**NAAN MUDHALVAN**

**PROJECT REPORT**

**Freelancing Application MERN**

TEAM MEMBER

1.SWATHI S

2.UDHAYAPRAKASH V

3.VIJAY BALA M

4.SWETHA R

**1. Overview**

**SB Works** is a robust, scalable freelancing platform designed to connect clients with skilled freelancers across a variety of industries. It emphasizes simplicity, transparency, and seamless collaboration between clients and freelancers, enabling efficient project bidding, progress tracking, communication, and secure payments—all managed through an intuitive, user-friendly interface.

**Goals of SB Works**

* **For Clients**: Provide a simple, reliable way to post projects, review freelancer portfolios, and manage work delivery.
* **For Freelancers**: Offer opportunities to showcase skills, bid on projects, and grow a reputable freelance career.
* **For Admins**: Ensure platform integrity, security, and smooth communication between clients and freelancers.

**2. Platform Features**

**2.1 Client Features**

* **Project Posting**: Clients can create projects with detailed descriptions, including budget, timelines, and specific requirements.
* **Freelancer Review**: Browse freelancer profiles, review portfolios, and assess past work.
* **Bidding System**: Evaluate bids from freelancers and select the best candidate.
* **Communication & Collaboration**: Interact with freelancers directly via integrated chat.
* **Project Feedback**: Provide ratings and feedback on completed projects.

**2.2 Freelancer Features**

* **Profile Creation**: Freelancers can showcase skills, portfolios, and work history.
* **Bidding on Projects**: Submit proposals to clients, including pricing, timelines, and past work.
* **Project Submission**: Upload completed work for review directly through the platform.
* **Payment & Ratings**: Receive payments securely and build reputation through client ratings.

**2.3 Admin Features**

* **User Management**: Oversee user profiles, verify freelancers, and maintain platform integrity.
* **Project Oversight**: Monitor active projects, ensuring compliance with platform rules.
* **Transaction Security**: Verify payments and handle disputes.
* **Analytics & Reports**: Track platform performance, user engagement, and financial metrics.

**3. User Flow**

**3.1 Client Journey**

* **Registration/Login**: The client registers and sets up their profile.
* **Post a Project**: Defines project requirements, budget, and timeline.
* **Select Freelancer**: Reviews proposals and selects a freelancer.
* **Collaborate**: Uses chat for ongoing communication.
* **Review & Pay**: Reviews submitted work, provides feedback, and completes payment.

**3.2 Freelancer Journey**

* **Registration/Login**: The freelancer creates an account and sets up a portfolio.
* **Browse Projects**: Searches for relevant projects and submits bids.
* **Communication**: Engages with clients through chat.
* **Work Submission**: Uploads deliverables for client review.
* **Receive Feedback & Payment**: Gains client feedback and completes payment.

**4. Technical Architecture**

**4.1 System Architecture Overview**

SB Works follows a **Client-Server Model** with a **three-tier architecture** comprising:

* **Frontend**: Built with React for a responsive and dynamic UI.
* **Backend**: Powered by Express.js, handling API requests, business logic, and database interaction.
* **Database**: MongoDB for flexible, schema-less data storage.

**4.2 Frontend**

* **Framework**: React
* **Libraries**:
* **Bootstrap** and **Material UI** for styling and layout.
* **Axios** for handling HTTP requests to the backend.
* **Features**:
* Real-time updates on project listings.
* Notification system for project updates, bid statuses, and chat messages.
* **UI Components**: Modular UI components include dashboards, forms, bid management, and chat.

**4.3 Backend**

* **Framework**: Express.js for efficient handling of server-side logic and routing.
* **APIs**: RESTful APIs to manage users, projects, bids, and communication.
* **Security**:
* **JWT Authentication**: Secures user sessions and prevents unauthorized access.
* **HTTPS and Data Encryption**: Ensures secure data transmission.
* **Business Logic**:
* Project listing and bid management.
* Secure payment handling and project completion workflows.
* **Real-time Communication**:
* **WebSocket Integration**: Enables live messaging between clients and freelancers.

**4.4 Database**

* **Database**: MongoDB
* **Schema Design**:
* **User Collection**: Stores user profiles, ratings, and transaction history.
* **Project Collection**: Contains project details, timelines, and budget.
* **Bid Collection**: Maintains records of all bids submitted by freelancers.
* **Message Collection**: Archives chat messages for project-specific communication.

**4.5 Third-Party Integrations**

* **Payment Gateway**: Stripe/PayPal integration for secure, trackable payments.
* **Notification Service**: Push notifications for project updates and chat messages.
* **Real-Time Analytics**: Dashboard for admin to monitor key metrics, integrated with Google Analytics.

**5. Scenario-Based Case Study**

**Scenario: Sarah’s Journey on SB Works**

* **User Story**: Sarah, a graphic design graduate, joins SB Works to build her freelance career. She finds a logo design project posted by "Sugar Rush Bakery."
* **Project Selection**: Sarah uses the platform to understand project needs, reviews "Sugar Rush's" existing brand materials, and submits her bid.
* **Collaboration**: Sarah and the client interact through the SB Works chat system to align on design concepts.
* **Work Submission**: Sarah uploads her final design via SB Works. The client provides feedback, leading to a successful project completion.
* **Outcome**: Sarah receives payment and a positive review, boosting her profile's visibility on SB Works.

**6. Detailed API Documentation**

**6.1 Authentication**

* **Endpoint**: /api/auth/login
* **Method**: POST
* **Description**: Logs in a user.

**Request Body**:  
json  
Copy code  
{

"username": "string",

"password": "string"

}

**Response**:  
json  
Copy code  
{

"token": "jwt-token-string",

"userId": "string"

}

**6.2 Project Management**

**Create Project**

* **Endpoint**: /api/projects
* **Method**: POST
* **Description**: Creates a new project.

**Request Body**:  
json  
Copy code  
{

"title": "string",

"description": "string",

"budget": "number",

"deadline": "date"

}

**Response**:  
json  
Copy code  
{

"projectId": "string",

"status": "success"

}

**6.3 Bid Management**

**Submit Bid**

* **Endpoint**: /api/projects/{projectId}/bids
* **Method**: POST
* **Description**: Freelancer submits a bid on a project.

**Request Body**:  
json  
Copy code  
{

"freelancerId": "string",

"bidAmount": "number",

"proposal": "string"

}

**Response**:  
json  
Copy code  
{

"status": "success",

"bidId": "string"

}

**6.4 Messaging**

**Send Message**

* **Endpoint**: /api/messages
* **Method**: POST
* **Description**: Sends a message in the project chat.

**Request Body**:  
json  
Copy code  
{

"senderId": "string",

"projectId": "string",

"message": "string"

}

**Response**:  
json  
Copy code  
{

"status": "sent",

"messageId": "string"

}

**7. Security & Compliance**

* **User Authentication**: JWT-based secure login and session management.
* **Data Encryption**: All sensitive information is encrypted using AES.
* **Transaction Logging**: Every transaction and message exchange is logged for security auditing.
* **Compliance**: GDPR-compliant data handling for EU-based users.

**8. Deployment & Maintenance**

**8.1 Deployment Strategy**

* **Cloud Provider**: AWS/Google Cloud Platform
* **Load Balancer**: NGINX or AWS ELB for distributing requests.
* **Containers**: Docker containers for microservices.
* **Monitoring**: Prometheus and Grafana for performance monitoring and alerts.

**8.2 Maintenance**

* **Automated Backups**: Daily backups for database and project files.
* **Logging**: Application and error logs stored in a centralized repository.
* **Scalability**: MongoDB’s sharding and Express.js’s modularity allow horizontal scaling.

This platform is designed to provide an efficient and secure environment for freelancers and clients alike, streamlining project collaboration and ensuring professional integrity throughout the freelancing process.