**eMed: A Web and Mobile-based Application for RXCue Pharmacy – A Digital Marketing Approach**

An

Application Development Project

Presented to the Faculty of

**Mindoro State University Bongabong Campus**

Masipit, Calapan City

Oriental Mindoro

In Partial Fulfillment

of the Requirements for the Degree

Bachelor of Science in Information Technology

by:

**Cabrera, Wendell**

**Gargullo, Cydie, S.**

**Tupas, Mark Jerlan**

**BSIT III-F4**

NOVEMBER 2023

**CHAPTER I**

**INTRODUCTION**

**Project Context**

Over the past several years, many pharmacies have found it difficult to retain staff and pay competitive salaries, resulting in staff shortages in the pharmacy and nursing sections. At the same time, the state and federal governments, the Institute of Safe Medication Practices (ISMP), and the JCAHO are emphasizing the need to reduce medication errors (Kimble & Chandra, 2001). Fortunately, advancements in technology provide healthcare professionals with the tools to achieve this mandate for a safer health delivery system.

Currently, a substantial shortage of pharmacists exists. Hospitals treat sick patients with complex drug therapies, while nursing homes and assisted living facilities face high drug use. The reports of federal Health Resources and Services Administration encompass an outstripped supply and medication shortages. The demand for pharmacy services is growing for specialized needs (Knapp, 2002), therefore, as developers, we made the decision to integrate statistical analysis and predictive functionality into the administrative interface. This enhancement enables us to monitor the distribution of clients based on their addresses, making it simpler to identify optimal locations for new branch openings. Additionally, this empowers us to forecast the demand for medicines over the next three years, aiding in better inventory management and resource allocation.

Other studies describe technology-based pharmacy as a user-friendly resource that can enhance the efficient management of pharmaceutical inventories within pharmacy operations (Verano\_ Revitalizing Health Service with E-Pharmacy\_, n.d.).

RXCUE Pharmacy, located in Calero, Calapan City, Oriental Mindoro, faces various manual problems in its traditional pharmacy operation. These issues encompass prescription filling errors, inventory tracking inefficiencies, sales and billing inaccuracies, and the potential for regulatory non-compliance due to outdated record-keeping methods. Moreover, relying on face-to-face communication with clients and suppliers can limit accessibility and responsiveness, while also impeding data analysis and reporting.

Within the context of our project, it is crucial to recognize that healthcare services stand as a cornerstone of human well-being. They constitute an indispensable requirement for individuals to engage in their daily activities with efficiency and vitality. This understanding forms the backdrop against which we delve into our project's objectives and significance.

To address these challenges, the researchers decided to develop an "eMed: A web and mobile based application for RXCue Pharmacy - A Digital Marketing Approach ", which can improve customer satisfaction, and enable better services while keeping up with evolving industry standards.

**Objectives of the Study**

Generally, this study aims to develop a system entitled "eMed: A web and mobile based application for RXCue Pharmacy - A Digital Marketing Approach ", that seamlessly integrates web, desktop, and mobile platforms to enhance medication management, facilitate customer transactions, improve branch expansion decisions, and ensure patient convenience.

Specifically, it aimed to

1. Design and deploy a user-friendly pharmacy system accessible through web browsers (Web), providing administrative and cashier functions via desktop (Admin/cashier/branch), and a client interface via mobile devices (Mobile).
2. Develop an automated notification system for medication expiration, allowing pharmacy staff to proactively manage and restock expired medicines to ensure patient safety.
3. Enable online browsing of available medicines with the option to reserve for walk-in pick-up, integrating an appointment number system to expedite the in-store transaction process.

**Scope and Limitations**

The scope of this project is to develop and implement a web and mobile-based application for RXCUE Pharmacy with a focus on integrating digital marketing strategies in Calapan City, Oriental Mindoro. The study will encompass the design, development, and deployment of the system, targeting improved efficiency in pharmacy operations. There are three users in the system; the admin, cashier, and the users. The admin can add new pharmacy branches and manage users for each branch including location details and contact information. On the other hand, cashier can check the availability of medicines in real-time, providing accurate information to clients, and can also process online purchases for clients, maintaining seamless operations between online and in-store transactions. Lastly, users can access an e-commerce feature, allowing them to browse and purchase medicines online. The system will be accessible in desktop, web, and mobile devices and will be available online so that its users can easily access it. The research will cover a six-month implementation period, with ongoing monitoring and evaluation.

**Definition of Terms**

In order to further explain the study in an explicit way, the researchers defined the words operationally as:

* **PHP** - PHP is an open-source server-side scripting language that many developers use for web development. It is also a general-purpose language that you can use to make lots of projects, including Graphical User Interfaces (GUIs) (Chris, 2021).
* **SMS** – In this study, it is a process where SMS was use as notification to notify the users for website promotions and updates as well as the availability of the medicine.
* **eMed** – In this study, eMed is the name of the web and mobile application. It serves as a specific identifier for the developer’s project.
* **Web and Mobile based Application** – In this study, web and mobile based application is a software that operates on both web browsers and mobile devices, allowing users to access its features and functions and cater a diverse range of users and their preferred platforms.
* **RXCue Pharmacy** – In this study, RXCue Pharmacy serves as the client organization, and the study's primary focus is directed toward addressing their needs and requirements.
* **Digital Marketing** - In this study, digital marketing refers to the use of online strategies to engage clients of RXCue Pharmacy for promotions and important announcements.

**CHAPTER II**

**REQUIREMENT SPECIFICATION**

This chapter explained the specific requirements and functionalities necessary for the effective completion of our project.

**Hardware Requirements**

Hardware Requirements refer to the representation of the hardware used by the system. Table 1 below presents the hardware requirements to be used by the project.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Hardware** | **Functions** | **Specifications** | | **Unit** |
| **Minimum** | **Recommended** |
| Computer/  Processor | It is used to handle the complex tasks associated with creating a strong, secure, and reliable system to provide us with the computing power we need to ensure our system is optimized for performance and reliability. | Intel  (R) Core  (TM) i3-7020U CPU | i3 and above of version | 1 |
| RAM | It acts as a temporary memory storage for the system. It enables the computer to quickly access the website's data and process it efficiently, providing a faster and smoother browsing experience. | 4GB | 6GB | 2 |
| Android Smartphone | It is used if the customer wants to access the system conveniently and for easy order the products they want. |  |  | 3 |

**Table 1: Hardware Requirements**

**Software Requirements**

Software Specifications refer to the representation of the software used by the system. Table 2 below presents the software specifications to be used by the project.

|  |  |
| --- | --- |
| Software Used | Description |
| Operating System | The researchers chose the Windows 10 operating system because it will suit and can support the system. |
| Visual Studio Code | Visual Studio Code is a code editor that supports development operations like debugging, task running, and version control. |
| Laravel | Laravel is a free and open-source PHP framework that provides a set of tools and resources to build modern PHP applications. |
| phpMyAdmin | phpMyAdmin is a web-based tool for managing MySQL databases. |
| Laragon | Laragon is a web development environment for Windows. |
| GitHub | GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere. This tutorial teaches you GitHub essentials like repositories, branches, commits, and pull requests. |

**Table 2: Software Requirements**

**Functional Requirements**

Functional requirements specify what the system can do. This will serve as the manual of the system that includes the functional specifications, the process of the system to create reports and the process of data that will be manipulated inside the system.

|  |  |
| --- | --- |
| **ID NO** | **Requirement Description** |
| **Admin Side** | |
| 1.1 | The system should allow the admin to add new pharmacy branches, including location details and contact information. |
| 1.2 | The admin should be able to add and manage users for each branch, assigning roles and permissions as needed. |
| 1.3 | The system should provide the admin with the capability to manage the inventory for each branch, including adding, updating, and tracking medicines and related items. |
| 1.4 | Admin should have access to statistical data, represented in pie graphs, to easily view the percentage of clients in different areas. This aids in data-driven decisions for new branch locations. |
| 1.5 | The system should offer predictive analytics to forecast pharmaceutical needs and trends for the upcoming three years. |
| 1.6 | Admin should be able to conduct digital marketing campaigns, sending SMS, push notifications, and email updates to clients for promotions and important announcements. |
| 1.7 | The system should generate notifications for medicines approaching expiration, following the First-In-First-Out (FIFO) method to ensure medication safety. |
| **Cashier Side** | |
| 1.8 | Cashiers should be able to check the availability of medicines in real-time, providing accurate information to clients. |
| 1.9 | The system should support sales management, allowing cashiers to process transactions efficiently. |
| 2.0 | Cashiers should have access to inventory management features, ensuring the accurate tracking of available medicines and related products. |
| 2.1 | Cashiers can process online purchases for clients, maintaining seamless operations between online and in-store transactions. |
| 2.2 | Cashier can assign appointment number transaction to customer in order to help in organizing and managing customer flow within the premises. |
| 2.3 | Cashier should be able to void cancellation and view the history of product cancelled by the user. |
| **User Side** | |
| 2.4 | Clients should have access to an e-commerce feature, allowing them to browse and purchase medicines online. |
| 2.5 | Clients can schedule appointments through the system, receiving appointment numbers to facilitate quick and convenient in-store transactions for pick-up and payment. |
| 2.6 | Clients should be able to cancel orders and view the cancelled product history. |

**Table 3: Functional Requirements**

**Non-Functional Requirements**

These are requirements that pertain to behavior properties that a system must have. It defines how a system is supposed to be or its system properties. It contains the following:

**Operational Requirement**

Table 4 represents the requirement description that will specify the operating environment(s) in which the system must perform and how these might change over time.

|  |  |
| --- | --- |
| ID No. | Requirement Description |
| 1.1 | The system is compatible with any operating system |
| 1.2 | The system should have an intuitive and user-friendly interface to ensure ease of use for applicants of varying technical expertise. |
| 1.3 | The user interfaces across web, desktop, and mobile platforms should be intuitive, ensuring ease of use for both clients and staff. |
| 1.4 | The system should be accessible 24/7 to accommodate patients who need medicine. |

**Table 4: Operational Requirement**

**Performance Requirement**

Table 4 represents the requirement description that will emphasize the response time, capacity, and reliability of the system.

|  |  |
| --- | --- |
| ID No. | Requirement Description |
| 1.5 | Each web page must load within 2 seconds |
| 1.6 | The user can track their orders 98% of the time without failure. |
| 1.7 | Any user can order and buy the products in the system. |
| 1.8 | The system should respond to user inquiry, such as what medicines are available. |

**Table 5: Performance Requirement**

**Security Requirement**

Table 5 represents the requirement description that will address issues with security, such as who has access to the system's data and must have the ability to protect data from disruption or data loss.

|  |  |
| --- | --- |
| ID No. | Requirement Description |
| 1.9 | The user’s information must be encrypted |
| 2.0 | Only the admin can view the inventory and sales statistics. |
| 2.1 | The user’s must be logged in before they can order or can access the homepage of eMed. |
| 2.2 | Database access should be restricted to authorized personnel only. |
| 2.3 | The system must implement strong authentication mechanisms, such as username/password combinations to ensure that only authorized personnel can access the system. |

**Table 6: Security Requirement**

**Chapter III**

**DESIGN AND DEVELOPMENT METHODOLOGIES**

**System Design**

The RXcue Pharmacy was developed through an inclusive plan to serve as the foundation of our implementation of the system. By providing an easy access through the web-app. This way patient can interact with the RXcue Pharmacy to appoint their schedule mainly, all the transaction will happen virtually. The system will be hosted on a website that requires an internet connection for both administrator, staff, and patient to execute the task. All the information will be provided by the patient to register and create account, to access the system.

**Database Design**

The database design in RXcue Pharmacy system allows users to track pharmacies, medicine, order details and schedule. Consider using a SQL query to gather data like the given information It can regularly update the inventory, pharmacies, and order history.

The table shows the fields, data types and a description.

Data are collected and validated for information that linked and related for data integrity.

The developers use MySql for DBMS, for the database design. This is database design that will implement for RXcue Pharmacy

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Size** | **Default** | **Description** |
| Id | Int | 11 | Not Null | Patient ID |
| First name | Varchar | 50 | Not Null | User Name |
| Last name | Varchar | 50 | Not Null | User Name |
| Middle name | Varchar | 50 | Not Null | User Name |
| Age | Int | 10 | Not Null | User Age |
| Gender | Text | 10 | Not Null | User Gender |
| Address | Varchar | 50 | Not Null | User Address |
| Role | Varchar | 50 | Not Null | User type |
| Password | Text | 20 | Not Null | User password |

**Table 7. Fields for User**

The table 7 show the field name, data type, size, default, and description in user table where ID is the primary key.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Size** | **Default** | **Description** |
| Id | Int | 11 | Not Null | Item id |
| Item Name | Text | 100 | Not null | Item name |
| Previous quantity | Int | 100 | Not null | Quantity |
| Quantity change | Int | 100 | Not null | Quantity |
| New quantity | Int | 100 | Not null | quantity |
| Change date | Date time |  | Not null | Date created |
| Created at | Date time |  | current timestamp | Time created |
| Updated at | Date time |  | current timestamp | Time updated |

**Table 8. Fields for Inventory**

The table 8 show the field name, data type, size, default, and description in inventory table where ID is the primary key.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Size** | **Default** | **Description** |
| Id | Int | 11 | Not null | Branch id |
| Branch name | Varchar | 100 | Not null | Branch name |
| Location | Float | 100 | Not null | Branch location |
| Contact | Int | 11 | Not null | Contact number |
| Status | Varchar | 100 | Not null | Branch status |
| Created at | Date time |  | current timestamp | Time created |
| Updated at | Date time |  | current timestamp | Time created |

**Table 9. Fields for Branch**

The table 9 show the field name, data type, size, default, and description in branch table where ID is the primary key.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Size** | **Default** | **Description** |
| Id | Int | 11 | Not null | Admin id |
| Name | Text | 50 | Not null | User name |
| Username | Text | 50 | Not null | User name |
| Email | Text | 50 | Not null | User email |
| Password | Text | 50 | Not null | User password |
| Created at | Date time |  | Current time stamp | Time created |
| Updated at | Date time |  | Current time stamp | Time updated |

**Table 10. Fields for Admin**

The table 9 show the field name, data type, size, default, and description in admin table where ID is the primary key.

**Architectural Diagram/ Block Diagram**

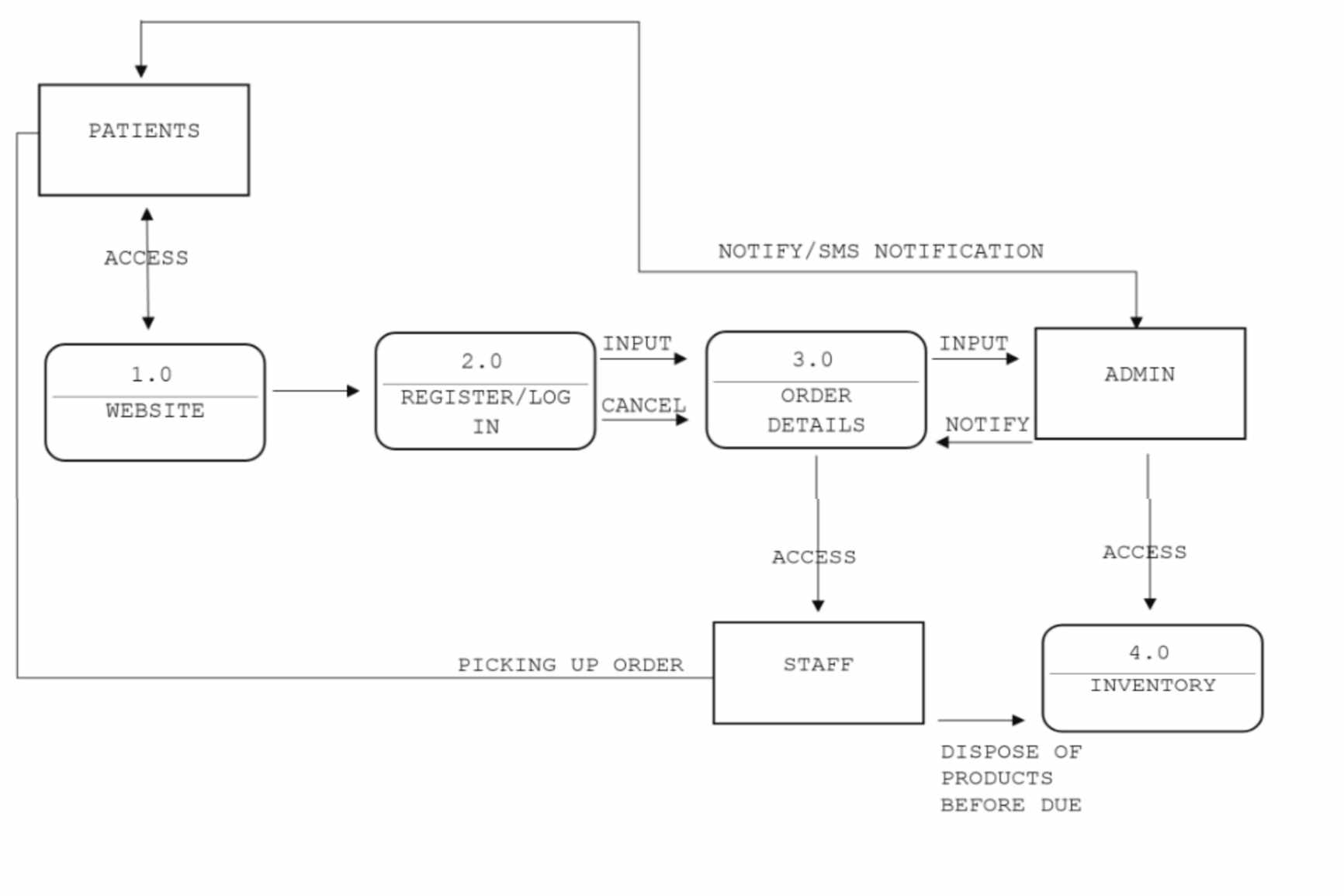
A diagram of a computer system

Description automatically generatedFigure 1 below shows the system architecture of the RXcue Pharmacy System. It was designed on how the system will work moderately with its functionalities.

**Figure 1. System Architecture of RXcue Pharmacy System**

The users of the system are the clients/patients, admin, and staff. It is web-based system application that will be implemented, the patients will be accessing the system through the internet. The system will fetch all the information, the server will manage all database queries and web transactions.

**Data Flow Diagram Level 0**

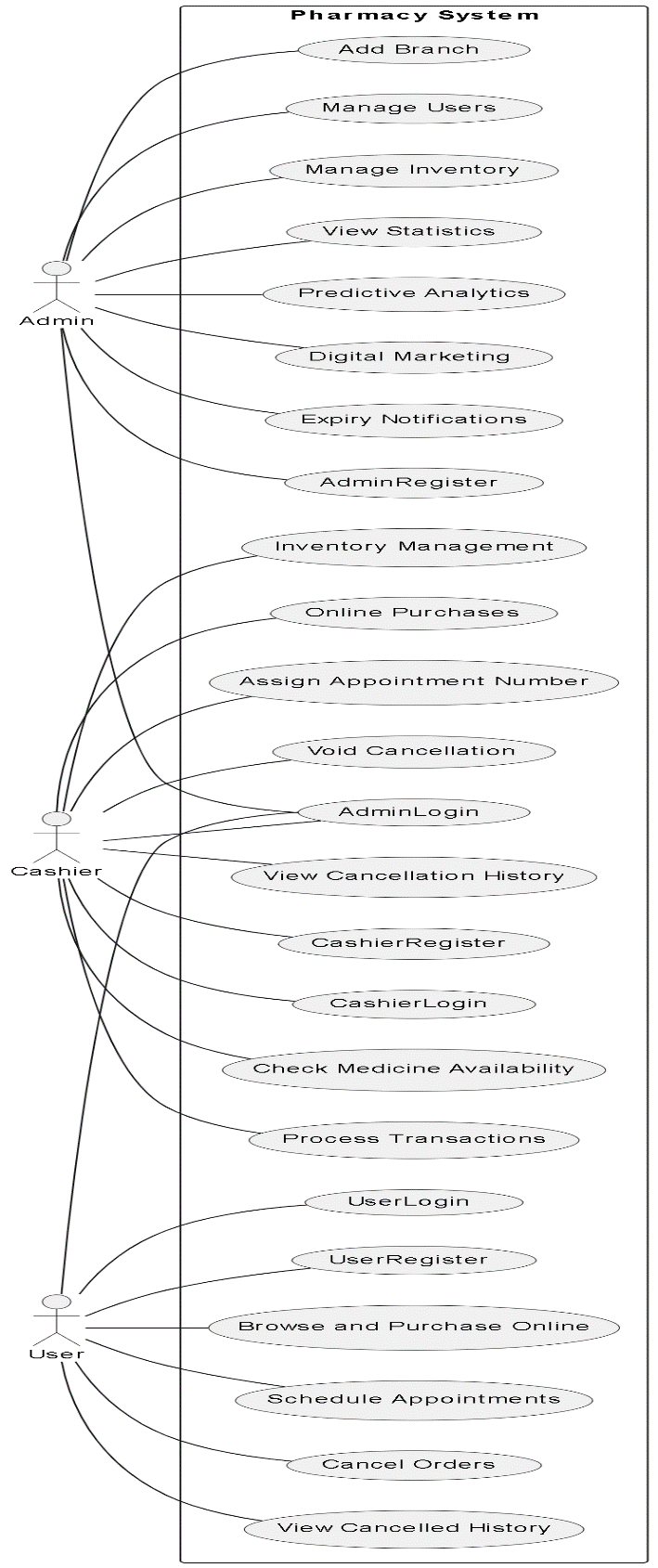
 This section shows the representation of data flows within the system. It illustrates the information between entities, process, and its destination.

**Figure 2. Data Flow Diagram Level 0**

Figure 2 diagram shows the sequences of the system for each entity. It indicates the flow of the system functionalities to fill the role for admin, staff, and patients. Admin can access the information, order details, and inventory, he/she can also send notifications and update/retrieve data. For the patient who is responsible for filling the information by logging in he/she can access the website and place the order, he/she can also receive notification from the system. For the staff he/she can only access inventory and order details of patient, by means staff is responsible for the dispose of the medicine that are three months before the expiration date.

**UML Use-case Diagram**

In this section, the use case diagram summarizes the high-level functions and a graphic representation that depicts the relationship between the system, admin, cashier, and users, it is how the system works within interactions of the actors. The diagram will help the researchers to identify and organize the functionality of the entire project.

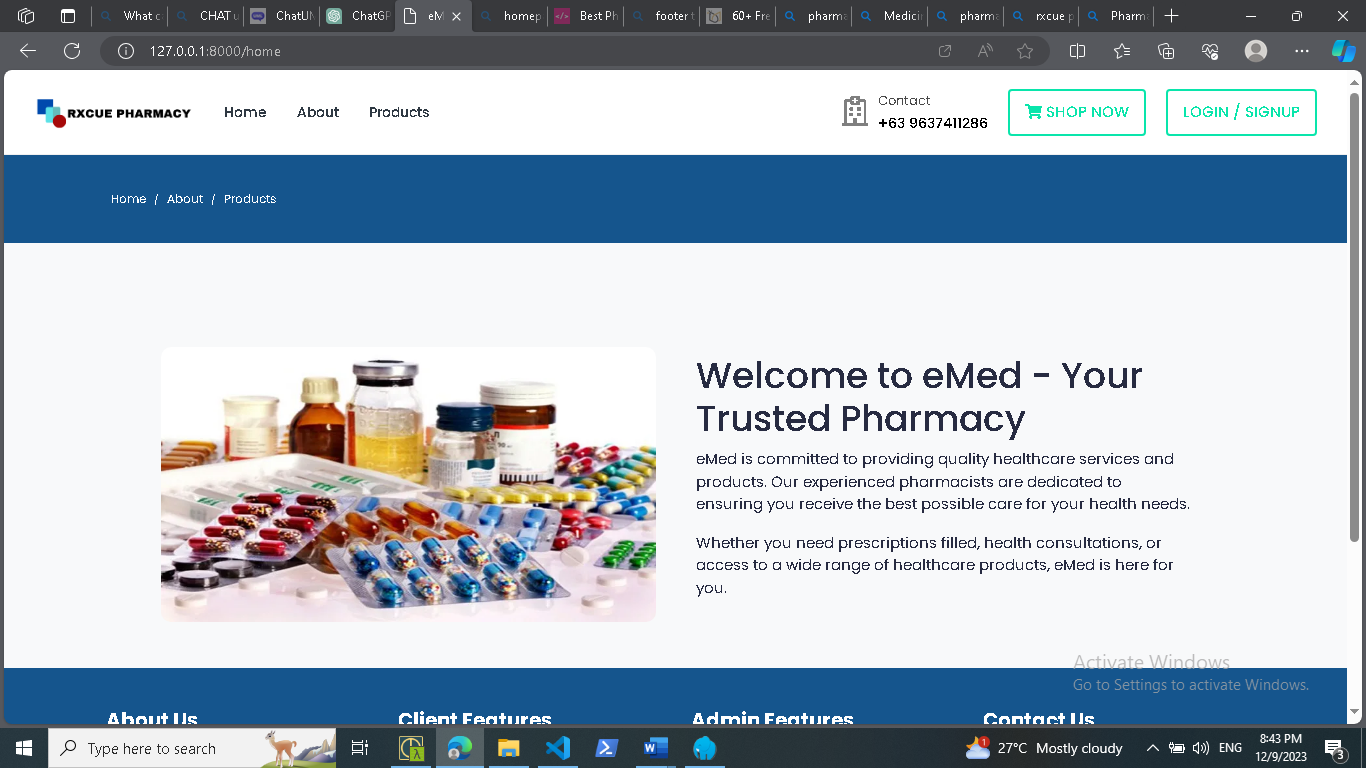
****

**Figure 3. UML Use-case Diagram**

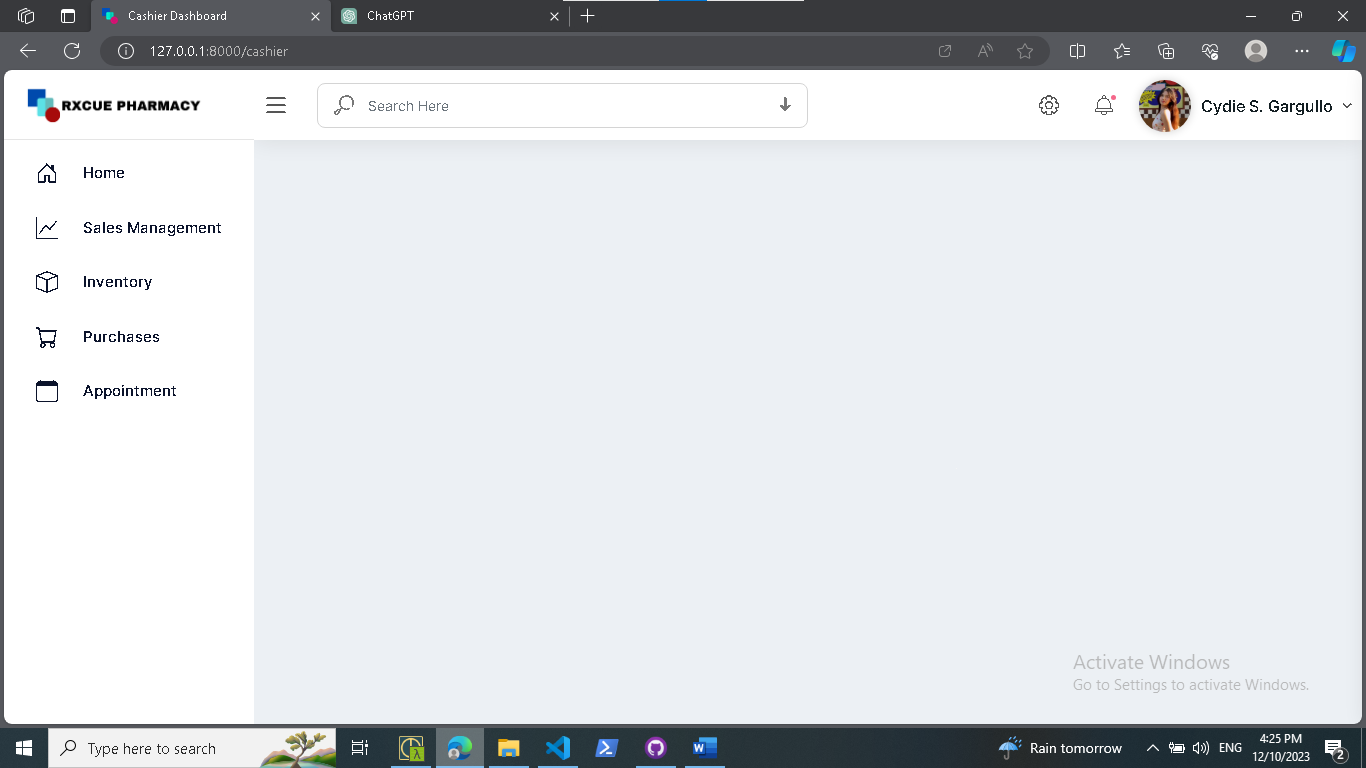
Figure 3 shows the roles of the Administrator and the customer to be executed in the whole process of the system.

**Sample Mock-up**

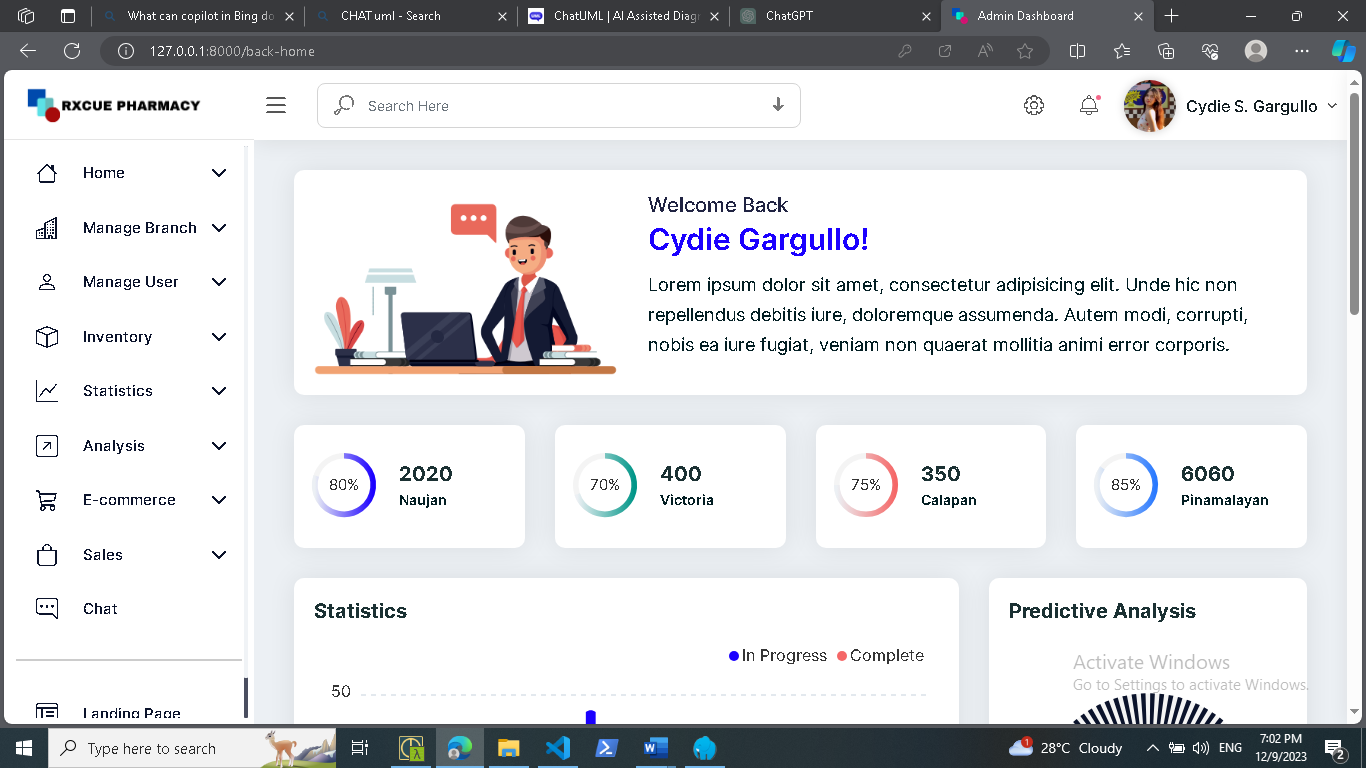
A sample mock-up is a visual representation of a website after it is built. It consists of visuals that show how the website should look and its function. It is used to refine the design, identify potential problems, and ensure that the system meets the user's needs and expectations. Below are the system users and admin interface.

****

**Figure 4. User Interface**

****

**Figure 5. Cashier Interface**

****

**Figure 6. Admin Interface**

**Development Method**

The “eMed: A web and mobile based application for RXCue Pharmacy with SMS Notification” will employ the System Development Life Cycle (SDLC) to ensure the production of an efficient and high-quality system. This project will use an Iterative model, which allows for improvements to be made throughout the development process.

The SDLC is divided into four phases: collecting requirements, design and development, testing, and implementation. These steps will be followed in order to create a successful system.

**Requirements Gathering**

**Testing**

**Implementation**

**Design & Development**

**Testing**

**Implementation**

**Design & Development**

Build 1

Build 2

**Testing**

**Implementation**

**Design & Development**

Build 3

**Planning**

**Maintenance**

**Figure 6. SDLC Iterative Model**

The research followed the following phase.

1. **Planning.** In this phase, researchers conduct interviews and observations according to the problem that needed to be resolved. Researchers developed the objectives and plans of the project that serves as the foundation to produce the desired output of the system at the end of the study.
2. **Requirements Gathering.** In this phase, researchers will gather more information and data requirements needed for the development of the system. Also, determine the functional and non-functional requirements of the project.
3. **Design.** During this phase, the architecture for the entire system is designed, taking into account the user interfaces, database structures, and the overall system flow. The design is modular to support the integration of web, desktop, and mobile components.
4. **Development.** During this phase, researchers will start the coding process for the functionality of the system through Visual Studio Text Editor Application and PhpMyAdmin as administrator tool for the database, wherein database will serve as data storage of the system and CodeIgniter 4 as the framework of the proposed system.
5. **Testing.** In this phase, researchers will execute the pre-deployment of the system for trial and error just to see if the device is properly functioning.
6. **Implementation.** In this phase, researchers will create the iteration of the project after the testing process in order to analyze and improve the design and functionality which needs more improvements to meet the project objectives.
7. **Maintenance.** In this phase, researchers must maintain the functionality of the system during deployment to maintain its performance. Also, researchers will determine and fix the errors that may occur over a period of time.

**Gantt Chart**

In this section, Gantt Chart is presented to show the plans and schedules of the project timeline. All the development stages up to the completion of the project were documented in this chart. This helps the researchers to know the deadlines needed to accomplish and show breakthroughs in various tasks.

**Table 11. Gantt Chart**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Task Name** | **Task Date** | | | | | | | | | | | | | | | | | | | |
| Oct | | | | Nov | | | | Dec | | | | Jan | | | | Feb | | | |
| Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 |
| **1.Planning** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.1 Conduct an interview |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.2 Define project objectives |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.3 Define project plan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.4 Approval of project plan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **2.Requirements Gathering** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.1 Data Collection |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.2 Functional |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.3 Non-Functional |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **3.Design** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.1 Frontend software design |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **4.Development** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.1 Back-end coding |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **5.Testing** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5.1 Functionality testing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5.2 User interface testing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **6.Implementation** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **7.Maintenance** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7.1 Project monitoring |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7.2 Resolve system errors |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**Table 12. Gargullo Gantt Chart**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Task Name** | **Task Date** | | | | | | | | | | | | | | | | | | | |
| Oct | | | | Nov | | | | Dec | | | | Jan | | | | Feb | | | |
| Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 |
| **1.Planning** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.1 Conduct an interview |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.2 Define project objectives |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.3 Define project plan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.4 Approval of project plan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **2.Requirements Gathering** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.1 Data Collection |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.2 Functional |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.3 Non-Functional |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **3.Design** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.1 Frontend software design |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **4.Development** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.1 Back-end coding |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **5.Testing** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5.1 Functionality testing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5.2 User interface testing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **6.Implementation** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **7.Maintenance** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7.1 Project monitoring |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7.2 Resolve system errors |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**Table 13. Cabrera Gantt Chart**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Task Name** | **Task Date** | | | | | | | | | | | | | | | | | | | |
| Oct | | | | Nov | | | | Dec | | | | Jan | | | | Feb | | | |
| Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 |
| **1.Planning** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.1 Conduct an interview |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.2 Define project objectives |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.3 Define project plan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.4 Approval of project plan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **2.Requirements Gathering** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.1 Data Collection |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.2 Functional |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.3 Non-Functional |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **3.Design** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.1 Frontend software design |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **4.Development** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.1 Back-end coding |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **5.Testing** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5.1 Functionality testing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5.2 User interface testing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **6.Implementation** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **7.Maintenance** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7.1 Project monitoring |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7.2 Resolve system errors |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**Table 14. Tupas Gantt Chart**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Task Name** | **Task Date** | | | | | | | | | | | | | | | | | | | |
| Oct | | | | Nov | | | | Dec | | | | Jan | | | | Feb | | | |
| Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 | Week 1 | Week 2 | Week 3 | Week 4 |
| **1.Planning** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.1 Conduct an interview |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.2 Define project objectives |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.3 Define project plan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1.4 Approval of project plan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **2.Requirements Gathering** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.1 Data Collection |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.2 Functional |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.3 Non-Functional |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **3.Design** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.1 Frontend software design |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **4.Development** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4.1 Back-end coding |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **5.Testing** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5.1 Functionality testing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5.2 User interface testing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **6.Implementation** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **7.Maintenance** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7.1 Project monitoring |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7.2 Resolve system errors |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**Legend:** - Completed/ Done

Table 11 shows the whole process of developing eMed. It displayed the various tasks and marks as completed oats certain date. Researchers will be kept informed of the progress of the development which will help them not to miss out on steps and differentiate tasks from the amount of time took to complete them.

Table 12, 13, 14 shows the individual process of developing eMed. It displayed the various tasks as completed by an individual in the group.

**Testing and Evaluation**

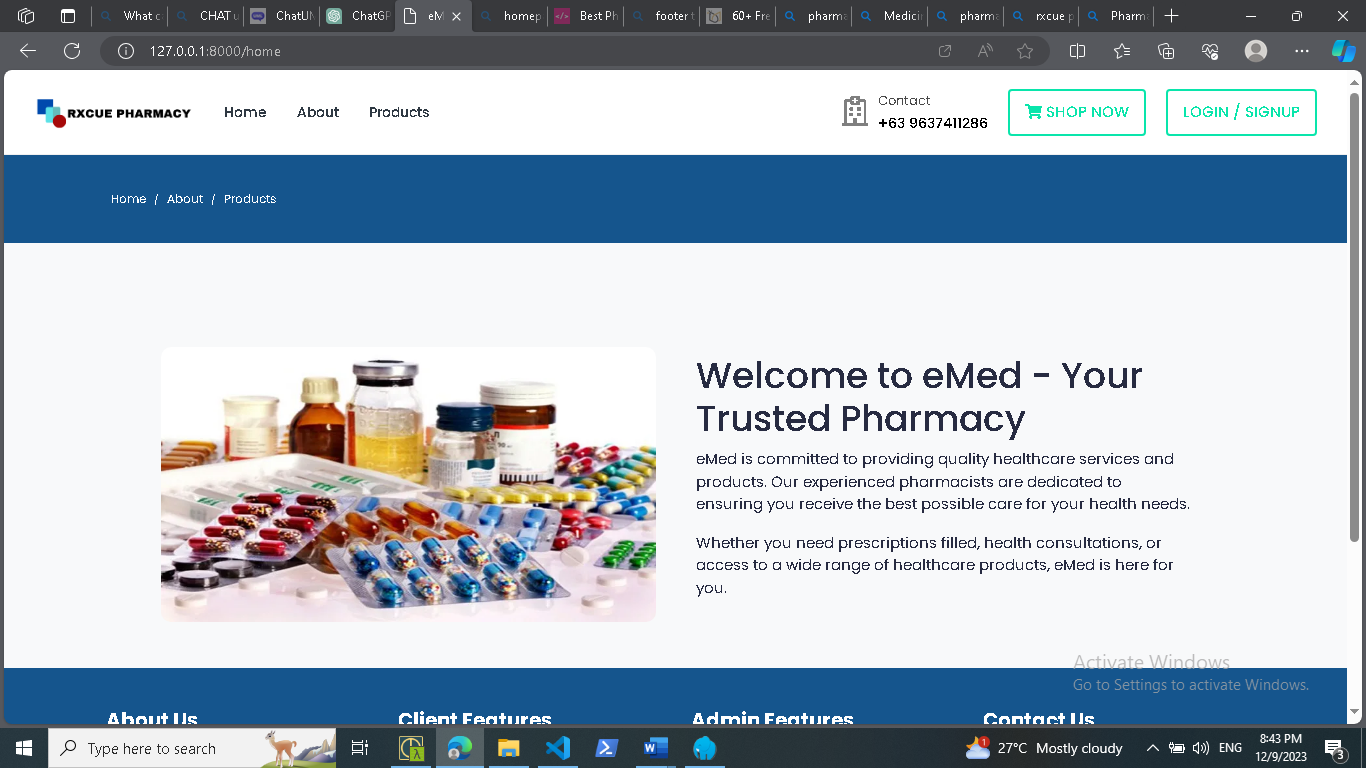
This section is referring to the testing and evaluation phase of the software development process. During this phase, developers are required to conduct tests on their system to determine its capabilities and limitations. This will allow them to identify any issues or potential problems prior to the production and deployment stages. The tests should include all the requirements outlined in the Requirements Phase, such as design, performance, supportability, etc. The results of these tests will be evaluated to assess the progress of the system and ensure it meets the requirements of the project. The developers considered the following:

1. **Unit Testing**- a type of software testing where individual units or components of a software system are tested. This type of testing is usually done by the developers, as it requires detailed knowledge of the internal structure of the system. Unit Testing is designed to test individual functions, modules, and features of the system, to ensure that they all behave as expected
2. **Component Testing**- involves testing the individual components of the system (modules, classes, objects, and programs) in isolation, without integrating them with other components. This helps to identify any defects or bugs in the individual components before they are integrated into the system. It also helps to identify any unexpected interactions or dependencies between components that could cause problems or errors in the system.
3. **System Testing**- a type of software testing that evaluates the entire system or application and its components to verify that all individual modules are working properly and that data is transferred accurately between modules and the entire system. System testing is meant to ensure that the system meets its requirements, performs as expected, and functions correctly in its intended environment. It is an overall test of the system and its components, and it is typically done after unit and integration testing.

**CHAPTER IV**

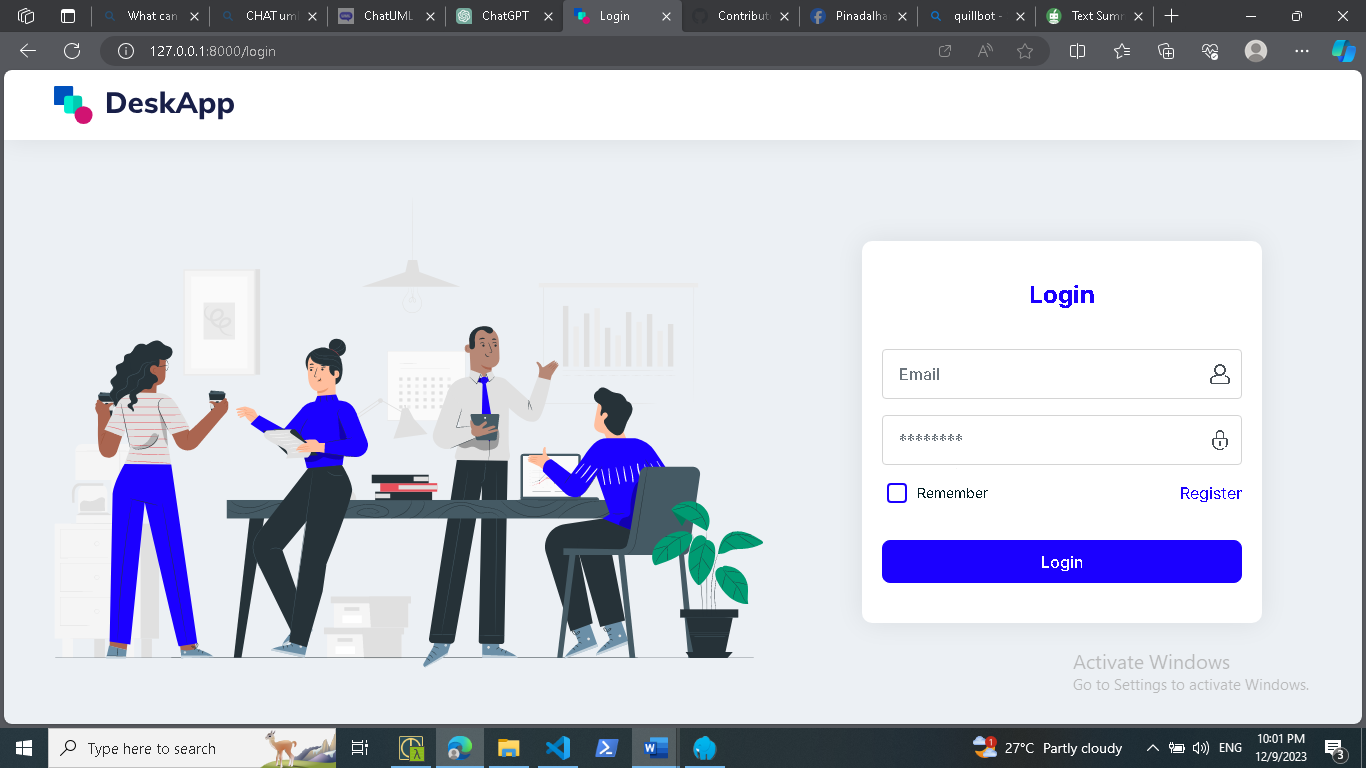
**DEVELOPMENT, TESTING AND EVALUATION RESULT**

**Presentation of the System Output**

****

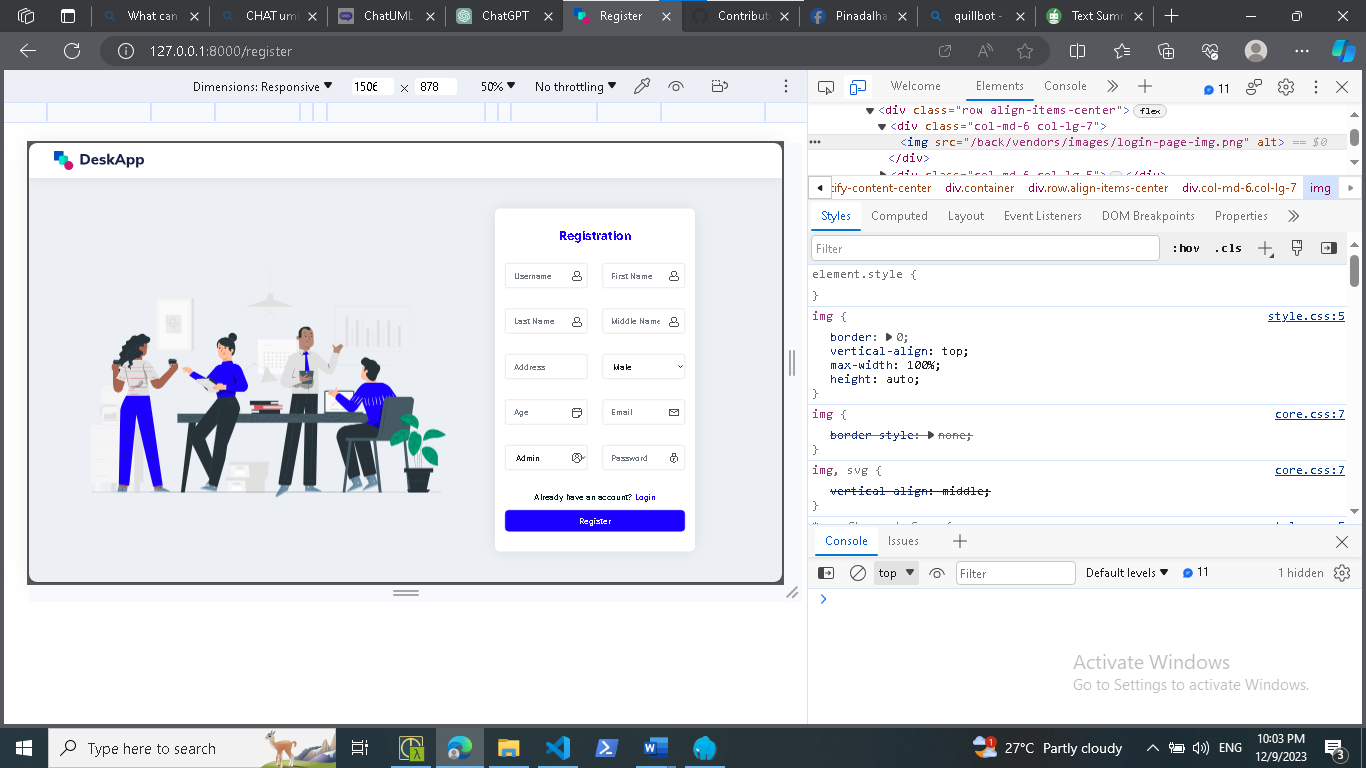
**Figure 7. Landing Page**

Figure 7 shows the landing page of the system where you users can access the see the home, about and products. If the user wants to purchase medicine in the pharmacy website, it will redirect first to login or registration.

****

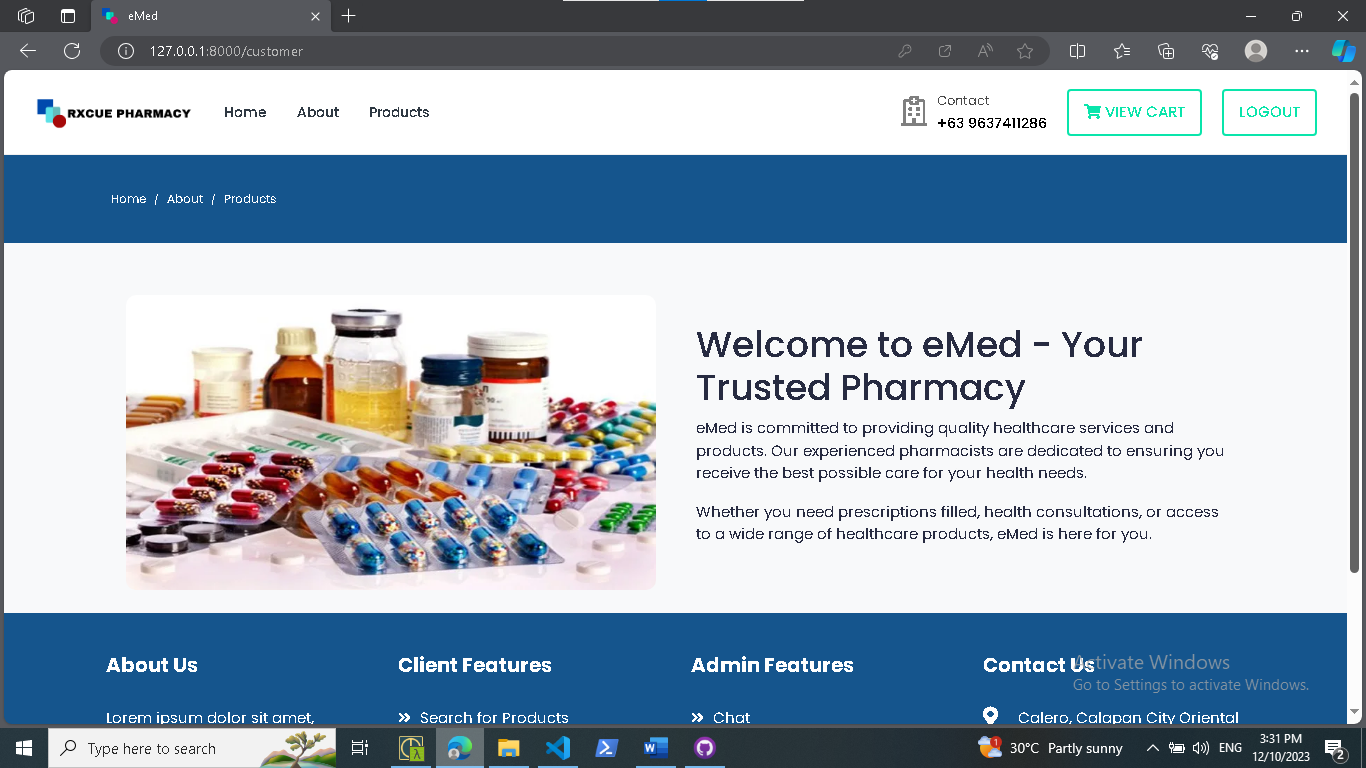
**Figure 8. Login**

Figure 8 shows the login page of the system where the users can sign in.

****

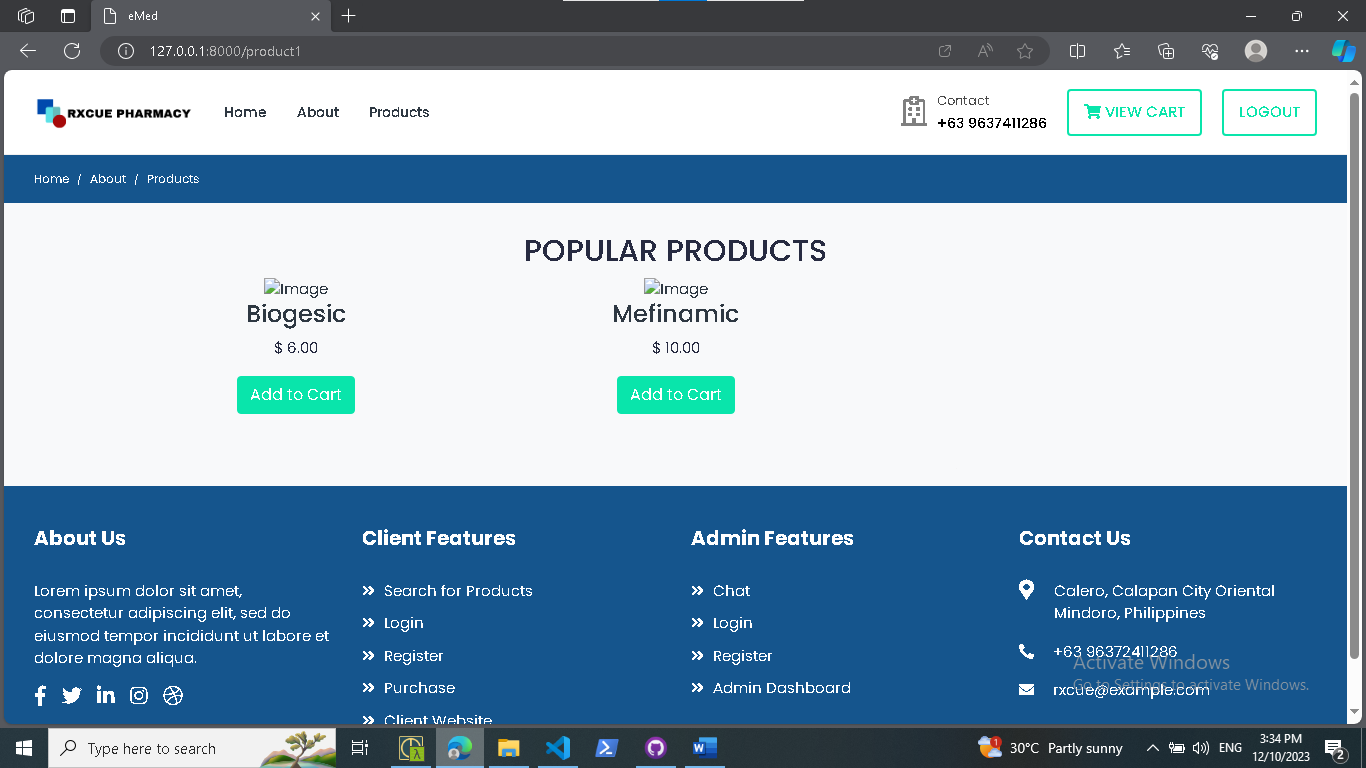
**Figure 9. Registration**

Figure 9 shows the registration page of the system where users can register if they don’t have an account yet.



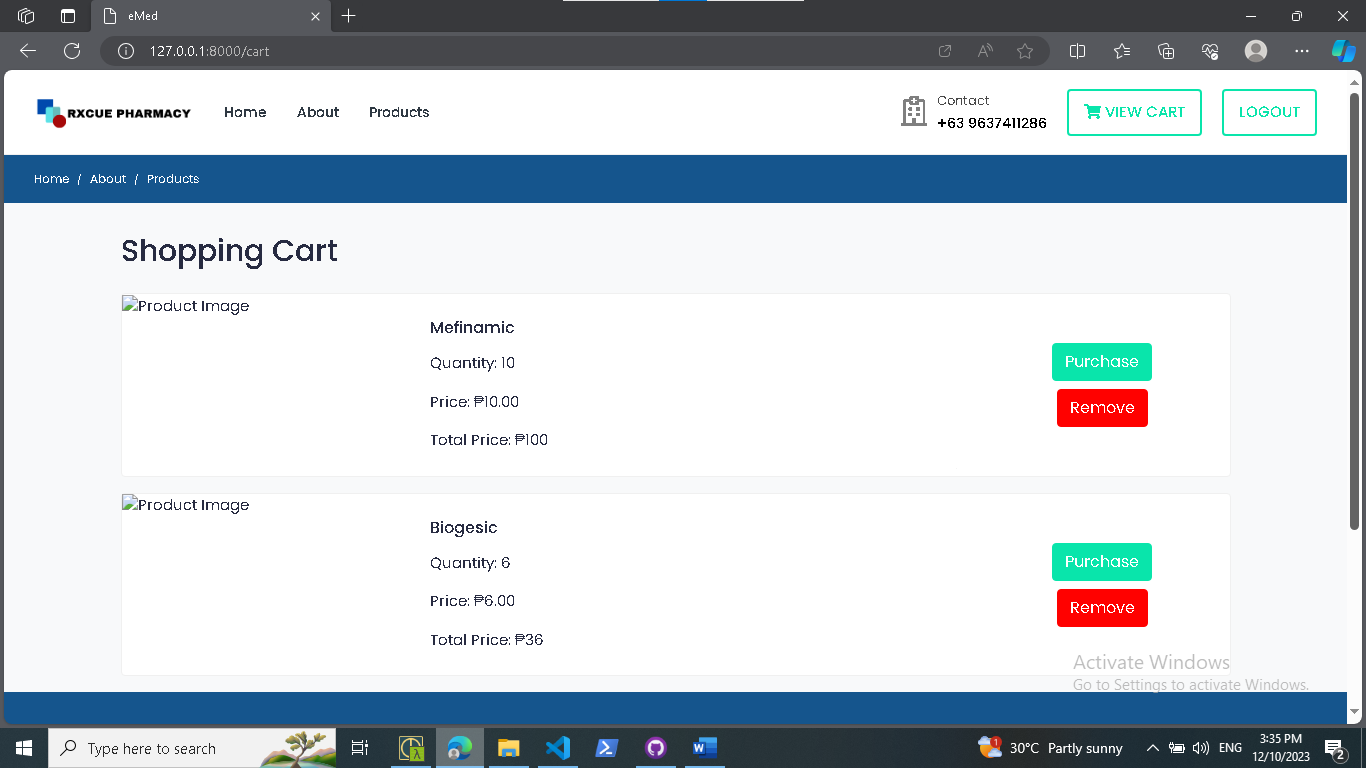
**Figure 10. Ecommerce Site**

Figure 10 shows the ecommerce site of the pharmacy where users can purchase medicine.

****

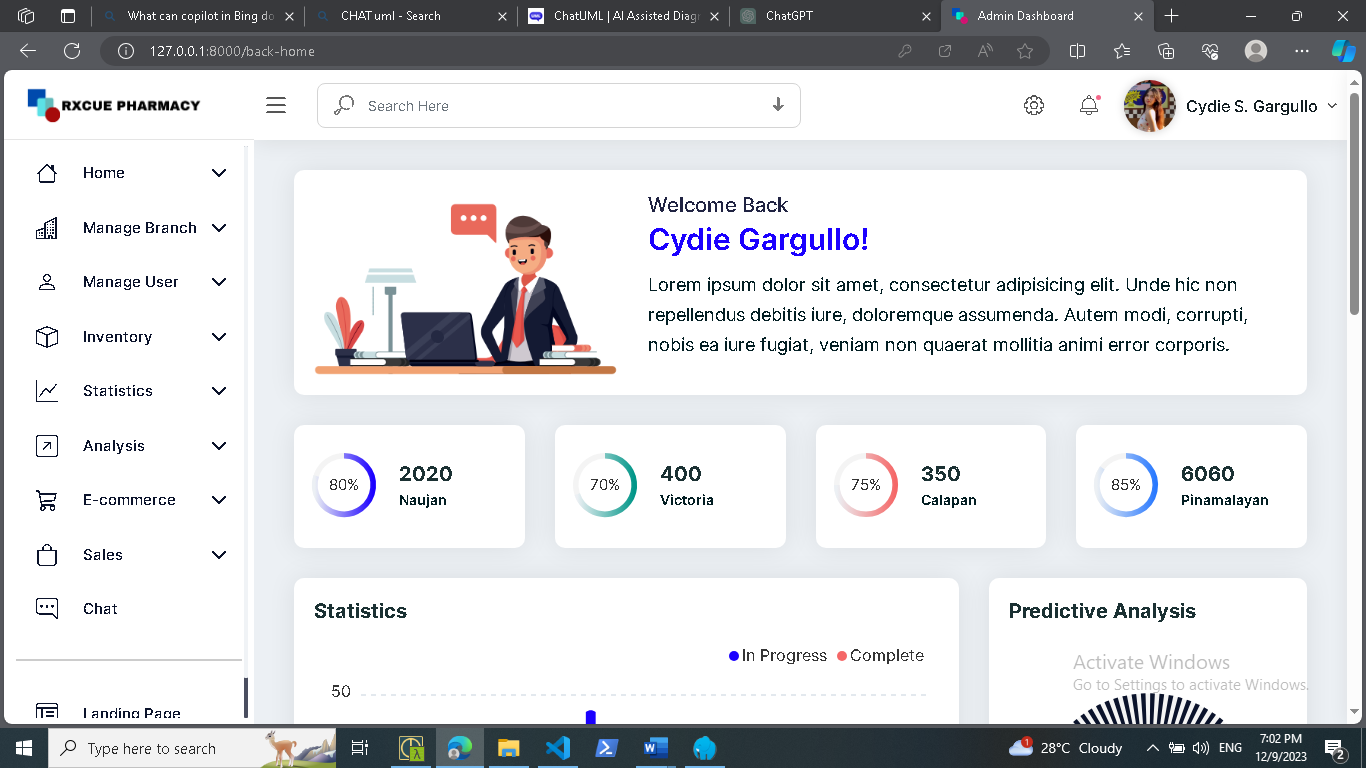
**Figure 10.1 Products**

Figure 10.1 shows the popular products that are being purchase by the users.

****

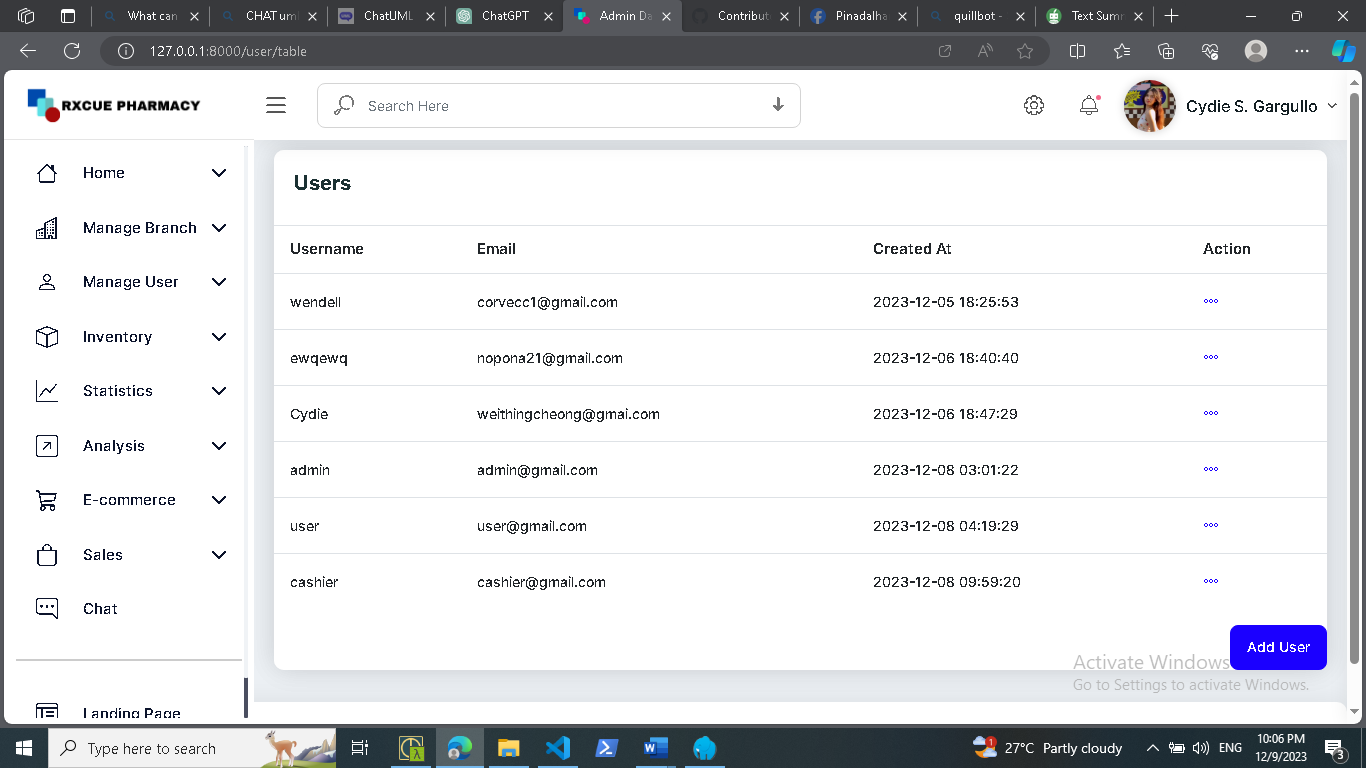
**Figure 10.2 Shopping Cart**

Figure 10.2 shows the shopping cart of the users where it can perform some tasks like purchase or remove cart product.

****

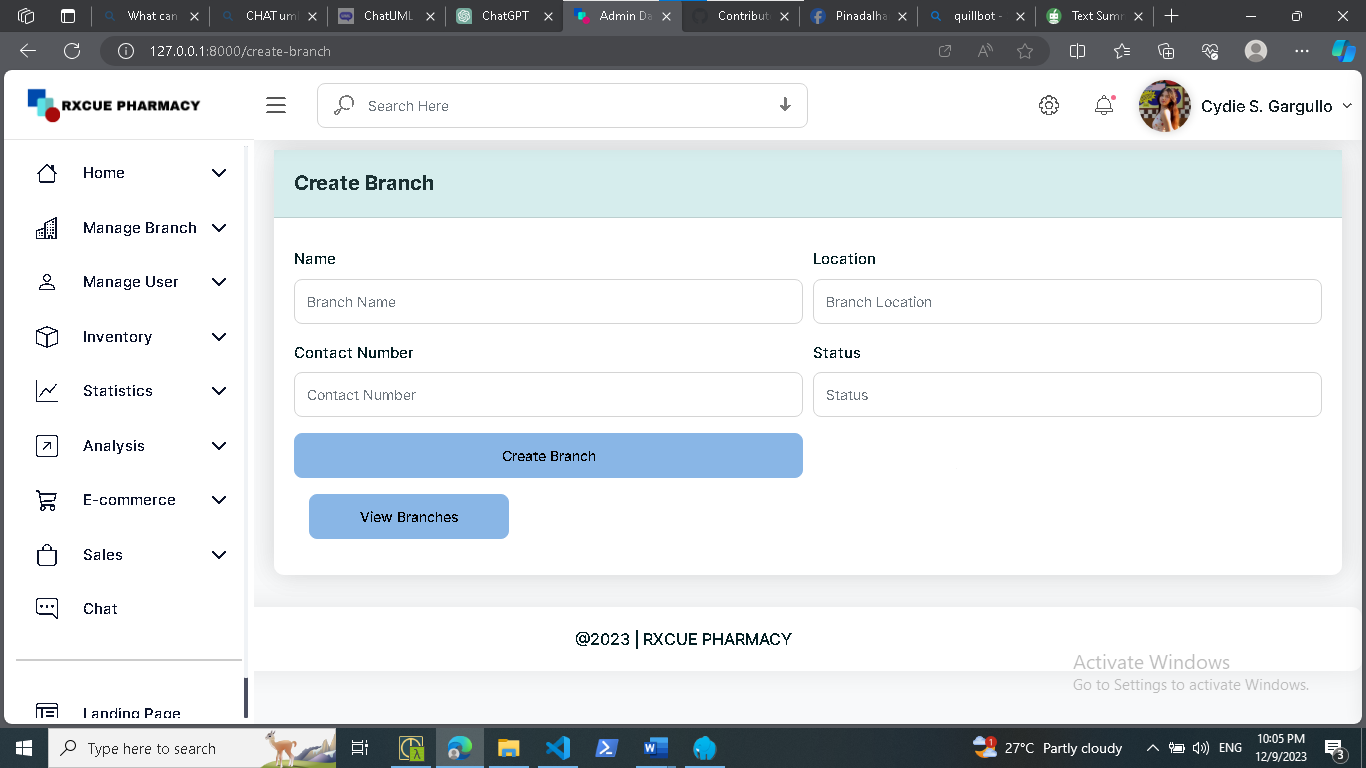
**Figure 11. Admin Dashboard**

Figure 11 shows the admin dashboard of the system where admin can do various tasks such as manage user, manage branch, manage inventory, etc.

****

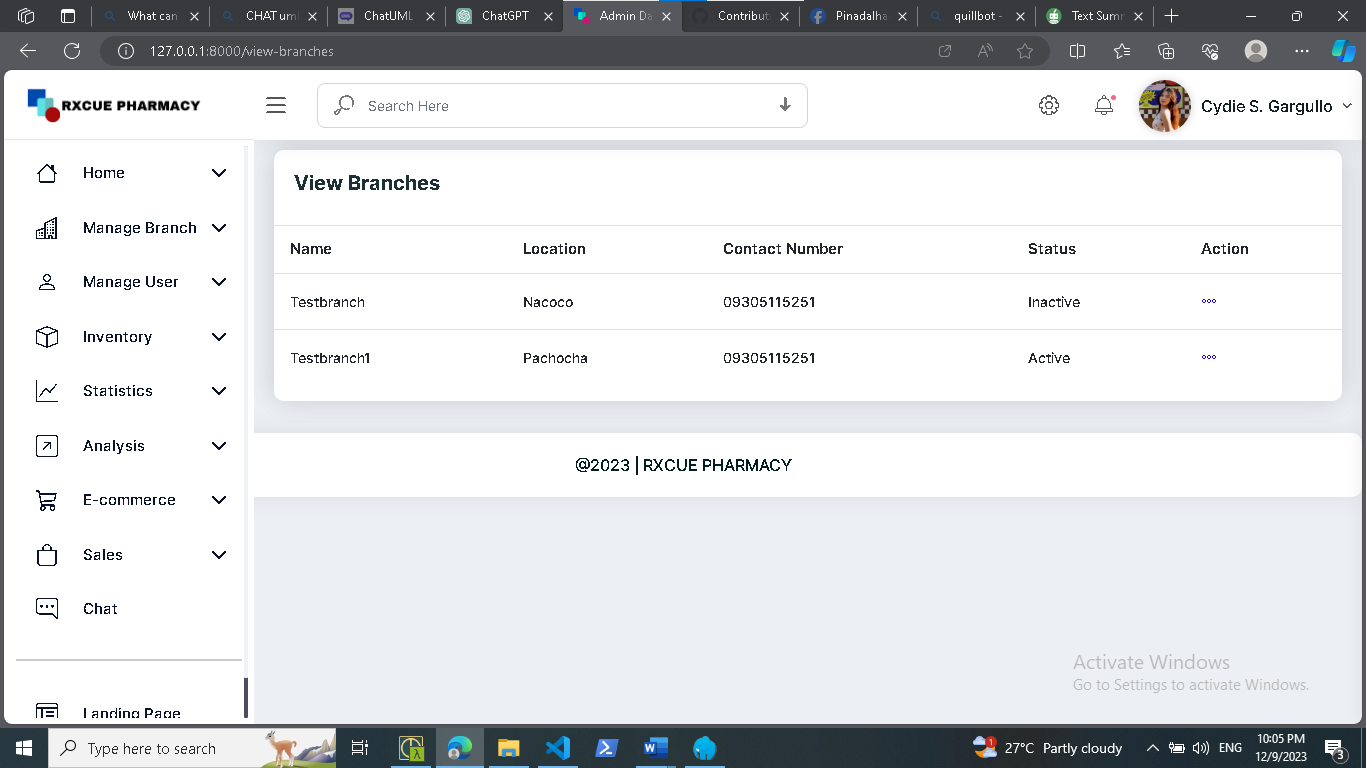
**Figure 11.1 Manage Users**

Figure 11.1 shows the users account where admin can manage and perform tasks such as add, edit or archive account.

****

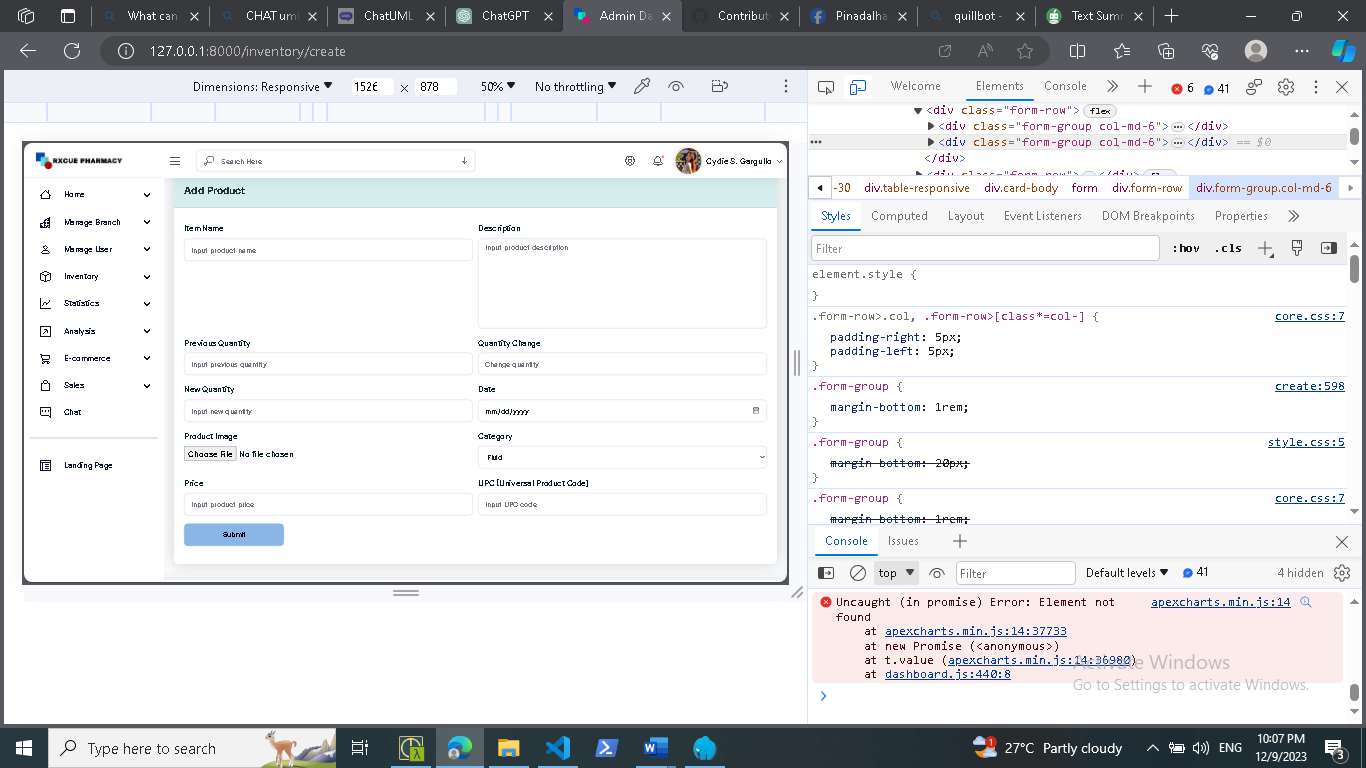
**Figure 11.2 Create Branch**

Figure 11.2 shows the branch feature of the system where admin can perform tasks such as creating a branch.

****

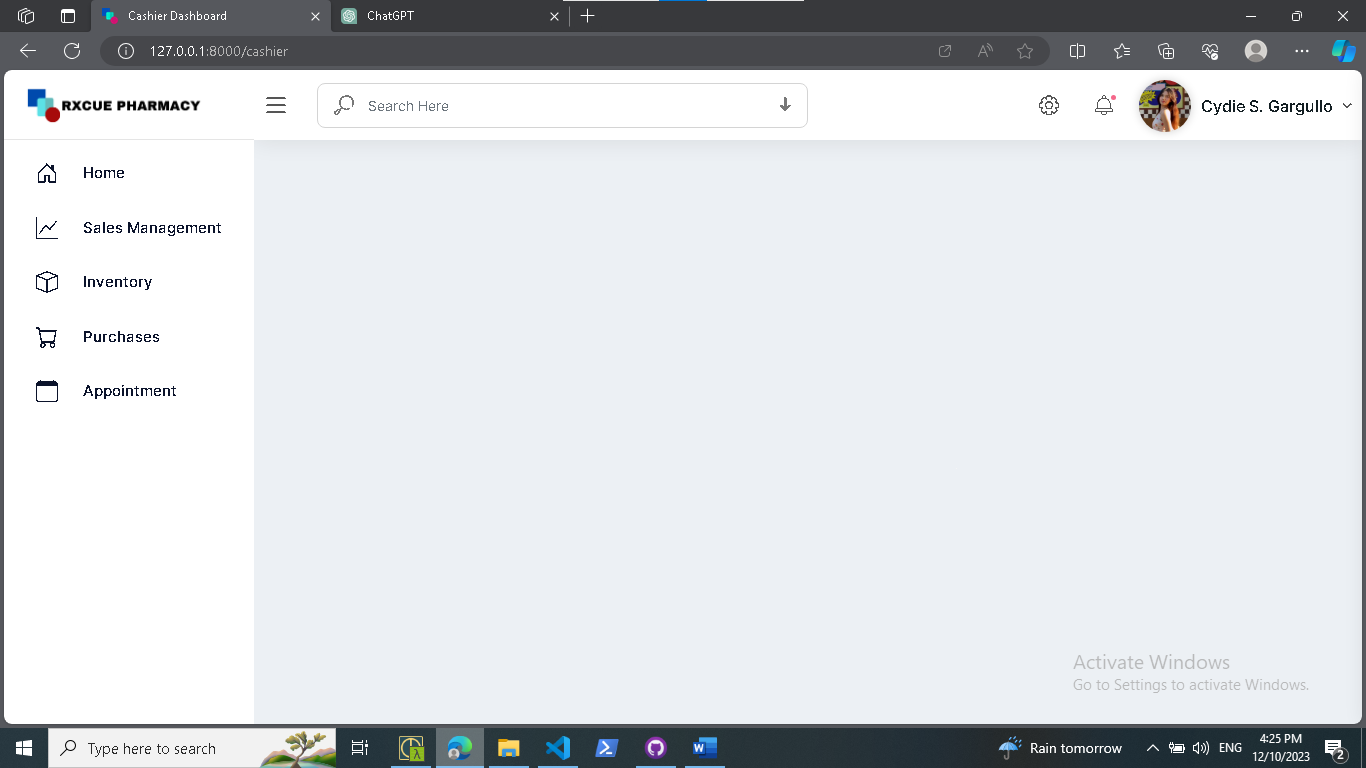
**Figure 11.3 View Branch**

Figure 11.3 shows the records of branches created by the admin.

****

**Figure 12. Inventory**

Figure 12 shows the inventory of the system where admin can perform various tasks such as adding product, view product, archive product.

****

**Figure 13. Cashier Dashboard**

Figure 13 shows the cashiers dashboard where cashier assign appointment number transaction, manage sales, have access to inventory, void cancellation and view order history.

**CHAPTER V**

**CONCLUSION AND RECOMMENDATION**

**Conclusion**

1. In summary, eMed, the web and mobile application developed for RXCue Pharmacy, serves as a computerized solution aimed at easing the responsibilities of pharmacists, cashiers, and clients, thereby enhancing the overall operational process.
2. The system incorporates diverse functionalities, including inventory management, user and branch administration, statistical analysis, predictive modeling, a chatbot, and an integrated e-commerce site.
3. Unfortunately, the system development did not achieve the anticipated milestones within the given time constraints, highlighting a need for reassessment and adjustment of the timeline for future developments.
4. Despite this setback, the implemented features address key aspects of pharmacy management, promising substantial benefits as the system evolves and reaches its full potential.

**Recommendation**

In the light of the findings and conclusions, the following recommendations are hereby offered:

1. Prioritize the completion of core features critical for the effective functioning of the system, such as inventory management and user administration. This ensures that essential functionalities are available for use while additional features are being developed.
2. Foster a close collaboration with RXcue Pharmacy stakeholders, including pharmacists, cashiers, and end-users, to gather feedback on system functionalities. Incorporate this feedback into the development process to align the system more closely with user expectations.
3. Conduct a thorough review of the initial project timeline and milestones. Identify the reasons for not achieving the anticipated milestones within the given time constraints. Reassess the timeline and set realistic expectations for future developments and updates.
4. Conduct regular evaluations of the eMed system to gauge user satisfaction, identify potential issues, and gather feedback from pharmacists, cashiers, and clients. This will serve as a continuous improvement mechanism to ensure the system aligns with evolving needs.
5. Implement training programs for pharmacy staff (pharmacists and cashiers) to enhance their proficiency in utilizing the eMed system effectively. Training should cover both basic functionalities and advanced features to maximize the system's potential.

**Checked and reviewed by:**

**Christian I. Cabrera**

**ITC311 - Facilitator**

**REFERENCES**

Kimble, C. A., & Chandra, A. (2001). Automation of pharmacy systems: Experiences and strategies of a rural healthcare system. *Hospital Topics*, *79*(2), 27–32. https://doi.org/10.1080/00185860109597904

Knapp, D. A. (2002). Professionally determined need for pharmacy services in 2020. *American Journal of Pharmaceutical Education*, *66*(4), 421–429.

*Verano\_ Revitalizing health service with e-pharmacy\_*. (n.d.).