**Project Title: FreelanceFinder**

**(Discovering Opportunities, Unlocking Potential)**

**1. INTRODUCTION**

**1.1 Project Overview**

SB Works is an innovative MERN-stack-based freelancing application that helps clients connect with freelancers easily and securely. It allows clients to post projects, review bids, communicate with freelancers, and manage work submissions—while freelancers can explore and apply to tasks that match their skills.

**1.2 Purpose**

The main aim of SB Works is to simplify the freelancing process by offering a one-stop platform for hiring, communication, submission, and evaluation, ensuring transparency and efficiency for both clients and freelancers.

**2. IDEATION PHASE**

**2.1 Problem Statement**

Freelancing platforms are often cluttered or lack proper management features. There is a need for a platform that enables seamless collaboration with built-in communication, transparent workflows, and proper role-based access.

**2.2 Empathy Map Canvas**

We focused on the needs and frustrations of three main roles: freelancers, clients, and admins. Freelancers want timely payments and clear instructions; clients want quality work and regular updates; admins want platform integrity and user satisfaction.

**2.3 Brainstorming**

Our team brainstormed to design a system that provides a good user interface, backend connectivity, real-time updates, and role-based functionalities—using technologies like React, Node, MongoDB, and Express.

**3. REQUIREMENT ANALYSIS**

**3.1 Customer Journey Map**

* Client registers → Posts project → Reviews bids → Hires freelancer → Collaborates → Reviews work
* Freelancer registers → Browses projects → Applies to tasks → Works → Submits
* Admin monitors everything: user activity, transactions, disputes

**3.2 Solution Requirement**

The platform should allow user authentication, project posting, bidding, chatting, and admin moderation.

**3.3 Data Flow Diagram**

Frontend ↔ Express Server ↔ MongoDB. REST APIs manage communication between UI and DB.

**3.4 Technology Stack**

* Frontend: React JS, Bootstrap, Material UI, Axios
* Backend: Node.js, Express.js
* Database: MongoDB with Mongoose
* Others: Git, Visual Studio Code

**4. PROJECT DESIGN**

**4.1 Problem Solution Fit**

To address fragmented freelance workflows, SB Works integrates end-to-end processes within a single platform.

**4.2 Proposed Solution**

We built a platform with a clear interface for job postings, real-time communication, role-based access, and efficient project handling.

**4.3 Solution Architecture**

Client-Server architecture using React on the frontend, Express + Node.js backend, and MongoDB as the database.

**5. PROJECT PLANNING & SCHEDULING**

**5.1 Project Planning**

**Milestone 1:** Setup of folder structure and dependencies  
**Milestone 2:** Backend development including database, authentication, and routes  
**Milestone 3:** Frontend development and UI integration with backend

**6. FUNCTIONAL AND PERFORMANCE TESTING**

**6.1 Performance Testing**

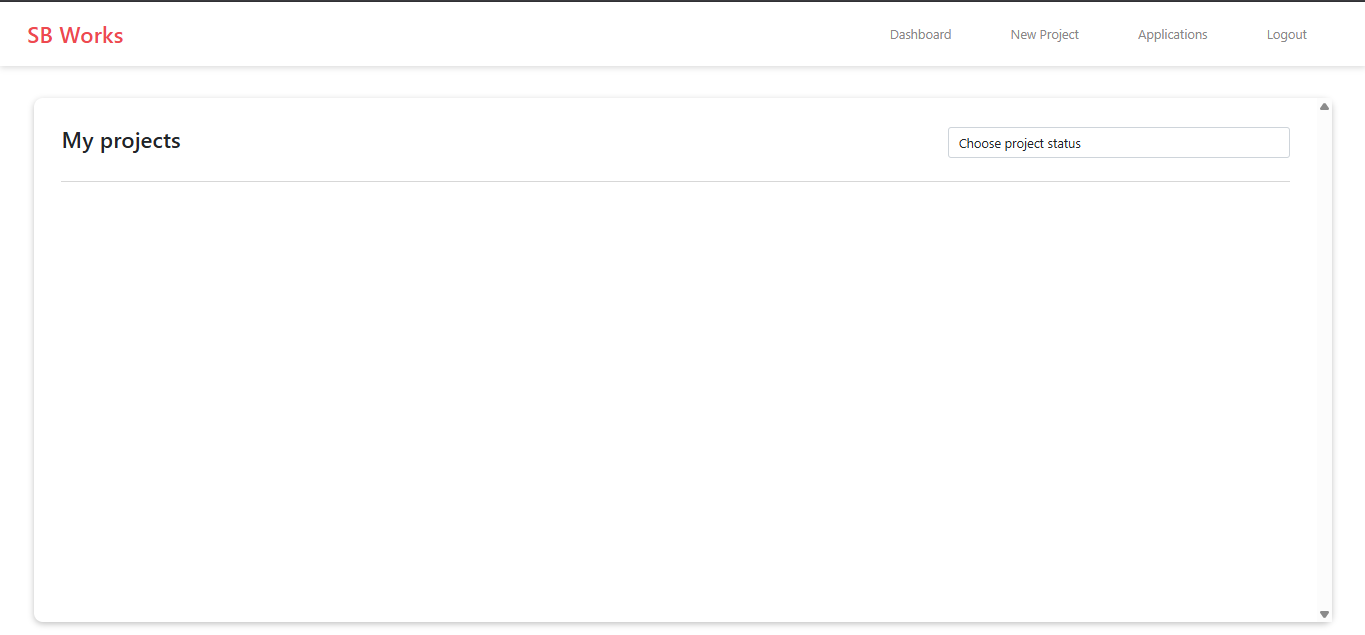
We tested the application by simulating multiple users posting and accepting tasks. CRUD operations, form validations, and authentication were tested thoroughly. MongoDB queries and Express server responses were also tested for speed.

**7. RESULTS**

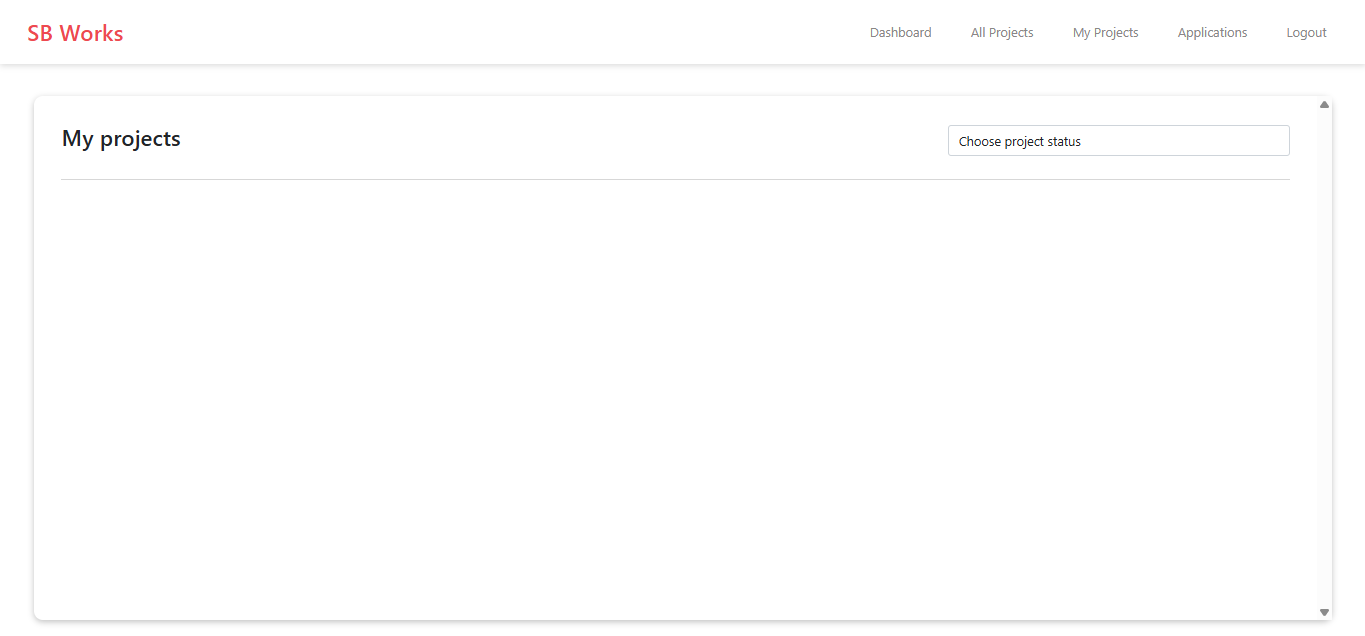
**7.1 Output Screenshots**

Screenshots included:

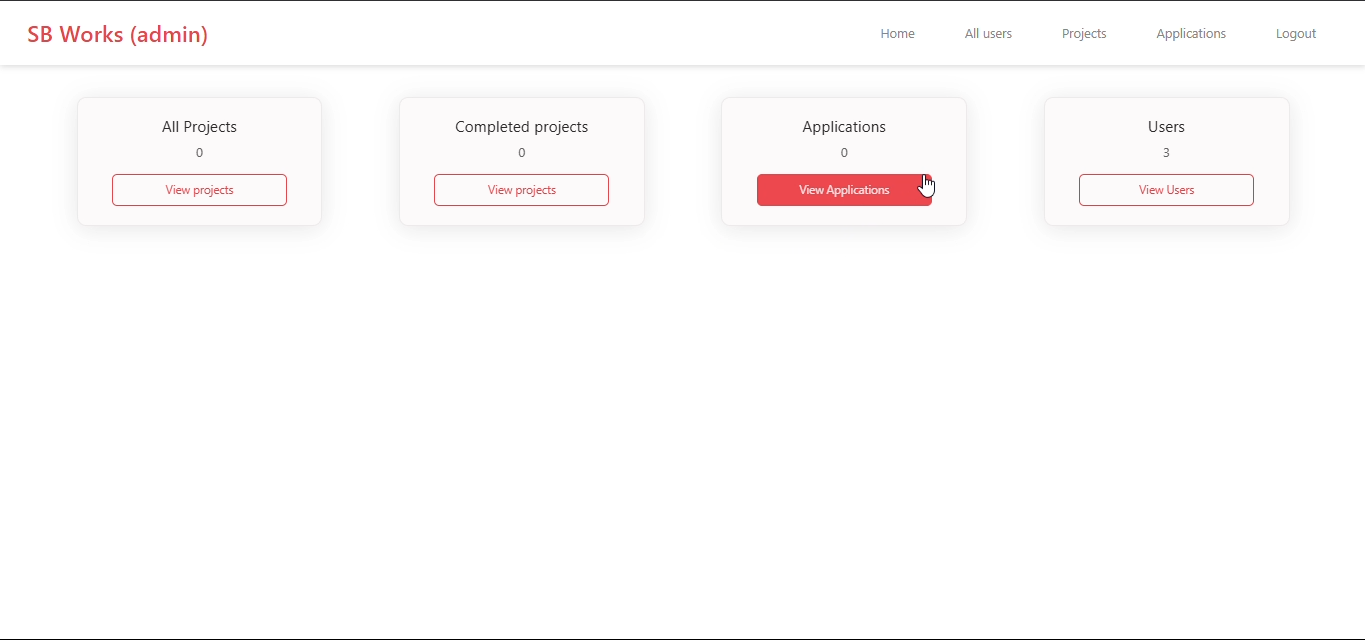
* Client dashboard



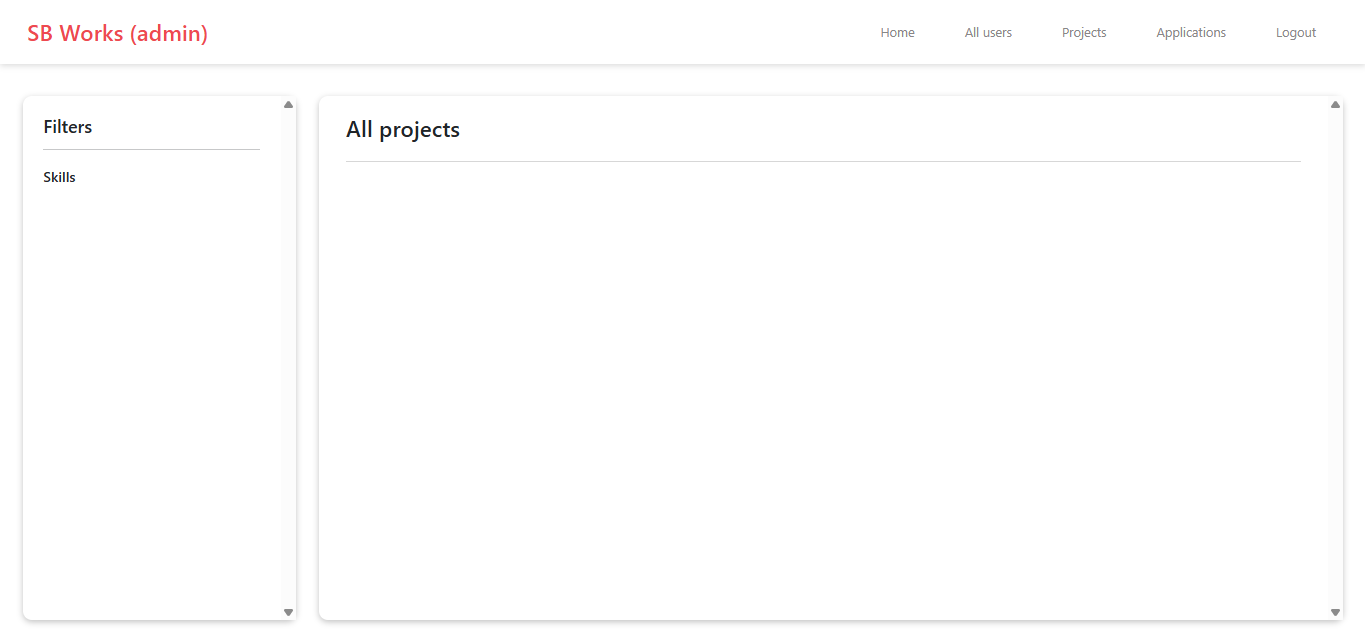
* Freelancer job listing



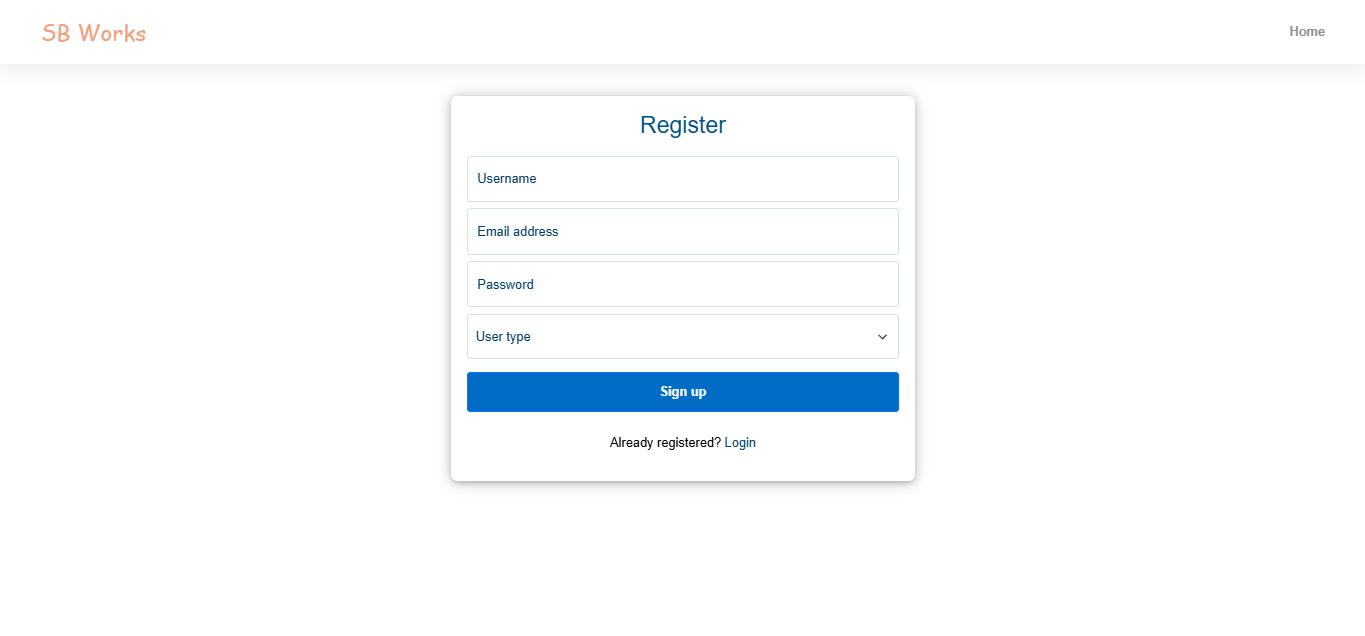
* Admin panel

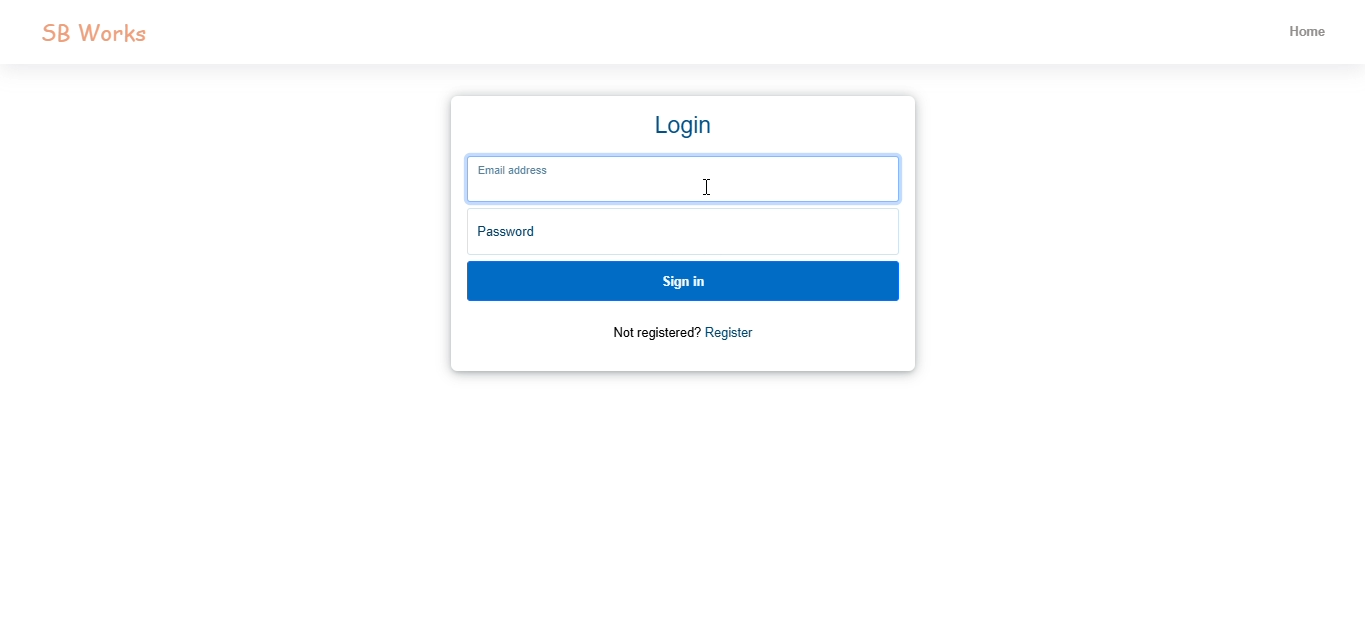


* Project details page



* Signup/Login forms





**8. ADVANTAGES & DISADVANTAGES**

**Advantages:**

* Smooth UI/UX
* Role-based access for clients, freelancers, and admins
* Real-time collaboration
* Secure login and data handling

**Disadvantages:**

* No built-in payment gateway
* Limited mobile responsiveness
* Performance may vary based on server

**9. CONCLUSION**

SB Works successfully bridges the gap between freelancers and clients through an organized, easy-to-use platform. The use of the MERN stack allows real-time interaction, modular design, and scalable performance.

**10. FUTURE SCOPE**

* Integrate payment gateway (e.g., Razorpay/Stripe)
* Add mobile app version
* Improve real-time chat using WebSockets
* AI-based recommendation of freelancers for clients

**11. APPENDIX**

* **GitHub & Project Demo Link** : <https://github.com/gubbala-santosh-kumar/Freelance-Finder/tree/main>
* **Demo Video Link:** <https://youtu.be/IqEgUFKWKbY?si=0avteM1NmCmbQKAG>