Project Report

|  |  |
| --- | --- |
| Product Name | Advanced Certificate in Web Development |
| Qualification Name (NICF) | NICF-Advanced Certificate in Infocomm Technology (Software & Applications) |
| Product Name | Web Development Foundations |
| Module Name (NICF) | NICF -Web Development Foundations |

|  |  |  |  |
| --- | --- | --- | --- |
| Student name | | Assessor name | |
| Joshua Ho Gwok Hin | |  | |
| Date issued | Completion date | | Submitted on |
| 14 March 2022 | 18 April 2022 | | 18 April 2022 |
|  | |  | |
| Project title | Design, Develop, Implement & Document Community Portal Website | | |

|  |
| --- |
| Learner declaration |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.  C:\Users\hghjo\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Signature.gif  Student signature: Date: 15 April 2022 |

Content

1. Project background
2. Project Objectives
3. UI Requirement Specifications
4. Task 1 – Technical Design
5. Task 2 – Eclipse IDE installation
6. Task 3 – List 3 features in Eclipse IDE
7. Task 4 – Coding & Documentation Standards
8. Task 5 – Views in JSP/HTML
9. Task 6 – Repository Interfaces, Controller, Model & Service Classes
10. Task 7 – Presentation of application with functionality
11. Task 8 – Debugging in Eclipse IDE
12. Task 9 – Demonstrate 3 troubleshooting
13. Task 10 – Propose new feature(s)
14. Task 11 – Junit Test cases in Eclipse IDE

Project Background

“ABC Jobs Pte Ltd” is looking to develop a community portal for Software Programmers /Developers. The scope of the project is to design a community portal similar to Linkedin.com. Users of the portal of the portal can search for other users using various parameters such as First Name, Last Name, Company Name, City and Country. User will be able to view the public profile of users after searching them. The portal allow users to login, request for forgotten password and update their profile information.

The scope of the project in this modules is for development of Spring Website.

The overview of the project is as below:

There are 2 types of users in this community portal. They are

1. Software Programmer
2. Administrator

Software Programmer should be able to perform following functions in the portal

1. Allow the programmers to register in the portal, show a thank you page @send a registration confirmation email.
2. Search and find other programmers after login and view their profile.
3. Provide login page
4. Provide password retrieval functionality.
5. Update their Profile after logging in.
6. Send Messages to each other on the portal
7. Create threads & post replies to a thread
8. Post job opportunities in the portal

Administrator should be able to perform following functions in the portal

1. Administer user data.
2. Send bulk email inviting programmers to register on the community portal

The portal consist of the following key pages

1. Community portal home page
2. Registration page
3. Registration Confirmation page
4. Update Profile page
5. Search users page
6. List search results
7. Public profile page
8. Registration Confirmation email
9. Login page
10. Forget password page
11. Design the forget password confirmation page
12. Send messages
13. Read messages
14. Post in message board
15. List message board
16. Read a thread
17. Post job opportunities
18. List job opportunities & responses

Project Objective

The main objectives of this project is to **develop “ABC Jobs Community Portal” using the Web Module of Spring MVC Framework** which includes features and functionality that comes with Spring Framework’s Inversion of Control(IoC) and Dependency Injection(DI) features.

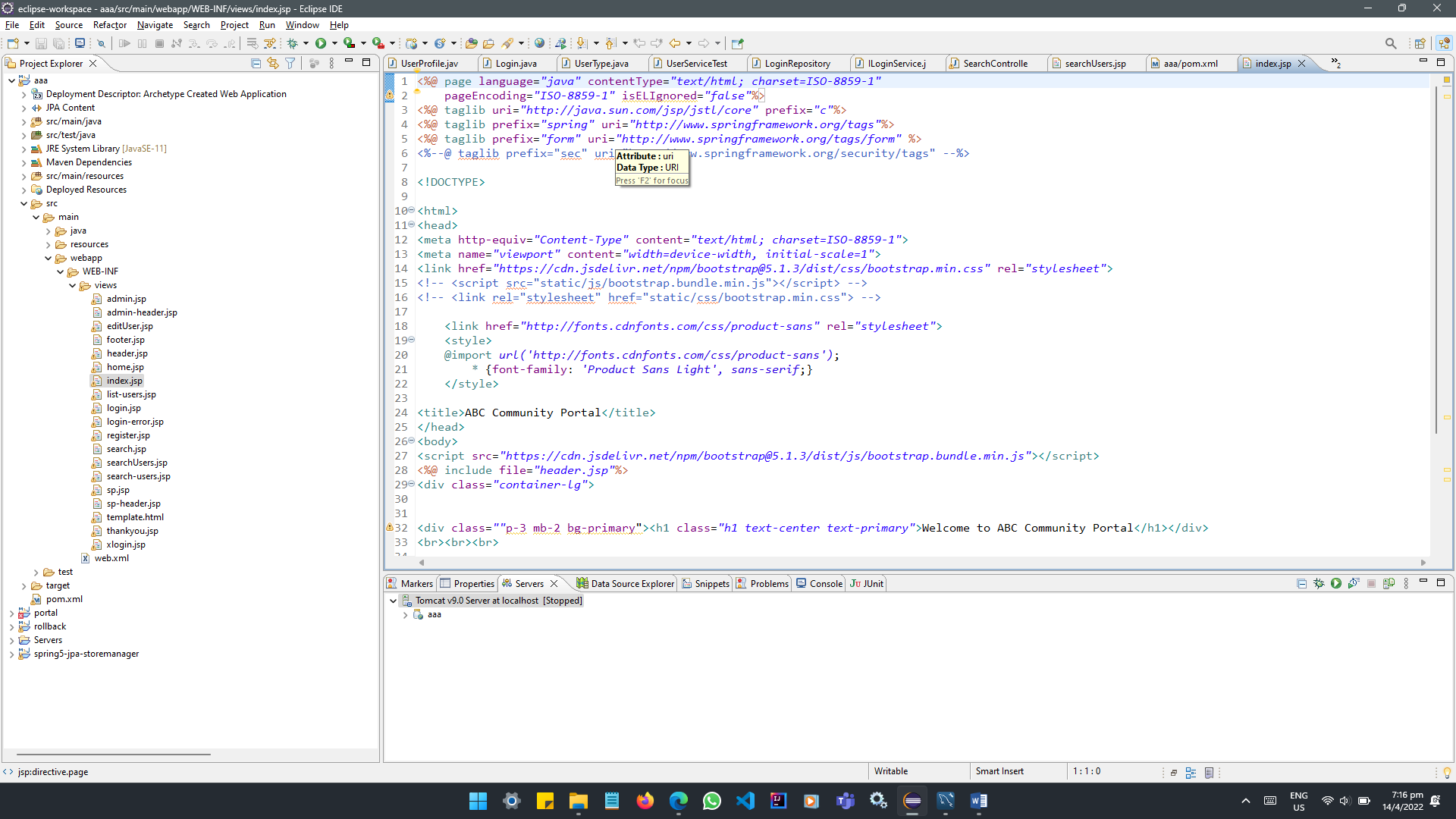
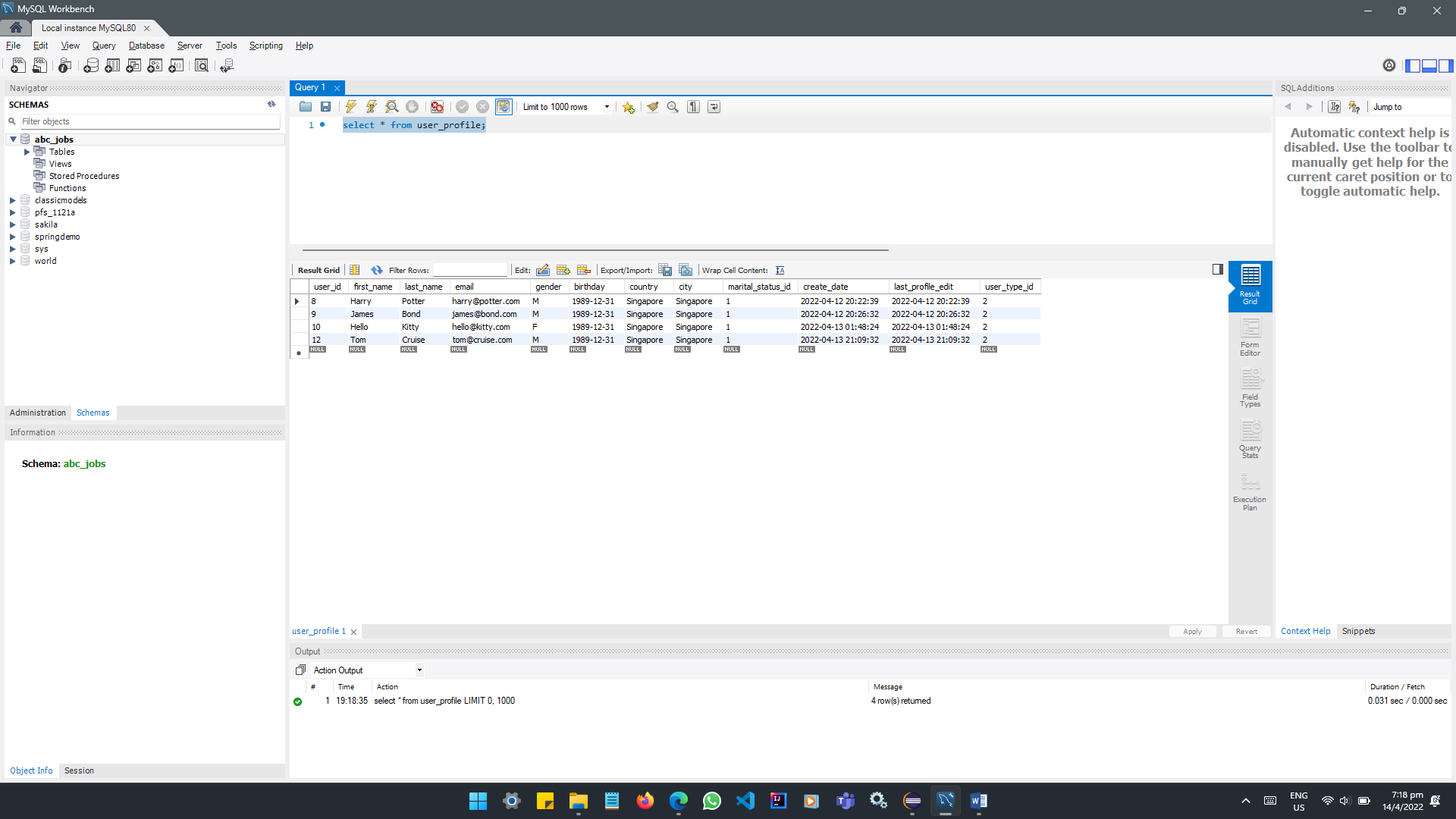
The business goal of this Web Application is to get software programmers to join the portal. This web application allows users to create and apply job listings.

There are 2 groups of users, namely the “Adminstrators” and “Software Programmers”. Software Programmers will be able to perform operations such as register, search users, establish virtual relationship (friend/connection), post threads, create and apply for jobs.

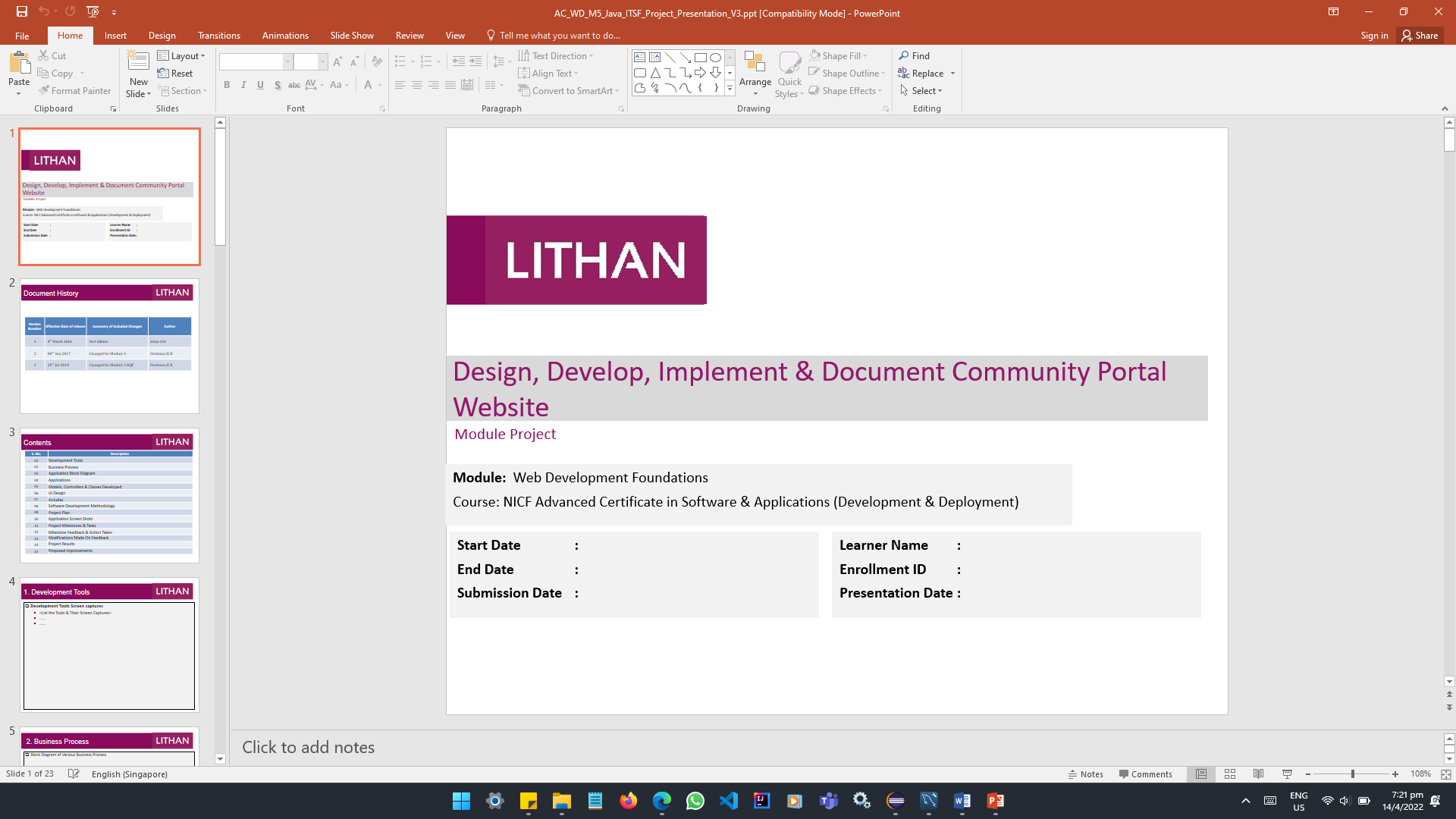
“Adminstrators” have additional access to functions such as sending bulk email and also make changes to delete or edit other “Software Programs” for the purpose of regulating/moderating user pool.

Project Requirements Specifications

Eclipse IDE MySQL Workbench 8.0

MS Word MS PowerPoint

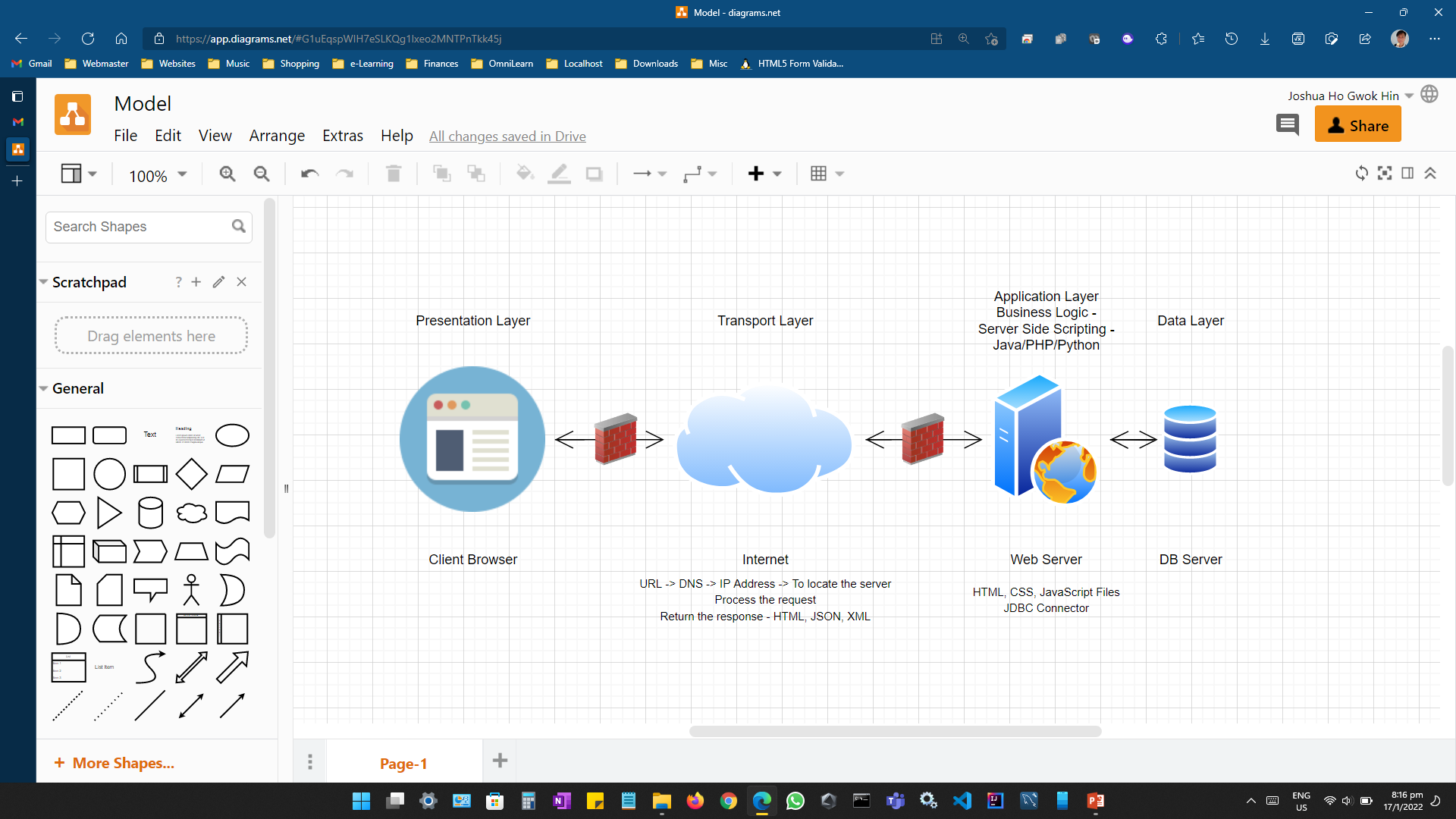
Tools & platform used

* Eclipse IDE - Version: 2021-12 (4.22.0)
* MySQL Workbench 8.0 - Version 8.0.27
* MS Word
* MS PowerPoint
* Tomcat v9.0 Server
* JRE System Library [JavaSE-11]
* Spring - Version 5.1.5.release
* Spring JPA – Version 2.1.5.release
* Hibernate – Version 5.4.1.final
* Junit Jupiter – Version 5.8.2
* Javax Servlet – Version 4.0.1
* JSTL for JSP – Version 1.2.6 (Jakarta)
* ERD - erdplus.com
* JSP, Java, HTML5, JavaScript, CSS3
* Bootstrap – Version 5.1.3

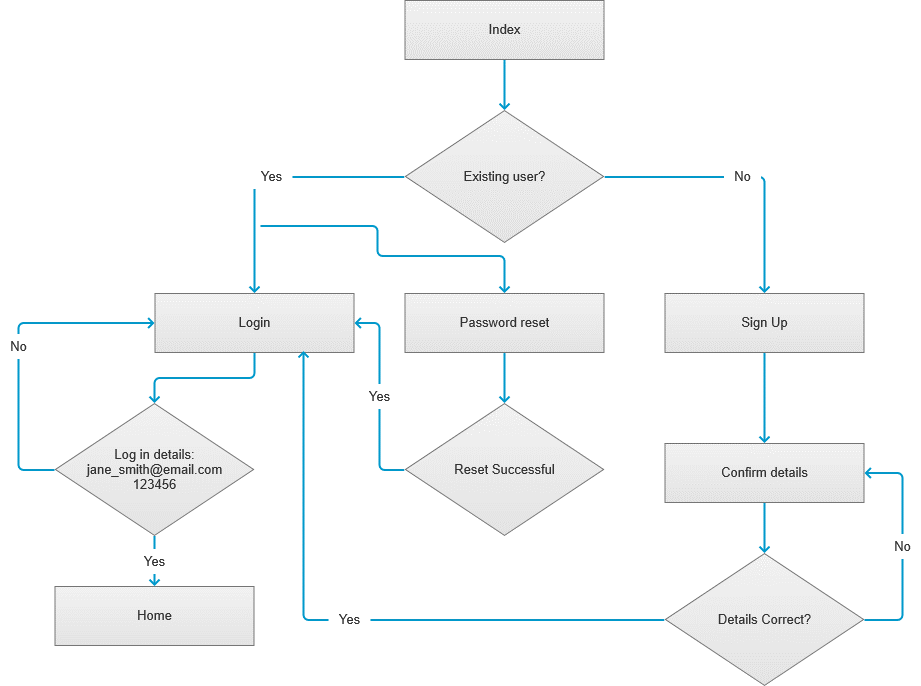
Task 1 – Technical Design

1. Architecture of the system
   1. Draw the Block diagram of the system

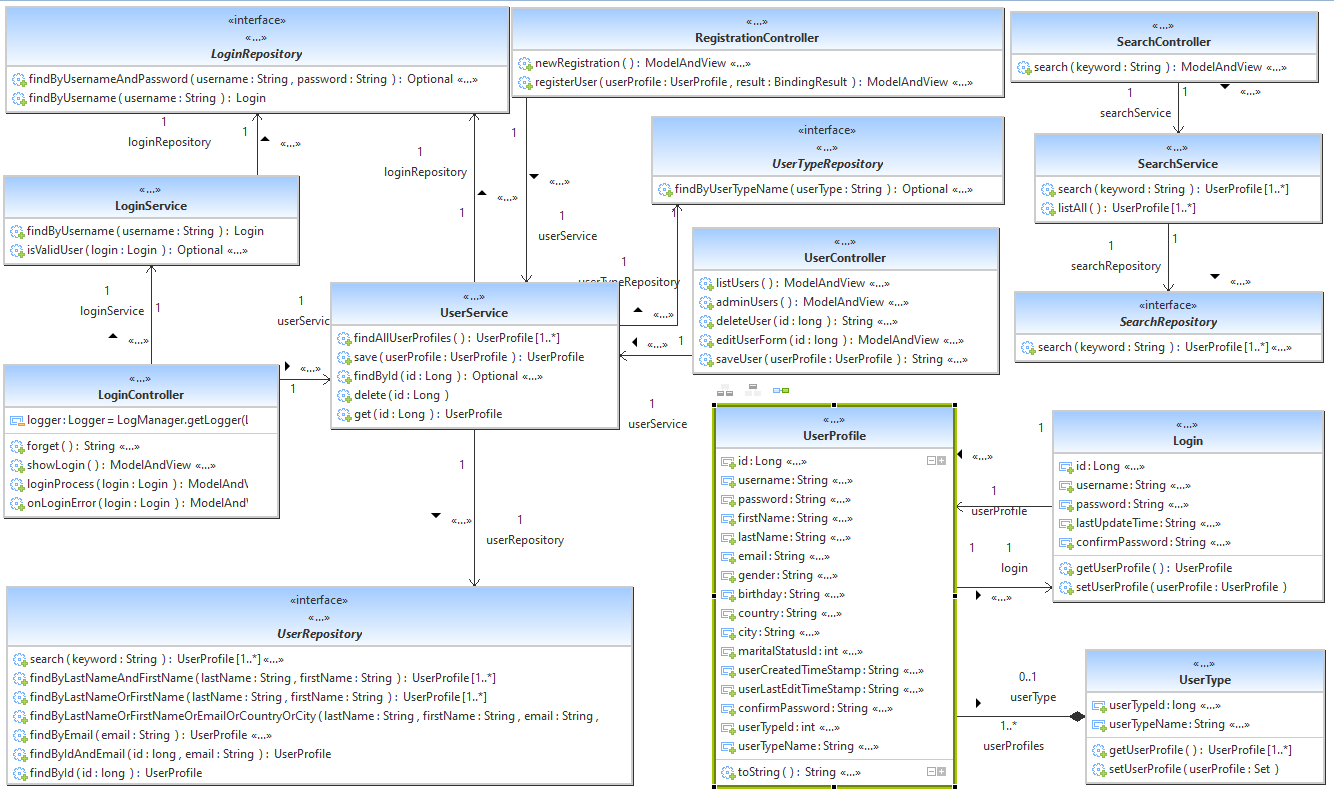
Web Application Architecture



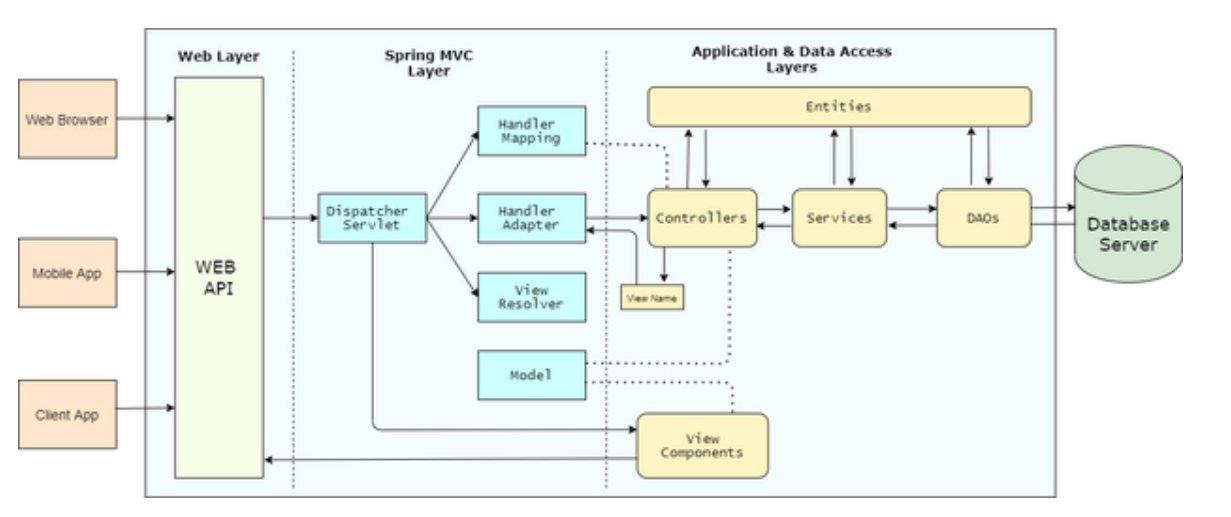
Business Logic Architecture



Class Diagram



Application Architect of MVC Pattern



* 1. Explain the Architecture of the application

|  |  |
| --- | --- |
| Architecture of the application | Senario: User keyed in “Singapore” into the search input field and click on the search button. |
| DispatcherServlet receives the request from the web layer. | The search button triggers action=”searchUserProfile” to appends the query string “/searchUserProfile?keyword=Singapore” to become <http://localhost:8080/aaa/searchUserProfile?keyword=Singapore>  (keyword(key)=Singapore(value)) |
| DispatcherServlet uses HandlerMapping to determine appropriate controller. HandlerMapping reads @RequestMapping annotation on Controller and matches the incoming request URL with a method on the Controller. | @GetMapping("/searchUserProfile ")  public ModelAndView search(@RequestParam String keyword) {  List<UserProfile> result = searchService.search(keyword);  System.*out*.println(result);  ModelAndView mav = new ModelAndView("searchUsers");  // appearInSearchResult is the model attribute that holds the  mav.addObject("appearInSearchResult", result);  return mav;  } |
| DispatcherServlet uses HandlerAdapter to call the method on the Controller selected by HandlerMapping in the previous step. | @GetMapping ("/searchUserProfile ")  public ModelAndView search(@RequestParam String keyword) {  List<UserProfile> result = searchService.search(keyword);  System.*out*.println(result);  ModelAndView mav = new ModelAndView("searchUsers");  mav.addObject("appearInSearchResult", result);  return mav;  } |
| Controller uses appropriate Services to execute the business logic. Controller stores the processing result in Model object and returns the logical name of the view to HandlerAdapter. | @Transactional  public class SearchService {  @Autowired  SearchRepository searchRepository;  public List<UserProfile> search(String keyword) {  return searchRepository.search(keyword);  }  } |
| DispatcherServlet uses ViewResolver to resolve the logical name of the view to appropriate View Component.  DispatcherServlet forwards request to the View component.  View component uses the processing result in the Model object to render the page. | searchUses.jsps  </tr><c:forEach items=*"*${appearInSearchResult}*"* var=*"var"*><tr>  <td>${var.firstName}</td>  <td>${var.lastName}</td>  <td>${var.email}</td>  <td>${var.gender}</td>  <td>${var.birthday}</td>  <td>${var.country}</td>  <td>${var.city}</td>  <td>  </c:forEach> |

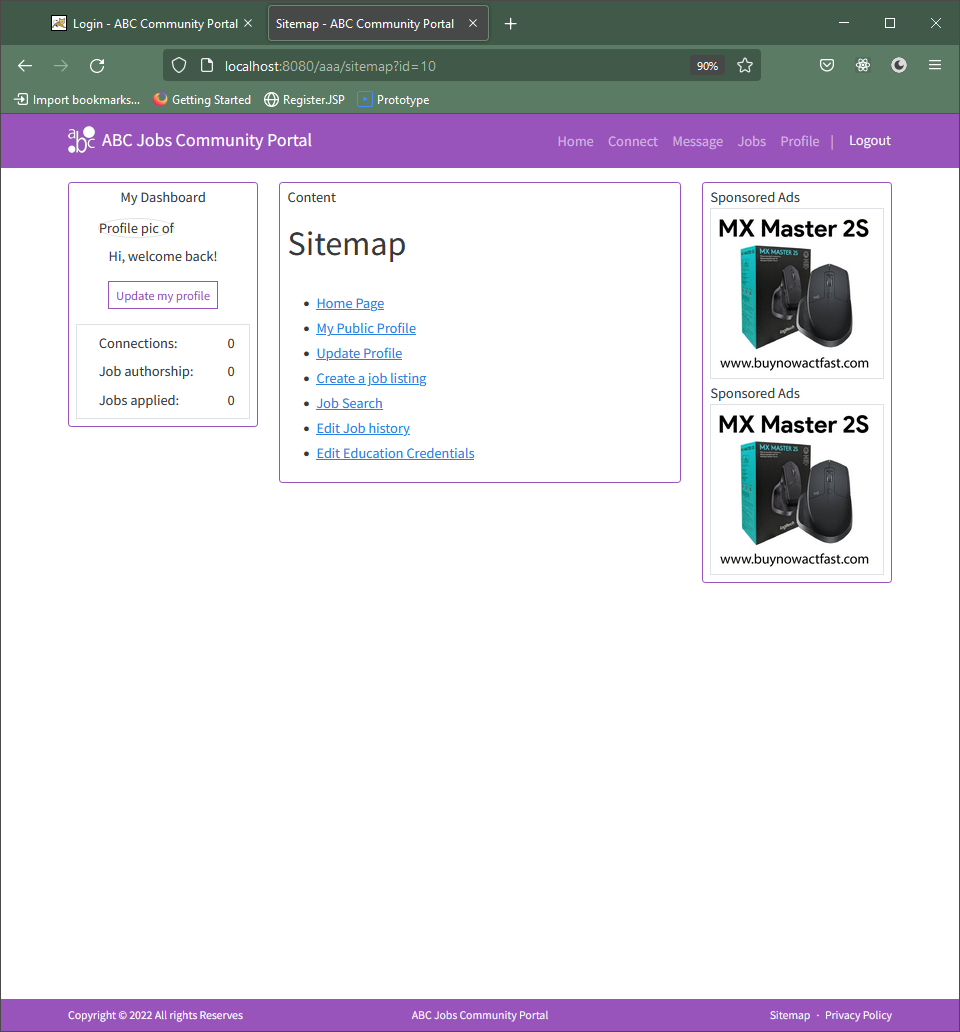
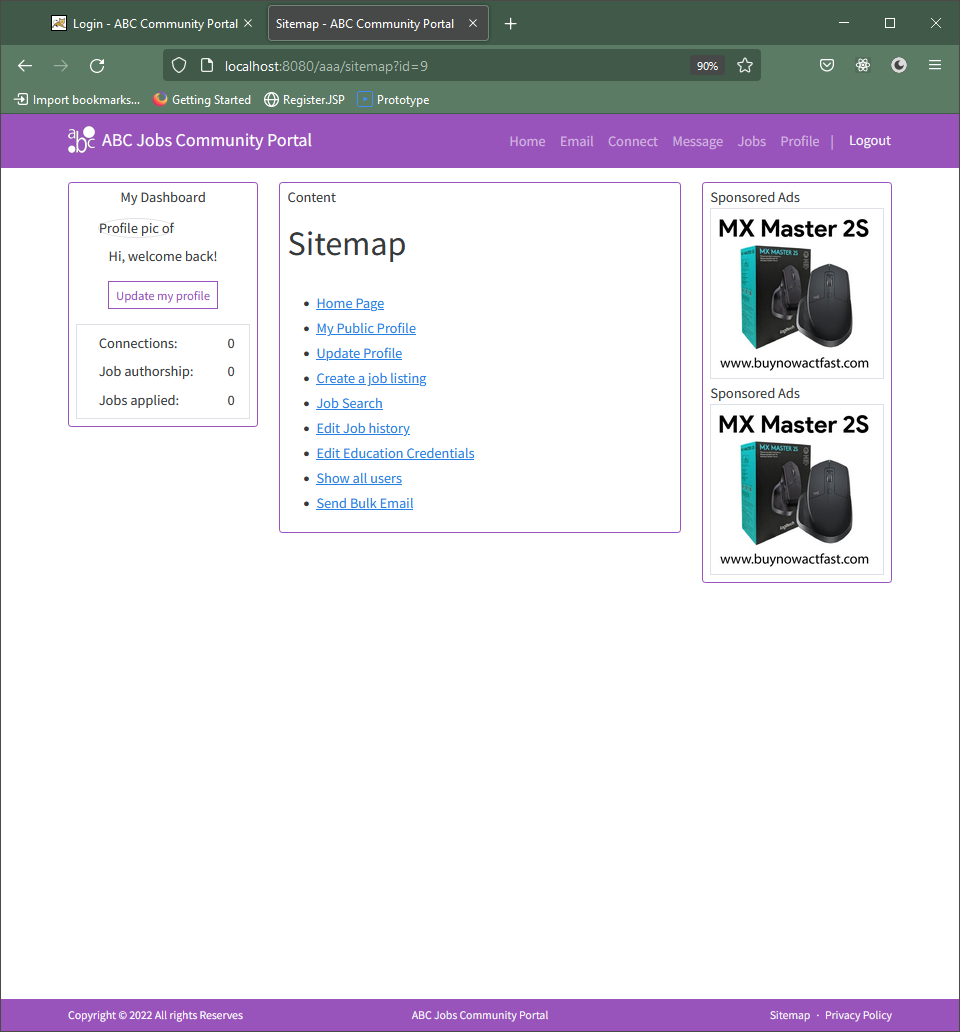
Brief general description of a Web Application

|  |  |
| --- | --- |
| Components of a Web Application | Description |
| Presentation Layer | The Presentation Layer can be somewhat represented by the JSP pages in the views folder of this maven project. This layer is responsible of collecting user input and displaying result set based on the business logic. Examples of user inputs are web forms, clicks, and etc. |
| Web Layer/ Transport Layer | The Web layer is typically handled by Servlet Engines. It receives request from the presentation layer and hands over to the application layer for processing. |
| Application Layer | The Application Layer is where all business logic resides. It includes model objects, components that control the application flow and components that implement business logic. Model objects are the POJO entities, it models with attributes of real world objects virtually. For example, a computer.java has attributes like RAM, Processor, SSD, etc. Other components are the controllers and services classes and also includes the repository interfaces. |
| Data Access Layer | The Data Access Layer is responsible for all database interactions. The logic can somewhat be represented with the repository interfaces where the standard Crud operations and custom JPA SQL are written. |
| Database Server | The Database Server is the where data of entities are stored in tables. The data are may be created, read, update and delete based on the request from the Data Access Layer |

1. System
   1. List down all the Modules to be developed
   2. List down all the pages in each of the module and briefly describe the purpose
   3. Note down which are Create, Edit & View pages

|  |  |  |  |
| --- | --- | --- | --- |
| a. Module | b. Pages | Brief description | c. Operations |
| Common | footer.jsp | Copyright, company name, privacy policy and sitemap | View |
| index.jsp | Default landing page and login. Users can login directly. | View |
| postjob.jsp | Create a new job | Create |
| searchjob.jsp | Display job search results | Search |
| viewjob.jsp | View job details | View |
| forgotpassword.jsp | Enter email to send password reset instructions | View |
| editpassword.jsp | Edit password | Edit |
| jobhistory.jsp | Save and edit job history | Create, Edit |
| education.jsp | Save and edit education history | Create, Edit |
| Public site | home.jsp | Landing page for successful login | View |
| header.jsp | Navigation bar | View |
| register.jsp | Register for an account to the portal | Create |
| searchusers.jsp | Display search results | View |
| viewprofile.jsp | Display public profile of user | Create (connection) |
| editprofile.jsp | Make changes to personal details | Edit |
| Admin Private site | adminheader.jsp | Admininstors’ Navigation bar | View |
| adminhome.jsp | Administrators’ landing page for successful login | View |
| adminlistusers.jsp | List all users and edit. | View |
| adminemail.jsp | Compose email | Create |
| adminviewprofile.jsp | Allow delete and edit user profile | Edit |

* 1. Create the Sitemap of Public & Private Site

1. Technical Environment Requirements

* Eclipse IDE - Version: 2021-12 (4.22.0)
* MySQL Workbench 8.0 - Version 8.0.27
* MS Word
* MS PowerPoint
* Tomcat v9.0 Server
* JRE System Library [JavaSE-11]
* Spring - Version 5.1.5.release
* Spring JPA – Version 2.1.5.release
* Hibernate – Version 5.4.1.final
* Junit Jupiter – Version 5.8.2
* Javax Servlet – Version 4.0.1

1. System Integration Requirements

* JMS (Java message Service)
* Junit Jupiter 5.6.2

1. Portability Requirements

* Windows, MacOS, Linux
* Web App - .zip
* Database - .sql

1. Maintainability Requirements

* MVC design pattern – Separation of concern
* Java naming conventions

1. Performance Requirements

* Avaliable 24hrs
* Load web application in less than 3 seconds

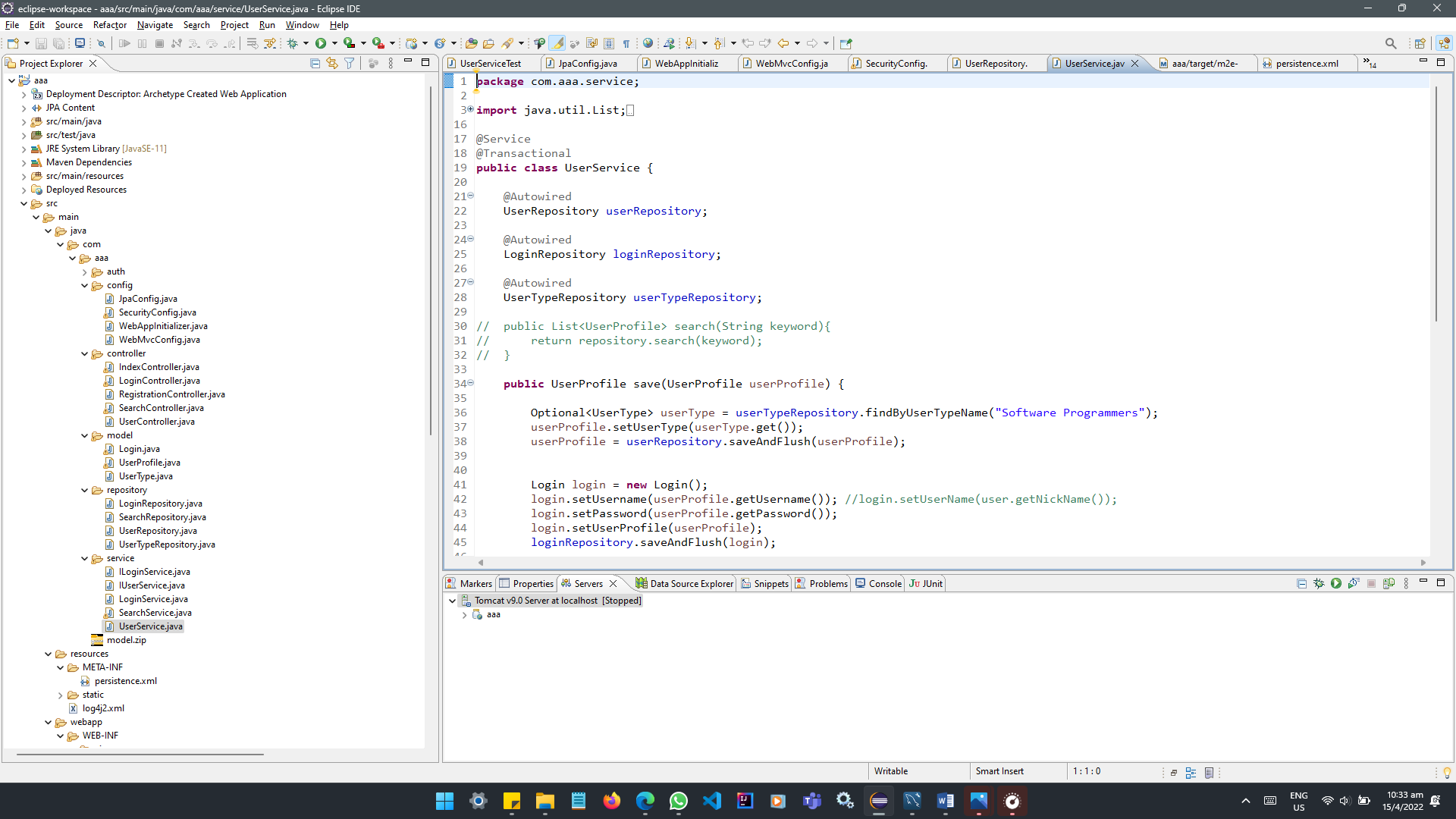
1. Security Requirements
   1. List down the Security Access for different kind of users

|  |  |  |  |
| --- | --- | --- | --- |
| .permitAll() | .hasAnyRole | .hasRole(“ABC Adminstrators”) | .hasRole(“Software Programmers”) |
| index.jsp | home.jsp | adminlistusers.jsp |  |
| register.jsp | postjob.jsp | adminemail.jsp |  |
| forgetpassword.jsp | searchjob.jsp | adminhome.jsp |  |
|  | viewjob.jsp |  |  |
|  | forgotpassword.jsp |  |  |
|  | jobhistory.jsp |  |  |
|  | education.jsp |  |  |

* 1. Briefly explain Login & Logout mechanism
* User logins by entering credentials.
* Server verifies that the user’s credentials are correct, and creates a cookie with the session info which is stored in the database.
* The cookie, with the session ID and other information, is also stored in the browser.
* When the user navigates through various pages on the browser, the cookie is verified against the database to validate if the user credentials are valid.
* When the user logs out, the session is also deleted from the database.

Task 2 – Eclipse IDE Installation

1. Install Eclipse in your laptop
2. Provide screen capture of Eclipse IDE

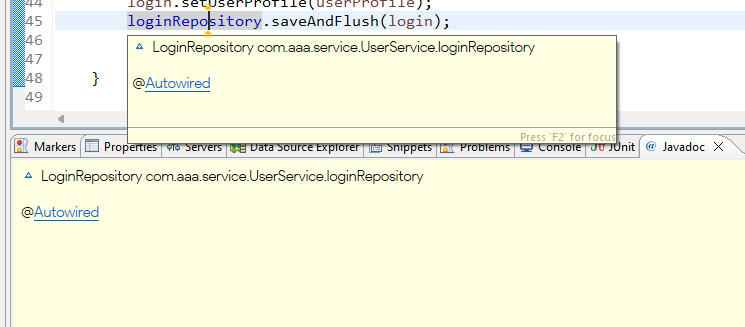
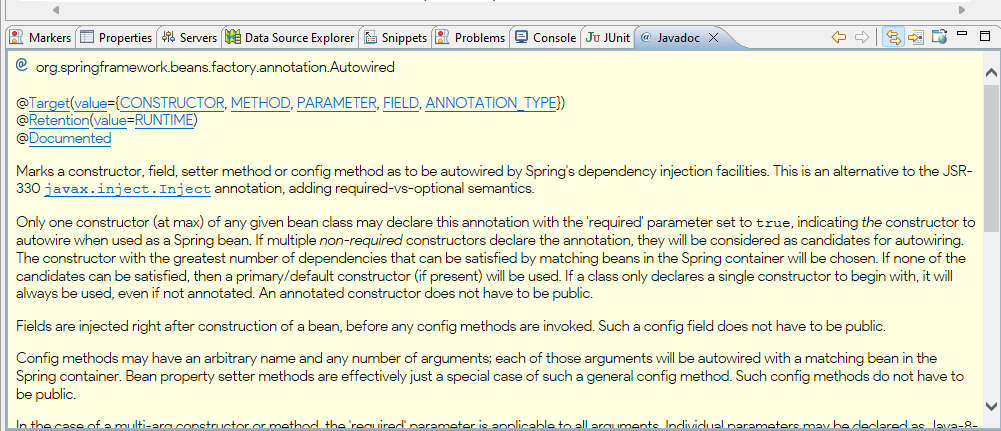


1. Include it as part of Project Presentation

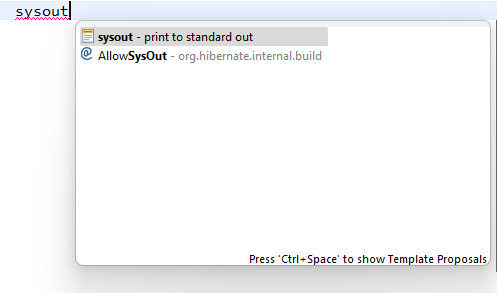
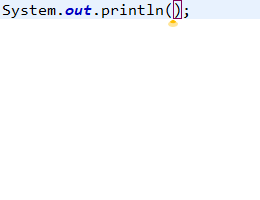
Task 3 – List 3 features in Eclipse IDE

1. Provide description of 3 features in IDE along with the screen capture of their use.

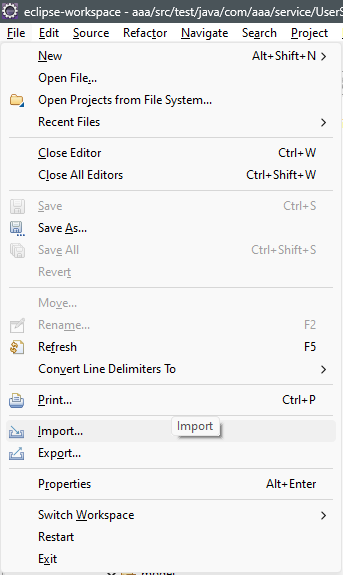
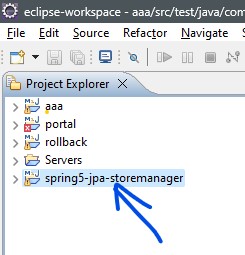
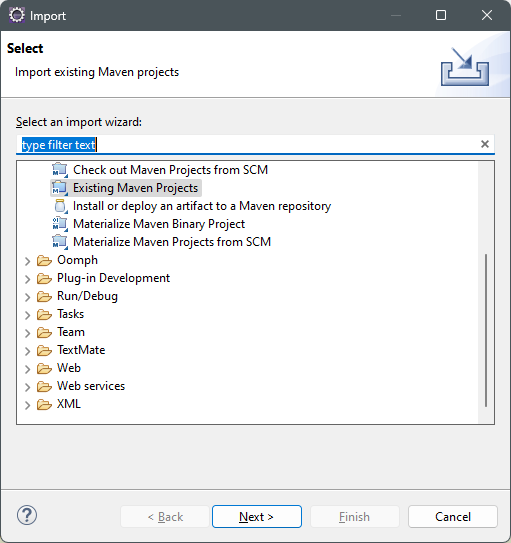
**Javadoc** – Provide easy access to the Java Documentation of an element when mouse-over.

**Template Proposal** – Ctrl + Spacebar, directional button and enter to auto generate snippets of codes.

**Import Projects** – Continue or get started with existing projects. Limitless possibilities.

1. Include it as part of Project Presentation
2. Task 4 – Coding & Documentation Standards
3. List 5 Coding Standards & Conventions to be used and explain shortly.

|  |  |
| --- | --- |
| Coding Standards & Conventions | Brief description |
| Pascal case and noun for Classes | Classes (created) models after real world objects and named after the object itself (which are nouns). These non-primitive data types should avoid camel case (reserved naming conventions for classes of primitive data types). |
| Camel case for variable instances | Lengthy variable names may be assigned for the purpose of making it “meaningful”. Camel case can enhance readability and also less error prone. Example: thisVariableRepresentsSomething is easier to read than thisvariablerepresentssomething. |
| Adjective for Interfaces | An interface (created) can be easily identified with naming ending with “able”. Examples are Readable, Serializable, Serviceable, and etc. |
| Screaming Snake case for Constants | Screaming snake case stands out and helps identifying CONSTANT\_VARIABLE almost instantly. |
| Single letter variable names | Especially useful in block scoped variables, these single letter temporary variable names such as i, j, k, m, n, c, d & e characters are conveniently placed near the middle &/or index finger and used widely by programmers. |

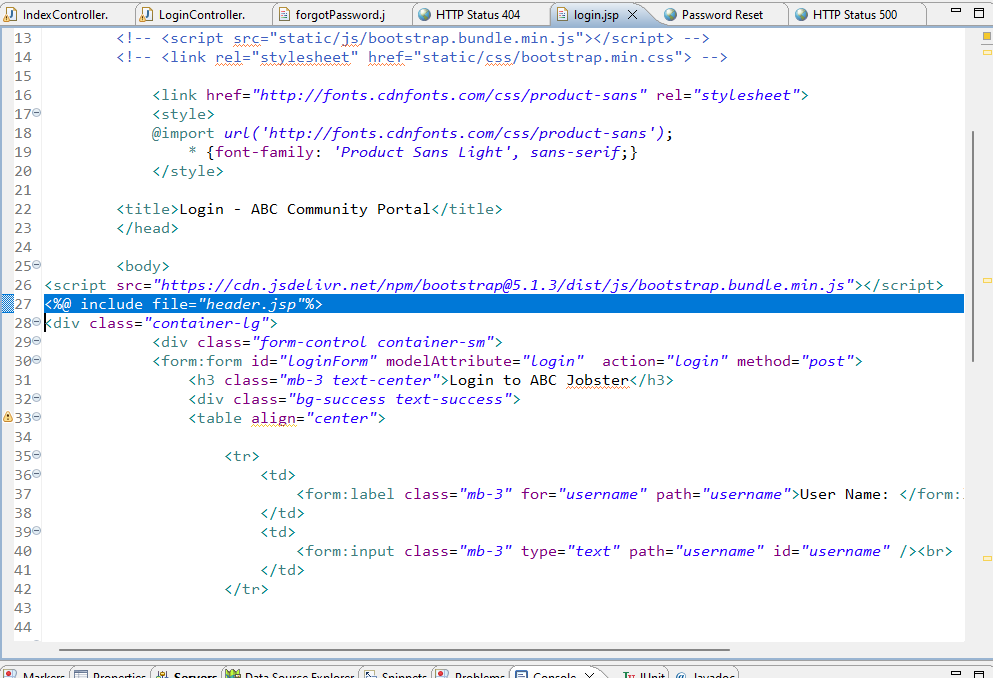
1. Describe briefly documentation tool, you will be using and its configuration.

|  |  |
| --- | --- |
| Javadoc Screenshot | Configuration & brief description |
|  | Private:  Javadoc for all classes and members to be included. |
|  | Select All:  Included all referenced archives and projects to which links should be presented. |

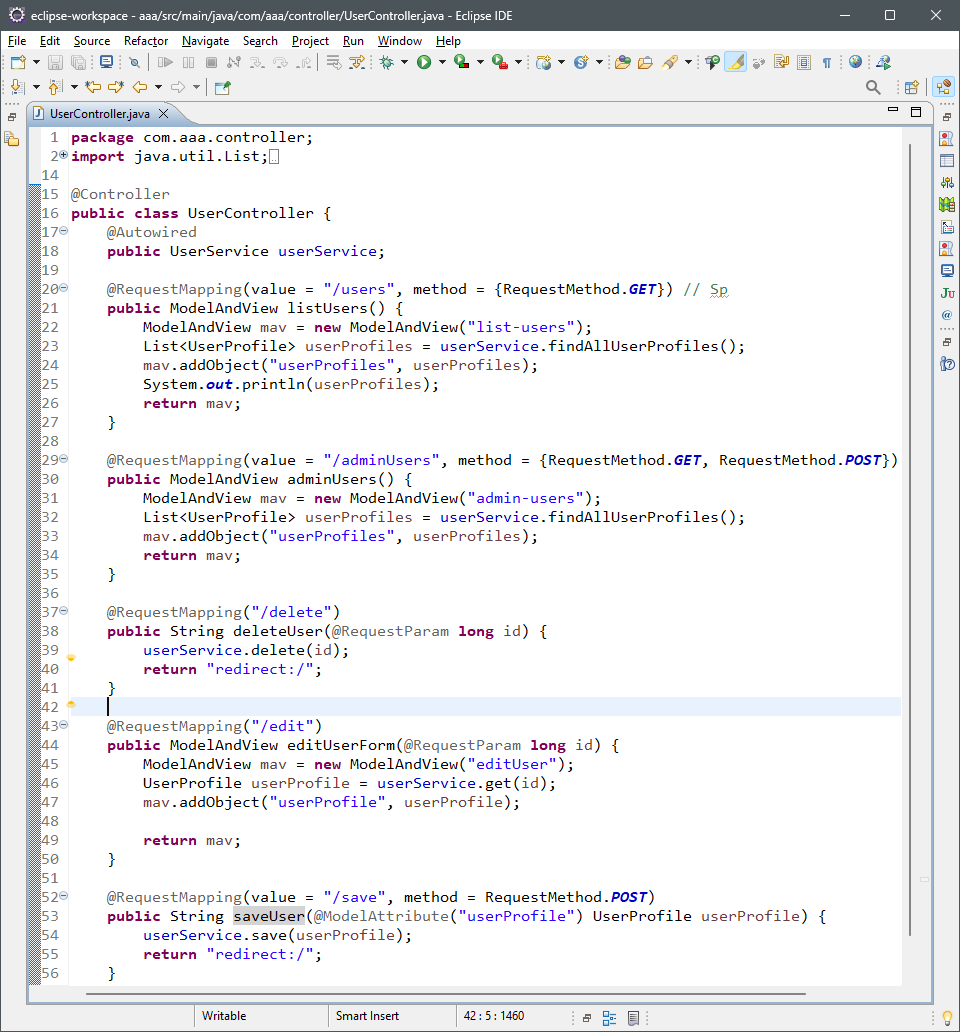
1. Task 5 – Views in JSP/HTML

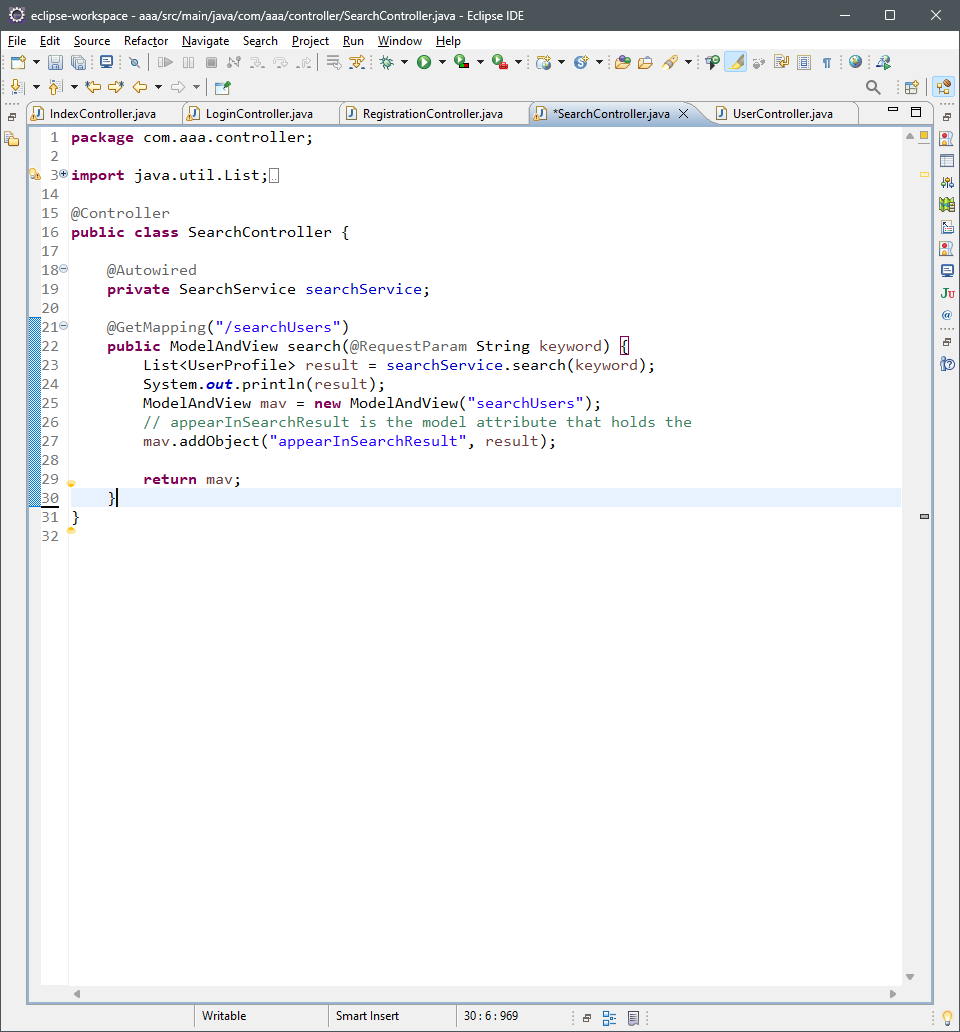
|  |  |
| --- | --- |
| 1. Develop the following pages in JSP | |
| * 1. Landing Page | * 1. Create/ Edit Page |
| * 1. Listing Page | * 1. Login/ Logout Page |
| * 1. Forget Password Screen | * 1. Administration Page |

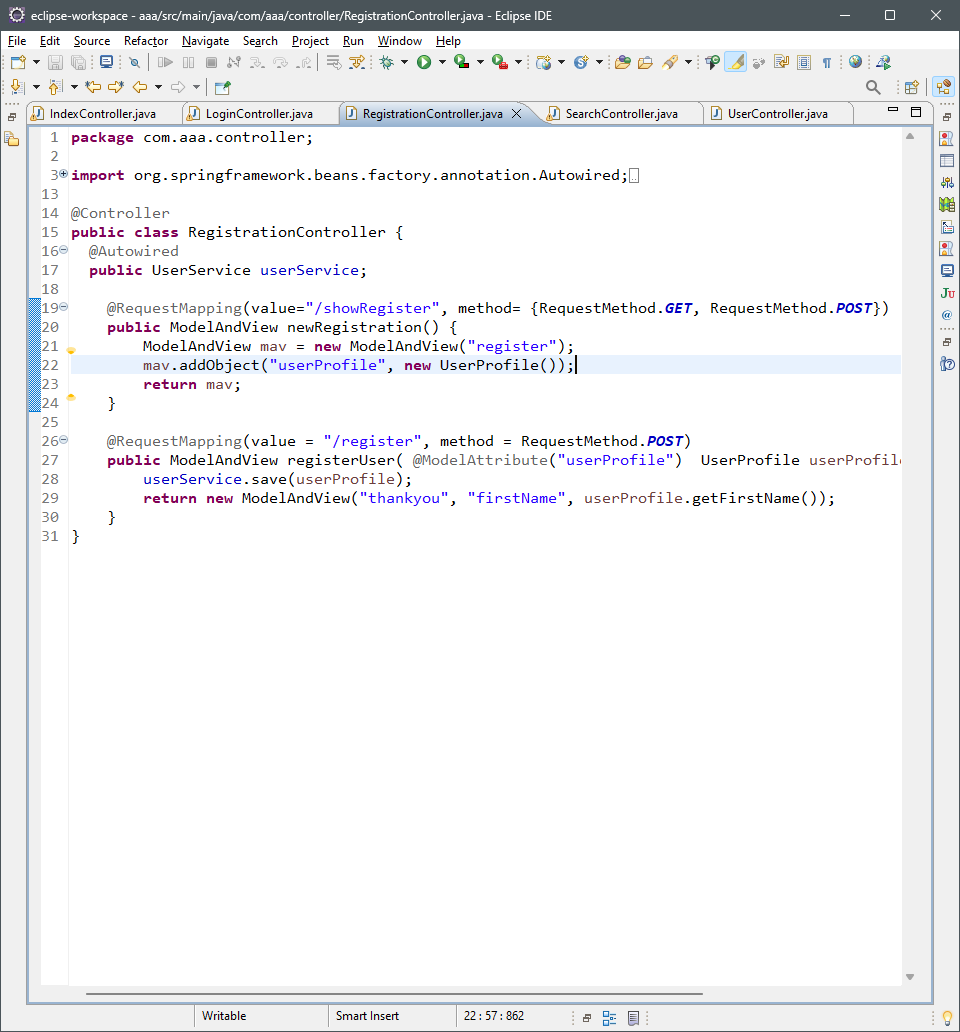
1. Template those by separating Headers/Footers and including them using Server Side includes.

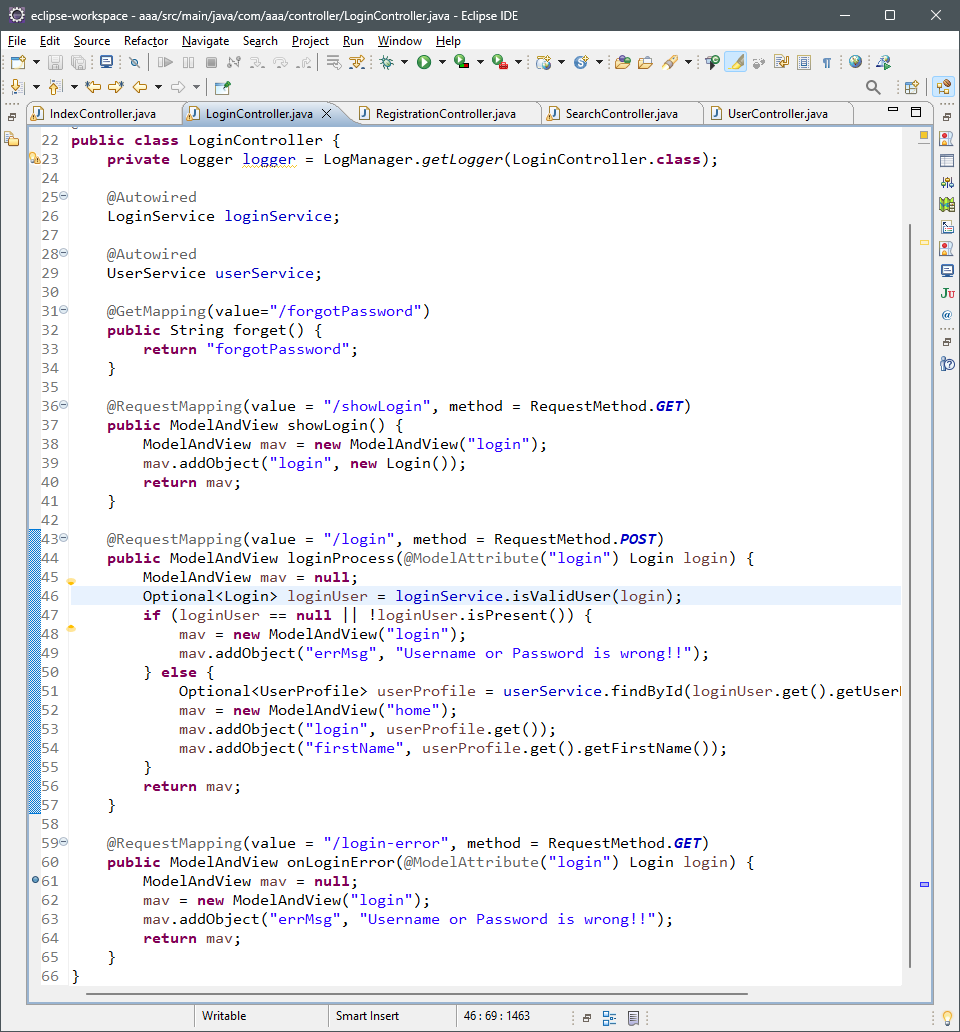


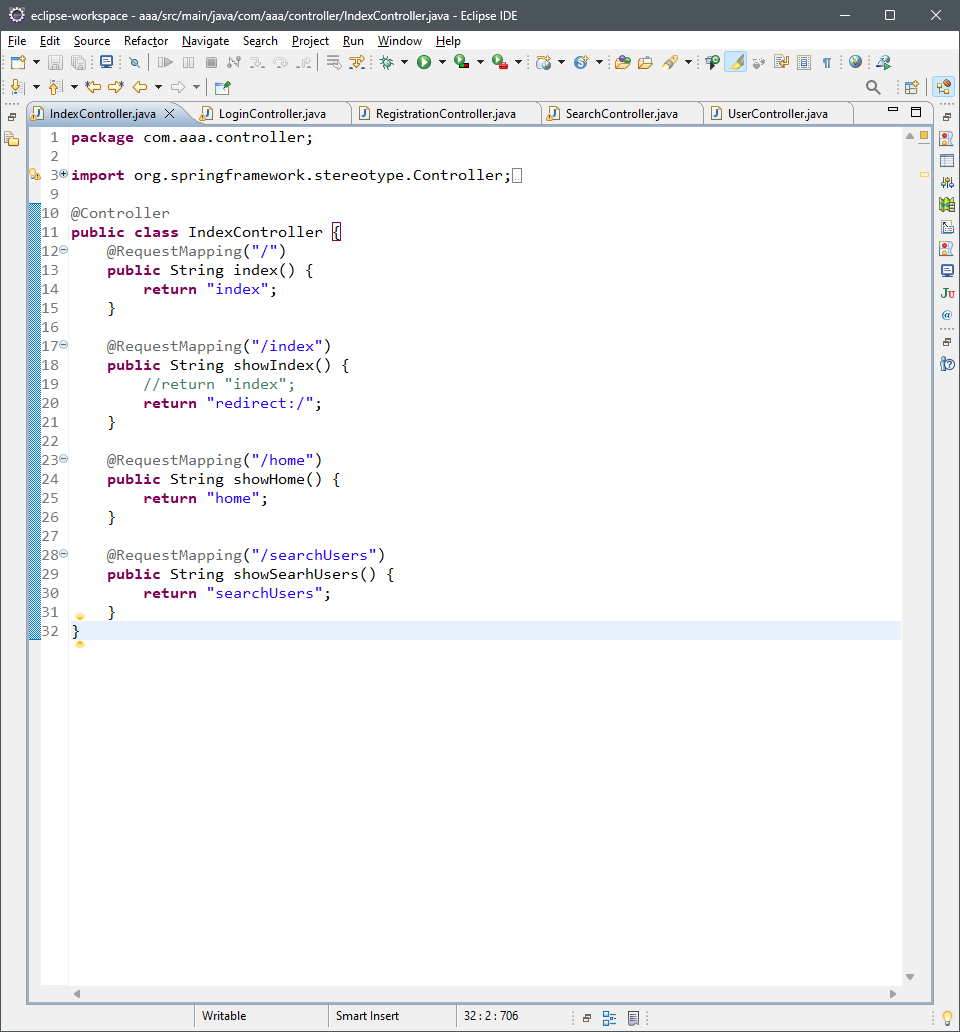
1. Provide the screen capture in Project Presentation
2. Task 6 – Repository Interfaces, Controller, Model & Service Classes
3. Create the required classes in Java





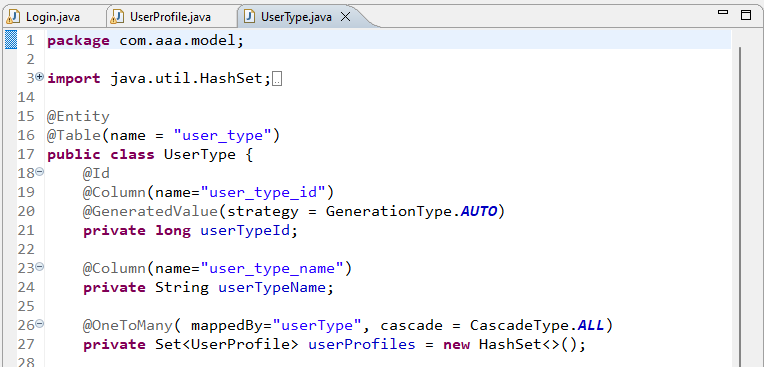


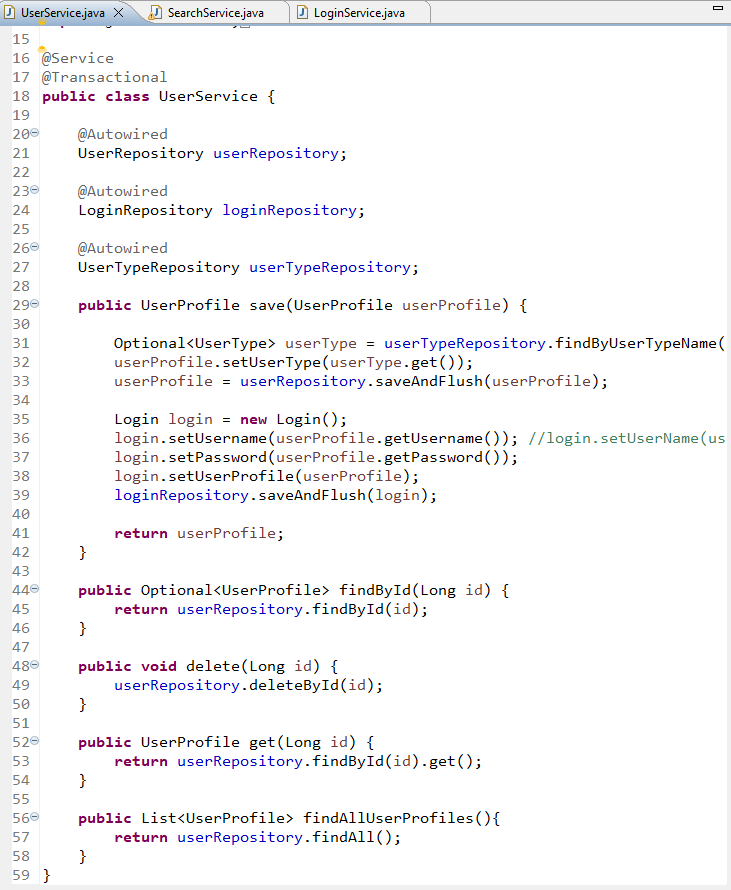
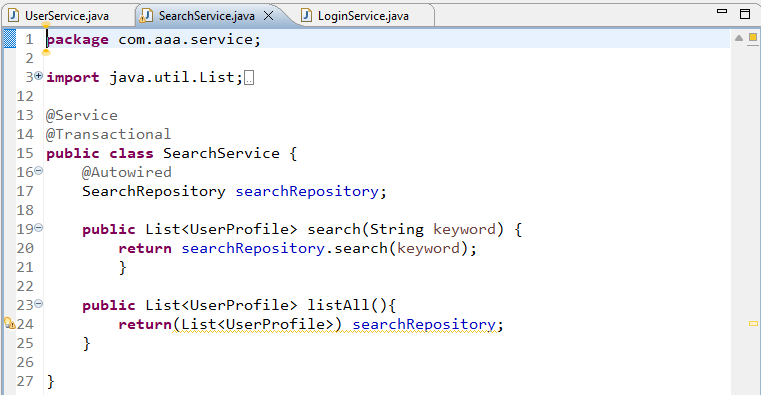


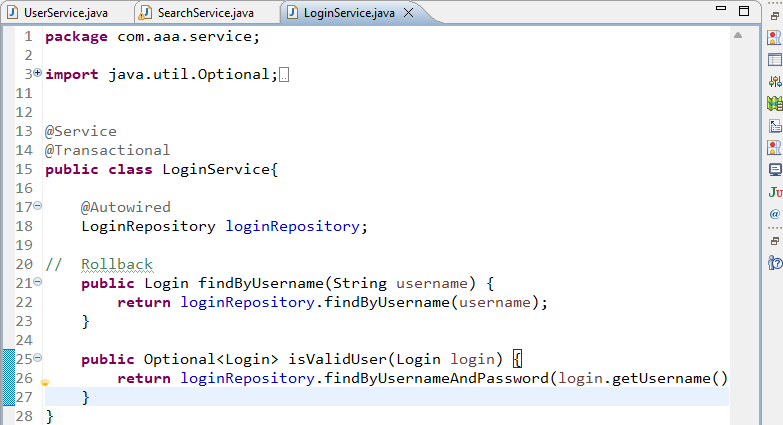




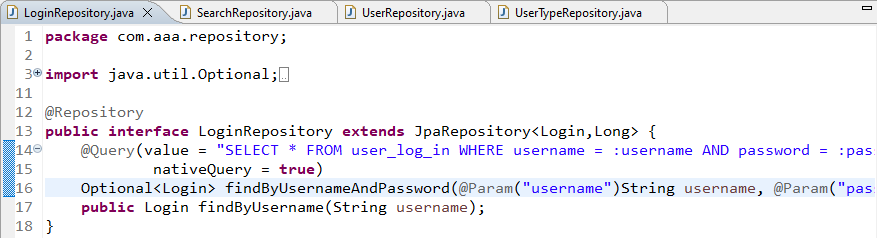
****

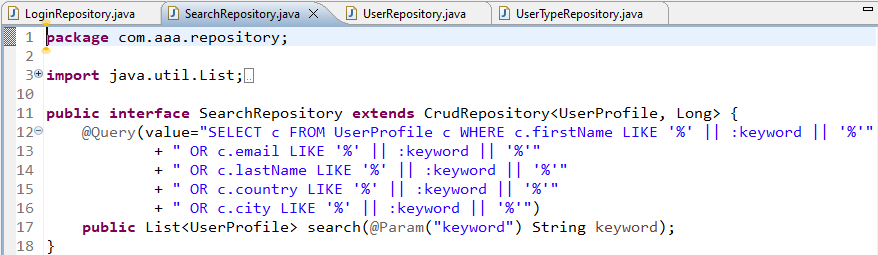
****

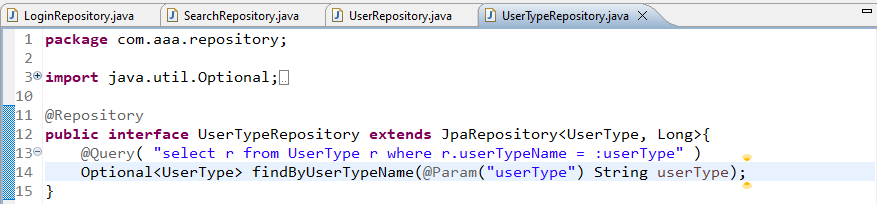
**  
**

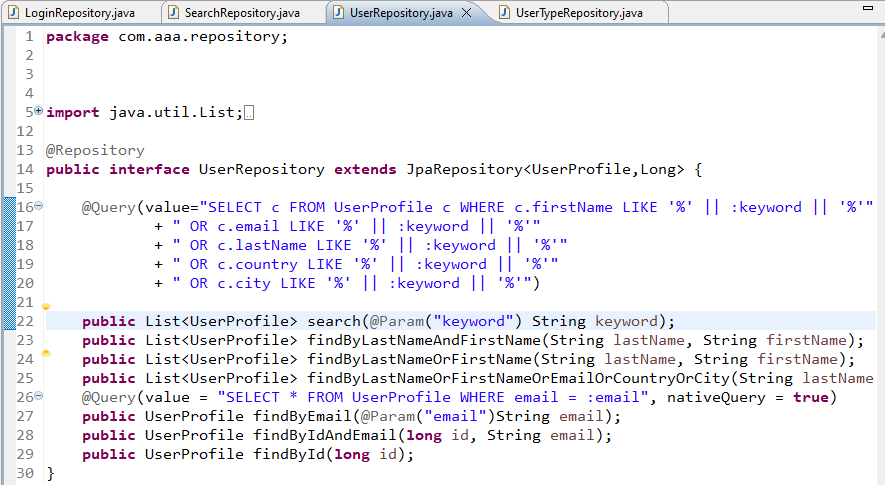
****

1. Include Data Access Layer classes









1. Include samples as part of Project Report
2. Task 7 – Presentation of application with functionality
3. Develop the Models, Views, Controller
4. Develop Home Page & Intergrade with Login System

|  |  |  |
| --- | --- | --- |
| Home Page | Login | Login Successful |
|  |  |  |

1. Develop Registration Form & Registration Thank you

|  |  |  |
| --- | --- | --- |
| Home Page | Register Form | Thank you |
|  |  |  |

1. Develop Forget Password mechanism

|  |  |
| --- | --- |
| Login | Forget Password |
|  |  |

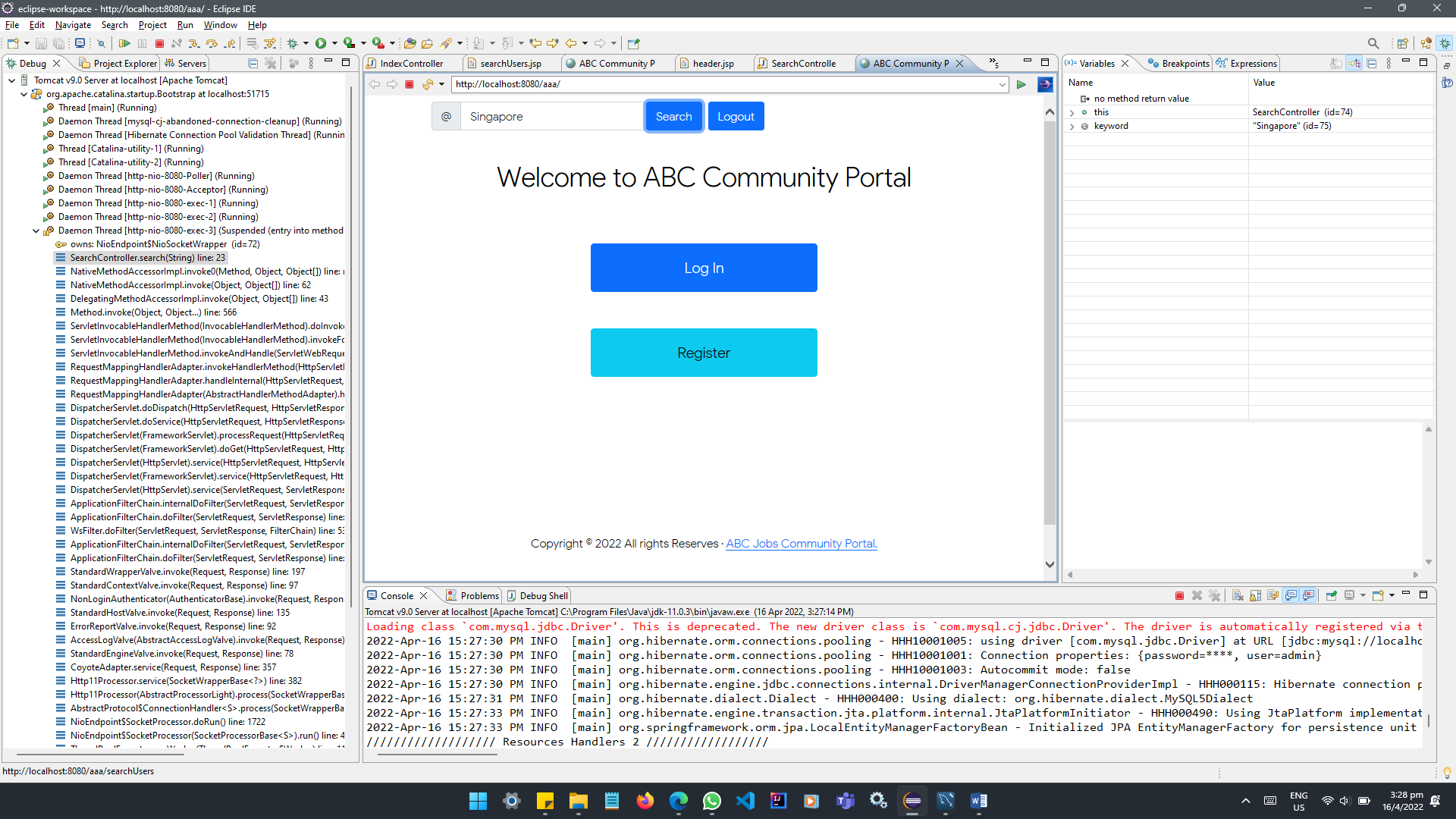
1. Develop Landing Page after login
2. Develop Search Box for Users
3. Develop View Search Results

|  |  |
| --- | --- |
| Search user | Shows Search Result |
|  |  |

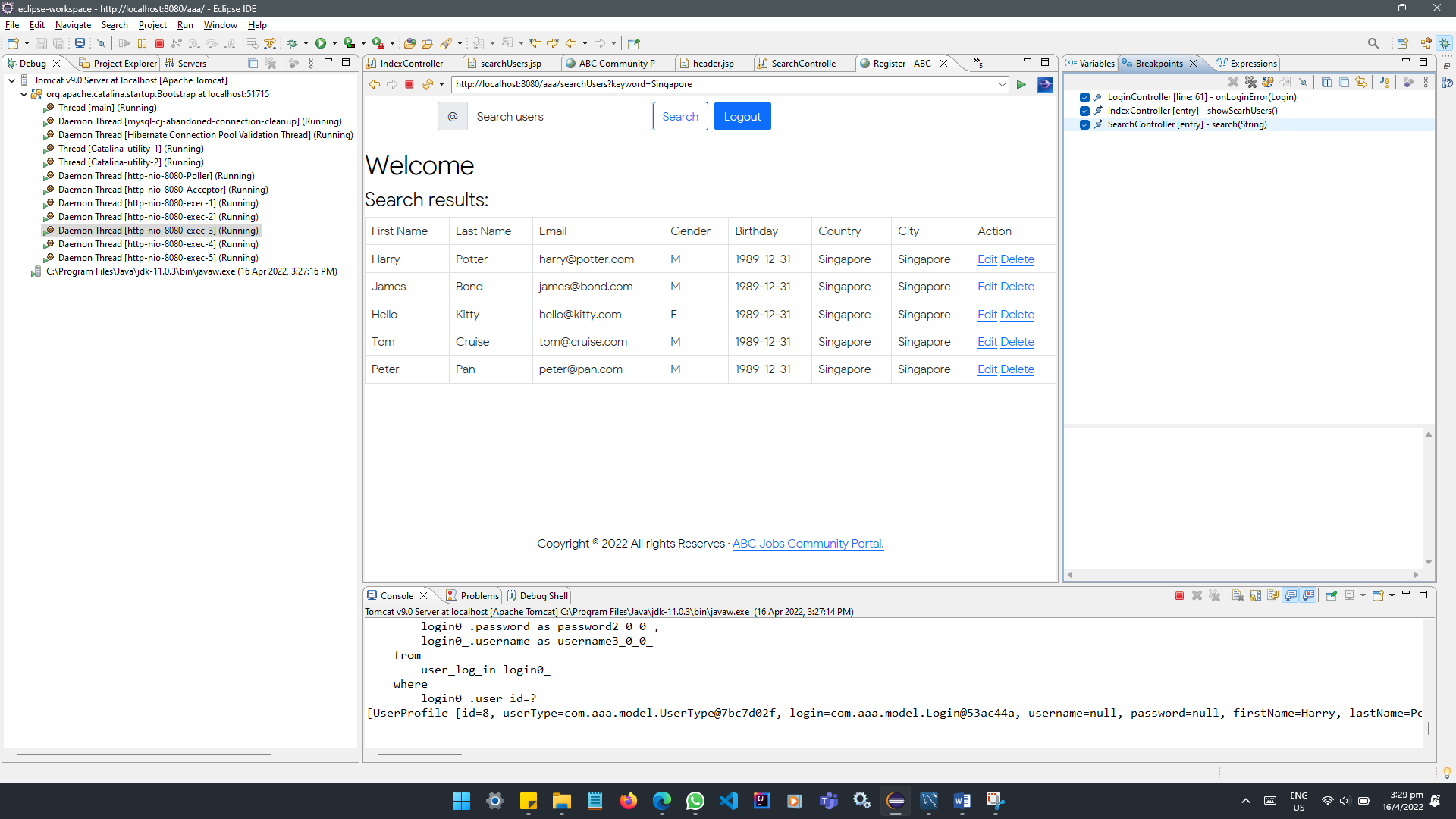
1. Develop View profile Page
2. Develop Edit Profile Functionality

|  |  |
| --- | --- |
| Edit Profile | Update Profile |
|  |  |

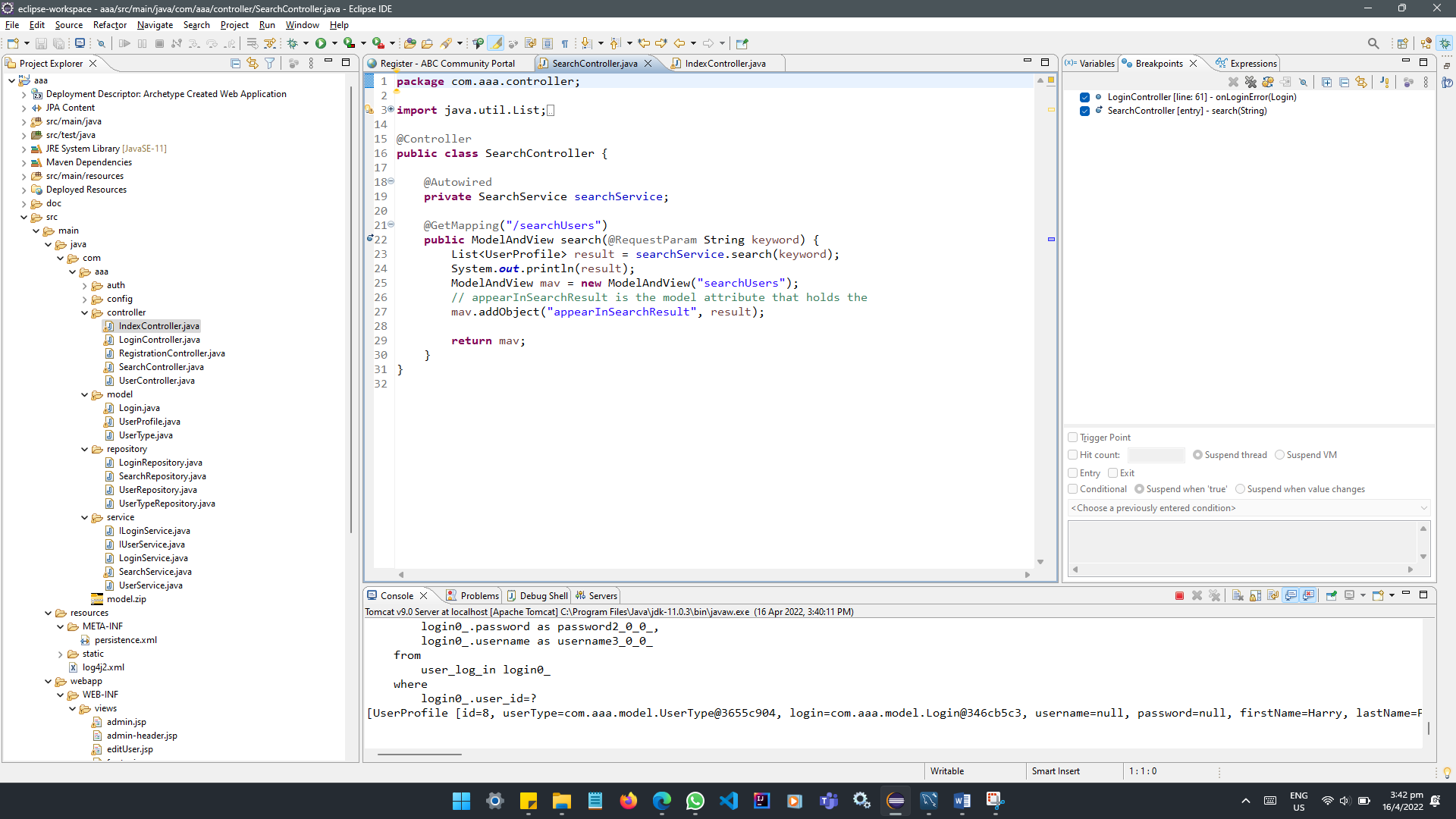
1. Task 8 – Debugging in Eclipse IDE
2. Debug the application in Eclipse
3. Provide the screen capture of your application code in Eclipse Debug Mode. Include it as part of Project



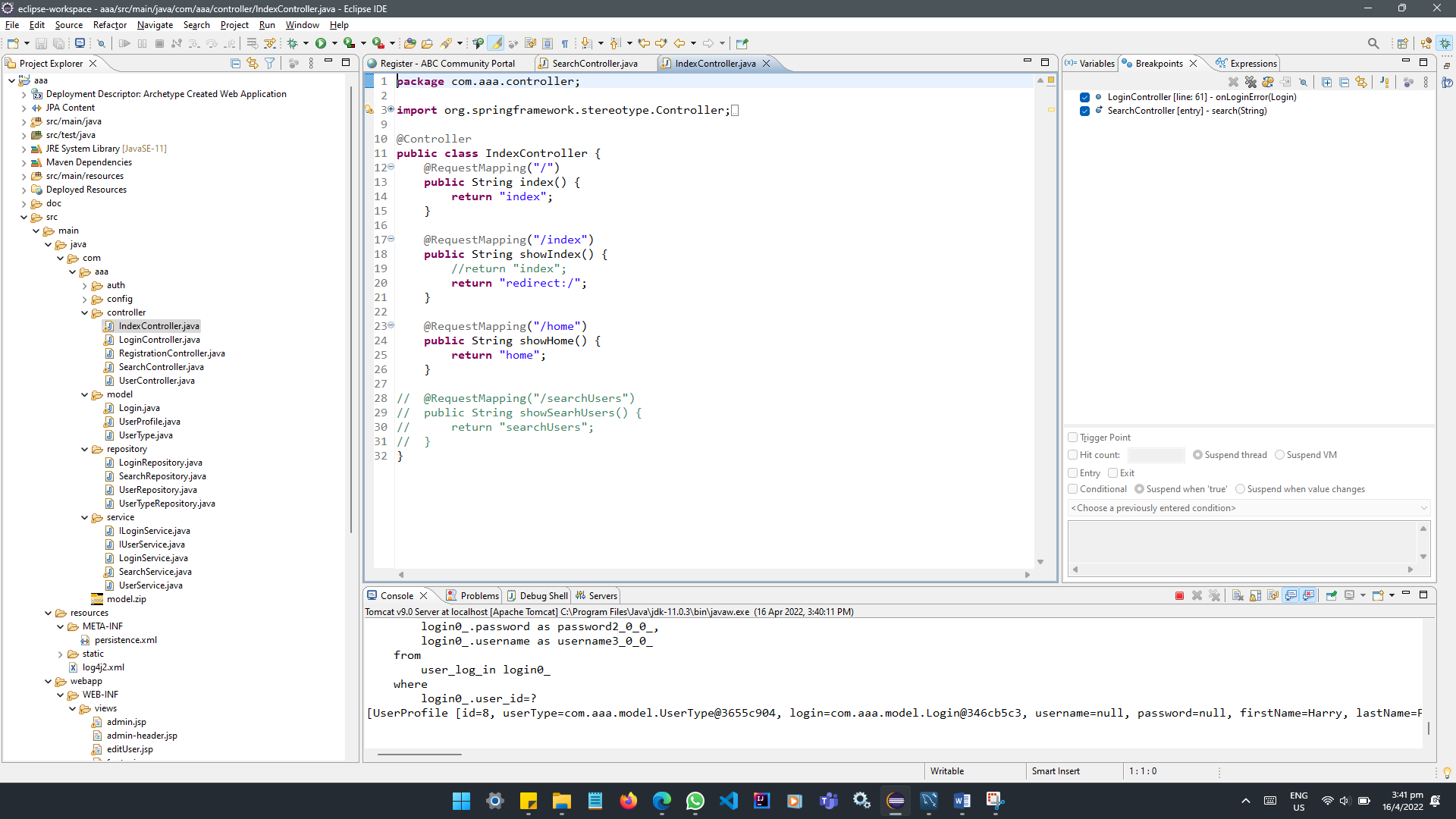
Found duplicate calling in IndexController.java & SearchController.java



The Mapping of searchUsers.jsp is correct in SearchController.java



Commented out Line 28 to 31 of IndexController.java and tested working as expected.



1. Task 9 – Demonstrate 3 troubleshooting
2. Provide screen capture of 3 Errors you encountered while developing the application.
3. How did you handle those 3 errors(Write in short bullet points)

|  |  |
| --- | --- |
| Error screen capture | Error handling |
|  | - Look at HTTP Status  - Check out the Root Cause  - Look at the last sentence  - Unable to create requested service [org.hibernate.engine.jdbc.env.spi.JdbcEnvironment]  - Likely caused by JDBC or repository interfaces  - Check persistence.xml  - Fixed jdbc:mysql://localhost:5306 to jdbc:mysql://localhost:3306 |
|  | - Look at HTTP Status  - Check out the Root Cause  - Look at the last sentence  - Unknown mapped By in: com.aaa.model.UserProfile.login, referenced property unknown: com.aaa.model.Login.user  - Error could be caused by Login.java or UserProfile.jav models or incorrectly joined.  - Check for any red cross  - Error indicator appearing at UserProfile.java  - Changed (MappedBy=”user”) to (MappedBy=”userProfile”) |
|  | - Look at HTTP Status  - HTTP Status 404 – Not Found  - Check that mapping for adminHome is available  - GetMapping(value=”/adminHome”) is missing  - Added GetMapping(value=””) to AdminController.java  - Check the return value html/jsp page is created  - Message JSP file [/WEB-INF/views/adminHome.jsp] not found  - Create the missing file and test again.  - Page is running. |

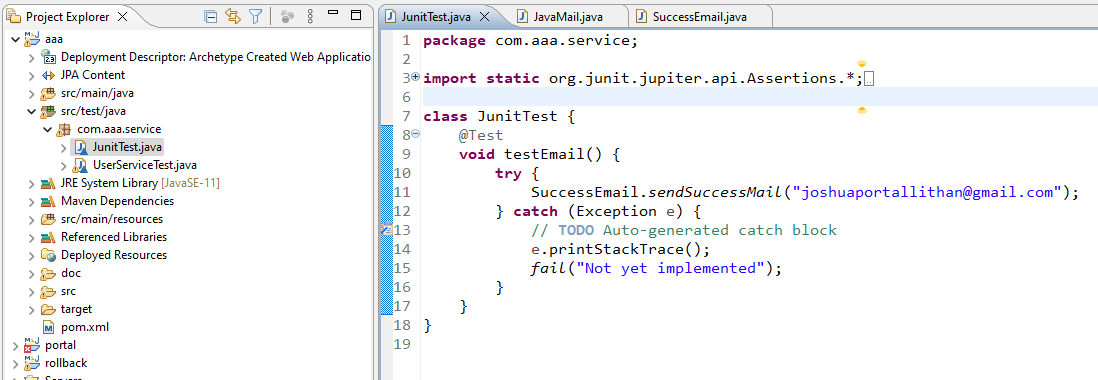
1. Include it as part of Project Presentation
2. Task 10 – Propose new feature(s)
3. Propose & Document a new feature to the application

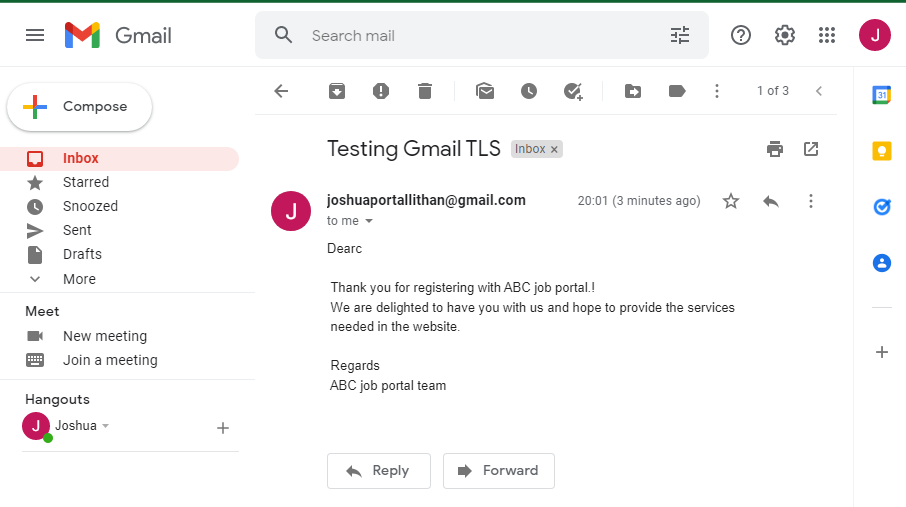
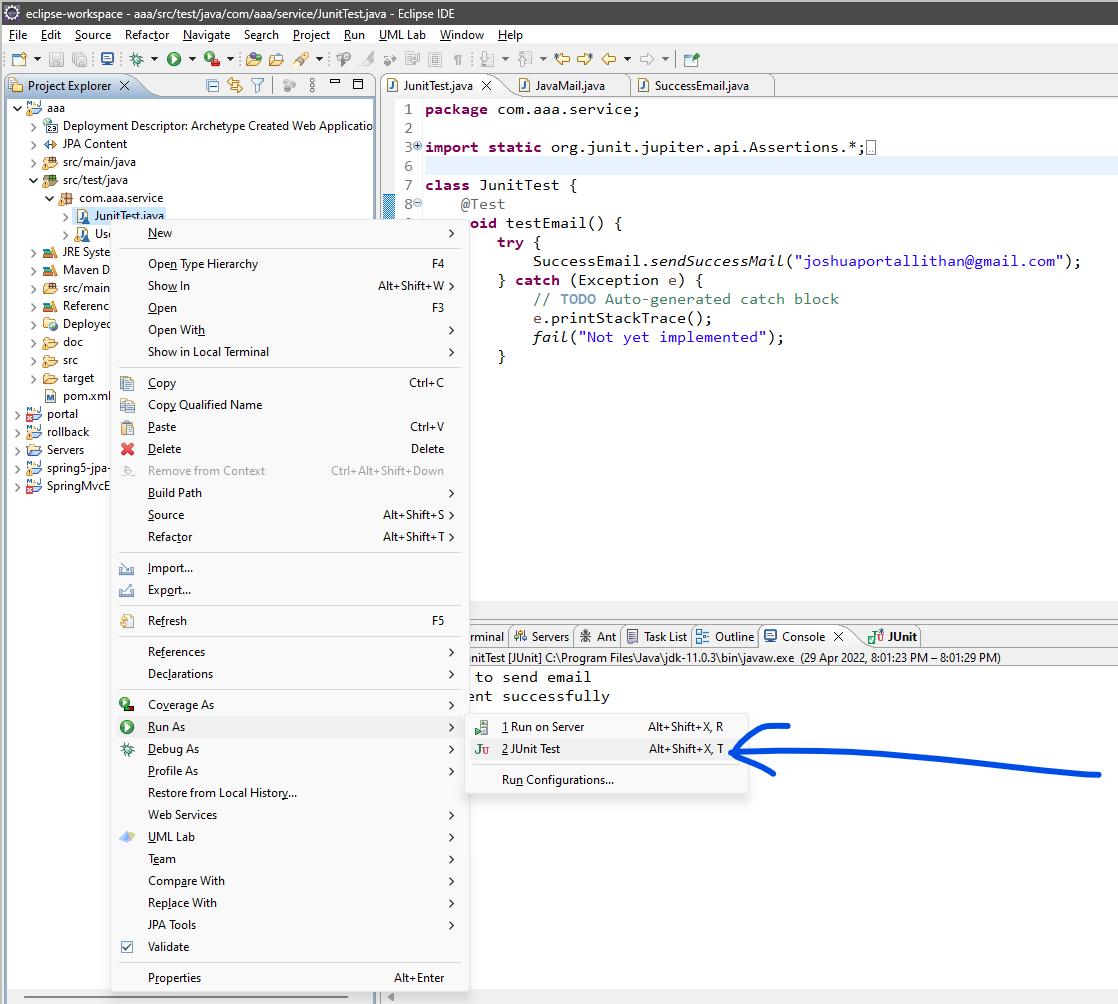
|  |  |
| --- | --- |
| Proposed new feature | How to develop or implement feature |
| Allow users to give a “like” to user. | Add a column to table user\_profile  Likes will be int and default value set to ‘0’.  Add getter & setter in userProfile.java  Add a button to allow other users to click |

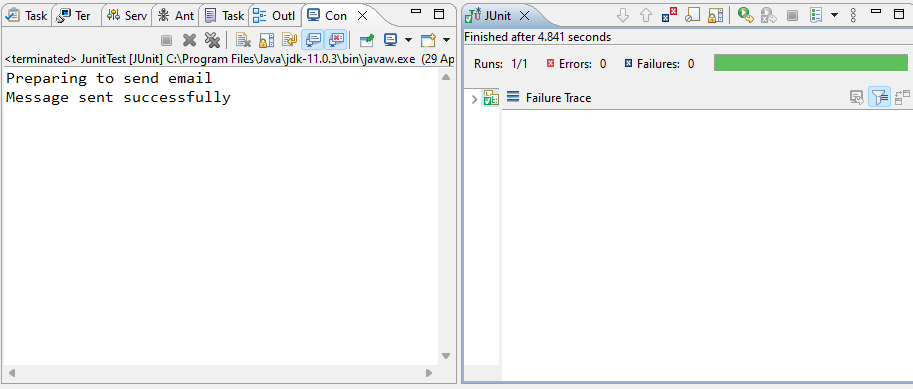
1. Develop the new feature and provide a report

|  |  |
| --- | --- |
| New feature “views” |  |
| New “Likes” counter: | Likes counter included in search results: |
| Added new column “likes” to user\_profile table: | |
| New feature in searchUsers.jsp: | |

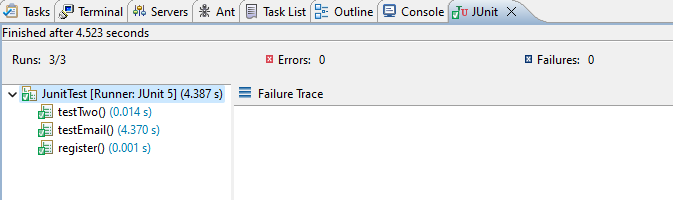
1. Include it as part of Project Report
2. Task 11 – Junit test cases in Eclipse IDE
3. Develop and execute unit testing for individual modules
4. Create Test Cases for the Individual Modules











1. List the bugs and document in Project Report