

**Project Design Phase-II**  
**Technology Stack (Architecture & Stack)**

Date	26-05-2025
Team ID	LTVIP2025TMID58668
Project Name	HouseHunt: Finding Your Perfect Rental Home
Maximum Marks	4 Marks

**Technical Architecture:**

House Hunt: Finding Your Perfect Rental Home is designed with a scalable 3-tier architecture consisting of:

- **Presentation Layer (Frontend):**

User-friendly interface for renters, property owners, and administrators to search listings, post properties, and manage bookings. The web interface is responsive and intuitive across devices.

- **Business Logic Layer (Backend):**

Handles property search and filtering, booking confirmations, user registration, lease negotiation, and admin approvals. RESTful APIs manage communication between clients and server.

- **Data Storage Layer:**

Securely stores user profiles, property listings, booking history, inquiry messages, and lease agreement data using a NoSQL database.

The system supports real-time interactions and uses token-based authentication (JWT) for secure access control.

**Table-1 : Components & Technologies:**

S.No	Component	Description	Technology
1.	User Interface	Responsive web interface for renters, owners, and admin	React.js, HTML, CSS, Bootstrap, MUI
2.	Application Logic-1	Property booking, inquiry submission, search filters	Node.js, Express.js
3.	Application Logic-2	Admin panel, owner verification, lease and property management	React.js, Node.js
4.	Database	Stores property details, user info, booking history, messages	MongoDB

**Table-2: Application Characteristics:**

S.No	Characteristics	Description	Technology
5.	Open-Source Frameworks	Modular frontend/backend development	React.js, Node.js, Bootstrap, Tailwind CSS and ant design
6.	Scalable Architecture	REST API integration with dynamic component handling	MERN Stack with JWT

## References:

- [React.js Official Documentation](#)
- [Node.js Best Practices](#)
- [MongoDB Data Modeling](#)
- [JWT Authentication](#)
- [Cloudinary](#) & [EmailJS](#) for Image Storage and Mailing Service respectively