Max Maybury
Julian Contreras
Yonglin Wu
Shidan Xu

Teamwork Plan
6.170 Final Project
Hiproute

Stakeholders

- Daily traveller who tries to maximizes efficient commute
- Providers such as Uber and Hubway can benefit from comparisons and see why they fall short

Resources

Computational constraints: Nothing much, but calling APIs might take some time
Costs: No costs
Time constraints: MVP deadline: Nov 18, Code for grading: Dec 2nd, Final delivery Dec 7th.

Tasks

1. Server side
    1.1. Integrating with external APIs (Uber, Google map, hubway)
    1.2. Integrating with cache
2. Client side
    2.1. dynamic google map background
    2.2. search box and search results display

Intermediate task (MVP):

1. integrating with google API - Shidan, Jason
2. integrating with Uber API - Julian
3. integrating with cache - Max
4. dynamic google map background - Julian
5. search box and search results display - Max

Addition task for Final:

1. Integrating with hub API - Jason, Julian
2. preference for costs vs time - Shidan
3. session search history - Max

Risks

1. Shidan's leg is broke. He has trouble moving. Solution: use Skype for meeting
2. Julian is going to SF for 3 days from Nov 19th to Nov 21st. Solution: three of us share his work when he's gone.

3. A person gets stuck on a task which blocks the progress of others. Solution: ask the team for help!
4. A person finishes too early. Solution: assign him something else to do.

Minimum viable product

For our Beta release, we have outlined that we must be able to provide an overlay onto google maps. This overlay is our user interface, which at a minimum will allow users to select their origin and destination for searching. We have determined that our Beta release will include costs, time estimates, and agony rankings from all supported google maps modes of transportation. We will also include Uber searches in our user interface.

Our data model will include a cache of Uber searches. We assume that google maps will be highly reliable, so we will only use our cache in the case that the uber API is down. If this occurs, we will use our searches within a half hour and a reasonable radius of both origin and destination. We know that Uber data becomes irrelevant rather quickly, so we will continuously flush our stored results that are more than a half hour old.

This MVP Excludes a few features. We will not initially support the Hubway API. This choice was made so that we get a better understanding of how to utilize the APIs of Google and Uber before we move on to more ambitious integrations.

Our MVP will cover all of our concepts: agony, search, route, provider, and cache. However the calculation of agony will be rather basic with predetermined algorithms. The user has less control over how much each factor (time, cost) weighs in the calculation of agony.

We will however postpone our security concerns to the final phase. The MVP will be more of getting a product that the user can use and compare the convenience between walking and taking an Uber to a specific location

In our MVP, we will include a link for users to enter their detailed feedbacks. Feedbacks will be carefully investigated and changes will be made in the final version.