EAST WEST UNIVERSITY

Department of CSE

CLINIC MANAGEMENT SYSTEM

Submitted by

Name of Student: MD. Shahrukh Hossain Shihab
Nushrat Jaben Aurnima
Zihad Khan
Shairin Akter Hashi
Sabera Zannat Sheba

Student Id: 2022-1-60-372
2022-2-60-146
2022-2-60-107
2022-2-60-102
2022-2-60-095

Course Title: Information System Analysis and Design

Course Code: CSE 347

Semester: Fall 2024

Submitted to

Anika Tabassum

Lecturer, Department of Computer Science and Engineering

Letter of Transmittal

Anika Tabassum

Lecturer

Department of Computer Science and Engineering

East West University

Subject: Submission of term report on “Clinic Management System”.

Ma’am,

With due respect, we would like to submit you the report on Software Requirement Specification on the above topic you assigned us. The report reflects our effort to gather requirements and analysing them. We have included every step what we have done through the whole time for requirement specification of the mentioned topic.

Therefore, we earnestly hope that you will excuse our error and obliged thereby.

Yours Sincerely,

MD. Shahrukh Hossain Shihab	2022-1-60-372
Nushrat Jaben Aurnima	2022-2-60-146
Zihad Khan	2022-2-60-107
Shairin Akter Hashi	2022-2-60-102
Sabera Zannat Sheba	2022-2-60-095

3rd year, 8th semester

Department of Computer Science and Engineering

East West University

Acknowledgment

We are highly indebted for getting such a tremendous opportunity to prepare the report on the Clinic Management System. We would like to thank whole-heartedly our course teacher, Anika Tabassum, Lecturer, Department of Computer Science and Engineering, East West University, for providing us with valuable guidance and insights on how to prepare this report. Her constructive feedback and encouragement have been pivotal in completing this project. We are also sincerely grateful to all the individuals and stakeholders who provided their requirements and shared their insights, enabling us to develop a system that addresses practical needs effectively.

Abstract

Clinic Management has become an essential strategy for ensuring efficient and effective healthcare services in today's fast-paced world. This study focuses on developing a system to manage the various aspects of clinic operations, with a primary emphasis on client (patient) management. The objective of this project is to create a Software Requirements Specification (SRS) for a Clinic Management System based on the requirements gathered from stakeholders. The proposed system aims to streamline patient registration, appointment scheduling, medical record management, and billing processes. This report represents the objectives of the system and emphasizes the growing need for such solutions in modern healthcare management to enhance both operational efficiency and patient satisfaction.

Table of Contents

Chapter 1: Introduction 1

1.1 Purpose 1

1.2 Intended Audience 1

1.3 Conclusion 1

Chapter 2: Inception 1

2.1 Introduction 1

2.1.1 Identifying Stakeholders 1

2.1.2 Recognizing Multiple Viewpoints 2

2.1.3 Working Towards Collaboration 2

2.1.4 Asking the First Questions 3

2.2 Conclusion 3

Chapter 3: Elicitation 3

3.1 Introduction 3

3.2 Eliciting Requirements 3

3.3 Collaborative Requirements Gathering 4

3.4 Quality Function Deployment (QFD) 4

3.4.1 Normal Requirements 4

3.4.2 Expected Requirements 4

3.4.3 Exciting Requirements 5

Chapter 4: Scenario-Based Modelling 5

4.1 Introduction 5

4.2 Usage Scenario 5

4.3 Use Case Scenario 6

4.4 Use Case Diagram 7

Figure 1: Authentication Module 7

Figure 2: Patient Module 7

Figure 3: Doctor Module 8

Figure 4: Admin Module 8

Figure 5: Billing Module 9

Figure 6: Appointment Module 9

4.5 Activity Diagram and Swimlane Diagram of generated Use Cases 10

Figure 7: Activity Diagram Log In 10

Figure 8: Swimlane Diagram for Log In 11

Figure 9: Activity Diagram Log Out 12

Figure 10: Swimlane Diagram for Log Out 13

Figure 11: Activity Diagram Book Appointment 14
Figure 12: Swimlane Diagram for Book Appointment 15
Figure 13: Activity Diagram for View Profile (Patient) 16
Figure 14: Swimlane Diagram for View Profile (Patient) 17
Figure 15: Activity Diagram for Update Profile (Patient) 18
Figure 16: Swimlane Diagram for Update Profile (Patient) 19
Figure 17: Activity Diagram for Access Prescription 20
Figure 18: Swimlane Diagram for Access Prescription 21
Figure 19: Activity Diagram for View Doctor List 22
Figure 20: Swimlane Diagram for View Doctor List 23
Figure 21: Activity Diagram for View Appointments 24
Figure 22: Swimlane Diagram for View Appointments 25
Figure 23: Activity Diagram for Cancel Appointments 26
Figure 24: Swimlane Diagram for Cancel Appointments..... 27
Figure 25: Activity Diagram for View Patient List (Doctor) 28
Figure 26: Swimlane Diagram for View Patient List 9Doctor0..... 29
Figure 27: Activity Diagram for View Profile..... 30
Figure 28: Swimlane Diagram for View Profile..... 31
Figure 29: Activity Diagram for Update Profile (Doctor)..... 32
Figure 30: Swimlane Diagram for Update Profile (Doctor)..... 33
Figure 31: Activity Diagram for View Patients (Admin) 34
Figure 32: Swimlane Diagram for View Patients (Admin)..... 35
Figure 33: Activity Diagram for View Doctors (Admin)..... 36
Figure 34: Swimlane Diagram for View Doctors (Admin)..... 37
Figure 35: Activity Diagram for Add Doctor 38
Figure 36: Swimlane Diagram for Add Doctor 39
Figure 37: Activity Diagram for Delete Doctor 40
Figure 38: Swimlane Diagram for Delete Doctor 41
Figure 39: Activity Diagram for Remove Appointment..... 42
Figure 40: Swimlane Diagram for Remove Appointment..... 43
Figure 41: Activity Diagram for Generate Bill..... 44
Figure 42: Swimlane Diagram for Generate Bill..... 45
Figure 43: Activity Diagram for View Bills 46
Figure 44: Swimlane Diagram for View Bills 47
Figure 45: Activity Diagram for Pay Bill..... 48
Figure 46: Swimlane Diagram for Pay Bill..... 49

Figure 47: Activity Diagram for Schedule Appointment 50
Figure 48: Swimlane Diagram for Schedule Appointment 51
Chapter 5: Class based modeling 52
Figure 49: UML Diagram 52
Chapter 6: Flow-Oriented Model 53
6.1 Data Flow Diagram (DFD) 53
Level 0 of data flow diagram 53
Figure 50: Level 0 for Clinic Management System 53
Level 1 of data flow diagram 54
Figure 51: Level 1.1 for Clinic Management System 54
Figure 52: Level 1.2 for Clinic Management System 55
Figure 53: Level 1.3 for Clinic Management System 56
Level 2 of data flow diagram 57
Figure 54: Level 2.1 for Clinic Management System 57
Figure 55: Level 2.2 for Clinic Management System 58
Figure 56: Level 2.3 for Clinic Management System 58
Figure 57: Level 2.4 for Clinic Management System 59
Chapter 7: Behavioral Model 60
7.1 State Machine Diagrams 60
Figure 58: State Diagram (Admin) 60
Figure 59: State Diagram (Doctor) 61
Figure 60: State Diagram (Patient) 62
7.2 Sequence Diagrams of Clinic Management System 63
Figure 61: Sequence Diagram (Log In) 63
Figure 62: Sequence Diagram (Log Out) 64
Figure 63: Sequence Diagram (Patient Management) 65
Figure 64: Sequence Diagram (Doctor Management) 66
Figure 65: Sequence Diagram (Admin Management) 67
Figure 66: Sequence Diagram (Billing Management) 68
Chapter 8: Structural modeling 69
Figure 67: Deployment Diagram for Clinic Management System 69
Chapter 9: Conclusion 70

Chapter 1: Introduction

This document outlines the Software Requirements Specification (SRS) for the "Clinic Management System".

1.1 Purpose

The purpose of this document is to detail the functional, non-functional, and support requirements of the Clinic Management System. This system aims to improve healthcare operations by offering a digital platform that facilitates communication between patients, doctors, and administrators, ensuring streamlined clinic management and enhanced user experiences.

1.2 Intended Audience

This SRS is intended for the following audiences:

- Customers: To verify that the developed system meets their needs.
- Project Managers: To plan milestones and ensure the development process is on track.
- Designers: To use as a basis for system design.
- Developers: To develop the system functionalities.

1.3 Conclusion

This document provides a comprehensive overview of the project, ensuring all stakeholders have a clear understanding of the system requirements and their alignment with project goals.

Chapter 2: Inception

2.1 Introduction

The inception phase defines how a software project gets started and outlines the scope and nature of the problem to be solved. The goal of this phase is to identify stakeholder needs, resolve conflicts, and establish a foundation for the system's requirements. The following factors were considered:

- Identifying Stakeholders
- Recognizing Multiple Viewpoints
- Working Towards Collaboration
- Asking the First Questions

2.1.1 Identifying Stakeholders

Stakeholders are individuals or groups who will be directly or indirectly affected by the system. These include:

- Patients: Manage appointments, access medical records, and receive reminders.

- Doctors: Manage appointments and access patient information.
- Admins: Maintain clinic data, manage appointments, and oversee billing.
- Software Developers: Design, build, and maintain the system.

An initial list of stakeholders will evolve as more contributors are identified through discussions.

2.1.2 Recognizing Multiple Viewpoints

Different stakeholders have varied requirements from the system. To satisfy their needs, most features must cater to these perspectives:

Patients' Viewpoint

- Error-free system
- Strong authentication for data security
- Easy access to medical records and appointment scheduling
- View doctor list and prescriptions.
- Make secure payments.
- Secure handling of personal health data

Doctors' Viewpoint

- User-friendly interface for managing appointments.
- Reliable access to patient data
- Strong authentication for data security

Admins' Viewpoint

- Comprehensive data management
- Accurate billing and financial record-keeping
- Ability to monitor doctor availability.
- Strong authentication for data security.

Developers' Viewpoint

- Clear and conflict-free requirements
- Adequate resources for development
- Secure and reliable design parameters

2.1.3 Working Towards Collaboration

To reconcile conflicting requirements and prioritize features, the following steps were followed:

1. Identify common and conflicting requirements.
2. Categorize the requirements.

3. Use stakeholder feedback to prioritize requirements.
4. Make final decisions collaboratively.

Common Requirements:

- Web-based interface accessible from any computer with internet access.
- Maintain a database of users, appointments, and medical records.
- Enable users to search for doctors.

Conflicting Requirements:

- Debate over requiring login for accessing general clinic information.
- Strong authentication challenges during patient check-in.

Final Requirements:

- Error-free, easily accessible system.
- Role-based access control for different users.
- Allow logged-in users to book and manage appointments.
- Enable administrators to oversee system parameters and user roles.
- Support online options.

2.1.4 Asking the First Questions

Initial inquiries focused on:

- Stakeholder needs and project goals.
- Specific challenges and constraints.
- Identifying priority features and requirements.

2.2 Conclusion

The inception phase establishes a foundational understanding of the Clinic Management System's objectives, guiding subsequent development phases.

Chapter 3: Elicitation

3.1 Introduction

Requirements elicitation is the process of gathering and refining the system requirements from stakeholders. This ensures the developed system aligns with stakeholder expectations and project goals.

3.2 Eliciting Requirements

This phase combines problem-solving, elaboration, and specification. Key tasks include:

- Collaborative requirements gathering.
- Quality function deployment.

- Defining use case scenarios.

3.3 Collaborative Requirements Gathering

Regular meetings and discussions with stakeholders were conducted to:

- Identify system needs and expectations.
- Resolve ambiguities and conflicts in requirements.
- Prioritize critical functionalities.

3.4 Quality Function Deployment (QFD)

QFD translates customer needs into technical requirements, focusing on maximizing satisfaction.

3.4.1 Normal Requirements

The normal requirements are the objectives and goals stated for a product or system during meetings with the customer. The presence of these requirements fulfils customers' satisfaction. These are the normal requirements for the Clinic Management System:

1. Provide role-based access control for functionalities.
2. Enable logged-in users to book and cancel appointments.
3. Provide detailed medical records to patients and authorized personnel.
4. Maintain a centralized database for storing patient, doctor, and appointment information.
5. Maintain accurate financial records, including billing and payment details.
6. Track doctor availability and allocate appointments accordingly.
7. Ensure data security and privacy compliance.
8. Allow online payment integration for billing.
9. Provide features for clinic staff to manage inventory and supplies.

3.4.2 Expected Requirements

These requirements are intrinsic to the system and may not always be explicitly stated by customers. Their absence, however, would result in significant dissatisfaction. Below are the expected requirements for the Clinic Management System:

1. Error-free software operation.
2. Robust authentication system to ensure secure access.
3. Intuitive and user-friendly interface for all user roles.
4. Effective system with quick response times.
5. Clearly defined and unambiguous features.
6. Alerts for actions.
7. Support for updating patient and clinic data by authorized personnel.
8. The system should automatically check doctor availability.

3.4.3 Exciting Requirements

These requirements exceed customer expectations and greatly enhance satisfaction when implemented. The exciting requirements for the Clinic Management System include:

1. Providing detailed and helpful error messages for invalid inputs.
2. Aesthetic and intuitive user interface consistent with modern web applications.
3. Personalized dashboards for patients, doctors, and administrators.

Chapter 4: Scenario-Based Modelling

4.1 Introduction

Scenario-based modelling describes the system from the user's perspective. It serves as input for other modelling elements and ensures alignment between design and user requirements.

4.2 Usage Scenario

The Clinic Management System offers a digital platform to improve patient experience and conduct healthcare operations by providing smooth communication between patients, doctors, and administrators.

Patients can manage their medical care by registering and logging into an account through the application. They can schedule appointments with available doctors by choosing a time and date that works for them. Additionally, patients can access diagnostic findings, view billing information, and read their profile. To select the appropriate expert for their needs, patients can use the program to peruse a list of doctors.

Doctors can view their appointments by logging in, because of this they can be prepared for the appointments on time, and they can request to cancel appointments.

Administrators use the system to maintain patient and doctor data, guaranteeing that the latest information is always accessible. They can view patient data, add, and delete doctor availability. Also, they can approve doctors request to cancel appointments. The technology guarantees a smooth billing process for patients by enabling administrators to create and view billing records.

This system streamlines clinic administration, allowing administrators to maintain effective healthcare facility and physicians to schedule appointments, and patients to get care conveniently.

4.3 Use Case Scenario

Level 0	Level 1	Level 2
Clinic Management System	Authentication	Log In
		Log Out
	Patient Management	Book Appointment
		View Profile
		Update Profile
		Access Prescription
		View Doctor List
	Doctor Management	View Appointments
		Cancel Appointments
		View Patient list
		View Profile
		Update Profile
	Admin Management	View Patients
		View Doctors
		Add Doctor
		Delete Doctor
		Remove Appointment
	Billing Management	Generate Bill
		View Bills
		Pay Bill
	Appointment Management	Schedule Appointment

interactions between users and the system. Key scenarios include:

- Authentication.
- Appointment management.
- Billing and payment processing.

4.4 Use Case Diagram

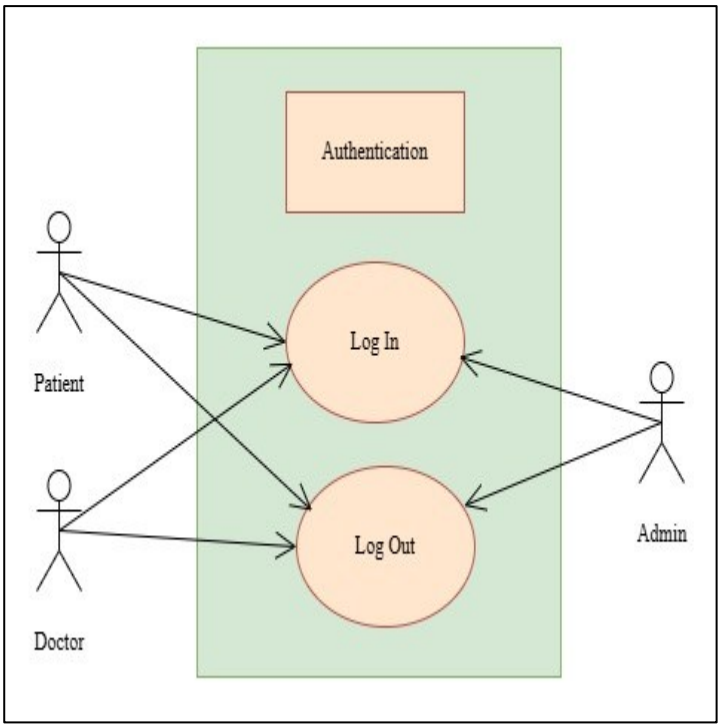

Figure 1: Authentication Module

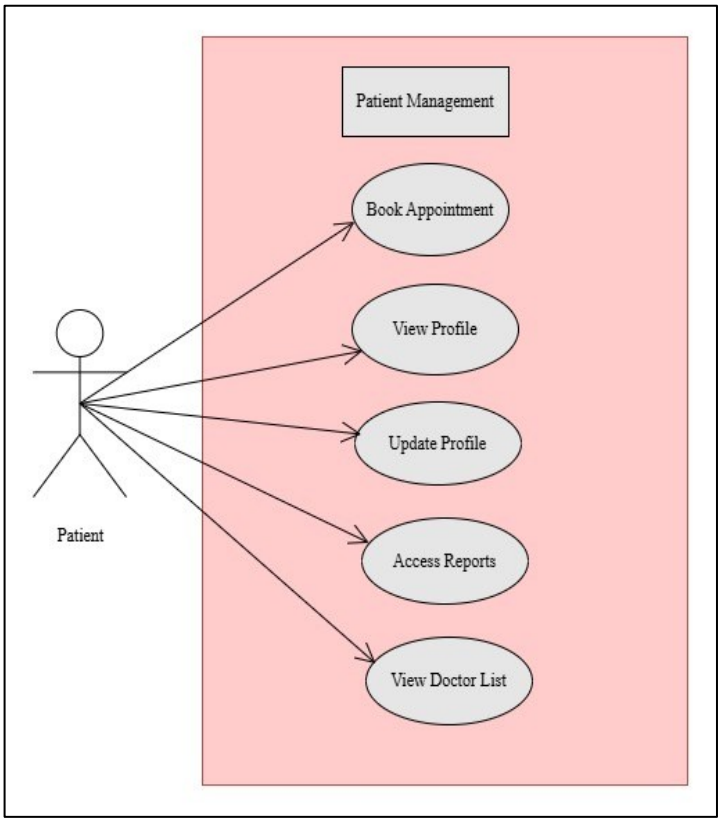

Figure 2: Patient Module

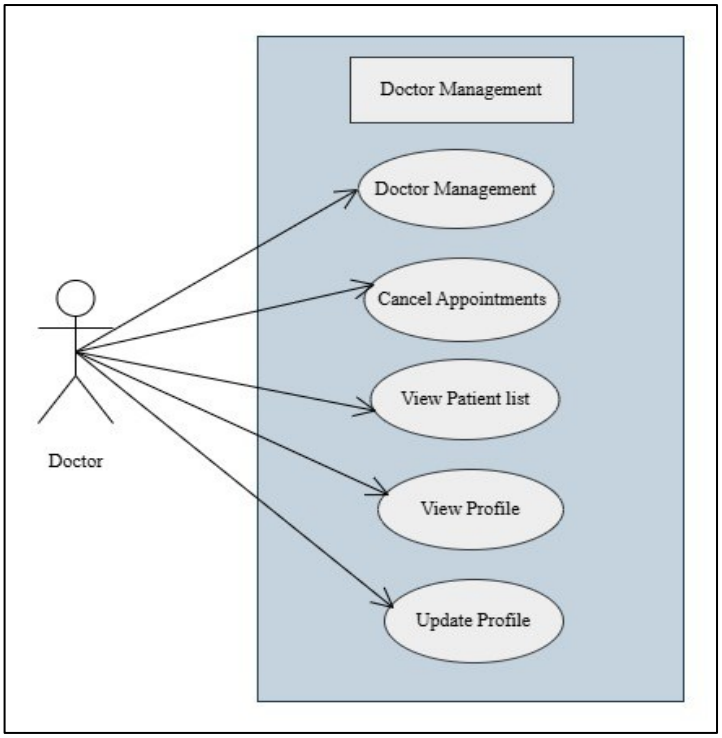

Figure 3: Doctor Module

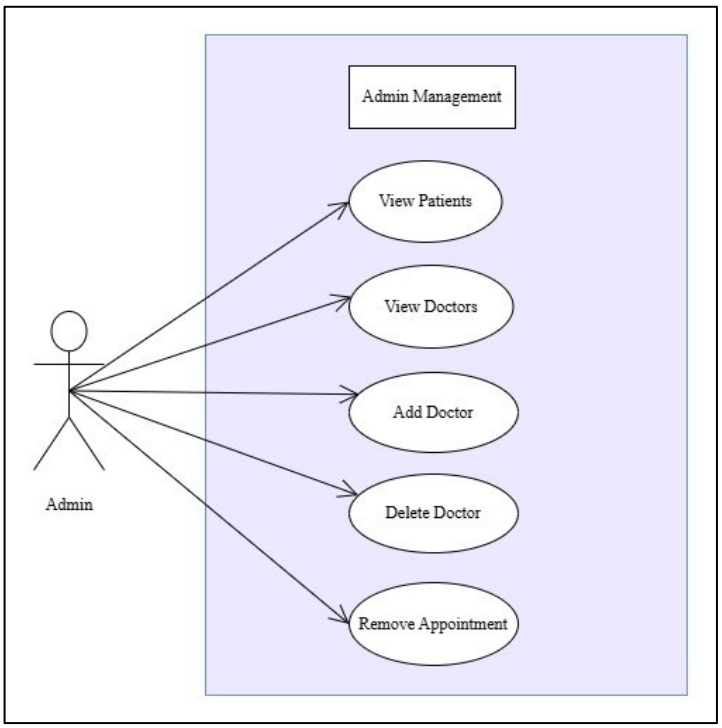

Figure 4: Admin Module

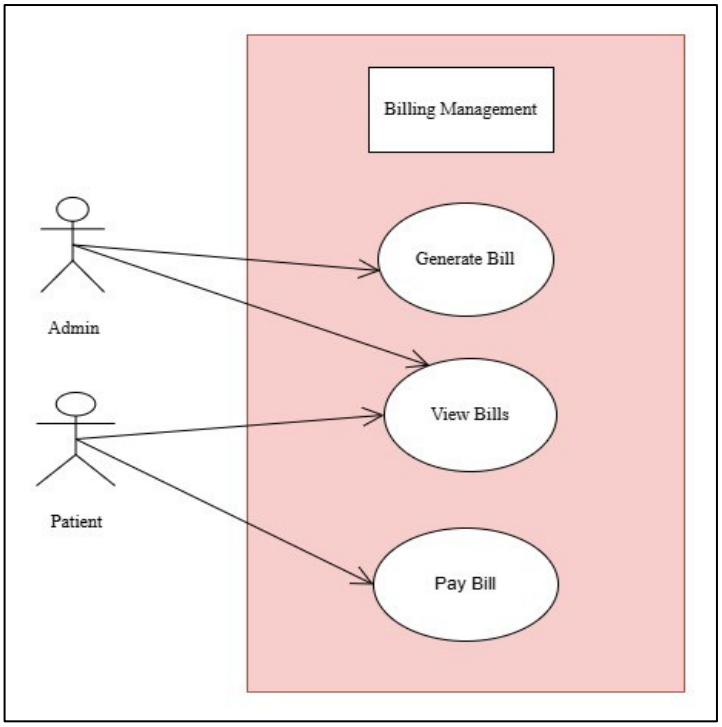

Figure 5: Billing Module

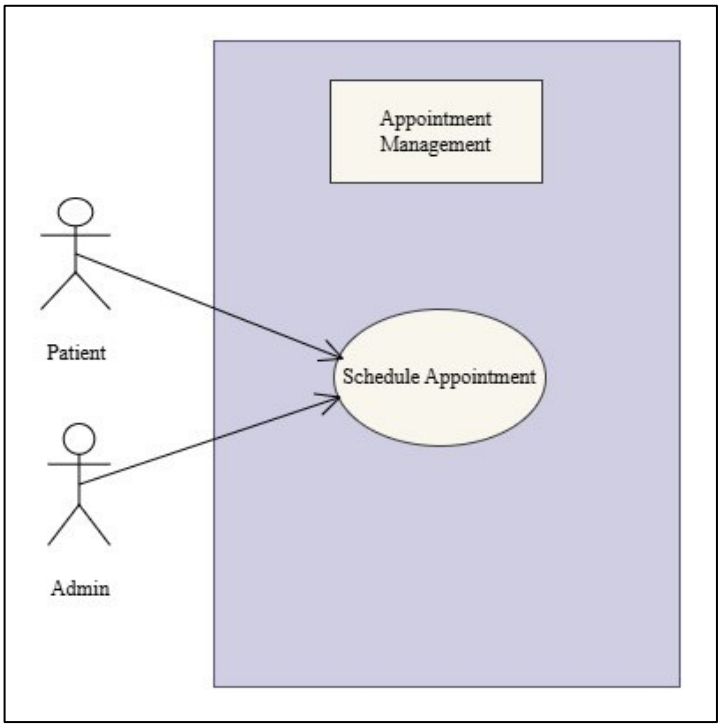

Figure 6: Appointment Module

4.5 Activity Diagram and Swimlane Diagram of generated Use Cases

Use case 1: Log In

Activity Diagram:

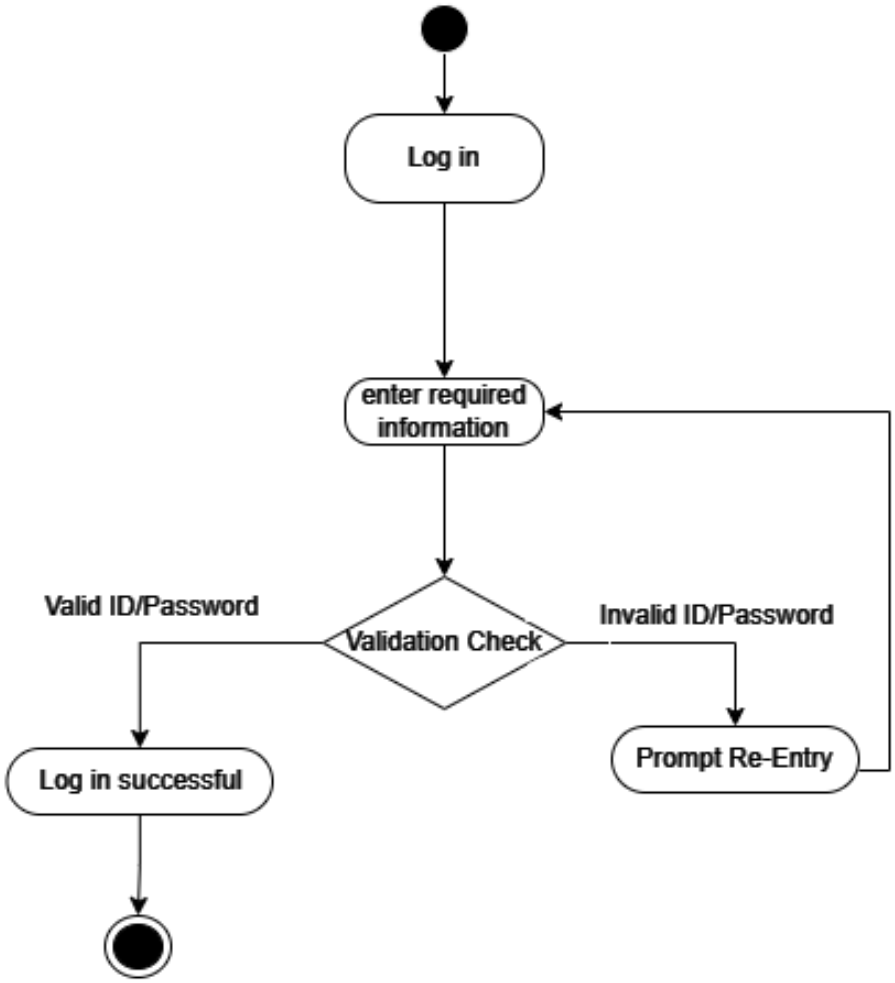

Figure 7: Activity Diagram Log In

Swimlane Diagram:

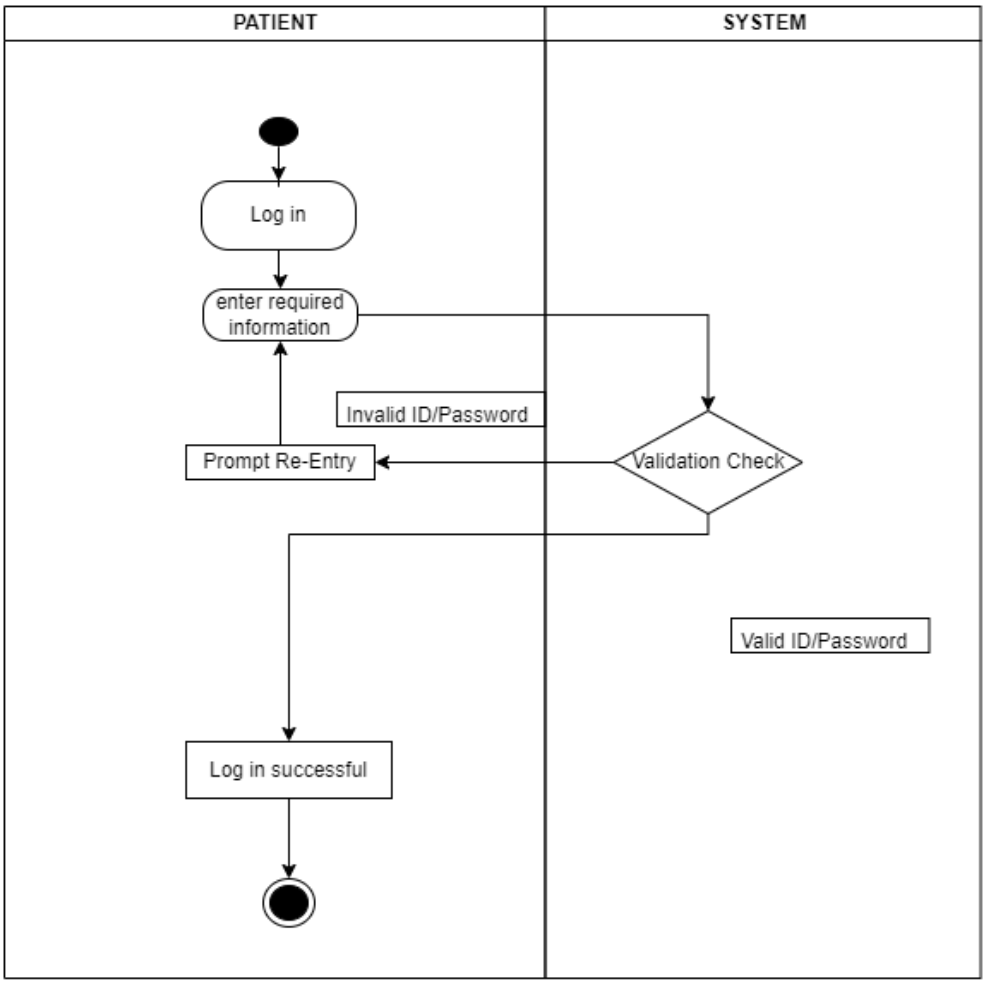

Figure 8: Swimlane Diagram for Log In

Use case 2: Log Out

Activity Diagram:

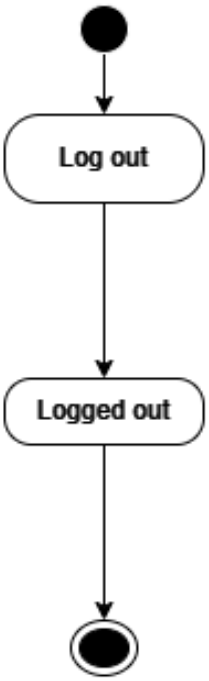

Figure 9: Activity Diagram Log Out

Swimlane Diagram:

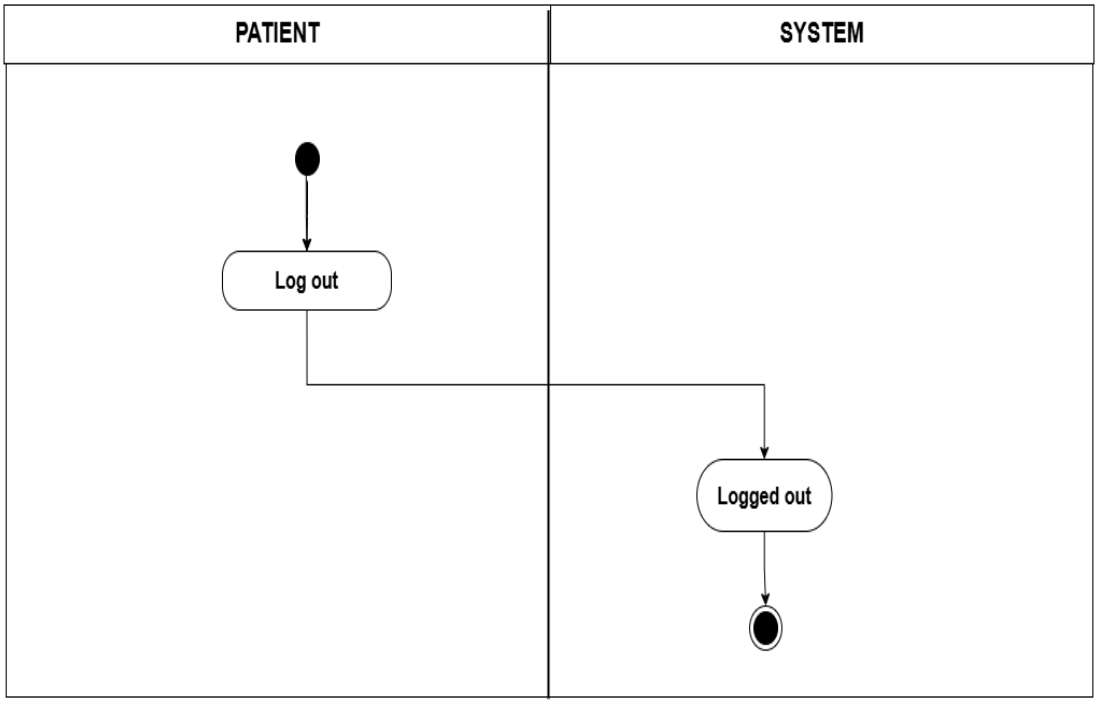

Figure 10: Swimlane Diagram for Log Out

Use case 3: Book Appointment

Activity Diagram:

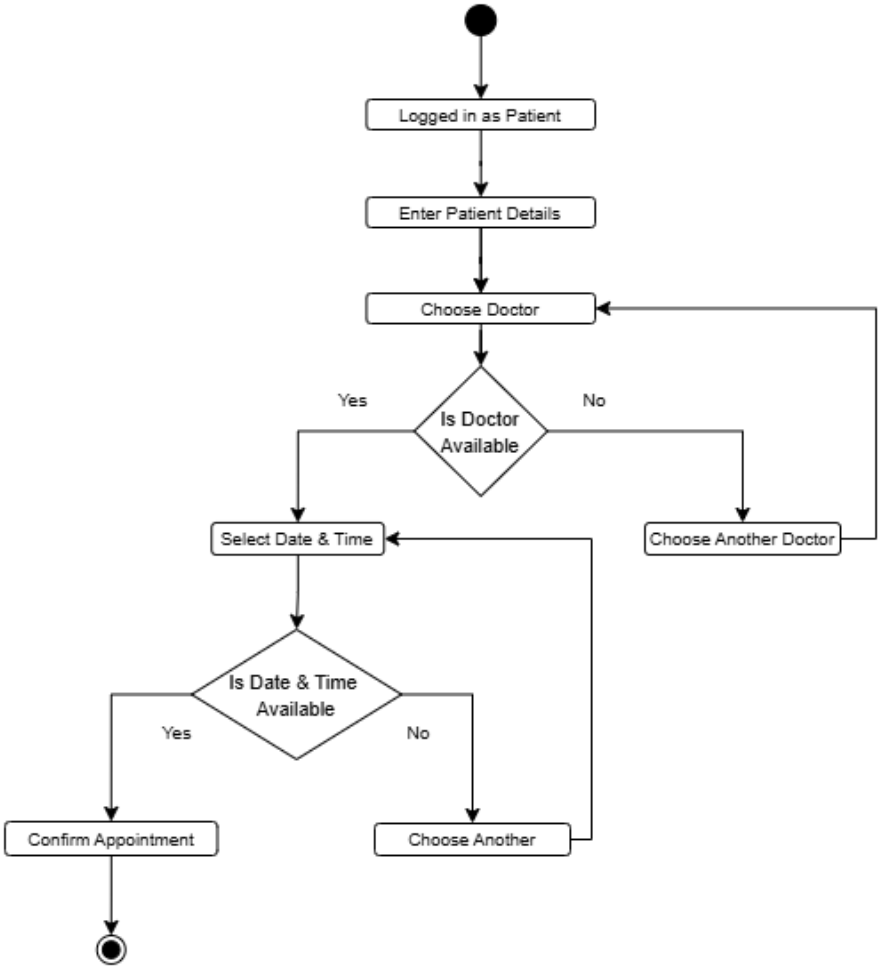

Figure 11: Activity Diagram Book Appointment

Swimlane Diagram:

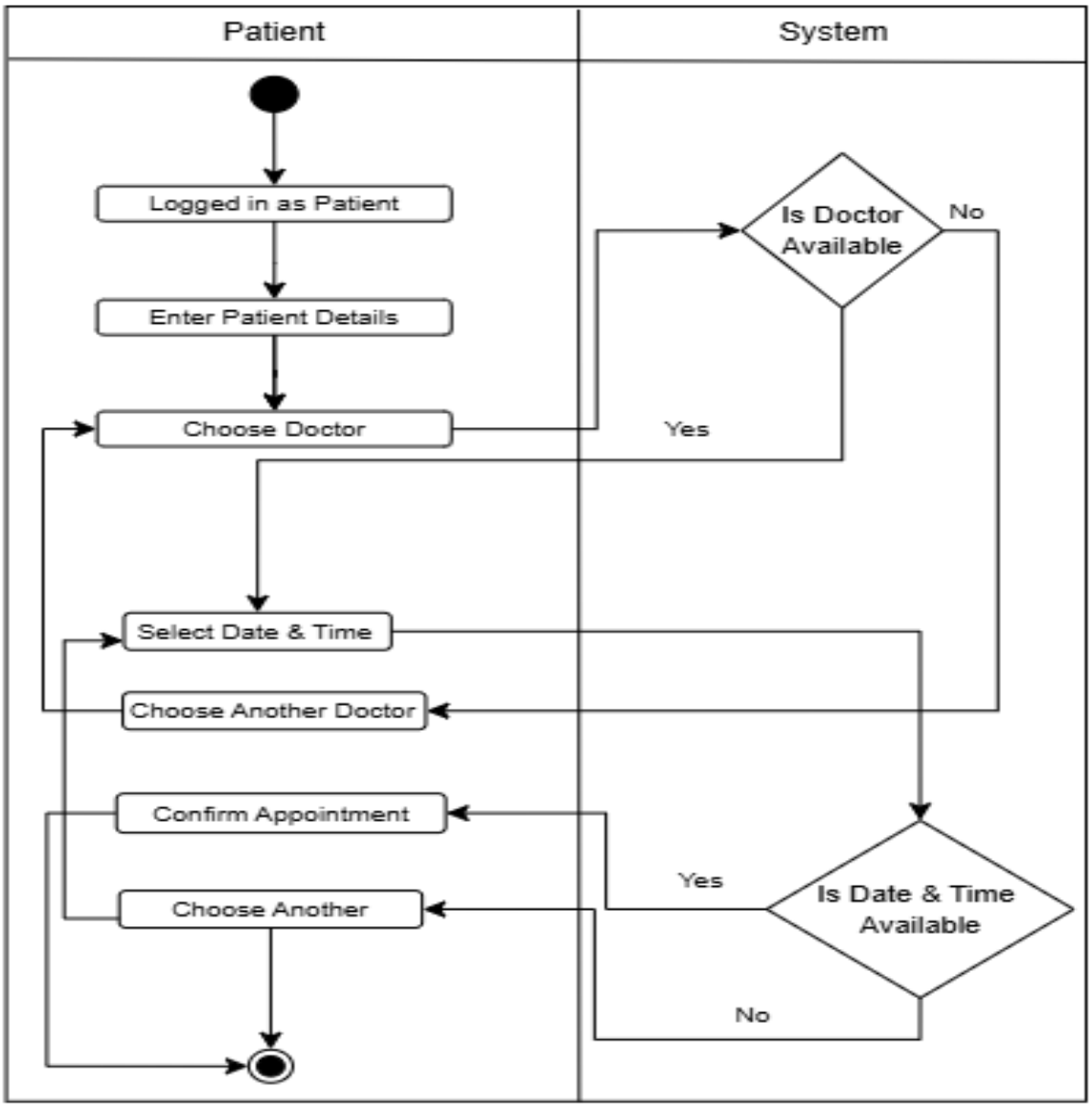

Figure 12: Swimlane Diagram for Book Appointment

Use case 4: View Profile (Patient)

Activity Diagram:

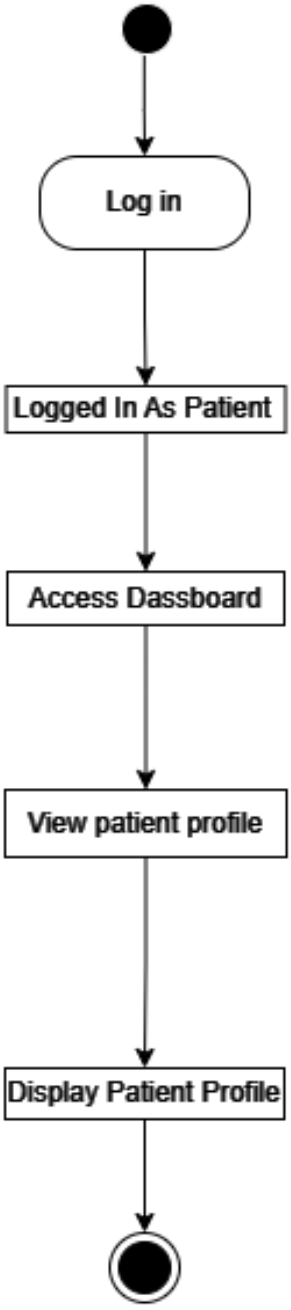

Figure 13: Activity Diagram for View Profile (Patient)

Swimlane Diagram:

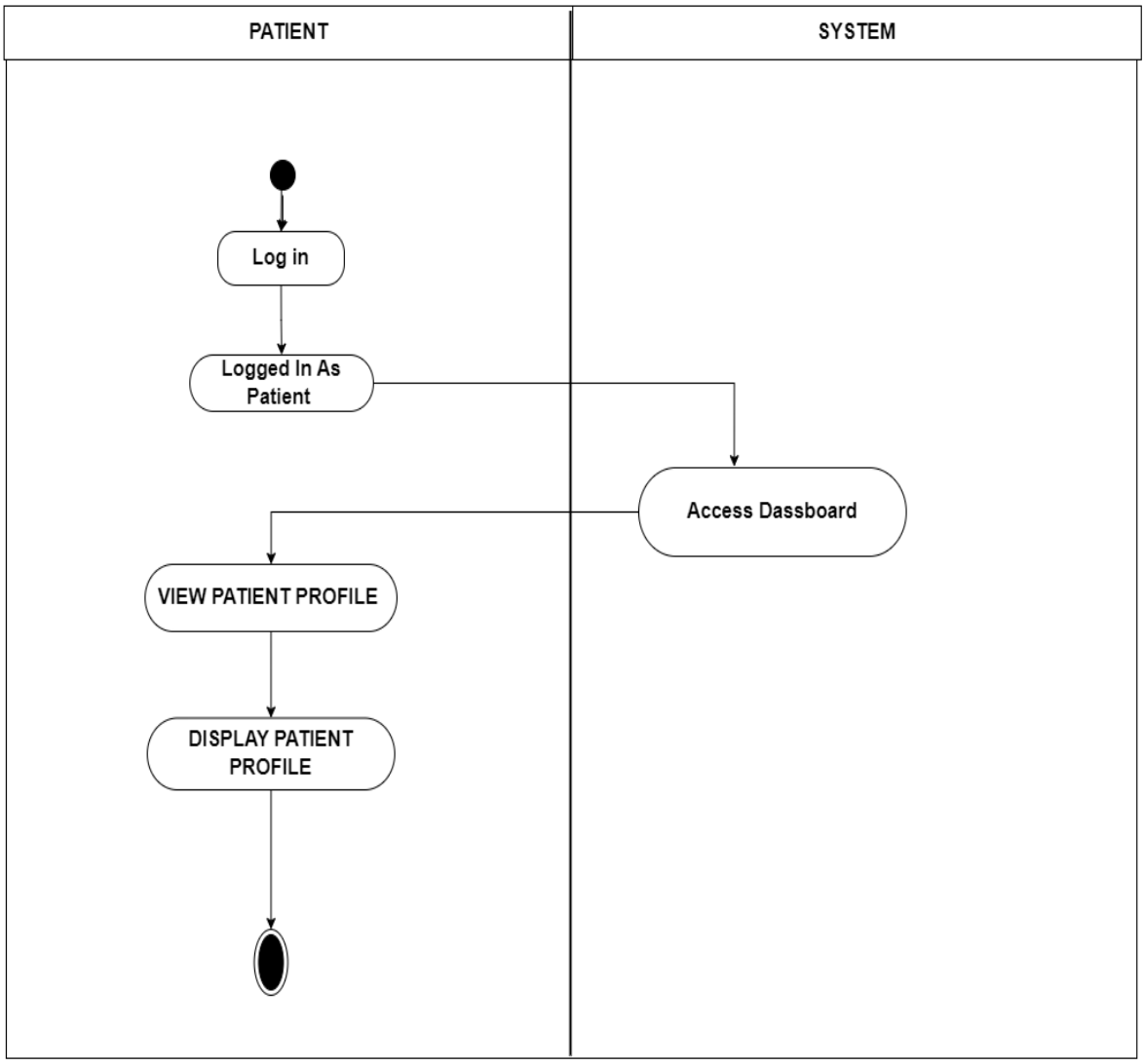

Figure 14: Swimlane Diagram for View Profile (Patient)

Use case 5: Update Profile (Patient)

Activity Diagram:

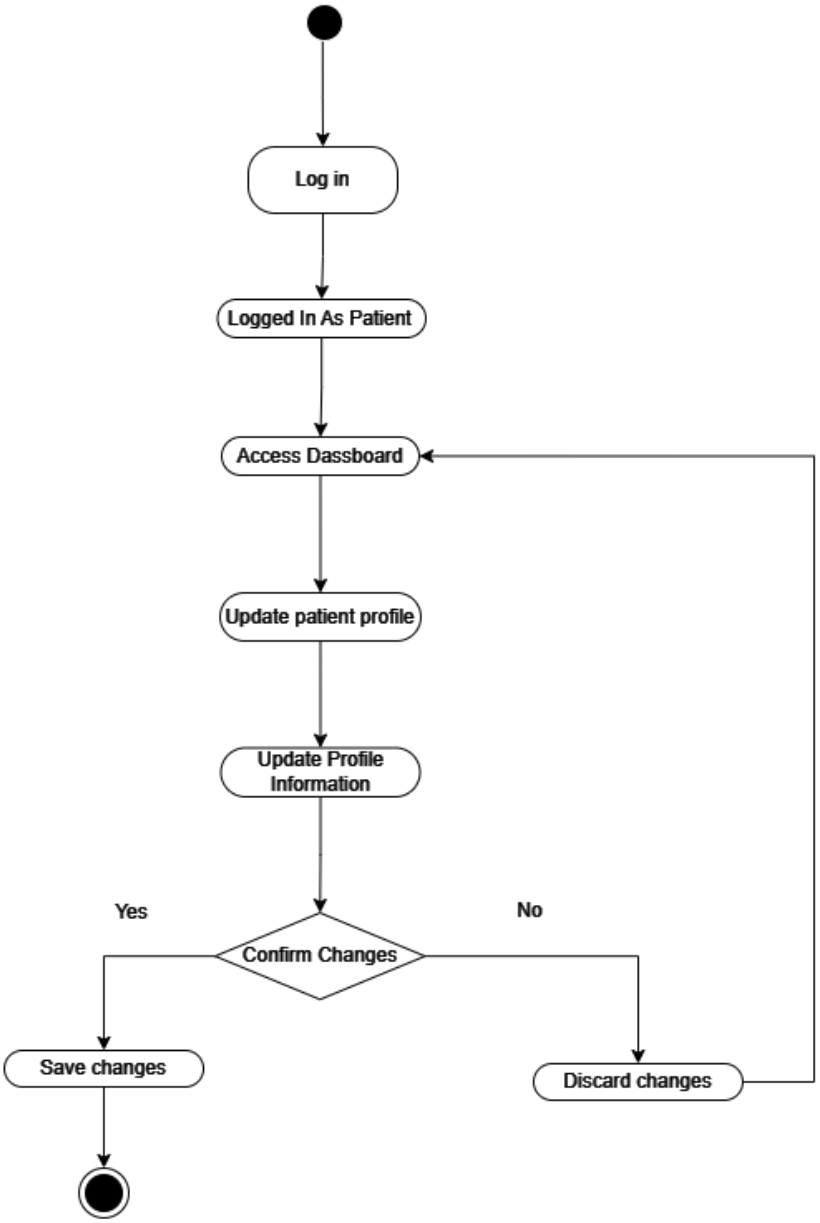

Figure 15: Activity Diagram for Update Profile (Patient)

Swimlane Diagram:

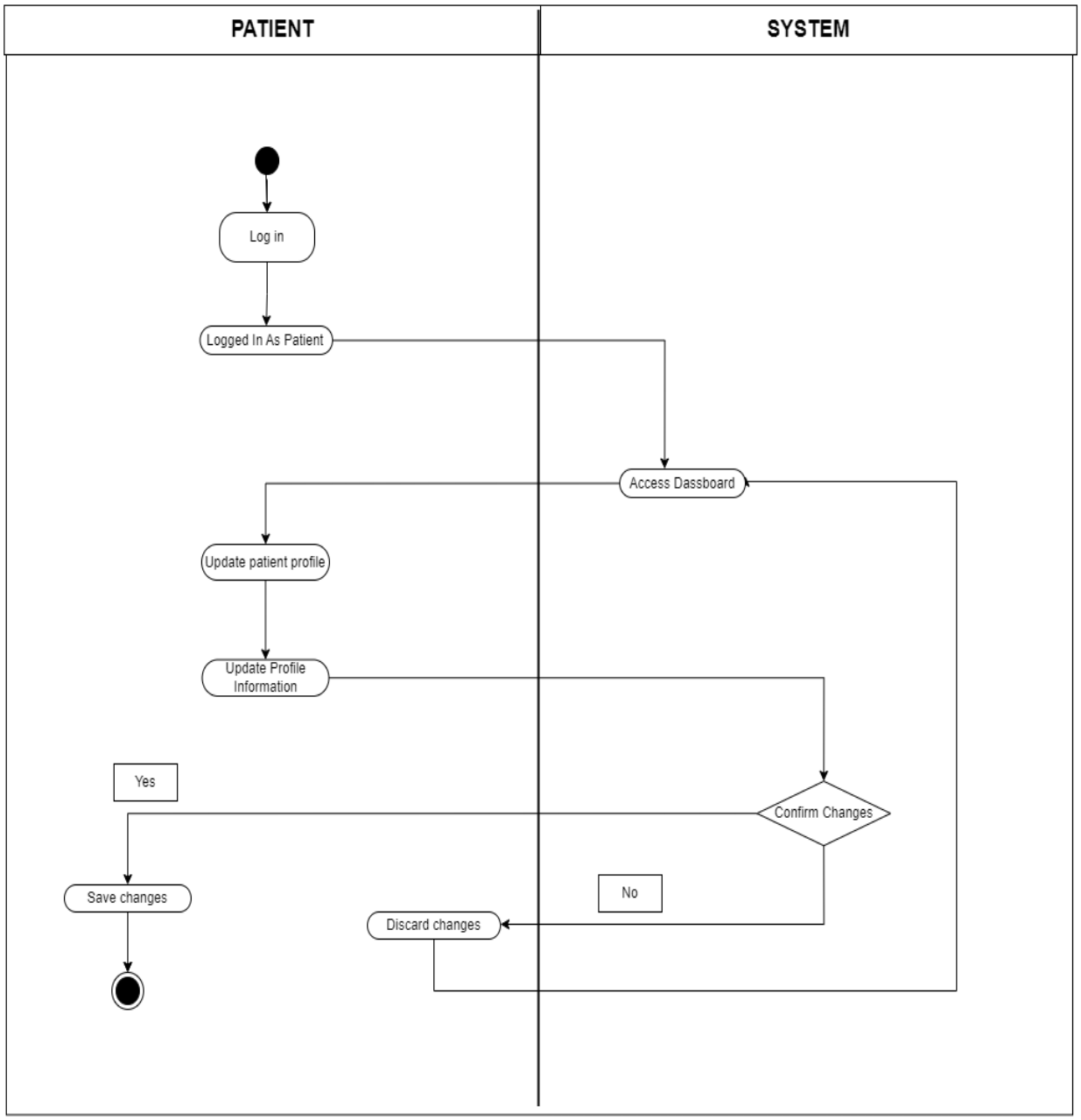

Figure 16: Swimlane Diagram for Update Profile (Patient)

Use case 6: Access Reports

Activity Diagram:

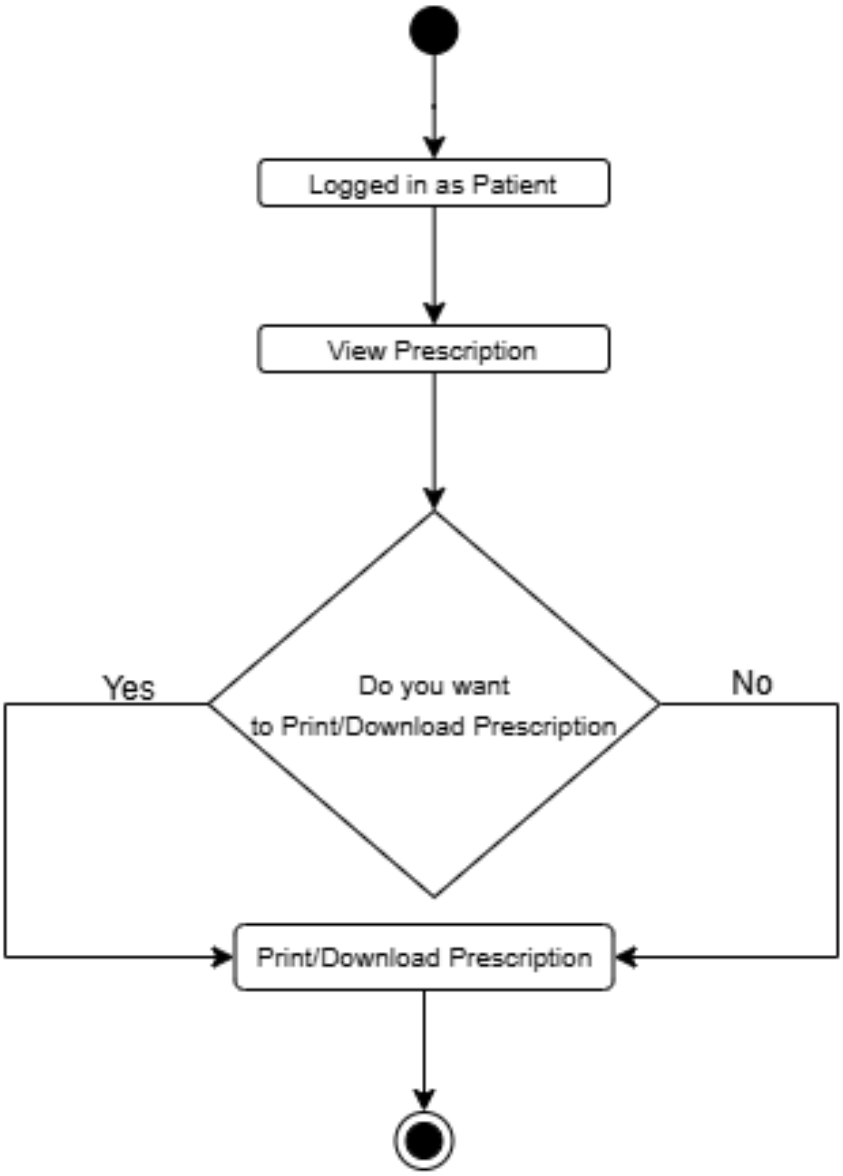

Figure 17: Activity Diagram for Access Prescription

Swimlane Diagram:

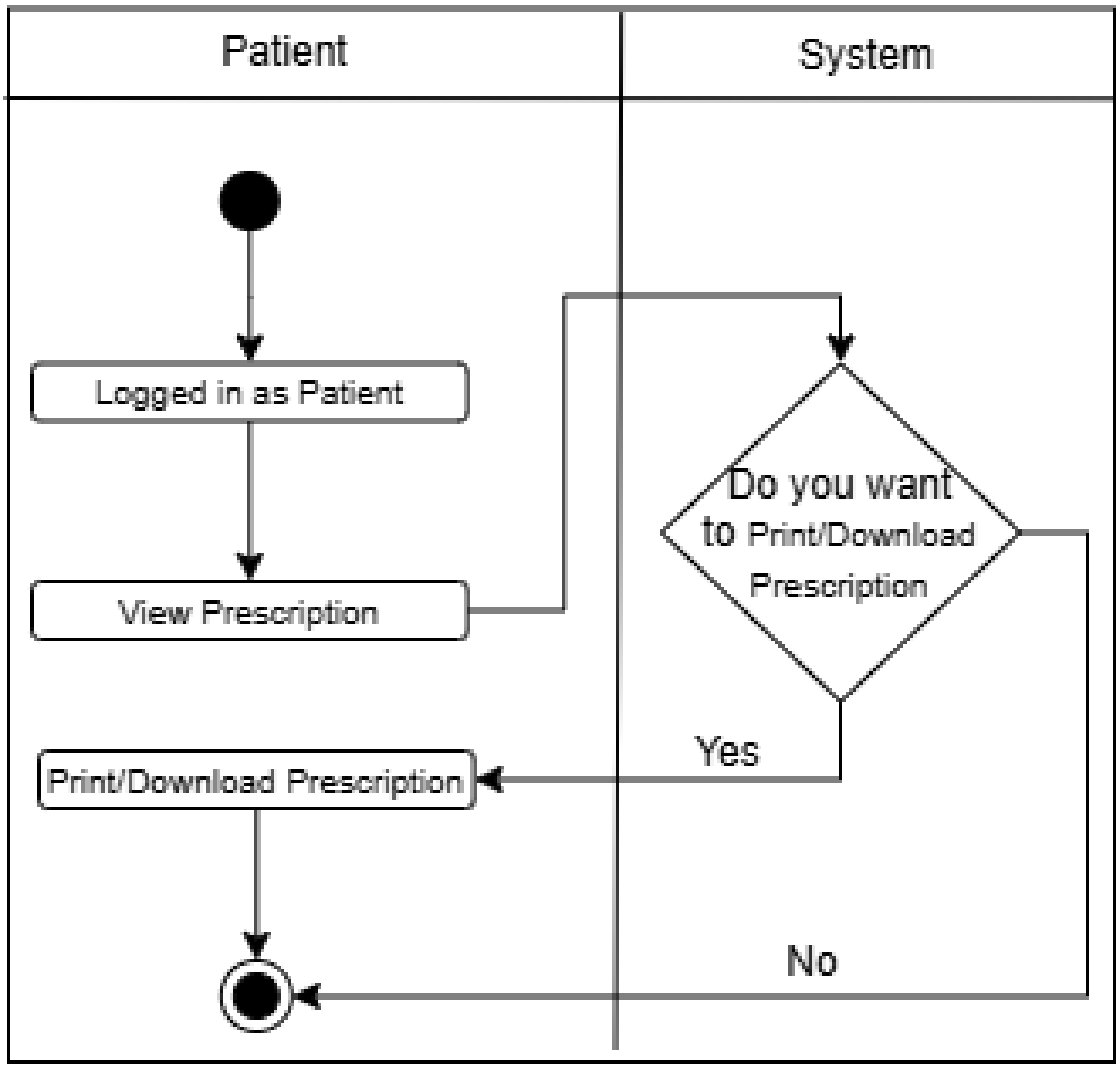

Figure 18: Swimlane Diagram for Access Prescription

Use case 7: View Doctor List

Activity Diagram:

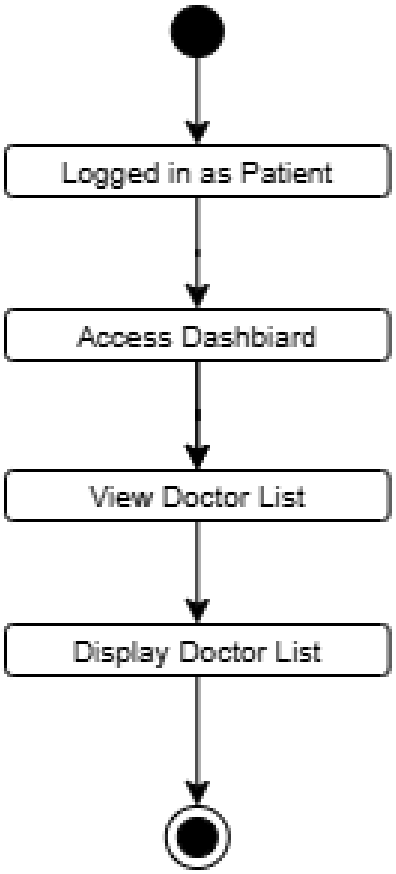

Figure 19: Activity Diagram for View Doctor List

Swimlane Diagram:

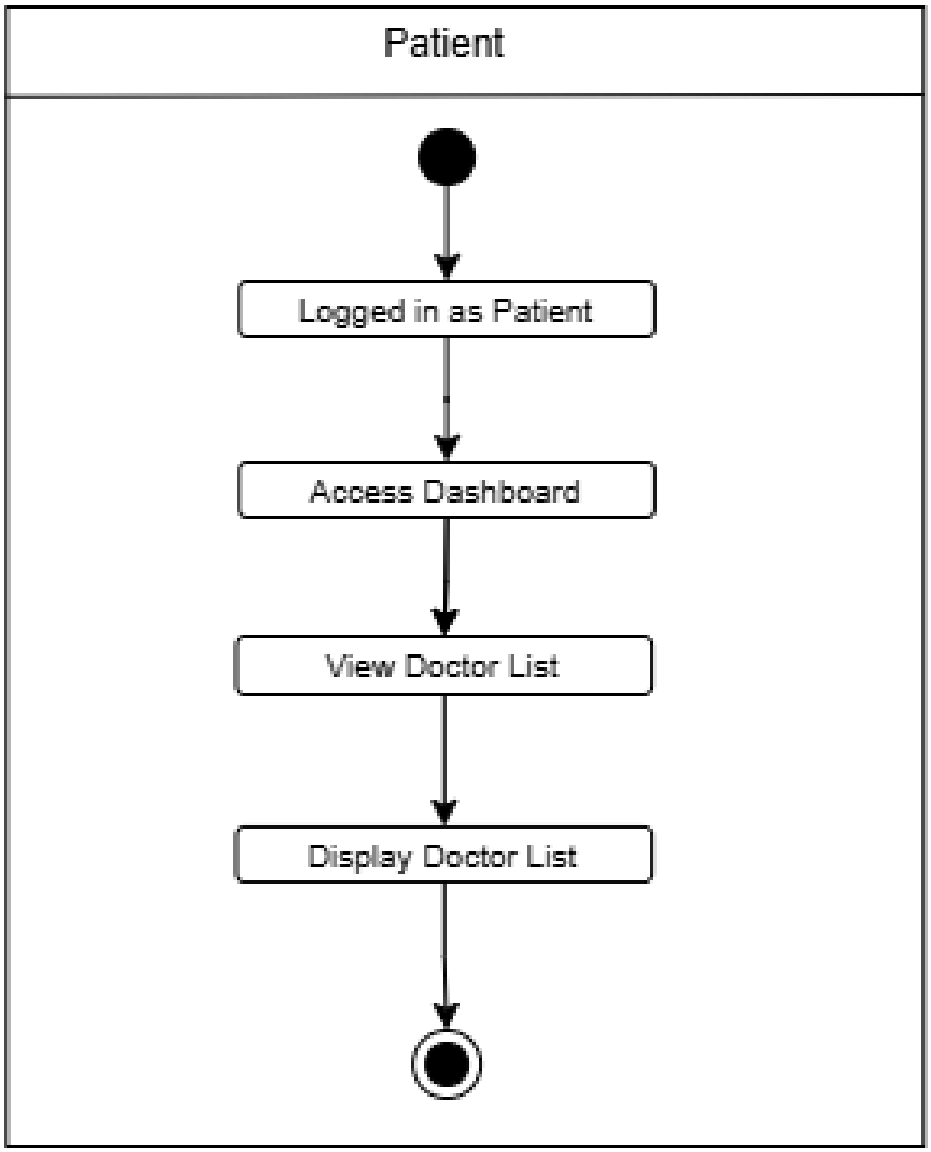

Figure 20: Swimlane Diagram for View Doctor List

Use case 8: View Appointments

Activity Diagram:

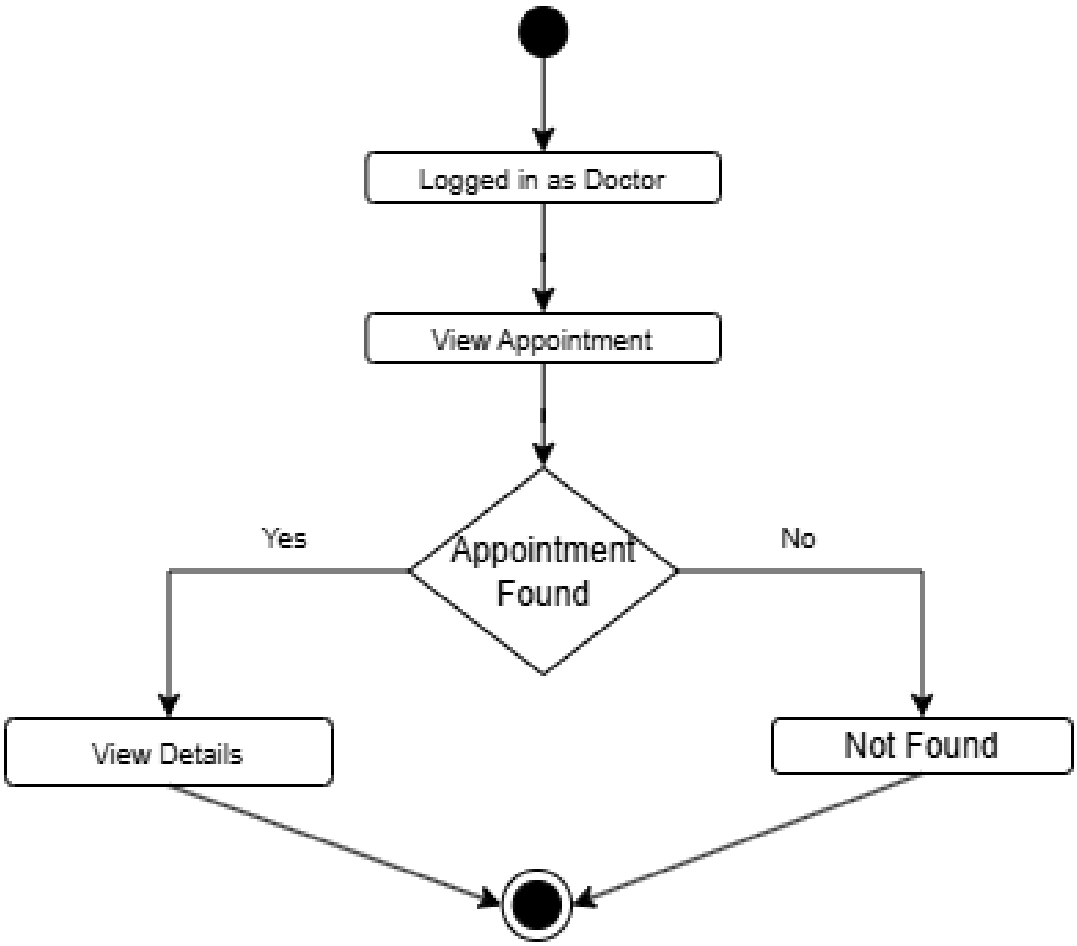

Figure 21: Activity Diagram for View Appointments

Swimlane Diagram:

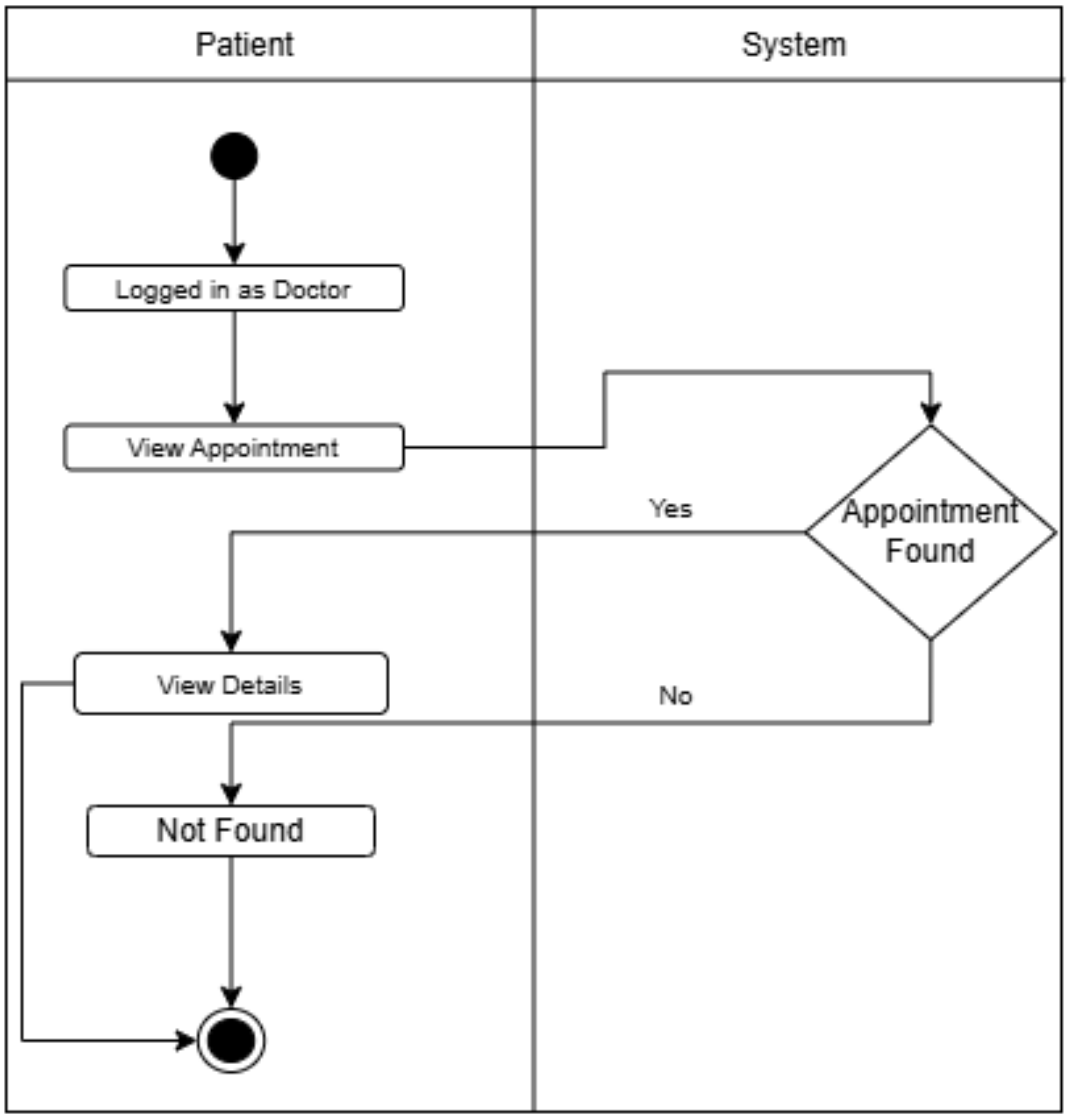

Figure 22: Swimlane Diagram for View Appointments

Use case 9: Cancel Appointments

Activity Diagram:

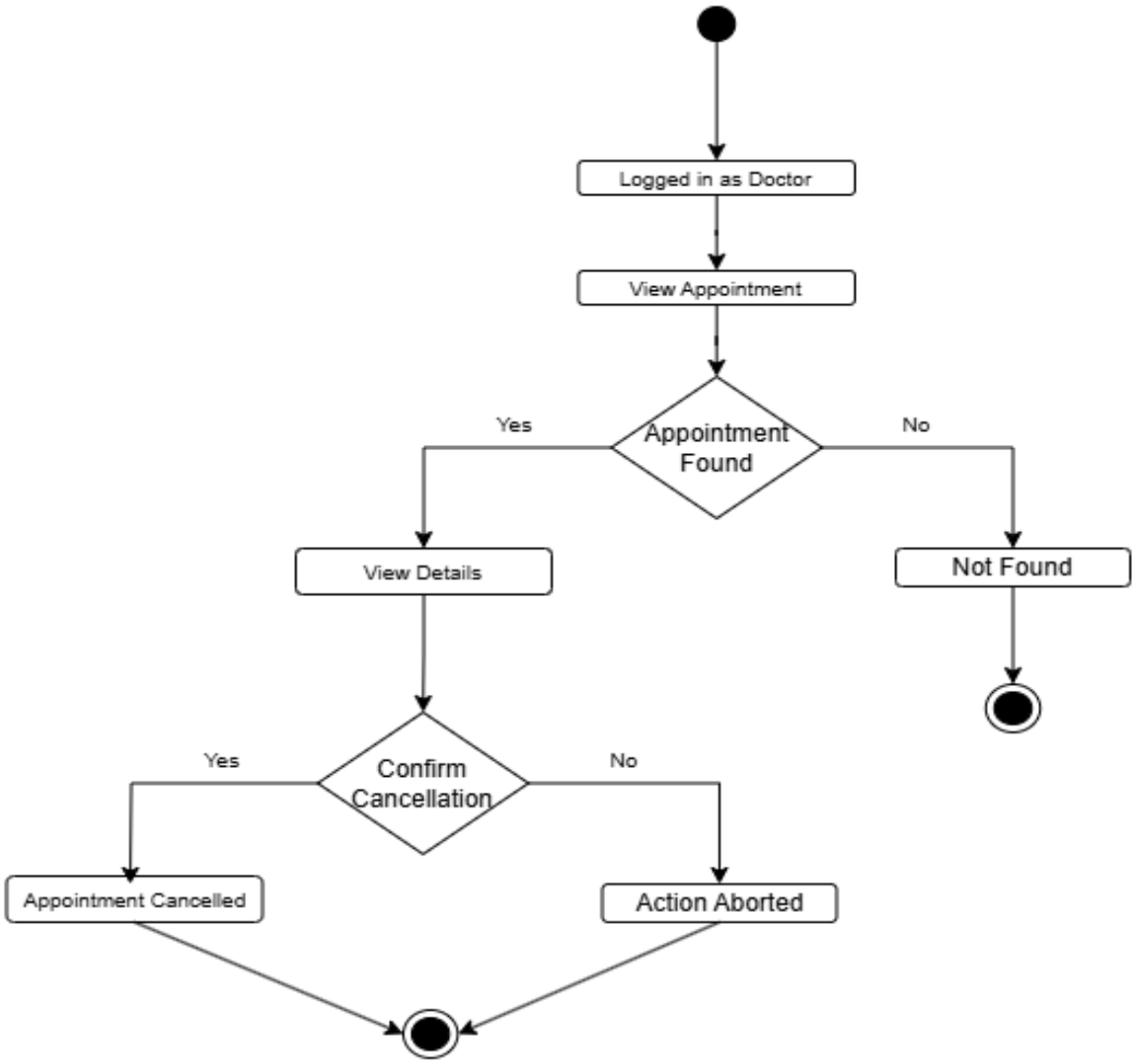

Figure 23: Activity Diagram for Cancel Appointments

Swimlane Diagram:

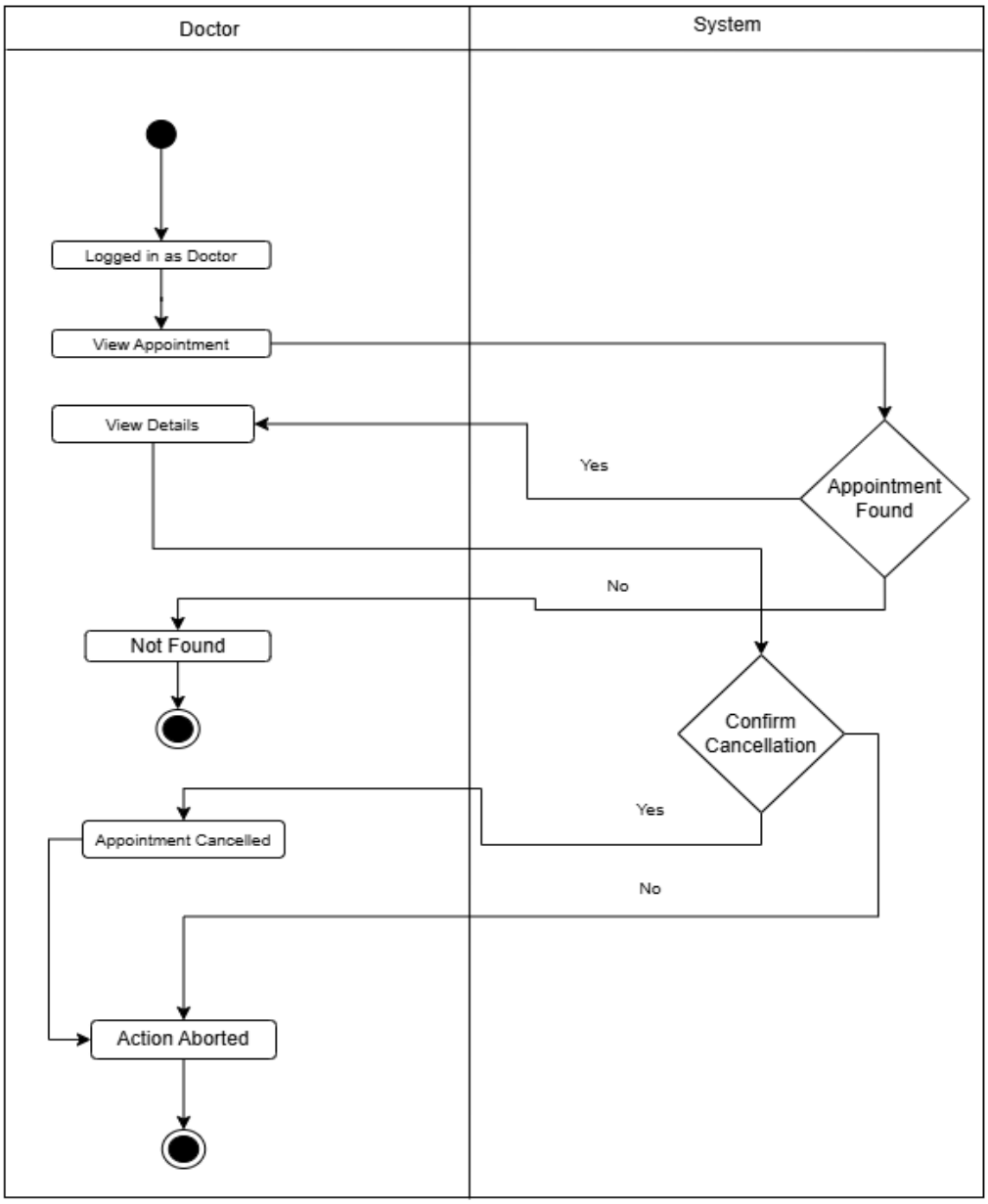

Figure 24: Swimlane Diagram for Cancel Appointments

Use case 10: View Patient List (Doctor)

Activity Diagram:

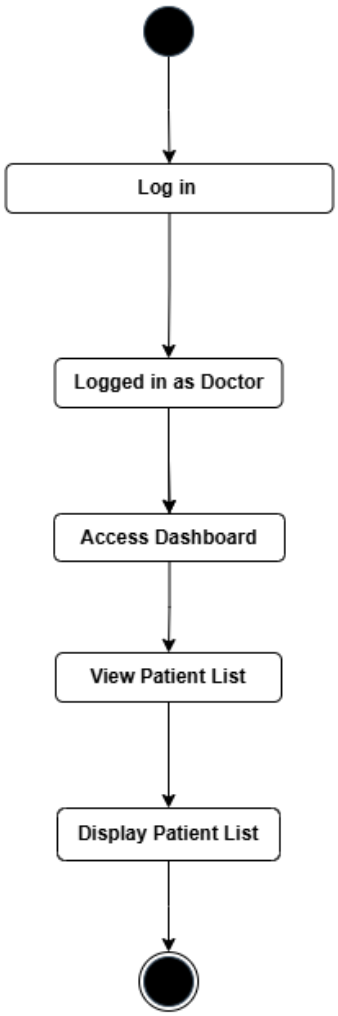

Figure 25: Activity Diagram for View Patient List (Doctor)

Swimlane Diagram:

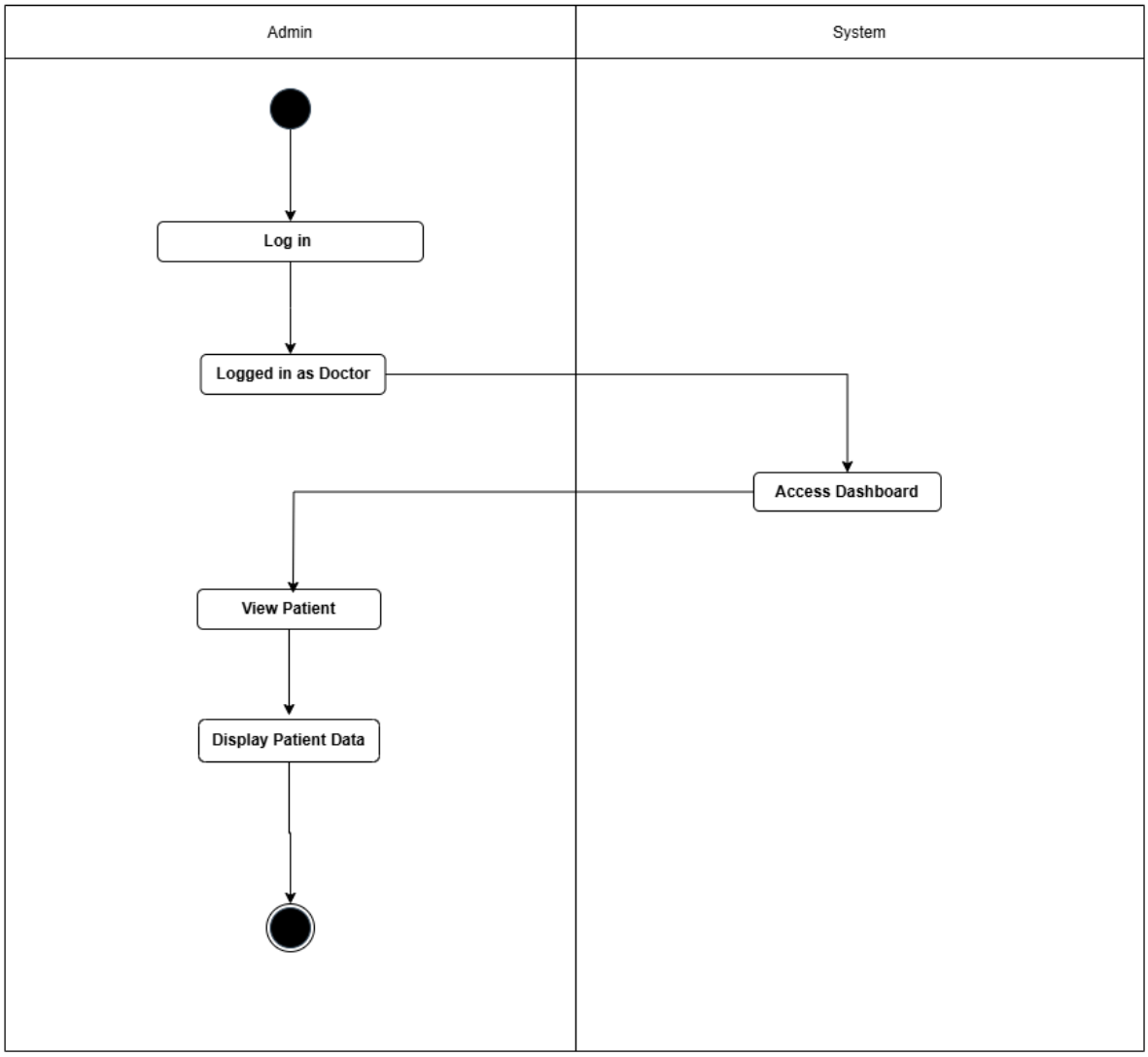

Figure 26: Swimlane Diagram for View Patient List 9Doctor0

Use case 11: View Profile (Doctor)

Activity Diagram:

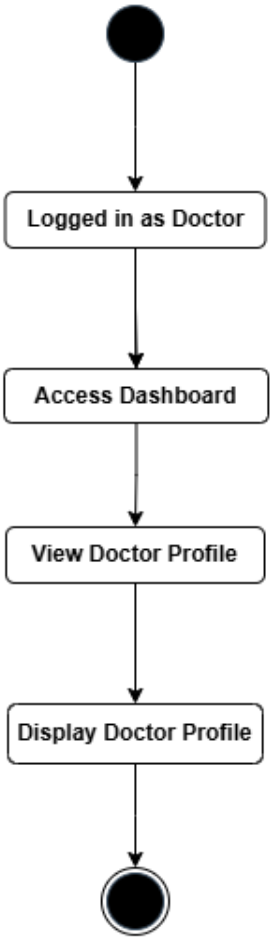

Figure 27: Activity Diagram for View Profile

Swimlane Diagram:

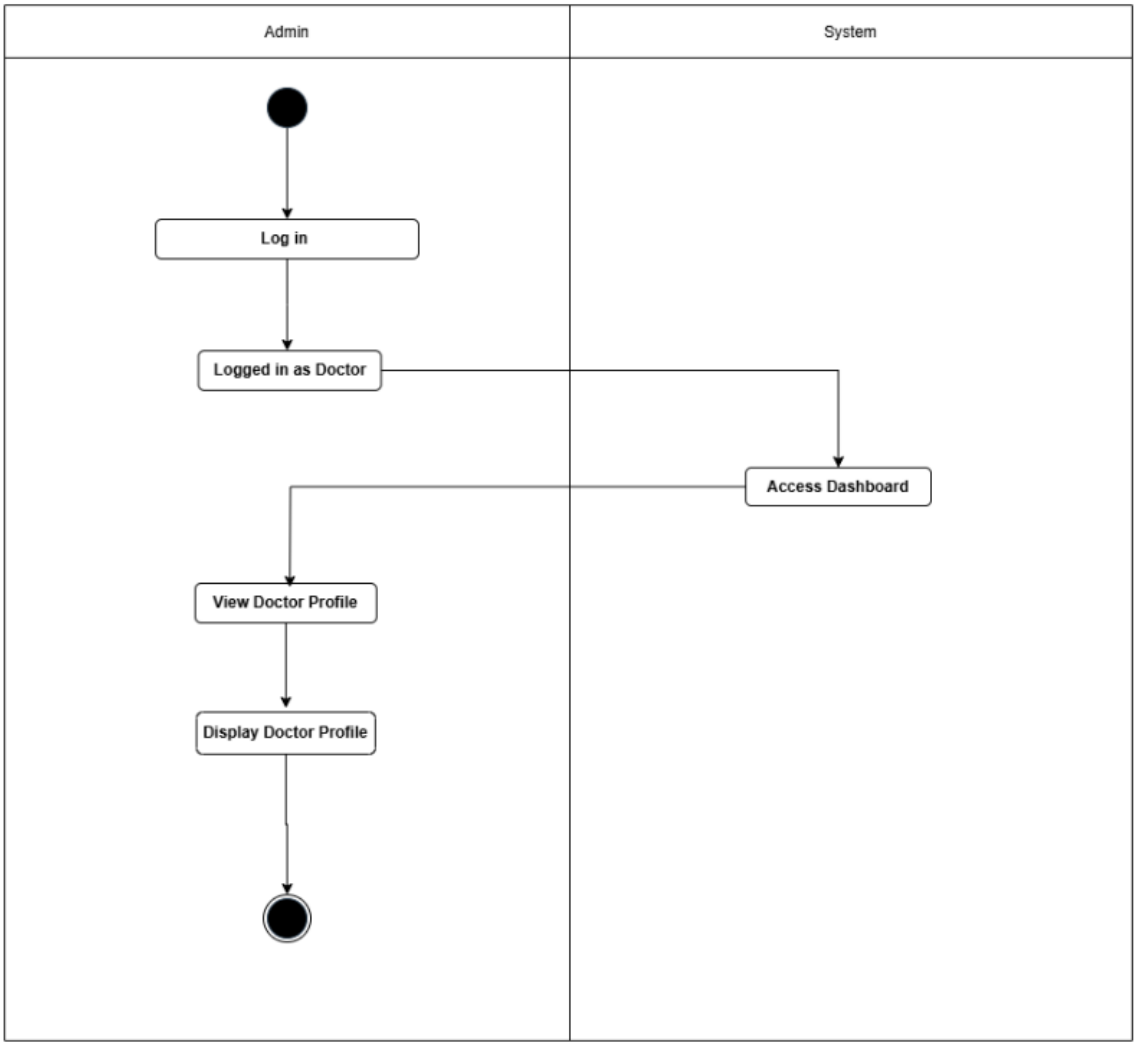

Figure 28: Swimlane Diagram for View Profile

Use case 12: Update Profile (Doctor)

Activity Diagram:

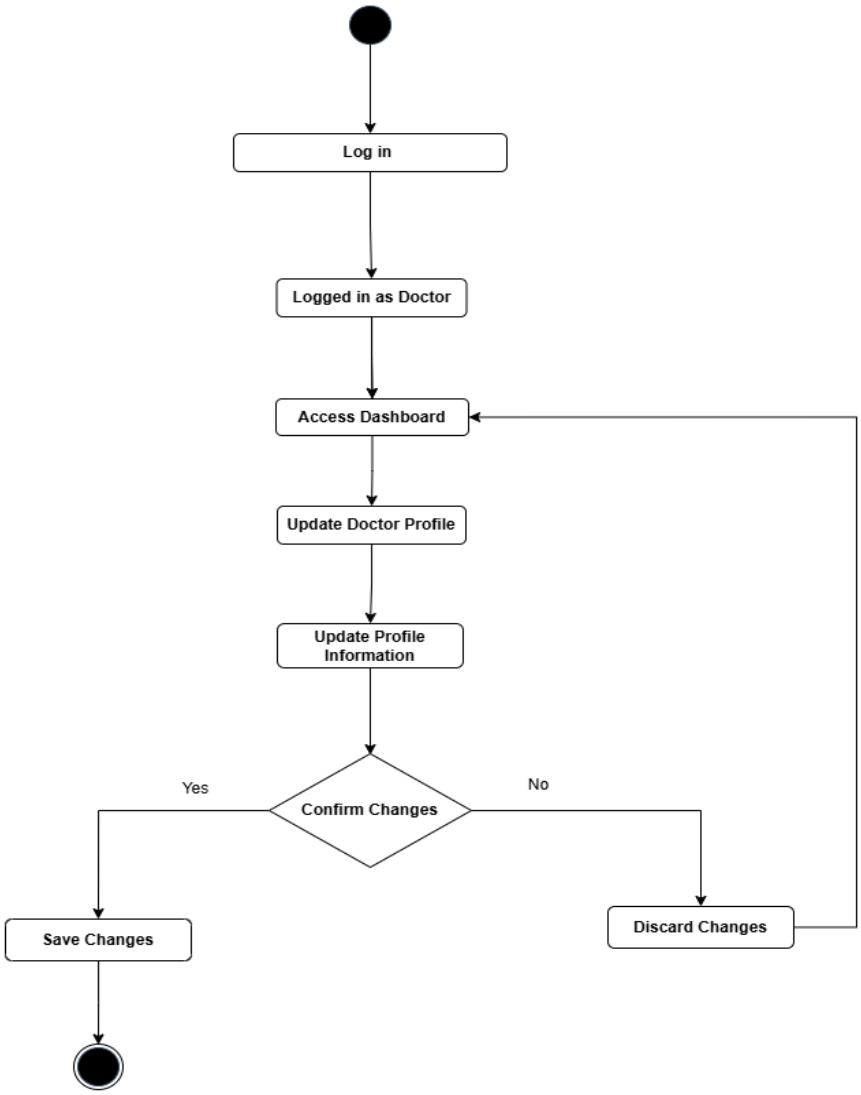

Figure 29: Activity Diagram for Update Profile (Doctor)

Swimlane Diagram:

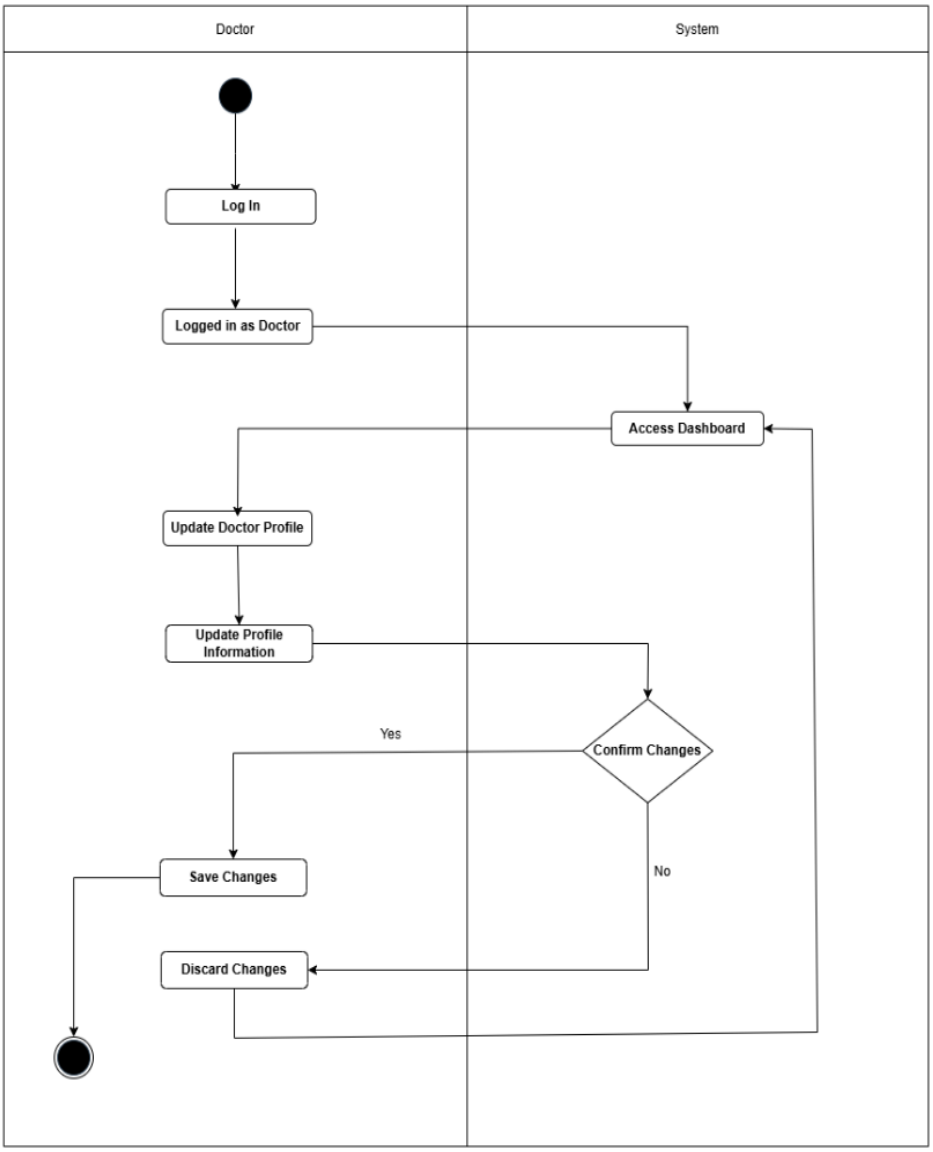

Figure 30: Swimlane Diagram for Update Profile (Doctor)

Use case 13: View Patients (Admin)

Activity Diagram:

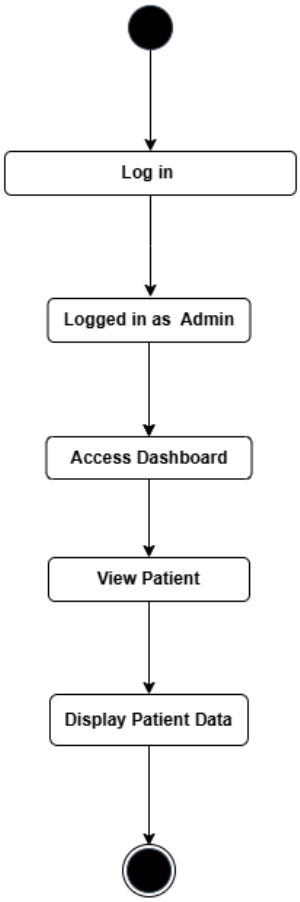

Figure 31: Activity Diagram for View Patients (Admin)

Swimlane Diagram:

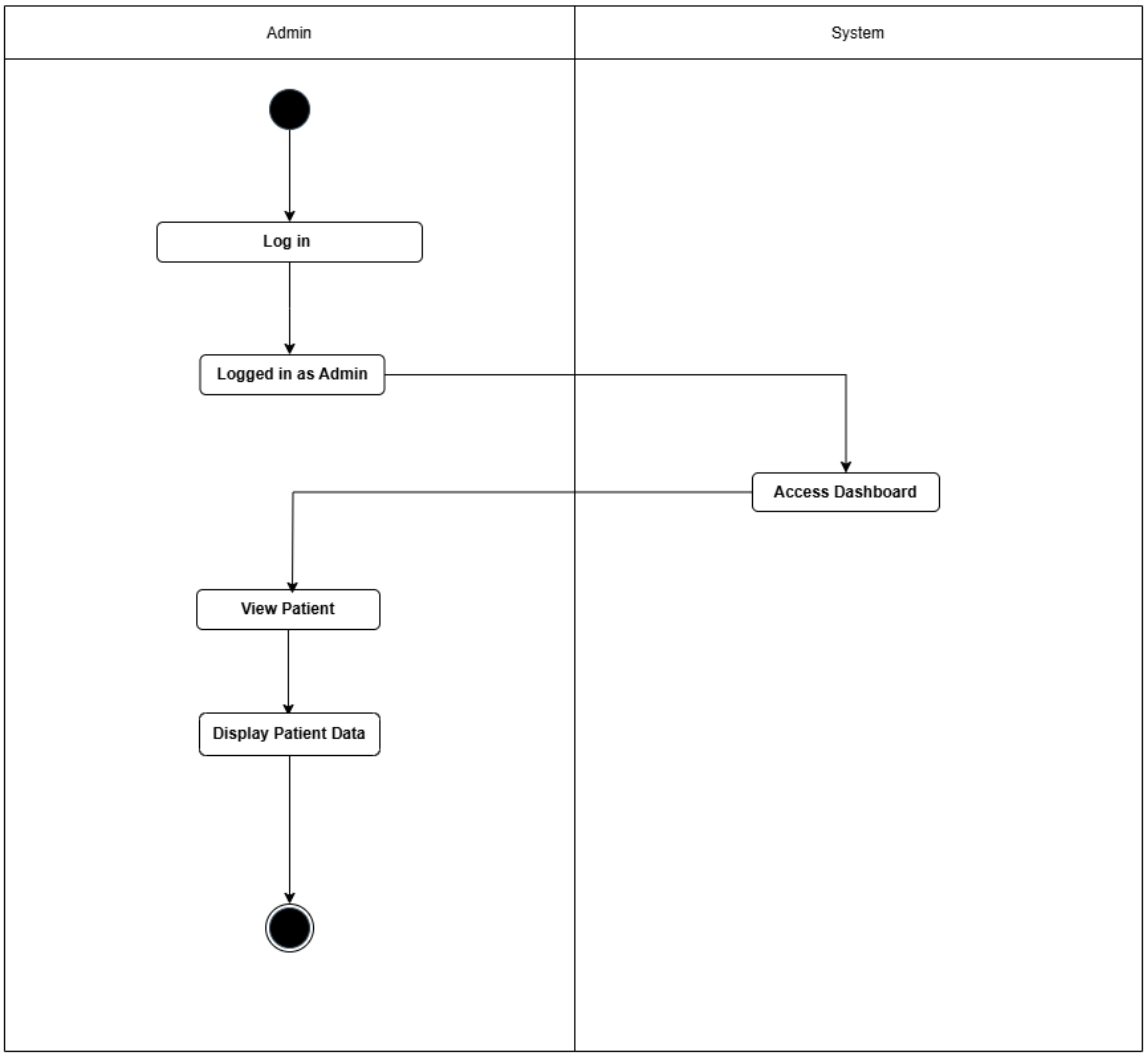

Figure 32: Swimlane Diagram for View Patients (Admin)

Use case 14: View Doctors (Admin)

Activity Diagram:

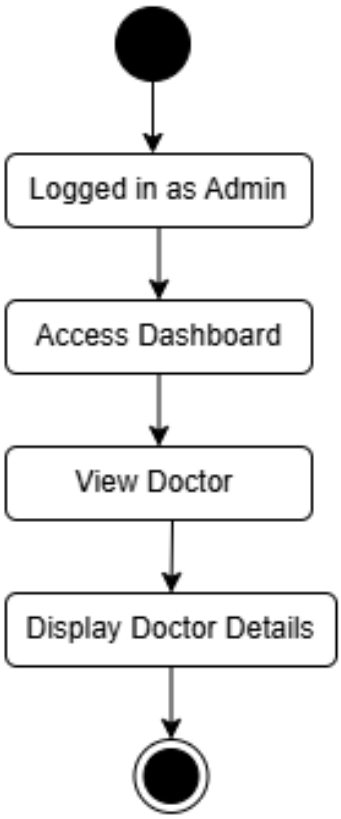

Figure 33: Activity Diagram for View Doctors (Admin)

Swimlane Diagram:

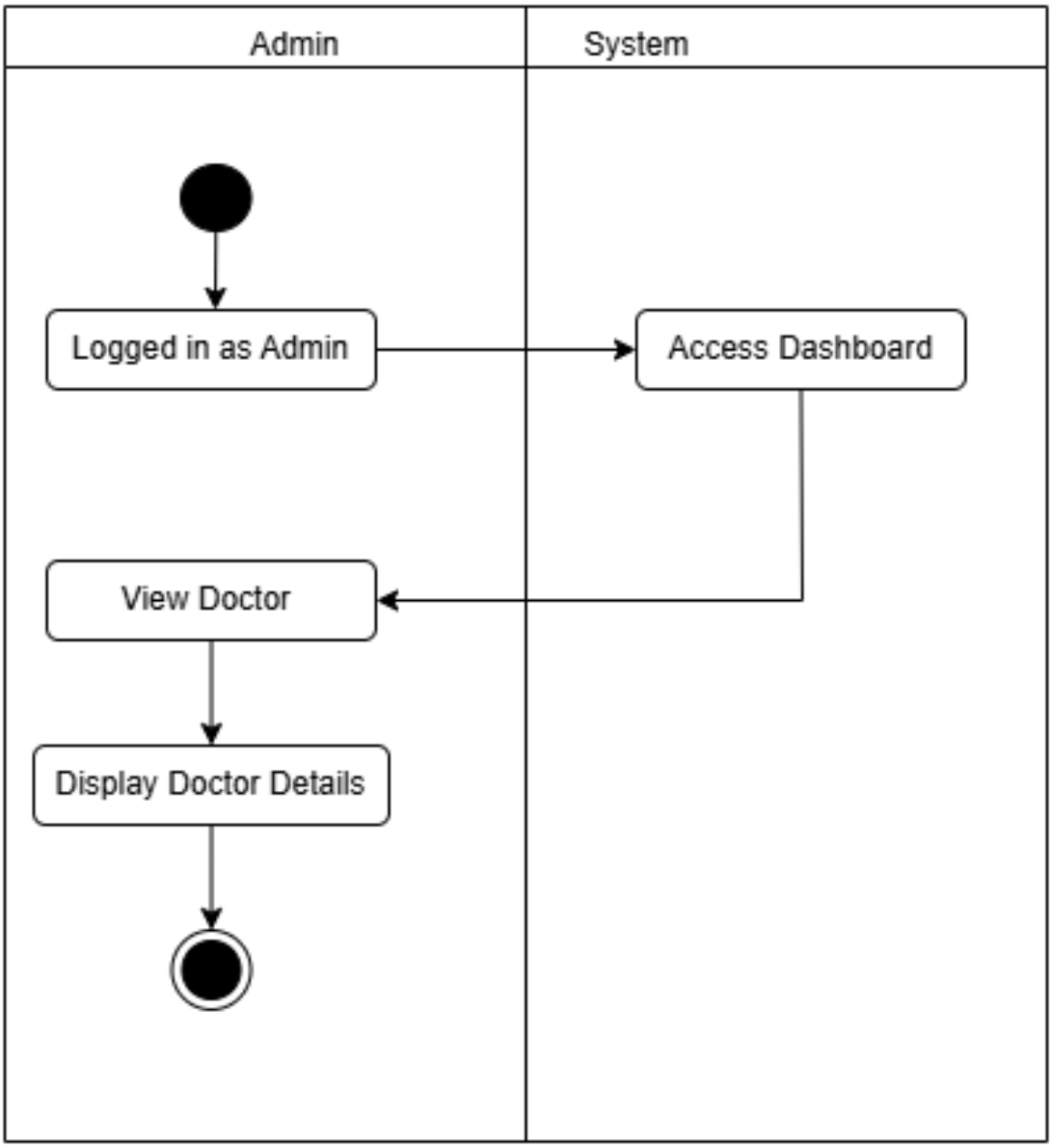

Figure 34: Swimlane Diagram for View Doctors (Admin)

Use case 15: Add Doctor

Activity Diagram:

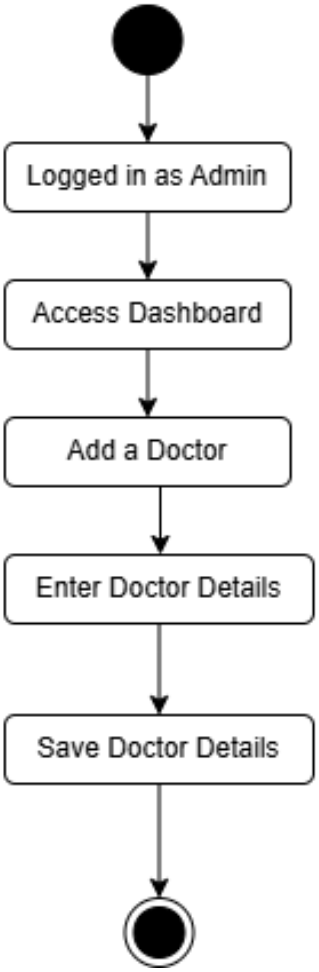

Figure 35: Activity Diagram for Add Doctor

Swimlane Diagram:

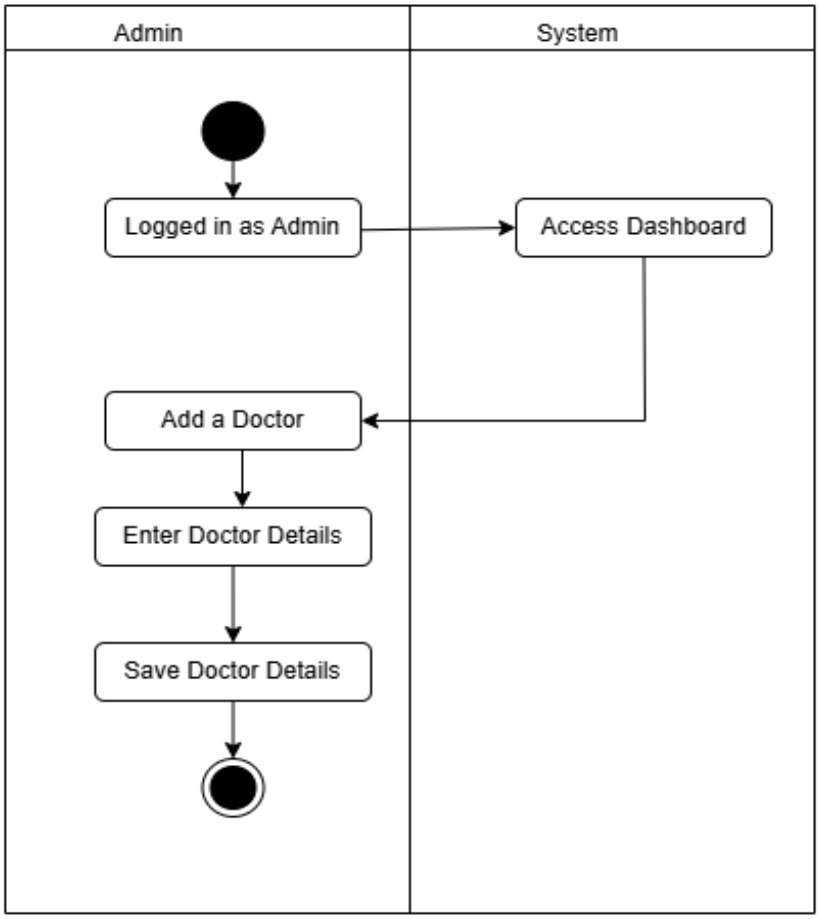

Figure 36: Swimlane Diagram for Add Doctor

Use case 16: Delete Doctor

Activity Diagram:

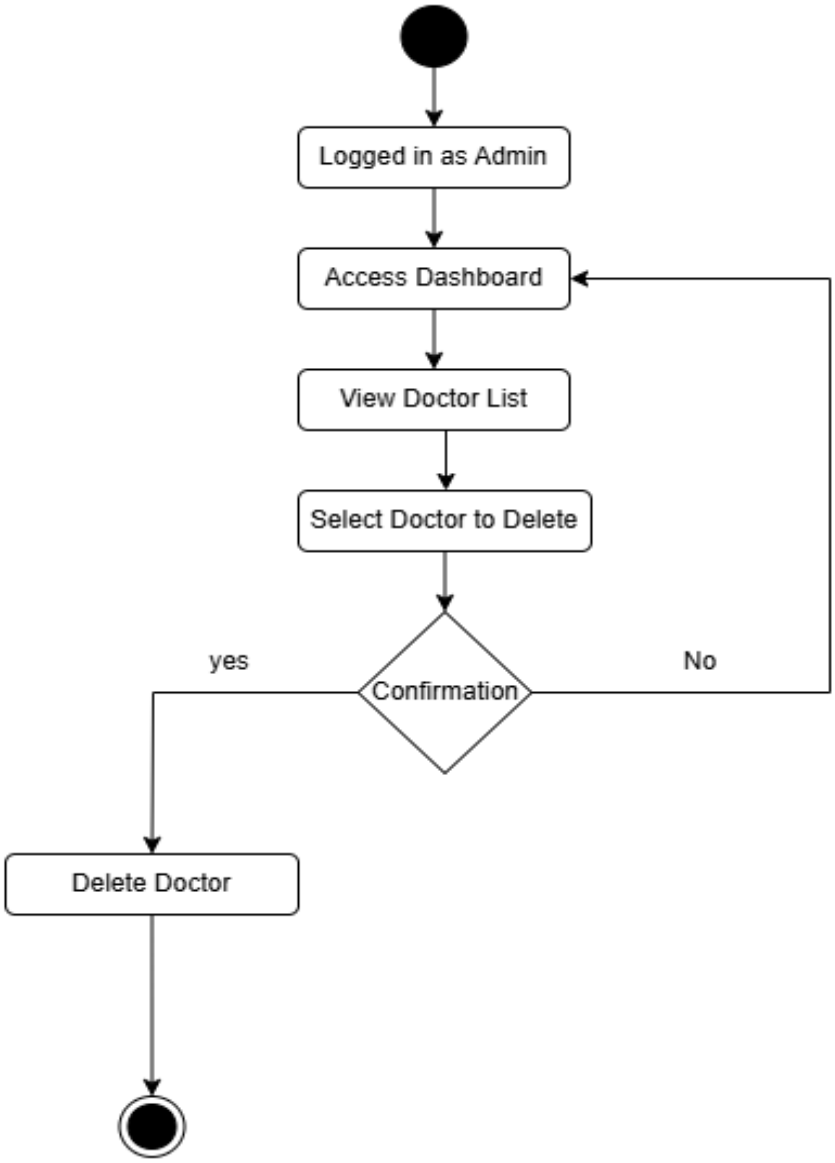

Figure 37: Activity Diagram for Delete Doctor

Swimlane Diagram:

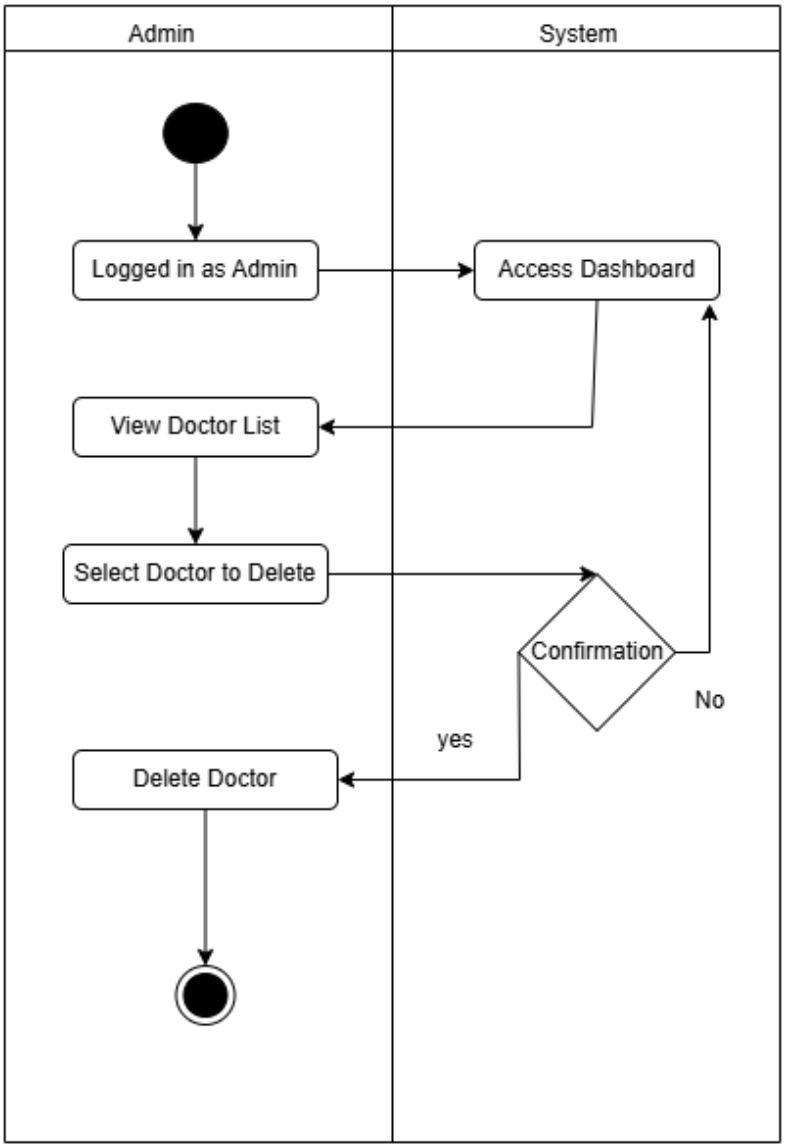

Figure 38: Swimlane Diagram for Delete Doctor

Use case 17: Remove Appointment

Activity Diagram:

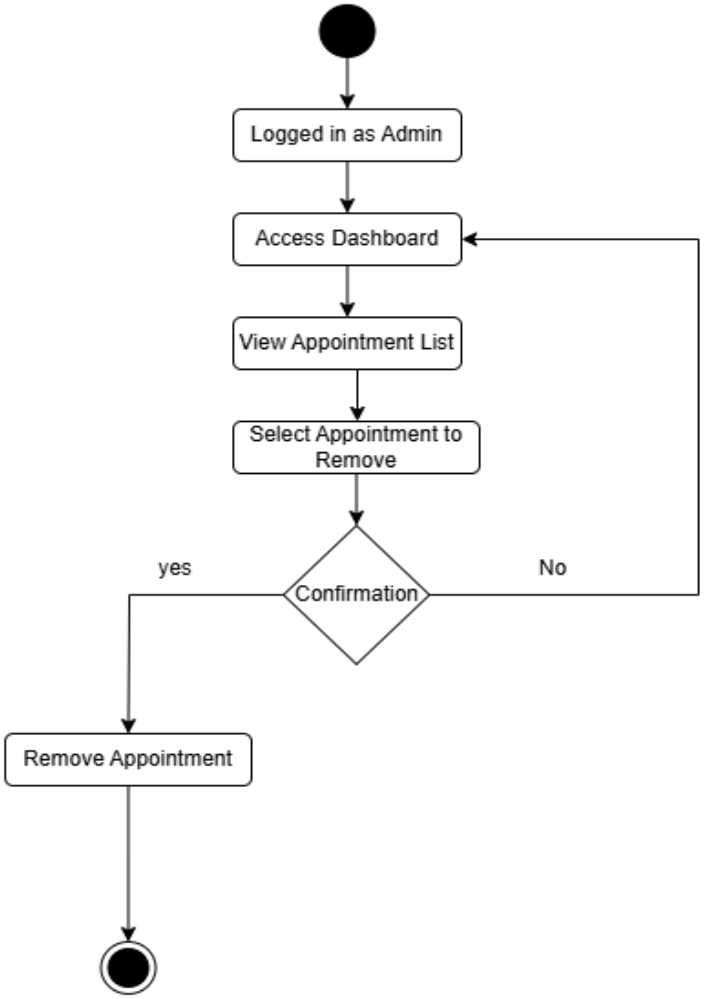

Figure 39: Activity Diagram for Remove Appointment

Swimlane Diagram:

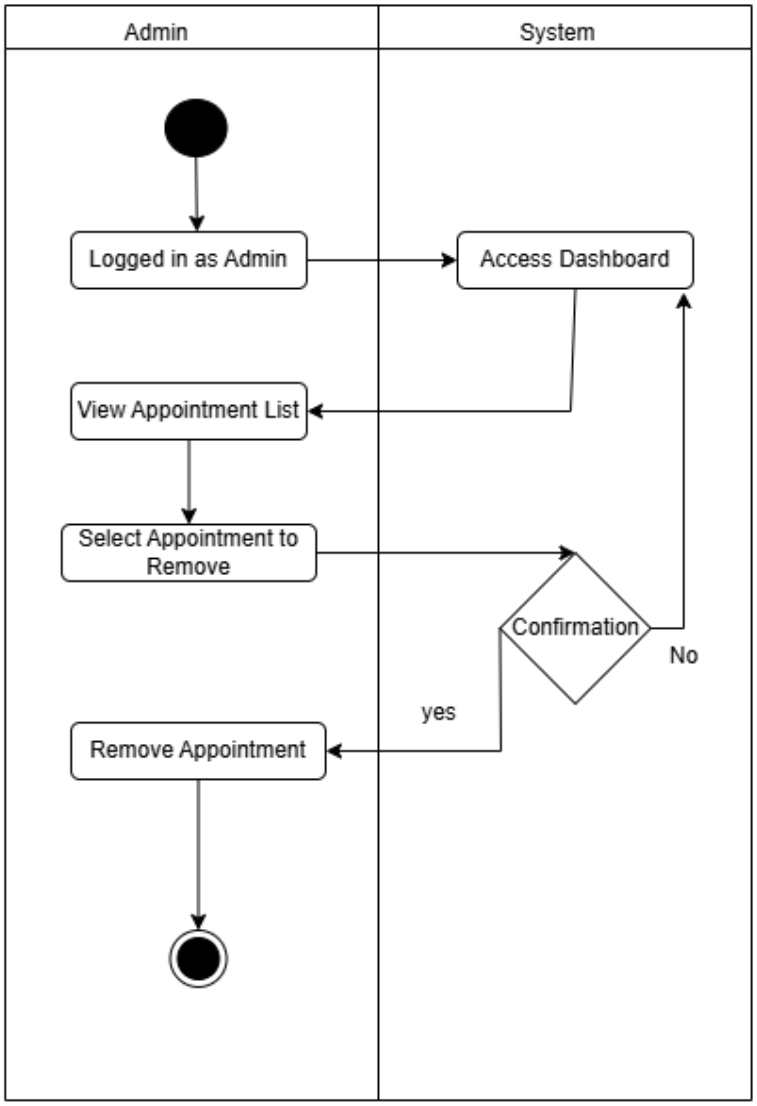

Figure 40: Swimlane Diagram for Remove Appointment

Use case 18: Generate Bill

Activity Diagram:

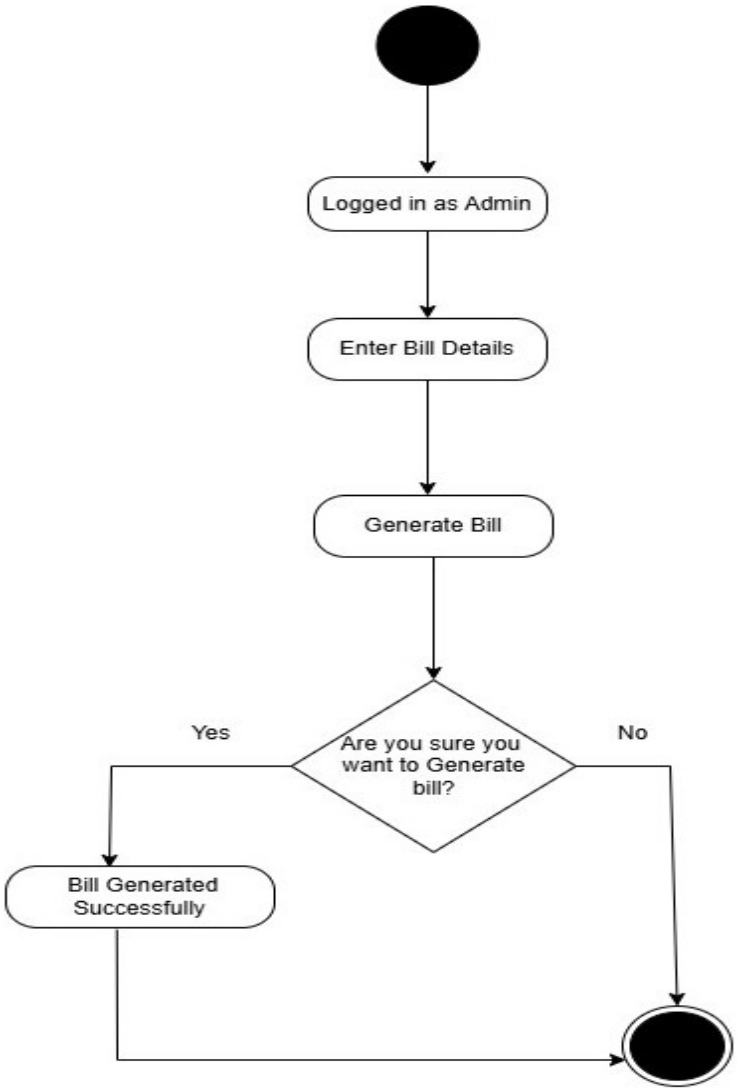

Figure 41: Activity Diagram for Generate Bill

Swimlane Diagram:

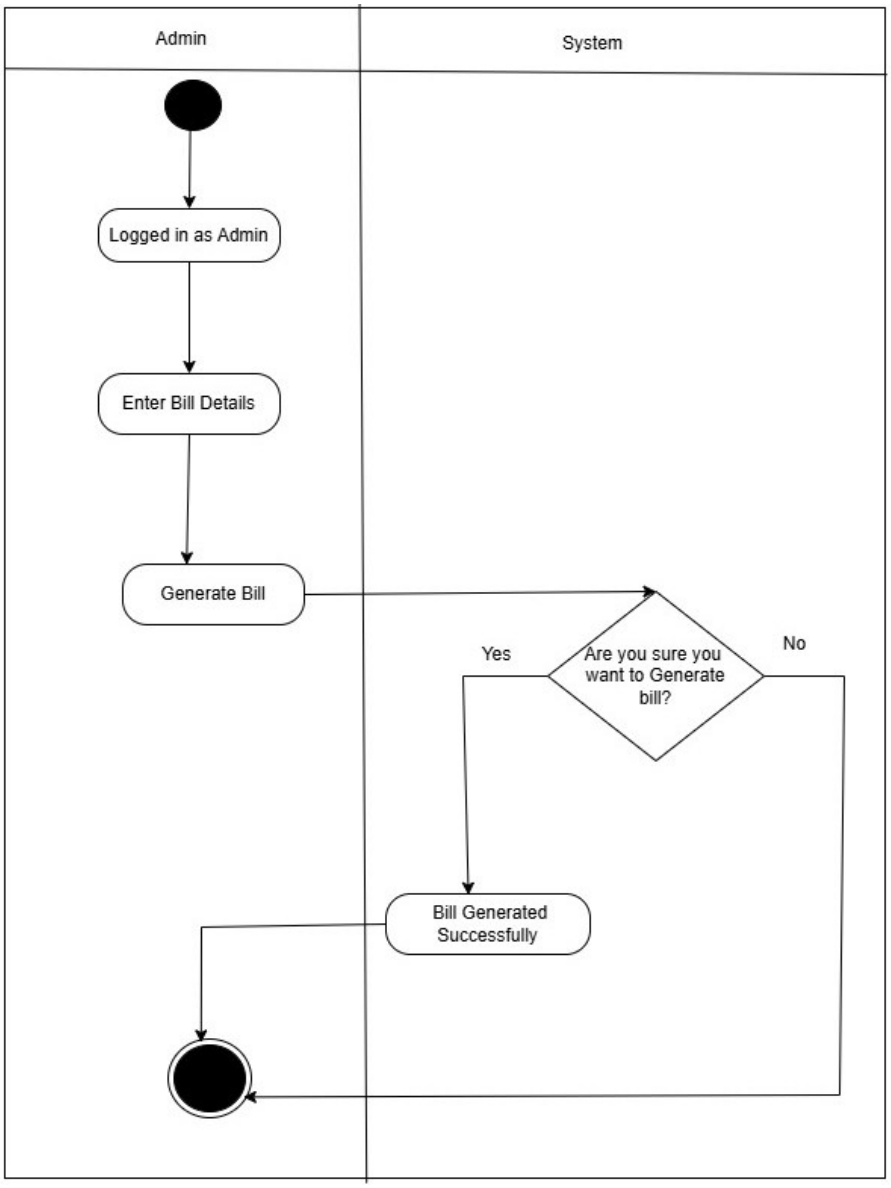

Figure 42: Swimlane Diagram for Generate Bill

Use case 19: View Bills

Activity Diagram:

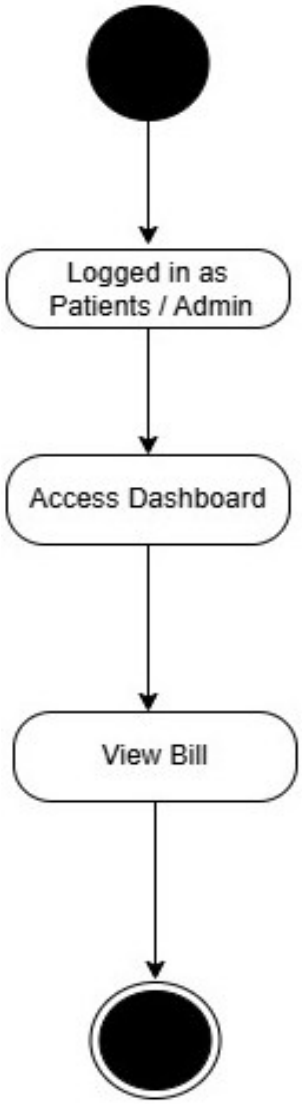

Figure 43: Activity Diagram for View Bills

Swimlane Diagram:

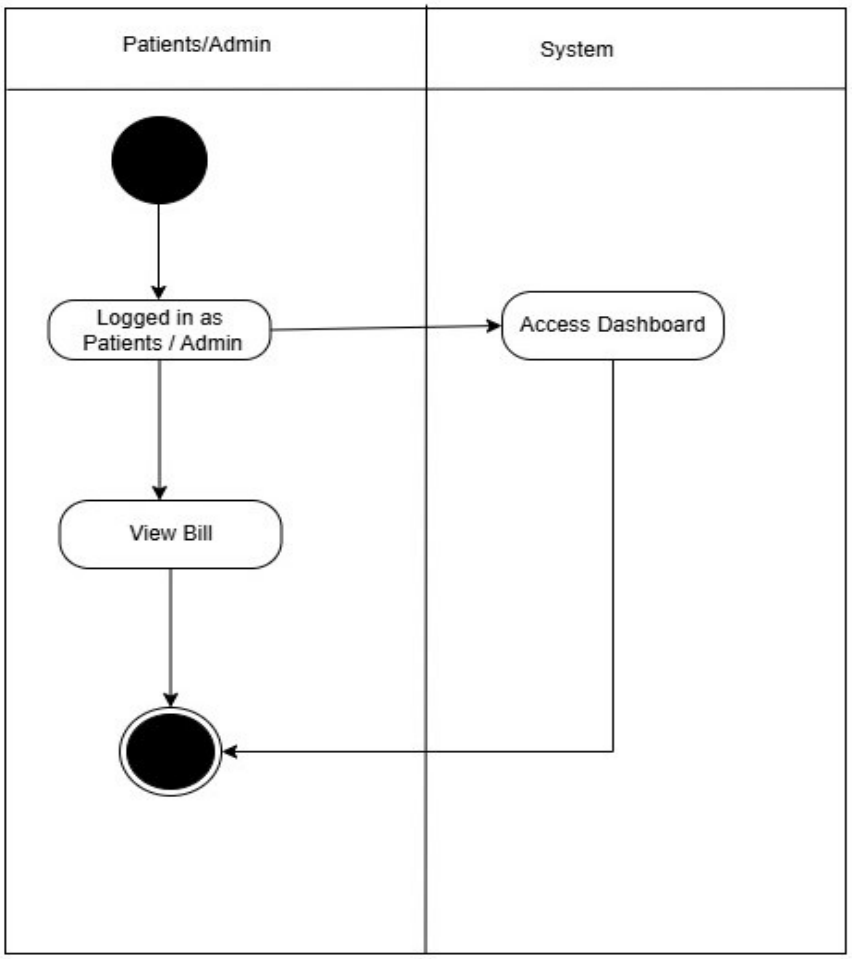

Figure 44: Swimlane Diagram for View Bills

Use case 21: Pay Bill

Activity Diagram:

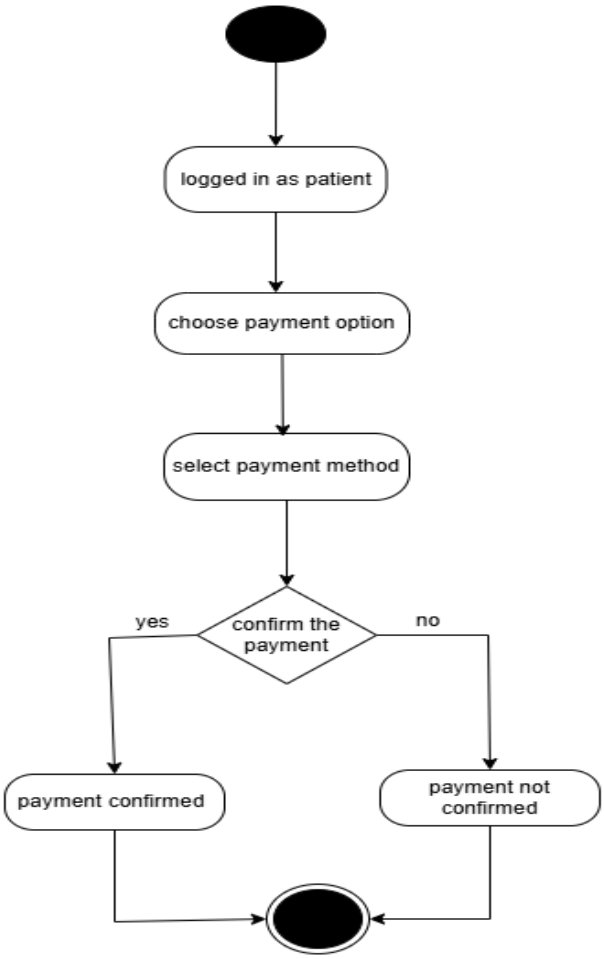

Figure 45: Activity Diagram for Pay Bill

Swimlane Diagram:

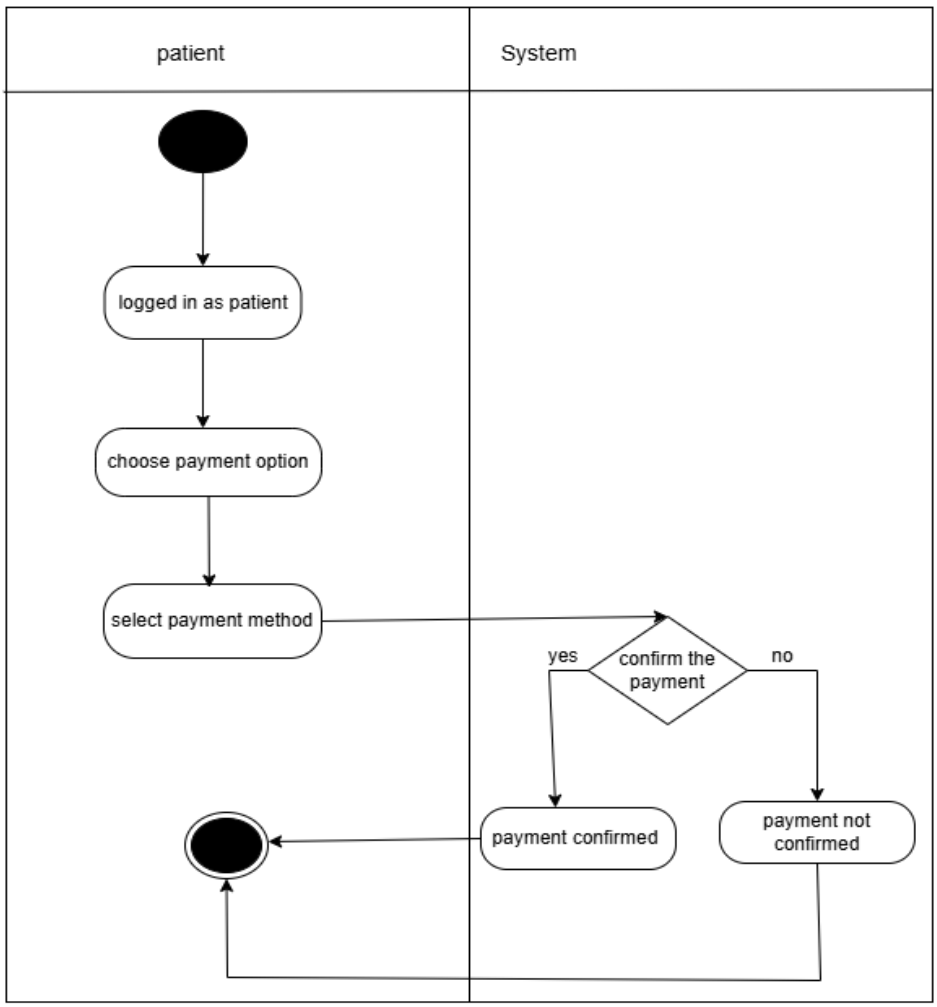

Figure 46: Swimlane Diagram for Pay Bill

Use case 21: Schedule Appointment

Activity Diagram:

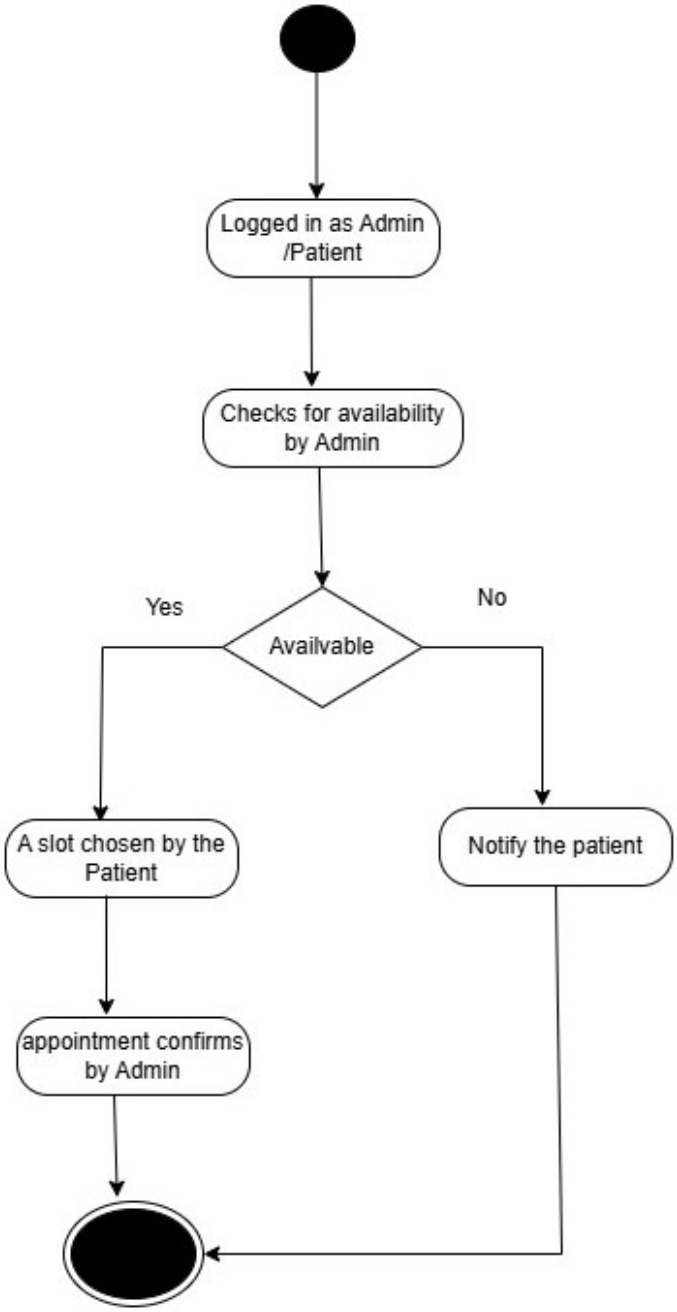

Figure 47: Activity Diagram for Schedule Appointment

Swimlane Diagram:

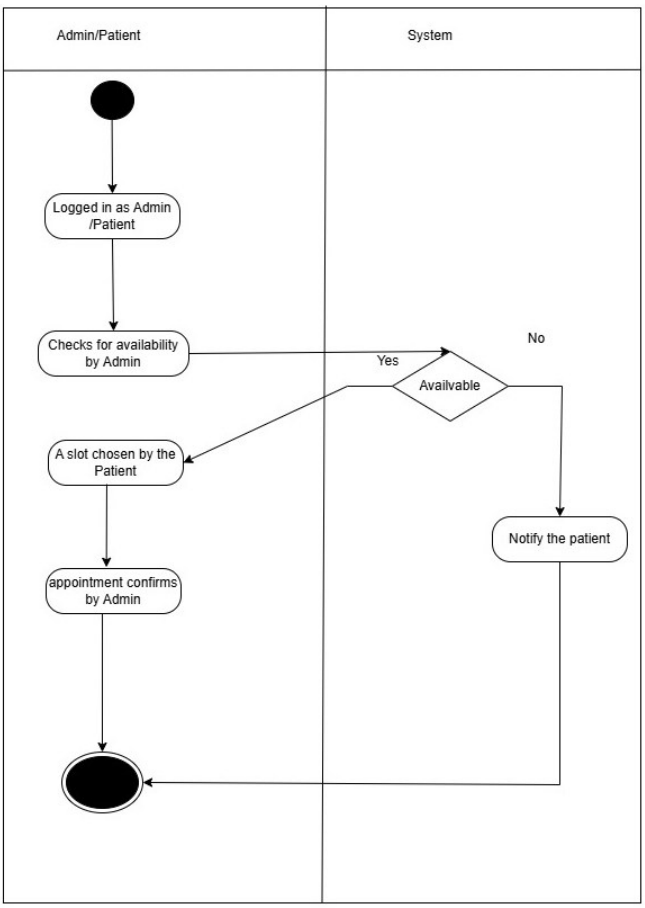

Figure 48: Swimlane Diagram for Schedule Appointment

Chapter 5: Class based modeling

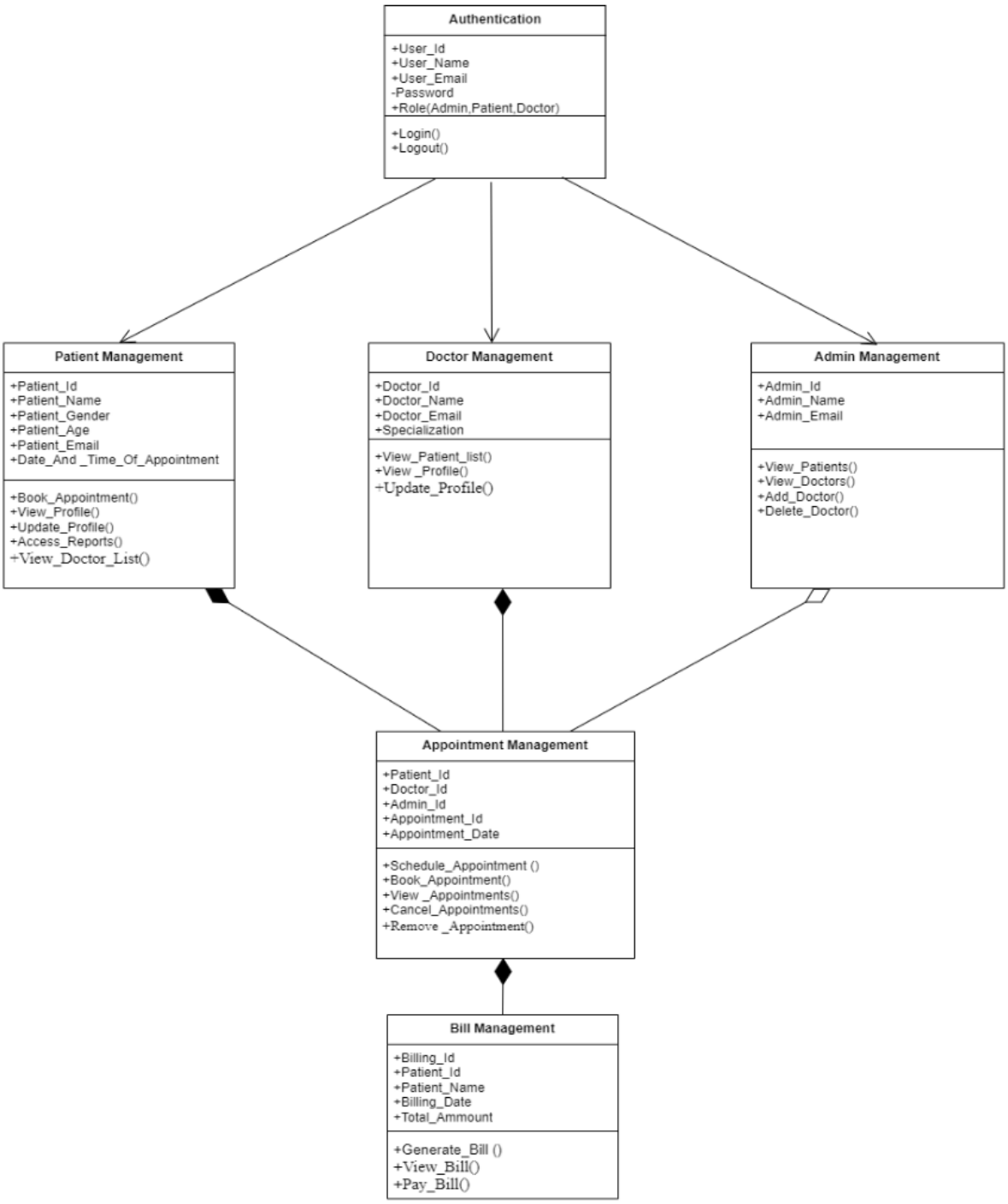

Figure 49: UML Diagram

Chapter 6: Flow-Oriented Model

6.1 Data Flow Diagram (DFD)

Level 0 of data flow diagram

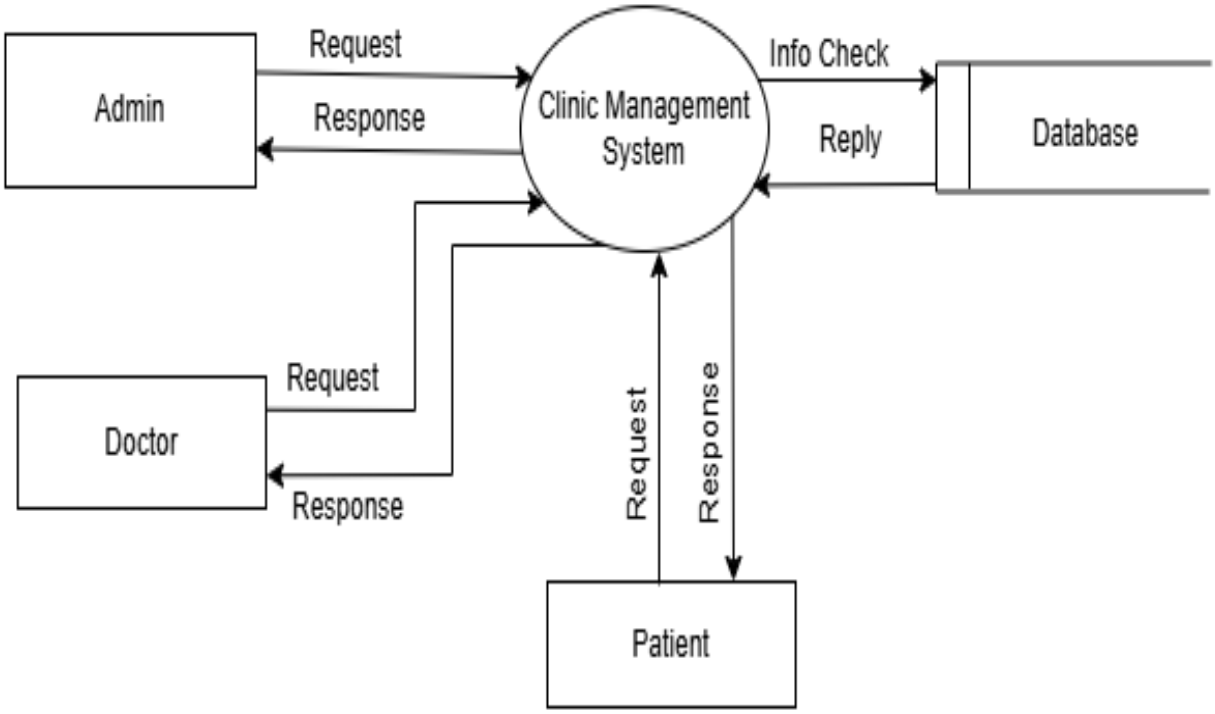

Figure 50: Level 0 for Clinic Management System

Level 1 of data flow diagram

Level 1(Admin):

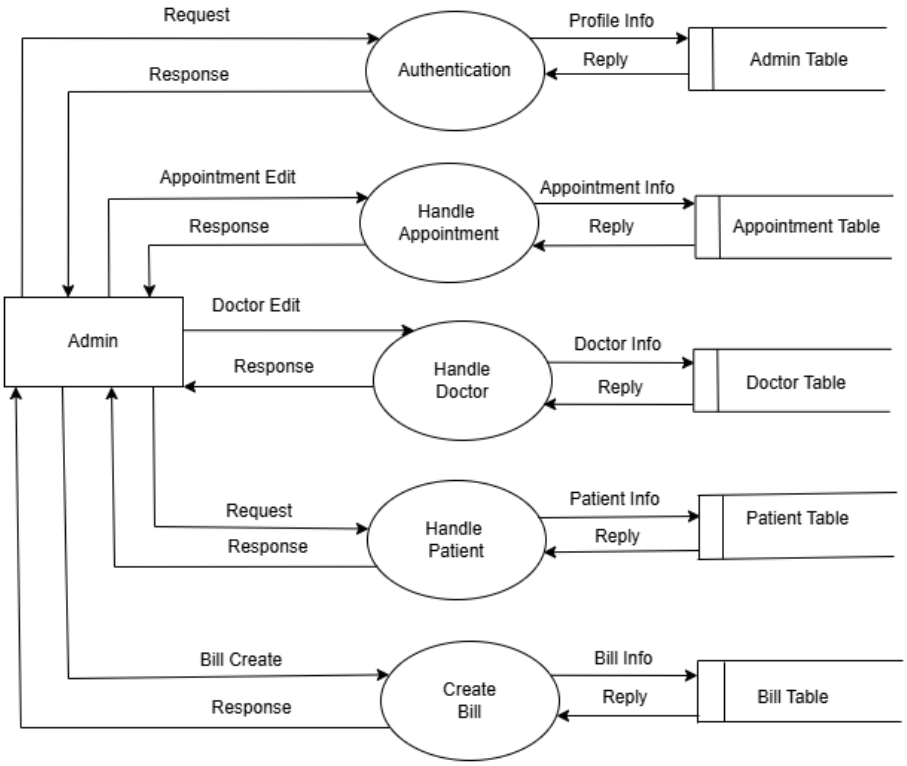

Figure 51: Level 1.1 for Clinic Management System

Level 1(Doctor):

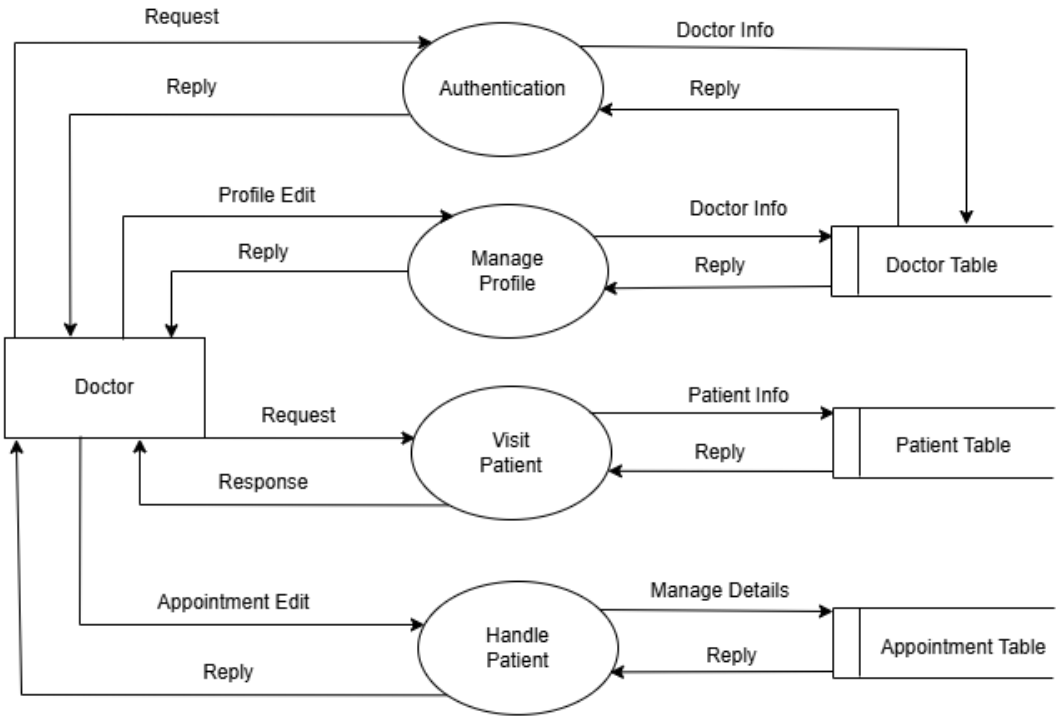

Figure 52: Level 1.2 for Clinic Management System

Level 1(Patient):

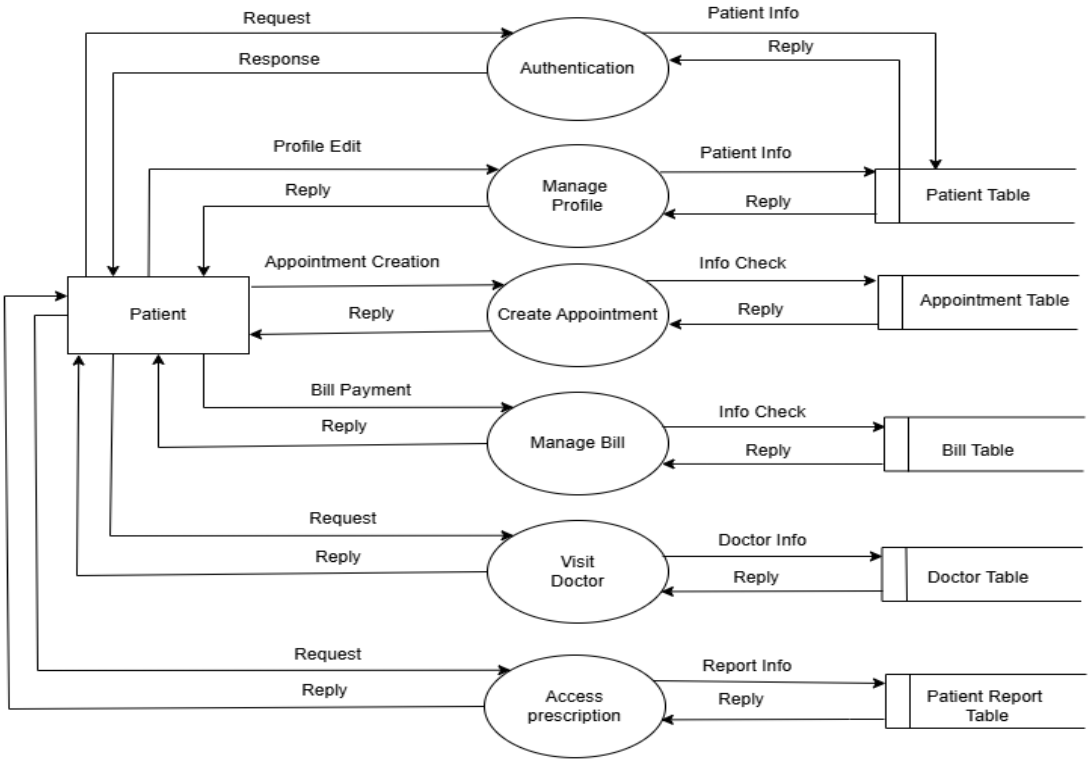

Figure 53: Level 1.3 for Clinic Management System

Level 2 of data flow diagram

Level 2(Authentication)

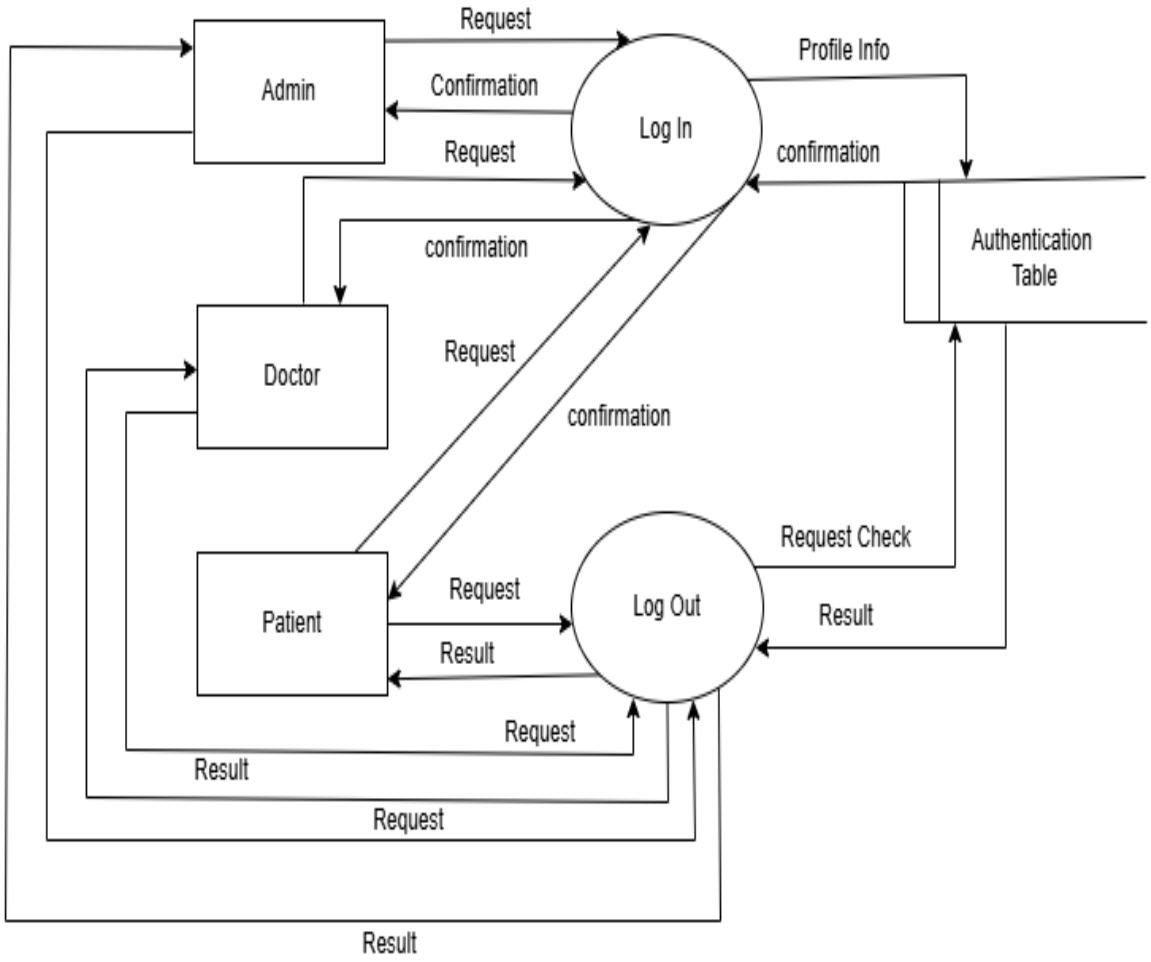

Figure 54: Level 2.1 for Clinic Management System

Level 2(Admin)

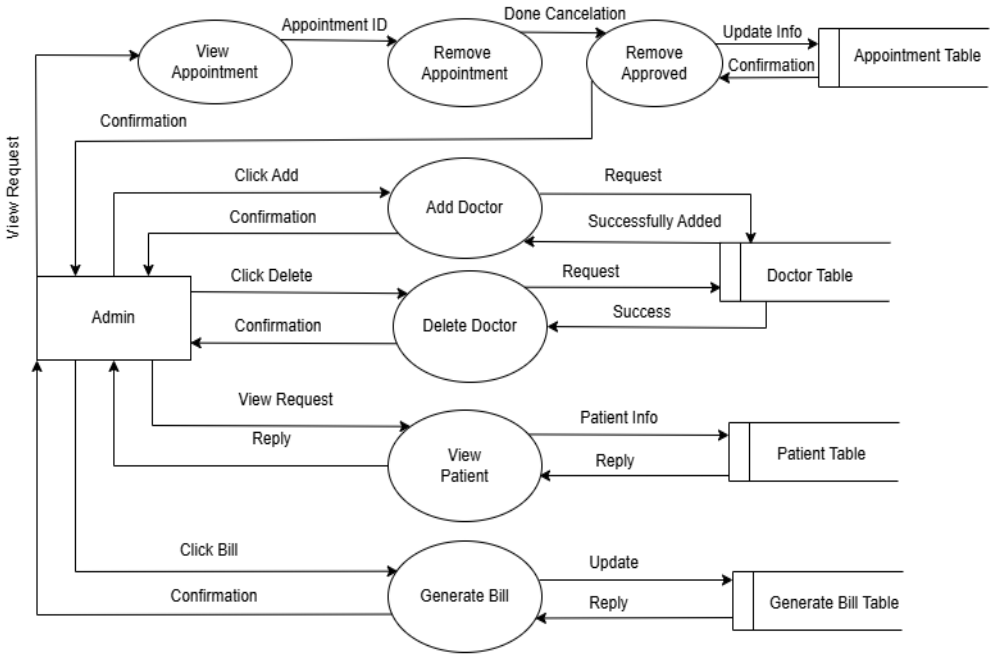

Figure 55: Level 2.2 for Clinic Management System

Level 2(Doctor)

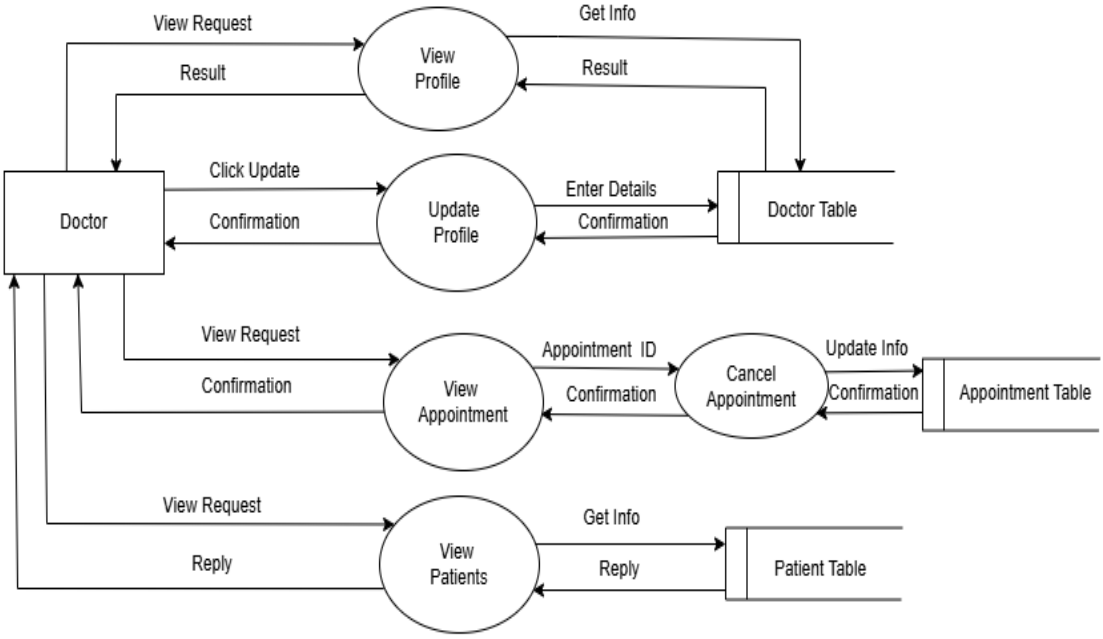

Figure 56: Level 2.3 for Clinic Management System

Level 2(Patient)

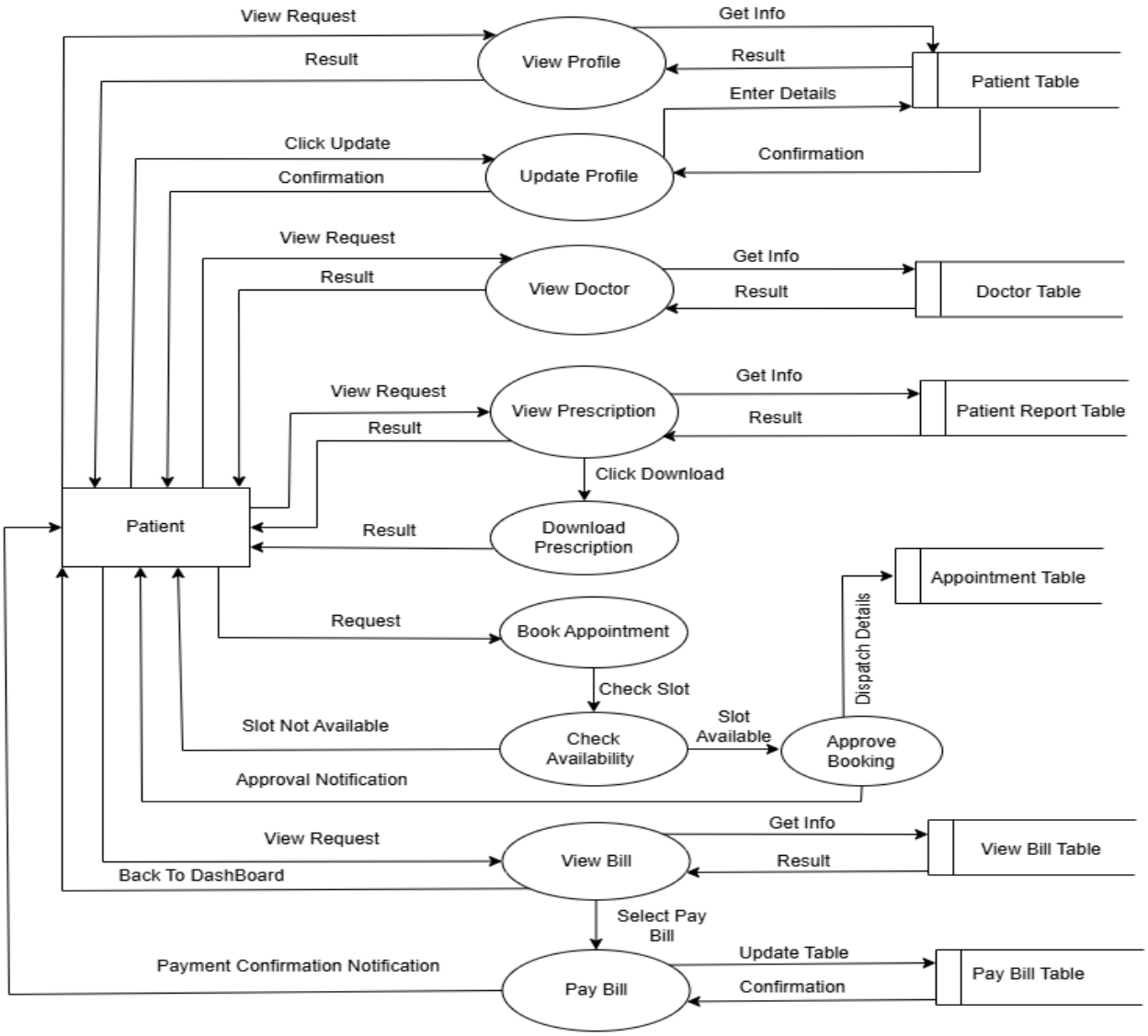

Figure 57: Level 2.4 for Clinic Management System

Chapter 7: Behavioral Model

7.1 State Machine Diagrams

Admin State Diagram:

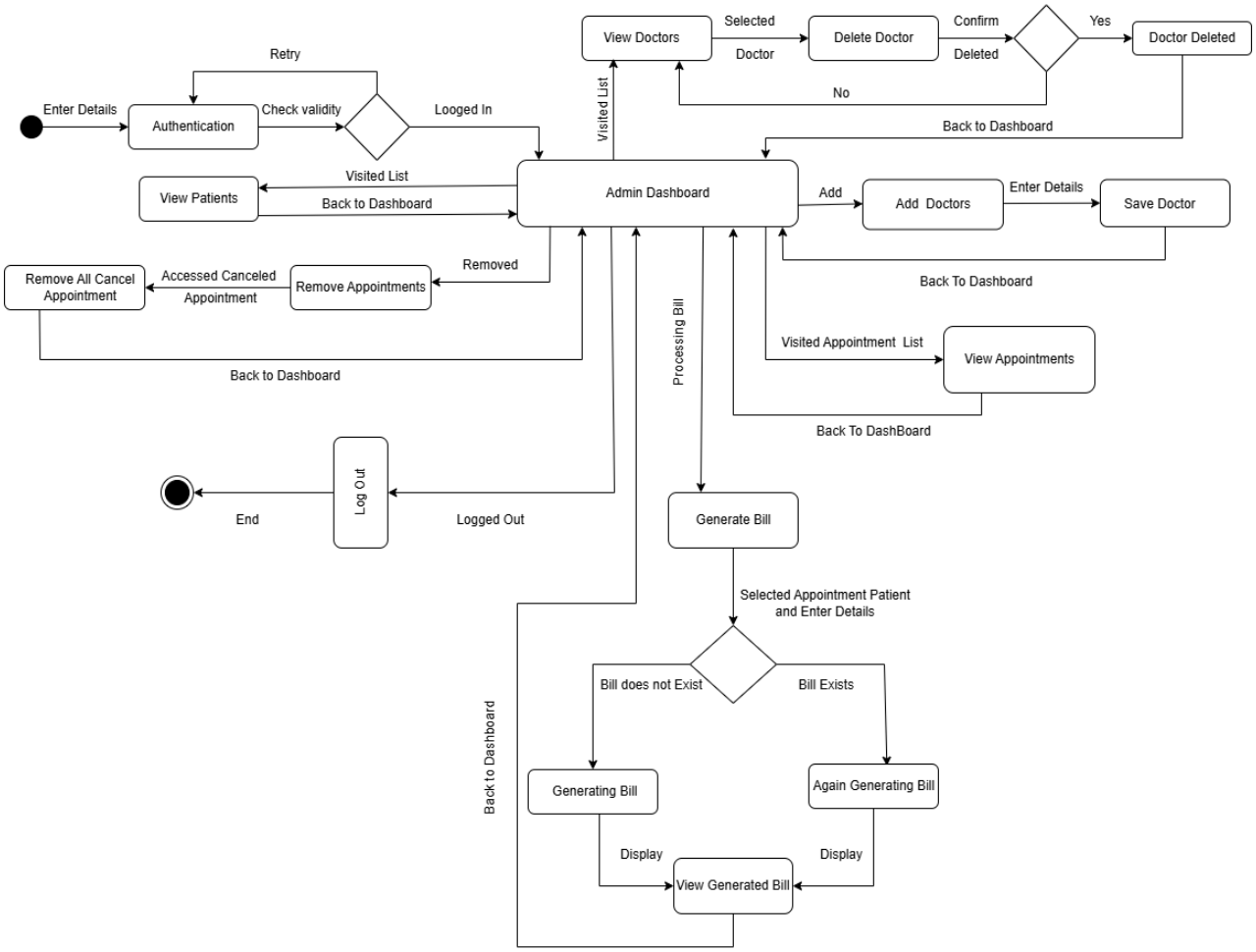

Figure 58: State Diagram (Admin)

Doctor State Diagram:

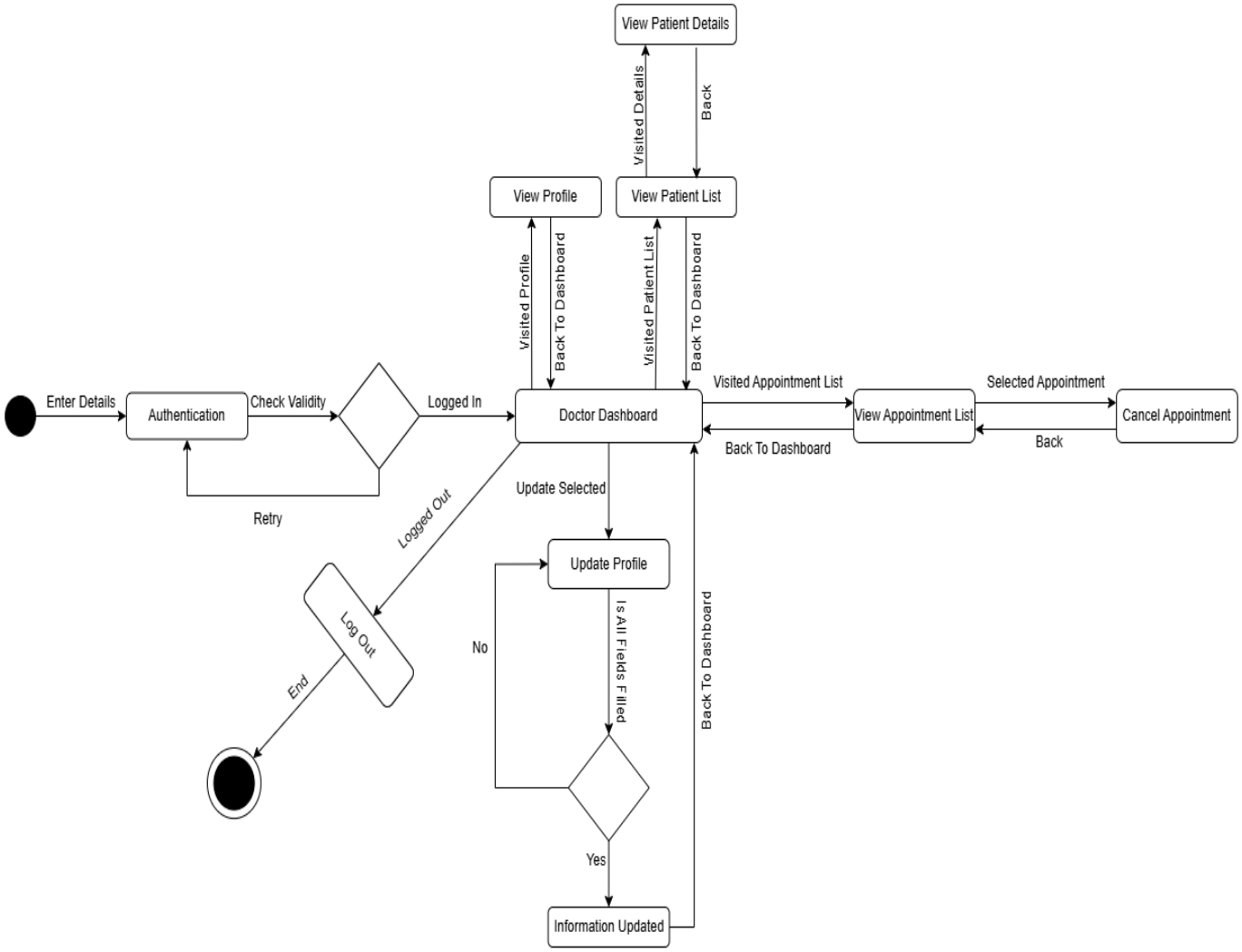

Figure 59: State Diagram (Doctor)

Patient State Diagram:

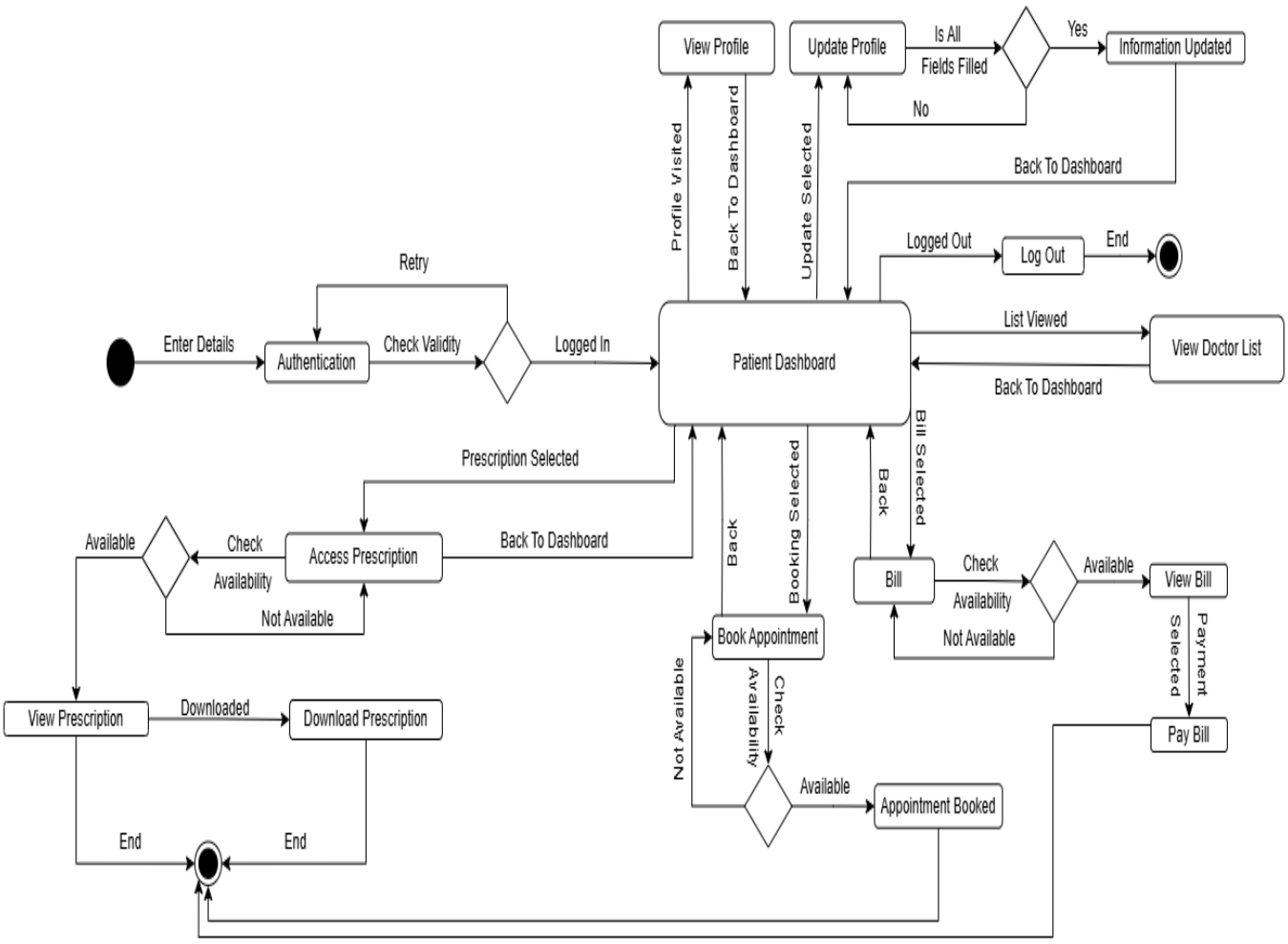

Figure 60: State Diagram (Patient)

7.2 Sequence Diagrams of Clinic Management System

Sequence Diagram: Log In

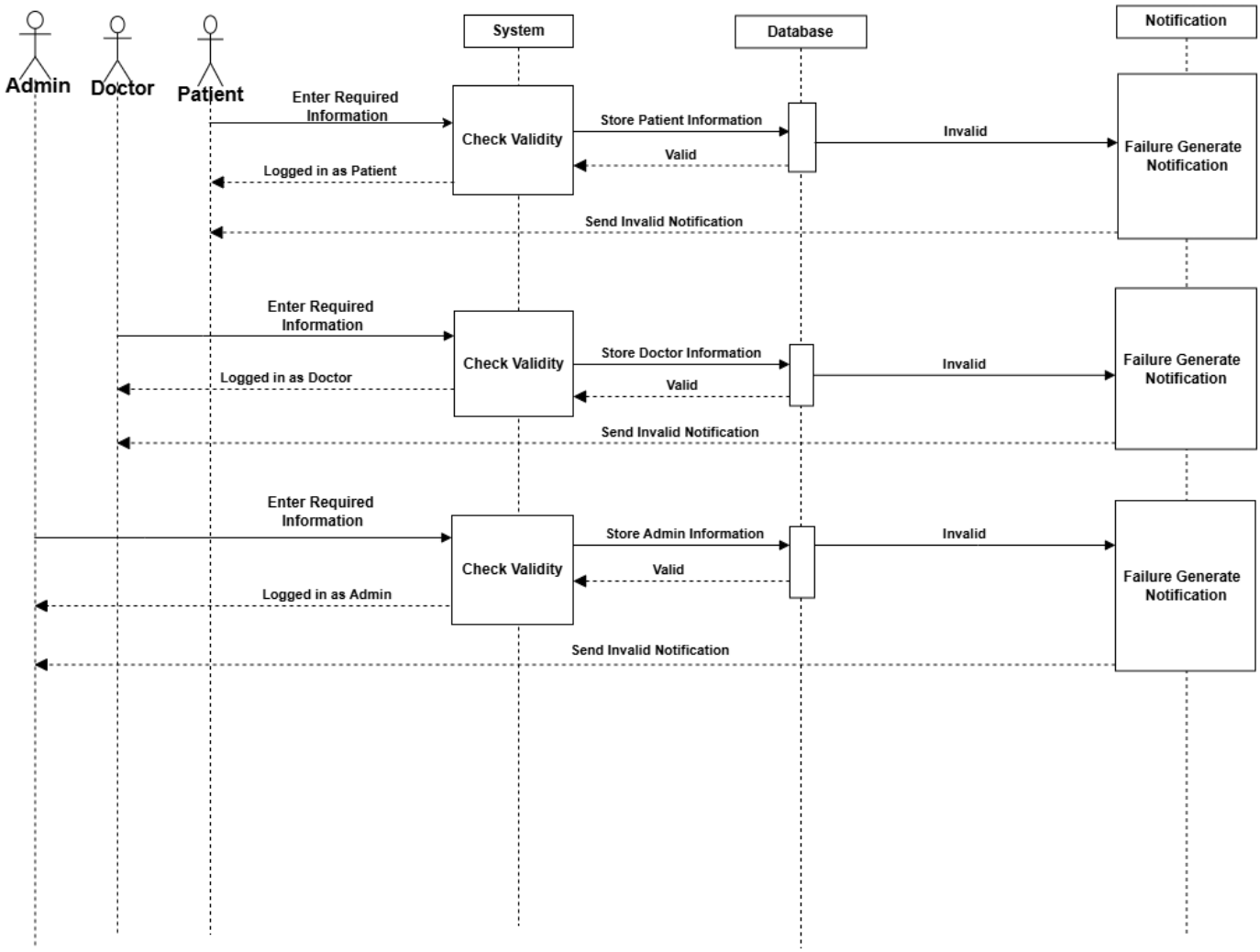

Figure 61: Sequence Diagram (Log In)

Sequence Diagram: Log Out

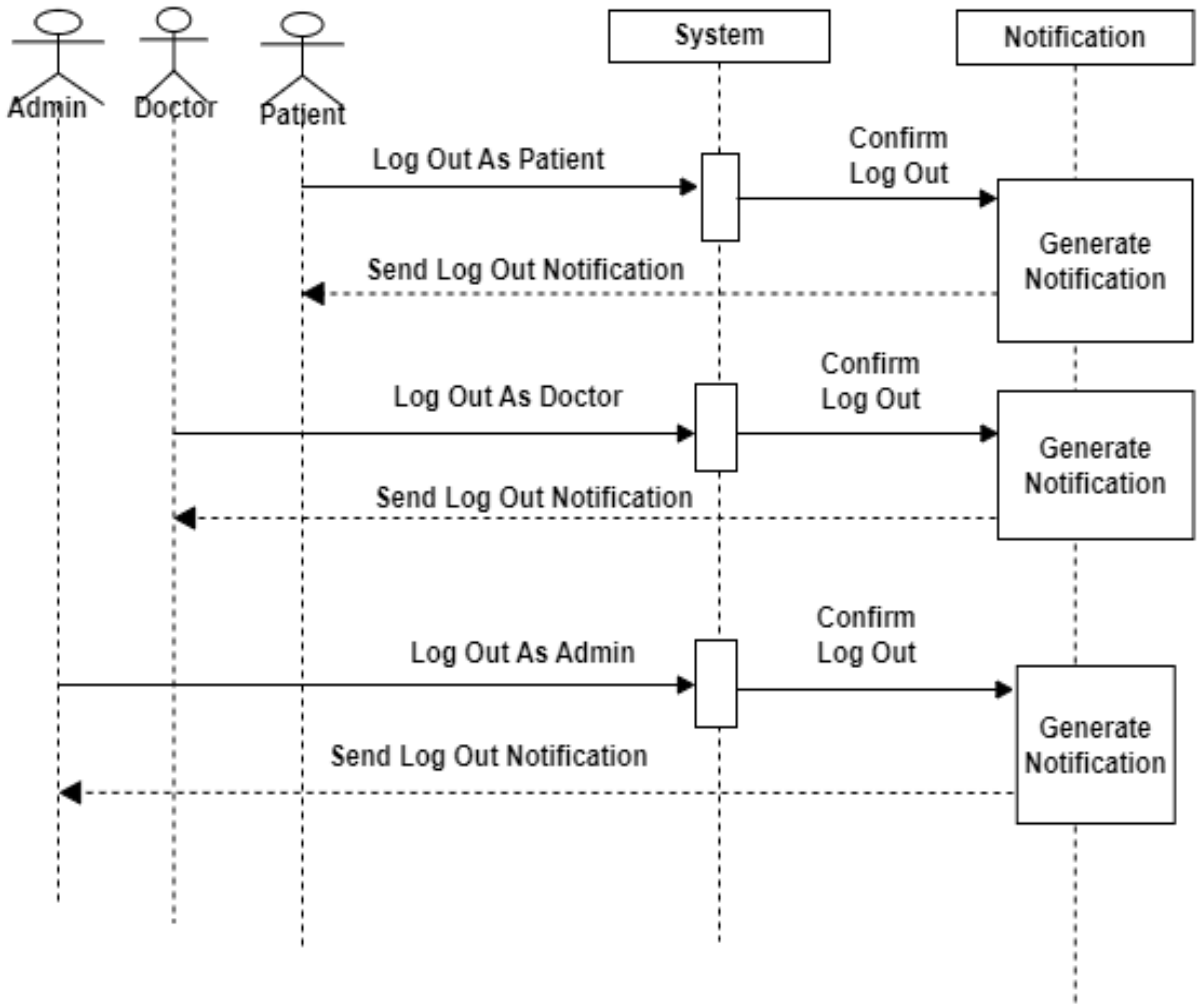

Figure 62: Sequence Diagram (Log Out)

Sequence Diagram: Patient Management

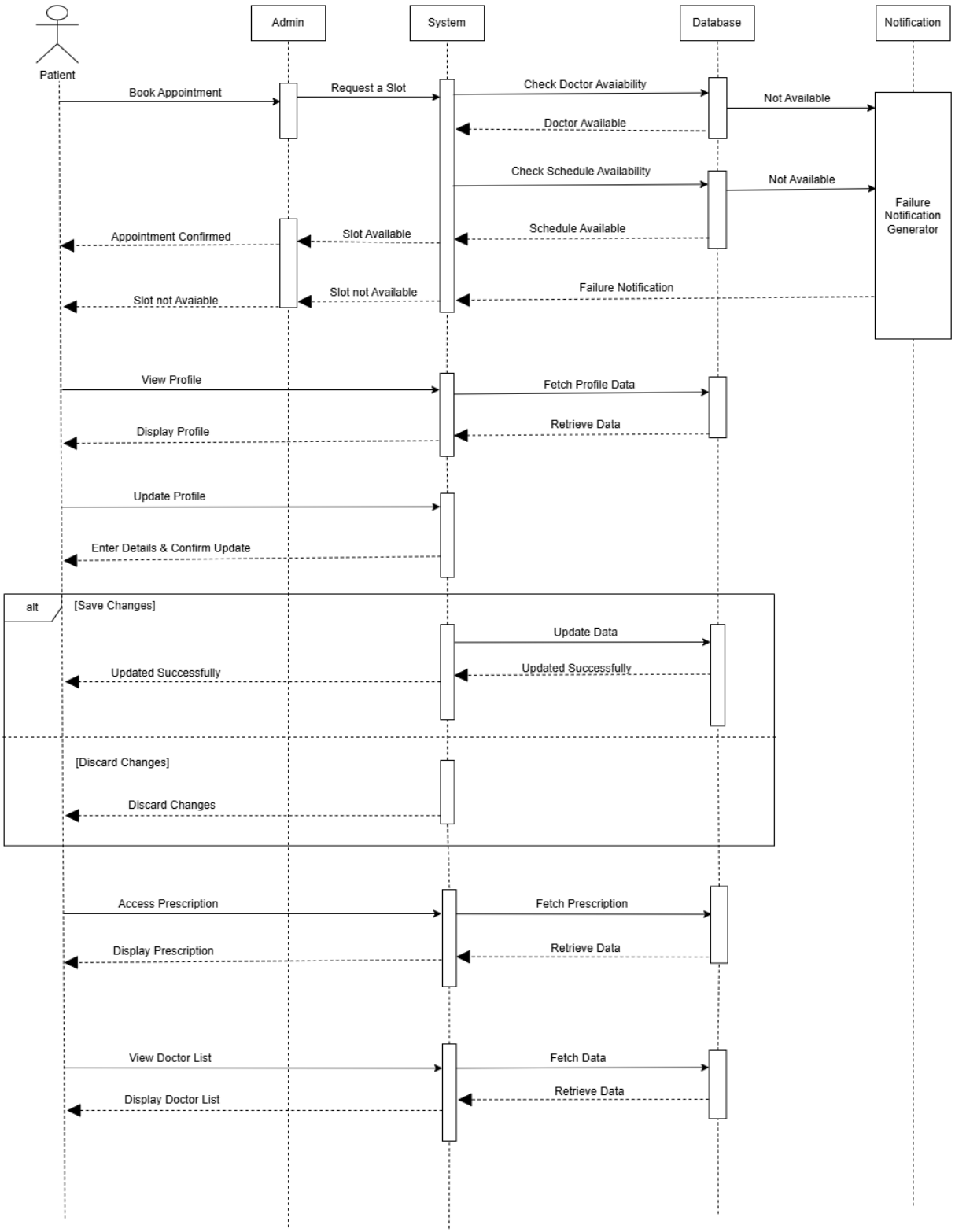

Figure 63: Sequence Diagram (Patient Management)

Sequence Diagram: Doctor Management

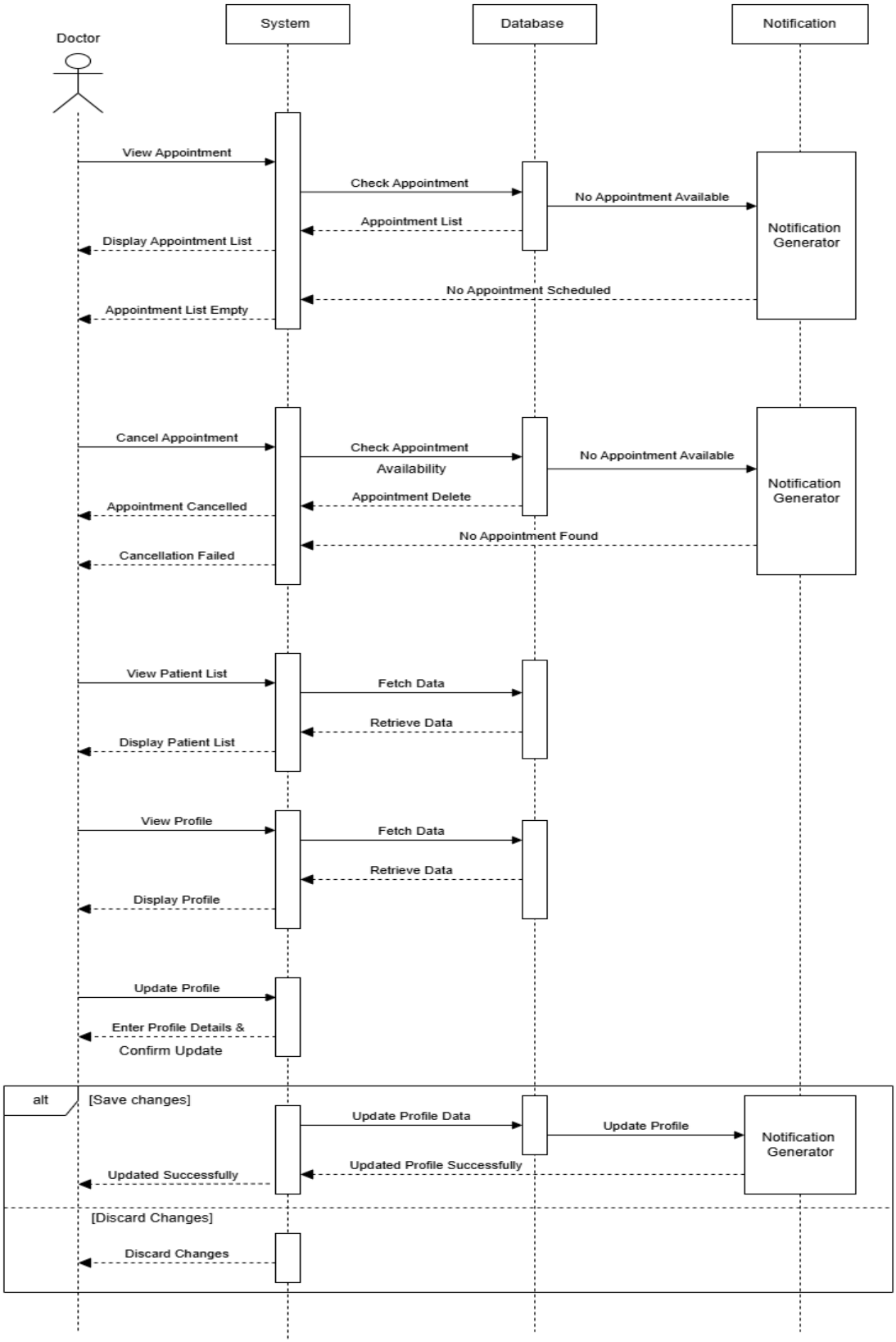

Figure 64: Sequence Diagram (Doctor Management)

Sequence Diagram: Admin Management

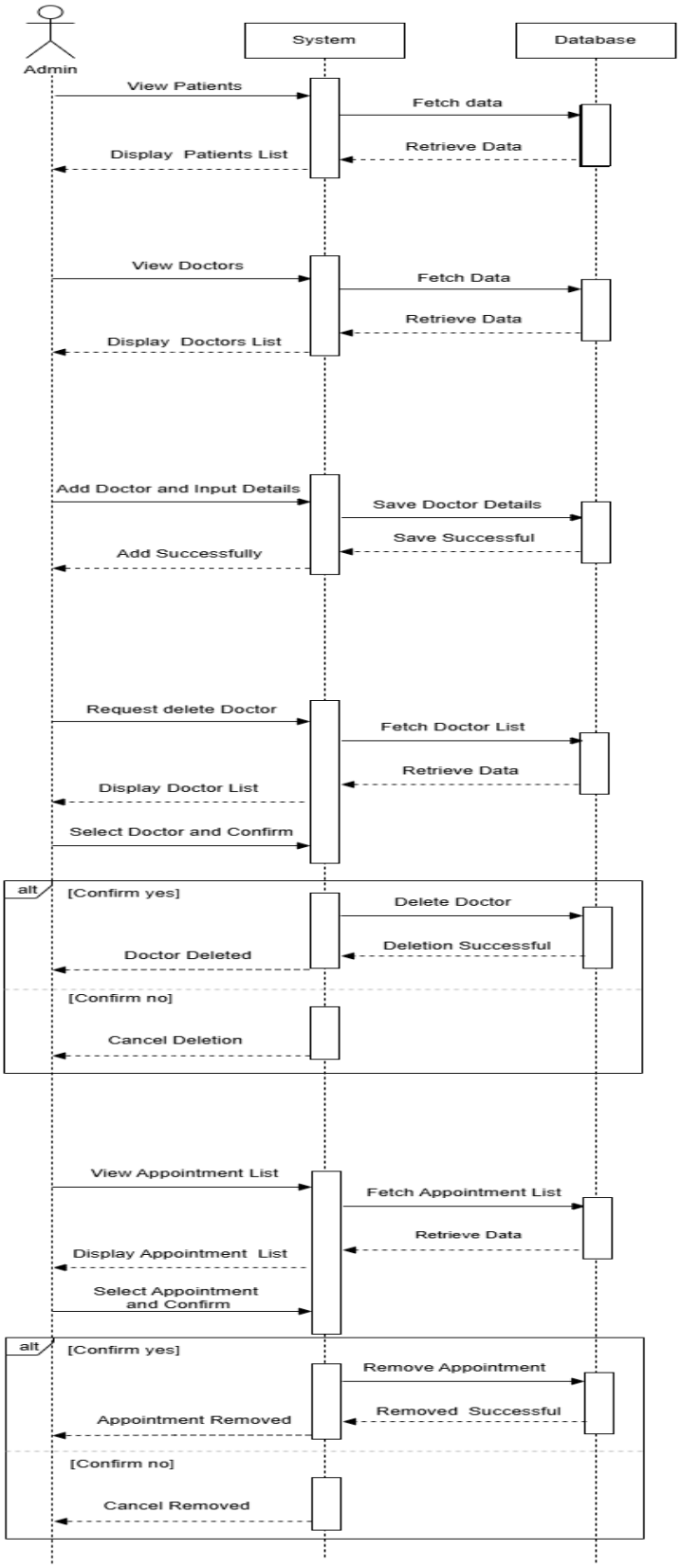

Figure 65: Sequence Diagram (Admin Management)

Sequence Diagram: Bill Management

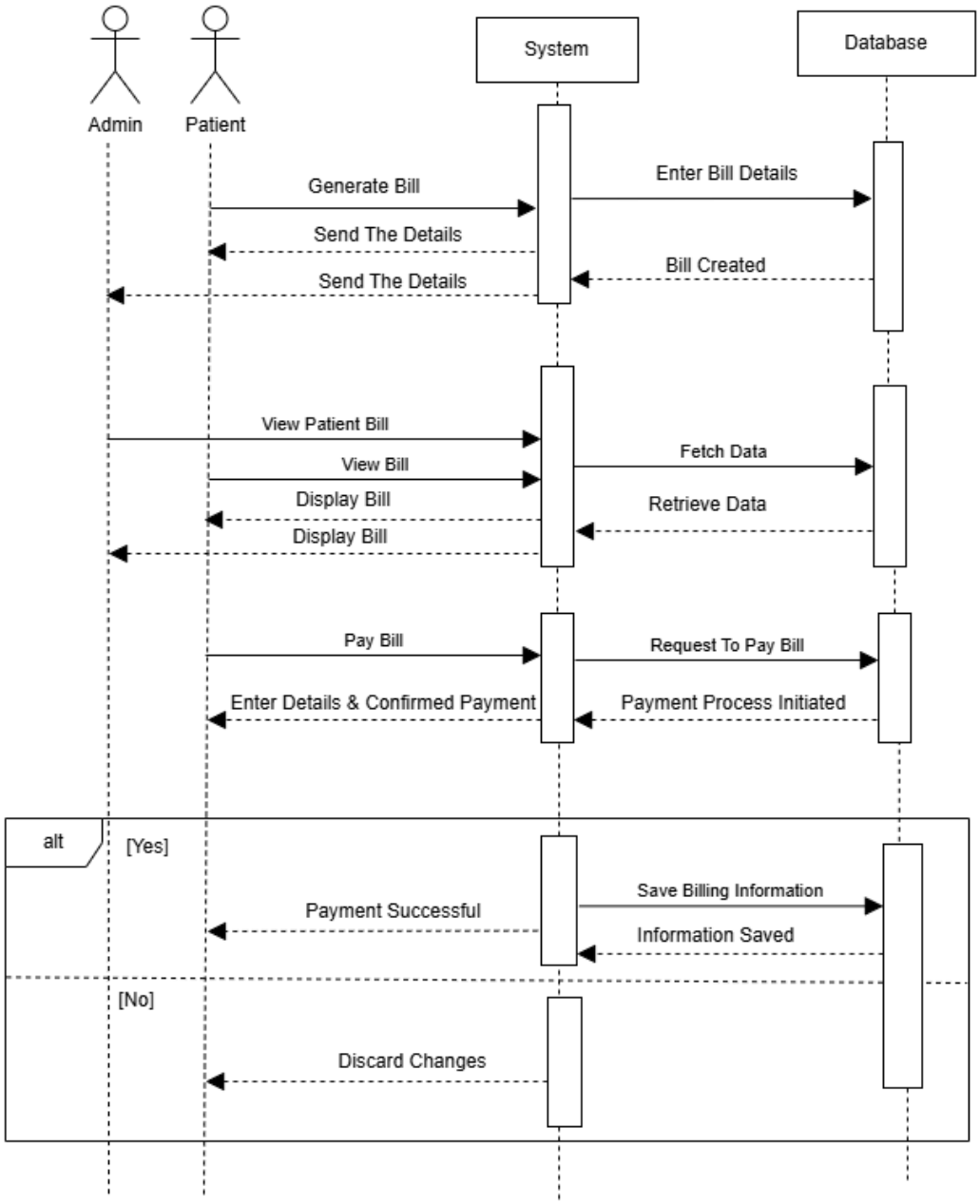

Figure 66: Sequence Diagram (Billing Management)

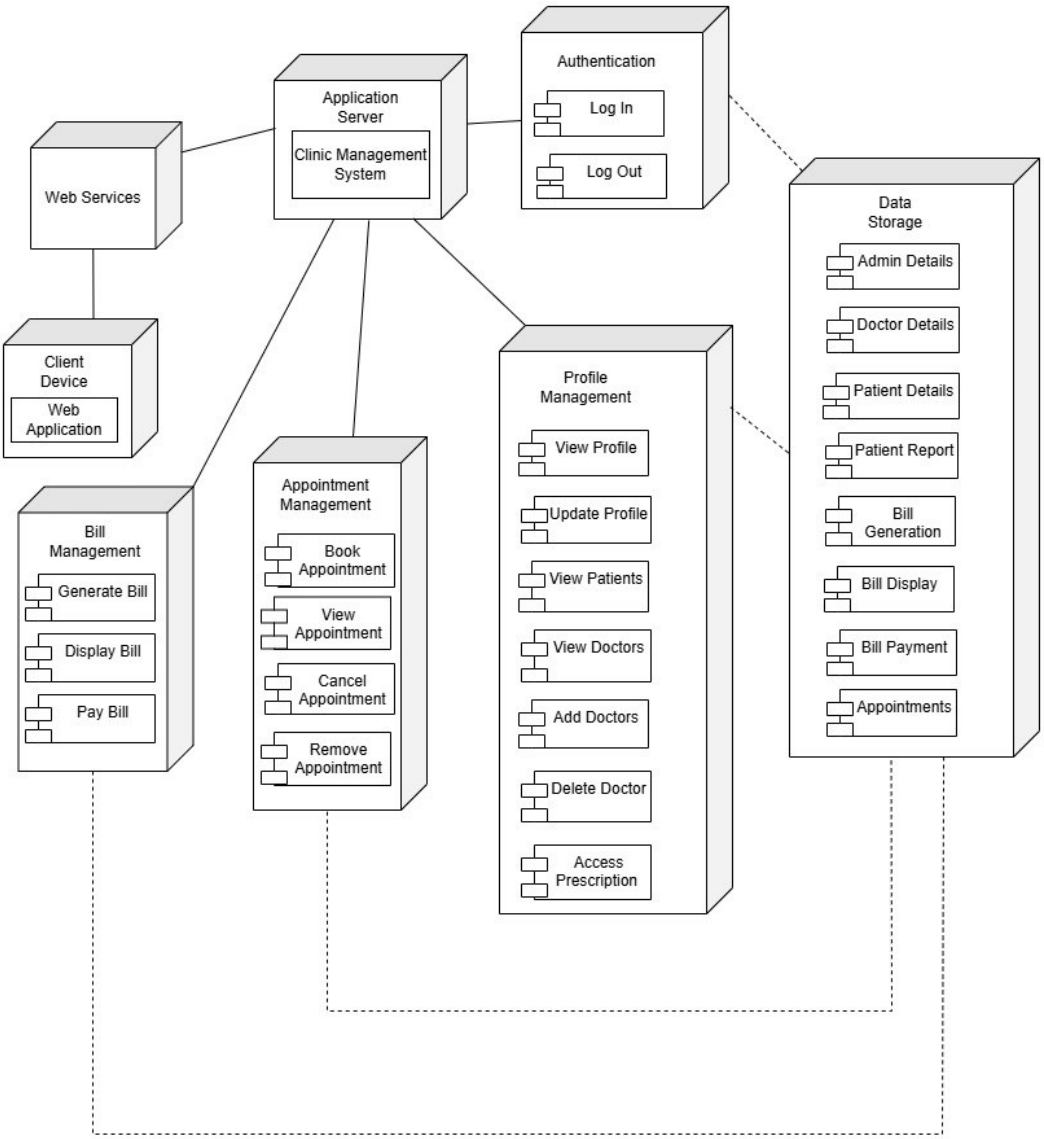

Figure 67: Deployment Diagram for Clinic Management System

Chapter 9: Conclusion

The development of a Clinic Management System is an essential step toward streamlining healthcare operations and improving patient care delivery. Through the analysis and specification of functional and non-functional requirements, this report demonstrates how the system addresses the needs of various stakeholders, including patients, doctors, and administrators.

By integrating features such as appointment scheduling, patient record management, billing, and secure authentication, the system ensures operational efficiency, reduces manual workload, and enhances user experience. The use of modern design and modelling methodologies has allowed for the creation of a robust and scalable solution that is adaptable to the evolving needs of healthcare facilities.

In conclusion, this system represents a significant advancement in the management of clinics, paving the way for better communication, improved organization, and enhanced patient satisfaction. With proper implementation, the Clinic Management System has the potential to set a new standard for healthcare management practices.