Boulder has over 10 pages of restaurants and fast food places that a user can pick from. Because of that, it is hard to pick a new type of restaurant without risking the fact that it could be bad. Another reason could be that they want to try similar foods. In order to minimize that problem, we plan to make a restaurant recommender that would recommend similar restaurants to those the user has already tried.

In this project, we will make a program that allows the user to enter a restaurant that they like, and from there the program will suggest similar restaurants to the user's input. For this project, we are thinking about using a graph data structure to make the recommendations to the user. For example, given a restaurant the user has already saved or inputted, the program will search for that restaurant in the graph. Then, it will return a certain amount similar restaurants using the edges of the node of the given restaurant. We would prioritize restaurants within a certain distance from the user and how well the restaurant is rated. Another structure would be a linked list where the user's data can be saved. The linked lists will save information such as the user's preferences, top restaurants (contained in a user specific linked list) , and even the number of users that have decided to use the app. Another concept that we thought of is to have the usage of an API. We are thinking of using Google Places API to incorporate similar restaurant suggestions. The usage of that API will enable us to gather the user's location and suggest similar restaurants within its category nearby the user. In addition to that, we might add distance an option for the user to decide how far they want the trip to be.

As far as the implementation goes, we are planning to bring this idea to mobile devices. There will be a version for IOS and Android. We plan to use a mobile framework by google called Flutter, which allows us to code in Dart that allows code to be converted into native swift and java to be run on both ios and android devices from a single code base. Flutter also allows us easy access to various api's designed by google as simple add on plugins that can be imported directly into our code. It also

allows an easy to build UI through simple code and widgets instead of having to mess around with xml's. We might also make a computer version and having the output through the terminal if we have issues with the dart language in creating our graph's.