**PHARMACY MANAGEMENT SYSTEM**

**BY**

**Foday M. Samura**

**SUBMITTED TO**

**COMPUTER SCIENCE DEPARTMENT  
SALEM STATE UNIVERSITY**

**COMPUTER SCIENCE CAPSTONE PROJECT PROPOSAL**

**December 2020**

**CSC 521 Project Advisor**

**Dr. Fatema Nafa**

**Table of Content**

Student Objectives

Problem Specification

Solution Processes and/or Design

Requirement Collection and Analysis

Identifying Need of the Users

Identifying Target Specification

Use Case Diagram

General Design Concepts

General Data Flow Diagram

Architectural Design

System Flowchart

Abstract Solution Design

Preliminary ER Diagram

Prototype of the final product (can be an initial version)

Benchmark Specifications

Tools List

Time Schedule

Grading Scheme

List of Deliverables

**Student Objectives**

**After the completion of this project my:**

***Primary objective***

* To gain practical experience by modeling a software based on real world problem.  
  To understand how to work on Front-end using HTML & CSS and Back-end (MySQL) by using server (XAMPP).
* learn what a data base is, about different types of databases, and why they are valuable assists for discussion making.
* Learn about relational database concepts, and their relationships.
* Create database and table structures, perform data manipulation, and query the database using Oracle SQL Developer
* Create and entity relationship (E-R) model.
* Develop a conceptual database design.
* Learn how to use PHP to connect to database.
* Learn how to create login form.
* Save the login form data into the database.

**Problem Specification**

Pharmacy Management System is a web-based application I am going to design and implemented with PHP using MySQL as the database. This project will be developed for medical stores to manage different inventories and items, customers’ and suppliers’ details as well as staffs (cashiers, managers, and pharmacists) working in the store. The application will have the entire basic module to manage the medical store operations.

The aim of the project is to create an effective software to help the pharmacist to maintain the records of the medicines, handle user details, generate invoice, check and renew validity and provide a scope of communication between users by using inbuilt messaging system. Pharmacy management system deals with the maintenance of drugs and consumables in the pharmacy unit. This pharmacy management system is user friendly.

Listed below are the main users of the system. They are required to register into the system to access records and details stored in the database. Only the head admin has the privilege or access over all records and users. The system access to other users in restricted as per their functions in the medical store.

* Admin
* Cashier
* Manager
* Pharmacist

**List of forms:**

* Log in and Log out
* Add, edit, update, and delete users
* Check
* Invoice
* Prescription: view and delete
* Stock: view and delete
* Stock pharmacist

**Solution Processes And/or Design**

**Requirement Collection and Analysis**

**Users of the system:**

Listed below are the main users of the pharmacy management system. They are required to register into the system to access records and details stored in the database. Only the head admin has the privilege or access over all records and users. The system access to other users in restricted as per their functions in the medical store.

* Admin
* Manager
* Cashier
* Pharmacist

**List of forms:**

* Log in and Log out
* Add, edit, update, and delete users
* Invoice
* Prescription: view and delete
* Stock: view and delete
* Stock pharmacist

**Function of the Admin**

* Admin Can Manage All users
* Admin Can Manage Manager
* Admin Can Manage Cashier
* Admin Can Manage Pharmacist

**Function of the Manager**

* View Users
* View Prescriptions
* Manage Stock

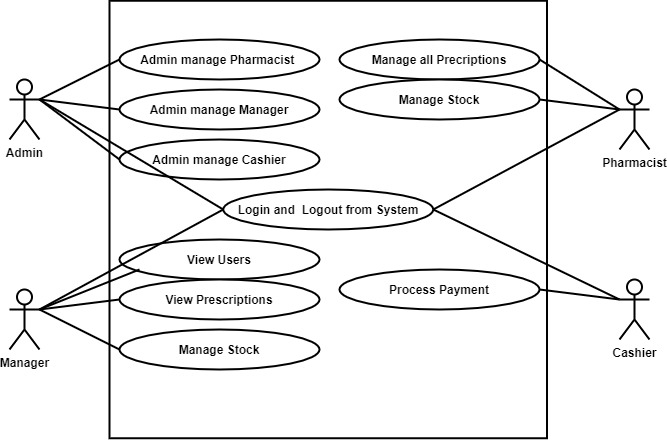
**Function of the Pharmacist**

* Manage Prescriptions
* Manage Stock

**Function of the Cashier**

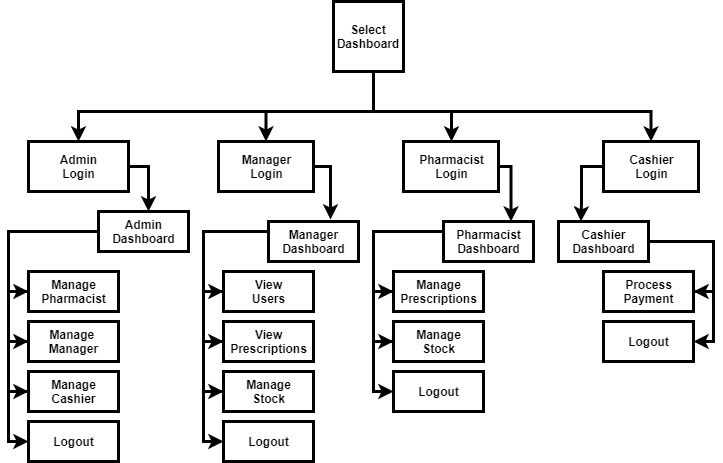
* Process Payment

**Use Case Diagram for The Pharmacy Management System**

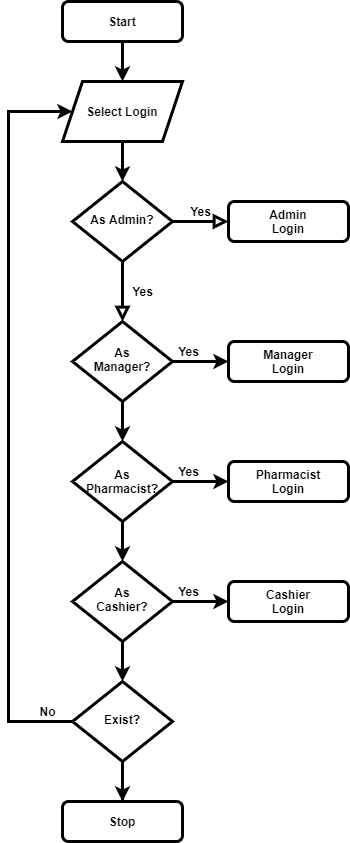
****

**Data Flow Diagram (DFD)**

**Pharmacy Management System**



**General System Flowchart**



**Use Case Description for Login to System for Admin**

***Use case name*** Login to System

***Summary*** System validates the Head admin

***Actor*** the Admin

***Precondition*** the login page must be displayed

***Flow of Event:***

This use case starts when a Head admin is not logged in to the system and goes to the login page.

1. The Head admin going to the login menu and click on it.

2. System prompts the Head admin for username and password

3. The Head admin inserts username and password to the fields.

4. System checks the username and password.

5. If the input were valid value then system will display general system of the pharmacy system.

***Alternate Flows:***

Step 5: if the username or password is not correct, the System displays an

Error message. And prompts for the correct username and password.

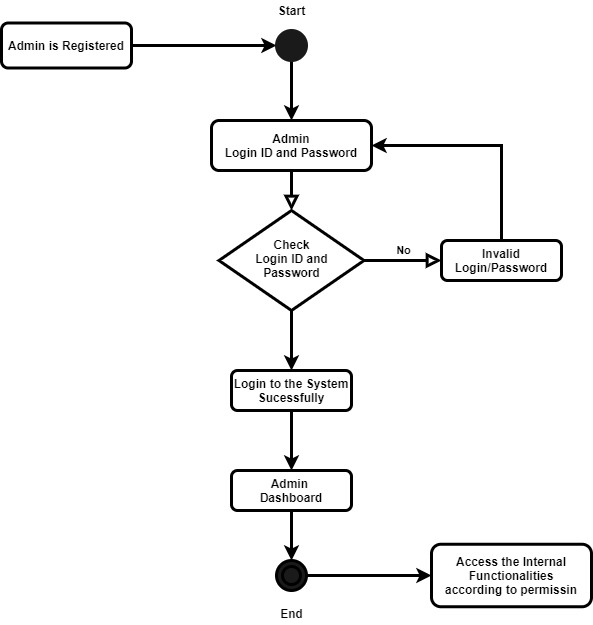
Step 5: if the Head admin tries to insert username and Password three times

Without restarting the System, it will close.

Step 1-4: if the user clicks on cancel, the system will go on the pharmacy Navigation page.

**Post condition:** username and password of the Head admin is Validated.

**Admin Details Data Flow**



**USE CASE DESCRIPTION FOR LOGIN TO SYSTEM FOR MANAGER**

***Use case name*** Login to System for manager admin

***Summary*** System validates the manager admin

***Actor*** the manager admin

***Precondition*** the login page must be displayed

***Flow of Event:***

This use case starts when the manager admin is not logged in to the system and goes to the login page.

1. The manager admin going to the login menu and click on it.

2. System prompts the user for username and password

3. The manager admin inserts username and password to the fields.

4. System checks the username and password.

5. If the input were valid value then system will display general system of the pharmacy system.

***Alternate Flows:***

Step 5: if the username or password is not correct, the System displays an

Error message. And prompts for the correct username and password.

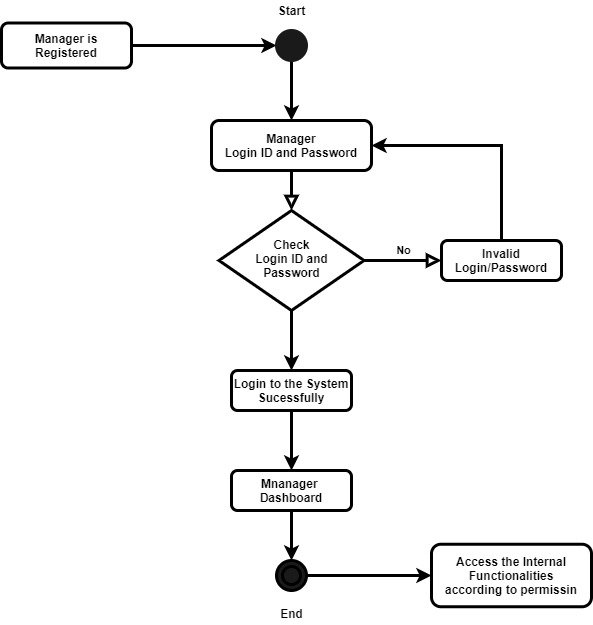
Step 5: if the manager admin tries to insert username and Password three times

Without restarting the System, it will close.

Step 1-4: if the manager admin clicks on cancel, the system will go on the pharmacy Navigation page.

**Post condition:** username and password of the manager admin is Validated. **Login Activity**

**Manager Details Data Flow**



**Use Case Description for Login to System for Pharmacist**

***Use case name*** Login to System

***Summary*** System validates the Pharmacist admin

***Actor*** the Pharmacist

***Precondition*** the login page must be displayed

***Flow of Event:***

This use case starts when the pharmacist admin is not logged in to the system and goes to the login page.

1. The Pharmacist admingoing to the login menu and click on it.

2. System prompts the Pharmacist adminusername and password

3. The Pharmacist admininserts username and password to the fields.

4. System checks the username and password.

5. If the input were valid value then system will display general system of the pharmacy system.

***Alternate Flows:***

Step 5: if the username or password is not correct, the System displays an

Error message. And prompts for the correct username and password.

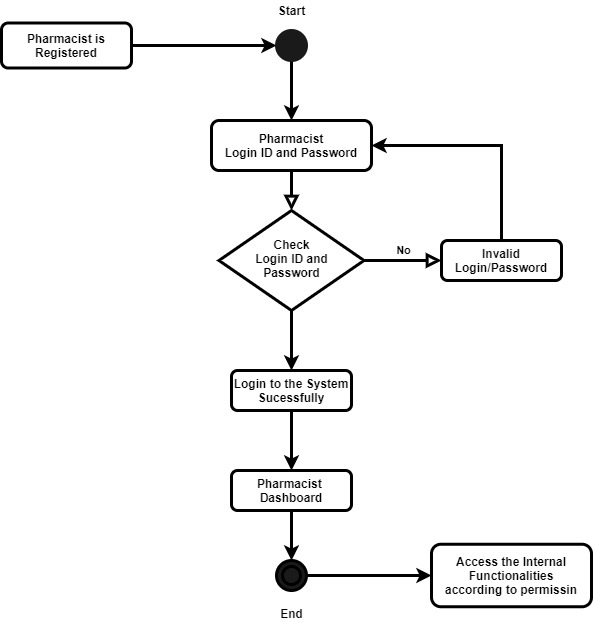
Step 5: if the Pharmacist admin tries to insert username and Password three times

Without restarting the System, it will close.

Step 1-4: if the Pharmacist adminclicks on cancel, the system will go on the pharmacy Navigation page.

**Post condition:** username and password of the Pharmacist admin is Validated.

**Pharmacist Details Data Flow**



**Use Case Description for Login to System for Cashier**

***Use case name*** Login to System for Cashier

***Summary*** System validates the Cashier admin

***Actor*** the **cashier**

***Precondition*** the login page must be displayed

***Flow of Event:***

This use case starts when the cashier admin is not logged in to the system and goes to the login page.

1. The pharmacy manager going to the login menu and click on it.

2. System prompts the Cashier admin for username and password

3. The pharmacy manager inserts username and password to the fields.

4. System checks the username and password.

5. If the input were valid value then system will display general system of the pharmacy system.

***Alternate Flows:***

Step 5: if the username or password is not correct, the System displays an

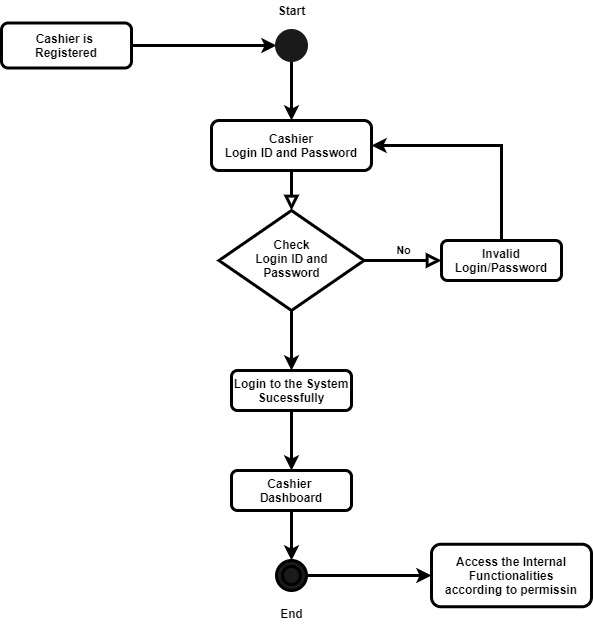
Error message. And prompts for the correct username and password.

Step 5: if the Cashier admin tries to insert username and Password three times

Without restarting the System, it will close.

Step 1-4: if the Cashier admin clicks on cancel, the system will go on the pharmacy Navigation page.

**Post condition:** username and password of the Cashier admin is Validated.

**Cashier Details Data Flow**

**Use Case Description the Head Admin of The System Modifies The System**

***Use case name*** modifies the system

***Summary*** the Head admin of the System modifies the system in the database

***Actor*** the Head admin

***Flow of Event:***

This use case starts when the Head admin accesses the system.

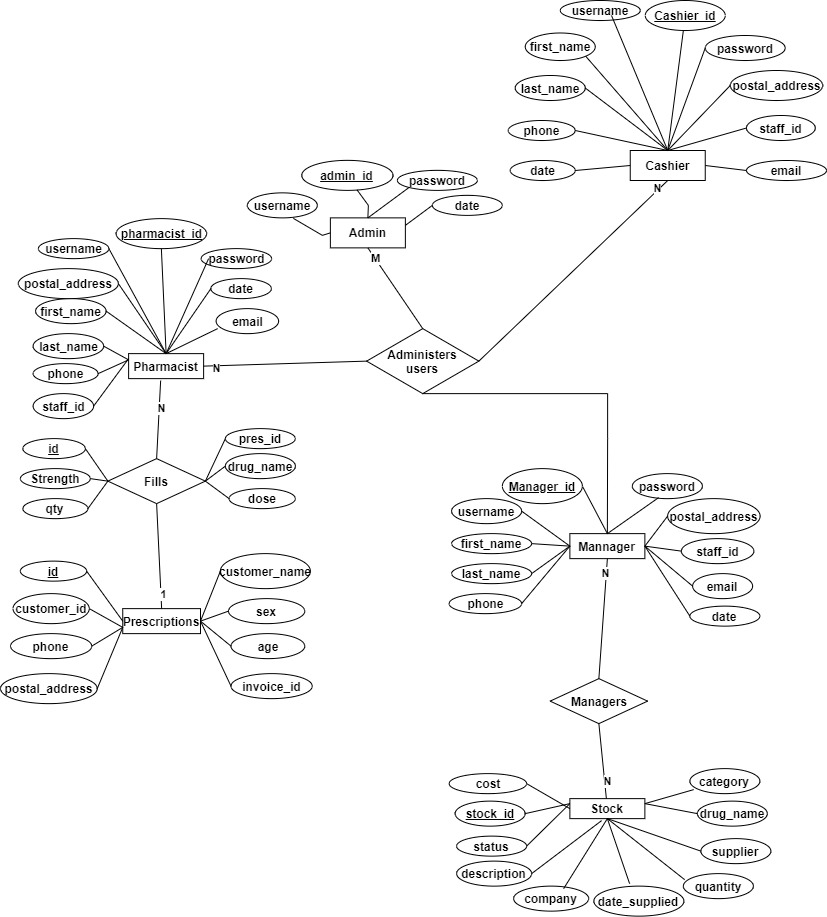
1. The System displays the current list of record.
2. The head admin selects the record he/she wants to modify.
3. The system displays the corresponding information.
4. The head admin edits the information accordingly and submits.
5. The system updates the information.

Use Case ends.

***Alternate Flows:***

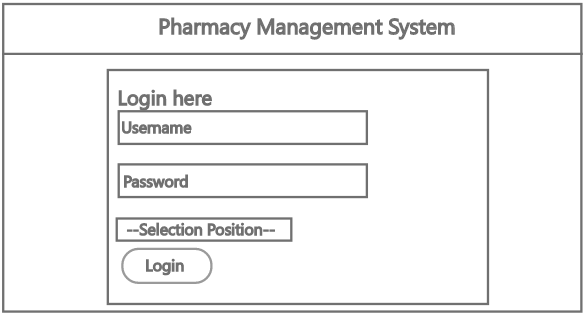
1. The system displays an “Invalid Information” error after the Head admin submits the information and asks the head admin to re-enter the information.
2. The Head admin re-enters information and clicks submit.

**Preliminary ER Diagram**

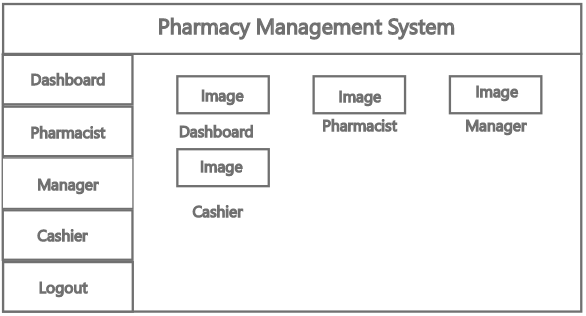


**Prototype of The Pharmacy Manage System**

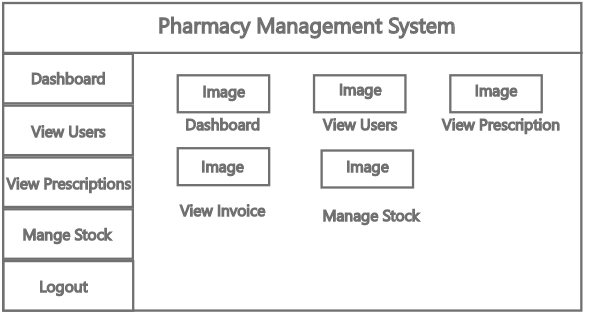
**Sample Form**



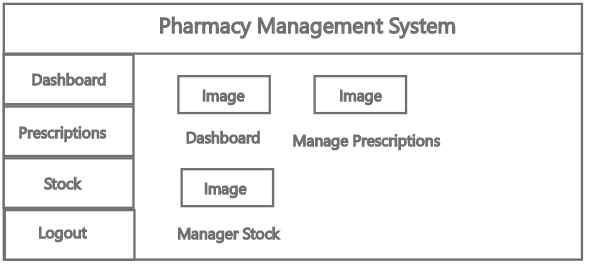
**Home Page Admin**



**Home Page Manager**



**Home Page Pharmacist**



**Home Page Cashier**



**Benchmark Specifications**

Benchmark 1: Backend Implementation

Benchmark 2: Admin GUI and Logic Implementation and Testing

Benchmark 3: Manager GUL and Logic Implement and Testing

Benchmark 4: Manager GUL and Logic Implement and Testing

Integrated Testing and Report Writing

**Tools List**

**Server-side programming Tools**

* PHP
* Apache

**Front-End Development Tools**

* HTML
* CSS
* JavaScript

**Back-End Database Programming Tools**

* MySQL
* phpMyAdmin

**IDEs**

* Adobe Dreamweaver
* Visual Studio Code

**Version Control Tools**

* Git
* GitHub

**Others**

* Draw.io
* Inkscape
* Adobe XD

**Server-side programming Tools**

**Hypertext Pre-Processor (PHP)**

PHP is a popular open-source general-purpose HTML-embedded server-side scripting language, which enables web developers to create dynamically generated web pages quickly**.** Php is a programming language that can do all sorts of things: evaluate form data sent from a browser, build custom web content to serve the browser, talk to a database, and even send and receive cookies (little packets of data that your brewer uses to remember things, like if you’re logged in to any website).

**Apache** **Webserver**

Apache Webserver is generally recognized as the world’s most popular Web server (HTTP server). Originally designed for UNIX environments, the Apache Web server has been ported to Window and other network operating systems. The name “Apache” derives from the word “patchy” that the Apache developers used to describe early versions of their software.

Apache Web server provide full range of Web server features, including CGI, SSL and virtual domains. Apache also supports plug-in modules for extensibility. Apache is free software, distributed by the Apache Software Foundation that promotes various free and open source advance Web technologies.

Apache features a modular design and supports dynamic selection of extension modules at runtime. Some of its strong points are its range of possible customization, dynamic adjustment of the number of server processes, and a whole range of available modules including many authentication mechanisms, server-parsed HTML, server-side includes access control, CERN http metafiles emulation, proxy caching, etc. Apache also supports multiple virtual homing. Separate Debi a package is available for PHP3, mod-Perl, Java Servlet support, Apache-SSL, and other common extensions

**Front-End Development Tools**

**Hyper Text Markup Language (HTML)**

HTML is the language for publishing web pages on the WWW (World-Wide Web, or World-Wide Wait?). HTML is a *Document Description Language* (aka *Document Mark-*up Language). HTML is NOT a programming language like C/C++/C#/Java, which is used to implement programming algorithm. An HTML document is a text document, and it is human-readable.

HTML uses mark-up *tags*, such as <p> (for Paragraph), <h1> to <h6> (for Heading Level 1 to 6), <img> (for Image), <a> (for Anchor or Hyperlink), to mark-up a document. HTML mark-up tags perform these functions:

1. Layout the documents, e.g., <p> (layout as a paragraph), <h1> to <h6> (layout as heading level 1 to 6), <br> (perform a line break), <hr> (draw a horizontal rule), <table> (tabulating data), <ol> (layout an ordered list).
2. Provide link (called hyperlink) to another HTML document, via the <a> (Anchor tag). These hyperlinks, a distinct feature in HTML, greatly help the users in navigating the web and enrich the users' experience. Hyperlinks make the HTML popular.
3. Embed images, audios, videos, programs (in JavaScript, VBScript, Applet, Flash, or MS ActiveX control), and objects within an HTML document. HTML is *multimedia*! The hypertext document may contain texts, images, audios, videos, and even programs.

HTML elements form the building blocks of all websites. HTML allows image and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, list, links, quotes and other items. It can embed scripts written in languages such as JavaScript which affect the behaviour of the HTML web pages.

**Cascading Style Sheets (CSS)**

Cascading Style Sheets (CSS) is a styles sheet language used for describing the look and formatting of a document written in the mark-up language.

Style Sheet Language called CSS (Cascading Style Sheet) for presentation and removing the presentation tags and attributes from HTML. CSS can be viewed as a *companion* of HTML. It allows web graphic designers to spice up the web pages, so that the content providers can focus on the document contents with HTML.

CSS is a cornerstone specification of THE web and almost all web pages use CSS style sheets to describe their presentation.

**JavaScript (JS)**

JavaScript is the most widely used *client-side* programming language that lets you supercharge your HTML with interactivity, animation and dynamic visual effect for better User Interface and User Experience (UI/UX). It is:

* a small, lightweight, object-oriented, cross-platform, special-purpose scripting language meant to be run under a host environment (typically a web browser).
* a *client-side scripting language* to enrich web user-interfaces and create dynamic web pages (e.g., form input validation, and immediate response to user's actions).
* the engine that supports AJAX (Asynchronous JavaScript and XML), which generate renew interest in JavaScript.

JavaScript works together with HTML/CSS. HTML provides the contents (or structure); CSS specifies the presentation; and JavaScript programs the behaviour. Together, they enrich the UI/UX of the web users.

**Back-End Database Programming Tools**

**MySQL**

 MySQL is a software system - a Relational Database Management System. MySQL is one of the most used, industrial-strength, open-source and free Relational Database Management System (RDBMS). MySQL is successful, not only because it is free and open-source (there are many free and open-source databases, such as PostgreSQL, Apache Derby (Java DB), mSQL (mini SQL), SQLite and Apache OpenOffice's Base), but also for its speed, ease of use, reliability, performance, connectivity (full networking support), portability (run on most OSes, such as Unix, Windows, macOS), security (SSL support), small size, and rich features.

On top that, it’s very commonly used in conjunction with PHP scripts to create powerful and dynamic server-side applications.

MySQL is easy to use, yet extremely powerful, secure, and scalable. And because of its small size and speed, it is the ideal database solution for web sites.

MySQL is often deployed in a LAMP (Linux-Apache-MySQL-PHP), WAMP (Windows-Apache-MySQL-PHP), or MAMP (macOS-Apache-MySQL-PHP) environment.

**phpMyAdmin**

phpMyAdmin is a free software tool written in PHP, intended to handle the administration of MySQL over the Web. phpMyAdmin supports a wide range of operations on MySQL and MariaDB. Frequently used operations (managing databases, tables, columns, relations, indexes, users, permissions, etc) can be performed via the user interface, while you still have the ability to directly execute any SQL statement.

**Version Control Tools**

**Git**

GIT is a Version Control System (VCS) (aka Revision Control System (RCS), Source Code Manager (SCM)). A VCS serves as a *Repository* (or *repo*) of program codes, including all the historical revisions. It records *changes to files* at so-called *commits* in a *log* so that you can recall any file at any commit point.

**GitHub**

GitHub is a website and cloud-based service that helps developers store and manage their code, as well as track and control changes to their code. Git is a command-line tool, but the centre around which all things involving Git revolve is the hub—GitHub.com—where developers store their projects and network with like-minded people.

**Other Tools**

**Inkscape**

**Inkscape** is an Imaging and Digital Photo application like PhotoScape, Optimizer, and ImageMagick from Inkscape Team. It has a simple and basic user interface, and most importantly, it is free to download. Inkscape is an efficient software that is recommended by many Windows PC users.

Inkscape is a very fast, small, compact and innovative Open Source Imaging and Digital Photo for Windows PC. It is designed to be uncomplicated for beginners and powerful for professionals. This app has unique and interesting features, unlike some other Imaging and Digital Photo apps. Inkscape works with most Windows Operating System, including Windows 7 64 / Windows 8 64 / Windows 10 64.

**Adobe Dreamweaver**

Adobe Dreamweaver is a software tool for front-end developers and web designers. It blends a robust design surface and code editor that helps easily code, design, and manage sites. The user interface (UI) has been designed keeping usability in mind, with selectable light and dark themes, and modern web workflow support. Bootstrap integration permits users create responsive websites that scale and adapt to multiple browsers.

**Draw.io**

Draw.io is a **free diagramming application** that allows users to create and share diagrams within a web browser. The online tool works with G Suite/Google Drive and Dropbox.

Users are able to create and edit a variety of diagrams including flowcharts, org charts, process diagrams, ER diagrams, UML, network diagrams, and more. draw.io’s rich functionality allows users to track and restore changes, import and export various formats, and automatically publish and share work.

**Adobe XD**

Adobe XD is a user interface creation software for macOS and Windows computers. It allows users to create and edit user interfaces for websites, mobile apps, computer programs, and other platforms. Much like other Adobe programs, Adobe XD has drag-and-drop tools, allowing users to visually place vector graphic elements on the screen and edit their properties through menus.

**Proposed Time Schedule**

**Task Name Duration**

* + Benchmark 1 2 Weeks
  + Benchmark 2 4 Weeks
  + Benchmark 3 4 Weeks
  + Benchmark 4 2 Weeks
  + Benchmark 5 2 Weeks
  + Integrated Testing 1 Weeks

**Proposed Grading Scheme**

Benchmark 1: 15%;

Benchmark 2: 20%;

Benchmark 3: 20%;

Benchmark 4: 15%;

Benchmark 5: 10%;

Integrated Testing and Report Writing: 10%;

Presentation: 10%

**List of Deliverables**

* Original proposal and presentation file (s)
* Amendments to the proposal
* System architecture diagram(s)
* Appropriately commented source code
* Documentation of project functionality
* Sample output
* Executables and/or projects
* Presentation documents (used to support the presentation of the completed CSC 521 project), including any presentation file(s)
* Project journal
* Project post mortem
* A list of what areas of the proposal (if any) were not completed, including reasons why
* Presentation of the completed project (PowerPoint format)
* User's manual