Hire Mitra (Job Portal)

Student Team:

Roll No.: 122CS0031
Name: Anurag Prajapati
Role: Developer/Tester

Roll No.: 122CS0067
Name: R. Krishnamurthi
Role: ML Engineer & Data Strategist

Roll No.: 122CS0081
Name: Aman Chourasia
Role: Developer/Tester

1. Introduction

This document provides a comprehensive overview of the Job Portal Web Application, detailing its features, functionalities, and system architecture. The application serves as a platform for job seekers and employers, enabling efficient job posting, application tracking, and recruitment management.

1.1 Purpose

The primary purpose of this job portal is to create a seamless connection between job seekers and employers. It aims to streamline the hiring process by offering an easy-to-use online platform for job searching, job posting, and application tracking.

1.2 Scope

The system is designed for:

- Job seekers who want to search and apply for job opportunities.

- Employers who need to post job listings and recruit candidates efficiently.

- Admins who monitor job postings, manage users, and maintain the system.

2. System Overview

The Job Portal Web Application is designed to facilitate job seekers in finding suitable employment opportunities while providing employers with a streamlined platform to post job vacancies and manage applicants. The system operates via a web-based interface accessible from any device.

2.1 System Modules

- **User Management:** Registration, login, and profile management.

- **Job Search & Application:** Search for jobs, apply online, and track application status.

- **Employer Dashboard:** Post jobs, manage applications, and communicate with candidates.

- **Admin Panel:** Monitor job postings, user activities, and system performance.

3. Features

3.1 Job Seeker Features

- User registration and login.

- Profile creation and resume upload.

- Job search and filtering by industry, location, and experience.

- Application submission and tracking.

- Job alert notifications.

- Bookmarking jobs for later applications.

- Viewing company profiles and employer ratings.

3.2 Employer Features

- Employer registration and login.

- Job posting and job listing management.

- Applicant tracking and shortlisting.

- Communication with candidates via messages or email.

- Subscription plans for premium job postings.

- Downloading applicant resumes and reviewing profiles.

3.3 Admin Features

- User management (job seekers and employers).

- Job posting moderation and approval.

- System analytics and reports.

- Handling user queries and complaints.

4. System Architecture

The application follows a client-server model with the following components:

4.1 Technology Stack

- **Frontend:** Developed using modern web technologies such as React.js with ShadCN UI.

- **Backend:** Implemented using Node.js with Express.

- **Database:** Uses PostgreSQL for data storage.

- **Authentication:** Uses JWT for secure user authentication.

- **Hosting:** Deployed on cloud platforms like AWS or Render.

4.2 Workflow

1. A job seeker registers and logs into the platform.

2. The job seeker creates a profile and uploads a resume.

3. Employers post job vacancies and review applications.

4. Job seekers apply for jobs and track application statuses.

5. Employers shortlist candidates and schedule interviews.

6. The admin moderates content and manages system performance.

5. User Interface Design

The user interface is designed for ease of use, featuring:

- A responsive layout suitable for both desktop and mobile users.

- Clear navigation for job search and application submission.

- Dashboard for employers to manage job postings and applications.

- Personalized job recommendations based on user profiles.

5.1 Wireframe and UI Elements

- Homepage with featured job listings.

- Job search bar with filtering options.

- Employer dashboard with job post management.

- User profile section with resume upload.

- UI components designed using ShadCN UI for a modern and consistent experience.

6. Security Measures

To ensure data security, the application includes:

- SSL encryption for secure data transmission.

- Password hashing and authentication via JWT.

- Role-based access control to restrict unauthorized actions.

- CAPTCHA verification to prevent bot registrations.

- Regular security audits and vulnerability assessments.

6.1 Data Privacy

- User consent management for data sharing.

- Two-factor authentication for added security.

7. Deployment and Maintenance

- The application is deployed using cloud services.

- Regular updates and bug fixes are implemented to ensure smooth operation.

- Backup mechanisms are in place to prevent data loss.

- Continuous monitoring for system performance and uptime.

7.1 Scalability Considerations

- Load balancing for handling high traffic.

- Database optimization for efficient querying.

- Auto-scaling on cloud infrastructure.

7.2 Maintenance Plan

- Monthly security patches and software updates.

- Customer support and feedback integration.

- System performance monitoring and optimization.

7.3 Performance Optimization

- Caching mechanisms for faster data retrieval.

- Database indexing for improved query performance.

- Content Delivery Network (CDN) for global accessibility.

8. Future Enhancements

The Job Portal Web Application aims to bridge the gap between job seekers and employers, offering a seamless and efficient recruitment process. Future enhancements may include:

- AI-driven job recommendations.

- Video interview capabilities.

- Integration with LinkedIn for importing professional profiles.

- Real-time chat feature between job seekers and employers.

- Mobile app version for Android and iOS.

8.1 Artificial Intelligence Integration

- Resume parsing and automatic job matching.

- Predictive analytics for job market trends.

- AI-powered chatbot for instant query resolution.

8.2 Mobile App Features

- Push notifications for job alerts.

- Voice search for job listings.

- Offline access to saved job listings.

8.3 Integration with Third-Party Services

- Payment gateways for premium job listings.

- API integration with job aggregators.

- Integration with social media for job sharing.

Workflow of website

01) Landing page:

The landing page serves as the first interaction point for users visiting the website. It
provides a visually appealing and informative layout, highlighting the platform's
purpose—connecting recruiters with job seekers. The page includes:
● A compelling tagline and brief introduction about Hire Mitra.
● Call-to-action buttons for recruiters and job seekers.
● Featured job listings and trending job categories.
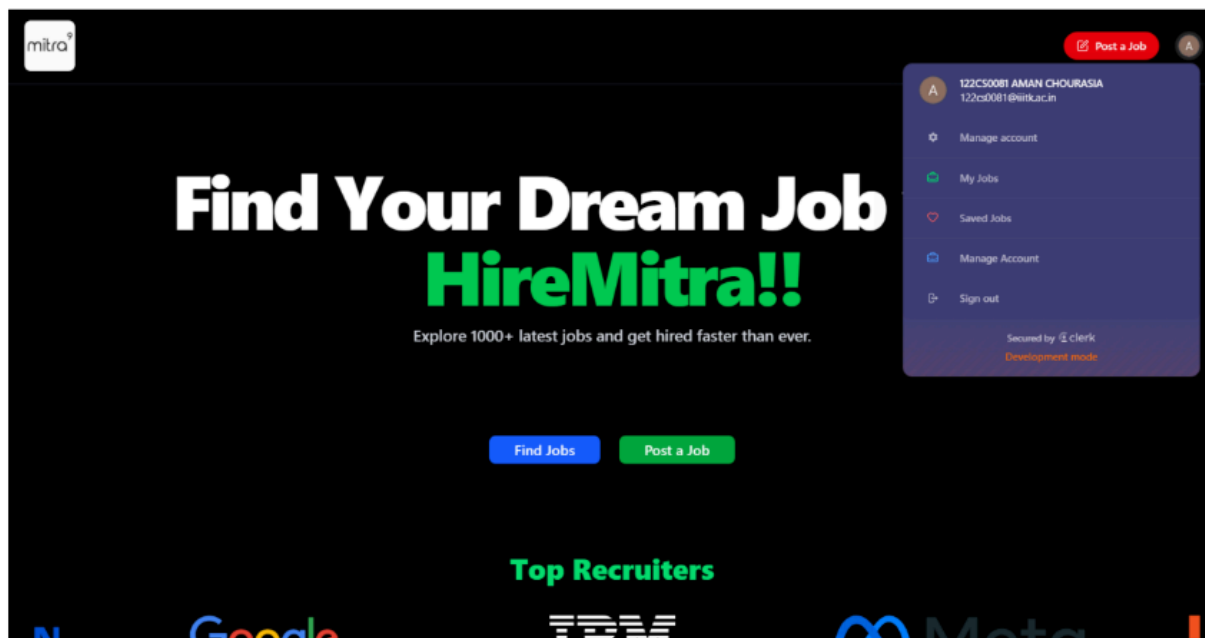
02) Login Page :

The login page allows users (both recruiters and candidates) to access their accounts
securely. It features:
● Email and password login fields.
● Google and LinkedIn and Microsoft login options.
● A "Sign Up" option for new users.
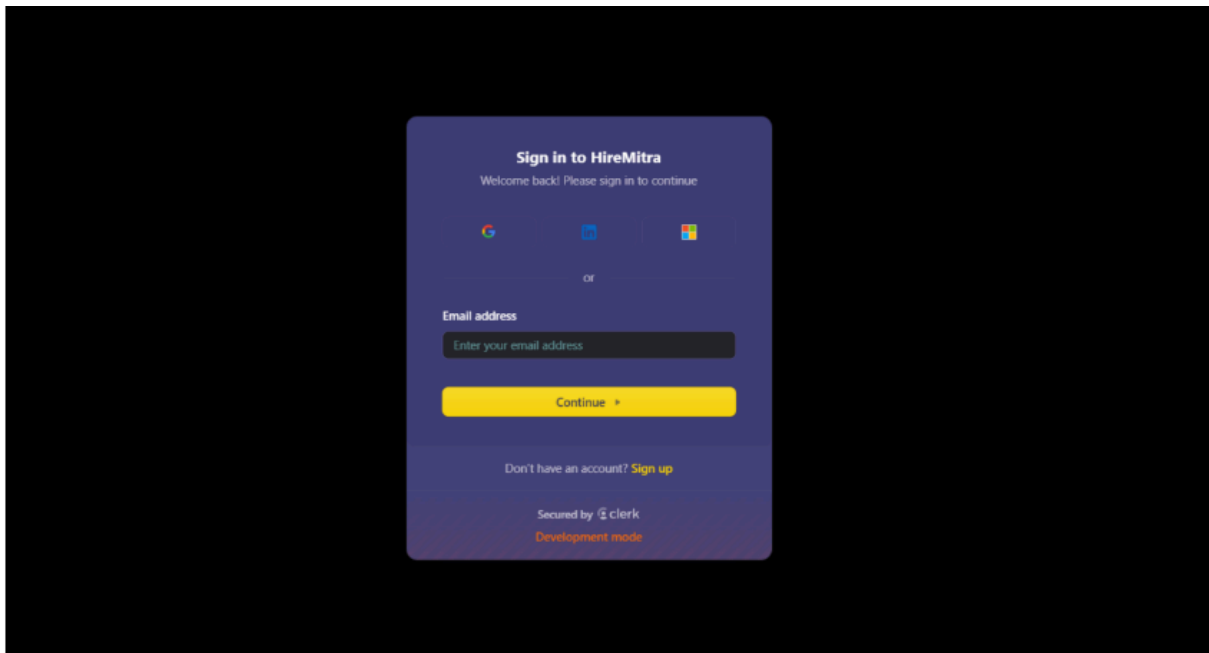
03) Recruiter side: (Post Job)

This section is designed for employers looking to hire candidates. It includes:

● A dashboard where recruiters can post job vacancies.

● Fields to enter job details such as title, description, salary, location, and required
qualifications.

● An option to manage and edit posted job listings.

04)Candidate side (Apply job)

This section is for job seekers looking for employment opportunities. It allows
candidates to:

● Browse job listings based on categories, locations, and skills.

● Apply for jobs by submitting their resumes and cover letters.

● Track application status and receive notifications about interview updates.

● Save favorite job listings for later.
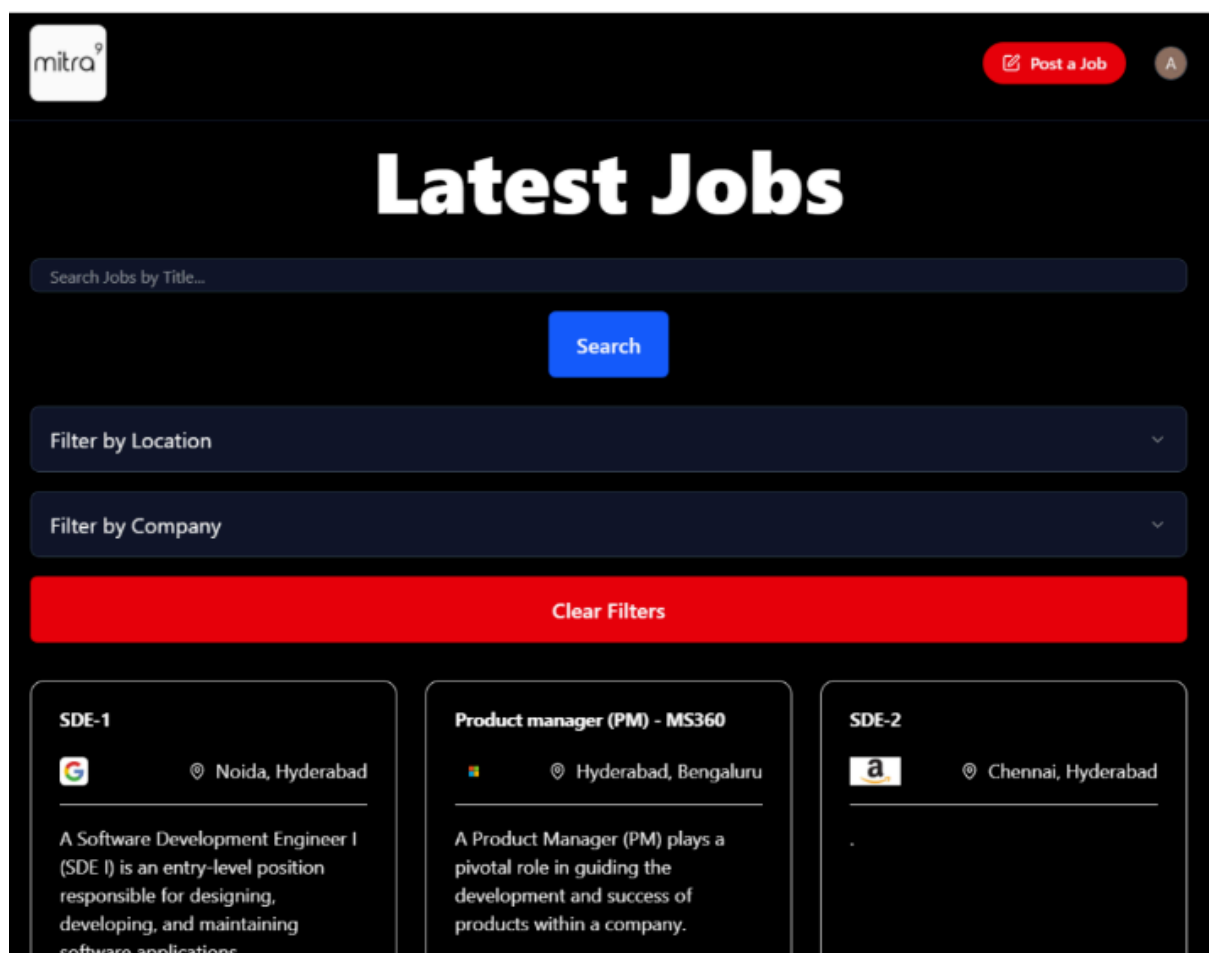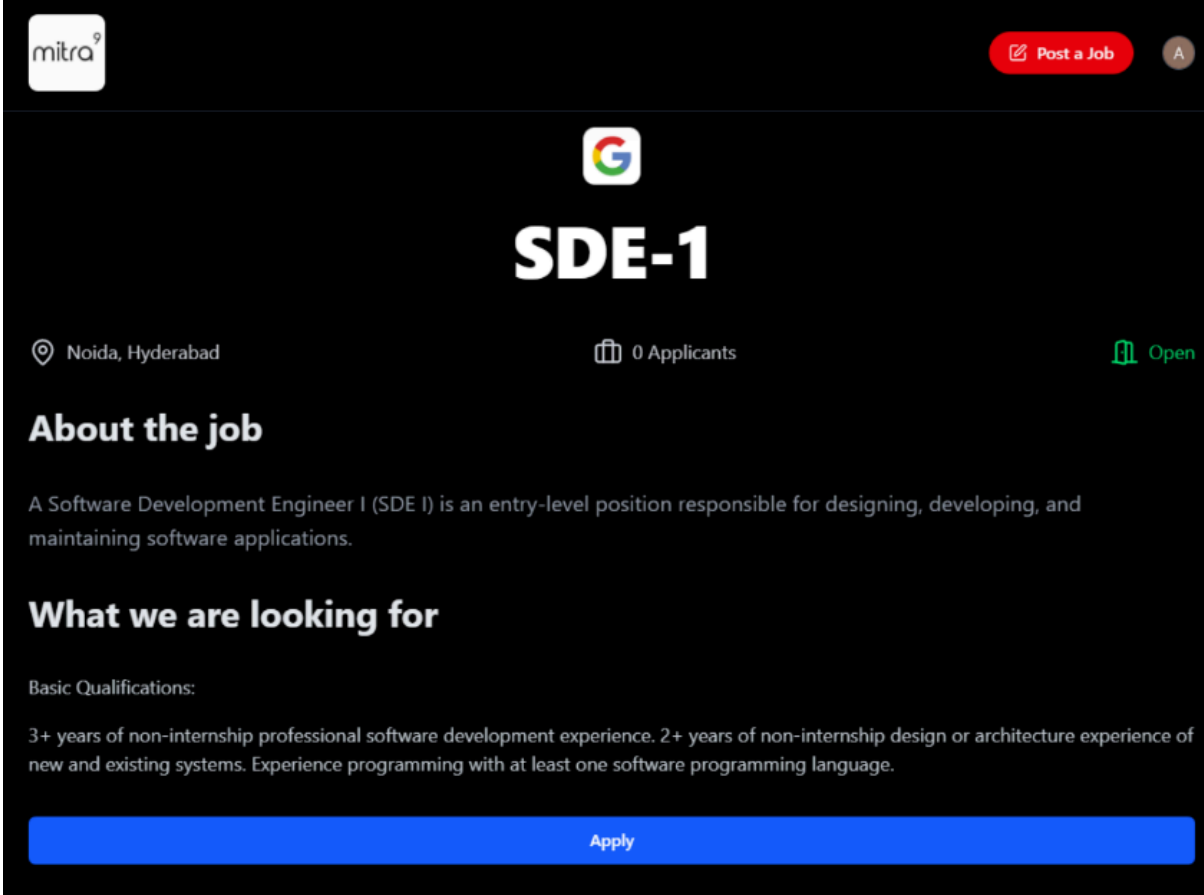
05)More details about the position:

Each job listing contains detailed information such as:

● Job title, description, and responsibilities.

● Required qualifications and skills.

- Work mode (remote/on-site/hybrid).
- Salary range and perks offered.
- Contact details of the recruiter for queries.

06) Resume Parsing:

07).Upload their Resume

08). Resume Analyzer
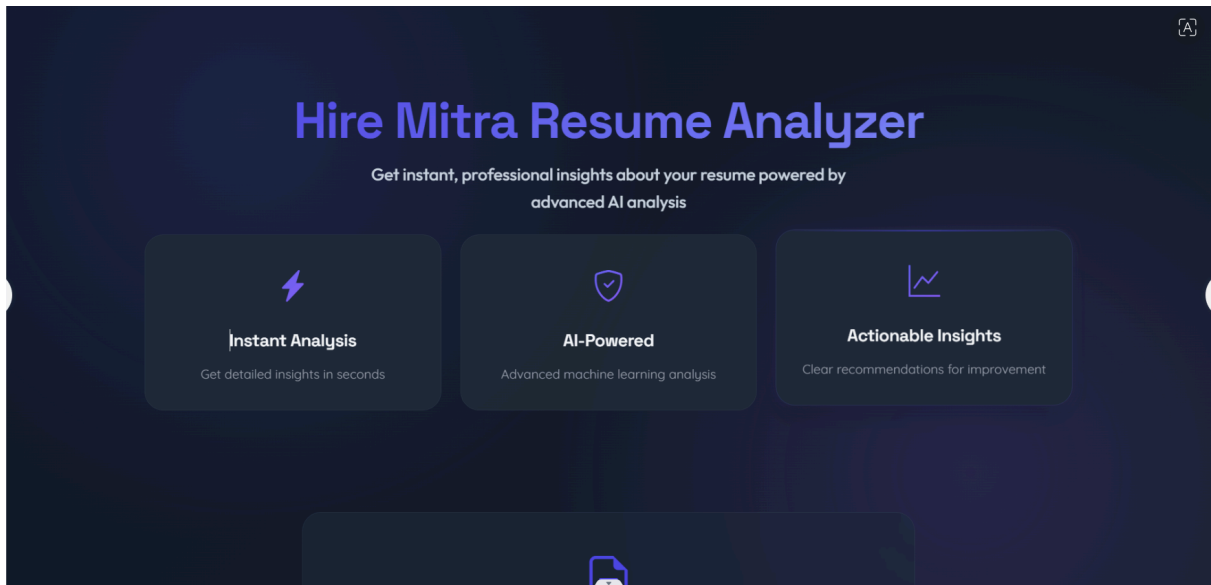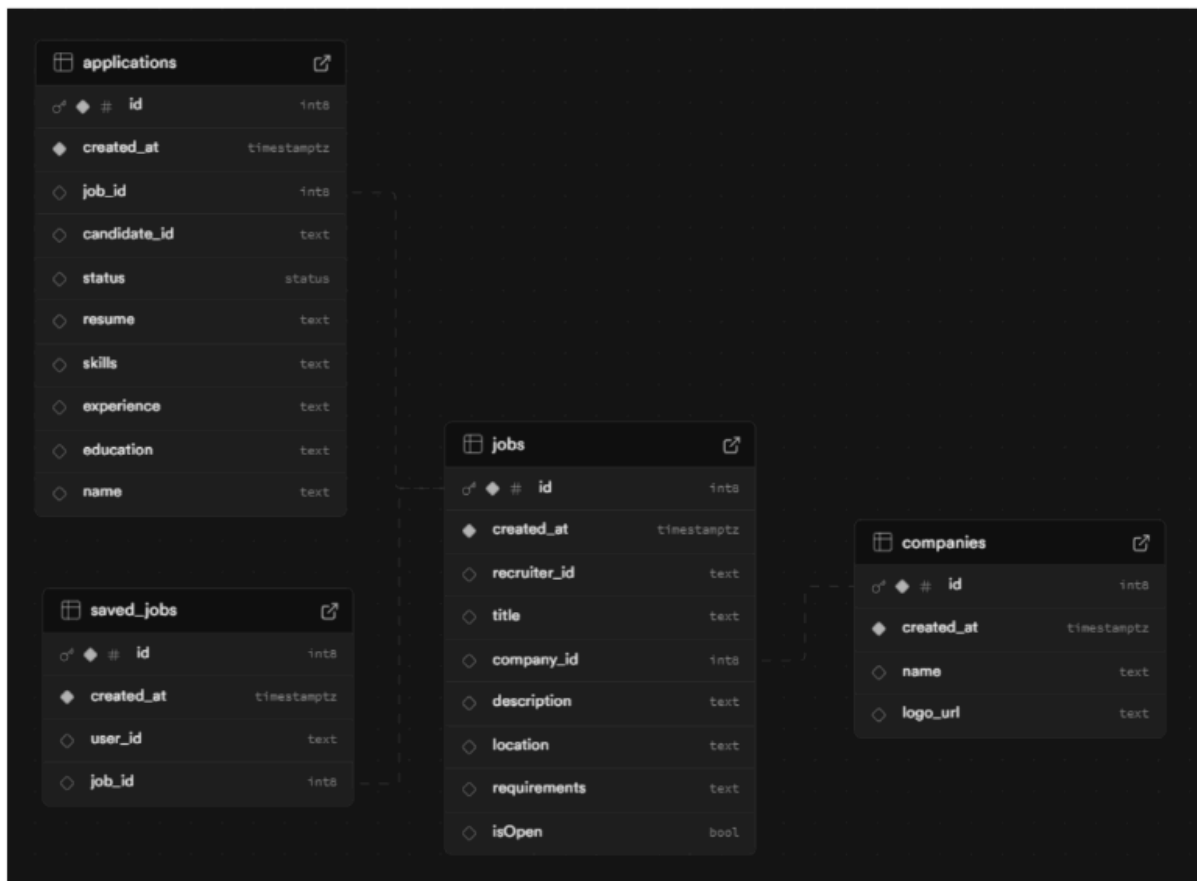
Mobile View:
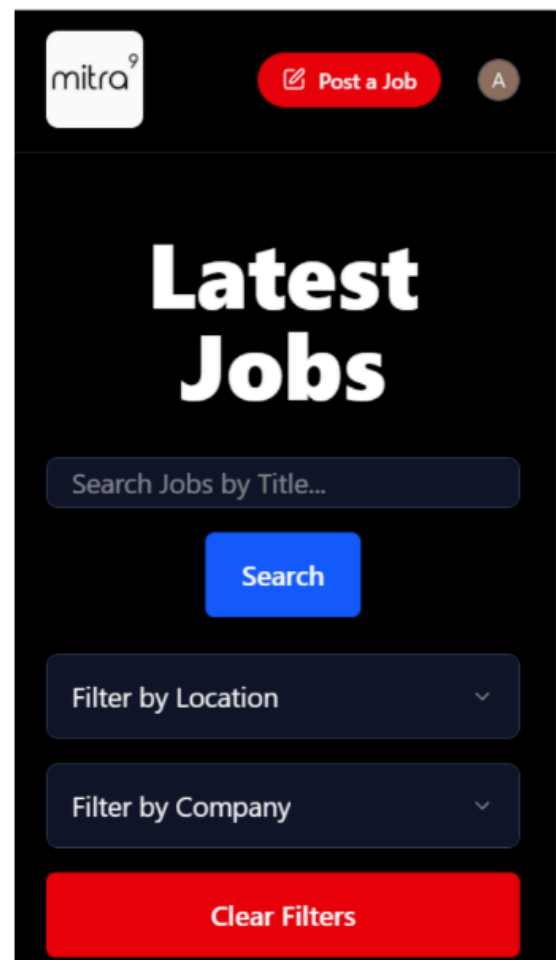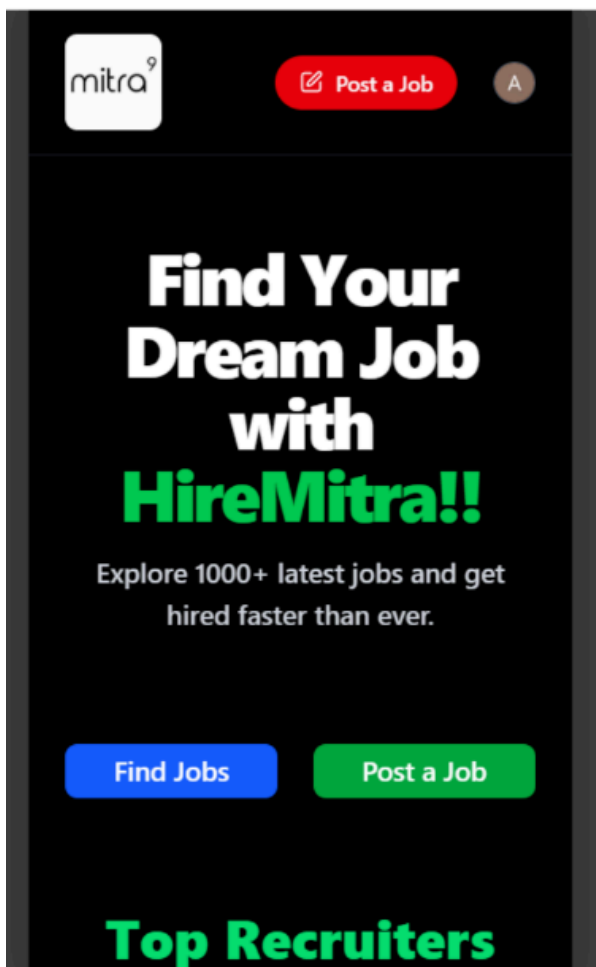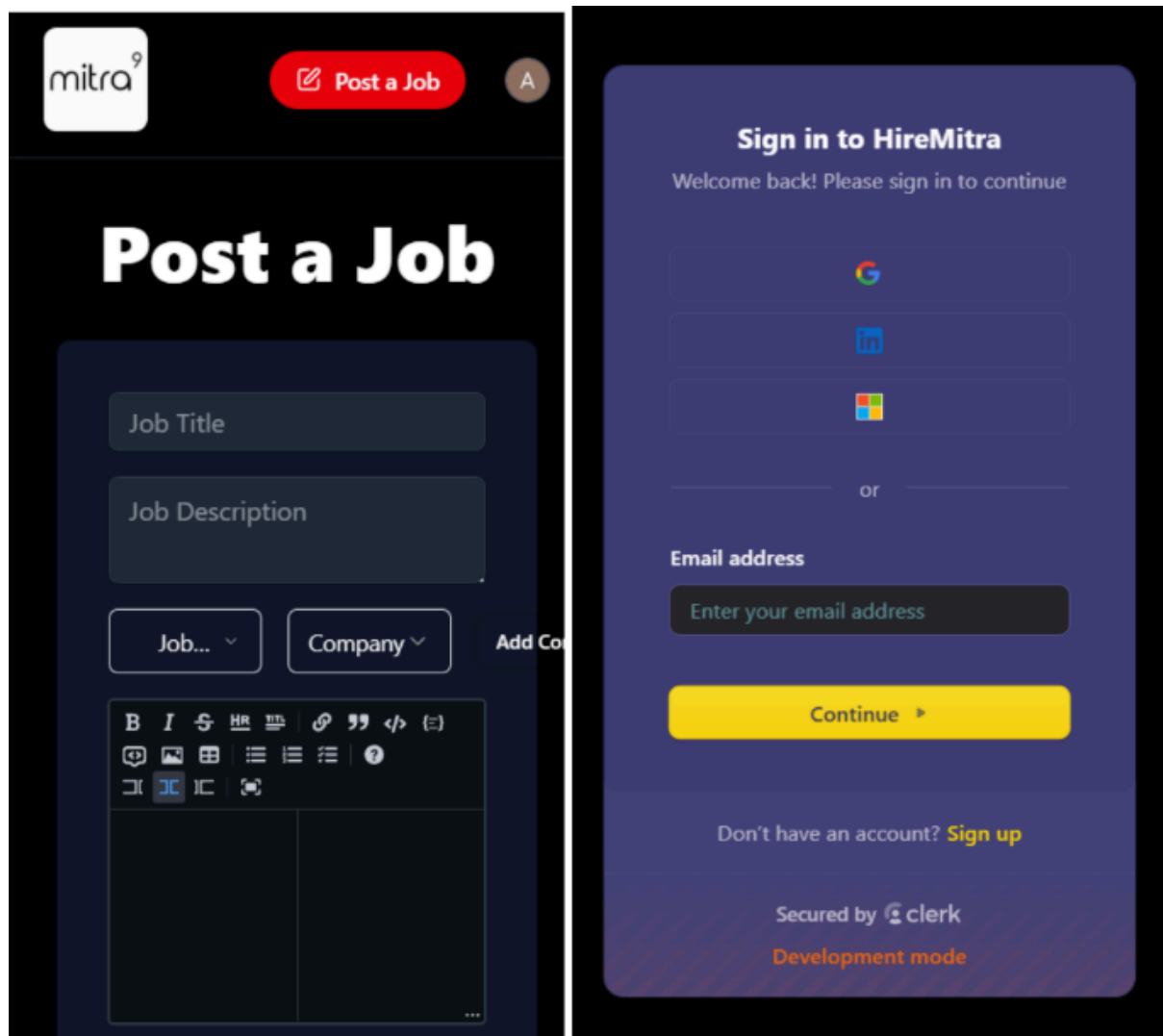
Database ER Diagram:

This is an Entity-Relationship (ER) Diagram for a job portal database, representing relationships between different entities involved in the recruitment process.

1) applications (Stores job applications submitted by candidates)

● id (int8) → Unique identifier for each application.
● created_at (timestamptz) → Timestamp when the application was submitted.
● job_id (int8) → Foreign key referencing the job the candidate applied for.
● candidate_id (text) → Unique identifier for the candidate applying.
● status (status) → Current status of the application (e.g., Pending, Accepted,
Rejected).
● resume (text) → Link or text field containing the candidate's resume.
● skills (text) → List of skills possessed by the candidate.
● experience (text) → Candidate's work experience.
● education (text) → Candidate's educational background.
● name (text) → Candidate's full name.

2) saved_jobs (Stores jobs that users have saved)

- id (int8) → Unique identifier for each saved job entry.
- created_at (timestamp) → Timestamp when the job was saved.
- user_id (text) → Unique identifier for the user who saved the job.
- job_id (int8) → Foreign key referencing the job that was saved.

3) jobs (Stores job postings)

- id (int8) → Unique identifier for each job posting.
- created_at (timestamptz) → Timestamp when the job was posted.
- recruiter_id (text) → Unique identifier for the recruiter who posted the job.
- title (text) → Job title (e.g., Software Engineer, Marketing Manager).
- company_id (int8) → Foreign key referencing the company offering the job.
- description (text) → Detailed description of the job role and responsibilities.
- location (text) → Location of the job (e.g., Remote, New York, Bangalore).
- requirements (text) → List of required skills, qualifications, and experience.
- isOpen (bool) → Boolean value indicating whether the job is still open for applications.

4) companies (Stores company details)

- id (int8) → Unique identifier for each company.
- created_at (timestamptz) → Timestamp when the company was added to the system.
- name (text) → Name of the company.
- logo_url (text) → URL for the company's logo.