# 24/11/2021

**Meeting Notes**

* Use WebGl, to visualise the map in a browser without plugins. Can be used on top of Python?
* Use a GitHub repository for keeping and updating the project.
* Find out how to calculate the average costs of fuel per nautical mile.
* Hull weight, port traffic, weather conditions.
* Take also into account that new regulation is to be imposed for fuel consumption, which could lead to increased costs and delays.
* Contacted ExactEarth for a quote regarding data related to shipping positions, including past data.
* The company contact, told me that they have the start, end port and the stops in between but they cannot provide tangible data to be used in software development, because they do not keep it that way. Their archive is consisted by text documents.

**Logic:**

* Use Dijkstra’s shortest path algorithm with weighted edges?
* Use dynamic weights on each edge.
* The weights will be dependent on variables that determine the efficiency of the route.

**Useful links:**

* UCL GlotraM: <https://www.ucl.ac.uk/energy-models/models/glotram>
* Exact Earth (will be used for the exact locations of the ship): <https://www.exactearth.com/industries-fleet-management>
* Previous applications of my idea: <https://www.shipmap.org>
* Fuel Costs (Link 1): <https://www.morethanshipping.com/fuel-costs-ocean-shipping/>
* Data (Link 1): <https://datasetsearch.research.google.com/search?query=ais%20vessel%20data&docid=L2cvMTFqOWJfZG1xaA%3D%3D>
* Calculate LatLong distances, library: <https://geopy.readthedocs.io/en/stable/#module-geopy.distance>

Background research:

1. I have a map that says nodes and routes between them. The found on the last meeting can be used. The map defines the paths. Then use a path finding algorithm.
2. No specific paths, only use the START and END points and use past live location.

Find and add information on the nodes.

Maybe use past data to determine the weights

If there is a lack of data, use region specific ones.

**Focus points:**

1. **Figure out how to load the geojson file.**
2. **How to calculate the distance between nodes.**
3. **How to implement and use Dijkstra’s algorithm.**

Reading a bit more on route planning algorithms.

Is the map going to be a planar or a curved surface?

If we assume that going through and extra node then it is longer, it should not matter if the surface is curved or flat.

Assume the route is direct between two points.