

Full Stack Development with MERN

1. Introduction

- **Project Title:** Freelancing application MERN
- **Team Members:** Kiruthika K, Kasthuri T, Swathi C, Swapna S

2. Project Overview

- **Purpose:**
 - The project aims to create a freelancing platform where clients can post jobs and freelancers can apply for projects. The goal is to streamline the freelancing ecosystem with robust features and seamless user experience.
- **Features:**
 - User authentication (Client and Freelancer roles)
 - Job posting and management
 - Application tracking and status updates
 - Chat feature for communication
 - Profile management for users

3. Architecture

- **Frontend:** Built using React, featuring reusable components and responsive design.
- **Backend:** Developed with Node.js and Express.js for robust API handling.
- **Database:** Mongo DB stores user details, job postings, and application data, ensuring efficient retrieval and updates.

4. Setup Instructions

- **Prerequisites:**
 1. Node.js
 2. Mongo DB
 3. A code editor like VS Code
- **Installation:**

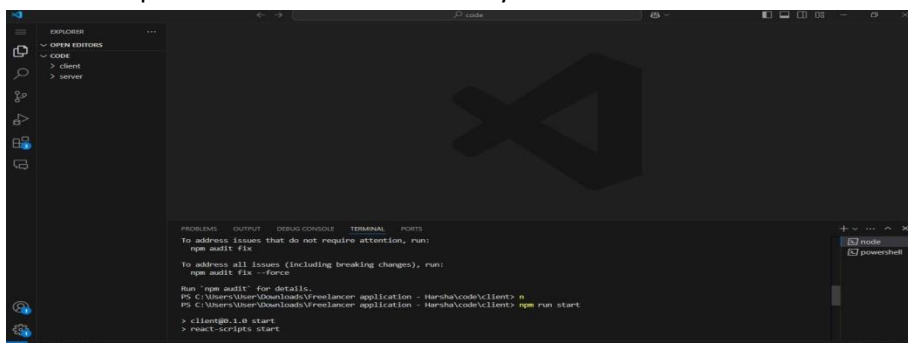
1. Clone the repository.
2. Navigate to the project directory and install dependencies using `npm install`.
3. Set up `.environment` files with necessary configurations for both frontend and backend.

5. Folder Structure

- **Client:** Organized into components, services, and pages.
- **Server:** Divided into routes, controllers, and models for clarity and efficiency.

6. Running the Application

- **Frontend:** `npm start` in the client directory.
- **Backend:** `npm start` in the server directory.



7. API Documentation

- `GET /jobs`: Retrieve all job postings.

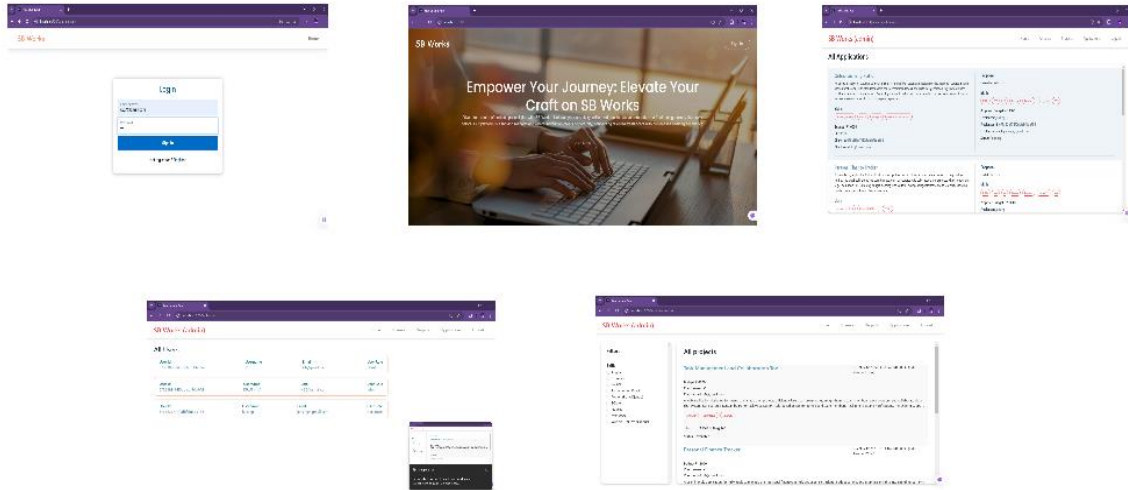
- POST /apply: Submit a job application.
- POST /login: Authenticate a user.
- Example responses are documented in the README file.

8. Authentication

- Handled using JSON Web Tokens (JWT).
- Secure token storage and validation ensure user authorization.

9. User Interface

- Screenshots of the user dashboard, job posting form, and application tracking pages will be included here.



10. Testing

- Tools used: Jest and Postman.
- Includes both unit tests and integration tests.

11. Screenshots or Demo

Screenshots of the user workflow will be added.

A link to the demo: [<https://drive.google.com/file/d/125MpE-HSik-yHb8Oixn899kC6RdMoC2A/view?usp=sharing>]

12. Known Issues

- Minor latency in job search functionality. Planned updates for improving real-time communication.

13. Future Enhancements

- Adding payment gateway integration.
- Introducing AI-based job recommendations for freelancers.
- Improving user analytics for job performance tracking.