

Network-Based Application Development (ITIS 4166/5166) **Summer 19**

**Assignment - Phase 2**

## (Submission: June 11th, 11:59 PM)

The goal of this phase to use the MVC design pattern to structure your web application. Most developers consider MVC pattern as the best practice. The frontend (JSP pages) developed in phase 1 are used as a starting point for this phase. In this phase students are to develop servlets for the controller and JavaBeans for the business logic, the details are provided below.

**Develop the backend**

For this phase, you’ll implement the backend for Product Management. Use the JSP files that were created in Phase 1.

# Operation

* When the application starts, it displays the Index page. This page contains a link that leads to the Products page that can be used to add, update, or delete products.
* Restrict access to product, products and confirmDelete pages. Only the Index page can be viewed by the anonymous user. Anonymous user is a user who is not logged in. Only allow authenticated users to access to product, products and confirmDelete pages in the Product Management application. Anonymous users should be redirected to Login page.
* One can register on this site by using Signup page. This page should be accessible from the Login page.
* To add a new product, the user selects the Add Product button. This displays the Product page with all text fields empty. Then, the user can fill in the text fields and click on the Update Product button to add the product.
* To edit an existing product, the user selects the Edit link for the product. This displays the Product page with all existing data for the product displayed. Then, the user can edit any entries and click on the Update Product button to update the data for the existing product.
* To delete a product, the user selects the Delete link for the product. This displays the Confirm Delete page. Then, if the user confirms the deletion by selecting the Yes button, the product is deleted, and the Products page is displayed to reflect the new data. If the user selects the No button, the Products page is displayed.

## Specifications

* + Create MembershipServlet class and map it to membership url
  + Create ProductManagementServlet and map it to productManagement url
  + ProductManagementServlet must have a parameter that is called action.
    - If action is equal to displayProducts show products.jsp.
    - If action is equal to addProduct show product.jsp
    - If action is equal to displayProduct show product.jsp
    - If action is equal to deleteProduct show confirmDelete.jsp
  + MembershipServlet must have a parameter that is called action.
    - If action is equal to login show login.jsp.
    - If action is equal to signup show signup.jsp
  + Use a Product object to store the product data. (Use Product.java on Canvas)
  + Use a User object to store the user data. (Use User.java on Canvas)
  + Use server-side validation to validate all user entries. In particular, make sure the user enters code, description, and price for each product. In addition, make sure the product’s price is a valid double value. Moreover, make sure the provided email address has a correct format, and the password length is more than 8 characters.
  + Create an arraylist of users and an arraylist of products and add them to the session (HTTP session).

# Project Phase Submissions

upload a zip file of the NetBeans project to course project phase 2 link on Canvas