MVP - Version 2- IshLa

**AIDI 2005 CAPSTONE II**

**Course Facilitator: Marcos Bittencourt**

**Prepared By:**

Izza Godinez - 100556078

Maviya Javed Shaikh - 100766785

Meryl Gabrielle Tubio – 100763231

Nandini Malhotra – 100768797

March 12th, 2021

Table of Contents

[1. Executive Summary 3](#_Toc66387105)

[2. Rationale Statement 3](#_Toc66387106)

[3. Problem Statement 3](#_Toc66387107)

[4. Data 3](#_Toc66387108)

[5. User Functionalities 4](#_Toc66387109)

[i. Retrieving User Address: 4](#_Toc66387110)

[ii. Retrieving Places User want to visit and creating a list: 4](#_Toc66387111)

[6. AI Agent 4](#_Toc66387112)

[7. Optimal Path Map 5](#_Toc66387113)

[8. Optimal Path Route 6](#_Toc66387114)

[9. Next Steps 7](#_Toc66387115)

[10. Project Plan 8](#_Toc66387116)

# Executive Summary

Optimal path algorithms are a major interest in artificial intelligence. It has many applications in business, including product delivery and shipment processes, as well as in navigation. This project aims to allow the user to create an optimal path between different stores to allow them to complete their shopping lists with various items in the shortest possible time.

# Rationale Statement

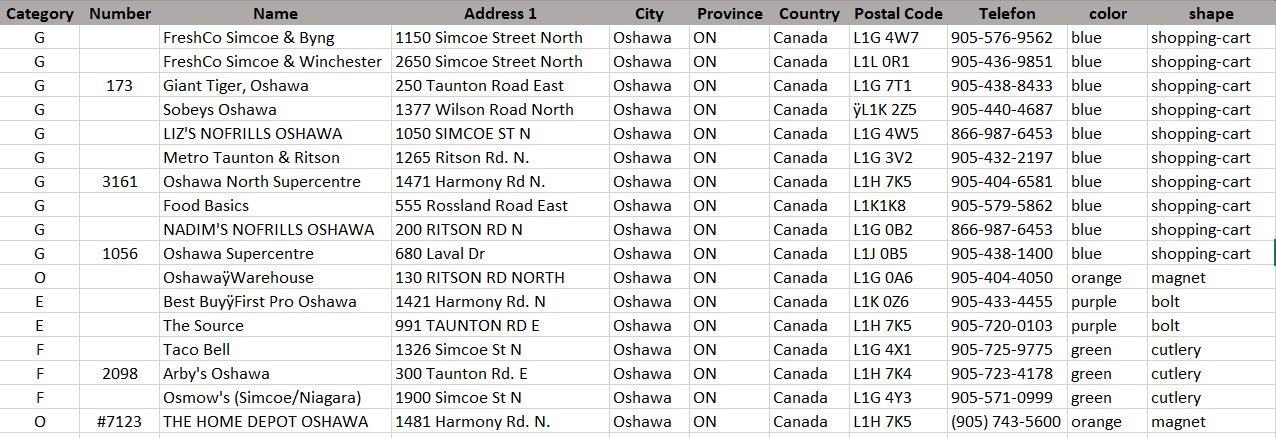
To develop a website that takes in user credentials and asks the user to select their TO-DO list by clicking on various sub-categories like groceries, electronics, clothing apparel. The user selects items from these sub-categories. Based on the items selected, the website suggests a route from the user's current location that he can take to complete all its errands in one go.

# Problem Statement

“To find the optimal path from the starting point (source), covering all the sub-destinations”.

The problem statements aim at determining the closest sub-destination from the current position, the starting point being the user’s current location.

# Data

A dataset containing the addresses of various destinations is used to help find the optimal path from the user’s current location to various sub-destinations. 

Where:

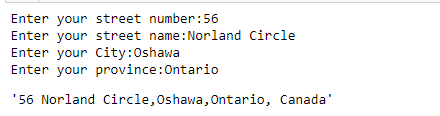
1. **Category:** It is the category of the store for various kinds of errands like Groceries(G),Electronics(E), Food(F), Clothing(C), Others(O), which might include Home Depot, Gas Stations and Car Wash
2. **Number:** It includes the Store Number if valid.
3. **Name:** It includes the name of the store.
4. **Address 1:** This field contains the street number and street name of the store.
5. **City, Province, Country, Postal Code:** These fields would specify the city, province, country and postal code of the store.
6. **Telefon:** This field would contain the telephone number of the store.

**color, shape:** These two fields are solely for the purpose of plotting the destinations on the map intuitively.

# User Functionalities

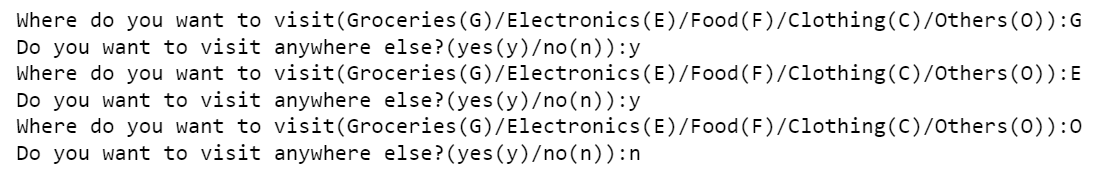
## Retrieving User Address:

Asking the user for their address to get the user’s street number, street name, city, and province.



## Retrieving Places User wants to visit and creating a list:

Asking the user to select the category of the product user wants to buy. In this case, the user is selecting three categories i.e. Groceries, Electronics, and Others.



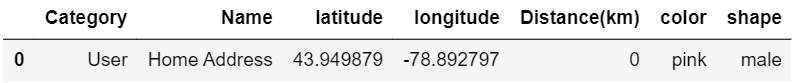
# AI Agent

AI agent is the artificial intelligence component that helps to achieve the goal of our project. Based on the business problem to find the shortest path for the user, we will be using Utility-based AI agents. The reason for choosing a Utility-based AI agent is that it not only helps to find the shortest path, but it also provides the best possible way to achieve it. Another advantage of using a Utility-based AI agent is that in our project there will be multiple alternatives i.e. multiple paths to choose from and agents must choose the best path amongst all.

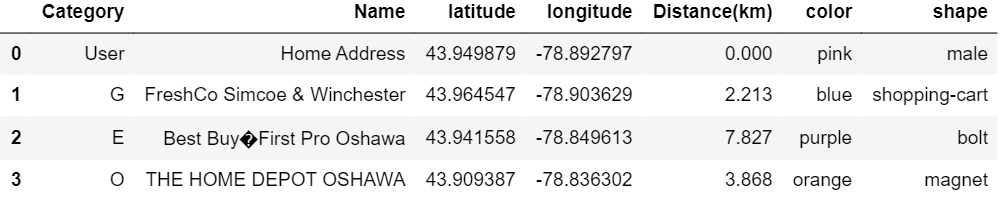
**Calculating Distance from one location to multiple locations using AI Agent:**

We are using two methods to calculate the distance. The first is **Open Source Routing Machine** and the second is **Haversine Distance.** It was found that Open Source Routing machine method provides more accurate distance than the Haversine distance method by validating the result with Google Map.

The distances are calculated iteratively depending on the current location. First, the user information is stored in the optimal path dataset as shown in the below table:



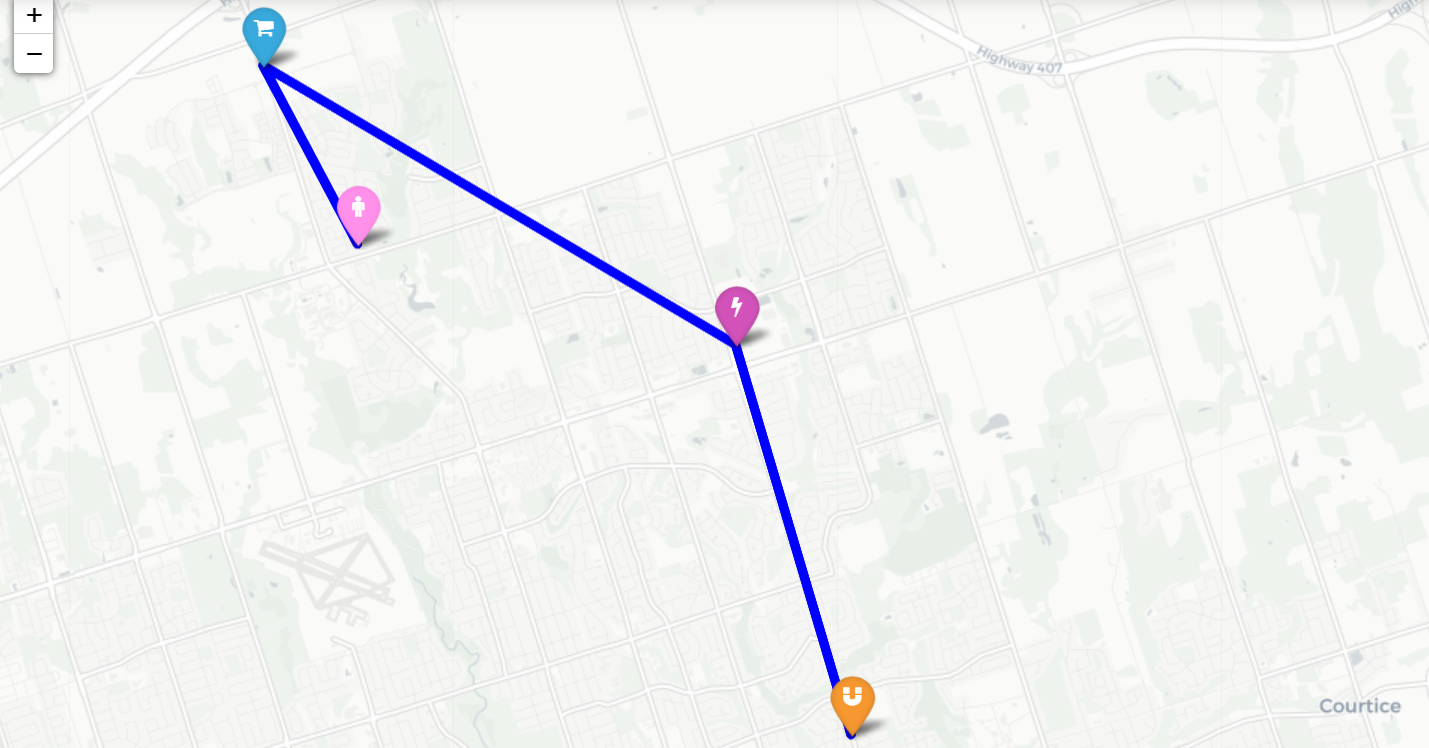
AI agent will first take the above user input as a source and then calculate the shortest path to show the nearest store using the Open Source Routing Machine method then in the next iteration user will choose that nearest store as the source and then calculate the shortest distance to show the other nearest store which user want to visit and so on. Therefore, the final Optimal Path dataset based on the user input is as shown below:



The nearest store the user can visit is the Grocery store Freshco then from the grocery store the nearest store will be the Electronics store Best Buy and the final store user want to visit is Home Depot.

# Optimal Path Map

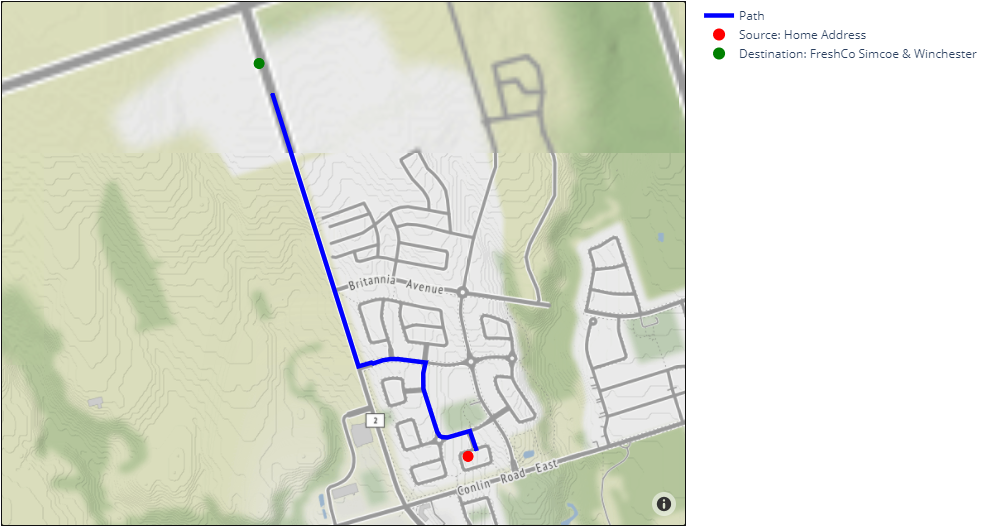
Here, we are visualizing the map of the nearest store user want to visit based on the optimal path dataset using the folium library.



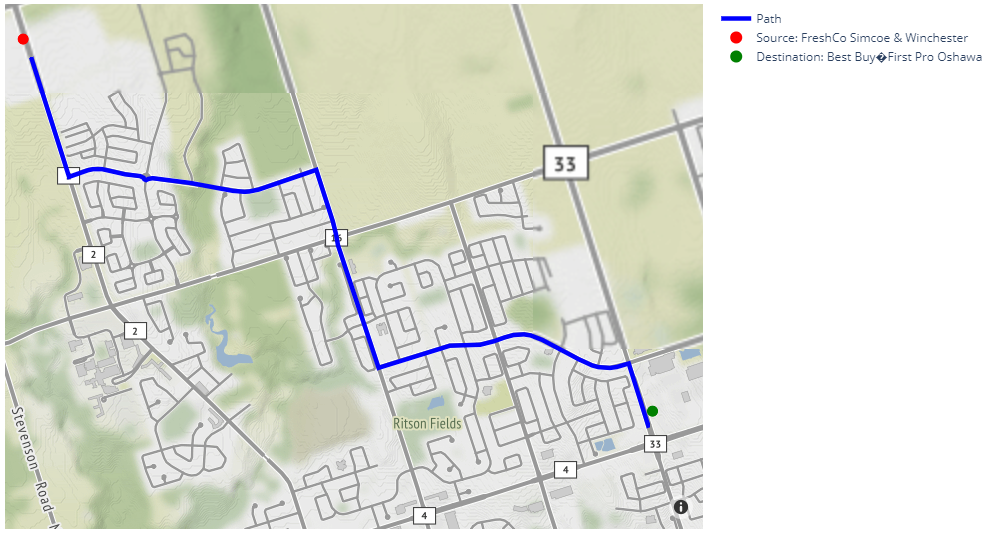
# Optimal Path Route

As shown in the above optimal path map, we have the entire location of all the stores from the user destination, but practically we need the driving route to visit the store one at a time. As users want to visit three stores, below are the three routes user can visit.

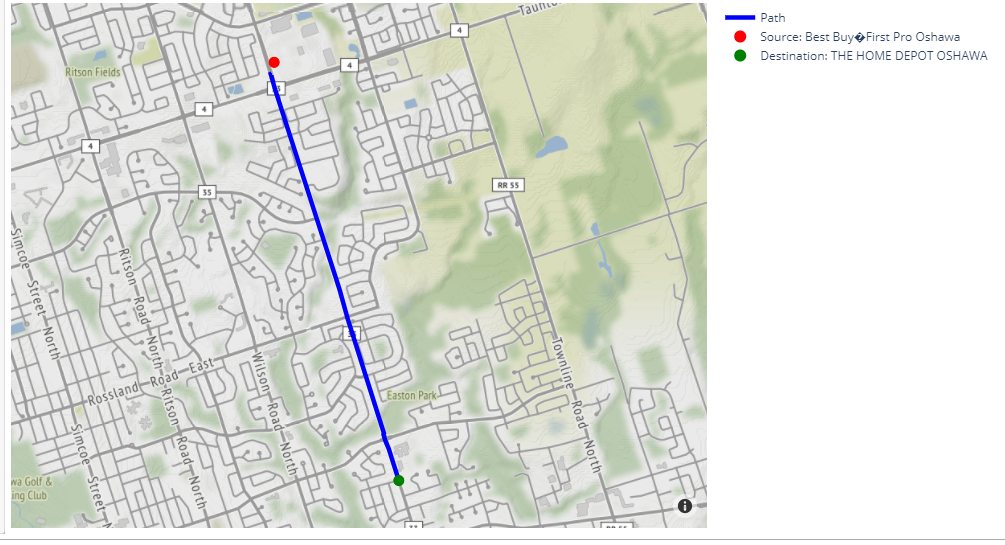
1. Route from user address to Grocery store FreshCo



1. Route from Grocery store FreshCo to Best Buy



1. Route from Best Buy to Home Depot



Finally, we get the driving route of the nearest store and the backend of our project is completed.

# Next Steps

The next steps to be taken are as follows:

* Designing the user interface
* Integrate user interface with backend
* Testing the route with different combinations

# Project Plan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Milestone** | **Submission Date** | **# of hours** | **# of Resources** |
| 1 | Submit Project proposal (including: SOW, ppt slides, github profile) | Feb 1, 2021 | 10 | 4 |
| 2 | Minimum Viable Product (MVP)  (version 1) | Feb 19, 2021 | 10 | 4 |
| 3 | MVP (version 2) | Mar 12, 2021 | 10 | 4 |
| 4 | Minimum Marketable Product (MMP) | March 26, 2021 | 10 | 4 |
| 5 | Product to launch | April 11, 2021 | 10 | 4 |
| 6 | Final Evaluation | April 16, 2021 | 10 | 4 |