RealtyIllustrations - Documentation

1. Introduction

Thank you for purchasing RealtyIllustrations. This is a full-stack real estate management system built with Next.js 15, TypeScript, MongoDB, MUI, and Cloudinary.

Key Features:
- Appointment scheduling (House Tour & Call)
- Lead management system
- Mortgage calculator
- Blog & property listing management
- Secure JWT authentication (2-hour session)
- Interactive charts with ApexCharts
- Cloudinary for image uploads

2. Installation & Setup

System Requirements:
- Node.js v18+
- MongoDB Database (Local or Cloud)
- Cloudinary Account (for image storage)
- NPM or Yarn package manager

Installation Steps:

1. Extract the downloaded source code.
2. Open a terminal and navigate to the project folder.
3. Install dependencies:
   ```
   npm install
   ```
4. Create a `.env.local` file with the following environment variables:
   ```
   MONGODB_URI=your_mongodb_connection_string
   JWT_SECRET=your_secret_key
   CLOUDINARY_CLOUD_NAME=your_cloudinary_name
   CLOUDINARY_API_KEY=your_api_key
   CLOUDINARY_API_SECRET=your_api_secret
   ```
5. Start the development server:
   ```sh
   ```

```
npm run dev
```

The project will run at `http://localhost:3000`.

3. Features & How to Use

- Authentication: Users sign up via email, session expires after 2 hours.
- Appointments: Manage House Tour & Call bookings.
- Leads: Captures leads from bookings and inquiries.
- Listings: Add, edit, delete properties, upload images with Cloudinary.
- Mortgage Calculator: Estimate monthly mortgage payments.
- Blog Management: Create, edit, and delete blog posts.
- Dashboard Analytics: View lead data via ApexCharts.

4. Deployment (Vercel & DigitalOcean)

Deploying on Vercel:

1. Install Vercel CLI

```sh
npm install -g vercel
```

2. Login & Deploy

```sh
vercel login
vercel
```

Deploying on DigitalOcean:

- Use Docker or PM2 for backend hosting.
- Configure MongoDB Atlas for cloud database hosting.

5. Customization Guide

- Update MUI theme in `src/theme.ts` for colors & branding.
- Modify appointment system logic in `src/app/api/appointments`.
- Extend MongoDB schema in `models/Listing.ts` to add more listing fields.

6. Troubleshooting & FAQ

Q: Why is my Cloudinary image upload failing?

A: Ensure Cloudinary API credentials are correct in `.env.local`.

Q: How do I reset user sessions?
A: Modify `sessionTimeout` in `auth.ts` (default is 2 hours).

Q: How do I add more analytics charts?
A: Edit `src/components/Dashboard/Charts.tsx`.

7. Support

For support, please contact SUPPORT_EMAIL with your purchase code.