Project Milestone 4

Features we showed in the demo:

Home page: (**Worked in demo**)

- The home page shows the map (from the map api).
- Search bar to look for specific locations
- Button to get user get to profile page

User profile page: (**Worked in demo**)

- Show a profile picture
- Upload new pictures
- Type in user information

Map API: (**Worked in demo**)

- Leaflet API
- Markers created (latitude, longitude), containing fields for populating weather info
- Map itself- features such as zooming in/out, and bounds for zoom out.

Weather API: (**Talk about, but not shown in demo,** but it does work)

- The weather data comes from weather.gov. It takes a latitude and longitude and returns information for what the weather is like near that area.

Database (locations and user profile data): (**Worked in demo**)

- Locations:
    - The locations (latitude, longitude) of 455 cities in Colorado come from simplemaps.com
    - The locations (latitude, longitude) of 58 14ers in Colorado come from kaggle.
    - Extracting the locations from csv files by using pandas data frame in python.
- Profile:
    - Holds locations that the users saved on the profile page
    - Also holds username and password
    - All information is populated by the user profile page

Issues during development/demo:

There is no make file for us all to get copies for the database locally.

Took a while to pick our APIs, but now we have a good setup.

We have not connected the database to the frontend yet.

We initially wanted to host the project on git, but it only lets you host static pages and we need something with a database so we will need to pick something else.

We don’t have input checking yet so a user could put an invalid input in and it would fail.

Things to think about after the demo:

What if someone tries to find a location that isn’t in our database?

What are we doing for the middle layer?

Where will this be hosted? Must be on a cloud server

Suggestions by TA:

Documentation of ‘Create’ statements for database- enables for easier local testing

Hosting app on heroku instead of git- allows us to host database and handle server calls, while also keeping real time updates through git

Figure out a way to handle misspelled/mistyped input- handle these user errors through a function