

HOUSE RENT APP USING MERN

 *GitHub Repository:*

https://github.com/abhiramvasishta/ABHI_HOMES-HOUSEHUNT-.git

INTRODUCTION

AbhiHomes is a comprehensive real estate rental platform built using the MERN stack. It allows users to browse, filter, and submit property listings while integrating a seamless Google login experience. The app provides a smooth user experience with a dark-themed modern UI and animated transitions.

PROJECT OBJECTIVE & SCOPE

The purpose of AbhiHomes is to streamline the rental process by providing a user-friendly platform where renters can find homes and owners can submit listings. The project supports both renters and owners with tailored dashboards and filtered property search.

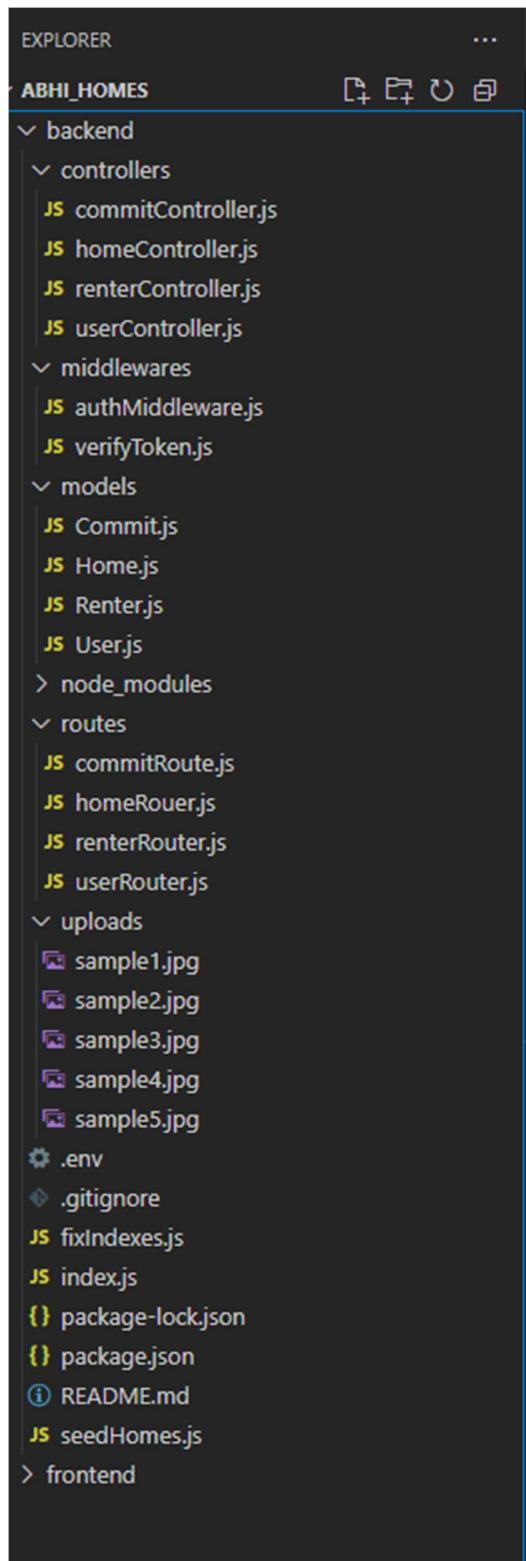
FEATURES IMPLEMENTED

- Google Login Integration via @react-oauth/google
- Dashboard with Welcome screen and Logout
- Property submission form with PlusCode and multiple file upload
- Search feature with filters (District, Pincode, State, Rent)
- Backend APIs with Express.js and MongoDB
- Tailwind-based modern UI with responsive layout and Framer Motion animations

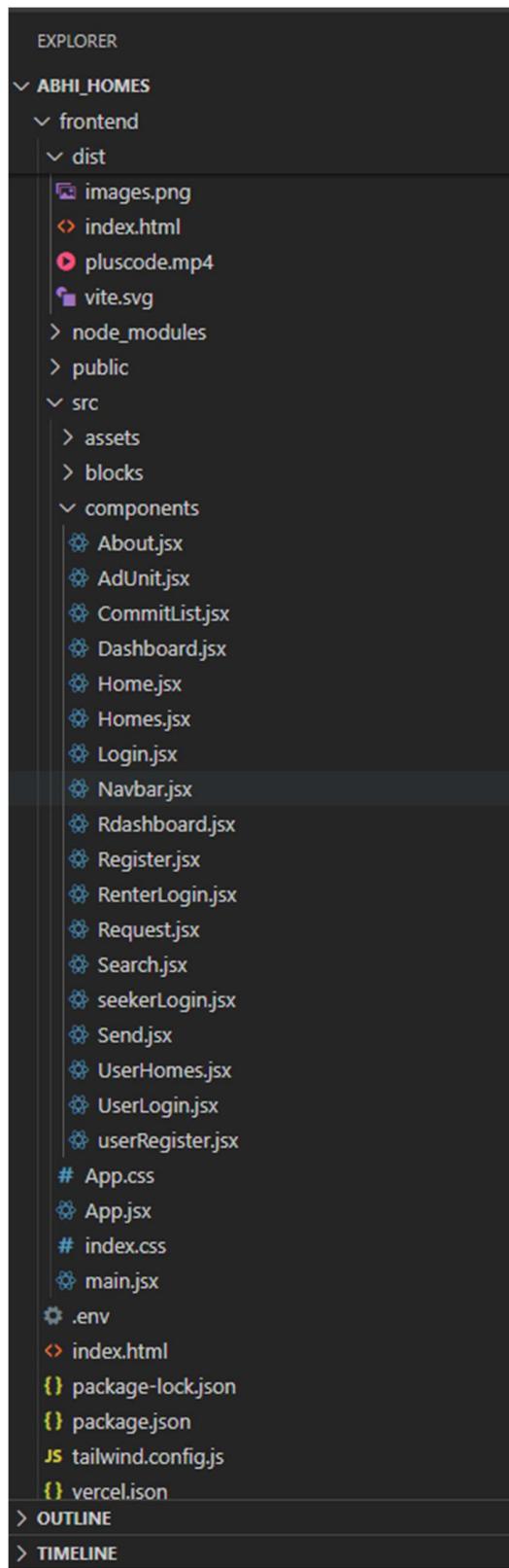
TECHNICAL ARCHITECTURE

- Frontend: Vite + React.js + Tailwind CSS
- Backend: Node.js + Express.js + MongoDB + Mongoose
- Auth: Google OAuth
- Image/Video Support: Multer for uploads, MP4 & JPEG display
- Deployment: Vercel (Frontend), Render (Backend)

FRONTEND STRUCTURE



BACKEND STRUCTURE



The frontend includes React components for pages like Home, Search, Dashboard, Renter Login, and more. It uses Tailwind CSS for styling, Framer Motion for animations, and Vite for fast build and development.

The backend is organized into controllers, routes, models, and middlewares. It manages user registration, login, property CRUD operations, image uploads, and authentication logic.

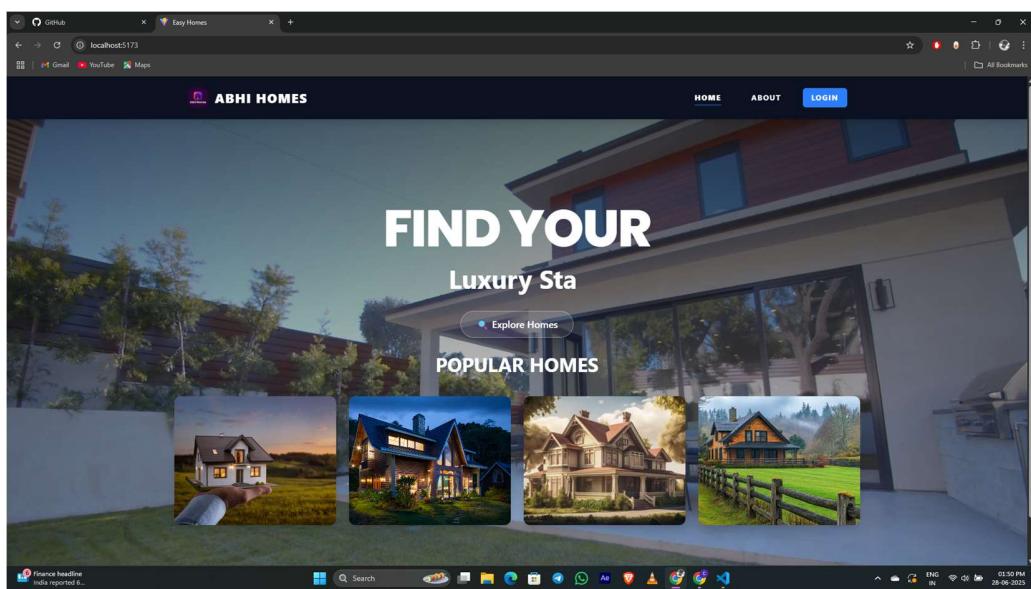
USER FLOW

1. User lands on homepage with animated video and navigation bar.
2. Clicks login to authenticate via Google OAuth.
3. Can explore available properties, filter using dropdowns or search by name.
4. Can submit their own home for rent using Renter form.
5. Accesses Dashboard to see their information and logout option.

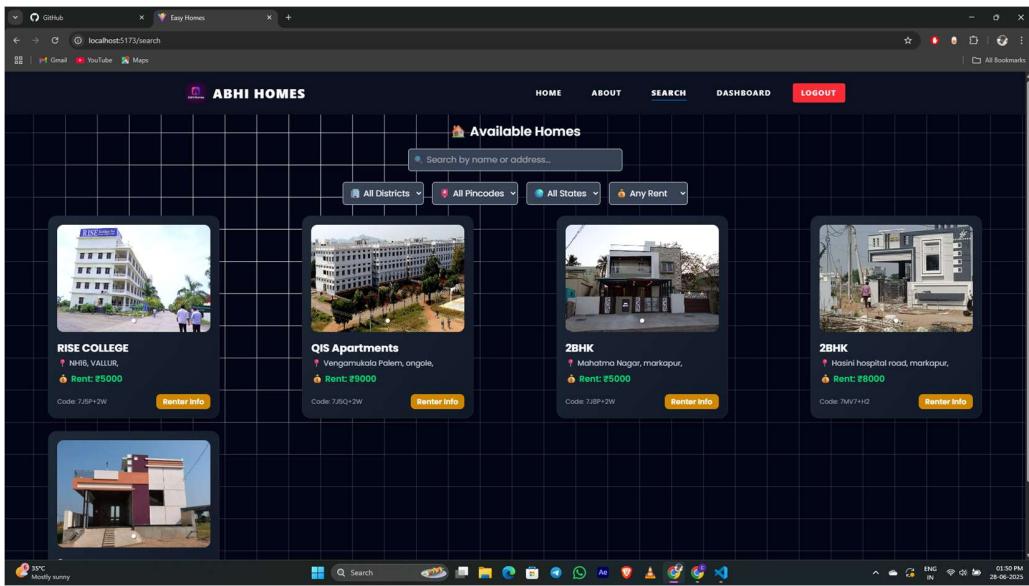
USER ROLES

- ◆ Renter: Can log in, view properties, and submit rental listings.
- ◆ Owner/Admin: Can approve or manage property listings, monitor renter submissions.

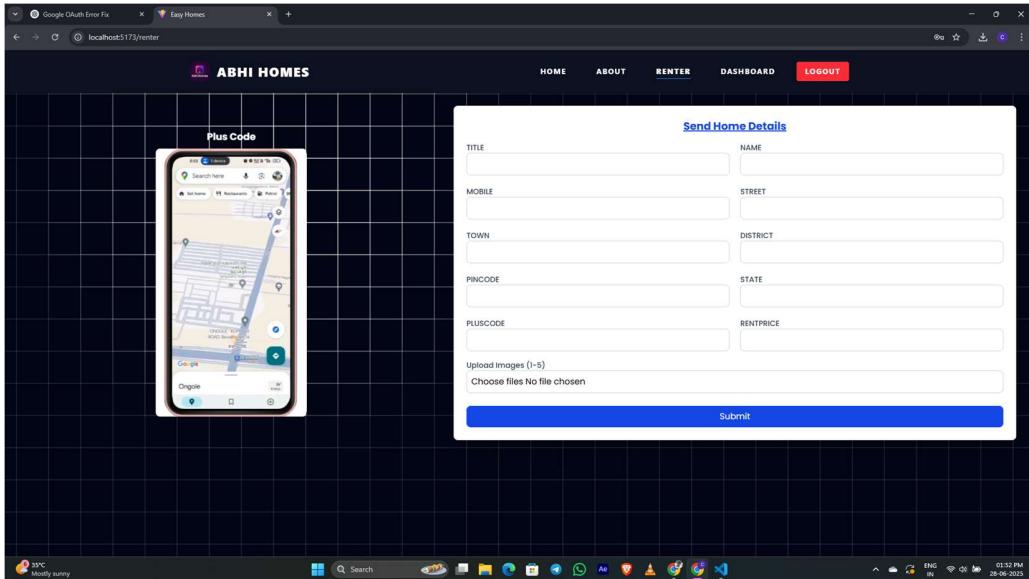
UI SCREENSHOTS



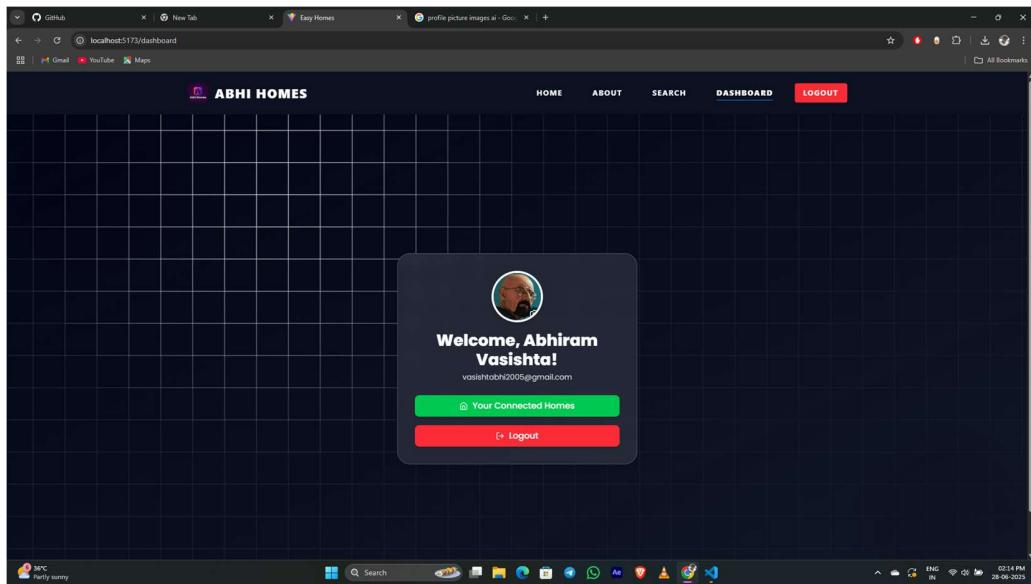
Homepage with Video Background



Search Filter Page



Property Submission Form



User Dashboard

Technical Architecture

This MERN stack project follows a client-server model. React handles the UI with API requests made using Axios. Express.js on the server manages routes and handles business logic. MongoDB is used for data persistence, including storing homes, users, and bookings. JWT is used for securing routes, and Multer handles file/image uploads.

Installation and Setup

1. Clone the repository and navigate into folders:

```
git clone https://github.com/your-username/abhihomes.git
```

2. For frontend:

```
cd frontend
```

```
npm install
```

```
npm run dev
```

3. For backend:

```
cd backend
```

```
npm install
```

```
node seedData.js
```

```
npm run dev
```

4. Environment variables:

- Frontend (.env):

VITE_API_URL=http://localhost:5000

VITE_GOOGLE_CLIENT_ID=<your-client-id>

- Backend (.env):

MONGO_URI=<your-mongodb-uri>

JWT_SECRET=<your-secret>

App Pages and Workflow

- Home: Video background with typed text and explore button
- Search: Filter-based home listings with modern cards
- Dashboard: Welcome page for Google logged-in users
- Renter Form: Plus code + home details submission form
- Admin Approval: Logic for owner role and listing visibility