1. **Introduction**

   - **Project Title: HOUSE RENT APPLICATION USING MERN**

   - **Team Members:**
     - Yaashwan S K
     - Libni Macklends K
     - Iliyas R
     - Rajesh V

2. **Project Overview**

   - **Purpose:** The app allows users to search for rental properties, book them, and manage listings, while property owners can list, manage, and rent their properties. The project implements modern web technologies to ensure responsiveness, security, and scalability.

   - **Features**:
     1. User Authentication and Authorization.
     2. Property Listings and Management.
     3. Booking and Availability Management.
     4. Responsive UI and Design.
     5. Real-Time Date Management.

3. **Architecture**

   - **Frontend:** The frontend is built with React, enhanced by libraries like Axios, Material UI, and Bootstrap to create an interactive and responsive user interface. such as property information, booking details, and user data, while JSON Web Tokens (JWT) handle secure access to restricted routes
   - **Backend:** The backend uses Node.js and Express to build RESTful API endpoints that handle business logic, data processing, and secure user authentication. JWT-based authentication ensures

only authorized users can access certain features, while bcrypt encrypts passwords for secure storage.

- **Database:** The database uses MongoDB, with Mongoose as an ORM for schema modelling and efficient interaction with data collections. The database stores all essential app information in structured collections, including users, property listings, and bookings, with MongoDB Atlas recommended for scalable, cloud-hosted data management.

4. Setup Instructions

- **Prerequisites:**

  1. Node.js (version 22.11.0)

  2. MongoDB (version 8.0.1)

- **Installation:**

  1. Navigate to the project directory.

  2. Install dependencies:

  - **Frontend: cd client && npm install**
  - **Backend: cd server && npm install**

  3. Set up environment variables:

  - **MongoDB URI**

5. Folder Structure

- **Client:** The client folder is dedicated to the frontend, built with React, and includes a public folder for static assets like images, and an src folder for the main application code**.**

- **Server:** The server folder hosts the Node.js and Express codebase. It includes a config folder for configuration files, such as the MongoDB connection setup. Inside server, an auth folder handles authentication and authorization middleware, and a models

folder holds Mongoose schemas for entities like users, properties, and booking.

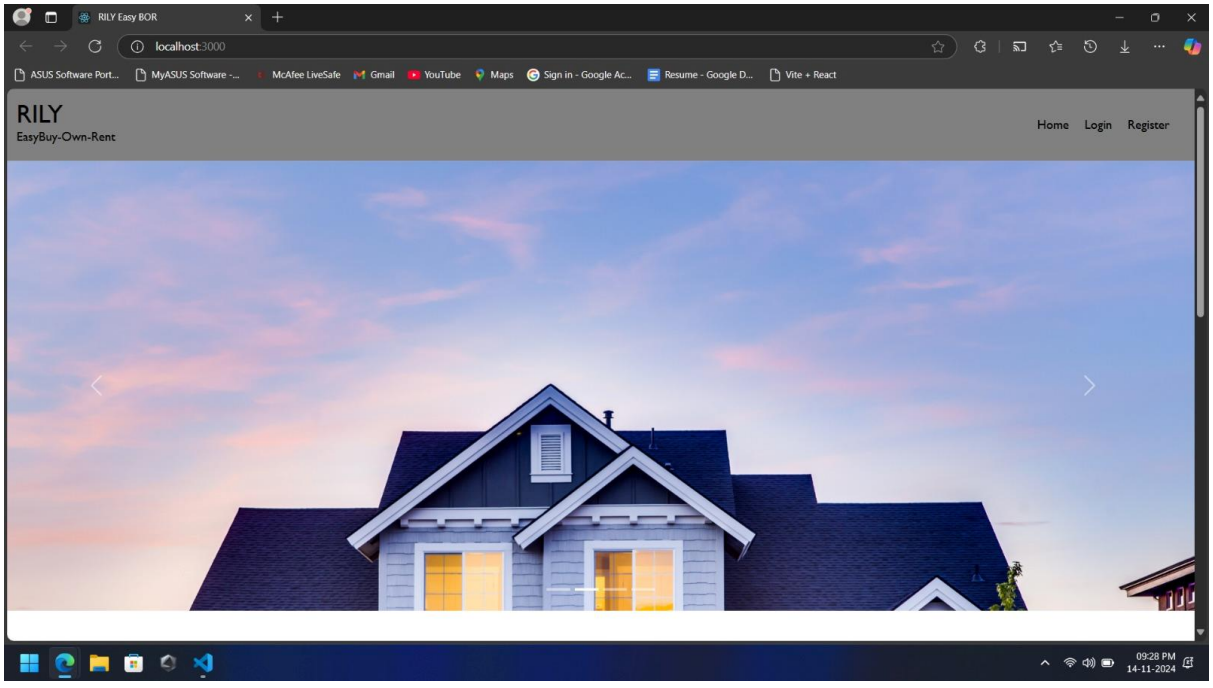
6. Running the Application:

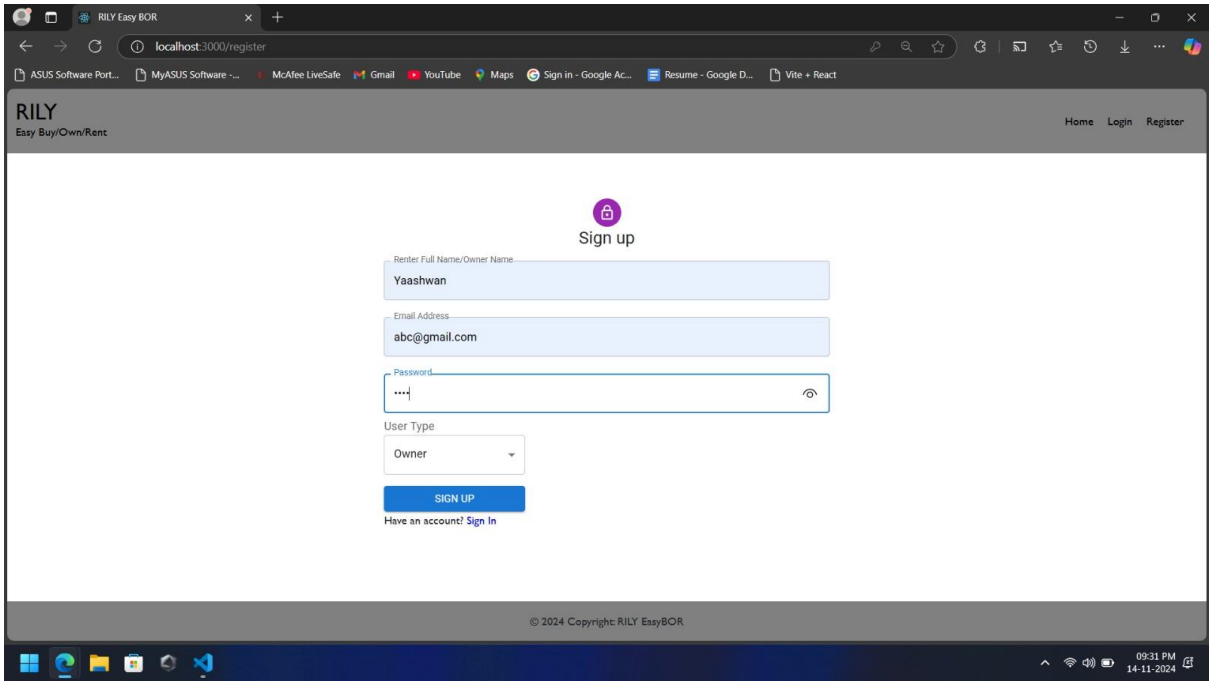
- To start the application locally:
    - Frontend: Run npm start in the client directory.
    - Backend: Run npm start in the server directory.

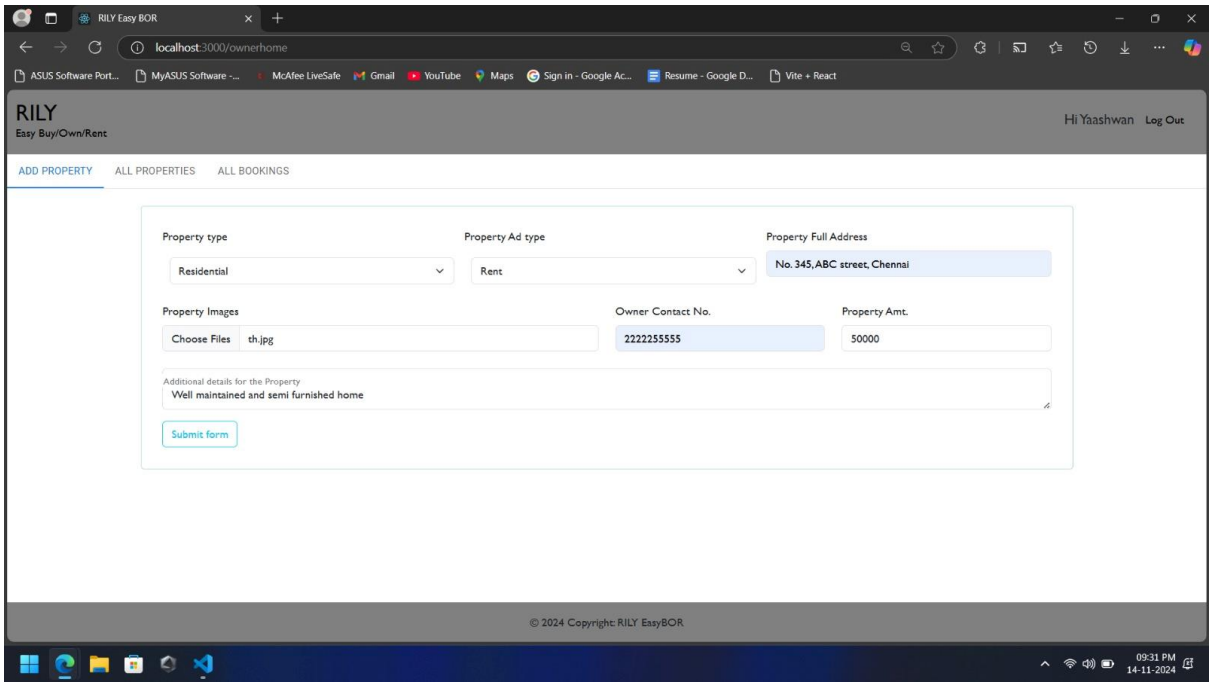
7. **Authentication**

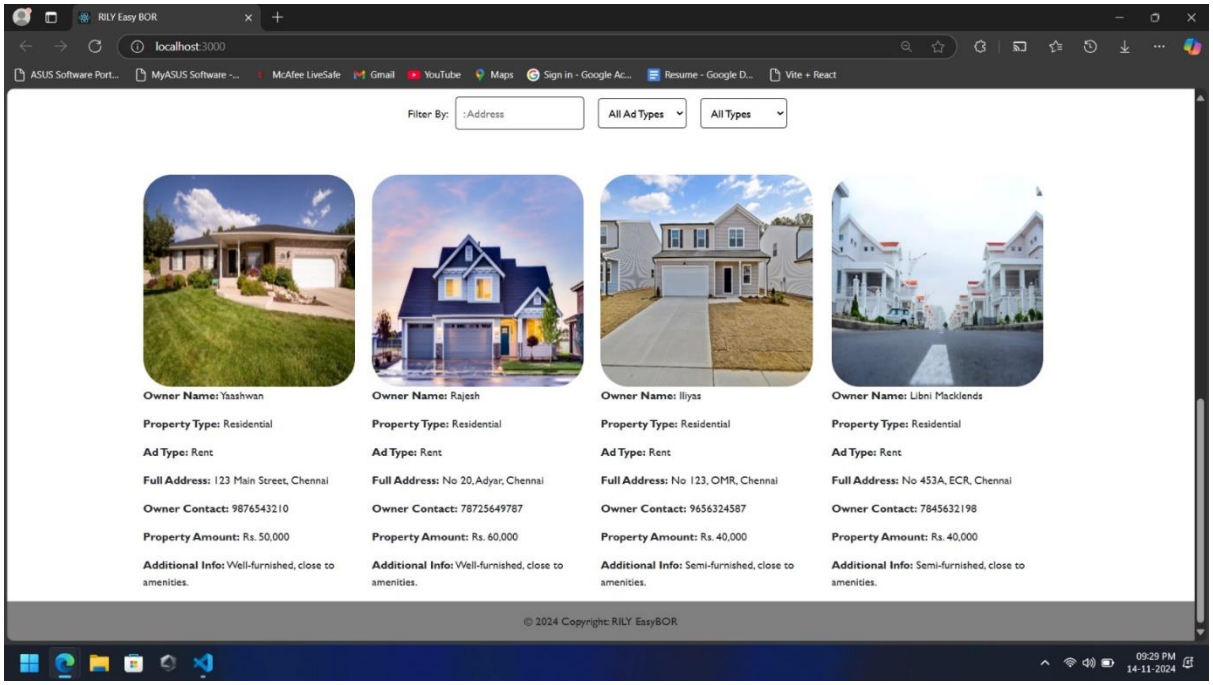
- Authentication is managed using JWT tokens. Upon login, a token is issued and stored on the client side to maintain session state. Middleware is implemented to verify tokens for protected routes, ensuring secure access to user-specific and sensitive data.

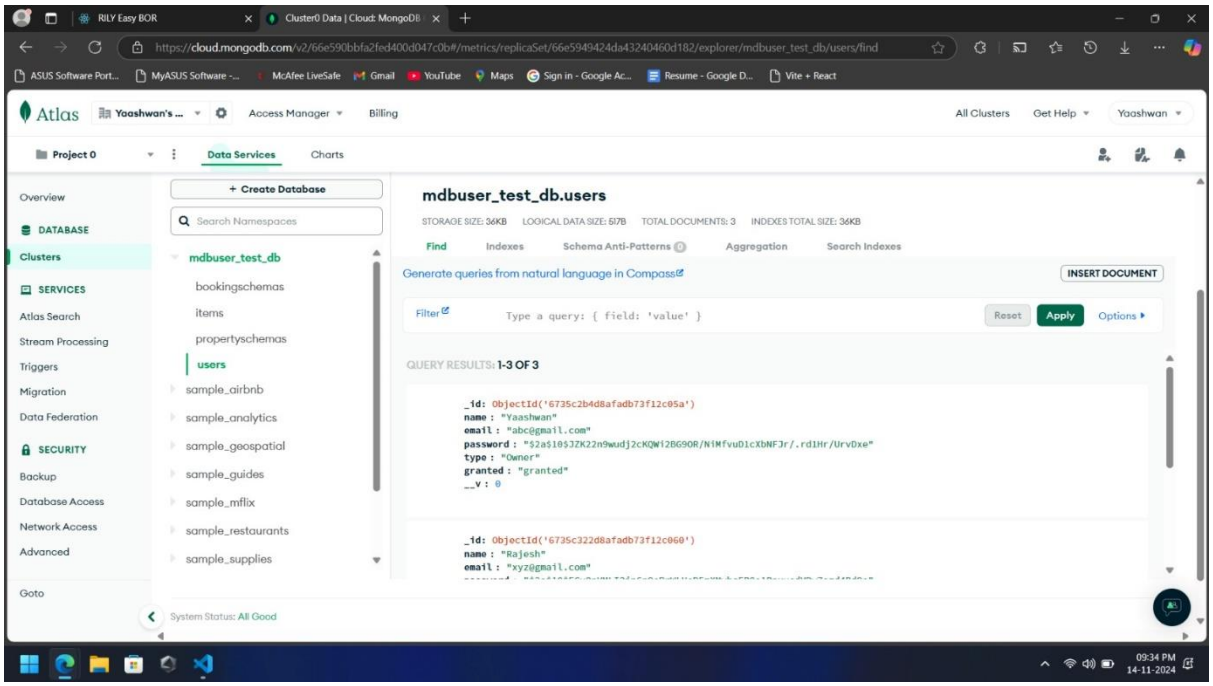
8. **Screenshots and Demo**




RILY Easy BOR
localhost:3000/register
RILY
Easy Buy/Own/Rent
Home Login Register
Sign up
Renter Full Name/Owner Name
Yaashwan
Email Address
abc@gmail.com
Password
User Type
Owner
SIGN UP
Have an account? Sign In
© 2024 Copyright: RILY EasyBOR


RILY Easy BOR
localhost:3000/ownerhome
RILY
Easy Buy/Own/Rent
Hi Yaashwan Log Out
ADD PROPERTY
ALL PROPERTIES
ALL BOOKINGS
Property type
Residential
Property Ad type
Rent
Property Full Address
No. 345, ABC street, Chennai
Property Images
Choose Files th.jpg
Owner Contact No.
2222255555
Property Amt.
50000
Additional details for the Property
Well maintained and semi furnished home
Submit form
© 2024 Copyright: RILY EasyBOR


RILY Easy BOR
localhost:3000
Filter By: :Address
All Ad Types
All Types
Owner Name: Yaashwan
Property Type: Residential
Ad Type: Rent
Full Address: 123 Main Street, Chennai
Owner Contact: 9876543210
Property Amount: Rs. 50,000
Additional Info: Well-furnished, close to amenities.
Owner Name: Rajesh
Property Type: Residential
Ad Type: Rent
Full Address: No 20, Adyar, Chennai
Owner Contact: 78725649787
Property Amount: Rs. 60,000
Additional Info: Well-furnished, close to amenities.
Owner Name: Iliyas
Property Type: Residential
Ad Type: Rent
Full Address: No 123, OMR, Chennai
Owner Contact: 9656324587
Property Amount: Rs. 40,000
Additional Info: Semi-furnished, close to amenities.
Owner Name: Libni Macklends
Property Type: Residential
Ad Type: Rent
Full Address: No 453A, ECR, Chennai
Owner Contact: 7845632198
Property Amount: Rs. 40,000
Additional Info: Semi-furnished, close to amenities.
© 2024 Copyright: RILY EasyBOR
09:29 PM
14-11-2024


RILY Easy BOR
Cluster0 Data | Cloud: MongoDB
https://cloud.mongodb.com/v2/66e590bbfa2fed400d047c0b#/metrics/replicaSet/66e5949424da43240460d182/explorer/mdbuser_test_db/users/find
Atlas
Yaashwan's ...
Access Manager
Billing
All Clusters
Get Help
Yaashwan
Project 0
Data Services
Charts
Overview
DATABASE
Clusters
SERVICES
Atlas Search
Stream Processing
Triggers
Migration
Data Federation
SECURITY
Backup
Database Access
Network Access
Advanced
Goto
+ Create Database
Search Namespaces
mdbuser_test_db
bookingschemas
items
propertyschemas
users
sample_airbnb
sample_analytics
sample_geospatial
sample_guides
sample_mflix
sample_restaurants
sample_supplies
mdbuser_test_db.users
STORAGE SIZE: 36KB
LOGICAL DATA SIZE: 517B
TOTAL DOCUMENTS: 3
INDEXES TOTAL SIZE: 36KB
Find
Indexes
Schema Anti-Patterns
Aggregation
Search Indexes
Generate queries from natural language in Compass
INSERT DOCUMENT
Filter
Type a query: { field: 'value' }
Reset
Apply
Options
QUERY RESULTS: 1-3 OF 3
_id: ObjectId('6735c2b4d8afadb73f12c05a')
name : "Yaashwan"
email : "abc@gmail.com"
password : "$2a$10$JZK22n9wudj2cKQWI2BG9OR/NiMfvuDlcXbNFJr/.rd1Hr/UrvDxe"
type : "Owner"
granted : "granted"
__v : 0
_id: ObjectId('6735c322d8afadb73f12c060')
name : "Rajesh"
email : "xyz@gmail.com"
System Status: All Good
09:34 PM
14-11-2024

10. **Known Issues**

- Managing frontend state across components can become complex, and concurrent bookings may cause double bookings. To secure the app, it's essential to protect against vulnerabilities like data exposure and injection attacks.

11. **Future Enhancements**

- **Real-Time Updates:** Implement WebSocket's for live updates on property listings and booking statuses.
- **Advanced Search Filters:** Add filters for amenities, price range, and location-based suggestions**.**
- **Payment Integration:** Integrate payment gateways like Stripe or PayPal for seamless bookings**.**
- **Admin Dashboard:** Develop an admin panel for managing users, properties, and bookings**.**
- **Push Notifications:** Enable push notifications for booking confirmations and new property listings**.**