

## Requirements for Assignment 1

React js-front end

Backend framework - expressjs

Nodejs - backend engine

Database - Mongo DB

Web application necessities:

- Job application portal
- Users-search and apply using skills.
- Recruiter- create job listing based on required skills
- LOGIN AND REGISTRATION
  - Dropdown for applicant/recruiter, fields display
  - Login portal to a user interface
  - Plugins with google etc
- Job details
  - Name of job
  - Name of R
  - Email id of R
  - Max no. of applicants and positions
  - Date of posting
  - Deadline for application-date and time
  - Required skills
  - Type of job- full/part-time/WFH
  - Duration of job
  - Salary
  - rating
- Applicants
  - Profile section (edit option)
    - Name of A
    - Email id of A
    - Multiple Education section: institution, start year(compulsory), end year(not mandatory)

- Skills, add language option.
    - Rating-dynamic
    - Upload resume, profile picture with upload buttons
  - Dashboard:
    - Search job using the title
    - Sort based on salary, duration, rating, ascending, descending
  - Fuzzy search
  - Filter section:
    - Job type
    - Salary-range x to y
    - Duration : dropbox-strictly lesser(1-7)
  - Job listing:
    - Title
    - Name of R
    - Rating of R
    - Salary
    - Duration
    - Deadline
    - Apply button: red, yellow-green colours for status
  - Apply clicked: prompt for writing SOP(250-word limit), submit button.
  - After submit, the application is sent to the recruiter.
  - After the limit is reached, apply changed to a full button for A
  - Max 10 open application, error message if more are tried
  - My applications sidebar:
    - Applications in table
    - Should have rejected applications also.
    - Title, date of joining, salary, name of R, rated R
- Recruiter
    - Profile section:
      - Name
      - Email
      - Contact
      - bio(250 max)
    - Dashboard to create a job listing
    - Dashboard for active job listings: each listing has
      - Title

- Date of posting
  - No. of applicants
  - Max no. of positions
  - deadline
- Job listing: delete option, edit option
  - Max no. of A edit
  - Max positions edit
  - Deadline
- .clicking job listing: redirect to another dashboard
  - Non-rejected applications
    - Name of A
    - Skills
    - Date of application
    - Education
    - SOP
    - Rating
    - State of application
  - Sort options: ascending and descending
    - Name
    - Date of application
    - Rating
  - Resume download buttons
- Beside an A entry: 2 buttons:
  - Shortlist / accept
  - Reject
- Application status:
  - Applied: the initial stage
  - Shortlisted: after choosing the button, after being shortlisted, button changes to accept
  - Accept: after accepting button, becomes an employee
  - Reject: after reject button can be done at any stage, and application gets deleted from R database
- Dashboard:
  - View employees
  - Each applicant:
    - Name
    - Job type

- Date of joining
- Job title
- Rate employee option
- Sort option: ascending, descending
  - Name
  - Job title
  - Date of joining
  - Applicant rating
  - Email to applicants on acceptance from [xyz@gmail.com](mailto:xyz@gmail.com)
- Applicant cant be allowed for another listing if already accepted into a listing
- All others are automatically rejected
- After deadline: should not appear
- Error handling for edge cases and data types
- Comments are necessary everywhere
- Normalisation needed
-