**PROJECT REPORT SYNOPSIS**

**ON**

**Estate Edge**

**SUBMITTED TO**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**FOR**

**Full Stack Engineering(22CS037)**

****

**Submitted By:**

**Name(s):** Gunjan, Arpita, Arshika

**University Roll No(s).:** 2210991588, 2210991359, 2210991369

**Semester:**6th

**Session:** 2022-2026

|  |  |
| --- | --- |
|  |  |

**Index**

# **Sr. no Topic Page No**

1. Problem Statement 3
2. Title of project 3
3. Objective & Key Learning’s 3
4. Options available to execute the project 4
5. Advantages/ Disadvantages 4
6. References 6

**Problem Statement:**

The goal is to develop an efficient real estate management platform, "Estate Edge", where administrators can list and manage properties, and users can browse, inquire, and access detailed information about various properties. The platform will simplify the property management process and offer a seamless browsing experience to potential buyers or renters, improving accessibility and user engagement in the real estate sector.

**Title of project:**

Estate Edge

**Objective:**

* Build a property listing system: Allow administrators to list properties with detailed descriptions, photos, and pricing.
* Implement a user-friendly interface: Provide clients with an intuitive platform to browse, search, and filter properties by various criteria such as location, price, and type.
* Create an efficient property management tool: Allow administrators to manage property data, update listings, and handle inquiries in real time.

**Key Learning’s:**

1. **User Management:**

Secure user authentication and authorization for renters and property owners. Role-based access control for renters, property managers, and support staff.

1. **Rental Processing and Management:**

APIs for handling rental inquiries, bookings, updates and cancellations. Real-time booking status tracking and notifications for both renters and property owners.

1. **Communication and Notifications:**

APIs for sending in-app messages, email notifications, and real-time updates on rental statuses. Tools for collecting and analysing renter feedback and property reviews.

**Options available to execute the project:**

* Programming Languages:
  + React.js: A front-end JavaScript library for building interactive user interfaces.
  + Node.js: A JavaScript runtime that powers the server-side, enabling scalable backend operations.
* Backend Frameworks:
  + Express.js (Node.js): A minimal web application framework used for creating the API and routing.
  + Mongoose: An ODM (Object Data Modeling) tool for MongoDB to manage database schemas.
* APIs:
  + Use standard RESTful APIs for client-server communication.
* Version Control:
  + GitHub: A version control system for collaboration, code review, and project management.
* Database:
  + MongoDB: A NoSQL database known for its high performance and scalability, ideal for managing real estate listings and user data.

**Advantages/ Disadvantages:**

* **Advantages:**
* Real Estate Market Efficiency:
  + Enhanced Property Management: Estae Edge allows property administrators to manage listings efficiently and in real-time, leading to better control over the properties.
  + Client Convenience: Clients can quickly browse properties, inquire, and make bookings directly, simplifying the rental or purchase process.
* Scalability and Flexibility:
  + Database Efficiency: The use of MongoDB ensures that the platform can handle large datasets and user traffic without performance bottlenecks.
  + APIs for Future Integration: The platform can easily integrate new features, such as online payment gateways or third-party real estate services.
* User Engagement:
  + Interactive Features: Features such as property search filters, inquiry systems, and interactive interfaces will keep users engaged, improving the likelihood of client conversion.
  + Data-Driven Insights: Administrators can access insights on user behavior, property views, and inquiries, helping them optimize listings.
* **Disadvantages:**
* Technical Complexity:
  + Property Management System: Implementing a real-time system for managing multiple property listings could be complex, requiring regular updates, maintenance, and debugging.
  + API Integration: Reliance on external services for inquiries, property data updates, and real-time management increases the complexity of backend development.
* Security Concerns:
  + User Data Protection: Handling sensitive user information (such as inquiries and booking details) requires robust security protocols to prevent data breaches.
  + Platform Security: The platform must implement secure authentication and authorization methods to prevent unauthorized access.
* Competitive Market:
  + Market Saturation: The real estate market is competitive, with platforms like Zillow and 99acres dominating. Differentiating Estate Edge from existing platforms will be key to attracting users.
  + Logistical Challenges: Managing bookings, inquiries, and property availability at scale can be challenging as more users join the platform.

**References:**

* 1. Node.js. (n.d.). Introduction to Node.js. Retrieved August 15, 2024, from https://nodejs.org/en/learn/getting-started/introduction-to-nodejs

2. React. (n.d.). Getting Started. Retrieved August 15, 2024, from https://reactjs.org/docs/getting-started.html

3. Mongoose. (n.d.). Mongoose v6.1.0: Documentation. Retrieved August 15,2024, from https://mongoosejs.com/docs/

4. Express. (n.d.). Installing. Retrieved August 15,2024, from https://expressjs.com/en/starter/installing.html

5. MongoDB. (n.d.). MongoDB Atlas. Retrieved August 15, 2024, from https://www.mongodb.com/cloud/atlas

6. Mozilla. (n.d.). JavaScript. Retrieved August 15,2024, from https://developer.mozilla.org/en-US/docs/Web/JavaScript

7. npm. (n.d.). npm. Retrieve August 15, 2024, from https://www.npmjs.com/