CS 122 – Project Proposal – Team: Wildcards

Problem Statement

Construct a web interface for graduating CPS seniors to help calculate the cost of visiting chosen colleges for a few days. This tool is meant to be an aid in the college decision making process

Problem Description

While high school seniors apply to college, many apply for application fee waivers due to the financial strain of simply applying. Extrapolating from this, our group posits that CPS seniors who have a range of offers from college will only end up visiting a few of them due to the financial cost associated with the short trip to and from the college. Our tool intends to be a one stop shop for information regarding estimated financial cost of visiting a select range of colleges (top 100 college destinations for CPS seniors as provided by the CPS data portal). We hope that this tool will aid in the decision-making process of where to visit. Our tool will help visualize the costs of the visit in addition to providing helpful information about the selected college (expected graduation rates, cost of attendance, fun facts etc). School visits tend to heavily influence where a student ends up attending college. Our project helps to elucidate the costs associated with visiting, so that CPS students can make the best choice based on their financial constraints.

Source of Data:

http://cps.edu/Performance/Documents/Datafiles/CollegeEnrollment_2015_Citywide.pdf -- where do CPS high school graduates attend college – top 100 colleges

http://cps.edu/SchoolData/Pages/SchoolData.aspx - CPS Data Portal

http://developer.expedia.com/sample-stories - expedia API for flights, hotels etc.

http://www.travelotas.com/travel-api-providers/car-rentals-api-providers/ - car rental aggregator with an API.

Uber API – we may potentially use this to help estimate travel costs.

College websites and google maps

Note: This list will most likely be refined throughout the project.

Sketch of Work:

We will use a control flow structure in order to split travels into two main categories:

- IL/Midwest Region
- US-Wide

The first category will estimate cost of travels using metrics such as car rentals, uber/bus fares, gas cost and hotels. The second category will estimate costs using uber fare to the airport (ORD), flight cost, uber cost to hotel near college, hotel cost and potentially the uber cost from the hotel to the college (travel from the college back to hotel will also be included). Calculation of uber costs may be made with a function using average uber cost per min and mile + % markup. Distance calculation will be made by using a distance function along with the longitude and latitude of starting (CPS student zip code) and end locations (hotel/college zip code). We are hoping to use SQLite3 in order to facilitate the relational database query of finding the shortest distance and then using this to optimize cost.

The Distance calculations will assist us in deciding whether walking or taking a mode of transportation will be most appropriate. A control flow structure will be used to implement this feature.

We will have to create a website using HTML – on this website we hope to implement a tree map structure to help display the breakdown of costs once we return a total estimated amount. This tree map will cover both recursion as well as visualisation (it will be colour coded). HTML website will include drop down menus for college selection and a zip code entry field for conversion into longitude and latitude on the backend.

Where an API is not available, we will resort to web-scraping. We imagine that not all data obtained from web-scraping/API will be in a 'ready to go' format, as such, we imagine that we will be utilising Pandas for data munging/cleaning and presentation. We may also utilise the Pandas data frame as a form of internal data storage along with dictionary data structures.

College websites will be scraped for with regards to tuition, enrolment statistics etc.

Decisions based on which hotels will be included in our analysis will be derived from google maps/Expedia API. We intend to select the top 5 and use that information to discern cost calculations.

Rough Timeline:

W4: data scraping/cleaning for Expedia, college websites

W5/6: Uber, zip-code data and SQL querying

W7/8: control flow and function implementation, tree map design

W9: Backend link between python and HTML website. Testing/Debugging

W10: final touches/Submission