## horizontal line



Project Proposal for Capstone 1

02.13.2025

**─**

Joshua Rogers

AteraWren

**Buying and selling home application**

### **Description**

This project is a web application that allows users to register and log in to create posts for renting or selling homes. Any user, including those without an account, can browse property listings. This app simplifies the process of listing and finding available housing options.

### **Stack**

**What tech stack will you use for your final project?** The project will use **Python (Flask) for the back end** and **React for the front end**. Flask will handle user authentication, post creation, and data management, while React will provide a dynamic and responsive user interface. PostgreSQL will serve as the database, and SQLAlchemy will be used for ORM.

### **Focus**

**Is the front-end UI or the back-end going to be the focus of your project? Or are you going to make an evenly focused full-stack application?** This will be a **full-stack application**, with equal focus on both front-end and back-end development. The front end will prioritize an intuitive UI/UX, while the back end will focus on secure authentication and efficient data handling.

### **Type**

**Will this be a website? A mobile app? Something else?** This will be a **web application** accessible via desktop and mobile browsers.

### 

### **Goal**

**What goal will your project be designed to achieve?** The goal of the project is to **simplify the process of listing and finding rental and sale properties**. Users can create and manage property listings, while visitors can browse available homes without needing an account.

### **Users**

**What kind of users will visit your app? In other words, what is the demographic of your users?**

* **Property owners and landlords** who want to rent or sell homes.
* **Prospective renters and buyers** looking for available properties.
* **General users** who want to browse property listings without an account.

### **Data**

**What data do you plan on using? How are you planning on collecting your data?**

* **User data** (for authentication and account management).
* **Property listings** (title, description, price, location, images, contact information).
* **Optional API integration** for pulling real estate market trends or location-based data.
* **Self-created API** to store and manage user posts.

### **Approach & Considerations**

#### **What does your database schema look like?**

* **Users Table**: id, username, email, hashed\_password, date\_joined.
* **Posts Table**: id, title, description, price, location, image\_url, user\_id (foreign key), created\_at.

#### 

#### 

#### **What kinds of issues might you run into with your API?**

* Ensuring **secure authentication** (password hashing, session management).
* Implementing **image uploads** for property listings.
* Managing **data validation and sanitization** for user-generated content.
* Handling potential **spam or fraudulent listings** (moderation tools may be needed).
* Optimizing **search and filtering functionalities** for ease of property browsing.

#### **Is there any sensitive information you need to secure?**

Yes, **user authentication data** (such as passwords) must be securely stored using hashing techniques (e.g., bcrypt). Additionally, any contact information in posts must be protected from scraping or abuse.

### 

### 

### **Functionality & Features**

#### **What functionality will your app include?**

* **User authentication (registration & login).**
* **Creating, editing, and deleting posts** (for logged-in users).
* **Browsing and searching listings** (for all users).
* **Image uploads** for property listings.
* **Filtering options** (e.g., by price, location, rental vs. sale).
* **Saved listings** (for logged-in users to favorite properties).
* **Contact system** (sellers and buyers can connect via a form or email).

#### **What will the user flow look like?**

1. **New users** can register and log in.
2. **Logged-in users** can create, edit, and delete their own property posts.
3. **All users (including guests)** can browse and search for property listings.

#### 

#### 

#### **What features make your site more than a CRUD app? What are your stretch goals?**

##### **Features Beyond CRUD:**

* **Image uploads** for property listings.
* **Search and filtering options** (e.g., by price, location, rental vs. sale).
* **Saved listings** (for logged-in users to favorite properties).
* **Contact system** (sellers and buyers can connect via a form or email).

##### **Stretch Goals:**

* **Interactive maps** to visualize property locations.
* **User messaging system** for communication between buyers and sellers.
* **Review system** where users can rate property listings.
* **Integration with third-party APIs** for pricing trends or mortgage calculations.

Once finalized, the project repository will be available on GitHub for review.