Project Report: Freelance Bidding Platform

Author : PREMSAGAR BANOTH

Introduction

The Freelance Bidding Platform (Mini Fiverr Clone) is a web application designed to connect clients and freelancers, enabling service-based job postings, bidding, messaging, and contract acceptance. This report documents the development process and key components of the project.

Abstract

This project leverages the MERN (MongoDB, Express, React, Node.js) stack to build a scalable, user-role-based bidding platform. Clients can post jobs and review bids, while freelancers can submit proposals and communicate through an integrated messaging system. Future enhancements include payment integration and notification systems.

Tools Used

- MongoDB: NoSQL database for storing user, job, and message data.

- Express.js: Backend framework for building RESTful API endpoints.

- React.js: Frontend library for building interactive user interfaces.

- Node.js: JavaScript runtime environment for server-side code.

- Mongoose: ODM for MongoDB schema modeling and data validation.

- React Router: Client-side routing and navigation.

- Axios: HTTP client for API communication.

- Tailwind CSS (optional): Utility-first CSS for styling.

Steps Involved in Building the Project

- Project Setup: Initialize client and server directories with package.json.

- Database Design: Define schemas for User, Job, Bid, and Message models.

- Authentication: Implement JWT-based signup/login and protected routes.

- Job Posting & Bidding: Build API endpoints and React pages to post jobs and place bids.

- Messaging System: Develop real-time or REST-based chat features with edit/delete.

- UI/UX Enhancements: Incorporate a modern CSS framework and responsive design.

- Testing: Verify API endpoints using Postman and conduct browser testing.

- Deployment Configuration: Set up environment variables and build scripts for production.

Conclusion

The Freelance Bidding Platform successfully demonstrates a full-stack application with role-based access, dynamic job bidding, and messaging functionality. The modular codebase and clear separation of concerns ensure maintainability and scalability. Future work can focus on payment integration, notification services, and performance optimizations.