**Building a Servlet-Based Web Application for Adding Products with Hibernate**

**Introduction:**

In this writeup, I will outline the steps to create a servlet-based web application that allows users to add new products to a MySQL database using Hibernate. This application is a fundamental component of an e-commerce platform and provides a user-friendly interface for adding and managing products. The development process will involve Eclipse IDE, Apache Tomcat, JSP for the front end, Servlets for backend processing, and Hibernate for database management.

**Key Requirements:**

**1. Eclipse IDE:** We will use Eclipse as our Integrated Development Environment (IDE) for building the web application.

**2. Apache Tomcat:** Apache Tomcat will serve as our web server, enabling us to run the web application locally.

**3. JSP Pages:** JavaServer Pages (JSP) will be used to create the user interface for adding products.

**4. Servlets:** Servlets will handle the backend processing, including form submission and data validation.

**5. Hibernate:** Hibernate, a popular Object-Relational Mapping (ORM) framework, will manage the interaction with the MySQL database.

**Step-by-Step Development:**

**1. Create a Dynamic Web Project:**

- We start by creating a Dynamic Web Project in Eclipse and configure it to use Apache Tomcat as the runtime environment.

**2. Create a Database Table:**

- We create a MySQL database table to store product information. The table typically includes fields such as product name, description, and price.

**3. Java Class for Product Entity:**

- We create a Java class representing the product entity (e.g., `Product.java`) and annotate it with Hibernate annotations to map it to the database table.

**4. Hibernate Configuration:**

- We set up Hibernate configuration by creating a `hibernate.cfg.xml` file. This file contains database connection details, dialect, and mapping resources. The mapping resources specify how the `Product` class is mapped to the database table.

**5. JSP Form Creation:**

- We design a JSP page (e.g., `addProduct.jsp`) that contains a form for adding new products. This form collects product details such as name, description, and price. Proper validation can be added as needed.

**6. Servlet for Adding Products:**

- We create a servlet (e.g., `AddProductServlet.java`) to handle the form submission.

- In this servlet, we retrieve the form data, create a `Product` object, and use Hibernate to store the product in the database.

**7.HibernateUtil Class:**

- We create a `HibernateUtil` class to manage the Hibernate `SessionFactory`. This class is responsible for building the `SessionFactory` based on the Hibernate configuration.

**8.Deployment and Execution:**

- We deploy the web application to Apache Tomcat.

- Users can access the `addProduct.jsp` page to add new products, and upon submission, the data is validated and stored in the database using Hibernate.

- A `productAdded.jsp` page confirms the successful addition of the product and provides a link to return to the home page.

**Conclusion:**

By following these steps and using the mentioned technologies, we have created a robust servlet-based web application for adding products to a MySQL database. This application provides a seamless user experience for adding products and effectively leverages Hibernate to manage database interactions. Future enhancements may include additional features such as product editing, listing, and searching to create a fully functional e-commerce platform.