

JobFix

An advanced Job Portal

Group 17

A large white circle with a subtle drop shadow, containing the 'Job Portal' logo. The logo features the word 'JOB' in a large, bold, sans-serif font, with the 'O' being a teal circle. Below 'JOB', the word 'PORTAL' is written in a smaller, teal, all-caps, sans-serif font.

JOB
PORTAL

Our Team



Tharindu
Senevirathna

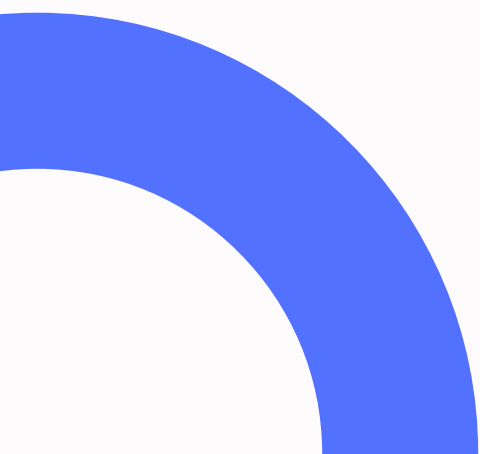


Didula
Thilakarathna



Devinda
Dilshan

Project Overview

- Job Fix is an innovative job portal designed to streamline the recruitment process by connecting job seekers with relevant job opportunities and helping recruiters find the right candidates efficiently.
 - This platform provides a seamless user experience by integrating key features such as Create profiles for Candidates and Job seekers ,job listings and application tracking
- 

Types of Users and their roles

Candidate


Can create a profile, upload a resume, and apply for jobs. Can track job applications

Recruiter

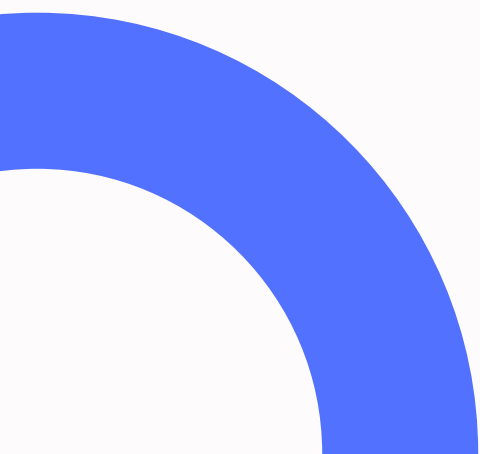
Can post job openings and manage applications, view job seeker profiles and contact shortlisted candidates, create a company profile and maintain job postings.

Administrator

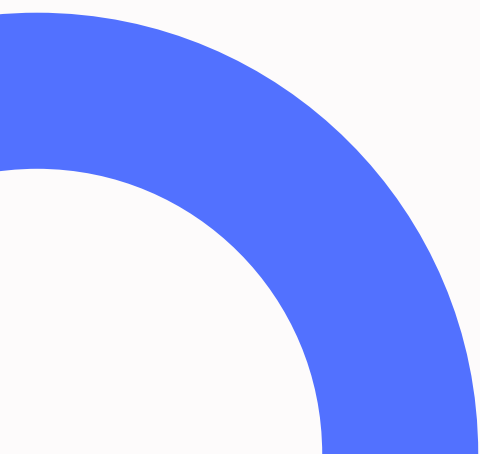
Can manage users, job postings, and platform security and Has the power to remove spam job listings and handle platform moderation.



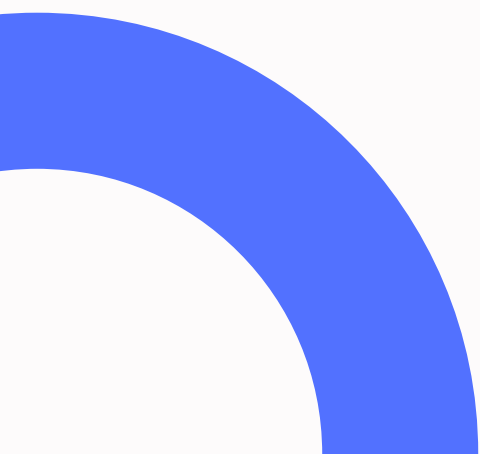
Features

- User Authentication
 - Job Posting & Management (For Recruiter)
 - Advanced Job Search & Filtering (For Job Seekers)
 - Application Management
 - Resume Upload & Profile Management
 - Admin Dashboard
- 
- A blue decorative arc is located in the bottom-left corner of the slide.

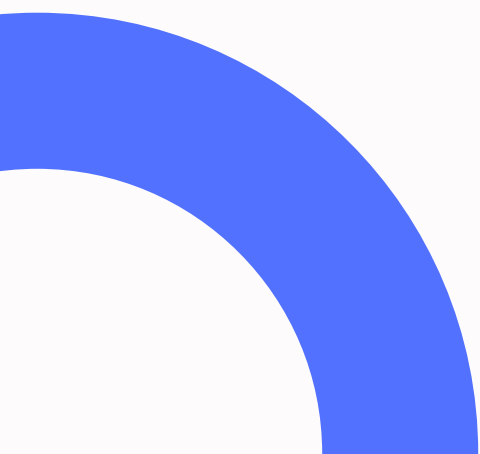
Technology Stack

- **Frontend:** React
 - **Backend:** SpringBoot
 - **Database:** MongoDB
 - **Version Control:** GitHub for collaboration and updates
- 

Challenges Faced

- **Frontend-Backend Integration:** Ensuring smooth communication between React and backend APIs was challenging but resolved with proper state management and error handling.
 - **Database Design:** Designing scalable MongoDB schemas while managing relationships between collections required experimentation and flexibility.
 - **Authentication:** Secure login and role-based access control were tricky but implemented using JWT and bcrypt for secure token-based authentication.
- 

Challenges Faced

- **Deployment:** Setting up AWS EC2 with MongoDB Atlas was complex, but by following tutorials and testing in stages, deployment was successful.
 - **Cross-Browser Compatibility:** Ensuring the app worked on all browsers and devices was resolved using responsive design and CSS media queries.
- 

Demonstration

