**PROPOSAL FOR PHARMACY MANAGEMENT SYSTEM**

**GROUP MEMBERS**

1. Vincent Ochieng’ SCT 211-0070/2022
2. Kiarie Zawadi Muthoni SCT 211-0462/2022
3. Matere Hans Walubengo SCT 211-0080/2022
4. Kaigu Ann Wangari SCT 211-0772/2022
5. Levi Mutugi SCT 211-0067/2022

**Introduction**

The purpose of this proposal is to outline the development of a comprehensive Pharmacy Management System (PMS) , tailored to meet the needs of modern pharmacies. With the increasing complexity of pharmaceutical operations and the growing demand for efficient management solutions, the implementation of a PMS is imperative for pharmacies to streamline operations, enhance customer service, and ensure compliance with regulatory standards.

**FUNCTIONAL REQUIREMENTS**

* Inventory Management:

Track inventory levels of medications, supplies, and products in real-time.

Generate automatic alerts for low stock levels and facilitate reordering processes.

Provide functionality for managing expiration dates and batch numbers.

* Prescription Management:

Capture and store prescription details, including patient information, prescribed medications, dosage instructions, and refills.

Allow pharmacists to verify prescriptions, dispense medications, and maintain a complete transaction history.

* Customer Management:

Maintain a centralized database of customer profiles, including contact information, prescription history, and insurance details.

Enable quick retrieval of customer records for efficient service delivery and personalized care.

* Reporting and Analytics:

Generate customizable reports on sales, inventory levels, prescription trends, and customer demographics.

Provide insights through analytics to identify opportunities for business growth and operational improvements.

**NON-FUNCTIONAL REQUIREMENTS**

* Security:

Implement robust data encryption protocols to safeguard sensitive information.

Ensure role-based access control to restrict unauthorized access to confidential data.

* Scalability:

Design the system to accommodate future growth and expansion of the pharmacy operations.

Allow for seamless integration with additional modules or functionalities as needed.

* Usability:

Develop an intuitive user interface with easy navigation and minimal training requirements.

Support multiple devices and platforms to enhance accessibility for users.

* Reliability:

Minimize system downtime through regular maintenance and backup procedures.

Implement error-handling mechanisms to prevent data loss or corruption.

Hardware and Software Requirements:

* Hardware:

Server: Minimum Intel Core i5 processor, 8GB RAM, 500GB HDD/SSD.

Workstations: Minimum Intel Core i3 processor, 4GB RAM, 250GB HDD/SSD.

Barcode scanners, printers, and POS terminals as required.

* Software:

Operating System: Windows Server for the server, Windows 10 for workstations.

Database Management System: Microsoft SQL Server for data storage.

Programming Language: Visual Basic (VB) for application development.

Integrated Development Environment (IDE): Visual Studio.

Version Control: Git for collaborative development.

**SCHEDULE/PROJECT TIMELINE**

**Week 1-2 (February 11 – February 16)**

Requirement Analysis: Gather detailed requirements from stakeholders and finalize the scope of the project.

Research and Selection: Evaluate potential hardware and software solutions based on functional and non-functional requirements.

**Week 3-4 (February 17 - February 25)**

Design and Architecture: Develop system architecture, database schema, and user interface mockups.

Procurement: Purchase necessary hardware and software licenses.

**Week 5-6 (February 26 - March 15)**

Development: Implement core functionalities of the Pharmacy Management System using Visual Basic (VB).Testing: Conduct rigorous testing to ensure reliability, security, and usability of the system.

**Week 7 (March 16 - March 21)**

Deployment: Install and configure the Pharmacy Management System on the server and workstations.

Training: Provide training sessions for pharmacy staff to familiarize them with the new system.

Go-Live: Transition from existing processes to the new Pharmacy Management System.

**PROGRAM DESIGN**

The software architecture will be structured to ensure scalability, modularity, and maintainability. It will follow a layered architecture approach, which involves separating the system into distinct layers, each responsible for specific functionalities.

**Presentation Layer**

The Presentation Layer will handle user interaction and interface rendering.

It will consist of forms, user controls, and UI components that facilitate user interaction.

VB.NET's WinForms or WPF (Windows Presentation Foundation) can be utilized for developing the presentation layer.

User-friendly interfaces will be designed to provide intuitive navigation and efficient data input.

This layer will interact with the Business Logic Layer to retrieve and display information to the user and process user inputs.

**Business Logic Layer**

The Business Logic Layer will implement the core functionalities of the Pharmacy Management System.

It will encapsulate business rules and logic related to inventory management, prescription processing, sales tracking, etc.

VB.NET will be used to write the code for the business logic layer, implementing algorithms and workflows required for various processes.

This layer will be designed to be independent of the presentation layer, facilitating easier testing, maintenance, and potential future changes.

It will provide services or APIs that the presentation layer can call to perform specific operations, maintaining separation of concerns.

**Data Access Layer**

The Data Access Layer will manage interactions with the database, including data retrieval, storage, and update operations.

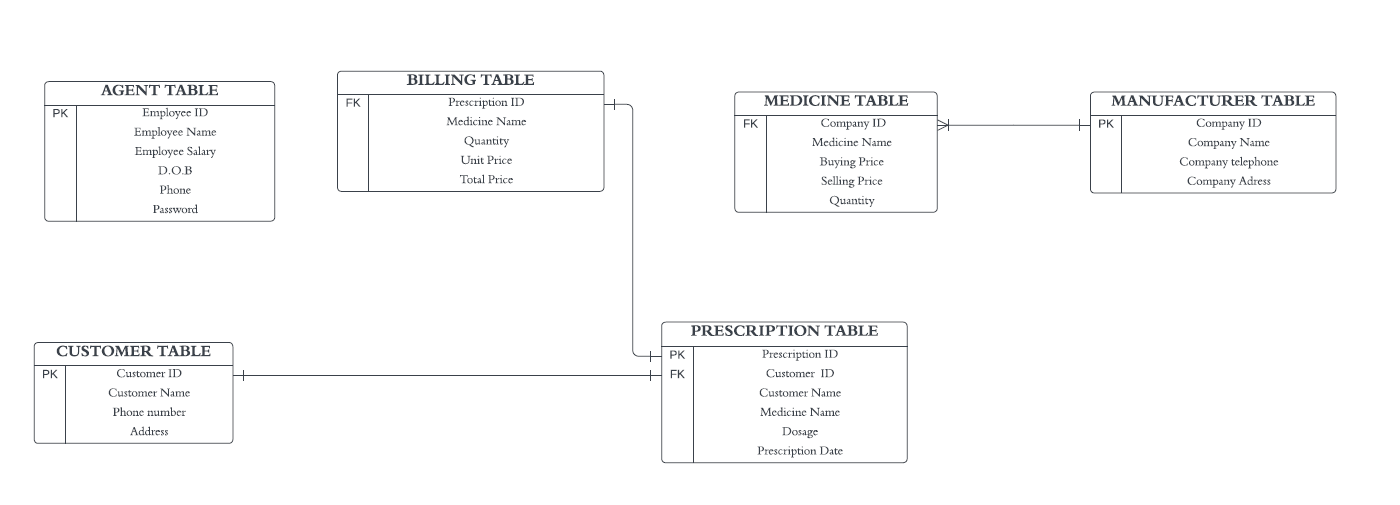
It abstracts the underlying database structure and operations, promoting flexibility and scalability.

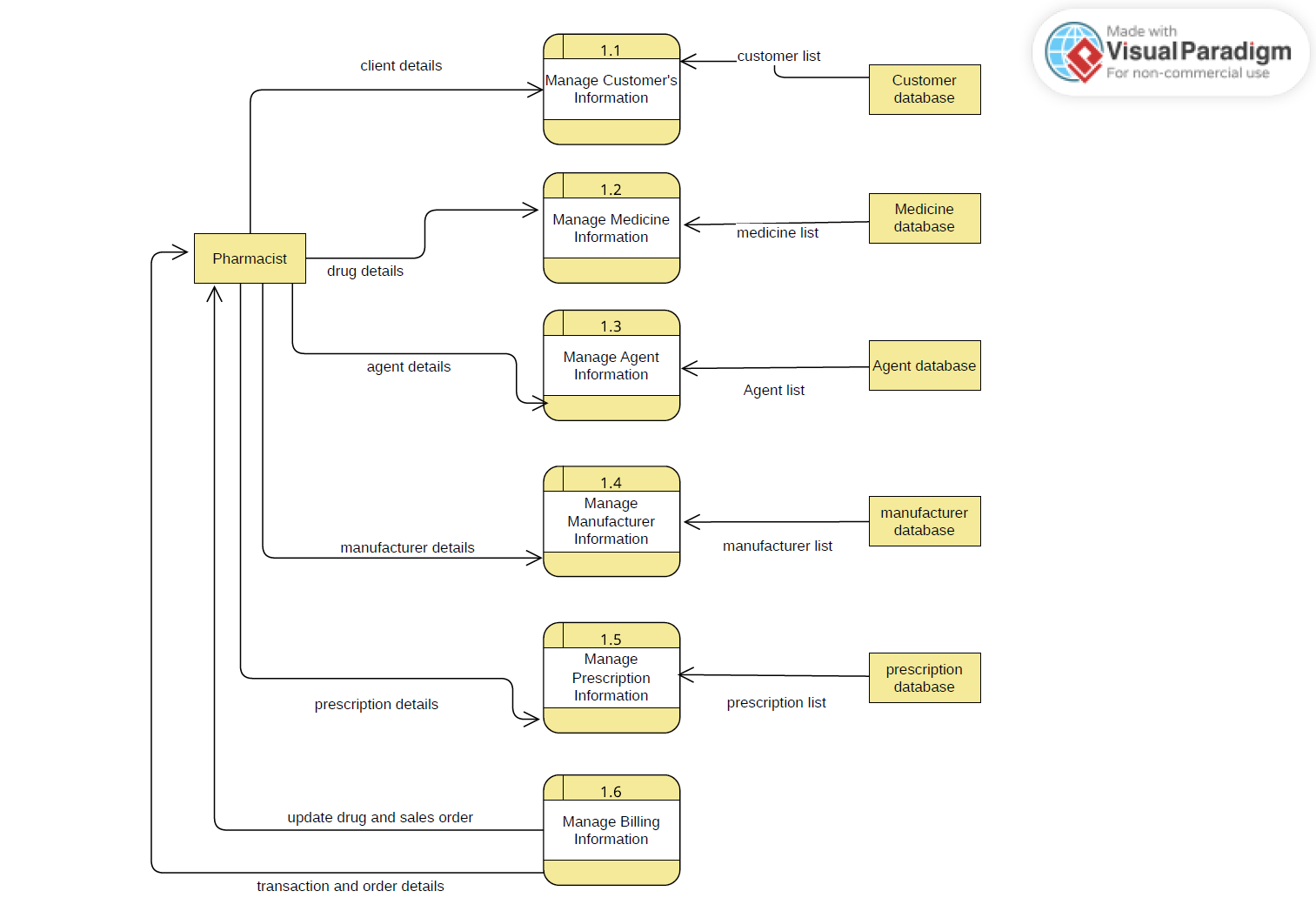
ADO.NET (ActiveX Data Objects .NET) will be used for database access in VB.NET applications.

The Data Access Layer will include classes and methods for connecting to the database, executing queries, and processing results.

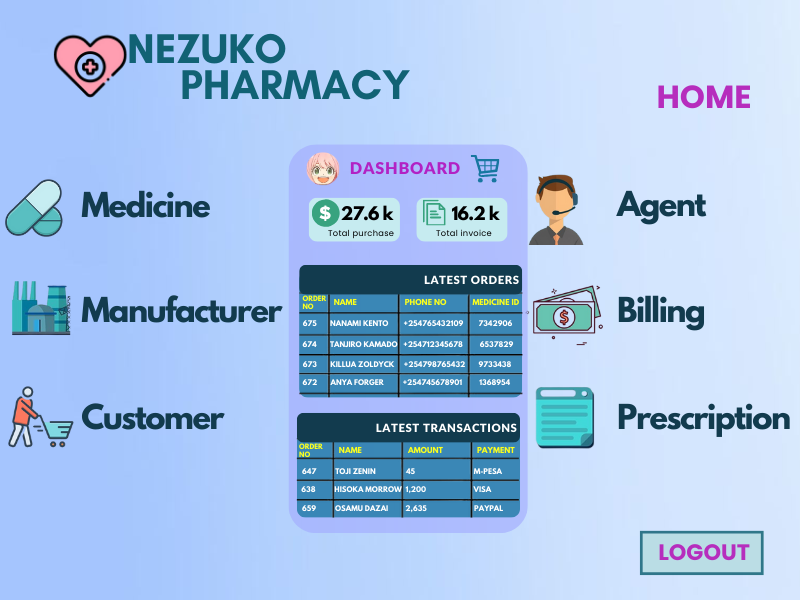
It will handle tasks such as CRUD (Create, Read, Update, Delete) operations on database entities related to inventory, prescriptions, sales, etc.

**DATABASE DESIGN**

****

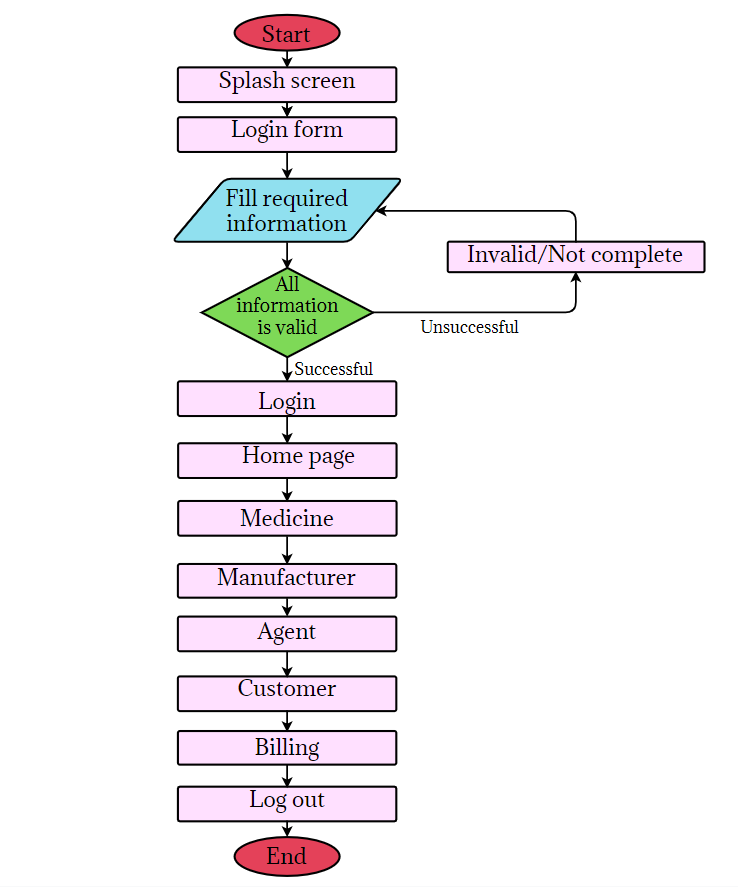
** DATA FLOW DIAGRAM**

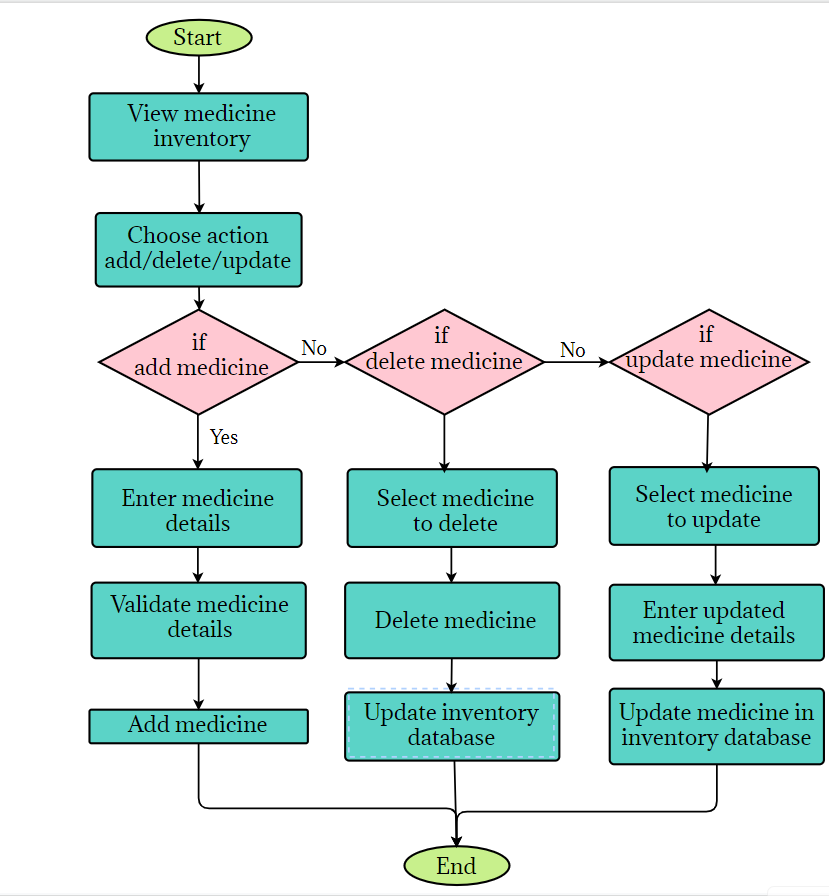
**UI/UX DESIGN**

****

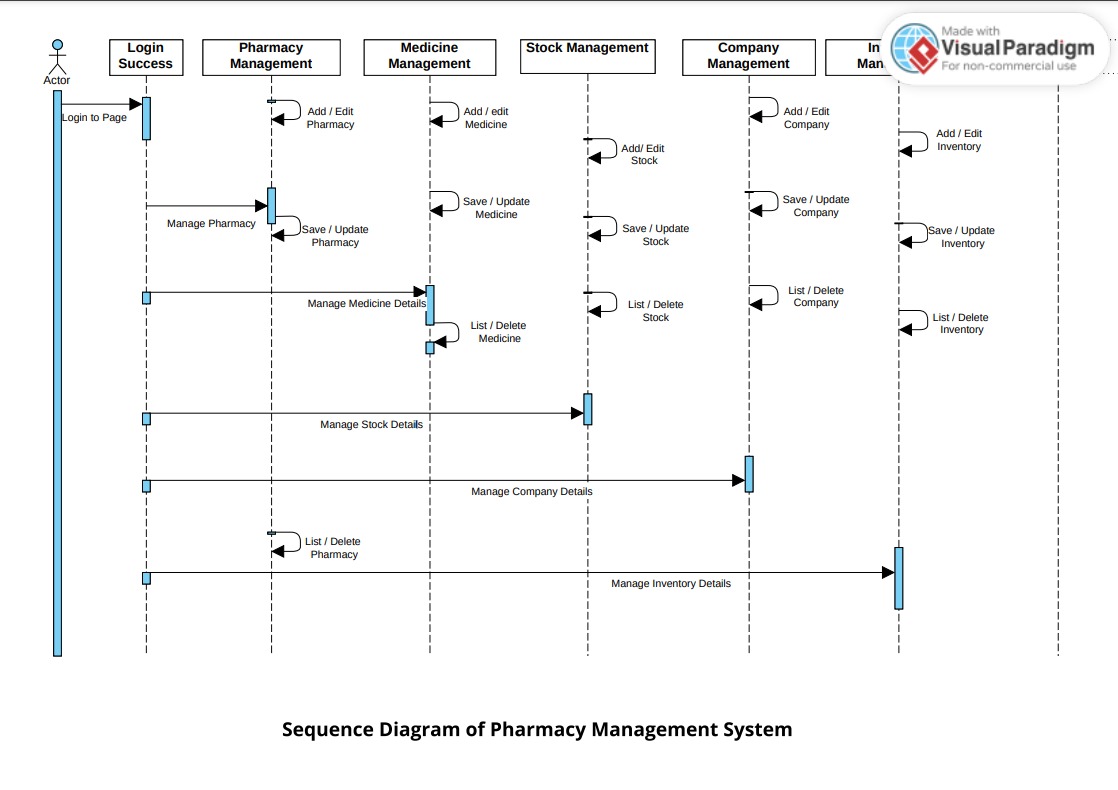
**FLOW CHARTS**

**System flow chart**

****

**Medicine Management**

**SEQUENCE DIAGRAM**

****