**Topic: Jobs Portal**

**Summary:** This project aims at minimizing the problems of applicants to find the apt job position. The applicant can search for the proper jobs and accordingly get the job alerts emails also. The applicant can further upload his resume on the jobs site. Users, employers will have their accounts and can perform their tasks accordingly.

**Sample Use Cases**

* Job Search
* Job Alerts
* Update job applications
* Changing Profiles

**Roles:**

* Admin
* Applicants
* Companies
* Employers
* HR

**Anticipated Features:**

* Job Alerts features
* Resume Upload features
* Job upload features

**Anticipated Tables:**

* Interviewer
* Employer
* Applicant
* Job

**Technologies:**

* Spring MVC
* Hibernate
* MySQL
* Spring File Upload
* JavaScript
* Bootstrap
* Apache Maven
* Ajax

Controllers

Applicant Controller

@Controller

**public** **class** ApplicantController {

@Autowired

**private** UserDAO userDao;

@RequestMapping(value = "/applicantsignup", method = RequestMethod.***GET***)

**public** String applicantSignUp(Model model)

{

Applicant applicant= **new** Applicant();

model.addAttribute("applicant",applicant);

**return** "applicantSignup";

}

@RequestMapping(value = "/applicantSignUpSubmit", method = RequestMethod.***POST***)

**public** String applicantSignUpSubmit(Model model,Applicant applicant,HttpSession session)

{

applicant.setRole("applicant");

Boolean flag = userDao.saveUser(applicant);

**return** "success";

}

}

Employer Controller

@Controller

**public** **class** EmployerController {

@Autowired

**private** UserDAO userDao;

@RequestMapping(value = "/employersignup", method = RequestMethod.***GET***)

**public** String applicantSignUp(Model model)

{

Employer employer= **new** Employer();

model.addAttribute("employer",employer);

**return** "employerSignup";

}

@RequestMapping(value = "/employerSignUpSubmit", method = RequestMethod.***POST***)

**public** String employerSignUpSubmit(Model model,Employer employer,HttpSession session)

{

employer.setRole("employer");

Boolean flag = userDao.saveUser(employer);

**return** "employerHomePage";

}

@RequestMapping(value = "/employerpostjob", method = RequestMethod.***GET***)

**public** String employerPostJob(Model model,Employer employer,HttpSession session)

{

Job job = **new** Job();

model.addAttribute("job", job);

**return** "employerPostJob";

}

@RequestMapping(value = "/employerpostJobSubmit", method = RequestMethod.***GET***)

**public** String employerpostJobSubmit(Model model,Job job,HttpSession session)

{

Employer employer=(Employer) session.getAttribute("employer");

**return** "employerPostJob";

}

}