

StayFinder - Full Stack Web App (Airbnb Clone)

Project Summary

StayFinder is a full-stack web application inspired by Airbnb. It allows users to browse property listings, view listing details, register/login, and book stays. Users can also view their recently visited listings and booking history.

Objectives

- Build a functional prototype with:
 - **Frontend:** Property listing, search, listing details, login/register pages
 - **Backend:** REST API for user auth, listings, bookings
 - **Database:** MongoDB to store users, listings, bookings
-

Deliverables

Frontend (React)

- **Homepage:** Grid of listing cards showing image, location, price
- **Listing Detail Page:** Title, description, image, location, calendar-style date pickers
- **Login/Register:** Form validation, error handling
- **Dashboard:**
 - Recently viewed listings (from localStorage)
 - Booking history per user

Backend (Node.js + Express)

- **Auth Routes:**
 - `POST /api/auth/register` — User registration
 - `POST /api/auth/login` — User login
- **Listings Routes:**
 - `GET /api/listings` — All listings
 - `GET /api/listings/:id` — Single listing by ID
- **Bookings Routes:**
 - `POST /api/bookings` — Book a property
 - `GET /api/bookings/user/:userId` — User's bookings
 - `DELETE /api/bookings/:bookingId` — Delete a booking

Database (MongoDB)

- **Models:**
 - `User`: name, email, password

- **Listing**: title, location, price, image, description
 - **Booking**: userId, listingId, checkInDate, checkOutDate
 - **Seed/Test Data**: Sample listings manually inserted into the database.
-

Bonus Features (Implemented)

- **Search Filters**: Users can filter by location and price range
- **Recently Viewed Listings**: Stored using `localStorage`

UI/UX Inspiration

- Clean, modern design similar to Airbnb
 - Layout and styling inspired by UI patterns from Dribbble and NomadX
-

Submission Information

Tech Stack Chosen & Why

- **Frontend**: React — for component-based architecture and fast rendering
- **Backend**: Node.js + Express — lightweight, fast, and easy to integrate with MongoDB
- **Database**: MongoDB Community Edition — schema-flexible and developer-friendly for prototyping

Full-Stack Capability

Yes, the applicant has built both frontend and backend end-to-end and is capable of full-stack development.

2 Unique Features to Improve Airbnb

1. **Listing View History** — Personalized dashboard showing recently viewed listings.
2. **Booking History with Cancel Option** — Users can view and delete past bookings.

App Security & Scaling Strategy

- Use JWT tokens for secure authentication
 - Store passwords securely using hashing (bcrypt)
 - CORS configuration to prevent cross-origin issues
 - Scale using MongoDB Atlas or Docker containers for production deployment
 - Future option: Load balancing + CDN for static assets
-

Final Project Status

All **required objectives and deliverables have been implemented** successfully. The app is fully functional with user login, listing browsing, bookings, and user dashboard features.

Next Steps (Optional)

- Add Stripe or Razorpay for real-time payments
 - Implement a host dashboard to manage properties
 - Add reviews/ratings system for listings
-

Project Structure

```
project/
├── stayfinder-client/    # React Frontend
├── stayfinder-server/    # Node.js Backend
│   ├── models/
│   ├── routes/
│   ├── app.js
│   └── .env
└── README.md
```

How to Run (Local Setup)

1. Clone the repo
2. Run backend:

```
cd stayfinder-server
npm install
node app.js
```

3. Run frontend:

```
cd stayfinder-client
npm install
npm start
```

4. Visit: `http://localhost:3000`
-