HOUSE RENT APP USING MERN

Introduction :

- **Project Title: House Rent Application**

- **Team Members:**

    1. **VISWANATHAN S**
    2. **ARAVINDHAN A R**
    3. **PUSHPARAJ**
    4. **RISHI KESAVAN**
    5. **ANBUMANI C**

Project Overview :

- **Purpose:** To create a platform for listing and renting houses where users can search, book, and manage properties.
- **Features:**
    - User authentication (login/signup).
    - Property listing with search and filter options.
    - Booking and payment integration.
    - Admin dashboard for managing properties and bookings.

Architecture :

- **Frontend**: Built with React.js for an interactive and responsive user interface.
- **Backend**: Node.js with Express.js for managing APIs and server-side logic.

- **Database**: MongoDB for storing user data, property details, and bookings.

Setup Instructions :

- **Prerequisites:**
    1. Node.js
    2. MongoDB
    3. npm or yarn package manager
- **Installation**:

    Navigate to folders:

    - Frontend
    - Backend
- **Install dependencies:**

    Run `npm install` in both .
- **Set environment variables:**

    Backend: Configure `.env` for MongoDB URI, JWT secret, etc.

Folder Structure :

- **Client**: Details for React frontend, including components, routes, and services.
- **Server**: Explanation of Node.js backend with routes, controllers, and middleware.

Running the Application :

- **Frontend:** Run `npm start` in the directory.
- **Backend:** Run `npm start` or use `nodemon index.js` in the directory.

API Documentation :

- Document key endpoints like:
    - **POST /api/auth/login** - User login.
    - **GET /api/properties** - Fetch property listings.
    - **POST /api/booking** - Book a property.

Authentication :

- Use JWT for secure user authentication.
- Include login, signup, and role-based access (e.g., admin and user).

User Interface

- Homepage
- Property listing page
- Booking page
- Admin dashboard