



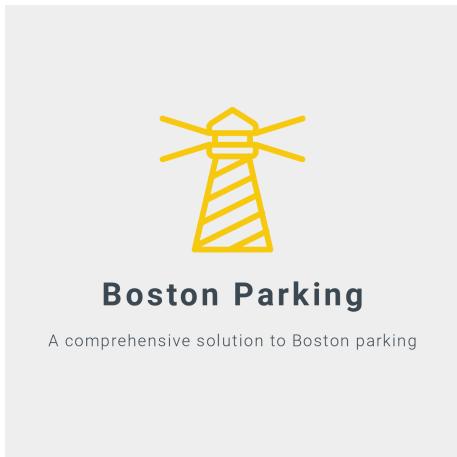
SangHyuk Kim

Final Project Report

Web-based Parking in Boston: A Comprehensive Solution

Boston, a city renowned for its rich culture, robust infrastructure, and top-tier educational institutions, presents a unique challenge when it comes to parking. Both residents and visitors often need help finding parking spots, face exorbitant fees, and must navigate a complex array of parking regulations. Despite the availability of interactive parking apps, city hall information, and on-site parking signs, these resources are often scattered and require users to conduct extensive 'research' to find suitable parking spots.

This project addresses these challenges by developing a comprehensive web-based parking application tailored for Boston as a testbed. We aim to consolidate all relevant parking information into a user-friendly platform. The application will provide real-time data on available parking spots, fee structures, and parking regulations, eliminating users needing to consult multiple sources. By providing a streamlined and efficient solution to parking in Boston, we aim to enhance the city experience for residents and visitors alike. Our project represents a significant step towards making Boston a truly smart city.



Description

Parking in Boston can be daunting. Whether you're a resident or a visitor, finding a suitable parking spot can feel like navigating a labyrinth. From metered parking to on-street parking with stickers, there are many options, but so are the regulations. One wrong move, and you could end up with a ticket or, worse, find your vehicle towed.

Existing solutions, such as innovative parking apps like Spot Hero, ParkWhiz, and Neighborhood.com, attempt to alleviate some of these challenges. However, these services are scattered, forcing users to sift through multiple platforms to find a parking spot manually. Furthermore, suppose a parking leaser only posts their spot on a specific application. In that case, users must sort things out by themselves again, a process far removed from the convenience of high-tech and accessible information collection.

Moreover, parking regulations, such as street sweeping dates or enforcement specifics, can vary by street and area, making them difficult to remember. This complexity is further compounded for those unfamiliar with the neighborhood, raising concerns about the safety and suitability of available parking spots.

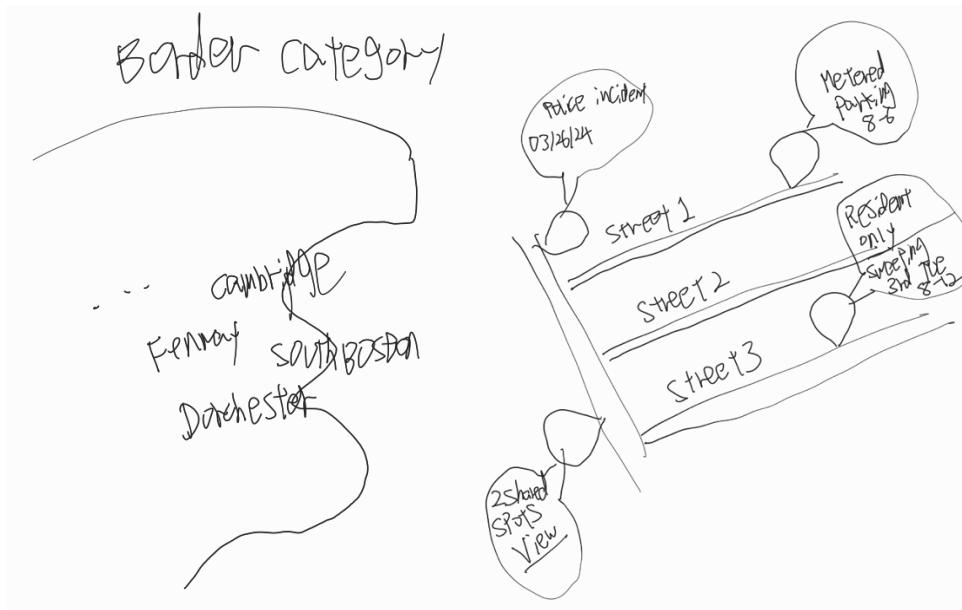
We propose a unique web-based parking application for Boston in response to these challenges. Our application will feature interactive maps and leverage big data to provide users with real-time shared parking spot information from diverse sources. By consolidating previously scattered information, we aim to create a one-stop solution for all parking needs in Boston, setting us apart from existing solutions.

Our application goes beyond providing parking enforcement details. It includes police incident information for users considering parking spots in unfamiliar neighborhoods. Our ultimate goal is to make parking in Boston a hassle-free experience, saving users valuable time and effort while ensuring their peace of mind. Stay tuned for more updates on this exciting project!

Datasets

Analyze Boston: Parking Meters, Snow Emergency Parking, Street Sweeping Schedules, Crime Incident Reports

Paper Sketches and Prototypes



Technologies, Development and Deployment

Web: JavaScript, React, React Map GL, Github host with npm deployment

Data preprocessing: Python, Pandas, JSON

Final Product

"Boston Parking" is a solution that collects parking data from multiple sources in one place, allowing users to check parking information and the history of the parking area on the responsive web at a glance. The parking meter data we used is in CSV format from Boston City Hall, and the data was preprocessed using Pandas. In the data preprocessing, we extract data to be displayed on the web, such as the GPS coordinates of the meter, the name of the street where the meter is located, the hourly parking rates, and the time when parking fees are not charged. The preprocessed data is then converted into JSON file format to display them on the responsive map of the React Map GL (Fig. 1).

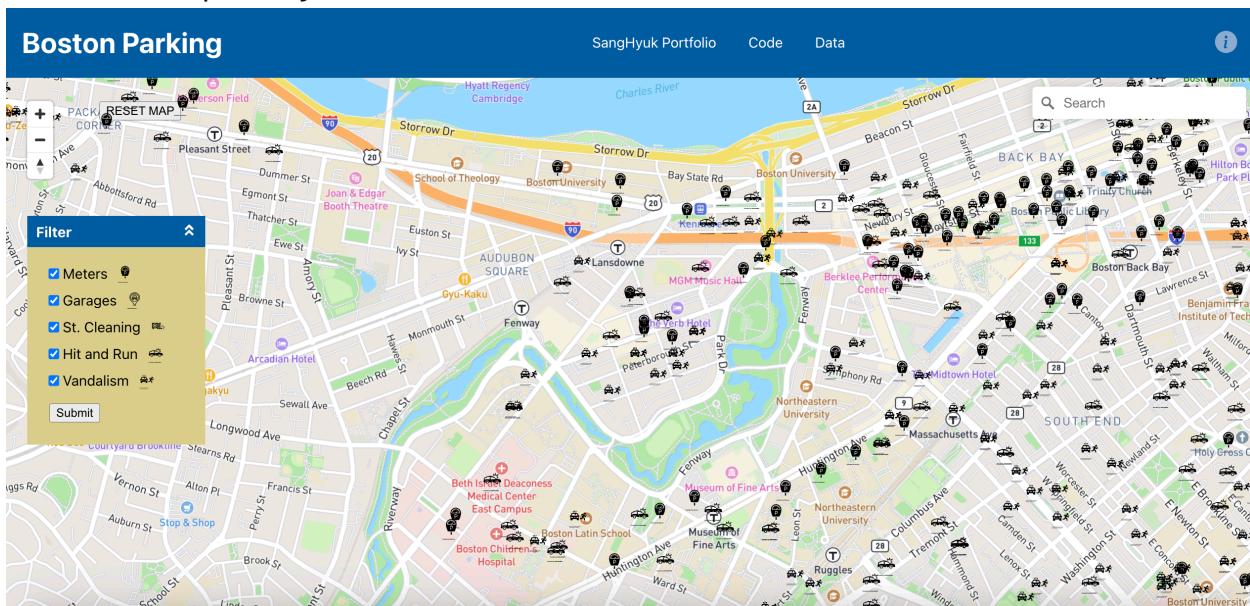


Fig 1. General overview of the Boston Parking

When a user clicks on the meter icon, a pop-up appears with detailed information about the meter we extracted in the preprocessing process (Fig. 2).

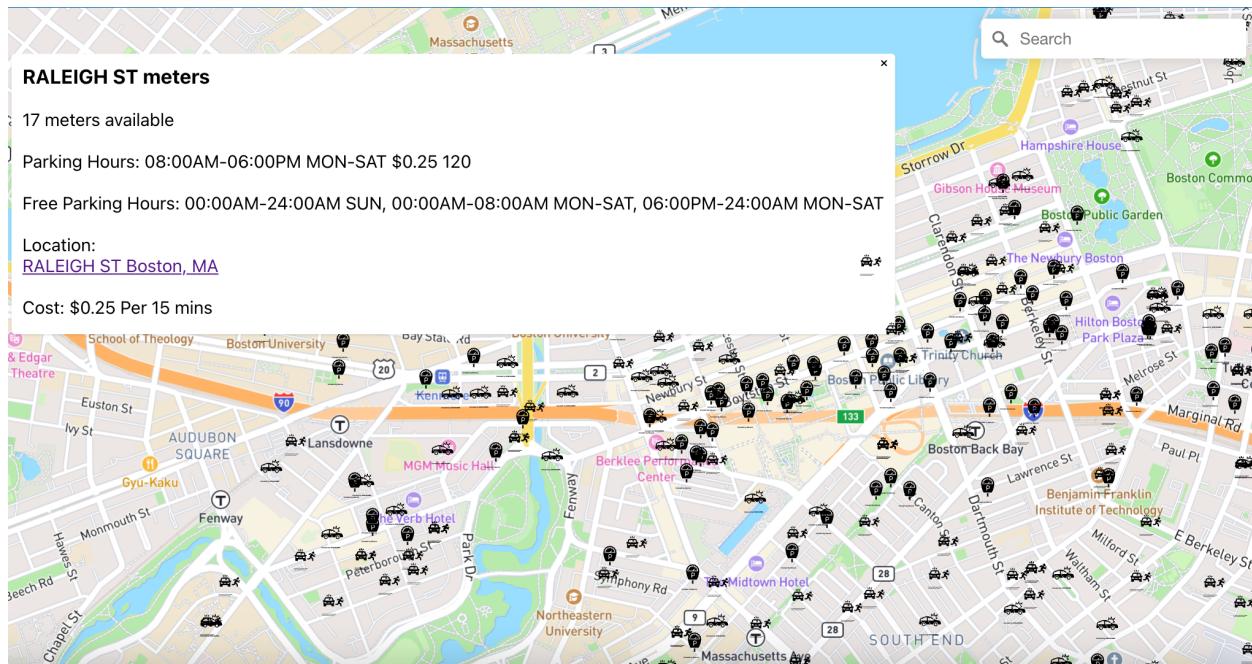


Fig. 2. Pop-up showing detailed information on parking spots

Aside from simple parking spot information, we wanted to give users insights about those parking spots. There may have been a case for you feeling unsure or unsafe when parking your vehicle in the area for the first time. We tackled this issue by leveraging hit-and-run and vandalism incidents in the Boston area and displayed them on the map (Fig. 3, 4, and 5).

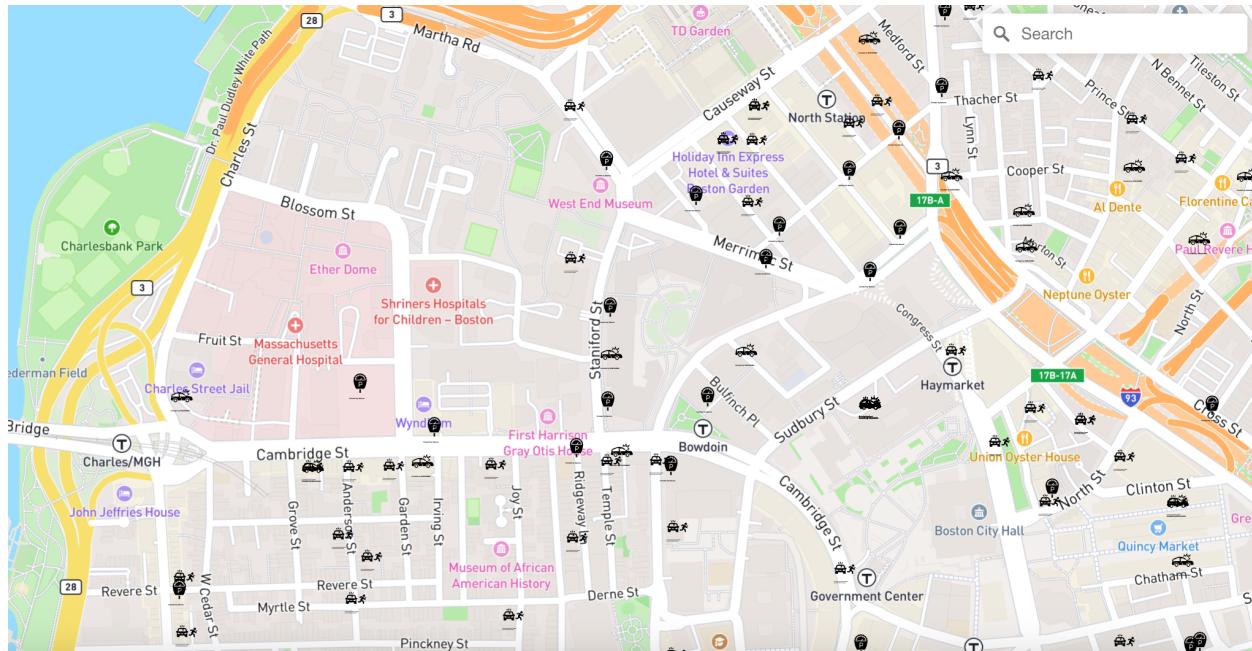


Fig. 3. An interactive map showing both parking meters and car-related crimes

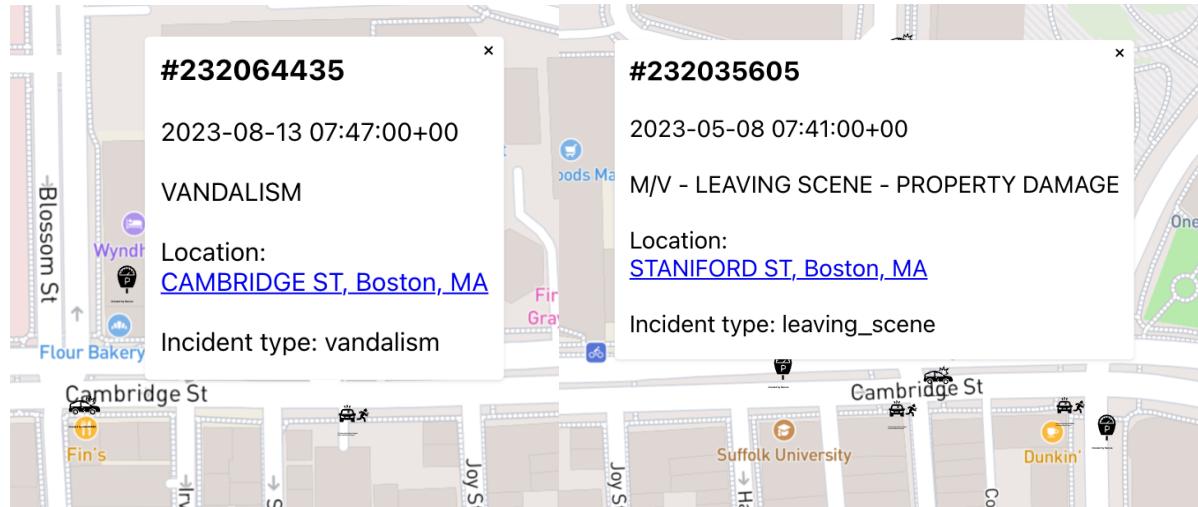


Fig. 4. Vandalism incident as a pop-up

Fig. 5. Hit-and-run incident as a pop-up

Interestingly, we found a pattern in which vandalism and hit-and-run incidents tend to occur near the parking area, as shown in Figures 6, 7, and 8. Therefore, we are confident that the parking and crime information we display is correct.



Fig. 6. Crime tends to occur near parking meters #1

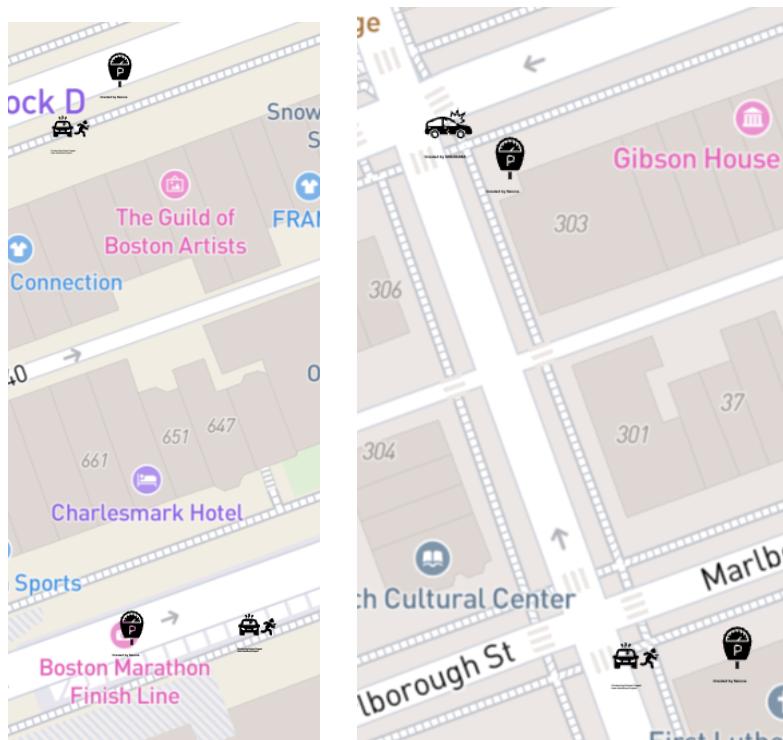


Fig. 7 and 8. Crime tends to occur near parking meters #2

Another interesting finding is that the frequency of crimes is not necessarily proportional depending on the number of meters. For example, we can see many more meters installed in the Back Bay area than in the South End area, but there are fewer vehicle-related crimes in the Back Bay (Fig. 9 and 10). Therefore, we can infer that we must pay more attention than the Back Bay when parking in the South End area.

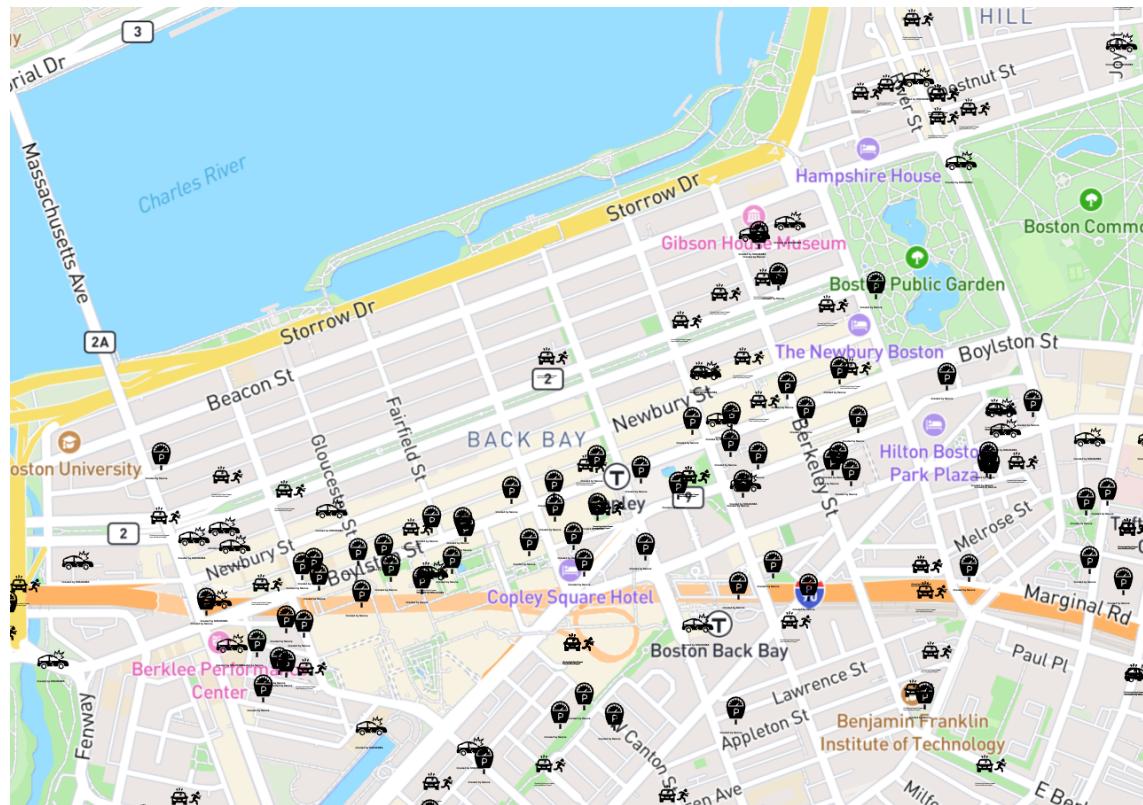


Fig. 9. Back Bay area parking information

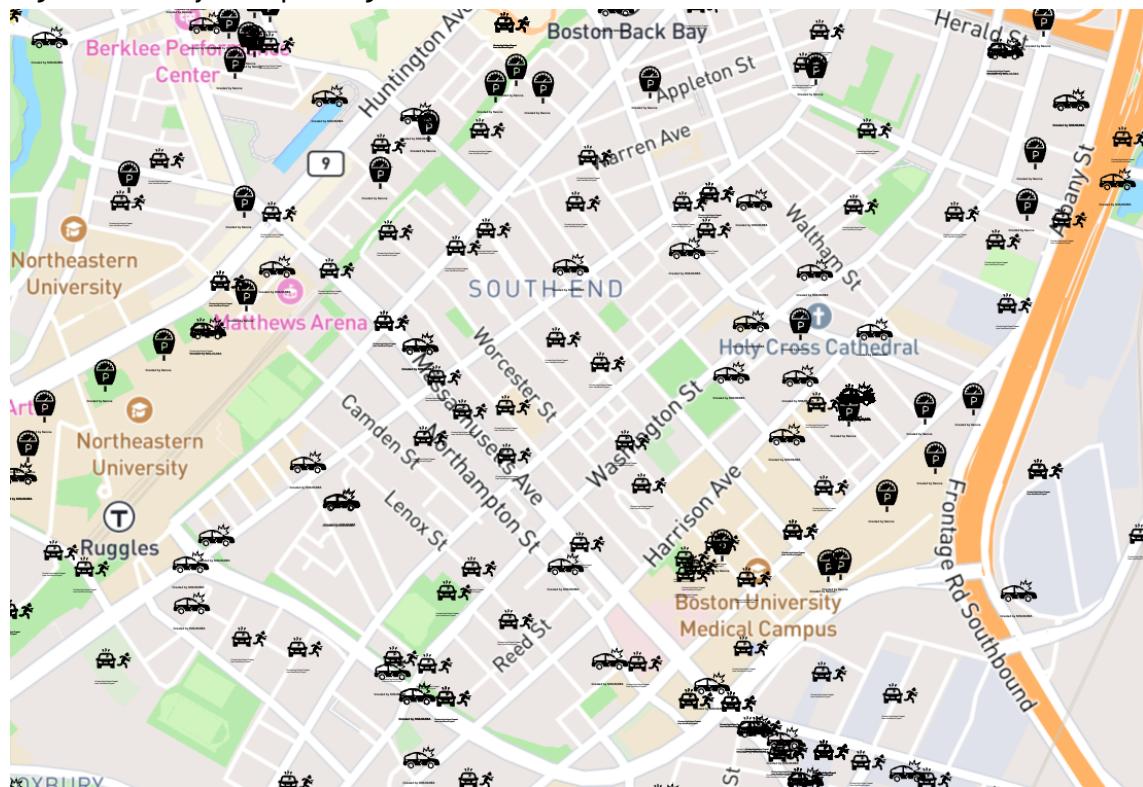


Fig. 10. South End area parking information

Finally, Boston Parking is designed to allow users to freely filter parking-related information displayed on the map, making it easy to extract and visually check the information necessary when referring to parking information. Figures 11, 12, and 13 below compare the filter menu and screenshots before and after using the filter.

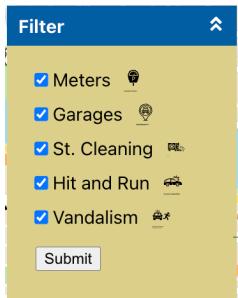


Fig. 11. Filter feature

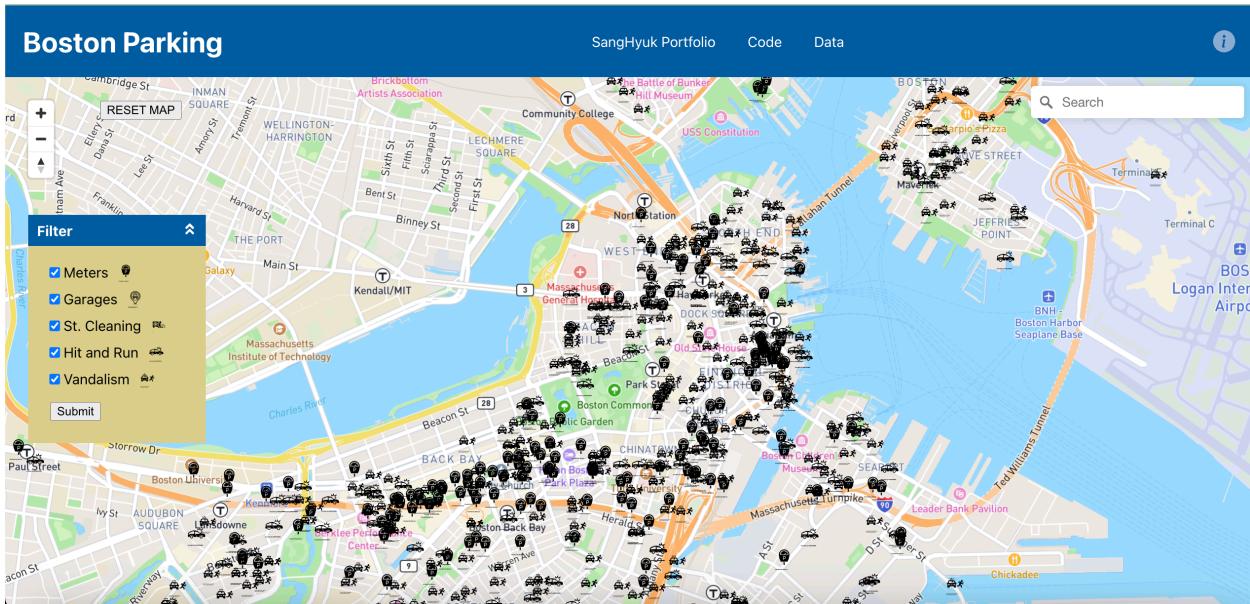


Fig. 12. Before applying the filter

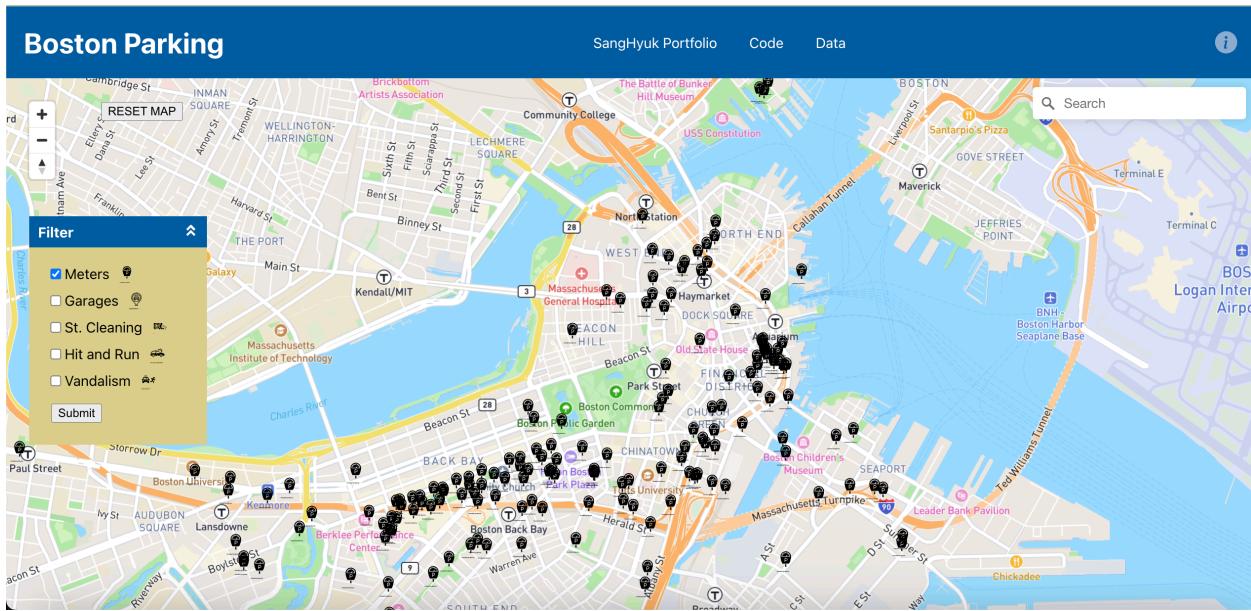


Fig. 13. After applying the filter

- Technical consideration -

In the early stages of Boston Parking development, we tried to display information on street cleaning and Snow Emergency Garages on the map. However, we confirmed a problem with the CSV data provided by Boston Open Data. Specifically, the approximate GPS coordinates in the Boston area are latitude: 42.343639 and longitude: -71.094861. However, the Snow Emergency Garage data contained GPS coordinates such as Latitude: 5213945.511 and Longitude: -7918414.198. These could not be displayed on the map due to a data assertion error in the Map GL framework. We believe that either the given data is wrong in the first place or there is another GPS coordinate system that can be converted into the metric we used; unfortunately, we could not use the Garage information for Boston Parking. In addition, the Street Sweeping schedule data did not contain GPS coordinates. We could have considered using the Google Map API to estimate the coordinates, but it could have taken much more time to implement this function; thus, Street Sweeping is not included in our final product. In addition, it was first planned to display information on various platforms in one place, such as by using the most famous SpotHero parking service API. However, the API key request webpage is not working from the SpotHero service, so we could not get access to SpotHero API. We leave this as a future task once the API key request comes back. Lastly, the program's runtime gets very slow if all cases of crime information alone are displayed using the React Map GL. We solved this issue by using partial information from crime data to

make reasonable runtime. However, there is room for further improvement in program performance through Lazy loading techniques in the future. Overall, I enjoyed this project.

Link: <https://shkimmie-umb.github.io/boston-parking/>

Code: <https://github.com/shkimmie-umb/boston-parking>

References

Inspired by <https://github.com/ArielleDOM/food-map>

React documents: <https://github.com/facebook/create-react-app>

<https://legacy.reactjs.org/docs/getting-started.html>