Retrieving Product Details Using Product ID - Servlet-based Application

Git link :- https://github.com/santhoshsai4517/Simplilearn_Phase2_2.14

Introduction

In the context of developing an e-commerce web application, a critical module is required for administrators to retrieve product information based on a specific product ID. To address this need, we will create a servlet-based application that allows users to input a product ID, validate it, and subsequently retrieve product details from a MySQL database. This process involves the use of HTML for user interaction, JDBC for database connectivity, and Apache Tomcat as the web server. The following will outline the steps involved in implementing this functionality.

Requirements:

1. **Integrated Development Environment (IDE):** Utilize Eclipse as the chosen Integrated Development Environment (IDE) for coding and project management.

2. **Web Server:** Set up Apache Tomcat as the web server to host and run the servlet-based application.

3. **Database Connectivity:** Use the MySQL Connector for JDBC functionality to interact with the MySQL database that stores product information.

Step-by-Step Implementation:

1. Create an HTML Form:

- Develop an HTML page that includes a form element to input the product ID. This form will serve as the user interface for input.

2. Set up JDBC:

- Configure JDBC (Java Database Connectivity) to enable communication between the servlet-based application and the MySQL database. Ensure that the MySQL Connector is included in your project's dependencies.

3. Create a Servlet:

- Develop a Java servlet that will handle the following tasks:

  - Receive the product ID input from the HTML form.

  - Establish a database connection using JDBC to query the product table in the MySQL database.

4. Database Query:

- Write SQL queries within the servlet to search for the product with the given ID in the database.

5. Display Product Details:

- If the product is found in the database, retrieve its details and display them to the user. This involves creating dynamic HTML content within the servlet to present the information.

6. Error Handling:

- Implement error handling in the servlet to display an error message if the product is not found or if there are issues with the database connection or query execution.

7. Deploy Application:

- Deploy the servlet-based application on the Apache Tomcat web server.

Conclusion:

This servlet-based application addresses the requirement of retrieving product information based on a product ID in the context of an e-commerce web application. By following the outlined steps and utilizing the specified tools, including Eclipse, Apache Tomcat, and JDBC with MySQL Connector, you can create a functional solution that enhances the admin backend of the e-commerce platform, enabling efficient product information retrieval.