Internship Project Report

On

Java Full Stack

Internship & Job Management Web Portal

Start Date: 16th-May-2025

End Date: 16th-Aug-2025

Submitted By:

Name: Anand Patel

Contact: +91 8081867443

Email: anandkp0852@gmail.com

GitHub: https://github.com/anandptl

Abstract

The project “Internship & Job Management Web Portal” is a web-based application developed to help students, job seekers, and recruiters connect on a single platform. The main aim of this project is to make the process of finding jobs or internships easier for students and allow recruiters to post job openings and hire suitable candidates in a simple way.

In this system, students can create their profile, upload their resume, and apply for jobs or internships. Recruiters can register their company, post job opportunities, and check student applications. Admins manage the entire system, verify users, and ensure smooth functioning of the portal.

The project is developed using HTML, CSS, JavaScript, and JSP for the frontend, Spring Boot for the backend, and MySQL for the database. Tools like Postman were used for testing APIs and GitHub for version control.

This project not only helps in bridging the gap between students and recruiters but also provides a digital solution for recruitment and internship management. It is designed to be simple, user-friendly, and scalable so that new features can be added in the future.

Table of Contents

1. Introduction

My project is titled **“Internship & Job Management Web Portal”**. The main idea of this project is to create a simple platform where students and job seekers can search and apply for jobs or internships, and recruiters can easily post job opportunities and hire candidates.

In today’s time, most companies hire online, and students also prefer applying through digital platforms. But there is no single portal that helps students build their profile, upload their resume, and directly apply for jobs or internships in one place. This project tries to solve that problem.

In this portal:

- **Students** can register, create their profile, upload their resume, and apply for jobs/internships.
- **Recruiters** can log in, post jobs, and check student applications.
- **Admin** has full control to manage both students and recruiters and keep the platform running smoothly.

For development, I used **HTML, CSS, JavaScript, and JSP** for the frontend, **Spring Boot** for the backend, and **MySQL** as the database. I also used **Postman** for API testing and **GitHub** for version control.

This project gave me a chance to learn how a real web application works from both the user’s and developer’s point of view.

2. Project Overview

The Internship & Job Management Web Portal is a web application where students, job seekers, and recruiters can interact on one platform. The main goal of this portal is to make the hiring process easier for both students and companies.

Students often face problems in finding the right job or internship and recruiters also face difficulties in reaching the right candidates. This project works as a bridge between them.

In this project:

- **Students**/Job Seekers can create their profile, add qualifications, upload resume, and apply for different jobs or internships. They can also track the status of their applications.
- **Recruiters** can register their company, post job or internship opportunities, and manage the applications they receive.
- **Admins** manage the whole system. They can add, update, or remove users and make sure the portal is safe and fair.

This project is built using simple but powerful technologies like HTML, CSS, JavaScript, and JSP for the frontend, Spring Boot for backend processing, and MySQL for the database. I also used Postman to test APIs and GitHub for code management.

3. Purpose

1. **Bridge the gap** between students seeking internships/jobs and companies offering them.
2. **Help students** build professional profiles, upload resumes, and track applications.
3. **Enable recruiters** to post job/internship opportunities, review applications, and shortlist candidates.
4. **Provide admins** with tools to monitor users, moderate content, and analyze system usage.
5. **Simplify the recruitment workflow** through one centralized portal instead of multiple offline or scattered processes.
6. **Enhance efficiency** using modern web technologies, ensuring scalability, maintainability, and real-time data access.

Many times, students don't have a proper platform where they can show their skills, upload their resume, and directly apply for jobs. On the other side, companies also face difficulty in reaching students quickly. This portal provides a **common platform** where both can connect.

4. Scope of the Project

The scope of my project "Internship & Job Management Web Portal" is to provide all the main features that are required in a simple and effective job/internship portal. It focuses on students, recruiters, and admin roles.

For Students / Job Seekers:

- Create and update their profile.
- Upload resume and add important details like skills, qualifications, experience, etc.
- Search and apply for jobs or internships.
- Track the status of their applications (applied, shortlisted, selected, or rejected).

For Recruiters:

- Register their company and log in to the system.
- Post new jobs or internships with details like role, skills needed, salary, location, etc.
- View applications from students and shortlist or reject candidates.

For Admins:

- Manage users (students and recruiters).
- Check authenticity of job postings.
- Keep the system secure and running properly.

Technical Scope:

- **Frontend**: HTML, CSS, JavaScript, JSP
- **Backend**: Spring Boot (Java)
- **Database**: MySQL
- **Tools**: Postman (for testing APIs) and GitHub (for version control)

Limitations (Out of Scope for now):

- No mobile app version (only web-based).
- No AI-based job recommendations (can be added in future).

5. Definitions

To understand the project clearly, here are some important terms used in the Internship & Job Management Web Portal:

- **Portal**: A website or platform where different users (students, recruiters, admins) can log in and perform their tasks.
- **User**: Any person who uses the portal. In this project, users can be Students/Job Seekers, Recruiters, or Admins.
- **Student / Job Seeker**: A person who is looking for a job or internship. They can create their profile, upload their resume, and apply for opportunities.
- **Recruiter:** A company representative or employer who posts jobs or internships and reviews student applications.
- **Admin:** The controller of the portal. Admin has the highest power and can manage all users, verify job postings, and maintain the system.
- **Job Posting:** A vacancy created by a recruiter which contains details like job title, description, skills required, salary, and location.
- **Application:** When a student applies for a job or internship, it is called an application. The status of the application can be Applied, Shortlisted, Selected, or Rejected.
- **Resume Upload:** A feature where students can upload their resume (PDF/DOCX) so recruiters can view it while checking applications.
- **Authentication**: The process of verifying a user's identity during login using email/username and password.
- **Role-Based Access**: Different users (student, recruiter, admin) get different permissions according to their role.
- **Database**: A system (MySQL in this project) where all information like users, student details, recruiter details, and job postings are stored.
- **Postman**: A tool used to test backend APIs and check if they are working properly.
- **GitHub**: A platform used to save and manage project source code and track changes.

6. Specific Requirements

In this section, I am listing the main requirements of my project "Internship & Job Management Web Portal". These requirements are divided into two parts: Functional Requirements (what the system should do) and Non-Functional Requirements (how the system should work).

6.1. Functional Requirements

These are the features that my system must have:

- **User Registration and Login**
  - Students, recruiters, and admins can create accounts and log in.
  - Each user will get access according to their role.
- **Student Features**
  - Students can create and update their profile.
  - They can upload their resume and add skills, education, and experience.
  - They can search and apply for jobs or internships.
  - They can check the status of their applications (applied, shortlisted, selected, rejected).
- **Recruiter Features**
  - Recruiters can register their company.
  - They can post new job or internship openings.
  - They can view student applications and shortlist or reject them.
- **Admin Features**
  - Admins can manage both students and recruiters.
  - Admins can check if job postings are genuine.
  - Admins can remove fake or unwanted accounts.

6.2. Non-Functional Requirements

These are not about features but about how the system should work:

- **Performance**: The portal should work smoothly for many users at the same time.
- **Security**: Passwords should be stored safely, and only the right person should be able to log in.
- **Usability**: The system should be easy to use for all types of users.
- **Reliability**: The portal should be available most of the time without errors.
- **Scalability**: In the future, new features like mobile app or AI-based job suggestions can be added.
- **Compatibility**: The system should run on all popular browsers like Chrome, Firefox, and Edge.

7. System Design

The Internship & Job Management Web Portal is designed in a way that it is easy to use, well organized, and can be expanded in the future. The design part mainly explains how the system works internally and how data flows between users, backend, and the database.

7.1 Architecture of the System

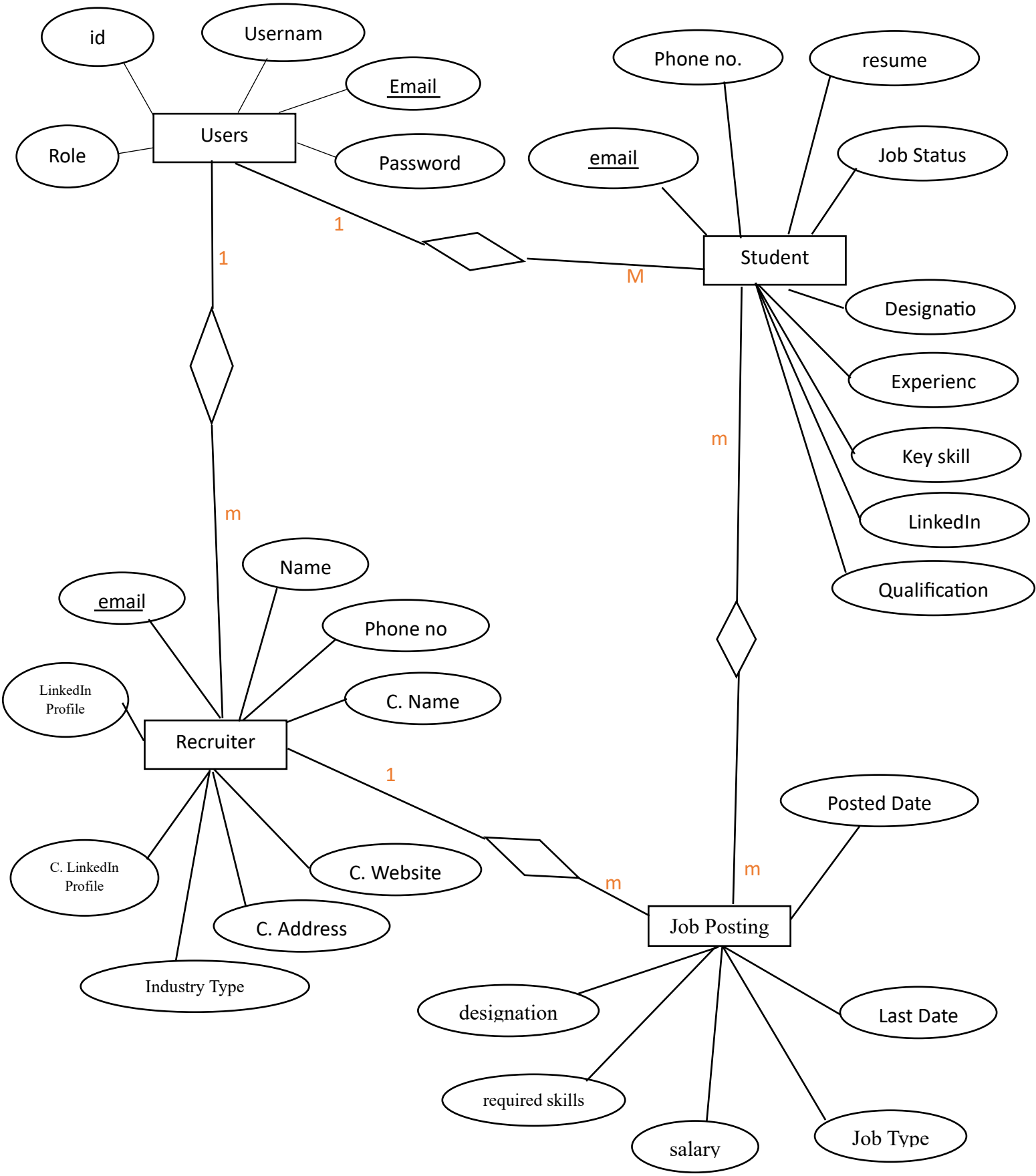
The system is divided into three layers:

- **Frontend (Presentation Layer):**
  - Built using HTML, CSS, JavaScript, and JSP.
  - This is where students, recruiters, and admins interact with the system.
- **Backend (Business Logic Layer):**
  - Built using Spring Boot.
  - It handles requests, applies logic, and connects frontend with the database.
- **Database (Data Layer):**
  - MySQL is used to store all data like users, student details, recruiter details, and job postings.

7.2 Entity Relationship Diagram

The database has four main tables:

1. **User Table**: Stores username, email, password, and role (student, recruiter, admin).
2. **Student Table**: Stores student details like name, phone, email, resume, skills, qualification, experience, job status, expected salary, and links (GitHub/LinkedIn).
3. **Recruiter Table**: Stores recruiter and company details like Full Name, Emai, Mobile no, Company Name, Company Website, Company Address, Industry Type, Company LinkedIn profile.
4. **Job Posting Table**: Stores job information like designation, required skills, salary, job type, location, etc.

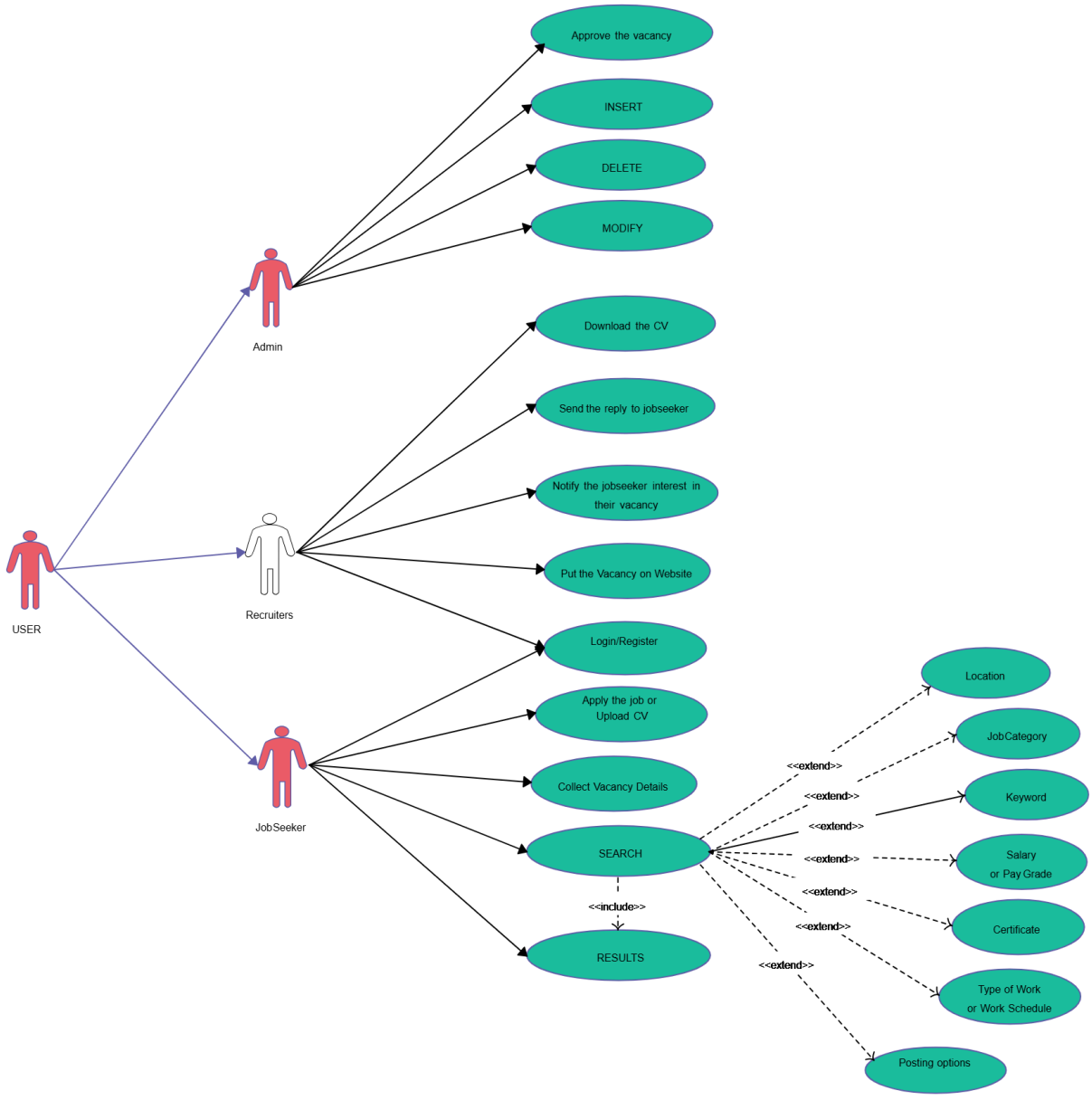


7.2 Entity Relationship Diagram

7.3 Use Case Diagram

- **Actors**: Student, Recruiter, Admin.
- **Student Use Cases**: Register, login, manage profile, upload resume, apply for jobs, check application status.
- **Recruiter Use Cases**: Register, login, post jobs, view applications, shortlist/reject candidates.
- **Admin Use Cases:** Manage users, verify job postings, control the system.

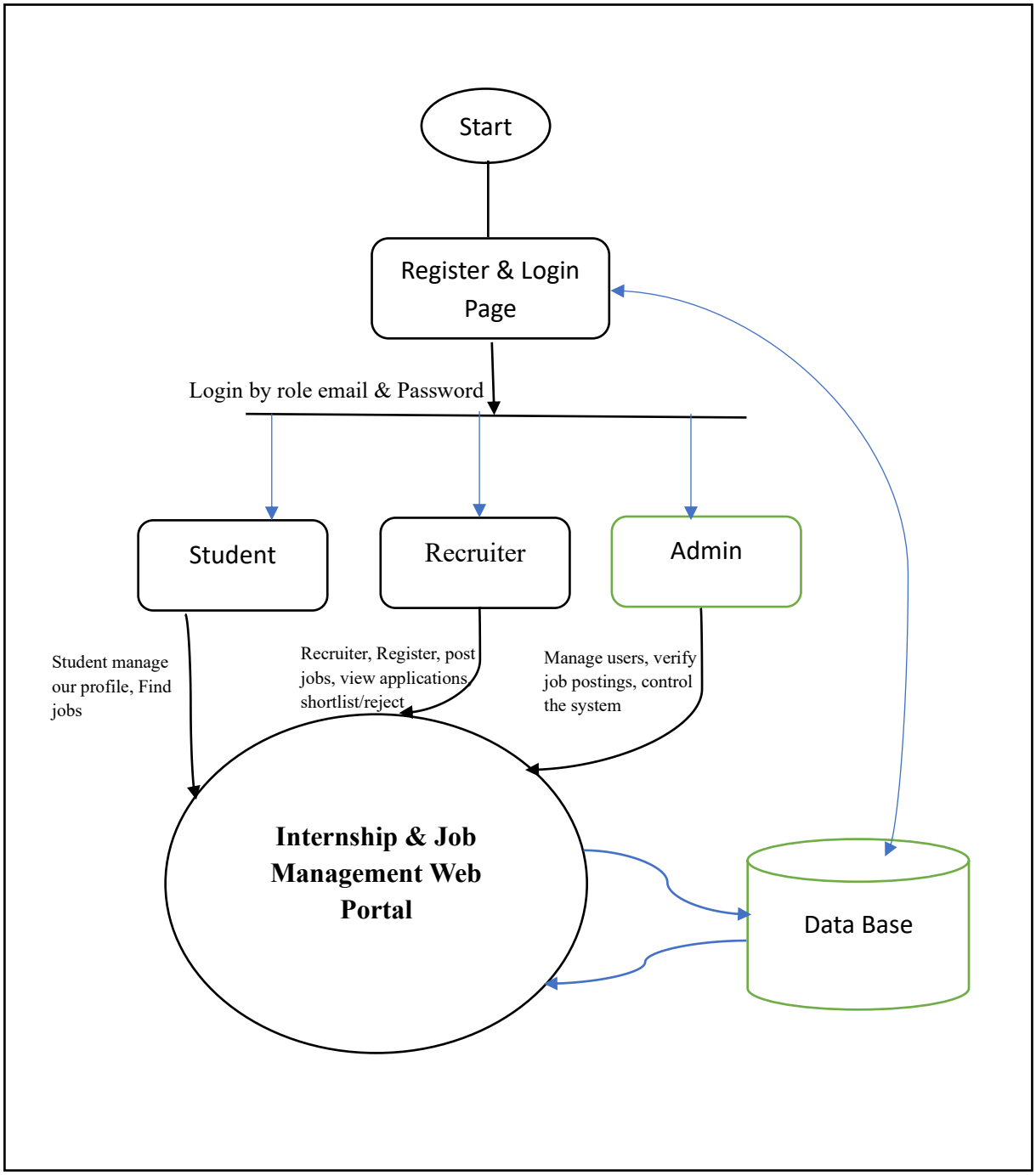
(In the actual report, we can also insert the Use Case diagram here)



Use Case Diagram

7.4 Data Flow (Simple View)

- A student logs in → fills profile → applies for a job → data goes to the database → recruiter sees the application → recruiter updates status → student gets notification.
- An admin can see both student and recruiter data and has control to approve or block accounts.

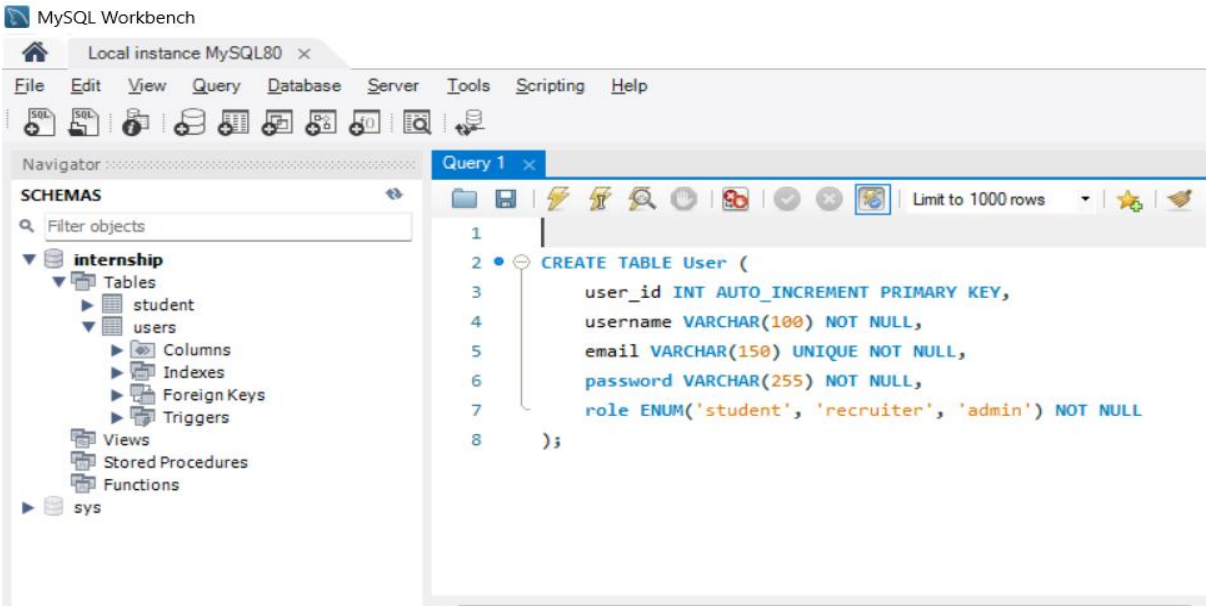


8. Database Design

The Internship & Job Management Web Portal uses MySQL as the database to store all the information about users, students, recruiters, and job postings. The database is designed in a structured way so that data is stored safely and can be easily retrieved when needed.

8.1. User Table

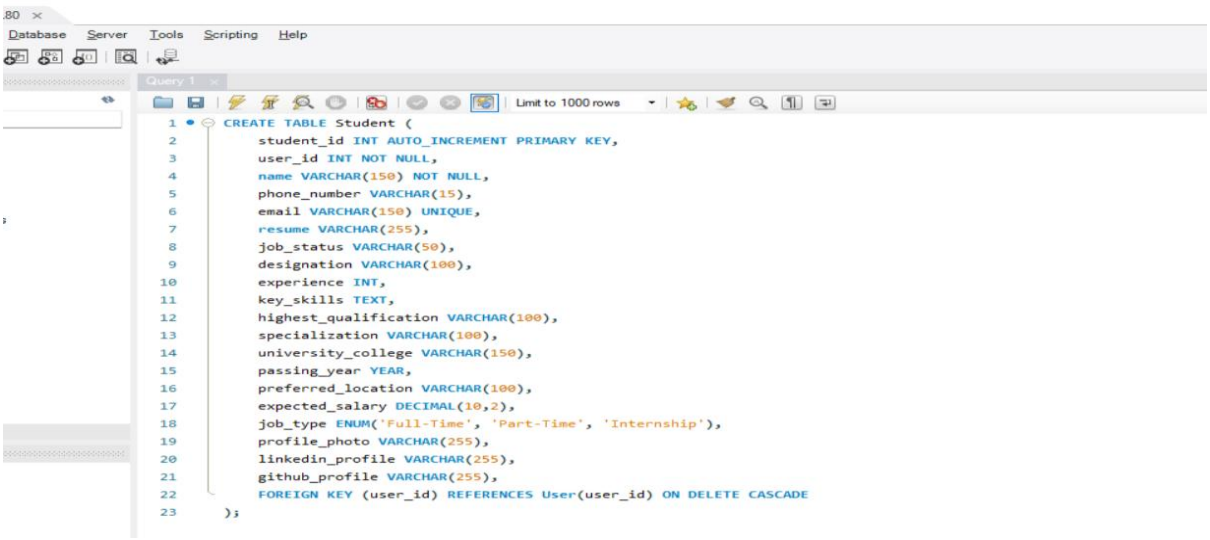
This table stores the login information of all users (students, recruiters, and admins). Each user has a role that defines what they can do in the system.



User Table

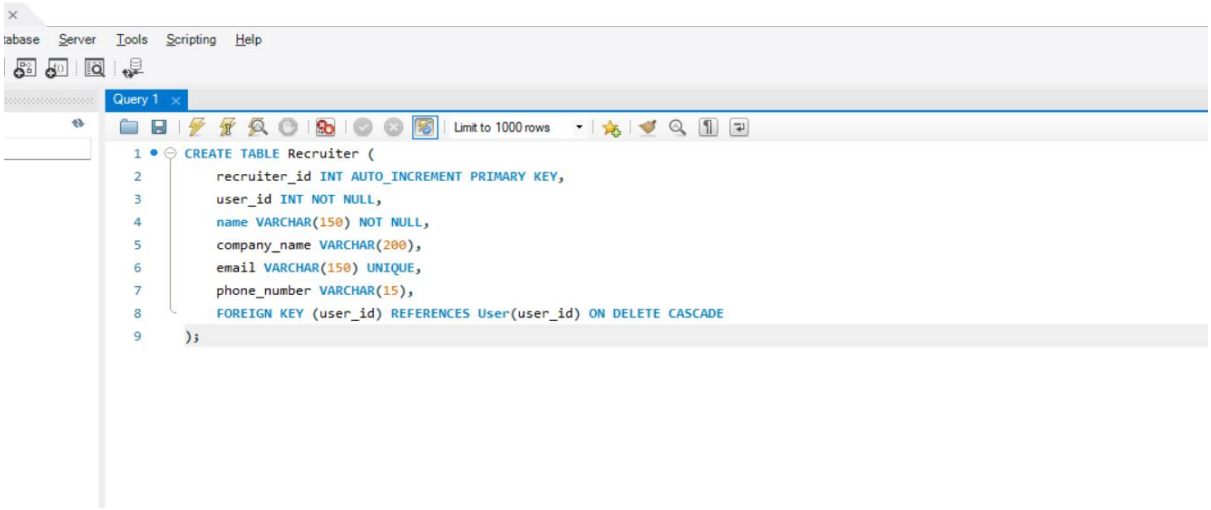
8.2. Student Table

This table stores complete details of students/job seekers. Each student is connected to a record in the User Table.



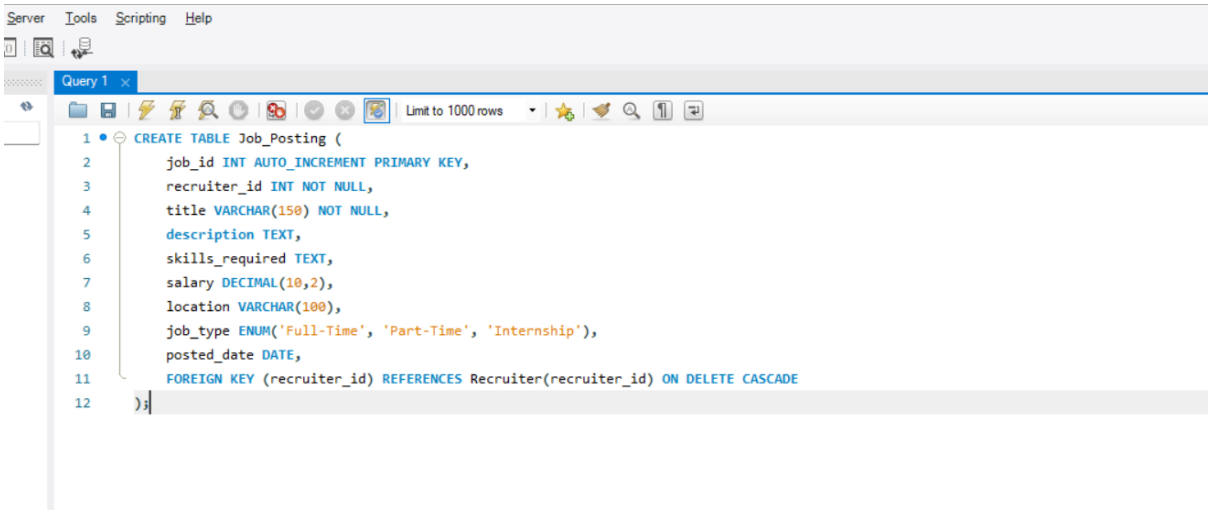
8.3. Recruiter Table

This table stores details of recruiters and their companies. Each recruiter is connected to the User Table.



8.4. Job Posting Table

This table stores details about the jobs and internships posted by recruiters. Each job is linked to the Recruiter Table.

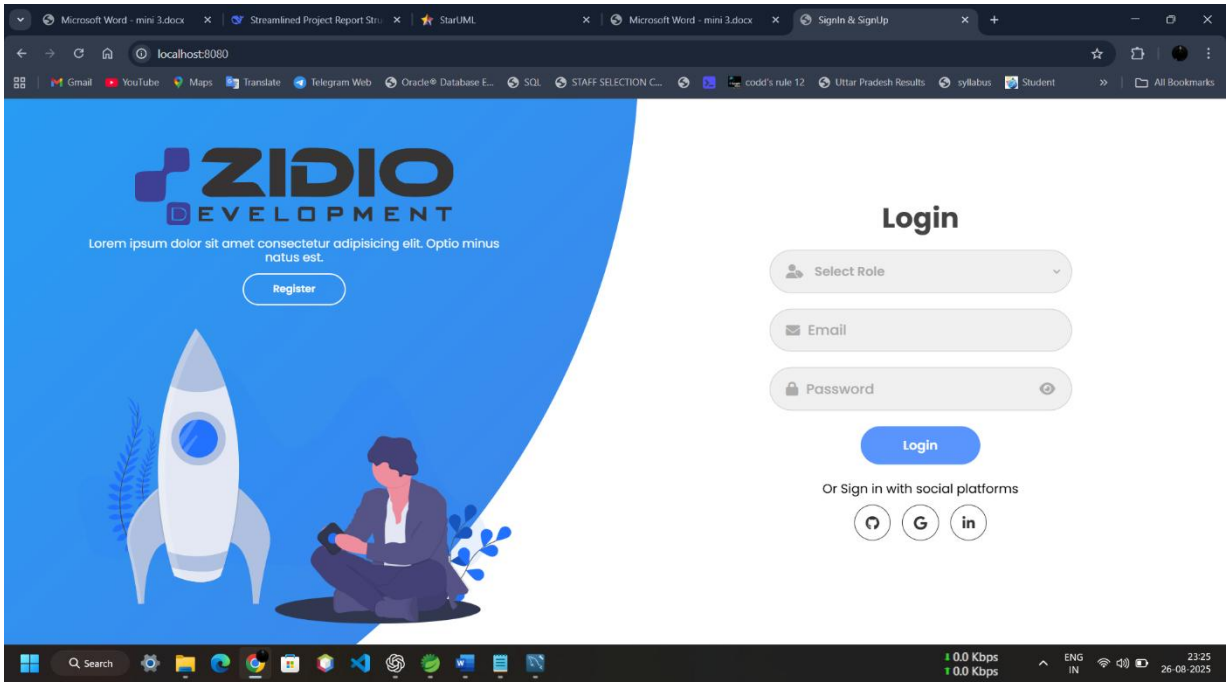


9. Implementation

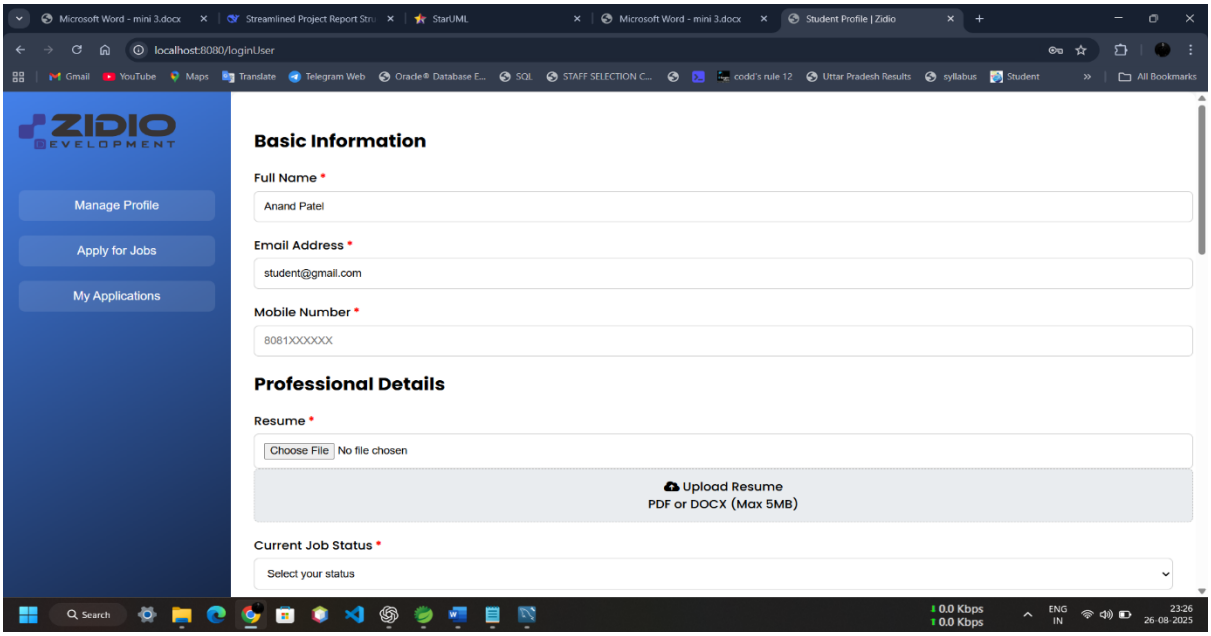
The implementation of my project “Internship & Job Management Web Portal” was done step by step using different technologies and tools. The project follows a simple layered structure where frontend, backend, and database are connected to each other.

9.1. Frontend Implementation

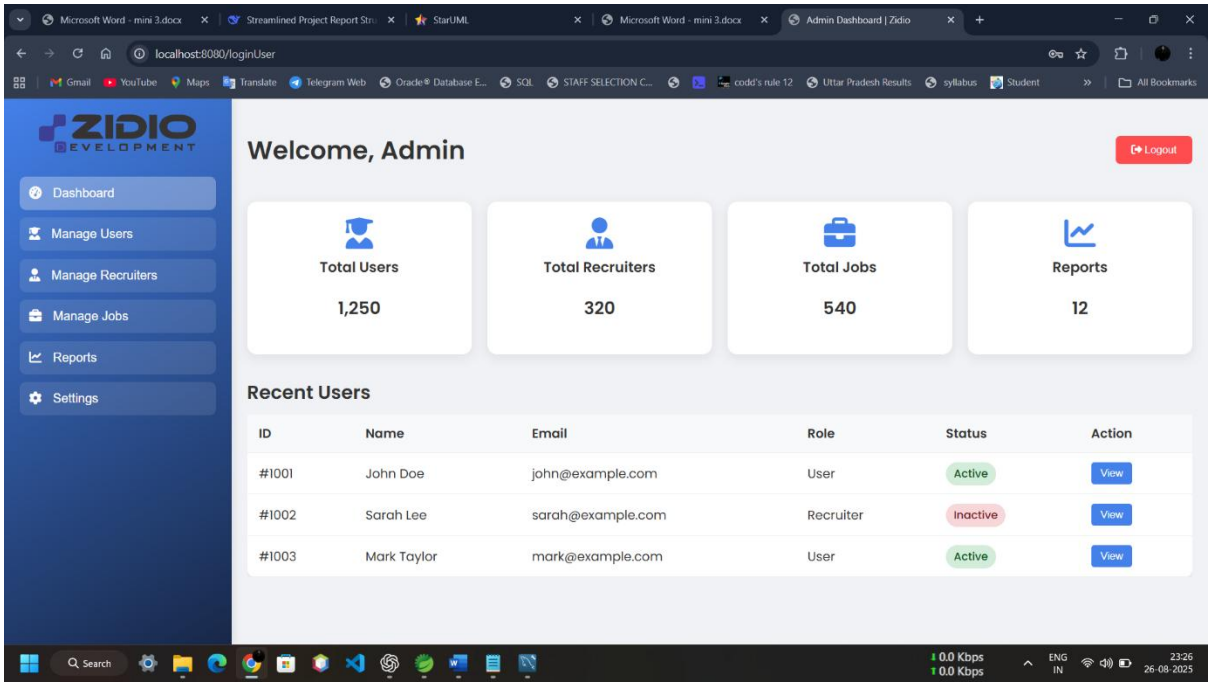
- I used HTML, CSS, and JavaScript to design the web pages.
- JSP (JavaServer Pages) was used to connect frontend pages with backend logic.
- The design is kept simple and user-friendly so that students and recruiters can easily use the system.
- Example pages: Login Page, Registration Page, Student Profile Page, Job Posting Page, and Admin Dashboard.



9.1.1 Login & Registration Page



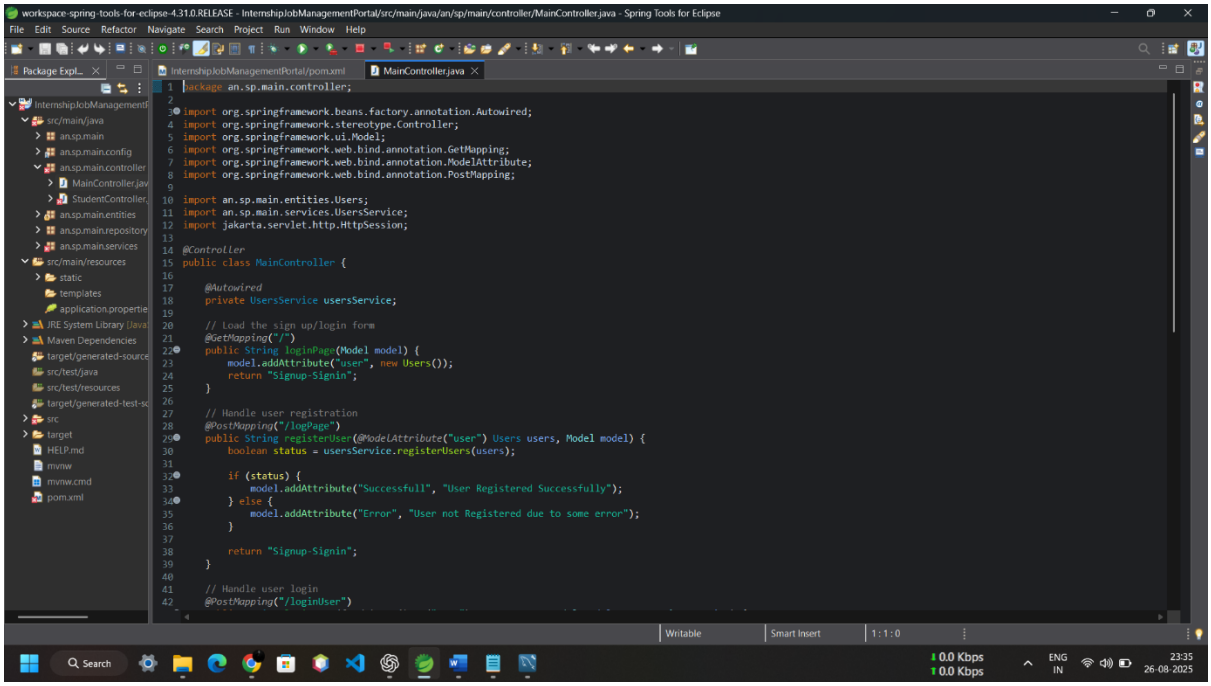
9.1.2 Student Page



9.1.3 Admin Page

9.2. Backend Implementation

- The backend is developed using Spring Boot (Java).
- I created controllers to handle user requests, services to implement business logic, and repositories (JPA) to interact with the database.
- Passwords are encrypted before storing in the database for better security.
- REST APIs were created for login, registration, posting jobs, applying for jobs, etc.



9.2.1 Controller Image

10 References

1. Spring Boot Documentation – Spring Framework Guides & API Reference
   https://spring.io/projects/spring-boot
2. MySQL Documentation – Relational Database Management System Reference
   https://dev.mysql.com/doc/
3. Java Documentation – Official Java SE API Documentation
   https://docs.oracle.com/en/java/
4. HTML, CSS, JavaScript Reference – MDN Web Docs
   https://developer.mozilla.org/
5. Used Icon - **Font Awesome 7**
   https://fontawesome.com/icons
6. JSP (Java Server Pages) Guide – Oracle Documentation
   https://docs.oracle.com/javaee/7/tutorial/servlets.htm#BNCIH
7. Postman Documentation – API Testing Tool Guide: https://learning.postman.com/