

Docease Simplifying Healthcare Navigation Through an Intuitive Appointment Web Solution

Mohamed Abdiraman Abdirahim¹, Abd Samad Hasan Basari^{1*}

¹ *Fakulti Sains Komputer dan Teknologi Maklumat,*

Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

*Corresponding Author: abdsamad@uthm.edu.my

DOI: <https://doi.org/10.30880/aitcs.2025.06.01.110>

Article Info

Received: 7 January 2025

Accepted: 18 June 2025

Available online: 30 June 2025

Keywords

Doctor appointment system, web-based, agile methodology, Pusat Kesihatan Universiti, patient satisfaction, healthcare management, telemedicine integration

Abstract

The integration of digital technology has substantially enhanced patient care and operational efficiency in healthcare; nonetheless, traditional appointment scheduling continues to be inefficient, characterized by prolonged wait times and ambiguous availability. This research is to create a Doctor Appointment Web System for Pusat Kesihatan Universiti (PKU) at Universiti Tun Hussein Onn Malaysia, emphasizing real-time scheduling, doctor profiles, and user-friendly interfaces. An agile methodology prioritizing iterative development and ongoing testing was adopted to guarantee system stability and adaptability. The results include decreased waiting periods, increased healthcare accessibility, and better patient satisfaction, alongside improved telemedicine and resource distribution. These findings highlight the capacity of digital technologies to enhance healthcare procedures and strengthen patient-provider relationships, with further efforts recommended to incorporate capabilities such as predictive analytics and integration with comprehensive health systems.

1. Introduction

The healthcare sector has progressively embraced digital technology to enhance both patient care and operational efficiencies. The healthcare sector has swiftly embraced digital technologies, enhancing patient care and accessibility. This advancement is driven by the transformative doctor appointment web-based system [1], which simplifies the scheduling of medical appointments. This extensive online platform links patients with healthcare professionals for smooth appointment booking.

This system aims to overcome the inherent constraints of traditional appointment booking systems, which are marked by inefficient phone-based procedures that frequently lead to prolonged waiting periods and a lack of clarity regarding the availability and qualifications of healthcare providers. These inefficiencies not only cause frustration for patients but also place a heavy load on healthcare staff, diminishing the quality of treatment they can deliver.

The primary motivation for this project arises from the necessity to improve the interaction between patients and healthcare providers by implementing an efficient, timely scheduling system. The system's objective is to enhance the operational dynamics within healthcare institutions by using digital technology [2]. This will be achieved by offering a user-friendly platform for booking appointments, rapid access to healthcare professional profiles, and efficient management of medical records.

This project employs an agile development method, with a primary focus on continuous testing and iterative feedback, to improve and enhance the web system. The applied methodologies include prototyping, system analysis, and user acceptability testing to guarantee that the platform fulfils the requirements of all stakeholders, including patients, doctors, and staff.

The anticipated results of this study include decreased patient waiting times, extended accessibility to healthcare services, and increased user satisfaction ratings. These improvements are expected to promote a more simplified and patient-focused healthcare setting.

2. Related Work

This section discusses the related works on patient management systems, the implementation of a web-based system for Pusat Kesehatan Universiti (PKU), and a comparison between existing systems and the proposed Docease system to highlight its innovative contributions.

2.1 Patient Management Systems

Patient management systems are essential for enhancing administrative and clinical operations in hospitals [3]. They include features such as electronic health records (EHRs), appointment scheduling, and follow-up treatment, which work together to increase patient satisfaction and outcomes by replacing outmoded manual processes with efficient digital alternatives.

2.1.1 Patient Registration and Record Maintenance

The transition to electronic health records has considerably improved patient information management by providing more accuracy, accessibility, and security. EHRs enable quick access to detailed medical histories, eliminate data input mistakes, and link several clinical systems, thereby improving decision-making and coordination among healthcare providers.

2.1.2 Scheduling Appointments

Appointment scheduling has been changed by digital patient management systems, which now allow for online bookings, modifications, and cancellations. These solutions reduce no-show rates and administrative workload with features such as real-time updates and automatic reminders, enhancing overall efficiency and patient experience.

2.2 Web-Based System

The proposed system is a web-based application designed for accessibility via web browsers, catering to patients, healthcare providers, and administrative staff at Pusat Kesehatan Universiti (PKU). Users can securely access the system without installations, allowing for appointment bookings, medical record access, and real-time updates [4]. By replacing manual processes, the system reduces inefficiencies and administrative burdens while ensuring centralized, cost-effective, and user-friendly healthcare management. This aligns with the project's goals of enhancing efficiency and improving patient satisfaction at PKU.

2.3 Pusat Kesehatan Universiti Manual System

The current system at Pusat Kesehatan Universiti (PKU) relies heavily on manual processes for managing patient appointments, records, and administrative tasks. Patients must visit the health center in person or rely on phone calls to schedule appointments, which often leads to long waiting times and inefficiencies. Medical staff manually record patient information, update records, and manage appointments, increasing the risk of human errors and administrative burdens. This paper-based system lacks integration and accessibility, making it difficult to track patient histories or streamline healthcare services. These challenges highlight the need for a digital solution to enhance operational efficiency, reduce errors, and improve the overall patient experience.

2.4 Study of the existing system

A study was conducted on three existing healthcare management systems: Hydras Medical Sdn Bhd (Mediklinik Mikhail), Rumah Sihat Kita, and Sunway Medical Centre. This analysis identifies their strengths and limitations to guide the development of the proposed Docease system. While the existing systems offer features like appointment booking and help modules, they lack certain functionalities such as external link modules, account management, and advanced patient feedback mechanisms.

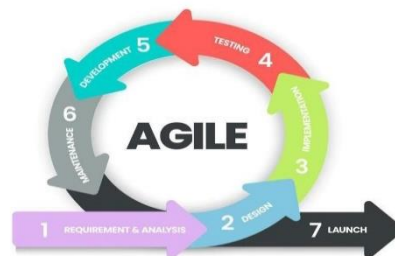
The proposed system addresses these gaps by integrating comprehensive features to enhance patient care and operational efficiency. The comparison is summarized in the table below which highlights the differences between the existing systems and the proposed solution.

Table 1 Comparison of existing systems with the proposed system

Features	Hydras Medical Sdn Bhd	Rumah Sihat Kita	Sunway Medical Centre	Proposed System
Patient Record Search	✓	✗	✗	✓
Informational Module	✓	✓	✓	✓
Login Module	✓	✗	✗	✓
Account Management Module	✗	✗	✗	✓
External Link Module	✗	✗	✗	✓
Booking Appointment Module	✓	✓	✓	✓
Help Module	✓	✓	✓	✓
Patient Feedback System	✓	✓	✓	✓

3. Methodology

The Agile methodology was selected as the project development model for the Docease system. Agile allows for managing the project by breaking it down into iterative stages, ensuring flexibility and adaptability throughout the development process. This approach involves continuous collaboration with stakeholders and iterative improvements across all phases of the system development life cycle, including brainstorming, system design, development, quality assurance, and system deployment [5], as shown in the figure below.

**Fig. 1** Agile Methodology

The Agile methodology provides a structured yet flexible approach to project management. Each phase is iterative, enabling developers to respond to feedback and evolving requirements. This iterative nature ensures the continuous refinement of the system, fostering efficient communication and collaboration between developers

and stakeholders [6]. The emphasis on adaptability makes Agile ideal for the development of the Docease system, as shown in the table below.

Table 2 System Development Workflow

Phase	Task	Output
Brainstorming	<ul style="list-style-type: none"> Proposed the project Determine the project schedule, activities and output 	<ul style="list-style-type: none"> Proposal Gantt Chart Literature Review
Design	<ul style="list-style-type: none"> Establish User Interface Clarify how Pusat Kesehatan Universiti operates Establishing the connection between entities and the database 	<ul style="list-style-type: none"> Design wireframe Context Diagram Data Flow Diagram Entity Relationship Diagram Database design Define software and hardware requirements
Development	<ul style="list-style-type: none"> Front-end development including its entirety Back-end development through its entirety 	<ul style="list-style-type: none"> Easy to use by user Database system
Quality Assurance	<ul style="list-style-type: none"> White box testing Black box testing 	<ul style="list-style-type: none"> The system is bug-tested by the developer and the intended user. Questionnaire feedback from the target user
Deployment	<ul style="list-style-type: none"> Deploy System to Pusat Kesehatan Universiti 	<ul style="list-style-type: none"> Both the developer and the intended user are pleased with the system that has been created. User guide

4. Analysis and Design

This section focuses on the analysis and design process for Pusat Kesehatan Universiti (PKU), including Data Flow Diagrams (DFD), Entity Relationship Diagrams (ERD), and system interfaces. It outlines the methodologies used to develop an efficient and user-friendly system that integrates IT, processes, and data to meet healthcare requirements. Additionally, it emphasizes the use of diagrams to demonstrate key system functions and the interaction between modules and system users.

4.1 DFD Context Diagram (DFD CD)

The Context Diagram provides a high-level view of the system, showcasing its interactions with external entities such as administrators, doctors, staff, and patients. It highlights the system's scope, relationships, and data flow without requiring technical expertise. The diagram illustrates processes like login authentication for users, doctors updating patient information, staff managing medication, and patients scheduling appointments and reviewing booking history, offering a clear understanding of the system's functionality and interactions.

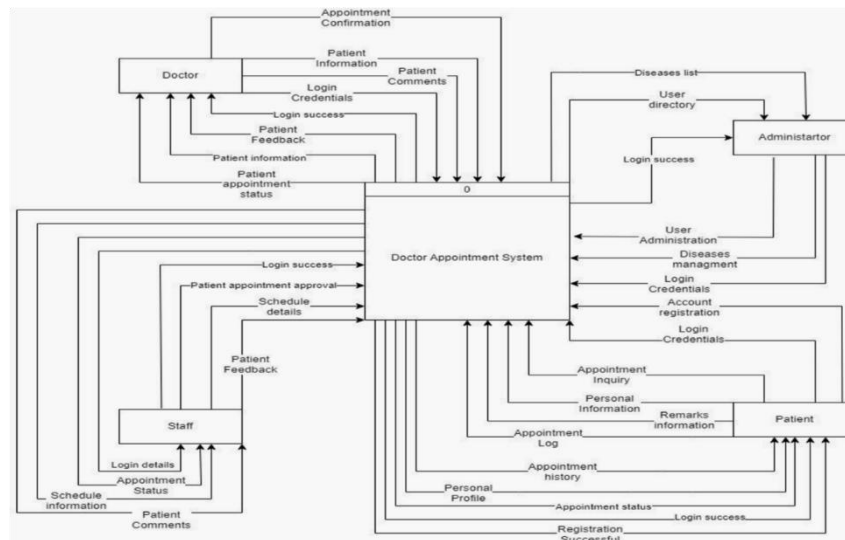


Fig. 2 Data Flow Diagram Context Diagram (DFD CD)

4.2 DFD Level 0

The Level 0 Data Flow Diagram (DFD) highlights key processes, including user and service management and appointment scheduling. Administrators manage user accounts, medical services, and system data, while patients and doctors interact for scheduling and care. The system improves efficiency by optimizing booking and enhancing overall service quality [7].

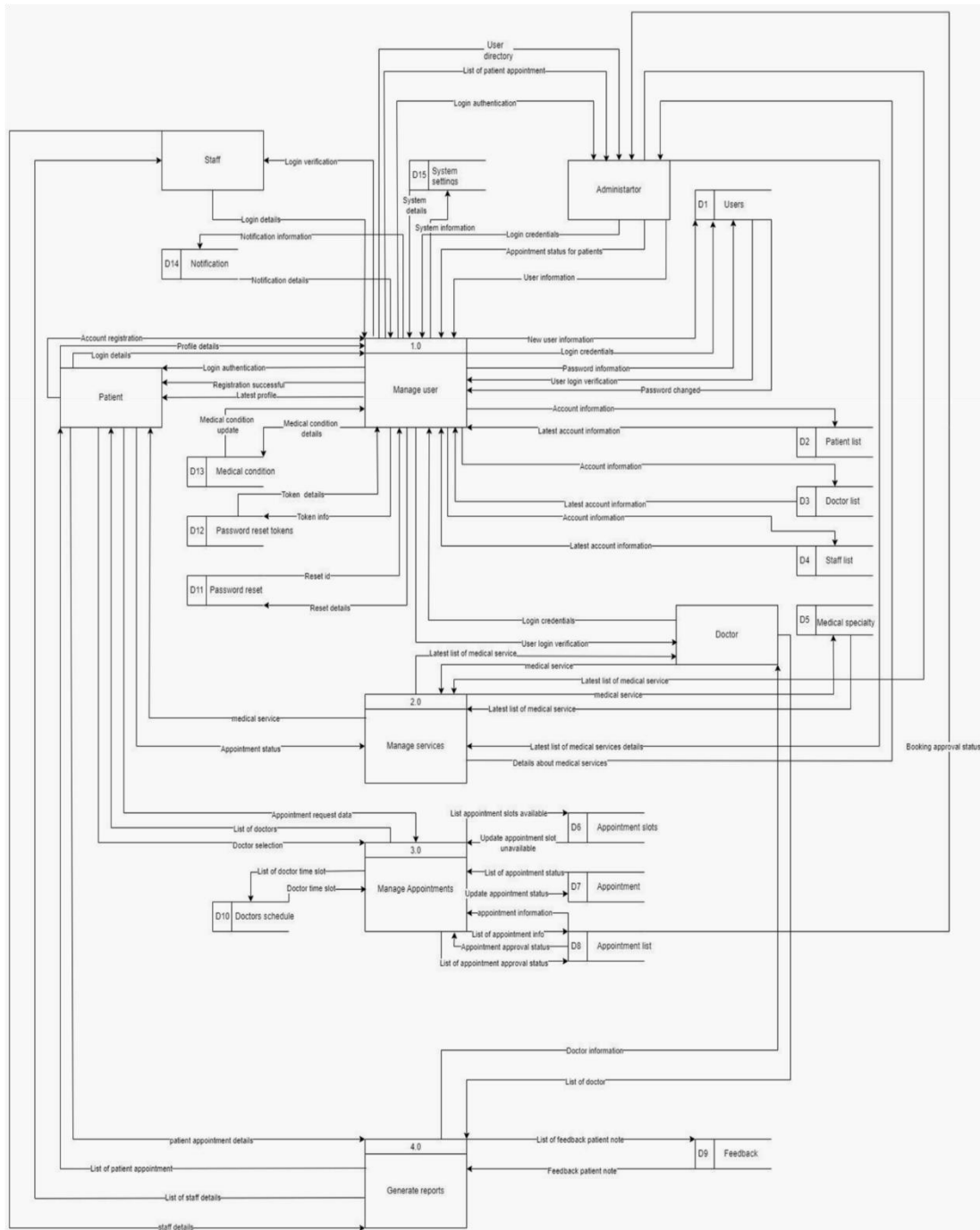


Fig. 3 Data Flow Diagram (DFD) Level 0

4.3 Entity Relationship Diagram (ERD)

The Entity-Relationship Diagram (ERD) illustrates the relationships between key entities in the system [8], including users, patients, doctors, appointments, and feedback. Users' data links to system settings, while patients' information connects to appointments and feedback. Doctors' data integrates with their schedules and specialties, and the appointments entity manages scheduling and links patients with doctors. Additional components like appointment slots, system settings, and staff support efficient system administration and data retrieval.

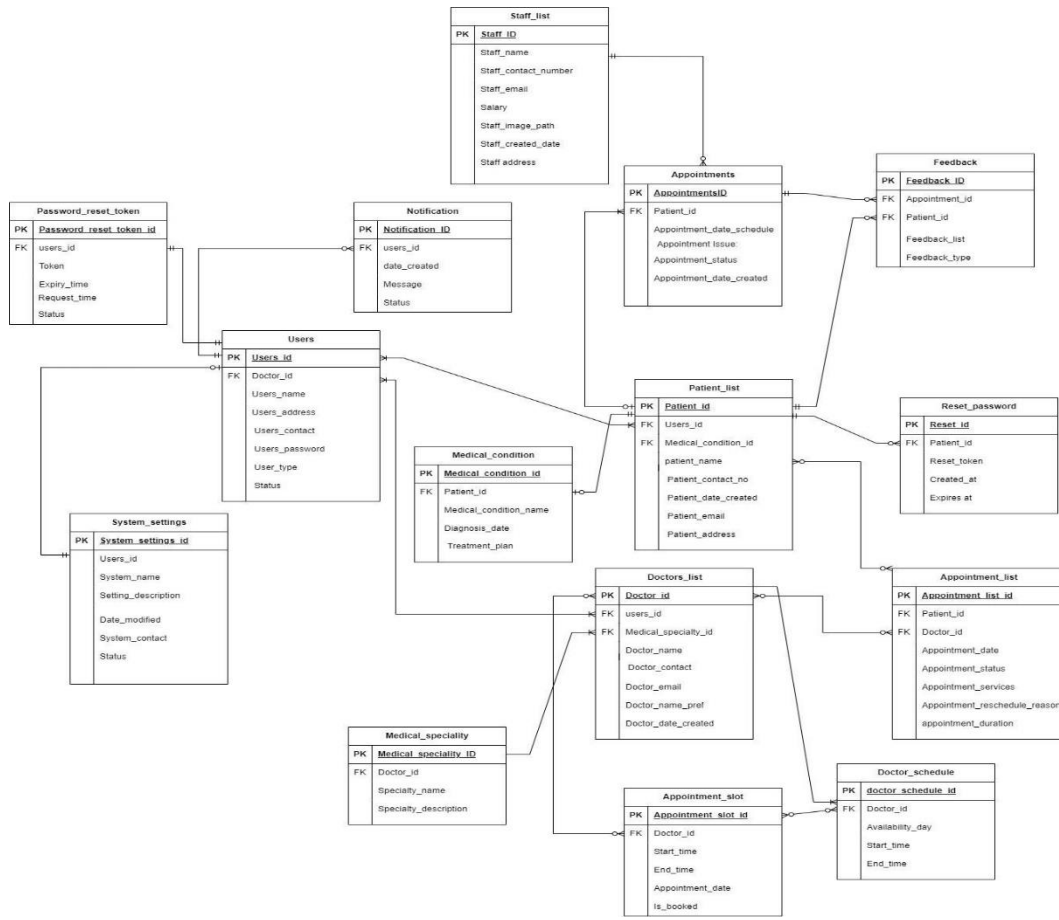


Fig. 4 Entity Relationship Diagram (ERD)

4.4 Interface Design

Interface design, also known as prototype design, encompasses interfaces that the viewer will interact with. The system interfaces, as shown below, are designed for user-friendly interaction tailored to specific roles. Fig. 5 Login Page provides secure access for users, doctors, and administrators with email and password fields. Fig. 6 Register Page enables new patients to create accounts via a simple form. Fig. 7 Patient Page offers a personalized platform for managing appointments and account details. Fig. 8 Appointment Page features a calendar interface for booking consultations, displaying doctor details and available time slots. Fig. 9 Admin Page serves as a centralized dashboard for administrators to oversee appointments, departments, and system metrics. Lastly, Fig. 10 Doctor Page provides doctors with an efficient dashboard to manage appointments, patients, and schedules.

Login

Email Address

Password

Remember Me

Don't have an account? [Register Now.](#)

Fig. 5 Login Page

Register

Name

Email Address

Password

Confirm Password

Already have an account? [Login Now](#)

Fig. 6 Register Page

Hello [abdurrahman](#).

Quick Links

- [My Appointments](#)
- [My Profile](#)
- [Change Password](#)
- [Support](#)
- [Log Out](#)

My Appointments

Apt. No.	Apt. Date	Apt. Time	Status	Action

Fig. 7 Patient Page



Fig. 8 Appointment Page



Fig. 9 Admin Page

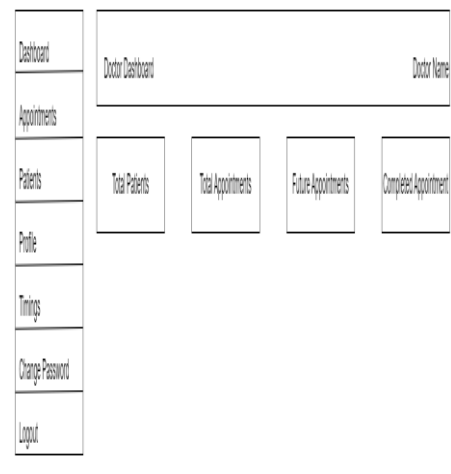


Fig. 10 Doctor Page

5. Results and Discussion

This section describes the development and evaluation of the Doctor Appointment System (DOCEASE). Built using PHP (Laravel), MySQL, and tools like Visual Studio Code, it is hosted on Hostinger for accessibility. Testing processes, including unit, integration, and user acceptance testing (UAT), ensured reliability, performance, and functionality while incorporating feedback from users to enhance the system.

5.1 Implementation

The implementation of the web system for the University Health Centre (PKU) integrates multiple functional modules with relational database architecture. Built using PHP, HTML, CSS, JavaScript, jQuery, and AJAX, the system development utilizes Visual Studio Code for coding and debugging. phpMyAdmin and MySQL handle database management, with JSON files supporting dependency management and version control. Shell scripts facilitate deployment and testing [9]. The implementation improves efficiency for staff, doctors, and patients by optimizing appointment scheduling and administrative processes.

Fig. 11 Login Page allows users, including patients, doctors, and administrators, to securely access the web system. It includes fields for email and password, a "Remember Me" checkbox for session retention, and a "Forgot Your Password" link for account recovery. New users are directed to the registration page via the "Register Now" link. Fig. 12 Register Page enables new users to create accounts by entering their name, email address, and password, with a confirmation field to ensure accuracy. A "Register" button submits the form, and an "Already have an account? Login Now" link redirects existing users to the login page.

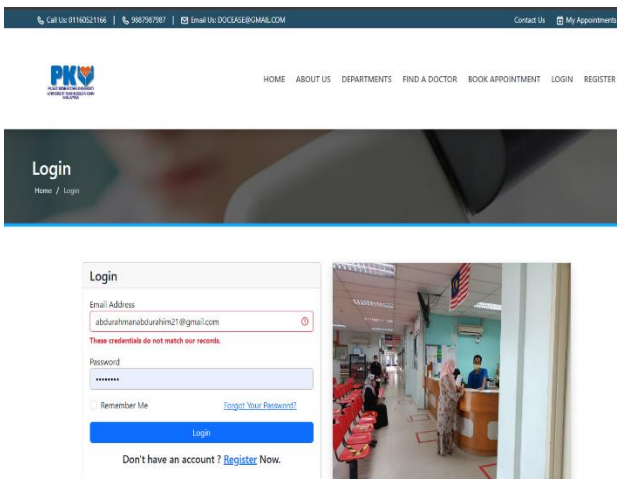


Fig. 11 Login Interface

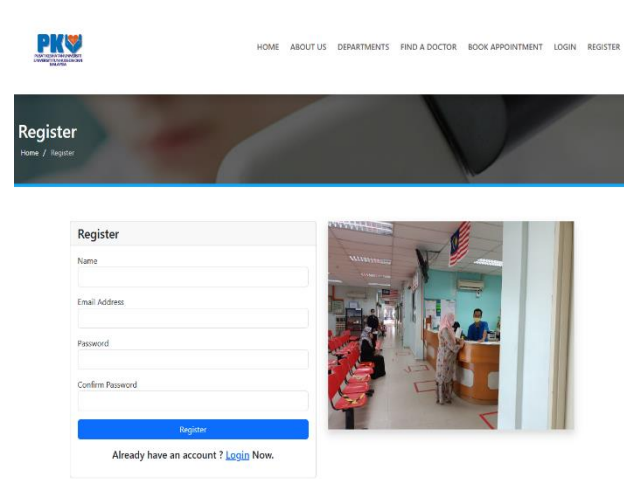


Fig. 12 Register Interface

Fig. 13 Login Functionality defines the process for authenticating users and redirecting them to their respective dashboards based on their roles, such as admin, doctor, or patient. It ensures proper session management and provides feedback upon successful login. Fig. 14 Registration Functionality implements a user registration form with fields for name, email, password, and password confirmation. It validates inputs, displays error messages for incorrect data, and submits the form to create new accounts, ensuring accuracy and enhancing the user experience.

```

21 use AuthenticatedUsers;
22
23 /**
24  * Where to redirect users after login.
25  */
26 * @var string
27 *
28 // protected $redirectTo = RouteServiceProvider::HOME;
29 // references [0 overrides] [Admin] [Edit] [Test] [Explain] [Document] [Ask]
30 public function authenticated(): mixed { RedirectResponse
31 {
32     if (Auth::user()->role_as == 'admin'){
33         return redirect(to: '/admin/dashboard')->with(key: 'status', value: 'Welcome to Dashboard');
34     }elseif (Auth::user()->role_as == 'doctor'){
35         return redirect(to: '/doctor/dashboard')->with(key: 'status', value: 'Welcome to Dashboard');
36     }elseif (Auth::user()->role_as == 'user'){
37         return redirect(to: '/user/appointments')->with(key: 'status', value: 'Logged In Successfully');
38     }else{
39         return redirect(to: '/')->with(key: 'status', value: 'Logged In Successfully');
40     }
41 }
42 }
43 /**
44  * Create a new controller instance.
45  */
46 * @return void
47 *
48 // references [0 overrides] [Admin] [Edit] [Test] [Explain] [Document] [Ask]
49 public function __construct()
50 {
51     $this->middleware(middleware: 'guest')->except(methods: ['logout']);
52 }
53 }
54 }
55 }
56 }
57 }
58 }
    
```

Fig. 13 Login Functionality

```

100 // references [0 overrides] [Admin] [Edit] [Test] [Explain] [Document] [Ask]
101 public function register(): mixed { RedirectResponse
102 {
103     // validation rules
104     $rules = [
105         'name' => 'required|string|max:255',
106         'email' => 'required|email|max:255|unique:users',
107         'password' => 'required|string|min:8|confirmed',
108     ];
109     $validator = Validator::make($request->all(), $rules);
110     if ($validator->fails()) {
111         return back()->withErrors($validator)->only(['email', 'password']);
112     }
113     // create user
114     $user = User::create([
115         'name' => $request->name,
116         'email' => $request->email,
117         'password' => $request->password,
118     ]);
119     // login user
120     $login = $this->login($request->email, $request->password);
121     // redirect to dashboard
122     return redirect($login->redirectTo)->with('status', 'Registration Successful');
123 }
124 }
125 }
126 }
127 }
128 }
129 }
130 }
131 }
132 }
133 }
134 }
135 }
136 }
137 }
138 }
139 }
140 }
141 }
142 }
143 }
144 }
145 }
146 }
147 }
148 }
149 }
150 }
151 }
152 }
153 }
154 }
155 }
156 }
157 }
158 }
159 }
160 }
161 }
162 }
163 }
164 }
165 }
166 }
167 }
168 }
169 }
170 }
171 }
172 }
173 }
174 }
175 }
176 }
177 }
178 }
179 }
180 }
181 }
182 }
183 }
184 }
185 }
186 }
187 }
188 }
189 }
190 }
191 }
192 }
193 }
194 }
195 }
196 }
197 }
198 }
199 }
200 }
    
```

Fig. 14 Registration Functionality

5.1.1 User Dashboard Interfaces

The User Dashboard Interfaces serve as the central hub for interaction within the system, tailored to meet the needs of different user roles, including patients, administrators, and doctors. These interfaces provide role-specific functionalities, allowing users to efficiently manage appointments, access key information, and perform their tasks seamlessly. Each dashboard is designed to enhance user experience and improve the overall efficiency of the healthcare management system.

The main pages for patients, administrators, and doctors are designed for efficient user experience. Fig. 15 Patient Main Page allows patients to view and manage their appointments, featuring a personalized greeting, quick navigation links, and a summary of appointment details.

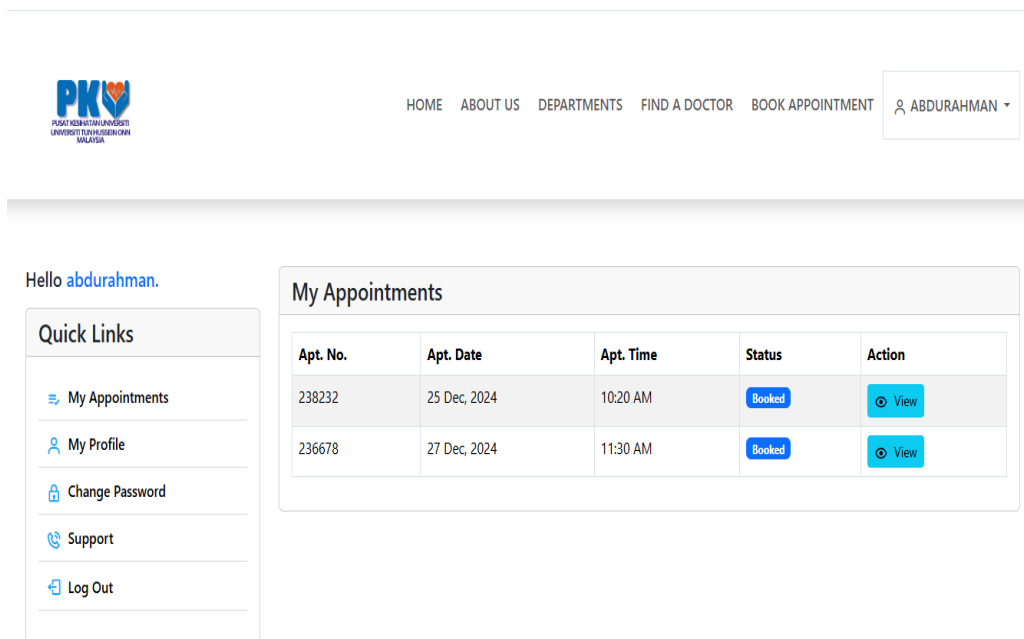


Fig. 15 Patient Main Page

Fig. 16 Admin Main Page provides administrators with a centralized dashboard to oversee appointments, departments, doctors, and patients, displaying key metrics for effective system management.

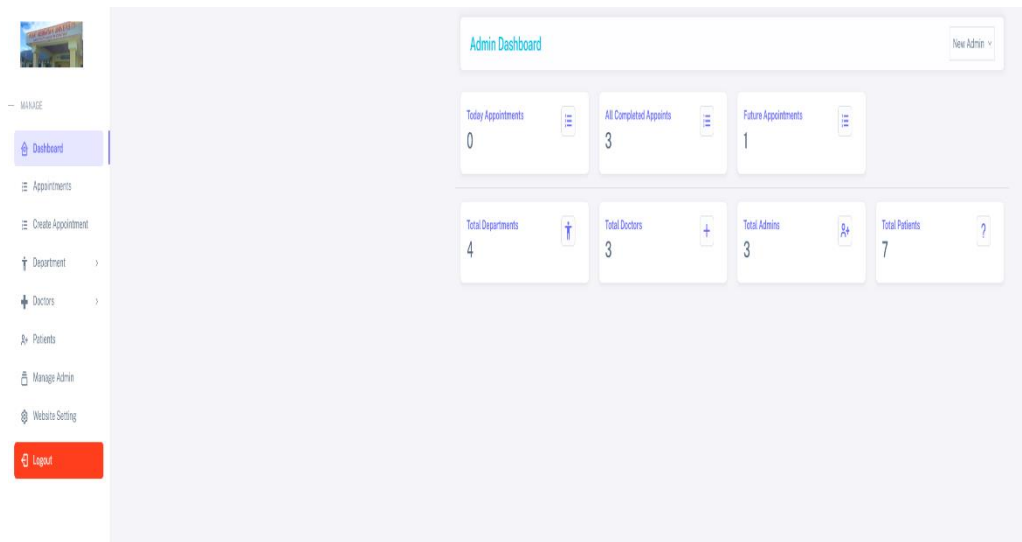


Fig. 16 Admin Main Page

Fig. 17 Doctor Main Page offers doctors an overview of their appointments and patients, highlighting metrics such as total and future appointments, enabling effective workload management.

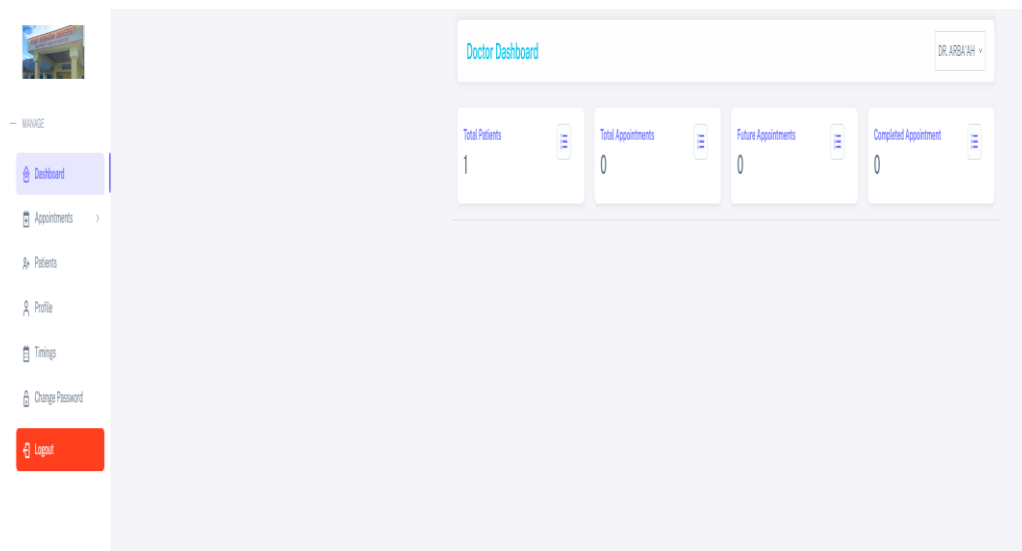


Fig. 17 Doctor Main Page

5.1.2 Appointment Booking Interface

The Appointment Booking Interface is designed to simplify the process of scheduling appointments for patients. It provides a platform to select doctors, view available time slots, and confirm bookings, ensuring a user-friendly experience.

As shown below in Fig. 18, this interface displays detailed information about the selected doctor, including qualifications, specialization, and consultation fees. A calendar-like scheduling section allows patients to view available dates and time slots for appointments. Available and unavailable slots are clearly highlighted, helping patients make informed decisions when booking.

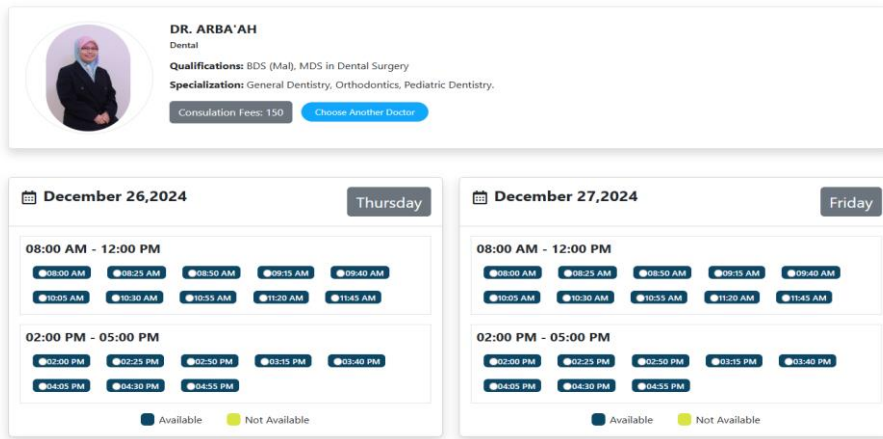


Fig. 18 Doctor Selection and Scheduling

The Slot Confirmation popup in Fig. 19 appears after a patient selects an appointment slot. Patients are prompted to confirm their booking by entering their name, email, and phone number. This step ensures that all necessary details are accurately recorded before finalizing the appointment.

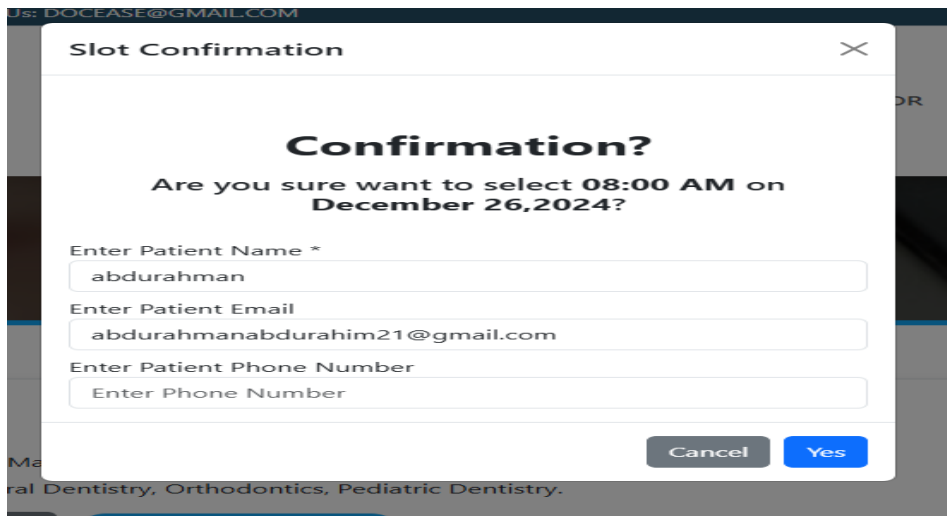


Fig. 19 Slot Confirmation

In Fig. 20 this section provides a detailed view of the available and unavailable time slots for a specific day. Each slot is clearly marked as "Available" or "Not Available," allowing patients to easily choose the most convenient time for their appointment. This feature ensures a smooth and user-friendly booking process.

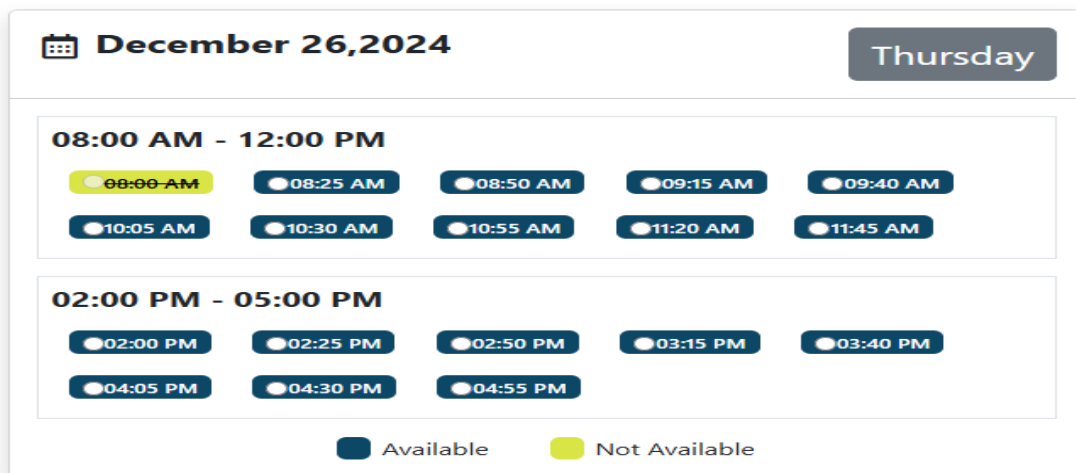


Fig. 20 Time Slot Availability

5.1.3 Password Reset Functionality

This section details the password reset functionality in the DOCEASE system, designed to help users securely regain access to their accounts when login credentials are forgotten. Implemented using Laravel's authentication system, it provides a token-based password reset process, ensuring security and usability [10]. The workflow includes sending password reset emails, validating reset tokens, and securely updating user passwords through an intuitive interface. Supporting features like confirmation messages, predefined language files, and user-friendly email templates enhance the overall accessibility and effectiveness of the password reset process.

Fig. 27 Reset Password Form View shows the interface where users can initiate the password reset process by entering their registered email address. Upon submission, a success message confirms that a password reset link has been sent, guiding the user to check their email. Fig. 28 Reset Password Controller handles the backend functionality for resetting passwords. Using Laravel's ResetsPasswords trait, it manages token validation, updates passwords securely, and redirects users to the appropriate dashboard after a successful reset, ensuring a smooth and secure process.

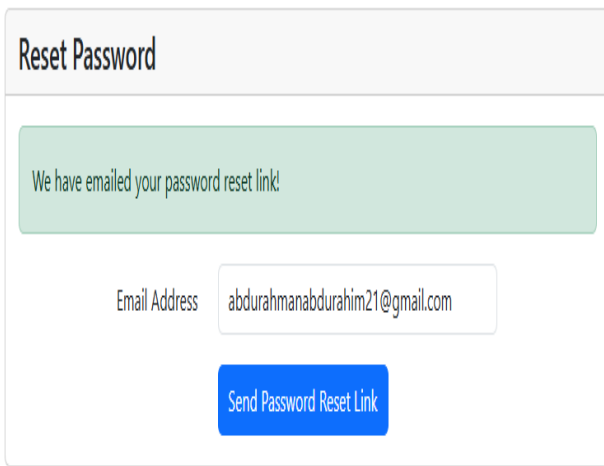


Fig. 27 Reset Password Form View

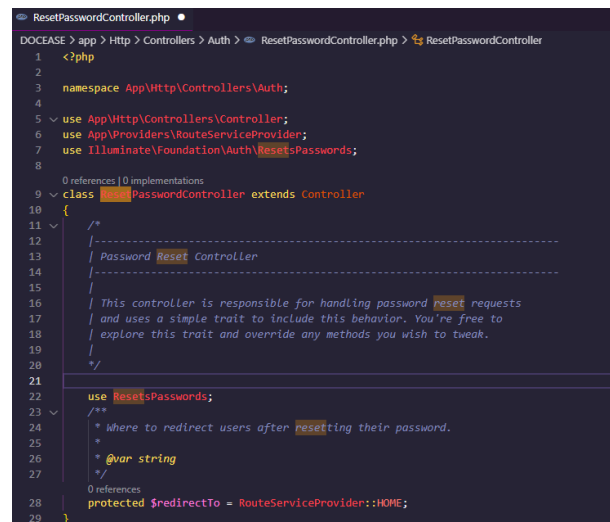


Fig. 28 Reset Password Controller

Fig. 29 Password Reset Form View provides users with a secure interface to reset their password after clicking the reset link sent to their email. The form includes fields for the user's email address, new password, and password confirmation to ensure accurate input. Upon submission, the system securely updates the password in the database. Fig. 30 Password Reset Language File contains predefined messages used throughout the password reset process, such as success notifications like "Your password has been reset!" and error alerts like "This password reset token is invalid." These messages ensure clear and user-friendly feedback for a smooth password reset experience.

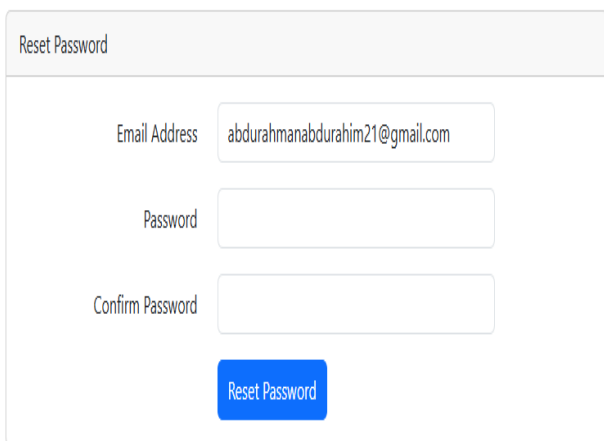


Fig. 29 Password Reset Form View

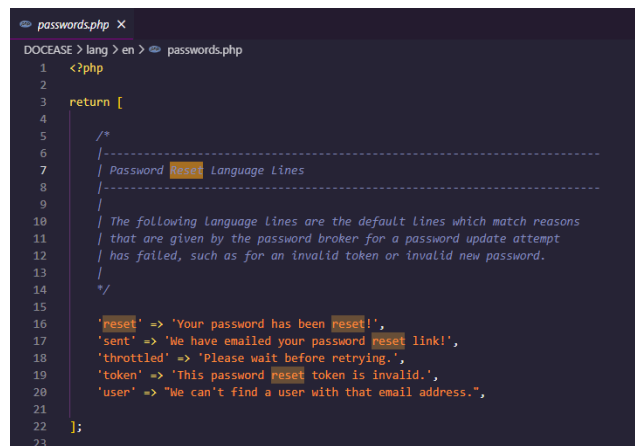


Fig. 30 Password Reset Language File

As shown below, the Password Reset Email provides users with a secure and time-sensitive link to reset their password. The email includes a "Reset Password" button that directs users to the password reset page, with a 60-minute expiration for added security. It also reassures users that no further action is required if they did not

request the reset and provides the full reset URL as an alternative option. This ensures accessibility and safeguards against unauthorized password changes.

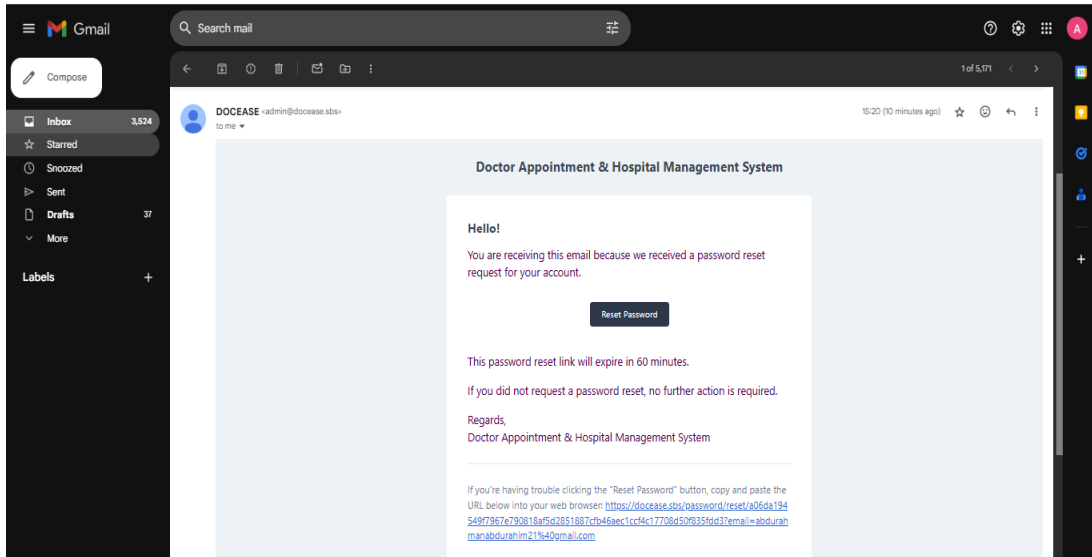


Fig. 31 Password Reset Email

5.2 Testing

During the testing phase, system and user acceptance testing were conducted to ensure functionality, reliability, and user satisfaction. System testing verified core functionalities such as patient registration, appointment booking, user authentication, and role-based dashboard access, ensuring all features performed without errors. User acceptance testing involves administrators, doctors, patients, and support staff, collecting feedback via Google Forms on ease of use, clarity, and efficiency. Feedback was analyzed using visual graphs to identify successes and areas for improvement, ensuring the system met user needs and provided a reliable and user-friendly platform.

5.2.1 System Functionality Testing

Functionality testing ensures that core system features, such as patient registration, appointment scheduling, user role-based access, and department search, operate reliably and meet specified requirements. Each module is rigorously tested under various conditions, comparing expected and actual results to validate performance. As shown in the table below, detailed documentation captures test scenarios, input data, expected outcomes, and actual results, ensuring the system functions as intended.

Table 3 List of Test Cases

Patient Module			
Function	Test Case	Expected Output	Actual Output
Appointment Booking	Patient clicks the "Book Appointment" button and selects a time slot.	Redirect to appointment booking page with available slots.	PASS
Appointment Confirmation	Patient confirms the selected appointment slot.	Patient confirms the selected appointment slot.	PASS
Profile Management	Patient accesses and edits "My Profile."	Profile information updated successfully.	PASS
Support Feature	Patients use the support contact to send a message.	Support message sent and acknowledged.	PASS
	Patient logs in and logs out of the system.	Login/logout processed without errors.	PASS

Login & Logout

Table 3: (cont)

Password Reset	Patient resets password via the reset link.	Password updated and login possible with the new password.	PASS
----------------	---	--	------

Admin Module

Function	Test Case	Expected Output	Actual Output
Appointment Management	Admin views, updates, and reschedules appointments.	Appointments updated and reflected in system.	PASS
Department Management	Admin creates, edits, and deletes departments.	Department changes saved and reflected in the system.	PASS
User Account Management	Admin adds, deletes, or updates user accounts.	User accounts managed and updated successfully.	PASS
System Settings	Admin updates website settings.	Changes saved and reflected on the platform.	PASS

5.2.2 User Acceptance Testing

User acceptance testing for the web system was conducted using Google Forms to gather feedback from administrators(staff), doctors, and patients. A total of 39 responses were received, including contributions from two administrators, one doctor, and 36 patients. The questionnaire assessed the user interface and functionality using a rating scale, where 1 indicated extreme dissatisfaction, and 5 represented extreme satisfaction.

The feedback was analyzed using a linear scale, with results visually represented in tables, as shown in the tables below. This evaluation provided valuable insights into user satisfaction and areas for improvement, ensuring the system aligns with the needs of all user groups.

Table 4 Result of user interface (users)

Features	Ranking					Total
	1	2	3	4	5	
The overall design of the interface	0	0	4	7	26	37
Font size and font colour in the system	0	0	2	10	25	37
Theme colour in the system	0	0	3	8	26	37
Navigation of the system	0	0	3	13	21	37
Responsiveness of interactive elements	0	0	3	10	24	37

Table 5 Result of System Functionalities (Administrator)

Features	Ranking					
	1	2	3	4	5	Total
The system provides secure and easy registration for staff accounts.	0	0	0	0	2	2
It is easy to search and retrieve patient information when needed.	0	0	0	2	0	2
The system makes registering doctors and staff efficient.	0	0	0	0	2	2
I can effectively manage appointment slots and prevent overlapping issues.	0	0	0	1	1	2
The system allows for efficient management of patients, doctors, and staff records.	0	0	0	1	1	2
The system allows me to effectively manage departments, including adding and viewing department details.	0	0	0	1	1	2
I can manage doctor information and assign them to respective departments with ease.	0	0	0	1	1	2
The website settings section is easy to use for making necessary adjustments to the system.	0	0	0	1	1	2

Table 6 Result of System Functionalities (Doctor)

Features	Ranking					
	1	2	3	4	5	Total
The dashboard provides a clear and easy layout for managing appointments and related tasks.	0	0	0	0	1	1
I can efficiently view all appointments, including total, future, and completed appointments.	0	0	0	1	0	1
The system allows me to access and manage patient information effectively.	0	0	0	0	1	1
The system provides an easy-to-use profile section for managing my personal information.	0	0	0	1	0	1
I can effectively configure and manage my availability through the "Timings" feature.	0	0	0	1	0	1
The option to change my password within the system is straightforward and secure.	0	0	0	0	1	1
The appointment history section is useful for reviewing past patient interactions and ensuring continuity of care.	0	0	0	0	1	1

Table 7 Result of System Functionalities

Features	Ranking					
	1	2	3	4	5	Total
The system provides secure registration and login for new accounts.	0	1	2	9	25	37

The booking appointment slot selection process is smooth and avoids overlapping.	0	0	3	10	24	37
I can view my upcoming appointments, including appointment number, date, time, and status.	0	0	4	8	25	37
The system allows me to update and manage my personal profile information effectively.	0	0	2	6	29	37
The option to change my password is easy to use and enhances security.	0	0	3	9	25	37
The support section provides accessible and helpful resources for resolving any issues.	0	1	4	9	23	37
I find the quick links (e.g., My Appointments, My Profile, Support) useful for easy navigation.	0	0	2	9	26	37
The system provides a clear and organized way to browse departments, find doctors, and view their details.	0	1	4	6	26	37

6. Conclusion

The Doctor Appointment System (DOCEASE) successfully modernizes healthcare navigation at Pusat Kesihatan Universiti (PKU), enhancing appointment scheduling, accessibility, and user satisfaction. By addressing inefficiencies in traditional systems, DOCEASE provides a secure, user-friendly platform with real-time booking, personalized dashboards, and efficient data management. Developed using Agile methodology, the system underwent rigorous testing and received high user satisfaction ratings. While achieving its primary goals, future improvements such as predictive analytics and broader system integration are recommended to further optimize its impact on patient care and operational efficiency.

Acknowledgement

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support.

Conflict of Interest

Authors declare that there is no conflict of interests regarding the publication of the paper.

Author Contribution

This journal requires that all authors take public responsibility for the content of the work submitted for review. The contributions of all authors must be described in the following manner:

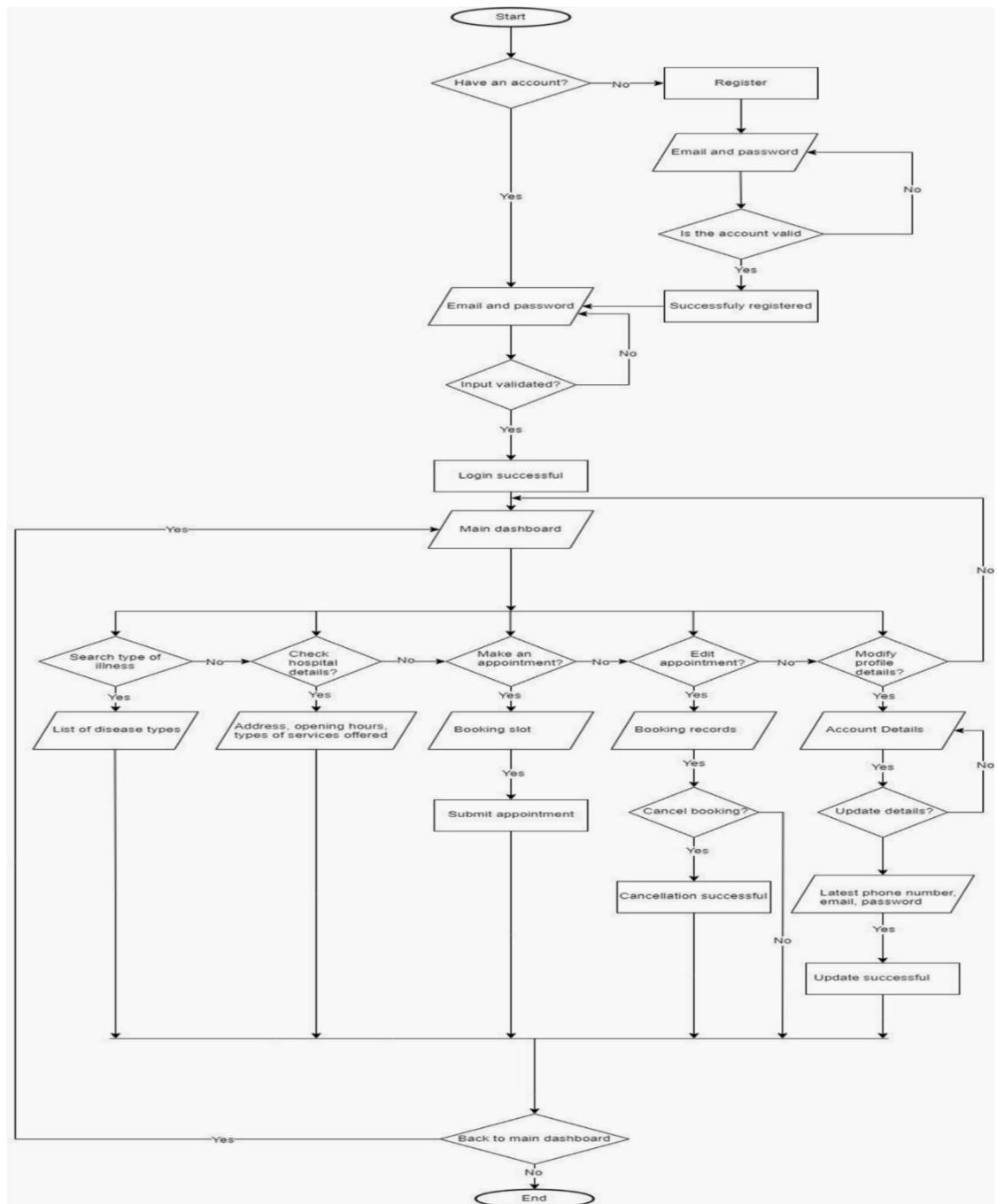
*The authors confirm contribution to the paper as follows: **study conception and design:** Mohamed Abdiraman Abdirahim, Abd Samad Bin Hasan Basari; **data collection:** Mohamed Abdiraman Abdirahim; **analysis and interpretation of results:** Mohamed Abdiraman Abdirahim, Abd Samad Bin Hasan Basari; **draft manuscript preparation:** Mohamed Abdiraman Abdirahim, Abd Samad Bin Hasan Basari. All authors reviewed the results and approved the final version of the manuscript.*

References

- [1] Cao, W., Wan, Y., Tu, H., Shang, F., Liu, D., Tan, Z., Sun, C., Ye, Q., & Xu, Y. (2011b). A web-based appointment system to reduce waiting for outpatients: A retrospective study. *BMC Health Services Research*, 11(1). <https://doi.org/10.1186/1472-6963-11-318>
- [2] Cooper, R. B., & Zmud, R. W. (1990). Information Technology Implementation Research: A Technological Diffusion approach. *Management Science*, 36(2), 123–139. <https://doi.org/10.1287/mnsc.36.2.123>

- [3] Arnett, D. K., Goodman, R. A., Halperin, J. L., Anderson, J. L., Parekh, A. K., & Zoghbi, W. A. (2014). AHA/ACC/HHS strategies to enhance application of clinical practice guidelines in patient's diseases conditions. *Circulation*, 130(18), 1662–1667. <https://doi.org/10.1161/cir.000000000000128>
- [4] Wang, W., & Gupta, D. (2011). Adaptive Appointment Systems with Patient Preferences. *Manufacturing & Service Operations Management*, 13(3), 373–389. <https://doi.org/10.1287/msom.1110.0332>
- [5] Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. *Journal of Systems and Software*, 85(6), 1213–1221. <https://doi.org/10.1016/j.jss.2012.02.033>
- [6] Levina, N. (2005). Collaborating on multiparty information Systems development projects: A Collective Reflection-in-Action View. *Information Systems Research*, 16(2), 109–130. <https://doi.org/10.1287/isre.1050.0055>
- [7] Smith, J., & Brown, L. (2023). Enhancing Healthcare Efficiency Through Digital Appointment Systems: A Comprehensive Review. *Journal of Health Informatics and Technology*, 12(4), 45-58. <https://doi.org/10.1016/j.jhit.2023.04.002>
- [8] Al-Masree, H. K. (2015). Extracting Entity Relationship Diagram (ERD) from Relational Database SCHEM. *International Journal of Database Theory and Application*, 8(3), 15–26. <https://doi.org/10.14257/ijdta.2015.8.3.02>
- [9] Mouelhi, T., Fleurey, F., Baudry, B., & Traon, Y. L. (2008). A Model-Based framework for security policy specification, deployment and testing. In *Lecture notes in computer science* (pp. 537–552). https://doi.org/10.1007/978-3-540-87875-9_38
- [10] Innocenti, T., Mirheidari, S. A., Kharraz, A., Crispo, B., & Kirda, E. (2021). You've got (a reset) mail: A security analysis of Email-Based password reset procedures. In *Lecture notes in computer science* (pp. 1–20). https://doi.org/10.1007/978-3-030-80825-9_1

Appendix A: Patient Flowchart



Appendix B: Administrator Flowchart

