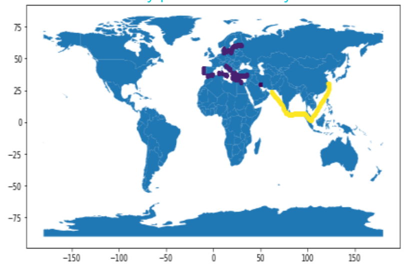
hackathon MERCURIA

Winning solutions: Sustainability Stars

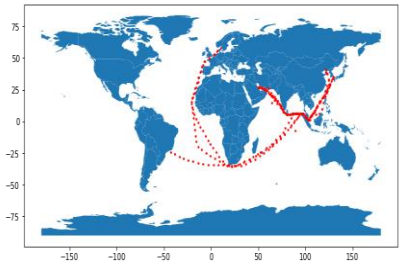
Challenge 1

- Started off analyzing the data for Step 1 in both Challenges 1 and 2 and did some data clustering to see how correlated it was.
- Visually comparing the data gave insights to move forward and correlate the data with same voyage path in the past and minimizes the exploration time in dataset.

Yearly plot of AIS hourly data



Voyage plot for Hercules 1



- Only found the noon reports mid-Saturday so knew not much time to analyze them with the AIS hourly dataset. Chose instead to simply plot the noon report data which required conversion of compass-based coordinates to actual Latitude and Longitude. Plotted the data for Hercules 1 and Amber ships, saw the similarity of the routes, checked the dates and understood that the data from Hercules 1 noon reports can be used further to extract the different trips.
- Selected one trip and reverse engineered to get a factor that could be used with AIS to derive average fuel consumption and then GHG emissions using some assumptions. Used an online API to calculate distance to then calculate GHG according to bunkers consumed.
- The function used in the previous step, produces a report using the AIS data and the factors calculated at midnight for the previous day:
  - **`total distance travelled 3146.7155295920047 km`**
  - **`average velocity  7.2820104802365595`**
  - **`average consumption  1.3380893139853856`**
  - **`Total GHG emissions  890.5519620298337`**
- Using a script produce a csv with AIS and calculated data on a daily basis and store on the server for each ship IMO. This is extended and used as a base to calculate the GHG emissions in Challenge 2.
- Open Meteo API was used to get the weather data from every hour according to the AIS GPS coordinates of the vessel, but did not give any additional information for the factors to calculate the different values in step 2. Maybe further analysis could provide additional information to utilize the weather data.
- The comparison with the manual reports revealed issues as the noon report data has just two points to calculate the trip distance as a straight line, whereas AIS data has more data points so distance can be calculated with more accuracy. Therefore the entries for a day can be compared to the AIS data and the “wrong” values highlighted.
- Comparing reports for an indicative journey of Hercules 1 found deviations to AIS data as follows:
  - Velocity 3%, GHG 30-35%, Distance 3-6%, Total fuel 8%

Time allocation

Friday evening + early Saturday:

- Getting to know each other and setting up code server
- Analyzing data for Step 1 for Challenge 1 and Challenge 2

Saturday morning:

- Writing python scripts to download data
- Containerizing and setting up Docker + Flask API
- Set-up of dev environment
- First clustering for data correlation
- Weather data correlation with trips, revealed minimum effect on results for the trips selected to analyze

Saturday afternoon:

- Found noon reports
- Lots of CI/CD debugging
- Calculating conversion factors from noon reports, and crosschecking trip information and AIS data
- Setting up challenge 3 API endpoints

Saturday evening:

- Debugging static web apps files not exposing to http server
- Distance calculation for challenge 1, clustering of the vessels, comparison of noon reports and AIS data
- Neural network regarding trips (limited data, model not able to generalize)

- Vessels clustered according to one trip performed between certain ports and their DWT. This revealed that only a limited number of clusters were present according to the data available and there was no added value in correlating additional ship information, only DWT and category is required.
- With this, compared trip data between the same ports for the largest 500 vessels and created the ranking of 100 worst performing ships.

Challenge 2

- Using templated trips (arrival-destination port), DWT and IMO, calculated overall distance for the trip using AIS data.
- Used the calculated factors from Challenge 1, this voyage and AIS data to calculate average fuel consumption for a certain trip by calculating the distance travelled every day from past AIS data.
- Projected GHG emissions for the trip was calculated using the projected fuel consumption.
- Expected time taken can be extracted from previous trips and averaged when the vessel has performed the trip multiple times.
- Built an LSTM neural network to analyse the trips and find the different possible routes according to past trip data.

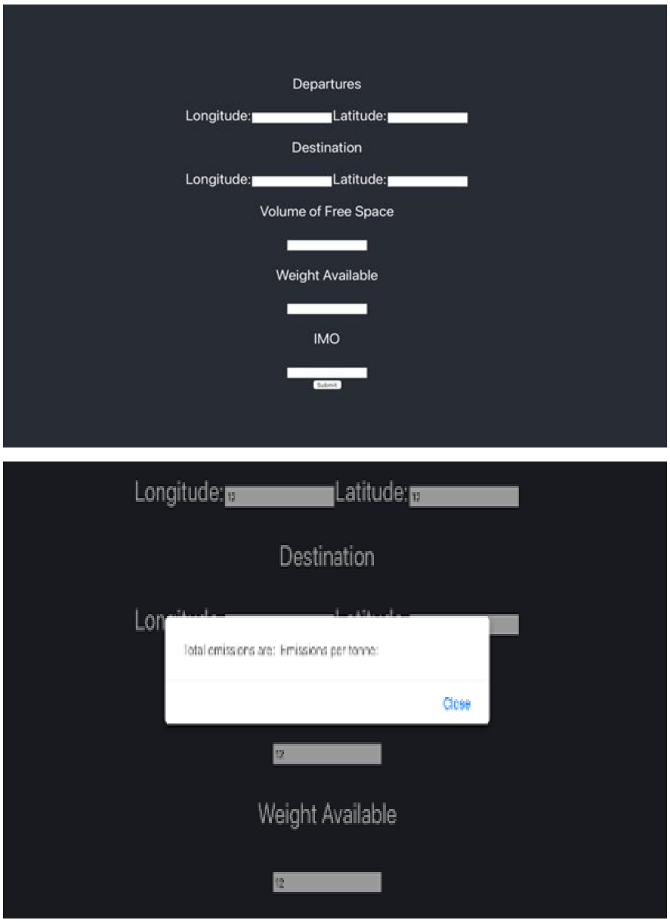
Sunday morning:

- Changing challenge 3 API schema and setting up/connecting MySQL database
- Debugging python not committing changes to database
- Finalizing challenge 1
- Connecting challenge 1 functions to calculate the voyage emissions, emissions per tonne of cargo, time required, and calculate 1-month trips for some vessels according to the different vessels categories for challenge 2

Challenge 3

- Database ←→ API (Python backend) ←→ React (web) app
- Used React as could be easily packaged and used within Docker, plus the server build would be condensed allowing browsers to compile efficiently.
- Simple design using 7 variables and sends a request to the API;
    - Vessel operator: IMO (vessel identifier), latitude/longitude of departure resp. arrival port, volume/weight of available space
    - Cargo operator: latitude/longitude of departure resp. arrival port, volume/weight of needed space
- Flask API + MySQL check for potential match (exact voyage, else nearest)
    - first checking volume and weight for matches
    - then searching for combination of nearest dep./arr. harbors (computationally more efficient)
- User is then shown the *Total emissions* and *Emissions per tonne*.

Example web app input & output screens



Sustainability Stars