

# Job Listing Website Database

## **Authors**

Ritvik Sharma, 11743027

Chance Bradford, 11720208

Ethan Lyons, 11758263

Alex Langland, 11806067

## **Instructor**

Parteek Kumar

## **Course**

CPTS 451

2/23/2025



# Table of Contents:

<a href="#">Table of Contents:</a>	<a href="#">1</a>
<a href="#">Abstract</a>	<a href="#">2</a>
<a href="#">Introduction</a>	<a href="#">3</a>
<a href="#">Overview:</a>	<a href="#">3</a>
<a href="#">Problem Statement:</a>	<a href="#">3</a>
<a href="#">Objectives and Scope:</a>	<a href="#">3</a>
<a href="#">Objectives:</a>	<a href="#">3</a>
<a href="#">Scope:</a>	<a href="#">4</a>
<a href="#">Importance and Impact:</a>	<a href="#">4</a>
<a href="#">Functional and Non-Functional Requirements</a>	<a href="#">5</a>
<a href="#">Functional Requirements:</a>	<a href="#">5</a>
<a href="#">User Stories:</a>	<a href="#">7</a>
<a href="#">Non-Functional Requirements</a>	<a href="#">9</a>
<a href="#">Database Design</a>	<a href="#">11</a>
<a href="#">Entity-Relationships ER Diagram</a>	<a href="#">11</a>
<a href="#">Keys:</a>	<a href="#">11</a>
<a href="#">Primary Keys (PK):</a>	<a href="#">11</a>
<a href="#">Foreign Keys (FK):</a>	<a href="#">11</a>
<a href="#">Relationships:</a>	<a href="#">12</a>
<a href="#">Cardinality:</a>	<a href="#">12</a>
<a href="#">Important Constraints and Relationships:</a>	<a href="#">12</a>
<a href="#">Conversion of ER Diagram to Relational Tables:</a>	<a href="#">13</a>
<a href="#">Table Schema:</a>	<a href="#">13</a>

# Abstract

This project aims to develop a comprehensive job listing website designed to streamline the hiring process for both employers and job seekers. The system facilitates user registration and secure login, enabling personalized experiences. Employers can post and manage job listings, review applications, and maintain company profiles. Job seekers can search for relevant job opportunities, apply with cover letters and uploaded resumes, track application statuses, and save jobs for later consideration. The platform includes an administrative panel for monitoring listings, approving employers, and managing users, ensuring site integrity. Key features include email notifications for application updates and job matches, automatic job expiration, and analytics for performance insights. The system employs JWT authentication for secure user logins and supports multi-user roles with tailored access. Expected outcomes include a user-friendly platform that enhances job search efficiency, simplifies employer hiring processes, and provides administrators with robust management tools, ultimately fostering successful employment connections.

# Introduction

## Overview:

This project involves the development of a dynamic and user-centric job listing website designed to connect employers with qualified job seekers. The platform aims to provide a comprehensive suite of features that facilitate efficient job searching, application management, and employer recruitment processes. The website will feature secure user authentication, role-based access control, and a robust database to manage job listings, applications, and user profiles.

## Problem Statement:

The current job market often presents challenges for both employers and job seekers. Employers struggle to efficiently find and manage qualified candidates, while job seekers face difficulties in navigating numerous job boards and tracking application statuses. Existing platforms often lack comprehensive features, such as integrated application tracking, robust employer management tools, and effective communication channels, leading to inefficiencies and frustration in the hiring process.

## Objectives and Scope:

### Objectives:

- To create a user-friendly platform for employers to post and manage job listings.
- To provide job seekers with advanced search and application tools.
- To implement secure user authentication and role-based access control.
- To develop an administrative panel for site management and monitoring.
- To supply a system that allows for application status tracking, and notification.
- To create a review and feedback system for job seekers.

## Scope:

- The project will encompass the development of a web-based application with features for user registration, job posting, job searching, application management, resume uploads, and administrative oversight.
- The system will support multiple user roles: job seekers, employers, and administrators.
- The project will include a database design to manage user data, job listings, and application information.
- The project will include Email notification and alerts.

## Importance and Impact:

This project is significant as it addresses the critical need for an efficient and streamlined job recruitment process. By providing a centralized platform with robust features, the website will empower employers to find qualified candidates more effectively, reducing time-to-hire and improving recruitment outcomes. For job seekers, the platform will simplify the job search process, enhance application tracking, and provide valuable feedback mechanisms. The anticipated impact includes increased efficiency in the job market, improved connectivity between employers and job seekers, and a reduction in the administrative burden associated with traditional recruitment methods. The admin tools will also provide data that can improve the overall platform.

# Functional and Non-Functional Requirements

## Functional Requirements:

These requirements define what the system must do to ensure core functionality for job seekers, employers, and administrators.

<b>ID</b>	<b>Requirement</b>	<b>Justification</b>
<b>FR1</b>	User Registration and Login	Users must create accounts (Job Seekers, Employers, Admins) to access platform features.
<b>FR2</b>	Job Posting	Employers should be able to post job listings with relevant details.
<b>FR3</b>	Job Searching & Filtering	Job Seekers must search for jobs by category, location, salary, and job type.
<b>FR4</b>	Job Applications	Job Seekers should apply for jobs by submitting resumes and cover letters.
<b>FR5</b>	Employer Dashboard	Employers should manage job listings, applications, and company details.
<b>FR6</b>	Job Application Tracking	Users should track the status of their job applications.
<b>FR7</b>	Resume Upload & Profile Management	Job Seekers should upload resumes and update their profiles.

<b>FR8</b>	Saved Jobs	Job Seekers should bookmark jobs for later applications.
<b>FR9</b>	Admin Panel	Admins should monitor job listings, approve employers, and manage users.
<b>FR10</b>	Notifications & Alerts	Users should receive email notifications about applications, job matches, and status updates.
<b>FR11</b>	Job Expiration Mechanism	Jobs should expire automatically after their deadline.
<b>FR12</b>	Reporting & Analytics	Admins should generate reports on job postings, user activity, and trends.
<b>FR13</b>	Multi-User Role Support	System should distinguish between Job Seekers, Employers, and Admins.
<b>FR14</b>	User Authentication & Security	Use JWT authentication for secure login sessions.
<b>FR15</b>	Review & Feedback	Job Seekers should be able to review employers based on their experience.

## User Stories:

- **FR1: User Registration and Login**
  - As a general user, I want to create an account so that I can store my relevant personal information.
  - As a general user, I want to login to my account so that I can access data that I have previously stored.
- **FR2: Job Posting**
  - As an employer, I want to post job listings so that I can find qualified candidates to hire.
- **FR3: Job Searching**
  - As a job seeker, I want to search available job listings filtered by relevant details so that I can find a job that fits my qualifications.
- **FR4: Job Applications**
  - As a job seeker, I want to apply for jobs so that I can notify employers that I would like to work for them.
  - As a job seeker, I want to submit resumes with my applications so that employers can see my qualifications.
- **FR5: Employer Dashboard**
  - As an employer, I want to manage job listings so that I can edit their information if it changes over time.
  - As an employer, I want to review and accept applications to my listings so that I can hire qualified applicants.
  - As an employer, I want to manage company details so that I can edit our information if it changes over time.
- **FR6: Job Application Tracking**
  - As a job seeker, I want to track the status of my applications so that I know whether they have been accepted or rejected.
- **FR7: Resume Upload & Profile Management**
  - As a job seeker, I want to upload my resume to my profile so that prospective employers can see my qualifications.



- As a job seeker, I want to update my profile so that I can change the information on it if it changes over time.
- **FR8: Saved Jobs**
  - As a Job Seeker, I want to bookmark jobs for later so that I can make a quick personal list of listings to apply to.
- **FR9: Admin Panel**
  - As an Admin, I want to monitor job listings so that I can ensure that fake or poorly-made listings do not clutter the website.
  - As an Admin, I want to approve employers so that I can keep out fake or shady employers.
  - As an Admin, I want to manage users so that I can change any information if an unhandled problem should arise.
- **FR10: Notifications & Alerts**
  - As a Job Seeker, I want to receive email notifications when the status of any of my applications updates so that I can respond to the update in a timely manner.
  - As a Job Seeker, I want to receive email notifications when the website has found possible job matches so that I can apply quickly to jobs I am qualified for.
- **FR12: Reporting & Analytics**
  - As an Admin, I want to generate reports about job postings, user activity, and trends so that I can understand how the website is performing.
- **FR13: Multi-User Role Support**
  - As a General User, I want to only access the relevant features to my user role so that my abilities are tailored to what I need to do.
  - As an Admin, I want each user to be limited to their relevant features so that attackers cannot wreak havoc on my website.
- **FR14: User Authentication & Security**
  - As a General User, I want to log in with JWT authentication so that my login process is secure.
- **FR15: Review & Feedback**
  - As a Job Seeker, I want to be able to leave reviews on employers so that I can share my experience with other job seekers.

## Non-Functional Requirements

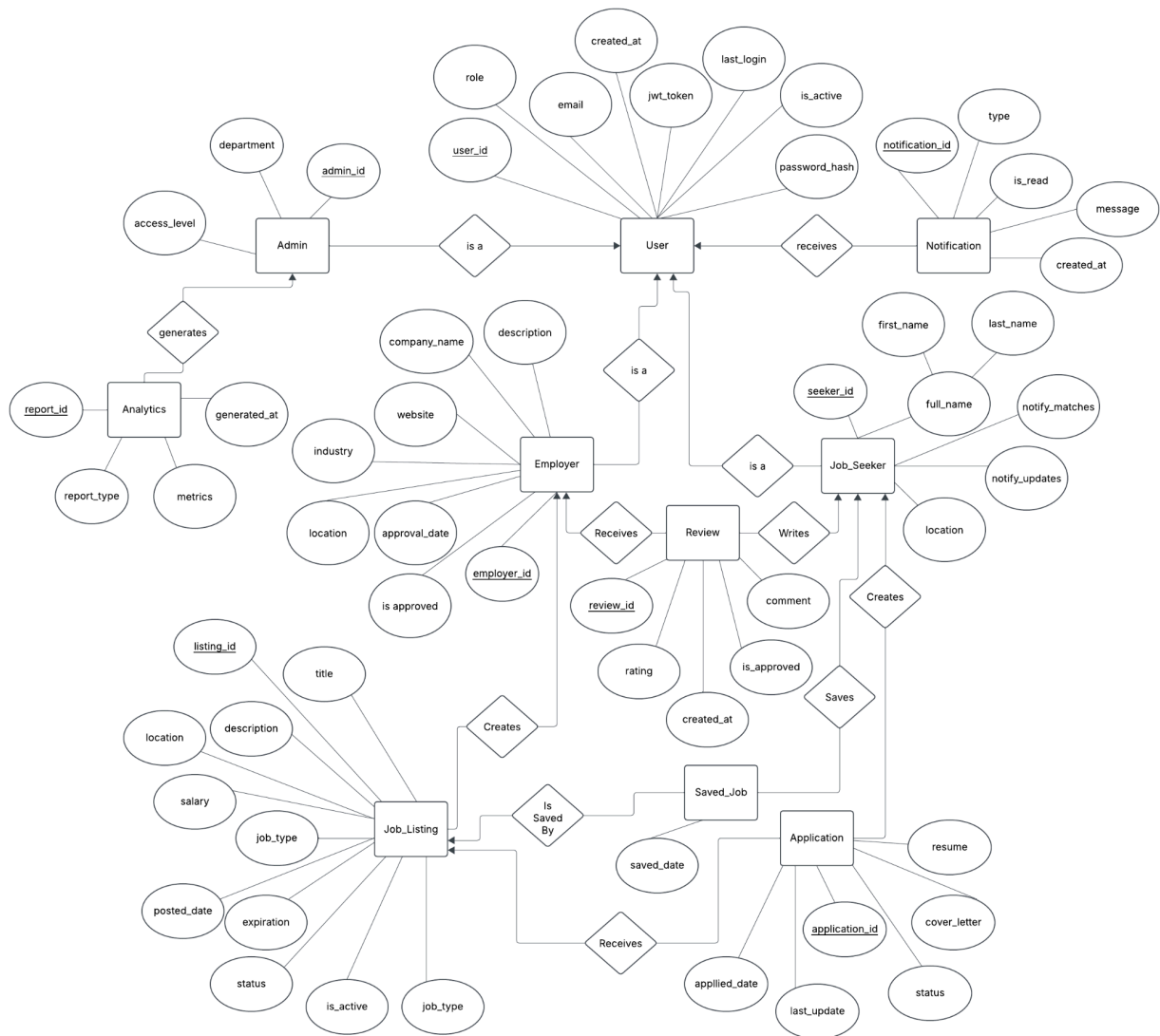
These requirements define how the system should perform rather than what it does.

<b>ID</b>	<b>Requirement</b>	<b>Justification</b>
<b>NFR1</b>	Scalability	The system should support high user traffic as more users join.
<b>NFR2</b>	Performance	Page load time should be <2 seconds to enhance user experience.
<b>NFR3</b>	Security	Use HTTPS, JWT authentication, and secure password hashing.
<b>NFR4</b>	Data Integrity	Ensure data consistency, preventing duplicate applications or expired job postings.
<b>NFR5</b>	Maintainability	Use modular code architecture for easier updates and debugging.
<b>NFR6</b>	Accessibility	The platform should follow WCAG guidelines to be accessible to users with disabilities.
<b>NFR7</b>	Database Reliability	PostgreSQL should handle concurrent queries efficiently.
<b>NFR8</b>	Backup & Recovery	Daily backups should prevent data loss.

<b>NFR9</b>	Mobile Responsiveness	The website should work seamlessly on mobile devices.
<b>NFR10</b>	Usability	The UI should be intuitive and easy to navigate for all user roles.
<b>NFR11</b>	Logging & Monitoring	System should log user activities and errors for debugging.
<b>NFR12</b>	Compliance	The system should adhere to data protection laws (e.g., GDPR).
<b>NFR13</b>	Version Control	Git should be used for tracking code changes.
<b>NFR14</b>	API Rate Limiting	Implement rate limiting to prevent abuse of APIs.
<b>NFR15</b>	High Availability	Ensure 99.9% uptime using reliable hosting solutions.

# Database Design

## Entity-Relationships ER Diagram



### Keys:

#### Primary Keys (PK):

- USER: user\_id
- JOB\_LISTING: listing\_id
- APPLICATION: application\_id
- REVIEW: review\_id

- NOTIFICATION: notification\_id
- ANALYTICS: report\_id

#### Foreign Keys (FK):

- ADMIN, JOB\_SEEKER, EMPLOYER: user\_id (referencing USER)
- JOB\_LISTING: employer\_id (referencing EMPLOYER)
- APPLICATION: listing\_id (referencing JOB\_LISTING), seeker\_id (referencing JOB\_SEEKER)
- SAVED\_JOB: seeker\_id (referencing JOB\_SEEKER), listing\_id (referencing JOB\_LISTING)
- REVIEW: seeker\_id (referencing JOB\_SEEKER), employer\_id (referencing EMPLOYER)
- NOTIFICATION: user\_id (referencing USER)
- ANALYTICS: admin\_id (referencing ADMIN)

#### Relationships:

#### Cardinality:

- USER to ADMIN/JOB\_SEEKER/EMPLOYER: 1:0..1 (User can be one type of user)
- USER to NOTIFICATION: 1:M (User can receive many notifications)
- EMPLOYER to JOB\_LISTING: 1:M (Employer can post many job listings)
- JOB\_SEEKER to APPLICATION: 1:M (Job seeker can submit many applications)
- JOB\_LISTING to APPLICATION: 1:M (Job listing can receive many applications)
- JOB\_SEEKER to SAVED\_JOB: 1:M (Job seeker can save many jobs)
- JOB\_LISTING to SAVED\_JOB: 1:M (Job listing can be saved by many seekers)
- JOB\_SEEKER to REVIEW: 1:M (Job seeker can write many reviews)
- EMPLOYER to REVIEW: 1:M (Employer can receive many reviews)

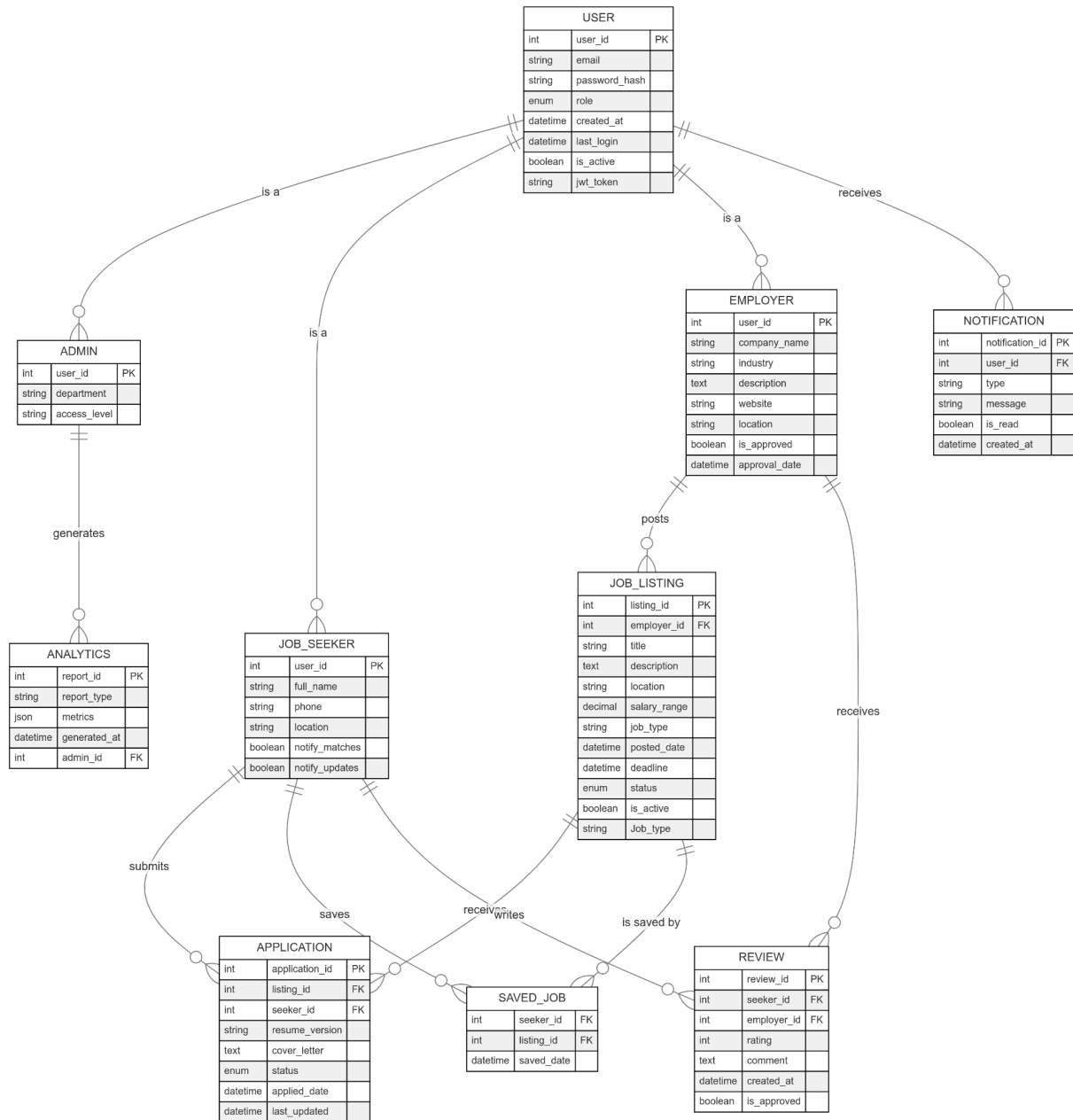
- ADMIN to ANALYTICS: 1:M (Admin can generate many analytics reports)

#### Important Constraints and Relationships:

1. Generalization/Specialization:
  - USER is the superclass/parent entity
  - ADMIN, JOB\_SEEKER, and EMPLOYER are subclasses/child entities
  - This is a disjoint specialization (a user can only be one type)
2. Composite Keys:
  - SAVED\_JOB uses a composite key of (seeker\_id, listing\_id)
3. Weak Entities:
  - NOTIFICATION is weak as it depends on USER
  - APPLICATION is weak as it depends on both JOB\_LISTING and JOB\_SEEKER
4. Additional Constraints:
  - EMPLOYER.is\_approved must be true for posting JOB\_LISTINGs
  - APPLICATION.status should track the application lifecycle
  - REVIEW.rating likely has a range constraint (e.g., 1-5)
  - Dates like deadline in JOB\_LISTING should be after posted\_date
  - Password\_hash in USER must not be null
  - Email in USER should be unique

# Conversion of ER Diagram to Relational Tables:

## Table Schema:



Here are the key design decisions and assumptions:

1. User Structure:
  - Implemented a base USER entity with role-based specialization into JOB\_SEEKER and EMPLOYER
  - This allows for flexible user management while maintaining distinct profiles for each role
  - Assumes each user can only have one role (job seeker or employer)
2. Job Postings and Applications:
  - Created a one-to-many relationship between EMPLOYER and JOB\_POSTING
  - Established a many-to-many relationship between JOB\_SEEKER and JOB\_POSTING through the APPLICATION entity
  - Includes status tracking and temporal attributes to support the application lifecycle
  - Assumes one application per job per seeker
3. Additional Features:
  - SAVED\_JOB implements the bookmarking functionality as a many-to-many relationship
  - REVIEW system allows job seekers to rate employers (many-to-one)
  - All entities include appropriate timestamps for tracking and analytics purposes
4. Assumptions:
  - One active resume per job seeker (with version tracking in applications)
  - Employers can have multiple job postings, but each posting belongs to one employer
  - Reviews are one-way (job seekers review employers, not vice versa)
  - Job status tracking is handled through status fields rather than separate entities