SALALE UNIVERSITY

COLLEGE OF NATURAL SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

Final Year Project documentation

Project Title: WEB-BASED PHARMACY MANAGEMENT SYSTEM FOR SALALE UNIVERSITY COMPREHENSIVE SPECIALIZED HOSPITAL

PREPARED BY

No	Name	Id no	Email Adress	Phone NO
1	**Anduamlak Walelign**	Ru 1242/14	manandu537@gmail.com	+251 928381772
2	**Ashebir Tesfaye**	RT/10018/16	ashuutesfaye8@gmail.com	+251 975069417
3	**Ermiyas Tesfaye**	Ru 1274/14	*ermiyastesfaye527@gmail.com*	+251 902737714

Advisor: Mr. Nega (MSc.)

Fiche, Ethiopia

2/12/2025

Declaration

We hereby declare that the Final Year Project Documentation titled **"Pharmacy Management System for Salale University Comprehensive Specialized Hospital"** is the outcome of our collective effort and shared commitment. The concepts and insights presented in this document represent the original contributions of each team member. As co-authors, we take full responsibility for the authenticity of the content and its alignment with ethical standards. By signing below, we affirm this declaration.

Name of Student	**ID No**	**Signature**
Anduamlak Walelign	Ru 1242/14	_____
Ashebir Tesfaye	RT/10018/16	_____
Ermiyas Tesfaye	Ru 1274/14	_____

Approval

This is to certify that Anduamlak Walelign, Ashebir Tesfaye & Ermiyas Tesfaye Regular students of Salale University carried out this project documentation under the guidance of Mr. Nega Firdissa And now ready for submission to the department of computer science at Salale University in Fiche, Ethiopia.

Advisor Name	**Date**	**Signature**
Nega Firdissa	1/30/2025	Nega Firdissa

Acknowledgment

We express our sincere gratitude to the Almighty God, whose guidance, grace, and blessings have been our constant source of strength throughout this journey. His support has provided us with the resilience and inspiration to complete this project successfully. We also extend our heartfelt appreciation to our families, whose unwavering love, encouragement, and sacrifices have been the foundation of our achievements. Your belief in us has been a driving force in this endeavor, and we are deeply grateful for your support. To our friends and colleagues, thank you for your companionship, motivation, and encouragement. Your support has brought joy to this journey and made the challenges easier to face. This project, **"Pharmacy Management System for Salale University Comprehensive Specialized Hospital,"** is a reflection of the collective strength we have drawn from God, our families, and our friends. We are profoundly grateful for your contributions and presence throughout this process.

Acronyms

CRC	Class-Responsibility-Collaborator
CSS	Cascading Style Sheets
ERD	Entity-Relationship Diagram
GDPR	General Data Protection Regulation
HTML	Hyper Text Markup Language
JS	JavaScript
NFRs	Non-functional requirements
OOAD	Object-Oriented Analysis and Design
PMS	Pharmacy Management System
UML	Unified Modeling Language

Table of Contents

List of Figure

List of Tables

Abstract

The Web-Based Pharmacy Management System (PMS) for Salale University Comprehensive Specialized Hospital is designed to address the inefficiencies and challenges of the current manual pharmacy operations. The existing system relies on paper-based record-keeping, leading to time-consuming processes, human errors, and difficulties in tracking drug inventory, expiration dates, and generating accurate reports. This project aims to automate and streamline these operations by developing a comprehensive, user-friendly, and secure web-based system. The PMS will provide functionalities such as drug registration, inventory management, expiration date tracking, sales processing, and report generation. It will support multiple user roles, including Managers, Admins, Pharmacists, Cashiers, and Customers, each with specific access controls and responsibilities. The system will enhance operational efficiency, reduce manual errors, and improve data accuracy, ultimately leading to better service delivery and patient safety. The system is developed using Object-Oriented Analysis and Design (OOAD) methodology, ensuring modularity, scalability, and reusability. The architecture is divided into three layers: Presentation, Application, and Database, ensuring a seamless flow of information and secure data handling. The system will be implemented using web technologies such as HTML, CSS, JavaScript, PHP, and a relational database for data storage.

Key features of the PMS include real-time inventory tracking, automated alerts for expired drugs, role-based access control, and efficient report generation. The system is designed to be technically, operationally, economically, and legally feasible, with a focus on user experience, security, and scalability. By transitioning from a manual to a computerized system, the PMS will significantly improve the efficiency and accuracy of pharmacy operations at Salale University Comprehensive Specialized Hospital, benefiting both employees and patients.

CHAPTER ONE

1. Introduction

1.1 Background Information

A Pharmacy Management System (PMS) is a comprehensive software solution designed to manage various pharmacy operations efficiently, including inventory management, prescription handling, drug expiration tracking, and reporting. By automating these processes, a PMS reduces human error, enhances accuracy, and improves overall workflow.

At Salale University Comprehensive Specialized Hospital, the pharmacy faces several challenges due to its reliance on a manual system. These issues include difficulty in tracking the quantity of drugs, inefficient monitoring of expiration dates, slow and error-prone report generation, and challenges in retrieving accurate and up-to-date drug information. Manual systems often lead to data redundancy, inaccuracies, and increased workload for employees. Additionally, manual drug tracking increases the risk of dispensing expired or insufficient medications.

To address these challenges, this documentation suggests the implementation of a computerized Pharmacy Management System for Salale University Comprehensive Specialized Hospital. The system will automate inventory tracking, monitor expiration dates, generate reports efficiently, and provide quick access to comprehensive drug information. By digitizing pharmacy operations, Salale Hospital can enhance service quality, minimize errors, and ensure better drug management, ultimately benefiting both employees and patients.

1.2. Description of the Existing System

The existing system at Salale University Comprehensive Specialized Hospital Pharmacy is entirely manual, relying on traditional record-keeping methods for managing medications, patient prescriptions, inventory, and transactions. Pharmacists and staff record drug stocks, sales, and customer information in physical logbooks or on paper forms, a process that demands meticulous attention to detail and constant updates to keep track of inventory levels and medication status. When prescriptions are filled, pharmacy staff must manually check available stock, write down each transaction, and update the drug quantities accordingly. This method not

only consumes significant time but also risks human error due to its complexity and the high volume of daily transactions.

Limitations of the Existing System

- Manual recording of inventory and transactions is time-consuming, reducing pharmacy efficiency.
- The manual system increases the risk of human error, leading to stockouts or overstocking.
- Physical records are prone to damage and redundancy, making data management inefficient.
- Manually tracking medication expiration dates is difficult, risking patient safety.
- Retrieving specific records is slow, delaying service and impacting patient satisfaction.

1.3. Statement of the Problem

Managing the extensive pharmacy operations at Salale University Comprehensive Specialized Hospital with paper records is increasingly impractical and inefficient. Relying on manual processes makes it difficult to keep accurate records and ensure timely access to important information. This outdated system poses several challenges, including:

- Difficulty in navigating complex interfaces, leading to inefficiency and errors in pharmacy operations.
- Lack of an organized system for registering and storing items, leading to confusion and inefficiency.
- Difficulty of getting full information about drugs when needed immediately.
- Inability to easily identify and remove expired drugs, leading to the risk of dispensing outdated medications.
- Preparing reports manually is slow and prone to redundancy, resulting in delayed and inaccurate information.

1.4. Objective of the project

1.4.1 General Objective

The general objective of this project is to design and develop Web Based pharmacy Management System for Salale University Comprehensive Specialized Hospital.

1.4.2 Specific objective

To achieve the above mention general objective, the project can address the following specific objectives. The system provides the following.

- To create a user-friendly interface.
- To develop a computerized system for recording data.
- To enable deletion or removal of expired drugs.
- To generate reports quickly and efficiently.
- To track and identify drug expiration dates.
- To maintain a system for item registration and storage.

1.5. Scope of the Project

The scope of this project is developing Web based System, that are bounded to

- Provide all necessary information about the Drugs in Salale University Comprehensive Specialized Hospital Pharmacy like position(location), types and amounts.
- Provides for manager ability to manage all activities of employees in pharmacy.
- Provides necessary information related to drugs for the customers.
- Provides interface for Manager, Cashier, Pharmacist.
- Provides interface for Manage accounts.

1.6. Limitations of the project

Despite the advantages provided by the Pharmacy Management System (PMS) at Salale University Comprehensive Specialized Hospital, the system has some limitations.

- The system relies on a stable internet connection, which may cause accessibility issues in areas with poor network coverage.
- The system does not include an integrated customer payment transaction method, requiring manual payment handling.

- Hardware failures such as server crashes or database corruption may disrupt system functionality.
- Employees may require training to adapt to the system, which could delay full adoption.
- The system may have limitations in integrating with other hospital management software.
- Regular maintenance and updates are necessary to ensure security and efficiency.

1.7. Significance of the project

The objective of the new system developed is to handle such Pharmacy Stock information Management in a better efficient and effective way. It also surely gives a lot of benefits for employees in Pharmacy, users and Patient are will have a web-based access to all possible records or information on the drugs available in pharmacy.

In general, the system provides the following significance: -

- Real-time alerts prevent the dispensing of expired or out-of-stock medications, ensuring patient safety.
- Automated tracking of drug stock levels, expiration dates, and pricing minimizes human errors.
- It provides efficient, flexible and reliable items' storing, locating and distributing.
- The system generates detailed reports on sales, stock movement, and drug usage trends for better decision-making.
- The system automates pharmacy operations, reducing processing time for inventory management, sales, and reporting.
- Enhance best controlling method for the drug.

1.8. Methodology

1.8.1 Data Gathering Methods

- **Primary Sources**: Gather data through observations, interviews.

  Observation: Data was gathered by directly observing daily pharmacy operations, including how pharmacists manage drug inventory, dispense medications, handle sales transactions, and track expired drugs. This method provided insights into workflow efficiency, common challenges, and areas for improvement.

Interview: Structured interviews were conducted with pharmacy employees, including pharmacists, store coordinators, cashiers, and managers. The questions focused on their roles, challenges faced in manual processes, system requirements, and expectations for a digital solution. These interviews helped in understanding user needs and identifying key features for the Pharmacy Management System.

- **Secondary Sources**: document analysis. To get historical information of the organization activities and to know the Organization rules and regulations the team members are analyzed documents which are relevant to the new System.

1.8.2 Methodology Development

For the design and analysis phase of the **Pharmacy Management System (PMS) project**, the **Object-Oriented Analysis and Design (OOAD)** methodology is highly compatible. This methodology models the system using objects that represent core real-world entities like drugs, Patient, pharmacists, and managers. It simplifies the representation of relationships and interactions between these components, ensuring clarity and cohesion in the system's structure. OOAD also enhances the project by encapsulating data and behavior, promoting modularity, scalability, and reusability—key attributes for managing pharmacy operations efficiently. Furthermore, the iterative nature of OOAD supports continuous refinement and adaptation, making it ideal for building a robust and user-friendly pharmacy management system [1].

1.8.3 Developmental Approach

For developing our system, we used prototype development modeling, because of the following reason.

- Users are actively involved in the development
- Since in this methodology a working model of the system is provided, the users get a better understanding of the system being developed.
- Errors can be detected much earlier.
- Quicker user feedback is available leading to better solutions.
- Missing functionality can be identified easily
- It is a quickly developed.
- Easily modifiable.
- Can easily Extendable.

- For small or medium-size interactive systems.
- For parts of large systems (e.g. the user interfaces).
- For short-lifetime systems [2].

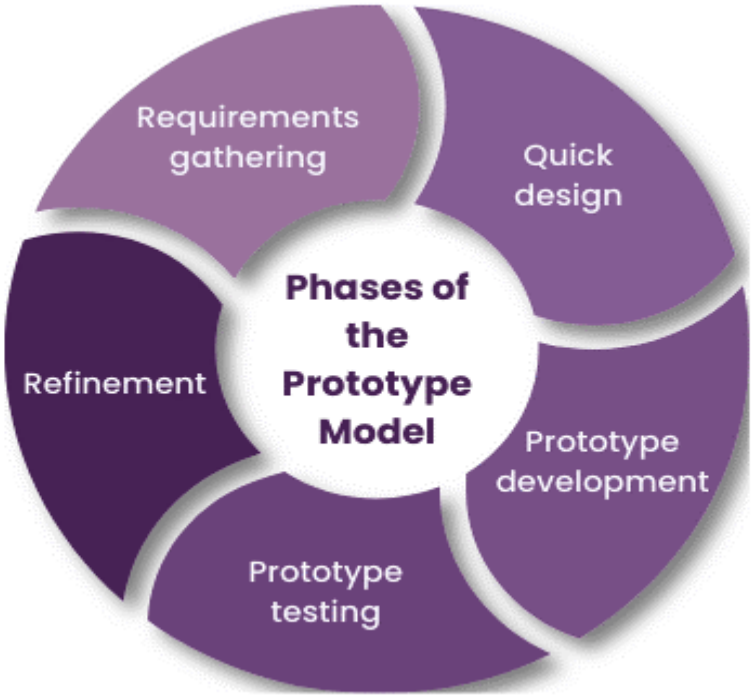


Figure 1: Prototype Development Modeling

1.8.4 Design Architecture

Our project will utilize a one-tier architecture during the development and testing phase, and will transition to a three-tier architecture upon completion. The one-tier architecture is chosen for its simplicity and efficiency in supporting the project's initial requirements. Once the project is finished, we will implement a three-tier architecture.

1.8.5 System Implementation Method

1. **Client Side**: The user interface, built with
   - **Hypertext Markup Language (HTML):** -we used it in order to make our project user interactive and to construct forms for our project [3].

- **Cascading Style Sheet (CSS): -**it is a small part of code that is saved externally out of a certain PHP code and internally PHP code which help us to made our system user interactive [3].
- **Java script: -** it is a code inside PHP tag used for validating, redirecting and instructing forms that we used in our project [3].
- **React.js:** ensures a dynamic and responsive frontend, improving the user experience for all roles (Manager, Pharmacist, Cashier, etc.) [4].
- **Bootstrap:** simplifies the design process and ensures the system is mobile-friendly [4].
- **jQuery:** To simplify HTML DOM tree traversal and manipulation.

2. **Server Side**: The backend is developed in
   - **Xampp server:** we used it in our project to create database and tables, to store records inside the database provided, to retrieve necessary records from database.
   - **PHP:** we used it at server side cross-platform technology and it came to mean with "Hypertext Preprocessor". It is widely general-purpose scripting language that is especially suited for dynamic web development and it can also have embedded into HTML (hypertext markup language) [2].

1.9. Developmental Tools

The development tools for the web-based pharmacy management system project encompass a range of software and hardware:

Table 1 : Hardware Tools

Tools	Quantity	Description
Desktop Computer	1	➢ For development and testing of the application.
Laptop	1	➢ To facilitate project development and documentation.
External Hard Disk	1TB	➢ For storing software, backups, and project files.
Testing Devices	Multiple	➢ To ensure functionality on different browsers and platforms [4].
Printer	1	➢ To print documentation

Table 2: Software Tools and Their Purposes

Tool	Purpose
Visual Studio Code	For writing and editing the application's source code.
Microsoft Edge / Brave/ Chrome	To browse the internet and access resources.
Microsoft Office 2019	To prepare documentation and reports.
XAMPP	To run PHP files locally and manage databases.
E-draw Max	To develop diagrams.
Windows 10/Windows 11	To run other software [7].

1.10. Feasibility Study

1.10.1 Technical Feasibility

The project demonstrates strong technical feasibility with a well-supported technology stack, including PHP, HTML, CSS, and JavaScript. This allows for seamless integration of front-end and back-end components, ensuring that the system can be easily developed and scaled to accommodate increasing user demands.

1.10.2 Operational Feasibility

The new system will increase and improve the activity of customers by replying the reliable data, response time, and increase efficiency of work in the pharmacy. The project is welcomed with great pleasure by the organization. The employees are more cooperated to give needed information. This indicates that the project is operationally feasible.

1.10.3 Economic Feasibility

The development cost is less when it is compared to the benefit that the project will bring to the organization. Hence the project is economically feasible.

This means the concrete benefit that can be expressed in terms of birr. So, the system proposed to develop will decrease a lot of birrs that was expensive to buy the hard copy document material such as paper, pencil, rubber, and so on. But after the system developed the data can be managed

with in computer. Give more readable, reliable, easily manageable, and database which contains all employees' track.

1.10.4 Legal Feasibility

Legal feasibility is a critical consideration, as the project will adhere to copyright laws and data protection regulations, such as GDPR. By implementing proper measures for data security and ethical handling of user information, the project can operate within legal boundaries.

1.10.5 Political Feasibility

The political feasibility of implementing a pharmacy management system at Salale Hospital is favorable, as it aligns with government priorities for advancing digital healthcare infrastructure, enhancing patient safety, and improving efficiency within public health institutions. By automating pharmacy operations, the system supports transparency, accurate data management, and better resource allocation, which are often encouraged by health policies. With clear communication about compliance with health regulations and privacy standards, the project can gain essential support from both government and hospital leadership, ensuring it aligns with policy goals while delivering improved patient care and operational effectiveness [8].

1.10.6. Time feasibility

The time feasibility for the Pharmacy Management System project at Salale University Comprehensive Specialized Hospital is realistic and achievable within the proposed 8-month timeline, from October11 to June 2025. Each phase has been carefully planned with specific milestones to ensure steady progress. The time allocated for requirements gathering, system design, development, testing, and training is balanced, allowing for sufficient time to address potential issues. Given the project scope and resource allocation, the timeline provides flexibility to accommodate minor adjustments without significantly affecting the overall delivery. However, the project's success hinges on timely feedback from stakeholders, the availability of key resources, and effective risk management to prevent delays.

1.10. Schedule of activities and Budget breakdown

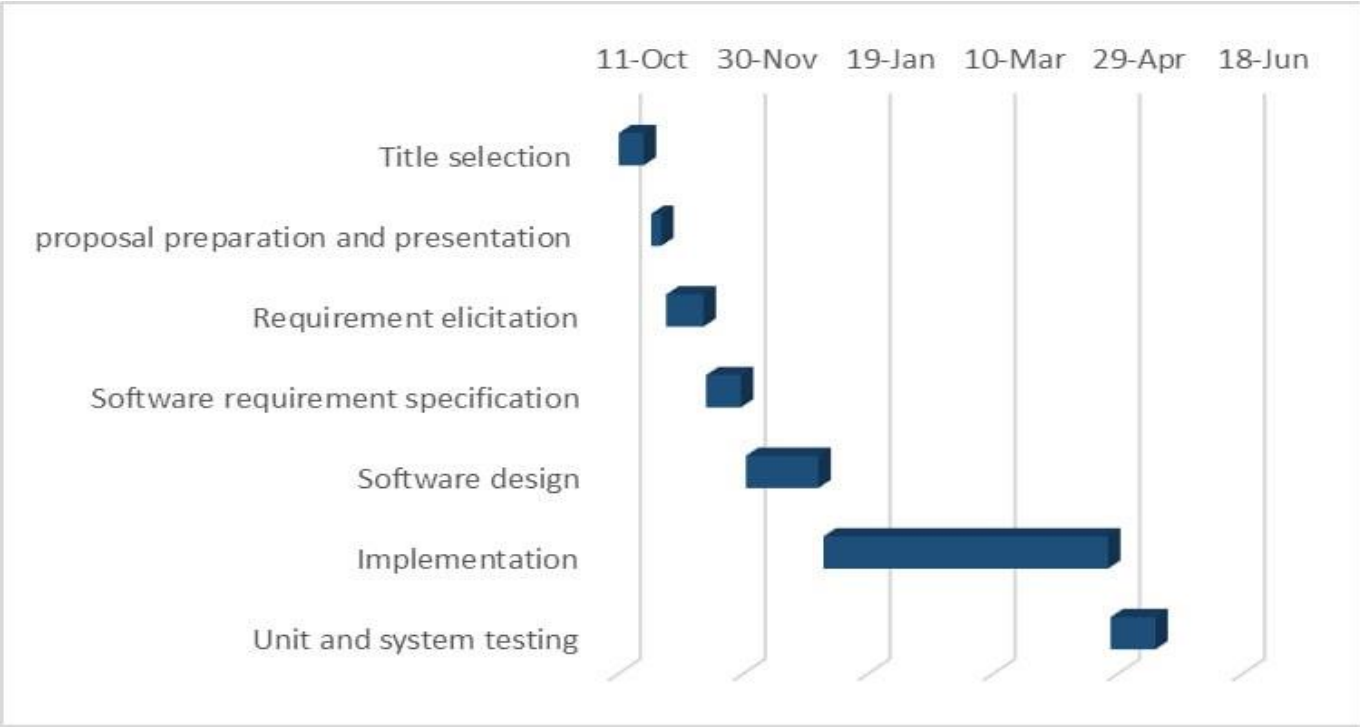


Figure 2 : Schedule of Activities chart

1.11. Team composition

Table 3 Team Composition

No	Name	Id no	Email Adress	Responsibility
1	**Anduamlak Walelign**	Ru 1242/14	manandu537@gmail.com	Documentation analyzer and Data Collector
2	**Ashebir Tesfaye**	RT/10018/16	ashuutesfaye8@gmail.com	Diagram Designer and Data Collector
3	**Ermiyas Tesfaye**	Ru 1274/14	*ermiyastesfaye527@gmail.com*	Documentation analyzer and Software Designer

1.12. Organization of the project

Chapter one covers a range of foundational topics, including the introduction, background, problem statement, overall and specific objectives, data collection methods, development methodologies, approaches, and tools utilized to bring the project to fruition. Additionally, it addresses the feasibility study, project scope, and its significance. Chapter two delves into requirement analysis, detailing the existing system, proposed system, functional and non-functional requirements, along with diagrams such as use case, sequence, state chart, activity, and class diagrams. It also includes user interface prototyping to illustrate the system's design. Chapter three focuses on the system design, outlining its purpose and objectives, the proposed software architecture, which incorporates subsystem decomposition, component and deployment diagrams, database design through E-R diagrams, and mechanisms for access control and security.

Chapter Two

2. Requirement Specification and Object-Oriented Analysis

2.1. Existing system

2.1.1. Existing system description

The current pharmacy system at Salale University Comprehensive Specialized Hospital is entirely manual. This means tasks such as checking expiration dates and verifying drug availability require physically inspecting each item within the pharmacy. Such a system is time-consuming and inefficient, leading to a significant loss of organizational time and resources.

When new drugs arrive, the manager instructs the store coordinator to store them in the drug store. The store coordinator then registers all drugs, including their type, quantity, and price. Once registered, the drugs are arranged on shelves according to their type. When a customer or user requests a drug, the store coordinator searches for the available drugs and retrieves them based on the pharmacist's instructions. The manager oversees all activities carried out by the employees.

The manual system also limits accessibility to records, as they are often tied to physical locations, making it challenging to retrieve or update information promptly, especially during emergencies. Report generation for inventory, sales, or expired drugs is labor-intensive and prone to inaccuracies. Furthermore, physical records are susceptible to loss, damage, and unauthorized access, compromising data security. These challenges highlight the urgent need for a computerized pharmacy management system to improve accuracy, efficiency, and overall service delivery [9].

2.1.2. Business Rule

The current system operates with specific guidelines for how patient are handled. These are as follows:

BR1: Pharmacists are required to treat patient courteously and respond effectively to their requests.

BR2: The cashier should receive the price of medicine honestly from patient and he/she should generate report for manager.

BR3: Manager should control the entire activity in the stock and should receive clear and appropriate report from the workers of the pharmacy.

BR4: Sold drug should order in their identifiable type to facilitate searching requested drug.

BR5: The manager must manage all information impartially and without any bias.

BR6: Forms should be designed to include accurate stock information.

BR7: Pharmacist doesn't sell the expired drug.

our project organization also has their own business rules to govern the activities in the pharmacy.

- **Give respecting for the customer.**

The existing system has its own mechanism in which its customers are treated.

These include:

- ✓ The pharmacist must treat customers in good manner and should address customer's request.
- ✓ The cashier should receive the price of medicine honestly from customers and he/she should generate report for manager.

- **Managing the drug properly.**
  - ✓ Manager should control the entire activity in the stock and should receive clear and appropriate report from the workers of the pharmacy.
  - ✓ Sold drug should order in their identifiable type to facilitate searching requested drug.
  - ✓ Manager should control the overall information from any biases properly.
  - ✓ Forms should contain stock information appropriately.
  - ✓ Pharmacist doesn't sell the expired drug.
- **Pay money for the employee**
  - ✓ The manager or the owner of the pharmacy must pay the cost for the worker once per month.

2.2 New System

The new Pharmacy Management System introduces several advanced features that address the challenges of manual and spreadsheet-based management systems. One of its key functionalities is the ability to automatically alert users about the expiration dates of drugs, ensuring timely action to remove or restock them and safeguarding patient safety. Additionally, the system incorporates a shelf location feature, enabling pharmacists to quickly locate drugs in their designated storage areas, which streamlines inventory handling and retrieval processes.

Another significant aspect is the automation of drug pricing, accompanied by built-in validation to ensure accuracy and consistency in pricing calculations. The system also provides regular updates on the stock levels of all drugs, keeping users informed and facilitating proactive inventory management. Its reporting capabilities allow users to access critical information, such as stock shortages or excesses, sales trends, and expiration statuses, in a clear and organized manner.

What sets this system apart from traditional manual processes is its automation, accuracy, and user-friendly interface. By eliminating the possibility of human errors, providing real-time notifications, and offering secure and centralized data management, the system significantly improves efficiency. Its ability to automate repetitive tasks, enhance reporting accuracy, and integrate advanced tracking features makes it a more reliable and efficient alternative to existing methods, transforming pharmacy operations.

2.2.1. Actor Description

Actor Descriptions for the Pharmacy Management System

- **Admin**

  The **Admin** is responsible for managing user accounts and ensuring system security. They create, activate, deactivate, and delete accounts, reset passwords, and oversee system access to maintain proper functionality. The admin does not directly handle pharmacy operations but ensures authorized users can perform their tasks.

- **Manager**

  The **Manager** oversees pharmacy operations and monitors system activities. Their responsibilities include managing employee records, viewing sales and inventory reports,

and ensuring smooth workflow within the pharmacy. The Manager has access to generate reports, track stock levels, and review notifications regarding low stock or expired drugs.

- **Pharmacist**

  The **Pharmacist** plays a key role in dispensing medications, managing patient interactions, and maintaining drug inventory. They check drug availability, register new patients, track prescription details, process sales, and monitor expiration dates to ensure patient safety. Pharmacists also generate reports related to drug usage and stock levels.

- **Cashier**

  The **Cashier** is responsible for handling financial transactions related to drug sales. They process payments, generate receipts for patients, and maintain records of all sales transactions. The Cashier also provides financial reports to the Manager for auditing and reconciliation purposes.

- **Store Coordinator**

  The **Store Coordinator** is in charge of inventory management, ensuring that drug stock levels are maintained and updated. They register new drug arrivals, update pricing, check for expired drugs, and prepare inventory reports. They work closely with the Pharmacist to ensure efficient stock distribution and timely restocking.

- **Patient**

  The **Patient** is the end-user who receives medications from the pharmacy. They can check drug availability, read drug information, and receive prescriptions from the Pharmacist. Patients interact with the system mainly through pharmacists and cashiers to complete their transactions.

2.2.2. Functional Requirements

Functional requirements define the core functionalities that the Pharmacy Management System must provide to meet the operational needs of Salale University Comprehensive Specialized Hospital. These requirements ensure that the system supports critical activities such as drug inventory management, prescription handling, employee information management, and report generation. By addressing these functionalities, the system aims to streamline workflows, reduce errors, and enhance overall efficiency in the pharmacy's operations [9].

- **Manager**

  - ✓ **Login/logout**: The Manager should be able to log in and log out of the system to access various functionalities related to managing pharmacy operations.
  - ✓ **Drug Registration Approval**: The Manager has the ability to review and approve newly registered drugs submitted by pharmacists, ensuring they meet necessary criteria.
  - ✓ **Employee Registration:** The system must allow a manager to register new employees into the system with the necessary personal and professional details.
  - ✓ **View Employee:** The system must allow the manager to view details of registered employees, either through a search or by listing all employees.
  - ✓ **View Reports**: Generate, customize, and export stock, sales, and expiration reports.

- **Admin**

  - ✓ **Activate/Deactivate Account:** The Admin is responsible for maintaining user access control. By activating or deactivating accounts, the admin ensures that only authorized users can access the system.
  - ✓ **Change Password:** To maintain system security and assist users facing login issues, the admin can reset or update passwords.
  - ✓ **Create Accounts**: Create secure user accounts for pharmacists, store coordinators, and cashiers.
  - ✓ **View Comments:** The admin has access to all user comments within the system. This capability allows for effective feedback collection.
  - ✓ **View Accounts**: A comprehensive overview of all user accounts is provided to the Admin. This includes details such as account statuses, roles, and associated data.

- **Pharmacist**

- ✓ **Login/logout**: The pharmacist should be able to log in and log out of the system for accessing their tasks, such as registering drugs and managing prescriptions.
- ✓ **Register Patient:** The system should allow the pharmacist to register new Patient, including capturing their personal details such as name, age, contact information, and any relevant medical history.
- ✓ **View Patient:** The system should enable the pharmacist to view existing Patient profiles and retrieve their details when needed for service or consultation.
- ✓ **Check Drug:** The pharmacist should be able to check the availability, stock levels, and details of drugs in the system, including expiration dates and batch information.
- ✓ **Sell Drug:** The system should facilitate the sale of drugs, recording the transaction details and updating the stock inventory automatically.
- ✓ **Search Patient:** The pharmacist should have the ability to search for Patient quickly by using identifiers such as name, ID, or phone number.

❖ **Store Coordinator**

- ✓ **Login/logout**: The store coordinator should be able to log in and log out to manage inventory tasks and track incoming and outgoing drugs.
- ✓ **Register Drug**: The system should allow the store coordinator to register new drugs into the inventory, including details like drug name, code, quantity, manufacturing date, expiry date, and supplier information.
- ✓ **Check Expired Drug**: The store coordinator should be able to identify and flag drugs nearing or past their expiration dates for removal or further action.
- ✓ **Update Price**: The system should enable the store coordinator to update the pricing information of drugs in the inventory based on adjustments or policy changes.
- ✓ **View Drug**: The store coordinator should be able to view detailed information about all drugs in the inventory, including stock levels, categories, and batch details.

❖ **Cashier**

- ✓ **Login/logout**: The cashier should have the ability to log in and log out of their system account to handle transactions.
- ✓ **View Sales:** The cashier should be able to access and review a list of all completed sales transactions, including details like date, time, items sold, and amounts received.
- ✓ **Give Receipt:** The system should enable the cashier to generate and print an itemized receipt for patient after each transaction, including details such as drug names, quantities, prices, and total cost.

❖ **Patient**

- ✓ **View Drug**: The system should allow Patient to browse and view available drugs, including their names, categories, and stock status.
- ✓ **Read Drug Information**: Patient should be able to access detailed information about each drug, such as usage instructions, dosage, payment, side effects, and expiry dates.

2.2.3. Non-functional Requirements and Constraints

Non-functional requirements (NFRs) refer to the criteria that define the operation and performance of a system, rather than its specific behaviors or functionalities. Unlike functional requirements, which describe what the system should do, non-functional requirements specify how the system should perform under various conditions. These requirements address aspects such as system performance, security, reliability, usability, and scalability, ensuring the system meets the broader needs of the users and stakeholders [6].

Performance Requirements

The system is designed to perform faster and more efficiently compared to the previous manual version. By transitioning to an automated solution, it ensures enhanced speed in operations, such as data retrieval and processing. The system's architecture incorporates robust security measures that react appropriately based on user actions; it remains secure if proper procedures are followed and responds with necessary safeguards

Security Issues

The system is protected from malicious users attempting to access the database, as most of the data is stored on the server. Authentication is carried out through password protection for

database manipulation. This means that the system will require a username and password before granting access to the database, preventing unauthorized modifications.

- The system follows role-based security, meaning that the system administrator sets the access level and privileges for each user.

Maintainability

Over time, changes may be required to add new functionalities based on user needs, when the system administrator identifies the need for modifications, or due to changes in the organization's workflow. Basic modifications, such as user interface adjustments, can be handled by a group of developers, but significant updates should be performed by the original system developers to ensure proper implementation and consistency.

Quality Issues

- **Usability:** The user interface should be simple to use and easy to learn.
- **Availability:** The system should be available at all times, provided there is power. It will be operational 24/7.
- **Reliability:** The application is designed to ensure consistent performance and dependability. By automating all processes and utilizing machine-driven operations, the system minimizes the risk of errors and maintains reliable functionality across various tasks.

Error Handling

The system handles errors by providing error messages and warnings to users. Error handling is applied in several areas. The system should verify a user's credentials during login. Users can access different parts of the system based on their assigned roles. The system should also manage errors caused by invalid user inputs, displaying an error message and additional information on how to correct the issue.

Compatibility

- **Browser Compatibility:** The system must ensure compatibility with major web browsers, including the latest versions of Chrome, Firefox, Safari, and Edge. This compatibility guarantees a consistent and reliable user experience across diverse browser

platforms, enabling users to access the system seamlessly, regardless of their preferred browser.

- **Device Compatibility:** The system should function efficiently across a range of devices, including desktops, tablets, and smartphones. Ensuring device compatibility is crucial for providing users with a smooth and uninterrupted experience, regardless of the type of device they use to interact with the pharmacy management system.

2.2.4. Use case diagram

A use case is a detailed description of how a user interacts with a system to achieve a specific goal. It outlines the steps involved, including the actions taken by the user and the system's responses. Use cases help to clarify functional requirements by providing real-world scenarios, ensuring that the system meets the needs of its users [1].

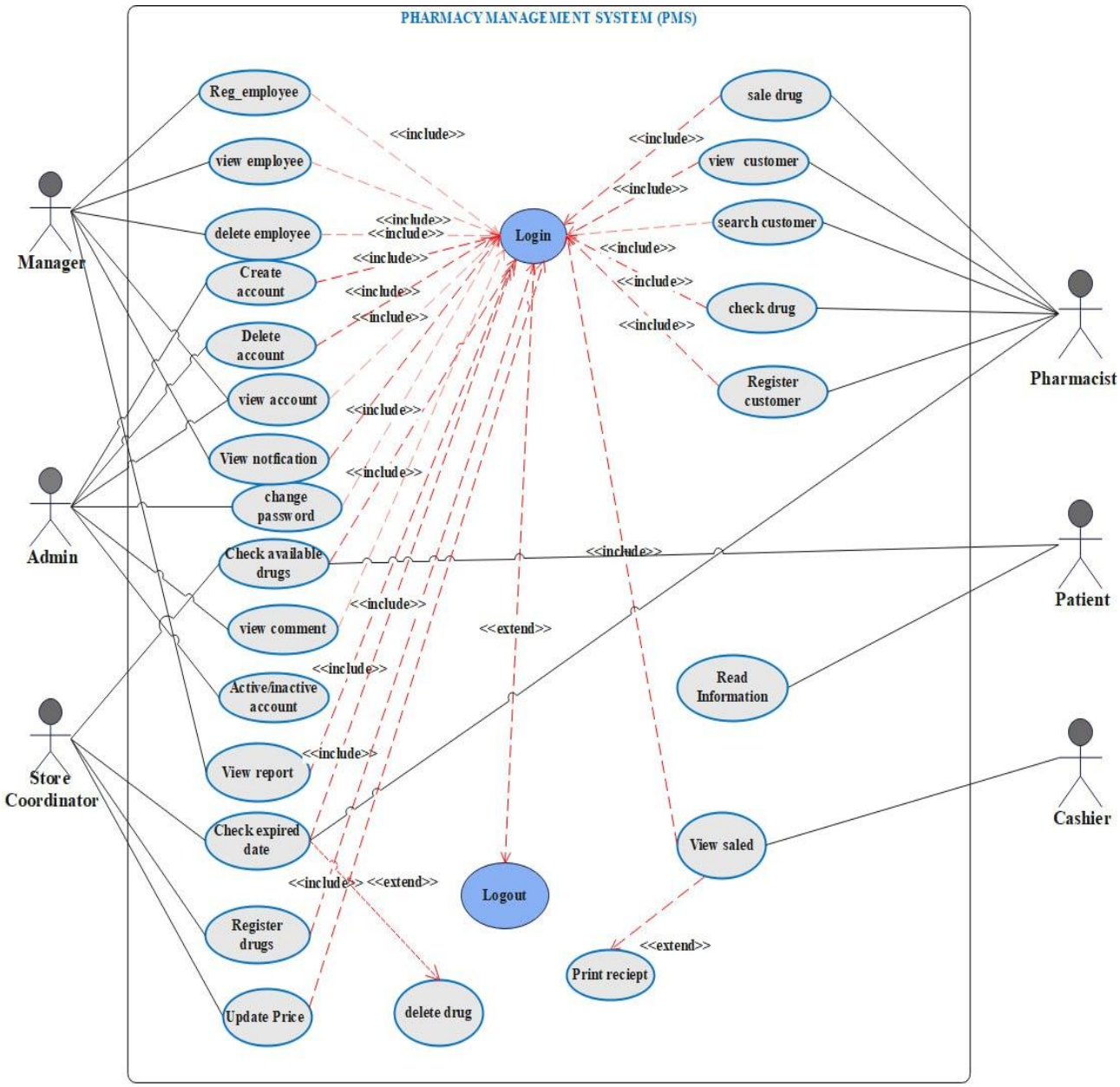


Figure 3 Use case diagram of the project

2.2.5. Use case documentation

A use case is a description of how a user interacts with a system or product. Companies build use cases to establish success scenarios, failure scenarios, and any important variants or exceptions. The use case documentation also provides a detailed description of the interactions between the system and its users or external systems, including the preconditions and post-conditions for each use case scenario. This information is critical in understanding the behavior of the system and in ensuring that it meets the needs of its users [12].

Table 4 Use case system for Login

Use case Name:	**Login**	
UID	**UC#01**	
Actor	**Manager, Admin, Pharmacist, store coordinator, cashier and Patient**	
Description	In order to get into or access the system	
Precondition	The Patient, Admin, Manager, Store Coordinator, Pharmacist, or Cashier must possess valid credentials (username and password) to log in	
Flow of events	**Actor action**	**System response**
	1. The user selects the Login option 3. The user completes the Login form and submits it.	2. The system shows the login interface. 4. The system verifies if the Username and Password fields are left blank 5. The system checks the validity of the provided username and password 6. The system navigates to the appropriate page.

Alternative action	If the provided username or password is invalid: 1. The system displays an error message and returns to the login screen. 2. The use case terminates.
Post condition	Successful access to the system.

Table 5 Use case system for Logout

Use case Name:	**Logout**	
ID	**UC#02**	
Actors	**Manager, Admin, Pharmacist, Cashier, Store-coordinator, Patient**	
Description	To exit the system securely.	
Precondition	The user must be logged into the system.	
Flow of events	**User Actions**	**System Actions**
	1. The logged-in user decides to log out of the system. 2. The user clicks the "Logout" button.	3. The system terminates the current session. 4. The user is redirected to the login screen or homepage. 5. The use case concludes.
Alternative action	If the user closes the application or browser without clicking "Logout": 1. The session ends automatically after a predefined timeout period or upon the next system access. 2. The use case concludes.	
Post condition	The user is logged out and the session is closed securely.	

Table 6 Use case system for Register drug

Use case Name:	**Register Drug**	
UID	**UC#03**	
Actors	**Store coordinator**	
Description	Adding a new drug from the store to the database.	
Pre-condition	Enter a username and password to log in to the system.	
Flow of event	**Actor Action**	**System Response**
	1. The Store Coordinator accesses your page. 2. The Store Coordinator selects the "Register Drug" link. 5. The Store Coordinator completes the form fields. 6. Finally, they click the "Submit" button.	3. The system displays the drug registration form. 7. The system checks for any empty fields. 8. The system verifies the validity of the entered data 9. The system displays a success message upon completion.
Alternative action	1. The system displays the message, “Please complete the form fields!” The Store Coordinator continues from step 5. 2. The system displays the message, “Incorrect data entered!” The Store Coordinator resumes from step 5.	
Post condition	Store coordinator successfully register.	

Table 7 Use case system for view drug

Use case Name:	**View drug**	
UID	**UC#04**	
Actor	**Patient, Store coordinator and pharmacist**	
Description	Pharmacist, Patient and Store coordinator View available drug in the store	
Precondition	Pharmacist, Customer and Store coordinator should login the system.	
Flow of event	**Actor action**	**System response**
	1. Click on "View Like." 3. Complete the form. 5. Press the "View" button	2. Display the "View Drug" form. 4. The system verifies if any required fields are empty. 6. The system validates the entered data. 7. Show the available drugs.
Alternative action	No	
Alternative case	1: Missing Information Alert. The system shows a message: "Please complete all required fields in the form!" The process returns to step 3. 2: Invalid Data Warning. The system displays: "Invalid data entered. Please try again." The workflow resets to step 3.	
Post condition	The Patient, pharmacist, and store coordinator can access and view the available drugs within the system.	

Table 8 Use case system for Sale drug

Use case Name:	**Sale drug**	
UID	**UC#05**	
Actor	**Pharmacist**	
Description	The pharmacist sells drugs to the customer.	
Precondition	The pharmacist logs in to the system.	
Flow of event	**Actor action**	**System response**
	1. Navigate to the pharmacist page. 3. Click on the "Sale Drug" link. 5. Complete the form and click the "Sale" button.	2. The pharmacist's home page is displayed. 4. The sales form is shown. 6. The system verifies if any fields are empty. 7. The system checks the validity of the entered data.
Alternative action	1. The system shows the message: "Please complete the field!" The pharmacist returns to step 5. 2. The system displays the message: "Incorrect data entry!" The pharmacist returns to step 5.	
Post condition	The pharmacist sells drugs to the customer.	

Table 9 Use case system for create account

Use case Name:	**Create Account**	
UID	**UC#06**	
Actor	**Admin**	
Description	The admin oversees the account.	
Precondition	The admin signs into the system.	
Flow of event	**Actor action**	**System response**
	1. Navigate to the Admin page. 3. Click on the "Create Account" link. 5. Complete all the required fields. 6. Click the "Sign Up" button.	2. The system shows the admin login page. 4.The system displays the account creation form. 7. The system checks for any empty fields. 8. The system validates the entered data. 9. The system shows the message, "Your account has been successfully created."
Alternative action	1. The system shows the message, "Please provide all the required information!" The admin returns to step 5. 2. The system displays the message, "Incorrect data entered!" The admin returns to step 5.	
Post condition	The admin sets up an account for the user.	

Table 30 Use case system for Employee registration

Name	**Employee Registration**	
UID	**UC#07**	
Actor's	**Manager**	
Description	Register the details of the workers in the pharmacy.	
Pre-condition	Have user name and password	
Flow of event	**Actor action**	**System action**
	1. The manager visits or logs into the registration page. 2. The manager clicks on the "Register Employee" link. 4. The manager enters the required employee information and clicks the "Register" button.	3. The system shows the employee registration form. 5. The system verifies if any fields are left empty. 6. The system validates the entered data. 7. The system displays the message: "Employee registered successfully."
Alternative action	1. The system shows the message: "Please complete the required fields!" The manager returns to step 4. 2. The system displays the message: "Incorrect data entered!" The manager returns to step 4	
Post condition	The manager successfully registers the pharmacy employee.	

Table 11 Use case system Delete employee

Name	**Delete Employee**	
UID	**UC#08**	
Actor's	Manager	
Description	Delete the employee when it is necessary.	
Pre-condition	The Manager can successfully login to the System.	
Flow of event	**Actor action**	**System action**
	1. The manager click "Delete Employee" link. 3. The manager fill the form filled then click "Delete" button.	2. The system can display Delete employee form. 4. The system can check empty filed **[1]** 5. The System can Validate the entered data **[2].** 6. The system can display Successfully message.
Alternative Action	**1:** Information Not Filled Message The system displays "Please fill the form!" message. The Manager resumes at **step 3**. 2: Invalid data The system displays "you have entered invalid data" message. Manager resume at **step3**	
Alternative action	No	
post condition	Employee can delete from the data base.	

Table 12 Use case system for patient registration

Name	Patient Registration	
UID	**UC#09**	
Actor's	**Pharmacist**	
Description	Register the patient who arrives at the pharmacy.	
Pre-condition	The pharmacist should have login into the system	
Flow of event	**Actor action**	**System action**
	1. The pharmacist selects the "patient Registration" link. 3. The pharmacist completes the form fields and presses the "Submit" button.	2. The system shows the patient registration form. 4. The system verifies if any fields are empty. 5. The system checks the validity of the entered data. 6. The system displays a success message.
Alternative action	1: Invalid Data The system shows the message: "You have entered invalid data." The pharmacist returns to step 3. 2: Missing Information Alert The system displays the message: "Please complete the form!" The pharmacist goes back to step 3.	
Post condition	Successfully register the customer.	

Table 13 Use case system for Check expired date

Use case Name:	**Check Expired Date**	
UID	**UC#10**	
Actors	**Store coordinator**	
Description	To check the drug's expiration date.	
Pre-condition	Start the system with a username and password.	
Flow of event	**Actor Action**	**System Response**
	1. The store coordinator clicks on the "Check Expiry Date" link. 3. The store coordinator clicks the "Delete" button.	2. The system shows the list of drugs with their expiration dates. 4. The system then displays a success message
Alternative action	No	
Alternative case	The system displays the message: "There are no expired drugs."	
Post Condition	The store coordinator checks the drug's expiration date and removes expired drugs from the store.	

Table 14 Use case system for print

Use Case Name:	**Print**	
UID	**UC#11**	
Actors	**Cashier**	
Description	Printing the receipt for the drugs sold to the patient.	
Pre-condition	1. The cashier must have a username and password. 2. A list of drugs that need to be sold by the pharmacist must be available.	
Flow of event	**Actor Action**	**System Response**
	1 The cashier navigates to their page. 3. The cashier clicks on the "Fetch Order Drug" link. 5. The cashier enters the patient ID and clicks the "Search" button. 9. The cashier clicks the "Print" button. 10. The cashier prints the receipt and hands it to the patient.	2. The system displays the cashier's page. 4. The system shows the form. 6. The system checks if any fields are empty. 7. The system validates the entered data. 8. The system displays the sold drugs along with the corresponding total price.
Alternative case	1. The system displays the message: "Please complete the field!" The pharmacist returns to step 5. 2. The system shows the message: "Incorrect data entry!" The pharmacist goes back to step 5.	
Post condition	The cashier prints the sold drugs and provides the receipt to the patient.	

1.5.2. Sequence diagram

A sequence diagram is a type of UML (Unified Modeling Language) diagram that illustrates how objects or components interact with each other over time. It shows the sequence of messages exchanged between objects to complete a specific task or process. The diagram represents the flow of control, highlighting the order of interactions, typically between a user, system components, and external entities. Sequence diagrams are useful for visualizing the behavior of a system and understanding the sequence of operations within a specific use case or scenario [6].

Figure 4 login sequence diagram

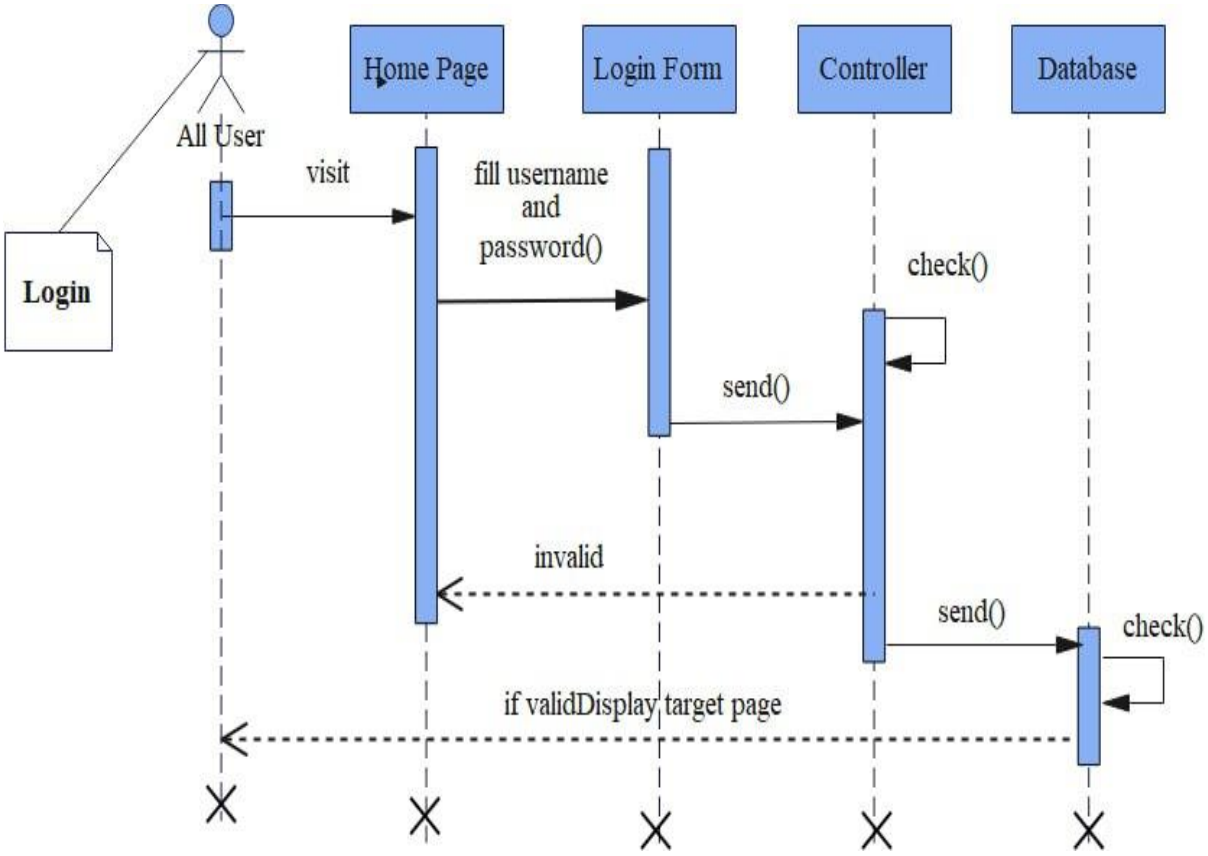


Figure 5 create account sequence diagram

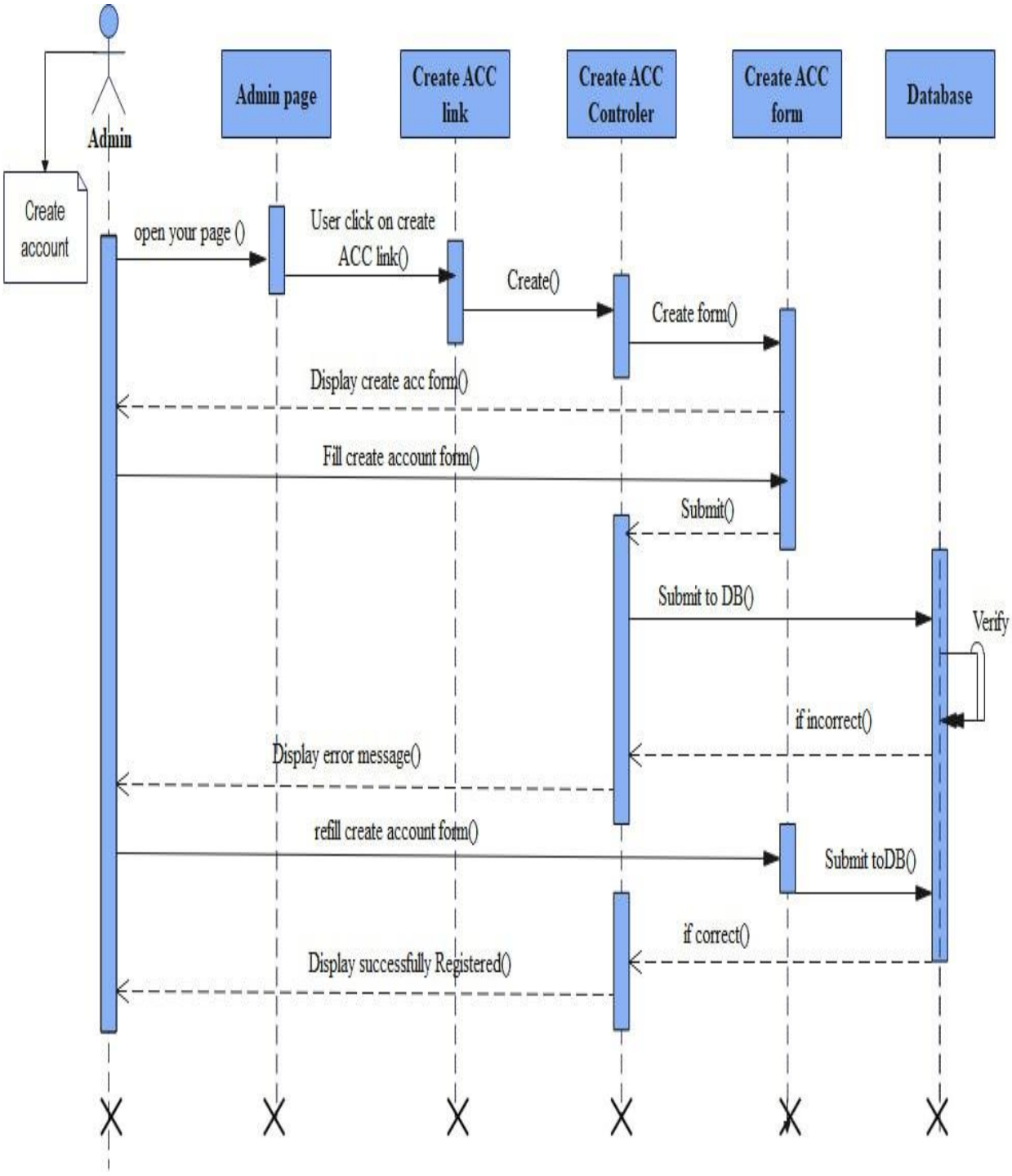


Figure 6 sale drug sequence diagram

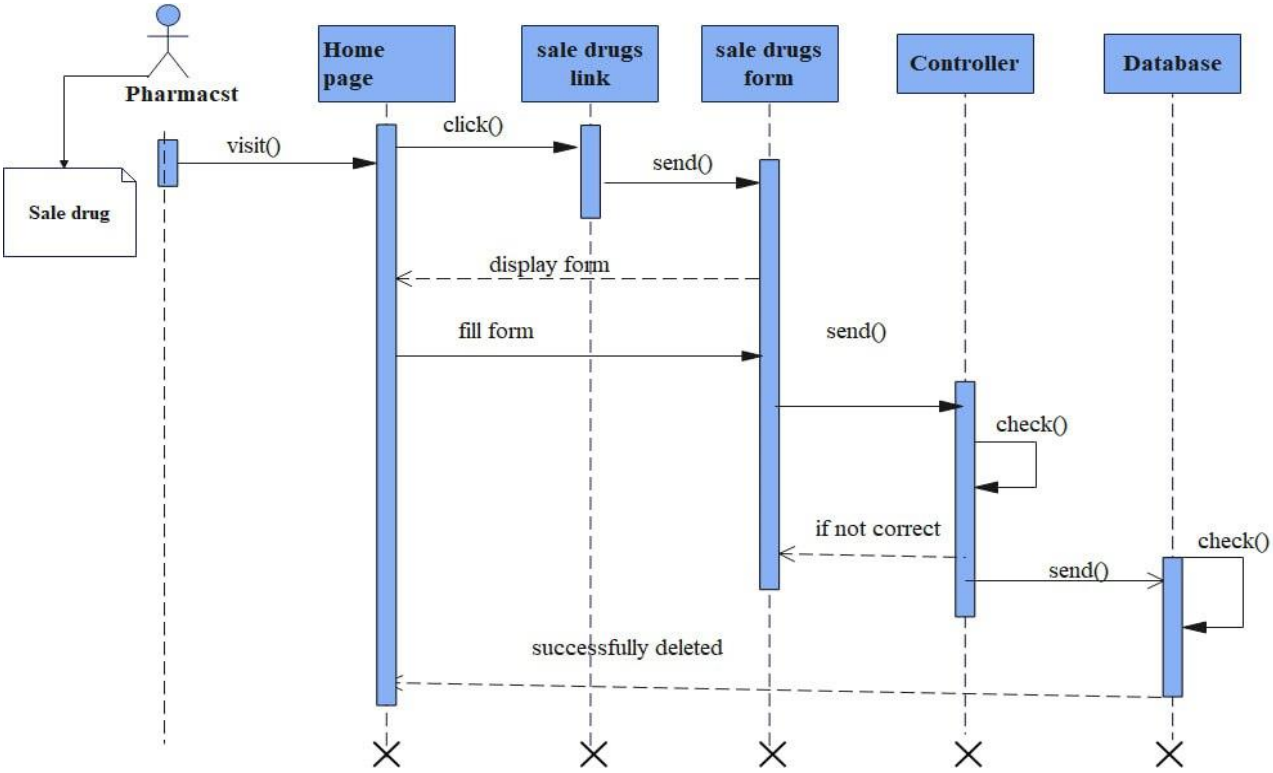


Figure 7 drug register sequence diagram

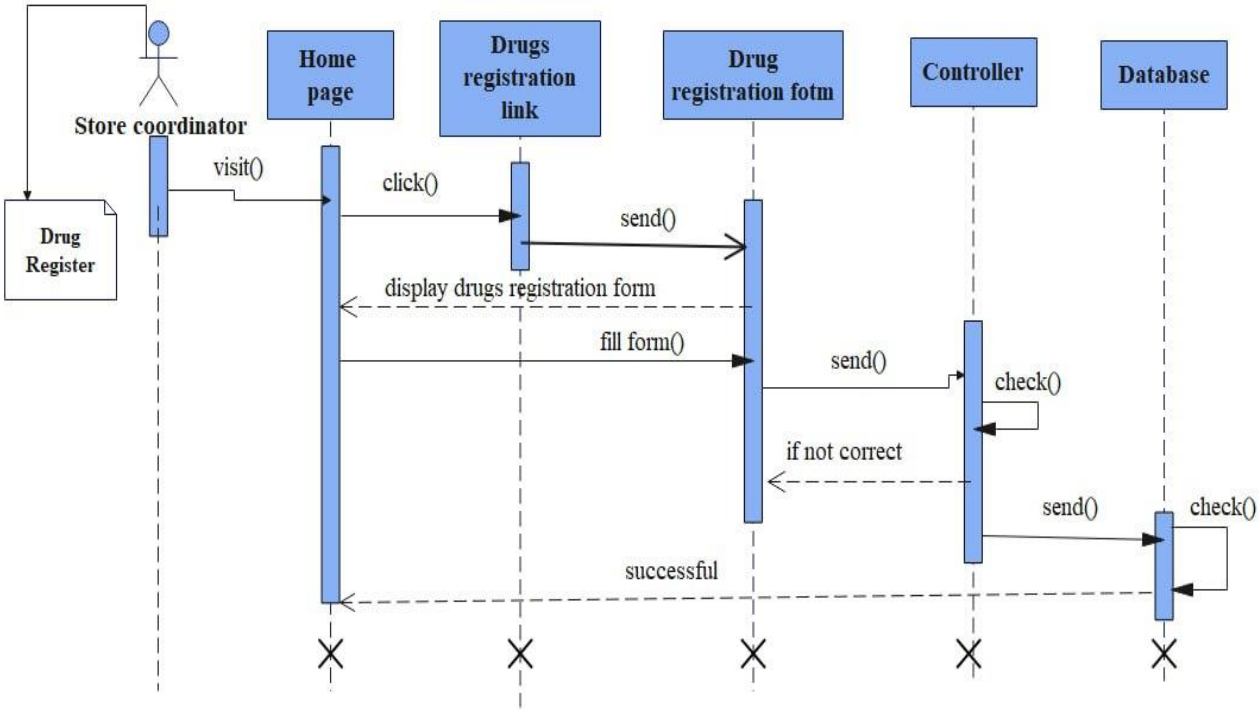
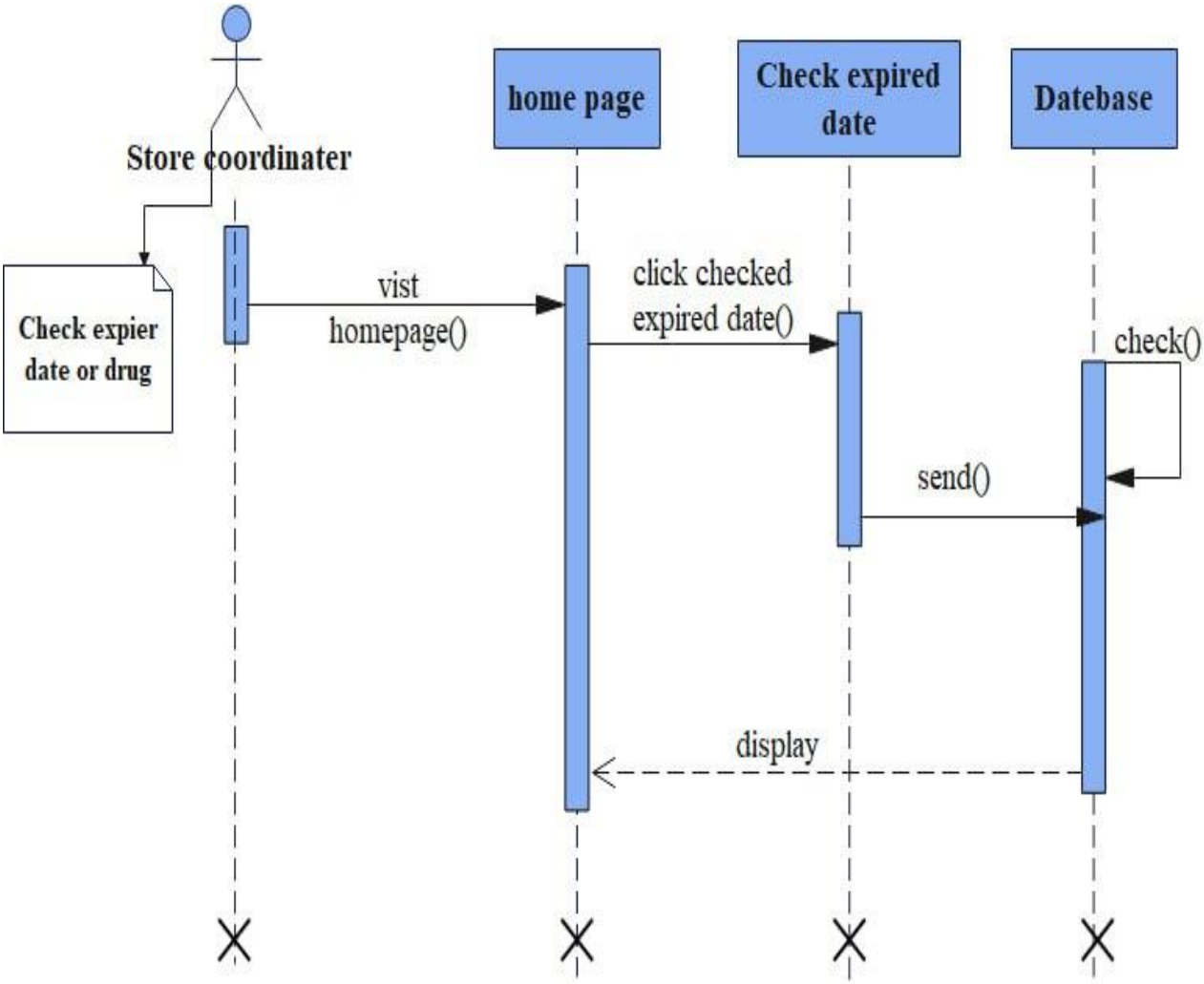


Figure 8 check expire-date of drug sequence diagram



2.5.3. State chart diagram

A state chart diagram, or state machine diagram, is a UML tool used to model the dynamic behavior of a system or object by illustrating its states and transitions. It shows how an object moves between states in response to events or conditions, highlighting the lifecycle of the object. Key elements include states, transitions, events, an initial state (where the flow begins), and a final state (where it ends). State chart diagrams are essential for understanding the behavior of system components and clarifying complex workflows [12].

Figure 9 Login state chart

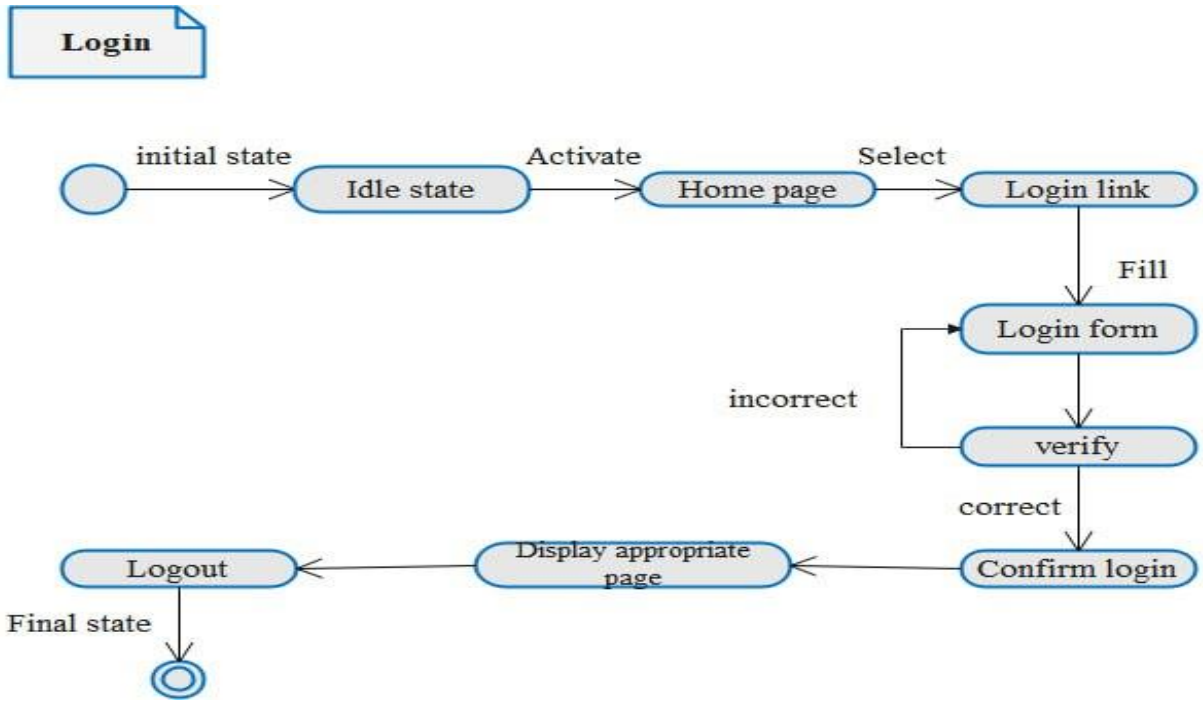


Figure 10 Create Account state chart

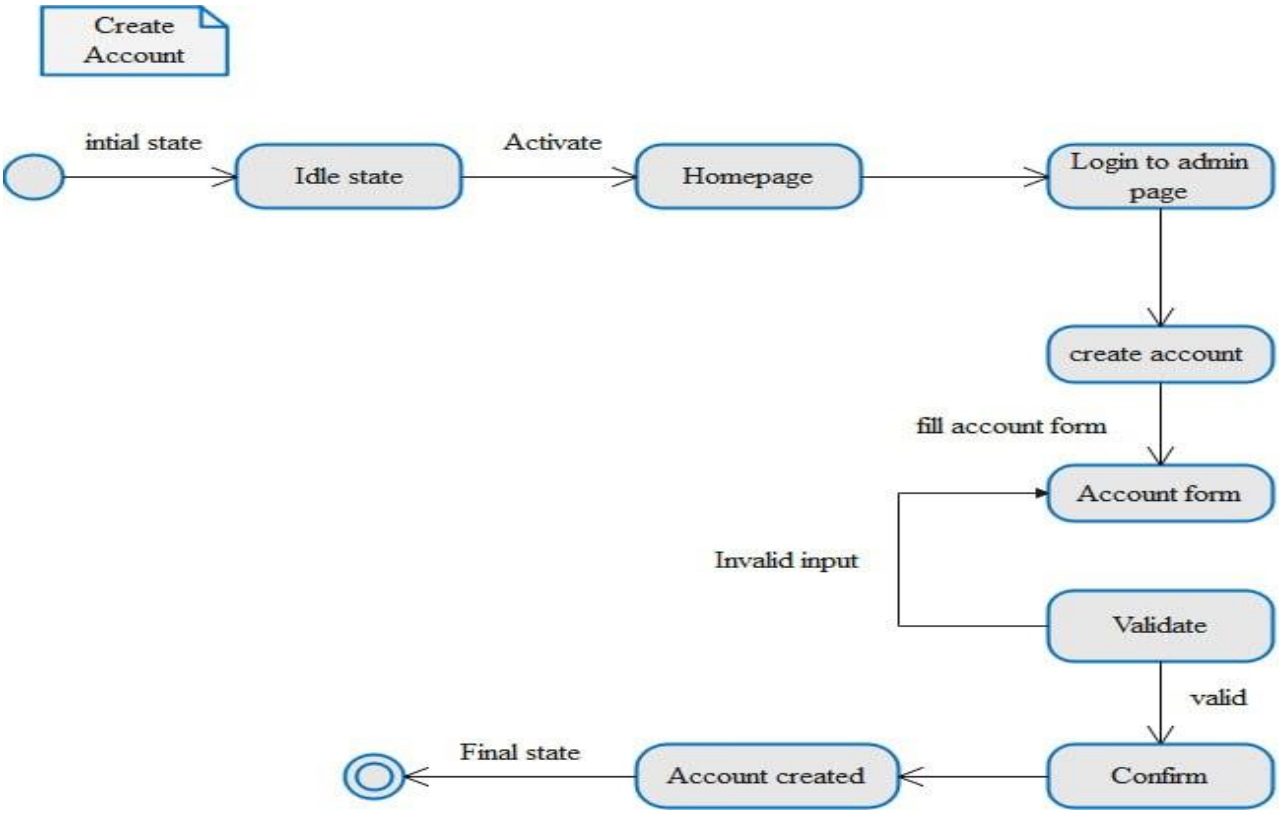


Figure 11 Sale drug state chart

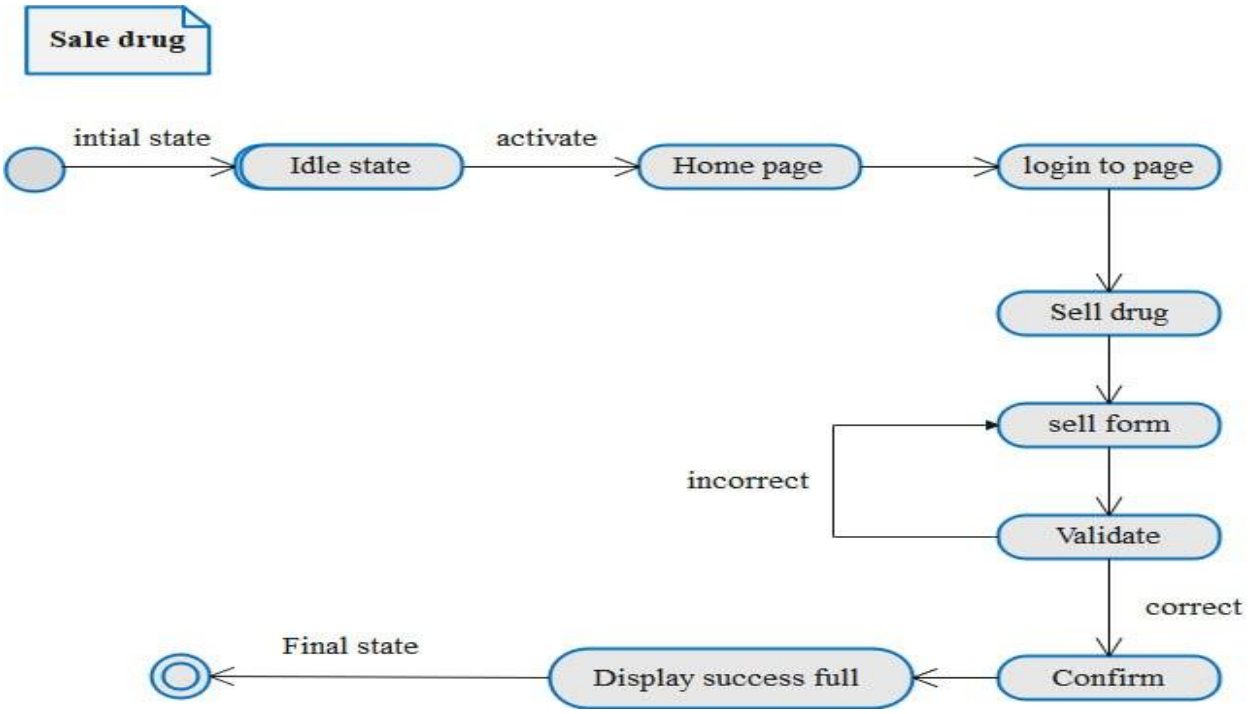
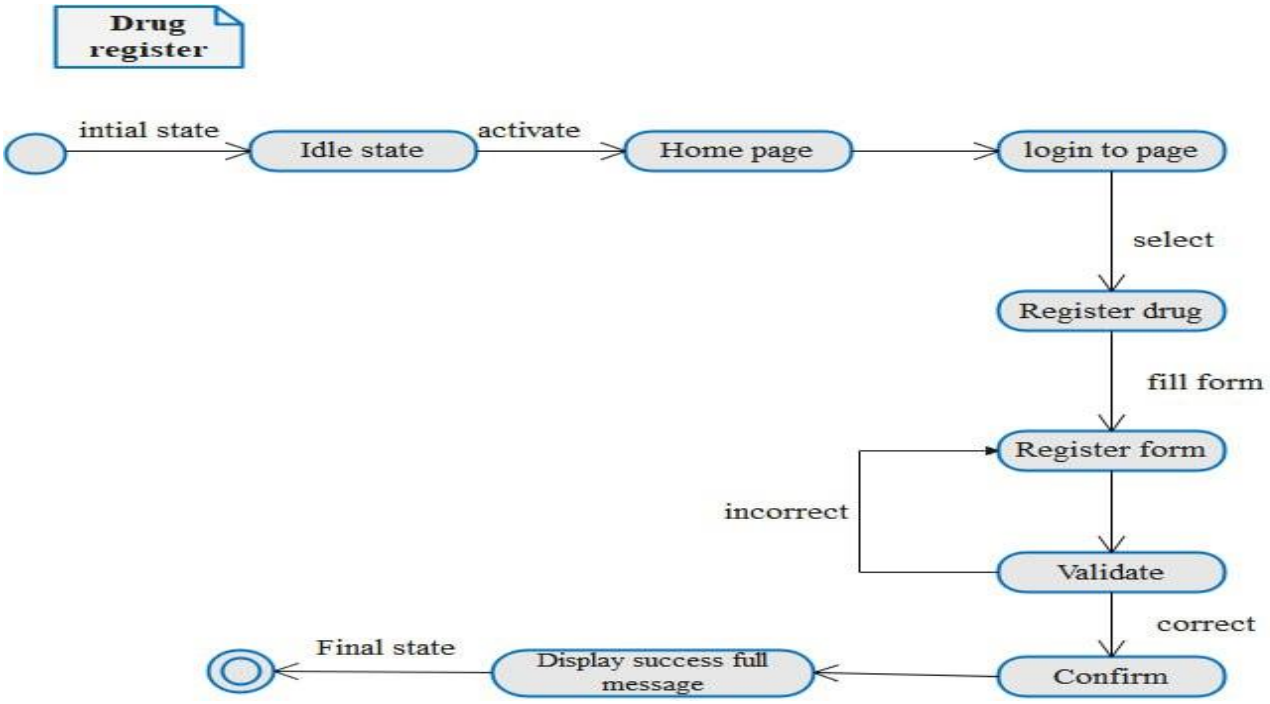


Figure 12 Drug register state chart



2.5.4. Activity diagram

Activity diagram is another important behavioral diagram in UML diagram to describe dynamic aspects of the system. Activity diagram is essentially an advanced version of flow chart that modeling the flow from one activity to another activity [12].

Figure 13 login activity diagram

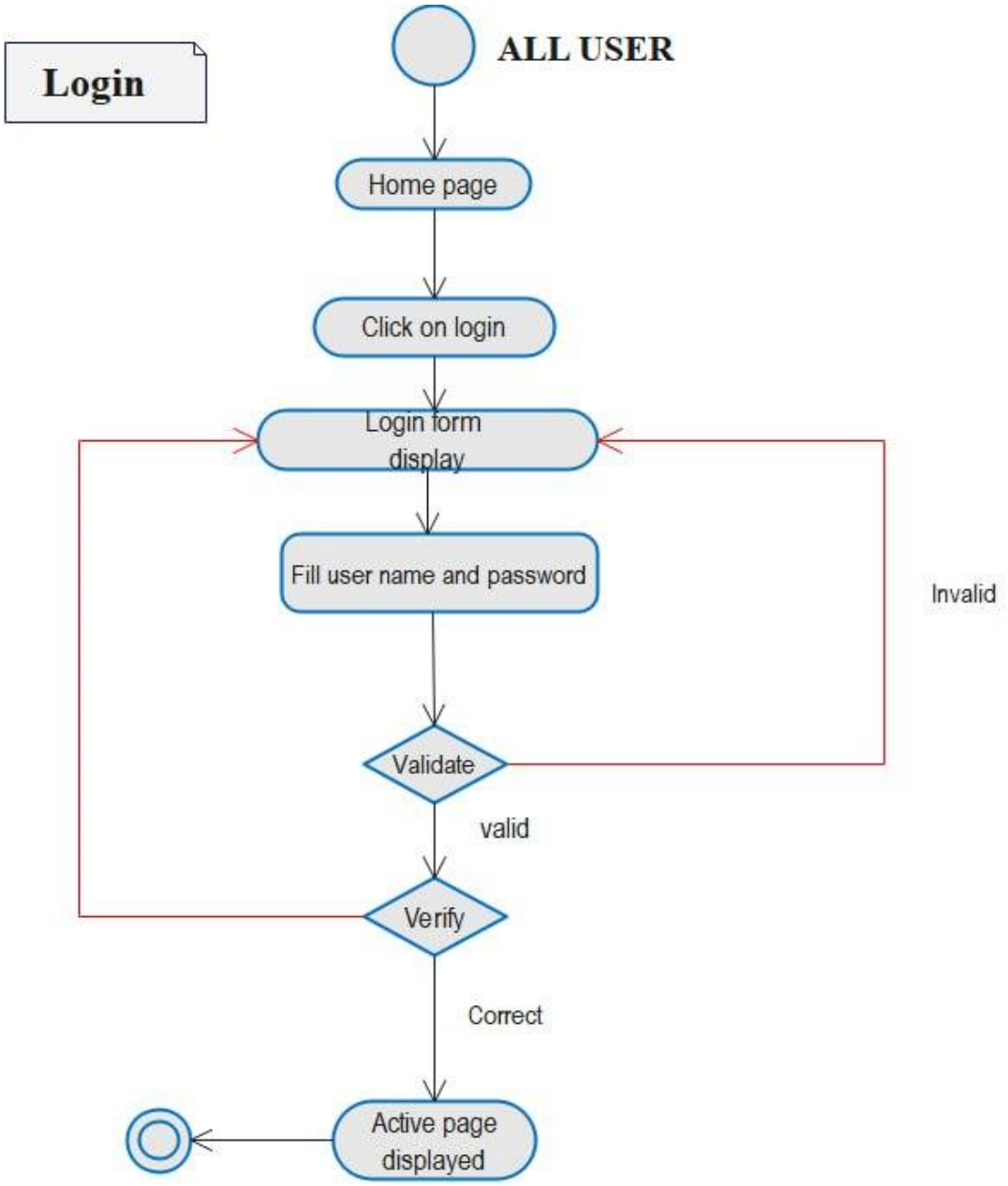


Figure 14 Create account activity diagram

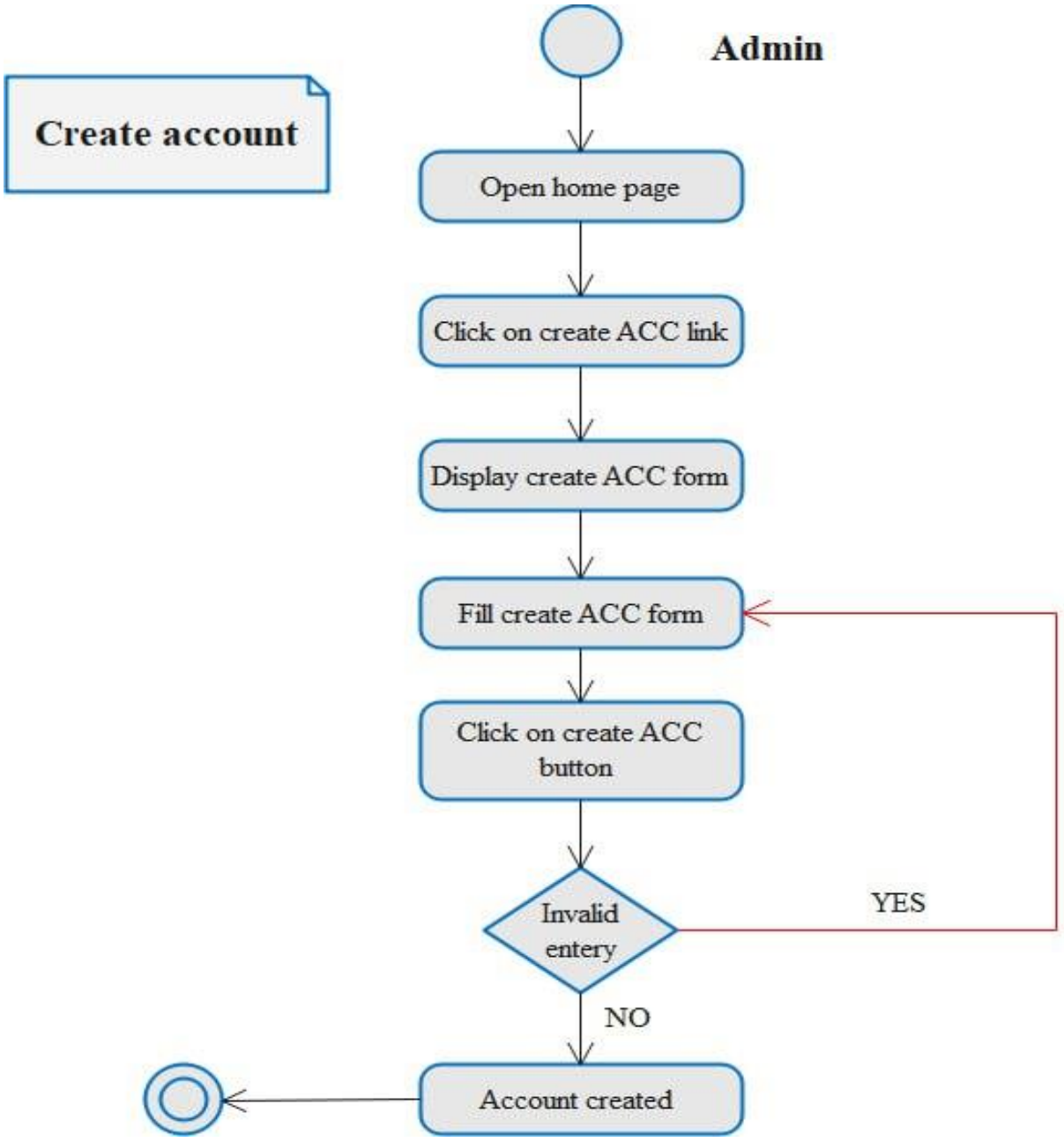


Figure 15 Sale drug activity diagram

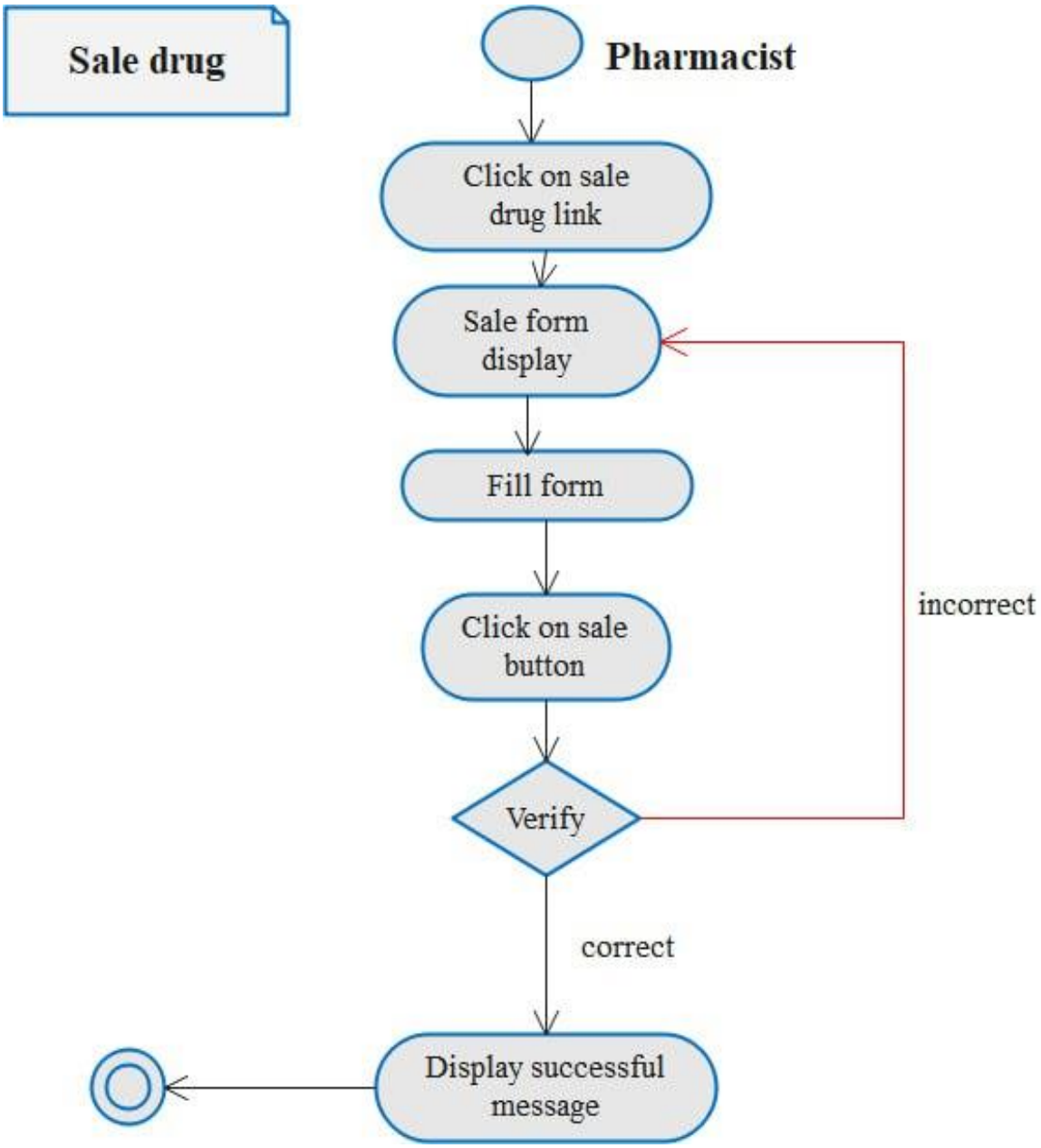
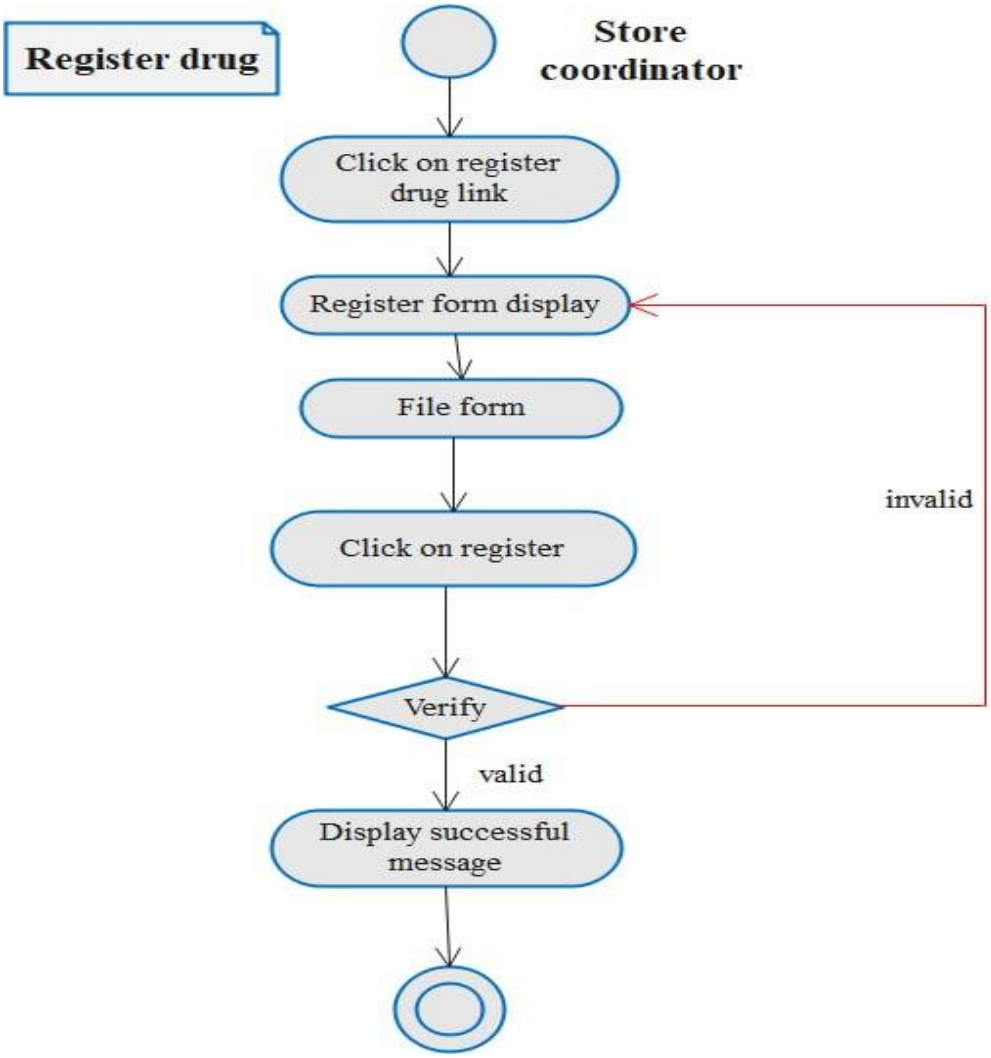


Figure 16 Register drug activity diagram



2.6. key abstraction with CRC analysis

CRC (Class-Responsibility-Collaborator) analysis is an effective method used in object-oriented design to model the components of a system. In the Pharmacy Management System for Salale University Comprehensive Specialized Hospital, CRC analysis identifies key classes, their responsibilities, and how they interact to perform essential tasks. This approach helps define the system's structure by connecting real-world entities like managers, admins, pharmacists, and patient to their system counterparts [1].

The classes in the system include actors like the **manager, admin, pharmacist, store coordinator, cashier, and patient**. Each class is assigned specific responsibilities based on its

real-world role. For instance, the **manager** oversees the entire system, ensuring smooth operations and reviewing reports, while the **pharmacist** sale and dispenses drugs, maintaining accurate inventory. The **admin** is responsible for user management, including creating and deactivating accounts, resetting passwords, and overseeing system access control. These classes collaborate seamlessly—for example, the **pharmacist** works with the **store coordinator** to monitor stock, the **cashier** interacts with the **manager** for reporting purposes, and the **admin** ensures secure access and system integrity.

By defining clear responsibilities and collaborations, CRC analysis ensures a cohesive and efficient system design. It enhances communication between system components, simplifies development, and prepares the system for future scalability. This structured approach ensures that the **pharmacy management system** effectively addresses the hospital's operational needs while maintaining security and efficient workflow.

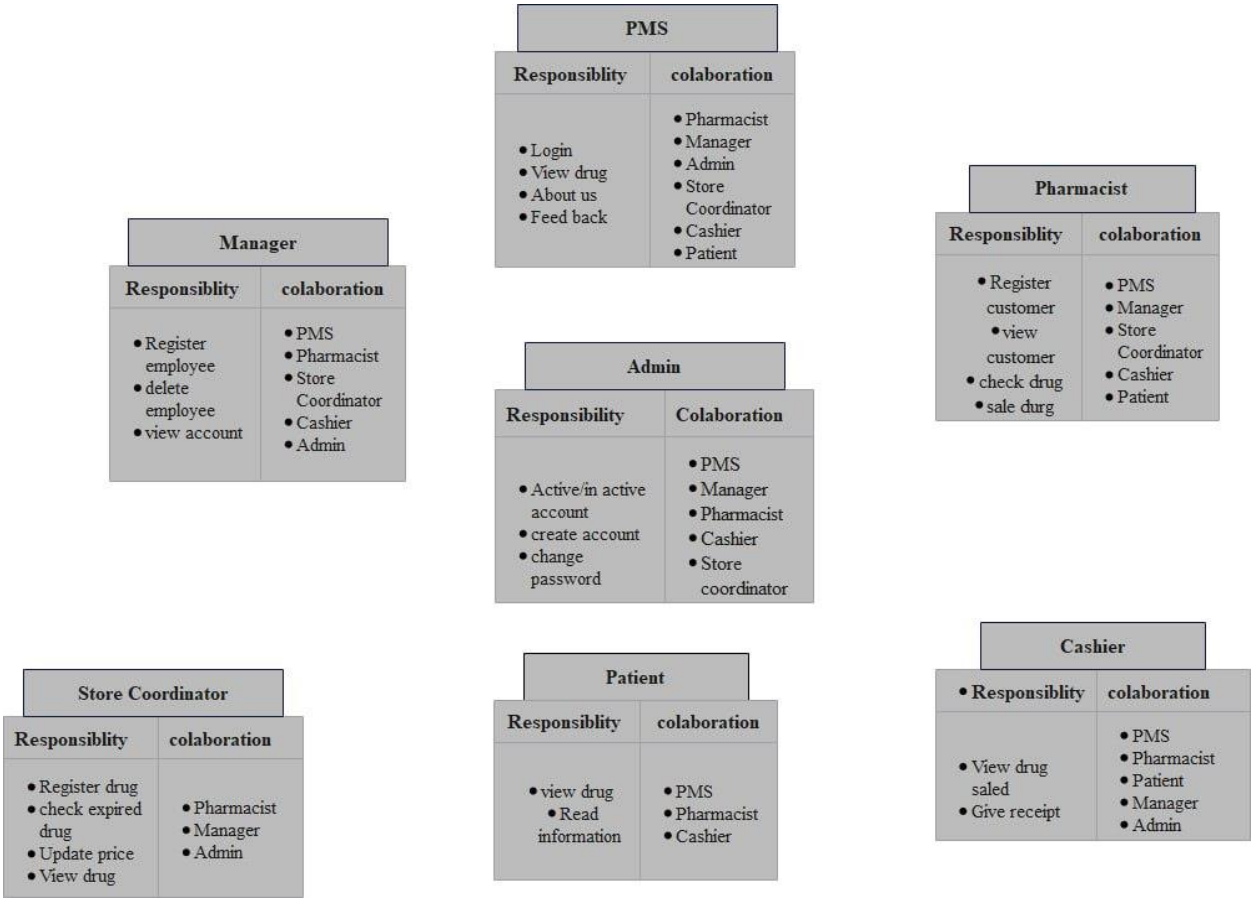


Figure 17 CRC Diagram

2.4 Class diagram

A **class diagram** represents the static structure of a system by illustrating its classes, attributes, methods, and relationships. Below is the conceptual **class diagram** for the Pharmacy Management System of Salale University Comprehensive Specialized Hospital [9].

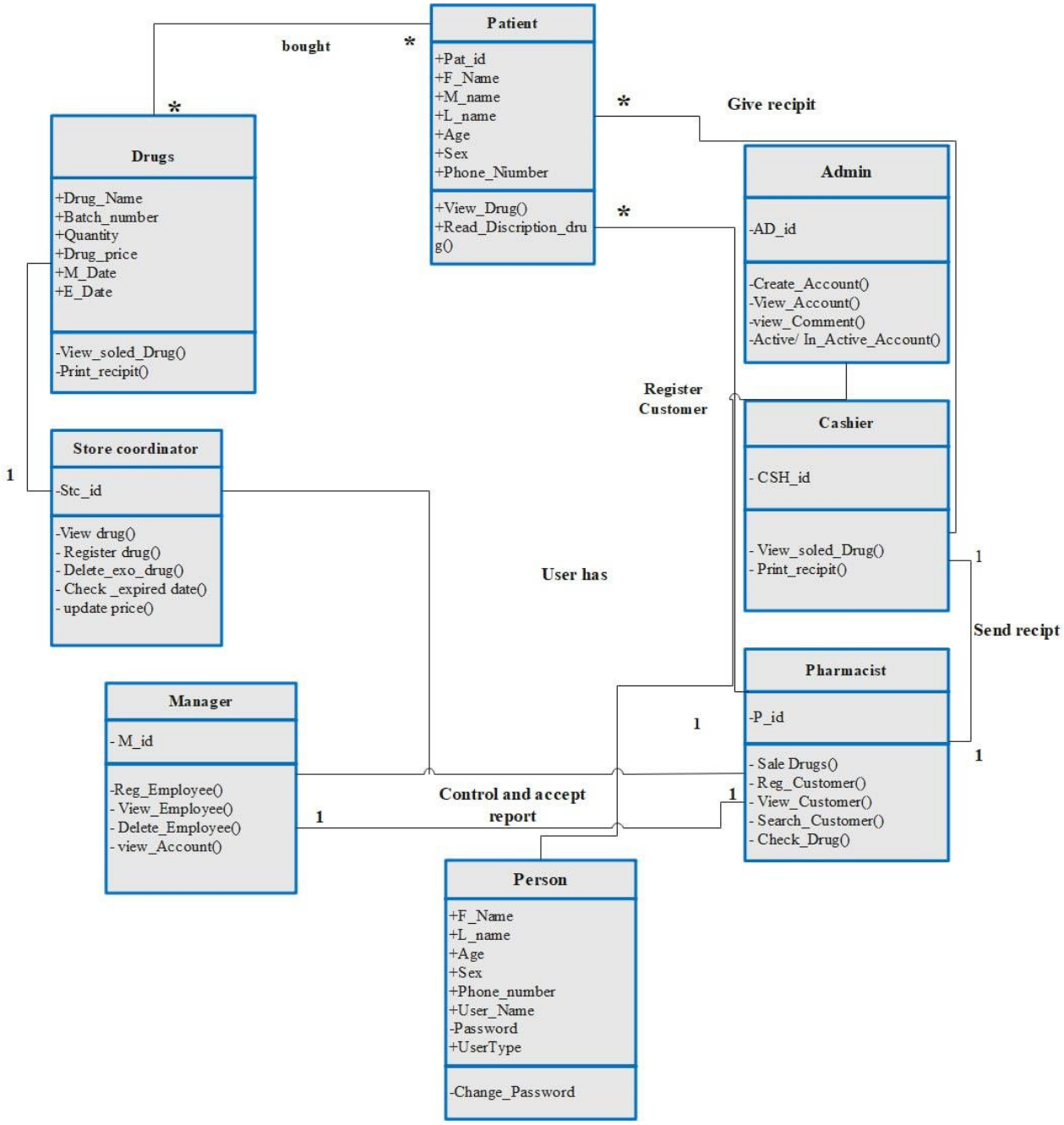


Figure 18 Class diagram

2.5 Identifying change cases

1. **Change in Drug Registration Process**
    - ✓ **Description**: The system might require updates to support new fields or compliance standards for drug registration, such as additional regulatory information.
    - ✓ **Likelihood**: Moderate, as pharmaceutical regulations often change.
    - ✓ **Impact**: High, as it affects the core functionality of registering drugs.
2. **Change in Reporting Formats**
    - ✓ **Description**: The system may need to generate reports in new formats or include additional data fields based on management requirements.
    - ✓ **Likelihood**: High, due to evolving business and audit needs.
    - ✓ **Impact**: Medium, requiring updates to the reporting module.
3. **Support for New Drug Categories**
    - ✓ **Description**: Including new classifications or special handling requirements for certain drug types, such as controlled substances.
    - ✓ **Likelihood**: Moderate, as the introduction of new drug categories is periodic.
    - ✓ **Impact**: Medium, requiring updates to inventory and management modules.
4. **Enhanced Security Measures**
    - o **Description**: Implementation of stronger authentication or encryption methods to meet new cybersecurity standards.
    - o **Likelihood**: High, due to increasing focus on data security.
    - o **Impact**: High, as it involves updates across user access and data handling.
5. **Data Entry Updates**: When new information is manually input into the system, such as customer details, inventory records, or employee roles, it's essential to track changes. For

example, a cashier manually updates the patient's purchase details or payment status. Any modification in records (e.g., correcting an address) needs to be logged for auditing purposes.

2.6 User Interface Prototyping

A **prototype** of your pharmacy management system for Salale University Comprehensive Specialized Hospital serves as a preliminary model to demonstrate and test the system's features and functionality. It provides a tangible representation of how the system will operate, enabling stakeholders to visualize the interface, workflows, and interactions between various actors [8].

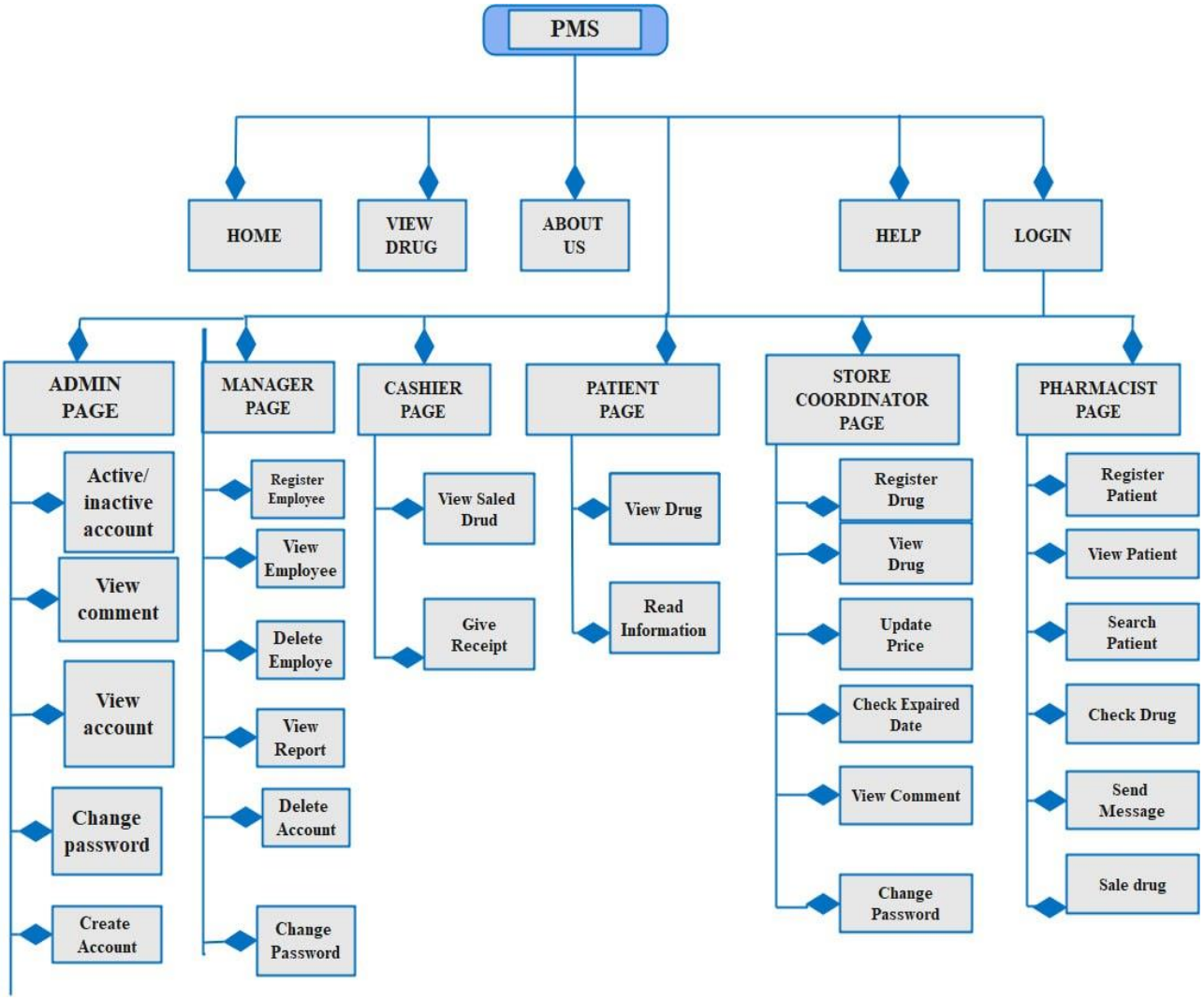


Figure 19 PMS user interface prototype

CHAPTER THREE

3. OBJECT ORIENTED SYSTEM DESIGN

3.1. Introduction

This phase focuses on creating a blueprint for how the system will work, detailing its structure, functionality, and interactions between components to meet the specified requirements. System design bridges the gap between problem domain and solution implementation, ensuring the system is both effective and efficient.

The design process defines the architecture, data flow, user interface layouts, and components required for the system. It considers both functional and non-functional requirements, ensuring scalability, security, and performance. This design also provides a clear understanding of how different actors (Manager, Admin, Pharmacist, Store Coordinator, Cashier, and Patient) will interact with the system and how their workflows will be supported.

By carefully planning and documenting the system's architecture, components, and their relationships, this phase lays the foundation for seamless development and deployment, minimizing potential issues during implementation [11].

3.2 Purpose and Goals of Design

The **Pharmacy Management System** for Salale University Comprehensive Specialized Hospital is designed to streamline operations, enhance accuracy, and improve the overall efficiency of the pharmacy. The system aims to address the challenges of manual processes while ensuring user satisfaction and system reliability. Below are the key design goals, categorized by relevant criteria:

3.1.1 Performance

The Pharmacy Management System is designed to deliver high-performance capabilities by ensuring quick data retrieval, seamless transaction handling, and optimized response times. This allows users to access information without delays, even during peak usage periods, enhancing operational efficiency and patient satisfaction.

3.1.2. Dependability

The system prioritizes dependability by implementing robust error-handling mechanisms, ensuring minimal downtime, and maintaining accurate data records. It is built to consistently function under various circumstances, providing a reliable solution that users can trust to manage critical pharmacy operations.

3.1.3. User Experience

A user-friendly interface is a core focus, ensuring that all users, including managers, admin, pharmacists, and cashiers, can navigate the system effortlessly. Clear workflows, intuitive design, and accessibility features make the system easy to use, regardless of technical proficiency.

3.1.4 Security

The system is equipped with advanced security measures, including user authentication, role-based access controls, and data encryption. These features ensure the protection of sensitive information, such as customer records and financial data, safeguarding it from unauthorized access and breaches.

3.1.5. Scalability

The architecture of the system is designed to accommodate future growth, whether through the addition of new users, integration of additional functionalities, or expansion to other departments. This ensures that the system remains effective as the hospital's needs evolve.

3.1.6. Maintainability

To facilitate long-term efficiency, the system employs a modular architecture that simplifies updates, debugging, and the incorporation of new features. This design choice ensures that maintenance activities are both time and cost-effective.

3.1.7. Cost-Effectiveness

By utilizing open-source tools and resource-efficient methods, the system minimizes development and operational costs while maximizing value. This approach ensures that the hospital can sustain the system within its budget constraints.

3.2. Current software Architecture

The current software architecture of the pharmacy management system at Salale University Comprehensive Specialized Hospital is primarily manual, relying heavily on traditional methods for handling operations. In this setup, activities such as drug registration, stock management, report generation, and patient interactions are carried out through paper-based documentation and verbal communication. Drug inventory is tracked manually using registers or logbooks, making it prone to human error and inefficiency. Similarly, patient records and transactions are managed without the support of a computerized system, which complicates retrieving historical data or analyzing trends. Reports are generated manually, consuming significant time and often leading to inaccuracies due to redundant or inconsistent data.

This architecture lacks the capability to provide real-time updates, efficient record retrieval, and dependable accuracy. As a result, it introduces challenges such as delays in service delivery, difficulty in tracking expired drugs, and risks of data loss. This manual approach underscores the urgent need for transitioning to a modernized, computerized system to improve efficiency, accuracy, and overall service quality.

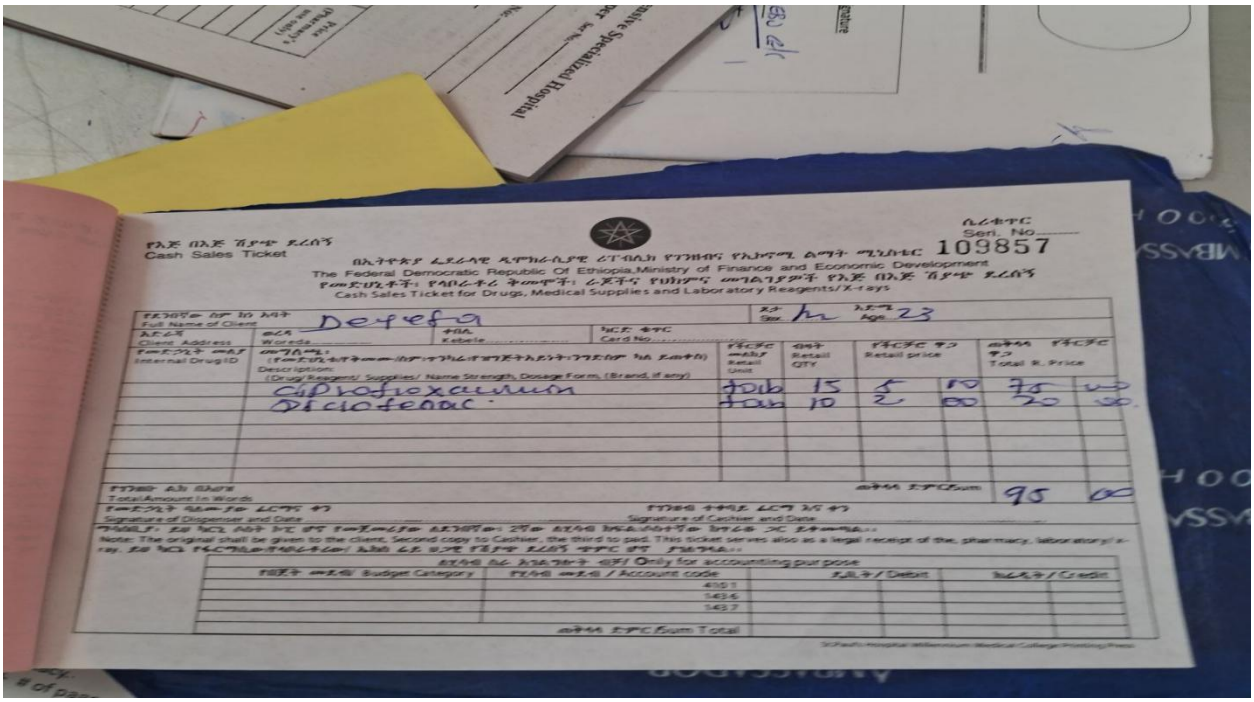


Figure 20 Current software architecture

3.3 Proposed System Architecture

The proposed system architecture consists of three main layers: **Presentation Layer**, **Application Layer**, and **Database Layer**. These layers work together to deliver a seamless and efficient Pharmacy Management System for Salale University Comprehensive Specialized Hospital.

1. Presentation Layer

The presentation layer acts as the interface between the users and the system. It provides a user-friendly and intuitive design tailored for different user roles, such as managers, admin, pharmacists, store coordinators, cashiers, and customers [3]. This layer ensures:

- **Web-based or desktop-based interfaces** for interaction.
- Accessibility and responsiveness for diverse devices and user needs.
- Role-specific dashboards, such as sales tracking for cashiers and inventory views for store coordinators.

2. Application Layer

The application layer handles the core business logic and processing. It acts as the intermediary between the presentation and database layers [2]. This layer is responsible for:

- Validating and processing user inputs from the interface.
- Executing operations such as drug registration, stock updates, and report generation.
- Managing role-based permissions to ensure secure and accurate data handling.

3. Database Layer

The database layer serves as the central repository for all system data. It securely stores and organizes critical information, ensuring integrity and reliability [4]. Key functionalities of this layer include:

- Storing inventory details, customer records, employee data, and sales transactions.
- Providing efficient query execution to retrieve data quickly.

- Ensuring data consistency, security, and backups to prevent data loss.

Together, these three layers form a robust architecture that streamlines pharmacy operations, enhances data accuracy, and improves overall system performance [2]

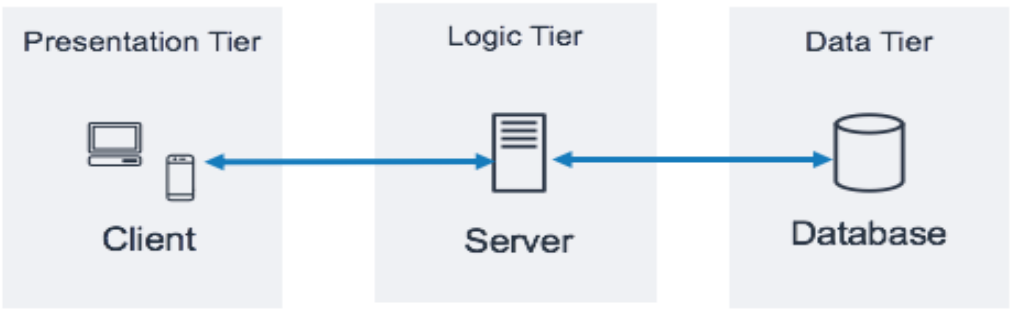


Figure 21 PMS Tier

3.3.1 Subsystem decomposition

The **Pharmacy Management System (PMS)** is divided into six primary subsystems, each associated with a specific actor. These subsystems define the roles and responsibilities of each user type within the system, ensuring efficient management and operation of the pharmacy. The diagram illustrates how different actors interact with the system through their respective functional areas [9].

1. Admin Subsystem

The **admin** is responsible for user management and system security. This subsystem includes:

- **View Comment:** Allows the admin to monitor and review user feedback.
- **View Account:** Provides access to user accounts, allowing the admin to manage their details.
- **Change Password:** Enables the admin to update or reset passwords for user security.
- **Active/Inactive Account:** Allows the admin to activate or deactivate user accounts based on their status.

- **Create Account:** Facilitates the registration of new users in the system.

2. Manager Subsystem

The **Manager** oversees pharmacy operations and employee management. The key functions include:

- **Register Employee:** Enables the manager to add new employees to the system.
- **View Employee:** Allows the manager to review employee details and monitor their activities.
- **Delete Employee:** Provides the ability to remove employees from the system if necessary.
- **View Report:** Allows the manager to access system-generated reports for decision-making.

3. Pharmacist Subsystem

The **Pharmacist** handles drug sales and patient interactions. This subsystem includes:

- **Sale Drug:** Manages the sale and dispensing of medications to patient.
- **View Patients:** Provides details about patients, ensuring personalized service.
- **Register Patient:** Enables the pharmacist to create new patient profiles in the system.
- **Check Available Drug:** Allows the pharmacist to verify stock before dispensing medication.

4. Cashier Subsystem

The **Cashier** is responsible for handling financial transactions. This subsystem includes:

- **Give Receipt:** Ensures accurate billing by generating receipts for patient.
- **View Saled:** Provides access to sales records for financial tracking and reporting.

5. Patient Subsystem

The **patient** interacts with the system to access drug-related information. The functionalities include:

- **Check Available Drug:** Allows patients to verify drug availability before making a purchase.
- **Read Information:** Provides detailed descriptions and usage guidelines for medicines.

6. Store Coordinator Subsystem

The **Store Coordinator** manages inventory and stock updates. The responsibilities include:

- **Check Available Drug:** Monitors drug stock levels to prevent shortages.
- **Check Expired Date:** Tracks drug expiration dates to ensure the removal of outdated stock.
- **Register Drugs:** Adds new drug stock to the system for inventory tracking.
- **Update Price:** Modifies drug pricing as needed to reflect market changes.

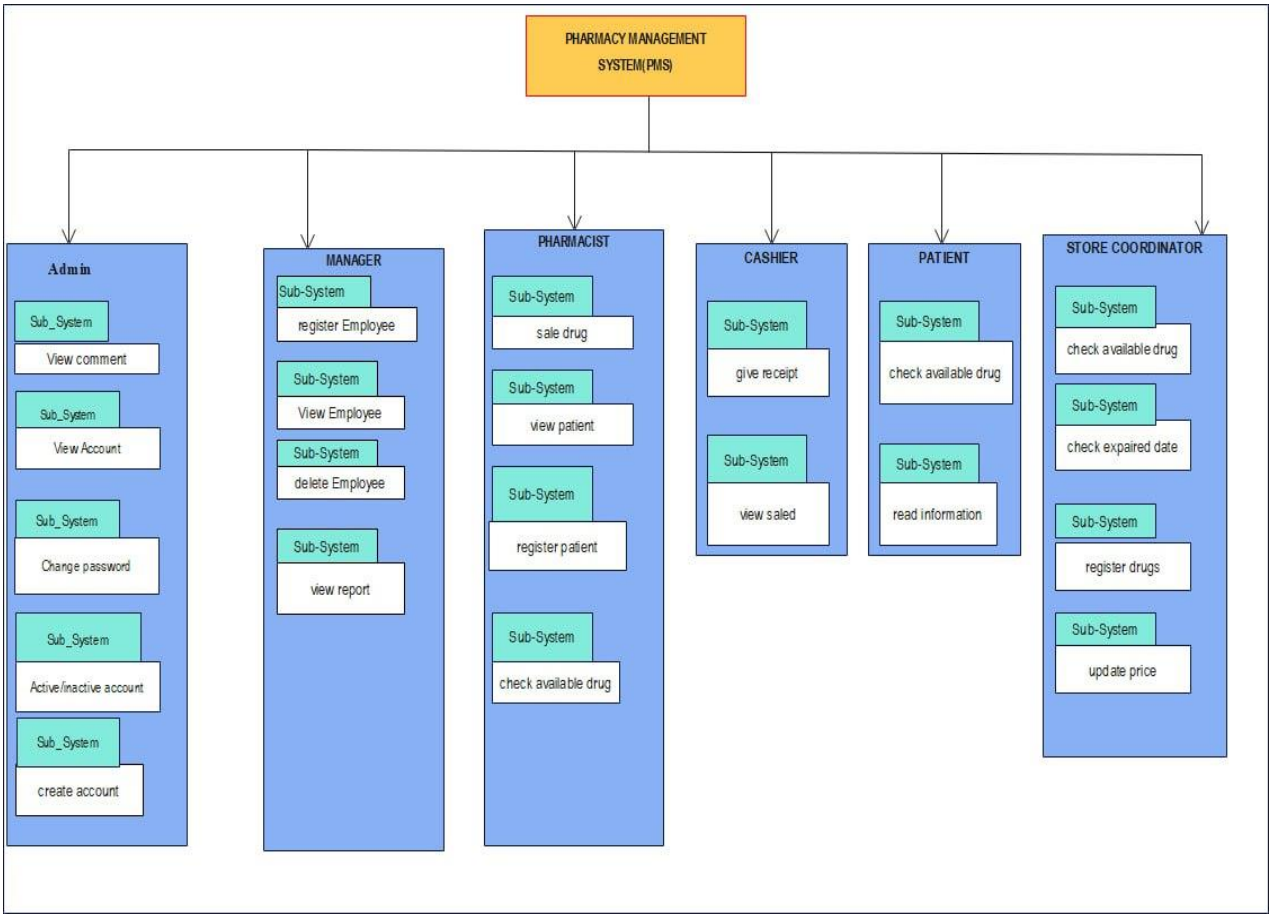


Figure 22 PMS Subsystem diagram

3.3.2 Component diagram

The component diagram for the Pharmacy Management System at Salale University Comprehensive Specialized Hospital provides a clear and engaging representation of its architecture. It includes components such as the **User Interface**, which ensures a user-friendly experience for actors like managers, admin, pharmacists, cashiers, store coordinators, and patient. The **Application Layer** processes core functionalities like drug management, sales, reporting, and notification handling. The **Database Component** efficiently stores and retrieves all relevant data, including drug inventory, sales records, and user information. This diagram illustrates the seamless interaction between components, ensuring a dependable and efficient

system for managing pharmacy operations. It highlights the system's capability to handle tasks reliably while maintaining an organized structure [10].

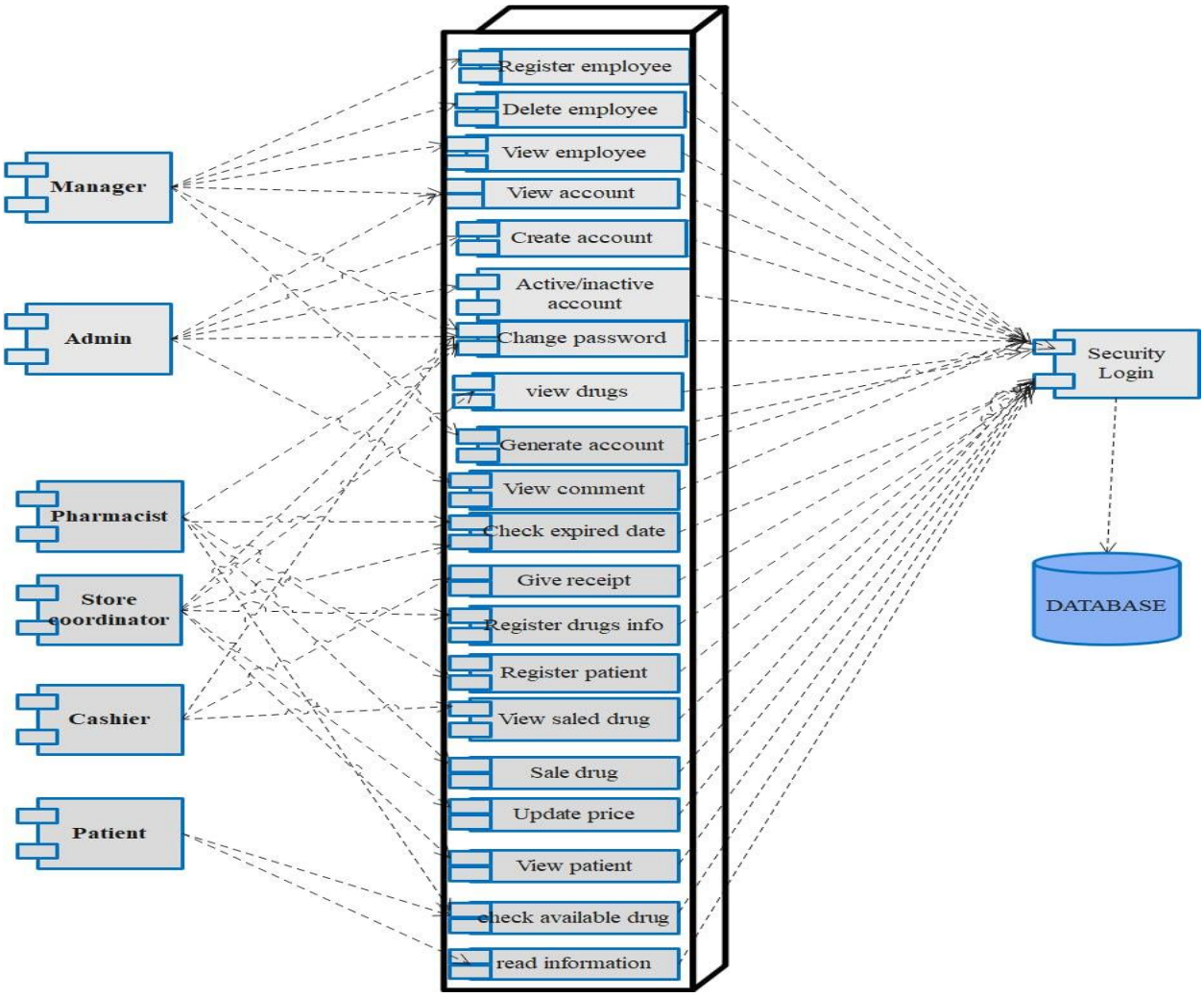


Figure 23 Component diagram

3.4 Deployment diagram

The deployment diagram for the Pharmacy Management System at Salale University Comprehensive Specialized Hospital illustrates the physical arrangement and interaction of its components, highlighting its distributed infrastructure. The diagram features the **Web Server**, which acts as the entry point for users, offering an intuitive and engaging interface for pharmacists, store coordinators, managers, cashiers, and patient. The **Application Server** handles critical operations like drug registration, sales, stock management, and reporting,

ensuring smooth and efficient functionality. The **Database Server** securely stores essential data, such as drug inventories, sales transactions, and user information. Various clients, including administrators and staff, interact with the system through web browsers or other interfaces, ensuring widespread accessibility. This visual representation underscores the system's scalability, reliability, and ability to deliver efficient performance, making it an essential tool for managing complex pharmacy operations effectively [12].

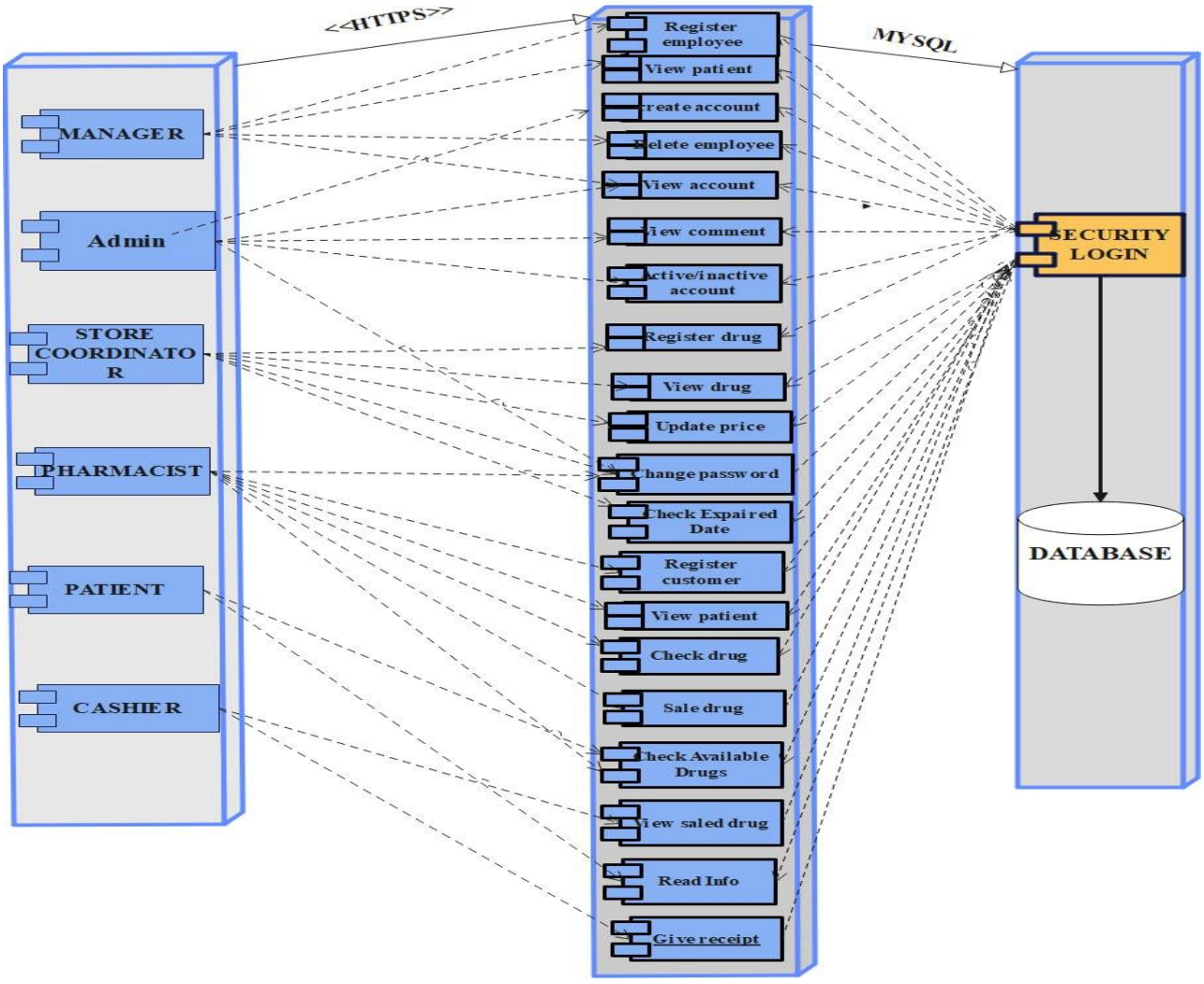


Figure 24 Deployment diagram

3.5. Database design (E-R diagram) for relational database

An Entity-Relationship Diagram (ERD) visually represents the entities within a system and the relationships between them. Entities are objects or things that have attributes, such as *Manager*

(with attributes like *ManagerID*, *Name*, etc.), *Admin* (with attributes like *AdminID* and *Role*), *Pharmacist* (with attributes like *PharmacistID*, name), *Cashier* (with attributes like *CashierID* and name), and *Patient* (with attributes like *PatientID*, *Name*, and *ContactInfo*). Relationships define how entities are linked, such as a *Manager* overseeing multiple *Pharmacists* (one-to-many relationship), a *Customer* purchasing from a *Cashier* (many-to-one relationship), or an *Admin* managing system settings. The diagram also includes primary keys to uniquely identify entities and foreign keys to establish connections between related entities. ERDs help in designing databases by visually mapping the structure and interrelations of data [4].

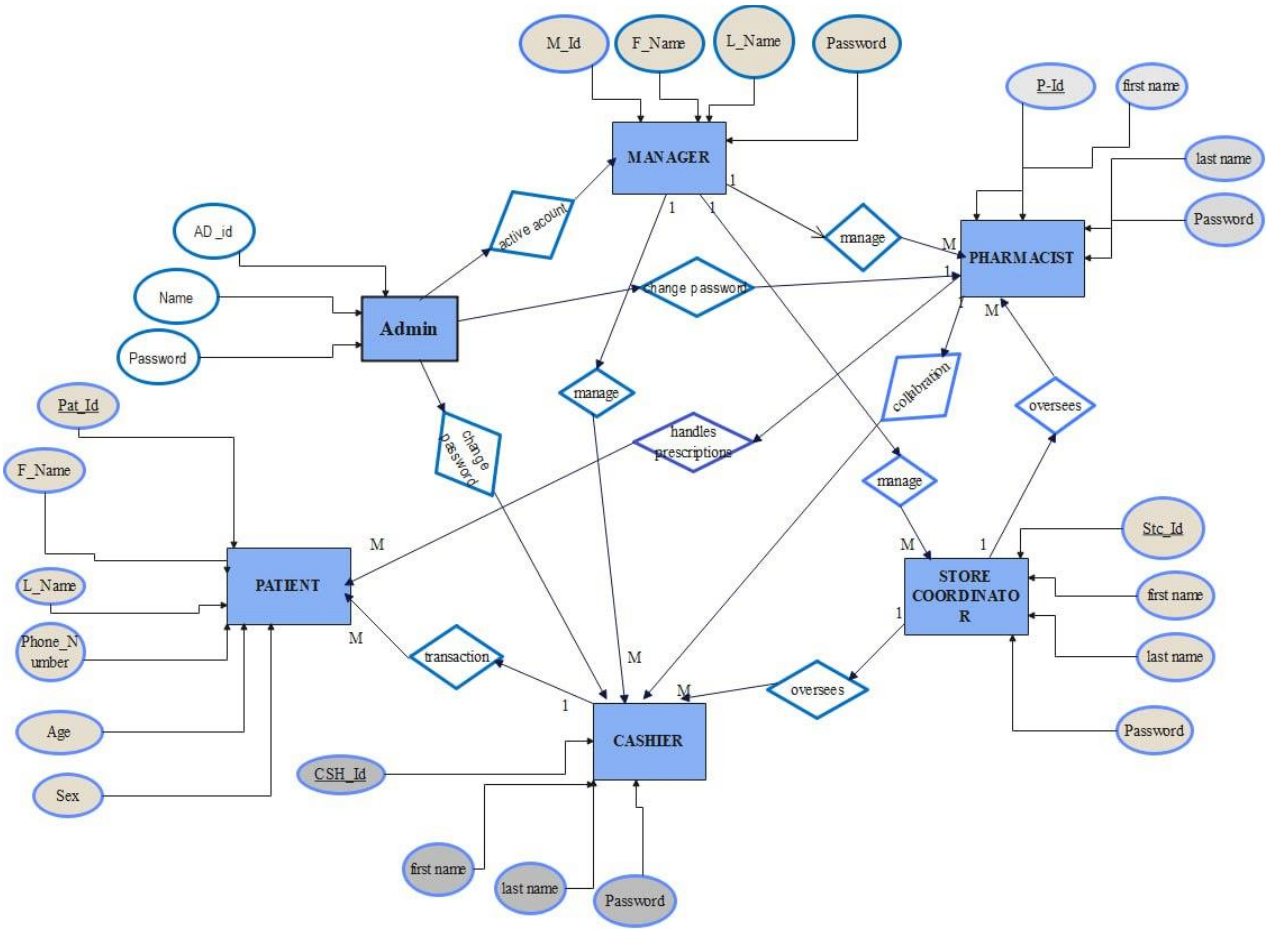


Figure 25 Er diagram

3.6. Access Control and Security

The Pharmacy Management System at Salale University Comprehensive Specialized Hospital places a high priority on access control and security from the outset. Users interact with an

intuitive and visually appealing login interface, where they enter their unique username and password. These credentials undergo a thorough authentication process to ensure security. Upon successful authentication, users gain access based on their role or group within the system. Managers have elevated privileges to oversee the system, access sensitive data, and manage accounts. Pharmacists can manage drug inventory, register patient, and process sales. Store coordinators handle stock updates, including registering new drugs and monitoring expiration dates, while cashiers manage sales and generate receipts. This robust and secure login mechanism ensures that only authorized users access the system's functionalities, supporting a smooth and efficient workflow for all stakeholders.

Here is a tailored table following the format you shared, but adjusted for your Pharmacy Management System (PMS) project. It includes functionality, access control, and security measures relevant to your project:

Table 15 Access Control and Security

Functionality	Access Control	Security Measures
Drug Management	Pharmacist, Store Coordinator	• Only authorized users can add, update, delete, and view drugs. • Ensure secure authentication for access to drug management.
Customer Registration	Pharmacist	• Only pharmacists can register patient. • Data encryption for sensitive patient information like contact details.
Sales Processing	Cashier, Pharmacist	• Only authorized users can handle sales and transactions. • Secure payment handling and session management to prevent unauthorized access.

Account Management	Admin	• Only Admin can create, update, and delete user accounts. • Password hashing and multi-factor authentication for account security.
Report Generation	Manager, Store Coordinator	• Only managers and store coordinators can generate sales and inventory reports. • Data encryption for report contents and restricted access based on roles.
Inventory Checking	Store Coordinator, Manager	• Only authorized users can check expired drugs and inventory levels. • Regular audits to ensure the accuracy of inventory data.
View Drug Information	Patient	• Open access for customers to view available drugs and their details. • HTTPS to secure customer browsing activities.
Notification Management	Manager, Store Coordinator	• Only authorized roles can send or view notifications regarding stock and updates. • Secure channels for sending notifications to avoid spoofing or interception.
System Updates	Manager, admin	Admin and managers can initiate system updates. • Secure update process and patch management to address vulnerabilities.

References

[1] "Software Advice," 2023. [Online]. Available: https://www.softwareadvice.com/. [Accessed 30 12 2024].

[2] "GeeksforGeeks," [Online]. Available: https://www.geeksforgeeks.org/software-engineering-prototyping-model/. [Accessed 11 11 2024].

[3] "W3School," [Online]. Available: https://www.w3schools.com/. [Accessed 12 11 2024].

[4] "Software Advice," 2023. [Online]. Available: https://www.softwareadvice.com/. [Accessed 25 12 2024].

[5] "Propoal Kit," [Online]. Available: www.proposalkit.com. [Accessed 11 11 2024].

[6] "Investopedia," [Online]. Available: https://www.investopedia.com/terms/f/feasibility-study.asp. [Accessed 09 11 2024].

[7] J. &. L. K. Smith, Pharmacy Management Systems: Technology and Applications., NewYork: HealthTech, 2018.

[8] "Pharmacy Tech Review," 2023. [Online]. Available: https://www.pharmacytechreview.com/. [Accessed 28 12 2024].

[9] "Oxford academic," [Online]. Available: www.ajhp.org/automation-expiration-dates. [Accessed 10 10 2024].

[10] "Healthcare IT News," 2022. [Online]. Available: https://www.healthcareitnews.com/. [Accessed 10 01 2025].

[11] "International Pharmaceutical Federation (FIP)," 2020. [Online]. Available: https://www.fip.org/. [Accessed 10 11 2024].

[12] T. &. R. L. Harrison, he The Challenges of Manual Data Entry in Healthcare Systems.

Health Information Management Journal, 2019.

[13] "pharmacytimes," [Online]. Available: https://www.pharmacytimes.com. [Accessed 10 10 2024].

[14] "Pharmacy Times," 2023. [Online]. Available: https://www.pharmacytimes.com/. [Accessed 26 12 2024].

Appendix

Interview Questions

1. Can you describe the primary operations and daily tasks performed within the pharmacy?
2. What are the main challenges you face in managing pharmacy operations, such as inventory, employee scheduling, and customer service?
3. How do you currently manage the flow of prescriptions and medical supplies in your pharmacy?
4. How do you manage customer prescriptions, deliveries, or refill?
5. How much employee is working in this pharmacy?
6. Do you face any challenges when it comes to inventory audits or stock discrepancies? How are these handled?
7. What tools do you use to keep track of employee performance, shifts, and responsibilities?