JEPPIAAR ENGINEERING COLLEGE

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

FREELANCING APPLICATION USING MERN

TEAM LEADER : ANUSUYA.B (310821104011)

TEAM MEMBER 1 : DHANALAKSHMI.B (310821104024)

TEAM MEMBER 2 : UMA MAHESHWARI.S (310821104104)

TEAM MEMBER 3 : MEGANA.S (310821104055)

SUPERVISOR NAME : Mrs.D.JEEVITHA M.E..,

ASSISTANT PROFESSOR ,

DEPARTMENT OF CSE.

INTRODUCTION :

- Freelancing is the reshaping modern work culture with flexibility and goal access.
- Our app bridges the gap between freelancers and clients effortlessly.
- Features like smart job matching , secure payments , and collaboration tools.
- Focused on creating a trustworthy and efficient platform for all users.

PROJECT OVERVIEW :

Objective :

Develop a user – friendly platform to connect freelancers and clients.

Key Features:

- Simple registration and profile creation for users.
- Basic job posting and application functionalities

Target Users :

Freelancers and businesses across various industries.

Outcome :

Simplify freelancing processes, ensuring accessibility and reliability

ARCHITECTURE:

The freelancing application is built using three parts :

Frontend (client) :

A react app users interact with for tasks like logging in, browsing projects and managing dashboards.

Send requests to the backend for data or actions.

Backend (Server) :

A Node.js/Express server that handles all business logic process requests from the frontend (eg: login checks, project updates) and sends responses.

Database :

Users : Stores user data.

Freelancer / Admin : Stores their profiles and works.

Project list : Stores new project, completed projects and active projects.

How it works :

User Interaction :

Users use the app to log in, browse projects and manage profiles.

Data Requests :

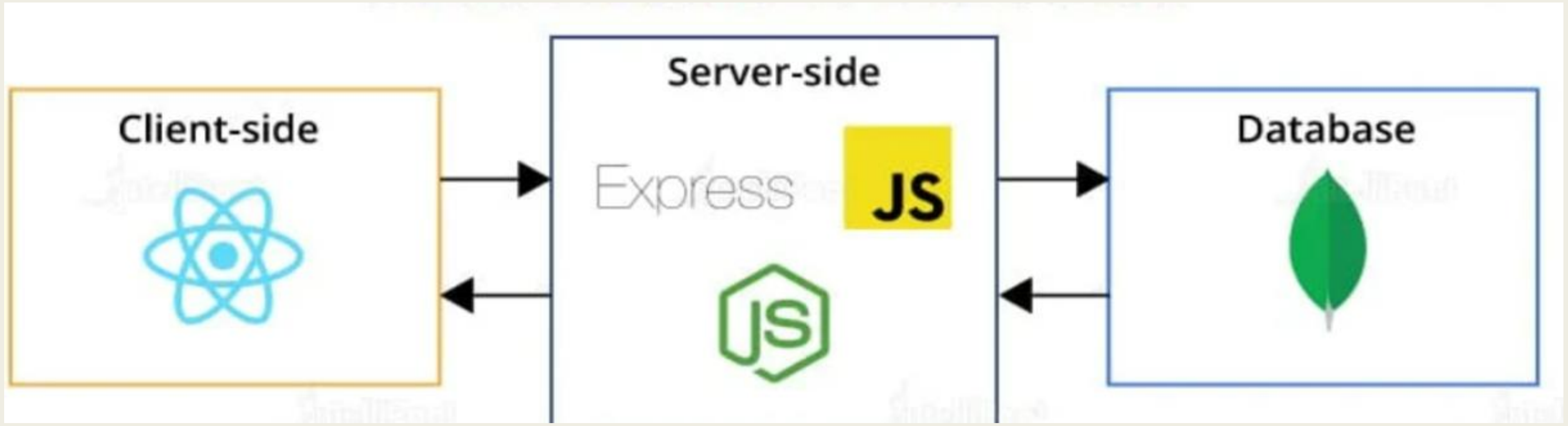
The app (frontend) send requests to the server (backend) for data or actions

Server Processing :

The backend processes requests and retrieves or updates data in the database.

Response :

The server sends the data back to the app to update what the user sees.



Architecture Diagram

Set Up Instruction

1 . Pre requisties :

Install Node.js , MongoDB and Git .

2 . Clone Repository :

git clone https://github.com/freelancing-apllication-mern.git

cd freelancing application mern

3 . Install Dependencies :

Backend (Server) – cd server

npm install

Frontend (Client) – cd ../client

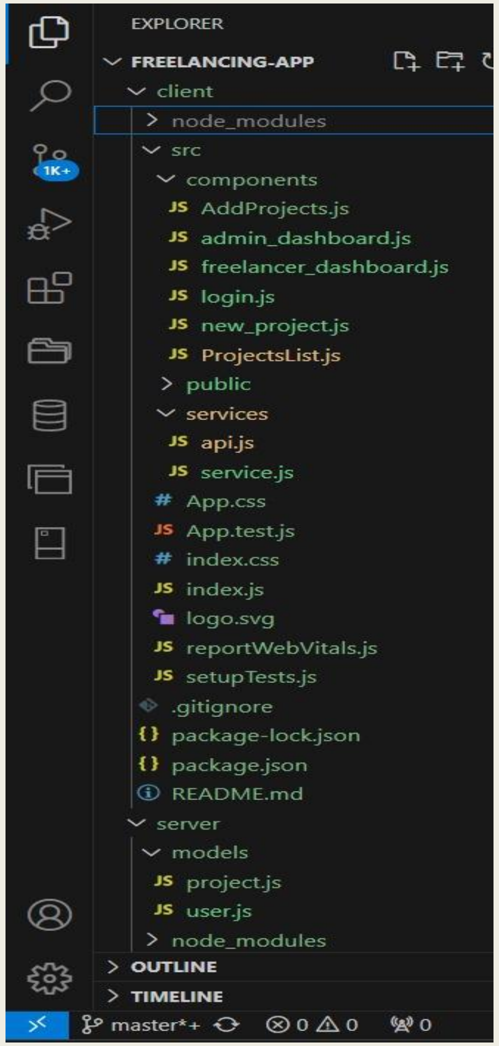
npm install

4 . Setup environment :

.env

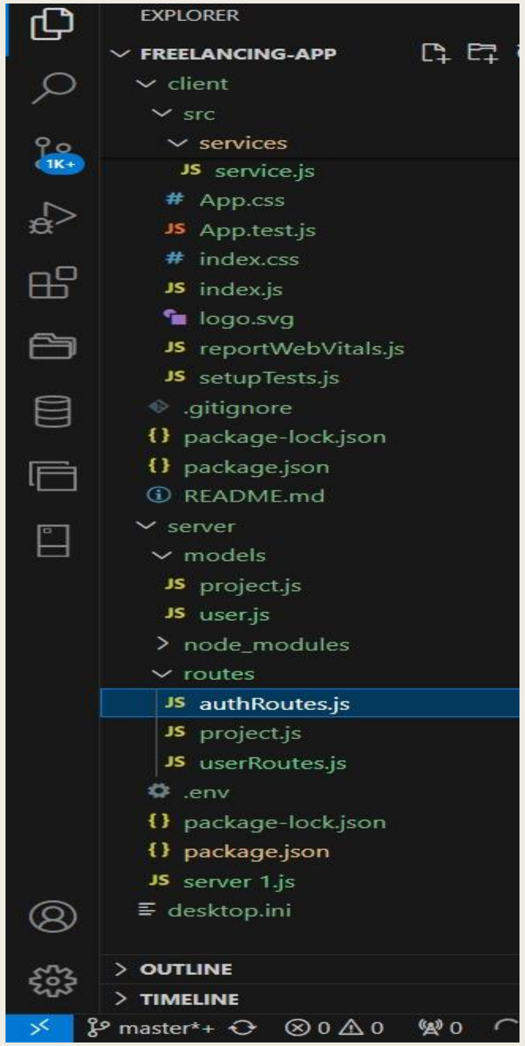
FOLDER STRUCTURE :

Frontend



```
EXPLORER
∨ FREELANCING-APP
  ∨ client
    > node_modules
    ∨ src
      ∨ components
          AddProjects.js
          admin_dashboard.js
          freelancer_dashboard.js
          login.js
          new_project.js
          ProjectsList.js
      > public
      ∨ services
          api.js
          service.js
        App.css
        App.test.js
        index.css
        index.js
        logo.svg
        reportWebVitals.js
        setupTests.js
      .gitignore
      package-lock.json
      package.json
      README.md
  ∨ server
    ∨ models
        project.js
        user.js
    > node_modules
> OUTLINE
> TIMELINE
master*+  ⊗ 0 ⚠ 0  0
```

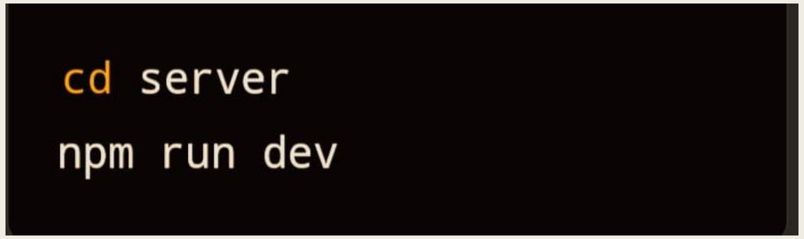
Backend



```
EXPLORER
∨ FREELANCING-APP
  ∨ client
    ∨ src
      ∨ services
          service.js
        App.css
        App.test.js
        index.css
        index.js
        logo.svg
        reportWebVitals.js
        setupTests.js
      .gitignore
      package-lock.json
      package.json
      README.md
  ∨ server
    ∨ models
        project.js
        user.js
    > node_modules
    ∨ routes
        authRoutes.js
        project.js
        userRoutes.js
      .env
      package-lock.json
      package.json
      server 1.js
    desktop.ini
> OUTLINE
> TIMELINE
master*+  ⊗ 0 ⚠ 0  0
```

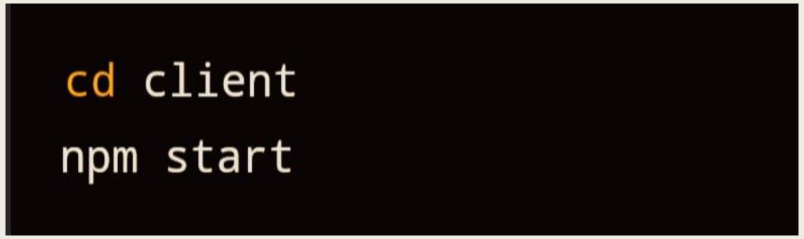
RUNNING THE APPLICATION :

Start the server (Backend):



```
cd server
npm run dev
```

Start the client (Frontend):



```
cd client
npm start
```

Access the Application :

Frontend : http://localhost:3000

Backend : http://localhost:5000

API DOCUMENTATION :

An API (Application Programming Interface) is like a bridge that helps frontend and backend to communicate with each other.

Tools used : Express.js, Mongo DB

Endpoints :

User management :

POST / register : Register new users

POST / login : Authenticate users

GET / profile : Fetch user details

Project listings :

GET / projects : Retrieve available projects

POST / projects : Add a new project

PUT / projects/:id Update project details

AUTHENTICATION :

Authentication ensures only registered users can access the application. It verifies a user's identity with credentials like email and password.

User Registration

Purpose: Create a new user account.

Example:

Input: Name: "John", Email: "john@example.com", Password: "123456“

Output: "Registration successful!“

User Login

Purpose: Access user account.

Example:

Input: Email: "john@example.com", Password: "123456“

Output: "Login successful!“

Session Management

Purpose: Maintain user authentication during the session.

Process:

JWT (JSON Web Token) issued on login.

Token is stored in the browser (e.g., local storage).

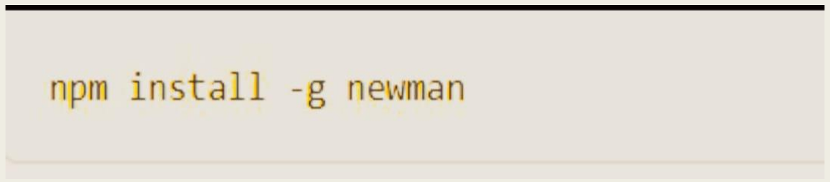
TESTING :

Manual Testing:

Purpose: Test the app like a real user (clicking buttons, filling forms).
Tools used: No tools required

API Testing:

Purpose: Ensure backend endpoints work (e.g. /login, /projects).
Tools used: Postman.

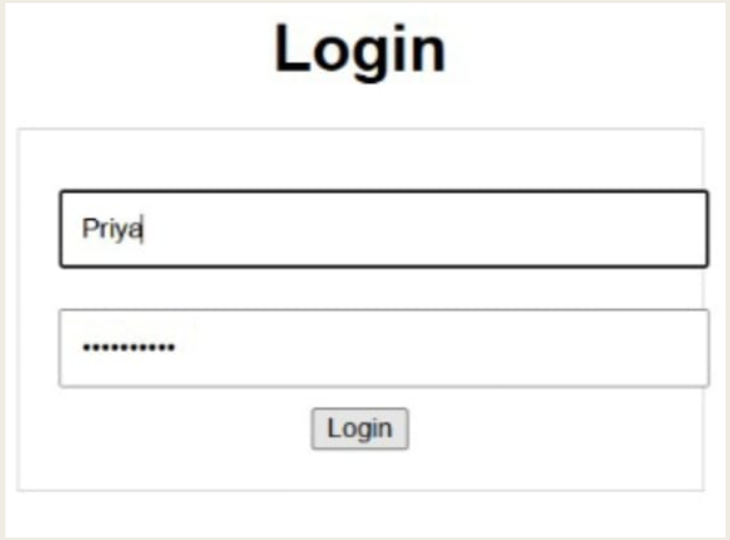


```
npm install -g newman
```

UI Testing:

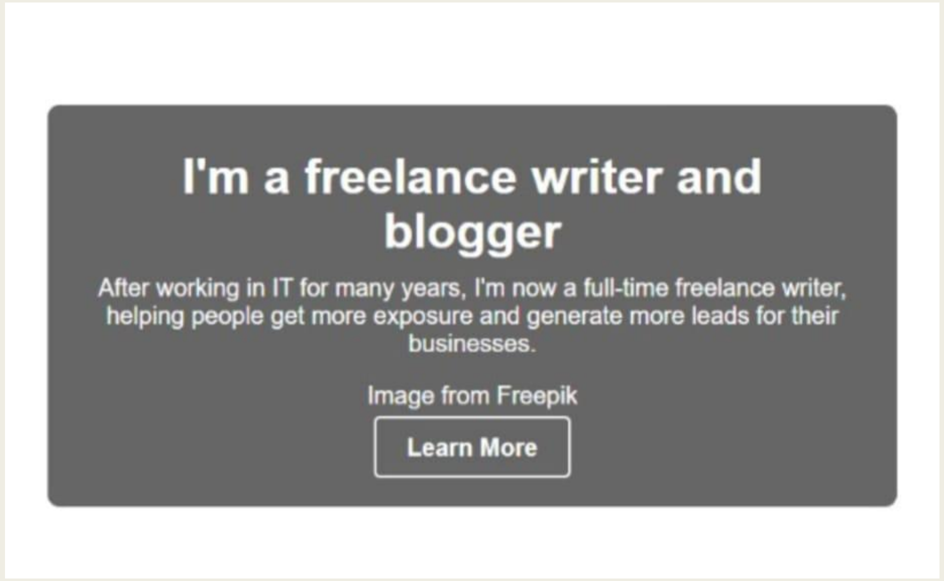
Purpose: Verify the interface looks and works correctly (buttons, forms, links).
Tools used: Browser Developer Tools

OUTPUT :


Login

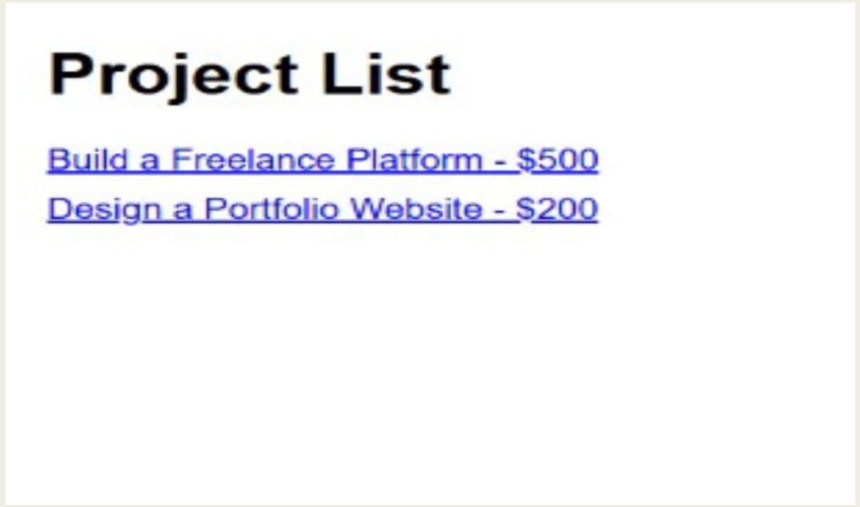
Priya

•••••••••••


I'm a freelance writer and blogger

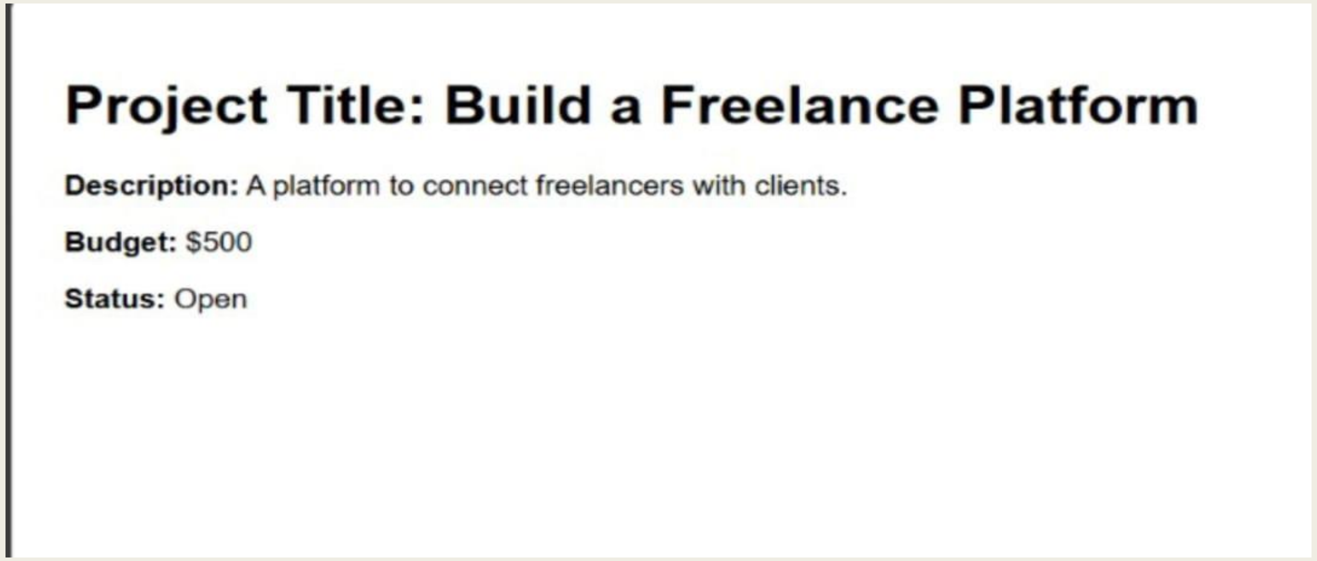
After working in IT for many years, I'm now a full-time freelance writer, helping people get more exposure and generate more leads for their businesses.

Image from Freepik


Project List

Build a Freelance Platform - $500

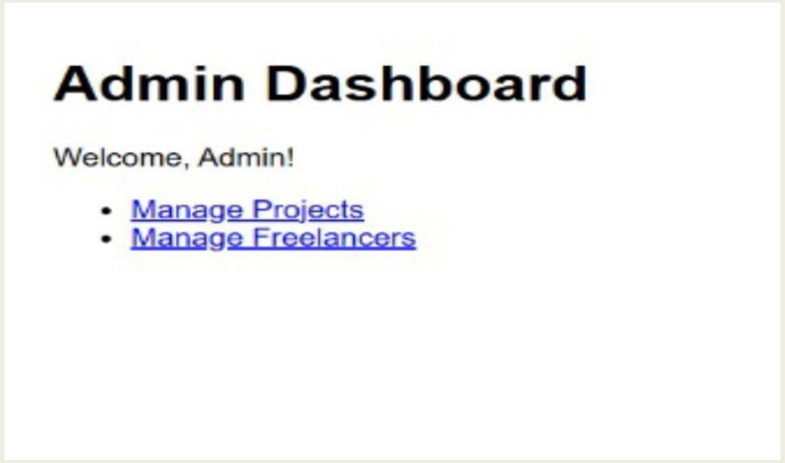
Design a Portfolio Website - $200


Project Title: Build a Freelance Platform

Description: A platform to connect freelancers with clients.

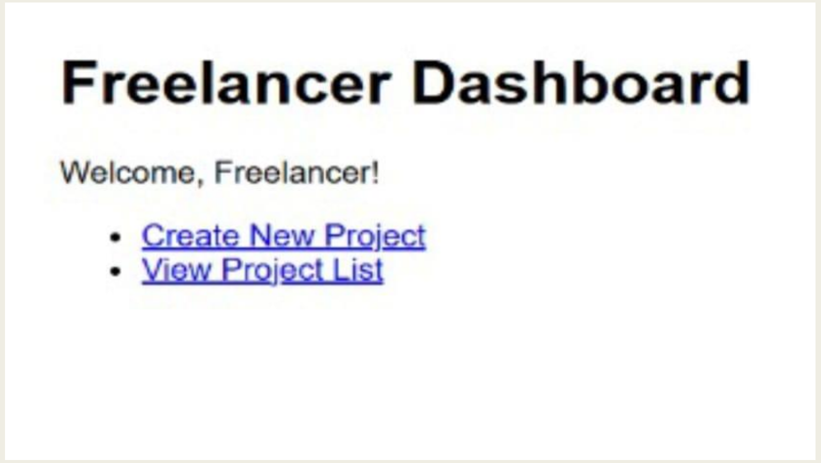
Budget: $500

Status: Open


Admin Dashboard

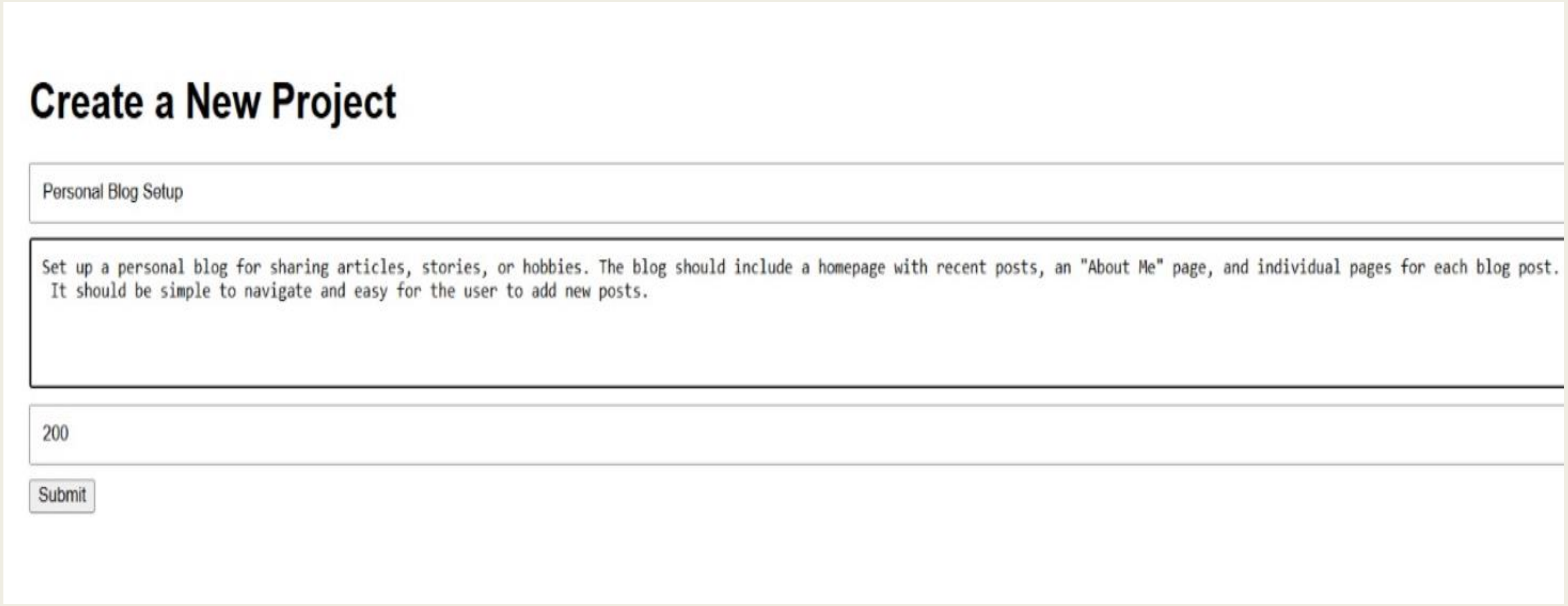
Welcome, Admin!

- Manage Projects
- Manage Freelancers


Freelancer Dashboard

Welcome, Freelancer!

- Create New Project
- View Project List


Create a New Project

Personal Blog Setup

Set up a personal blog for sharing articles, stories, or hobbies. The blog should include a homepage with recent posts, an "About Me" page, and individual pages for each blog post.
It should be simple to navigate and easy for the user to add new posts.

200

FUTURE ENHANCEMENTS :

1 . Payment Gateway

2 . AI Recommendation

3 . Rating System

4 . Dark Mode Customization

Thank You