Diabetes Monitoring System with Ingredient Scanning For Informative Food Filtering

By

Ong Jia Hui

DEPARTMENT OF COMPUTING AND INFORMATION TECHNOLOGY

TUNKU ABDUL RAHMAN UNIVERSITY OF MANAGEMENT AND TECHNOLOGY
PULAU PINANG

ACADEMIC YEAR
2023/2024

Diabetes Monitoring System with Ingredient Scanning For Informative Food Filtering

By

Ong Jia Hui

Supervisor: Ts. Chau Guan Hin

A project report submitted to the

Department of Computing and Information Technology

in partial fulfillment of the requirement for the

Bachelor of Software Engineering (Honours)

Department of Computing and Information Technology

Tunku Abdul Rahman University Of

Management and Technology

Pulau Pinang

2023/2024

Declaration

The project submitted herewith is a result of my own efforts in totality and in every aspect of the project works. All information that has been obtained from other sources has been fully acknowledged. I understand that any plagiarism, cheating or collusion or any sorts constitutes a breach of TAR University of Management and Technology rules and regulations and would be subjected to disciplinary actions.

_____JH_____________

Ong Jia Hui

Bachelor of Software Engineering (Honours)

ID: 22PMR05969

Abstract

In recent times, the escalating prevalence of non-communicable diseases, notably diabetes, has raised public concern due to the increasing death rates and prevalence. The consequences of diabetes, compounded by its potential for severe health complications and mortality, poses a formidable challenge for affected individuals. A significant issue in managing diabetes effectively is the lack of accessible information and guidance for patients, leading them to rely primarily on consultations with healthcare professionals. Despite the existence of some diabetes management systems for self-monitoring, their limitations, particularly in providing comprehensive dietary information, contribute to individuals continuing consuming inappropriate packaged foods, exposing them to adverse health outcomes. Therefore, this project proposed a system that consists of both web application and mobile application, designed to cater both the patient and healthcare providers, ensuring them to work on the same system and enhance the data synchronisation. This aims to establish a platform that facilitates seamless data synchronisation between the diabetes individual to the healthcare providers. The mobile application incorporates an innovative ingredient scanning feature, empowering individuals to make informed decisions about food products. By implementing the Optical Character Recognition(OCR) technology, this feature extracts text from images of food product labels, including ingredient items and nutrition facts. The extracted data is then input into a machine learning model, specifically a K-Nearest Neighbors (K-NN) algorithm trained on a prepared dataset using open-source libraries such as TensorFlow. The machine learning model performs classification activities, providing users with information regarding the suitability of the consumed product. The overall performance of the system achieved an accuracy of %. To ensure the practicality and compatibility of the system with the routines of individuals managing diabetes, an iterative process for the Agile model of Scrum framework will be utilised and several sessions of interviews and questionnaire surveys have been conducted. In conclusion, the implementation of this system streamline the management and monitoring processes , fostering a greater healthcare environment.

Acknowledgement

I would like to express my deepest gratitude to Tunku Abdul Rahman University College for providing me with the invaluable opportunity to undertake my final year project during the last year of my Bachelor of Software Engineering studies. This experience has been a pivotal moment in my academic journey, allowing me to apply the knowledge gained over the years in a practical setting. Additionally, I would like to express my utmost gratitude and respect to my project supervisor, Ts. Chau Guan Hin, for his dedication and guidance throughout the project. His unwavering support and insightful feedback have been crucial elements in refining and improving the quality of my work. I am sincerely grateful for the mentorship that contributes significantly to my overall academic and professional growth. Finally, I want to express my appreciation to my family for their encouragement, understanding, and patience throughout this period. Their support has been a source of strength and motivation, making this accomplishment possible.

Table of Content

Chapter 1

Introduction

1.1 Project Background

In this era of globalisation, health-related issues have been getting more and more attention from the global community. There is a growing recognition of health issues and the need for collaborative efforts to address them. One of the significant health-related issues gaining attention is the non-communicable diseases (NCDs) such as cardiovascular diseases, cancer, diabetes and respiratory disease which have become global concerns. Noncommunicable diseases (NCDs), including heart disease, stroke, cancer, diabetes and chronic lung disease, are collectively responsible for 74% of all deaths worldwide (World Health Organization, 2022). Diabetes is one of the prominent NCDs, and its prevalence is showing a rapidly increasing trend. According to the IDF Diabetes Atlas 10th edition 2021, around 1 in every 10 adults are living with diabetes worldwide. This number is projected to escalate to 643 million by 2030 and 783 million by 2045 (IDF Diabetes Atlas | Tenth Edition, 2021). However, many individuals with diabetes lack proper knowledge and guidance, unable to perform self-management efficiently. They often rely solely on the advice given by their healthcare providers. Moreover, individuals with diabetes may struggle to achieve optimal diet control due to a lack of clear knowledge regarding food ingredients. Furthermore, the lack of a standardised management approach complicates their efforts to maintain consistent and accurate information. Consequently, healthcare providers may face challenges in analysing the glucose level trends and understanding their patients' actual health conditions.

To address this issue and empower individuals with diabetes to take an active role in managing themselves, a diabetes monitoring system is proposed. With systematic guidance and information provided to the individuals, they can always make informed decisions about their health. Individuals are willing to get guidance or education to maintain their health (Yang Lee et al, 2020). Thus, developing a diabetes monitoring system can enhance the awareness and effectiveness of the individual in managing themselves. The diabetes monitoring system allows individuals to record and keep track of their daily blood glucose levels. By visualising the data recorded over time, the individuals can identify the pattern and trend in their blood glucose levels. Sufficient filtering and validations are included within the system to ensure the users, particularly diabetes, will be able to perform accurate data recording even without professional knowledge. Notifications and alerts within the system are responsible for drawing attention to abnormalities in blood glucose levels or reminding individuals to take their medications on time. These reminders can contribute to better adherence to prescribed treatments and prevent potential complications associated with poorly controlled diabetes.

In addition, the system will implement an ingredient scanning feature with image processing technology. This feature can provide individuals with relevant information about food products, allowing them to make informed choices about their diet. By capturing the ingredient list from the packaging of food products, the system can process the information and generate output that determines the suitability for consumption of individuals with diabetes. This ensures that individuals understand the impact of specific ingredients on their health before their consumption. When the scanned food item is unsuitable for consumption, the system will highlight the possible ingredients that contribute to this output. This aims to help the individuals to understand and be exposed to the correct information. Providing individuals with information about their food consumption reduces the reliance solely on the healthcare provider and empowers them to make proper assessments of their consumption to maintain their health as diabetes patients (Makhsous et al., 2020).

The proposed diabetes monitoring system consists of a web system for medical healthcare providers and a mobile application for individuals with diabetes. The web system enables healthcare providers to observe their patients' conditions and update medical records. The mobile application focuses on aiding individuals in monitoring and managing their diabetes. It allows users to record daily updates such as blood glucose levels, and visualise the data for a better view. The app also provides notification features for reminders and data sharing with healthcare providers. Additionally, the mobile application includes an ingredient scanning feature to help individuals with diabetes make informed dietary choices. The patients can make appointments with their doctor via the mobile platform, ensuring a streamlined process.

In this project, the agile software development model used is the Agile methodology. In this methodology, there will be six phases for each iteration, which are planning, design, development, testing, deployment and feedback. The reason for selecting an agile methodology is that it can manage requirement changes by providing proper change management methodologies to achieve successful product delivery. The system delivered is consistently evaluated against the expected outcomes to ensure the development is on the right path (Onix, 2022). Throughout every sprint, the system developed will be tested, minimising potential risks that may develop system failures.

1.2 Problem Statement

Diabetes mellitus, commonly known as diabetes, is a metabolic disorder characterised by high blood glucose levels resulting from either insufficient insulin production or ineffective use of insulin. It is a chronic health condition that affects the body's ability to transform food into energy, leading to excessive glucose in the bloodstream. According to the International Diabetes Federation (IDF) Atlas 10th edition 2023, there are approximately 537 million adults living with diabetes worldwide, meaning that roughly one in every ten adults has diabetes. The majority of individuals with diabetes reside in low- and middle-income countries, with around three out of four diabetes cases originating from these regions. The prevalence of diabetes has been predicted by the IDF to continue to rise, reaching approximately 643 million by 2030 and 783 million by 2045 (IDF Diabetes Atlas | Tenth Edition, 2021). In Malaysia, the Ministry of Health(MOH) conducted a National Health and Morbidity Survey (NHMS) in 2019, revealing a diabetes prevalence of 18.3%, which is equivalent to 1 in every 5 adults living with diabetes, considering a total of 3.9 million Malaysians that are aged 18 and above having diabetes (NIH, 2020). The survey also indicated an increase in prevalence compared to the data in 2015, reporting a rate of 13.4% (IPH et al., 2019). Neglecting proper monitoring and management of diabetes can lead to severe complications, including organ damage and tissue failure, eventually resulting in death. According to the 10 principal death causes in Malaysia, diabetes is ranked as the sixth leading cause of death, accounting for a rate of 2.0% for all recorded deaths (Department of Statistics, 2021). These statistics highlight the urgent need for immediate attention and comprehensive monitoring of diabetes as a significant public health issue.

Currently, there is no standardised approach among individuals with diabetes to aid them in recording their daily measurements. This raises concerns about preserving the consistency in measurement frequency and timing. Many individuals collect their measurements based on their personal preferences, which can result in inconsistent or unclear information. They may tend to perform the monitoring and measurement based on their convenience, leading to an irregular measuring routine. This can pose challenges for later analysis and treatment evaluation that rely on the collected data. A recent study found that while around 95% of diabetics measured their blood glucose levels, only 72% did so daily. Among those who monitored their glucose levels, 76% recorded the measurements. Of these, 55.6% preferred paper-based methods, while 44.4% opted for digital devices (Jamal et al., 2021). These statistics indicate that individuals with diabetes record their data according to their preferences, and some do not adhere to daily measurements. The lack of consistency in data recording can create difficulties for healthcare providers in tracking and understanding their patients' health conditions. The data may also be insufficient to identify meaningful trends or may not be considered valid due to sporadic recording. Moreover, individuals with diabetes may face challenges in monitoring or measuring their condition effectively when they receive inconsistent or insufficient information from healthcare professionals. This can hinder their ability to self-manage their diabetes properly (Maneze et al., 2019). For example, they may lack awareness of the correct frequency or timing for measurements, leading to inaccurate readings. Inadequate information further complicates the task of healthcare providers in accurately understanding and analysing the individual's current condition.

Many individuals with diabetes often struggle and sometimes get trapped with purchasing unsuitable food due to a lack of professional knowledge in healthcare and food science. Understanding ingredient lists on product packaging together and identifying the possible impact on health can be challenging for them. Additionally, subjective assessments of food value can lead to biassed decision-making (Aschemann-Witzel et al., 2019). This puts them at risk of consuming unsuitable food without being aware of its potential negative effects. For individuals with diabetes, maintaining strict control over their diet is crucial, as their dietary patterns play a significant role in managing diabetes. Adhering to a diabetes-friendly diet is essential to avoid frequent spikes in blood glucose levels (CDC, 2022). The Mediterranean diet pattern, which is mostly plant-based, has been shown to reduce the risk of severe diabetes complications by approximately 23% (Papamichou et al., 2019). However, achieving an optimal diet can be challenging as individuals need to have a clear understanding of food ingredients before purchasing or consuming them. This requires a certain level of knowledge to analyse the items listed in ingredient lists. Ordinary individuals with only general knowledge about food and ingredients may find it difficult to determine whether a particular food is suitable for them.

The main competition for this diabetes monitoring system comes from the market's existing diabetes management system. Some of the example applications are Glucose Tracker - Diabetic Diary, which was developed by Mel Studio (Mel Studio, 2023) and mySugr application (mySugr, 2023). The application Glucose Tracker - Diabetic Diary, is designed to assist individuals with diabetes in effectively managing their condition. It allows its user to record their regular haemoglobin levels, sugar levels and blood pressure. The user can then keep track of their previous records which are presented by the system in charts for better data analysis. It also supports the user to export the records for external use such as printing or sharing the recorded information with their healthcare providers. This application also offers features such as reminders, and goal setting to enhance the experience of the users. In comparison, mySugr application is another popular diabetes management tool. It aims to simplify the management of diabetes by offering features such as glucose monitoring, meal tracking, insulin dosa logging and data analysis. One of its outstanding features is that it integrates with blood glucose metres and can automatically transfer the individuals' daily blood glucose measurement via Bluetooth. It analyses and identifies the pattern of blood glucose, generating clear reports that can be shared with doctors. To aid the individuals in understanding their management progress, it provides visualisations such as graphs and trends when presenting the data recorded by the individuals. Nevertheless, this application is implemented with a feature which can predict future HbA1c rates based on past information to show the ongoing trend.

The current market offers several systems designed to assist individuals with diabetes in monitoring and managing their health condition. However, these systems possess similar functionalities, lacking the ability to provide comprehensive information, guidance or suggestions and are unable to synchronise the recorded data to the healthcare providers. Furthermore, these applications tend to focus primarily on diabetes individuals, neglecting the involvement of healthcare providers. Consequently, the existing systems in the market play a limited role in effectively addressing the healthcare challenges faced by the global population.

1.3 Project Objective(s)

The objectives of this project are clearly stated below:

To provide a structured platform to assist diabetes individuals in systematic monitoring

By introducing the diabetes monitoring system to diabetes individuals, the diabetes individuals can utilise the system to record their daily necessary measurements. The system can organise the data stored, keeping the records for future traceback. Furthermore, the system offers better insight into their blood glucose level trend, understanding how their levels fluctuate over time or how their lifestyle impacts their health. Understanding the trends helps individuals to identify potential risks or abnormalities proactively, minimising the consequences of diabetes comorbidities. Not only that, the system offers customisable alerts and reminders which help individuals adhere to their diabetes management routine. The reminder module will be able to keep the individuals engaged and reminded of their important tasks in time intervals, reducing the likelihood of irregular measurements or missed medication.

This objective can be achieved by receiving positive feedback from at least 70% of users, indicating satisfaction with the platform’s structure and maintaining a sustained usage rate of 75% across all registered diabetes individuals after its implementation for one year.

To provide adequate ingredient information for informative decisions

An ingredient scanning feature is implemented in this project, allowing diabetes individuals to make informed decisions about their dietary choices. This feature can provide further explanation of the ingredients listed on the packaging, aiding individuals in determining the suitability for their consumption. The individual will have to capture the ingredient list shown on the packaging. The system will then process the input and generate outputs, which include the suitability level of the manufactured food for consumption and diabetes-related nutrition facts. Certain ingredients with excessively high nutritional facts will be highlighted when the captured food is found unsuitable or may lead to a negative impact on the blood glucose level. The unsuitable or unrecommended ingredient will be highlighted to the user to increase awareness. By incorporating this feature, the system helps individuals to identify healthier and more suitable consumptions, control their intake and manage their blood glucose levels better. It increases the ability of the individuals to obtain a better understanding of the nutritional content and empowers their knowledge to take active roles in managing themselves.

This objective can be achieved by achieving a minimum accuracy rate of 85% in the ingredient information provided, as verified through third-party audits or consumer feedback and by providing users with relevant ingredient information within a maximum of 5 seconds.

To streamline the process of remote monitoring and enhance data sharing between diabetes individuals and healthcare providers

The diabetes monitoring system ensures a comprehensive monitoring process by integrating the system with the healthcare provider. This integration can establish seamless synchronisation and data-sharing capabilities between diabetes individuals and healthcare providers. By allowing healthcare providers to access their patient's blood glucose data remotely, they can provide timely feedback or suggestions to that particular patient to ensure better treatment effects. As the information traced through the system is real-time based and always up-to-date, the healthcare provider can discover and detect any health abnormalities happening, and carry out treatment adjustments if necessary. As data is recorded systematically in the system, healthcare providers can acquire sufficient information about the patients. This information aids healthcare providers in making better medication strategies and lifestyle recommendations. Nevertheless, healthcare providers can keep track of the effectiveness of the treatment plans by identifying the trends or patterns of the patient's blood glucose levels, especially to ensure that treatment is suitable and effective for the individual. Remote monitoring of the patient's health trends enables healthcare providers to perform timely interventions when a certain irregular trend is detected, optimising diabetes management.

This objective can be achieved in efficiency rating by the users of at least 90% via the feedback form.

To enhance the reliability of data collected by the individuals

The diabetes monitoring system provides a standard interface with necessary guidance for the individual to record their measurement data. By offering clear instructions and predefined fields for data entry, the system reduces the possibility of incomplete or ambiguous data, improving the reliability of the data collected. Fragmented or incomplete records can be avoided as the system can perform necessary validation on the user's input. Promoting standardisation of the data guarantees the validity and completeness of the collected data, which is crucial for future analysis or tracking. By establishing a consistent format for data recording, the system guarantees that the collected data adheres to a certain level of reliability. Quality data enables the system to generate reliable and meaningful information based on the recorded data. For instance, the system can generate an individual's blood glucose trend graph or chart to aid their understanding of their current health status. This helps the individuals to know their health condition more efficiently rather than going through the past recorded data one by one, making comparisons of records over time. For the healthcare provider, the system can generate comprehensive reports about a particular patient’s health conditions over a specific timeline. These reports are able to convey to the healthcare provider a consolidated view of the individuals, allowing them to quickly assess the patient and make informed decisions regarding treatment strategies.

This objective can be measured through the retrieval of data by the individuals or healthcare providers available 99% of the time and the ability of the system to detect and provide relevant error messages for 95% of input errors.

To provide convenience to diabetes individuals in getting diabetes care

By ensuring data synchronisation between diabetes individuals and their healthcare providers, the system enables remote monitoring of their health. This remote monitoring could bring significant benefits, particularly to those who have difficulties in attending physical follow-ups, such as individuals with limited mobility, those living in remote areas, or those with busy schedules. The system can facilitate communication between them and their healthcare providers by securely sharing their daily recorded data for monitoring and analysis. With access to real-time data, the healthcare provider can carry out timely interventions, and provide necessary suggestions based on their current health. Even without frequent physical consultations, the system can assist the healthcare provider in obtaining regular updates from the individuals. In this way, the individuals remain connected to their healthcare providers to receive ongoing support.

The achievement of this objective can be measured by achieving a minimum of 70% individual satisfaction rate with the features provided and ensuring the users can carry out requested functions regardless of their locations or time with a minimum availability rate of 95%.

1.4 Advantages & Contributions

The proposed system is designed to cater for the needs of diabetes individuals for a reliable and convenient system to manage themselves through a friendly interface. One of the key features of the system is its ability to store real-time blood glucose data, allowing individuals to easily access and review their past measurements. By offering comprehensive records of their blood glucose levels, the system empowers individuals to gain valuable insights into their health trends and detect any possible abnormalities at an earlier stage. Additionally, the system allows users to customise reminders or notifications based on their specific needs. This feature helps individuals in receiving timely alerts for regular monitoring routines, enhancing their diabetes management.

The proposed system can facilitate a seamless connection between diabetes individuals and their healthcare providers, addressing the limitations of existing applications that primarily focus on individuals with diabetes alone. By considering the involvement of the diabetes individuals and their healthcare providers in the same system, the proposed system enables automatic data synchronisation and enhances communication between both parties. The proposed diabetes monitoring system ensures that data recorded by diabetes individuals, such as blood glucose levels, can be automatically synchronised with the healthcare provider. This eliminates the need for manually exporting data from certain applications to share with their healthcare providers, streamline the process and ensure the healthcare providers have access to up-to-date information. Healthcare providers can then access the data in real-time to monitor the individual's health trend and status, review the data, analyse the trends, and provide personalised advice or adjustments to the individual's diabetes management plan. This will help in ensuring effective remote monitoring, particularly valuable when frequent physical monitoring is not feasible or realistic.

Currently, many diabetes management systems available in the market focus primarily on data recording without offering additional guidance or suggestions on diabetes management. However, the proposed diabetes monitoring system provides better services by incorporating an ingredient scanning feature that provides valuable dietary guidance. The ingredient scanning feature can analyse the captured ingredient list and offer recommendations on the consumption suitability for diabetes individuals. By leveraging this feature, individuals can make healthier and more suitable food choices. This empowers them to take full control of their diet and avoid developing any potential risk associated with consuming foods that may adversely affect their conditions. The ingredient scanning feature of the proposed system enhances the overall effectiveness of diabetes management in terms of diet. By providing targeted information, the system helps diabetes individuals maintain healthier eating habits, manage their blood glucose levels, and prevent potential complications.

From another perspective, the proposed diabetes monitoring system offers greater convenience and economic value compared to other applications available in the market. Unlike existing systems that require users to download separate applications for each feature, such as ingredient scanning, glucose level monitoring, and data synchronisation, the proposed system combines all these features into a single comprehensive platform. By registering themselves as a user of this single system, they are given access to all the features within the diabetes monitoring system, providing them with comprehensive diabetes monitoring and management capabilities. The integrated approach ensures all individuals with diabetes can freely access and utilise the platform without incurring additional expenses. This not only offers economic benefits to individuals but also encourages the participation of a wider range of the population, making diabetes management more accessible to all.

Furthermore, the proposed system also provides advantages to healthcare providers. Integrating a system that can perform real-time data synchronisation promotes the effectiveness of healthcare providers to monitor and manage the health of their patients remotely. This eliminates the need for manual data entry or physical consultations, significantly reducing the costs associated with data synchronisation and real-time updates.

The system has the potential to contribute to society by achieving one of the goals of the United Nations Sustainable Development Goals (SDGs) 2030 (United Nations, 2023), specifically the third goal which aims to ensure healthy lives and promote well-being for all at all ages. Within this goal, sub-target 3.4 will be focused on, aiming to reduce premature mortality from non-communicable diseases. The proposed system will primarily focus on diabetes, which is one of the non-communicable diseases that has shown a rapid prevalence rate in recent years. By providing the proposed system, the diabetes society can utilise the features provided by the system to manage themselves more efficiently and receive better monitoring, promoting healthy eating habits and lifestyles via systematic guidance. Additionally, the system allows the accessibility of diabetes individuals at a low cost. According to the International Diabetes Federation, the majority of the diabetes population, approximately 79.4%, come from low- and middle-income backgrounds (International Diabetes Federation, 2019). Introducing this system can get rid of the economic concern that stops certain individuals from receiving proper diabetes care due to being unable to afford the expenses. This promotes equitable access to healthcare, reducing the development of comorbidities resulting from a lack of timely guidance and support.

The proposed system can play a great role in contributing to the healthcare industry by reducing the healthcare providers’ workload. Due to the increasing number of diabetes individuals, healthcare providers who offer diabetes care consultation or follow-up are facing high working rates (Khan et al., 2020). From the perspective of healthcare providers, the system can offer significant convenience by allowing remote monitoring of patients. Remote monitoring eliminates the need for frequent physical monitoring, which can be unnecessary or unfeasible in certain conditions. For instance, healthcare providers can monitor their patients' current health index, and identify and detect any abnormal happening, ensuring timely interventions when necessary. Consequently, healthcare centres can accommodate more patients under their supervision, increasing the chances of other individuals getting proper care. Routine monitoring can be replaced by the system, allowing healthcare providers to allocate more attention to critical situations that require extra care and interventions. With the system's ability to synchronise data between the diabetes individuals and the healthcare provider, healthcare providers can streamline their operations and treatment plans, focusing on critical cases while relying on the system to perform routine monitoring and updates.

1.5 System Development Methodology & Project Plan

1.5.1 Software Development Model

Figure 1.1: Agile Scrum Methodology

The software development methodology selected to apply for this project is the agile software development model with the scrum framework. Scrum is a lightweight framework that allows work to be done within a small piece of time, with continuous experimentation and feedback loops along the way to learn and improve (Scrum Org., 2023). By applying Scrum methodology, the most important features will be focused in each sprint to come out with a potentially deliverable product while additional features will be added to the project subsequently in the following sprint and adjusted based on the requirements of the stakeholders (Business News Daily, 2023). This allows the work to be organised efficiently, prioritising the task that has a heavier weight. Scrum methodology ensures the project is implemented with high adaptability. It will enable requirement adaptation based on the feedback from the customers and the end user. At the end of each sprint, it provides plenty of room to respond to changes, redefine and replan the activities. Thus, the developers can identify the possible deviations or gaps to improve the system. This ensures the result of the project's development is highly compatible with the expectations of the users and complete with all requirements. Besides that, Scrum promotes frequent communication between the developers and the stakeholders. The stakeholders are involved in the evaluation at the end of each sprint to determine whether the developed output has reached their expectations. By involving the participation of the stakeholders, the development of the products can always be on the right track.

1.5.2 Milestones

Table 1.1: Schedule of Project Development Lifecycle

Sprint 1	**Duration**: 14/7/2023 - 23/8/2023	
Milestone	**Milestone Goal**	**Deadline**
Planning	Planning for the implementation of core functionalities of the system.	14/7/2023
Design	Design the core functionalities details, including the interface, data, report, security, process, software architecture, algorithm, etc. Design the database and the interaction with the core functionalities. Design the communication between the functionalities of the system.	28/7/2023
Development	Develop the core functionalities based on the design done	11/8/2023
Testing	Test the developed core functionalities based on the requirements and compare them to the expectations	18/8/2023

Deployment	Integrate all core functionalities and check the interaction between the functionalities and the database within the system. Define the deployment environment, and processes/ steps for the developed system. Define the possible issues and prepare workable solutions. Fix the errors or defects found.	22/8/2023
Feedback	Review work done and make necessary improvements.	26/8/2023
Sprint 2	**Duration**: 29/8/2023 - 12/10/2023	
Milestone	**Milestone Goal**	**Deadline**
Planning	Planning for additional functionalities and activities to enhance the usability of the system.	29/8/2023
Design	Design the functionalities, including the interface, input/output, validation, error handling, etc.	15/9/2023
Development	Develop the additional functionalities	30/9/2023
Testing	Test the developed additional functionalities based on the requirements and compare them to the expectations	7/10/2023
Deployment	Integrate all additional functionalities and check the interaction between the functionalities and the database within the system. Define the deployment environment, and processes/ steps for the developed system. Define the possible issues and prepare workable solutions. Fix the errors or defects found.	11/10/2023
Feedback	Review work done and make necessary improvements.	16/10/2023
Sprint 3	**Duration**: 20/10/2023 - 04/12/2023	
Milestone	**Milestone Goal**	**Deadline**
Planning	Planning for testing activities.	20/10/2023
Design	Design test plan.	30/10/2023
Development	Develop and prepare test plans/cases or experiment plans. System Preview with supervisor.	13/11/2023

Testing	Execute test cases.	27/11/2023
Deployment	Final System Preview with Supervisor and Moderator.	04/12/2023
Feedback	Review work done and make necessary improvements.	18/12/2023

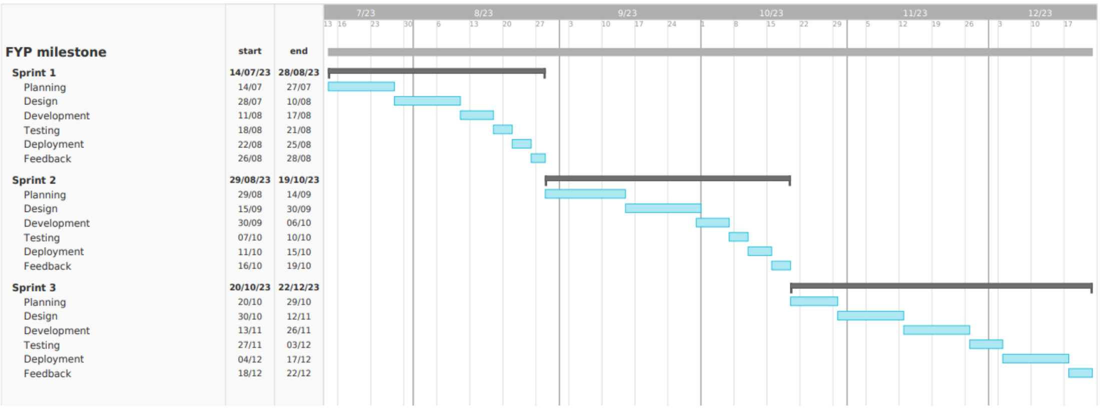


Figure 1.2: Gantt Chart of Software Development Stage

1.6 Project Team & Organisation

The diabetes system is designed to assist diabetes individuals in monitoring and managing themselves effectively. One of the key focuses of this system is to provide additional ingredient information, allowing diabetes individuals to make informed decisions and manage their diet more efficiently. Through the ingredient scanning module, the system can provide appropriate dietary explanations, guidance and suggestions to the users. In addition to the ingredient scanning module, the system comprises a series of modules which aim to facilitate a systematic approach to diabetes management. The modules of the system are listed below:

- User Management Module
  - This module manages the users, including the patient and doctors, and the relevant information.
- Glucose level recording module
  - This module allows the users to record their daily measured blood glucose levels. It stores the glucose readings over time, allowing the users to review and analyse their glucose patterns. The data is shared with the healthcare providers to enable remote monitoring.
- Ingredient scanning module
  - This module provides the user with relevant packaging food consumption suitability suggestions by scanning the ingredient list on the packaging with necessary glycemic information included.
- Reminder and notification module
  - Customisable notifications are allowed in this module to remind the user to carry out their monitoring task or medications. Alerts will be prompted to the user when certain situations that require immediate attention are detected such as the glucose level approaching a risky status.
- Reporting module
  - This module analyses and visualises the collected data, enabling individuals with diabetes or healthcare providers to have better insight. It can also generate reports that can be useful during healthcare consultations.

1.7 Chapter Summary & Evaluation

As a summary of this chapter, the prevalence of diabetes is on the rise and is expected to continue increasing in the future. Therefore to address the growing population of diabetes and to ensure every individual can receive proper and timely diabetes care, this project proposes a diabetes monitoring system. The proposed system aimed to facilitate systematic monitoring and management of diabetes, alleviating the potential burden on healthcare providers while allowing the patients to effectively self-monitor and manage themselves. The proposed system includes features such as monitoring blood glucose levels with customisable reminders to keep individuals with diabetes on track with their monitoring routine. Additionally, an ingredient scanning feature that provides dietary suggestions tailored to individuals with diabetes assists individuals in making informed dietary choices. By scanning the ingredient list on the packaging, the system can provide insights such as suitability for consumption to aid individuals in navigating the complexities of food choices. To achieve the objective of this project, image processing technology will be utilised in the implementation to extract the information from the ingredient list captured. Optical Character Recognition (OCR) technique will be employed to recognize and extract the text accurately. The extracted text will then be analysed and interpreted to generate related dietary suggestions.

Chapter 2

Literature Review

2.1 Project Background

A Diabetes monitoring system refers to a systematic and organised approach to monitoring and managing diabetes. It should involve providing convenience to diabetic individuals in monitoring their current status and improving the effectiveness in identifying possible abnormalities. This can be achieved by providing a user-friendly interface for diabetes individuals to record their necessary measurements which is essential for future tracking. Other than that, the system should be able to support the measurement data retrieval actions by the users and present them in a simple visualisation format. This can eliminate the issue of diabetes individuals recording irregular or inconsistent data, causing data loss or retrieval failure in the future. Not only that, by integrating the diabetes monitoring system through both patients and their healthcare providers, remote monitoring can be achieved. According to Nasrabadi et al, diabetes individuals are easily falling into missing care, leading to unsuccessful diabetes management due to inadequate knowledge, negligence in prevention, extreme poverty and so on (Nasrabadi et al., 2021). To ensure timely intervention in diabetes, and help individuals in preserving their health, a diabetes monitoring system can be the best solution in providing guidance and acting as an assistance tool to the individuals associated with remote monitoring from their healthcare providers. Remote monitoring can perform more effectively than usual care in managing diabetes (Tchero et al., 2019).

Remote monitoring can offer many benefits in the healthcare industry such as allowing early and ongoing discovery of ailments, capacity to constantly screen patients, avoidance of exacerbating illnesses and death, cost decrease in hospitalisation, and acquiring progressively exact reading while promising the productivity of the healthcare centres (Chatrati *et al.*, 2022). While there are several applications available in the market to assist individuals with diabetes in carrying out daily monitoring routines, most of those applications provide functions and features limited to the individual with diabetes. Those applications are not integrated or connected to any verified healthcare centres, which increases the ability of timely detection by healthcare professionals when abnormalities or potential issues arise. When individuals with diabetes rely solely on self-monitoring, they must be highly vigilant in their daily routines, carefully analysing and observing their health trends to stay informed about their current status. However, due to shortcomings in self-monitoring, there is a risk of certain conditions or situations being overlooked or dismissed, potentially leading to severe consequences. On the other hand, the current diabetes monitoring systems are implemented with limited monitoring and management functionality. Most of these systems primarily focus on data recordings, such as blood glucose levels and insulin doses, without providing additional guidance or recommendations to individuals for promoting better and healthier habits. While data recording is undoubtedly an essential aspect of diabetes management to ensure future tracking, it is insufficient on its own to address the complex nature of the condition. Individuals with diabetes require comprehensive support beyond mere data collection. They need guidance on various aspects of their lifestyle, including diet, medication adherence, and overall self-care.

Individuals with diabetes must strictly adhere to dietary control as diet is crucial in managing diabetes. Different dietary habits can have notable effects on their overall health and well-being (Adeva-Andany *et al.*, 2019). Therefore, the individual requires careful consideration of the types and quantities of food consumed. They need to have a comprehensive understanding of the ingredients and be aware of potential risks associated with their dietary choices. However, due to unprofessionalism, identifying the possible risks from consumption is challenging for them. The complexity of nutrition, the limited access to diabetes-related nutritional facts and the vast array of available food products can make it difficult for individuals to accurately assess the health value of the foods they consume. Moreover, subjective assessment without professional knowledge of the food value may lead to biassed decision-making (Aschemann-Witzel, Varela, & Peschel, 2019).

Not only that, the information conveyed through the food packaging often brings challenges in terms of limited literacy level to the consumers. Diabetes individuals must have a clear understanding of the nutritional content or the ingredients contained within the food they consume to effectively manage their conditions. However, when food labels use complex and scientific jargon, individuals with limited literacy skills may struggle to comprehend the information presented. This can lead to confusion and potentially impact their ability to select suitable foods that align with their dietary needs. Moreover, the small font size used on food packaging may increase the challenges for individuals, especially those with visual impairments or the elderly, to read and extract the necessary information. This causes individuals to struggle to comprehend these labels, hindering their ability to make informed decisions about their food choices.

Individuals with diabetes often face inconvenience when understanding the information about the ingredients listed on food packaging. The process of reading and understanding the information listed may require extra time and effort. This can happen when individuals encounter unfamiliar vocabularies or jargon that require them to conduct online searches for related information, increasing the inconvenience of purchasing suitable food. Despite efforts to provide ingredient information, food packaging labels sometimes lack clarity and are not detailed enough. The listed ingredients might use generic terms or include ambiguous descriptions, making it difficult to discern the exact composition of the product and its potential impact on blood glucose levels. This requires the individual to carry out repeated searching to get further details about the food to preserve safe consumption. These challenges cause extra difficulties in understanding the ingredient items listed on the packaging.

2.2 Literature Review

2.2.1 Analysis and Review on Ingredient Scanning System/Feature

An ingredient list is detailed information about the components present within a specific product. It is typically provided by the food manufacturer or producer and displayed on the food packaging. Providing an ingredient list aims to empower the consumer's ability to make informed nutritional decisions on the product they are considering (Idachaba, 2022). Through an ingredient scanning system or feature, users can acquire information from the listed ingredients, allowing them to make well-informed dietary choices that align with their specific needs and preferences. Several studies have explored ingredient scanning systems, each offering unique strengths and capabilities.

In a case study accomplished by Khairani et al (2022), a specific ingredient scanning system focused on detecting halal food ingredients through named-entity recognition (NER) and optical character recognition technology was explored. The objective of this study was to develop an approach that can read and recognise the compositional entities listed on the packaged products for halal consumption classification. The proposed system utilised OCR to extract the ingredient information from the packaging, which was then used for the training process of the NER model. The NER model was trained to identify and classify ingredients such as Halal, Haram (forbidden), and Syubhat (doubtful). The proposed system achieved an OCR accuracy value of 90% and a NER model accuracy of 84% in the testing process involving 24 packaged products. With an overall accuracy score of 79% on the entire system performance, the authors concluded the primary goal was achieved even though there were some errors. The authors highlighted the combination of OCR and NER technology to develop an ingredient scanning system with the ability to classify the detected ingredients with a workflow shown in the figure below.

Figure 2.2.1.1: System architecture by Khairani et al

The proposed system begins with the user scanning the ingredient list on the packaging. The OCR technology of AWS Textract System, which is a machine learning service that can automatically extract textual content, was implemented to facilitate the text extraction process. Subsequently, the system employed template matching techniques to identify object features within the images and recognized individual characters. The extracted text was then subjected to food entity classification using Named Entity Recognition services provided by the spaCy library. A classification process then categorises the food entities based on their groups into relevant halal labels (Khairani *et al.*, 2022).

This research emphasises the implementation of NER technology within an ingredient scanning system. NER is a computational technique that aims to recognise mentions of rigid designators from text belonging to predefined semantic types such as a person, location, and organisation (Li et al., 2022). It serves as the foundation of natural language processing to understand the entity and its traits. By leveraging NER, the ingredient canning system can effectively categorise the ingredient items on the list into their belonging groups, accelerating the process of classification of food items.

Similar work on a mobile application for food ingredient halal status identification with Optical character recognition has been researched by Kartiwi et al (2019). In this research, they focused on the functionalities and procedures of the system to allow the user to identify and verify the halal status of certain food ingredients. The system workflow involved scanning the ingredients using the mobile application's OCR feature. The OCR technology will extract text from the image and send it to the server for halal verification. Once the verification process is completed, the ingredient information along with its halal classification will be displayed to the user. If there are any missing ingredients from the database, a request will be triggered to report the incident (Kartiwi et al., 2019).

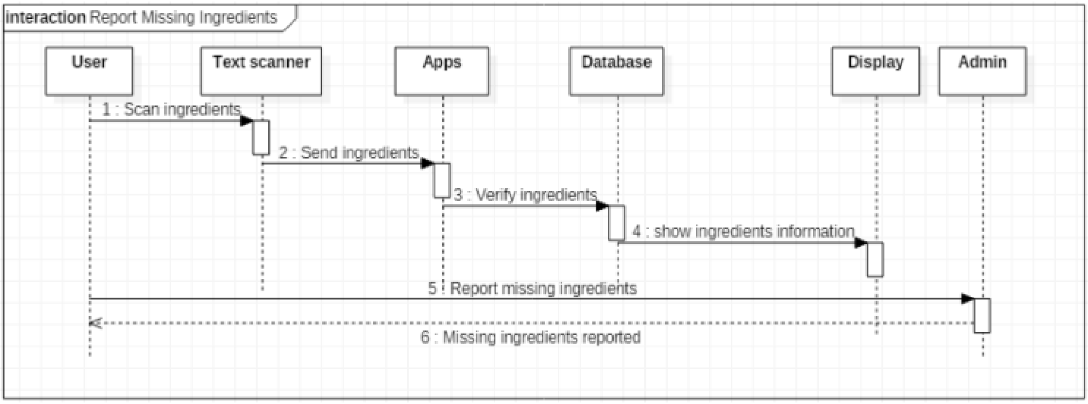


Figure 2.2.1.2: Sequence diagram for the food ingredient halal status identification

In a recent study by Javadi et al (2020), a smart food scanner system leveraging the mobile edge computing concept was introduced. The proposed system aimed to empower users in effectively measuring and analysing their food intake, thereby facilitating informed nutritional decision-making. The system architecture was designed with a focus on edge services, consisting of an IoT interface that facilitated interaction with the edge server to perform requested functions. Additionally, the cloud services are designed to support activities such as computationally demanding to be executed in mobile devices or the edge, data storage and visualisation.

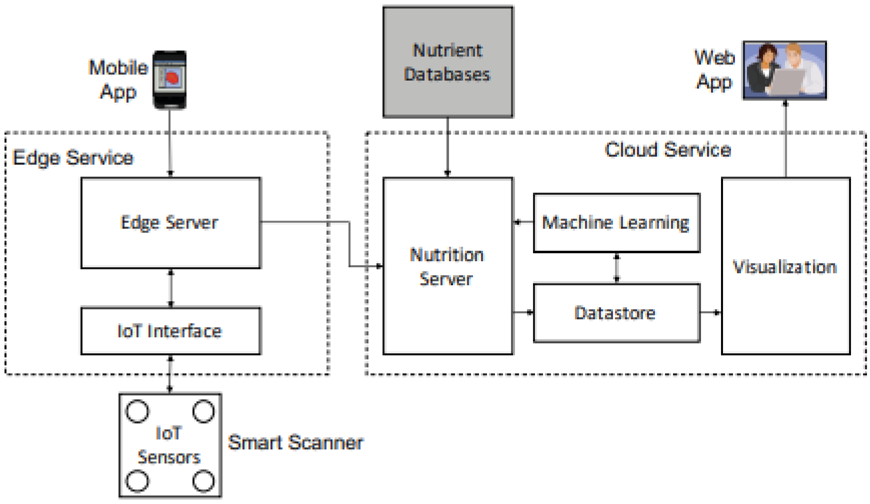


Figure 2.2.1.3: System architecture of a smart food scanner system with mobile edge computing concept

Mobile edge computing is a paradigm to alleviate the heavy workload on the cloud by delegating parts of the computation to devices at the edge of the network, in closer proximity to the end users and IoT devices. The proposed system utilised mobile edge computing techniques to offload the application computation and communications to the edge. This brings the computation closer to end users and IoT devices, resulting in benefits such as improved battery life, enhanced processing capacity, and an enhanced user experience.

The system development involved the integration of various IoT devices and sensors for data collection. These devices collected relevant information about the food being scanned, including nutritional content, ingredients, and portion sizes. The collected data was then processed and analysed within the mobile edge computing environment. The authors indicated that the developed system was able to demonstrate its usability in consuming fewer mobile resources in terms of computation, communication and storage (Javadi *et al.*, 2020). This research emphasises the implementation of mobile edge technology within a food scanner system to address potential issues that may arise from large-scale computing processes on the hosted cloud. The issues, such as user experience degradation and latency, can significantly impact the system's performance. By adopting mobile edge clouds, the proposed system can overcome these potential challenges and ensure better service continuity and a smoother user experience (Siriwardhana et al., 2021).

Another study held by Mahdi, Buckland and Chicott(2023) that focused on the economic and health impacts of a food scanner system, named Change4Life Food Scanner app, revealed another insight into ingredient scanning systems. This app offers users information regarding the nutritional content of packaged food through barcode scanning. Upon scanning the barcode, the app retrieves relevant details about the product, which are then displayed with a colour-coded system to highlight the food's nutritional quality (Mahdi et al., 2023). The effectiveness of the Change4Life Food Scanner app in providing effective information to reduce sugar intake among children is investigated in another recent research. The research assessed the feasibility and acceptability of the Change4Life Food scanner app, revealing an acceptability rate of 64% among the participants who took part in the survey. Not only that, 86% of the participants provided positive feedback, indicating that the system was useful in supplying information related to the nutritional labels on the packaging and rated the system as easy to understand (Mahdi et al., 2022). This research has revealed the positive attitude of the community towards the implementation of such a food scanning system. The system that can provide users with easy-access information about the nutritional quality of food items is accepted by the community.

2.2.2 Object detection using TensorFlow and OpenCV

Optical character recognition (OCR) is a computerised process of converting text from images, such as documents or printed text, into digital text. It involves the processes of extracting textual content from images and transforming them into digital characters format to preserve them digitally for further analysis. The OCR framework can be applied to various types of text, including handwritten text, machine-printed text and even specific text such as Urdu languages. The early OCR systems required training on each character image and being able to work only one character at a time. In their initial stages, these systems were capable of recognizing approximately one word per minute. However, modern OCR systems have exhibited improved performance by providing developers with some powerful tools, open-source libraries, and frameworks that can speed up image processing (Rishabh & Garg, 2020). Among those commonly seen are TensorFlow, OpenCV and Tesseract.

Computer vision is an interdisciplinary scientific field that concerns the issue related to gaining a high-level understanding of digital images or video at the computer level. A study performing automatic number plate recognition (ANPR) utilised OCR technology to read the vehicle registered plates. In this study, the libraries, TensorFlow and EasyOcr were employed in the proposed system to carry out the detection and recognition tasks (Burkpalli et al., 2022).

The ANPR system consisted of four stages, mainly image acquisition, licence plate extraction, character segmentation and character recognition. An image captured in RGB format is muchly affected by elements such as system distortion, system commotion, presentation lacking or vehicle motion, leading to quality degradation. Thus, the captured images were converted into grey, performing pre-processing steps such as clamour evacuation and border enhancement for brightness in this ANPR system.

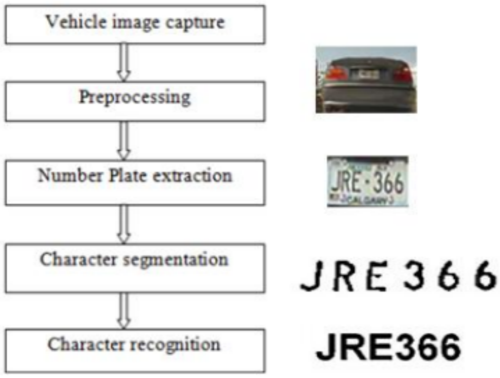


Figure 2.2.2.1: Processes of ANPR system and the output of each process

TensorFlow was implemented to detect the character on the number plate. It is a framework developed by Google, providing a set of comprehensive methods and functionalities for building machine learning models. It supports computer platforms and deep learning algorithms and is highly stable, allowing the user to build a neural network structure without complex coding. This reduces the cost of deep learning and allows easy verification of algorithms (Sheng et al., 2020). TensorFlow was utilised by the authors to build a character detection model that can locate and isolate the characters on the licence plates. EasyOCR, a font-dependent printed character recognition library based on a template-matching algorithm, was integrated to process the detected characters. It compared the detected characters from the number plate image to a predefined set of templates to extract the correct alphanumeric information of the number plate from the images.

This research demonstrated the integration of TensorFlow and EasyOCR in an ANPR system. It has shown the effectiveness of these technologies to detect and extract characters from images and perform accurate recognition output. This has been proven by the ability of the system in the research studied to carry out accurate character recognition on number plates.

Figure 2.2.2.2: Sample output from the system to extract number plate characters

Other than the aforementioned OCR tools, other OCR tools are widely used such as Tesseract. Anuradha and Sengar(2020) have conducted similar research that studies OCR technologies suitable for automatic information extraction from images. In this research, they implemented a classic workflow model for their proposed OCR system, encompassing preprocessing, segmentation, feature extraction, classification and recognition.

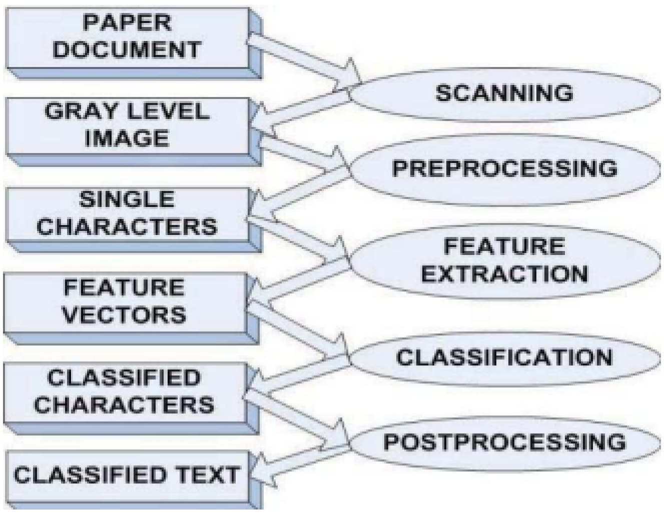


Figure 2.2.2.3: Workflow model of an OCR system

The proposed system was implemented as a web application via the Django web development framework as the system's backend. Several libraries were incorporated into the development process, including OpenCV, TensorFlow, and Tesseract. OpenCV, known for its comprehensive image processing capabilities, was utilised for image segmentation and extraction while Tesseract, an OCR engine that can recognise more than 100 languages in-built, was included to support Unicode recognition processes. As for TensorFlow, it was used to train and infer deep neural networks for predicting characters within the OCR system proposed. TensorFlow is a machine learning open-source software that can perform several tasks but is particularly focused on model training in this research.

Predefined functions from OpenCV were employed for the preprocessing phase, which involved applying grayscale filters, removing noise, thresholding, and performing morphological operations on the images. After preprocessing, the authors implemented a technique to create bounding boxes around the text, enhancing the result by enclosing the text within a border. The OCR operation was conducted on the processed text through the method by Tesseract. A machine learning model was then trained using the framework of Keras. The built model was able to obtain an accuracy of 96% in recognising the scanned text (Anuradha & Sengar, 2020).

These researches have demonstrated the effectiveness of integrating different image-processing tools for textual extraction activities. This combination of implementing OpenCV, Tesseract and TensorFlow shows the system's ability to carry out necessary image processing tasks while maintaining a high level of accuracy in the text extraction process.

2.2.3 Analysis of feature extraction as image processing techniques

Incorporating image feature extraction can enhance the effectiveness of the input images for augmentation. Keras is one of the open-source libraries that provides sets of state-of-the-art deep-learning CNN models with pre-trained weights. Some of the example pre-trained models are VGG16, ResNet50 and InceptionV3 are useful in image feature extraction (Sun et al., 2018). These pre-trained models have been trained with large-scale image datasets such as ImageNet and can extract high-level features from images.

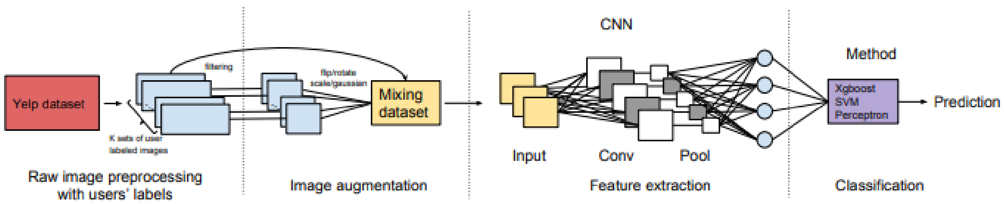
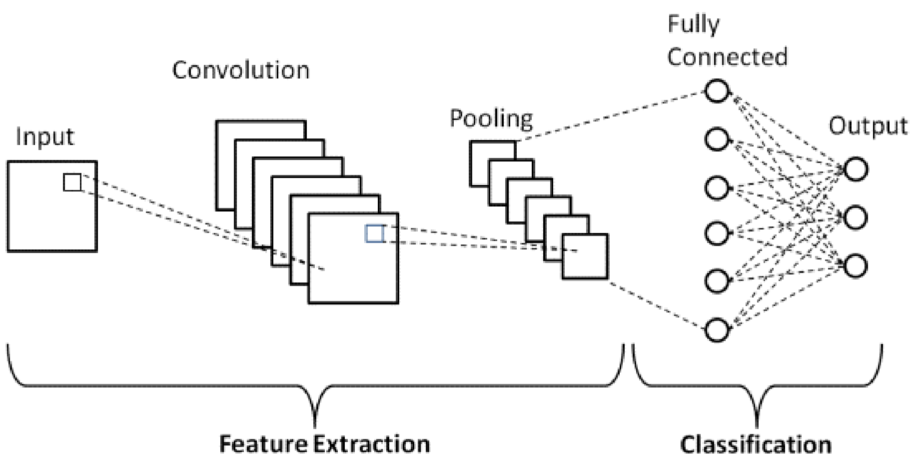


Figure 2.2.3.1: Architecture of machine learning pipeline

CNN is a type of neural network specifically designed for visual data analysis. It employed a hierarchical architecture consisting of convolutional layers, pooling layers and fully connected layers to effectively learn and recognise complex patterns in visual data (Alzubaidi et al., 2021). A CNN owns layers that each learn to detect different features of an image.

The structure of CNNs was inspired by the structure of neuron networks in human and animal brains. It is made up of four important layers, including the convolution layer, rectified linear unit (ReLU) layer, pooling layer and fully connected layer. The convolutional layer applies multiple filters to the input data, performing a convolution operation. This process extracts valuable features by convolving the filters with the input data and producing feature maps that highlight important patterns and structures. The following RELU layer performs an element-wise operation, introducing non-linearity to the network and generating the output of a rectified feature map. This layer effectively enhances the network's ability to learn complex relationships and capture nonlinear patterns underlying the data. In the pooling layer, a down-sampling operation that aims to reduce the dimensionality of the feature map is carried out. Different pooling techniques, such as max pooling or average pooling, are applied to create a pooled feature map. A pooled feature map will be generated in a pooling layer. After the pooling layer, a flattering process will convert the resultant 2-D arrays of the pooled feature maps into a single long continuous linear vector. The flattened vector will be fed as the input to the fully connected layer. The fully connected layer will be able to carry out the classification. Each neuron in the fully connected layer will be responsible for receiving input from the neurons in the previous layer and producing output that will be fed as input for the neuron in the next layers until a desired output is generated (simplilearn, 2023).



\

Figure 2.2.3.2: Architecture of CNNs layers

CNN is able to automatically detect significant features without any human supervision through its conventional layers. Sparse interactions are preserved within CNN to reduce the number of parameters and enhance computational efficiency. Weight sharing in CNNs is a crucial feature that reduces the number of trainable network parameters, leading to enhanced generalisation and avoidance of overfitting. By sharing weights, the network can effectively learn from limited training samples and extract common features (Sarker, 2021). Additionally, CNNs simultaneously learn feature extraction layers and the classification layer, resulting in highly organised and feature-reliant model outputs.

The research conducted by Rohini et al presented a framework that identifies the allergen and nutrient information in fruits and packaged food through OCR and Convolutional Neural Network (CNN). The primary objective of this research is to raise awareness about potential allergic effects that may arise from consuming certain foods. The framework was designed to provide the users with a platform to query about the presence of allergens along with nutritional facts about the food. To achieve this, the framework involved detecting and extracting information from packaged food using the OCR Pytesseract tool. Pytesseract provides various features such as rescaling, de-skewing, binarisation, noise removal and segmentation that help to accurately extract textual information from the packaging.

Additionally, it also involved training a CNN model using a dataset comprising images of various fruits and vegetables, along with their corresponding nutrition facts and allergen information. The CNN model learned to recognize and classify allergens and nutrients based on the dataset. Several CNN architectures identified from Keras, such as InceptionV3, VGG16, VGG19, Inception ResNetV2, ResNet50, Xception, and MobileNetV2, were considered. Transfer learning techniques were employed to leverage pre-trained models with weights, facilitating model development, reuse, and fine-tuning based on the specific requirements of the system.

The final fine-tuned model was employed on the Heroku platform and a Python web framework named Flask was utilised to enable image uploads to the system. The evaluation of this framework indicates the success of the proposed system in achieving its objective. The framework is able to achieve a testing accuracy of 96.6%, precision rate of 97.14%, recall rate of 96.38% and F1 score of 96.75%, revealing a high-performance rate (Rohini et al., 2021).

Another study was carried out by Reddy et al. to classify fish species by, utilising the CNN algorithm. To address the challenges in predicting the species of fish that may cause an allergic reaction in a certain population, they proposed a system that utilised the CNN algorithm for fish detection and classification. The authors discovered that existing systems often exhibit a unique strategy that combines deep learning and machine learning techniques. These innovative works produced a variety of outcomes and established an effective system for classification. Hence, the authors presented a methodology that integrated machine learning and deep learning, involving a series of processes, including image dataset reading, data pre-processing, CNN modelling, training and testing the model and evaluation. The algorithm CNN is able to reduce the large dimensionality of images without causing any information loss, retaining the actual content of the images. Therefore, the authors reported that CNNs are the ideal solution to filter the input data and extract their features. It can filter the fish images and extract features such as colours, edges and so on. During the training process, a large dataset containing 7,200 fish images was used to ensure the model's learning. The effectiveness of the built model was evaluated using 1,800 test images. Throughout their research, it was found that CNN remains the most accurate prediction technique, compared to other algorithms such as Artificial Neural Networks (ANN), FishNet, Support Vector Machines (SVM) and K Nearest Neighbour (KNN) in performing feature extraction for image detection, showing an accuracy rate of 99.6% (Reddy *et al.*, 2023).

The research by Shehan et al. investigated the performance of another algorithm, which is of the YOLO algorithm through real-time traffic sign recognition. They proposed a system that detects traffic signs and provides timely instructions to the users, ensuring awareness of the traffic restrictions and information. The YOLOv3 object detection network was fine-tuned and employed for traffic sign detection, while the classification of the detected signs was accomplished using the CNN-based method. The performance of this proposed system was evaluated and compared to a previous work that utilised Faster R-CNN for real-time traffic sign detection. The previous system achieved a mean average precision (mAP) of 84.5% and a recall of 97.81% (Li & Wang, 2019). The YOLO-based traffic sign detection further improved the previous work by implementing YOLOv3 for sign detection and a custom CNN classifier algorithm for sign classification. Training and evaluation of the proposed system was conducted on the German Traffic Sign Detection Benchmark (GTSDB) and German Traffic Sign Recognition Benchmark (GTSRB) dataset. The proposed system was able to outperform the previous work, achieving an mAP of 92.2% for detection and classification. Furthermore, the time taken for each image to be processed is significantly reduced (P. Rajendran et al., 2019).

2.2.4 Analysis of Machine Learning Algorithm

To determine the degree of suitability for consumption, especially to the diabetes individual, based on the items in the ingredient list, machine learning technology can be employed. Machine learning is a subfield of artificial intelligence(AI) that focuses on developing algorithms and models that can learn from data and make predictions or decisions without being explicitly programmed. It focuses on the analysis and interpretation of data to recognize patterns and extract meaningful insights. By training on large datasets, machine learning algorithms can identify underlying patterns and relationships within the data input, allowing them to generalise and make accurate predictions on new, unseen data (Bi et al., 2019). There are several types of machine learning, the most commonly used are supervised learning and unsupervised learning.

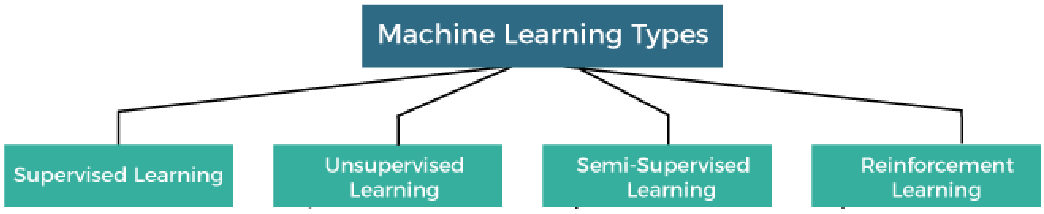


Figure 2.2.4.1: Types of machine learning

2.2.4.1 Supervised Machine Learning

Supervised learning is a machine learning process that focuses on classification and prediction tasks. It aims to describe prediction tasks with the goal of forecasting and classifying a specific outcome of interest (Jiang et al., 2020). In supervised learning, the machine learning model is trained on a set of labelled datasets, where each data point is associated with a known category or outcome. Throughout the training process, it learns to recognise the underlying pattern of the dataset inputted and establish the relationships between the input variables and the target variable. The objective of machine learning is to create a model that can generalise the training data and predict a desired outcome for new unseen data.

2.2.4.1.1 K-Nearest Neighbour (KNN)

Harumy, Ginting, and Manik (2023) have researched and compared the classification methods for detecting sensitive ingredients in food products. The objective of this research was to address the lack of literacy received by the community regarding the composition of ingredients in food products, which has led to a rise in food poisoning issues, and propose an approach to detect sensitive ingredients in food products. Several classifier algorithms have been employed to perform the classification based on product images and categorical information related to food composition, nutrition value, and food characteristics, including K-Nearest Neighbors (KNN), Decision Tree, Support Vector Machine (SVM), Neural Network, and linear regression. The dataset collected for the training of classifier algorithms includes images of food products and categorical information describing the composition, nutrition value and food characteristics. Throughout the entire research, the model developed with the KNN algorithm generates the most satisfied output. The evaluation metrics for the KNN model included a root mean squared error (RMSE) of 0.085, a mean squared error (MSE) of 0.007, a mean absolute error (MAE) of 0.068, and an R2 value of 0.158. The results indicated that the KNN model outperformed the other classification algorithms, with the lowest error rate.

Similar research has been conducted by Hu, Ahmed and L'Abbe (2023), focusing on food categorisation and nutrition profiling for food products. The authors applied the KNN algorithm for food multiclass classification, aiming to predict the food major categories and subcategories and estimate their nutrition quality score. In this research, the authors trained a language model through Natural Language Processing (NLP) to extract unstructured textual content from food labels and the extracted information was utilised alongside supervised machine learning algorithms to automate the classification of food category and nutrition quality score calculation. The KNN classification model is constructed with a k-value of 5, indicating 5 data points in one neighbourhood. By leveraging the distancing traits associated with KNN that assume similar food products exist in close proximity, the Manhattan distance metric between the query food product and other food products is calculated to form decision boundaries. The established decision boundaries form a clear neighbourhood for the KNN algorithm to perform classification. The authors reported an F1 score exceeding 0.9 for the model, which outperformed the traditional bag-of-words method (32%–70%) and structured nutrition facts model (62%).

2.2.4.1.2 Naive Bayes

Hubert et al. (2021) proposed another idea for the implementation of an OCR system with a Naive Bayes classifier to classify promotion images. In this research, an OCR system with Naive Bayes was introduced to automatically determine whether the Internet images contain any promotional information, and define the categories of the images belonging through their textual content. The proposed system is designed to automatically extract textual content from the images and perform classification based on the content extracted. Other than the Naive Bayes model, two more models integrated with Random Forest and K-Nearest Neighbour(KNN) algorithms were developed for comparison purposes. Prior to applying the models for classification, preprocessing techniques were employed to transform the collected images into a suitable format. These processed images were then passed through Pytesseract, an OCR library, to extract the textual content. The extracted text underwent post-processing steps to ensure text consistency and remove unnecessary characters. Among the models developed, the Naive Bayes classifier has demonstrated a commendable performance. It shows 94.31% accuracy, 94.33% precision and 94.11% recall in the classification of promotion images based on the textual content extracted. According to the authors, the observed improvement in accuracy compared to a previous research study, which achieved 75% accuracy on a similar system, can be attributed to the additional preprocessing steps implemented in their approach, which included processes such as stemming, character removal and so forth. By incorporating these steps, the proposed system can achieve a satisfactory classification performance.

Naive Bayes classifier works on the principle of conditional probability. It provides a simple and fast approach to making predictions. The classifier is considered "naive" because it assumes independence among the predictors, even though this assumption may not hold in reality (Ray, 2019). Naive Bayes is able to perform the classification efficiently as it carries out the prediction on the class of the new data point based on the conditional probabilistic of the training data. It can handle both binary and multi-class classification problems and provide probabilistic predictions, indicating the confidence or likelihood of each predicted class. Not only that, Naive Bayes possesses scalability. It can handle large numbers or data points, suitable for applications with high-dimensional data.

2.2.4.1.3 Support Vector Machine (SVM)

In a recent study conducted by Mrouj Almuhajri and Ching Y. Suen (2022), a system was proposed to automatically analyse images and classify sign boards detected into different categories. The system combines the technology of OCR, Natural Language Processing (NLP) techniques and SVM classifier algorithm to carry out the classification task.

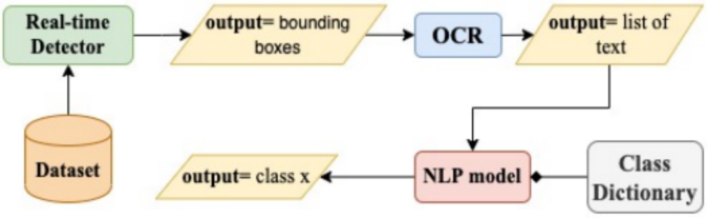


Figure 2.2.4.1.3.1: Abstract view of the framework of shop signboard detection and classification

This study utilised the Street View Text SVT dataset, which contains images of retail store signs, to train and test the model. Two OCR techniques, ABBYY FineReader and Google Cloud Vision OCR, were employed to extract textual content from the signboards in the images. The authors compare the effectiveness of both OCR techniques to determine their ability to detect and extract textual information from the images. According to the result, it was found that the Google Cloud Vision OCR technique achieved a higher percentage of exact matching, indicating a higher text extraction accuracy. The author attributed the low performance of ABBY FineReader to its difficulties in detecting stylised typefaces.

Before classifying the extracted keywords from the signboards, the system applied several NLP techniques to remove potential sources of noise. The steps included lowercase converting, superfluous text data removal, stopwords elimination and spelling correction. The preprocessing processes aim to enhance the quality by providing keywords that represent the actual content. The processed data was then split into a 70/30 ratio for training and testing purposes. Two classifiers, SVM and Multinomial Naive Bayes (MNB), were used for the signboard classification. The SVM model demonstrated promising results with an accuracy of 90.01%. Additionally, evaluation metrics such as precision, recall, and F1-score for each class reached a rate of at least 76%, indicating the effectiveness of the SVM model in accurately classifying signboards (Almuhajri & Ching, 2022).

Research conducted by Sun, Gu and Feng (2018) focuses on food identification via feature extraction and classification. In this research, the authors presented a simple yet effective approach to identifying different kinds of food through raw images. The proposed system incorporated various image preprocessing techniques, including filtering and image augmentation. To ensure the images uploaded to the system are usable, image augmentation techniques such as flipping, rotation, scaling and so forth were applied. These techniques aim to improve the quality of images uploaded, ensuring that the images will not compromise the final accuracy of the classification or cause any degradation of performance. The process of feature extraction is carried out with a Convolutional Neural Network as CNNs can easily transform the origin picture's pixel values into final scores with relatively low computation cost based on fewer parameters in the hidden layers. The implementation of the image feature extraction process utilised Keras, a deep learning framework that supports CNN-based models. Three classification algorithms have been implemented within the proposed system. The authors demonstrated the classification accuracy of the system by tuning the parameters related to augmentation, CNN, and classification techniques, aiming to identify the most effective approach. The system performance was evaluated based on the accuracy of classification on the fraction of testing data. To avoid overfitting, the authors implemented cross-validation and performed the operations on each subset. According to the output, the system showed a significant improvement in the system performance when the image augmentation technique was combined with Support Vector Machines (SVM) to carry out the classification of the images, yielding an enhanced classification accuracy.

2.2.4.1.4 Decision Tree

Aguilar-Loja et al (2022) have researched providing nutritional plans with a decision tree classifier. The authors utilised the decision tree algorithm to manage and provide personalised nutritional plans based on individual patient information. The model focuses on two phases. In the first phase, the nutritionist records data and stores them in the database. The collected data includes various aspects such as weight, height, age, daily physical activities, and health information. The decision tree algorithm is employed in the model to classify the patient data and assign a suitable diet plan based on the information provided. The algorithm considers all the relevant patient perspectives and uses them as input features for the decision tree. By analysing these features, the algorithm generates a classification that corresponds to an adequate nutritional plan for the patient. To ensure an efficient implementation process, the authors develop the model in Python and utilise the scikit-learn library, which provides comprehensive methods of machine learning. This research shows preliminary results of the model with the proposed algorithm, showing an accuracy of 78.5% in defining that employment can provide accurate classification from a considerable amount of historical data.

A similar study has been conducted by Sai et al (2022) to predict personalised meal plans for individuals, promoting proper eating habits and ensuring sufficient nutrition for maintaining good health. The proposed system implemented the decision tree classifier to generate and classify meal plans based on the user's information such as age, exercise lifestyle, blood sugar and so on. The proposed system is developed on the Django framework and Django's SQLlite3 relational database management system is included to store data for the system. With the user information such as gender, age, exercise, and health condition as the training data of the system, the system with a decision tree classifier is trained to predict a suitable diet plan personalised to the individual. The author concludes that the decision tree can perform with high accuracy and F1-score, achieving values of 84.33% and 0.6275, respectively in the study (Sai *et al.*, 2022).

The research conducted by Chitayae and Sunyoto in 2020 aimed to classify different types of mushrooms and determine whether they are poisonous or safe for consumption using a decision tree method. The Classification and Regression Tree (CART) algorithm, one of the Decision Tree methods, is employed in this research. This algorithm is chosen due to its capability to derive decision rules, handle missing values and apply them to work on large amounts of data. To assess the performance of the decision tree method selected, they compare it to another classification algorithm, known as K-Nearest Neighbors (KNN), to find out the better classifier. By conducting experimental analysis on the UCI mushroom dataset, the authors evaluated and compared the effectiveness of the proposed decision tree algorithm against the KNN algorithm in performing mushroom-type classification. The results obtained indicate that the Decision Tree method has better performance. It achieved a performance with an accuracy value of 0.9193, a precision of 0.9227, a recall of 0.9193, and an F1 score of 0.9210.

2.2.4.2 Unsupervised Machine Learning

Unsupervised learning is a machine learning approach that deals with clustering tasks. In unsupervised learning, it infers a function that captures the relationship within the dataset without having a measured outcome (Jiang et al., 2020). It aims to uncover the meaningful structure or patterns underlying, identifying similarities between the data points to form clusters. By examining the statistical properties and relationships, unsupervised learning algorithms generate outputs that consist of groups of data points, known as clusters.

2.2.4.2.1 K-Means

The research conducted by Anis Suraya Rosmahadi and Fauzee Hamda (2022) focused on clustering food based on their nutritional information using the K-means method. It examines and clusters the food through their composition such as calories, carbohydrates, protein, fat, cholesterol, sodium and sugar. The dataset used in the research consisted of information from a food nutritional facts database, which provided the necessary nutritional information for each food item. A total of 60 different foods were selected for clustering in this study. The finding of this research revealed the optimal solution of clustering was achieved when the k-value equals 4. This indicates that the dataset was being clustered into a total of 4 groups. The authors have built models with different k-values to observe their performance and number of iterations. The model with k-values equal to 4, which is accomplished within three iterations, was determined as the ideal solution after comparing the clustering outputs by different k-values. For other models, they undergo a minimum of four iterations to generate the clustering output, indicating that more processing power is required for the clustering processes.

K-Means is an iterative algorithm that works on a distance metric to cluster data points into groups or clusters based on their similarity. With distance as the metric and a k-value class given, the algorithm calculates the distance means, giving the initial centroid, with each class described by the centroid (Yuan & Yang, 2019). Based on the research by Anis Suraya Rosmahadi and Fauzee Hamda (2022), to ensure a satisfied clustering output, testing on different k-values will have to take place. This iterative process consumes significant computational resources. Not only that, this algorithm is unsuitable for this project on determining the consumption suitability based on the ingredients. The vast variety and complexity of food composition make it difficult to determine the appropriate K-value accurately. Choosing an incorrect K-value may lead to suboptimal clustering results. Furthermore, the performance of K-Means drops when the clusters are non-globular or irregular in density (Ray, 2019). Given the unpredictable nature of cluster shapes and densities in an ingredient composition dataset, it is difficult to guarantee that K-Means is capable of capturing the underlying structures effectively. Therefore, considering the limitations, K-Means will not be selected for the implementation.

2.2.4.2.2 Hierarchical Clustering

Hierarchical clustering can be an effective alternative to the k-means algorithm that consumes lower overall memory utilisation (B. *et al.*, 2020). Unlike k-means, which requires storing all data points and assigning the cluster in memory, hierarchical clustering does not require the entire dataset to be loaded at once. It is built in the hierarchical structure of clusters, representing a dendrogram.

A recent research aimed to discover the degree of food processing by clustering foods based on their nutritional properties. The research proposed a system that implemented several unsupervised machine learning algorithms, including K-means clustering, hierarchical clustering, probabilistic clustering and spectral clustering, to ensure the accuracy of the generated output. These algorithms were trained with data retrieved from the Food and Nutrient Database for Dietary Studies (FNDDS). To ensure accurate results, the research includes training for both raw and normalised data to compare the results obtained in each scenario. Among the algorithms tested, Gaussian mixture models and K-means clustering demonstrated the most favourable outcomes in certain cases. However, overall, K-means clustering was found to be the most suitable algorithm for the given task. On the contrary, hierarchical clustering, specifically the Birch method, was the weakest among all the clustering algorithms performed and computationally expensive. The research explained the poor result from the hierarchical algorithm as it is more suitable for scenarios that allow flexible cluster creation to maximise their metric. On the other hand, spectral clustering exhibited satisfactory performance in distinguishing the degree of food processing based on nutritional composition. It achieved a Gini coefficient of 0.237, which was the lowest value among all the clustering methods used (Marcos *et al.*, 2022).

Clustering algorithm	Gini coefficient		
	Dataset 1	Dataset 2	Dataset 3
K-means	0.2099	0.4885	0.5079
Agglomerative	0.3406	0.4349	0.3773
Birch	0.0914	0.3133	0.2731
Gaussian mixture	0.3150	0.5437	0.4568
Spectral	0.2723	0.4492	0.3811

Figure 2.2.4.2.2.1: Result comparison for different clustering algorithms

The authors acknowledge that the clustering algorithms can perform well in certain conditions. However, based on the evaluation of the Gini coefficient obtained from the generated output, it was observed that these clustering algorithms did not yield a satisfactory performance. The Gini coefficient is the metric used to measure the dissimilarity or imbalance within clusters, ranging from 0 to 1. However, the values of the algorithm from this research reflected that these clustering algorithms are not the most suitable algorithms to implement in such cases, therefore performing poorly.

2.2.4.3 Analysis of the selected algorithm

Unsupervised machine learning identifies patterns to form numbers of clusters, in which the data points within each cluster share certain properties. It is a counterpart of supervised learning, discovering the patterns from intrinsic data structures without crafted labels (Wang & Biljecki, 2022). However, based on the above study, the overall performance of unsupervised machine learning algorithms is not satisfactory, performing weakly compared to supervised machine learning. Some of the algorithms consume much computational resources to achieve the best possible output while some produce output that is found to be imbalanced. Therefore, the algorithms of unsupervised machine learning are not considered suitable for the implementation of this project to determine the consumption suitability for diabetes individuals based on the ingredient list of the packaged food. This is also due to the behaviour of unsupervised machine learning algorithms that are unable to provide exact consumption suitability predictions as the final output. Without specific labels or guidelines on the incoming data, it may cause extra challenges in the proposed system to carry out desired outcomes and provide related information to the users.

Therefore, the focus of the selected algorithm will be focused on the algorithm within the supervised machine learning category. Based on the above studies, the algorithm K-Nearest Neighbour has been selected. This can be supported by the achievement of the research on the food categorisation and nutrition profiling system which is highly similar to the ingredient scanning feature of the proposed system. This research on KNN classification shows the ability of KNN to outperform other algorithms such as SVM, decision tree and linear regression in the same classification case. While their research aimed to estimate nutrition scores, the proposed system focuses on determining the consumption suitability of a particular ingredient for individuals with diabetes. Although the research on the algorithm Naive Bayes has proven its effectiveness through a high accuracy rate for classification and shown a better result when compared to KNN, Naive Bayes, which is an algorithm working based on conditional probability, is unsuitable. This algorithm works by indicating the confidence and likelihood of each predicted class, as it may highly rely on the training dataset. However, ingredients composition may vary from each other, making conditional probability prediction unsuitable and inefficient to work.

KNN performs classification of the new data point through majority votes of its neighbourhood. This makes it relatively simple to understand and implement. By considering the consumption suitability of similar ingredients, the proposed system can perform prediction on the new ingredient items, determining the suitability through the foundation set by the neighbourhood. Not only that, KNN provides an extremely flexible scheme as it can handle a variety of data types and applies to both classification and regression tasks (Ray, 2019). In the scenario of multi-model classes or categories not strictly separable by linear boundary, KNN can make better predictions through consideration of the majority votes within the neighbourhoods.

This makes KNN suitable for the proposed system as ingredients may not have a clear separable category. Moreover, KNN is robust to noisy data. Since KNN focuses more on the data points within its neighbourhoods, individual noisy data points are less likely to significantly impact the prediction. This robustness is beneficial for the proposed system, as it can handle ingredient data that may contain noise or outliers.

2.3 Feasibility Study

Technical Feasibility

Technical feasibility refers to evaluating the technical complexity of the system and determining whether the system can be implemented with the existing technical resources. It is important to ensure current technology resources can support the development of the system and achieve the goal. In the proposed system, Flutter will be used as the development tool for both the front end and back end of the system. To build the ingredient scanning feature, open-source libraries such as TensorFlow and Tesseract will be employed. Tesseract is used to perform text recognition, extracting the textual content from the images captured. The interaction of Tesseract allows the system to perform the OCR task efficiently. Not only that, documentation and guidance are provided by these libraries to ease the integration of the libraries into the system. TensorFlow is an end-to-end open-source platform that provides a comprehensive environment to develop and train machine learning models efficiently. In this system, TensorFlow will be utilised to carry out necessary classification tasks on consumption suitability. The camera embedded in the smartphone of the users is necessary and will be used for the ingredient list image capturing. By providing the necessary guidance and validation, the users are expected to be able to capture the images accurately.

Social feasibility

Social feasibility aims to define the degree of acceptance by the public regarding the proposed system. This can be evaluated through the social impact that potentially happened. In this system, the social feasibility will be assessed from the perspective of the diabetes individuals and the healthcare providers. Through the implemented ingredient scanning feature into the system, individuals can reduce their reliance on frequent consultation with their healthcare for proper consumption. This provides convenience for those individuals who have difficulties to make frequent physical consultations. However, certain users such as elderly generations and those slow learners may face challenges in learning and aspiring to the new system. Therefore, it is crucial to consider their concern through communication, making the user interface of the system friendly to them. Additionally, users may also be concerned about the accuracy of the information provided by the system, questioning whether the system is capable of providing reliable information. Furthermore, data stored in the system such as the user's personal health information must be handled securely and with strict confidentiality, so that the users have the confidence to continue using the system.

From the healthcare provider's perspective, the implementation of a diabetes monitoring system can potentially result in an increased workload for healthcare providers. Additional responsibilities will have to be handled such as remotely monitoring their patient’s health status through the system, providing necessary feedback and so forth. Not only that, they are forced to learn the procedures of the system and accommodate the system with their daily working routine. It is crucial to assess the impact on their workload to ensure the system integrations do not overwhelm them. Adequate training can be provided to them to facilitate their learning and adaptation.

Economic Feasibility

Economic feasibility defines whether the development of the system is cost-effective and feasible to be completed within the budget. The system development is considered to start with a low medium startup cost as the system will utilise open-source libraries such as TensorFlow, Tesseract, etc. These libraries are free of charge and provide sufficient materials for the users to learn and implement the techniques provided. During the beginning phase of the system development, the main requirement involves a laptop or desktop computer for the coding and debugging processes which are already available within the development team or can be acquired at low cost. Android Studio is available to be used for the system development for free, this further minimises the expenses during the development phase. During the integration or deployment phase, this system does not require additional significant expenses. The diabetes individual can download and access this software freely once the system is developed. However, to ensure the security of the stored data, additional efforts and investment in the database should be placed, implementing robust security measures to protect the information. Furthermore, by considering possible conditions of increasing users adopting this system, the database must be designed to be scalable and able to accommodate the growing data volume and user demands in future, supporting a large user population.

Operational Feasibility

In terms of operational feasibility, the ability of the proposed system to resolve the existing problems and contribute to society is demonstrated in several ways. The proposed system can increase the willingness of individuals with diabetes to take an active role in managing themselves by providing a more streamlined and user-friendly approach to routine management tasks. For instance, they can gain insights into their current health trend by observing the visualised data by the system, reducing the inefficient manual analysis. This empowerment can lead to increased engagement and better self-management, ultimately preserving their health and well-being. On the other hand, the system has the potential to alleviate some of the burdens placed on healthcare providers. The system can serve as an assistant to reduce some of the workloads by the healthcare providers such as monitoring the patient's daily glucose level and offering guidance on dietary choices by filtering the ingredients, acting as a valuable tool that can enhance the efficiency of their operations. Although the system can make significant contributions, it is crucial to emphasise that the system does not aim to replace healthcare providers entirely. They remain vital in supervising and overseeing the ongoing conditions associated with the system, ensuring comprehensive care and personalised treatment. Implementing the system requires the healthcare providers to invest time and effort into learning and integrating with the system. To ensure seamless operations on both physical and online platforms, the healthcare providers are required to put in more effort which may affect the morale and motivation of the employees within the healthcare centres.

2.4 Chapter Summary and Evaluation

In summary, the proposed system will implement the ingredient scanning feature using the K-Nearest Neighbors (KNN) algorithm, which has been proven effective in similar research. KNN offers a flexible scheme to handle a variety of data types, allowing easier integration. Not only that, it is simple to understand and its implementation is based on the majority votes of the data points within the same neighbourhood, optimising the impact of noisy data that may affect the prediction accuracy. To ensure the performance of information extraction from the images, libraries such as TensorFlow and Tesseract will be utilised. Tesseract will be incorporated to perform Optical Character Recognition(OCR), extracting the textual information from the images while Tensorflow will be employed for machine learning model building and training. Android Studio will be used as the development tool for the effective development of the system based on the methods provided. These libraries and development tools provide open-source and sufficient materials and resources for the developers to easily assess their implementation approach and methods. Thus they are being selected for the implementation of the proposed system to ensure an effective and structured development environment.

Chapter 3

Methodology and Requirements Analysis

3.1 Methodology

Agile software methodology is an iterative and incremental approach to software development that prioritises delivering valuable and functional software frequently, in short and iterative development cycles. It is a lightweight approach that provides flexibility to adopt changes in requirements at any stage of development. Agile manages the task and the coordination of the activities throughout the system development with a set of values and principles. It values customer collaboration, flexibility and continuous improvement, enabling the development team to deliver valuable software in shorter cycles (Al-Saqqa et al., 2020).

Figure 3.1.1: Agile Scrum Methodology

Agile Scrum is a project management framework that embraces the principles of Agile software development. It emphasises collaboration, adaptability, and customer feedback to deliver high-quality software efficiently. In this methodology, the entire development process is organised into several periods, known as “sprint”. The system to be developed will be decomposed into several small increments and each sprint will focus on only one decomposed part. In each sprint, the developer team focuses on implementing a specific set of functionalities, representing one of the decomposed parts of the overall system. The end of each sprint will produce a shippable deliverable that can be checked with each sprint’s end. An agile scrum methodology acknowledges the management of rapid requirements changes throughout the system development processes. Through the iterative development sprint, the team can manage the changes in requirements and adapt them accordingly to the system. This approach also encourages continuous involvement and regular feedback from the stakeholders. Regular review meetings, known as Sprint Reviews or Demos, allow stakeholders to provide feedback on the product increment delivered at the end of each sprint. This promotes evaluation to be carried out with the involvement of related stakeholders to ensure the newly integrated parts fulfil the system requirements (Hema et al., 2020).

3.2 Requirements Gathering Techniques

Requirement gathering is the crucial process that serves as the foundation of system development, gathering the necessary information and data for the system development. Effective requirement gathering remains a fundamental prerequisite for developing and implementing successful systems. The process of requirement gathering, also known as requirement elicitation, focuses on understanding the requirements and documenting them before establishing the system. During requirement gathering, various information will be collected such as the expected system behaviours, features and functionalities to be implemented, specific requirements, etc, to ensure the system can perform satisfactorily and fulfil the expectations of its users. The goal is to gather comprehensive input from all stakeholders that will utilise the system, considering their perspectives and needs. Different requirement-gathering techniques possess unique strengths, and no single technique can address all the peculiarities of a system. Therefore, a combination of various techniques is used to ensure an effective requirement elicitation process. The success of a system development hinges on the effectiveness of requirement elicitation. Complete and accurate requirements collected enhance the ability of the software developer team to clearly understand the requirements to be delivered, reducing the possibility of building incorrect details (Zachariah et al., 2020).

3.2.1 Data Collection Method

Effective requirement collection and management are crucial from the perspective of a project as they perform the groundwork of successful software development, ensuring that the delivered software products fulfil the proposed objectives. In this proposed system, the requirements will be acquired through the techniques of interviews and questionnaires to ensure a comprehensive and well-rounded understanding of the stakeholders' needs and requirements.

Interviews have been selected as the requirement elicitation technique for the proposed project, specifically involving healthcare providers and individuals with diabetes as the interviewees. The healthcare providers and diabetes individuals are selected as the group for the interview as they constitute the primary user base of the proposed system. Thus, it is important to consider their requirements to ensure the smooth execution of their daily routine and the high compatibility of the system with their routine. Healthcare providers such as doctors will be interviewed to gain insight into the consultation process, particularly the information related to dietary advice for individuals with diabetes. On the other hand, individuals with diabetes will be considered as another interviewee to identify their specific demands and preferences when accessing ingredient information, ensuring the system meets their requirements effectively.

The interview has been selected as it allows direct communication, facilitating a deeper understanding of the interviewees' perspectives, concerns and expectations. By conducting interviews, an in-depth interaction between the healthcare provider and diabetes individuals can be facilitated to gain valuable insights and concerns from them. This direct approach enables the team to observe and discuss the system in detail, ensuring a comprehensive understanding of the specific needs and requirements of healthcare providers and diabetes individuals. Not only that, the interview promotes flexibility and adaptability, which allows the interviewer to make necessary adjustments to the ongoing conversation to elicit necessary information and responses from the interviewee. This ensures that all important topics will be explored and the potential ambiguity will be clarified clearly. In the interview session, open-ended questions will be utilised to encourage the interviewee to express their ideas and requirements with rich and unrestricted content. By using open-ended questions, thoughts and expectations from the healthcare providers and the diabetes individuals can be explored further to capture nuanced and detailed responses.

Questionnaires have been chosen as an additional data collection technique to gather information from the public, focusing primarily on diabetes individuals, their guardians, and potential diabetes individuals. By involving these groups as the target of the questionnaire, a comprehensive understanding of the requirements and perspective can be included for assessment. This technique was selected to collect information due to its ability to collect required data effectively and efficiently from a large and diverse population. By distributing the questionnaire online or sending links to the respondents, a wide audience can be reached to gather a significant amount of data in a relatively short time, considering sufficient responses from the target population. Online questionnaires can be carried out regardless of the respondents' geographical locations, allowing them to complete the questionnaires at their convenience. This is especially useful when engaging busy healthcare providers, guardians, or individuals who may have time constraints. Questionnaires are generally more cost-effective than other data collection methods, such as interviews or focus groups, making them a practical choice for large-scale data gathering.

3.2.2 Interview Plan

1st Interview

Interviewee: Healthcare provider

Table 3.2.2.1: 1st Interview Plan

Interview Details:	
Interview Method	Physical Interview
Date & Time	8 July 2023 (10.30 am - 11.15 am)
Duration	45 minutes
Objective	1. To understand the management of patients' information

<table>
<tr><td></td><td>2. To define and understand the details of the patient's health monitoring process
3. To investigate the impact of dietary choices on diabetes management
4. To gather features and functionalities expected for a diabetes monitoring system.</td></tr>
<tr><td colspan="2">Company Details</td></tr>
<tr><td>Name</td><td>Klinik Bersatu Relau</td></tr>
<tr><td>Address</td><td>823-G-12, Jln Paya Terubong, Relau, 11900 Bayan Lepas, Pulau Pinang</td></tr>
<tr><td>Phone Number</td><td>04-646 2600</td></tr>
<tr><td colspan="2">Interviewee Details</td></tr>
<tr><td>Name</td><td>Dr Lim Chong Keng</td></tr>
<tr><td>Position</td><td>Medical Clinic and Health Care</td></tr>
<tr><td>Phone Number</td><td>012-4995600</td></tr>
<tr><td>Email</td><td>chongkenglimdr@yahoo.com</td></tr>
<tr><td>Name Card</td><td>


</td></tr>
</table>

2nd Interview

Interviewee: Diabetes Individual

Table 3.2.2.2: 2nd Interview Plan

Interview Details:

Interview Method	Whatsapp Video Call Interview
Date & Time	9 July 2023
Duration	30 minutes
Objective	1. To explore the routine of diabetes individuals in diabetes management. 2. To gain insight into the utilisation of ingredient lists by diabetes individuals 3. To explore the potential integration of an ingredient scanning feature with existing diabetes management practices.
Interviewee Details	
Name	Ms Lee
Phone Number	012-5386738

3.2.3 Interview Transcript

<u>1st Interview</u>

Interviewee: Healthcare provider

Objective: To understand the management of patients' information

1. **How do you currently collect and store information about diabetes patients in your practice?**

   The system we are currently adopting is a system that looks similar to Excel, consisting of rows and columns to collect the patient's information. There are mainly two interfaces in the system, one displaying the incoming patients list, who are currently waiting outside the consultation room, while the other interface shows the patient's previous medical records. Typically, when a patient arrives, my receptionist will make a record in the incoming list and I will be able to know the awaiting patients.

   Other than the electronic system, we also maintain a manual recording method, which involves handwritten records. This handwritten method records similar information but is a simpler method as it mainly serves as a backup. During a consultation session, I record the relevant information on the patient's card. Later, my receptionist takes these handwritten records and enters them into the electronic system.

2. **What types of patient data or information are typically recorded and monitored in the context of diabetes management?**

   When it comes to diabetes individuals, their blood glucose is the utmost important parameter for our observations. It shows the most direct reflection of their current health status. Not only that, we also keep track of the vital signs such as blood pressure and weight of the patients. These vital signs measured in the current visit will be compared with those from previous visits to assess their health status. These three are the factors that we will make a record to assess their health.

Objective: To define and understand the details of the patient's health monitoring process

3. **What information do you ask the patients to record by themselves for better monitoring?**

   We encourage the patient to record their daily blood glucose as part of the management of diabetes. These self-measurements serve an important role as they not only provide us with the necessary data for monitoring their condition but also empower the individuals by offering immediate insights into the fluctuation of their condition. Even though they may not possess professional knowledge, they can still recognise and detect when their blood glucose readings are high. Regarding the frequency of measurements, we generally recommend patients measure their blood glucose levels before and after breakfast, after lunch, and after dinner. However, the frequencies may vary depending on the situation of the patients. Patients with severe diabetes have to measure their blood glucose levels more frequently throughout the day while for those with mild diabetes, measuring twice a day will be sufficient. Typically, we suggest measuring before breakfast to get the fasting measurement and at random times after meals to assess postprandial glucose levels.

4. **How do the patients share those recorded data with you?**

   The recording methods can vary among patients, particularly among different age groups. Those younger, maybe below 50 years old, normally use free diabetes monitoring applications available from platforms like Google Play to record their measurements electronically. Whereas elderly patients prefer paper-based methods to record their blood glucose levels. Therefore, to ensure a consistent recording, we provide them with a guideline form that can serve as a template for them to measure systematically.

5. **What are the data points or trends that you pay attention to when reviewing patients' information during monitoring?**

As aforementioned, the trend of patterns of blood glucose measurement can provide valuable information and reflect their daily health status most directly. We will assess their health status by comparing it to the standard blood glucose range. In the case of diabetes mellitus patients, two key indicators commonly used are fasting glucose level and random glucose level. Fasting glucose level refers to the measurement taken before any food or drink intake, typically in the morning before breakfast. Random glucose level, on the other hand, is collected at a random time during the day, regardless of food intake. The normal range of the glucose level for fasting should be between 3.9 to 6.0mmol/L while for random measure it should be between 3.9 to 7.7mmol/L. So, to review the patient's health, we will pay attention to these two readings and assess whether their glucose levels remain within a controlled range.

6. **How do you identify abnormal conditions that require immediate attention during the monitoring process?**

   Immediate attention will be required when the patient's blood glucose level remains high for several days. In this case, the patients themselves can detect the abnormalities and meet us immediately when they find the abnormalities. On the other hand, we will get feedback to know if they are facing any diabetes symptoms recently. For patients with abnormal conditions, they will commonly face symptoms such as extreme thirst and frequent urination. Additionally, we will observe their skin surface. Extreme dry skin is another reflection of abnormalities. When these symptoms are found in the patients, we will ask the patient to carry out immediate testing. Their weight during each visit will be recorded and compared. If we find that their weight drops or increases within a short period, this can mean that the patients are not managing themselves well.

7. **What challenges or difficulties do you face in the monitoring process, and how do you overcome them?**

   The major challenge faced is the attitude of the patients towards monitoring. We often find that patients are dishonest or attempt to deceive healthcare professionals during their self-monitoring or consultation sessions. One common example is when patients manipulate their fasting period before the consultation. Ideally, patients should fast for a maximum of 6 hours before the consultation to obtain an accurate representation of their blood glucose levels. However, some individuals may extend their fasting period, sometimes up to 10 hours, in an attempt to artificially lower their glucose levels and make them appear normal during the consultation. Not only that, they sometimes even tend to present us with a “modified” blood glucose measurement to pass the consultation. To overcome this issue, we will request those suspicious patients to perform a HbA1c test immediately. The HbA1c test provides valuable information by reflecting the average blood glucose levels over the past three months. We will try to foster a trusting relationship to encourage the patient to be honest about their self-monitoring and educate them on the importance of providing accurate reporting. Other than this, some of the patients, especially the elderly, tend to forget their medication. For this issue, we try hard to get cooperation from the patient's family to remind them to take their medication on time.

Objective: To investigate the impact of dietary choices on diabetes management

8. **How do the patient's dietary choices affect their diabetes status and overall health?**

   This can vary among different patients. In common cases, when individuals consume a large amount of carbohydrates without engaging in sufficient exercise or physical activities to utilise that energy, it can lead to an excess of glucose remaining in the body. This will lead to an imbalance between carbohydrate intake and energy expenditure, causing excess glucose to contribute to elevated blood glucose levels over time.

9. **What criteria or information should individuals with diabetes consider when making dietary choices?**

   One important criterion that individuals with diabetes should consider when making dietary choices is the glycemic index (GI) level of foods. The GI is a measure of how quickly carbohydrates in food raise blood sugar levels. It provides valuable information on how different foods can impact blood glucose levels. However, it can be challenging for them to get sufficient information about the GI value and include it as a factor of consideration.

10. **What are the common dietary misconceptions among patients regarding dietary patterns?**

    The dietary misconception is common among the elderly population, particularly due to limited exposure to nutrition knowledge. One prevalent misconception is the belief that they are only restricted to consuming foods that taste sweet. They are not aware of the true relationship or factors that cause the health fluctuations. Although we do mention that they should reduce sugar intake, carbohydrates and such glucose-relevant items, some of them could not follow well. Furthermore, some patients express difficulty in feeling satiated without consuming carbohydrates. This perception can make it challenging for them to eliminate carbohydrates from their diet, as they believe it is necessary for achieving a sense of fullness.

11. **What dietary guidelines or approaches do you commonly recommend to patients with diabetes based on their individual needs and preferences?**

    When it comes to providing guidelines to the patient, it is more about educating them. Many individuals have a general understanding that they should reduce the consumption of sweet foods but may lack knowledge about hidden sugars and the broader concept that carbohydrates are not always sweet-tasting. Therefore, we will explain to them that carbohydrates will eventually be broken down into glucose in the body. If the consumed carbohydrates are excessive, it would bring negative impacts as its consequences.

Objective: To gather features and functionalities expected for a diabetes monitoring system.

12. **What features or functionalities do you expect in a diabetes monitoring system to streamline the patient care and monitoring routine?**
    I would expect the system to be able to enable patients to record their blood glucose levels. It would be ideal if the system could include real-time data synchronisation so that I can access and monitor the patients through real-time data. Not only that, I would expect features or functionalities that can provide advice to the patients in their daily lifestyle, probably in terms of diet or exercise.

13. **With an ingredient scanning feature implemented in the diabetes monitoring system, do you think it can help diabetes individuals to perform better self-management and monitoring?**
    Definitely yes. The inclusion of an ingredient scanning feature can greatly benefit individuals with diabetes by providing them with more relevant information about the products they consume. This feature enables them to make informed decisions when purchasing food items, thereby enhancing their ability to understand and analyse the information conveyed through the ingredient list. If it is possible, I would recommend adding extra GI value information being conveyed to the user through this feature. This could effectively help the patient to filter their consumption.

Conclusion of Interview

According to Dr. Lim, the ingredient scanning feature proposed can help assist diabetes individuals in obtaining sufficient information from the ingredient list. This feature could indeed enhance the monitoring approach for diabetes patients, allowing them to make more informed choices about the food they consume. Other than that, he also looks forward to the system's ability to enable doctors to assess real-time measuring data recorded by patients. This capability can help identify possible abnormalities and prompt timely interventions when necessary, contributing to comprehensive and accurate patient care. While the system supports remote monitoring, Dr Lim emphasises the ongoing importance of physical examinations in certain cases. Physical monitoring allows doctors to assess patients' overall physical conditions, observe any visible symptoms, and ensure the accuracy of the data recorded remotely. This hybrid approach, combining both remote monitoring and physical examinations, ensures comprehensive and accurate patient care. Based on this interview, it is evident that the proposed system that can support a hybrid monitoring approach aligns with the objectives and expectations of healthcare providers. The capabilities of offering real-time monitoring and identifying abnormalities, ultimately enhance the quality of care provided to the patients. The idea of implementing the ingredient scanning feature is also supported by the doctor in allowing the patients to manage and monitor themselves better.

<u>2nd Interview</u>

Interviewee: Diabetes Individual

Objective: To explore the routine of diabetes individuals in diabetes management.

1. **How do you currently manage your diabetes on a day-to-day basis?**

   I perform daily blood glucose checks with a glucose metre daily. Every three to six months, I will meet with my doctor based on the scheduled appointments for regular check-ups and adjustments. During the consultation, the doctor will take my blood glucose level on the spot, along with other important measurements such as blood pressure and weight. Other than that, I have reduced carbohydrate intake in my daily practice and replaced them with alternatives. For example, opting for brown rice instead of white rice and avoiding consumption of food that tastes sweet.

2. **What challenges do you face in monitoring and managing your blood glucose levels?**

   The biggest challenge I faced was the inconsistency in recording my daily blood glucose measurements. As I rely on noting down the data in a notebook, it is difficult for me to track the trend of my health over time. I need to spend more time analysing the trends and patterns. I heavily rely on sharing these recorded data with my doctor during appointments to ensure my health is consistently well-controlled. Other than that, I encounter difficulty in tracking the effect of my diet on my health. I can hardly tell which items that I consumed are negatively impacting me. I heavily rely on sharing these recorded data with my doctor during appointments to ensure my health is consistently well-controlled.

3. **What are the steps you followed to manage your dietary intake to preserve your diabetes status?**

   To manage my dietary intake and preserve my diabetes status, I primarily focus on portion control by reducing or eliminating food items that may be unsuitable for my condition. I am cautious about the ingredients and nutritional content of the food I consume. As a result, I prefer not to have meals outside where I have limited knowledge about the ingredients used in the dishes. Additionally, I am mindful of my desire for instant foods that have been listed by my doctor as highly unsuitable for my consumption. This may involve avoiding high-sugar or high-carbohydrate foods, limiting my consumption of processed or fried foods, and incorporating healthier alternatives.

4. **What sources of information do you rely on to make informed decisions about your diet?**

   I usually focus on the information provided by the doctor during consultations. During consultations, my doctor provides me with recommendations and guidance on dietary choices. They also provide me with a list of foods to avoid or eliminate from their diet, particularly those that may significantly impact blood glucose levels or pose other health risks. They also warn me of the consequences of taking excessive sugar or carbohydrates.

Objective: To gain insight into the utilisation of ingredient lists by diabetes individuals

5. **What are the ingredients criteria or factors that you are concerned with to determine if a food item is suitable for you?**

   When making dietary choices, I will focus on prioritising food that has labels stating they are recommended for diabetes or are sugar-free, rather than focusing on the ingredient list. This is because the ingredient list is difficult for me to understand and interpret.

6. **Are there any particular ingredients or additives that you consistently watch out for due to their impact on your blood glucose levels? If so, what are they?**

   I do not consistently watch out for certain ingredients or addictive ingredients when purchasing food products. However, I do notice an impact on my blood glucose levels when I consume certain food items such as Nescafe 3-in-1

Coffee. Although it does not taste sweet, it actually causes a spike in my blood glucose levels. My doctor mentioned that even if they don't taste sweet, they can contain added sugars or other ingredients that can affect blood glucose levels.

7. **How do you incorporate information from ingredient lists into your decision-making process when selecting foods?**

   I prioritise identifying ingredients that are sources of added sugars, such as white sugar, brown sugar, corn syrup, high-fructose corn syrup, honey, and others when reading ingredient lists. If I find those items listed on the ingredient list, I am not going to purchase it has specifically mentioned that the particular item is suitable for me.

8. **What are the challenges you faced when trying to identify suitable ingredients while purchasing food items?**

   One challenge I encounter is the insufficient information I can retrieve from the ingredient list. The ingredient list can sometimes include technical terms or unfamiliar ingredients that are difficult for me to understand. Not only that, the ingredient list does not provide enough information regarding the appropriate or suitable consumption portion in one serving. It is difficult to determine how much of a particular food item constitutes a single serving and how it may impact my blood glucose levels.

Objective: To explore the potential integration of an ingredient scanning feature with existing diabetes management practices.

9. **What specific dietary information would you expect in an ingredient scanning feature to enhance your ability to make diabetes-friendly food choices?**

   I would expect the ingredient scanning feature to be able to provide me with sufficient information so that I can interpret and analyse the ingredient list even if I am not a professional. This feature can simplify complex ingredient information into user-friendly terms and provide. If it is possible, I would like to suggest alternatives for food items as well to provide me with a variety of dietary choices.

10. **How do you see the ingredient scanning feature complementing or enhancing your current dietary management strategies?**
    I think it will be able to guide in making informed decisions on dietary. It would serve as a valuable tool for individuals like me to get the necessary support to navigate the ingredient list and make better decisions. This feature, I believe, would help me understand the potential impact of specific ingredients on my overall health management.

11. **What are the functionalities or features that you would like to see to support your diabetes management goals?**
    I would anticipate the ability to visualise the data recorded. Displaying the blood glucose reading on a graph or chart allows me to better understand and identify my health status more efficiently. Through the visualised data, I can gain insight to identify my health trends and make informed decisions regarding my diabetes management.

Conclusion of Interview

After interviewing the diabetes individual, it is found that the individual heavily relies on the information conveyed by the doctor during their consultations. The interviewee expressed the desire and anticipation for the system to offer data visualisation capabilities for the daily measurements. Visualising the data through graphical representations such as charts or graphs, can help the interviewee to have a better understanding and analyse their health trends. Additionally, the individual does not follow any specific approach to filter their food choices and only becomes aware of the impact of their food consumption if it has caused a noticeable effect on their blood glucose levels. The interviewee specifically highlights the challenges faced when attempting to understand the ingredient list, particularly when encountering technical terms or unfamiliar items. Consequently, it can be inferred that the proposed ingredient scanning feature would be highly valuable to them if it can be seamlessly integrated into their daily routine, ultimately enhancing their diabetes management practices.

3.2.4 Questionnaire and Survey

The questionnaire is created through Google Forms in Google Drive. It is shared with the public through social media platforms such as Facebook, Instagram, WeChat and WhatsApp. This questionnaire aims to assess and explore the respondents' awareness of dietary choices and their utilisation of ingredient lists. This questionnaire consists of four sections and a total of 29 questions. The first section, Section A, collects demographic information of the respondents. Section B investigates the respondents' understanding and utilisation of ingredient lists on food packaging while section C focuses on the respondents' awareness of the impact of their dietary choices on their health. The last section aimed to gauge the respondents' likelihood of using an ingredient scanning feature. In 5 days, a total of 187 respondents have been collected.

Result and Analysis

Section A: Demographics

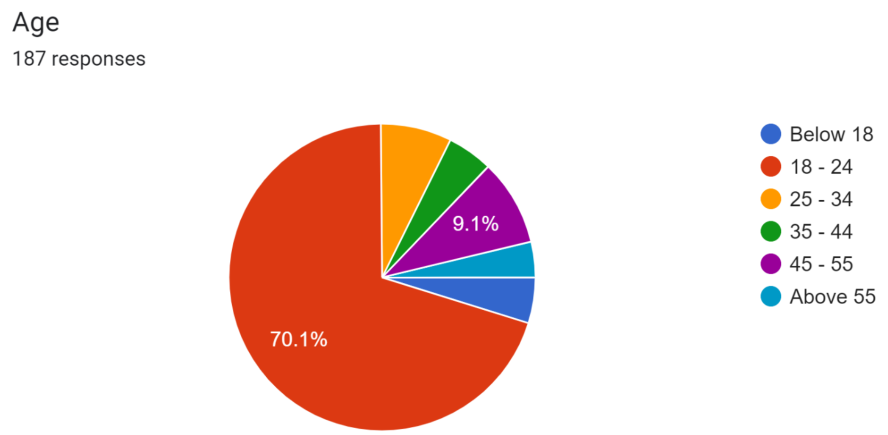


Figure 3.2.4.1: Pie Chart of Question 1 in Section A

In Figure 3.2.4.1, the majority of respondents are within the age group of 18 to 24, 70.1% of the total. The next highest percentage, 9,1%, is found in the age group of 45 to 55 followed by the age group of 25 to 34 (7.5%). The age groups of 25 to 34 and below 18 both constitute 4.8% of the respondents, while the age group above 55 accounts for 3.7%.

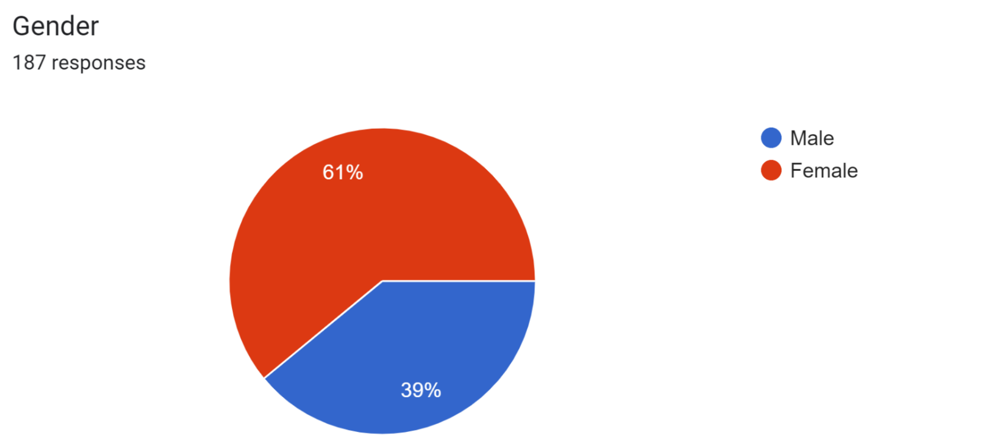


Figure 3.2.4.2: Pie Chart of Question 2 in Section A

Out of all respondents, 114 individuals (61%) of the respondents are female. On the other hand, 73 individuals (39%) of the respondents are male.

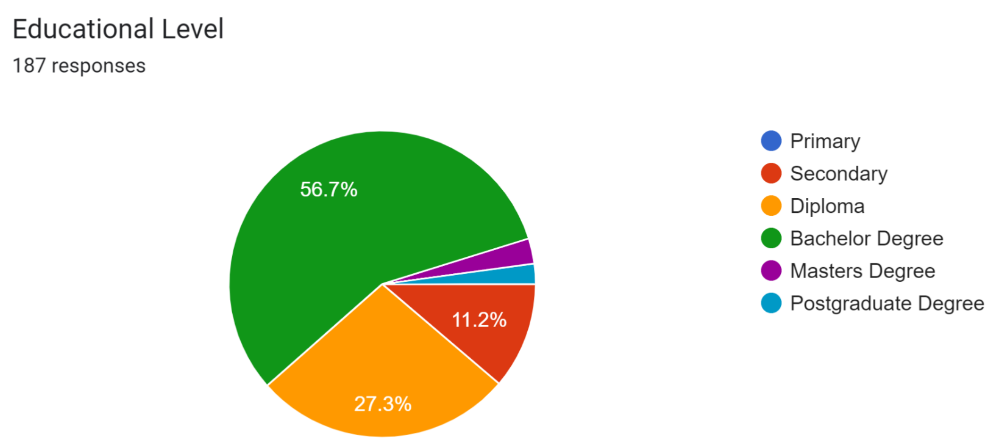


Figure 3.2.4.3: Pie Chart of Question 3 in Section A

The majority of the respondents (56.7%) held a Bachelor's degree as their highest educational level. 27.3% of the respondents had obtained a Diploma and 11.2% of the respondents had secondary education. A small group of the respondents held a Master's degree (2.7%) and a Postgraduate degree (2.1%).

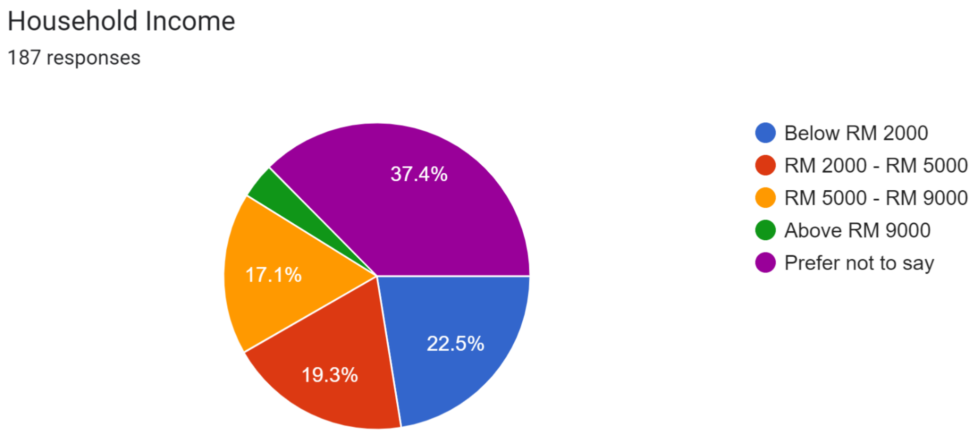


Figure 3.2.4.4: Pie Chart of Question 4 in Section A

For household income, 37.4% of respondents chose not to disclose. For the remaining, 22.5% of the respondents had a household income below RM 2000, 19.3% for household income of RM 2000 to RM 5000, 17.2% of RM 5000 to RM 9000 and a minority of them (3.7%) had an income of above RM 9000.

Section B: Understanding and Utilisation of Ingredient List

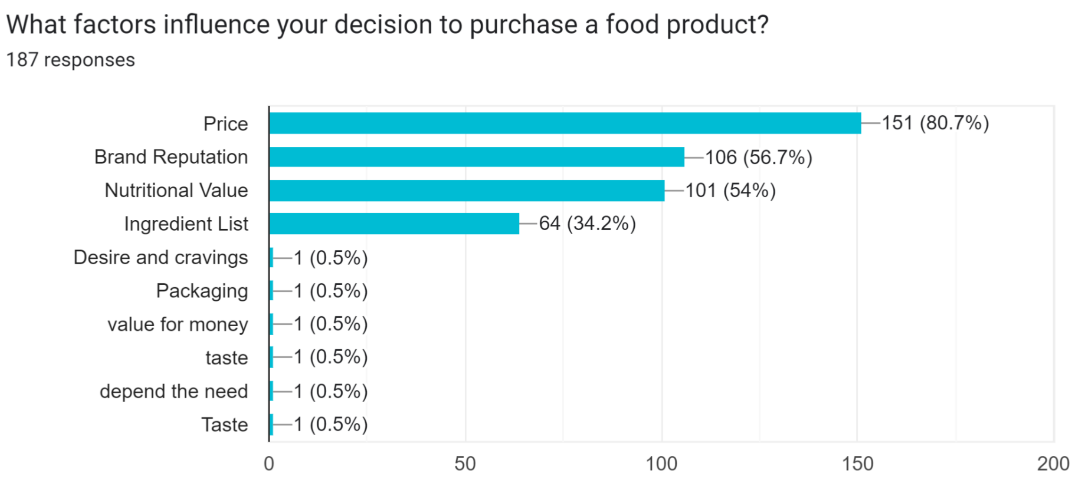


Figure 3.2.4.5: Bar Chart of Question 1 in Section B

According to the statistics from the figure, 80.7% of respondents indicate that price is the main factor when they consider purchasing food items. Apart from price, other factors such as brand reputation and nutritional value also significantly influence the public in their decision-making, as reported by 56.7% and 54% of the respondents respectively. 34.2% of respondents mentioned that they consider the ingredient list. Individual respondents have expressed their opinions on some other factors such as desire and craving, packaging, value for money, taste, need and necessity. This question aims to understand the factor that affects the public in making their decision to purchase food items. It is found that price is considered an important factor that the majority will consider.

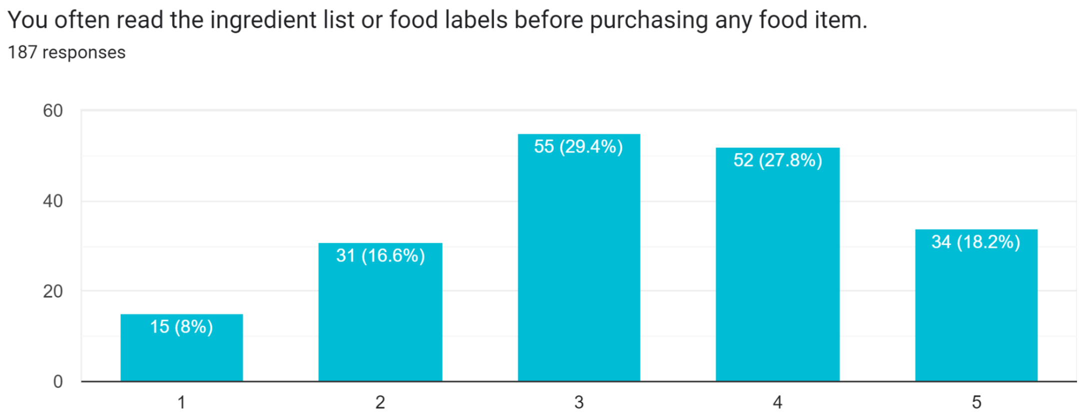


Figure 3.2.4.6: Bar Chart of Question 2 in Section B

The question aims to assess the public behaviour on reading the ingredient list before making a purchasing decision. Among all the respondents, 29.4% of the respondents expressed a neutral stance on this statement. However, the majority of the respondents exhibited the habit of reading ingredient lists, specifically 27.8% of respondents agree and 18.2% of respondents strongly agree with this statement.

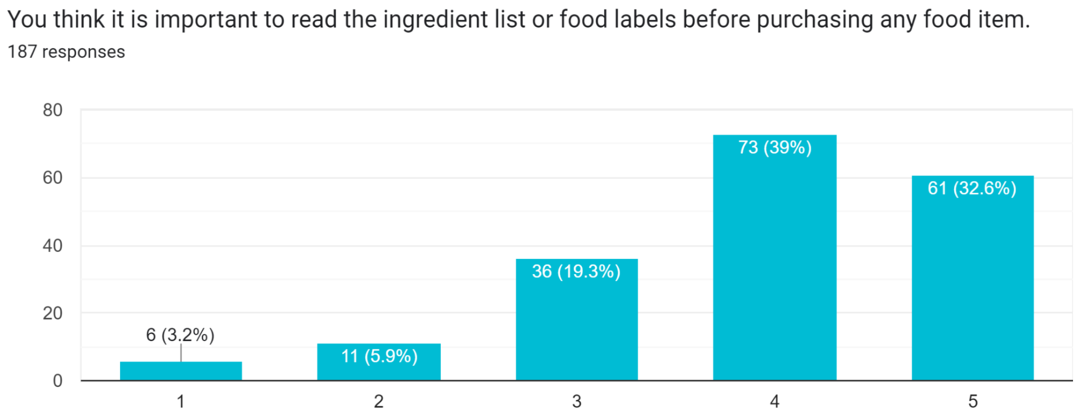


Figure 3.2.4.7: Bar Chart of Question 3 in Section B

In this question exploring the respondents' perception of the importance of reading the ingredient list before purchasing any food items, the results indicate that 39% of the respondents agreed and 32.6% of the respondents strongly agreed with the statement. This finding proves that the public is aware that the ingredient list conveys essential and valuable information that can be useful in preserving physical health.

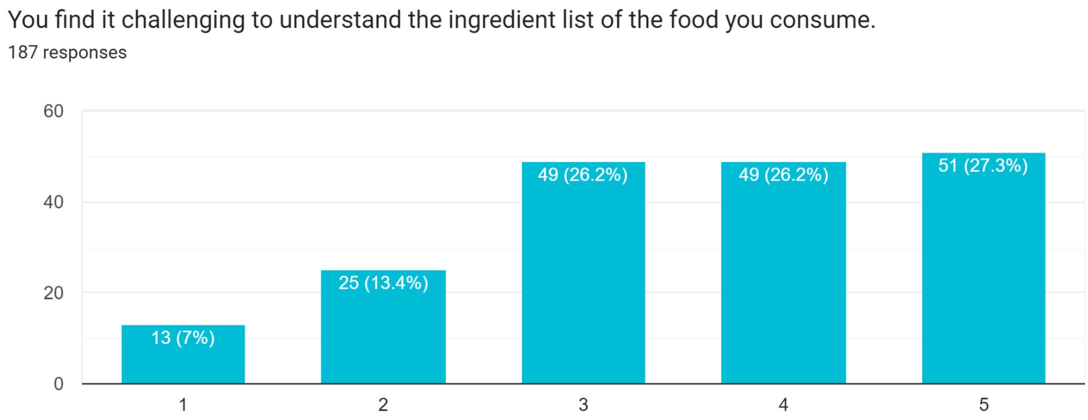


Figure 3.2.4.8: Bar Chart of Question 4 in Section B

The result of the question exploring the extent of difficulty in understanding the ingredient list reveals that 27.3% of the respondents strongly agree and 26.2% of respondents agreed that it is challenging to comprehend the ingredient list. The finding has shown that a considerable number of the public found it difficult to understand and obtain sufficient information that they required.

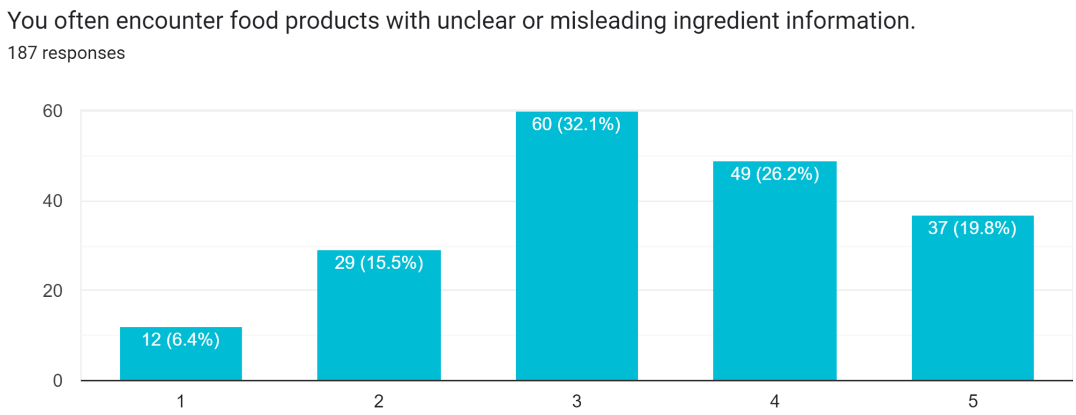


Figure 3.2.4.9: Bar Chart of Question 5 in Section B

The public's response to the issue of unclear and misleading ingredient information was predominantly neutral, with 32.1% of the respondents indicating this stance. The neutral response implies that the respondents may be unclear about the transparency of the ingredient details, unable to determine whether the ingredient list is reliable. However, the number of respondents who agreed with this statement is still higher than those who disagreed, indicating a portion of the public acknowledges the concerns surrounding the clarity of the ingredient list.

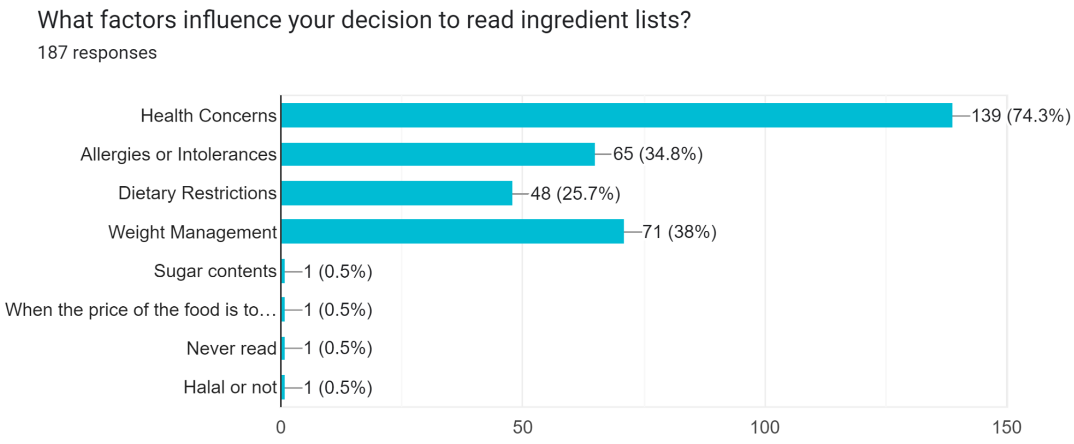


Figure 3.2.4.10: Bar Chart of Question 6 in Section B

In exploring the factors that influence the public's decision to read the ingredient list, it was found that health concerns are the primary driving factor, with 74.3% of the respondents indicating its importance. The respondents also respond to the importance of other factors such as weight management (38%), allergies or intolerances (34.8%) and dietary restrictions(25.7%). Furthermore, individual cases such as evaluating the sugar contents, halal status and instances where the price of a product seemed unexpectedly low were identified.

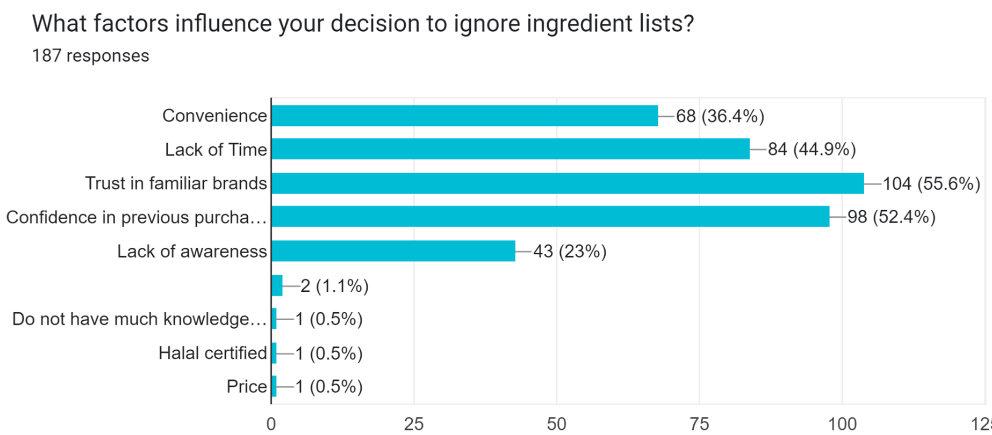


Figure 3.2.4.11: Bar Chart of Question 7 in Section B

In examining the factors that influence the public to ignore the ingredient lists, trust in familiar brands (55.6%) has significantly become the primary factor contributing to this behaviour. While 52.4% of the respondents expressed confidence in the previous purchases (52.4%), 44.9% of them think it is time-consuming to read. Other factors include inconvenience (36.4%) and awareness lacking (23%).

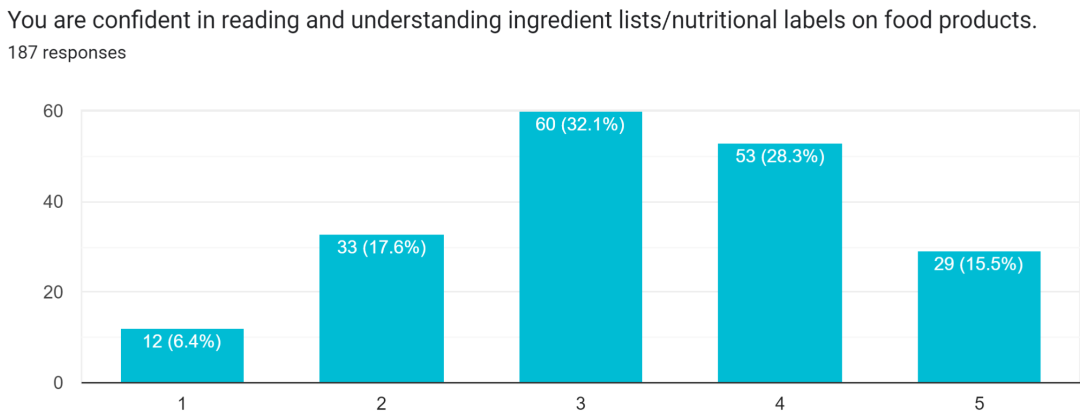


Figure 3.2.4.12: Bar Chart of Question 8 in Section B

The public shows a majority of neutral responses (32.1%) in self-confidence to read and understand the ingredient list or nutritional labels on the food products. Following this, 28.3% of the respondents agreed that they possess the ability to comprehend such information, while 17.6% disagreed. Nevertheless, there are quite a few respondents who expressed their confidence (15.5%) to understand and interpret the information conveyed.

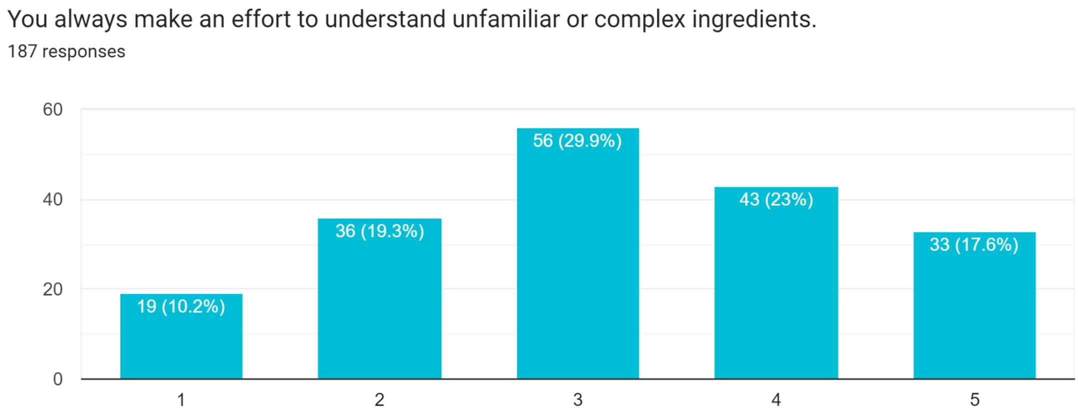


Figure 3.2.4.13: Bar Chart of Question 9 in Section B

In terms of making an effort to understand unfamiliar or complex ingredients, the responses collected fall primarily into three categories: neutral (29.9%), agree (23%), and disagree (19.3%). The statistics that focus on the neutral stance indicate that the public may not have a strong inclination to investigate effort. However, a minority of the respondents (17.6%) clarified that they are willing to put in the effort to learn and gain relevant knowledge.

Figure 3.2.4.14: Bar Chart of Question 10 in Section B

When understanding the willingness of the public to seek advice to understand food ingredients, most of the respondents expressed a neutral opinion. However, there is a larger number of respondents indicating that they will not seek professional advice compared to those who are willing to do so. This has shown a neglecting attitude among the public when encountering and dealing with unfamiliar or unrecognised food ingredients.

Section C: Awareness of the impact of their dietary choices on their health

Figure 3.2.4.15: Bar Chart of Question 1 in Section C

The public's response indicates that a significant majority, 39% of them agree and 20.9% of them strongly agree that they notice changing patterns in their health when they have different dietary habits. This finding suggests that the majority of the public is conscious and concerned about the potential impact of their diet on their overall health and well-being.

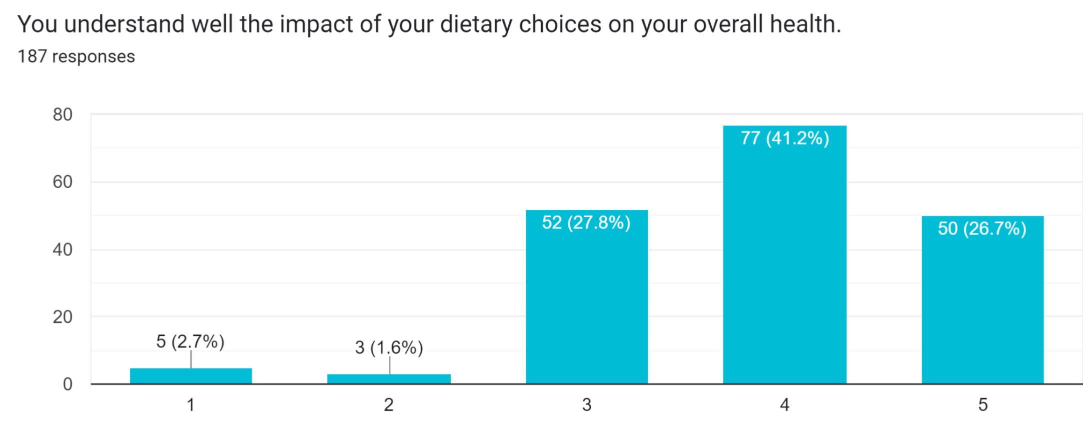


Figure 3.2.4.16: Bar Chart of Question 2 in Section C

Based on the outcome of this question, it is found that a number of the public demonstrates a good understanding of the impact of their dietary choices on their overall health. Specifically, 41.2% of the respondents agreed and 26.7% of them strongly agreed with this statement, indicating a high awareness among the public. However, about 27.8% of the respondents remain neutral in this statement, indicating that they may have limited knowledge about the possible impact or may not be aware of the potential impact

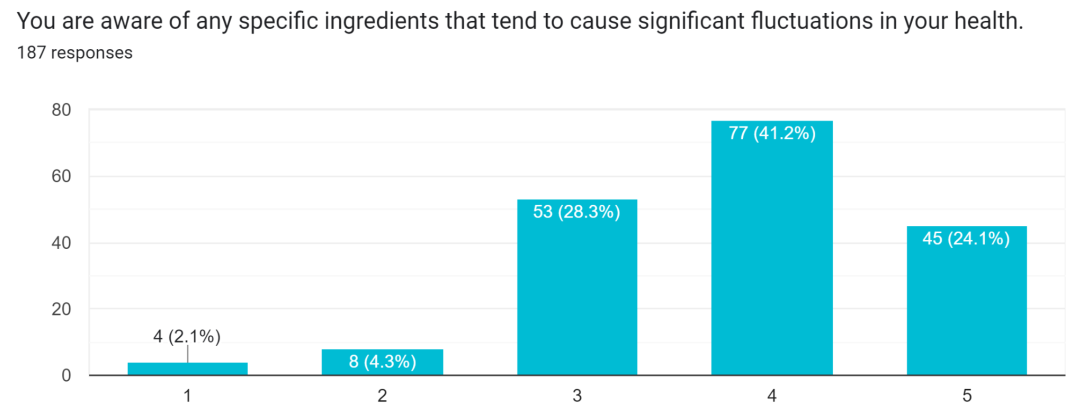


Figure 3.2.4.17: Bar Chart of Question 3 in Section C

The statistics from this question indicate a positive attitude among the public towards being aware of ingredients that may significantly impact their health. A total of 41.2% of respondents agreed with this statement, reflecting their recognition and concern regarding this issue. While 28.3% of respondents who showed a neutral attitude may potentially have low awareness of the potential health implications associated with the ingredients.

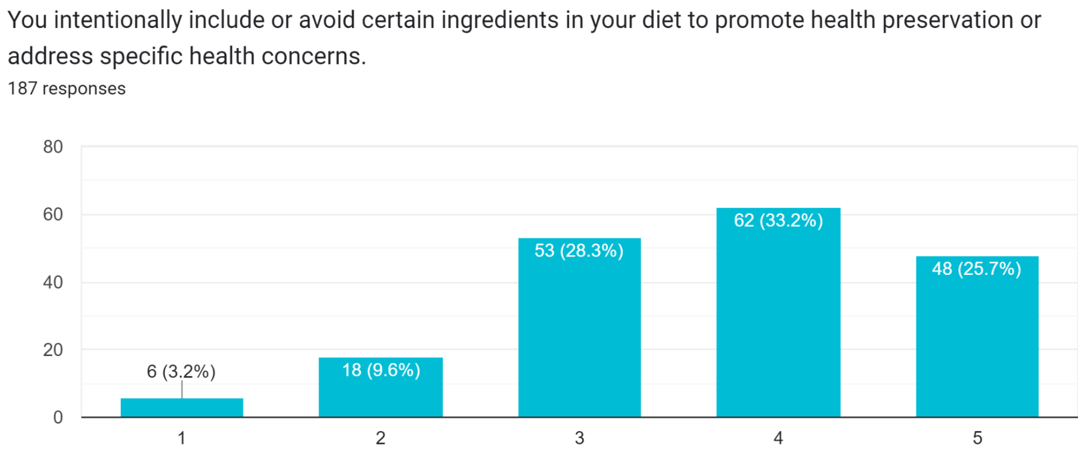


Figure 3.2.4.18: Bar Chart of Question 4 in Section C

33.2% of the respondents from the public agreed, and 25.7% of the respondents strongly agreed that they will intentionally include or avoid certain ingredients to promote health preservations on this statement. This indicates a high level of awareness of the respondents in considering ingredients that may bring health concerns and the willingness to take actions based on their awareness.

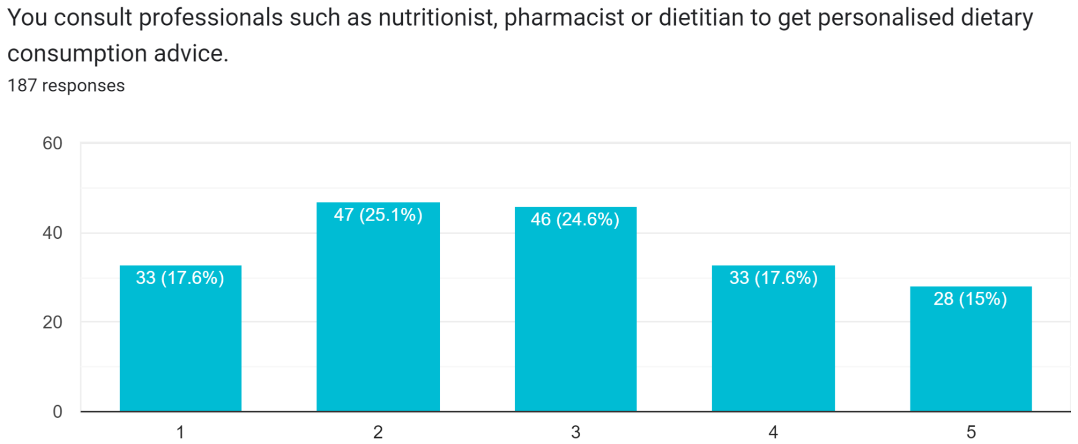


Figure 3.2.4.19: Bar Chart of Question 5 in Section C

This question evaluates the willingness of the public to consult the professional in order to get professional advice and personalise their consumption for better health preservation. From the result, the majority of the public is unwilling to pursue this action. This reluctance may be due to inconvenience, time consumption, and a perception of it being unworthy.

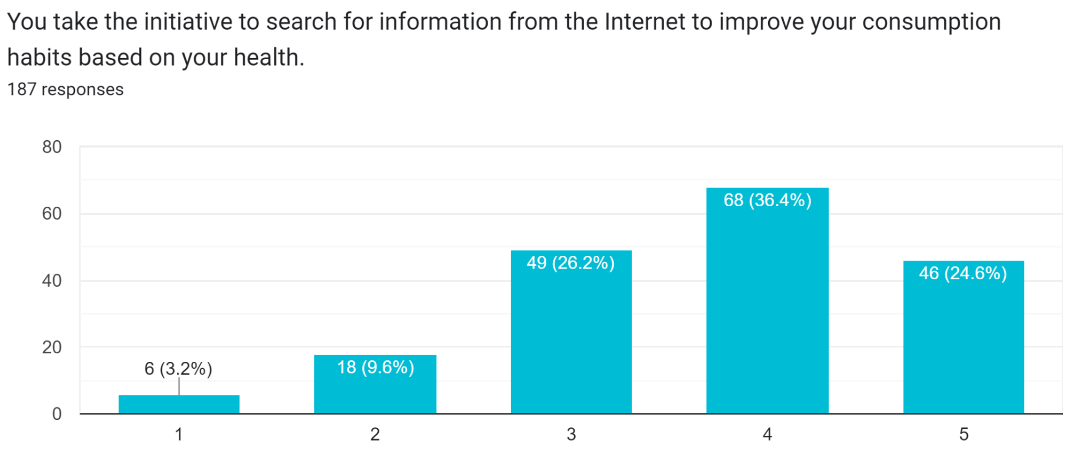


Figure 3.2.4.20: Bar Chart of Question 6 in Section C

When it comes to making efforts to improve consumption habits by searching for information from the Internet, the public shows a positive attitude with 36.4% of them agreeing with this statement and 24.6% of them strongly agreeing to indicate their willingness to utilise online resources for obtaining information. Compared to the previous questions that assessed the willingness to consult professionals, the public appears to show greater attentiveness to carrying out self-directed information searching.

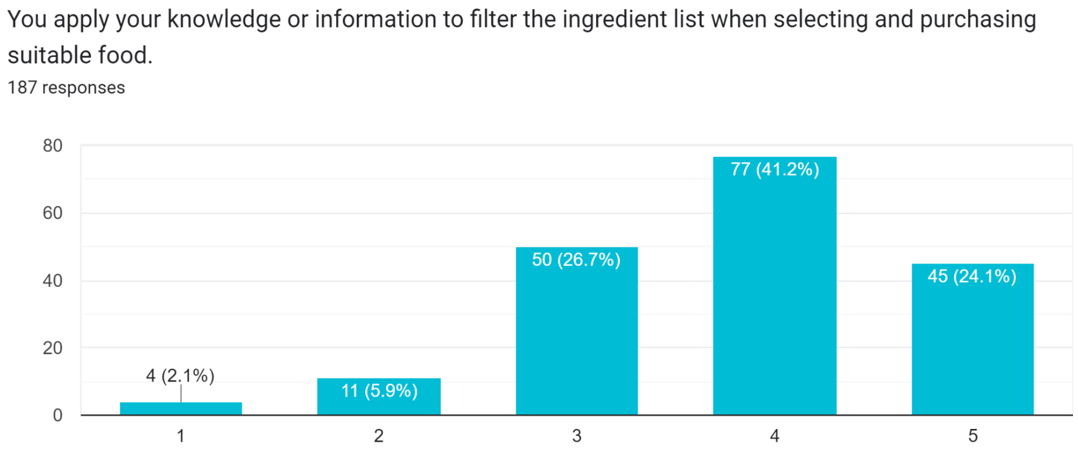


Figure 3.2.4.21: Bar Chart of Question 7 in Section C

A total of 41.2% of the respondents indicate that they will apply knowledge or information to filter ingredient lists when purchasing food. This positive attitude has reflected the commitment of the public to improve their dietary habits for a healthier lifestyle.

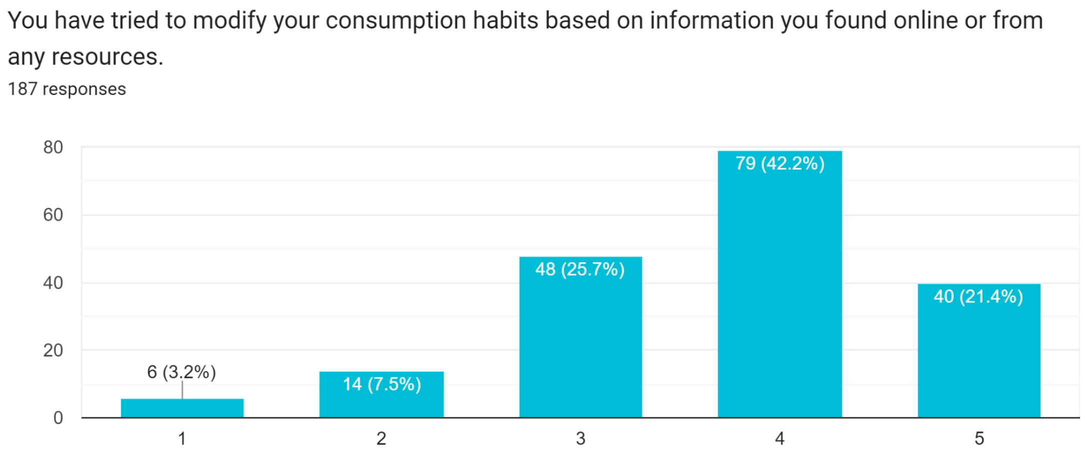


Figure 3.2.4.22: Bar Chart of Question 8 in Section C

The public maintains a positive attitude towards modifying their consumption habits based on the information they obtain. The survey results indicate that 42.2% of the respondents agreed to take modification actions, while an additional 21.4% strongly agreed. However, 25.7% of the respondents remain neutral on this statement, indicating a lack of motivation to carry out any actions.

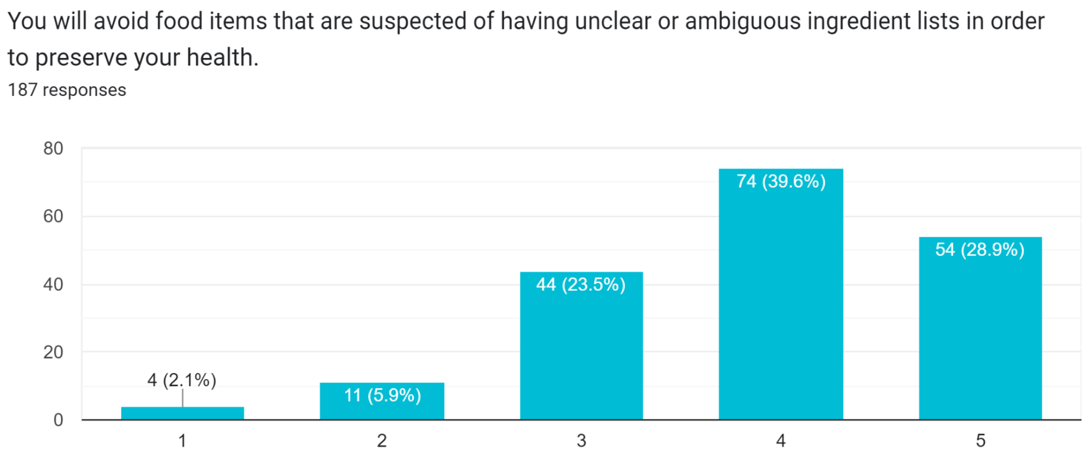


Figure 3.2.4.23: Bar Chart of Question 9 in Section C

When the ingredient list is suspected to be unclear or ambiguous, the majority of the respondents (39.6%) indicate that they would choose to avoid that particular food item and 28.9% of the respondents express a strong willingness to avoid those foods to preserve their health.

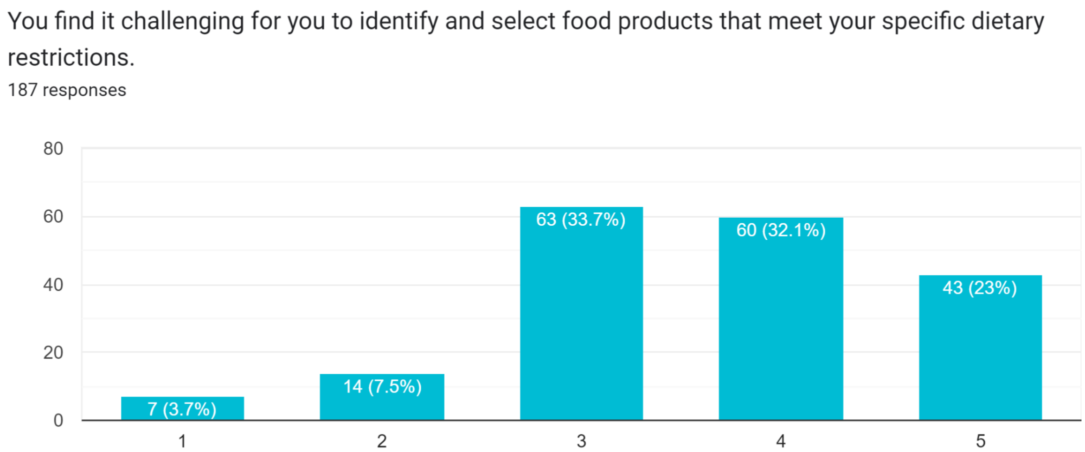


Figure 3.2.4.24: Bar Chart of Question 10 in Section C

In identifying the food products that meet the dietary restriction, the majority of respondents (33.7%) exhibit a neutral attitude. This neutrality may be because many individuals have received a list of consumable foods from their healthcare providers, which assists them in effectively filtering ingredients. However, comparing the remaining responses, it is found that most of the respondents find it challenging to identify and select suitable food items that can meet their dietary restrictions.

Section D: The likelihood of using an ingredient scanning feature

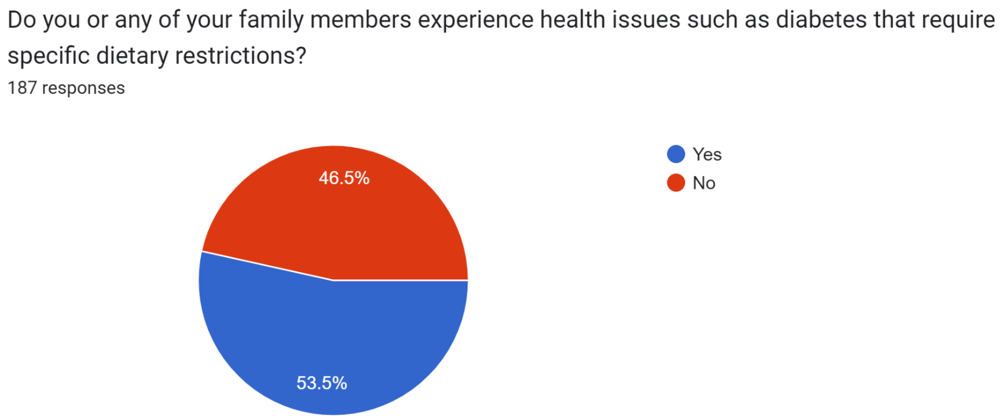


Figure 3.2.4.25: Bar Chart of Question 1 in Section D

This question aims to assess the experience and understanding of the public regarding dietary restrictions. From the responses, 53.5% of respondents declared that they or their family members have encountered health issues that necessitate specific dietary restrictions.

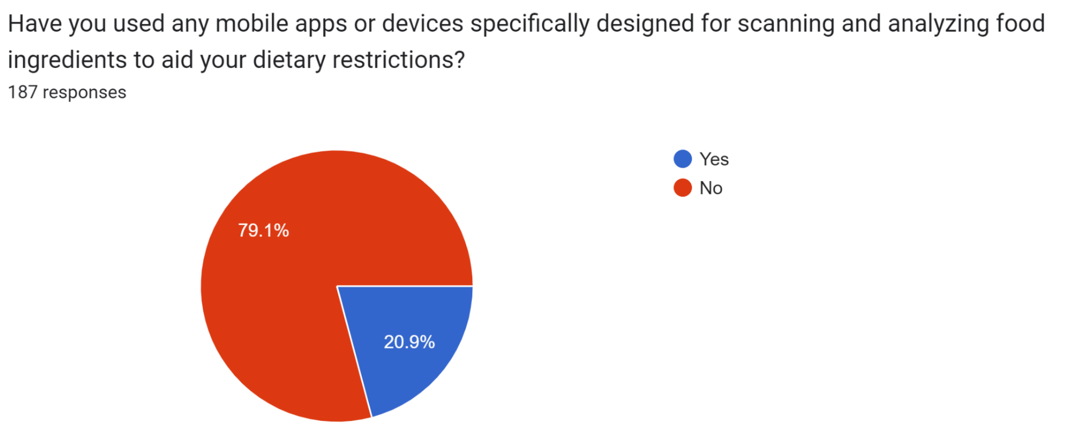


Figure 3.2.4.26: Bar Chart of Question 2 in Section D

The question aimed to assess the public's experience in using an ingredient scanning or analysing system that provides them with relevant information. As a result, only a small portion of the public (20.9%) have used such a system before.

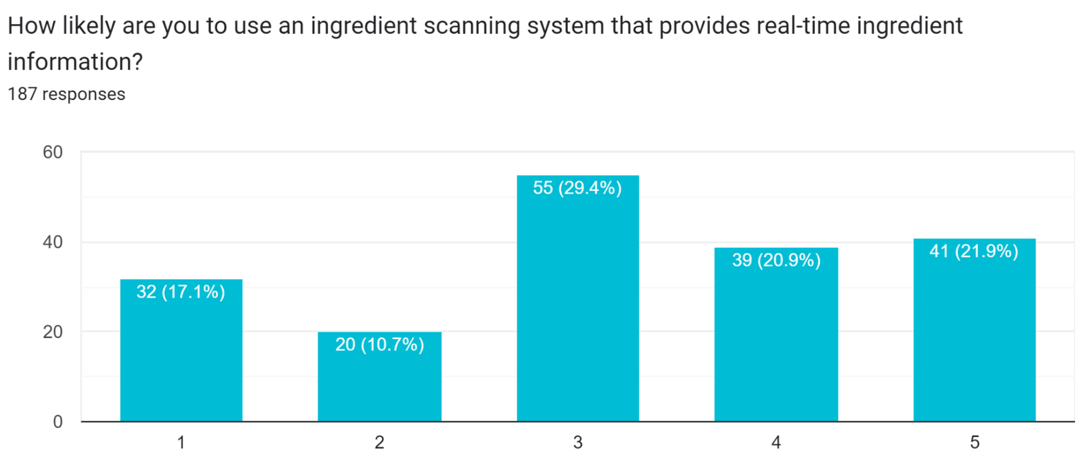


Figure 3.2.4.27: Bar Chart of Question 3 in Section D

This question evaluates the willingness of the public to accept and incorporate an ingredient scanning system into their daily routine, providing them with real-time ingredient information. The responses from the public (21.9%) express a strong willingness to use such a system.

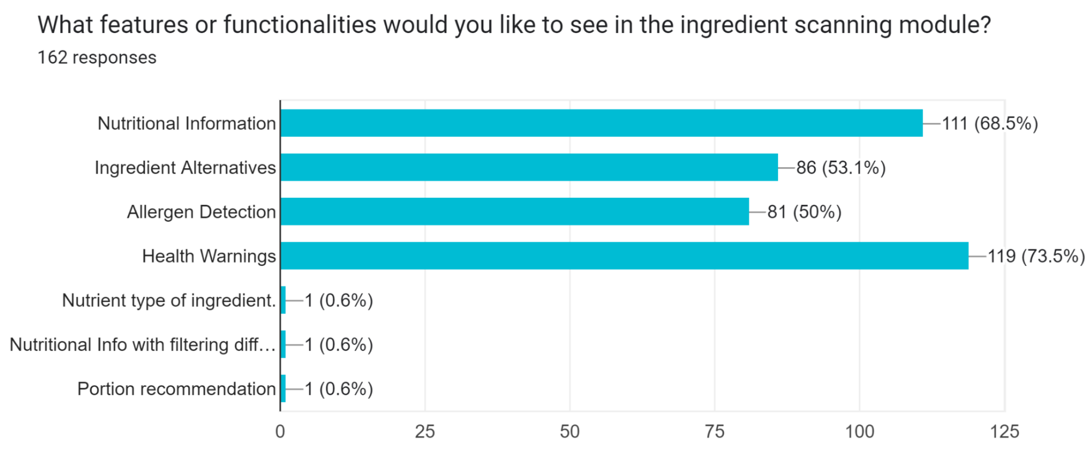


Figure 3.2.4.28: Bar Chart of Question 4 in Section D

This question aims to understand the expectations of the users in the implementation of an ingredient scanning feature. As a result, the popular options are health warnings (73.5%), followed by nutritional information (68.5%), ingredient alternatives (53.1%) and allergen detection (50%). Some other suggestions from the respondents are portion recommendations and nutrient type.

Conclusion of Questionnaire

After conducting the questionnaire that evaluates the awareness of the public on the impact of dietary habits on overall health, it is evident that the public possesses a certain level of awareness regarding the importance of dietary habits in maintaining good health. Furthermore, the public is willing to put in effort to modify and improve their dietary habits, preserving good health. However, the public finds it challenging to understand the ingredient list and figure out the compatibility with their dietary restrictions based on the information conveyed through the ingredient list. As a result, respondents express a willingness to use an ingredient scanning system that can provide them with sufficient information to help them achieve their health objectives. They anticipate features such as ingredient alternatives, allergen detections, nutritional information converting and health warnings to be implemented within the feature.

3.2.5 Conclusion of Research

In conclusion, the interview and survey results demonstrate a positive attitude towards the implementation of a diabetes monitoring system with an ingredient scanning feature. The interview has shown a positive attitude from the healthcare provider and diabetes individual regarding the implementation of the system. With the system, the healthcare provider can carry out a more efficient monitoring routine on the diabetes individuals. They also agree that this feature can enhance the management of diabetes by providing better dietary filtering options. This empowers diabetes individuals to have more food choices and reduces their reliance on frequent doctor consultations. Furthermore, the survey conducted through questionnaires indicates that the public has a high awareness of the importance of dietary habits. The public is willing to utilise the information on the ingredient list if the difficulties in understanding can be tackled.

3.3 Requirement Analysis

3.3.1 Project Scope

The proposed diabetes monitoring system comprises six modules, including the login module, patient module, doctor module, appointment module, blood glucose monitoring module and ingredient scanning module. Each module is designed to provide comprehensive support and enhance the performance of the system in various ways. Below are some descriptions of the functions and features provided by each module.

Login Module

This module allows the users to authenticate their identity and access the system. It consists of an interface that allows the user to log in by entering their credentials. This module also includes a password reset functionality. This allows the user to reset their password by requesting an email with an embedded password reset link to recover their forgotten password.

Doctor Module

The doctor module is designed to support the management tasks and responsibilities of the doctor. It includes the activities that allow adding new doctor records into the system. Not only that, it enables the doctor themselves to view or edit their records. They can proceed with doctor deactivation when the doctor is no longer working in the centre and carry out necessary updates on the charge duties.

Patient module

This module focuses on the management of patients. It allows healthcare providers to register patients, create profiles, assign doctors to patients, etc. It centralises the information of the patients together with their health records for better management. This allows healthcare providers to access the information conveniently, leading to better decision-making.

Appointment Module

This module streamlines the appointment scheduling process. It allows individuals to check available slots and make appointments. The module updates healthcare providers' schedules once appointments are reserved.

Blood Glucose Monitoring Module

The blood glucose monitoring module enables users to record their daily blood glucose levels. It visualised the stored glucose readings over time, allowing users to review and analyse the patterns. The data can be assessed by healthcare providers to facilitate remote monitoring. This module generates reports that summarise the condition of the patients to help the healthcare provider have better insights.

Ingredient Scanning Module

This module allows the users to capture the ingredient list on the food packaging to retrieve related information. It processes the image to analyse nutritional content, including glycemic-relevant information. The user can view the scanned results that will highlight potential restricted items within the ingredient list to make healthier food choices.

3.3.2 Development Environment

Hardware

In this project, the hardware utilised includes a laptop and a smartphone. The laptop serves as the primary development platform for the project. It is used to develop this system and carry out the necessary coding and debugging process, allowing the developer to identify and resolve issues or bugs that exist within the system. The smartphone will be utilised mainly to test the functionality of the ingredient scanning module. It captures the ingredient list on the food packaging to test if the textual content can be extracted from the images accurately. This will be achieved by utilising the camera equipped with the smartphone.

Programming Languages

The proposed system of this project will involve several programming languages. The selected languages include the Dart language of the Flutter framework and Python. Dart language will be used to develop the system, mainly the web application and the mobile application while Python will be used to develop the model for the ingredient scanning feature.

Database

Database service from Firebase will be utilised as the Backend Server for this proposed system. As a Backend-as-a-Service (BaaS) platform, Firebase offers developers a range of tools and services that simplify the development process. It is categorised as a NoSQL database program, which stores data in JSON-like documents. Firebase is selected as the database to store the data of the proposed system due to its ability to support multiple platforms, including mobile and web.

Development Tools

Table 3.3.2.1: List of Development Tools

Web & Mobile Application	• Android Studio • Google Colab • Jupyter Notebook

Open Source Software	• TensorFlow • Tesseract
Subscription Software	• Firebase
Documentation Software	• Google Docs • Google Slide • Google Sheet • Microsoft Office Word 2019 • Mendeley
Web Browser	• Google Chrome • Microsoft Edge

Machine Learning Library

The proposed software will leverage two main libraries, Tesseract and TensorFlow for classification tasks. Tesseract, on the other hand, is utilised as a popular optical character recognition (OCR) engine to extract textual content from the processed images. By integrating Tesseract, the software will be able to efficiently perform OCR and extract relevant text information from images. In addition to Tesseract, the proposed system also incorporates TensorFlow. By integrating TensorFlow, the proposed system will be able to access the pre-trained models and tools provided to train and evaluate the classification models. As the trained model will be utilised in the mobile platform, TensorFlow will be utilised to export the trained model.

3.3.3 Operation Environment

The proposed diabetes monitoring system is optimised for web and mobile applications that have browser capability and internet connectivity. Internet connectivity is crucial to establishing a connection between the proposed system and the Firebase system, enabling the system to access the services provided and ensuring the tasks performed on different platforms are updated to the Firebase server, synchronising across devices. To effectively utilise the Firebase system and its services, a stable Internet connection is necessary. To support the operation of the proposed system, hosting on Firebase is required. With Firebase Hosting, the web and mobile applications of the system can be effectively hosted, ensuring reliable availability and efficient delivery of the system's functionality. Firebase provides hosting with different services and pricing as stated in the table below.

Table 3.3.3.1: Comparison of Firebase Hosting Plan

Services	**Spark Plan (Free)**	**Blaze Plan**
Authentication		
Phone Auth - All regions	10 SMS sent/day	Billed per SMS sent (Based on region)
Monthly active users	50k/month	No-cost up to 50k MAUs Then Google Cloud pricing
Monthly active users - SAML/OIDC	50/month	No-cost up to 50 MAUs Then Google Cloud pricing
Cloud Firestore		
Stored Data	1 GiB total	No-cost up to 1 GiB total Then Google Cloud pricing
Network Egress	10 GiB/month	No-cost up to 10 GiB/month Then Google Cloud pricing
Document Writes	20K writes/day	No-cost up to 20K writes/day Then Google Cloud pricing
Document Reads	50K reads/day	No-cost up to 50K reads/day Then Google Cloud pricing
Document Deletes	20K deletes/day	No-cost up to 20K deletes/day Then Google Cloud pricing
Cloud Functions		
Invocations	Not applicable	No-cost up to 2M/month Then $0.40/million
GB-seconds	Not applicable	No-cost up to 400K/month Then Google Cloud pricing

CPU-seconds	Not applicable	No-cost up to 200K/month Then Google Cloud pricing
Outbound networking	Not applicable	No-cost up to 5GB/month Then $0.12/GB
Outbound networking	Not applicable	No-cost up to 120 min/day Then $0.003/min
Container storage	Not applicable	No-cost up to 500MB of storage Then $0.10/GB/month
Cloud Storage		
GB stored	5 GB	$0.026/GB
GB downloaded	1 GB/day	$0.12/GB
Upload operations	20K/day	$0.05/10k
Download operations	50K/day	$0.004/10k
Multiple buckets per project	No	Yes
Hosting		
Storage	10 GB	$0.026/GB
Data transfer	360 MB/day	$0.15/GB
Custom domain & SSL	Yes	Yes
Multiple sites per project	Yes	Yes
Firebase ML		
Custom Model Deployment	Yes	Yes
Cloud Vision APIs	No	$1.50/K
Realtime Database		
Simultaneous connections	100	200k/database
GB stored	1 GB	$5/GB
GB downloaded	10 GB/month	$1/GB
Multiple databases per project	No	Yes

Test Lab		
Virtual Device Tests	10 tests/day	No-cost up to 60 min/day Then $1/device/hour
Physical Device Tests	5 tests/day	No-cost up to 30 min/day Then $5/device/hour

$1 = 4.59 MYR (USD/Myr Currency Exchange Rate & News, 2023)

In this project, the proposed system will utilise the Spark plan instead of the Blaze plan. While the Blaze plan offers improved services and a larger storage capacity, the capacity provided by the Spark plan is deemed sufficient to support the project's requirements. The Spark plan offers most of the services and features available in Firebase mostly free, albeit with certain quotas. However, the allocated quota is satisfied enough for the proposed system to carry out the operations efficiently. As the proposed system does not require advanced features to support the system, opting Spark plan will be sufficient (Firebase pricing, 2023).

Minimum Computer Hardware Requirement

Table 3.3.3.2: Criteria of Minimum Computer Hardware Requirement

Processor	2.00GHz or faster processor. Core i5 or above.
Memory	8.0GB RAM minimum, higher RAM is recommended
Storage	Minimum 128 GB and up to 1 TB available spaces.
Additional Requirement	Stable Internet connection

The table above shows the minimum computer hardware requirement to operate the system. High-performance processors are required to expedite tasks and boost the overall operating process. The requirement of the system in terms of memory should be at least 8.0GB RAM to handle larger datasets and ensure smooth execution. The device should fulfil a minimum of 500GB of storage for fast reading and writing speed, improving the speed of the execution. Lastly, a stable Internet connection by the devices is necessary for downloading software libraries, updates, documentation, and accessing online resources.

Minimum Computer Software Requirement

Table 3.3.3.3: Criteria of Minimum Computer Software Requirement

Operating System	Windows 11
Database	Google Firebase
Hosting Subscription	Spark Plan (Free)
Development Tool	● Android Studio ● Jupyter Notebook ● Google Colab ● Flutter

The minimum computer software requirements are stated in the table above. The latest available operating system version, Windows 11 is selected, leveraging the latest features and compatibility to support the operating environment. The database service from Google Firebase will be employed to handle data storage and retrieval. For system hosting, the spark plan is selected as the free plan is sufficient to support the system operation. The development tools used are Android Studio and Jupyter Notebook. Flutter allows the development of both web and mobile applications using a single codebase while Jupyter Notebook provides an interactive environment to run Python code, especially for machine learning tasks.

Minimum Mobile Requirement

Table 3.3.3.4: Criteria of Minimum Mobile Requirement

Operating System	Android
Memory	2.0 GB RAM minimum, higher RAM is recommended
Storage	16 GB storage minimum, higher storage is recommended
Display	Minimum resolution of 720p (1280x720)
Additional Requirement	Equipped with a functioning camera

The Flutter-based system requires an Android operating system for mobile device compatibility. A minimum of 2 GB is required to carry out the functionalities of the system and higher RAM is recommended for better performance. To ensure adequate storage, the mobile device should have a minimum storage capacity of 16 GB. To provide a better user experience when interacting with the system interface, the mobile device should have a minimum resolution of 720p (1280x720). Lastly, a functioning camera is necessary on the mobile device to support image capturing within the system.

3.3.4 Use Case Diagram

3.3.4.1 Overview Use Case

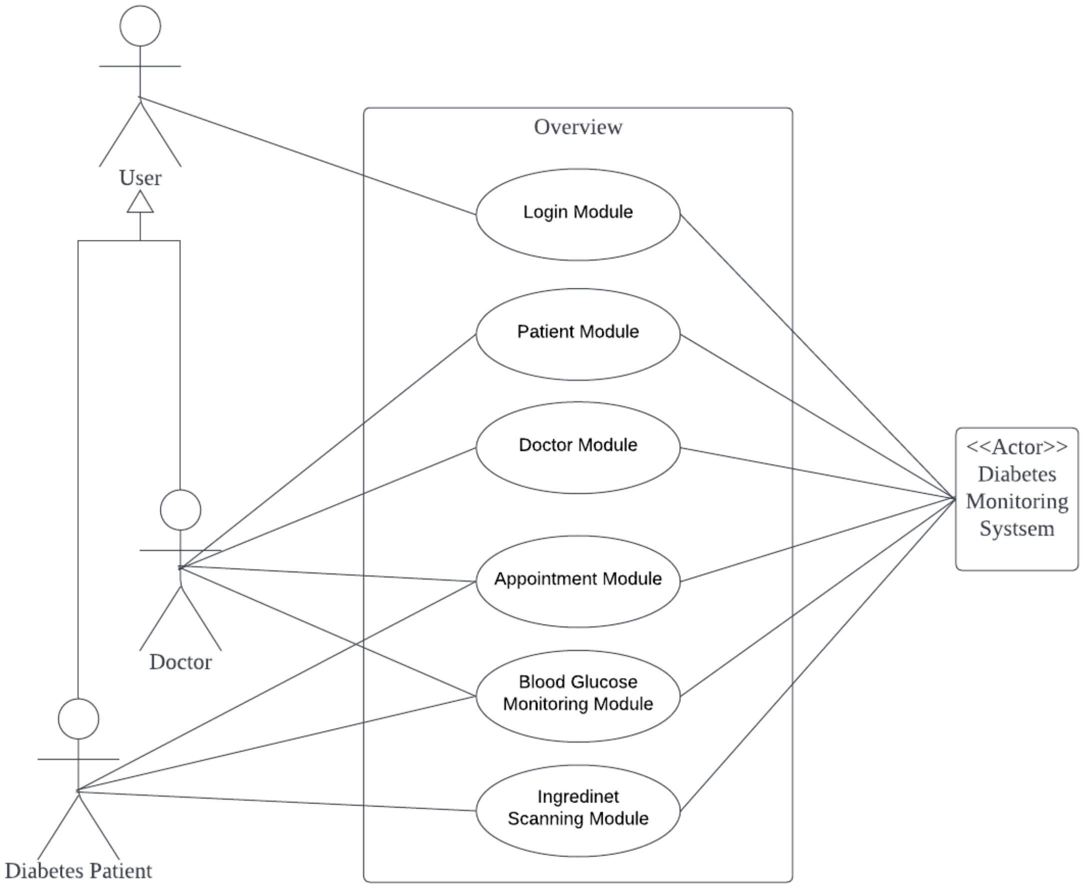


Figure 3.3.4.1.1: Overview of Use Case in Diabetes Monitoring System

3.3.4.2 Login Module Use Case

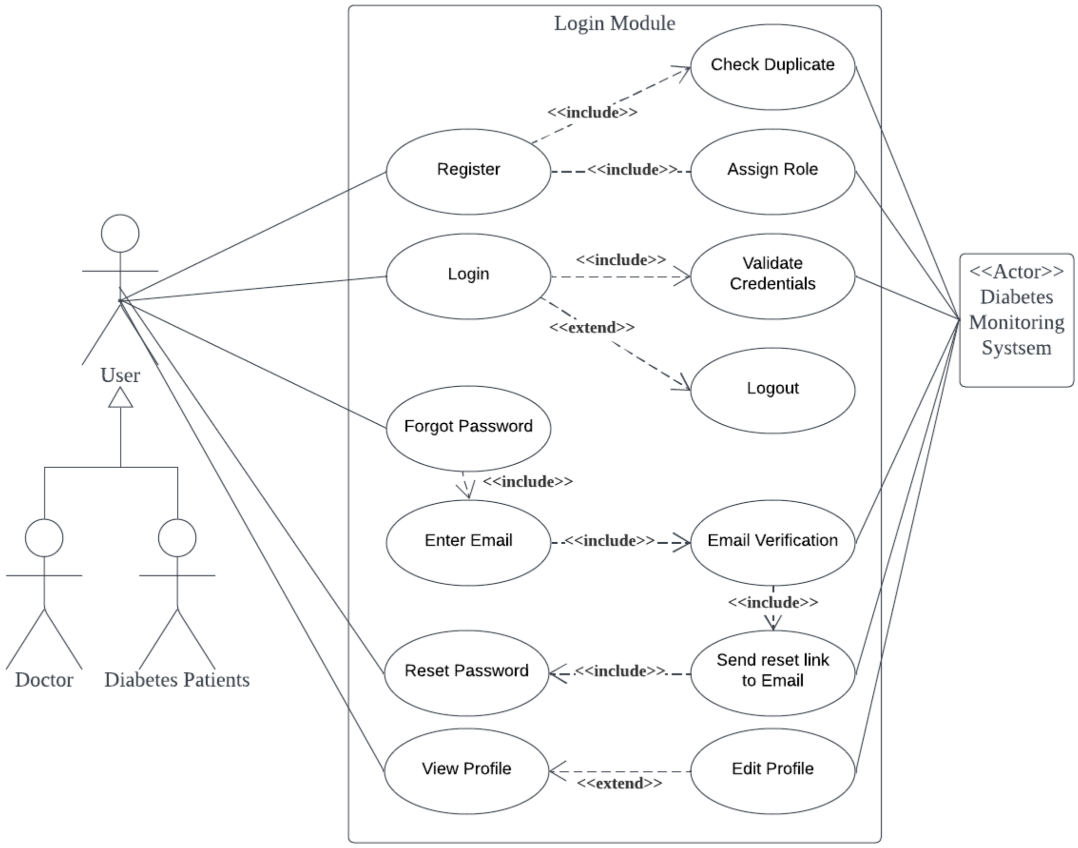


Figure 3.3.4.2.1: Use Case of Login Module

Table 3.3.4.2.1: Use Case Description of Login

Name of Use Case: Login	
Brief Description: The use case describes the process of the user login into the system.	
Actors: Patients, Doctors	
Pre-Condition: 1. The user has registered and owns an account. 2. The user has a pair of valid username and password for login.	
Actor Action	**System Response**
	1. The system presents the login page.

2. The user fills in their credentials. 3. The users press login.	4. The system validates and verifies the username and password entered. 5. The user is directed to the system homepage.
6. The user clicks on the logout button to log out.	
Alternative Flow: **A1:** Step 4 If the entered credential is incorrect, the system will display an error message to ask the user to reenter their credentials. **A2:** Step 4 If the entered credential is incorrect and the user has forgotten their password, they can opt to forget the password. (Use Case: Forgot Password)	
Post Condition: The user has logged into their account successfully.	

Table 3.3.4.2.2: Use Case Description of Forgot Password

Name of Use Case: Forgot Password	
Brief Description: The use case describes the process of the user recovering their forgotten password.	
Actors: Patients, Doctors	
Pre-Condition: 1. The user has registered and owns an account.	
Actor Action	**System Response**
1. The user clicks on the forgot password option.	2. The system prompts the user to enter their registered email for recovery.
3. The user enters their email and submits the email address.	4. The system verifies the entered email, ensuring the address matches with the database. 5. The system sends a password reset email for a password reset.
6. The user clicks the password reset link provided in their email.	7. The system verifies the validity of the link.

	8. The system displays a password reset page.
9. The users enter their new password.	10. The system updates the new passwords.
Alternative Flow: **A1:** Step 4 If the entered email is invalid or unregistered, the system will display an error message to the user, requesting them to re-enter. **A2:** Step 7 If the reset password link has expired, the system will display an error message and return to step 2 for the user to request a new recovery link.	
Post Condition: The user has reset their password successfully and the new password updated in the system.	

Table 3.3.4.2.3: Use Case Description of View Profile

Name of Use Case: View Profile	
Brief Description: The use case describes the process of viewing the profile information of the logged-in account.	
Actors: Patient**,** Doctor	
Pre-Condition: 1. The user has logged into the system.	
Actor Action	**System Response**
1. The user navigates to the profile management page.	2. The system retrieves the user profile information. 3. The system displays the user profile information.
4. The user reviews the displayed profile information.	
Alternative Flow:	
Post Condition: The user successfully retrieves and views their profile information.	

Table 3.3.4.2.4: Use Case Description of Edit Profile

<table>
<tr><td colspan="2">Name of Use Case: Edit Profile</td></tr>
<tr><td colspan="2">Brief Description: The use case describes the process of editing the profile information of the logged-in account.</td></tr>
<tr><td colspan="2">Actors: Patient, Doctor</td></tr>
<tr><td colspan="2">Pre-Condition:
1. The user has logged into the system.</td></tr>
<tr><td>Actor Action</td><td>System Response</td></tr>
<tr><td>1. The user navigates to the profile management page.</td><td>2. The system retrieves the user profile information.
3. The system displays the user profile information.</td></tr>
<tr><td>4. The user selects the option Edit Profile.</td><td>5. The system activates the edit mode.</td></tr>
<tr><td>6. The user modifies the profile details.
7. Once complete modification, the user submits the updated data.</td><td>8. The system prompts the user for confirmation.</td></tr>
<tr><td>9. The user confirms the edit actions.</td><td>10. The system validates the entered information to ensure the inputs meet the required format.
11. The system updates the profile information based on the modified data
12. The system displays a modification successful message to inform the action completed.</td></tr>
<tr><td colspan="2">Alternative Flow:
A1: Step 6
If the user would like to cancel the activated edit mode, they can click on the “Cancel” button.

A2: Step 9
If the user refuses to continue the add patient actions, may click on the ‘No’ button to abort the current action.

A3: Step 10
If the entered information is invalid, the system will highlight the input with an error and display an error message to get the error input corrected.</td></tr>
</table>

Post Condition: The user successfully edits their profile information.

Table 3.3.4.2.5: Use Case Description of Register

Name of Use Case: Register	
Brief Description: The use case describes the process of the patients registering themselves as a new user of the system.	
Actors: Patient	
Pre-Condition: 1. The user does not register an account in the system.	
Actor Action	**System Response**
1. The user navigates to the login page. 2. The user clicks on the “Guest Register”.	3. The system prompts the user with the registration form.
4. The user fills in all necessary details to create a new doctor. 5. The user submits the form with completed profile details.	6. The system validates the entered information to ensure the inputs meet the required format. 7. The system prompts the user with a doctor list.
8. The user selects the doctor desired. 9. The user initiates the register function by clicking on the “Register” button.	10. The system prompts the user for confirmation.
11. The user confirms the registered actions.	12. The system adds the user into the system. 13. The system displays a successful message indicating the action completed
Alternative Flow: **A1:** Step 6 If the entered information is invalid, the system will highlight and focus the input with error and display an error message to get the error input corrected. **A2:** Step 11 If the user refuses to continue the add doctor actions, may click on the ‘No’ button to abort the current action.	
Post Condition: The user successfully registers into the system.	

3.3.4.3 Doctor Module Use Case

Figure 3.3.4.3.1: Use Case of Doctor Module

Table 3.3.4.3.1: Use Case Description of Add Doctor

Name of Use Case: Add Doctor	
Brief Description: The use case describes the process of adding a new doctor.	
Actors: Doctor	
Pre-Condition: 1. The doctor has not been registered in the system.	
Actor Action	**System Response**

1. The user initiates the operation of adding a new doctor into the system.	2. The system displays a new doctor-adding form.
3. The user fills in all necessary details to create a new doctor. 4. The user submits the form with completed profile details.	5. The system validates the entered information to ensure the inputs meet the required format. 6. The system prompts the user with a doctor's image uploading page.
7. The user uploads the doctor's profile image. 8. The user initiates the add doctor function by clicking on the "Add New Doctor" button.	9. The system prompts the user for confirmation.
10. The user confirms the add actions.	11. The system checks whether the database consists of similar records. 12. The system adds the doctor into the system. 13. The system displays a successful message indicating the action completed
Alternative Flow: **A1:** Step 5 If the entered information is invalid, the system will highlight and focus the input with error and display an error message to get the error input corrected. **A2:** Step 7 If the user would like to upload the image later, after the doctor's creation, the user may directly proceed to Step 8. **A3:** Step 10 If the user refuses to continue the add doctor actions, may click on the 'No' button to abort the current action. **A4:** Step 11 If there is an identical find existing in the system, the system will prompt an error message.	
Post Condition: A new doctor profile is successfully created.	

Table 3.3.4.3.2: Use Case Description of View Doctor

Name of Use Case: View Doctor	
Brief Description: The use case describes the process of viewing the information of a doctor.	
Actors: Doctor	
Pre-Condition: 1. The doctor has been registered into the system.	
Actor Action	**System Response**
1. The user navigates to the doctor management page.	2. The system displays all doctors' information stored in the system.
3. The user searches for certain doctor information by inputting the search criteria.	4. The system filters the doctor list based on the entered search information. 5. The system displays a list of doctor records that match the search criteria.
6. The user clicks on the sorting criteria.	7. The system sorts the doctor list based on the selected sorting criteria.
8. The user selects the desired record from the list.	9. The system displays the selected doctor's information.
10. The user clicks on the profile tab to view the respective details of the doctor.	11. The system displays the profile details of the doctor.
Alternative Flow: **A1:** Step 5 The entered information for searching does not yield any matching records, the system will display a message indicating no results found. The doctor may verify the entered information and perform another search attempt. **A2:** Step 10 The user clicks on the appointment tab and the appointment under this doctor will be displayed. **A3:** Step 10 The user clicks on the patient list tab and the patients in charge by this doctor will be displayed.	

Post Condition: The user successfully retrieves and views the patient's information.

Table 3.3.4.3.3: Use Case Description of Edit Doctor

Name of Use Case: Edit Doctor	
Brief Description: The use case describes the process of editing the doctor's information to update their details such as profile.	
Actors: Doctor	
Pre-Condition: 1. The doctor's information exists in the system.	
Actor Action	**System Response**
1. The user navigates to the doctor management page.	2. The system displays all doctors' information in a list.
3. The user selects the records to edit.	4. The system displays the selected doctor's information.
5. The user selects the Edit option.	6. The system activates the edit mode.
7. The user modifies the details of the doctor. 8. Once complete modification, the user submits the updated data.	9. The system prompts the user for confirmation.
10. The user confirms the update actions.	11. The system validates the entered information to ensure the inputs meet the required format. 12. The system updates the doctor's information based on the modified data. 13. The system displays a modification successful message to inform the action completed.
Alternative Flow: **A1:** Step 7 If the user desires to exit the edit mode, may click on the "Cancel" button and the system will prompt the user back to view the doctor page. **A2:** Step 10 If the user refuses to continue the add doctor actions, may click on the 'No' button	

to abort the current action. **A3:** Step 11 If the entered information is invalid, the system will highlight and focus the input with an error and display an error message to get the error input corrected.
Post Condition: The doctor's information is successfully modified and updated.

Table 3.3.4.3.4: Use Case Description of Deactivate Doctor

Name of Use Case: Deactivate Doctor	
Brief Description: The use case describes the process of deactivating the doctors who quit the healthcare centre.	
Actors: Doctor	
Pre-Condition: 1. The doctor's information exists in the system. 2. The doctor has quit the healthcare centre.	
Actor Action	**System Response**
1. The user navigates to the doctor management page.	2. The system displays all doctors' information in a list.
3. The user selects the doctor to deactivate.	4. The system displays the selected doctor's information.
5. The user selects the deactivate option for the doctor.	6. The system prompts the user for confirmation.
7. The user confirms the deactivation.	8. The system updates the doctor's status. 9. The system displays a deactivation successful message to inform the action completed.
Alternative Flow: **A1:** Step 7 If the user refuses to continue the deactivate doctor actions, may click on the 'No' button to abort the current action.	
Post Condition: The doctor's information is successfully modified and updated.	

3.3.4.4 Patient Module Use Case

Figure 3.3.4.4.1: Use Case of Patient Module

Table 3.3.4.4.1: Use Case Description of Register Patient

Name of Use Case: Register Patient
Brief Description: The use case describes the process of registering a new patient, recording and assigning the patient a doctor.
Actors: Doctor
Pre-Condition: 1. The patient has not been registered in the system.

<table>
<tr><th>Actor Action</th><th>System Response</th></tr>
<tr><td>1. The user initiates the operation of adding a new patient into the system.</td><td>2. The system displays a new patient-adding form.</td></tr>
<tr><td>3. The user fills in all necessary details to create a new patient.
4. The user submits the form with completed profile details.</td><td>5. The system validates the entered information to ensure the inputs meet the required format.
6. The system prompts the user with a doctor’s image uploading page.</td></tr>
<tr><td>7. The user selects a doctor for the patient.
8. The user proceeds with the patient registration to the next steps.</td><td>9. The system validates the doctors selected.
10. The system prompts the user with a patient image uploading page.</td></tr>
<tr><td>11. The user uploads the patient’s profile image.
12. The user initiates the add patient function by clicking on the “Add New Patient” button.</td><td>13. The system prompts the user for confirmation.</td></tr>
<tr><td>14. The user confirms the add actions.</td><td>15. The system checks whether the database consists of similar records.
16. The system adds the patient into the system.
17. The system displays a successful message indicating the action completed</td></tr>
<tr><td colspan="2">Alternative Flow:
A1: Step 5
If the entered information is invalid, the system will highlight and focus the input with error and display an error message to get the error input corrected.

A2: Step 9
If the user does not select any doctor to be assigned to the patient, an error message will be prompted to notify the doctor selection.

A3: Step 11
If the user would like to upload the image later, after the doctor's creation, the user may directly proceed to Step 12.</td></tr>
</table>

A4: Step 15 If there is an identical find existing in the system, the system will prompt an error message. **A4:** Step 14 If the user refuses to continue the add patient actions, may click on the ‘No’ button to abort the current action.
Post Condition: The patient's profile is successfully registered and created.

Table 3.3.4.4.2: Use Case Description of View Patient

<table>
<tr><td colspan="2">Name of Use Case: View Patient</td></tr>
<tr><td colspan="2">Brief Description: The use case describes the process of viewing the information of a patient.</td></tr>
<tr><td colspan="2">Actors: Doctor</td></tr>
<tr><td colspan="2">Pre-Condition:
1. The patient has a record in the system.</td></tr>
<tr><td>Actor Action</td><td>System Response</td></tr>
<tr><td>1. The user navigates to the patient management page.</td><td>2. The system displays all patients’ information stored in a list.</td></tr>
<tr><td>3. The user searches for certain patient information by inputting the search criteria.</td><td>4. The system filters the patient list based on the entered search information.
5. The system displays a list of patient records that match the search criteria.</td></tr>
<tr><td>6. The user clicks on the sorting criteria.</td><td>7. The system sorts the patient list based on the selected sorting criteria.</td></tr>
<tr><td>8. The user selects the desired record from the list.</td><td>9. The system displays the selected patient’s information.</td></tr>
<tr><td>10. The user clicks on the profile tab to view the respective details of the patient.</td><td>11. The system displays the profile details of the patient.</td></tr>
<tr><td colspan="2">Alternative Flow:
A1: Step 5</td></tr>
</table>

The entered information for searching does not yield any matching records, the system will display a message indicating no results found. The doctor may verify the entered information and perform another search attempt. **A2:** Step 10 The user clicks on the blood glucose record tab and the blood glucose record of the patient will be displayed. **A3:** Step 10 The user clicks on the patient medical record tab and the necessary record list will be displayed.
Post Condition: The user successfully retrieves and views the patient's information.

Table 3.3.4.4.3: Use Case Description of Edit Patient

<table>
<tr><td colspan="2">Name of Use Case: Edit Patient</td></tr>
<tr><td colspan="2">Brief Description: The use case describes the process of editing the patient's information to update their details such as their profile.</td></tr>
<tr><td colspan="2">Actors: Doctor</td></tr>
<tr><td colspan="2">Pre-Condition:
1. The user is authorised to access the patient's information.
2. The patient's information exists in the system.</td></tr>
<tr><td>Actor Action</td><td>System Response</td></tr>
<tr><td>1. The user navigates to the patient management page.</td><td>2. The system displays all patients' information in a list.</td></tr>
<tr><td>3. The user selects the records to edit.</td><td>4. The system displays the selected patient information.</td></tr>
<tr><td>5. The user selects the Edit option.</td><td>6. The system activates the edit mode.</td></tr>
<tr><td>7. The user modifies the details of the patients.
8. Once complete modification, the user submits the updated data.</td><td>9. The system prompts the user for confirmation.</td></tr>
<tr><td>10. The user confirms the update actions.</td><td>11. The system validates the entered information to ensure the inputs meet the required format.</td></tr>
</table>

	12. The system updates the patient's record with the modified data 13. The system displays a modification successful message to inform the action completed.
Alternative Flow: **A1:** Step 7 If the user desires to exit the edit mode, may click on the "Cancel" button and the system will prompt the user back to view the patient page. **A2:** Step 10 If the user refuses to continue the edit patient actions, may click on the 'No' button to abort the current action. **A3:** Step 11 If the entered information is invalid, the system will highlight and focus the input with an error and display an error message to get the error input corrected.	
Post Condition: The patient's profile is successfully modified and updated.	

Table 3.3.4.4.4: Use Case Description of Change Doctor

Name of Use Case: Change Doctor	
Brief Description: The use case describes the process of changing a patient's assigned doctor.	
Actors: Doctor	
Pre-Condition: 1. The patient's profile exists in the system. 2. The patient has been assigned to a doctor.	
Actor Action	**System Response**
1. The user navigates to the page that displays that particular patient profile.	2. The system retrieves and displays the patient's information.
3. The user selects the Edit option.	4. The system activates the edit mode.
5. The user selects the doctor text field that displays the	6. The system prompts the user with a list of doctors currently available in the system.

information of the doctor in charge.	
7. The user selected the doctor to be assigned.	8. The system displays the selected doctor's ID in the text field, replacing the previous ID.
9. Once complete modification, the user submits the updated data.	10. Once complete modification, the system prompts the user for confirmation.
11. The user confirms the update actions.	12. The system validates the entered information to ensure the inputs meet the required format. 13. The system updates the patient's record with the modified data 14. The system displays a modification successful message to inform the action completed.
Alternative Flow: **A1:** Step 5 If the user desires to exit the edit mode, may click on the "Cancel" button and the system will prompt the user back to view the patient page. **A2:** Step 11 If the user refuses to continue the edit patient's assigned doctor actions, may click on the 'No' button to abort the current action. **A3:** Step 12 If the entered information is invalid, the system will highlight and focus the input with an error and display an error message to get the error input corrected.	
Post Condition: The assigned doctor for that particular patient is changed and updated in the system.	

Table 3.3.4.4.5: Use Case Description of Deactivate Patient

Name of Use Case: Deactivate Patient
Brief Description: The use case describes the process of deactivating the patient.
Actors: Doctor
Pre-Condition:

1. The doctor's information exists in the system. 2. The doctor has quit the healthcare centre.	
Actor Action	**System Response**
1. The user navigates to the patient management page.	2. The system displays all patient information in a list.
3. The user selects the patient to deactivate.	4. The system displays the selected patient's information.
5. The user selects the deactivate option for the patient.	6. The system prompts the user for confirmation.
7. The user confirms the deactivation.	8. The system updates the patient's status. 9. The system displays a deactivation successful message to inform the action completed.
Alternative Flow: **A1:** Step 7 If the user refuses to continue the deactivated patient actions, may click on the 'No' button to abort the current action.	
Post Condition: The patient's information is successfully modified and updated.	

3.3.4.5 Blood Glucose Monitoring Module Use Case

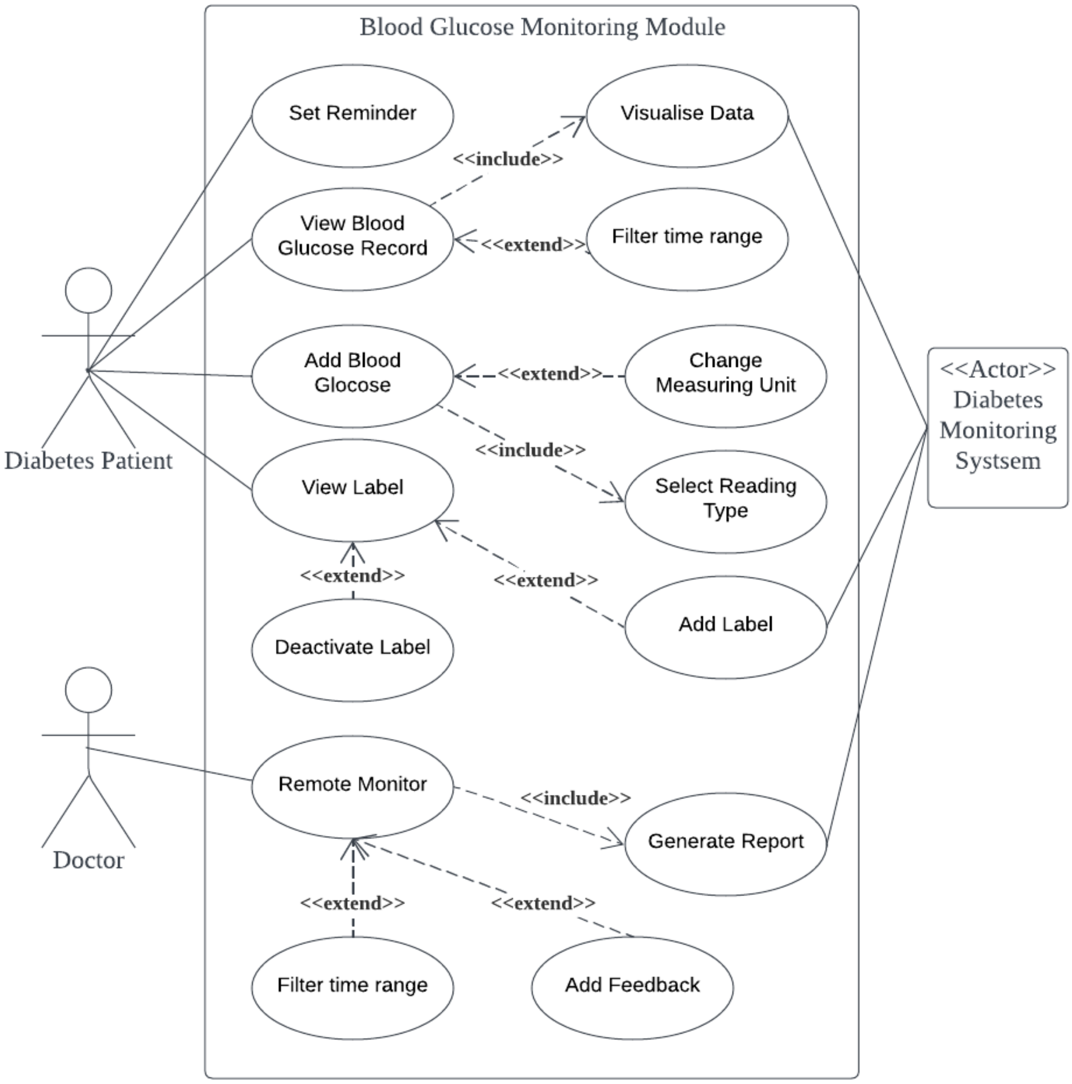


Figure 3.3.4.5.1: Use Case of Blood Glucose Monitoring Module

Table 3.3.4.5.1: Use Case Description of Record Blood Glucose

Name of Use Case: Record Blood Glucose
Brief Description: The use case describes the process of adding new blood glucose records by patients, including information such as date time, value, measurement units and so on.
Actors: Diabetes Patients
Pre-Condition:

Actor Action	System Response
1. The user navigates to the blood glucose management page and selects the option to record blood glucose.	2. The system presents the interface for blood glucose reading type.
3. The user selects the type.	4. The system presents the interface for blood glucose recording.
5. The user fills in the necessary information, including the blood glucose level, and the date and time of the reading.	6. The system prompts the user for confirmation. 7. The system validates the entered data to ensure appropriate recording.
8. The user confirms the add record actions.	9. The system validates the entered information to ensure the inputs meet the required format. 10. The system adds the details as a new blood glucose record. 11. The system displays a successful message to inform the action completed
Alternative Flow: **A1:** Step 2 When the user selects the type “fasting”, a confirmation will be prompted to ensure the user has fasted for at least 8 hours. **A2:** Step 5 While the default measurement unit of the blood glucose record is ‘mm/dL’, the user may change the unit by clicking on the change unit icon. **A3:** Step 8 If the user refuses to continue the edit patient actions, may click on the ‘No’ button to abort the current action. **A4:** Step 9 If the entered information is invalid, the system will highlight and focus the input with error and display an error message to get the error input corrected.	
Post Condition: The blood glucose record is saved in the system and associated with the user's profile.	

Table 3.3.4.5.2: Use Case Description of Set Reminder

Name of Use Case: Set Reminder	
Brief Description: The use case describes the process of setting customisable reminders based on their requirement to generate reminders at specific times to perform necessary activities such as medication, blood glucose testing routine etc.	
Actors: Diabetes Individuals	
Pre-Condition:	
Actor Action	**System Response**
1. The user navigates to the reminder management page.	2. The system presents the interface for reminders.
3. The user selects the option to set reminders.	4. The system prompts a reminder setting form.
5. The user customises their reminder based on their requirement, and time intervals.	6. The system validates the entered data to ensure the reminder is set accurately. 7. The system will trigger the reminder based on the time setting.
8. The user receives the alert or notification to take necessary action.	
Alternative Flow: **A1:** Step 3 If the users select the option to deactivate a reminder, a confirmation message will be prompted. When the user confirms the deactivated actions, that particular reminder will be deactivated. **A2:** Step 6 If the entered information is invalid, the system will highlight and focus the input with error and display an error message to get the error input corrected.	
Post Condition: The reminder settings are saved in the system and associated with the user's profile to generate notifications at the specified time to remind the user of their routine.	

Table 3.3.4.5.3: Use Case Description of Remote Monitor

Name of Use Case: Remote Monitor

Brief Description: The use case describes the process of doctors remotely monitoring their patients' blood glucose levels.	
Actors: Doctor	
Pre-Condition: 1. The doctor is assigned to the patients.	
Actor Action	**System Response**
1. The user navigates to the patient management page.	2. The system presents the interface of the patient.
3. The user selects the remote monitor options.	4. The system displays the data of the patient's blood glucose levels through visual representations such as graphs or charts.
5. The user can filter and view specific timeframes or date ranges for blood glucose data.	6. The system presents the data of the patients within the timeframe specified.
7. The user may provide feedback according to the observed data.	8. The system presents the interface for adding feedback.
9. The user fills in the feedback and submits the feedback.	10. The system prompts the user for confirmation.
14. The user confirms the add actions.	15. The system adds the feedback into the system. 16. The system displays a successful message indicating the action completed
Alternative Flow: **A1:** Step 9 If the user thinks that the feedback is unnecessary, may stop at this step.	
Post Condition: The doctor can monitor the data of the desired patient remotely.	

Table 3.3.4.5.4: Use Case Description of View Label

Name of Use Case: View Label
Brief Description: The use case describes the process of managing the label such as viewing labels, adding a label and deactivating a label.
Actors: Diabetes Individuals

Pre-Condition:	
Actor Action	**System Response**
1. The user navigates to the label management page.	2. The system presents the interface for labels.
3. The user selects the option to add a new label.	4. The system prompts a label-adding form.
5. The user enters the information for the labels.	6. The system validates the entered data before adding the label. 7. The system displays a successful toast message.
Alternative Flow: **A1:** Step 3 If the users select the option to deactivate a label, a confirmation message will be prompted. When the user confirms the deactivated actions, that particular label will be deactivated. **A2:** Step 6 If the entered information is invalid, the system will highlight and focus the input with error and display an error message to get the error input corrected.	
Post Condition: The new label is saved into the system.	

3.3.4.6 Ingredient Scanning Module Use Case

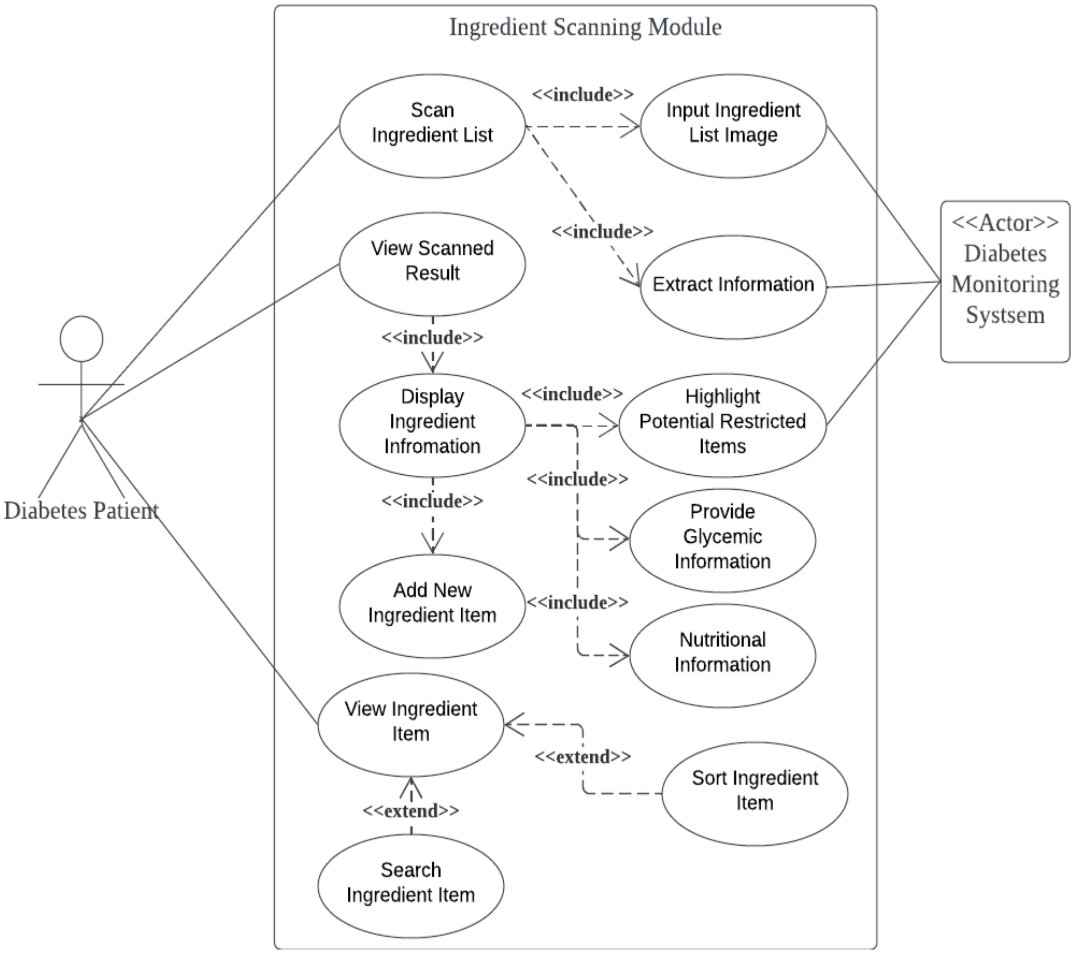


Figure 3.3.4.6.1: Use Case of Ingredient Scanning Module

Table 3.3.4.6.1: Use Case Description of Scan Ingredient List

Name of Use Case: Scan Ingredient List	
Brief Description: The use case describes the process that allows the diabetes patients to scan the ingredient list of a food item and provide necessary information to them.	
Actors: Diabetes patients	
Pre-Condition: 1. The user has a smartphone that is equipped with a camera to capture the image.	
Actor Action	**System Response**

1. The user selects the option to scan the ingredient list.	2. The system prompts the user with an interface to capture the ingredient list.
3. The user captures the images of the product's ingredient list via camera.	4. The system receives the input images from the patients. 5. The input image will be validated. 6. The system processes the image. 7. The textual content from the image will be extracted. 8. The extracted ingredient items will be displayed in a list of text fields.
9. The user confirms the extracted items.	10. The system will analyse and produce results based on the information.
11. The user can view the scanned result.	12. The result will be displayed. 13. The system will display the respective information and highlight the potentially restricted items to the user.
Alternative Flow: **A1:** Step 5 If the captured image is invalid, the system will display an error message to request the user capture another image. **A2:** Step 8 If the information extracted is invalid, the system will display an error message to request the user capture another image. **A3:** Step 9 The user can modify the extracted items if it is incorrect.	
Post Condition: The user can receive information regarding the ingredient list they captured.	

Table 3.3.4.6.2: Use Case Description of View Scanned Result

Name of Use Case: View Ingredient Item
Brief Description: The use case describes the process of obtaining the glycemic-related information of certain ingredient items.

Actors: Diabetes Individuals	
Pre-Condition:	
Actor Action	**System Response**
1. The user navigates to the ingredient item page.	2. The system displays all the ingredient items associated with their glycemic-related available information in the database.
3. The user clicks on the search button.	4. The system prompts a search box.
5. The user enters the search criteria in the search box.	6. The system displays the information based on the searched item.
Alternative Flow: **A1:** Step 6 If the searched items are not found, the system will display a not found message.	
Post Condition:	

3.3.4.7 Appointment Module Use Case

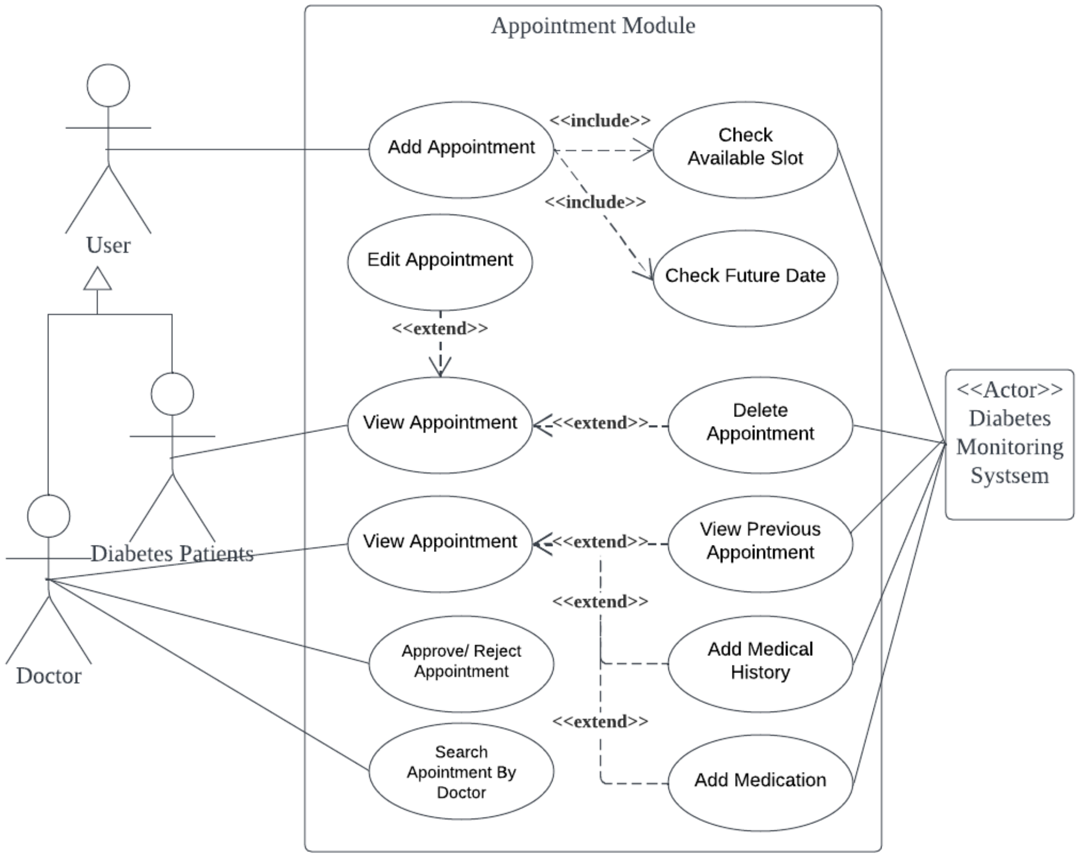


Figure 3.3.4.7.1: Use Case of Appointment Module

Table 3.3.4.7.1: Use Case Description of Add Appointment

Name of Use Case: Add Appointment	
Brief Description: The use case describes the process of making an appointment to schedule for consultation.	
Actors: Diabetes Patients, Doctor	
Pre-Condition: 1. The user has a registered account in the system. 2. The patient has been assigned to a doctor.	
Actor Action	**System Response**
1. The user navigates to the appointment management page and selects the option to schedule a new appointment.	2. The system presents the user with an add appointment interface.

3. The user fills in the necessary details. 4. The user submits the appointment application.	5. The system prompts the user for confirmation.
6. The user confirms the add actions.	7. The system validates the entered information. 8. The system checks there is no duplicate appointment on that particular slot 9. The system adds new appointments.
Alternative Flow: **A1:** Step 6 If the user refuses to continue the add appointment actions, may click on the 'No' button to abort the current action. **A2:** Step 7 If the entered information is invalid, the system will highlight and focus the input with error and display an error message to get the error input corrected. **A3:** Step 8 If the appointment slot has been occupied, the system will display an error message to the user.	
Post Condition: The appointment is successfully scheduled and confirmed.	

Table 3.3.4.7.2: Use Case Description of Cancel Appointment

Name of Use Case: Delete Appointment	
Brief Description: The use case describes the process of deleting an appointment scheduled.	
Actors: Diabetes Patients	
Pre-Condition: 1. The diabetes patient has one or more future appointments.	
Actor Action	**System Response**
1. The user navigates to the appointment management page and selects the scheduled appointment to delete.	2. The system presents the user with the details of that scheduled appointment

3. The user selects the option to delete the appointment.	4. The system prompts the user with a confirmation message, seeking their final confirmation on the cancellation.
5. The user confirms the cancellation.	6. The system removes the appointment from the user's appointment list. 7. The system updates the appointment status on the doctor's schedule.
Alternative Flow: **A1:** Step 5 The user decides not to cancel the appointment, the system will return the user to the appointment details page without cancelling the appointment.	
Post Condition: The scheduled appointment is cancelled and removed.	

Table 3.3.4.7.3: Use Case Description of View Appointment

Name of Use Case: View Appointment	
Brief Description: The use case describes the process of viewing their scheduled appointments and access to their appointment details.	
Actors: Diabetes Patients,	
Pre-Condition: 1. The patient has one or more scheduled appointments.	
Actor Action	**System Response**
1. The user navigates to the appointment management page.	2. The system presents the user with all of the scheduled appointments made.
3. The user searches for certain appointments by inputting the search criteria.	4. The system filters the appointment list based on the entered information. 5. The system displays a list of appointment records that match the search criteria.
6. The user selects the desired appointment record from the list.	7. The system displays the selected appointment and the respective details.

Alternative Flow: **A1:** Step 5 The entered information for searching does not yield any matching records, the system will display a message indicating no results found. The user may verify the entered information and perform another search attempt.
Post Condition: The patient successfully views their scheduled appointments.

Table 3.3.4.7.4: Use Case Description of Edit Appointment

<table>
<tr><td colspan="2">Name of Use Case: Edit Appointment</td></tr>
<tr><td colspan="2">Brief Description: The use case describes the process of editing and modifying the details of the scheduled appointments.</td></tr>
<tr><td colspan="2">Actors: Diabetes Patients</td></tr>
<tr><td colspan="2">Pre-Condition:
1. The diabetes patient has one or more future appointments.</td></tr>
<tr><td>Actor Action</td><td>System Response</td></tr>
<tr><td>1. The user navigates to the appointment management page.</td><td>2. The system presents the user with all of the scheduled appointments made by that user.</td></tr>
<tr><td>3. The user selects the scheduled appointment that they wish to modify.</td><td>4. The system displays the selected appointment and the respective details.</td></tr>
<tr><td>5. The user modifies the details for that appointment and submits the appointment again.</td><td>6. The system prompts the user for confirmation.</td></tr>
<tr><td>7. The user confirms the edit actions.</td><td>8. The system validates the entered information to ensure the inputs meet the required format.
9. The system updates the appointment details in the system.</td></tr>
<tr><td colspan="2">Alternative Flow:
A1: Step 3
The system shows an error message when the user attempts to perform editing on past appointment records.

A2: Step 7</td></tr>
</table>

If the user refuses to continue the edit patient actions, may click on the 'No' button to abort the current action. **A3:** Step 8 If the entered information is invalid, the system will highlight and focus the input with error and display an error message to get the error input corrected.
Post Condition: The patient successfully views their scheduled appointments.

Table 3.3.4.7.5: Use Case Description of Update Appointment

Name of Use Case: Update Appointment	
Brief Description: The use case describes the process of viewing the appointment and updating the appointment with necessary follow-up information.	
Actors: Doctor	
Pre-Condition: 1. The patient has one or more scheduled appointments.	
Actor Action	**System Response**
1. The user navigates to the appointment management page.	2. The system presents the user with all of the scheduled appointments.
3. The user selects the desired appointment record from the list.	4. The system displays the selected appointment and the respective details.
5. The user enters the medication and follow-up information. 6. The users submit the follow-up information.	7. The system prompts the user for confirmation.
8. The user confirms the update actions.	9. The system validates the entered information. 10. The system adds new medication and follow-up.
Alternative Flow: **A1:** Step 3 If the selected appointment is not approved, it will prompt an error message. **A2:** Step 8 If the user refuses to continue the update appointment actions, may click on the	

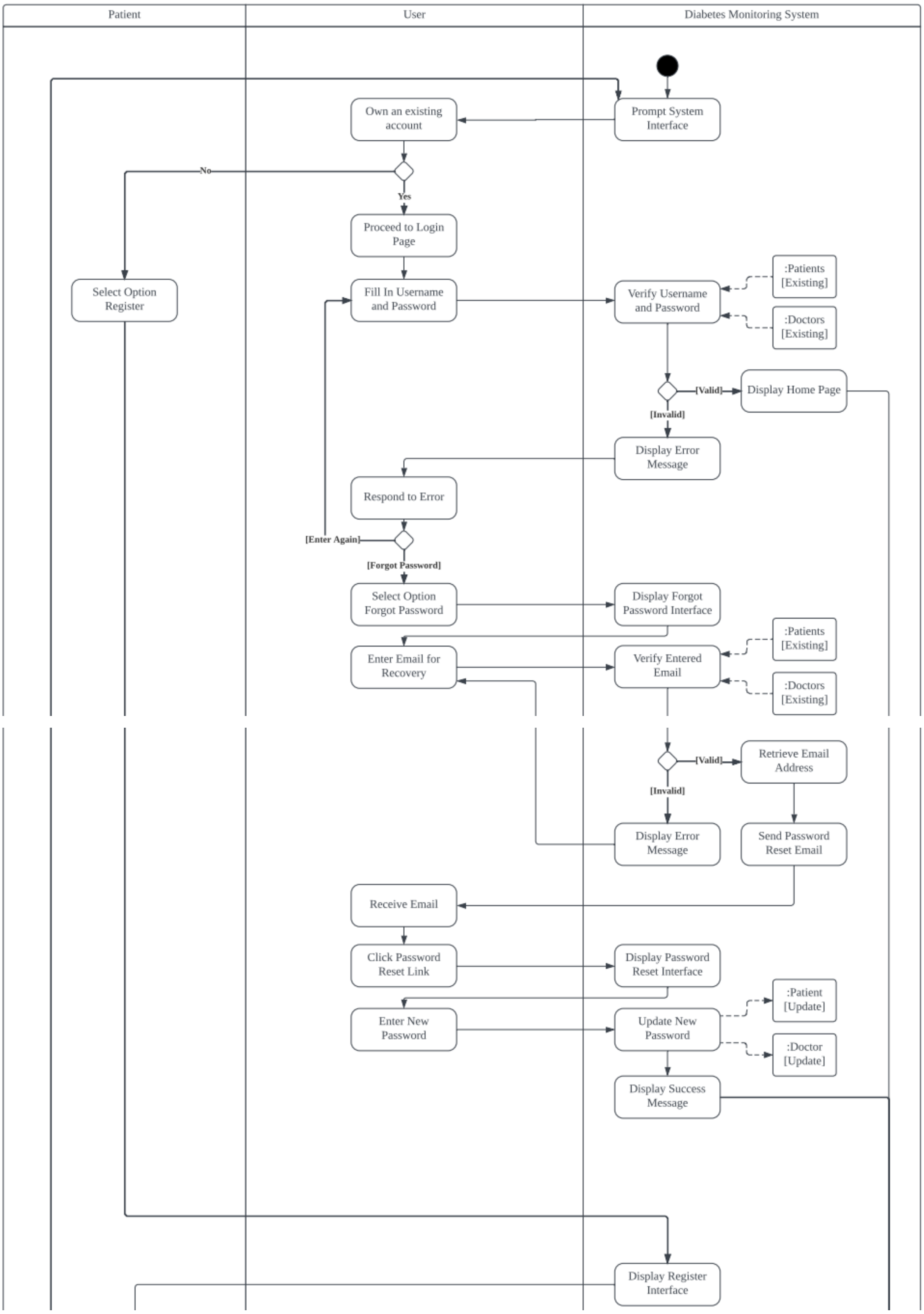
‘No’ button to abort the current action. **A3:** Step 9 If the entered information is invalid, the system will highlight and focus the input with error and display an error message to get the error input corrected.
Post Condition: The doctor successfully updates the scheduled appointment.

Table 3.3.4.7.6: Use Case Description of Approve Appointment

Name of Use Case: Approve Appointment	
Brief Description: The use case describes the process of approving a pending appointment.	
Actors: Doctor	
Pre-Condition: 1. The diabetes patient has one or more future appointments.	
Actor Action	**System Response**
1. The user navigates to the appointment management page.	2. The system presents the user with the list of scheduled appointments
3. The user selects the option to approve the appointment.	4. The system prompts the user with a confirmation message, seeking their final confirmation on the approval.
5. The user confirms the approval.	6. The system updates the appointment status in the system.
Alternative Flow: **A1:** Step 3 The user decides to reject the appointment, the system will prompt the user confirmation message for rejection confirmation. **A2:** Step 5 The user decides not to approve the appointment, the system will return the user to the appointment details page without cancelling the appointment and aborting the current action.	
Post Condition: The scheduled appointment is approved and updated.	

3.3.5 Activity Diagram

3.3.5.1 Login Module



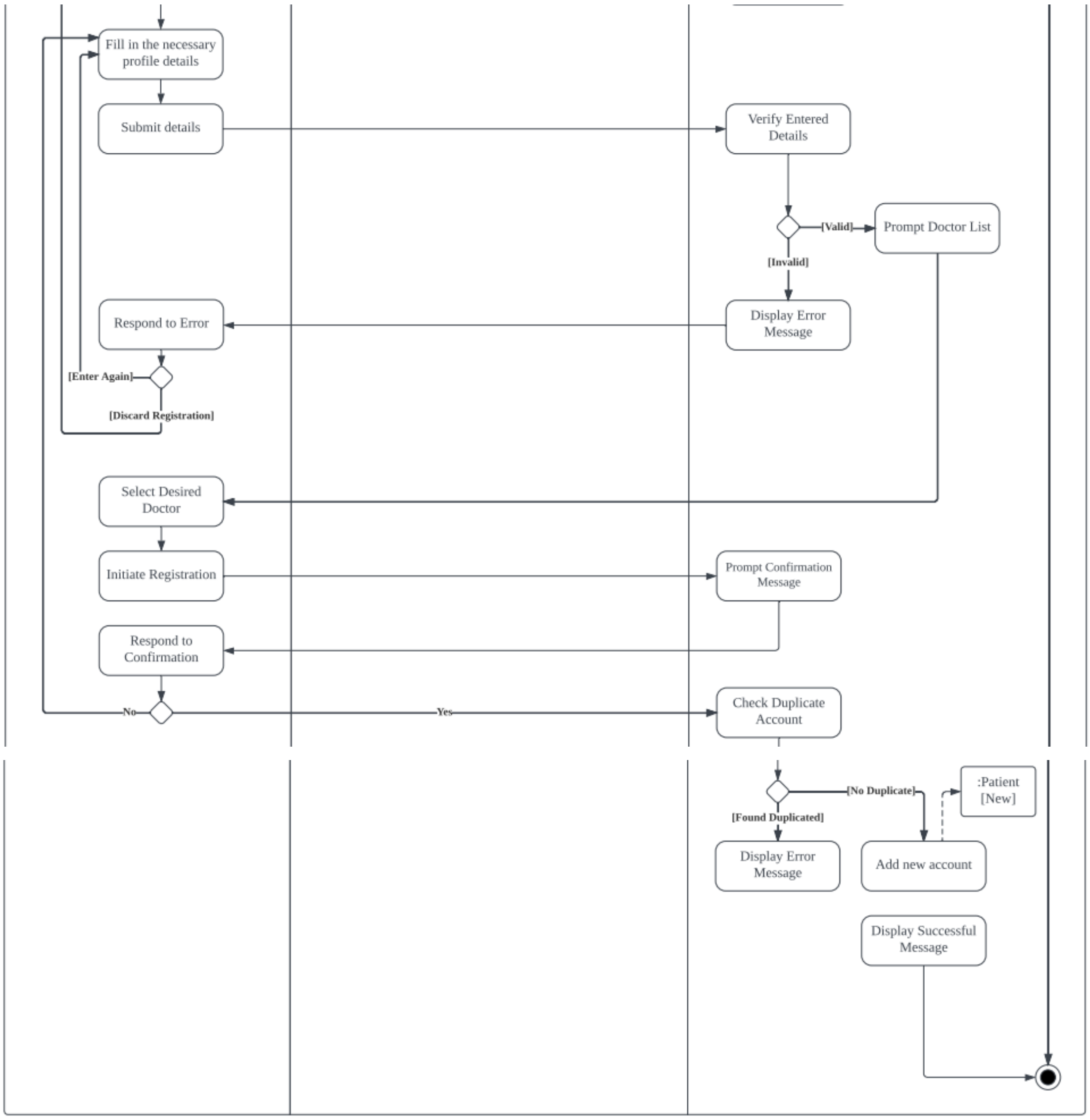


Figure 3.3.5.1.1: Activity Diagram of Login Module I

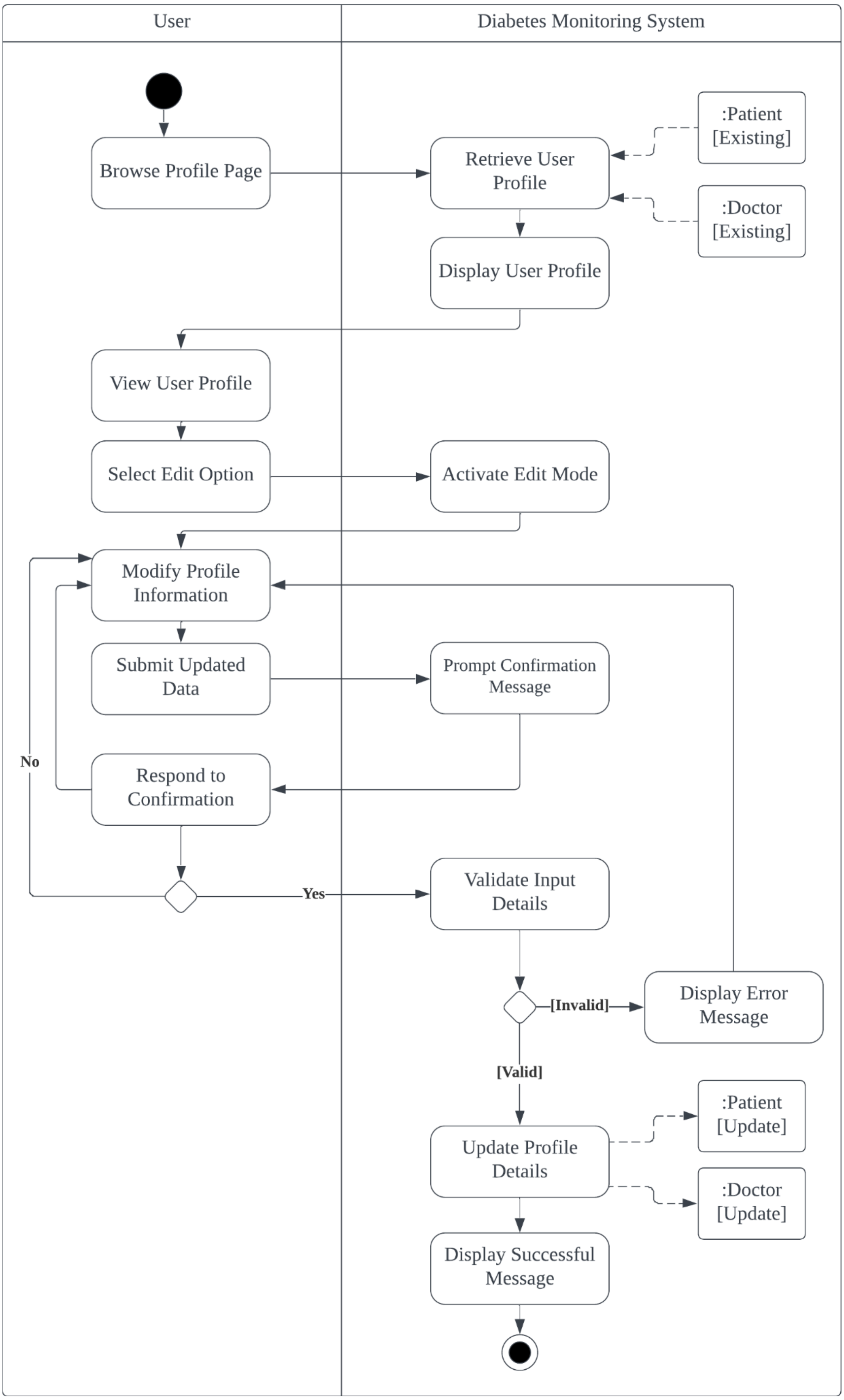


Figure 3.3.5.1.2: Activity Diagram of Login Module II

3.3.5.2 Doctor Module

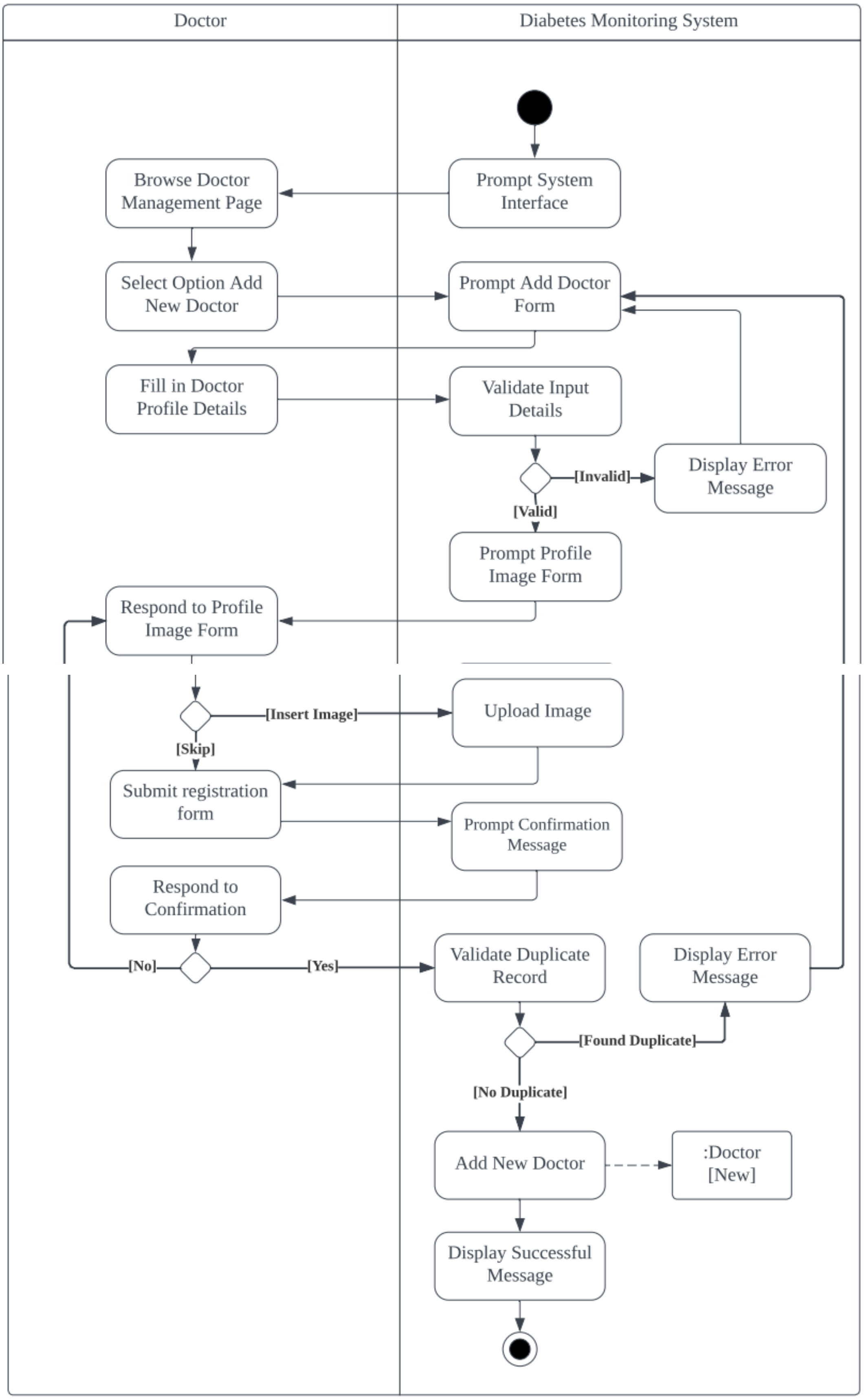
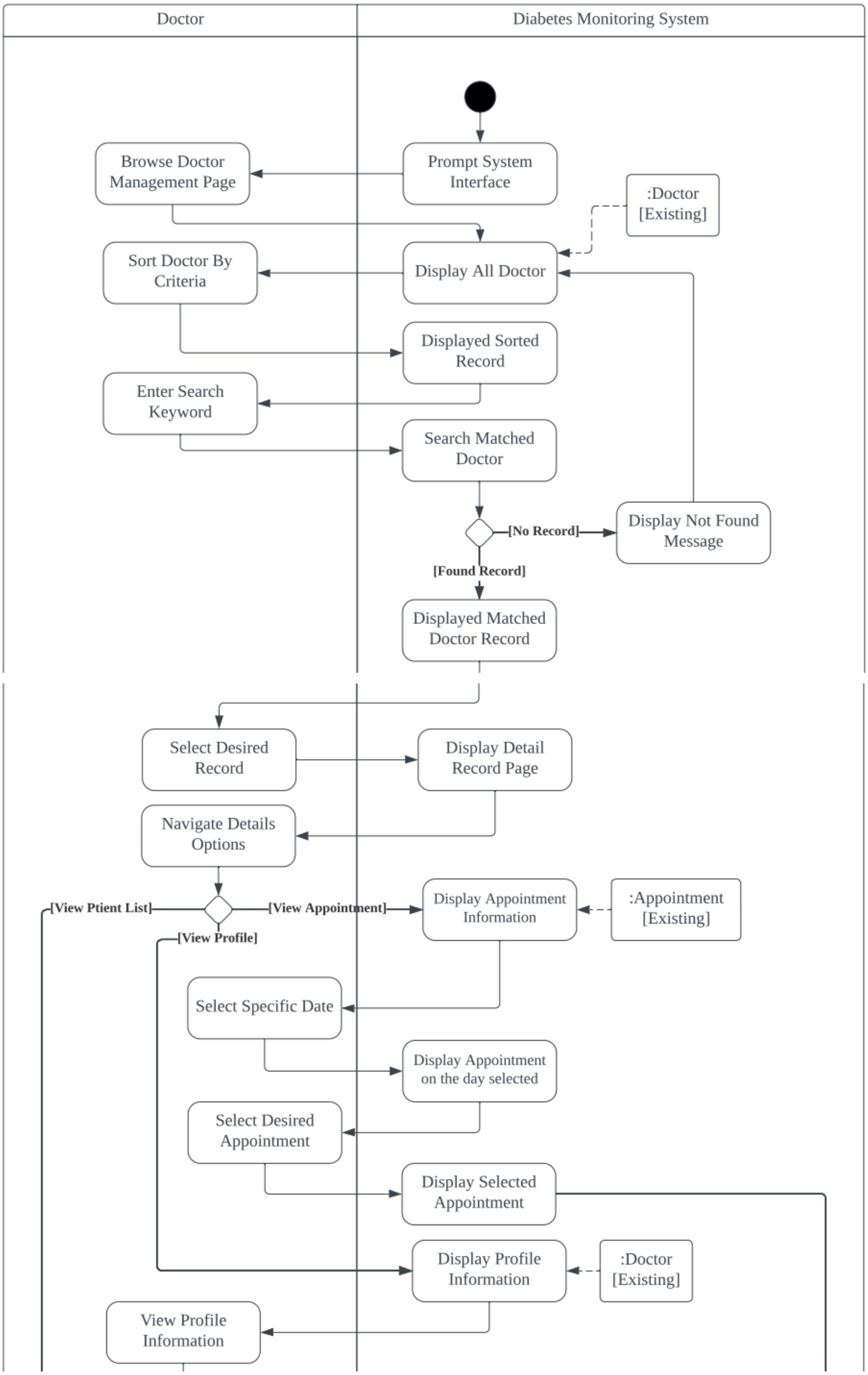
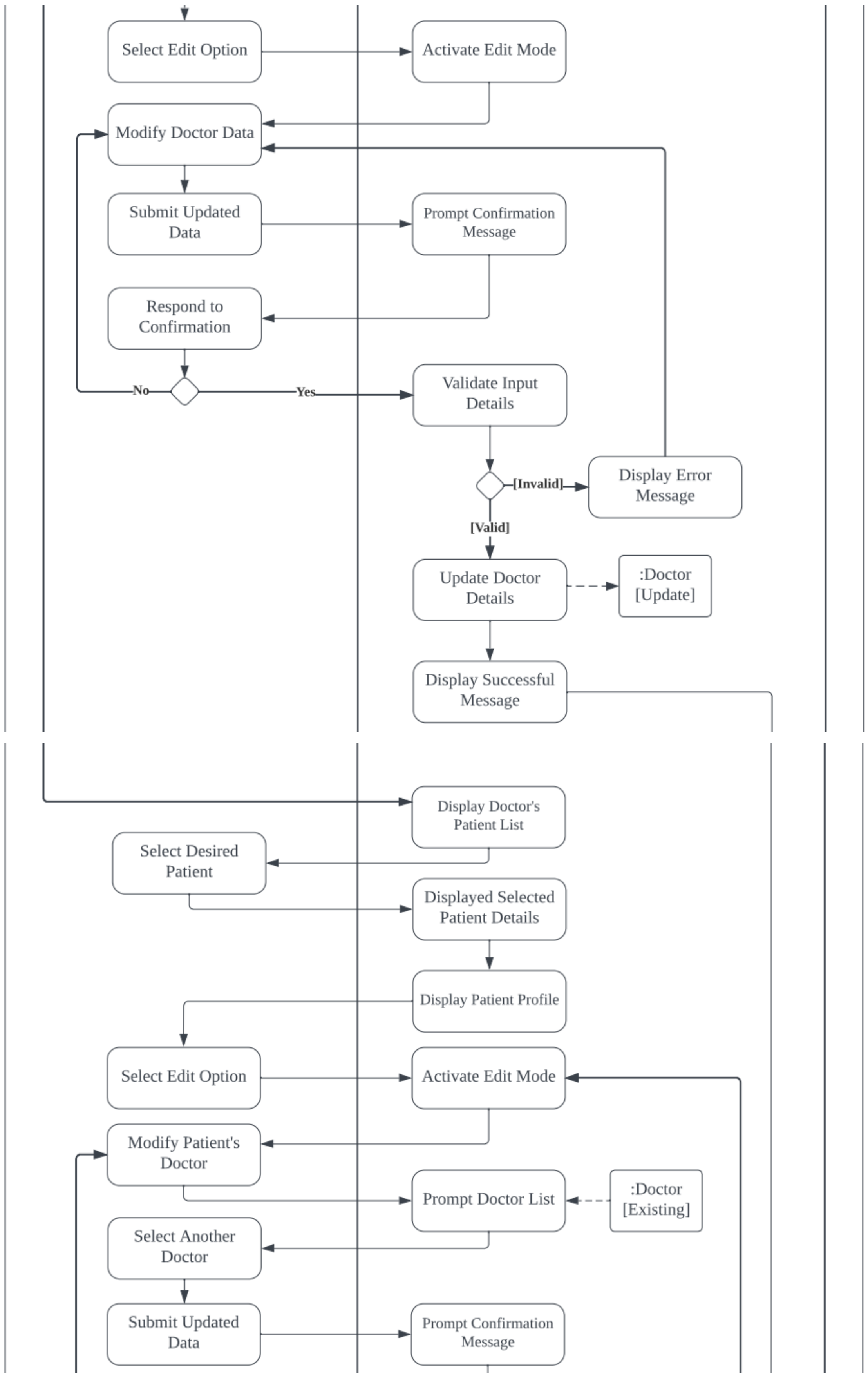


Figure 3.3.5.2.1: Activity Diagram of Doctor Module I


Doctor
Diabetes Monitoring System
Browse Doctor Management Page
Prompt System Interface
:Doctor [Existing]
Sort Doctor By Criteria
Display All Doctor
Displayed Sorted Record
Enter Search Keyword
Search Matched Doctor
[No Record]
Display Not Found Message
[Found Record]
Displayed Matched Doctor Record
Select Desired Record
Display Detail Record Page
Navigate Details Options
[View Ptient List]
[View Appointment]
Display Appointment Information
:Appointment [Existing]
[View Profile]
Select Specific Date
Display Appointment on the day selected
Select Desired Appointment
Display Selected Appointment
Display Profile Information
:Doctor [Existing]
View Profile Information


Select Edit Option
Activate Edit Mode
Modify Doctor Data
Submit Updated Data
Prompt Confirmation Message
Respond to Confirmation
No
Yes
Validate Input Details
[Invalid]
Display Error Message
[Valid]
Update Doctor Details
:Doctor [Update]
Display Successful Message
Display Doctor's Patient List
Select Desired Patient
Displayed Selected Patient Details
Display Patient Profile
Select Edit Option
Activate Edit Mode
Modify Patient's Doctor
Prompt Doctor List
:Doctor [Existing]
Select Another Doctor
Submit Updated Data
Prompt Confirmation Message

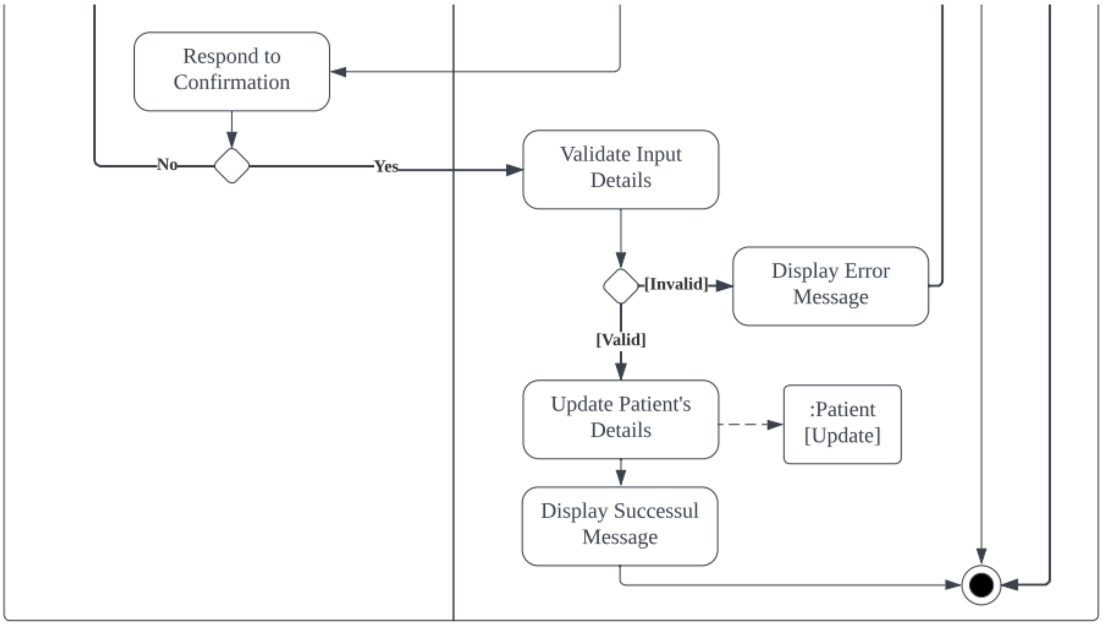
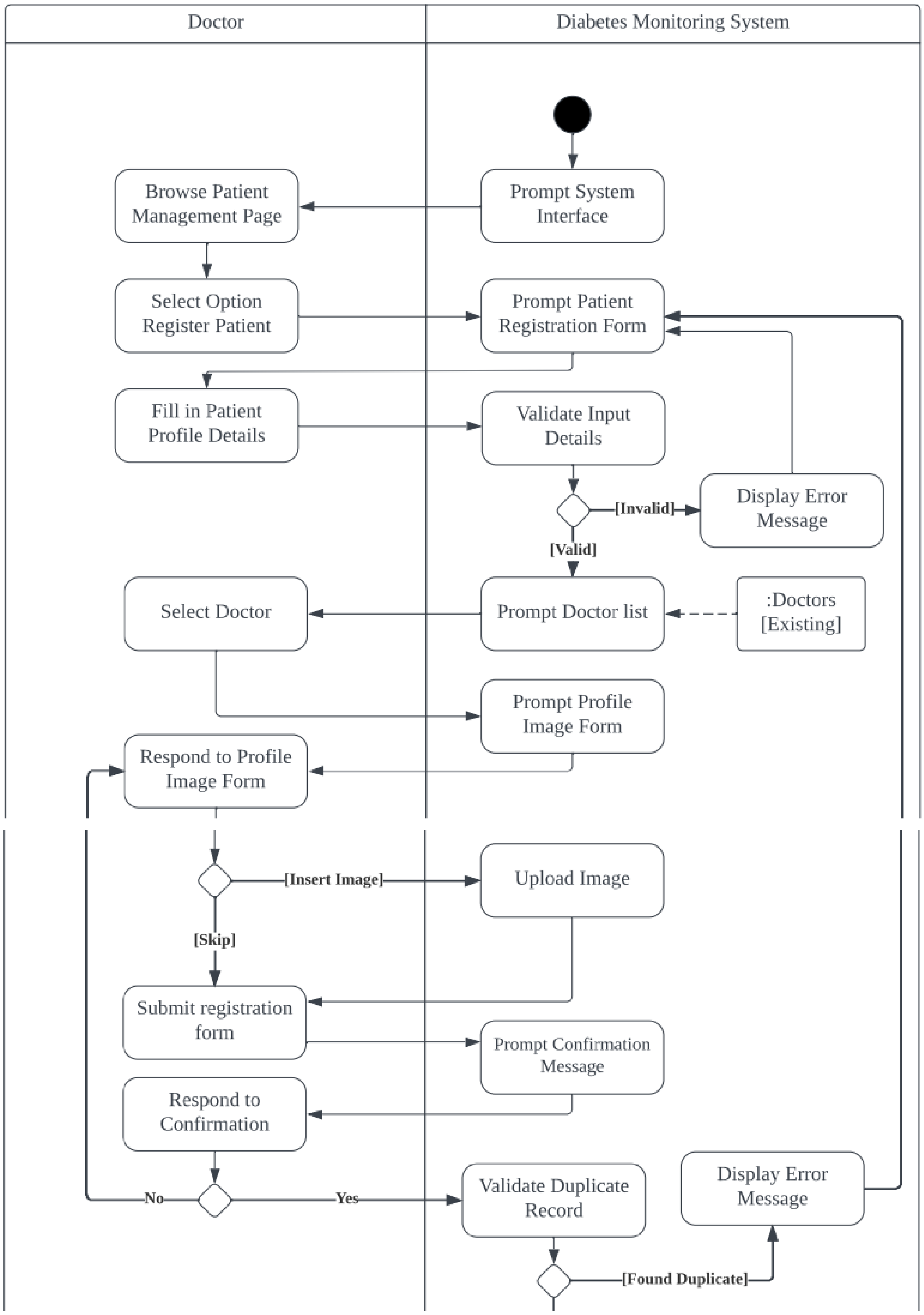


Figure 3.3.5.2.2: Activity Diagram of Doctor Module II

3.3.5.3 Patient Module



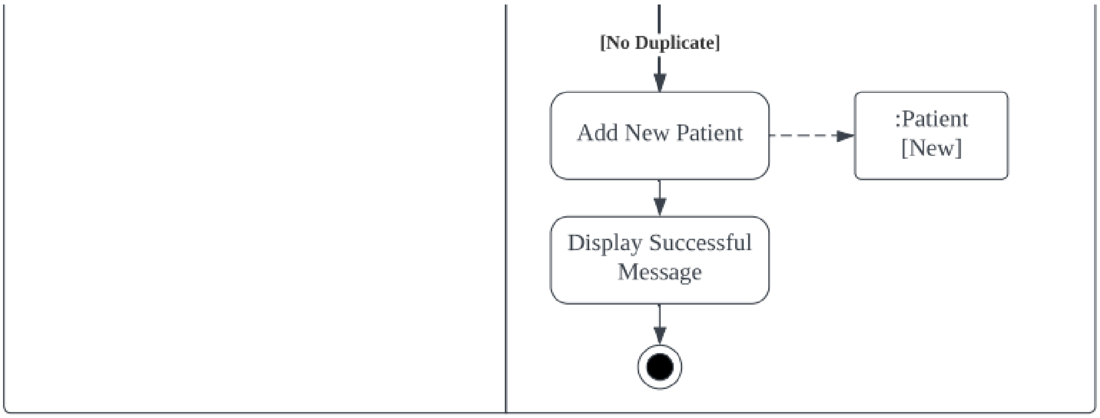
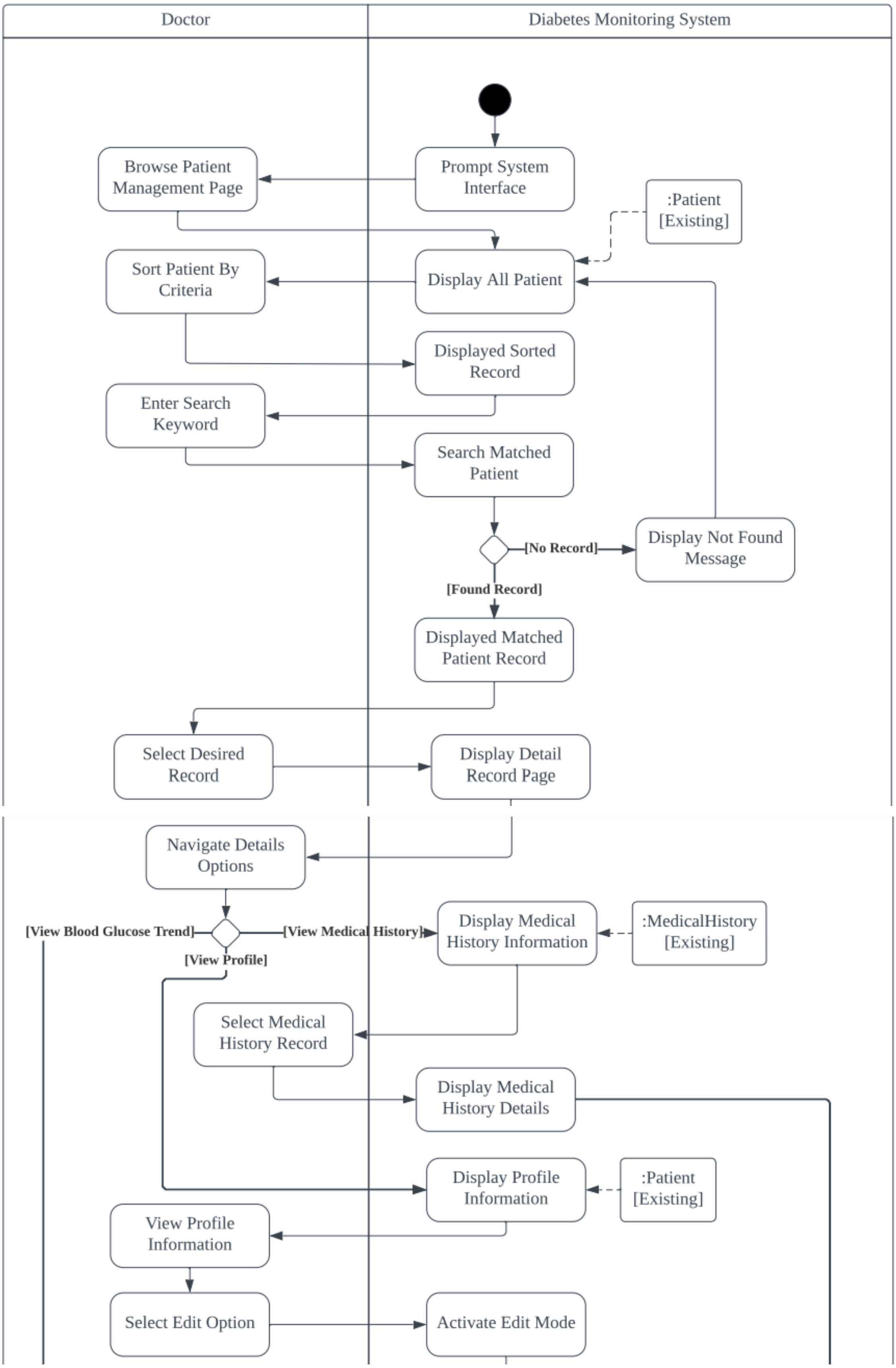
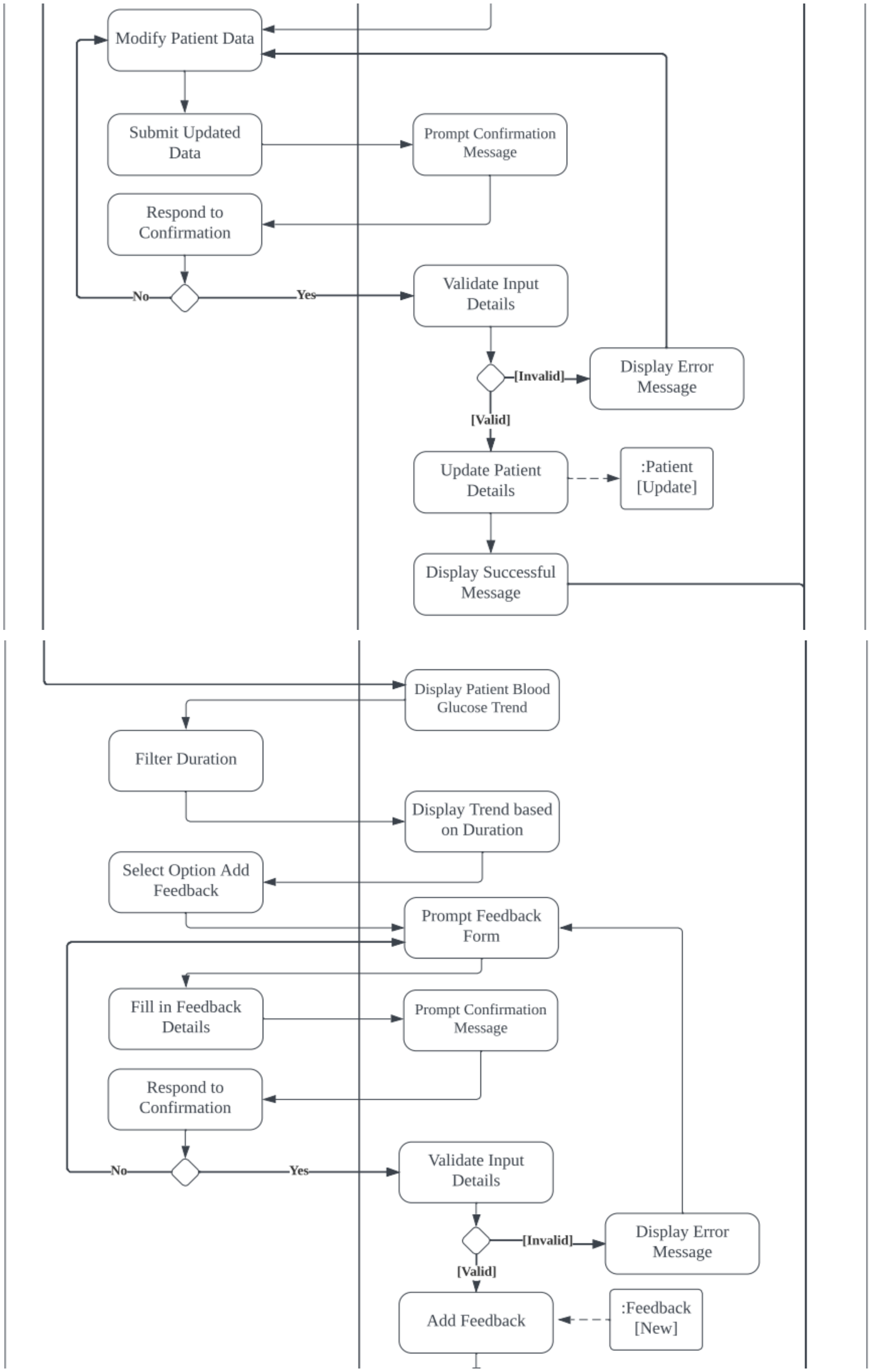


Figure 3.3.5.3.1: Activity Diagram of Patient Module I


Doctor
Diabetes Monitoring System
Browse Patient Management Page
Prompt System Interface
:Patient [Existing]
Sort Patient By Criteria
Display All Patient
Displayed Sorted Record
Enter Search Keyword
Search Matched Patient
[No Record]
Display Not Found Message
[Found Record]
Displayed Matched Patient Record
Select Desired Record
Display Detail Record Page
Navigate Details Options
[View Blood Glucose Trend]
[View Medical History]
Display Medical History Information
:MedicalHistory [Existing]
[View Profile]
Select Medical History Record
Display Medical History Details
Display Profile Information
:Patient [Existing]
View Profile Information
Select Edit Option
Activate Edit Mode


Modify Patient Data
Submit Updated Data
Prompt Confirmation Message
Respond to Confirmation
No
Yes
Validate Input Details
[Invalid]
Display Error Message
[Valid]
Update Patient Details
:Patient [Update]
Display Successful Message
Display Patient Blood Glucose Trend
Filter Duration
Display Trend based on Duration
Select Option Add Feedback
Prompt Feedback Form
Fill in Feedback Details
Prompt Confirmation Message
Respond to Confirmation
No
Yes
Validate Input Details
[Invalid]
Display Error Message
[Valid]
Add Feedback
:Feedback [New]

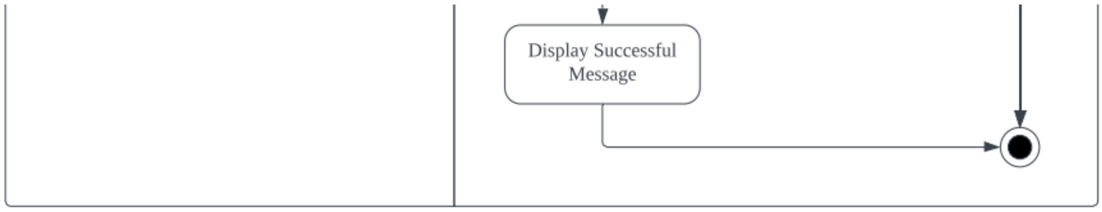
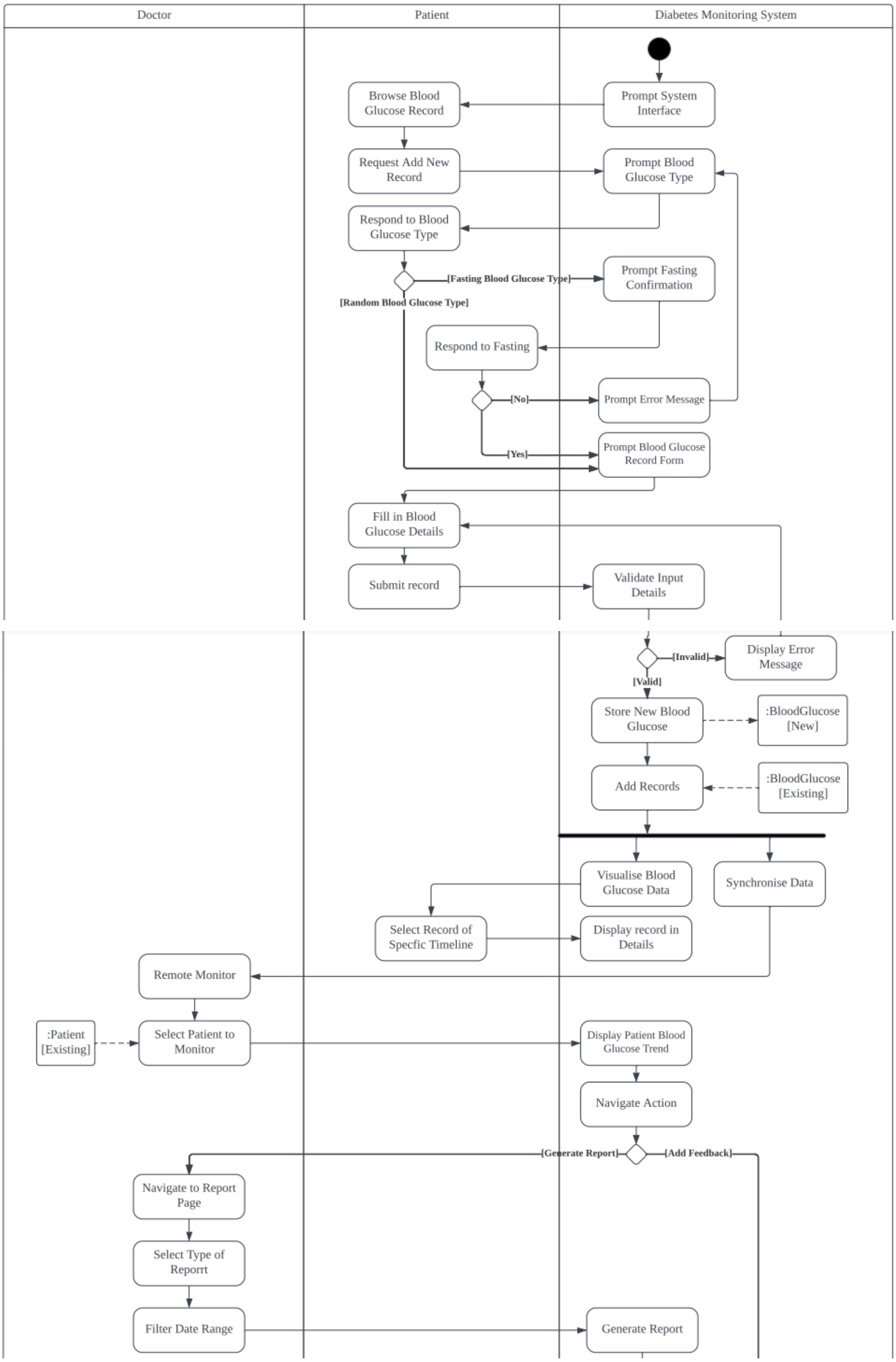


Figure 3.3.5.3.2: Activity Diagram of Patient Module II

3.3.5.4 Blood Glucose Monitoring Module



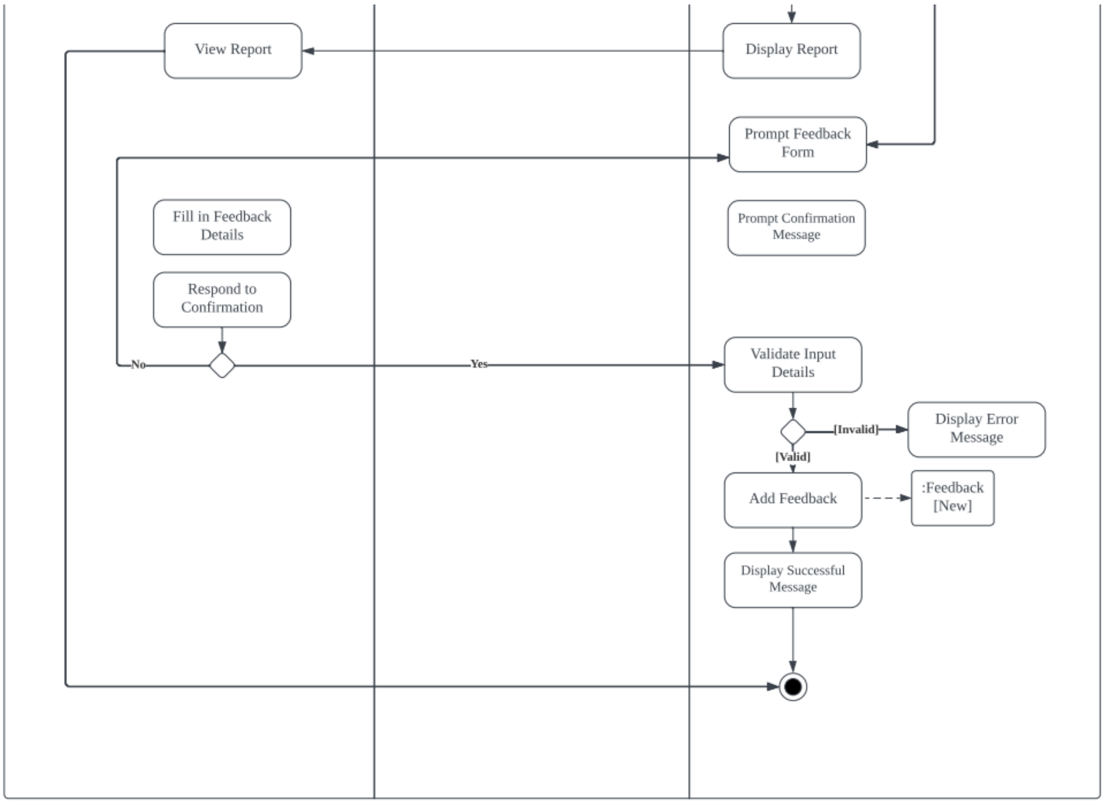
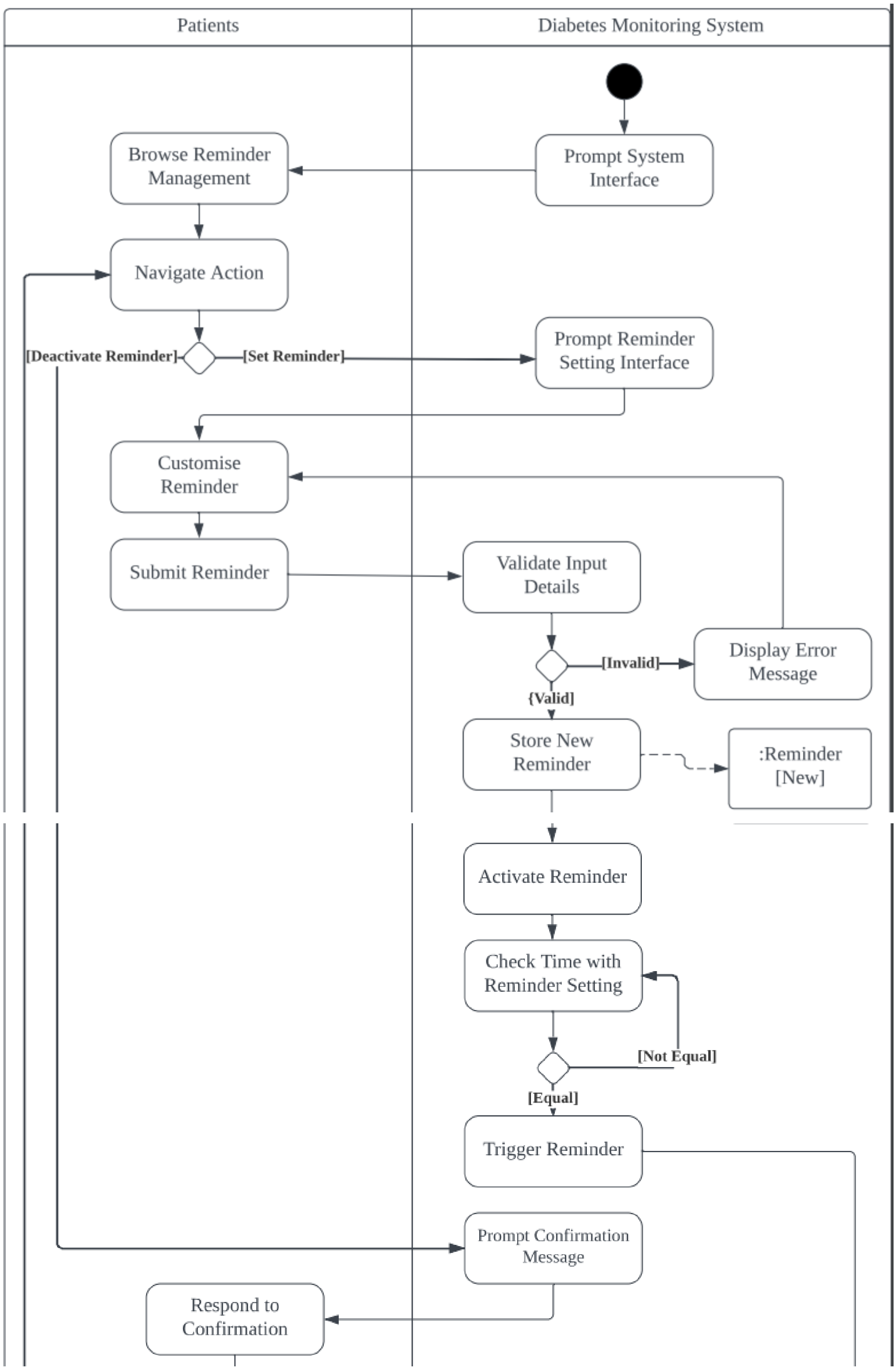


Figure 3.3.5.4.1: Activity Diagram of Blood Glucose Monitoring Module I


Patients
Diabetes Monitoring System
Browse Reminder Management
Prompt System Interface
Navigate Action
[Deactivate Reminder]
[Set Reminder]
Prompt Reminder Setting Interface
Customise Reminder
Submit Reminder
Validate Input Details
[Invalid]
Display Error Message
{Valid]
Store New Reminder
:Reminder [New]
Activate Reminder
Check Time with Reminder Setting
[Not Equal]
[Equal]
Trigger Reminder
Prompt Confirmation Message
Respond to Confirmation

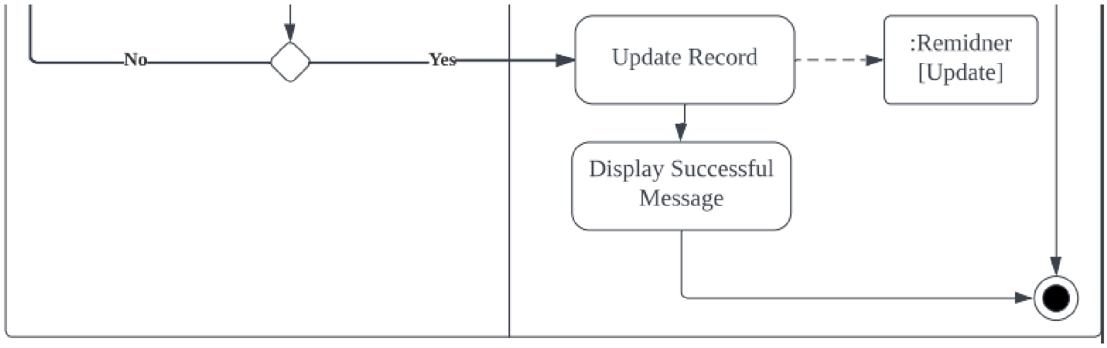


Figure 3.3.5.4.2: Activity Diagram of Blood Glucose Monitoring Module II

3.3.5.5 Ingredient Scanning Module

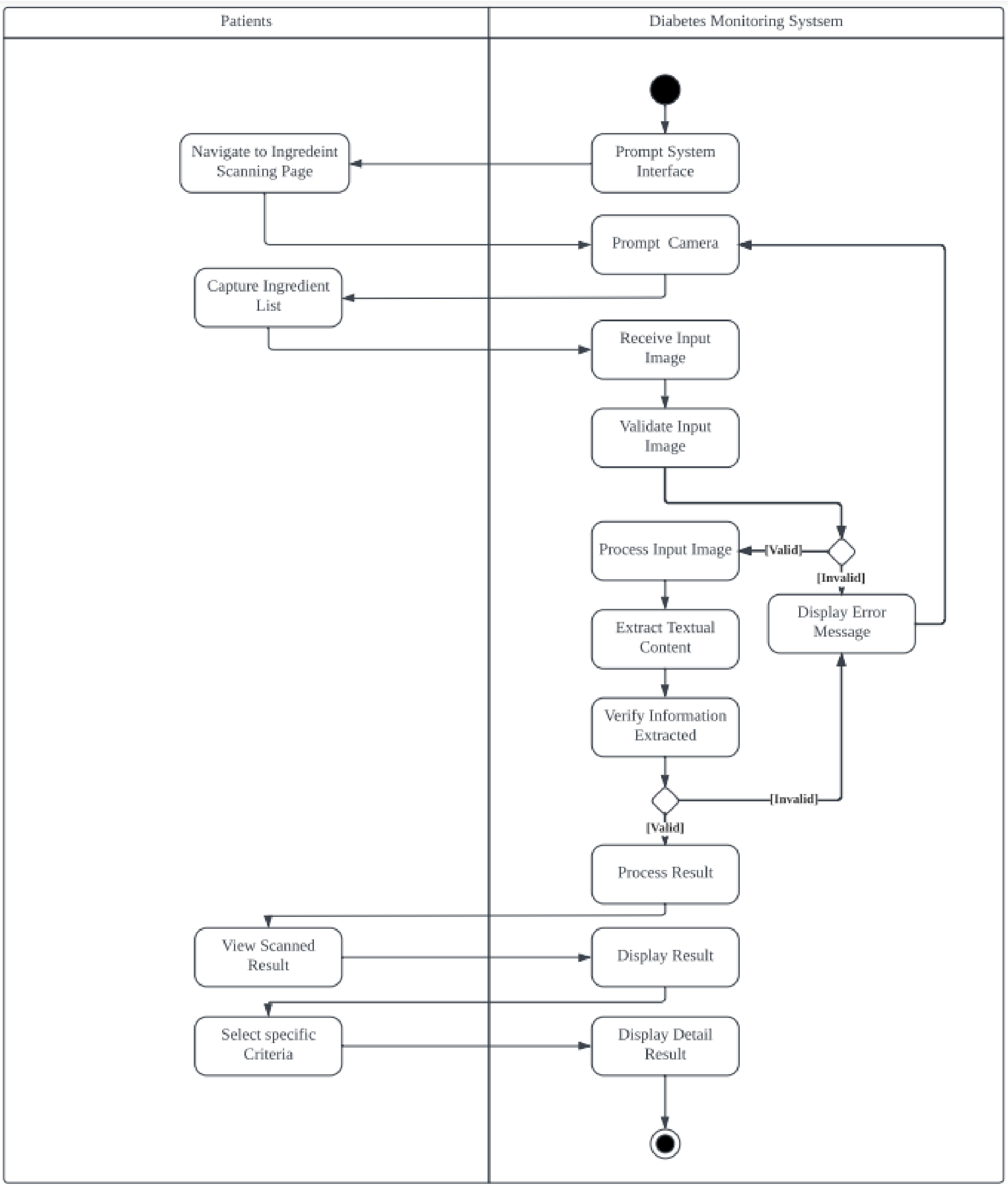


Figure 3.3.5.5.1: Activity Diagram of Ingredient Scanning Module I

3.3.5.6 Appointment Module

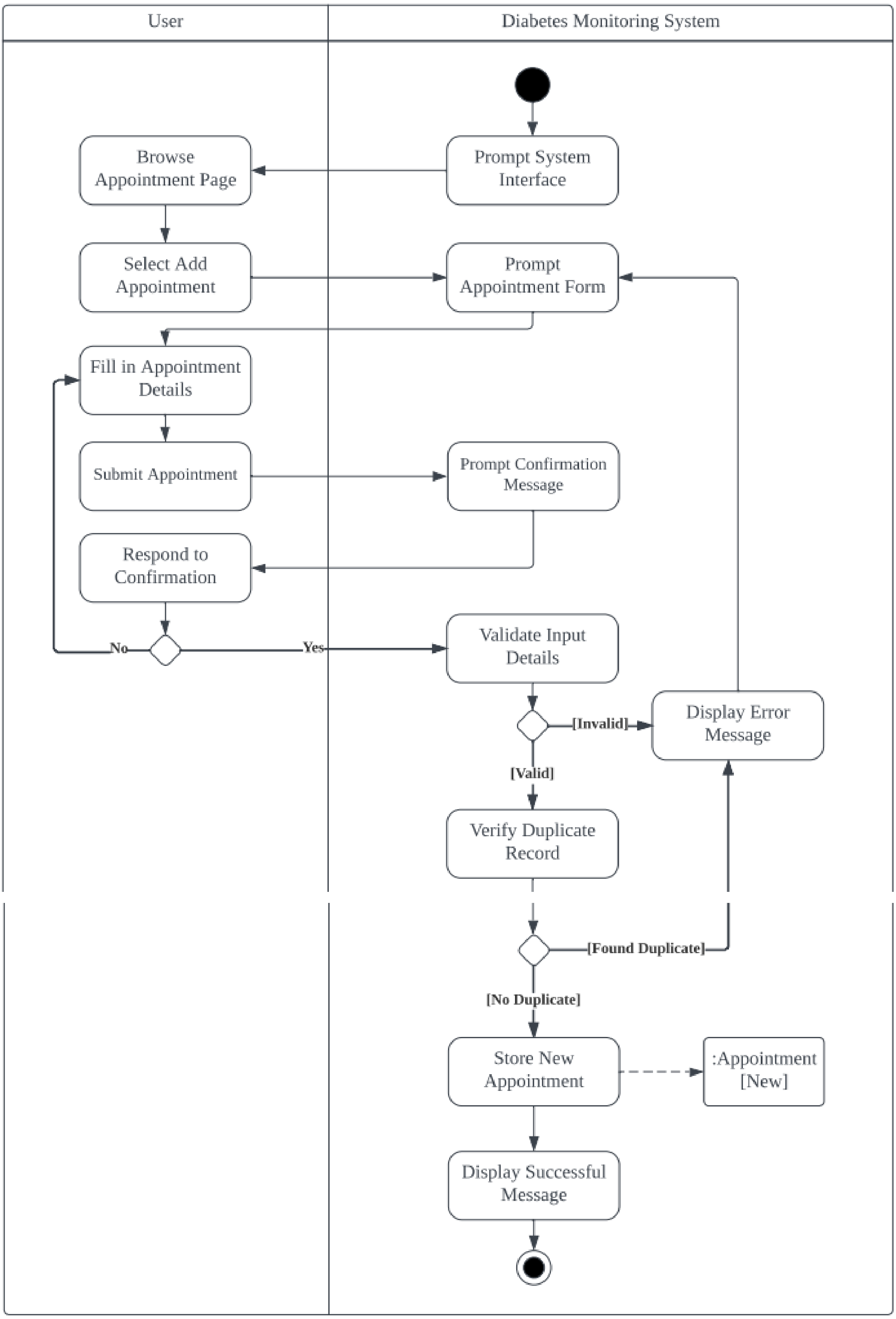
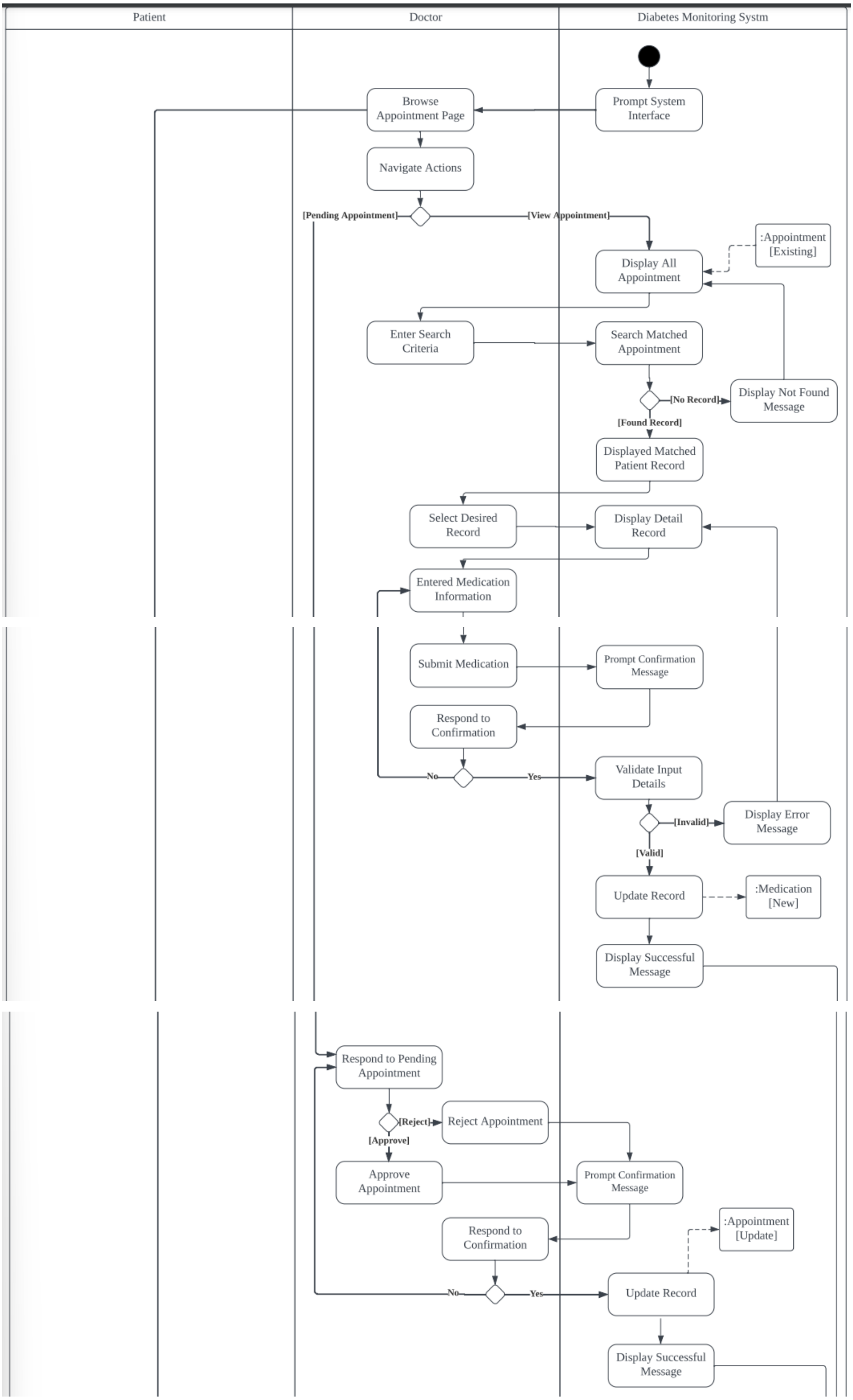


Figure 3.3.5.6.1: Activity Diagram of Appointment Module I


Patient
Doctor
Diabetes Monitoring Systm
Prompt System Interface
Browse Appointment Page
Navigate Actions
[Pending Appointment]
[View Appointment]
:Appointment [Existing]
Display All Appointment
Enter Search Criteria
Search Matched Appointment
[No Record]
Display Not Found Message
[Found Record]
Displayed Matched Patient Record
Select Desired Record
Display Detail Record
Entered Medication Information
Submit Medication
Prompt Confirmation Message
Respond to Confirmation
No
Yes
Validate Input Details
[Invalid]
Display Error Message
[Valid]
Update Record
:Medication [New]
Display Successful Message
Respond to Pending Appointment
[Reject]
Reject Appointment
[Approve]
Approve Appointment
Prompt Confirmation Message
Respond to Confirmation
:Appointment [Update]
No
Yes
Update Record
Display Successful Message

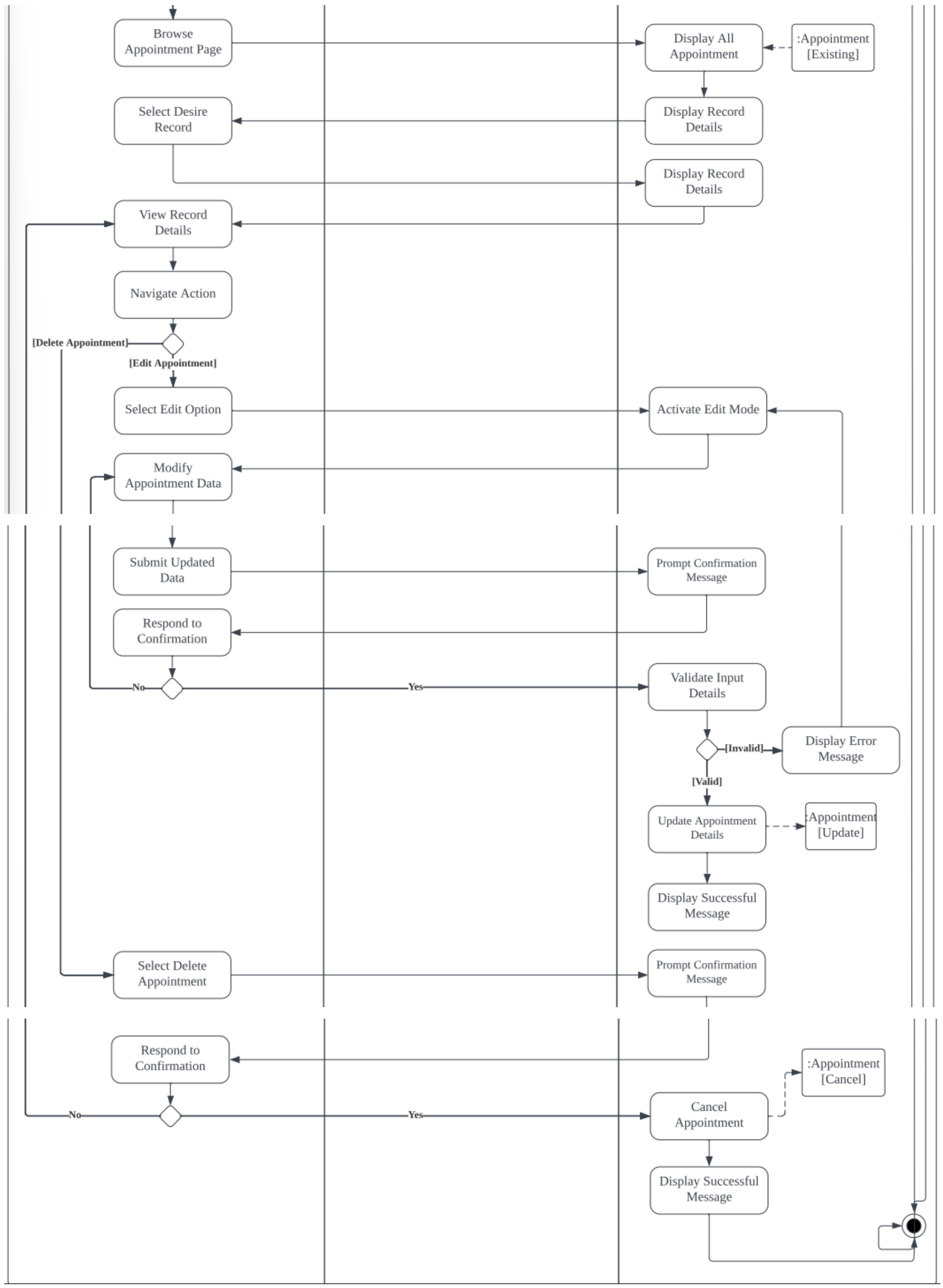


Figure 3.3.5.6.2: Activity Diagram of Appointment Module II

3.3.6 Functional Requirement

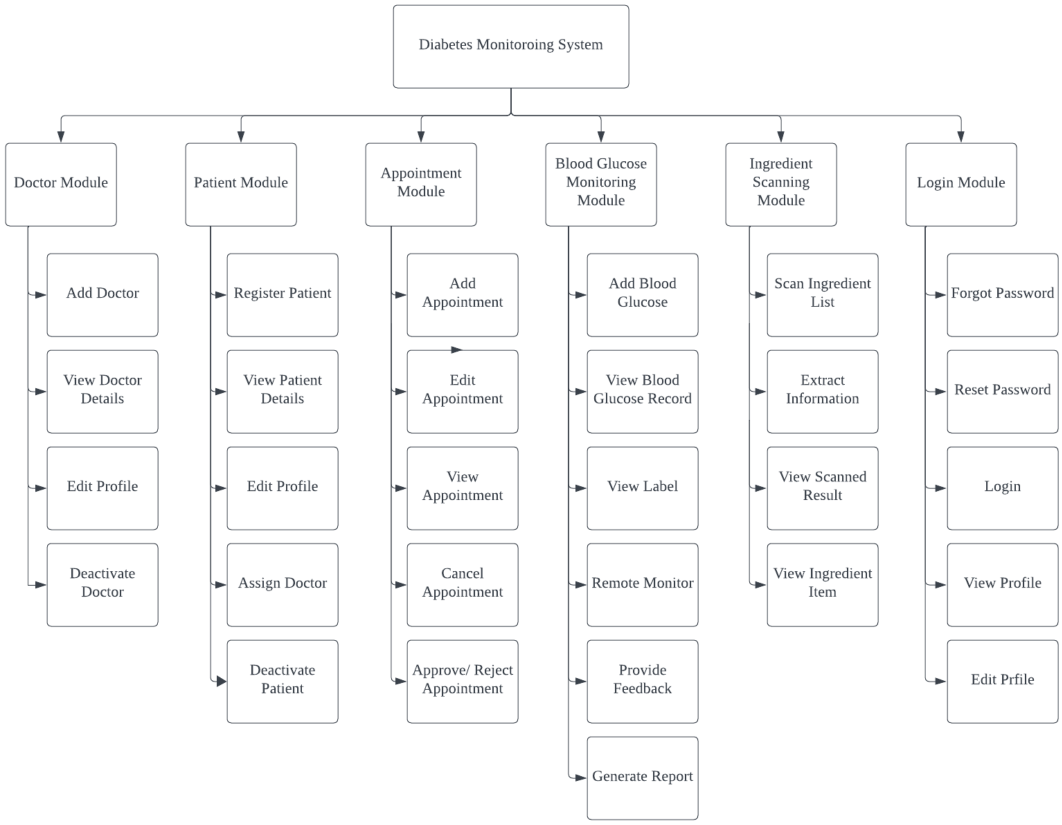


Figure 3.3.6.1: Functional Requirement

1.0 Login Module

- The system shall allow the user to log in.
- The system shall allow the user to register.
- The system shall validate the user credentials before login.
- The system shall allow the user to reset their password.
- The system shall allow the user to request a password recovery email.
- The system shall allow the user to view their profile.
- The system shall allow the user to edit their profile.
- The system shall allow the user to log out.

2.0 Doctor Module

- The system shall allow the user to add a new doctor.

- The system shall be able to detect duplicate doctor accounts during doctor creation.
- The system shall allow the user to view all the doctor records.
- The system shall allow the user to view the doctor's profile information.
- The system shall allow the user to view the doctor's appointment.
- The system shall allow the user to view the doctor's patient list.
- The system shall allow the user to edit the doctor's information.
- The system shall allow the user to deactivate the doctor.
- The system shall allow the user to search for doctor information.
- The system shall allow the user to sort the doctor's information.

3.0 Patient Module

- The system shall allow the user to register new patients.
- The system shall be able to detect duplicate patient accounts during patient registration.
- The system shall allow the user to assign a doctor to the newly registered patient.
- The system shall allow the user to view all the patient records.
- The system shall allow the user to view the patient's profile information.
- The system shall allow the user to view the patient's medical/follow-up records.
- The system shall allow the user to view the patient's blood glucose record.
- The system shall allow the user to edit the patient records.
- The system shall allow the user to search patient information.
- The system shall allow the user to sort patient information.
- The system shall allow the user to change the assigned doctor of a particular patient.

4.0 Blood Glucose Monitoring Module

- The system shall allow the patients to record their blood glucose.
- The system shall allow the patients to view all recorded blood glucose.

- The system shall visualise the recorded data for representations.
- The system shall allow the doctor to remotely monitor the data recorded.
- The system shall be able to synchronise the data between the doctor and the patients accurately.
- The system shall allow the doctor to filter the time range during monitoring.
- The system shall allow the doctor to provide timely feedback.
- The system shall allow the doctor to generate reports.

5.0 Ingredient Scanning Module

- The system shall allow the user to scan the ingredient list.
- The system shall extract the information from the ingredient list.
- The system shall allow the user to view scanned results.
- The system shall be able to display ingredient information.
- The system shall be able to highlight potential restricted items.

6.0 Appointment Module

- The system shall allow the users to add a new appointment.
- The system shall allow the users to approve a pending appointment.
- The system shall allow the users to reject a pending appointment.
- The system shall allow the users to edit a pending appointment.
- The system shall allow the users to delete a pending appointment.
- The system shall check the selected date for appointment in the future.
- The system shall check the available slot on the selected date.
- The system shall allow the patient to view their appointment.
- The system shall allow the patient to view their appointment history.
- The system shall allow the doctor to update the appointment.
- The system shall allow the doctor to approve the appointment.
- The system shall allow the doctor to reject the appointment.
- The system shall allow the doctor to add medication records.

3.3.7 Non-Functional Requirement

Performance

Performance requirements focus on the ability of the system to perform or carry out the demands of the users in a fast and efficient manner under a certain amount of workload. It can also be defined as the expected behaviour and characteristics of a system in terms of speed, efficiency and responsiveness. Performance requirements are essential to be considered during the system development to ensure the system meets the expectations of the user, ensuring satisfactory system usage. The performance requirement of the diabetes monitoring system must be fulfilled to deliver a satisfactory experience to the users. It should be able to fulfil the demand and request of the users within an acceptable time range and provide the functionalities required. The performance of the system is measured through metrics such as response time, throughput, etc. Response time is the maximum time allowed for the system to respond to the user's request or perform a specific action. For instance, the diabetes monitoring system should be able to respond within five seconds for a particular operation requested by the users. Throughput refers to the number of transactions the system can handle and manage within a given timeframe. The diabetes monitoring system is expected to achieve the throughput by processing 100 transactions of records smoothly without failure or data loss per minute.

Reliability

Reliability refers to the requirement that defines the degree to which a software product performs its functions for a certain period under predetermined conditions (ÖZCAN et al., 2020). It defines the expected level of dependability and stability of a system. The requirements related to reliability focus on the ability of the system to operate consistently and reliably, and carry out the functionalities and requirements without or with minimum failure over a specific period. In other words, the requirement of reliability provides the user with a dependable and trustworthy user experience through consistent system performance. Reliability requirements can include the availability of the system that is specified through the percentage of time that the system should be accessible and operations. For example, the proposed system should be able to have 99% availability, able to be assessed and provide desired functions at least 99% of the time. Other than availability, reliability can also be measured through Mean Time Between Failures (MTBF) and Mean Time to Repair (MTTR). MTBF is the measure of expected time between system failure while MTTR refers to the acreage time the system takes to repair or restore itself from any failure (Atlassian, 2023). The proposed system should aim for an MTBF of 30 days and an MTTR of 1 hour.

Usability

The usability of the system is evaluated through the system performance in terms of ease of use, user-friendliness and the user's satisfaction. The usability of the system can be evaluated through the learnability rate of the users, which refers to the ability and ease with which the users can understand and learn to operate the system. This can be determined by observing the time required by the user to understand and manage the operation of the system efficiently. For instance, the users shall be able to learn how to perform basic tasks or navigate through the system within one hour. Other than that, the efficiency of the interaction between the user and the system is another method to address the system's usability. The efficiency can be achieved by reducing unnecessary steps, and optimising workflow for faster and more efficient user interactions. By reducing the time and effort required for users to accomplish their tasks within the system, efficiency can be improved. The achievement of the usability requirement can also be evaluated through user satisfaction. For example, the proposed system should be able to achieve a user satisfaction rate of at least 80%. Feedback from the users of the system can be collected through surveys or questionnaires to gather insight into the accomplishment of the usability requirements.

Security

Security requirements play a crucial role in ensuring the confidentiality, integrity and availability of the system and its data. They are essential to protect the system and prevent issues such as unauthorised access, data breaches or any potential security threats. Confidentiality is one of the approaches that protect sensitive information from unauthorised disclosure. It allows only authorised individuals to access certain resources or data within the system. For instance, the proposed system should implement an authentication mechanism, requiring users to enter their credentials before logging into the system. This can ensure that only authorised individuals with valid credentials can access their accounts and perform operations. In addition to authentication, authorisation is another crucial aspect to be included. It is a process that grants or denies access rights to users based on their authenticated identities and permissions associated with those identities. It defines the level of access that an authenticated user has to the resources or services provided by the system. For example, the system should assign roles to each user in the system based on their requirement to ensure they are only allowed to access the resources they require.

3.4 Chapter Summary and Evaluation

The Agile Scrum methodology is chosen for this project because of the numerous advantages it offers. Implementing an iterative and incremental approach can ensure that the system development progresses step by step in small and manageable steps to deliver a software product that fulfils the requirements of the potential users. Agile Scrum methodology emphasises active and frequent participation from the client or stakeholders. The involvement of these participants enables a better understanding of the system requirements and facilitates continuous improvement throughout the system development process.

The technique used for data collection is interview and questionnaires. The target interviewees are healthcare providers such as doctors and diabetes individuals. The purpose of interviewing doctors is to gather insights into the patient monitoring process and the consultation sessions, while interviewing diabetes individuals aims to understand their daily monitoring routines and the challenges they face in maintaining their health. By interviewing the doctor, valuable information can be obtained regarding their practices and experiences in patient monitoring and consultation. The input is crucial to form the foundation of the system to be developed, ensuring it possesses the correct flow. During the interviews, the opinion of the interviewees on the implementation of a diabetes monitoring system with an ingredient scanning feature is collected and their attitudes towards the integration of such a system in their activities is observed. It is found that both the doctor and the individuals show a positive attitude toward the development of such a system. In addition to the interview, a questionnaire was distributed to the public to explore the understanding of the public on the ingredient list and their awareness of the impact of their dietary choices towards their health. A total of 187 respondents were collected. Based on the collected responses, it is found that while a portion of the public utilises the ingredient list, they will not seek professional advice if they encounter difficulties in comprehending. However, some respondents indicated they would conduct their research through the Internet and make modifications to their dietary habits accordingly. This has shown that the public is aware of the importance of information on the ingredient list. Through the interview and questionnaire, it is concluded that the proposed system can bring significant benefits and contributions to a certain group of people.

The development environment of this project will be using a laptop as the main hardware to carry out the development processes. Development tools such as Android Studio, Google Colab and Jupyter Notebook will be employed to support the development activities. In addition to those development tools, open-source software such as OpenCV, TensorFlow and Tesseract will be utilised to enhance the development of the proposed system. For the operating environment, the Firebase hosting service will be selected as the main operating environment by subscribing to the Spark plan, which is the free plan offered. Although there is a better plan offered by Firebase with advanced functions and features, the Spark plan is sufficient for the development and operation of the proposed system.

In this project, the proposed system is decomposed into six modules. To promote better understanding and clarity, several graphical representations such as the activity diagram and the use case diagram are presented. The use case descriptions for each module are written to further elaborate the functionalities and the processes of the modules, clarifying the modules in detail. In terms of non-functional requirements, the focus will be on several key aspects including performance, reliability, usability, and security to ensure the system can provide satisfactory performance.

Chapter 4

System Design

4.1 UI Design

4.1.1 User Interface (Web Application)

4.1.1.1 Login Module

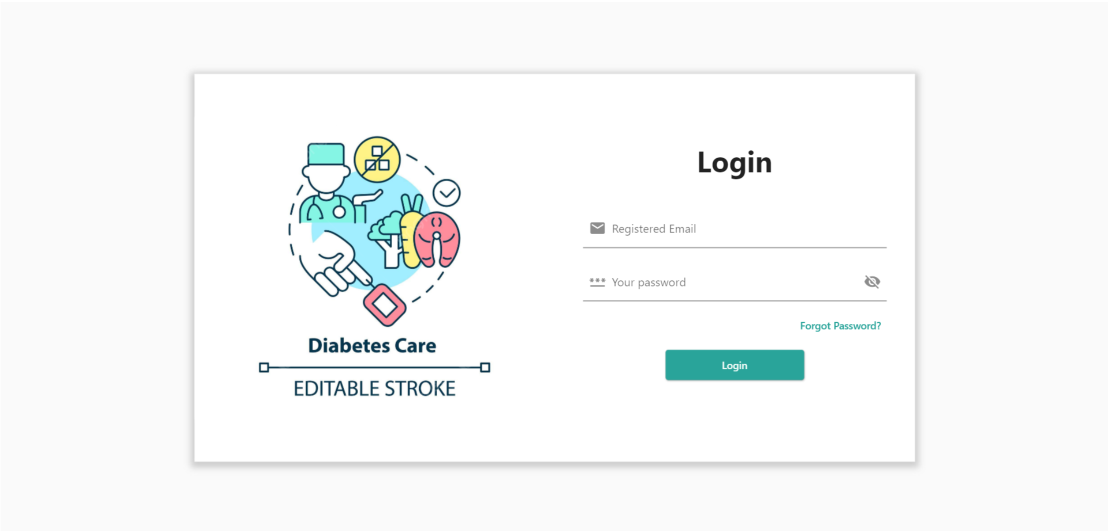


Figure 4.1.1.1.1: User Interface of Login Page

The system requires users to log in before accessing the system. During the login process, the users have to enter their login credentials for authentication. If the entered credentials match an existing user's data, the system will redirect them to the corresponding homepage based on their role within the system. However, if the user enters invalid credentials, the system will display an error message stating "Invalid login credentials." In the case the user forgets their account credentials, they can initiate the password recovery process through the "Forgot Password" feature.

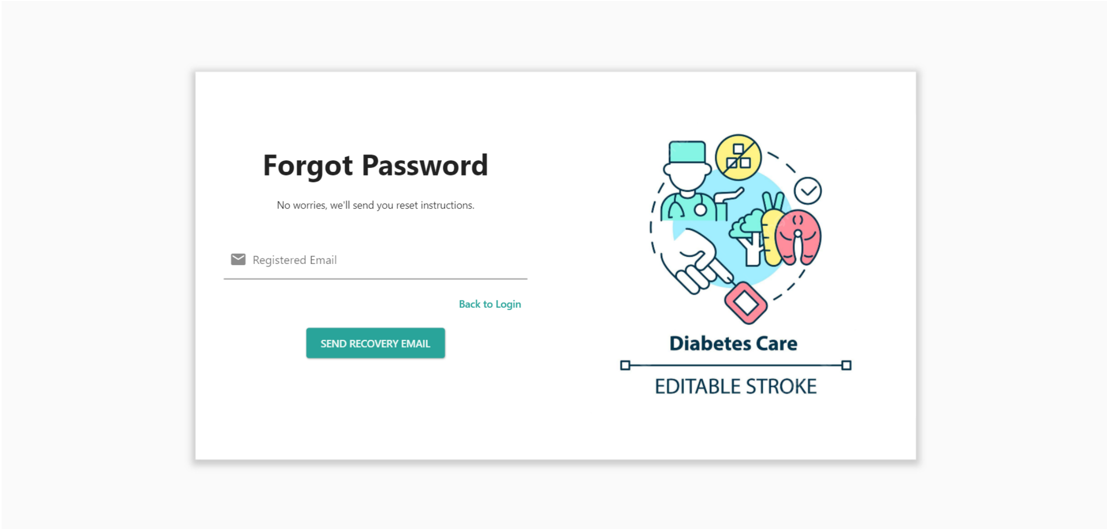


Figure 4.1.1.1.2: User Interface of Email Verification

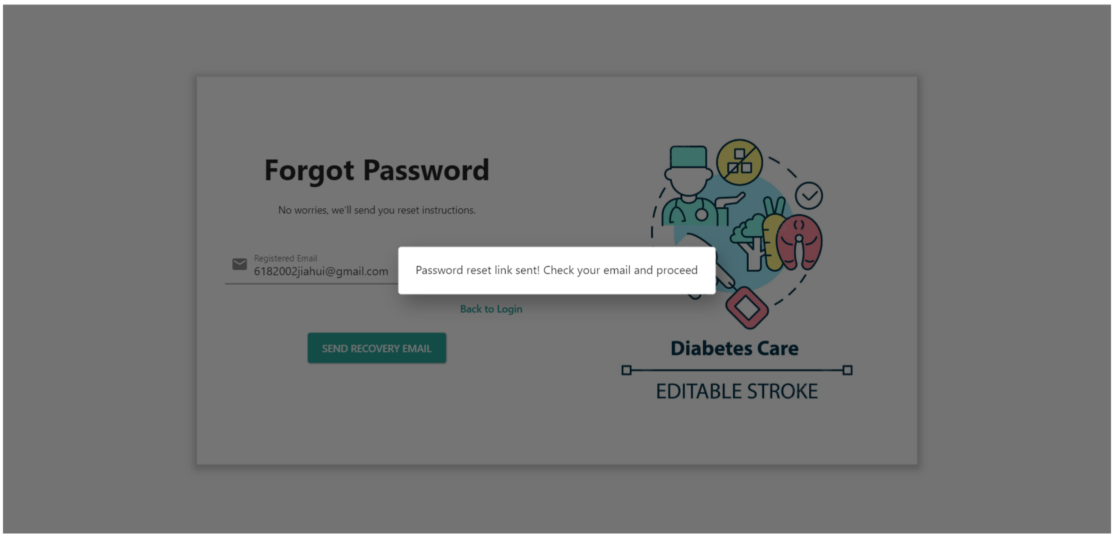


Figure 4.1.1.1.3: User Interface of Email Sent Alert Message

The email verification page is designed to ensure security when the users request password recovery. On this page, the users need to enter their email addresses. If the entered email address is valid and matches an existing user record in the database, the system will generate an email which consists of a password reset link. To notify the user about the email sending, an alert message will be prompted.

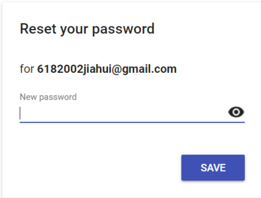


Figure 4.1.1.1.4: User Interface of Reset Password

On this page, the users must enter their new password. After they enter the new password and initiate the reset action by clicking on the “save” button, their password stored in the database will be updated accordingly. The user then can utilise the new password to log in to their account.

Figure 4.1.1.1.5: User Interface of Homepage

Upon successful login, users, particularly the doctor, will be directed to the system's homepage. The system displays the total number of active doctors and patients, attempting to provide information such as the ratio of doctors to patients. This helps the healthcare providers to have a brief and better overview of how many patients each doctor has to care about, ensuring that the patient can have the proper care.

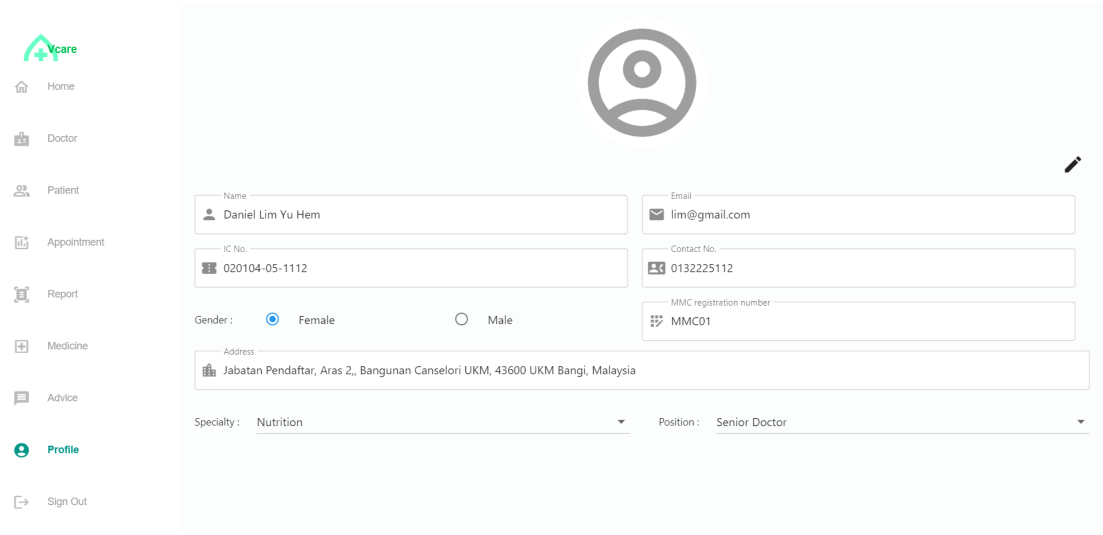


Figure 4.1.1.1.6: User Interface of Profile

From the navigation sidebar, the user can navigate to their profile page after login. In this profile page, by clicking on the image icon, the user can upload a new image, updating their profile. The user only can edit their profile information after activating the edit mode. While the edit mode is activated, the email for the doctors remains uneditable as the email serves the login purpose. Not only that, the options of the speciality and position will be constrained to prevent the user from changing.

4.1.1.2 Doctor Module

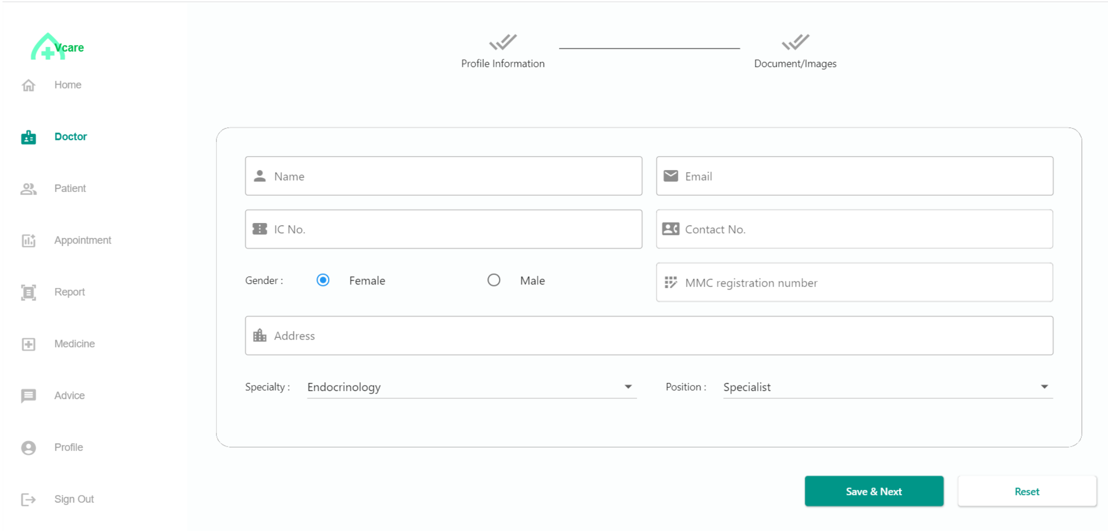


Figure 4.1.1.2.1: User Interface of Add Doctor I

The add doctor page is only accessible by the doctor. The form for adding new doctors is divided mainly into two sections. In the first session, the system will request the user to enter the doctor's profile information, which includes name, IC, email, contact, MMC registration number, address, speciality and position. Above the form displays the progress bar that provides information to the user, showing that currently they are in the first step of adding a doctor. The two buttons below the form, “Save & Next” will temporarily store the data entered when being clicked, allowing data to be restored when the users come back to this page after proceeding to the next step of adding a new doctor. Whereas the button “Reset”, will clear off every field, returning an empty form to the user.

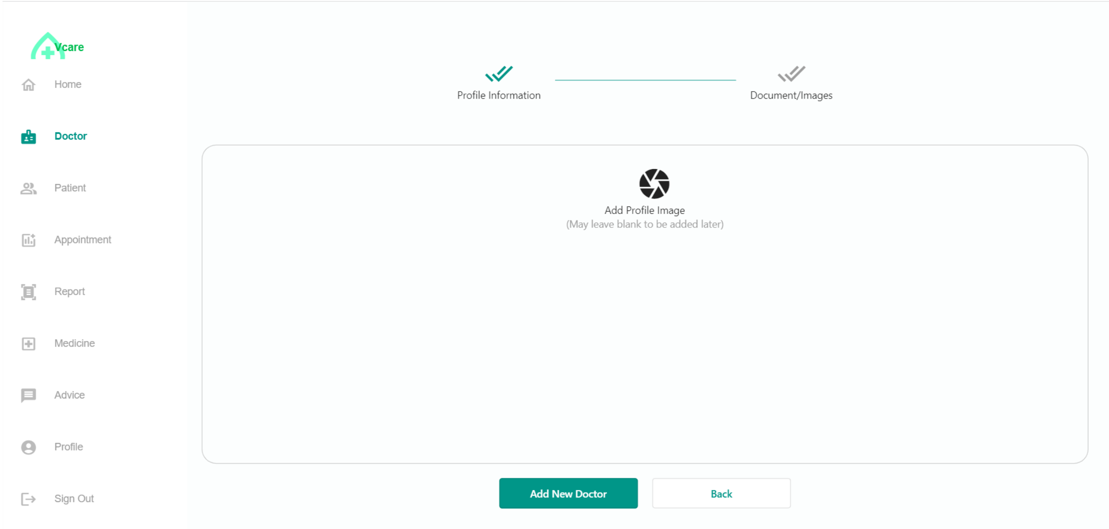
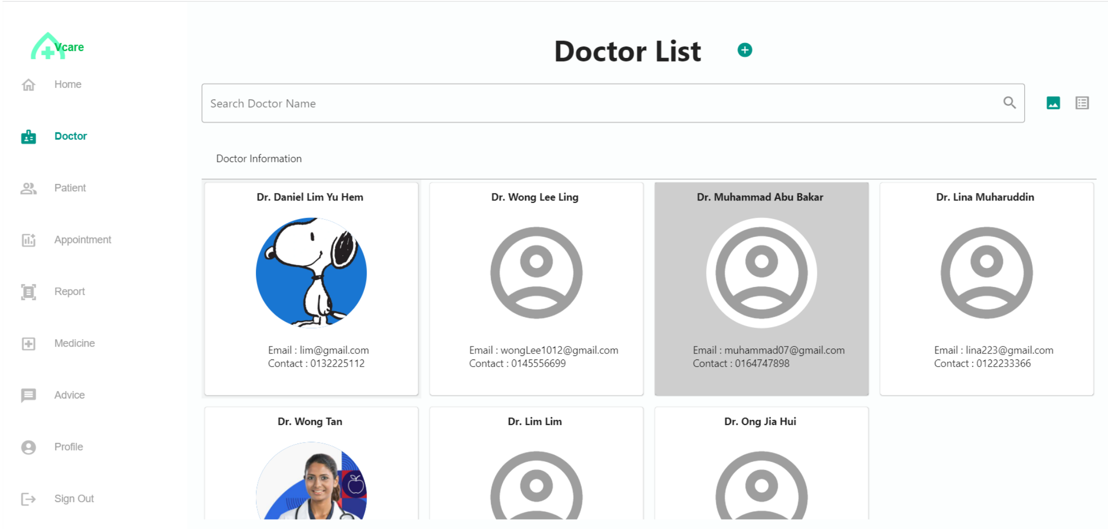


Figure 4.1.1.2.2: User Interface of Add Doctor II

Section II of the doctor adding includes an action of uploading the doctor's profile image. In this step, the user can opt to skip the profile image uploading by directly clicking on the "Add New Doctor" button, this will immediately initiate the doctor adding action into the system although without an image. If the users want to include a profile image when adding a new doctor, they can click on the section labelled "Add Profile Image" and select the image from the computer. The progress bar above indicates that the adding doctor has partially completed. The "Back" button beside the "Add New Doctor" button allows the users to navigate to the previous step, the profile information form, if modification is required.



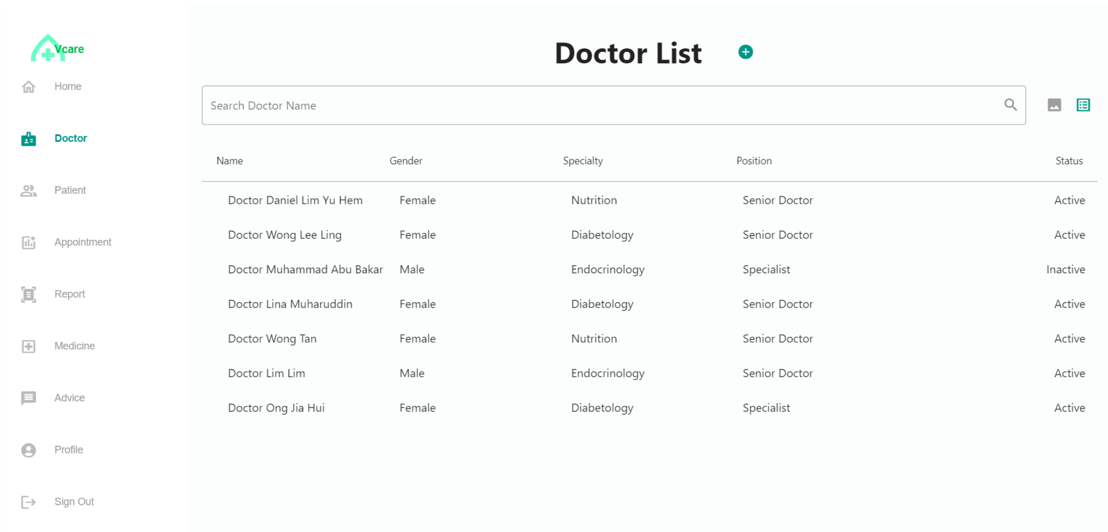


Figure 4.1.1.2.3: User Interface of View Doctor

On the doctor's main page, all the doctors' information in the system will be displayed. The user can use the search box to search for certain doctors. To initiate the search function, the user will have to enter the search keywords. This search box will search through the doctor's name, such that when ‘Lim’ is searched, all the doctors with ‘Lim’ within their name will be displayed. Not only that, the user can opt for different methods in displaying the information, the list view where information is displayed in rows and the details view where the doctors’ profile images are displayed. By clicking on the header of the doctor list, the doctor information will be sorted based on the selected criteria. By selecting the doctor record from the list, the system will redirect the user to the detailed information of the doctor selected.

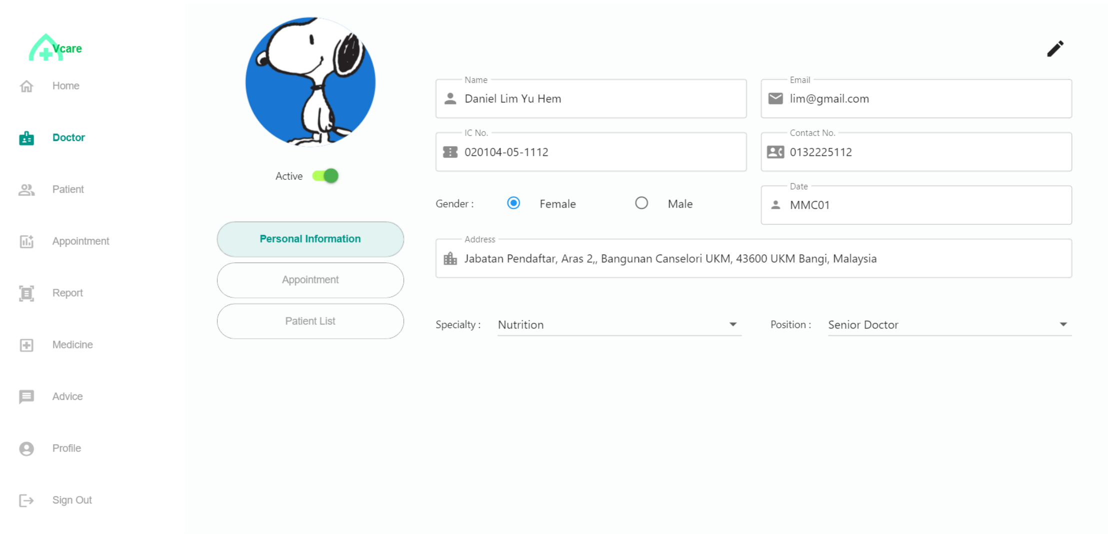


Figure 4.1.1.2.4: User Interface of View Doctor Personal Information

The doctor information will be divided into three sections, personal information, appointment and patient list. Under the personal information section, information such as the doctor's name, contact number, IC, email and so on. The user only can edit the information on this page by clicking the pencil button to activate the edit mode. While the edit mode is activated, the email for the doctors remains uneditable as the email serves the login purpose. The user can toggle the 'Active' switch to deactivate this particular doctor when his/ her account is no longer active.

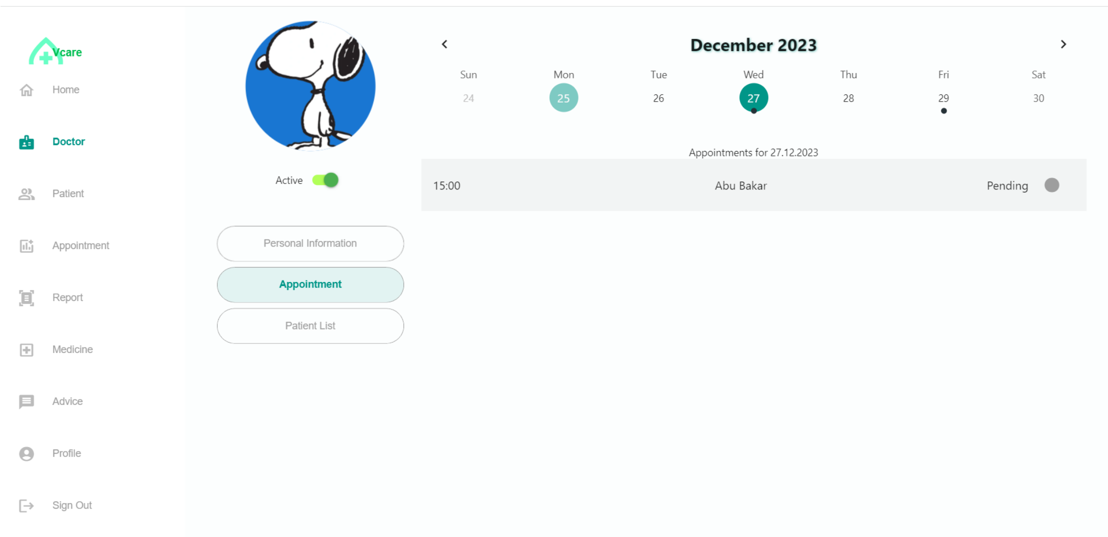


Figure 4.1.1.2.5: User Interface of View Doctor's Appointment

Under the section on appointments, the system displays a one-week range calendar and at the right bottom of the calendar is the list of appointments. The user can select any day from the calendar and the system will automatically display the appointment of this doctor under the selected day. By clicking on the appointment list tile, the system will navigate the doctor to a follow-up page for this appointment.

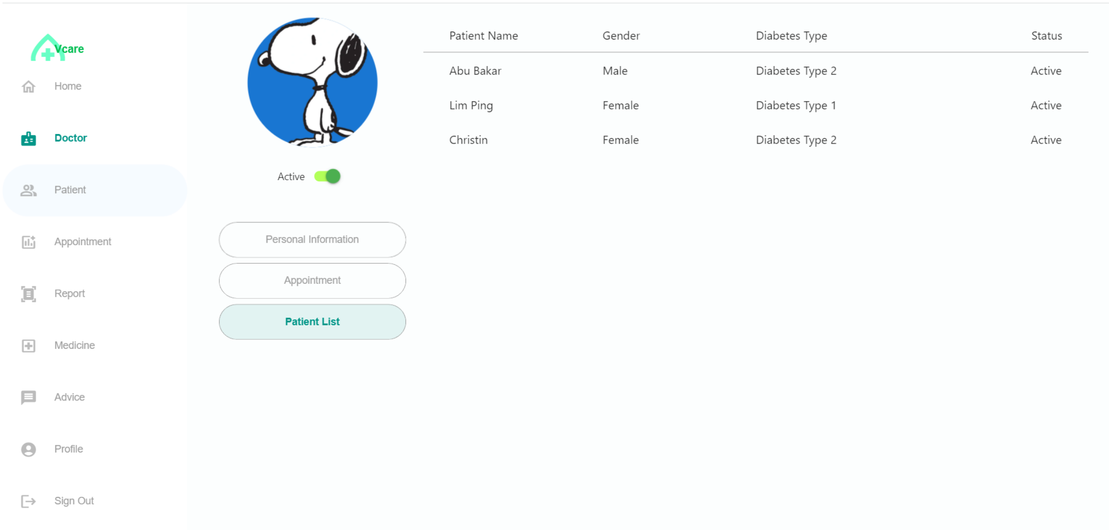


Figure 4.1.1.2.6: User Interface of View Doctor Active Patient List

Within the patient list section, all the patients under this doctor will be displayed. The user can select any record from the list to view the detailed information of the selected patients.

4.1.1.3 Patient Module

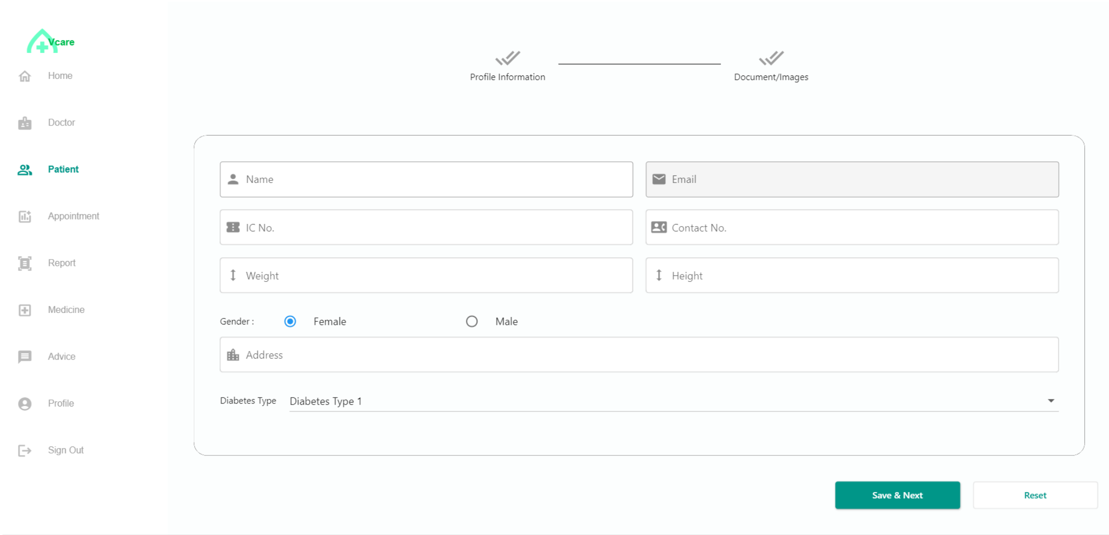


Figure 4.1.1.3.1: User Interface of Add Patient I

The add patient page is only accessible by the doctor. The form for adding new patients is divided mainly into three sections. In the first session, the system will request the user to enter the patient’s profile information, which includes name, IC, email, contact, weight, height, address and diabetes type. Above the form displays the progress bar that provides information to the user, showing that currently they are in the first step of adding a patient. The two buttons below the form, “Save & Next” will temporarily store the data entered when being clicked, allowing data to be restored when the users come back to this page after proceeding to the next step of adding a new patient. Whereas the button “Reset”, will clear off every field, returning an empty form to the user.

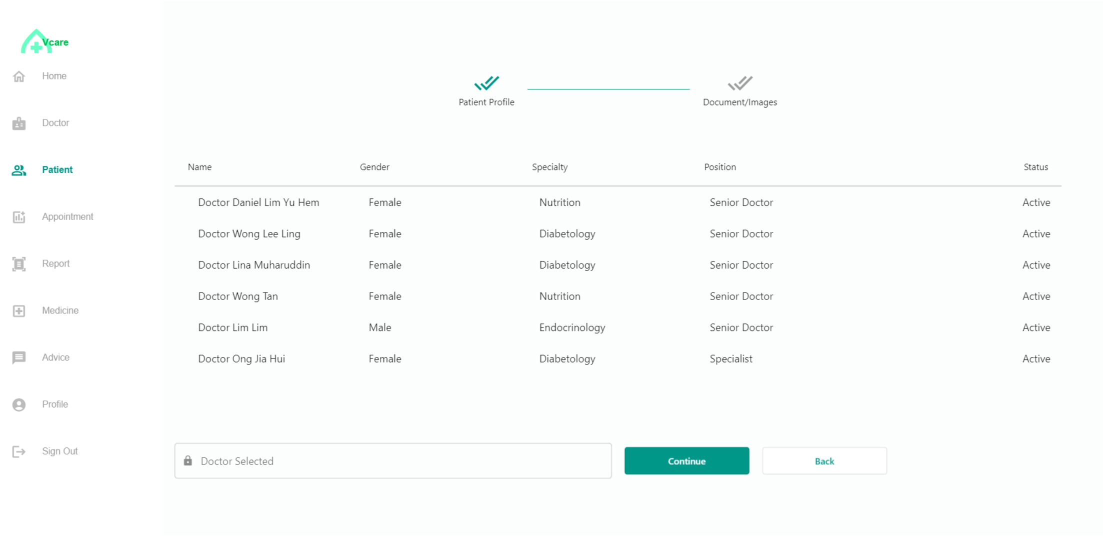


Figure 4.1.1.3.2: User Interface of Add Patient II

In section II, the doctor will have to assign a doctor to the patient they are currently registering by selecting a doctor from the list displayed. The list includes only active doctors. To notify the user about their progress on the current operation, the system will display the title of the previously completed section “Personal Information” in green. The system will then proceed to section III when the user clicks on ‘Continue’ and the ‘Back’ button will navigate the user to the previous page, the patient’s profile information page.

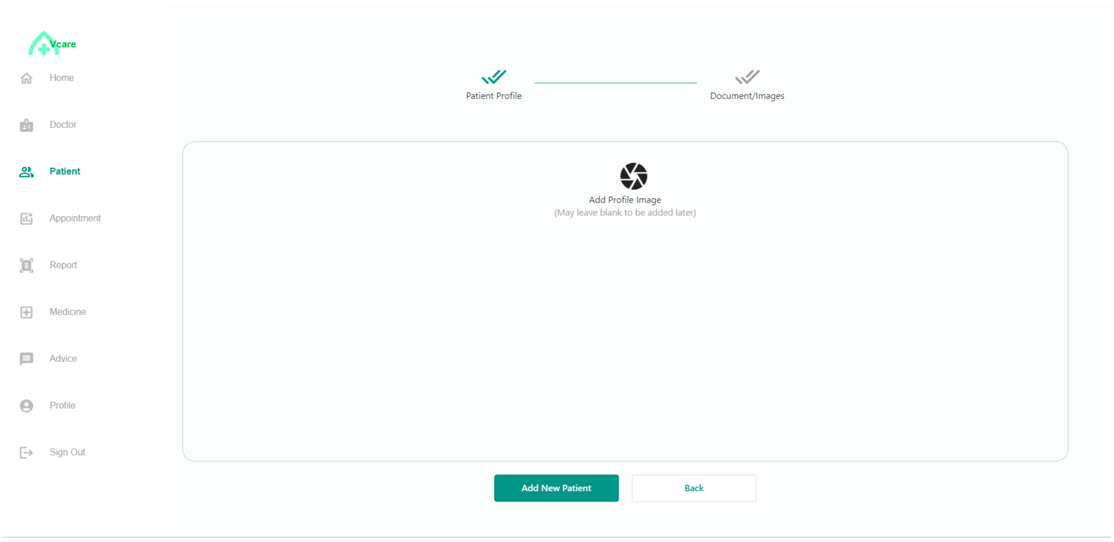


Figure 4.1.1.3.3: User Interface of Add Patient III

Section III of the patient adding includes an action of uploading the patient's profile image. In this step, the user can opt to skip the profile image uploading by directly clicking on the "Add New Patient" button, this will immediately initiate the patient adding action into the system although without an image. If the users want to include a profile image when adding a new patient, they can click on the section labelled "Add Profile Image" and select the image from the computer. The progress bar above indicates that the adding patient has partially completed. The "Back" button beside the "Add New Patient" button allows the users to navigate to the previous step, the doctor's assignment, if modification is required.

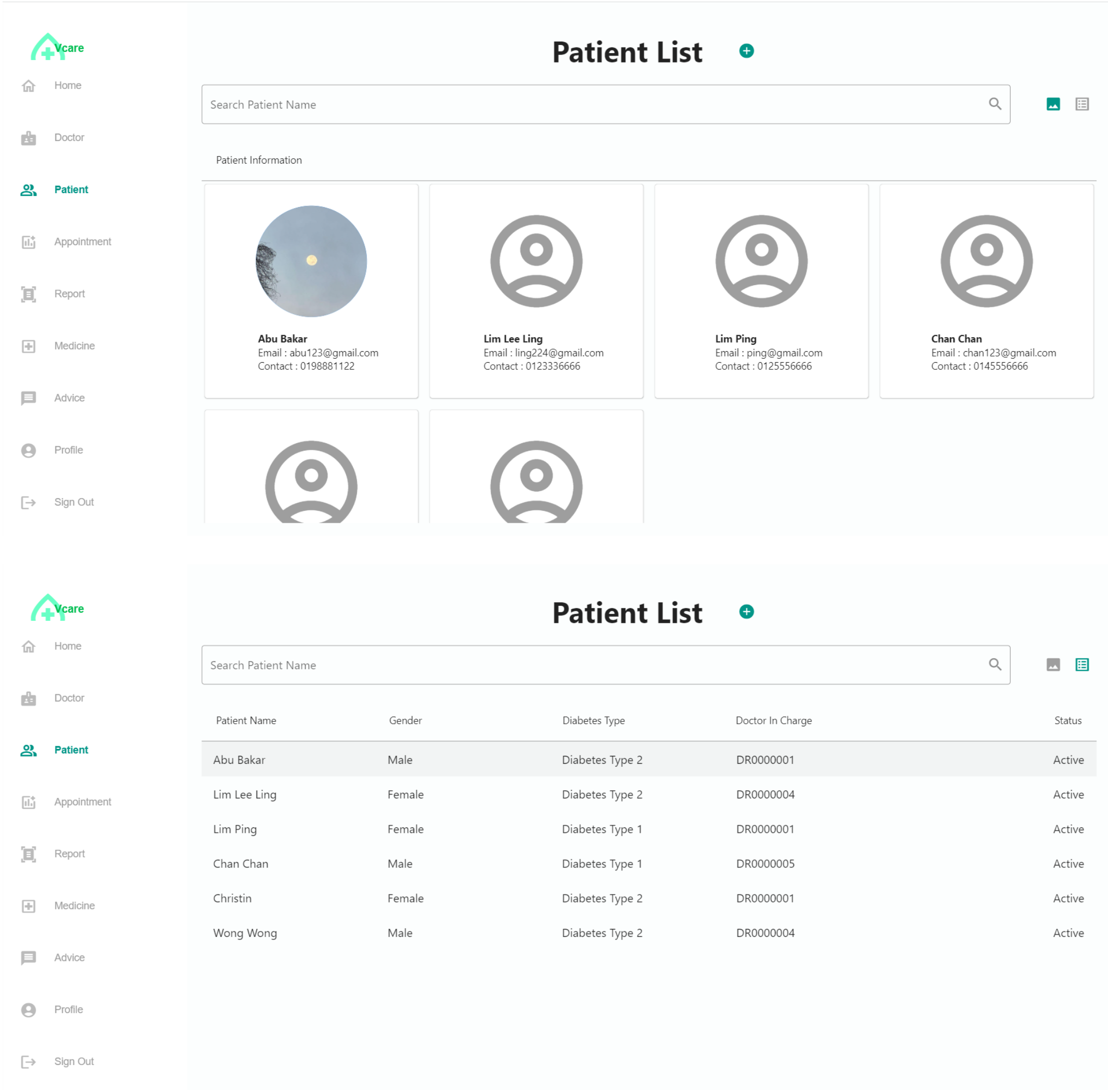


Figure 4.1.1.3.4: User Interface of View Patient

On the patient's main page, all the patient's information in the system will be displayed. The user can use the search box to search for certain patients. To initiate the search function, the user will have to enter the search keywords. This search box will search through the patients' names, such that when 'Lim' is searched, all the patients with 'Lim' within their name will be displayed. Not only that, the user can opt for different methods in displaying the information, the list view where information is displayed in rows and the details view where the patients' profile images are displayed. By clicking on the header of the patient list, the patient information will be sorted based on the selected criteria. By selecting the patient record from the list, the system will redirect the user to the detailed information of the patient selected.

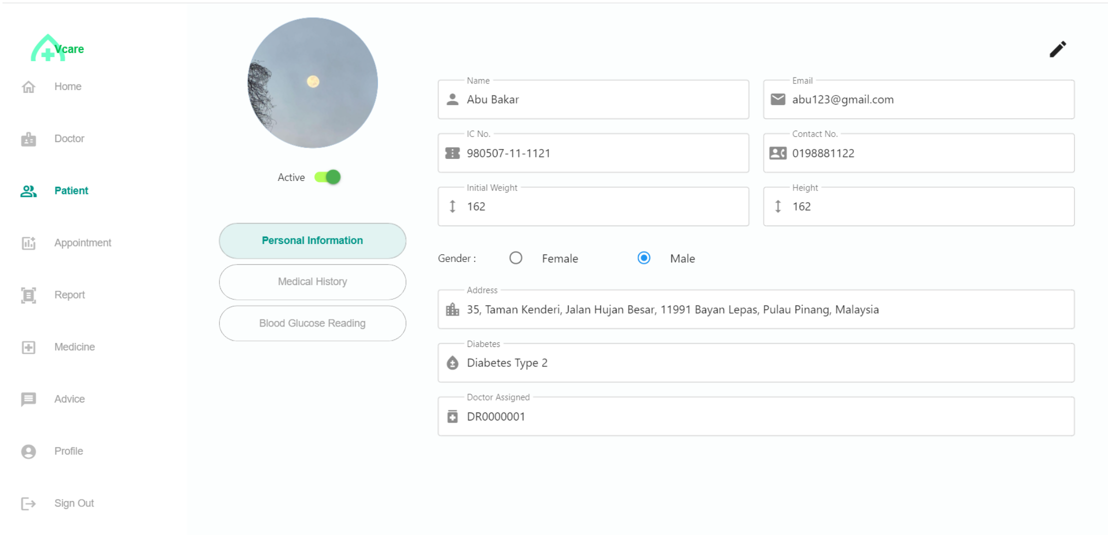


Figure 4.1.1.3.5: User Interface of View Patient Personal Information

The patient information will be divided into three sections, personal information, medical history and Blood Glucose Reading. Under the personal information section, information such as the patient's name, contact number, IC, email and so on. The user only can edit the information on this page by clicking the pencil button to activate the edit mode. While the edit mode is activated, the email for the patient remains uneditable as the email serves the login purpose. The user can toggle the 'Active' switch to deactivate this particular patient when his/ her account is no longer active.

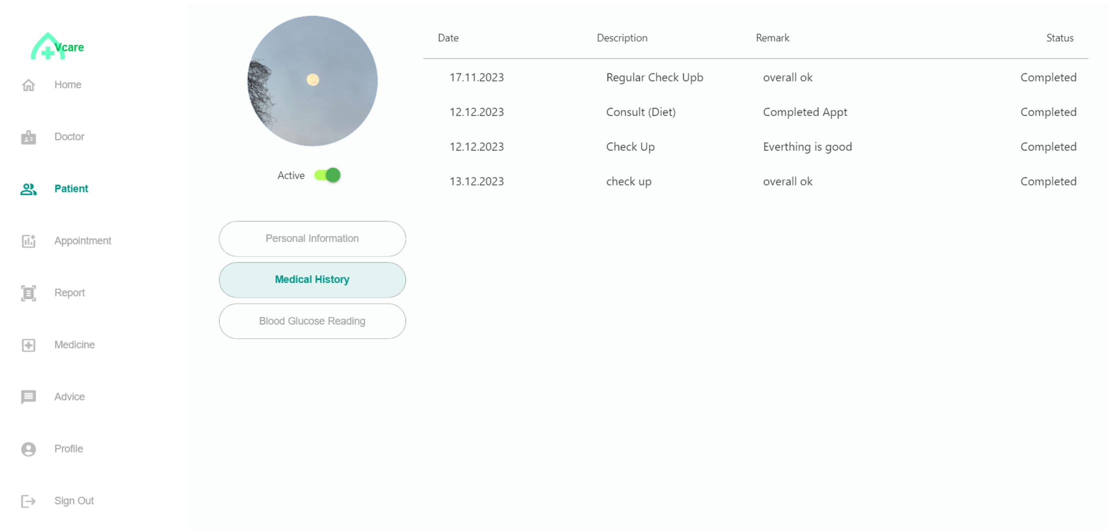


Figure 4.1.1.3.6: User Interface of View Patient Medical/Follow-Up Record

The patient's medical follow-up records are located in the second section. Within this page, all the medical/follow-up records are listed for easy access. Users have the option to click on any specific record to view its details in-depth.

Figure 4.1.1.3.7: User Interface of View Patient Blood Glucose

The doctors can navigate to the third section of the View Patient page to monitor the health of their patients. Within this page, the doctor can retrieve the blood glucose readings recorded by the patients. The data will be visually represented through informative line graphs, allowing the doctor to quickly grasp trends and changes over time. The doctor can filter the displayed time range to focus on a specific period. In cases where the data indicate potential risks or require attention, the doctor can leave a feedback message to notify the respective patients

4.1.1.4 Appointment Module

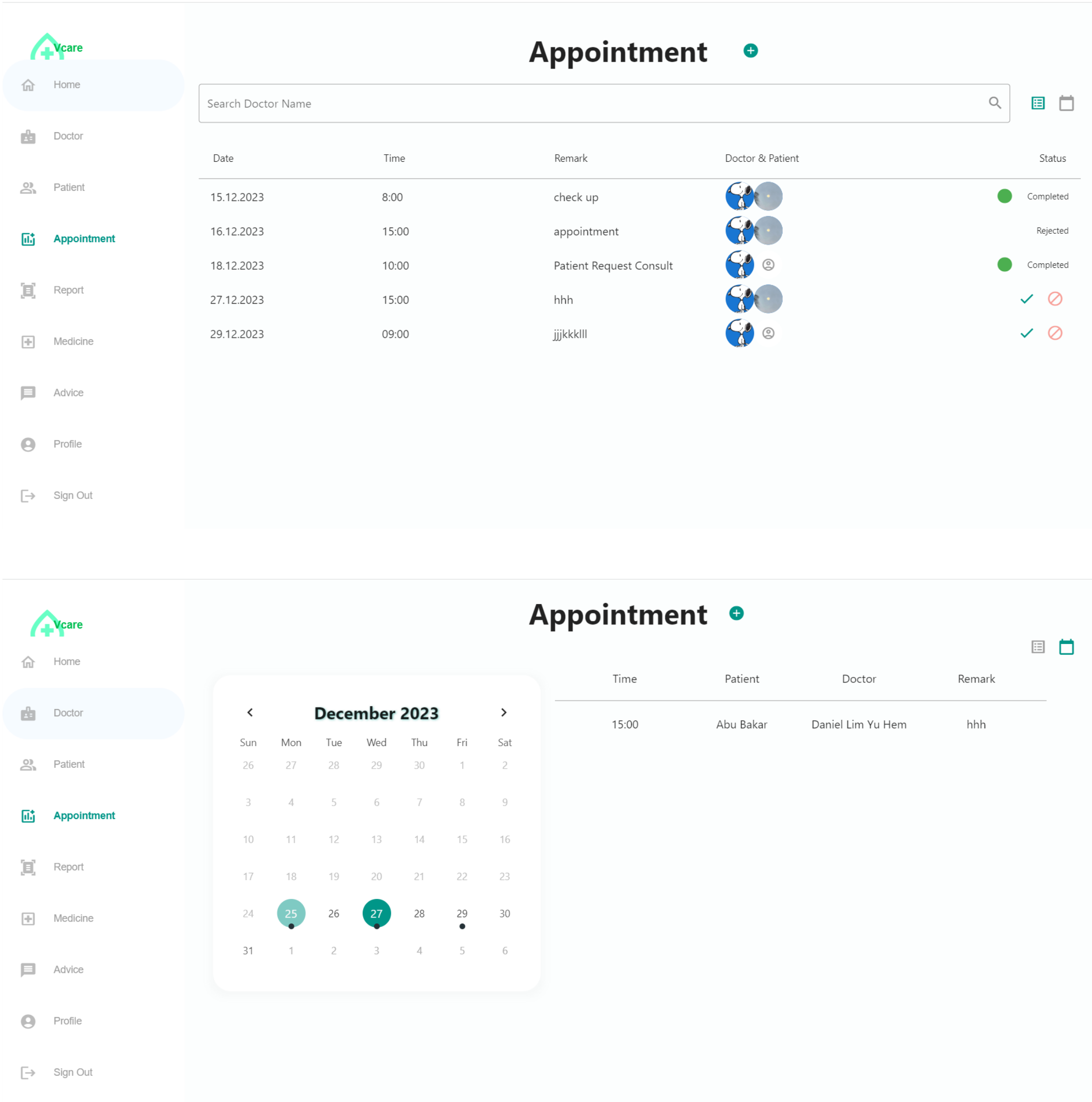


Figure 4.1.1.4.1: User Interface of View Appointment

The appointment main page displays all the appointments of that particular doctor for this month. However, the user may view the appointment list of the other doctors by searching their names in the search bar. On this appointment's main page, the doctor can approve or reject their pending appointment by clicking on the respective icon of the particular row. The appointment status will be updated once action is taken. When the doctor attempts to navigate into the appointment's detail page, they will only succeed if the appointment has been approved and the date matches the date today or it has been completed. If the appointment is still pending approval or on a future date, the user will be prompted with an error message. The user may also consider the appointment to be displayed with a calendar view by changing the display format.

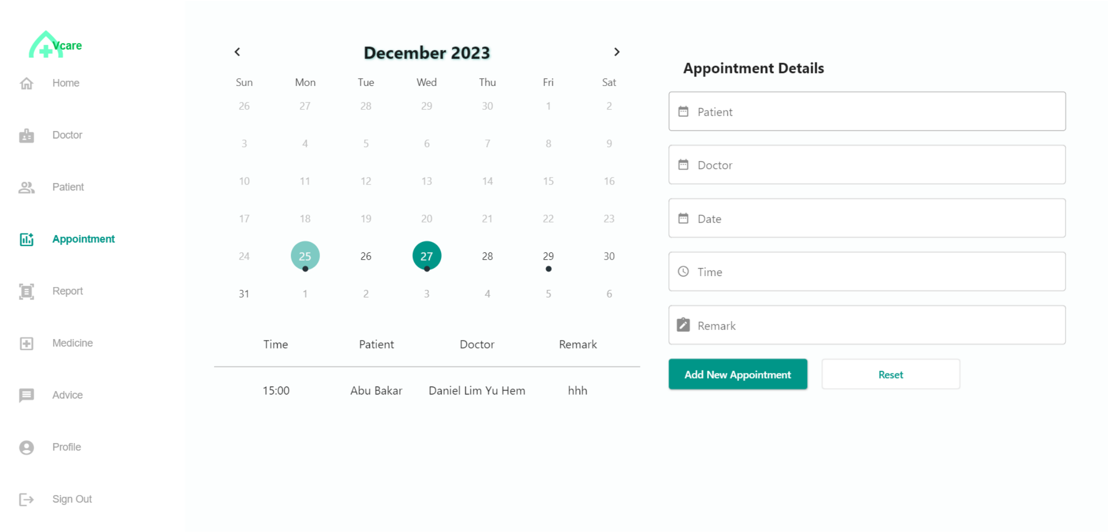


Figure 4.1.1.4.2: User Interface of Add Appointment

When the user opts for an appointment, they will be prompted with this appointment form. The calendar on the side acts as a reference for the doctor to check for available schedules. While the user clicks on the date or time field, the respective date picker and time picker will be prompted to ensure accurate input. When the user clicks on the patient or doctor list to reserve an appointment for them, the respective doctor list and patient list will be prompted for selections.

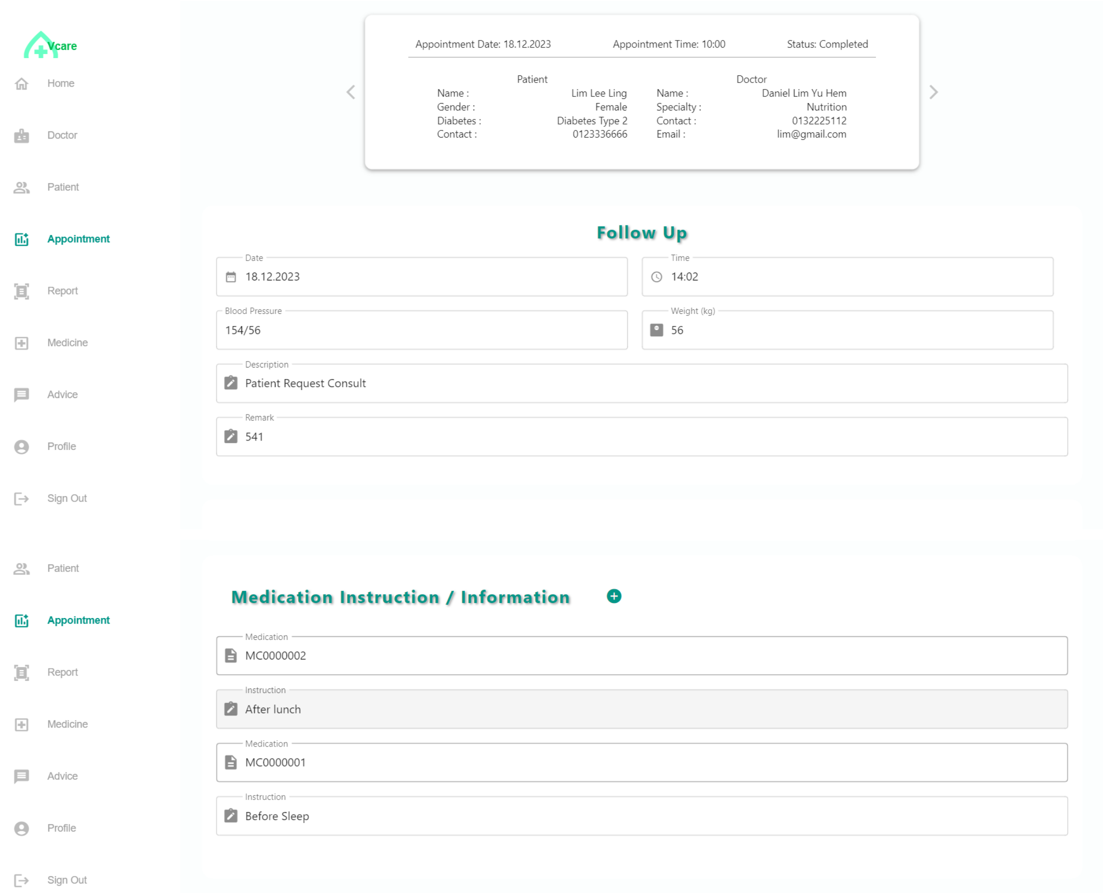


Figure 4.1.1.4.3: User Interface of Update Appointment Details

The doctor will be redirected to this page once they select any record from the appointment list. Information about the patients' appointments including the patient's name, appointment date, and time will be displayed in the upper part. The page consists of a follow-up form that allows doctors to record the patient's blood pressure level and weight during the consultation, adding remarks for future reference. Furthermore, there will be a separate form to document the medications prescribed to the patients. If more than one medication is required, the doctor can click on the add button to add in more medications.

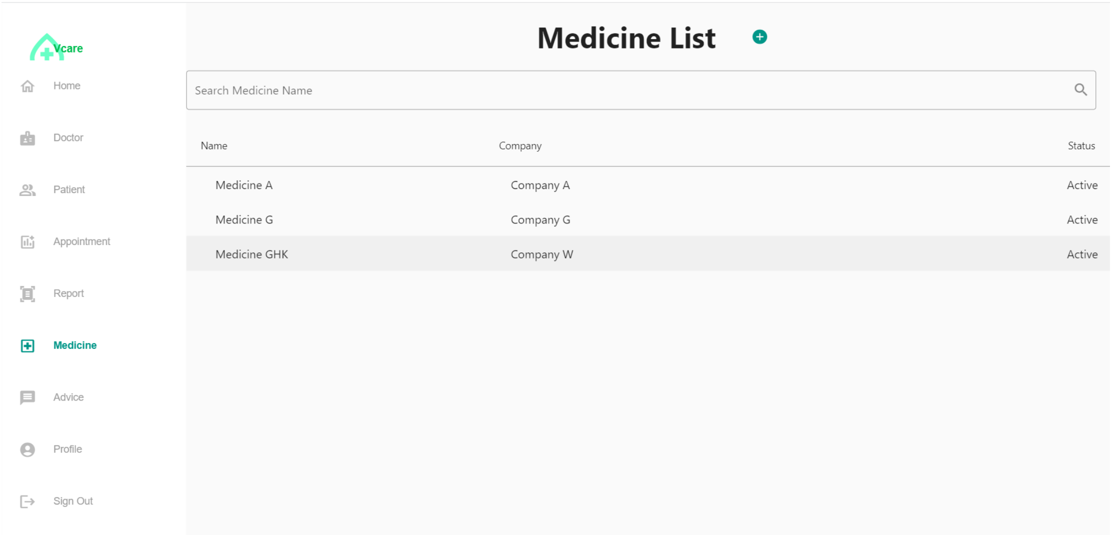


Figure 4.1.1.4.4: User Interface of View Medicine

The medicine main page displays all the medicine records currently stored in the system. It allows the user to perform a search on the medicine name through the search bar.

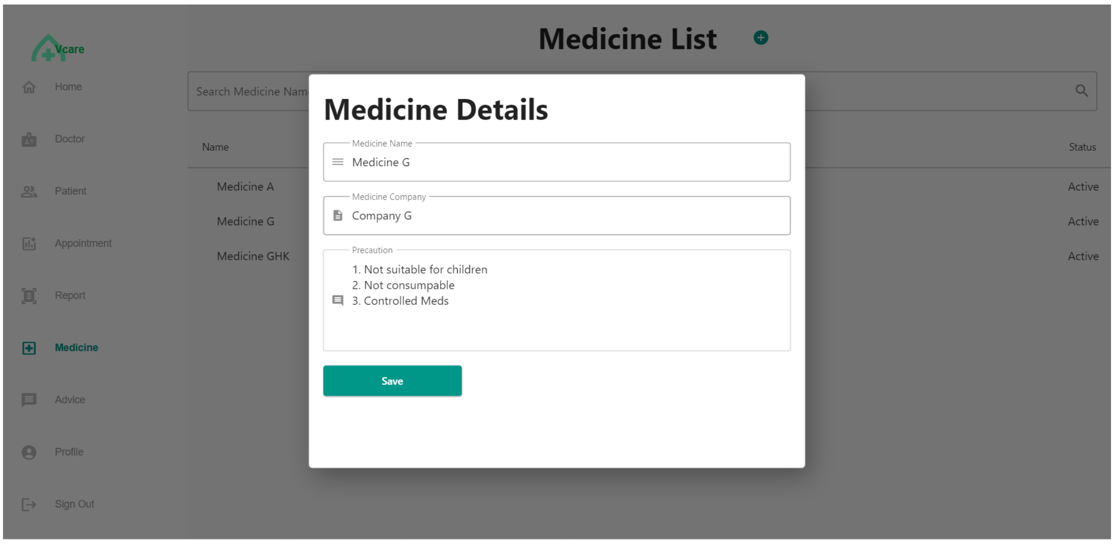


Figure 4.1.1.4.5: User Interface of Add/ Edit Medicine Details

To add a new medicine, the user can click on the add button. If the user wants to edit the information on existing medicine, they can click on the record to be edited on the list and make the necessary modifications. A similar pop-up box will appear for both add and edit actions for the medicine.

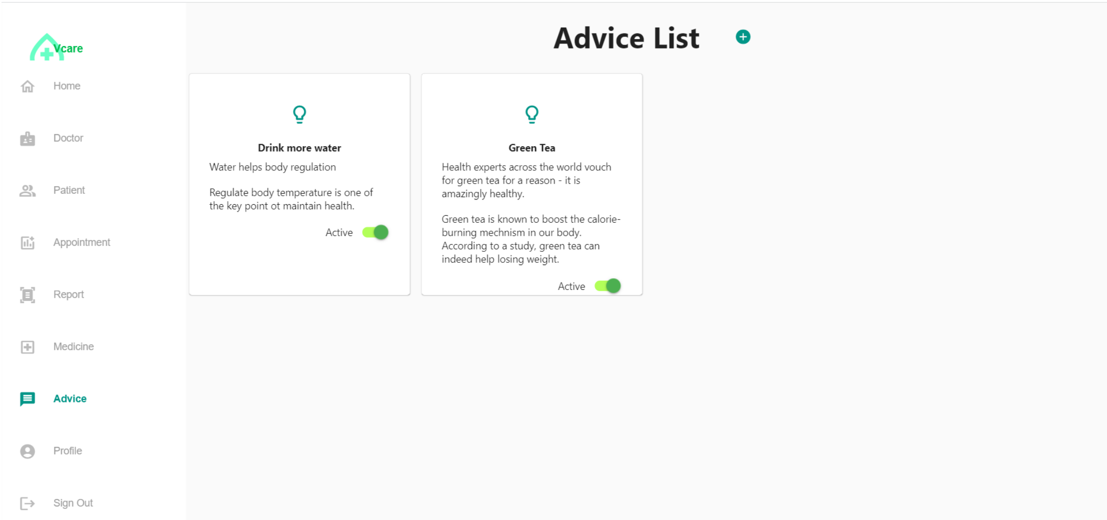


Figure 4.1.1.4.6: User Interface of View Advice

The advice main page displays all the information which the doctors wish to share with the patients, allowing the patients to access some additional information regarding their health management. If the doctor would like to remove some information from being accessed, they can deactivate that particular advice.

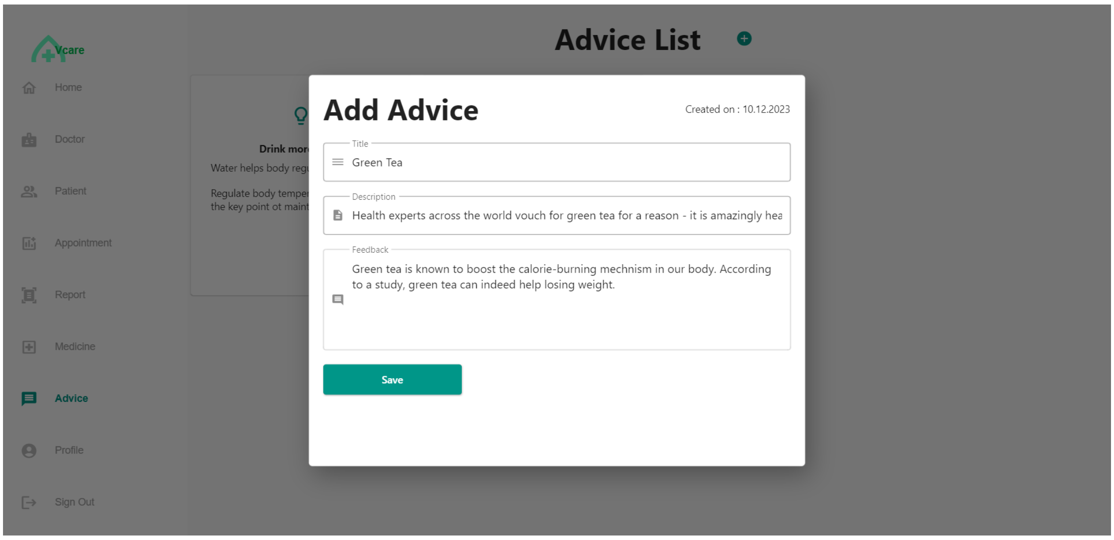


Figure 4.1.1.4.7: User Interface of Add/ Edit Advice

To add a new piece of advice, the user can click on the add button. If the user wants to edit the information on an existing piece of advice, they can click on the record to be edited on the list and make the necessary modifications. A similar pop-up box will appear for both add and edit actions for the advice.

4.1.5 Report Module

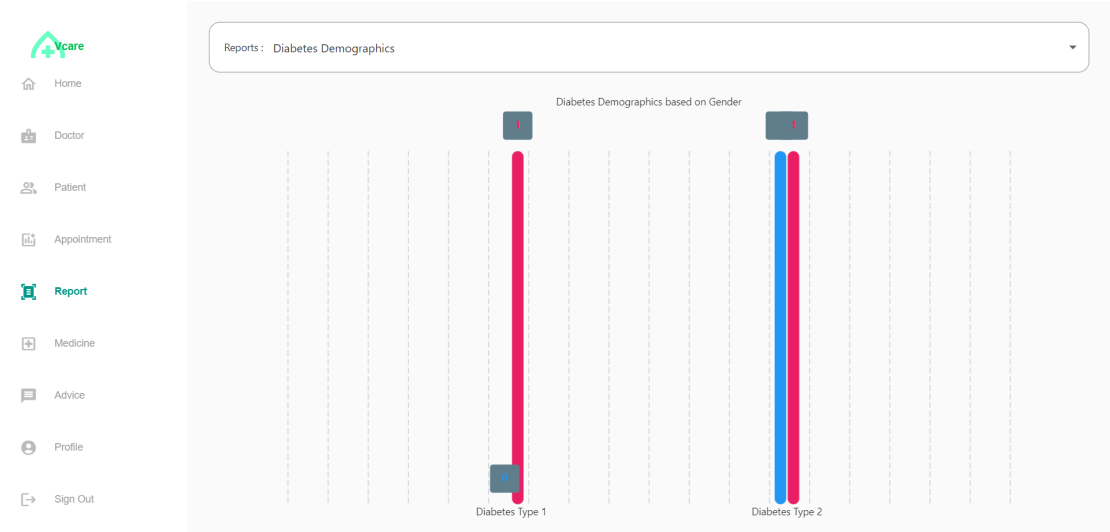


Figure 4.1.1.5.1: User Interface of Report I

The user can navigate to this report by selecting the ‘Report’ option from the navigation sidebar and selecting the respective report title from the drop-down options. This report aims to provide an overview of the prevalence of diabetes patients based on gender. It will calculate the total number of patients who have registered in the system based on their diabetes type.

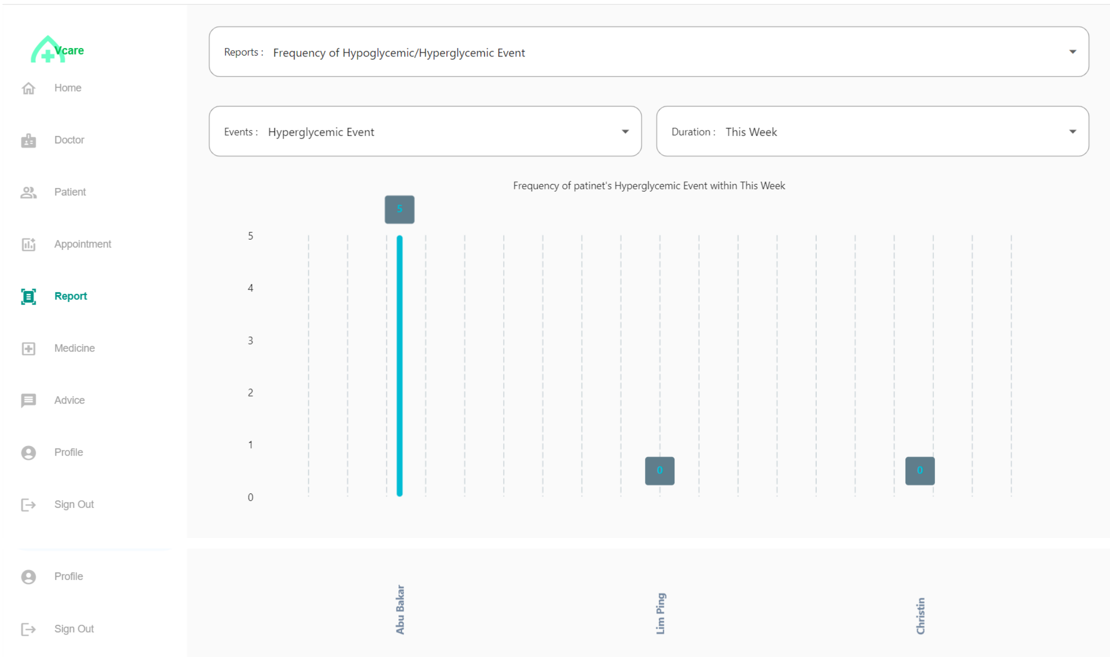


Figure 4.1.1.5.2: User Interface of Report II

The user can navigate to this report by selecting the ‘Report’ option from the navigation sidebar and selecting the respective report title from the drop-down options. The user can further select the type of event and duration before generating the report. This report focuses on analysing the frequency of both hypoglycemic (low blood sugar) and hyperglycemic (high blood sugar) events experienced by patients over a specific period. It presents an overview of the frequency of hypoglycemic events and hyperglycemic events.

4.1.2 User Interface (Mobile Application)

4.1.2.1 Login Module

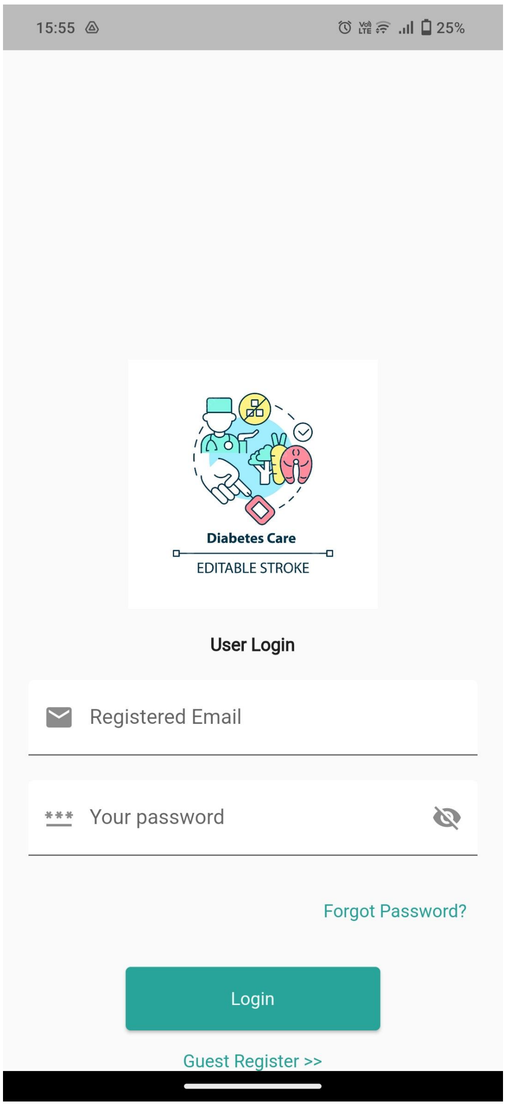


Figure 4.1.2.1.1: User Interface of Login

The login page in the mobile version is exclusively accessible to patients who have been registered by the healthcare centre's admin in the system. This ensures that only authorised patients can use the mobile application. If the patient forgot their login password, they can utilise the Forgot Password feature to reset their password. However, those unregistered patients are also being welcomed to use the application. They can proceed with registration by clicking on the 'Guest Register' link.

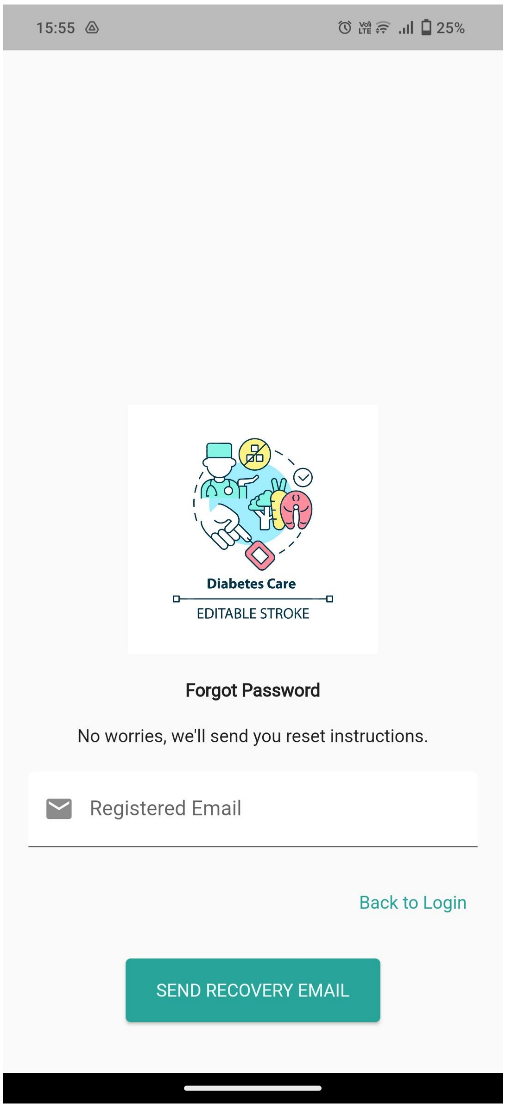


Figure 4.1.2.1.2: User Interface of Email Verification

The email verification page is designed to ensure security when patients request password recovery. On this page, the patients need to enter their email addresses. If the entered email address is valid and matches an existing user record in the database, the system will generate an email attached with a link to reset the password. The user may proceed with the password reset through the link given.

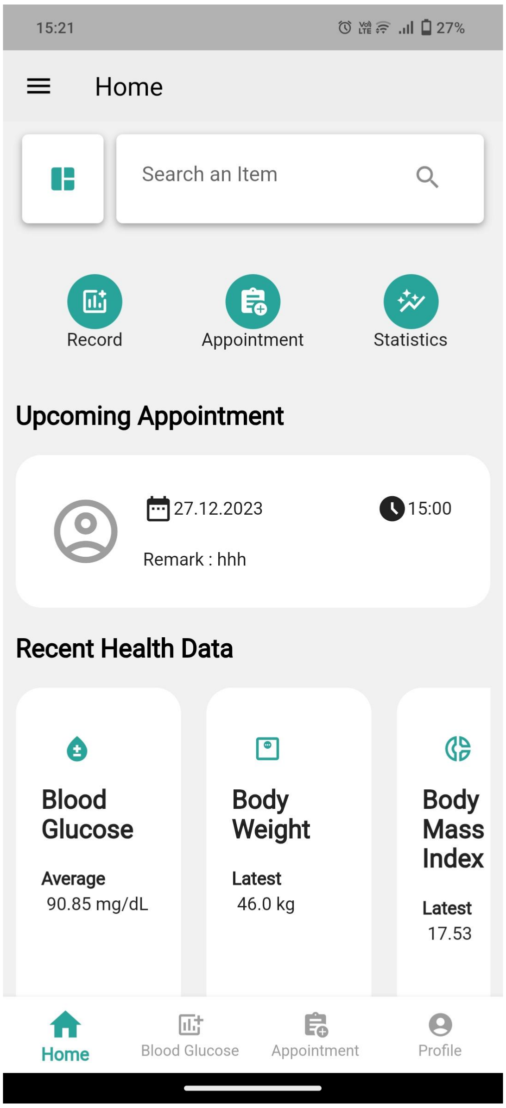


Figure 4.1.2.1.3: User Interface of Homepage

The patient will be redirected to the application homepage once they log in successfully. On the homepage, the system shows a summary of their current health status by calculating the average blood glucose level based on the data recorded by the patients over the past 7 days. It will also display other information such as the latest body weight and latest body mass index based on the data they have recorded. This data can significantly provide information about their health status. Not only that, there will be a section displaying the upcoming appointment information to notify the user regarding the appointment and some shortcut keys for the features provided by the application.

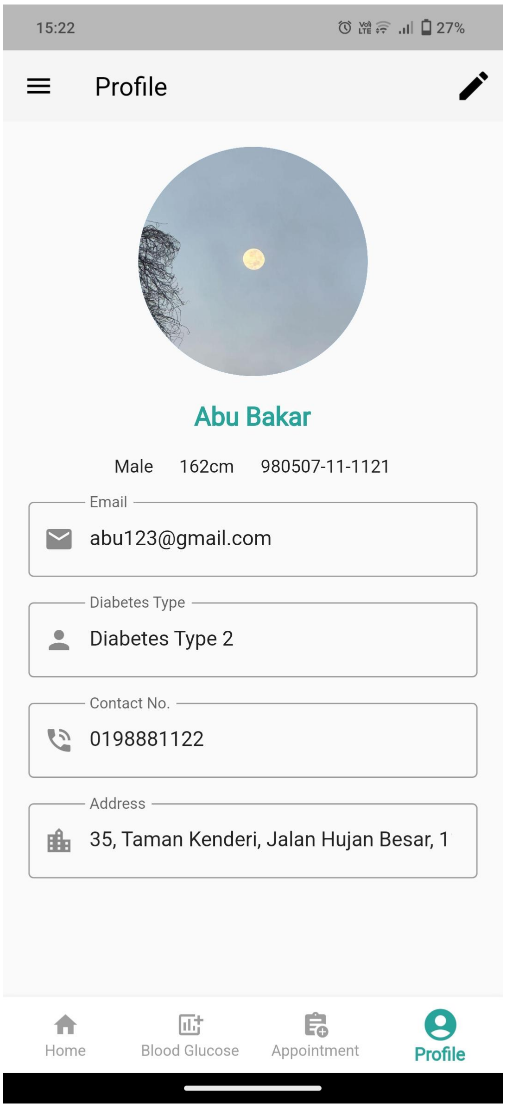


Figure 4.1.2.1.4: User Interface of Profile

The information of the patient will be displayed on this page. The patients can only edit their information after activating the edit mode by clicking on the pencil icon, but some fields such as the IC number, and email address are not editable. They can also change their profile pictures by initiating the uploading function.

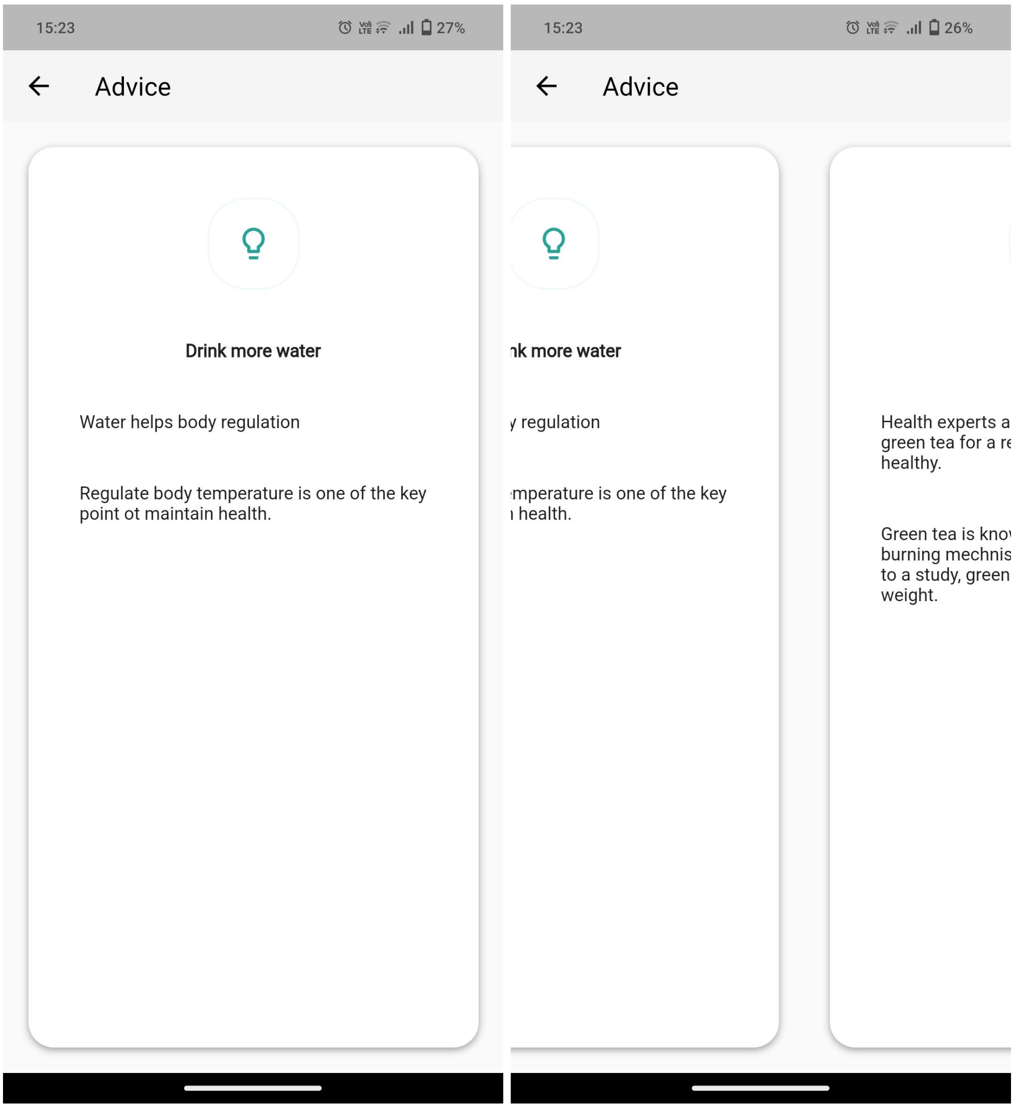


Figure 4.1.2.1.5: User Interface of View Advice

The advice page serves as a centralised platform for healthcare centres to share information with the user. As the healthcare provider has more professional knowledge, they can share this information as additional advice through this platform to the user, aiding them to have better knowledge. The user can scroll horizontally to get more advice from the healthcare providers.

4.1.2.2 Blood Glucose Module

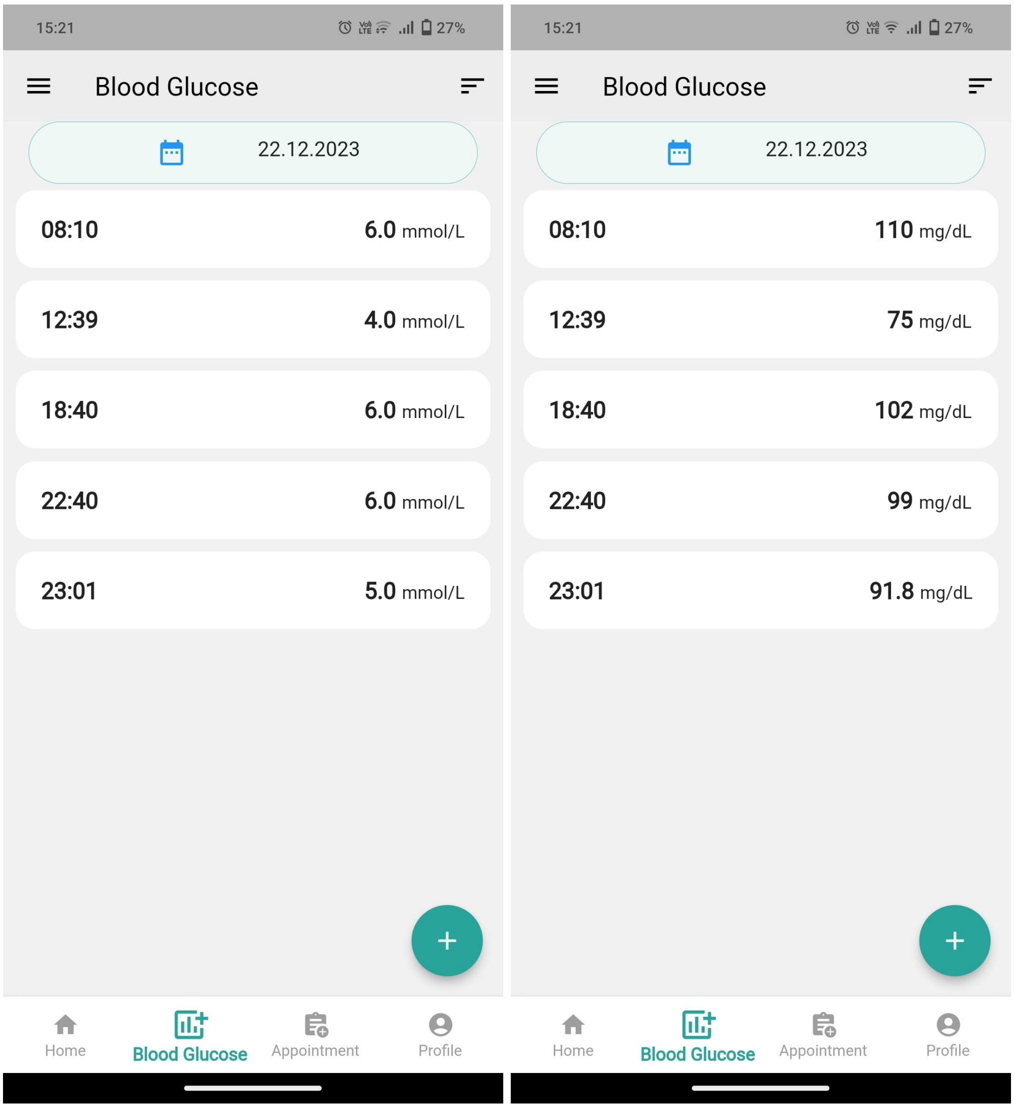


Figure 4.1.2.2.1: User Interface of View Record List View

In the main page of blood glucose, the system initially displays today's record in the default measuring init. The user may change the display unit through the top right icon. Furthermore, the user may view the blood glucose records of another day by clicking on the current date and navigating through the date picker. The records of the selected day will then be displayed accordingly. The add button at the right bottom corner of the page will redirect the user to create a new record.

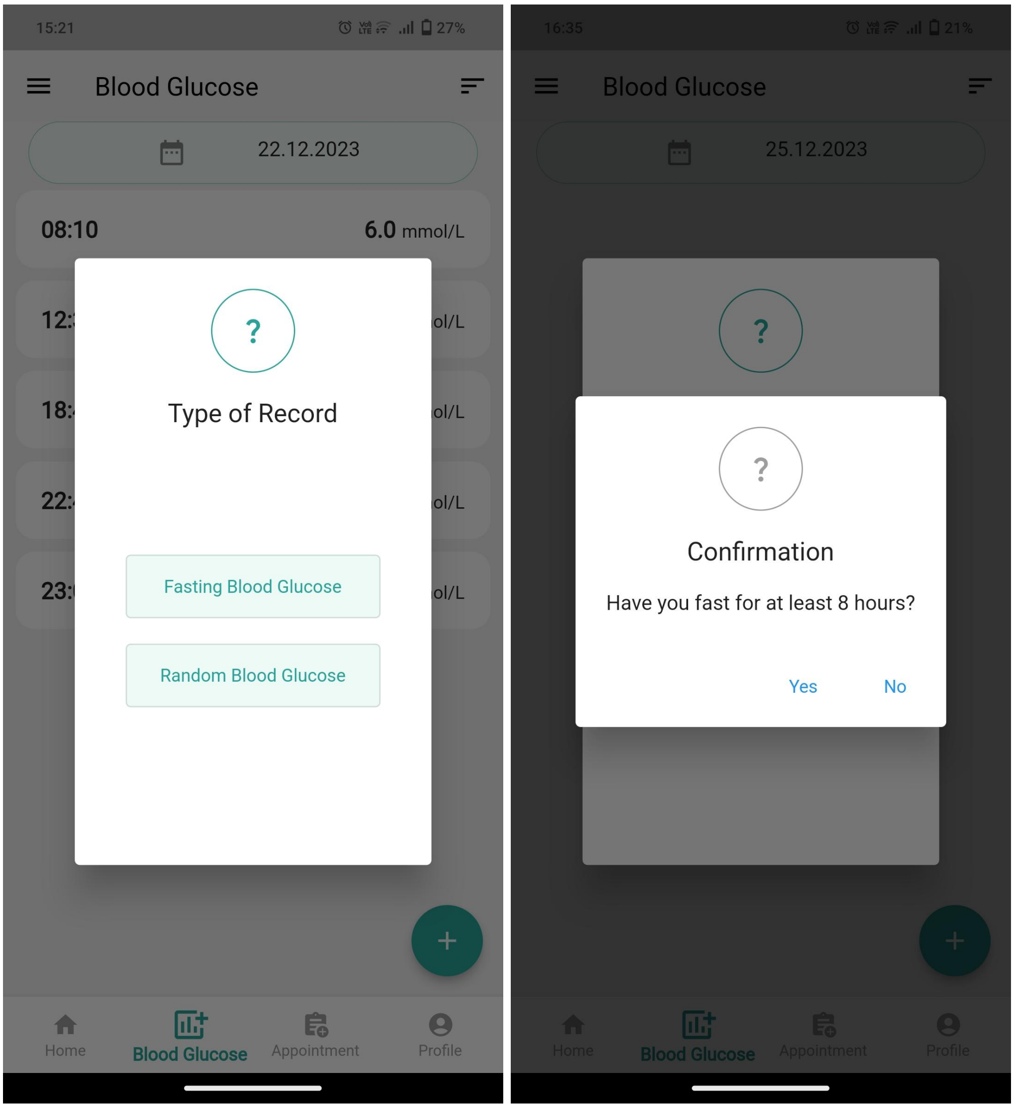


Figure 4.1.2.2.2: User Interface of Add Record Type

When the user clicks on the add button in the blood glucose main, the system will prompt a dialogue asking the type of record the user tried to make. If the user opts for the random blood glucose record, the system will direct the user to the add blood glucose page. However, if the user selects the fasting blood glucose type, the system will prompt the user with an 8-hour fasting confirmation before prompting them to the page to add a new blood glucose record.

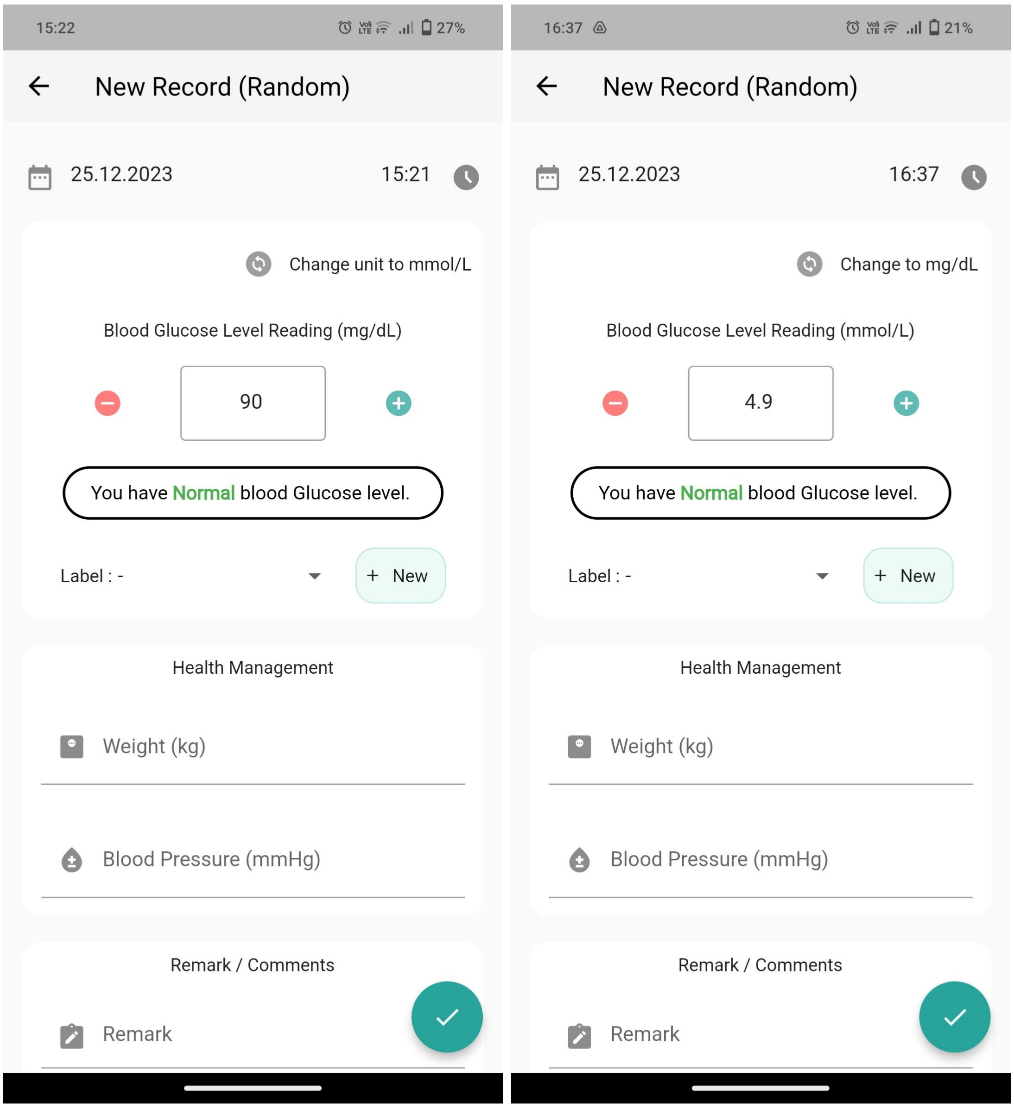


Figure 4.1.2.2.3: User Interface of Add Blood Glucose Record

On this page, the patients can record their measured blood glucose levels. The system will automatically capture the measured date and time. The default measuring unit of the blood glucose is ‘mm/dL’, if the user would like another measuring unit, ‘mmol/L’, they can change the measuring unit by clicking on the exchange symbol. A small message of the user’s current health will be displayed based on the inputted value. The “Label” field is used to provide additional information about the record, such as whether it is a fasting measurement, random measurement, measurement before lunch, measurement after lunch and so forth. The label is manageable by the patients. They can add additional labels according to their routine.

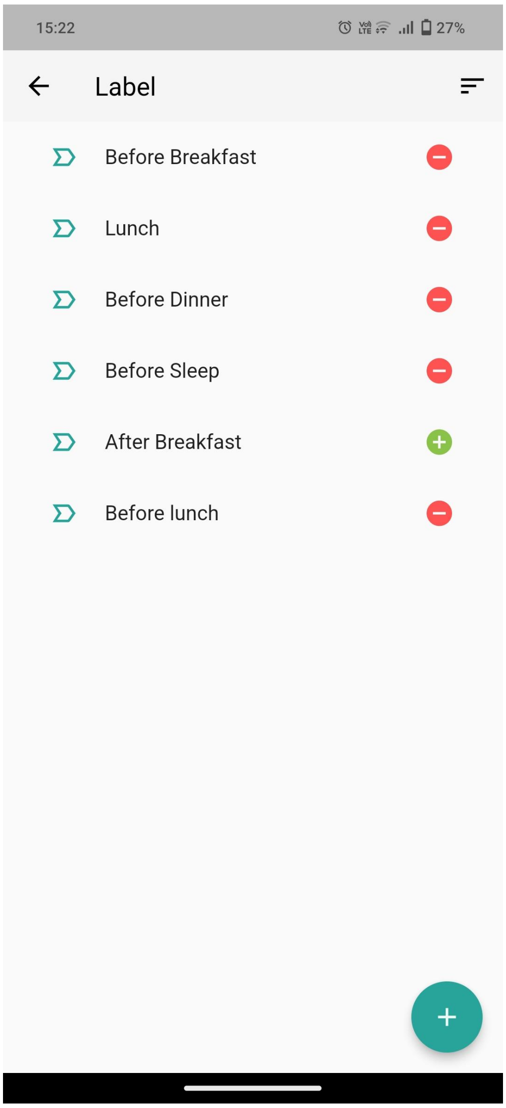


Figure 4.1.2.2.4: User Interface of Label

This page serves as the main page to manage the labels for the blood glucose reading records. The users can add a new label by clicking the add button below. They can perform sorting on the labels based on the description to have a better view. If the user would like to deactivate a label, they can simply click on the red deactivate button. Once the label is deactivated, it will no longer appear in the list of labels when adding a new blood glucose record. The users can make it appear again by activating it again.

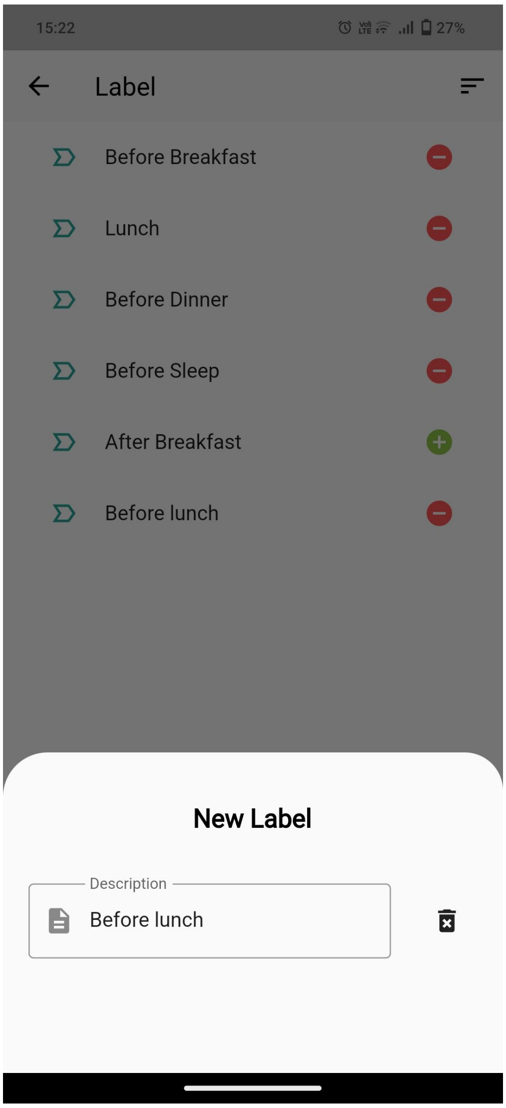


Figure 4.1.2.2.5: User Interface of Add Label

When the user initiates the add label functions, the system will prompt a new label bottom-up dialogue. The user can enter the label that they desire into the field and save the label.

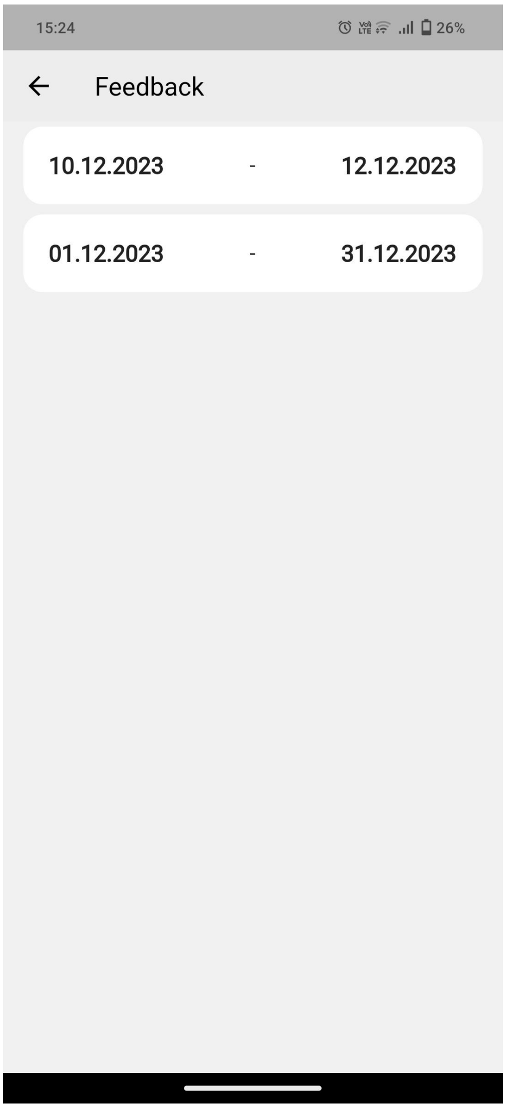


Figure 4.1.2.2.6: User Interface of Feedback

The doctors can make necessary feedback based on the reading recorded by the patient. Once feedback is added by the doctors, the patient will be able to retrieve the feedback from this page. This page displays a list of feedback duration that was made by the doctors.

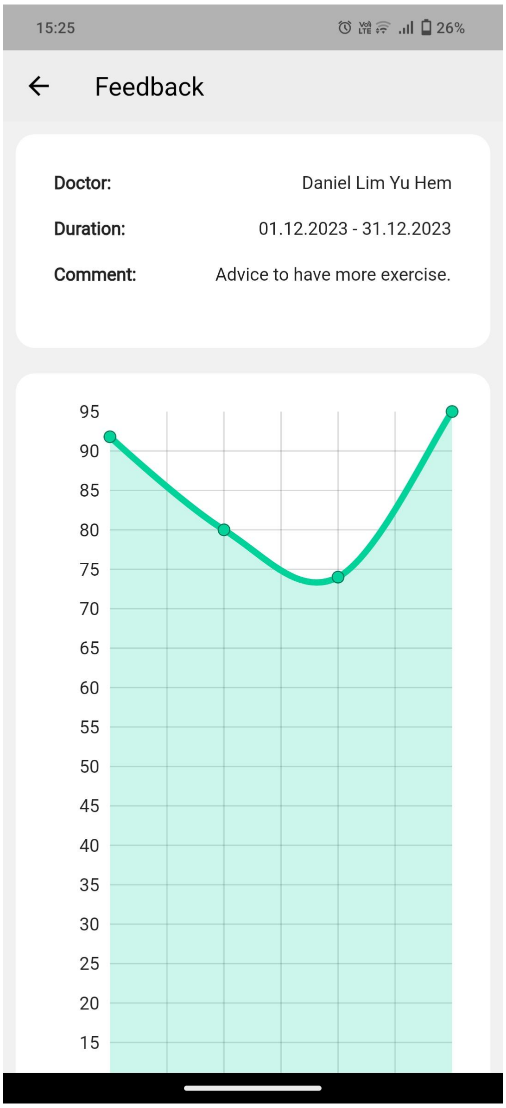


Figure 4.1.2.2.7: User Interface of Feedback Details

Based on the user's selection on the feedback main page, the system will prompt the relevant record. The detailed page of the feedback will display the information of the doctor who has made the feedback associated with the comment. The duration of the trend the feedback is referring to will also be included as one of the details. The blood glucose visualisation graph of the duration this feedback is referring to will be displayed below the information to enhance the understanding of the users.

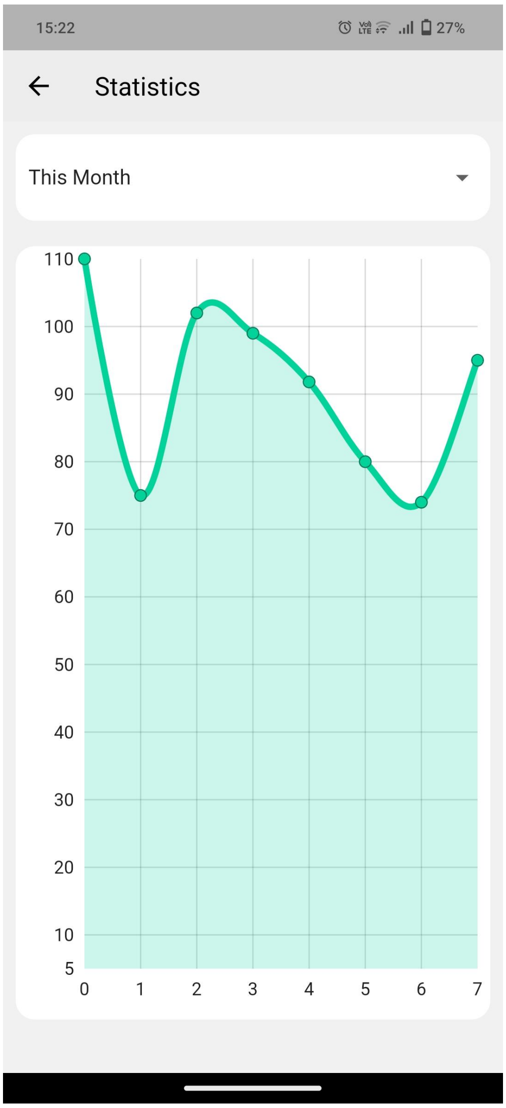


Figure 4.1.2.2.8: User Interface of Statistics

On this page, the user can visualise all blood glucose readings recorded by them. This statistics page allows the user to filter the data presentation on different time durations, allowing them to understand themselves better.

4.1.2.3 Appointment Module

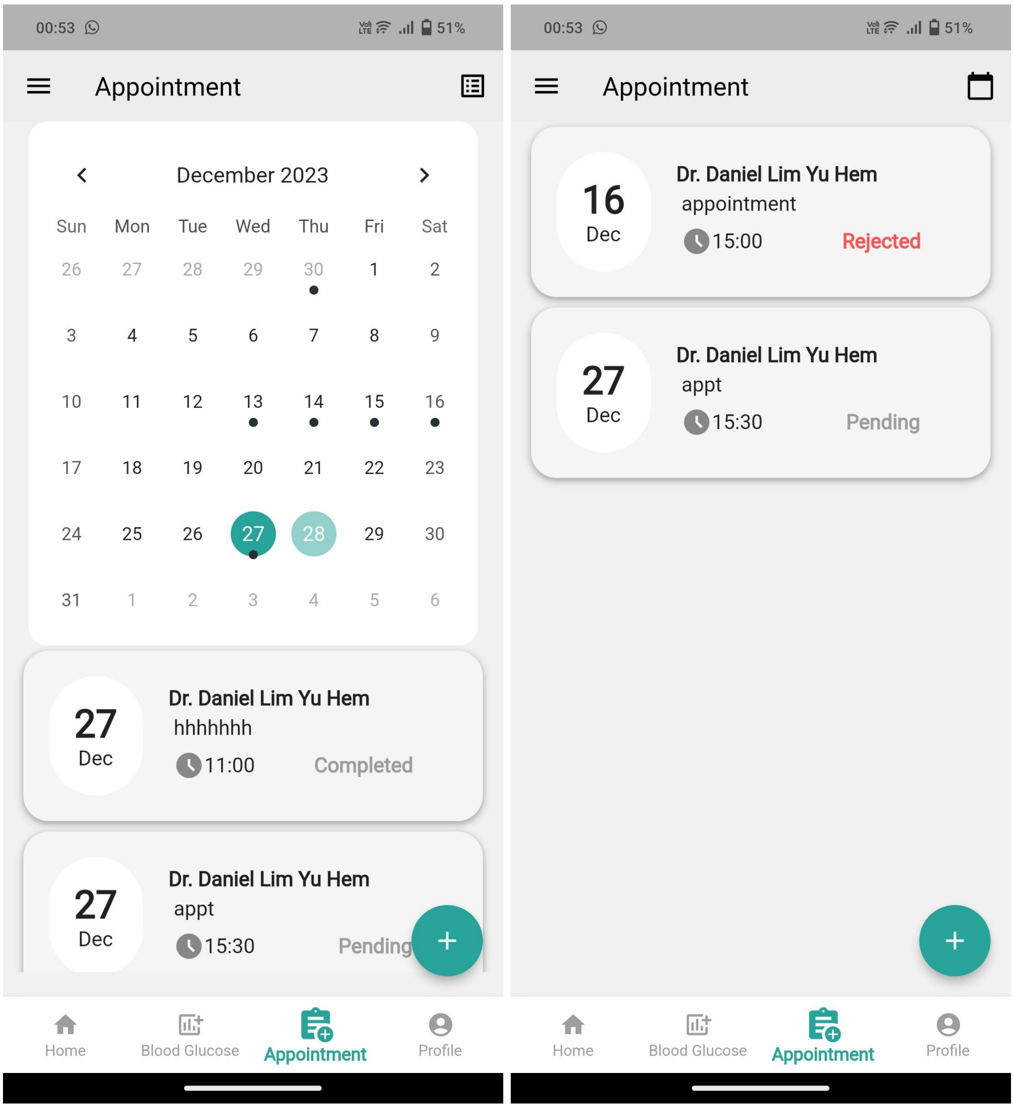


Figure 4.1.2.3.1: User Interface of Appointment

The appointment management page allows the patient to schedule an appointment with the doctor. They can also schedule new appointment records with their doctor by clicking the add button. Clicking the record will redirect them to the details of the selected appointment.

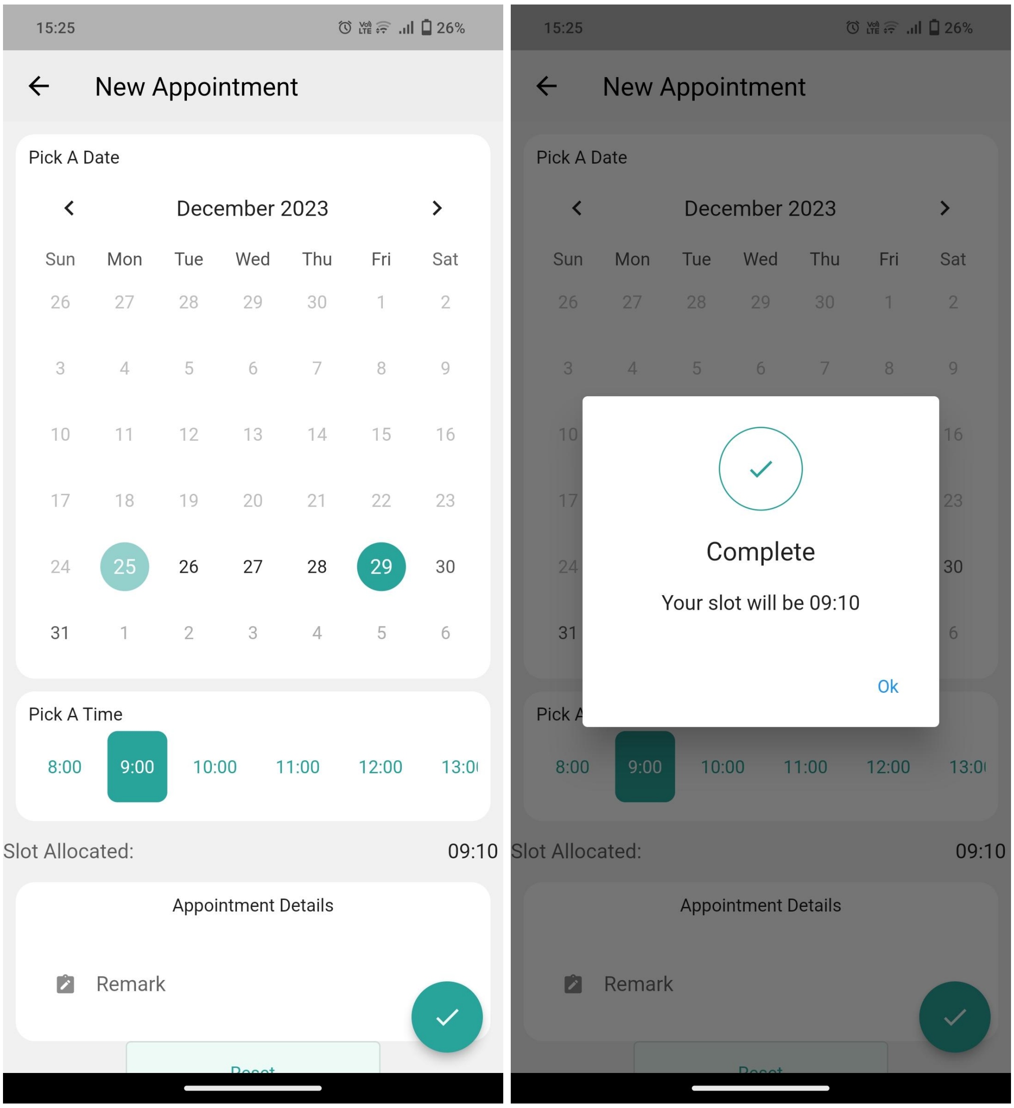


Figure 4.1.2.3.2: User Interface of Add Appointment

When the patient opts to add a new appointment, the system will display an appointment-adding page as above. The user will select the date from the calendar and the duration they desire. Once selected, the system will auto-assign a slot based on the duration selected. There will be a total of 6 slots available within one hour and if the particular slot has been fully booked, the system will display an error message.

Figure 4.1.2.3.3: User Interface of View Appointment

The user can view the appointment that they have reserved and make the necessary s or cancellations. They can modify the appointment that they made by activating the edit mode and cancelling through the icons on the top right. However, these modifications and cancellations are only applicable if the appointments are still pending and yet to be completed.

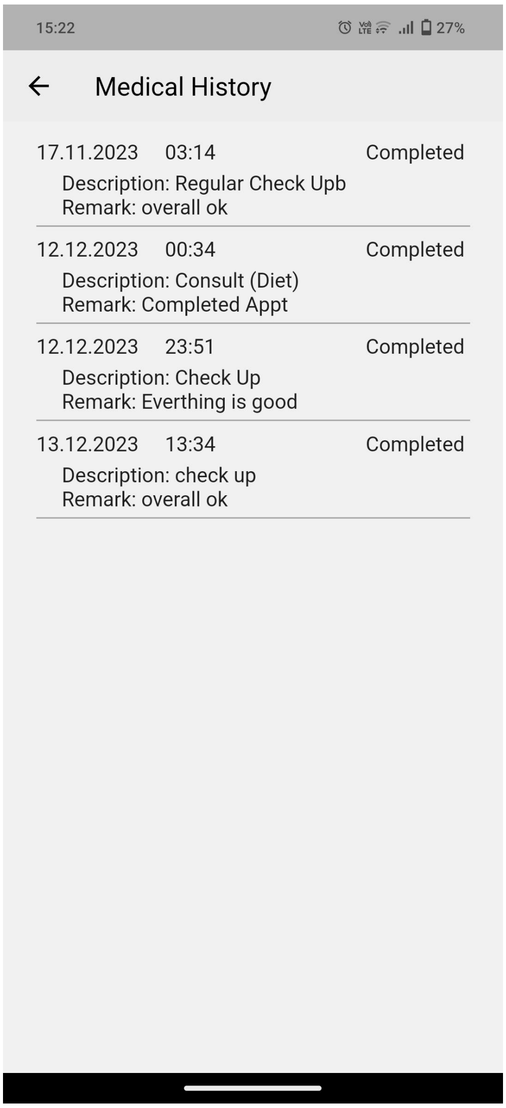


Figure 4.1.2.3.4: User Interface of Medical History

Once the appointment has been completed, the record will no longer appear on the appointment page. It will turn up as a medical history record that displays the information of that particular follow-up. In the medical history main page, it displays the list of medical history.

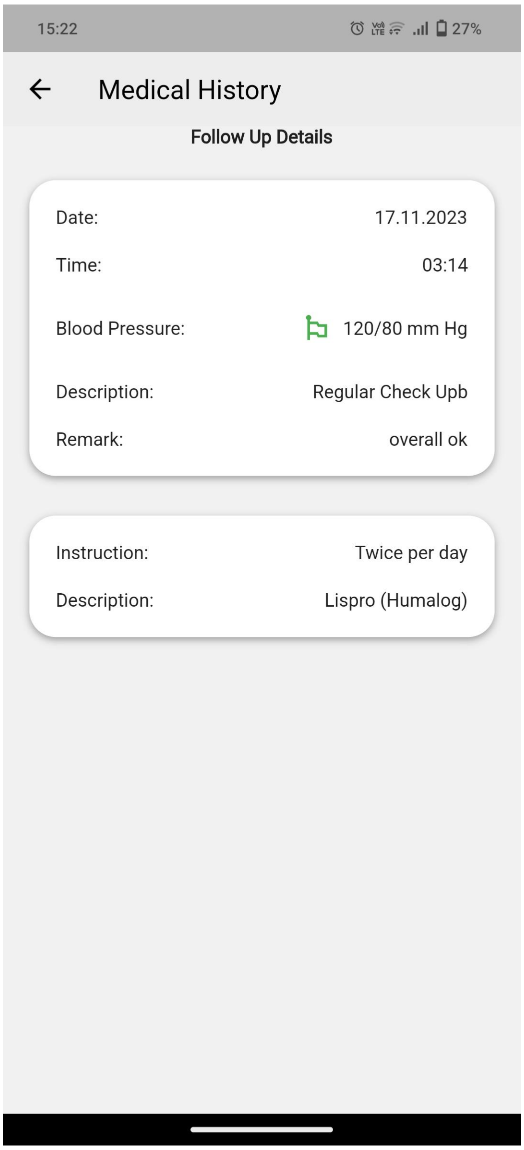


Figure 4.1.2.3.5: User Interface of Medical History Details

This is the detailed page of the medical history. It displays the follow-up details and some medication information that is recorded by the doctor when attending the appointment. This information will act as a reference to the patients when they attempt to look for previous medical records.

4.1.2.4 Ingredient Scanning Module

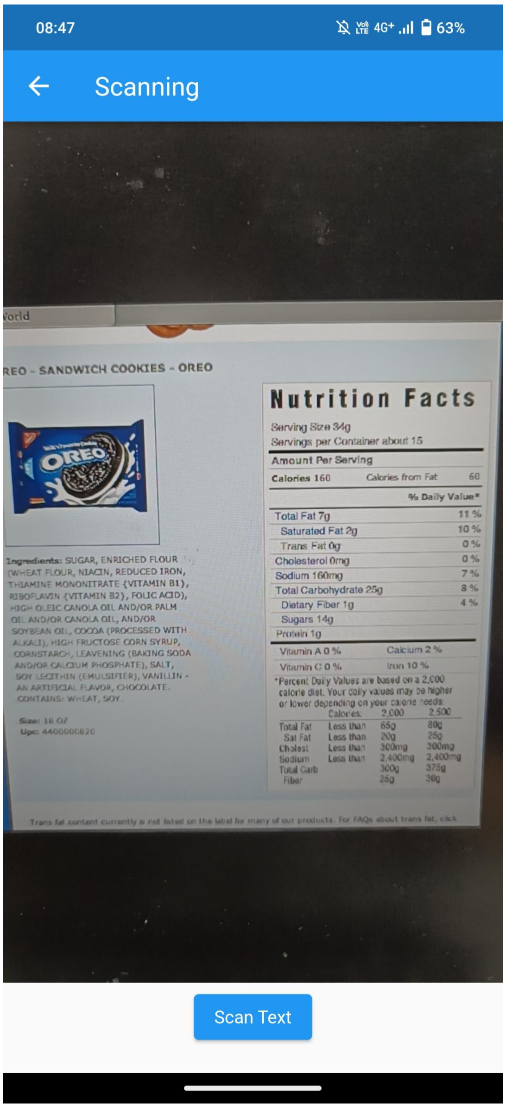


Figure 4.1.2.4.1: User Interface of Ingredient Scanning Menu

When the user navigates to the ingredient scanning page, it will display such a page to the user with a button to initiate the scanning action when the user focuses on the correct content. The ‘Scan Text’ button will extract the text from the scanned image and prompt the user to another page if the image is valid. However, if the image appears to be blurred or does not contain the required information, an error message toast will be prompted.

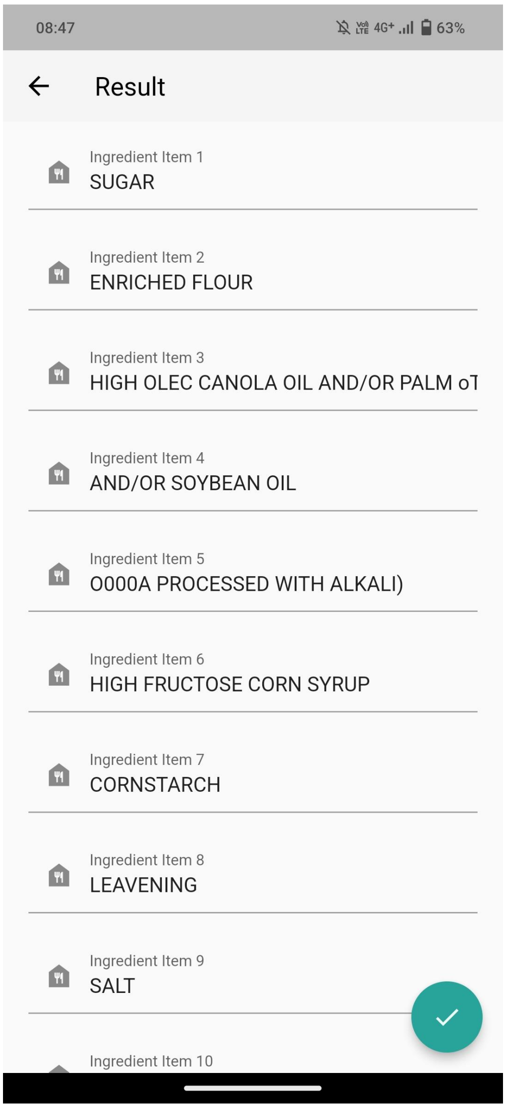


Figure 4.1.2.4.2: User Interface of Scanned Result Displaying I

After the system initiates the scanning process, the system will extract each ingredient item from the ingredient list, with a maximum of 10 items. These items extracted will be inserted into an editable text field to allow the users to correct the possible recognition error. When the ingredient items are correct and satisfied, the user may click on the ‘Done’ button to proceed with the next step.

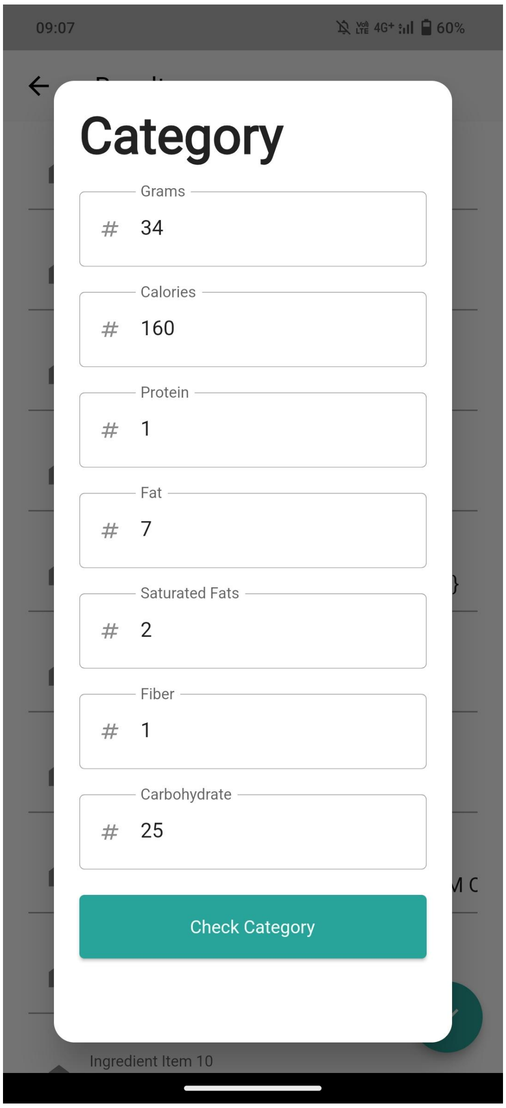


Figure 4.1.2.4.3: User Interface of Scanned Result Displaying II

After the ingredient items have been verified, a page for the confirmation of nutrition facts will be prompted. The user can validate the nutrition fact value being detected or manually enter the value if the values are not detected. To prevent invalid input of nutrition facts, there are error validations on those text fields which do not allow these fields to be left empty. A 'Check Category' button will be shown at the bottom of the pop-up category page, allowing the user to proceed to the result classification.

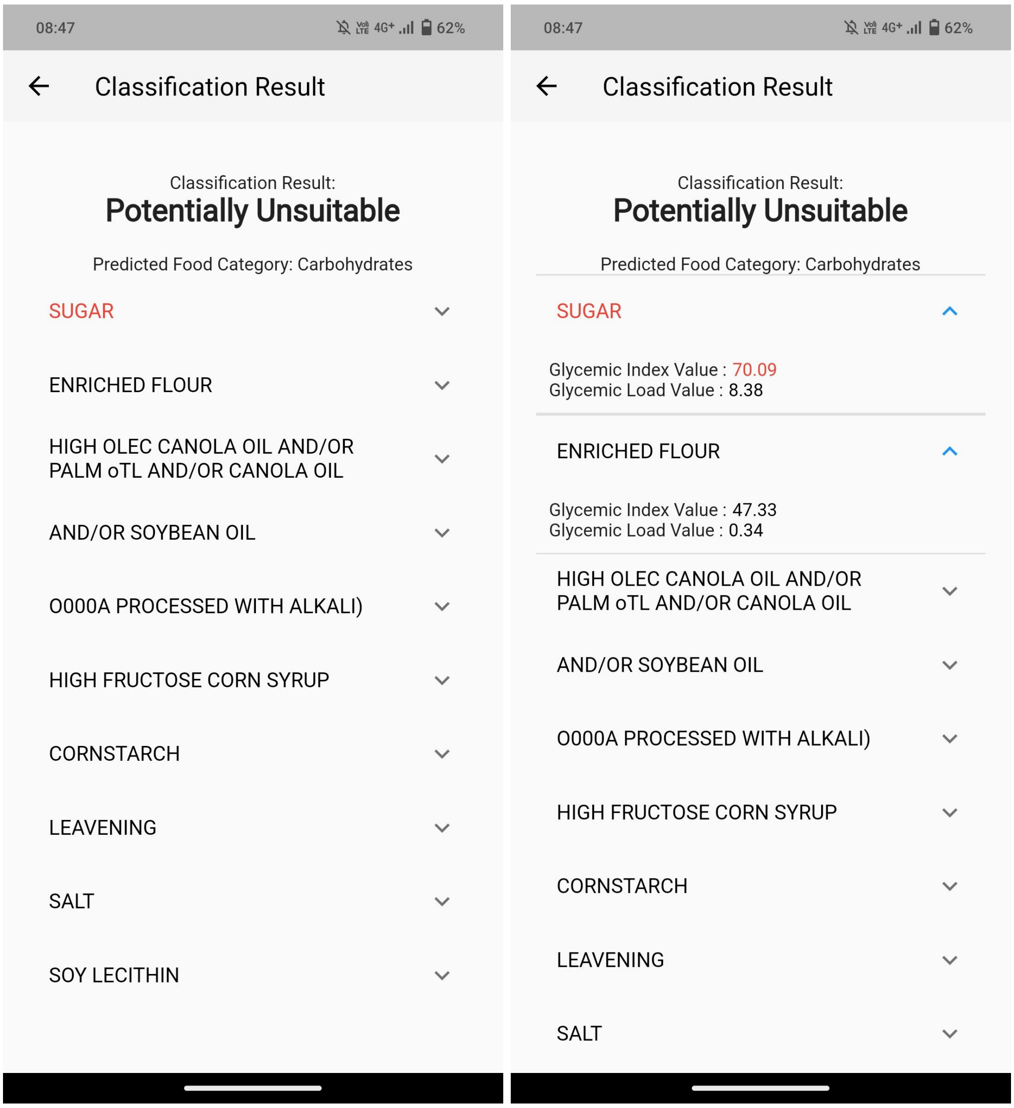


Figure 4.1.2.4.4: User Interface of Classification Result

After executing the classification model with the input scanned, the system will display the classification result, converting the consumption suitability of the scanned food product to the users. Along with the classification result, there will be a long list of items previously scanned with their respective glycemic-related information. If any of the glycemic value is within high range, the ingredient items will be displayed in red colour, emphasising this information.

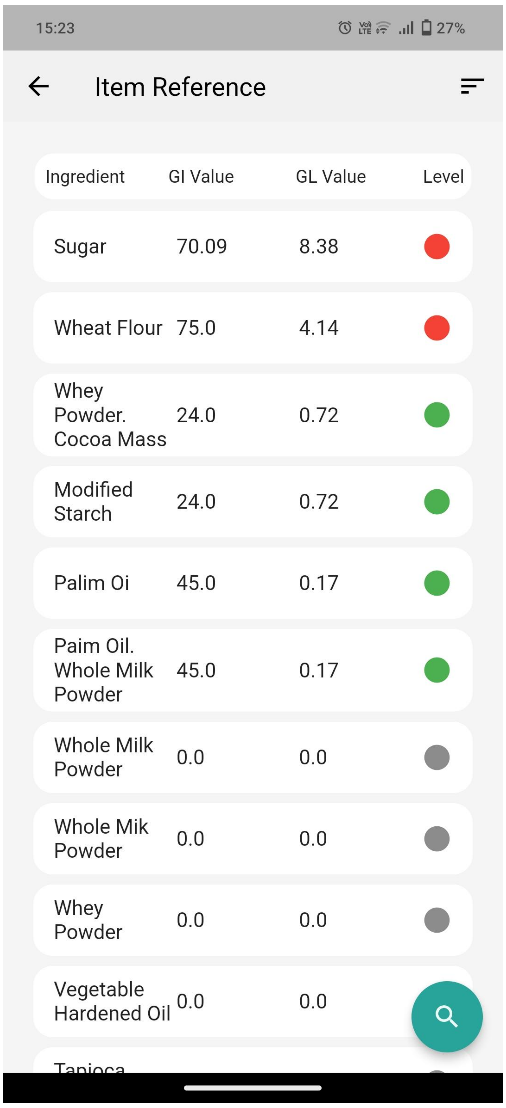


Figure 4.1.2.4.5: User Interface of Item Reference

As an additional platform to get information, this page displays a list of ingredient items with their respective glycemic index and glycemic lead value. These items are added to the database when a new ingredient item is detected during the scanning process. The users may sort the items according to the options provided or search for a particular item to retrieve their glycemic information.

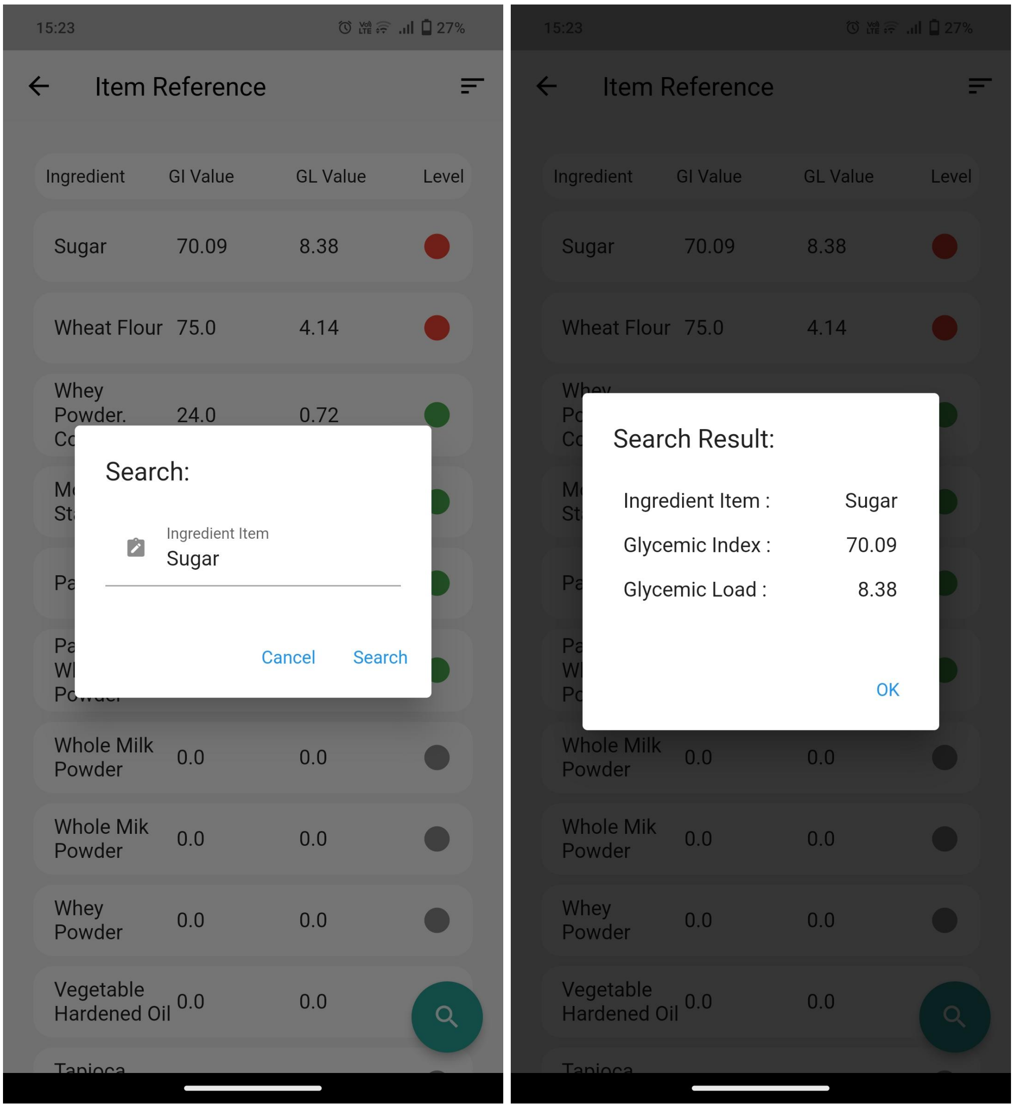


Figure 4.1.2.4.6: User Interface of Search Item

The user can search for a particular item through the search icon. When the search is initiated, a prompt dialogue will be displayed, allowing the user to enter the search item. When the item is found, it will display a search result that conveys the necessary information as shown above.

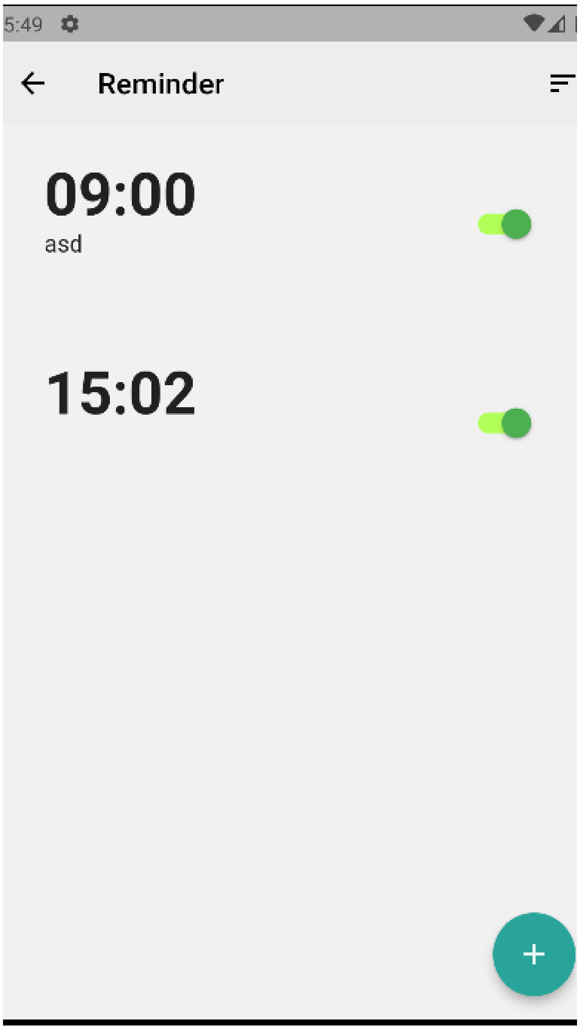


Figure 4.1.2.4.7: User Interface of Set Reminder

The system allows the patients to set reminders to alert themselves with necessary routines. They can set the reminder time based on their requirements. To clarify the category of reminder, the patients can select the items to remind on the "Reminder On" field and put in necessary remarks.

4.1.3 UI principle

The design of the user interface (UI) will refer to a set of guidelines and design principles known as Shneidderman's eight golden rules to ensure system usability and user experience. The introducer of these rules, Ben Shneiderman, proposed these principles to allow a productive designing process and successful user interface (Rua et al., 2019). The proposed system will consider these eight rules while designing the user interface to ensure a satisfying design.

A. Strive for consistency

This principle focuses on consistency design throughout the entire system. The design for the layout, and interactive components such as buttons, textbox and so on should have an identical design. This consistency is to increase the familiarity of the users towards the system and reduce the learning time required by them. In the proposed system, the entire system will be designed with a predefined colour scheme. All the buttons within the system will be the same colour to keep the users' attention. This can be found in the login page that the login button has the same colour as its background.

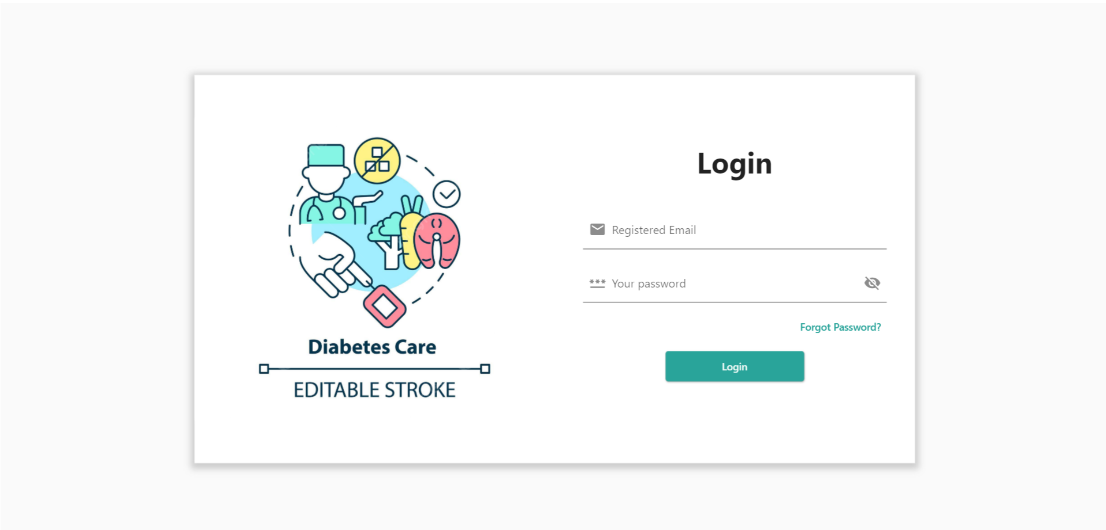


Figure 4.1.3.1: Same colour scheme for login page

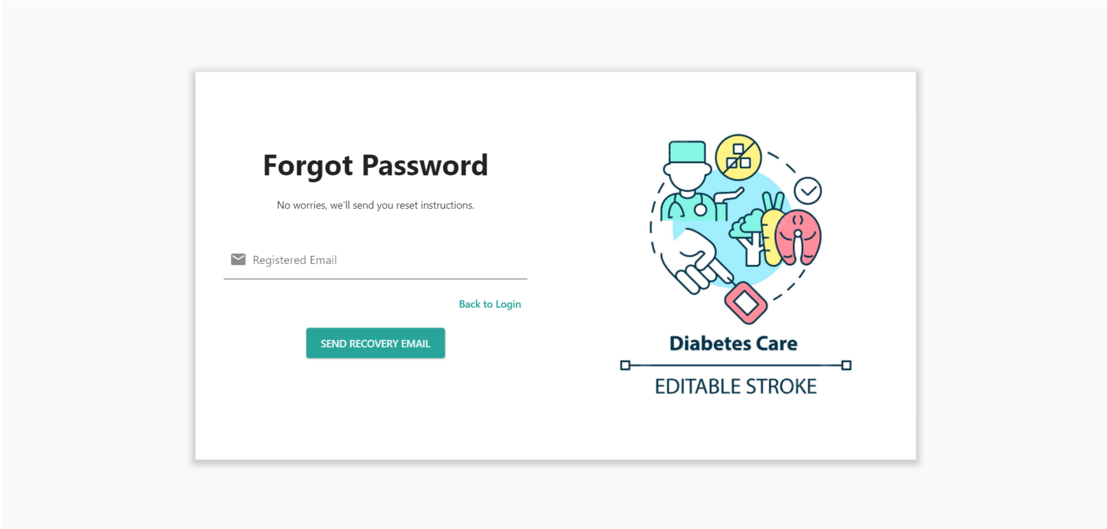


Figure 4.1.3.2: Same colour scheme for forgot password

B. Enable frequency users to use shortcuts

By enabling the users to access the desired functions within the website with shortcuts, the system can increase the pace of interaction, especially when the frequency of users increases. The objective of this principle can be achieved through a good hierarchy in the menu to make everything within the system clear. The proposed system will be incorporated with a navigation menu that allows users to visit pages or sections easily. This way, users can directly access the content they need without having to go through multiple clicks or searches. This feature helps in providing easy access to relevant content.

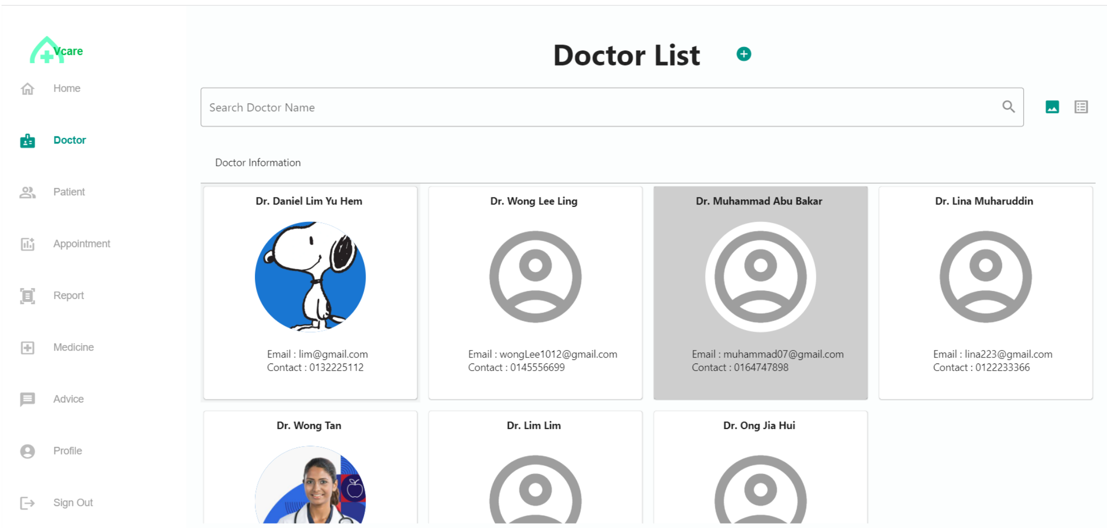


Figure 4.1.3.3: Navigation menu that categorises the pages

C. Offer information feedback

Providing informative feedback aims to increase the experience of the users. Through immediate and relevant feedback from the system, the users can understand the outcome of their actions and get updated on their progress. For example, in the first step of forgetting a password, the system will request the email address of the user to send a password recovery email to them. The system displays necessary guidance on the page to help the user get to know the action they should take.

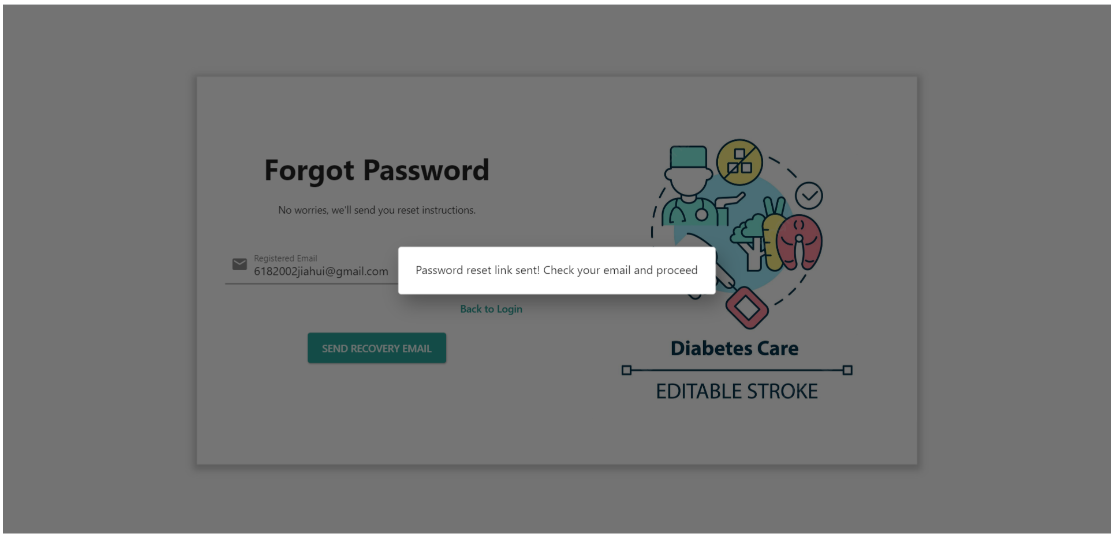


Figure 4.1.3.4: Forgot password page with guidance to lead the users

D. Design dialogues to yield closure

The process or sequences of the system are displayed clearly. The user can easily identify the beginning to the end of a certain task. This provides a sense of completion for the user's tasks. One of the designs within the system that considers this principle is the adding form for doctors and patients. On these pages, a progress indicator bar will be shown to notify the users about their current progress. When the user has completed one section and proceeded to the next, the system will update the progress bar.

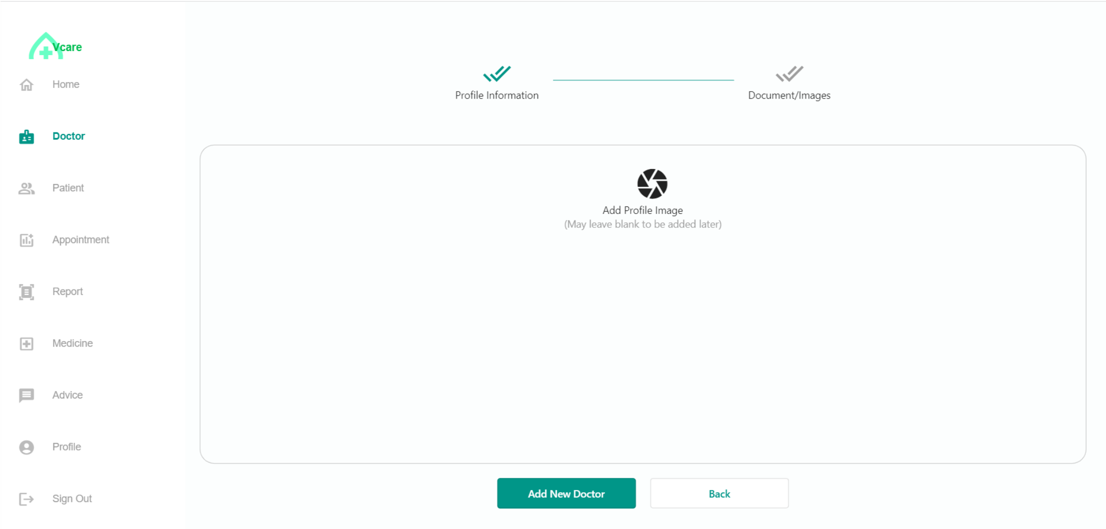


Figure 4.1.3.5: Progressive bar in the add doctor form

E. Permit Easy Reversal of Actions

The system should include the feature that allows their user to perform undo operations. By providing such a feature, the users will be encouraged to explore the interfaces without fear of making irreversible mistakes. One of the example user interfaces that follow this rule is the add patient form. On this page, the system will provide the user with a cancel button to allow them halt the operations according to their requirement.

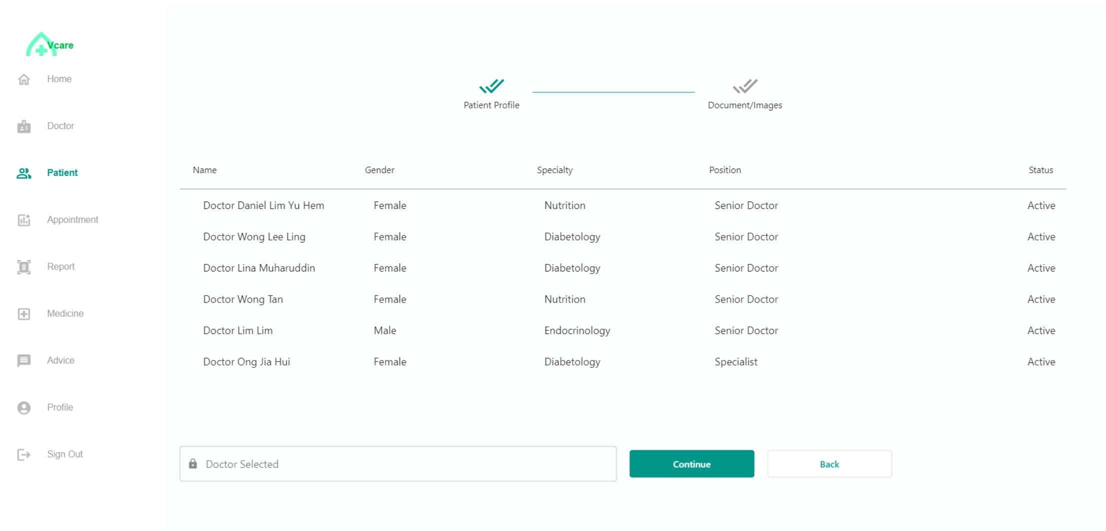


Figure 4.1.3.6: Back button in the new patient form

4.2 Data Design

4.2.1 Entity Relationship Diagram

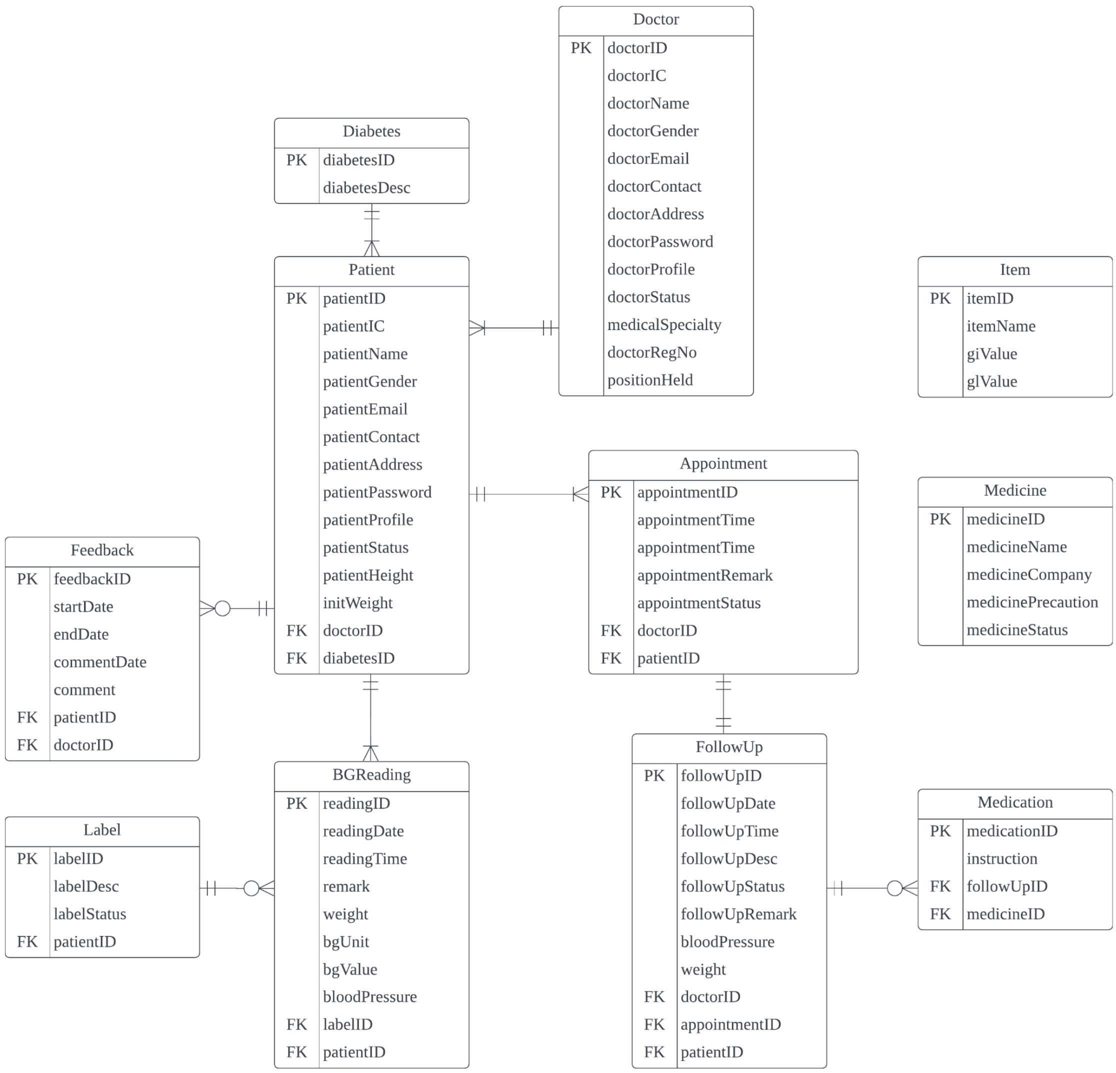


Figure 4.2.1.1: Entity Relationship Diagram of Diabetes Monitoring System

4.2.2 Class Diagram

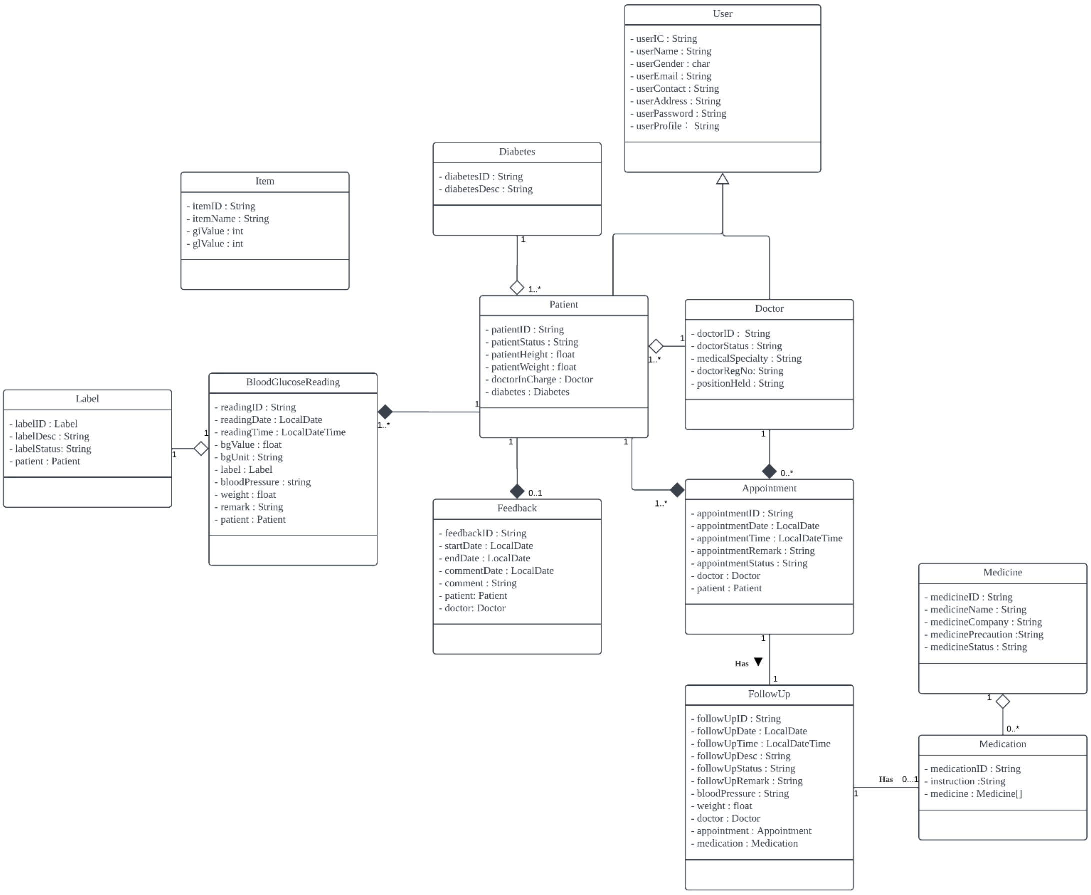


Figure 4.2.2.1 : Class Diagram of Diabetes Monitoring System

4.2.3 Data Dictionary

Patient Table

Table 4.2.3.1: Patient Table

Attributes	**Data Type**	**Description**	**Example**
patientID (Primary Key)	Varchar(9)	ID of the patient	PT0000001
patientIC	Varchar(14)	IC of the patient	090900-09-8877
patientName	Varchar(50)	Name of the patient	Melissa Lim
patientGender	Char	Gender of the patient	‘F’ or ‘M’
patientEmail	Varchar(50)	Email of the patient	melissa09@gmail.com
patientContact	Varchar(11)	Contact number of the patient	0198887777
patientAddress	Varchar(100)	Address of the patient	241, Level 2, Suria KLCC, Kuala Lumpur
patientPassword	Varchar(50)	Password of the patient	melissa09
patientProfile	Varchar(100)	Profile picture URL of the patient	https://images/patient1.jpg
patientStatus	Varchar(20)	Status of the patient	“Active” or “Inactive”
patientHeight	float	Height of the patient	151
patientWeight	float	Weight of the patient during the first visit	45
doctorID (Foreign Key)	Varchar(7)	ID of the doctor in charge	DR00001
diabetesID (Foreign Key)	Varchar(7)	ID of diabetes type	DT00001

Doctor Table

Table 4.2.3.2: Doctor Table

Attributes	**Data Type**	**Description**	**Example**

doctorID (Primary Key)	Varchar(9)	ID of the doctor	DR0000001
doctorIC	Varchar(14)	IC of the doctor	800911-02-0304
doctorName	Varchar(50)	Name of the doctor	Por Lee Lim
doctorGender	Char	Gender of the doctor	'F' or 'M'
doctorEmail	Varchar(50)	Email of the doctor	porLee9009@Gmail.com
doctorContact	Varchar(11)	Contact number of the doctor	0123336666
doctorAddress	Varchar(100)	Address of the doctor	88 U, Kompleks Sri Bukit Jambul, 11900 Pulau Pinang
doctorPassword	Varchar(50)	Password of the doctor	drPor01
doctorProfile	Varchar(100)	Profile picture URL of the doctor	images/doctor1.jpg
doctorStatus	Varchar(20)	Status of the doctor	"Active" or "Inactive"
medicalSpecialty	Varchar(100)	Medical Specialty of doctor	"Endocrinologist"
doctorRegNo	LocalDate	Employment date of doctor	25 Jul 2015
positionHeld	Varchar(50)	Position of the doctor	"Consulting Doctor"

Diabetes Table

Table 4.2.3.3: Diabetes Table

Attributes	**Data Type**	**Description**	**Example**
DiabetesType	Varchar(4)	ID of diabetes type	DB01
DiabetesDesc	Varchar(20)	Description for diabetes	"Type 1 Diabetes"

Appointment Table

Table 4.2.3.4: Appointment Table

Attributes	**Data Type**	**Description**	**Example**

appointmentID (Primary Key)	Varchar(9)	ID of the appointment	AP0000001
appointmentDat e	LocalDate	Date of the appointment reserved	16.07.2023
appointmentTi me	LocalDateTi me	Time of the appointment reserved	15:00
appointmentRe mark	Varchar(100)	Remark of appointment by patient	“Regular Diabetes Consultation”
appointmentStat us	Varchar(20)	Status of the appointment	“Pending” or “Completed” or “Approved” or “Rejected”
doctorID (Foreign Key)	Varchar(9)	ID of the doctor	DR0000001
patientID (Foreign Key)	Varchar(9)	ID of the patient	PT0000001

Follow-Ups Table

Table 4.2.3.5: FollowUp Table

Attributes	**Data Type**	**Description**	**Example**
followUpID (Primary Key)	Varchar(9)	ID of the follow up	FW0000001
followUpDate	LocalDate	Date of the follow up	16.07.2023
followUpTime	LocalDateTi me	Time of the follow up	15:00
followUpDesc	Varchar(100)	Description of the follow up	“Follow Up 17/07”
followUpStatus	Varchar(20)	Status of the follow up	“Completed” or “Pending”
followUpRemar k	Varchar(50)	Remark of the follow up	“Slight increase in weight”
bloodPressure	Varchar(6)	Blood pressure of the patient during that follow-up	110/90
weight	float	Weight of the patient during that follow-up	50

doctorID (Foreign Key)	Varchar(9)	ID of the doctor	DR0000001
appointmentID (Foreign Key)	Varchar(9)	ID of the appointment	AP0000001
medicationID (Foreign Key)	Varchar(9)	ID of the medication record for this follow up	MD0000001

Medication Table

Table 4.2.3.6: Medication Table

Attributes	Data Type	Description	Example
medicationID (Primary Key)	Varchar(9)	ID of the medication	MD0000001
instruction	Varchar(100)	Instruction of the medication to the patient	"Take after meal"
followUpID (Foreign Key)	Varchar(9)	ID of the medication's follow up	FW0000001
medicineID (Foreign Key)	Varchar(9)	ID of the medication's medicine	MC0000001

Medicine Table

Table 4.2.3.7: Medicine Table

Attributes	Data Type	Description	Example
medicineID (Primary Key)	Varchar(9)	ID of the medicine	MC0000001
medicineName	Varchar(100)	Name of the medicine	"Aspirin"
medicineCompany	Varchar(100)	Company of the medicine	"Panadol"
medicinePrecaution	Varchar(100)	Precaution of the medicine	"Not suitable for patient with high blood pressure"
medicineStatus	Varchar(100)	Status of the medicine	"Active"

Blood Glucose Reading Table

Table 4.2.3.8: Blood Glucose Reading Table

Attributes	Data Type	Description	Example
readingID (Primary Key)	Varchar(9)	ID of daily reading	BG0000001
readingDate	LocalDate	Date of the reading	16.12.2023
readingTime	LocalDateTime	Time of the reading	15:00
bgValue	float	Blood glucose level of the patient	90.8
bgUnit	Varchar(20)	Unit of the measured blood glucose level	“mmol/L” or “mg/dL”
bloodPessure	Varchar(10)	Blood pressure level of the patient	120/80
weight	float	Weight of the patient	54
remark	Varchar(100)	Remark by the patient on that particular reading	“Slightly higher that yesterday”
labelID	Varchar(9)	ID of the label	LB0000001
patientID (Foreign Key)	Varchar(9)	ID of the patient	PT0000001

Feedback Table

Table 4.2.3.9: Feedback Table

Attributes	Data Type	Description	Example
feedbackID (Primary Key)	Varchar(9)	ID of the feedback	FB0000001
startDate	LocalDate	Start time of the monitored blood glucose	16.11.2023
endDate	LocalDate	End date of the monitored blood glucose	16.12.2023
commentDate	LocalDate	Date the comment made	30.12.2023
comment	Varchar(100)	Comment by the doctor	“Need more exercise”
patientID	Varchar(9)	ID of the patient	PT0000001

(Foreign Key)			
doctorID (Foreign Key)	Varchar(9)	ID of the doctor	DR0000001

Label Table

Table 4.2.3.10: Label Table

Attributes	**Data Type**	**Description**	**Example**
labelID (Primary Key)	Varchar(9)	ID of label	LB0000001
labelDesc	Varchar(20)	Description for the reading	"Fasting Measurement" or "Random Measurement"
labelStatus	Varchar(20)	Status of the label	"Active"
patientID (Foreign Key)	Varchar(9)	ID of the patient	PT0000001

Ingredient Item Table

Table 4.2.3.11: Item Table

Attributes	**Data Type**	**Description**	**Example**
itemID (Primary Key)	Varchar(9)	ID of the item	IT0000001
itemName	Varchar(100)	Name of the item in the ingredient list	"Sugar"
giValue	float	Glycemic index value of the food item	50
glValue	float	Glycemic load value of the food item	50

4.3 Report Design

4.3.1 Blood Glucose Trend Report

Title: Frequency of hypoglycemic/Hyperglycemic Event within this month

Used by: Doctor

Content: This report focuses on analysing the frequency of both hypoglycemic (low blood sugar) and hyperglycemic (high blood sugar) events experienced by patients over a specific period. It presents an overview of the frequency of hypoglycemic events and hyperglycemic events. The data of this report will be gathered from the daily blood glucose readings recorded by the patients, defining the happenings of these events.

Purpose: This report aims to enable the doctor to gauge the effectiveness of medication or treatment plans of their patients. Understanding the occurrence of these events helps the doctor to effectively identify potential improvements in the treatment plan of the patients.

Table 4.3.1.1: Report of Frequency of Hypoglycemic Event

Hypoglycemic (Low blood glucose level)	
Patient Name	Frequency
Wong Xiao Ming	2
Ali Muhammad	3
Tan Li Li	1
Karen	2
Meg	3

Table 4.3.1.2: Report of Frequency of Hyperglycemic Event

Hyperglycemic (High blood glucose level)	
Patient Name	Frequency
Wong Xiao Ming	2

Ali Muhammad	3
Karen	2
Meg	3

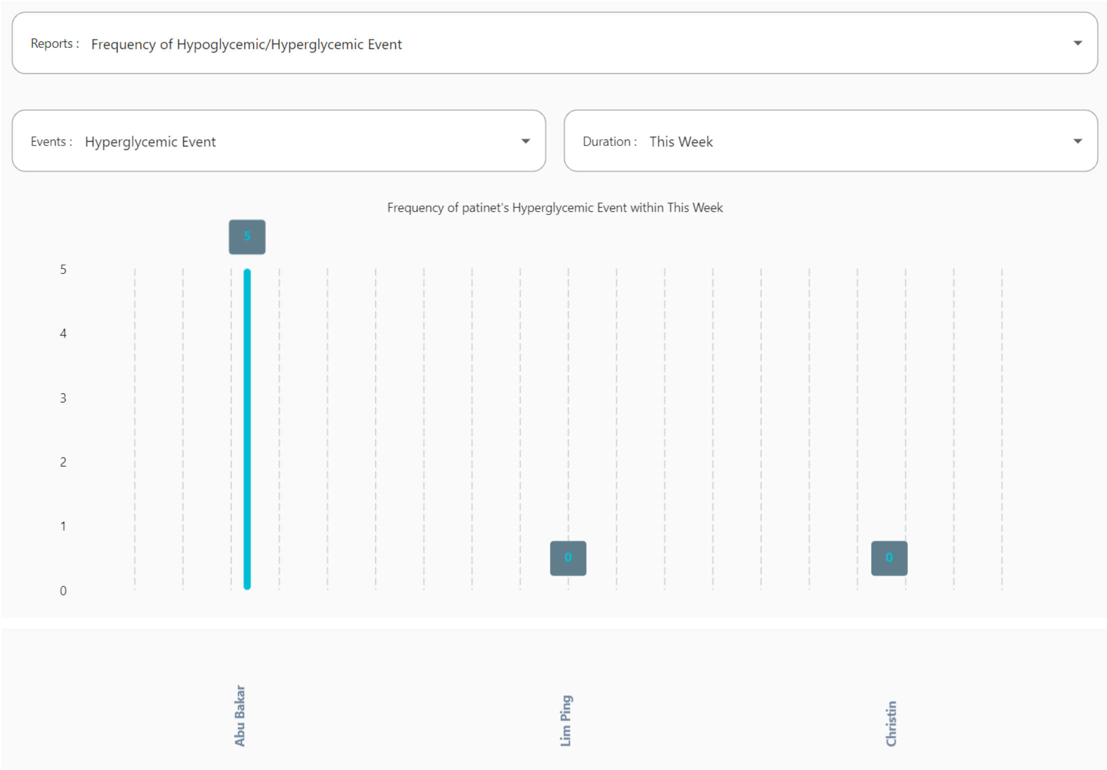


Figure 4.3.1.1: Presentation of blood glucose trend report in the system

4.3.2 Diabetes Demographics Report

Title: Diabetes Demographics group by Gender Report

Used By: Doctor

Content: This report aims to provide an overview on the prevalence of diabetes patients based on gender. It will calculate the total number of patients who have registered in the system based on their diabetes type.

Purpose: This report aims to help the doctor to know the prevalence of diabetes patients in charge of him. This can help him to produce better medication strategies based on the patient's prevalence.

Table 4.3.2.1: Report of Diabetes Demographics

Gender	Total Number of Patient
Diabetes Type 1	
Male	5
Female	5
Diabetes Type 2	
Male	7
Female	6

Figure 4.3.2.1: Presentation of diabetes demographics report in the system

4.4 Implementation of Algorithm

The k-nearest neighbour (KNN) algorithm is a supervised machine learning algorithm primarily used for classification purposes. Generally, the KNN algorithm can classify datasets using a training model similar to the testing query by taking into account the k nearest training data points (neighbours), which are the closest to the query it is testing. Finally, the algorithm performs a majority voting rule to check which classification to finalise (Uddin et al., 2022).

K-Nearest Neighbors (KNN) is a simple and widely used classification and regression algorithm in machine learning. It is a type of distance-based classifier where the algorithm makes predictions based on the similarity between the new data point and its k-nearest neighbours in the training dataset. KNN will carry out its task by calculating the distance between the data points to find the nearest neighbours. The distance metric commonly used is the Euclidean distance. The Euclidean distance is used to find the distance between two points on a plane. It can also be known as the distance between two points. The formula of the Euclidean distance is as follows:

Euclidean Distance, $d = \sqrt{[(x2 - x1)^2 + (y2 - y1)^2]}$.

Another common formula utilised by KNN is the Manhattan Distance. In contrast to the Euclidean distance that calculates the straight-line distance between two points in a Cartesian coordinate system, the Manhattan distance considers only horizontal and vertical movements and does not account for diagonals. It can also be known as the sum of the absolute differences between their corresponding coordinates. The formula of the Manhattan distance is as follows:

Manhattan Distance, $d = | x1 - x2 | + | y1 - y2 |$

In KNN, when a new data point is presented, KNN calculates its distances to all data points in the training dataset. The k data points with the smallest distances to the new data point are identified. To classify the new data point, the algorithm looks at the class labels of these k neighbours and selects the class that appears most frequently among them. This class label is then assigned to the new data point as its predicted classification.

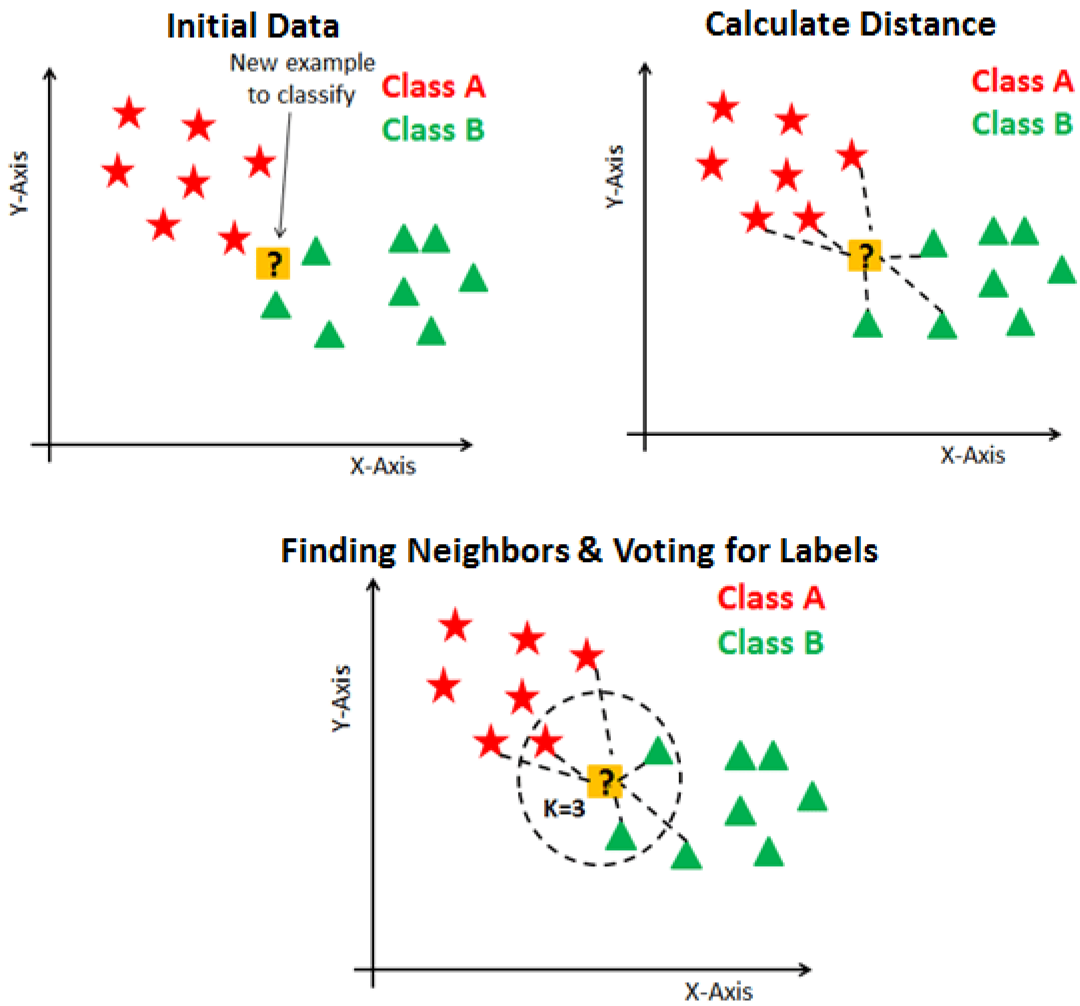


Figure 4.4.1: Concept of K-Nearest Neighbours (KNN) algorithm

The figure above shows that when there is a new example to classify, the KNN algorithm will first calculate the distance. Following, it will get the vote for the new example from the majority labels of the neighbour. For instance, the new example in the figure above belongs to class B.

4.5 Security Design

4.5.1 Access Control Methods

In the proposed system, a role-based access control mechanism will be implemented to ensure the users are granted access only to necessary resources and functions based on their roles. The access control methods are being considered to reduce possible security risks such as data breaches and unauthorised access, thereby safeguarding the system from abuse by malicious parties. The roles of the users will be identified during the login process through their login credentials. To achieve this, the system utilises Firebase Authentication Services provided by Google Firebase (Google Firebase, 2023). Upon successful authentication, users will be redirected to the appropriate page or module in the system, according to their role. Based on the determined roles, the system will effectively manage access to specific functions and pages. For instance, users with the "doctor" role will have access to all functions in the system, including doctor management, patient management, appointment scheduling, and more. This comprehensive access ensures administrators can oversee and manage the entire system. While those users identified as patients are eligible for the blood glucose module for daily recording, and the appointment module for scheduling appointments with doctors and managing their profiles. On the other hand, the users identified as doctors will have access to patients' information, enabling them to monitor their patients. The role-based access control helps to eliminate the possibility of users accessing unrelated information or functions. This improves the overall user experience by offering a streamlined and focused interface, tailored to each user's specific responsibilities and needs to carry out their tasks or activities.

4.5.2 Hashing

Hashing will be included in the system to secure sensitive data such as the user's password. The implementation of hashing in the proposed system is to enhance the privacy and integrity of the user accounts, ensuring the user data is secure. By applying hashing, the password data will be stored as encrypted data instead of plain text which can easily expose the user’s credentials. When the user signs up or updates their passwords, Firebase Authentication will apply the hashing algorithm to generate a unique hash value (Google Firebase, 2023). When the user attempts to log in, the password they entered will be hashed with the same algorithm and compared to the resulting hash with the previously stored hashed value. The user will only be allowed to enter the system if the hashes match, indicating the login is successful. By including password hashing in the proposed system, the proposed system can ensure that unauthorised parties will be unable to access and retrieve the original passwords if the servers were compromised as hashing is one-way and not reversible. The image below shows the password hashing function provided by Firebase Authentication:

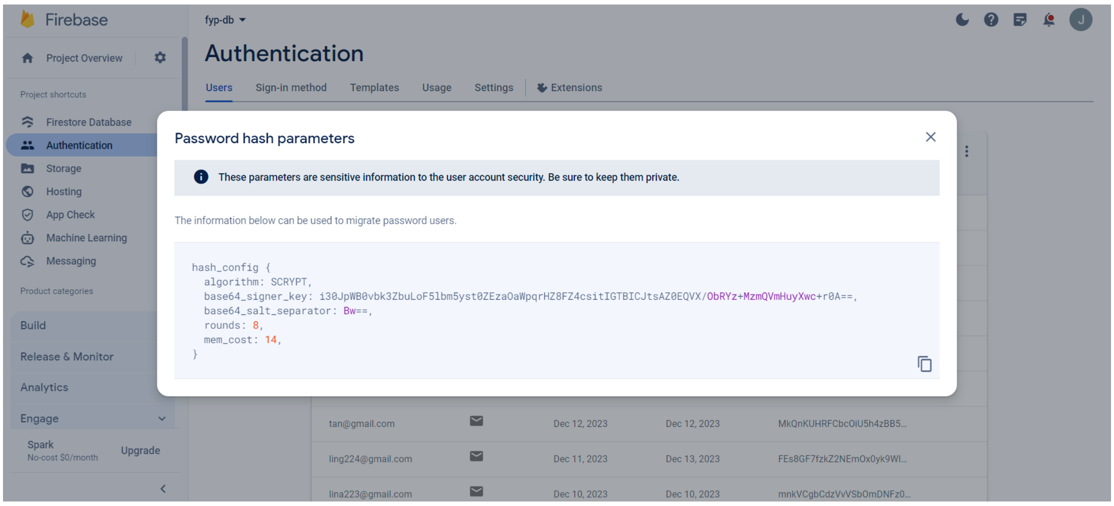


Figure 4.5.2.1: Password Hashing Function by Firebase Authentication

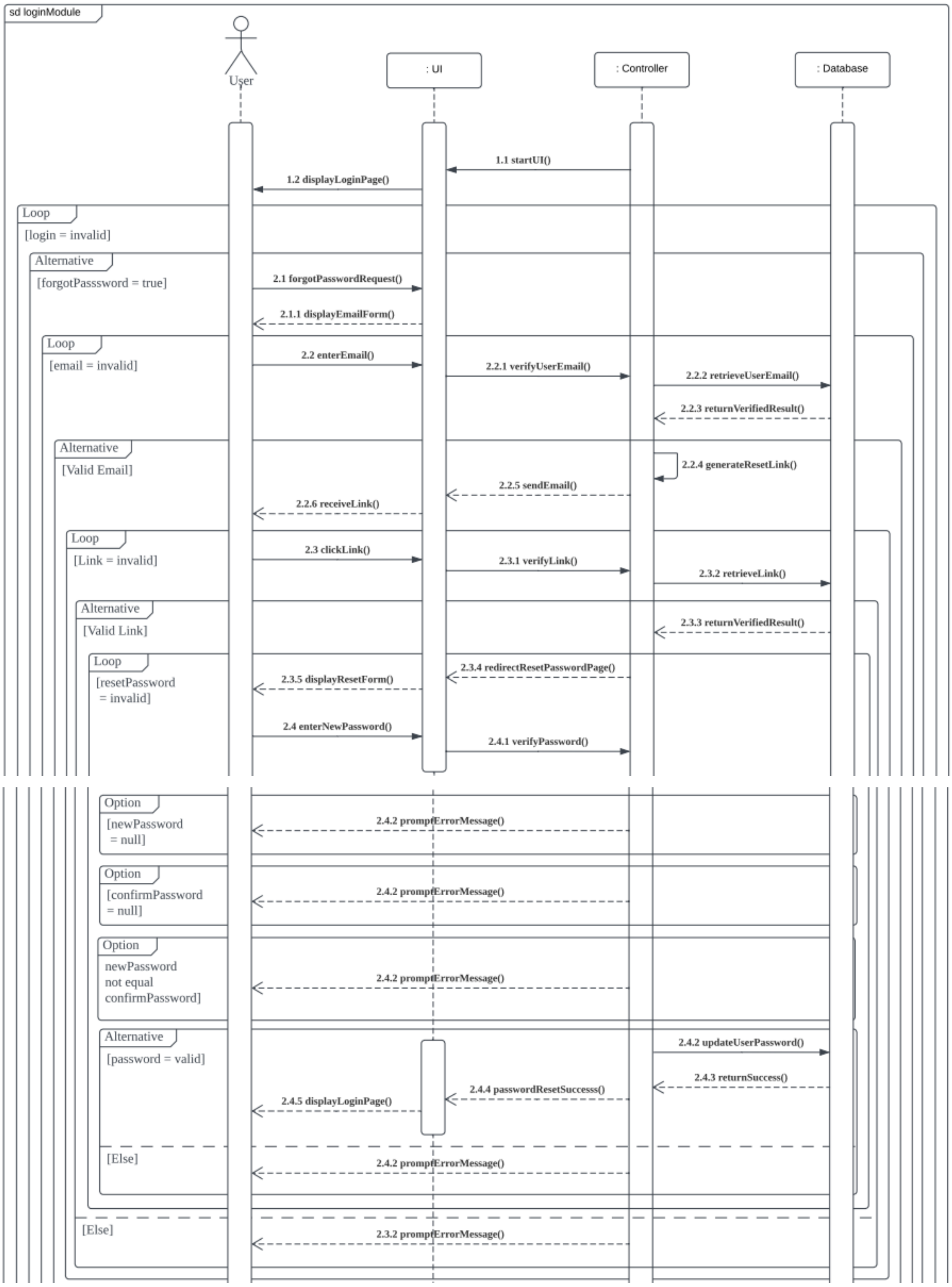
4.5.3 Authentication

The authentication of the system will be implemented using the Firebase Authentication SDK (Google Firebase, 2023). Users will be required to provide their email and password during the login process to associate themselves with their accounts. Firebase Authentication SDK, which offers various authentication methods, provides users with flexibility in choosing the preferred authentication method. The methods provided include passwords, phone numbers and popular federated identities such as Google and Social media accounts. By implementing the authentication technique, the system can enable email verification for the password reset process. This additional layer of security helps confirm the authenticity of the user and prevents spamming activities, such as creating fake accounts within the system.

4.6 Process Design

4.6.1 Sequence Diagram

4.6.1.1 Login Module



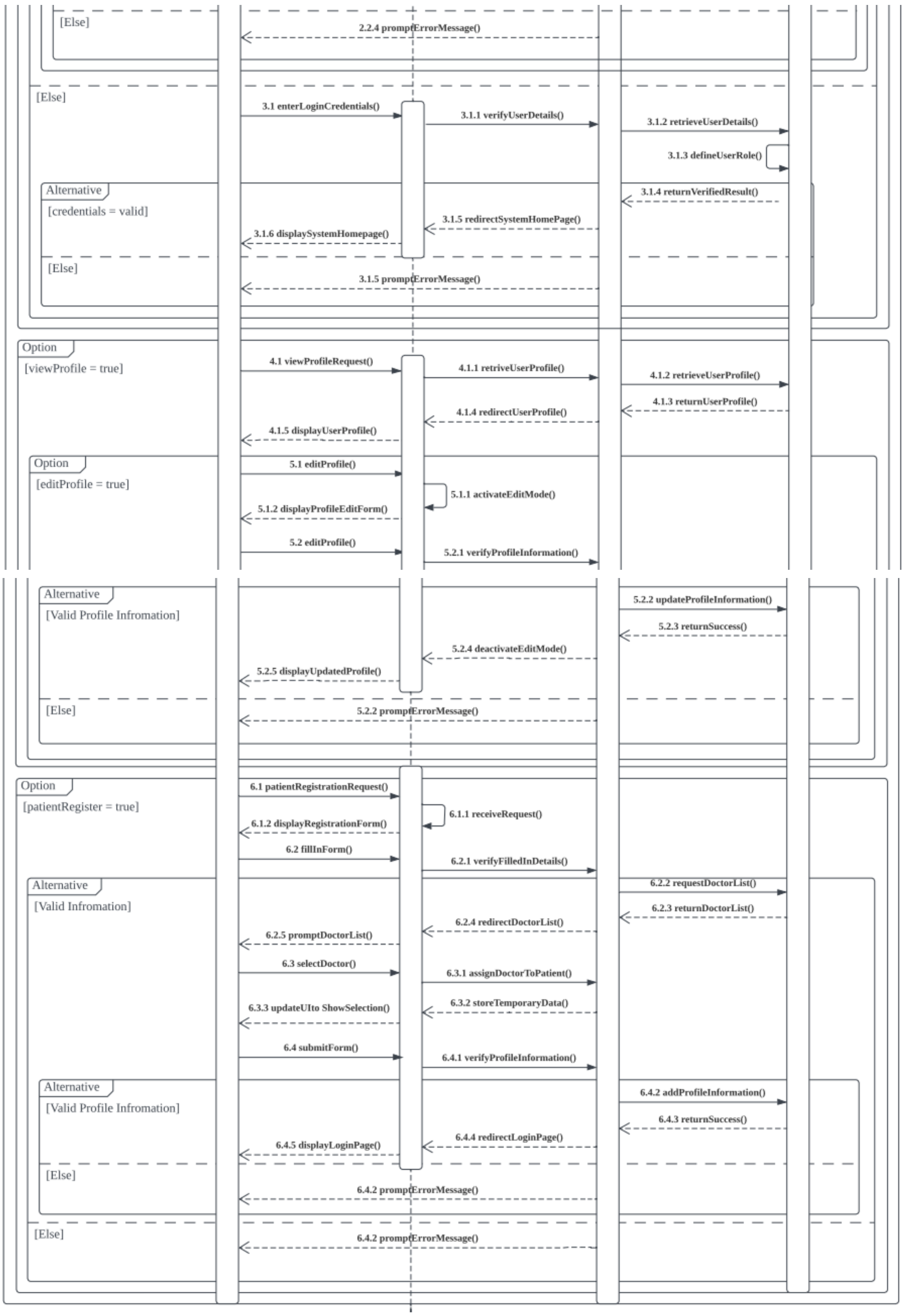
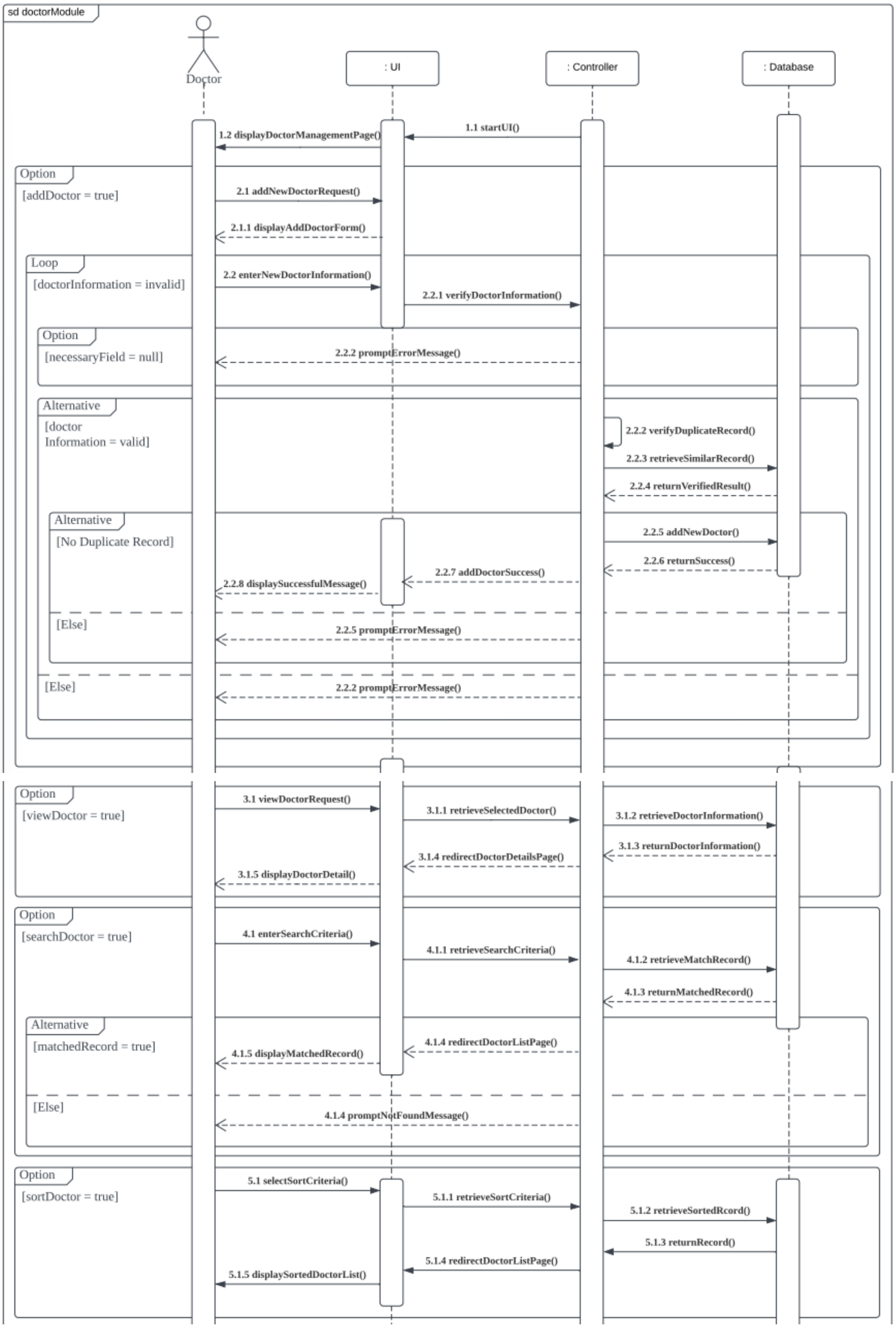


Figure 4.6.1.1.1 Sequence Diagram of Login Module

4.6.1.2 Doctor Module



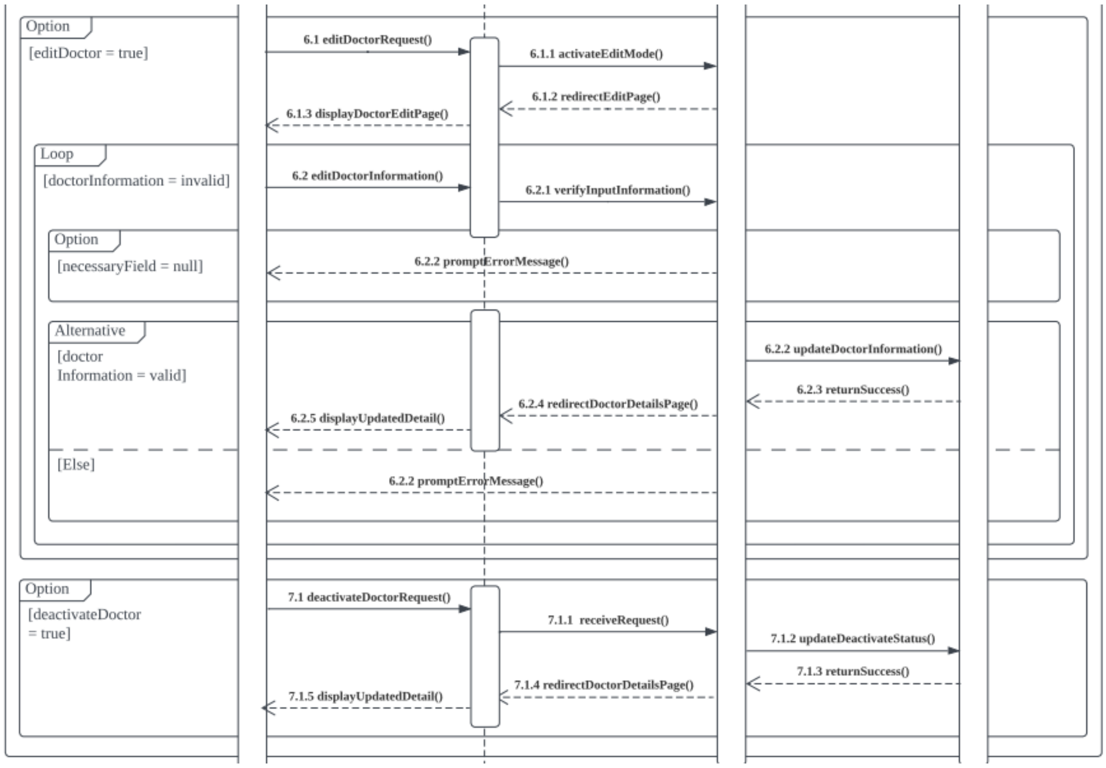
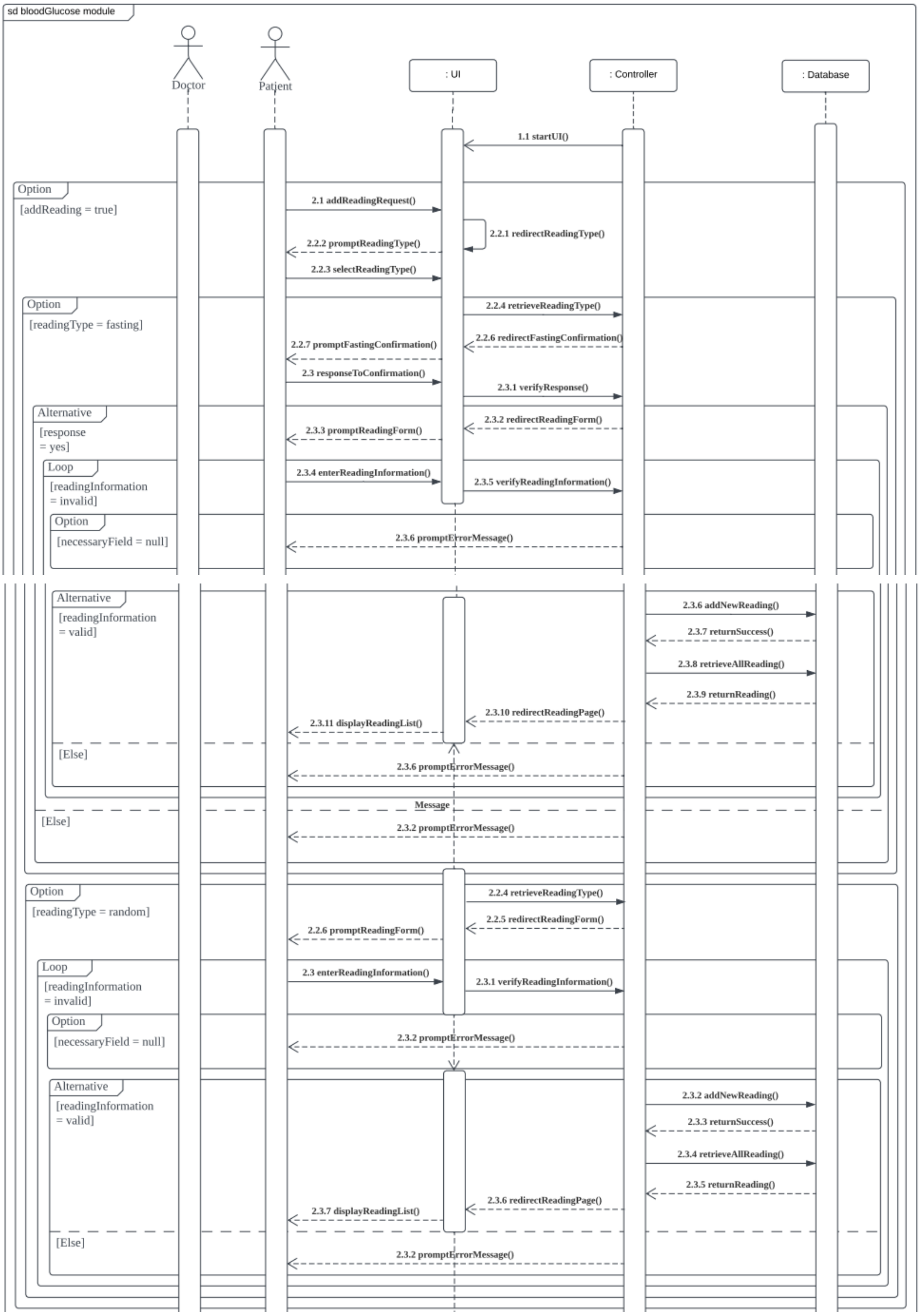


Figure 4.6.1.2.1: Sequence Diagram of Doctor Module

4.6.1.3 Blood Glucose Module



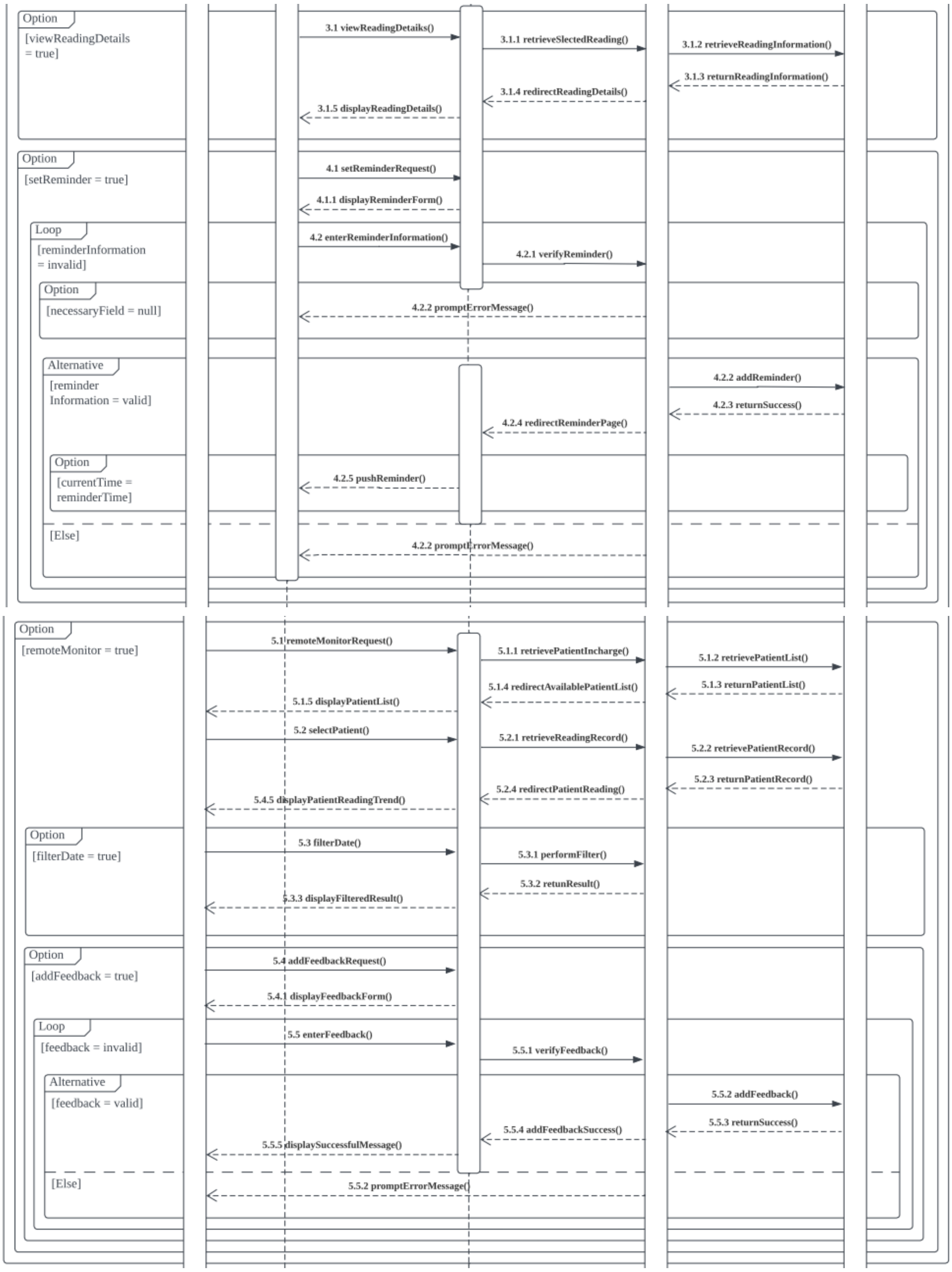


Figure 4.6.1.3.1: Sequence Diagram of Blood Glucose Module

4.6.1.4 Patient Module

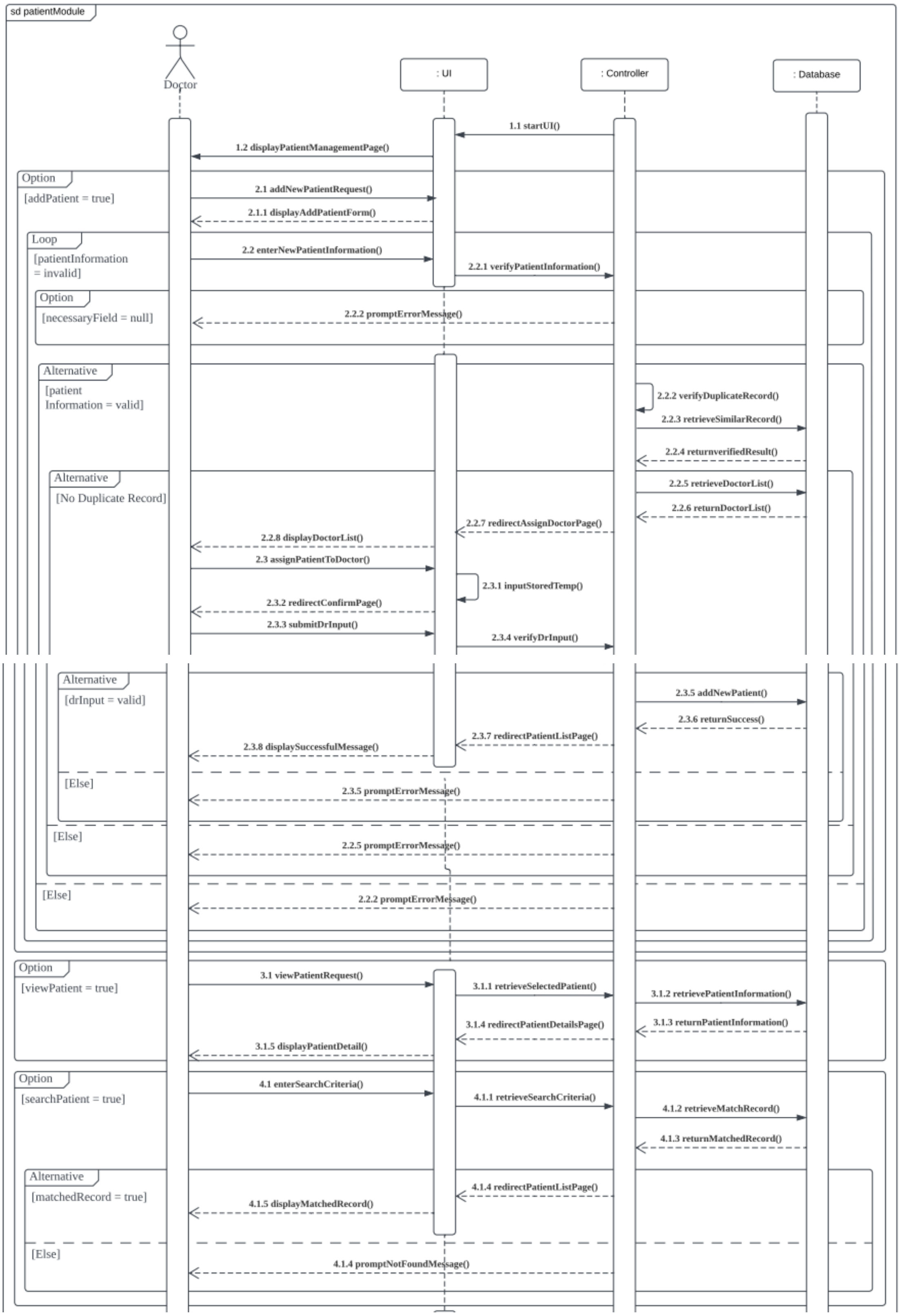


Figure 4.6.1.4.1: Sequence Diagram of Patient Module

4.6.1.5 Ingredient Scanning Module

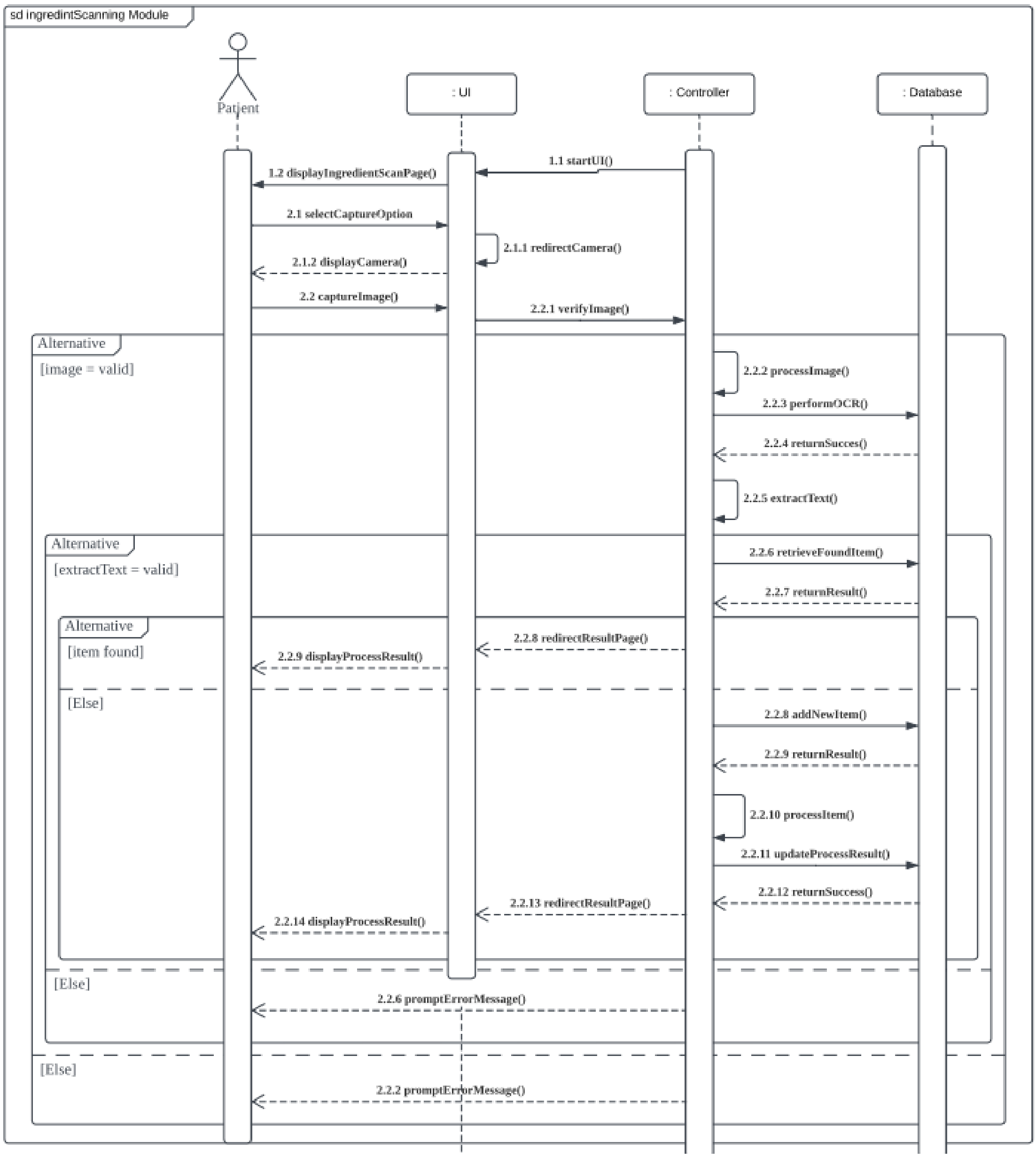


Figure 4.6.1.5.1: Sequence Diagram of Ingredient Scanning Module

4.6.1.6 Appointment Module

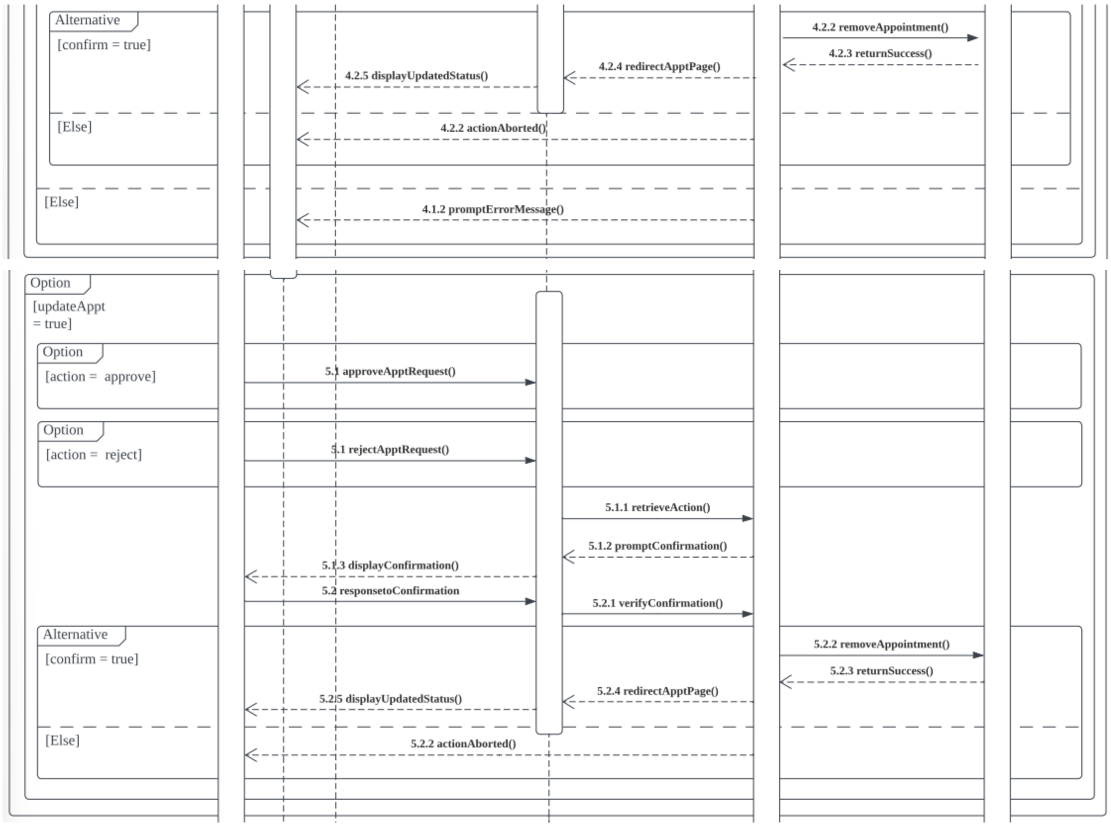


Figure 4.6.1.6.1: Sequence Diagram of Appointment Module

4.6.2 State Diagram

4.6.2.1 Login Module

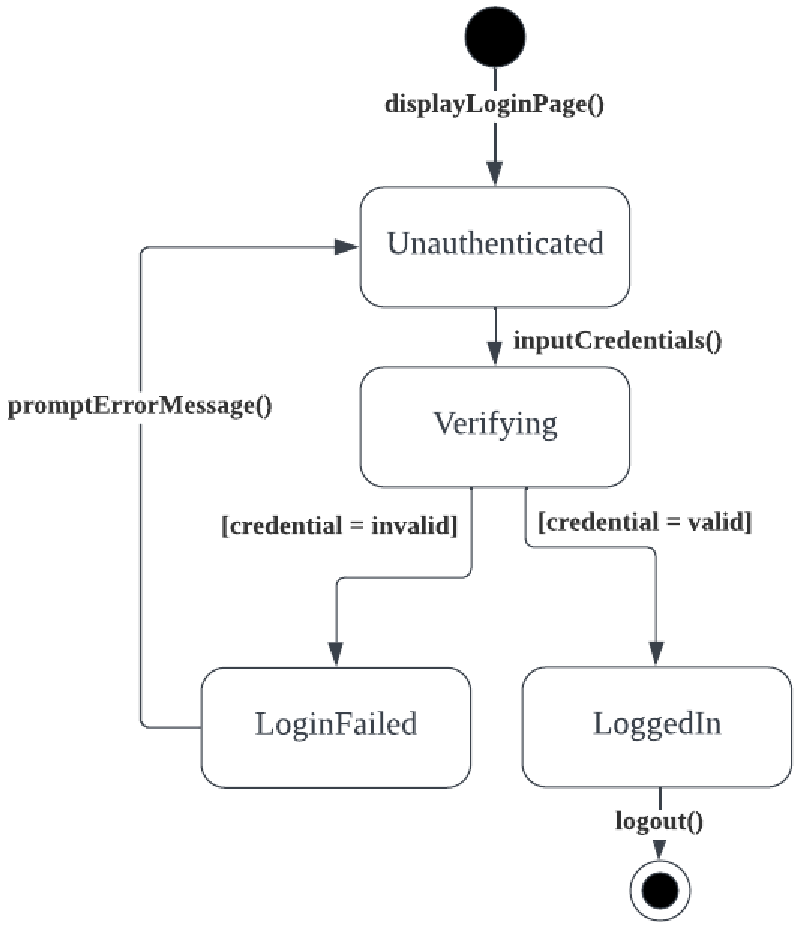


Figure 4.6.2.1.1: State Diagram of Login

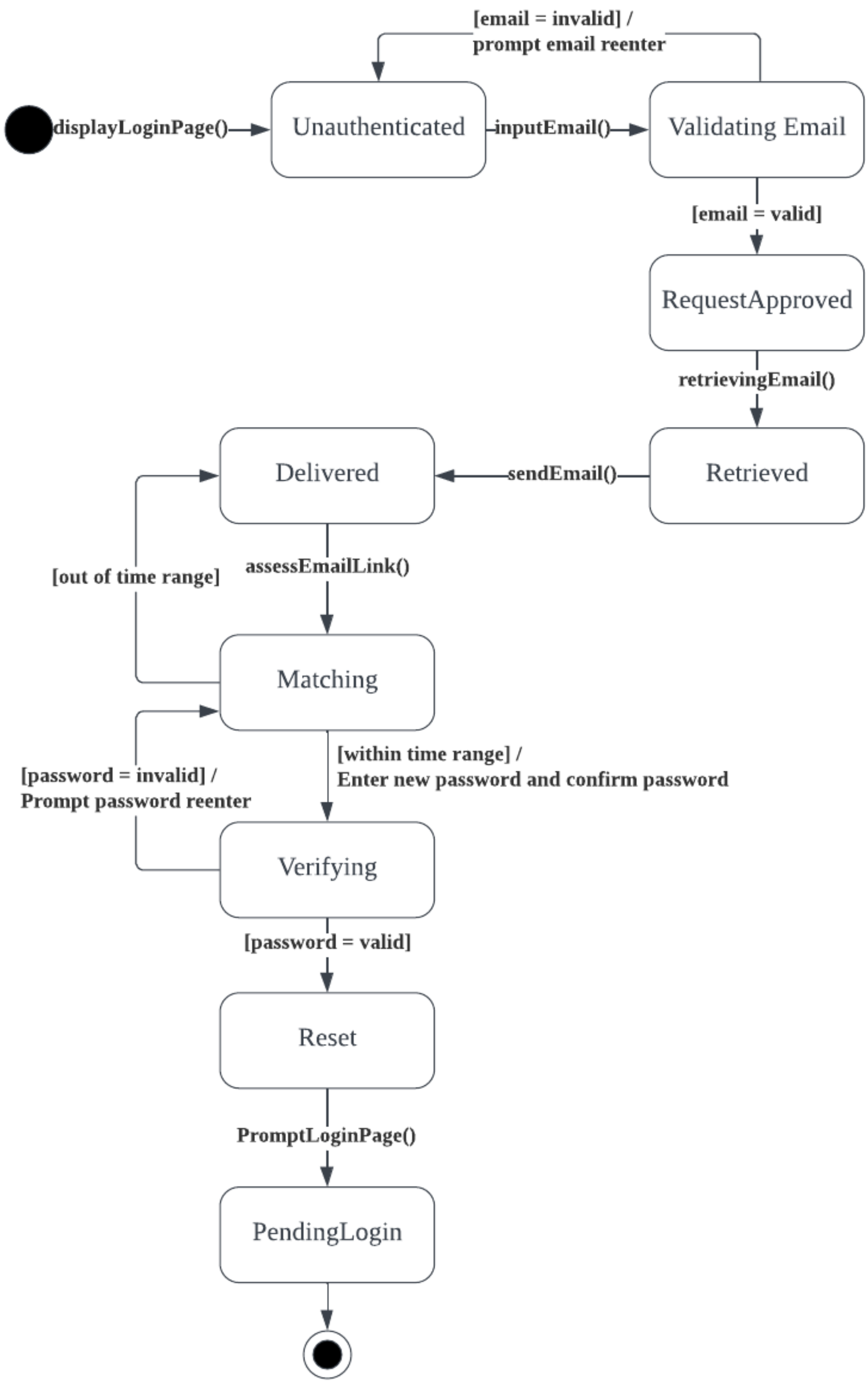


Figure 4.6.2.1.2: State Diagram of Forgot Password

4.6.2.2 Doctor Module

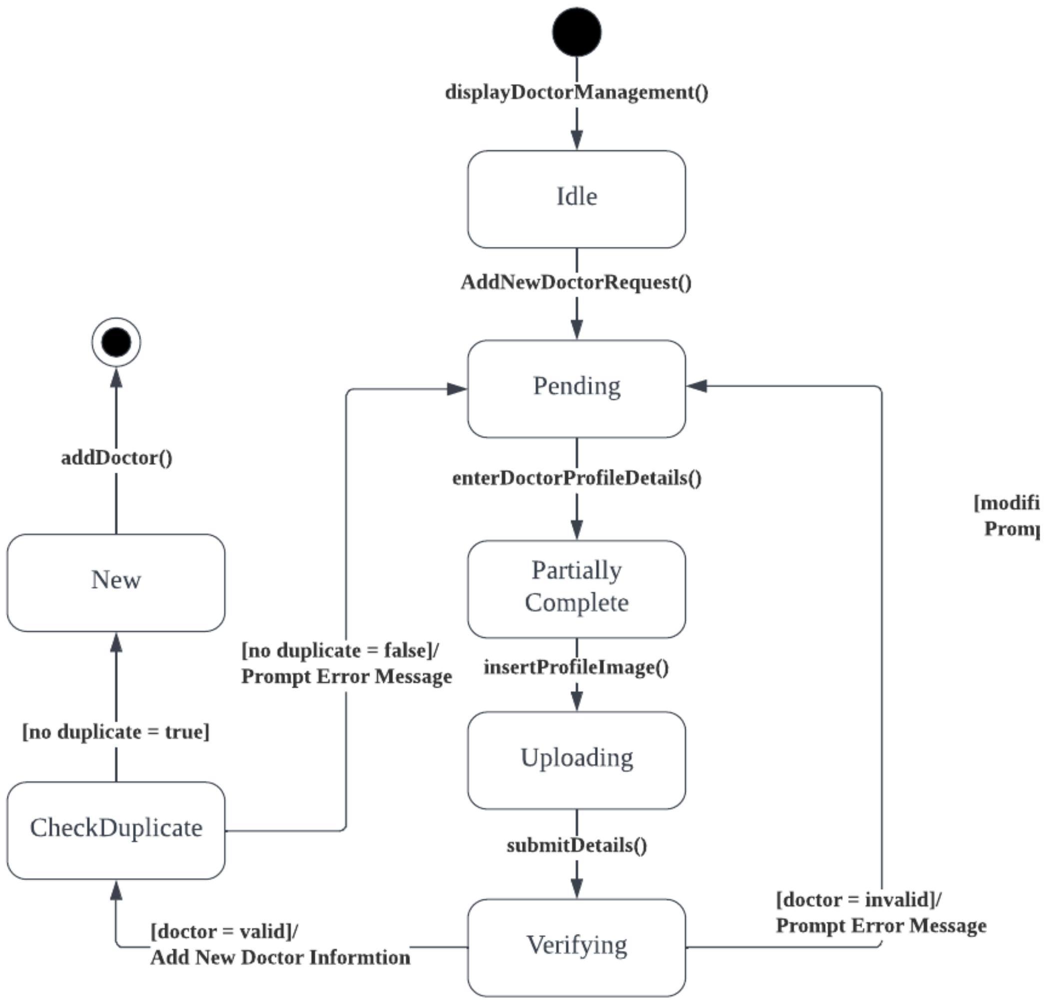


Figure 4.6.2.2.1: State Diagram of Add Doctor

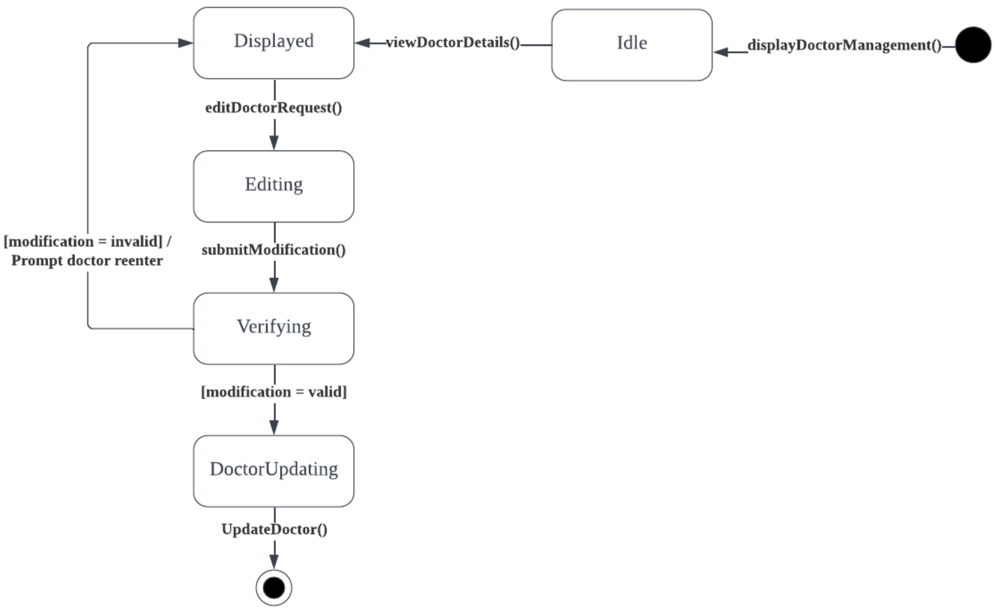


Figure 4.6.2.2.2: State Diagram of View and Edit Doctor

4.6.2.3 Appointment Module

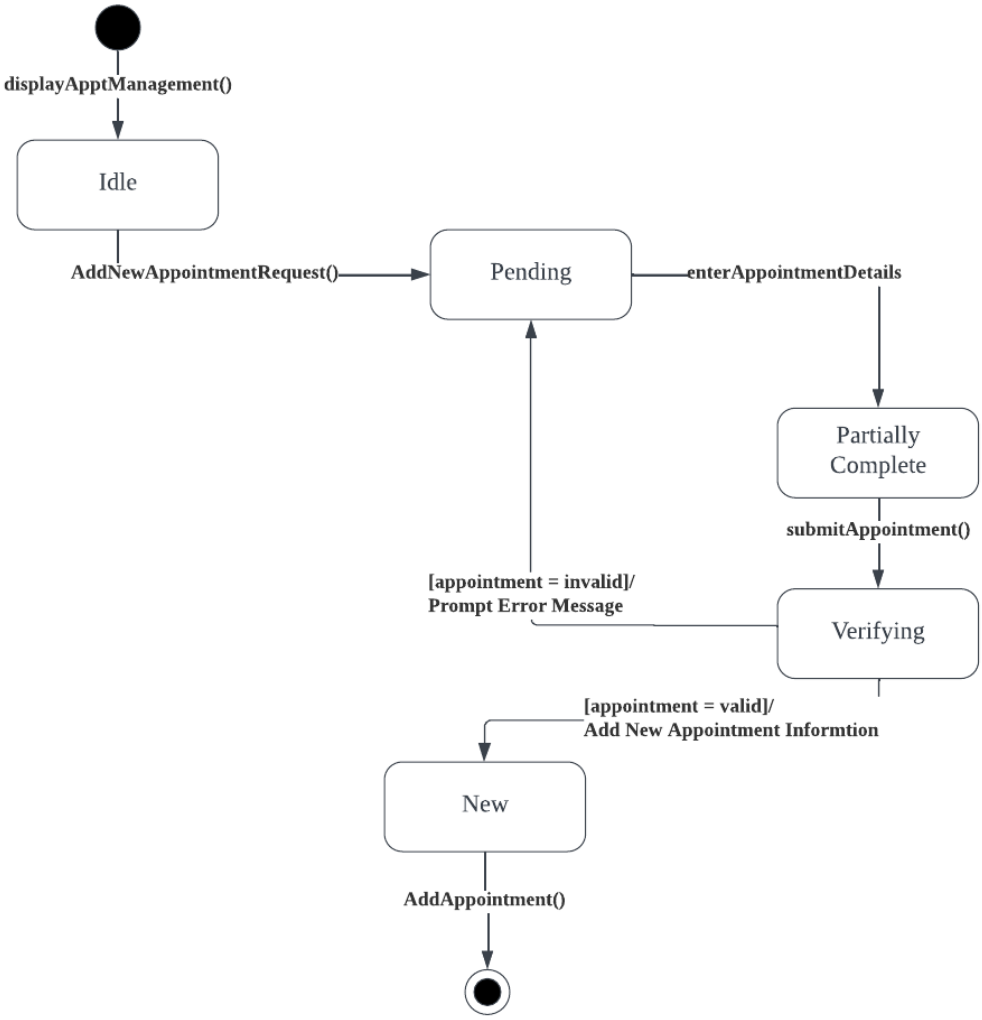


Figure 4.6.2.3.1: State Diagram of Add Appointment

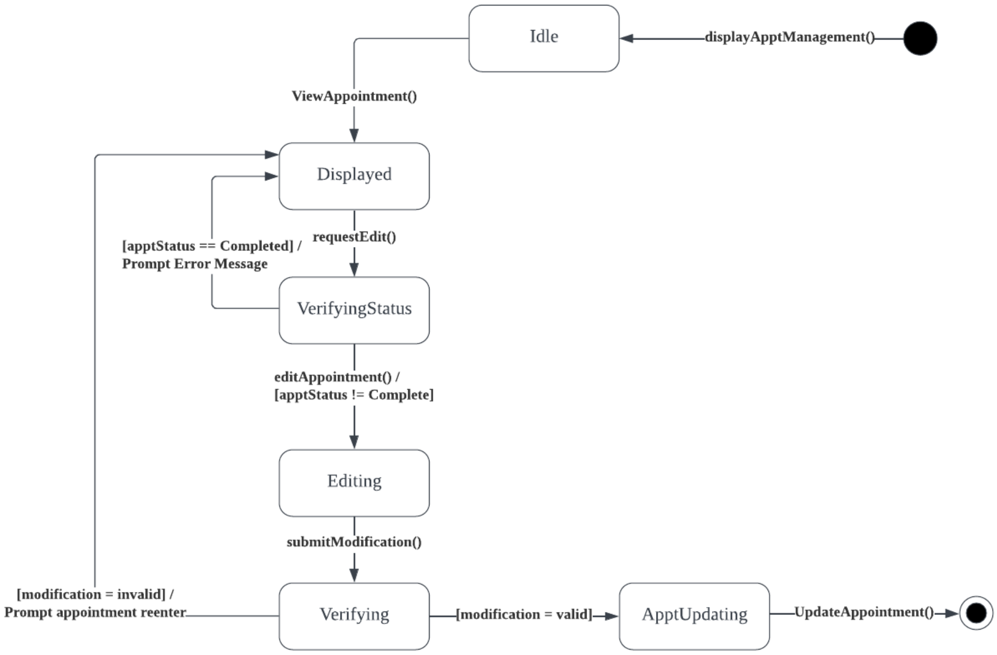


Figure 4.6.2.3.2: State Diagram of View and Edit Appointment

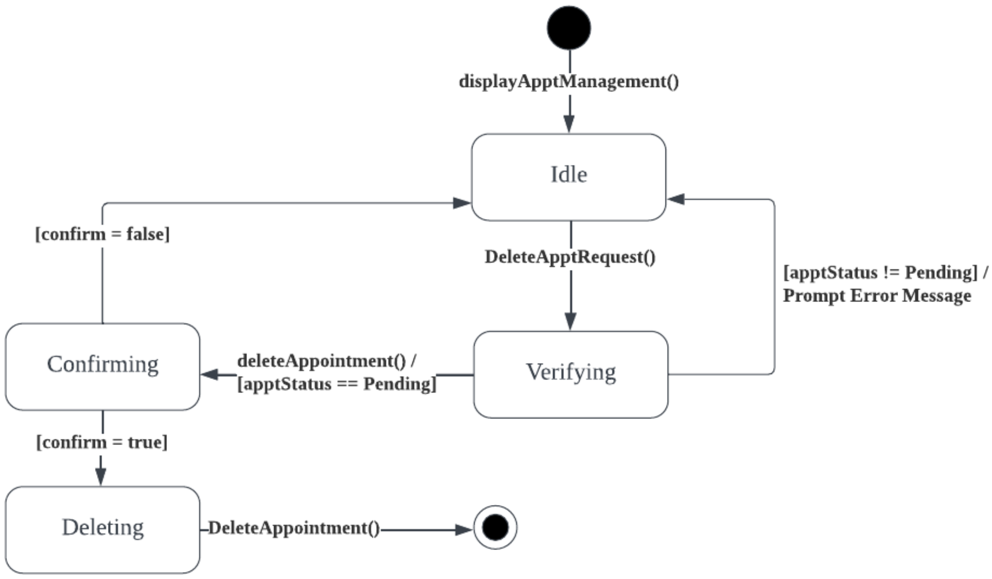


Figure 4.6.2.3.3: State Diagram of Cancel Appointment

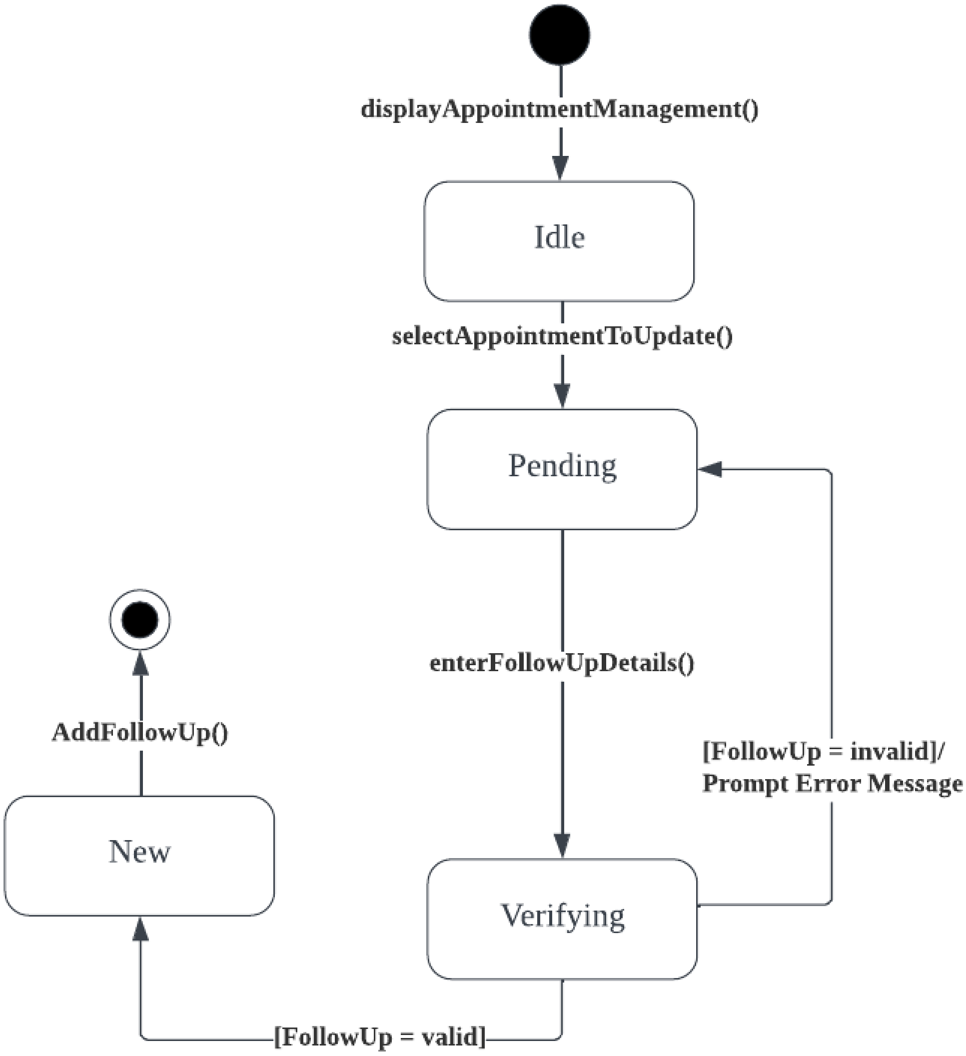


Figure 4.6.2.3.4: State Diagram of Update Appointment

4.6.2.4 Patient Module

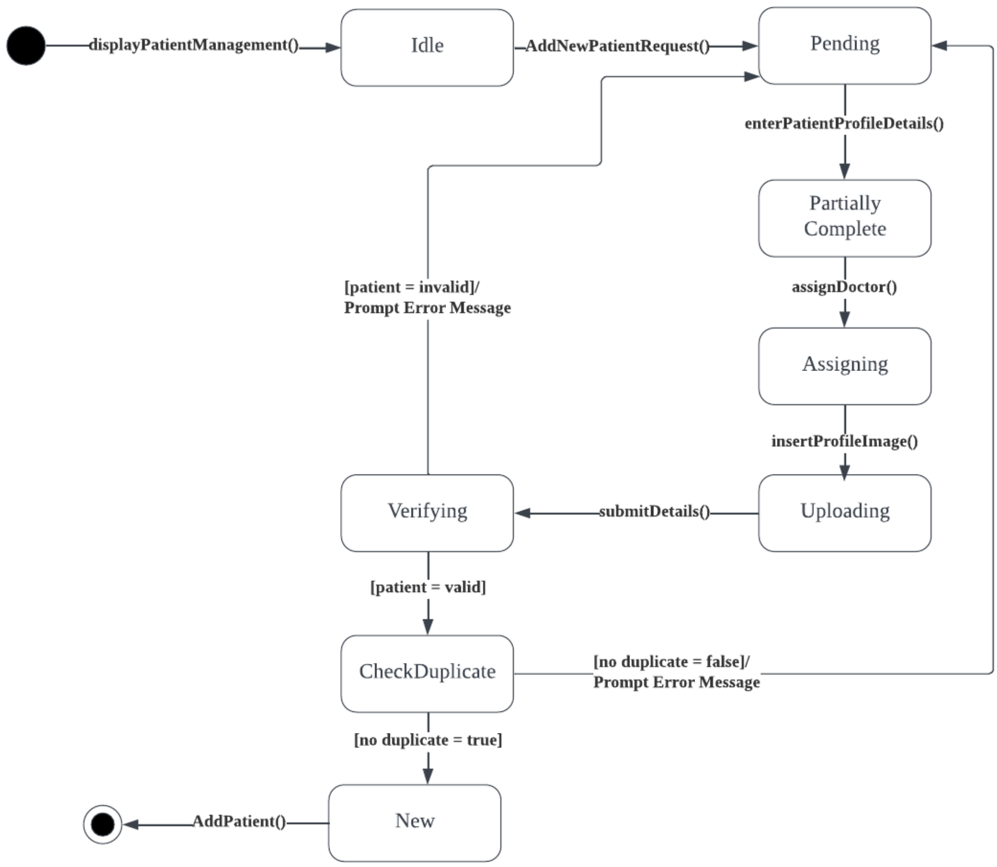


Figure 4.6.2.4.1: State Diagram of Add Patient

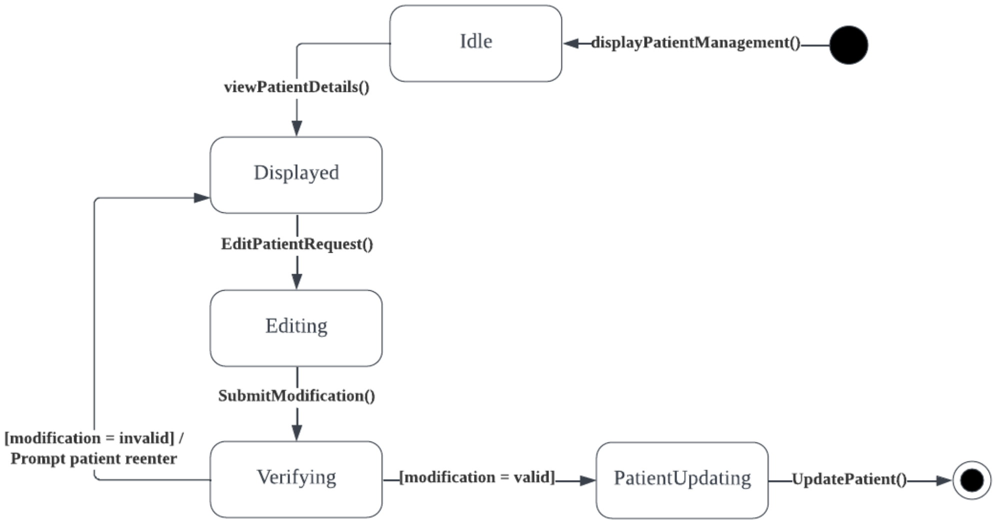


Figure 4.6.2.4.2: State Diagram of View and Edit Patient

4.6.2.5 Ingredient Scanning Module

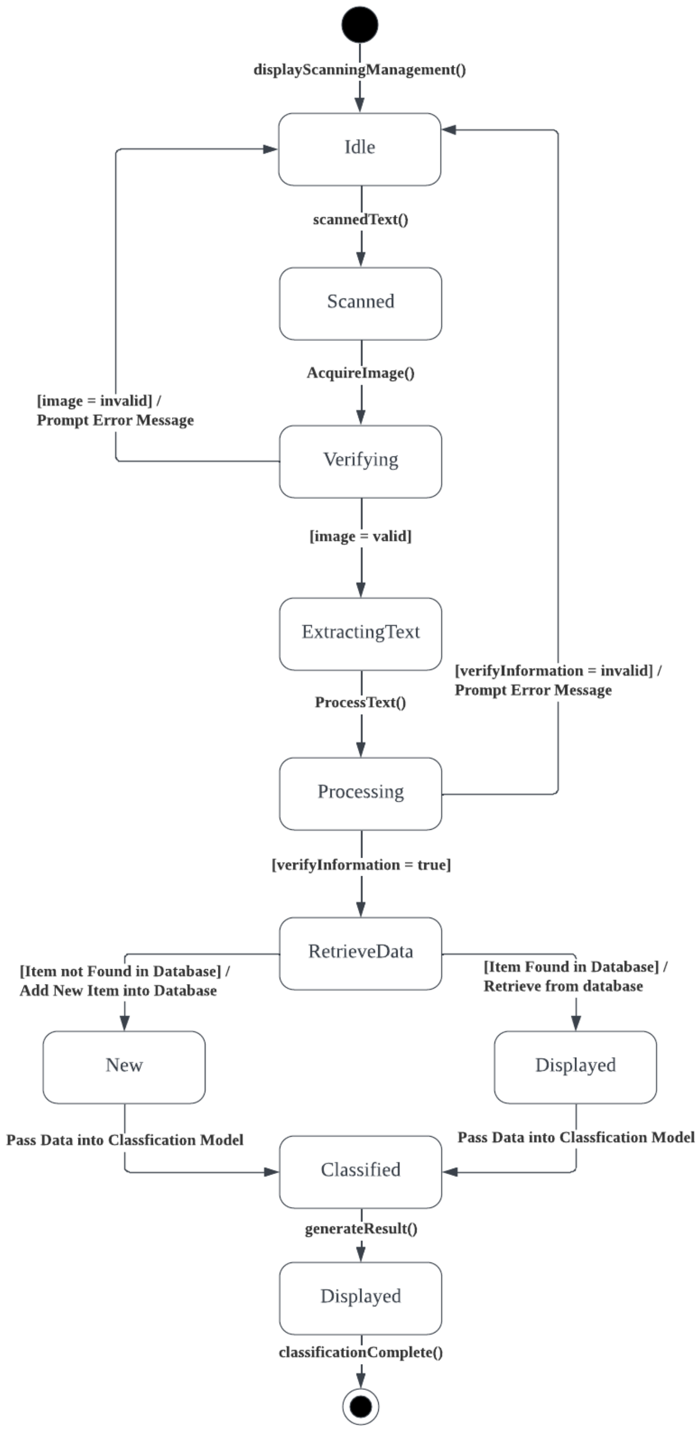


Figure 4.6.2.5.1: State Diagram of Ingredient Scanning

4.6.2.6 Blood Glucose Module

Figure 4.6.2.6.1: State Diagram of Manage Blood Glucose Reading

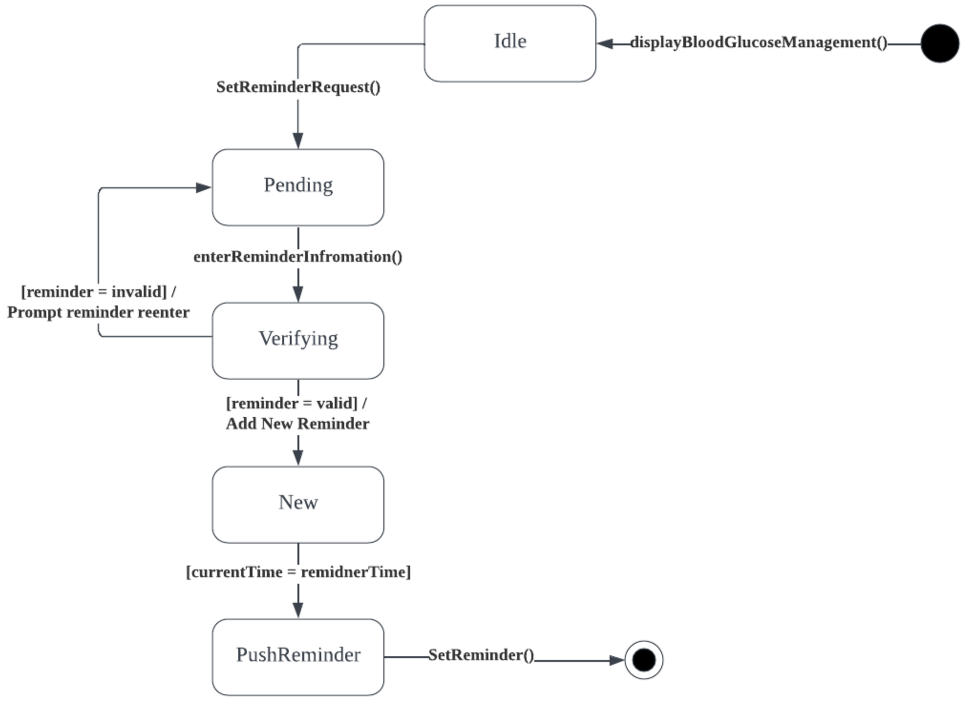


Figure 4.6.2.6.2: State Diagram of Set Reminder

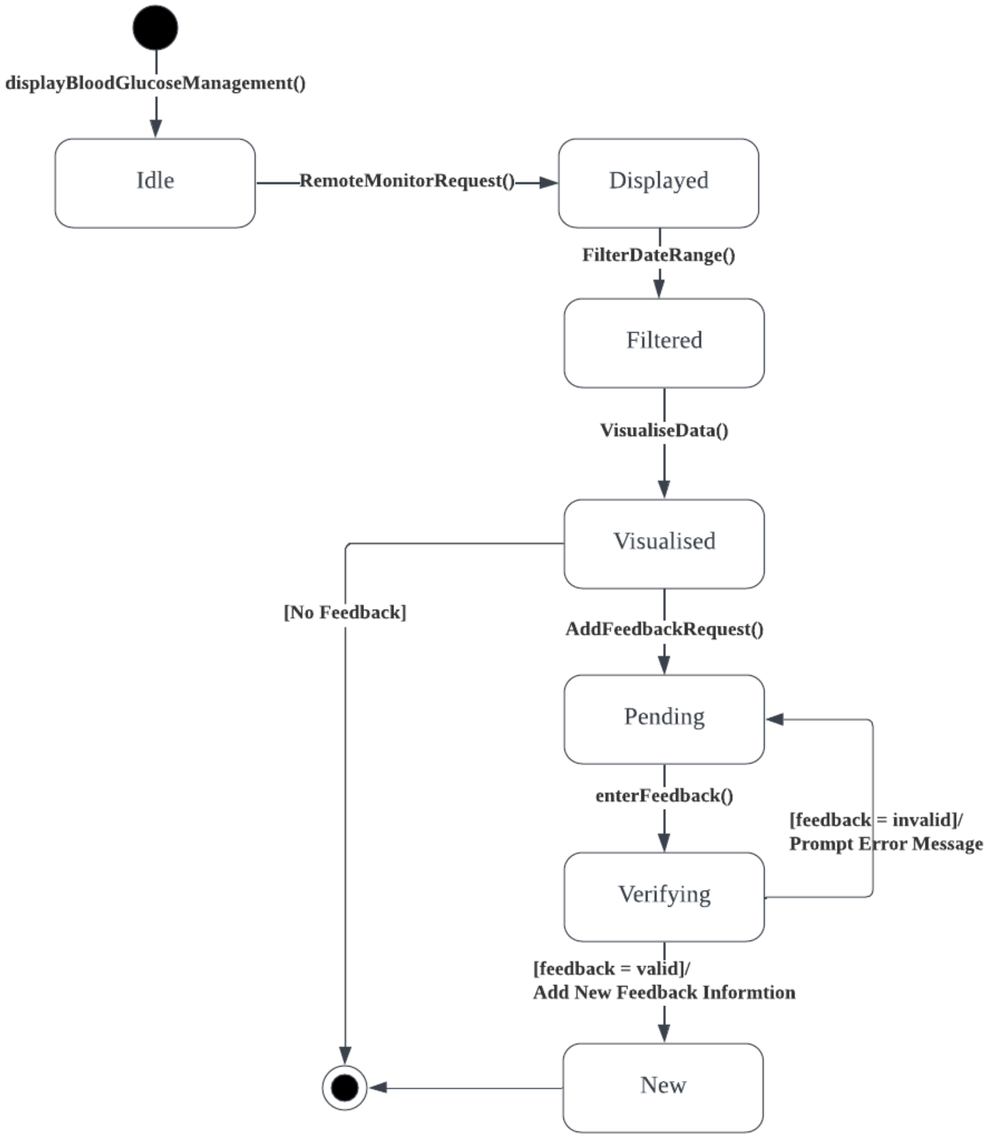


Figure 4.6.2.6.3: State Diagram of Remote Monitor

4.7 Software Architecture Design

4.7.1 Package Diagram

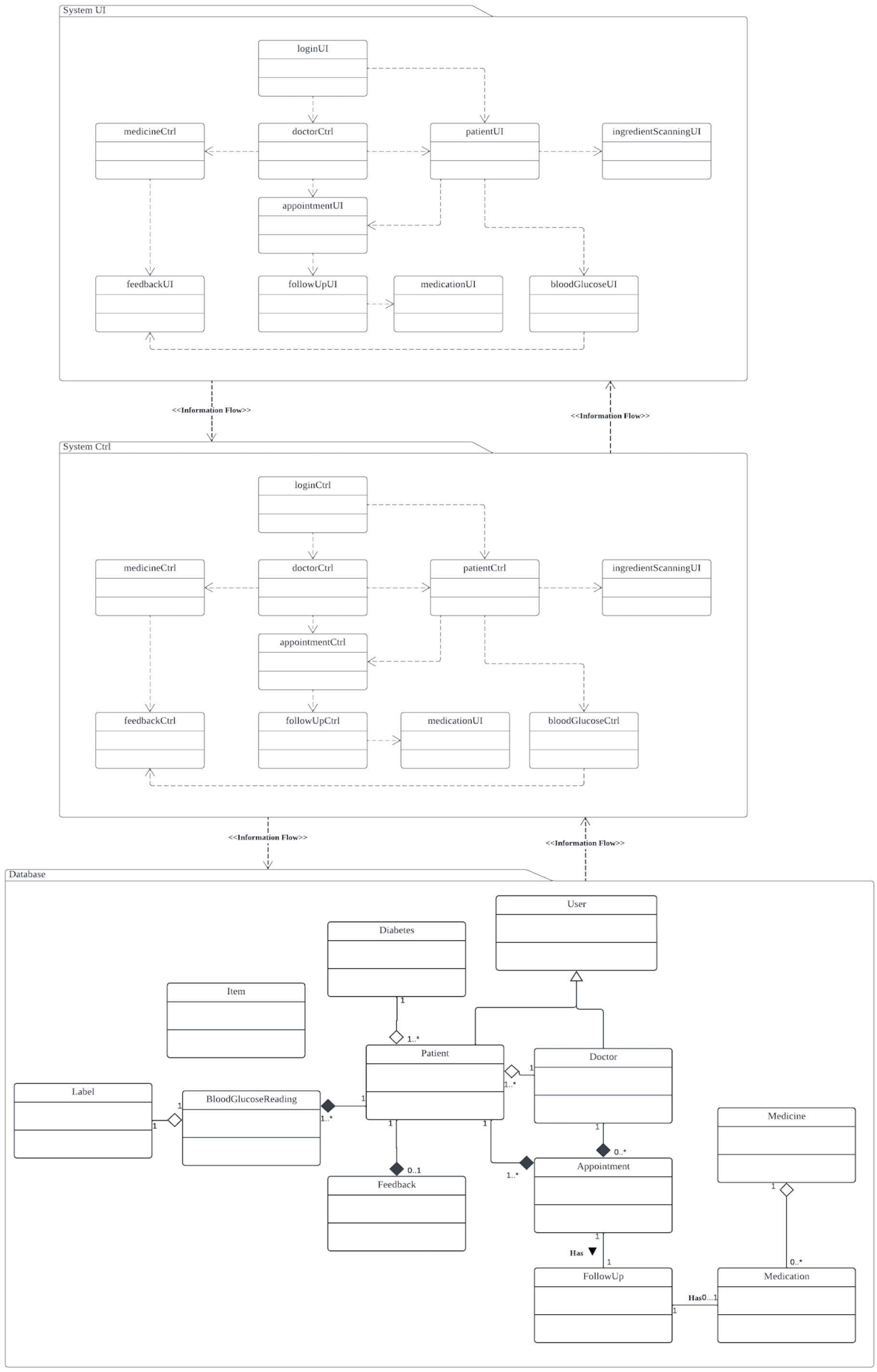


Figure 4.7.1.1: Package Diagram of Diabetes Monitoring System

4.7.2 Deployment Diagram

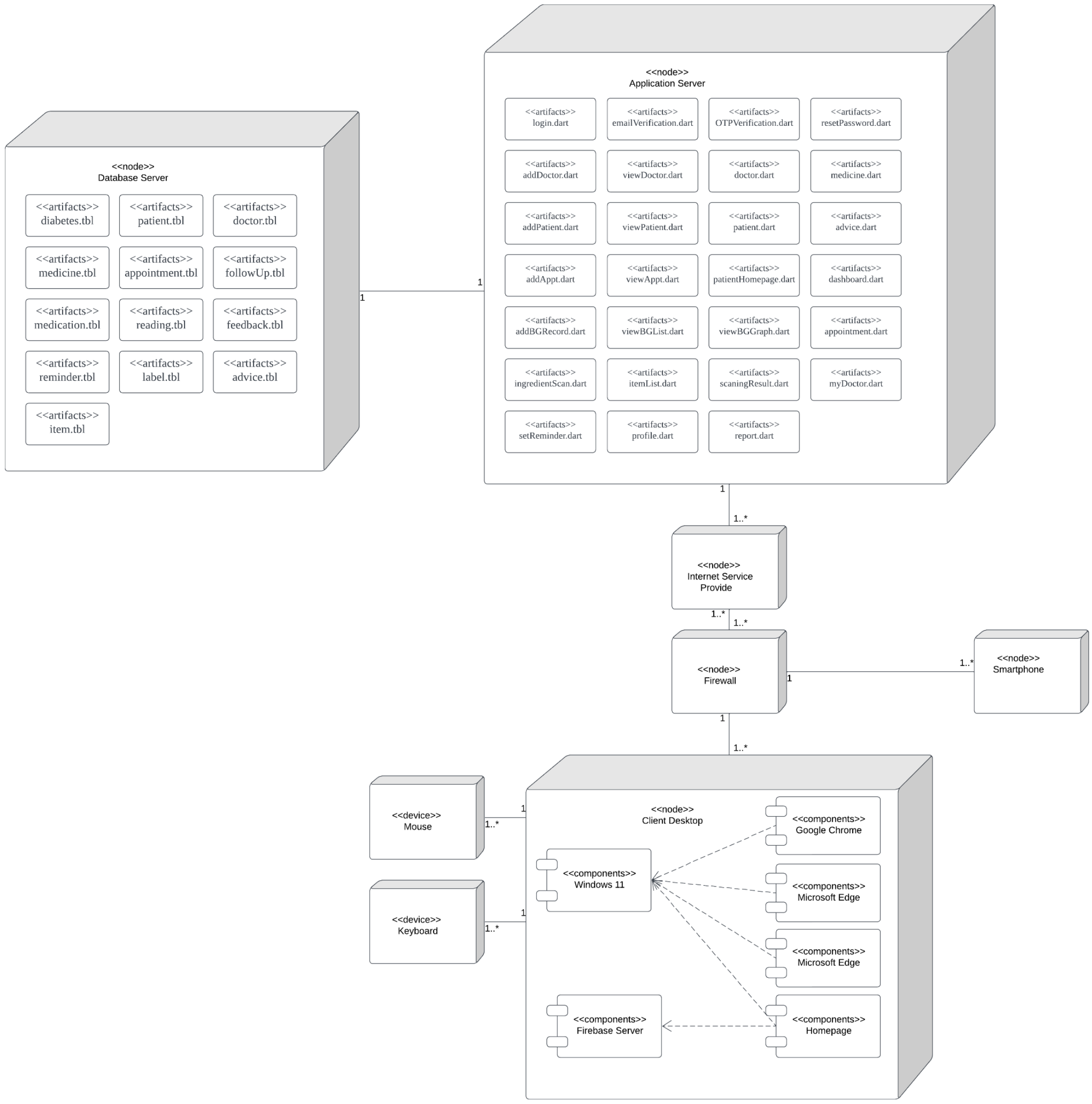


Figure 4.7.2.1: Deployment Diagram of Diabetes Monitoring System

4.8 Chapter Summary and Evaluation

In summary, this chapter focuses on designing various types of diagrams to illustrate the system's details effectively. The types of diagrams included are entity relationship diagram (ERD), class diagram, sequence diagram, state diagram, package diagram and deployment diagram. ERD will show the structure of the data and their relationship while the class diagram will depict the system structure in classes, attributes and methods. Sequence diagrams are used to visualise the flow and order of processes within the system for each module, showing the interactions between different components over time. State diagrams define the behaviour and status of the system throughout the operations within various modules. Furthermore, package and deployment diagrams help to understand the software architecture and components we need for our proposed project. Not only that, a data dictionary is included in this chapter to serve as a reference to clarify the meaning, purpose and attributes of the variables to be used within the system.

For the implementation of the algorithm, k-nearest neighbours are selected for the proposed system to perform the ingredient classification task. To ensure the security of the system, some services from the Firebase will be utilised to manage the authentication and authorisation of the users of the system.

Chapter 5

Implementation and Testing

5.1 Implementation

5.1.1 Dataset Preparation

The dataset used to train the classification model will be created via the following implementation approach due to the absence of an appropriate existing dataset. Before executing the following code, a list of food products and their associated ingredient items will be collected. An example of the list is shown in the figure below:

```
# Specify the path to your Excel file and the range of rows to read
api_key = '79417edaa2c246689b55bf06036b702b'
file_path = "ingredient.xlsx"


df = pd.read_excel(file_path)
df.head()
```

Figure 5.1.1.1: Code to Load Dataset

	food	item	item 2	item 3	item 4	item 5	item 6	item 7	item 8	item 9	item 10	Category
0	Hummus Chips Lightly Salted with Sea Salt	Lentils (64%)	Chickpeas (16%)	Rice Bran Oil	Baking Soda	Cumin Seeds	Sea Salt	Seasoning	NaN	NaN	NaN	Yes
1	Dark Chocolate Brownie Chips	coconut sugar	almond flour	wheat flour	cocoa powder	unsweetened coconut shreds	coconut oil	egg white powder	almond flakes	chia seeds	sea salt	Yes
2	three wishes grain free cereal	chickpea	tapioca	pea protein	organic cane sugar	cinnamon	natural flavors	salt	monk fruit	NaN	NaN	Yes
3	Fifty 50 peanut butter	Peanuts	hydrogenated vegetable oil	rapeseed	cottonseed	soybean	NaN	NaN	NaN	NaN	NaN	Yes
4	highkey mini chocolate bar	almond flour	erythritol	monk fruit	stevia	chocolate chip	coconut oil	butter	dried eggs	natural vanilla flavor	baking powder	Yes

Figure 5.1.1.2: Sample Rows of Dataset

Based on the figure above, the first column describes the name of the food product, acting as an identifier for that particular row. The following columns show the ingredients items of the corresponding food product, which takes a maximum of 10 items from the ingredient list of the food packaging. The last column of the row indicates the suitability of consumption for that particular food product. These data are collected via the Internet by searching those diabetes-friendly snacks and those that are unfriendly, recording them one by one into an excel file.

```python
def excel_to_ingredient_list(row_value):
    try:
        # Format each value and add it to the ingredient list
        ingredient_list = [f"{value}" for value in row_value if pd.notna(value)]
        return ingredient_list

    except Exception as e:
        print(f"An error occurred: {e}")
        return None
```

Figure 5.1.1.3: Code to Retrieve Ingredient Items from Excel Based on Row

The code above specifies the extraction of ingredient items from a particular row, defined by the parameter variable “row_value”. Based on the code above, it will extract the available items in the row specified and return as a list.

```python
def calculate_glycemic_index(api_key, ingredients):
    # Spoonacular API endpoint for glycemic load calculation
    endpoint = "https://api.spoonacular.com/food/ingredients/glycemicLoad"

    # Spoonacular API parameters
    params = {
        "apiKey": api_key,
    }

    # Request payload
    payload = {
        "ingredients": ingredients,
    }

    try:
        # Make the API request
        response = requests.post(endpoint, params=params, json=payload)
        response.raise_for_status()  # Check for errors

        # Parse the JSON response
        data = response.json()

        # Extract total glycemic load and individual ingredient information
        ingredient_details = data.get('ingredients', [])

        # glycemic_index = data.get('glycemicIndex', 'N/A')

        return ingredient_details

    except requests.exceptions.RequestException as e:
        print(f"Error making API request: {e}")
        return None, None
```

Figure 5.1.1.4: Code to Retrieve Glycemic Related Information via API

The code above demonstrates the execution of glycemic information retrieval via an API, known as Spoonacular. To use this API, the API key is identified and passed as a request associated with the ingredient list extracted from the Excel. The response by the API includes the total glycemic index of the entire ingredient list, the glycemic index and the glycemic load of each ingredient item.

```
def write_to_excel(file_path, ingredient_details):
    # Try to read the existing Excel file if it exists
    try:
        df = pd.read_excel(file_path)

        # Add a new row with glycemic indices for each ingredient list
        new_row = {}
        for i, ingredient in enumerate(ingredient_details):
            new_row = {'Total Glycemic Load': ingredient['totalGlycemicLoad']}
            glycemic_index = ingredient.get('glycemicIndex', 'N/A')
            # Replace 'N/A' with 0
            glycemic_index = 0 if glycemic_index == 'N/A' else glycemic_index
            new_row[f'Glycemic Index {i + 1}'] = glycemic_index
            new_row[f'Glycemic Load {i + 1}'] = ingredient['glycemicLoad']

        df = df.append(new_row, ignore_index=True)

    except FileNotFoundError:
        # If the file doesn't exist, create a new DataFrame
        print("Creating a new file.")
        df = pd.DataFrame({})
        new_row = {}
        for i, ingredient in enumerate(ingredient_details):
            glycemic_index = ingredient.get('glycemicIndex', 'N/A')
            # Replace 'N/A' with 0
            glycemic_index = 0 if glycemic_index == 'N/A' else glycemic_index
            new_row[f'Glycemic Index {i + 1}'] = glycemic_index
            new_row[f'Glycemic Load {i + 1}'] = ingredient['glycemicLoad']

        df = df.append(new_row, ignore_index=True)

    # Write the DataFrame to the Excel file
    df.to_excel(file_path, index=False)
    print(f"Data written to {file_path}")
```

Figure 5.1.1.5: Code to Load Glycemic Information to Excel

This portion of the code specifies the creation of a dataset that will be used for model training. Based on the glycemic information retrieved from the API, the information is recorded into another Excel file, with each ingredient list as a new row. Within this part of the code, some data preprocessing is included such as when the API is unable to provide the glycemic information for a particular ingredient item, providing an 'N/A' value, it will automatically update the value to 0 in the Excel.

```python
try:
    df = pd.read_excel(file_path, header=None)

    # Iterate through each row and generate the ingredient list
    for index, row in df.iterrows():
        ingredients = excel_to_ingredient_list(row)

        print('asdfghj ',index+1)
        ingredient_details = calculate_glycemic_index(api_key, ingredients)
        print()
        print()
        print('Row ',index+1)

        if ingredient_details is not None:
            print("Individual Ingredient Details:")
            for ingredient in ingredient_details:
                print(f" - {ingredient.get('original', 'N/A')}: Glycemic Index {ingredient.get('glycemicIndex', 'N/A')}")

            excel_file_path = 'glycemic_index_data.xlsx'
            write_to_excel(excel_file_path, ingredient_details)

        else:
            print("Error calculating total glycemic load.")


except Exception as e:
    print(f"An error occurred: {e}")
```

Figure 5.1.1.6: Code to Execution

```
Row  1
Individual Ingredient Details:
 - 100g Palm Fruit Oil: Glycemic Index 0.0
 - 100g Pea Protein: Glycemic Index 0.0
 - 100g Allulose: Glycemic Index 0.0
 - 100g Cane Sugar: Glycemic Index 70.09
 - 100g Cassava Flour: Glycemic Index 0.0
 - 100g Chicory Root Fiber: Glycemic Index 45.0
 - 100g Tapioca Starch: Glycemic Index 0.0
 - 100g Xanthan Gum: Glycemic Index 0.0
 - 100g Natural Flavor: Glycemic Index 100.0
 - 100g Sea Salt: Glycemic Index 0.0
 - 100g Baking Powder: Glycemic Index 92.0
 - 100g Sunflower Lecithin: Glycemic Index 0.0
 - 100g Turmeric: Glycemic Index 5.0
 - 100g Citric Acid: Glycemic Index 0.0
 - 100g Stevia Extract: Glycemic Index 0.0
Creating a new file.
Data written to glycemic_index_data.xlsx
asdfghj  2
<ipython-input-4-f6e512c2fcc2>:27: FutureWarning: The frame.append method is deprecated and will be removed from pandas in a future version. Use pandas.concat instead.
  df = df.append(new_row, ignore_index=True)


Row  2
Individual Ingredient Details:
 - 100g peanut butter: Glycemic Index 14.0
 - 100g cane sugar: Glycemic Index 70.09
 - 100g plant protein blend: Glycemic Index 30.0
 - 100g dark chocolate chips: Glycemic Index 0.0
 - 100g vegetable glycerin: Glycemic Index 45.0
 - 100g gluten free oat flout: Glycemic Index 54.5
 - 100g oat flour: Glycemic Index 0.0
 - 100g dutch cocoa powder: Glycemic Index 0.0
 - 100g potato starch: Glycemic Index 0.0
 - 100g oat fiber: Glycemic Index 54.5
 - 100g baking powder: Glycemic Index 92.0
Data written to glycemic_index_data.xlsx
asdfghj  3
<ipython-input-4-f6e512c2fcc2>:14: FutureWarning: The frame.append method is deprecated and will be removed from pandas in a future version. Use pandas.concat instead.
  df = df.append(new_row, ignore_index=True)
```

Figure 5.1.1.7: Sample Output of Execution

The above images show the sample of running the code execution and API, returning the result of the glycemic information.

```
# Load the data from Excel
excel_file_path = 'Book3.xlsx'
df = pd.read_excel(excel_file_path)


df.head()
```

	food	Total Glycemic Load	Category	Glycemic Index 1	Glycemic Load 1	Glycemic Index 2	Glycemic Load 2	Glycemic Index 3	Glycemic Load 3	Glycemic Index 4	Glycemic Load 4	Glycemic Index 5	Glycemic Load 5	Category.1
0	Hummus Chips Lightly Salted with Sea Salt	10.0	carbohydrates	28.0	48.0	48.0	25.0	45.0	54.0	48.0	15.0	39.0	59.0	No
1	Dark Chocolate Brownie Chips	10.0	carbohydrates	25.0	17.0	17.0	14.0	22.0	78.0	51.0	59.0	26.0	78.0	No
2	three wishes grain free cereal	10.0	carbohydrates	41.0	50.0	11.0	42.0	41.0	78.0	32.0	69.0	65.0	54.0	No
3	Fifty 50 peanut butter	10.0	carbohydrates	21.0	48.0	21.0	10.0	31.0	33.0	21.0	21.0	47.0	68.0	No
4	highkey mini chocolate bar	12.0	carbohydrates	13.0	28.0	22.0	34.0	18.0	20.0	56.0	49.0	25.0	54.0	No

Figure 5.1.1.8: Sample Rows of Dataset Created

In general, the generated dataset for the model training will look like the above. The first column indicates the name of the food product. The following column will be the category of the food, this category is retrieved from another model, which the steps will be explained later. Starting from the third column, it will be glycemic-related information such as total glycemic load, glycemic index and glycemic load value for each ingredient item. The last column in the dataset will be the category of this food item in terms of consumption suitability.

The category consists of five common values, carbohydrates, protein, fruit, vegetables and fats. While preparing the dataset, this category is retrieved by inputting the nutrition facts of the food product into a classification model and the outcome will be its category. The following shows the steps of training the category classification model.

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.metrics import accuracy_score
import joblib
import tensorflow as tf
from tensorflow.keras import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.utils import to_categorical
```

Figure 5.1.1.9: Code to Import Library

The necessary libraries are imported for the later use of the model training. This includes libraries such as pandas to handle the dataset, processing the data as desired and numpy to work with the array. Standard Scalar is included to remove the mean and scales each feature/variable to unit variance, ensuring the performance of feature-wise in an independent way while the LabelEncoder is used to change the categorical label into numerical format, ensuring better handling by the model.

```
# Load the data from Excel
excel_file_path = 'Book2.xlsx'
df = pd.read_csv("nutrients_category.csv")

df.head()
```

	Food	Grams	Calories	Protein	Fat	Sat.Fat	Fiber	Carbs	Category
0	Cows' milk	976	660	32	40	36	0.0	48.0	protein
1	Milk skim	984	360	36	0	0	0.0	52.0	protein
2	Buttermilk	246	127	9	5	4	0.0	13.0	protein
3	Evaporated, undiluted	252	345	16	20	18	0.0	24.0	protein
4	Fortified milk	1419	1373	89	42	23	1.4	119.0	protein

Figure 5.1.1.10: Code to Load Dataset and Sample Rows of Dataset

The dataset used to train the category classification consists of 9 columns as shown above. The first column indicates a unique value, showing the name of the food, and the possible ingredient items. The following 7 columns are the attributes that will be used for the model training, retrieving from the nutrition facts of the food packaging. The last column will define the category of the food items based on their attributes.

```
# Assuming the first column contains labels or IDs, and the rest are features (features are columns 2 to 8)
features = df.iloc[:, 1:8]

# Column 9 is the classification variable (y)
labels = df.iloc[:, 8]
```

```
label_encoder = LabelEncoder()
labels_encoded = label_encoder.fit_transform(labels)
```

```
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(features, labels_encoded, test_size=0.2, random_state=42)
```

Figure 5.1.1.11: Code to Preprocess and Split Dataset

The feature included for the model training comes from the value of columns 2 to column 8, excluding the column of food name as these unique values are not able to contribute to model training. The library LabelEncoder is used to convert the classification labels into numeric values for classification purposes. The dataset is later split into train data and test data with a ratio of 80 to 20.

```
# Standardize the features
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

Figure 5.1.1.12: Code to Normalise Dataset

StandardScaler library is used to standardise the features before passing them for model training, ensuring the scale of each feature's value will not dominate the classification result.

```
unique_categories = labels.unique()
num_classes = len(unique_categories)

y_train_one_hot = to_categorical(y_train, num_classes=num_classes)
y_test_one_hot = to_categorical(y_test, num_classes=num_classes)

model = Sequential([
    Dense(64, activation='relu', input_shape=(X_train.shape[1],)),
    Dense(num_classes, activation='softmax')
])
```

Figure 5.1.1.13: Code to Train Model or Classification

As the last column of the dataset holds the category of classification, now stored in the variable 'labels', therefore, we will retrieve the categories through the method, unique(). This returns the five categories in Excel. When all the preprocessing has been completed, model training takes place, utilising the data that has been split and processed.

```
# convert into tflite file
from google.colab import files
converter = tf.lite.TFLiteConverter.from_keras_model(model)
tflite_model = converter.convert()
open("food_category.tflite", "wb").write(tflite_model)
files.download('food_category.tflite')
```

Figure 5.1.1.14: Code to Export Trained Model

The above code demonstrates the step to convert the trained model into a tflite file, which is compatible to work in the Flutter framework, and able to perform optimally. This model will be utilised to classify the category of the food product based on the nutrition facts and the outcome will be one of the attributes of another model, shown below, to carry out the classification on the consumption suitability.

5.1.2 Model Implementation

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score
import joblib
import tensorflow as tf
from tensorflow.keras import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.utils import to_categorical
```

Figure 5.1.2.1: Code to Import Library

Importing the necessary libraries to support the code execution, and ensure the performance of the model.

```
# Load the data from Excel
excel_file_path = 'Book2.xlsx'
df = pd.read_excel(excel_file_path)

# Assuming the first column contains labels or IDs, and the rest are features (features are columns 2 to 12)
features = df.iloc[:, 1:13]

# Column 13 is the classification variable (y)
labels = df.iloc[:, 13]
```

Figure 5.1.2.2: Code to Load Dataset

The code above specifies the Excel file used for model training, named ‘Book2.xlsx’. Based on Excel, there are a total of 13 columns consisting of data. However, as the first column contains the name of the food products, it will not be included as a feature of training. The features included in the training are data from columns 2 to column 12 while the last column will act as the label, describing the classification result.

```
# Convert 'yes' and 'no' to 1 and 0 for binary classification
label_encoder = LabelEncoder()
labels_encoded = label_encoder.fit_transform(labels)
```

```
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(features, labels_encoded, test_size=0.2, random_state=42)
```

```
# Standardize the features (optional but recommended for KNN)
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

Figure 5.1.2.3: Code to Preprocess and Split Dataset

The library LabelEncoder is used to change the classification labels into binary values, converting the ‘yes’ and ‘no’ to 1 and 0 for classification purposes. The dataset is later split into train data and test data with a ratio of 80 to 20. StandardScaler library is used to standardise the features before passing them for model training, ensuring the scale of each feature’s value will not dominate the classification result.

```
# Create a KNN classifier - find better accuracy
for k in range(2, 10):
    knn_classifier = KNeighborsClassifier(n_neighbors=k)

    # Train the classifier
    knn_classifier.fit(X_train, y_train)

    # Make predictions on the test set
    y_pred = knn_classifier.predict(X_test)

    # Evaluate the accuracy of the classifier
    accuracy = accuracy_score(y_test, y_pred)
    print(k)
    print(f'Accuracy: {accuracy}')
```

Figure 5.1.2.4: Code to KNN Classification Experimenting to Get Better Accuracy

The process above includes a loop that experiments with the training of a KNN model, attempting to figure out the most suitable neighbour, the k-value, with the highest possible accuracy.

```
knn = KNeighborsClassifier(n_neighbors=6)

# Fit the KNN model with the training data
knn.fit(X_train, y_train)

# Predict using the KNN model
knn_predictions = knn.predict(X_test)

knn_predictions = knn_predictions[:len(y_test)]
```

Figure 5.1.2.5: Code to KNN Classification Model Training

As a result, 6 has been chosen as the number of neighbours as it has the best and most satisfying performance.

```
# convert into tflite file
from google.colab import files
converter = tf.lite.TFLiteConverter.from_keras_model(model)
tflite_model = converter.convert()
open("glycemic_classification.tflite", "wb").write(tflite_model)
files.download('glycemic_classification.tflite')
```

Figure 5.1.2.6: Code to Export Trained Model

This part of the code demonstrates how the system converts the trained model into a tensorflow lite file(tflite), which is compatible with working and integrated with the Flutter project. With this model, the system will be able to perform the classification on consumption suitability once it has collected all the attributes required such as the glycemic information of each ingredient item and the possible category of the food item, which will be retrieved from another model.

5.1.3 Integration with Flutter

```
assets:
  - assets/classification.tflite
  - assets/food_category.tflite
  - assets/images/
```

Figure 5.1.3.1: Code to Include Exported Model as Asset

Before running the trained model in the Flutter project to carry out necessary classification, the exported 'tflite' model must be included as the assets. Without this step, the system cannot access the trained model even if the code is well implemented, attempting to return an error when accessed.

```
class ModelResult extends StatefulWidget {
  final List<double> inputData;
  final List<IngredientModel> items;
  final String category;

  const ModelResult({super.key, required this.inputData, required this.items, required this.category});

  @override
  State<ModelResult> createState() => _ModelResultState();
}
```

Figure 5.1.3.2: Code of Passing Input as Parameter

The above code demonstrates the integration of the trained model into the Flutter project using a stateful widget. The stateful widget is used to keep the same state object when moving from one location to another. In this stateful widget named ModelResult, three variables are required as parameters, known as the input data, items and the category. The input data represents a list of double values, consisting of the glycemic index and load for each ingredient item while the items stored the list of ingredient items being extracted from the ingredient list. The category will be used to store the category of that particular food product based on the nutritional fact value.

```
Future<void> loadModel() async {
  try {
    final modelExists = await rootBundle.load('assets/classification.tflite');
    if (modelExists != null) {
      print('Model file exist.');
      final interpreter = await tfl.Interpreter.fromBuffer(
        (await rootBundle.load('assets/classification.tflite')).buffer.asUint8List(),
      );

      classifyInputData(interpreter);
    } else {
      print('Model file does not exist.');
    }
  } catch (e) {
    print('Error loading model: $e');
  }
}
```

Figure 5.1.3.3: Code to Load Model

The code above shows the model loading into the system. When the model, classification.tflite, exists and is found as an asset in the Flutter project, it will carry out the classification, and invoke the relevant functions.

```
Map<int, String> classIndexToLabel = {
  0: 'No',
  1: 'Yes',
};
```

Figure 5.1.3.4: Code to Map Classification Result to Actual Label

As the label encoder has been utilised during model training to convert the classes of food categories into numeric values, therefore a mapping of the integer value, which represents the classification result, to the actual label should be included. This ensures the result conveyed to the users is understandable and readable.

```
Future<void> classifyInputData(tfl.Interpreter interpreter) async {
  try {
    // Prepare the input tensor
    interpreter.allocateTensors();
    interpreter.getInputTensor(0).data = Float32List.fromList(widget.inputData).buffer.asUint8List();

    // Run inference
    interpreter.invoke();

    // Get the output tensor and interpret the result
    var output = interpreter.getOutputTensor(0);

    if (output != null) {
      var outputData = output.data;
      if (outputData != null) {
        // Assuming the model outputs probabilities for each class
        var probabilities = Float32List.sublistView(outputData);

        // Find the index with the highest probability
        var predictedClassIndex = probabilities.indexOf(probabilities.reduce((a, b) => a > b ? a : b));

        // Retrieve the corresponding label from classes
        var predictedLabel = classIndexToLabel[predictedClassIndex];
```

```
      }
  } catch (e) {
    print('Error during inference: $e');
  }
}
```

Figure 5.1.3.5: Code to Execute Classification

The code above demonstrates the execution of the loaded classification model. Firstly, it inputs the list of double values, storing the list of glycemic load and index values, into the model. The model then processes the data inputted and performs the necessary classification process. When there is an output from the classification, the system will convert the output based on the mapped value, receiving the actual label.

5.2 Test Plan

5.2.1 Test Scope

Table 5.2.1.1: Test Scope Table

IS	IS NOT
All functional requirements ● Login module ● Doctor module ● Patient module ● Appointment module ● Blood Glucose module ● Ingredient Scanning module ● Report module	Scalability
Localization (English only)	Localization (other than Malay and English)
User interface	Network compatibility
Web browser compatibility	Network file sharing options
Mobile devices compatibility	Performance
Error handling	Maintainability
Reliability	Security and Privacy
Usability	

5.3 Testing Strategies / Approaches

System testing is one of the critical phases in the development process aimed at ensuring the deliverables meet and fulfil the stakeholder requirements. Its primary goal is to guarantee a high level of satisfaction with the deliverables and ensure the deliverables are errorless, thereby preventing the release of products that deviate from the actual user demand or system failure. In the context of the proposed system, a comprehensive testing strategy will be employed, encompassing various testing methodologies such as unit testing, integration testing, system testing and acceptance testing.

5.3.1 Test SubProcesses

The incremental approach has been selected for testing due to its traceability and accountability. This approach performs independent testing on individual modules, gradually integrating the modules and eventually testing the entire system which consists of all modules. This approach of testing makes the verification of functionalities and performance of each component easy as it covers a wide range, performing thorough testing and providing a clear visibility into the progress of testing efforts. Furthermore, incremental testing is able to enhance issue identification and resolution. As testing is carried out at a granular level, defects or problems existing can be identified immediately and resolved in time. This prevents the problems from propagating to higher levels of the system, escalating the failure risk of the entire system. By figuring out and fixing defects early in the development process, the overall quality of the system can be enhanced, ensuring the system delivers a satisfaction rate from the stakeholders. While the testing progresses, starting with lower-level components or modules, the isolation makes it easier to identify and fix issues.

Under the incremental approach, the bottom-up strategy is selected as the proposed diabetes monitoring system has several modules and the integration among these modules should be included in the testing. Bottom-up strategy in the incremental approach testing methods are software testing methods in which testing is done incrementally, starting with lower-level components or modules and gradually integrating them into larger subsystems or complete systems. It allows for early integration of lower-level modules, which can help identify and fix issues at the lower levels before moving to higher-level modules. This approach is suitable when a major flaw is suspected to occur towards the bottom of the program. This suits the proposed diabetes monitoring system as the error is more likely to happen on the individual module rather than the integration of several modules.

5.3.2 Testing Design Techniques

5.3.2.1 Unit Testing

Unit testing refers to the examination of individual components or modules in isolation to validate and verify their functionalities, comparing their outcome to the expected results. The primary purpose of unit testing is to ensure that each specific feature or function works as intended before integrating them into a more complex system flow. In the context of the proposed system, a meticulous approach to unit testing will be adopted, with each functional unit tested separately. For example, within the patient module, functions such as add patient, edit patient, view patient, and deactivate patient will undergo individual unit testing. This ensures that each of these functions operates independently and functions correctly without encountering issues. Unit testing possesses the ability to quickly identify and pinpoint errors in certain parts of the system, making it more efficient than other testing methods. The isolation of components during testing minimises complexity, facilitating easier identification and localization of errors.

5.3.2.2 Integration Testing

Integration testing is another testing that focuses on validating the interactions between interconnected modules to ensure seamless collaboration. The testing includes verifying correct page navigation, accurate data transfer or processing, data retrieval between modules and so forth. The primary objective of integration testing is to evaluate the compliance of the system, ensuring the system can operate smoothly. For example, interaction testing will be carried out to the patient and doctor module in the proposed. A specific test scenario might involve validating the accurate retrieval of the patient list under a particular doctor. This process helps identify and rectify any issues that may arise from and after the integration of these modules. The integration testing begins with minimal module integration, aimed to verify the collaboration of a small set of modules, gradually including additional modules and eventually summing up all modules for the testing. Integration testing ensures a thorough testing of the system, ensuring the reliability of the system performance.

5.3.2.3 System Testing

During system testing, the software tester evaluates the performance of the entire system against the requirements specified. This involves performing testing on the functional and non-functional requirements, aiming to ensure the system delivered meets specified requirements and expectations. Furthermore, the functions and features of the system are validated and verified to ensure their ability to work accordingly, guaranteeing the quality of the system developed. The system testing is carried out to verify the system against business, functional and technical requirements that have been agreed on with its end users to discover bugs or defects before going for acceptance testing.

5.3.2.4 Acceptance Testing

Acceptance testing is another testing phase that evaluates the degree to which the system delivered satisfies the user's requirements or expectations to consider the system's readiness for development. The purpose of this testing is to ensure that the software satisfies the requirements and needs of the users, working as expected by the stakeholders. This testing is crucial as it helps to demonstrate the functions and process flow of the system in a manner that is suited to real-world tasks and circumstances according to user specifications. In acceptance testing, users interact with the system before its official release to check whether it contains any bugs or if any features may have been overlooked. The system will only proceed to deployment if the users provide approval after the acceptance test.

5.4 Test Cases & Test Results

5.4.1 Login Module

Table 5.4.1.1: Test Case for Login (Valid)

<table>
<tr><td colspan="3">Test Case #: T001</td><td colspan="4">Test Case Name: To test the login function</td></tr>
<tr><td colspan="3">System: Diabetes Monitoring System (Web)</td><td colspan="4">Subsystem: Login module</td></tr>
<tr><td colspan="3">Designed By: Ong Jia Hui</td><td colspan="4">Design Date: 21/12/2023</td></tr>
<tr><td colspan="3">Executed By: Ong Jia Hui</td><td colspan="4">Execution Date: 21/12/2023</td></tr>
<tr><td colspan="7">Short Description: To verify that the system correctly allows users access when valid credentials are entered during the login process.</td></tr>
<tr><td colspan="7">Pre-conditions:
The user has registered an account in the system.
The user logged in with credentials that belong to the role of a doctor in the system.</td></tr>
<tr><th>Step</th><th>Action / Step</th><th>Test Data</th><th>Expected System Response</th><th>Actual System Response</th><th>Pass/ Fail</th><th>Comments</th></tr>
<tr><td>1</td><td>Navigate to the login page.</td><td>-</td><td>The login page will be displayed.</td><td>Same as Expected</td><td>Pass</td><td>-</td></tr>
<tr><td>2</td><td>Enter valid credentials.</td><td>Valid email and password</td><td>N/A</td><td>Same as Expected</td><td>N/A</td><td>-</td></tr>
<tr><td>3</td><td>Click on the “Login” button.</td><td>-</td><td>An error message indicating invalid credentials will be displayed.</td><td>Same as Expected</td><td>Pass</td><td>-</td></tr>
</table>

Post-Conditions: The user is logged into his/her account and remains logged in until they manually log out or close the browsing window.

Table 5.4.1.2: Test Case for Login (Invalid)

Test Case #: T002			**Test Case Name:** To test the login function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Login module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly denies users access when invalid credentials are entered during the login process.						
Pre-conditions: The user has registered an account in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the login page.	-	The login page will be displayed.	Same as Expected	Pass	-
2	Enter invalid credentials.	Invalid email and password / Leave empty	N/A	-	N/A	-
3	Click on the “Login” button.	-	An error message indicating invalid credentials will be displayed.	Same as Expected	Pass	-

Post-Conditions: The user is not logged in and remains on the login page.

Table 5.4.1.3: Test Case for Register (Valid)

Test Case #: T003			**Test Case Name:** To test the register function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Login module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows users to register in the system.						
Pre-conditions: The user does not register an account in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the login page.	-	The login page will be displayed.	Same as Expected	Pass	-
2	Click on the “Guest Register” button.	-	The register page will be displayed.	Same as Expected	Pass	-
3	Fill in all necessary details to perform registration.	Valid Profile Data including name, IC No, email, contact, gender, doctor registration number and address.	N/A	-	N/A	-

4	Click on the “Complete” button.	-	A pop-out message with a doctors’ list will be displayed for the user's selection.	Same as Expected	Pass	-
5	Select the desired doctor.	-	The selected doctor ID will be displayed in the related text field.	Same as Expected	Pass	-
6	Click on the “Save” button.	-	A confirmation message on the register action will be displayed.	Same as Expected	Pass	-
7	Click “Yes” to proceed.	-	A completion message will be displayed.	Same as Expected	Pass	-
Post-Conditions: A new account is registered to the system.						

Table 5.4.1.4: Test Case for Register (Invalid)

<table>
<tr><td colspan="3">Test Case #: T004</td><td colspan="4">Test Case Name: To test the register function</td></tr>
<tr><td colspan="3">System: Diabetes Monitoring System (Mobile)</td><td colspan="4">Subsystem: Login module</td></tr>
<tr><td colspan="3">Designed By: Ong Jia Hui</td><td colspan="4">Design Date: 21/12/2023</td></tr>
<tr><td colspan="3">Executed By: Ong Jia Hui</td><td colspan="4">Execution Date: 21/12/2023</td></tr>
<tr><td colspan="7">Short Description: To verify that the system correctly denies users to register in the system when invalid data are entered during the login process.</td></tr>
<tr><td colspan="7">Pre-conditions:
The user does not register an account in the system.</td></tr>
<tr><th>Step</th><th>Action / Step</th><th>Test Data</th><th>Expected System Response</th><th>Actual System Response</th><th>Pass/ Fail</th><th>Comments</th></tr>
<tr><td>1</td><td>Navigate to the login page.</td><td>-</td><td>The login page will be displayed.</td><td>Same as Expected</td><td>Pass</td><td>-</td></tr>
<tr><td>2</td><td>Click on the “Guest Register” button.</td><td>-</td><td>The register page will be displayed.</td><td>Same as Expected</td><td>Pass</td><td>-</td></tr>
<tr><td>3</td><td>Fill in invalid details in the registration form</td><td>Invalid Profile Data including name, IC No, email, contact, gender, doctor registration number and address.</td><td>N/A</td><td>-</td><td>N/A</td><td>-</td></tr>
<tr><td>4</td><td>Click on the “Complete” button.</td><td>-</td><td>Related error messages indicating invalid data input will be displayed.</td><td>Same as Expected</td><td>Pass</td><td>-</td></tr>
<tr><td colspan="7">Post-Conditions: The user is unable to register a new account.</td></tr>
</table>

Table 5.4.1.5: Test Case for Forgot Password

Test Case #: T005			**Test Case Name:** To test the forgot password function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Login module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows users to request for password reset in the system.						
Pre-conditions: The user has registered an account in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the login page.	-	The login page will be displayed.	Same as Expected	Pass	-
2	Click on the “Forgot Password” button.	-	The forgot password page will be displayed.	Same as Expected	Pass	-
3	Fill in the registered email address to perform password recovery.	Valid email address	N/A	-	N/A	-
4	Click on the “Send Recovery Email” button.	-	A toast message indicating the completion of email sending.	Same as Expected	Pass	-
5	Open the email received.	-	The email contained an attached link for password recovery.	Same as Expected	Pass	-

6	Click on the link.	-	The page for password recovery will be displayed.	Same as Expected	Pass	-
7	Enter a new password.	Valid Password	N/A	Same as Expected	N/A	-
8	Click on the “Reset Password” button.	-	A message indicating the completion of a password reset will be displayed.	Same as Expected	Pass	-
Post-Conditions: The user password is reset and updated.						

Table 5.4.1.6: Test Case for Logout

Test Case #: T006			**Test Case Name:** To test the logout function			
System: Diabetes Monitoring System (Web/Mobile)			**Subsystem:** Login module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows users to log in when logout is initiated.						
Pre-conditions: The user has logged into the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Click on the “Sign Out” button.	-	The user is redirected to the login page.	Same as Expected	Pass	-
Post-Conditions: The user is logged out of the system.						

Table 5.4.1.7: Test Case for View and Edit Profile (Valid)

Test Case #: T007			**Test Case Name:** To test the view and edit profile function			
System: Diabetes Monitoring System (Web/Mobile)			**Subsystem:** Login module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify the system correctly allows users to view and edit their profile information.						
Pre-conditions: The user owns an account in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the profile page.	-	The profile page will be displayed with accurate current user information.	Same as Expected	Pass	-
2	Attempt to edit the profile information without activating the edit mode.	-	The text fields are not editable.	Same as Expected	Pass	-
3	Click on the “Edit” button.	-	Edit mode will be activated, and all fields except the email address will become editable.	Same as Expected	Pass	-
4	Edit the field with the necessary data.	Valid Profile Data including name, IC No, email, contact, gender, doctor registration number and	N/A	-	N/A	-

		address.				
5	Click on the “Update Profile” button.	-	A confirmation message on the update action will be displayed.	Same as Expected	Pass	-
6	Click “Yes” to continue the action.	-	A completion message will be displayed.	Same as Expected	Pass	-
7	Click “OK” to proceed.	-	The updated information will be reflected on the profile page.	Same as Expected	Pass	-
Post-Conditions: The profile information for the current user is updated.						

Table 5.4.1.8: Test Case for View and Edit Profile (Invalid)

Test Case #: T008			**Test Case Name:** To test the view and edit profile function			
System: Diabetes Monitoring System (Web/Mobile)			**Subsystem:** Login module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify the system correctly denies users to edit their profile information when invalid information entered.						
Pre-conditions: The user owns an account in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the profile page.	-	The profile page will be displayed with accurate current user information.	Same as Expected	Pass	-
2	Attempt to edit the profile information without activating the edit mode.	-	The text fields are not editable.	Same as Expected	Pass	-
3	Click on the "Edit" button.	-	Edit mode will be activated, and all fields except the email address will become editable.	Same as Expected	Pass	-
4	Edit the field with invalid data.	Invalid Profile Data including name, IC No, email, contact, gender, doctor registration number and address.	N/A	-	N/A	-

5	Click on the “Update Profile” button.	-	Related error messages indicating invalid data input will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

5.4.2 Doctor Module

Table 5.4.2.1: Test Case for Add Doctor (Valid)

Test Case #: T009			**Test Case Name:** To test the add doctor function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Doctor module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to add a new doctor into the system.						
Pre-conditions: The user has logged into the system as a doctor. The new doctor has not been registered in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the doctor's main page.	-	The doctor page will be displayed.	Same as Expected	Pass	-
2	Click on the “Add” button.	-	The add doctor page will be displayed.	Same as Expected	Pass	-
3	Fill in all necessary details to create a new doctor.	Valid Doctor Profile Data including name, IC No, contact, doctor registration number and address.	N/A	-	N/A	-
4	Click on the “Save & Next” button to proceed.	-	The current profile information will be temporarily stored and a	Same as Expected	Pass	-

			page to upload a profile image will be displayed.			
5	Optionally upload a profile image for the particular doctor.	-	The image will be displayed if uploaded.	Same as Expected	Pass	-
6	Click on the “Add New Doctor” button.	-	A confirmation message on the add action will be displayed.	Same as Expected	Pass	-
7	Click “Yes” to continue the action.	-	A completion message will be displayed.	Same as Expected	Pass	-
Post-Conditions: The new doctor information is added to the system accurately.						

Table 5.4.2.2: Test Case for Add Doctor (Invalid)

Test Case #: T010			**Test Case Name:** To test the add doctor function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Doctor module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly denies the users to add a new doctor into the system when invalid data entered.						
Pre-conditions: The user has logged into the system as a doctor.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the doctor's main page.	-	The doctor page will be displayed.	Same as Expected	Pass	-
2	Click on the “Add” button.	-	The add doctor page will be displayed.	Same as Expected	Pass	-
3	Fill in all invalid data details to create a new doctor.	Invalid Doctor Profile Data including name, IC No, contact, doctor registration number and address. / Duplicated Doctor Email / Leave Empty	N/A	-	N/A	-
4	Click on the “Save & Next” button to proceed.	-	Related error messages indicating invalid data input will be displayed.	Same as Expected	Pass	-

Post-Conditions:

Table 5.4.2.3: Test Case for View Doctor

Test Case #: T011			**Test Case Name:** To test the view doctor function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Doctor module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to view all doctors in the system.						
Pre-conditions: There is at least one existing doctor record in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the doctor's main page.	-	The doctor page will be displayed with all the existing doctors' information within the system in a grid view.	Same as Expected	Pass	-
2	Click on the “List View” icon.	-	The doctors’ information will be displayed in a list view.	Same as Expected	Pass	-
3	Click on the “Grid View” icon.	-	The doctors’ information will be displayed in a grid view.	Same as Expected	Pass	-
4	Click on one record from the doctor lists.	-	The details of the doctor will be displayed.	Same as Expected	Pass	-

5	Click on the "Personal Information" tab within the details page.	-	The information about this particular doctor will be displayed.	Same as Expected	Pass	-
6	Click on the "Appointment" tab within the details page.	-	The appointments belonging to this doctor will be displayed.	Same as Expected	Pass	-
7	Click on the "Patient List" tab within the details page.	-	The list of patients under this particular doctor will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

Table 5.4.2.4: Test Case for Edit Doctor (Valid)

Test Case #: T012			**Test Case Name:** To test the edit doctor function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Doctor module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to edit doctor's information in the system.						
Pre-conditions: There is at least one existing doctor record in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the doctor's main page.	-	The doctor page will be displayed.	Same as Expected	Pass	-
2	Click on the record from the doctor lists to edit.	-	The details of the doctor will be displayed.	Same as Expected	Pass	-
3	Attempt to edit the profile information without activating the edit mode.	-	The text fields are not editable.	Same as Expected	Pass	-
4	Click on the "Edit" button.	-	Edit mode will be activated, and all fields except the email address will become editable.	Same as Expected	Pass	-
5	Edit the field with the necessary information.	Valid Doctor Profile Data	N/A	-	N/A	-

6	Click on the “Update Profile” button.	-	A confirmation message on the update action will be displayed.	Same as Expected	Pass	-
7	Click “Yes” to continue the action.	-	A completion message will be displayed.	Same as Expected	Pass	-
8	Click “OK” to proceed.	-	The updated information will be reflected on the profile page.	Same as Expected	Pass	-
Post-Conditions: The edited information for the doctor is updated accordingly.						

Table 5.4.2.5: Test Case for Edit Doctor (Invalid)

Test Case #: T013			**Test Case Name:** To test the edit doctor function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Doctor module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly denies the users to edit doctor's information in the system when invalid data entered.						
Pre-conditions: There is at least one existing doctor record in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the doctor's main page.	-	The doctor page will be displayed.	Same as Expected	Pass	-
2	Click on the record from the doctor lists to edit.	-	The details of the doctor will be displayed.	Same as Expected	Pass	-
3	Attempt to edit the profile information without activating the edit mode.	-	The text fields are not editable.	Same as Expected	Pass	-
4	Click on the "Edit" button.	-	Edit mode will be activated, and all fields except the email address will become editable.	Same as Expected	Pass	-
5	Edit the field with the necessary information.	Invalid Doctor Profile Data	N/A	Same as Expected	N/A	-

6	Click on the "Update Profile" button.	-	Related error messages indicating invalid data input will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

Table 5.4.2.6: Test Case for Deactivate Doctor

<table>
<tr><td colspan="3">Test Case #: T014</td><td colspan="4">Test Case Name: To test the deactivate doctor function</td></tr>
<tr><td colspan="3">System: Diabetes Monitoring System (Web)</td><td colspan="4">Subsystem: Doctor module</td></tr>
<tr><td colspan="3">Designed By: Ong Jia Hui</td><td colspan="4">Design Date: 21/12/2023</td></tr>
<tr><td colspan="3">Executed By: Ong Jia Hui</td><td colspan="4">Execution Date: 21/12/2023</td></tr>
<tr><td colspan="7">Short Description: To verify that the system correctly allows the users to deactivate a doctor in the system.</td></tr>
<tr><td colspan="7">Pre-conditions:
The doctor to be deactivated is currently in active state.</td></tr>
<tr><th>Step</th><th>Action / Step</th><th>Test Data</th><th>Expected System Response</th><th>Actual System Response</th><th>Pass/ Fail</th><th>Comments</th></tr>
<tr><td>1</td><td>Navigate to the doctor's main page.</td><td>-</td><td>The doctor page will be displayed.</td><td>Same as Expected</td><td>Pass</td><td>-</td></tr>
<tr><td>2</td><td>Click on the record from the doctor lists to edit.</td><td>-</td><td>The details of the doctor will be displayed.</td><td>Same as Expected</td><td>Pass</td><td>-</td></tr>
<tr><td>3</td><td>Toggle the switch to change the doctor's current status.</td><td>-</td><td>The doctor’s status will be updated to inactive.</td><td>Same as Expected</td><td>Pass</td><td>-</td></tr>
<tr><td colspan="7">Post-Conditions: The status of the particular doctor is updated to inactive.</td></tr>
</table>

Table 5.4.2.7: Test Case for View Doctor Information by Patients

Test Case #: T015			**Test Case Name:** To test the view doctor function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Doctor module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows patients to access the information of their doctor in charge.						
Pre-conditions: The user has logged into the system under the role of a patient. The user owns an account in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the “My Doctor” page.	-	The information of the doctor in charge will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

5.4.3 Patient Module

Table 5.4.3.1: Test Case for Add Patient (Valid)

Test Case #: T016			**Test Case Name:** To test the add patient function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Patient module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to add a new patient into the system.						
Pre-conditions: The user has logged into the system as a doctor.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the patient's main page.	-	The patient page will be displayed.	Same as Expected	Pass	-
2	Click on the "Add" button.	-	The add patient page will be displayed.	Same as Expected	Pass	-
3	Fill in all necessary details to create a new patient.	Valid Patient Profile Data including name, IC No, email, contact, gender, diabetes type and address.	N/A	-	N/A	-
4	Click on the "Save & Next" button to proceed.	-	The current profile information will be temporarily stored and a	Same as Expected	Pass	-

			page to assign a doctor will be displayed.			
5	Select a doctor for the particular patient.	-	The selected doctor ID will be displayed in the text field below.	Same as Expected	Pass	-
6	Optionally upload a profile image for the particular patient.	Patient Profile Image	The image will be displayed if uploaded.	Same as Expected	Pass	-
7	Click on the “Add New Patient” button.	-	A confirmation message on the add action will be displayed.	Same as Expected	Pass	-
8	Click “Yes” to continue the action.	-	A completion message will be displayed.	Same as Expected	Pass	-
Post-Conditions: The new patient information is added to the system accurately.						

Table 5.4.3.2: Test Case for Add Patient (Invalid)

Test Case #: T017			**Test Case Name:** To test the add patient function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Patient module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly denies the users to add a new patient into the system when invalid data is entered.						
Pre-conditions: The user has logged into the system as a doctor.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the patient's main page.	-	The patient page will be displayed.	Same as Expected	Pass	-
2	Click on the "Add" button.	-	The add patient page will be displayed.	Same as Expected	Pass	-
3	Fill in invalid data for all fields.	Invalid Patient Profile Data including name, IC No, email, contact, gender, diabetes type and address.	N/A	Same as Expected	N/A	-
4	Click on the "Save & Next" button to proceed.	-	Related error messages indicating invalid data input will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

Table 5.4.3.3: Test Case for View Patient

Test Case #: T018			**Test Case Name:** To test the view patient function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Doctor module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to view all patients in the system.						
Pre-conditions: The user has logged into the system as a doctor.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the patient's main page.	-	The patient page will be displayed with all the existing patients' information within the system in a grid view.	Same as Expected	Pass	-
2	Click on the "List View" icon.	-	The patient's information will be displayed in a list view.	Same as Expected	Pass	-
3	Click on the "Grid View" icon.	-	The patient's information will be displayed in a grid view.	Same as Expected	Pass	-
4	Click on one record from the patient lists.	-	The details of the patient will be displayed.	Same as Expected	Pass	-

5	Click on the “Personal Information” tab within the details page.	-	The information about this particular patient will be displayed.	Same as Expected	Pass	-
6	Click on the “Medical History” tab within the details page.	-	The medical history belonging to this patient will be displayed.	Same as Expected	Pass	-
7	Click on the “Blood Glucose Reading” tab within the details page.	-	The blood glucose recorded by this patient will be displayed in a graph view.	Same as Expected	Pass	-
Post-Conditions:						

Table 5.4.3.4: Test Case for Edit Patient (Valid)

Test Case #: T019			**Test Case Name:** To test the edit patient function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Patient module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to edit patient's information in the system.						
Pre-conditions: There is at least one existing patient record in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the patient's main page.	-	The patient page will be displayed.	Same as Expected	Pass	-
2	Click on the record from the patient lists to edit.	-	The details of the patient will be displayed.	Same as Expected	Pass	-
3	Attempt to edit the profile information without activating the edit mode.	-	The text fields are not editable.	Same as Expected	Pass	-
4	Click on the "Edit" button.	-	Edit mode will be activated, and all fields except the email address will become editable.	Same as Expected	Pass	-

5	Edit the field with the necessary information.	Valid Patient Profile Data including name, IC No, email, contact, gender, diabetes type and address.	N/A	-	N/A	-
6	Click on the “Update Profile” button.	-	A confirmation message on the update action will be displayed.	Same as Expected	Pass	-
7	Click “Yes” to continue the action.	-	A completion message will be displayed.	Same as Expected	Pass	-
8	Click “OK” to notify the completion.	-	The updated information will be reflected on the profile page.	Same as Expected		-
Post-Conditions: The profile information for the patient editing is updated.						

Table 5.4.3.5: Test Case for Edit Patient (Invalid)

Test Case #: T020			**Test Case Name:** To test the edit patient function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Patient module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly denies the users to edit patient's information in the system when invalid data entered.						
Pre-conditions: There is at least one existing patient record in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the patient's main page.	-	The patient page will be displayed.	Same as Expected	Pass	-
2	Click on the record from the patient lists to edit.	-	The details of the patient will be displayed.	Same as Expected	Pass	-
3	Attempt to edit the profile information without activating the edit mode.	-	The text fields are not editable.	Same as Expected	Pass	-
4	Click on the "Edit" button.	-	Edit mode will be activated, and all fields except the email address will become editable.	Same as Expected	Pass	-

5	Edit the field with the invalid data.	Invalid Patient Profile Data including name, IC No, email, contact, gender, diabetes type and address.	N/A	-	N/A	-
6	Click on the “Update Profile” button.	-	Related error messages indicating invalid data input will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

Table 5.4.3.6: Test Case for Deactivate Patient

Test Case #: T021			**Test Case Name:** To test the deactivate patient function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Patient module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to deactivate a patient in the system.						
Pre-conditions: There is at least one existing patient record in the system. The patient to be deactivated is currently in an active state.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the patient's main page.	-	The patient page will be displayed.	Same as Expected	Pass	-
2	Click on the record from the patient lists to edit.	-	The details of the patient will be displayed.	Same as Expected	Pass	-
3	Toggle the switch to change the patient’s current status.	-	The patient’s status will be updated to inactive.	Same as Expected	Pass	-
Post-Conditions: The status of the patient is updated to inactive.						

5.4.4 Appointment Module

Table 5.4.4.1: Test Case for Add Appointment (Valid)

Test Case #: T022			**Test Case Name:** To test the add appointment function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Appointment module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to add a new appointment into the system.						
Pre-conditions: There is at least one existing patient record in the system. There is at least one existing doctor record in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the appointment's main page.	-	The appointment page will be displayed.	Same as Expected	Pass	-
2	Click on the “Add” button.	-	The add appointment page will be displayed.	Same as Expected	Pass	-
3	Fill in all necessary details to create a new appointment.	Valid Appointment Data including date, time and remark	N/A	-	N/A	-
4	Click on the “Add New Appointment” button.	-	A confirmation message on the add action will be displayed.	Same as Expected	Pass	-

5	Click “Yes” to continue the action.	-	A completion message will be displayed.	Same as Expected	Pass	-
Post-Conditions: The new appointment information is added to the system accurately.						

Table 5.4.4.2: Test Case for Add Appointment (Invalid)

Test Case #: T023			**Test Case Name:** To test the add appointment function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Appointment module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly denies the users to add a new appointment into the system when invalid data entered.						
Pre-conditions: There is at least one existing patient record in the system. There is at least one existing doctor record in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the appointment's main page.	-	The appointment page will be displayed.	Same as Expected	Pass	-
2	Click on the "Add" button.	-	The add appointment page will be displayed.	Same as Expected	Pass	-
3	Fill in all necessary details to create a new appointment.	Invalid Appointment Data including date, time and remark / Leave empty	N/A	-	N/A	-
4	Click on the "Add New Appointment" button.	-	Related error messages indicating invalid data input will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

Table 5.4.4.3: Test Case for View Appointment

Test Case #: T024			**Test Case Name:** To test the view appointment function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Appointment module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to view all appointments in the system.						
Pre-conditions: There is at least one existing appointment record in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the appointment's main page.	-	The appointment page will be displayed with all the existing appointments' information within the system in a list view.	Same as Expected	Pass	-
2	Click on the “Calendar View” icon.	-	The appointments’ information will be displayed in a calendar view.	Same as Expected	Pass	-
3	Click on the “List View” icon.	-	The doctors’ information will be displayed in a list view.	Same as Expected	Pass	-
4	Click on one record from the appointment lists.	-	The details of the appointment will be	Same as Expected	Pass	-

			displayed with the necessary follow-up and medication field below.			
Post-Conditions:						

Table 5.4.4.4: Test Case for Approve Appointment

Test Case #: T025			**Test Case Name:** To test the approved appointment function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Appointment module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to approve pending appointments in the system.						
Pre-conditions: The appointment is currently in the status of pending.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the appointment's main page.	-	The appointment page will be displayed.	Same as Expected	Pass	-
2	Click on the “Approve” icon for an appointment to approve.	-	A confirmation message on the approved action will be displayed.	Same as Expected	Pass	-
3	Click “Yes” to continue the action.	-	The respective appointment has been updated to “Approved” status.	Same as Expected	Pass	-
Post-Conditions: The status of the appointment is updated to the status of approved.						

Table 5.4.4.5: Test Case for Reject Appointment

Test Case #: T026			**Test Case Name:** To test the reject appointment function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Appointment module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to reject pending appointments in the system.						
Pre-conditions: The appointment is currently in the status of pending.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the appointment's main page.	-	The appointment page will be displayed.	Same as Expected	Pass	-
2	Click on the “Reject” icon for an appointment to reject.	-	A confirmation message on the approved action will be displayed.	Same as Expected	Pass	-
3	Click “Yes” to continue the action.	-	The respective appointment has been updated to “Rejected” status.	Same as Expected	Pass	-
Post-Conditions: The status of the appointment is updated to the status of rejected.						

Table 5.4.4.6: Test Case for Update Appointment (Valid)

Test Case #: T027			**Test Case Name:** To test the view and update appointment function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Appointment module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to edit appointment's information in the system.						
Pre-conditions: The appointment is currently in the status of approved.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the appointment's main page.	-	The appointment page will be displayed.	Same as Expected	Pass	-
2	Click on the record from the appointment lists.	-	The details of the appointment will be displayed together with the patient and doctor information.	Same as Expected	Pass	-
3	Enter necessary follow-up information.	Valid Follow-Up Information including date, time, desc, remark, weight, and blood pressure	N/A	-	N/A	-
4	Enter necessary medication information.	Valid Medication Information including	N/A	-	N/A	-

		instruction and description				
5	Click on the “Update Appointment” button.	-	A confirmation message on the update action will be displayed.	Same as Expected	Pass	-
6	Click “Yes” to continue the action.	-	A completion message will be displayed.	Same as Expected	Pass	-
7	Click “OK” to proceed.	-	The updated information will be reflected on the profile page.	Same as Expected	Pass	-
Post-Conditions: The appointment information is updated along with the follow-up and medication details.						

Table 5.4.4.7: Test Case for Update Appointment (Invalid)

Test Case #: T028			**Test Case Name:** To test the view and update appointment function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Appointment module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly denies the users to edit appointment's information in the system when invalid data entered						
Pre-conditions: The appointment is currently in the status of pending.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the appointment's main page.	-	The appointment page will be displayed.	Same as Expected	Pass	-
2	Click on the record from the appointment lists.	-	The details of the appointment will be displayed together with the patient and doctor information.	Same as Expected	Pass	-
3	Enter invalid follow-up data.	Invalid Follow-Up Information including date, time, desc, remark, weight, and blood pressure / Leave Empty	N/A	-	N/A	-

4	Enter invalid medication data.	Invalid Medication Information including instruction and description / Leave Empty	N/A	-	N/A	-
5	Click on the “Update Appointment” button.	-	Related error messages indicating invalid data input will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

Table 5.4.4.8: Test Case for Add Medicine (Valid)

Test Case #: T029			**Test Case Name:** To test the add medicine function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Appointment module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the doctor to add a new medicine into the system.						
Pre-conditions: The doctor has logged into the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the medicine's main page.	-	The medicine page will be displayed.	Same as Expected	Pass	-
2	Click on the "Add" button.	-	The add medicine page will be displayed.	Same as Expected	Pass	-
3	Fill in all necessary details to create a new medicine.	Valid Medicine name, production company and precaution.	N/A	-	N/A	-
4	Click on the "Save" button.	-	A confirmation message on the add action will be displayed.	Same as Expected	Pass	-
5	Click "Yes" to continue the action.	-	A completion toast message will be displayed.	Same as Expected	Pass	-

Post-Conditions: The new medicine information is added to the system accurately.

Table 5.4.4.9: Test Case for Add Medicine (Invalid)

<table>
<tr><td colspan="3">Test Case #: T030</td><td colspan="4">Test Case Name: To test the add medicine function</td></tr>
<tr><td colspan="3">System: Diabetes Monitoring System (Web)</td><td colspan="4">Subsystem: Appointment module</td></tr>
<tr><td colspan="3">Designed By: Ong Jia Hui</td><td colspan="4">Design Date: 21/12/2023</td></tr>
<tr><td colspan="3">Executed By: Ong Jia Hui</td><td colspan="4">Execution Date: 21/12/2023</td></tr>
<tr><td colspan="7">Short Description: To verify that the system correctly allows the doctor to add a new medicine into the system.</td></tr>
<tr><td colspan="7">Pre-conditions:
The doctor has logged into the system.</td></tr>
<tr><th>Step</th><th>Action / Step</th><th>Test Data</th><th>Expected System Response</th><th>Actual System Response</th><th>Pass/ Fail</th><th>Comments</th></tr>
<tr><td>1</td><td>Navigate to the medicine’s main page.</td><td>-</td><td>The medicine page will be displayed.</td><td>Same as Expected</td><td>Pass</td><td>-</td></tr>
<tr><td>2</td><td>Click on the “Add” button.</td><td>-</td><td>The add medicine page will be displayed.</td><td>Same as Expected</td><td>Pass</td><td>-</td></tr>
<tr><td>3</td><td>Fill in all necessary details to create a new medicine.</td><td>Invalid Medicine name, production company and precaution.</td><td>N/A</td><td>-</td><td>N/A</td><td>-</td></tr>
<tr><td>4</td><td>Click on the “Save” button.</td><td>-</td><td>Related error messages indicating invalid data input will be displayed.</td><td>Same as Expected</td><td>Pass</td><td>-</td></tr>
<tr><td colspan="7">Post-Conditions:</td></tr>
</table>

Table 5.4.4.10: Test Case for Add Appointment (Valid)

Test Case #: T031			**Test Case Name:** To test the add appointment function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Appointment module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the patient to add a new appointment into the system.						
Pre-conditions: The patient has logged into the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the appointment's main page.	-	The appointment page will be displayed.	Same as Expected	Pass	-
2	Click on the “Add” button.	-	The add appointment page will be displayed.	Same as Expected	Pass	-
3	Select and fill in all necessary details to create a new appointment.	Valid Appointment Data including date, time and remark	N/A	-	N/A	-
4	Click on the “Add New Appointment” button.	-	A confirmation message on the add action will be displayed.	Same as Expected	Pass	-
5	Click “Yes” to continue the action.	-	A completion toast message will be displayed.	Same as Expected	Pass	-

Post-Conditions:	The new appointment information is added to the system accurately.

Table 5.4.4.11: Test Case for Add Appointment (Invalid)

Test Case #: T032			**Test Case Name:** To test the add appointment function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Appointment module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly denies the patient to add a new appointment into the system when invalid data entered.						
Pre-conditions: The patient has logged into the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the appointment's main page.	-	The appointment page will be displayed.	Same as Expected	Pass	-
2	Click on the “Add” button.	-	The add appointment page will be displayed.	Same as Expected	Pass	-
3	Select and fill in invalid data details to create a new appointment.	Invalid Appointment Data including date, time and remark	N/A	-	N/A	-
4	Click on the “Add New Appointment” button.	-	Related error messages indicating invalid data input will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

Table 5.4.4.12: Test Case for View Appointment

Test Case #: T033			**Test Case Name:** To test the view appointment function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Appointment module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the patient to view all the appointments that belong to them in the system.						
Pre-conditions: The patient has logged into the system. The patient has made at least one appointment record.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the appointment's main page.	-	The appointment page will be displayed with all the existing appointment information within the system under a calendar view.	Same as Expected	Pass	-
2	Click on the date with an appointment(s) made.	-	A column of appointment information will be displayed.	Same as Expected	Pass	-
3	Click on the "List View" icon.	-	The appointment's information will be displayed in a listview.	Same as Expected	Pass	-

4	Click on the “Calendar View” icon.	-	The appointment’s information will be displayed in a calendar view.	Same as Expected	Pass	-
5	Click on the record from the appointment lists.	-	The details of the appointment will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

Table 5.4.4.13: Test Case for Edit Appointment (Valid)

Test Case #: T034			**Test Case Name:** To test the edit appointment function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Appointment module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to edit appointment's information in the system.						
Pre-conditions: The patient has logged into the system. The patient has made at least one appointment record. The desired appointment record is not under the status completed.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the appointment's main page.	-	The appointment page will be displayed.	Same as Expected	Pass	-
2	Click on the record from the appointment lists to edit.	-	The details of the appointment will be displayed.	Same as Expected	Pass	-
3	Attempt to edit the appointment information without activating the edit mode.	-	The text fields are not editable.	Same as Expected	Pass	-
4	Click on the "Edit" button.	-	Edit mode will be activated, and all fields will become editable.	Same as Expected	Pass	-

5	Edit the field with the necessary information.	Valid Appointment Data including date, time and remark	N/A	-	N/A	-
6	Click on the “Update Profile” button.	-	A confirmation message on the update action will be displayed.	Same as Expected	Pass	-
7	Click “Yes” to continue the action.	-	A completion toast message will be displayed.	Same as Expected	Pass	-
Post-Conditions: The information of the particular appointment is updated in the system accurately.						

Table 5.4.4.14: Test Case for Edit Appointment (Invalid)

Test Case #: T035			**Test Case Name:** To test the edit appointment function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Appointment module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description To verify that the system correctly denies the users to edit appointment's information in the system when invalid data entered.						
Pre-conditions: The patient has logged into the system. The patient has made at least one appointment record. The desired appointment record is not under the status completed.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the appointment's main page.	-	The appointment page will be displayed.	Same as Expected	Pass	-
2	Click on the record from the appointment lists to edit.	-	The details of the appointment will be displayed.	Same as Expected	Pass	-
3	Attempt to edit the appointment information without activating the edit mode.	-	The text fields are not editable.	Same as Expected	Pass	-
4	Click on the "Edit" button.	-	Edit mode will be activated, and all fields will become editable.	Same as Expected	Pass	-

5	Edit the field with the invalid data.	Invalid Appointment Data including date, time and remark / Leave Empty	N/A	-	N/A	-
6	Click on the “Update Profile” button.	-	Related error messages indicating invalid data input will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

Table 5.4.4.15: Test Case for View Medical History

Test Case #: T036			**Test Case Name:** To test the view medical history function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Appointment module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to view the medical history information in the system.						
Pre-conditions: There is at least one existing medical history record in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the medical history main page.	-	The medical history page will be displayed.	Same as Expected	Pass	-
2	Click on the record from the medical history lists.	-	The details of the medical history will be displayed together with the follow-up and medication information.	Same as Expected	Pass	-
Post-Conditions:						

Table 5.4.4.16: Test Case for View Feedback

<table>
<tr><td colspan="3">Test Case #: T037</td><td colspan="4">Test Case Name: To test the view feedback function</td></tr>
<tr><td colspan="3">System: Diabetes Monitoring System (Mobile)</td><td colspan="4">Subsystem: Appointment module</td></tr>
<tr><td colspan="3">Designed By: Ong Jia Hui</td><td colspan="4">Design Date: 21/12/2023</td></tr>
<tr><td colspan="3">Executed By: Ong Jia Hui</td><td colspan="4">Execution Date: 21/12/2023</td></tr>
<tr><td colspan="7">Short Description: To verify that the system correctly allows the patient to view the feedback information in the system.</td></tr>
<tr><td colspan="7">Pre-conditions:
There is at least one feedback record in the system.</td></tr>
<tr><th>Step</th><th>Action / Step</th><th>Test Data</th><th>Expected System Response</th><th>Actual System Response</th><th>Pass/ Fail</th><th>Comments</th></tr>
<tr><td>1</td><td>Navigate to the feedback main page.</td><td>-</td><td>The feedback page will be displayed.</td><td>Same as Expected</td><td>Pass</td><td>-</td></tr>
<tr><td>2</td><td>Click on the record from the feedback lists to view.</td><td>-</td><td>The details of the feedback will be displayed.</td><td>Same as Expected</td><td>Pass</td><td>-</td></tr>
<tr><td colspan="7">Post-Conditions:</td></tr>
</table>

5.4.5 Blood Glucose Module

Table 5.4.5.1: Test Case for Add Blood Glucose (Valid)

Test Case #: T038			**Test Case Name:** To test the add glucose record function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Blood Glucose module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to add a new blood glucose record into the system.						
Pre-conditions: The patient has logged into the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the blood glucose main page.	-	The blood glucose page will be displayed.	Same as Expected	Pass	-
2	Click on the “Add” button.	-	A blood glucose record type will pop up.	Same as Expected	Pass	-
2	Select the type of blood glucose recording.	-	The add blood glucose page will be displayed.	Same as Expected	Pass	-
3	Fill in all necessary details to create a new blood glucose record.	Valid Blood Glucose Data including date, time, reading value, unit, blood pressure, remark and weight	N/A		N/A	-

4	Click on the “Done” button.	-	A confirmation message on the add action will be displayed.	Same as Expected	Pass	-
5	Click “Yes” to continue the action.	-	A completion toast message will be displayed.	Same as Expected	Pass	-
Post-Conditions: The new blood glucose reading information is added to the system accurately.						

Table 5.4.5.2: Test Case for Add Blood Glucose (Invalid)

Test Case #: T039			**Test Case Name:** To test the add glucose record function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Blood Glucose module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly denies the users to add a new blood glucose record into the system when invalid data entered.						
Pre-conditions: The patient has logged into the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the blood glucose main page.	-	The blood glucose page will be displayed.	Same as Expected	Pass	-
2	Click on the “Add” button.	-	A blood glucose record type will pop up.	Same as Expected	Pass	-
2	Select the type of blood glucose recording.	-	The add blood glucose page will be displayed.	Same as Expected	Pass	-
3	Fill in invalid blood glucose data to create a new blood glucose record.	Invalid Blood Glucose Data including date, time, reading value, unit, blood pressure, remark and weight	N/A	-	N/A	-

4	Click on the "Done" button.	-	Related error messages indicating invalid data input will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

Table 5.4.5.3: Test Case for View Blood Glucose

Test Case #: T040			**Test Case Name:** TTo test the view glucose record function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Blood Glucose module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to view the blood glucose record(s) in the system.						
Pre-conditions: The patient has logged into the system. There is at least one existing blood glucose record by this patient in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the blood glucose main page.	-	The blood glucose page with today's records will be displayed.	Same as Expected	Pass	-
2	Click on the date button and modify the date.	-	The blood glucose record under the selected date will be displayed.	Same as Expected	Pass	-
3	Select the sort button and modify the glucose value display unit.	-	The records will be displayed based on the selection.	Same as Expected	Pass	--
4	Click on the record to view details.	-	The details of the particular record are displayed.	Same as Expected	Pass	-

Post-Conditions:

Table 5.4.5.4: Test Case for Edit Blood Glucose (Valid)

Test Case #: T041			**Test Case Name:** To test the edit glucose record function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Blood Glucose module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to edit the blood glucose record(s) in the system.						
Pre-conditions: The patient has logged into the system. There is at least one existing blood glucose record by this patient in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the glucose record main page.	-	The glucose record page will be displayed.	Same as Expected	Pass	-
2	Click on the record from the blood glucose lists to edit.	-	The details of the glucose record will be displayed.	Same as Expected	Pass	-
3	Attempt to edit the information without activating the edit mode.	-	The text fields are not editable.	Same as Expected	Pass	-
4	Click on the “Edit” button.	-	Edit mode will be activated, and all fields will become editable.	Same as Expected	Pass	-
5	Edit the field with the necessary information.	Valid Blood Glucose Data including date, time, reading value, unit, blood	N/A	-	N/A	-

		pressure, remark and weight				
6	Click on the “Done” button.	-	A confirmation message on the update action will be displayed.	Same as Expected	Pass	-
7	Click “Yes” to continue the action.	-	A completion toast message will be displayed.	Same as Expected	Pass	-
Post-Conditions: The edited blood glucose record is updated successfully.						

Table 5.4.5.5: Test Case for Edit Blood Glucose (Invalid)

Test Case #: T042			**Test Case Name:** TTo test the edit glucose record function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Blood Glucose module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly denies the users to edit glucose records in the system when invalid data entered.						
Pre-conditions: The patient has logged into the system. The patient has made at least one glucose record.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the glucose record main page.	-	The glucose record page will be displayed.	Same as Expected	Pass	-
2	Click on the record from the blood glucose lists to edit.	-	The details of the glucose record will be displayed.	Same as Expected	Pass	-
3	Attempt to edit the information without activating the edit mode.	-	The text fields are not editable.	Same as Expected	Pass	-
4	Click on the "Edit" button.	-	Edit mode will be activated, and all fields will become editable.	Same as Expected	Pass	-
5	Edit the field with the invalid data.	Invalid Blood Glucose Data including date, time, reading value, unit, blood	N/A	-	N/A	-

		pressure, remark and weight				
6	Click on the “Done” button.	-	Related error messages indicating invalid data input will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

Table 5.4.5.6: Test Case for View Blood Glucose Statistics

Test Case #: T043			**Test Case Name:** To test the blood glucose statistics generation function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Blood Glucose module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly generates visualised graphs for the blood glucose record in the system.						
Pre-conditions: The patient has logged into the system. There is at least one existing blood glucose record by this patient in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the statistics main page.	-	The graph with the default time duration will be displayed.	Same as Expected	Pass	-
2	Selected the time range for the report.	-	The graph based on the time duration selected will be displayed accordingly in a graph.	Same as Expected	Pass	-
Post-Conditions:						

Table 5.4.5.7: Test Case for Add Advice (Valid)

Test Case #: T044			**Test Case Name:** To test the add advice function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Blood Glucose module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to add new advice into the system.						
Pre-conditions: The doctor has logged into the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the advice main page.	-	The advice page will be displayed.	Same as Expected	Pass	-
2	Click on the “Add” button.	-	The add advice pop-up page will be displayed.	Same as Expected	Pass	-
3	Fill in all necessary details to create a new advice.	Valid Advice Data including description, more description and title.	N/A	-	N/A	-
4	Click on the “Add” button.	-	A confirmation message on the add action will be displayed.	Same as Expected	Pass	-
5	Click “Yes” to continue the action.	-	A completion message will be displayed.	Same as Expected	Pass	-

6	Click on the "Deactivate" switch for the respective advice to deactivate.	-	The advice will be deactivated.	Same as Expected	Pass	-
Post-Conditions: The new advice information is added to the system accurately.						

Table 5.4.5.8: Test Case for Add Advice (Invalid)

Test Case #: T045			**Test Case Name:** To test the add advice function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Blood Glucose module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly denies the users to add new advice into the system when invalid data entered						
Pre-conditions: The doctor has logged into the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the advice main page.	-	The advice page will be displayed.	Same as Expected	Pass	-
2	Click on the “Add” button.	-	The added advice pop-up page will be displayed.	Same as Expected	Pass	-
3	Fill in invalid data to create a new advice.	Invalid Advice Data including description, more description and title.	N/A	Same as Expected	N/A	-
4	Click on the “Add” button.	-	Related error messages indicating invalid data input will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

Table 5.4.5.9: Test Case for Manage Label

Test Case #: T046			**Test Case Name:** To test the label function			
System: Diabetes Monitoring System (Mobile)			**Subsystem:** Blood Glucose module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly allows the users to manage the labels into the system.						
Pre-conditions: The patient has logged into the system. There is at least one existing label by this patient in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the label main page.	-	The label page will be displayed with all label information.	Same as Expected	Pass	-
2	Click on the “Add” button.	-	The bottom pop-up for adding a new label will be displayed.	Same as Expected	Pass	-
3	Fill in the new label.	Valid Label Data including label description	N/A	-	N/A	-
4	Click on the “Add” button.	-	A completion toast message will be displayed.	Same as Expected	Pass	-
5	Click on the “Deactivate” button	-	A confirmation message	Same as Expected	Pass	-

	for the respective label to deactivate.		on the deactivate action will be displayed.			
6	Click “Yes” to continue the action.	-	The labels list will be displayed with the label status updated.	Same as Expected	Pass	-
7	Click on the “Activate” button for the respective label to activate.	-	A confirmation message on the activate action will be displayed.	Same as Expected	Pass	-
8	Click “Yes” to continue the action.	-	The labels list will be displayed with the label status updated.	Same as Expected	Pass	-
Post-Conditions: The new label information is added to the system accurately.						

Table 5.4.5.10: Test Case for Add Label (Valid)

Test Case #: T047			**Test Case Name:** To test the add label function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Blood Glucose module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly denies the users to add new labels into the system when invalid data entered.						
Pre-conditions: The patient has logged into the system. There is at least one existing label by this patient in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the label main page.	-	The label page will be displayed with all label information.	Same as Expected	Pass	-
2	Click on the “Add” button.	-	The bottom pop-up for adding a new label will be displayed.	Same as Expected	Pass	-
3	Fill in the new label with invalid data	Invalid Label Data including label description / Leave Empty	N/A	Same as Expected	N/A	-
4	Click on the “Add” button.	-	Related error messages indicating invalid data input will be displayed.	Same as Expected	Pass	-

Post-Conditions:

Table 5.4.5.11: Test Case for Reminder (Valid)

Test Case #: T048	**Test Case Name:** To test the reminder function
System: Diabetes Monitoring System (Web)	**Subsystem:** Blood Glucose module
Designed By: Ong Jia Hui	**Design Date:** 21/12/2023
Executed By: Ong Jia Hui	**Execution Date:** 21/12/2023
Short Description: To verify that the system correctly allows the users to manage the reminder into the system.	
Pre-conditions: The patient has logged into the system.	

Step	Action / Step	Test Data	Expected System Response	Actual System Response	Pass/ Fail	Comments
1	Navigate to the reminder main page.	-	The reminder page will be displayed with all reminder information.	Same as Expected	Pass	-
2	Click on the “Add” button.	-	The bottom pop-up for adding a new reminder will be displayed.	Same as Expected	Pass	-
3	Fill in the new reminder.	Valid Reminder Data including time and remark	N/A	-	N/A	-
4	Click on the “Add” button.	-	A completion toast message will be	Same as Expected	Pass	-

			displayed.			
5	Click on the "Deactivate" button for the respective reminder to deactivate.	-	A confirmation message on the deactivate action will be displayed.	Same as Expected	Pass	-
6	Click "Yes" to continue the action.	-	The reminder list will be displayed with the reminder status updated.	Same as Expected	Pass	-
7	Click on the "Activate" button for the respective reminder to activate.	-	A confirmation message on the activate action will be displayed.	Same as Expected	Pass	-
8	Click "Yes" to continue the action.	-	The reminder list will be displayed with the reminder status updated.	Same as Expected	Pass	-
Post-Conditions: The reminder information is updated to the system accurately.						

Table 5.4.5.12: Test Case for Reminder (Invalid)

Test Case #: T049			Test Case Name: To test the add reminder function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Blood Glucose module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly denies the users to add the reminder into the system when invalid data entered.						
Pre-conditions: The patient has logged into the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the reminder main page.	-	The reminder page will be displayed with all reminder information.	Same as Expected	Pass	-
2	Click on the “Add” button.	-	The bottom pop-up for adding a new reminder will be displayed.	Same as Expected	Pass	-
3	Fill in the new reminder with invalid data.	Invalid Reminder Data including time and remark	N/A	-	N/A	-
4	Click on the “Add” button.	-	Related error messages indicating invalid data input will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

5.4.6 Ingredient Scanning Module

Table 5.4.6.1: Test Case for Ingredient Scanning (Valid)

Test Case #: T050			**Test Case Name:** To test the ingredient scanning function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Ingredient Scanning module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly performs the ingredient scanning.						
Pre-conditions: The patient has logged into the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the ingredient scanning main page.	-	The ingredient scanning page will be displayed.	Same as Expected	Pass	-
2	Scan the ingredient list on the food packaging	Vald Ingredient List Image	N/A	-	N/A	-
3	Click on the “Scan Text” button.	-	The system will capture a maximum of 10 ingredient items and display them in a list.	Same as Expected	Pass	-
4	Edit the ingredient items when necessary.	Valid Ingredient Item Name	N/A	-	N/A	-
5	Click on the “Done” button	-	The scanning result will	Same as Expected	Pass	-

			be displayed.			
Post-Conditions:						

Table 5.4.6.2: Test Case for Ingredient Scanning (Invalid)

Test Case #: T051			**Test Case Name:** To test the ingredient scanning function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Ingredient Scanning module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly defines the image scanned is invalid in the ingredient scanning.						
Pre-conditions: The patient has logged into the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the ingredient scanning main page.	-	The ingredient scanning page will be displayed.	Same as Expected	Pass	-
2	Scan the ingredient list on the food packaging	Invalid Ingredient List image	N/A	-	N/A	-
3	Click on the “Scan Text” button.	-	Related error messages indicating invalid data input will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

5.4.7 Report Module

Table 5.4.7.1: Test Case for Report I

Test Case #: T052			**Test Case Name:** To test the report generation function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Report module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly generates the blood glucose events report that is required by the doctor.						
Pre-conditions: There is at least one existing patient record in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the report's main page.	-	The report page will be displayed.	Same as Expected	Pass	-
2	Select the desired report type.	-	The respective report page will be displayed.	Same as Expected	Pass	-
3	Select the event for the report.	-	The report based on the event selected will be displayed accordingly in a graph.	Same as Expected	Pass	-
4	Selected the time range for the report.	-	The report based on the time duration selected will be displayed accordingly in a graph.	Same as Expected	Pass	-

Post-Conditions:

Table 5.4.7.2: Test Case for Report II

Test Case #: T053			**Test Case Name:** To test the report generation function			
System: Diabetes Monitoring System (Web)			**Subsystem:** Report module			
Designed By: Ong Jia Hui			**Design Date:** 21/12/2023			
Executed By: Ong Jia Hui			**Execution Date:** 21/12/2023			
Short Description: To verify that the system correctly generates the blood glucose events report that is required by the doctor.						
Pre-conditions: There is at least one existing patient record in the system.						
Step	**Action / Step**	**Test Data**	**Expected System Response**	**Actual System Response**	**Pass/ Fail**	**Comments**
1	Navigate to the report's main page.	-	The report page will be displayed.	Same as Expected	Pass	-
2	Select the desired report type.	-	The respective report page will be displayed.	Same as Expected	Pass	-
Post-Conditions:						

5.5 Chapter Summary & Evaluation

The development phase of the proposed system has encountered several challenges, including limited time, difficulties in searching for compatible tools and techniques for the development and ensuring compatibility, retrieving sufficient online resources to enhance the performance of the system, each requiring various consideration and effort to overcome. In order to implement the ingredient scanning feature, lots of effort has been inserted to make the flow reasonable and systematic. Another notable challenge emerged during the preparation of the dataset for the model training. Existing datasets in the market did not meet the system's specific requirements, prompting the necessity to collect data one by one. The utilisation of the Spoonacular API proved valuable in gathering relevant nutritional facts, addressing the information gap and contributing to the completeness of the dataset. Furthermore, the integration of the two platforms, developing in different programming languages in which the system is developed in Dart language while the model is trained in Python, introduced extra challenges in terms of compatibility. Time and effort were inverted to ensure seamless integration and correct flow across two platforms.

To ensure the completion of the project on time, the planned schedule for the project is amended timely based on the daily progress. Necessary adjustments to the schedule have been made to ensure the remaining time is sufficient for each task. Experiments were conducted on the ingredient scanning features from different perspectives to cover most of the scenarios the user will encounter, ensuring the flow is reasonable and user-friendly. This meticulous testing ensures the system will be able to respond appropriately, providing relevant error messages based on the different situations. Moreover, online resources have been referred to figure out the method of integrating the system and the trained model, which both developed in different platforms.

Chapter 6

System Deployment

6.1 Deployment Architecture

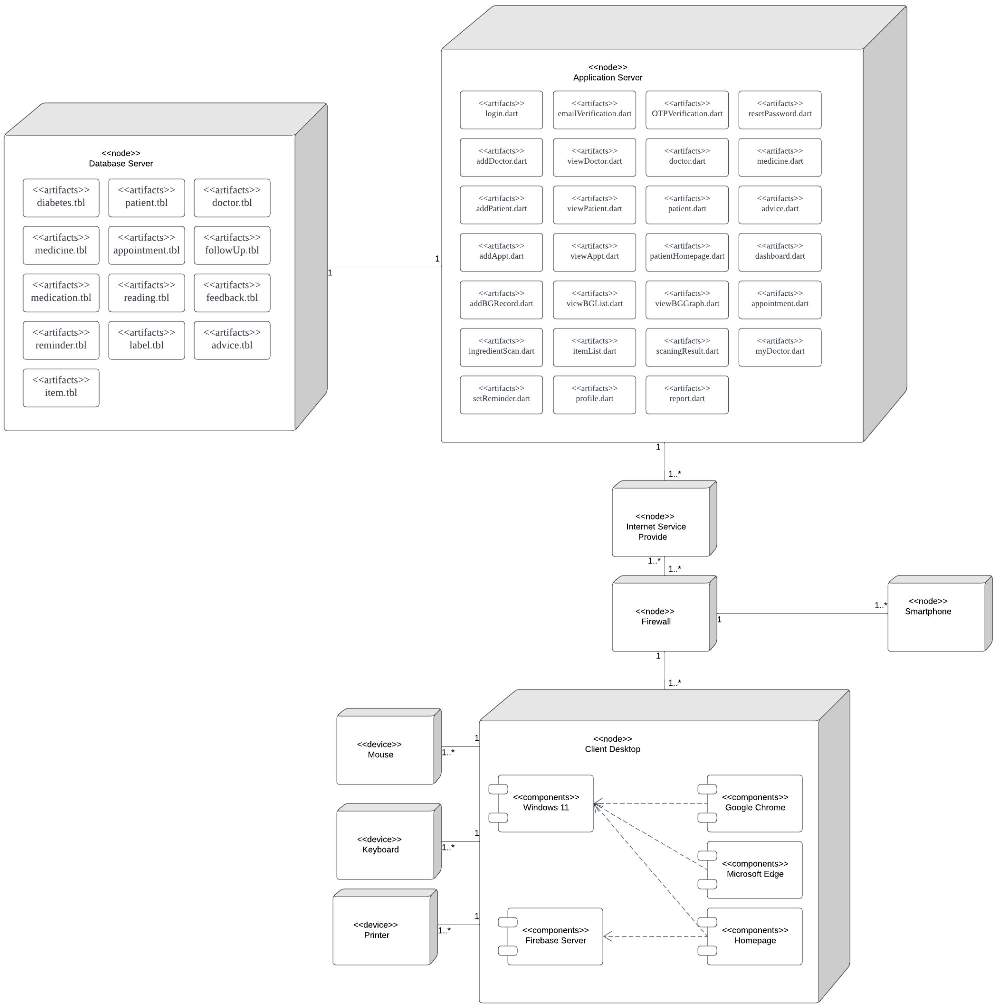


Figure 6.1 Deployment Diagram of the Proposed System

The diagram provides an overview of the deployment architecture for the diabetes monitoring system. As shown in the deployment diagram, the webpages will be stored in GitHub Repository and accessible through GitHub Pages. This leverages GitHub's backup feature for version control, ensuring that future updates to the system can be immediately pushed to GitHub, facilitating seamless web page hosting and storing the latest backup. Utilising GitHub as the application server offers advantages such as minimal setup required for web page hosting. Deployment involves executing a few commands and streamlining the deployment process. Users can access the system through web pages via Google Chrome, Microsoft Edge, or Mozilla Firefox with a stable network connection, where the system dynamically retrieves necessary information from Firebase Cloud Storage based on the user's identity. Firebase Cloud Storage acts as the server for image storing, data collection, and handling user authentication during the login process. Users access the website by entering the URL to which the system resides in GitHub, allowing them to navigate through the pages and interact with the diabetes monitoring system seamlessly. For the mobile platform, the developed system is packaged into a mobile application package, which is then uploaded to the Google Play Store. This enables interested users to download and install the native mobile app onto their smartphones. Once the app is successfully downloaded, users gain access to the diabetes monitoring system through the dedicated native app. The mobile app seamlessly connected to the same Firebase Cloud Storage utilised by the web version of the system to ensure consistent and up-to-date data across both web and mobile platforms.

6.2 Deployment Environment

The diabetes monitoring system is designed to function seamlessly as both a web application and a native mobile app. Via the web application, the users can opt for web browsers, such as Google Chrome. Mozilla Firefox and Microsoft Edge, to access the system via a stable Internet Connection based on their preferences of the capabilities of their devices. The chosen deployment environment for the web application, GitHub Pages, offers easy and direct deployment from the project repository. Any updates or changes to the source code in the repository will automatically trigger updates to the hosted web pages. This ensures that the deployed system is synchronised to the latest version. Utilising GitHub Pages as the platform to deploy the web application ensures a cost-effective solution as additional hosting expenses or hardware to support the hosting can be eliminated.

As for the mobile application, it will be made available to the public by publishing the application package(APK) on the Google Play Store. This ensures a straightforward process for the users to easily discover, download, and install the mobile app on their Android devices. To ensure optimal functionalities on mobile devices, certain constraints on mobile specifications are imposed. For example, mobile devices must be equipped with a camera, and they should run on an Android system of version 11.0 or above, ensuring a seamless user experience. When there are new updates to the mobile application, a new application package (APK) will be generated and uploaded to the Google Play Developer Console. Users are notified of the update by Google Play Store and are prompted to download and install the latest version.

6.3 Deployment Process

6.3.1 Deployment of Web Application

To deploy the web application of the diabetes monitoring system on GitHub Pages, there are several steps and commands to be followed:

1. Create a GitHub Repository (one time)

log in to the GitHub account which will be hosting the web application and create a new repository. An empty repository will be created in that GitHub account once completed.

Remark: it is necessary to create the repository in such format: {yourGithubName}.github.io.

2. Getting Flutter Application Ready

To ensure stable release of the web application, upgrade the project to the latest Flutter SDK version using the command, “flutter upgrade”. Verify the Flutter application is supported by Web and works as expected. When everything is good, the Web can be built by simply running the command “flutter build web”. After the command completes the execution, a folder by the name web under \build\ will appear.

3. Change the directory to build/web

Navigate to the directory build/web via the command, “cd build/web/”.

4. Host Flutter Web App on GitHub Pages

With the build under build\web\, the web application can be uploaded to GitHub Pages for deployment. Run the commands as follows:

- git init

- git remote add origin https://github.com/{yourusername}/{yourusername}.github.io
- git pull origin
- git add.
- git commit -m "Initial Commit"
- git push -u origin main

5. Configure Settings

Under the “Settings” tab of the “GitHub Pages” section, ensure the main branch is selected for the “branch” option (may alter based on actual condition).

After the file has completed uploading, a URL will be provided and the web application will be able to access through this URL (Negi, 2022).

6.3.2 Deployment of Mobile Application

In order to deploy the mobile application on Google Play Store that allows public access, there are several steps and commands to be followed:

1. Create a Google Play Console Developer Account

Create a Google developer account on the Google Play Console with a one-time fee of $25. The Google Play Console will act as the backend operating system for the developers to publish their applications on the Play Store.

2. Set up a Google Merchant Account

Link the developer account created in the previous step to this merchant account if a Google merchant account is owned. Otherwise, it will automatically link to a Google ePla Console account once a new merchant account is created. This step allows monitoring, managing, and analysing the app sales and generating reports.

3. Create Application

Create an application and enter some basic app details such as app name, default language, category and so on.

4. App Store Listing Details

Upload the app on the Google Play Console and fill in some necessary details to set up the app.

5. Content Rating

Rate the mobile application content via a content rating section that is divided into three subsections – Categories, Questionnaire, and Summary.

6. Create and upload the Android App to Google Play

Upload the APK exported from the Flutter project to Google Play Console to publish the app. Google will take some time to review and approve the application before the application goes live. It may take a few hours or up to 7 days.

6.3 Training Procedure

In the context of a diabetes monitoring system, user training will act as a starting guideline to the users, especially the doctors, to streamline their understanding of the system's functionalities. This aims to minimise the need for the users to explore the functionalities on their own, thus saving time.

The training conducted is a one-time session for the doctors while the patients will have to explore the system functionalities on their own. The reason for training only the doctors is to ensure doctors can seamlessly integrate the system and effectively use it within their daily routine. The training will be carried out by the developers to the doctors before the system operates in reality. As sufficient user guidelines and hints are provided by the system, even without formal training, the new doctors would still be able to work on the system and understand how the flow is ongoing.

The training includes demonstrations on the system flow, providing a concept and practical knowledge to the users, which include managing the patients, managing the doctors, handling appointments and remotely monitoring their patients. The goal of this training is to provide the necessary information and equip the users with skills to carry out their daily routine within the system. Moreover, the training session includes a session that answers the user's query, aiding them to have a better understanding of the system's performance.

6.4 Chapter Summary & Evaluation

This chapter outlines the deployment architecture, environment, and processes of the diabetes monitoring system. The system is designed to function as both a web application and a native mobile app. The deployment of the web application will utilise the hosting by GitHub Repository, allowing accessibility through GitHubPages. For the mobile application, the deployment will focus on the Google Play Store, requiring the interested public to download to use. Both applications access the same storage by Firebase Cloud Storage to ensure data accuracy for information retrieval. Once the uploading of web applications via GitHub Pages and the mobile application via Google Play Console is completed, the user can access the system through the website URL or by downloading from the Google Play Store.

Chapter 7

Discussion and Conclusion

7.1 Project Summary

The diabetes monitoring system is designed as a system that caters to both mobile and web platforms, recognising the needs of different users on each platform. The web platform, with its expansive screen size, is tailored to serve healthcare providers and professionals. It provides an ideal environment for healthcare providers to view patient records, visualise data trends and effectively generate comprehensive reports. In contrast, the mobile platform leverages the inherent portability of mobile devices to enhance convenience, targeting the population of diabetes patients. The streamlined communication and data-sharing capabilities inherent in the diabetes monitoring system contribute to achieving effective and efficient diabetes monitoring, eliminating the need for frequent physical follow-ups that can be both infeasible and inconvenient. The web application, designed for use by healthcare providers, incorporates essential functionalities such as patient management, doctor management, report management, diabetes announcements to patients, and monitoring modules. This empowers healthcare professionals to manage the patient's health remotely, owning a centralised platform to oversee this information. Modules such as blood glucose recording, appointment scheduling, retrieval of relevant diabetes information, and tracking of previous records aim to assist the daily routines of diabetes individuals are implemented in the mobile platform. To provide essential dietary information to diabetes individuals, an ingredient scanning feature is incorporated. This feature allows users, especially those with diabetes, to scan the ingredient list on the food packaging effortlessly. Upon scanning, the feature will automatically capture the food items listed in the ingredient list, retrieve pertinent diabetes-related information such as glycemic index (GI) values and glycemic load (GL) values and present all this information to the users. Additionally, the application assesses the consumption suitability of the scanned food items, offering valuable insights to the user to make informed decisions, especially when purchasing packaging products. This helps them to eliminate the risk of consuming potentially harmful ingredients that could lead to severe consequences and complications associated with diabetes. Overall, this system functions as an assistance tool that aims to enhance diabetes self-management with an ingredient scanning feature that helps to reduce consumption risk while integrating with healthcare professionals, ensuring timely and effective monitoring.

7.2 Achievements and Contributions

7.2.1 Achievements

The implementation of the diabetes monitoring system has successfully achieved the objective of establishing a structured platform that facilitates systematic monitoring of diabetes individuals. This centralisation ensures both healthcare providers and patients are accessing a centralised and organised system, promoting the consistency of routines and processes for both parties involved. With a centralised platform and integrated system for both parties, the complex processes and routine for monitoring the patients and at the same time synchronising the data recorded by the patients are aligned consistently. In this way, remote monitoring can take place effectively, reducing infeasible and unnecessary physical follow-up. The streamlined approach to data management contributes to the effectiveness of remote monitoring, minimising the need for infeasible and unnecessary physical follow-ups.

The goal of providing adequate information for informative making has been successfully realised through the implementation of an ingredient scanning feature on the Mobile platform. The implementation of this feature involves several steps, including dataset preparation, utilisation of APIs for essential information retrieval, model training, and seamless integration with the system. To utilise this feature, they have to navigate through the mobile application and scan the ingredient list description on the food packaging. If the image captured is considered valid, consisting of the ingredients list, the system will extract the first 10 items from the list and get the glycemic nutritional value of those extracted items through APIs. Based on these values, the system will perform classification on the consumption suitability via the model trained, providing information to the diabetes individuals. This information becomes crucial for individuals with diabetes, as it empowers them to make informed decisions about the suitability of a particular food product for their dietary needs.

The optimisation of remote monitoring and facilitation of data sharing between diabetes individuals and healthcare providers has been achieved by integrating both healthcare providers and the patient in the same system with single data storage. With this integration, the daily blood glucose readings recorded by the patient are instantly available to healthcare providers in real-time, allowing them to carry out necessary monitoring routines of the patient's health condition. This integration enables healthcare providers to effectively monitor the patient's health data and perform their other routine. The streamlined process of the system enhances the convenience of the healthcare providers to perform essential tasks, making the data organised and highly accessible. This integration ensures a consolidated approach to managing health information, enhancing the efficiency of monitoring processes.

The objective of enhancing the reliability of data collected by diabetes individuals is attained by including sufficient validation and related error messages to the users. When the users are attempting to make a record of the blood glucose data, the system will provide them with a consistent user interface over time so that they can get familiar with the application and suit them into their daily routine. To ensure the data inputted can provide reliable information, necessary validation is included for every user input. When the user accidentally enters irrelevant information, a related error message will be prompted to notify the user about the error input. Different error scenarios will trigger different types of error messages to provide information to the users, allowing them to make necessary changes or corrections.

By successfully implementing the system with streamlined processes and user-friendly features, it can provide convenience to diabetes individuals in getting diabetes care. The implementation ensures the diabetes individual can access the diabetes care resources and information such as ingredient materials, appointment scheduling and monitoring tools within the same platform. This achievement minimises the need for physical visits, reducing the necessity for diabetes individuals to physically meet up with healthcare providers while promoting timely access to healthcare professionals.

7.2.2 Contributions

The diabetes monitoring system can provide significant contributions to diabetes individuals. As compared with existing applications that offer identical functionalities, this system outperforms the others by possessing an ingredient scanning feature that can provide consumption suitability while integrating the systems of the healthcare providers and the patients, ensuring high and efficient information synchronisation. This helps the diabetes individuals to obtain timely invention when their condition requires attention. The system offers a centralised platform for diabetes individuals to carry out their daily routine, such as blood glucose recording, appointment scheduling with their healthcare providers and so forth. By providing a centralised platform that unites various activities, the system streamlines the management process, making it convenient for diabetes individuals to adhere to and remain consistent with their monitoring and healthcare routine.

The system demonstrates a notable contribution in streamlining the processes and flow of the routine by the healthcare providers. As healthcare providers have complex routines and tasks every day, they may miss out on some information or encounter constraints to manage their patients from different perspectives. Via the system, healthcare providers can achieve efficient and effective performance as their workflow has been streamlined by the system, especially the information flow. This ensures the critical data related to the patients are always available and up-to-date when it is required, allowing quick responses to improve the patient's health status. Via the association of a well-organised system, healthcare providers can allocate more time and resources, focusing on more critical issues.

7.3 Limitations and Future Improvements

7.3.1 Limitations

One significant limitation of the system is found in the ingredient scanning feature. The current implementation of the ingredient scanning feature allows the users to perform scanning on the ingredient list of the food packaging. Upon verifying the scanned text is valid, consisting of the content required, the system will generate the consumption suggestions along with glycemic information for the identified ingredients. However, the output of this ingredient scanning feature is currently constrained to suggesting the consumption suitability, either yes or no. This binary classification may confuse the users as it fails to communicate efficiently through detailed information on the degree of suitability. It leaves the users with limited knowledge of to what extent the consumption is suitable or vice versa.

Another notable limitation within the ingredient scanning feature is the inability of the feature to provide sufficient information as output. Presently, the output from this feature supports only binary classification on consumption suitability. It lacks additional information such as portion recommendations, ingredient explanations and so forth to increase the knowledge and understanding of users. This information may enhance the capability of the users to make better-informed decisions as when more factual information is provided, the users can have a higher understanding of the particular food.

7.3.2 Future Improvements

To address the limitations on the presentation of consumption suitability, a better visualisation will be implemented. Instead of including merely the suitability binary category of either yes or no, consider adding a dynamic range bar that features a gradient colour schema, transitioning from red to green. The position of the pointer will indicate the suitability level of the particular scanned food, providing information on the extent of the consumption level. This promotes a more intuitive understanding of suitability levels, deterring potential information insufficient to the suitability extent.

While to address the limitations of insufficient information offered as the output, efforts on exploring suitable APIs that can provide that information such as recommended portions, and explanations of the ingredient items will be considered. If possible, similar APIs that can provide the information required will be compared and the most suitable will be selected.

7.4 Issues and Solutions

One of the challenges faced during the system development is to allow responsiveness for the project while working on the web platform. Although Flutter supports system development for mobile and web platforms in a single codebase, it does not have built-in libraries that have features such as BootStrap to handle responsiveness. This requires extra effort in developing to ensure the system adapts to the changing screen sizes effectively. Some Flutter capabilities such as the attribute ‘MediaQuery’, found able to return the device characteristics such as weight and height, have been utilised to dynamically determine the screen size and adjust the layout or styling based on the actual screen size. Conditional logic is also implemented to ensure the system responds to different screen sizes, delivering a satisfying user experience.

Another challenge during the system development is the integration of the trained machine learning model into the Flutter project to ensure effective communication between the model and the system. As the model is developed via the Python programming language, it is incompatible with working directly with the system which is developed in Dart language. One of the solutions is converting the well-trained model into a compatible format or package via tools like TensorFlow Converter. TensorFlow Converter enables the export of the model to the 'tflite' format. By leveraging the TensorFlow Converter library, the model trained can be exported as a ‘tflite’ file and integrated with the Flutter project.

7.5 Conclusion

Overall, the implemented diabetes monitoring system is a comprehensive solution designed to cater for the diverse needs of healthcare providers and diabetes individuals. The web platform has a wider and clearer screen and serves the healthcare provider by offering functionalities such as patient and doctor management, report generation, and data visualisation. The mobile platform that has high portability serves diabetes individuals with features such as blood glucose recording, appointment scheduling, and retrieval of relevant diabetes information. An additional feature included in the mobile platform is the ingredient scanning feature, which allows the individual to scan the ingredient list on food packaging, obtaining information such as glycemic index(GI) and glycemic load(GL) values. It further assesses the consumption suitability of the scanned food items, providing valuable insights to the users for informed decision-making, and reducing potential risks such as diabetes comorbidities.

The current state of the proposed system is constrained in some perspectives and is expected to have improvement in the future. This includes the representation of the classification result that does not provide sufficient information to showcase the extent of the actual suitability. This limitation is planned to be improved by enhancing the representation of the classification result through a range bar. Furthermore, the ingredient scanning feature that is implemented in the system provides limited information as the output. Therefore, the effort will be considered to explore suitable APIs that can provide the required information in the future.

Some challenges encountered during the system development include achieving responsiveness to the web application of the system. Due to the absence of built-in libraries such as Bootstrap that offer responsive, extra effort such as utilisation of the ‘MediaQuery’ attribute is required to dynamically determine the screen size, adjusting the layout and styling accordingly. Another notable challenge is the integration of trained machine-learning models into the Flutter project. To ensure the compatibility of the model and system, conversion on the trained model is performed, exporting the model as a ‘tflite’ file before integrating with the Flutter project.

References

Adeva-Andany, M. M., Rañal-Muíño, E., Vila-Altesor, M., Fernández-Fernández, C., Funcasta-Calderón, R., & Castro-Quintela, E. (2019). Dietary habits contribute to define the risk of type 2 diabetes in humans. In Clinical Nutrition ESPEN (Vol. 34, pp. 8–17). Elsevier Ltd. https://doi.org/10.1016/j.clnesp.2019.08.002

Aguilar-Loja, O., Armas-Aguirre, J., Dioses-Ojeda, L., & Gonzalez, P. A. (2022). A decision tree–based classifier to provide nutritional plans recommendations. https://ieeexplore.ieee.org/document/9820144

Al-Saqqa, S., Sawalha, S., & Abdelnabi, H. (2020). Agile software development: Methodologies and trends. International Journal of Interactive Mobile Technologies, 14(11), 246–270. https://doi.org/10.3991/ijim.v14i11.13269

Almuhajri, M., & Ching, Y. S. (2022). A Complete Framework for Shop Signboards Detection and Classification. IEEE Xplore. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9956399

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. Journal of Big Data, 8(1). https://doi.org/10.1186/s40537-021-00444-8

Anis Suraya Rosmahadi, N., & Fauzee Hamdan, M. (2022). Statistical Analysis of Food Nutrition by Using K-Means Clustering. IEEE Xplore, 8, 159–169. https://www.myfitnesspal.com/.

Anuradha, & Sengar, N. (2020). HANDWRITTEN TEXT RECOGNITION USING TENSORFLOW. IEEE Xplore, 11. www.jespublication.com

Atlassian. (2023). MTBF, MTTR, MTTF, MTTA: Understanding incident metrics. Atlassian. https://www.atlassian.com/incident-management/kpis/common-metrics

Aschemann-Witzel, J., Varela, P., & Peschel, A. O. (2019). Consumers' categorization of food ingredients: Do consumers perceive them as 'clean label' producers expect? An exploration with projective mapping. Food Quality and Preference, 71, 117–128. https://doi.org/10.1016/J.FOODQUAL.2018.06.003

B, K., George, D. J., Manikandan, G., & Thomas, T. (2020). A Comparative Study on K-Means Clustering and Agglomerative Hierarchical Clustering. International Journal of Emerging Trends in Engineering Research, 8(5), 1600–1604. https://doi.org/10.30534/ijeter/2020/20852020

Bi, Q., Goodman, K. E., Kaminsky, J., & Lessler, J. (2019). What is machine learning? A primer for the epidemiologist. American Journal of Epidemiology, 188(12), 2222–2239. https://doi.org/10.1093/aje/kwz189

Burkpalli, V., Joshi, A., Warad, A. B., & Patil, A. (2022). AUTOMATIC NUMBER PLATE RECOGNITION USING TENSORFLOW AND EASYOCR. Www.Irjmets.Com @International Research Journal of Modernization in Engineering, 493. www.irjmets.com

Business News Daily. (2023). What Is Agile Scrum Methodology? - businessnewsdaily.com. https://www.businessnewsdaily.com/4987-what-is-agile-scrum-methodology.html

CDC. (2022). Food and Nutrition Insecurity and Diabetes : Understanding the Connection. CDC. https://www.cdc.gov/diabetes/library/features/diabetes-and-food-insecurity.htm

Chatrati, S. P., Hossain, G., Goyal, A., Bhan, A., Bhattacharya, S., Gaurav, D., & Tiwari, S. M. (2022). Smart home health monitoring system for predicting type 2 diabetes and hypertension. Journal of King Saud University - Computer and Information Sciences, 34(3), 862–870. https://doi.org/10.1016/j.jksuci.2020.01.010

Chitayae, N., & Sunyoto, A. (2020). Performance Comparison of Mushroom Types Classification Using K-Nearest Neighbor Method and Decision Tree Method. 2020 3rd International Conference on Information and Communications Technology, ICOIACT 2020, 308–313. https://doi.org/10.1109/ICOIACT50329.2020.9332148

Department of Statistics, M. (2021). Press Release Statistics on Causes of Death, Malaysia. 1, 8. https://www.dosm.gov.my/v1/index.php?r=column/cthemeByCat&cat=401&bul_id=R3VrRUhwSXZDN2k4SGN6akRhTStwQT09&menu_id=L0pheU43NWJwRWVSZklWdzQ4TlhUUT09

Google. (2023). Firebase pricing. Google. https://firebase.google.com/pricing

Google. (2023). USD/Myr Currency Exchange Rate & News. Google Finance. https://www.google.com/finance/quote/USD-MYR?sa=X&sqi=2&ved=2ahUKEwilqvas3IuAAxV4cGwGHdngAjwQmY0JegQIDRAc

Google Firebase. (2023). Firebase authentication. Google Firebase. https://firebase.google.com/docs/auth

Google Firebase. (2023). Namespace: Hashing | Firebase. Google Firebase. https://firebase.google.com/docs/reference/rules/rules.hashing

Harumy, T. H. F., Ginting, D. S., & Manik, F. Y. (2023). Comparison of Classification Methods and Clustering Hybrid Deep Neural Network Detection of Sensitive Ingredients in Food Products. Journal of Theoretical and Applied Information Technology, 31(2). www.jatit.org

Hema, V., Thota, S., Naresh Kumar, S., Padmaja, C., Rama Krishna, C. B., & Mahender, K. (2020). Scrum: An Effective Software Development Agile Tool. IOP Conference Series: Materials Science and Engineering, 981(2). https://doi.org/10.1088/1757-899X/981/2/022060

Hu, G., Ahmed, M., & L'Abbé, M. R. (2023). Natural language processing and machine learning approaches for food categorization and nutrition quality prediction compared with traditional methods. The American Journal of Clinical Nutrition, 117(3), 553–563. https://doi.org/10.1016/j.ajcnut.2022.11.022

Hubert, Pheonix, P., Sudaryono, R., & Suhartono, D. (2021). Classifying Promotion Images Using Optical Character Recognition and Naïve Bayes Classifier. ScienceDirect. https://doi.org/10.1016/j.procs.2021.01.033

IDF Diabetes Atlas | Tenth Edition. (2021). IDF Diabetes Atlas. https://diabetesatlas.org/

Jamal, A., Tharkar, S., Babaier, W. S., Alsomali, S. F., Alsulayhim, A. S., Alayuni, M. A., Aldakheel, N. A., Al-Osaimi, S. S., Alshehri, N., & Batais, M. (2021). Blood glucose monitoring and sharing amongst people with diabetes and their facilitators:

Cross-sectional study of methods and practices. JMIR Diabetes, 6(4). https://doi.org/10.2196/29178

Javadi, B., Trieu, Q. L., Matawie, K. M., & Calheiros, R. N. (2020). Smart Food Scanner System Based on Mobile Edge Computing. Proceedings - 2020 IEEE International Conference on Cloud Engineering, IC2E 2020, 20–27. https://doi.org/10.1109/IC2E48712.2020.00009

Jiang, T., Gradus, J. L., & Rosellini, A. J. (2020). Supervised Machine Learning: A Brief Primer. ScienceDirect. https://www.sciencedirect.com/science/article/pii/S0005789420300678?via%3Dihub

Khairani, D., Bangkit, D. A., Rozi, N. F., Masruroh, S. U., Oktaviana, S., & Rosyadi, T. (2022). Named-Entity Recognition and Optical Character Recognition for Detecting Halal Food Ingredients: Indonesian Case Study. 2022 10th International Conference on Cyber and IT Service Management, CITSM 2022. https://doi.org/10.1109/CITSM56380.2022.9935966

Khan, M. A. B., Hashim, M. J., King, J. K., Govender, R. D., Mustafa, H., & Kaabi, J. Al. (2020). Epidemiology of Type 2 Diabetes – Global Burden of Disease and Forecasted Trends. Journal of Epidemiology and Global Health, 10(1), 107. https://doi.org/10.2991/JEGH.K.191028.001

Kartiwi, M., Gunawan, T. S., Anwar, A., Siti, #, & Fathurohmah, S. (2019). Mobile Application for Halal Food Ingredients Identification using Optical Character Recognition. In Measurement and Applications. https://www.researchgate.net/publication/332434951

Li, J., Sun, A., Han, J., & Li, C. (2022). A Survey on Deep Learning for Named Entity Recognition. IEEE Transactions on Knowledge and Data Engineering, 34(1), 50–70. https://doi.org/10.1109/TKDE.2020.2981314

Li, J., & Wang, Z. (2019). Real-Time Traffic Sign Recognition Based on Efficient CNNs in the Wild. IEEE Xplore. https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8392744&casa_token=LCtsU_geF3cAAAAA:c5Dz9phOavqayYk7Otx-RPkflzVpL4MEytL90wYYKkEFVRUGGzHWfJhBRqkcChRVThg7HdfF6z9y6Vo&tag=1

International Diabetes Federation. (2019). IDF DIABETES ATLAS Ninth edition 2019. International Diabetes Federation. https://diabetesatlas.org/upload/resources/material/20200302_133351_IDFATLAS9e-final-web.pdf

IPH, I. for P. H., NIH, N. I. of H., & Malaysia, M. of H. (2019). National Health and Morbidity Survey (NHMS) 2019: NCDs - Non-Communicable Diseases: Risk Factors and other Health Problems. In Institute for Public Health, National Institutes of Health (NIH), Ministry of Health Malaysia (Vol. 1). http://www.iku.gov.my/nhms-2019

Mahdi, S., Buckland, N. J., & Chilcott, J. (2023). Economic and health impacts of the Change4Life Food Scanner app: Findings from a randomized pilot and feasibility study. Frontiers in Nutrition, 10. https://doi.org/10.3389/fnut.2023.1125542

Mahdi, S., Chilcott, J., & Buckland, N. J. (2022). Meeting Abstracts Evaluating the Change4Life Food Scanner app in reducing children’s sugar intake: a randomised pilot and feasibility study. https://doi.org/10.1016/S0140-6736(22)02223-1

Maneze, D., Weaver, R., Kovai, V., Salamonson, Y., Astorga, C., Yogendran, D., & Everett, B. (2019). “Some say no, some say yes”: Receiving inconsistent or insufficient information from healthcare professionals and consequences for diabetes self-management: A qualitative study in patients with Type 2 Diabetes. Diabetes Research and Clinical Practice, 156. https://doi.org/10.1016/J.DIABRES.2019.107830

Marcos, I. F.-V., Bordel, B., Cira, C.-I., & Alcarria, R. (2022). A Methodology Based on Unsupervised Learning Techniques to Identify the Degree of Food Processing. IEEE Xplore. https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9820513&casa_token=AhZGVeqLa_oAAAAA:rDbZHpw0SXrLuoB4TVkyry0_oGlee_eHAy0PlpUxkiOv4iNlfXR1S8ZbtCEOAQnPeJqGdemdph2gX0g

Makhsous, S., Bharadwaj, M., Atkinson, B. E., Novosselov, I. V, & Mamishev, A. V. (2020). DietSensor: Automatic Dietary Intake Measurement Using Mobile 3D Scanning Sensor for Diabetic Patients. https://doi.org/10.3390/s20123380

Mel Studio. (2023). Diaries of health. Mel Studio. Retrieved 17 May 2023, from https://melstudio.info/en/apps/dnevniki-zdorovya

mySugr. (2023). mySugr + iPDM | mySugr Global | mySugr. MySugr. Retrieved 17 May 2023, from https://www.mysugr.com/en/ipdm/

Nasrabadi, A. N., Dehkordi, L. M., Khoshkesht, S., & Najafi, F. (2021). Unsuccessful diabetes management: A qualitative study. Clinical Diabetology, 10(2), 195–199. https://doi.org/10.5603/DK.a2021.0012

Negi, A. (2022) How to host your flutter web app on GitHub Pages?, BrewYourTech. Available at:

https://brewyourtech.com/how-to-host-your-flutter-web-app-on-github-pages/ (Accessed: 26 December 2023).

NIH, N. I. of H. (2020). National Health and Morbidity Survey 2019. 20, 2–2. https://doi.org/10.18356/be4d1601-en

Onix. (2022). 7 Best Software Development Models: Which One to Choose? Onix. https://onix-systems.com/blog/7-basic-software-development-models-which-one-to-choose

ÖZCAN , A., ÇATAL, Ç., TOĞAY , C., TEKİNERDOĞAN, B., & DÖNMEZ , E. (2020). Turkish Journal of Electrical Engineering and Computer Sciences. https://journals.tubitak.gov.tr/cgi/viewcontent.cgi?article=1294&context=elektrik

P. Rajendran, S., Shine, L., Pradeep, P., & Vijiyaraghavan, S. (2019). Real-Time Traffic Sign Recognition using YOLOv3 based Detector. IEEE Xplore. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8944890

Papamichou, D., Panagiotakos, D. B., & Itsiopoulos, C. (2019). Dietary patterns and management of type 2 diabetes: A systematic review of randomised clinical trials. Nutrition, Metabolism and Cardiovascular Diseases, 29(6), 531–543. https://doi.org/10.1016/j.numecd.2019.02.004

Ray, S. (2019). A Quick Review of Machine Learning Algorithms. 14–16. https://translateyar.ir/wp-content/uploads/2021/12/A-Quick-Review-of-Machine.pdf

Reddy, P. S. V., Krishna, M. V., Aishwarya, R., Yogitha, R., & Kumar, K. A. (2023). Fish Species Classifier for Allergic People using CNN Algorithm. Proceedings - 7th

International Conference on Computing Methodologies and Communication, ICCMC 2023, 489–494. https://doi.org/10.1109/ICCMC56507.2023.10084124

Rishabh, M., & Garg, A. (2020). Text extraction using OCR: A Systematic Review. IEEE Xplore. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9183326

Rohini, B., Madhuri, P., Naresh Kuamr, L. S., Soorya, V., & Aravinth, J. (2021). A Framework to Identify Allergen and Nutrient Content in Fruits and Packaged Food using Deep Learning and OCR. IEEE Xplore. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9441800

Rua, R., Silva, V., Muhammad, S., & Duarte, F. (2019, January 31). MAPiS 2019 - first map-I seminar proceedings - core. https://core.ac.uk/download/pdf/231952855.pdf

Sai, P. H., Rao, N., Sri, T., Kailash, S., Chandra, A. B., & Ralhan, C. (2022). MEAL PLAN PREDICTION USING DECISION TREE CLASSIFIER: A REVIEW. In International Research Journal of Modernization in Engineering Technology and Science www.irjmets.com @International Research Journal of Modernization in Engineering (Vol. 165). www.irjmets.com

Sarker, I. H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. In SN Computer Science (Vol. 2, Issue 3). Springer. https://doi.org/10.1007/s42979-021-00592-x

Siriwardhana, Y., Porambage, P., Liyanage, M., & Ylianttila, M. (2021). A Survey on Mobile Augmented Reality with 5G Mobile Edge Computing: Architectures, Applications, and Technical Aspects. In IEEE Communications Surveys and Tutorials

(Vol. 23, Issue 2, pp. 1160–1192). Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/COMST.2021.3061981

simplilearn. (2023). Convolutional Neural Network Tutorial [Update]. Simplilearn. https://www.simplilearn.com/tutorials/deep-learning-tutorial/convolutional-neural-network

Scrum Org. (2023). Scrum Guides. Scrum.Org. https://scrumguides.org/scrum-guide.html

Sun, F., Gu, Z., & Feng, B. (2018). Yelp Food Identification via Image Feature Extraction and Classification. IEEE International Conference on Program Comprehension, 2022-March, 36–47. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Tchero, H., Kangambega, P., Briatte, C., Brunet-Houdard, S., Retali, G. R., & Rusch, E. (2019). Clinical Effectiveness of Telemedicine in Diabetes Mellitus: A Meta-Analysis of 42 Randomized Controlled Trials. Telemedicine Journal and E-Health : The Official Journal of the American Telemedicine Association, 25(7), 569–583. https://doi.org/10.1089/TMJ.2018.0128

Uddin, S., Haque, I., Lu, H., Moni, M. A., & Gide, E. (2022). Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction. Scientific Reports, 12(1). https://doi.org/10.1038/s41598-022-10358-x

United Nations. (2023). Goal 3 | Department of Economic and Social Affairs. United Nations. Retrieved 21 May 2023, from https://sdgs.un.org/goals/goal3

Wang, J., & Biljecki, F. (2022). Unsupervised machine learning in urban studies: A systematic review of applications. Cities, 129. https://doi.org/10.1016/j.cities.2022.103925

What is Firebase?. Educative. (2023). https://www.educative.io/answers/what-is-firebase#:~:text=Firebase%20is%20a%20Backend%2Das,data%20in%20JSON%2Dlike%20documents.

World Health Organization, W. (2022). Noncommunicable diseases. World Health Organization. https://www.who.int/health-topics/noncommunicable-diseases

Yang Lee, J., Piau Wong, C., & Wen Huey Lee, S. (2020). Health and Well-being Cluster, Global Asia in the 21st Century (GA21) Platform. National Library of Medicine. https://doi.org/10.21037/mhealth.2019.09.16

Yuan, C., & Yang, H. (2019). Research on K-Value Selection Method of K-Means Clustering Algorithm. J, 2(2), 226–235. https://doi.org/10.3390/j2020016

Zachariah, B., Nonyelum, O. F., & Student, P. D. (2020). A Comparative Analysis of Requirement Gathering Techniques A Comparative Analysis of Requirement Gathering Techniques. https://www.proquest.com/docview/2434436046/fulltextPDF/8ED913A2CD134158PQ/1?accountid=38945

Appendix

Appendix 1: Proposal

2022/23 FORM 2: Project Proposal

TARUMT
TUNKU ABDUL RAHMAN UNIVERSITY OF MANAGEMENT AND TECHNOLOGY

ONG JIA HUI
RSW
016-4679618
ongjh-pm20@student.tarc.edu.my
Ts. Chau Guan Hin

PROJECT TITLE: DIABETES MONITORING SYSTEM WITH INGREDIENT SCANNING FOR INFORMATIVE FOOD FILTERING

ABSTRACT

Neglected management of diabetes can lead to severe complications and detrimental effects. However, this negligence often stems from unfamiliarity or lack of knowledge regarding proper monitoring and management approaches, rather than intentional disregard. Many individuals with diabetes also face challenges in managing their dietary habits due to confusion about ingredients. Therefore, this diabetes monitoring system is here to help individuals in recording their daily measurements with guidance on the recommended frequency and timing, providing recommendations on food suitability by checking through the ingredients list. Besides, to enhance the effectiveness of monitoring and management, the recorded information will be synchronised with their healthcare provider for professional analysis. This system offers individuals convenience in managing themselves as it presents the data in an easily understandable format so that they can gain insight into their health trends. Furthermore, the system provides comprehensive information about ingredients, empowering individuals to make informed decisions about their diet.

PROBLEM

Diabetes mellitus is defined as a metabolic disorder characterised by hyperglycemia resulting from either deficiency in insulin secretion or insufficient insulin. According to the IDF Diabetes Atlas 10th edition 2021, a total of 537 million adults, around 1 in 10 adults, are living with diabetes worldwide. The finding by the IDF showed that about 3 in 4 diabetes individuals are living in low- and middle-income countries. Additionally, the number is predicted to rise spirally to 643 million by 2030 and 783 million by 2045 (IDF Diabetes Atlas | Tenth Edition, 2021). In the year 2019, the National Health and Morbidity Survey (NHMS) done by the Ministry of Health Malaysia(MOH) recorded a diabetes prevalence of 18.3% in Malaysia, estimated about 1 in 5 adults are living with diabetes. In other words, there are about 3.9 million people that are aged 18 years and above having diabetes (NIH, 2020). This was a noticeable increase when compared to the prevalence concluded from the survey done in 2015, which recorded a rate of 13.4% (IPH et al., 2019). Paying little attention to diabetes may lead over time to cause serious damage to the body, organ or tissue failure and eventually causing death. In the statistics of death causes in Malaysia, diabetes has become one of the 10 principal death causes in Malaysia, with a rate of 2.0%, ranking sixth (Department of Statistics, 2021). Therefore, diabetes has turned up to be one of the significant health issues that require immediate attention and monitoring.

Until now, there is no standardised method that diabetes individuals practise to record their daily measurements, especially in terms of frequency and time. Most individuals are collecting their daily measurements in the way that they prefer. This can potentially lead to irregular or ambiguous information being collected, causing insufficient information for their treatment analysis later. A recent

1

Figure A.1: Proposal Page 1

ONG JIA HUI
RSW
016-4679618
ongjh-pm20@student.tarc.edu.my
Ts. Chau Guan Hin

study stated that the majority of diabetics, about 95%, measured their blood glucose levels but only 72% of them will do the measuring daily. Of those who monitored their blood glucose levels, 76% recorded the measurements, and among them, 55.6% preferred paper-based methods and 44.4% preferred digital devices (Jamal et al., 2021). As shown in the statistics from the study, the individuals note their data according to their preferences and some do not even carry out daily measurements. The inconsistency in recording the data by the individuals may cause inconvenience to health providers when trying to trace back their health conditions. The data may be insufficient to show an observable and useful trend or unable to act as valid data due to non-continuous recording. Not only that, diabetes individuals may lack the consensus to monitor themselves properly and professionally, especially when they are receiving inconsistent and insufficient information from healthcare professionals. This undermines their ability to effectively self-manage their diabetes (Maneze et al., 2019). For instance, they may be unaware of the correct frequency or timing to perform the measurements, leading to inaccurate readings. This improper information recording exacerbates the challenge for the healthcare provider in understanding or analysing the individual's current condition properly.

Due to a lack of professional knowledge in healthcare and food science, many diabetes individuals are easily trapped in purchasing food that may contain unsuitable ingredients. This is the most common condition that happens when they are unable to understand the ingredient list or calorie table shown on the product packaging. Not only that, consumers sometimes tend to categorise the value of food based on their own subjective individual assessment, which may be biassed (Aschemann-Witzel, Varela and Peschel, 2019). This will end up exposing themselves to food that is unsuitable for them but they are not aware of the conditions. For individuals who have diabetes, it is important for them to strictly control their diet as their dietary patterns are one of the factors affecting diabetes. Maintaining a diabetes-friendly diet is critical in preventing the consumption of items that may bring hazardous effects such as frequent spikes in blood glucose levels (CDC, 2022). One of the dietary styles, the Mediterranean diet, which is a diet that is mostly but not exclusively plant-based, has been proven can significantly reduce the risk of severe diabetes complications by approximately 23% (Papamichou et al., 2019). However, achieving the objective of implementing an optimal diet is not simple as individuals must be able to understand the ingredient of the food clearly before purchasing or consuming it. Not only that, they must also equip themselves with a certain level of knowledge to analyse the items listed in the ingredient list. This will undoubtedly bring difficulties to those ordinary individuals, who have only general knowledge about food and ingredients, to identify whether a certain food is suitable for them.

SOLUTION

In response to the rising prevalence of diabetes, a diabetes monitoring system is introduced to assist individuals with diabetes in managing themselves more effectively and efficiently. An ingredient scanning feature will be included in this system to aid the individuals in managing themselves in terms of their consumption. Through the ingredient scanning feature, the individual will be able to perform a proper assessment of their consumption, which is critical for maintaining the health of a diabetic patient (Makhsous et al., 2020).

The diabetes monitoring system provides an effective and structured platform for diabetes individuals to monitor their blood glucose levels, standardise their measurement recording and visualise their data

Figure A.2: Proposal Page 2

2022/23 FORM 2: Project Proposal

ONG JIA HUI
RSW
016-4679618
ongjh-pm20@student.tarc.edu.my
Ts. Chau Guan Hin

systematically. The recorded blood glucose measurement will be presented in an observable trend to aid the identification of any abnormal rising or dropping effectively. Notifications and alerts will be generated to the individuals when the glucose levels need immediate attention such as going too high or approaching a risky level for a prolonged period. Meanwhile, healthcare providers can review the data recorded by their patients in the system remotely. This helps the healthcare provider to observe the trend of the individuals from time to time since the measurements are real-time based. If any abnormalities are detected, the healthcare providers can immediately communicate with the individuals to carry out any necessary check-ups and provide feedback to preserve the blood glucose trend back to normal trend. This helps to standardise the measurement recording, ensuring the data readability by the individuals or their healthcare provider.

The ingredient scanning feature will be implemented in the system, aiming to encounter the concern of unclear information regarding the ingredients. Thus, this feature will provide guidance and recommendations to diabetes individuals based on the ingredient list of the food product, ensuring they are away from risky consumption that could lead to diabetes comorbidities. To access this functionality, the individuals are required to first capture the ingredient list on the food packaging and the system will then process a recommendation output based on the ingredient list captured. The system will recognise and extract possible useful information from that particular capture to come out with the necessary guidance and information. When the individuals capture the ingredient list or the calorie intake table from the food packaging, information is retrieved based on the image. Information from the calorie table that displays the calorie amount, percentage of carbohydrates, proteins and fats intake will also be taken into account in the analysis of the suitability of the consumption. The system will then determine whether any of the ingredients within are unsuitable for the consumption of diabetes individuals by checking through the glycemic index (GI) level, calorie percentages for each item on the ingredient list, comparing the values to the standard that a diabetes individual must obey. After performing analysis on those criteria, the system will generate output that determines the food's suitability for the consumption of a diabetes individual.

The system will provide individuals with a comprehensive analysis and categorise the food items into three main categories: suitable, not recommended, and unsuitable. For foods in the "suitable" category, the system will calculate the total nutritional intake based on the serving amount. Along with the total nutrition intake, the system will provide recommendations on the portion size that falls within the suitable range for consumption. Additionally, alternative food choices that are better than the captured food will be suggested. These alternatives will be selected based on factors such as nutritional content, glycemic index, or other relevant considerations. For instance, if the captured food is high in saturated fat, the system may suggest lean protein sources as alternatives. For foods in the "unsuitable" category, the system will convey information about why the consumption is not suitable. It will highlight specific reasons, such as high sugar content, unhealthy fats, or ingredients that may lead to adverse effects. In the case of foods categorised as "not recommended," the system will provide calorie and nutrition information for the ingredient in the not recommended category. Along with the calorie and nutrition information, the system will explain the reasons why the food or ingredients are not recommended. It may point out factors such as high sodium content, excessive added sugars, or low nutritional value. The system differentiate between the categories of "unsuitable" and "not recommended" based on the percentages of presence of ingredients that exceed the consideration criteria, such as GI levels, carbohydrate or protein percentages, and other relevant factors. If more than half of the ingredients

3

Figure A.3: Proposal Page 3

Ong Jia Hui
RSW
016-4679618
ongjh-pm20@student.tarc.edu.my
Ts. Chau Guan Hin

exceed the standards, the food will be categorised as "unsuitable," while it will be categorised as "not recommended" if it does not meet certain specific criteria. By providing this feature in the system, it can empower individuals' knowledge in making healthier food choices to manage their dietary habits more effectively.

Technically, the ingredient scanning features in the system will utilise image processing technology to extract information from images captured by individuals. In particular, the algorithm used for this proposed solution is k-nearest neighbours (KNN), which can be used to perform the task of classifying the consumption suitability for the diabetes individuals. When the individuals capture the ingredient table and the calorie intake table, optical character recognition (OCR) will be performed to extract the ingredient information from the images. By extracting the ingredient information, the system can gather important data such as glycemic index (GI) levels that are relevant to the ingredients. These details are then taken into account when determining the level of suitability for a particular food item.

TARGET MARKET

The target market of this system will focus on all diabetes individuals, including those diagnosed with prediabetes, type 1 or type 2 diabetes, gestational diabetes or insulin resistance, the healthcare providers, who are the specialists that work to monitor and advise diabetes patients.

The primary target market focus of the system is those individuals with diabetes, who require a systematic tool in monitoring and managing their blood sugar levels regularly to prevent the development of possible complications. The individuals may require a system to help them store their real-time blood sugar information consistently so that they can have an easy insight into their lifetime measurements. Retrieving the measurement information of a certain timeline will become convenient if everything is recorded neatly within the system. They can utilise the information generated by the system in analysing their current health condition so that they can detect possible deterioration earlier. For diabetes individuals, one of the ways to remain healthy is through well-managed dietary habits. The system also provides significant information and guidance about the food ingredients. This helps to eliminate the misleading and misunderstanding issue of the ingredient information, directing a proper consumption manner.

Additionally, the system can become a powerful supportive tool, particularly for those from medium or low-income populations. The majority of the diabetes population, approximately 79.4%, come from low- and middle-income backgrounds (International Diabetes Federation, 2019). This indicates that these individuals are facing significant challenges in accessing proper healthcare due to limited financial resources. People from low- or middle-income groups are at a higher risk of developing comorbidities of diabetes due to the hazardous home environment, unhealthy lifestyle, obesity or stress (Lam et al., 2021) as they are lacking economic support to get professional advice and guidance. Therefore, the system can be used by these individuals to lead themselves efficiently for better health outcomes with the minimum cost required. By leveraging this system, they can utilise it for continuous and remote monitoring of their health and well-being with negligible additional cost.

From the perspective of healthcare providers, this system can foster their operations, making their daily routine work smoother. Due to the increasing number of diabetes individuals recently, healthcare

Figure A.4: Proposal Page 4

ONG JIA HUI
RSW
016-4679618
ongjh-pm20@student.tarc.edu.my
Ts. Chau Guan Hin

providers who offer diabetes care consultation or follow-up are facing high working rates (Khan et al., 2020). Therefore, they may consider implementing an automation tool that can reduce their workload, allowing them to focus on more critical issues that require manual monitoring. The diabetes monitoring system will be useful in such conditions as it can take over the work of monitoring diabetes individuals daily measurements and only alert the healthcare providers in necessary conditions. This can lighten the burden of the healthcare provider to a certain level.

For guardians or family members of diabetes individuals, they are as well the potential population that can utilise the system to monitor their loved one's condition. Once diabetes individuals make the necessary records into the system, these data will be able to be retrieved by their guardians or family members. In this way, the system makes it helpful for them to monitor diabetes individuals, checking if they have done necessary meditation or if any possible abnormal levels are developing. It will be especially friendly for those who wish to monitor diabetes individuals but are busy to always keep themselves around diabetes individuals. For example, children who wish to monitor their parents on time may utilise this system to get to know their parents' real-time condition even if they are not staying with them.

Last but not least, the system provides an excellent platform for those health-conscious individuals to prioritise their health and well-being. Regardless of having diabetes, they can still use the functionalities implemented in the system to achieve their objectives. They can use the ingredient scanning feature provided by the system to pay attention to their dietary choices, striving for a well-rounded, nutritious diet. Through the information generated by the system, they can make known decisions to decide or limit their intake of certain processed food that may contain ingredients that violate their healthy principle. Health-conscious individuals usually practise mindful eating habits, thus this system can be useful for them in determining consumption.

COMPETITION/CONTRIBUTION

In 2021, artificial intelligence (AI) in the healthcare market was worth around 11 billion U.S. dollars worldwide. It was forecast that the global healthcare AI market would be worth almost 188 billion U.S. dollars by 2030, increasing at a compound annual growth rate of 37 percent from 2022 to 2030 (Statista, 2023). The adoption of AI in healthcare, particularly in remote patient monitoring, has seen significant growth due to its ability to handle tasks ranging from physical activity classification to chronic disease monitoring and vital signs monitoring in emergency settings (Shaik et al., 2022). The advantages of AI in healthcare monitoring, such as reducing the burden on healthcare providers and minimising human errors, have made it a major trend in the healthcare industry.

With the increasing use of AI technology in healthcare, several applications have emerged that can be useful for individuals with diabetes in monitoring their condition. One such system is the Glucose tracker - Diabetic Diary, developed by Mel Studio (Mel Studio, 2023). This mobile application is designed to help diabetes individuals keep track of their regular haemoglobin, sugar levels, and blood pressure. It includes insulin reminders to ensure timely insulin administration. It offers more than 7 charts to analyse the input data and allows the users to export the records for external use such as sharing the records with their doctors. Detailed graphical well-being analysis is provided to facilitate user understanding. Another competitor is the mySugr application (mySugr, 2023). This application can

Figure A.5: Proposal Page 5

TARUMT TUNKU ABDUL RAHMAN UNIVERSITY OF MANAGEMENT AND TECHNOLOGY

ONG JIA HUI
RSW
016-4679618
ongjh-pm20@student.tarc.edu.my
Ts. Chau Guan Hin

automatically transfer the individuals' daily blood glucose measurement from the blood glucose metres via Bluetooth. It analyses and identifies the pattern of blood glucose, generating clear reports that can be shared with doctors. To aid the individuals understand their management progress, it provides visualisations such as graphs and trends when presenting the data recorded by the individuals. Additionally, it calculates the average blood glucose level based on their daily measurement and estimates the future HbA1c rate, which is the amount of blood glucose attached to haemoglobin, based on the past measurement.

Application	**Glucose Tracker - Diabetic Diary**	**mySugr**
Description	Application for diabetes individuals to record their daily measurement reading for self-monitoring purpose	
Similarities	1. Diabetes-related data management 2. Data visualisation 3. Medication and Insulin Reminders 4. Data Export and Sharing	
Strength	1. Detailed graphical well-being analysis in a visually appealing 2. Comprehensive tracking that includes regular haemoglobin, sugar level and blood pressure.	1. Seamless data transfer that can be performed through Bluetooth 2. Estimation of HbA1c 3. Reminders and motivation prompt
Weaknesses	1. Manual input of data measurement 2. No meditation reminder	1. Estimation of HbA1c without explanation or solution suggestion

Table 1: Competition analysis summary

Although there are numerous diabetes monitoring and management systems available in the market, many of them primarily focus on data inputting and visualisation, lacking comprehensive advice or guidance for individuals with diabetes. These applications provide limited advice or guidance to diabetes individuals. For instance, while mySugr estimates future HbA1c levels to raise individuals' awareness if their blood glucose levels are going high, it could be further improved by incorporating additional features that provide suggestions or recommendations on their daily habits to help the individuals maintain normal blood glucose levels. Therefore, the proposed diabetes monitoring system can be an alternative application that helps diabetes individuals in managing themselves. Other than those core functions of maintaining diabetes-related data, the system offers an ingredient scanning feature to help individuals in getting advice for their diet, filtering possible unsuitable products or ingredients.

The ingredient scanning feature that can provide diabetes-related information in this system sets it apart from other diabetes management systems in the market. While various ingredient scanning systems can recognise and provide information about ingredients, this system can deliver additional diabetes-related information, making it highly relevant and valuable for individuals managing diabetes.

Figure A.6: Proposal Page 6

2022/23 FORM 2: Project Proposal

ONG JIA HUI
RSW
016-4679618
ongjh-pm20@student.tarc.edu.my
Ts. Chau Guan Hin

This feature is particularly important to individuals with diabetes as they need to be very mindful of their intake which can easily bring impacts onto their health status. By utilising the image provided by the users, the system extracts information from the images and provides suggestions on the ingredient detected from the perspective of a diabetes individual. This level of specialisation sets it apart and makes it a valuable tool for individuals with diabetes who are seeking accurate and targeted information about the ingredients they consume.

From another perspective, this system is more worth using compared to other applications in the market in terms of economic value. Those existing applications require users to register themselves as members or subscribe to access all features, whereas this system is completely free. Users can utilise the system and its additional ingredient scanning feature at no cost. This is possible because the system performs the ingredient analysis using artificial intelligence, while the individuals themselves maintain diabetes-related information.

In addition to the aforementioned benefits, the system enables both diabetes individuals and their healthcare providers to easily monitor their blood glucose trends. The system allows diabetes individuals to record their daily readings or measurements within the system. The stored data will be presented back to the users in a trend graph so that the data can be observed easily. The system provides the healthcare provider convenience in accessing the records made by diabetes individuals through the system itself. Unlike many existing diabetes monitoring systems in the market that require users to manually export their data from the application and share it externally with their healthcare providers, this system streamlines the process. The healthcare provider can directly access the records without the necessity of external data transfer. This significantly enhances the efficiency of healthcare providers in monitoring the effectiveness of strategies formulated for individuals with diabetes as they can closely monitor the individuals' conditions, making timely adjustments to their treatment when necessary.

The system has the potential to make significant contributions to society by achieving the United Nations Sustainable Development Goals (SDG) 2030 (United Nations, 2023). Specifically, the system can address the third goal which has the objective of ensuring healthy lives and promoting well-being for all at all ages. Under this category, sub-target 3.4 will be focused on, aiming to reduce premature mortality from non-communicable diseases. Diabetes is one of the non-communicable diseases that is associated with a high mortality rate. By providing individuals with a diabetes monitoring system that can perform ingredient scanning, individuals can promote healthy eating habits and lifestyles with the assistance of the system. The risk of developing severe diabetes impacts such as chronic disease or complications due to improper eating habits therefore can be reduced. Additionally, as the system is free of charge, it can ensure all diabetes individuals are given opportunities to receive diabetes care, regardless of their socioeconomic status. This can help to promote equitable access to healthcare, cutting down the possibilities of comorbidities due to not receiving guidance in time.

The system can also play a great role in contributing to the healthcare industry. From the perspective of healthcare providers, the system offers convenience by allowing remote monitoring of patients through the system. This allows the healthcare provider to easily detect any abnormal trend in the patient's blood glucose level through the recorded data. They can then carry out necessary follow-up and or adjustments on the medical treatment to ensure the patients' conditions remain under control.

7

Figure A.7: Proposal Page 7

ONG JIA HUI
RSW
016-4679618
ongjh-pm20@student.tarc.edu.my
Ts. Chau Guan Hin

Consequently, healthcare centres can accommodate more patients under their supervision, increasing the chances for the individuals to get proper care. They can allocate more attention to critical situations that require extra care as the routine monitoring can be handled by the system and will show necessary updates for abnormalities. This facilitates the healthcare provider in their operations, treatment plans and timely interventions to the patients.

MILESTONES

The development of this system will be based on an agile development model. The agile model is an iterative and incremental process model that focuses on an adaptive approach. An agile model can handle changes in requirements conveniently by providing proper change management methodologies to achieve successful product delivery. The system delivery is consistently evaluated against the expected outcomes to ensure the development is on the right path (Onix, 2022). Throughout every sprint, the system developed will be tested, minimising potential risks that may develop failures of the system.

Sprint	Stage	Description	Deadlines
Sprint 1	Planning	Implement core functionality into the system.	14/7/2023
	Design		28/7/2023
	Development		11/8/2023
	Testing		18/8/2023
	Deployment		22/8/2023
	Feedback		23/8/2023
Sprint 2	Planning	Implement additional functionalities and enhance the usability of the system.	29/8/2023
	Design		15/9/2023
	Development		30/9/2023
	Testing		7/10/2023
	Deployment		11/10/2023
	Feedback		12/10/2023
Sprint 3	Planning	Complete unfinished functionalities and polish the product for official release.	20/10/2023
	Design		30/10/2023

8

Figure A.8: Proposal Page 8

2022/23 FORM 2: Project Proposal

ONG JIA HUI
RSW
016-4679618
ongjh-pm20@student.tarc.edu.my
Ts. Chau Guan Hin

	Development		13/11/2023
	Testing		27/11/2023
	Deployment		4/12/2023

Table 2: Schedule of Project Development Lifecycle

REFERENCES

Aschemann-Witzel, J., Varela, P., & Peschel, A. O. (2019). Consumers' categorization of food ingredients: Do consumers perceive them as 'clean label' producers expect? An exploration with projective mapping. Food Quality and Preference, 71, 117-128. https://doi.org/10.1016/J.FOODQUAL.2018.06.003

CDC, F. and N. I. and D. |. (2022). Food and Nutrition Insecurity and Diabetes: Understanding the Connection. 1-3. https://www.cdc.gov/diabetes/library/features/diabetes-and-food-insecurity.htm

Department of Statistics, M. (2021). Press Release Statistics on Causes of Death, Malaysia. 1, 8. https://www.dosm.gov.my/v1/index.php?r=column/cthemeByCat&cat=401&bul_id=R3VrRUhwSXZDN2k4SGN6akRhTStwQT09&menu_id=L0pheU43NWJwRWVSZklWdzQ4TlhUUT09

IDF Diabetes Atlas | Tenth Edition. (2021). IDF Diabetes Atlas. https://diabetesatlas.org/

International Diabetes Federation. (2019). IDF DIABETES ATLAS Ninth edition 2019. International Diabetes Federation. https://diabetesatlas.org/upload/resources/material/20200302_133351_IDFATLAS9e-final-web.pdf

IPH, I. for P. H., NIH, N. I. of H., & Malaysia, M. of H. (2019). National Health and Morbidity Survey (NHMS) 2019: NCDs - Non-Communicable Diseases: Risk Factors and other Health Problems. In Institute for Public Health, National Institutes of Health (NIH), Ministry of Health Malaysia (Vol. 1). http://www.iku.gov.my/nhms-2019

Jamal, A., Tharkar, S., Babaier, W. S., Alsomali, S. F., Alsulayhim, A. S., Alayuni, M. A., Aldakheel, N. A., Al-Osaimi, S. S., Alshehri, N., & Batais, M. (2021). Blood glucose monitoring and sharing amongst people with diabetes and their facilitators: Cross-sectional study of methods and practices. JMIR Diabetes, 6(4). https://doi.org/10.2196/29178

Khan, M. A. B., Hashim, M. J., King, J. K., Govender, R. D., Mustafa, H., & Kaabi, J. Al. (2020). Epidemiology of Type 2 Diabetes - Global Burden of Disease and Forecasted Trends. Journal of Epidemiology and Global Health, 10(1), 107. https://doi.org/10.2991/JEGH.K.191028.001

9

Figure A.9: Proposal Page 9

2022/23 FORM 2: Project Proposal

ONG JIA HUI
RSW
016-4679618
ongjh-pm20@student.tarc.edu.my
Ts. Chau Guan Hin

Lam, A. A., Lepe, A., Wild, S. H., Jackson, C., & Lam, A. (2021). Diabetes comorbidities in low-and middle-income countries: An umbrella review. Journal of Global Health. https://doi.org/10.7189/jogh.11.04040

Makhsous, S., Bharadwaj, M., Atkinson, B. E., Novosselov, I. V, & Mamishev, A. V. (2020). DietSensor: Automatic Dietary Intake Measurement Using Mobile 3D Scanning Sensor for Diabetic Patients. https://doi.org/10.3390/s20123380

Maneze, D., Weaver, R., Kovai, V., Salamonson, Y., Astorga, C., Yogendran, D., & Everett, B. (2019). "Some say no, some say yes": Receiving inconsistent or insufficient information from healthcare professionals and consequences for diabetes self-management: A qualitative study in patients with Type 2 Diabetes. Diabetes Research and Clinical Practice, 156. https://doi.org/10.1016/J.DIABRES.2019.107830

Mel Studio. (2023). Diaries of health. Mel Studio. Retrieved May 17, 2023, from https://melstudio.info/en/apps/dnevniki-zdorovya

mySugr. (2023). mySugr + iPDM | mySugr Global | mySugr. MySugr. Retrieved May 17, 2023, from https://www.mysugr.com/en/ipdm/

NIH, N. I. of H. (2020). National Health and Morbidity Survey 2019. 20, 2-2. https://doi.org/10.18356/be4d1601-en

Onix. (2022). 7 Best Software Development Models: Which One to Choose? Onix. https://onix-systems.com/blog/7-basic-software-development-models-which-one-to-choose

Papamichou, D., Panagiotakos, D. B., & Itsiopoulos, C. (2019). Dietary patterns and management of type 2 diabetes: A systematic review of randomised clinical trials. Nutrition, Metabolism and Cardiovascular Diseases, 29(6), 531–543. https://doi.org/10.1016/j.numecd.2019.02.004

Shaik, T., Tao, | Xiaohui, Higgins, N., Li, L., Gururajan, R., Zhou, X., & Rajendra Acharya, | U. (2022). Remote patient monitoring using artificial intelligence: Current state, applications, and challenges > Artificial Intelligence Technologies > Internet of Things. Wires. https://doi.org/10.1002/widm.1485

Statista. (2023, March 16). AI in healthcare market size worldwide 2030 | Statista. Statista. https://www.statista.com/statistics/1334826/ai-in-healthcare-market-size-worldwide/

United Nations. (2023). Goal 3 | Department of Economic and Social Affairs. United Nations. Retrieved May 21, 2023, from https://sdgs.un.org/goals/goal3

10

Figure A.10: Proposal Page 10

Appendix 2: Setup Guide

System (hardware and software requirements)

To allow the system to operate smoothly, both the computer and mobile phones must have stable network connections. The mobile phones must be equipped with a working camera.

Installation

To access the system there is no installation required for the web application. To access the web application, the user may use Android Studio or Visual Studio Code and follow the steps below:

1. In the IDE, click Open from the Welcome window, or File > Open from the main IDE window.
2. Browse to the directory holding the Flutter source code files for this project
3. Click Open.
4. Run the command ‘flutter get’ in the terminal window.
5. Select the debug window (E.g. Chrome)
6. Execute the debug.

The mobile application can be retrieved and downloaded from this link below:

https://drive.google.com/file/d/1C-hJRNwSZUzZ91lPdlBIrULR790NHFQ9/view?usp=drive_link

Appendix 3: User Manual

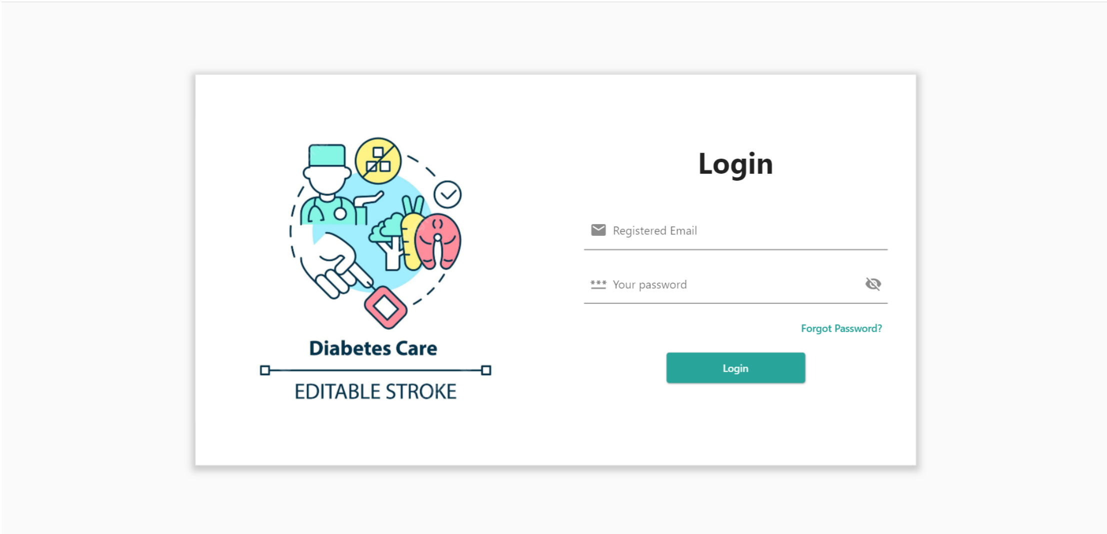


Figure C.1: Login page

On the login page, the doctor can log in to their account by using their registered email and password during account registration.

Example email and password: lim@gmail.com, password

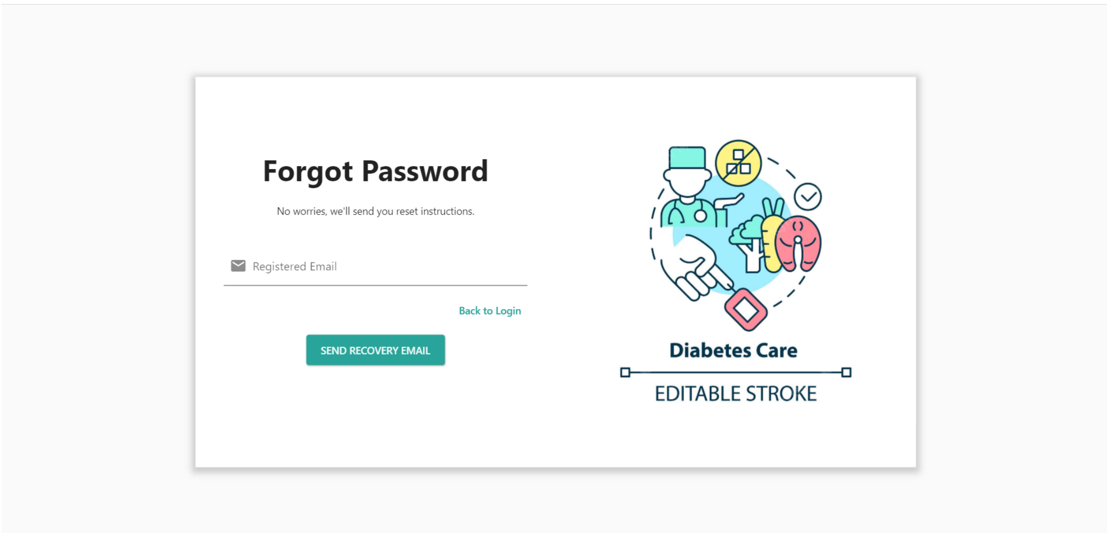


Figure C.2: Forgot Password page

If the doctor has forgotten their password, they can click on the 'Forgot Password' on the login page to initiate password recovery. The doctor needs to enter his/ her registered email address before he/ she can access the password recovery.

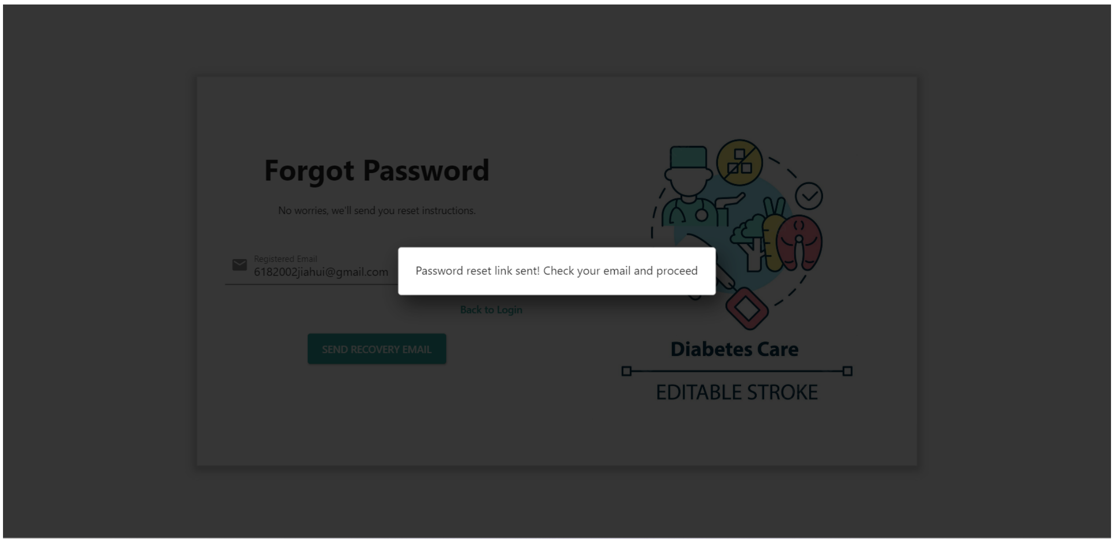


Figure C.3: Email Verification Success Page

If the email address entered is correct, the system will prompt a message, indicating the password recovery email has been sent.

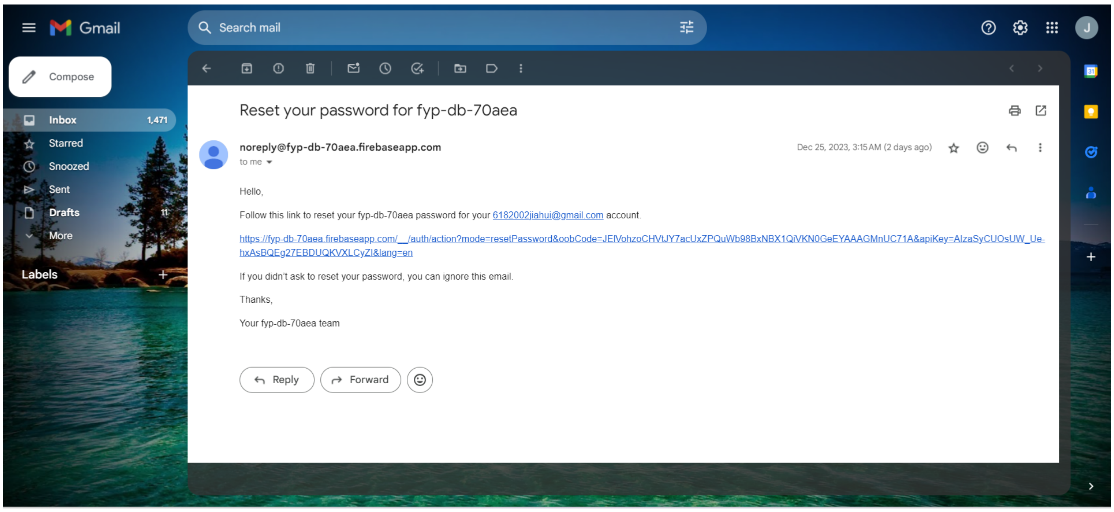


Figure C.4: Email for Password Recovery

The doctor will receive an email attached with the link for password recovery. By clicking on the link, the doctor will be navigated to a password reset page.

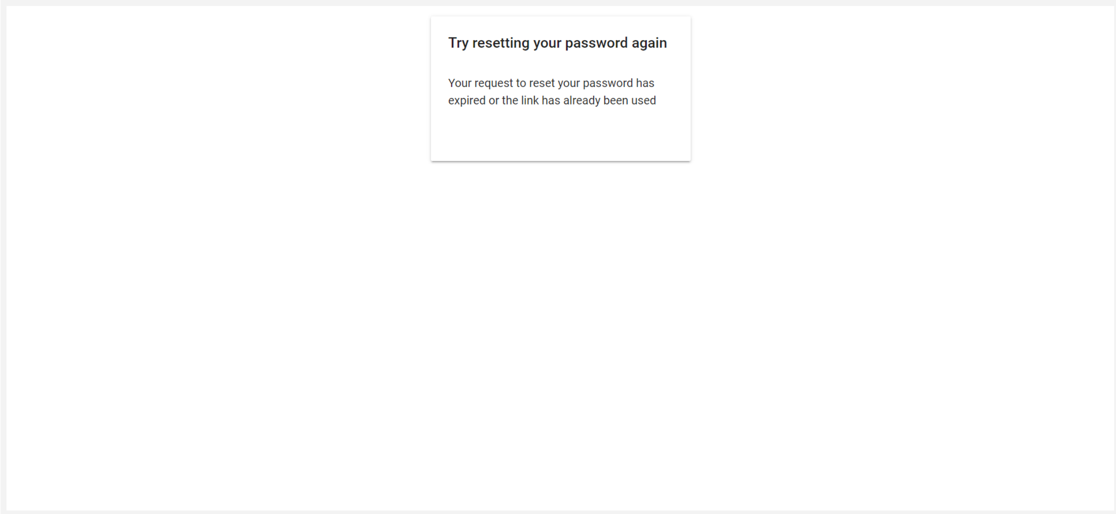


Figure C.5:Link Expired Page

If the link for password reset has expired, out of the valid time range, an error message will be displayed to notify the user.

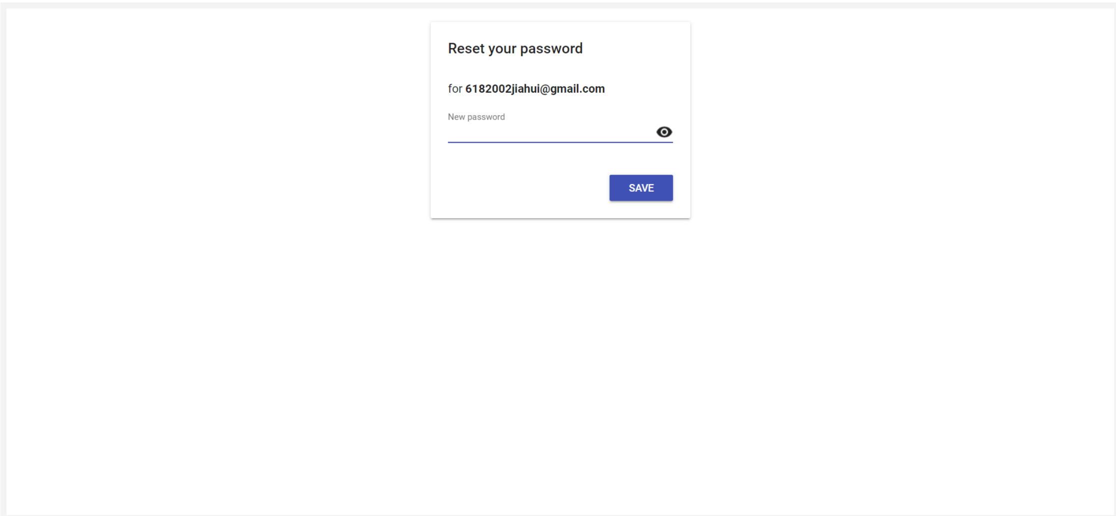


Figure C.6: Password Reset page

On this password reset page, the doctor can enter their new password and update their account credentials with the new password.

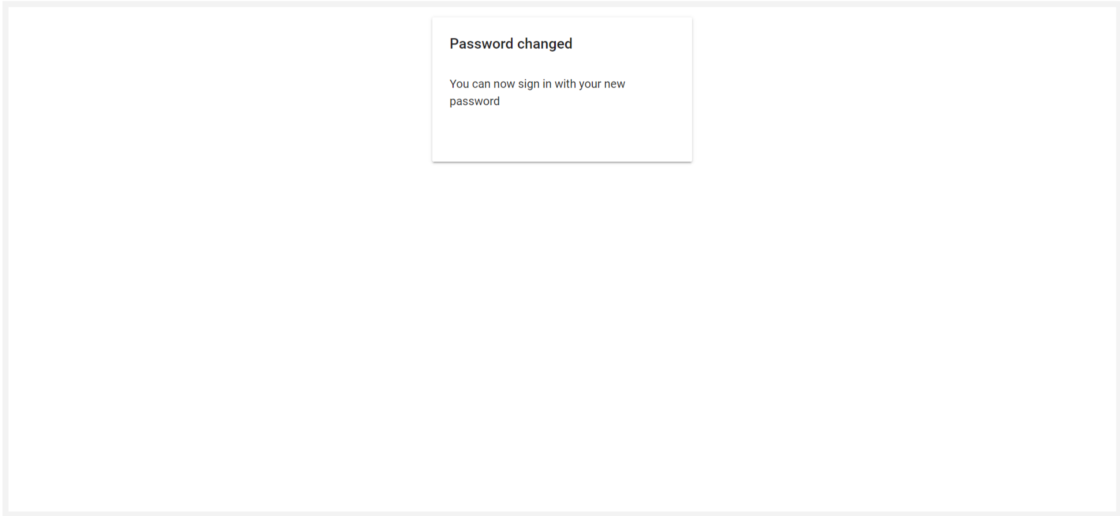


Figure C.7: Password Reset Successful page

Upon successfully resetting the password, the system will prompt the user with an action completion message.

Figure C.8: Doctor’s homepage

After the doctor successfully logs into the system, the system will redirect the user to the system homepage. On the homepage, it will provide some information such as total active doctors, active patients and total count of appointments to the doctors.

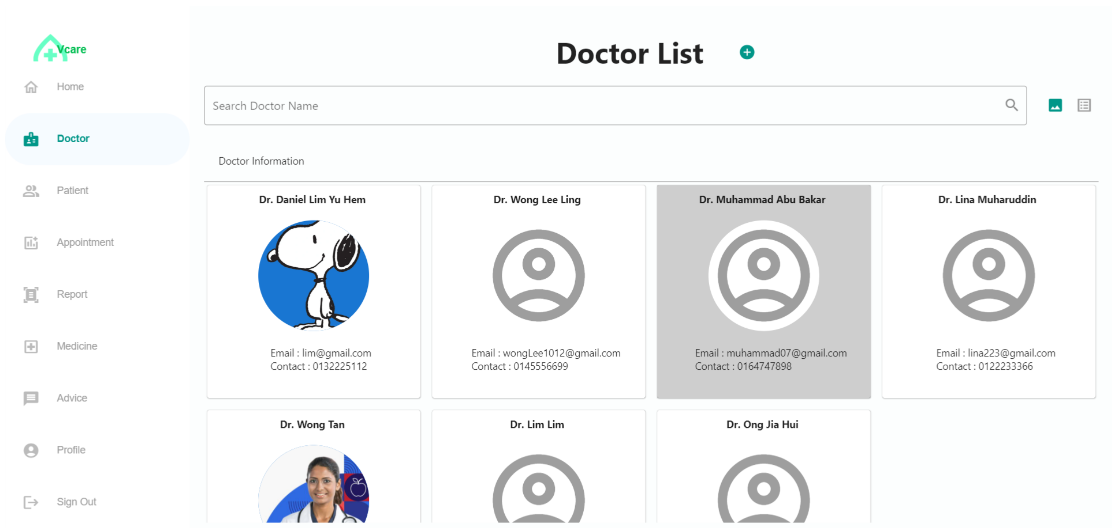


Figure C.9: Doctor Management page (Detailed View)

The doctor can navigate to the doctor management page by selecting the 'doctor' option from the navigation sidebar. The system by default displays the doctor's information in a detailed view, including the profile picture of the doctor.

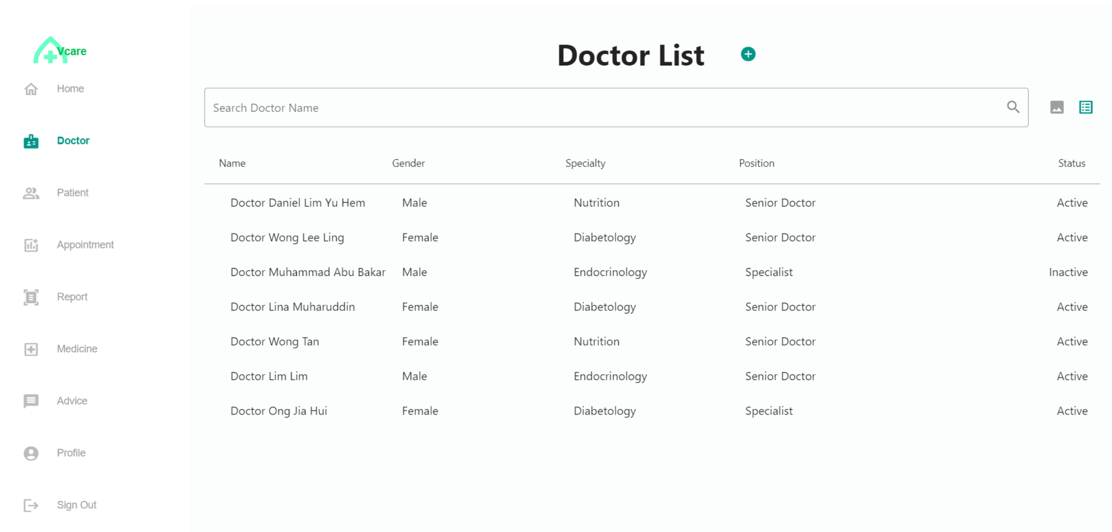


Figure C.10: Doctor Management page (List View)

The user can change the display format into a list view by selecting another display option. Within this list view, the doctor can sort the doctors' information by selecting the sorting criteria.

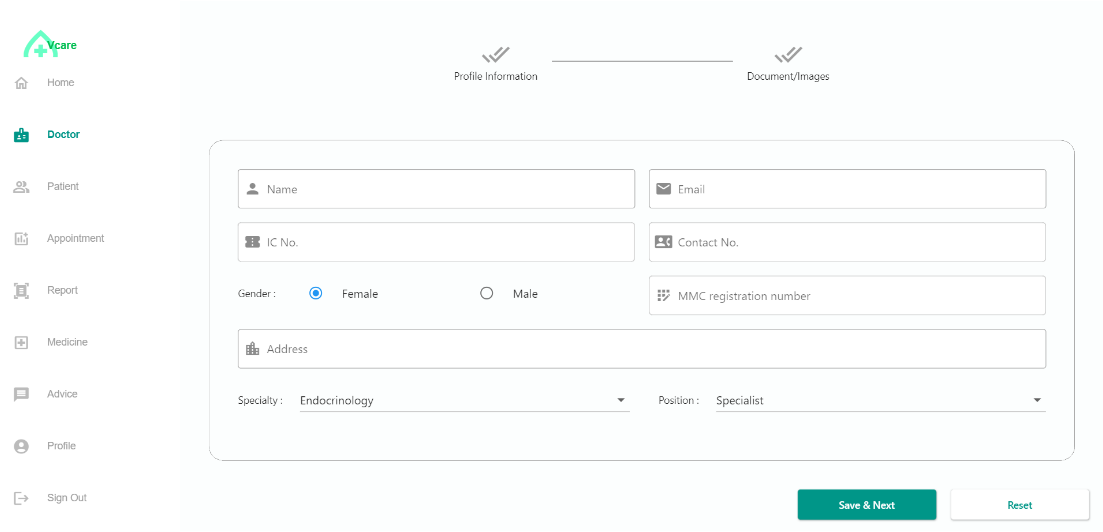


Figure C.11: Add New Doctor Profile page

By selecting the add button on the doctor management page, the doctor will be navigated to the add doctor form. After inputting every field in the page, the doctor can click the ‘Save & Next’ button to continue to the next part. The user can reset the form by clicking on the ‘Reset’ button.

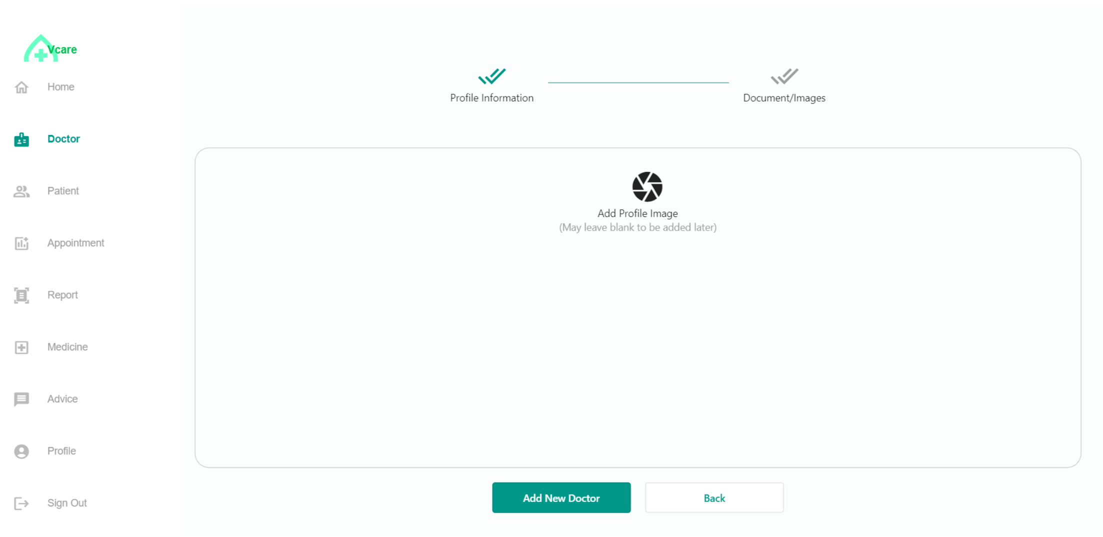


Figure C.12: Add New Doctor Image page

When the user completes the first part of a new doctor adding, the system will display the profile image adding page, allowing the user to upload the profile image. As the profile image uploading is optional, the user can directly proceed with the completion of the new doctor adding. If the user clicks on the 'Back' button, the system will navigate the user to the previous doctor profile adding form.

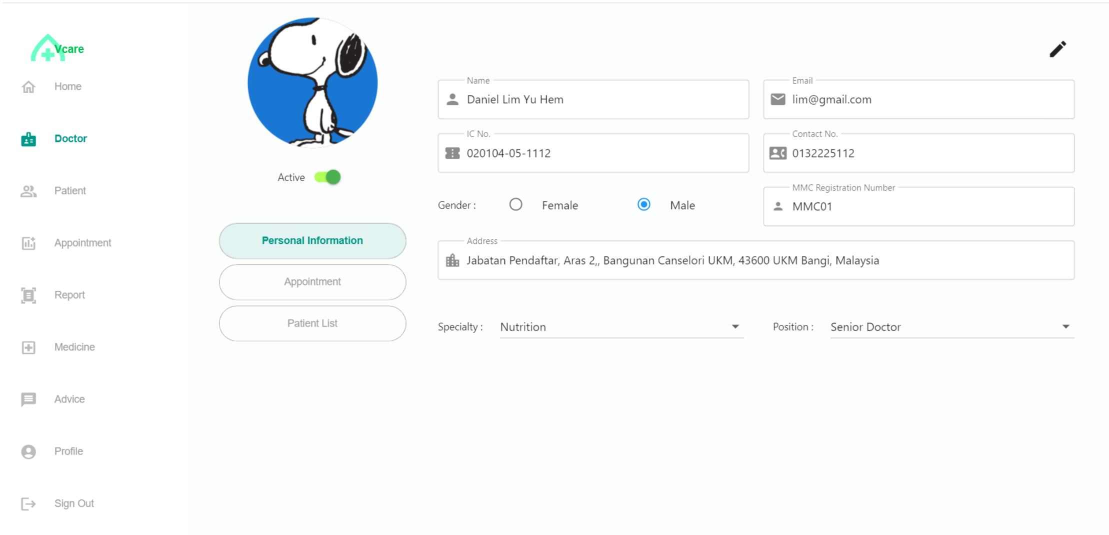


Figure C.13: View Doctor Profile page

When the user clicks on any of the doctor records from the doctor list, the system will prompt the user with the doctor's profile information.

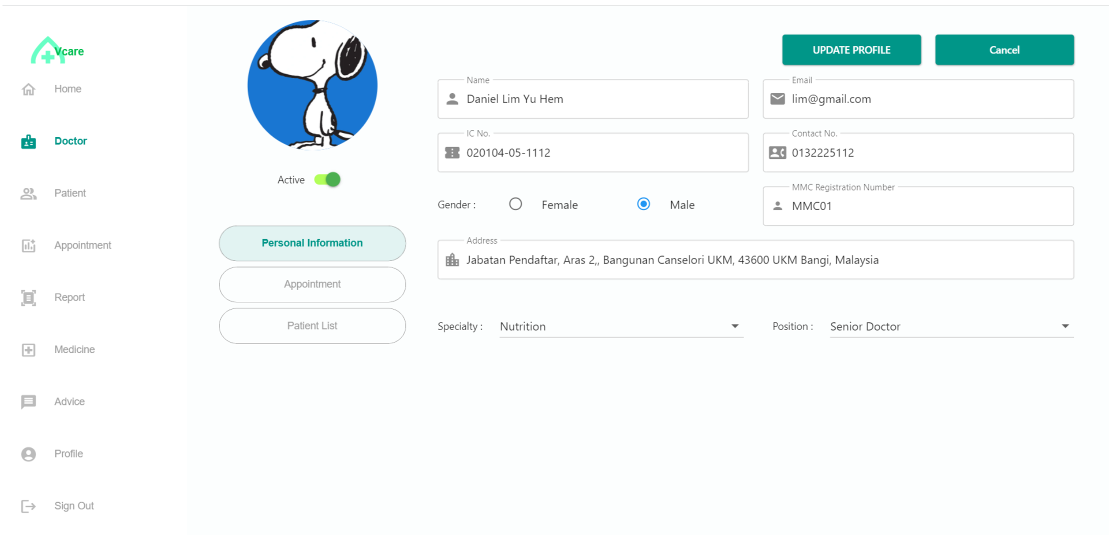


Figure C.14: Edit Doctor Profile page

The user may edit the doctor's information by activating the edit mode. The button to update the profile only appears after the edit mode is activated.

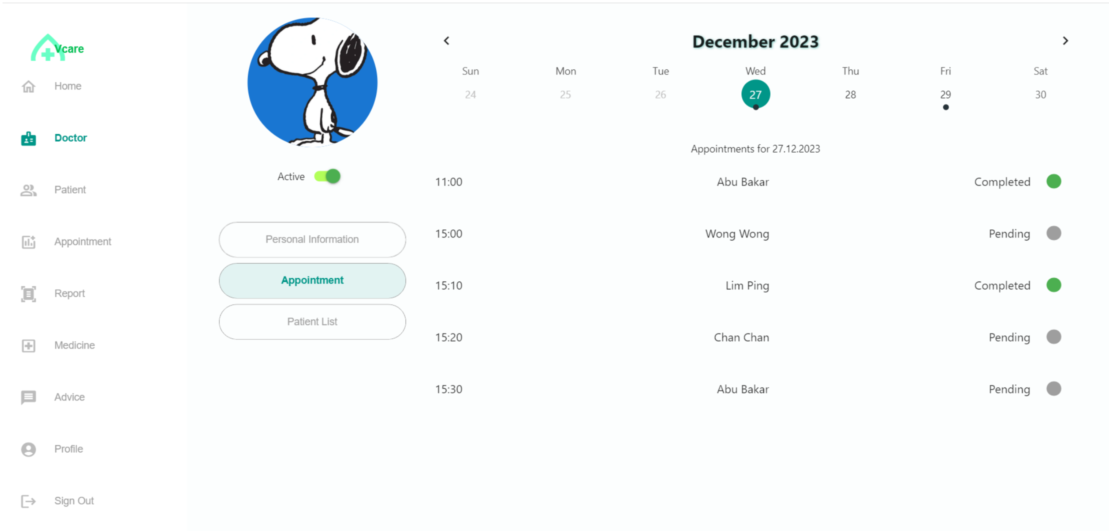


Figure C.15: View Appointment page

In the second tab of the doctor details page, the user will be displayed with the corresponding appointment.

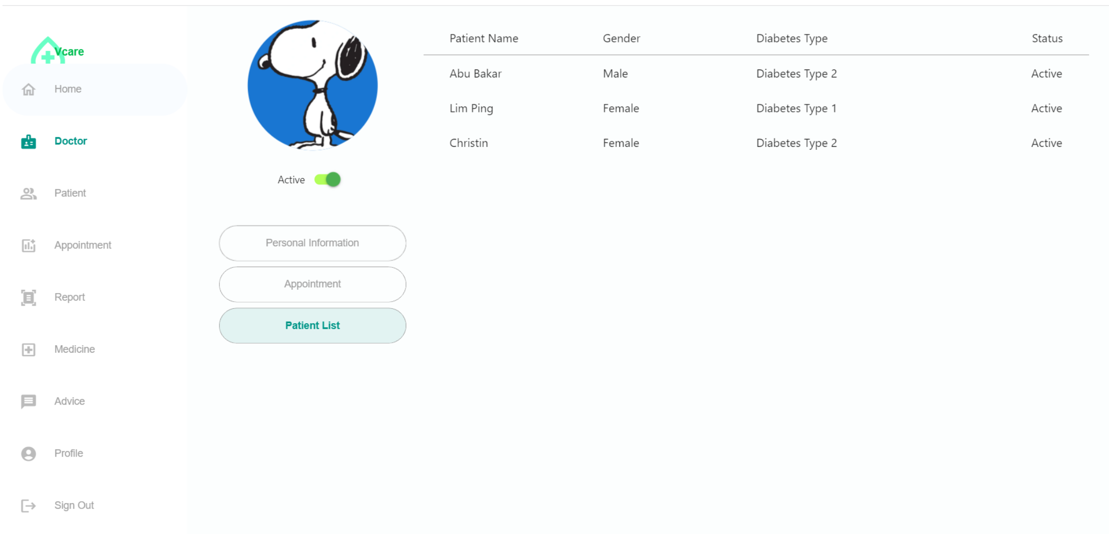


Figure C.16: View Doctor Patient List page

The third tab of the doctor details page shows the patient list in charge of the doctor.

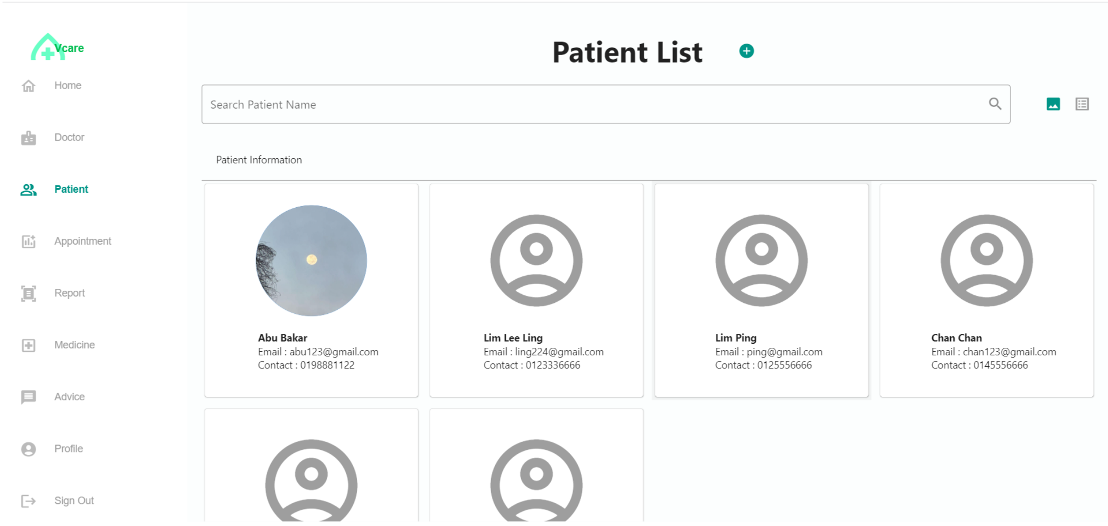


Figure C.17: View Patient Management page (Detailed View)

The doctor can navigate to the patient management page by selecting the 'patient' option from the navigation sidebar. The system by default displays the patient's information in a detailed view, including the profile picture of the patient.

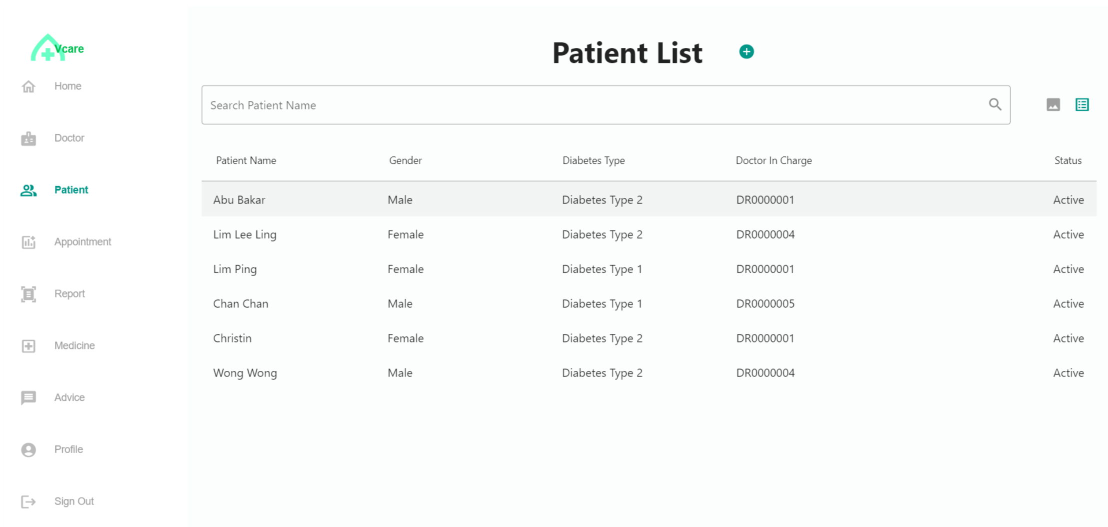


Figure C.18: View Patient Management page (List View)

The user can change the display format into a list view by selecting another display option. Within this list view, the doctor can sort the patients' information by selecting the sorting criteria.

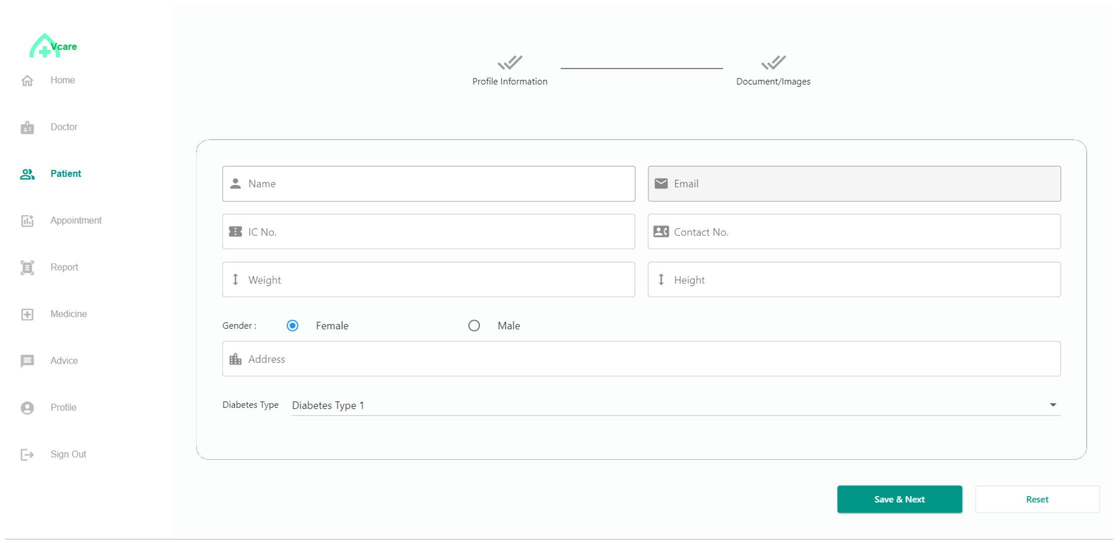


Figure C.19: Add Patient's Profile page

By selecting the add button on the patient management page, the user will be navigated to the add patient form. After inputting every field on the page, the user can click the 'Save & Next' button to continue to the next part. The user can reset the form by clicking on the 'Reset' button.

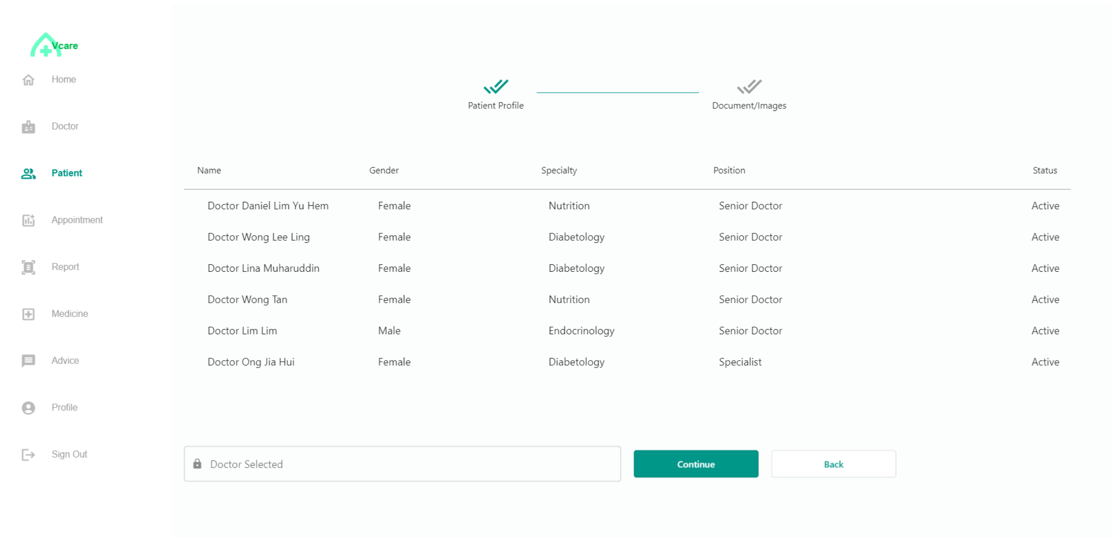


Figure C.20: Add Patient's Doctor page

The user will be prompted to this doctor assignment page to allow the assignment of a doctor to the patients currently added. By clicking on the 'Back' button, the system will navigate the user back to the patient profile adding page.

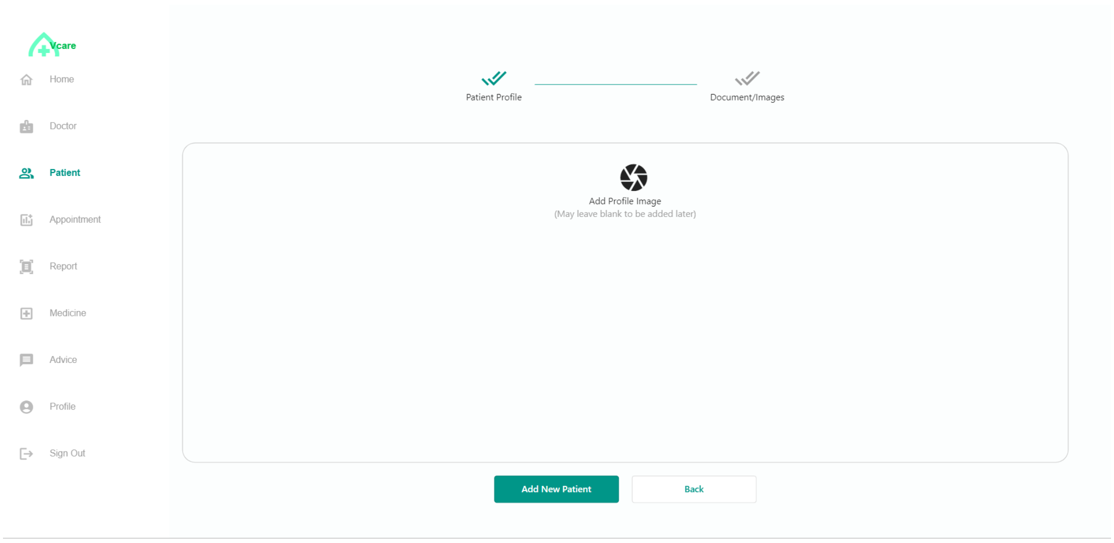


Figure C.21: Add Patient's Image page

When the user completes the first part of a new doctor adding, the system will display the profile image adding page, allowing the user to upload the profile image. As the profile image uploading is optional, the user can directly proceed with the completion of the new patient adding. If the user clicks on the 'Back' button, the system will navigate the user to the previous doctor assignment page.

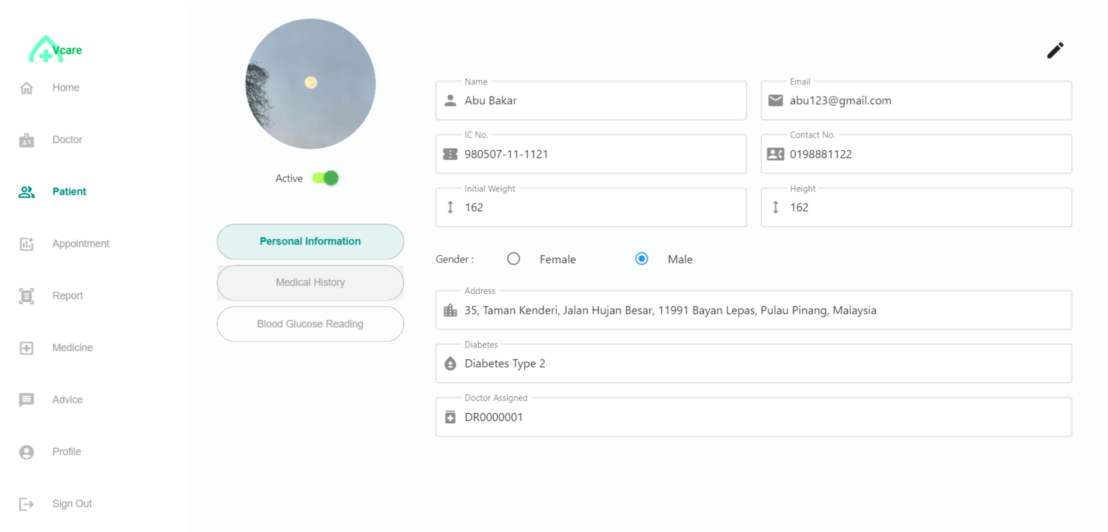


Figure C.22: View Patient Profile page

When the user clicks on any of the patient records from the patient list, the system will prompt the user with the patient's profile information.

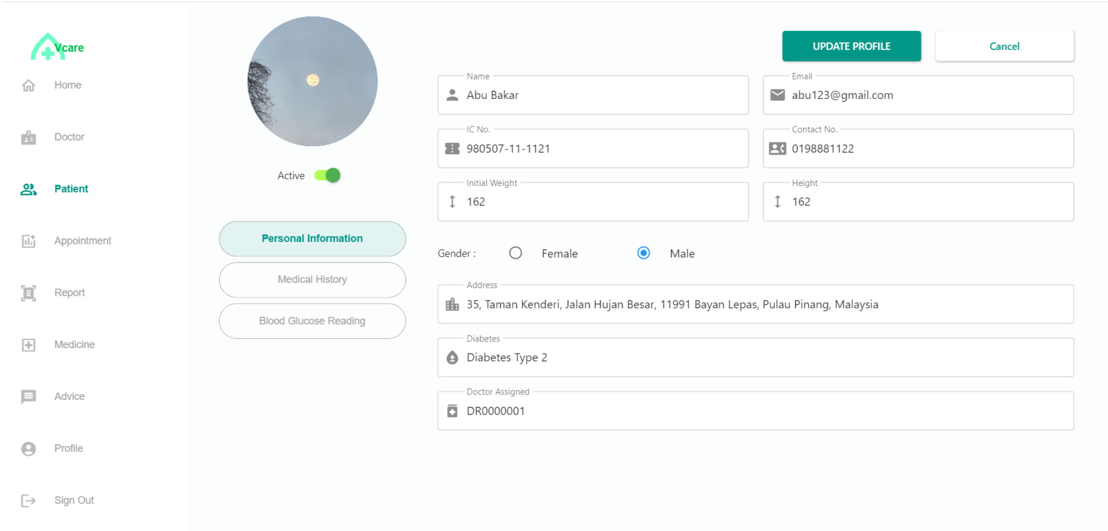


Figure C.23: Edit Patient Profile page

The user may edit the patient's information by activating the edit mode. The button to update the profile only appears after the edit mode is activated.

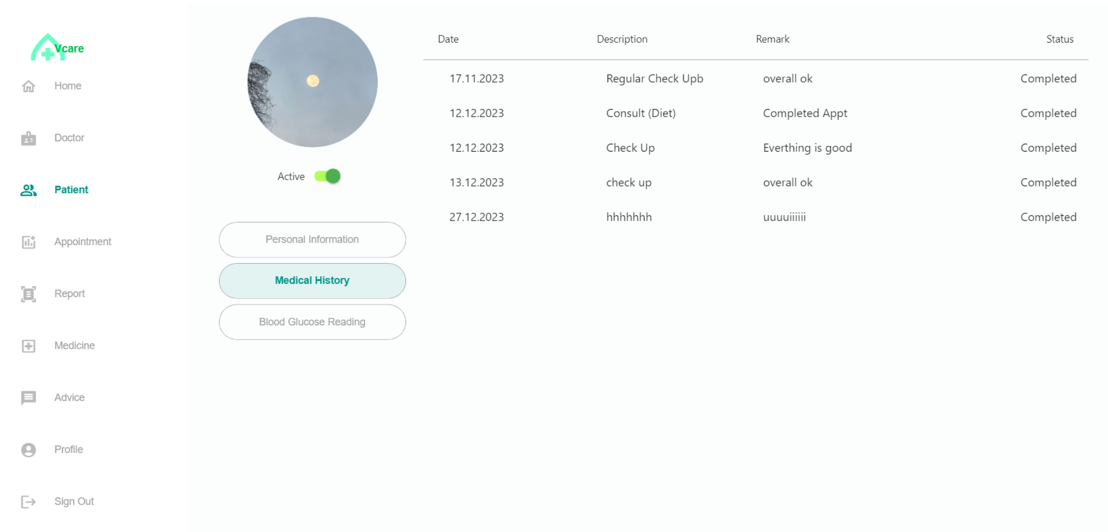


Figure C.24: View Patient Profile page

In the second tab of the patient details page, the user will be displayed with the corresponding medical history records.

Figure C.25: View Patient Blood Glucose page

The third tab of the patient details page shows the blood glucose reading trend recorded by the patient.

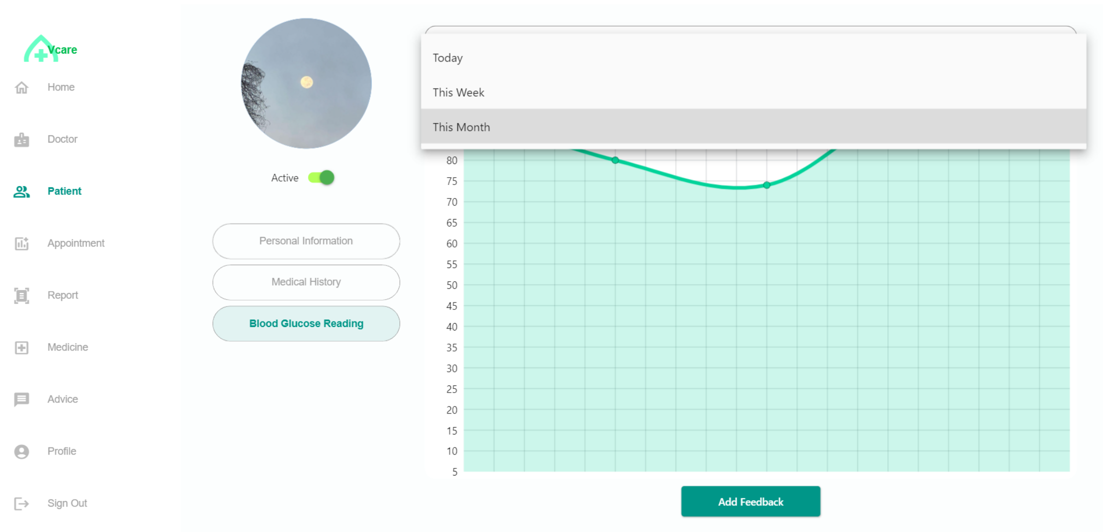


Figure C.26: Change Patient Blood Glucose Duration page

The user can select the duration of the blood glucose reading trend to make the necessary filters on the reading values.

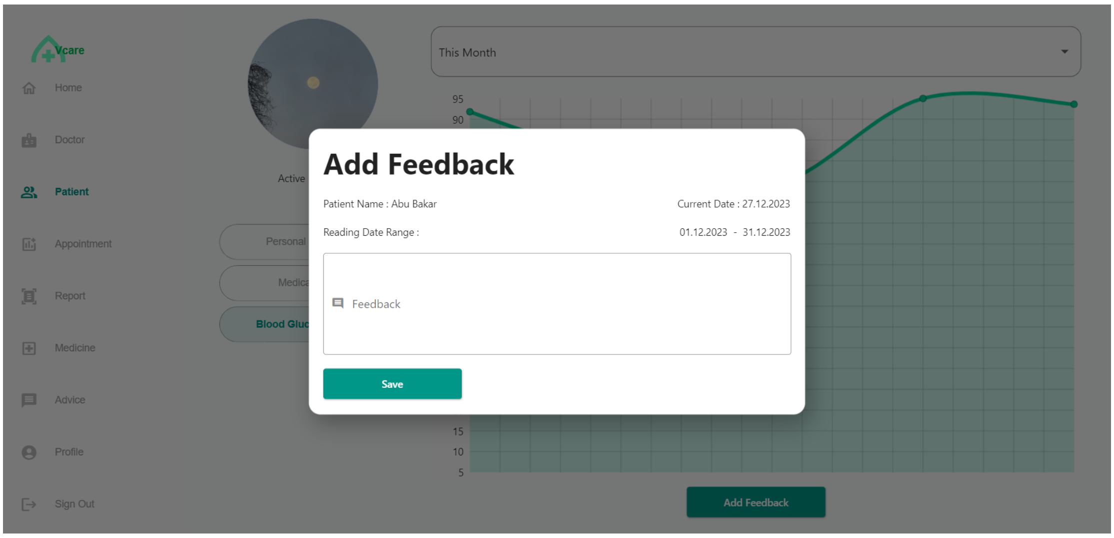


Figure C.27: Add Feedback page

The user can add new feedback by clicking the ‘Add Feedback’ button. The user can enter necessary feedback to the patients through the column provided.

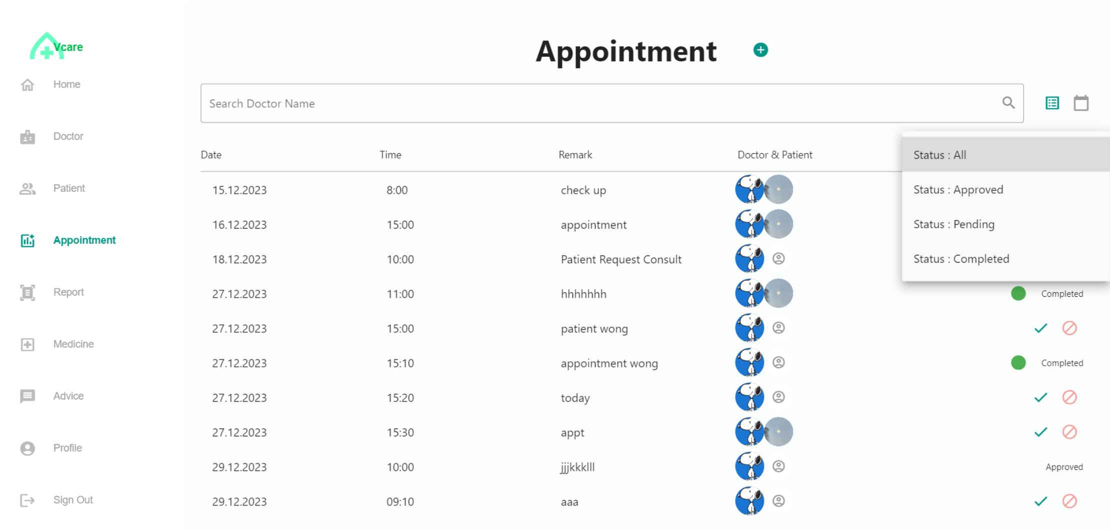


Figure C.28: Appointment Management page

By selecting the ‘Appointment’ option from the navigation sidebar, the system will present the appointment management page. The user can perform a filter onto those records by selecting the necessary appointment status to view.

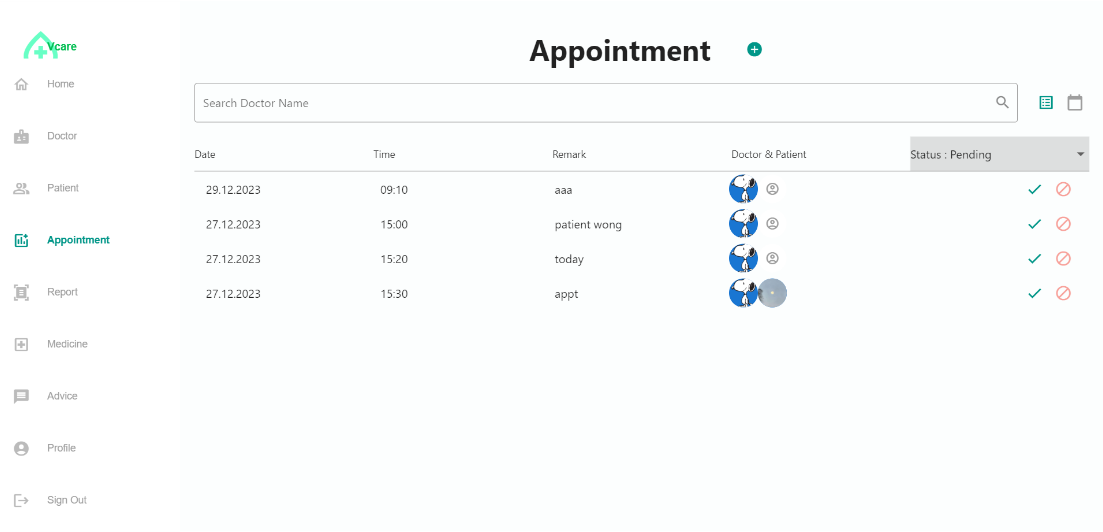


Figure C.29: Appointment Management page (Status: Pending)

For the appointment under the status pending, the user can perform necessary actions such as approve or reject.

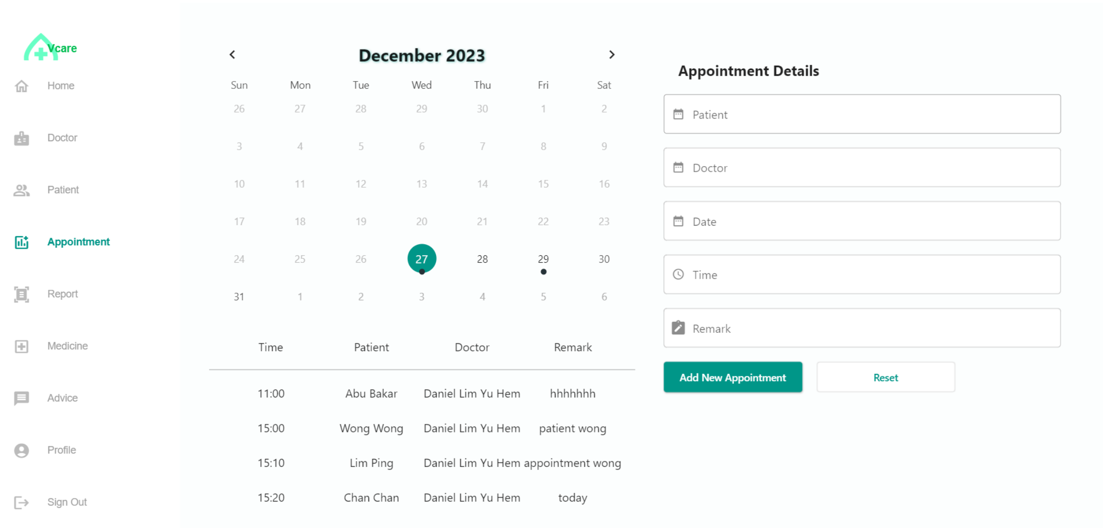


Figure C.30: Add Appointment page

When the user opts to add a new appointment, the system will present the user with an appointment-adding form. The user then can add in necessary appointment information to create a new appointment.

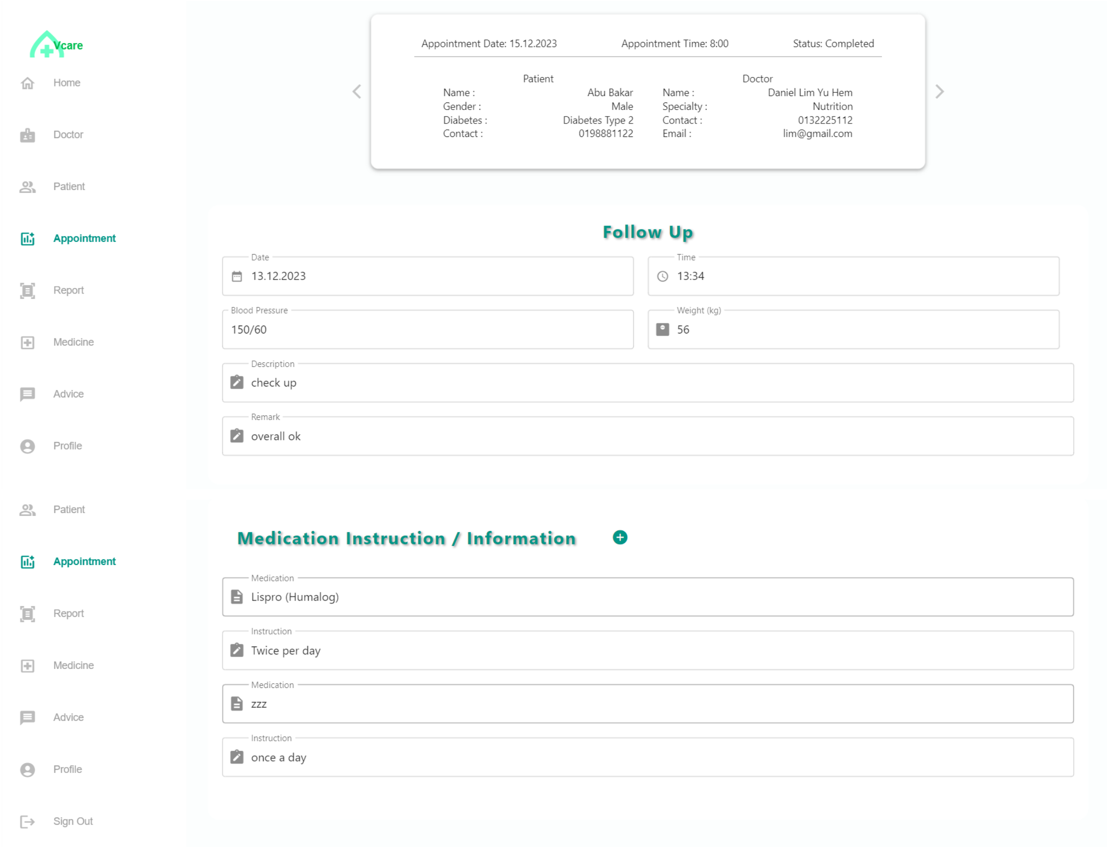


Figure C.31: Update Appointment page

When the user selects any record from the appointment list, the system will prompt the user to an appointment updating page. On this page, the user can input the follow-up and medication information.

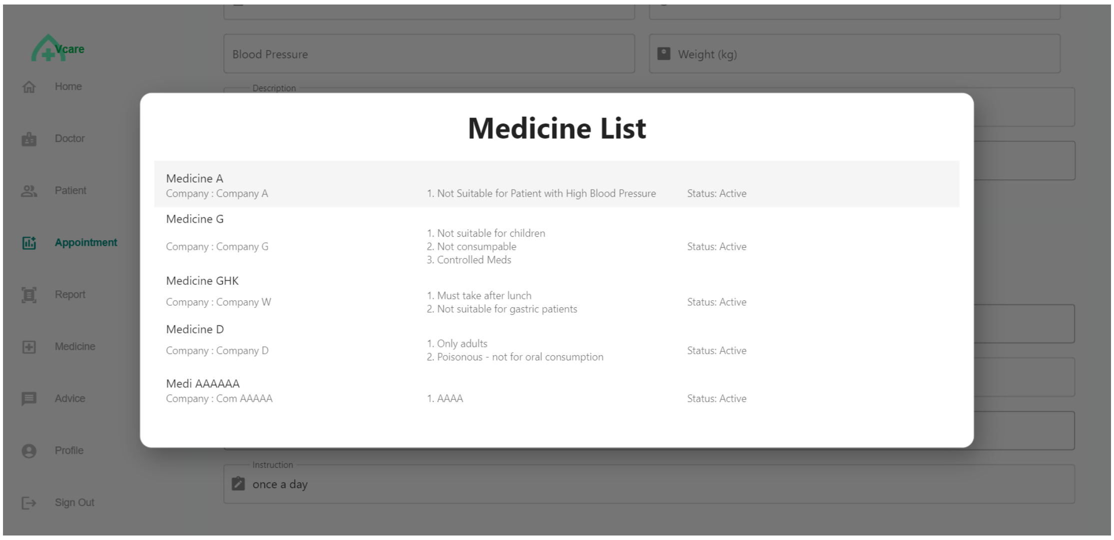


Figure C.32: Medicine List page

In this appointment update page, the system will prompt a list of medicines for selection when the user attempts to fill in medicine information.

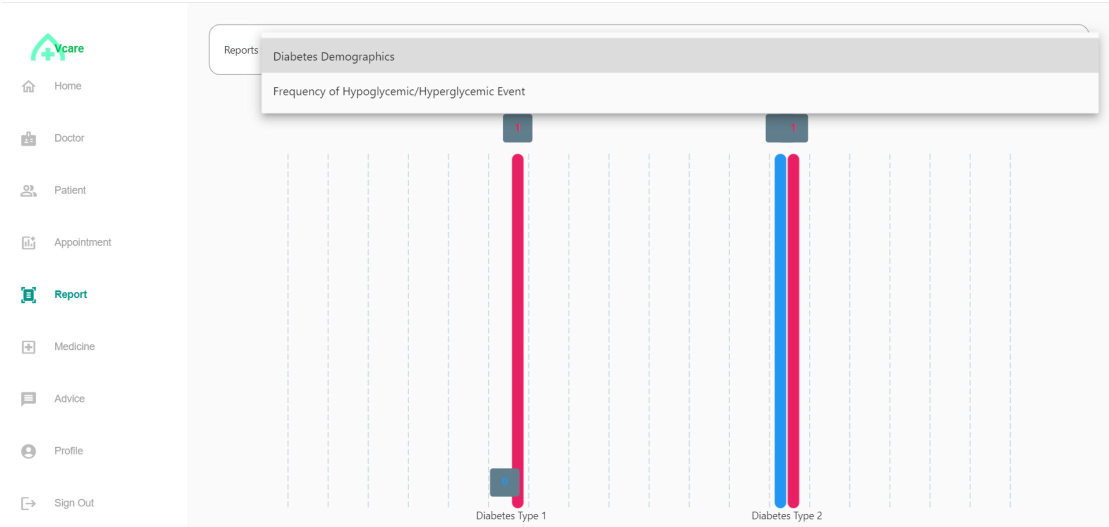


Figure C.33: Report I page

Selecting the 'report' option from the navigation sidebar will display the report page of the system. The user can select different types of reports by changing the report option.

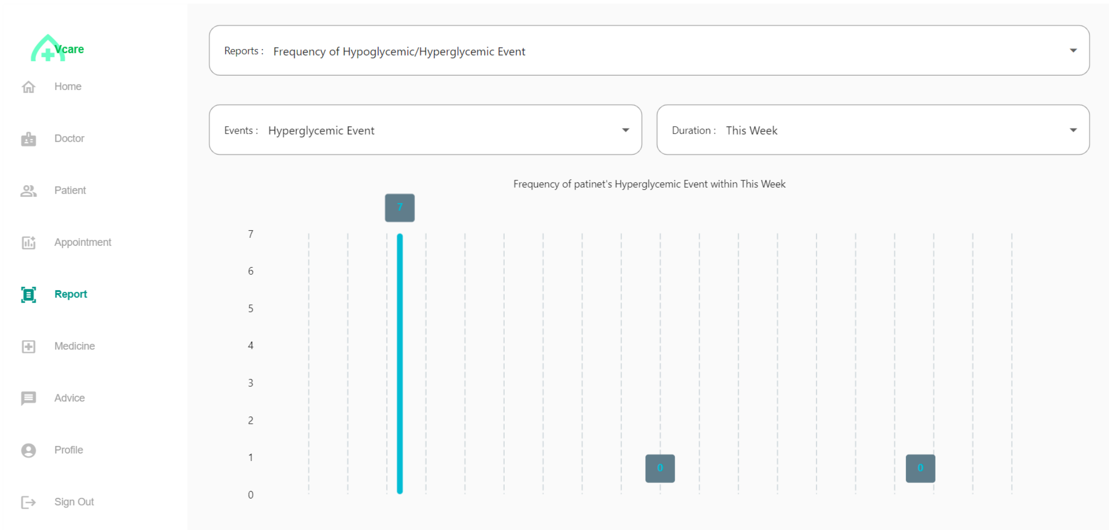


Figure C.34: Report II page

For the report ‘frequency of hypoglycemic and hyperglycemic events’, the user can select the type of events and the duration for this report.

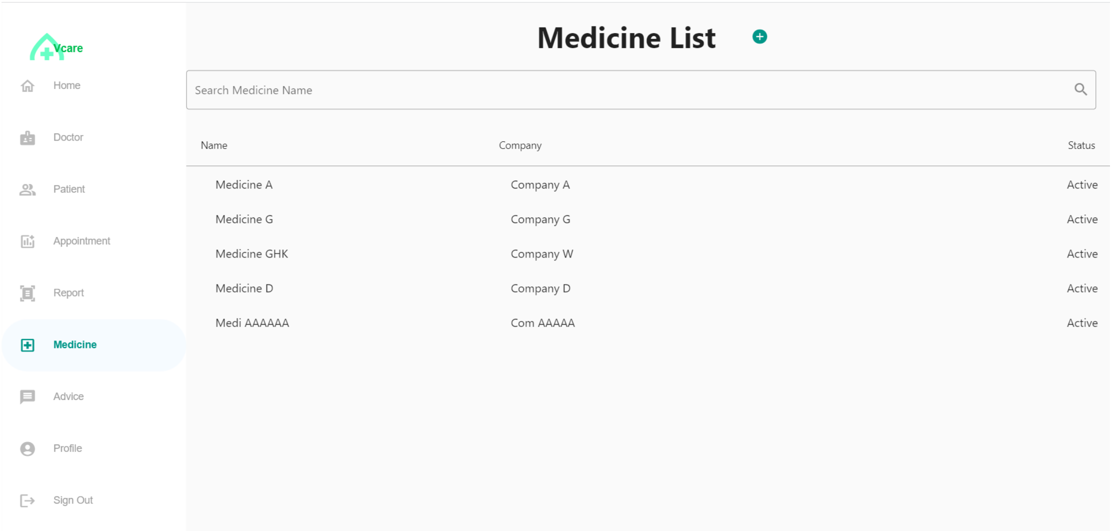


Figure C.35: Medicine Management page

The medicine management page will be displayed when the user opts for ‘medicine’. On this page, the information on medicines will be displayed accordingly.

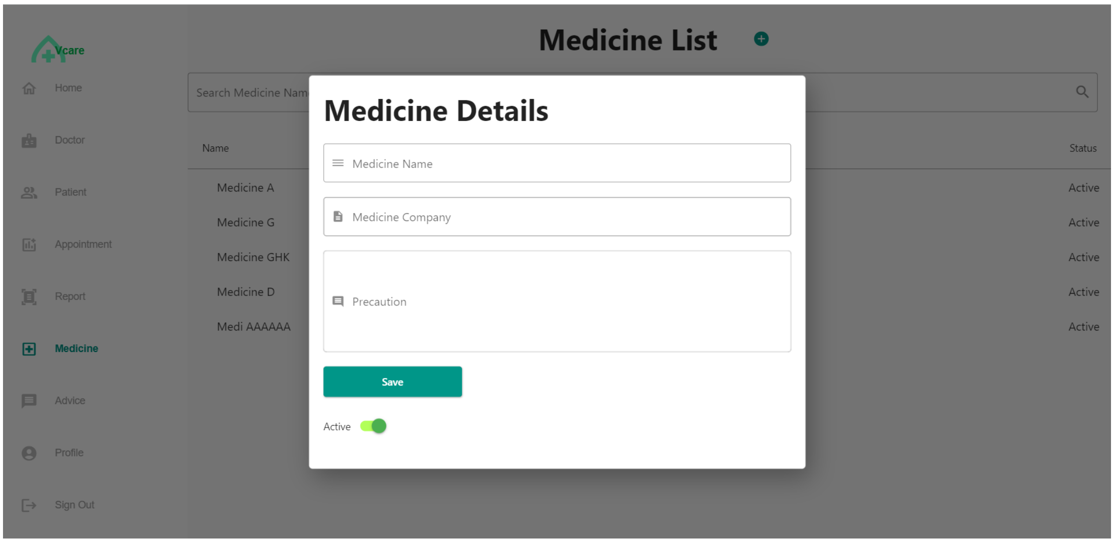


Figure C.36: Add Medicine page

The user can add a new medicine by clicking on the ‘add new medicine’ button. The user can enter the new medicine information into the pop-up dialogue box.

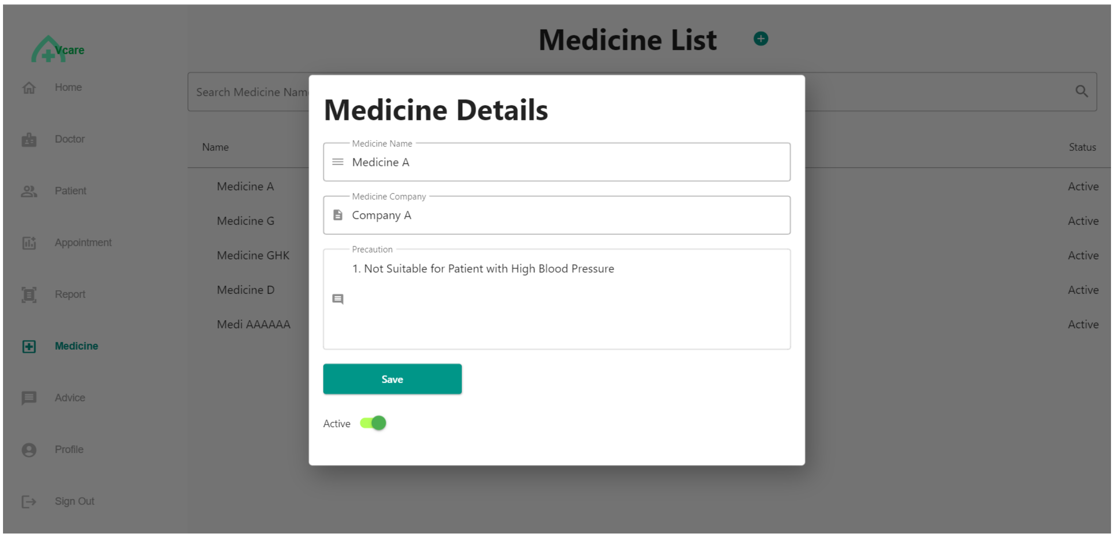


Figure C.37: Edit Medicine page

To edit the existing medicine, the user can select the medicine record to edit and make the necessary editing onto the field required.

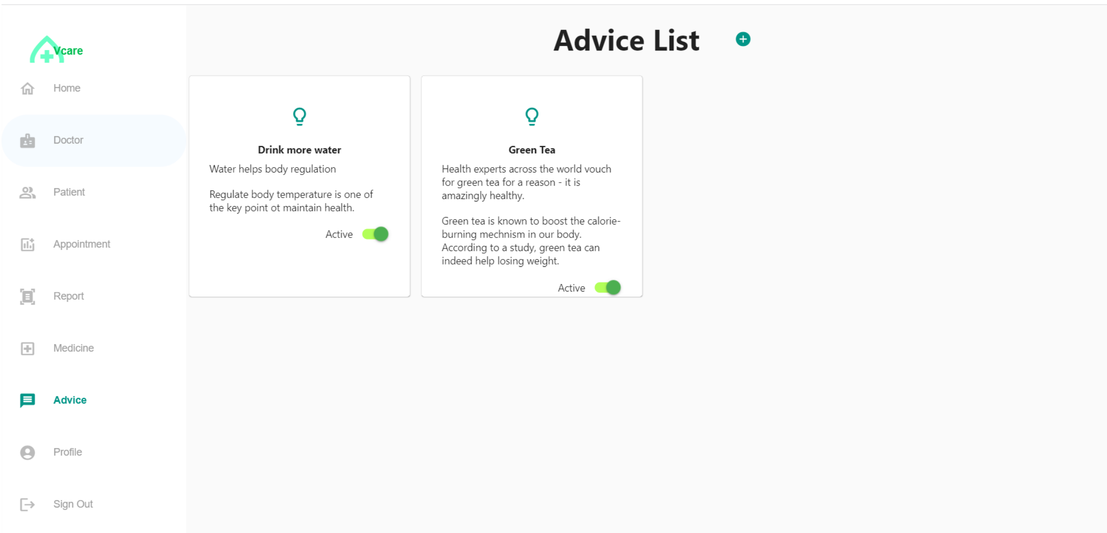


Figure C.38: Advice Management page

The advice management page will be displayed when the user opts for ‘advice’. On this page, the information of advice will be displayed accordingly.

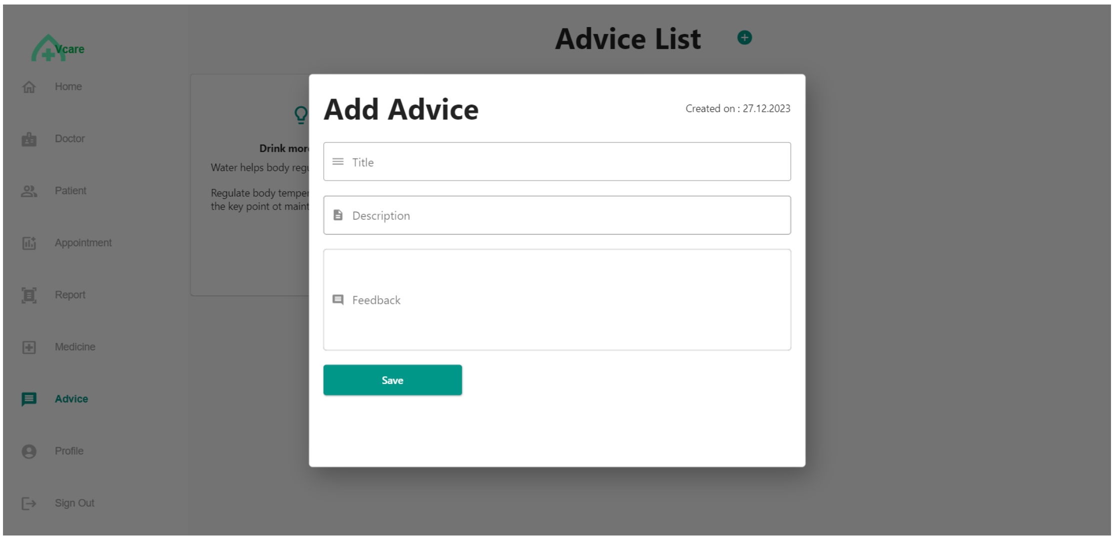


Figure C.39: Add Advice page

The user can add a new advice by clicking on the 'add new advice' button. The user can enter the new advice information into the pop-up dialogue box.

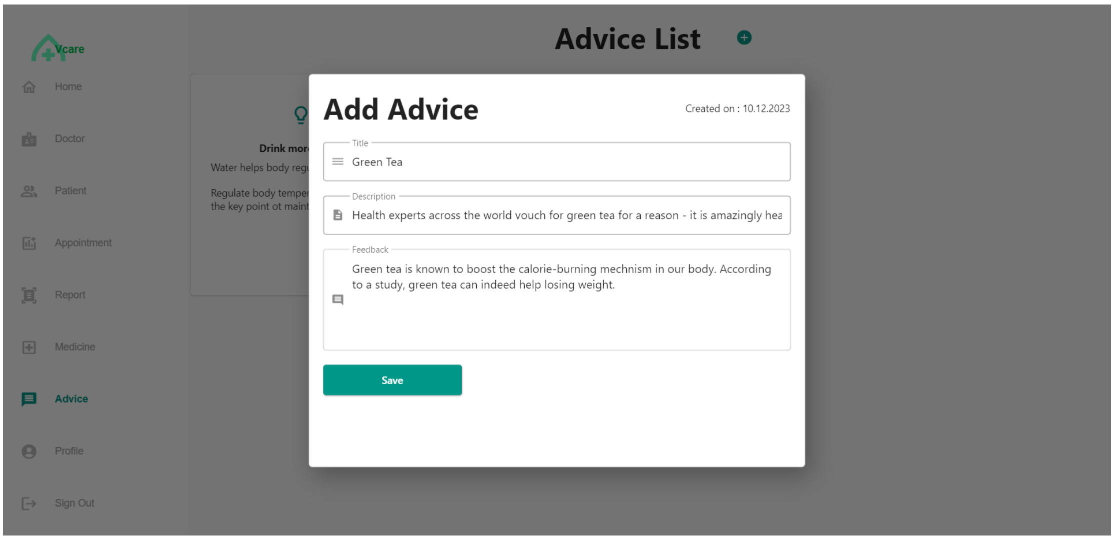


Figure C.40: Edit Advice page

To edit the existing advice, the user can select the advice record to edit and make the necessary editing onto the field required.

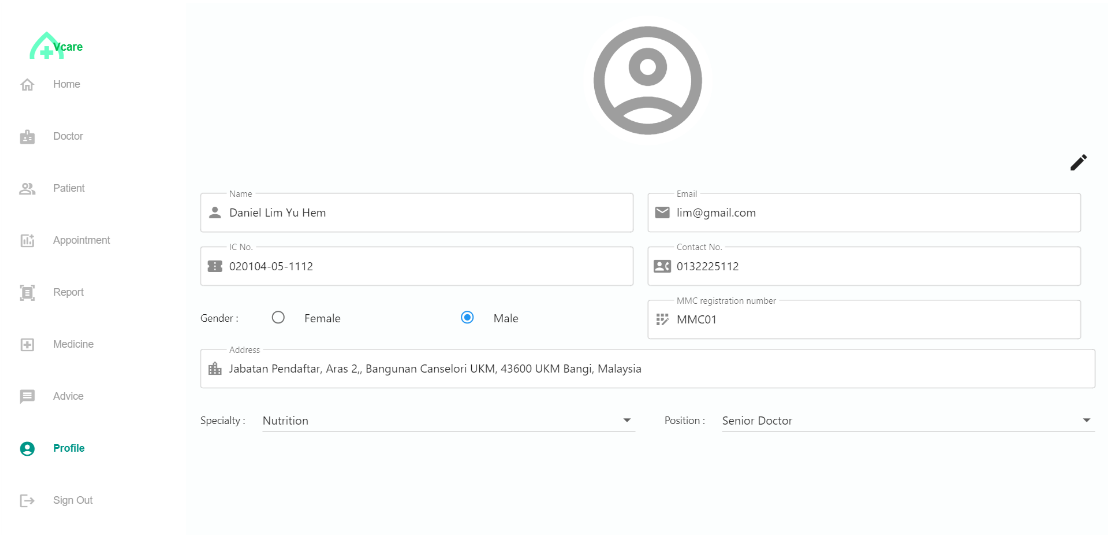


Figure C.41: Profile Management page

To view the profile information, the user can navigate to the profile page by selecting the ‘profile’ option from the navigation sidebar.

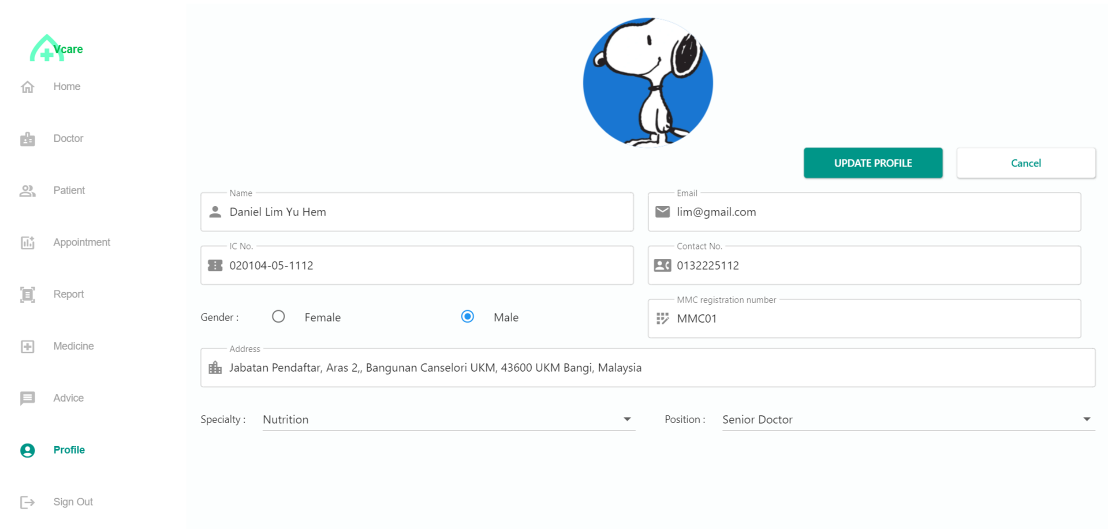


Figure C.42: Edit Profile page

The user may edit their profile information after activating the edit mode. The button to update the profile only appears after the edit mode is activated.

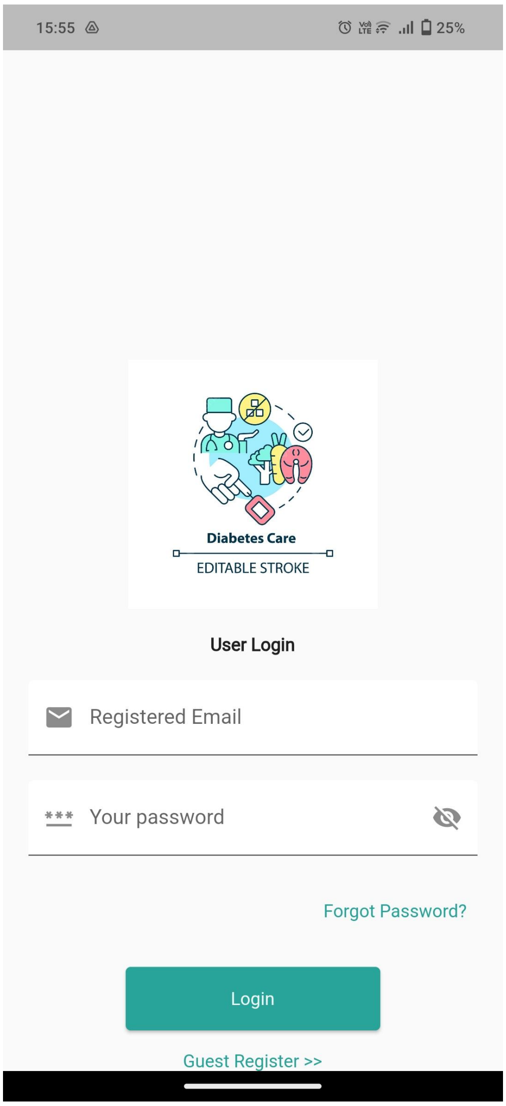


Figure C.43: Login page

On the login page, the patient can log in to their account by using their registered email and password during account registration.

Example email and password: abu123@gmail.com, password

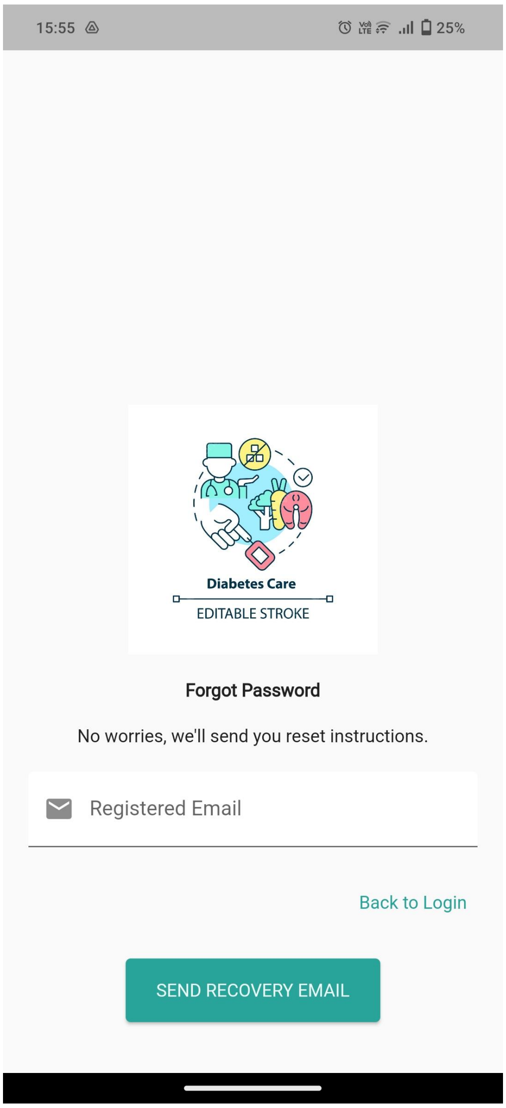


Figure C.44: Forgot Password page

If the patient has forgotten their password, they can click on the 'Forgot Password' on the login page to initiate password recovery. The patient needs to enter his/ her registered email address before he/ she can access the password recovery.

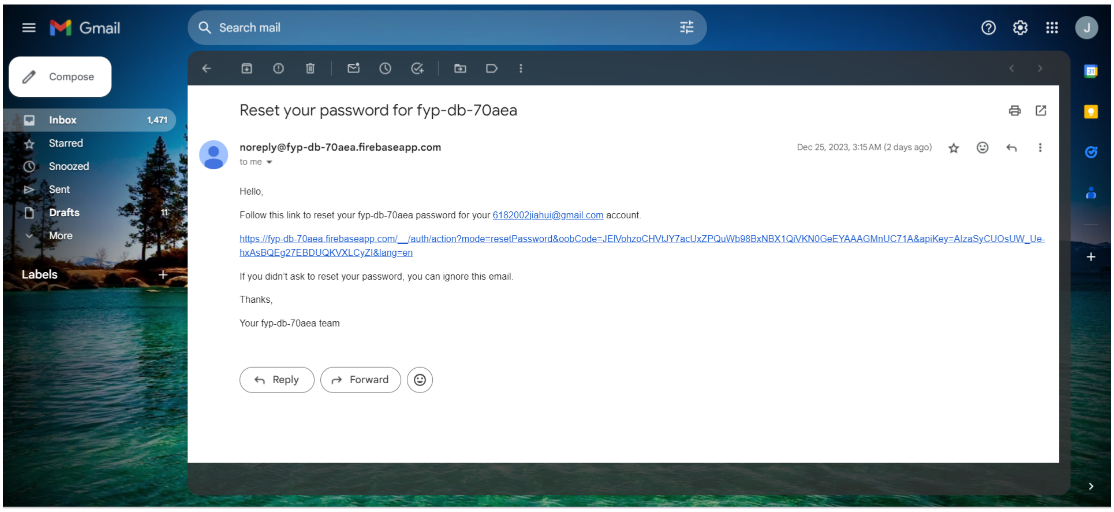


Figure C.45: Email for Password Recovery page

When the patient completes the forgot password action, he/ she will receive an email attached with the link for password recovery. By clicking on the link, the patient will be navigated to a password reset page.

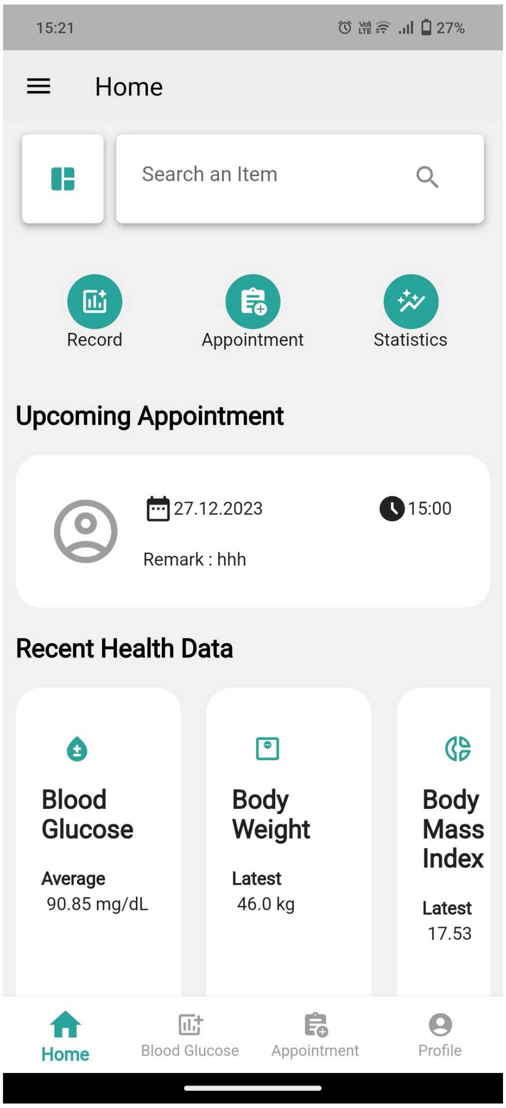


Figure C.46: Email for Homepage

After the user successfully logs in, they will be navigated to the homepage. In the system homepage, the user can navigate to the blood glucose main page, appointment main page, statistics or even the ingredient scanning page via the shortcut provided.

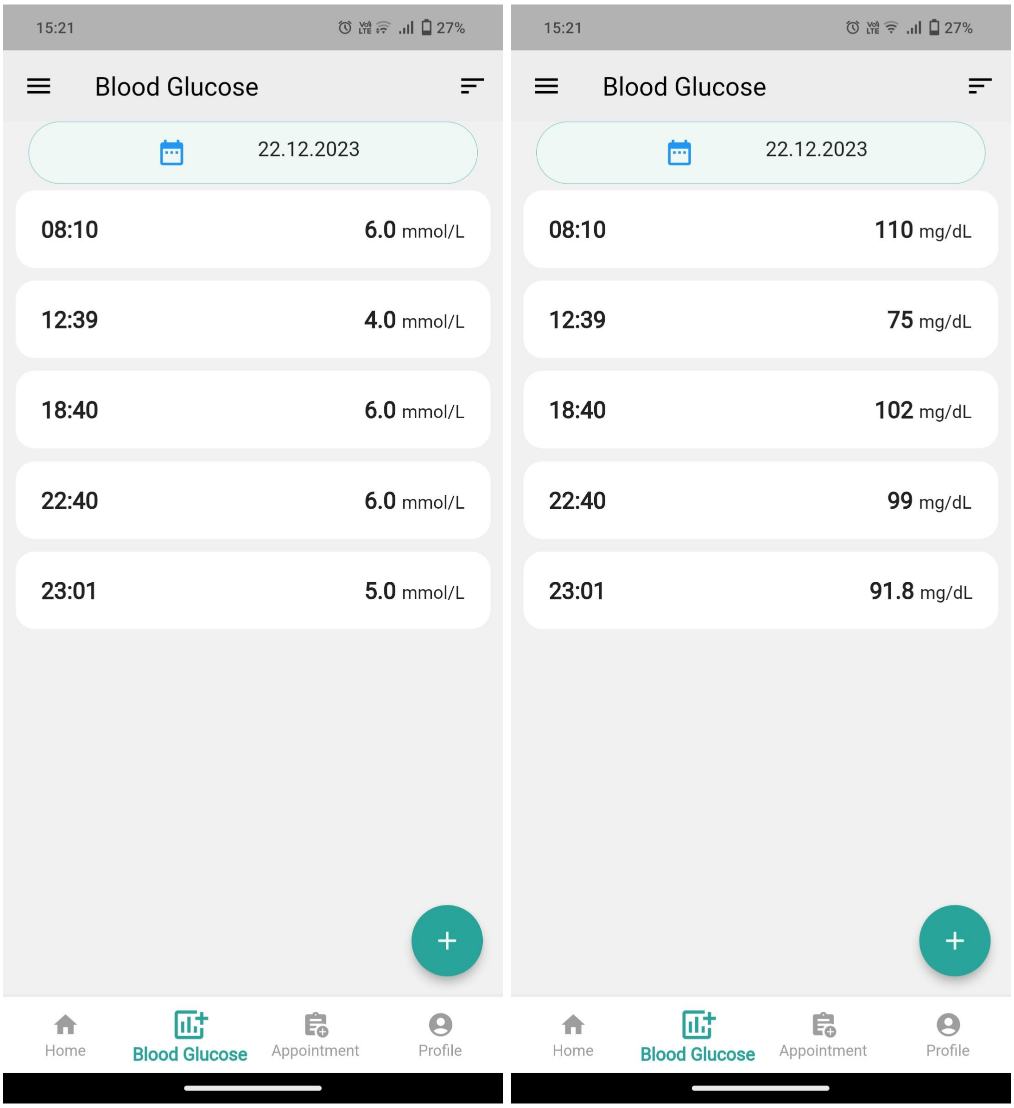


Figure C.47: Blood Glucose Management page

By selecting the ‘blood glucose’ option from the bottom navigation, the system will present the user with a blood glucose management page. On this page, the user can check all the records that were made previously by navigating the date.

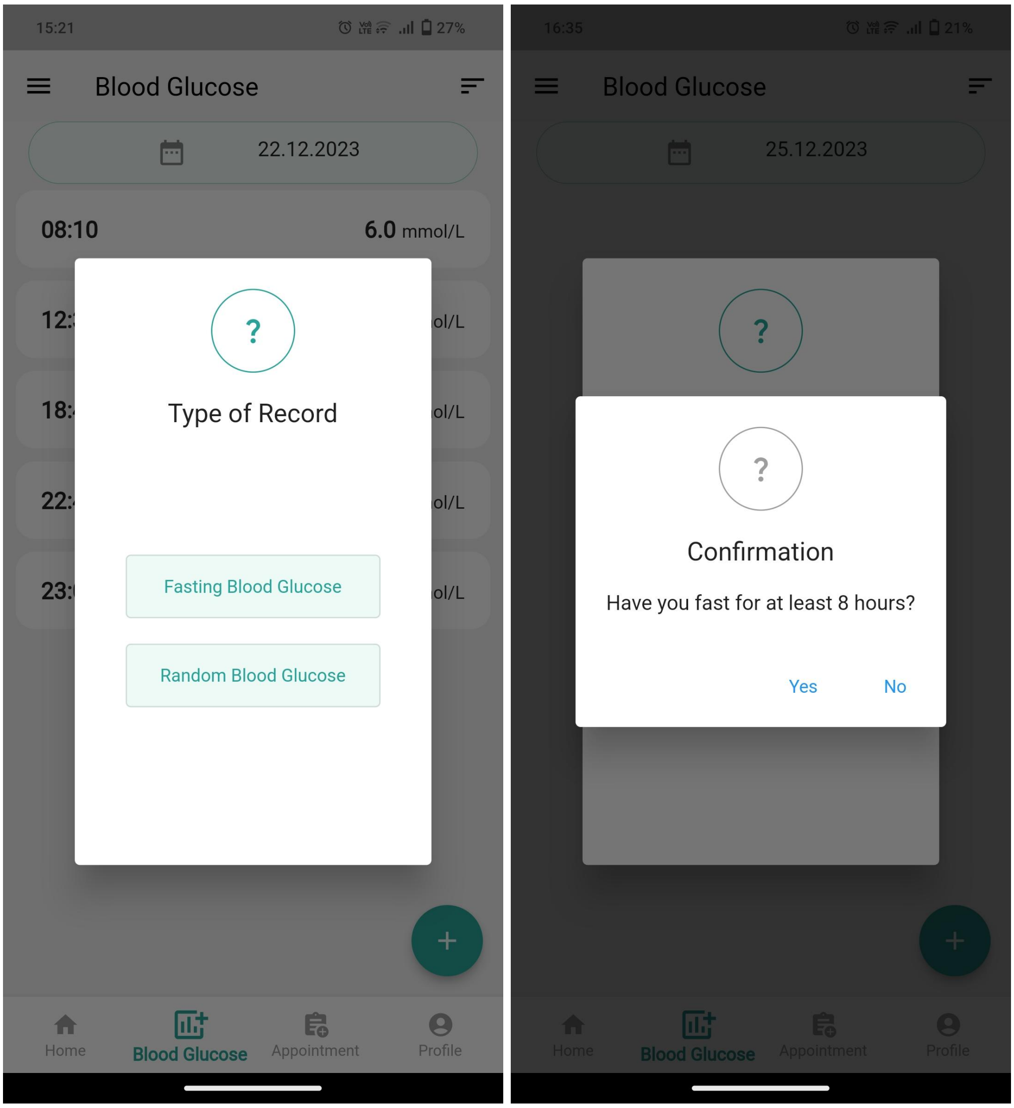


Figure C.48: Blood Glucose Type page

When the user attempts to add a new record, they have to first choose the type of record. If they select the fasting blood glucose, a confirmation of the fasting of at least 8 hours will be prompted. The user will only be directed to the add record page if they have obeyed the fasting requirement.

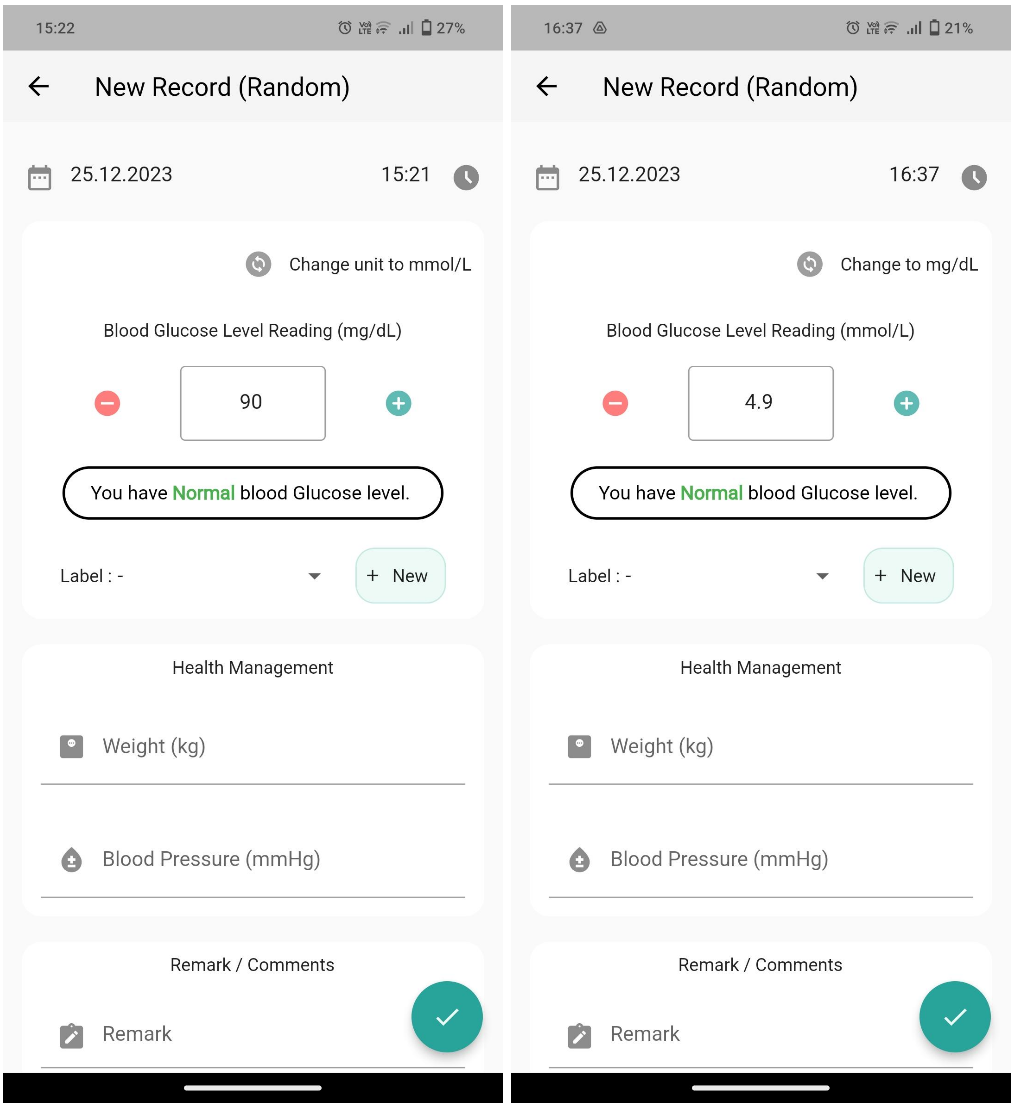


Figure C.49: Add Blood Glucose Type page

When the user requests for a new record addition, the system will prompt the user with the page above. The user is allowed to change the measuring unit based on their preference. Upon completing the necessary information, the user may click on the ‘done’ button.

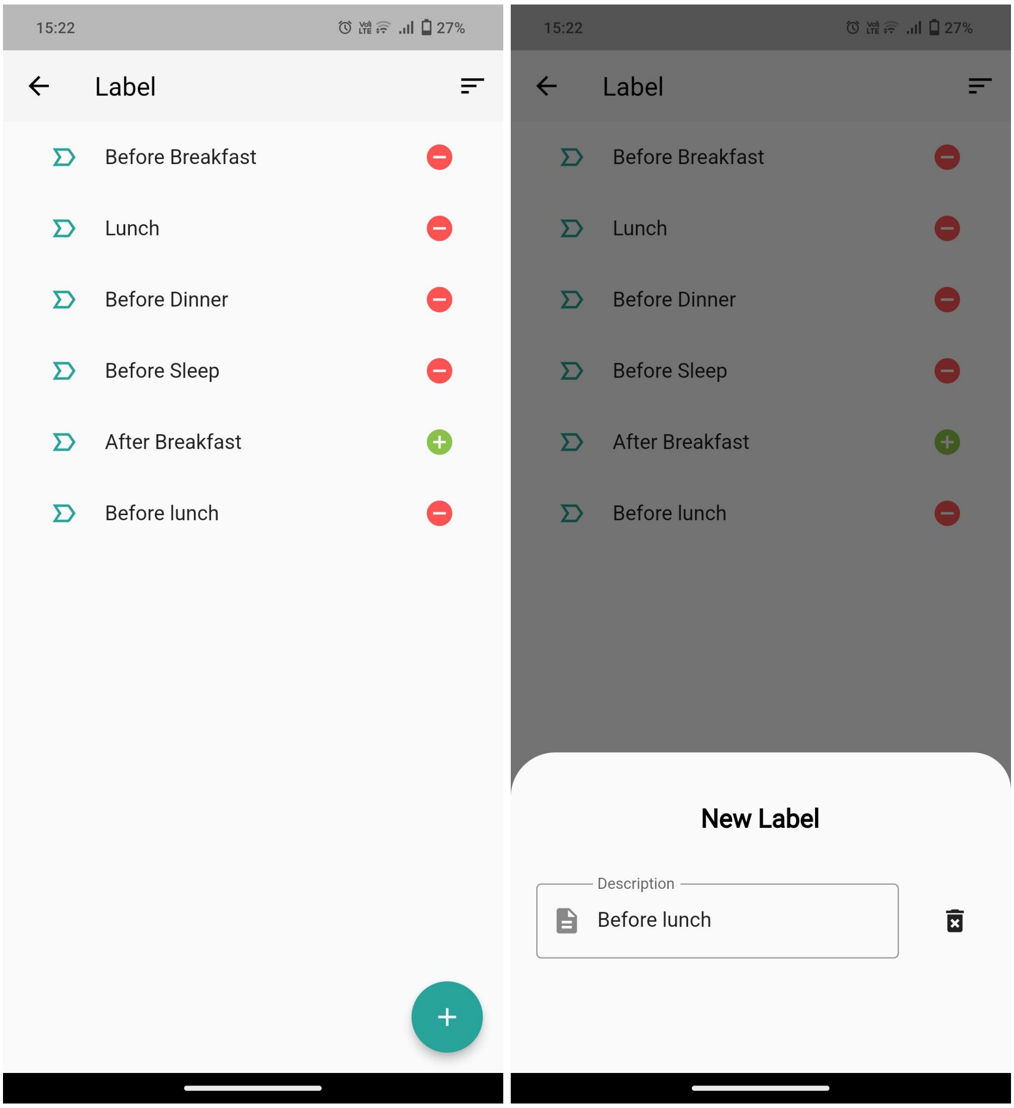


Figure C.50: Label Management page

If the user would like to add more labels for the record, they may click on the ‘New’ button asn the system will navigate the user to this label management page. In this page, the user can add a new label and edit the existing label.

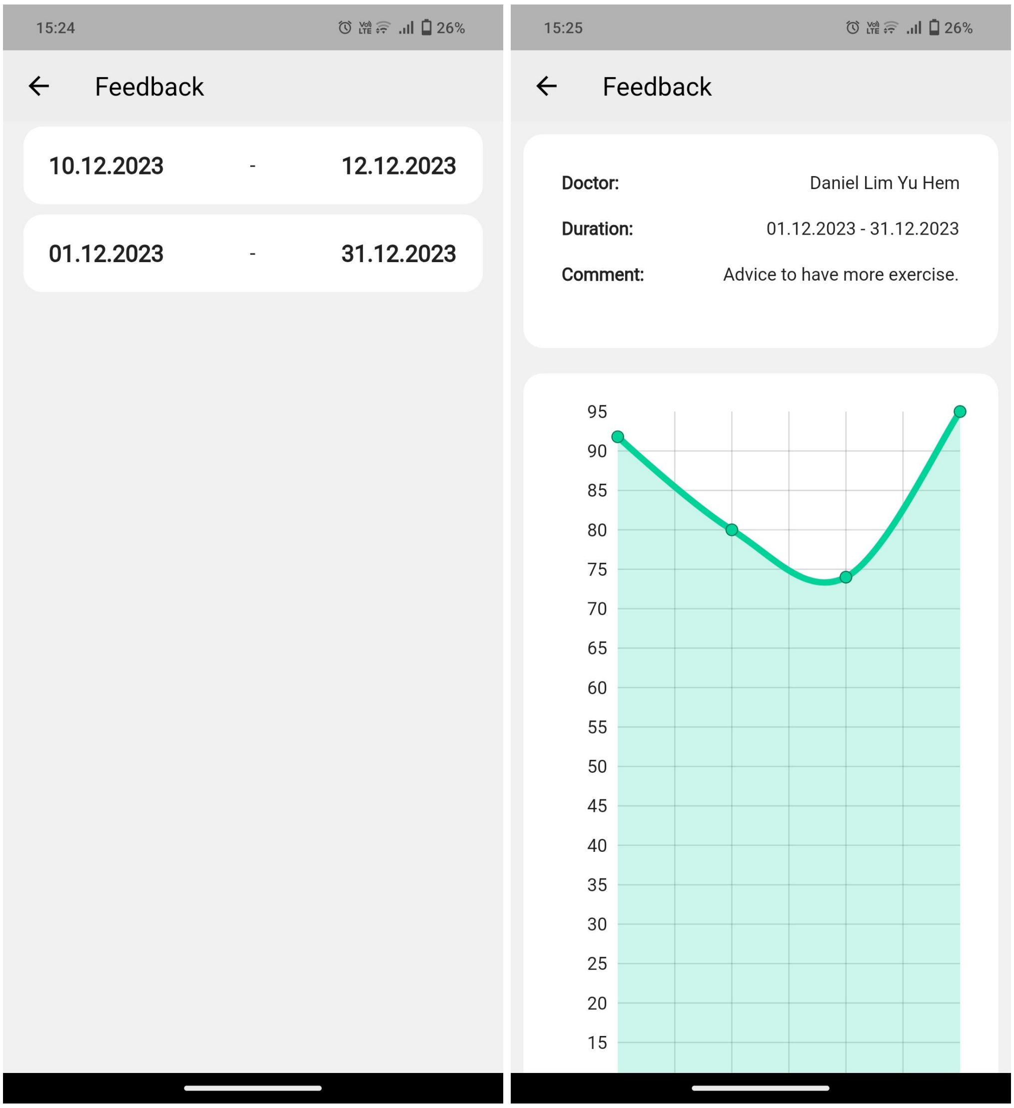


Figure C.51: Feedback Management page

Based on the blood glucose recorded by the patients, the doctor can make necessary feedback. The user may approach this feedback information through the feedback page. By clicking on the record, the relevant feedback will be displayed.

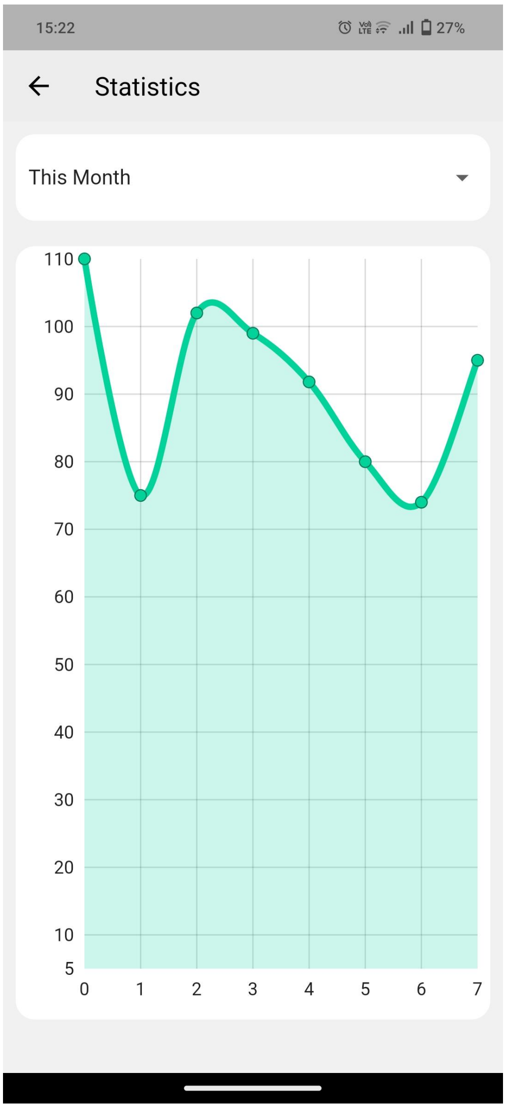


Figure C.52: View Statistics page

If the user would like to visualise their own data, they can navigate to the statistics page and filter the duration.

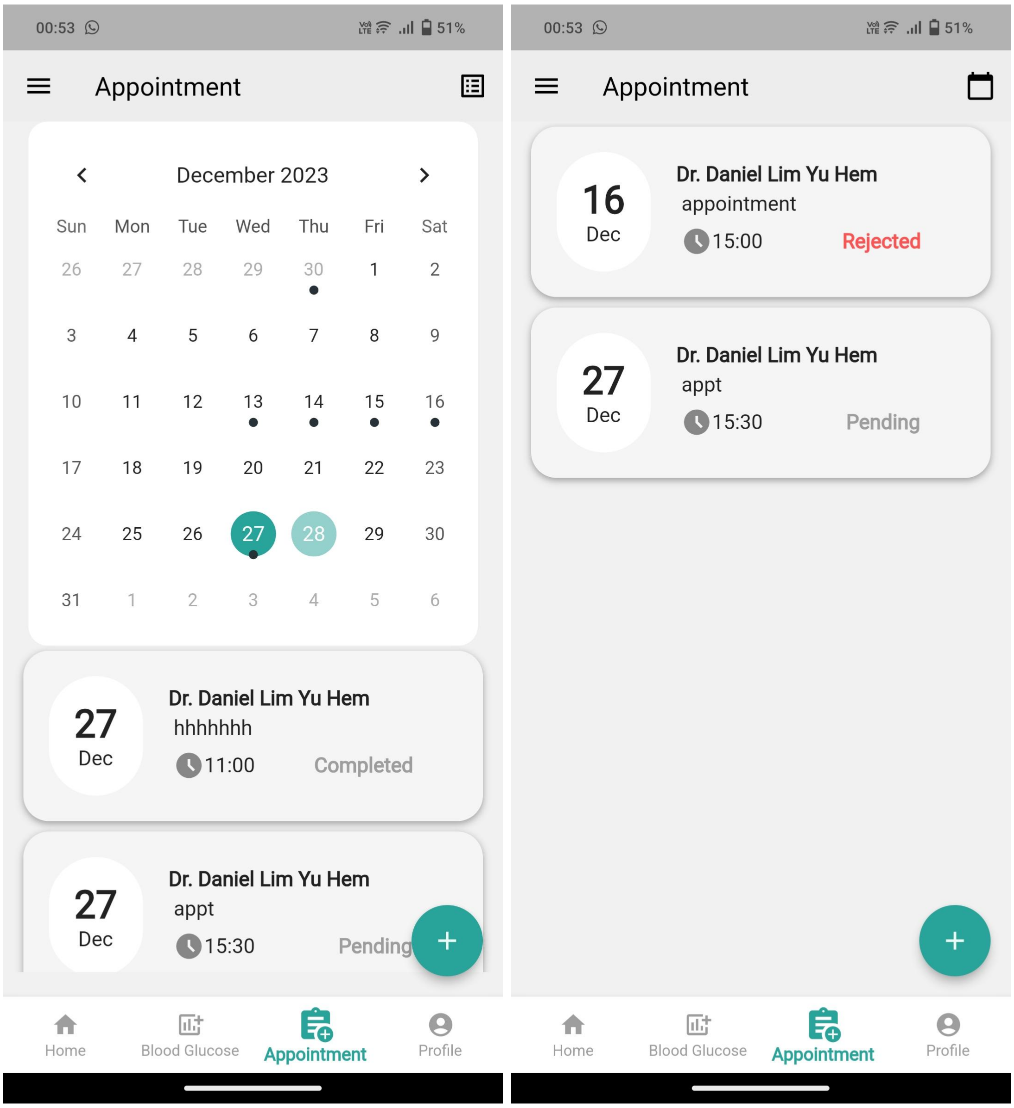


Figure C.53: Appointment Management page

By selecting the ‘appointment’ option from the bottom navigation, the system will present the user with the appointment management page. On this page, the user can check all the appointments that were made previously.

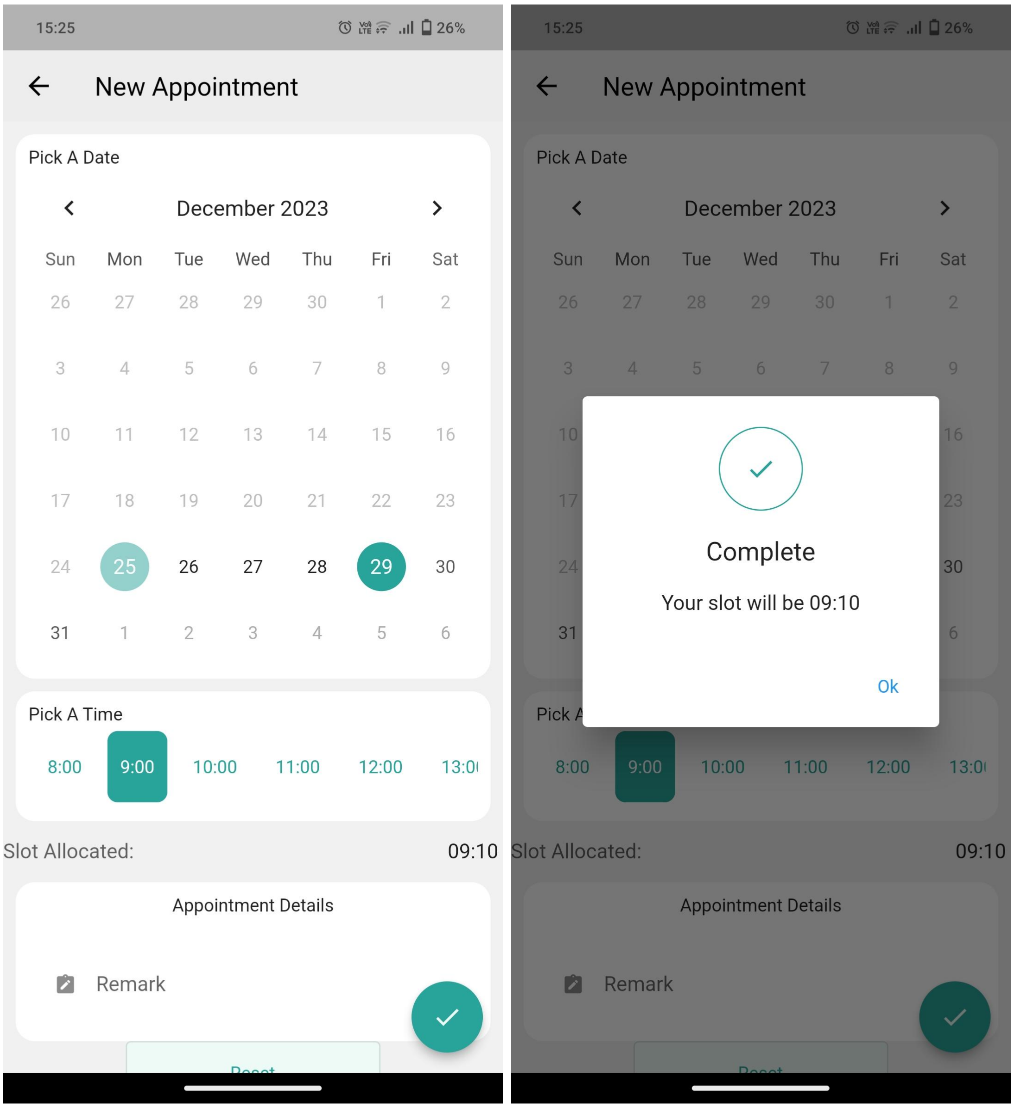


Figure C.54: Add Appointment page

When adding a new appointment, the user has to select the date and time duration. While the time duration is selected, the system will automatically assign a slot to them. Upon completion of the information adding, the user can submit the appointment, waiting for approval.

Figure C.55: View or Edit Appointment page

When the user selects any previous record, the system will navigate the user to the appointment details page. In this page, the user can view the appointment details, edit or even delete the appointment.

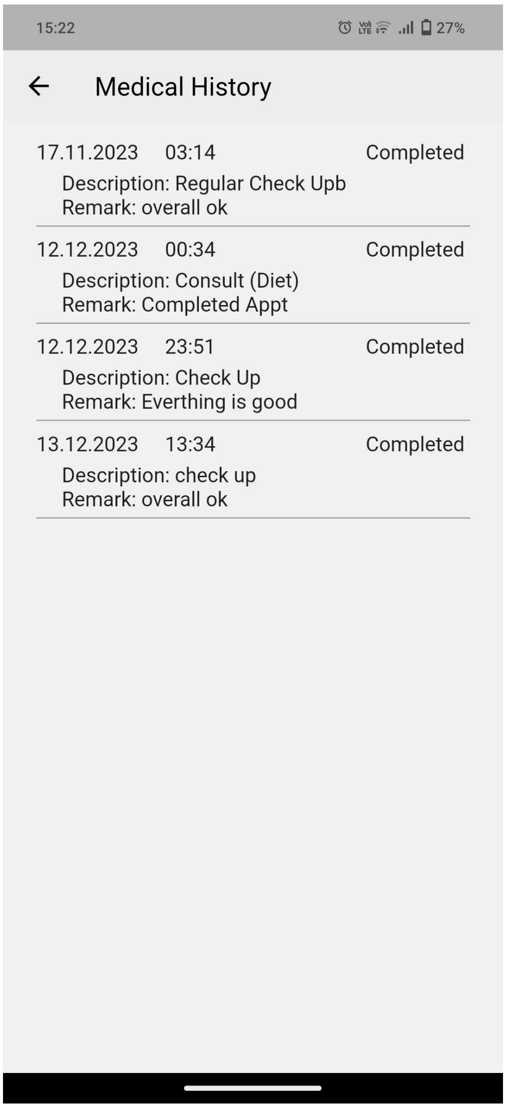


Figure C.56: Medical History Management page

For the option of 'Medical History', the user will be displayed with all the medical history that he / she has completed. By selecting the record, the detailed information will be displayed.

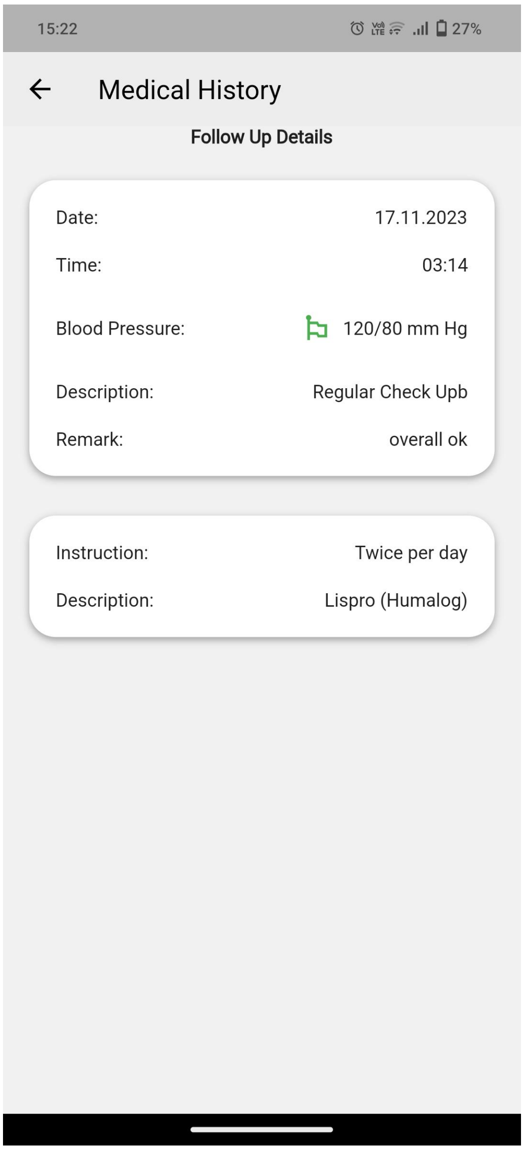


Figure C.57: View Medical History page

By selecting the record, the detailed information will be displayed.

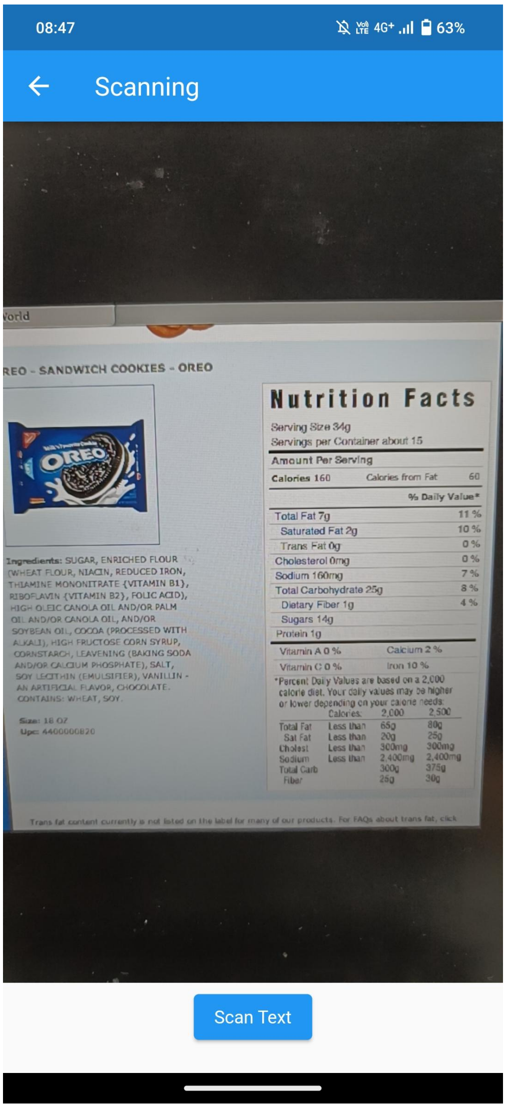


Figure C.59: Ingredient Scanning page

The user can perform the ingredient scanning by selecting the option ‘Ingredient Scanning from the drawer. In the scanning page, the user can focus their camera onto the ingredient image and perform the scanning.

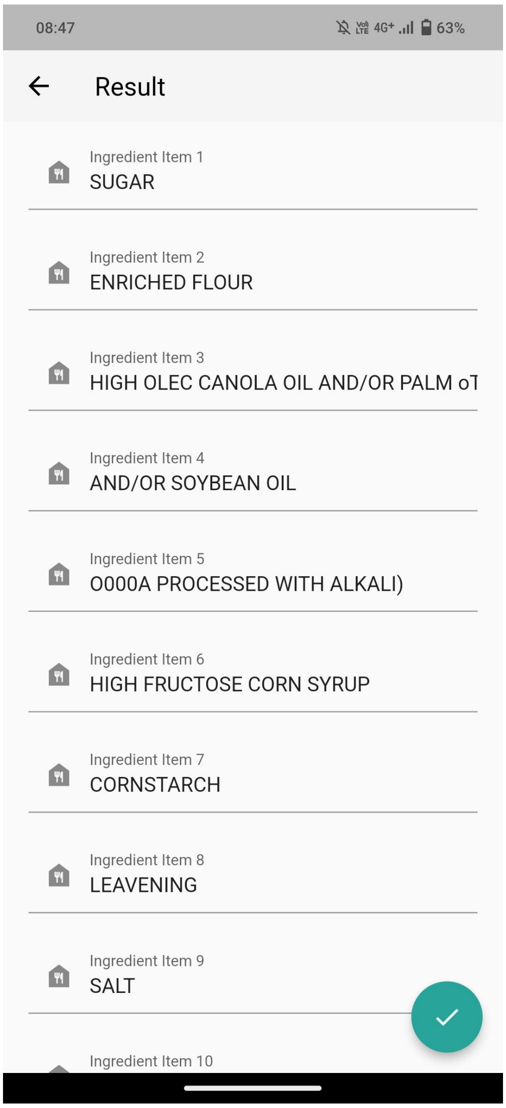


Figure C.60: Result Verification page

When the image has been scanned, the system will present the user with a list of extracted items. The user can modify the items if necessary and confirm the result of scanning.

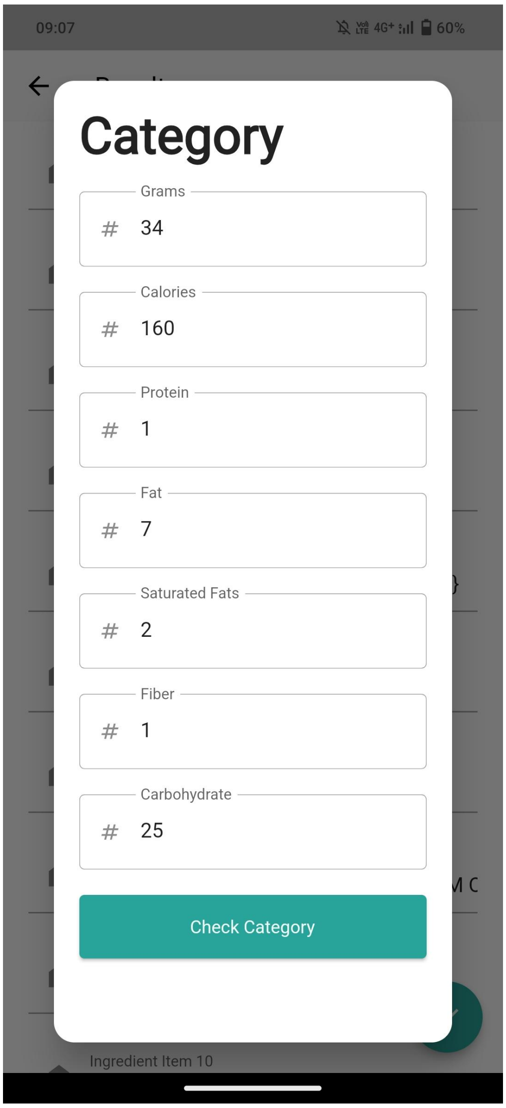


Figure C.61: Nutrition Fact Verification page

A pop-up for the verification of nutrition facts will be prompted, allowing the user to verify and modify the values when necessary..

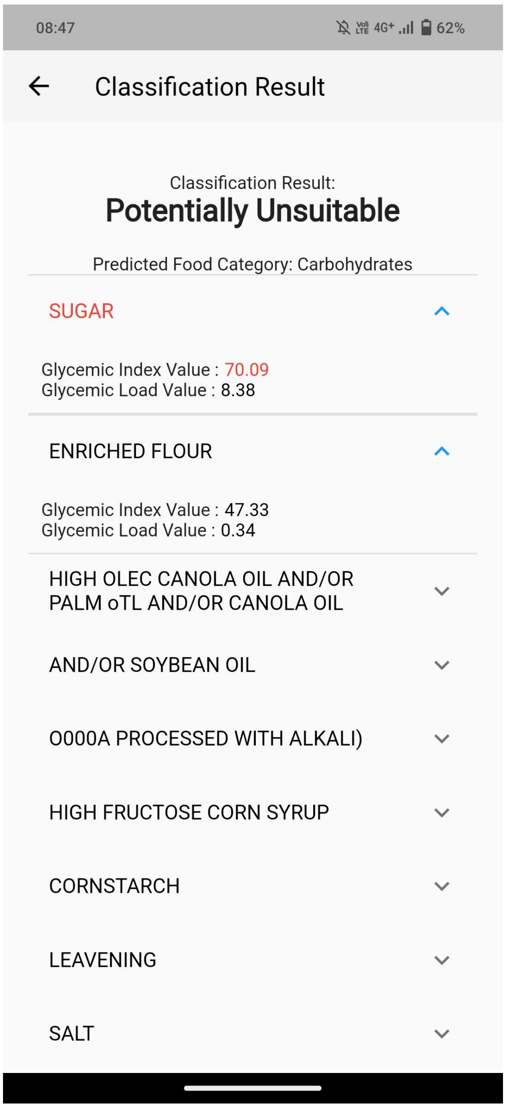


Figure C.62: Classification Result page

The classification result will be displayed to the user. The user can extend each ingredient item to view the glycemic related information.

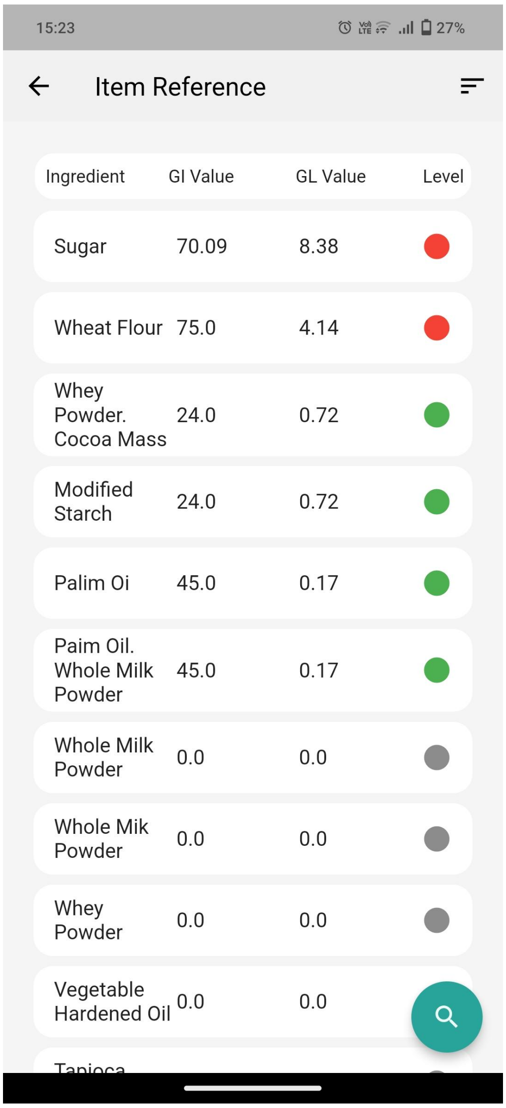


Figure C.63: Item Reference page

The user can navigate to the 'Item Reference' page to retrieve more information. They can also perform the searching or sorting onto the ingredient items.

Appendix 4: Originality Report

FYP Final Document

ORIGINALITY REPORT

6%	5%	3%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	eprints.utar.edu.my Internet Source	1 %
2	apps.dtic.mil Internet Source	<1 %
3	fdocuments.in Internet Source	<1 %
4	firebase.google.com Internet Source	<1 %
5	hdl.handle.net Internet Source	<1 %
6	Lucjan Stapp, Adam Roman, Michaël Pilaeten. "Chapter 7 Chapter 4 Test Analysis and Design", Springer Science and Business Media LLC, 2024 Publication	<1 %
7	www.mkcr.cz Internet Source	<1 %
8	Puchalapalli Surya Vradhan Reddy, Mannepalli Vamsi Krishna, R. Aishwarya, R. Yogitha, K.	<1 %

Appendix D.1: Originality Report