HOSPITAL MANAGEMENT SYSTEM

Software Requirement Specification Project

SE-406 : Software Requirements Specification and Analysis

Submission Date : September 30th, 2021

Submitted By

Group 1

Mohsin Ibna Amin (BSSE 1101)

Samiha Tahsin Noshin (BSSE 1107)

Md. Inzamam-Ul Haque Sobuz (BSSE 1113)

Jubaer Hossain (BSSE 1119)

Shafiq-us Saleheen (BSSE 1125)

Ahmed Adnan (BSSE 1131)

Junaed Islam (BSSE 1019)

Submitted To

Dr. Kazi Muheymin-Us-Sakib

Professor

Institute of Information Technology

University of Dhaka

Table of Contents

Introduction

Healthcare is a part of life we all consider to be something we are entitled to – it is our right to have access to healthcare whenever we need it. For a better life and our overall well being we can't ignore the importance of a well organized healthcare approach which is digitalized , neatly managed and responsive to every of our calls of need. An effectively performing health system is essential in improving the population's health status, providing safeguard against health-related financial threat and enhancing the health sector's responsiveness to customers needs. A health system consists of all organizations, people and actions, whose primary intent is to promote, restore or maintain health.

It is obvious that the healthcare system in our country is lacking all kinds of technological collaboration and an availability of user friendliness.
So with the advancement of technology and time, it has become a crucial need for our country to make advancements in healthcare policy with the collaboration of technology to make it efficient, user friendly and remove all types of manual mismanagement.

Moreover, the impact of advanced technology in the medical field can make Bangladesh a step ahead to be one of the developed countries by engaging in telemedicine, remote job service and use of AI in various fields. Thus, people of this country will be showered with the blessings of digital technology, remote communication and will be provided with health service at each and every corner of the country irrespective of all sufferings caused before.

Purpose

This document is a concise explanation of the Software requirements and specifications of the hospital management system. It is the bible that contains all the necessary requirements and documentation. It lays the framework that every team involved in development will follow. It's used to provide critical information to multiple teams — development, quality assurance, operations, and maintenance ensuring the teams are in agreement..this document defines what is needed from the product. It states the product's purpose and what it must achieve. All the functionalities, objects, and developing processes are explicitly described in this document which will later be helpful for the project verification and testing.

Intended Audience

In order to create a versatile, efficient and convenient system, developers must fully comprehend its future functions, users and supposed benefits. Only after this step, they will be able to plan and coordinate their efforts, choose the most appropriate technologies, anticipate potential problems and come up with the solutions. A hospital information management system, as its name suggests, is a complex software suite of interconnected program modules that serve to provide both information and management capabilities to a large variety of users.

The supposed users of a hospital management system may be generally divided into three categories:

1. Hospital administration
2. Doctors and other authorized employees
3. Patients.

Each category should have a separate set of rules for clearance and restrictions within the system. This set depends on a user's role and varies from unrestricted access granted to the highest administration members, to limited access to a certain portion of information for patients. For example, a doctor should have access to his patients'

medical data, but should not be allowed to get information on other doctor's patients without permission. Every user should have a unique identifier, which associates the person with that person's rank in the system. Users with higher ranks and more permissions within the system may need additional means of authorization.

Conclusion

This software requirement specification is more than helpful to identify the owner and stakeholder of the project. As it is working as the baseline of the project which reflects both the demand of the hospital managing party as well as the consumer of the system so that the system is up to the functional it needs to be. As this document is addressed as the bible of the project, the document content may evolve with the passage of modification and enhancement of the software whenever it is necessary.

Inception of the Hospital Management System of Ibn Sina

Inception of the project is attributed as the establishment or starting point of the activity of software requirements specification of Ibn Sina's hospital management system, defining the scope and limitations of the project. The actual goal of this term is to bring out the meaning of understanding and identifying the requirements of the software team and the stakeholders. So the significance of this term can be phased in the following order:

- Understanding the system utilities from the point of view of client
- Understanding the system utilities from the point of view of users (stakeholders)
- Icebreaking
- Basic structure of the project as an effective system

Understanding the system utilities from the point of view client

Generally the hospital managing authority , the admin is the client of this software project. Patients or customers, doctors and staff are defined as stakeholders in this project.

Understanding the system utilities from the point of view users (stakeholders)

In simple words, anyone having any type of relation/interest in the project is known as stakeholder. So from this perspective, the stakeholders are pointed as:

- Patients
- Doctors
- Staff

Icebreaking

Icebreaking is all about improving communication early and often, allowing the team to develop relationships with one another as the Agile development process progresses. It's a vital part of the software requirements specification as it initiates the communications and acceptance of the scheme.

Basic structure of the project as an effective system

A software system's structure is a division of that system into a set of parts and the relations between those parts. Confusion often arises because of the many types of parts and the many important relations between them. As there are basically two parties for this system , the structure can be visualized from different viewpoints from their perspective.

Hospital authority viewpoints:

- Above all, the system has a user friendly interface and much responsive
- The software is available irrespective of mobile and desktop platforms
- Notifies all the activities of patients and doctors
- Easy access to desk service
- Flexible scheduling availability of appointment from home
- Also maintains a small portion secured of account system carrying the transaction of doctor and patient
- Well organized database handling of patients' records
- Automated token generation for the appointment serial and follow ups

Customers' viewpoint:

- Instant service from help desk
- Platform oriented
- Both physical and online consultation facilities
- Flexibility of booking and cancelling appointments

Story Behind the Hospital Management

We are doing Software Requirements Specifications and Analysis in some sectors for automating the Hospital Management System of Ibn Sina Hospital, Dhaka. Though the medical sector is one of the most prominent sectors of our country, it is one of the most under-utilized and incoherent sectors of the country. Lack of communications on various levels, lack of accountability towards higher authorities, and the absence of a structured and digitized system are the main reasons behind it. Even though there have been some enormous technological surges in some sectors of our country, the medical sector is really lagging behind in terms of automation and digitalization. So, we have decided to move in this direction.

We have decided to choose Ibn Sina Hospital as our primary go-to place because their hospital management system has glimpses of automation. But it is not sufficient and some major and necessary sectors are not automated yet. Here is the existing condition of some significant sectors of a hospital management system that needs to be automated:

Appointment for Regular Medical Checkup and Test

The patients have to come to the hospital and stand in a large queue for a long time for diagnostic checkups and tests. Therefore, a huge amount of time is wasted and there is no guarantee whether a patient can make the cut or not in that small window of time. Standing in a large queue also puts the patients at risk in case of contagious diseases.

Payment Method

Paying for medical checkups, tests, medicines, and using hospital facilities is another hectic job. There is no structured payment system and the hospital management asks for money at any moment without any further notice. The payment system is all manual which causes more problems. But the hospital authority stores payment data which is a positive thing.

Receiving Medical Reports

Getting medical reports is another tiresome job. A patient needs to go to the hospital to collect the report manually and has to pay another visit to the doctor's chamber for the report's feedback. There is no option for receiving the report online and getting an online consultation from the doctor regarding the report.

Doctor's Timing and Scheduling

Though a fixed schedule is appointed to every doctor for treating patients, it is not followed properly by the doctors due to lack of accountability to higher authorities. As a result, valuable time and efforts are wasted by the patients and they don't get their treatments in time.

Prescriptions and Medicines

The prescriptions are manually written and a patient has to collect it and give it to the pharmacy physically. There is no system to automate it in such a way that the prescriptions would be created in a software and would be sent to the pharmacy online. This would make the system of collecting medicine a lot easier.

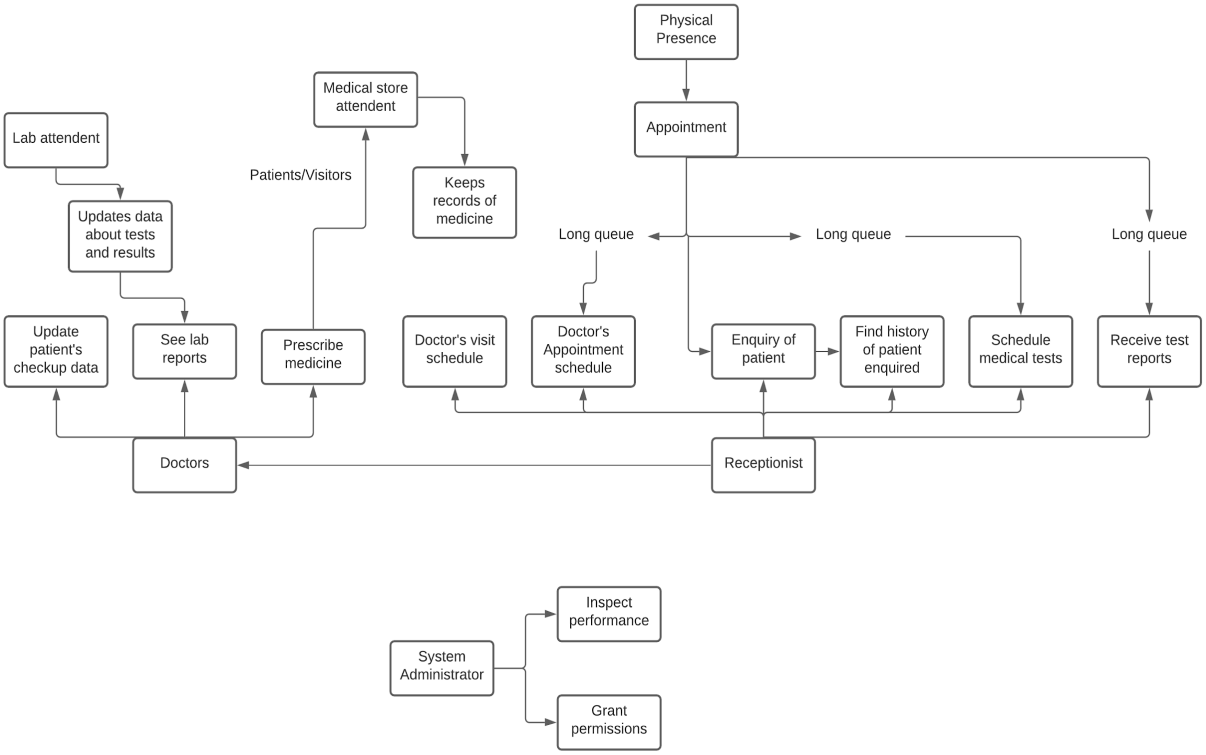


Figure-1: Block diagram of existing Hospital Management System

After observing the existing system and taking the points mentioned above into consideration, we have decided to do automation in the following areas:

1. **Online booking system** for medical tests and checkups.
2. **Online payment system** in a structured way.
3. **Online medical report exchange** and **doctor's consultation**.
4. **Automation of the schedule** for medical tests and checkups.
5. **Automated prescription creation** and getting medicines online using that prescription.

Quality Function Deployment (QFD)

Quality function deployment (a model for product development and production) is a focused methodology for carefully listening to the customers or consumers of that particular product and then effectively responding to those identified needs and expectations.

We visited Ibn Sina Hospital in Dhanmondi and had a discussion with a few doctors, patients and staff members. They expressed their dissatisfaction over a few things in their current hospital management system. Their reactions and suggestions have given us a clear view of the product's expected requirements and a possible solution.

Normal requirements in QFD

In normal requirements , the motive and perspective or the aim of the project are defined by collecting the statements of the customers. These statements' collection upholds the basic system procedure that is to be followed while turning the system product in an automated manner.

- **Login system:** The patients will hold individual profiles on the hospital website once they get an appointment.The doctors will also have their profiles on the dashboard classified or divided according to their respective departments.

- **Notification system:** the patients will login to their profile for availing service, information and receive notification about their appointment schedule and other procedures, and leave feedback about services.

- **Authority activity:** The hospital authority will supervise and maintain distribution and scheduling, transactions.

Expected requirements in QFD

In an existing system or service, the main demand of the customer beholds the user friendly objects and system approach. As we recently visited Ibn Sina, we came across the expectations of customers classified into three different categories according to their position. They are the doctors, the hospital staff and most importantly the patients.

According to doctors' points of view:

- **Online Consultation:** A responsive online consultation system for follow-up of the patients who have already got a consultation from respective doctors regarding their previous medicine course outcome (In case of physical check-up not necessary).

- **Online Appointment System:** Switch to a website based appointment system from phone call and desk based ones. Though the hospital has a website for some of those requirements but unfortunately it's abandoned at present.

- **Digital Prescription:** Primarily the prescriptions are in handwritten form which is quite ambiguous most of the time for the patients and toiling for doctors. So the doctors suggest a computerized prescription system served to the patient in soft copies. But they are also afraid of engaging manpower in this job, but also expecting a new territory of job space.

- **Referral Section:** The most important one that they emphasized on is group networking, that means an organized communication between different sectors of the hospital and even referring the patients to other specialized hospitals when needed. So, we can add an option for the referral section.

- **Digital Medical Records:** A patient appointment log system will be maintained so that the records can be preserved efficiently for further and future procedures. A paperless hospital management system is what they desire for the future.

From the point of view of the patients, the expected requirements we noted:

- **Schedule Management:** First of all, the patient's main concern was about the time wasted in booking appointments physically on desk and in the queue before visitation. So, the appointment scheduling system will use statistical data to analyze and predict the time which will have the minimum possibility of being delayed.

- **Online Test Reports:** All the lab reports will be delivered to the patient's profile. Follow up routine should be held online through phone calls or other virtual conference media (when any physical check-up is not necessary).

- **Online Desk Service**: A few help desk sections will be available for enquiries where a few representatives will always be available to reply. These desks will be-

    - Information Desk
    - Bill Investigation Desk
    - Cardiac Service
    - Emergency Help
    - Bill Payment Desk

The hospital staffs also upheld their expectations in the existing system to be further automated:

- The staff expects less duty time physically attending the job while they can render the services online as much as feasible.

- In an ongoing pandemic situation, they suggest avoiding the physical presence of patients for those procedures which are both irritating for desk staff and patients (except admitted patients).

Exciting requirements in QFD

These features go beyond the customer's expectations and prove to be very satisfying when present. Often exciting requirements enable the scope of innovation in the process or new ways of handling functionality. Stakeholder satisfaction with the application can be dramatically improved through the implementation of a few exciting requirements. For our project, we've identified some of the requirements which are given below.

- ❏ An AI powered chatbot can monitor mental health and provide suggestions on how to tackle depression or anxiety. It can also provide actual medical help from doctors if necessary.

- ❏ Hospital authorities will be able to manage their resources, analyze staff work, reduce inactivity and optimize work progress with an active monitoring feature.

- ❏ Staff interaction is necessary for improved coordination and teamwork. An instant messaging system with the feature of selecting your work state can ensure a quick response from the staff and can diminish the communication gap.

- ❏ An active blood bank, which will also be open for blood donations.

Usage Scenario of HMS

User Account Management

In an automated system of the hospital, all the people related to hospital service (both service provider and user) must hold a designated unique profile, which will be considered as their account through which they can continue and communicate with for any type of procedures. The profile will be of four types according to users' perspective in this system: patients, doctors, staff and the admin of the hospital.

1. Patient's Profile

In order to book an appointment, every patient should **create an accoun**t on the **hospital website**. For creating an account, the patient should provide his/her -

- First Name
- Last Name
- Gender
- Date of Birth
- E-mail Address
- Mobile Number
- Password

An **OTP** will be sent to their mail or mobile number to confirm the verification.
Or - They can use- "**Sign in with Google"** or "**Sign in with Facebook"** option to sign in.
A restriction should be imposed on email accounts and mobile numbers. Only **one account** can be created using the same email address/mobile number.

2. Doctor's Profile

Every doctor of the hospital will also hold their personal profile containing their name, specialized department, degrees, speciality, availability schedule etc. given by **the administrator**.

3. Staff's Profile

Staff profiles are also created and managed by **the admin.**

4. Admin's Profile

As the admin is in charge of the whole system, this account holder type is **pre-set** while the system is built and maintained cohesively and consistently.

Patient's Usage Scenario

1. Search Bar

A search bar will be shown at the top of this page where the user can search for- **Doctors, specialities, symptoms, diseases, treatments etc.** This section will automatically suggest the available best doctors according to the speciality, symptom, disease or treatments provided by the user in the bar. The user can also sort the shown list using different variables like experience, visiting fee etc.

2. Doctor's Information

- Photo
- ID
- Name
- Gender
- Qualification
- Department of Specialization
- Consultant at - (Name of the branch)
- Experience
- Language
- Availability - Slots with time division (morning, day & evening)

3. Book Appointment

For booking an appointment, a filter window (select a time slot to book an appointment) will be shown. Filter's options are:

- Branch
- Date
- Time (Morning, Afternoon, Evening)

Then, the user will be shown appointment booking status (Available/No slots available). If status is positive, he/she can see the cost of the visit and confirm the appointment. Then, he/she will get an online payment pop-up message. After the payment, an online token will be provided.

Once appointment is done, patient's user profile will have the following sections:

(i) Appointment Section

- **Date and time of appointment schedule**
- **Doctor's profile**
- **Type of appointment** -
    - Physical check-up
    - Online appointment
- **Token number**
- **Option for rescheduling**
- **Cancel appointment** - Online cancellation shall be permitted only up to six hours before the scheduled appointment. x% refunding is available only if the user cancels the appointment before that six hours deadline.

If there is no scheduled appointment, the section will have a "create appointment" button.

(ii) Check-up History

After the check-up, the section will show all the history including-

- **Date of the check-ups**
- **Online prescription**
- **Comments** (if doctors have any)
- **Referred to** - If the doctor suggests the patient to any other departments or hospitals, then the information will be shown here.
- **Surgery history**

(iii) Medical Tests Reports and Scheduling

Once the doctor adds to this section, the patient user will see a list of given medical tests. Under this option, there will be two possible choices-

- **Schedule the test** - Available slot sections will be shown. If the user selects one and confirms, he will be shown the **online payment pop-up message** along with the cash memo.

- It will automatically generate an expected date for the test reports to be given. If the report is available, the user will be notified and the soft-copy will be found here in this section.

- **Upload the test reports** - If the patient wishes to do all the medical tests from somewhere else, he/she can simply complete their tests and upload the report copy here as PDF.

Once the report has been delivered or being uploaded, the user can schedule an appointment for report checking with his/her doctor. It can also be an online consultation request. For online consultations, we have the next section.

(iv) Online Consultation Request

The user can request for an online consultation meeting with the doctor. The reason could be **follow up routine checking, test reports checking or random check-ups**. The doctor will provide a time slot for the meeting. If the doctor thinks a **physical check-up** is necessary rather than the online consultation, he/she will suggest booking an appointment for the physical check-up.

If the online meeting happens, on the allotted time slot, a chat box will appear with the doctor having the following options-

- Message
- Audio Call
- Video Call

The patient and the doctor can decide in which way they would like to communicate.

4. Quick Book

This option can be used for direct phone calls to book an appointment, schedule a medical test or online consultation if it's an emergency.

5. Health Library

- Doctor's can write blogs.
- Health Tips
- Hospital's news
- Special sections: (like covid-19)
- Q&A section (open for all)

Doctor's Usage Scenario

1. Personal Information

- Basic Information (name, gender, age, photo etc.)
- Educational Qualifications
- Preferable area for practicing
- Experience
- Research and Other Activities
- Contact Information

After logging in, the doctor will see-

- **Patient's Appointment List**
- **Online Consultation Request List**
- **Hospital Information Section**

2. Patient's Appointment List

The doctor will see the list of all the patients who have scheduled appointments with him/her. The patient's list will contain-

- Patient's Name
- Patient's Contact Info
- Appointment Time

After doctor **selects** one of the patient's from the appointment list, he/she will see the following features:

- **Patient's Medical History**
- **Test and Report Display**
- **Prescription creation Tab**

3. Patient's Medical History

This section will show the patient's past medical history. It will contain-

- History of Past Checkups
- History of Previous Diagnostic tests
- Previous Medical Reports
- Previous Prescription

4. Test and Report Display

There will be a section for displaying **required test results** and **reports**. So, this test and report display section will help the doctor to **keep track** of the ongoing checkup.

5. Prescription Creation Tab

This section will enable the doctor to create a prescription digitally. The prescription tab will contain the following things-

- **Auto Generated Patient's Information**
- **A List of Medicines and Medical Accessories :** This section will show the following sections chronologically-
    - **The list of Diseases/Problems :** The doctor can select the patient's disease/problem and all the related medicines and accessories will be shown in the suggestions. The doctor can select from there and the selected things will automatically be added to the prescription.

  After selecting and adding a medicine to the list, the doctor will have to select-
    - For how many days
    - How many times a day
    - If side effects happen, the alternatives (if required)

6. Online Consultation Request List

Sometimes, a patient needs to see a doctor for further consultation. So, in this tab, the doctor will see the consultation request list of both online and offline separately. This list will contain-

- Patient's Name and contact details
- Patient's Suggested Time for Consultation

After seeing a consultation request, the doctor can-

- Arrange a physical consultation instead
- Set up a Preferable Consultation Time according to the Doctor's Preference

7. Hospital Information Section

This section will contain hospital related news, health library, administrative info and other updates.

Most of these jobs will be done by the doctor's assistant.

Online Desk Service

In online desk service, the facilities provided by information desks in the hospital need to be added. Following types of online desk service will be available for dedicated issues:

1. Help Desk

One or more staff will be assigned to the help desk to render the patients or other investigators with required information, direction and instant reply. Also-

- Any department or doctor
- Doctors availability
- Bill investigations
- Recent addition in service
- Previous any record

2. Cardiac Desk

This specialized section will have staff to provide information regarding cardiac issues, tests and other information and suggestions through instant reply 24/7 .

Staff's Usage Scenario

Staff will be assigned to different desks set by the administrator.

The doctor's assistant will also be considered as staff and hold an account with his/her respective doctor's appointment list, online consultation list etc. Once the check-up is completed, he/she will mark them as completed.

Notification

Once a patient login to his/her profile using secured password, he/she will be notified with-

- **Appointment Reminder** - One reminder will be given 24 hours before the appointment and another will be 6 hours before.
- **Payment Confirmation** - A cash memo will be received as well.
- **All the Latest** - New updates at the health library will be notified.

Online Payment

This system will have all the current popular payment options available in Bangladesh including bKash, Rocket, Nagad etc. Payment with credit card (VISA, Mastercard etc.) will also be allowed. In case of rescheduling or timely cancellation, the fee will be returned to the user account following a few rules and regulations and will be notified (some percentage will be deducted as appointment cancellation fee).

Database

The database will have **a few tables**. Admins will be able to create or update any table.

1. **Doctors**
2. **Staffs**
3. **Patients**
4. **Medicines**
5. **Test_info**
6. **Payment**

Use Case Diagram : Hospital Management System

What is Use Case Diagram

A use case is a written description of how users will perform tasks on your website. It outlines, from a user's point of view, a system's behavior as it responds to a request. Each use case is represented as a sequence of simple steps, beginning with a user's goal and ending when that goal is fulfilled. Use cases specify the expected behavior (what), and not the exact method of making it happen (how). Use cases once specified can be denoted both textual and visual representation (i.e. use case diagram). A key concept of use case modeling is that it helps us design a system from the end user's perspective. It is an effective technique for communicating system behavior in the user's terms by specifying all externally visible system behavior.

The Purpose of Use Case Diagram

The reasons why an organization would want to use case diagrams include:

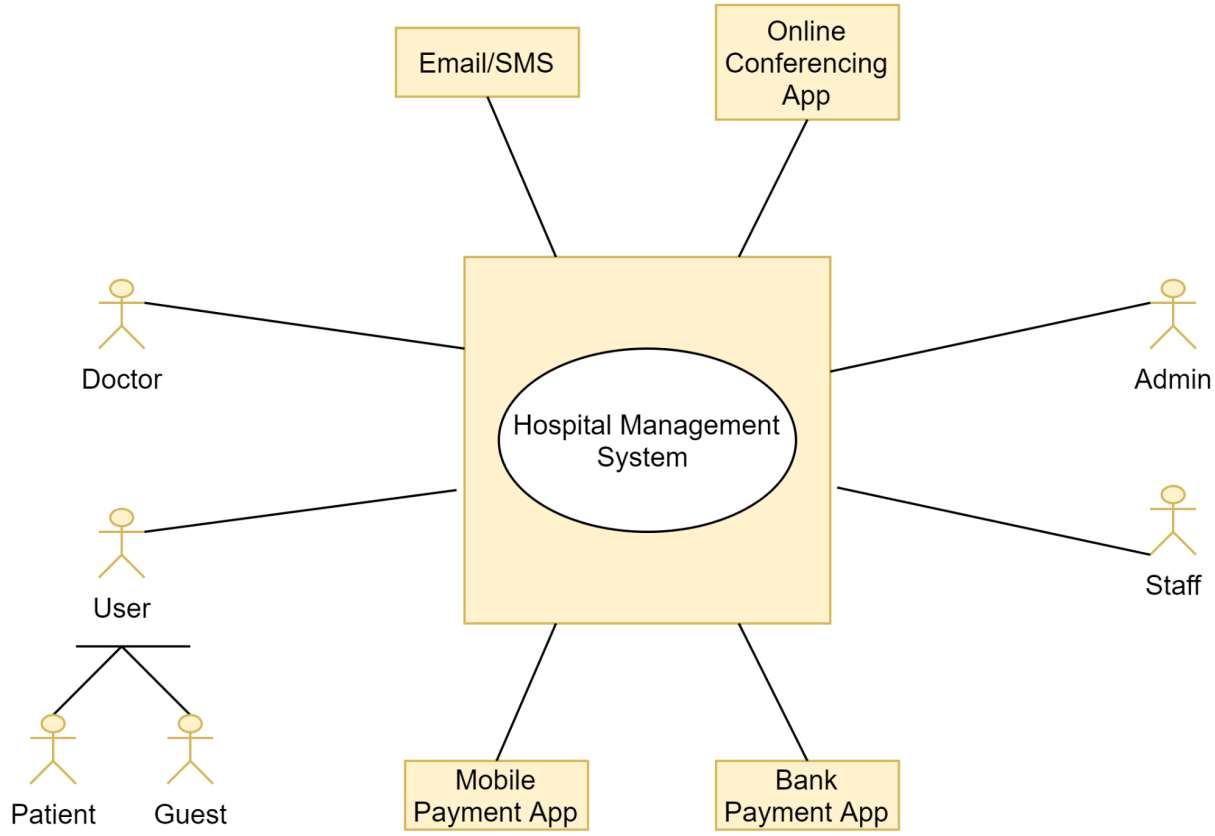
- Represent the goals of systems and users.
- Specify the context a system should be viewed in.
- Specify system requirements.
- Provide a model for the flow of events when it comes to user interactions.
- Provide an outside view of a system.
- Show’s external and internal influences on a system.

Level - 0

Name : Hospital Management System

Primary Actors : Doctor, User(Guest, Patient)

Secondary Actors : Admin, Staff, Email/SMS, Online Conferencing App, Mobile Payment App, Bank Payment App



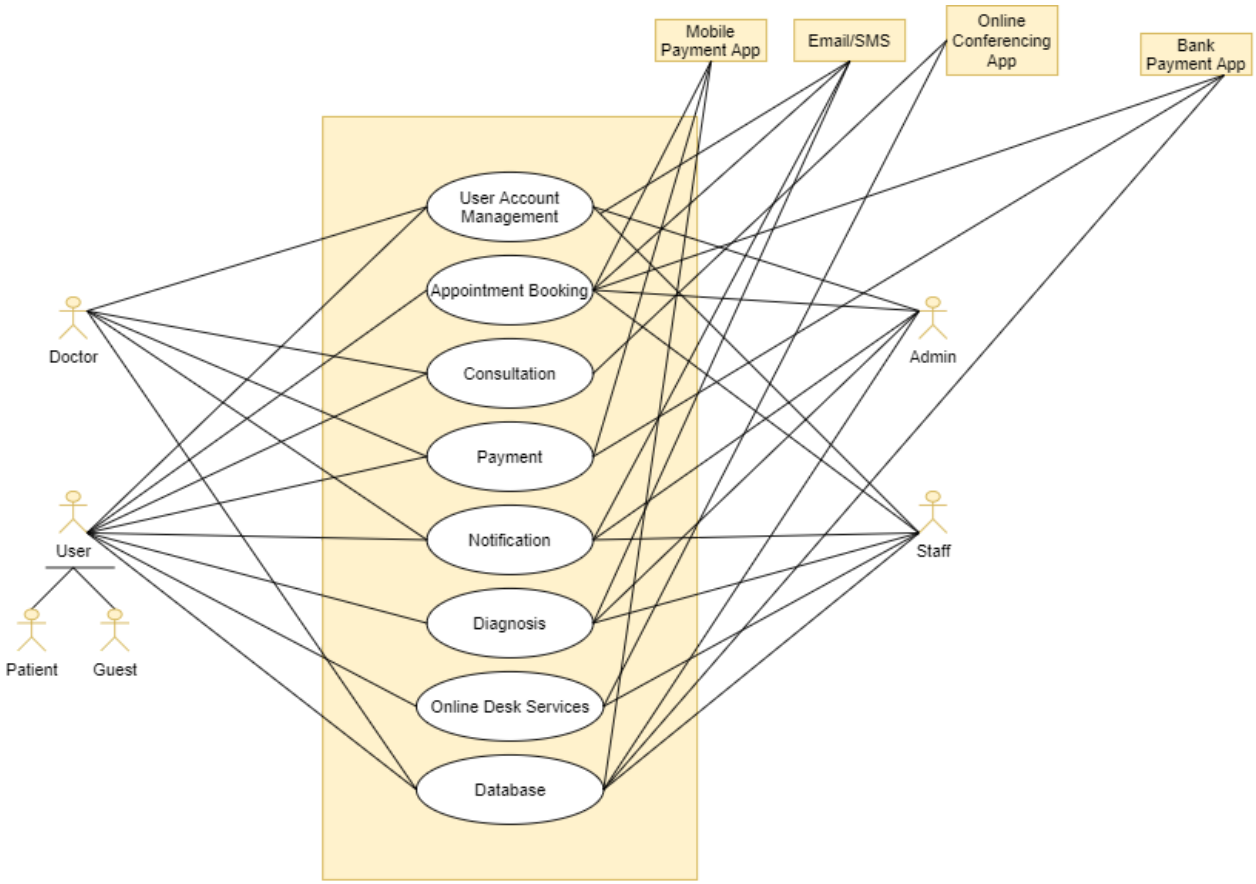
Level- 0 - Hospital Management System

Level-1

Name : Hospital Management System

Primary Actors : Doctor, User(patient, guest)

Secondary Actors : Admin, Staff, Email/SMS, Online Conferencing App, Mobile Payment App, Bank Payment App



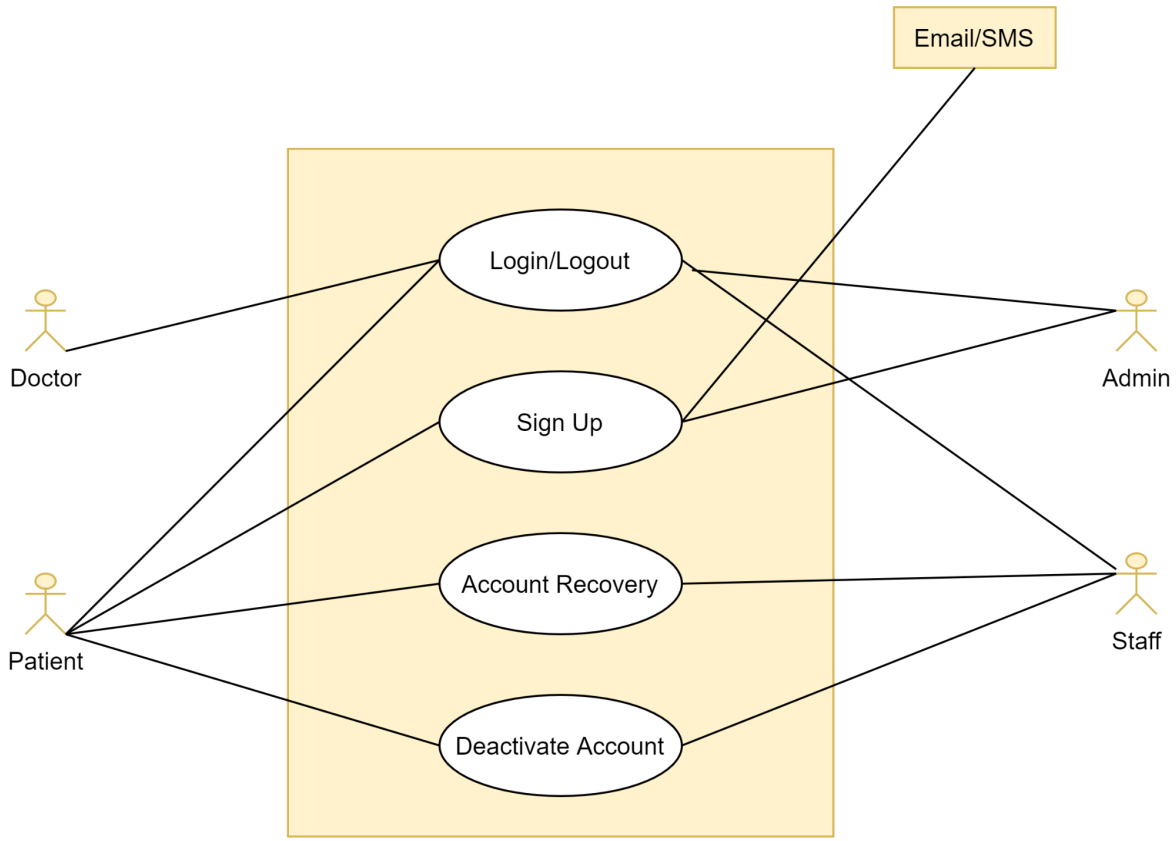
Level- 1 - Hospital Management System

Level-1.1

Name : User Account Management

Primary Actors : Doctor, Patient

Secondary Actors : Admin, Staff, Email/SMS



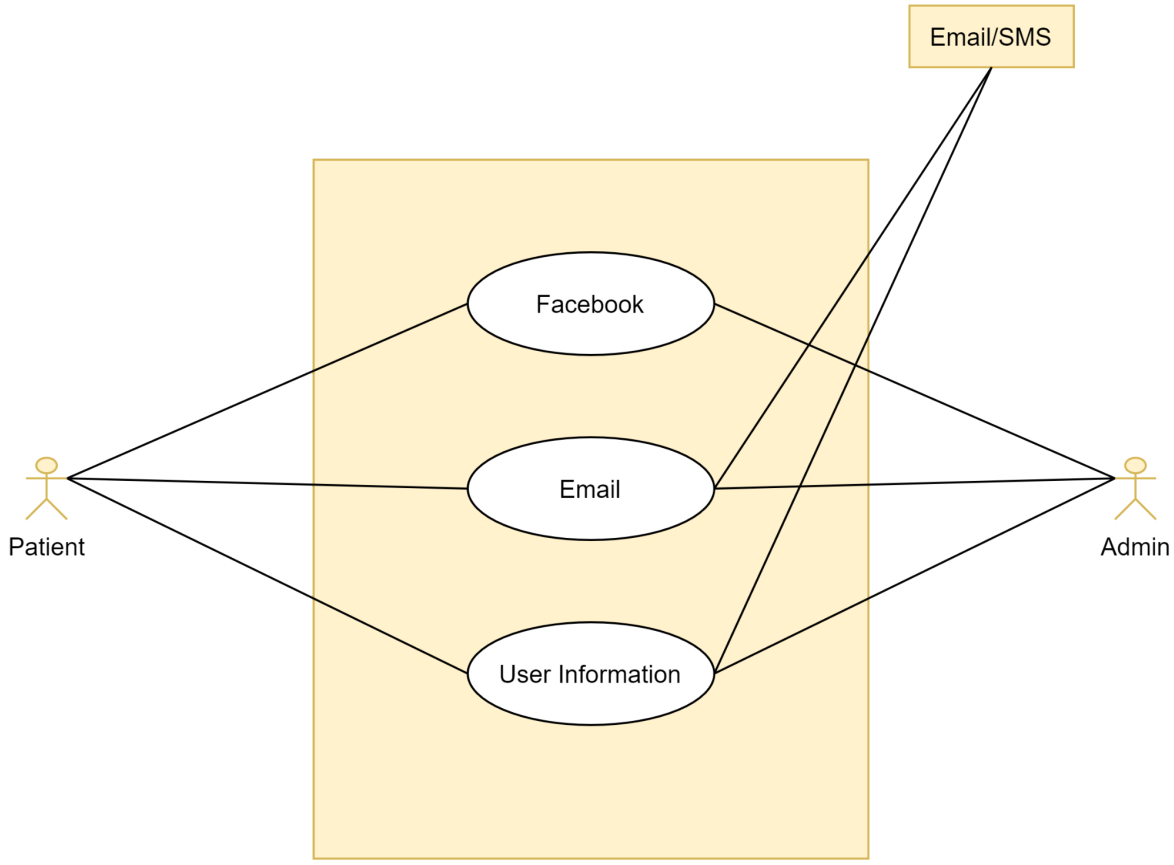
Level- 1.1 - User Account Management

Level-1.1.2

Name : Sign Up

Primary Actors : Patient

Secondary Actors : Admin, Email/SMS



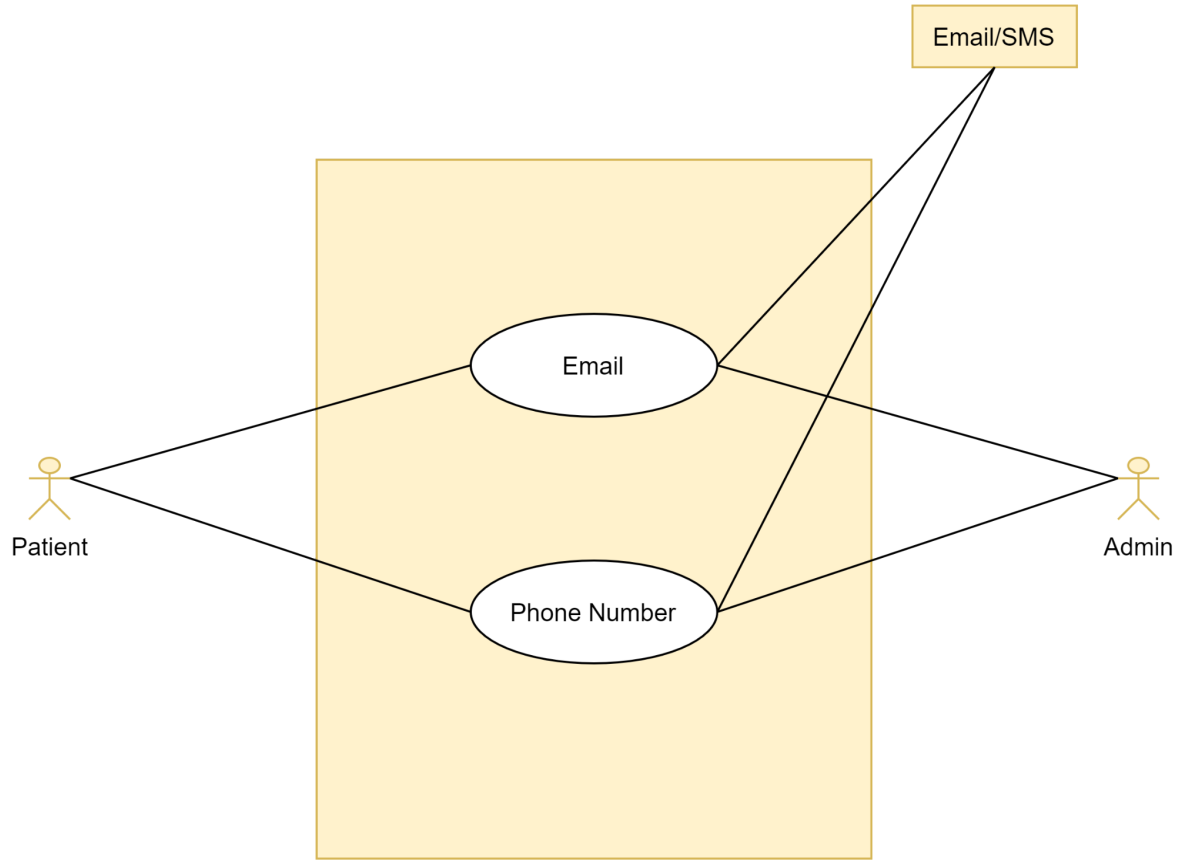
Level 1.1.2 - Sign up

Level-1.1.3

Name : Account Recovery

Primary Actors : Patient

Secondary Actors : Admin, Email/SMS



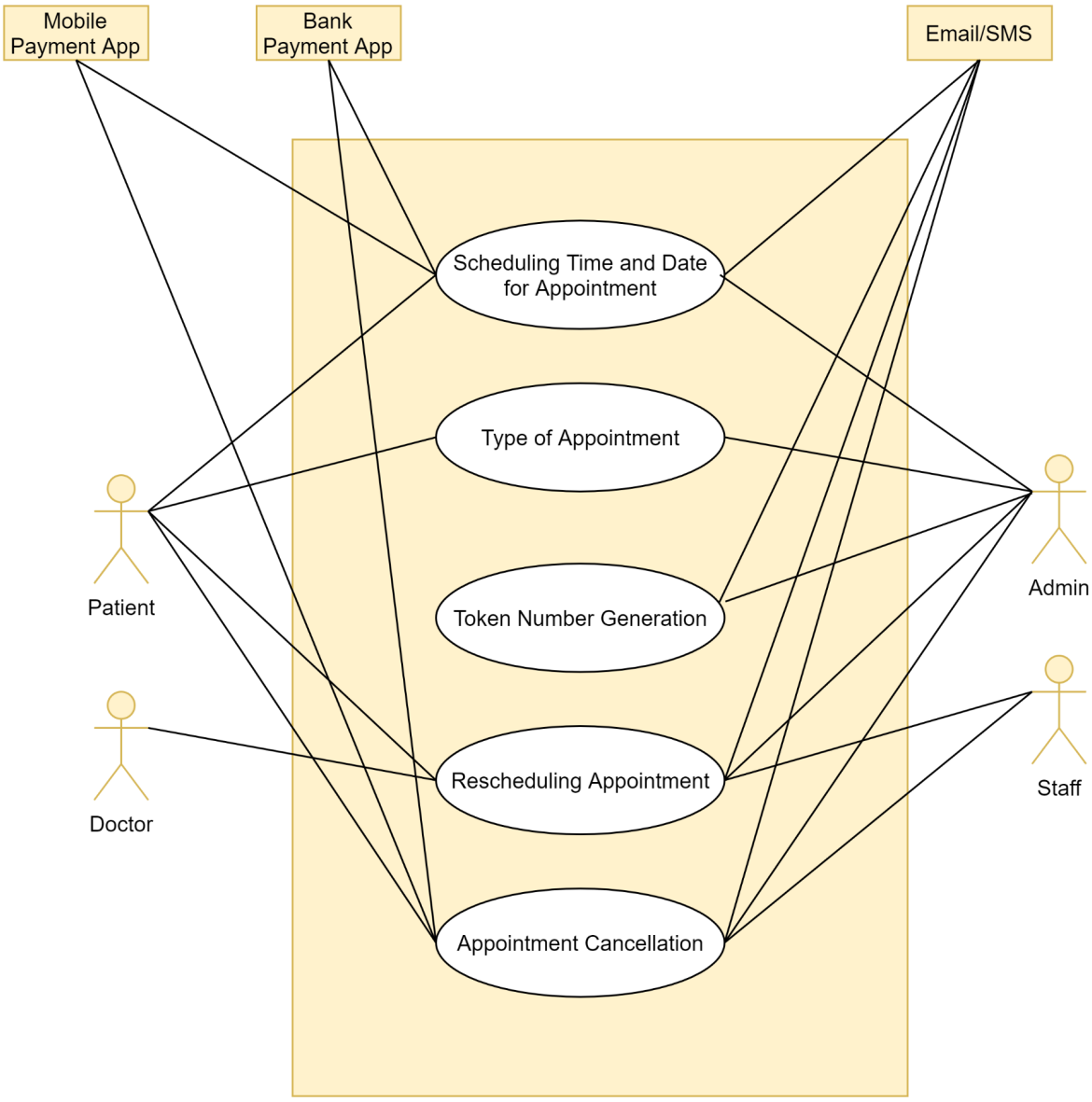
Level 1.1.3 - Account Recovery

Level-1.2

Name : Appointment Manager

Primary Actors : Patient, Doctor

Secondary Actors : Admin, Staff, Email/SMS, Mobile payment app, Bank payment app



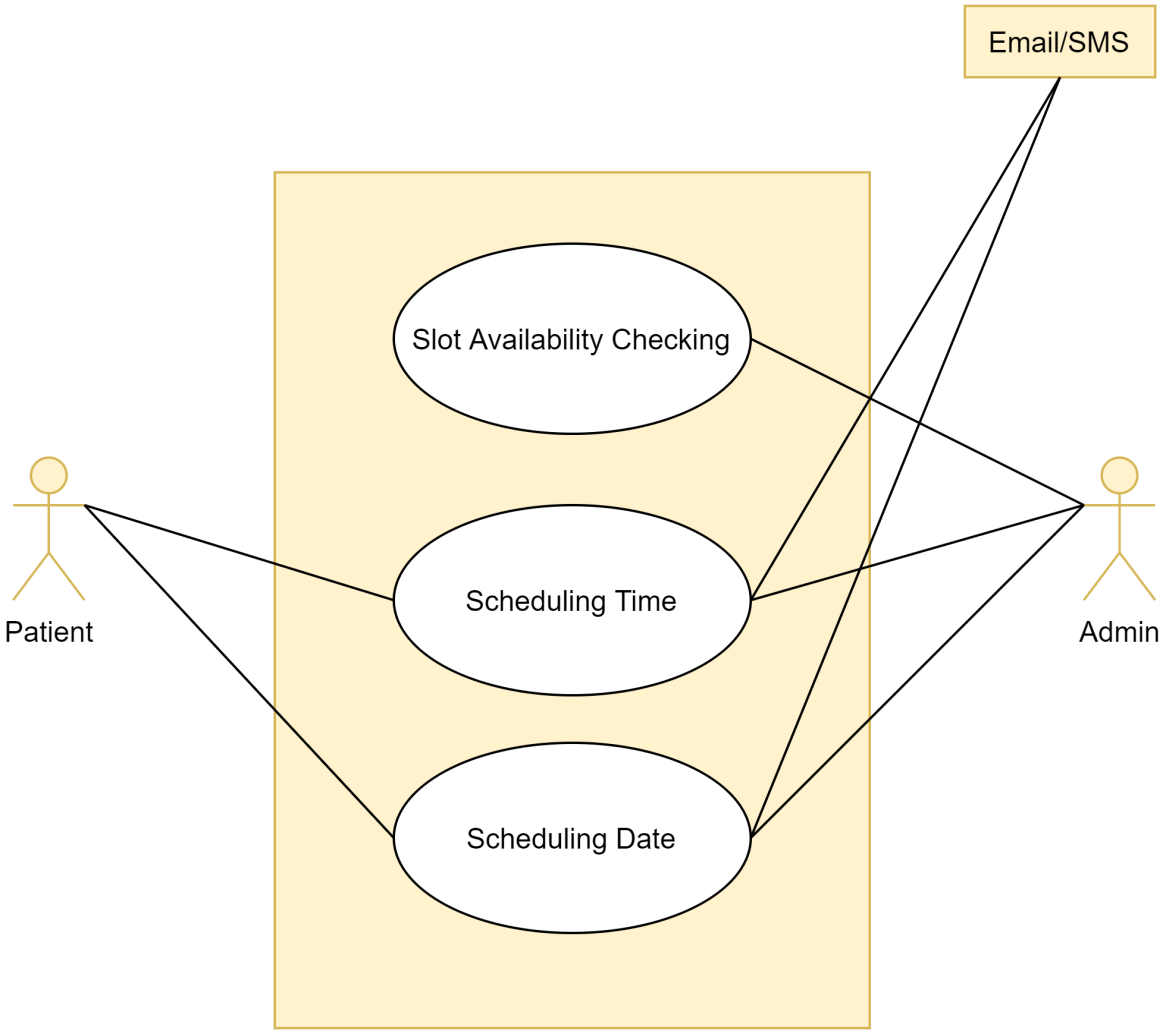
Level 1.2 - Appointment Booking

Level-1.2.1

Name : Scheduling time and date for appointment

Primary Actors : Patient

Secondary Actors : Admin, Email/SMS



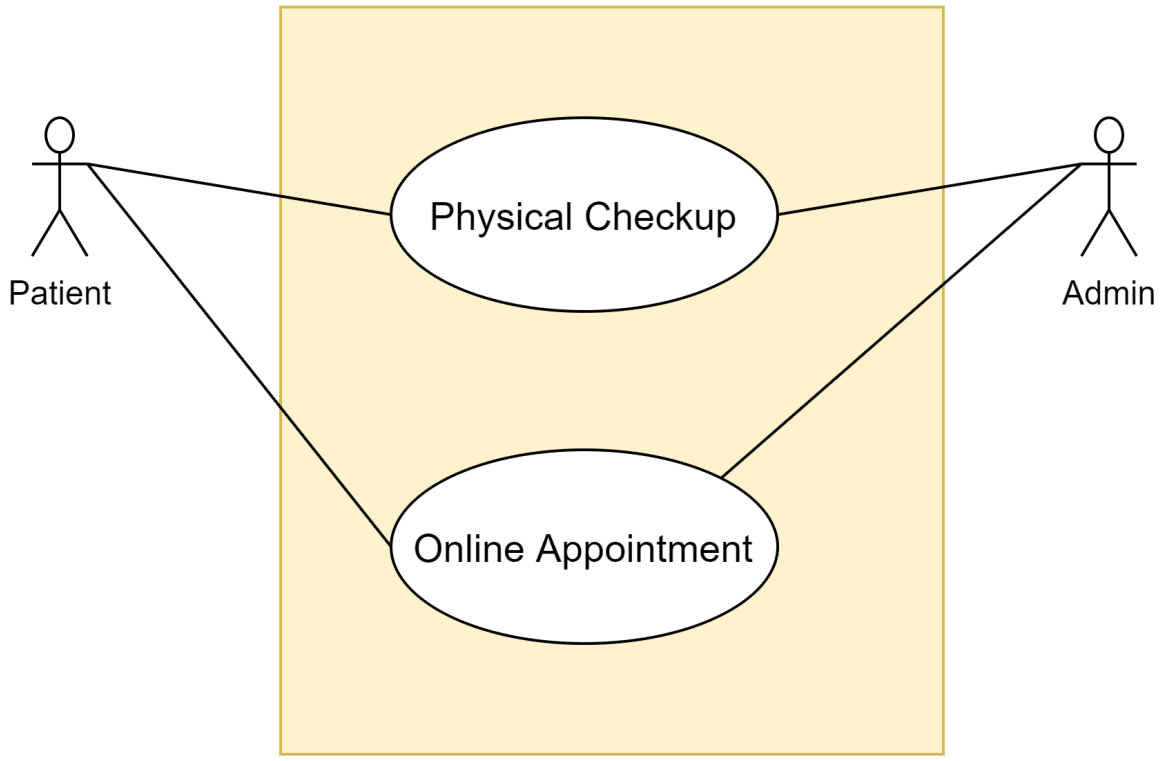
1.2.1 - Scheduling Time and Date for Appointment

Level-1.2.2

Name : Types of Appointment

Primary Actors : Patient

Secondary Actors : Admin



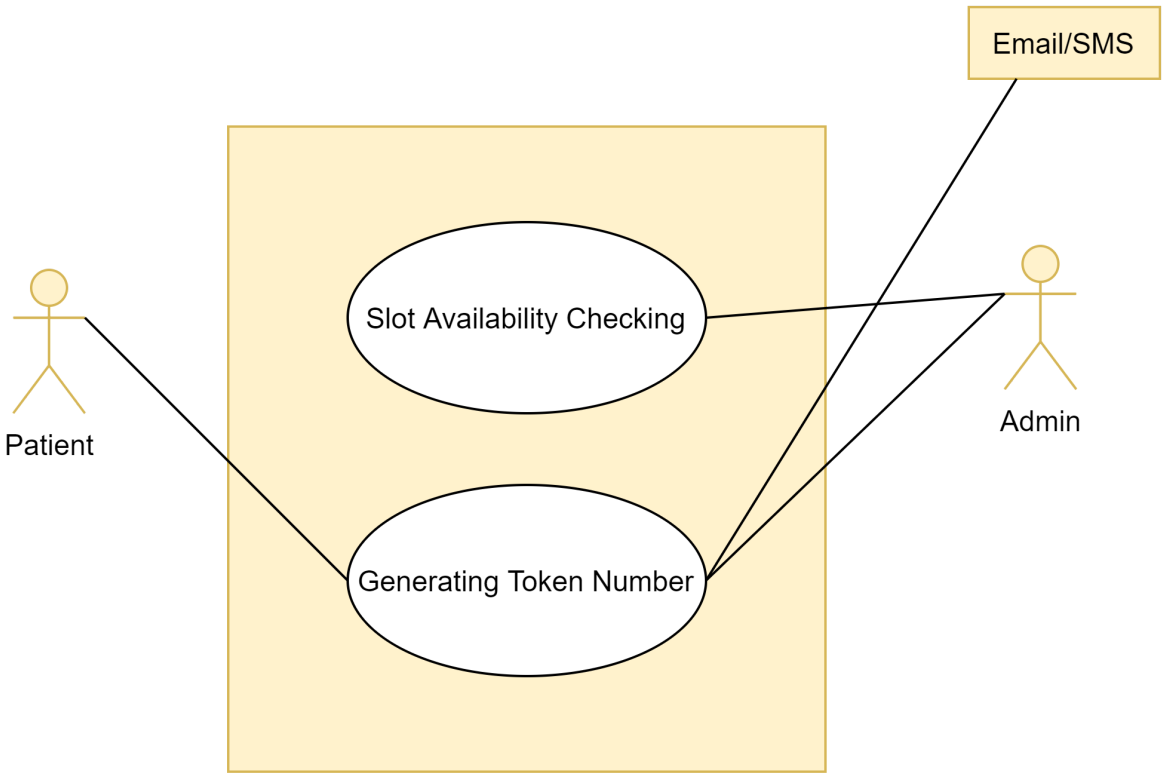
1.2.2 - Type of Appointment

Level-1.2.3

Name : Token Number Generation

Primary Actors : Patient

Secondary Actors: Admin, Email/SMS



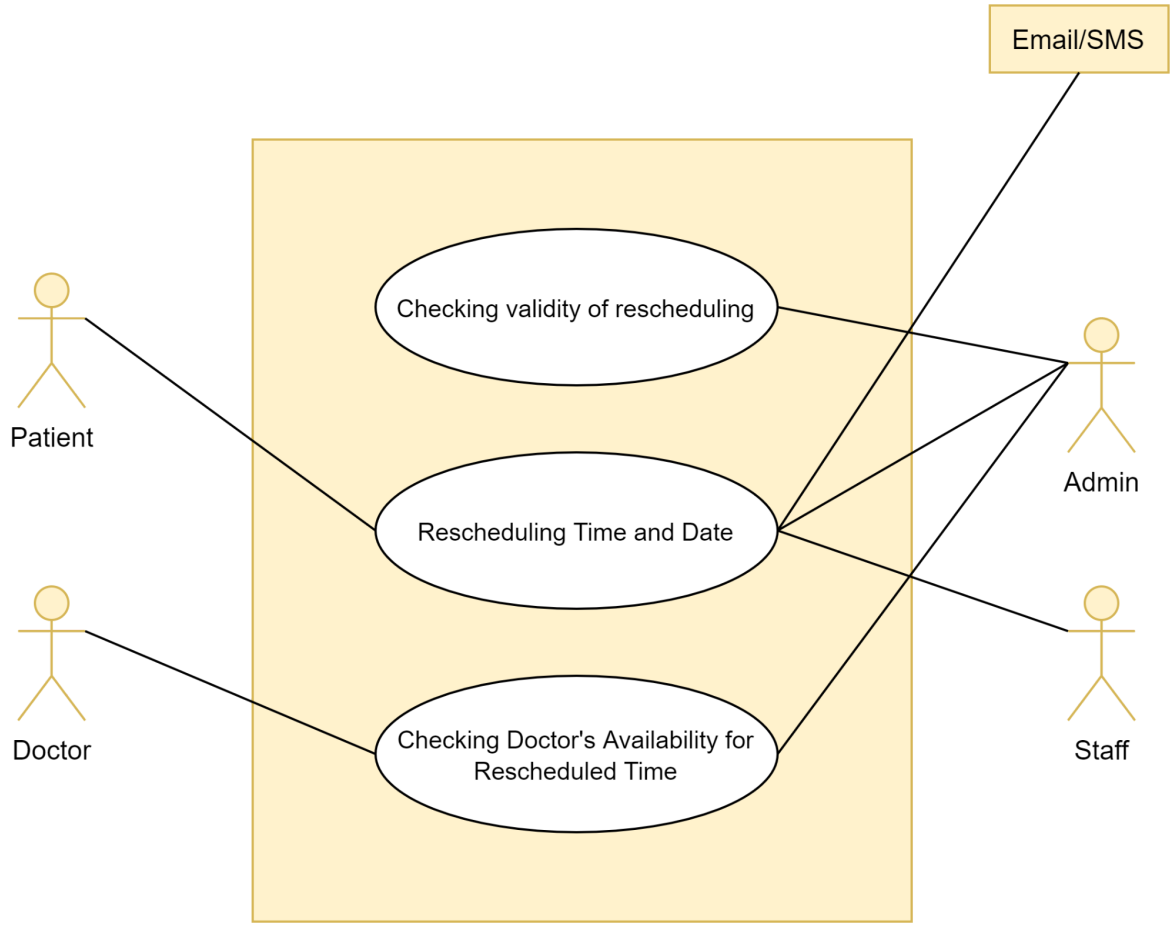
1.2.3 - Token Number Generation

Level-1.2.4

Name : Rescheduling Appointment

Primary Actors : Patient , Doctor

Secondary Actors: Admin , Staff, Email/SMS



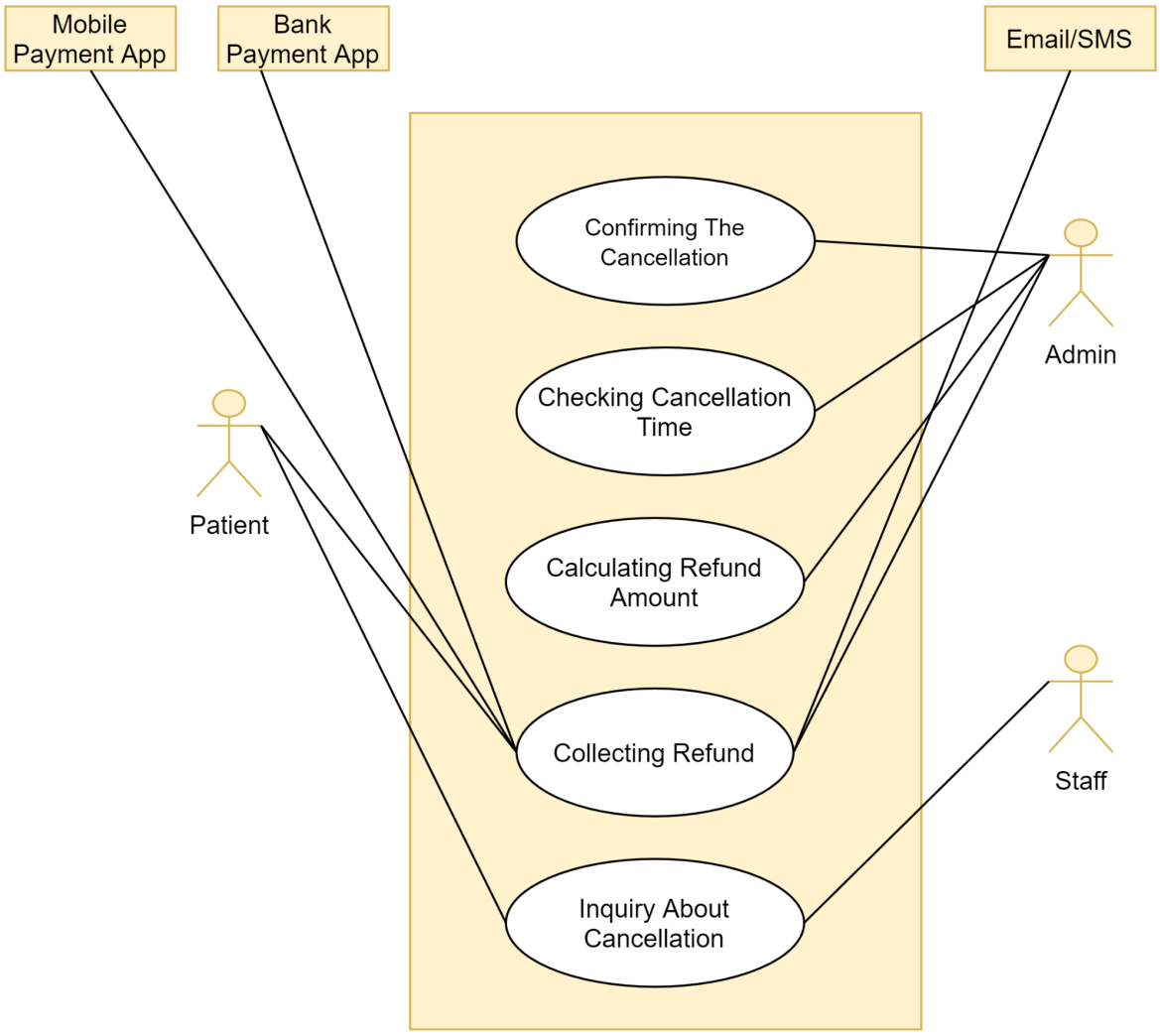
1.2.4 - Rescheduling Appointment

Level-1.2.5

Name : Appointment Cancellation

Primary Actors : Patient

Secondary Actors: Admin, Staff, Email/SMS, mobile payment app, bank payment app



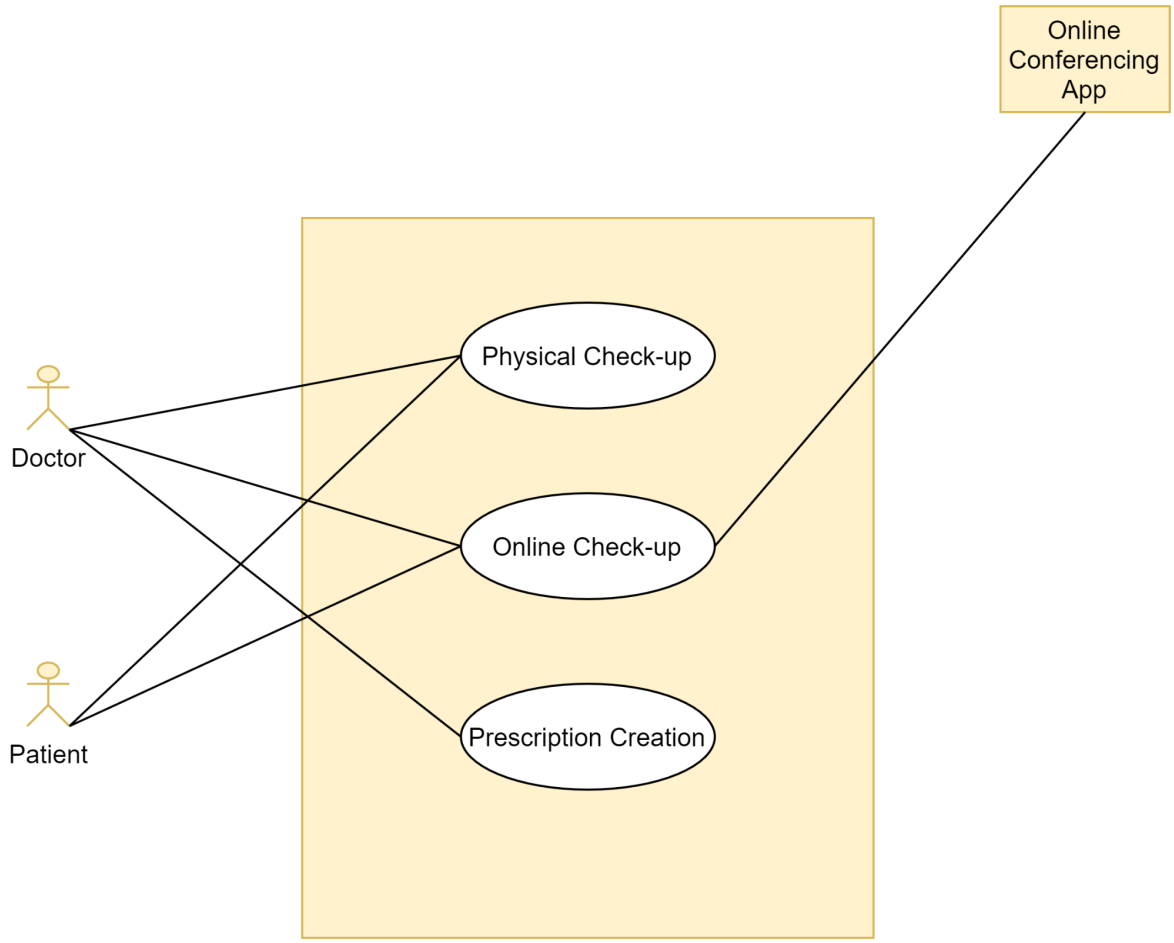
1.2.5 - Appointment Cancellation

Level-1.3

Name : Consultation

Primary Actors : Doctor, Patient

Secondary Actors : Online Conferencing App

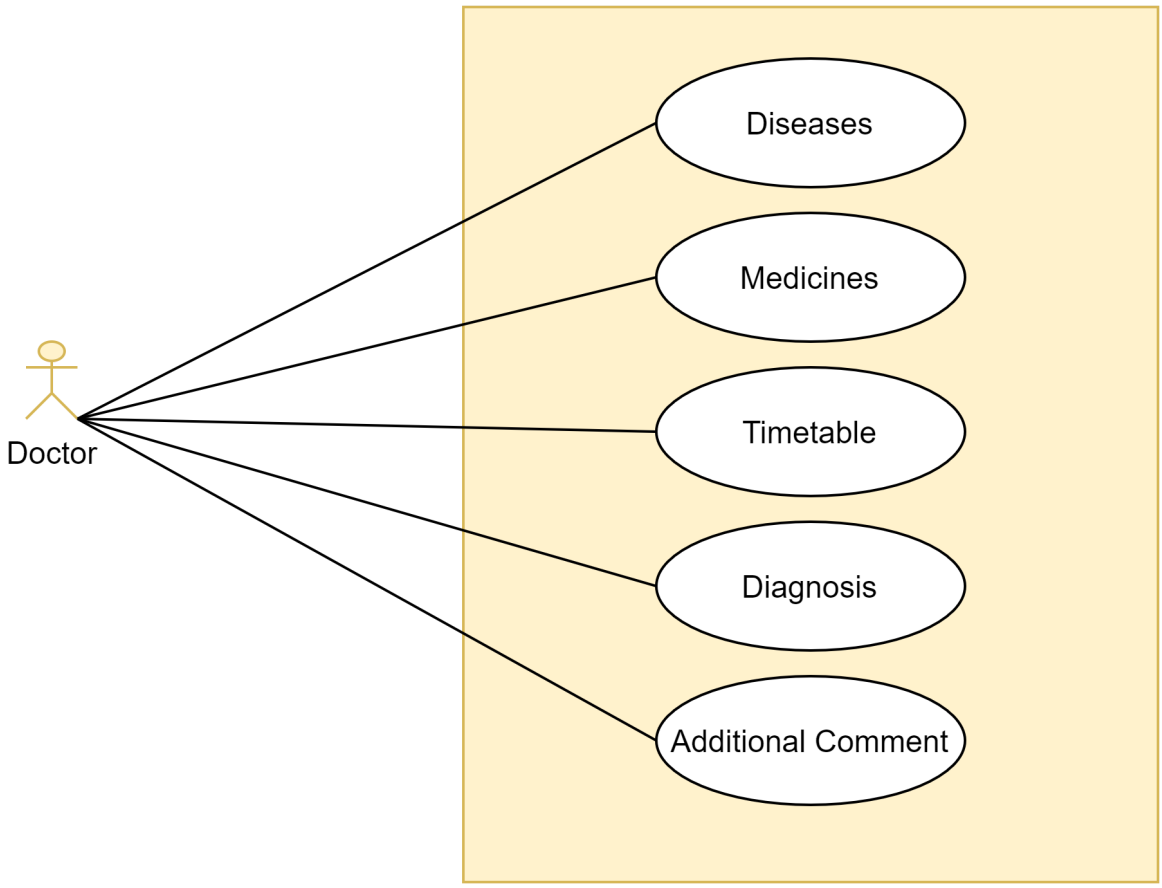


Level- 1.3 - Consultation

Level-1.3.3

Name : Prescription Creation

Primary Actors : Doctor



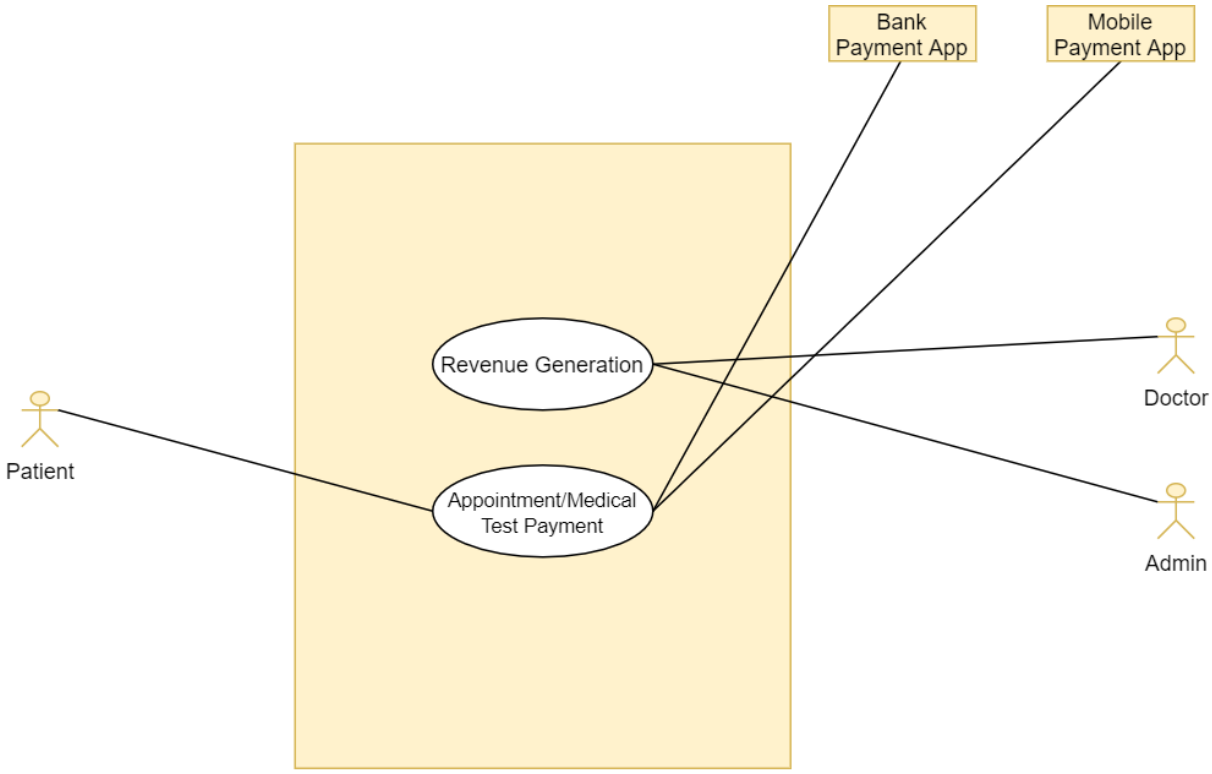
Level 1.3.3 - Prescription Creation

Level-1.4

Name : Payment

Primary Actors : Patient

Secondary Actors : Doctor, Admin Mobile Payment App, Bank Payment App

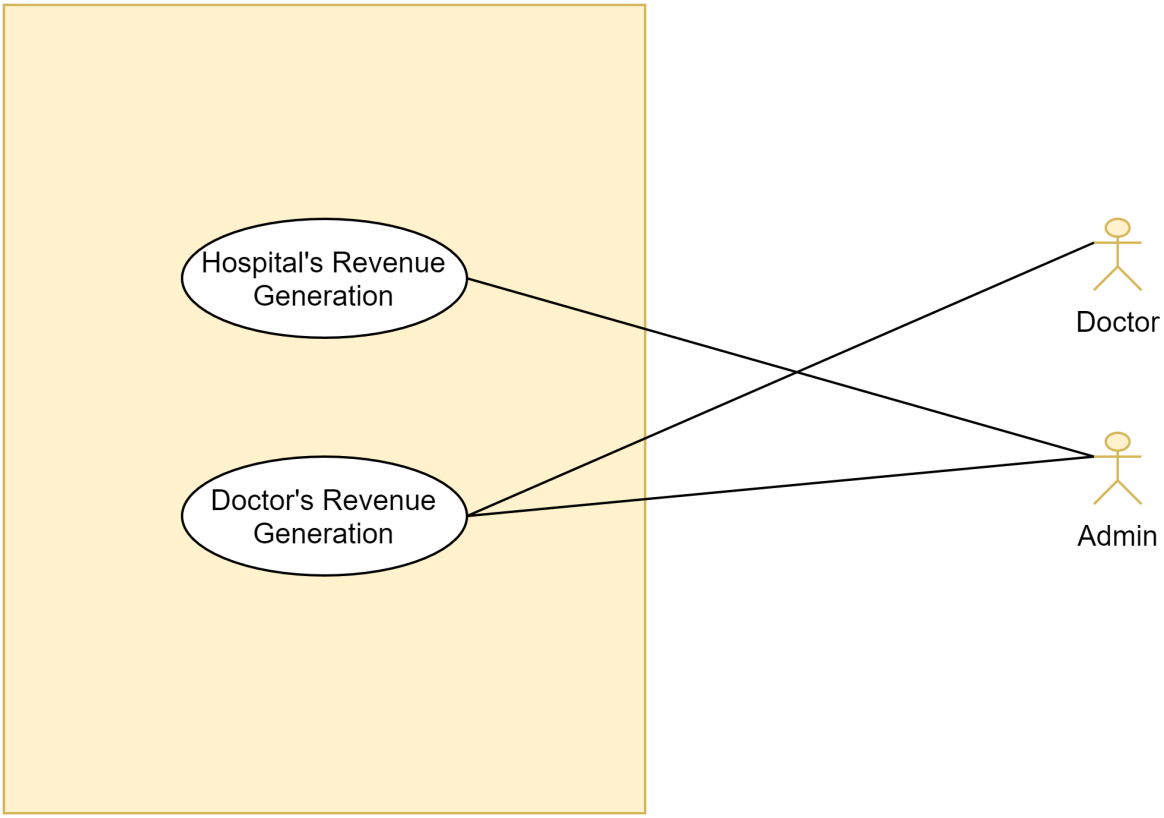


Level 1.4 - Payment

Level 1.4.1

Name : Revenue generation

Secondary actors: doctor, admin



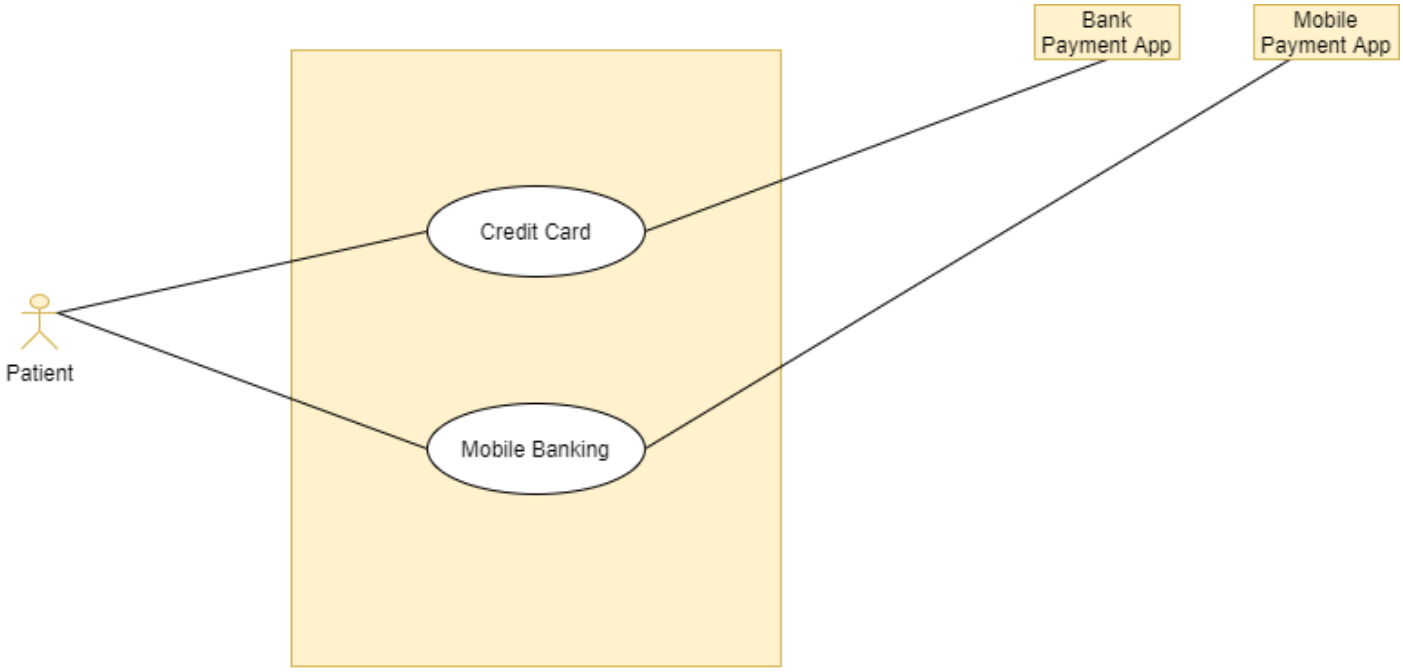
Level 1.4.1 - Revenue Generation

Level 1.4.2

Name: Appointment /medical test payment

Primary actors: patient

Secondary actor: bank payment app, mobile payment app



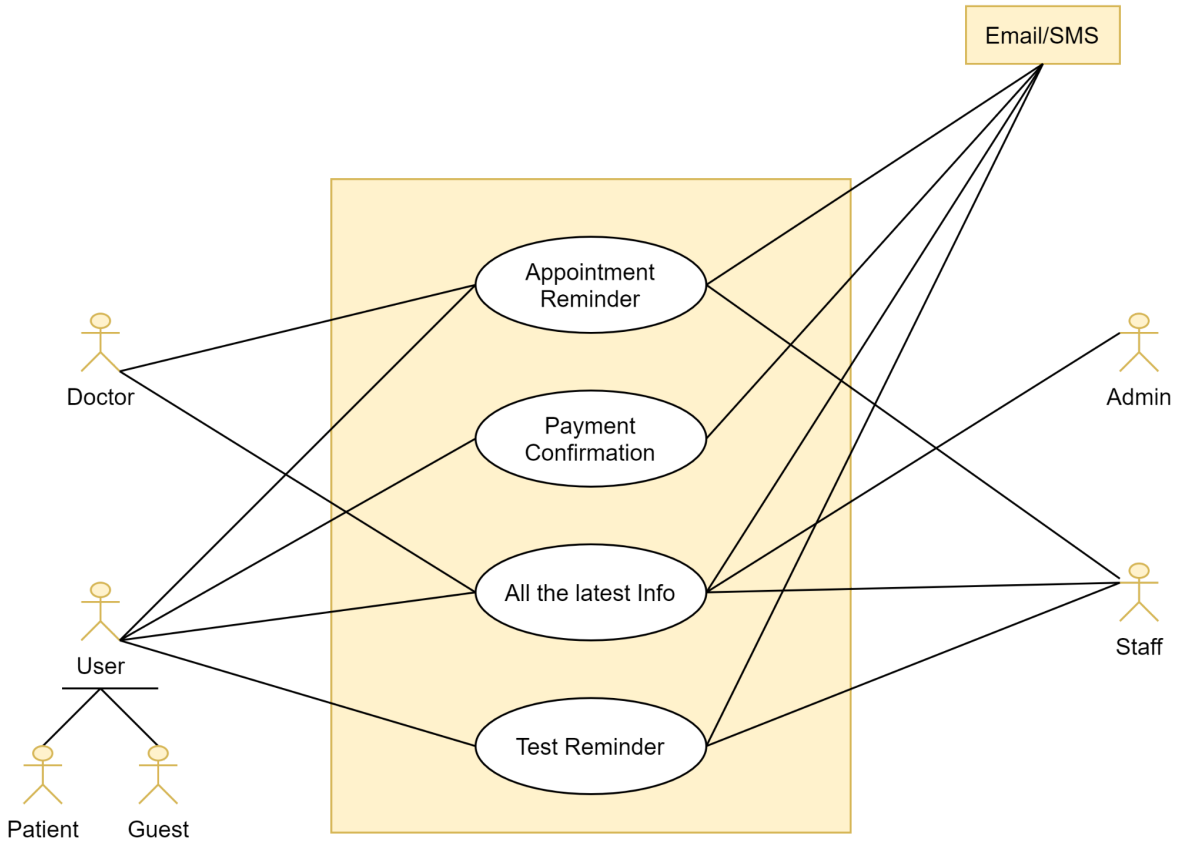
Level 1.4.2 - Appointment/Medical Test Payment

Level-1.5

Name : Notification

Primary Actors : Doctor, user(patient, guest)

Secondary Actors : Admin, Staff, Email/SMS



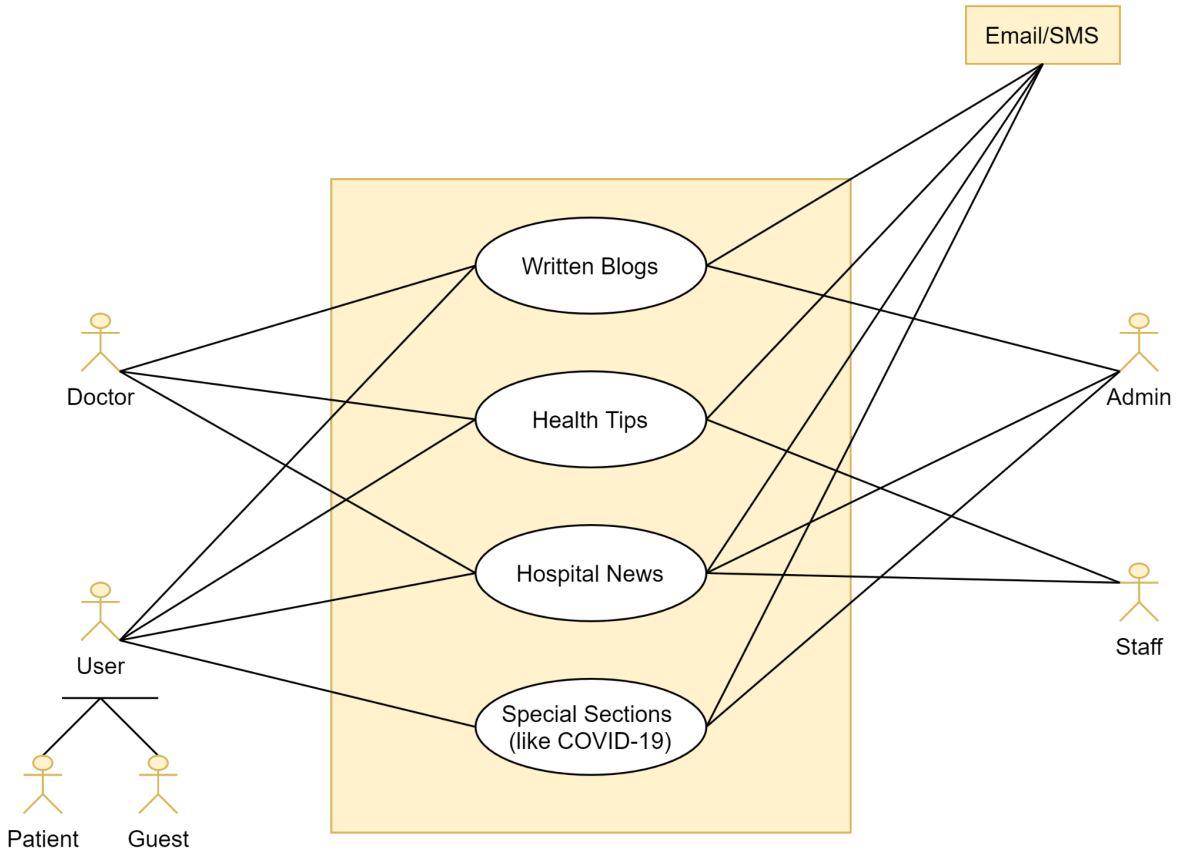
Level 1.5 - Notification

Level-1.5.3

Name : All the latest news and information

Primary Actors : Doctor,User (patient, guest)

Secondary Actors : Admin, Staff, Email/SMS



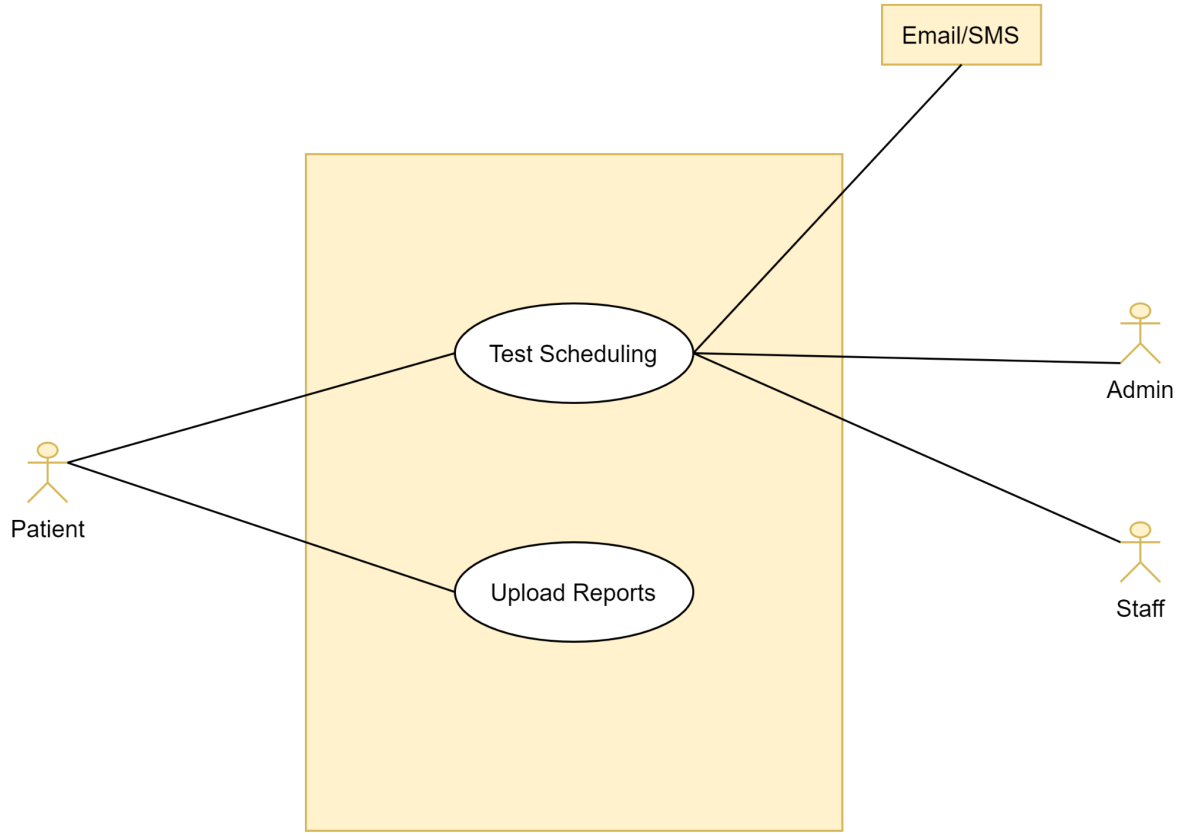
Level 1.5.3 - All the Latest

Level 1.6

Name : Diagnosis

Primary actors: Patient

Secondary actors : admin, staff, Email/SMS



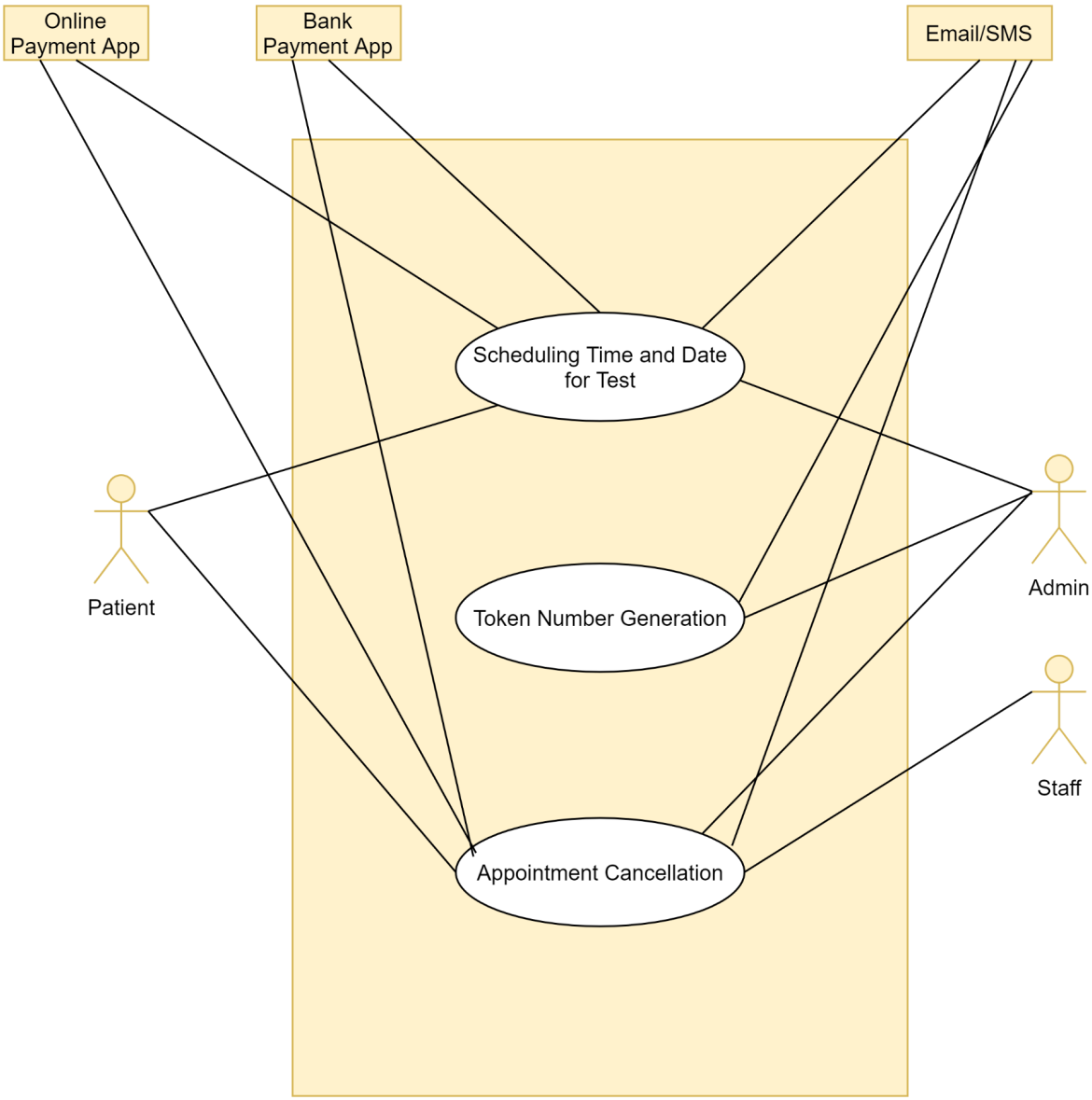
Level 1.6 - Diagnosis

Level-1.6.1

Name : Test Scheduling

Primary Actor : Patient

Secondary Actors : Admin, Staff, Email/SMS, Mobile Payment App, Bank Payment App



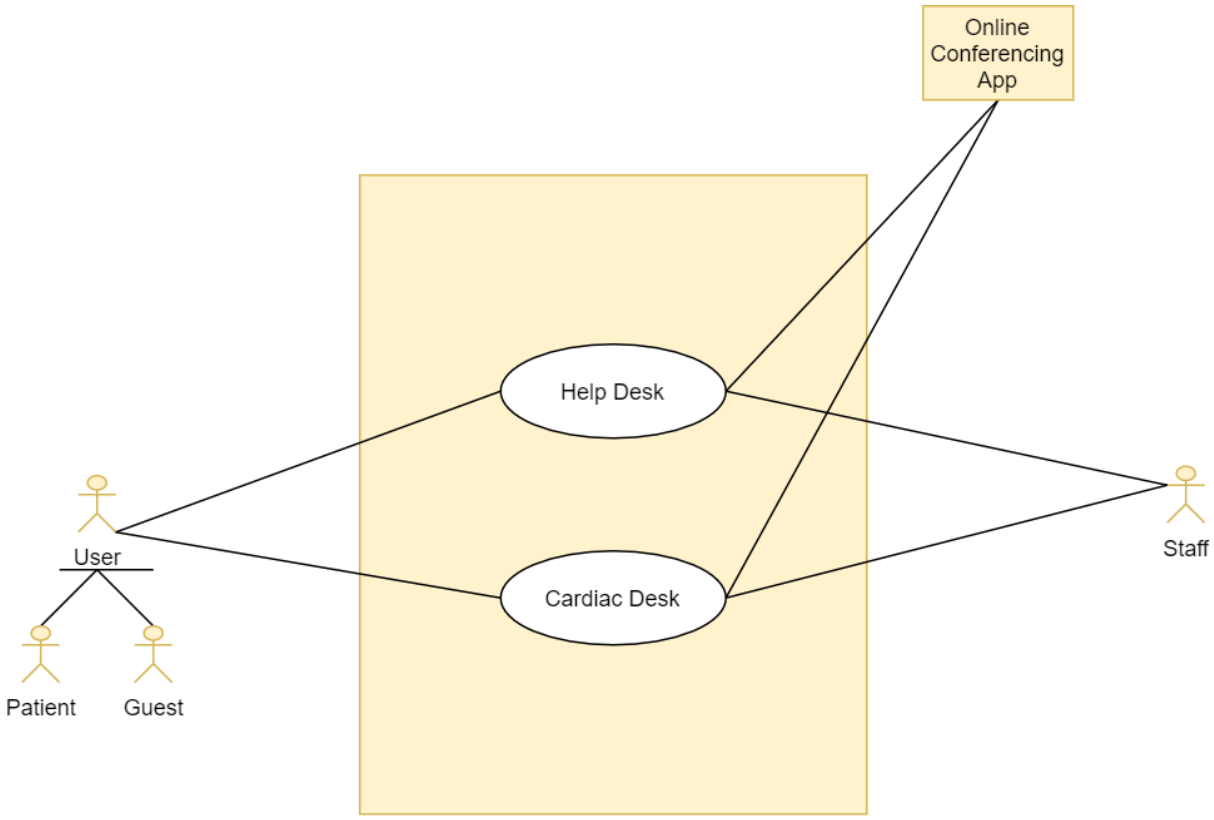
Level 1.6.1 - Test Scheduling

Level-1.7

Name : Online Desk Services

Primary Actors : user(patient, guest)

Secondary Actors : Staff, Online Conferencing App



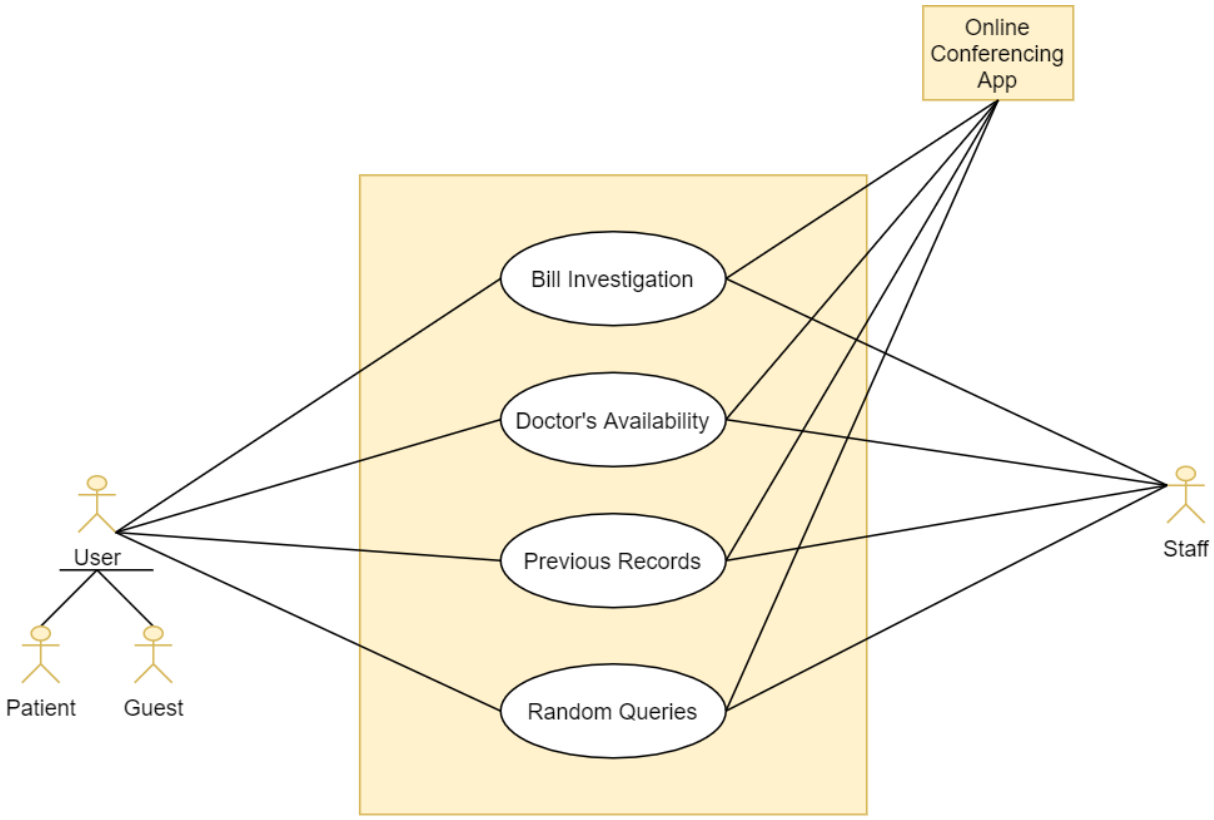
Level 1.7 - Online Desk Services

Level-1.7.1

Name : Help Desk

Primary Actors : User (Guest, Patient)

Secondary Actors : Staff, Online Conferencing App



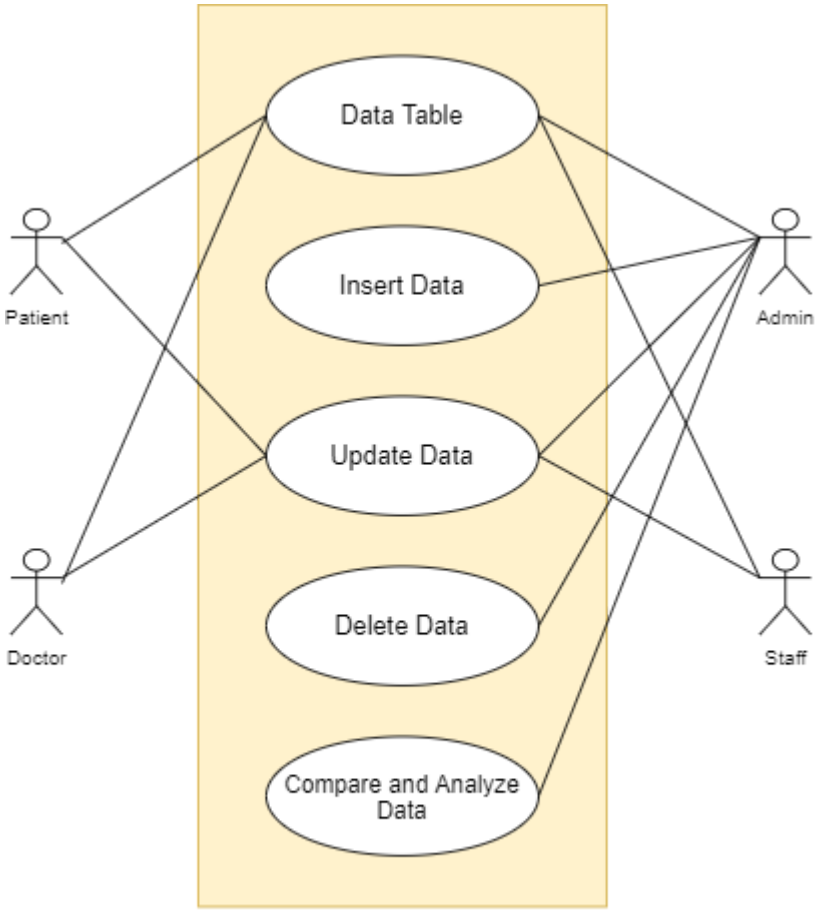
Level 1.7.1 - Help Desk

Level 1.8

Name: Database

Primary actors: Doctor , Patient

Secondary actor: Admin, Staff



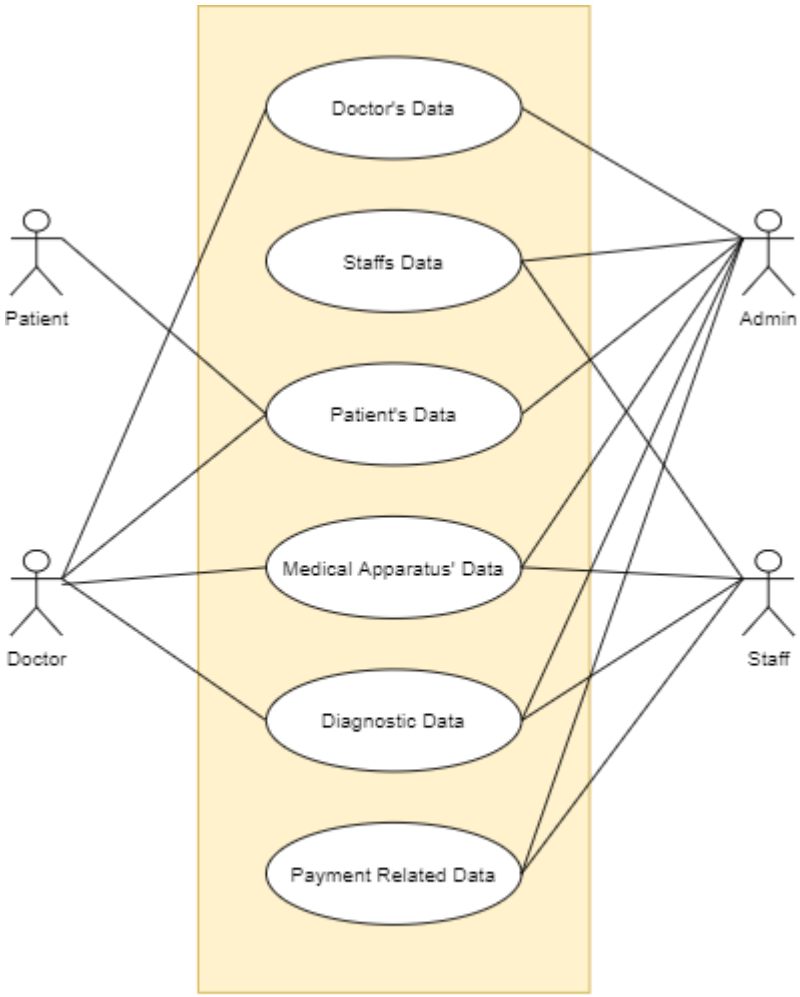
Level 1.8 - Database

Level 1.8.1

Name: Data Tables

Primary actors: Doctor , Patient

Secondary actor: Admin, Staff



Level 1.8.1 - Data Tables

Activity Diagram : Hospital Management System

Activity Diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The basic purpose of activity diagrams is to capture the dynamic behavior of the system.

It focuses on the execution and flow of the behavior of a system instead of implementation. Activity diagrams consist of activities that are made up of actions that apply to behavioral modeling technology.

An activity can be attached to any modeling element to model its behavior. Activity diagrams are used to model-

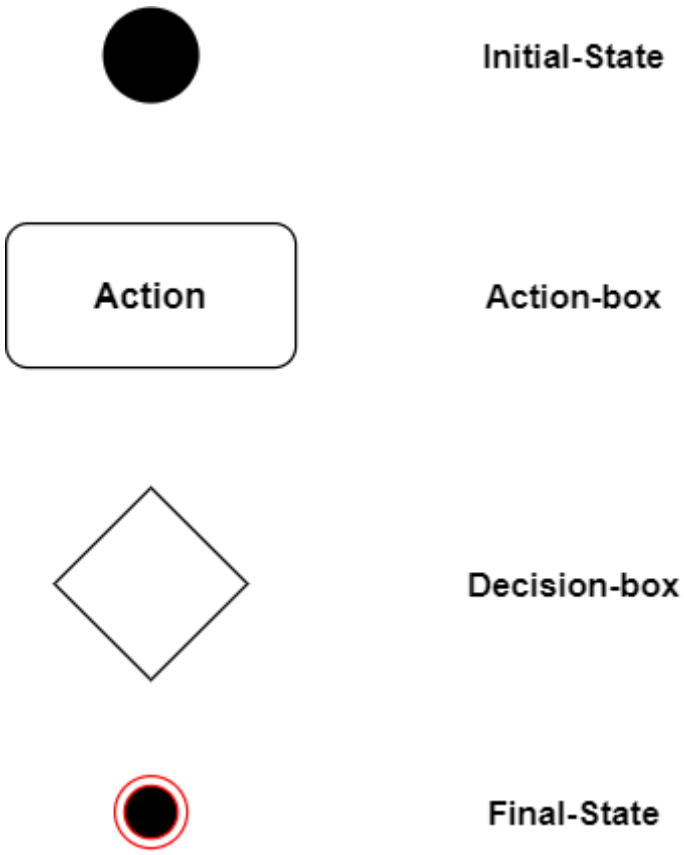
- Use cases
- Classes
- Interfaces
- Components etc.

Following rules must be followed while developing an activity diagram-

- All activities in the system should be named.
- Activity names should be meaningful.
- Constraints must be identified.
- Activity associations must be known.

To draw an activity diagram, one must understand and explore the entire system. All the elements and entities that are going to be used inside the diagram must be known by the user. The central concept which is nothing but an activity must be clear to the user. After analyzing all activities, these activities should be explored to find various constraints that are applied to activities. If there is such a constraint, then it should be noted before developing an activity diagram.

All the activities, conditions, and associations must be known. Once all the necessary things are gathered, then an abstract or a prototype is generated, which is later converted into the actual diagram.



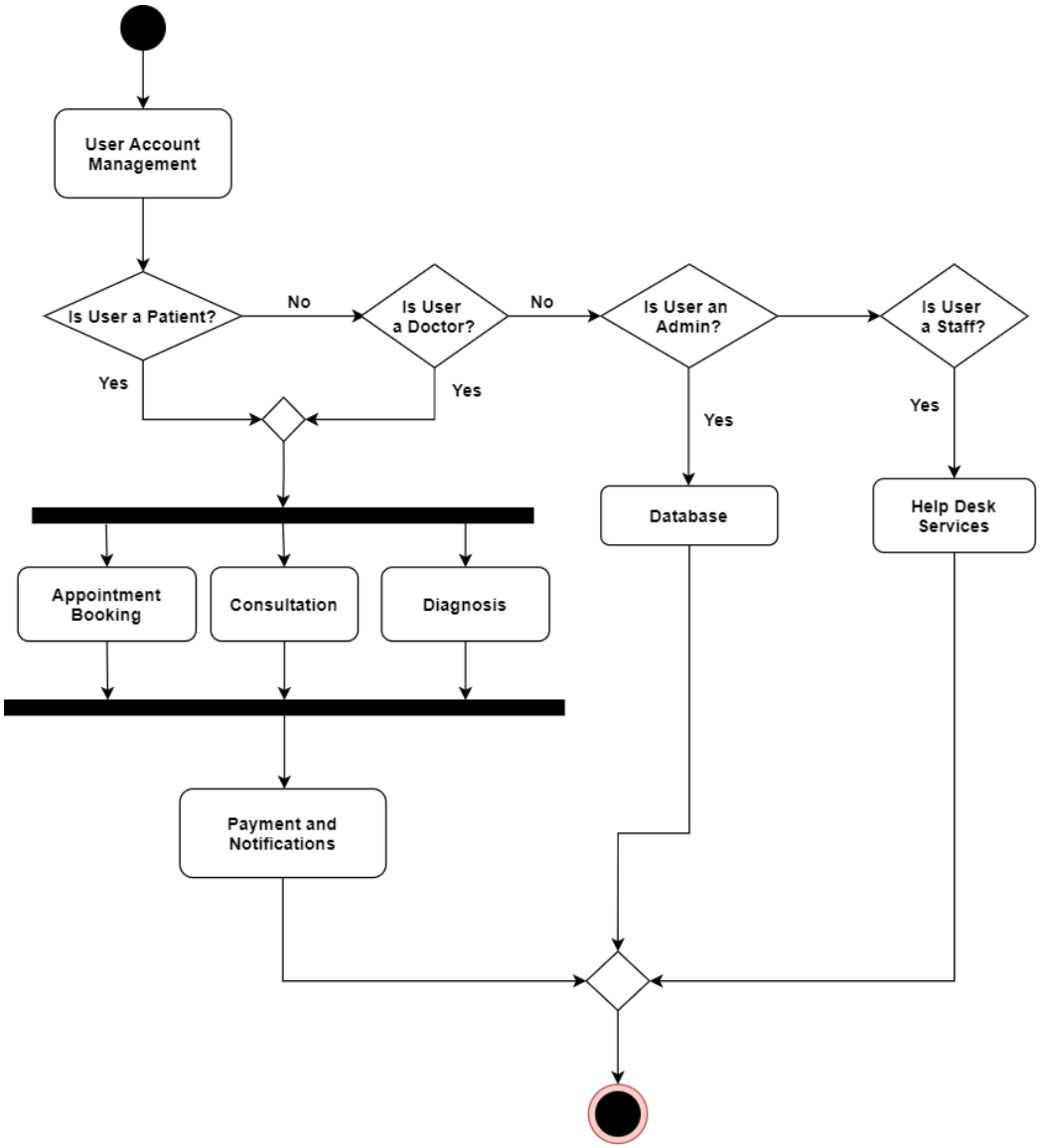
Activity Diagram Notation and Symbol

Now we have to consider our SRS project- Hospital Management System as the sample. We will use our use case diagrams as reference to draw activity diagrams to show the whole flow of the system.

Level-1

Name : Hospital Management System

Reference : Use Case Level 1

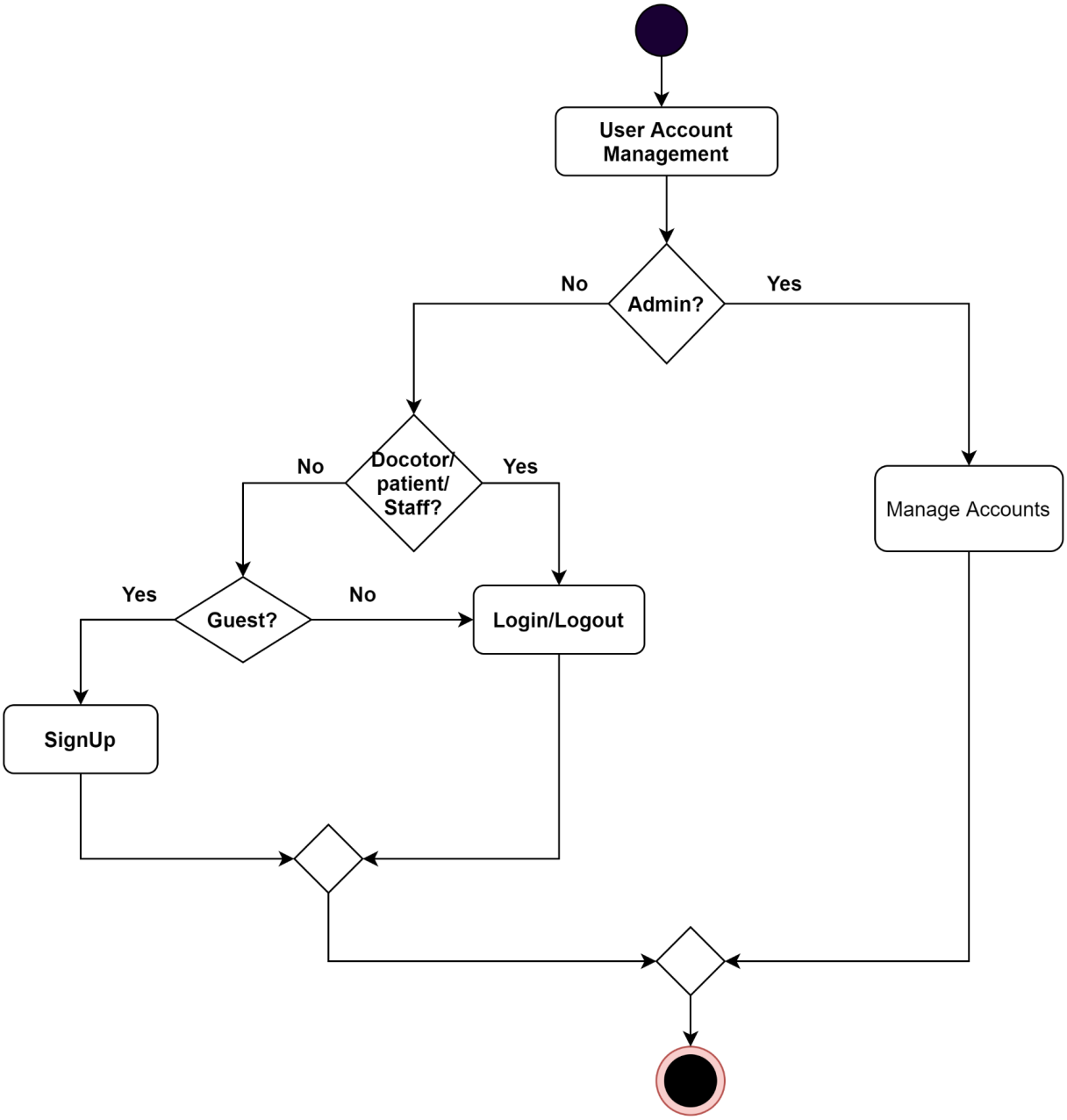


Level 1- Hospital Management System

Level-1.1

Name : User Account Management

Reference : Use Case Level 1.1

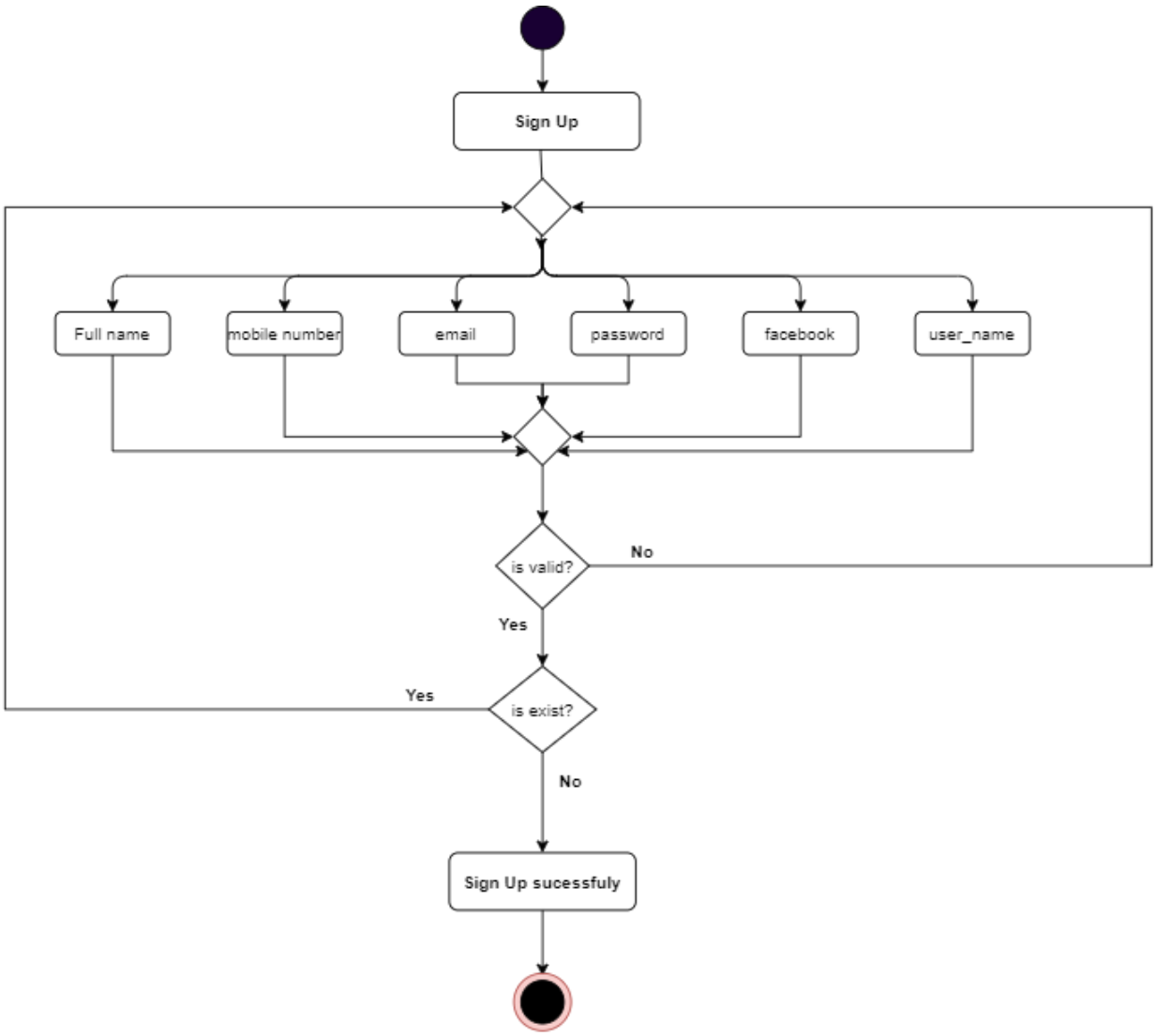


Level 1.1: User Account Management

Level-1.1.2

Name : Sign Up

Reference : Use Case Level 1.1.2

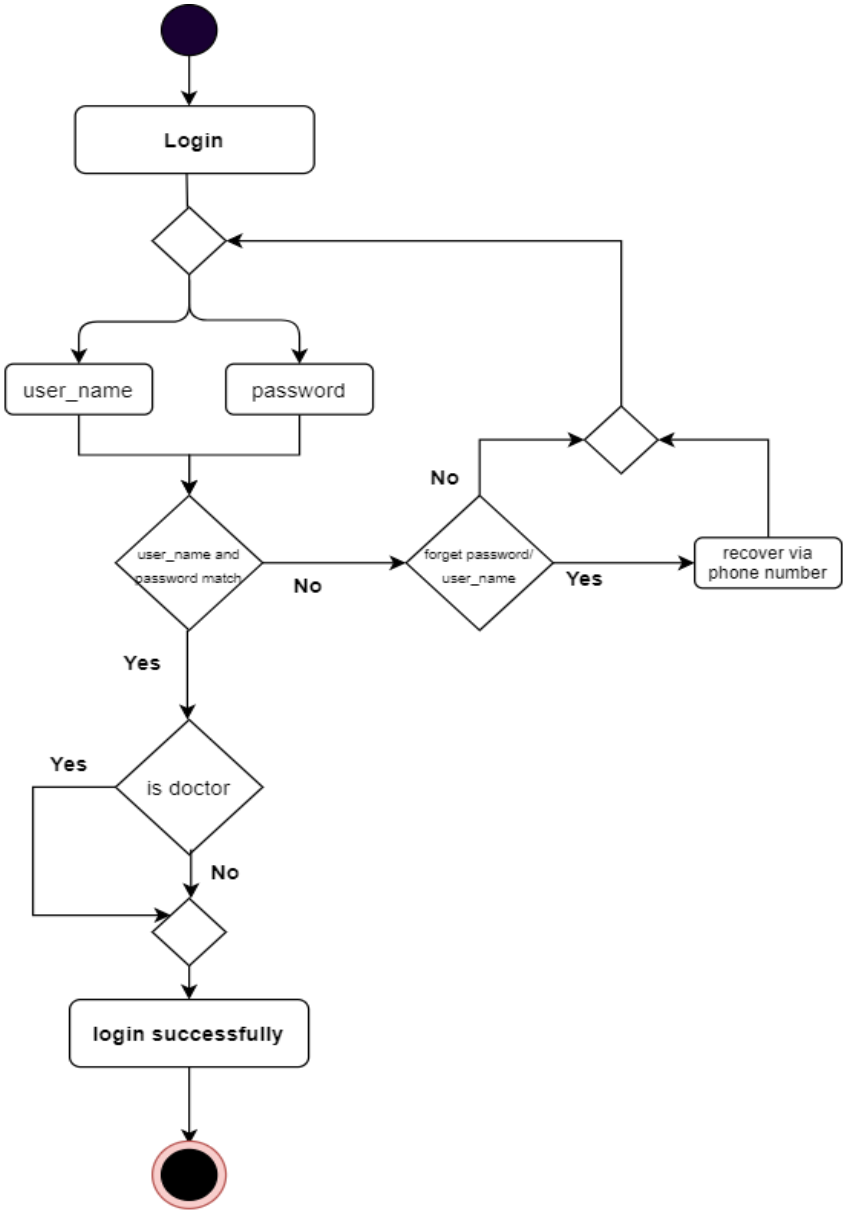


Level 1.1.2: Sign Up

Level-1.1.3

Name : Account Recovery

Reference : Use Case Level 1.1.3

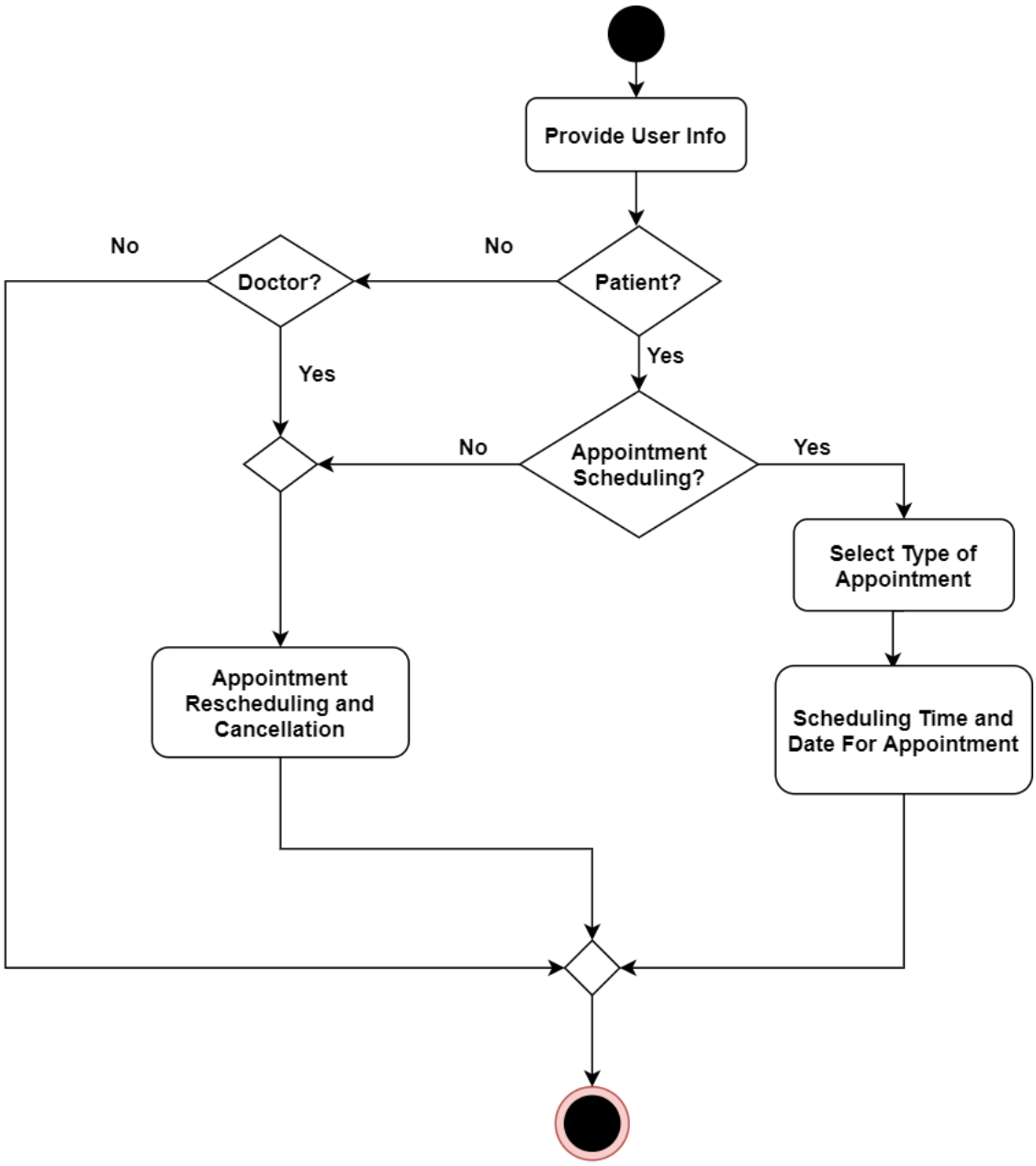


Level 1.1.3: Login

Level-1.2

Name : Appointment Manager

Reference : Use Case Level 1.2

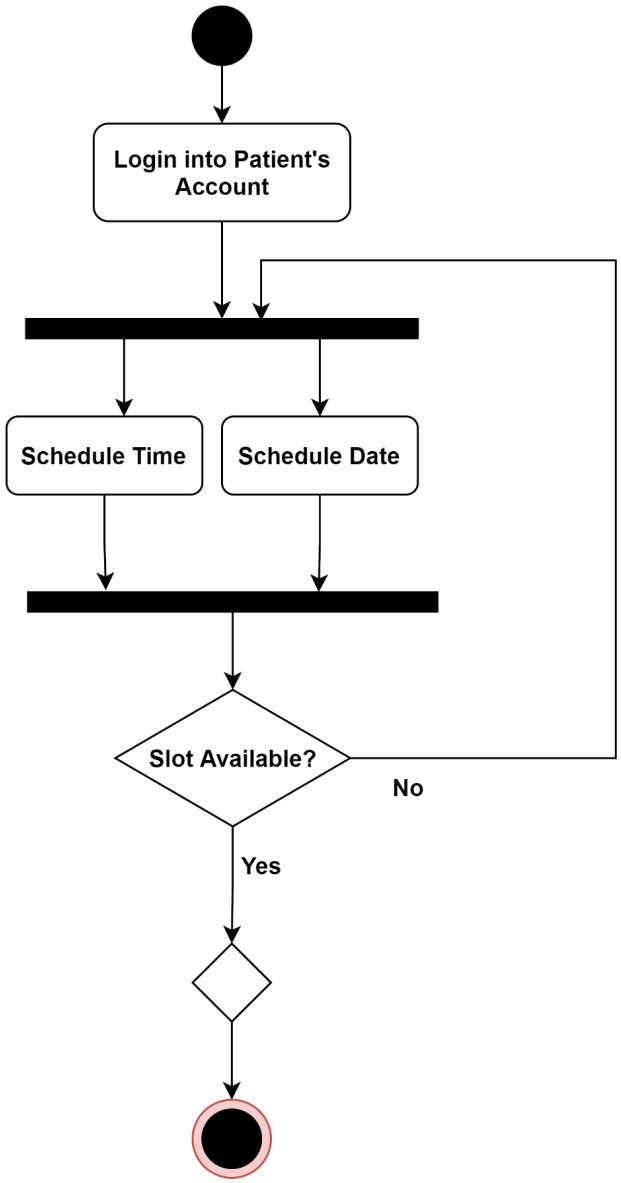


Level 1.2 - Appointment Manager

Level-1.2.1

Name : Scheduling time and date for appointment

Reference : Use Case Level 1.2.1

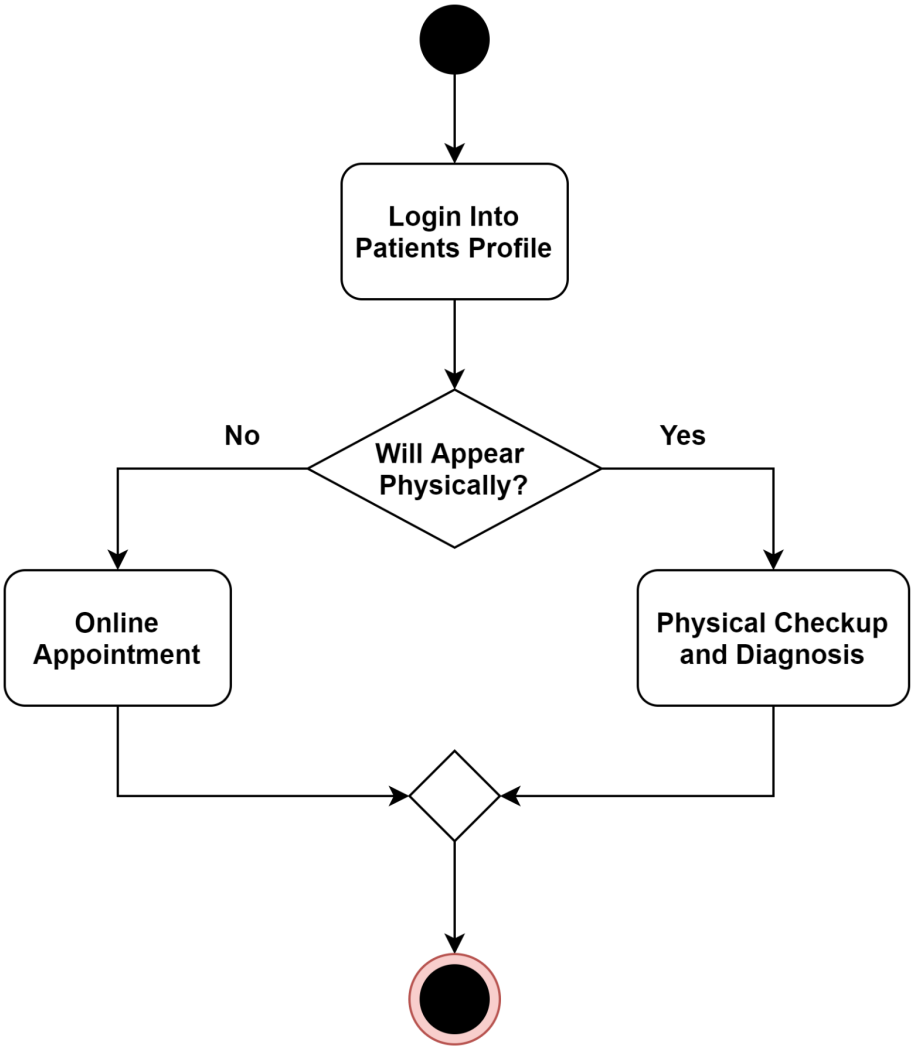


Level 1.2.1- Scheduling Time and Date for Appointment

Level-1.2.2

Name : Types of Appointment

Reference : Use Case Level 1.2.2

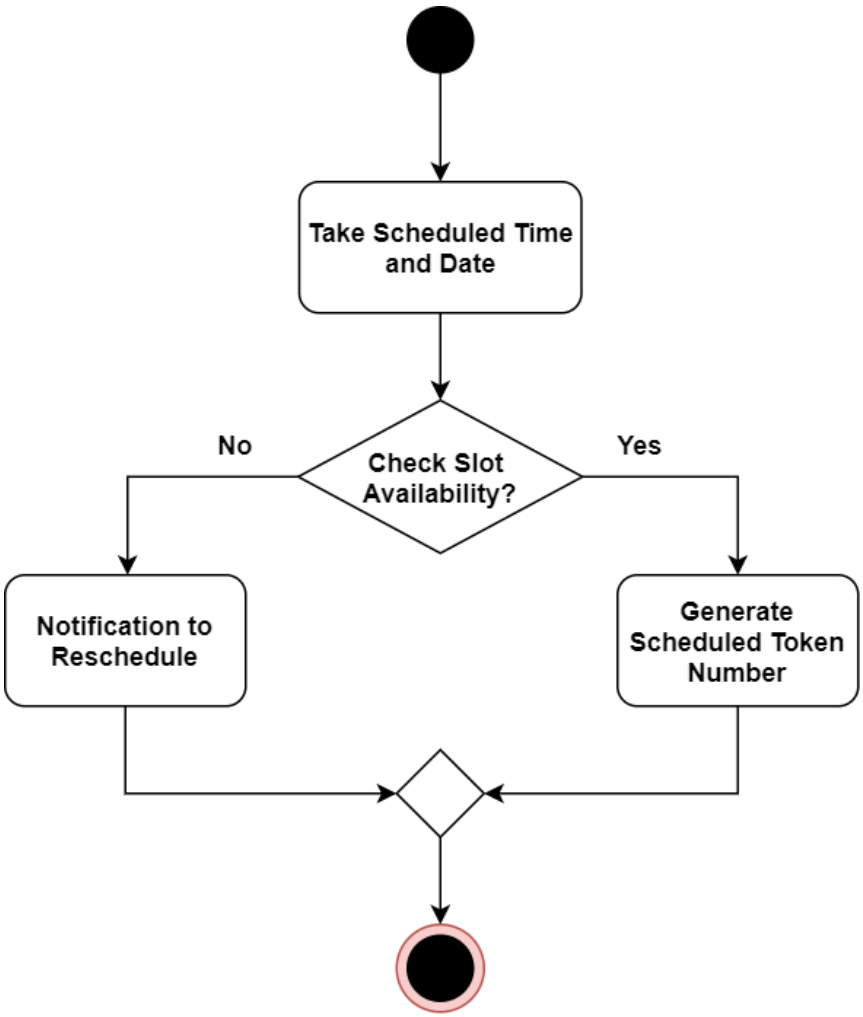


Level 1.2.2- Type of Appointment

Level-1.2.3

Name : Token Number Generation

Reference : Use Case Level 1.2.3

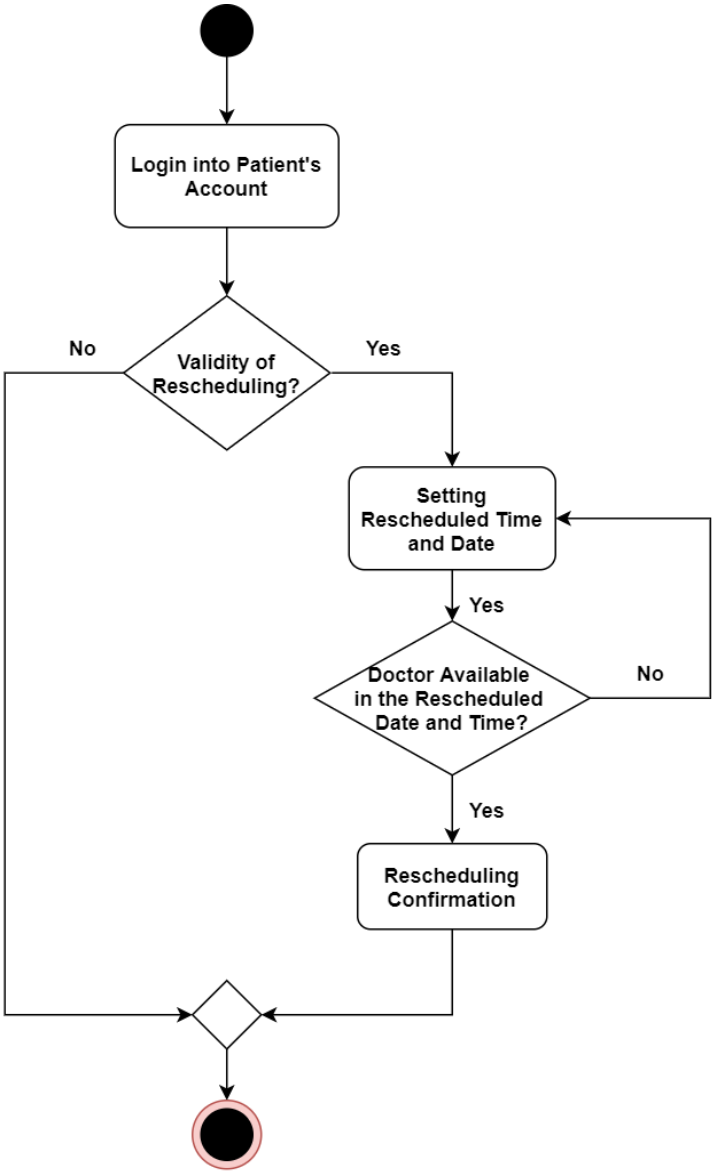


Level 1.2.3- Token Number Generation

Level-1.2.4

Name : Rescheduling Appointment

Reference : Use Case Level 1.2.4

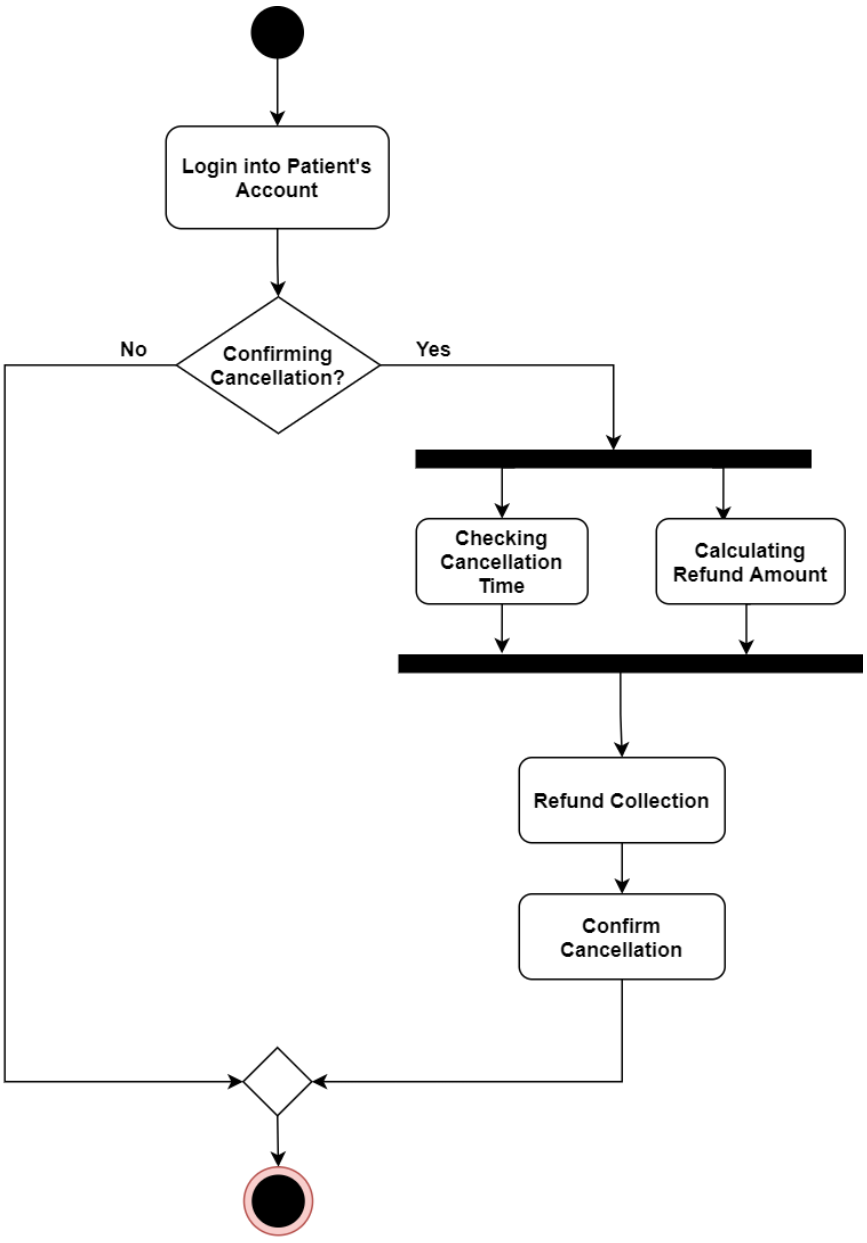


Level 1.2.4- Rescheduling Appointment

Level-1.2.5

Name : Appointment Cancellation

Reference : Use Case Level 1.2.5

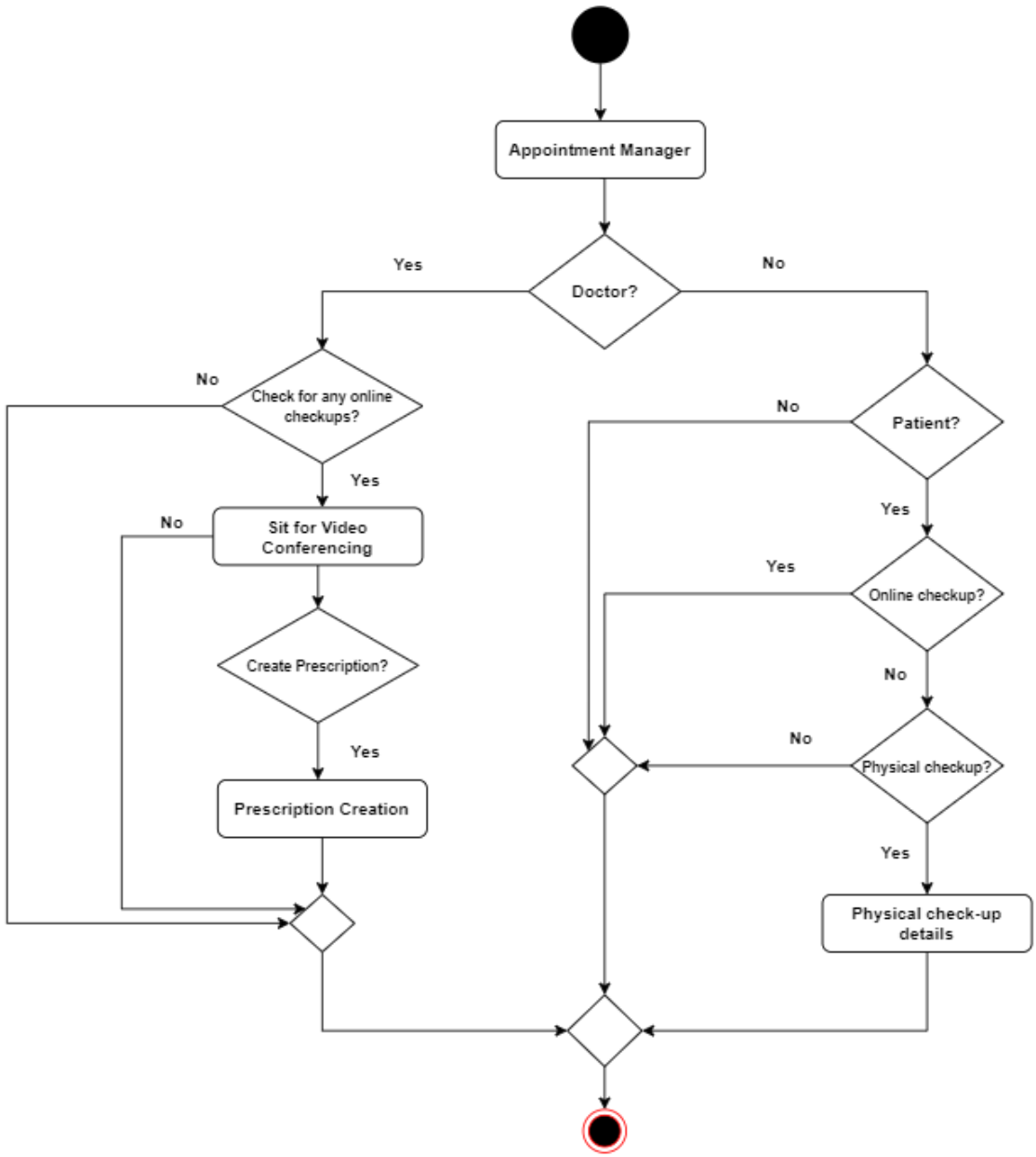


Level 1.2.5- Appointment Cancellation

Level-1.3

Name : Consultation

Reference : Use Case Level 1.3

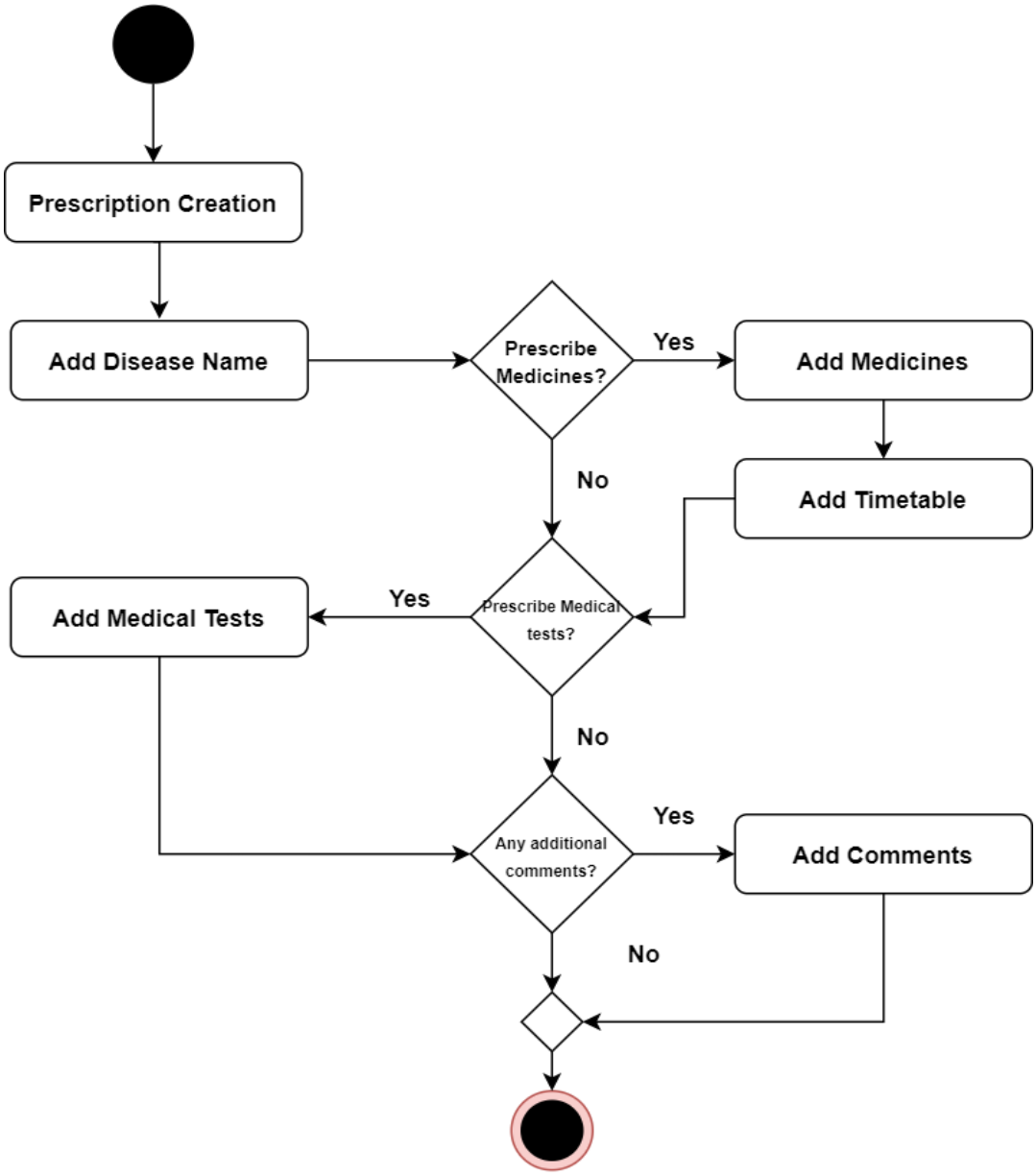


Level- 1.3 - Consultation

Level-1.3.3

Name : Prescription Creation

Reference : Use Case Level 1.3.3

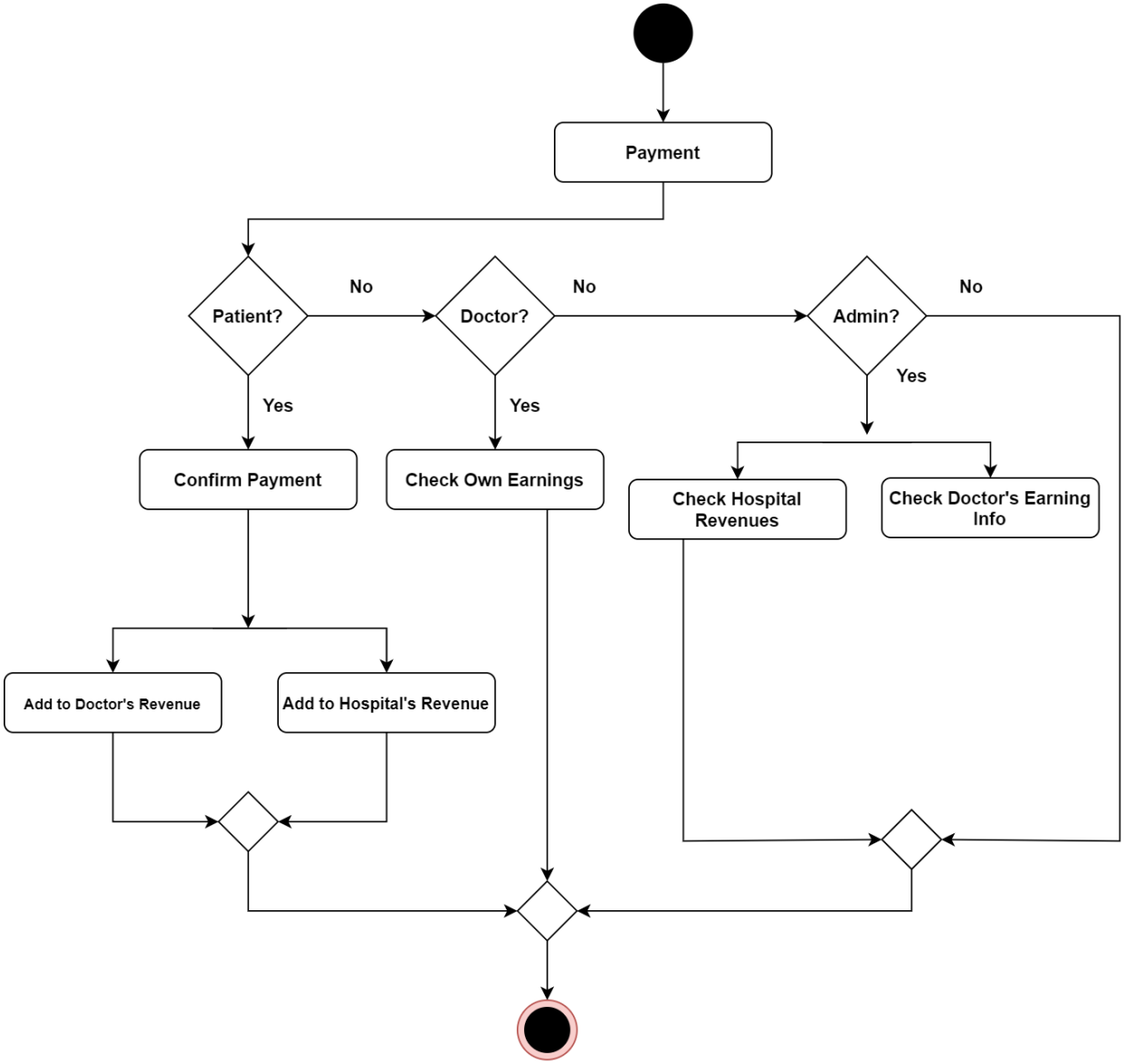


Level 1.3.3- Prescription Creation

Level-1.4

Name : Payment

Reference : Use Case Level 1.4

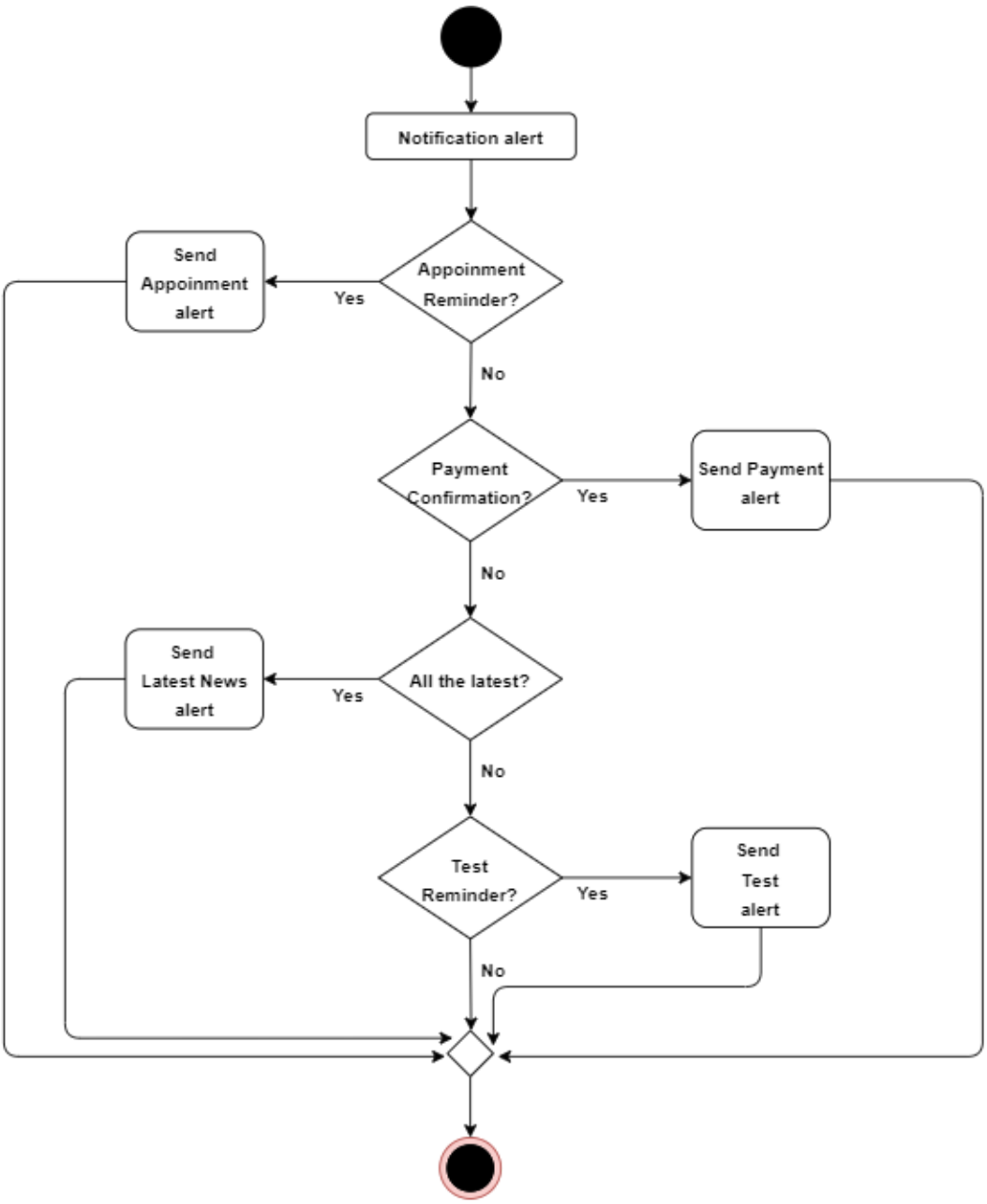


Level 1.4- Payment

Level-1.5

Name : Notification

Reference : Use Case Level 1.5

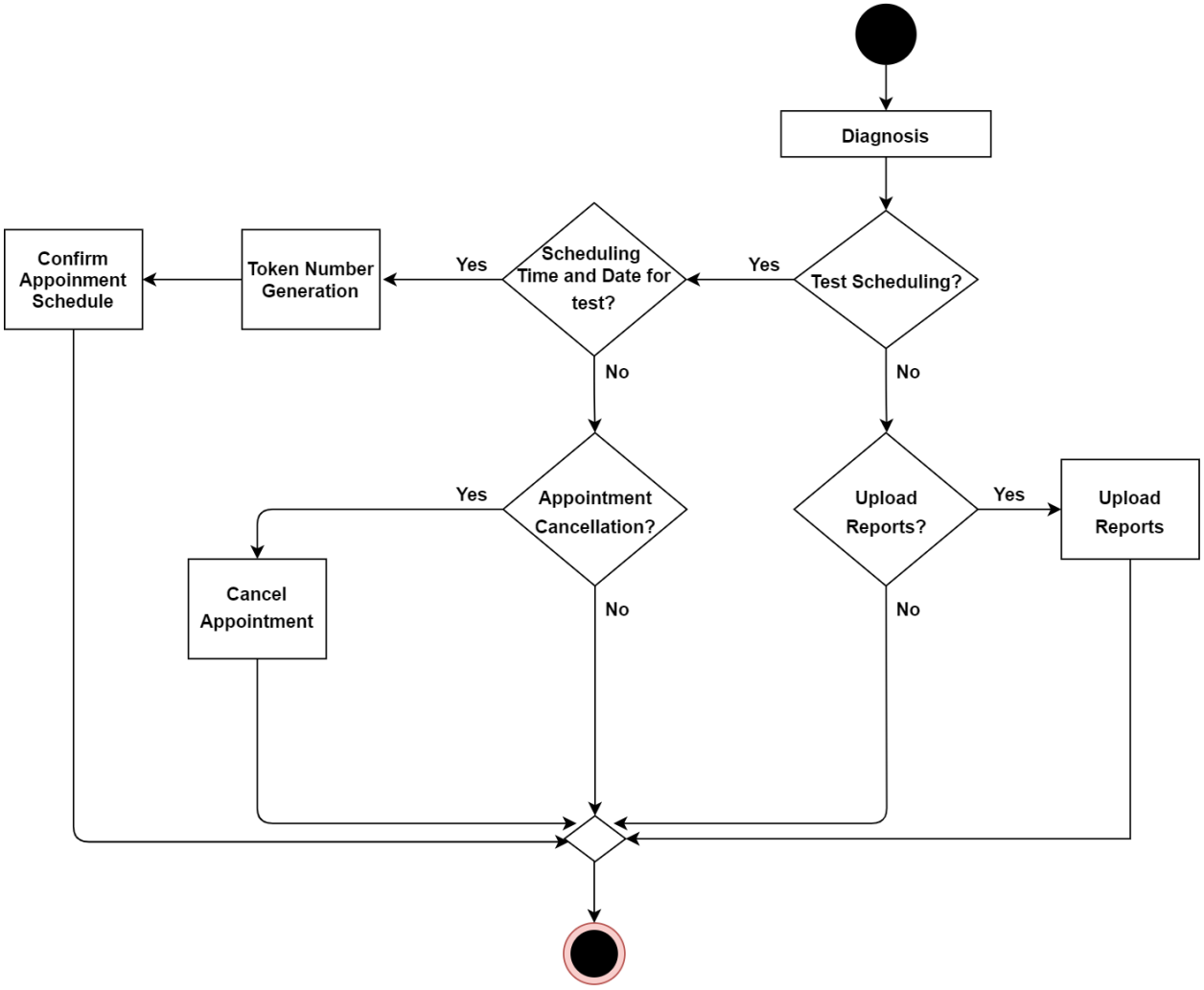


Level 1.5- Notification

Level 1.6

Name : Diagnosis

Reference : Use Case Level 1.6

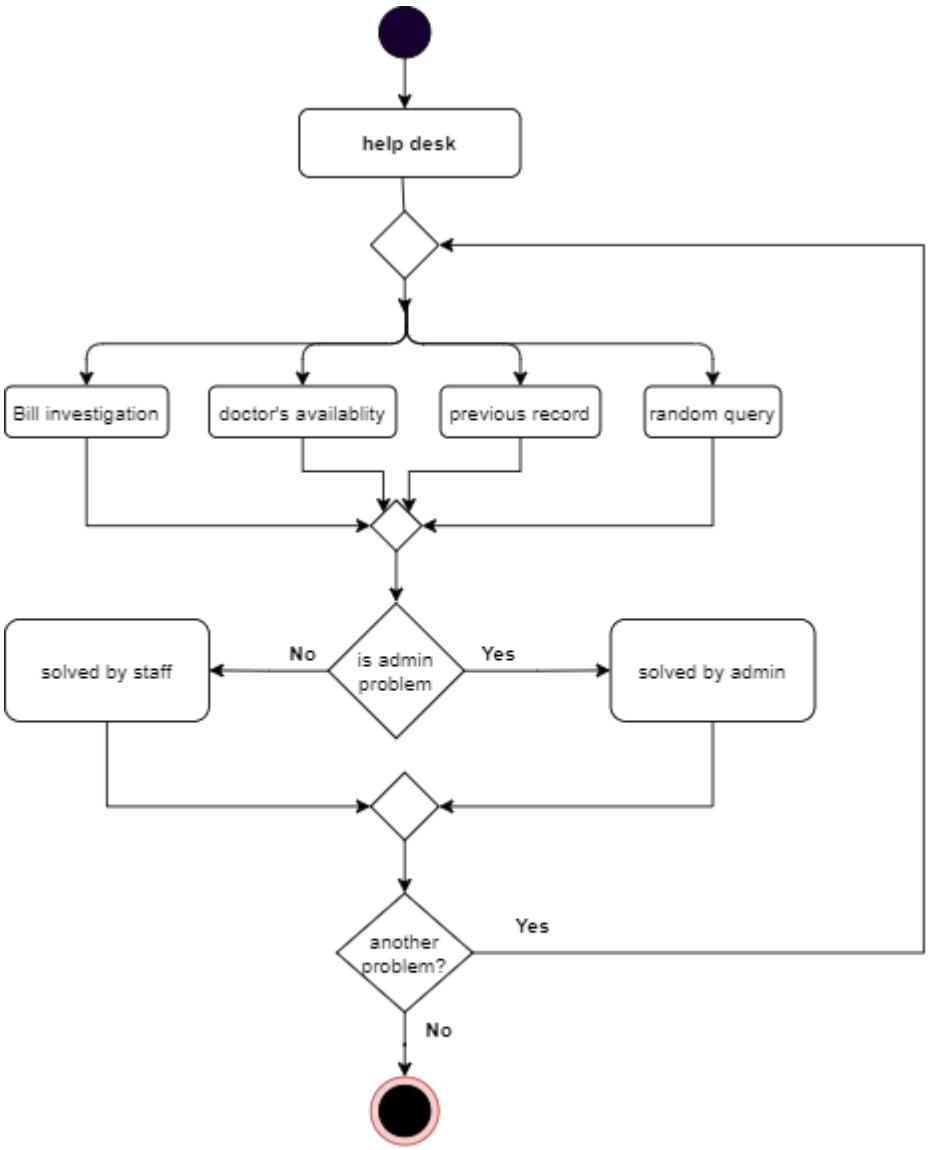


Level 1.6- Diagnosis

Level-1.7

Name : Online Desk Services

Reference : Use Case Level 1.7

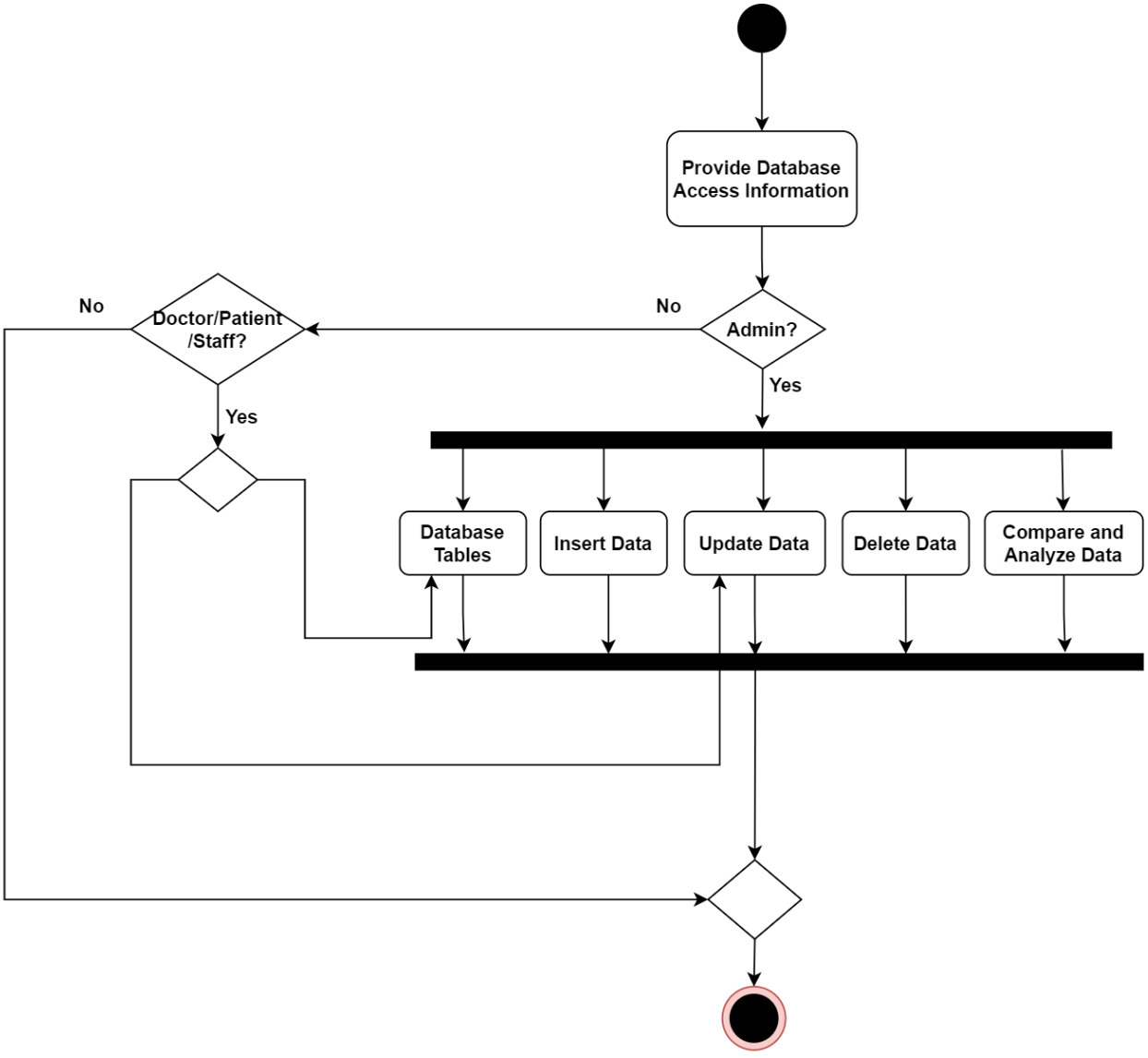


Level 1.7: Help desk

Level 1.8

Name: Database

Reference : Use Case Level 1.8



Level 1.8- Database

Swimlane Diagram: Hospital Management System

A Swimlane diagram is a type of flowchart that outlines who does what in a given process. Based on the analogy of lanes in a pool, a swimlane diagram places process steps within the horizontal or vertical "swimlanes" of a particular department, work group or employee, thus ensuring clarity and accountability. Highlighting connections and communications between these lanes, it can serve as an indicator of waste, redundancy and inefficiency in a process.

Like any other flowchart, it visualizes a process from beginning to end, using the metaphorical lanes of an actual swimming pool to place the steps of mapping the lanes either vertically or horizontally.

A swim lane diagram is typically used for projects that extend over various departments and distinguish channels according to a specific set of objectives. By organizing the responsibilities in various directions, it can clearly distinguish the objective of each department and individuals inside the team.

Swimlanes (also written as "swim lanes") represent a valuable element in process flow diagrams (PFDs), as well as in what's called the Business Process Model and Notation (BPMN) and its software design counterpart – Unified Modeling Language (UML). They introduce parallel (vertical or horizontal) lines to group process steps by actor (which can be a department, work group, employee or even an information system). A swimlane diagram not only spells out processes designated to a specific actor, it also shows how different actors interact to keep a process rolling efficiently.

How to Make a Swimlane Diagram

- Identify the lanes. Then decide what "Actors" are needed to be represented by swimlanes and label them.
- The separation of processes into lanes—either horizontally or vertically—and organizing discrete tasks in sequential order along the other axis.
- Add steps. Each step should be connected to the one before it with a line.

The Purpose of Swimlane Diagram

Providing an easy-to-read representation of responsibilities within a process, a swimlane diagram can serve the following purposes:

- To communicate and highlight which process steps or sub-processes are assigned to a particular actor.

- To identify bottlenecks and other inefficiencies, which in turn helps detect redundancies between various lanes, duplicative steps, process delays or capacity constraints that can be later addressed and resolved. All this leads to increased performance and quality.

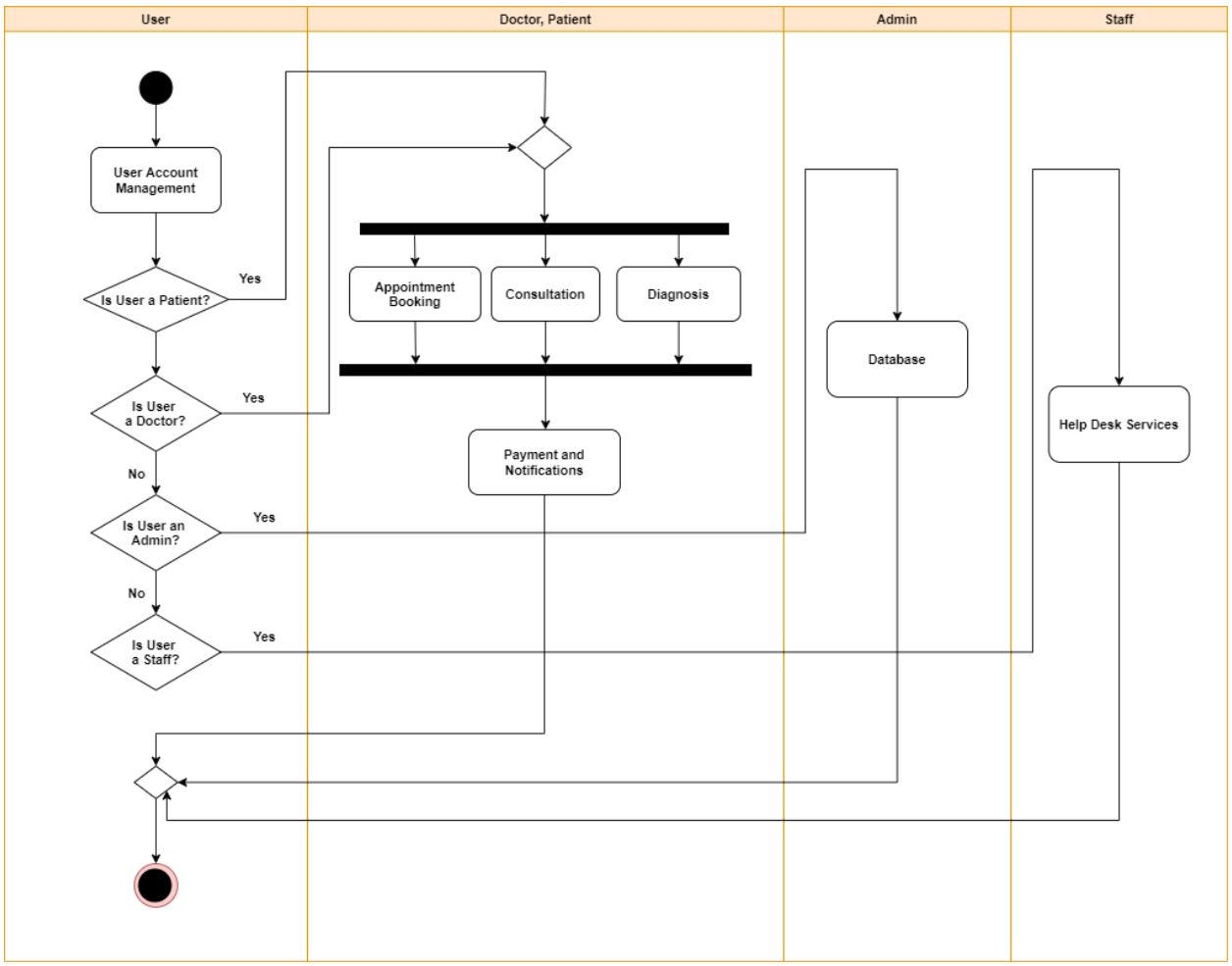
- To better structure a given process and account for evolving circumstances, such as staffing or technology changes.

- To provide a formal model of integrating processes between teams and departments, which results in clearer, more organized workflows on an ongoing basis.

Level-1

Name : Hospital Management System

Reference : Use Case & Activity Diagram- Level 1

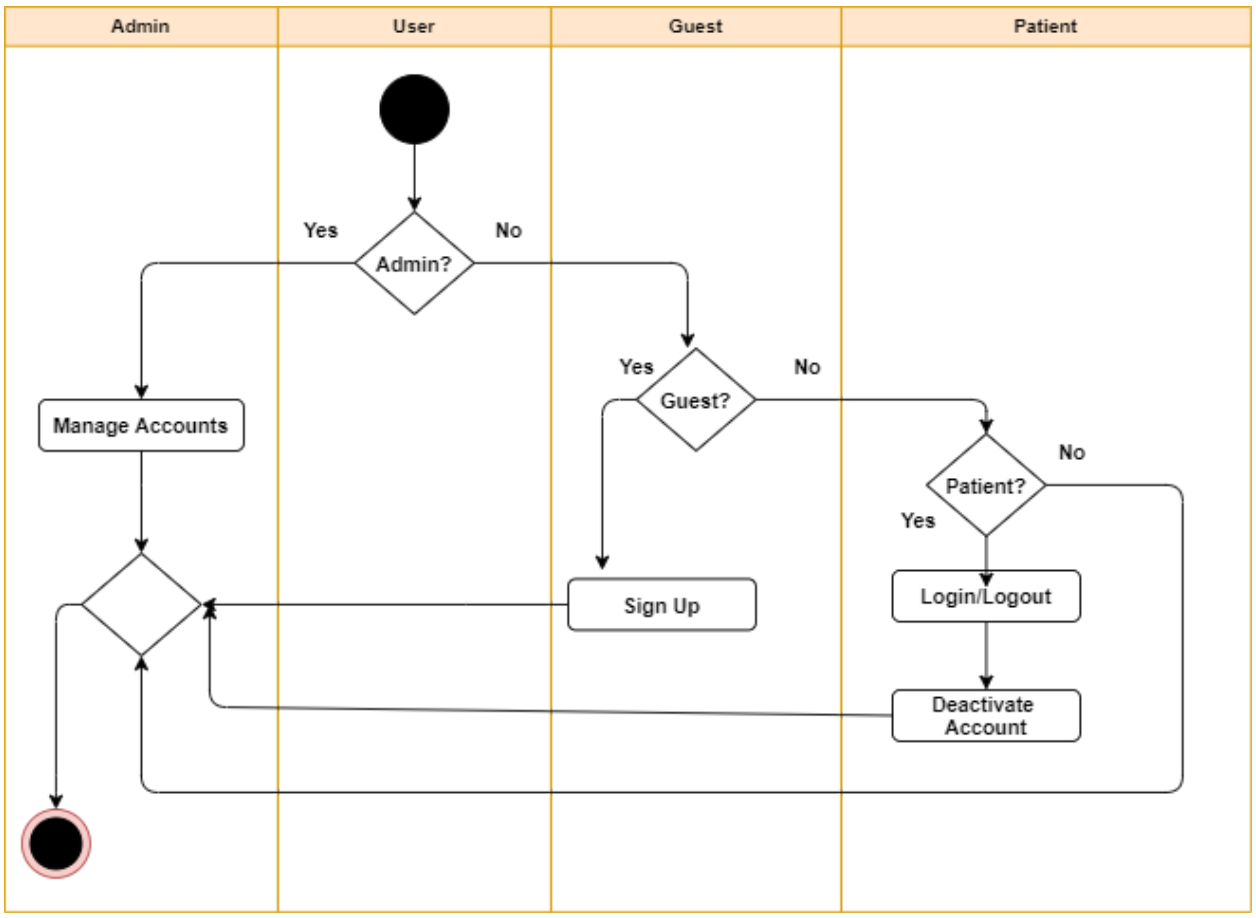


Hospital Management System (Swimlane Diagram Level - 1)

Level-1.1

Name : User Account Management

Reference : Use Case & Activity Diagram- Level 1.1

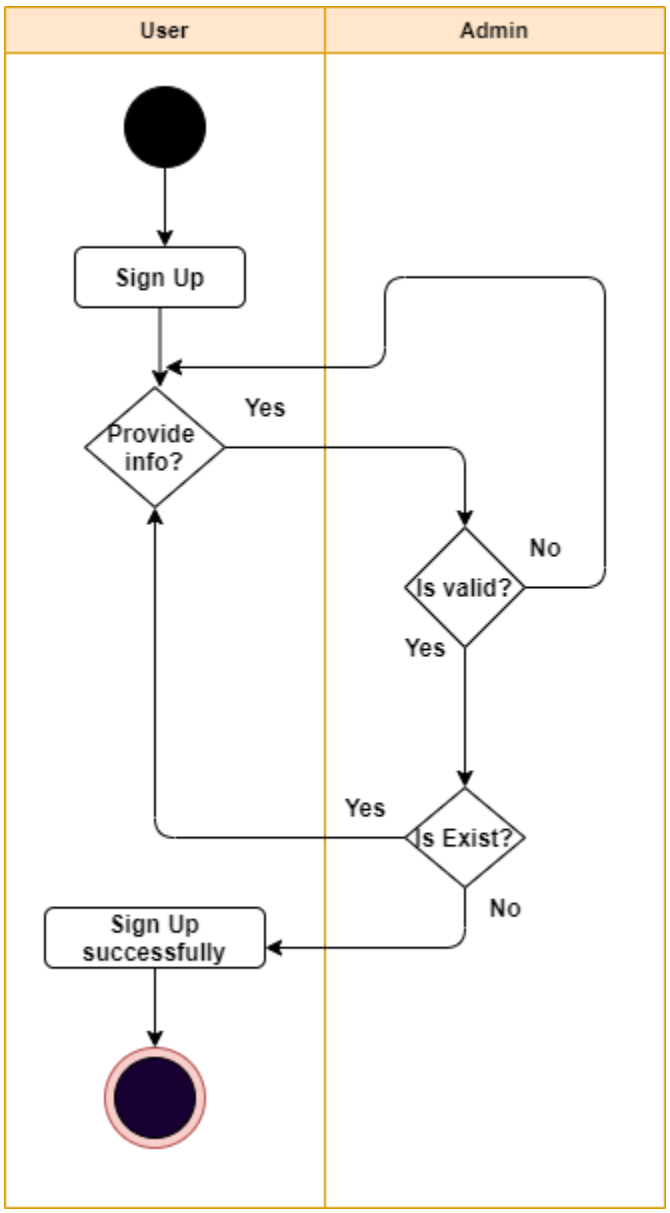


User Account Management (Swimlane Diagram Level - 1.1)

Level-1.1.2

Name : Sign Up

Reference : Use Case & Activity Diagram- Level 1.1.2

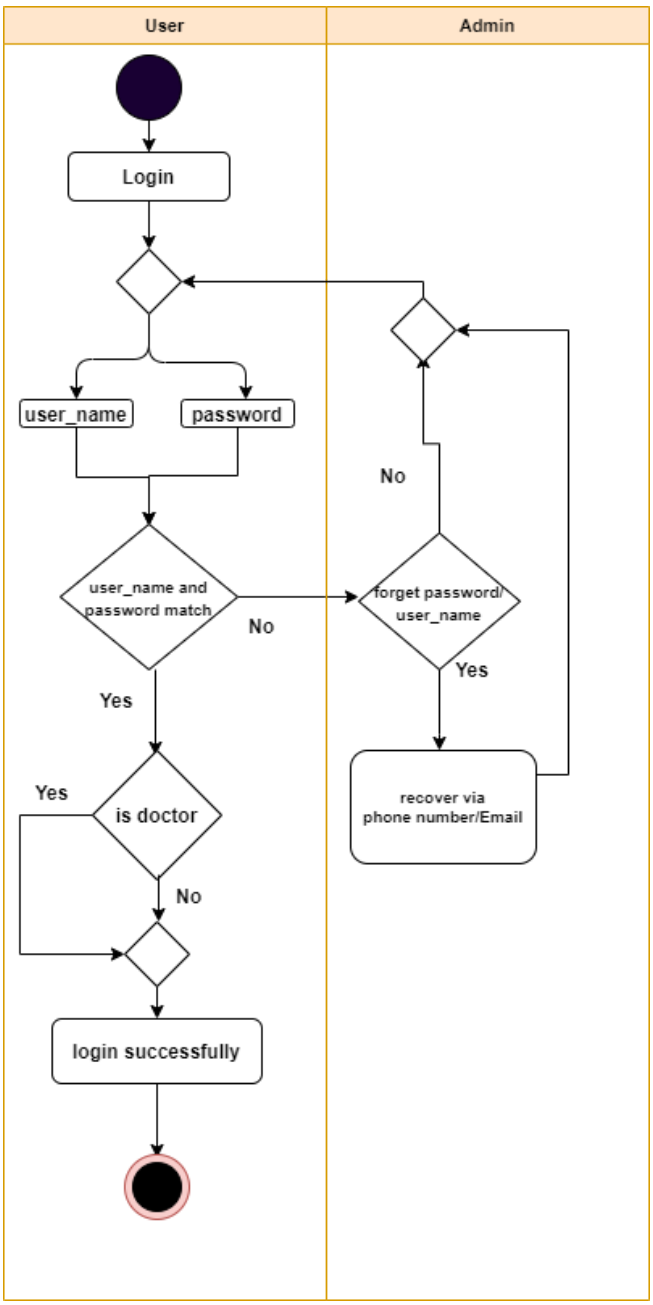


Sign Up (Swimlane Diagram Level - 1.1.2)

Level-1.1.3

Name : Account Recovery

Reference : Use Case & Activity Diagram- Level 1.1.3

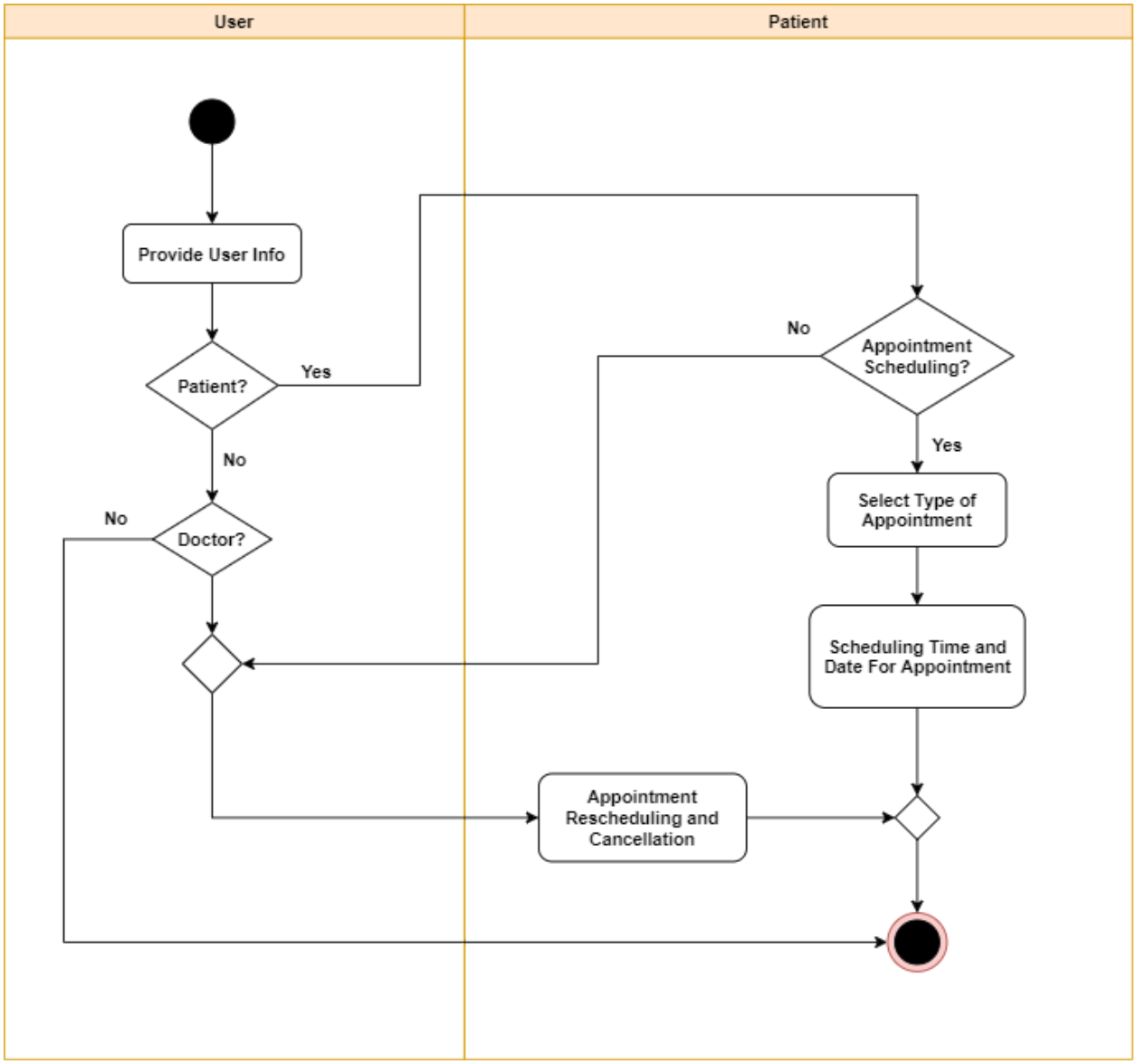


Account Recovery (Swimlane Diagram Level - 1.1.3)

Level-1.2

Name : Appointment Manager

Reference : Use Case & Activity Diagram- Level 1.2

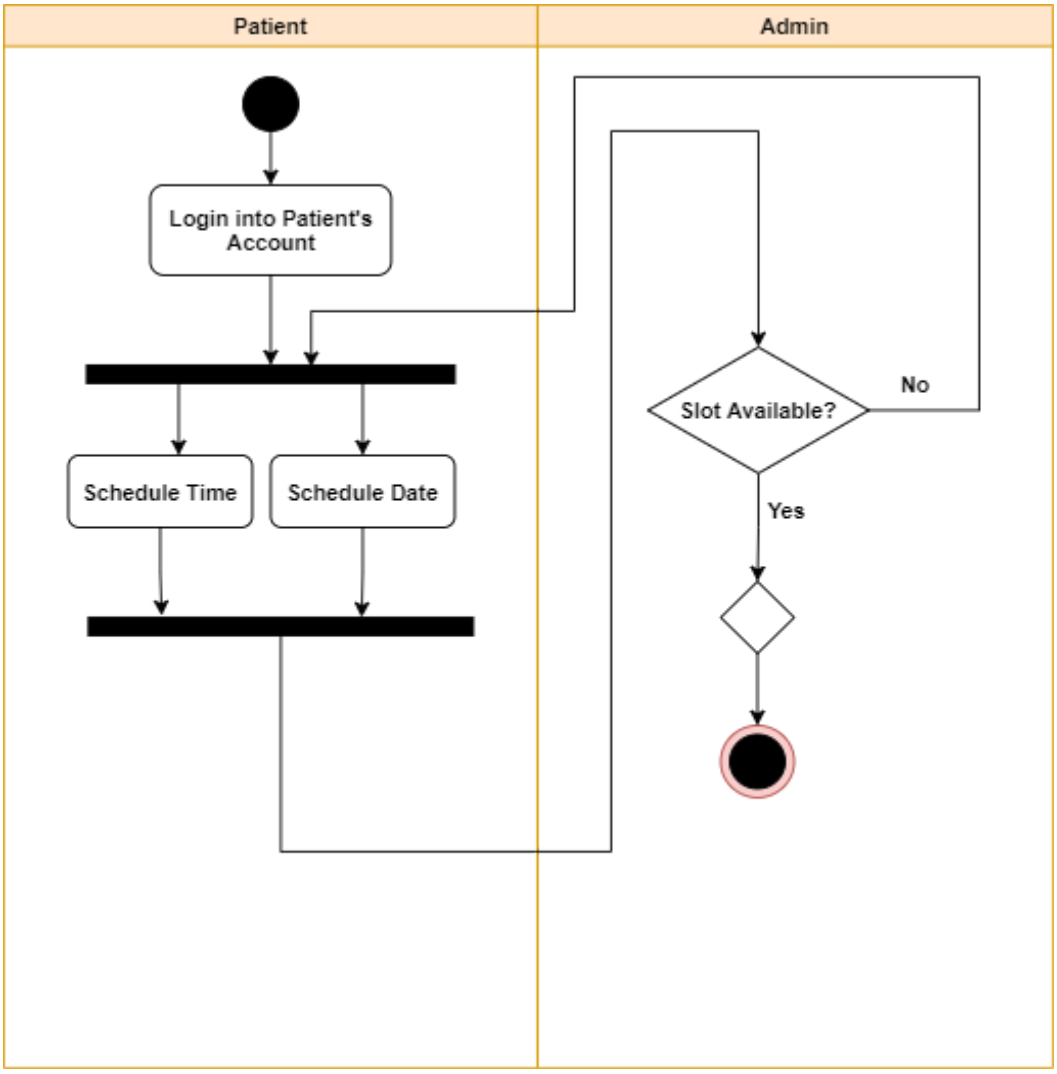


Appointment Manager (Swimlane Diagram Level - 1.2)

Level-1.2.1

Name : Scheduling time and date for appointment

Reference : Use Case & Activity Diagram- Level 1.2.1

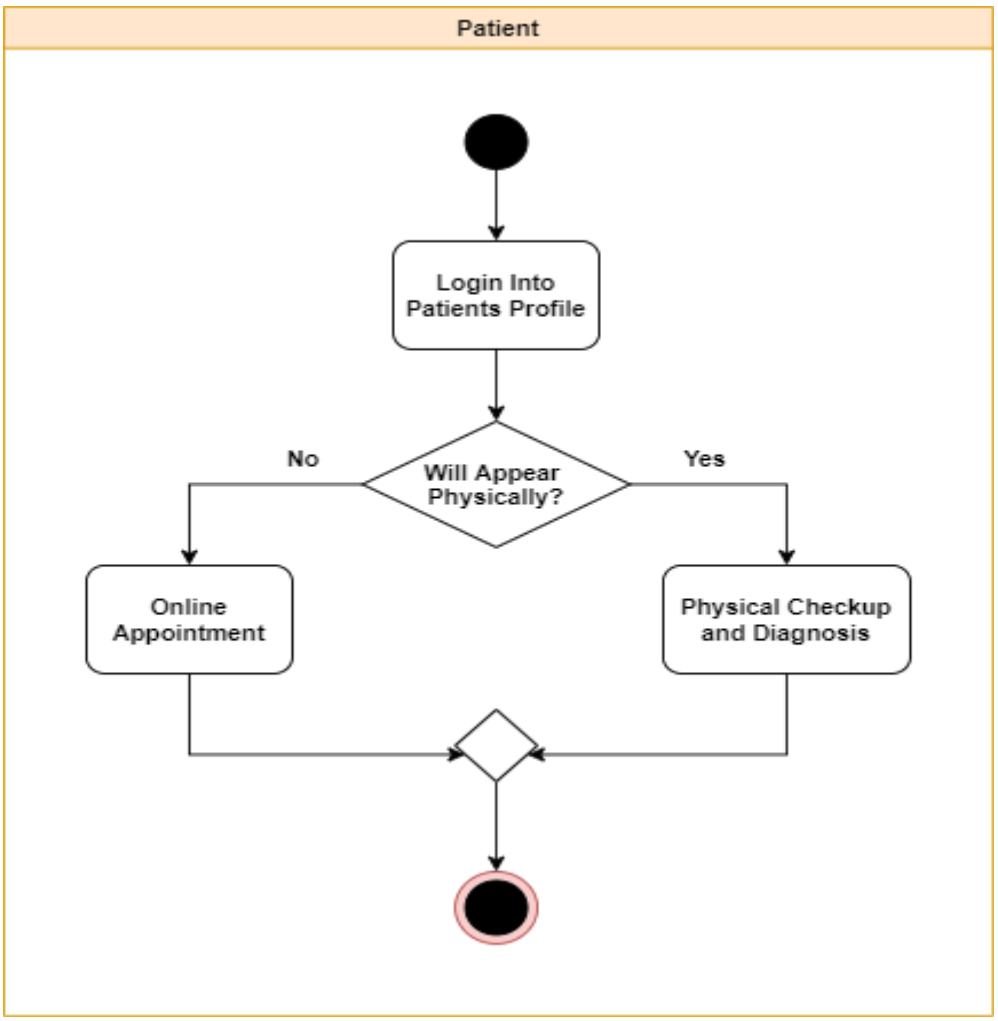


Scheduling time and date for appointment (Swimlane Diagram Level - 1.2.1)

Level-1.2.2

Name : Types of Appointment

Reference : Use Case & Activity Diagram- Level 1.2.2

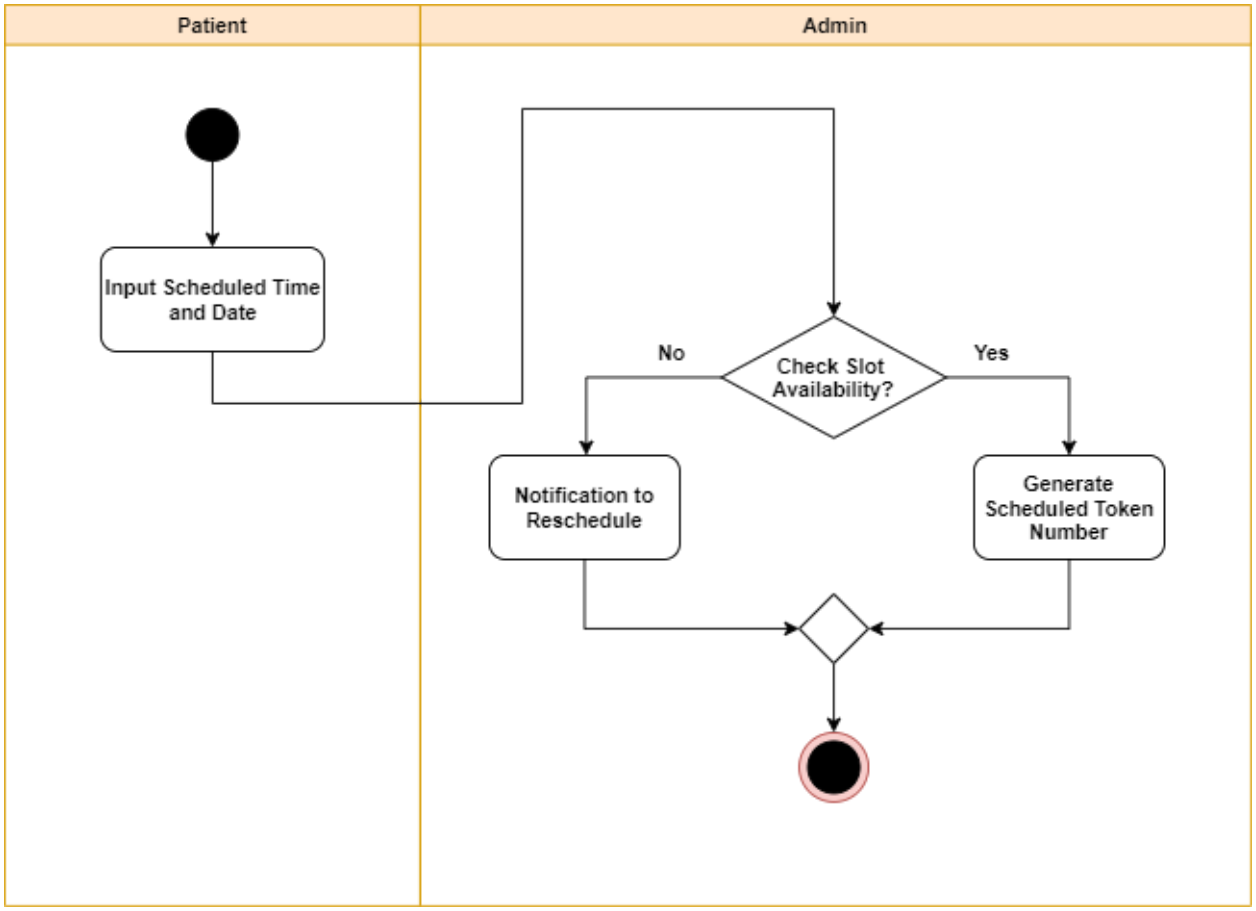


Types of Appointment (Swimlane Diagram Level - 1.2.2)

Level-1.2.3

Name : Token Number Generation

Reference : Use Case & Activity Diagram- Level 1.2.3

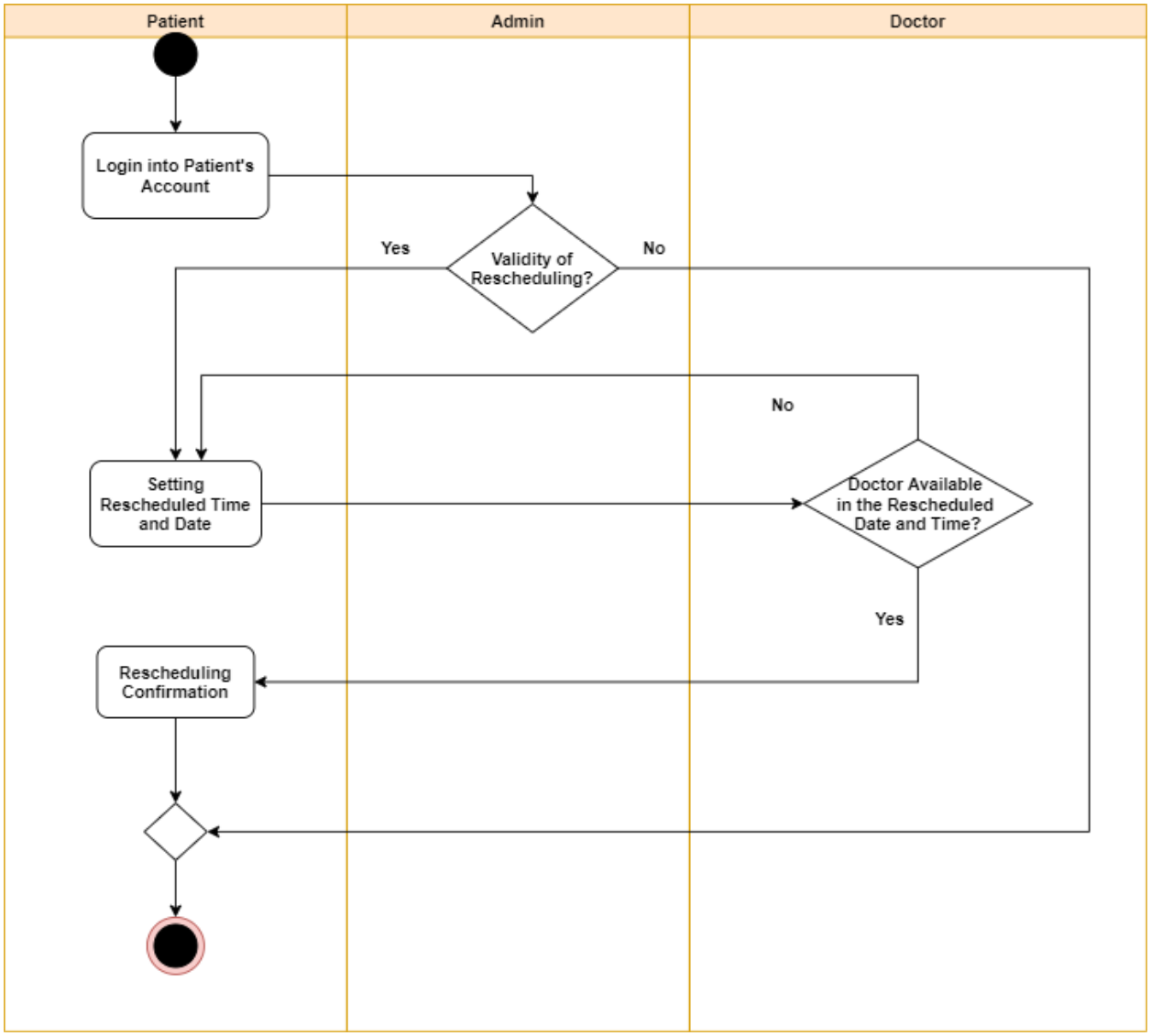


Token Number Generation (Swimlane Diagram Level - 1.2.3)

Level-1.2.4

Name : Rescheduling Appointment

Reference : Use Case & Activity Diagram- Level 1.2.4

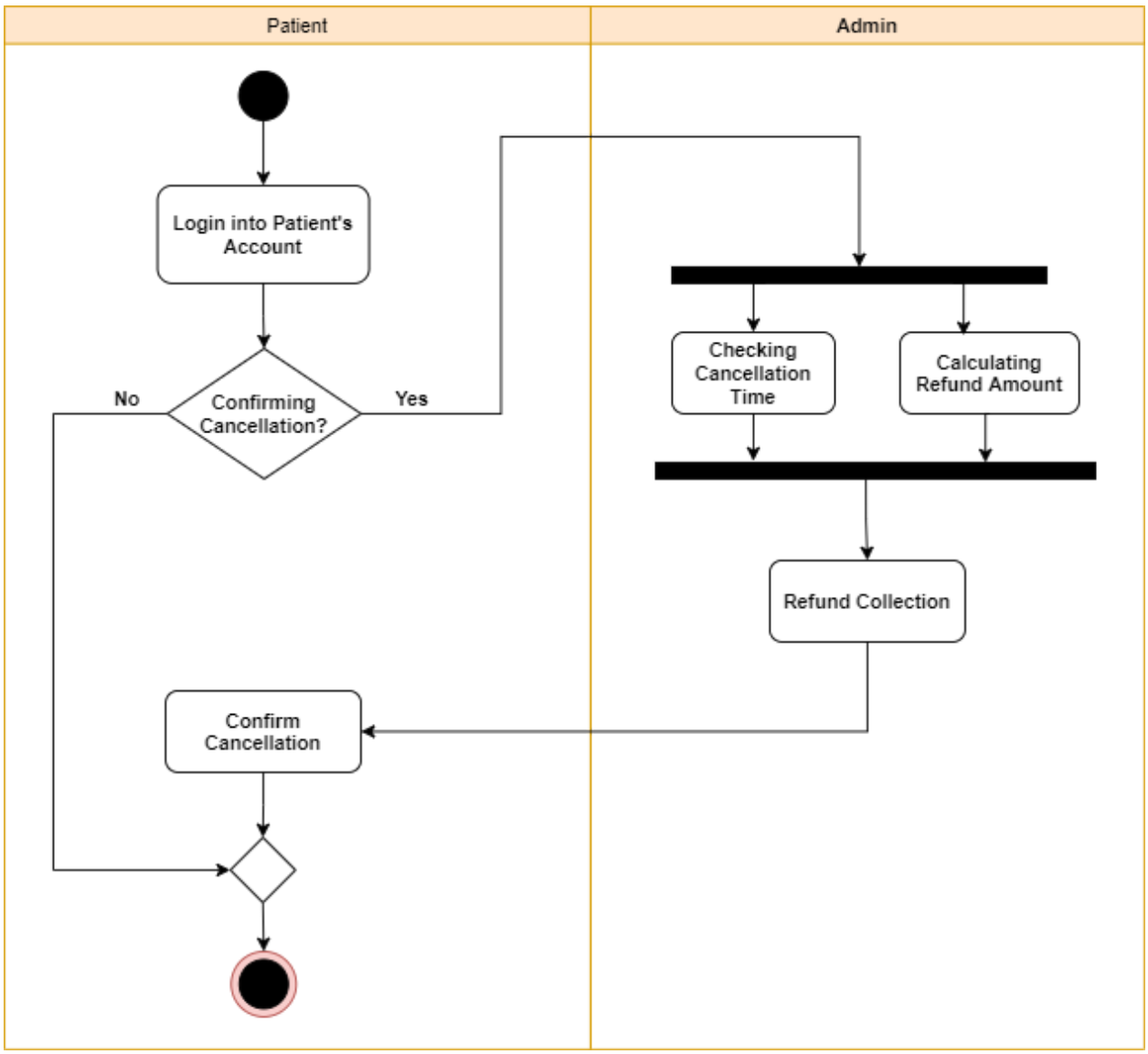


Rescheduling Appointment (Swimlane Diagram Level - 1.2.4)

Level-1.2.5

Name : Appointment Cancellation

Reference : Use Case & Activity Diagram- Level 1.2.5

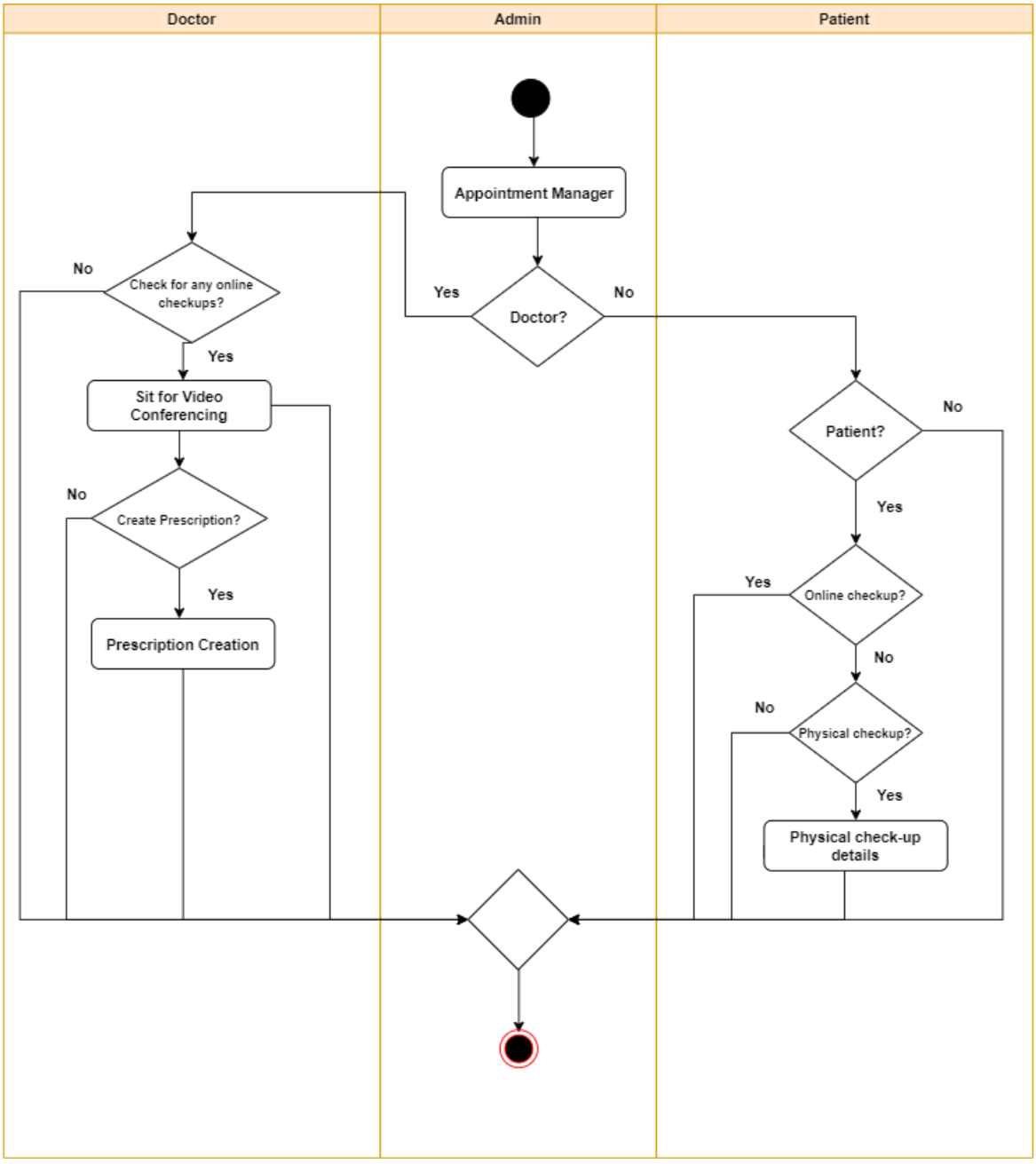


Appointment Cancellation (Swimlane Diagram Level - 1.2.5)

Level-1.3

Name : Consultation

Reference : Use Case & Activity Diagram- Level 1.3

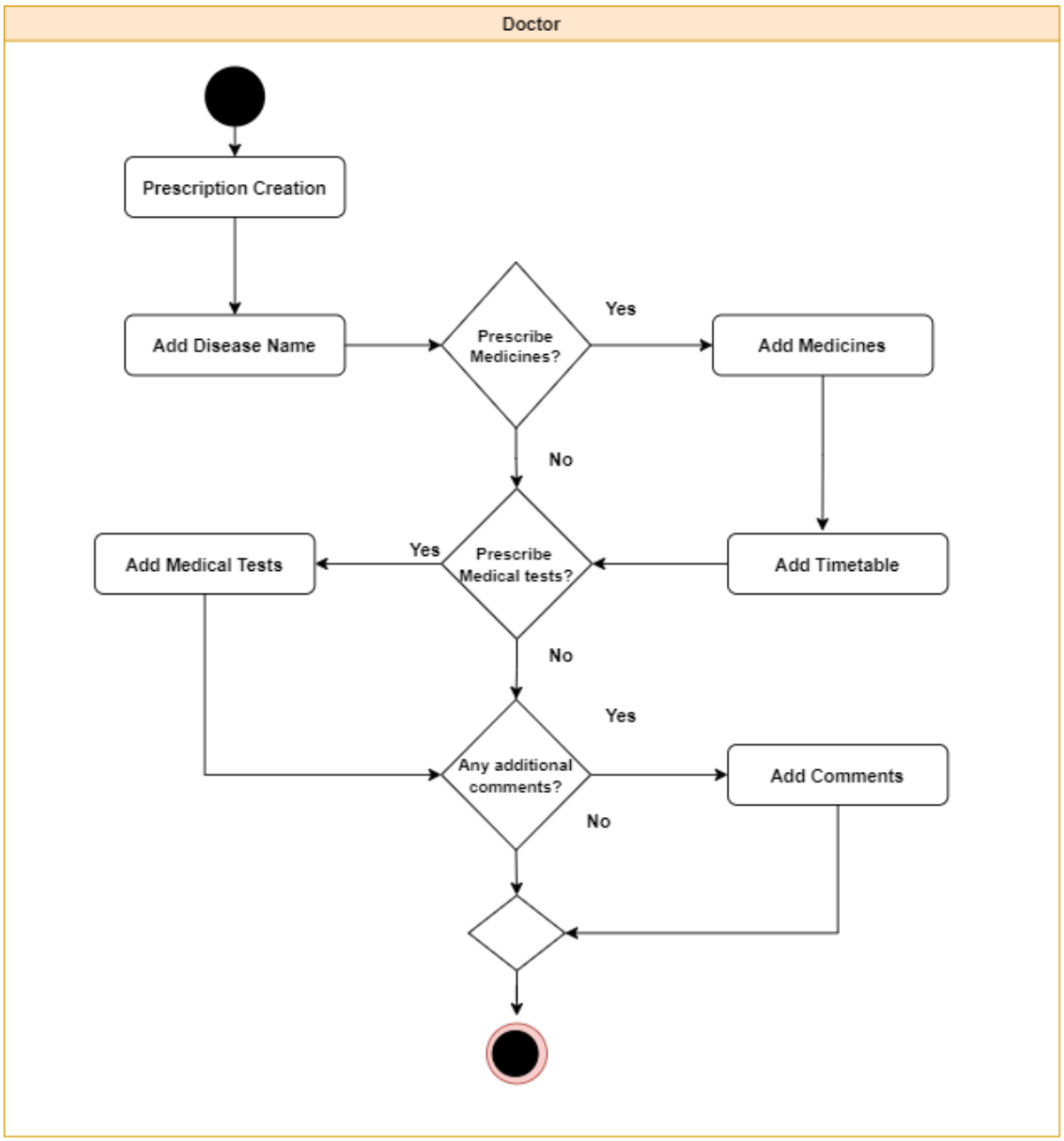


Consultation(Swimlane Diagram Level - 1.3)

Level-1.3.3

Name : Prescription Creation

Reference : Use Case & Activity Diagram- Level 1.3.3

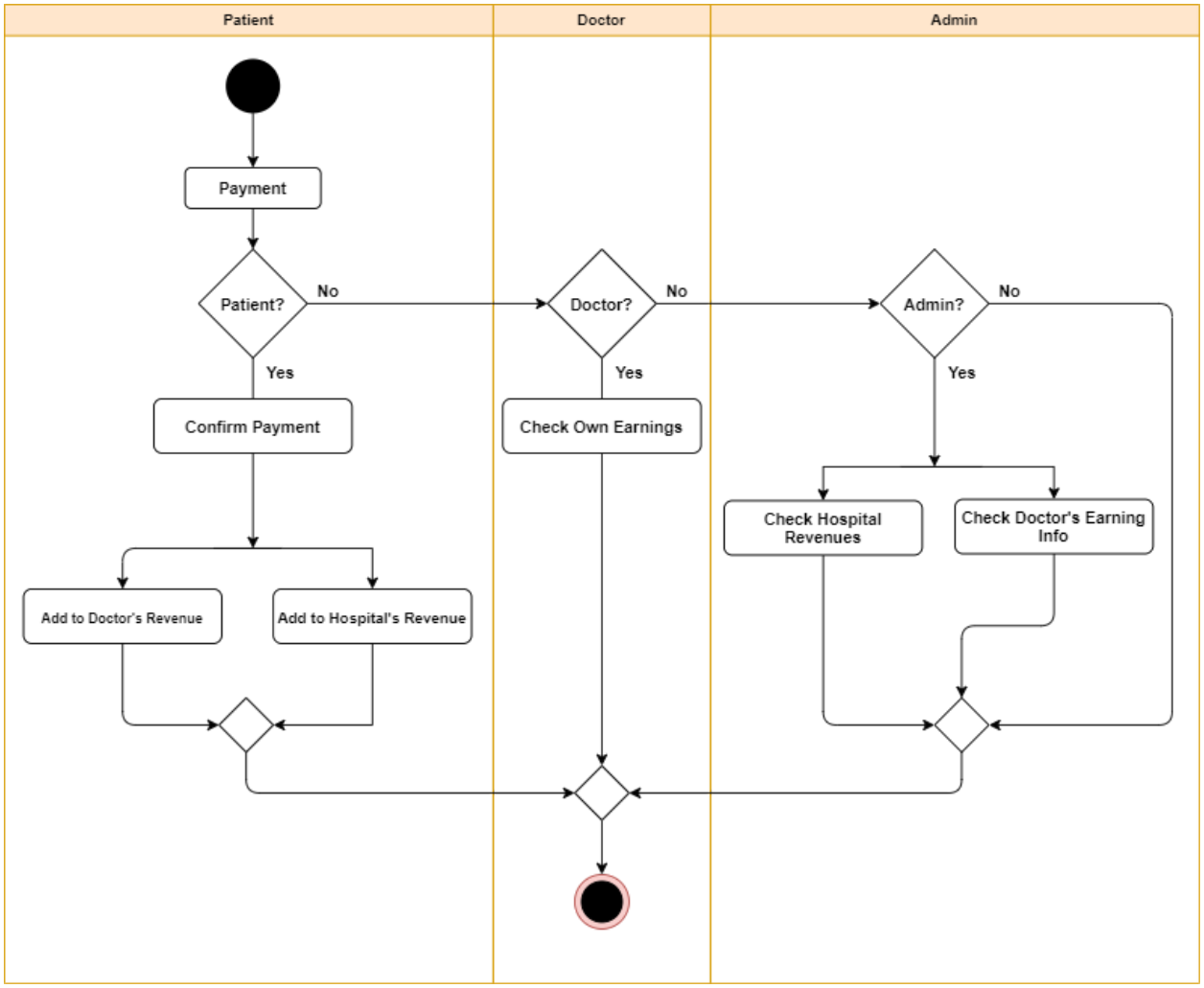


Prescription Creation (Swimlane Diagram Level - 1.3.3)

Level-1.4

Name : Payment

Reference : Use Case & Activity Diagram- Level 1.4

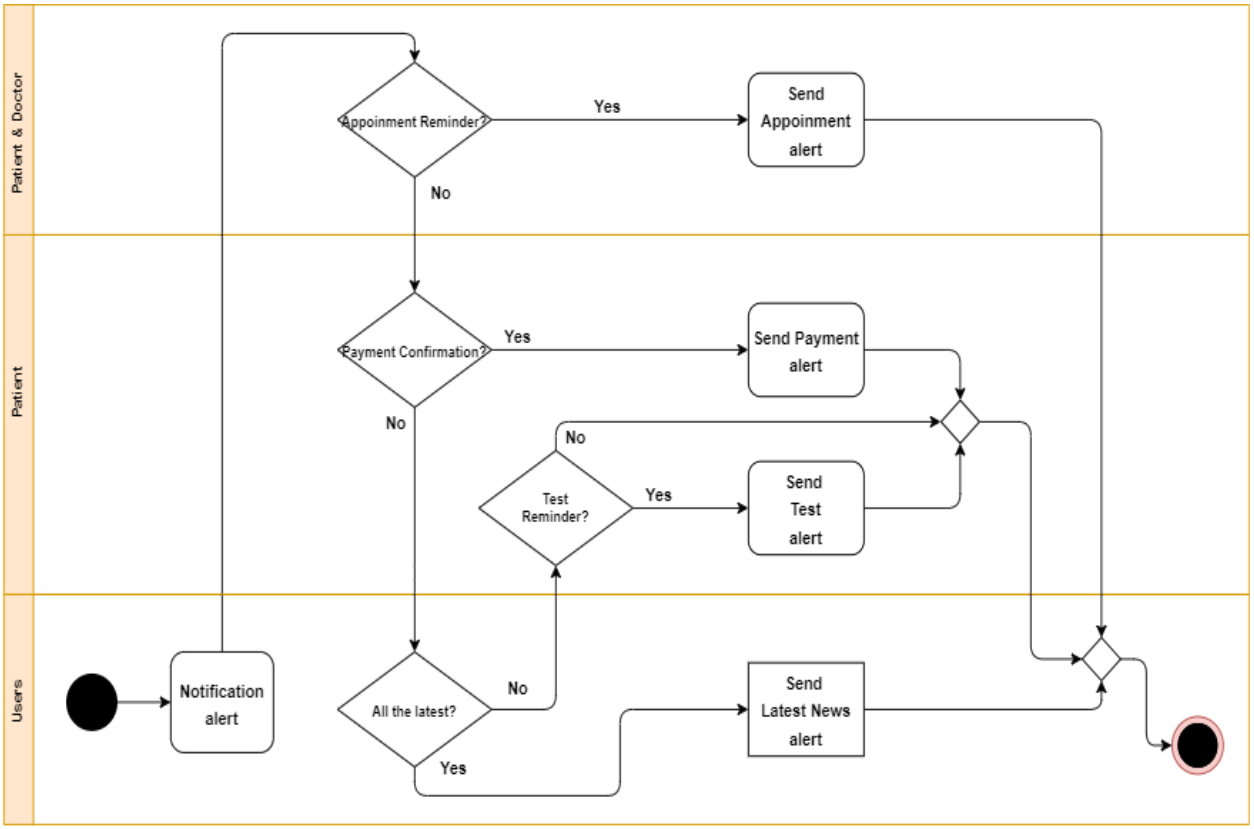


Payment (Swimlane Diagram Level - 1.4)

Level-1.5

Name : Notification

Reference : Use Case & Activity Diagram- Level 1.5

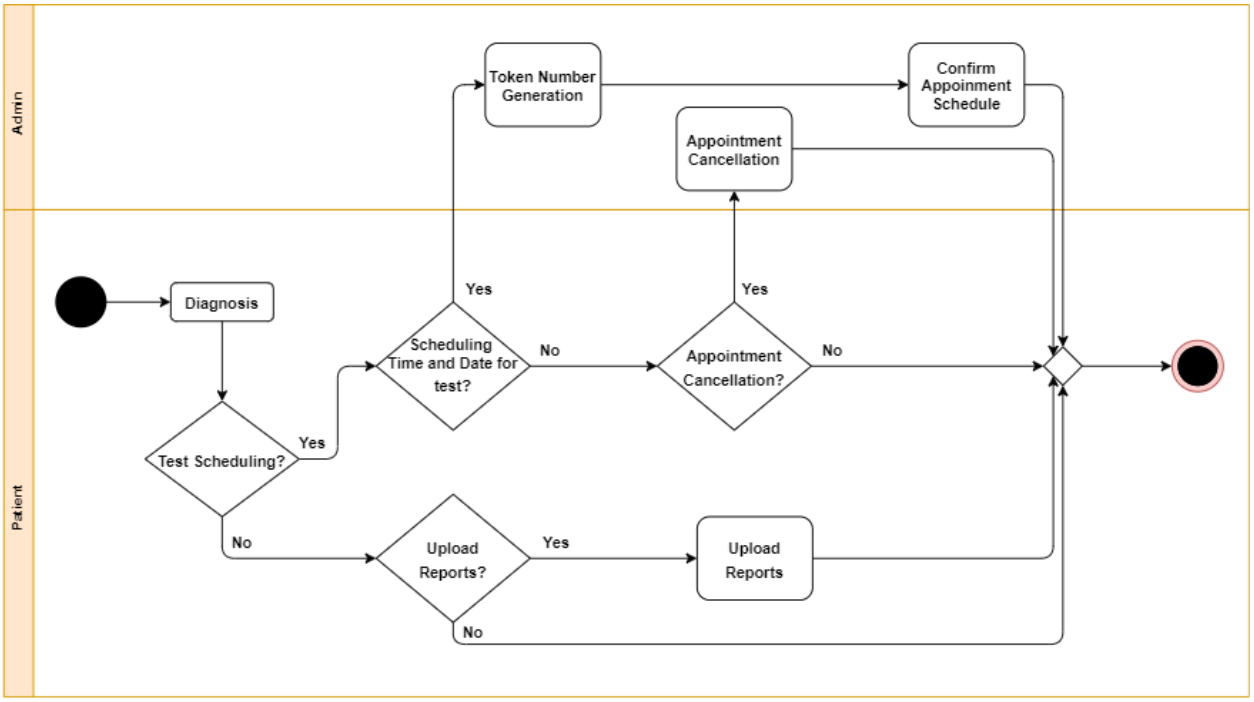


Notification (Swimlane Diagram Level - 1.5)

Level 1.6

Name : Diagnosis

Reference : Use Case & Activity Diagram- Level 1.6

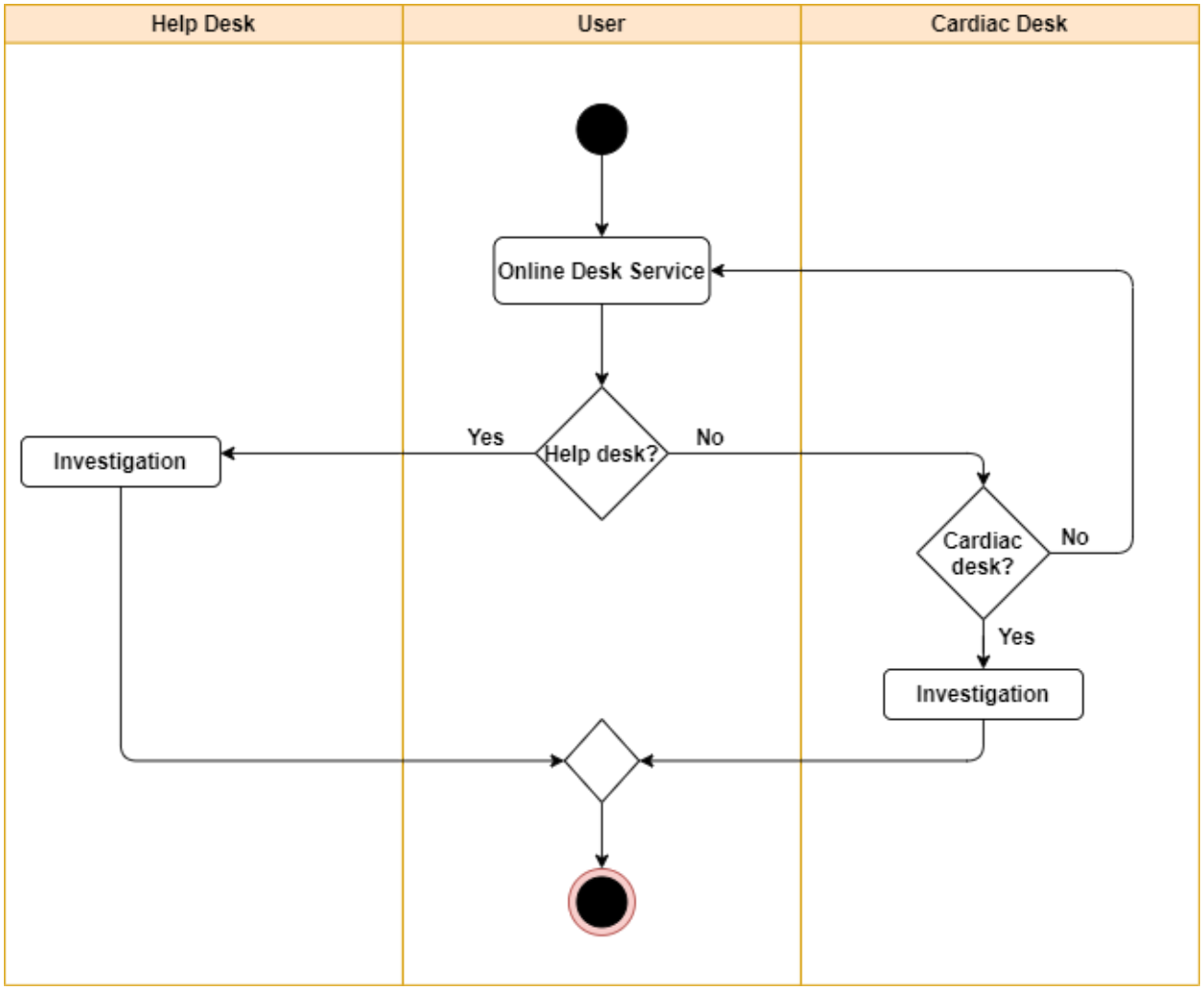


Diagnosis (Swimlane Diagram Level - 1.6)

Level-1.7

Name : Online Desk Services

Reference : Use Case & Activity Diagram- Level 1.7

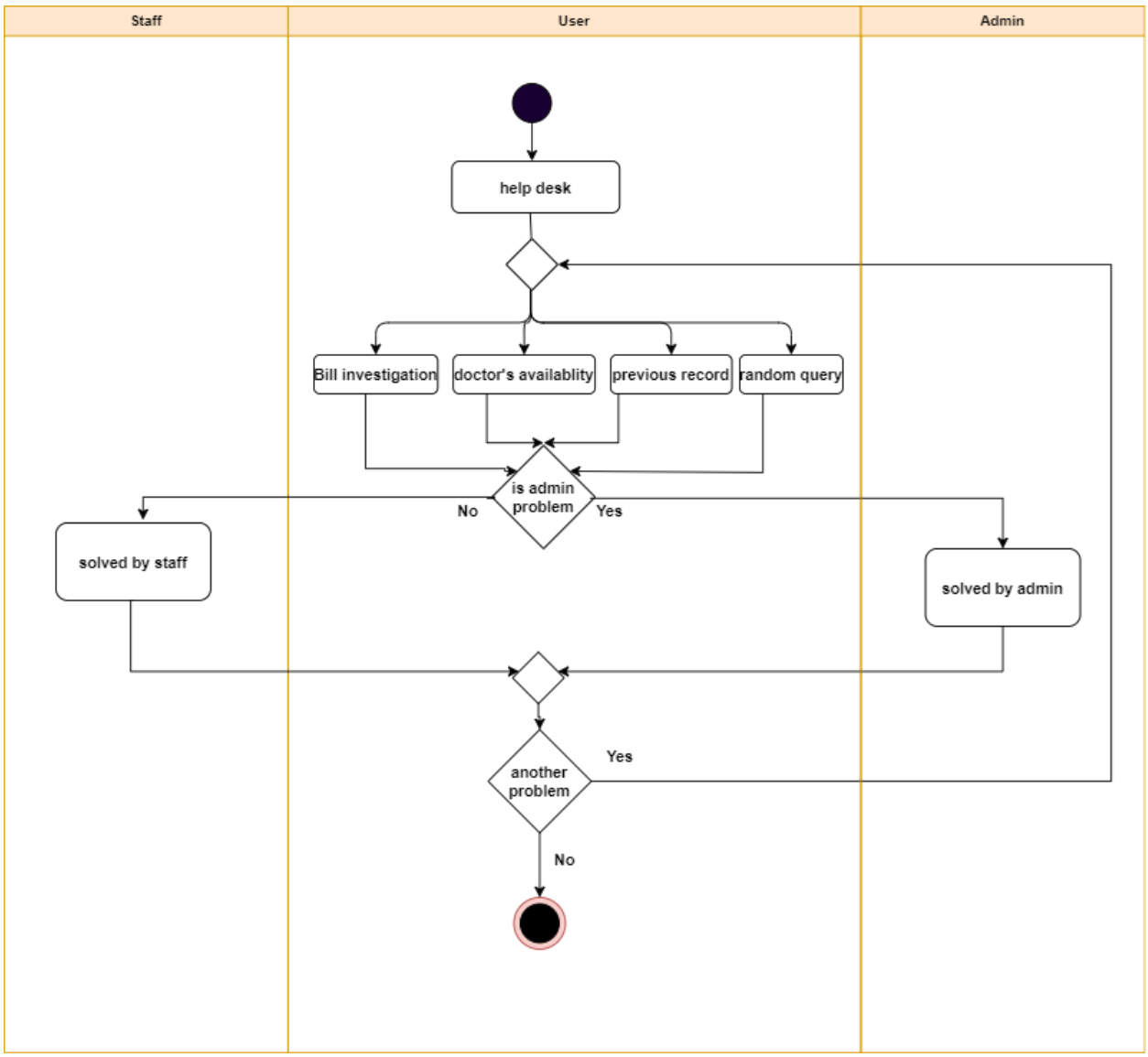


Online Desk Services (Swimlane Diagram Level - 1.7)

Level-1.7.1

Name : Help Desk

Reference : Use Case & Activity Diagram- Level 1.7.1

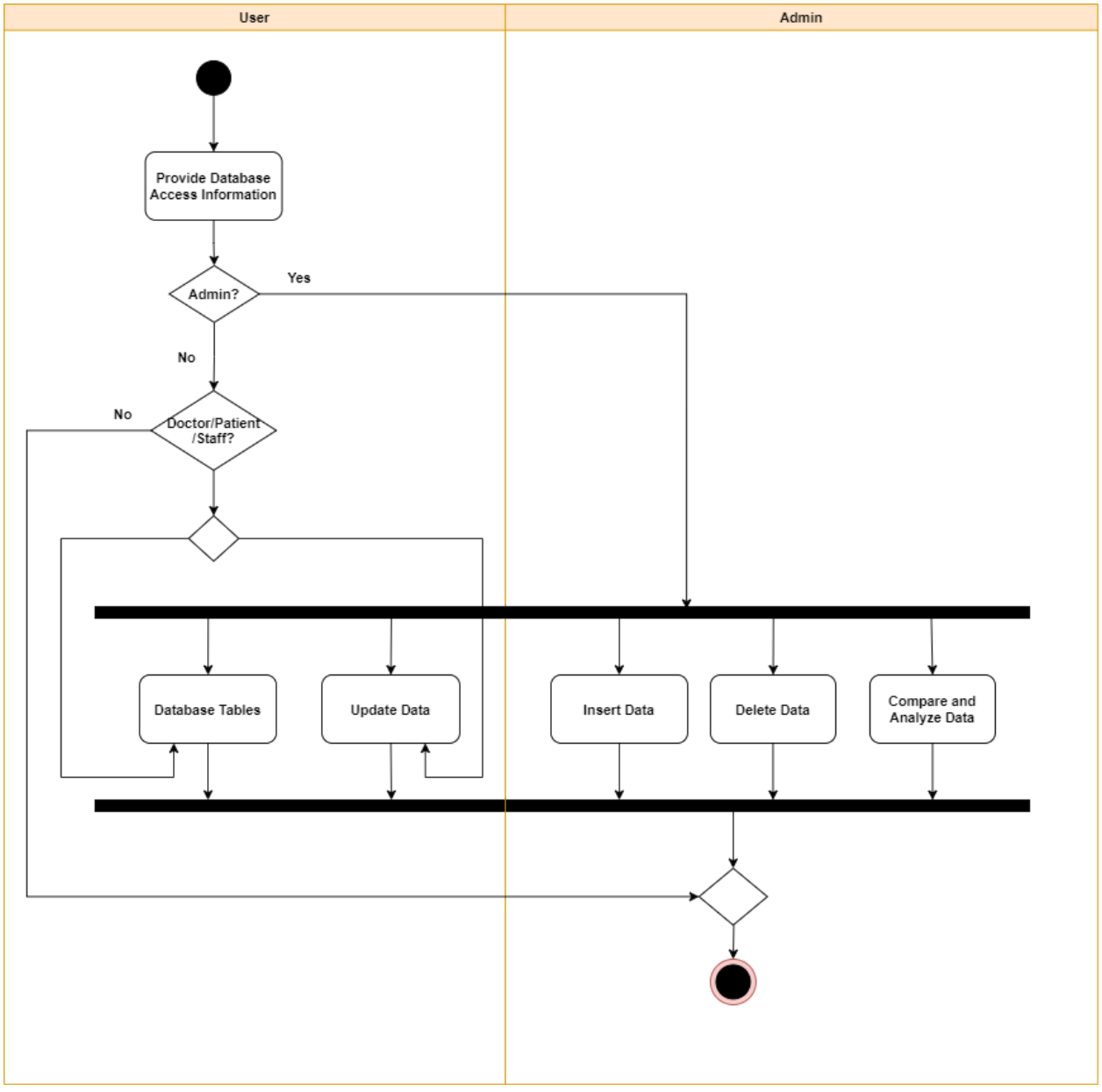


Help Desk (Swimlane Diagram Level - 1.7.1)

Level 1.8

Name: Database

Reference : Use Case & Activity Diagram- Level 1.8



Database (Swimlane Diagram Level - 1.8)

Data Based Modeling : Hospital Management System

Data Modeling Concept

A Data Model is an organized view of database concepts and their relationships. The purpose of creating a conceptual data model is to establish entities, their attributes, and relationships. In this data modeling level, there is hardly any detail available on the actual database structure.

The 3 basic elements of Data Modeling are-

- Entity: A real-world thing
- Attribute: Characteristics or properties of an entity
- Relationship: Dependency or association between two entities

The entity relationship diagram (ERD) defines all data objects that are processed within the system, the relationships between the data objects and the information about how the data objects are entered, stored, transformed and produced within the system.

The main goal of a designing data model is to make certain that data objects offered by the functional team are represented accurately. It should be detailed enough to be used for building the physical database.

Data Objects

A data object is a collection of one or more data points that create meaning as a whole. In other words, it's a storage or container to store data values. More specifically, data objects are usable, functional, and meaningful artefacts whose form and function is to encode data. It can be used in type checking operations. A data object can be an external entity, a thing, an occurrence, a role, an organizational unit, a place or a structure.

Data object identification

Serial	Noun	Solution(s)/ Problem(p) space	Attribute
1	HMS	p	
2	Account	p	
3	Admin	s	4,5,6,7,8,12,13,25
4	Full_name	s	
5	Email	s	
6	Password	s	
7	Mobile_no	s	
8	Address	s	
9	Doctor	s	4,5,6,7,8,12, 13,16,17,18,51
10	Patient	s	4,5,6,7,8,12,13, 14,15,28,45,52
11	Staff	s	17,18,19
12	Gender	s	
13	Age	s	
14	History	s	
15	Diseases	s	
16	Speciality	s	
17	Work_hour	s	
18	Designation	s	
19	Salary	s	

Serial	Noun	Solution(s)/ Problem(p) space	Attribute
20	Appointment	p	22,23,24,26,27,28,29
21	Revenue	s	
22	Schedule	s	
23	Date	s	
24	Time	s	
25	ID	s	
26	Slot	p	
27	Reschedule	p	
28	Token	s	
29	Online	p	
30	Cancel	p	
31	Checkup	p	
32	Desk	p	
33	Payment	s	21,34,35,36, 37,41,44,64
34	Type	p	
35	Timestamp	s	
36	Transaction_id	s	
37	SSLcommerz	s	
38	Notification email	s	
39	Notification sms	s	
40	Refund	p	

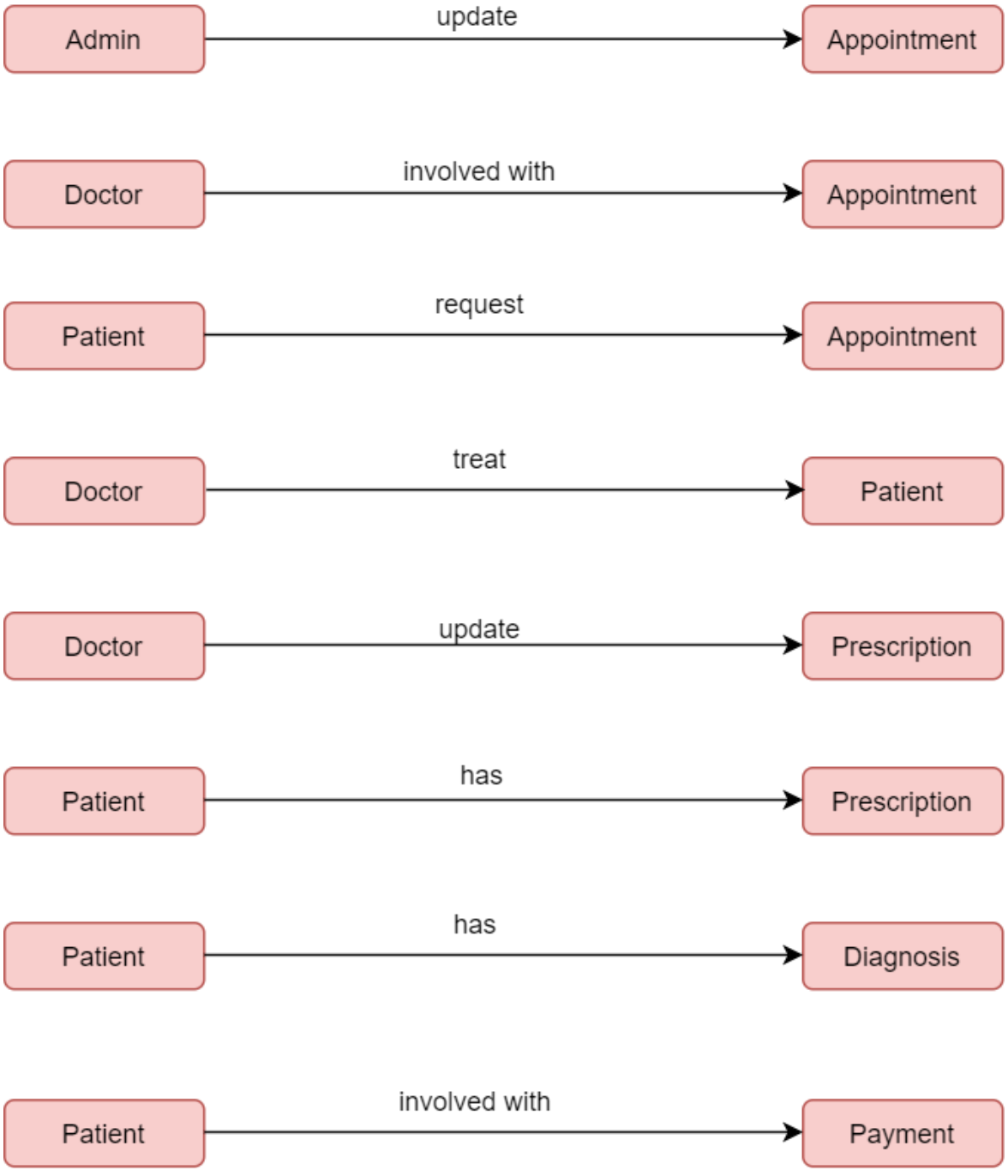
Serial	Noun	Solution(s)/ Problem(p) space	Attribute
41	Revenue	p	
42	Remainder	p	
43	Calculate	p	
44	Transaction_history	s	
45	Prescription	s	46,47,48,49,50
46	Timetable	s	
47	Medicine	s	
48	Comment	s	
49	Remark	s	
50	Diagnosis	s	53,54,55,56,57
51	Doctor_id	s	
52	Patient_id	s	
53	Report	s	
54	Reference	s	
55	Cost	p	
56	Bill	s	
57	Diagnosis_id	s	
58	Notify_patient	p	
59	Notification	s	60,61,62,63,38,39
60	Reminder	p	
61	Notification_from	s	
62	Notification_to	s	

Serial	Noun	Solution(s)/ Problem(p) space	Attribute
63	Body	s	
64	Payment_info	s	
65	Doctor_list	s	
66	Tools	s	
67	Apparatus	s	
68	Audio_call	p	
69	VIdeo_call	p	
70	feedback	s	
71	availability	p	
72	news_info	s	
73	health_tips	s	
74	conference	s	
75	message	s	
76	amount	s	
77	records	s	
78	consultation	p	
79	recovery	p	

Final data objects

1. Admin
2. Doctor
3. Patient
4. Appointment
5. Prescription
6. Diagnosis
7. Payment
8. Notification

Relationship Between Data Objects



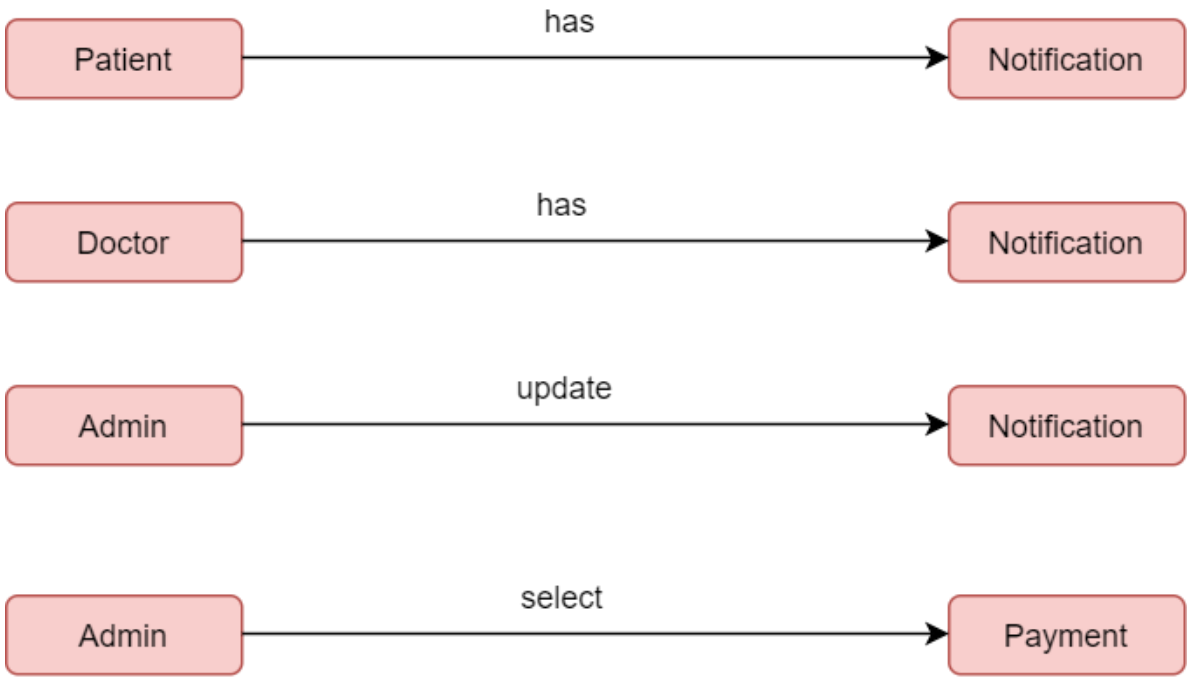


Figure: Relationship between Data Objects

ER Diagram

ER Diagram stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships.

ER Diagrams contain different symbols that use rectangles to represent entities, ovals to define attributes and diamond shapes to represent relationships.

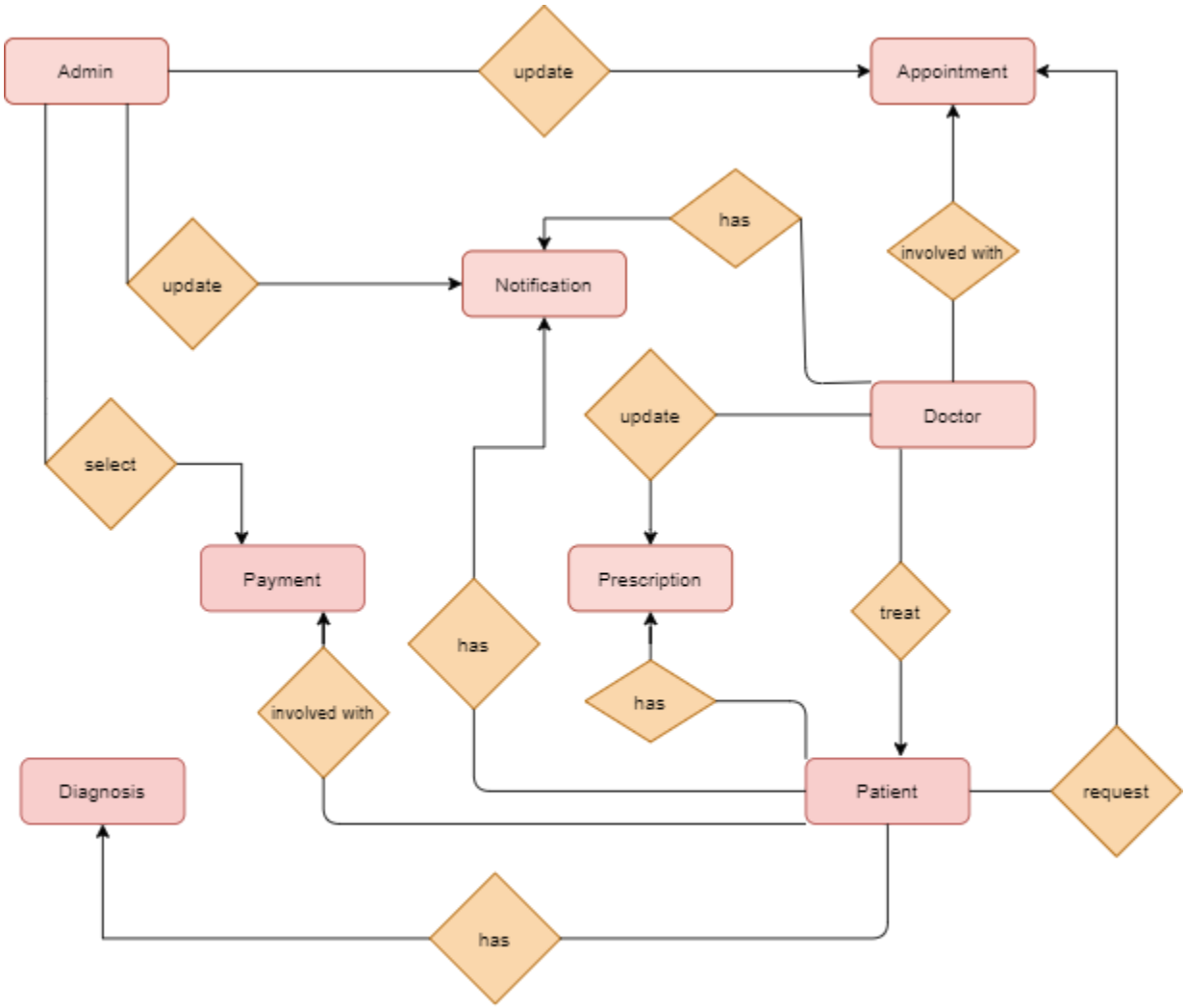


Figure: ER Diagram of Data Objects

Schema Diagram

Data object	Attribute	Type	Size
Admin	-id	VARCHAR	24
	-full_name	VARCHAR	20
	-email	VARCHAR	40
	-password	VARCHAR	20
	-mobile_no	VARCHAR	15
	-address	VARCHAR	60
	-gender	VARCHAR	8
	-age	NUMBER	3
Doctor	-id	VARCHAR	24
	-full_name	VARCHAR	20
	-email	VARCHAR	40
	-password	VARCHAR	20
	-mobile_no	VARCHAR	15
	-address	VARCHAR	60
	-gender	VARCHAR	8
	-age	NUMBER	3
	-speciality	VARCHAR	20
	-designation	VARCHAR	20
	-work_hour	VARCHAR	8
Patient	-id	VARCHAR	24
	-full_name	VARCHAR	20

	-email	VARCHAR	40
	-password	VARCHAR	20
	-mobile_no	VARCHAR	15
	-address	VARCHAR	60
	-gender	VARCHAR	8
	-age	NUMBER	3
	-history	VARCHAR	100
	-diseases	VARCHAR	100
	-token	VARCHAR	10
Appointment	-appointment_id	VARCHAR	24
	-schedule	VARCHAR	50
	-date	DATETIME	
	-time	TIME	
	-slot	VARCHAR	50
	-reschedule	VARCHAR	50
	-token	VARCHAR	10
	-online	VARCHAR	10
Prescription	-prescription_id	VARCHAR	24
	-timetable	TIME	
	-medicin	VARCHAR	100
	-comment	VARCHAR	100
	-remark	VARCHAR	100
	-diagnosis	DIAGNOSIS	

Diagnosis	-diagnosis_id -report -reference -cost -bill	VARCHAR VARCHAR VARCHAR NUMBER BILL	24 100 24 (2,6)
Payment	-transaction_id -type -timestamp -amount -revenue -payment_history -payment_info	VARCHAR VARCHAR DATETIME NUMBER NUMBER PAYMENT VARCHAR	40 8 (2,6) (2,6) 256
Notification	-notification_id -reminder -notification_from -notification_to -body	VARCHAR VARCHAR VARCHAR VARCHAR VARCHAR	24 50 24 24 100

Class Based Modeling : Hospital Management System

Class Based Modeling Concept

Class-based modeling identifies classes, attributes and relationships that the system will use. It represents the object. The system manipulates the operations.

The elements of the class based model consist of classes and object, attributes, operations, class – responsibility - collaborator (CRC) models.
Classes are determined using underlining each noun or noun clause and entering it into the simple table. Attributes are the set of data objects that are defining a complete class within the context of the problem. The operations define the behavior of an object.

Noun List from Hospital Management System

Serial No.	Noun
1	HMS
2	Account
3	Admin
4	Full_name
5	Email
6	Password
7	Mobile_no
8	Address
9	Doctor
10	Patient
11	Staff

Serial No.	Noun
12	Gender
13	Age
14	History
15	Diseases
16	Speciality
17	Work_hour
18	Designation
19	Salary
20	Appointment
21	Revenue
22	Schedule
23	Date
24	Time
25	ID
26	Slot
27	Reschedule
28	Token
29	Online
30	Cancel
31	Checkup
32	Desk
33	Payment
34	Type
35	Timestamp

Serial No.	Noun
36	Transaction_id
37	SSLcommerz
38	Notification email
39	Notification sms
40	Refund
41	Revenue
42	Remainder
43	Calculate
44	Transaction_history
45	Prescription
46	Timetable
47	Medicine
48	Comment
49	Remark
50	Diagnosis
51	Doctor_id
52	Patient_id
53	Report
54	Reference
55	Cost
56	Bill
57	Diagnosis_id
58	Notify_patient
59	Notification

Serial No.	Noun
60	Reminder
61	Notification_from
62	Notification_to
63	Body
64	Payment_info
65	Doctor_list
66	Tools
67	Apparatus
68	Audio_call
69	VIdeo_call
70	feedback
71	availability
72	news_info
73	health_tips
74	conference
75	message
76	amount
77	records
78	consultation
79	recovery
80	visitation

Verb List from Hospital Management System

Serial No.	Verb
1	Create Account
2	Show Info
3	Sign up
4	Login
5	Recover Account
6	Deactivate Account
7	Schedule
8	Generate (Token)
9	Reschedule
10	Cancel
11	Check
12	Valid (to reschedule)
13	Confirm
14	Calculate (Refund amount)
15	Collect
16	Inquire
17	Consult
18	Create (Appointment/Prescription)
19	Call
20	Connect
21	Refund
22	Upload (Reports)
23	Help

Serial No.	Verb
24	Assist
25	Investigate Bill
26	Record Info
27	Query
28	Diagnose
29	Insert
30	Update
31	Delete
32	Compare
33	Analyze
34	Generate (Revenue)
35	Pay
36	Remind
37	Notify
38	Write (Blogs)
39	Add Info
40	Select (Options)
41	Request (Follow-up check)
42	Prescribe
43	Suggest diagnosis
44	Resolve
45	Post tips
46	Post news
47	Create report

Serial No.	**Verb**
48	Calculate slot

General Classification

Candidate classes were then characterized in seven general classes. The seven general characteristics are as follows :

1. External entities
2. Things
3. Events
4. Roles
5. Organizational units
6. Places
7. Structures

Table of General Classification

Serial No.	**Noun**	**General Classification**
1	HMS	2
2	Doctor	4, 5, 7
3	Person	4, 5, 7
4	Name	2, 7
5	User name	2, 7
6	Speciality	2
7	Working hour	2, 3

Serial No.	Noun	General Classification
8	Patient	4, 5, 7
9	Gender	4, 7
10	Age	2
11	History	2
12	Staff	4, 5, 7
13	Date of birth	2, 3, 7
14	Designation	2, 4, 5
15	Admin	4, 5, 7
16	Email	1, 2
17	Password	2
18	Mobile number	1, 2
19	Address	2, 6
20	Credentials	7
21	Appointment	2, 3, 7
22	Schedule	3, 7
23	Date and time	2, 3
24	Slot	2
25	Token	2
26	Online appointment	2, 3
27	Cancel appointment	2, 3
28	Reschedule	2, 3
29	Payment	1, 2, 7
30	Online	2

Serial No.	Noun	General Classification
31	sslcommerz	1
32	Refund	2
33	Revenue	2, 7
34	Diagnosis	2, 5, 7
35	Diseases	2, 7
36	Report	2, 7
37	Bill	2, 7
38	Prescription	2, 3, 7
39	Medicine	2
40	Time table	3
41	Comment	7
42	Remark	2
43	Notification	2, 3, 7
44	Payment confirmation	2,3,7
45	Appointment reminder	2,3
46	Diagnosis time	2,3
47	Diagnosis report	2,7
48	Database	2,7
49	Account	1,2,5,7
50	Guest	4
51	Manager	4,7
52	Consultation	3

Serial No.	Noun	General Classification
53	Visitation	3
54	Online desk	2,7
55	Online conference	3
56	Check -up	3
57	Video call	3
58	Audio call	3
59	Message	3
60	Health tips	3
61	Hospital news	3
62	Special service	3
63	Medical apparatus	2
64	Availability	3
65	Help desk	2,6,7
66	Random queries	3
67	Feedback	3,7
68	Calendar	2

Potential to be classes

- HMS
- Account
- Admin
- Doctor
- Patient
- Staff
- Appointment
- Notification
- Diagnosis
- Prescription
- Payment
- Database
- Help Desk
- Online Desk

Selection Criteria

The candidate classes are then selected as classes by six Selection Criteria.

1. Retain information
2. Needed services
3. Multiple attributes
4. Common attributes
5. Common operations
6. Essential requirements

A candidate class generally becomes a class when it fulfills around three characteristics.

Table of Selection Criteria

Serial No.	Noun	Selection Criteria
1	HMS	1,2,3,4,5
2	Account	1,2,3,4,5
3	Admin	1,2,3,4,5
4	Doctor	1,2,3,4,5
5	Patient	1,2,3,4,5
6	Staff	1,2,3,4,5
7	Appointment	2,3,6
8	Notification	2,3,6
9	Diagnosis	2,3,6
10	Prescription	2,3,6

11	Payment	2,3,6
12	Database	6
13	Help Desk	3,4,5
14	Online Desk	3,4,5

Selected Classes

- HMS
- Account
- Admin
- Doctor
- Patient
- Staff
- Appointment
- Notification
- Diagnosis
- Prescription
- Payment
- Database
- Help Desk
- Online Desk

Attribute and Method Identification

HMS	
Attribute	**Method**
-hospital_name -address	+addAdmin() +getAccount() +getAdmin()

Account	
Attribute	**Method**
-account_id -user_name -account_type	+sign_up() +login() +get_type() +forget_password() +recover_account() +get_id()

Admin	
Attribute	**Method**
-full_name	+addDoctor()

-email -phone_number -address -password	+addPatient() +createAppointment() +verifyDoctor() +addStaff() +updateDatabase() +viewHelpDesk() +receivePayment() +processPayment() +notifyDoctor() +calculateRevenue() +setAppointment() +cancelAppointment() +rescheduleAppointment()

Patient	
Attribute	**Method**
-full_name -phone_number -address -gender -age -sickness -history -query	+requestAppointment() +getDiagnosisResult() +getDiagnosisTime() +getPrescription() +viewPastHistory() +pay() +cancelAppointment() +getRefund() +requestCheckUp()

	+viewNotification() +viewDoctorList() +viewTimetable() +selectDoctor() +getToken() +askHelpDesk() +askOnlineDesk()

Doctor	
Attribute	**Method**
-full_name -address -speciality -working_hour -date_of_barth	+createPrescription() +getPatient() +getRevenue() +setAppointment() +treatPatient() +suggestDiagnosis() +viewNotification() +rescheduleMeet()

Staff	
Attribute	**Method**
-full_name -designation -date_of_barth -working_hour	+createDiagnosis() +resolveHelpDesk() +postHospitalNews() +postHealthTips() +getWorkingHour() +getDesignation() +getSallary() +createReport() +createBill() +notifyPatient()

Appointment	
Attribute	**Method**
-date -time -patient_id -diagnosis_id -doctor_id	+getSlot() +calculateSlot() +notifyOnTime() +giveAppointment() +cancelAppointment() +getDateTime() +setDateTime() +getDoctor()

	+getDiagnosis() +notifyUsers()

Payment	
Attribute	**Method**
-payment_id -payment_type -timestamp -transaction_id -isPaymentComplete	+calculatePayment() +getPaymentType() +notifyComplete() +getPaymentDetails() +callPaymentMethod() +transactionHistory() +refund()

Prescription	
Attribute	**Method**
-time_table -prescription_id -doctor_id -patient_id	+getDoctorId() +getPatientId() +getMedicine() +getTimeTable() +getComment() +getRemark()

	+getDiagnosis() +showPrescription() +diagnosis()

Diagnosis	
Attribute	**Method**
-diagnosis_id -reference -timestamp -cost -diseases -report -bill	+getTime() +getSlot() +createReport() +createBill() +getDoctorId() +getPatientId() +getReference() +notifyPatient() +setDeliveryTime()

Notification	
Attribute	**Method**
-timestamp -notification_id -isNotified -notification_from	+getNotificationId() +getBody() +getFrom() +getTo()

-notification_to	+setTime() +setNotificationNumber() +setDuration()

Database	
Attribute	**Method**
-madical_apparatus -doctor_list -diagnosis_tools -staff_list -payment_info	+showPaymentInfo() +showDoctor() +updateDoctor() +updateStaff() +updateApparatus() +manageTools()

Help desk	
Attribute	**Method**
-check_up -audio_call -vedio_call -availability -feedback	+getInfo() +call() +receiveCall() +getFeedback() +randomQuery() +getStaff() +resolve() +billInvestigation()

Online desk	
Attribute	**Method**
-news_info -health_tips -conference	+getNews() +getTips() +getConferance() +showNews() +showTips() +manageConferance() +notifyUsers()

CRC Card :

Table : CRC card for HMS

HMS	
Responsibilities	**Collaborator**
• Verify information • Add admin	Account Admin

Table : CRC card for Account

Account	
Responsibilities	**Collaborator**
• Sign up users • Login users • Recover user account • Recover password	Admin Doctor Staff Patient

Table : CRC card for Admin

Admin	
Responsibilities	**Collaborator**
• Add new doctor • Add new staff • Verify doctor • Update database • Receive payment from patient • Process payment for doctor and hospital • Notify doctor • Notify patient • Calculate revenue • Set appointment • Reschedule appointment • Cancel appointment • Manage database • Query	Doctor Patient Staff Appointment Payment Notification Database HelpDesk

Table : CRC card for Patient

Patient	
Responsibilities	**Collaborator**
• Request doctor appointment • Payment	Doctor Admin

• Request cancel appointment • Get refund • Request Checkup • See doctor list • See time table • Get token number • Follow prescription	Staff Appointment Notification Prescription Diagnosis HelpDesk OnlineDesk

Table : Class card for Doctor

Doctor	
Responsibilities	**Collaborator**
• Treat patient • Make prescription • Get patient info • Get revenue • Set appointment	Patient Admin Prescription Appointment Notification

Table : CRC card for Staff

Staff	
Responsibilities	**Collaborator**
• Resolve patient's help desk problem	Prescription Notification

• Post hospital news • Post health tips • Create diagnosis • Create diagnosis report • Create diagnosis bill • Notify patient	Diagnosis HelpDesk OnlineDesk

Table : CRC card for Appointment

Appointment	
Responsibilities	**Collaborator**
• Calculate correct slot • View doctor's list • Set date and time • Update date and time • Notify users	Admin Patient Doctor Staff

Table : CRC card for Payment

Payment	
Responsibilities	**Collaborator**
• Calculate payment • Show payment option	Patient Admin

• Show payment details • Get payment method • Show transaction history • Calculate refund	Doctor Notification

Table : CRC card for Prescription

Prescription	
Responsibilities	**Collaborator**
• Show prescription • Get medicine • Get time table • Get diagnosis • Get comment • Get remark • Get report	Patient Doctor Diagnosis Notification Staff

Table : CRC card for Diagnosis

Diagnosis	
Responsibilities	**Collaborator**
• Notify patient • Create report	Patient Staff

• Create Bill • Set delivery date and time • Save reference • Send report	Notification Payment

Table : CRC card for Notification

Notification	
Responsibilities	**Collaborator**
• Notify users • Notify time	Admin Patient Doctor Staff Diagnosis Payment

Table : CRC card for Database

Database	
Responsibilities	**Collaborator**
• Manage lists • Update payment info	Admin

Table : CRC card for Help desk

Help desk	
Responsibilities	**Collaborator**
• Get staff • Show doctor's info • Attend phone calls • Receive feedback • Investigate bill and response	Admin Staff Patient

Table : CRC card for Online desk

Online desk	
Responsibilities	**Collaborator**
• Post news • Post health tips • Manage online conference • Show conference details	Staff

Class Card

After identifying our final classes we have generated the following class cards-

Class : HMS	
Attribute	**Method**
-hospital_name -address	+addAdmin() +getAccount() +getAdmin()
Responsibilities	**Collaborator**
• Verify information • Add admin	Account Admin

Class : Account	
Attribute	**Method**
-account_id -user_name -account_type	+sign_up() +login() +get_type() +forget_password() +recover_account() +get_id()

Responsibilities	Collaborator
• Sign up users • Login users • Recover user account • Recover password	Admin Doctor Staff Patient

Class : Admin	
Attribute	**Method**
-full_name -email -phone_number -address -password	+addDoctor() +addPatient() +createAppointment() +verifyDoctor() +addStaff() +updateDatabase() +viewHelpDesk() +receivePayment() +processPayment() +notifyDoctor() +calculateRevenue() +setAppointment() +cancelAppointment() +rescheduleAppointment()

Responsibilities	Collaborator
• Add new doctor • Add new staff • Verify doctor • Update database • Receive payment from patient • Process payment for doctor and hospital • Notify doctor • Notify patient • Calculate revenue • Set appointment • Reschedule appointment • Cancel appointment • Manage database • Query	Doctor Patient Staff Appointment Payment Notification Database HelpDesk

Class : Patient	
Attribute	**Method**
-full_name -phone_number -address -gender -age	+requestAppointment() +getDiagnosisResult() +getDiagnosisTime() +getPrescription() +viewPastHistory()

-sickness -history -query	+pay() +cancelAppointment() +getRefund() +requestCheckUp() +viewNotification() +viewDoctorList() +viewTimetable() +selectDoctor() +getToken() +askHelpDesk() +askOnlineDesk()
Responsibilities	**Collaborator**
• Request doctor appointment • Payment • Request cancel appointment • Get refund • Request Checkup • See doctor list • See time table • Get token number • Follow prescription	Doctor Admin Staff Appointment Notification Prescription Diagnosis HelpDesk OnlineDesk

Class : Doctor	
Attribute	**Method**
-full_name -address -speciality -working_hour -date_of_barth	+createPrescription() +getPatient() +getRevenue() +setAppointment() +treatPatient() +suggestDiagnosis() +viewNotification() +rescheduleMeet()
Responsibilities	**Collaborator**
• Treat patient • Make prescription • Get patient info • Get revenue • Set appointment	Patient Admin Prescription Appointment Notification

Class : Staff	
Attribute	**Method**
-full_name -designation -date_of_barth -working_hour	+createDiagnosis() +resolveHelpDesk() +postHospitalNews() +postHealthTips() +getWorkingHour() +getDesignation() +getSallary() +createReport() +createBill() +notifyPatient()
Responsibilities	**Collaborator**
• Resolve patient's help desk problem • Post hospital news • Post health tips • Create diagnosis • Create diagnosis report • Create diagnosis bill • Notify patient	Prescription Notification Diagnosis HelpDesk OnlineDesk

Class : Appointment	
Attribute	**Method**
-date -time -patient_id -diagnosis_id -doctor_id	+getSlot() +calculateSlot() +notifyOnTime() +giveAppointment() +cancelAppointment() +getDateTime() +setDateTime() +getDoctor() +getDiagnosis() +notifyUsers()
Responsibilities	**Collaborator**
• Calculate correct slot • View doctor's list • Set date and time • Update date and time • Notify users	Admin Patient Doctor Staff

Class : Payment	
Attribute	**Method**
-payment_id -payment_type -timestamp -transaction_id -isPaymentComplete	+calculatePayment() +getPaymentType() +notifyComplete() +getPaymentDetails() +callPaymentMethod() +transactionHistory() +refund()
Responsibilities	**Collaborator**
• Calculate payment • Show payment option • Show payment details • Get payment method • Show transaction history • Calculate refund	Patient Admin Doctor Notification

Class : Prescription	
Attribute	**Method**
-time_table -prescription_id -doctor_id	+getDoctorId() +getPatientId() +getMedicine()

-patient_id	+getTimeTable() +getComment() +getRemark() +getDiagnosis() +showPrescription() +diagnosis()
Responsibilities	**Collaborator**
• Show prescription • Get medicine • Get time table • Get diagnosis • Get comment • Get remark • Get report	Patient Doctor Diagnosis Notification Staff

Class : Diagnosis	
Attribute	**Method**
-diagnosis_id -reference -timestamp -cost -diseases -report	+getTime() +getSlot() +createReport() +createBill() +getDoctorId() +getPatientId()

-bill	+getReference() +notifyPatient() +setDeliveryTime()
Responsibilities	**Collaborator**
• Notify patient • Create report • Create Bill • Set delivery date and time • Save reference • Send report	Patient Staff Notification Payment

Class : Notification	
Attribute	**Method**
-timestamp -notification_id -isNotified -notification_from -notification_to	+getNotificationId() +getBody() +getFrom() +getTo() +setTime() +setNotificationNumber() +setDuration()

Responsibilities	Collaborator
• Notify users • Notify time	Admin Patient Doctor Staff Diagnosis Payment

Class : Database	
Attribute	**Method**
-madical_apparatus -doctor_list -diagnosis_tools -staff_list -payment_info	+showPaymentInfo() +showDoctor() +updateDoctor() +updateStaff() +updateApparatus() +manageTools()
Responsibilities	**Collaborator**
• Manage lists • Update payment info	Admin

Class : Help desk	
Attribute	**Method**
-check_up -audio_call -vedio_call -availability -feedback	+getInfo() +call() +receiveCall() +getFeedback() +randomQuery() +getStaff() +resolve() +billInvestigation()
Responsibilities	**Collaborator**
• Get staff • Show doctor's info • Attend phone calls • Receive feedback • Investigate bill and response	Admin Staff Patient

Class : Online desk	
Attribute	**Method**
-news_info -health_tips -conference	+getNews() +getTips() +getConferance() +showNews() +showTips() +manageConferance() +notifyUsers()
Responsibilities	**Collaborator**
• Post news • Post health tips • Manage online conference • Show conference details	Staff

CRC Diagram

ID : 1

Name : Account

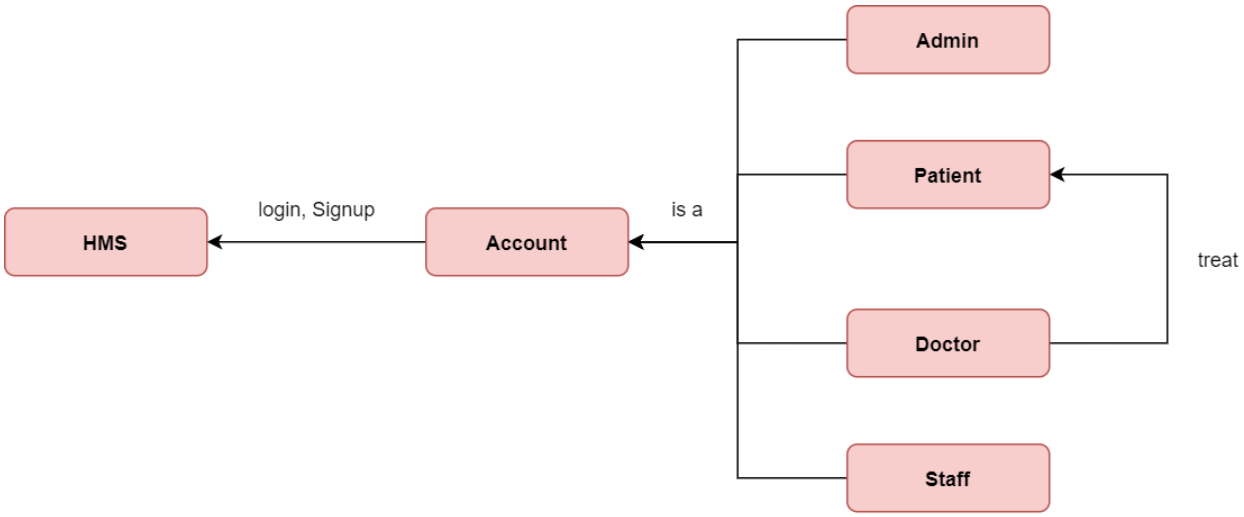


Figure: CRC Diagram for Account Class

ID : 2

Name : Admin

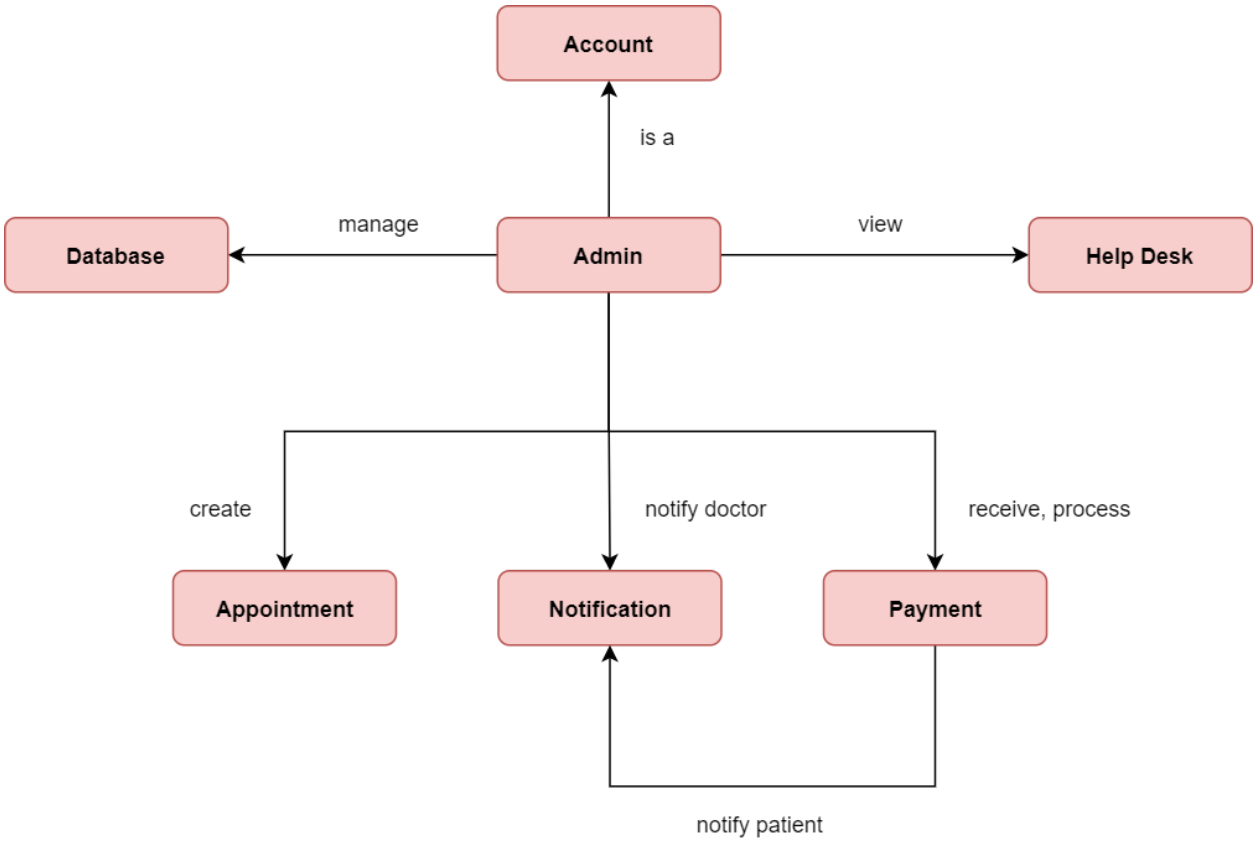


Figure: CRC Diagram for Admin Class

ID : 3

Name : Patient

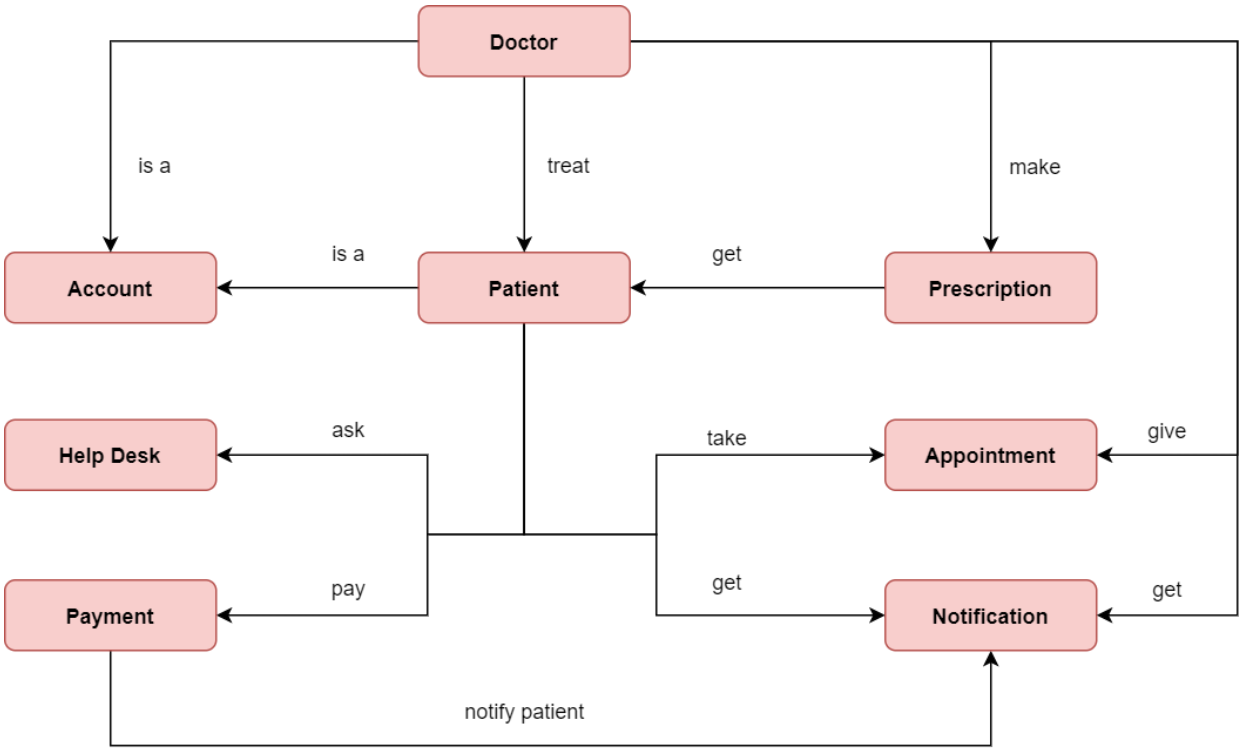


Figure: CRC Diagram for Patient Class

ID : 4

Name : Doctor

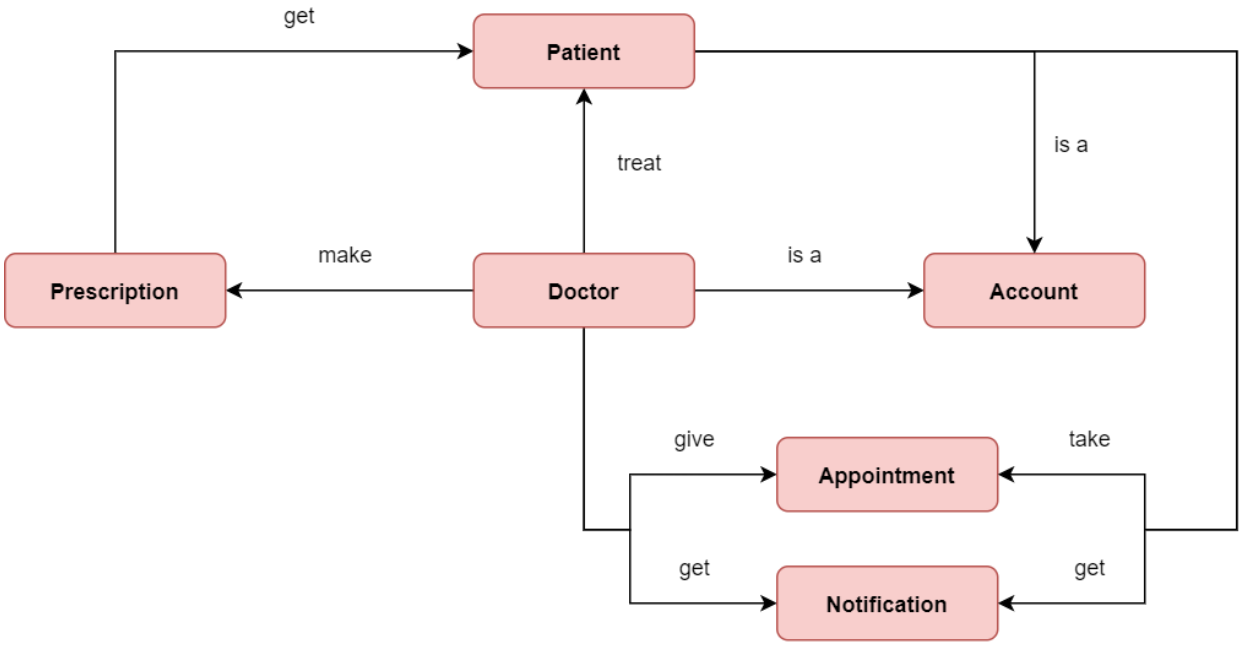


Figure: CRC Diagram for Doctor Class

ID : 5

Name : Staff

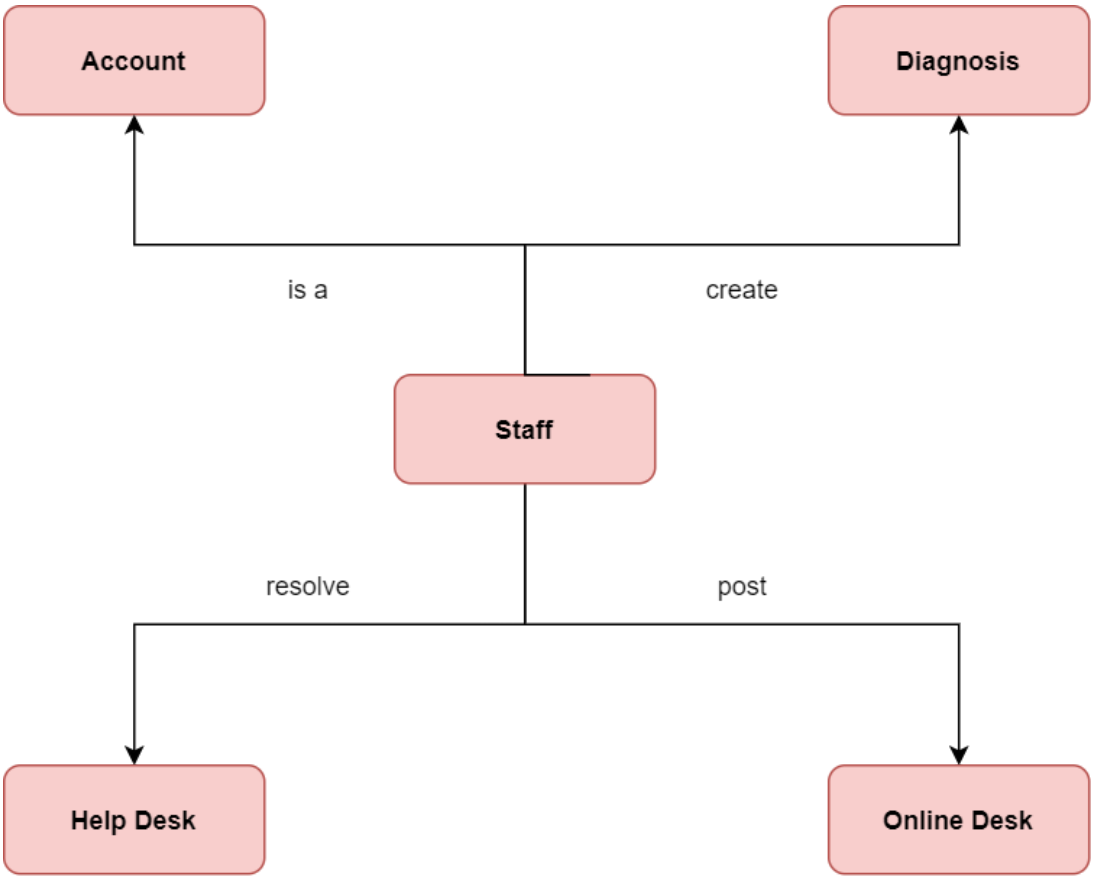


Figure: CRC Diagram for Staff Class

ID : 6

Name : Appointment

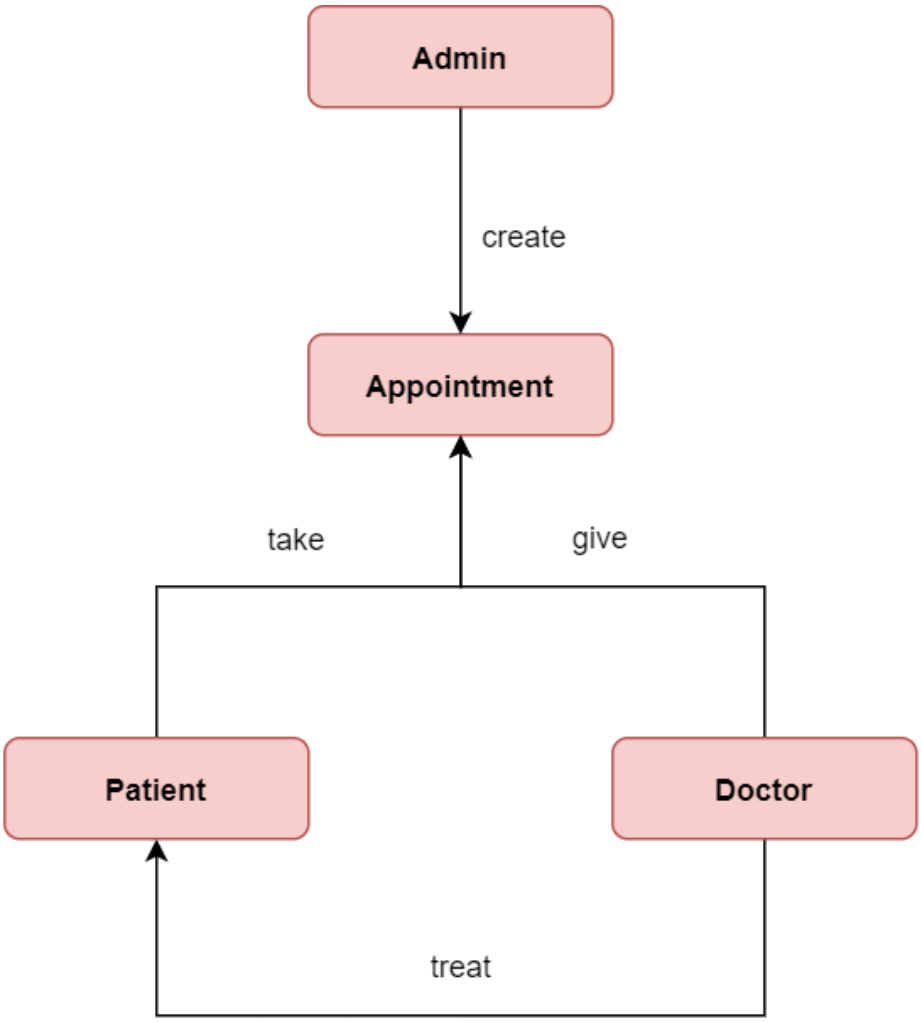


Figure: CRC Diagram for Appointment Class

ID : 7

Name : Payment

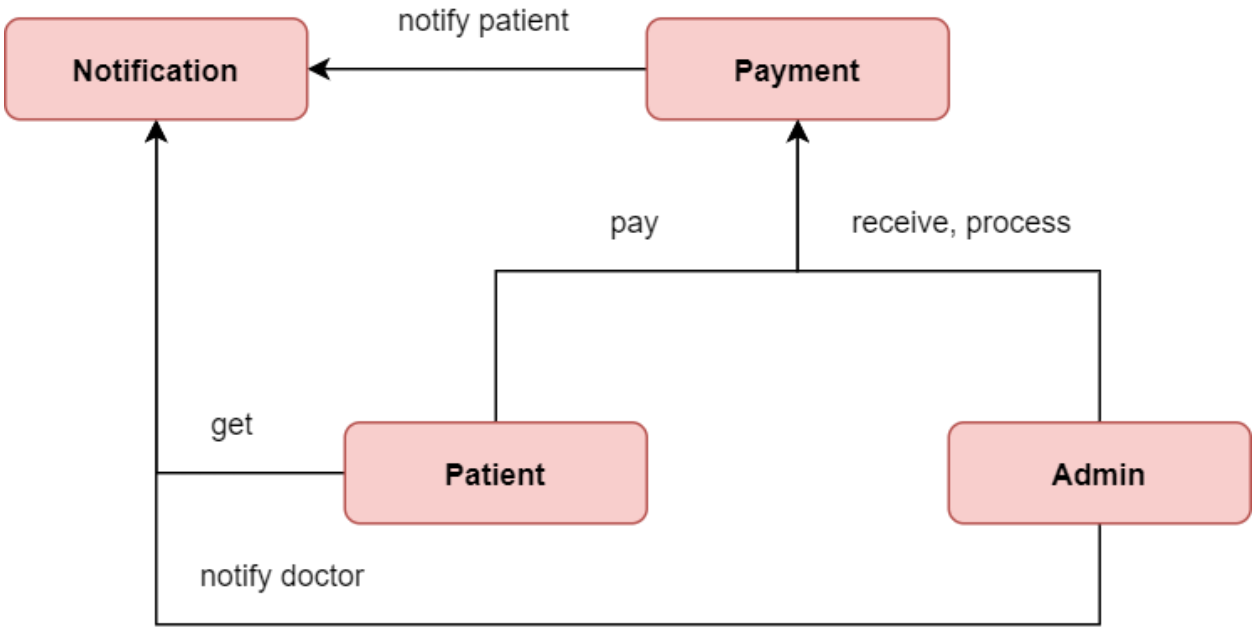


Figure: CRC Diagram for Payment Class

ID : 8

Name : Prescription

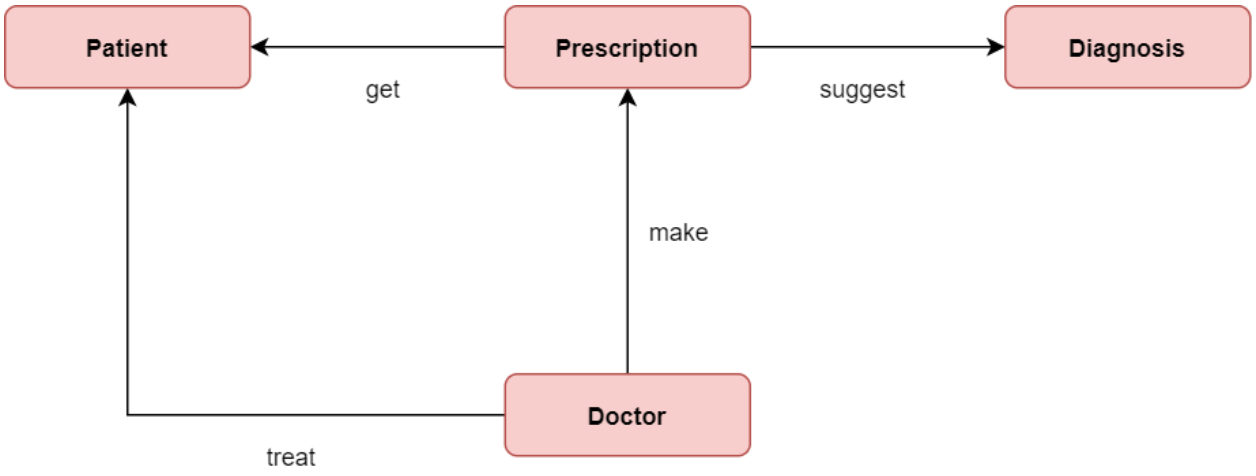


Figure: CRC Diagram for Prescription Class

ID : 9

Name : Diagnosis

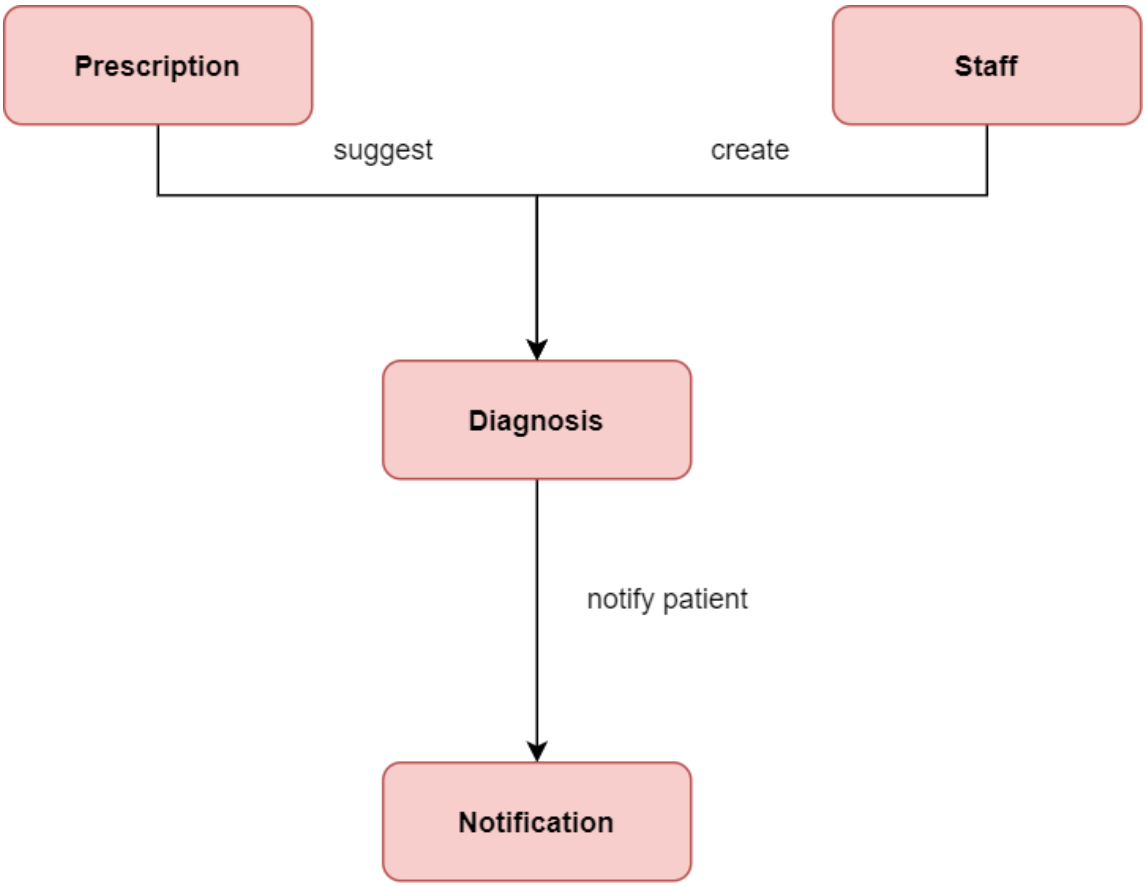


Figure: CRC Diagram for Diagnosis Class

ID : 10

Name : Notification

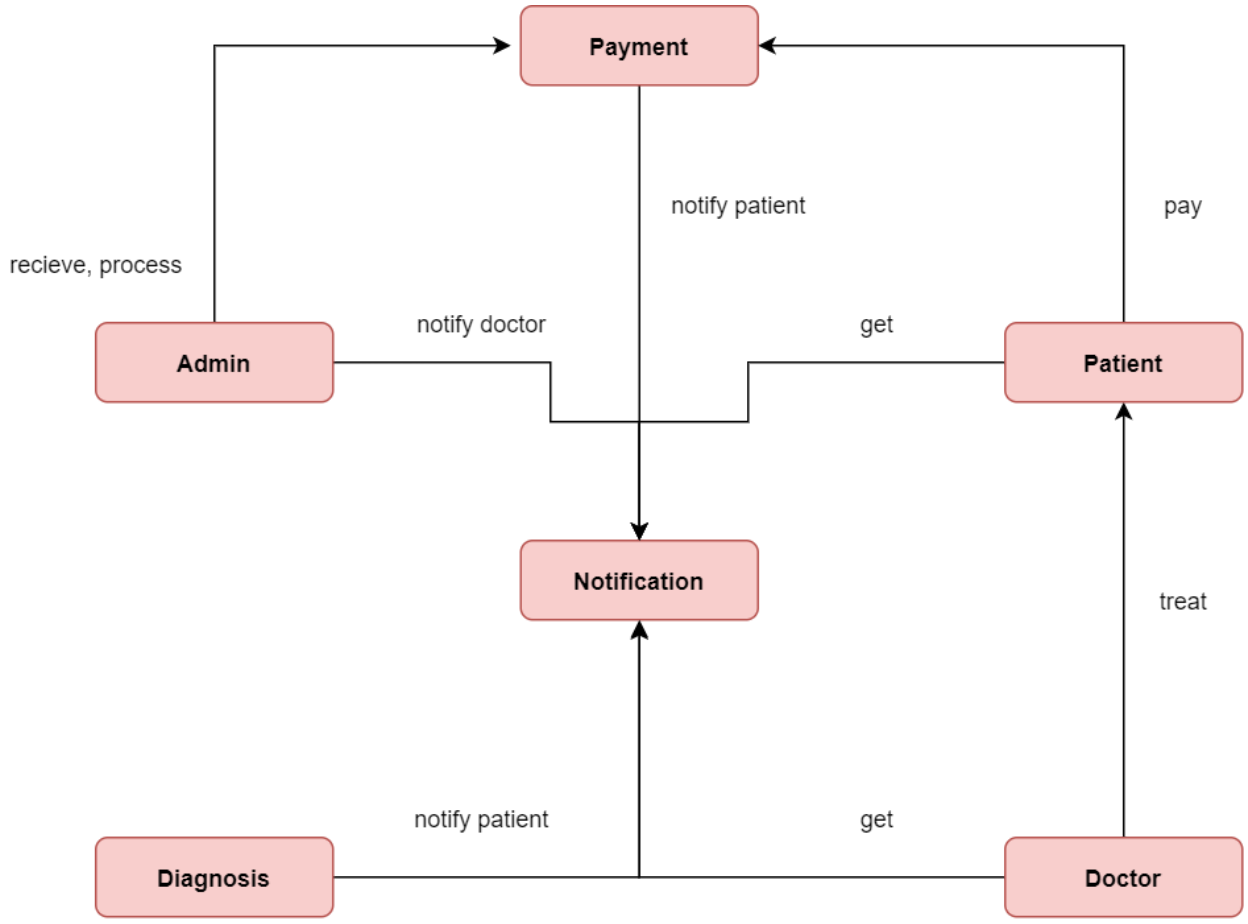


Figure : CRC Diagram for Notification Class

ID : 11

Name : Help Desk

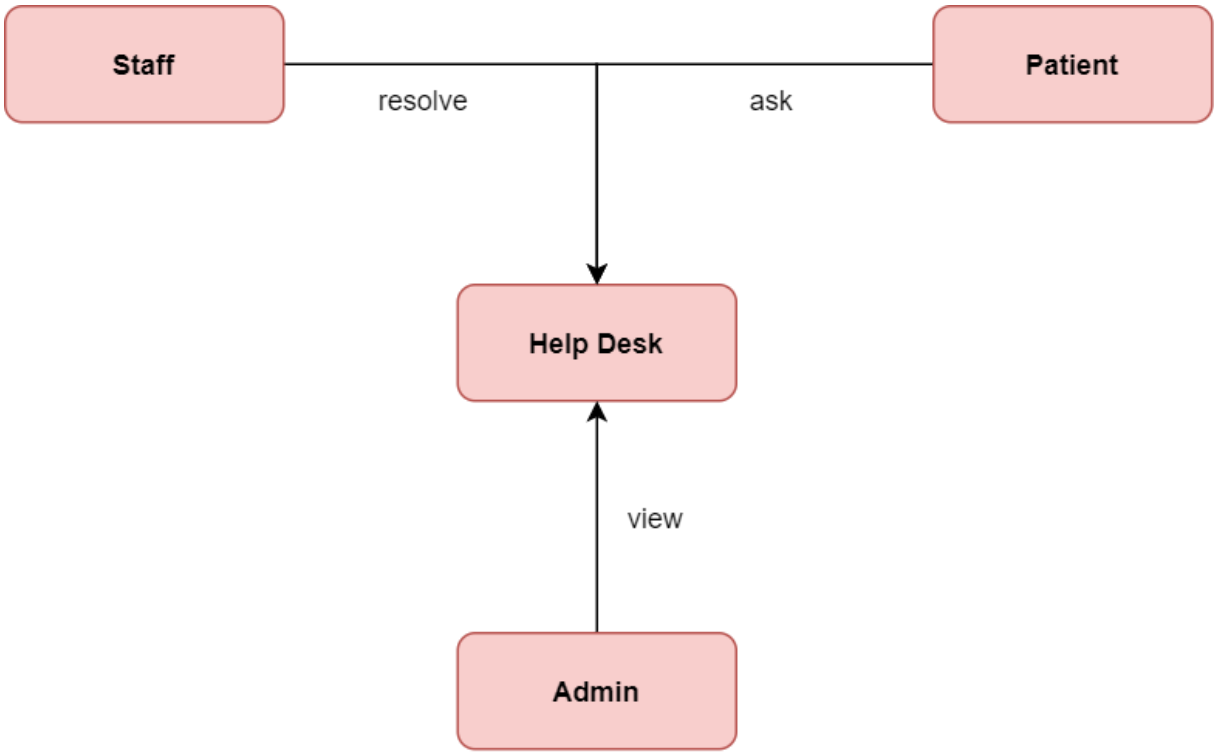


Figure: CRC Diagram for Help Desk Class

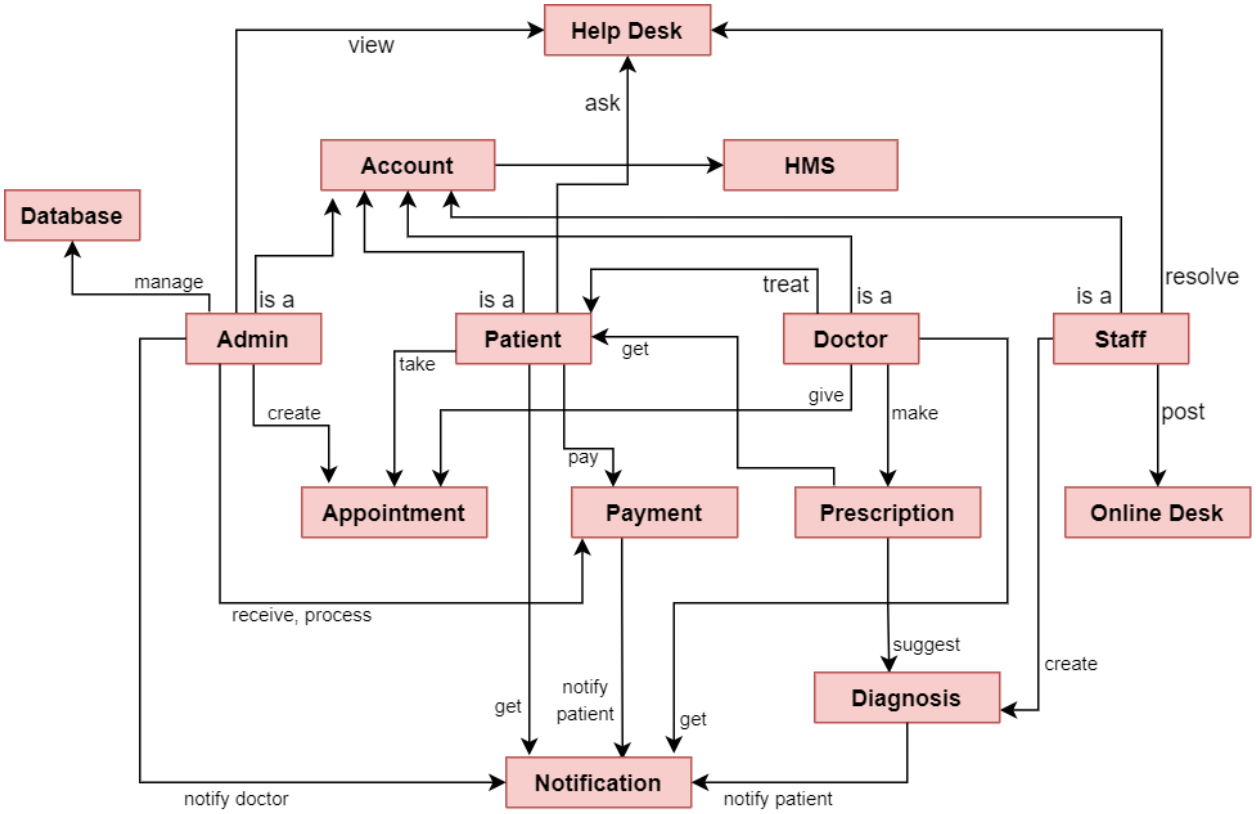


Figure: CRC Diagram for All Classes

Behaviour Modeling : Hospital Management System

The **Behaviour Modeling** indicates how the system will behave to external events or stimuli. It is represented as a function of time and event,

It describes interactions between objects. It shows how individual objects collaborate to achieve the behavior of the system as a whole. In UML behavior of a system is shown with the help of use case diagram, sequence diagram and activity diagram.

To create behavioral model following things can be considered-

- Evaluation of all use-cases to fully understand the sequence of interaction within the system.
- Identification of events that drive the interaction sequence and understand how these events relate to specific classes.
- Creating sequence for each use case.
- Building a state diagram for the system.
- Reviewing the behavioral model to verify accuracy and consistency.

State Transition Diagram

State Transition Diagram represents active states for each class of events (triggers). For this we identified all the events, their initiators and collaborators. In the **State Transition Diagram** the states are shown in boxed texts, and the transition is represented by arrows. It is also called State Chart or Graph. It is useful in identifying valid transitions.

In the state transition table all the states are listed on the left side, and the events are described on the top. Each cell in the table represents the state of the system after the event has occurred. It is also called the State Table. It is useful in identifying invalid transitions.

Table: Event Identification

Serial	Event	Event Name	Initiator	Collaborator
1	Add an admin	Add_admin	HMS	Admin, Account
2	Provide information	Provide_info	HMS	Admin
3	Verify credentials	Verify_info	HMS	Admin, Account
4	Create an account	Create_account	Account	Admin, Doctor, Patient, Staff
5	User login	Login	Account	Admin, Doctor, Patient, Staff
6	User recover account	Recover_account	Account	Admin, Doctor, Patient, Staff
7	User recover password	Recover_password	Account	Admin, Doctor, Patient, Staff
8	Add new doctor	Add_doctor	Admin	Doctor, Account
9	Add new Staff	Add_staff	Admin	Staff, Account
10	Verifying new doctor	Verify_doctor	Admin	Doctor, Account
11	Show doctors list	Doctor_info	Admin	Doctor
12	Receive payment from patient	Receive_payment	Admin	Patient
13	Set appointment for patient	Set_appointment	Admin	Patient, Doctor, Appointment
14	Send notification	Notify_users	Admin	Patient, Doctor, Notification
15	Processing payment	Process_payment	Admin	HMS, Doctor, Payment
16	Managing database	Manage_database	Admin	Database
17	Set schedule	Scheduling	HMS, Admin	Doctor, Patient, Appointment

Serial	Event	Event Name	Initiator	Collaborator
18	Select doctor	Select_doctor	Patient	Admin, Doctor
19	Show past history	View_history	Patient	Prescription
20	Request appointment	Get_appointment	Patient	Admin, Appointment
21	Show doctor's time	View_doctor	Patient	Admin, Doctor
22	Get token number	Get_token	Patient	Admin, Appointment
23	Cancel appointment	Cancel_appointment	Patient	Admin, Appointment
24	Request reschedule	Reschedule	Patient	Admin, Doctor, Appointment
25	Get bill information	Get_bill	Patient	Admin, Staff
26	Bill payment	Pay_bill	Patient	Admin, Payment
27	Request checkup	Checkup	Patient	Admin, Appointment, Staff
28	Request query	Get_help	Patient	Admin, Staff, Help_desk
30	Get prescription	Get_prescription	Patient	Doctor, Prescription
31	Treat patient	Treat_patient	Doctor	Patient, Prescription
32	Create prescription	Create_prescription	Doctor	Patient, Prescription
33	Show patient info	View_patient_info	Doctor	Patient
34	Post hospital news	Post_news	Staff	Online_desk
35	Resolve patient query	Resolve_query	Staff	Patient, Help_desk
36	Diagnosis patient	Create_diagnosis	Staff	Diagnosis, Staff, Patient,Prescription

Serial	Event	Event Name	Initiator	Collaborator
37	Create billing info	Create_bill	HMS	Diagnosis
38	Calculate slot	Allocate_slot	Appointment	Admin, Patient, Doctor
39	Update time	Update_time	Appointment	Admin
40	Show payment option	Pay_option	Payment	Patient
41	Show payment details	Pay_info	Payment	Patient, Admin
42	Show transaction history	View_transaction	Payment	Patient, Admin
43	Calculate refund	Calculate_refund	Payment	Patient, Admin
44	Show prescription	View_prescription	Prescription	Patient, Doctor, Staff, Diagnosis
45	Add medicine	Add_medicine	Prescription	Doctor
46	Add diagnosis	Add_diagnosis	Prescription	Doctor
47	Create report	Create_report	Diagnosis	Staff
48	Show reference	Show_reference	Diagnosis	Staff, Patient
49	Send report	Send_report	Diagnosis	Staff, Patient
50	Get body	Get_body	Notification	Account, Admin, Doctor, Patient, Staff, Diagnosis, Appointment, Payment
51	Set time	Set_time	Notification	Account, Admin, Doctor, Patient, Staff, Diagnosis, Appointment, Payment
52	Manage list	Manage_list	Database	Admin
53	Show information	View_info	Database	Admin
54	Manipulate data	Data_manipulation	Database	Admin

State Transition Diagram

ID : 1

Name : HMS

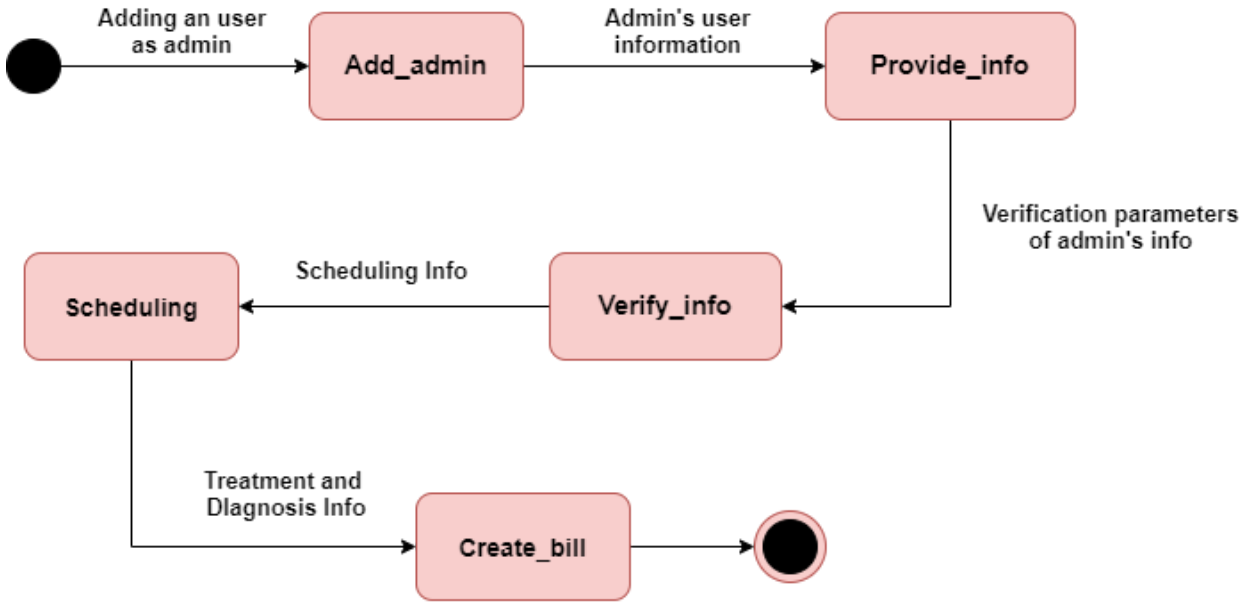


FIGURE : State Transition Diagram for HMS Class

ID : 2

Name : Account

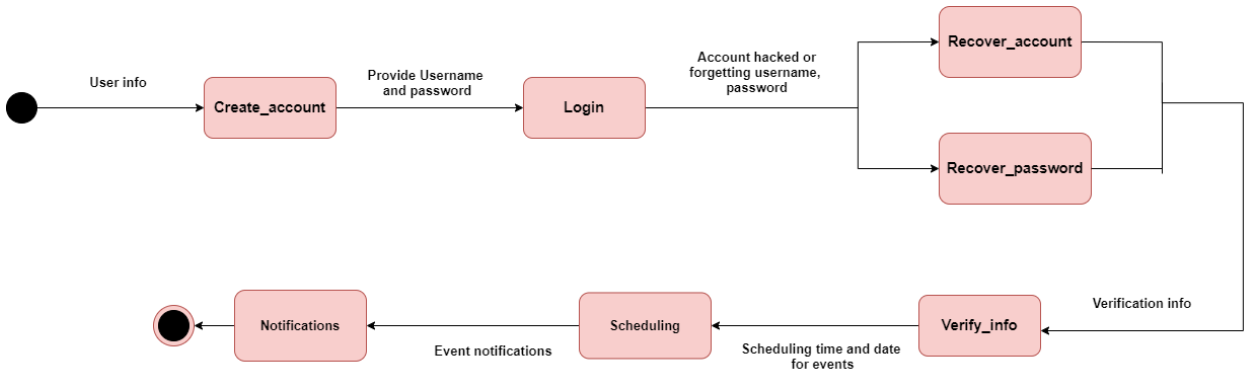


FIGURE : State Transition Diagram for Account Class

ID : 3

Name : Admin

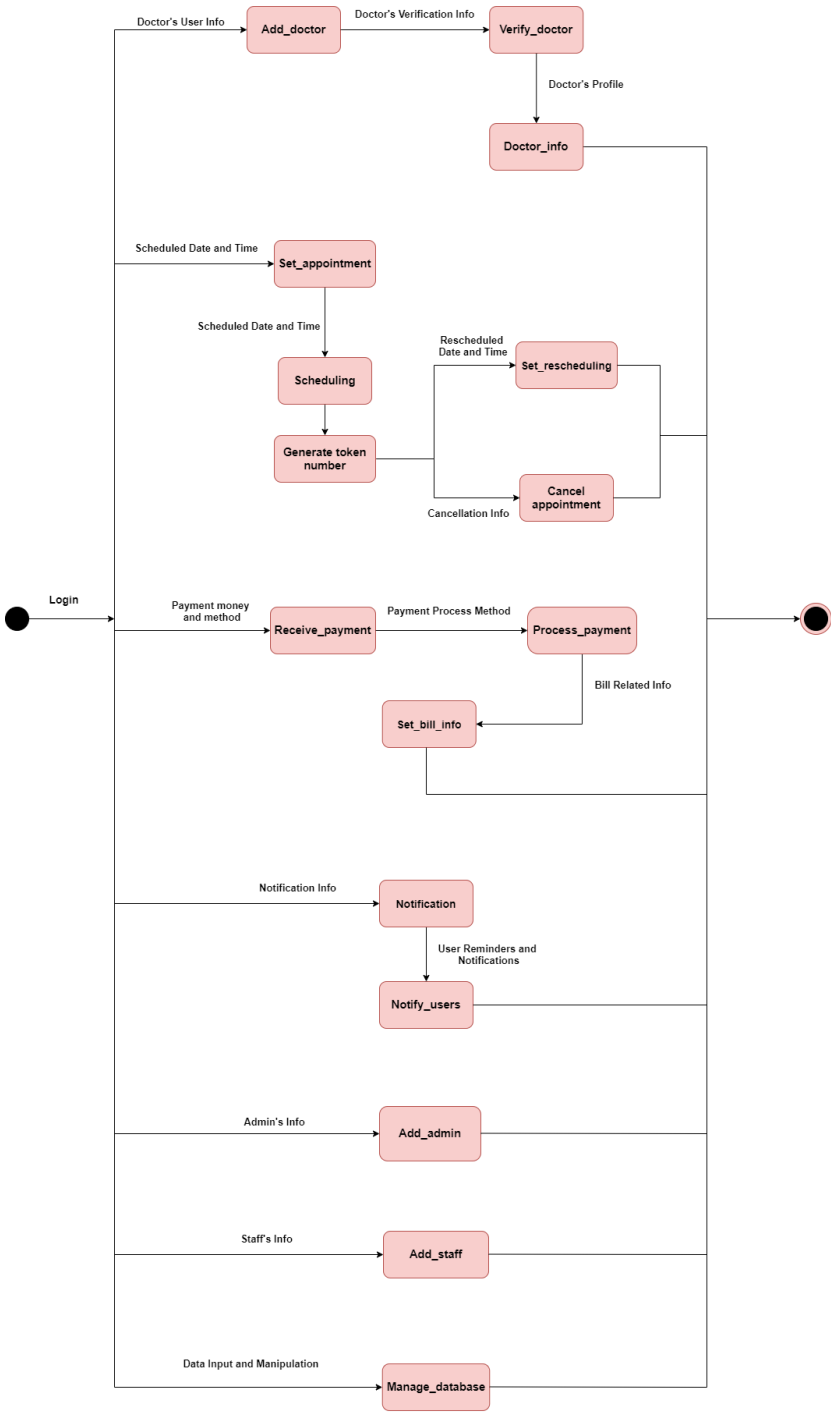


FIGURE : State Transition Diagram for Admin Class

ID : 4

Name : Patient

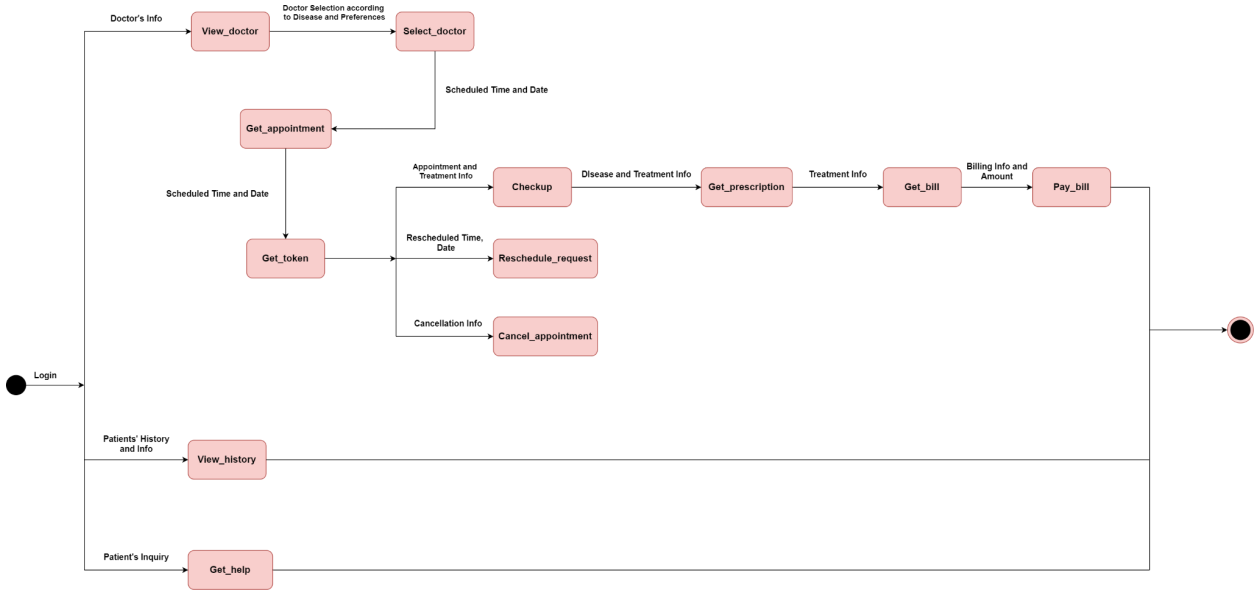


FIGURE : State Transition Diagram for Patient Class

ID : 5

Name : Doctor

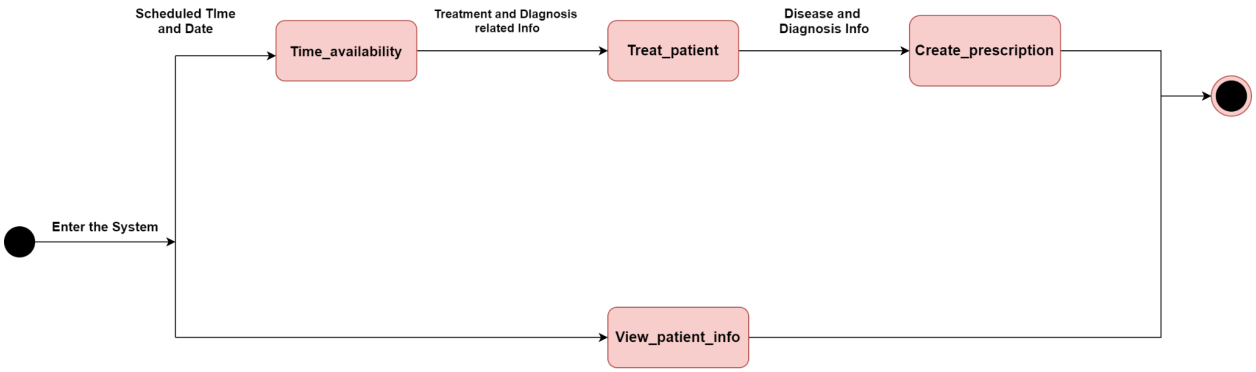


FIGURE : State Transition Diagram for Doctor Class

ID : 6

Name : Staff

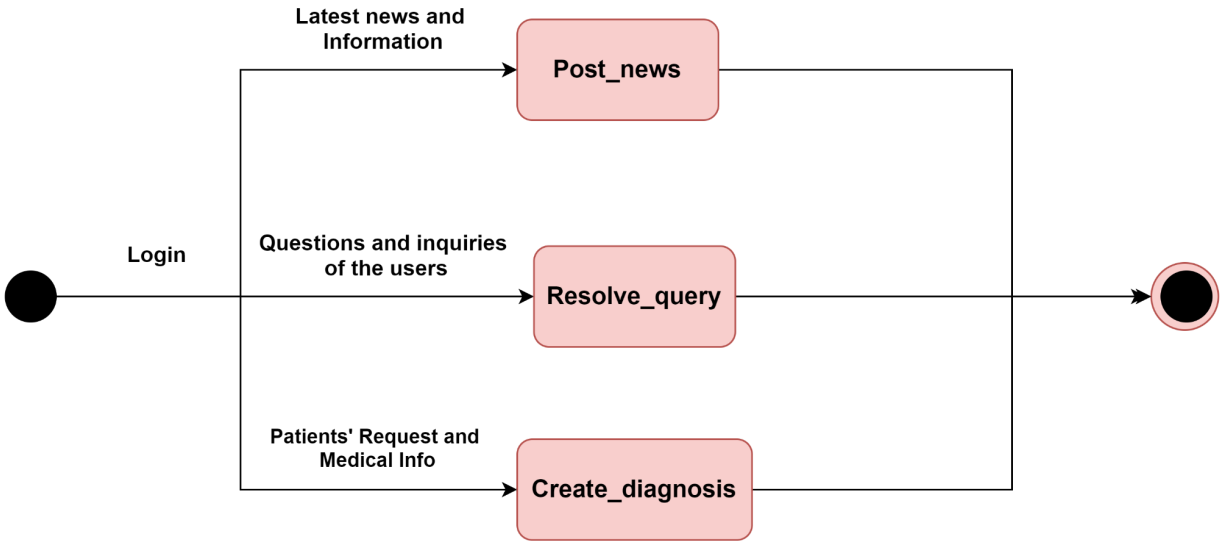


FIGURE : State Transition Diagram for Staff Class

ID : 7

Name : Appointment

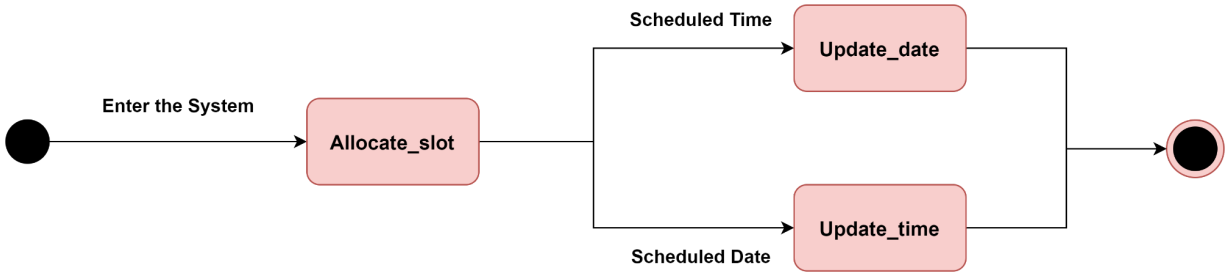


FIGURE : State Transition Diagram for Appointment Class

ID : 8

Name : Payment

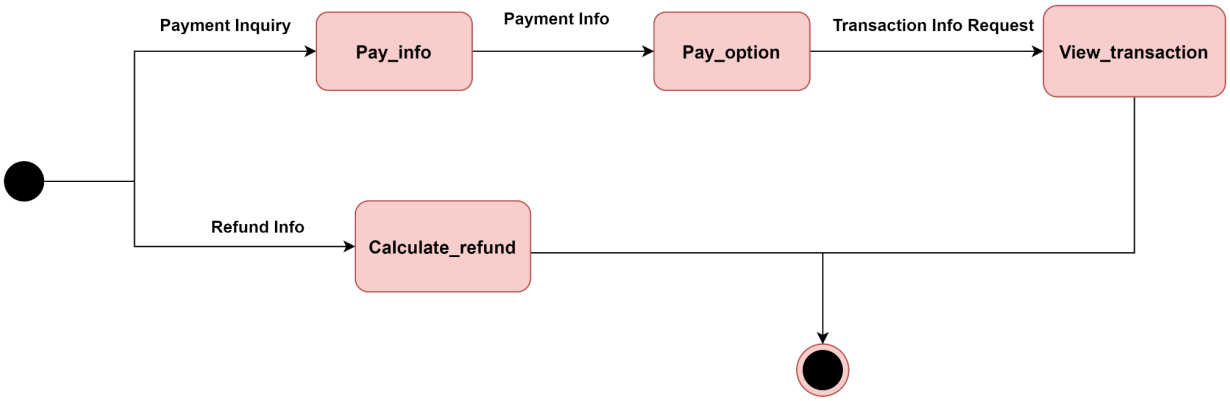


FIGURE : State Transition Diagram for Payment Class

ID : 9

Name : Prescription

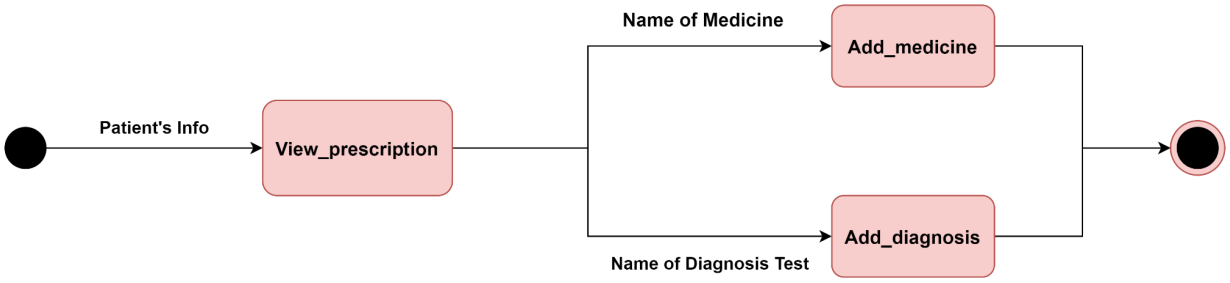


FIGURE : State Transition Diagram for Prescription Class

ID : 10

Name : Diagnosis

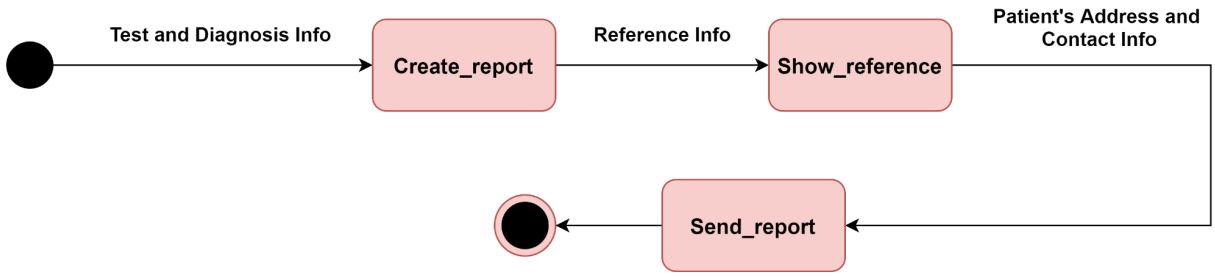


FIGURE : State Transition Diagram for Diagnosis Class

ID : 11

Name : Notification

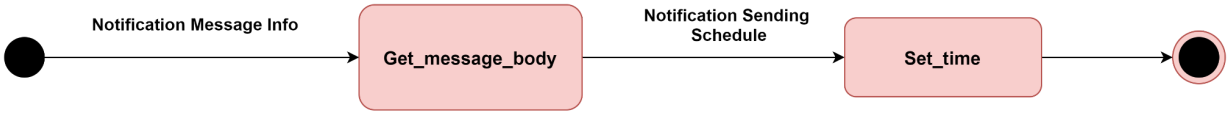


FIGURE : State Transition Diagram for Notification Class

ID : 12

Name : Database

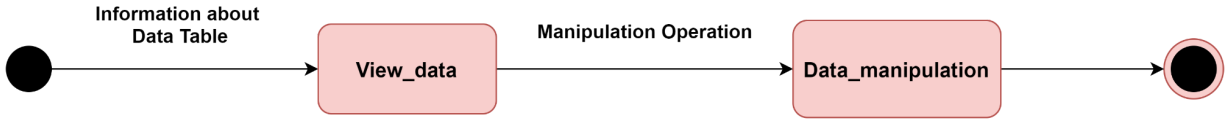


FIGURE : State Transition Diagram for Database Class

Sequence Diagram

Another type of representation in behaviour modeling is **Sequence Diagrams**.

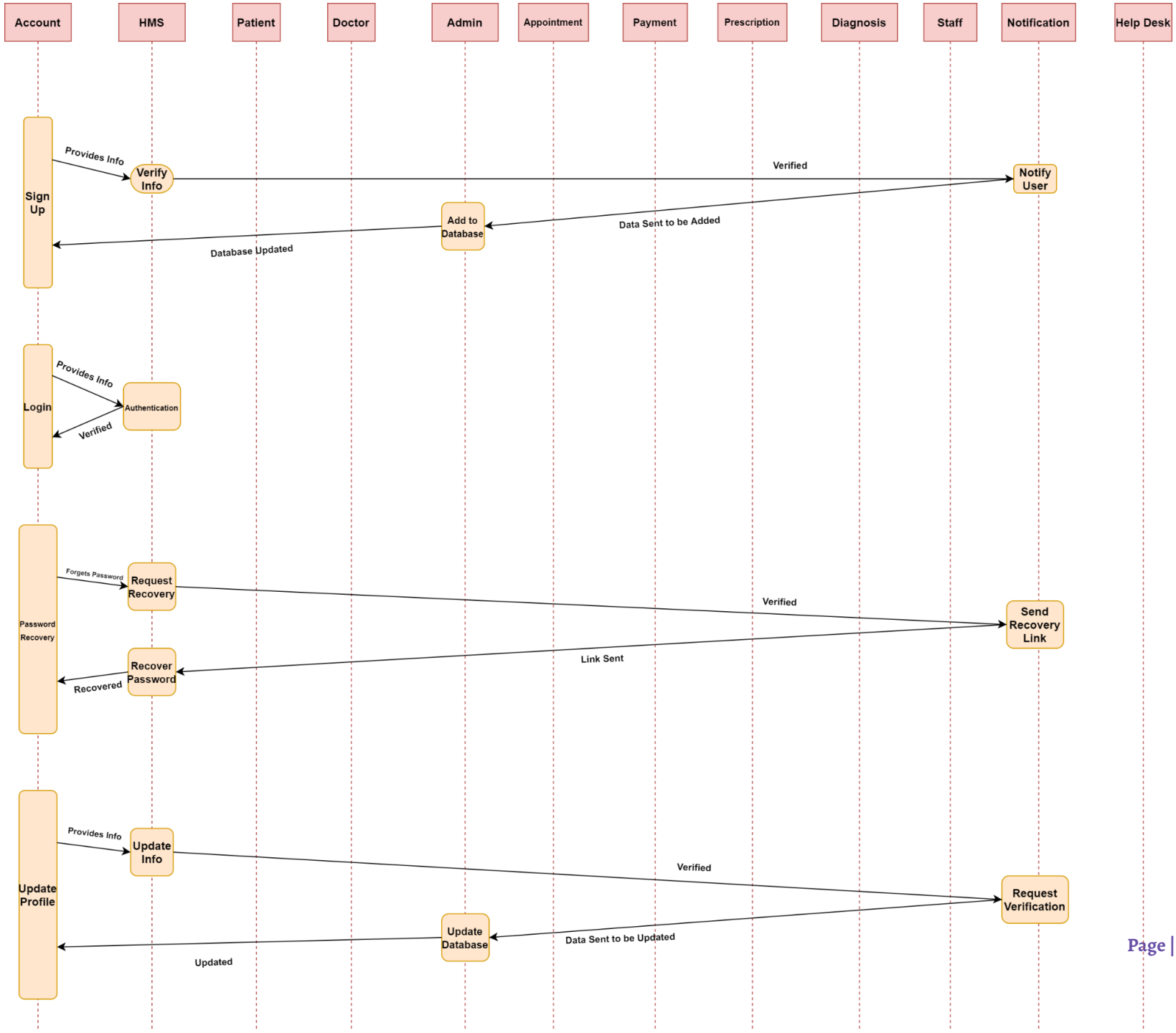
Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focused and they show the order of the interaction visually by using the vertical axis of the diagram to represent time, what messages are sent and when.

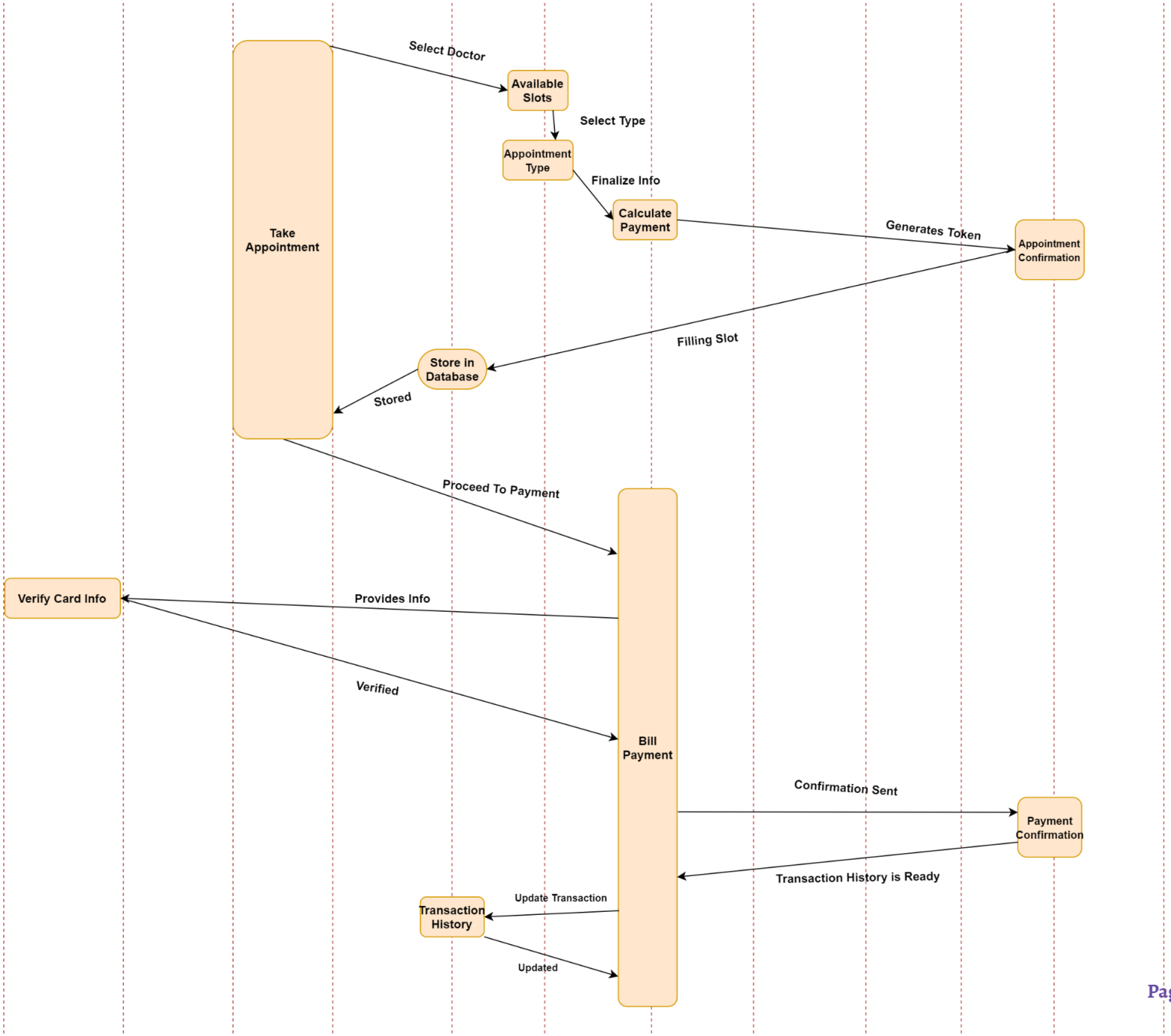
These diagrams are nothing more than a shorter version of the use case diagrams. It has objects structured horizontally and time vertically.

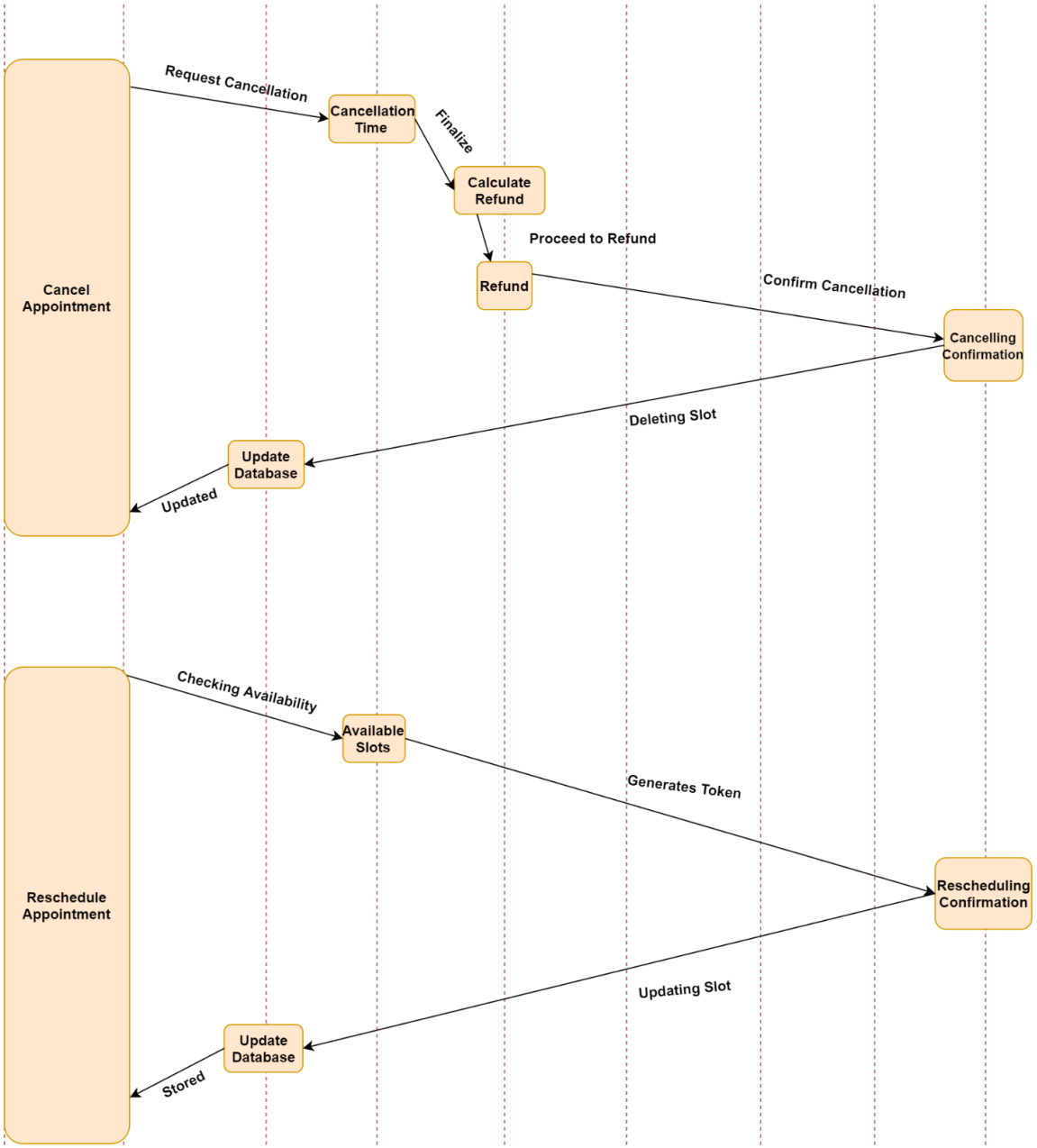
Purpose of Sequence Diagram

- High-level interaction between active objects in a system
- The interaction between object instances within a collaboration that realizes a use case
- The interaction between objects within a collaboration that realizes an operation
- Either model generic interactions (showing all possible paths through the interaction) or specific instances of a interaction (showing just one path through the interaction)

In our HMS, we have to show the interactions a user goes through while surfing the whole system from start to finish and organize them according to the segmentation of time. It can be useful to represent the details of UML use cases, to see how objects and components interact with each other to complete a process. Moreover, it'll help us to plan and understand the detailed functionality of the existing scenario that we're exploring.


Account
HMS
Patient
Doctor
Admin
Appointment
Payment
Prescription
Diagnosis
Staff
Notification
Help Desk
Sign Up
Provides Info
Verify Info
Verified
Notify User
Add to Database
Data Sent to be Added
Database Updated
Login
Provides Info
Authentication
Verified
Password Recovery
Forgets Password
Request Recovery
Verified
Send Recovery Link
Recover Password
Link Sent
Recovered
Update Profile
Provides Info
Update Info
Verified
Request Verification
Update Database
Data Sent to be Updated
Updated


Take Appointment
Select Doctor
Available Slots
Select Type
Appointment Type
Finalize Info
Calculate Payment
Generates Token
Appointment Confirmation
Filling Slot
Store in Database
Stored
Proceed To Payment
Verify Card Info
Provides Info
Verified
Bill Payment
Confirmation Sent
Payment Confirmation
Transaction History is Ready
Update Transaction
Transaction History
Updated


Cancel Appointment
Request Cancellation
Cancellation Time
Finalize
Calculate Refund
Proceed to Refund
Refund
Confirm Cancellation
Cancelling Confirmation
Deleting Slot
Update Database
Updated
Reschedule Appointment
Checking Availability
Available Slots
Generates Token
Rescheduling Confirmation
Updating Slot
Update Database
Stored

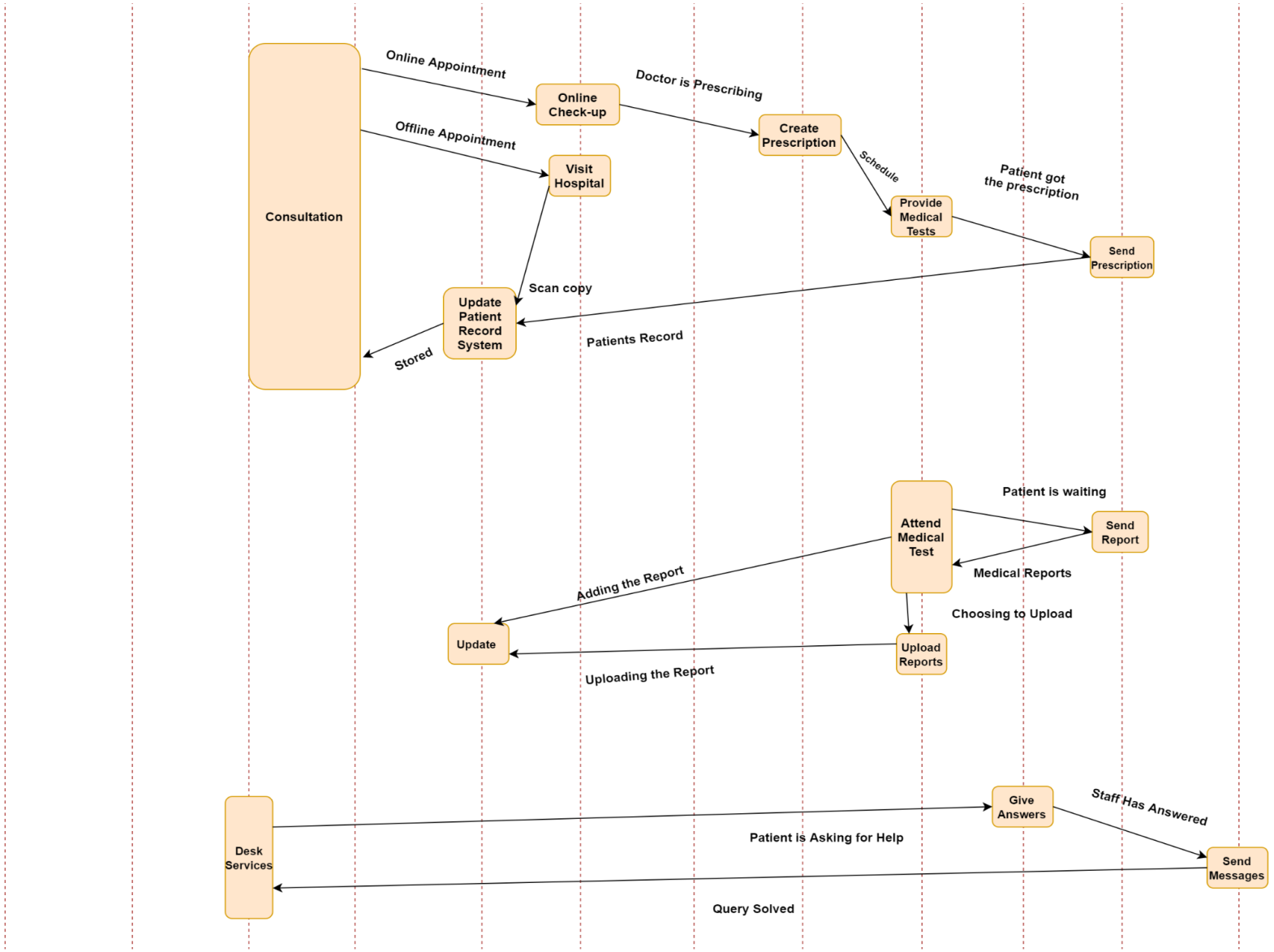


FIGURE : Sequence Diagram