Atal Bihari Vajpayee
Indian Institute of Information Technology
and Management (ABV-IIITM), Gwalior
(An Institute of National Importance, Ministry of Education, Government of India)

SOFTWARE ENGINEERING

Design Report

Pharmacy Management System

Submitted to

Dr.Santosh Singh Rathore

Group members:

MARAM VIGNESH(2021 BCS-039)

NARRA ABHIGNA(2021 BCS-048)

PATHLAVATH SUDEENDRA(2021 BCS-052)

RAVI JWALANA(2021 BCS-056)

Table of Contents

1.Introduction

A pharmacy management system is a type of management that is intended to boost efficiency, safety, and accuracy in a pharmacy. It is a computer-based system that aids pharmacists in managing inventories, costs, and other factors including medical safety. The system will also provide a report with a list of products that will expire after a certain date before the product itself does. For a set period, such as every month, it also requires manual entry when new batches of medications arrive and when they are moved out of the pharmacy. This involves gathering information about the medications, such as their expiration date, date of purchase, how many of each type are still in stock, and where they are located within the pharmacy. In the pharmacy, a manual system is now in use. Each drug that is on hand in the pharmacy needs to be manually checked by the pharmacist. This typically results in errors as the pharmacist's workload rises.

The pharmacy has a very big customer base as a result of its size and high level of customer care. These clients typically visit the drugstore for medicines is growing at the moment, which adds to the pharmacists already hearvices after they leave work. The number of consumers visiting increases the burden.

It is challenging for the pharmacist in this situation to serve consumers quickly. In the meantime, the pharmacist must guarantee consumer pleasure with their services to maintain them. The problems cause delays in providing services to customers, which slows down sales and increases the chance of losing important clients in the long run.

Writing the order requires a significant amount of time since the pharmacist must review the stock balance and make an educated guess as to how much to order based on the figures .

Medicines should not be used after their expiration dates.

This project work will notify the pharmacist of medications that are ready to expire, prohibiting the sale of those medications and offering a remedy to the above mentioned issues.

2.Design considerations

The section highlights the issues and difficulties that need to be addressed or resolved before moving onto a comprehensive design solution. The foundation of this document is the SRS document version 1.0. In the event of any section not being understood or feeling incomplete, there is a need for reference.

2.1 Assumptions

As stated in the SRS document, the Pharmacy management system makes a number of assumptions regarding the necessary hardware and software.

2.2 Design constraints

The system shall be developed using the following technologies:

1. HTML5

2. CSS3

3. JavaScript

4. MongoDB

5. React

6. Node.js

7. Express.js

2.3 Design methodology

The system will be designed using Agile software development methodology, specifically the scrum and eXtreme Programming methodologies, which promotes adaptive planning, evolutionary development, and delivery, a time-boxed iterative approach, and encourages rapid and flexible response to changes. This decision is based on the small team and project size.

The design will take the following approach:

1. Identifying Modules

2. Module Specification

3. Designing module interfaces and creating relationships

4. Designing user interfaces

2.4 System environment

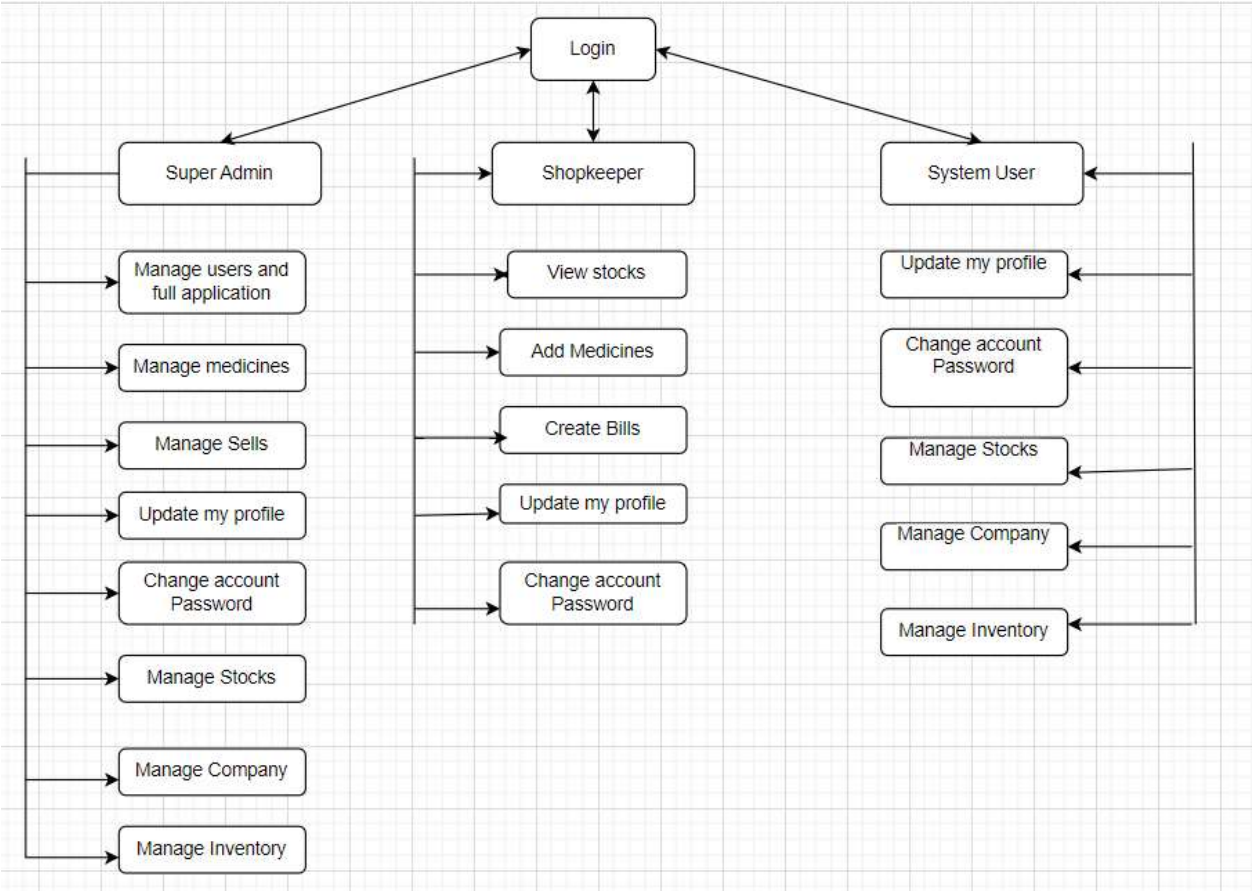
Scalability and Flexibility: The system environment is created to be scalable and adaptable, allowing for future development, adjustments in requirements, and advancements in technology. This includes taking into account aspects like system performance, extensibility, and capacity to meet changing requirements and changes.

Security: The system environment's security features are essential for preserving the privacy, accuracy, and accessibility of sensitive data in the pharmacy management system. To protect against unauthorized access, data breaches, and other security threats, this includes putting in place safeguards like firewalls, intrusion detection systems, data encryption, access limits, and regular security audits.

3. Architecture

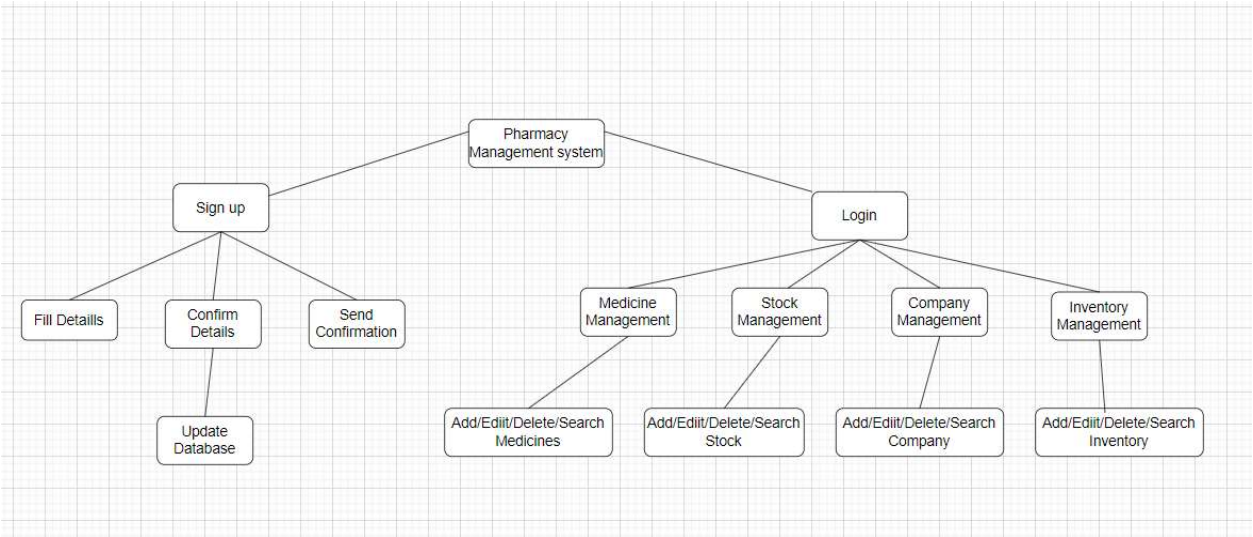
3.1 System Design

Systems design means a methodical approach to design the system. Regardless of whether it uses a bottom-up or top-down method, the process is systematic and takes into account all relevant system-related aspects, including the architecture, necessary hardware, and software, right down to the data and how it moves through the system. Systems analysis, systems engineering, and systems architecture then overlap with systems design. During this stage, the intricate task of system development is broken down into a number of more manageable sub-activities that work in concert to accomplish the overall goal of system development.



3.2 Functional Decomposition tree

The system's primary functions are divided up into smaller subfunctions, submodules, and so on. After implementation, the system will operate according to the organizational structure. Since the decomposition is stable, cohesiveness should be increased in the functions.



1. Sign up

   This module is required for the registration of new users or us vendors into accessing the system. This module has three sub-modules

   1.1 Fill details: This sub-module is required to input the user entered details from the interface and store it in a data structure.

   1.2 Confirm details: This sub-module is required to verify the details stored in previous data structure

      1.2.1 Update database: Update new user into database

   1.3 send confirmation: Intimate user about the registration credentials and confirmation via email

2. login

2.1 medicine management

2.1.1 add medicine

2.1.2 edit medicine

2.1.3 delete medicine

2.1.4 search medicine

2.2 stock management

2.2.1 add stock

2.2.2 edit stock

2.2.3 delete stock

2.2.4 search stock

2.3 company management

2.3.1 add company

2.3.2 edit company

2.3.3 delete company

2.3.4 search company

2.4 inventory management

2.4.1 add inventory

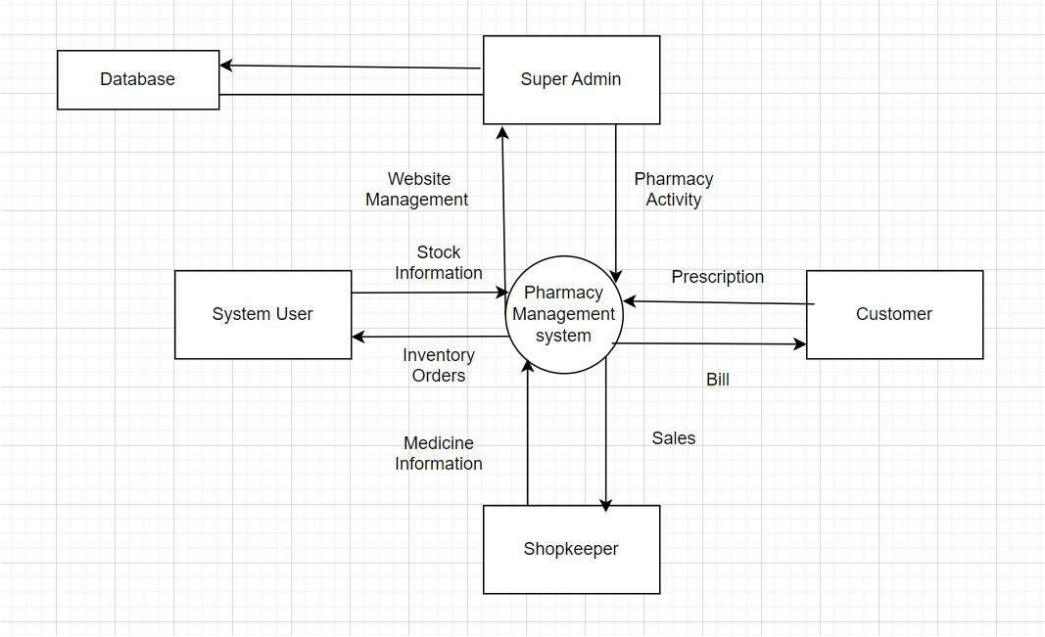
2.4.2 edit inventory

2.4.3 delete inventory

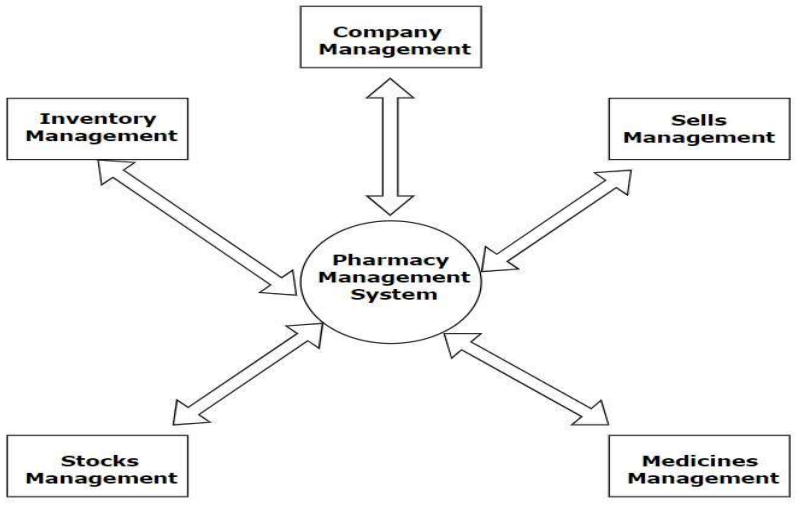
2.4.4 search inventory

3.3 Context Diagram

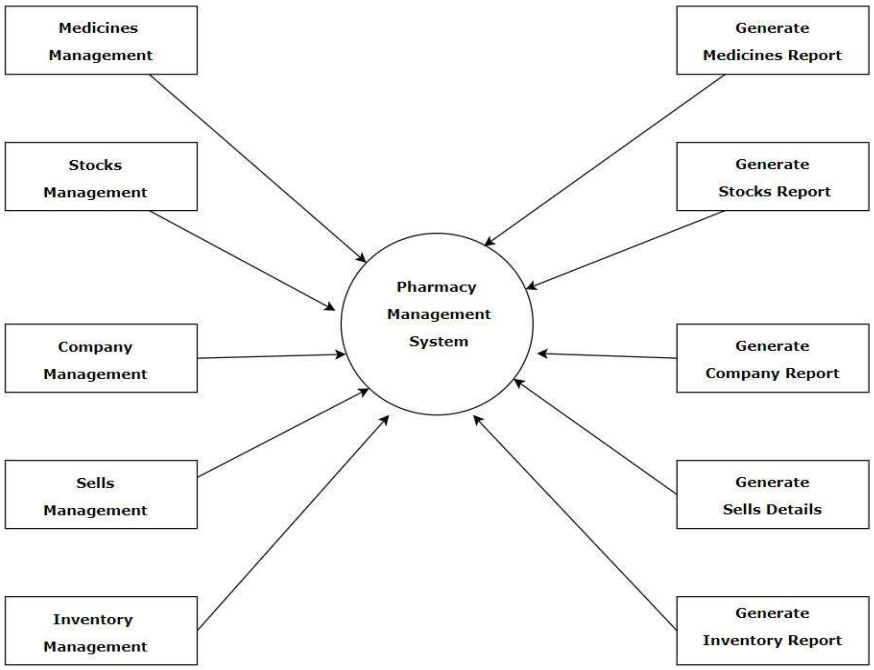
Context diagram describes the main actors interacting with the system.



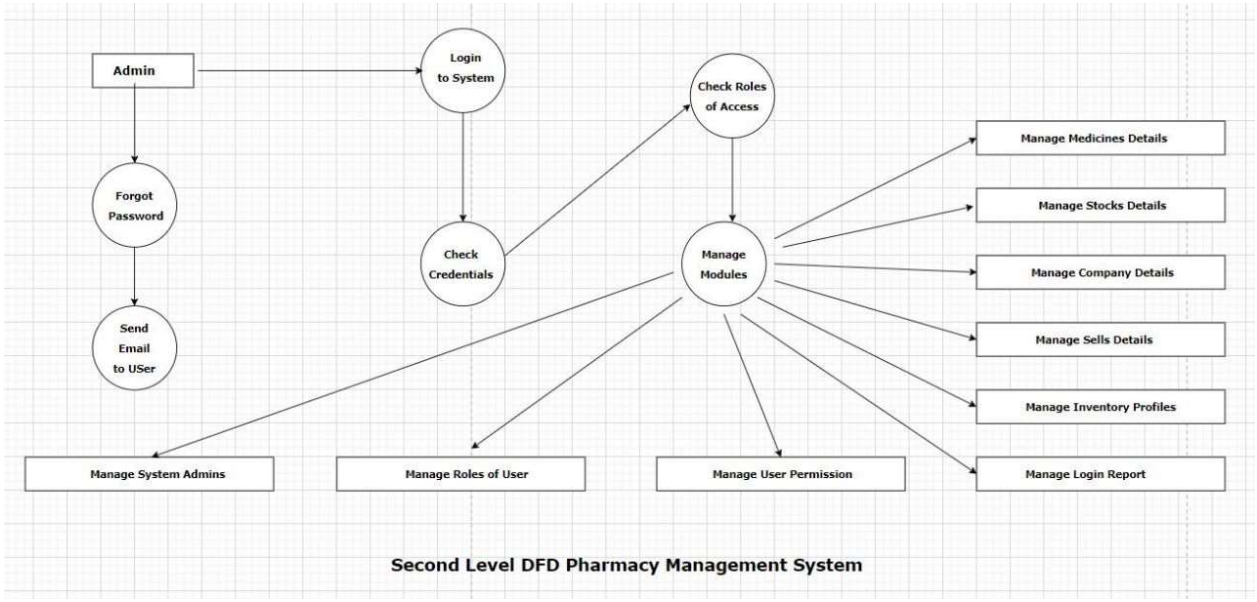
3.4 Data flow Diagram



Zero Level DFD Pharmacy Management System



First Level DFD Pharmacy Management System



Second Level DFD Pharmacy Management System

3.5 Data Dictionary

User

Field	Type	Null	Default
user_id	int	No	None
user_role_id	int	No	None
user_name	string	No	None
user_email	string	No	None
user_dob	date	No	None
user_address	string	No	None

Company

Field	Type	Null	Default
company_id	int	No	None
company_description	string	No	None
company_type	string	No	None
company_name	string	No	None

Medicine

Field	Type	Null	Default
medicene_id	int	No	None
medicene_name	string	No	None

medicene_type	string	No	None
medicene_description	string	No	None
medicene_company	string	No	None
medicene_cost	string	No	None
medicene_dose	string	No	None

Inventory

Field	Type	Null	Default
inventory_id	int	No	None
inventory_number	string	No	None
inventory_description	string	No	None
inventory_tyoe	string	No	None
inventory_items	string	No	None

Sales

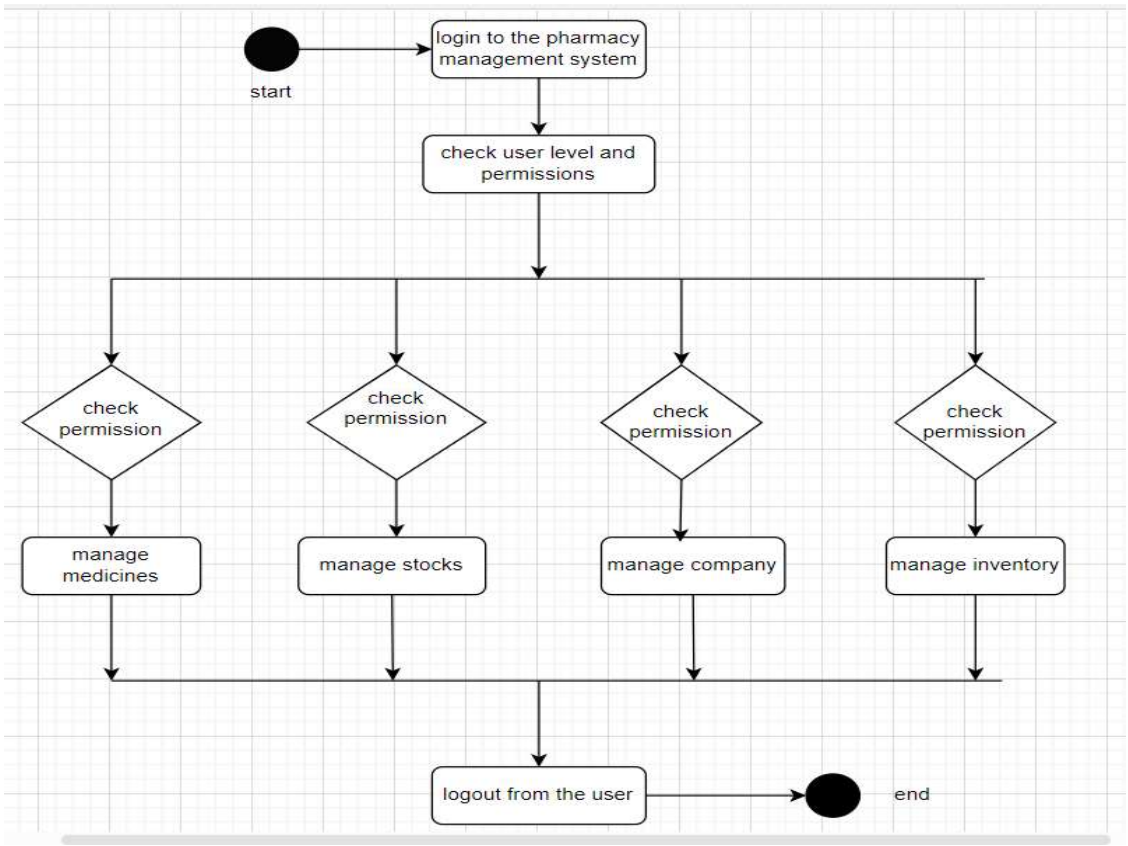
Field	Type	Null	Default
sales_id	int	No	None
sales_name	string	No	None
sales_type	string	No	None
sales_description	string	No	None

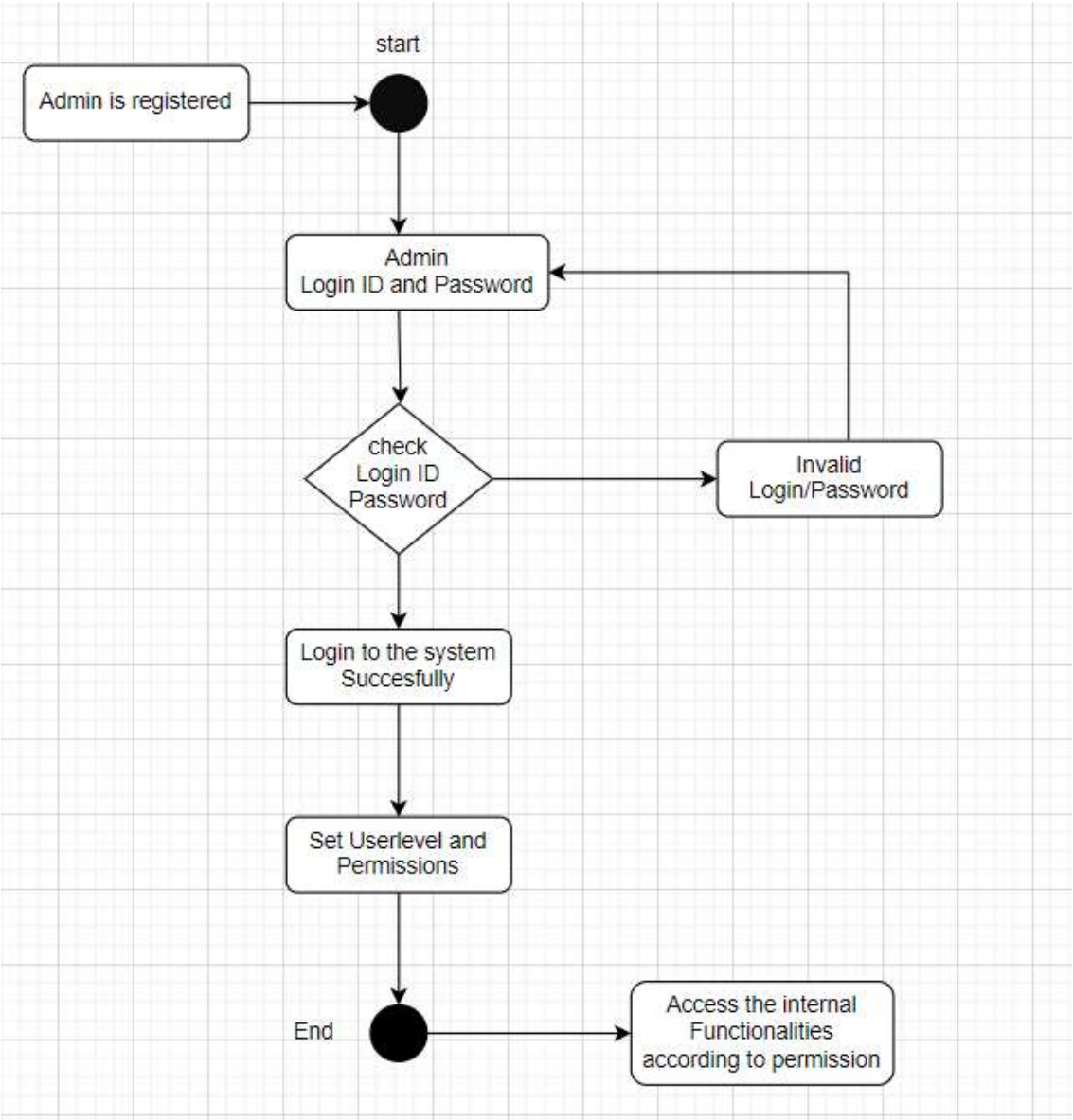
Stock

Field	Type	Null	Default
stock_id	int	No	None
stock_item	string	No	None
stock_type	string	No	None
Stock_number	string	No	None
stock_description	string	No	None

4. Component Design

4.1 Activity Diagram





5. User Interface Design

UI design principles are followed when creating UI.

The organizing principle is that related things are grouped together and irrelevant things are kept apart in user interfaces (UI).

The simplicity principle states that the offered interface is simple to use. If a mistake is made, the system will display an error message.

The visibility principle states that the Interface provides access to all system features. Users are not overrun by the number of options.

The feedback principle states that users are kept aware of activities, mistakes, or exceptions by a system of messages in the design.

The reuse principle: To avoid ambiguity, the same names were used in design to carry out the same operations on several objects.