Develop a simple web application, using J2EE components, that allows the user to see a list of Products which they can add, delete, and update to an online store. Assume the user is a store admin and NOT the end user. Their day-day duties are basically to login to the application, update products list i.e. add, remove , update. The candidate is expected to demo this using their favorite editor and web server. We prefer to use eclipse and Tomcat. For simplicity sake, do not use any Database. The candidate can use simple servlets or use any MVC like struts. Elaborate styling is not needed. Just use any simple CSS (optional) Focus more on the functionality. Attached are the CSS, Images and the wireframes to guide the developer to create the application.

For demonstration, download the ATT connect participant application beforehand at the following URL:

<http://downloads.uc.att.com/participant/public/V108117/EN/MSI/Student/Student_Auto_Register/Full/ATT_Connect_Participant.msi> (this link is for windows OS)

If any instruction is not clear for any reason, make assumptions and explicitly state it.

Add the WAR file to a public GIT repository and send the link and if possible send us this a day before the interview/demo.

The Product will contain the following properties at a minimum (can add more if needed):

* Product Id
* Product name
* Product Description
* Related Products e.g. if the product is a Shoe, display items such as shoe polish, laces etc.
* Product Image (Optional but this will give you Bonus Marks if you do)

The following functions need to be implemented:

1. Login :
   1. Save the user id and password in a properties file and retrieve it. Hard code two users and their passwords.
   2. The user will go to a Home page after successful login. Invalid login takes the user to Error page.
2. Home page
   1. The initial rendering of the Home page will
      1. Start with an empty list if the user does not yet added any products to their inventory.
      2. Start with the existing products if the user has added any in the **same application server session** (not the user session).
   2. The home page will provide the following functions :
      1. Add Product
      2. Delete Product
      3. Update Product
      4. List products
3. Add Product
   1. Clicking on the “add Product” link / button on home page will take the user to the add products page.
   2. Add products page will provide all the above product fields as inputs. The input for the image could be a URL or a local image.
   3. Validate all the above fields are entered. Do the client side validation using jQuery and provide dialog box for invalid inputs.
   4. Upon successfully adding the product
      1. Update the in-memory List or (HashMap) of the Products
      2. Return to product listings page with the product added. Use **AJAX** to update the product details without refreshing the whole screen.
      3. Display a Generic Error Page if any Unhandled Exceptions are encountered
4. Delete Product
   1. On the Home page , provide a “Delete” link next to each product (or a check box next to each product and a delete button so we can delete multiple)
   2. Upon clicking “delete”
      1. Update the in-memory List or (HashMap) of the products (i.e. delete the selected product from the list)
      2. Return to same products listing page with the product deleted. Use **AJAX** to update the product details without refreshing the whole screen.
      3. Display a Generic Error Page if any Unhandled Exceptions are encountered
5. Update product
   1. On the Home page , provide a “Update” link next to each product
   2. Take the user to the update product page where the product details are pre-populated.
   3. Let the User update whatever fields they want to.
   4. Upon clicking “Update”
      1. Perform input Validation
      2. Update the in-memory List or (HashMap) of the products (i.e. Update the selected product from the list)
      3. Return to same products listing page with the product updated. Use **AJAX** to update the product details without refreshing the whole screen.
      4. Display a Generic Error Page if any Unhandled Exceptions are encountered.
6. **List products page**
   1. **List all the products added so far. Use the following technologies to demonstrate (note you can use the same while coding the add/delete/update functions also)**
   2. **Send the list of products data in JSON format**
   3. **Use jQuery templates (or any template engine) to loop through the products and display them in a tabular format.**
7. Add ample logging to your application. Prefer to use the log4j package.

FYI, here are the Skills we are looking for and pretty much use day-to-day. Highlighted are what are in demand now. Of course we always need strong server side candidates.

* Client side
  + HTML
  + CSS
  + jQuery
  + AJAX
  + JSON
  + Templates (e..g Handle Bar)
  + Mobile View / Hybrid UI
* Server Side
  + MVC Framework (Struts and/or Spring)
  + Servlets
  + JSPs
  + Tiles
  + EJB
  + Web services SOAP Based (both consumption and production)
  + Single sign On communication with other applications
  + Security fundamentals (OWASP)
* Database
  + JDBC
  + PL / SQL