Max Maybury
Julian Contreras
Yonglin Wu
Shidan Xu

6.170 Software Studio
Fall 2014
Hiproute

Motivation

Hiproute is a web platform that aggregates and synthesizes searches for directions across multiple sources and methods of travel. Hiproute estimates the distance, time, and cost associated with each route, and gives a recommendation to the user.

While Google Maps, the unequivocal leader among directions providers, is a very useful tool, it lacks access to information from Hubway, the local bikeshare system. Furthermore, Google Maps does not give users access to cost information. These factors can have a big impact on a user's travel choices.

Purposes:
- **Eliminate the need of users to search on multiple platforms when comparing transportation to go between places in the city.** Many times when users compare ways to go from point A to point B in the city, they have to search on multiple platforms, e.g google maps, uber app, and hubway map. Our app aggregates the search results and display them together in a simple way, which simplifies the search and comparison process for the user.
- **Allow users to take both money and time into account when comparing transportation in the city.** People value their time and money in different ways. Users would likely be more cost-conscious during the peak travel hours of the day while wanting to arrive as quickly as possible during off-peak hours. Each search could potentially be different in terms of preferences.

Context diagram
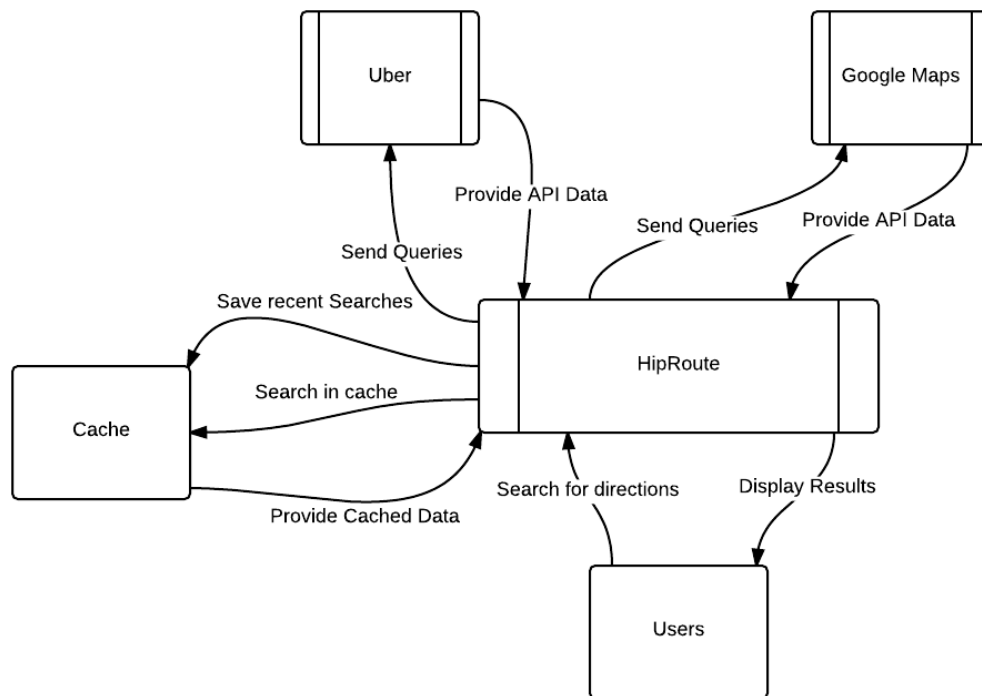
Concepts

Agony - the weighted evaluation of a particular transportation choice focused on cost/time. This satisfies the second purpose: users have various major concerns when evaluating a transportation choice.

Search - combines multiple searches of different transportation together, and returns a collection of routes that are keyed on time or cost. This satisfies the first purpose: eliminates the user's need to search on multiple platforms.

Route - A route is a path going from point A to B using some transportation method. For same point A and B there can be several "routes", each for each type of transportation.

Providers: Ways of getting from point A to point B. In Hiproute there are four providers, public transportation, Uber, hubway and walking.

Cache: save previous search results to deal with scenarios where APIs are broken. All cache are only saved in database for a short time (e.g 30 minutes), since old search results can be very different from search results at the moment (e.g uber surcharge). Also it is assumed that google API will always be up so google search results are not stored in cache.

Data model
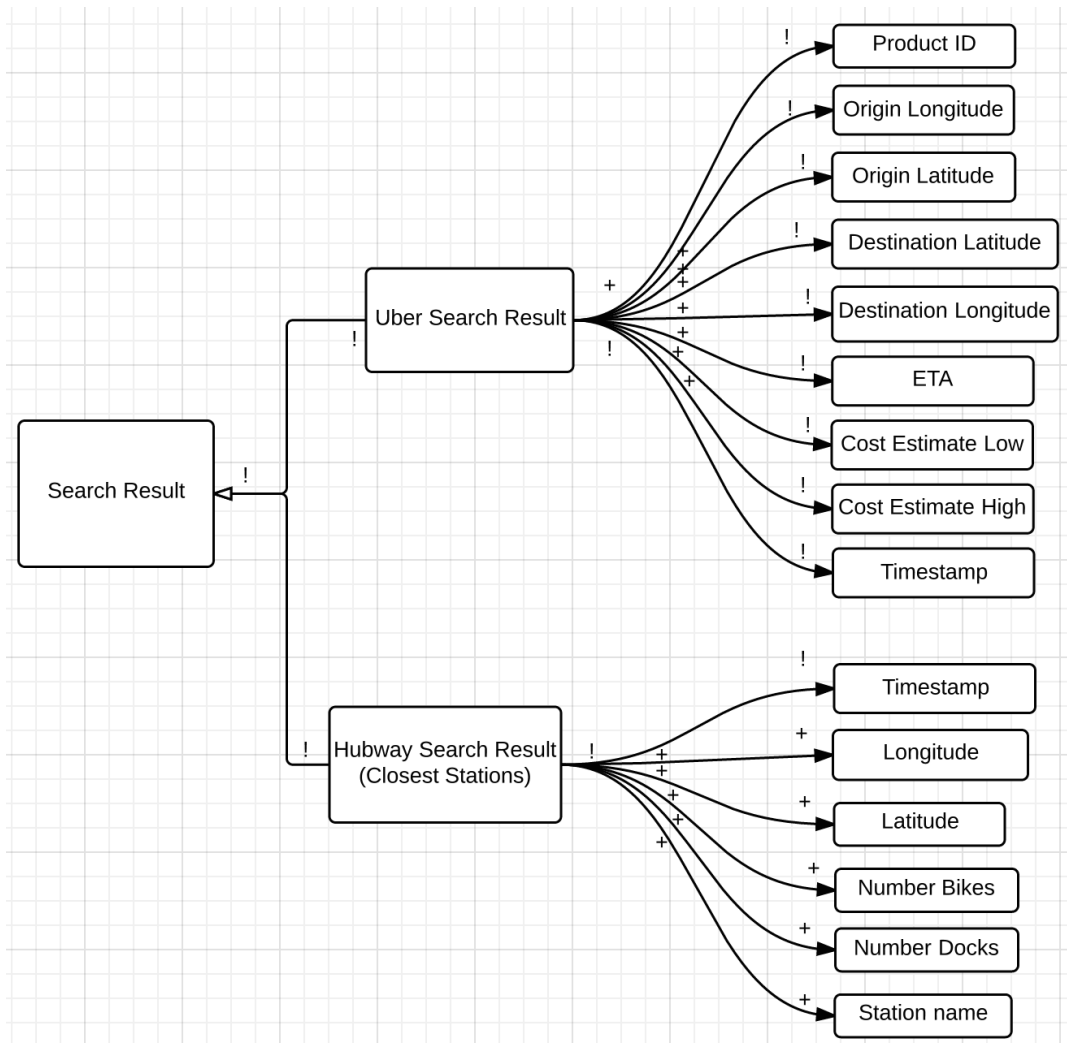
Security concerns

In general , for Hiproute, security is not a major concern, because all the search results we display to the user are publicly available.

Summary of key security requirements
● Past search history is stored internally in server and no users have access to it.

Threat Model
● We assume that google and uber data are reliable over an https connection
● We assume that any vector of attack will originate from a malicious api call or malicious input into the search area.

- Can assume no interest from state actors or criminal syndicates, since no profile information stored (name, email, credit card).
- Since this web service is completely free and there is no money transaction between any parties, there is very little risk of fraud.

Potential risks and mitigations:
- Hackers break into the system and get access to all past search history on Hiproute. Our resolution is that Hiproute does not expose public API to access past search history, and database is hosted on the server with basic security, which is fairly hard to get through. Also Hiproute deletes all serach history that is older than 30 minutes, which reduces the amount of data stored in database. Even in the event where hackers have access to search history database, it is not a huge threat to our users as Hiproute does not store where the searches are coming from, so it is very difficult to infer the identity of the user who made the search. Also the information the users provide: address of places they are going to, are not usually sensitive information. On top of that, Hiproute only stores search history from the last 30 minutes.
- Another potential hack is the man-in-the-middle attack, where a hacker tries to redirect the user to a place that is different from where the user searched for. Our resolution is to use https to prevent wrong directions returning to users.
- Code injection into search query area. Our resolution is to check the sanity of searches before processing them.
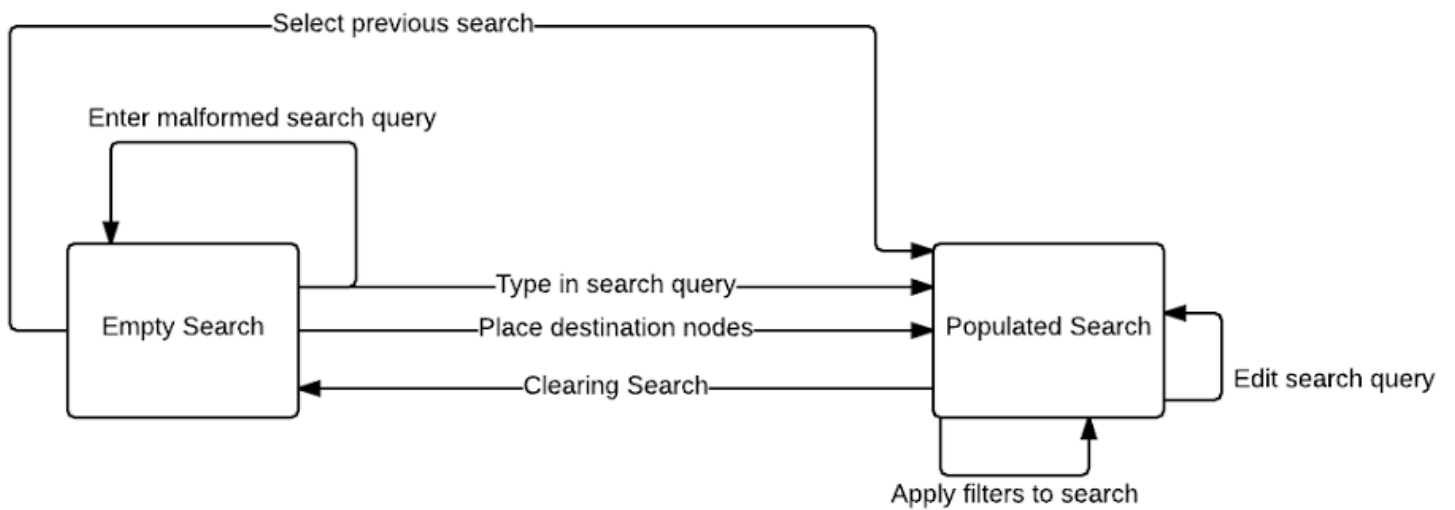
Wireframes
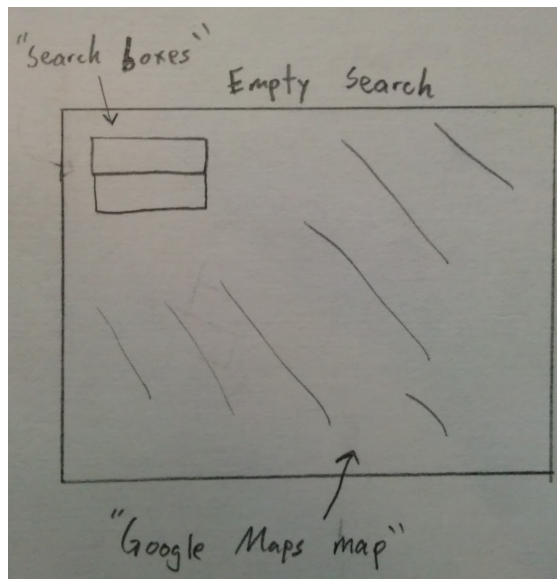
Empty Search

Populated Search
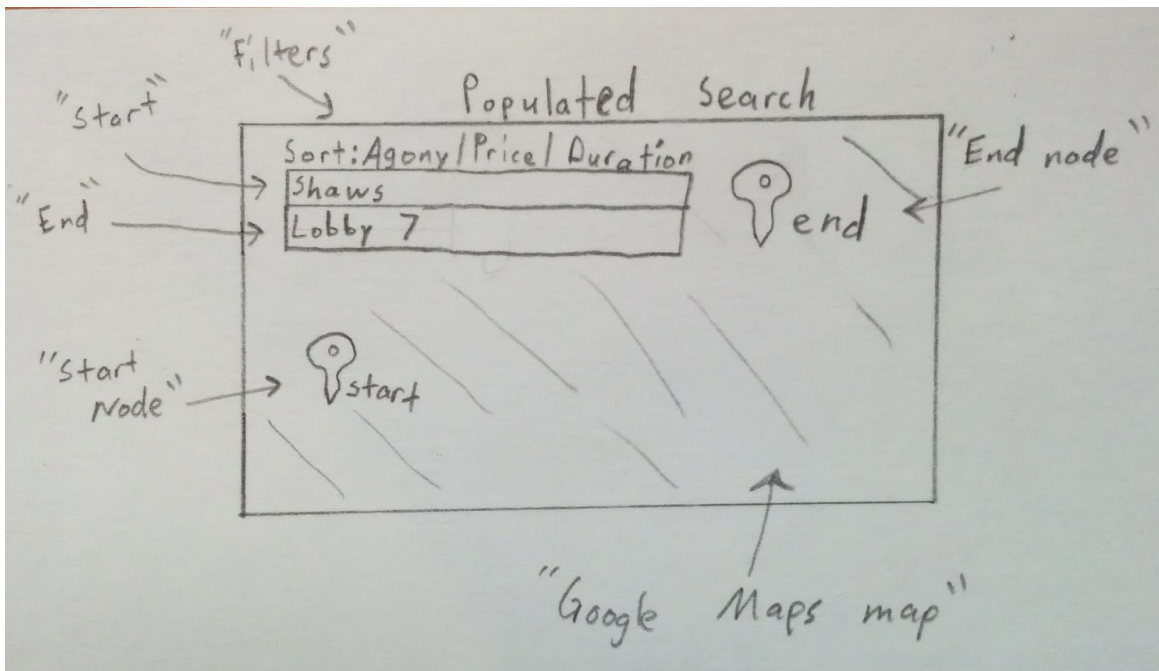
Flow chart

Design challenges

Issue: How much past search history data does Hiproute cache? Hiproute caches all users' search results in database for a short amount of time. If one of the APIs are down and a user makes a search on Hiproute, Hiproute can try to find a similar search in database and returns the cached results. The questions is how much search history data Hiproute saves.

Potential Solutions
- Save all search results, including google API, uber API and hubway API. In this cases, Hiproute still works even when all three API providers are down.
- Only save uber and hubway API data. This reduces the amount of data to store, but it makes Hiproute dependant on Google.

Resolution:
For the cache, we decided on just caching the Uber and Hubway api data instead of the google api data. This was done because Google's api is more complex and has a very high uptime. On the other hand, Uber's api is simpler and changes less often, a better candidate for caching. Another reason for not caching Google api data is that a Google api outage ultimately means our whole site is unusable anyway.

Issue: How does Hiproute let user choose their preference of time and money? Hiproute ranks search results based on combined "agony" of time and money. In user interface, how do users indicate their preferences?

Potential Solutions
- Have two tabs, one for "prefer less money" and one for "prefer less time", and users can choose either one.
- Have a slider from "time" and "money so that user can choose anything in between.

Resolution:
A person may have a preference over time vs. money, but its not one extreme or another. So we plan to include a slider (in the final product) for the user to change the weight they give to each factor. When we calculate our agony, we will adjust the numbers accordingly to the desired weights.

Issue: How should the UI look like?

Potential Solutions
- resemble Google maps, with map overlay on entire page.
- Resemble Hipmunk, only shows the routes info but not the map
- have a 2-column screen, left side for search and right side for map.

Resolution:
We chose to use the Google maps approach, overlaying the search results on top of the map. We first rejected the Hipmunk approach because we are showing the users routes, unlike air travel, the user wants to see how he reaches there. We then chose the Google maps approach because the map is a crucial map to our UI, having a bigger map can let the user understand where he is going to a higher accuracy.