Freelancing Platform Web Application

### Developed By

* **Name:** Shrikant Samudre
* **Course:** BCA (Bachelor of Computer Applications)
* **Technology Stack:** React.js, Node.js (optional), CSS, JavaScripte

### Objective of the Project

The main objective of this project is to create a user-friendly freelancing platform where:

* Clients can post jobs
* Freelancers can browse and apply for jobs
* Both can communicate and manage project workflow efficiently

### Tools & Technologies Used

|  |  |
| --- | --- |
| **Tool/Language** | **Purpose** |
| React.js | Frontend UI Development |
| HTML/CSS | Designing and Layout |
| JavaScript | Logic & Interactions |
| Git & GitHub | Version Control & Hosting |
| VS Code | Code Editor |

### System Architecture

**Frontend Only (SPA - Single Page Application):**

* Pages: Home, Login, Register, Post Job, Browse Jobs, Freelancer Profile, etc.

### Modules / Features

#### 👤 User Module:

* User Registration & Login (Freelancer/Client)
* Profile Creation

#### 📃 Job Module:

* Job Posting (Client)
* Job Listing & Filters (Freelancer)
* Job Application

#### 💬 Communication (Optional Future Scope):

* Chat between client & freelancer

#### 🎯 Admin Module (Optional Future Scope):

* View users, jobs, manage reports

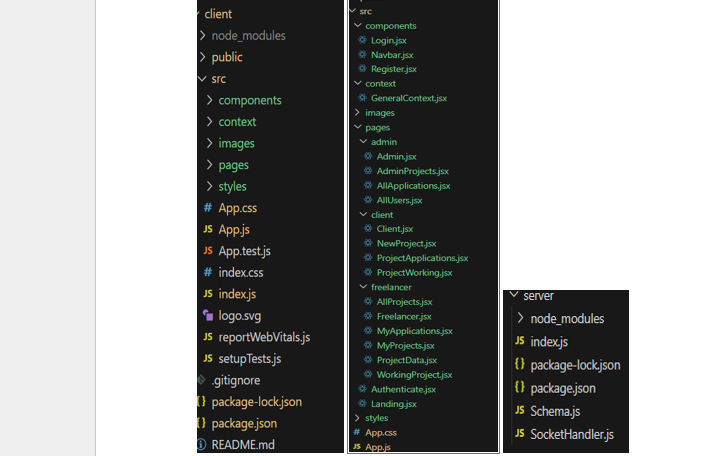
### 📸 7. Output Screenshots

Include these in /screenshots folder and also paste in Word/PDF:

* Home Page
* Login/Register Page
* Post Job Page
* Browse Jobs Page
* Freelancer Dashboard

**PROJECT STRUCTURE**

SB Works leverages React.js for the user interface. The client-side code likely consists of reusable components for profiles, projects, and chat, assembled into pages like project browsing or freelancer profiles. Shared data like user info or search filters might be managed with React Context. On the server side, Node.js handles API requests for user management, project actions, and communication. Mongoose models ensure structured interaction with the MongoDB database. This breakdown provides a foundational understanding of SB Works' architecture.



**PRE-REQUISTIC:**

Here are the key prerequisites for developing a full-stack application using Express Js, MongoDB, React.js:

✔**Node.js and npm:**

Node.js is a powerful JavaScript runtime environment that allows you to run JavaScript code on the server-side. It provides a scalable and efficient platform for building network applications.

Install Node.js and npm on your development machine, as they are required to run JavaScript on the server-side.

Download: https://nodejs.org/en/download/

Installation instructions: <https://nodejs.org/en/download/package-manager/>

✔**Express.js:**

Express.js is a fast and minimalist web application framework for Node.js. It simplifies the process of creating robust APIs and web applications, offering features like routing, middleware support, and modular architecture.

Install Express.js, a web application framework for Node.js, which handles server-side routing, middleware, and API development.

Installation: Open your command prompt or terminal and run the following command:

**npm install express**

✔**MongoDB:**

MongoDB is a flexible and scalable NoSQL database that stores data in a JSON-like format. It provides high performance, horizontal scalability, and seamless integration with Node.js, making it ideal for handling large amounts of structured and unstructured data.

Set up a MongoDB database to store your application's data.

Download: https://www.mongodb.com/try/download/community

Installation instructions: <https://docs.mongodb.com/manual/installation/>

✔**React.js:**

React.js is a popular JavaScript library for building user interfaces. It enables developers to create interactive and reusable UI components, making it easier to build dynamic and responsive web applications.

Install React.js, a JavaScript library for building user interfaces.

Follow the installation guide: <https://reactjs.org/docs/create-a-new-react-app.html>

✔**HTML, CSS, and JavaScript**: Basic knowledge of HTML for creating the structure of your app, CSS for styling, and JavaScript for client-side interactivity is essential.

✔**Database Connectivity**: Use a MongoDB driver or an Object-Document Mapping (ODM) library like Mongoose to connect your Express Js server with the MongoDB database and perform CRUD (Create, Read, Update, Delete) operations

✔**Front-end Framework**: Utilize React Js to build the user-facing part of the application, including entering booking room, status of the booking, and user interfaces for the admin dashboard. For making better UI we have also used some libraries like material UI and bootstrap.

✔**Version Control**: Use Git for version control, enabling collaboration and tracking changes throughout the development process. Platforms like GitHub or Bitbucket can host your repository.

Git: Download and installation instructions can be found at: https://git-scm.com/downloads

✔**Development Environment**: Choose a code editor or Integrated Development Environment (IDE) that suits your preferences, such as Visual Studio Code, Sublime Text, or WebStorm.

• Visual Studio Code: Download from <https://code.visualstudio.com/download>

To run the existing Freelancer App project downloaded from Drive:

Use the code:

Drive link: <https://drive.google.com/drive/folders/10mSn2lMTaVMDWWFNjeJjiOLfmcD3-87C?usp=sharing>

Install Dependencies:

• Navigate into the cloned repository directory:

cd freelancer-app-MERN

• Install the required dependencies by running the following commands:

cd client

npm install

../cd server

npm install

Start the Development Server:

• To start the development server, execute the following command:

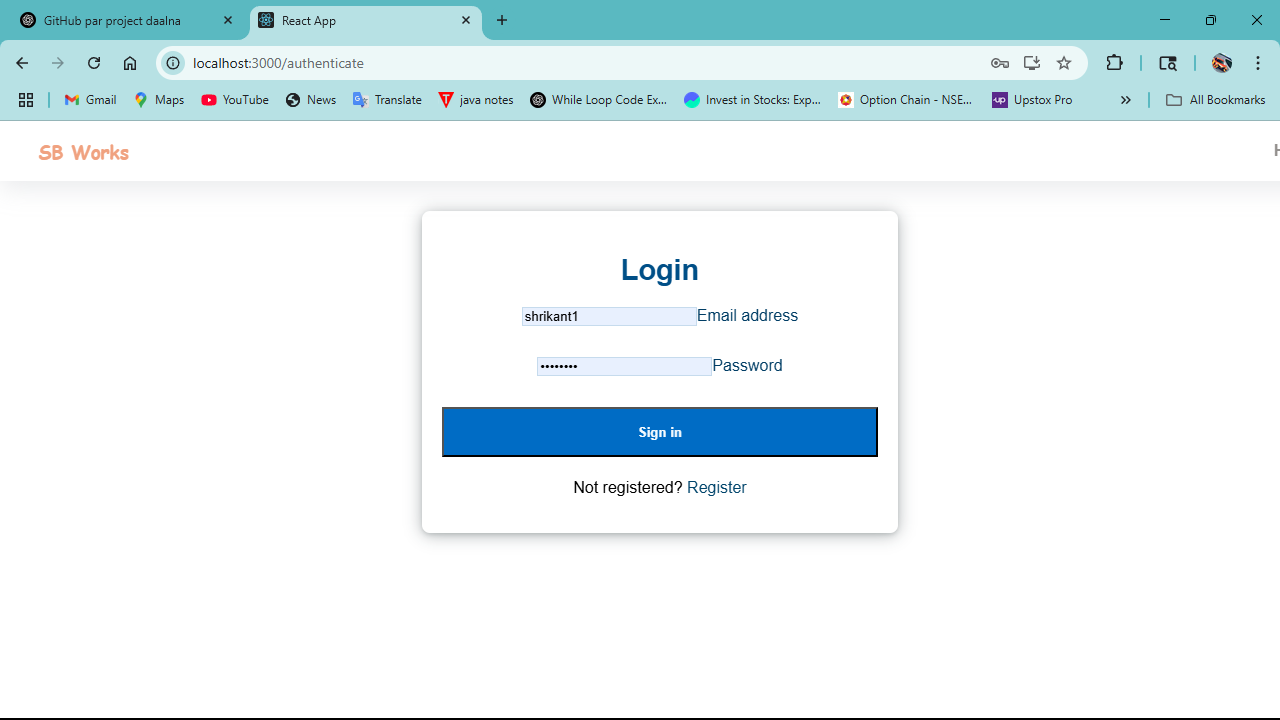
npm start

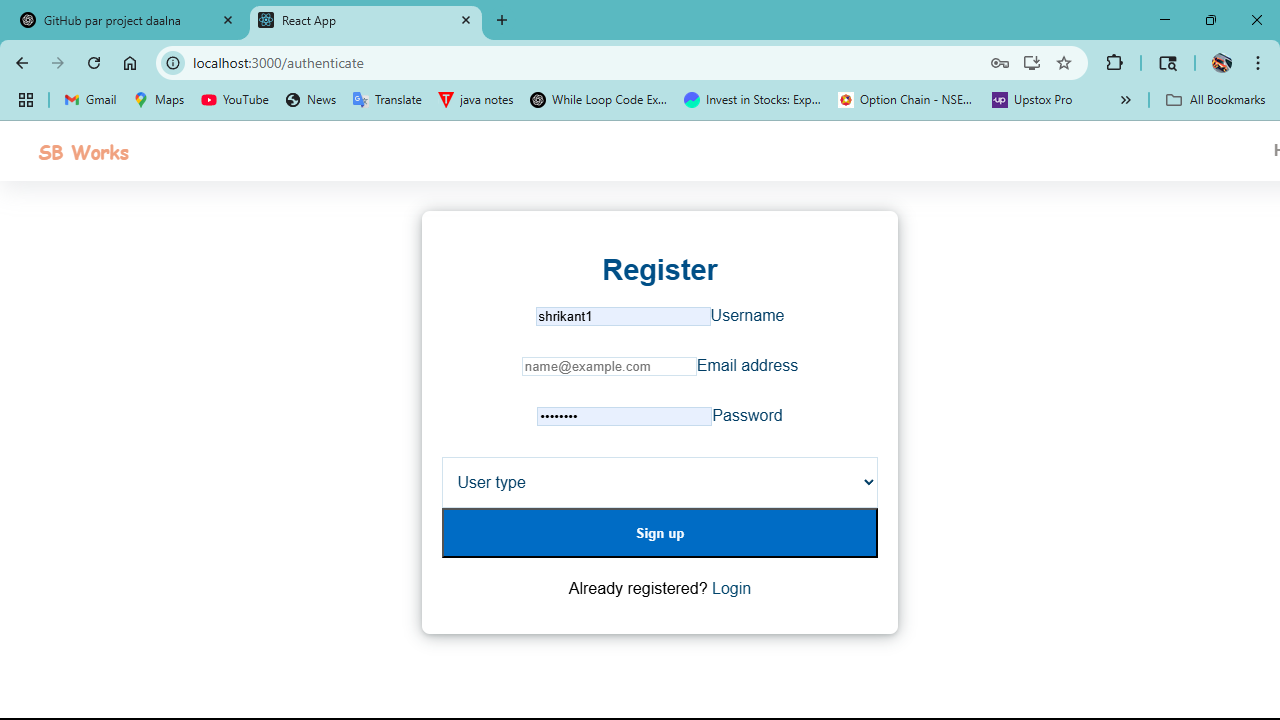
• The SB Works app will be accessible at <http://localhost:3000>

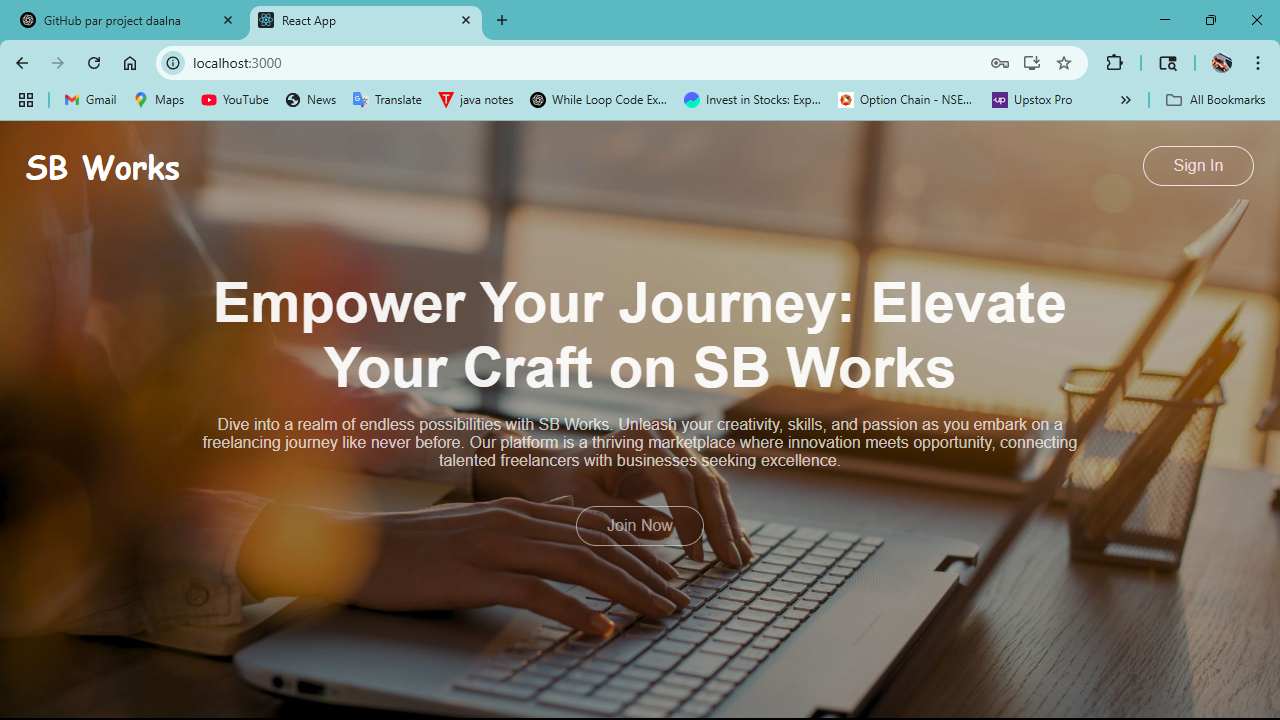
You have successfully installed and set up the SB Works application on your local machine. You can now proceed with further customization, development, and testing as needed.

### 📄 8. Future Scope

* Add Payment Gateway
* Real-time chat with socket.io
* Admin Dashboard
* Rating & Reviews system
* Mobile App version (React Native or Flutter)







### Signature

**Submitted By:** Shrikant Samudre