Pharmacy Management System Using Java Swing and GUI

Omkar Gupta and Kushal Anikhindi Department of Electronics and Computer
K J Somaiya School of Engineering
Email: omkar.gupta@somaiya.edu and kushal.a@somaiya.edu

***Abstract*—The Pharmacy Management System project is designed to automate the essential tasks within a pharmacy, including managing medicine inventory, customer records, and processing orders. Using Java and Swing for GUI development, the system provides a streamlined solution aimed at small and medium-sized pharmacies. Testing showed the system's reliability and effectiveness in inventory and customer management. This paper covers the system's objectives, design, implementation, and key outcomes.**

Java, Pharmacy Management System, Inventory Management, Automation, GUI

I. Introduction

A. Problem Statement

Manual operations in pharmacies are prone to errors and can lead to inefficient management of inventory, customer details, and orders. This Pharmacy Management System addresses these issues by providing an automated, user-friendly platform.

B. Project Objective

The objective is to develop a software solution that:

- Manages medicine inventory, customer records, and orders.
- Provides a colorful, intuitive GUI for ease of use.
- Minimizes manual tasks and human error.

C. Scope

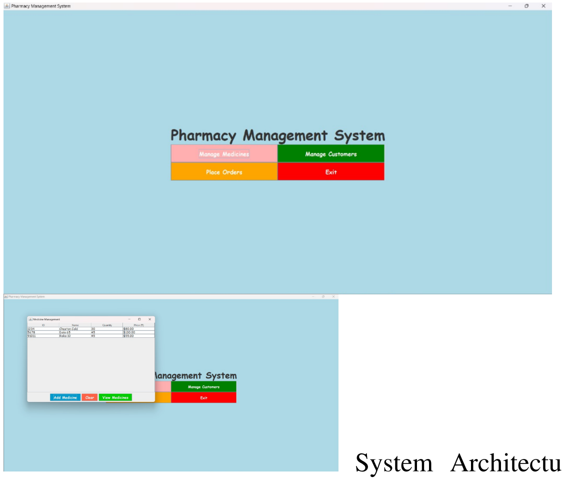
The system is designed as a standalone application with the scope limited to managing operations within small pharmacies. The prototype lacks multi-user support and cloud-based storage, which are suitable only for small to medium operations.

II. System Overview

A. System Architecture

The Pharmacy Management System is structured into the following core modules:

- **Medicine Management Module**: Manages stock details and updates inventory.
- **Customer Management Module**: Stores and displays customer information.
- **Order Management Module**: Tracks orders associated with customers and inventory.



ID	Name	Quantity	Price (₹)
1234	Cheston Cold	30	$80.00
5678	Dolo 65	45	$130.00
91011	Rako 10	45	$55.00

System Architecture Diagram

B. Technology Stack

The system uses:

- Java as the programming language.
- Java Swing for graphical user interface (GUI) development.
- Optional MySQL database with JDBC for backend storage.

III. Design and Implementation

A. Module Breakdown

Each module was implemented using object-oriented principles:

- **Medicine Management**: Uses JFrame, JTextField, and JButton for inventory management.
- **Customer Management**: Manages customer details with ArrayLists and JFrames.
- **Order Management**: Associates orders with customers and updates stock in real time.

B. Technical Depth and Key Features

The system utilizes core object-oriented programming (OOP) concepts:

- **Encapsulation**: Encapsulates data for medicines, customers, and orders in separate classes.
- **Inheritance and Polymorphism**: Ensures reusable and modular code.
- **GUI Design**: Includes Java Swing components for interactive windows and user input fields.

C. Challenges

Development challenges included:

- Integrating dynamic updates in GUI elements.
- Handling Java Swing exceptions.
- Connecting to MySQL database (optional) and managing SQL queries.

IV. Testing and Results

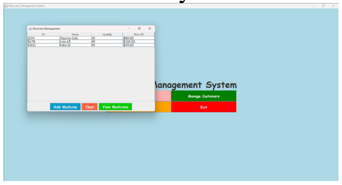
A. Test Cases

The following test cases were conducted:

- Adding and viewing medicines.
- Updating stock levels upon order placements.
- Managing customer records and searching by customer ID.

B. Results

Testing showed accurate results for stock updates, record management, and order processing. The following screenshot demonstrates the system in action:



ID	Name	Quantity	Price (₹)
1234	Cheston Cold	30	$80.00
5678	Dolo 65	45	$130.00
91011	Roko 10	45	$55.00

Sample Output of Pharmacy Management System

V. Conclusion and Future Work

A. Summary

The Pharmacy Management System meets its design objectives, providing an efficient and easy-to-use application for pharmacy management. The system significantly improves the accuracy of inventory management and streamlines customer order processing.

B. Future Enhancements

Potential enhancements include:

- Multi-user support with access control for different roles.
- Cloud integration for centralized data storage.
- Comprehensive reporting features to track sales and customer insights.

References

Java Swing Documentation, *Oracle Documentation*, available: https://docs.oracle.com/javase/tutorial/uiswing/. JDBC Documentation, available: https://dev.mysql.com/doc/connector-j/. J. Gosling, et al., ”The Java Language Specification,” 3rd ed., Addison-Wesley, 2005.