Nitte job portal

Java

Amogha\_ K

**1. Introduction**

The Indeed Website Clone project is a comprehensive full-stack web application designed to mimic the functionalities of the popular job search platform, Indeed. The project was developed using a modern technology stack, including React.js and Material-UI for the frontend, Spring Boot and Hibernate for the backend, and MongoDB for the database. This report details the development process, key features, and technical implementation of the project.

**2. Project Objectives**

The primary objective of the Indeed Website Clone project was to create a functional and visually appealing web application that offers the core features of the Indeed platform. This includes job listing and search capabilities, user authentication, and responsive design. An additional goal was to include an image upload feature, where users can upload images related to job posts, with proper spacing between them to ensure a clean and organized interface.

**3. Technologies Used**

* **Frontend:**
  + **React.js:** A JavaScript library used for building the user interface, offering efficient rendering and component-based architecture.
  + **Material-UI:** A popular React UI framework used for creating responsive and visually consistent components.
* **Backend:**
  + **Spring Boot:** A Java-based framework used for developing the backend services, providing a robust and scalable foundation.
  + **Hibernate:** An ORM (Object-Relational Mapping) tool used to manage the database interactions, ensuring seamless data persistence.
* **Database:**
  + **MongoDB:** A NoSQL database used for storing and managing data, offering flexibility and scalability.

**4. Key Features**

**4.1 Job Listing and Search**

The application allows users to browse and search for jobs based on various criteria such as location, job title, and company. The frontend is designed using React.js and Material-UI, providing an intuitive and responsive user interface.

**4.2 User Authentication**

Spring Boot was used to implement secure user authentication and authorization. This ensures that only registered users can access specific features such as job application tracking and profile management.

**4.3 Responsive Design**

Material-UI was instrumental in creating a responsive design that adapts to different screen sizes, ensuring a seamless experience across desktops, tablets, and mobile devices.

**5. Implementation Details**

**5.1 Frontend Development**

The frontend was developed using React.js, with Material-UI providing the necessary components for building a cohesive and responsive interface. The image upload functionality was implemented using React’s state management and controlled components, allowing users to preview images before uploading them.

**5.2 Backend Development**

The backend was developed using Spring Boot, which provided the necessary RESTful services for handling user authentication, job listing management, and image uploads. Hibernate was used to map Java objects to MongoDB collections, ensuring smooth data transactions.

**5.3 Database Management**

MongoDB was chosen for its flexibility in handling unstructured data. The image upload feature leverages MongoDB’s GridFS, which allows for efficient storage of large files. The images are retrieved and displayed on the frontend with proper spacing, enhancing the overall user experience.

**6. Challenges and Solutions**

**6.1 Data Synchronization**

Synchronizing data between the frontend and backend, especially with image uploads, required careful handling of asynchronous operations. This was managed using React’s built-in hooks and Spring Boot’s asynchronous processing capabilities.

**7. Conclusion**

The Indeed Website Clone project successfully achieved its objectives, creating a functional and user-friendly job search platform. The integration of image upload functionality with proper spacing between images added an extra layer of usability and visual appeal. This project provided valuable experience in full-stack development and reinforced the importance of responsive design, data management, and user experience in web development

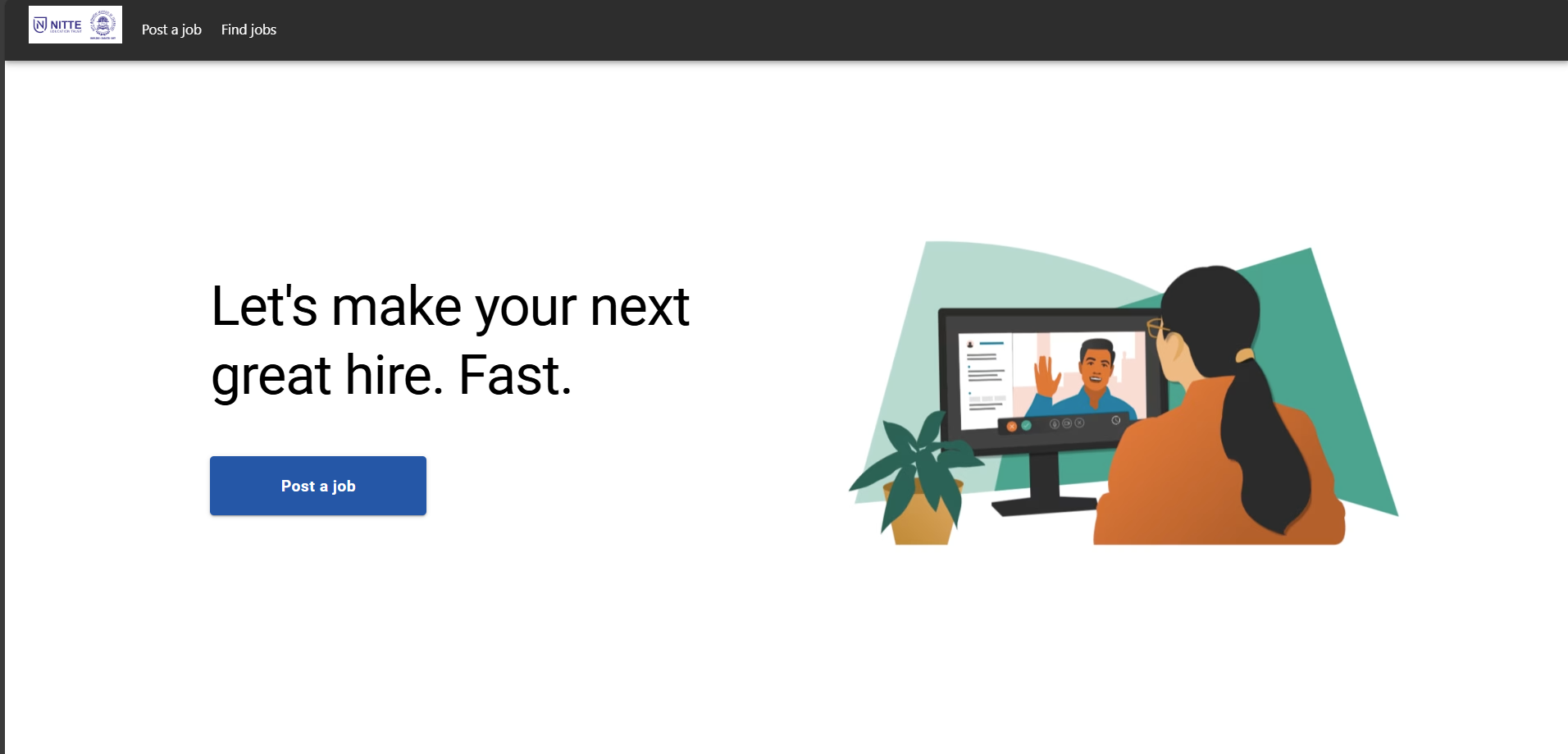


Fig 1 :Front end UI

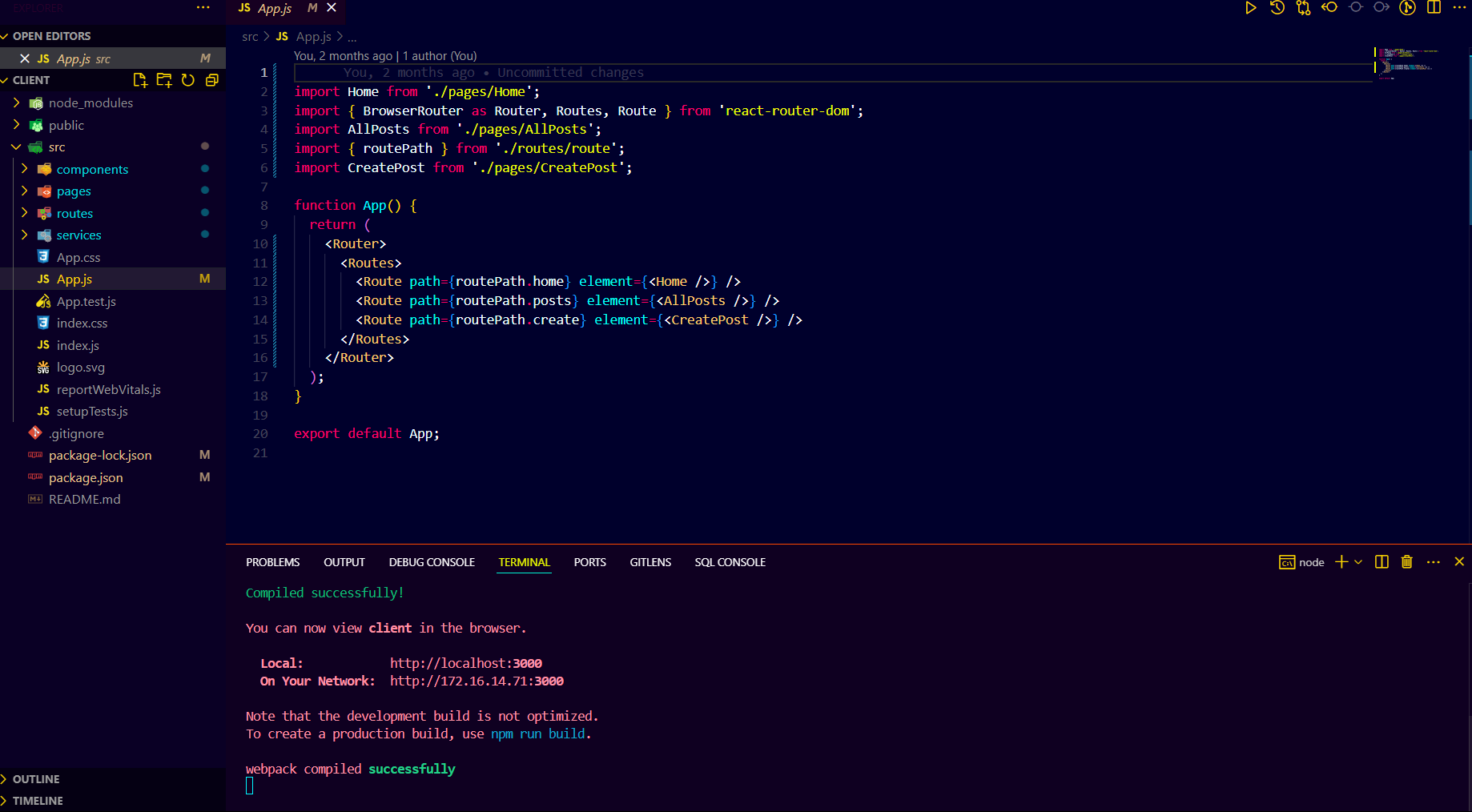


Fig 2 : Folder structure in VS code IDE

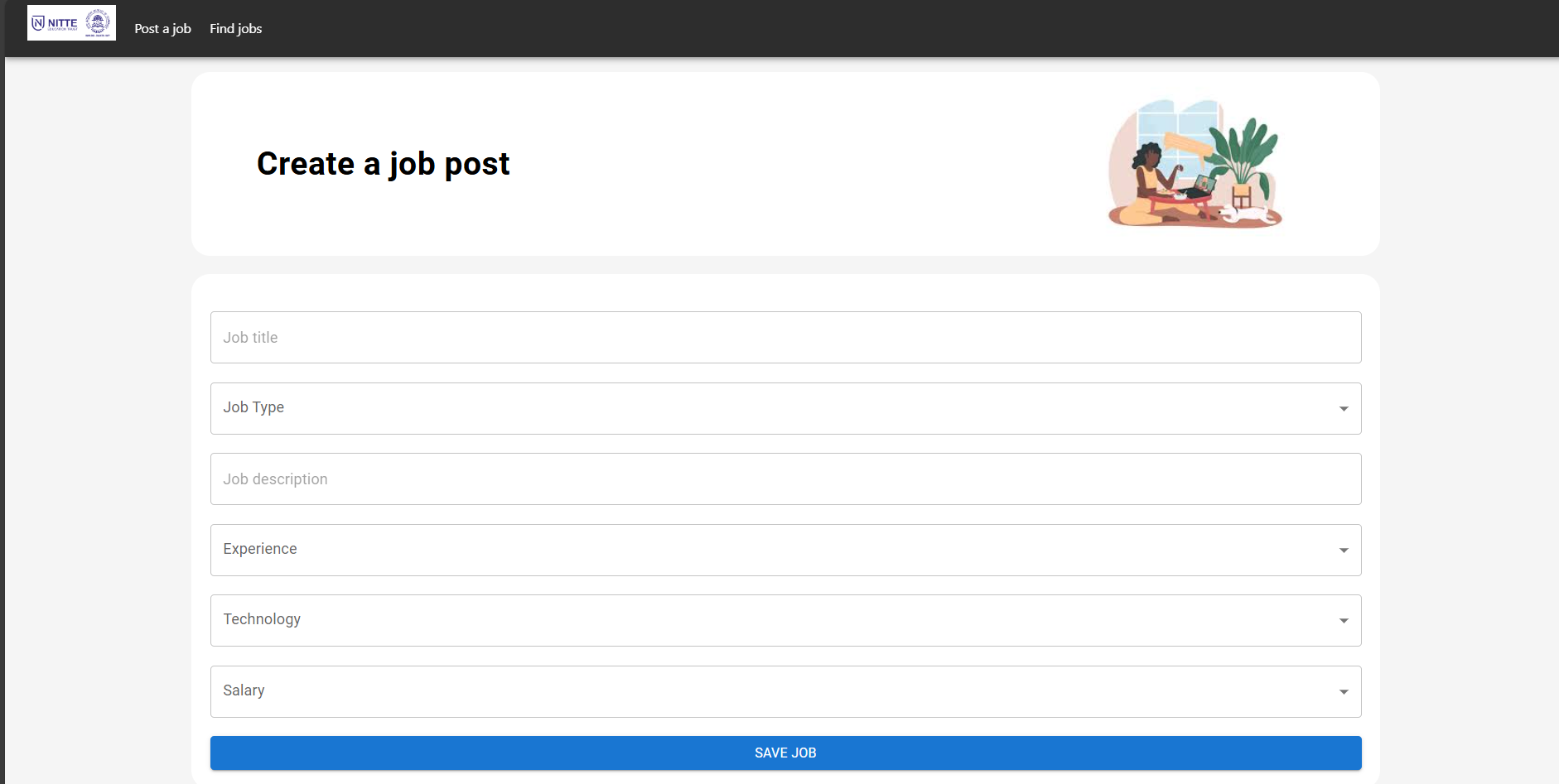


Fig 3: Create job Section

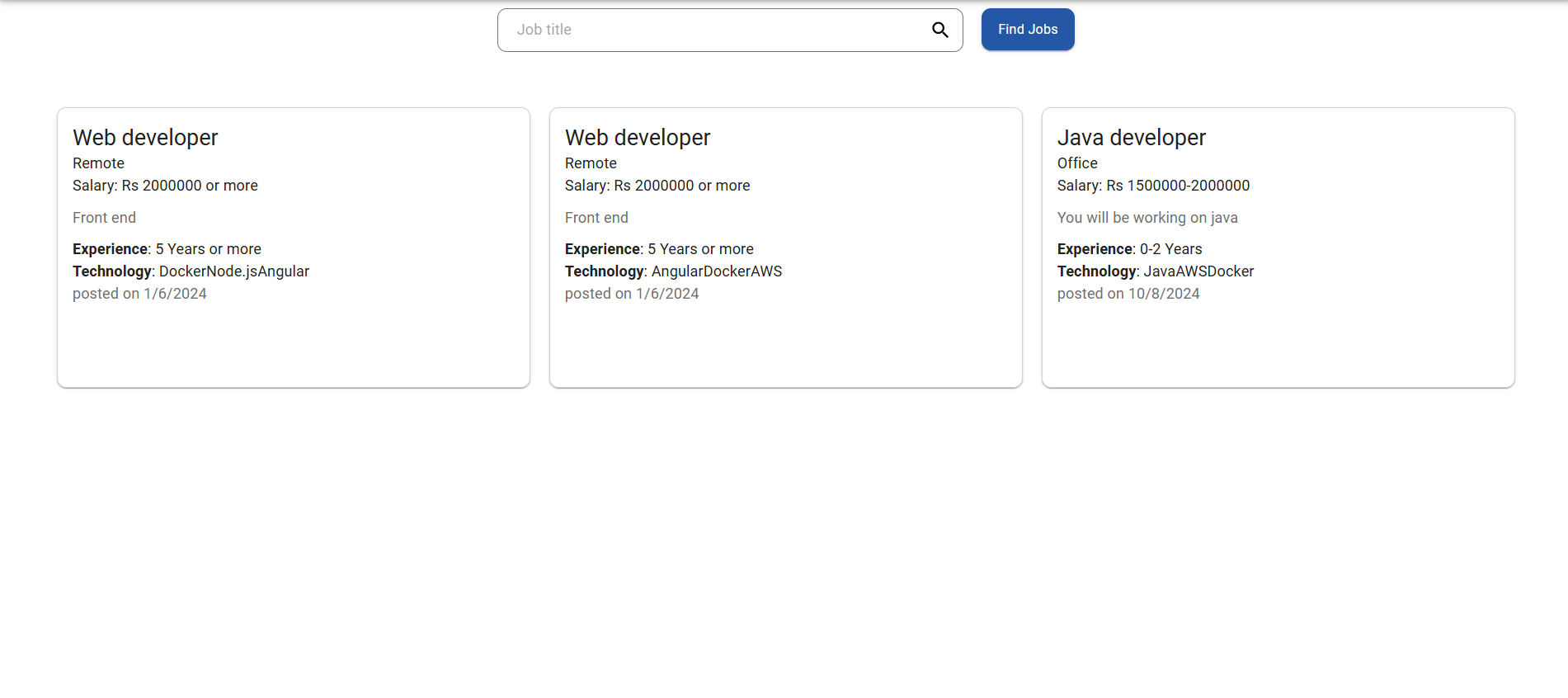


Fig 4: Job section

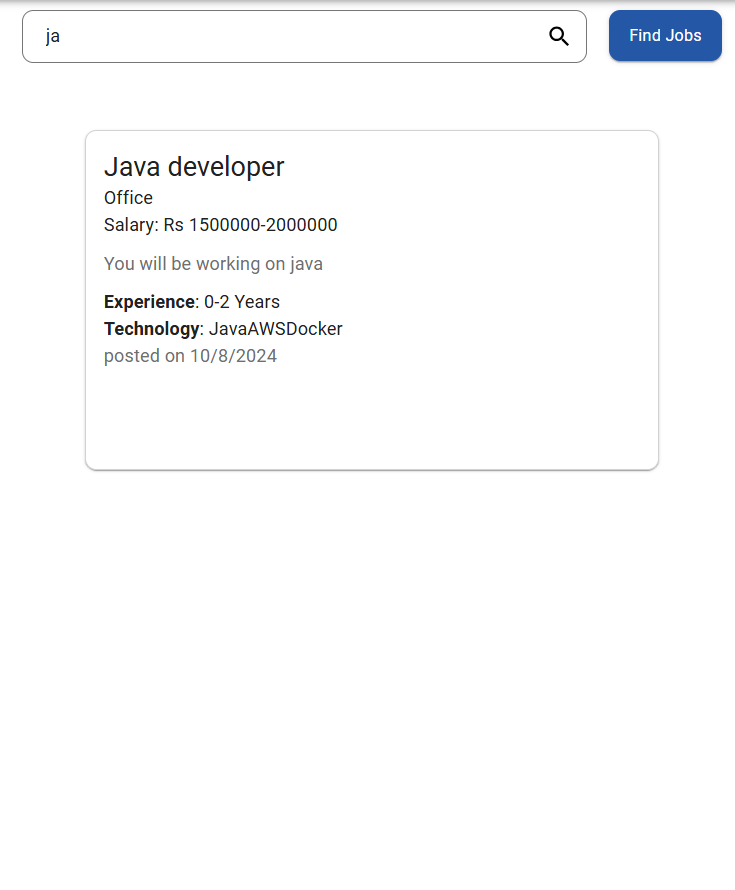


Fig 5: Search bar implementation

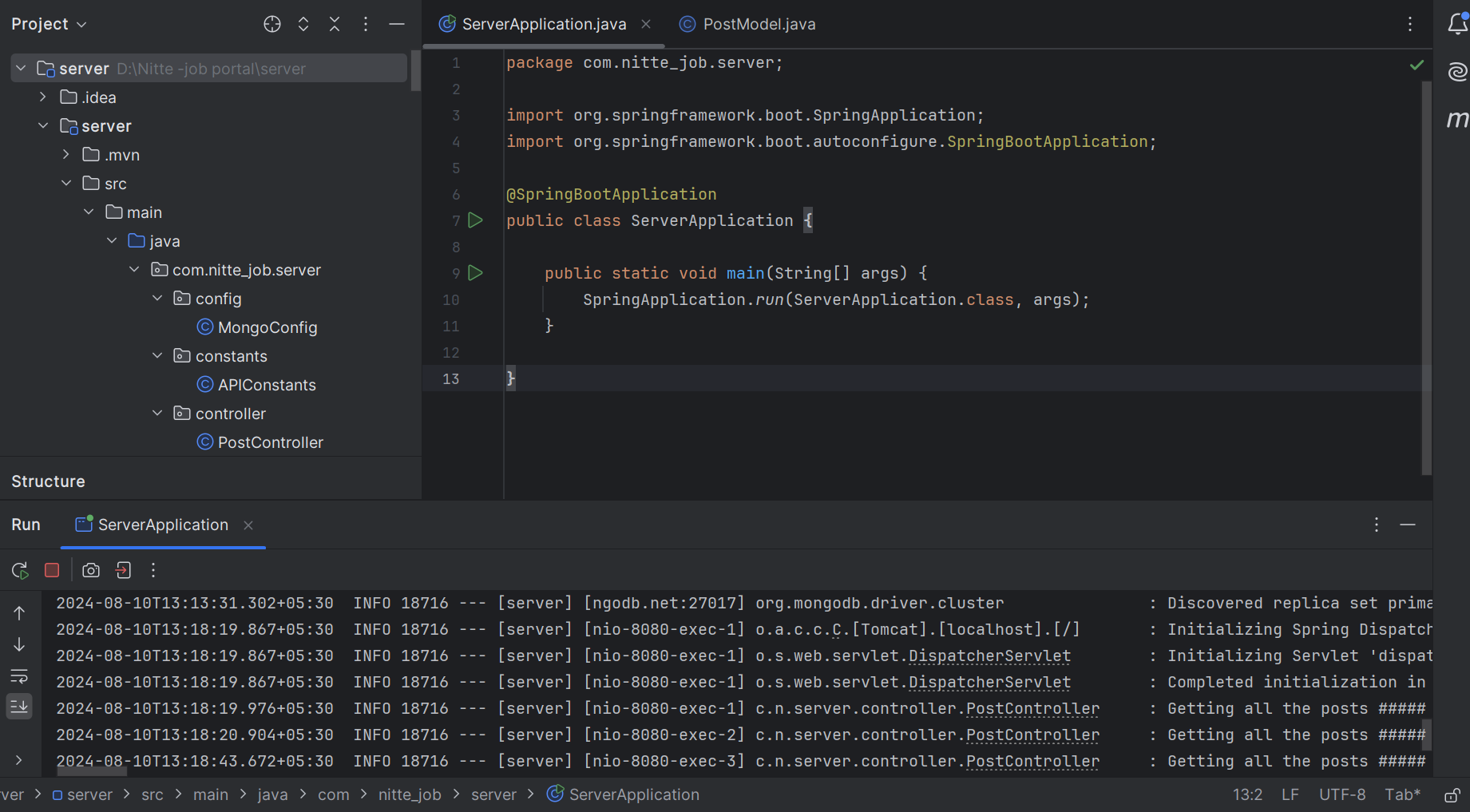


Fig 6 : Folder structure for backend(IntelliJ IDE)