

Step-by-Step Document of Project

Step 1: I created my maven project and pieced together the pom.xml file with the required add-on, dependencies, and plugins.

Step 2: I created the web.xml, index.jsp files and put the web.xml together to point to the index.jsp file during startup of the program.

Step 3: I created the format of the index.jsp file that led to the need for more .jsp files like my enroll.jsp and Acct.jsp files.

Step 4: I wired up all the .jsp file to point to each other in a closed loop format.

Step 5: I then made my pojo class and called it Client.

Step 6: Next I made the hibernate.cfg.xml file; connected the property tags and mapped it to my pojo class.

Step 7: Then I made my servlet classes; one to add a new user and another to log into the landing Acct.jsp landing page.

Step 8: I created a session object in both servlets. I was shooting to just finish the project I would never do this in real world applications.

Step 9: In the AddServlet I needed to grab the user provided values and store them into an object of type client before saving them to the session object to be ready for the transaction commit.

Step 10: After a successful commit in the AddServlet the Servlet lands on the index.jsp page, or gives an error otherwise.

Step 11: I created a criteria builder that searches the database for the username provided. If found it stores the input in a client object so the getPassword() method can be called to verify the login. If not found it throws a NoResultException and in this exception I simply redirect the user to the index.jsp page with an error message.