HEALTH CARE MANAGEMENT

Software Documentation

Members:

Muhammad Muneeb Zaidi

Aliyan Ul Haq

Ajlal Haider

Sayyan Abubakkar

Project Statement:

The Health Centre Management System (HCMS) is a software project aimed at improving the efficiency and effectiveness of healthcare facilities by implementing a comprehensive and user-friendly system for managing various aspects of health center operations. This project will follow the Software Development Life Cycle (SDLC) to ensure the successful development, deployment, and maintenance of the HCMS.

Objectives:

- **Streamline Patient Management:** Develop a patient management system that enables efficient registration, appointment scheduling, and medical record management.
- **Enhance Staff Productivity:** Implement features to manage healthcare professionals' schedules, streamline communication, and automate administrative tasks.
- **Optimize Resource Allocation:** Enable effective management of medical equipment, supplies, and facility resources.
- **Ensure Data Security and Privacy:** Implement robust security measures to protect patient data and comply with healthcare data privacy regulations.
- **Improve Reporting and Analytics:** Develop reporting and analytics capabilities to help health center administrators make data-driven decisions.
- **Support Accessibility and User-Friendliness:** Design an intuitive user interface accessible to both healthcare professionals and patients.

Key Stakeholders:

- Health Center Administrators
- Working Staff
- Doctors
- Nurses
- Patients
- Finance Department
- Project Team (Developers, Testers, Designers)
- IT Support Team

Requirements Gathering:

Techniques used to gather requirements for this project are given below:

- ***Conducted interviews:*** Scheduled interviews with the stakeholders to discuss their requirements and understand their pain points. This was done in person, over the phone, or through video conferencing.

- ***Performed observations:*** Observed the workflow and processes in the health center to identify areas that needed improvement. This was done by shadowing staff and observing patient interactions.

- ***Used surveys:*** Developed and distributed surveys to patients, healthcare providers, and administrative staff to gather feedback on their experiences and expectations.

- ***Conducted focus groups:*** Organized focus groups with representatives from each stakeholder group to discuss their requirements and expectations in a group setting.

- ***Analyzed existing documentation:*** Reviewed existing documentation such as policies, procedures, and regulations to identify requirements and constraints that needed to be considered in the design of the health center management system

Project Deliverables:

- Comprehensive Health Centre Management System
- User Manuals and Training Materials
- Documentation of SDLC Processes
- System Maintenance Plan
- Data Security and Privacy Compliance Report

Success Criteria:

The success of the HCMS project will be measured by its ability to streamline health center operations, enhance patient care, and ensure data security and privacy. Key performance indicators include improved appointment scheduling, resource optimization, reduction in administrative overhead, and compliance with relevant healthcare data regulations.

Budget and Resources:

A budget of is Rs 25,000,000/- is allocated for this project. Resources will include the project team members, hardware, software, and any external consultants or services required.

Project Risks:

- Data security breaches
- Integration challenges with existing systems
- Resistance to change from health center staff
- Regulatory compliance issues

High Level Requirements:

1. **User Authentication and Access Control:**
    - The system must provide secure user authentication to ensure that only authorized individuals can access and modify patient data and system settings.
2. **Patient Registration and Profile Management:**
    - The HCMS should allow for the efficient registration of patients, including capturing personal information, medical history, and contact details.
    - Patients must be able to create and manage their profiles, including updating contact information and medical records.
3. **Appointment Scheduling:**
    - Health center staff should be able to schedule and manage patient appointments, including setting availability, rescheduling, and cancellations.
    - Patients must have the capability to request appointments and receive confirmation notifications.
4. **Medical Record Management:**
    - The system must enable healthcare professionals to create and maintain electronic health records (EHRs) for patients, including diagnoses, treatment plans, prescriptions, and test results.

5. **Staff Scheduling and Task Management:**
    - The HCMS should support the scheduling and assignment of tasks to healthcare professionals (doctors, nurses, etc.) based on their availability and expertise.
    - Staff members should have access to their schedules and task lists.
6. **Resource Allocation:**
    - The system must allow for the efficient management of medical equipment, supplies, and facility resources to ensure their availability for patient care.
7. **Reporting and Analytics:**
    - The HCMS should provide reporting and analytics capabilities, allowing administrators to generate reports on patient demographics, appointment statistics, and resource utilization.
    - The system must support data visualization to aid decision-making.
8. **Data Security and Privacy:**
    - The system must adhere to industry standards and regulations to ensure the security and privacy of patient data. This includes encryption, role-based access control, and compliance with healthcare data protection laws (e.g., HIPAA).
9. **User-Friendly Interface:**
    - The user interface should be intuitive, user-friendly, and accessible to both healthcare professionals and patients. It should support multiple languages if necessary.
10. **Integration with Existing Systems:**
    - The HCMS should be designed to integrate seamlessly with existing healthcare management systems, electronic health record (EHR) systems, and external data sources to facilitate data sharing and coordination.
11. **Scalability and Performance:**
    - The system should be scalable to accommodate the growth of the health center and ensure optimal performance even under increased usage.

12. **Compliance and Regulatory Requirements:**
    - The HCMS must comply with all relevant healthcare industry regulations and standards to avoid legal issues and potential penalties.
13. **Training and Support:**
    - The project should include training materials and support mechanisms to ensure that health center staff and users can effectively use the system.

Functional Requirements:

- The system should manage patient information, such as demographics, medical history, and treatment plans.
- The system should manage patient appointments, allowing for scheduling, rescheduling, appointment history viewing, and appointment reminders.
- The system should be able to maintain the financial record of patients and the staff.
- The system should provide reporting and analytics capabilities, such as generating reports on patient outcomes, resource utilization, and financial performance.
- The system should manage staff information, such as schedules, roles, and performance evaluations.
- The system should manage prescriptions, such as prescribing medications, tracking medication history, and sending prescription orders to pharmacies.
- The system should manage medical supplies and equipment inventory, including tracking stock levels and ordering supplies.
- The system should provide a patient portal where patients can access their medical records, view test results, and communicate with healthcare providers.

Non-Functional Requirements:

- The system must be user-friendly and easy to learn and use, with a clear and intuitive user interface and efficient workflows.
- The system must be reliable and available, with high system uptime and minimal errors or crashes.

- The system must be responsive and perform efficiently, with fast loading times and minimal downtime.
- The system must ensure the confidentiality, integrity, and availability of patient information, as well as protect against unauthorized access, data breaches, and cyber-attacks.
- The system must be cost-effective and provide a good return on investment, with reasonable pricing and minimal maintenance costs.

Time:

The project will be completed in 1 and a half years (18 months). Starting from 1st April, 2023 to the day of delivery, 30th September, 2023. In case of non-compliance with the agreed period, penalties will be imposed.

Milestones:

- **Requirement Gathering Phase:** First 2 months.
- **Analysis and Design phase:** 3 months (3rd to 5th month)
- **Development phase:** 7 months (6th to 12th month)
- **User testing phase:** 3 months (13th to 15th month)
- **System training and support phase:** 1 months (16th month)
- **Implementation and Rollout phase** 2 months **(**17th to 18th month)

Assumptions and Constraints:

Assumptions:

- The project team assumes that neither the project's requirements nor its scope will be significantly altered.
- The project team assumes that all stakeholders will cooperate with the team and be available to attend any necessary project meetings and reviews.
- The team working on the project is counting on the availability, compatibility, and long-term viability of the necessary software and hardware components.
- The project team is under the impression that the Health Center’s policies and regulations will not be significantly altered, which could influence the system.

- The project team assumes that all users and stakeholders are knowledgeable, that dealing with them will be simple, and that they will quickly comprehend the online system.
- The project timeline may be impacted by unforeseen events, such as natural disasters or pandemics.
- The project budget may be impacted by changes in market conditions or unexpected costs.

Constraints:

- The developed system will strictly meet the specified requirements of the Health Centre.
- The system cost will not exceed the provided budget.
- The Project must be finished in the given Time. If the team fails to complete the project, then the Owners/stakeholders can file a case against the Team because they have the right to do so.
- The project is limited by the software and technologies that are already in use; we rely on these components, which are currently on the market or in the company.

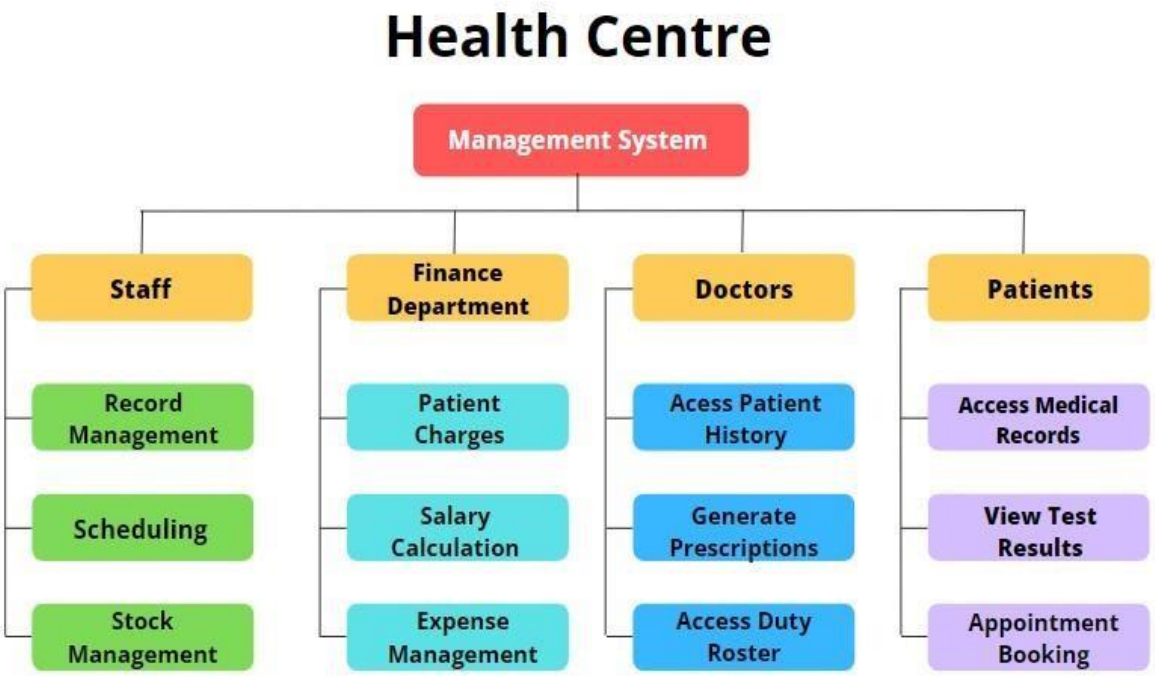
Terms and Conditions:

- The project will be able to function properly on most architectural hardware.
- The project will be completely tested and debugged before delivery.
- Health Center will hold all ownership rights of the developed system.
- The software company will be responsible for the implementation process of the software.
- The software company will offer a 12-month warranty, starting from the day when the software is delivered. The project team will provide technical support and resolve any system issues during this time.
- The software company will comply with all project specifications, timelines, and budgets as outlined in the project charter.
- Payment will be made in monthly installments of Rs. 10, 00,000/- per month.
- In case of any natural disaster both Health Center and Software Company will deal with the loss equally.
- The software company will provide regular progress updates to the Health Center.

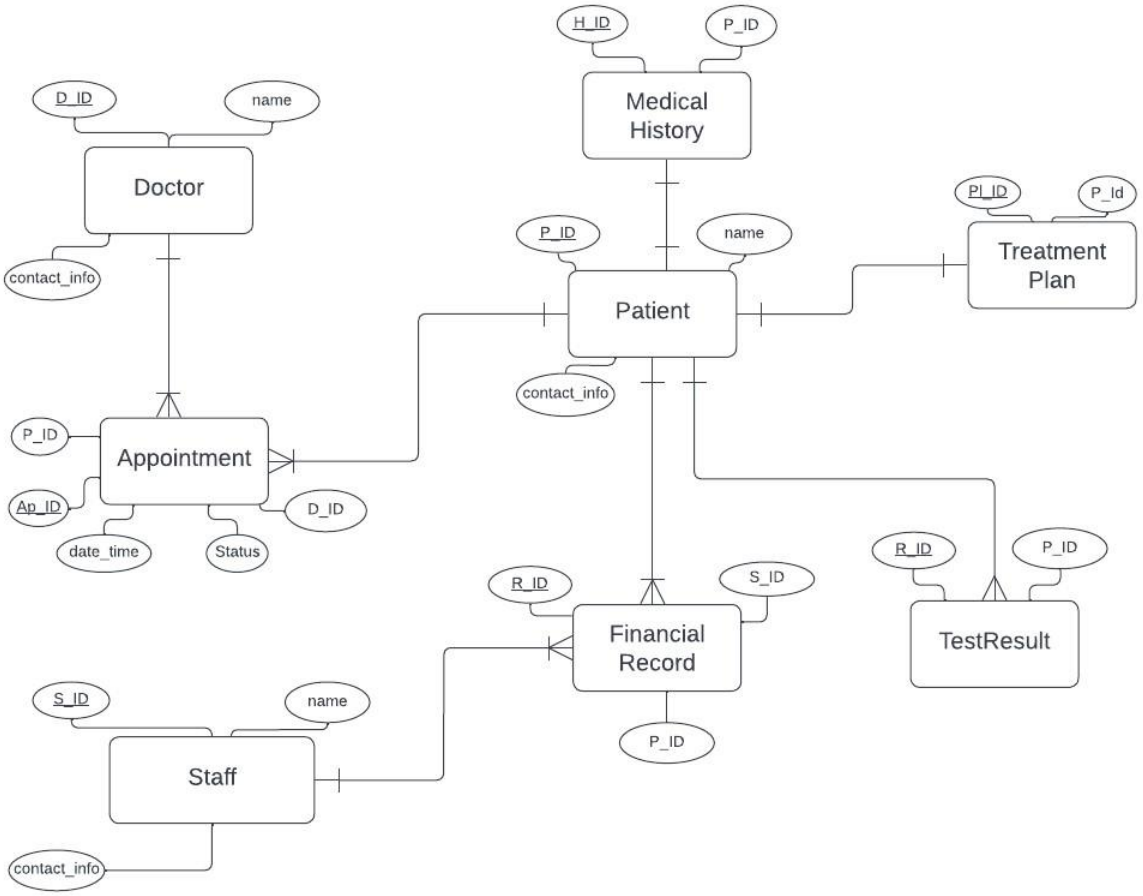
SDLC Approach:

For the development of the Health Centre Management System, we adopted the Agile Software Development Life Cycle (SDLC) approach. Agile was chosen for its iterative and flexible nature, allowing us to adapt to evolving requirements and deliver incremental updates. This approach was particularly suitable for our project due to its emphasis on collaboration, customer feedback, and the ability to quickly respond to changing needs in the healthcare environment. The Agile methodology enabled us to deliver a functional and user-friendly system while ensuring that stakeholders' requirements were continuously met throughout the development process. This approach facilitated regular interactions with stakeholders, ensuring their active involvement in shaping the system's functionality. Additionally, Agile's iterative cycles allowed for early testing and validation, leading to a more robust and reliable Health Centre Management System.

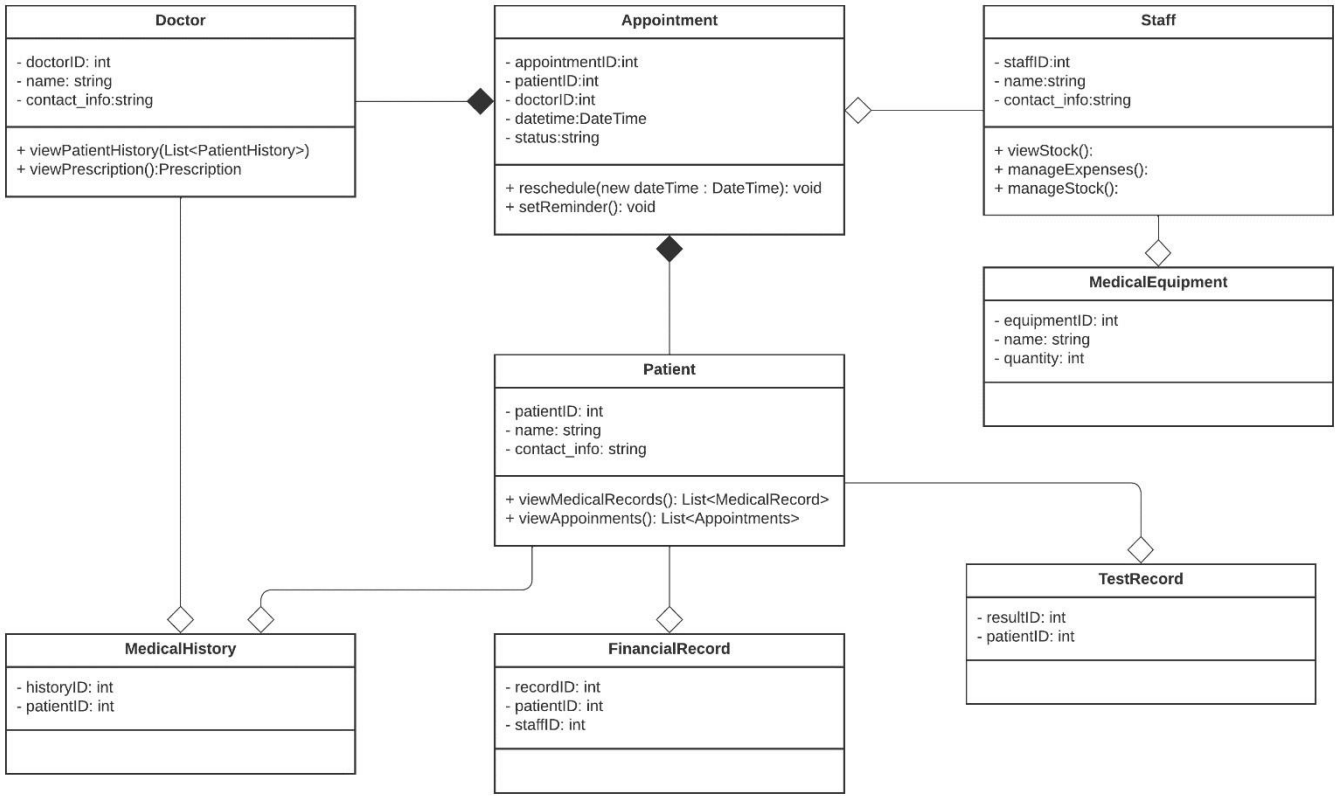
Work Breakdown Structure:



ENTITY-RELATIONSHIP DIAGRAM (ERD):



CLASS DIAGRAM:



Traceability Matrix:

RID	SID	Priority	Use Case	Description	Objective	Work Package	UID	Class Name	Act ID	Test Case
R-01	Receptionist	1	Register a Patient	To keep record of all the patients.	1	Patient Information Management	UC-1.1.1	Patient		
R-02	Doctor	1	Update Medical History	To make it easier for the doctor to diagnose the same patient in future.	0	Patient Information Management	UC-1.2.2	Patient		
R-03	Admin	2	Update Staff Roster	To keep track of staff duties	1	Staff Management	UC-1.3.2	Staff		
R-04	Finance Department	1	Generate Patient Expenses	To calculate equipment and service charges spent on a patient	1	Accounts Management	UC-1.4.1	Accounts		
R-05	Finance Department	1	Generate Staff Salary	To calculate total staff salary according to the number of duties they performed	1	Accounts Management	UC-1.4.3	Accounts		

Use Cases:

UC: <1.1.1>

Priority: 1

Actors: Receptionist, Patient

Use Case Summary: Lets the Receptionist create a record for every new patient so that all their data is stored.

Pre-condition: Patient or their attendant must provide the patient's details to the receptionist.

Scenarios:

1. A new patient comes to the health center.
2. All data regarding the patient is provided to the receptionist.
3. Receptionist records the data in the system.
4. The system asks for confirmation if the user wants to confirm entry.
5. The receptionist confirms the entry.

Alternate Scenarios:

1. The receptionist does not confirm entry.

Exception:

1. The same patient cannot be registered twice.

Post condition:

1. A new patient has now been registered in the system.

UC: <1.2.2>

Priority: 1

Actors: Doctor, Patient

Use Case Summary: Allows the doctor to update patient's medical history so that if the same patient comes back again then this history would help in their diagnosis.

Pre-condition: The registered patient is diagnosed by the doctor.

Scenarios:

1. The doctor logs onto the system and accesses patients' records.
2. In the medical history section, the doctor creates a new entry.
3. The doctor updates the medical history with the new diagnosis.
4. The doctor presses "ok" and the system asks for confirmation.
5. The doctor confirms the entry.

Alternate Scenarios:

1. The doctor does not confirm the entry.

Exception:

1. The doctor records a blank or incomplete entry.

Post-condition:

1. The medical history of the patient is updated.

UC: <1.3.2>

Priority: 2

Actors: Admin, Staff

Use Case Summary: This will allow the admin to create a new roster for upcoming week and allot duties to the staff members decided by the admin.

Pre-Condition: User is logged on to the system as an admin.

Scenarios:

1. Admin creates a new roster entry.
2. Admin allots duties to the staff according to time and availability.
3. System asks for confirmation.
4. Admin confirms the roster.

Alternate Scenarios:

1. Admin does not confirm the roster.

Exception:

1. The system will not allow the admin to allot the same duty to more than one staff member.

Post Condition:

1. A new roster is created for the following week.

UC: <1.4.1>

Priority: 1

Actors: Finance Department

Use Case Summary: This use case allows the finance department to calculate all the expenses of a patient scaling from equipment used on them, service fee to medication charges.

Pre-Condition: The registered patient is discharged/diagnosed.

Scenarios:

1. The finance department enters the gathered expenses data into the system.
2. System asks for confirmation.
3. The finance department confirms the entry.

4. The system then double checks the entered data.
5. The system sums up the total expenses and calculates discounts if any.

Alternate Scenarios:

1. The finance department does not confirm the entry.

Exception:

1. The system halts the billing process if the same expense is added twice.

Post Condition:

1. A receipt containing the expenses of the patient is generated.

UC: <1.4.2>

Priority: 1

Actors: Finance Department

Use Case Summary: This use case allows the finance department to calculate the staff salary according to the number of duties they performed.

Pre-Condition: It is the end of the month.

Scenarios:

1. The finance department selects the staff person whose salary must be calculated.
2. The system asks for confirmation.
3. The finance department confirms its choice.
4. The system then sums up the salary according to the number of duties that staff person performed and deducts miscellaneous fines.

Alternate Scenarios:

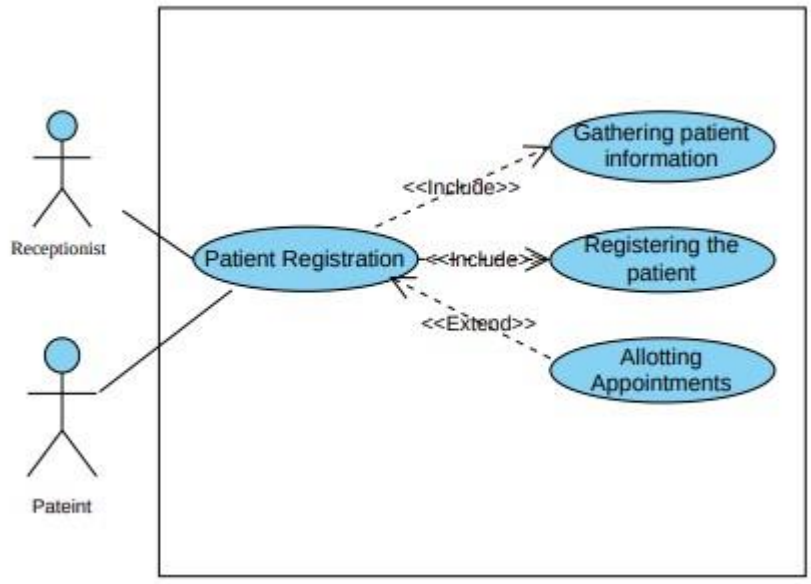
1. The finance department does not confirm the entry.

Post Condition:

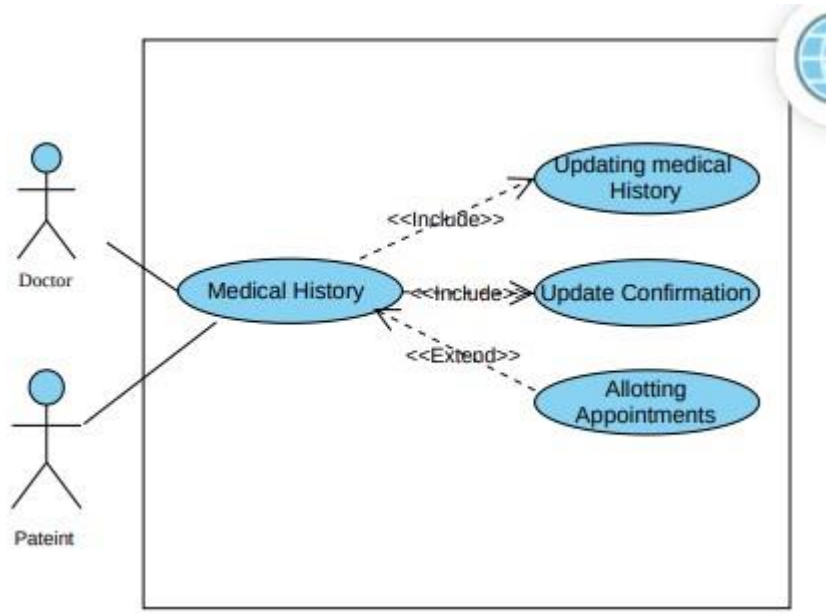
A salary slip of the staff person is generated by the system.

USE CASE DIAGRAM:

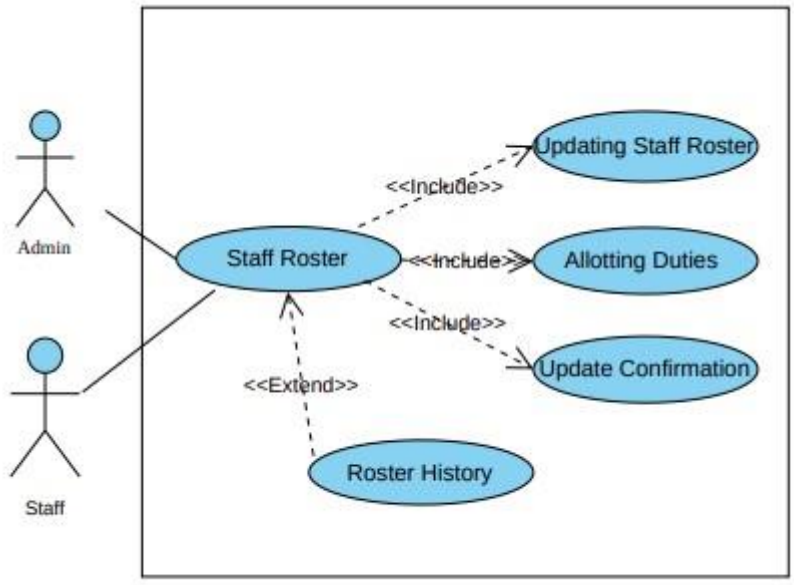
<1.1.1> Register a Patient:



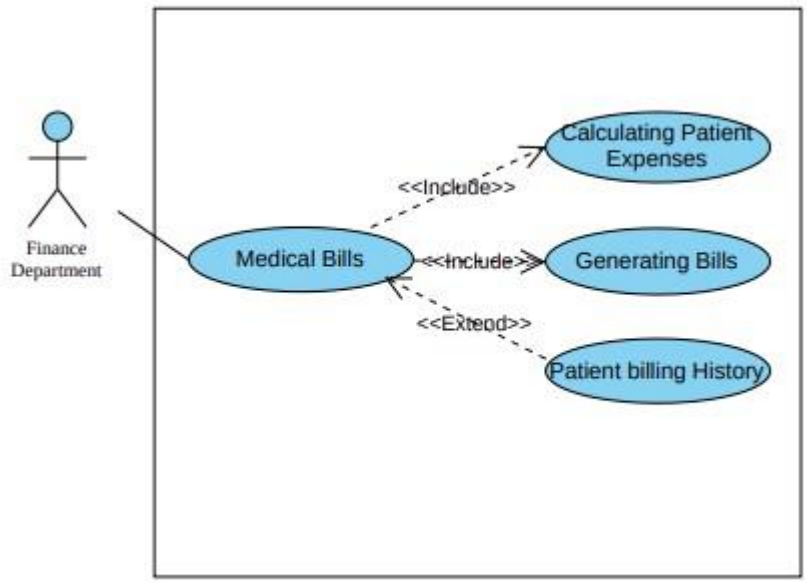
<1.2.2> Update Medical History:



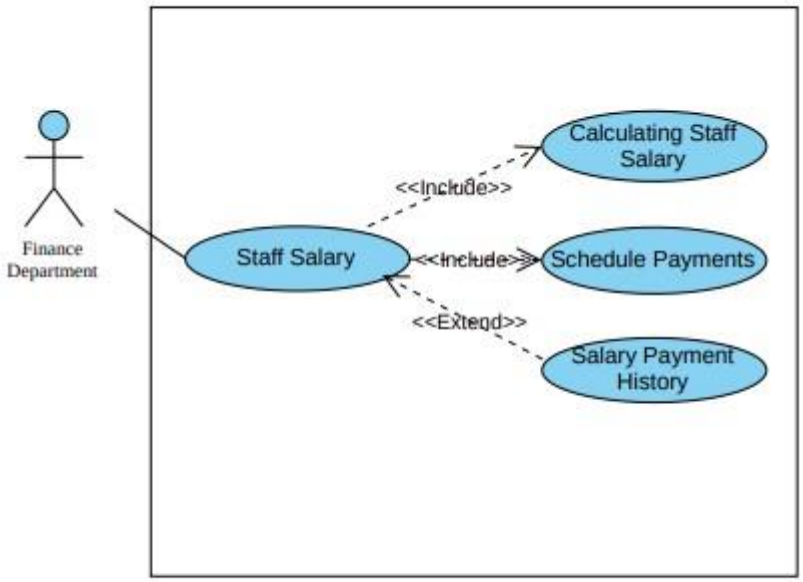
<1.3.2> Update Staff Roster:



<1.4.1> Generate Patient Expenses:

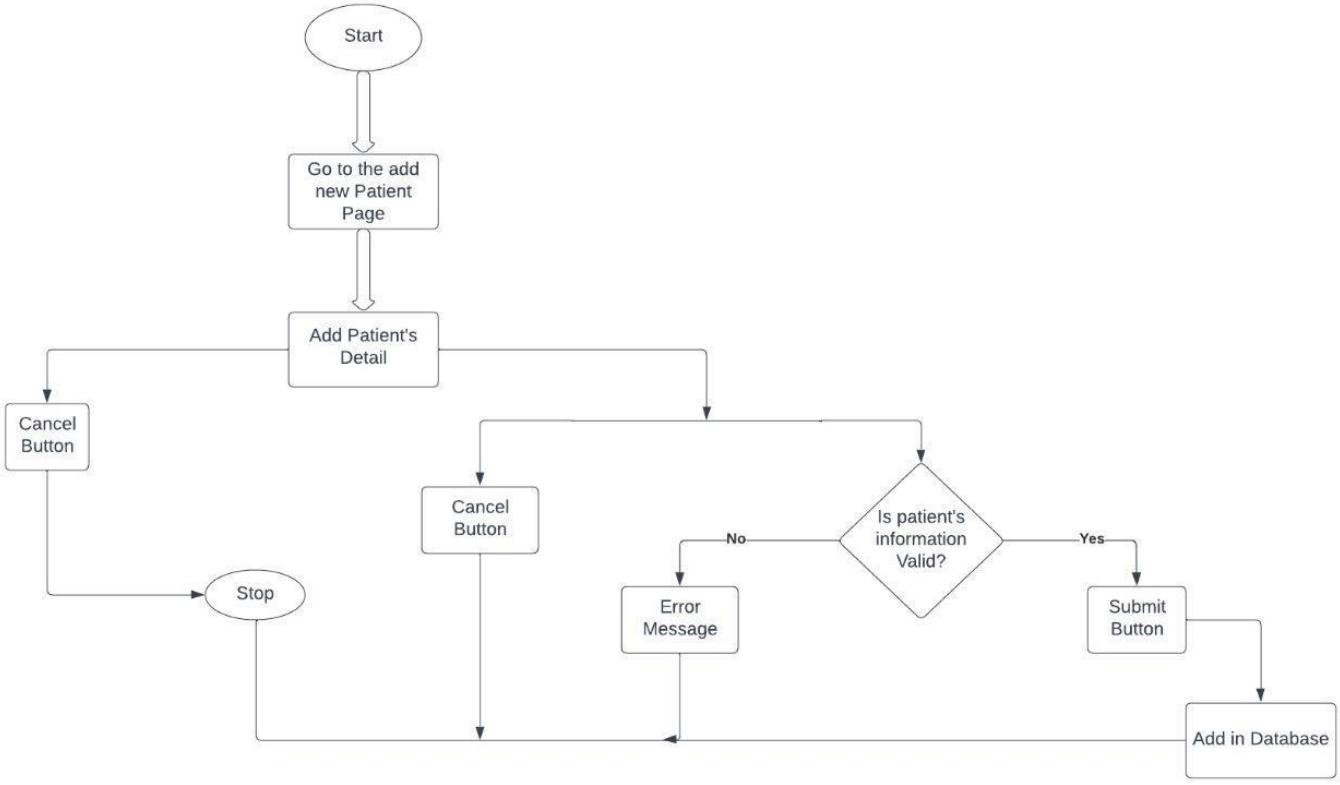


<1.4.3> Generate Staff Salary:

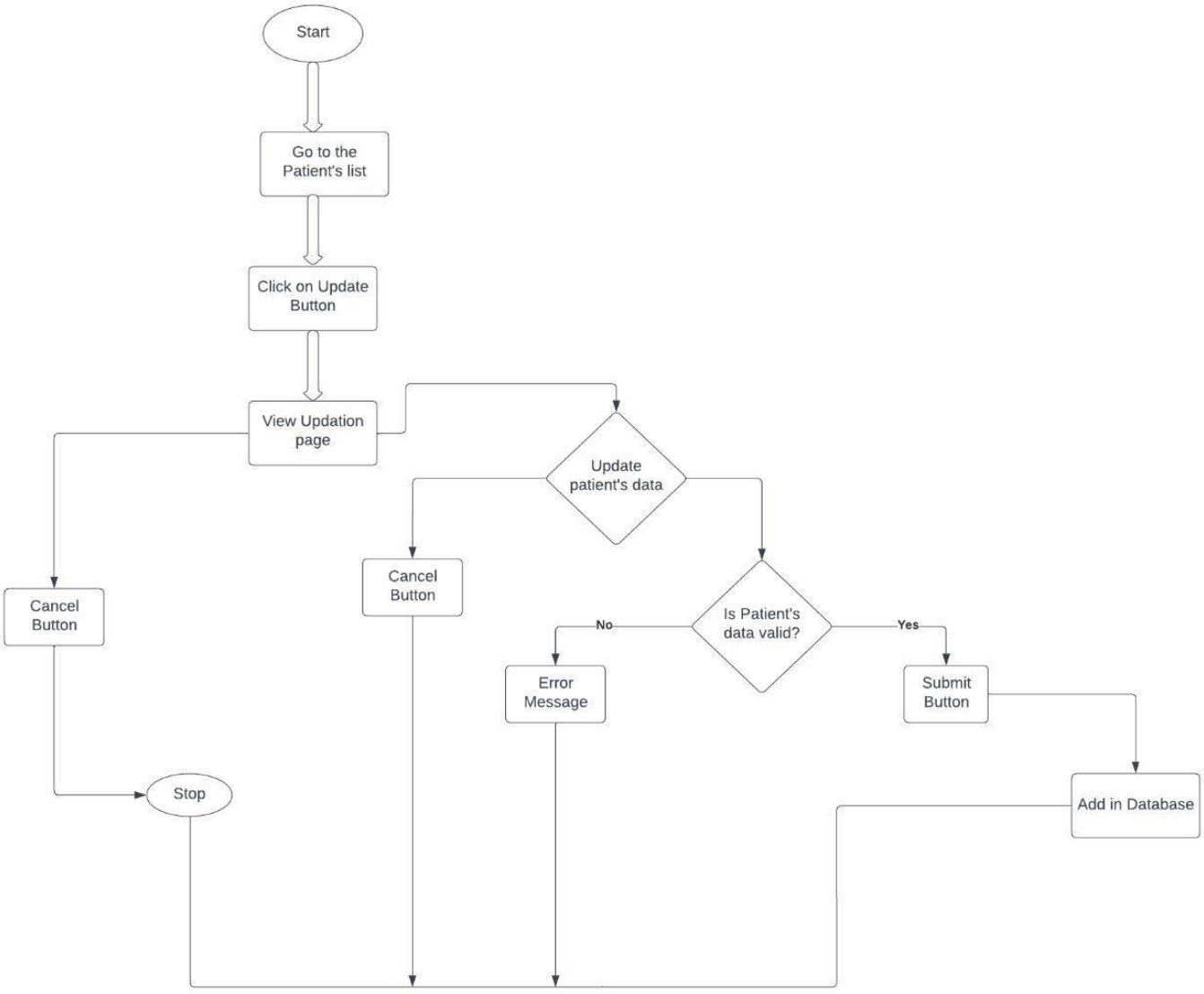


ACTIVITY DIAGRAM:

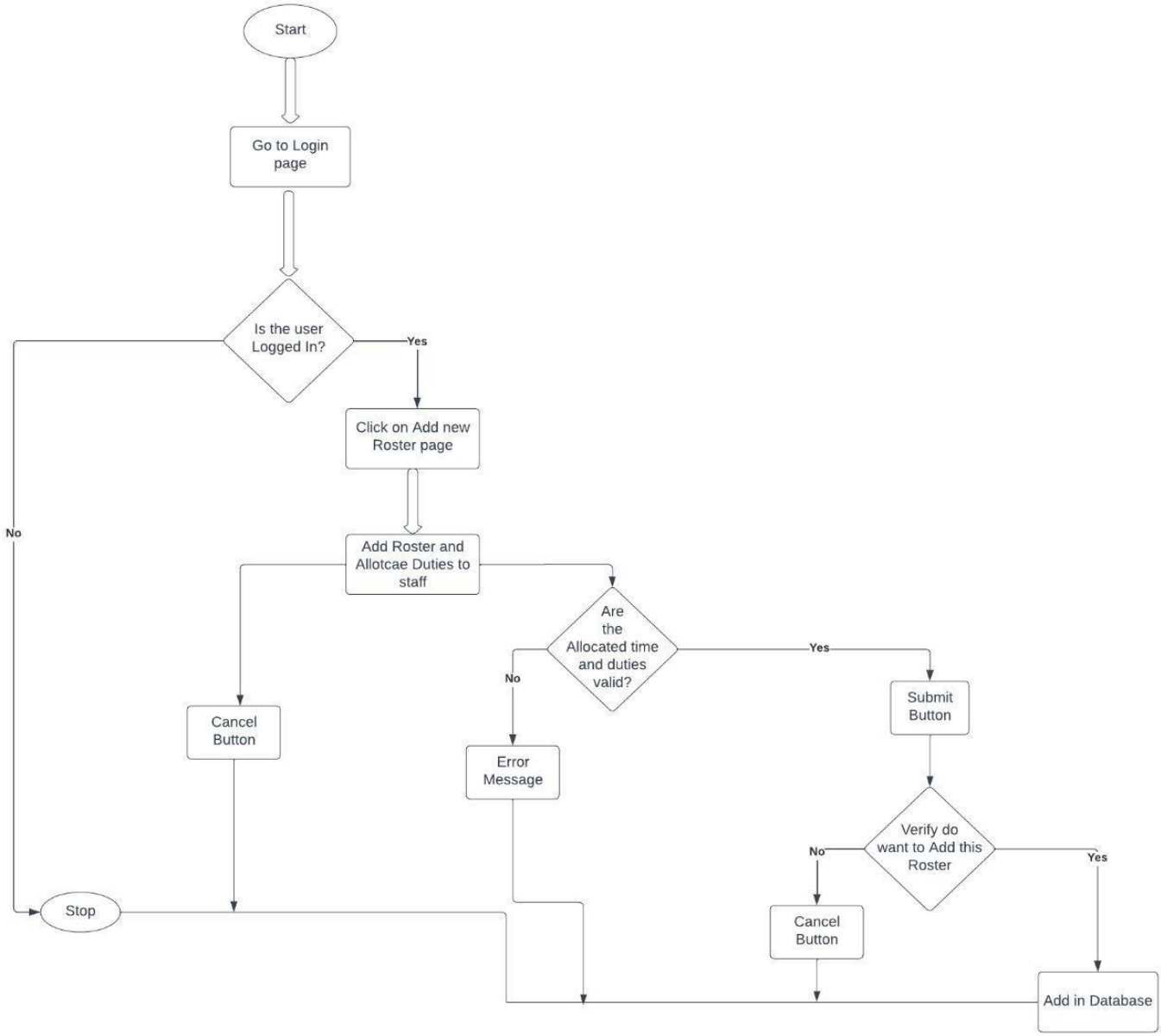
<1.1.1> Register a Patient:



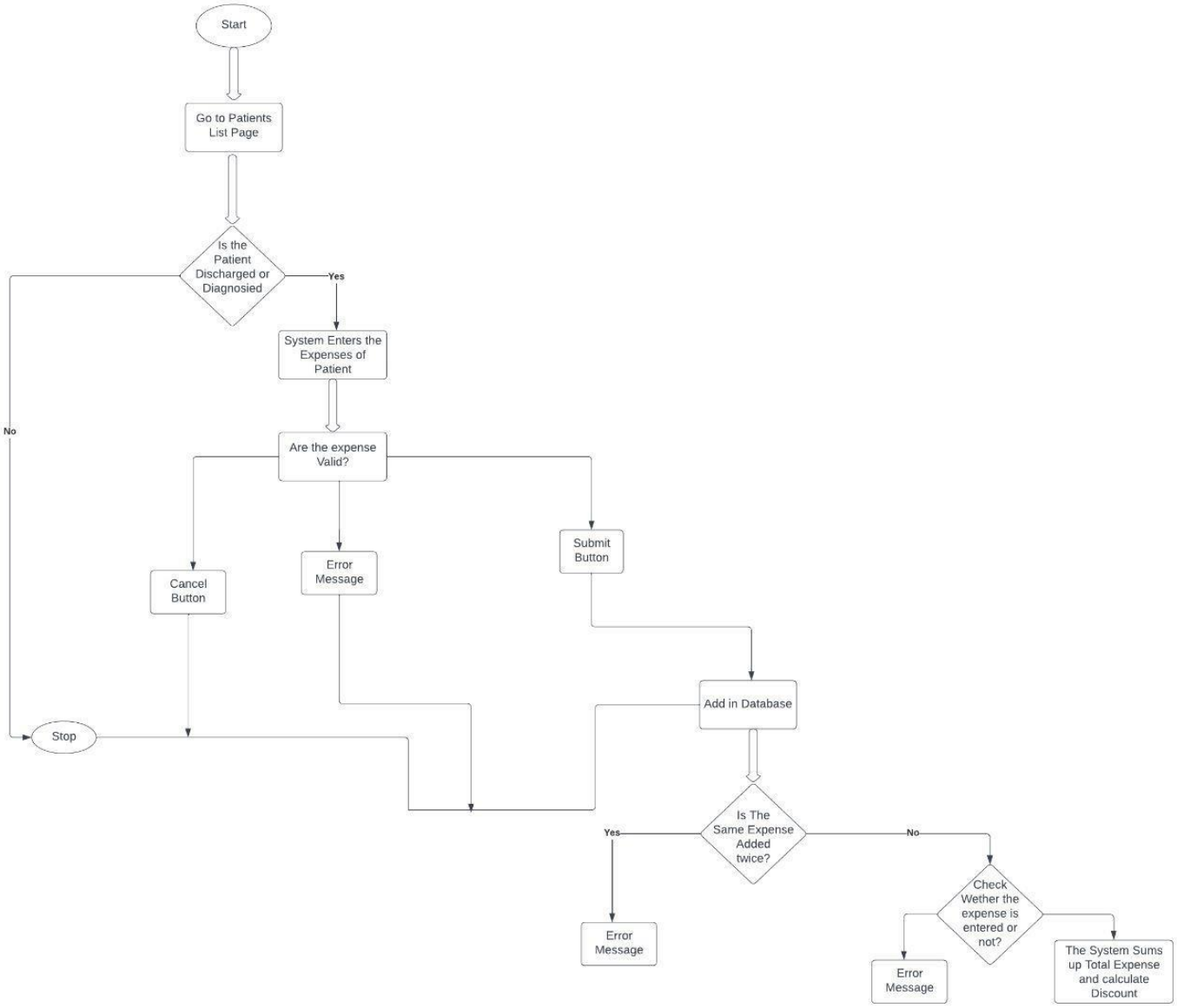
<1.2.2> Update Medical History:



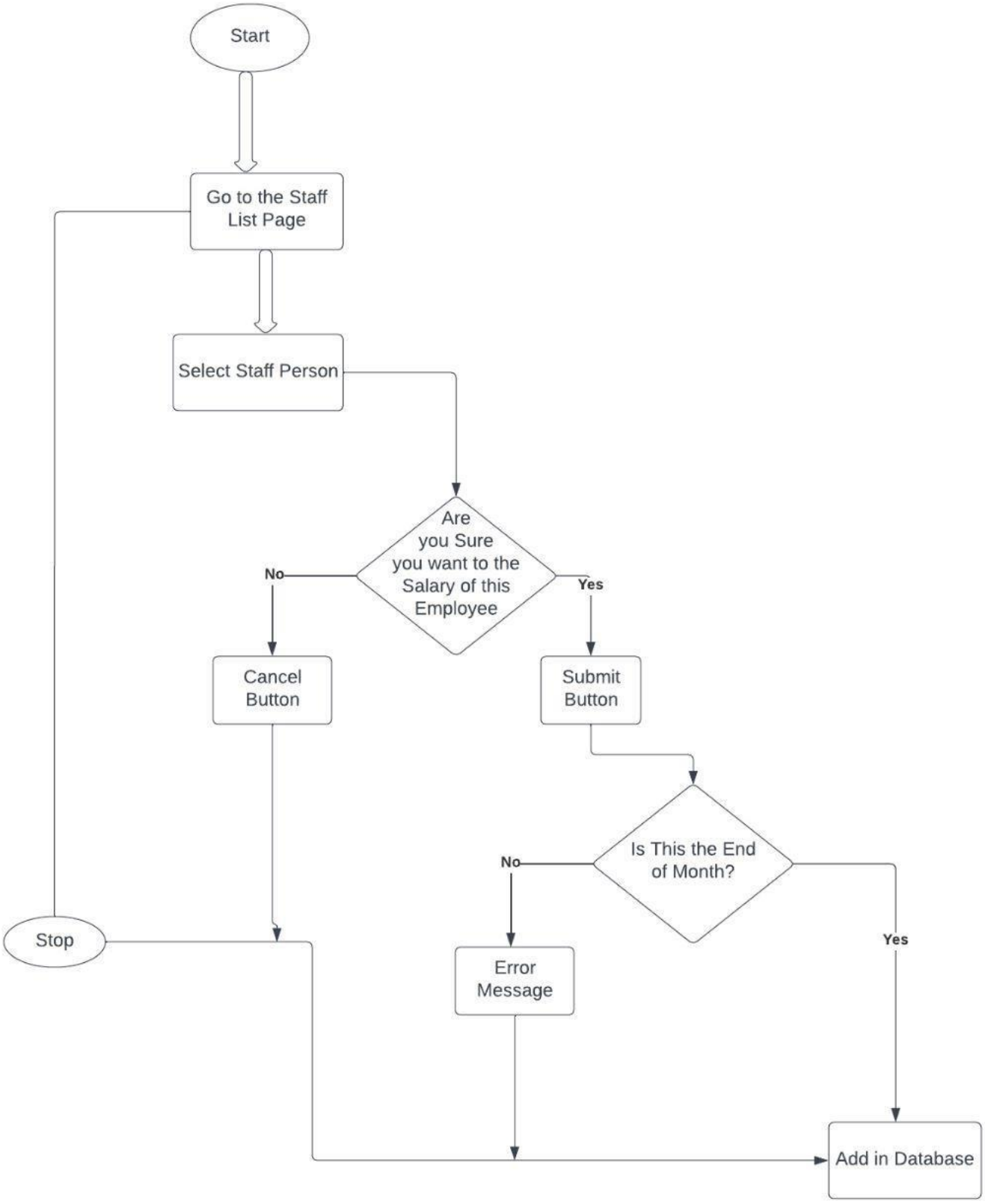
<1.3.2> Update Staff Roster:



<1.4.1> Generate Patient Expenses:

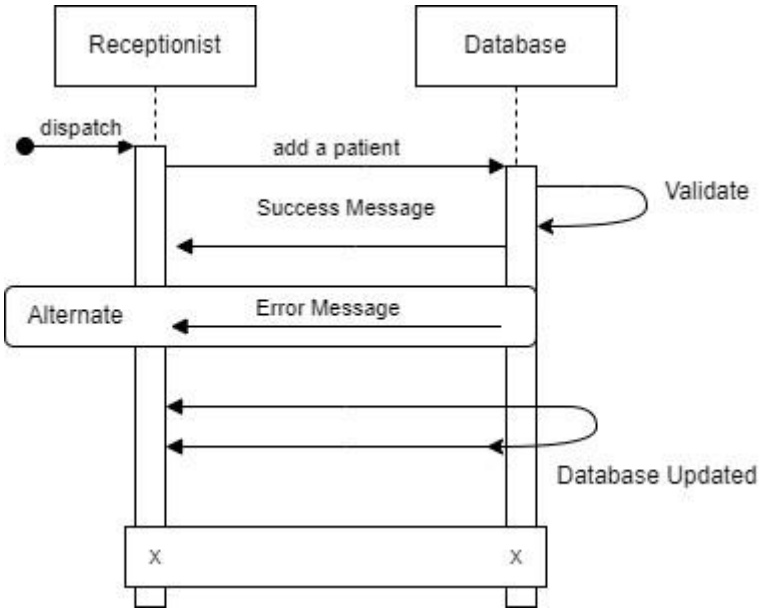


<1.4.3> Generate Staff Salary:

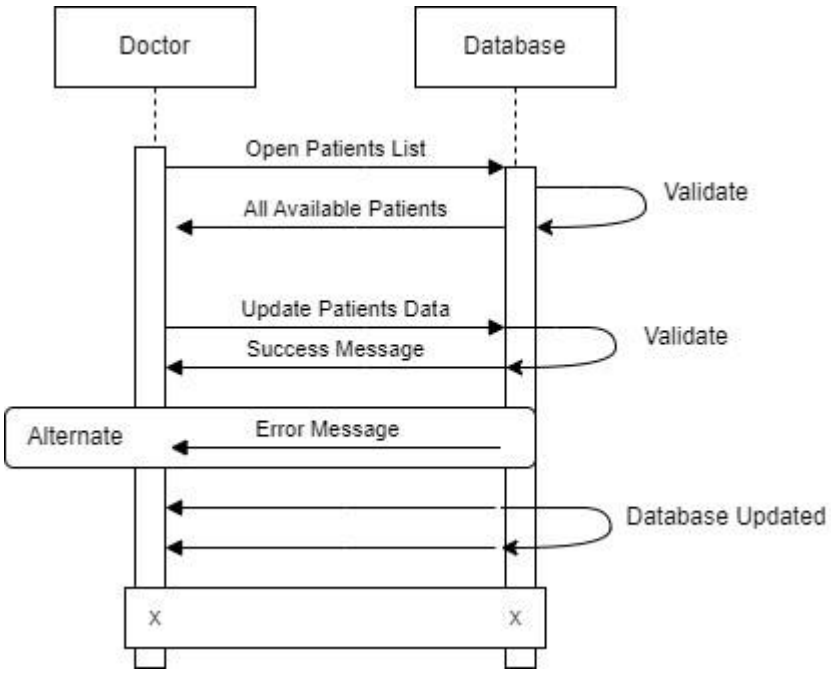


SEQUENCE DIAGRAM:

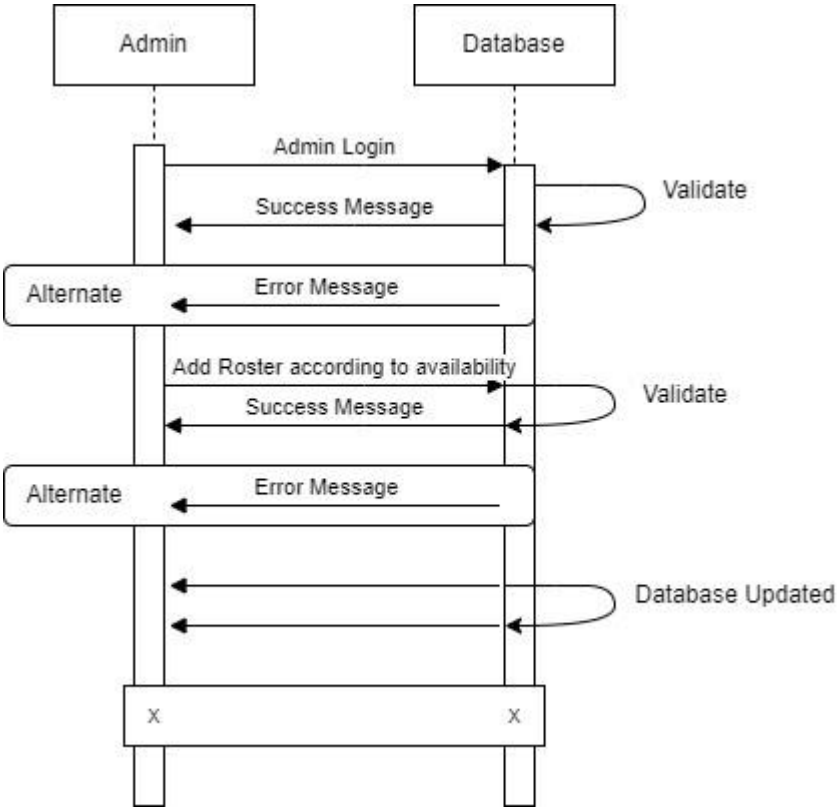
<1.1.1> Register a Patient:



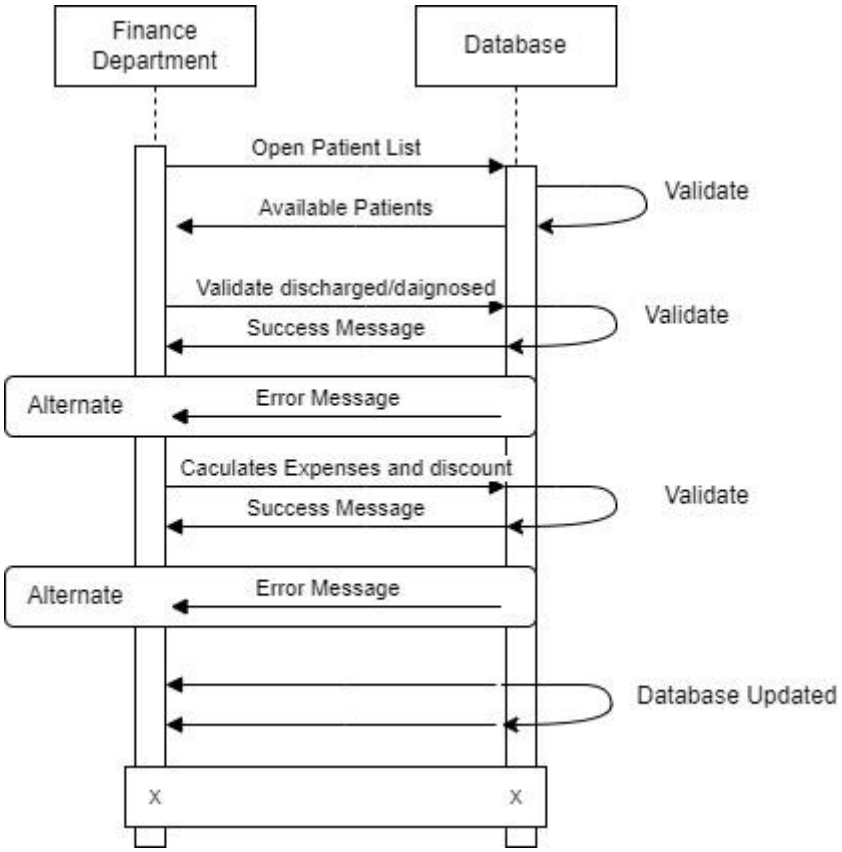
<1.2.2> Update Medical History:



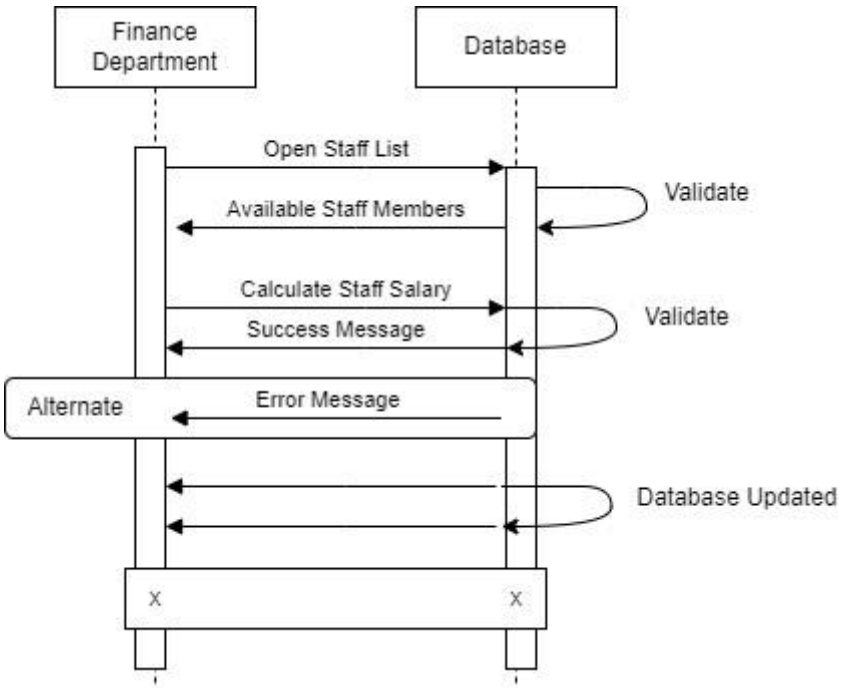
<1.3.2> Update Staff Roster:



<1.4.1> Generate Patient Expenses:



<1.4.3> Generate Staff Salary:



TEST CASES:

<UCS_1.1.1> Register a Patient:

Test Case ID	Test Case Name	Actors	Test Case Description	Pre-Conditions
UCS_1.1.1	Register a Patient	Receptionist, Patient	Let's the Receptionist create a record for every new patient so that all their data is stored.	The patient or their attendant must provide the patient's details to the receptionist.

Steps	Expected Result	Actual Result	Status
1: From the home page, press the "Add a Patient" button. **2: On the "Add a Patient" page, enter all the given patient details.** **3: The receptionist confirms the entry.**	The patient's record will be successfully created in the system.	The patient's record is successfully created in the system.	Pass
	Alternative Flow		
The Receptionist enters data of an already existing patient.	An alert will pop with the message: "Patient already registered".	An alert pop with the message: "Patient already registered".	Pass
	Alternative Flow		
The Receptionist denies the confirmation.	The patient's record will not be created in the system.	The patient's record is not created in the system.	Pass

Test Data:

Name	Age	Gender	Phone Number	Diagnosis	Confirm Registration	Result
"Hamza"	"20"	"Male"	"+923338485105"	"Spasm"	Yes	True
"Hamza"	"20"	"Male"	"+923338485105"	"Spasm"	Yes	False
"Rohail"	"22"	"Male"	"+9233100522212"	"Broken Leg"	No	False

<UCS_1.2.2> Update Medical History:

Test Case ID	Test Case Name	Actors	Test Case Description	Pre-Conditions
UCS_1.2.2	Update Medical History	Doctor, Patient	Allows the doctor to update patient's medical history so that if the same patient comes back again then this history would help in their diagnosis.	The registered patient is diagnosed by the doctor.

Steps	Expected Result	Actual Result	Status
1: From the home page, press the Update Patient Data button. 2: On the Update patient data page, enter all the updated patient details. 3: The doctor confirms the entry.	The patient's record will be successfully updated in the system.	The patient's record is successfully updated in the system.	Pass
	Alternative Flow		
The doctor enters incomplete data.	An alert will pop with the message: "Update Failed".	An alert pop with the message: "Update Failed".	Pass
	Alternative Flow		
The doctor denied confirmation.	The patient's record will not be updated in the system.	The patient's record is not updated in the system.	Pass

Test Data:

Name	Age	Gender	Phone Number	Diagnosis	Confirm Registration	Result
“Muneeb”	“20”	“Male”	“+923338485105”	“Malaria”	Yes	True
“Ajlal”	“25”	“Male”	“”	“Paralysis”	Yes	False
“Rohail”	“22”	“Male”	“+9233100522212”	“Broken tendon”	No	False

<UCS_1.3.2> Update Staff Roster:

Test Case ID	Test Case Name	Actors	Test Case Description	Pre-Conditions
UCS_1.3.2	Update Staff Roster	Admin, Staff	This will allow the admin to create a new roster for upcoming week and allot duties to the staff members decided by the admin.	User is logged on to the system as an admin.

Steps	Expected Result	Actual Result	Status
1: From the home page, press the Update Roster button. 2: On the Update Roster page, enter the new assigned duties in the roster. 3: The admin confirms the roster.	A new roster will be created for the following week.	A new roster is created for the following week.	Pass
	Alternative Flow		
The admin assigns the same duty to more than one staff member.	An alert will pop with the message: “Multiple assignations are not allowed”.	An alert pop with the message: “Multiple Assignations are not allowed”.	Pass

	Alternative Flow		
The admin denied confirmation.	The roster will not be updated in the system.	The roster is not updated in the system.	Pass

Test Data:

Duty	Assigned Staff Member	Slot	Date	Confirm Registration	Result
“Indoor”	“Shaqeela”	“Evening”	“20/02/2023”	Yes	True
“Indoor”	“Fahad”	“Evening”	“20/02/2023”	Yes	False
“Emergency”	“Sayyan”	“Morning”	“22/02/2023”	No	False

<UCS_1.4.1> Generate Patient Expenses:

Test Case ID	Test Case Name	Actors	Test Case Description	Pre-Conditions
UCS_1.4.1	Generate Patient Expenses	Finance Department	This use case allows the finance department to calculate all the expenses of a patient scaling from equipment used on them, service fee to medication charges.	1: Logged in as finance department member. 2: The registered Patient is discharged/diagnosed.

Steps	Expected Result	Actual Result	Status
1: From the home page, press the Generate Expense Button. **2: On the Generate Expense page, enter the Patients Name and add their expenses.** **3: The finance department confirms the expenses.**	An expense report will be generated by the system	A expense reports is generated by the system.	Pass
	Alternative Flow		
The finance department adds the same expense twice.	An alert will pop with the message: “Same Expense cannot be added twice”.	An alert pop with the message: “Same Expense cannot be added twice”.	Pass
	Alternative Flow		
The finance department denied confirmation.	An expense report will not be generated by the system	An expense report is not generated by the system	Pass

Test Data:

Patient ID	Patient Name	Expense Detail	Cost	Confirm Registration	Result
“2”	“Bajwa”	“Room Charges”	“20000”	Yes	True
“2”	“Bajwa”	“Room Charges”	“20000”	Yes	False
“31”	“Furqan”	“Medicines”	“10000”	No	False

<UCS_1.4.3> Generate Staff Salary:

Test Case ID	Test Case Name	Actors	Test Case Description	Pre-Conditions
UCS_1.4.3	Generate Staff Salary	Finance Department	This use case allows the finance department to calculate the staff salary according to the number of duties they performed.	1: Logged in as finance department member. 2: It is the end of the month.

Steps	Expected Result	Actual Result	Status
1: From the home page, press the Generate Salary Button. 2: On the Generate Salary page, select the Staff person's Name. 4: Click on generate salary. 3: The finance department confirms the salary generation.	A salary slip will be generated by the system.	A salary slip is generated by the system.	Pass
	Alternative Flow		
The finance department denied confirmation.	A salary slip will not be generated by the system	A salary slip is not generated by the system	Pass

Test Data:

Staff ID	Staff Name	Confirm Registration	Result
"5"	"Atif"	Yes	True
"31"	"Hashim"	No	False