

Project Overview: Servlet-Based User Login Validation Web Application

This project involves the development of a servlet-based web application designed to handle user authentication, providing a secure and user-friendly login experience. The application includes several key components such as a login page, validation of user credentials, a dashboard for authenticated users, and the ability to log out. The project is built using Eclipse as the Integrated Development Environment (IDE), Apache Tomcat as the web server, HTML for front-end presentation, and Java Servlets to manage server-side logic.

Project Components and Features:

1. Index Page:

The project begins with the creation of a simple yet effective login page. This page serves as the entry point for users to access the application.

2. Servlet for Login Validation:

- A Java Servlet, named `LoginServlet`, is responsible for handling the login validation process.
- It receives user input (email and password) from the login page and compares it against predefined, hard-coded values email is 'user@example.com' and password is 'password' for demonstration purposes.
- If the input matches the predefined values, a new session is created, and the user is redirected to the dashboard.
- In case of invalid login attempts, users are redirected to an error page.

3. Dashboard Servlet:

- The `DashboardServlet` manages the dashboard page, which is accessible to authenticated users.
- It uses the session management feature to determine if a user is logged in. If a valid session exists, the dashboard is displayed.

- The dashboard page may include user-specific information, such as their email address, for a personalized experience.
- Users who are not logged in are redirected back to the login page.

4. Logout Functionality:

- A separate servlet, `LogoutServlet`, is dedicated to handling the logout process.
- When users choose to log out, the servlet invalidates their session, effectively logging them out of the application.
- After logout, users are redirected to the login page, ensuring secure session management.

5. Error Page:

- An HTML error page is created to handle invalid login attempts.
- Users are presented with a clear error message indicating that their login credentials are invalid.
- The error page also provides a convenient link for users to return to the login page and make another attempt.

6. Deployment Descriptor (web.xml):

- The `web.xml` file is configured to manage servlet mappings and error page redirection.
- It ensures that the servlets are properly mapped and handles 404 errors by directing users to the error page.

7. Deployment and Access:

- The final step involves deploying the web application on the configured Apache Tomcat server.
- Users can access the application via a web browser using the appropriate URL (e.g., `http://localhost:8080/UserLoginValidation/login.html`).

In summary, this project provides a complete solution for user login validation in a web application. It incorporates user-friendly front-end elements, secure server-side logic with Java Servlets, and effective session management. Users can log in, access a personalized dashboard upon successful login, receive informative error messages for invalid login attempts, and securely log out of the application.