INTRODUCTION

With over 275 million tonnes of plastic waste on the planet, waste disposal is no minor issue. Every year, around 100 million marine animals and plants perish due to oceanic waste alone. With 4 billion pieces of microfibers and microplastics in the ocean, humans end up being a victim of this global issue as well. This definitely tells us that proper management of waste is essential for our well-being and an efficient system to do so is definitely a huge asset. Through efficient waste management methods, we are not only preserving marine life but also the environment and humankind.

DESIGN AND DEVELOPMENT

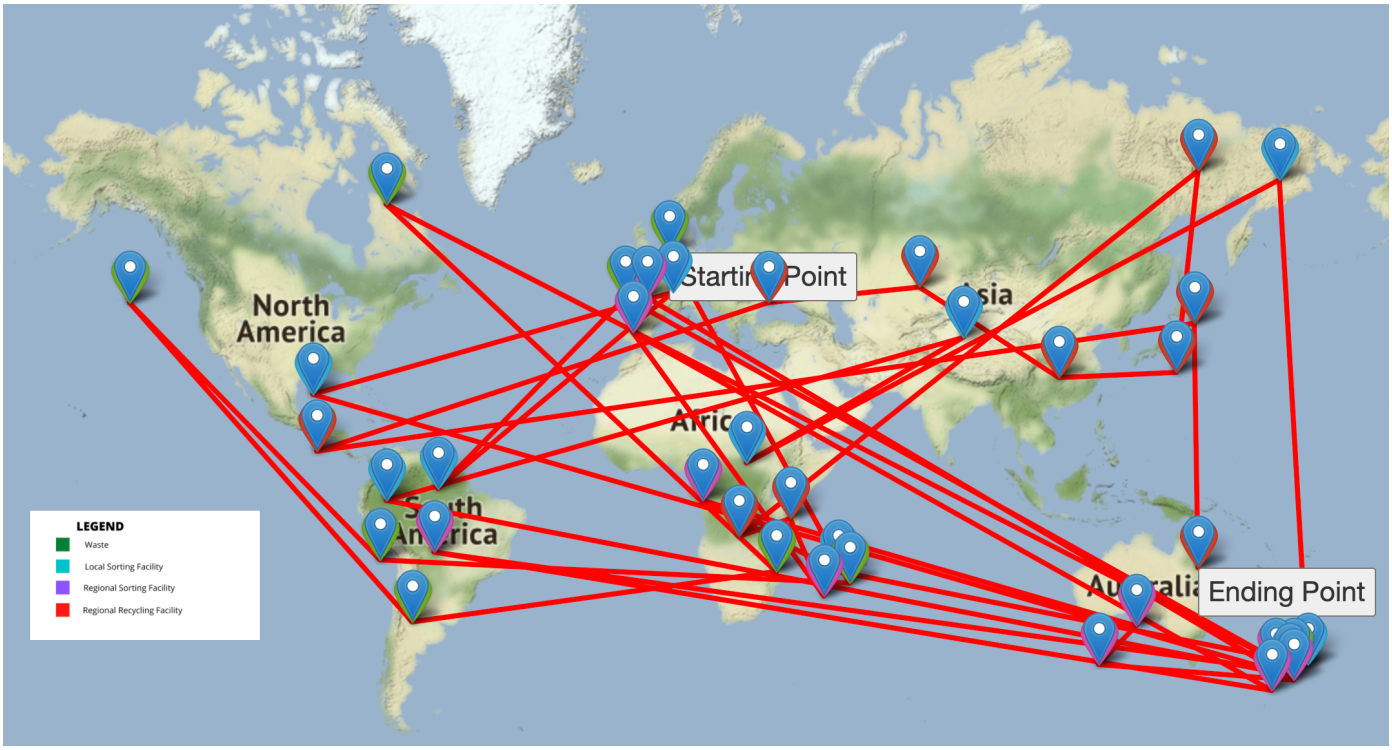
The GUI runs on the **Tkinter** Python framework where users can interact with garbage collection and disposal data. Once the user enters the filename and required parameters, we call the main algorithm to devise an efficient path for the garbage truck.

The back-end involves an algorithm to read the given CSV files and return a reordered CSV file based on the path of the garbage truck.

First, all the garbage is collected from each garbage pickup location before heading to processing. A **virtual boundary** is created around the collection of points and a **hashmap** is created.

Starting from within hash where the **starting point** is, we move to the closest adjacent point within the hash and iterate till all the points within a hash are covered before moving to the adjacent hash. This is repeated till all the hashes are covered.

From the location of the **ending point,** the most efficient combination of facilities is acquired using a **pseudoQoR** that's calculated using **distance** and **waste lost. Merge sort** is used with the pseudoQoR formula to boost efficiency.



CONCLUSION

The data from the CSV file that is produced once the algorithm is executed is then read into a **pandas** data frame. The latitude and longitude data are then read to create a **GeoDataFrame** object from the **geopandas** library.

A terrain world map is developed using **Folium** by reading the GeoDataFrame. The points are labelled, the starting and ending points marked, and the legend tabulated. Finally, the points are connected in by the order of index to show the path that the garbage truck takes.

The map is stored in an HTML file using the **webbrowser** library and is then opened in a web browser where the user can interact with the map.

The map is generated in a HTML file with the help of the webbrowser library in Python. All the **coordinates** are mapped and each point is represented by a green, blue, purple, or red point that can be inferred from the legend.

The **starting point** and **ending point** are clearly mentioned. The user can zoom in and move around the map to trace the **route** taken by the garbage truck.

The necessary Python libraries for using the program are tkinter, pandas, geopandas, folium, webbrowser, and csv. The main.py file must be executed and the tkinter library ensures that the other files are run thereby automatically creating an HTML file that shows the required map.