PEOPLE-WARE:

recruiting WEBapp

Table of Contents

[INTRODUCTION 2](#_Toc521335226)

[SPECIFICATIONS 2](#_Toc521335227)

[SOFTWARES REQUIRED 3](#_Toc521335242)

[SOURCE CODE 3](#_Toc521335246)

[INSTALLATION NOTES 3](#_Toc521335250)

[ABOUT THE APPLICATION 5](#_Toc521335267)

[CLASS DIAGRAMS 9](#_Toc521335271)

# INTRODUCTION

The company PEOPLE-WARE is dedicated to recruit personnel. They want to automate the selection process of candidates for the job offers requested by other companies (i.e. its clients). The selection should be based on the skills and academic degrees of the applicants. For this, it is needed to maintain the information of the applicants searching for a job and the job offers from the companies searching for candidates.

# SPECIFICATIONS

When a company wants to subscribes to the PEOPLE-WARE web app, the following information is requested:

* Company name
* Username
* Password

When a company wants to post a job offer, the following information should be provided:

### Name and description of the job position: For example: “Software Developer” and “Java developer with insights in the latest trends of web development”

### Salary (in CAD)

### Working time: Full-time or Part-time

### Job Requirements:

#### Mandatory fields:

##### Academic degree(s)

##### Job offer must have at least one skill

##### Technical skills, each with the minimum required level that is expressed with a scale from 1 to 5 (low to high).

When an applicant subscribes to the PEOPLE-WARE web app, the following information is requested:

### Full name

### Username

### Password

When an applicant applies for a posted job, the following information is requested:

* Full name
* Email
* Contact Number
* Salary
* Mandatory fields: Academic Degree(s) and Technical Skills

The web application should select the applicants that qualify for each job offer, based on

the following rules:

### Display the eligible applicant(s) Full name, Email and Contact

### Salary within the salary range of the job offer

### Mandatory fields:

##### Academic Degree(s)

##### Technical skills: defined using the scale (1-5). At least one skill must be specified

# SOFTWARES REQUIRED

The software that are needed for the application to be edited or executed

### ‘*backend*’ 🡪 Java and Spring Boot.

### ‘*frontend*’ 🡪 Angular (2/4/6), Node, Express, Bootstrap.

### database 🡪 MySQL.

# SOURCE CODE

The Source code for the application is in the ‘*recruitingApp*’ folder

### ‘*backend*’ 🡪 This folder has the source code for the API.

### ‘*frontend*’ 🡪 This folder has the source code for the Front User Interface.

### ‘*scripts*’ 🡪 This folder has a ‘*recruitingAppScripts.sql*’ file for the Database table creations.

# INSTALLATION NOTES

All the execution of this application will be done via command prompt terminal

## *Database*

## *Open the SQL workbench and import the ‘recruitingApp\_Scripts.sql’ file from the ‘scripts’ folder.*

## *Edit the name of the Database if you wish. Run the script in your workbench terminal.*

## *This will create a new Database by ‘localDb’ name with 4 tables namely ‘users’, ‘jobs’, ‘requirements’ and ‘applicants’.*

## ‘*backend’*

## *Import the project ‘recruitingApp’ into the IDE and open the ‘application.properties’ in the ‘com/java/resources/’ location.*

## *Edit ‘****spring.datasource.username****’ and ‘****spring.datasource.password****’ attributes. If you have changed the name of the database then replace ‘localDb’ with the name of your database in ‘****spring.datasource.url****’ attribute. Then Save it.*

## *Open a terminal and navigate to the ‘recruitingApp’ folder using the command:* **cd yourpath/recruitingApp/**

## *Once inside the ‘recruitingApp’ folder, run the command:* **mvn clean install**

## *This will install all the required softwares. Then enter into the ‘backend’ folder using:* **cd backend/**

## *Then execute the command:* **mvn spring-boot:run**

## *This will start the java files that possess the logic for the whole application to function.*

## *The code listens on port 9090 by default. If you wish to change, then you can do it in the ‘application.properties’ file.*

## *‘frontend’*

## *Open another Terminal window and navigate into the ‘frontend’ folder using the command:*

## **cd yourpath/recruitingApp/frontend/src/main/frontend/**

## *Then type* **npm start** *or* **yarn start** *to start the front end of the application.*

## *The User Interface runs on port 4500 by Default*.

# ABOUT THE APPLICATION

This application can be accessed by visiting the url ‘*http:localhost:4500/’* after you have installed and initiated the step given in the *‘INSTALLATION NOTES’* section*.*

The application data cannot be preloaded as it makes use of the logged in Session user information to store and retrieve information from the database. The application begins at the page displayed in screen shown in Figure 1.

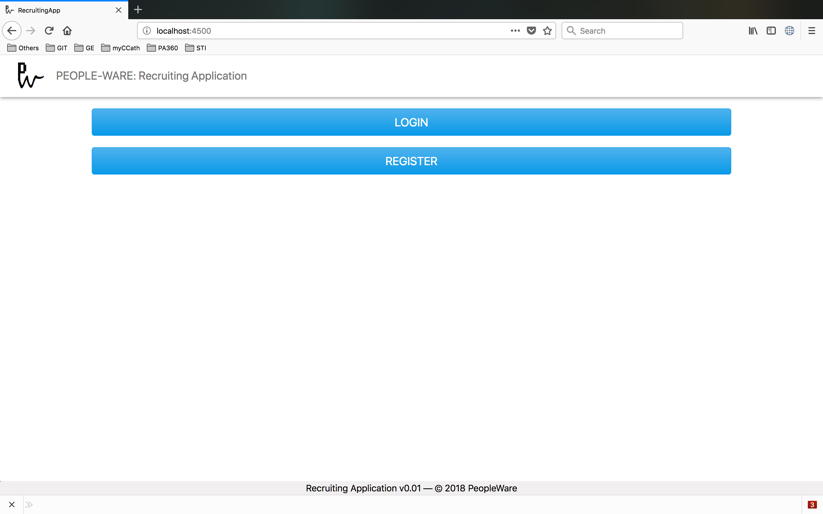


Figure : Start-up screen of UI

The new user has to Register himself either by playing the role of a Company or an Applicant. The Username and Password are encrypted using a simple Base64 conversion and then stored into the database. When a user attempts to log-in, this value is retrieved and decrypted to get the actual string and then the values are validated.

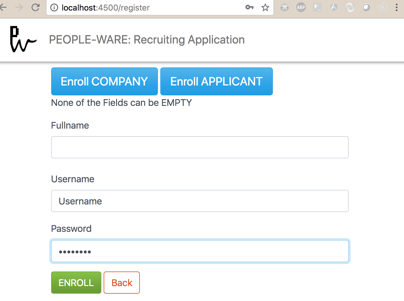
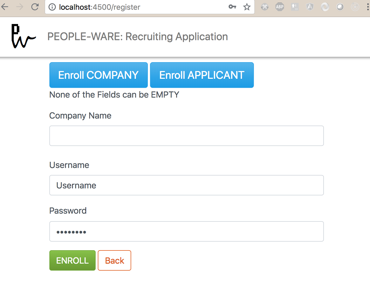
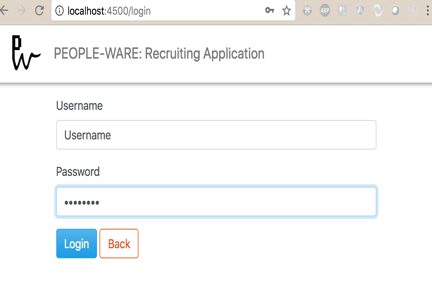
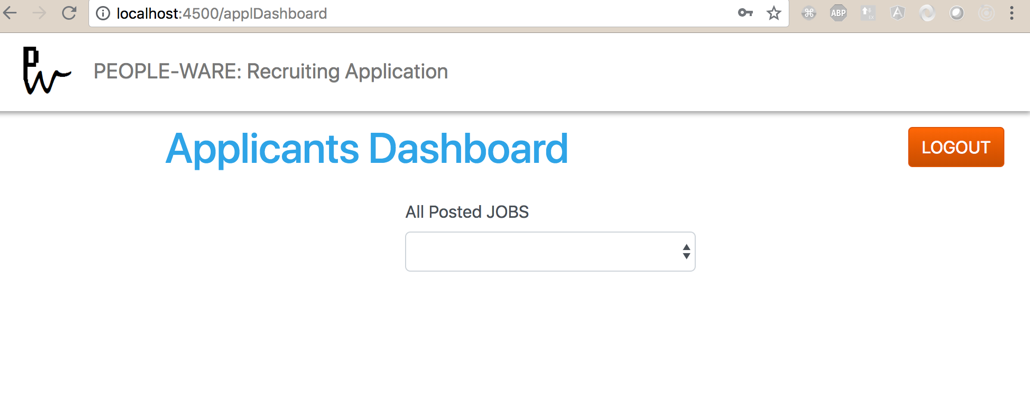
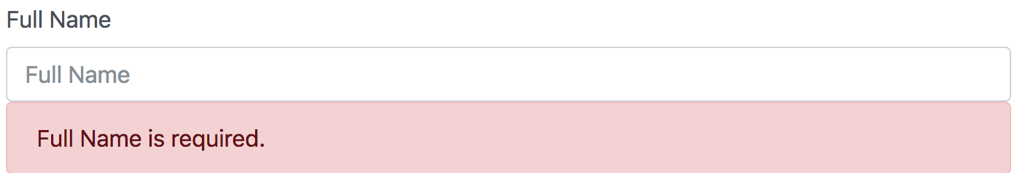


Figure : [From left to right] Login, Register(Company) and Register(Applicant) Screens.

Once the user is Validated then he is redirected to either the Company Dashboard or the Applicant Dashboard based on the information stored at Registration.

If the user was registered as an Applicant then, the user sees a dropdown menu (Figure 3) on the screen where he can choose the desired job from the list of all the available jobs. Upon selection, the user is transferred to another screen which has the job-related information (Figure 4). The User can fill the details and click on ‘Apply’ button or choose to close (Figure 4). If there are any unfulfilled fields, the error message is either displayed in the Status Bar at the bottom of the field or at the bottom or the screen (Figure 3).





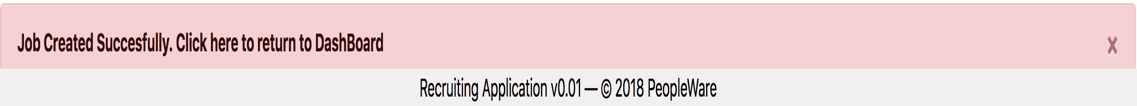


Figure : [From top to bottom] Applicant Dashboard, Status Message at Field, Status Message at the Bottom of screen

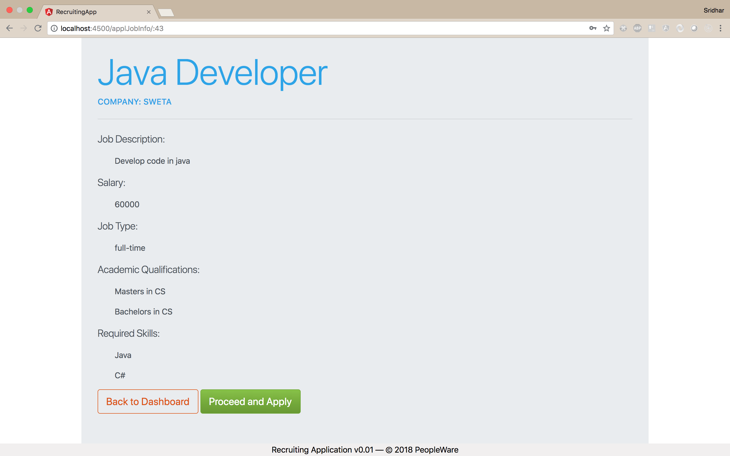
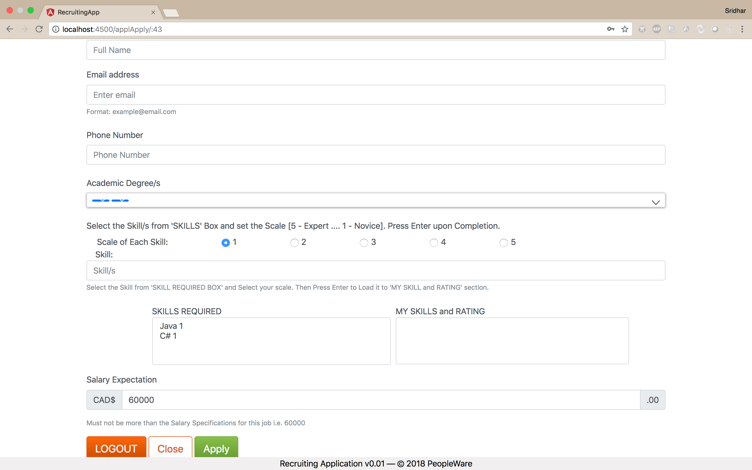
 

Figure :[From left to right] Job-Info Screen, Job-Apply Screen

If the user was registered as being associated to a company then, there are three options to choose from (Figure 5)

### SEE ALL THE APPLICANTS

### POST A NEW JOB

### EDIT AN EXISTING JOB

The user can Post a new based on the conditions mentioned in the ‘REQUIREMENTS’ section. If all the conditions are met, then the Job is inserted into the database. To enter the skills and academic degrees, the user must first enter the values and press the ‘Return’/’ENTER’ key. This pushes the values into a box which will be added as qualifications for the job.

The main edit screen has a dropdown which gets populated with the list of all the jobs this user has posted. To edit an existing, the actions remain similar to that of posting a job but here the user cannot edit the Name and Description of the Job. Here when the user clicks on the ‘X’ button on the ‘Academic Degrees’ and ‘Skills’ boxes, it clears the entries and user can fill in new information again. This box will not be available unless one entry is made into it.

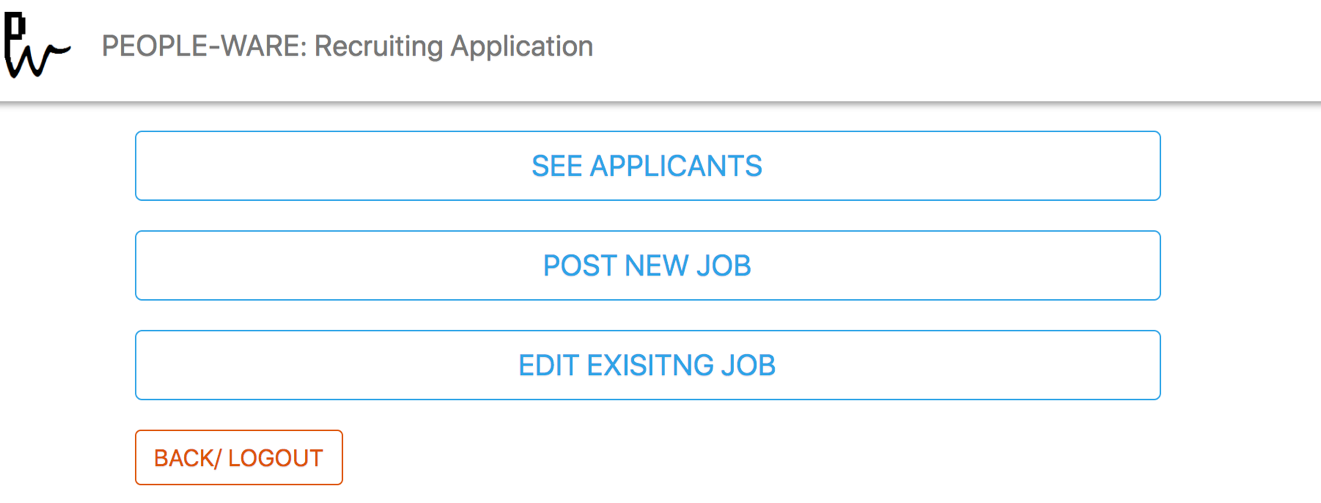
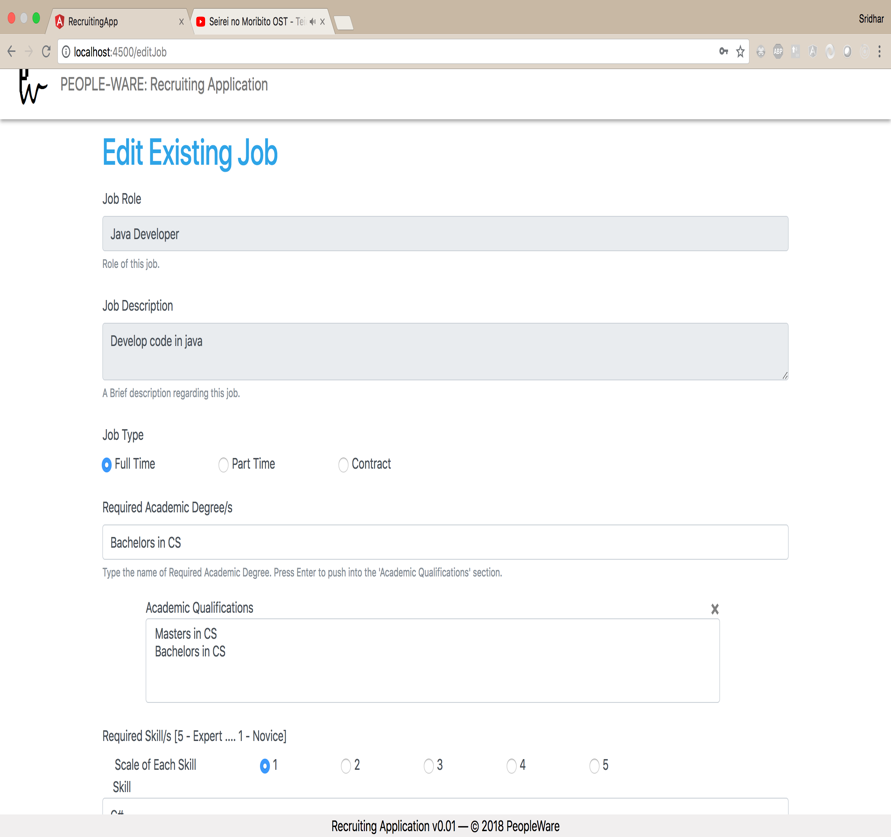
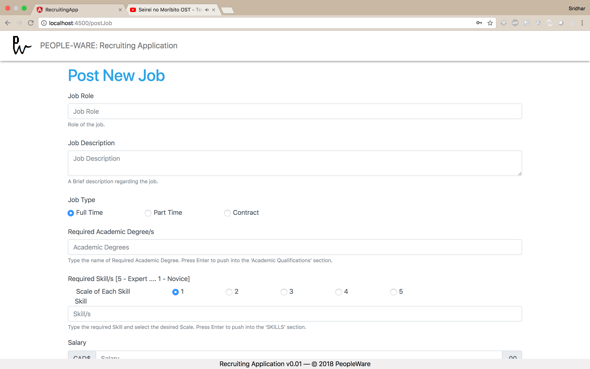


Figure : Dash-Board of the User registered on behalf of a company.

Figure :[From left to right] Post Job Screen and Qualification box with X mark

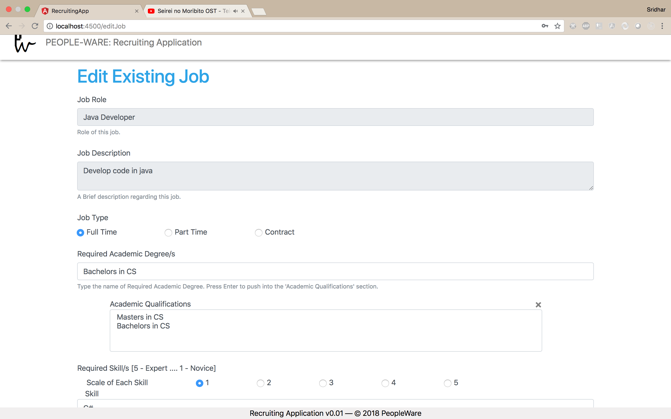
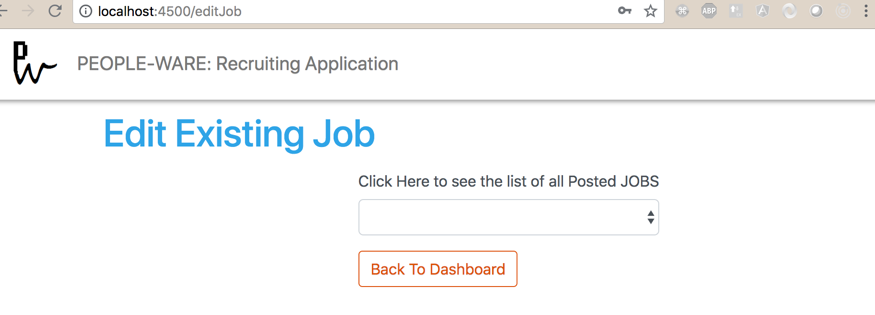


Figure :[From left to right] Edit screen dropdown, Edit Job Screen

The ‘SEE ALL APPLICANTS’ screen also begins with a dropdown which is populated with the jobs this user has posted. Upon selecting the job, the screen gets loaded with the list of all the eligible applicants.

The Eligibility criteria is such that it rates the applicants based on the total sum of the scales of all the skills the applicant has which matches with the skills required for the job. If the applicant had fed a skill which is not mentioned in the job requirement, then the scale of that particular skill is ignored and not taken into consideration during the calculation of the total scale. The eligible applicant list is sorted in descending order i.e. applicant with the maximum scale on top followed by lower ones.

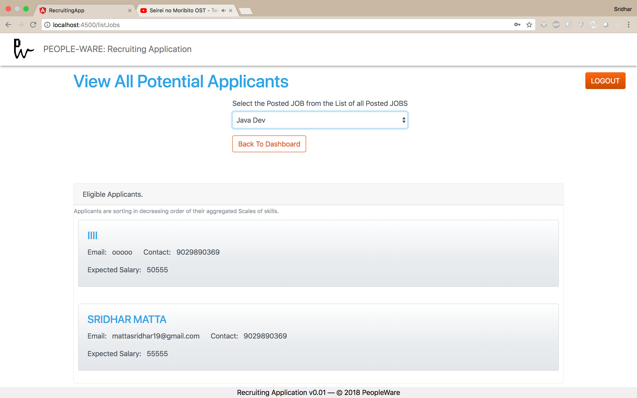
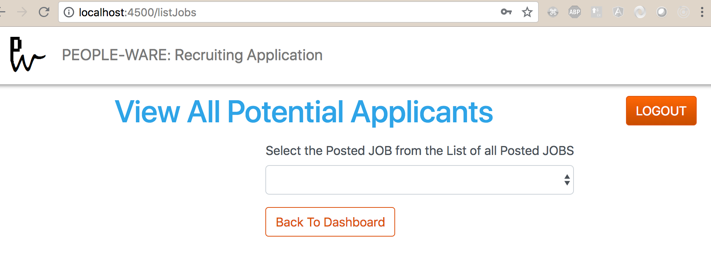
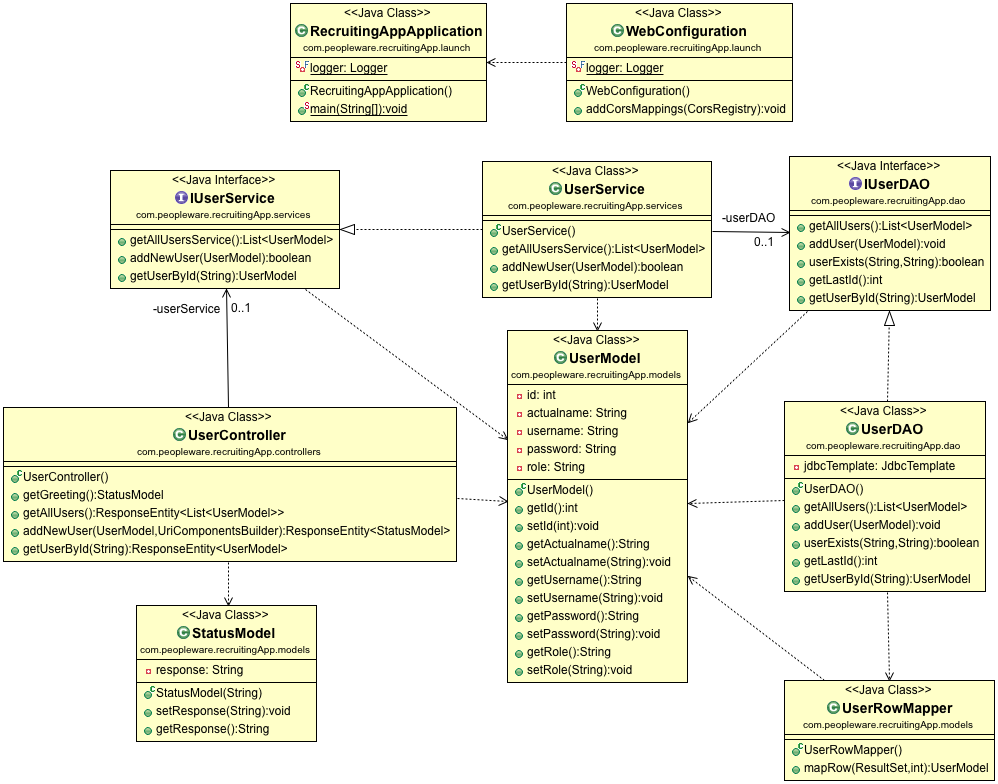


Figure :[From left to right] Dropdown to list all the job and Screen with eligible applicants

When the User hits the LOGOUT button, all the session information is cleared, and the user is redirected to the Main screen (Figure 1).

# CLASS DIAGRAMS

All the class diagrams are located in the ‘*classDiagrams*’ folder within the ‘*recruitingApp*’ folder. The class diagrams for User, Applicant, Company-Posting/Edit/Retrieval Jobs and Eligible Applicants are Figures 9 through 12 respectively.

Figure : Class Diagram for User

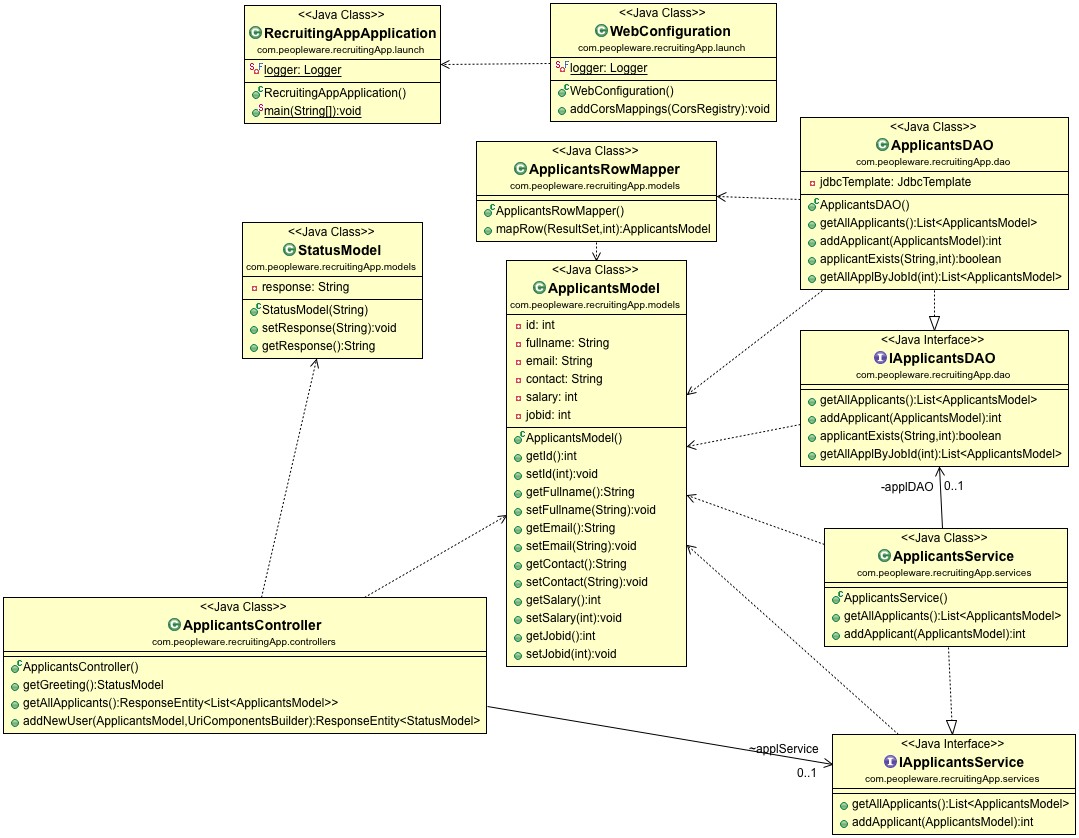


Figure : Class Diagram for Applicant

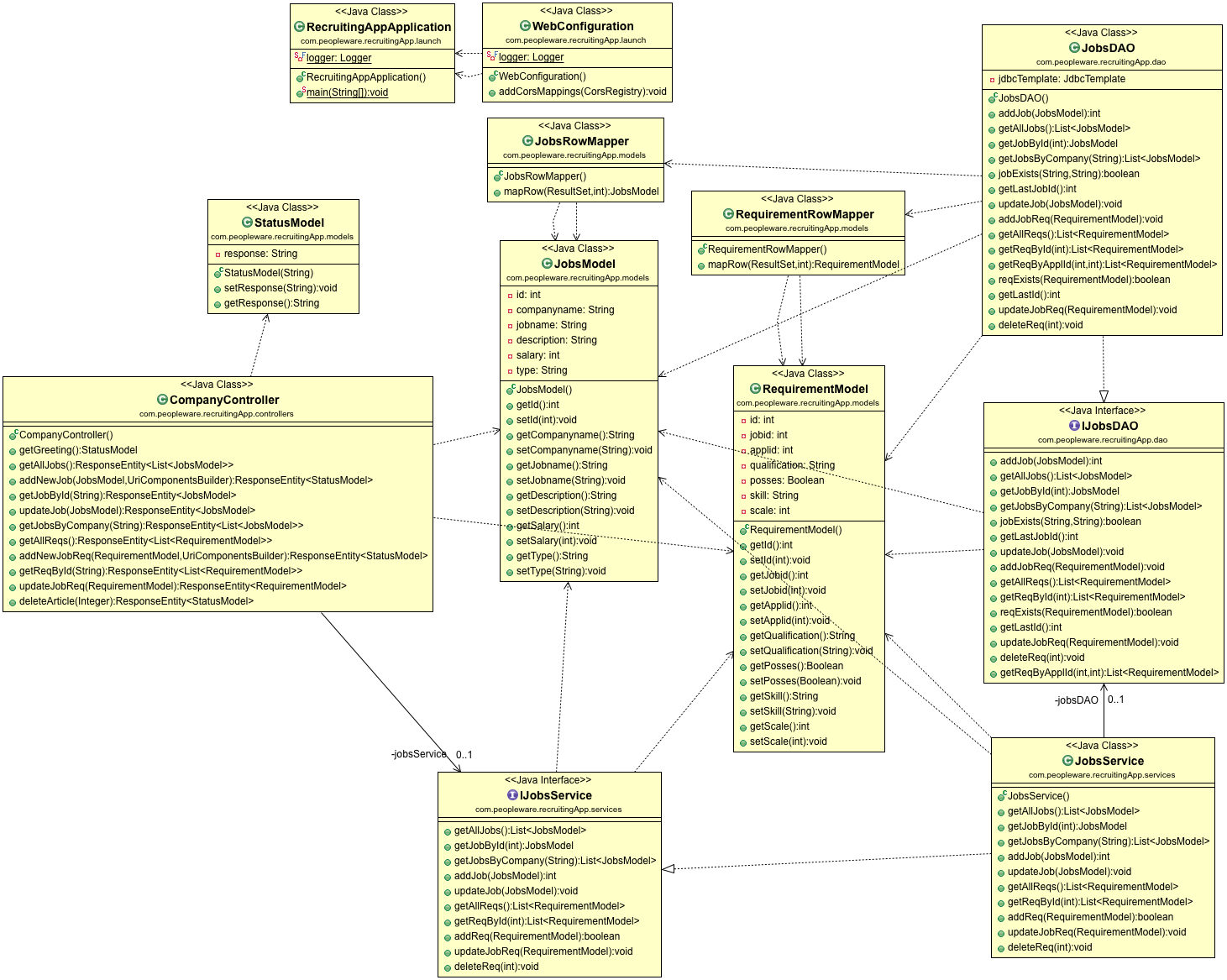


Figure : Class Diagram for Company-Posting/Edit/Retrieval Jobs

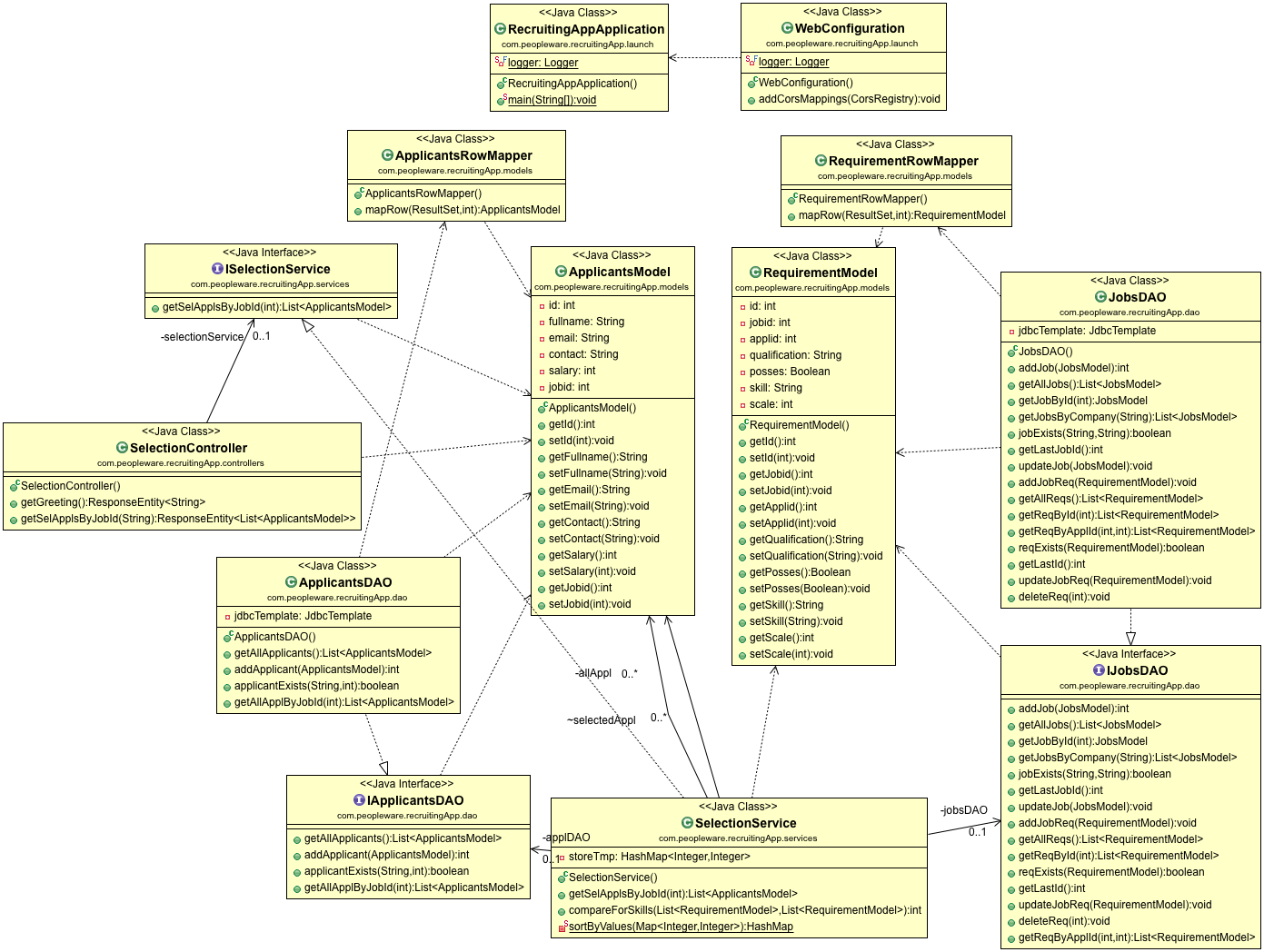


Figure : Class Diagram for Returning Eligible Applicants