WEB ENGINEERING:
COMPLEX ENGINEERING PROBLEM (CEP)

Group Members:

Lavesh Kumar	[20SW074]
Muhammad Musadiq Yasin	[20SW078]
Junaid Aslam	[20SW084]
Muhammad Ahsan	[20SW106]

Submitted to Sir,

Mr. Mansoor Samo,
Department of Software Engineering,
MUET, Jamshoro.

Mehran University of Engineering and Technology Jamshoro, Sindh

Assignment Topic:

Complex Engineering Problem - CEP

Group Members:

S. No.	Name	Roll No.
1.	Lavesh Kumar	20SW074
2.	Muhammad Musadiq Yasin	20SW078
3.	Junaid Aslam	20SW084
4.	Muhammad Ahsan	20SW106

Subject: Web Engineering (SW-417)

Teacher: Mr. Mansoor Samo

Department of Software Engineering, Mehran University of Engineering & Technology Jamshoro, Sindh.

Problem Description

Students are asked to develop website project of their own choice. The project development is based on programming skills they have already learned in web engineering or they already know the programming skills used in the website project. Students are asked to submit the report which contain the solution of these queries:

- Provide a clear articulation of the problem your website project aims to solve or the purpose it serves. What motivated you to choose this particular project?
- What programming languages, frameworks, libraries, and tools did you utilize in developing your website project and why?
- Provide a description of the overall architecture and design of your website project.
- Main features and functionalities of your website project. Explain how users interact with your website and what actions they can perform. Sketch diagrams.
- What were the major challenges you encountered during the development process? How did you address or overcome these challenges?
- Possibilities for future enhancements or iterations of your website project. Did you learn any valuable lessons or insights during the development process?

Provide a clear articulation of the problem your website project aims to solve or the purpose it serves. What motivated you to choose this particular project?

The title of our project is "Ecommerce Site," which is a convenient online shopping platform specializing in women's clothing, watches, shoes, sandals, jewelry, and other articles. The motivation behind this choice is to tap into the growing e-commerce market while catering to the specific fashion needs of women.

1. **Convenience**: Providing an easy-to-use platform for customers to shop from anywhere at any time.
2. **Variety**: Offering a diverse range of products to suit different tastes and preferences.
3. **User Experience**: Ensuring a smooth browsing and purchasing journey for enhanced customer satisfaction.
4. **Market Opportunity**: Identifying and addressing gaps in the women's fashion and accessory market to capitalize on untapped potential.

In essence, the project aims to combine convenience, variety, and a seamless user experience to meet the evolving needs of online shoppers in the women's fashion sector.

What programming languages, frameworks, libraries, and tools did you utilize in developing your website project and why?

In developing our website project, "Ecommerce Site," We utilized the following programming languages, frameworks, libraries, and tools:

1. **HTML (HyperText Markup Language):** Utilized as the foundational markup language for structuring the website's content, providing the backbone of its layout and organization.
2. **CSS (Cascading Style Sheets):** Employed to enhance the visual presentation and aesthetics of the website, including styling elements such as colors, fonts, and layouts.
3. **JavaScript (JS):** Integrated to add interactive functionality and dynamic features to the website, enhancing user engagement and experience.
4. **Bootstrap:** Leveraged as a front-end framework to expedite the development process and ensure responsiveness across various devices and screen sizes. Bootstrap's pre-built components and grid system facilitated efficient design implementation.
5. **jQuery:** Used to simplify JavaScript programming tasks and streamline DOM manipulation, enabling smoother client-side scripting and interaction.
6. **PHP (Hypertext Preprocessor):** Implemented for server-side scripting to handle dynamic content generation, database interactions, and backend logic. PHP was chosen for its robust capabilities in web development and its compatibility with various databases.
7. **phpMyAdmin (Database Management Tool):** Employed for managing the MySQL database associated with the website. phpMyAdmin facilitated database administration tasks such as creating, querying, and modifying database structures and contents.
8. **Visual Studio Code (VS Code):** Selected as the primary integrated development environment (IDE) for its lightweight yet powerful features, including syntax highlighting, debugging support, and an extensive library of extensions, enhancing productivity and code quality.
9. **XAMPP (Cross-Platform Apache, MySQL, PHP, and Perl):** Utilized as a local server environment to facilitate website development and testing on a personal computer. XAMPP provided the necessary server components and services for running PHP-based applications locally, ensuring a seamless development workflow.

By leveraging these programming languages, frameworks, libraries, and tools, we aimed to create a robust, visually appealing, and user-friendly e-commerce website that effectively meets the project objectives and user expectations.

Provide a description of the overall architecture and design of your website project.

The architecture and design of my website project, "Ecommerce Site," follow a straightforward and intuitive approach:

- **Frontend Structure:**
  - The frontend is built using HTML, CSS, and JavaScript (along with Bootstrap and jQuery for enhanced styling and interactivity).
  - HTML provides the structure of the web pages, defining elements such as headers, navigation bars, product listings, and footers.
  - CSS styles the HTML elements, controlling aspects like colors, fonts, layout, and responsiveness to ensure a visually appealing and user-friendly interface.
  - JavaScript adds dynamic behavior to the frontend, enabling features such as image sliders, dropdown menus, form validations, and interactive elements for improved user engagement.

- **Backend Structure:**
  - The backend utilizes PHP for server-side scripting, handling tasks such as user authentication, product management, and database operations.
  - PHP scripts interact with the MySQL database through SQL queries, retrieving and updating data related to products, user accounts, orders, and other relevant information.
  - The website architecture follows a Model-View-Controller (MVC) pattern, separating concerns between data (Model), presentation (View), and application logic (Controller) to ensure maintainability and scalability.

- **Database Management:**
    - The MySQL database stores essential data for the website, including product details, user accounts, order history, and transaction records.
    - Tables within the database are structured to maintain relational integrity, ensuring efficient data storage and retrieval.
    - phpMyAdmin serves as the administration tool for managing the database, allowing for easy manipulation of tables, queries, and records.

- **Server Environment:**
    - The website is deployed on a local server environment provided by XAMPP, which includes Apache as the web server, MySQL as the database server, and PHP for server-side scripting.
    - XAMPP enables seamless testing and development of the website on a personal computer before deployment to a live server.

- **Overall Interaction Flow:**
    - Users interact with the website through their web browsers, navigating through product categories, viewing item details, adding products to their cart, and proceeding to checkout.
    - Backend scripts handle user authentication, session management, product catalog management, and order processing, ensuring a smooth and secure user experience.

In summary, the architecture and design of the "Ecommerce Site" project prioritize simplicity, functionality, and usability, leveraging a combination of frontend and backend technologies to deliver a seamless online shopping experience for customers.

Main features and functionalities of your website project. Explain how users interact with your website and what actions they can perform. Sketch diagrams.

Here's a breakdown of the main features and functionalities of the website project, categorized into admin mode and customer mode:

Admin Mode:

1. **Authentication and Authorization:**
   - Admin logs in using email and password.
   - Authentication and authorization processes ensure security and access control.
2. **Admin Dashboard:**
   - Upon login, the admin is presented with a dashboard displaying key metrics:
     - ✓ Number of products
     - ✓ Pending orders
     - ✓ Completed orders
     - ✓ Completed shipments
     - ✓ Active customers
     - ✓ Subscribers
     - ✓ Available shippings
     - ✓ Top, mid, and end categories
3. **Site Management:**
   - Ability to change site banner, logo, contact details, payment settings, and footer details.
   - Manage home settings, sliders, FAQ, and social media links for site customization.
4. **Product Management:**
   - Add, delete, and edit products with details such as name, price, size, quantity, description, and images.
   - View sales details and manage product inventory.

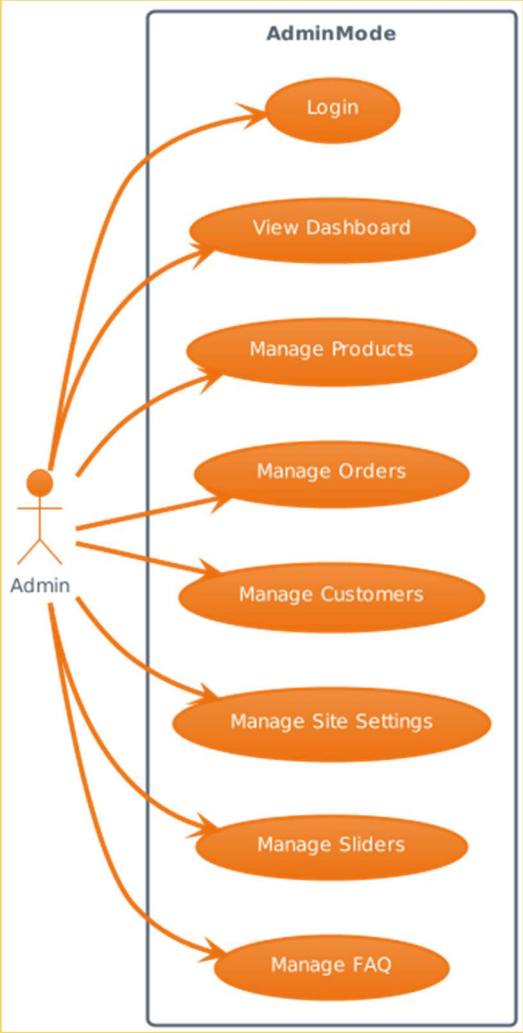
5. **Order Management:**
    - Process orders, manage shipments, and handle customer inquiries regarding orders.
    - Set policies for returns, exchanges, shipping, and payment.
6. **Customer Management:**
    - Manage customer accounts, including registration, authentication, and authorization.
    - View and edit customer details, such as shipping and billing addresses.
7. **User-Friendly Interface:**
    - Admin actions are performed through a user-friendly dashboard, facilitating ease of use and navigation.

Customer Mode:

1. **Authentication and Registration:**
    - Customers can sign in or sign up for an account to access personalized features.
    - Authentication ensures security and access control for customer accounts.
2. **Browsing and Shopping:**
    - Browse products by categories, view latest and popular items.
    - Add products to the shopping cart for purchase.
3. **Checkout Process:**
    - Proceed to checkout, providing shipping and billing details.
    - Choose preferred payment method and complete the transaction securely.
4. **User Interaction:**
    - Contact the website through contact forms, subscribe to newsletters, and access social media links for further engagement.

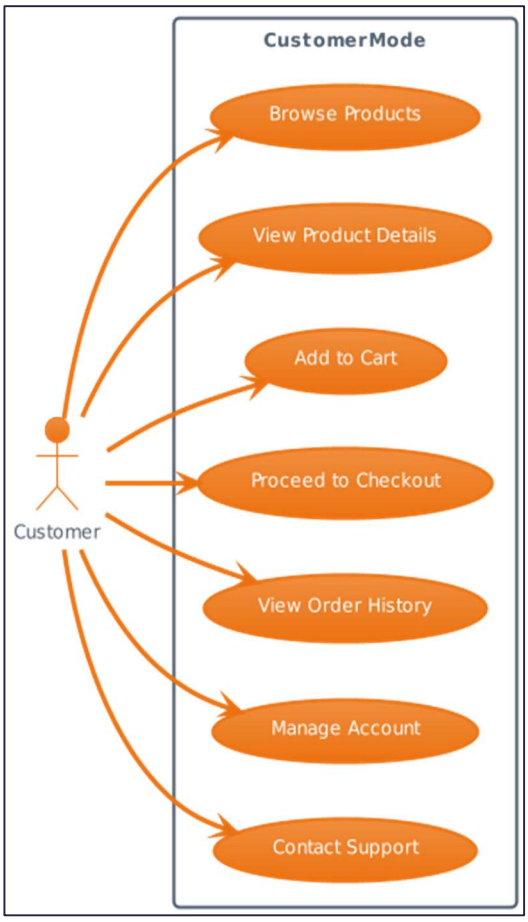
Admin Interaction Flow:

Use Case Diagram

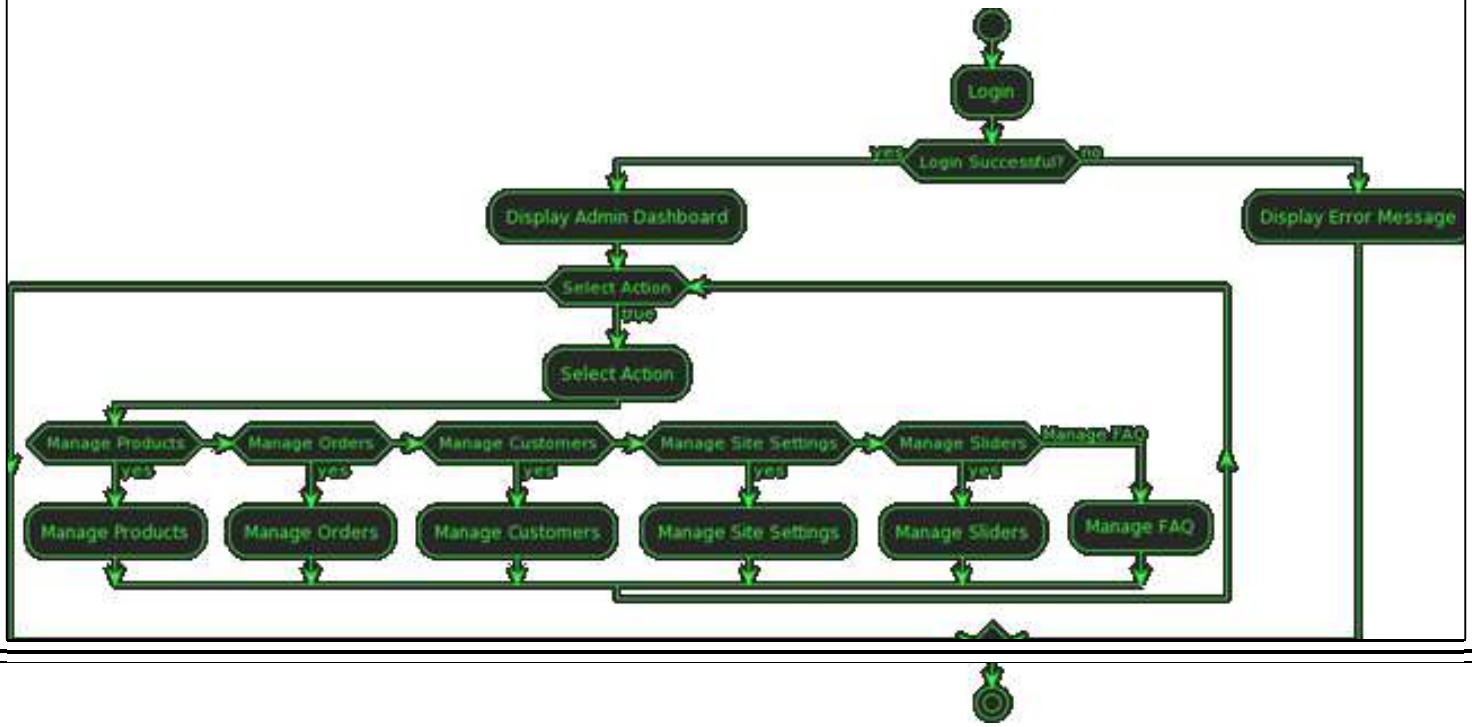


Customer Interaction Flow:

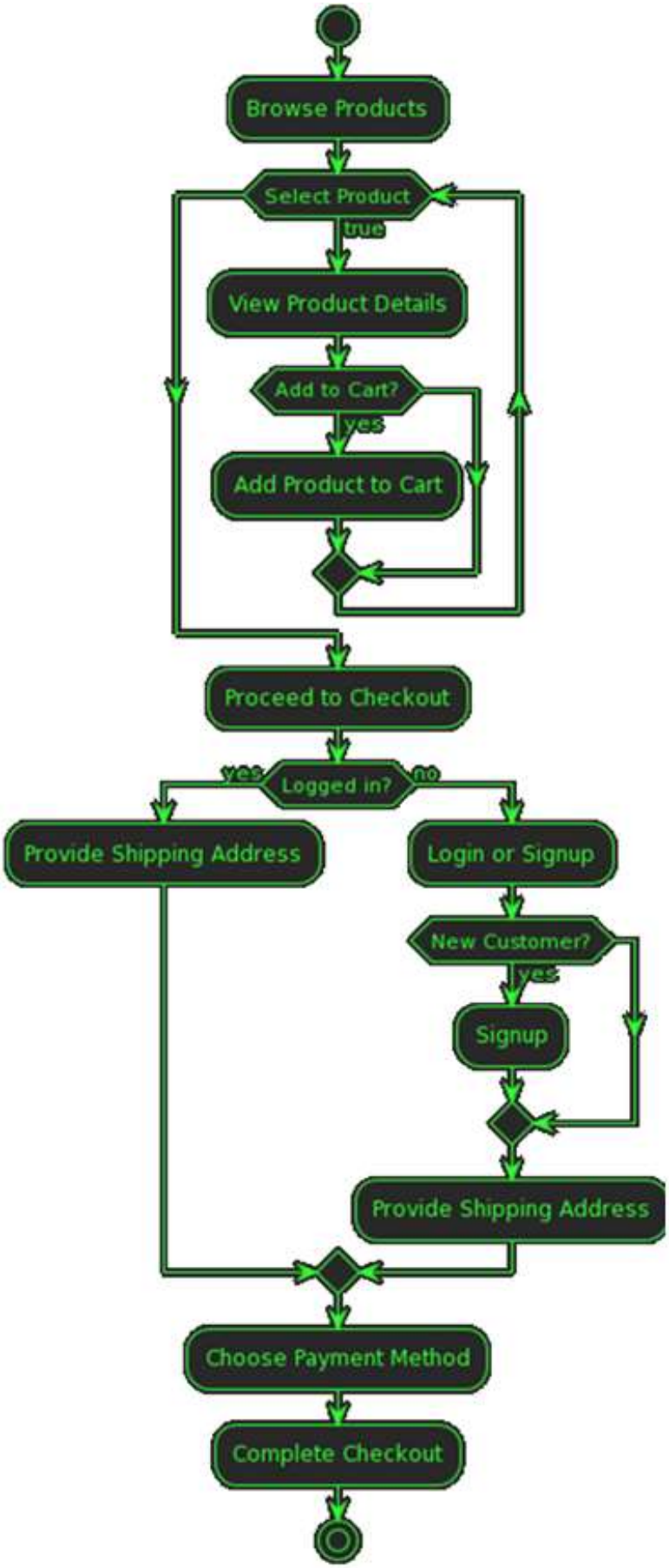
Use Case Diagram



Admin Interaction Flow: Activity Diagram



Customer Interaction Flow: Activity Diagram



What were the major challenges you encountered during the development process? How did you address or overcome these challenges?

During the development process of the website project, several major challenges were encountered, which required careful consideration and effective problem-solving strategies to overcome. Some of the general challenges faced included:

1. **Complexity of E-commerce Functionality:**
    - **Challenge:** Implementing comprehensive e-commerce functionalities such as product management, order processing, and payment integration can be complex and time-consuming.
    - **Solution:** Break down the development tasks into smaller, manageable components. Utilize frameworks, libraries, and existing solutions for e-commerce platforms to streamline the development process. Regularly review and test each functionality to identify and resolve any issues promptly.
2. **Port Conflict**
    - **Challenge:** During the development process, we encountered a challenge when attempting to run XAMPP and access phpMyAdmin. It became apparent that we were facing a port conflict issue due to the port already being used by another service, such as Oracle.
    - **Solution:** To overcome this obstacle, we followed these steps:
        1. **Identified Port Conflict**: Noticed the port conflict issue when starting XAMPP and trying to access phpMyAdmin.
        2. **Found Available Port**: Identified an available port on our system that was not being utilized by any other service. We opted for port 8080 as it was available.
        3. **Configured Apache and MySQL Ports in XAMPP**: Accessed the XAMPP control panel and navigated to the Apache and MySQL configurations. Changed the default ports (e.g., Apache port 80 to 8080 and MySQL port 3306 to 3307) to the available ports we identified.
        4. **Updated phpMyAdmin Configuration**: Accessed the phpMyAdmin configuration file located in the XAMPP installation directory. Updated the configuration to reflect the new MySQL port (e.g., changed 'localhost:3306' to 'localhost:3307').
        5. **Restarted Services**: Saved the changes made to the configurations and restarted both Apache and MySQL services in the XAMPP control panel.
        6. **Accessed phpMyAdmin**: Successfully accessed phpMyAdmin through the updated port (e.g., http://localhost:8080/phpmyadmin) in our web browser.

By addressing the port conflict in this manner, we were able to seamlessly continue our development process without any further hindrance from the conflicting services.

Possibilities for future enhancements or iterations of your website project. Did you learn any valuable lessons or insights during the development process?

Future enhancements for the website project include:

1. **Personalized Experience:** Make product recommendations based on user preferences and history and allow users to create wish lists.
2. **Improved Search:** Add advanced filtering options and predictive search to help users find products more easily.
3. **Social Integration**: Enable sharing on social media and allow social login for easy access.
4. **Mobile App:** Develop a mobile app for a better shopping experience with features like push notifications.
5. **Global Reach:** Support multiple languages and currencies to cater to a wider audience.
6. **Performance Boost:** Optimize loading times and responsiveness by implementing caching and image optimization.
7. **Analytics:** Use advanced tools for better insights into user behavior and sales trends.
8. **More Payment Options:** Integrate additional payment gateways for flexibility, such as digital wallets.

During development, we learned the importance of planning, testing, user feedback, staying updated, and teamwork for success.

[THE END]