

Project Report

(Review III report)

On

Pharmacy Management System

Submitted by

Shah Aryan Rakesh – 20BCE2662

Patel Viraj Shaileshkumar – 20BCE2503

Maria Rennie – 20BCE2273

Kirit Govindaraja Pillai – 20BCE2404

Vadarevu Sai Samprreet Khushal – 20BCE2537

For

Software Engineering

CSE3001

Slot: B1, J Component

B.Tech. in Computer Science and Engineering



November 2022

Table of Contents

Chapter 1- Introduction to the Project	4
1.1 Introduction	4
1.2 Problem Definition	4
1.3 Project Scope	5
1.4 Motivation	5
1.5 Background Study/Literature Survey	5
1.6 SDLC Approach	6
Chapter 2 - Project Planning	7
2.1 Project Schedule	7
2.2 Effort and Resource Estimation	8
Chapter 3 - Requirement Gathering and Analysis	10
3.1 SRS	15
3.2 Data Modelling	15
3.3 Structural Analysis	16
<i>Data Flow Diagrams</i>	16
Chapter 4 - Designs	18
4.1 UML Designs	18
<i>Use-Case Diagram:</i>	18
<i>Class Diagram:</i>	19
<i>Component Diagram</i>	20
<i>Sequence Diagram</i>	21
<i>Collaboration Diagram</i>	23
<i>Collaboration Diagram</i>	23
Chapter 5 - Development	24
5.1 Tools Description and Development approach used	24
5.2 Pseudocodes of Important Modules	25
Chapter 6 - Testing	38
6.1 Test cases for Important Modules	38
Chapter 7 - Screenshots of Developed Product	45

Chapter 8 - Conclusion and References	51
8.1 Conclusion	51
8.2 References	51
Annexures	52
1. Review I Presentation	52
2. Review II Presentation	52
3. Review III Presentation	53

Chapter 1- Introduction to the Project

1.1 Introduction

Increasing health concerns among people demands efficiency in medical stores. The need to migrate from originally paper-based to electronic notebooks with data storage, computational features and reliable electronic documentation has arrived. For this project, we have designed a Medical Store Management System based on MySQL Database for small pharmacies. The software has been designed with a basic user interface to allow stock maintenance, database access and electronic documentation for billing purposes with valued customer support. These features have been later used to compute discounts for customers and daily revenues. Being an open-source software, it can be used by a wide variety of retail and wholesale outlets to automate the process of manually maintaining the records and cash flows. The results have been potentially generalizable. This software saves your time and money.

1.2 Problem Definition

- a. Most pharmacies rely on paper-based receipts. This can easily lead to human errors, such as incorrect medicines given to a patient due to illegible receipts received.
- b. Management of the large quantities of supplies happens through physical records- data about inventory, expiry dates, and drug details and alternatives can easily be erased due to human errors, or natural disasters. Both of the above points further increase the use of precious natural resources such as paper.

1.3 Project Scope

The aim of this project is to make the management of medicines sold within a pharmacy easy for all parties involved- for admins, cashiers, other employees, and for patients as well. We attempt to eliminate the problems in existing systems by deploying a robust, online system that makes it easier to retrieve and store data related to a pharmacy.

1.4 Motivation

In this era of growing diseases there has been a surge in medical companies and medicines in general. Due to this it is very difficult for the pharmacy units to keep a track of medicines and in turn customers and suppliers. There is a huge investment of time, money and effort. Also, there is a huge waste of resources like paper. So we are required to build software for the sake of ensuring effective data saving and manipulation. This software is named Pharmacy Management System. The system should deal with the maintenance of

drugs and consumables in the pharmacy unit. Pharmacy management system deals with the maintenance of drugs and consumables in the pharmacy unit. It also provides a way for the patients to communicate with the pharmacy and request medicines. This pharmacy management system is user friendly. The aim of the project is to create an effective software to help the pharmacist to maintain the records of the medicines, handle user details, generate invoice, check and renew validity and a request-confirm system for medicines requested by patients. We have built a robust and integrated Pharmacy Management System. It deals with the maintenance of drugs and consumables in the pharmacy unit. The set-up of this pharmacy management system will ensure availability of sufficient quantities of drugs and consumable materials for the patient. The Pharmacy Management System is built for the sake of ensuring effective data saving and manipulation. This refers to the fact that our software highly minimizes the time and resources of the Pharmacy unit.

1.5 Background Study/Literature Survey

a. Survey of the Existing Models/Work:

The systems which have been surveyed are mainly one-sided in nature. The pharmacy management systems mainly for employees alone, or only for cashiers when they ring up the final bill. The medicine purchase systems do not have a prescription uploading functionality.

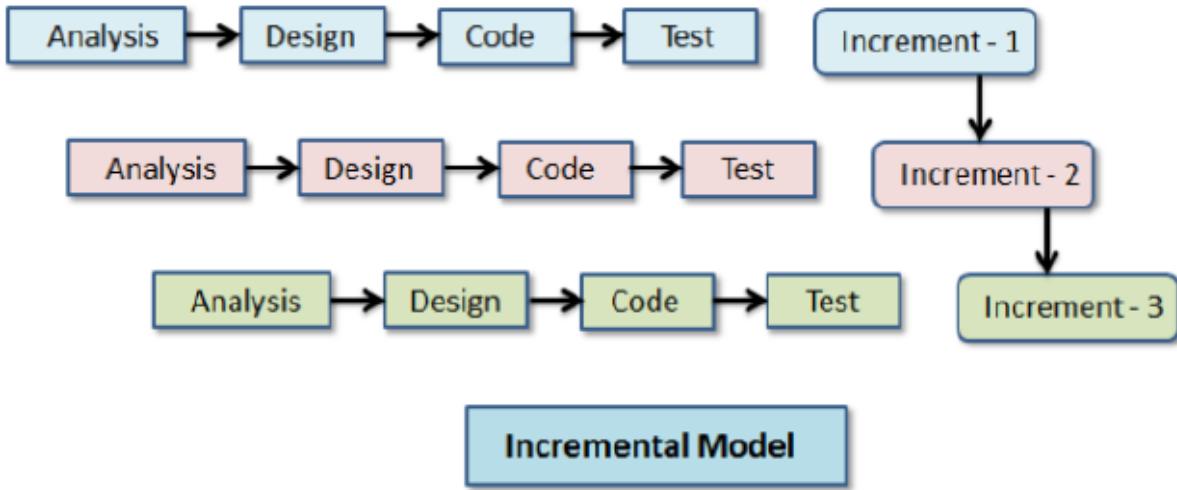
b. Summary/Gaps identified in the Survey:

Current pharmacy systems are largely paper based. This poses a difficulty for employees to check if a particular medicine is available or not. Majority of the current pharmacy systems are paper based. So, they might not have a proper record of the invoices generated. The method of finding the medicines is extremely tedious as it is manual.

1.6 SDLC Approach

The incremental model of system development life cycle, which allows for scalability as time goes on, is the approach utilized in the software's implementation. All parties involved in pharmaceutical practices would benefit from the creation of an online pharmaceutical management system.

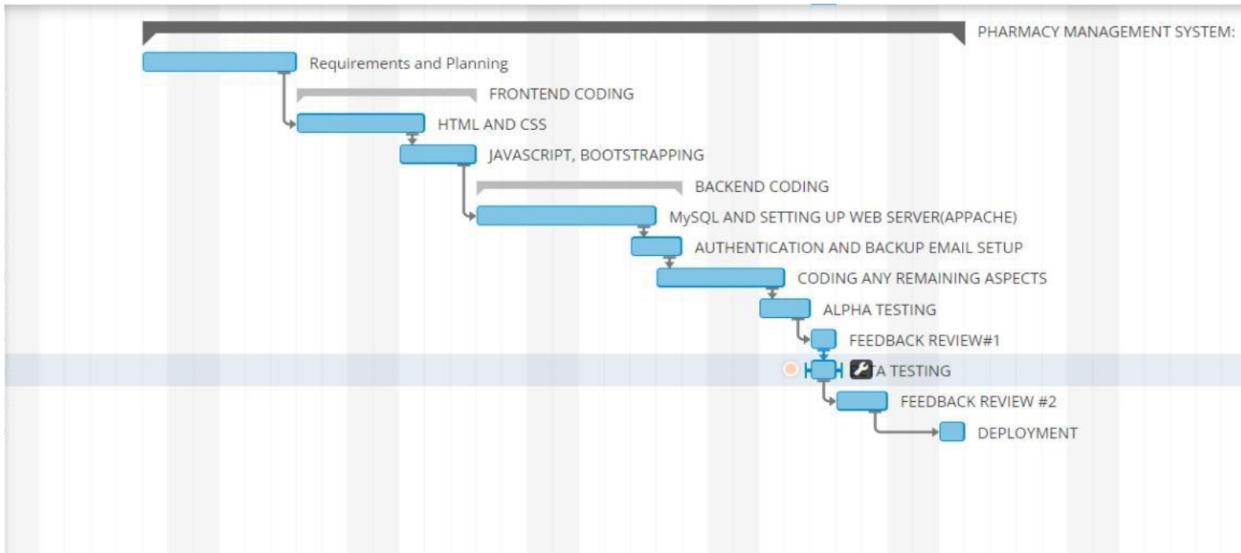
Software development using the incremental model divides requirements into several independent software development cycle modules. Each module in this paradigm undergoes the steps of requirements, design, implementation, and testing. The module's functionality is increased with each new release. The process continues until the complete system is achieved.



<i>Progressive Phases</i>	<i>Activities carried out in successive stages</i>
Requirement Analysis	Software requirements and specifications are gathered.
Design	During this stage, some advanced features are designed.
Code	Coding of software is performed at this phase.
Test	Once the system has been implemented, the testing phase commences.

Chapter 2 - Project Planning

2.1 Project Schedule



Gantt charts are grid-based diagrams that break down projects across calendar days. They are typically used as a project management tool for large-scale development projects with multiple stages and a firm deadline.

A Gantt chart is a project management tool that illustrates a project plan. It typically includes two sections: the left side outlines a list of tasks, while the right side has a timeline with schedule bars that visualize work. The Gantt chart can also include the start and end dates of tasks, milestones, dependencies between tasks, and assignees.

2.2 Effort and Resource Estimation

The software development effort estimation is an essential activity before any software project initiation. For effort estimation we will be required to break our task into various sub tasks. Then find out the number of staff required for the individual task and then estimate the resources, cost per task.

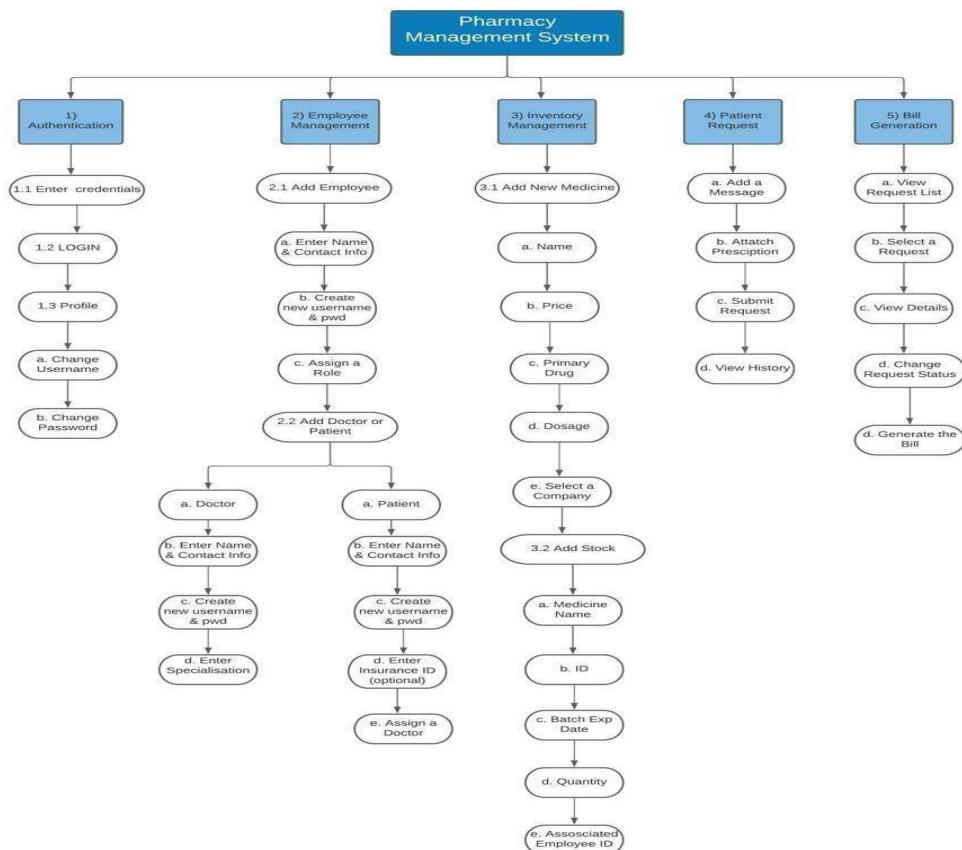
We will require 4 teams - Frontend , Backend , Design and Testing & Deployment Teams. Each team will have a minimum of 3 members. Frontend team will develop a responsive frontend and the backend team will develop API's and will link it to the frontend page. The Design team will coordinate with both Frontend & Backend to create interactive interfaces. Testing team will test all the modules and will verify the interfaces. Corrections to be implemented are provided to the - Frontend & Backend teams and after all testing is successful the project is deployed. Each team will require computer systems, good internet connection and various tools to develop the software.

Let's assume

- There are 4 members in each team.
- Daily wages per person per day is Rs.500
- So, per team cost is = Rs.2000 (4 * 500)
- Total cost = Rs.8000 (4 * 2000)
- Per day expense of company is around = Rs.2000
- Total cost per day = Rs.10000

Work - Breakdown Structure:

A Work Breakdown Structure includes dividing a large and complex project into simpler, manageable and independent tasks. The root of this tree (structure) is labelled by the Project name itself. For constructing a work breakdown structure, each node is recursively decomposed into smaller sub-activities, until at the leaf level, the activities become undividable and independent. It follows a Top-Down approach. The WBS is split into four parts: Authentication, Employee Management, Inventory Management, Patient Request and Bill generation. Within the Authentication phase of the WBS, we have performed a literature survey and identified the specifications necessary to successfully authenticate users while ensuring the security of the software. These include the Credentials of the user, the Login Page design, the Profile of the users stored in a database to retrieve the information and the ability to change the username and password for the users in case of human errors.



Within the Employee Management Phase, we were able to identify the necessities of the administrator of the organization and hence, arrived at the steps necessary to ensure the successful registration of an employee, doctor and patient. This includes the setup of an account with the username and the password along with the assignment of a role within the organization. Depending on the role assigned, specific details are collected to ensure accurate details are received to ensure a smooth transition into the workplace.

Within the Inventory Phase, we have identified the necessities of the pharmacist to be able to keep track of the medicines and their respective details. These include the Name of the Medicine, the Price of the medicine and its dosage along with the manufacturer of the medicine, quantity and its respective expiration date which are all entered into a database.

Within the Patient Request Phase, we have identified the difficulties of the Patients using the system and designed the software to be able to ease the difficulty of the patient. This is ensured via a simple page which is easy-to-use with the ability to add a message to send the request, Ability to attach the prescription provided by the doctor for confirmation and to receive the further details necessary, and also provide the ability to view the patients' history of transactions.

Finally, Within the Bill Generation Phase, we have developed an efficient and hassle-free way of bill generation which would ease the payment process for the patients and the pharmacists. This allows the viewing and selection of requests from the request list, while being able to change the request status and allow the bill generation. This smoothens the process of payment as the process of bill generation is streamlined, unlike within the manual records approach.

Chapter 3 - Requirement Gathering and Analysis

3.1 SRS

Software Requirements Specification

for

Pharmacy Management System

Prepared by
Shah Aryan Rakesh [20BCE2662]
Patel Viraj Shaileshkumar [20BCE2503]
Maria Rennie [20BCE2273]
Kirit Govindaraja Pillai [20BCE2404]
Vadarevu Sai Samprreet Khushal [20BCE2537]

Date Created:18/8/2022

Table of Contents

Table of Contents.....	ii
1. Introduction.....	1
1.1 Purpose.....	1
1.2 Product Scope.....	1
1.3 References.....	1
2. Overall Description.....	1
2.1 Product Perspective.....	1
2.2 Product Functions.....	2
2.3 Literature Survey.....	2
2.4 Operating Environment.....	2
2.5 Assumptions and Dependencies.....	2
3. System Features.....	3
4. Other Nonfunctional Requirements.....	3
4.1 Performance Requirements.....	3
4.2 Security Requirements.....	3
4.3 Software Quality Attributes.....	3
5. Other Requirements.....	4

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed pharmaceutical management system. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate, and how the system will react to the constraints added by the user. This document is intended for both the stakeholders and the developers of the system.

1.2 Product Scope

The scope of the project is to design and develop a versatile, interactive, and easily maintainable system to store, retrieve and alter data for a pharmaceutical chain or unit. The goal of this software is to be implemented in pharmaceutical chains or retail shops to keep track of medicines in realtime and facilitate automated electronic documentation and stock maintenance

1.3 References

[1] Durward K. Sobek, Cindy Jimmerson, Applying the Toyota Production System to a Hospital Pharmacy

[2] Hewa Majeed Zangana, Design an Information Management System for a Pharmacy, IJARCCE-2018

2. Overall Description

2.1 Product Perspective

Increasing health concerns among people demand efficiency in Medical Stores. The need to migrate from paper-based to electronic notebooks with data storage, computational features, and reliable electronic documentation has aroused. For this project, we have designed a Medical Store Management System based on the MySQL Database for small pharmacies. The software with a user-friendly interface has been designed to allow stock maintenance, database access, and electronic documentation for billing purposes with valued customer support. These features have later been used to compute discounts for customers and daily revenues. Being open-source software, it can be used by a wide variety of retail and wholesale outlets to automate the process of maintaining the records and cash flows. The results have been generalizable. This software saves your time and money.

2.2 Product Functions

- The pharmacist should have access to the database to store and retrieve customer and medication data and keep track of inventory.
- The Manager must have access to the database, view and modify user details and medicines and can approve requests from the user's side.
- The user can add requests for medications, view the invoice and can maintain their records in one place using a login system.

2.3 Literature Survey

Survey of the Existing Models/Work

The systems which have been surveyed are mainly one-sided in nature. The pharmacy management systems are mainly for employees alone. The medicine purchase systems do not have a prescription uploading functionality.

Summary/Gaps identified in the Survey

Current pharmacy systems are largely paper-based. This poses a difficulty for employees to check if a particular medicine is available or not. The majority of the current pharmacy systems are paper-based. So, they might not have a proper record of the invoices generated. The method of finding the medicines is extremely tedious as it is manual.

2.4 Operating Environment

The software will operate on any operating system with the latest version of a chromium-based browser.

Recommended specs

- Processor: Dual core or above
- Ram: 2GB or greater
- Network Interface Card (NIC)

2.5 Assumptions and Dependencies

- It is assumed that all the available resources are available.
- The user of the pharmacy management system is equipped with good quality internet and knows basic computer operation
- The team developing the software has proper skills as well as the employees.
- Developed the software with limited resources like hardware and computational power.
- Business constraints like single thread implementation and Scaling issues.

3. System Features

- Login authentication system for users.
- Ability to modify personal details.
- Patients can request medicines by uploading the images of the prescription.
- The ability of employees to review requests.
- The ability of admin and managers to add new medicines to the inventory.
- Generate bills.
- Admin can manage employees.
- The ability of employees to view the patient's invoice history.
- The ability of patients to view their past orders.

4. Other Nonfunctional Requirements

4.1 Performance Requirements

- The user must have a device that can run the latest version of any chrome-based browser.
- The user must have at least have a dual core processor system to ensure smooth running of the program.

4.2 Security Requirements

The Pharmacy Management System requires database storage just like many other applications. Moreover, all the private user data uploaded by the user including their details and prescriptions is confidential to the other users

4.3 Software Quality Attributes

- **Usability:** Anyone familiar with using a browser can operate the system since it has a user-friendly user interface. Also include an instruction guide on how to use the system.
- **Reliability:** The pharmacy system is available based on the user needs, can work properly, and do transactions efficiently including safe data management of the pharmacy. The pharmacy system is password protected to change things on the system. Here the admin controls the system by login into the pharmacy system. Any user can't use the system. As a result, data is protected and controlled by only the administrator.
- **Supportability:** The code and supporting modules of the system should be well documented and easy to understand.
- **Maintainability:** A commercial database is used for maintaining the database and the application server takes care of the site. In case of failure, a re-initialization of the project will be done. Also, the software design is being done with modularity in mind so that maintainability can be done efficiently.

5. Other Requirements

User Interface Requirement

- The proposed system is mainly made using MySQL. Node.js is used in order to interact with the database and set the routes. Express.js is used in order to present the UI.
- The design follows the golden rules stated by Theo Mandel and reduces the user's memory load by keeping all the menu bar options visible at all times.
- The design mainly uses a menu selection interaction style along with certain places which use direct manipulation.
- The functionalities which the user has are visible on the left in a navigation bar. Thus, menu selection is used in the system.
- To modify data, there are text fields which use direct manipulation. Thus, direct manipulation interaction style is also used.

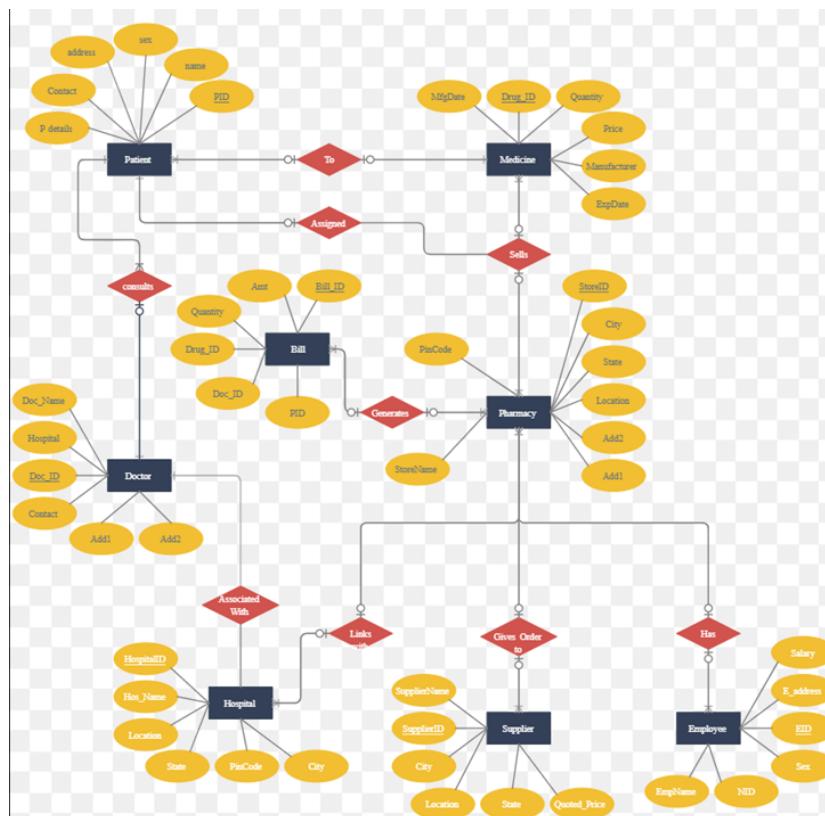
3.2 Data Modelling

Data modeling is the process of creating a simplified diagram of a software system and the data elements it contains, using text and symbols to represent the data and how it flows. Data models provide a blueprint for designing a new database or reengineering a legacy application. Overall, data modeling helps an organization use its data effectively to meet business needs for information.

- *ER Diagram:*

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

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



3.3 Structural Analysis

Structured Analysis and Structured Design (SA/SD) is a diagrammatic notation that is designed to help people understand the system. The basic goal of SA/SD is to improve quality and reduce the risk of system failure. It establishes concrete management specifications and documentation. It focuses on the solidity, pliability, and maintainability of the system.

Basically, the approach of SA/SD is based on the Data Flow Diagram. It is easy to understand SA/SD but it focuses on well-defined system boundary whereas the JSD approach is too complex and does not have any graphical representation.

Data Flow Diagrams

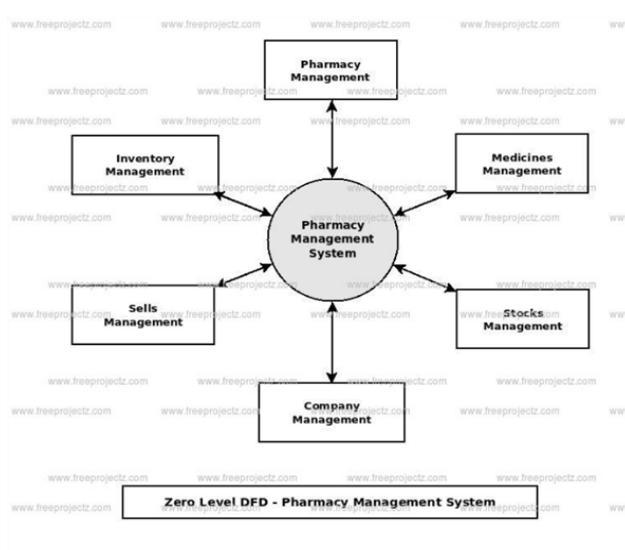
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both.

It shows how data enters and leaves the system, what changes the information, and where data is stored.

The objective of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system. The DFD is also called as a data flow graph or bubble chart.

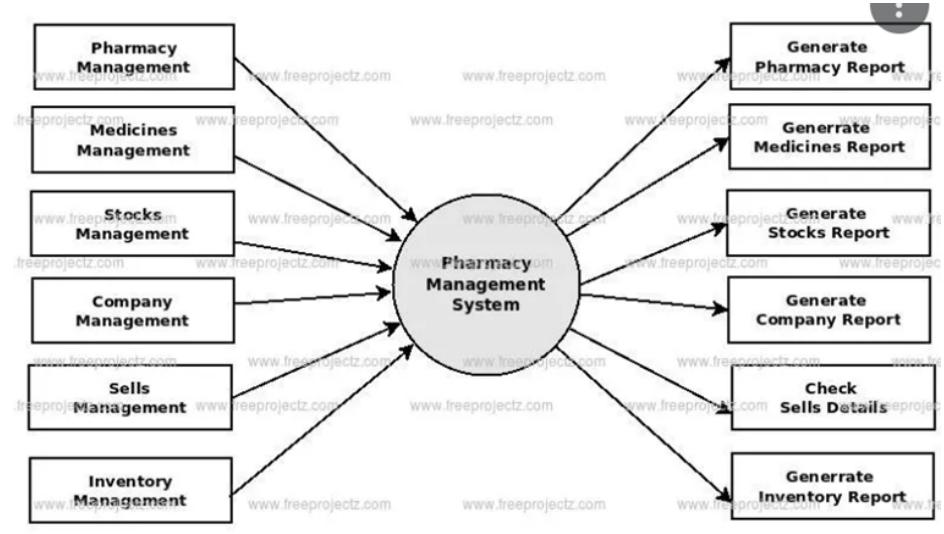
0-level DFD:

It is also known as a context diagram. It's designed to be an abstraction view, showing the system as a single process with its relationship to external entities. It represents the entire system as a single bubble with input and output data indicated by incoming/outgoing arrows.



1-level DFD:

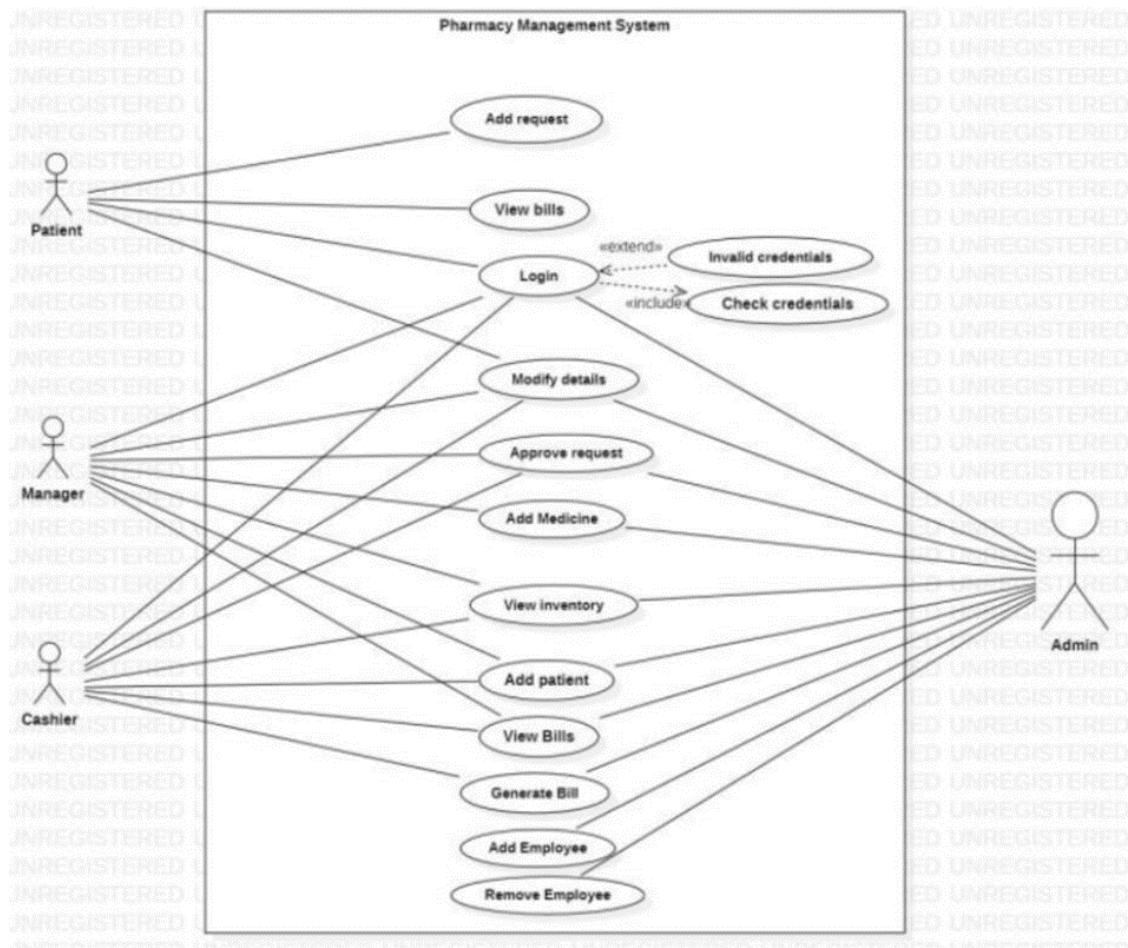
In 1-level DFD, the context diagram is decomposed into multiple bubbles/processes. In this level, we highlight the main functions of the system and breakdown the high-level process of 0-level DFD into subprocesses.



Chapter 4 - Designs

4.1 UML Designs

Use-Case Diagram:



1. **Actors:** Patient, Manager, Cashier, Admin

2. **Use Cases:**

a) Add request: The patient can put in a request for specific medicines to the pharmacy.

b) View Bills: The patient can view his history of medical bills. The manager, cashier and Admin can also view the history of bills of the patient to ensure there are no bills left to pay/ not paid yet.

c) Login: The patient, manager and cashier can login using a set of credentials unique to them. The system determines if these credentials are valid or not.

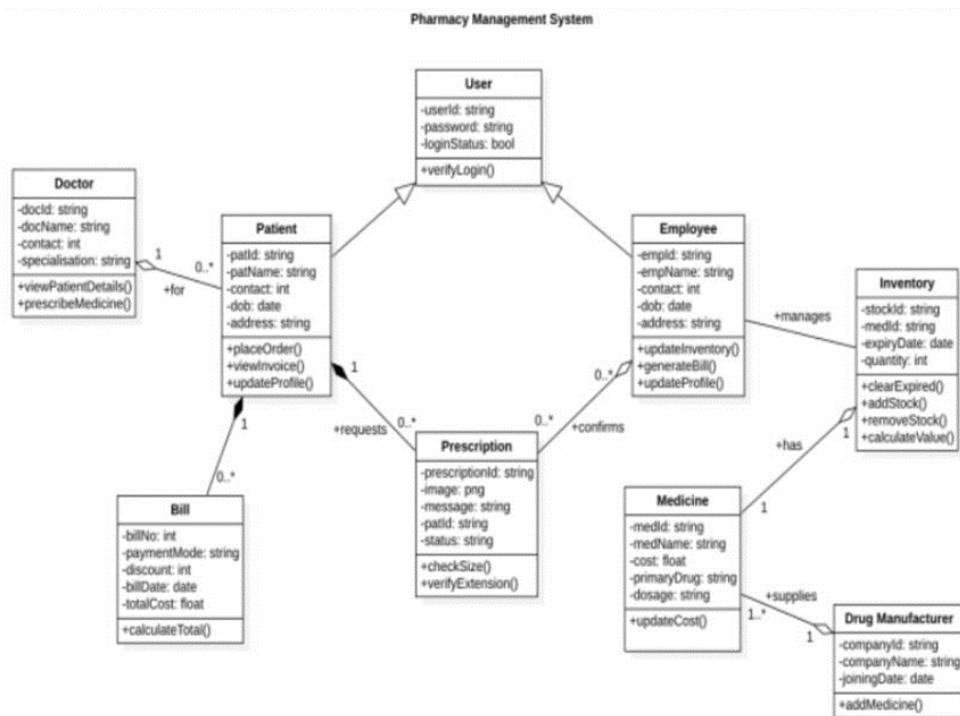
i) Invalid Credentials: It is an extension use-case of the Login use case, and is used if the credentials entered are incorrect.

ii) Check Credentials: Its functionalities are included in the Login use case, and they are used to check if the credentials entered by the user are valid.

d) Modify Details: The user, manager and cashier can modify the details of their account.

- e) Approve request: The manager, admin and cashier can approve of the request from the patient.
- f) Add medicine: The manager and Admin can add medicines they are newly introducing to the pharmacy.
- g) View Inventory: The manager, cashier and admin can view the inventory to see if any medicines are out of stock, expired or need to be restocked.
- h) Add Patient: The manager, cashier and Admin can add users if the patient is new and does not have an account.
- i) Generate Bill: The Cashier and Manager can generate a bill for the patient.
- j) Add / Remove employee: The Admin can remove existing employees and add new employees.

Class Diagram:



The class diagram depicts the entities of the model along with their relationships with each other. In the pharmacy management system, the classes are: Doctor, Patient, User, Employee, Inventory, Medicine, Drug Manufacturer, Prescription, and Bill.

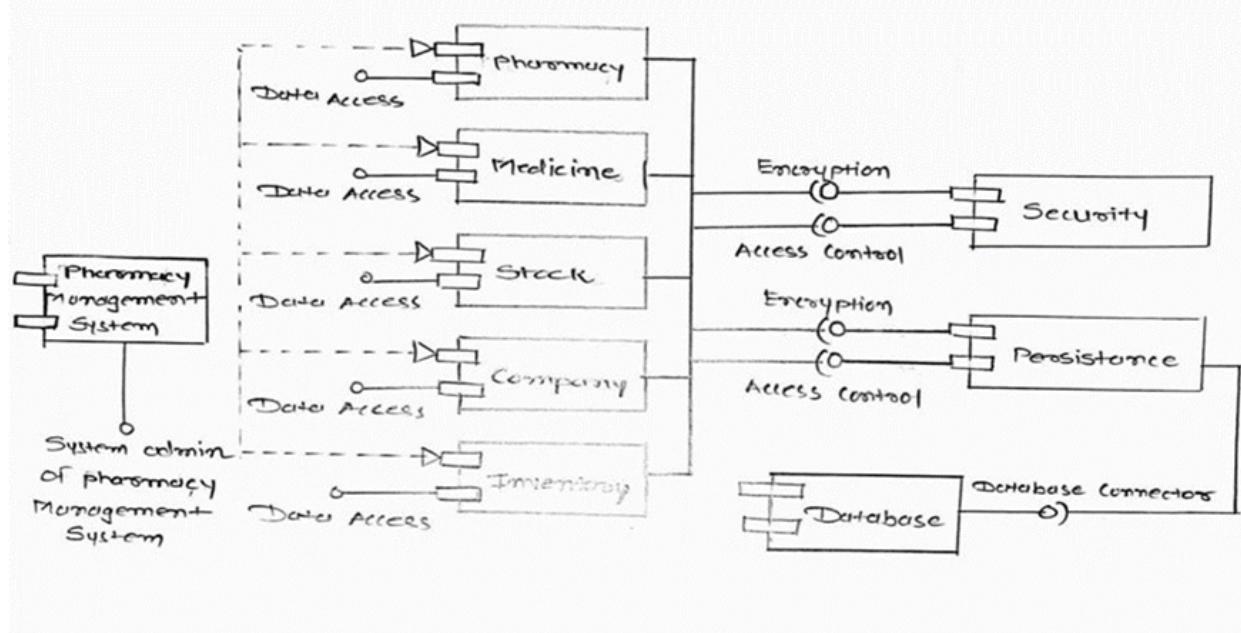
A doctor within the system can have 0 to many patients.

Both the patients and the employees need to be logged in before using the system.

The drug manufacturer supplies the medicines which are then stored in the inventory . A medicine can only be in one inventory. A drug manufacturer can supply one or more medicines.

The patient requests for the prescription and the patient confirms it. A patient can have multiple prescriptions. One or more bills can be generated per patient.

Component Diagram:

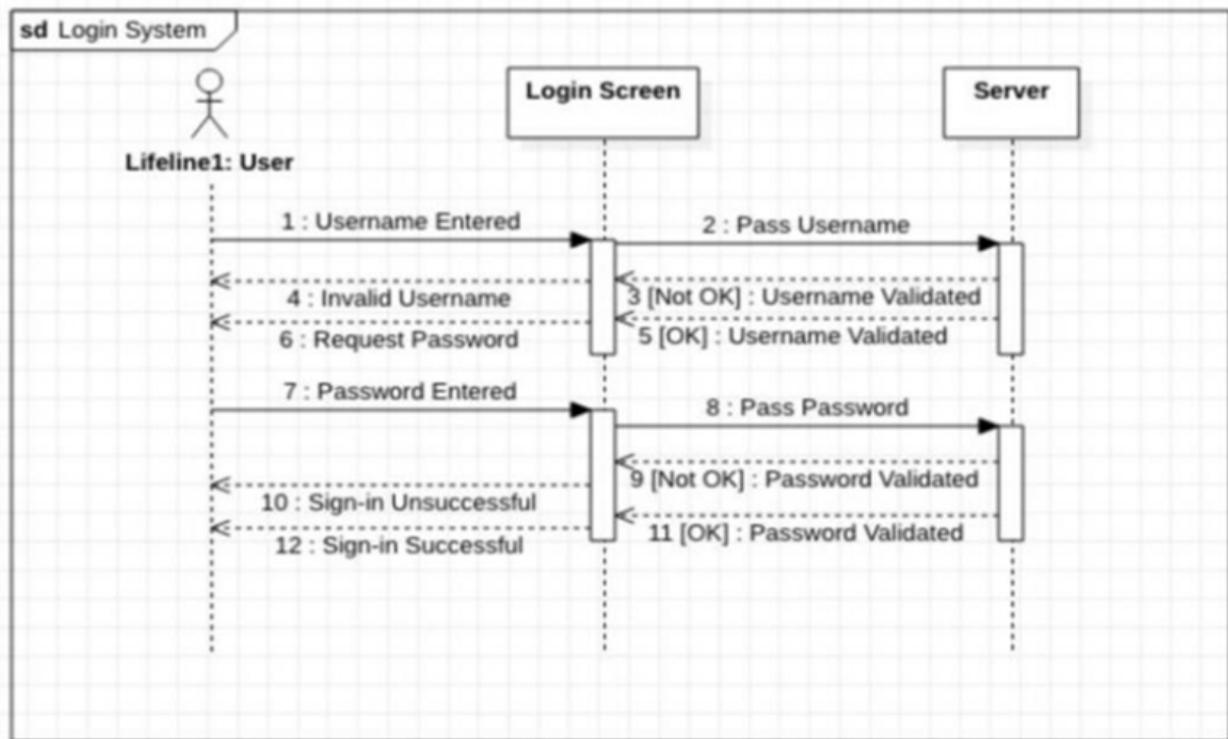


a component diagram depicts how the components are connected to form larger components or software systems. The components in the pharmacy management system are as follows:-

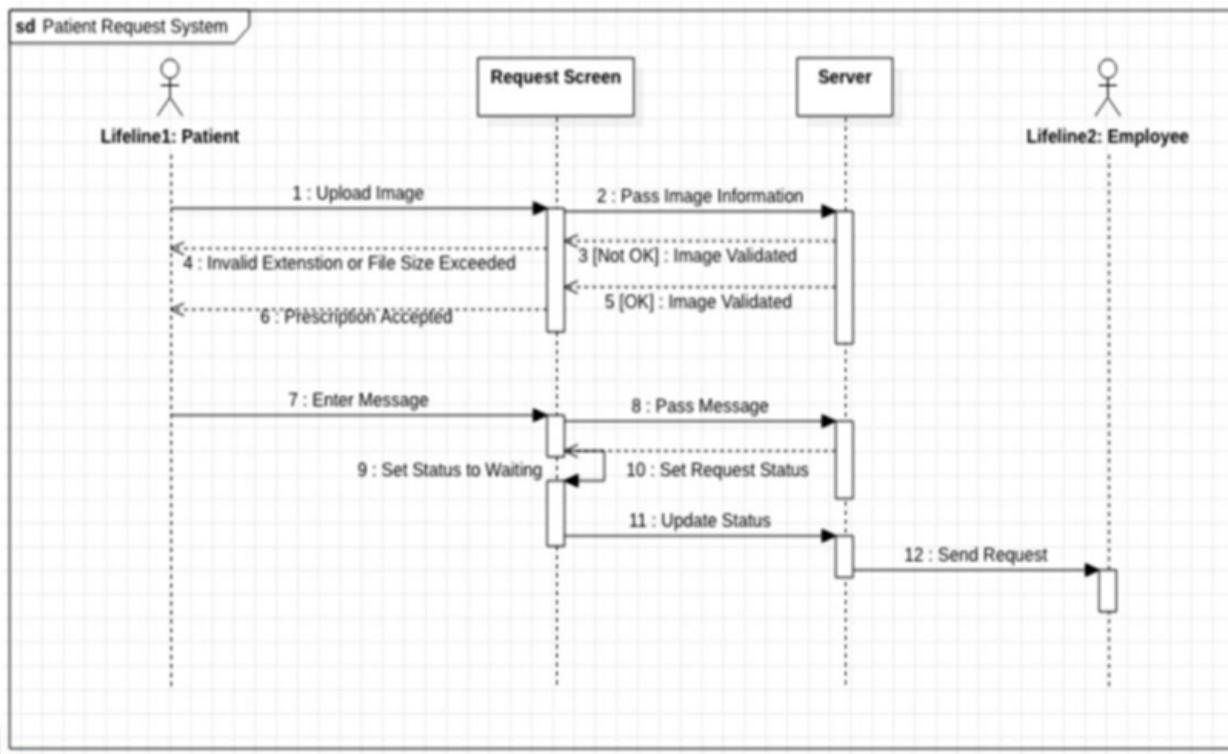
1. Inventory
2. Medicine
3. Pharmacy
4. Stock
5. Company
6. Security
7. Database
8. Persistence
9. Pharmacy management system

The pharmacy management system (Large component) contains the components : Inventory, Medicine, Pharmacy, Stock, Company which all have access to the respective data (Small Components). These smaller components are encrypted using the Security component and the access control and storing of sensitive information from these components is done by the persistence component and has connectivity to the database component.

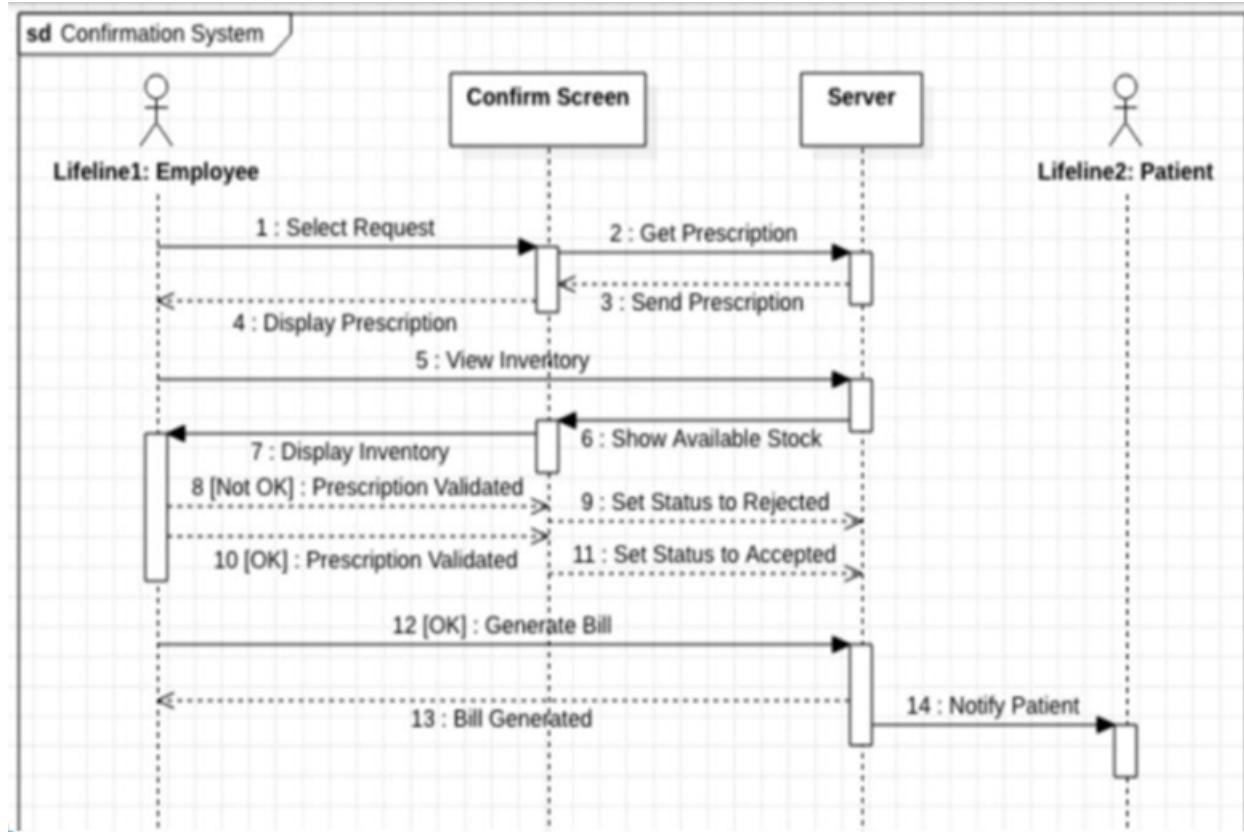
Sequence Diagram:



User enters the username into the login page and the server checks that the username is invalid or not. After that user enters the password and then server checks the credential from the database and if the account exists then server allows to enter otherwise not.

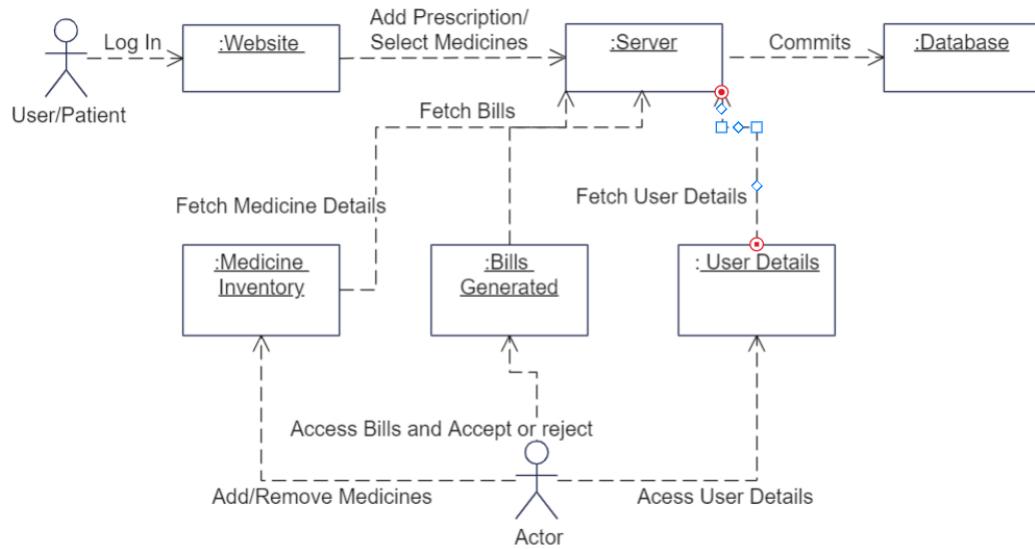


Now after login and entering into the system, the patient can upload the prescription(image), then the server checks whether the uploaded document is in valid format or not and also checks the size of the document. Patients can also add the message. After checking all the constraints, server transfers this document and message to the employee account.

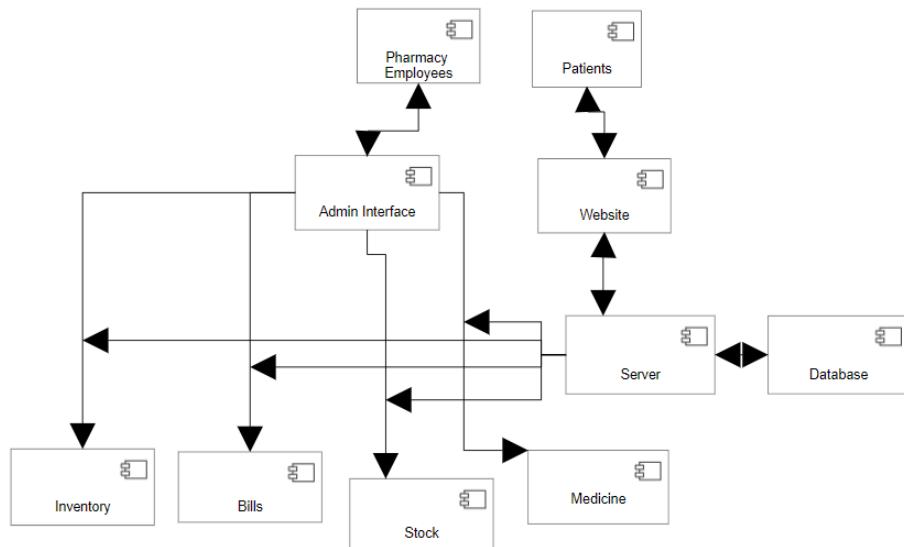


After getting the document and message from the server for the various patients, the employee can select the patient request and the server provides the prescription and message from the particular patient. Employees can view the inventory also. Server provides the available stock to the employee. Employees check whether this document is legit or not. If it is valid then set the status as accepted otherwise set as rejected. If the particular medicine is available then the employee can provide the medicine. At the end the employee generates the bill and the server provides this generated bill to the patient.

Collaboration Diagram:



Collaboration Diagram:



Chapter 5 - Development

5.1 Tools Description and Development approach used

The Interface of the Pharmacy Management System is an Interface which helps with an excellent experience of the pharmacy process. The Interface proposed for the pharmacy management system is developed using *HTML*, *CSS* and *JavaScript* for the front end, and *Node.js* and *PHP my Admin* for the back end, while using *MySQL* for the server.

HTML was used for the construction of the structure of the webpage.

CSS was used for the styling of the webpage.

Node.js was an excellent choice for this project as *Node.js* Provides excellent packages like express, body parser and cookie parser, which allow for inputting the data from the user on the web page to the database. Cookie parser helps in parsing the cookies which are stored on the client side.

Enhancing the system with *PHP myAdmin* was beneficial due to the clean processing and interface provided by the software. It also allows the efficient management of the databases of the pharmacy management system.

MySQL for the database was preferred as the best data storage tool due to its uniformity and management of data using tables, and also the ease of entering and accessing information from the tables. All of the tools combined provide a simple but efficient pharmacy management system.

Development Approach:-

The incremental development approach was chosen for the development of the pharmacy management system where each module was added to the system on an incremental basis. The increments also included the redesigning of the frontend web page and the re-allocation of data resources into the MySQL tables on the backend catered to the users experience. Each part of the iteration was used for includes the communication, planning, designing and development and feedback phases.

Reasons for Choosing the Incremental Approach:-

The main reason for choosing the incremental approach was the correction and rectification of potentially catastrophic or inconvenient errors in the system; this could not have been possible if models such as the waterfall model were chosen.

It allows for the feedback and review process which helps the development of the system as a whole to cater towards the users needs and add or remove the respective components into the system.

It allows for the system to be updated with increments regularly so that it does not go outdated, and is kept customer friendly at all times.

Issues Faces Using the Incremental Approach:-

The incremental approach brought up the problem of time as the constant iteration of the phases as increments are added into the system consumes extra time which could have caused delays in the development of the full model.

Chapter 6 - Testing

6.1 Test cases for Important Modules

Types of Testing

White box testing:

- Statement coverage testing has been performed to check if the stock of the medicine goes below 0.
- Statement coverage testing has been performed to check if the Quantity of the medicine in bill generation does not go below 0.
- Branch Coverage testing has been performed to check if valid patient name is selected in Bill generation.
- Branch Coverage testing has been performed to check if valid employee id is used to in Inventory Management
- Branch Coverage testing has been performed to check if the password is valid and length greater than 7

Black box testing:

- Authentication was tested.
- Request system was tested.
- Inventory management was also tested.
- Invoice Bill was Generated

TESTING REPORT - 1

Test Case ID	SWE_001	Test Case Description	Test the Prescription Request-Confirmation Functionality		
Created By	19BCI0012	Test Case (Pass/Fail)	Pass	Version	1
Serial	Prerequisites:		Serial	Test Data	
1	Access to Browser		1	File Extension = ".jpg"	
2	User id logged in		2	File Size = 2MB	
3	File Access Allowed				
Test Scenario	Verify on entering valid file extension and file size. If ok, the request is approved otherwise redirect back to request page				
Step	Step Detail	Expected Result	Actual Results	Pass/Fail/Not Executed/Suspended	
1	Navigate to Request Page	Site should open	As Expected	Pass	
2	Verify File Extension	File Extension is verified	As Expected	Pass	
3	Verify File Size	File size not exceeded	As Expected	Pass	
4	Click Submit Button	File Uploaded to server	As Expected	Pass	

TESTING REPORT - 2

Test Case ID	SWE_002	Test Case Description	Test the Login Functionality		
Created By	20BCE2803	Test Case (Pass/Fail)	Pass	Version	1
Serial	Prerequisites:		Serial	Test Data	
1	Access to Browser		1	Username='Shreyas'	
2	Database Access		2	Password = 12345	
Test Scenario	Verify on entering valid username and password, the customer can login				
Step	Step Detail	Expected Result	Actual Results	Pass/Fail/Not Executed/Suspended	
1	Navigate to Login Page	Site should open	As Expected	Pass	
2	Enter Username	Username is verified	As Expected	Pass	
3	Enter Password	Password is Verified	As Expected	Pass	
4	Click Submit Button	Customer is Logged in	As Expected	Pass	

TESTING REPORT - 3

Test Case ID	SWE_003	Test Case Description	Test the Inventory Functionality		
Created By	20BCE2362	Test Case (Pass/Fail)	Pass	Version	1
Serial	Prerequisites:		Serial	Test Data	
1	Access to Browser		1	Medicine ='Paracetamol'	
2	Admin/Manager Log in		2	Quantity =10	
3	Database		3	Expiry Date = 19-11-2021	
Test Scenario	Verify that on entering valid medicine quantity and expiry date, the medicine stock is updated.				
Step	Step Detail	Expected Result	Actual Results	Pass/Fail/Not Executed/Suspended	
1	Navigate to Inventory Page	Site should open	As Expected	Pass	
2	Enter the Name and Quantity	Detail is verified	As Expected	Pass	
3	Verify Expiry Date	Date is Verified	As Expected	Pass	
4	Click Update Button	Inventory Updated	As Expected	Pass	

TESTING REPORT - 4

Test Case ID	SWE_004	Test Case Description	Test to Generate the invoice bills		
Created By	19BCI0012	Test Case (Pass/Fail)	Pass	Version	1
Serial	Prerequisites:		Serial	Test Data	
1	Access to Browser		1	Patient ='Pbj'	
2	Admin/Cashier Log in		2	Medicine= 'Paracetamol'	
3	Database		3	Quantity =10	
			4	Doctor = 'Rahul Gandhi'	
			5	Discount = '200'	
			6	Type ='Card'	
Test Scenario	Verify that on entering valid Patient, Medicine, Quantity, Doctor, Discount, Type.				
Step	Step Detail	Expected Result	Actual Results	Pass/Fail/Not Executed/Suspended	
1	Navigate to Inventory Page	Site should open	As Expected	Pass	
2	Enter the Patient and Doctor	Detail is verified	As Expected	Pass	
3	Enter the Medicine, Quantity	Detail is verified	As Expected	Pass	
4	Enter Discount and Type	Date is Verified	As Expected	Pass	
5	Click Update Button	Bill Generated	As Expected	Pass	

Testing LOGIN INTERFACE

Valid Login: {Username: Pbj, Password: prabal2, exist in database}

The login interface consists of the following fields:

- Username:** Pbj
- Password:** (Redacted)
- Remember Me:** (unchecked checkbox)
- Login:** (orange button)

VIT Pharmacy	Login ID: 73	Username: Pbj	Role: Patient
Dashboard			
Invoice History			
Request Medicines			
Medicine History			

Invalid Login: {Username : Jaya, Password: kumar, not exist in database}

➡ Login

Username

Password

Remember Me

Login

➡ Login

No user found.

Username

Password

Remember Me

Login

Employee Mapping to Corresponding Roles: {Cashier/Admin/Manager}

Admin->Admin

➡ Login

Username

Password

Remember Me

Login

VIT Pharmacy

Login ID: 66

Username: vader

Role: Admin

Sign out

Employee ID	Employee Name	Old Username	New Username	Update Contact
11	Darth Vader	vader	vader	112233
Update Address				
Darth Nagar, Death Star				

- [Dashboard](#)
- [Manage Users](#)
- [Generate bill](#)
- [Patient](#)
- [Doctors](#)
- [Inventory](#)

Cashier->Cashier

 **Login**

Username

Password

Remember Me

Login

Manager->Manager

 **Login**

Username

Password

Remember Me

Login

VIT Pharmacy

Login ID: 74

Username: tony

Role: Manager

Sign out

Employee ID	Employee Name	Old Username	New Username	Update Contact
15	Tony	tony	tony	2424113000
Update Address				
Stark Tower				

Update

- [Dashboard](#)
- [Manage Users](#)
- [Patient](#)
- [Doctors](#)
- [Inventory](#)
- [Invoice History](#)
- [Requests by patients](#)

Testing Add Patient Functionality: {Patient Name/Password/Doctor}

Case1: Valid Patient

Add a Patient

Patient Name	Contact					
Shivam	98787690012					
Address	High Court Road					
Gender	Age	Insurance ID	Password			
Male	12	Mi4456	*****			
Doctor Name	Doctor B (10)					
Add						
72	Abhirag	8000070000	Asha Town	Male	25	Pranav S
73	Shivam	98787690012	High Court Road	Male	12	Doctor B

Case2: Invalid Entry due to Wrong password length

Add a Patient

Patient Name	Contact			
Shivam	98787690012			
Address	High Court Road			
Gender	Age	Insurance ID	Password	
Male	12	Mi4456	***	
Doctor Name	Kamla Harris (4)	! Please lengthen this text to 7 characters or more (you are currently using 3 characters)		
Add				

Testing File Upload in Add Request Functionality: File Upload Valid

Login ID: 73 Username: Pbj Role: Patient

Add a Message

Message
I need Medicine

Image
Choose File patient_login_1.png

Add

Invalid Upload Same File Again

Add a Message

Message

I need Medicine

Image

test Cases.xlsx

Chapter 7 - Screenshots of Developed Product

→ Login

Username

Password

Remember Me

Login

Fig. (a) Login page: The admins, patients and other employees login through this login page using their username and password, after which they are taken to their appropriate dashboards.

Admin:

VIT Pharmacy Login ID: 6 Username: tester Role: Admin Sign out

Employee Details

Employee ID	Employee Name	Old Username	New Username	Update Contact
1	temp	tester	tester	12345

Update Address
New add

Update

- Admins can manage users, generate bills for patients, view the details of patients and doctors registered, inventory of all medicines, Invoice history of the patients and any requests made by patients.

VIT Pharmacy

Login ID: 6

Username: tester

Role: Admin

Sign out

Dashboard

[Manage Users](#)

Generate bill

Patient

Doctors

Inventory

Invoice History

Requests by patients

Employee details

[Remove an Employee](#)

Employee ID	Name	Contact	Address	Username	Role
1	temp	12345	Nwe add	tester	Admin
9	Xyz	123456	XYZPur.Xyz	xyz	Cashier
11	Kirit	12345678	ABC STREET	kiritpillai	Admin
12	Aryan	12345678	DEF City	aryan1	Cashier
14	Khushal	123456789	XYZ	khushal	Admin
15	Maria	123456789	XYZ	maria133	Cashier
16	Viraj	3209321211	Delhi	viraj123	Cashier

Add an Employee

Employee Name

Employee name	Contact	DOB
	Contact No.	dd/mm/yyyy

Address

1234 Main St

Username

Username	Password	Role
tester	*****	Cashier

[Add](#)

Fig, (a)- Manage users such as other admins, cashiers and managers.

VIT Pharmacy

Login ID: 6

Username: tester

Role: Admin

Sign out

Dashboard

[Manage Users](#)

Generate bill

Patient

Doctors

Inventory

Invoice History

Requests by patients

Click here to check medicines list

VIT Pharmacy | Invoice

Bill No. Patient Id Patient Name Doctor Name Age Billing Date

22	68	Balaji Vyas	Kamla Harris	10/11/2022
----	----	-------------	--------------	------------

Sr. No.	Medicine Name	Medicine ID	Stock ID	MRP (In ₹)	Quantity	Expiry Date	Total
1	Wellbutrin	7	19	4096	2	21/06/2025	8192
2	Xanax	30	24	904	1	31/05/2023	904
							Total 9096

Payment Mode

Cash	Discount (in ₹)
	100

Final Cost

8996

[Generate a Bill](#)

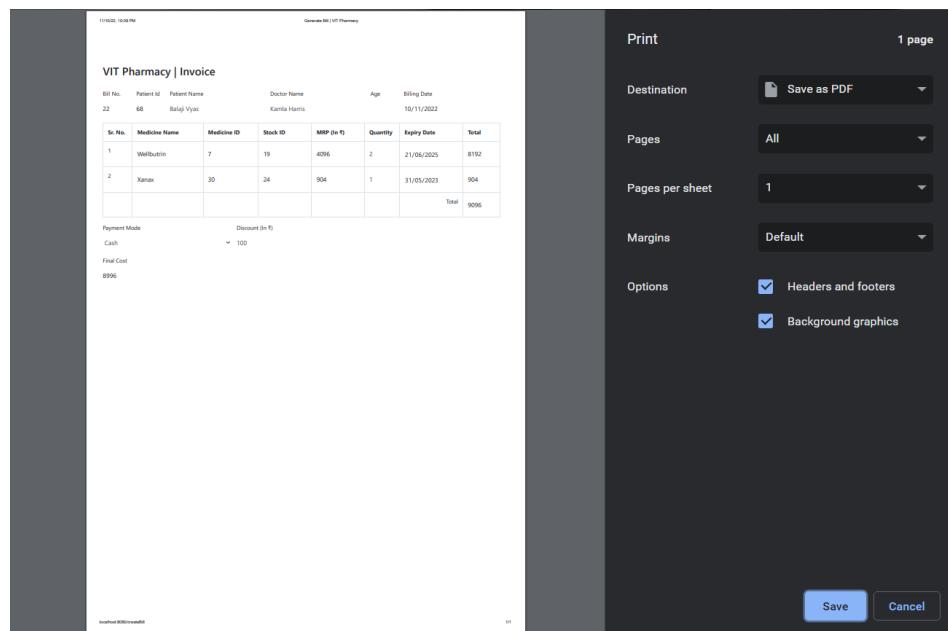
Medicine List

Search a medicine

Medicine ID	Stock ID	Medicine Name	MRP (In ₹)	Company Name	Expiry Date	Available Stock
7	19	Wellbutrin	4096	Cadila Healthcare	06-21-2025	50
9	22	Metoprolol	804	Abbott	11-24-2028	50
14	33	Levamlodipine	110	Biosense	04-31-2029	75

Fig (b). The generate bill feature enables the admin to generate an invoice for a patient using the medicines currently in stock (see bottom)

- On generating the bill, the user is prompted to save the invoice with all the relevant details as a pdf file.



- This invoice generated then reflects in the invoice history.

Invoice History						
Search a bill by patient name						
Bill No.	Patient Name	Payment Mode	Bill Amount	Discount	Billing Date	
21	Balaji Chandra	Cash	1808	0		
22	Balaji Vyas	Cash	8996	100		

- The admin can also add a patient by entering relevant details, and this gets reflected at the bottom of the same screen.

- The admin adds doctors and their contact and specialization to the database, so that the patients can easily book an appointment.

Add a Doctor

Doctor Name

Contact

Specialization
 Ex: Cardiology

Add

Doctor Details

Search a doctor

Name	Contact	Specialization
Abhishek Ramgirkar	1234567890	Cardiology
Ranjit Gandhi	4567890123	Radiology
Siddharth Modi	1237890456	Ophthalmology
Kamla Harris	1278903456	Oncology
Rahul Gandhi	5678901234	Neurology
Rama Ramachandran	9900877212	Ophthalmology

- Here the admin can see the medicine list. We can either add new medicines to this list or add stock of medicines from this list to the inventory.

VIT Pharmacy

Dashboard
Manage Users
Generate bill
Patient
Doctors
Inventory
Invoice History
Requests by patients

Login ID: 6 Username: tester Role: Admin Sign out

Add new Stock

Medicine Name Medicine ID

Batch Expiry Date dd/mm/yyyy

New Stock Quantity

Associated Employee Name Associated Employee ID

Add

Add new Medicine

Medicine Name MRP

Primary Drug

Dosage

Company Name Company ID

Add

VIT Pharmacy	Login ID: 6	Username: tester	Role: Admin	Sign out
Medicine List				
Medicine ID	Medicine Name	MRP	Primary Drug	Company Name
2	Lansoprazole	₹120	Amoxicillin	Ranbaxy
3	Hydrochlorothiazide	₹56	Myalept	Sun Pharmaceutical
4	Benzonatate	₹60	Ravicti	Cipla Limited
5	Coronil	₹2040	Mavenclad	Abbott
6	Xanax	₹904	Actimmune	Dr. Reddy's Laboratories
7	Wellbutrin	₹4096	Oxervate	Cadila Healthcare
8	Viagra	₹420	Takhyro	Biocon Limited
9	Metoprolol	₹804	Daraprim	Abbott
10	Kevzara	₹36	Juxtapid	Sun Pharmaceutical
11	Metformin	₹1118	Cinryze	Cipla Limited
12	Gilenya	₹917	Clindamycin	Biocon Limited

- Finally, the admin can view message requests for medicines from patients.

Cashier:

- The cashier is an employee who can *generate bills for patients, add patients who are new to the system when they place a prescription, check doctor details, inventory and invoice history*.

VIT Pharmacy	Login ID: 73	Username: viraj123	Role: Cashier	Sign out
Employee Details				
Employee ID	Employee Name	Old Username	New Username	Update Contact
16	Viraj	viraj123	viraj123	3209321211
Update Address	Delhi			
	Update			

Patient:

VIT Pharmacy	Login ID: 63	Username: Pewdiepie	Role: Patient	Sign out
Patient Details				
Patient ID	Patient Name	Old Username	New Username	Update Contact
60	Pewdiepie	Pewdiepie	Pewdiepie	98402802
Update Address	Pewdss			
	Update			

- In the Requests page, a patient can add an image of their prescription and give details as to which medicine they are requesting. The request history of that patient will show below.

The screenshot shows a web-based application for managing prescription requests. At the top, there's a navigation bar with links for Dashboard, Invoice History, Request Medicines, and Medicine History. The main area has a header "Add a Message". Below it, there are two input fields: "Message" (containing placeholder text "Type your message here") and "Image" (with a file upload input showing "Choose file No file chosen"). A "Send" button is located below these fields. To the right, there's a section titled "Request History" containing a table with six rows of data. The columns in the table are "Message", "Date", and "Status". The data entries are:

Message	Date	Status
Thyronorm 100mg needed	Fri Feb 26 2021 15:51:04 GMT+0530 (India Standard Time)	Request awaiting confirmation
Send meds	Fri Feb 26 2021 15:52:25 GMT+0530 (India Standard Time)	Request awaiting confirmation
Thyronorm 200mg needed	Fri Feb 26 2021 22:47:58 GMT+0530 (India Standard Time)	Approved
Abc	Fri Feb 26 2021 22:53:38 GMT+0530 (India Standard Time)	Request awaiting confirmation
Thyronorm 50mg needed	Sun Feb 28 2021 14:48:21 GMT+0530 (India Standard Time)	Request awaiting confirmation
Need the medicines	Tue Apr 13 2021 10:51:01 GMT+0530 (India Standard Time)	Request awaiting confirmation

Chapter 8 - Conclusion and References

8.1 Conclusion

In this project, we developed and designed a system which ensures effective data saving and manipulation related to a pharmacy. In this era of growing diseases there has been a surge in medical companies and medicines in general. Therefore, we developed an efficient and user-friendly software to manage various needs of pharmacies. We performed a literature review of recent papers related to pharmacy management and realized that most pharmacies are paper based and even the online software systems are one-sided designed mainly for employees and inventory. We developed a system in which patients can directly request for medicines by uploading the image of the prescription. We also developed an inventory management system which can be used to manage medicine stock and details. We listed the various functional and non-functional requirements of the project and also highlighted the various modules encompassing it. Finally, we perform testing by considering various test cases in each sub-system. Although this system is very all-inclusive of most of the necessities of a pharmacy, it still has some limitations. Prescriptions uploaded by patients have to be manually verified for a bill to be generated. With improvements like automatic processing of prescriptions, and more effective database management, we can develop a much more scalable and efficient system.

8.2 References

1. Sobek, Durward & Jimmerson, Cindy. (2003). C: Applying the Toyota Production System to a Hospital Pharmacy.
2. Zangana, Hewa. (2018). *Design an Information Management System for a Pharmacy*. 7. 52-55. 10.17148/IJARCCE.2018.71012.
3. Node.js tutorial. (n.d.). Retrieved November 9, 2022, from <https://www.w3schools.com/nodejs/>
4. HTML tutorial. (n.d.). Retrieved November 9, 2022, from <https://www.w3schools.com/html/>
5. W3.CSS tutorial. W3.CSS Home. (n.d.). Retrieved November 9, 2022, from <https://www.w3schools.com/w3css/default.asp>
6. JavaScript tutorial. (n.d.). Retrieved November 9, 2022, from <https://www.w3schools.com/js/>
7. SQL tutorial. (n.d.). Retrieved November 9, 2022, from <https://www.w3schools.com/sql/>
8. PHP mysql database. PHP: MySQL Database. (n.d.). Retrieved November 9, 2022, from https://www.w3schools.com/php/php_mysql_intro.asp