

## REDOT

Singapore is not a complex city to most of us but it is rather confusing to the tourists as we have street names that are adopted from multiple languages. It would definitely not be a fulfilling travelling experiences to Singapore if not being able to taste the wonderful local food and other Asia cuisines. This is where REDOT comes in, we help you find you way if lost, we lead you to the wonderful restaurants if hungry, what's more, we have a smart solution for all these. With our algorithm, we can help you find a way leading to your destination with a reasonable fare while also having an acceptable duration.

### Not just tourist attractions

When visiting a country, we usually go for the landmarks first. They are not that hard to find, even if happens to get lost on the way, we can always ask people passing by to help, and the familiarity of these buildings among people, we usually managed to go there without much difficulties. However, this is not the case for food, and food is definitely something people don't want to miss while traveling in Singapore. Restaurants are not like landmarks, you can't just find somebody on the street and ask about their favorite restaurants, whatever they like may not be the best fit for your taste and yet their favorite restaurants may be miles away. The solution we found is by using JSON, we can refer to the restaurant review website such as TripAdvisor, and by doing so, we ensured the reviews and results we get would be based on a rather large amount of data and should be the best possible fit to your appetites.

### Two algorithms

#### Brute force

We have brute force as a very basic algorithm. This might be a rather abundant for other types of inputs and selectors for it would confirmed too much time and space to solve a question like this. If the data of places, we have collected is not too complicated, brute force is actually a rather acceptable algorithm to solve the problem of minimizing and in this case, how to find a lowest cost. As it is going with all possible solutions, this method can give us a more accurate result with something that is definitely the shortest way leading to the destination.

#### Fast solver

If we changed our idea or being in a hurry, we can always go for the fast solver and this can lead us to a way of approaching our destination with the shortest amount of time. However, there can be mistakes in this method because of the choices towards different types of transports. It is possible that the path with minimum walking time can actually take shorter time if some part travel by taxi.

### User Interface

The UI is clean and easily understood. We start with an interface prompting us with the famous tourist attractions of Singapore, from which we are able to choose some as the places we are planning to visit today. We are able to type in the budget we have for transportation so that the program itself will filter out the means for transportation that might cost more than the budget. We will also need to choose a way we want to travel, from brute force and fast solver. When the first option is chosen, the system will show us a way with shortest duration for transportation while the cost shall be under the limit for our

traveling budget. The second option will lead to a way with shortest walking time so that we can enjoy the trip while not being too tired, and the cost for this ways of transportation will also be controlled by the budget we just typed in.

The second part of the three main interfaces is the google map. We will be able to see our real time location and can search how to travel from one place to another. The different means of transportation are marked using different colours on the map where green stands for taxi, red for public transportation and blue for walking.

The third part is the wish list. We can add places here so that it would be easier to access and easier to check out.

The common part among these three parts is a directory for these parts, allowing the user to moving freely between these functions.