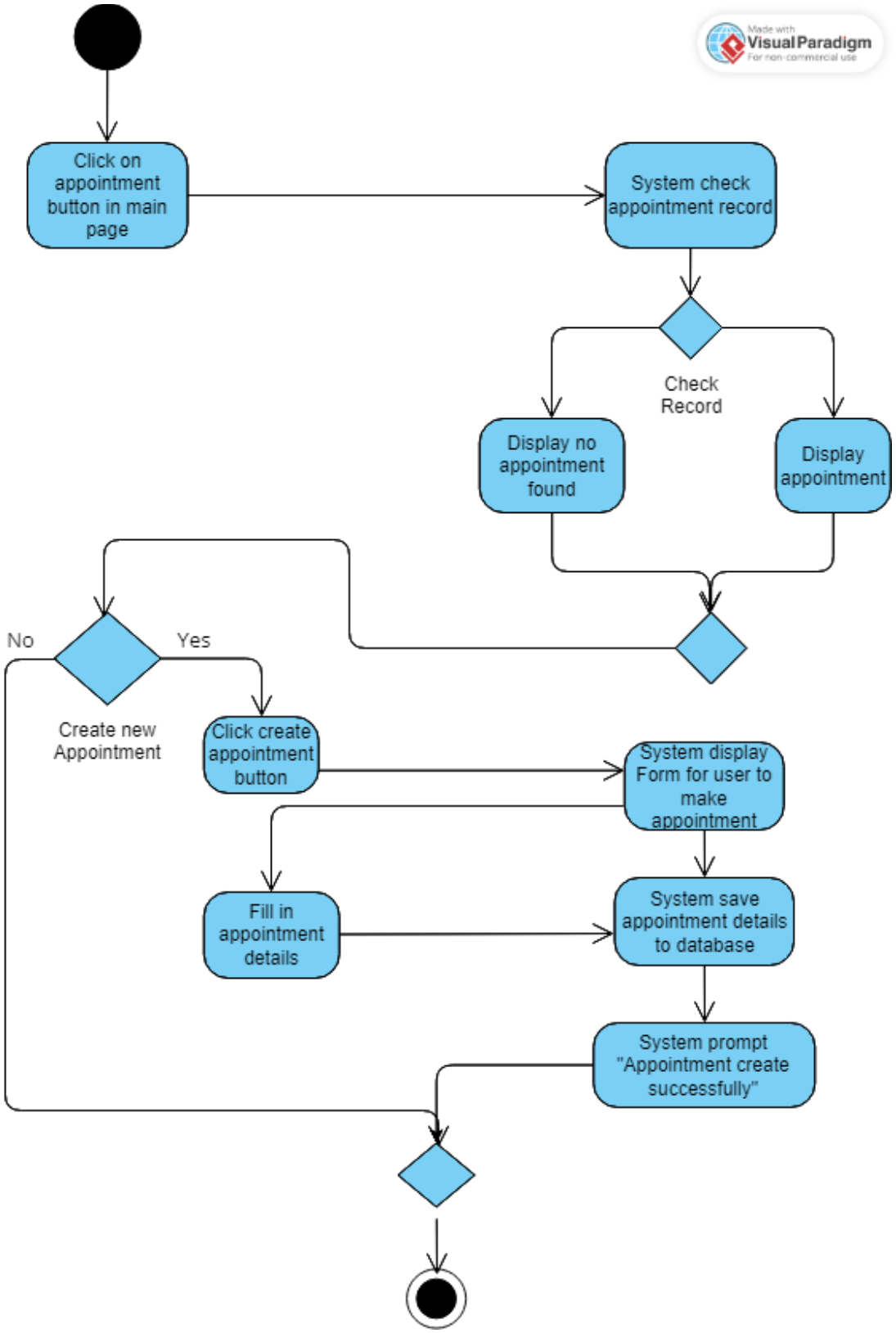


Figure 3.2.4.1.1 View Appointment Activity Diagram– Patient Perspective

Based on the activity diagram above, if patient willing to view their appointment and make an appointment, they have to click the appointment button in the main page. After clicking, in backend, system will check the patient appointment record based on its patient id. If here is not any record, system will show “no appointment record” in the appointment page. If patient want to make an appointment, in this page, there will be a create appointment button. After clicking, system will direct user to the appointment page and a calendar will be showed to the patient, patient need to first select the day and time slot to make appointment, fill in patient information such as disease, symptoms and descriptions. After done, patient click the make appointment button, a new appointment will be created, and system backend database will be updated at the same time.

3.2.4.2 View Health Report – Patient Perspective

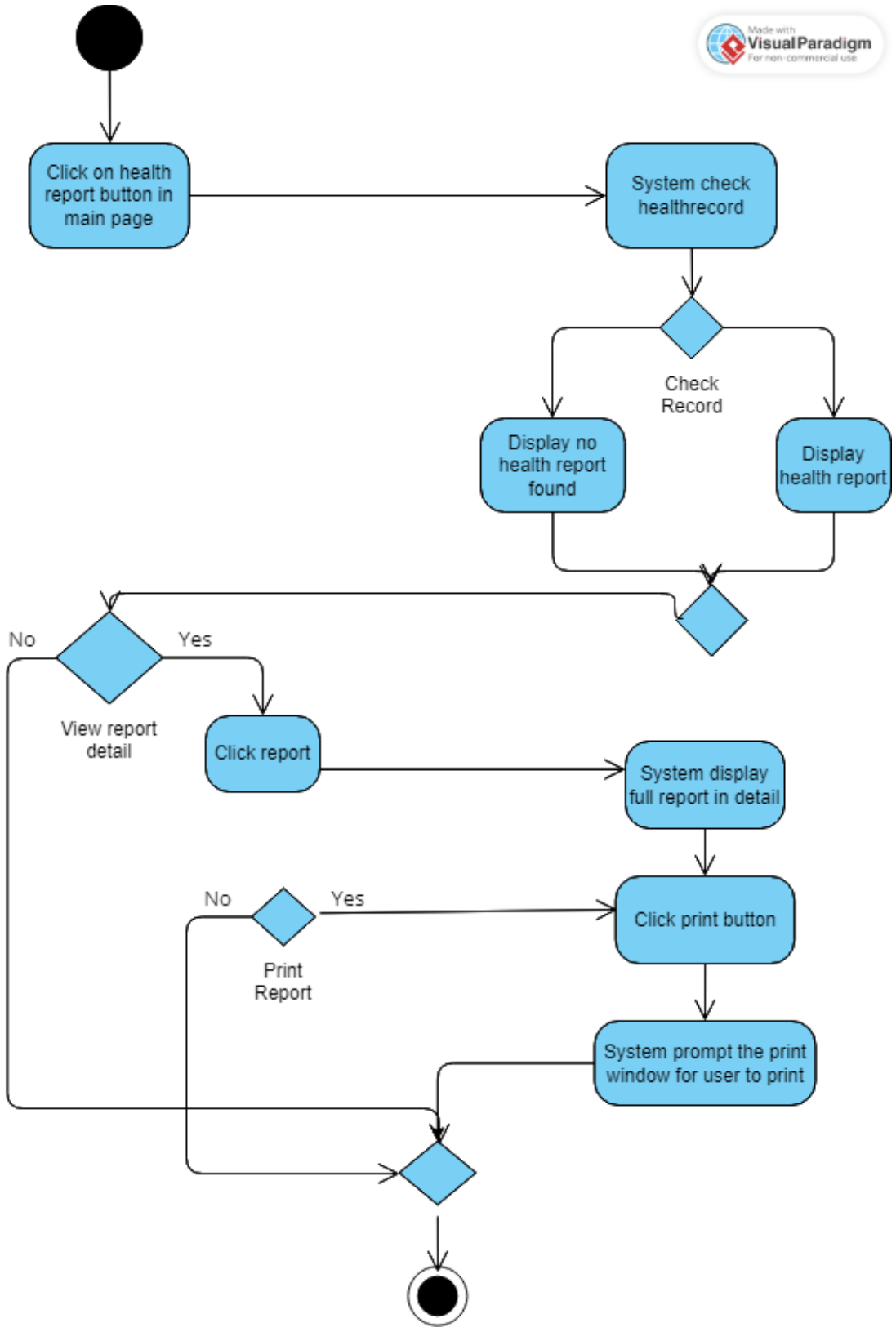


Figure 3.2.4.2.1 View Health Report Activity Diagram– Patient Perspective

Based on the activity diagram above, when patient want to view their health report generated by the doctor, they will click on the button in the patient main page. After that, system will direct patient to a page that contains the list of health report. The list will be containing health report pdf document and the generated date, with the name of the doctor. If patient are willing to print the health report, there is a print option button in the page, after clicking it, the system will prompt out the computer printing windows, user can modify the print page, print method and so on to print.

3.2.4.3 View Billing Record– Patient Perspective

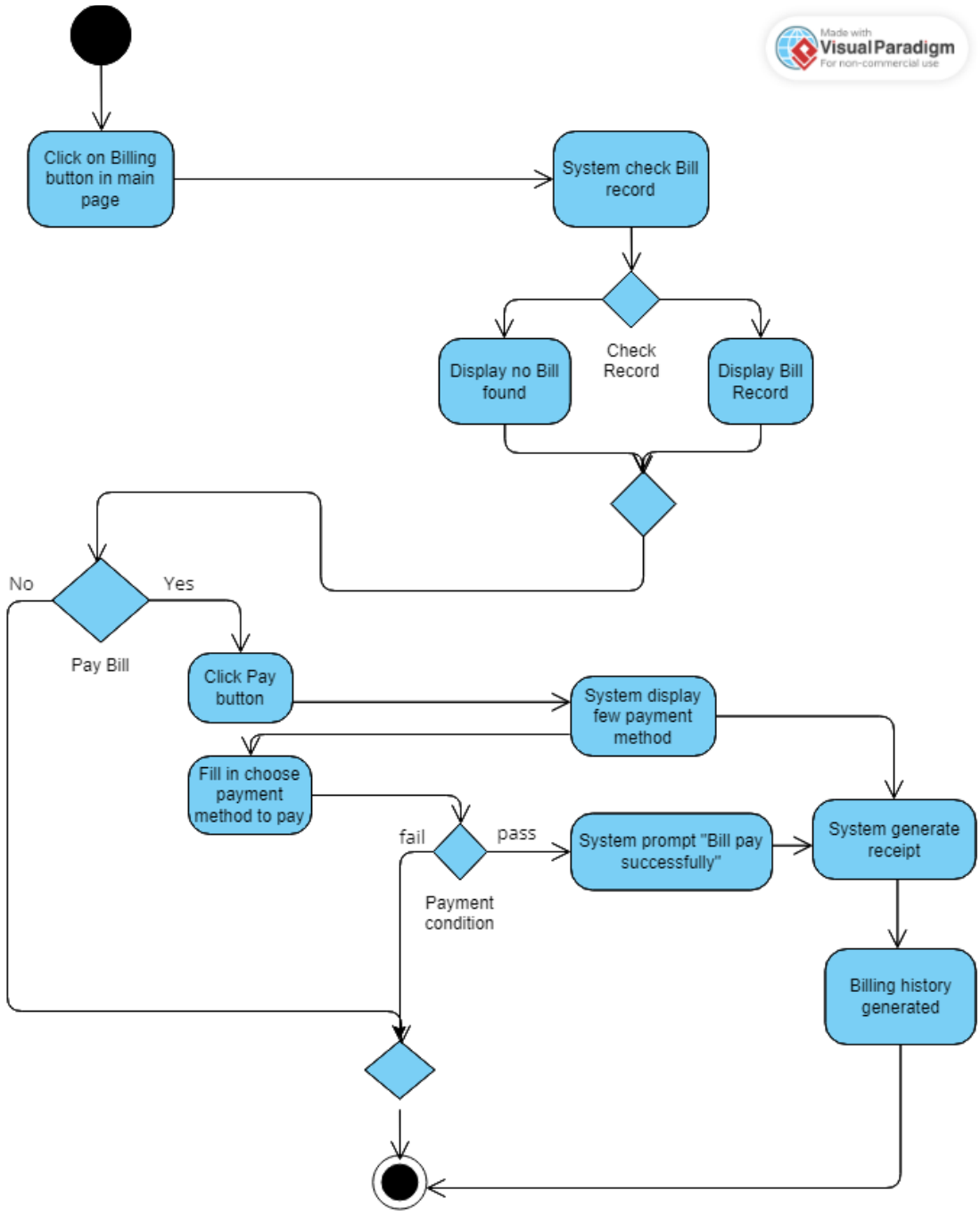


Figure 3.2.4.3.1 View Billing Activity Diagram– Patient Perspective

Based on the activity diagram above, when patient want to view the billing record, the patient can click on the bill button in the main page. In the backend, the system will check there is any bill record that belongs to the patient. If no, system will show the patient "no billing record", if there is billing report found, it will display to the patient. The patient can print the bill and pay the bill. If patient pay the bill, he/she will need

to select the payment method to pay. After done payment, the system will generate a resit for the patient.

3.2.4.4 View Personal Information – Patient Perspective

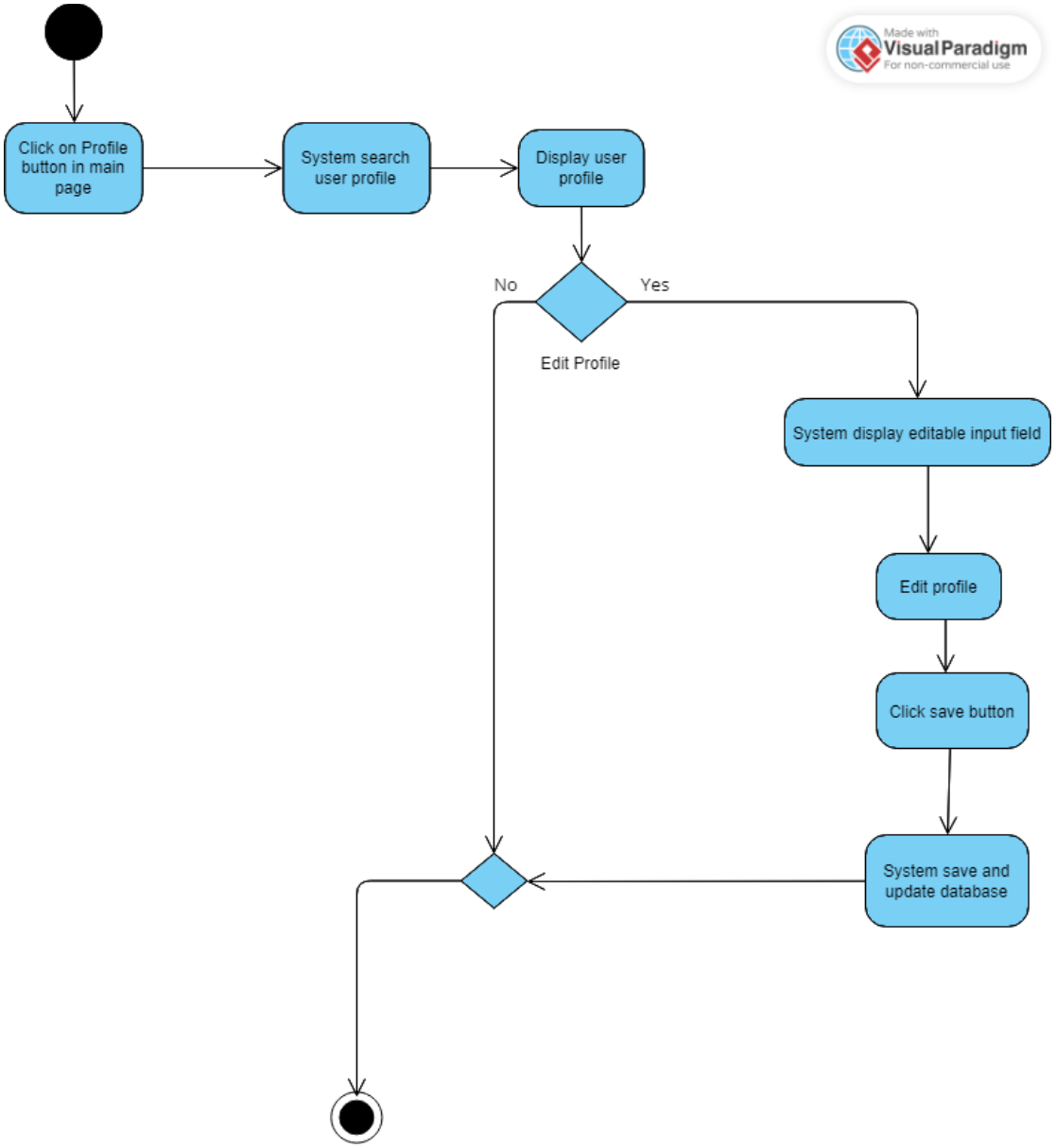


Figure 3.2.4.4.1 View Personal Information Activity Diagram– Patient Perspective

Based on the activity diagram above, the patient can view their personal information in the profile page which is a button name profile in main page. After entering the page, patient can view all of their information. If patient want to change information, they can click on the edit button. All the information will be change to the editable input field for patient to adjust. In the page also contain change account password. After change, the database will updated with patient new information.

3.2.4.5 View BMI Record – Patient Perspective

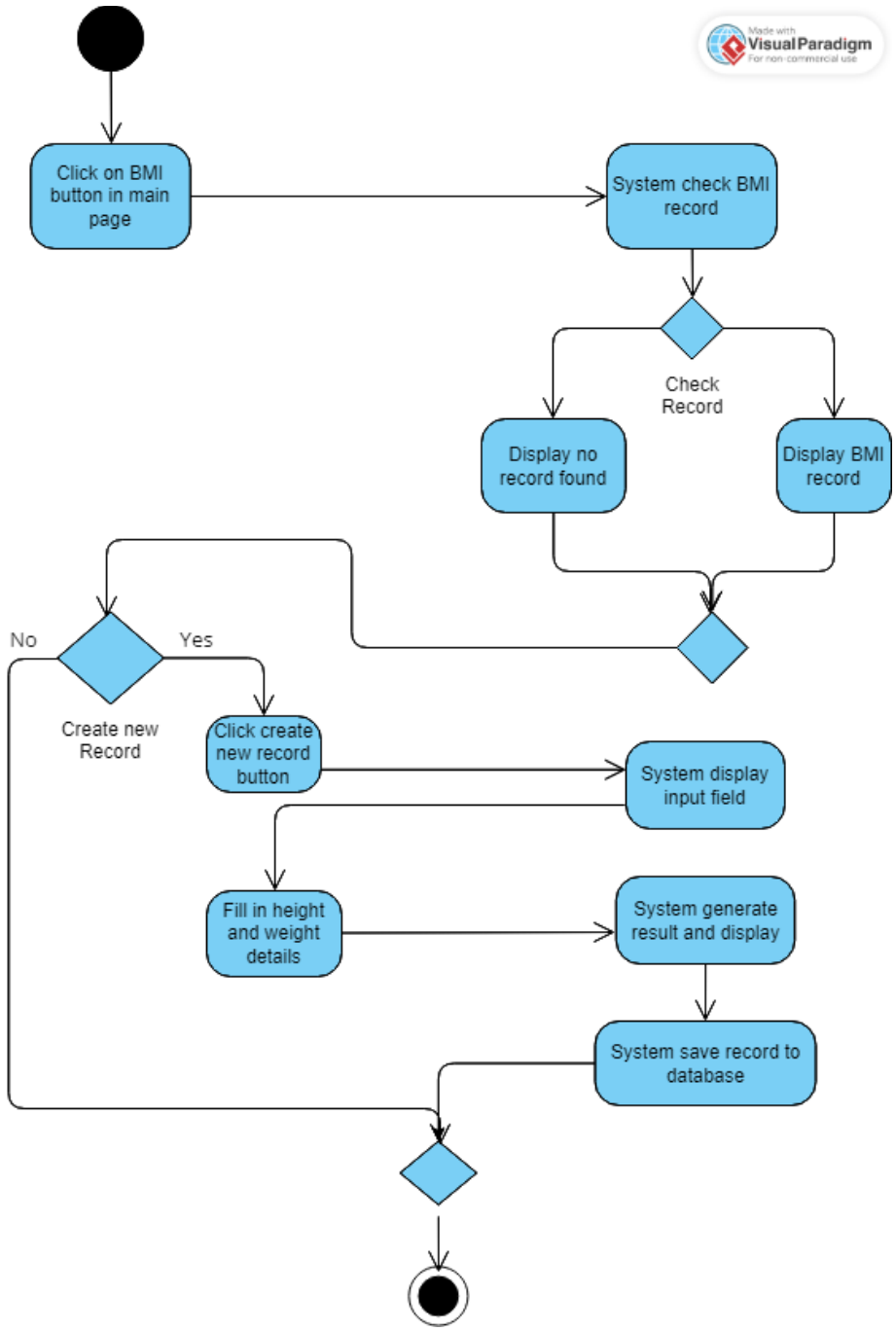


Figure 3.2.4.5.1 View BMI Record Activity Diagram– Patient Perspective

Based on the activity diagram above, if patient want to view their BMI record, they can click on a BMI record button on the main page. After that they will be direct to the BMI record page which they can view their BMI record. If they want to add a new record, they may click on add new record button and a form will prompt out. They are required to fill in their body information such as weight and height to measure their BMI. After generated, a new record will be automatically generated and saved in

database, other than the BMI record, an appropriate advice and commend will be generated also.

3.2.4.6 View Patient Appointment Detail – Admin Perspective

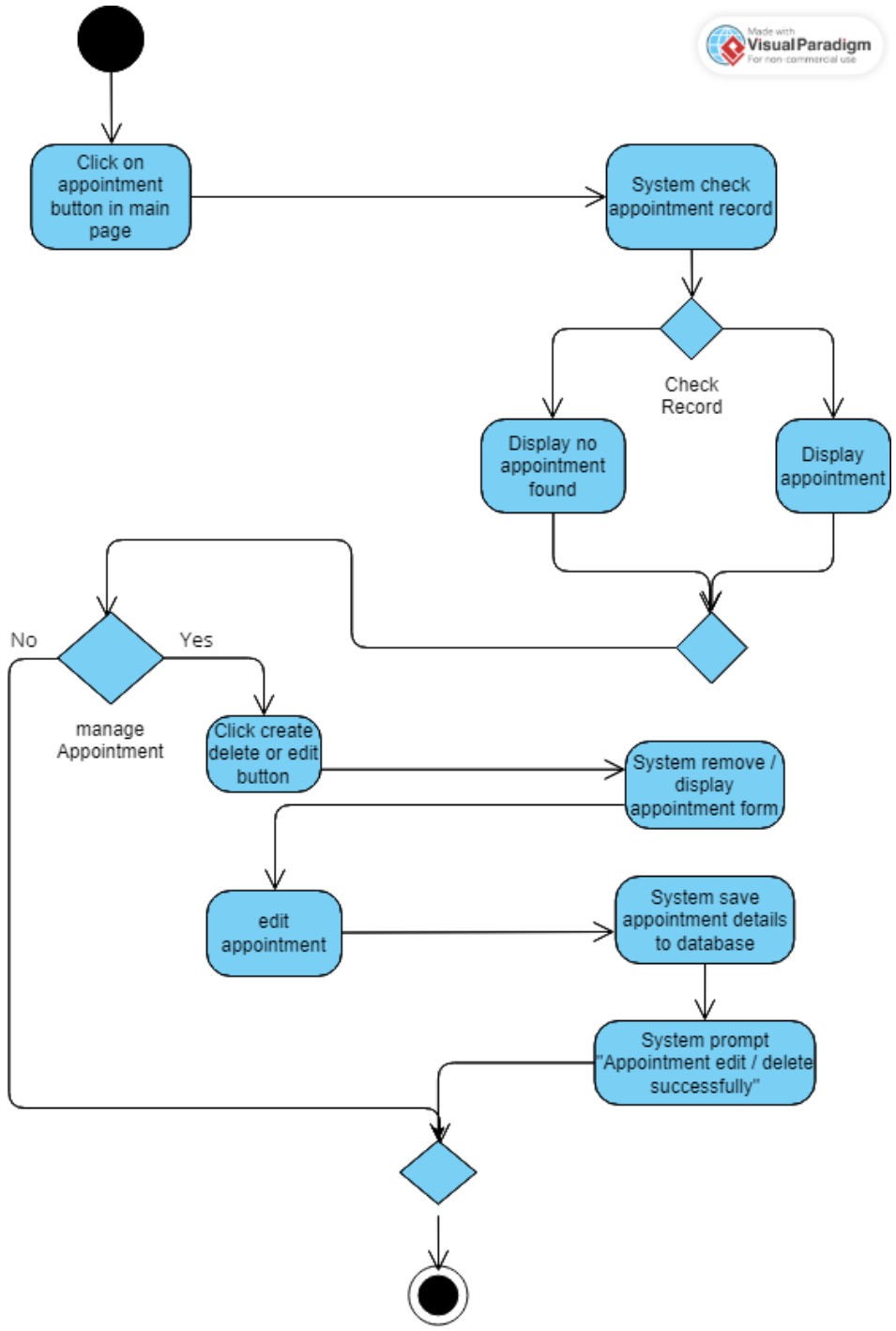


Figure 3.2.4.6.1 View Patient Appointment Detail Activity Diagram– Admin Perspective

Based on the activity diagram above, if admin want to view patient appointment detail, they can click on the appointment button in the main page. After that, system will display a list of appointment with the time, date and patient name. They can select the appointment and view the appointment detail. They can also manage all the patient appointment by delete, edit those appointments. After they edit and update, the system will prompt out “appointment edited successfully”

3.2.4.7 View Patient Information – Admin Perspective

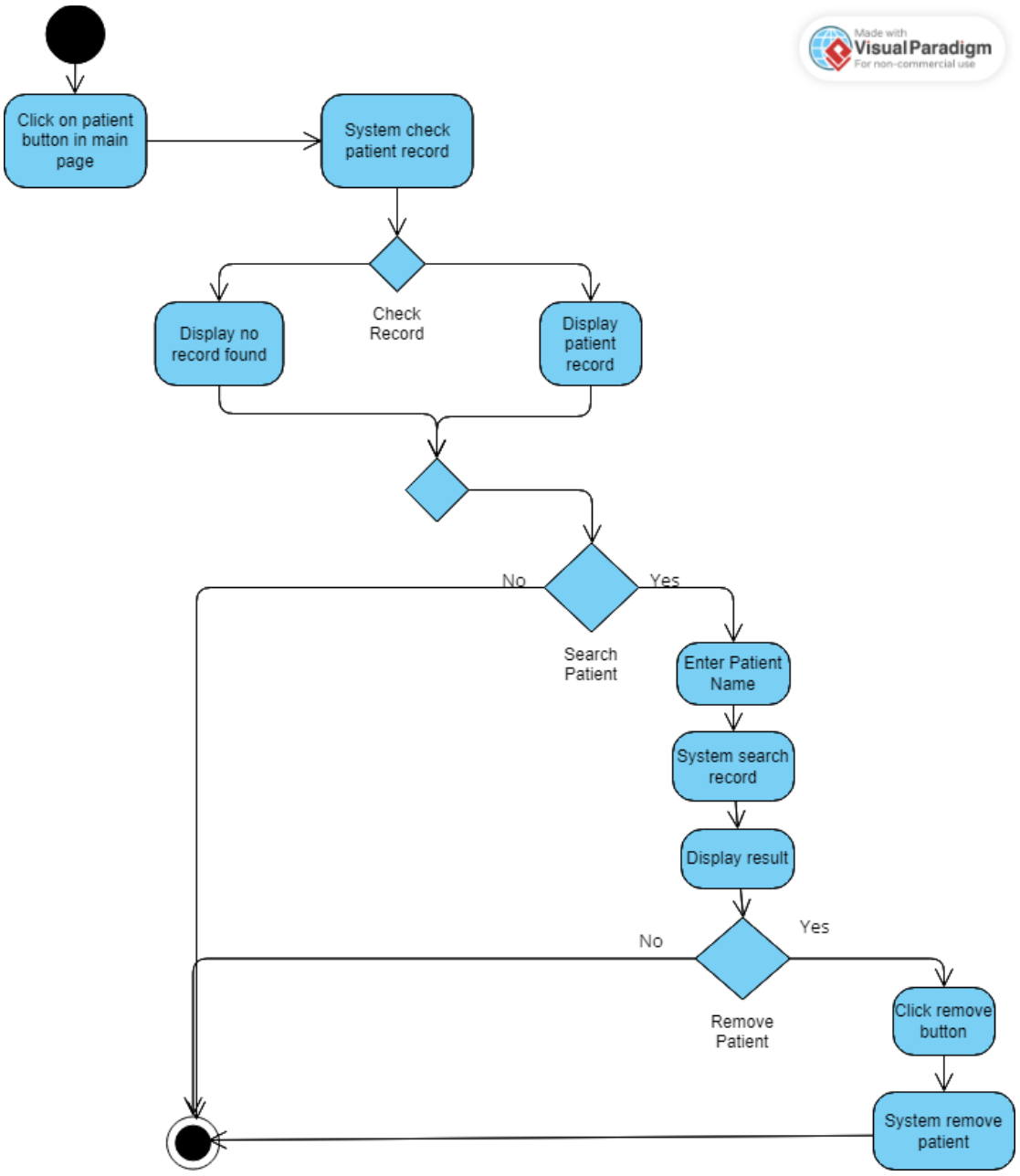


Figure 3.2.4.7.1 View Patient Information Activity Diagram– Admin Perspective

Based on the activity diagram above, when admin willing to view the patient data and information, admin can click on the patient button in the main page, a list of patient record will be display. Other than that, admin can search specific patient record by searching patient by their name. After result generated, patient can click the record and view into more detail patient information. Admin also edit the patient information and update to database**.**

3.2.4.8 Approve Doctor Job Request– Admin Perspective

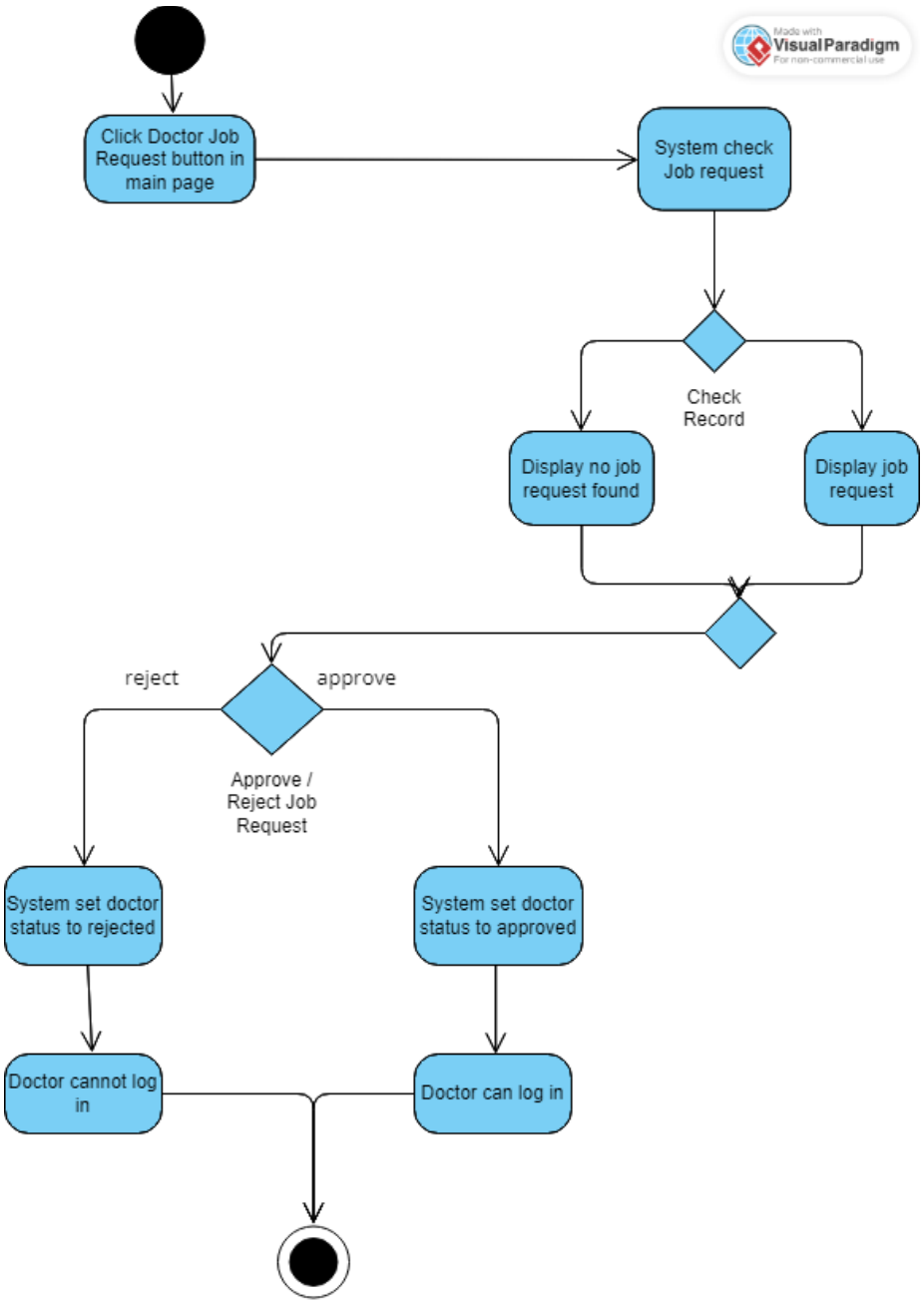


Figure 3.2.4.8.1 Approve Doctor job Request Activity Diagram– Admin Perspective

Based on the activity diagram above, when admin click into doctor job request button, the system will check the job request of doctor which doctor status is pending, and the system will display out to the admin. If no, it will display no job request. After that, admin need to select whether reject or approve job request to prevent others register as clinic doctor also. If that register account is really a doctor, admin will click approve button. If no, the admin will set to rejected button and the user will not be able to login to doctor page.

3.2.4.9 Generate Patient Bill – Admin Perspective

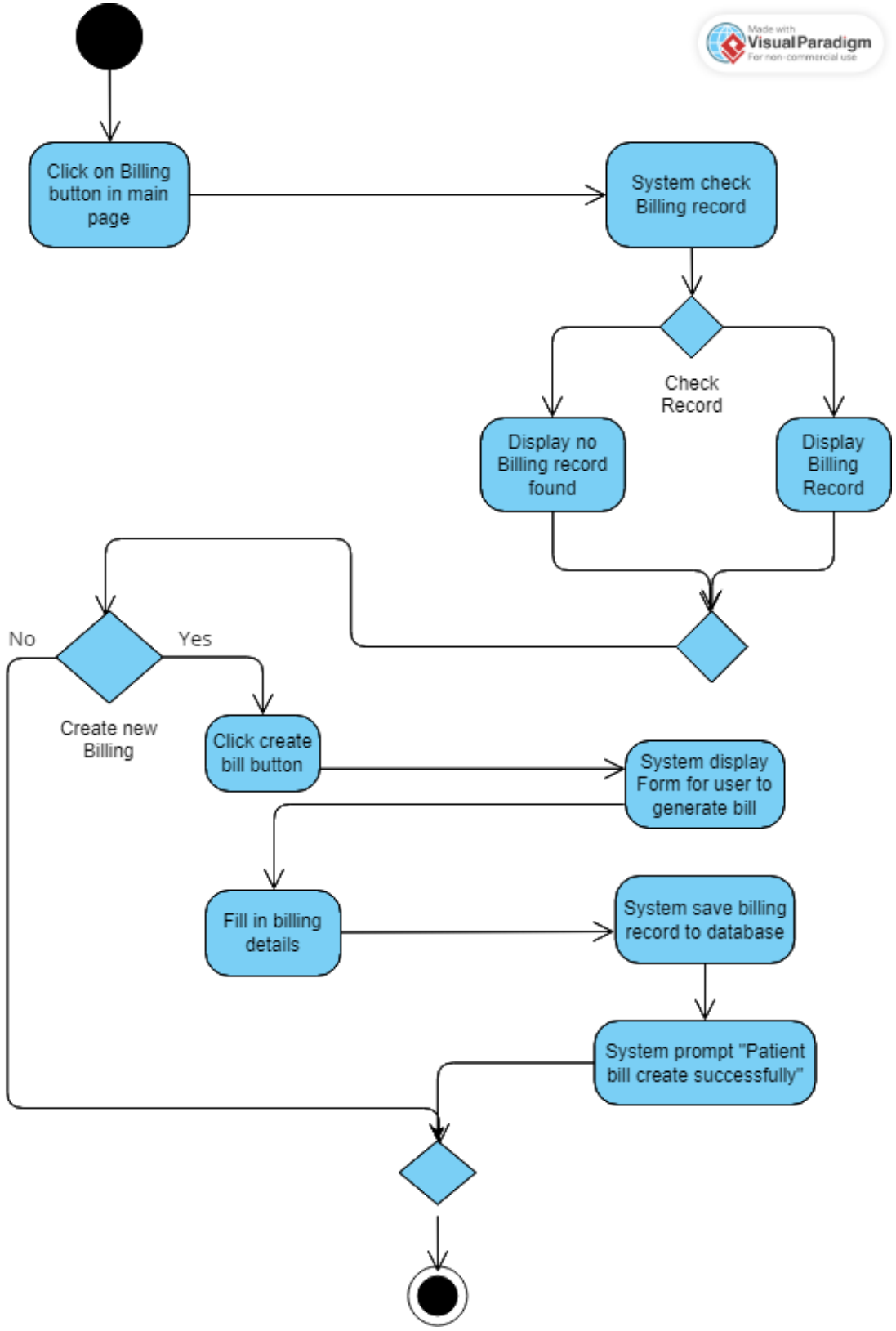


Figure 3.2.4.9.1 Generate Patient Bill Activity Diagram– Admin Perspective

Based on the activity diagram above, when admin click on the generate bill button in the admin main page, the system will check the billing record history and display out all. If admin want to generate a new bill, he/she has to click on a button name "generate new bill" and this will direct them to generate new bill. In the generate new bill page, admin will be ask to fill in the information needed for generating a bill, and after input validation a pdf document will be generated as patient bill, after they click save, the bill will automatically upload based on the patient id and the patient will be able to view the bill.

3.2.4.11 Generate Patient Health Report – Doctor perspective

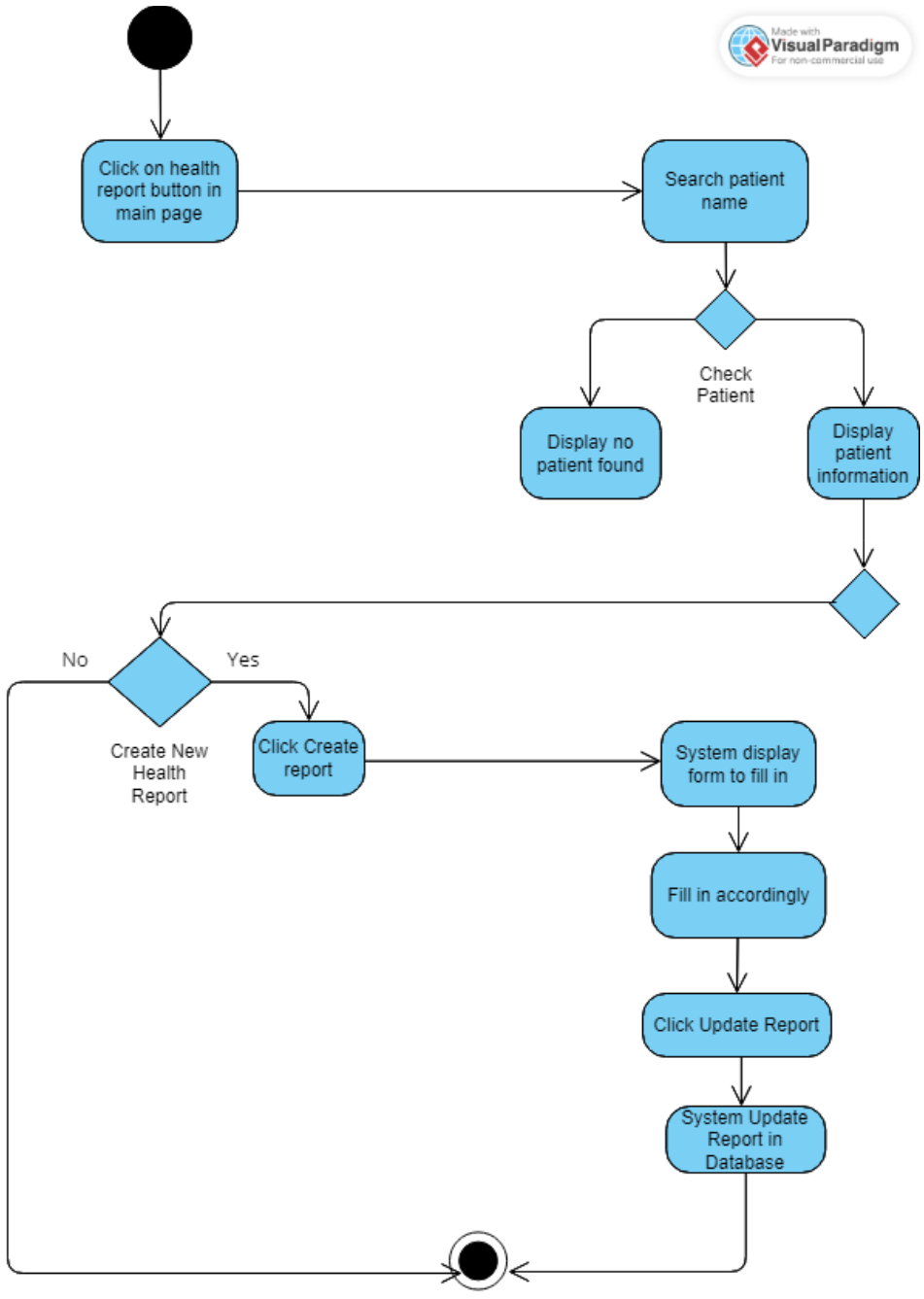


Figure 3.2.4.11.1 Generate patient health report Activity Diagram– Doctor Perspective

Based on the activity diagram above, if doctor want to generate health report to patient, they have to click on the health report button in the main page. After doctor click inside, the system will automatically generate all the patient report history and display to doctor. If doctor want to generate new health report, they need to click the new report button and fill in the information in the system generated form. After

submitting, the system will automatically generate a new health report to the doctor and the patient are able to view the health report and print it.

3.2.4.12 View Patient Appointment – Doctor perspective

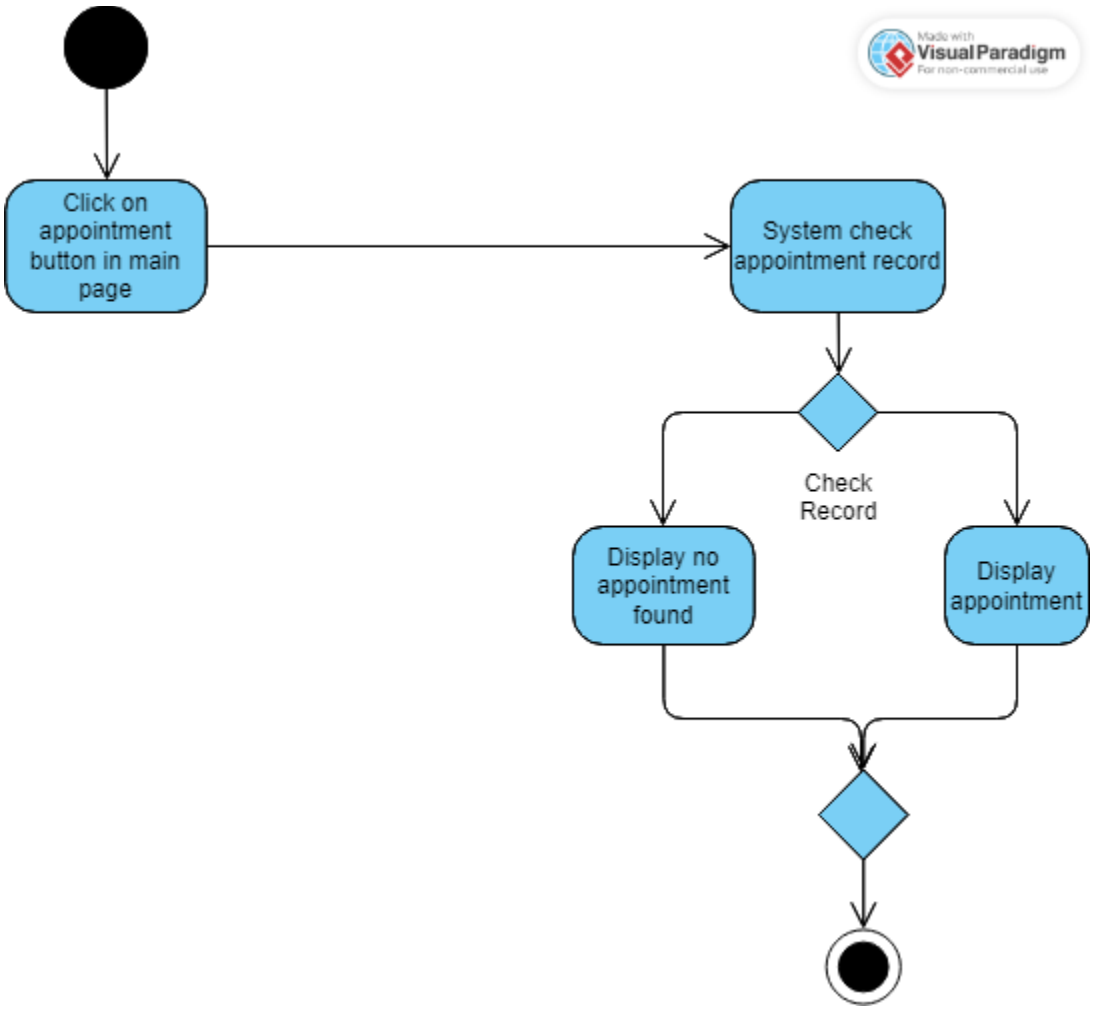


Figure 3.2.4.12.1 View Patient Appointment Activity Diagram– Doctor Perspective

When the doctor clicks on the appointment button on the main page the system will prompt to check the appointment records. A decision is then made based on whether an appointment exists. If an appointment is found, the system displays the details to the doctor; if no appointment is found, it informs the doctor that no appointment is available.

3.2.4.13 View Patient Details – Doctor perspective

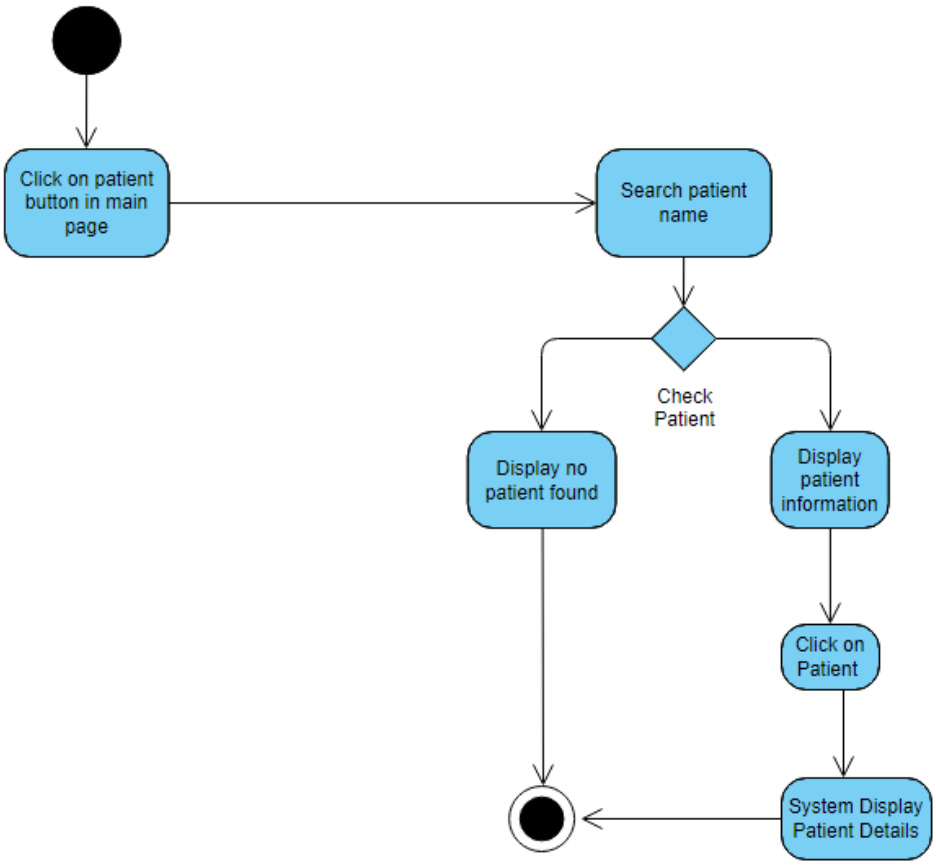


Figure 3.2.4.13.1 View Patient Detail Activity Diagram– Doctor Perspective

When the doctor clicks on the patient button on the main page, prompting the system to search for the patient's name. The system then checks if the patient exists in the database. If no patient is found, a message is displayed to inform the doctor. If the patient is found, the system displays the patient's information. The doctor can then click on the patient to view more detailed information, which the system subsequently displays.

3.2.4.14 View Personal Information – Doctor perspective

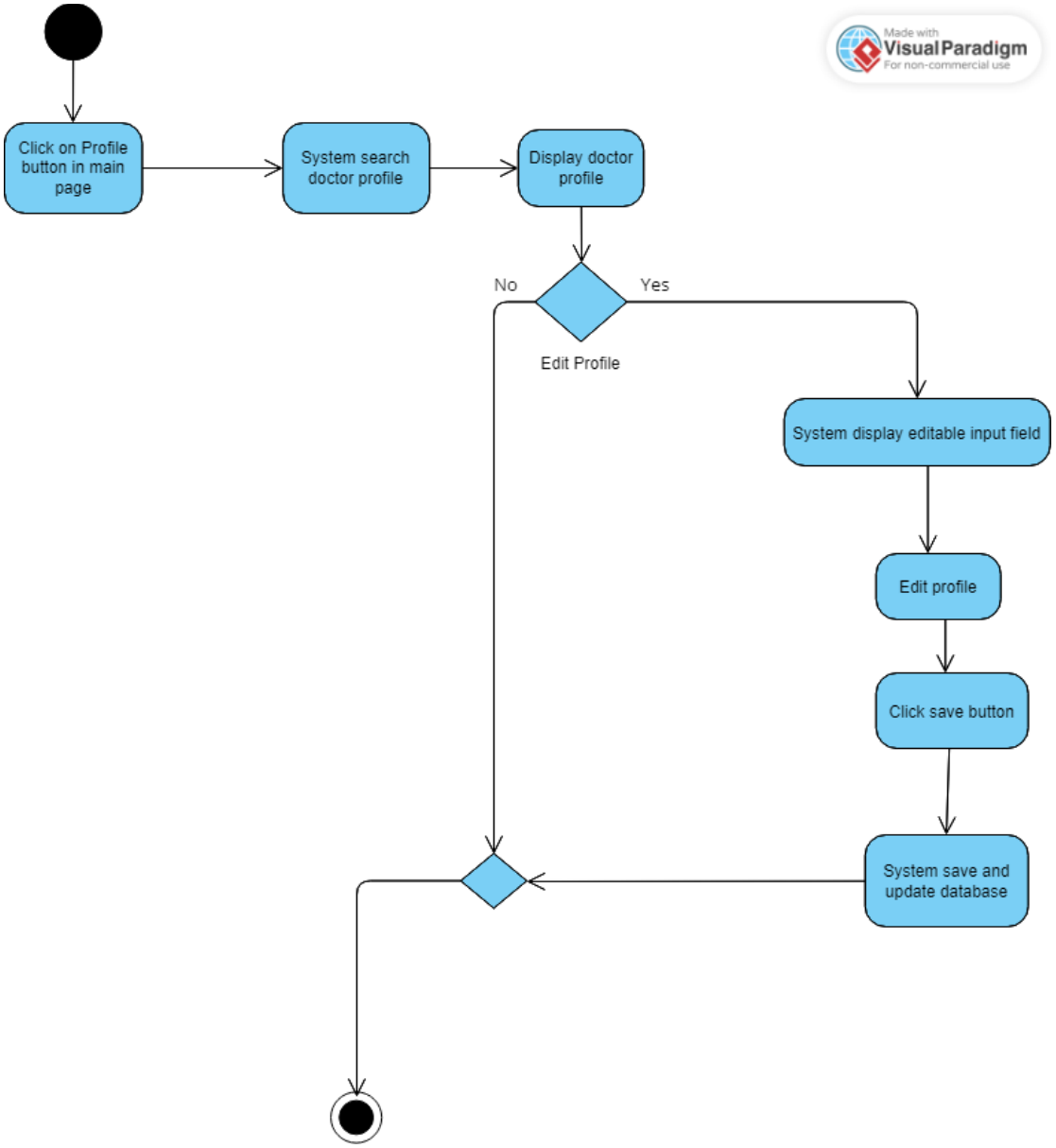


Figure 3.2.4.14.1 View Personal Detail Activity Diagram– Doctor Perspective

When the doctor clicks on the profile button on the main page. The system then searches for his/her profile and displays it. A decision is made to either edit the profile or not. If doctor want to edit the profile, the system displays editable input fields. The doctor can then make changes to the profile and click the save button. Finally, the system saves the changes and updates the database.

3.2.4.15 Login

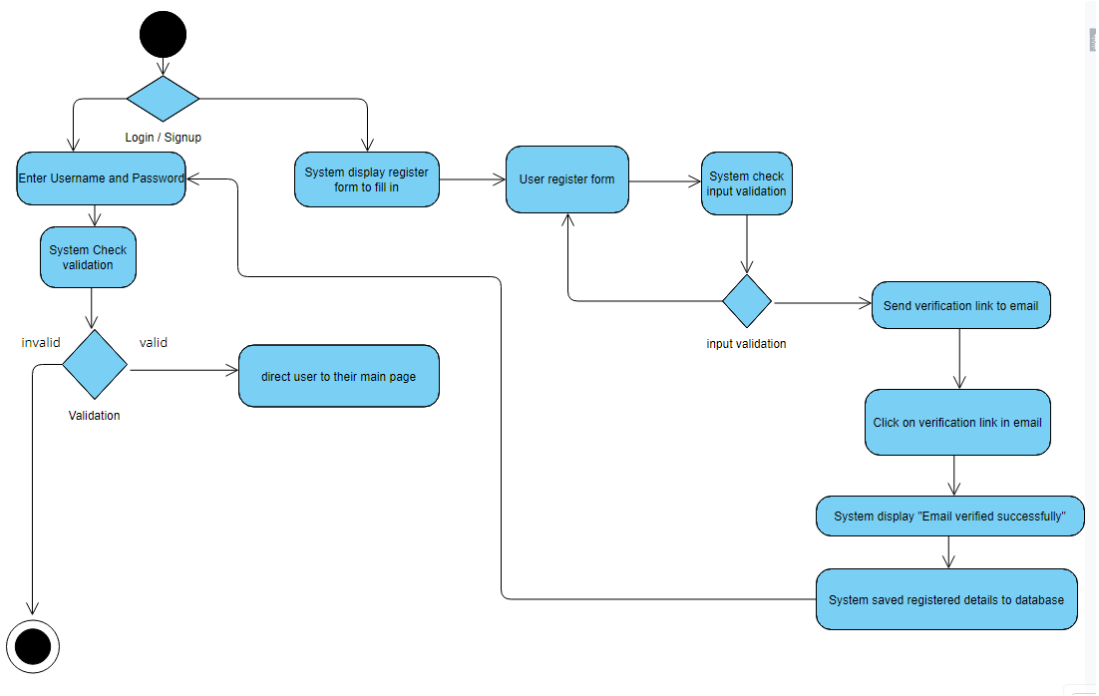


Figure 3.2.4.15.1 Login Activity Diagram– All user

3.2.4.18 Log Out

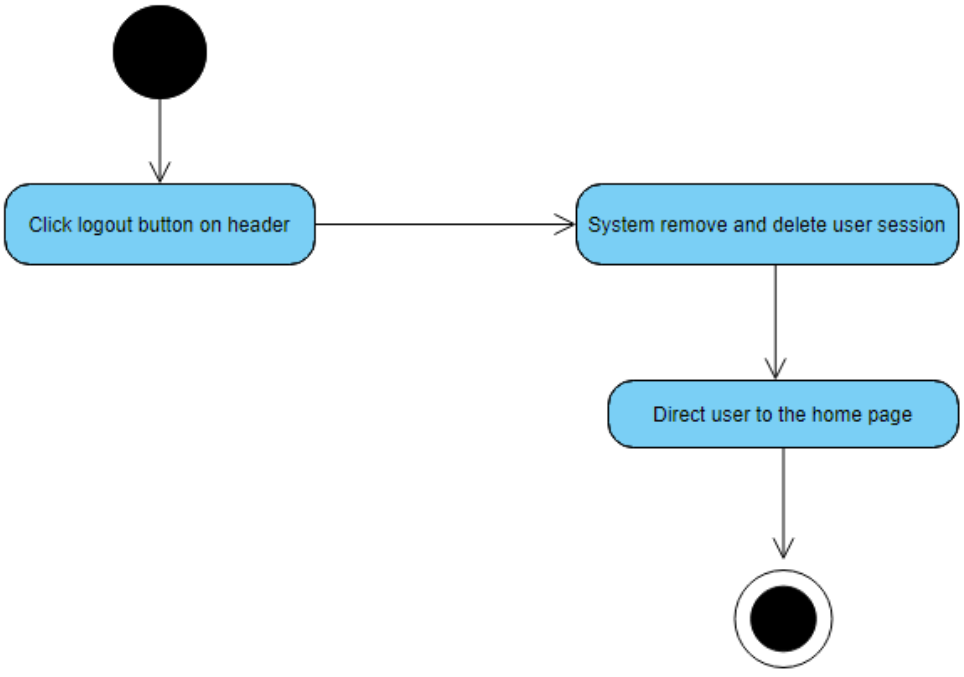


Figure 3.2.4.16.1 Log out Activity Diagram– All user

3.2.5 Sequence Diagram

3.2.5.1 View Appointment – Patient Perspective

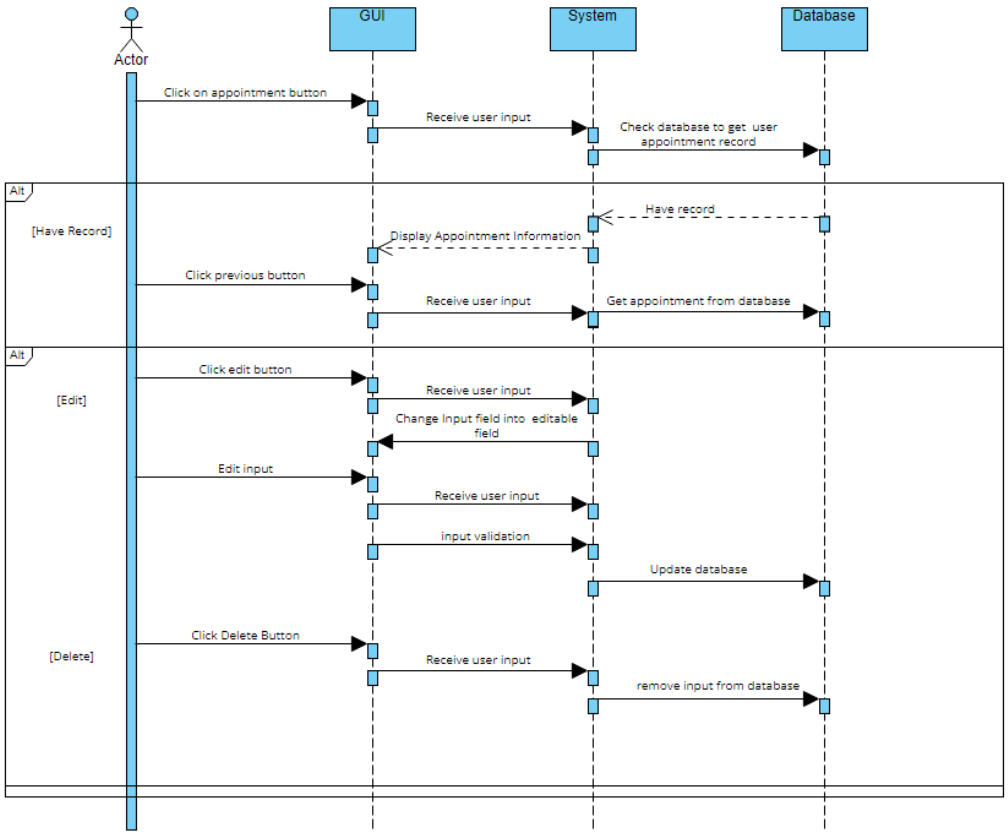


Figure 3.2.5.1.1 View Appointment Sequence Diagram– Patient Perspective

3.2.5.2 View Health Report – Patient Perspective

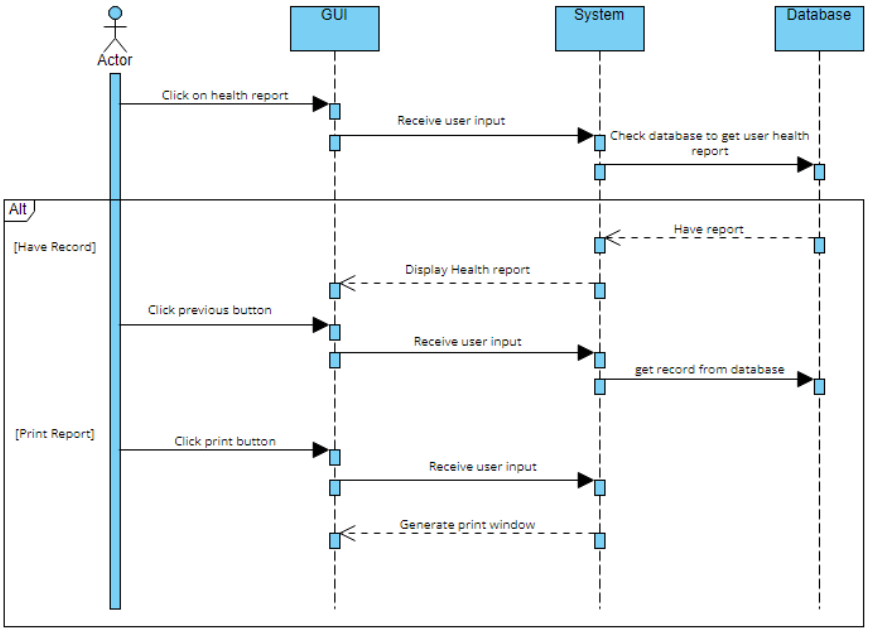


Figure 3.2.5.2.1 View Health report Sequence Diagram– Patient Perspective

3.2.5.3 View Billing Report – Patient Perspective

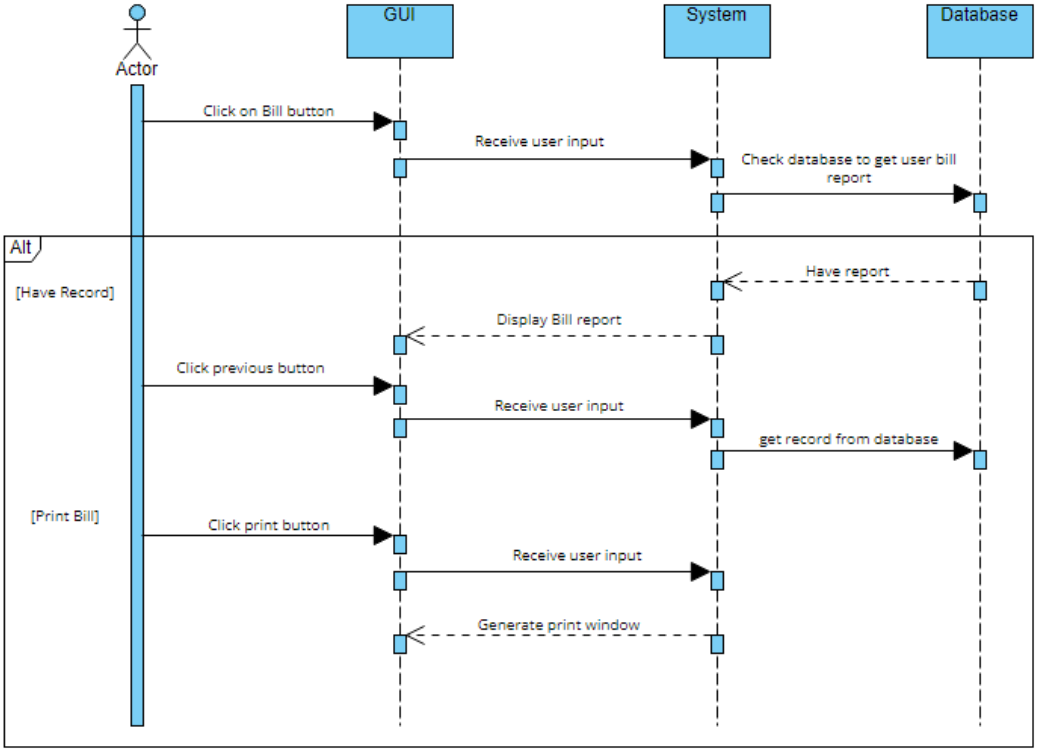


Figure 3.2.5.3.1 View Billing report Sequence Diagram– Patient Perspective

3.2.5.4 View Personal Information – Patient Perspective

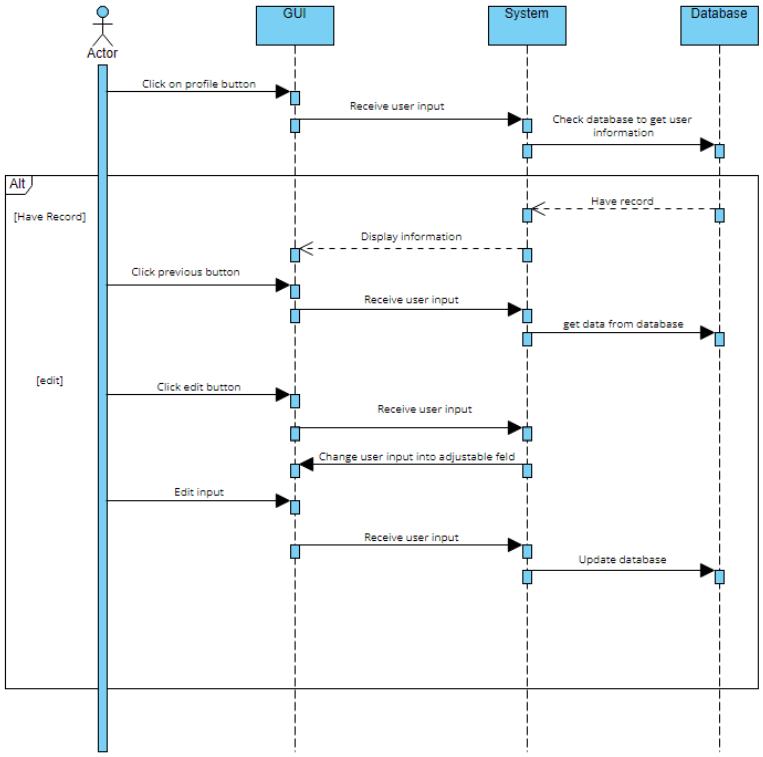


Figure 3.2.5.4.1 View Personal Information Sequence Diagram– Patient Perspective

3.2.5.4 View BMI Report – Patient Perspective

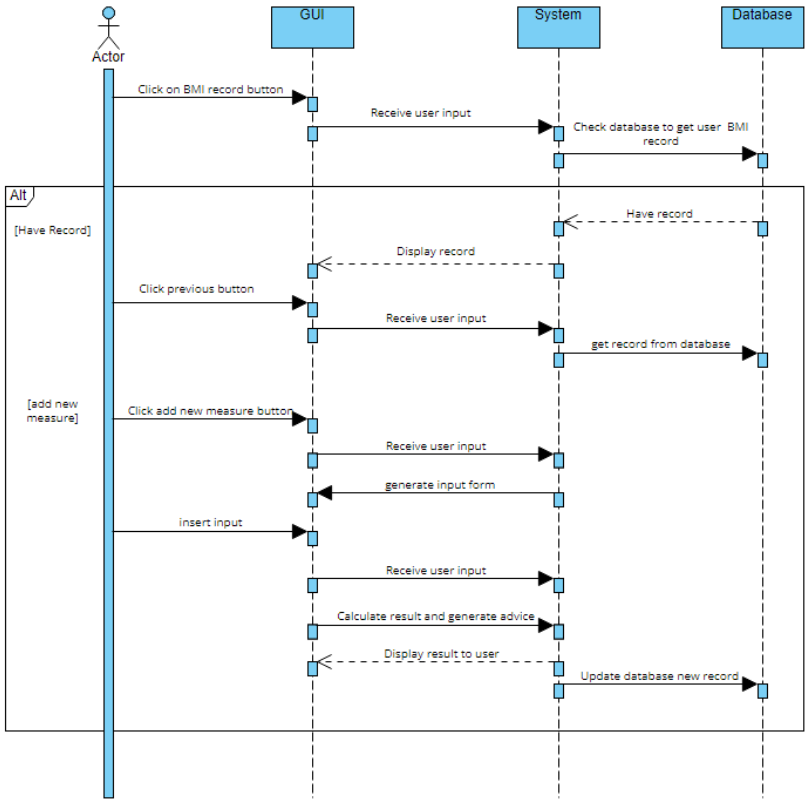


Figure 3.2.5.4.1 View BMI record Sequence Diagram– Patient Perspective

3.2.5.5 View Patient Appointment Detail – Admin Perspective

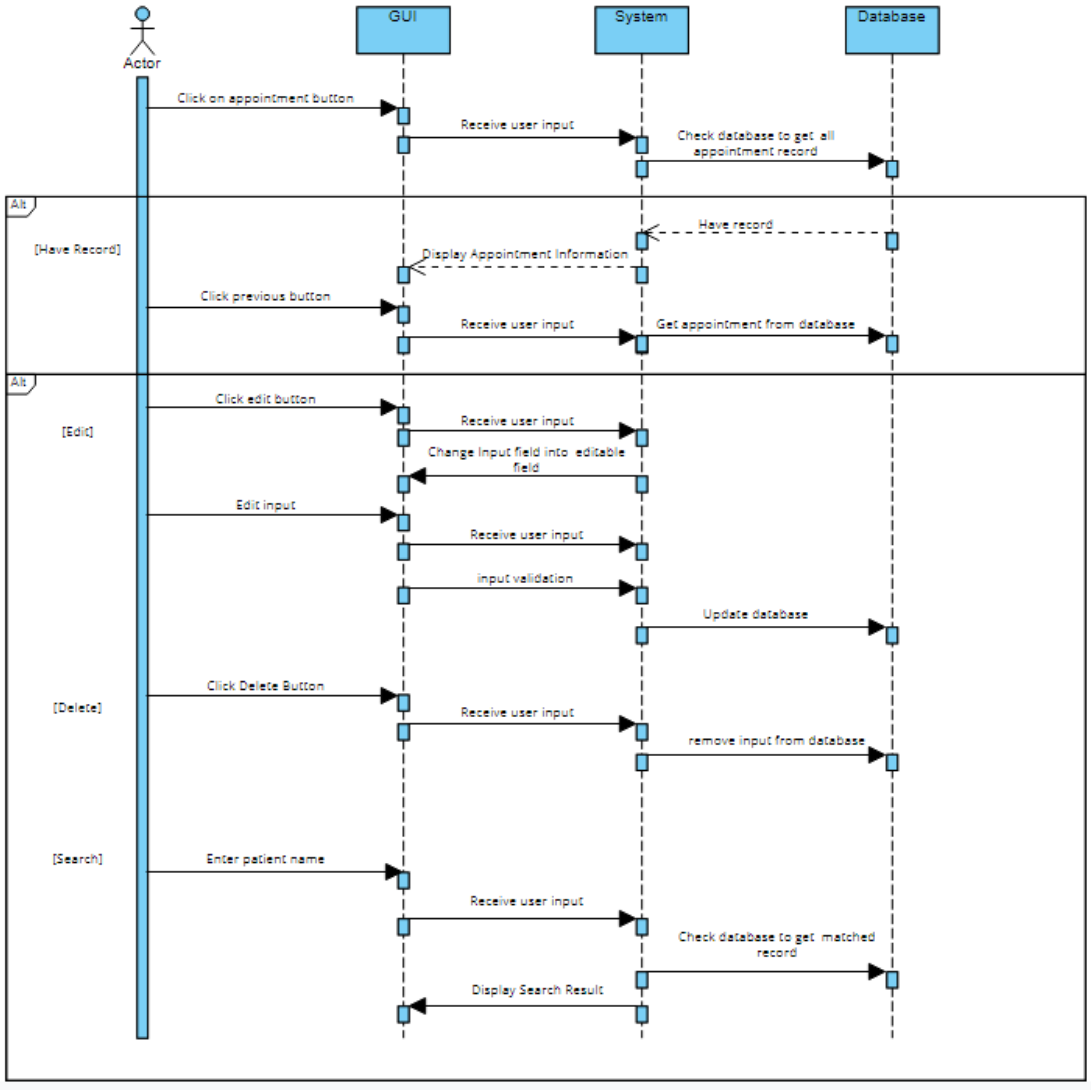


Figure 3.2.5.5.1 Patient Appointment Detail Sequence Diagram– Admin Perspective

3.2.5.6 View Patient Information– Admin Perspective

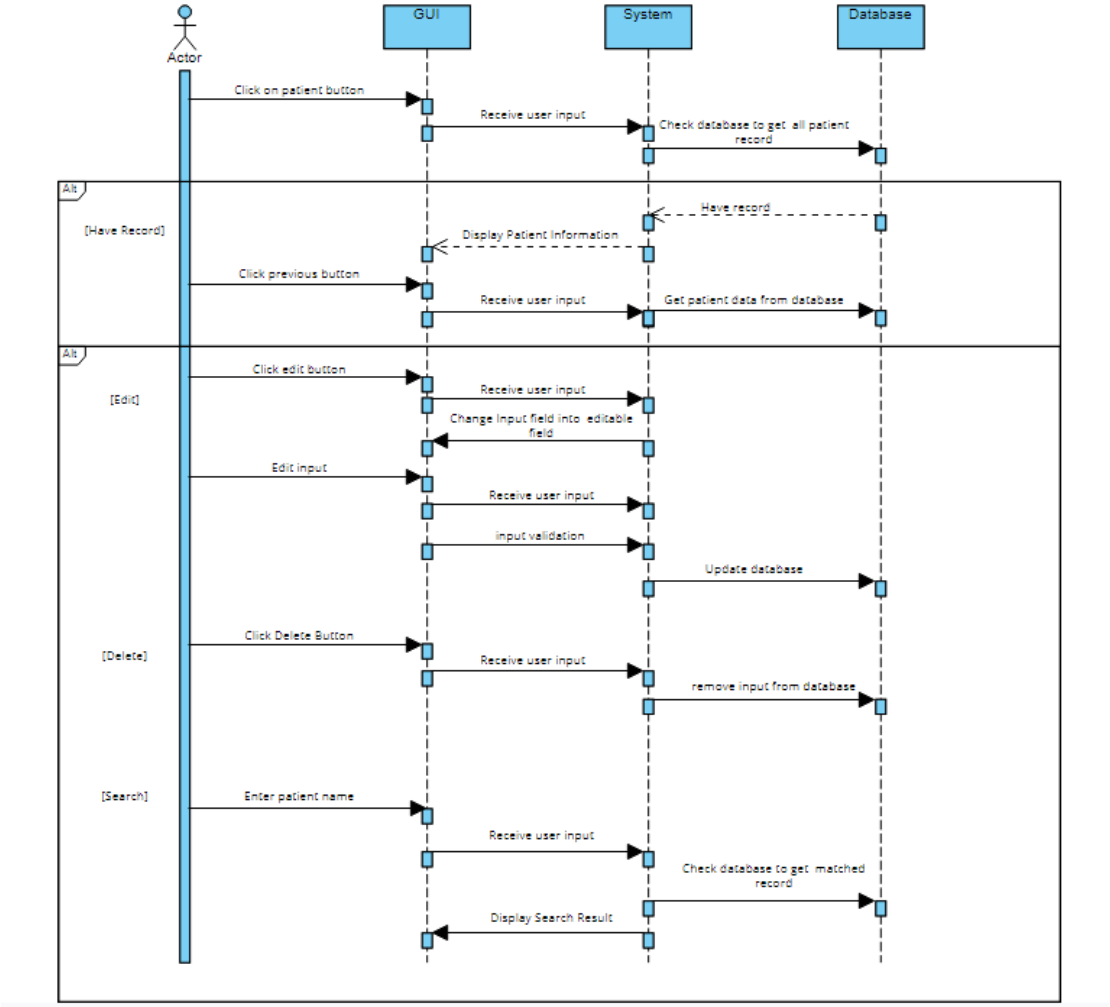


Figure 3.2.5.6.1 View Patient Information Sequence Diagram– Admin Perspective

3.2.5.7 Approve Doctor Job Request– Admin Perspective

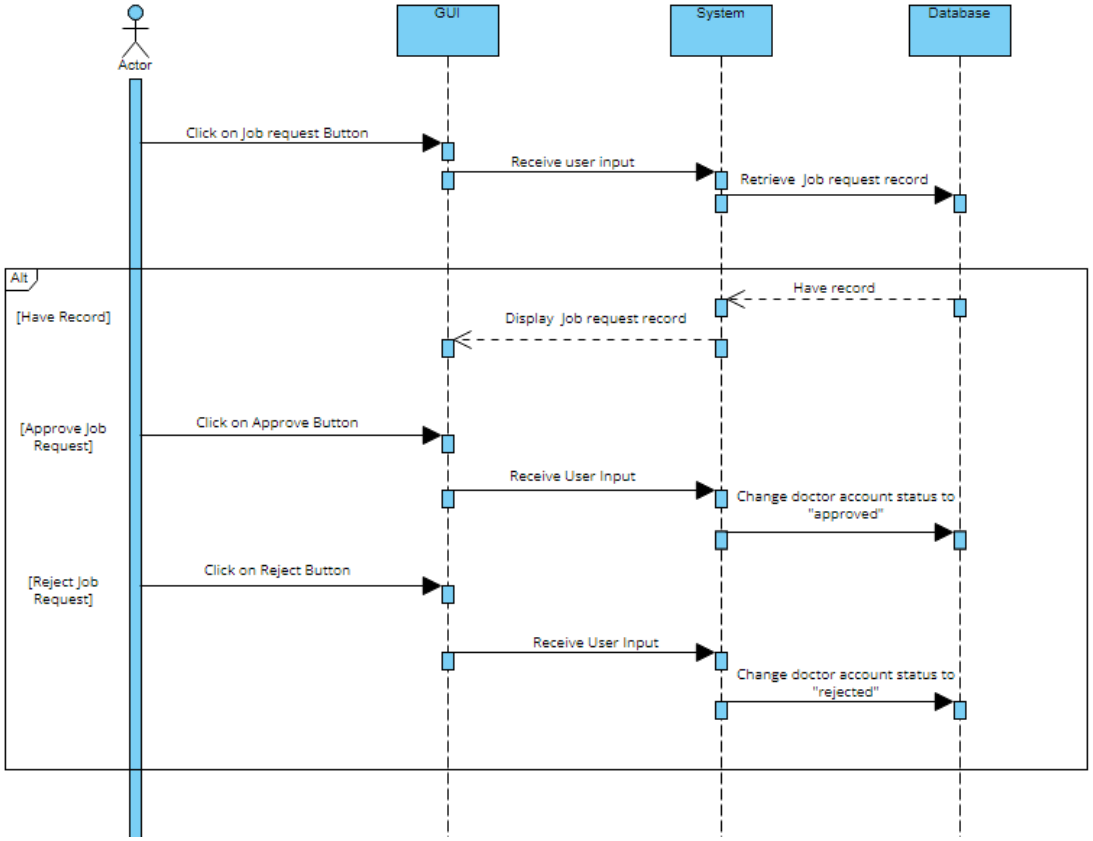


Figure 3.2.5.7.1 Approve Doctor Job request Sequence Diagram– Admin Perspective

3.2.5.8 Generate Patient Bill – Admin Perspective

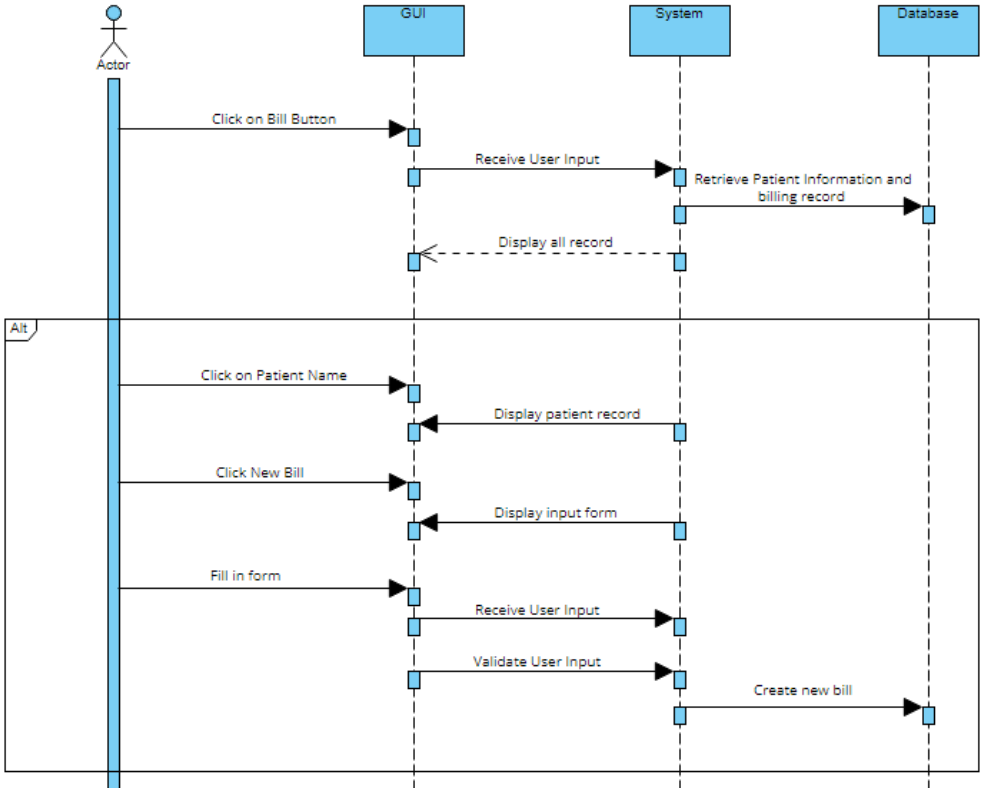


Figure 3.2.5.8.1 Generate Patient Bill Sequence Diagram– Admin Perspective

3.2.5.10 Generate Patient Health Report – Doctor Perspective

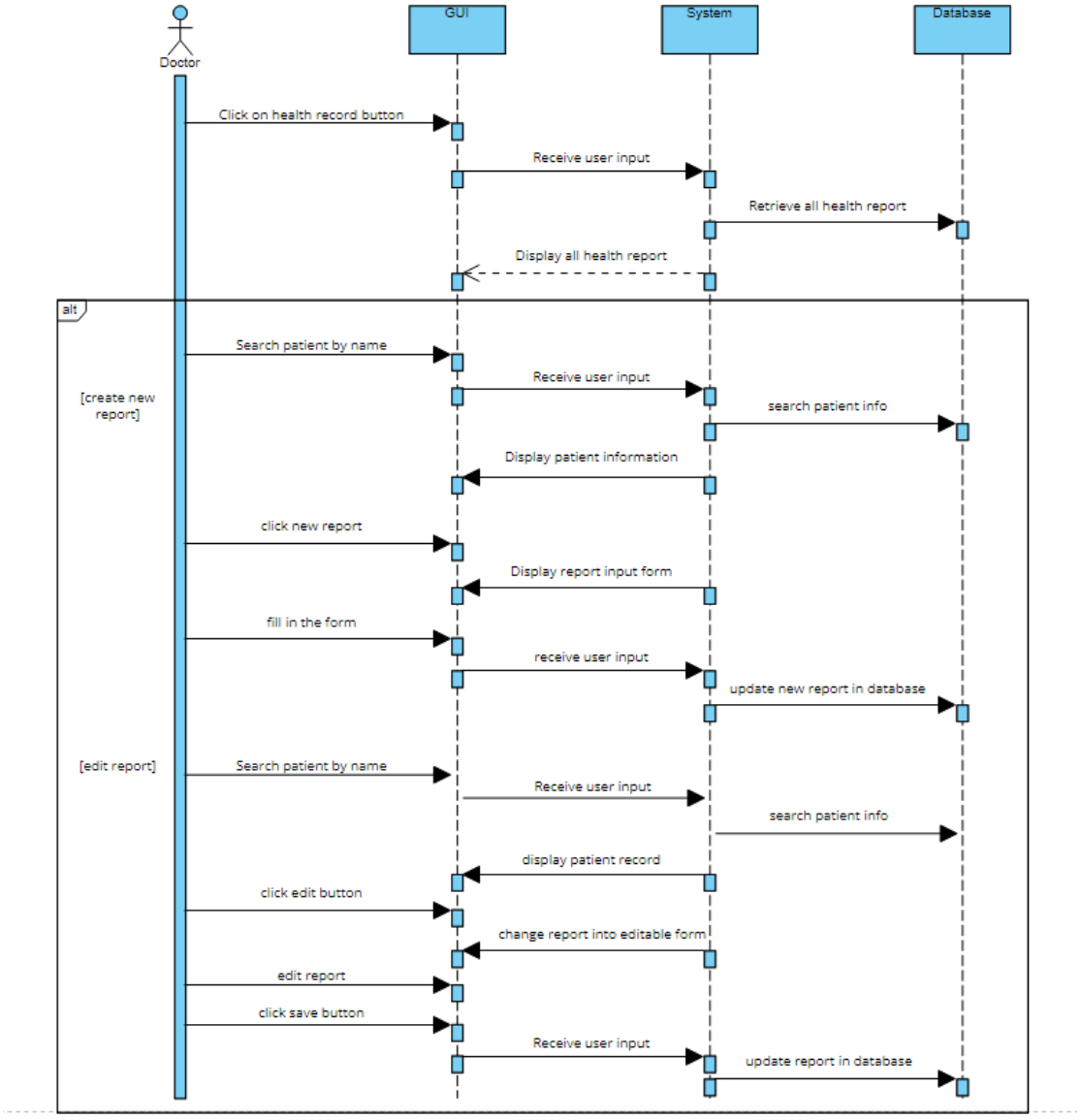


Figure 3.2.5.10.Generate Patient Health Record Sequence Diagram– Doctor Perspective

3.2.5.11 View Patient Appointment– Doctor Perspective

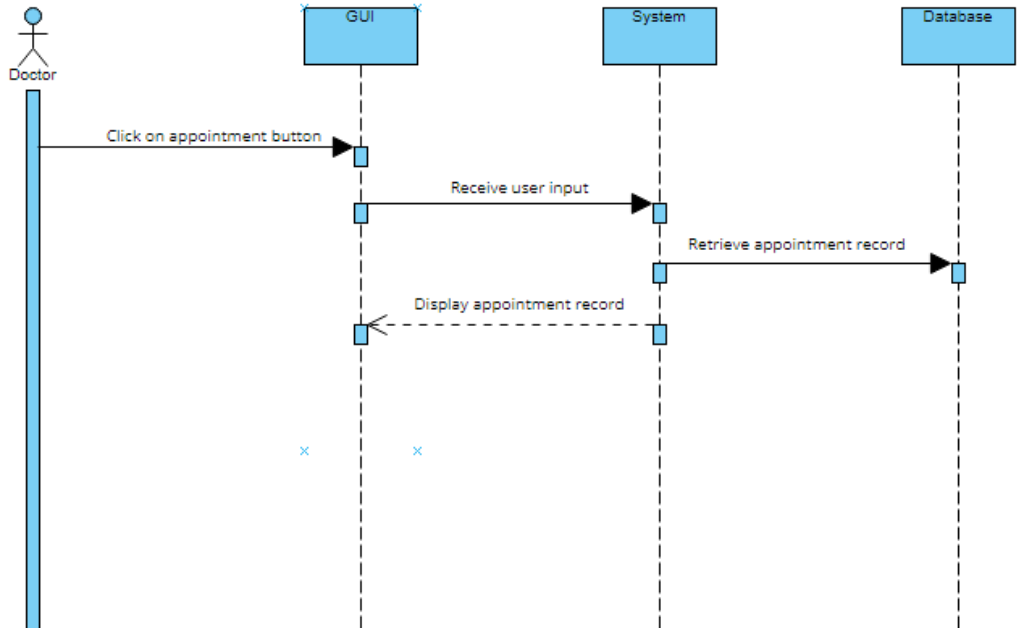


Figure 3.2.5.11.1 View Patient Appointment Sequence Diagram– Doctor Perspective

3.2.5.12 View Patient Details– Doctor Perspective

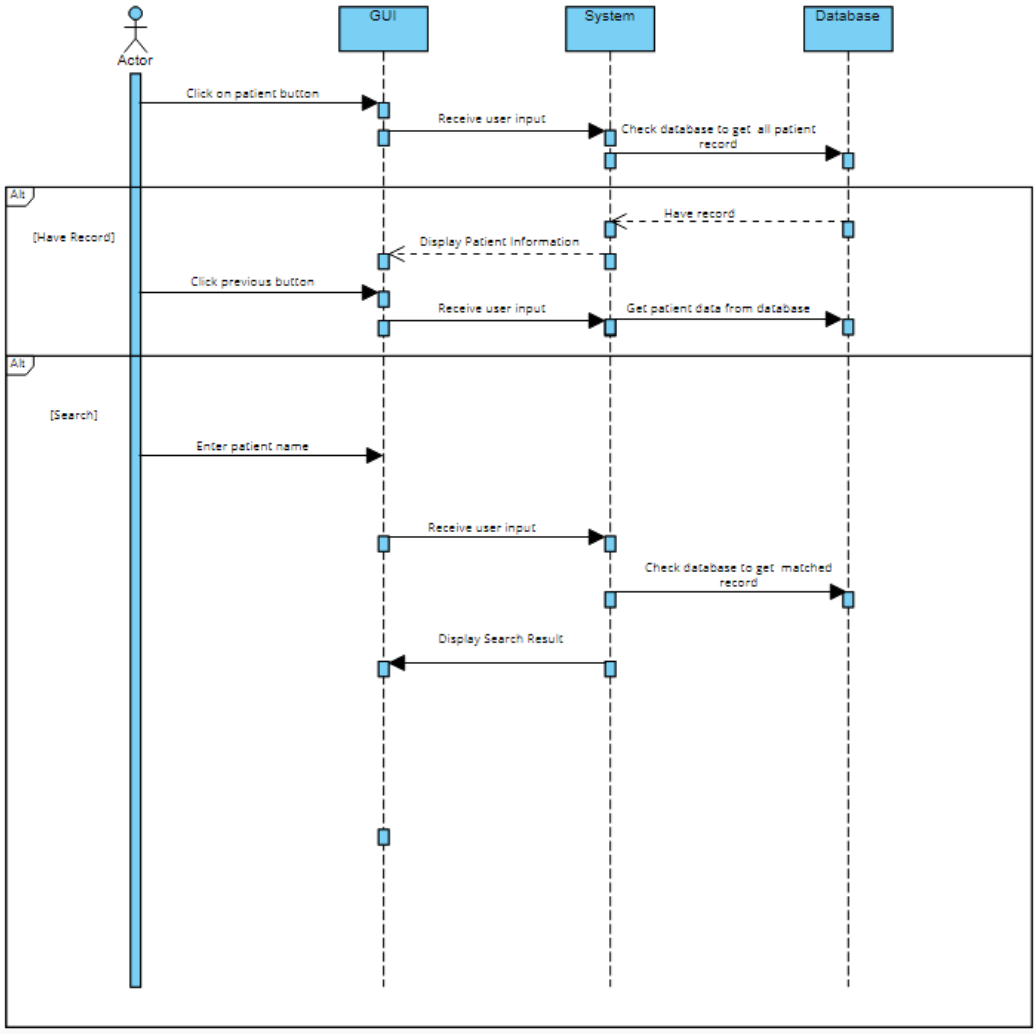


Figure 3.2.5.12.1 View Patient Details Sequence Diagram– Doctor Perspective

3.2.5.13 View Personal Information– Doctor Perspective

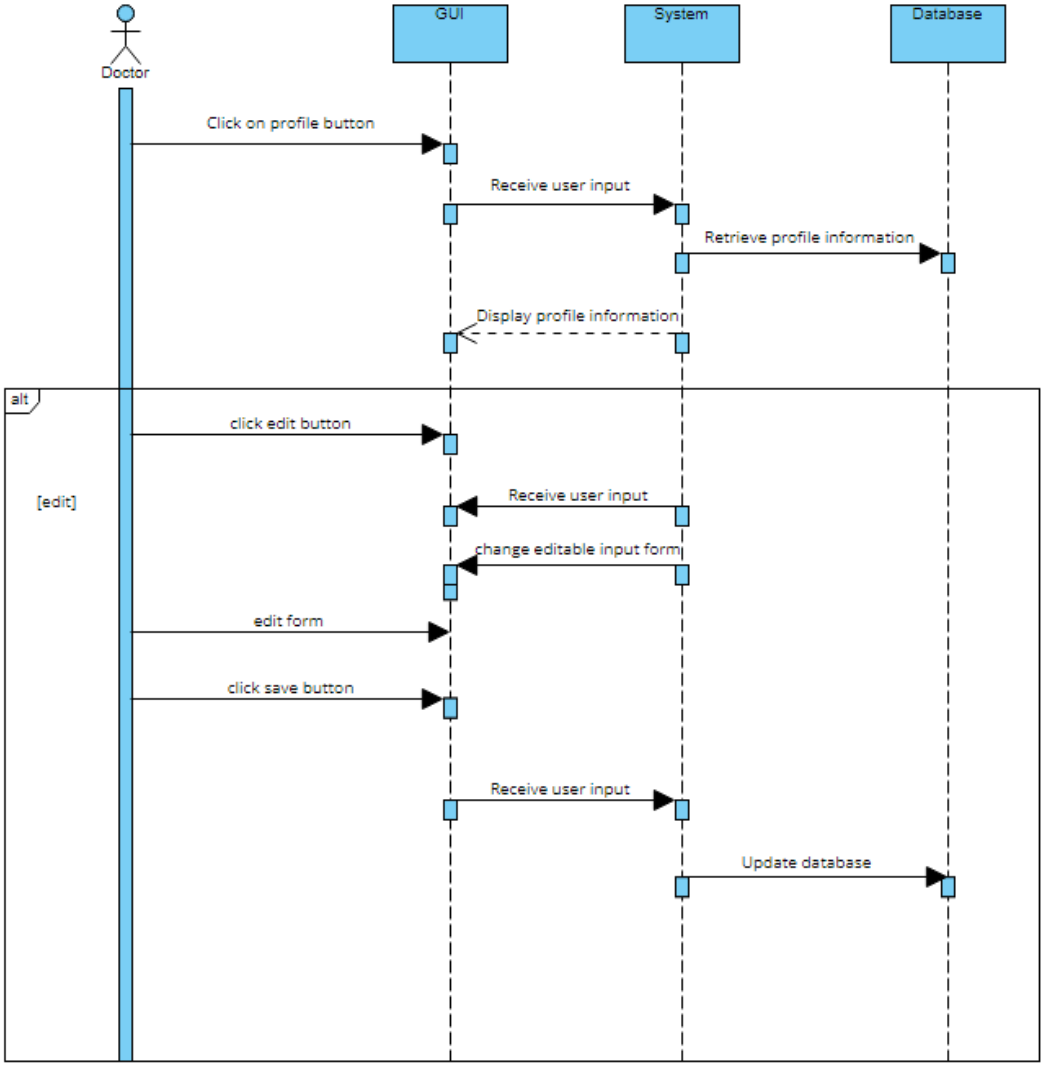
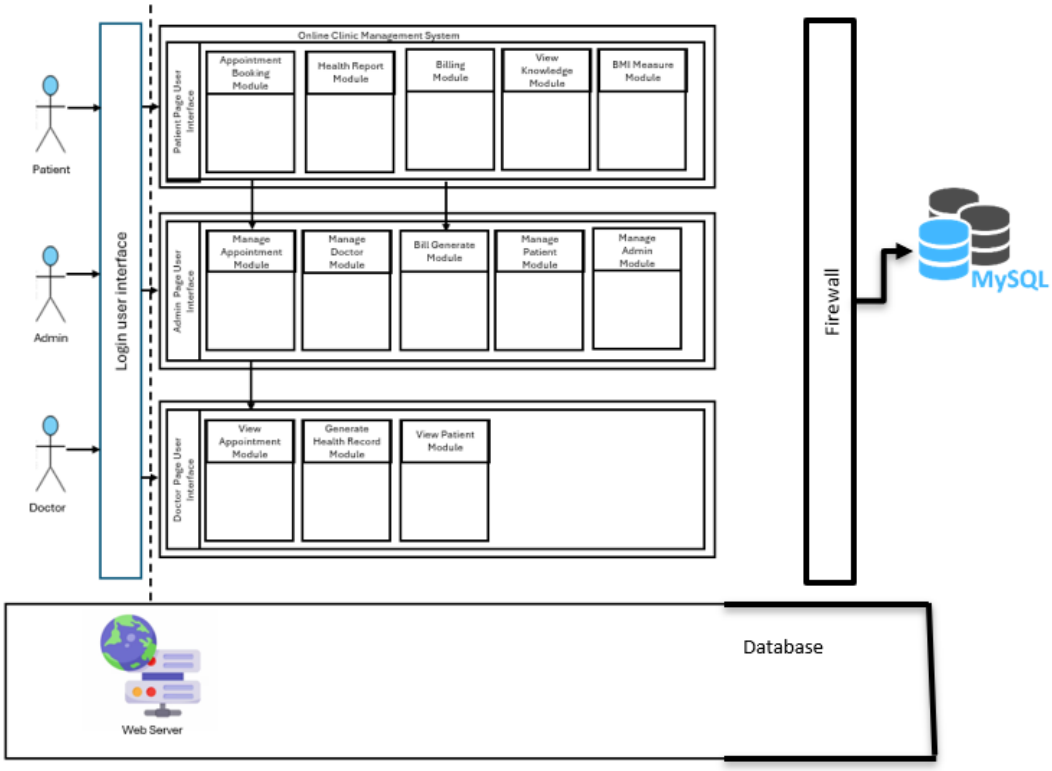


Figure 3.2.5.13.1 View personal information Sequence Diagram– Doctor Perspective

Chapter 4 System Implementation

4.1 System Design

System Architecture Diagram



The web-based clinic management system is designed to streamline healthcare operations by integrating three main user roles, patients, doctors, and administrators. Each accessing the system through a secure login interface with role-specific functionalities. Patients can book appointments in real time, view their health reports, check billing information, and use a BMI measurement tool for personal health tracking. Administrators have access to modules that allow them to manage and monitor appointments and patient BMI data, ensuring operational efficiency. Doctors can view their appointment schedules, access patient profiles, and generate medical reports to support diagnosis and treatment, although the updated diagram simplifies this section visually. All interactions are handled by a centralized web server, which communicates with a secure MySQL database protected by a firewall. This architecture ensures data integrity, security, and efficient access to clinical functions, enabling a seamless experience for all users and supporting effective clinic management.

4.2 System Main Page

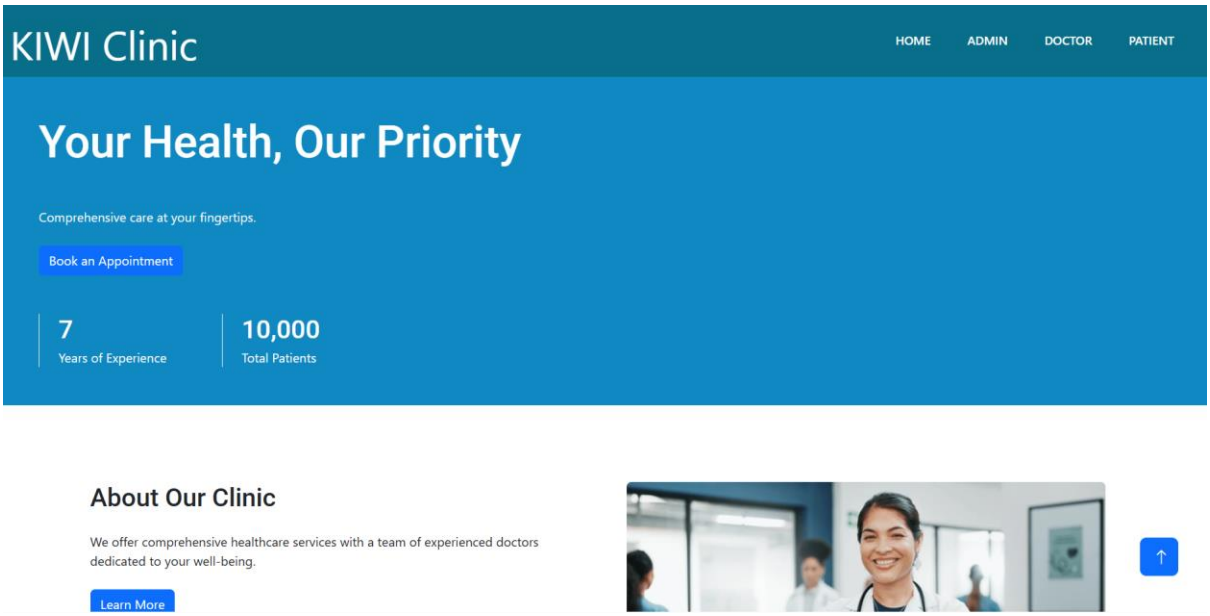


Figure 4.2.1 Clinic Main Page

Based on the System Main Page, there are rough introductions to the clinic, enabling both 3 user to login to their own page.

4.3 Admin Page

4.3.1 Admin Login

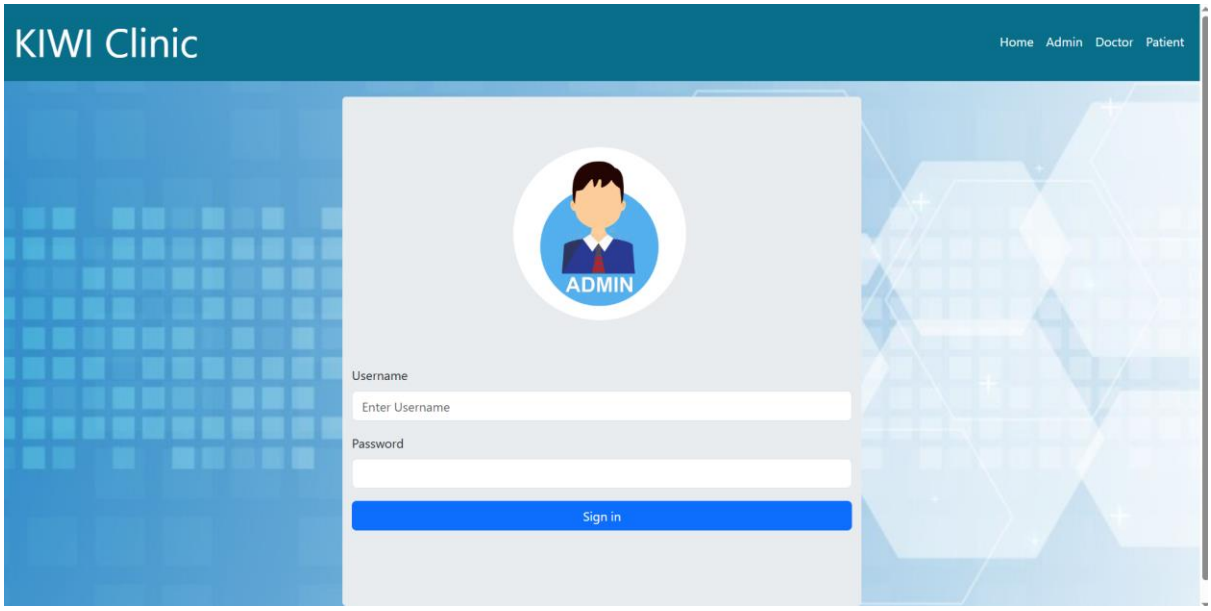
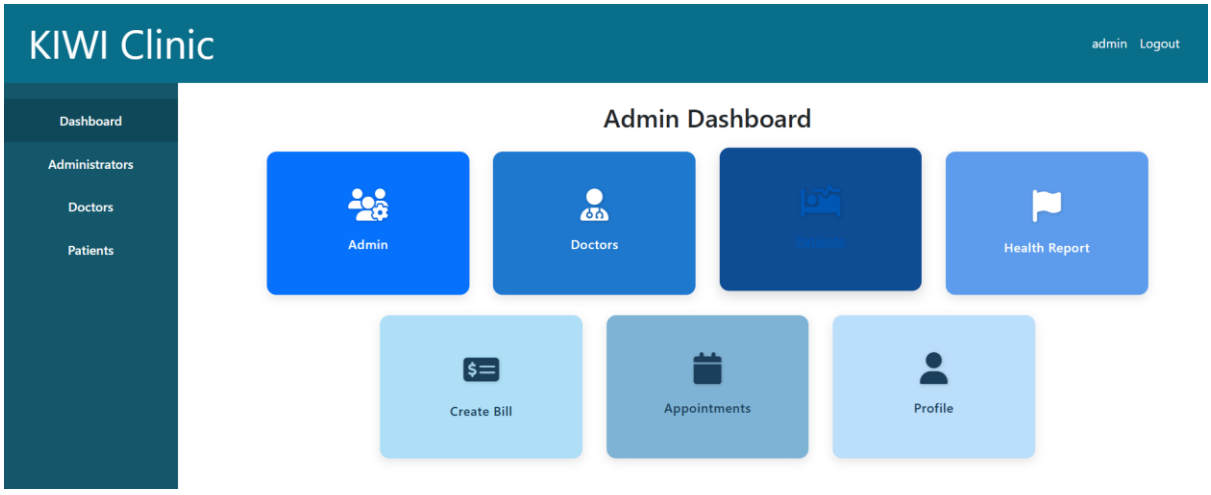


Figure 4.3.1.1 Admin Login Page

Based on Figure 4.2.1.1, this is the result when the admin clicks on the admin button on the system main page, and it will direct the admin to this page. In this page, admin is required to fill in their username and password to login to their dashboard page.

4.3.2 Admin Main Page



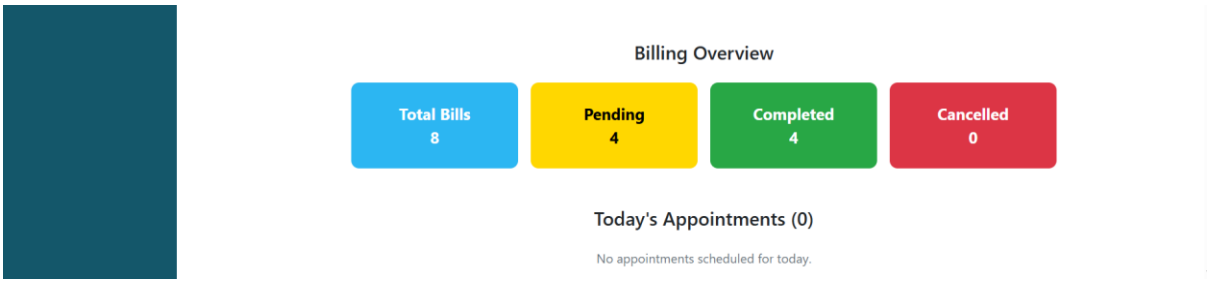


Figure 4.3.2.1 Admin Dashboard

Figure 4.2.1.1 showed the admin dashboard, which contains a few function such as manage admin, manage doctors, manage patient reports, create patient bill, manage appointments and view admin profile.

4.3.3 Admin Profile

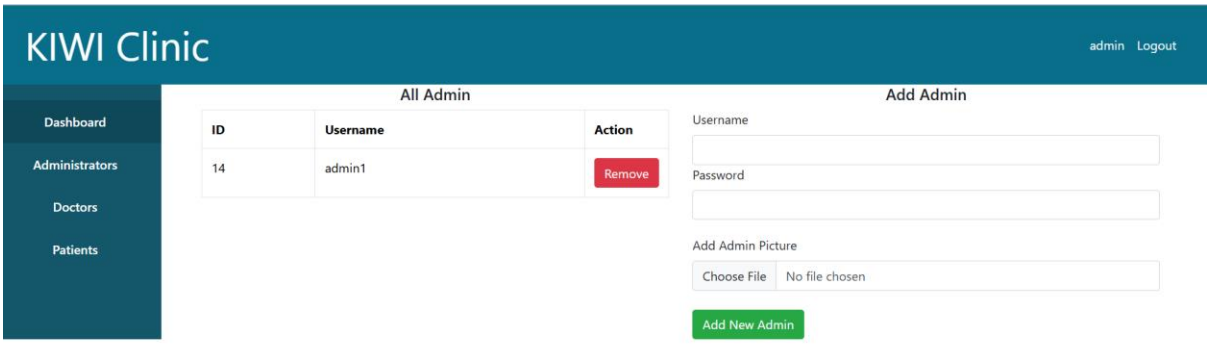


Figure 4.3.3.1 Admin Profile

Figure 4.3.3.1 showed the admin profile which enable admin to add their new admin and manage other admin.

4.3.4 Manage Doctor

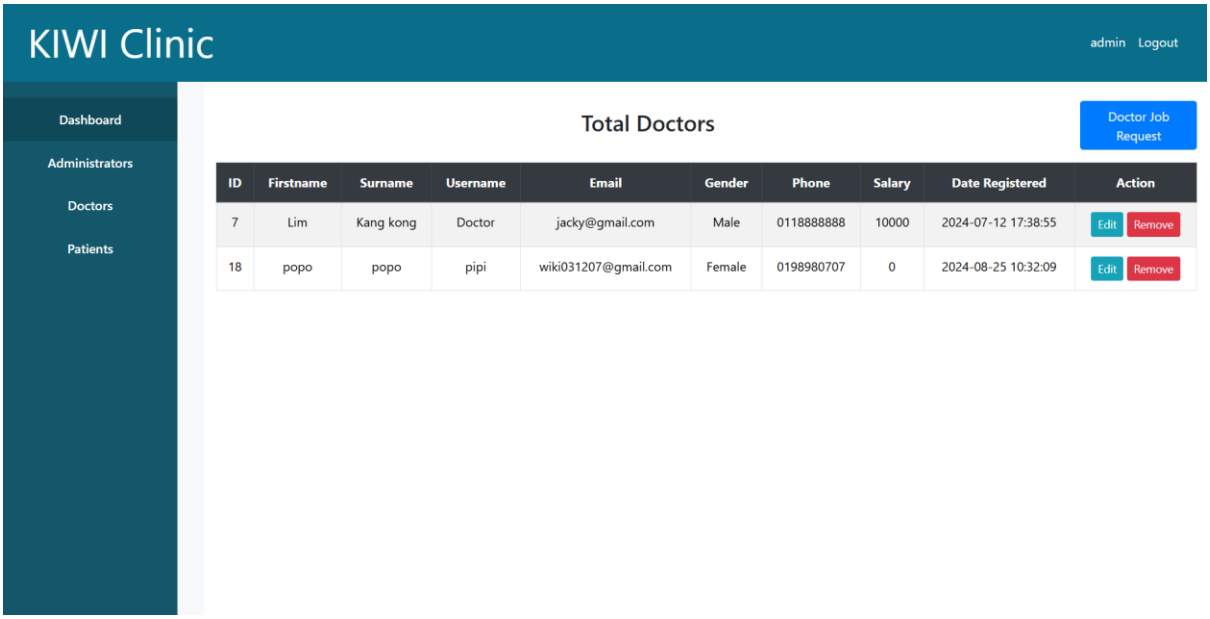


Figure 4.3.4.1 Manage Doctor Page

Figure 4.2.3.1 showed the doctor management page in admin perspective. In this page, the admin can manage doctor by viewing, editing and removing the doctor. Other than that, there is another button “Doctor Job Request” which enable admin to approve doctor job request when doctor request to apply an account.

4.3.5 Manage Patient

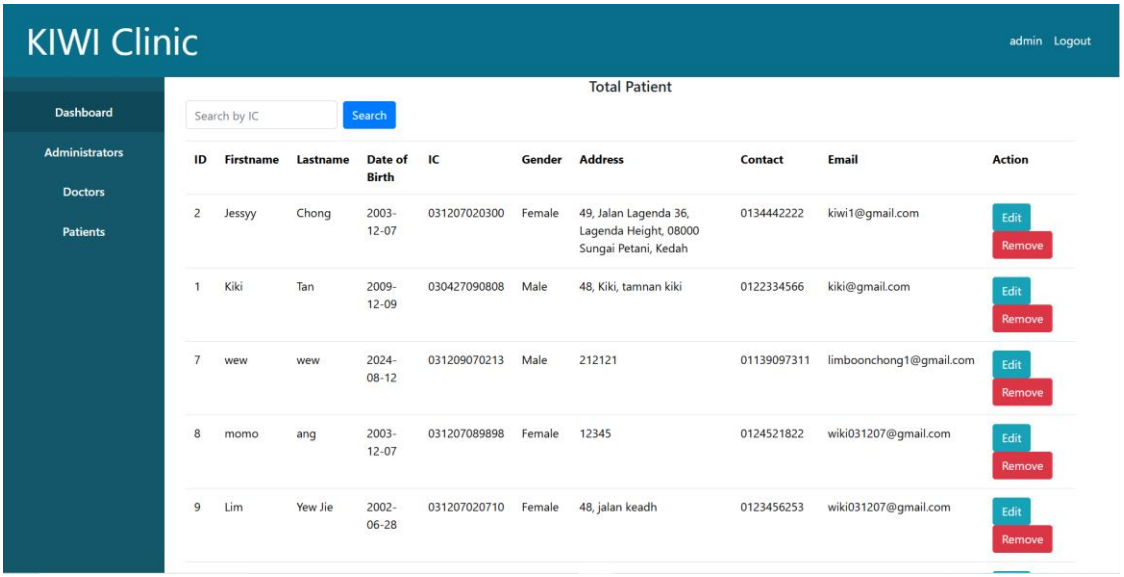


Figure 4.3.5.1 Manage Patient Page

The figure above shows the patient management in an admin perspective which admin can either modify or delete patient information. Other than that, an admin can search patients by entering their IC number.

4.3.6 Manage Patient Health Report

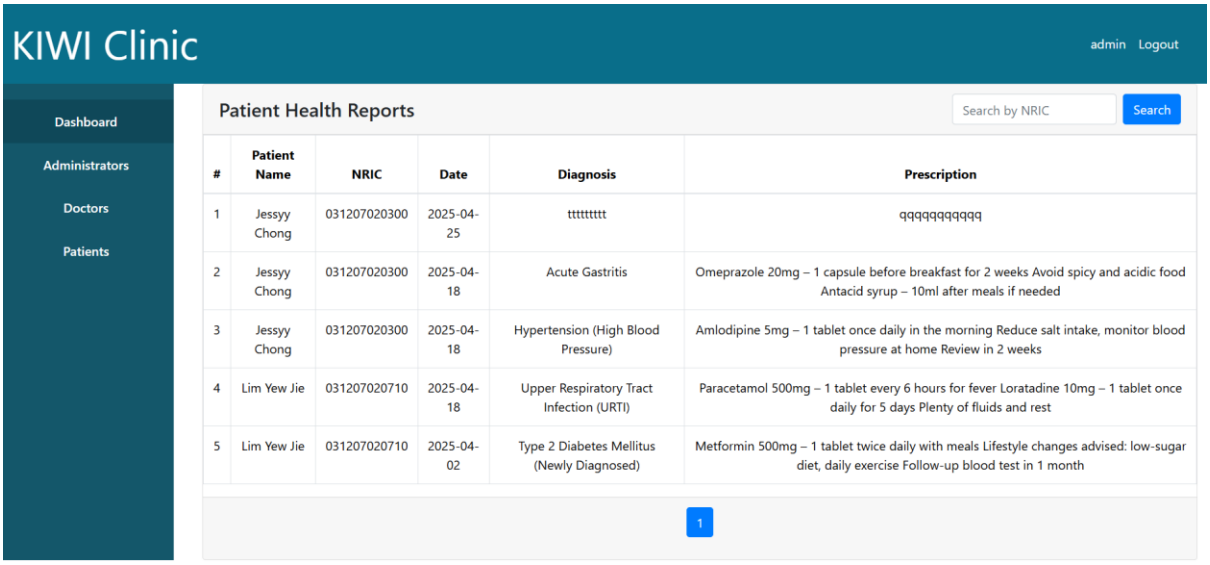


Figure 4.3.6.1 Manage Patient Health report Page

The figure 4.2.5.1 above showed the patient health reports generated by the doctor in admin perspective. In this page, admin is unable to edit the health reports, but they can view all report and print it out. There also contain search function for them to search for the report they want.

4.3.7 View Patient Health Report Detail

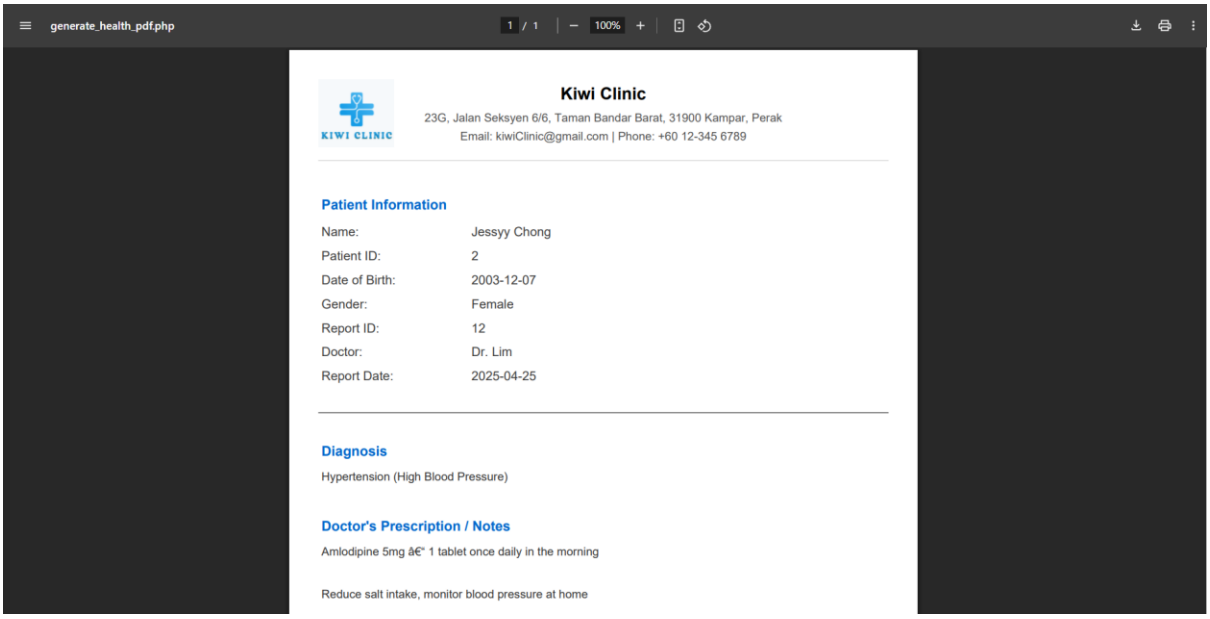


Figure 4.3.7.1 Patient Health Report Detail Page

Based on the figure above, the admin can view the health report details, and either download it or print it out as hardcopy.

4.3.8 Manage Patient Bill

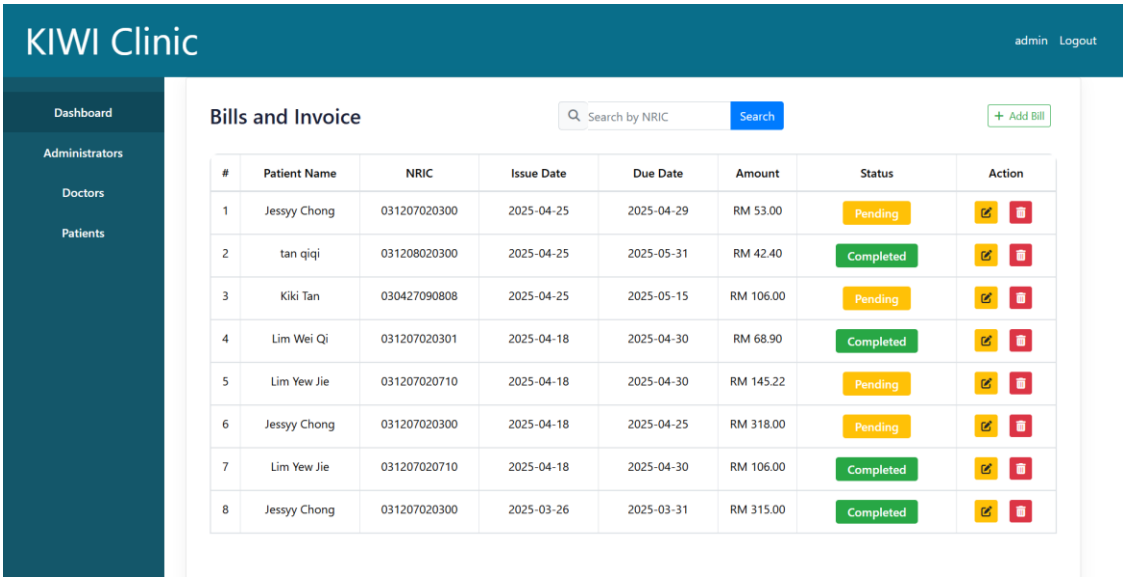


Figure 4.3.8.1 Manage Patient Bill Page

Based on the bill and invoice page above, the admin can view all patient bill in the page and modify , and delete it. Other than that, they can create new bill by clicking on the add bill button on the top right corner above.

4.3.9 Add Patient Bill

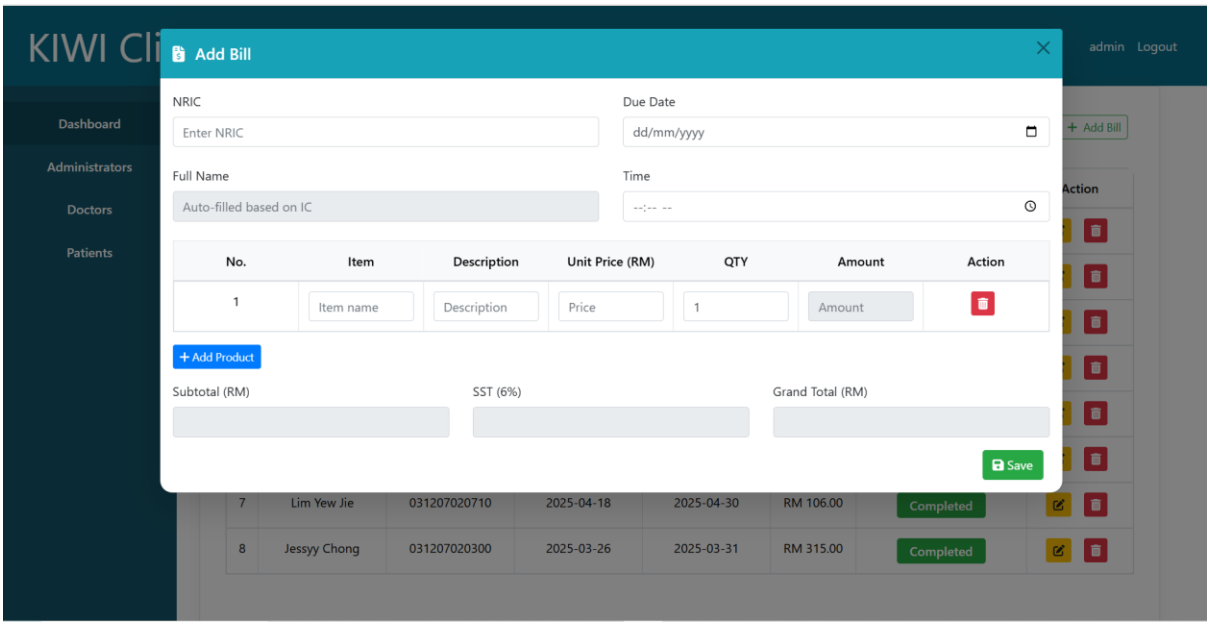


Figure 4.3.9.1 Add Patient Bill Page

When the admin click on the add bill button, a window will pop up to let admin fill in the patient ic number, if the ic number is valid and present in database, the form will automatically display the patient name which matches with the ic number. Besides, admin can key in the product in the form, key in the description, price, quantity and amount. The system will automatically add 5% of SST tax and generate the total amount.

4.3.10 Edit Bill Status

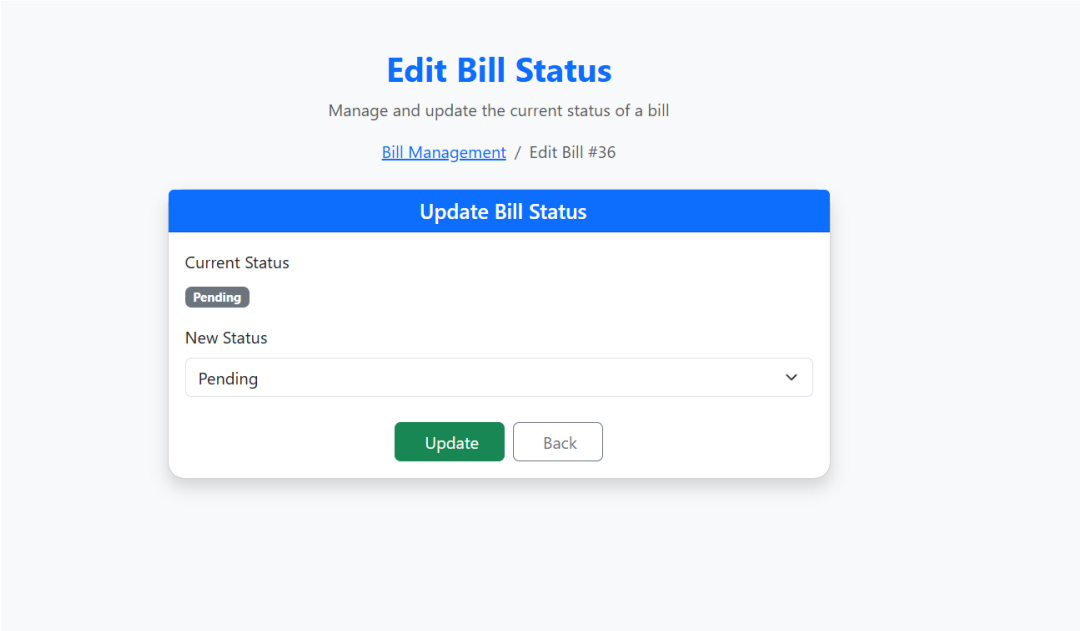


Figure 4.3.10.1 Edit Bill Status Page

Based on the figure above, after the admin receive the patient payment, confirm they already paid, admin can change the billing status to “completed”. They can change the status whether to completed, pending, or rejected.

4.3.11 Manage Patient Appointments

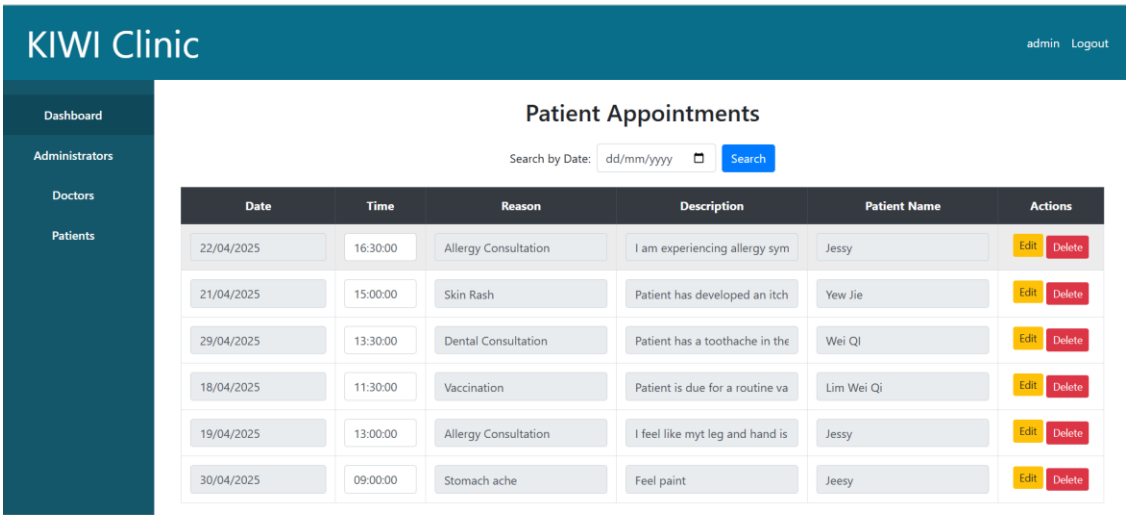


Figure 4.3.11.1 Manage Patient Appointment Page

Based on the figure 4.2.101 above, the admin can view all patient appointments and search the appointments by entering the date. Furthermore, admin can edit the appointments, and also delete the appointments.

4.3.12 Edit Patient Appointments

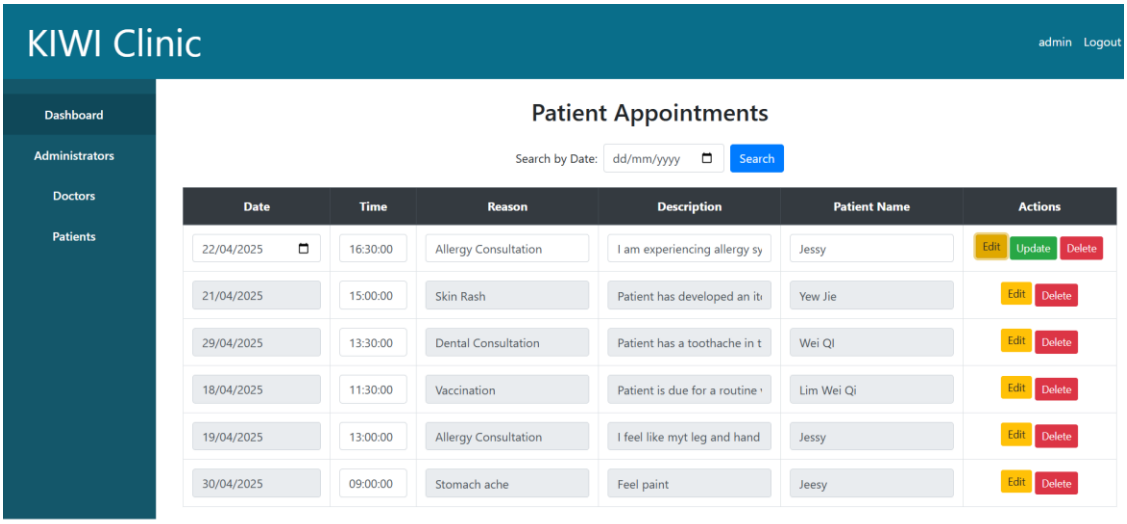


Figure 4.3.12.1 Edit Patient Appointment Page

When the admin click on the edit button, the update button will occurred and the field will become editable and the admin can edit the appointment information, then update it.

4.3.13 Manage Admin Profile

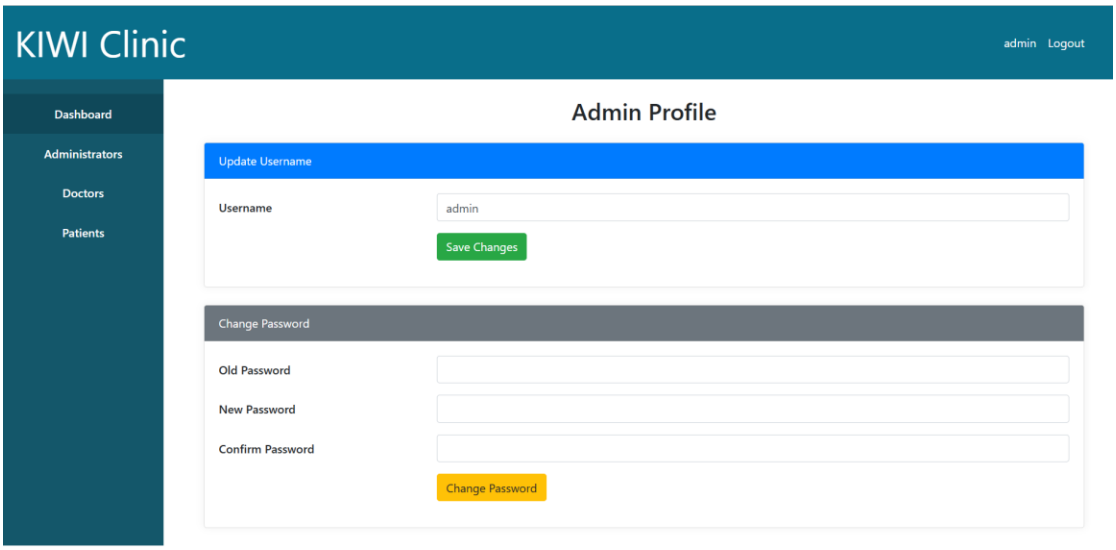


Figure 4.3.13.1 Manage Admin Profile Page

Based on the figure above, the admin can manage it own profile by changing the password and username.

4.4 Doctor Page

4.4.1 Doctor Login Page

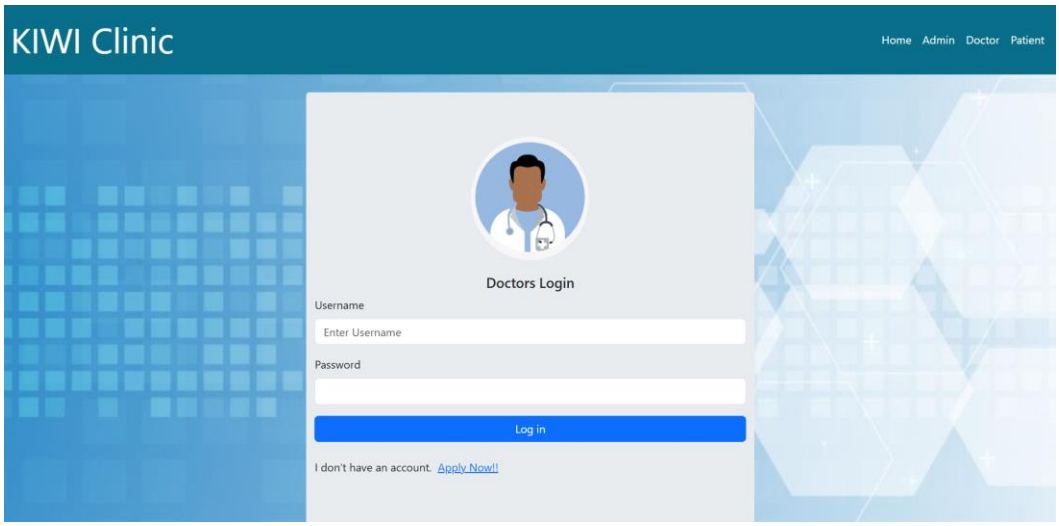


Figure 4.4.1.1 Doctor login Page

Figure 4.3.1.1 showed the doctor's login page. The doctor can login to their account by key in their username and password. If they have no account, they can create a new account under the apply now button, but they need to wait for admin approval after apply the account to make sure he/she is a real doctor.

4.4.2 Doctor Dashboard

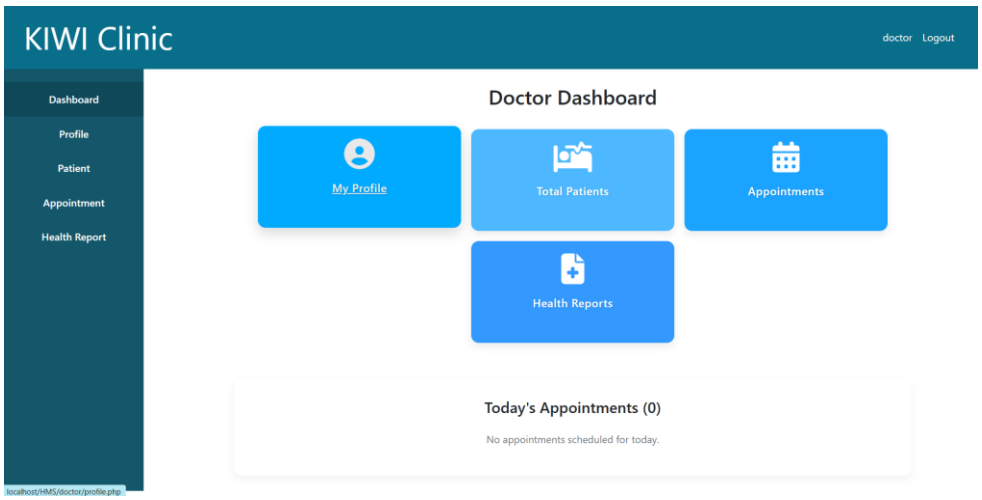


Figure 4.4.2.1 Doctor Dashboard Page

The figure above shows the doctor dashboard which contains doctor's function such as doctor's profile, total patients, appointments, and health reports. Under the page will display appointments today so that the doctor can know how many appointments today.

4.4.3 Doctor Profile Page

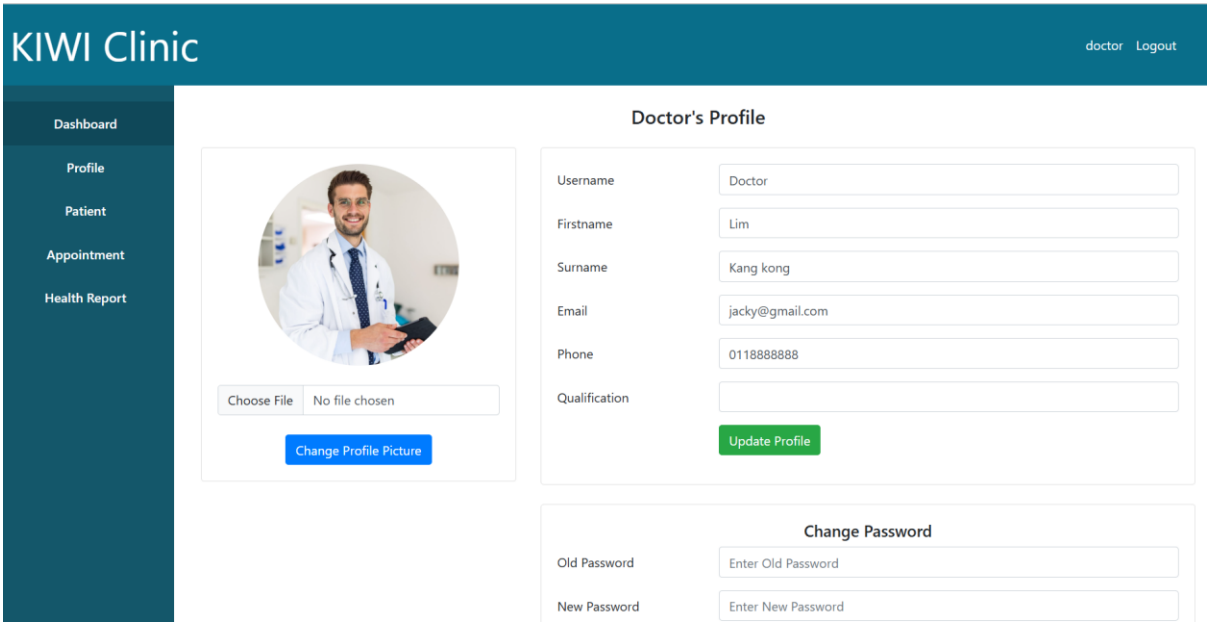


Figure 4.4.3.1 Doctor Profile Page

The figure above shows the doctor's profile in which doctor can view their profile picture, update their profile picture, edit their first name, surname, email, phone and qualification. Besides, doctors can change his/her password on this page.

4.4.4 View Patient Page

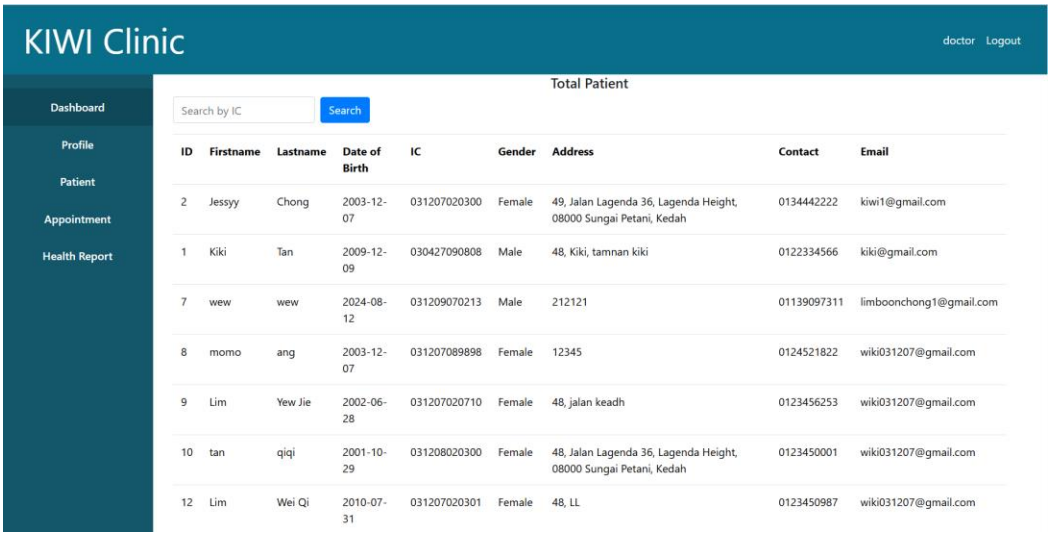


Figure 4.4.4.1 View Patient Page

Based on the figure above, the doctor can view all the patient information in the page. The doctor can search for the patient by entering the patient IC number and search for the patient he wants. However, doctor cannot edit patient information and only admin can edit patient information.

4.4.5 View Appointment Page

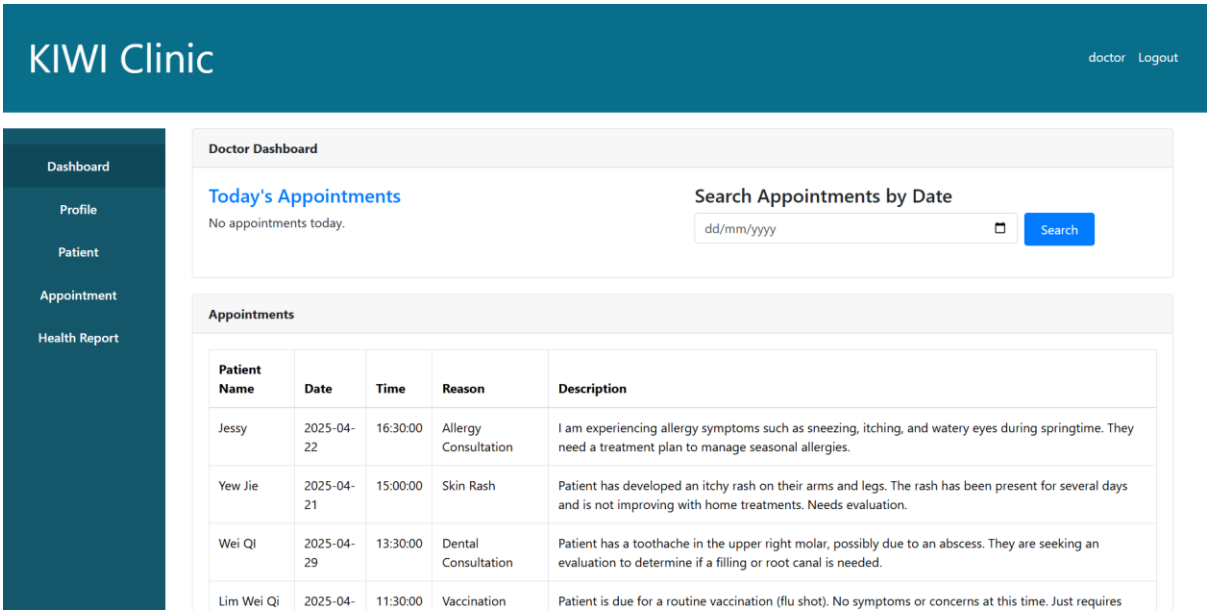


Figure 4.4.5.1 View Appointment Page

In this page, the doctor can view all the patient appointments and today appointments. Other than that, the doctor can search for patient appointments by entering the appointment date he wants to search for, and the system will filter the patient appointment.

4.4.6 Manage Health Report Page

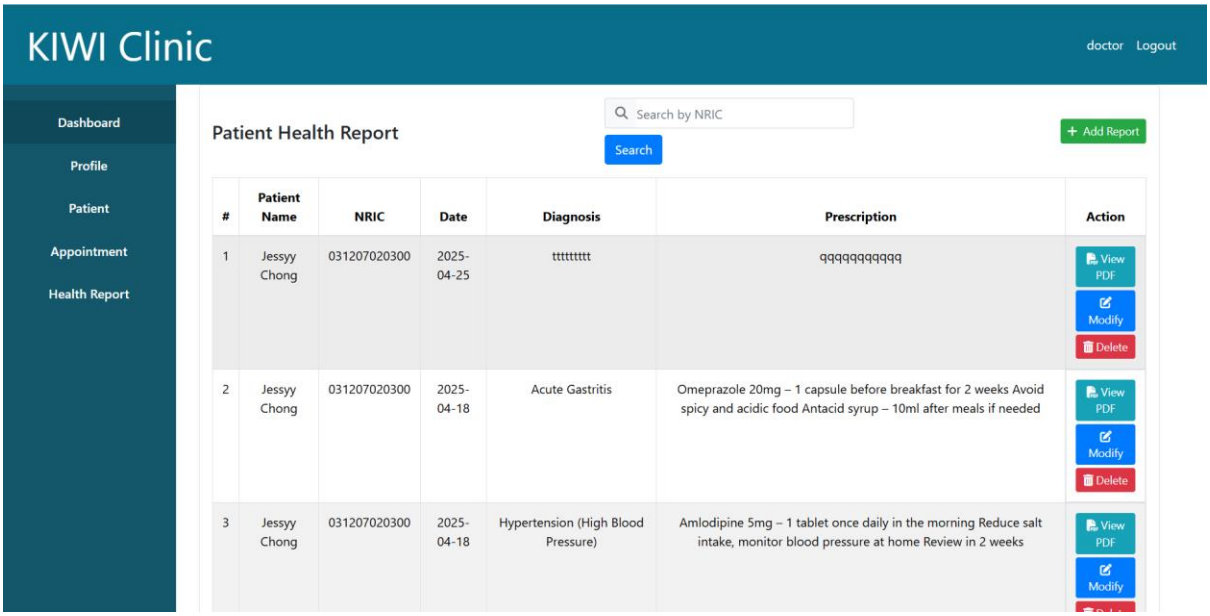


Figure 4.4.6.1 Manage Health Report Page

Based on the figure above, the doctor can manage the patient health report by adding the patient health report, modifying the health report, and also deleting the report. Other than that, he/she can view the report in pdf form by clicking on the view pdf button.

4.4.7 Add Health Report Page

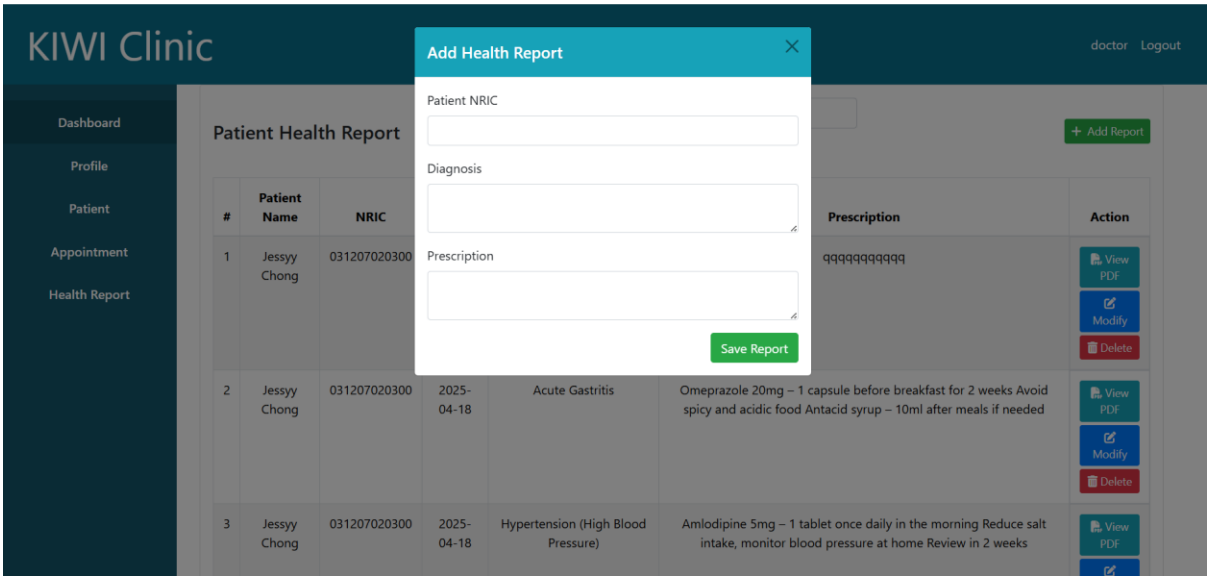


Figure 4.4.7.1 Add Health Report Page

The figure above shows the add health report page. The doctor can enter the report details including the patient IC, diagnosis, and prescription. After click on the save report button, the report will be generated and display in the view report page.

4.5 Patient Page

4.5.1 Patient Dashboard

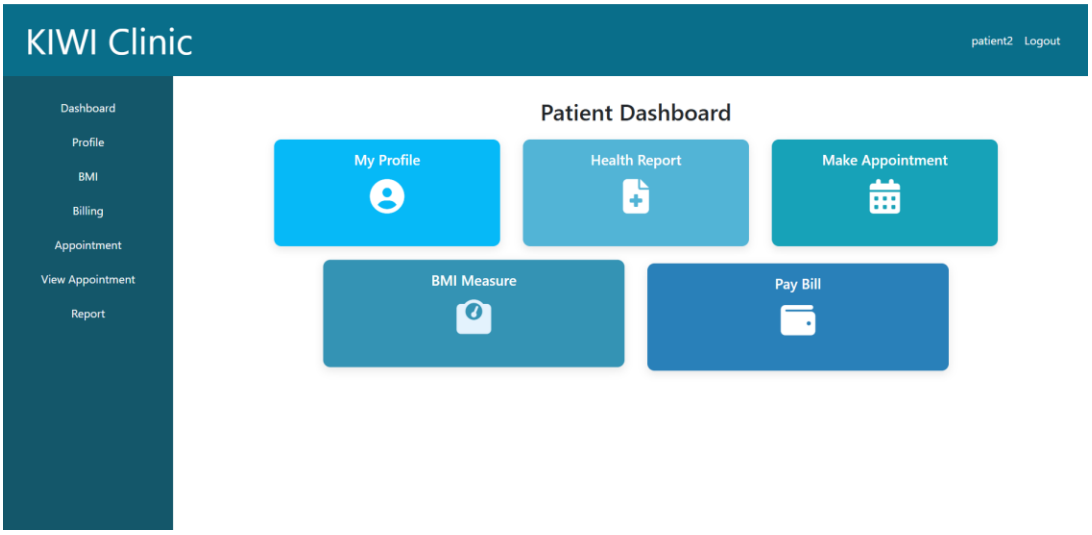


Figure 4.5.1.1 Patient Dashboard

The figure above shows the patient dashboard containing the profile function, health report, make appointment, BMI measure and pay bill function.

4.5.2 Patient Profile Page

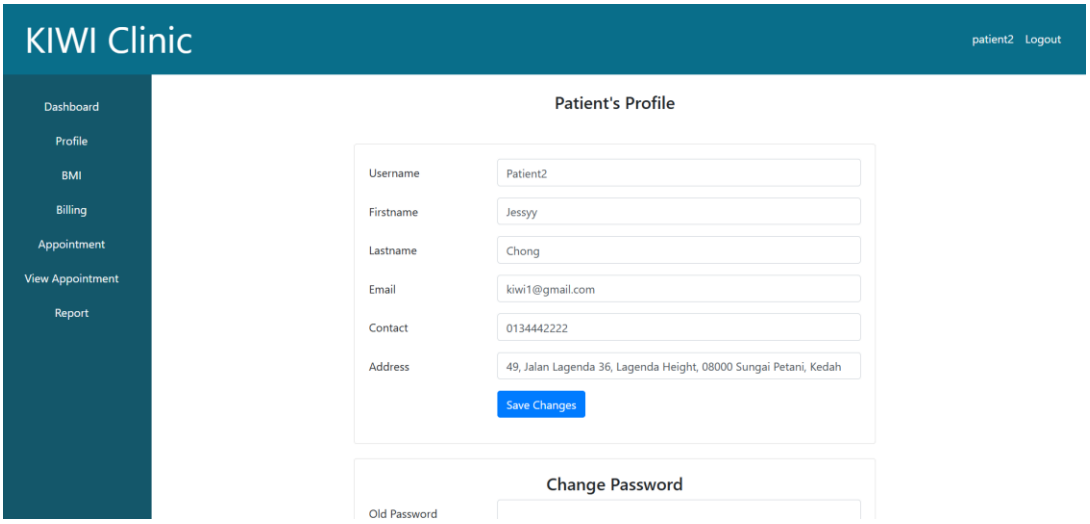


Figure 4.5.2.1 Patient Profile Page

The figure above shows the profile page from a patient perspective. In the page, patient can edit their account information to change their account details.

4.5.3 View Health Report page

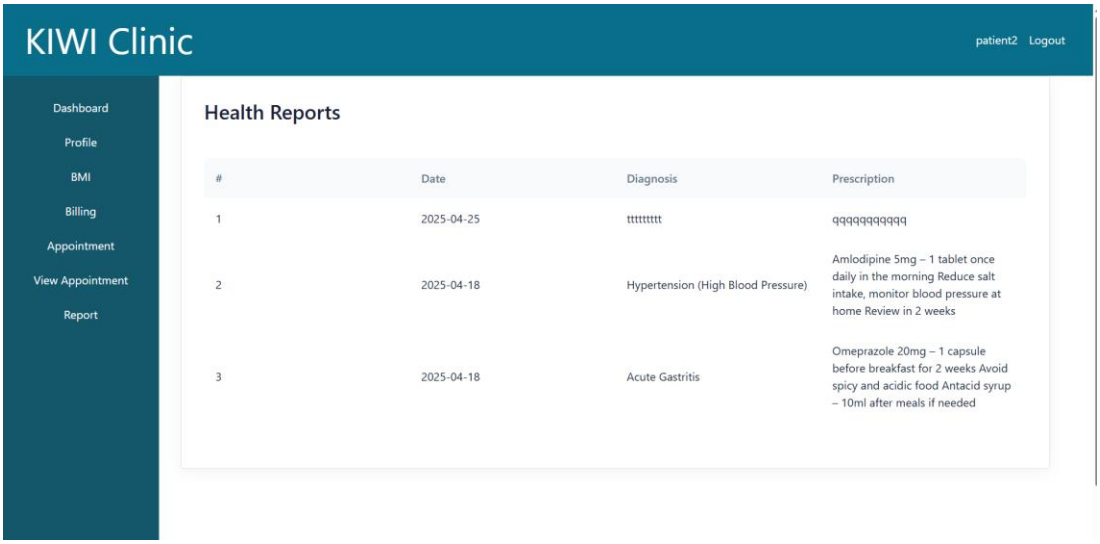


Figure 4.5.3.1 View Health Report

The figure above shows the patient health reports. By clicking on the reports the health reports will be generated in pdf form for patient to download and print the health reports. (figure 4.4.3.2)

Figure 4.5.3.2 View Health Report Details

4.5.4 Appointment Page

Figure 4.5.4.1 Appointment Page

The figure above shows the book appointment page in the patient perspective. The patient can select the date from the calendar above, a window will pop up as below(Figure 4.4.5.1) and the patient is required to fill in the appointment details to make an appointment.

4.5.5 Add New Appointment

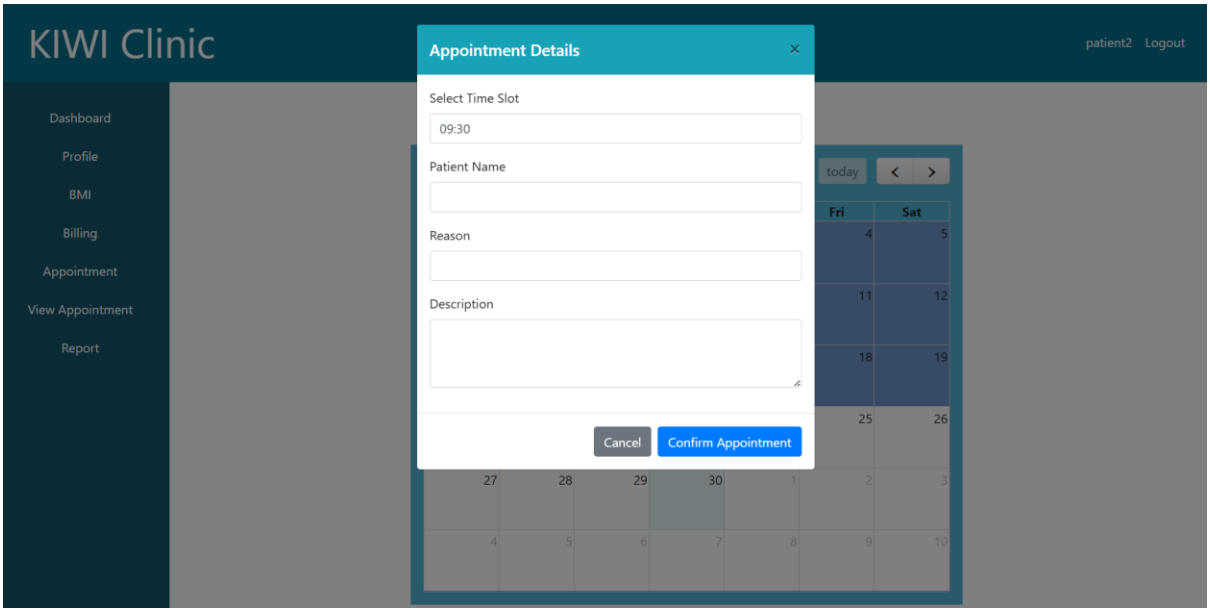


Figure 4.5.5.1 Add Appointment Page

4.5.6 View Appointment Page

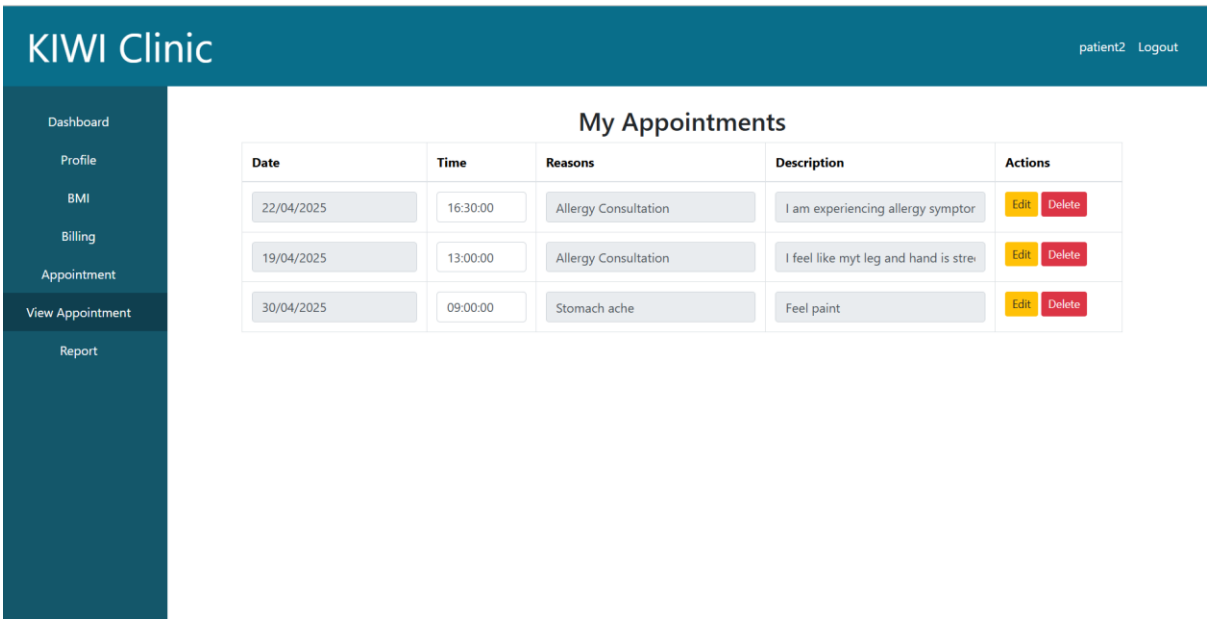
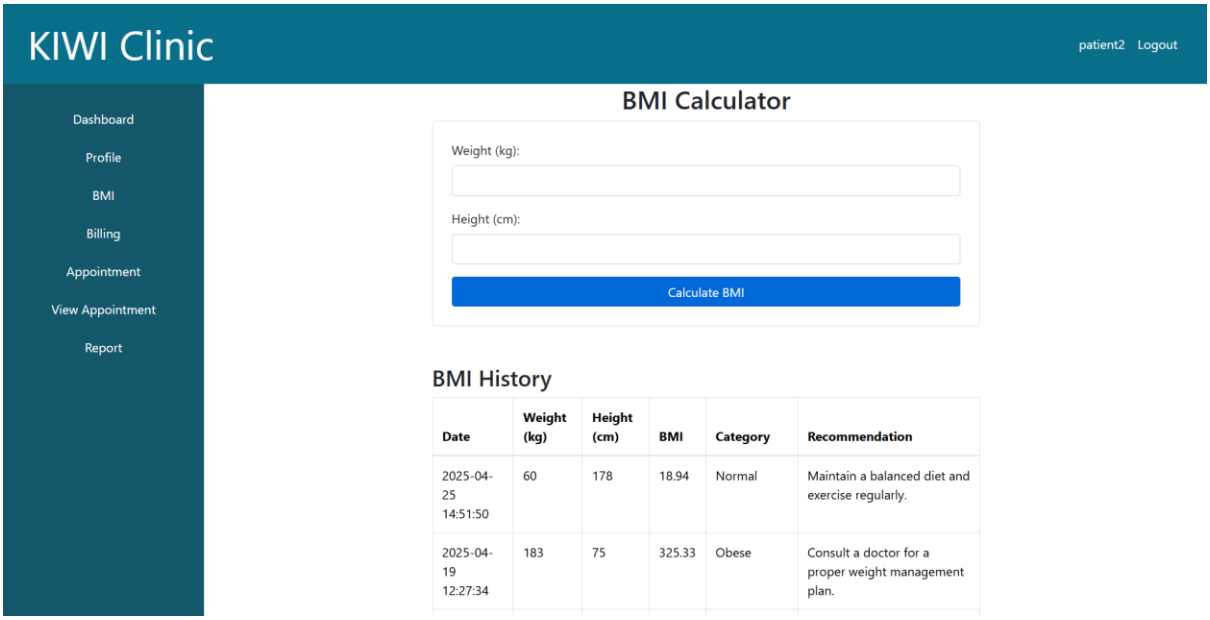


Figure 4.5.6.1 View Appointment Page

Based on the figure above, patients can view their appointment by click on the view appointment button in the side navigation bar and the appointment will displayed to them. They can edit and delete the appointment to change their appointment date and details.

4.5.7 BMI Calculation Page



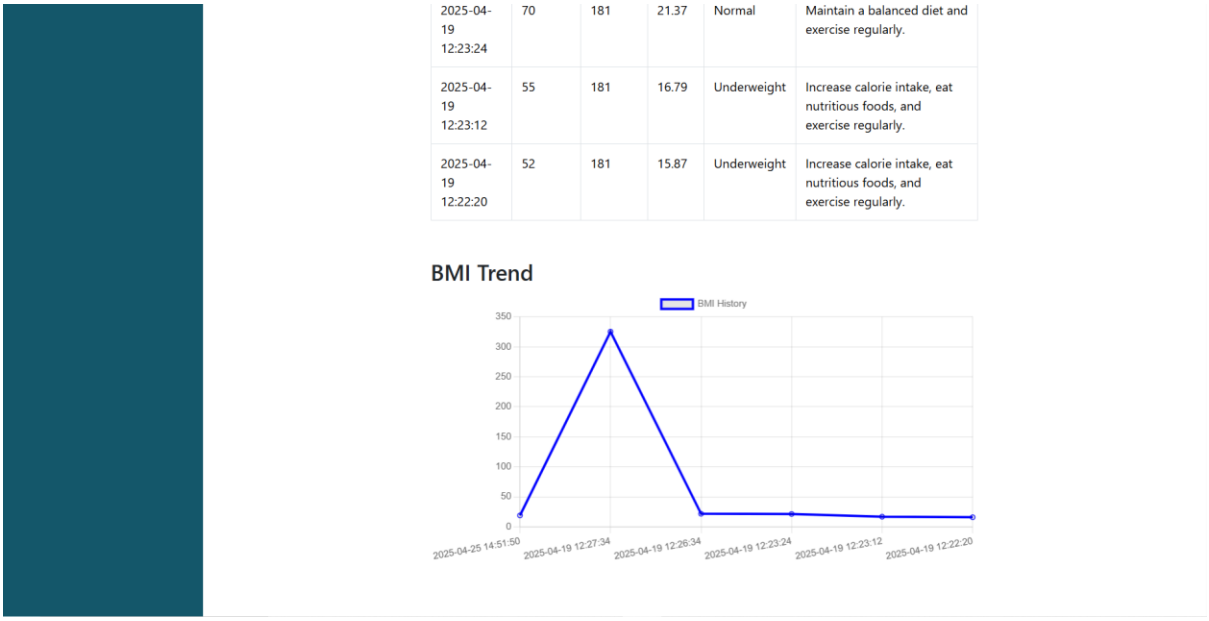


Figure 4.5.7.1 BMI Calculation Page

Based on the figure above, patients can calculate their BMI by entering their weight and height details in the form above. After they click on the calculate button, the record will saved and display below the table. Besides, the patient can monitor their BMI trend to observe their BMI performance. The recommendation will be given based on different results.

4.5.8 Billing and Payment Page

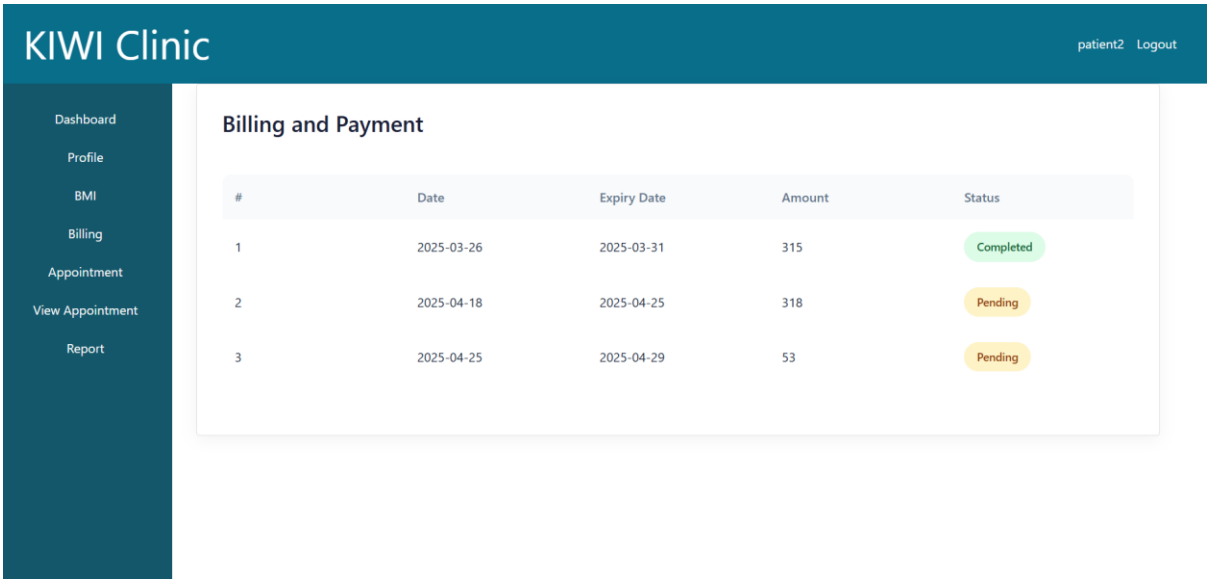


Figure 4.5.8.1 Billing and Payment Page

Based on the figure above, the patient can view their bill on the billing page. The bill will display each status to let the patient know whether the bill is paid. By clicking on those bills, bill details in the pdf form will be displayed and the patient can print the bill from the page below, also make payment through the page. (Figure 4.4.9.1)

4.5.9 Bill Details Page & Payment Page

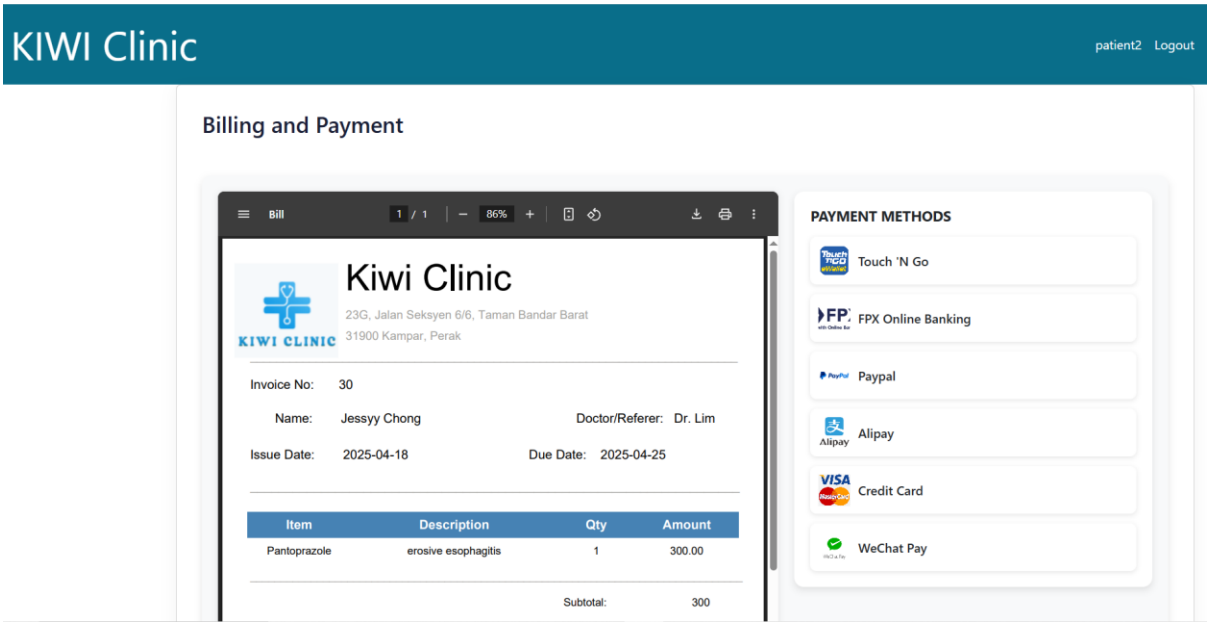


Figure 4.5.9.1 Bill Details Page & Payment Page

Chapter 5 System Evaluation and Discussion

5.1 White Box Testing

White box testing was conducted to evaluate each functional module in the clinic management system across different user roles—Admin, Doctor, and Patient. This testing approach focused on validating the flow of inputs through the code, ensuring expected outputs, and verifying that each feature performs as designed based on the underlying implementation. The system's core functionalities, including billing, appointment scheduling, health report generation, and BMI tracking, were systematically tested under the respective user interfaces. Each test case involved direct interaction with specific functions to ensure the accuracy of data processing, responsiveness of user actions, and the reliability of operations such as saving, editing, searching, and generating PDF outputs.

5.1.1 Admin Page Testing

Bill Function Testing

Function	Expected Output	Actual Output
Auto-filled function after entering patient IC in add bill page	Patient Name Occurred after enter patient IC	Passed
Automatic calculated total bill amount by adding 6% of SST and total up the product amount	Correct amount generated in grand total, SST and subtotal column.	Passed
Successfully saved patient bill after clicking on saved button	Patient bill displayed in the total bill page	Passed
Able to change patient bill status	After changing the status, the bill page will display a different status	Passed
View bill in pdf format when clicking on the patient bill row	PDF is generated and print and download function are available	Passed

Search patient bill by entering patient NRIC	Only the bill which match the patient NRIC will occurred in the search page display	Passed

Appointment Function Testing

Function	**Expected output**	**Actual Output**
Search function is working by entering the appointment date to search appointment	Appointment related to the search date is displayed	Passed
Edit Patient Appointment details by clicking on the edit button	After editing, the information of appointments in the display page will be changed	Passed

Health Report Function Testing

Function	**Expected Output**	**Actual Output**
All the patient health reports are displayed in the report page	The report information is showed	Passed
The search function is working by entering patient IC number to search	The patient report that contains the IC number searched will displayed in the page	Passed
View report in pdf format when clicking on the patient report row	PDF is generated and print and download function are available	Passed

5.1.2 Doctor Page Testing

Appointment Function Testing

Function	Expected output	Actual Output
Search function is working by entering the appointment date to search appointment	Appointment related to the search date is displayed	Passed
View Patient appointment on the main page	Patient appointments are displayed on the page	Passed
Today appointment dashboard is displayed	Today's appointment is shown at the top of page	Passed

Health Report Function Testing

Function	Expected Output	Actual Output
Successfully saved patient report after clicking on saved button	Patient report displayed in the total report page	Passed
Able to change patient report details	After changing the status, the report page will display the newest update information	Passed
View report in pdf format when clicking on the patient report row	PDF is generated and print and download function are available	Passed
Search patient report by entering patient NRIC	Only the report which matches the patient NRIC will occur in the search page display	Passed

5.1.3 Patient Page Testing

Health Report Function Testing

Function	Expected Output	Actual Output
All health reports are displayed in the patient health page	Patient report displayed in the total report page	Passed
View report in pdf format when clicking on the patient report row	PDF is generated and print and download function are available	Passed

Bill Function Testing

Function	Expected Output	Actual Output
All bills are displayed in the patient bill page with the status of bill	Patient bill displayed in the total bill page with status	Passed
View report in pdf format when clicking on the patient report row	PDF is generated and print and download function are available	Passed

Appointment Function Testing

Function	Expected Output	Actual Output
Able to select the appointment date and fill in the appointment information	After filled in, the appointment is make and saved.	Passed
Able to change patient appointment details	After changing the details, the appointment displayed page will display the newest update appointment information	Passed

BMI Calculation Function Testing

Function	Expected Output	Actual Output
The BMI result is generated after entering the weight and height	The result is generated in the history side	Passed
Able to display patient BMI recommendation to different result	The result is displayed in history with the result of BMI	Passed
The BMI Trend is displayed to show patient BMI changes	The line graph is displayed successfully	Passed

CHAPTER 6

6.1 Conclusion

The goal of the proposed project is to create a Clinic Management web-based system for clinics. This application's functions include online appointment scheduling for patients and additional BMI measuring tools for the patients and users. Additionally, all the features and services offered by this project ought to enable users to experience and benefit from less costly, superior, and real-time services through web applications, elevating their overall customer experience.

Real-time appointment scheduling, patient health report generating, patient and online billing, and additional BMI measuring tools are the main functions of this system. All medical records will also be computerized so that users can access and see data from the system with ease. As a result, with this clinic management system, all previously manual procedures will be online and automated, enabling users to access clinic anytime and anywhere.

Similar online clinic management systems, including those from the UR Clinic, BMI clinic and Care Clinic will be analyzed, evaluated, and reviewed. As a result, the findings will be utilized as references to help improve the features and functionalities of the proposed system based on the analytical results of the literature review.

At last, some additional features are added to advance the doctor and admin to access to the patient record and appointments. They have their own account and interface for them to make a better management on the clinic patient to provide a great service and increase their consultation experience.

6.2 RECOMMENDATION

The web-based clinic management system is recommended that the system be further enhanced with user-friendly interfaces, role-based access control, and responsive design to ensure smooth and secure interactions for admins, doctors, and patients across all devices. Implementing robust data encryption and audit trails can help maintain data integrity and confidentiality, especially in billing and medical information. Additionally, integrating automated reminders for appointments, real-time status updates, and comprehensive billing history for patients will improve engagement and reduce no-shows or payment delays. For long-term scalability, the system should be designed with modular architecture, enabling the addition of more features like e-prescriptions, medical record tracking, and analytical dashboards in the future. Continued user testing and feedback from all three user roles should guide iterative improvements, ensuring the system remains aligned with the needs of modern clinics and enhances overall healthcare delivery efficiency

REFERENCES

[1] J. Muhammad and S. Garba, "Web-based Clinic Management System (CMS)," *International Journal of Science and Engineering Applications*, vol. 8, no. 5, pp. 131–135, May 2019, doi: 10.7753/ijsea0805.1001

[2] R. T. Sutton, D. Pincock, D. C. Baumgart, D. C. Sadowski, R. N. Fedorak, and K. I. Kroeker, "An overview of clinical decision support systems: benefits, risks, and strategies for success," *Npj Digital Medicine*, vol. 3, no. 1, Feb. 2020, doi: 10.1038/s41746-020-0221-y.

[3] "Health care information systems," *Google Books*. https://books.google.com.my/books?hl=en&lr=&id=io1SEAAAQBAJ&oi=fnd&pg=PR15&dq=benefit+of+clinic+management+system+in+billing&ots=X6Hjko96e8&sig=i3qO1J5MAUMWn1cs8oStq5kLLaU&redir_esc=y#v=onepage&q=benefit%20of%20clinic%20management%20system%20in%20billing&f=false

[4] D. Plus, "What is a Clinic Management System and Why is it Important?," *Medium*, Jan. 13, 2024. [Online]. Available: https://medium.com/@drlogydigit/what-is-a-clinic-management-system-and-why-is-it-important-dfacc6246fbd#:~:text=Cost%20Transparency%3A%20Offers%20transparency%20in,financial%20records%20for%20the%20clinic.

[5] G. Advertising, "URKlinik | Index." https://urklinik.com/home/appointment.php

Appendices

Admin Bill

```php
<?php
    session_start();
    include("../include/connection.php");
    $errors = array();

    $count = 1;
    $date = date('Y-m-d');

    $records_per_page = 9;
    $current_page = isset($_GET['page']) ? (int)$_GET['page'] : 1;
    $offset = ($current_page - 1) * $records_per_page;

    $stmt = $connect->prepare("SELECT COUNT(*) FROM bill");
    $stmt->execute();
    $stmt->bind_result($total_records);
    $stmt->fetch();
    $stmt->close();

    $total_pages = ceil($total_records / $records_per_page);

    $stmt = $connect->prepare("SELECT * FROM bill ORDER BY bill_id DESC LIMIT ?, ?");
    $stmt->bind_param("ii", $offset, $records_per_page);
    $stmt->execute();
    $result = $stmt->get_result();

    $search = isset($_GET['search']) ? $_GET['search'] : "";
    $records_per_page = 9;
    $current_page = isset($_GET['page']) ? (int)$_GET['page'] : 1;
    $offset = ($current_page - 1) * $records_per_page;

    // Modify SQL to filter results based on search input
    $sql = "SELECT bill.*, patients.firstname, patients.lastname, patients.ic
            FROM bill
            JOIN patients ON bill.patient_id = patients.id";

    if (!empty($search)) {
        $sql .= " WHERE patients.ic LIKE ?";
    }

    $sql .= " ORDER BY bill_id DESC LIMIT ?, ?";
```

adminbill.php(1)

```php
<?php

    $sql .= " ORDER BY bill_id DESC LIMIT ?, ?";

    $stmt = $connect->prepare($sql);

    // Bind parameters based on search input
    if (!empty($search)) {
        $searchParam = "%$search%";
        $stmt->bind_param("sii", $searchParam, $offset, $records_per_page);
    } else {
        $stmt->bind_param("ii", $offset, $records_per_page);
    }

    $stmt->execute();
    $result = $stmt->get_result();

    //Add bill details to database

    if ($_SERVER['REQUEST_METHOD'] == 'POST' && isset($_POST['update_status'])) {
        $bill_id = $_POST['bill_id'];
        $new_status = $_POST['status'];

        $updateStmt = $connect->prepare("UPDATE bill SET status = ? WHERE bill_id = ?");
        $updateStmt->bind_param("si", $new_status, $bill_id);

        if ($updateStmt->execute()) {
            header("Location: adminbill.php");
            exit;
        } else {
            echo "Failed to update status: " . $updateStmt->error;
        }

        $updateStmt->close();
    }
```

adminbill.php(2)

```php
<?php

if ($_SERVER['REQUEST_METHOD'] == 'POST') {

    $nric = $_POST['patient_nric'];
    $due_date = $_POST['bill_date'];
    $issue_date = date('Y-m-d');
    $products_json = $_POST['products_json'];

    $grandTotal = $_POST["grandTotal"];
    $status = "Pending"; // Default status

    $decoded_products = json_decode($products_json, true);
    if(json_last_error() !== JSON_ERROR_NONE) {
        die("Invalid JSON format");
    }

    $billPatientStmt = $connect->prepare("SELECT id FROM patients WHERE ic = ?");
    $billPatientStmt->bind_param("s", $nric);
    $billPatientStmt->execute();
    $billPatientStmt->bind_result($patient_id);
    $billPatientStmt->fetch();
    $billPatientStmt->close();

    if ($patient_id) {
        // Insert Bill into Appointments Table
        $stmt = $connect->prepare("INSERT INTO bill (issue_date, due_date, patient_id, products, amount, status) VALUES (?, ?, ?, ?, ?
        $stmt->bind_param("ssssds", $issue_date, $due_date, $patient_id, $products_json, $grandTotal, $status);
    } else {
        echo "Patient not found.";
    }

    $connect->close();
}
?>
```

adminbill.php(3)

```
</head>
<body>
    <?php include("../include/header.php"); ?>

    <div class="container-fluid">
        <div class="col-md-12">
            <div class="row">
                <div class="col-md-2"style="margin-left: -30px;">
                    <?php
                        include("sidenav.php");
                    ?>

                </div>
                <div class="col-md-10">
                    <div class="card">
                        <div class="header d-flex align-items-center justify-content-between">

                            <h4 class="title mb-0">Bills and Invoice</h4>

                            <div class="search-box">
                                <form method="GET" action="">
                                    <div class="input-group">
                                        <span class="input-group-text bg-light border-end-0">
                                            <i class="fa fa-search text-muted"></i>
                                        </span>
                                        <input type="text" class="form-control border-start-0 ps-0" name="search"
                                            value="<?= htmlspecialchars($search) ?>" placeholder="Search by NRIC">
                                        <button type="submit" class="btn btn-primary">Search</button>
                                    </div>
                                </form>
                            </div>

                            <button class="btn btn-outline-success btn-sm add">
                                <i class="fas fa-plus me-2"></i>Add Bill
                            </button>
                        </div>
```

adminbill.php(4)

```
437   <body>
440       <div class="container-fluid">
476               <table class="table table-bordered text-center">
477               <thead>
478                   <tr>
479                       <th style="width: 3%;">#</th>
480                       <th style="width: 12%;">Patient Name</th>
481                       <th style="width: 12%;">NRIC</th>
482                       <th style="width: 12%;">Issue Date</th>
483                       <th style="width: 12%;">Due Date</th>
484                       <th style="width: 8%;">Amount</th>
485                       <th style="width: 16%;">Status</th>
486                       <th style="width: 10%;">Action</th>
487                   </tr>
488               </thead>
489
490                   <tbody>
491                       <?php
492                       if ($result) {
493                           mysqli_data_seek($result, 0);
494                           while ($row = mysqli_fetch_assoc($result)) {
495                               $formattedIssueDate = (new DateTime($row["issue_date"]))->format('Y-m-d');
496                               $formattedDueDate = (new DateTime($row["due_date"]))->format('Y-m-d');
497
498                               $patientStmt = $connect->prepare("SELECT * FROM patients WHERE id = ?");
499                               $patientStmt->bind_param("i", $row["patient_id"]);
500                               $patientStmt->execute();
501                               $patientResult = $patientStmt->get_result();
502                               $patient = mysqli_fetch_assoc($patientResult);
503                       ?>
504                               <tr onclick="window.location.href='../patient/generate_pdf.php?id=<?= $row['bill_id']; ?>'" style="cursor: poi
505                                   <td style="width: 3%;"><?= $count ?></td>
506                                   <td style="width: 12%;"><?= $patient["firstname"] . " " . $patient["lastname"]; ?></td>
507                                   <td style="width: 12%;"><?= $patient["ic"]; ?></td>
508                                   <td style="width: 12%;"><?= $formattedIssueDate; ?></td>
509                                   <td style="width: 12%;"><?= $formattedDueDate; ?></td>
510                                   <td style="width: 8%;">RM <?= number_format($row["amount"], 2); ?></td>
511                                   <td style="width: 16%;">
512                                       <div class="text-center">
```

adminbill.php(5)

```
                        <td style="width: 16%;">
                            <div class="text-center">
                                <?php
                                $status = $row["status"];
                                $badgeClass = match($status) {
                                    "Completed" => "success",
                                    "Pending" => "warning",
                                    "Cancelled" => "danger",
                                    default => "secondary"
                                };
                                ?>
                                <span class="badge bg-<?= $badgeClass ?>"
                                    style="font-size: 1rem; font-weight: 600; text-transform: capitalize; padding: 0.5em 1em;">
                                    <?= htmlspecialchars($status) ?>
                                </span>
                            </div>
                        </td>



                        <td style="width: 10%;">
                            <div class="d-flex align-items-center gap-2 justify-content-center">
                                <a href="edit_bill.php?id=<?= $row['bill_id']; ?>" class="btn btn-sm btn-warning">
                                    <i class="fas fa-edit"></i>
                                </a>
                                <a href="delete_bill.php?id=<?= $row['bill_id']; ?>" class="btn btn-sm btn-danger"
                                    onclick="return confirm('Are you sure you want to delete this bill?');">
                                    <i class="fas fa-trash-alt"></i>
                                </a>
                            </div>
                        </td>

                    </tr>
                <?php
```

adminbill.php(6)

```
                <div class="pagination">
                    <?php
                    if ($total_pages > 1) {
                        $current_page = isset($_GET['page']) ? (int)$_GET['page'] : 1;
                        if ($current_page > 1) {
                            echo '<a href="?page=' . ($current_page - 1) . '"><i class="fas fa-chevron-left"></i></a>';
                        }

                        for ($i = 1; $i <= $total_pages; $i++) {
                            echo '<a href="?page=' . $i . '" ' . ($i == $current_page ? 'class="active"' : '') . '>' . $i . '</a>';
                        }

                        if ($current_page < $total_pages) {
                            echo '<a href="?page=' . ($current_page + 1) . '"><i class="fas fa-chevron-right"></i></a>';
                        }
                    }
                    ?>
                </div>
            </div>
        </div>

                <div id="addBillModal" class="modal fade" tabindex="-1" aria-hidden="true">
                    <div class="modal-dialog modal-xl">
                        <div class="modal-content shadow-lg rounded-4">
                            <div class="modal-header bg-info text-white">
                                <h5 class="modal-title"><i class="fas fa-file-invoice-dollar me-2"></i> Add Bill</h5>
                                <button type="button" class="btn-close" data-bs-dismiss="modal"></button>
                            </div>

                            <div class="modal-body">
                                <form method="post" id="billForm">
                                    <input type="hidden" name="products_json" id="products_json">
                                    <div class="row g-3">
                                        <div class="col-md-6">
                                            <label for="nric" class="form-label">NRIC</label>
                                            <input type="text" class="form-control" id="patient_nric" name="patient_nric" placeholder
                                        </div>
```

adminbill.php(7)

```
                    <div class="col-md-6">
                        <label for="nric" class="form-label">NRIC</label>
                        <input type="text" class="form-control" id="patient_nric" name="patient_nric" placeholder=
                    </div>
                    <div class="col-md-6">
                        <label for="bill_date" class="form-label">Due Date</label>
                        <input type="date" class="form-control" id="bill_date" name="bill_date">
                    </div>
                </div>

                <div class="row g-3 mt-2">
                    <div class="col-md-6">
                        <label class="form-label">Full Name</label>
                        <input type="text" class="form-control" placeholder="Auto-filled based on IC" id="patient_
                    </div>
                    <div class="col-md-6">
                        <label class="form-label">Time</label>
                        <input type="time" name="time" id="time" class="form-control">
                    </div>
                </div>

                <div class="mt-4">
                    <table class="table table-bordered text-center">
                        <thead class="table-light">
                            <tr>
                                <th>No.</th>
                                <th>Item</th>
                                <th>Description</th>
                                <th>Unit Price (RM)</th>
                                <th>QTY</th>
                                <th>Amount</th>
                                <th>Action</th>
                            </tr>
                        </thead>
                        <tbody id="productTable">
```

adminbill.php(8)

```
                        </thead>
                        <tbody id="productTable">
                            <tr>
                                <td>1</td>
                                <td><input type="text" class="form-control item-name" placeholder="Item name"></td>
                                <td><input type="text" class="form-control description" placeholder="Description"></td>
                                <td><input type="number" class="form-control unit-price" placeholder="Price"></td>
                                <td><input type="number" class="form-control qty" placeholder="Qty" value="1"></td>
                                <td><input type="text" class="form-control amount" placeholder="Amount" disabled></td>
                                <td><button type="button" class="btn btn-danger btn-sm remove-row"><i class="fas fa-trash-alt"></i></butt
                            </tr>
                        </tbody>
                    </table>
                    <button type="button" class="btn btn-primary btn-sm" id="addRow"><i class="fas fa-plus"></i> Add Product</button>
                </div>

                <div class="row mt-3">
                    <div class="col-md-4">
                        <label class="form-label">Subtotal (RM)</label>
                        <input type="text" class="form-control" id="subtotal" name="subtotal" readonly>
                    </div>
                    <div class="col-md-4">
                        <label class="form-label">SST (6%)</label>
                        <input type="text" class="form-control" id="tax" name="tax" readonly>
                    </div>
                    <div class="col-md-4">
                        <label class="form-label">Grand Total (RM)</label>
                        <input type="text" class="form-control" id="grandTotal" name="grandTotal" readonly>
                    </div>
                </div>

                <div class="d-flex justify-content-end mt-3">
                    <button type="submit" class="btn btn-success"><i class="fas fa-save"></i> Save</button>
                </div>
            </form>
```

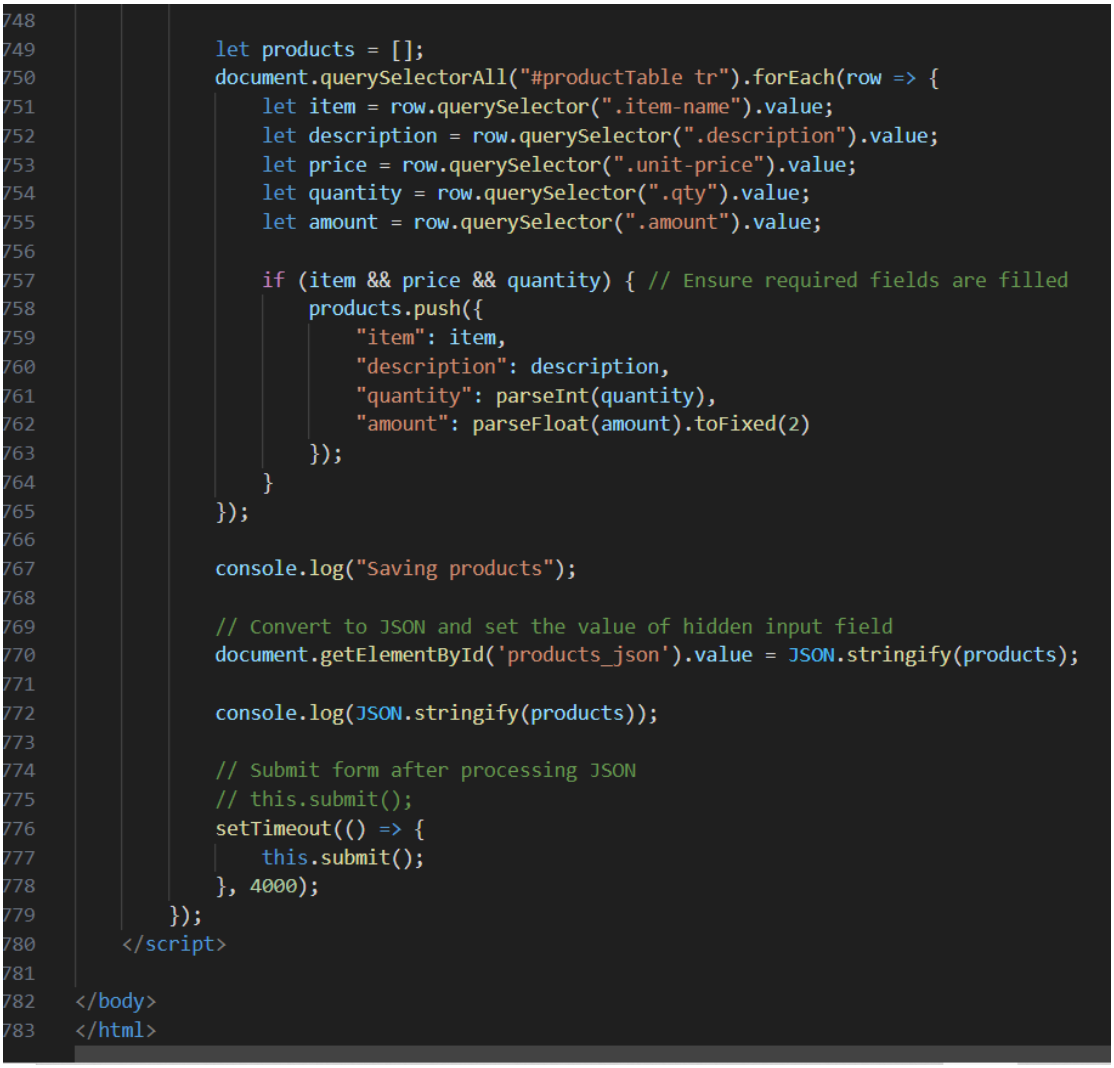
adminbill.php(9)

```
<script>
    $(document).ready(function() {
        $(".add").click(function(){
            $("#addBillModal").modal("show");
        });

        $('#patient_nric').on("input", function () {
            var nric = $(this).val();

            if(nric.length >= 6){
                $.ajax({
                    url: "bill_search_patient.php",
                    method: "GET",
                    data: {nric: nric},
                    success: function (response) {
                        $("#patient_name").val(response);
                    }
                });
            } else {
                $("#patient_name").val("");
            }
        });
    });

    document.addEventListener("DOMContentLoaded", function () {
        const productTable = document.getElementById("productTable");

        document.getElementById("addRow").addEventListener("click", function () {
            let rowCount = productTable.rows.length + 1;
            let newRow = `
                <tr>
                    <td>${rowCount}</td>
                    <td><input type="text" class="form-control item-name" placeholder="Item name"></td>
                    <td><input type="text" class="form-control description" placeholder="Description"></td>
                    <td><input type="number" class="form-control unit-price" placeholder="Price"></td>
                    <td><input type="number" class="form-control qty" placeholder="Qty" value="1"></td>
                    <td><input type="text" class="form-control amount" placeholder="Amount" disabled></td>
                    <td><button type="button" class="btn btn-danger btn-sm remove-row"><i class="fas fa-trash-alt"></i></button></td>
                </tr>`;
```

adminbill.php(10)

```
            productTable.insertAdjacentHTML("beforeend", newRow);
        });

        productTable.addEventListener("click", function (e) {
            if (e.target.closest(".remove-row")) {
                e.target.closest("tr").remove();
            }
        });

        productTable.addEventListener("input", function (e) {
            if (e.target.classList.contains("unit-price") || e.target.classList.contains("qty")) {
                let row = e.target.closest("tr");
                let unitPrice = row.querySelector(".unit-price").value || 0;
                let qty = row.querySelector(".qty").value || 0;
                row.querySelector(".amount").value = (unitPrice * qty).toFixed(2);
                updateTotal();
            }
        });
    });

    function updateTotal() {
        let subtotal = 0;
        document.querySelectorAll(".amount").forEach(amountField => {
            let amount = parseFloat(amountField.value) || 0;
            subtotal += amount;
        });

        let tax = subtotal * 0.06; // 5% tax
        let grandTotal = subtotal + tax;

        document.getElementById("subtotal").value = subtotal.toFixed(2);
        document.getElementById("tax").value = tax.toFixed(2);
        document.getElementById("grandTotal").value = grandTotal.toFixed(2);
    }
```

adminbill.php(11)



```
748
749             let products = [];
750             document.querySelectorAll("#productTable tr").forEach(row => {
751                 let item = row.querySelector(".item-name").value;
752                 let description = row.querySelector(".description").value;
753                 let price = row.querySelector(".unit-price").value;
754                 let quantity = row.querySelector(".qty").value;
755                 let amount = row.querySelector(".amount").value;
756
757                 if (item && price && quantity) { // Ensure required fields are filled
758                     products.push({
759                         "item": item,
760                         "description": description,
761                         "quantity": parseInt(quantity),
762                         "amount": parseFloat(amount).toFixed(2)
763                     });
764                 }
765             });
766
767             console.log("Saving products");
768
769             // Convert to JSON and set the value of hidden input field
770             document.getElementById('products_json').value = JSON.stringify(products);
771
772             console.log(JSON.stringify(products));
773
774             // Submit form after processing JSON
775             // this.submit();
776             setTimeout(() => {
777                 this.submit();
778             }, 4000);
779         });
780     </script>
781
782 </body>
783 </html>
```

adminbill.php(12)

Doctor Health Report

```php
<?php
    session_start();
    include("../include/connection.php");
    include("../include/header.php");
    $errors = array();

    $count = 1;
    $date = date('Y-m-d');

    $records_per_page = 9;
    $current_page = isset($_GET['page']) ? (int)$_GET['page'] : 1;
    $offset = ($current_page - 1) * $records_per_page;

    $stmt = $connect->prepare("SELECT COUNT(*) FROM health");
    $stmt->execute();
    $stmt->bind_result($total_records);
    $stmt->fetch();
    $stmt->close();

    $total_pages = ceil($total_records / $records_per_page);

    $stmt = $connect->prepare("SELECT * FROM health ORDER BY report_id DESC LIMIT ?, ?");
    $stmt->bind_param("ii", $offset, $records_per_page);
    $stmt->execute();
    $result = $stmt->get_result();

    $search = isset($_GET['search']) ? $_GET['search'] : "";
    $records_per_page = 9;
    $current_page = isset($_GET['page']) ? (int)$_GET['page'] : 1;
    $offset = ($current_page - 1) * $records_per_page;

    // Modify SQL to filter results based on search input
    $sql = "SELECT health.*, patients.firstname, patients.lastname, patients.ic
            FROM health
            JOIN patients ON health.patient_id = patients.id";

    if (!empty($search)) {
        $sql .= " WHERE patients.ic LIKE ?";
    }
```

health_report.php(1)

```php
    $sql .= " ORDER BY report_id DESC LIMIT ?, ?";

    $stmt = $connect->prepare($sql);

    // Bind parameters based on search input
    if (!empty($search)) {
        $searchParam = "%$search%";
        $stmt->bind_param("sii", $searchParam, $offset, $records_per_page);
    } else {
        $stmt->bind_param("ii", $offset, $records_per_page);
    }

    $stmt->execute();
    $result = $stmt->get_result();
?>




<div class="container-fluid">
    <div class="col-md-12">
        <div class="row">
            <div class="col-md-2" style="margin-left: -30px;">
                <?php include("sidenav.php"); ?>
            </div>
            <div class="col-md-10">
                <div class="card p-3">
                <div class="d-flex flex-wrap justify-content-between align-items-center mb-3">
                    <h4 class="mb-2 mb-md-0">Patient Health Report</h4>
                    <form method="GET" action="" class="d-flex flex-wrap gap-2">
                        <div class="input-group">
                            <span class="input-group-text bg-light border-end-0">
                                <i class="fa fa-search text-muted"></i>
                            </span>
                            <input type="text" class="form-control border-start-0 ps-0" name="search"
                                value="<?= htmlspecialchars($search) ?>" placeholder="Search by NRIC">
                        </div>
```

health_report.php(2)

```php
        <button type="submit" class="btn btn-primary">Search</button>
    </form>
    <button class="btn btn-success btn-sm" data-bs-toggle="modal" data-bs-target="#addReportModal">
        <i class="fas fa-plus me-1"></i> Add Report
    </button>
</div>

    <div class="table-container">
        <table class="table table-bordered table-striped table-hover text-center align-middle">
            <thead>
                <tr>
                    <th>#</th>
                    <th>Patient Name</th>
                    <th>NRIC</th>
                    <th>Date</th>
                    <th>Diagnosis</th>
                    <th>Prescription</th>
                    <th>Action</th>
                </tr>
            </thead>
            <tbody>
            <?php
                    if ($result) {
                        mysqli_data_seek($result, 0);
                        while ($row = mysqli_fetch_assoc($result)) {
                            $formattedDate = (new DateTime($row["date"]))->format('Y-m-d');
                            $patientStmt = $connect->prepare("SELECT * FROM patients WHERE id = ?");
                            $patientStmt->bind_param("i", $row["patient_id"]);
                            $patientStmt->execute();
                            $patientResult = $patientStmt->get_result();
                            $patient = mysqli_fetch_assoc($patientResult);
                            ?>
                            <tr>
                                <td><?php echo $count; ?></td>
                                <td><?php echo $patient["firstname"] . " " . $patient["lastname"]; ?></td>
```

health_report.php(3)

```
            <td><?php echo $patient["ic"]; ?></td>
            <td><?php echo $formattedDate; ?></td>
            <td><?php echo $row["diagnosis"]; ?></td>
            <td><?php echo $row["prescription"]; ?></td>
            <td class="d-flex flex-column gap-1">
                <!-- View PDF Button -->
                <a href="../patient/generate_health_pdf.php?id=<?= $row['report_id']; ?>"
                target="_blank"
                class="btn btn-info btn-sm w-100">
                    <i class="fas fa-file-pdf"></i> View PDF
                </a>

                <!-- Modify Button -->
                <button class="btn btn-primary btn-sm edit-btn w-100"
                        data-id="<?= $row['report_id']; ?>"
                        data-patientid="<?= $row['patient_id']; ?>"
                        data-date="<?= $row['date']; ?>"
                        data-diagnosis="<?= $row['diagnosis']; ?>"
                        data-prescription="<?= $row['prescription']; ?>"
                        data-bs-toggle="modal" data-bs-target="#editReportModal">
                    <i class="fas fa-edit"></i> Modify
                </button>

                <!-- Delete Button -->
                <a href="delete_health_report.php?id=<?= $row['report_id']; ?>"
                class="btn btn-danger btn-sm w-100"
                onclick="return confirm('Are you sure you want to delete this report?');">
                    <i class="fas fa-trash-alt"></i> Delete
                </a>
            </td>


        </tr>
        <?php
        $count++;
```

health_report.php(3)

```
<!-- Edit Health Report Modal -->
<div id="editReportModal" class="modal fade" tabindex="-1" aria-hidden="true">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header bg-warning text-white">
                <h5 class="modal-title">Edit Health Report</h5>
                <button type="button" class="btn-close" data-bs-dismiss="modal"></button>
            </div>
            <div class="modal-body">
                <form method="post" action="update_health_report.php">
                    <input type="hidden" id="edit_report_id" name="report_id">

                    <div class="mb-3">
                        <label for="edit_date" class="form-label">Date</label>
                        <input type="date" class="form-control" id="edit_date" name="date" required>
                    </div>

                    <div class="mb-3">
                        <label for="edit_diagnosis" class="form-label">Diagnosis</label>
                        <textarea class="form-control" id="edit_diagnosis" name="diagnosis" rows="3" required></textarea>
                    </div>

                    <div class="mb-3">
                        <label for="edit_prescription" class="form-label">Prescription</label>
                        <textarea class="form-control" id="edit_prescription" name="prescription" rows="3" required></textarea>
                    </div>

                    <div class="d-flex justify-content-end">
                        <button type="submit" class="btn btn-success">Save Changes</button>
                    </div>
                </form>
            </div>

        </div>
    </div>
</div>
```

health_report.php(4)

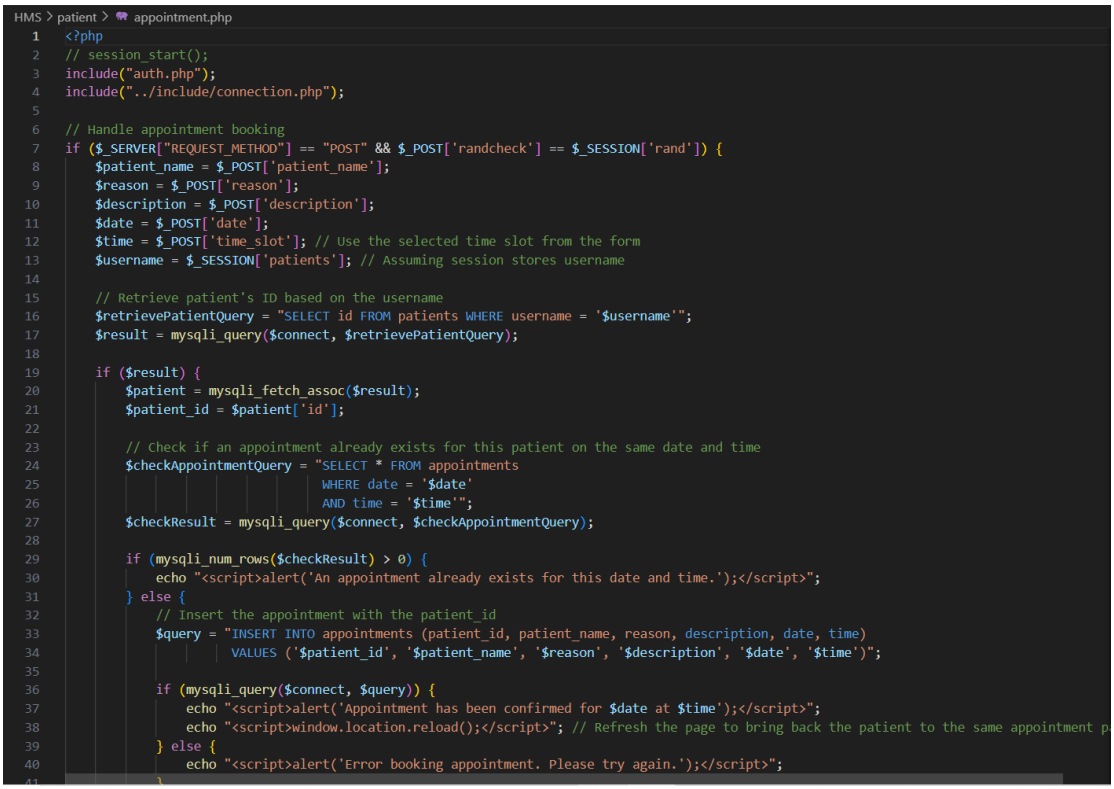
```
<!-- Add Health Report Modal -->
<div id="addReportModal" class="modal fade" tabindex="-1" aria-hidden="true">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header bg-info text-white">
                <h5 class="modal-title">Add Health Report</h5>
                <button type="button" class="btn-close" data-bs-dismiss="modal"></button>
            </div>
            <div class="modal-body">
                <form method="post" action="add_health_report.php">
                    <div class="mb-3">
                        <label for="patient_nric" class="form-label">Patient NRIC</label>
                        <input type="text" class="form-control" id="patient_nric" name="patient_nric" required>
                    </div>
                    <div class="mb-3">
                        <label for="diagnosis" class="form-label">Diagnosis</label>
                        <textarea class="form-control" id="diagnosis" name="diagnosis" required></textarea>
                    </div>
                    <div class="mb-3">
                        <label for="prescription" class="form-label">Prescription</label>
                        <textarea class="form-control" id="prescription" name="prescription" required></textarea>
                    </div>
                    <div class="d-flex justify-content-end">
                        <button type="submit" class="btn btn-success">Save Report</button>
                    </div>
                </form>
            </div>
        </div>
    </div>
</div>
```

health_report.php(5)

```
<script>
document.addEventListener("DOMContentLoaded", function () {
    const editButtons = document.querySelectorAll(".edit-btn");

    editButtons.forEach(button => {
        button.addEventListener("click", function () {
            document.getElementById("edit_report_id").value = this.dataset.id;
            document.getElementById("edit_date").value = this.dataset.date;
            document.getElementById("edit_diagnosis").value = this.dataset.diagnosis;
            document.getElementById("edit_prescription").value = this.dataset.prescription;
        });
    });
});
</script>
```

health_report.php(6)

Patient Appointment Flow Description



```
HMS > patient > appointment.php
<?php
// session_start();
include("auth.php");
include("../include/connection.php");

// Handle appointment booking
if ($_SERVER["REQUEST_METHOD"] == "POST" && $_POST['randcheck'] == $_SESSION['rand']) {
    $patient_name = $_POST['patient_name'];
    $reason = $_POST['reason'];
    $description = $_POST['description'];
    $date = $_POST['date'];
    $time = $_POST['time_slot']; // Use the selected time slot from the form
    $username = $_SESSION['patients']; // Assuming session stores username

    // Retrieve patient's ID based on the username
    $retrievePatientQuery = "SELECT id FROM patients WHERE username = '$username'";
    $result = mysqli_query($connect, $retrievePatientQuery);

    if ($result) {
        $patient = mysqli_fetch_assoc($result);
        $patient_id = $patient['id'];

        // Check if an appointment already exists for this patient on the same date and time
        $checkAppointmentQuery = "SELECT * FROM appointments
                                  WHERE date = '$date'
                                  AND time = '$time'";
        $checkResult = mysqli_query($connect, $checkAppointmentQuery);

        if (mysqli_num_rows($checkResult) > 0) {
            echo "<script>alert('An appointment already exists for this date and time.');</script>";
        } else {
            // Insert the appointment with the patient_id
            $query = "INSERT INTO appointments (patient_id, patient_name, reason, description, date, time)
                      VALUES ('$patient_id', '$patient_name', '$reason', '$description', '$date', '$time')";

            if (mysqli_query($connect, $query)) {
                echo "<script>alert('Appointment has been confirmed for $date at $time');</script>";
                echo "<script>window.location.reload();</script>"; // Refresh the page to bring back the patient to the same appointment p
            } else {
                echo "<script>alert('Error booking appointment. Please try again.');</script>";
```

appointment.php(1)

```
                } else {
                    echo "<script>alert('Error booking appointment. Please try again.');</script>";
                }
            }
        } else {
            echo "<script>alert('Error retrieving patient information.');</script>";
        }
    }
    ?>

    <!DOCTYPE html>
    <html lang="en">
    <head>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title>Book Appointment</title>
        <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
        <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/fullcalendar/3.10.2/fullcalendar.min.css" />
        <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/jquery-timepicker/1.13.1/jquery.timepicker.min.css">
        <style>
            .dashboard-card {
                height: 150px;
                border-radius: 8px;
                box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
                transition: transform 0.2s ease;
                color: #fff;
            }
            .dashboard-card:hover {
                transform: translateY(-5px);
            }
            .calendar-card {
                background-color: #52b4d6; /* Green */
            }
            .modal-header {
                background-color: #17a2b8; /* Teal */
                color: white;
            }
```

appointment.php(2)

```
        </style>
    </head>
    <body>
        <?php include("../include/header.php"); ?>

        <div class="container-fluid">
            <div class="row">
                <div class="col-md-2" style="margin-left: -30px;">
                    <?php include("sidenav.php"); ?>
                </div>
                <div class="col-md-10 d-flex justify-content-center">
                    <div class="col-md-7">
                        <h2 class="text-center" style="margin-top: 20px; margin-bottom: 20px;">Book an Appointment</h2>
                        <div class="d-flex justify-content-center">
                            <div id="calendar" class="calendar-card p-3" style="min-width: 500px; width: 100%;"></div>
                        </div>
                    </div>
                </div>
            </div>
        </div>

        <!-- Appointment Modal -->
        <div class="modal fade" id="appointmentModal" tabindex="-1" role="dialog" aria-labelledby="appointmentModalLabel" aria-hidden="true">
            <div class="modal-dialog" role="document">
                <form id="appointmentForm" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>" method="POST">
                    <?php
                        $rand = rand();
                        $_SESSION['rand'] = $rand;
                    ?>
                    <input type="hidden" value="<?php echo $rand; ?>" name="randcheck" />
                    <div class="modal-content">
                        <div class="modal-header">
                            <h5 class="modal-title" id="appointmentModalLabel">Appointment Details</h5>
                            <button type="button" class="close" data-bs-dismiss="modal" aria-label="Close">
                                <span aria-hidden="true">&times;</span>
                            </button>
                        </div>
                        <div class="modal-body">
                            <input type="hidden" name="date" id="appointmentDate">
```

appointment.php(3)

```
                    <div class="modal-body">
                        <input type="hidden" name="date" id="appointmentDate">
                        <div class="form-group">
                            <label for="timeSlot">Select Time Slot</label>
                            <select class="form-control" name="time_slot" id="timeSlot" required>
                                <!-- Time slots will be populated dynamically -->
                            </select>
                        </div>
                        <div class="form-group">
                            <label for="patientName">Patient Name</label>
                            <input type="text" class="form-control" name="patient_name" id="patientName" required>
                        </div>
                        <div class="form-group">
                            <label for="reason">Reason</label>
                            <input type="text" class="form-control" name="reason" id="reason" required>
                        </div>
                        <div class="form-group">
                            <label for="description">Description</label>
                            <textarea class="form-control" name="description" id="description" rows="3"></textarea>
                        </div>
                    </div>
                    <div class="modal-footer">
                        <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Cancel</button>
                        <button type="submit" class="btn btn-primary">Confirm Appointment</button>
                    </div>
                </div>
            </form>
        </div>
    </div>

    <script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/moment.js/2.29.1/moment.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/fullcalendar/3.10.2/fullcalendar.min.js"></script>
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
    <script>
```

appointment.php(4)

```
    <script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/moment.js/2.29.1/moment.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/fullcalendar/3.10.2/fullcalendar.min.js"></script>
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
    <script>
        $(document).ready(function() {
            $('#calendar').fullCalendar({
                selectable: true,
                selectHelper: true,
                minDate: moment().format('YYYY-MM-DD'), // Disable past dates
                select: function(start, end) {
                    var selectedDate = moment(start).format('YYYY-MM-DD');
                    var today = moment().format('YYYY-MM-DD');

                    if (selectedDate >= today) {
                        $('#appointmentDate').val(selectedDate);

                        // Fetch available time slots
                        generateTimeSlots(selectedDate);

                        $('#appointmentModal').modal('show');
                    } else {
                        alert("You cannot select a past date for an appointment.");

                    }
                },

                dayRender: function(date, cell) {
                    var today = moment().startOf('day');

                    if (date.isBefore(today)) {
                        cell.css('background-color', '#6e97cc');
                    } else if (date.isAfter(today)) {
                        cell.css('background-color', '#ffffff');
                    }
                }
            });
```

appointment.php(5)

```
function generateTimeSlots(selectedDate) {
    var timeSlotSelect = $('#timeSlot');
    timeSlotSelect.empty(); // Clear existing options

    var startTime = moment('09:00', 'HH:mm');
    var endTime = moment('18:00', 'HH:mm');

    // Fetch booked time slots for the selected date
    $.ajax({
        url: 'getBookedSlots.php',
        type: 'POST',
        data: { date: selectedDate },
        success: function(response) {
            var bookedSlots = JSON.parse(response);
            console.log("Booked Slots:", bookedSlots); // Debugging line

            // Convert booked slots to "HH:mm" format for comparison
            var unavailableSlots = new Set(bookedSlots.map(slot => moment(slot, 'HH:mm:ss').format('HH:mm')));

            while (startTime <= endTime) {
                var timeFormatted = startTime.format('HH:mm');
                var option = $('<option>', {
                    value: timeFormatted,
                    text: timeFormatted
                });

                if (unavailableSlots.has(timeFormatted)) {
                    option.prop('disabled', true);
                    // Custom class or inline styles for styling
                    option.addClass('unavailable-slot'); // Use CSS for styling
                }

                timeSlotSelect.append(option);
                startTime.add(30, 'minutes'); // Increment by 30 minutes
            }
        }
    });
```

appointment.php(6)

POSTER


KIWI CLINIC
WEB-BASED CLINIC MANAGEMENT SYSTEM
Real-time booking
This system offer real-time booking for the patient so that they no need go to the clinic only to make appointment. Patient can make their appointment anytime, anywhere. Reminder will be send before patient appointment day.
Online Billing
This system offer online billing feature with few type of payment method so that customer can select their preferred payment method to pay. Recipt will be generated automatically after payment is completed.
BMI Measure and Track
An additional tools offered by this system to allow user to measure and track their BMI condition. Makeing the web-based system multi-functional, providing the BMI result with the auto-generated advice and comment to improve user body health condition
PAY
My BMI
healthy!!!
About Us
THE ONLINE CLINIC MANAGEMENT SYSTEM IS A WEB-BASED SYSTEM DESIGNED TO ALLOW PATIENT TO MAKE APPOINTMENT ANYTIME AND ANYWHERE AND PAY THEIR BILL ONLINE WITHOUT KEEP WAITING FOR THE OTHERS QUEUE. THSES SYSTEM ALSO OFFER ADDITIONAL TOOLS SUCH AS BMI MEASUREMENT FOR THE USER TO KEEP TRACK THEIR BMI CONDITION. OTHER THAN THAT, MANY DIFFERENT FUNCTIONS ARE OFFER TO PROVIDE A BETTER SERVICE AND EXPERIENCE FOR USER.
UTAR
UNIVERSITI TUNKU ABDUL RAHMAN
Prepared by: Lim Wei Qi (21ACB01742)