SKYSCALP_1

Online Web Platform for Property Marketplace

PROJECT SUMMARY

Introduction

Our platform, built with cutting-edge technologies like React, Node.js, MongoDB, and Express, is designed to revolutionize the way you buy and sell property and construction materials. Whether you're looking to purchase your dream home, sell commercial real estate, or find high-quality construction supplies, our user-friendly marketplace has got you covered.

Key Features:

- **Buy and Sell with Ease**: Our intuitive interface allows users to effortlessly browse, filter, and purchase a wide range of properties and construction materials. Listing items for sale is quick and simple, making it easy for sellers to reach a broad audience.
- **Advanced Search and Filters**: Find exactly what you're looking for with our powerful search and filtering options. Narrow down your choices based on location, price, property type, material quality, and more.
- **User-Friendly Admin Panel**: Each user gets access to a dedicated admin panel, providing a comprehensive overview of their listings. Track your products, manage received messages, and keep an eye on your favorite items all in one place.
- **Messaging System**: Communicate directly with buyers or sellers through our integrated messaging system. This feature ensures clear, direct, and efficient negotiations and inquiries.
- **Favorites and Tracking**: Keep track of your preferred listings by adding them to your favorites. This makes it easy to monitor price changes and availability of the items you're interested in.

Technologies Stack:

Frontend:

- **React.js**: Utilized for building the dynamic and interactive user interface, React.js allows us to create reusable components and manage the application's state effectively.
- **Bootstrap**: Integrated for responsive and mobile-first design, Bootstrap ensures that our platform is visually appealing and accessible across a wide range of devices.

Backend:

- **Node.js**: Serving as the runtime environment, Node.js allows us to build scalable and high-performance server-side applications.
- **Express.js**: Used as the web application framework, Express.js simplifies the development of robust APIs and handles routing efficiently.

Database:

- **MongoDB**: Chosen for its flexibility and scalability, MongoDB stores our data in a NoSQL format, allowing for efficient data management and quick retrieval of information.

Code/Folder Structure:

FRONTEND:

The project is organized with a main **`src`** directory that houses all the core application code for the React frontend. Within **`src`**, the **`components`** folder contains reusable components that can be utilized across different parts of the application, ensuring modularity and reusability. The **`pages`** directory includes all the primary page components, each representing different views or sections of the application.

The **`admin`** folder contains its own set of pages tailored for administrative tasks, enabling administrators to manage the platform efficiently. Inside the **`pages`** directory of **`admin`** folder, there is a specialized **`Owner`** folder that holds private admin pages exclusively for property or material owners, providing them with dedicated functionalities.

Additionally, the **`assets`** folder is designated for static resources such as CSS files and media images, ensuring that all styling and visual assets are organized and easily accessible. This structured approach ensures clear separation of concerns and enhances maintainability and scalability of the codebase.

BACKEND:
The Node.js backend is structured following the MVC architecture to ensure a clear separation of concerns and facilitate maintainability and scalability.

The **`models`** directory contains the data models, defining the structure and schema for various entities used in the application, such as users, properties, and materials.

The **`controllers`** directory houses the business logic and functions for handling incoming API requests, interacting with the models, and sending appropriate responses to the client.

The **router** file has all the paths or endpoints needed to implement or call the api.

This setup ensures that the data handling logic is separated from the request handling logic, promoting clean and organized code. By following this structure, the application maintains a modular and organized codebase, making it easier to manage and extend.

This structured approach with React on the frontend and Node.js on the backend provides a robust foundation for developing and scaling your online marketplace efficiently.

Media/Images Management:

To streamline image management and enhance user experience, our platform integrates with imgBB.com, a reliable third-party image hosting service. By utilizing imgBB.com, we ensure efficient and secure handling of images uploaded by users, including property photos and construction material images. This integration allows users to easily upload, store, and retrieve images without burdening our own server resources, thereby improving performance and scalability. The use of imgBB.com ensures that all images are hosted on a dedicated platform, providing fast loading times and robust storage solutions. This approach not only simplifies image management but also enhances the overall functionality and user experience of our marketplace.

Code Hosting:

Our platform's code is hosted on GitHub, providing a reliable and collaborative environment for development. By utilizing GitHub, we ensure version control, code management, and seamless collaboration among our development team. Our repository is publicly accessible and can be found at https://github.com/muhammad-atif-74/skyscalp. This setup facilitates continuous integration and deployment, allowing us to manage updates and improvements efficiently while maintaining a high standard of code quality and project organization.

Project Timeline:

The project is completed in the duration of almost 2 and half month span, and development was done in 3 phases/milestones.

1. Design

19 March 2024 – 26 March 2024

Source: https://www.figma.com/design/UuPwDwgXN0yD0dZnc9kMU8/Skyscalp?node-id=0-1&t=jkCoAlxQZ4HuvvW9-1

2. Frontend Development

27 March 2024 – 28 April 2024

Source: Github repository https://github.com/muhammad-atif-74/skyscalp.

3. Backend Development

29 April 2024 – 26 May 2024

Source: Github repository https://github.com/muhammad-atif-74/skyscalp.

Conclusion

In summary, our online marketplace for buying and selling property and construction materials is built using a modern and robust technology stack. The frontend is developed

with React.js and Bootstrap, ensuring a dynamic, responsive, and user-friendly interface. The backend is powered by Node.js and Express.js, providing a scalable and high-performance server environment. Data management is handled by MongoDB, offering flexible and efficient storage solutions. We leverage imgBB.com for secure and efficient image hosting, enhancing the overall user experience. Our code is hosted on GitHub, facilitating version control, collaborative development, and continuous integration. This well-structured and thoughtfully organized approach ensures a seamless, reliable, and scalable platform for our users.