WELLBE
Personalized Patient Care System

Project Interim Proposal

SCS 2202 - Group Project I - CS Group 22

Table of Contents

Details of Project Supervisor and Co-supervisor

Proposed Project Supervisor

Name of the supervisor: Mr. D. A. D. Vithanage
Signature of the supervisor:
Date:

Proposed Project Co-Supervisor

Name of the co-supervisor: Mr. Thulasigaran
Signature of the co-supervisor:
Date:

1. Introduction to Project

Personalized patient care systems represent a transformative approach in modern healthcare facilities, aiming to tailor medical services and treatments to streamlining healthcare processes. These systems integrate advanced technologies and comprehensive functionalities to enhance patient engagement, improve clinical outcomes, and optimize healthcare delivery.

 Key components and functions include:
● Patient- Centric Approach
● Clinical Decision Support (CDS)
● Data-Driven Insights
● Patient Engagement and Communication
● Care Coordination and Collaboration
● Regulatory Compliance and Security

2. Domain Description

In modern healthcare environments, the challenges of fragmented patient care, inefficient communication channels, administrative complexity, and data security and compliance concerns underscore the urgent need for a comprehensive healthcare management system like WellBe: Personalized Patient Care System. Current healthcare systems often fail to provide a seamless patient care experience, with patients struggling due to disjointed medical records, cumbersome appointment scheduling processes, and difficulties in coordinating with their healthcare provider. This fragmentation leads to delays in accessing care, inconsistent treatment plans, and increased patient dissatisfaction. Additionally, communication gaps between doctors, labs, pharmacies, and administrative staff hinder effective collaboration and information exchange, contributing to delays in diagnosis, treatment initiation, and medication distribution, ultimately impacting patient outcomes. Improving communication efficiency is crucial for enhancing care coordination and ensuring timely, accurate healthcare delivery. Healthcare facilities also face the complexities of managing patient profiles, scheduling appointments, and complying with regulatory requirements such as HIPAA and GDPR. Manual administrative processes increase operational costs and pose challenges in maintaining data accuracy, securing patient information, and meeting regulatory standards. Streamlining administrative workflows is essential for optimizing operational efficiency and improving overall service delivery.

WellBe addresses these challenges by offering a unified, integrated platform that enhances patient care coordination, improves communication among healthcare providers, simplifies administrative tasks, and ensures data security and regulatory compliance. By leveraging advanced technologies such as electronic health records (EHR), real-time communication tools, and predictive analytics,

WellBe empowers healthcare facilities to deliver personalized care plans, streamline workflows, and enhance patient outcomes. Through personalized care planning, streamlined communication channels, and robust security protocols, WellBe aims to transform healthcare delivery, mitigate operational challenges, and elevate patient care standards.

3. Current System and limitations

In a typical healthcare setting, patients and healthcare providers often face challenges due to busy schedules and fragmented communication. The WellBe: Personalized Patient Care System addresses these challenges by providing an interface that allows patients and healthcare providers to stay informed about the patient's well-being without constant calls or visits. The system simplifies and streamlines healthcare management by giving doctors, nurses, and healthcare staff easy access to patient records, medical histories, appointment schedules, and treatment plans through a user-friendly app. This helps save time, reduce errors, and enhance the overall quality of care. While healthcare staff are experienced, they often have limited time to monitor every patient's progress. WellBe offers essential tools to help healthcare providers effectively utilize their time, ensuring that each patient's care is monitored closely and accurately.

- **Dependence on Technology**: The system relies heavily on technology, which can be problematic in areas with limited internet access or in cases of technical failures. Any disruption in the system could affect the accessibility of critical patient information.

- **Data Security and Privacy**: Despite robust security measures, there is always a risk of data breaches or unauthorized access, especially with the sensitive nature of healthcare data. Ensuring compliance with regulations such as HIPAA and GDPR can be challenging.

- **Learning Curve for Healthcare Providers**: Healthcare staff, particularly those not well-versed with technology, may face a learning curve when adapting to the new system. This could lead to delays or errors during the initial implementation phase.

- **Integration with Existing Systems**: WellBe may face challenges when integrating with other legacy systems in place at healthcare facilities. Incompatible data formats or software architectures can hinder smooth data exchange.

- **Over-reliance on Automated Monitoring**: While the system helps in monitoring patient care efficiently, it may not capture every nuance or special condition that requires direct human observation. Healthcare providers may miss important cues without personal interaction.

- **Cost of Implementation**: For smaller healthcare providers or clinics, the cost of implementing and maintaining the WellBe system could be a significant barrier, especially if it requires hardware upgrades or additional resources.

- **Patient Engagement**: Although the system allows patients to stay informed, some patients may not fully engage with the digital interface or may prefer more traditional methods of communication, limiting the system's effectiveness in some cases.

- **Scalability Issues**: As healthcare facilities grow or expand, scaling the system to handle increased data and user load may present challenges, requiring additional resources and updates to maintain performance.

- **Limited language Support:** System is only available in English, and patients are required to register before using the platform

4. Goals

The main goal of the WellBe: Personalized Patient Care System project is to develop a comprehensive and user-friendly healthcare management system that enhances patient care by streamlining the processes of patient management, system administration, medical records handling, communication, and operational efficiencies across doctors, patients, labs, and pharmacies. The system aims to provide personalized care, improve access to healthcare services, ensure data security, and foster seamless interactions between all stakeholders in the healthcare ecosystem.

5. Objectives

Objective 1: Enhance Patient Engagement by personalized Health Plans

● Patients receive customized health plans based on their medical history, preferences, and lifestyle, leading to improved adherence and better health outcomes.

Objective 2: Improve Communication Between Patients and Healthcare Providers

● Access to Medical Records: Patients and doctors can access medical records anytime, ensuring accurate and up-to-date information during consultations.
● Secure Messaging: Patients can easily communicate with their healthcare providers, ask questions, and receive timely responses, reducing the need for frequent in-person visits.

Objective 3: Streamline Administrative Processes by automated Appointment Scheduling

● Patients can book, reschedule, or cancel appointments online, reducing administrative burden and minimizing no-shows.

Objective 4: Simplify and Enhance the Prescription Process

● Patient Profile Access: Doctors receive the patient's complete medical
profile, including medical history, allergies, and current medications, ensuring

informed decision-making and personalized care.
● Integrated Prescription System: When prescribing medications, doctors can

select from an integrated list of medicines within the system, streamlining the
prescription process and reducing the risk of errors.

Objective 5: Ensure Patients Attend Appointments by automated Appointment Reminders

● Patients receive automated reminders via SMS, prior to their appointments,
reducing missed appointments and improving clinic efficiency

6. Assumptions

Patients have to make payments to have their lab tests done and to get the
results.
● All medical records and prescriptions follow relevant medical standards and
regulations.
● Staff members such as doctors, lab technicians, and pharmacists are verified
and authorized through administrative staff before gaining access to the
system.
● Users are assumed to have basic knowledge of the system to perform their
tasks effectively.
● The WellBe system will not include coverage for emergency medical cases.

7. Project Feasibility

In Sri Lanka, the healthcare system faces numerous challenges that impact patient satisfaction and overall experience. Issues such as long waiting times for appointments, unclear handwritten prescriptions, delays in receiving test results, and the burden of carrying physical medical records are common. To address these problems, we are developing a comprehensive patient care system designed to improve patient comfort and streamline healthcare processes.

7.2. Technical Feasibility

This section measures the flexibility of practical implementation of building our patient care system using the selected technical solutions. The main deliverable of this project is a web app built using HTML, CSS, and JavaScript on the frontend and PHP on the backend, which utilizes MySQL as the datastore. Apache HTTP Server is used to run the PHP applications.
The platform relies on the following components:
● SMS Gateway: To send notifications, reminders, and alerts to patients and caregivers.

● Payment Gateway: To process payments for services.

● Google Meet: To manage consultations and meetings between healthcare professionals and patients.
● Figma or Canva: For UI design.
● GitHub: For code collaboration and version control.
Most of these technologies are either open source or freely available. The team must gain adequate technical knowledge before and during the development process. It is a significant task, but the timeline allows us to acquire considerable technical expertise before starting the actual development work and to continue learning during the build.

7.3. Operational Feasibility

This section measures how well the proposed solution meets the user requirements of the system to solve the issues in the existing system. Therefore, it is necessary to meet desired requirements to be operationally feasible. Currently, most activities related to patient care are carried out through fragmented channels, including phone calls, emails, and paper records, which is time-consuming and often inefficient. These methods do not always provide the necessary details in an organized manner.

Our platform proposes to classify and streamline relevant activities with respect to patient types and needs. Additionally, due to the ongoing healthcare challenges, including the

need for remote consultations, it is essential to have a proper platform to connect patients with healthcare providers for medical opinions and care.

Our solution is a web application accessed through the internet. To use and operate the system:

● Users are required to have an internet connection.
● A computer with a recent version of a web browser should be available.
● Users must have a basic knowledge of IT and using the internet.
● For appointments, consultations, and payments, users need an intermediate level of knowledge in online and card-based transactions.

Because there is no need to provide special training to use the system or any dedicated HR resources to maintain the system, the project is operationally feasible.

Our platform will ensure that patients receive timely and organized care, reducing the inefficiencies present in the current system and improving overall patient satisfaction and outcomes.

7.4. Economical Feasibility

This section provides a detailed overview of the cost estimations and cost-benefit analysis for our solution:

● Hosting Costs: The platform requires hosting for both the application and the database. Utilizing the free tier capacity from a suitable cloud provider initially meets our needs for moderate traffic. As user base and traffic increase, scalability options may incur additional costs.

● Domain Name: Acquiring a ".LK" domain name costs approximately Rs. 3000 per year, enabling public access to the web application.

● Operational Costs: The primary operational cost of our system is SMS notifications, priced at Rs. 0.84 per notification. This cost ensures effective communication with users.

● Payment Gateway: We integrate a payment gateway to facilitate transactions. The chosen gateway offers a free plan with limitations, which currently suffices for our system requirements.

● Development Tools: We leverage open-source technologies and free tools such as Visual Studio Code for development, eliminating the need for paid licenses and reducing development expenses.

● Labor Costs: Development is undertaken by university undergraduates, mitigating labor costs associated with professional developers.

● Cost vs. Benefits Analysis: While there are initial costs for hosting, domain registration, SMS notifications, and potential scalability expenses, these investments

are justified by the significant benefits our system provides, including enhanced patient communication and operational efficiency in healthcare delivery.

● Savings: Certain costs are minimized through strategic decisions, such as using localhost environments for development and leveraging free development tools. This

approach optimizes resource allocation and enhances the overall economic feasibility of our solution.

By balancing costs with anticipated benefits and leveraging cost-saving measures like free payment gateway plans, our solution demonstrates strong economic feasibility, ensuring affordability and sustainability throughout its deployment and operation.

7.5. Legal and Ethical Feasibility

The patient care system ensures robust legal and ethical compliance across various dimensions:

● Verification of User and Healthcare Provider Identities: All users and healthcare

providers undergo identity verification during registration, ensuring accountability and security.

● Compliance with Healthcare Regulations: The system adheres to applicable healthcare laws and regulations, including HIPAA (Health Insurance Portability and Accountability Act) and GDPR (General Data Protection Regulation), ensuring the secure handling of patient data.

● Use of Open-Source Software: Open-source software used in the application complies with respective licenses, fostering transparency and legal compliance in software development and deployment.

● Transparency in Financial Transactions: The system provides clear visibility into payment processes, billing, and transactions, ensuring transparency and accountability in financial dealings.

● Prevention of Misuse: Measures are in place to prevent misuse of prescriptions and medical advice, mitigating legal and ethical risks associated with improper use of healthcare services.

● Data Security and Privacy: Robust data security measures, including encryption protocols and access controls, safeguard patient information against unauthorized access and breaches.

● Patient Consent: Patient consent is obtained transparently for data handling and treatment procedures, respecting individual rights and privacy preferences.

● Terms and Conditions: Clear terms and conditions govern the use of the system, outlining responsibilities and expectations for users and providers.

● Professional Integrity: The system upholds professional integrity standards, promoting ethical conduct among healthcare providers and maintaining trust with

patients.
These measures collectively ensure that the patient care system operates within legal boundaries, respects patient privacy, and upholds ethical standards in healthcare service delivery.

7.6. Schedule Feasibility

● The project spans a duration of 9 months.
● According to that, the estimated man hours for the project completion can be mentioned as follows:
○ Weekdays working hours = 5 hours
○ Weekend working hours = 5 hours
○ Number of team members = 4
○ Number of weeks = 36
○ Total man hours = (5 + 5) * 4 * 36 hours = 1440 hours
● We are using an iterative waterfall model for development, and with requirement gathering almost done, we can predict that with the identified features and scope it is possible to finish the other tasks of the SDLC and complete the development of the product by April 2025.
● Additionally, requirements are almost stable at this point, so there won't be any major changes to the requirements that can affect the schedule of the Project.

8. Functional Requirements

8.1 Administrative Staff

1. Create Patient Profiles
○ Create profiles for patients.
○ Fill in patient details (name, age, contact information, medical history, etc.).
2. Edit Patient Profile
○ Search for the patient profile by name or ID.
○ Select the profile to edit.
○ Update the necessary details.
3. Delete Patient Profile
○ Search for the patient profile by name or ID.
○ Select the profile to delete.
○ Confirm the deletion.

4. Checking the present appointment numbers and updating the queue
○ Check and update the presence of the next appointment number.

○ Update the queue of patients according to the appointment number.
○ Manage the absent patients' appointments.
5. Create Doctor/Lab/Pharmacy Profiles
○ Create profiles for doctors, labs, and pharmacists.
○ Select the type (Doctor, Lab, Pharmacy).
○ Enter the details (name, specialty, contact information, etc.).

6. Edit Doctor/Lab/Pharmacy Profiles
○ Search for the profile by name or ID.
○ Select the profile to edit.
○ Update the necessary details.
7. Delete Doctor/Lab/Pharmacy Profiles
○ Search for the profile by name or ID.
○ Select the profile to delete.
○ Confirm the deletion.
8. Generate Needed Reports
○ Select the type of report needed (e.g., daily patient report, lab test report).
○ Apply filters (date range, department, etc.).
○ Generate Reports.
○ Download or print the report.
9. Give Access to Users
○ Search for the user by name or ID.
○ Select the user and set the required permissions.
10. Add and update the doctors' available time slots for consultations
○ Add and update available time slots of doctors for consultation.
○ Manage the time slots of the appointment scheduler.

8.2. Doctor

1. Login
○ Login to the system using the credentials.
2. View the Patient Profile
○ Access the patient profile section.
○ Review personal details and medical history.
3. View Previous Medical Records
○ Review patient's past visits, diagnoses, and treatments.
○ Review previously prescribed medicine.
4. Create New Medical Record
○ Enter visit details, diagnosis, and treatment plan.
○ Enter newly prescribed medicine.
5. Suggest Lab Tests
○ Suggest required lab tests for patients.

○ Mention the type of test that needs to be done.

○ Submit the request to the lab.

6. Prescribe Medicines

○ Enter new records to the prescribed medicines.

○ Select medications and dosages.

○ Submit the prescriptions.

7. Chat with Patients

○ Open the chat section.

○ Select the patient from the list.

○ Send messages.

8. Generate Treated Patients Report

○ Generate reports summarizing the treated patient statistics.

○ Apply filters (current day) to get specified reports.

○ Download or print the report.

9. Recommending Patients the Next Appointment

○ Suggest a date and time for the next visit.

○ Update the patient's appointment schedule in the system.

10. Check the Scheduled Appointments

○ View the list of scheduled appointments for a particular day.

○ Manage and update appointments as needed.

8.3. Patient

1. View Website

○ Open the WellBe website in a browser.

2. Create Profile

○ Create profiles entering required personal data.

○ Fill in personal details and medical history.

3. Edit Profile

○ Login to the system.

○ Update personal information as needed.

○ Save the updated data.

4. Login

○ Login to the system using credentials.

5. View Medical Reports

○ View past medical records.

○ View and download available reports.

6. View Lab Results

○ Navigate to the lab results section.

○ View and download lab reports after the tests are done.

7. Chat with Doctors

○ Open the chat section.

○ Send messages to doctors via the management interface.

○ Receive replies from doctors directly.

8. View Next Appointment

○ View details of the next scheduled appointment.

○ Get reminders for the next appointment.

9. Find a Doctor

○ Search for doctors by specialty or name.

○ View doctor profiles and select a preferred doctor.

10. Schedule an Appointment with the Doctor

○ Select an available date and time from the doctor's schedule.

○ Book the appointment.

○ Receive confirmation.

○ If the appointment is canceled by either the doctor or the patient, reschedule the appointment (no refund provided).

11. Pay for the Appointment

○ Navigate to the payment section.

○ Complete the payment process online.

12. Receive Notification for Appointment

○ Receive reminders 1 day and 2 hours before the appointment via email or SMS.

○ Receive a notification when the doctor arrives, including the appointment number.

8.4. Lab Technician

1. View Lab Requests

○ View a list of requested test requests.

○ Manage incoming test requests.

2. Update the Status of Lab Requests

○ Change the status to "In Progress", "Pending" or "Completed" as appropriate.

3. Upload Lab Results

○ Upload the lab results to the system.

○ Notify the patient that the results are available.

4. Print Lab Reports

○ Generate and print lab reports if requested by the patient.

5. Update Remarks

○ Add remarks to lab reports, especially if handed over physically.

6. Chat with the Management

○ Use the chat section to communicate with management for clarifications or updates.

7. Lab Report Ready Notification

○ Notify patients when lab reports are ready for viewing in the system.

8.5. Pharmacist

1. View Medication Requests
○ View and manage incoming prescription requests.
○ View the list of medicines prescribed to a patient.
2. Update the Status of Medication Requests
○ Change the status to "Processing", "Ready to Pick", or "Issued" as
appropriate.
3. Hand Over the Medicines
○ Update the status to "Issued".
4. Update Remarks
○ Add remarks for physical handover of medicines or unavailability.
5. Print Prescriptions
○ Generate and print prescriptions if a medicine is not available or if
requested by the patient.
6. Chat with the Management
○ Use the chat section to communicate with management for any issues
or updates.
7. Send Medicine Ready Notification
○ Notify patients when medicines are ready for pick-up.

9. Quality Attributes Requirements

When developing a patient care management system, several key quality attributes are
essential to ensure it meets the needs of both healthcare providers and patients effectively.
Here are some of the most important quality attributes:

9.1. Usability

User-Friendly Interface: Simple and intuitive design for patients, doctors, and management.
Accessibility: Compliance with accessibility standards (e.g., WCAG) to support users with
disabilities.

Language Support: Currently available in English; consider expanding to multiple languages for
future scalability.

9.2. Reliability

Reliability in a patient care management system refers to the system's ability to consistently

perform its required functions under stated conditions for a specified period. Key aspects of
reliability include:
● Error Handling: It should handle errors gracefully and ensure data integrity
in the event of a failure.

● Data integrity: Ensuring that all patient data is accurately recorded, stored, and retrieved without corruption or loss.

9.3. Performance

Fast Response Time: Quick data retrieval and minimal latency for both patients and healthcare providers.

Efficient Data Processing: Optimization for tasks like retrieving medical history, generating reports, and scheduling appointments

9.4. Security

Data Privacy: Ensure strict compliance with data privacy laws (e.g., HIPAA, GDPR). Authentication & Authorization: Secure login mechanisms with role-based access control for patients, doctors, and administrators.

Encryption: Data in transit and at rest should be encrypted.

9.5. Data Accuracy

Data accuracy in a patient care management system is crucial for ensuring that the information used for patient care is reliable, precise, and trustworthy. Here are key aspects of data accuracy in this context:
● Precision: Ensuring the data entered and processed is accurate and reliable.
● Validation: Incorporating checks to validate data input and updates.
● Standardization: Using standardized formats and terminologies (e.g. medical codes, units of measurement) to ensure consistency and reduce ambiguities.

9.6. Maintainability

Maintainability in a patient care management system refers to the ease with which the system can be modified to fix defects, improve performance, or adapt to a changing environment.

Here are key aspects of maintainability in this context:
● Modularity: The system should be designed in a way that makes it easy to update and maintain.
● Reusable components: Using reusable components to avoid redundancy and simplify updates across the system.
● Documentation: Comprehensive documentation for users and developers, including system architecture, code comments, user manuals, and troubleshooting guides.
● Up-to-Date Information: Ensuring that all documentation is current and reflects the latest changes and update s in the system.

● Code quality: Adhering to consistent coding standards and best practices to maintain uniformity across the codebase.
● Version Control: Using version control system (Git) to track changes, manage code versions, and facilitate collaboration among developers.

9.7. Adaptability

Customizable Features: Allow users to personalize settings, such as notifications, preferred doctors, and interface themes.

Flexible Workflows: Support various workflows depending on the healthcare facility's needs.

9.8. Scalability

Handling Growth: Ability to handle an increasing number of users (patients, doctors, and staff) without performance degradation.

Modularity: Easy addition of new features, such as language support, new analytics dashboards, or integration with wearables.

10. Scope of the project

10.1. Users

● Patient
● Doctor
● Lab technician
● Pharmacist
● Administrative staff

10.2. In-scope

● Create a platform to centralize patient's medical and lab reports.
● Allow patients to register in the system either through administrative staffs or independently.
● Enable administrative staff to register doctors, pharmacists, and lab technicians into the system
● Allow the administrative staff to check patient presence and manage appointment orders.
● Allow patients to view their medical reports, prescriptions and health

history.

● Enable doctors to call the next patient based on appointment order, then the patient's profile will be displayed to the doctor automatically.Then the doctor can access the patient's previous medical reports and health history.
● Provide patients with the ability to schedule appointments with doctors and make payments through the system.
● Enable doctors to view and manage their appointment schedules, including rescheduling after consulting with management.

● Allow doctors to create patient prescriptions.
● Enable pharmacists to view prescriptions by entering patient IDs and update prescription status after dispensing medications.
● Enable pharmacists and lab technicians to update remarks.
● Allow lab technicians to view required tests for patients by entering patient IDs.

● Enable lab technicians to release test's results and update the status of tests once they are completed.
● Facilitate chat functionality for patients to communicate with doctors for consultations or follow-ups and for lab technicians and pharmacists to communicate with hospital management.
● Send automated notifications for appointment reminders, test results availability, and prescription updates to patients, doctors, lab technicians, and pharmacists.
● Allow management to oversee and manage profiles of patients, doctors, lab technicians, and pharmacists.
● Implement security measures to protect sensitive medical information.

10.3. Out-of-Scope

● Provide a mobile application for patients to interact with the system easily.
● Facilitate video call functionality for patients to communicate with the doctors.
● Enable lab technicians and pharmacists to manage stock and inventory.

● Allow to manage employee records, including hiring, payroll, performance evaluations, and other human resources-related activities.
● In our system, we only consider doctors, pharmacists, and lab technicians as staff members.

11. Proposed System's Architecture

This section provides an detailed description on the architecture design for the proposed system. Including the high-level structure, main component and their roles. It explains how the system will be organized.

The proposed system is based on the MVC design pattern. This architecture divides the application into three interconnected components and separation and ease of maintenance. Components and functionalities

1.   Modal
•   Represents the data layer of the system.
•   Manages data processing like protected columns and database interactions.

User, Parent, Child, Guardian, Reservation Modal files that correspond to tables in the database.

2.   View
•   Represents the User Interface of the system.
•   Displays data to users and collects inputs from these files.
Web pages for Landing page, Login, Home page, Reservation page which contains html codes.

3.   Controller
•   Acts as an intermediary between the Modal and View files.
•   Processes user request applies logic and updates the Model and View.
•   Which renders view files and sends needed data from the database to view.
PHP files like Home, Attendance, reservations files

4.   Core
The core folder is a crucial part of the system's architecture and serves as a repository for shared resources, utilities, and foundational components used across the system.

App.php – Handles loading of controllers from the given URL.
Config.php – database connection and base paths of CSS, JS, ROOT, IMAGE, VIDEO and UPLOAD files
Controller.php – Provides view function to import file and if not show 404 error page
Database.php – Executes SQL queries generated by the Modal.php file.
Functions.php - Reusable utility functions such as date formatting, data validation, or encryption.
Init.php – Automatically requires all files in the modal folder and registers class names for the system.

Modal.php – Contains functions for generating SQL queries based on given parameters and sends them to Database.php.
Session.php – class to handle Session creation and deletion.

12. Component Interactions

Each interaction is explained with reference to key workflows with a simple example of login.
When a person start the interaction with the web application from the landing page which is
renders from the controller to the view page and displays the web page view
When the login button is clicked the App looks for the specific file controller file and through that
the view file is rendered.
When a user attempts to log in, their credentials (username and password) are validated from the
front end using JavaScript if there are any errors it will be shown.
If there are no errors The front-end sends an HTTP POST request containing the credentials to
the back-end via the App.php component

The back-end controller (Controller.php) processes the request and calls the Modal.php to
generate an SQL query.
The SQL query is executed by the Database.php component to validate the credentials against
stored user data.
If validation succeeds, a session is created using the Session.php component, and a success
response is sent back to the front-end.
The front-end updates the user interface to reflect the successful login and redirect to home page
depending on the role or displays an error message for invalid credentials.
If there are any errors the errors will be sent through the controller to the view page.
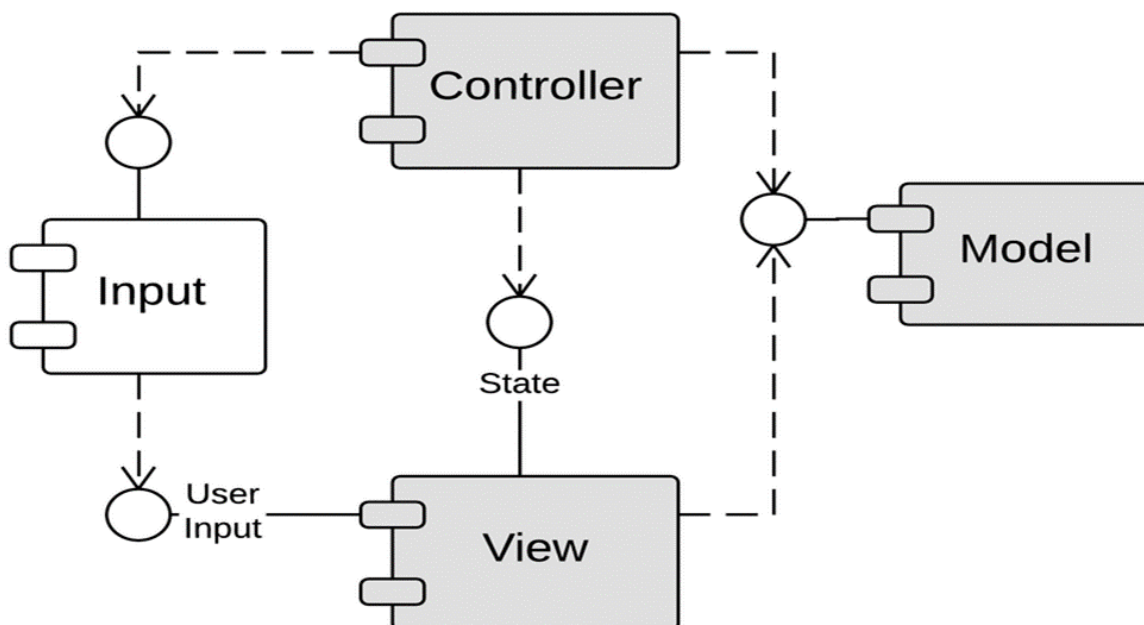
13. System development Progress

The project is currently on track with some functionalities completed. Some noncritical modules are scheduled for upcoming days. The current system with the implemented functionalities are integrated and tested for smooth functionality.

•　　　Login / Signup – fully implemented, tested for various scenarios and integrated with session handling.
•　　　UI - Main pages and user UI is completed and integrated.
•　　　API - Integrated Payment gateway
•　　　Appointment– Appointment process for Registered patients CRUD is completed and integrated and fully tested.
•　　　Package – Package creation for managers CRUD completed and tested.
•　　　Database – Fully created the tables in the Database as needed

14. Estimated competitions

　The development of the system is on track, with significant progress made in core functionalities. Based on the current status and planned milestones, the estimated completion date for the system is **March 31, 2024.**

The project has been divided into phases to systematically address key system requirements.

- **Phase 1**: Focused on the implementation of critical functionalities like **Login/Signup**, **Appointment Management**, and **Database Setup**. These modules are completed and fully tested.
- **Phase 2**: Scheduled to begin in the next sprint and will cover advanced features such as **Refund option, Medical/Lab report viewing**, **Messager**, **Appointment Queue implementing** and web pages for each user.

- **Phase 3**: Dedicated to system-wide testing, debugging, and deployment preparation.

From the above phase we have the phase 1 completed from the interim presentation.

15. Completed Percentages

Component	Completion Percentage	Status
Frontend Development	100%	Completed
Backend Development	20%	In Progress
UI Design	100%	Completed
Database	60%	In Progress
Authentication	40%	In Progress

16. Remaining Tasks

Component	Remaining	Details
Frontend Development	0%	Completed
Core Functionalities	80%	Major functionality development is ongoing.
Messaging API Integration	0%	API for messaging needs to be selected and integrated.
Database	30%	Payment management system needs to be restructured and integrated to backend
User Authentication	50%	Basic authentication implemented, but redirect logic is incomplete.

18. System Design Diagrams

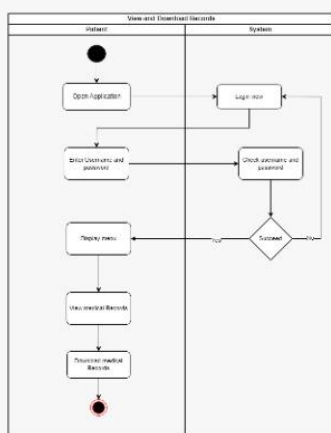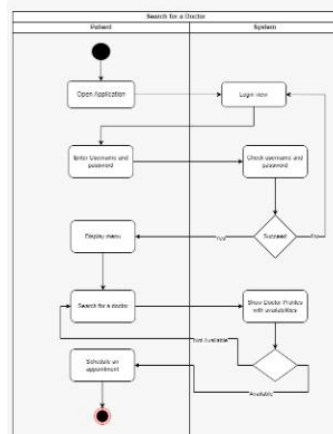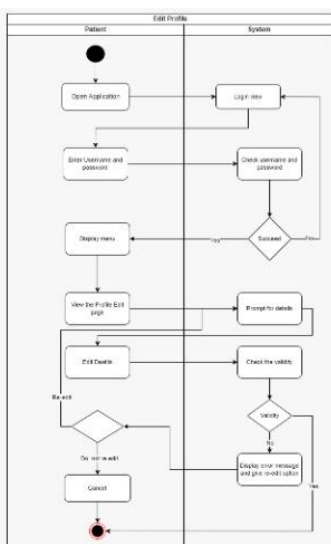## 18. 1. Use case diagrams
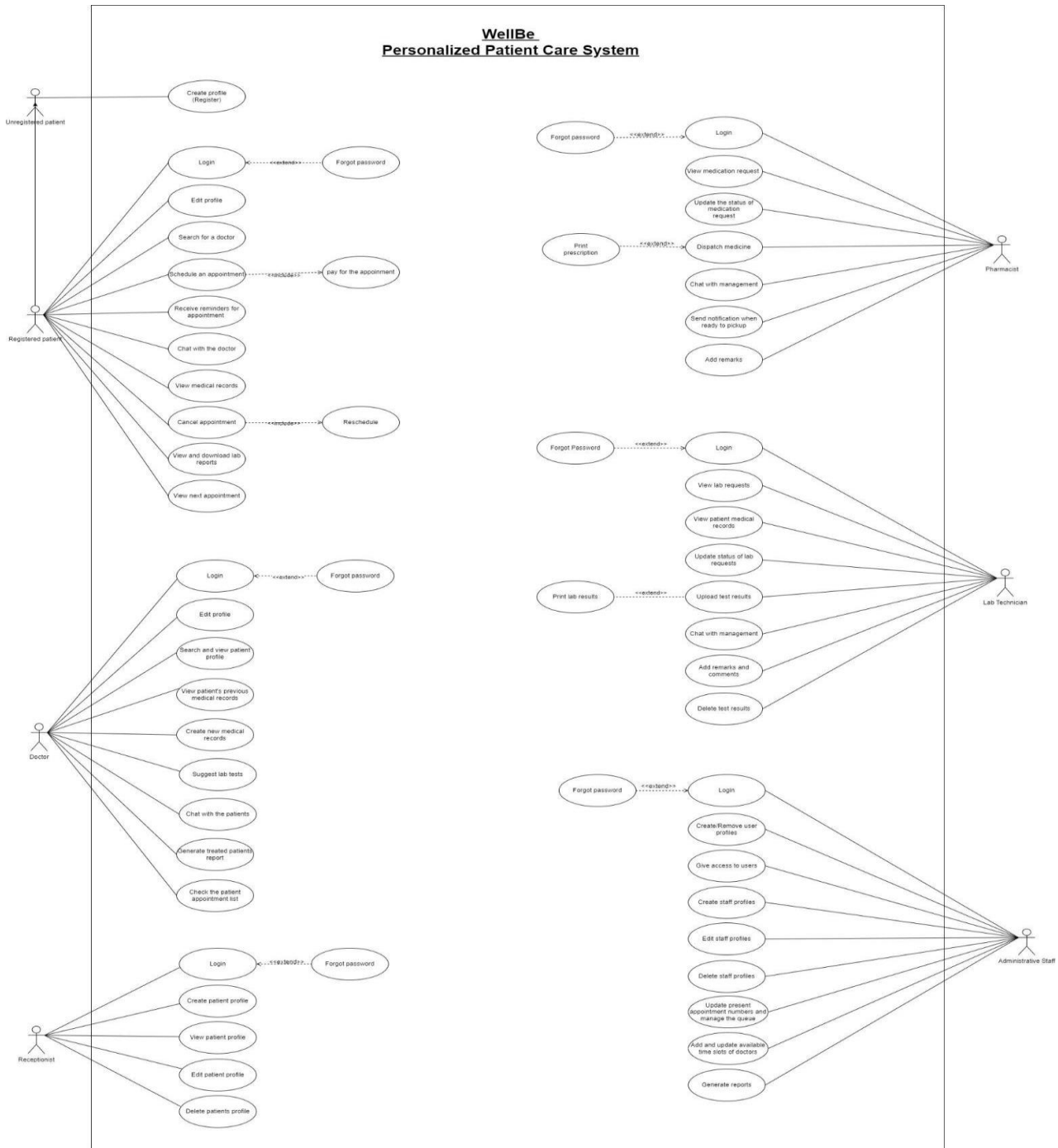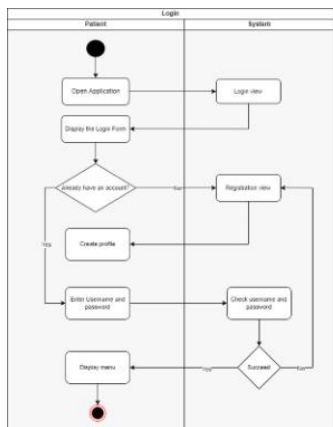
18. 2. Class diagrams
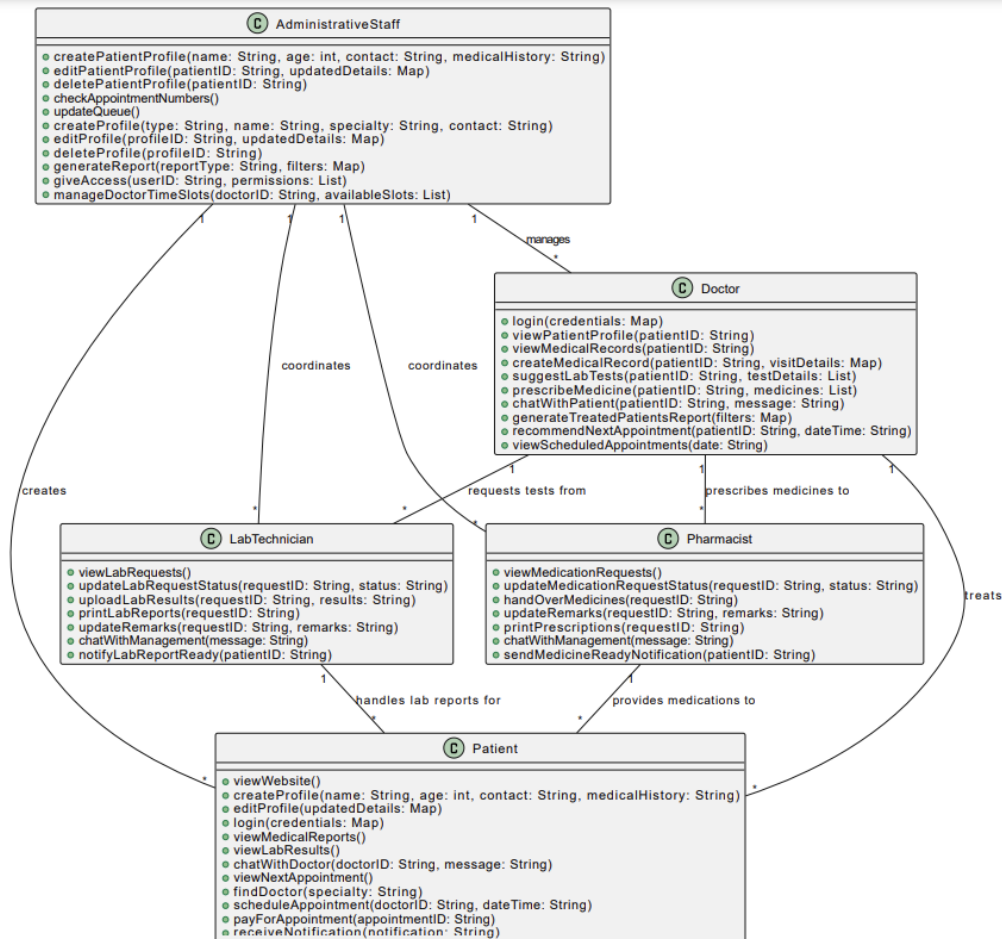
18. 3. ER Diagram
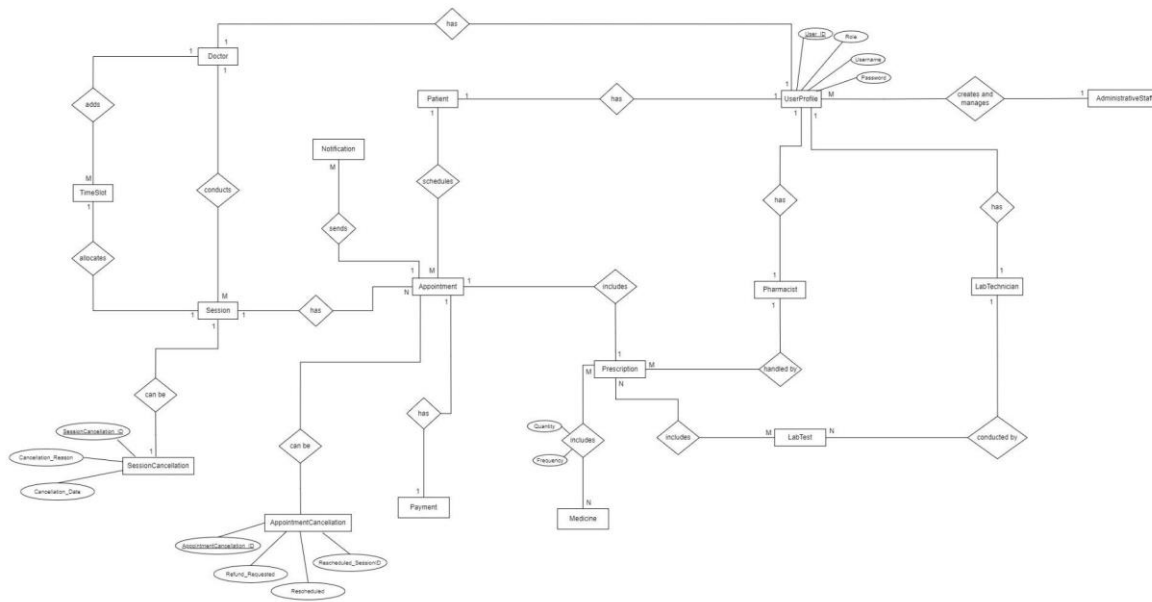
18. 4. Activity diagrams
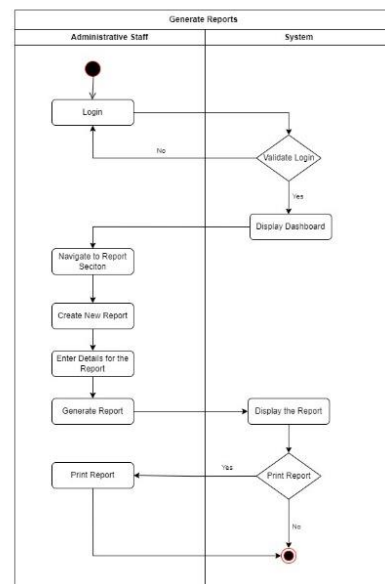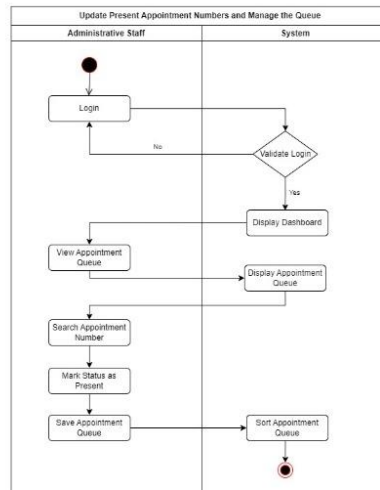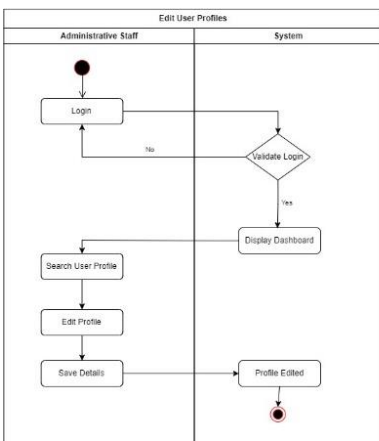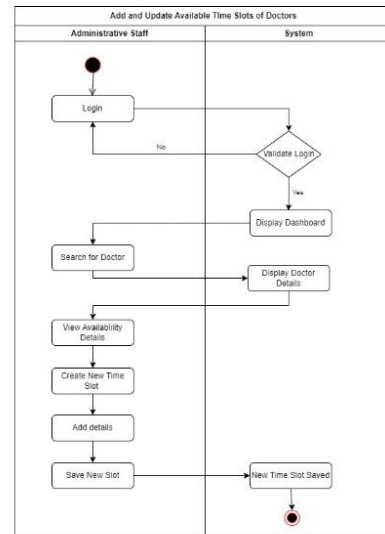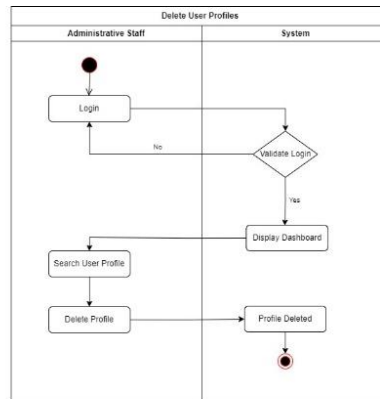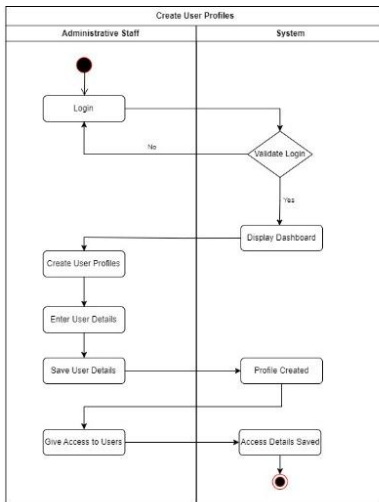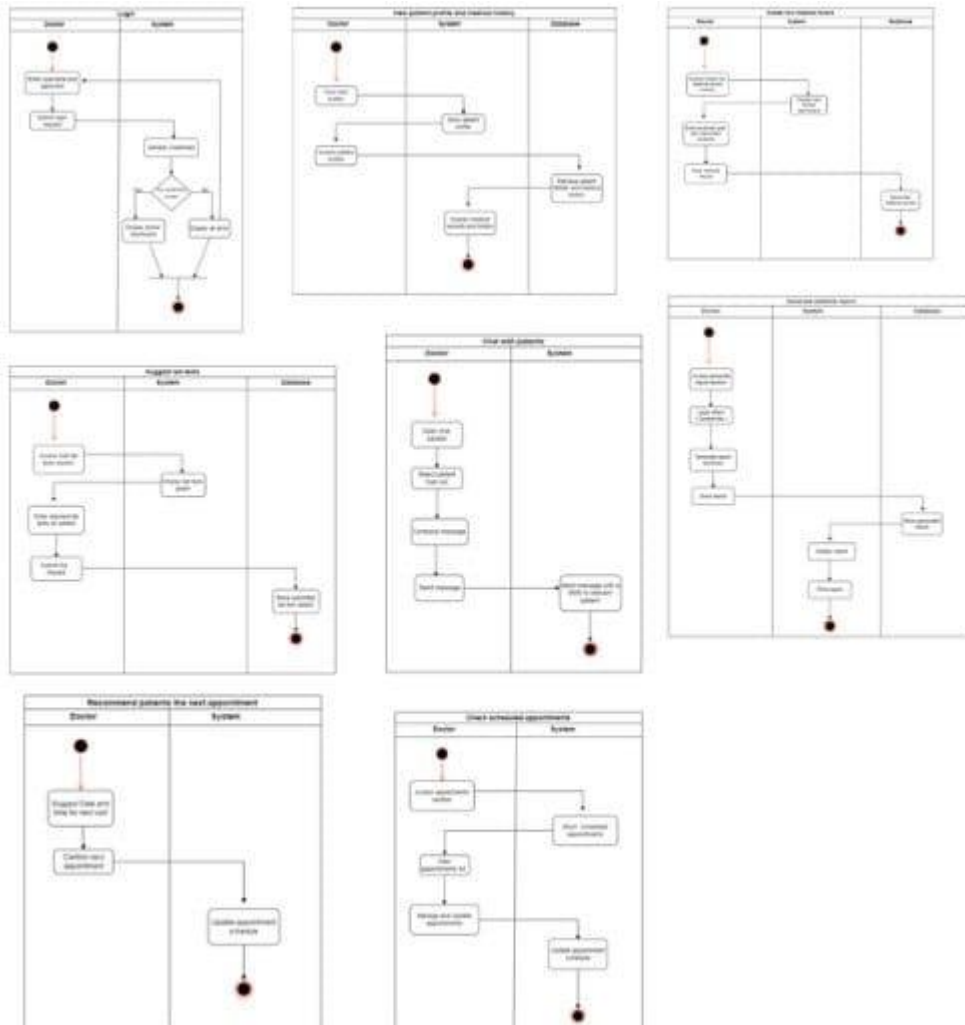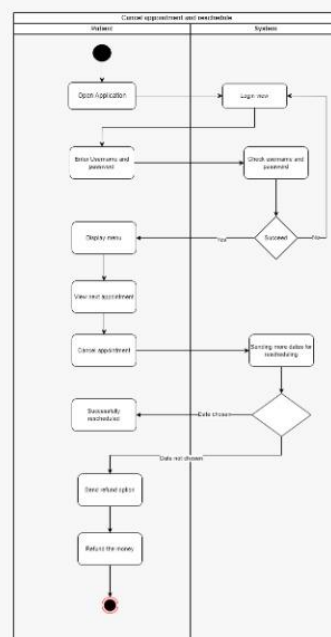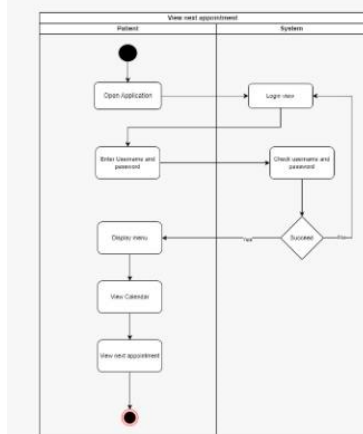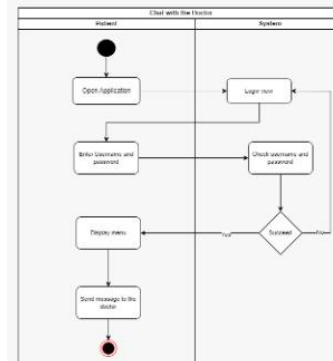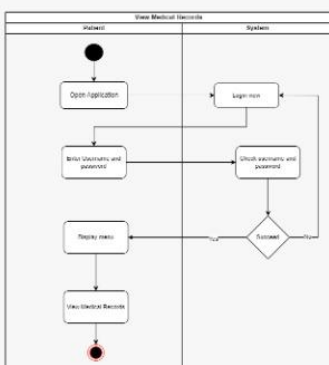
19. Member Contributions

This section highlights the individual contributions of team members to the development of the system, specifying the components or modules they were responsible for.

For each member's contribution is divided into their user roles in the system.

1. Amrah Slamath 22001931

   · Patient progress contribution.

     o Implemented all Registered and Unregistered patient Ui design

     o key user-facing pages of the system such as landing page's UI.

     o Patient appointment booking CRUD.

2. Raveesha Samarasekera 22001761

   · Admin progress contribution

     o Implemented admin's UI designs.

     o Creating profiles for staff.

3. Himesh Dinidu 22000399

   · Doctor progress contribution

     o Implemented Doctor's UI

     o Created MVC architecture

4. Benshekniel Thayalan 22000232

   · Lab Assistant and Pharmacist

     o Implemented   Lab Assistant and Pharmacist UI

     o Chat live feature creation

     o Updating the medication and lab tests status