

Zachary Andrews, Austin Bristol

Progress Report I

The project we chose to pursue deals with picking an activity and/or location for a group of people to go to within a given area, ideally equidistant and interesting to the majority of the group. In order to perform this task we decided to use a third party API called Foursquare. This API supplies the user with a database of venues that have a wide arrange of information associated with each venue. Venues have an id, list of categories that they fall under, location, menu, hours, and a variety of other useful variables. So far we have created a majority of the base system for the user to interact with. This includes a simple command structure for creating users, viewing categories of venues, searching for nearby venues, and ranking user interests by popularity. A user has a variety of different values associated with it. These include a location, either as a city or latitude and longitude, an id, a list of food interests, and a list of general interests. These are used in order to find venues within a user's area which will later be narrowed down based on interest ranking. This ranking is based on the number of times a category is entered in by different users as an interest. These rankings will be able to be used in different ways in order to determine what venue is picked. The base way will be to simply use the category that has the highest number of occurrences within the group of users. Other methods include taking the top three categories to ensure that there is a bigger selection of venues and taking the category with the most occurrences and the least occurrences to try and achieve the same goal. The algorithm we have designed and believe will be able to properly get the venues that fall into the picked categories will first calculate the rankings for each category entered by the user, if multiple categories have the same rank they will all be used for comparison. The next

step is to use the information stored within each user- location (as either a city and state or a latitude and longitude) to search the region for venues that match the categories that are selected by the ranking function. After this the venues will then be compiled into a single list where they will be sorted by proximity to each of the users. This process will be use a conventional sorting method, though we have not determined which method will be used at this moment. After this the information (venue name, location, and other venue specific information) will be returned to the users so that they can pick which location they want to go to. Much of this plan is subjected to change as we expect that we will make changes in order to make the program run more efficiently upon runtime.

The pseudocode for this process will look something like this:

User creation (user id, location, interests inputted)

Users = list(users)

Function get_venues(users):

 Venues = list()

 Interest = rankInterest(users)

 For user in users:

 venues.append(location_search(user))

 For venue in pick_location(venues, interest):

 print(venue(name),venue(address),venue(hours))

Function location_search(user):

Return search(user(location))

Function pick_location(venues, interests):

For venue in venues:

For interest in interests:

If interest not in venue(categories):

Remove venue from venues

Return sort_method(venues)