**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 26-05-2025 |
| Team ID | LTVIP2025TMID55949 |
| Project Name | House Hunt |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

**House Hunt** is designed using a scalable **3-tier architecture**, ensuring a robust, maintainable, and scalable system:

**Technology Stack**

The House Hunt app uses the MERN stack, supported by various libraries and tools to enhance performance

and user experience.

**Frontend Technologies:**

• React.js is used to build interactive UI components.

• Libraries like Bootstrap, Material UI, and Ant Design help with styling and layout.

• Axios is used to make API calls to the backend server.

**Backend Technologies:**

• Node.js serves as the runtime environment.

• Express.js manages routing and server logic.

• Multer is used to upload images (for property listings).

• JWT and bcryptjs are used for secure authentication and password encryption.

**Database:**

• MongoDB stores user profiles, property listings, and booking data.

• Mongoose is used as an ODM to simplify schema creation and data validation.

Other Tools and Libraries:

• Moment.js is used for date/time formatting in bookings.

• Nodemon helps in development by automatically restarting the server on changes.•

Git and GitHub are used for version control and team collaboration.

• Postman is used for API testing during development.

This technology stack provides a reliable, scalable, and secure foundation for the application.

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

| **S. No** | **Component** | **Description** | **Technology** |
| --- | --- | --- | --- |
|  | User Interface | Web/mobile-friendly interface for tenants & landlords | HTML, CSS, JavaScript / React Js etc. |
|  | Application Logic-1 | Booking, calendar, chat, and visit scheduling. | Node.js, Express.js |
|  | Application Logic-2 | |  | | --- | | Admin panel, user management, reporting | | React js, Node js |
|  | Database | User data, listings, visits, messages, payments | MongoDB |

**Table-2: Application Characteristics:**

| **S. No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
|  | Open-Source Frameworks | Frontend frameworks | React.js, Node.js, BootStrap, CSS |
|  | Scalable Architecture | 3-tier architecture with RESTful APIs | Microservices |

