**Ngimbi Misuna Olivier**

**ID: 22371**

**Web Technology Final Project Documentation**

**Project Name: E-Commerce Web Application**

1. **Project Requirements**

The E-Commerce web application project aims is to create a robust and user-friendly online web application platform for conducting electronic commerce activities. The purpose of the project is to provide a convenient and efficient way for customers to browse, search, purchase, and manage products or services online. The application will also include features for order processing, payment processing, and customer support.

* **Expected Outcomes:**

1. **User Registration and Authentication:** Users will be able to create an account, log in, and manage their profile information. This will allow them to track their orders and maintain a personalized shopping experience.
2. **Product Catalog:** The application will have a comprehensive product catalog with proper categorization and search functionality. Users will be able to browse through different categories, view product details, and read reviews.
3. **Shopping Cart:** Users will be able to add products to their shopping cart, modify quantities, and remove items as needed. The shopping cart will provide a seamless and intuitive experience for users to manage their purchases.
4. **Order Management:** The system will provide an order management feature that allows users to track the status of their orders, view order history, and request returns or exchanges.

* **Constraints or Limitations:**

1. **Security:** The application must prioritize security to protect user information, payment details, and prevent unauthorized access or data breaches.
2. **Performance:** The web application should be scalable and capable of handling a large number of concurrent users without significant performance degradation.
3. **Usability and User Experience:** The application should have an intuitive and user-friendly interface, ensuring a smooth and enjoyable shopping experience for users.
4. **Project Plan**
5. **Project Scope:**

* Develop a fully functional E-Commerce web application with user management, product catalog, shopping cart, checkout, order management, customer support, and administrative features.
* Implement responsive design for optimal user experience across different devices.
* Integrate secure payment processing and shipping options.
* Ensure compliance with legal and regulatory requirements.
* Develop analytics and reporting functionalities for business insights.

1. **Timeline:**

The timeline provided below is a general estimation and may vary based on the project's complexity and resource availability.

**Phase 1: Project Initiation and Planning:**

* Define project scope and objectives.
* Identify key stakeholders and gather requirements.
* Create a detailed project plan and timeline.

1. Resources:

The resources used to accomplish the project include:

**Java Programming language:** responsible for implementing the application's features and functionalities.

**Spring Boot framework**, that handles server-side development, database integration, and API implementation.

**HTML, CSS, AJAX and JavaScript** to handle the user interface and client-side development.

**MySQL, PostgreSQL** is about design, configuration, optimization, and security. Proficiency in SQL and experience with relational databases.

1. **Database Schema**

These are the tables created for the E-Commerce Project:

* **Table: Product**
* **Table: Category**
* **Table: Coupon**
* **Table: ShoppingCart**
* **Table: Carousel**
* **Table: Cartltem**
* **Table: WishList**
* **Table: WishListltem**

1. **User Documentation**

This user documentation will guide you through the process of using our E-Commerce Web Application effectively. Please follow the instructions below:

1. **Accessing the Application:**

The first step is to lunch the IDE Application(Eclipse or Spring Tool suite 4) and the lunch the main class program of the application

Open your web browser and enter the URL of our E-Commerce web application. You will land on the homepage, where you can start exploring products and making purchases.

1. **User Registration and Login:**

To access personalized features, you need to create an account. Click on the "Sign Up" or "Register" link on the homepage. Provide the required information such as name, email address, and password. After successful registration, you can log in using your credentials on the login page.

1. **Browsing and Searching for Products:**

Once logged in, you can start browsing and searching for products. Use the navigation menu or category links to explore different product categories. You can also use the search bar to find specific products based on their name or description.

1. **Product Details:**

Click on a product to view its detailed information. Read the product description, specifications, and customer reviews. If interested, add the product to your shopping cart.

1. **Shopping Cart:**

To access your shopping cart, click on the "Cart" or "My Cart" link. In the shopping cart, you can view the products you have added, adjust quantities, and remove items if necessary. Review the total cost, including any applicable taxes or shipping charges.

1. **Technical Documentation**
2. **Architecture Overview:**

Our E-Commerce web application follows a layered architecture that separates the presentation layer, business logic layer, and data persistence layer. The main components of the application architecture are as follows:

Presentation Layer: Handles the user interface and interaction with the application. It includes HTML, CSS, JavaScript, and Ajax.

Business Logic Layer: Contains the core logic and functionality of the application. It handles user authentication, product management, order processing, and other business-specific operations. This layer is implemented using **Java and the Spring Boot framework.**

Data Persistence Layer: Manages the storage and retrieval of application data. For this E-Commerce Web Application we use relational database management system ( MySQL) to store data related to users, products, orders, and other entities.

1. **Technology Stack:**

**Backend:**

**Java:** A robust and widely used programming language for implementing the business logic layer.

**Spring Boot:** A powerful framework that simplifies the development of Java applications, providing features like dependency injection, MVC architecture, and security.

**Spring MVC:** A module of the Spring framework that handles the web layer, routing requests, and managing controllers.

**Spring Security:** A module of the Spring framework that provides authentication and authorization functionality to secure the application.

**Hibernate or Spring Data JPA:** Libraries for object-relational mapping (ORM) to manage database operations and interact with the data persistence layer. Maven or Gradle: Build tools for managing dependencies and building the project.

Frontend:

**HTML, CSS, JavaScript and Ajax:** Standard web technologies used for structuring web pages, styling, and client-side scripting.

**Database:**

**MySQL relational databases:** Used for storing and managing application data.

Other Tools and Technologies:

**IDE:** Integrated Development Environment like Eclipse, IntelliJ IDEA, or Visual Studio Code for coding and debugging.

1. **Application Workflow:**

The typical workflow of the E-Commerce web application involves the following steps:

**User Registration and Login:**

Users register by providing their information and creating an account.

User authentication is handled using Spring Security, ensuring secure access to the application's features.

**Product Management:**

Admin users can manage product details, including adding, updating, and deleting products.

Product information, such as name, description, price, and category, is stored in the database.

**Shopping Cart and Checkout:**

Users can add products to their shopping cart while browsing.

The shopping cart stores the selected products and quantities.

During checkout, the user provides shipping details and selects a payment method.

The order is processed, and the payment is verified using external payment gateways.