# Annexure-III

**PROJECT SYNOPSIS REPORT ON**

**Air Bnb**

**SUBMITTED**

**TO**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING FOR**

**Full Stack Engineering**



**Submitted To: Submitted By:**

Mr. Rahul Ishika Goel (2210991691)

Ishneet Kaur (2210991690)



**Index**

**Sr. no Topic Page No**

1. Problem Statement 01
2. Title of project 01
3. Objective & Key Learning’s 01
4. Options available to execute the project 02
5. Advantages/ Disadvantages 03
6. References 04



**1.** **Problem Statement**

In today's dynamic travel and hospitality industry, property owners and travelers face numerous challenges in managing bookings and accommodations effectively. For property owners, fragmented platforms, cumbersome listing processes, and difficulties in tracking bookings and payments often lead to inefficiencies and lost opportunities. On the other hand, travelers struggle with inconsistent user experiences, unclear property details, and unreliable communication with hosts, making the booking process stressful and time-consuming.

For both property owners and travelers seeking a streamlined solution, the challenge is finding a platform that provides clarity and efficiency without unnecessary complexity. Whether it's listing properties, browsing personalized accommodation options, managing bookings, or facilitating secure payments, users need an intuitive, user-friendly interface that simplifies the rental process and enhances their experience, allowing them to focus on what truly matters: creating memorable travel experiences.

**2. Title of Project**

"AIR BNB”

**3. Objective & Key Learning’s**

**Objective**:

The objective The objective of the Airbnb project is to develop a simple yet effective vacation rental platform that enables hosts and travelers to:

* Efficiently manage property listings, bookings, and payments.
* Organize reservations and tasks into upcoming and completed categories for better tracking.
* Prioritize key features such as secure communication and timely responses to enhance user satisfaction.
* Offer an intuitive, user-friendly interface that simplifies the booking experience and boosts engagement for both hosts and travelers.

**Key Learnings:**

* Task Structuring and Organization: Understand the importance of organizing tasks such as property listings, bookings, and payments into actionable segments to streamline workflows for hosts and travelers.



* 2.Backend-Frontend Integration: Explore how backend systems (e.g., property, booking, and user data in MongoDB) integrate seamlessly with a user-friendly frontend interface, ensuring smooth operations on the vacation rental platform.
* 3. User-Centric Design Principles: Emphasize intuitive navigation and user-centric features such as marking bookings as confirmed, canceled, or completed, ensuring the application caters to the needs of both hosts and travelers.
* 4. Data Relationships in Databases: Learn how to establish and manage relationships (e.g., host-property or booking-traveler) within the database using schemas and object models to ensure accurate tracking and updates.

**4. Options Available to Execute the Project**

In our project, we are leveraging the MERN stack, which stands for MongoDB, Express.js, React.js, and Node.js. This powerful combination allows us to build a full-stack application with a seamless integration between the front-end and back-end. MongoDB provides a flexible NoSQL database solution, while Express.js and Node.js handle the server-side logic, making the application highly efficient. React.js is used for developing a dynamic and responsive user interface, ensuring a smooth and interactive shopping experience for users. Utilizing the MERN stack enables me to create a robust, scalable, and maintainable e-commerce platform.

**5. Advantages/Disadvantages**

**Advantages :**

The MERN stack, comprising MongoDB, Express.js, React.js, and Node.js, offers a robust solution for full-stack development by enabling developers to use JavaScript across the entire application.

This consistency simplifies the development process and reduces the learning curve, as developers can work on both the front-end and back-end with a unified language.

React.js allows for the creation of dynamic and responsive user interfaces, while Node.js, with its non-blocking architecture, ensures high performance and scalability for handling concurrent requests.

The rich ecosystem and large community support around each component of the MERN stack further enhance its appeal, providing access to numerous libraries, tools, and resources that can accelerate development.



**Disadvantages:**

However, the MERN stack also comes with certain challenges. The learning curve can be steep, as developers need to master multiple technologies to fully leverage the stack.

MongoDB, being a NoSQL database, may not be ideal for applications requiring complex relational data and transactions. Additionally, React.js, while excellent for single-page applications, can present SEO challenges, and implementing server-side rendering to mitigate these issues can add complexity to the project.

Node.js, despite its efficiency with I/O operations, might struggle with CPU-intensive tasks, potentially impacting performance in specific scenarios.

**References.**

* https://www.youtube.com/
* https://www.google.co.in/