

Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	20 February 2026
Team ID	LTVIP2026TMIDS77212
Project Name	FreelanceFinder: Discovering Opportunities, Unlocking Potential
Maximum Marks	4 Marks

Technical Architecture:

The SB Works application is developed using modern full-stack technologies based on the MERN stack architecture.

1. Frontend Technologies

- React.js – For building dynamic and responsive user interfaces
- Bootstrap & Material UI – For designing attractive and responsive UI components
- Axios – For handling HTTP requests and communicating with backend APIs

2. Backend Technologies

- Node.js – Server-side runtime environment
- Express.js – Web framework for building RESTful APIs
- JWT (JSON Web Token) – For secure authentication and authorization
- Multer – For handling file uploads (project documents, attachments)

3. Database

- MongoDB – NoSQL database for storing user data, project details, bids, transactions, and reviews

Table-1: Components & Technologies:

S. No	Component	Description	Technology
1.	User Interface	Web interface for Clients, Freelancers, and Admin dashboards	React.js, HTML, CSS, Bootstrap, Material UI
2.	Application Logic-1	Authentication & Role Management (Client / Freelancer / Admin)	Node.js, Express.js, JWT
3.	Application Logic-2	Project Posting, Bidding & Management	Express.js REST APIs
4.	Application Logic-3	Notification, Review & Status Management	Node.js Controllers
5.	Database	Stores users, projects, bids, reviews, and transactions	MongoDB, Mongoose
6.	Cloud Database	Cloud-hosted NoSQL database for storing user profiles, project data, bids, and notifications with remote access and scalability	MongoDB Atlas
7.	File Storage	Project document upload & attachment storage	Local Filesystem / Cloud Storage Service
8.	External API-1	HTTP communication between frontend & backend	Axios
9.	External API-2	Payment Gateway Integration (if implemented)	—

10.	Machine Learning Model	Not used in this project	—
11.	Infrastructure (Server / Cloud)	Local development deployment & backend hosting	Node.js Local Server

Table-2: Application Characteristics:

S. No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Frameworks used to build frontend UI and backend REST APIs	React.js, Express.js, Node.js
2.	Security Implementations	Role-based authentication (Client / Freelancer / Admin), encrypted passwords, protected routes	JWT, bcryptjs, Middleware
3.	Scalable Architecture	3-Tier architecture separating frontend, backend logic, and database layer	REST Architecture, MongoDB
4.	Availability	Web application accessible anytime via browser for clients, freelancers, and admins	Node.js Server
5.	Performance	Fast API responses, asynchronous data handling, and smooth frontend-backend communication	Axios, Express.js

