Find Your Home - Product Backlog - Team 12

Team Members: Jack Krueckeberg, McKenzie Patterson, James Goldberg, Jason Bodzy, Quinn Bello, and Nick Adair

Problem Statement:

The current issue with finding a city to live in is accounting for all the differences between each city. There are many sources of information; however, they are spread out across the web and not easily agglomerated by users.

Background Information:

Audience:

Finding the right city to live in is a difficult task. People have to scour the internet to find all of the information they need. Find Your Home is an application for anyone and everyone; those looking for a new place to live or those browsing new cities. We plan to create an application where users can input their preferences and the application will find cities that match their needs.

Similar Platforms:

There are a couple of existing services similar to ours like Areavibes, Dwellics, and moving.com. These systems allow the user to choose which cities they would like to compare and provide side-by-side comparisons but with different criteria. Moving.com provides comparisons about demographic, ethnic, residential, and income and jobs of the city, state, and specified zip code. Dwellics only compares two cities and has no options to save the selected cities. Areavibes provides comparisons on a letter grade.

Limitations:

Although these services exist, they all do not provide all information in the same location as well as only providing data over cities the user inputs. Users may not know all of the cities that match what they want. We believe that the user should be provided a list of cities that match their criteria and should have all of the information they need in one place. They should be able to save the cities that they liked as well as compare their favorite cities.

User Authentication System:

Functional:

1. **User Story:** As a user, I want to be able to register an account in order to save my preferences for the future.
    a. **Tasks:**
        i. Design a password restoration page
        ii. Implement a registration form that takes a username, email, password, etc
        iii. Implement a way to validate the username/password
        iv. Store user details in a database
2. **User Story:** As a user, I want to be able to sign in so that I can access my previously saved preferences.
    a. **Tasks:**
        i. Design the Login page
        ii. Implement a login method form to take email/username and password
        iii. Check for user authentication
        iv. Provide incorrect login feedback/limit the number of login attempts
3. **User Story:** As a user, I want to be able to ensure that I have signed out and know my account is secure.
    a. **Tasks:**
        i. Implement a signout button to return to the login page
        ii. Save user's preferences and data when signed out, clear locally

Non-Functional:

1. Users will only be able to access an account's data if they know the password. Recovering an account's data will be possible via email. There will be an option to change, transfer, or merge account information.
2. Loading an account's data will take less than 2 seconds provided a strong enough internet connection and normal instantaneous traffic. The upper limit of the number of people that can be connected at once will be ~100.

Browse City Preferences:

Functional:

1. **User Story:** As a user, I want to select city preferences so that I can find cities that match my desires.
    a. **Tasks:**
        i. Design a list view of the available preferences to choose from/filter through
        ii. Implement a system that weighs the importance of each specific preference
        iii. Implement a system that ranks the cities in the database by the selected user preferences

2. **User Story:** As a user, I want to view city data such as diversity, cost of living, job opportunities, etc., in order to make informed decisions.
    a. **Tasks:**
        i. Design a page that displays detailed information for each city
        ii. Collate and display the various data points about the city in an easy-to-read format
    b. **If time allows:**
        i. Multiple city viewer pages can be opened at once to allow easier comparison
3. **User Story:** As a user, I want to be able to add cities that I like to a favorites list to later revisit.
    a. **Tasks:**
        i. Implement an "Add to Favorites" button within the city information page
        ii. Store what cities the specific user favorited on their profile
        iii. Display the list of favorite cities under the user's profile
    b. **If Time allows:**
        i. Ability to share favorite city or city lists with other users.
        ii. Create and share posts about what you like/dislike about cities, a hypothetical dream city, ability to search through posts by keyword, username, or attributes.
4. **User Story:** As a user, I want to be able to view pictures and maps of the cities to help visualize the type of environment I would be living in.
    a. **Tasks:**
        i. Integrate an image gallery view within the city information pages
        ii. Integrate an API to display an interactive map of the city within the city information pages

Non-Functional:

1. The list of preferences will be easily navigable, a user will be able to find any given preference in less than 10 seconds. Searching criteria for preferences will have leniency to account for spelling errors.
2. City images will come from online sources to provide up-to-date results

View City Information:

Functional:

1. **User Story:** As a user, I want to group and compare specific attributes, such as average income vs. cost of living for a given city
    a. **Tasks:**
        i. Design a city viewer that can show and compare multiple attributes
        ii. Have an option for 2+ cites to be compared at once

iii. Integrate an API where this data is stored
iv. Compare attribute data visually

2. **User Story:** As a user, I want to see the potential growth of a given city
    a. **Tasks:**
        i. Design a city view that shows the population change in a city
        ii. Integrate an API where population data is stored by the city over time such as the American Census API
        iii. Compile these numbers into a graphical representation and display them n the site

Non-Functional:

1. No software limit to the number of city information pages that can be displayed at once.

Backend Development:

Functional:

1. **User Story:** As a developer, I need a backend server to store the city information, user profiles, and preferences.
    a. **Tasks:**
        i. Setup a server using either ASP.NET or NodeJS
        ii. Design and implement a relational database
        iii. Create APIs to fetch and store the required data

Non-Functional:

1. The code will be readable with comments and a standard style. This will also ensure maintainability.
2. Algorithms for sorting and searching will be efficient and quick.
3. The server will have the ability to be hosted on multiple machines if traffic becomes overwhelming.

User Interface:

Functional:

1. **User Story:** As a user, I want an easy-to-use and visually appealing user interface so that I can navigate that application easily and without confusion.
    a. **Tasks:**
        i. Design a consistent and intuitive user interface
        ii. Implement responsive designs that will adjust for use on different devices
        iii. Test UI with potential users to gather feedback on responsiveness and overall design

Non-Functional:

1. There will be smooth animations that don't affect performance, response times below 500ms, and logical organization of the interface to minimize the need for instructions.