DASS Assignment 1

Job Application Portal

Deadline: 20ᵗʰ January, 11.55PM

Concerned TAs - Vikrant Dewangan, Kishan, Kanish Anand, Paryul Jain
Post all doubts on Moodle or Teams (TA Discussions and Queries channel).
Messenger requests WILL NOT BE entertained!

Introduction

In this assignment you will be learning how to build a web application following REST principles. You will be using MERN stack to build the application:
- **React.js** for Frontend
- Backend Framework using **Express JS** implementing *REST* API.
- **Node.js** for Backend Engine
- **MongoDB** - Database

As part of the assignment you will build a Job Application Portal following where users have the option to search and apply for various job profiles while showcasing their skillset. At the same time, recruiters from various companies have the ability to create job listings based on their requirement for various skill sets, experiences.

Requirements [100 marks]

- Your website hosts two kinds of users - Job applicants and Recruiters.
- Login and Registration [10 marks]
    - Common registration page with drop down to choose their type (Job Applicant/ Recruiter) and have fields appear as per their profile.
    - Common login portal redirecting to respective UI after login.
    - You can also have login using Google/ Facebook/Twitter API. You can have them select Applicant or Recruiters for the first time in API login. (**Bonus**) [5 marks]
- We first describe how a Job should be like, after that we describe the Job Applicant and Job Recruiter.

Job Details

- Each job, should contain the following with it
  - Title of job
  - Name, Email ID of Recruiter who posted it.
  - Maximum number of
    - Applications - No. you allow to apply on website
    - Positions - No. of available jobs
  - Date of posting (Day, Month, Year)
      Example: Day-23, Month-June, Year-2021.
  - Deadline for application (Day, Month, Year, Hour, Minute).
  - Required Skill sets (List of Languages)
  - Type of Job (Full-time / Part-time / Work from Home)
  - Duration - Integer indicating number of months. (0 (indefinite) - 6 months)
  - Salary per month (Integer)
  - Rating (0-5) - Should be dynamic

Job Applicants Use Cases [45 marks]

- Profile Section Details with editing option [5 marks] -
  - Name, Email ID
  - Education - Each education instance contains the following
    - Institution Name - A textbox for the college/ school they studied in.
    - Start Year (YYYY) - compulsory, and End Year (YYYY) - (not compulsory)
      Job Applicants must be able to add multiple education instances with the start and end year(if applicable) for each.
  - Skills
    - A list of languages they are comfortable in. They should be able to add additional language not already present.
  - Rating (0-5) - Should be dynamic
  - Upload Resume / CV PDF (**Bonus - 2.5 marks**)
    - An upload button asking them to submit a pdf.
  - Profile Image Upload (**Bonus - 2.5 marks**)
    - An image upload option for profile photo.
- Dashboard to view job listings. There should be an option for searching a job based on its title. Sort option based on each Salary, Duration and Rating both ascending and descending. [10 marks]
- Implement fuzzy searching based upon Job title. (**Bonus - 2.5 marks**)

- Should have a filter section like e-commerce websites, with following filters provided [5 marks]:
  - Job Type - Should be able to select and filter from Full-time / Part-time / Work from Home.
  - Salary - Range: filter from X to Y range. You can have two textboxes for X and Y or you can also have a Slider with 2 pins for filtering.
  - Duration - Dropbox, with each option filtering for job listings with duration **strictly lesser** than itself. Entries are in increments of a month from 1-7.
- Each job listing should have fields like Title, Recruiter Name, Recruiter Rating, Salary, Duration, Deadline of application displayed with an "Apply" button. [5 marks]
  - *Note that for each job listing where the applicant has already applied to, the apply button should be changed to "Applied" (different colors are required).*
  - *For a job listing with the maximum number of applications reached or positions filled, the "Apply" button should be changed to "Full" (different colors are required).*
- Upon clicking the "Apply" option, the applicant should then be prompted to write a text SOP (Statement of Purpose) of not more than 250 words, with a "Submit" button down. After "Submit", application is sent to Job Recruiter.. [5 marks]
- For each applicant, there must **NOT** be more than 10 open applications. An error popup should appear in case the applicant tries to apply to more. [5 marks]
- A UI titled "My Applications'' to view their applications in a table. Note that it should **also** contain the Rejected Applications. The table should contain details like Title, date of joining, salary per month, name of recruiter. along with an option to rate the recruiter from 0-5. [10 marks].

Recruiters Use Cases [45 marks]

- Profile section details with editing option [5 marks] -
  - Name, Email ID, Contact Number, Bio (max 250 words)
- Dashboard to Create Job listings (Should provide all required fields). [5 marks]
- A dashboard for all **active** job listings posted by the recruiter. For each listing, display the Title, Date of posting, Number of Applicants, Maximum Number of Positions. [5 marks]
- For each job listing, there should be an option to delete it. It should also have an edit option for the following fields [5 marks] -
  - Maximum Number of Applicants
  - Maximum Number of Positions
  - Deadline for Application

- For each job listing above, clicking them should lead to another dashboard with all **non-rejected** applications in view, with each applicant's Name, Skills, Date of Application, Education, SOP, Rating, Stage of Application in view. The table should have a sort option for Name, Date of Application and applicant's rating both in ascending and descending order. [10 marks]
  - Resume download button (**Bonus - if implemented upload resume -2.5 marks**),
- Beside each such entry of an applicant, there should be 2 buttons - "Shortlist"/"Accept" and "Reject". [5 marks]
  Note:- An application can be in any of the following 4 stages -
  - Applied - Initial stage when first applied.
  - Shortlisted - After receiving the applications for a job, the recruiter can choose to either "Shortlist" or "Reject" the application.
  - Accepted - For an application in Stage 1, the recruiter can choose to either "Accept" or "Reject" the application. After Accepting the application, the Job Applicant joins the Recruiter's company as an Employee.
  - Rejected - At any stage, the Recruiter may reject the application, after which the Application gets deleted from the Recruiter database.
  Thus the first option will be "Shortlist", or "Accept" depending on the current stage of the application.
- A dashboard to view all employees accepted by the recruiter. For each applicant, you should display their Name, Date of Joining, Job Type (Full-Time/ Part-Time/ WFH) and the Job Title they got accepted into. The recruiter should also get an option to rate the employee from 0-5. The table should have a sort option for Name, Job Title, Date of Joining, and Applicant's rating both in ascending and descending order. [10 marks]

Bonus [20 marks]

- Email to the applicants on acceptance of their application. You can create a common email ID for mailing purposes stating that an "xyz" recruiter accepted your application. (**5 marks**)
- *Other bonuses are explained along with concerned points. Marks for every bonus are written along with them.* **(Total 15 marks)**

Please take care of the following -
- An applicant should **NOT** be allowed to apply for another listing if his or her application is already accepted into a listing. All applications into other listings must be **rejected automatically** whenever any applicant is accepted into a job .
- A job listing should not appear in the search/filter results after it's deadline.

- Error Handling should be well implemented such that all the edge cases are taken care of like the field of job salary value should accept only positive integers.

Deliverables

Submission must be in the following format:

```
<roll_no>/
├── backend/
├── frontend/
└── README.md
```

where backend/ and frontend/ are directories with the Express.js and React app in it respectively. Zip this up and submit one <roll_no>.zip file. Note that there should be a **package.json** inside each of the frontend and backend folders.
There will be a **20% penalty** for not following the submission format and **straight 0** for submitting corrupt zips.

Additional Instructions

Submission must be in the following format:
- You may use any npm package, as long as it's not off-the-shelf. (Confirm it from TAs)
- You are allowed to use [Bootstrap/CSS templates](), or [Material UI]().
- Follow good coding practices and good database designs.
- In the final submission, ensure that you have the correct **package.json** files. Make sure it includes all the packages you have worked with.
  Do NOT submit the 'node_modules/' folder.
- Cases of plagiarism will be given a straight zero. **DO NOT COPY ANY PART FROM ANY SOURCE** including your friends, seniors or the internet.
- Marks indicated for requirements assume proper working of frontend and handling of corresponding database operations.
- Please test your application thoroughly. Any errors occurring during the evaluation will be penalized.
- Please start early. The deadline is a hard one, and won't be extended.