TalentBridge – Project Documentation

Project Overview

TalentBridge is a full-stack recruitment agency platform built with Angular for the frontend and Spring Boot for the backend. It connects employers with job seekers, enabling job postings, candidate applications, and streamlined hiring processes. The platform offers tailored experiences for candidates, employers, and administrators, ensuring smooth recruitment operations from start to finish.

Core Features

For Job Seekers:
- Create and update professional profiles
- Upload CV/Resume
- Search & filter job opportunities
- Apply for jobs directly
- Track application status in real-time

For Employers:
- Company registration and profile creation
- Post, update, and delete job listings
- Review and shortlist candidates
- Contact applicants directly

For Administrators:
- Manage and verify employer accounts
- Moderate job postings and applications
- Generate recruitment reports and statistics

Tech Stack

- Frontend: Angular 17, TypeScript, HTML5, CSS3, TailwindCSS / Bootstrap
- Backend: Spring Boot 3.x, Java 17, Spring Data JPA, Spring Security
- Database: MySQL / PostgreSQL
- API Communication: RESTful APIs
- Authentication: JWT (JSON Web Token)
- Hosting:
* Frontend → Netlify / Vercel
* Backend → AWS EC2 / Render / Azure
* Database → AWS RDS / Azure SQL Database

System Architecture

The system consists of three layers:
1. Frontend (Angular SPA) – Presents the user interface, interacts with backend via API calls.
2. Backend (Spring Boot) – Handles business logic, data processing, and security.
3. Database (Relational DB) – Stores user profiles, job postings, and application records.

Development Workflow

1. Backend Setup:
- Create entity models: User, Job, Application
- Implement repositories and services
- Expose REST controllers for API communication
- Secure endpoints using Spring Security & JWT

2. Frontend Setup:
- Create Angular components for jobs, profiles, and admin tools
- Use Angular services to connect to backend APIs
- Implement routing, forms, and state management

3. Testing:
- Backend: JUnit + Mockito
- Frontend: Jasmine + Karma
- Integration testing of API calls

4. Deployment:
- Deploy backend and database
- Deploy frontend and configure API URL

Security

- JWT-based authentication and authorization
- Role-based access control (Admin, Employer, Candidate)
- Input validation and SQL injection prevention
- Encrypted password storage (BCrypt)

Future Enhancements

- AI-powered job-candidate matching
- Email/SMS notifications for job updates
- Advanced analytics dashboard for employers
- Integrated video interview scheduling
- Mobile app version (Ionic or React Native)

System Architecture Diagram