Mini Project - 3

Student Name/ID Number:	Chathushi Jayarathna
Unit Number and Title:	ACWD Module 5 – Web Development Foundations
Academic Year:	**2022/23**
Unit Assessor:	Ms. Arvinder Kaur
Project Title:	Development of a Registration Form in JSP with Database (Not Struts)
Issue Date:	22/09/2022
Submission Date:	19/10/2022
Internal Verifier Name:	
Date:	19/10/2022

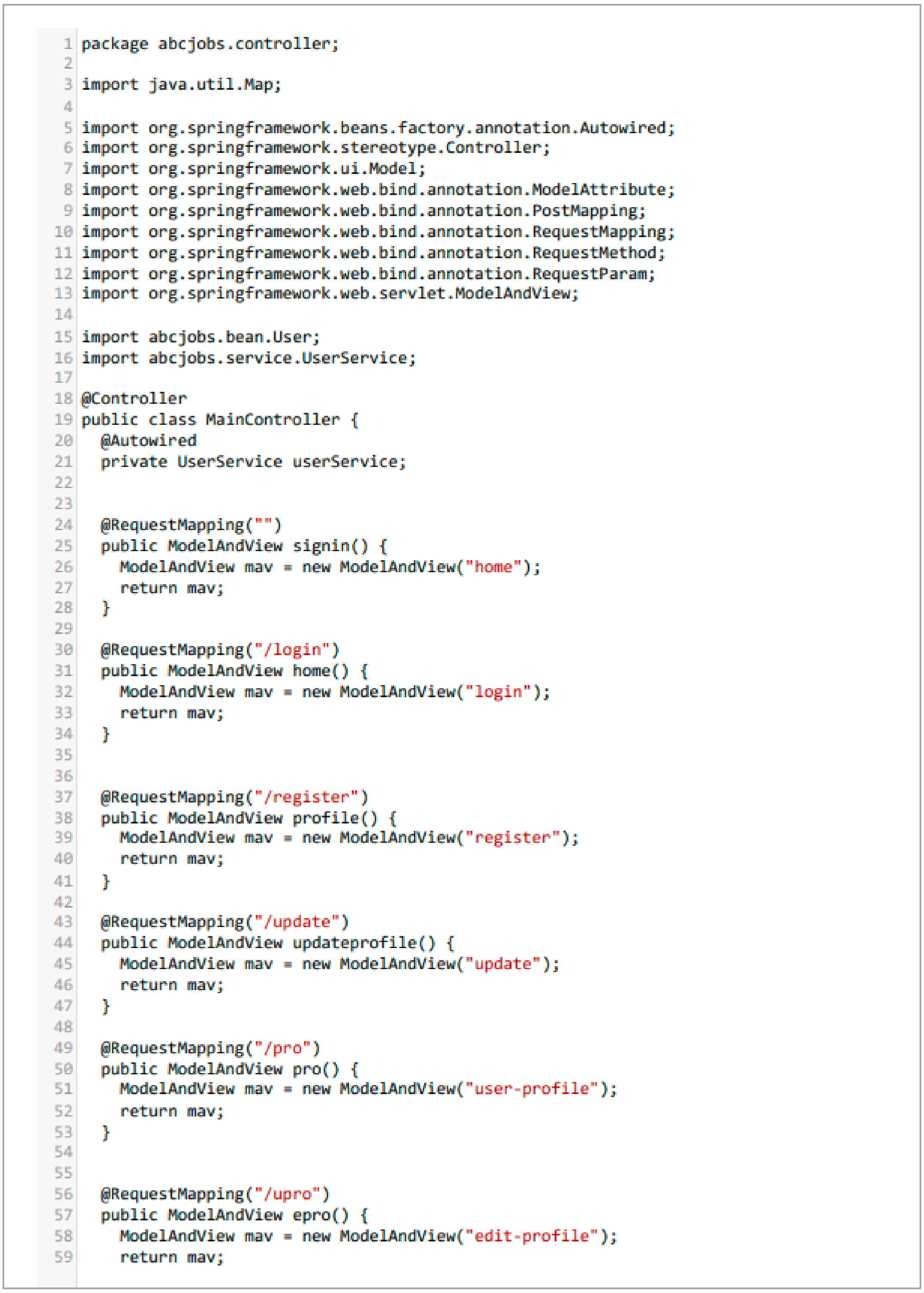
Learner declaration

I certify that the work submitted for this assignment is my own and that research sources are fully acknowledged.

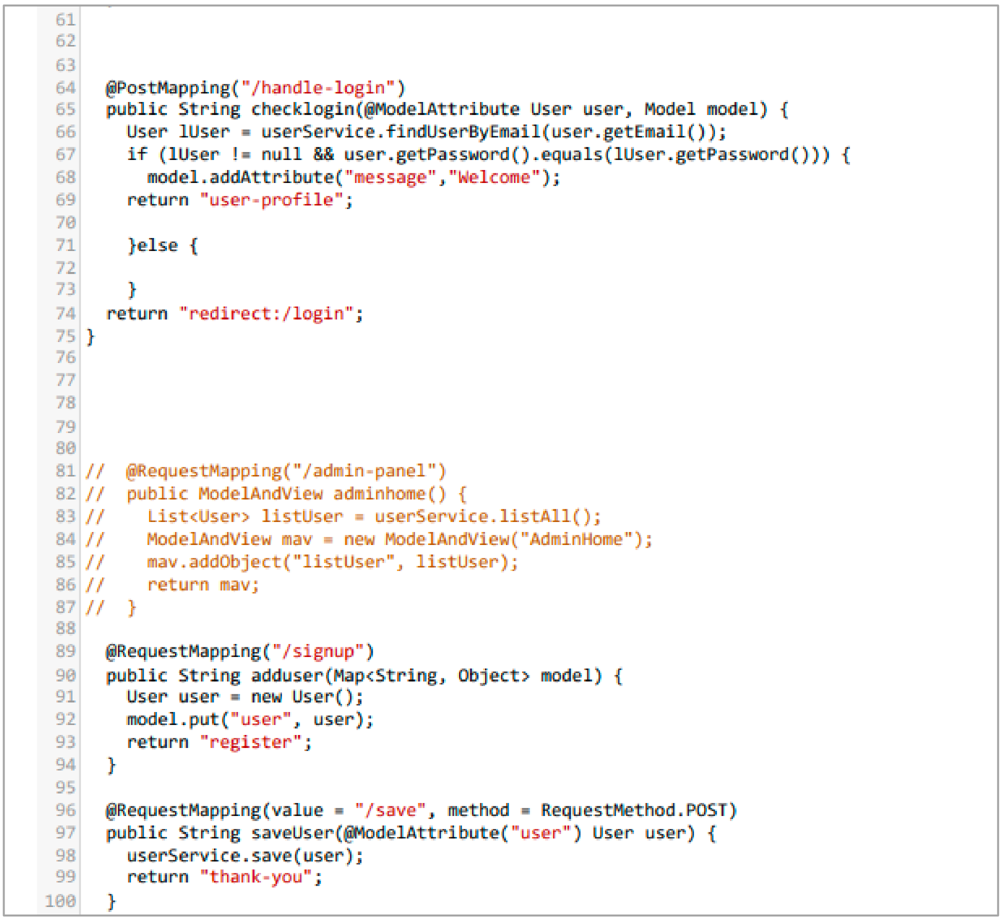
Student signature: **Chathushi** Date:17/10/2022

01) Create the Login System

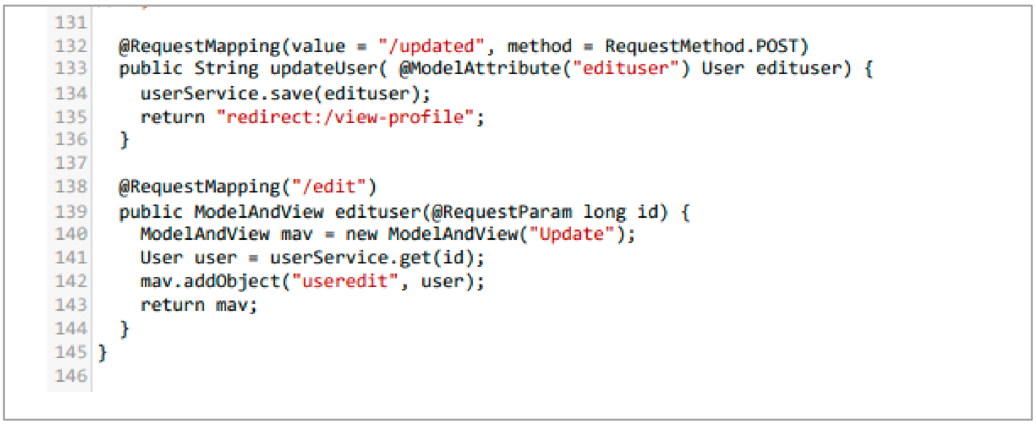
- **Controller**



```java
package abcjobs.controller;

import java.util.Map;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.servlet.ModelAndView;

import abcjobs.bean.User;
import abcjobs.service.UserService;

@Controller
public class MainController {
  @Autowired
  private UserService userService;


  @RequestMapping("")
  public ModelAndView signin() {
    ModelAndView mav = new ModelAndView("home");
    return mav;
  }

  @RequestMapping("/login")
  public ModelAndView home() {
    ModelAndView mav = new ModelAndView("login");
    return mav;
  }


  @RequestMapping("/register")
  public ModelAndView profile() {
    ModelAndView mav = new ModelAndView("register");
    return mav;
  }

  @RequestMapping("/update")
  public ModelAndView updateprofile() {
    ModelAndView mav = new ModelAndView("update");
    return mav;
  }

  @RequestMapping("/pro")
  public ModelAndView pro() {
    ModelAndView mav = new ModelAndView("user-profile");
    return mav;
  }


  @RequestMapping("/upro")
  public ModelAndView epro() {
    ModelAndView mav = new ModelAndView("edit-profile");
    return mav;
```

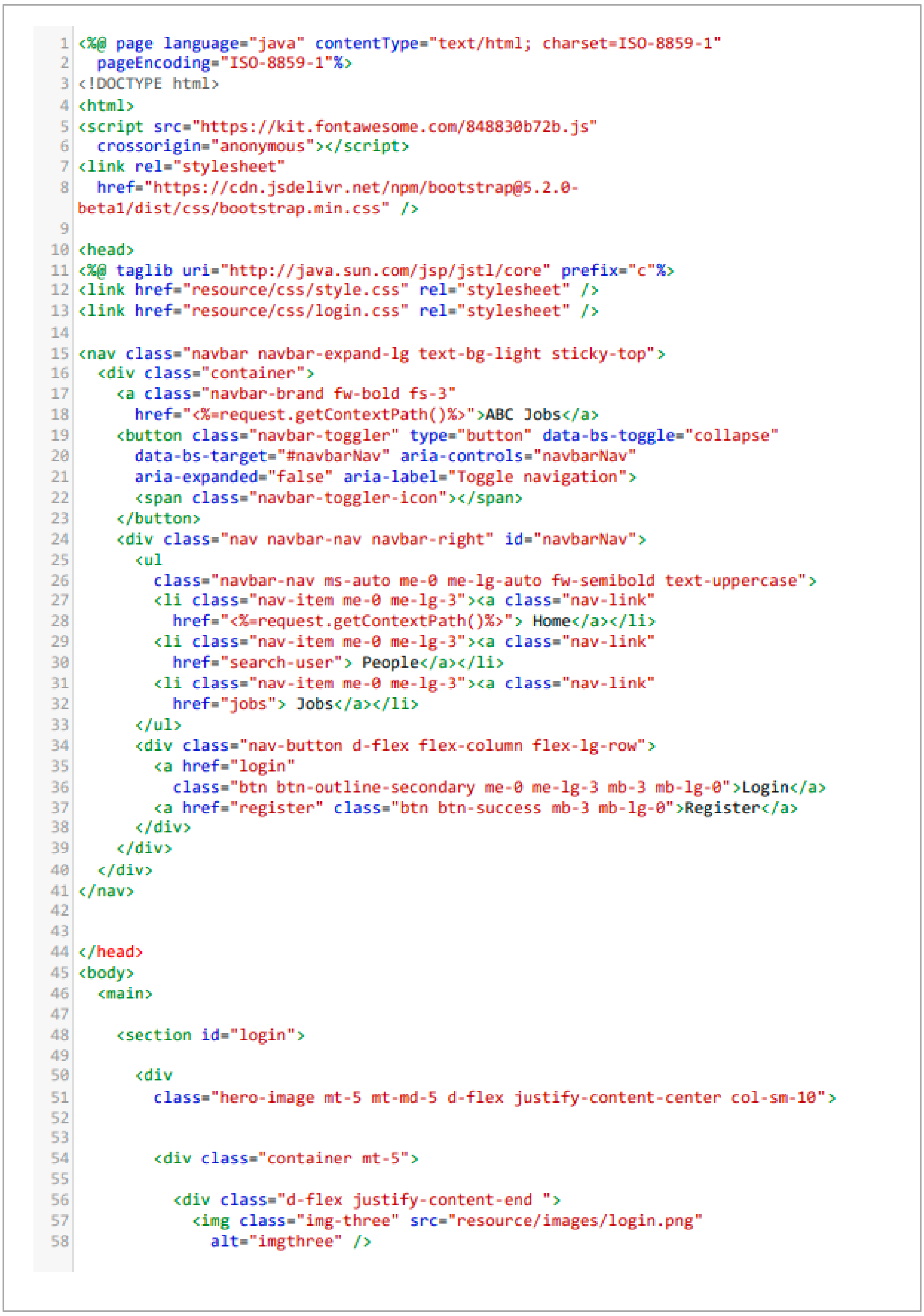


```



  @PostMapping("/handle-login")
  public String checklogin(@ModelAttribute User user, Model model) {
    User lUser = userService.findUserByEmail(user.getEmail());
    if (lUser != null && user.getPassword().equals(lUser.getPassword())) {
      model.addAttribute("message","Welcome");
    return "user-profile";

    }else {

    }
  return "redirect:/login";
}





//  @RequestMapping("/admin-panel")
//  public ModelAndView adminhome() {
//    List<User> listUser = userService.listAll();
//    ModelAndView mav = new ModelAndView("AdminHome");
//    mav.addObject("listUser", listUser);
//    return mav;
//  }

  @RequestMapping("/signup")
  public String adduser(Map<String, Object> model) {
    User user = new User();
    model.put("user", user);
    return "register";
  }

  @RequestMapping(value = "/save", method = RequestMethod.POST)
  public String saveUser(@ModelAttribute("user") User user) {
    userService.save(user);
    return "thank-you";
  }
```

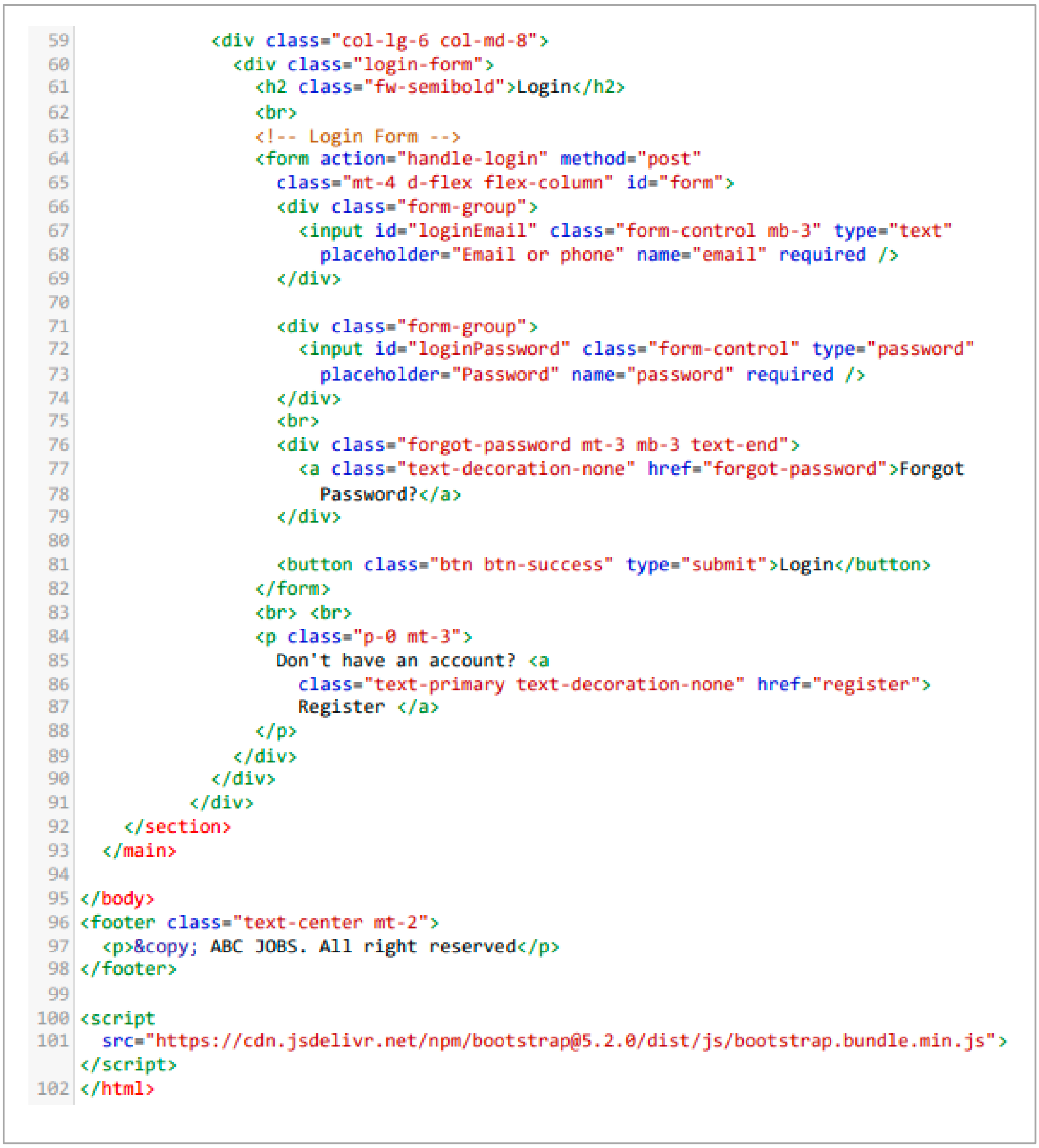


```

  @RequestMapping(value = "/updated", method = RequestMethod.POST)
  public String updateUser( @ModelAttribute("edituser") User edituser) {
    userService.save(edituser);
    return "redirect:/view-profile";
  }

  @RequestMapping("/edit")
  public ModelAndView edituser(@RequestParam long id) {
    ModelAndView mav = new ModelAndView("Update");
    User user = userService.get(id);
    mav.addObject("useredit", user);
    return mav;
  }
}

```

- **JSP**

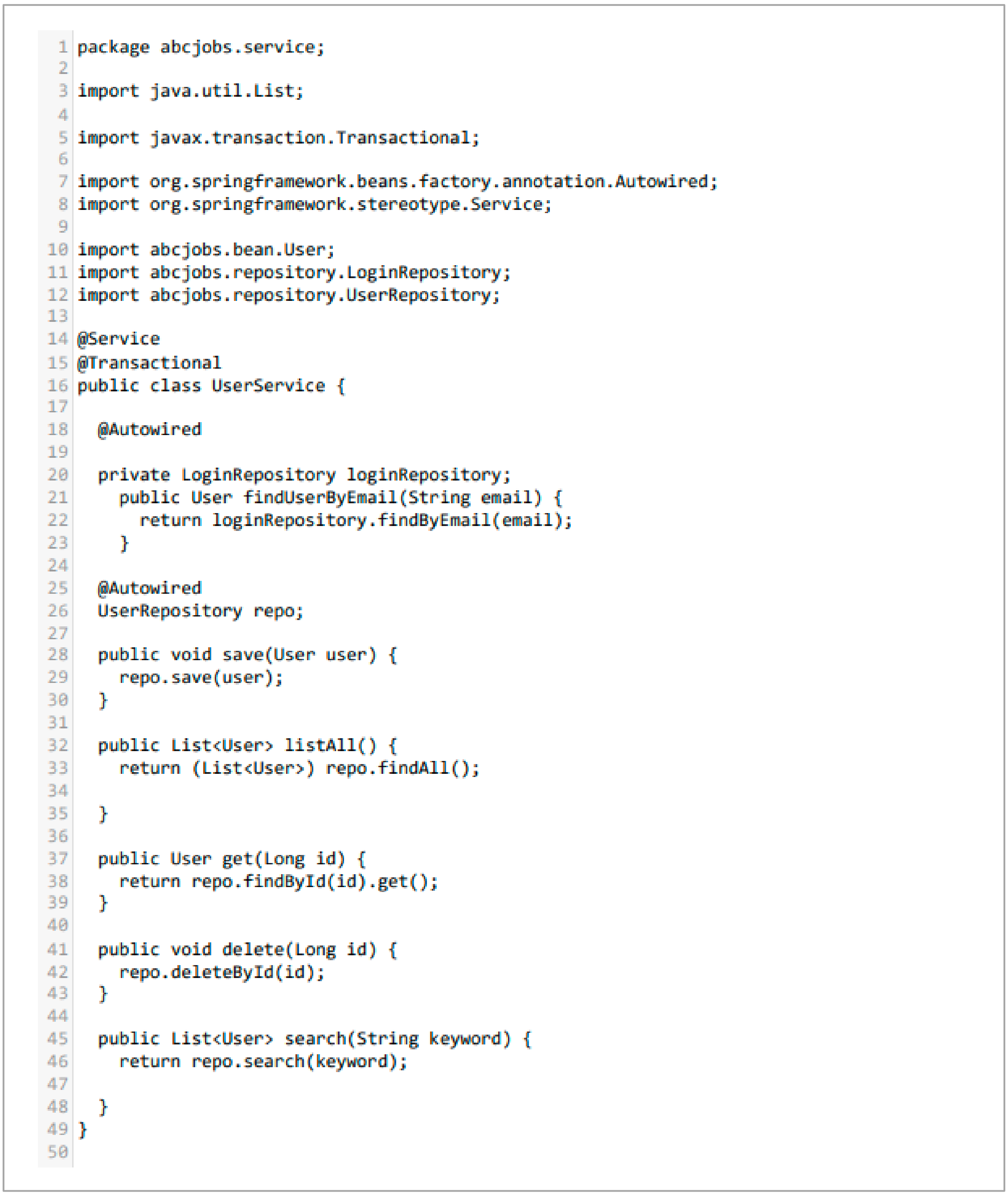


```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
  pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<script src="https://kit.fontawesome.com/848830b72b.js"
  crossorigin="anonymous"></script>
<link rel="stylesheet"
  href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-beta1/dist/css/bootstrap.min.css" />

<head>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<link href="resource/css/style.css" rel="stylesheet" />
<link href="resource/css/login.css" rel="stylesheet" />

<nav class="navbar navbar-expand-lg text-bg-light sticky-top">
  <div class="container">
    <a class="navbar-brand fw-bold fs-3"
      href="<%=request.getContextPath()%>">ABC Jobs</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
      data-bs-target="#navbarNav" aria-controls="navbarNav"
      aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="nav navbar-nav navbar-right" id="navbarNav">
      <ul
        class="navbar-nav ms-auto me-0 me-lg-auto fw-semibold text-uppercase">
        <li class="nav-item me-0 me-lg-3"><a class="nav-link"
          href="<%=request.getContextPath()%>"> Home</a></li>
        <li class="nav-item me-0 me-lg-3"><a class="nav-link"
          href="search-user"> People</a></li>
        <li class="nav-item me-0 me-lg-3"><a class="nav-link"
          href="jobs"> Jobs</a></li>
      </ul>
      <div class="nav-button d-flex flex-column flex-lg-row">
        <a href="login"
          class="btn btn-outline-secondary me-0 me-lg-3 mb-3 mb-lg-0">Login</a>
        <a href="register" class="btn btn-success mb-3 mb-lg-0">Register</a>
      </div>
    </div>
  </div>
</nav>


</head>
<body>
  <main>

    <section id="login">

      <div
        class="hero-image mt-5 mt-md-5 d-flex justify-content-center col-sm-10">


        <div class="container mt-5">

          <div class="d-flex justify-content-end ">
            <img class="img-three" src="resource/images/login.png"
              alt="imgthree" />
```



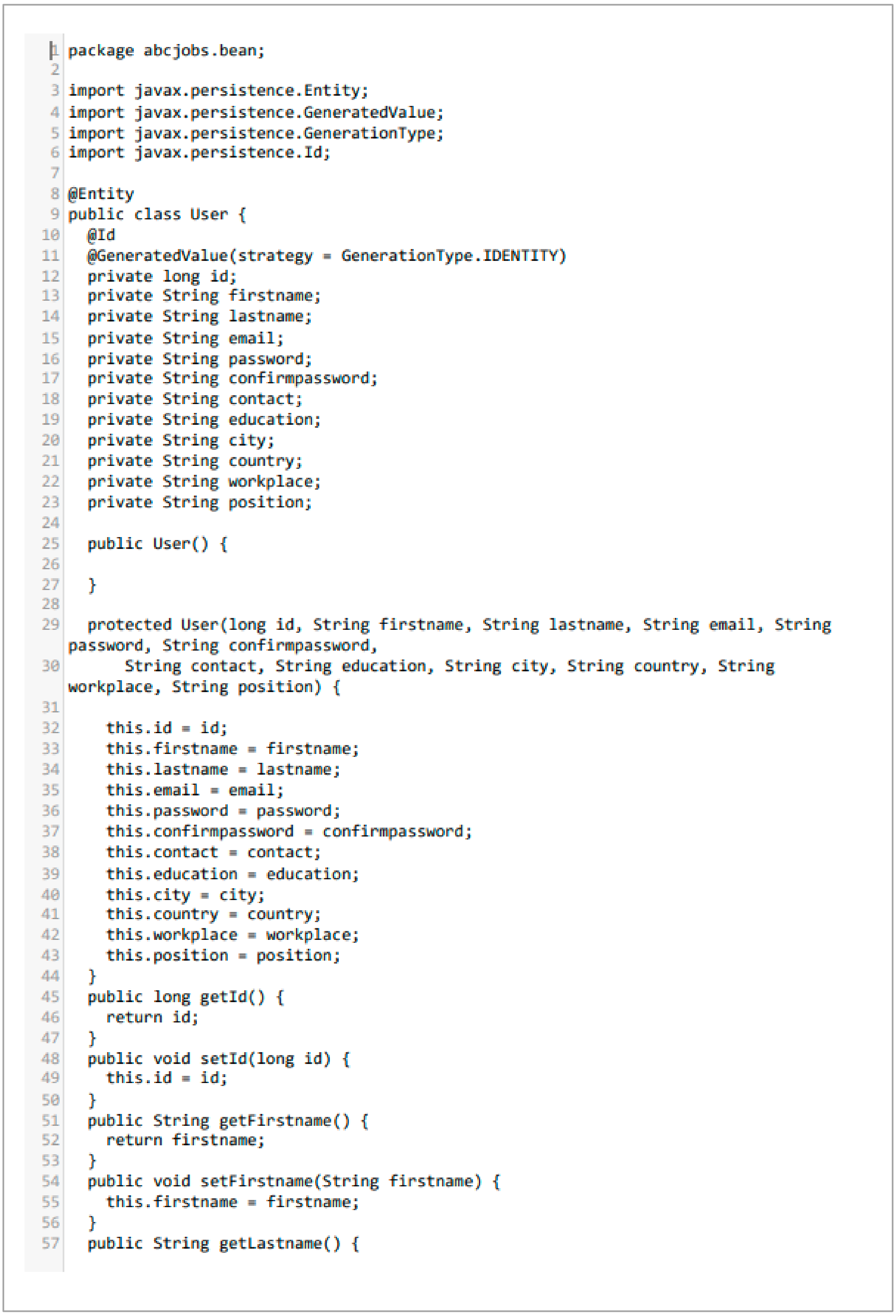
```html
            <div class="col-lg-6 col-md-8">
              <div class="login-form">
                <h2 class="fw-semibold">Login</h2>
                <br>
                <!-- Login Form -->
                <form action="handle-login" method="post"
                  class="mt-4 d-flex flex-column" id="form">
                  <div class="form-group">
                    <input id="loginEmail" class="form-control mb-3" type="text"
                      placeholder="Email or phone" name="email" required />
                  </div>

                  <div class="form-group">
                    <input id="loginPassword" class="form-control" type="password"
                      placeholder="Password" name="password" required />
                  </div>
                  <br>
                  <div class="forgot-password mt-3 mb-3 text-end">
                    <a class="text-decoration-none" href="forgot-password">Forgot
                      Password?</a>
                  </div>

                  <button class="btn btn-success" type="submit">Login</button>
                </form>
                <br> <br>
                <p class="p-0 mt-3">
                  Don't have an account? <a
                    class="text-primary text-decoration-none" href="register">
                    Register </a>
                </p>
              </div>
            </div>
          </div>
    </section>
  </main>

</body>
<footer class="text-center mt-2">
  <p>&copy; ABC JOBS. All right reserved</p>
</footer>

<script
  src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0/dist/js/bootstrap.bundle.min.js">
</script>
</html>
```

- **Service**

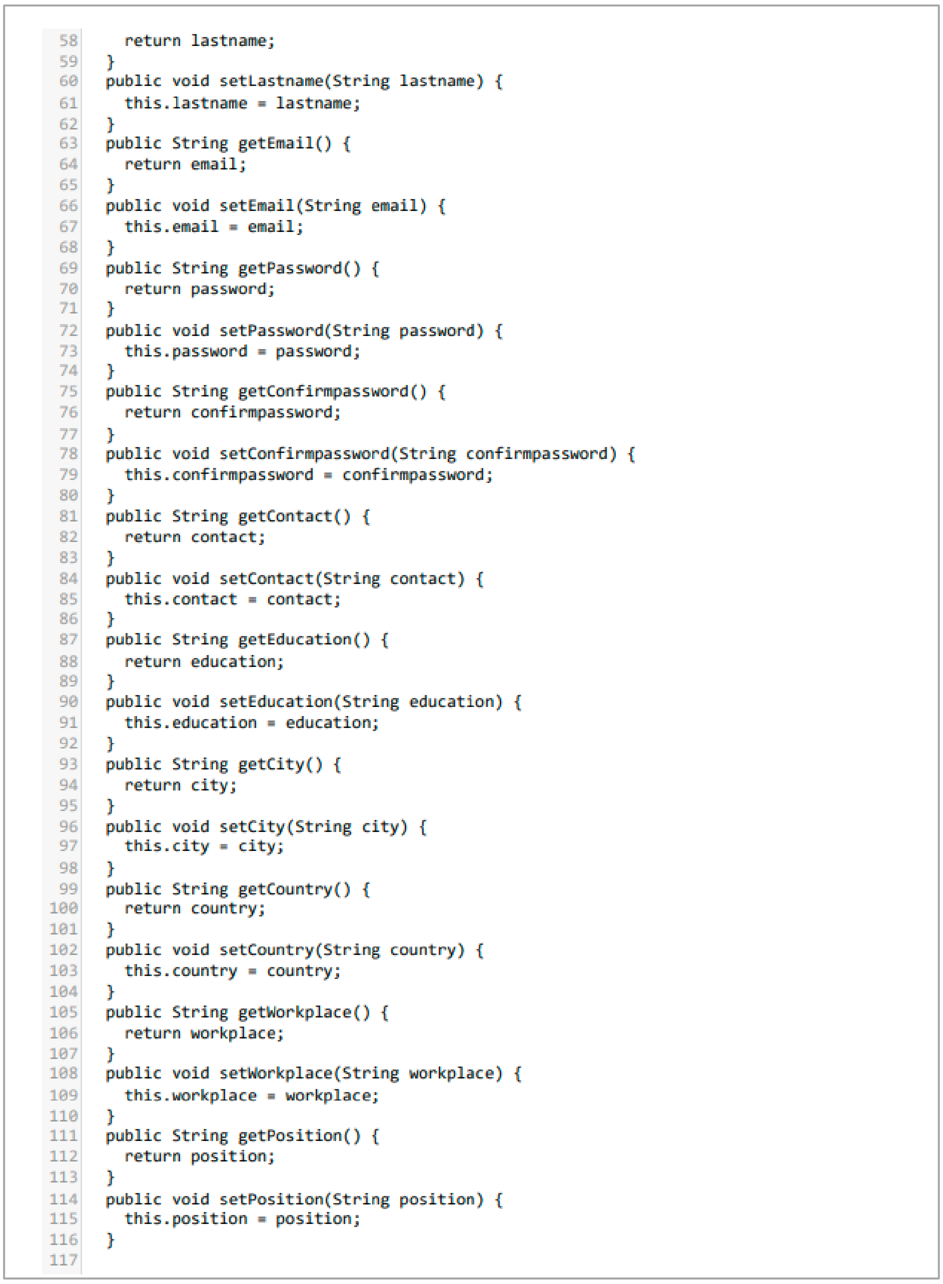


```
package abcjobs.service;

import java.util.List;

import javax.transaction.Transactional;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import abcjobs.bean.User;
import abcjobs.repository.LoginRepository;
import abcjobs.repository.UserRepository;

@Service
@Transactional
public class UserService {

  @Autowired

  private LoginRepository loginRepository;
    public User findUserByEmail(String email) {
      return loginRepository.findByEmail(email);
    }

  @Autowired
  UserRepository repo;

  public void save(User user) {
    repo.save(user);
  }

  public List<User> listAll() {
    return (List<User>) repo.findAll();

  }

  public User get(Long id) {
    return repo.findById(id).get();
  }

  public void delete(Long id) {
    repo.deleteById(id);
  }

  public List<User> search(String keyword) {
    return repo.search(keyword);

  }
}

```

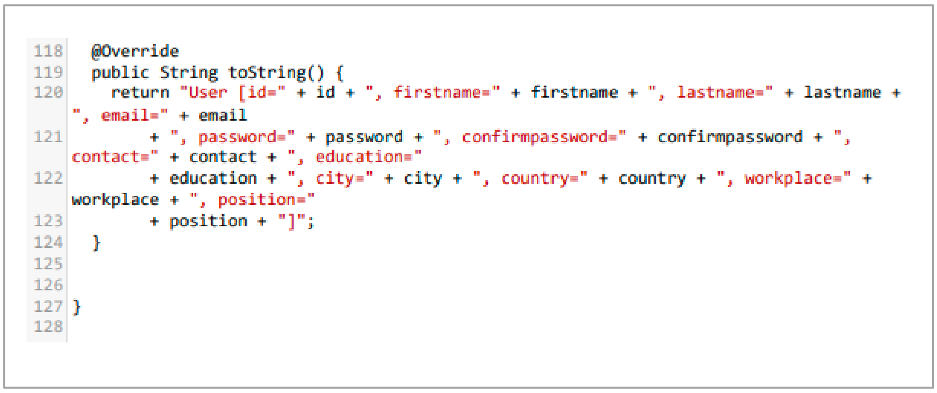
- **Bean**



```
package abcjobs.bean;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;

@Entity
public class User {
  @Id
  @GeneratedValue(strategy = GenerationType.IDENTITY)
  private long id;
  private String firstname;
  private String lastname;
  private String email;
  private String password;
  private String confirmpassword;
  private String contact;
  private String education;
  private String city;
  private String country;
  private String workplace;
  private String position;

  public User() {

  }

  protected User(long id, String firstname, String lastname, String email, String
password, String confirmpassword,
      String contact, String education, String city, String country, String
workplace, String position) {

    this.id = id;
    this.firstname = firstname;
    this.lastname = lastname;
    this.email = email;
    this.password = password;
    this.confirmpassword = confirmpassword;
    this.contact = contact;
    this.education = education;
    this.city = city;
    this.country = country;
    this.workplace = workplace;
    this.position = position;
  }
  public long getId() {
    return id;
  }
  public void setId(long id) {
    this.id = id;
  }
  public String getFirstname() {
    return firstname;
  }
  public void setFirstname(String firstname) {
    this.firstname = firstname;
  }
  public String getLastname() {
```

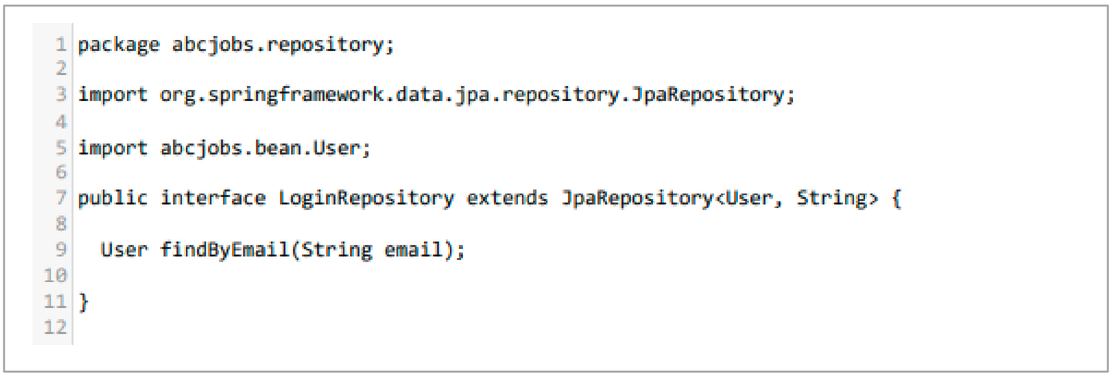


```java
    return lastname;
  }
  public void setLastname(String lastname) {
    this.lastname = lastname;
  }
  public String getEmail() {
    return email;
  }
  public void setEmail(String email) {
    this.email = email;
  }
  public String getPassword() {
    return password;
  }
  public void setPassword(String password) {
    this.password = password;
  }
  public String getConfirmpassword() {
    return confirmpassword;
  }
  public void setConfirmpassword(String confirmpassword) {
    this.confirmpassword = confirmpassword;
  }
  public String getContact() {
    return contact;
  }
  public void setContact(String contact) {
    this.contact = contact;
  }
  public String getEducation() {
    return education;
  }
  public void setEducation(String education) {
    this.education = education;
  }
  public String getCity() {
    return city;
  }
  public void setCity(String city) {
    this.city = city;
  }
  public String getCountry() {
    return country;
  }
  public void setCountry(String country) {
    this.country = country;
  }
  public String getWorkplace() {
    return workplace;
  }
  public void setWorkplace(String workplace) {
    this.workplace = workplace;
  }
  public String getPosition() {
    return position;
  }
  public void setPosition(String position) {
    this.position = position;
  }

```



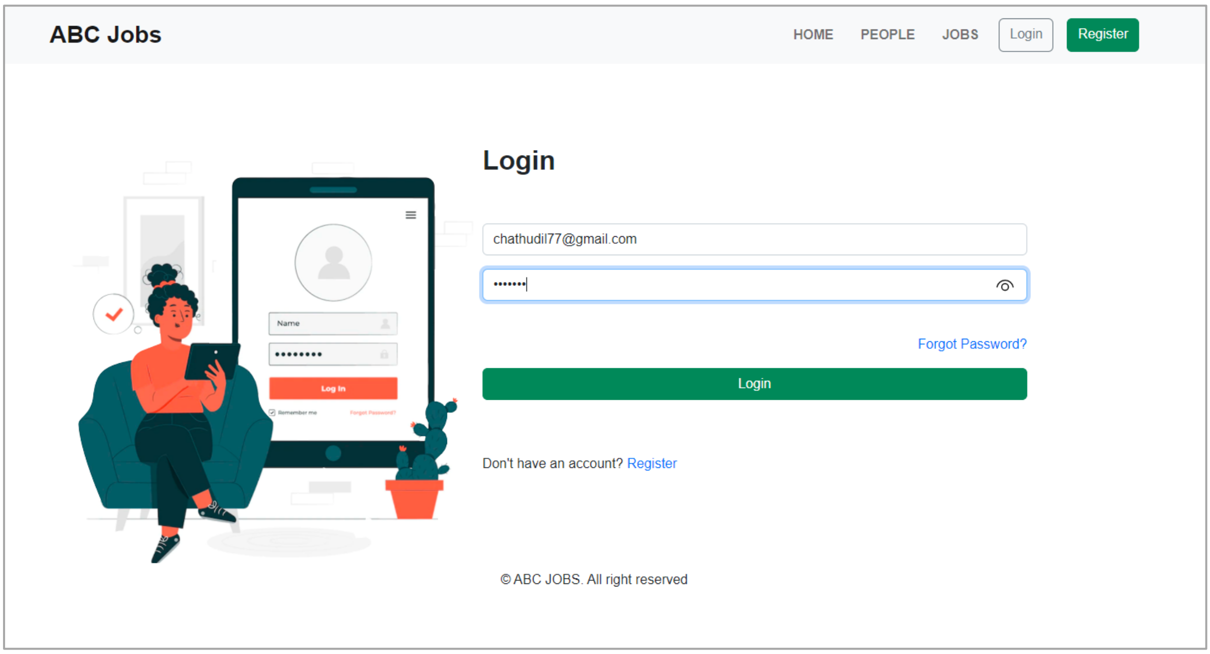
```java
  @Override
  public String toString() {
    return "User [id=" + id + ", firstname=" + firstname + ", lastname=" + lastname + ", email=" + email
        + ", password=" + password + ", confirmpassword=" + confirmpassword + ", contact=" + contact + ", education="
        + education + ", city=" + city + ", country=" + country + ", workplace=" + workplace + ", position="
        + position + "]";
  }


}

```

- **Repo**



```java
package abcjobs.repository;

import org.springframework.data.jpa.repository.JpaRepository;

import abcjobs.bean.User;

public interface LoginRepository extends JpaRepository<User, String> {

  User findByEmail(String email);

}

```

- **Login Page**



Login

chathudil77@gmail.com

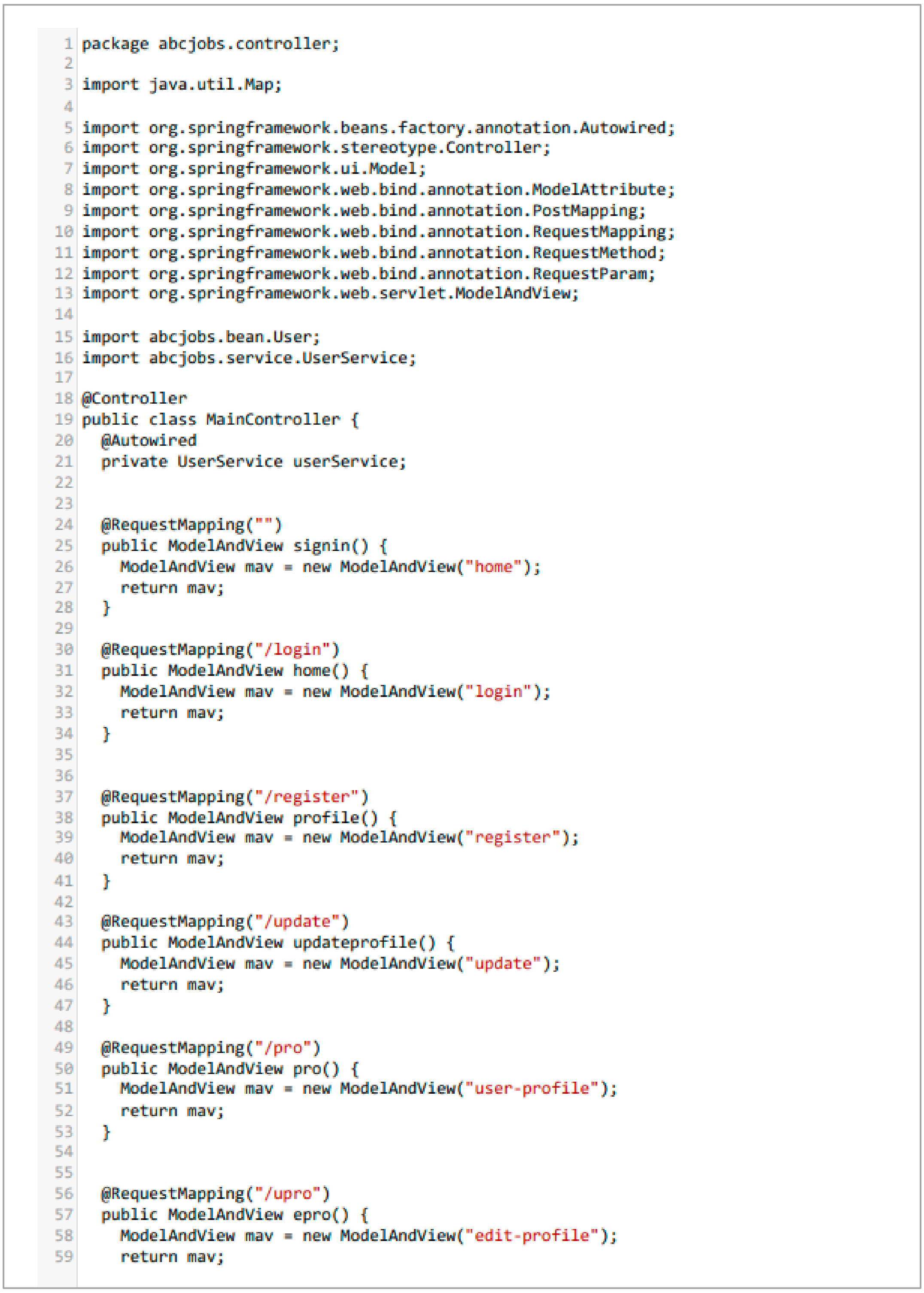
•••••••

Don't have an account? Register

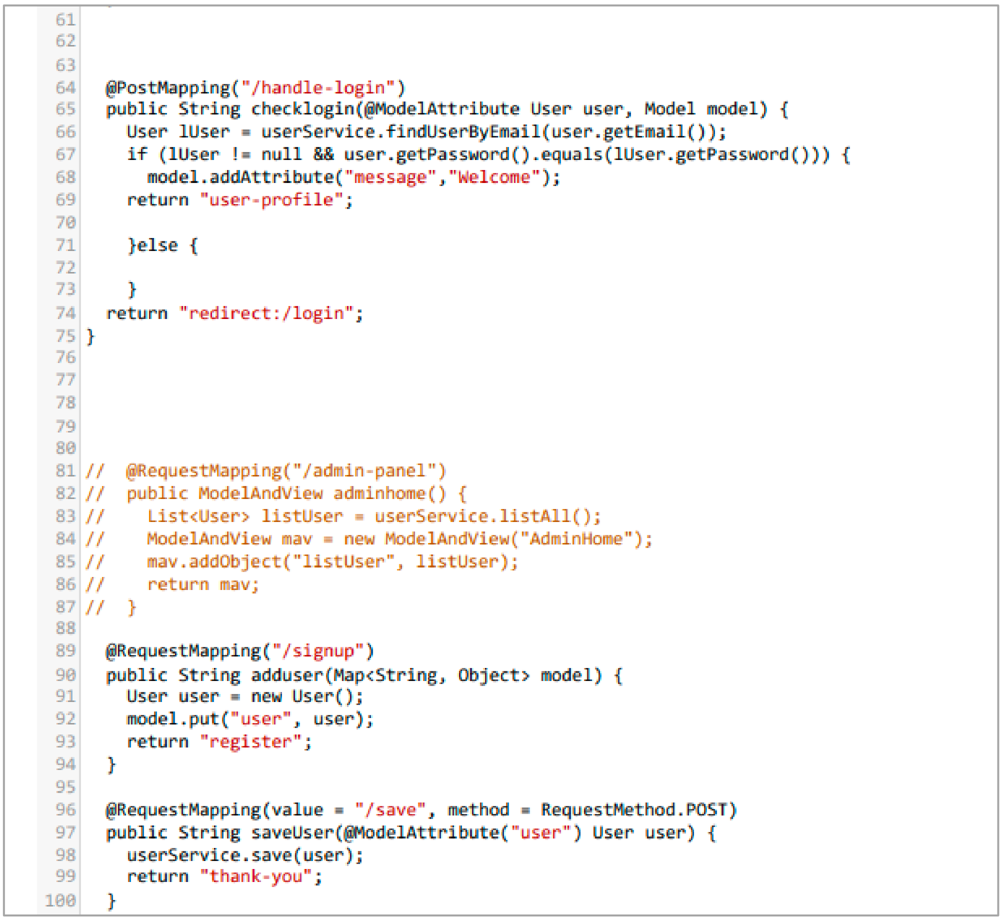
© ABC JOBS. All right reserved

02) Create a Registration Page

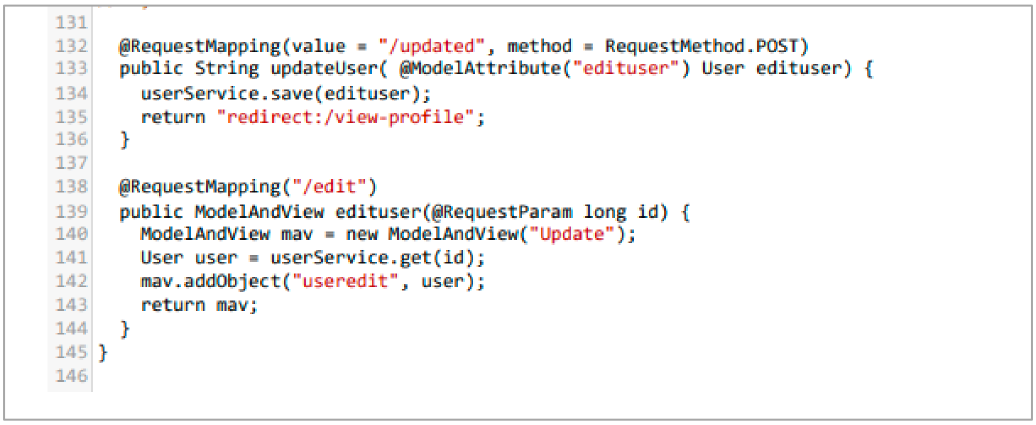
- **Controller**



```java
package abcjobs.controller;

import java.util.Map;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.servlet.ModelAndView;

import abcjobs.bean.User;
import abcjobs.service.UserService;

@Controller
public class MainController {
  @Autowired
  private UserService userService;


  @RequestMapping("")
  public ModelAndView signin() {
    ModelAndView mav = new ModelAndView("home");
    return mav;
  }

  @RequestMapping("/login")
  public ModelAndView home() {
    ModelAndView mav = new ModelAndView("login");
    return mav;
  }


  @RequestMapping("/register")
  public ModelAndView profile() {
    ModelAndView mav = new ModelAndView("register");
    return mav;
  }

  @RequestMapping("/update")
  public ModelAndView updateprofile() {
    ModelAndView mav = new ModelAndView("update");
    return mav;
  }

  @RequestMapping("/pro")
  public ModelAndView pro() {
    ModelAndView mav = new ModelAndView("user-profile");
    return mav;
  }


  @RequestMapping("/upro")
  public ModelAndView epro() {
    ModelAndView mav = new ModelAndView("edit-profile");
    return mav;
```

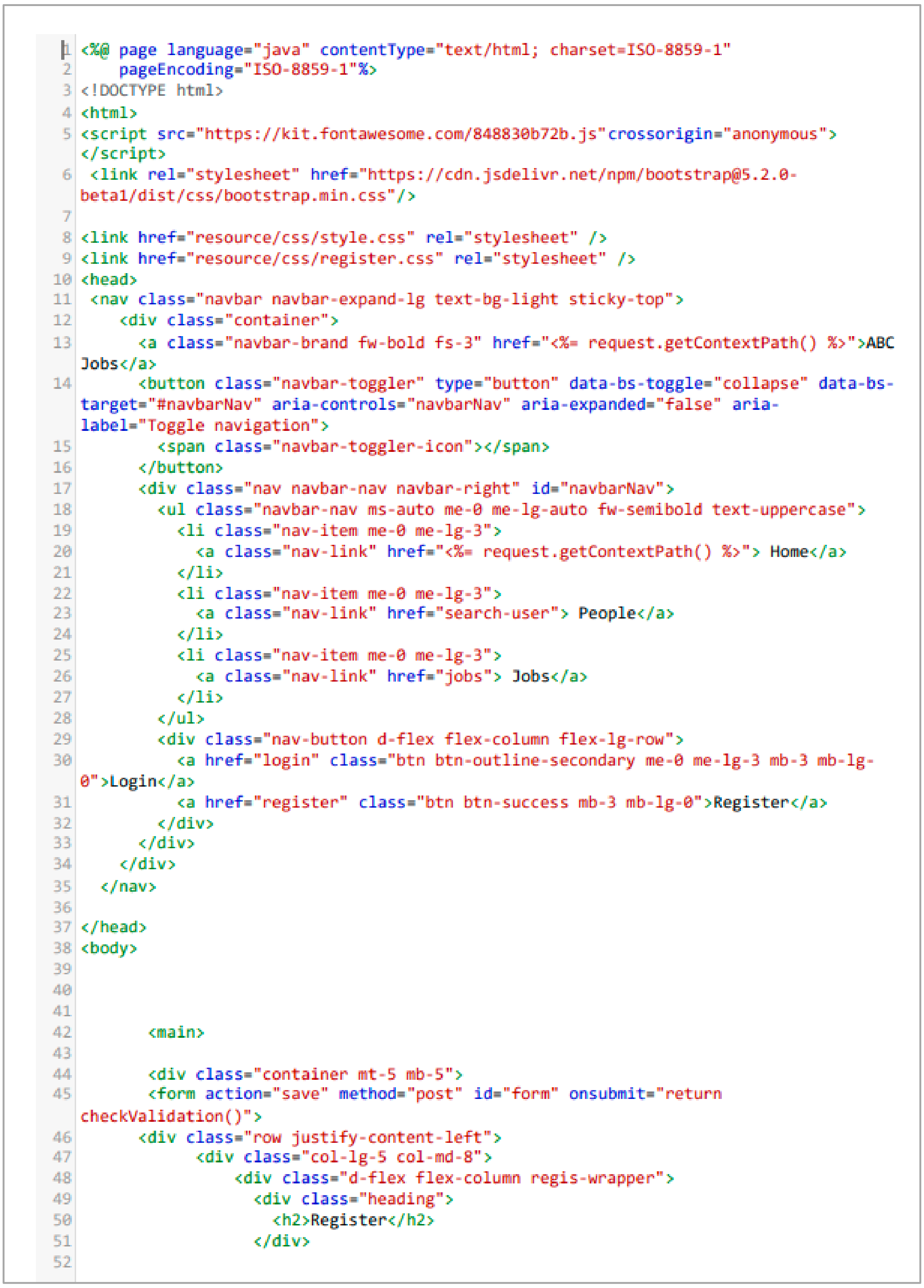


```java
  @PostMapping("/handle-login")
  public String checklogin(@ModelAttribute User user, Model model) {
    User lUser = userService.findUserByEmail(user.getEmail());
    if (lUser != null && user.getPassword().equals(lUser.getPassword())) {
      model.addAttribute("message","Welcome");
    return "user-profile";

    }else {

    }
  return "redirect:/login";
}





//  @RequestMapping("/admin-panel")
//  public ModelAndView adminhome() {
//    List<User> listUser = userService.listAll();
//    ModelAndView mav = new ModelAndView("AdminHome");
//    mav.addObject("listUser", listUser);
//    return mav;
//  }

  @RequestMapping("/signup")
  public String adduser(Map<String, Object> model) {
    User user = new User();
    model.put("user", user);
    return "register";
  }

  @RequestMapping(value = "/save", method = RequestMethod.POST)
  public String saveUser(@ModelAttribute("user") User user) {
    userService.save(user);
    return "thank-you";
  }
```

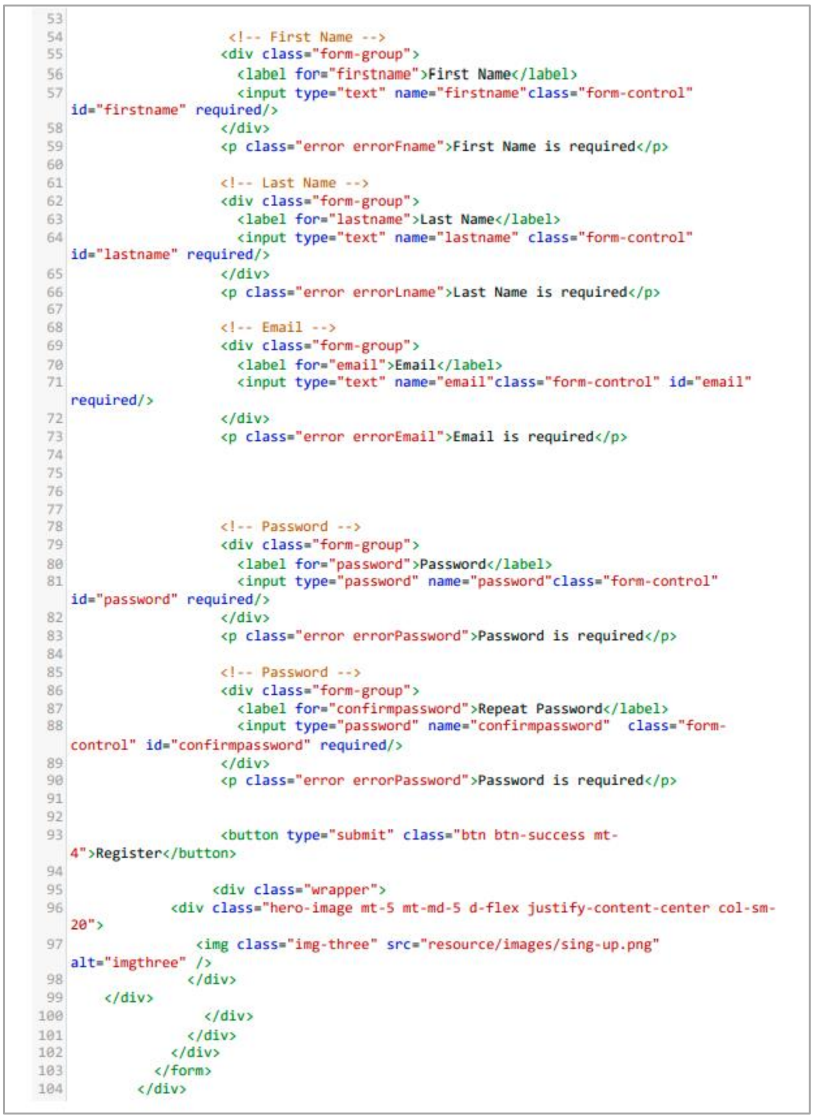


```java

  @RequestMapping(value = "/updated", method = RequestMethod.POST)
  public String updateUser( @ModelAttribute("edituser") User edituser) {
    userService.save(edituser);
    return "redirect:/view-profile";
  }

  @RequestMapping("/edit")
  public ModelAndView edituser(@RequestParam long id) {
    ModelAndView mav = new ModelAndView("Update");
    User user = userService.get(id);
    mav.addObject("useredit", user);
    return mav;
  }
}
```

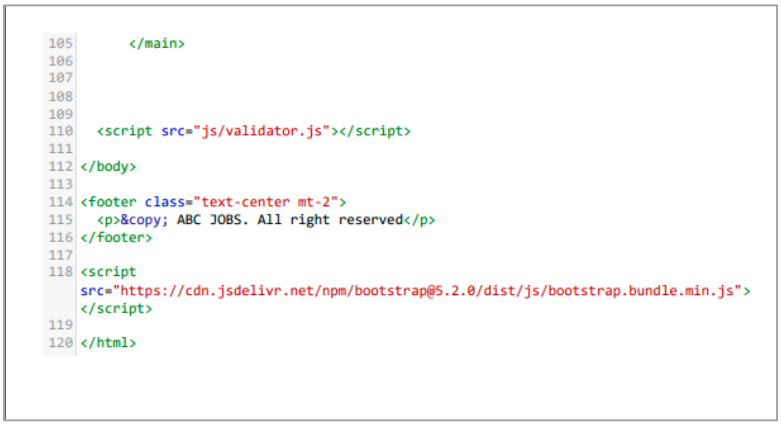
- **JSP**



```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<script src="https://kit.fontawesome.com/848830b72b.js"crossorigin="anonymous"></script>
 <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-beta1/dist/css/bootstrap.min.css"/>

<link href="resource/css/style.css" rel="stylesheet" />
<link href="resource/css/register.css" rel="stylesheet" />
<head>
 <nav class="navbar navbar-expand-lg text-bg-light sticky-top">
    <div class="container">
      <a class="navbar-brand fw-bold fs-3" href="<%= request.getContextPath() %>">ABC Jobs</a>
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="nav navbar-nav navbar-right" id="navbarNav">
        <ul class="navbar-nav ms-auto me-0 me-lg-auto fw-semibold text-uppercase">
          <li class="nav-item me-0 me-lg-3">
            <a class="nav-link" href="<%= request.getContextPath() %>"> Home</a>
          </li>
          <li class="nav-item me-0 me-lg-3">
            <a class="nav-link" href="search-user"> People</a>
          </li>
          <li class="nav-item me-0 me-lg-3">
            <a class="nav-link" href="jobs"> Jobs</a>
          </li>
        </ul>
        <div class="nav-button d-flex flex-column flex-lg-row">
          <a href="login" class="btn btn-outline-secondary me-0 me-lg-3 mb-3 mb-lg-0">Login</a>
          <a href="register" class="btn btn-success mb-3 mb-lg-0">Register</a>
        </div>
      </div>
    </div>
  </nav>

</head>
<body>



       <main>

       <div class="container mt-5 mb-5">
       <form action="save" method="post" id="form" onsubmit="return checkValidation()">
      <div class="row justify-content-left">
            <div class="col-lg-5 col-md-8">
                <div class="d-flex flex-column regis-wrapper">
                  <div class="heading">
                    <h2>Register</h2>
                  </div>

```

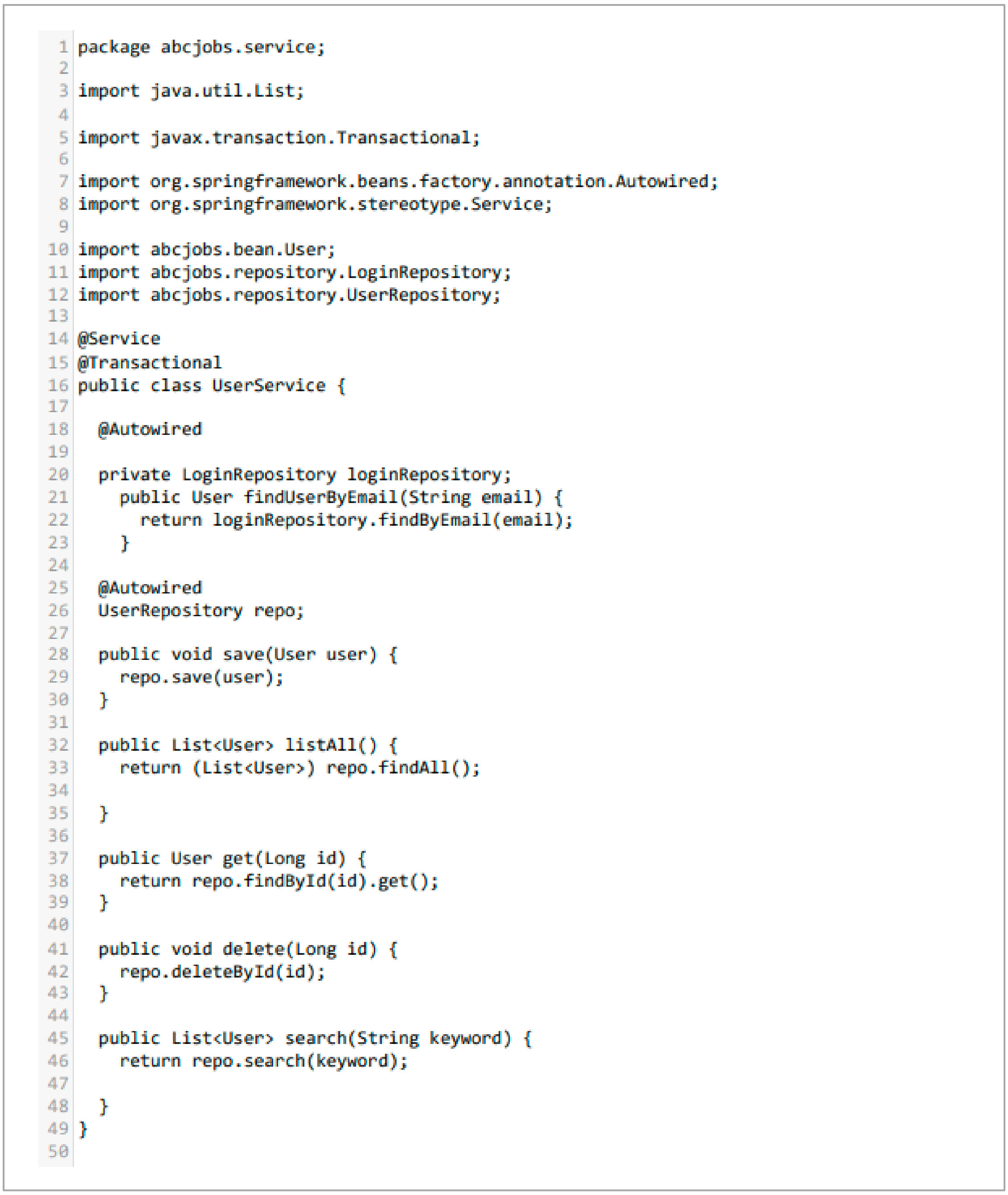


```html

                   <!-- First Name -->
                  <div class="form-group">
                    <label for="firstname">First Name</label>
                    <input type="text" name="firstname"class="form-control"
id="firstname" required/>
                  </div>
                  <p class="error errorFname">First Name is required</p>

                  <!-- Last Name -->
                  <div class="form-group">
                    <label for="lastname">Last Name</label>
                    <input type="text" name="lastname" class="form-control"
id="lastname" required/>
                  </div>
                  <p class="error errorLname">Last Name is required</p>

                  <!-- Email -->
                  <div class="form-group">
                    <label for="email">Email</label>
                    <input type="text" name="email"class="form-control" id="email"
required/>
                  </div>
                  <p class="error errorEmail">Email is required</p>




                  <!-- Password -->
                  <div class="form-group">
                    <label for="password">Password</label>
                    <input type="password" name="password"class="form-control"
id="password" required/>
                  </div>
                  <p class="error errorPassword">Password is required</p>

                  <!-- Password -->
                  <div class="form-group">
                    <label for="confirmpassword">Repeat Password</label>
                    <input type="password" name="confirmpassword"  class="form-
control" id="confirmpassword" required/>
                  </div>
                  <p class="error errorPassword">Password is required</p>


                  <button type="submit" class="btn btn-success mt-
4">Register</button>

                 <div class="wrapper">
            <div class="hero-image mt-5 mt-md-5 d-flex justify-content-center col-sm-
20">
               <img class="img-three" src="resource/images/sing-up.png"
alt="imgthree" />
              </div>
    </div>
                </div>
              </div>
            </div>
          </form>
        </div>
```



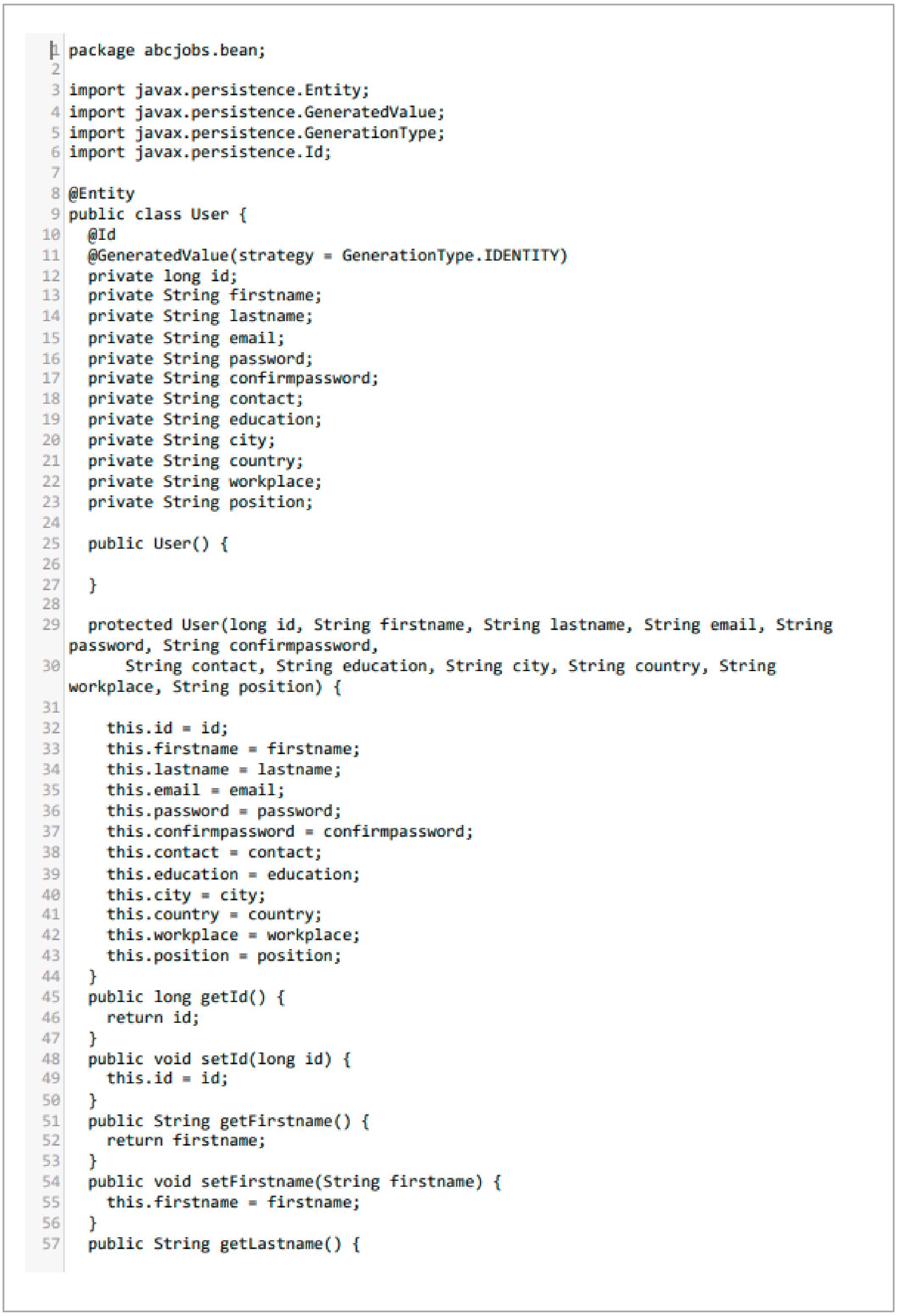
```html
      </main>




  <script src="js/validator.js"></script>

</body>

<footer class="text-center mt-2">
  <p>&copy; ABC JOBS. All right reserved</p>
</footer>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0/dist/js/bootstrap.bundle.min.js">
</script>

</html>
```

- **Service**

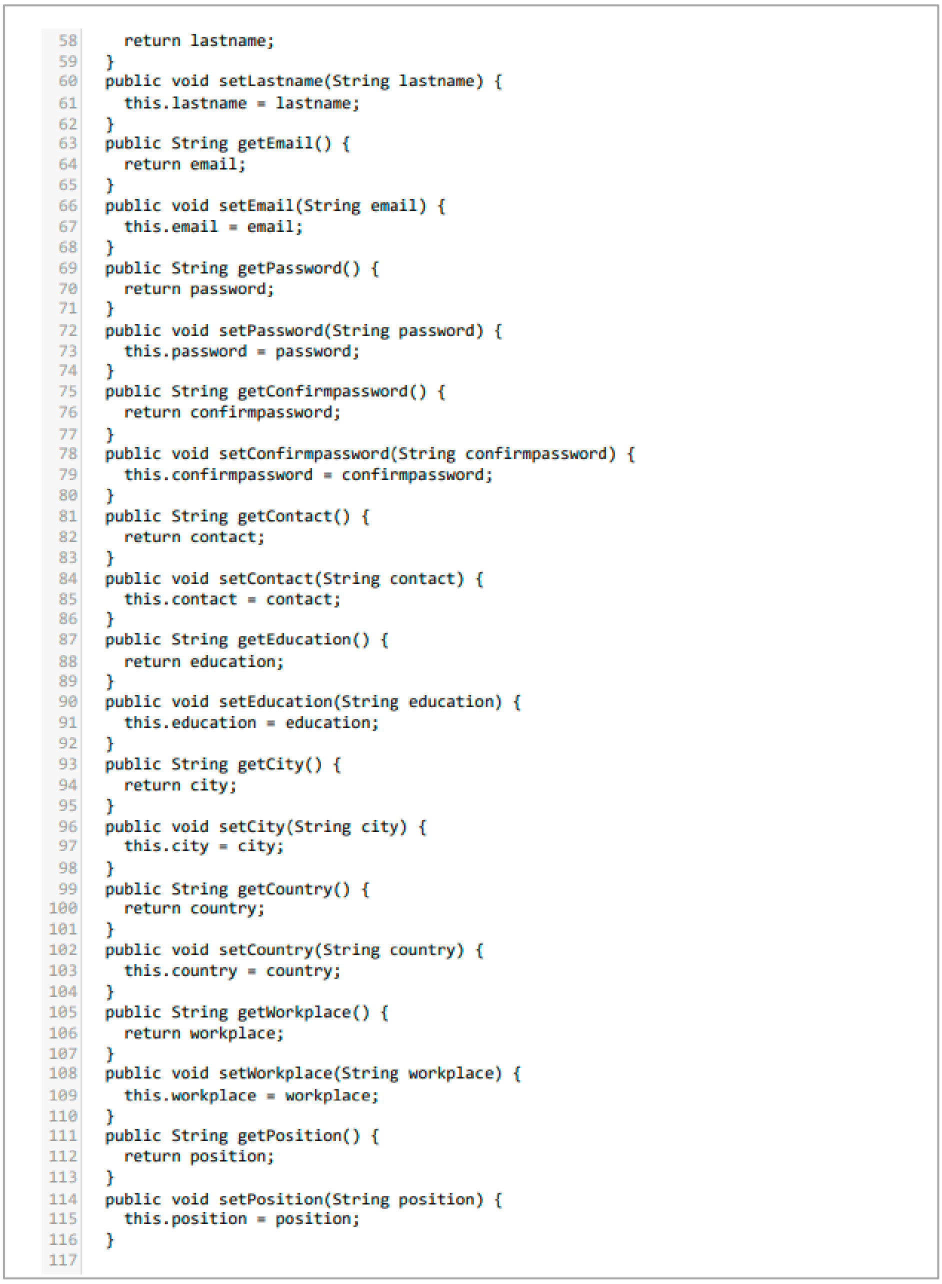


```java
package abcjobs.service;

import java.util.List;

import javax.transaction.Transactional;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import abcjobs.bean.User;
import abcjobs.repository.LoginRepository;
import abcjobs.repository.UserRepository;

@Service
@Transactional
public class UserService {

  @Autowired

  private LoginRepository loginRepository;
    public User findUserByEmail(String email) {
      return loginRepository.findByEmail(email);
    }

  @Autowired
  UserRepository repo;

  public void save(User user) {
    repo.save(user);
  }

  public List<User> listAll() {
    return (List<User>) repo.findAll();

  }

  public User get(Long id) {
    return repo.findById(id).get();
  }

  public void delete(Long id) {
    repo.deleteById(id);
  }

  public List<User> search(String keyword) {
    return repo.search(keyword);

  }
}

```

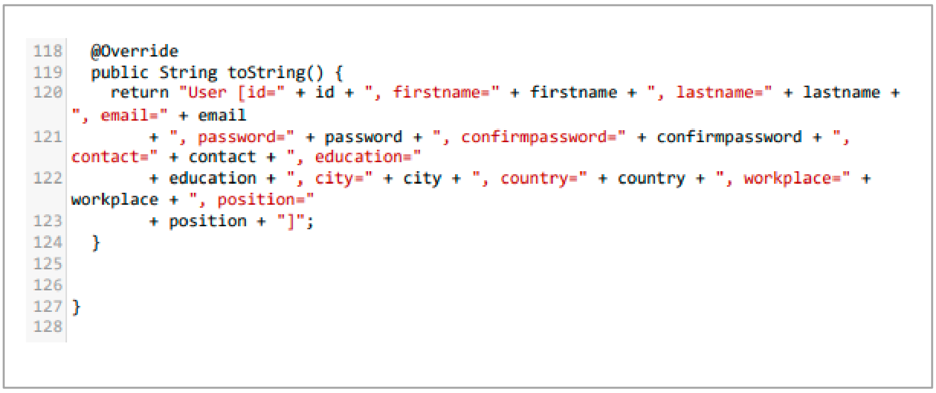
- **Bean**



```java
package abcjobs.bean;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;

@Entity
public class User {
  @Id
  @GeneratedValue(strategy = GenerationType.IDENTITY)
  private long id;
  private String firstname;
  private String lastname;
  private String email;
  private String password;
  private String confirmpassword;
  private String contact;
  private String education;
  private String city;
  private String country;
  private String workplace;
  private String position;

  public User() {

  }

  protected User(long id, String firstname, String lastname, String email, String password, String confirmpassword,
        String contact, String education, String city, String country, String workplace, String position) {

    this.id = id;
    this.firstname = firstname;
    this.lastname = lastname;
    this.email = email;
    this.password = password;
    this.confirmpassword = confirmpassword;
    this.contact = contact;
    this.education = education;
    this.city = city;
    this.country = country;
    this.workplace = workplace;
    this.position = position;
  }
  public long getId() {
    return id;
  }
  public void setId(long id) {
    this.id = id;
  }
  public String getFirstname() {
    return firstname;
  }
  public void setFirstname(String firstname) {
    this.firstname = firstname;
  }
  public String getLastname() {
```

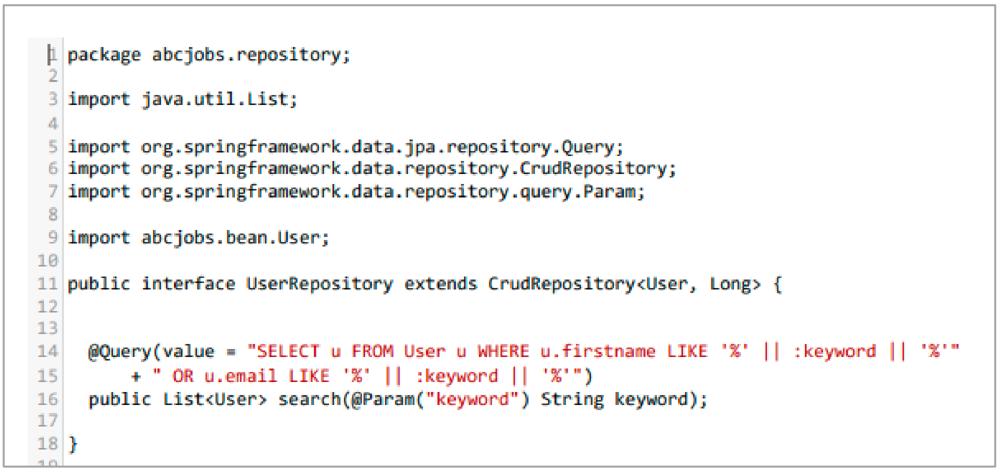


```
    return lastname;
  }
  public void setLastname(String lastname) {
    this.lastname = lastname;
  }
  public String getEmail() {
    return email;
  }
  public void setEmail(String email) {
    this.email = email;
  }
  public String getPassword() {
    return password;
  }
  public void setPassword(String password) {
    this.password = password;
  }
  public String getConfirmpassword() {
    return confirmpassword;
  }
  public void setConfirmpassword(String confirmpassword) {
    this.confirmpassword = confirmpassword;
  }
  public String getContact() {
    return contact;
  }
  public void setContact(String contact) {
    this.contact = contact;
  }
  public String getEducation() {
    return education;
  }
  public void setEducation(String education) {
    this.education = education;
  }
  public String getCity() {
    return city;
  }
  public void setCity(String city) {
    this.city = city;
  }
  public String getCountry() {
    return country;
  }
  public void setCountry(String country) {
    this.country = country;
  }
  public String getWorkplace() {
    return workplace;
  }
  public void setWorkplace(String workplace) {
    this.workplace = workplace;
  }
  public String getPosition() {
    return position;
  }
  public void setPosition(String position) {
    this.position = position;
  }

```



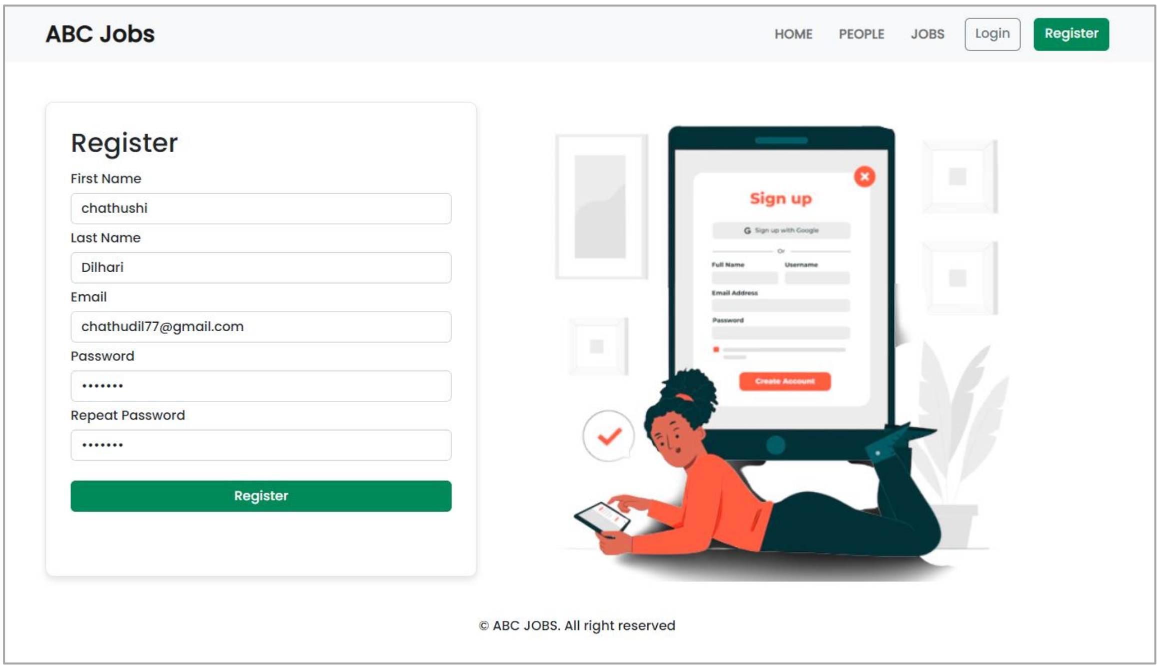
```
  @Override
  public String toString() {
    return "User [id=" + id + ", firstname=" + firstname + ", lastname=" + lastname + ", email=" + email
        + ", password=" + password + ", confirmpassword=" + confirmpassword + ", contact=" + contact + ", education="
        + education + ", city=" + city + ", country=" + country + ", workplace=" + workplace + ", position="
        + position + "]";
  }


}

```

- **Repo**



```
package abcjobs.repository;

import java.util.List;

import org.springframework.data.jpa.repository.Query;
import org.springframework.data.repository.CrudRepository;
import org.springframework.data.repository.query.Param;

import abcjobs.bean.User;

public interface UserRepository extends CrudRepository<User, Long> {


  @Query(value = "SELECT u FROM User u WHERE u.firstname LIKE '%' || :keyword || '%'"
      + " OR u.email LIKE '%' || :keyword || '%'")
  public List<User> search(@Param("keyword") String keyword);

}
```

- **Registration Page**



Register

First Name: chathushi

Last Name: Dilhari

Email: chathudil77@gmail.com

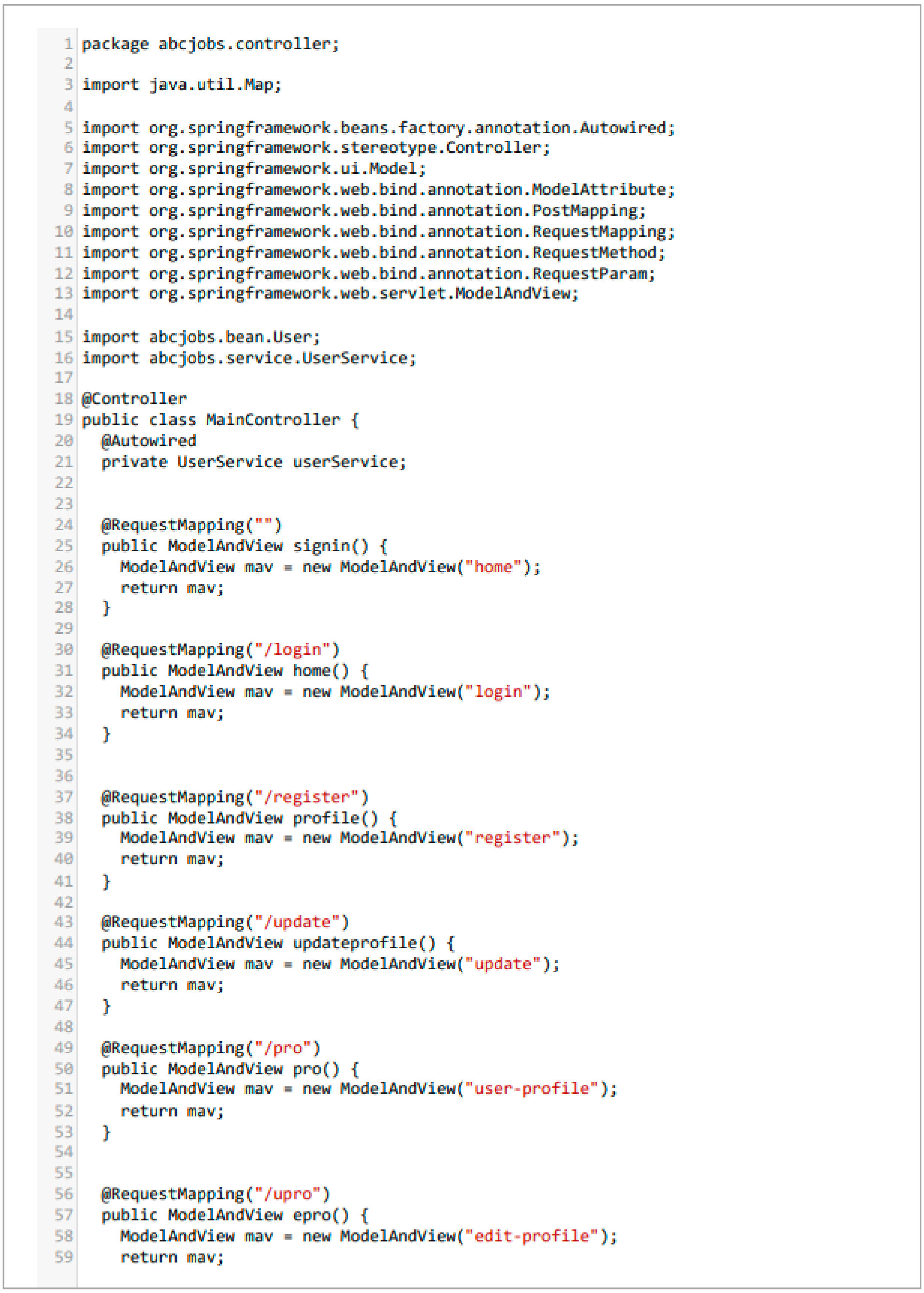
Password: •••••••

Repeat Password: •••••••

© ABC JOBS. All right reserved

03)Create an Update Profile Page

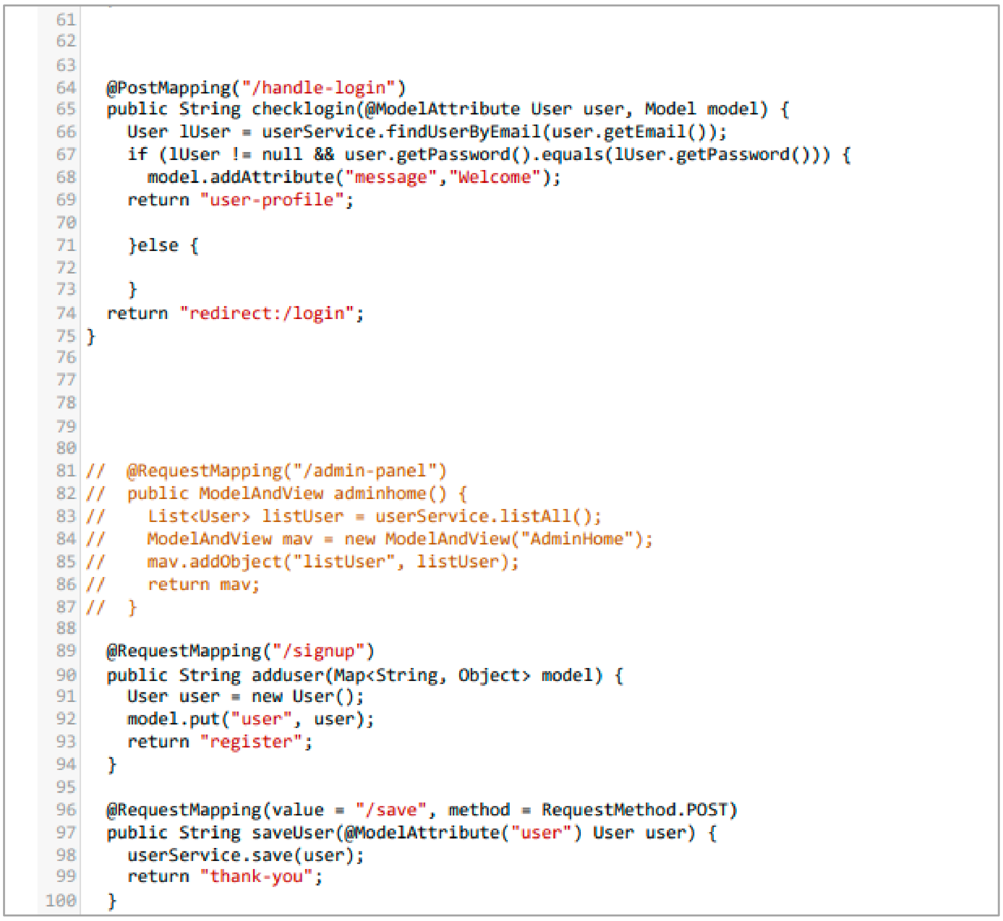
- **Controller**



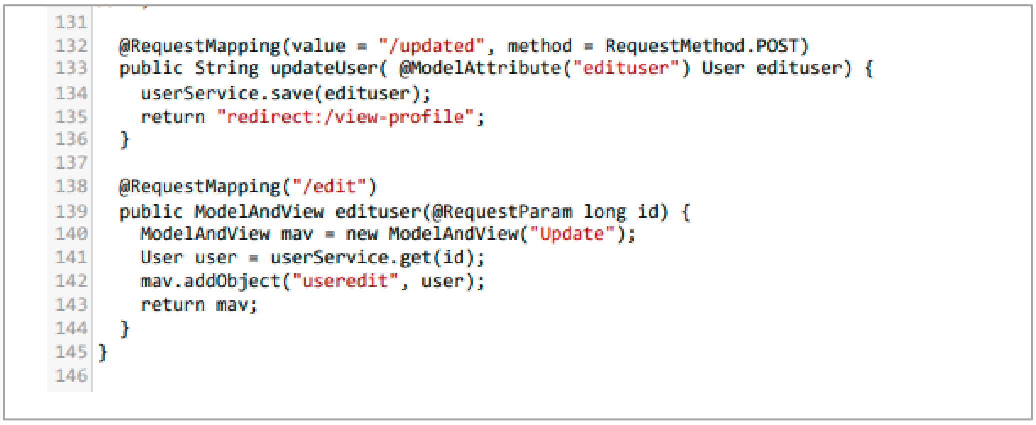
```
package abcjobs.controller;

import java.util.Map;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.servlet.ModelAndView;

import abcjobs.bean.User;
import abcjobs.service.UserService;

@Controller
public class MainController {
  @Autowired
  private UserService userService;


  @RequestMapping("")
  public ModelAndView signin() {
    ModelAndView mav = new ModelAndView("home");
    return mav;
  }

  @RequestMapping("/login")
  public ModelAndView home() {
    ModelAndView mav = new ModelAndView("login");
    return mav;
  }


  @RequestMapping("/register")
  public ModelAndView profile() {
    ModelAndView mav = new ModelAndView("register");
    return mav;
  }

  @RequestMapping("/update")
  public ModelAndView updateprofile() {
    ModelAndView mav = new ModelAndView("update");
    return mav;
  }

  @RequestMapping("/pro")
  public ModelAndView pro() {
    ModelAndView mav = new ModelAndView("user-profile");
    return mav;
  }


  @RequestMapping("/upro")
  public ModelAndView epro() {
    ModelAndView mav = new ModelAndView("edit-profile");
    return mav;
```

-

-

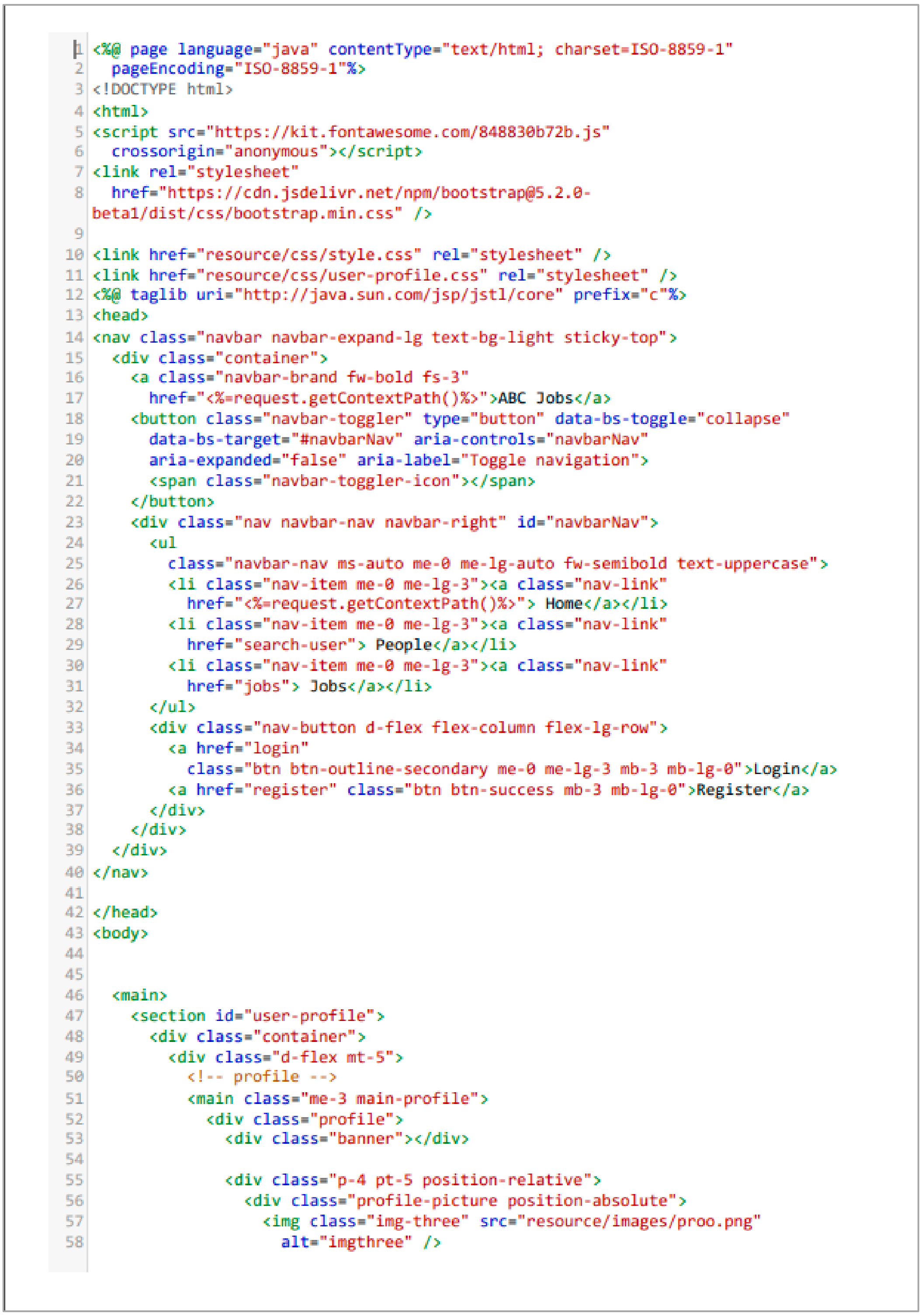


```java



  @PostMapping("/handle-login")
  public String checklogin(@ModelAttribute User user, Model model) {
    User lUser = userService.findUserByEmail(user.getEmail());
    if (lUser != null && user.getPassword().equals(lUser.getPassword())) {
      model.addAttribute("message","Welcome");
    return "user-profile";

    }else {

    }
  return "redirect:/login";
}





//  @RequestMapping("/admin-panel")
//  public ModelAndView adminhome() {
//    List<User> listUser = userService.listAll();
//    ModelAndView mav = new ModelAndView("AdminHome");
//    mav.addObject("listUser", listUser);
//    return mav;
//  }

  @RequestMapping("/signup")
  public String adduser(Map<String, Object> model) {
    User user = new User();
    model.put("user", user);
    return "register";
  }

  @RequestMapping(value = "/save", method = RequestMethod.POST)
  public String saveUser(@ModelAttribute("user") User user) {
    userService.save(user);
    return "thank-you";
  }
```

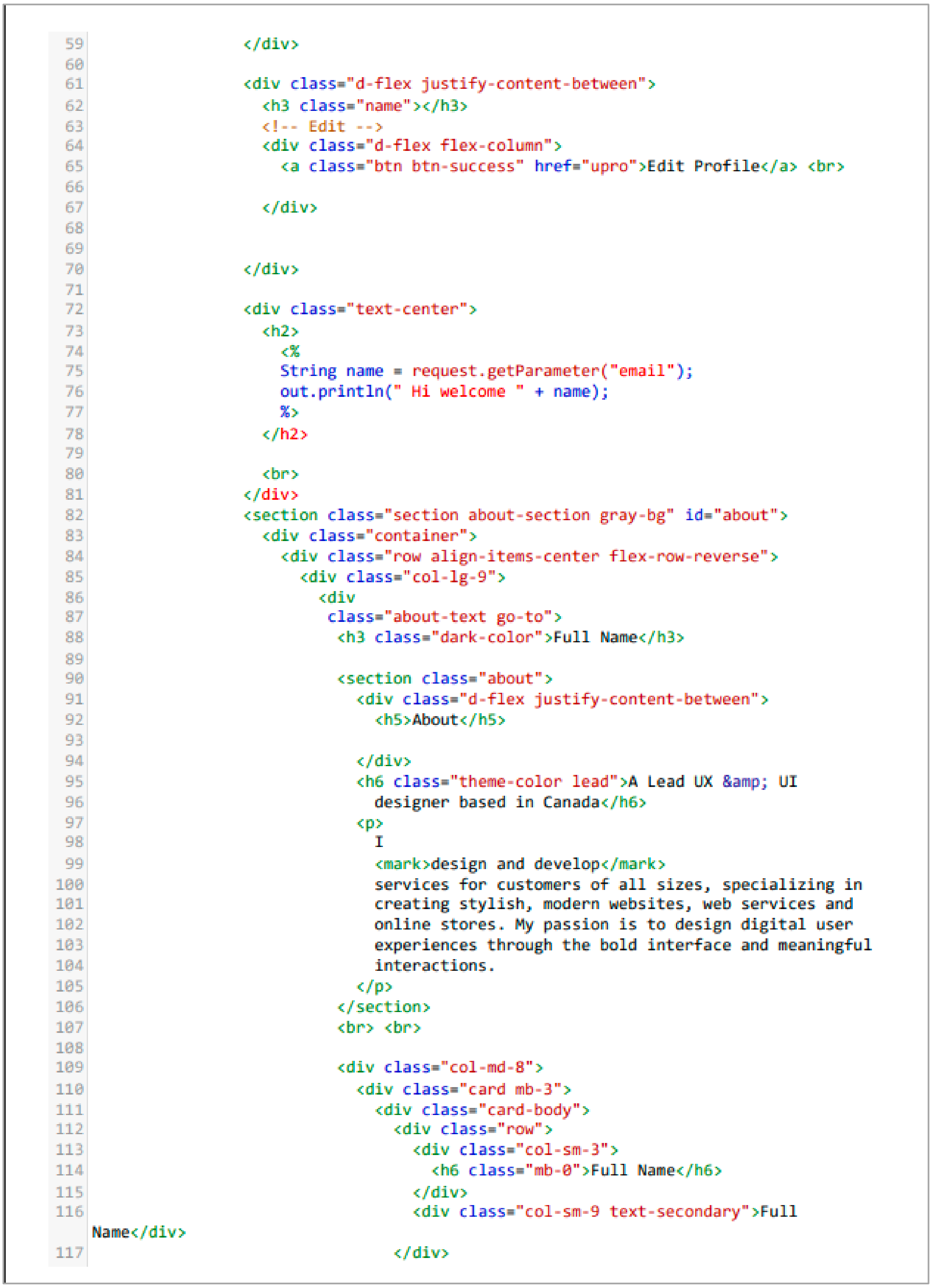


```java

  @RequestMapping(value = "/updated", method = RequestMethod.POST)
  public String updateUser( @ModelAttribute("edituser") User edituser) {
    userService.save(edituser);
    return "redirect:/view-profile";
  }

  @RequestMapping("/edit")
  public ModelAndView edituser(@RequestParam long id) {
    ModelAndView mav = new ModelAndView("Update");
    User user = userService.get(id);
    mav.addObject("useredit", user);
    return mav;
  }
}

```

- **JSP**

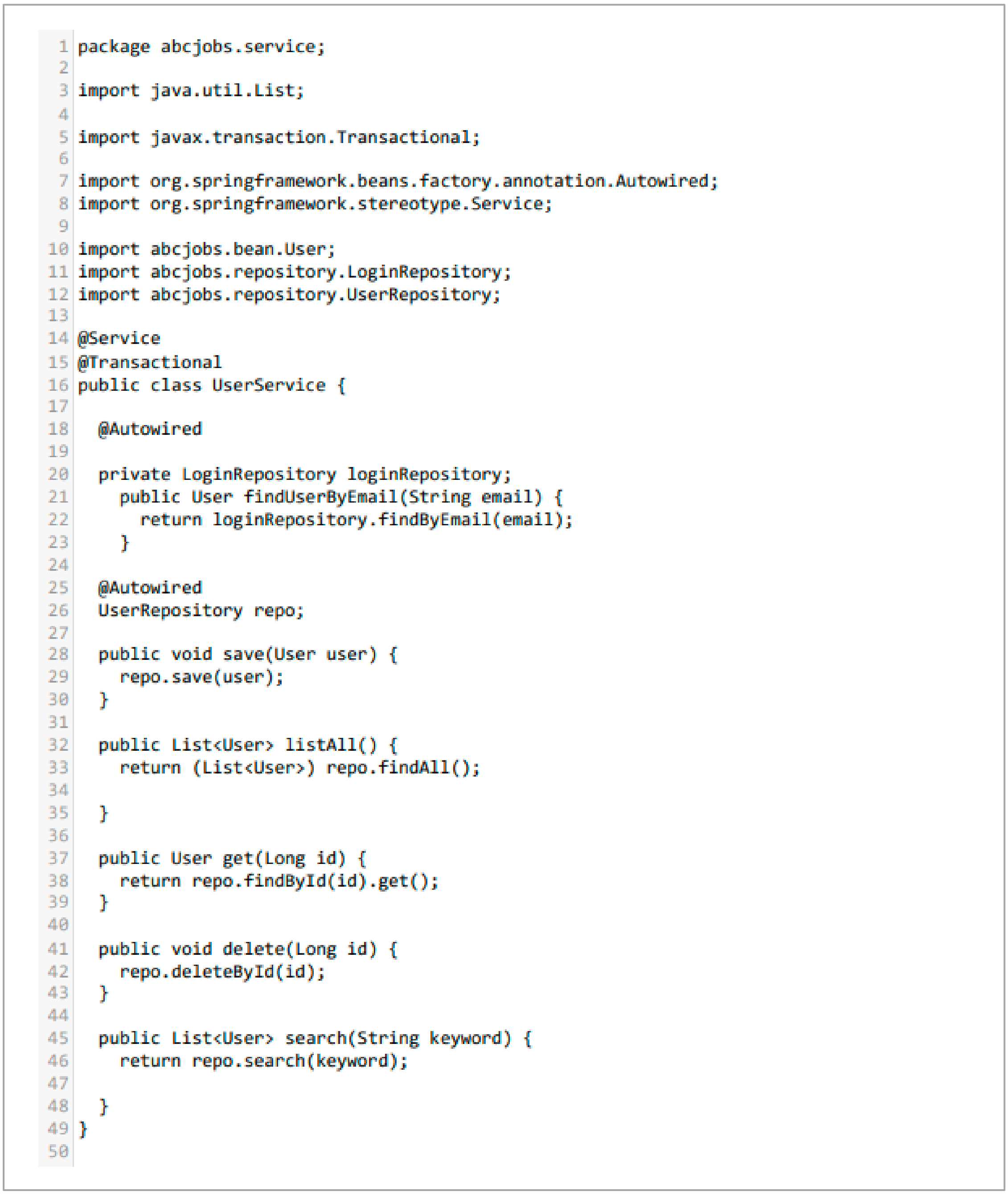


```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
  pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<script src="https://kit.fontawesome.com/848830b72b.js"
  crossorigin="anonymous"></script>
<link rel="stylesheet"
  href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-
beta1/dist/css/bootstrap.min.css" />

<link href="resource/css/style.css" rel="stylesheet" />
<link href="resource/css/user-profile.css" rel="stylesheet" />
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<head>
<nav class="navbar navbar-expand-lg text-bg-light sticky-top">
  <div class="container">
    <a class="navbar-brand fw-bold fs-3"
      href="<%=request.getContextPath()%>">ABC Jobs</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
      data-bs-target="#navbarNav" aria-controls="navbarNav"
      aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="nav navbar-nav navbar-right" id="navbarNav">
      <ul
        class="navbar-nav ms-auto me-0 me-lg-auto fw-semibold text-uppercase">
        <li class="nav-item me-0 me-lg-3"><a class="nav-link"
          href="<%=request.getContextPath()%>"> Home</a></li>
        <li class="nav-item me-0 me-lg-3"><a class="nav-link"
          href="search-user"> People</a></li>
        <li class="nav-item me-0 me-lg-3"><a class="nav-link"
          href="jobs"> Jobs</a></li>
      </ul>
      <div class="nav-button d-flex flex-column flex-lg-row">
        <a href="login"
          class="btn btn-outline-secondary me-0 me-lg-3 mb-3 mb-lg-0">Login</a>
        <a href="register" class="btn btn-success mb-3 mb-lg-0">Register</a>
      </div>
    </div>
  </div>
</nav>

</head>
<body>


  <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">
                  <img class="img-three" src="resource/images/proo.png"
                    alt="imgthree" />
```



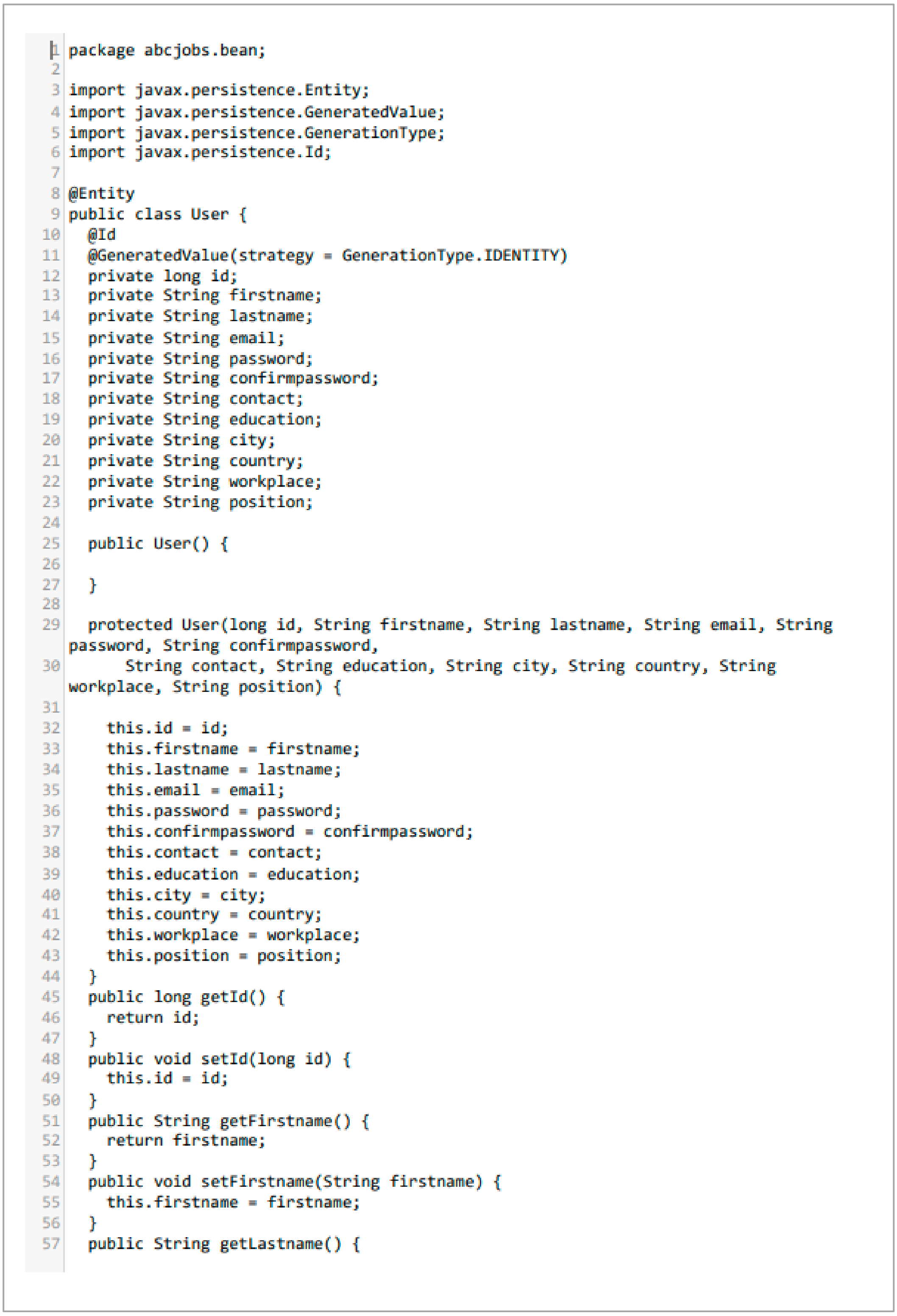
```
                </div>

                <div class="d-flex justify-content-between">
                  <h3 class="name"></h3>
                  <!-- Edit -->
                  <div class="d-flex flex-column">
                    <a class="btn btn-success" href="upro">Edit Profile</a> <br>

                  </div>


                </div>

                <div class="text-center">
                  <h2>
                    <%
                    String name = request.getParameter("email");
                    out.println(" Hi welcome " + name);
                    %>
                  </h2>

                  <br>
                </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">
                          <h3 class="dark-color">Full Name</h3>

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

                            </div>
                            <h6 class="theme-color lead">A Lead UX &amp; 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>
                          </section>
                          <br> <br>

                          <div class="col-md-8">
                            <div class="card mb-3">
                              <div class="card-body">
                                <div class="row">
                                  <div class="col-sm-3">
                                    <h6 class="mb-0">Full Name</h6>
                                  </div>
                                  <div class="col-sm-9 text-secondary">Full
Name</div>
                                </div>
```

- ## Service

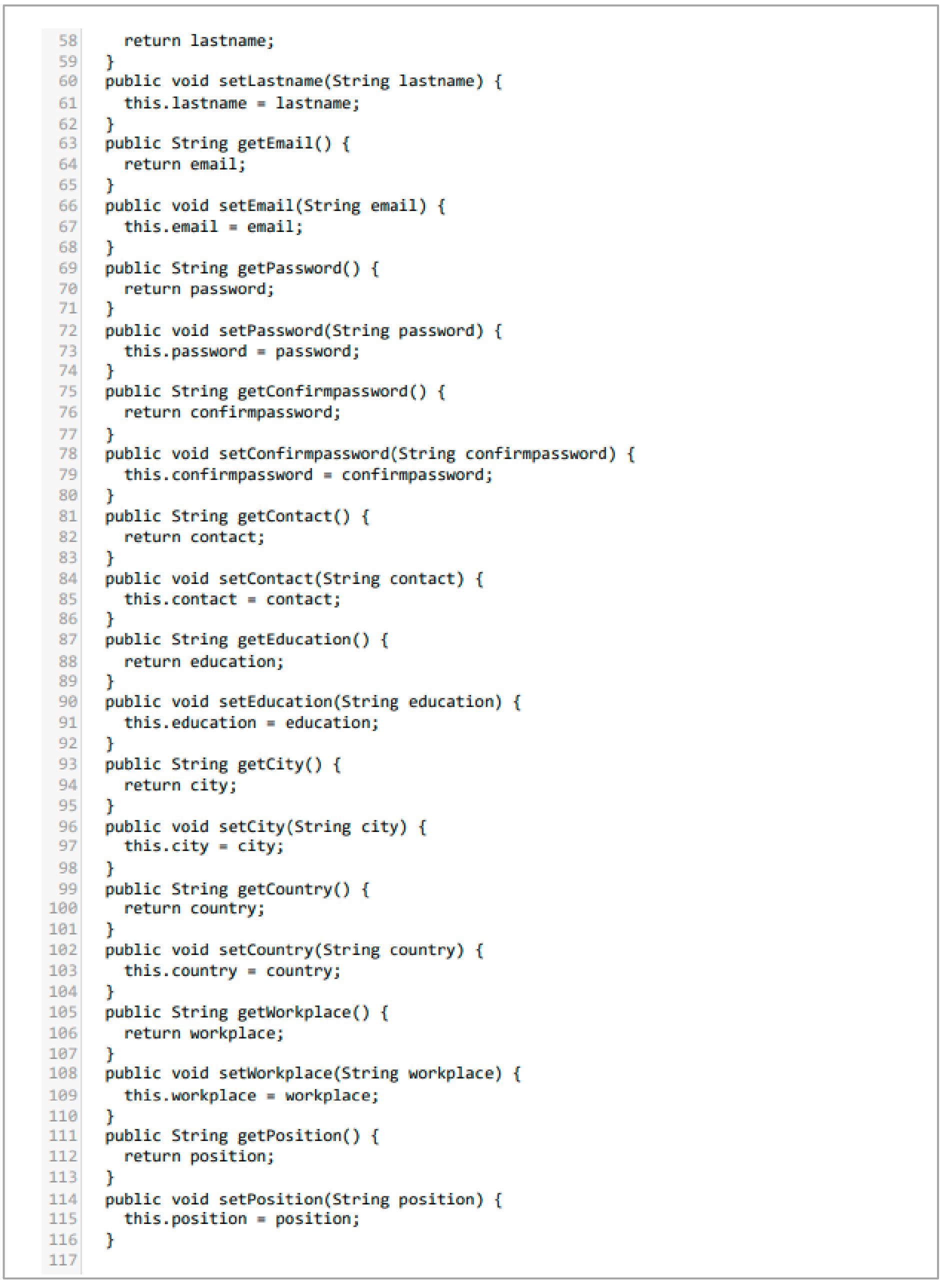


```java
package abcjobs.service;

import java.util.List;

import javax.transaction.Transactional;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import abcjobs.bean.User;
import abcjobs.repository.LoginRepository;
import abcjobs.repository.UserRepository;

@Service
@Transactional
public class UserService {

  @Autowired

  private LoginRepository loginRepository;
    public User findUserByEmail(String email) {
      return loginRepository.findByEmail(email);
    }

  @Autowired
  UserRepository repo;

  public void save(User user) {
    repo.save(user);
  }

  public List<User> listAll() {
    return (List<User>) repo.findAll();

  }

  public User get(Long id) {
    return repo.findById(id).get();
  }

  public void delete(Long id) {
    repo.deleteById(id);
  }

  public List<User> search(String keyword) {
    return repo.search(keyword);

  }
}

```

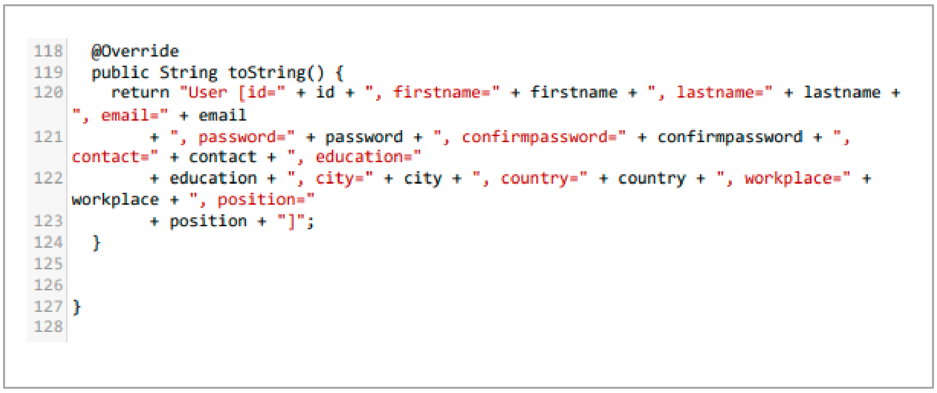
- **Bean**



```java
package abcjobs.bean;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;

@Entity
public class User {
  @Id
  @GeneratedValue(strategy = GenerationType.IDENTITY)
  private long id;
  private String firstname;
  private String lastname;
  private String email;
  private String password;
  private String confirmpassword;
  private String contact;
  private String education;
  private String city;
  private String country;
  private String workplace;
  private String position;

  public User() {

  }

  protected User(long id, String firstname, String lastname, String email, String password, String confirmpassword,
      String contact, String education, String city, String country, String workplace, String position) {

    this.id = id;
    this.firstname = firstname;
    this.lastname = lastname;
    this.email = email;
    this.password = password;
    this.confirmpassword = confirmpassword;
    this.contact = contact;
    this.education = education;
    this.city = city;
    this.country = country;
    this.workplace = workplace;
    this.position = position;
  }
  public long getId() {
    return id;
  }
  public void setId(long id) {
    this.id = id;
  }
  public String getFirstname() {
    return firstname;
  }
  public void setFirstname(String firstname) {
    this.firstname = firstname;
  }
  public String getLastname() {
```

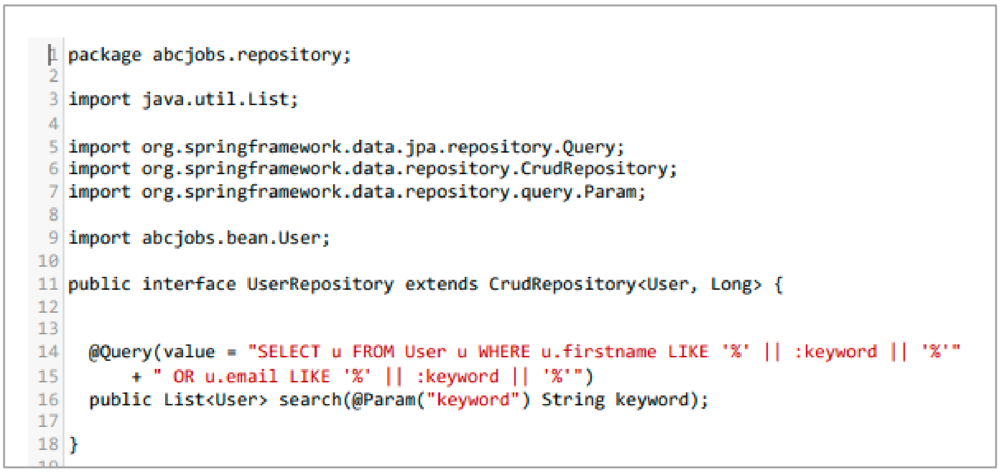


```java
    return lastname;
  }
  public void setLastname(String lastname) {
    this.lastname = lastname;
  }
  public String getEmail() {
    return email;
  }
  public void setEmail(String email) {
    this.email = email;
  }
  public String getPassword() {
    return password;
  }
  public void setPassword(String password) {
    this.password = password;
  }
  public String getConfirmpassword() {
    return confirmpassword;
  }
  public void setConfirmpassword(String confirmpassword) {
    this.confirmpassword = confirmpassword;
  }
  public String getContact() {
    return contact;
  }
  public void setContact(String contact) {
    this.contact = contact;
  }
  public String getEducation() {
    return education;
  }
  public void setEducation(String education) {
    this.education = education;
  }
  public String getCity() {
    return city;
  }
  public void setCity(String city) {
    this.city = city;
  }
  public String getCountry() {
    return country;
  }
  public void setCountry(String country) {
    this.country = country;
  }
  public String getWorkplace() {
    return workplace;
  }
  public void setWorkplace(String workplace) {
    this.workplace = workplace;
  }
  public String getPosition() {
    return position;
  }
  public void setPosition(String position) {
    this.position = position;
  }

```



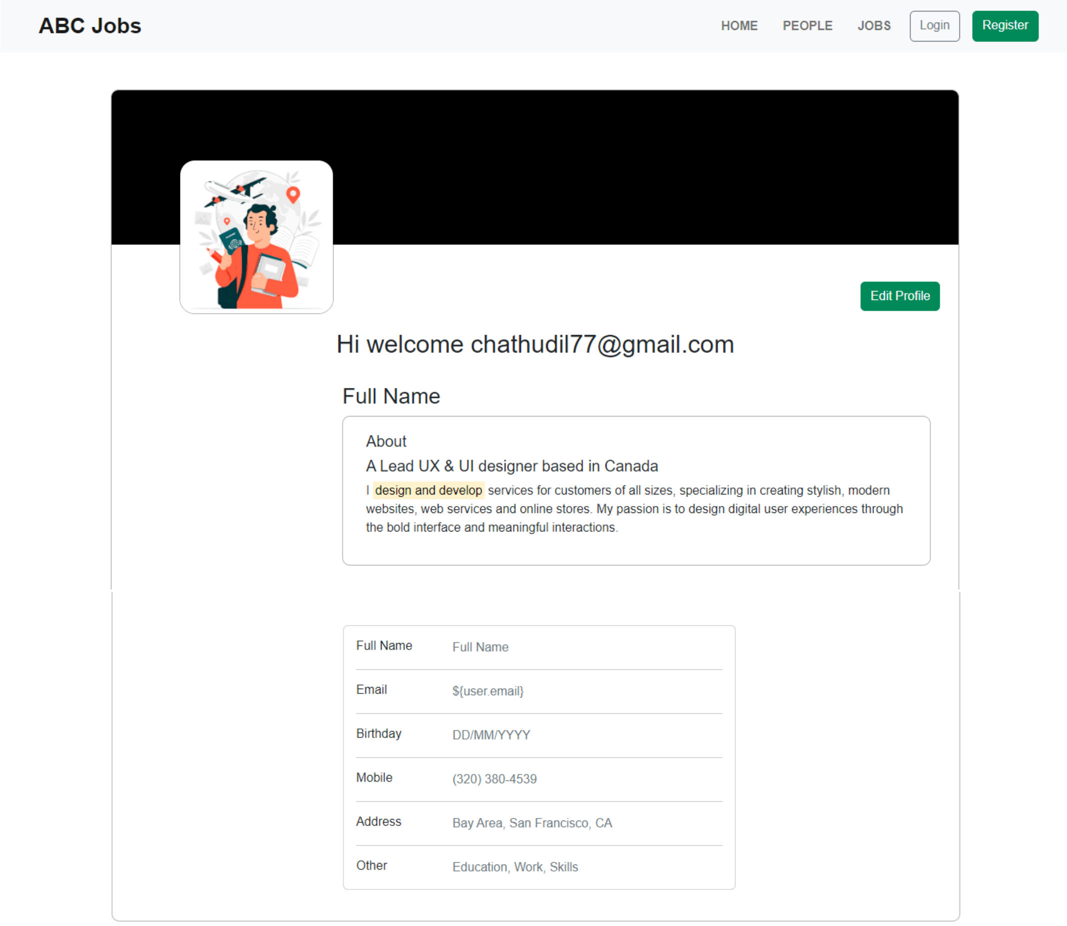
```java
  @Override
  public String toString() {
    return "User [id=" + id + ", firstname=" + firstname + ", lastname=" + lastname + ", email=" + email
        + ", password=" + password + ", confirmpassword=" + confirmpassword + ", contact=" + contact + ", education="
        + education + ", city=" + city + ", country=" + country + ", workplace=" + workplace + ", position="
        + position + "]";
  }


}

```

Repo



```
package abcjobs.repository;

import java.util.List;

import org.springframework.data.jpa.repository.Query;
import org.springframework.data.repository.CrudRepository;
import org.springframework.data.repository.query.Param;

import abcjobs.bean.User;

public interface UserRepository extends CrudRepository<User, Long> {


  @Query(value = "SELECT u FROM User u WHERE u.firstname LIKE '%' || :keyword || '%'"
      + " OR u.email LIKE '%' || :keyword || '%'")
  public List<User> search(@Param("keyword") String keyword);

}
```

User Profile



Hi welcome chathudil77@gmail.com

Full Name

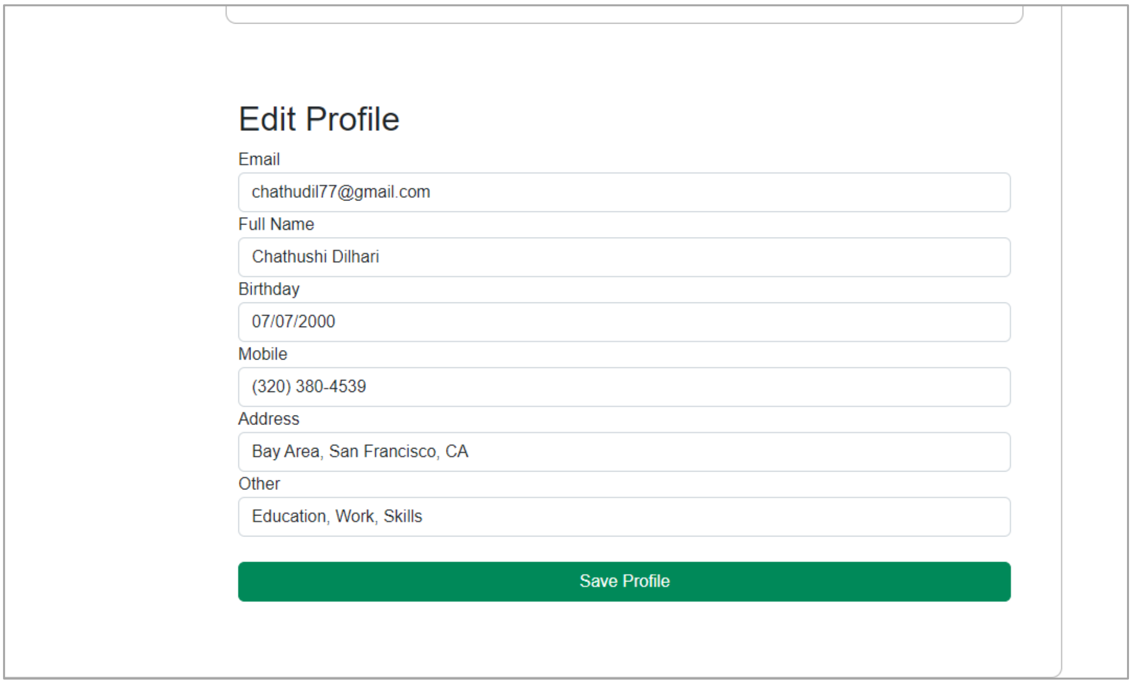
About

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.

Full Name	Full Name
Email	${user.email}
Birthday	DD/MM/YYYY
Mobile	(320) 380-4539
Address	Bay Area, San Francisco, CA
Other	Education, Work, Skills

Edit Profile



Edit Profile

Email

chathudil77@gmail.com

Full Name

Chathushi Dilhari

Birthday

07/07/2000

Mobile

(320) 380-4539

Address

Bay Area, San Francisco, CA

Other

Education, Work, Skills

Source Code

Source Code.zip