

# ■ PropBot

A modern real estate web application built with React, Tailwind CSS, and Firebase Authentication.

## ■ Features

- Authentication (Firebase Email/Password Auth)
- Protected Routes → Users must log in to access the homepage
- Modern UI with Tailwind CSS
- Responsive Design
- Powered by Vite for fast builds

## ■■ Tech Stack

Frontend: React 19, React Router DOM 7  
Styling: Tailwind CSS 3  
State Management: Redux Toolkit + React Redux  
Auth: Firebase Authentication  
Build Tool: Vite  
Icons: React Icons

## ■ Project Structure

```
proppbot/  
  ■■■ public/           # Static assets  
  ■ ■■■ _redirects      # SPA redirect  
  ■■■ src/  
    ■ ■■■ components/   # Reusable components (Hero, Footer, etc.)  
    ■ ■■■ assets/       # Images & icons  
    ■ ■■■ pages/        # Page components (Home, Login, Signup)  
    ■ ■■■ App.jsx       # App entry with routes  
    ■ ■■■ main.jsx      # React root  
  ■■■ package.json  
  ■■■ tailwind.config.js  
  ■■■ vite.config.js  
  ■■■ README.md
```

## ■ Setup & Installation

1. Clone the repo: `git clone https://github.com/your-username/proppbot.git && cd proppbot`
2. Install dependencies: `npm install`
3. Configure Firebase in `src/firebaseconfig.js` with your project credentials.

### ***Firebase Config Example***

```
import { initializeApp } from "firebase/app";  
import { getAuth } from "firebase/auth";  
  
const firebaseConfig = {  
  apiKey: "YOUR_API_KEY",  
  authDomain: "YOUR_AUTH_DOMAIN",  
  projectId: "YOUR_PROJECT_ID",  
  storageBucket: "YOUR_STORAGE_BUCKET",  
  messagingSenderId: "YOUR_MESSAGING_SENDER_ID",  
  appId: "YOUR_APP_ID"  
};  
  
const app = initializeApp(firebaseConfig);
```

```
export const auth = getAuth(app);
```

## ■■ Running Locally

```
npm run dev
```

The app will run at <http://localhost:5173/>

## ■ Build for Production

```
npm run build
```

This will create a `dist/` folder with production-ready assets.

## ■ Authentication Flow

New users → Sign Up page

Returning users → Login page

If not logged in → users are blocked from accessing `/home` and redirected to `/login` with an alert