# Optimizing Search for Lost Objects in Water Bodies

Project Report

by


**Anushka Jadhav**

**Atharva Jagtap**

**Manan Kher**


Guide Names:
**Prof. Anand Godbole**
**Mr. Raj Mehta**


Computer Engineering Department
Bharatiya Vidya Bhavan's
Sardar Patel Institute of Technology
Munshi Nagar, Andheri(W), Mumbai-400058
University of Mumbai


May 2024

# Acknowledgements

---

We feel great pleasure in presenting the report of our mini project titled **'Optimizing Search for Lost Objects in Water Bodies'**. We have channelized our best efforts towards a systematic approach to the project, keeping in mind the aim we need to achieve.

We are truly grateful to our project guides: **Prof. Anand Godbole and Mr. Raj Mehta**, Department of Computer Engineering, Sardar Patel Institute of Technology (SPIT) for constant encouragement, effort and guidance. They have always been involved in discussing our topic at each phase to make sure that our approach was designed and carried out in an appropriate manner and that our conclusions were appropriate, given our results.

**Anushka Jadhav**

**Atharva Jagtap**

**Manan Kher**

# Abstract

This project utilizes **Bayesian Search Theory and A* Search Algorithm** to efficiently locate plane crash sites in oceans. A website interface guides users through inputting flight data and recovered debris locations. Automated data acquisition via Selenium and simulation utilizing various probability distributions optimize search paths. Key findings include efficient retrieval paths and probability updates via **Bayes' law**. The project yields downloadable CSV files detailing optimal search paths and associated probabilities, enhancing search efficiency and effectiveness.

# Contents

# Chapter 1

# Introduction

## 1.1 Problem Statement

Create a product for economical and efficient retrieval of objects such as airplanes, ships and boats from extensive water bodies like seas and oceans. Use relevant data like flight path, last known position of plane, recovered bodies/debris as well as geographical data like wind and ocean currents to create a probability distribution. Perform an optimal search process on the created probability grid.

## 1.2 Literature Survey

- Search Analysis for the Location of the AF447 Underwater Wreckage, Report to Bureau d'Enquêtes et d'Analyses (Lawrence D. Stone, Colleen Keller, Thomas L. Kratzke, Johan Strumpfer.) (Dated: 20 January, 2011)
- Search for the Wreckage of Air France Flight AF 447 (Lawrence D. Stone, Colleen M. Keller, Thomas L. Kratzke and Johan P. Strumpfer.) (Dated: 19 May, 2014)
- A Bayesian Approach to Finding Lost Objects. (Katie Howgate)

## 1.3 Objectives

**Developing algorithmic solution for maximizing the probability of finding the missing object.**

The project aims to develop an algorithmic solution to maximize the probability of locating any missing object within extensive water bodies. This results into the creation of a framework capable of efficiently analyzing input data, including flight paths, last known positions, and geographical factors. By leveraging advanced algorithms and techniques, the project seeks to devise a systematic approach that optimally allocates search resources, thereby enhancing chances of successful object retrieval.

**Implementing Bayesian Search Theory, for dynamic probability updates.**

This project involves the implementation of Bayesian Search Theory to facilitate dynamic probability updates throughout the search process. It incorporates the continuous refinement of probability distributions based on incoming data and search outcomes. By adopting Bayesian principles, the algorithm can adjust the chances of finding the object in different regions of the search area, ensuring that search efforts remain focused and effective in real-time scenarios.

**Optimizing search paths, minimizing search time and cost.**

The project aims to optimize search paths with the goal of minimizing search time and cost. This optimization process involves careful consideration of different factors like geographical and environmental conditions, and available resources. By employing advanced optimization techniques and heuristic algorithms, the project seeks to identify the most efficient search paths that maximize coverage while minimizing resource expenditure. Through careful path planning, the project aims to streamline search operations, thereby accelerating the object retrieval process and reducing associated costs.

**Integrating real-time data for creating considerably accurate probability distributions.**

Another critical objective is the integration of real-time data to create considerably accurate probability distributions. This involves sourcing and incorporating up-to-date information such as climate patterns, ocean currents, and environmental conditions into the probability modeling process. By leveraging real-time data sources and advanced data processing techniques, the project aims to generate probability distributions that accurately reflect the current search environment. This integration of real-time data enhances the reliability and effectiveness of the search operation, enabling search and rescue teams to make informed decisions based on the most up-to-date information available.

## 1.4 Assumptions

- User should be a part of Search and Rescue team.
- User should have relevant data about the flight ready.
- User should authorize themselves before the service.
- Stable internet connection is required.

## 1.5 Constraints

- The search grid has been limited to 1.5° latitude $\times$ 1.5° longitude.
- The search process should commence within 2-3 days from the time of last signal.
- API call has a limitation of only 1 day before the current day.
- Gives correct results when the searching is conducted above water bodies.

# Chapter 2

# Proposed System

## 2.1 Workflow

The user of this service (Search and Rescue Authorities) will either make an account and register themselves for using this service, or if they already have an account, they will log themselves in the account. After verifying their credentials, authority personnel will be logged in their account, where they will be able to use the prediction services.

On the services page, user will enter data about the source city and destination city of the flight, and the last known location of the flight (location from where last signal was received). The data will help generate the flight path along which the Gaussian elliptical distribution will be plotted. After entering the data, user will proceed by clicking the 'proceed' button.

This will start a process of extracting bathymetric data from the Gebco website by automation using Selenium in the background, which will be used for further calculations and distributions.

If authorities discover any debris or dead bodies beforehand, they can report debris related data in the website by including the debris location and its recovery time. The debris' trajectory will be projected backward, with locations calculated for each hour preceding the recovery time. Utilizing the unsupervised machine learning model: the Kernel Density Estimation (KDE) technique, these projections generate the Reverse Drift Distribution. Inputting debris data is optional, and the Reverse Drift Distribution is only generated when such data is provided.

As mentioned in the published paper 'Search Analysis for the Location of the AF447 Underwater Wreckage, Report to Bureau d'Enquêtes et d'Analyses – Lawrence D. Stone, Colleen Keller, Thomas L. Kratzke, Johan Strumpfer. (20 January, 2011)', "During flight, a commercial aircraft sends messages via satellite containing maintenance and logistic information about the aircraft. Every 10 minutes it sends a GPS position for the aircraft". This gives us the last known location, around which we distribute the probabilities on map, and these distributions help to find the lost airplane. The distribution incorporates the Ocean Depth Distribution and the Uniform Circular Distribution, which are combined together (multiplied), and added to the Elliptical Gaussian Distribution in a 1:1 ratio. The resulting distribution is then combined with the Reverse Drift Distribution in a 7:3 ratio to produce the final distribution.

Upon proceeding with the simulation, the final distribution is visualized on a pygame GUI embedded within the webpage using pygbag. A red-colored searching entity represents the optimal search path visually. The algorithm for determining this optimal path is discussed subsequently. Following simulation viewing, users can download a CSV file listing locations sorted according to the optimal path. This file facilitates the efficient and economical execution of the actual search operation.

The figures 2.1 and 2.2 on the next page explain the workflow of the web application using a flowchart.

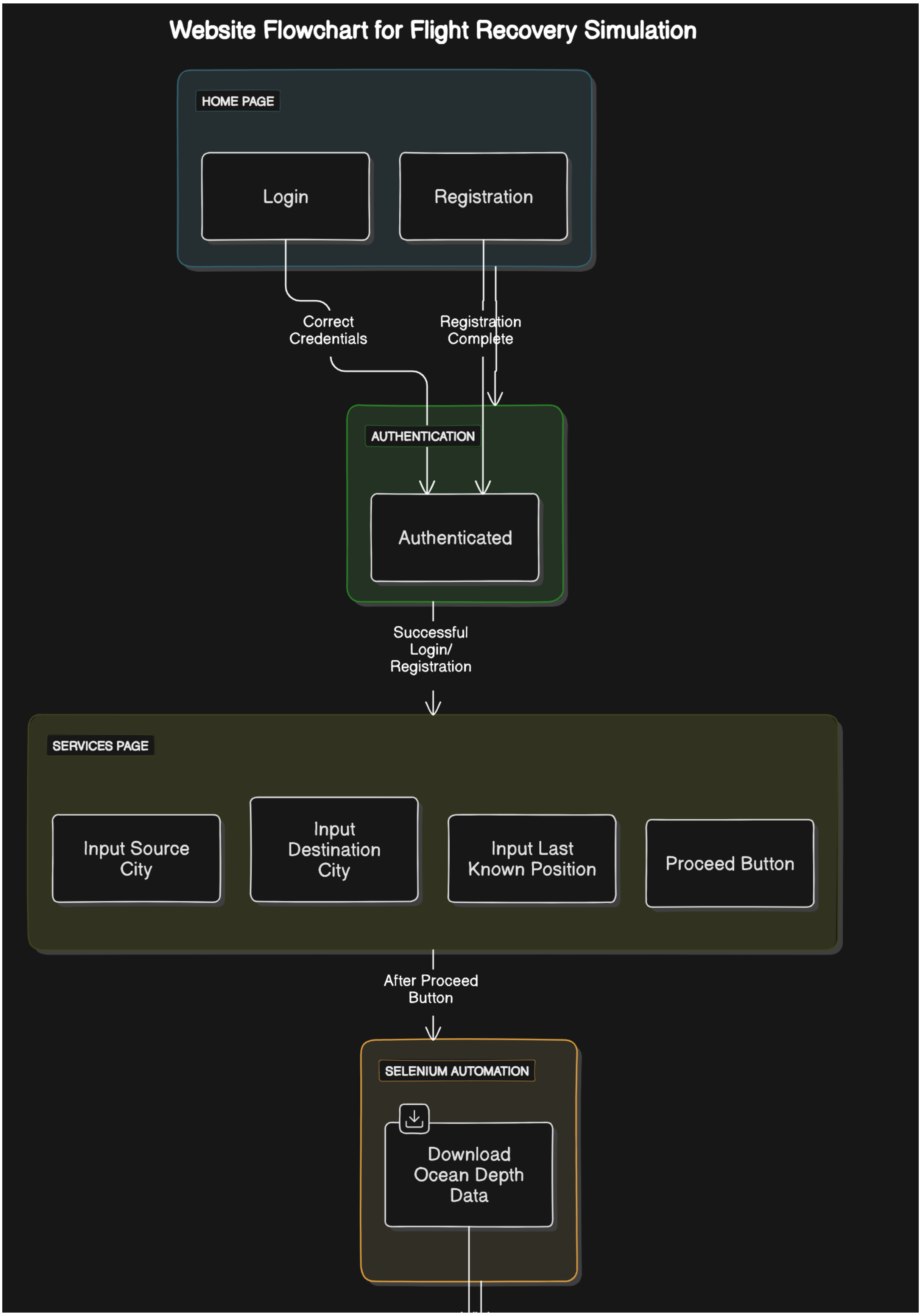


Figure 2.1: First half of the flow

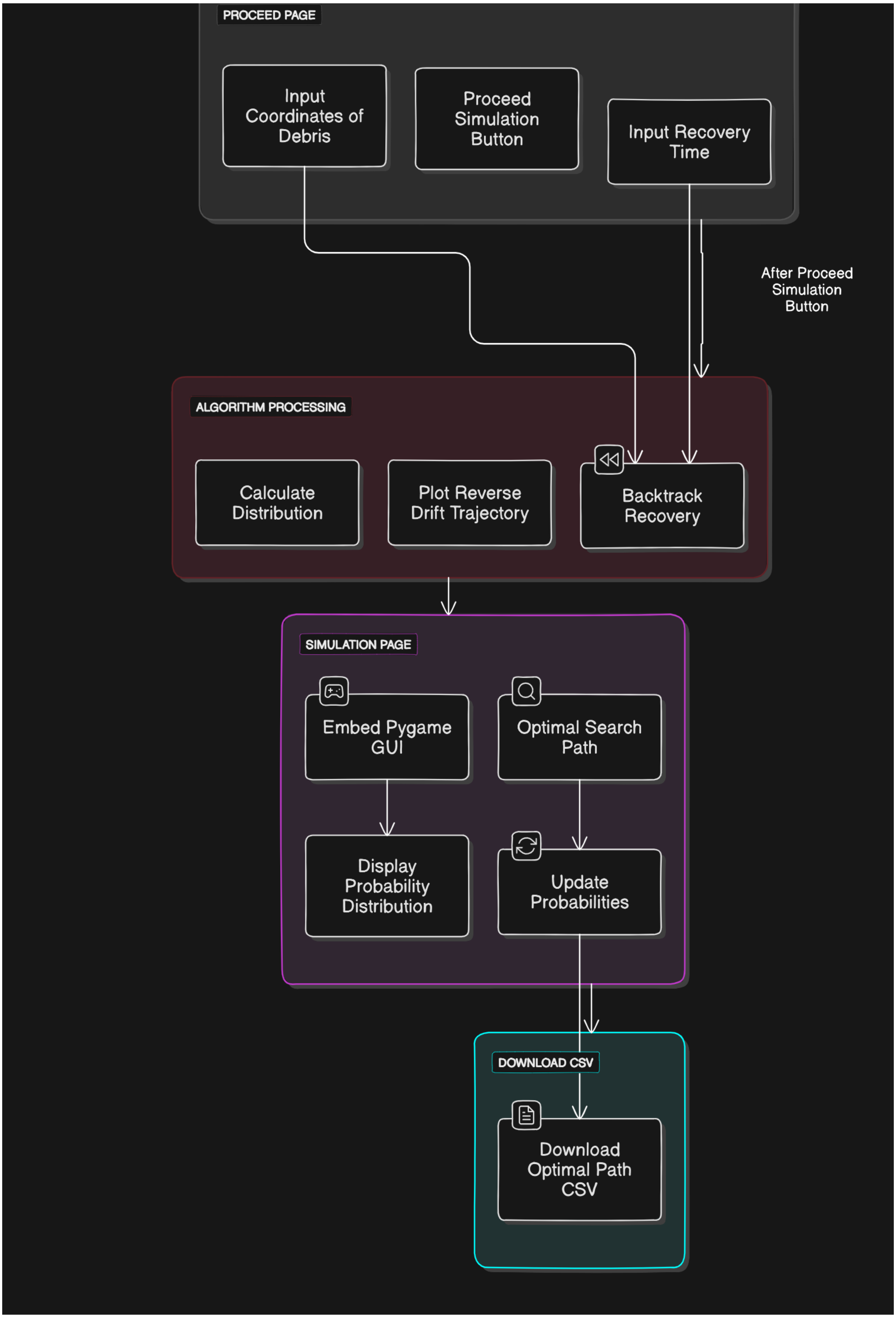


Figure 2.2: Second half of the flow

## 2.2 Algorithms

The project is based on Bayesian Search Theory, the updation logic for sub regions/cells in the map is as follows:

- Updating probability of the explored cell where the object was not found:

$$P[IsThere|NotFound] = \frac{P[IsThere](1 - P[Found|IsThere])}{(1 - P[IsThere]) + P(IsThere)(1 - P[Found|IsThere])}$$

- Updating probability of other cells:

$$P[IsThere] = \frac{P[IsThere]}{(1 - P[IsThere]) + P(IsThere)(1 - P[Found|IsThere])}$$

The figure 2.3 on the next page explains the the optimal path finding algorithm diagrammatically, and the figure 2.4 explains the possible event when a cell in the grid is explored.

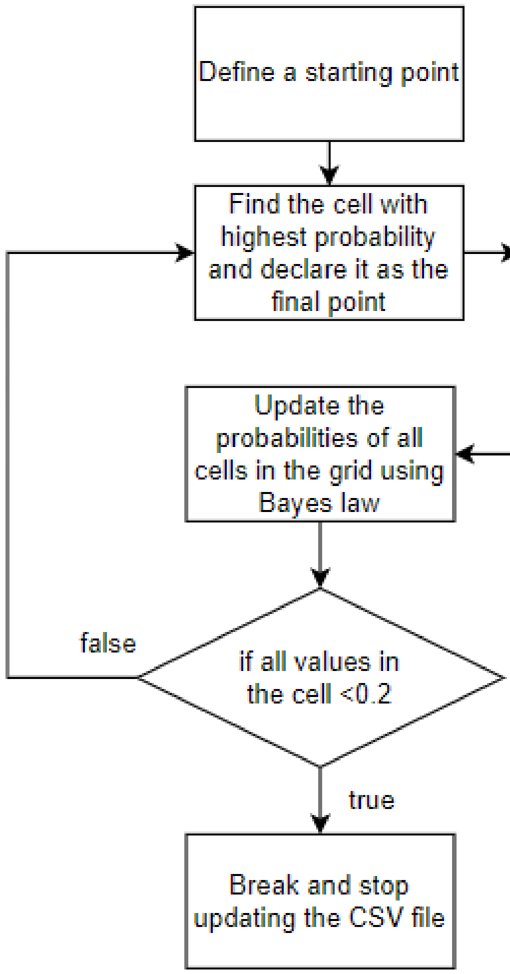


Figure 2.3: Flowchart of the algorithm for finding the optimal path in the given distribution of probabilities.

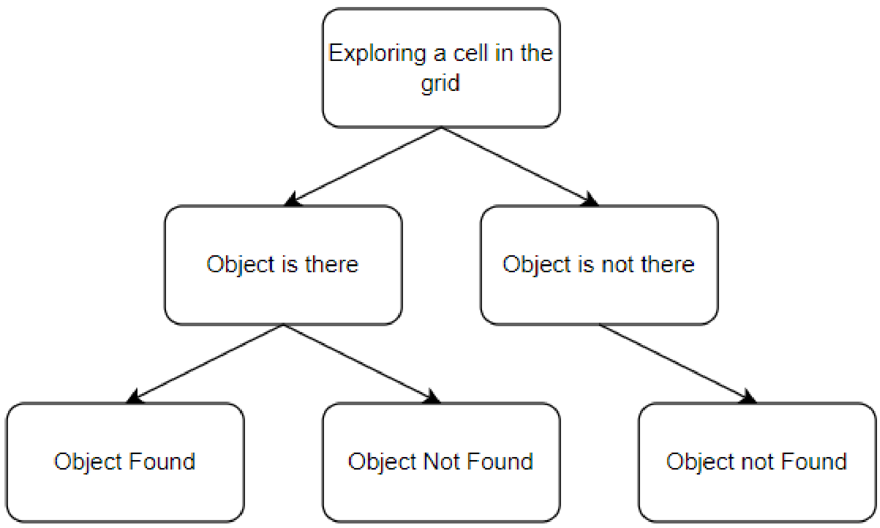


Figure 2.4: Possible events when a cell in the grid (a particular location in ocean/sea) is explored.

# Chapter 3

# Project Plan

Our project aims to develop a comprehensive system for analyzing oceanic data, incorporating various functionalities and successfully predicting the final location of the object which is to be found. The project begins with a focus on understanding algorithms crucial for data processing and analysis. This foundational step sets the stage for subsequent development.

We then proceed to develop robust updation logic, ensuring that our system can adapt to new data seamlessly. Leveraging the powerful pygame library, we create a map that provides users with a visually immersive experience. To populate these maps with accurate depth information, we employ Selenium to extract bathymetric data from GEBCO.

Taking a step further, we embed the pygame-based map within a web browser using pygbag. Utilizing APIs, we integrate real-time geographical data such as wind and ocean velocity, which provides our system with live information. To streamline user interaction, we develop a user-friendly website with an authentication system, ensuring data security and user privacy. The integration of our dynamic maps within the website enhances its functionality, providing users with a centralized platform for data analysis and visualization.

Incorporating statistical analysis techniques, we create Gaussian distributions to model various oceanic phenomena. We implement algorithms for calculating and plotting reverse drift trajectories and their distributions, facilitating deeper insights into oceanic patterns.

Finally, we design a mechanism for users to download processed data in CSV format. Through these comprehensive functionalities, our project aims to provide researchers and oceanic enthusiasts with a powerful tool for understanding and exploring the complexities of the ocean environment.

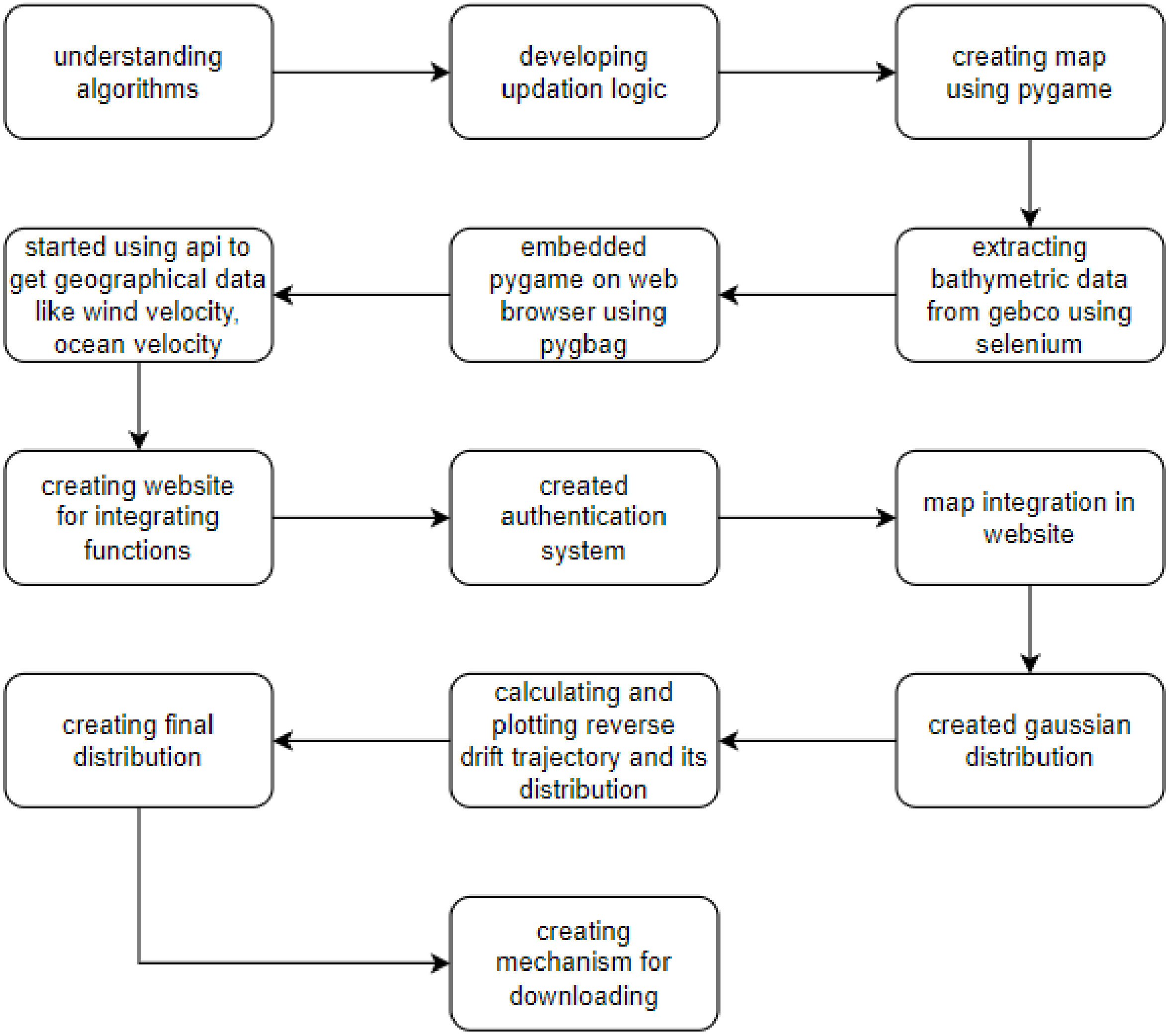


Figure 3.1: Flow of development

# Chapter 4

# Implementation

## 4.1 Final Website

The website is named **'WAVE'**, which is an acronym for **'Wayfinder for Aviation and Vessel Exploration'**. Throughout the development process, special attention was paid to creating a user-friendly interface that facilitates seamless navigation and data input. Recognizing the critical nature of search and rescue operations, the interface was meticulously crafted to be intuitive.

The figures on the following page display the final format of the website, embodying the culmination of our efforts to integrate all discussed functionalities into a web application.

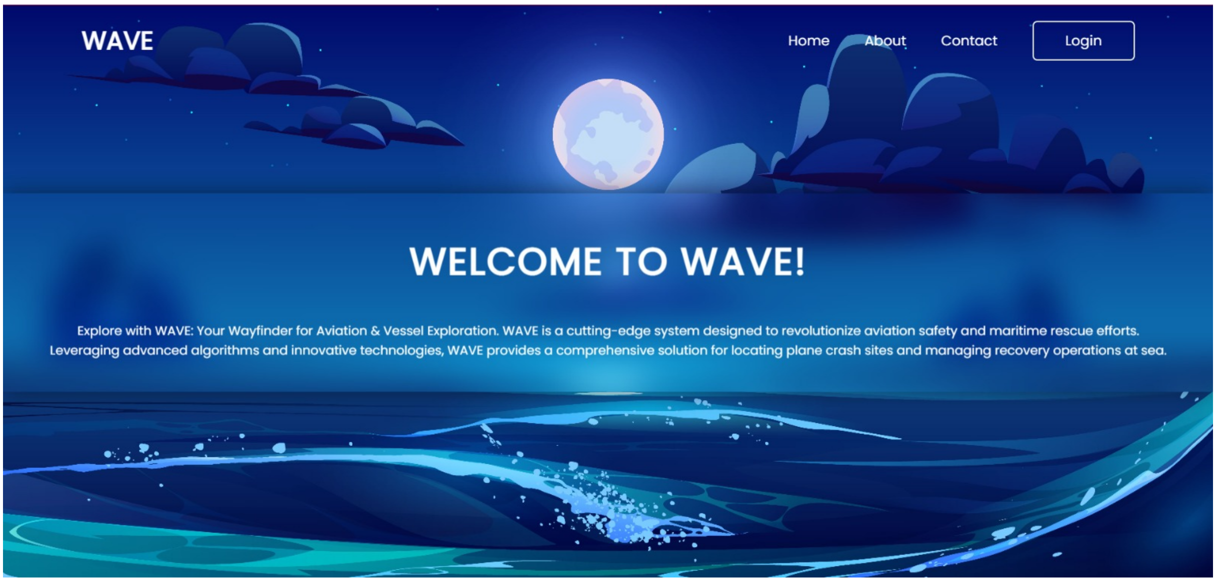


Figure 4.1: Home Page

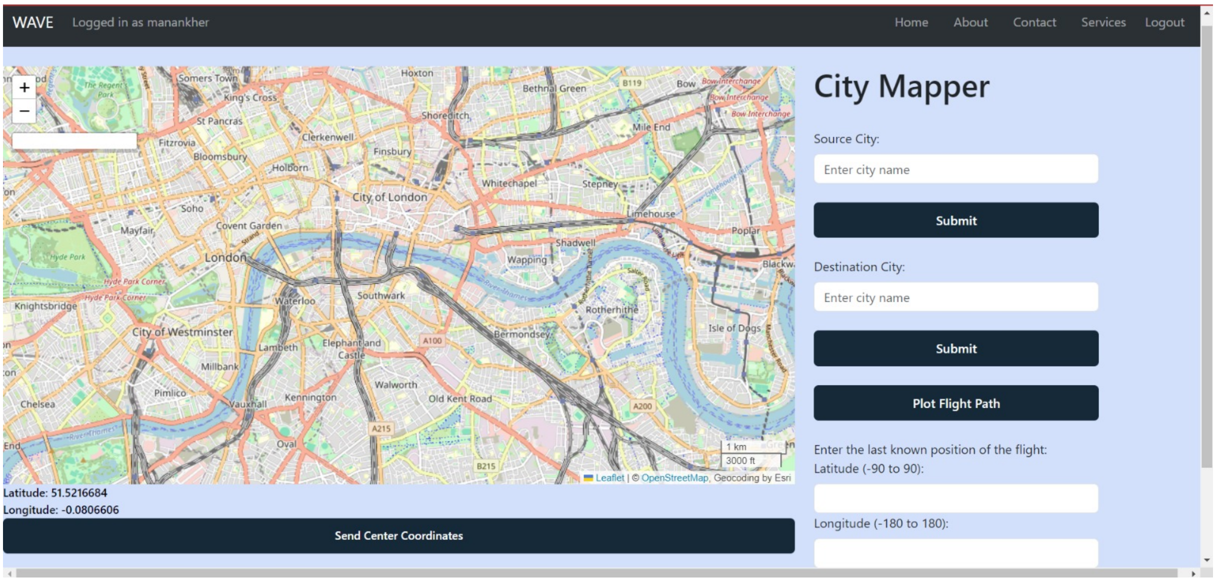


Figure 4.2: Services Page

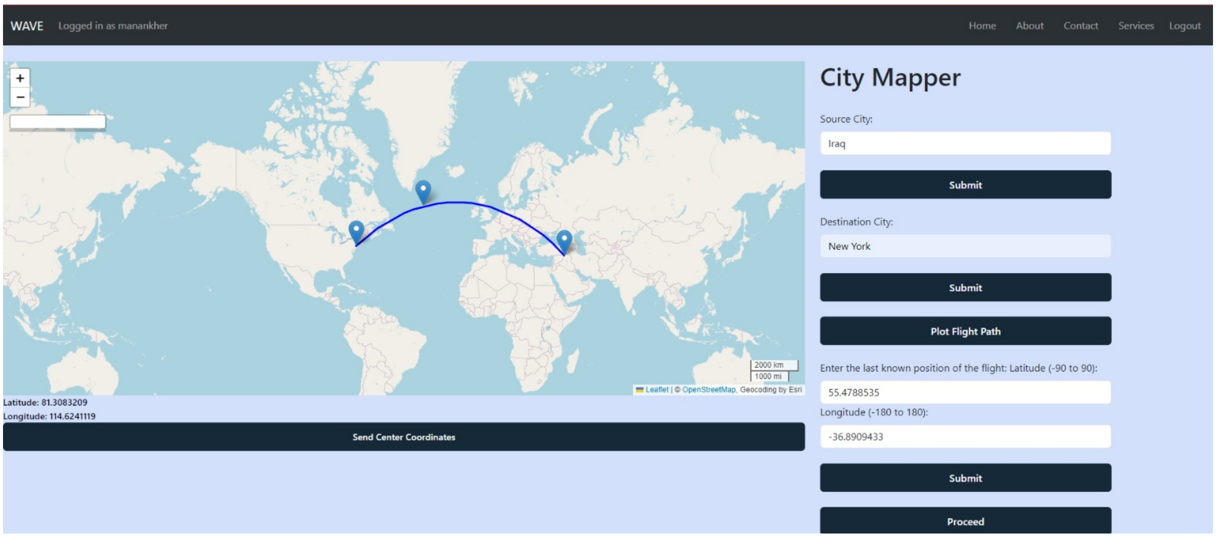


Figure 4.3: Services page after entering data

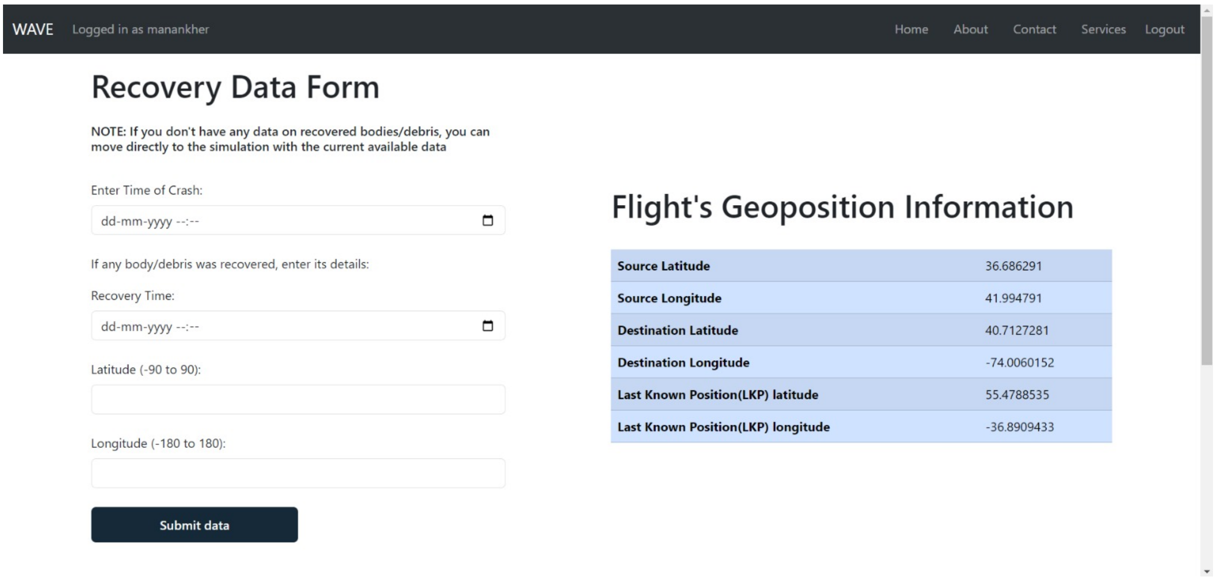


Figure 4.4: Reporting recovery of any debris

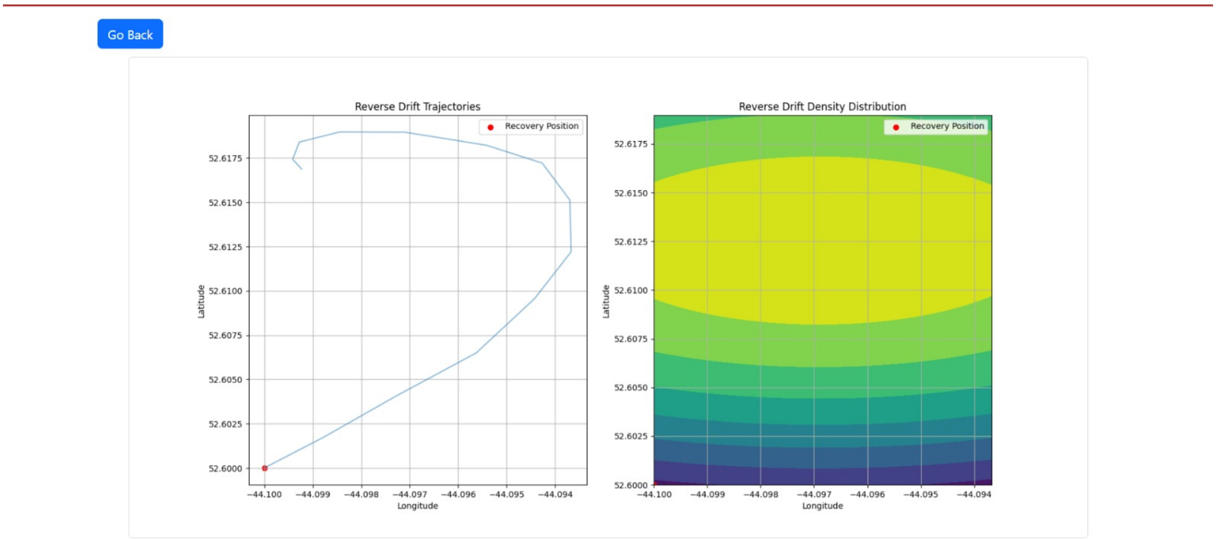


Figure 4.5: Plotting of Reverse Drift Trajectory

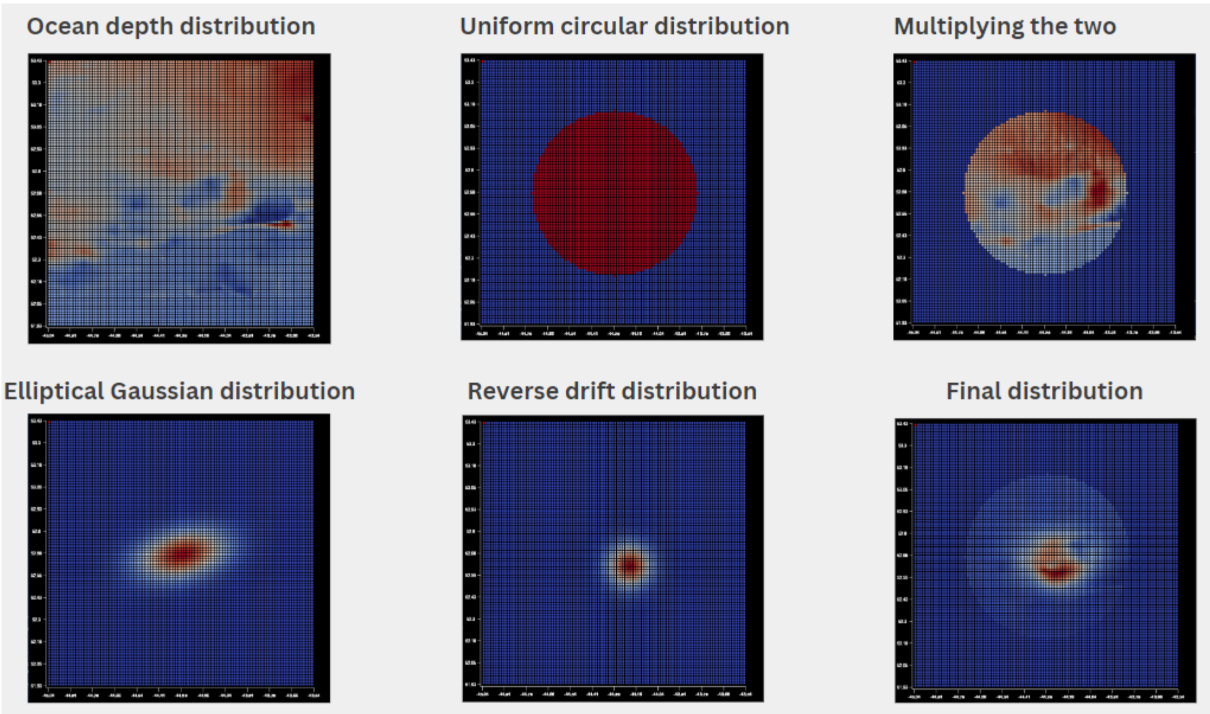


Figure 4.6: Different distributions used for getting final distribution

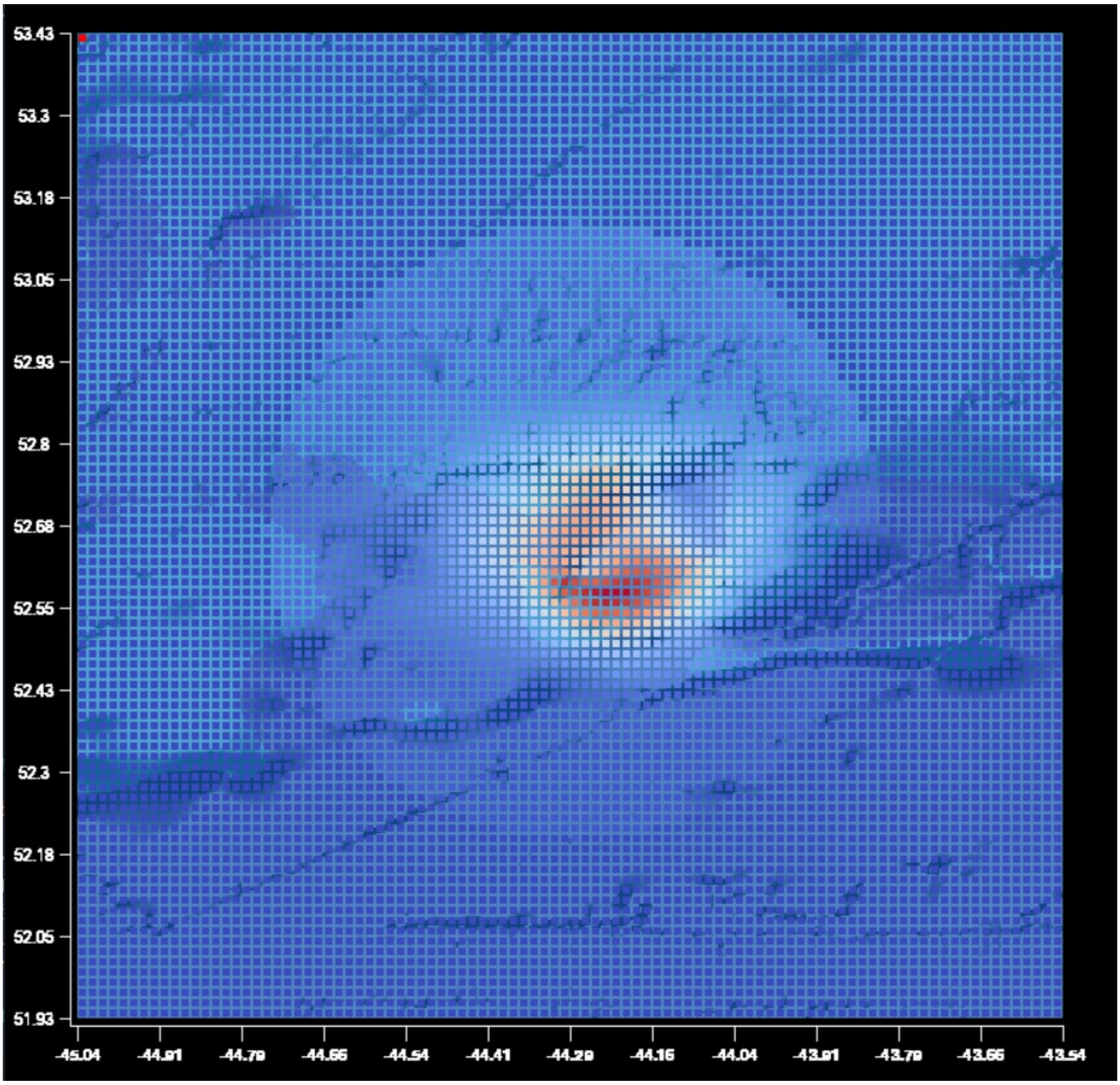


Figure 4.7: Final Distribution

## 4.2 Tech Stack

### Backend: Django, SQLite

Django serves as backend framework; SQLite serves as database engine. Django provides an authentication system along with an admin page for routine administrative tasks. Django's ORM (Object Relational Model) provides seamless interaction with SQLite database, performing storage and retrieval of data.

### Frontend: HTML, CSS, JavaScript, Leaflet JS, Turf JS

Leaflet JS is used for displaying interactive map on webpage along with markers for key locations and drawing flight routes. Turf JS performs geo-spatial analysis and helps providing accurate flight route taking earth's curvature into account.

### Python Libraries: NumPy, Pandas, Matplotlib, Scikit-Learn

NumPy, Pandas and Matplotlib is used for data analysis and visualization. Scikit-Learn provides KDE model for performing unsupervised learning on predicted reverse drift trajectory.

### GUI: Pygame, Pygbag

Pygame is used for attractive simulation and visualization of the search path and probability distribution. Pygbag is used for embedding the pygame GUI on webpage.

### Automation: Selenium, Subprocess

Selenium is used for scraping bathymetric data from GEBCO website. Subprocess automates the task of dynamically building APK file from the pygame simulation so that it can be embedded in the web browser.

## 4.3 APIs Used

### Meteomatics

To obtain both real-time and historical data on ocean current speed, direction, wind current speed, and wind current direction, after specifying the latitude, longitude, and time parameters.

### Geocoding

To obtain latitude and longitude of a city.

# Chapter 5

# Conclusion and Future Scope

## 5.1 Conclusion

The project leverages cutting-edge technologies such as Bayesian search theory, A* search algorithm, and machine learning models to optimize search paths and provide accurate probabilistic estimations. The automation capabilities powered by Selenium streamline the process of obtaining critical data, while the simulation component allows users to visualize search strategies and probability distributions in real-time.

Through extensive data analysis and simulation results, we have demonstrated the effectiveness and reliability of our product in assisting search and rescue teams in locating plane crash sites and recovering victims.

## 5.2 Future Scope

- Improve flexibility of our product by exploring larger search areas, for longer time durations, and implement the feature of adding previous search data as input to the algorithm. Expanding the scope of the project to other scenarios.
- Enable collaboration among users by allowing them to share data, insights, and search strategies. Implementing community features such as discussion forums or user-generated content.
- Commercialization by offering the platform as a service to government agencies, aviation companies, or search and rescue organizations. Additionally, forming partnerships with relevant stakeholders.
- Explore advanced analytics techniques such as predictive modelling, anomaly detection, or spatial analysis to further refine the search algorithm.

# References


1. Stone, L. D., Keller, C., Kratzke, T. L., Strumpfer, J. (20 January, 2011). Search Analysis for the Location of the AF447 Underwater Wreckage. Report to Bureau d'Enquêtes et d'Analyses.

2. Stone, L. D., Keller, C. M., Kratzke, T. L., Strumpfer, J. P. (19 May, 2014). Search for the Wreckage of Air France Flight AF 447.