**Pharmacy System**



Session: 2023 – 2027

# Submitted by:

Salman Naseem 2023-CS-78

Huzaifa Arshad 2023-CS-86

# Supervised by:

Mr. Nazeef Ul Haq

Department of Computer Science

# University of Engineering and Technology Lahore Pakistan

### **Table of Contents**

1. **Project Overview**  
   1.1 Summary  
   1.2 Motivation  
   1.3 Objectives
2. **Target Audience and Stakeholders**  
   2.1 Target Audience  
   2.2 Stakeholders
3. **Operational Details**  
   3.1 Medicine Management  
   3.2 Order Management  
   3.3 Prescription Management  
   3.4 Supplier Management  
   3.5 User Roles
4. **Use Cases**  
   4.1 Pharmacist Use Cases  
   4.2 Customer Use Cases  
   4.3 Supplier Use Cases
5. **Wireframes**  
   5.1 Main Logo  
   5.2 Main Menu  
   5.3 Pharmacist Menu  
   5.4 Add Medicine  
   5.5 Add Prescription  
   5.6 Customer Menu  
   5.7 Place Order  
   5.8 View Medicines  
   5.9 Receive Orders  
   5.10 View Prescriptions  
   5.11 Supplier Menu  
   5.12 Add Supplier  
   5.13 Connection Added Between Suppliers  
   5.14 Supplier Network  
   5.15 Shortest Path to Supply Medicine

**Pharmacy System**

* **Project Overview**
* **Summary**

The Pharmacy System is a comprehensive solution designed to streamline pharmacy operations, including medicine management, customer order handling, prescription tracking, and supplier relationships. The system is divided into specific modules that manage key tasks:

* **Medicine Management**: Add, remove, and search for medicines while tracking inventory details like quantity and price.
* **Order Management**: Utilize a priority queue to process customer orders based on urgency and confirm orders to ensure the correct medicines are dispensed.
* **Prescription Management**: Track and manage patient prescriptions, allowing pharmacists to add, view, and display prescriptions with ease.
* **Supplier Management**: Leverage a graph structure to track suppliers, manage relationships, and calculate the shortest path for efficient stock replenishment.

The system also supports different user roles, including pharmacists, customers, and suppliers, each with distinct functionalities. Ultimately, the Pharmacy System aims to enhance operational efficiency, reduce errors, and improve the accuracy and timeliness of pharmacy operations.

* **Motivation**

The motivation behind developing the Pharmacy System project is to address the inefficiencies and errors commonly found in traditional pharmacy management systems. In manual systems, tasks such as inventory management, order processing, and supplier coordination are prone to human errors, delays, and miscommunication, which can negatively impact pharmacy operations.

The Pharmacy System seeks to streamline these processes by automating tasks, reducing errors, and improving the accuracy and efficiency of managing medicines, orders, prescriptions, and supplier relationships. Key features of the system include:

* **Priority Queue for Order Management**: Ensures timely processing of orders based on urgency.
* **Graph-Based Supplier Network**: Optimizes stock replenishment by managing supplier data and relationships.
* **Comprehensive Data Management**: Generates reports and maintains accurate records of medicines, orders, and prescriptions.

By automating tasks and improving communication across different stakeholders, the Pharmacy System ensures better customer service, optimized supplier coordination, and more effective pharmacy operations, ultimately leading to improved pharmacy management and a more efficient workflow.

* **Objectives**

1. **Data Structures and Algorithms**: To design and implement key data structures such as stacks, queues, linked lists, graphs, and trees. The goal is to demonstrate a deep understanding of their properties, use cases, and applications in real-world scenarios.
2. **Optimization**: To apply efficient algorithms that solve common problems, ensuring the system performs optimally in terms of time and space complexity, especially when handling large datasets or complex operations.
3. **System Performance and Scalability**: To leverage appropriate data structures and algorithms to improve the overall performance, scalability, and responsiveness of the system, ensuring it can handle large-scale operations effectively.

* **Target Audience**

Professors of Data Structures and Algorithms

* **Stakeholders**

Key stakeholders for the Pharmacy System project include:

* **Pharmacists**: Manage the day-to-day operations of the pharmacy, including inventory, customer orders, and prescriptions.
* **Customers**: Place orders and interact with the system to purchase medicines and view their prescriptions.
* **Suppliers**: Provide medicines and manage supply chain relationships, ensuring timely delivery and stock availability.
* **System Administrators**: Maintain the system, ensure smooth operations, and handle user management and system updates.

Each stakeholder plays a critical role in ensuring the efficient functioning of the Pharmacy System.

* **Operational Details**

The Pharmacy System includes several key components that work together to ensure smooth and efficient operations:

* **Medical Management**
* Pharmacists can add, remove, and update medicines with details such as quantity and price.
* Inventory tracking ensures pharmacists know the available stock and can adjust quantities as needed.

### **Order Management**:

* Orders are processed using a priority queue system, which ensures that orders are handled based on urgency.
* Pharmacists confirm orders, ensuring correct medicines are dispensed to customers in a timely manner.
* **Prescription Management**:
* Pharmacists can add, view, and manage patient prescriptions, ensuring accurate record-keeping and preventing medication errors.
* **Supplier Management**:
* A graph structure is used to manage supplier data, enabling the addition, removal, and connection of suppliers.
* The system can display the shortest path between suppliers, helping ensure the efficient flow of medicines and timely stock replenishment
* **User Roles**:
* **Pharmacists** can manage medicines, orders, and prescriptions.
* **Customers** can view available medicines, place orders, and check their prescriptions.
* **Suppliers** can manage their profiles and supply chain relationships.

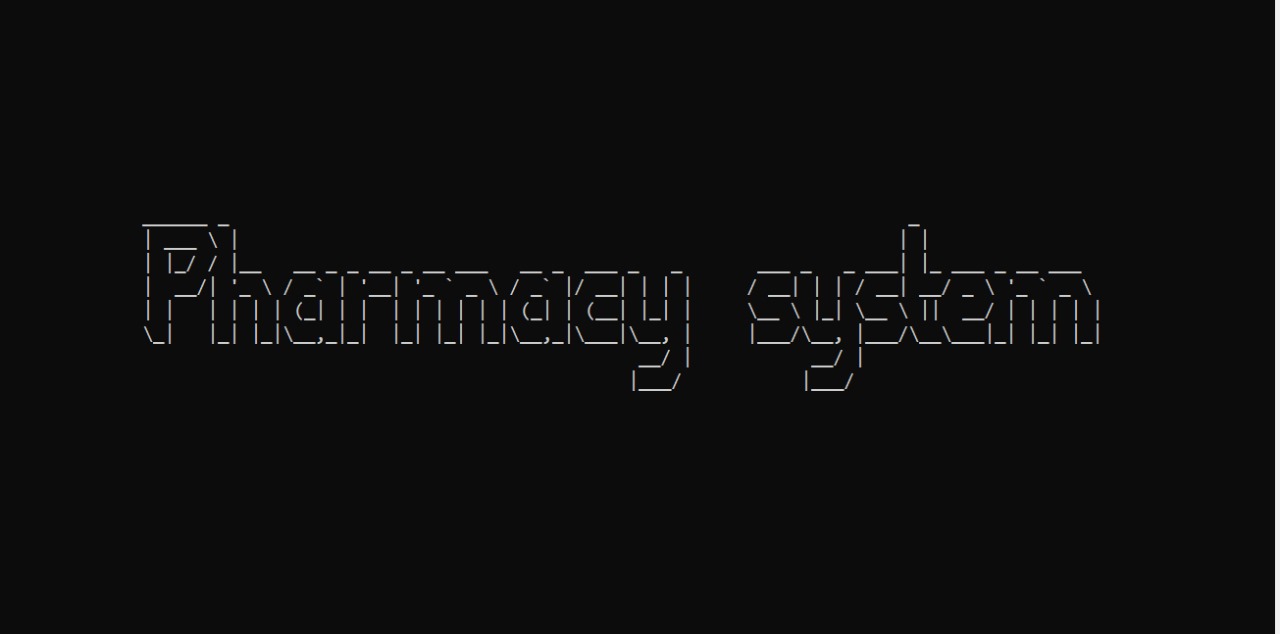
The system is designed to improve the efficiency of pharmacy operations, reduce errors, and streamline communication among pharmacists, customers, and suppliers. It automates key processes and ensures better coordination, ultimately enhancing overall operational effectiveness and reducing miscommunication.

* **Use Cases**
* **Pharmacist Use Cases**
* **Add Medicine**: The pharmacist can add new medicines to the system by providing details such as the name, price, and quantity of the medicine.
* **Remove Medicine**: The pharmacist can remove a medicine from the system if it is no longer available or needed.
* **Search Medicine**: The pharmacist can search for a particular medicine by name to check its availability, price, and stock.
* **Display Medicines**: The pharmacist can display a list of all available medicines with their details such as name, price, and stock quantity.
* **Add Order**: The pharmacist can add customer orders to a priority queue, specifying the customer name, medicine, quantity, and priority.
* **Confirm Orders**: The pharmacist can confirm orders based on priority, ensuring that the medicines are available and dispensing them accordingly.
* **Add Prescription**: The pharmacist can add a prescription for a patient, including prescription details like the patient's ID and prescribed treatment.
* **View Prescription**: The pharmacist can view a patient's prescription by entering the patient's ID and display all prescriptions associated with that patient.
* **Display All Prescriptions**: The pharmacist can display a list of all prescriptions in the system.
* **Customer Use Cases**
* **View Medicines**: The customer can view a list of all available medicines with details like name and price.
* **Search Medicine**: The customer can search for specific medicines by name to check availability and price.
* **Place Order**: The customer can place an order for a specific medicine by providing details such as the medicine name and quantity.
* **Display Order Queue**: The customer can view the status of their order queue and track the progress of their order.
* **Supplier Use Cases**
* **Add Supplier**: The supplier can add their details to the system, enabling them to become part of the supplier network.
* **Add Connection**: The supplier can establish connections with other suppliers, representing relationships or logistics partnerships between suppliers.
* **Remove Supplier**: The supplier can remove themselves from the system if they are no longer part of the network.
* **Remove Connection**: The supplier can remove connections with other suppliers, altering the relationships in the supplier network.
* **Display Suppliers**: The supplier can view a list of all suppliers in the system and their respective details.
* **Display Shortest Path**: The supplier can request the shortest path between two suppliers in the network, using the graph to determine the most efficient route or connection.
* **Wireframes**

Following are the wireframes of the pharmacy management system which is the final project of data structure and Algorithms.

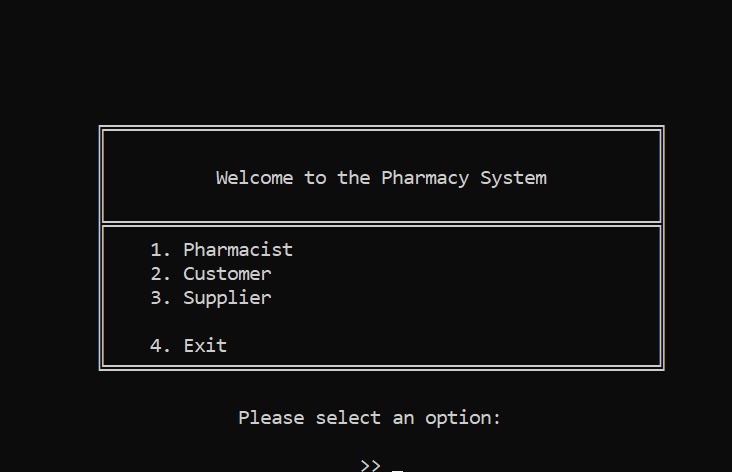
* **Mainlogo**

As we run our program, This wireframe will be shown as the logo or our pharmacy system.

****

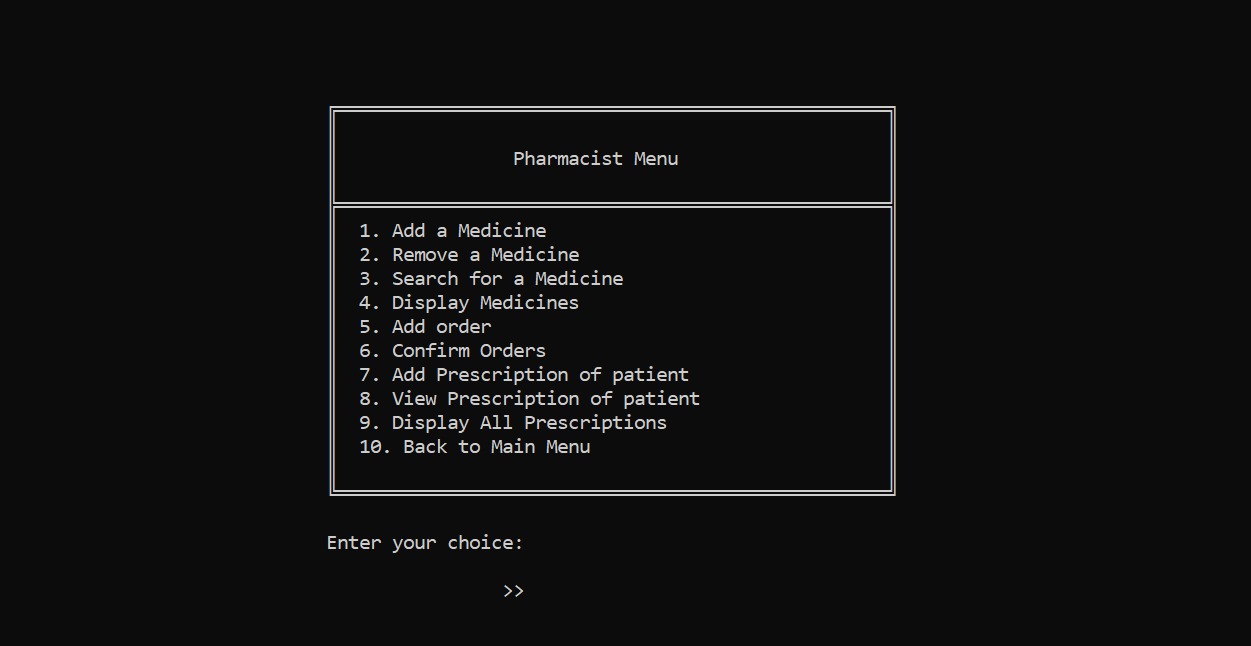
* **Main Menu**

This is the main menu of the pharmacy system, providing access to three user roles: Pharmacist, Customer, and Supplier. Each user can perform specific operations related to their role. The menu streamlines navigation, ensuring efficient management and user interaction.

****

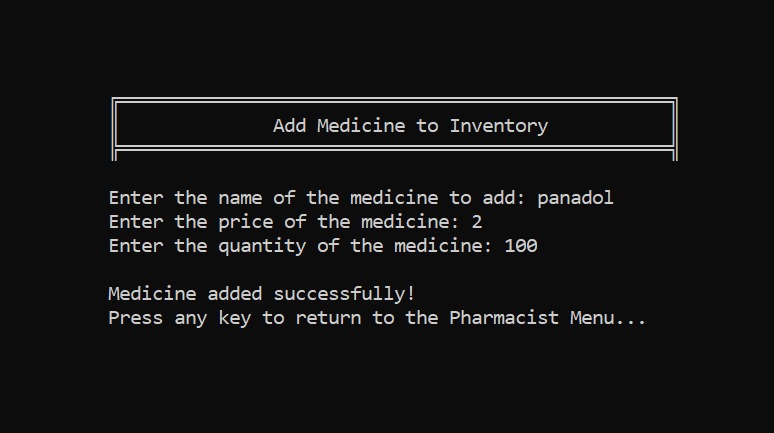
* **Pharmacist Menu**

This is the Pharmacist Menu that includes different functionalities. It includes Add, Remove, Search, Display Medicines, Add order, Confirm order, Add Prescription of patient, View Prescription of the patient and Display all prescriptions.



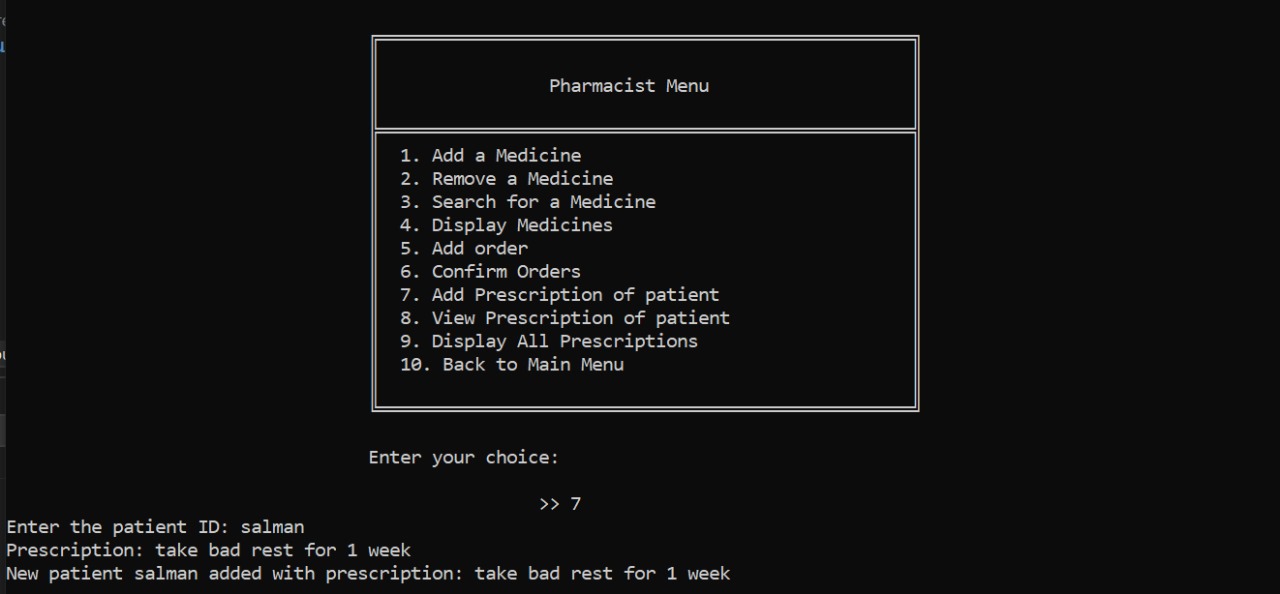
* **Add Medicine**

This functionality allows the pharmacist to manage the inventory by adding medicines using a Binary Search Tree (BST). The BST ensures efficient storage and retrieval of medicine data. It helps maintain a well-organized inventory, allowing quick access to medicines when needed.

****

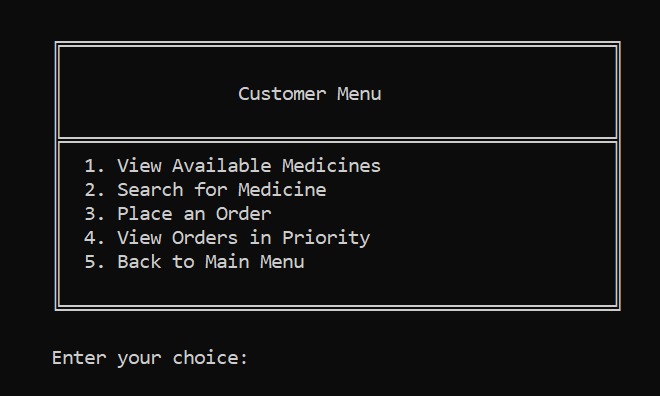
* **Add Prescription**

In this wireframe, we implement Hash Table. In this functionality, the prescription of the patient is added. It uses the hashing which is the core concept of the data structure and algorithms.

****

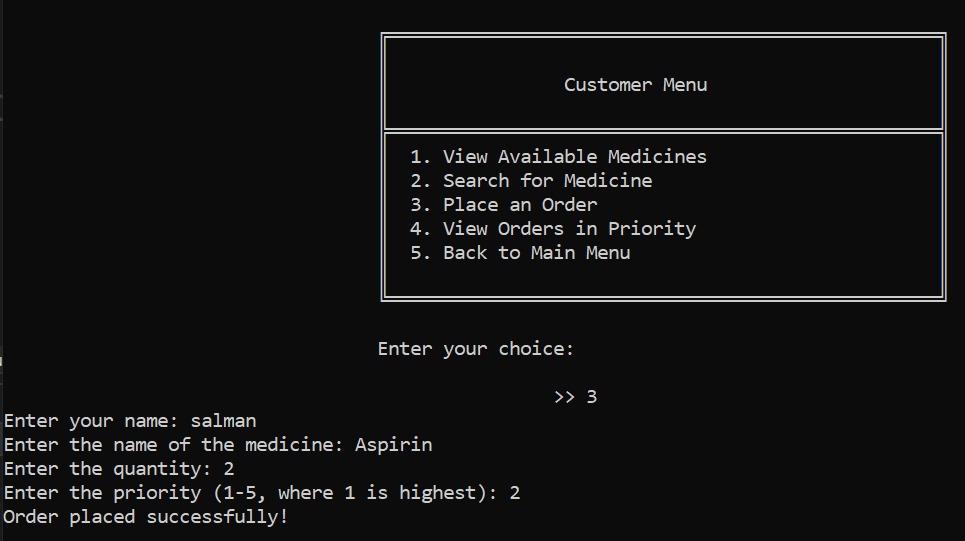
* **Customer Menu**

The customer menu allows users to view available medicines, search for specific products, and place orders. It utilizes a queue to manage and prioritize customer orders efficiently. This ensures a smooth order process and timely delivery based on order priority.

****

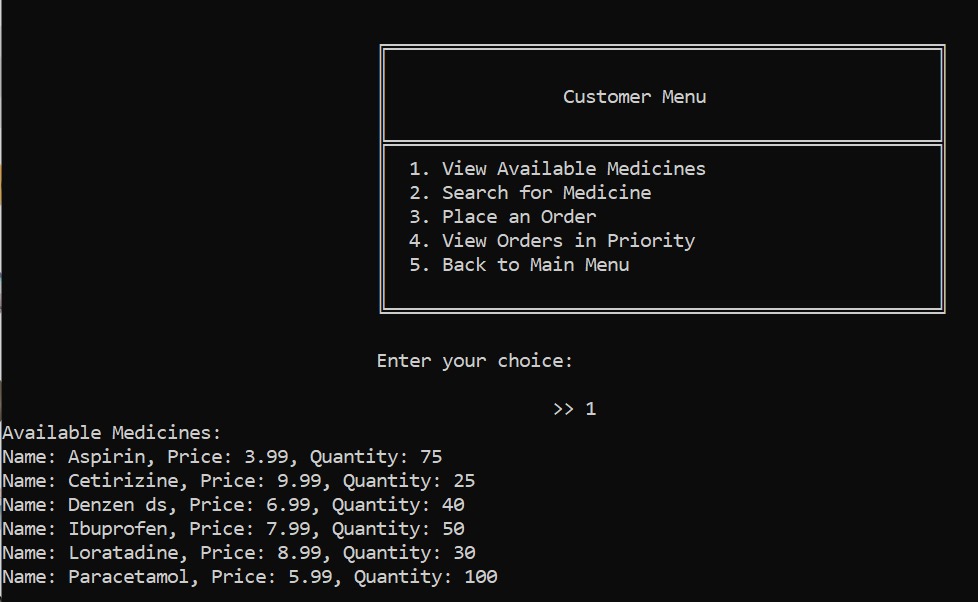
* **Place Order**

In this wireframe, the customer place an order which the pharmacist will later confirm if there is the stock of that medicine. It asks the name of medicine and priority of the medicine. There is the implementation of priority queue in this functionality and the pharmacist will receive the order based on priority.

****

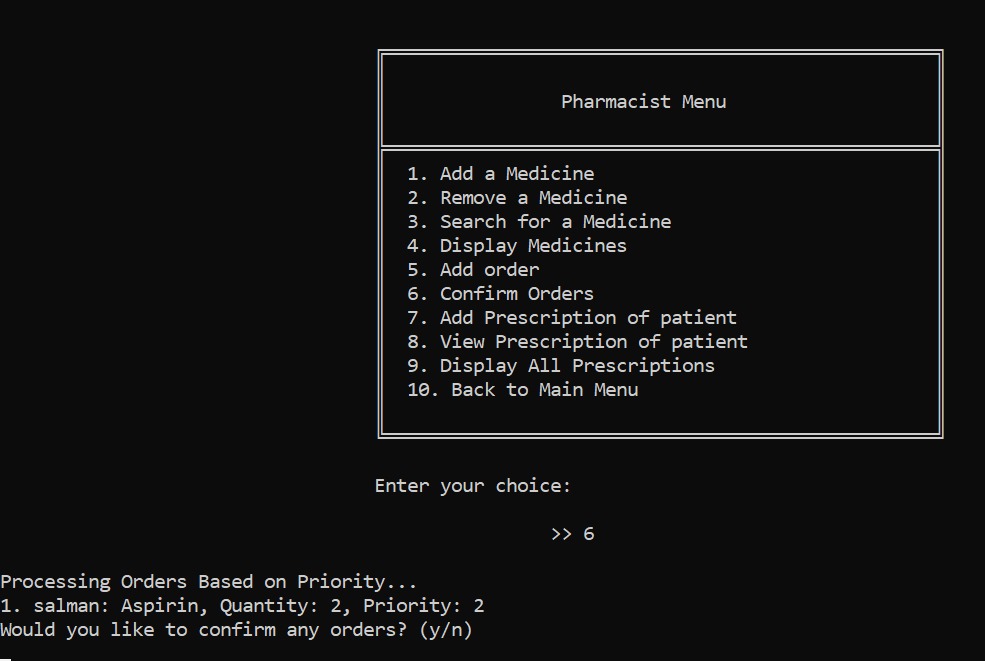
* **View Medicines**

In this wireframe, the customer views all the available medicines. Pharmacist also view medicines. It implements Inorder traversal of Binary Search Tree which gives sorted list of medicines based on the first character of medicine name.

****

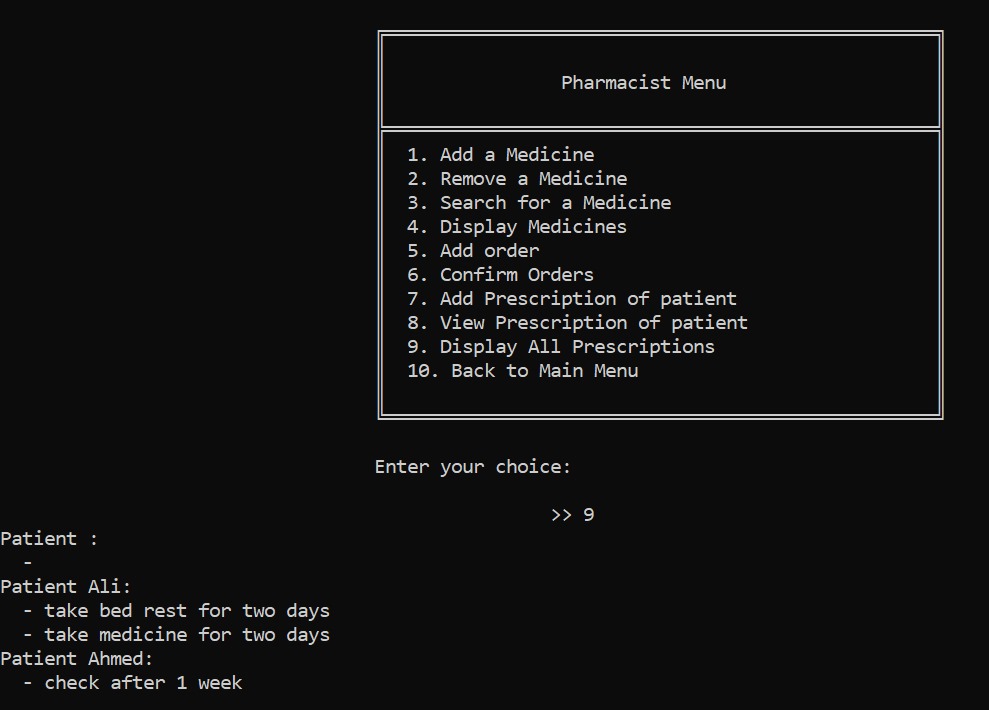
* **Receive Orders**

Pharmacist receives the order from customer based on the priority and then confirm the order if that medicines will be available in the stock. If the Aspirin has the priority 3 and the Panadol has the priority 4 then the pharmacist will receive the order of Panadol because it has high priority.



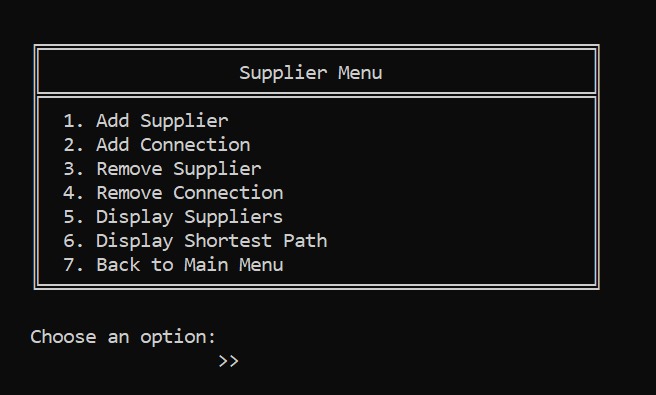
* **View Prescriptions**

In this wireframe, the pharmacist view all prescriptions of the patients that what prescription is given to Huzaifa and which prescription is given to the Salman. It will get data from the Hash Table.



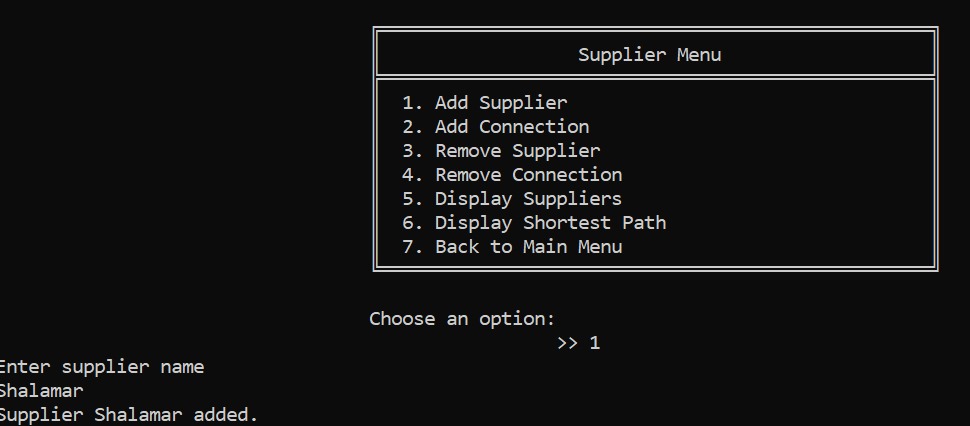
* **Supplier Menu**

The Supplier Menu manages medicine suppliers using a weighted graph, where nodes represent supplier locations, and edges represent distances between them. It optimizes supplier routes by calculating shortest paths and ensuring efficient logistics. Users can add, update, or view supplier details and their connections.



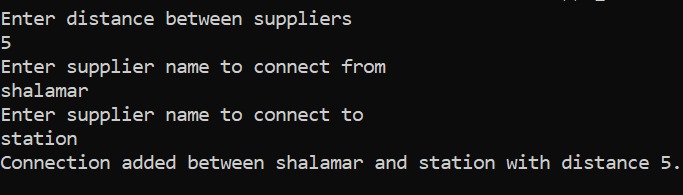
* **Add Supplier**

In this wireframe, users can add a supplier name, which corresponds to adding a new node to the graph. The graph represents supplier locations, with nodes indicating suppliers and edges showing distances. This feature helps expand and manage the supplier network efficiently.

****

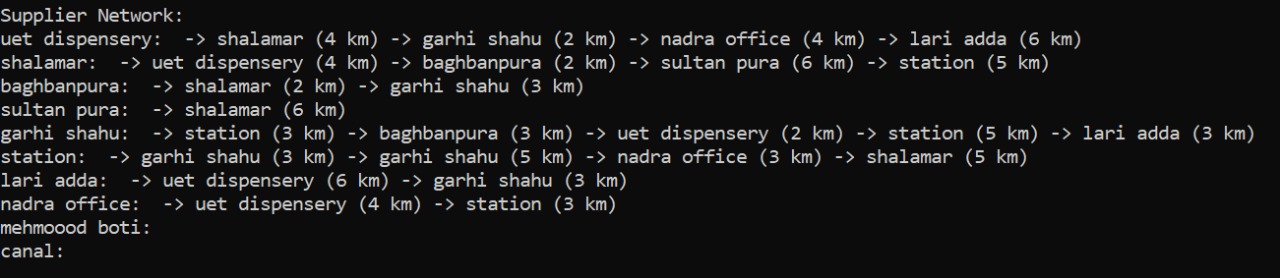
* **Connection Added between Supplier**

In this wireframe, connections between suppliers are added by creating edges with weights between nodes. The weights represent distances, enabling efficient management of supplier relationships and routes.

****

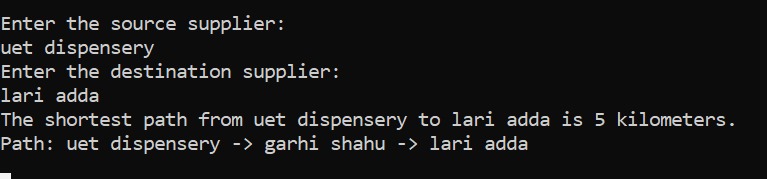
* **Supplier Network**

This wireframe visualizes the graph with nodes representing suppliers and edges indicating distances between them. It helps users understand the connections and distances in the supplier network. By using Dijkstra's algorithm, the shortest path between supplier locations can be found efficiently

****

* **Shortest path to supply medicine**

This wireframe calculates the shortest path to supply medicines from various supplier locations to the pharmacy. It implements Dijkstra's algorithm to determine the most efficient route based on distances. This ensures timely delivery and optimal logistics management.

****