AVANTHI PG COLLEGE

Department of Computer Science

CERTIFICATE

This is to certify that Mr **Md Azhar** has satisfactory completed the Project Work in **Job Portal System** for Computer Science Course in the Laboratory for the year 2024 – 2025.

Course : MCA
Semester : 3
HT No :130523862091

EXTERNAL EXAMINER	INTERNAL EXAMINER	HEAD OF DEPARTMENT

Table of Contents :

DECLARATION

I am **Md Azhar** hereby declare that the project entitled “Job Portal System” has been carried out by me. This project is submitted to Osmania University Hyderabad in partial fulfillment of requirements for the award of the degree of “MASTERS OF COMPUTER APPLICATIONS”. The results embodied in this dissertation have not been submitted to any other University or institution for the award of Degree or Diploma

Project Report: Job Portal System

1. Introduction

The **Job Portal System** is a comprehensive platform designed to bridge the gap between job seekers and employers. It serves as an efficient and user-friendly solution for managing job applications, streamlining recruitment processes, and enhancing job search experiences. The system caters to both job seekers looking for employment opportunities and employers seeking suitable candidates to fill their vacancies.

For job seekers, the portal provides a seamless experience to browse job listings, filter search results based on preferences, and apply for positions directly through the platform. They can create and manage their profiles, upload resumes, and track application statuses. Additionally, job seekers receive notifications on relevant job postings, helping them stay updated with new opportunities.

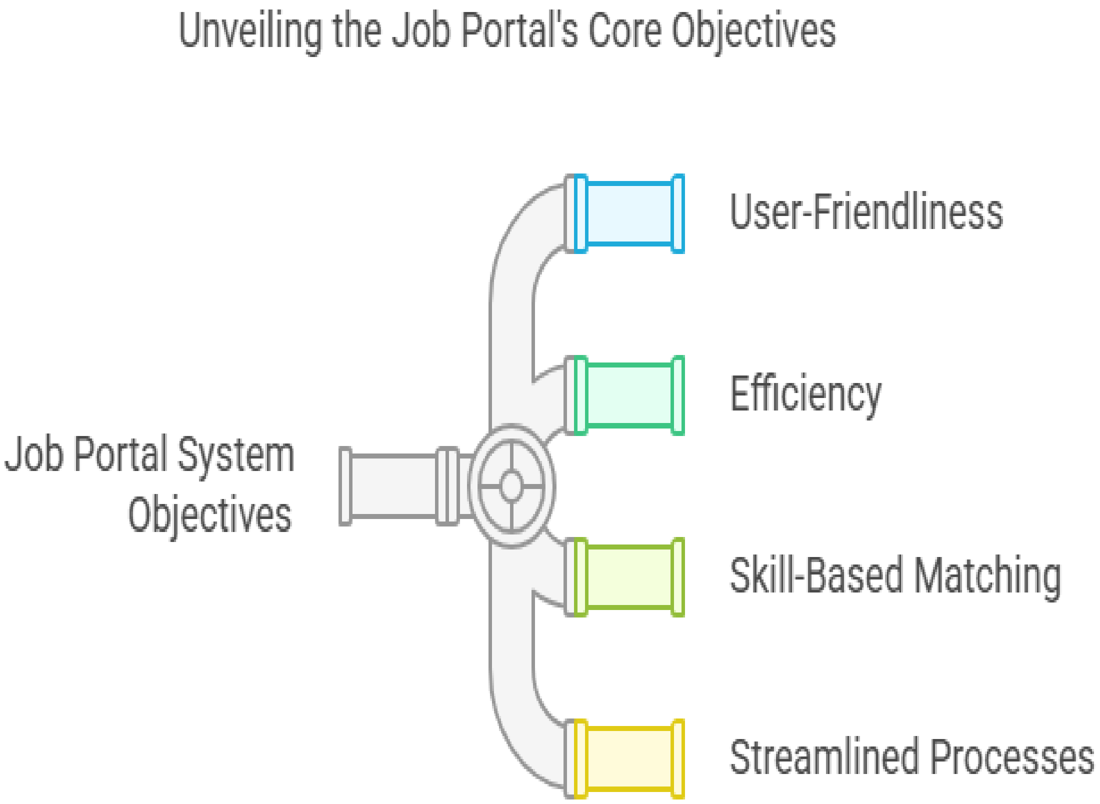
Employers benefit from the portal by gaining access to a vast pool of potential candidates. They can post job openings, specify job requirements, and review applications efficiently. The system also facilitates applicant tracking, enabling employers to shortlist, schedule interviews, and manage the hiring process effectively.

The Job Portal System incorporates modern technological features such as real-time notifications, advanced search filters, and a user-friendly interface, ensuring a smooth and productive experience for all users. By automating various aspects of recruitment and job searching, the platform enhances efficiency, reduces manual effort, and increases the chances of finding the right match for both employers and job seekers.

2. Objectives

The key objectives of the Job Portal System are:

- **User-Friendly Platform for Job Seekers:** The system offers an intuitive and accessible interface, allowing job seekers to search, filter, and apply for jobs with ease. It also provides profile management and job alerts for a personalized experience.
- **Efficient Job Posting and Application Management:** Employers can easily post job vacancies, set hiring criteria, and manage applications through an organized dashboard, enabling a more streamlined recruitment process.
- **Skill-Based Job Searching and Hiring:** The platform enables job seekers to find jobs that match their skills and qualifications while allowing employers to filter and shortlist candidates based on specific skill sets, ensuring a better job fit.
- **Streamlined Interview Scheduling and Hiring Process:** Employers can schedule interviews, send notifications, and manage the hiring process efficiently with automated tools, reducing administrative overhead and improving hiring timelines.


Unveiling the Job Portal's Core Objectives
Job Portal System Objectives
User-Friendliness
Efficiency
Skill-Based Matching
Streamlined Processes

3. System Features

3.1 For Job Seekers:

- User registration and profile management.
- Job search functionality with category, skill, and location filters.
- Online job application submission.
- Resume upload and management.
- Application status tracking.
- Feedback system for employer reviews.

3.2 For Employers:

- Employer registration and company profile setup.
- Job posting and management.
- Application tracking and review system.
- Interview scheduling and status updates.
- Feedback submission for job seekers.

4. System Architecture

The Job Portal System consists of the following key entities and their attributes:

4.1 JobSeeker

- Attributes: **jobSeekerId, name, email, password, phone, createdAt, updatedAt, skillId**
- Functionality: Register, search jobs, apply for jobs, track applications.

4.2 Employer

- Attributes: **employerId, companyName, email, password, phone, address, createdAt, updatedAt**
- Functionality: Post jobs, review applications, schedule interviews, provide feedback.

4.3 JobPost

- Attributes: **jobPostId, employerId, skillId, categoryId, title, description, jobTypeId, location, salary, datePosted, status, createdAt, updatedAt**
- Functionality: Job postings with category and skill association.

4.4 Application

- Attributes: **applicationId, jobPostId, jobSeekerId, status, dateApplied, createdAt, updatedAt**
- Functionality: Job applications tracking.

4.5 Interview

- Attributes: **interviewId, applicationId, date, time, location, status, createdAt, updatedAt**
- Functionality: Interview scheduling and status updates.

4.6 Category

- Attributes: **categoryId, name, createdAt, updatedAt, jobPostId**
- Functionality: Job categorization.

4.7 Skill

- Attributes: **skillId, name, createdAt, updatedAt, jobPostId**
- Functionality: Skill tagging for job seekers and job posts.

4.8 Feedback

- Attributes: **feedbackId, employerId, jobSeekerId, message, createdAt**
- Functionality: Employer feedback on job seekers.

4.9 JobType

- Attributes: **jobTypeId, name, createdAt, updatedAt, jobPostId**
- Functionality: Job type classification (e.g., full-time, part-time).

4.10 Role

- Attributes: **roleId, name, createdAt, updatedAt, jobSeekerId, employerId**
- Functionality: Role-based access control.

4.11 Resume

- Attributes: **resumeId, jobSeekerId, filePath, createdAt, updatedAt**
- Functionality: Resume upload and management.

5. Relationships :

1. **JobSeeker → Resume**

- One-to-One: A single job seeker can upload single resume.

2. **JobSeeker → Application**

- One-to-Many: A job seeker can apply for multiple job posts.

3. **Employer → JobPost**

- One-to-Many: An employer can post multiple job listings.

4. **JobPost → Category**

- Many-to-One: A job post belongs to a specific category (e.g., IT, Finance).

5. **JobPost → Application**

- One-to-Many: A job post can have multiple applications submitted.

6. **Application → Interview**

- One-to-One: Each application can have a scheduled interview.

7. **Application → Resume**

- One-to-One: Each application can result in one Resume.

8. **JobPost → JobType**

- Many-to-Many: A job post is associated with a job type (e.g., Full-time, Part-time, Work from Home etc). One JobType can be associated with multiple JobPost and a JobPost can be associated with multiple JobType.

9. **JobSeeker → Skill**

- Many-to-Many: A job seeker can have multiple skills, and a skill can belong to multiple job seekers.

10. **JobPost → Skill**

- Many-to-Many: A job post can require multiple skills, and a skill can belong to multiple job posts.

11. **Employer → Feedback**

- One-to-One: An Employer can send feedback.

12. **Jobseeker → Feedback**

- One-to-One: A Jobseeker can send feedback.

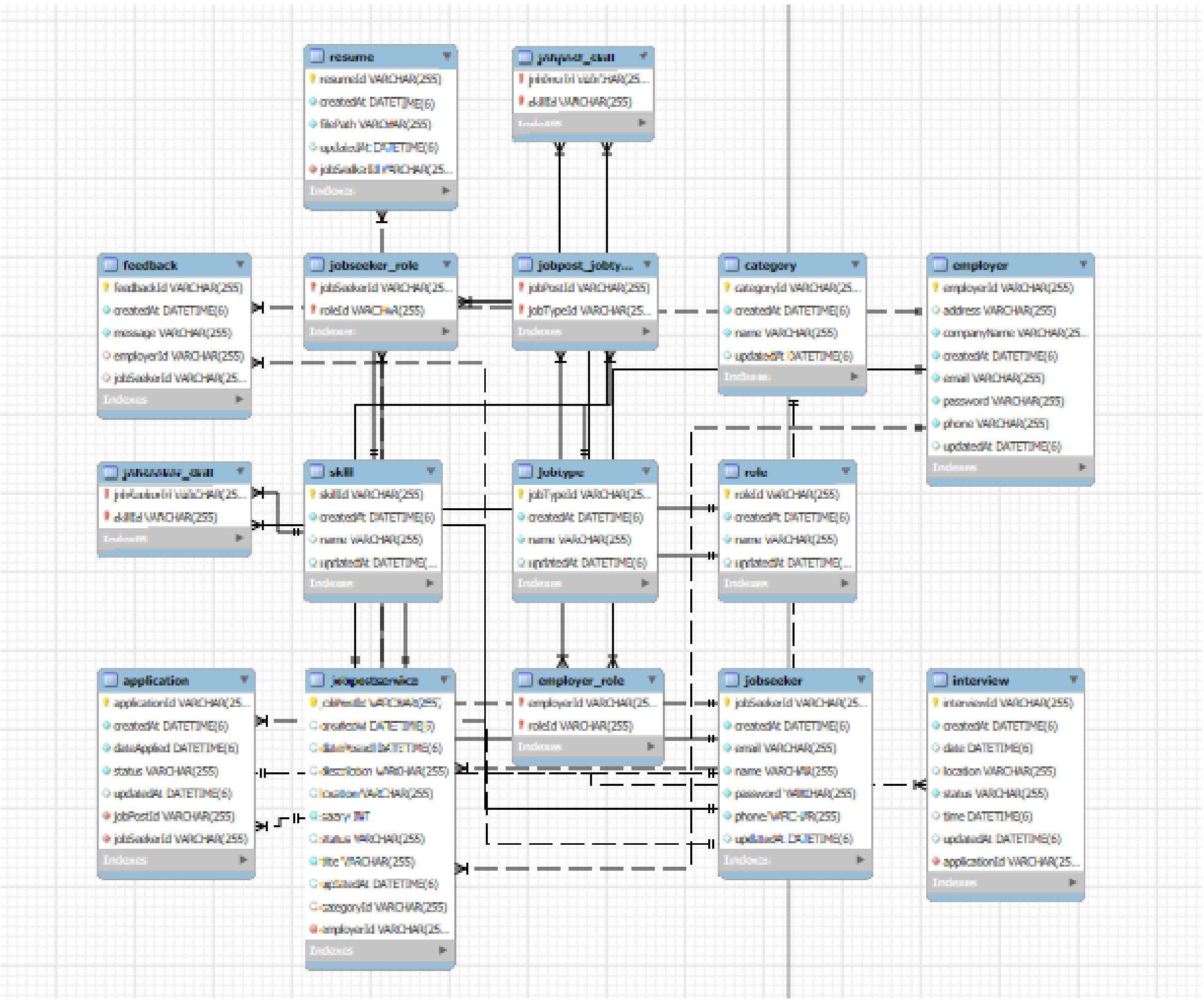
13. **Employer → Role**

- Many-to-Many: Each Employer can have multiple Roles.

14. **JobSeeker → Role**

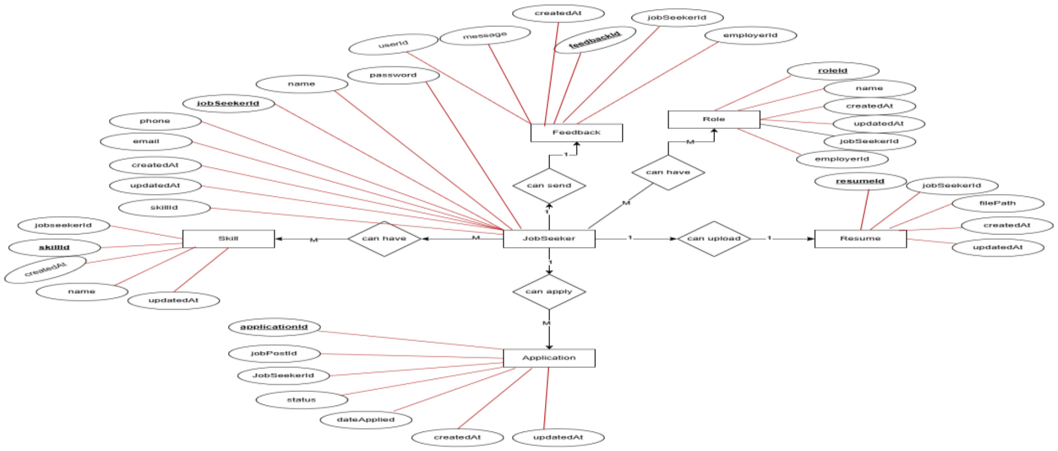
- Many-to-Many: Each JobSeeker can have multiple Roles.

6. Database Design

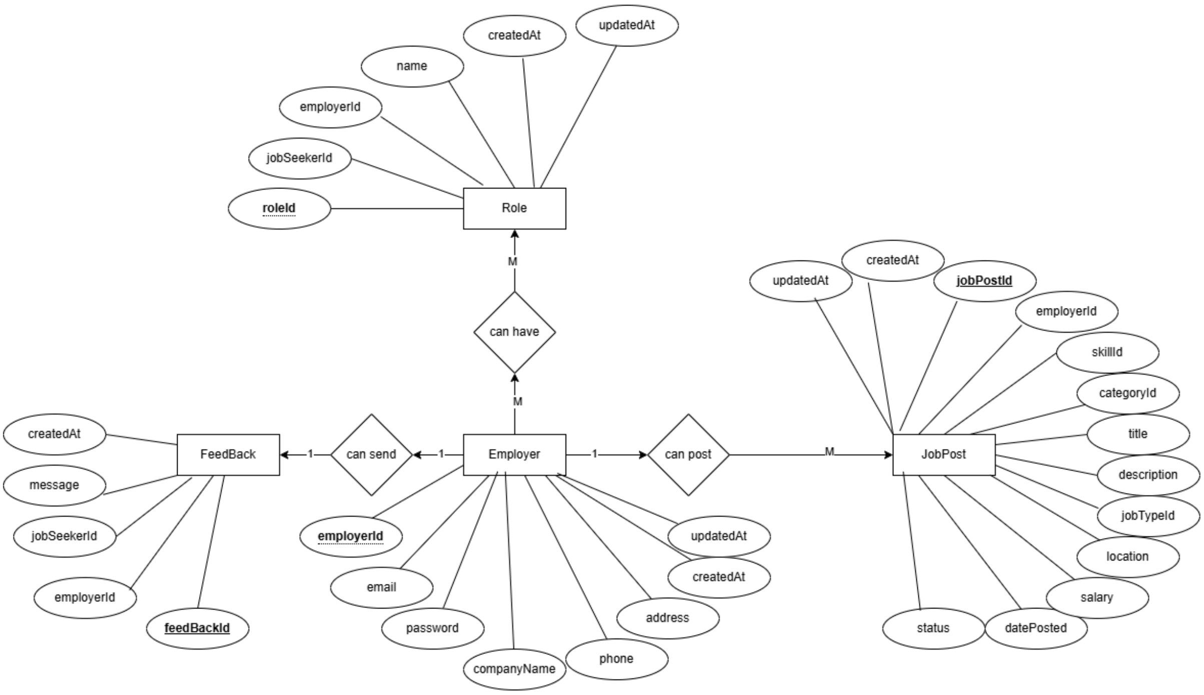


7. ER DIAGRAM

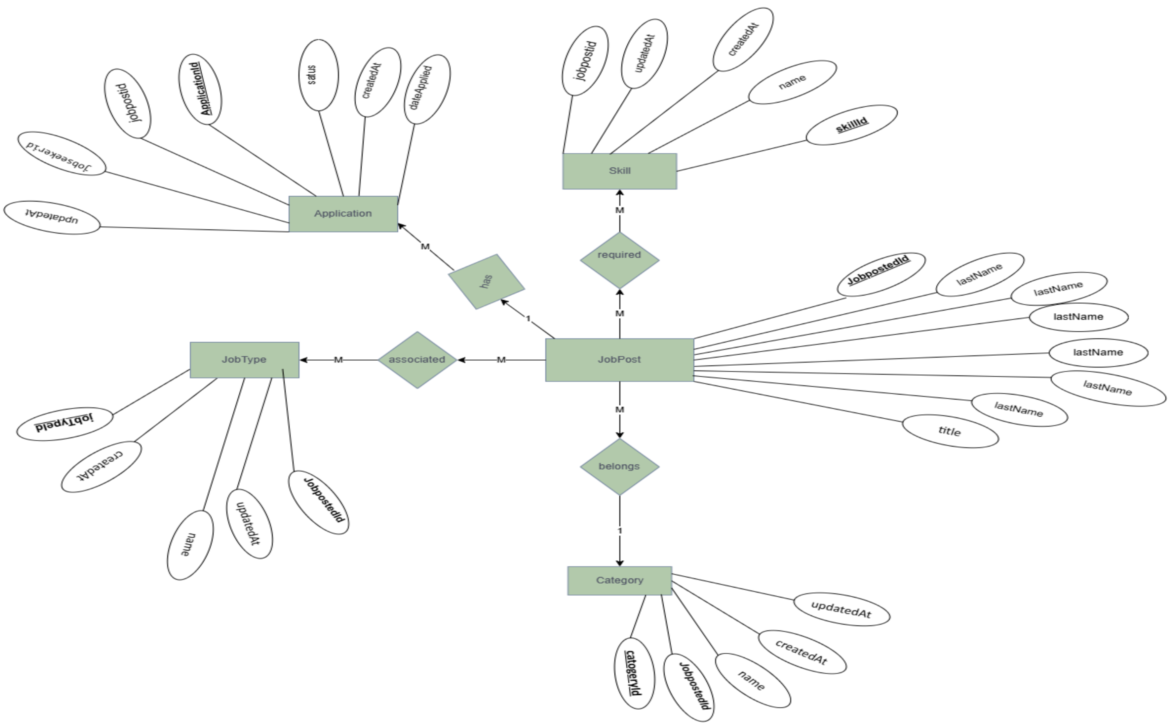
1. **JobSeeker** → Resume, Feedback, Application, Skill, Role



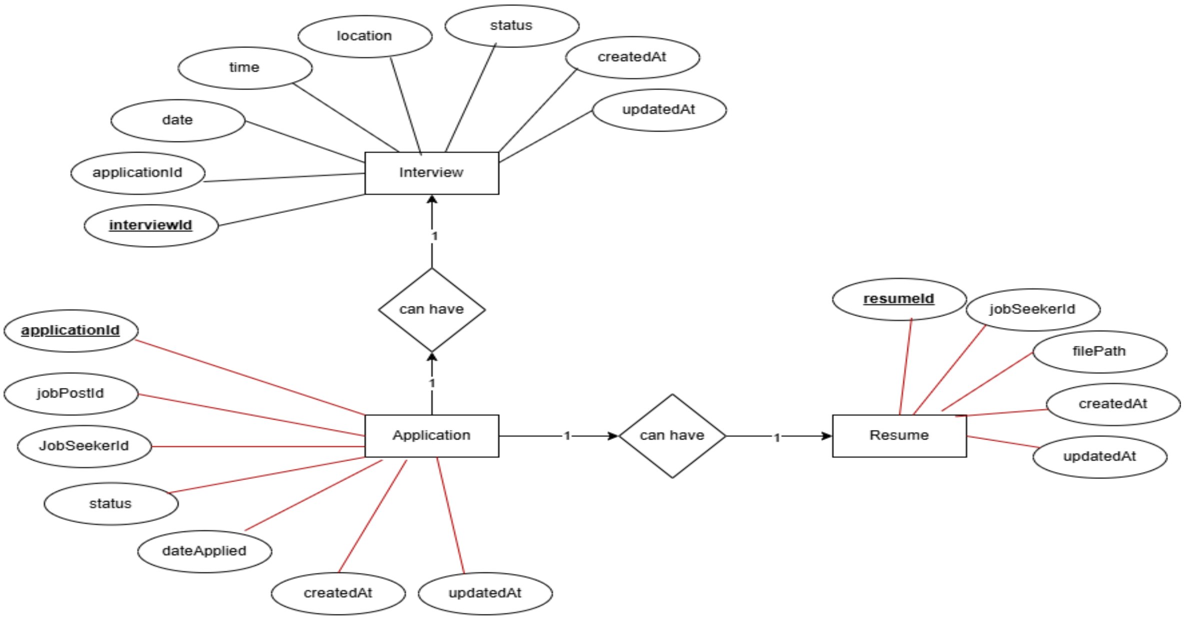
2. **Employer** → Feedback, JobPost, Role



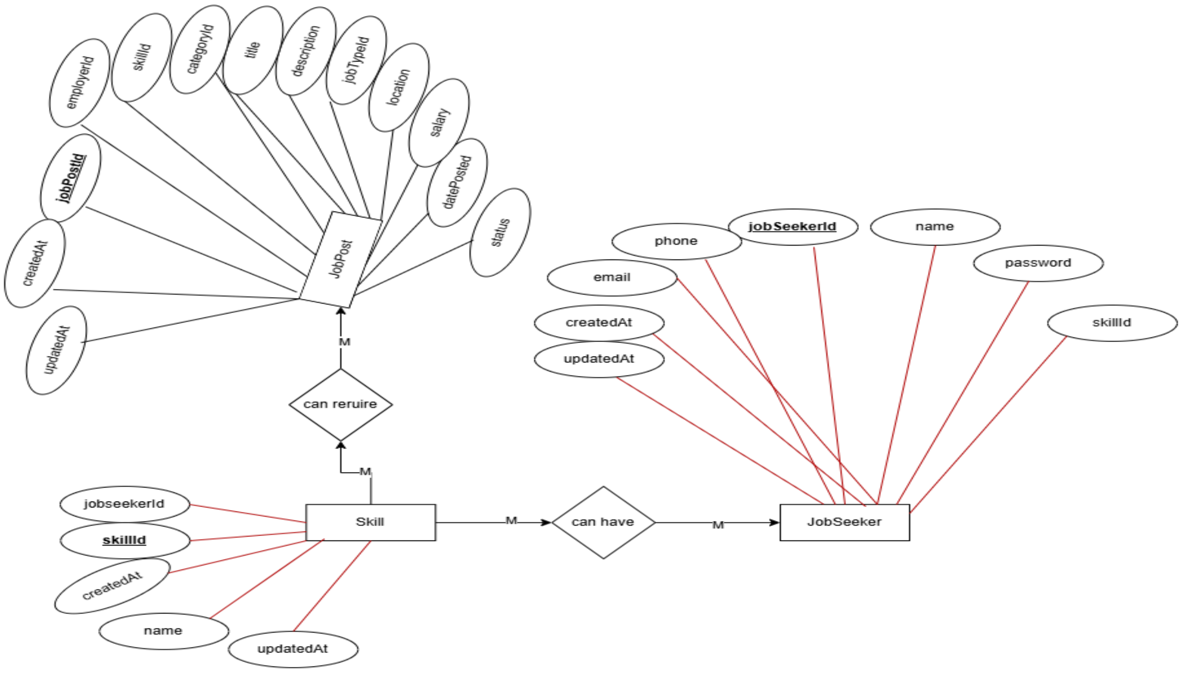
3. Job Post → **Employer, Skill, Category, JobType**



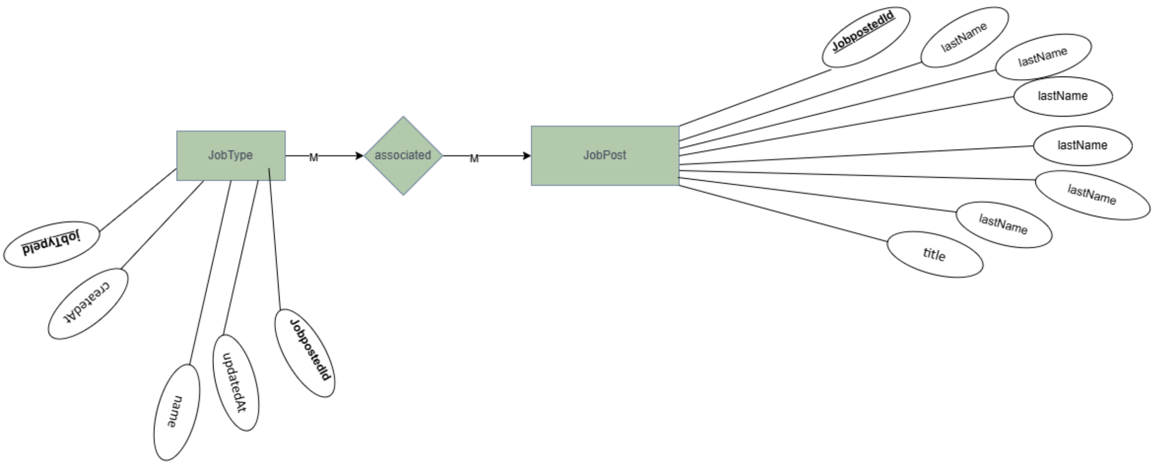
4. Application → Resume, Interview



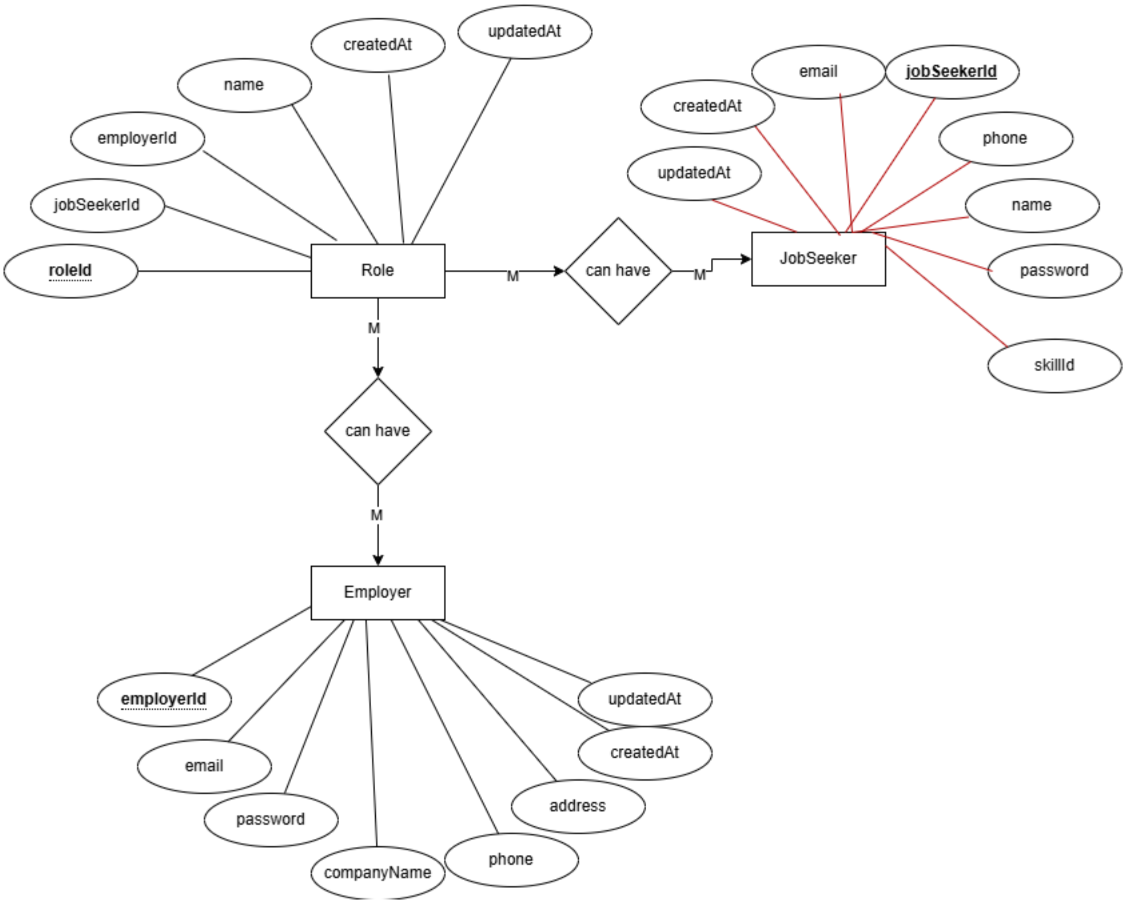
5. Skill → JobSeeker, JobPost.



6. JobType → JobPost



7. Role → JobSeeker, Employer



8. Technologies Used

The project utilizes a combination of modern development tools, programming languages, and frameworks to ensure efficiency, scalability, and maintainability. Below is a detailed explanation of the technologies used:

1. Integrated Development Environment (IDE): Eclipse

Eclipse is a widely used open-source IDE that provides a robust development environment for Java applications. It offers several advantages:

- **Code Assistance:** Features like code completion, syntax highlighting, and error detection help improve productivity.
- **Debugging Tools:** Provides an integrated debugger for identifying and fixing issues efficiently.
- **Plugin Support:** Supports a vast range of plugins for various frameworks, database tools, and version control systems like Git.
- **Refactoring and Code Management:** Helps in better code structuring and optimization.

2. Database: MySQL

MySQL is an open-source relational database management system (RDBMS) known for its reliability, scalability, and performance. It is used to store and manage application data efficiently. Key features include:

- **Structured Query Language (SQL):** Provides a powerful way to query and manipulate data.
- **ACID Compliance:** Ensures data integrity and consistency.
- **Scalability:** Supports large datasets and high transaction volumes.
- **Security Features:** Offers authentication, authorization, and encryption options.
- **Integration:** Easily integrates with Java applications via JDBC (Java Database Connectivity) or ORM tools like Hibernate.

3. Programming Language: Java

Java is the core programming language used in the project, chosen for its robustness, platform independence, and object-oriented nature. Some of the advantages include:

- **Write Once, Run Anywhere (WORA):** Java applications can run on different platforms without modification.
- **Strong Memory Management:** Uses automatic garbage collection to optimize performance.
- **Multithreading Support:** Enables efficient handling of concurrent processes.
- **Rich API:** Provides extensive libraries for networking, file handling, and data structures.

4. Hibernate (ORM Framework)

Hibernate is a powerful Object-Relational Mapping (ORM) framework that simplifies database interactions in Java applications. It provides:

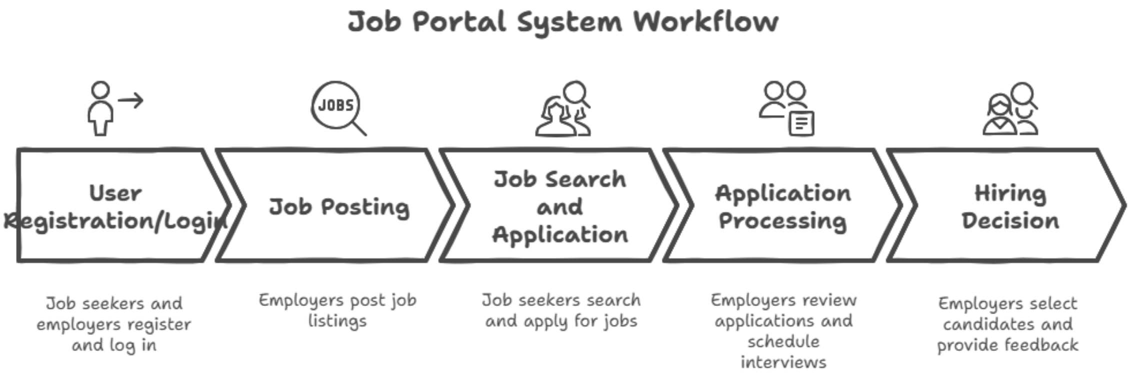
- **Automatic SQL Generation:** Reduces the need for writing complex SQL queries.
- **Transparent Data Persistence:** Allows Java objects to be seamlessly stored and retrieved from the database.
- **HQL (Hibernate Query Language):** Provides an alternative to SQL with better abstraction and database portability.
- **Lazy Loading and Caching:** Improves application performance by optimizing data retrieval.
- **Transaction Management:** Supports ACID transactions and integrates with Java EE frameworks.

- By combining **Eclipse, MySQL, Java, and Hibernate**, this project ensures a **robust, scalable, and maintainable** architecture. The selected technologies provide a strong foundation for developing efficient and high-performance enterprise applications.

9. System Workflow

1. **User Registration/Login** – Job seekers and employers register and log in.
2. **Job Posting** – Employers post job listings.

3. **Job Search and Application** – Job seekers search and apply.
4. **Application Processing** – Employers review applications and schedule interviews.
5. **Hiring Decision** – Employers select candidates and provide feedback.


Job Portal System Workflow
User Registration/Login
Job seekers and employers register and log in
JOBS
Job Posting
Employers post job listings
Job Search and Application
Job seekers search and apply for jobs
Application Processing
Employers review applications and schedule interviews
Hiring Decision
Employers select candidates and provide feedback

8. Conclusion

The Job Portal System provides an efficient platform for job seekers and employers to interact, reducing the complexity of job searching and hiring processes. With a well-structured database and an intuitive interface, the system enhances user experience and streamlines recruitment activities.