# MINI PROJECT 3 READ ME

**Aim:**

To explore the use of Enterprise Java Beans as the business layer of an enterprise application, integrate MP2 JPA domain model with the business layer, and introduce security to the application

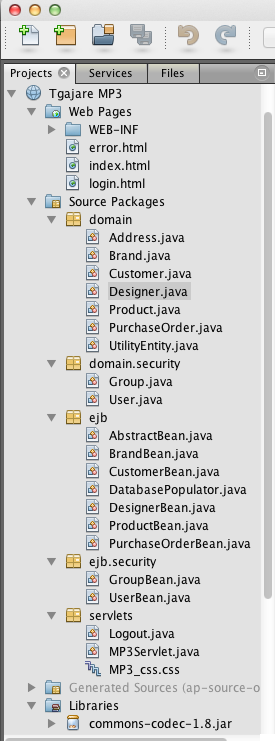
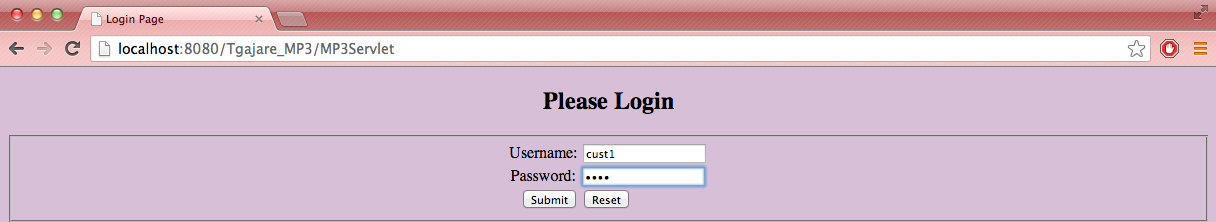
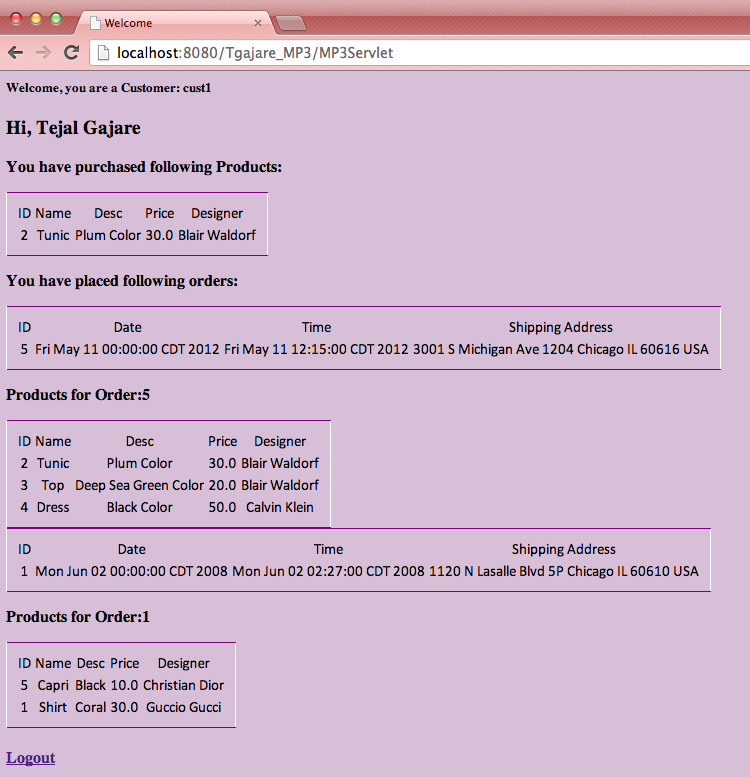
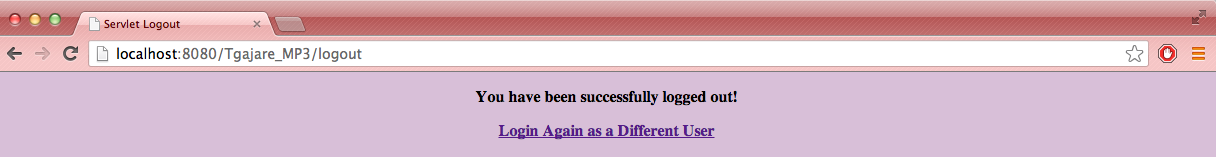
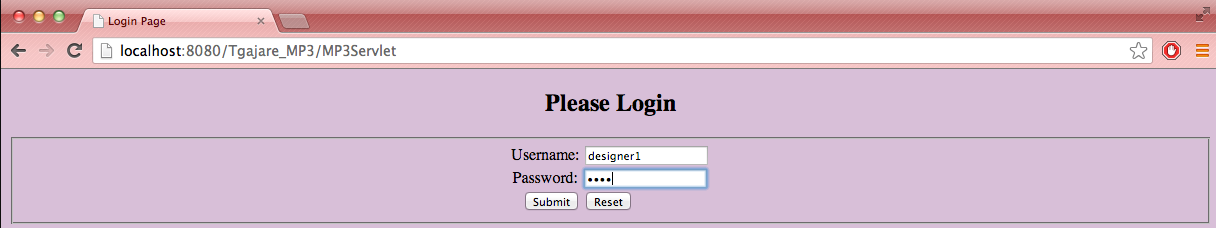
**Project Summary:**

* The MP3 consists of a NetBeans Web Application project, and will incorporate the MP2 domain model as the persistence layer using the EclipseLink JPA implementation and MySQL database connectivity
* Reused the domain entities from MP2 but created an abstract entity – UtilityEntity in order to refactor the common fields
* Made use of entity callbacks such as @PrePersist,@PreUpdate and @PreRemove in the UtilityEntity
* Created the User and Group Entities in order to introduce security
* Created the various bean entities : stateless EJB components to extend the abstract bean entity that provides implementation for the database operations(CRUD)
* Created a DatabasePopulator bean that is the singleton EJB component which kicks off at the start and is responsible for populating the database relations
* Created a new security realm in the glassfish admin console to reflect the user and group database information for providing authentication and authorization
* In the java web application added the descriptor web.xml and specified the roles for different users
* Also specified various constraints
* Role mapping was done by configuring the glassfish.xml
* In order to encrypt the stored password, added the apache codecs common library to the JAR libraries of the project and utilized the DigestUtils.md5Hex method
* Added the NamedQueries in the user entities(Customer and Designer) in order to recover the username of the logged in person
* Created basic html files for login, error and logout purposes
* Also, created servlets to display data based on authenticated user and display sections on the user’s mapped roles
* Provided the logout functionality through the logout servlet
* This MP3 was based on the Instructor’s example from the class

**Design & Development Insights:**

* This project helped to understand the concept of layering and the process of relating the security domain to the business domain
* Prof. Spyrison’s EJB example helped to understand the process very well
* I liked the step-wise approach in building up the entire project due to which I was able to implement all the details without any confusion
* I had trouble implementing the external CSS but due to lack of time could not figure out a way to resolve the issue so used inline styling. The servlet and CSS incorporation was what I did not like. I should have used JSP for better organization of the code, I will implement it in the final project

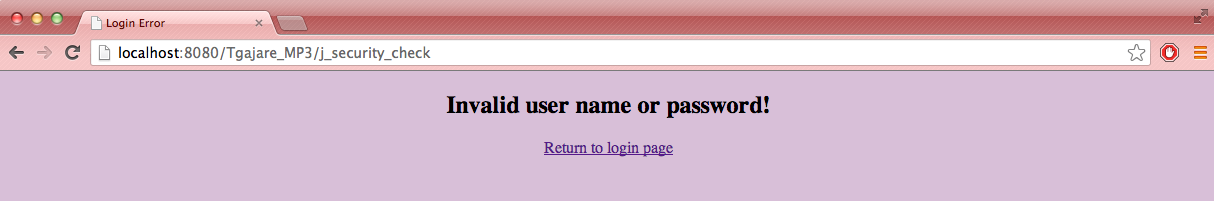
**Requirements:**

* *OS*: Mac OX 10.8
* *Installation*: MySQL WorkBench, NetBeans and JDK 7 for macosx
* *MySQL user*: itmd4515 with password as itmd4515
* Right-Click on the Tgajare MP3 web application in NetBeans and click run
* Once the browser is opened append **“/MP3Servlet”** to the existing URL and press enter
* This should navigate the user to the login page
* *Libraries*: Apache Common Codec 1.8 jar for encryption of password in the database
* **Screen Captures:**
  + ****
* Project structure
  + 
* Login Page after navigating to /MP3Servlet
  + 
* Username entered was : cust1 with password as cust
  + ****
* After clicking on Logout customer is navigated to the above page
  + ****

After clicking on Login as a different user, user is navigated to the login page

****

Logged in as a designer1 with password desi to display the relevant designer data



When the user enters incorrect username/password

**Expected Results:**

Please enter the following usernames and passwords:

*For Customers:*

'cust1' 'cust'

'cust2' 'cust'

'cust3' 'cust'

'cust4' 'cust'

'cust5' 'cust'

*For Designers:*

'designer1' 'desi'

'designer2' ‘desi'

'designer3' 'desi'

'designer4' 'desi'

'designer5' 'desi'