FACULTY OF INFORMATION SCIENCE STUDIES

UNIVERSITI TEKNOLOGI MARA (UiTM) SELANGOR, CAMPUS PUNCAK PERDANA

BACHELOR OF INFORMATION SCIENCE (HONOURS) INFORMATION SYSTEMS MANAGEMENT (CDIM262)

ADVANCED WEB DESIGN DEVELOPMENT AND CONTENT MANAGEMENT (IMS566)

GROUP PROJECT :

CLINIC BOOKING SYSTEM

PREPARED FOR:

DR. MUHAMMAD ASYRAF BIN WAHI ANUAR

PREPARED BY:

NAME	STUDENT ID.
‘AINUL WAFA BINTI ISMAIL NASARUDDIN	2025140157
NUR AULA ZULFA BINTI ZAMRI	2025121345
NUR SYAHIQAH BINTI JUZARNOR	2025112451
NURIN AFIFAH BINTI ZAKARIA	2025156507
WAN HAZRINA BINTI WAN ZAHARI	2025147975

GROUP:

CDIM2624A

SUBMISSION DATE:

2 FEBRUARY 2025

ADVANCED WEB DESIGN DEVELOPMENT AND CONTENT MANAGEMENT (IMS566)

GROUP:

CDIM2624A

GROUP PROJECT :

CLINIC BOOKING SYSTEM

PREPARED BY:

NAME	STUDENT ID.
‘AINUL WAFA BINTI ISMAIL NASARUDDIN	2025140157
NUR AULA ZULFA BINTI ZAMRI	2025121345
NUR SYAHIQAH BINTI JUZARNOR	2025112451
NURIN AFIFAH BINTI ZAKARIA	2025156507
WAN HAZRINA BINTI WAN ZAHARI	2025147975

FACULTY OF INFORMATION SCIENCE STUDIES

UNIVERSITI TEKNOLOGI MARA (UiTM) SELANGOR, CAMPUS PUNCAK PERDANA

ACKNOWLEDGEMENT

First and foremost, we would like to express our deepest gratitude to Allah SWT for His blessings, guidance, and strength throughout the completion of this project. His continuous support allowed us to overcome challenges and complete the clinic management system successfully. Without His grace, this project would not have been possible.

We would like to extend our sincere appreciation to Dr. Muhammad Asyraf bin Wahi Anuar for his guidance, valuable advice, and constructive feedback throughout this project. His supervision and encouragement have greatly helped us in understanding the project requirements and ensuring that our work meets the expected standards.

Our heartfelt thanks also go to all group members for their dedication, cooperation, and teamwork, as well as to our friends for their encouragement and moral support. We also acknowledge Universiti Teknologi MARA (UiTM) for providing the academic support, resources, and opportunities that enabled us to develop our skills and complete this project successfully.

TABLE OF CONTENTS

1.0 INTRODUCTION

1.1 Overview System 1

1.2 Purpose 2

1.3 Objectives 2

2.0 ENTITY-RELATIONSHIP DIAGRAM (ERD) 3

3.0 SYSTEM REQUIREMENTS

3.1 Software 4

3.2 Version Compatibility 4

3.3 Dependencies 5

4.0 USER INTERFACE OVERVIEW

4.1 Layout 6

4.2 Navigation 12

5.0 FEATURES AND FUNCTIONALITY 17

6.0 WORKFLOW OF FORM 25

7.0 TEAM ROLES

7.1 Contributions 27

7.2 Contact Information 27

8.0 CONCLUSION & REFLECTION

8.1 Conclusion 28

8.2 Reflection 28

9.0 REFERENCES

List of Figures

Figure 1	**Entity-Relationship Diagram**
Figure 2	**Admin Dashboard - Interface**
Figure 3	**Admin Profile - Interface**
Figure 4	**Admin Patient List - Interface**
Figure 5	**Admin Appointment List - Interface**
Figure 6	**Admin Doctor List - Interface**
Figure 7	**User Dashboard - Interface**
Figure 8	**User Profile Patient - Interface**
Figure 9	**User Appointment List (My Appointment)**
Figure 10	**User Appointment Details (View)**
Figure 11	**Patient Appointment Letter PDF (View)**
Figure 12	**Patient Appointment Letter (PDF Download)**
Figure 13	**Admin Navigation Bar**
Figure 14	**Search and Add Navigation Bar (Admin Patient List)**
Figure 15	**Search and Add Navigation Bar (Admin Appointment List)**
Figure 16	**Search and Add Navigation Bar (Admin Doctor List)**
Figure 17	**User Navigation Bar**
Figure 18	**View and Edit Navigation (User Profile Patient)**
Figure 19	**View and Export PDF (User Appointment List)**
Figure 20	**Log in Interface**
Figure 21	**List Patient Features**
Figure 22	**Actions to view the patient's data**
Figure 23	**Patient's details**
Figure 24	**Actions to edit the patient's data**
Figure 25	**Forms of patient's data to edit**

Figure 26	**Actions to delete the patient's data**
Figure 27	**Confirmation to delete pop-up message**
Figure 28	**Search features to acquire patient's data**
Figure 29	**Features to add to the patient's data form**
Figure 30	**Features from the appointment interface**
Figure 31	**Features from the profile patient interface**
Figure 32	**Features from the user's appointment interface**
Figure 33	**User (patient) appointment view**
Figure 34	**PDF of user (patient)**
Figure 35	**User Flowchart**

1.0 INTRODUCTION

In this age of widespread innovation in business systems, we decided there could be no better approach than designing a clinic management system. We named the system Awariqna Clinic, inspired by combining all of our short names together. This initiative is important for improving patient experiences, enhancing administrative operational efficiency, and elevating clinic workflows. Thus, we have decided to build a comprehensive clinic system using a carefully selected technology stack that supports our learning objectives and project requirements. We have chosen Laragon as our primary development environment because our coursework emphasized its use, providing an all-in-one local server with MySQL and PHP that allowed us to set up, test, and iterate quickly without needing external servers.

In building the database, we utilized phpMyAdmin to design and manage our relational database schema, ensuring secure storage and fast access for key features such as patient_id, doctor_id, and appointments. We used CakeBake from the CakePHP framework to expedite the development of our backend structure, which included models, controllers, and views with complete CRUD functionality. This allowed us to concentrate on tailoring the logic to our system's requirements. Furthermore, we have integrated the RECRUD framework, which is owned by our lecturer, to display and project the system's functions and features on the frontend. This made our work much easier by offering ready-made components for responsive interfaces, data displays, and interactive elements, allowing us to effectively demonstrate how the system fulfills the assignment's objectives, such as simulating real clinic workflows.

1.1 Overview System

Awariqna Clinic provides a comprehensive web-based platform designed to handle key operations in a small clinic setting, including patient registration, appointment scheduling, and medical records management. Invented by using Laragon for local development, phpMyAdmin for database handling, CakeBake for efficient backend scaffolding via CakePHP, and the RECRUD framework for intuitive frontend display, the system ensures secure data management, user-friendly interfaces, and real-time functionality to support staff in delivering efficient healthcare services while fulfilling the assignment objectives.

1.2 Purposes

We serve a few purposes by implementing the project, including:

- To generate a fully functional system
- To provide a friendly accessibility system for users
- To develop an efficient management system for data within the database

The primary purposes of our clinic management system include generating a fully functional prototype that integrates patient registration, appointments and providing user-friendly accessibility through intuitive RECRUD interfaces, and developing efficient database management via phpMyAdmin and CakePHP for secure, optimized handling of sensitive data like records and reports.

1.3 Objectives

To develop a fast, accurate, and straightforward clinic management system for efficient data management and daily operations.

2.0 ENTITY-RELATIONSHIP DIAGRAM (ERD)

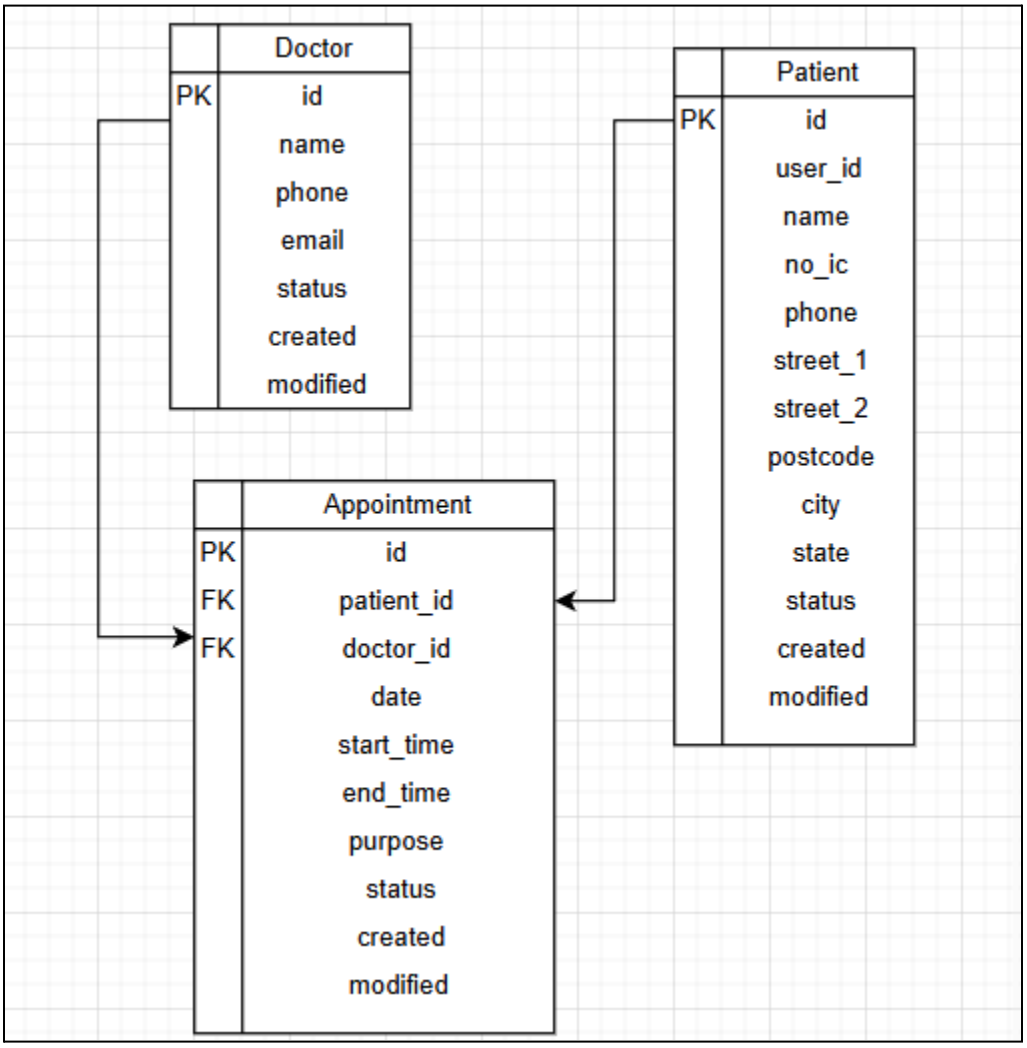

Figure 1: Entity-Relationship Diagram

Link Repository GitHub (master): https://github.com/AulaZulfa/Klinik_Awariqna_IMS566.git

Backup Link Google Drive:

https://drive.google.com/drive/folders/1C-hOULkSNeLSaSQbbMMqsNIunJ6WA1gf?usp=drive_link

3.0 SYSTEM REQUIREMENT

These are some of the requirements in developing the Clinic Booking System, including:

3.1 Software

- Laragon
- MySQL
- PHP
- phpMyAdmin
- Visual Studio Code (VSC)
- Re-CRUD (Dr. Asyraf Framework)

3.2 Version Compatibility

- Laragon: 6.0 (64bit) (*220916 Components*)
- MySQL: 8.0.30 (64bit)
- PHP: 8.1
- phpMyAdmin: 5.2.x
- Visual Studio Code (VSC): 1.108.2 (*VSCodeUserSetup - x64-1.104.2.exe*)

3.3 Dependencies

- Laragon 6.0 (64bit) (*220916 Components*)
    - Operate through our laptops
    - Through Visual Studio Code (VSC)
    - Require APACHE and MySQL

- Re-CRUD (Dr. Asyraf Framework)
    - PHP Extensions - Specifically requires the intl extension to be enabled in the other php.ini. It also relies on standard extensions like pdo_mysql, mbstring, and openssl.
    - Database Structure - The framework requires tables to include a status (int) attribute with a default value of 1 for its internal record management (Active/Disabled/Archived).
    - SQL Mode - Requires the sql-mode in the other MySQL configuration (my.ini) to be set to blank (sql-mode = "") to avoid strict mode errors during data entry.
    - BaseEngine: It is built on the CakePHP framework, meaning it follows the MVC (Model-View-Controller) architecture.

- phpMyAdmin 5.2.x
    - Version: Compatible with PHP 8.1
    - Browser: Requires a web browser with JavaScript enabled
    - Database connection: Depends on the MySQL service being active within Laragon.

- Visual Studio Code (VSC)1.108.2
    - Hardware: Minimum 1 GB RAM and a 1.6 GHz or faster processor.
    - Needs to setup user.
    - Extensions: To work effectively with the other user stack, it usually depends on extensions like PHP Intelephense, MySQL, and Auto Rename Tag.

4.0 USER INTERFACE OVERVIEW

Our Clinic Booking System displays the user interface. This explains that a user interface (UI) is the digital link that connects a human user to a computer system. It includes all of the visual components, interactive elements, and sensory feedback mechanisms that enable a user to engage with hardware or software. In graphical user interfaces (GUIs)), it serves as a translation layer that converts complicated machine code into a digestible environment of windows, icons, menus, and buttons. Usability and accessibility are given top priority in an efficient UI overview, guaranteeing that the design is simple enough for a user to accomplish their objectives with little difficulty. Therefore, our system uses an intuitive layout and streamlined navigation to provide clear structure and easy access to all project documentation.

4.1 Layout

There are two distinct layouts to serve the different needs of our system, divided to admin interface and the user interface. Due to their serving different purposes, there are some design variations across both platforms.

➢ **Admin Interface**

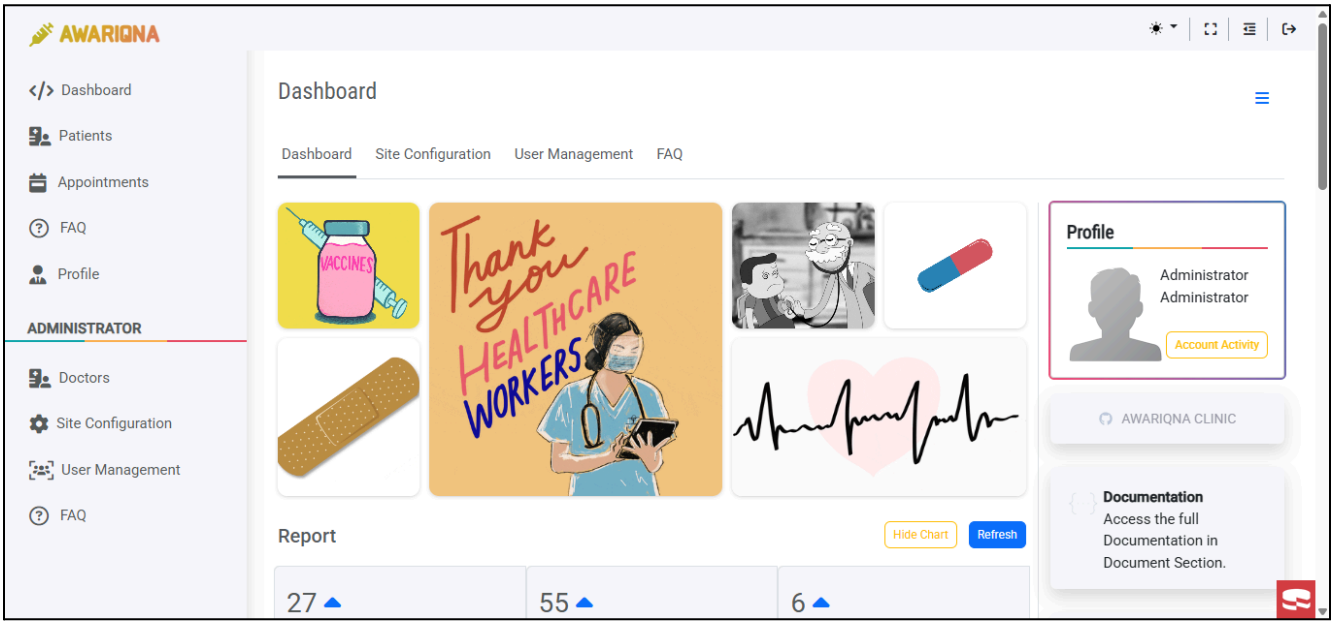

Figure 2: Admin Dashboard - Interface

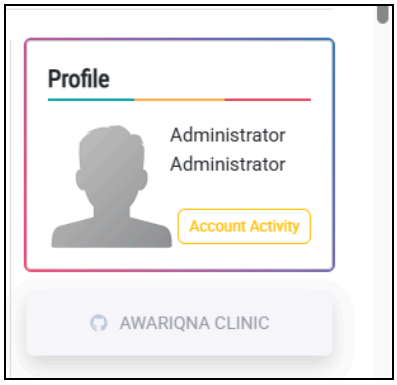

Figure 3: Admin Profile - Interface

- Determine the personalization
- Admin to authorize any important processes

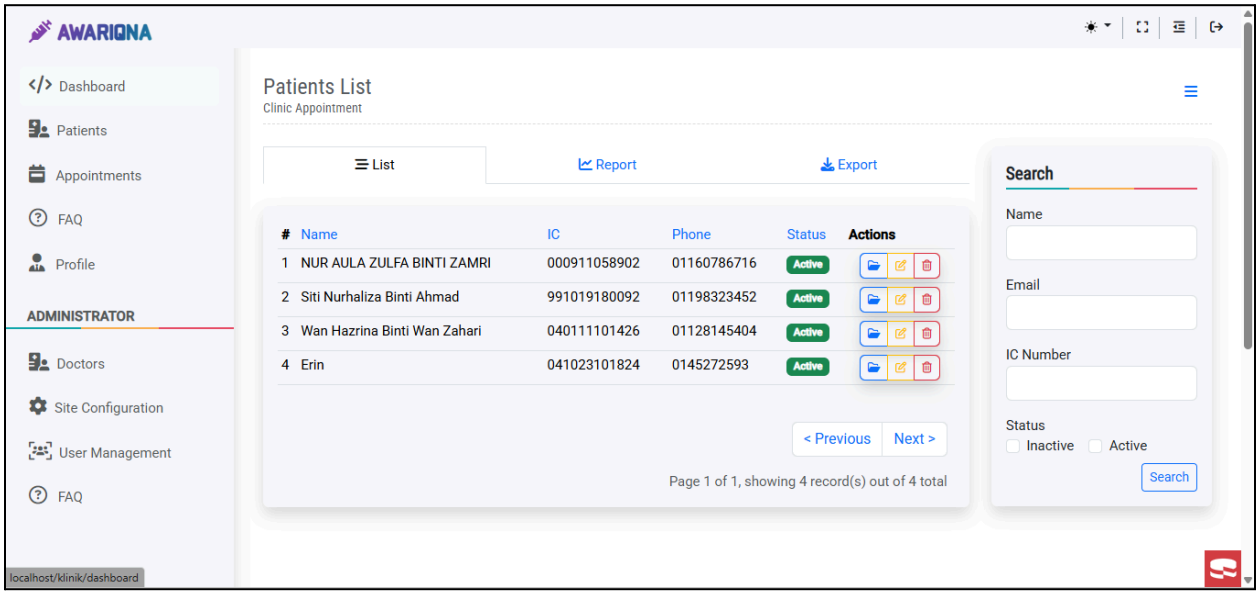

Figure 4: Admin Patient List - Interface

- We implement the search to fulfill the RECRUD principles, as what the S meant as Search.

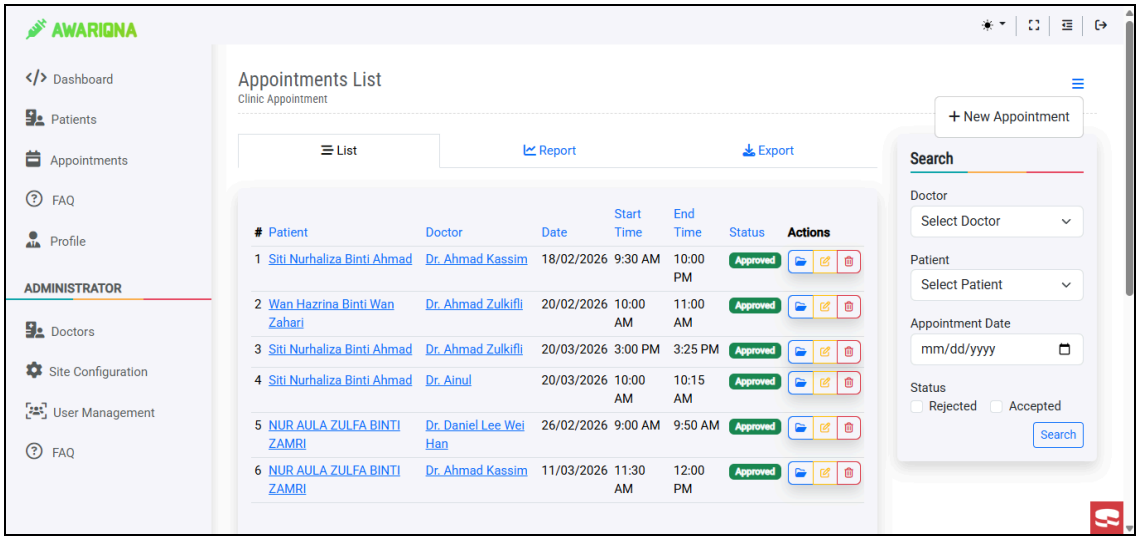

Figure 5: Admin Appointments List - Interface

- The list follows the patient name, the doctor who is available for the patients, as admin assigned to authorize any of the processes.
- Also providing the start and end time of the session, and authorizing the status of the appointments, whether it approves or not.

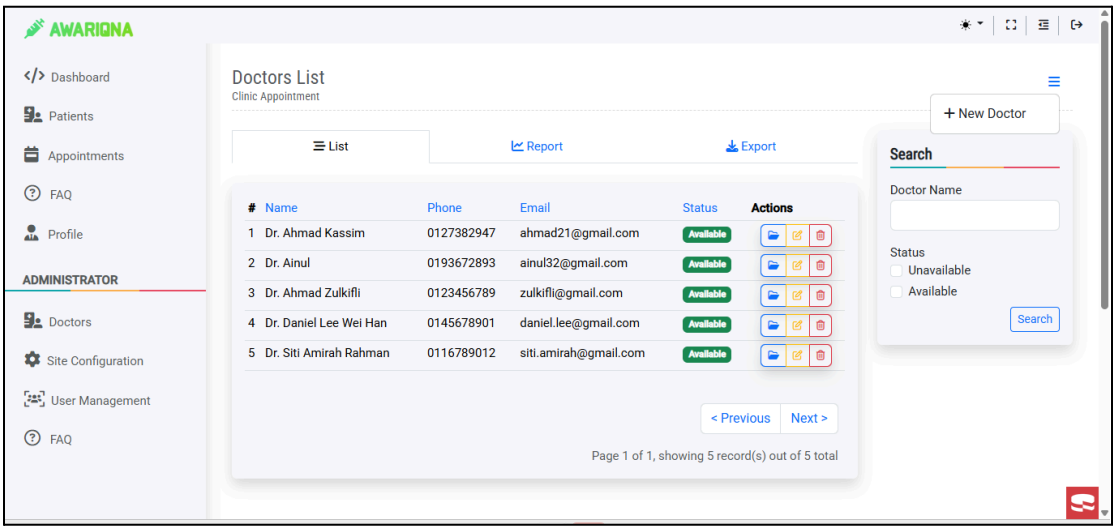

Figure 6: Admin Doctor List - Interface

- The list follows the name, phone number, email, and the status of availability of the doctor's schedule.
- We also apply the RECRUD as a search for easier access.
- On the right, there could be new data from another doctor at the toggle three lines button.

➢ User Interface

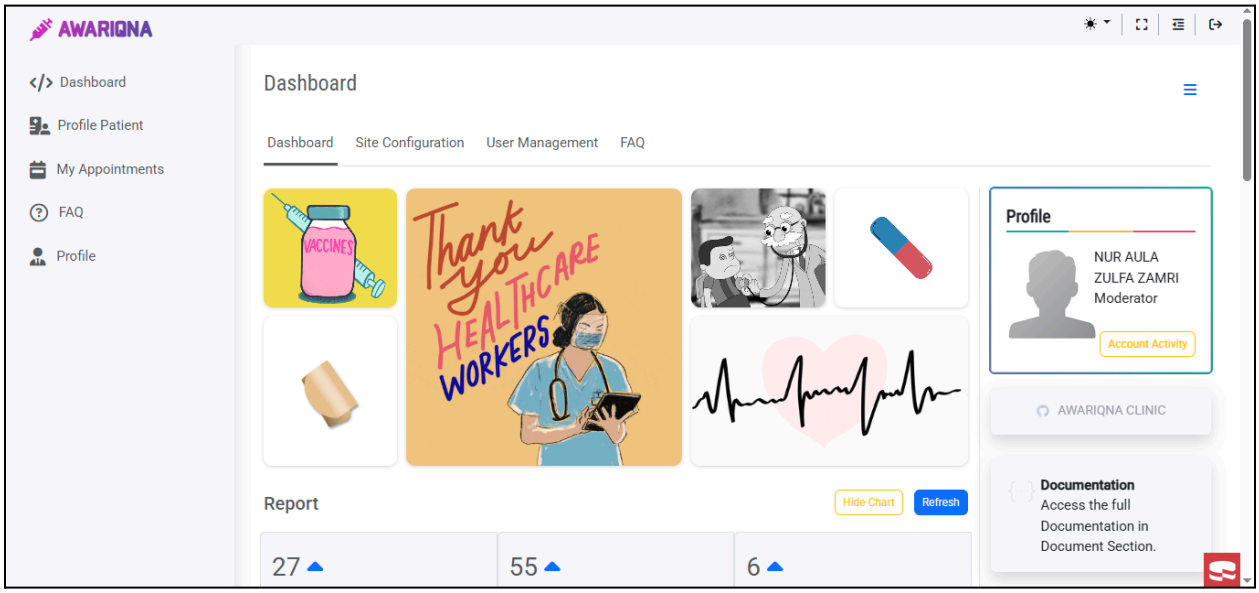

Figure 7: User Dashboard-interface

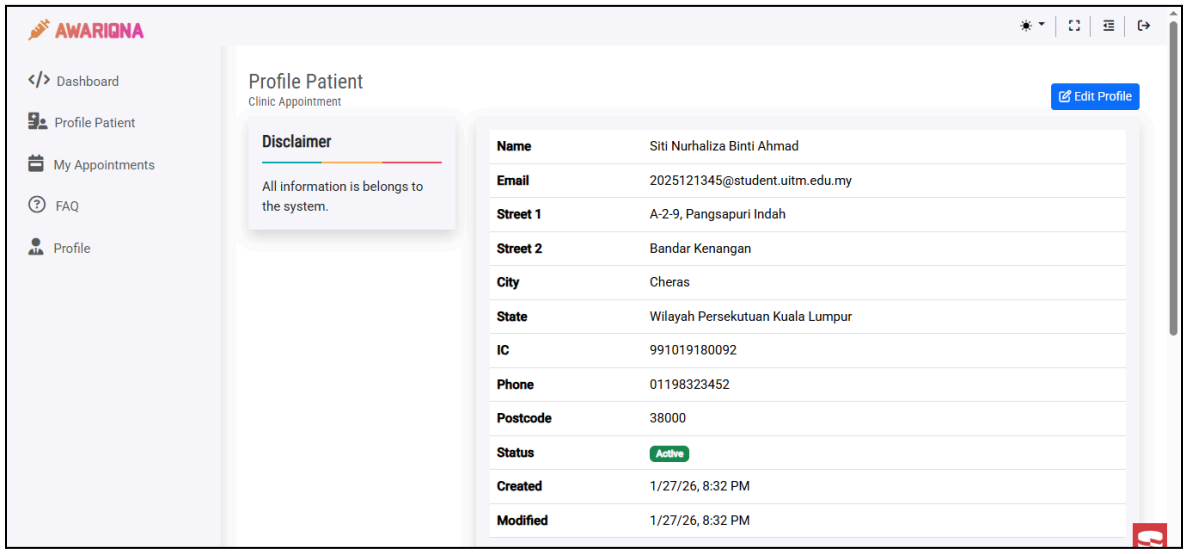

Figure 8: User Profile Patient - interface

- Patients have to register their personal information.
- Profile Patient can be edited by clicking the edit profile button.
- This profile patient's function lets the admin assign them their appointments.

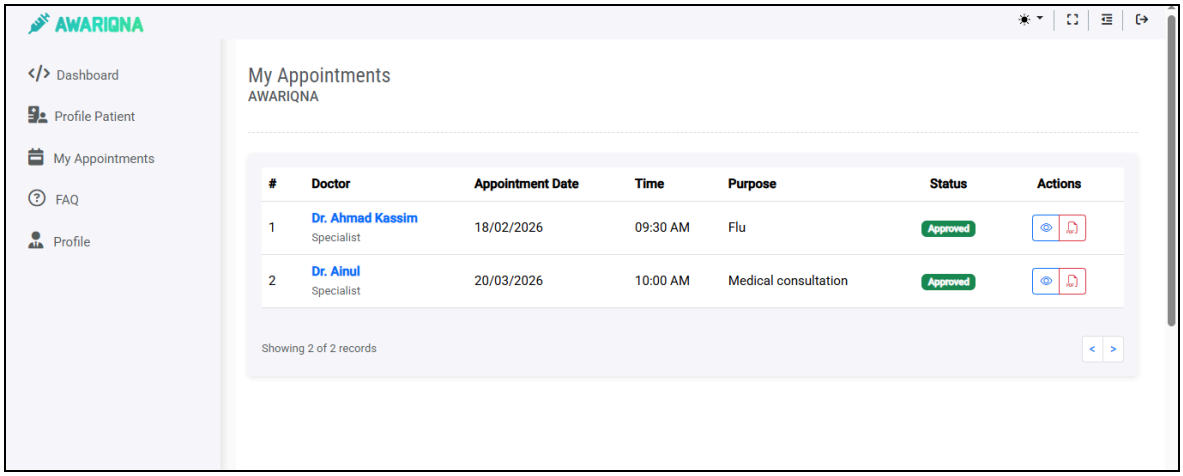

Figure 9: User Appointment List (My Appointment)

- This is the patient appointments page.
- Patients can view their appointments on this page and also know the doctor assigned for them, the date and time, and the purpose of the appointment.
- Patients can view and download the appointment letter directly on this page.

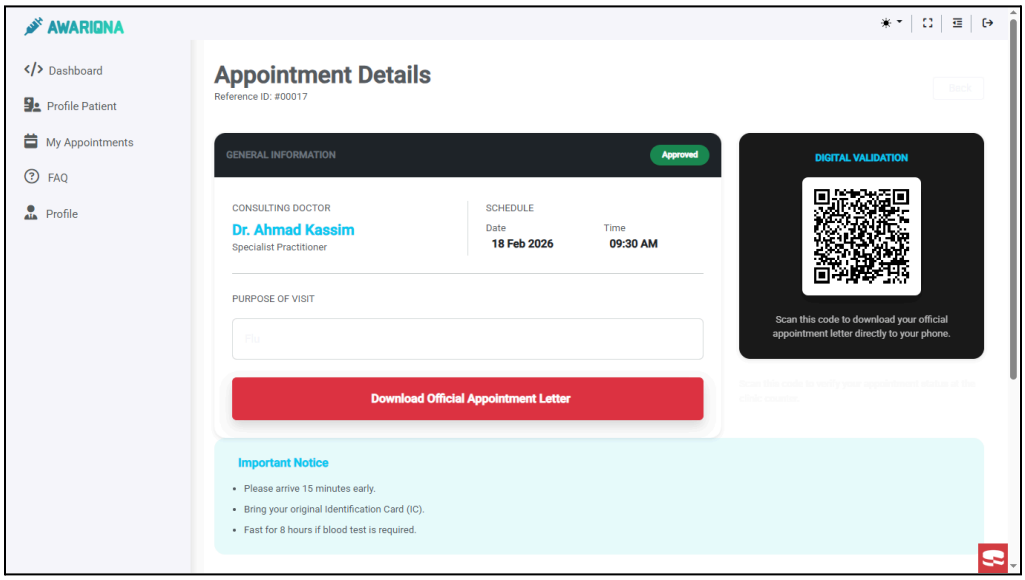

Figure 10: User Appointment Details (View)

- This appears after the patients click the ‘eye icon’ at My Appointment page.
- On this page, patients can view and download the appointment letter.
- They can scan the QR code to download the appointment letter on their device.
- Patients can also download directly by clicking the red button.
- This appointment details page helps patients view their appointment clearly.

Figure 11: Patient Appointment Letter PDF (view)

- When patients click to view the appointment letter, they can view our clinic appointment letter, as shown in the figure above.
- It contains the patient's name, address, appointment date and time, and the doctor assigned for the patient.

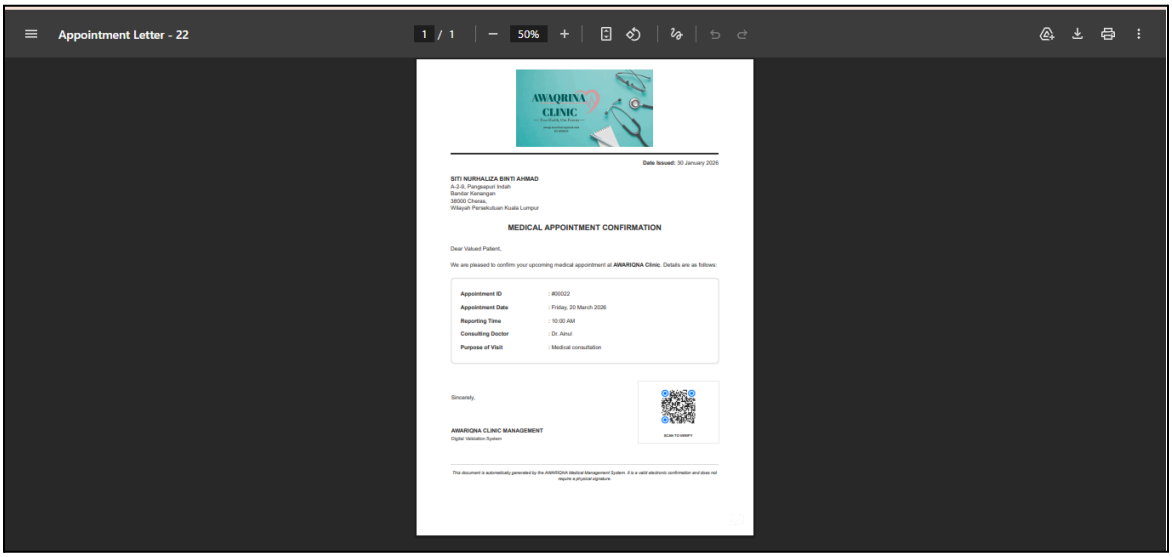

Figure 12: Patient Appointment Letter (PDF Download)

- Our system lets patients download the appointment letter as evidence when they visit our clinic, as assigned to them.

4.2 Navigation

➢ Admin Navigation

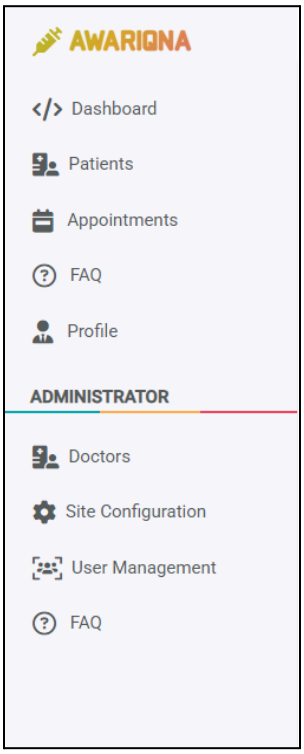

Figure 13: Admin Navigation Bar

- Includes all of the important stuff for the authorization that includes the patients, appointments, doctors, and the dashboard.
- As for other navigation bars, it is fixed to the websites, as it seems that we only add a few to adjust it to our personalized system to be ours, just by using the structure, while it benefits us a lot, by just designing a few stuff into the system.

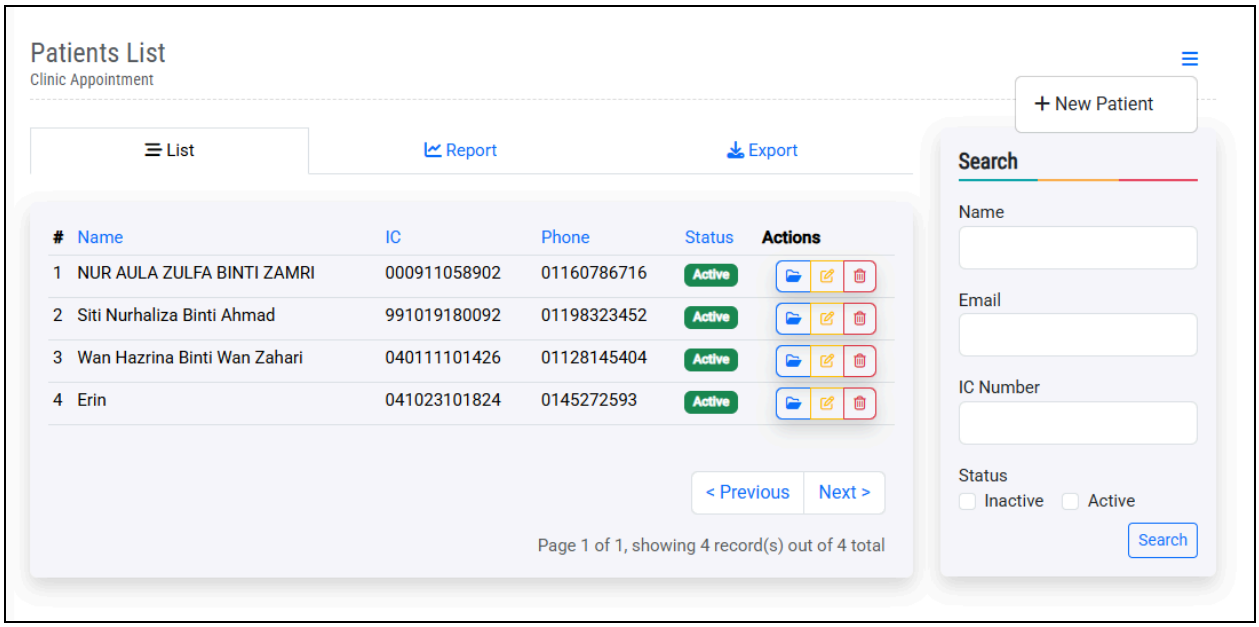

Figure 14: Search and Add Navigation Bar (Admin Patient List)

- Based on the RECRUD principles, the search involves name, email, IC number, and the status of the user.
- In navigating through the system, these features are specifically designed for UX in maintaining the user interface.

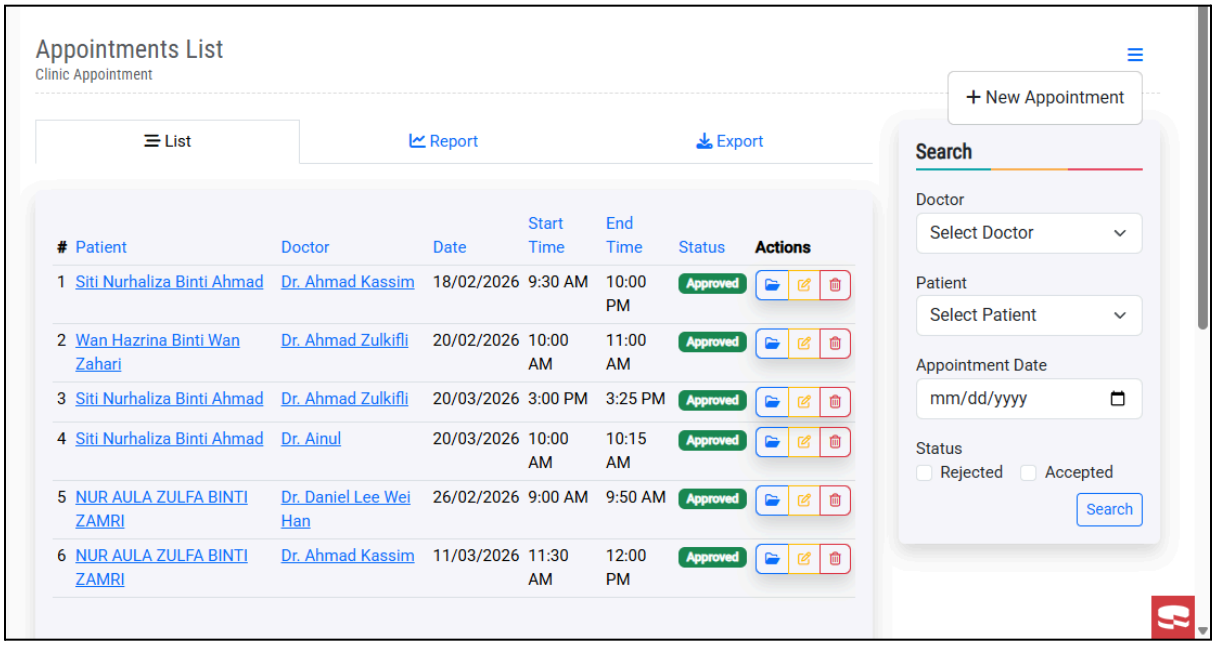

Figure 15: Search and Add Navigation Bar (Admin Appointment List)

- Based on the RECRUD principles, the search involves doctor name, patient name, appointment date and the status of the user.
- In navigating through the system, these features are specifically designed for UX in maintaining the user interface.

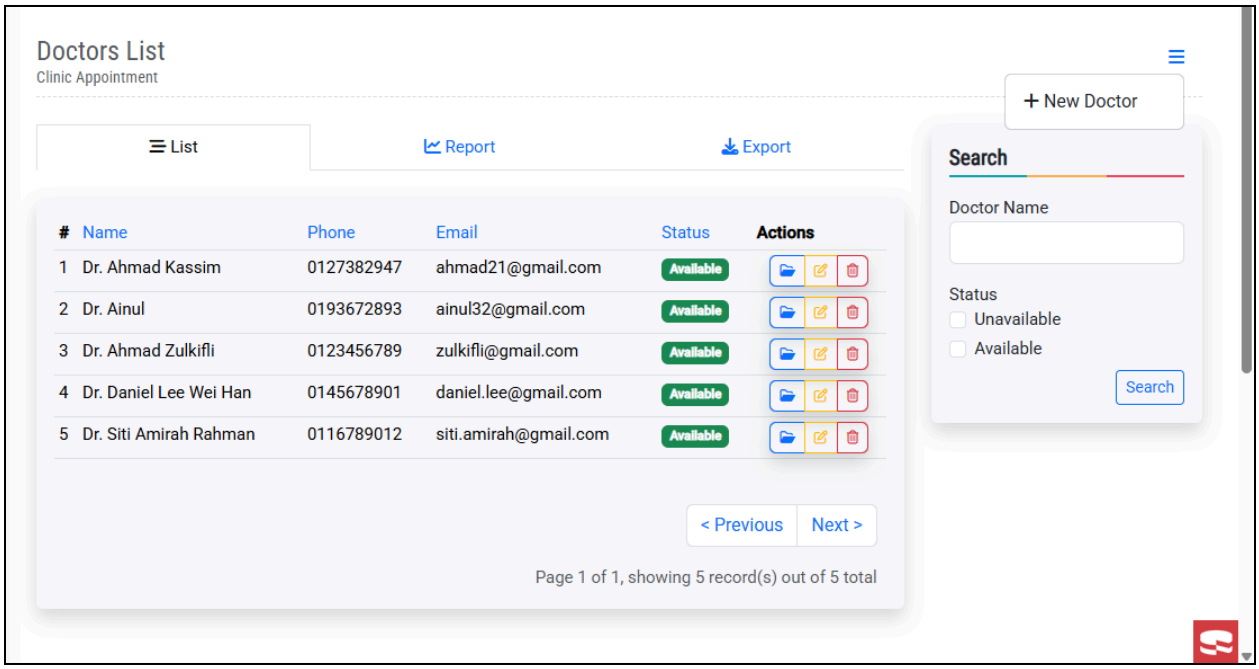

Figure 16: Search and Add Navigation Bar (Admin Doctor List)

- Based on the RECRUD principles, the search involves doctor name and the status of the user.
- In navigating through the system, these features are specifically designed for UX in maintaining the user interface.

➢ User Navigation

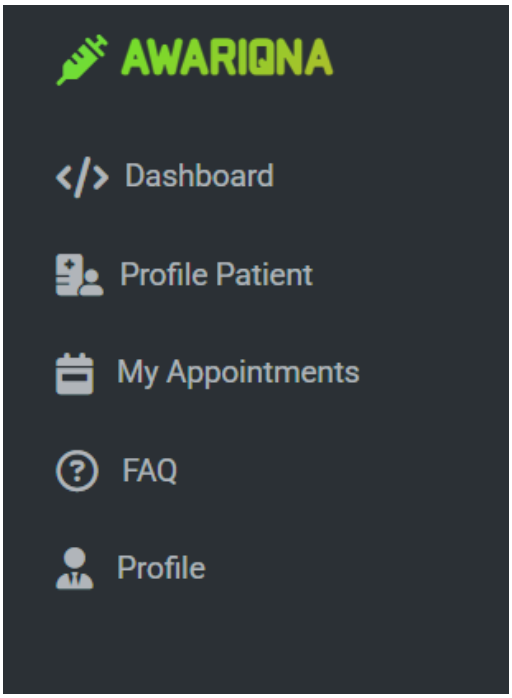

Figure 17: User Navigation bar

- User sidebar contains dashboard, profile patient for admin purpose, my appointments, FAQ, and profile.

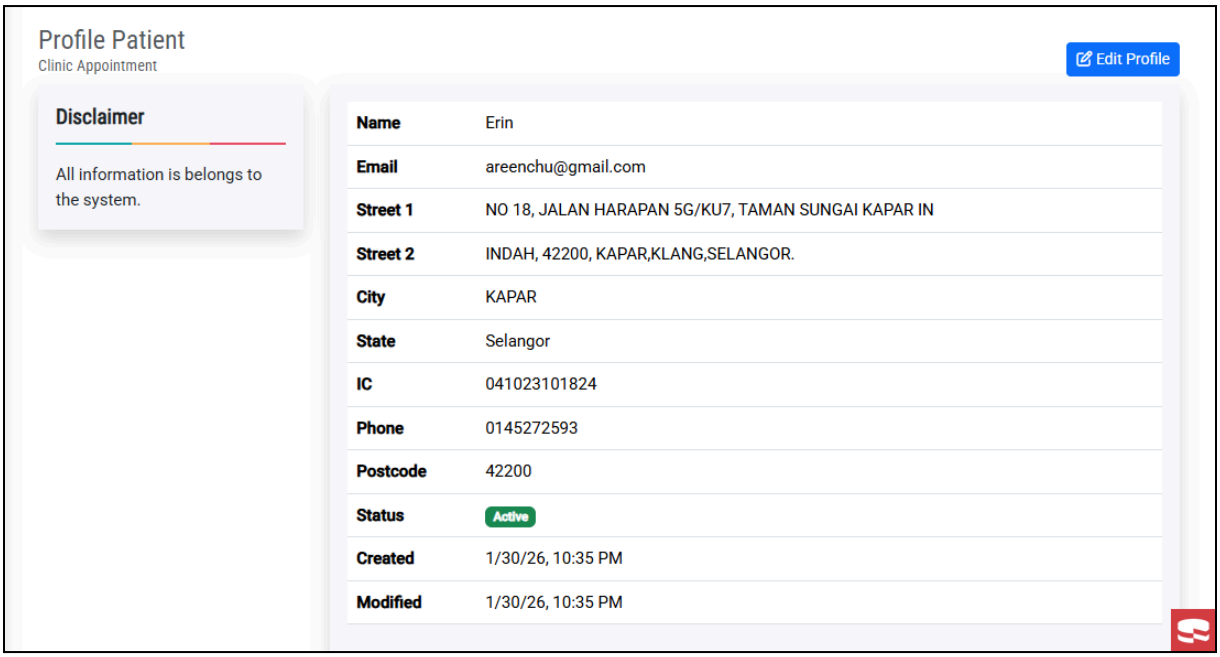

Figure 18:View and Edit Navigation (User Profile Patient)

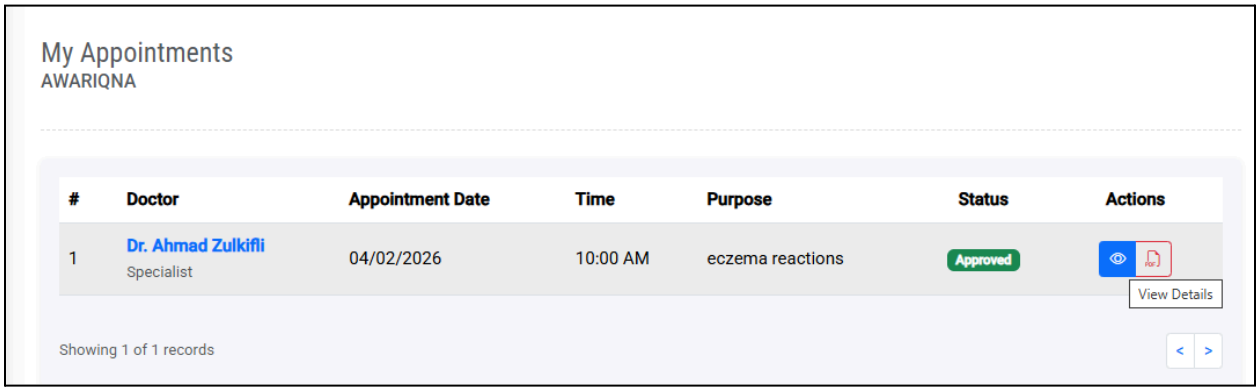

Figure 19: View and Export PDF (User Appointment List)

- The My Appointments (Navigation Sidebar) navigation button leads to the user's appointments list, where the user can only view their appointment.
- These views and exporting the appointments are to help users navigate users in retrieving the appointment document in a simple way.
- By thoroughly going towards these navigation bars, it leads the user to acknowledge which date, by which doctors they are assigned to.

5.0 FEATURES AND FUNCTIONALITY

Here are some of the essential features and functionalities:

1. User Authentication & Profile Management

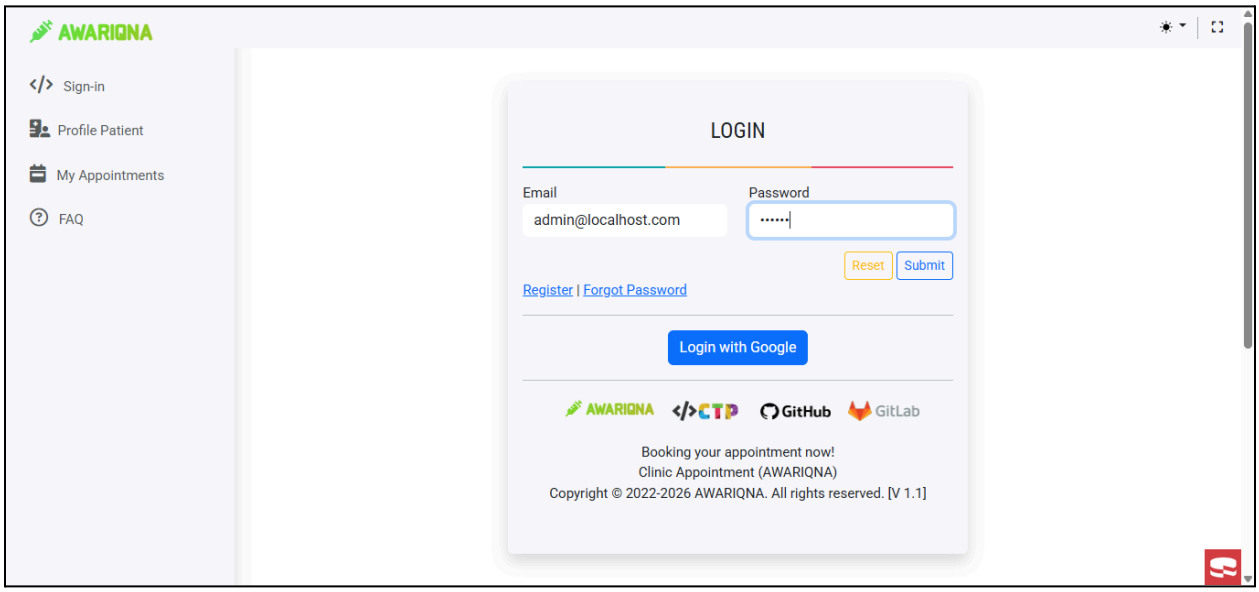

Figure 20: Log In interface

- Secure Login/Logout: Role-based access control (RBAC) ensures Users (Patients) and Admins see only their relevant dashboards.
- Admin Profiles: Could only log in with this username and password.
    - Username: admin@localhost.com
    - Passwords: 123456
- This username and password are used as admin profiles, as it is fixed in the Re-CRUD Framework
- User Profiles: For the user (Patients), users could log on through Google, as it seems at the login page.

2. Admin Interface
    a. Patient Section

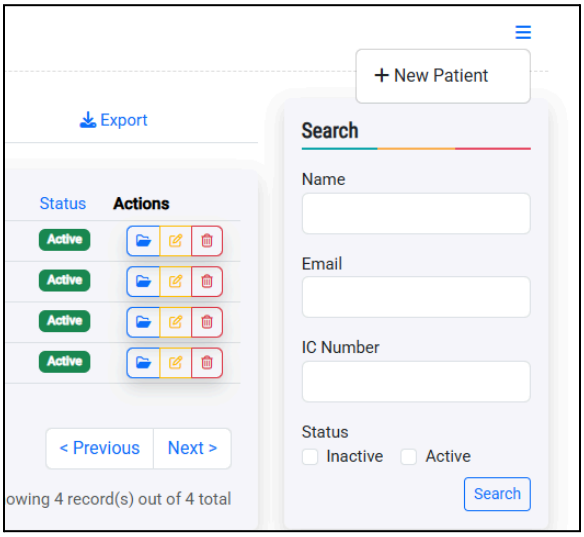

Figure 21: List Patient Features

- On the Patient list for the admin to verify the patient's data, in the actions section, the admin could be able to retrieve.

    ○ Actions: View

Figure 22: Actions to view the patient's data

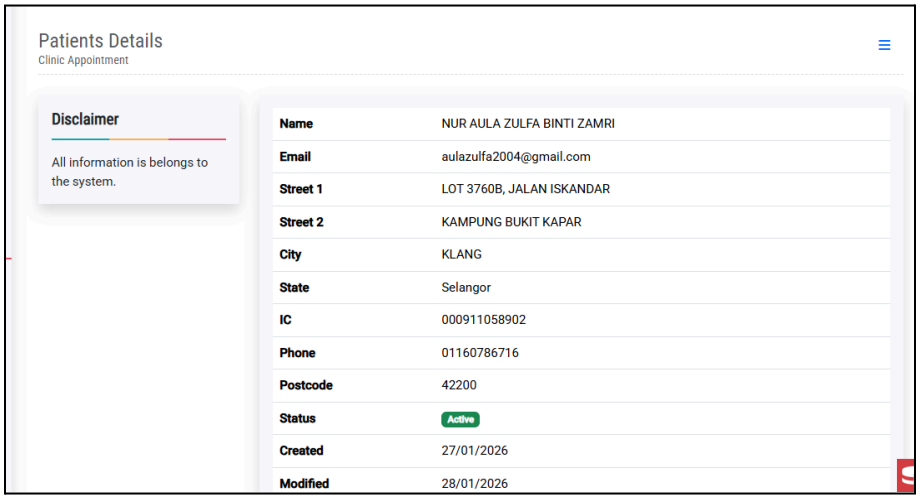

Figure 23: Patient's details

- Actions: Edit

Figure 24: Actions to edit the patient's data

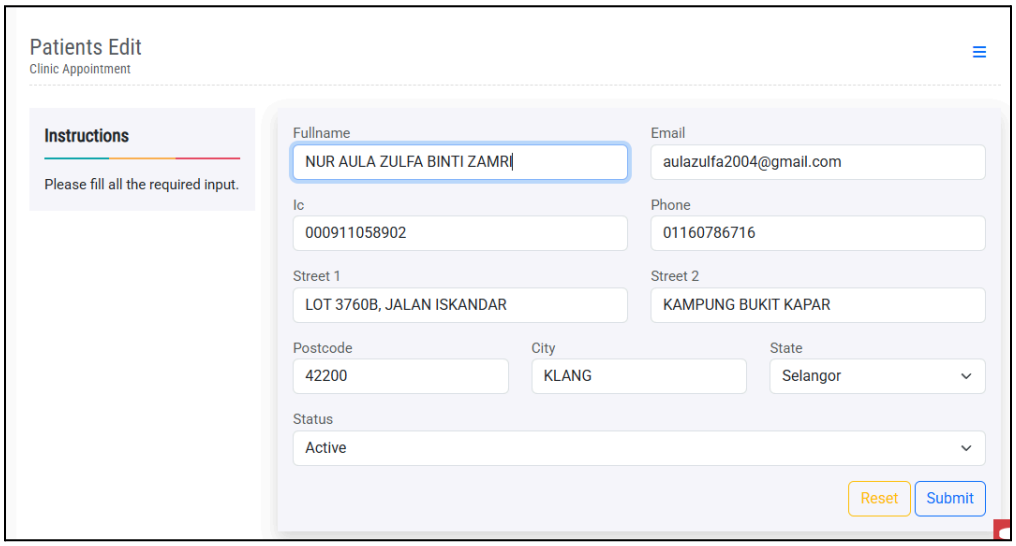

Figure 25: Form of patient's data to Edit

- Actions: Delete

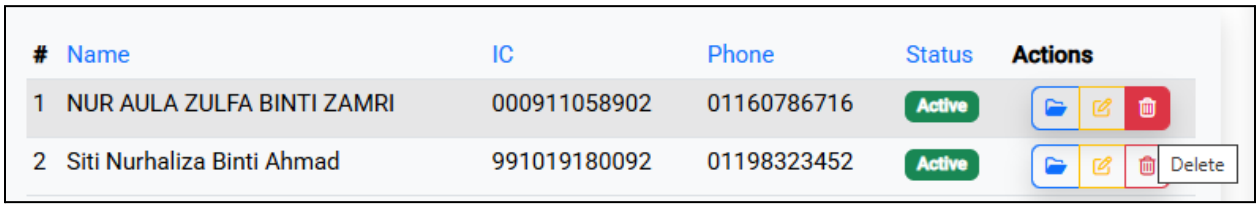

Figure 26: Actions to delete the Patient's data

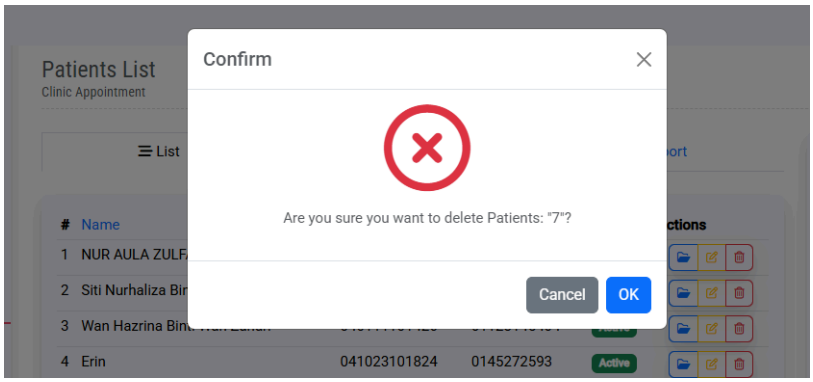

Figure 27: Confirmation to delete Pop-up message

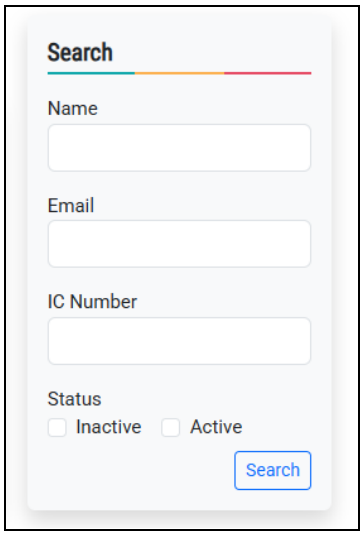

Figure 28: Search features to acquire patients' data

- These features enable the admin user to authorize the data easily by filtering the name, email, IC number, and the status of booking an appointment at the clinic.

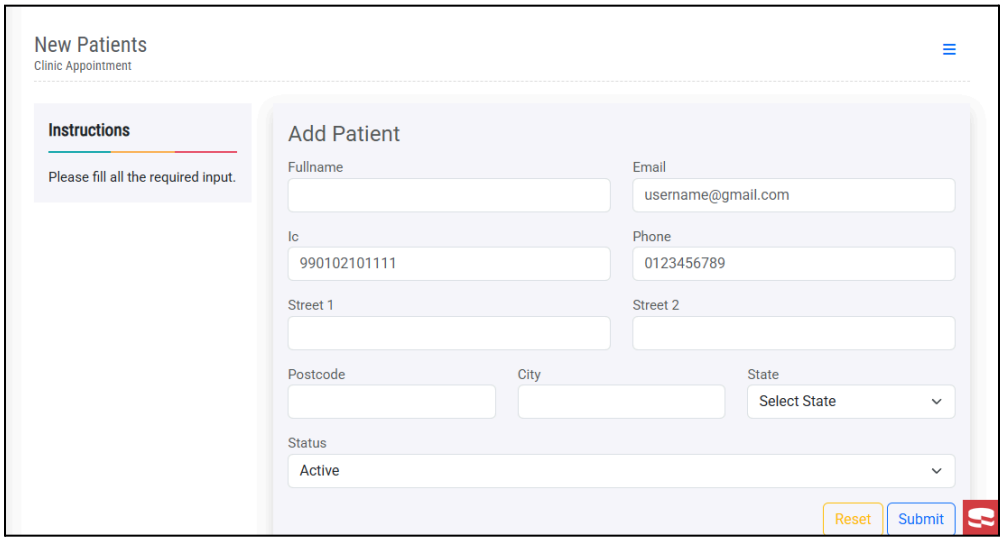

Figure 29: Features to add to the patient's data form

- These features allow admin to authorize any processes within the Clinic Booking System.

b. Appointment's section

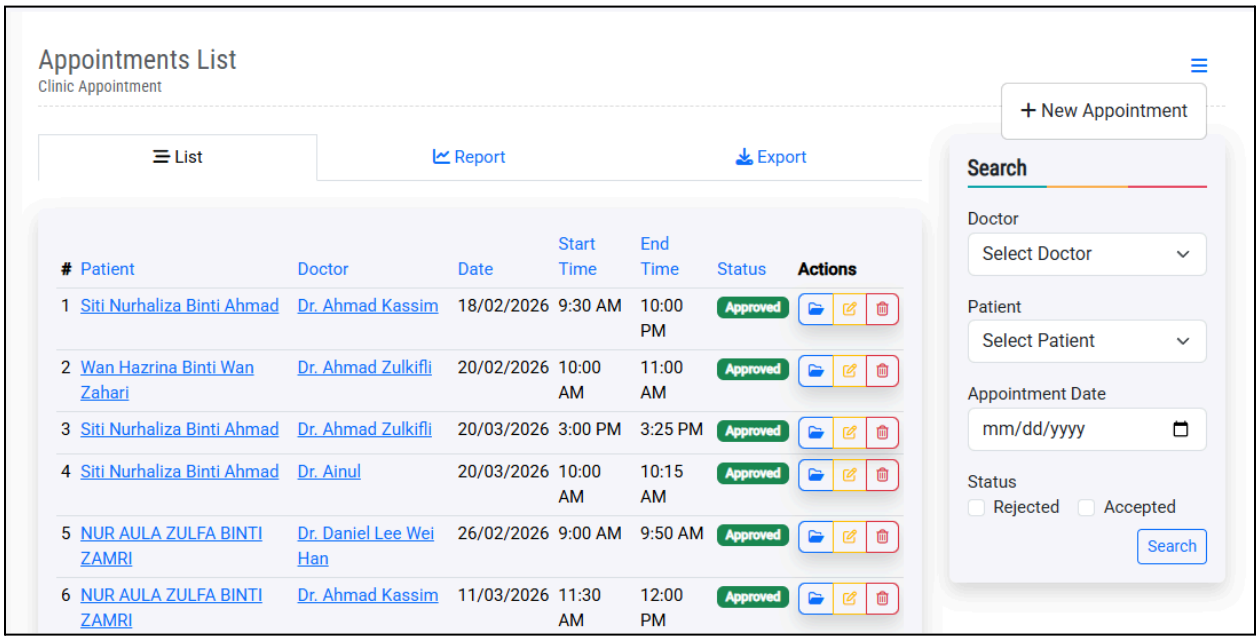

Figure 30: Features from the appointment interface

- On the Appointment list, follows the patient's name, the doctor that the admin assigned to, date of the appointment, the start time and end time of the session, and the status of approval.
- The features are the same as those of the Patient section, including the actions of viewing, editing, and deleting each of the appointment's data.
- Following the search engine, filtering by the assigned doctors, selecting the patient's name, appointment date, and the status of the appointment, whether it's being rejected or accepted.
- On the right side of the interface, the three lines above the search form are for adding new appointments for other patients.
- While the form of adding new appointments required the name of the patient, assigning the doctor, and setting the duration of the start and end time of the session, and fill the purpose with any medical reasons for doctors to review later.
- This feature by submitting the form the updated data of new appointments are added to the list.
- The appointments are listed in a structured format to ensure data manageability and neat organization.

c. Doctor’s section

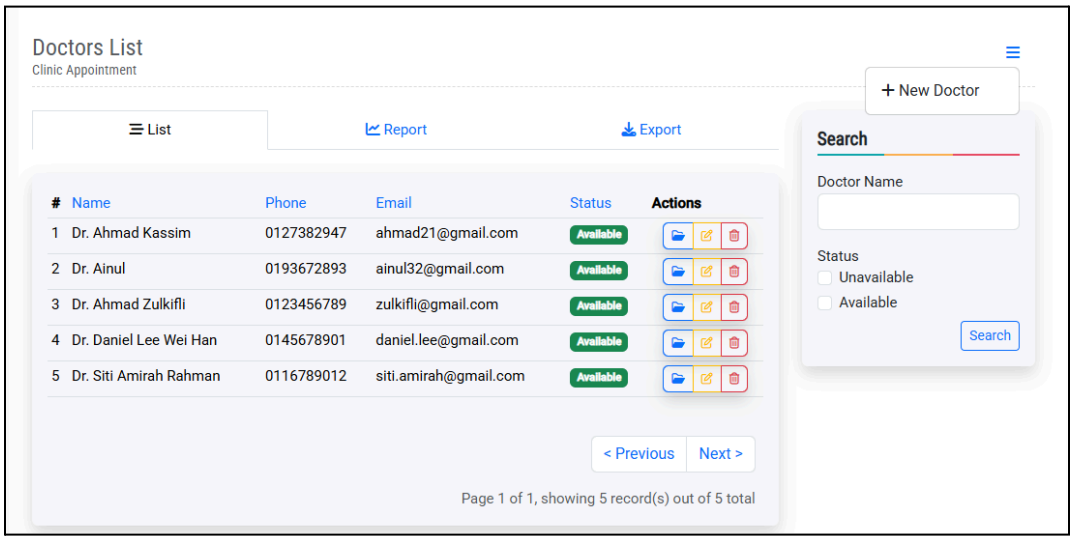

Figure 31: Features from the doctor interface

- On the Doctor's list follows the name, phone, email, and status of activity.
- Also, the same as the other two interfaces that include the actions, which combine the features of viewing, editing, and deleting the data from the list.
- The search features the name of the doctor and the status of the doctor’s activity. Filtering these requirements makes it easier for the admin to authorize.
- On the right side of the interface features to add new data to the doctor’s form.
- While the form requires the name, phone number, email, and their status of activity.
- These features are almost the same as those of others in keeping the data organized.

3. User Interface (Patient)

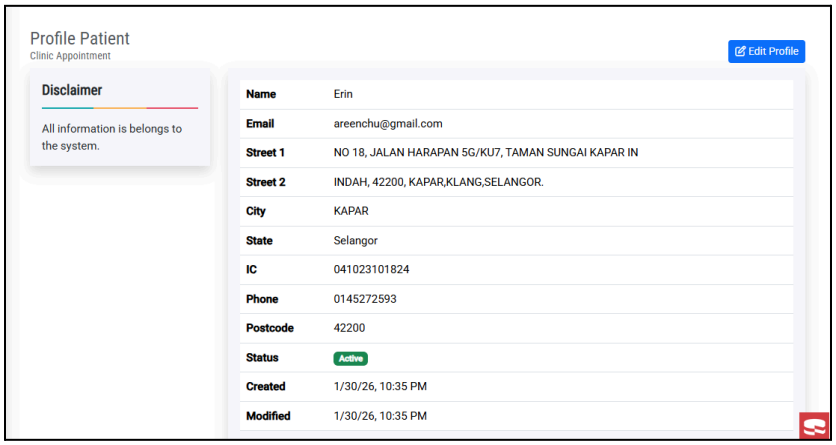

Figure 32: Features from the profile patient interface

- Users (Patient) could only view their profile on their page, this include all of the information above.
- There is an edit button for users (Patient) to update any information on their profile.

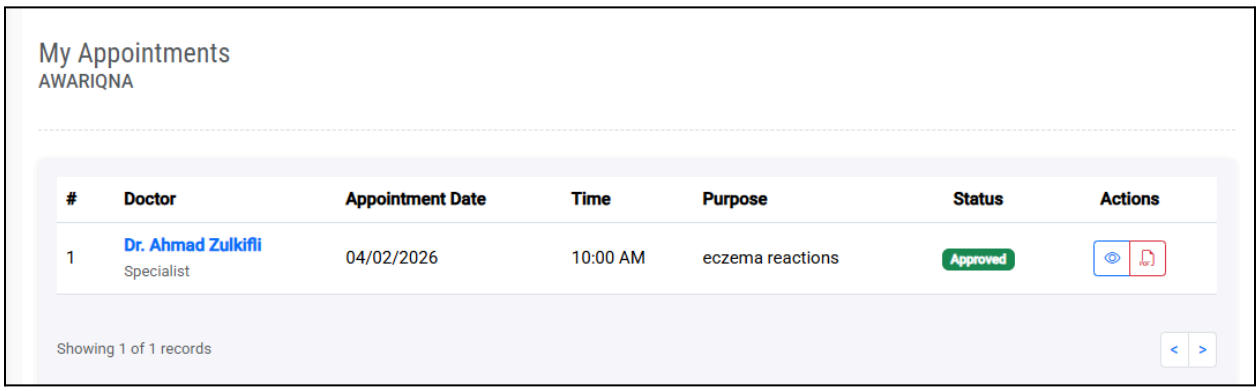

Figure 33: Features from the user's appointment interface

- The appointment that is authorized by admins, user could only view their appointment, only when the admin has approved through the status activity.
- The action features for the appointment list for users interface include the view and export through PDF.

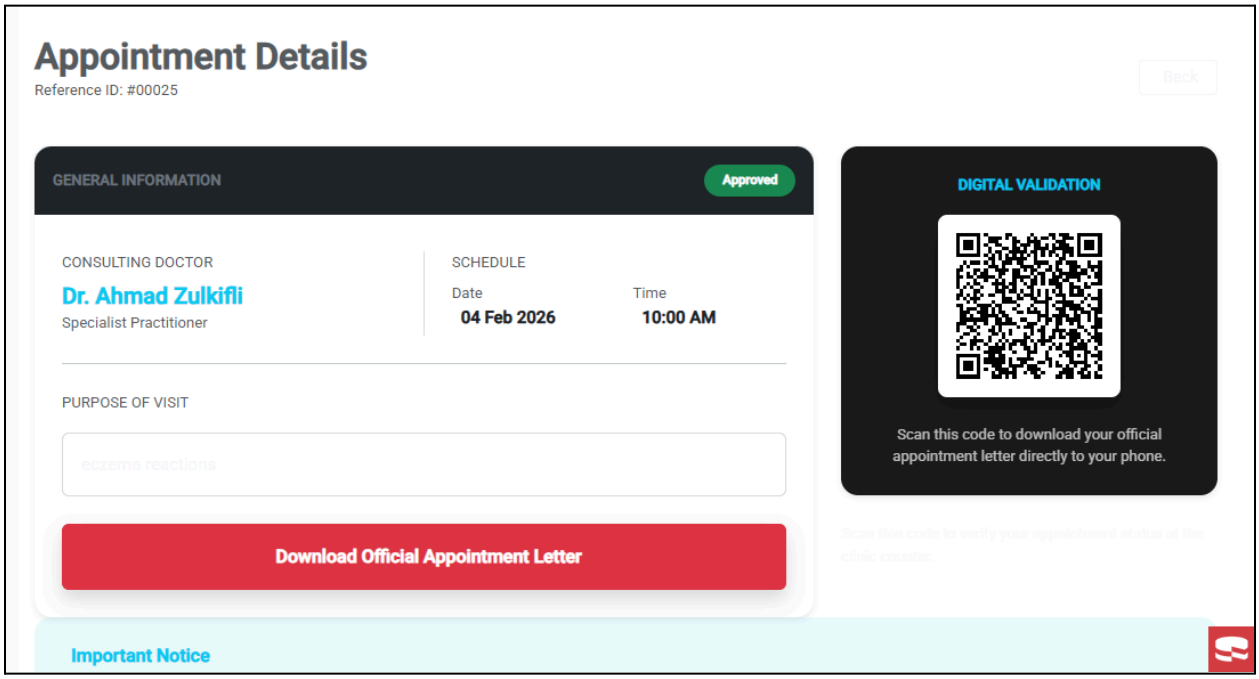

Figure 34: User (Patient) appointment view

- The interface explains the appointment for the user (Patient) and summarizes the whole data for the user to be aware.
- The appointment details could also be viewed from the QR code given or even download the official appointment letter. Featuring these features makes it easier for the user (Patient).

Figure 35: PDF of User (Patient) Appointment

6.0 WORKFLOW OF FORM

6.1 User (Patient) Workflow

The system uses Google OAuth to simplify access and automatically links the user to their profile.

- **Login (Google Auth):** The user authenticates via Google. The system creates a record in the users table.
- **Create (Profile Registration):** If it is a new user, they fill in the Patient Detail form (register.php). This creates a record in the patients table, automatically linked via user_id.
- **Read & Update (Profile Management):** Users can view and update their own patient details (index.php and view.php).
- **Read (Appointments):** Users view a list (index.php) of appointments created by the Admin and can view details (view.php) of their appointment. The system ensures only appointments matching their user_id are displayed.
- **Output (PDF & QR):** Users can generate a PDF version (pdf.php) of their appointment details and scan a QR Code.

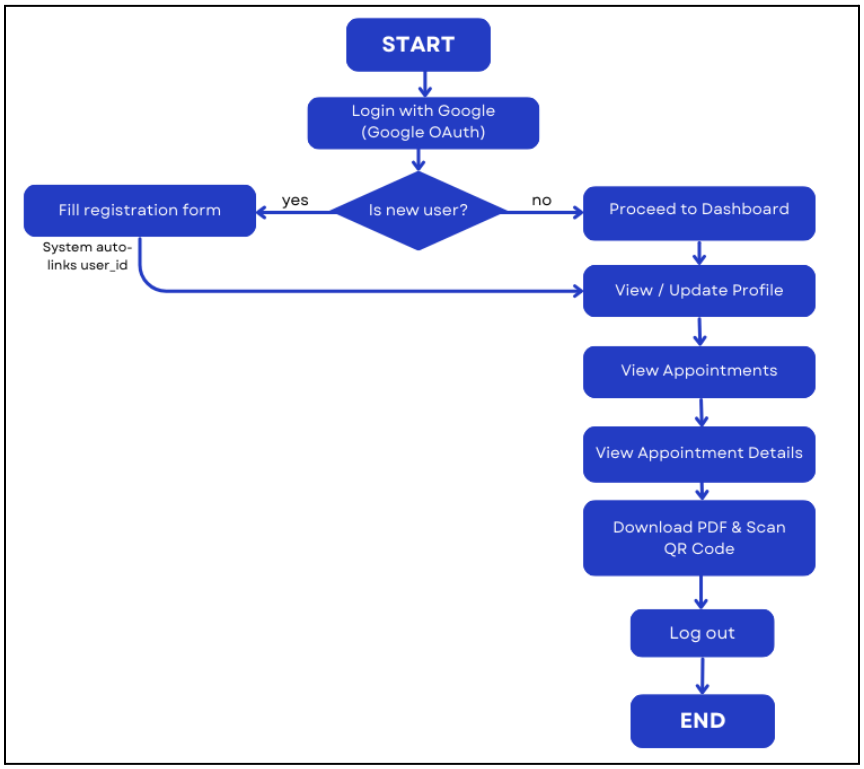

Figure 33: User Flowchart

6.2 Admin Workflow

The Admin flow represents the Management & Control and having full CRUD authority over all tables to ensure smooth clinic operations.

- **Login (Manual Auth):** Admin logs in using a secure username and password stored in the local database.

- Table Patient (CRUD):
    - **Create & Update:** Admin can register patients or update their profile records.
    - **Read:** Admin sees the list of all patients (index.php) and views the patient's details (view.php). Once a user registers their profile, their name automatically appears in this list.
    - **Delete:** Admin can remove invalid or duplicate records.

- Table Appointment (CRUD):
    - **Create & Update:** Admin creates appointments by selecting a registered Patient and a Doctor.
    - **Read:** The admin can view a list of all appointments (index.php) and view detailed information for each appointment (view.php). Once the admin creates an appointment for a user (patient), the appointment will automatically appear in the user's appointment list.
    - **Delete:** Admin can remove invalid or duplicate records.
    - The system uses **Auto-Mapping** logic to ensure the appointment is linked to the correct User account.

- Table Doctor (CRUD):
    - **Create & Update:** Admin manages the database of healthcare providers and their availability
    - **Read:** The admin can view a list of all doctors (index.php) and view detailed information for each doctor (view.php).
    - **Delete:** Admin can remove invalid or duplicate records.

7.0 TEAM ROLES

We distributed the system development tasks equitably among group members to ensure fair handling in building the Clinic Booking System.

7.1 Contributions

- Nur Aula Zulfa – Project Manager
- Nurin Afifah – Documentation
- Wan Hazrina – Backend Developer
- Nur Syahiqah – Frontend Developer
- ‘Ainul Wafa – System Analyst

7.2 Contact Information

- Nur Aula Zulfa – 01160786716
- Nurin Afifah – 0145272593
- Wan Hazrina – 01128145404
- Nur Syahiqah – 0138836438
- ‘Ainul Wafa – 01111498532

8.0 CONCLUSION & REFLECTION

8.1 Conclusion

In conclusion, Awariqna Clinic System was successfully developed to manage lists of doctors, patients, appointments records in fully digital format. The system enables administrations, doctors and patients to access and manage their information efficiently any time and anywhere especially for doctors and patients.

Moreover, the system is designed with a user-friendly interface that is simple, clear and easy to use and understand. It is also responsive which can be used in every device without having a hard time or unnecessary effort. The system layout supports mobile layout. With this, patients can use the system without confusion, even for first-time users. The intuitive layout and straightforward workflow help users complete tasks such as viewing appointment letters and updating profiles smoothly.

Furthermore, Awariqna Clinic follows the trend with the technology growth nowadays in their business organisation. This helps the system improve data accuracy, reduce manual record handling and enhance overall operational efficiency.

8.2 Reflection

Throughout the development of Awariqna Clinic System, we gained a better understanding about how the system workflow operates. We learned how controllers process user (patient) requests, how views display the data and how information flows between them.

Managing data flow between the controller and view was more challenging than expected. It made us face issues where data does not appear after submission which helped us understand the importance of proper request handling. The updated data has to be read again and that allows patients to view their latest information such as updated personal details without the data disappearing.

9.0 REFERENCES

Link Repository GitHub: https://github.com/AulaZulfa/Klinik_Awariqna_IMS566.git

Backup Link Google Drive:

https://drive.google.com/drive/folders/1C-hOULkSNeLSaSQbbMMqsNIunJ6WA1gf?usp=drive_link