

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	
Chathushi Jayarathna	Ms. Arvinder Kaur	
Date issued	Completion date	Submitted on
12/09/2022	25/09/2022	02/11/2022

Project title	Design, Develop, Implement & Document
----------------------	--

Learner declaration
I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.
Student signature: Chathushi Date: 25/09/2022

Table of Contents

1. Project Background	5
2. Project Objectives	6
3. Block diagram of the system.....	10
4. Case Diagram.....	11
5. Architecture of the system	16
5.1 Explanation of Architecture of the application	16
6. Developed Modules.....	16
6.1 Private Module - Administrator - Manage user Data and Bulk Emailing.	16
6.2 Public Module - Software Programmers	16
7. All the pages in each of the modules and Description	17
8. Create, Edit & View pages.....	18
9. Sitemap of Public & Private Site.....	19
10. Technical Environment Requirements.....	19
10.1 Hardware Requirements.....	19
10.2 Software Requirements	20
10.3 System Integration Requirements	20
10.4 Portability Requirements.....	20
10.5 Maintainability Requirements.....	20
10.6 Performance Requirements.....	20
10.7 Security Requirements	20
12. Login & Logout mechanism	21
12.1 Login	21
12.2 Log out	21
13. Eclipse Installation.....	21
14. Features in IDE	26
14.1 Smart Code Completion	26
14.2 Quick Fixes	26
14.3 Getter and Setter	27
15. Coding & Documentation Standards.....	28
15.1 Comments.....	28
15.2 Naming Convention.....	28
15.3 File and Folder Organization.....	29
16. Documentation tool and its configuration.....	30

17.	HTML Pages Source Code.....	34
17.1	Registration Page (register.jsp)	34
17.2	Login Page (login.js).....	36
17.9	Check Mail (checkMail.jsp).....	45
	45
18.	Java Classes	53
12.	Implemented pages	72
✓	Home Page	72
13.	Debug the application in Eclipse.....	83
14.	Errors you encountered while developing the application.....	87
15.	New feature to the application	88
16.	Unit Testing code.....	89

Figure 1:Screenshot of Eclipse	7
Figure 2;Screenshot of MySQL Workbench	7
Figure 3:Screenshot of XAAMP Server	8
Figure 4:Screenshot of Flow Drawio.....	8
Figure 5:Screenshot of Microsoft Word	9
Figure 6;Screenshot of Microsoft PowerPoint.....	9
Figure 7:Block Diagram of System	10
Figure 8;Login Process	11
Figure 9:Registration Flow	12
Figure 10: Architecture of the system.....	16
Figure 11:Sitemap of Public and Private Site	19
Figure 12:Screenshot of address bar in browser	22
Figure 13:Screenshot of Eclipse installation step 3	22
Figure 14:Screenshot of Eclipse installation step 4	23
Figure 15:Screenshot of eclipse installation step 5	23
Figure 16:Screenshot of eclipse installation step 6.....	23
Figure 17:Screenshot of eclipse installation step 7	24
Figure 18:Screenshot of eclipse installation step 8.....	24
Figure 19:Screenshot of eclipse installation step 9.....	24
Figure 20:Screenshot of eclipse installation step 10.....	25
Figure 21:Screenshot of eclipse installation step 11	25
Figure 22:Screenshot of of smart code completion in eclipse	26
Figure 23:Screenshot of quick fixes in eclipse	27
Figure 24:Screenshot of coding & documentation Standards in eclipse (getters and setters)	27
Figure 25:Screenshot of coding & documentation Standards in eclipse (comments)	28
Figure 26:Screenshot of coding & documentation Standards in eclipse (Naming Conventions)	29
Figure 27:Screenshot of coding & documentation Standards in eclipse (File and Folder Organization)	29
Figure 28:Javadoc configuration step 2	30
Figure 29:Screenshot of Javadoc configuration step 7	31
Figure 30:Screenshot of Javadoc configuration step 3,4,5,6	31
Figure 31:Javadoc configuration step 8	32
Figure 32:Javadoc configuration step 9	32
Figure 33:Screenshot of Javadoc	33
Figure 34:Screenshot of source code of registration.....	Error! Bookmark not defined.
Figure 35:Screenshot of source code of login.....	Error! Bookmark not defined.
Figure 36:Screenshot of source code of thank you page	Error! Bookmark not defined.
Figure 37:Screenshot of source code of registration forgot password....	Error! Bookmark not defined.
Figure 38:Screenshot of source code of forgot password confirmation .	Error! Bookmark not defined.
Figure 39:Screenshot of source code of user profile	Error! Bookmark not defined.
Figure 40:Screenshot of source code of edit profile	Error! Bookmark not defined.
Figure 41:Screenshot of source code of search user.....	Error! Bookmark not defined.
Figure 42:Screenshot of source code of update profile	Error! Bookmark not defined.
Figure 43:Screenshot of source code of MvcController.java ..	Error! Bookmark not defined.
Figure 44:Screenshot of source code of userDao.java.....	Error! Bookmark not defined.
Figure 45:Screenshot of source code of User.java.....	Error! Bookmark not defined.
Figure 46:Screenshot of Registration Page.....	Error! Bookmark not defined.

Figure 47:Screenshot of Thank you page	Error! Bookmark not defined.
Figure 48:Screenshot of login.....	Error! Bookmark not defined.
Figure 49:Screenshot of forgot password validations.....	Error! Bookmark not defined.
Figure 50:Screenshot of forgot password page.....	Error! Bookmark not defined.
Figure 51:Screenshot of forgot password confirmation.....	Error! Bookmark not defined.
Figure 52:Screenshot of user profile page	Error! Bookmark not defined.
Figure 53:Screenshot of edit profile page	Error! Bookmark not defined.
Figure 54:Screenshot of search user page.....	Error! Bookmark not defined.
Figure 55:Screenshot of debugging in eclipse step 1.....	83
Figure 56:Screenshot of debugging in eclipse step 2.....	83
Figure 57:Screenshot of debugging in eclipse step 3.....	84
Figure 58:Screenshot of debugging in eclipse step 4.....	84
Figure 59:Screenshot of debugging in eclipse step 5.....	85
Figure 60:Screenshot of debugging in eclipse step 6.....	85
Figure 61:Screenshot of debugging in eclipse step 7.....	85
Figure 62:Screenshot of debugging in eclipse step 8.....	86
Figure 63:Screenshot of debugging in eclipse step 9.....	86

1. Project Background

ABC Jobs Pte Ltd is a career guidance organization that gives tasks to individuals. That organization needs to foster a community portal to spread its name to additional clients.

ABC Jobs Pte Ltd suggests occupations and recruits' programmers. With this community portal, clients can enlist for themselves and quest the most prominent positions in their occupations. Likewise, clients can make networks with other clients by interfacing with them in their local area. Likewise, the gateway is principally utilized for proficient collaborating and vocational development

The primary motivation behind this task is to design and develop the community portal as they wish. Locally community portal clients will actually want to enroll and look for another client. After they can log in, reset their password, and update their profile. User details are stored in a database. This ABC Community portal entryway is very valuable to the client side since clients do not have to go to ABC Jobs organization to look through positions, they can look online with the local area gateway. Likewise, clients should need to safeguard a client profile to acquaint themselves. This people group entryway has private and public modules. Confidential modules approach for the administrators and public modules approach programming software engineers.

For this local community portal, we have carried out a java-based web application with a spring MVC structure for the client necessities, business needs, and cycles. Likewise, we want to design, make, plan and execute the tests for this undertaking. As a component of this task, we really want to recognize this problem and oversee it across its lifecycle

Here, the software developer connects the database to get the data of those who register. Then the details of everyone who registers will be mentioned in the database. After that you need to generate data from database and login to the job portal. The user profile will be displayed

according to the user login. The user should be able to login to the user profile and the admin to the admin profile through the same login form. It should also be possible to update the user profile. In case of security reasons, ABC JOBS reserve the right to prohibit/restrict any users accessibility and can remove any user's data from the portal. To provide a safe net browsing for the users of ABC JOBS it provides 24/7 hours administrator support. There, admin can view all users details and can edit and delete them. These are implemented as grants and in addition, search user page, forgot password page, forgot password confirmation page, thank you page also have been implemented.

The main functions are,

- Once the programmers register can view confirmation after displacing thank you page
- Providing the login page
- Provide password retrieval functionality.
- After logging they can update their profile.

2. Project Objectives

We need to design, develop, implement, test and document for ABC Jobs community portal using spring frameworks

- Gathering and identifying requirements and specifications for the system
- Design wireframes and prototypes.
- Conduct usability testing.
- Design the database.
- Develop and implement the application using Eclipse IDE.
- Maintenance and document the application developed.

2.1 Tools and Platforms used

- ✓ **Eclipse IDE for Enterprise Java and Web Developers - 2022- 03 – To implement application**

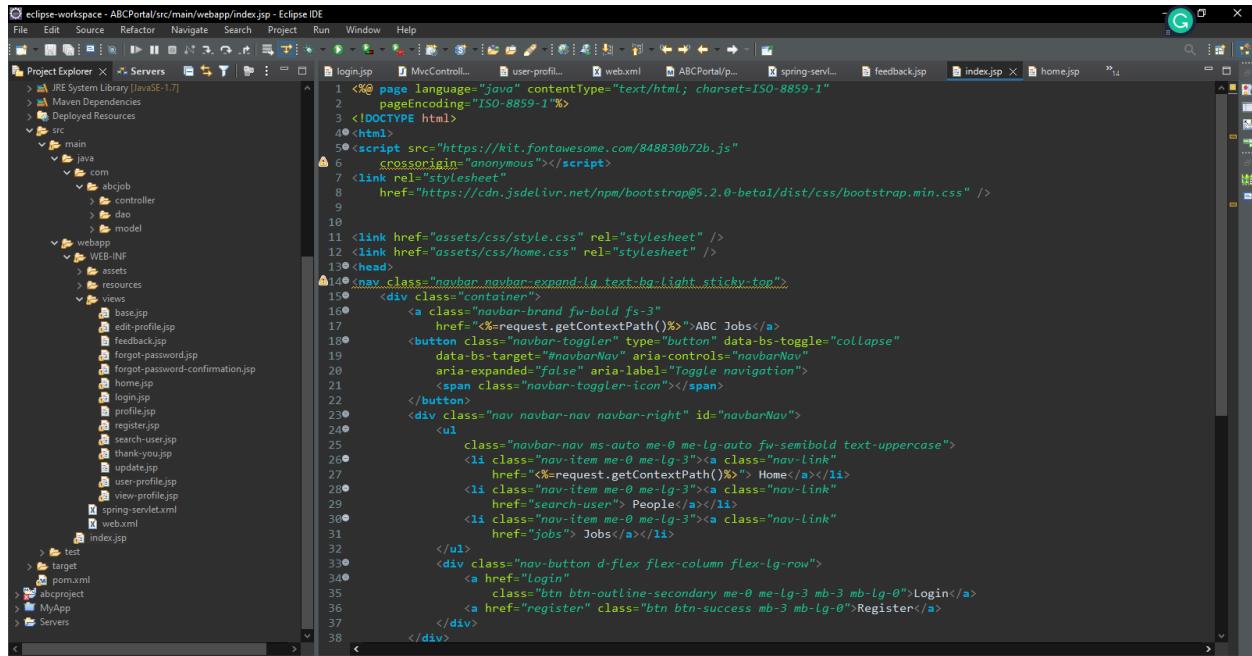


Figure 1: Screenshot of Eclipse

- ✓ **MySQL Workbench 8.0 CE – To create ate database**

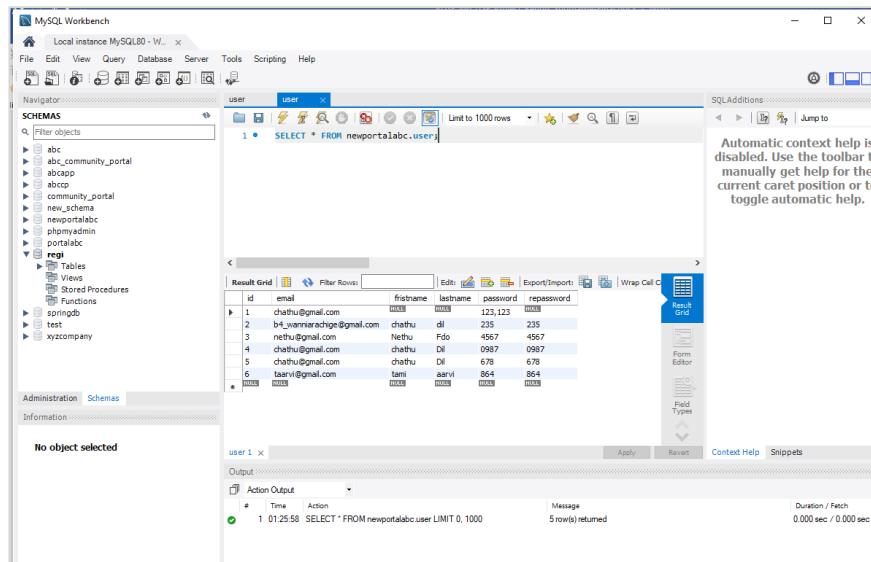


Figure 2; Screenshot of MySQL Workbench

✓ XAAMP Control Panel v3.3.0

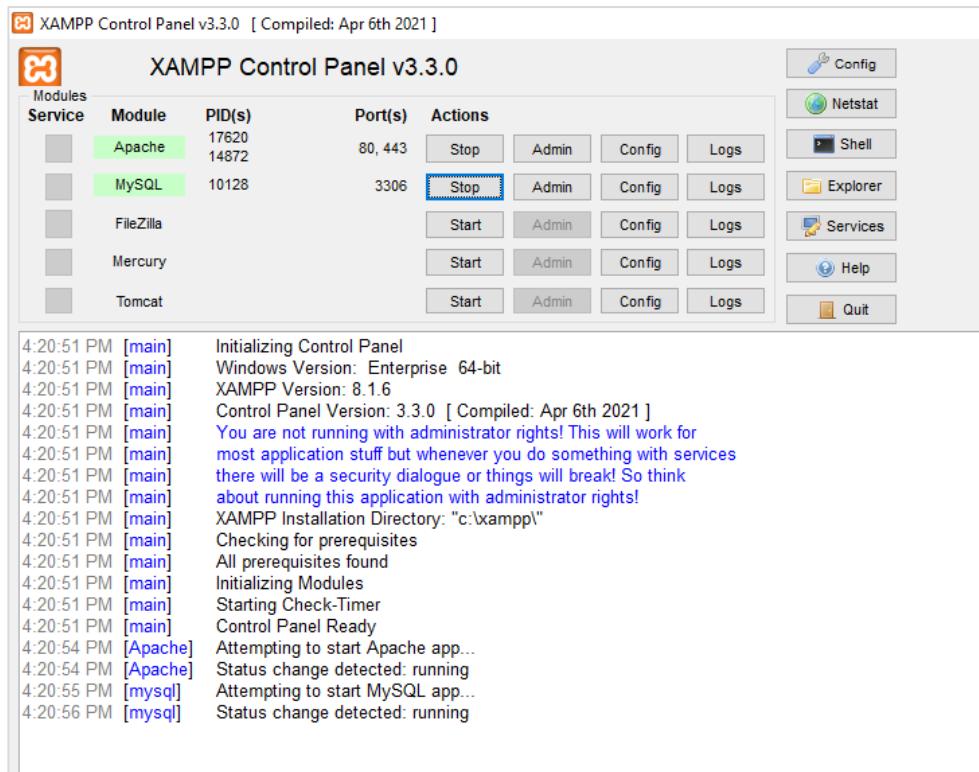


Figure 3: Screenshot of XAAMP Server

✓ Flow Drawio – To create a flow chart

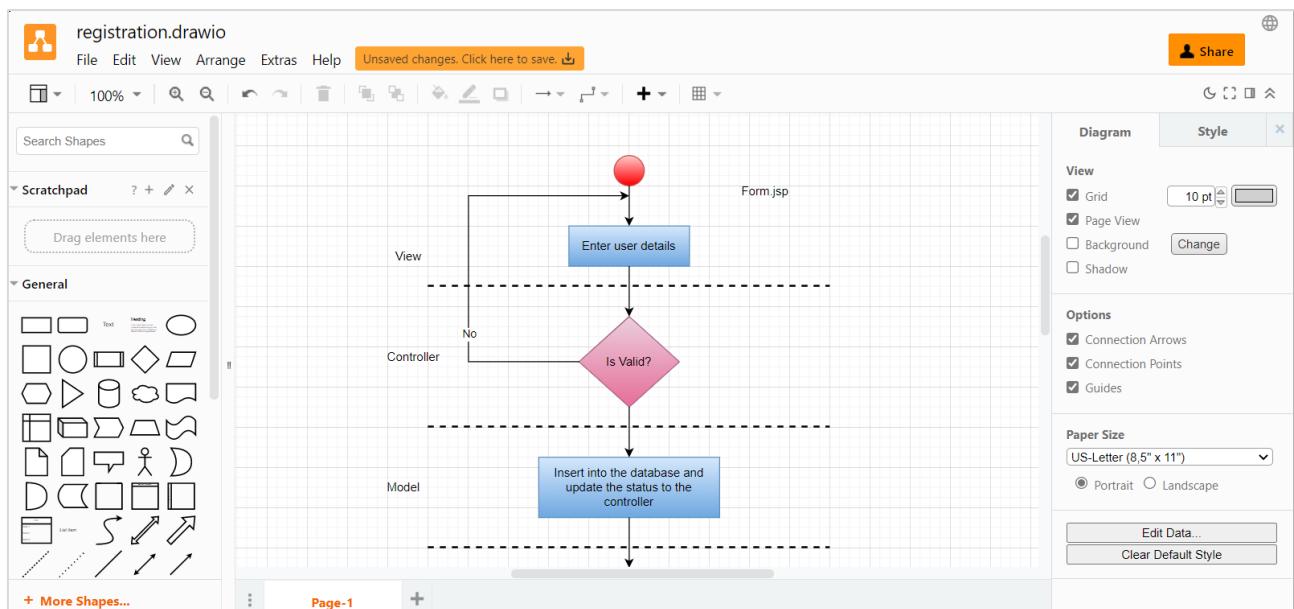


Figure 4: Screenshot of Flow Drawio

• Microsoft Word – To create project report

File Tools View chathushi.docx - Word

2. System

a. List down all the modules to be developed

- Private Module - Administrator - Manage user Data and Bulk Emailing.
- Public Module - Software Programmers
- Registration Page
- Login Page
- Forgot Password Page
- User Profile Page
- Update Profile Page
- Search User Page

b. List down all the pages in each of the modules and briefly describe the purpose

- Registration Page – Using this Page visitors can register to the community portal by filling out the registration form.
- Login Page – Registered users can log in to the community portal by entering their email/username and password

c. Note down which are Create, Edit & View pages.

- Insert - Create Page for users to register and data are inserted into database
- Select - View Page for the users can view the others profile page
- Update - Update Page for the user to edit and update their profile page
- <you may answer this way>

d. Create the Sitemap of Public & Private Site

a. Module	b. Pages	C. Brief description	c. Operations
Registration page	Register.jsp	Getting register data from new users	View
Login Page	Login.jsp	Provide the series where can users can login into their account	Insert
Search User Page	Search-user.jsp	To search for other users in the community portal	Search
User Profile	User-profile.jsp	To see the details of user profile	View
Update profile page	Update.jsp	update the details	Update
Forgot password page	Forgot-password.jsp	Reset the user password	Insert

Figure 5: Screenshot of Microsoft Word

• Microsoft PowerPoint – To create project Presentation

AC_WD_MS_Java_ITSF_Project_Presentation_V3.ppt [Protected View] - PowerPoint

File Home Insert Draw Design Transitions Animations Slide Show Review View Help Tell me what you want to do

PROTECTED VIEW Be careful—files from the Internet can contain viruses. Unless you need to edit, it's safer to stay in Protected View.

1. Development Tools LITHAN

Design, Develop, Implement & Document Community Portal Website

Module Project

Module: Web Development Foundations

Course: NICF Advanced Certificate in Software & Applications (Development & Deployment)

Start Date : End Date : Submission Date : Learner Name : Enrollment ID : Presentation Date :

2. Business Process LITHAN

Block Diagram of Various Business Process

3. Application Block Diagram LITHAN

Block Diagram of the Overall Application

Document History LITHAN

Version Number	Effective Date of release	Summary of Included Changes	Author
1	4 th March 2016	First Edition	Satya CVS
2	08 th Sep 2017	Changed for Module 5	Shrinivas K R
3	20 th Jul 2018	Changed for Module 5 RQF	Shrinivas K R

Contents LITHAN

S. No.	Description
01	Development Tools
02	Business Process
03	Application Block Diagram
04	Modular Structure
05	Models, Controllers & Classes Developed
06	UI Design
07	Includes
08	Software Development Methodology
09	Project Plan
10	Application Screen Shots
11	Project Milestones & Tasks
12	Milestone Feedback & Action Taken
13	Modifications Made On Feedback
14	Project Result
15	Proposed Improvements

Figure 6: Screenshot of Microsoft PowerPoint

Task 1

3. Block diagram of the system

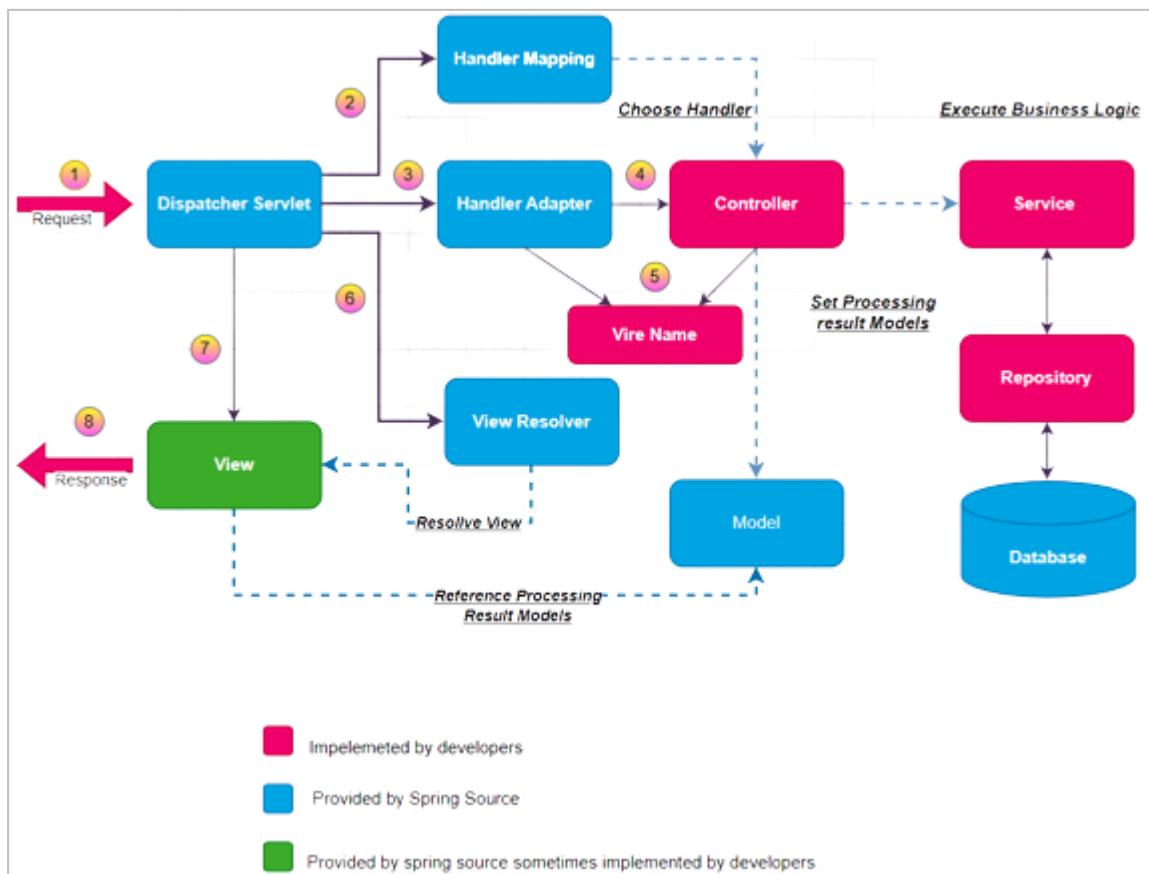


Figure 7:Block Diagram of System

4. Case Diagram

✓ Login

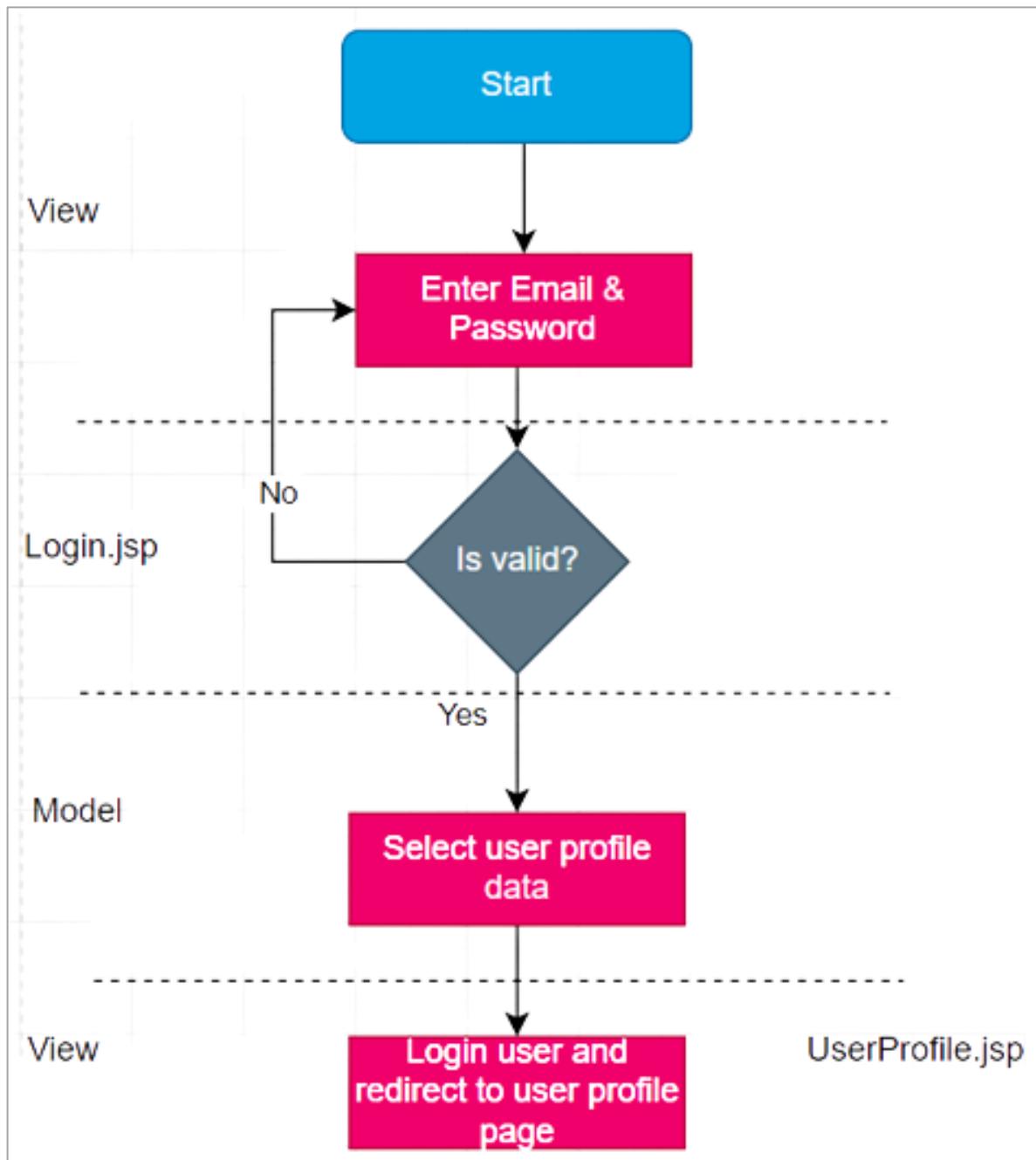


Figure 8;Activity diagram of Login

✓ **Registration**

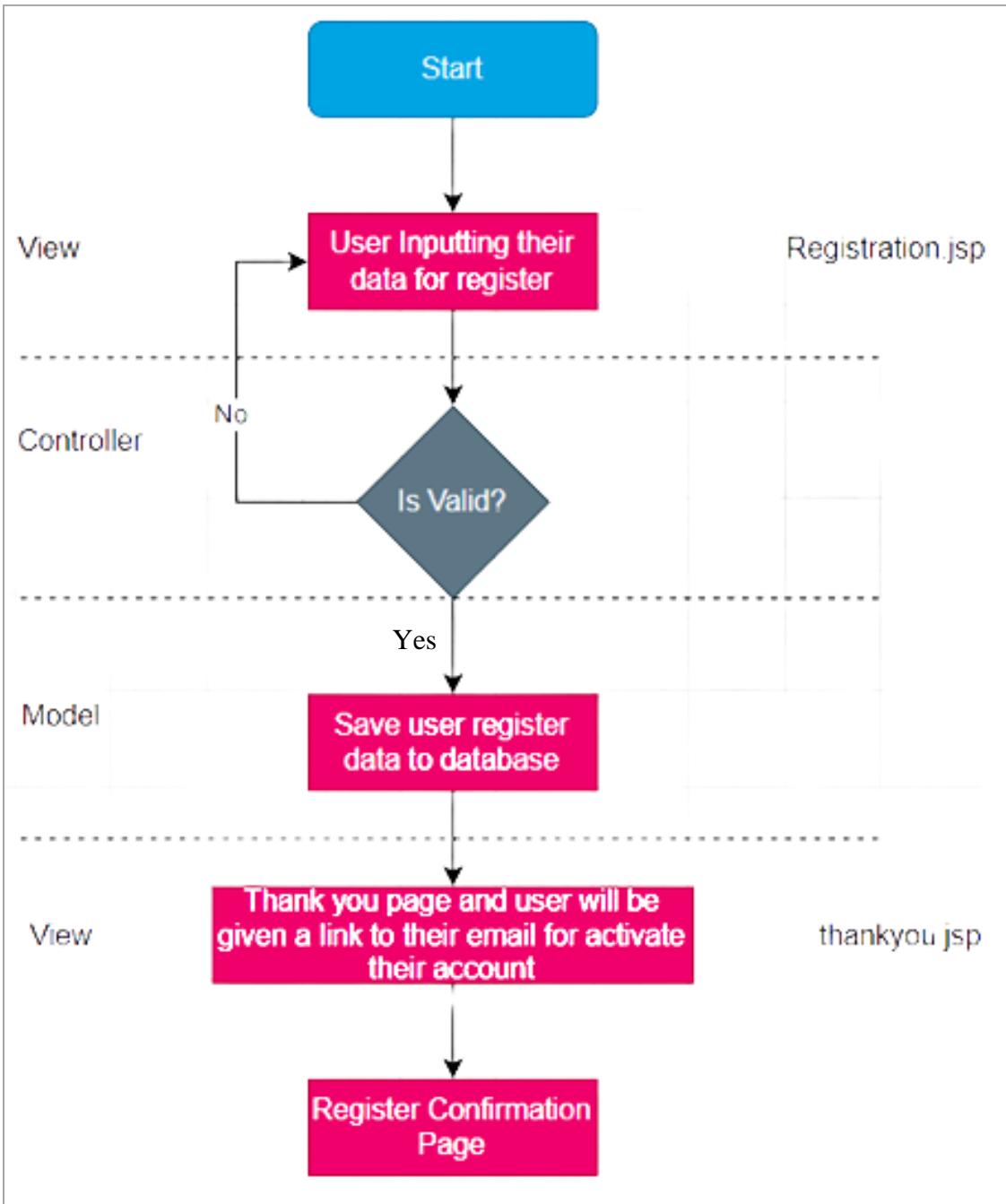


Figure 9:Activity diagram of Registration

✓ **Forgot Password**

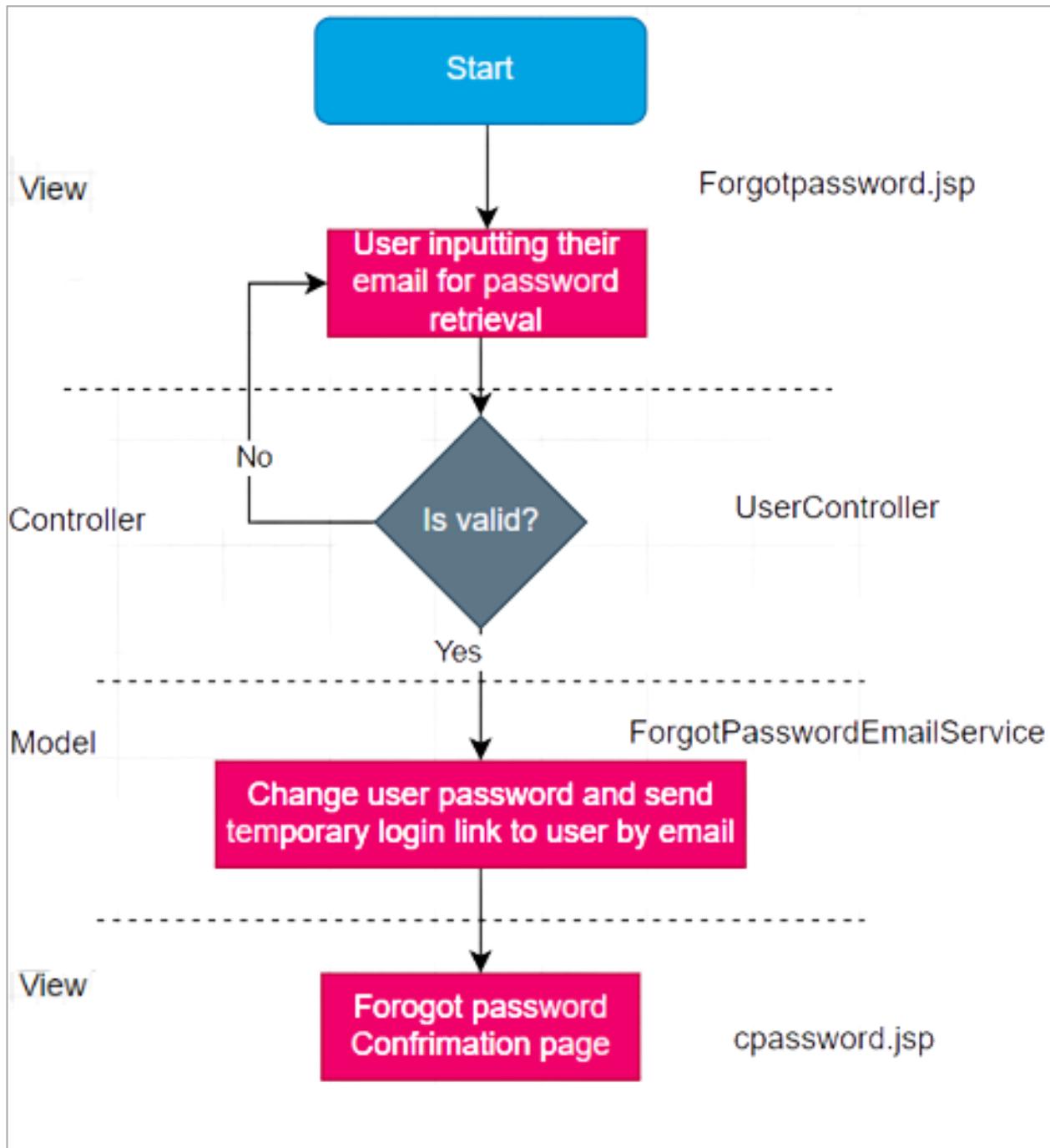


Figure 10: Activity Diagram of Forgot Password

✓ **Search User**

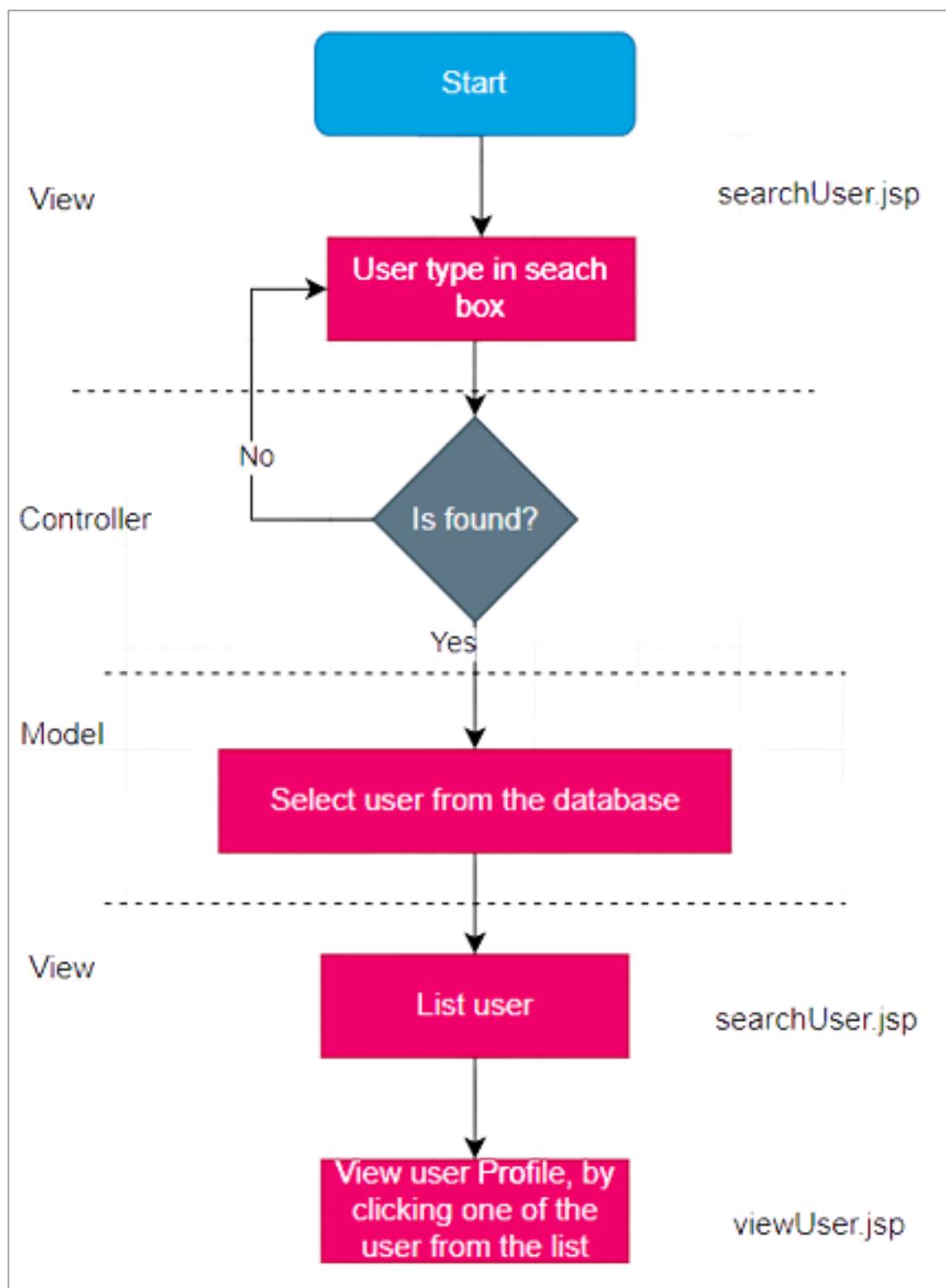


Figure 11: Activity Diagram of Search User

✓ **Admin Login**

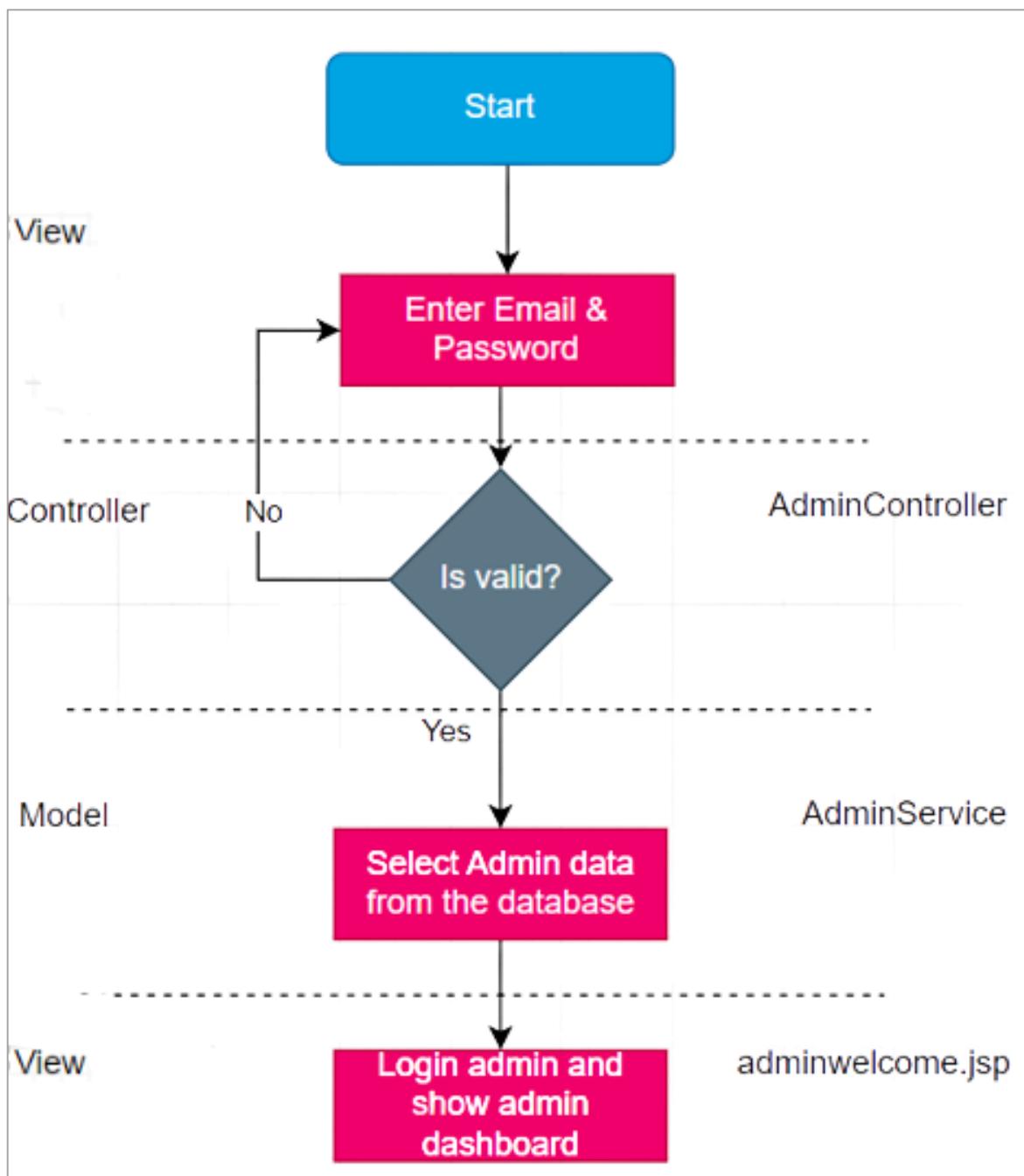


Figure 12:Activity diagram of Admin Login

5. Architecture of the system

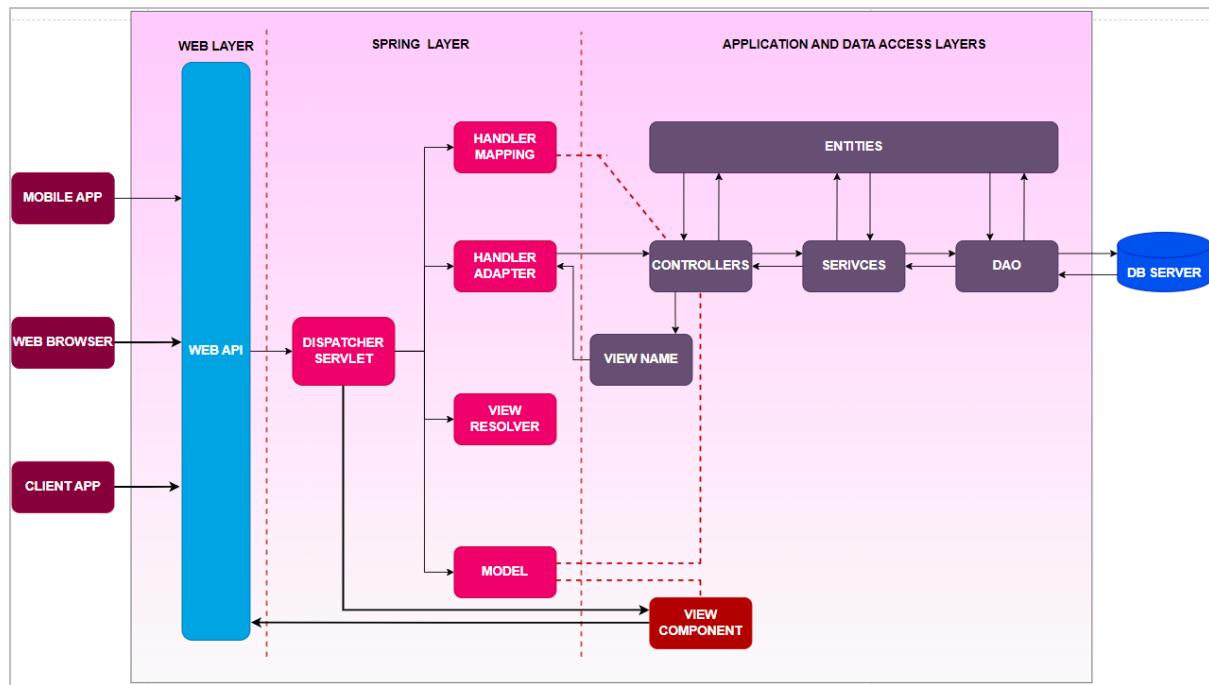


Figure 13: Architecture of the system

5.1 Explanation of Architecture of the application

All the jsp pages of this ABC community portal is handled by the controller. The controller requests a response from the service and after receiving the response from the service it calls the dao layer which is the data access layer. Then Dao layer takes data from Database. After this whole process application will work properly

6. Developed Modules

6.1 Private Module - Administrator - Manage user Data and Bulk Emailing.

6.2 Public Module - Software Programmers

- Registration Page
- Login Page
- Forgot Password Page
- Forgot Password Reset
- User Profile Page
- User Home Page
- Edit User Profile
- Search User Page

7. All the pages in each of the modules and Description

✓ Public Modules

- **Index Page** – Landing Page of community portal
- **Registration Page** – Using this Page visitors can register to the community portal by filling out the registration form.
- **Login Page** – Registered users can log in to the community portal by entering their email/username and password
- **Forgot Password Page** - If user forgot their password, they can reset the password using this
- **Forgot Password Confirmation**
- **User Home page – After successful login, the user home page will appear to the user**
- **User Profile Page** – User profile page which include edit profile, search user
- **Update Profile Page** – Using this page users can update their profile
- **Search User Page** – Users can search other users in the community portal

Private Module

- **Admin Home Page**

- **Admin View User page** – Using this page admin can view, edit, and delete users
- **Bulk Email** – Using this page admin can send mails to users

8. Create, Edit & View pages.

Insert - Create a Page for users to register and data are inserted into the database

Select - View Page for the users can view the other's profile page

Update - Update Page for the user to edit and update their profile page

a. Module	b. Pages	Brief description	c. Operations
Registration page	Registration.jsp	Getting register data from new users	Insert
Login Page	Login.jsp	Provide the series where can users can login into their account	Insert
Search User Page	Searchuser.jsp	To search for other users in the community portal	Search/Select
User Profile	UserProfile.jsp	To see the details of user profile	View
Update profile page	Update.jsp	update the details	Update
Forgot password page	Forgot-password.jsp	Reset the user password	Insert
Bulk Email	Emailform.jsp	Send emails to users	Insert

9. Sitemap of Public & Private Site

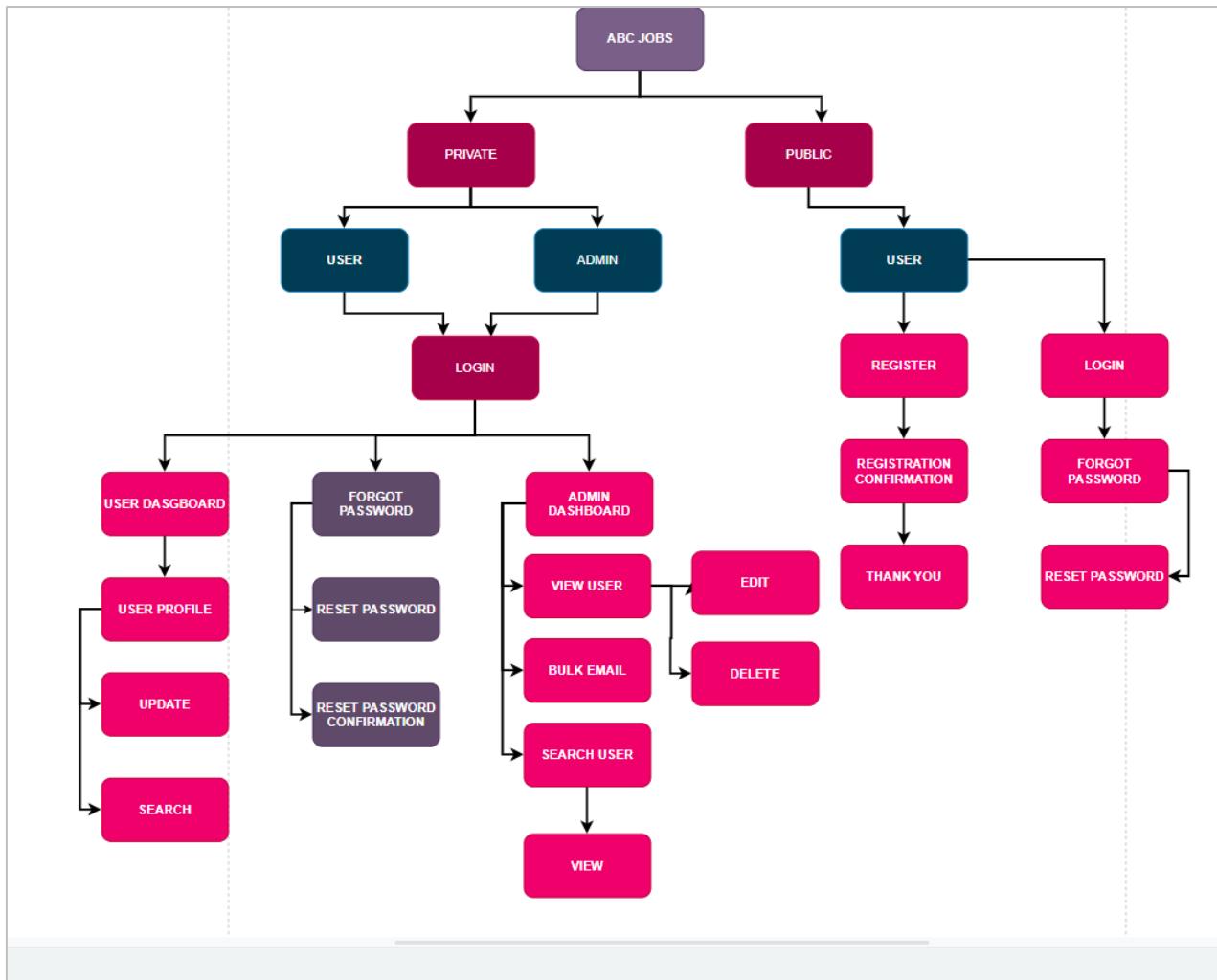


Figure 14:Sitemap of Public and Private Site

10. Technical Environment Requirements

10.1 Hardware Requirements

- Processor: Inter(R) Core(TM) i5
- RAM: 12GB
- 1TB Disk Space
- MySQL Server 8.0.30
- Apache Tomcat 9

10.2 Software Requirements

- Development Environment - JDK 8.1
- Framework - Spring MVC
- Database Server - MySQL Server 8.0.30
- Firewall Settings - Port 9001-Http, Port 3306 - Database

10.3 System Integration Requirements

- JDBC-JDBC Connector 8.0.11
- Spring security Dependency

10.4 Portability Requirements

- Multi OS- Windows, Linux, and Mac
- Multiple Browsers - Chrome, Safari, Microsoft Edge
- Multiple Devices - Desktop, Tablet, Mobile

10.5 Maintainability Requirements

- Back up the database every 6 hours.
- Vulnerability check – periodical scanning.
- Each request should be processed within 10 seconds.

10.6 Performance Requirements

Concurrent number of users- 100

Loading time – 15secs

10.7 Security Requirements

.permit All()	.hasAnyRole	.hasRole("ABC Administators")	.hasRole("Software Programmers")
----------------------	--------------------	--------------------------------------	---

home.jsp	Home.jsp	Admin.jsp	Profile.jsp
Update.jsp	Update.jsp	Login.jsp	Forgot-password.jsp
	Login.jsp		

12. Login & Logout mechanism

12.1 Login

Using the Spring security login process is accomplished. user.java has the data variables in store in the database. when user input their username and password it gets the information from the base of the records and select if a user is an admin or software programmer. After that, it shows the perfect pages for the user

12.2 Log out

A simple disconnection lets in you to leave the portal. this indicates terminating access to a computer system or internet site and notifying the pc or internet site that the user needs to end the login session.

Task 2

13. Eclipse Installation

Step 1 - Install JDK

To use Eclipse for Java programming, developers need to install Java Development KIT.

- Step 1 - Download JDK
- Step 2 - Install JDK
- Step 3 Include JDK's "bin" Directory in the path
- Step 4 - Verify the JDK Installation

Step 2 - Open Browser and go to <https://www.eclipse.org/>

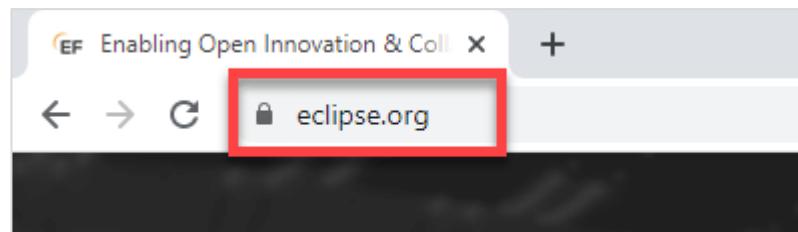


Figure 15: Screenshot of address bar in browser

Step 3 - Click on download

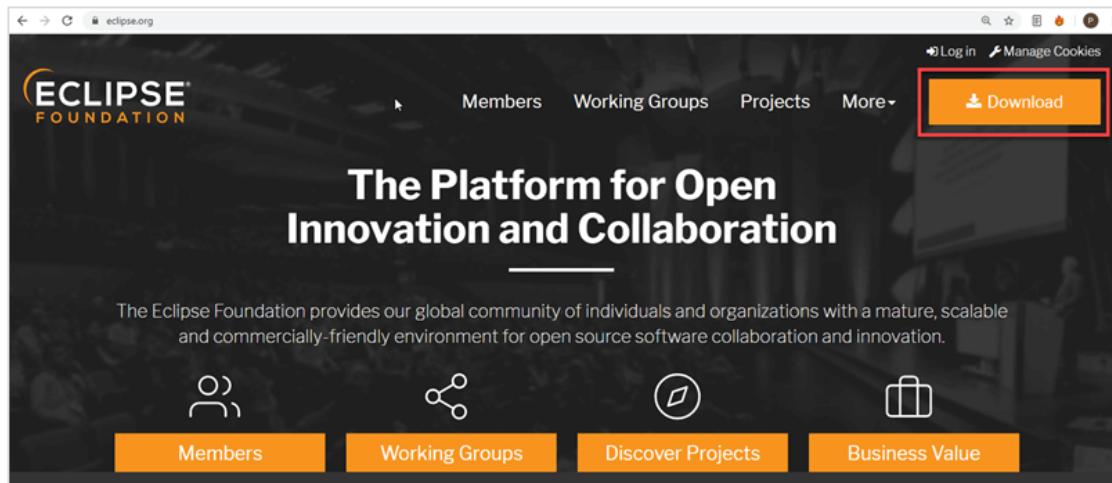


Figure 16: Screenshot of Eclipse installation step 3

Step 4 - Then click on download 64bit button

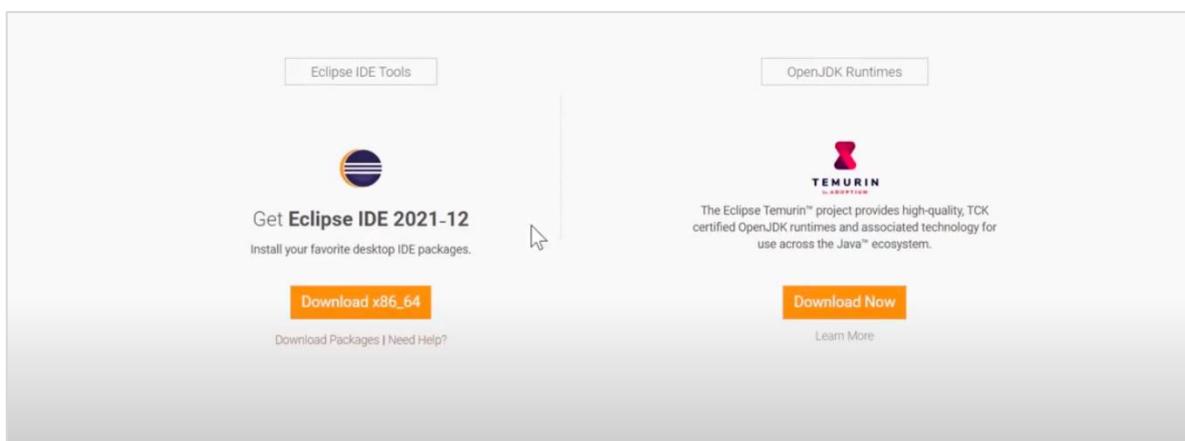


Figure 17: Screenshot of Eclipse installation step 4

Step 5 - Click on download button

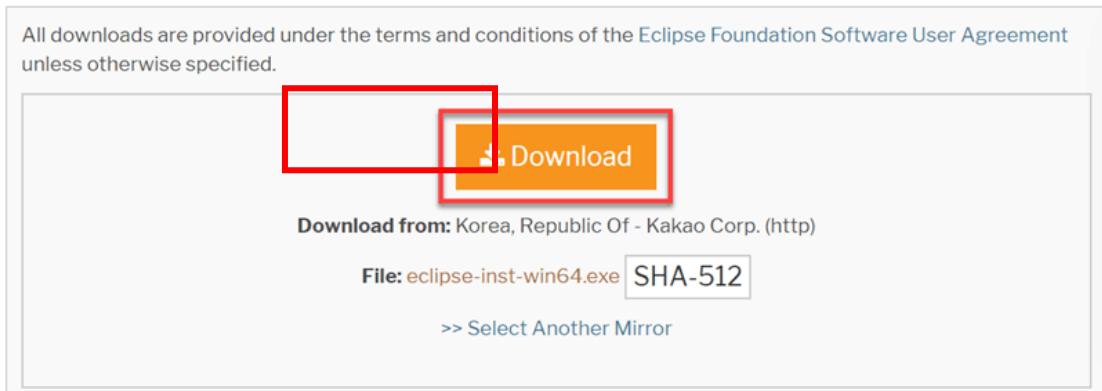


Figure 18: Screenshot of eclipse installation step 5

Step 6 - Install Eclipse

Click on “downloads” in Windows file explorer.

Click on “eclipse-inst-win64.exe” file.

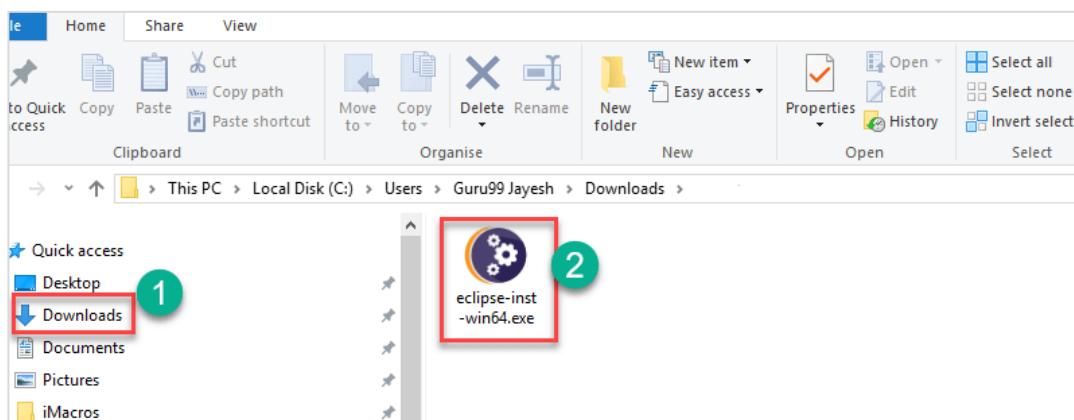


Figure 19: Screenshot of eclipse installation step 6

Step 7 - Run Application

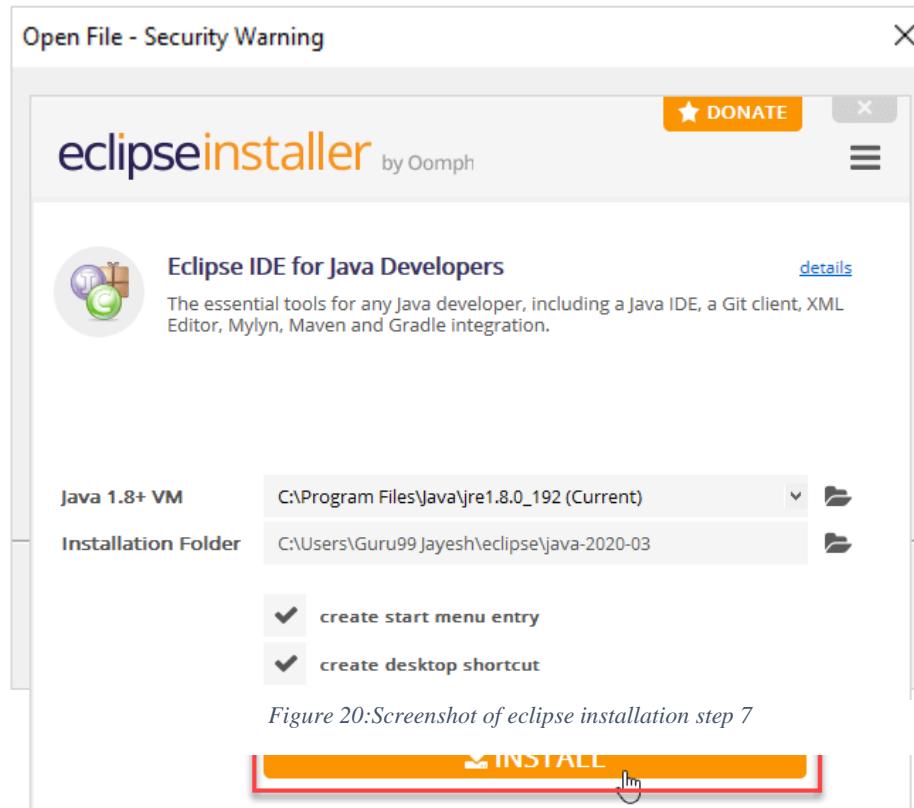


Figure 20: Screenshot of eclipse installation step 7



Figure 21: Screenshot of eclipse installation step 8

Step 8 - Click on the install button

Step 9 - Click on the launch button.

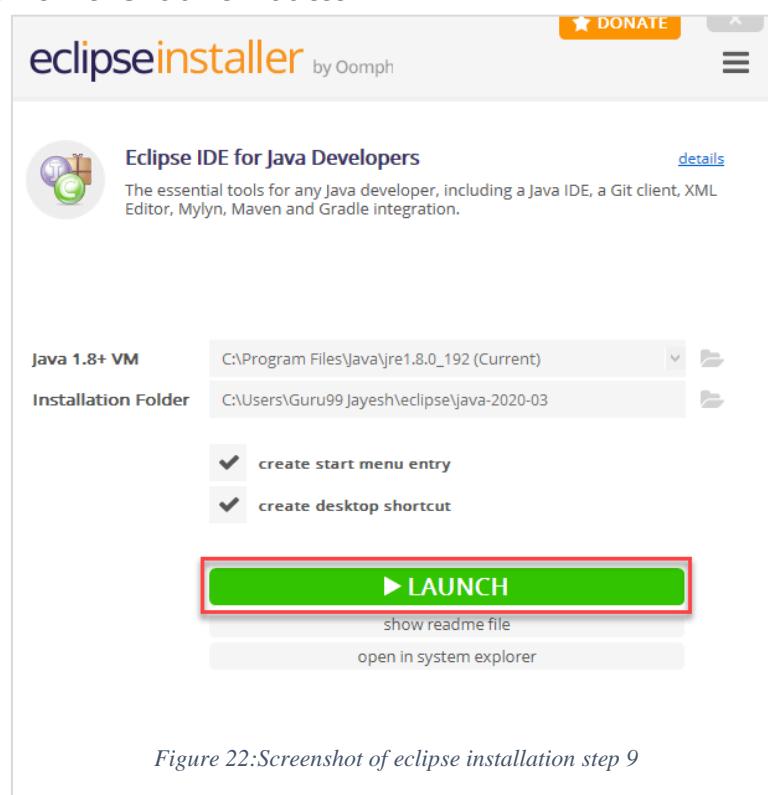


Figure 22: Screenshot of eclipse installation step 9

Step 10 - Click on the Launch button.

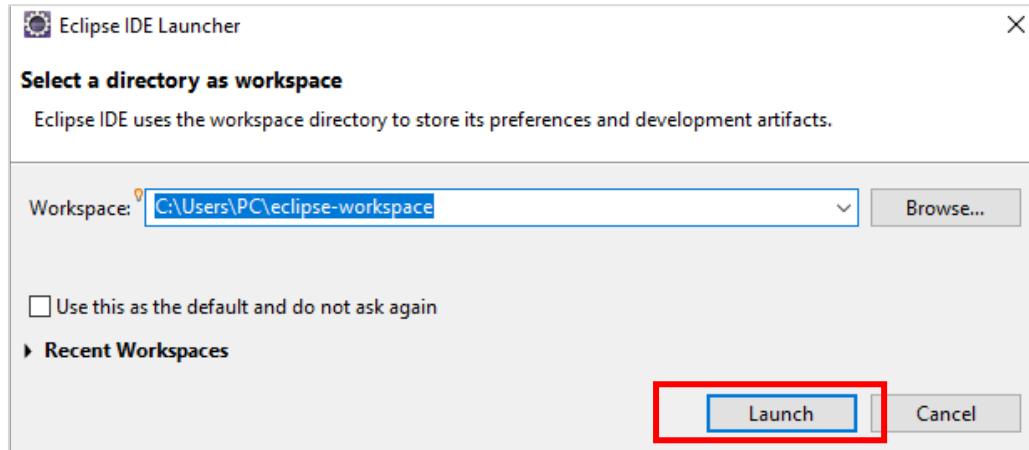


Figure 23: Screenshot of eclipse installation step 10

Step 11 -

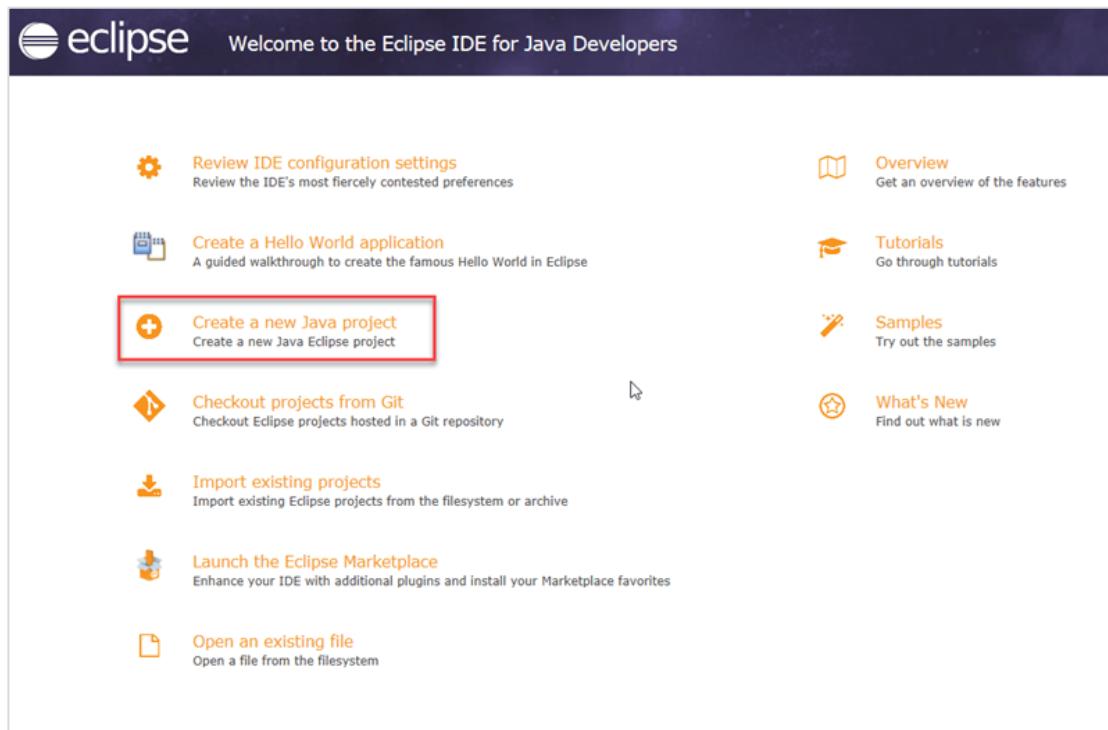


Figure 24: Screenshot of eclipse installation step 11

Task 3

14. Features in IDE

14.1 Smart Code Completion

We get code of completion as quickly as we start typing in eclipse, we don't always need to apply a keyboard shortcut to deliver the suggestions. suggestions are also context sensitive so that they must make feel for the current place we're typing

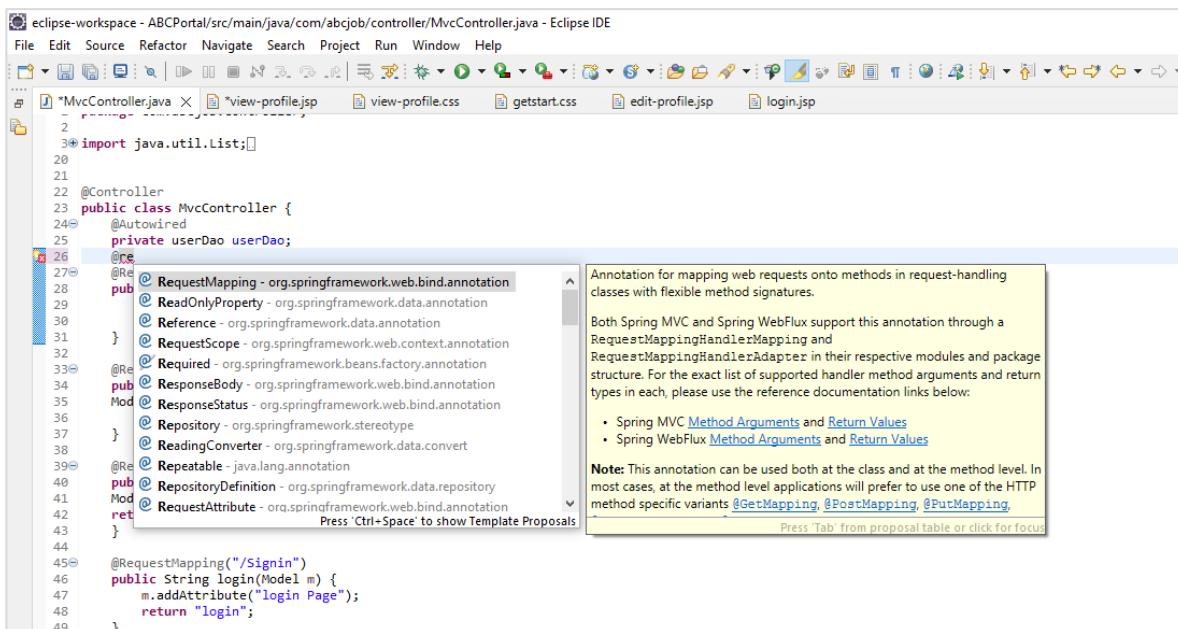


Figure 25: Screenshot of smart code completion in eclipse

14.2 Quick Fixes

Each time Ide detects an error, the eclipse highlights the error the use of a wavy red line underneath the offending code and a marker in the left editor margin. moving the mouse over the underlined code or the marker will indicate the error. The marker may be selected with the left mouse to set off the quick restoration pop-up, indicating moves that may be undertaken to repair the error. rather, pressing Ctrl+1 will activate brief restore from the keyboard.

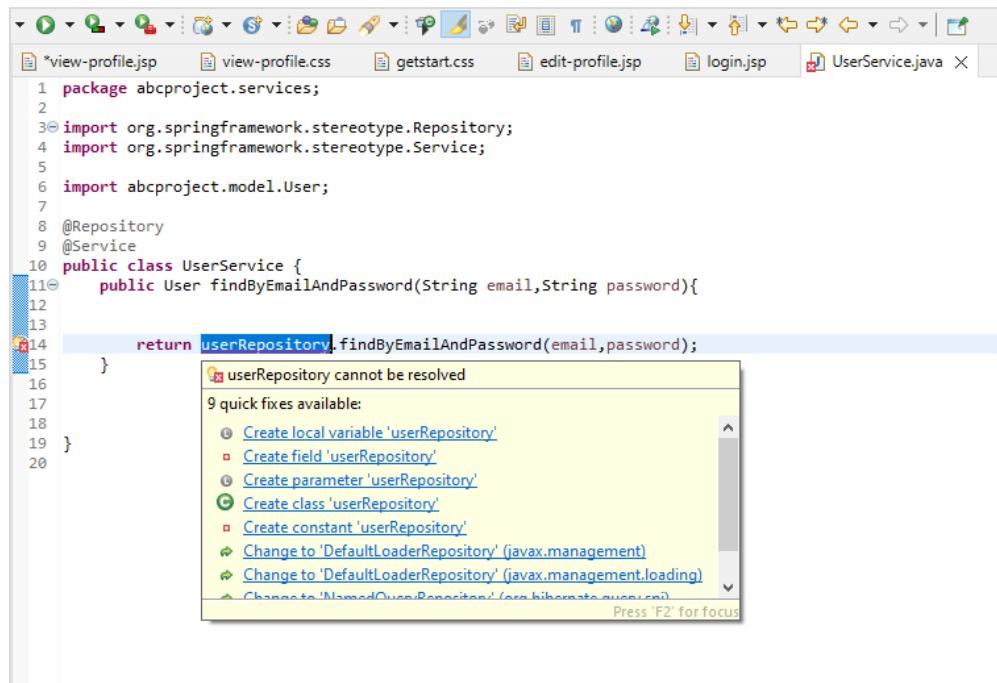


Figure 26: Screenshot of quick fixes in eclipse

14.3 Getter and Setter

Getters and setters are used to shield your data, in particular when developing classes. For each example variable, a getter method returns its value while a setter method sets or updates its value. Given this, getters and setters are also called accessors and mutators, respectively

```

    public String getFristname() {
        return fristname;
    }
    public void setFristname(String fristname) {
        this.fristname = fristname;
    }
    public String getLastname() {
        return lastname;
    }
    public void setLastname(String lastname) {
        this.lastname = lastname;
    }
    public String getEmail() {
        return email;
    }
    public void setEmail(String email) {
        this.email = email;
    }
}

```

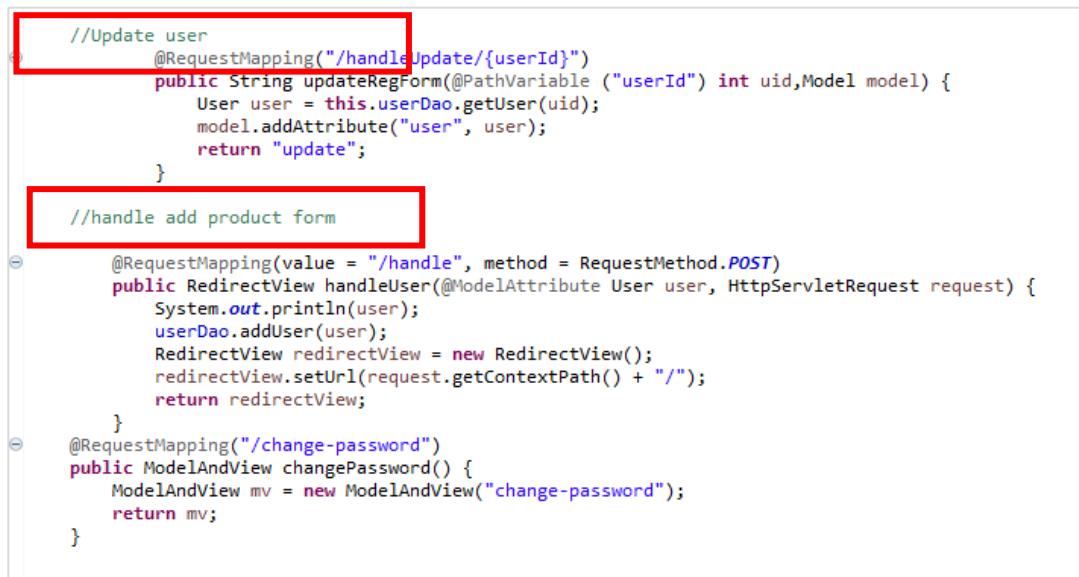
Figure 27: Screenshot of coding & documentation Standards in eclipse (getters and setters)

Task 4

15. Coding & Documentation Standards

15.1 Comments

Comments are textual content notes added to the program to offer explanatory statistics about the supply code. They're utilized in a programming language to record the program and remind programmers of what difficult things they simply did with the code and additionally allows the later generation for know-how and upkeep of code.



```
//Update user
    @RequestMapping("/handleUpdate/{userId}")
    public String updateRegForm(@PathVariable ("userId") int uid, Model model) {
        User user = this.userDao.getUser(uid);
        model.addAttribute("user", user);
        return "update";
    }

//handle add product form

    @RequestMapping(value = "/handle", method = RequestMethod.POST)
    public RedirectView handleUser(@ModelAttribute User user, HttpServletRequest request) {
        System.out.println(user);
        userDao.addUser(user);
        RedirectView redirectView = new RedirectView();
        redirectView.setUrl(request.getContextPath() + "/");
        return redirectView;
    }

    @RequestMapping("/change-password")
    public ModelAndView changePassword() {
        ModelAndView mv = new ModelAndView("change-password");
        return mv;
    }
```

Figure 28: Screenshot of coding & documentation Standards in eclipse (comments)

15.2 Naming Convention

Class Name - Class names should be nouns, in mixed case with the first letter of each internal word capitalized.

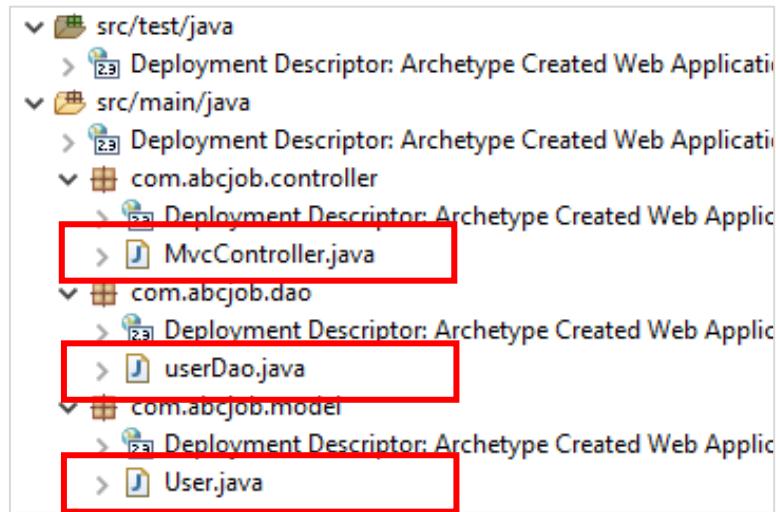


Figure 29: Screenshot of coding & documentation Standards in eclipse
(Naming Conventions)

15.3 File and Folder Organization

By keeping files and folders in order, it is possible to get the necessary information quickly when using coding.

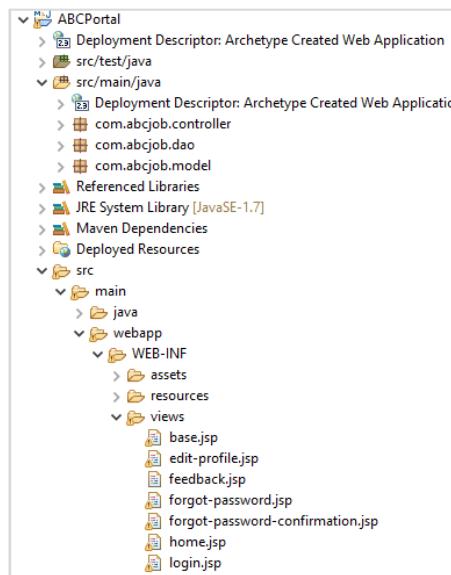


Figure 30: Screenshot of coding & documentation Standards in eclipse (File and Folder Organization)

16. Documentation tool and its configuration.

Step 1 - In Eclipse, go to **File > Export**

Step 2 - Expand Java and select Javadoc and then click next

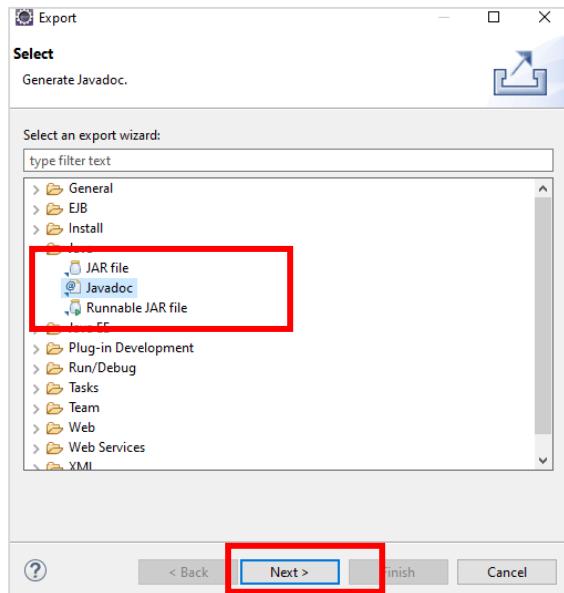


Figure 31:Javadoc configuration step 2

Step 3 - Select your project and package. Then inside the right pane, select the classes you need to be included in the Javadoc

Step 4 - Select which visibility option you want: Private, Package, Protected, or Public.

Step 5 - Make sure the Use standard doclet radio button is selected

Step 6 - Click the Browse button and select the output location in which you want the Javadoc generated and click next

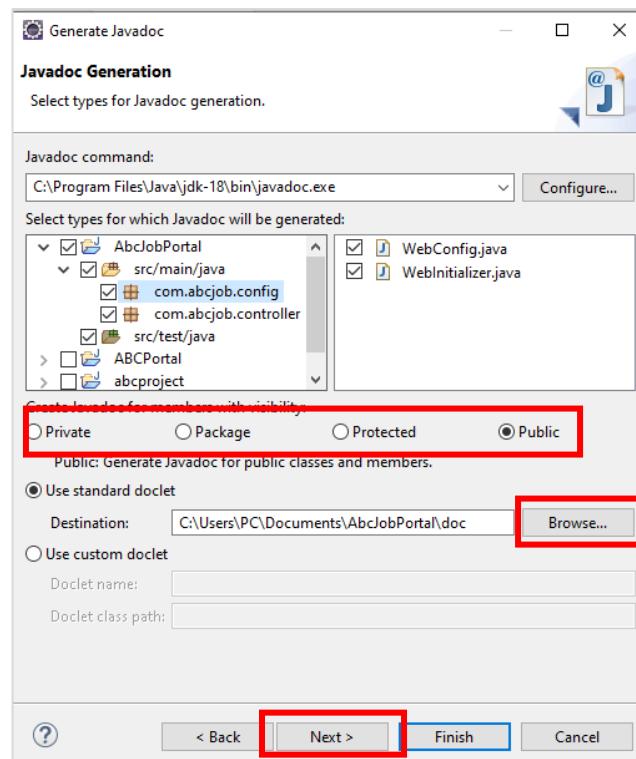


Figure 33: Screenshot of Javadoc configuration step 3,4,5,6

Step 7 – Click Next

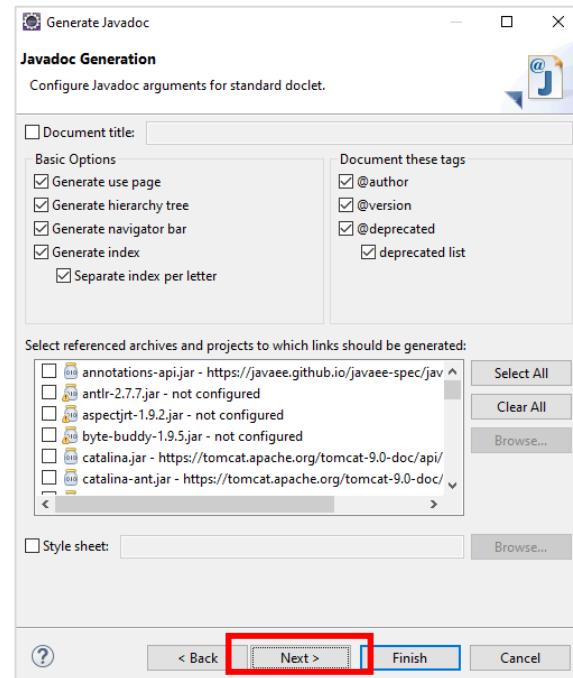


Figure 32: Screenshot of Javadoc configuration step 7

Step 8 - Select an HTML page that you want to be your overview page in Javadoc and click finish

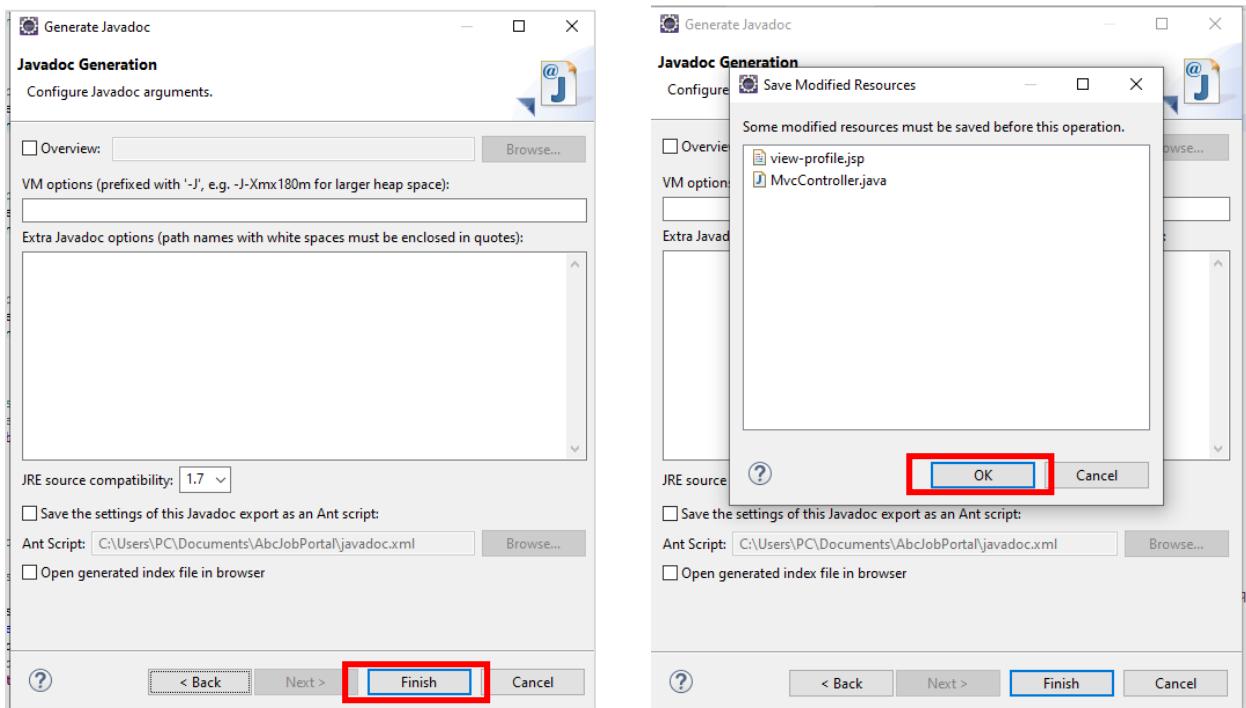


Figure 34: Javadoc configuration step 8

Step 9 – Javadoc and error checking

```
eclipse-workspace - ABCPortal/src/main/java/com/abcjob/controller/MvcController.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Data Source Explorer Console X Servers
terminated: Javadoc Generation
C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\config\WebConfig.java: warning: no comment
public class WebConfig implements webConfig{
C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\config\WebConfig.java: warning: use of default constructor, which does not provide a comment
public class WebConfig implements webConfig{
C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\config\WebConfig.java: warning: no comment
public InternalResourceViewResolver viewResolver() {
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\config\WebInitializer.java...
C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\config\WebInitializer.java: warning: no comment
public InternalResourceViewResolver extends AbstractAnnotationConfigDispatcherServletInitializer {
C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\config\WebInitializer.java: warning: use of default constructor, which does not provide a comment
public class WebInitializer extends AbstractAnnotationConfigDispatcherServletInitializer {
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\package-summary.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\package-tree.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\package-summary.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\package-tree.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\config\class-use\We...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\config\class-use\We...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\class-use\...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-use.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-tree.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-classes-index.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-files-index.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-files-index-2.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-files-index-3.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-files-index-4.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-files-index-5.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-files-index-6.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-files-index-7.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-files-index-8.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-files-index-9.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-files-index-10.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-files-index-11.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-files-index-12.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-files-index-13.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-overview-summary.html...
Generating C:\Users\PC\Documents\ABCJobPortal\src\main\java\com\abcjob\controller\index-overview.html...
18 warnings
```

Figure 35: Javadoc configuration step 9

OVERVIEW PACKAGE CLASS USE TREE INDEX HELP

SUMMARY NESTED FIELD CONSTR METHOD DETAIL FIELD CONSTR METHOD SEARCH: X

Package com.abcjob.controller
Class MvcController

```
java.lang.Object<@
    com.abcjob.controller.MvcController
    @Controller
    public class MvcController
    extends Object<
```

Constructor Summary

Constructors	Constructor	Description
	MvcController()	

Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method	Description
org.springframework.web.servlet.ModelAndView	changePassword()	
org.springframework.web.servlet.ModelAndView	forgotPassword()	
org.springframework.web.servlet.ModelAndView	forgotPasswordConfirmation()	
org.springframework.web.servlet.view.RedirectView	handleUser(User user, javax.servlet.http.HttpServletRequest request)	
org.springframework.web.servlet.ModelAndView	home()	
org.springframework.web.servlet.ModelAndView	login()	
String<@	login(org.springframework.ui.Model m)	
String<@	processLogin(User user)	
String<@	processRegister(User user)	
org.springframework.web.servlet.ModelAndView	register()	
String<@	register(org.springframework.ui.Model m)	

Figure 36: Screenshot of Javadoc

Task 5

17. HTML Pages Source Code

17.1 Registration Page (register.jsp)

```

85          </div>
86
87          <!-- phone -->
88          <div class="form-group"
89              style="width: 36rem; height: 4rem">
90              Phone <label for="phone"></label> <input
91                  type="number"
92                  name="contact" class="form-control" id="contact"
93                  required />
94          </div>
95
96          <!-- city -->
97          <div class="form-group"
98              style="width: 36rem; height: 4rem">
99              City <label for="city"></label> <input type="text"
100                 name="city" class="form-control" id="city" required
101             />
102         </div>
103
104         <!-- Password -->
105         <div class="form-group"
106             style="width: 36rem; height: 4rem">
107             Password <label for="password"></label> <input
108                 type="password" name="password" class="form-
109                 control"
110                 id="pass" required />
111         </div>
112
113         <!-- Password -->
114         <div class="form-group"
115             style="width: 36rem; height: 4rem">
116             Repeat Password <label for="repPassword"></label>
117             <input
118                 type="repPassword" name="re_pass" class="form-
119                 control"
120                 id="re_pass" required />
121         </div>
122
123         </div>
124         </div>
125         <button type="submit" class="btn btn-success btn-block mt-
126             4">Register</button>
127     </form>
128
129     </div>
130     </div>
131     <div
132         class="col-md-10 col-lg-6 col-xl-7 d-flex align-items-center
133         order-1 order-lg-2">
134         
136     </div>
137 </div>
138 </section>
139
140
141
142     <%@ include file="footer.jsp"%>
143     <script
144         src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.9.3/dist/umd/popper.min.js"
145         integrity="sha384-
eMNCOe7tC1doHpGoWe/6oMVeemdAVTMs2xqW4mwXrXsW0L84Iytr2wi5v2QjrP/xp"
146         crossorigin="anonymous"></script>
147     <script
148         src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.0/dist/js/bootstrap.min.js"
149         integrity="sha384-
cn7l7gDp0eyniUwwAZgrzD06kc/tftFf19TOAs2zVinnD/C7E91j9yyk5//jjpt/"
150         crossorigin="anonymous"></script>
151 </body>
152 </html>
```

17.2 Login Page (login.js)

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <%@ taglib prefix="c" uri="http://java.sun.com/jstl/core"%>
4 <%@ taglib prefix="f" uri="http://www.springframework.org/tags/form"%>
5
6 <!DOCTYPE html>
7 <html>
8 <link href="resource/css/style.css" rel="stylesheet" />
9 <link href="resource/css/login.css" rel="stylesheet" />
10
11 <head>
12 <meta charset="ISO-8859-1">
13 <title>Login</title>
14 <script src="https://kit.fontawesome.com/848830b72b.js"
15   crossorigin="anonymous"></script>
16 <link rel="stylesheet"
17   href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-
18   beta1/dist/css/bootstrap.min.css" />
19 </head>
20 <body>
21   <nav class="navbar navbar-expand-lg text-bg-light sticky-top">
22     <div class="container">
23       <a class="navbar-brand fw-bold fs-3"
24         href=<%=request.getContextPath()%>>ABC Jobs</a>
25       <button class="navbar-toggler" type="button"
26         data-bs-toggle="collapse" data-bs-target="#navbarNav"
27         aria-controls="navbarNav" aria-expanded="false"
28         aria-label="Toggle navigation">
29         <span class="navbar-toggler-icon"></span>
30       </button>
31       <div class="collapse navbar-collapse text-center" id="navbarNav">
32         <ul
33           class="navbar-nav ms-auto me-0 me-lg-auto fw-semibold text-uppercase">
34
35           </ul>
36           <div class="nav-button d-flex flex-column flex-lg-row">
37             <a href="login"
38               class="btn btn-outline-secondary me-0 me-lg-3 mb-3 mb-lg-0">Login</a>
39             <a href="registration" class="btn btn-success mb-3 mb-lg-0">Register</a>
40           </div>
41         </div>
42       </div>
43     </nav>
44   </head>
45
46 <div class="container mt-5">
47   <div class="card" style="height: 40rem;">
48
49     <c:if test="${success != null}">
50       <div style="color: green; font-size: 20px;" role="alert">
51         <c:out value="${success}" />
52       </div>
53     </c:if>
54     <c:if test="${error != null}">
55       <div style="color: red; font-size: 20px;" role="alert">
56         <c:out value="${error}" />
57       </div>
58     </c:if>
59   </div>
```

```

60
61     <main>
62         <section class="vh-100">
63             <div class="container py-5 h-100">
64                 <div
65                     class="row d-flex align-items-center justify-content-center h-100">
66                     <div class="col-md-8 col-lg-7 col-xl-6">
67                         
69                     </div>
70                     <div class="col-md-7 col-lg-5 col-xl-5 offset-xl-1">
71                         <h1>Log in here</h1>
72                         <form action="login" method="post"
73                             class="mt-4 d-flex flex-column" id="form">
74
75                             <!-- Email input -->
76                             <div class="form-outline mb-4">
77                                 Email address <input type="text" id="email" name="email"
78                                     class="form-control form-control-lg" /> <label
79                                     class="form-label" for="form1Example13"></label>
80                             </div>
81
82                             <!-- Password input -->
83                             <div class="form-outline mb-4">
84                                 Password <input type="password" id="password" name="password"
85                                     class="form-control form-control-lg" /> <label
86                                     class="form-label" for="form1Example23"></label>
87                             </div>
88
89                             <div
90                                 class="d-flex justify-content-around align-items-center mb-4">
91                                 <!-- Checkbox -->
92                                 <div class="form-check">
93                                     <input class="form-check-input" type="checkbox" value=""
94                                         id="form1Example3" checked /> <label
95                                         class="form-check-label" for="form1Example3"></label>
96                                     <div class="forgot-password mt-3 mb-3 text-end">
97                                         <a class="text-decoration-none" href="forgotpassform"
98                                             style="position: absolute; top: 450px; right: 50px;">Forgot
99                                             Password?</a>
100                                </div>
101                            </form>
102                            <br> <br>
103                            <p class="p-0 mt-3"
104                                style="position: absolute; top: 435px; right: 320px;">
105                                Don't have an account? <a
106                                    class="text-primary text-decoration-none" href="register">
107                                    Register </a>
108                            </p>
109
110                            <div class="sub" style="width: 150px;">
111                                <!-- Submit button -->
112                                <button type="submit" class="btn btn-success btn-lg btn-block"
113                                    name="signin" id="signin">Sign in</button>
114
115                            </div>
116                        </div>
117                    </div>
118                </div>
119            </div>
120        </section>
121    </main>
122
123
124    <script
125        src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0/dist/js/bootstrap.bundle.min.js">
126    </script>
127
128 </body>
129 </html>

```

17.3 Thank you Page(thank-you.jsp)

```
1  <%@ page language="java" contentType="text/html; charset=UTF-8"
2  pageEncoding="UTF-8"%>
3
4  <%@ taglib prefix="f" uri="http://www.springframework.org/tags/form"%>
5  <!DOCTYPE html>
6  <html>
7  <head>
8  <link href="resource/css/style.css" rel="stylesheet" />
9  <head>
10 <title>thankyou</title>
11 <script src="https://kit.fontawesome.com/848830b72b.js"
12 crossorigin="anonymous"></script>
13 <link rel="stylesheet"
14 href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-
beta1/dist/css/bootstrap.min.css" />
15
16 <link href="assets/css/style.css" rel="stylesheet" />
17 <%@include file=".base.jsp"%>
18
19
20
21 <meta charset="ISO-8859-1">
22 <title>Register</title>
23 </head>
24 <body>
25   <div class="nav-option">
26     <nav class="navbar navbar-expand-lg text-bg-light sticky-top">
27       <div class="container">
28         <a class="navbar-brand fw-bold fs-3"
29           href=<%=request.getContextPath()%>>ABC Jobs</a>
30         <button class="navbar-toggler" type="button"
31           data-bs-toggle="collapse" data-bs-target="#navbarNav"
32           aria-controls="navbarNav" aria-expanded="false"
33           aria-label="Toggle navigation">
34           <span class="navbar-toggler-icon"></span>
35         </button>
36         <div class="collapse navbar-collapse text-center" id="navbarNav">
37           <ul
38             class="navbar-nav ms-auto me-0 me-lg-auto fw-semibold text-uppercase">
39             <li class="nav-item me-0 me-lg-3"><a class="nav-link"
40               href=<%=request.getContextPath()%>>Home</a></li>
41             <li class="nav-item me-0 me-lg-3"><a class="nav-link"
42               href="search-user"> People</a></li>
43             <li class="nav-item me-0 me-lg-3"><a class="nav-link"
44               href="jobs">Jobs</a></li>
45           </ul>
46           <div class="nav-button d-flex flex-column flex-lg-row">
47             <a href="login"
48               class="btn btn-outline-secondary me-0 me-lg-3 mb-3 mb-lg-0">Login</a>
49             <a href="registration" class="btn btn-success mb-3 mb-lg-0">Register</a>
50           </div>
51         </div>
52       </div>
53     </nav>
54   <section id="registration-confirmation" class="mt-5">
55     <div class="container d-flex justify-content-center">
56       <div class="confirmation d-flex flex-column text-center">
57
58         <h2 class="fw-semibold">
59           Thank You <br> For <br> Registration
60         </h2>
61         <section id="registration-confirmation" class="mt-6">
62           <div class="container d-flex justify-content-center">
63
64             <div class="confirmation d-flex flex-column text-center">
65
66               <div class="wrappers"></div>
67
68               
70               <p class="text-secondary">
71                 Your account has been created successfully<br> A
72                 verification email has been send to your registered email
73                 address.<br> Please click on the verification link
74                 included in the email to activate your account
75               </p>
76
77               <div class="wrapper">
78                 <a href="login" class="btn btn-success mt-3">CONTINUE</a>
79               </div>
80
81             </div>
82           </div>
83         </section>
84
85       </div>
86     </div>
87   </section>
88   <%@ include file="footer.jsp"%>
89 </body>
90 </html>
```

17.4 Forgot Password Page (forgot-password.jsp)

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2  pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5 <link href="resource/css/style.css" rel="stylesheet" />
6
7 <head>
8 <meta charset="UTF-8">
9 <title>Insert title here</title>
10 <%@include file=".base.jsp"%>
11
12 </head>
13 <body>
14   <%@ include file="headbeforelogin.jsp"%>
15
16   <div class="container padding-bottom-3x mb-2 mt-5">
17     <div class="row justify-content-center">
18       <div class="col-lg-5 col-md-6">
19         <div class="forgot">
20
21           <h2>Forgot your password??????</h2>
22           
24
25           <p class="mb-3 text-secondary">
26             Enter the email address associated with your account.<br> We
27             will email you a link to reset your password
28           </p>
29
30         </div>
31         <form action="emailConfirm" method="post" id="formValidation">
32
33           <div class="form-group">
34             <label for="exampleInputEmail1" class="font-weight-bold"></label>
35             <input type="email" class="form-control" name="email"
36               id="exampleInputEmail1" aria-describedby="emailHelp"
37               placeholder="Enter email" required>
38           </div>
39
40           <div class="container text-center mb-5">
41             <input type="submit" value="Reset Password"
42               onclick="submitData();" class="btn btn-success  mt-9">
43           </div>
44         </form>
45       </div>
46     </div>
47   </div>
48
49
50   <%@ include file="footer.jsp"%>
51   <script
52     src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.6.0/jquery.min.js">
53
54   </script>
55   <script
56     src="https://cdnjs.cloudflare.com/ajax/libs/jquery-
validate/1.19.3/jquery.validate.min.js"></script>
57   <script type="text/javascript">
58     </body>
59   </html>
60
```

17.5 Forgot Password Confirmation Page (Reset_password.jsp)

```

1 <%@page isELIgnored="false"%>
2 <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
3 <%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%>
4 <!DOCTYPE html>
5 <html>
6 <link href="resource/css/style.css" rel="stylesheet" />
7 <head>
8 <meta charset="UTF-8">
9 <title>Insert title here</title>
10 <link
11 href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
12 rel="stylesheet"
13 integrity="sha384-EVSTQN3+azprG1Anm3QDgpJLIm9Nao6Yz1ztcQTwpSpd3yD65VohhpuuCOmLASjC"
14 crossorigin="anonymous">
15 <style type="text/css">
16 .error {
17 color: #ad1457;
18 }
19 </style>
20 </head>
21 <body class="" style="background: #e2e2e2">
22
23 <div class="container my-5">
24 <div class="row">
25 <div class="col-md-6 offset-md-3">
26 <div class="card m-3 shadow">
27 <div class="card-body p-5">
28 <h1 class="fs-4 fw-bold mb-4 text-center">Reset Password</h1>
29 <form action="${pageContext.request.contextPath }/resetPass"
30 method="post" id="formValidation" modelAttribute="bean">
31
32
33 <div class="form-group">
34 <label for="email"> Email</label> <input name="email"
35 type="text" class="form-control" id="email" value="${emailid}"
36 placeholder="Enter Password" required>
37 </div>
38
39 <div class="form-group">
40 <label for="password"> Password</label> <input name="password"
41 type="text" class="form-control" id="password"
42 placeholder="Enter Password" required>
43 </div>
44 <div class="form-group">
45 <label for="cpassword">Confirm Password</label> <input
46 name="cpassword" type="text" class="form-control"
47 id="cpassword" placeholder="Enter Confirm Password" required>
48 </div>
49
50 <div class="text-center mb-3">
51 <input type="submit" value="Reset Password"
52 onclick="submitData(); class="btn btn-primary mt-2">
53 </div>
54 </form>
55 </div>
56 </div>
57 </div>
58 </div>
59 </div>
60 =
61 <script
62 src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.6.0/jquery.min.js">
63 </script>
64 <script
65 src="https://cdnjs.cloudflare.com/ajax/libs/jquery-validate/1.19.3/jquery.validate.min.js"></script>
66 <script type="text/javascript">
67
68
69
70 $("#formValidation").validate({
71 rules:{
72 password:{ 
73 required : true,
74 minlength : 8
75 },
76 cpassword : { 
77 required : true,
78 minlength : 8,
79 equalTo : '[name="password"]'
80 },
81 },
82 messages : {
83 password : {
84 required : "Please Enter Password"
85 },
86 cpassword : {
87 required : "Please Enter Confirm Password"
88 }
89 },
90 submitHandler : function(form) {
91 form.submit();
92 },
93
94 });
95
96
97 });
98
99
100 $(function () {
101 $("#btnSubmit").click(function () {
102 var password = $("#password").val();
103 var confirmPassword = $("#cpassword").val();
104 if (password != confirmPassword) {
105 alert("Passwords do not match.");
106 return false;
107 }
108 return true;
109 });
110 });
111 </script>
112 </body>
113 </html>

```

17.6 User Profile (userProfile.jsp)

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
   pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<link href="resource/css/style.css" rel="stylesheet" />
<head>
<meta charset="UTF-8">
<title>userprofile</title>

<link href="resource/css/style.css" rel="stylesheet" />
<link href="resource/css/UserProfile.css" rel="stylesheet" />
<link href="resource/css/user-profile.css" rel="stylesheet" />
<%@include file=".base.jsp"%>

</head>
<body>
<%@ include file="headafterlogin.jsp"%>
<main>

    <section id="user-profile">
        <div class="container">
            <div class="d-flex mt-5">
                <!-- profile -->
                <main class="me-3 main-profile">
                    <div class="profile">
                        <div class="banner"></div>

                        <div class="p-4 pt-5 position-relative">
                            <div class="profile-picture position-absolute">
                                
                            </div>

                            <div class="d-flex justify-content-between">
                                <h3 class="name"></h3>

                                </div>
                            </div>
                        </div>
                    </main>
                </div>
            </div>
        </div>
    <section class="section about-section gray-bg" id="about">
        <div class="container">
            <div class="row align-items-center flex-row-reverse">
                <div class="col-lg-9">
                    <div class="about-text go-to">

                        <section class="about">
                            <div class="d-flex justify-content-between">
                                <h5>About</h5>

                                </div>
                                <h6 class="theme-color lead">A Lead UX & UI
                                    designer based in Canada</h6>
                                <p>
                                    I
                                    <mark>design and develop</mark>
                                    services for customers of all sizes, specializing in
                                    creating stylish, modern websites, web services and
                                    online stores. My passion is to design digital user
                                    experiences through the bold interface and meaningful
                                    interactions.
                                </p>
                            </div>
                        </section>
                    </div>
                </div>
            </div>
        </div>
    </section>
</main>
</body>
```

```

55         </section>
56     <div class="container emp-profile">
57         <form method="post">
58             <div class="row"></div>
59         </div>
60         <div class="col-md-6">
61             <div class="profile-head">
62                 <h5>${user.name}</h5>
63
64
65                 <ul class="nav nav-tabs" id="myTab" role="tablist">
66
67                     </ul>
68                 </div>
69             </div>
70             <div class="col-md-8" style="background-color: lightgray">
71                 <a class="nav-link" href="edit/${user.id}">Edit
72                     profile</a>
73                     <!-- <input type="submit" class="profile-edit-btn"
74 name="btnAddMore" value="Edit Profile"/> -->
75                 </div>
76             </div>
77
78             <div class="col-md-8">
79                 <div class="tab-content profile-tab" id="myTabContent">
80                     <div class="tab-pane fade show active" id="home"
81                         role="tabpanel" aria-labelledby="home-tab">
82                         <div class="row">
83                             <div class="col-md-6">
84                                 <label>User Id</label>
85                             </div>
86                             <div class="col-md-6">
87                                 <p>${user.id}</p>
88                             </div>
89                         </div>
90                         <div class="row">
91                             <div class="col-md-6">
92                                 <label>Name</label>
93                             </div>
94                             <div class="col-md-6">
95                                 <p>${user.name}</p>
96                             </div>
97                         </div>
98                         <div class="row">
99                             <div class="col-md-6">
100                                 <label>Email</label>
101                             </div>
102                             <div class="col-md-6">
103                                 <p>${user.email}</p>
104                             </div>
105                         </div>
106                         <div class="row">
107                             <div class="col-md-6">
108                                 <label>Phone</label>
109                             </div>
110                             <div class="col-md-6">
111                                 <p>${user.contact}</p>
112                             </div>
113                         </div>
114                         <div class="row">
115                             <div class="col-md-6">
116                                 <label>City</label>
117                             </div>
118                             <div class="col-md-6">
119                                 <p>${user.city}</p>
120                             </div>
121                         </div>
122                         <div class="row">
123                             <div class="col-md-6">
124                                 <label>Profession</label>
125                             </div>
126                             <div class="col-md-6">
127                                 <p>Lead UX & UI Designer</p>
128                             </div>
129                         </div>
130                     </div>
131                 </div>
132             </div>
133
134         </div>
135
136     </div>
137
138
139
140 </body>
141 </html>

```

17.7 Edit Profile (editprofile.jsp)

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <%@page isELIgnored="false"%>
4 <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
5 <%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%>
6 <!DOCTYPE html>
7 <html>
8   <link href="resource/css/style.css" rel="stylesheet" />
9   <head>
10  <meta charset="UTF-8">
11  <title>edit</title>
12  <link href="resources/css/UserProfile.css" rel="stylesheet">
13
14 </head>
15 <body>
16
17  <%@ include file="headafterlogin.jsp"%>
18
19  <div
20    class="d-flex flex-wrap align-items-center justify-content-center mx-auto pt-3
pb-5 mb-5 rounded bg-crm"
21    style="width: 50%">
22    <form method="post" class="col-8" id="edit" action="update">
23      <%-- <c:forEach var="user" items="${listuser}"> --%>
24      <div class="mb-3">
25        <input class="form-control" value="${user.id}" name="id" id="id"
26          hidden="hidden" />
27      </div>
28      <div class="mb-3">
29        <label class="form-label" for="inputname"> Name</label> <input
30          type="text" class="form-control" value="${user.name}" name="name"
31          id="name" />
32      </div>
33      <div class="mb-3">
34        <label class="form-label" for="inputEmail"> Email</label> <input
35          type="text" class="form-control" name="email" id="email"
36          value="${user.email}" />
37      </div>
38      <div class="mb-3">
39        <label class="form-label" for="inputcity"> City</label> <input
40          type="text" class="form-control" name="city" id="city"
41          value="${user.city}" />
42      </div>
43      <div class="mb-3">
44        <label class="form-label" for="inputname"> Phone</label> <input
45          type="text" class="form-control" name="contact" id="contact"
46          value="${user.contact}" />
47      </div>
48
49      <button type="submit" class="btn btn-success" id="update"
50        name="update"> Save </button>
51      <%-- </c:forEach> --%>
52    </form>
53  </div>
54
55 </body>
56 </html>
```

17.8 Search User (searchUser.jsp)

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
4 <%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%>
5 <%@page isELIgnored="false"%>
6 <!DOCTYPE html>
7 <html>
8 <link href="resource/css/style.css" rel="stylesheet" />
9 <head>
10 <meta charset="UTF-8">
11 <title>Insert title here</title>
12 </head>
13 <body>
14
15 <%@ include file="headafterlogin.jsp"%>
16 <div class="container d-flex flex-column">
17   <div class="heading mt-4">
18     <h2 class="fw-semibold">Search People Here</h2>
19   </div>
20   <form:form action="searchUser" method="post" modelAttribute="bean"
21     class="search-user-form d-flex">
22     <form:input path="name" class="src" type="text" id="search-user"
23       placeholder="Search" />
24     <button type="submit" value="Search"
25       class="btn btn-outline-secondary btn-search">
26       <i class="fa-solid fa-magnifying-glass"></i>
27     </button>
28   </form:form>
29   <div class="style">
30     <div class="container">
31       <div class="row">
32         <div class="col-md-8">
33           <div class="people-nearby">
34             <c:forEach var="tab" items="${data}">
35               <div class="nearby-user">
36                 <div class="row">
37                   <div class="col-md-2 col-sm-2">
38                     
40                   </div>
41                   <div class="col-md-7 col-sm-7">
42
43                     <h5>
44                       <input type="text" value="${tab.city}" name="city">
45                     </h5>
46                     <p>
47                       <input type="text" value="${tab.contact}" name="contact">
48                     </p>
49                     <p class="text-muted">
50                       <input type="text" value="${tab.email}" name="email">
51                     </p>
52                   </div>
53                   <div class="col-md-3 col-sm-3">
54                     <a href="viewProfile/${tab.id}"
55                         class="btn btn-success pull-right">View Profile</a>
56                   </div>
57                 </div>
58               </c:forEach>
59             </div>
60           </div>
61         </div>
62       </div>
63     </div>
64   </div>
65 </body>
66 </html>
```

17.9 Check Mail (checkMail.jsp)

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3
4 <!DOCTYPE html>
5 <html>
6 <link href="resource/css/style.css" rel="stylesheet" />
7 <head>
8 <meta charset="UTF-8">
9 <title>Insert title here</title>
10 <link
11   href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
12   rel="stylesheet"
13   integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwpSpd3yD65VohhpooC0mLASjC"
14   crossorigin="anonymous">
15 </head>
16 <body>
17   <section id="forgot-password-confirmation" class="mt-5">
18     <div class="container d-flex justify-content-center">
19       <div class="forgot-password d-flex flex-column text-center">
20         <i class="fa-solid fa-check text-success"></i>
21         <h2 class="fw-semibold">Forgot Password Confirmation</h2>
22         
24         <p class="text-secondary">
25           As per your request,<br> you will receive an email with a
26           temporary password.<br> Use this password to login.<br>
27           Once logged in, you will be able to change your password.
28         </p>
29         <div class="wrapper">
30           <a href="<%=request.getContextPath()%>"'
31             class="btn btn-success mt-3">Back To Home</a>
32         </div>
33       </div>
34     </div>
35   </section>
36 </html>
```

17.10 Admin Profile(adminwelcome.jsp)

```
1  \<%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3
4 <!DOCTYPE html>
5 <html>
6 <link href="resource/css/style.css" rel="stylesheet" />
7 <head>
8 <meta charset="UTF-8">
9 <title>adminwelcome</title>
10 <%@include file=".base.jsp"%>
11 <script src="https://kit.fontawesome.com/848830b72b.js"
12   crossorigin="anonymous"></script>
13 <link rel="stylesheet"
14   href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-
15   beta1/dist/css/bootstrap.min.css" />
16 <%@include file=".base.jsp"%>
17 </head>
18 <%@ include file="headforadmin.jsp"%>
19 <section style="background-color: #eee;">
20   <div class="container py-5">
21     <div class="row">
22       <div class="col">
23         <nav aria-label="breadcrumb" class="bg-light rounded-3 p-3 mb-4">
24           <h1>Hello Admin</h1>
25         </nav>
26       </div>
27     </div>
28
29   <div class="row">
30     <div class="col-lg-4">
31
32     <div class="card mb-4 mb-lg-0">
33       <div class="card-body p-0">
34         <div class="card-body">
35           
37
38           <div class="d-flex justify-content-center mb-2"></div>
39           <p class="mb-4">
40             <span class="text-success font-italic me-1"></span> Projects
41           </p>
42           <p class="mb-1" style="font-size: .77rem;">Server Migration</p>
43           <div class="progress rounded" style="height: 10px;">
44             <div class="progress-bar" role="progressbar"
45               style="width: 80%; background: green;" aria-valuenow="80"
46               aria-valuemin="0" aria-valuemax="100"></div>
47           </div>
48           <p class="mt-4 mb-1" style="font-size: .77rem;">Sales
49             Tracking</p>
50           <div class="progress rounded" style="height: 10px;">
51             <div class="progress-bar" role="progressbar"
52               style="width: 72%; background: pink;" aria-valuenow="72"
53               aria-valuemin="0" aria-valuemax="100"></div>
54           </div>
55           <p class="mt-4 mb-1" style="font-size: .77rem;">Customer
56             Database</p>
57           <div class="progress rounded" style="height: 10px;">
58             <div class="progress-bar" role="progressbar"
59               style="width: 89%; background: red;" aria-valuenow="89"
60               aria-valuemin="0" aria-valuemax="100"></div>
61           </div>
62           <p class="mt-4 mb-1" style="font-size: .77rem;">Payout
63             Details</p>
64           <div class="progress rounded" style="height: 10px;">
65             <div class="progress-bar" role="progressbar" style="width: 55%"
66               aria-valuenow="55" aria-valuemin="0" aria-valuemax="100"></div>
67           </div>
68           <p class="mt-4 mb-1" style="font-size: .77rem;">Account
69             Setup</p>
70           <div class="progress rounded mb-2" style="height: 10px;">
71             <div class="progress-bar" role="progressbar"
72               style="width: 66%; background: purple;" aria-valuenow="66"
73               aria-valuemin="0" aria-valuemax="100"></div>
74           </div>
75         </div>
76       </div>
77     </div>
78   </div>
```

```

76      </div>
77      <div class="col-lg-8">
78          <div class="card mb-4">
79              <div class="card-body">
80                  <div class="row">
81                      <div class="col-sm-3">
82                          <p class="mb-0">Modify</p>
83                      </div>
84                      <div class="col-sm-9">
85                          <a href="searchUser" type="button"
86                              class="btn btn-primary btn-lg btn-block">Search Users</a>
87                      </div>
88                  </div>
89              </div>
90              <hr>
91              <div class="row">
92                  <div class="col-sm-3">
93                      <p class="mb-0">View users</p>
94                  </div>
95                  <div class="col-sm-9">
96                      <a href="viewUser" type="button"
97                          class="btn btn-success btn-lg btn-block">view users</a>
98                  </div>
99              </div>
100             <div class="row">
101                 <div class="col-sm-3">
102                     <p class="mb-0">Email</p>
103                 </div>
104                 <div class="col-sm-9">
105                     <a href="emailForm" type="button"
106                         class="btn btn-warning btn-lg btn-block">Compose email</a>
107                 </div>
108             </div>
109             <div class="row">
110                 <div class="col-sm-3">
111                     <p class="mb-0">Post Jobs</p>
112                 </div>
113                 <div class="col-sm-9">
114                     <a href="postjob" type="button"
115                         class="btn btn-danger btn-lg btn-block">Post Jobs</a>
116                 </div>
117             </div>
118         </div>
119         <div class="row">
120             <!-- Earnings (Monthly) Card Example -->
121             <div class="col-xl-3 col-md-6 mb-4">
122                 <div class="card border-left-primary shadow h-100 py-2">
123                     <div class="card-body">
124                         <div class="row no-gutters align-items-center">
125                             <div class="col mr-2">
126                                 <div
127                                     class="text-xs font-weight-bold text-primary text-uppercase
128                                     mb-1">
129                                     Earnings (Monthly)</div>
130                                     <div class="h5 mb-0 font-weight-bold text-gray-
131                                     800">$40,000</div>
132                                 </div>
133                             <div class="col-auto">
134                                 <i class="fas fa-calendar fa-2x text-gray-300"></i>
135                             </div>
136                         </div>
137                     </div>
138                 </div>
139             </div>
140         </div>
141     </div>
142 </div>

```

```

144      <!-- Earnings (Monthly) Card Example -->
145      <div class="col-xl-3 col-md-6 mb-4">
146          <div class="card border-left-success shadow h-100 py-2">
147              <div class="card-body">
148                  <div class="row no-gutters align-items-center">
149                      <div class="col mr-2">
150                          <div
151                              class="text-xs font-weight-bold text-success text-uppercase
mb-1">
152                              Earnings (Annual)</div>
153                              <div class="h5 mb-0 font-weight-bold text-gray-
800">$215,000</div>
154                          </div>
155                          <div class="col-auto">
156                              <i class="fas fa-dollar-sign fa-2x text-gray-300"></i>
157                          </div>
158                      </div>
159                  </div>
160              </div>
161          </div>
162
163      <!-- Earnings (Monthly) Card Example -->
164      <div class="col-xl-3 col-md-6 mb-4">
165          <div class="card border-left-info shadow h-100 py-2">
166              <div class="card-body">
167                  <div class="row no-gutters align-items-center">
168                      <div class="col mr-2">
169                          <div
170                              class="text-xs font-weight-bold text-info text-uppercase mb-
1">Tasks
171                          </div>
172                          <div class="row no-gutters align-items-center">
173                              <div class="col-auto">
174                                  <div class="h5 mb-0 mr-3 font-weight-bold text-gray-
800">50%</div>
175                              </div>
176                              <div class="col">
177                                  <div class="progress progress-sm mr-2">
178                                      <div class="progress-bar bg-info" role="progressbar"
179                                          style="width: 50%" aria-valuenow="50" aria-valuemin="0"
180                                          aria-valuemax="100"></div>
181                                  </div>
182                              </div>
183                          </div>
184                      </div>
185                      <div class="col-auto">
186                          <i class="fas fa-clipboard-list fa-2x text-gray-300"></i>
187                      </div>
188                  </div>
189              </div>
190          </div>
191      </div>
192
193      <!-- Pending Requests Card Example -->
194      <div class="col-xl-3 col-md-6 mb-4">
195          <div class="card border-left-warning shadow h-100 py-2">
196              <div class="card-body">
197                  <div class="row no-gutters align-items-center">
198                      <div class="col mr-2">
199                          <div
200                              class="text-xs font-weight-bold text-warning text-uppercase
mb-1">
201                              Pending Requests</div>
202                              <div class="h5 mb-0 font-weight-bold text-gray-800">18</div>
203                          </div>
204                          <div class="col-auto">
205                              <i class="fas fa-comments fa-2x text-gray-300"></i>
206                          </div>
207                      </div>
208                  </div>
209              </div>
210          </div>
211      </div>
212      </div>
213  </div>
214 </div>
215
216 </section>
217
218 <%@ include file="footer.jsp"%>
219 </body>

```

17.11 Index Page (home.jsp)

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html lang="en">
5 <link href="resource/css/style.css" rel="stylesheet" />
6 <head>
7 <meta charset="utf-8" />
8
9 <title>Home</title>
10 <%@include file=".base.jsp"%>
11 <link href="resources/css/Home.css" rel="stylesheet">
12
13 </head>
14 <body id="page-top">
15   <!-- Navigation-->
16   <%@ include file="headafterlogin.jsp"%>
17
18   <link
19     href="https://maxcdn.bootstrapcdn.com/font-awesome/4.3.0/css/font-
awesome.min.css"
20     rel="stylesheet">
21   <section id="content" class="container">
22
23
24   <div class="row">
25     <div class="col-md-4">
26       <div class="panel">
27         <div class="panel-heading">
28           <span class="panel-icon"> <i class="fa fa-star"></i>
29           </span> <span class="panel-title"> Manage My Work</span>
30         </div>
31         <div class="panel-body pn">
32           <table class="table mbn tc-icon-1 tc-med-2 tc-bold-last">
33             <thead>
34               <tr class="hidden">
35                 <th class="mw30">#</th>
36                 <th>All</th>
37                 <th>Revenue</th>
38               </tr>
39             </thead>
40             <tbody>
41               <tr>
42                 <td><span class="fa-solid fa-users text-warning"></span></td>
43                 <td>Connections</td>
44                 <td>500+</td>
45               </tr>
46               <tr>
47                 <td><span class="fa-solid fa-calendar-days text-success"></span></td>
48                 <td>Events</td>
49                 <td>250+</td>
50               </tr>
51               <tr>
52                 <td><span class="fa fa-newspaper-o text-info"></span></td>
53                 <td>Pages</td>
54                 <td>1765</td>
55               </tr>
56             </tbody>
57           </table>
58         </div>
59       </div>
60     </div>
```

```

61      <div class="panel">
62        <div class="panel-heading">
63          <span class="panel-icon"> <i class="fa fa-trophy"></i>
64          </span> <span class="panel-title"> My Skills</span>
65        </div>
66        <div class="panel-body pb5">
67          <span class="label label-warning mr5 mb10 ib lh15">Default</span>
68          <span class="label label-primary mr5 mb10 ib lh15">Primary</span>
69          <span class="label label-info mr5 mb10 ib lh15">Success</span> <span
70            class="label label-success mr5 mb10 ib lh15">Info</span> <span
71            class="label label-alert mr5 mb10 ib lh15">Warning</span> <span
72            class="label label-system mr5 mb10 ib lh15">Danger</span> <span
73            class="label label-info mr5 mb10 ib lh15">Success</span> <span
74            class="label label-success mr5 mb10 ib lh15">Ui Design</span> <span
75            class="label label-primary mr5 mb10 ib lh15">Primary</span>
76
77      </div>
78    </div>
79    <div class="panel">
80      <div class="panel-heading">
81        <span class="panel-icon"> <i class="fa-solid fa-briefcase"></i>
82        </span> <span class="panel-title">Jobs</span>
83      </div>
84      <div class="panel-body pb5">
85
86
87        <h4>Senior Software Engineer (Python)</h4>
88        <p class="text-muted">
89          Virtusa <br> Sri Lanka (Remote) <br> Full Time <br>
90          <button type="button" class="btn btn-success">More</button>
91        </p>
92
93        <hr class="short br-lighter">
94
95
96
97
98        <h4>Full Stack Developer</h4>
99        <p class="text-muted">
100          Soul Studio <br> Maldives <br>
101          <button type="button" class="btn btn-success">More</button>
102        </p>
103
104        <hr class="short br-lighter">
105
106        <h4>Resort Housekeeper</h4>
107        <p class="text-muted pb10">
108          Sun Communities & Sun Doors <br> 45/C St.Francis Road,
109          Colombo 7 <br>
110          <button type="button" class="btn btn-success">More</button>
111        </p>
112
113      </div>
114    </div>
115  </div>

```

```

116 <div class="col-md-8">
117
118     <div class="tab-block">
119         <ul class="nav nav-tabs">
120             <li class="active"><a href="#tab1" data-toggle="tab">Activity</a>
121             </li>
122             <li><a href="#tab1" data-toggle="tab">Social</a></li>
123             <li><a href="#tab1" data-toggle="tab">Media</a></li>
124         </ul>
125         <div class="tab-content p30" style="height: 730px;">
126             <div id="tab1" class="tab-pane active">
127                 <div class="media">
128                     <a class="pull-left" href="#"> 
131                 </a>
132                 <div class="media-body">
133                     <h5 class="media-heading mb20">
134                         Leo Ashley Posted <small> - 3 hours ago</small>
135                     </h5>
136                     
137                     
138                     
139                     <div class="media-links">
140                         <span class="text-light fs12 mr10"> <span
141                             class="fa fa-thumbs-o-up text-primary mr5"></span> Like
142                         </span> <span class="text-light fs12 mr10"> <span
143                             class="fa fa-share text-success mr5"></span> Share
144                         </span> <span class="text-light fs12 mr10"> <span
145                             class="fa fa-floppy-o text-dark mr5"></span> Save
146                         </span> <span class="text-light fs12 mr10"> <span
147                             class="fa fa-comment text-danger mr5"></span> Comment
148                         </span>
149                     </div>
150                 </div>
151             </div>
152             <div class="media mt25">
153                 <a class="pull-left" href="#"> 
156                 </a>
157                 <div class="media-body mb5">
158                     <h5 class="media-heading mbn">
159                         Leo Ashley Posted <small> - 3 hours ago</small>
160                     </h5>
161                     <p>Finished Project Management, Spring MVC, HTML
162                         Foundations</p>
163                     <div class="media pb10">
164                         <a class="pull-left" href="#"> 
167                         </a>
168                         <div class="media-body mb5">
169                             <h5 class="media-heading mbn">
170                                 Jessica Wong <small> - 3 hours ago</small>
171                             </h5>
172                             <p>Congratulations</p>
173                         </div>
174                     </div>

```

```

175         <div class="media mtn">
176             <a class="pull-left" href="#"> 
179             </a>
180             <div class="media-body mb5">
181                 <h5 class="media-heading mbn">
182                     Kevin Wix <small> - 3 hours ago</small>
183                 </h5>
184                 <p>Congrats..Keep Going</p>
185             </div>
186         </div>
187     </div>
188 </div>
189 <div class="media mt25">
190     <a class="pull-left" href="#"> 
193     </a>
194     <div class="media-body">
195         <h5 class="media-heading mb20">
196             Jessica Wong Posted <small> - 5 hours ago</small>
197         </h5>
198         
199         
200         
201         <div class="media-links">
202             <span class="text-light fs12 mr10"> <span
203                 class="fa fa-thumbs-o-up text-primary mr5"></span> Like
204             </span> <span class="text-light fs12 mr10"> <span
205                 class="fa fa-share text-success mr5"></span> Share
206             </span> <span class="text-light fs12 mr10"> <span
207                 class="fa fa-floppy-o text-dark mr5"></span> Save
208             </span> <span class="text-light fs12 mr10"> <span
209                 class="fa fa-comment text-danger mr5"></span> Comment
210             </span>
211         </div>
212     </div>
213 </div>
214 <div id="tab2" class="tab-pane"></div>
215 <div id="tab3" class="tab-pane"></div>
216 <div id="tab4" class="tab-pane"></div>
217 </div>
218 </div>
219 </div>
220 </div>
221 </div>
222 </section>
223
224 <!-- Footer-->
225 <footer class="bg-light"> </footer>
226 <!-- Bootstrap core JS-->
227 <script
228     src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js">
229 </script>
230 <!-- SimpleLightbox plugin JS-->
231 <script
232     src="https://cdn.jsdelivr.net/simpleLightbox/2.1.0/simpleLightbox.min.
233 s"></script>
234 <script src="https://cdn.startbootstrap.com/sb-forms-latest.js"></script>
235 </body>

```

Task 6

18. Java Classes

18.1 com.spring.mvc.config

- **ViewRes**

```
1 package com.spring.mvc.config;
2
3 import org.springframework.context.annotation.Bean;
4 import org.springframework.context.annotation.ComponentScan;
5 import org.springframework.context.annotation.Configuration;
6 import org.springframework.web.servlet.config.annotation.EnableWebMvc;
7 import org.springframework.web.servlet.config.annotation.ResourceHandlerRegistry;
8 import org.springframework.web.servlet.config.annotation.WebMvcConfigurer;
9 import org.springframework.web.servlet.view.InternalResourceViewResolver;
10
11 @EnableWebMvc
12 @Configuration
13 @ComponentScan("abcjobportal")
14 public class ViewRes implements WebMvcConfigurer {
15
16     @Bean(name = "viewResolver")
17     public InternalResourceViewResolver getViewResolver() {
18         InternalResourceViewResolver viewResolver = new InternalResourceViewResolver();
19         viewResolver.setPrefix("/WEB-INF/views/");
20         viewResolver.setSuffix(".jsp");
21         return viewResolver;
22     }
23 }
24
25 @Override
26 public void addResourceHandlers(ResourceHandlerRegistry registry)
27 {
28     registry.addResourceHandler("/resource/**")
29         .addResourceLocations("/resource/css")
30         .addResourceLocations("/resource/images")
31         .addResourceLocations("/resource/js")
32     ;
33 }
34 }
35 }
36 }
```

18.2 com.spring.mvc.controller

✓ ApplicationController.java

```
1 package com.spring.mvc.controller;
2
3 import java.util.List;
4
5 import org.springframework.beans.factory.annotation.Autowired;
6 import org.springframework.context.ApplicationContext;
7 import org.springframework.context.support.ClassPathXmlApplicationContext;
8 import org.springframework.stereotype.Controller;
9 import org.springframework.ui.Model;
10 import org.springframework.web.bind.annotation.ModelAttribute;
11 import org.springframework.web.bind.annotation.PathVariable;
12 import org.springframework.web.bind.annotation.RequestMapping;
13 import org.springframework.web.bind.annotation.RequestMethod;
14 import org.springframework.web.bind.annotation.RequestParam;
15
16 import com.spring.mvc.dao.ApplicationDAO;
17
18 import com.spring.mvc.model.User;
19
20 @Controller
21 public class ApplicationController {
22
23     private ApplicationContext context;
24     @Autowired
25     private ApplicationDAO applicationDAO;
26
27     @RequestMapping(value = "/resetPass", method = RequestMethod.POST)
28     public String edit(@ModelAttribute("bean") User user, Model model) {
29         context = new ClassPathXmlApplicationContext("ApplicationContext.xml");
30         ApplicationDAO obj = context.getBean("dao", ApplicationDAO.class);
31         List list = obj.resetPassword(user);
32         model.addAttribute("data", list);
33         model.addAttribute("reset", " Reset Password Successful");
34         return "login";
35     }
36
37     @RequestMapping("/edit/{userId}")
38     public String updaeUserForm(@PathVariable("userId") Long uId, Model model) {
39         User user = this.applicationDAO.getUser(uId);
40         model.addAttribute("user", user);
41         return "editprofile";
42     }
43
44     // for update profile
45     @RequestMapping(value = "edit/update", method = RequestMethod.POST)
46     public String updateUser(@RequestParam("id") Long id, @RequestParam("name") String
47 name,
48         @RequestParam("email") String email, @RequestParam("contact") String contact,
49         @RequestParam("city") String city) {
50         User user = new User();
51         user.setId(id);
52         user.setName(name);
53         user.setEmail(email);
54         user.setCity(city);
55         user.setContact(contact);
56
57         applicationDAO.updateprofile(user);
58         return "editprofile";// will come edit profile page
59     }
60 }
```

```

62  @RequestMapping("/find")
63  public String search(Model model) {
64      context = new ClassPathXmlApplicationContext("ApplicationContext.xml");
65      User user = context.getBean("argon", User.class);
66      model.addAttribute("bean", user);
67      return "search";
68  }
69
70  @RequestMapping(value = "/searchview", method = RequestMethod.GET)
71  public String searchView(@ModelAttribute("bean") User user, Model model) {
72      return "search";
73  }
74
75  @RequestMapping(value = "/searchUser", method = RequestMethod.POST)
76  public String searchUsrs(@ModelAttribute("bean") User user, Model model) {
77      context = new ClassPathXmlApplicationContext("ApplicationContext.xml");
78      ApplicationDAO obj = context.getBean("dao", ApplicationDAO.class);
79
80      List list = obj.search(user);
81      if (!list.isEmpty()) {
82          model.addAttribute("data", list);
83      } else {
84
85          model.addAttribute("msg", "User Not found");
86      }
87      return "searchUser";
88  }
89
90  @RequestMapping("searchUser")
91  public String searchUser(Model model, @ModelAttribute("bean") User user) {
92      context = new ClassPathXmlApplicationContext("ApplicationContext.xml");
93
94      ApplicationDAO obj = context.getBean("dao", ApplicationDAO.class);
95      List<User> list = obj.display();
96
97      model.addAttribute("data", list);
98
99      return "searchUser";
100 }
101 @RequestMapping("viewUser")
102 public String viewUser(Model model, @ModelAttribute("bean") User user) {
103     context = new ClassPathXmlApplicationContext("ApplicationContext.xml");
104
105     ApplicationDAO obj = context.getBean("dao", ApplicationDAO.class);
106     List<User> list = obj.display();
107
108     model.addAttribute("data", list);
109
110     return "viewUser";
111 }
112
113 @RequestMapping("/viewProfile/{userId}")
114 public String userProfilesearch(@PathVariable("userId") Long userId, Model model) {
115
116     User user = this.applicationDAO.getUser(userId);
117     model.addAttribute("user", user);
118     return "viewProfile";
119
120 }
121
122 }
```

```

124 @RequestMapping("admindash")
125 public String admindasdisplay(Model model, @ModelAttribute("bean") User user) {
126     context = new ClassPathXmlApplicationContext("ApplicationContext.xml");
127
128     ApplicationDAO obj = context.getBean("dao", ApplicationDAO.class);
129     List<User> list = obj.display();
130
131     model.addAttribute("data", list);
132
133     return "admindash";
134 }
135
136
137 @RequestMapping("/findadmin")
138 public String adminsearch(Model model) {
139     context = new ClassPathXmlApplicationContext("ApplicationContext.xml");
140     User user = context.getBean("argon", User.class);
141     model.addAttribute("bean", user);
142     return "admindash";
143 }
144
145 @RequestMapping(value = "/adminsearchview", method = RequestMethod.GET)
146 public String adminsearchView(@ModelAttribute("bean") User user, Model model) {
147     return "admindash";
148 }
149 @RequestMapping(value = "/adminsearchUsrs", method = RequestMethod.POST)
150 public String adminsearchUsrs(@ModelAttribute("bean") User user, Model model) {
151     context = new ClassPathXmlApplicationContext("ApplicationContext.xml");
152     ApplicationDAO obj = context.getBean("dao", ApplicationDAO.class);
153
154     List list = obj.search(user);
155     if (!list.isEmpty()) {
156         model.addAttribute("data", list);
157     } else {
158
159         model.addAttribute("msg", "User Not found");
160     }
161     return "admindash";
162 }
163
164 @RequestMapping(value = "/editor", method = RequestMethod.POST)
165 public String adminedit(@ModelAttribute("bean") User user, Model model) {
166     context = new ClassPathXmlApplicationContext("ApplicationContext.xml");
167     ApplicationDAO obj = context.getBean("dao", ApplicationDAO.class);
168     List list = obj.update(user);
169     model.addAttribute("data", list);
170     model.addAttribute("msg", " Record update successfully");
171     return "admindash";
172 }
173
174 @RequestMapping(value = "/delete", method = RequestMethod.GET)
175 public String admindedetele(@ModelAttribute("bean") User user, Model model) {
176     context = new ClassPathXmlApplicationContext("ApplicationContext.xml");
177     ApplicationDAO obj = context.getBean("dao", ApplicationDAO.class);
178     List list = obj.delete(user);
179     model.addAttribute("data", list);
180     model.addAttribute("msg", " Record deleted successfully");
181     return "viewUser";
182 }
```

✓ MainController.java

```
1 package com.spring.mvc.controller;
2
3 import org.springframework.stereotype.Controller;
4 import org.springframework.web.bind.annotation.RequestMapping;
5
6 @Controller
7 public class MainController {
8
9     @RequestMapping("/registration")
10    public String registration() {
11        return "registration";
12    }
13
14    @RequestMapping("/thankyou")
15    public String thankyou() {
16        return "thankyou";
17    }
18
19
20    @RequestMapping("/login")
21    public String login() {
22        return "login";
23    }
24
25    @RequestMapping("/jobs")
26    public String jobs() {
27        return "jobs";
28    }
29
30
31    @RequestMapping("/adminwelcome")
32    public String adminwelcome() {
33        return "adminwelcome";
34    }
35
36    @RequestMapping("/postjob")
37    public String postjob() {
38        return "postjob";
39    }
40
41    @RequestMapping("/forgotpassform")
42    public String forgotpassform() {
43        return "forgotpassform";
44    }
45
46    @RequestMapping("/forgot")
47    public String forgot() {
48        return "forgot";
49    }
50
51    @RequestMapping("/home")
52    public String home() {
53        return "home";
54    }
55
56    @RequestMapping("/userProfile")
57    public String userProfile() {
58        return "userProfile";
59    }
60
61    @RequestMapping("/editprofile")
62    public String editprofile() {
63        return "editprofile";
64    }
65
66    @RequestMapping("/jobDetails")
67    public String jobDetails() {
68        return "jobDetails";
69    }
70
71    @RequestMapping("/viewProfile")
72    public String viewProfile() {
73        return "viewProfile";
74    }
75
76
77    @RequestMapping("/emailForm")
78    public String emailForm() {
79        return "emailForm";
80    }
81
82    @RequestMapping("error")
83    public String error() {
84        return "error";
85    }
86
87    @RequestMapping("success")
88    public String success() {
89        return "success";
90    }
91
92}
```

✓ UserController.java

```
1 package com.spring.mvc.controller;
2
3 import java.util.Map;
4
5 import javax.servlet.http.HttpServletRequest;
6 import javax.servlet.http.HttpSession;
7
8 import org.springframework.beans.factory.annotation.Autowired;
9 import org.springframework.stereotype.Controller;
10 import org.springframework.ui.Model;
11 import org.springframework.web.bind.annotation.GetMapping;
12 import org.springframework.web.bind.annotation.ModelAttribute;
13 import org.springframework.web.bind.annotation.PathVariable;
14 import org.springframework.web.bind.annotation.PostMapping;
15 import org.springframework.web.bind.annotation.RequestMapping;
16 import org.springframework.web.bind.annotation.RequestMethod;
17 import org.springframework.web.bind.annotation.RequestParam;
18 import org.springframework.web.servlet.ModelAndView;
19
20 import com.spring.mvc.model.Admin;
21 import com.spring.mvc.model.User;
22 import com.spring.mvc.servic.AdminService;
23 import com.spring.mvc.servic.ForgotPassEmailService;
24 import com.spring.mvc.servic.MailService;
25 import com.spring.mvc.servic.UserService;
26
27 @Controller
28 public class UserController {
29
30     @Autowired//To call the userservice
31     private UserService userService;
32     @Autowired//To call the mail servic
33
34     private MailService mailService;
35     @Autowired
36     private AdminService adminService;
37     @Autowired
38     private ForgotPassEmailService forgotPassEmailService;
39
40
41     //to get value from registration form
42     @GetMapping(value="/registration")
43     public ModelAndView registration(Model model) {
44         User user=new User();
45         model.addAttribute("user",user);
46         System.out.println(user);
47         ModelAndView modelAndView=new ModelAndView("registration");
48
49         return modelAndView;
50
51     }
```

```

53  @PostMapping("/registration")
54  public String register(@ModelAttribute("user") User user, Model model,
55   HttpServletRequest request,
56   String emailToRecipient, String userName, String EmailID) {
57   emailToRecipient = request.getParameter("email");
58   userName = request.getParameter("name");
59
60   userService.registerUser(user);
61   mailService.sendMail(emailToRecipient, userName); // sending mail to the user
62   //this registration is completed
63   request.getSession().setAttribute("name", userName);
64   request.getSession().setAttribute("email", emailToRecipient);
65
66   model.addAttribute("success", "Registration is successful");
67
68   return "thankyou";
69 }
70 @RequestMapping(value = "/login", method = RequestMethod.POST)
71 public String login(@ModelAttribute("user") User user, Model model, HttpSession
72 session,
73   @ModelAttribute("admin") Admin admin) {
74   User user2 = userService.loginUser(user); // For user
75   Admin user3 = adminService.adminlogin(admin); // For admin
76   System.out.println("user2" + user2);
77   System.out.println("user3" + user3);
78
79   if (user2 != null) { // If user not null it will come welcome page for him
80     System.out.println("hello");
81     model.addAttribute("user", user2);
82     session.setAttribute("user", user2);
83     return "home";
84   }
85
86   if (user3 != null) { // If admin not null it will come adminwelcome page for him
87     System.out.println("hello");
88     model.addAttribute("admin", user3);
89     session.setAttribute("admin", user3);
90     return "adminwelcome";
91   }
92
93 //If user could not get it will show error
94   if (user2 == null) {
95     System.out.println("on");
96     model.addAttribute("error", "Invalid Credential");
97
98 //If user information can not get it will show error
99   if (user3 == null) {
100     System.out.println("on");
101     model.addAttribute("error", "Invalid Credential");
102   }
103
104   return "login";
105 }
106

```

```

108 //logout
109 public String loginDisplay(Model m, HttpSession session) {
110     User user=new User();
111     if(session.getAttribute("user") !=null){
112         session.invalidate();
113         System.out.println("here");
114         m.addAttribute("success","You have logout successfully");
115     }
116     m.addAttribute("user",user);
117     return "login";
118 }
120
121
122
123
124 //For checking email is it in database or not and send mail to the definite email
125 //start
126 @RequestMapping(value = "/emailConfirm", method = RequestMethod.POST)
127 public String checkMail(@ModelAttribute("user") User user, Model model,
HttpSession session,
128     @RequestParam(value = "email") String email, HttpServletRequest request) {
129     User user4 = userService.checkMail(user);
130     System.out.println(user4);
131     if (user4 != null) {
132         System.out.println("hello");
133         model.addAttribute("user", user4);
134         session.setAttribute("user", user4);
135         forgotPassEmailService.resetMail(email); // sending mail to the user to reset
his/her password
136         request.getSession().setAttribute("email", email);
137
138     //return "checkMail";
139     return "checkMail";
140
141 }
142 if (user4 == null) {
143     System.out.println("on");
144     model.addAttribute("error", "Invalid Credential");
145 }
146
147 return "forgotpassform";
148
149
150
151
152 }
153
154 @RequestMapping(value="/resetPassword/{email}")
155 public String resetPassword(@PathVariable String email,Map<String, String> model)
156 {
157     //check if the email id is valid and registered with us.
158     model.put("emailid", email);
159     return "reset_forgotpassform";
160 }
161
162
163
164 }

```

18.3 com.spring.mvc.dao

✓ Admindao.java

```
1 package com.spring.mvc.dao;
2
3 import com.spring.mvc.model.Admin;
4
5 public interface AdminDAO {
6
7     public Admin adminlogin(Admin admin);
8 }
```

✓ AdminDaoImpl.java

```
1 package com.spring.mvc.dao;
2
3 import javax.persistence.NoResultException;
4
5 @Repository
6 @Transactional
7 public class AdminDAOImpl implements AdminDAO{
8     @Autowired
9     private SessionFactory factory;
10
11     public Admin adminlogin(Admin admin) {
12
13         Session session=factory.getCurrentSession();
14         try {
15             Query<Admin> query=session.createQuery("FROM admin_table where email=:email and password=:password",Admin
16
17             query.setParameter("email",admin.getEmail());
18             query.setParameter("password",admin.getPassword());
19             admin=(Admin) query.getSingleResult();
20             query.setMaxResults(1).uniqueResult();
21
22             return admin;
23         }catch(NoResultException e) {
24             return null;
25         }
26
27     }
28
29 }
```

✓ ApplicationDao.java

```
1 package com.spring.mvc.dao;
2
3 import java.sql.PreparedStatement;
4 import java.sql.ResultSet;
5 import java.sql.SQLException;
6 import java.util.ArrayList;
7 import java.util.List;
8
9 import javax.transaction.Transactional;
10
11 import org.springframework.beans.factory.annotation.Autowired;
12 import org.springframework.dao.DataAccessException;
13 import org.springframework.jdbc.core.JdbcTemplate;
14 import org.springframework.jdbc.core.PreparedStatementCallback;
15 import org.springframework.jdbc.core.ResultSetExtractor;
16 import org.springframework.orm.hibernate5.HibernateTemplate;
17 import org.springframework.stereotype.Repository;
18
19 import com.spring.mvc.model.Job;
20 import com.spring.mvc.model.User;
21
22 @Repository
23 public class ApplicationDAO {
24     @Autowired
25     private JdbcTemplate jdbcTemplate;
26
27     @Autowired
28     private HibernateTemplate hibernateTemplate;
29
30     //getting the data from jdbc template
31     public JdbcTemplate getJdbcTemplate() {
32         return jdbcTemplate;
33     }
34
35     //setting the data from jdbc
36     public void setJdbcTemplate(JdbcTemplate jdbcTemplate) {
37         this.jdbcTemplate = jdbcTemplate;
38     }
39
40     public List search(User user) {
41         final List list = new ArrayList();
42         String sql = "select * from user_table where name = '" + user.getName() + "' or
43         email ='" + user.getEmail()
43         + "' or contact=''" + user.getContact() + "' or password=''" + user.getPassword() +
44         "' or id=''" + user.getId() + "'";
45         jdbcTemplate.query(sql, new ResultSetExtractor<User>() {
46
47             public User extractData(ResultSet rs) throws SQLException, DataAccessException
48             {
```

```

47     while (rs.next()) {
48         User user = new User();
49         user.setId(rs.getLong(1));
50         user.setName(rs.getString(2));
51         user.setEmail(rs.getString(3));
52         user.setContact(rs.getString(4));
53         user.setCity(rs.getString(5));
54         user.setPassword(rs.getString(6));
55
56         list.add(user);
57     }
58     return null;
59 }
60 });
61 return list;
62 }
63
64 public List display() {
65     final List list = new ArrayList();
66     String sql = "select * from user_table";
67     jdbcTemplate.query(sql, new ResultSetExtractor<User>() {
68         public User extractData(ResultSet rs) throws SQLException, DataAccessException
69     {
70         while (rs.next()) {
71             User user = new User();
72             user.setId(rs.getLong(1));
73             user.setName(rs.getString(2));
74             user.setEmail(rs.getString(3));
75             user.setContact(rs.getString(4));
76             user.setCity(rs.getString(5));
77             user.setPassword(rs.getString(6));
78
79             list.add(user);
80         }
81     }
82     return null;
83 }
84 });
85 return list;
86 }
87
88 public List resetPassword(final User user) {
89     String sql = "update user_table set password=? where email=?";
90     jdbcTemplate.execute(sql, new PreparedStatementCallback<Boolean>() {
91
92         public Boolean doInPreparedStatement(PreparedStatement ps) throws SQLException,
93             DataAccessException {
94             ps.setString(1, user.getPassword());
95             ps.setString(2, user.getEmail());
96
97             return ps.execute();
98         }
99
100 });
101
102     List list = display();
103     return list;
104 }
105
106 public List updateprofile(final User user) {
107     String sql = "update user_table set name=?,email=? ,contact=? where id=?";
108     jdbcTemplate.execute(sql, new PreparedStatementCallback<Boolean>() {
109
110         public Boolean doInPreparedStatement(PreparedStatement ps) throws SQLException,
111             DataAccessException {
112             ps.setLong(1, 1);
113             ps.setString(2, user.getName());
114             ps.setString(3, user.getContact());
115             ps.setString(4, user.getEmail());
116
117             return ps.execute();
118         }
119     }
120 }

```

```

117    });
118    List list = display();
119    return list;
120 }
121
122 public List<User> findUserByID(Long id) {
123     final List list = new ArrayList();
124     String sql = "select * from user_table where id=?";
125     jdbcTemplate.query(sql, new ResultSetExtractor<User>() {
126
127         public User extractData(ResultSet rs) throws SQLException, DataAccessException
128     {
129         while (rs.next()) {
130             User user = new User();
131             user.setId(rs.getLong(1));
132             user.setName(rs.getString(2));
133             user.setEmail(rs.getString(3));
134             user.setContact(rs.getString(4));
135             user.setCity(rs.getString(5));
136             user.setPassword(rs.getString(6));
137
138             list.add(user);
139         }
140         return null;
141     }
142
143 });
144 return list;
145 }
146
147 @Transactional
148 public User getUser(Long userId) {
149     return this.hibernateTemplate.get(User.class, userId);
150 }
151
152
153 @Transactional
154 public List<Job> getjobDetails() {
155     List<Job> jobs = this.hibernateTemplate.loadAll(Job.class);
156
157     return jobs;
158 }
159
160
161 public List update(final User user) {
162     String sql = "update user_table set name=?,email=?,contact=?,city=? where id=?";
163
164     jdbcTemplate.execute(sql, new PreparedStatementCallback<Boolean>() {
165
166         public Boolean doInPreparedStatement(PreparedStatement ps) throws SQLException,
167             DataAccessException {
168             ps.setString(4, user.getCity());
169
170             ps.setString(3, user.getContact());
171             ps.setString(2, user.getEmail());
172             ps.setString(1, user.getName());
173
174             ps.setLong(5, user.getId());
175             return ps.execute();
176         }
177
178     });
179
180     // TODO Auto-generated method stub
181     List list = display();
182     return list;
183 }
184
185 public List delete(User user) {
186     String str = "delete from user_table where id='" + user.getId() + "'";
187     jdbcTemplate.update(str);
188     List list = display();
189     return list;
190 }

```

✓ **UserDao.java**

```
1 package com.spring.mvc.dao;
2
3 import com.spring.mvc.model.User;
4
5 public interface UserDAO {
6     public void registerUser(User user);
7     public User loginUser(User user);
8     public User checkMail(User user);
9 }
10
```

✓ **UserDaoImplementation.java**

```
1 package com.spring.mvc.dao;
2
3
4 import javax.persistence.NoResultException;
5
6 import org.hibernate.Session;
7 import org.hibernate.SessionFactory;
8 import org.hibernate.query.Query;
9 import org.springframework.beans.factory.annotation.Autowired;
10 import org.springframework.stereotype.Repository;
11 import org.springframework.transaction.annotation.Transactional;
12 import com.spring.mvc.model.User;
13 @Repository
14 @Transactional
15 public class UserDAOImpl implements UserDAO {
16 //To call the session factory
17     @Autowired
18     private SessionFactory factory;
19
20     public void registerUser(User user) {
21         Session session=factory.getCurrentSession();
22         session.save(user);
23     }
24 }
```

```

26 public User loginUser(User user) {
27     // TODO Auto-generated method stub
28     Session session = factory.getCurrentSession();
29     try {
30
31         Query<User> query = session.createQuery("from User_table where email=:email and
32             password=:password",
33             User.class);
34
35         query.setParameter("email", user.getEmail());
36         query.setParameter("password", user.getPassword());
37         user = query.getSingleResult();
38         //query.setMaxResults(1).uniqueResult();
39
40         return user;
41     } catch (NoResultException e) {
42         return null;
43     }
44
45 }
46
47 public User checkMail(User user) {
48     // TODO Auto-generated method stub
49
50     // TODO Auto-generated method stub
51     Session session = factory.getCurrentSession();
52     try {
53
54         Query<User> query = session.createQuery("from User_table where email=:email",
55             User.class);
56
57         query.setParameter("email", user.getEmail());
58
59         //user=query.getSingleResult();
60         query.setMaxResults(1).uniqueResult();
61
62         return user;
63     } catch (NoResultException e) {
64         return null;
65     }
66
67
68
69 }
70
71 //to login user

```

18.4 com.spring.mvc.model

✓ admin.java

```
1 package com.spring.mvc.model;
2
3 import javax.persistence.GeneratedValue;
4 import javax.persistence.GenerationType;
5 import javax.persistence.Id;
6 import javax.validation.constraints.NotNull;
7
8 @javax.persistence.Entity(name = "admin_table")
9 @javax.persistence.Table(name = "admin_table")
10 public class Admin {
11     @Id
12     @GeneratedValue(strategy = GenerationType.IDENTITY)
13     private Long id;
14     @NotNull
15     private String name;
16     @NotNull
17     private String email;
18     @NotNull
19     private String password;
20     public Long getId() {
21         return id;
22     }
23     public void setId(Long id) {
24         this.id = id;
25     }
26     public String getName() {
27         return name;
28     }
29     public void setName(String name) {
30         this.name = name;
31     }
32     public String getEmail() {
33         return email;
34     }
35     public void setEmail(String email) {
36         this.email = email;
37     }
38     public String getPassword() {
39         return password;
40     }
41     public void setPassword(String password) {
42         this.password = password;
43     }
44     public Admin(Long id, String name, String email, String password) {
45         super();
46         this.id = id;
47         this.name = name;
48         this.email = email;
49         this.password = password;
50     }
51     @Override
52     public String toString() {
53         return "Admin [id=" + id + ", name=" + name + ", email=" + email + ", password=" +
54             password + "]";
55     }
56     public Admin() {
57         super();
58         // TODO Auto-generated constructor stub
59     }
60 }
```

✓ User.java

```
1 package com.spring.mvc.model;
2
3 import javax.persistence.Column;
4 import javax.persistence.GeneratedValue;
5 import javax.persistence.GenerationType;
6 import javax.persistence.Id;
7
8 @javax.persistence.Entity(name = "User_table")
9 @javax.persistence.Table(name = "User_table")
10 public class User {
11
12     @Id
13     @GeneratedValue(strategy = GenerationType.IDENTITY)
14
15     @Column(name="id")
16     private Long id;
17     @Column(name="name")
18     private String name;
19     @Column(name="city")
20     private String city;
21     @Column(name="password")
22     private String password;
23     @Column(name="contact")
24     private String contact;
25     @Column(name="email")
26     private String email;
27     public Long getId() {
28         return id;
29     }
30     public void setId(Long id) {
31         this.id = id;
32     }
33     public String getName() {
34         return name;
35     }
36     public void setName(String name) {
37         this.name = name;
38     }
39     public String getCity() {
40         return city;
41     }
42     public void setCity(String city) {
43         this.city = city;
44     }
45     public String getPassword() {
46         return password;
47     }
48     public void setPassword(String password) {
49         this.password = password;
50     }
51     public String getContact() {
52         return contact;
53     }
54     public void setContact(String contact) {
55         this.contact = contact;
56     }
57     public String getEmail() {
58         return email;
59     }
60     public void setEmail(String email) {
61         this.email = email;
62     }
63     public User(Long id, String name, String city, String password, String contact,
64     String email) {
65         super();
66         this.id = id;
67         this.name = name;
68         this.city = city;
69         this.password = password;
70         this.contact = contact;
71         this.email = email;
72     }
73     @Override
74     public String toString() {
75         return "User [id=" + id + ", name=" + name + ", city=" + city + ", password=" +
76             password + ", contact=" + contact
77             + ", email=" + email + "]";
78     }
79     public User() {
80         super();
81         // TODO Auto-generated constructor stub
82     }
83 }
```

18.5 com.spring.mvc.service

✓ AdminService.java

```
1 package com.spring.mvc.servic;
2
3 import com.spring.mvc.model.Admin;
4
5 public interface AdminService {
6     public Admin adminlogin(Admin admin);
7 }
8
```

✓ AdminServiceImpl.java

```
1 package com.spring.mvc.servic;
2
3 import javax.transaction.Transactional;..]
4 @Service
5 @Transactional
6 public class AdminSeviceImpl implements AdminService {
7
8     @Autowired
9     private AdminDAO adminDAO;
10    public Admin adminlogin(Admin admin) {
11        // TODO Auto-generated method stub
12        return adminDAO.adminlogin(admin);
13    }
14}
15
16
17
18
19
20
21 }
```

✓ BulkMailService.java

```
11 import org.springframework.stereotype.Service;
12
13
14 @Service
15 public class BulkMailService {
16
17
18     @Autowired
19     private JavaMailSender mailSender;
20
21     public void sendMails(String receiver , String subject, String body) {
22
23         MimeMessage message = this.mailSender.createMimeMessage();
24         MimeMessageHelper mimeHelper;
25
26         try {
27             mimeHelper = new MimeMessageHelper(message, true);
28             mimeHelper.setTo(receiver);
29             mimeHelper.setFrom("abcjob36@gmail.com");
30             mimeHelper.setSubject(subject);
31             mimeHelper.setText(body, true);
32             mailSender.send(message);
33
34         } catch (MessagingException e) {
35             System.out.println("Error Sending message" + e.getMessage());
36         }
37     }
38 }
```

✓ ForgotPasswordMail.java

```
1 package com.spring.mvc.servic;
2
3 import javax.mail.MessagingException;
4
5 @Service
6 public class ForgotPassEmailService {
7
8     @Autowired
9
10    private JavaMailSender mailSender;
11
12    public void resetMail(String emailID) {
13        MimeMessage message = this.mailSender.createMimeMessage();
14        MimeMessageHelper messageHelper;
15        try {
16            messageHelper = new MimeMessageHelper(message, true);
17            messageHelper.setTo(emailID);
18            messageHelper.setFrom("abcjobs36@gmail.com");
19            messageHelper.setSubject("Forgot password reset");
20            messageHelper.setText(
21                "<html><body style='font-size: 1rem;'><h2>Good day! This is ABC JOBS</h2><h4>Use this link to
22                + <button style='border-radius: 1.5rem; text-decoration:none;background:white;padding:15px; color:
23                + "white">
24                + " border-radius: 1.5rem; text-decoration:none;background:white;padding:15px; color:
25                + "a href='http://localhost:8082/portalAbc/reset_forgotpassform/" + emailID
26                + "/'> Reset your Password</a></button> </body></html>",
27                true);
28            mailSender.send(message);
29        } catch (MessagingException e) {
30            System.out.println("Error sending mail" + e.getMessage());
31        }
32    }
33 }
34 }
```

✓ MailService.java

```
1 package com.spring.mvc.servic;
2
3 import java.util.Date;
4
5 @Service
6 public class MailService {
7
8     @Autowired
9     private JavaMailSender mailSender;
10
11     public void sendMail(String emailId, String userName) {
12
13         MimeMessage message = this.mailSender.createMimeMessage();
14         MimeMessageHelper mimeHelper;
15
16         try {
17             mimeHelper = new MimeMessageHelper(message, true);
18             mimeHelper.setTo(emailId);
19             mimeHelper.setFrom("abcjob36@gmail.com");
20             mimeHelper.setSubject("Registration Confirmation");
21             mimeHelper.setText("<html><body><h1>ABC Ptd Ltd</h1>" + "\n" + "<h3>Hi " + userName + "</h3>" + "\n"
22                 + "<h3 style='color: #0000ff; font-size: 1.2em; margin-top: 10px;'>You have registered successfully</h3>\r\n"
23                 + "" + "\n" + new Date() + "</body></html>", true);
24             mailSender.send(message);
25
26         } catch (MessagingException e) {
27             System.out.println("Error Sending message" + e.getMessage());
28         }
29     }
30 }
31 }
```

✓ UserService.java

```
1 package com.spring.mvc.servic;
2
3 import com.spring.mvc.model.User;
4
5
6 public interface UserService {
7     public void registerUser(User user);
8     public User loginUser(User user);
9     public User checkMail(User user);
10 }
11 }
```

✓ UserServiceImplementation.java

```
1 package com.spring.mvc.servic;
2
3④ import javax.transaction.Transactional;□
4 @Service
5 @Transactional
6 @Repository
7 public class UserServiceImpl implements UserService {
8    @Autowired
9    private UserDAO userDAO;
10
11    public void registerUser(User user) {
12        userDAO.registerUser(user);
13    }
14
15    public User loginUser(User user) {
16        return userDAO.loginUser(user);
17    }
18
19    public User checkMail(User user) {
20        // TODO Auto-generated method stub
21        return userDAO.checkMail(user);
22    }
23
24}
```

Task 7

12. Implemented pages

✓ Home Page

The screenshot shows the homepage of the ABC Jobs website. At the top, there's a navigation bar with 'ABC Jobs' on the left, 'Login' and 'Register' buttons on the right, and a search bar in the center. Below the header is a large, colorful illustration of a hand holding a red magnet over a smartphone screen, which displays a target with an arrow hitting it. The background of the page features a yellow banner on the left with the text 'Let the right people know you're open to work' and a 'View More' button. To the right of the banner is a cartoon illustration of a man sitting at a desk with a laptop, surrounded by books, papers, and a cat, with a speech bubble above him. The main content area has a white background with three circular callout boxes: 'Find the right job', 'Post your Job for million of people', and 'Connect with people'. Each box contains a brief description and a 'Get a Quote' button. At the bottom, there's a footer section with links for 'ABOUT US', 'OUR SERVICES', 'USEFUL LINKS', and 'CONTACT', along with social media icons and a copyright notice.

ABC Jobs

Login Register

Welcome to your Professional Community

The Easiest Way to Get Yours New Job Find Jobs, Receive Instant Job Alerts, Apply Online. Employers Post jobs for free!

Let the right people know you're open to work

With the Open To Work feature, you privately tell recruiters or publicly share with the ABC Community that you are looking for new job opportunities

View More

ABC Jobs Community

Find the right job

Once your profile is built out, you can integrate a few strategic features that ABC offers specifically to job seekers to boost your visibility and help with targeting the right opportunities

Post your Job for million of people

With ABC Jobs, you'll get easy-to-use tools to help you find the right hire quickly. You'll be able to reach and engage with a community of millions of job seekers who visit ABC every week.

Connect with people

Building your ABC network is a great way to stay in touch with alumni, colleagues, and recruiters, as well as connect with new, professional opportunities

Services

ABC JOBS

Marketing and salespeople frequently express their love/hate relationship with LinkedIn. Do you think of the site as a form of social stalking for businesspeople? There are various powerful ABC features that are very helpful in your regular problems. If you simply use the site to search up a prospect's credentials or to check a connection request email, you're missing out on important possibilities to further your career, increase your sales leads, and, ultimately, expand your business.

Get a Quote

ABOUT US

ABC is an Sri Lanka business and employment-oriented online service that operates via websites and mobile apps. Launched on July 7, 2022, the platform is primarily used for professional networking and career development, and allows job seekers to post their CVs and employers to post jobs.

OUR SERVICES

[Web Design](#)
[Web Development](#)
[Marketing](#)
[Product Management](#)
[Offer Jobs](#)

USEFUL LINKS

[Home](#)
[About Us](#)
[Services](#)
[Term of Service](#)
[Privacy Policy](#)

CONTACT

23/B
Virtusa Road
Colombo 8
abcjobs36@example.com
+ 94 762269976
+ 94 760528233

© 2021 Copyright: All Rights Reserved by ABC JOBS

Figure 37: Screenshot of Home Page

✓ Registration Page

ABC Jobs

[Login](#) [Register](#)

Sign up

Name

Email

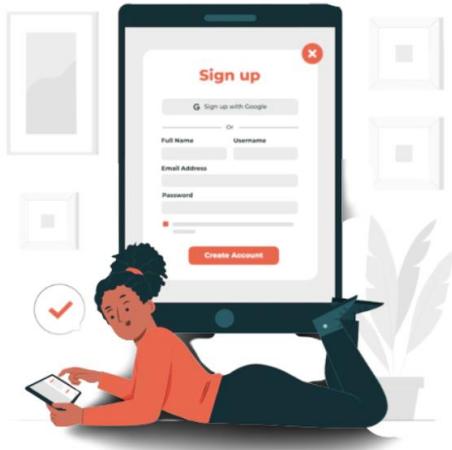
Phone

City

Password

Repeat Password

[Register](#)



ABOUT US
ABC is an Sri Lanka business and employment-oriented online service that operates via websites and mobile apps. Launched on July 7, 2022, the platform is primarily used for professional networking and career development, and allows job seekers to post their CVs and employers to post jobs.

OUR SERVICES
Web Design
Web Development
Marketing
Product Management
Offer Jobs

USEFUL LINKS
Home
About Us
Services
Term of Service
Privacy Policy

CONTACT
23/B
Virtusa Road
Colombo 8
abcjobs36@example.com
+ 94 762269976
+ 94 760528233

[f](#) [t](#) [g](#) [in](#)

© 2021 Copyright: All Rights Reserved by ABC JOBS

Figure 38: Screenshot of Registration Page

✓ Registration Thankyou page

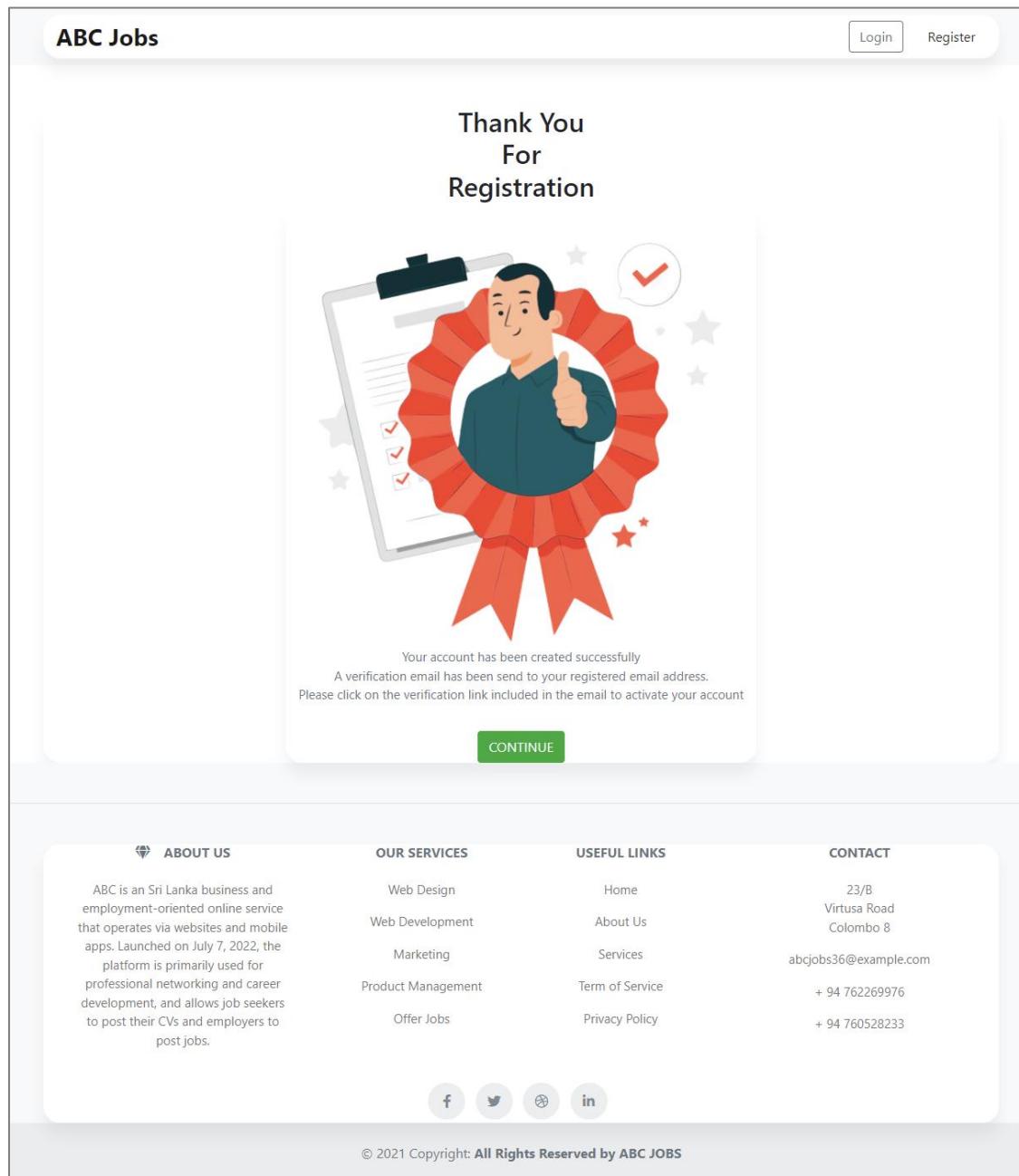


Figure 39: Screenshot of Registration Thank you Page

✓ **Login Page**

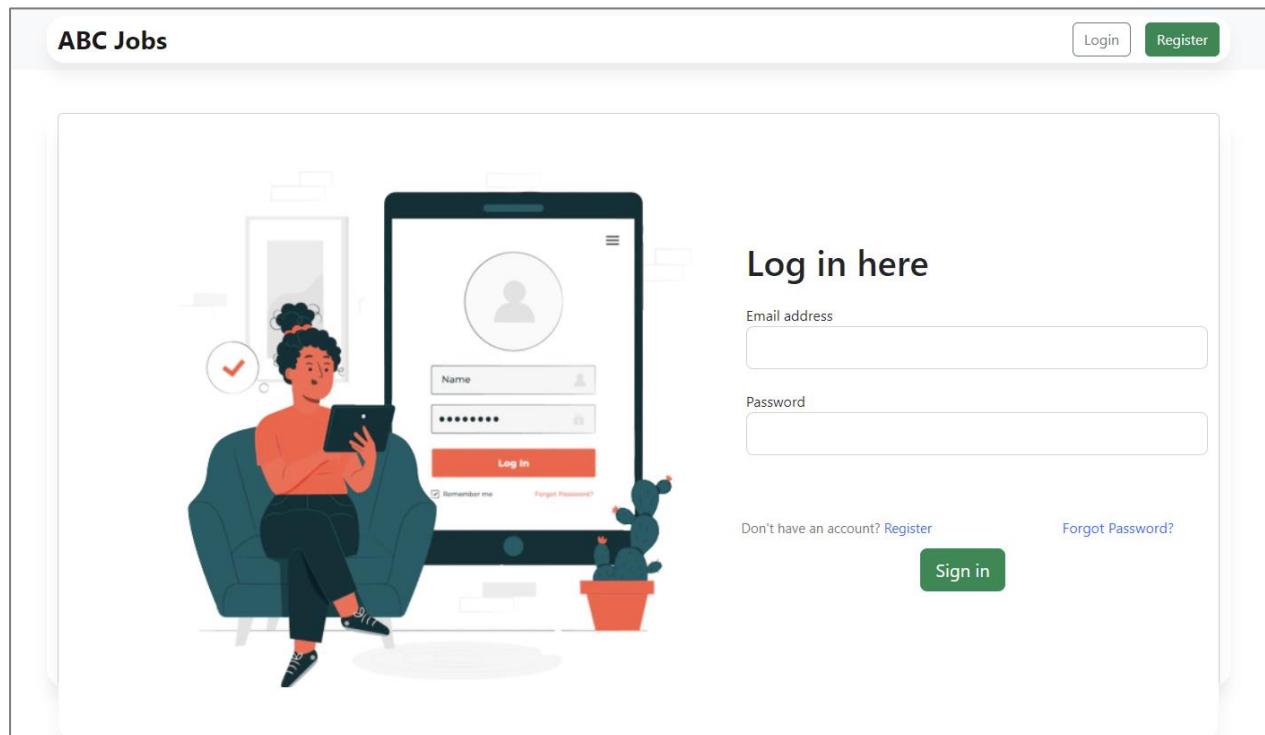


Figure 40: Screenshot of Login Page

✓ Forgot Password

The screenshot shows the 'Forgot Password' page of the ABC Jobs website. At the top, there is a navigation bar with links for HOME, PEOPLE, JOBS, Login (which is highlighted in red), and Register. Below the navigation bar, the heading 'Forgot your password???????' is displayed. A central illustration depicts a person with their hands on their head, looking stressed, with thought bubbles above them containing question marks and exclamation marks. Below the illustration, a text box contains the placeholder 'Enter the email address associated with your account. We will email you a link to reset your password'. There is a red input field labeled 'Enter email' and a green button labeled 'Reset Password'.

ABOUT US
ABC is an Sri Lanka business and employment-oriented online service that operates via websites and mobile apps. Launched on July 7, 2022, the platform is primarily used for professional networking and career development, and allows job seekers to post their CVs and employers to post jobs.

OUR SERVICES
Web Design
Web Development
Marketing
Product Management
Offer Jobs

USEFUL LINKS
Home
About Us
Services
Term of Service
Privacy Policy

CONTACT
23/B
Virtusa Road
Colombo 8
abcjobs36@example.com
+ 94 762269976
+ 94 760528233

© 2021 Copyright: All Rights Reserved by ABC JOBS

Figure 41: Screenshot of Forgot Password Page

✓ **Forgot Password Confirmation**

Forgot Password Confirmation



As per your request,
you will receive an email with a temporary password.
Use this password to login.
Once logged in, you will be able to change your password.

[Back To Home](#)

Figure 42: Screenshot of Forgot Password Confirmation Page

✓ User Home Page

The screenshot displays the User Home Page for the ABC Jobs platform. The top navigation bar includes links for HOME, PEOPLE, JOBS, MY PROFILE, Logout, and Register. The main content area is divided into several sections:

- Activity:** Shows recent posts from users like Leo Ashley, Jessica Wong, and Kevin Wix.
- Manage My Work:** A summary of user metrics including Connections (500+), Events (250+), and Pages (1765).
- My Skills:** A section showing Default Primary Success Info Warning Danger Success Ui Design Primary.
- Jobs:** A list of job postings:
 - Senior Software Engineer (Python):** Virtusa, Sri Lanka (Remote), Full Time. Includes a "More" button.
 - Full Stack Developer:** Soul Studio, Maldives. Includes a "More" button.
 - Resort Housekeeper:** Sun Communities & Sun Doors, 45/C St.Francis Road, Colombo 7. Includes a "More" button.

Figure 43: Screenshot of User Home Page

✓ User Profile Page

The screenshot shows the ABC Jobs user profile page for a user named Chathu. At the top, there's a navigation bar with links for HOME, PEOPLE, JOBS, MY PROFILE, Logout, and Register. Below the navigation is a large black rectangular area containing a placeholder image for a profile picture. To the left of this area is a small circular placeholder icon featuring a cartoon character holding a book and a travel pass. To the right of the black area is a white box containing the user's profile information.

About
A Lead UX & UI designer based in Canada
I design and develop services for customers of all sizes, specializing in creating stylish, modern websites, web services and online stores. My passion is to design digital user experiences through the bold interface and meaningful interactions.

Chathu

[Edit profile](#)

User Id	6
Name	Chathu
Email	chathushi0707@gmail.com
Phone	0760528233
City	Bandarawela
Profession	Lead UX & UI Designer

Figure 44: Screenshot of User Profile Page

✓ Search User Page

The screenshot shows a search results page for users. At the top, there is a navigation bar with links for HOME, PEOPLE, JOBS, MY PROFILE, Logout, and Register. Below the navigation bar, a search bar is labeled "Search People Here" with a placeholder "Search" and a magnifying glass icon. A table displays four user profiles, each with a profile picture, name, email, contact number, and a "View Profile" button.

User Picture	Name	Email	Contact	Action
	Chathushi	chathudi77@gmail.com	0742269976	View Profile
	Rithu	rithu@gmail.com	0553423456	View Profile
	Nethu	nethu@gmail.com	0712345678	View Profile
	Chathu	chathushi0707@gmail.com	0760528233	View Profile

Figure 45: Screenshot of Search User Page

✓ Admin View/edit/delete User Page

The screenshot shows an administrator view of user data. At the top, there is a navigation bar with links for ADMIN HOME, PEOPLE, JOBS, and Log-out. A table lists user information, including User id, User city, User email, User contact, User name, User password, and Action (with edit and delete icons). Each row corresponds to a user entry from the previous screenshot.

User id	User city	User email	User contact	User name	User password	Action
1		chathudi77@gmail.c	0742269976	Chathushi	12345	
3		rithu@gmail.com	0553423456	Rithu	56789	
5	Colombo	nethu@gmail.com	0712345678	Nethu	09876	
6	Bandarawela	chathushi0707@gma	0760528233	Chathu	777777	
12	Tokyo	aarvi@gmail.com	0764567893	Aarvi	56789	
13	Kandy	silent@gmail.com	0745687546	Silent	1234567	

✓ Admin Profile Page

The screenshot shows the Admin Home Page for ABC Jobs. At the top, there is a navigation bar with links for ADMIN HOME, PEOPLE, JOBS, and Log-out. Below the navigation bar, a greeting "Hello Admin" is displayed. On the left side, there is a section titled "Projects" showing progress bars for various tasks: Server Migration (green, ~80%), Sales Tracking (pink, ~60%), Customer Database (red, ~70%), Payout Details (blue, ~50%), and Account Setup (purple, ~40%). Above these projects are four circular progress indicators, each showing 37% completion. In the center, there are several interactive buttons: "Search Users" (blue), "view users" (green), "Compose email" (yellow), and "Post Jobs" (red). To the right, there are four summary boxes: "EARNINGS (MONTHLY) \$40,000" (blue), "EARNINGS (ANNUAL) \$215,000" (green), "TASKS 50%" (grey), and "PENDING REQUESTS 18" (orange). At the bottom, there are sections for "ABOUT US", "OUR SERVICES", "USEFUL LINKS", and "CONTACT". The "ABOUT US" section contains a paragraph about ABC Jobs. The "OUR SERVICES" section lists Web Design, Web Development, Marketing, Product Management, and Offer Jobs. The "USEFUL LINKS" section includes Home, About Us, Services, Term of Service, Privacy Policy, and external links. The "CONTACT" section provides address, email, and phone numbers. Social media icons for Facebook, Twitter, LinkedIn, and YouTube are also present at the bottom.

ABC Jobs

ADMIN HOME PEOPLE JOBS Log-out

Hello Admin

Projects

- Server Migration
- Sales Tracking
- Customer Database
- Payout Details
- Account Setup

Modify

- Search Users
- view users
- Email
- Post Jobs

EARNINGS (MONTHLY) \$40,000

EARNINGS (ANNUAL) \$215,000

TASKS 50%

PENDING REQUESTS 18

ABOUT US

ABC is an Sri Lanka business and employment-oriented online service that operates via websites and mobile apps. Launched on July 7, 2022, the platform is primarily used for professional networking and career development, and allows job seekers to post their CVs and employers to post jobs.

OUR SERVICES

- Web Design
- Web Development
- Marketing
- Product Management
- Offer Jobs

USEFUL LINKS

- Home
- About Us
- Services
- Term of Service
- Privacy Policy

CONTACT

23/B
Virtusa Road
Colombo 8
abcjobs36@example.com
+ 94 762269976
+ 94 760528233

© 2021 Copyright: All Rights Reserved by ABC JOBS

Figure 46: Screenshot of Admin Home Page

✓ **Admin Compose Mail Page (Bulk Email)**

The screenshot shows a web-based application for managing job posts. At the top, there's a navigation bar with the logo 'ABC Jobs' and links for 'ADMIN HOME', 'PEOPLE', 'JOBS', and a green 'Log-out' button. Below the navigation, the page title 'Admin Email' is displayed above a large text input area labeled 'Compose Email'. To the right of this input area is a green button with a user icon. Below the input area, there's a 'To:' field, a 'Subject:' field, and an 'Attachment:' section containing a 'Choose File' button which shows 'No file chosen'. At the bottom of the page is a prominent green 'Send Email' button.

Figure 47: Screenshot of Bulk Email Page

Task 8

13. Debug the application in Eclipse

Step 1 – Set Breakpoints

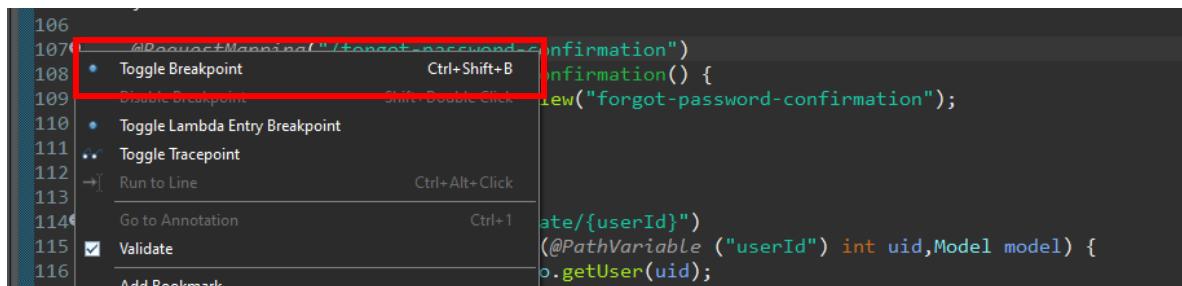
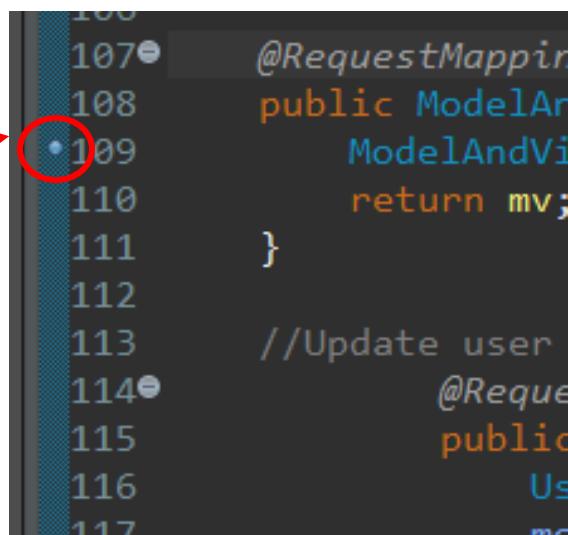


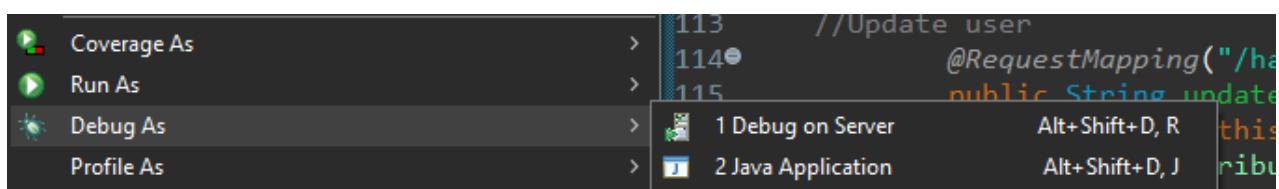
Figure 48: Screenshot of debugging in eclipse step 1

Breakpoint: A blue circle should then appear next to the line:



Step 2 – Start the program in Debug mode

(Right click on project → Debug as → Debug as server)



'ppppppppppppp' 'Figure pp' 000'0: Screenshot of debugging in eclipse step 2

Step 3 – Showing debugger details in the console

The screenshot shows the Eclipse IDE interface with the Java code for `MvcController.java` in the editor. The code defines a `MvcController` class with methods for handling login and user profile requests. Below the editor is the Java console window, which displays the Tomcat startup logs. The logs show the server starting up, loading configuration files, initializing SSL, and starting the Catalina service. The log entries are timestamped from Sep 22, 2022, at 8:54:07 PM.

```
Tomcat v9.0 Server at localhost [Apache Tomcat] C:\Program Files\Java\jdk-19\bin\java.exe (Sep 22, 2022, 8:54:02 PM) [pid: 13236]
INFO: Command line argument: -Dcatalina.home=C:\eclipse IDE\apache-tomcat-9.0.65
Sep 22, 2022 8:54:07 PM org.apache.catalina.startup.VersionLoggerListener log
INFO: Command line argument: -Dtp.deploy=C:\Users\PC\eclipse-workspace\metadata\plugins\org.eclipse.wst.server.core\tmp0\wtpwebapps
Sep 22, 2022 8:54:07 PM org.apache.catalina.startup.VersionLoggerListener log
INFO: Command line argument: -Djava.util.concurrent.ForkJoinPool.common.parallelism=16
Sep 22, 2022 8:54:07 PM org.apache.catalina.startup.VersionLoggerListener log
INFO: Command line argument: --add-opens=java.base/java.io=ALL-UNNAMED
Sep 22, 2022 8:54:07 PM org.apache.catalina.startup.VersionLoggerListener log
INFO: Command line argument: --add-opens=java.base/java.util=ALL-UNNAMED
Sep 22, 2022 8:54:07 PM org.apache.catalina.startup.VersionLoggerListener log
INFO: Command line argument: --add-opens=java.base/java.util.concurrent=ALL-UNNAMED
Sep 22, 2022 8:54:07 PM org.apache.catalina.startup.VersionLoggerListener log
INFO: Command line argument: --add-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED
Sep 22, 2022 8:54:07 PM org.apache.catalina.startup.VersionLoggerListener log
INFO: Command line argument: --add-opens=java.rmi/javax.naming=ALL-UNNAMED
Sep 22, 2022 8:54:07 PM org.apache.catalina.core.AprLifecycleListener lifecycleEvent
INFO: APR capabilities: IPv6 [true], sendfile [true], accept filters [false], random [true], UDOS [true].
Sep 22, 2022 8:54:07 PM org.apache.catalina.core.AprLifecycleListener lifecycleEvent
INFO: Loaded Apache Tomcat Native library [1.2.35] using APR version [1.7.0].
Sep 22, 2022 8:54:07 PM org.apache.catalina.core.AprLifecycleListener lifecycleEvent
INFO: APR SSL Configuration Support [disabled]. Valid cipherSuiteOrder: [true]
Sep 22, 2022 8:54:07 PM org.apache.catalina.core.AprLifecycleListener initializeSSL
INFO: OpenSSL successfully initialized [OpenSSL 1.1.1q 5 Jul 2022]
Sep 22, 2022 8:54:07 PM org.apache.coyote.AbstractProtocol init
INFO: Initializing ProtocolHandler ["http-nio-52226"]
Sep 22, 2022 8:54:07 PM org.apache.catalina.startup.Catalina load
INFO: Server initialization in (2151) milliseconds
Sep 22, 2022 8:54:07 PM org.apache.catalina.core.StandardService startInternal
INFO: Starting service [Catalina]
Sep 22, 2022 8:54:07 PM org.apache.catalina.core.StandardEngine startInternal
INFO: Starting Servlet engine: [Apache Tomcat/9.0.65]
Sep 22, 2022 8:54:10 PM org.apache.jasper.servlet.TldScanner scanJars
<-->
```

Figure 49: Screenshot of debugging in eclipse step 3

Step 4 - Controlling the program execution

The call stack indicates the parts of the program which can be currently executed and the way they relate to each other. The present-day stack is displayed inside the Debug view

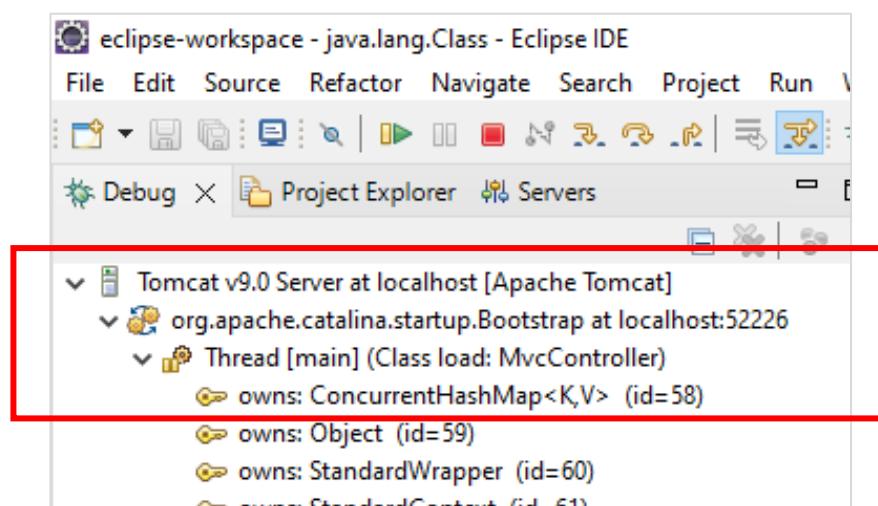


Figure 50: Screenshot of debugging in eclipse step 4

Step 5 - Breakpoints view and deactivating breakpoints

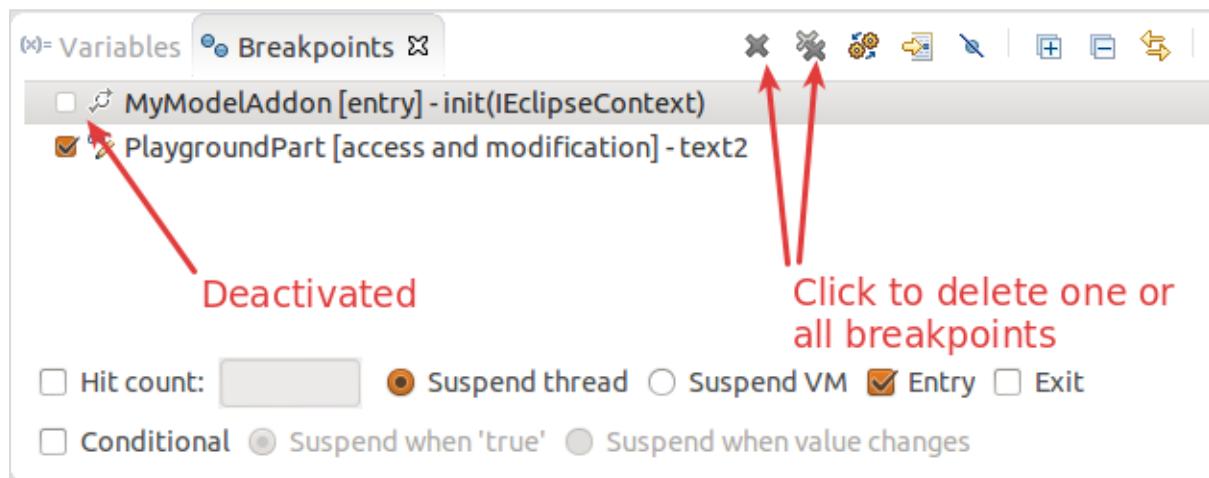


Figure 51: Screenshot of debugging in eclipse step 5

Step 6 - Evaluating variables in the debugger

The variables view shows fields and nearby variables from the contemporary executing stack.

Name	Value
validateUserSettings() returned	true
args	String[0] (id=15)
connection	ServerConnection (id=35)
userSettings	true

Figure 52: Screenshot of debugging in eclipse step 6

Step 7 – Use the drop-down menu to display static variables

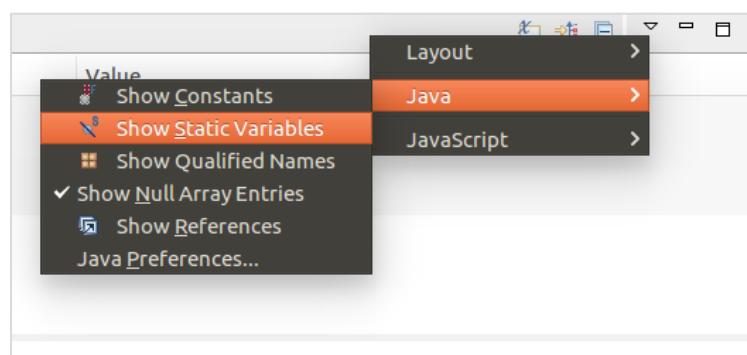


Figure 53: Screenshot of debugging in eclipse step 7

Step 8 - Changing variable assignments in the debugger

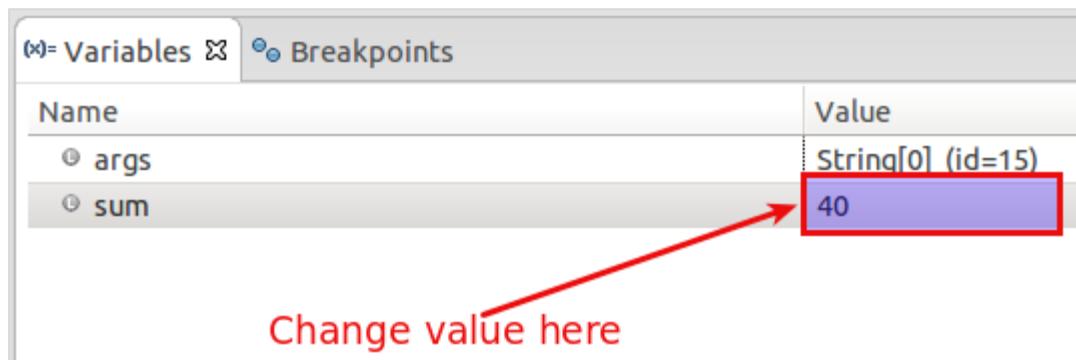


Figure 54: Screenshot of debugging in eclipse step 8

Step 9 - Controlling the display of the variables with detail formatter

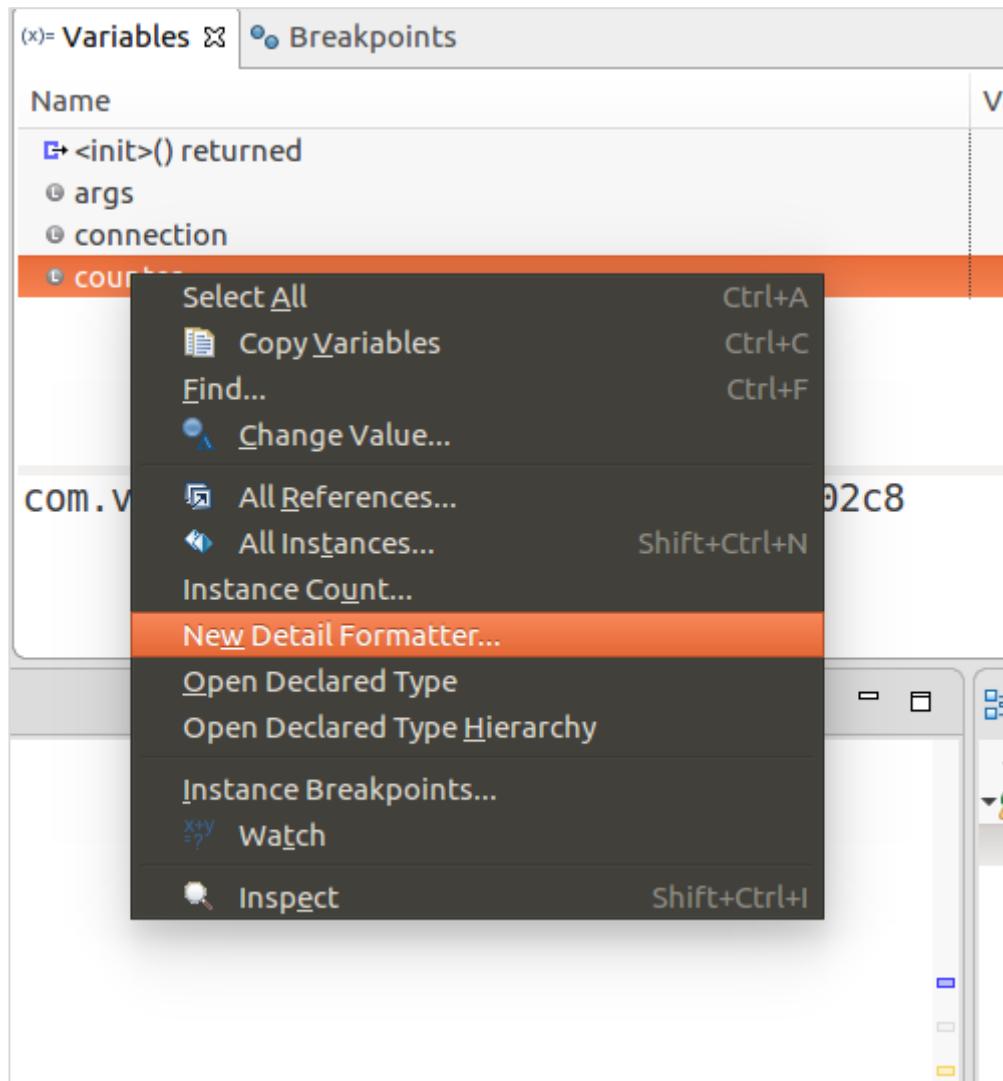
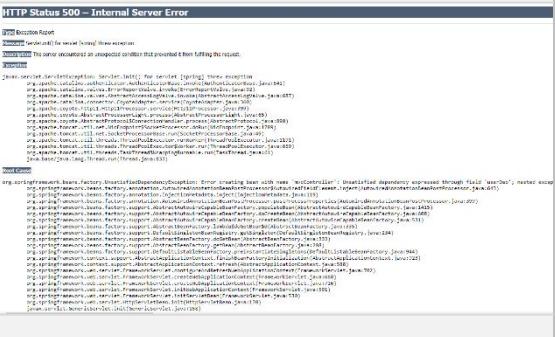
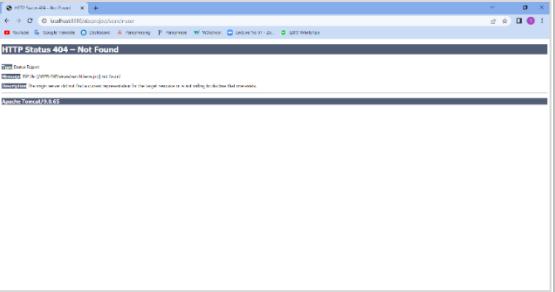
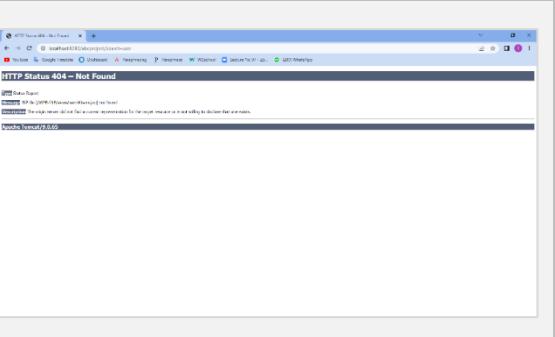


Figure 55: Screenshot of debugging in eclipse step 9

Task 9

14. Errors you encountered while developing the application.

Error	Error handling (Solution)
	<ul style="list-style-type: none"> - 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:5476 to jdbc:mysql://localhost:3306
	<ul style="list-style-type: none"> - Look at HTTP Status - Check out the Root Cause - Look at the description - The origin server did not find a current representation for the target resource or is not willing to disclose that one exists. - Check the MainController.java class - Add request mapping for home.jsp
	<ul style="list-style-type: none"> - Look at HTTP Status - Check out the Root Cause - Look at the message - JSP file [/WEB-INF/views/searchUsers.jsp] not found - Check the MainController.java class - Add return page to search method

Task 10

15. New feature to the application

I received feedback to add suggestions to the search user page. So I added account suggestions to the search user page

✓ Before

ABC Jobs

HOME PEOPLE JOBS Login Register

Search User

Name	Position	Place	Age	Connect
Tiger Nixon	System Architect	London	25	Connect
Garrett Winters	Accountant	Sri Lanka	46	Connect
Ashton Cox	Junior Technical Author	Tokyo	45	Connect
Cedric Kelly	Senior Javascript Developer	Edinburgh	30	Connect
Airi Satou	Accountant	Tokyo	28	Connect
Brielle Williamson	Integration Specialist	New York	22	Connect

Figure 56:Search User Page before Implementation new feature

✓ After

ABC Jobs

HOME PEOPLE JOBS Login Register

Search User

Name	Position	Place	Age	Connect
Tiger Nixon	System Architect	London	25	Connect
Garrett Winters	Accountant	Sri Lanka	46	Connect
Ashton Cox	Junior Technical Author	Tokyo	45	Connect
Cedric Kelly	Senior Javascript Developer	Edinburgh	30	Connect
Airi Satou	Accountant	Tokyo	28	Connect
Brielle Williamson	Integration Specialist	New York	22	Connect

Suggestions

 Tiger Nixon
System Architect
Lorem ipsum dolor sit, amet consectetur adipisicing elit. Accusamus numquam assumenda hic aliquam vero sequi velit molestias doloremque molestiae dicta?

 Garrett Winters
Accountant
Lorem ipsum dolor sit, amet consectetur adipisicing elit. Accusamus numquam assumenda hic aliquam vero sequi velit molestias doloremque molestiae dicta?

 Ashton Cox
Junior Technical Author
Lorem ipsum dolor sit, amet consectetur adipisicing elit. Accusamus numquam assumenda hic aliquam vero sequi velit molestias doloremque molestiae dicta?



Figure 57:Search User Page After Implementation new feature

Task 11

16. Unit Testing code

Scenario ID	Test Case	Preconditions	Test Step	Test Data	Expected Results	Actual Results	Pas / Fail
TS001	Check all the mandatory fields are entered	The website is configured to Google chrome by Apache Tomcat Server.	Go to Google chrome Type http://localhost:8080/ABCPortal/login Enter the fields. Click "Sign in now".	Email= "chathu@gmail.com" Password=	It will Display error message "Password is required"	As expected,	Pass
TS002	Check each hyperlink menu is loading the correct page	The website is configured to Google chrome by Apache Tomcat Server.	Go to Google chrome Type http://localhost:8080/ABCPortal/login	Click on the profile link Click on the search users link	Loaded profile page Loaded Search users page	As expected,	Pass

TS002	Check log out hyperlink is loading correct page	The website is configured to Google chrome by Apache Tomcat Server.	Go to Google chrome Type http://localhost:8080/BCPortal/home	Click on the log out link at the top nav bar	Loaded login page	As expected,	Pas
-------	---	---	---	--	-------------------	--------------	-----