

**Software Engineering**  
**Deliverable-2**  
**Project Plan Document**

**Medical Supply Management**

**By**

**Samuel D S Gnanaseelan: PES1UG22CS518**

**Sai Mohananshu: PES1UG22CS501**

## **Lifecycle**

Since we are doing the project where we are still learning the subject and implementing features which might not be needed, we will focus on doing the project in an agile cycle.

Allocating work to be done in each sprint and meeting with the teacher after every sprint in the form of reviews is ideal as we can produce a much-refined product at the end.

The team members will also be in constant touch to make sure the needed work for the sprint is finished before the reviews held by the teacher.

The teacher will also be requested to give any inputs and correct us if we go wrong anywhere.

## **Tools**

### **Planning Tools:**

- File sharing: Google Drive is used to share documents such as images, code etc.
- Communication: Discord and WhatsApp will be used to communicate deadlines and the work to be done everyday.

### **Design Tools:**

Traditional pen and paper are used to draw the ER diagrams and the relational schema which will be verified by the teacher.

Once verified an online design tool (draw.io) will be used to draw the finalised ER diagram and the relational schema.

Drawing apps like MS Paint and Excalidraw to be used in describing the user interface and various views of the project.

### **Development Tools:**

XAMPP (which includes Apache, MySQL, PHP, and phpMyAdmin) for both the frontend and backend. Local server for PHP and MySQL development. Using Apache and MySQL services for backend server and database.

Visual Studio Code: Lightweight, feature-rich editor with extensions for PHP, MySQL, HTML, and JavaScript.

Web Browser to test and debug the frontend (HTML/CSS/JavaScript).

phpMyAdmin: Included with XAMPP for easy database management and interaction. Helps us create and manage tables, run SQL queries, and handle database tasks without writing manual SQL.

### **Version Control Tools :**

Git: Version control system to track code changes and collaborate.

GitHub: Cloud-based platforms for hosting our repository. Let's us push our code to GitHub, track changes, and roll back when needed.

### **Testing Tool:**

Browser Developer Tools: Use browser developer tools (F12) to inspect elements, view console logs, and debug.

Chrome DevTools: Useful for inspecting HTML/CSS, debugging JavaScript, and tracking network requests.

The app will be hosted on a site(locally) where the user can interact and make sure all functionalities are fulfilled. The user can access this site on any browser.

## **Deliverables**

### **Reuse Components**

1. Database Management System: SQL Server (MySQL) will be used for the database, since the course we are doing covers SQL
2. Authentication: SQL will be used to create new users and grant the respective roles to them
3. Front-End: HTML/CSS and JavaScript will make up our frontend, as we have done a course and have a good understanding of it.

### **Build Components**

1. Database Schema: The design of tables and relational schema will be first done on paper. They will be later verified and approved by the teacher to be implemented.
2. Inventory Management: Build a system to manage stock levels, track sales/purchases, and automatically update inventory quantities. This is the main functionality of the project which cannot be imported from existing modules
3. User Interface: A good user-friendly interface will be done, with inputs from the teacher and other peers. A default UI cannot be used as it must be custom made for the users based on their requirements
4. Searching Functionality: We need to search for products based on the attributes defined by us.
5. Alerts: When stock level is low alerts must be given to replenish that stock level

## **Work Breakdown Structure**

### **System Design**

- Database Design:
  - ER Diagram and Relational Schema is first drawn and verified
  - Tables are defined using the schema
- Front-End
  - A rough look for the user-interface using online paint tools
- Integration
  - A flowchart to make sure the correct tools are used to integrate the front-end and backend for a good user experience

### **Development**

- Database:
  - Tables are defined along with their primary and foreign keys
  - Adding the values to the database
- Authentication
  - Relevant user roles to be created and assigned to each user
  - User role authentication to prevent unauthorized access
- Front-End
  - UI for inventory tracking and searching
  - Alerts when stock level is low

### **Testing**

- Functionality
  - Inventory management features
  - Alert system
  - Relations between each table
- Performance
  - Efficient searching standards must be met
  - Consistent data is retrieved
  - No atomicity must be there
- Authentication
  - Each user should be able to see what they are supposed to see
  - No leakage of data
- Front-End
  - Simple Design
- Integration
  - Changes in front-end must be reflected in the database as well

## Timeline

Project is divided into 6 weeks

Week1: Defining of Tables, primary keys foreign keys

Week2: Creation Of user roles and implementing alert system

Week3: Coding of queries for searching and monitoring the inventory

Week4: Front End part 1

Week5: Front End part 2(Mainly Integration with the database)

Week6: Testing of all functionalities

Task Name	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6
Defining Tables & Keys	<div></div>					
Creation of User Roles & Alerts		<div></div>				
Coding of Queries			<div></div>			
Front End - Part 1				<div></div>		
Front End - Part 2 (Integration)					<div></div>	
Testing All Functionalities						<div></div>

## **Coding Details :**

The Medical Supply Management System will be developed to manage the inventory and tracking of medical supplies, user roles, and order management. This system includes a responsive web interface, database management, and backend functionalities to ensure the smooth operation of inventory management in a medical context.

### **Languages and Frameworks Used:**

1. **HTML/CSS:** For designing and styling the web interface.
2. **JavaScript :** For dynamic interactivity, client-side validation, and asynchronous communication with the backend.
3. **PHP:** For server-side logic, database interaction, and handling business processes.
4. **MySQL:** For storing and managing data related to medical supplies, users, and orders.
5. **phpMyAdmin:** To manage and interact with the MySQL database.