

Aussie Dogs

Project Summary:

Aussie Dogs is a web application that allows people to determine how dog-friendly towns are. Most online providers of this information use high level, summary statistics that rank a city in abstract. In addition to taking advantage of these resources (especially official statistics), our application will offer a view closer to the neighborhood level while informally crowdsourcing ratings. Features like a dog-friendliness rating, the number and location of dog parks, or the number of dogs registered in the area will be available and may be derived from services that offer real time geolocation data.

Description:

Our objective with this project is to create a tool that helps users to obtain a deeper insight into the relationship between communities and their dogs, particularly when selecting a new area to move into. This could be a tool for users who might have specific allergies or aversions to specific breeds of dogs, or even dogs in general. Alternatively, this tool will be helpful to those looking to raise their own dogs in certain neighborhoods or areas, and who are looking for generalized information about what kinds of services or potential playmates are available for their dogs. This could include outside areas like dog parks or other areas conducive to regular dog walking as well as more specialized services like veterinarians or dog shops, which sell food and other accessories.

As such, our project will have several key functionalities. In order to make this application a living and consistently useful tool, it will be important that the database(s) associated with the project can be updated on a regular basis. Accordingly, we may incorporate data with some degree of asynchronicity from Real Time Location services like Google Maps. Furthermore, it is clearly useful to allow the user to tailor each search with a few specific criteria or provide basic ratings, which will be discussed later in the **Functionality** section of the proposal. Effective data visualization will be a primary driver of unique appeal. This represents a creative component and provides significantly for ease of use.

Importantly, our application will be based specifically around the South Australian and Victorian regions of Australia, since that is where the data in our primary database is focused around. However, the utility of our application can be expanded with the implementation of other databases or test markets in the future. Minimally, our application will serve as an effective proof-of-concept.

Usefulness:

When moving into a new neighborhood, it is often important to get a genuine understanding of the pet situation in the surrounding area. Whether one is allergic to pets, or someone that wants to know if their neighborhood is right for their littlest (human and non-human) family members, it can be incredibly difficult to find all of the information you need in a legible format, especially using one source. Depending on where one is moving, it may be

easy to find amenities immediately nearby (such as within the apartment complex), but not in surrounding areas. Our application strives to make the search for information a lot easier for users by providing most of the important information all within one application. Other online resources can tell you some general information about a city or town as a whole, without giving the specifics you might want for a particular neighborhood or region.

Alternatively, our application will be able to give more detailed information about dog resources in a certain area. The user will be able to learn about dog parks and other resources for a particular neighborhood, or even about the breeds and genders of dogs that are registered in that area. Ultimately, this is information that users might be able to find and piece together themselves if they were to sift through a variety of different sources. However, the true utility of our tool comes from the fact that all of this information can be found through minimal searches, without the user having to jump between multiple websites and web applications.

Realness:

The main functionalities of our application are based on very specific information on registered dogs in South Australia and Victoria, which can be found at this [link here](#). This database has individual entries for each registered dog, which can be referred to with a Reference ID, which is unique to each dog. The database also gives other characteristic information on each dog, such as the breed, gender, age, location (longitude and latitude coordinates, as well as suburb/neighborhood), and name. The information for South Australian dogs and Victorian dogs are separated into two files, which can be explored on the Kaggle website itself.

In order to implement some other functions of our tool, such as dog park and dog store information, it will be helpful to use public information found on Google Maps. By using this data in tandem with the previous database, we will be able to provide a much more comprehensive set of functions for our users.

Functionality:

The main functionality of our application will revolve around a search-engine type system. The user will be able to tailor their search by modifying the search parameters. These parameters would include data on the types of dogs to be included in the search (such as the dog breed, gender, or age). Furthermore, by utilizing the location data for each dog, we can create a very compelling visualization of the data that the user is looking for. In essence, the result of a search would be like a map with interactive elements, displaying the physical location of dog parks or veterinarians, and allowing the user to see the density of dogs in particular areas. To help us achieve this visualization, we plan to take inspiration from some of Professor Wade Fagen-Ulmschneider's [past projects](#). To help us set up Google Maps functionality, we also plan to take inspiration from Anum Fatima, found [here](#).

Users will also be able to insert new data into our database through the web application as well. They can add new entries into the database that includes all of the dog data found in our Kaggle databases already. A link on the website will take them to a form (likely a Google form) through which they would be able to specify the dog breed, gender, age,

location/neighborhood, and name. Our website will also allow users to informally give their own dog-friendliness rating. In addition to these crowdsourced ratings, we can also calculate the dog-friendliness score using the relative location of amenities (parks, pet stores, veterinarians, etc.).

Low-Fidelity UI Mockup:

[Link to slides](#)

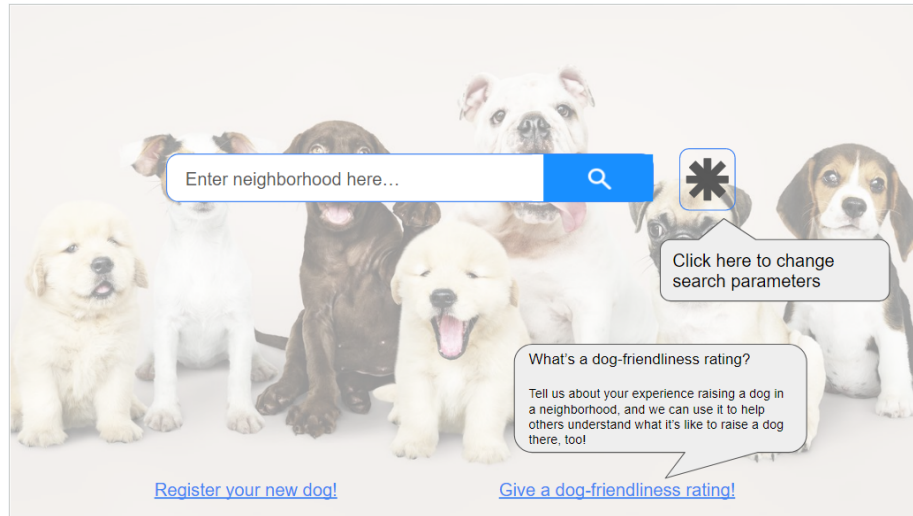


Figure 1: Mockup of our home page

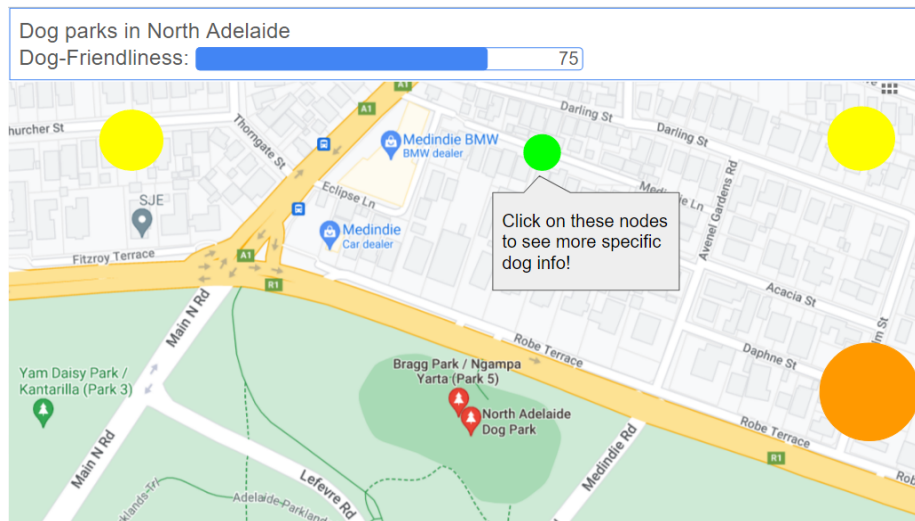


Figure 2: Sample visualization of search results

Project Work Distribution:

Patrick:

- Create basic search function
 - Includes the tools that allow users to change search parameters
- Return all dog information from search
- Return all maps information from search
 - Not including visualization, but all of the info used by visualization
- Integrating Google Maps into database
 - Will work with Ebaad to achieve this

Ebaad:

- Create forms for dog registration
 - Includes inserting new information into the database
- Integrating Google Maps info into existing database
 - Will work with Patrick to achieve this
- Overall UI manager
 - Each member will be responsible for creating the UI for their own functionalities, Ebaad will have to oversee how these UIs are combined together

Arman:

- Dog-friendliness calculation
- Create visualization for search results
 - Just the visualization, not including the info
 - Includes making interactive hot-spots that contain more specific dog info
- Handle any otherwise necessary data manipulation
 - Might include cleaning up garbage data

** Each member will be responsible for creating the necessary backend procedures to make their respective functionalities actually functional. Depending on the circumstances, we can change these responsibilities around, especially if one person thinks their functionalities are particularly harder to implement.