TEAM 14 - PROJECT REPORT ON JOB PORTAL

ABSTRACT

This project describes a job portal system designed to make the job search process easier for job seekers, employers, and administrators. The system is made up of several parts, each with its own specific functions

This job portal is designed to be user-friendly, efficient, and scalable, providing a better experience for job seekers and employers. It aims to streamline the job search and recruitment process.

INTRODUCTION

In the current job market, finding the right job or the right candidate can be challenging. To address this, we have developed a comprehensive job portal system that connects job seekers with employers seamlessly. This system is designed to simplify the job search and recruitment process for everyone involved.

The job portal is divided into several key components:

**Person Module:** Handles user authentication, ensuring secure access to the system.

**Jobseeker Module**: Allows job seekers to create profiles, manage resumes, and apply for jobs effortlessly.

**Admin Module:** Empowers administrators to manage job postings, user accounts, and oversee applications.

**Company Module**: Enables employers to register, post job vacancies, and specify job requirements.

**Job Module:** Facilitates job posting, searching, and applications, making it easy for job seekers to find suitable jobs.

**Resume and Resume Selection Modules:** Manage and evaluate resumes, storing academic and personal details to match candidates with job openings.

**Feedback Module:** Collects user feedback to continuously improve the system.

The system is designed to provide a range of functionalities, including user authentication, job seeker profile management, job posting, resume evaluation, and administrative control. By integrating these features, the job portal aims to enhance the overall job search experience and facilitate better matches between candidates and job vacancies.

This job portal system aims to provide an efficient, user-friendly platform that bridges the gap between job seekers and employers, making the recruitment process more effective and less time-consuming.

FUNCTIONAL REQUIREMENTS

**1. User Management:**

- Login/Logout: Users should be able to log in and log out of the system using their username and password.

- User Registration: Users should be able to register by providing their username, password, and email.

**2. Job Seeker Features:**

- Profile Management:

- Create Profile: Job seekers should be able to create a profile with their personal details like name, email, address, and contact number.

- Update Profile: Job seekers should be able to update their profile information as needed.

- Delete Profile: Job seekers should have the option to delete their profile from the system.

- Resume Management:

- Create Resume: Job seekers should be able to create a resume with their academic and personal details.

- job Search and Application:

- Search Jobs: Job seekers should be able to search for available jobs.

- Apply for Jobs: Job seekers should be able to apply for jobs they are interested in.

- View Applied Jobs: Job seekers should be able to view the list of jobs they have applied for.

**3. Job Management:**

- Add Job: Authorized users should be able to add new job postings to the system.

- Search Jobs: Users should be able to search for jobs based on various criteria.

- View Jobs: Users should be able to view details of available jobs.

**4. Company Features:**

- Company Profile Management:

- Create Company Profile: Companies should be able to create a profile with their name, address, and description.

- Add Requirements: Companies should be able to add job requirements.

- Add Vacancies: Companies should be able to add job vacancies.

**5. Admin Features:**

- Job Management :

- View Job Applications: Admins should be able to view all job applications.

- Approve Job Posts: Admins should be able to approve job postings before they are visible to job seekers.

- Remove Users: Admins should be able to remove users from the system if necessary.

**6. Feedback Management:**

- Submit Feedback: Users should be able to provide feedback about their experience or any issues encountered.

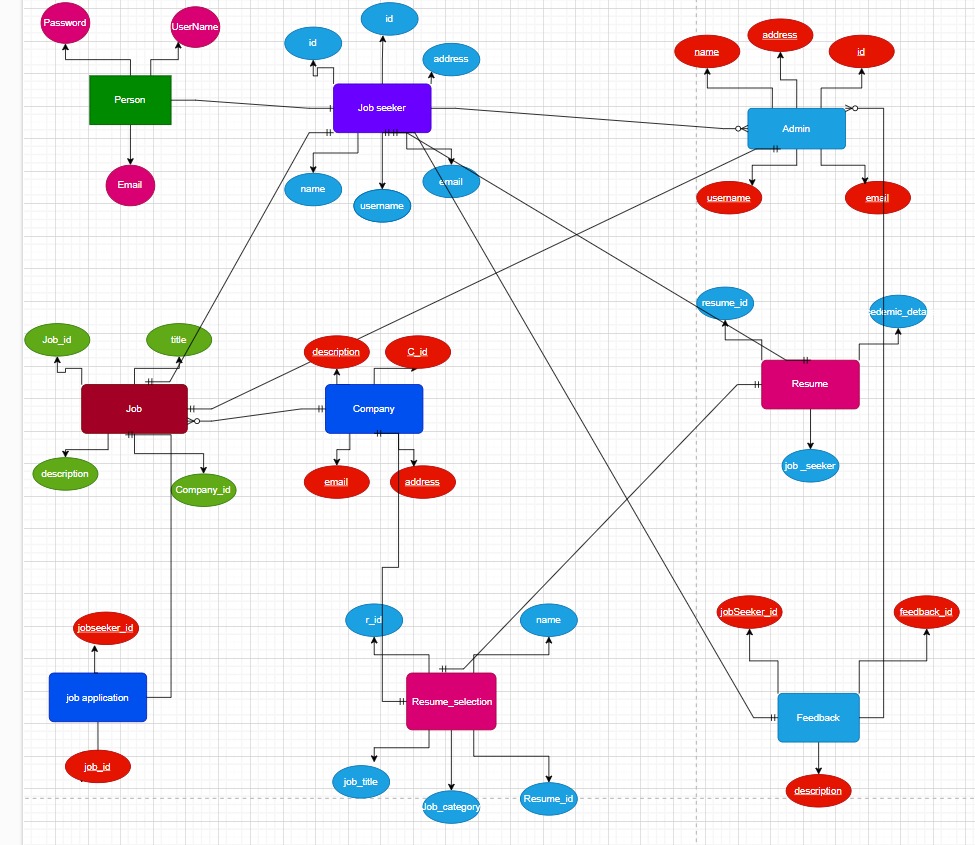
**7. Resume Selection:**

- Get Resume Results: Users or companies should be able to retrieve results of resume matching based on job titles and categories.

- Set Resume Results: The system should be able to update resume matching results.

These functional requirements outline the key features and functionalities that the system should provide to meet the needs of job seekers, companies, and administrators.

(ER) ENTITY RELATIONSHIP DIAGRAM



**Entities and Attributes:**

**1. Person:**

- Represents individuals interacting with the system.

- **Attributes** (columns) might include:

- `person\_id`: A unique identifier for each person.

- `first\_name`, `last\_name`: Names of the person.

- `email`: Email address for communication.

- Other relevant attributes like phone number, address, etc.

**2. Job seeker:**

- Refers to individuals actively seeking employment.

- **Attributes**:

- `job\_seeker\_id`: Unique identifier for each job seeker.

- `skills`: Skills possessed by the job seeker.

- `experience`: Work experience details.

- `desired\_position`: The type of job they are looking for.

**3. Admin:**

- Represents system administrators or staff managing the recruitment process.

- **Attributes**:

- `admin\_id`: Unique identifier for each admin.

- `username`, `password`: Credentials for system access.

- Other relevant attributes related to admin roles.

**4. Company:**

- Represents organizations offering job positions.

- **Attributes**:

- `company\_id`: Unique identifier for each company.

- `company\_name`: Name of the company.

- `industry`: Industry sector (e.g., technology, healthcare).

- `location`: Company's geographical location.

**5. Job:**

- Represents specific job openings.

- **Attributes**:

- `job\_id`: Unique identifier for each job.

- `job\_title`: Title of the job position.

- `description`: Detailed job description.

- `salary`: Salary information.

**6. Resume:**

- Contains details about a job seeker's qualifications.

- **Attributes**:

- `resume\_id`: Unique identifier for each resume.

- `content`: Textual content of the resume.

- Other relevant attributes like education, certifications.

**Relationships**

1. Person and Jobseeker: One-to-One

- A person can be a jobseeker.

2. Jobseeker and Feedback: One-to-Many

- A jobseeker can give multiple feedbacks.

3. Jobseeker and Job: Many-to-Many

- A jobseeker can apply for multiple jobs, and a job can have multiple jobseekers applying for it.

4. Admin and Job: One-to-Many

- An admin can manage multiple jobs.

5. Company and Job: One-to-Many

- A company can post multiple jobs.

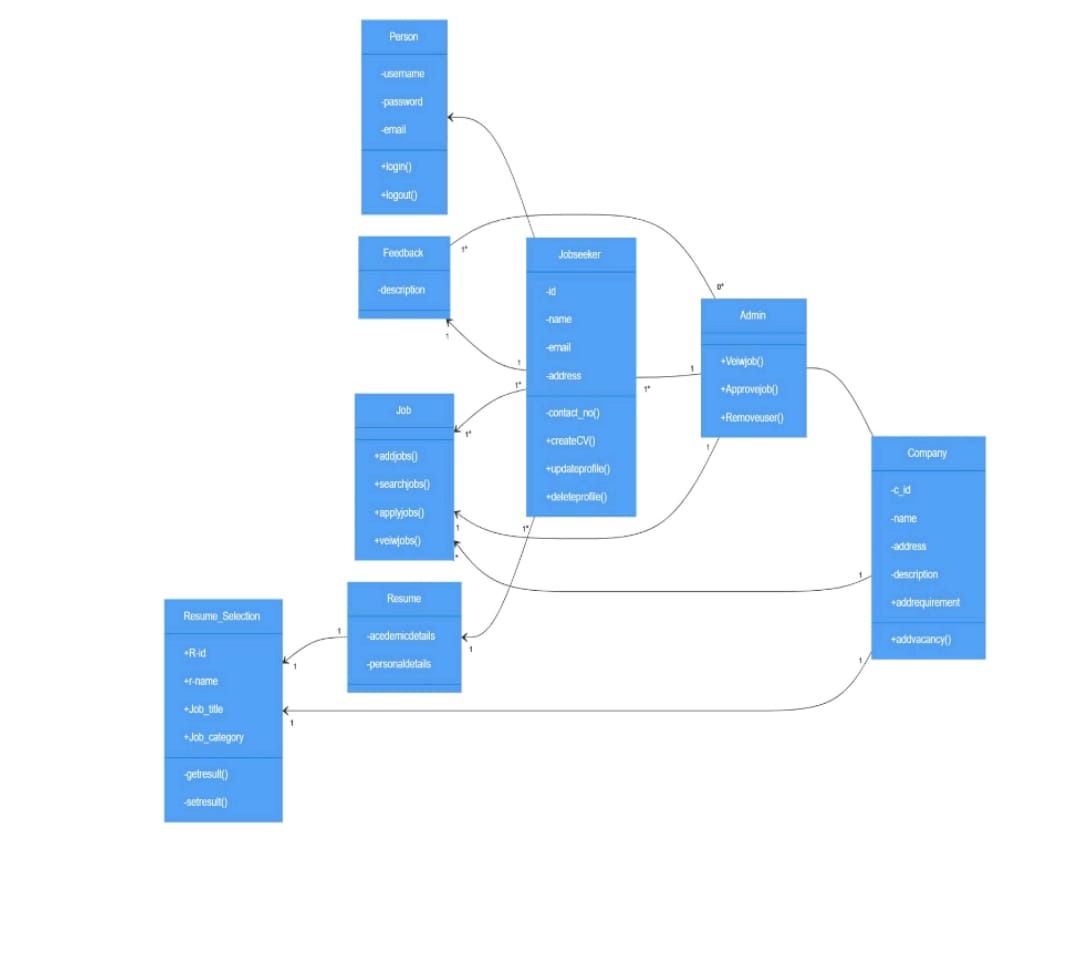
6. Jobseeker and Resume: One-to-One

- A jobseeker has one resume.

7. Resume and Resume\_Selection: One-to-One

- A resume is linked to one resume selection.

CLASS / UML ATTRIBUTE & METHOD DIAGRAM



**Classes and Attributes:**

**Person:**

Attributes: Username, Password

Methods: login(), logout()

**admin :**

Methods: ViewJobs(), ApproveJobs(), RemoveUser()

**Jobseeker :**

Attributes: id, Name, Address, Email\_ID, Contact\_No

Methods: CreateCV(), UpdateProfile(), DeleteProfile()

**company:**

Attributes: c\_id, name, address, description

Methods: AddVacancy(), AddRequirement()

**Feedback:**

Attributes: Description

Job:

Methods: AddJobs(), SearchJobs(), ApplyJobs(), ViewJobs()

**Resume:**

Attributes: academicDetails, PersonalDetails

Relationships:

The admin manages Jobseekers ( admin to Jobseeker with a 1…n ).

Admin approves Feedback ( admin to Feedback with 1 multiplicity).

Jobseekers apply to Jobs (Jobseeker to Job with 1…n multiplicity).

Companies perform Resume Selection (Resume with 1 multiplicity).

CHALLENGES FACED

* We have faced challenges when inserting more values into the tables
* Faced many challenges when giving relations between two tables
* While writing java code we faced many exceptions like IO Expections,Constructor Mismatch etc.