**Title:** CU Spots

**Team:** Owen Taylor, Evan Gattis, and Saragam Gadal

**Description:**

The purpose of this application is to rate various places on campus so people know what spots are the best. This includes eating spots, bathrooms, places to rest, and study. This will give people information on the best places to do anything at CU Boulder. Users will be able to see places people have rated on a map and choose to visit them. Users can also rate and review new spots they have found or existing spots.

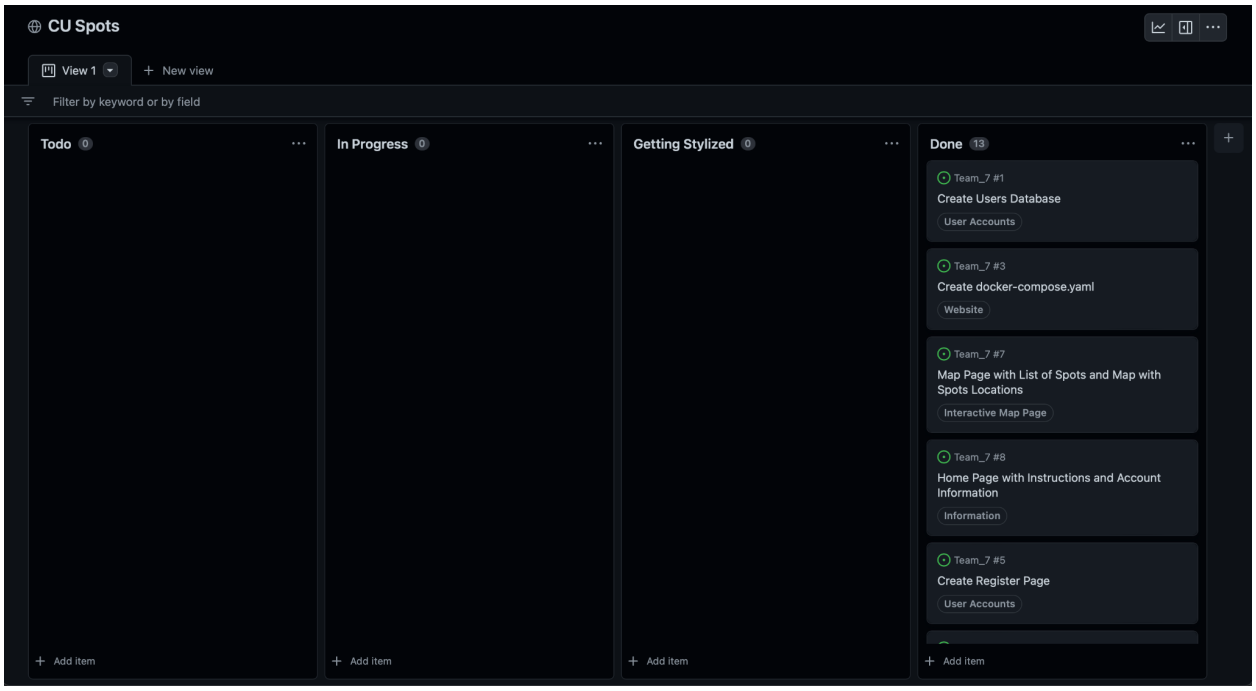
Using the Google Maps API, users can view locations of the spots on an interactive map. Users can also create a new spot and identify its location by clicking on a map. This process is incredibly intuitive since it uses a popular service.

Using the "rate spot" feature, users can select how much they liked the spot out of 10. The spot will display the average rating amongst all the users. This makes it easy to compare ratings to other spots and determine which one is the best.

To use the website, the user must create an account and login. The user is then greeted with a map and Spots with reviews added by other users. The user can navigate to the side bar and request to view a spot location, rate a spot they have been to, or click on the "add spot" button to add a spot that is not already there.

**Project Tracker:**

**https://github.com/users/owen-taylor/projects/2**




**Video:**

https://youtu.be/9gLeW1FeWpE

**Repo:**

https://github.com/owen-taylor/Team_7.git

**Contributions:**

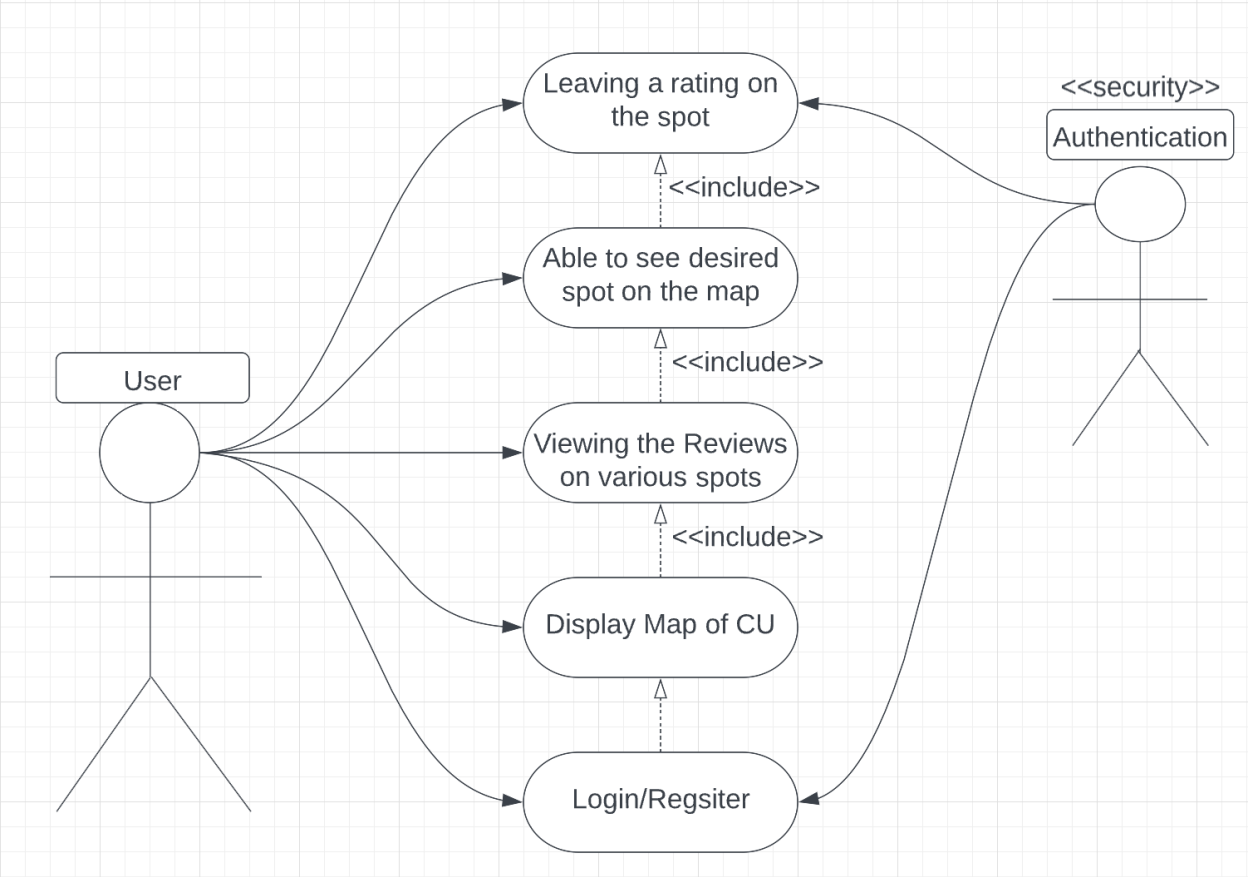
**Owen Taylor:** My main contributions to the project was creating the first iteration of the "view spots" feature, implementing the "add spots" feature, implementing the "review spots" feature, and setting up the spots database. The first feature I worked on was the "view spots" feature. The first iteration of this was using a static map call (not interactive). The feature in the final product uses most of the same principles as the prototype. The next feature I worked on was the "add spots" function. In the first iteration, you had to enter the coordinates of the spot manually (this was built upon later). The feature in the final product was built upon the prototype, but the back end is mostly the same. The last feature I worked on was the "review

spots" functionality. This feature worked very well from the initial prototype and is identical to the final version.

**Evan Gattis:** My contributions were setting up the base files and libraries, implementing login and register pages, working on the map, and adding cleaner UI. For the map, I took Owen's base code for the static map and used that in an interactive one. Then I learned how to adjust the map to fit to the screen, and I added a side navigation bar to make the UI better. I also fixed some bugs that came up along the way.

**Saragam Gadal:** I helped stylize the user interface for our login, registration, and homepage for a more appealing and user friendly look. I utilized css in order to add a background, edited the various color schemes of the pages, align the different components of our html code, and made the overall user interface easily accessible and pleasing to the consumer. I also created the case diagram to visualize how the user would be able to navigate and interact with the website.

**Case Diagram:**




**Test Results:**

**Feature 1 Login:** This feature was tested extensively with various usernames and passwords. The feature works flawlessly without any issues. If a user exists in the database and they provide the correct username and password combination, login is successful. If the user does not exist or does not provide the correct combination, login is not successful.

**Feature 2 Register:** This feature was tested extensively with various usernames and passwords. The feature works flawlessly without any issues. A user can create an account successfully with a username and password of their choosing if the username is not already taken. If the username is taken, the registration is not successful.

**Feature 3 View Location:** This feature works flawlessly. All the locations of the spots are stored and displayed accurately. When adding a new spot, the location is stored accurately.

**Feature 4 Adding Location:** This feature works flawlessly. When adding the location, the name is displayed correctly and the location is correct when viewing it.

**Feature 5 Rating Location:** This feature works flawlessly. When rating the location, the average rating is updated to reflect the new rating.

**Deployment:** The app was deployed with CU Boulder's computer science servers, but it did not work after we left the shell on our personal laptops. So here is how to use it locally:

Instructions to run locally:

- clone the repository to your local machine
- navigate to the repo in your shell
- execute "docker compose up"
- if all goes smoothly, enter: "http://localhost:3000/" into your browser