

公立はこだて未来大学 2023 年度 システム情報科学実習
グループ報告書

Future University Hakodate 2023 Systems Information Science Practice

Group Report

プロジェクト名

スマート水産業

Project Name

Smart Fisheries

グループ名

琵琶湖チーム

Group Name

Team Lake Biwa

プロジェクト番号/Project No.

6-琵琶湖チーム

プロジェクトリーダー/Project Leader

倉田颯太 Kurata Souta

グループリーダー/Group Leader

関崎証 Sekizaki Akashi

グループメンバ/Group Member

関崎証 Sekizaki Akashi

椋島司遠 Kabashima Shion

横田凌也 Yokota Ryoya

松本葉太 Mastumoto Yota

古明地颯天 Komeji Sota

指導教員

和田雅昭 長崎健 大澤英一

Advisor

Wada Masaaki Nagasaki Takeshi Osawa Ei-Ichi

提出日

2024 年 1 月 17 日

Date of Submission

January 17, 2024

概要

水産業は日本の重要な産業の一つであり、政府はスマート漁業を推進している。しかし、水産業には ICT を活用できる場面がまだ多くあると考える。例えば、漁業者の働きや経験を可視化することが挙げられる。他にも、漁獲量報告を効率化できると考える。本プロジェクトでは、琵琶湖の漁業の見える化を試みる。

関 [1] によると、琵琶湖では昔ながらの漁法が継承されてきた。しかし、近年の漁業者の高齢化や後継者不足が問題になっている。そのため、技術継承の危機に瀕している。そこで、本プロジェクトでは、従来は熟練漁業者の働きに頼って行われていた漁業の知識と経験を可視化する。そして、それらの技術を次世代に継承することを目的として、データを分かりやすく伝える Web アプリケーションの作成を行う。

前期では、データベースの構築、航跡の可視化、漁場の面積の計算の一部、Web アプリケーションのモックアップの作成を行った。

後期では、航跡、漁場の面積、漁獲量を見ることのできる Web アプリケーション「湖マップ」を完成させた。そして、実際に琵琶湖に伺い、ヒアリングを実施した。その結果、アプリケーションの使用感や、改善点、漁業に対する貴重な知見を得ることができた。

キーワード 水産業, IoT, 可視化, 琵琶湖

(※文責: 関崎証)

Abstract

The fisheries industry is one of the important industries in Japan, and the government has promoted the smart fisheries. However, there are still many situations where ICT can be utilized. The first is to visualize fishermen's intuition and experience; the second is to automate data entry. In this project, we try to visualize the fishing industry in Lake Biwa.

Seki[1] reported that traditional fishing methods have operated in Lake Biwa. However, in recent years, the aging of fishermen and the lack of successors have become a problem. As a result, the opportunities for skills transfer are decreasing. Our team will visualize the knowledge and experience of the skilled fishermen, which have traditionally relied on their intuition. Therefore, we aimed to hand over these skills to the next generation and we will create a Web application which can visualize the data in the easy-to-understand way.

In the first semester, we built the database, visualized the wake, calculated some of the area of the fishing grounds, and created a mock-up of the Web application.

In the second semester, we have created a Web application which called "Umi Map". The application can visualize wake, fishing area, and catch. Then, we actually visited Lake Biwa to conduct an interview. As a result, we were able to get the feedbacks about the application, points for improvement, and valuable insights about fisheries.

Keyword Fisheries industry, IoT, Visualization, Lake Biwa

(※文責: 関崎証)

目次

第 1 章	背景	1
1.1	琵琶湖の現状	1
1.2	琵琶湖で行われているスマート化の先行事例	1
1.2.1	漁獲報告 WEB アプリ「湖（うみ）レコ」	1
1.3	湖（うみ）レコを踏まえた提案	3
1.4	プロダクトの概要	3
1.5	新規性	4
第 2 章	到達目標	5
2.1	本プロジェクトにおける目的	5
2.2	通常の授業ではなく、プロジェクト学習で行う利点	5
2.3	具体的な手順・課題設定	5
第 3 章	前期活動内容	7
3.1	活動目標	7
3.2	琵琶湖の漁業の現状	8
3.2.1	琵琶湖の漁業についての情報収集	8
3.3	プロダクトの構想	8
3.3.1	機能の構想	8
3.3.2	具体的な機能	9
3.3.3	システムの構成	10
3.4	役割分担	10
3.4.1	フロントエンド班	10
3.4.2	バックエンド班	10
3.4.3	データベース班	11
3.5	前期の成果	12
3.5.1	フロントエンド班	12
3.5.2	バックエンド班	12
3.5.3	データベース班	13
3.6	中間発表のまとめ	13
3.6.1	発表内容	13
3.6.2	いただいた意見・質問・回答	14
3.7	後期に向けての課題	15
第 4 章	前期結果	16
4.1	前期プロジェクトの結果	16
4.2	成果の評価	16
4.2.1	データベースの構築	16

4.2.2	天気を表示する機能	17
4.2.3	航跡のプロットを行う機能	17
4.2.4	面積を求める機能	17
4.2.5	Web アプリケーションの作成	17
4.3	担当分担課題の評価	18
4.3.1	関崎証	18
4.3.2	椛島司遠	18
4.3.3	古明地颯天	18
4.3.4	松本葉太	19
4.3.5	横田凌也	19
第 5 章	後期活動内容	20
5.1	活動目標	20
5.2	役割分担	21
5.2.1	フロントエンド班	21
5.2.2	バックエンド班	21
5.2.3	データベース班	25
5.3	後期の成果	25
5.3.1	フロントエンド班	25
5.3.2	バックエンド班	29
5.3.3	データベース班	31
5.4	琵琶湖ヒアリング結果	32
5.4.1	スケジュール	32
5.4.2	準備	32
5.4.3	行ったこと	32
5.4.4	結果	32
5.5	成果発表会まとめ	34
5.5.1	発表内容	34
5.5.2	いただいた意見・質問・回答	34
5.6	今後の展望	36
5.6.1	面積の計算精度の向上	36
5.6.2	UI や UX の向上	36
5.6.3	Web アプリケーションの表示機能の追加	36
第 6 章	後期結果	37
6.1	後期プロジェクトの結果	37
6.2	成果の評価	38
6.2.1	Web アプリケーションの作成	38
6.2.2	漁場の面積計算	38
6.2.3	Web アプリケーションをホストする作業	39
6.2.4	データベースの構築	39
6.3	担当分担課題の評価	39
6.3.1	関崎証	39

6.3.2	梶島司遠	40
6.3.3	古明地颯天	40
6.3.4	松本葉太	41
6.3.5	横田凌也	41
付録 A	新規習得技術	42
付録 B	活用した講義	43
参考文献		44

第 1 章 背景

1.1 琵琶湖の現状

琵琶湖の面積は約 670 平方キロメートルであり、日本で最も広い湖として知られている。また関 [1] は、琵琶湖について次のようにまとめている。琵琶湖には、約 50 種類の魚が生息し、縄文時代から漁業が営まれていたと考えられている。現在でも沖びき網漁業や貝びき網漁業等、多彩な漁業が営まれており、古くから漁労技術が暗黙知として受け継がれている。しかし、近年の琵琶湖漁業は、漁業者の高齢化と後継者不足が深刻な課題となっている。そのため、滋賀県では 2016 年度に漁業研修制度を導入し、新規就業者の確保と育成に取り組んでいるが、新規就業者の参入が漁業者の高齢化に追いついておらず、技術継承の機会喪失が危ぶまれている。

1.2 琵琶湖で行われているスマート化の先行事例

1.2.1 漁獲報告 WEB アプリ「湖（うみ）レコ」

滋賀県 [2] は、図 1.1 が示すように、全国に先駆け漁業許可等を所有する全ての漁業者がスマートフォンなどを用いて漁獲情報を報告できるシステム「湖（うみ）レコ」を開発し、運用している。

湖（うみ）レコという愛称は、琵琶湖の漁業者が琵琶湖のことを「うみ」と言うこと、また、漁業者が操業を通じて、水産業から見た今の琵琶湖を記録（record）し、未来につなげたいとの思いが込められている。



図 1.1 湖（うみ）レコ [2]

背景

滋賀県 [2] が湖（うみ）レコを開発した背景として、従来は漁獲情報の報告が紙媒体で行われていたことがある。これは、漁業者にとって大きな負担となっているだけでなく、集計や報告内容のデータ化にも時間を要している。漁業者の負担の軽減を図りつつ、漁獲報告の履行と資源評価などに必要なデータ収集を一元的に可能とするシステムが必要とされた。さらに、2020 年に施行された改正漁業法により、知事許可漁業等における資源管理の状況等の報告が義務化された。この資源管理は、資源水準を維持・回復させ、中長期的に漁獲できる量を増やし、漁業者の所得を向上させ

ることを目的としている。

システム概要

1. 名称：滋賀県漁獲報告システム 「湖（うみ）レコ」
2. 形態：WEB アプリケーション (<https://umirec.com>)
3. 事業主体：滋賀県
4. 利用対象：漁業許可数 約 1,500
5. 対象漁業者数：約 520 名
6. 報告可能事項
 - 操業日
 - 漁法とその規模（網の数や操業時刻等）
 - 魚種毎の漁獲量
 - 操業場所（マップ上で指定）
7. 漁業者が出来る主な機能
 - 漁獲情報をスマホ等に入力することで、県に迅速な報告ができる
 - 自身が報告した漁獲情報を集計、分析、可視化できる
 - 県からのお知らせや所有する漁業許可を確認できる

操作の流れ

1. 操業日の入力
 - 「操業日」という項目を押すとカレンダーが表示される
 - 操業日を選択する
2. 漁業種類の入力
 - 「漁業種類」という項目を押すと許可のある漁業種類の選択肢が表示される
 - 該当する漁業種類を選択する
3. 操業時刻等の入力
 - 漁業種類に応じて表示される回数を入力
 - 操業時刻を入力
4. 魚種と漁獲量の入力
 - 漁業種類を選択すると魚種の項目が表示される
 - 該当する魚種を選択する
 - 漁獲量をキログラム単位で入力
5. 操業場所の入力
 - 該当する操業場所「琵琶湖・内湖」もしくは「河川」のいずれかを選択する
 - 操業している湖上で「現在地を取得する」を押すと現在地が取得される
 - 5箇所まで操業場所が選択できる
6. 報告の送信

導入によるメリット

滋賀県 [2] は、湖（うみ）レコを導入することによるメリットは4つあるとしている。1つ目に、湖（うみ）レコで報告した場合、用紙による報告が必要なくなる。2つ目に、報告した情報がシステムに蓄積されるため、過去の漁獲量の閲覧が可能になる。3つ目に、分析画面により、漁獲量の推移を漁法別、魚種別などで絞り込んでグラフ化することや、一覧表示することができる。また、前年比も表示することもできる。4つ目に、漁船漁業では、操業場所の情報から、どの水域でよく漁獲したかを地図上で確認することができるようになる。

今後の活用

滋賀県 [2] では、水産試験場などの研究機関による資源調査と合わせて、漁獲情報を迅速に収集することにより、資源水準（資源が多いか少ないか）と漁獲の強さ（獲りすぎか否か）を評価し、目標とする資源水準に向けて資源管理型漁業の実践を行う。そして、これからの資源管理は漁獲を抑える、我慢するだけではなく、資源に余裕のある魚種は、積極的に漁獲し、漁家所得に変えていくことにも重点を置くとしている。また、今回のシステム開発は、漁獲量だけではなく、操業場所や操業の規模も同時に収集できることから、精度の高い資源評価が迅速に可能となるものとして期待できる。漁業者が減少するなか、これまで取り組んできた環境保全や稚魚の放流などに加え、資源管理の強化を図り、少数でも精鋭な儲かる水産業の実現を目指すとしている。

1.3 湖（うみ）レコを踏まえた提案

和田教授が現在取得しているデータを活用することで、より正確な資源管理を行うことができると考える。

1.2.1 漁獲報告 WEB アプリ「湖（うみ）レコ」の操作の流れで示したように、報告までの6ステップの内、すべてのステップにおいて情報の入力か選択を行うことになる。具体的には、操業日、漁法とその規模（網の数や操業時刻等）、魚種毎の漁獲量、操業場所（マップ上で指定）を報告することになっている。

ここで、和田教授が収集している日付、時刻（UTC）、漁船 id、緯度、経度、航行速度、漁船の進路方向の情報を活用することを提案する。これにより、従来手動で入力していた操業場所や規模の情報をより正確に記録できると考える。また、その記録した正確な情報を活用することで、後継者育成に活用できると考える。

1.4 プロダクトの概要

本プロジェクトでは、従来は熟練漁業者の勘に頼って行われていた漁業に関する知識と経験を可視化し、次世代に伝えることを目的とする。

そのために、漁業の見える化を行い、それを分かりやすく漁業者に伝え共有できる Web アプリケーションの作成を行う。Web アプリケーションでは、自動取得されるデータから操業日時、航跡、漁場の面積を求める機能を実装する。また、天気予報と漁獲量も外部サイトから取得し、表示することを目指す。

1.5 新規性

湖（うみ）レコでは、操業日、漁業種類（沖びき網漁業など）、漁が行われた回数、操業時刻、魚種、漁獲量、漁場の7つすべての情報を漁業者に手入力させるシステムである。しかし先に述べたように、和田教授の収集しているデータを活用することで、湖（うみ）レコよりも正確にデータを収集することができると思う。湖（うみ）レコにより、従来の紙媒体での報告に比べ漁業関係者の負担が減少したと考える。だが、和田教授の収集しているデータを活用することで、さらに漁業関係者への負担をさらに減らすことができると思う。

本プロジェクトの新規性は4つある。1つ目に、湖（うみ）レコで手入力が必要だった7つの情報のうち、操業日、漁が行われた回数、操業場所の3つの情報を漁船から自動で取得することができる。2つ目に、従来収集できないなかったすべての航跡などの情報を収集することができる。3つ目に、取得データを分析し、漁場の面積を求めることができる。4つ目にそれらの結果を漁業者間で共有することができる。

（※文責: 関崎証）

第2章 到達目標

2.1 本プロジェクトにおける目的

本グループでは、琵琶湖の漁業に関するデータを自動的に収集することで漁業者の手間を減らし、収集したデータを見える化することで、次世代の漁業者に熟練漁業者の技術を受け継ぐことを目的とした。米川 [3] によると、見える化とは企業や組織の活動を具体的にして目で客観的に捉えられるようにすることを指す。つまり、収集したデータを直接グラフや表にするだけでなく、収集したデータをもとにグループで分析し考察することで、熟練漁業者や次世代の漁業者に効率的な漁が行えるよう提案できるようにする。

(※文責: 古明地颯天)

2.2 通常の授業ではなく、プロジェクト学習で行う利点

プロジェクト学習で行う利点について、複数人で活動することに様々なメリットがあることがあげられる。1つ目は、複数人で行う作業の進め方について学ぶことができ、これは通常の授業や卒業研究では基本的に個人で活動するため、差別化できる。例えば、複数人での活動を通して同じ環境を構築する方法について学ぶことができる。2つ目は、人数がいることにより役割分担ができ、一つの作業に集中することや、一人で活動するよりも作業量を増やすことができる。同じバックエンド班でも、複数人いることにより全く別の方法をそれぞれ試し、互いの結果を比較することができる。3つ目は、通常授業よりも、長期間そして自由活動できることにある。授業時間外に時間を取り活動でき、夏季休暇を有効的に使うことができる。

(※文責: 松本葉太)

2.3 具体的な手順・課題設定

琵琶湖の漁業のスマート化を行うために以下の手順で課題設定を行った。

1. 琵琶湖の現状に関する調査

課題：インターネットを活用し、琵琶湖の漁業についての情報収集を行う。

担当：関崎証、椛島司遠、横田凌也、松本葉太、古明地颯天

2. 琵琶湖の現役漁業者への質問

課題：作成した質問事項をもとにインタビューを行う。

担当：関崎証、椛島司遠、横田凌也、松本葉太、古明地颯天

3. プロダクトの構想

課題：どのようなプロダクトが求められているのか・使いやすいのかを検討する。

担当：関崎証、椛島司遠、横田凌也、松本葉太、古明地颯天

4. 使用するプログラミング言語・技術の検討

課題：様々なプログラミング言語があるなか、どのプログラミング言語や技術を採用するかを検討する。

担当：関崎証、椛島司遠、横田凌也、松本葉太、古明地颯天

5. データベース構築

課題：今回はデータを扱うシステムであり毎分データの読み書きが行われることを想定した。そのためどのようなシステムであれば構築が簡単になるかを検討した。

担当：椛島司遠

6. 分析手法の検討と実装

課題：Python を用いて、漁船の航跡から漁場の面積を求める。

担当：松本葉太、古明地颯天

7. UI の検討と実装

課題：Figma を用いて UI をデザインする。その後、Vue.js を用いて Web アプリケーションを作成する。

担当：関崎証、横田凌也

8. 統合テスト

課題：上記で実装したものを統合し、テストする。

担当：関崎証、椛島司遠、横田凌也、松本葉太、古明地颯天

9. 実施テスト

課題：実装したアプリケーションを琵琶湖漁業者に体験してもらい、意見をもらうことでどのように修正していくか検討する。

担当：関崎証、椛島司遠、横田凌也、松本葉太、古明地颯天

10. 修正・完成

課題：琵琶湖漁業者の意見をもとに、改良を行い完成とする。

担当：関崎証、椛島司遠、横田凌也、松本葉太、古明地颯天

(※文責: 関崎証)

第 3 章 前期活動内容

3.1 活動目標

前期の活動目標については以下に示す。

- プロジェクト目標の設定、方針を決定
課題: プロジェクト開始時に指定されたテーマについてプロジェクトメンバー全員で話し合いを行い、プロジェクト目標を決定し、2つのチームに分割する。その後、各チームごとにグループ課題を決定する。
- 琵琶湖に関する情報収集
課題: 琵琶湖の漁業に関する知識が少ないため、グループ課題を具体的に決定することは難しい。そのため、琵琶湖の漁業をスマート化するために必要な情報を調べ、どのようにスマート化するかを決定する。
- データベースの構築
課題: 和田教授が収集している日付、時刻 (UTC)、船の識別番号、緯度、経度、航行速度、漁船の進路方向のデータを格納しているデータベースから我々のグループのデータベースにデータを格納する。我々のデータベースでデータを取得し、時刻を JST に変換する等の前処理を行う。データを我々のデータベースに格納する。データベースを自動更新する処理を行う。
- 天気を表示する機能
課題: 複数のサイトからデータを取得する。得られた天気データを csv 形式で保存する。
- 航跡のプロットを行う機能
課題: 我々のデータベースからデータを取得する。航跡を可視化する。
- 面積を求める機能
課題: 我々のデータベースからデータを取得する。面積を求める機能を実装する。
- 漁獲量を出力する機能
課題: 滋賀県に依頼を行う。そして、湖 (うみ) レコで得ている漁獲量データを取得し、出力する機能を実装する。
- Web アプリケーションの作成
課題: Web アプリケーションの UI をデザインするために、Figma を使用し、モックアップを作成する。その後、Vue.js を用いて Web アプリケーションを作成する。

(※文責: 横田凌也)

3.2 琵琶湖の漁業の現状

3.2.1 琵琶湖の漁業についての情報収集

滋賀県 [2] によると琵琶湖では現在 6 種類の漁法が行われている。1 つ目は、小型定置網（えり）漁業で、湖岸から沖合に向かい矢印型に網を張り、湖岸によってきた魚の習性をうまく利用し、「つぼ」と呼ばれる部分に誘導し閉じこめて漁獲する方法である。主な漁獲物はアユ、フナ、ホンモロコで、主に 8 月中旬から 11 月中旬を除くほぼ周年営まれている。2 つ目は、沖びき網漁業で、漁船を使用して水中に網を入れて、これを引き寄せて漁獲する方法である。主な漁獲物はモロコ、ヨシノボリ、エビ、イサザ、アユで、主に 5 月から 7 月中旬を除くほぼ周年営まれている。3 つ目は、追いさで網漁業で、春になり岸辺付近で群をつくっているアユを、カラス等の羽を付けた棒で、さで網を持つ受け手の方へ追い込み漁獲する方法である。主な漁獲物はアユで、主に 3 月から 6 月頃の期間で営まれている。4 つ目は、刺網漁業（小糸網漁業）で、水中にカーテンのように張った網で漁獲する方法である。主な漁獲物はアユ、フナ、ホンモロコ、ビワマスで、周年営まれている。5 つ目は、貝びき網漁業で、マングワと呼ばれる漁具を湖底に沈め、漁船を曳いてシジミ等を採る漁法である。主な漁獲物はシジミで、主に 5 月から 7 月を除く周年営まれている。6 つ目は、沖すくい網漁業で、湖面で大きな群れ（マキ）になっているアユを漁船の先に付けた網ですくい取る漁法である。主な漁獲物はアユで、主に 6 月初旬から 7 月下旬の期間で営まれている。漁獲量は、1955 年頃には 1 万トン前後あったが、2017 年には 713 トンまで減少した。漁獲量の減少の原因には、産卵繁殖の場となるヨシ群落や内湖の減少、外来魚やカワウの食害、水草の異常繁茂による漁場の悪化があげられる。さらに、漁業者の高齢化がすすみ、併せて後継者不足も深刻な要因としてあげられる。現在滋賀県は、ヨシ帯の造成や在来魚介類の放流、外来魚やカワウの駆除、漁場環境の改善などの取り組んでいる。さらに、2016 年度に漁業研修制度を導入し、新規就業者の確保と育成に取り組んでいる。さらに紙媒体で行っていた漁獲情報の報告が漁業者にとって大きな負担となり、データの集計にも時間を要していた。そのため、滋賀県は、適切な水産資源の管理を行い、資源水準を維持、回復させていくための Web アプリケーションである滋賀県漁獲報告システム「湖（うみ）レコ」を開発した。

（※文責: 横田凌也）

3.3 プロダクトの構想

3.3.1 機能の構想

私たちは琵琶湖の水産業の見える化によって漁業の支援するプロダクトを作ることとした。3 つの要件と 4 つの機能が案に出た。

要件について

1 つ目の要件は、複数種類のデバイスに対応できる UI を作ることである。出港する際に手元の携帯やタブレットで今欲しいデータを見れる状況と今までの漁の振り返りとして PC などから見る状況を想定した。このことからスマートフォン、タブレットおよび PC それぞれに対応した UI を備えることとした。2 つ目の要件は、その日の様々な情報を一覧できる画面を作ることである。単

に情報を列挙するだけでなく使いやすいような表示にしていくこととした。3つ目の要件は、必要な情報を収集し、表示することである。これは琵琶湖の漁業者と相談しながら必要な情報と見やすい方法を検討していく予定である。

機能について

1つ目の機能は、航跡をプロットして見れる機能である。これはより正確に漁についてどこで漁をしたか分析するのに必要な機能であると考えた。2つ目は、簡単に天気予報を確認できる機能である。漁をする際には天気が重要であると考え、幾つものサイトを横断してみる時間や手間を減らす目的で案が出た。また、この機能を応用し過去の航跡データと天気を見ることができるようになり、データの分析をするための要素の一つとする案も出た。3つ目は、漁をしたおおよその面積を出力する機能である。沖びき網漁業は囲むように魚を捕獲するため漁場を評価するためには漁獲量だけではなく漁場の面積も重要となるからである。4つ目は、漁獲量を出力する機能である。これはすでに運用されている湖（うみ）レコというアプリケーションにデータ共有のお願いをすることにより、こちらの Web アプリケーション側での漁獲量の入力を不要とする計画を立てた。

また、これらを組み合わせるとその日の航跡、天気、漁場の面積及び漁獲量を一覧し比較できる機能を実装する構想を立てた。

3.3.2 具体的な機能

航跡をプロットする機能について

現在想定している機能は船の識別番号ごとに色をつけて出力すること、識別番号ごとに航跡を出力する機能を想定している。またフロントエンド班の要求次第では出力する範囲を調整したものも想定している。

天気予報を確認できる機能について

天気予報もさまざまな種類があり、予報を出しているサイトも様々である。それらを一覧して出力できる機能を想定している

漁場の面積を出力する機能について

沖びき網漁業をする際には漁場で囲むように魚を捕獲する。その面積を求めるための機能を想定している。

漁獲量を出力する機能について

現在運用されている湖（うみ）レコというアプリケーションに入力しているデータの共有を依頼している段階である。どこまでの情報を共有していただけるかは前期の段階では不明だが、最低限船ごとの漁獲量を表示する機能を想定している。

3.3.3 システムの構成

我々は1つのサーバー上にデータベース、Webサーバーおよび情報を加工するプログラムを実行することとした。そのため図3.1のようになった。

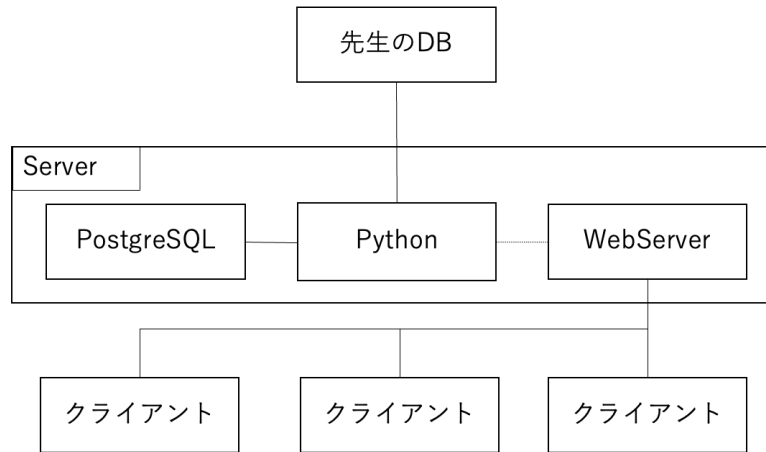


図 3.1 システム構成図

(※文責: 梶島司遠)

3.4 役割分担

3.4.1 フロントエンド班

Webアプリケーションを作成する。UIデザインを作成するためにコラボレーションインターフェースデザインツールであるFigmaを用いる。FigmaのプラグインであるTeleportHQを用いて、Figmaで作成したUIデザインをVue.jsで使用できるコードに変換する。変換したコードに画面遷移の実行や航跡のHTMLを埋め込む。

(※文責: 横田凌也)

3.4.2 バックエンド班

航跡の可視化と漁場の規模を調べる

漁船に取り付けたセンサーから得た情報をもとに、その日の航跡を漁船ごとに色分けしてマップ上に可視化する(図3.2)。センサーからは主に、日付、時刻(UTC)、漁船番号、緯度、経度、航行速度、進行方向が取得できる。前期ではまだデータベースと連携できていないため、数隻の漁船について可視化する。航跡を可視化するために、PythonのFoliumというライブラリを用いる。これはPythonでマップ上に何かプロットしようとしたときによく用いられるものであり、マップの生成にはOpen Street Mapを用いる。

また、可視化された航跡の中から扇形の漁場を見つけ出し、その面積を求めることがバックエンド班の目標である。これは、扇形の中にある楕円に沿って網を引いているため、扇形の漁場の面積

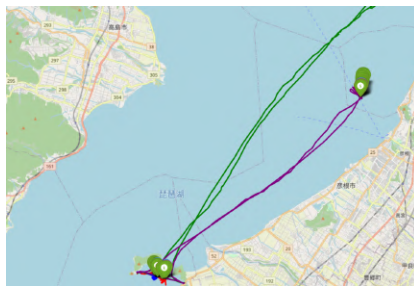


図 3.2 航跡全体図

がそのままその日の網を引いた広さとなり、この情報が知りたいと漁業者から要望があった。まずは、扇形の中心点となる緯度経度を探し出す。手法は、扇形の中心は何度も船が通るため一定の範囲内に航跡が集まる特徴を利用する。時間が少し経ち中心点から離れた地点を漁が始まった始点とする。その後、再び中心点との距離が近くなった地点を終点とする。こうして始点と終点の時刻を判断する。つまり、始点と終点の時刻で挟まれた緯度経度データを式 3.1 に代入すると楕円型の漁場の面積が求められる。マップ上では、式 3.1 に代入した緯度経度データは図 3.3 である。

$$\frac{1}{2} \left| \sum_{j=1}^n (x_j - x_{j+1}) * (y_j + y_{j+1}) \right| \quad (\text{式 3.1})$$

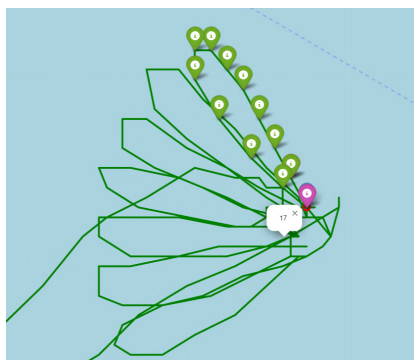


図 3.3 楕円型の漁場

気象データの取得

Python の Requests を使い、スクレイピングで気象情報を取得する。取得する気象データの時刻は今後、参照する航跡データの時刻と合わせる。

(※文責: 松本葉太)

3.4.3 データベース班

大量のデータを扱うためプログラムで扱いやすいデータベースを利用する。また、授業で使った PostgreSQL を用いる。インターネット上にある随時更新される csv データを我々のデータベースに差分として反映するため、日時を UTC から JST に変換するために Python を用いる。

(※文責: 椛島司遠)

3.5 前期の成果

3.5.1 フロントエンド班



図 3.4 Web アプリケーションのデザイン

Figma を用いて図 3.4 のような UI デザインのモックアップが完成した。Web アプリケーションの画面遷移の実行までは完了した。

(※文責: 横田凌也)

3.5.2 バックエンド班

面積を求めた結果と航行速度の特徴

3.4.2 の手法で、一部の面積を求めることに成功はしたが、その値が正しいかどうかは疑問が残る結果となった。例として、地球は大きく見れば球体の形をしているため、二次元平面で使われることが想定された式では正確に面積を求められていないのではないかとこの疑問がある。これについて、琵琶湖という地球規模で見れば小さな領域であれば無視できる誤差となるのかについて調べる必要がある。他にも一定の範囲内を設定するといっても、いまだ特に決まった値はなくデータが変わればこの値も変えざるを得ないため、値の設定規則を見分ける必要がある。今の段階では、漁船が漁港の近くから遠くへ行くにつれ、漁場の規模も大きくなり、航跡も比較的単純になることから、設定する値を大きくするべきだと予想した。

また、別のアプローチで漁場を特定しようとした結果、図 3.5 のような漁船の航行速度について注目すると漁をしている時だけ特徴的なグラフを描くことが分かった。そしてその特徴は、楕円型の漁場と数が一致していることが分かった。つまり、楕円型の漁場の面積の情報と合わせることで、漁場全体の面積を大まかに求めることができる。今後はこの特徴をコンピューターに判定させることが課題となる。他にも、漁港から遠くの大きな漁場ほど漁船の全体的な航行速度が速いことがわかった。楕円型の漁場の数の特定以外にも、扇形の漁場がどこにあるかなどの判断材料になる可能性もある。

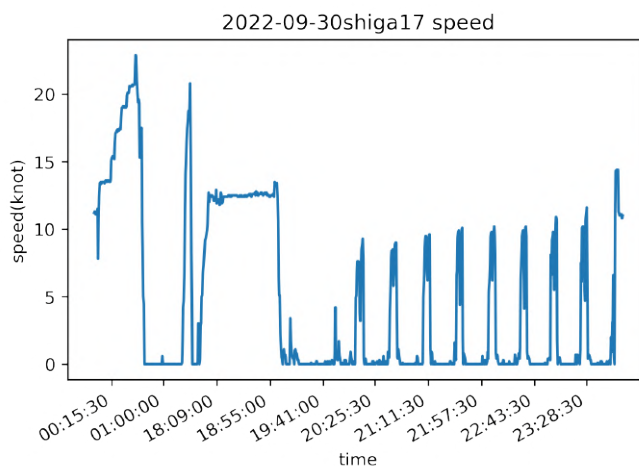


図 3.5 時間と航行速度の関係

気象データの取得

tenki.jp から、日付、時刻、気温、降水量、風速、風向きを取得することに成功した。

(※文責: 松本葉太)

3.5.3 データベース班

データベース班はデータベースの構築および運用を担当した。テストサーバーにテスト用データベースを構築および運用した。また、データベースはデータベース工学の講義で使用した PostgreSQL を使用した。データベースで扱うデータのテーブルを定義した。データベースの情報を更新するためのコードを書いた。

(※文責: 椛島司遠)

3.6 中間発表のまとめ

3.6.1 発表内容

中間発表は7月7日金曜日の15:20~17:30に本学で行われた。発表は前半3回と後半3回に分けられており、前半3回は函館水産チームが発表し、前半と後半の間に10分の休憩をはさみ、後半3回は琵琶湖チームが発表した。また、指定された場所に、事前に作成したポスターを設置し、モニターを用いてプレゼンテーションを表示しながら行った。また、発表に関するアンケートをGoogleフォームにて行ってもらい、発表技術と発表内容に関して評価を受けた。我々の発表は、フロントエンド班、バックエンド班、データベース班がそれぞれプレゼンテーションのスライドを作成し、発表を行うことで、内容に誤りがないようにした。また、質疑応答の時間を5分間設けることで、スライドだけではわからない内容に関する質疑に対して、具体的に説明できるようにした。発表の内容としては、研究背景、先行事例、現在の課題を始めに説明することで、本プロジェクトの目的を決定した過程がわかりやすいようにした。その後、フロントエンド班、バックエ

ンド班、データベース班に分け、それぞれ十分な時間を確保することで、専門的な内容について具体的に説明できるようにした。最後に今後の目標を現在のプロジェクトの成果と課題を取り上げながら説明した。

(※文責: 古明地颯天)

3.6.2 いただいた意見・質問・回答

意見

いただいた意見、質問に関しては以下にまとめる。

- 高い目標に向けて計画的に行うことが出来ていた。
- 具体的なデータをスライドにまぜるとわかりやすくなると思う。
- 目的のアプリが漁師さんへの利用という指針をずらさずに設計を行うとしているのは重要だと思います。
- 後期の目標がしっかり決められていて何が問題か、それをどう改善するかが分かりやすかった。
- 論旨が明快で、とてもわかりやすかったです。ゴールとタスクが十分に練られているなど思いました。欲を言えば、もう少し個々のユーザーの視点から、活動がどう変わるのか、何がどんな風に嬉しいのかといったお話も聴きたかったです。
- 先行研究や事例を踏まえた上で、その課題点を克服するという目標があり、明確でわかりやすい。
- 課題、先行事例などをしっかりと把握し、現状のシステムの間隙を検討していたため。
- 目標が高く設定されているが、具体的な方法などについて考えられていたので頑張っ欲しいと思う。
- 既存のアプリの機能へ追加機能を与えるという形で開発を行う点がよかった。
- 計画的に取り組めるように感じた。実現可能性の部分が少し気になった。
- 目標、背景、理由付けがしっかりしている。
- アプリとしての最終完成が素晴らしかった。
- これから実装する機能が明確だったのでわかりやすかった。
- 高齢化に DX に対応していて素晴らしいと思った。
- プロットした航跡を見れるのが重要なところだと思うので時系列も見れるようにして欲しい。

質問とその回答

中間発表で質問された内容と回答を以下にまとめる。

質問： Web アプリケーションの対象となるユーザーは誰なのか。高齢の漁師なのかそれとも新規の漁師なのか。

回答： 我々の Web アプリケーションは全ての琵琶湖の漁師を対象としたものである。

質問： 漁場と漁獲量の関係がわかると漁師間で漁場の取り合いになってしまうのではないか。

回答： 我々の Web アプリケーションでは活用されていないデータを見える化し、それによって、漁師間でお互いの活動を正確に把握することが出来るようになり、水産資源の保護につなが

る。そのため、漁師間で漁場の取り合いになることはない。

質問：なぜ定置網の発表をしないのか。

回答：前半と後半でチームを分割したため、片方のチームしか説明できなかった。

質問：プロットした航跡は時系列を見れるのか。

回答：現時点では見ることは出来ない。今後の実装でみれるようにしたい。

質問：この Web アプリケーションの新規性とはなにか。

回答：我々の Web アプリケーションの新規性は、漁船から送られてきているデータを自動的に収集し、漁師に情報を提供する点である。

(※文責: 横田凌也)

3.7 後期に向けての課題

前期ではデータベースの構築、航跡から漁を行った一部の楕円の面積計算が可能となった。また、Web アプリケーションのモックアップを作成することができた。これらの成果から後期に向けて大きく 4 つの課題が生じた。

1 つ目は、漁を行った全ての楕円の面積計算を可能にすることである。楕円の大きさにはばらつきがあり、漁を行った一部の楕円の面積だけでは、他の漁船における楕円の面積と比較することができない。後期では面積を求めるプログラムを改善し、全ての楕円の面積を求める。さらに、面積計算の精度をあげることで、漁を行った面積を互いに比較し、漁を行う上で最適な楕円の大きさを漁業者に提案したいと考えている。

2 つ目は、Web アプリケーションにデータベースに存在する最新のデータを常に反映させるようにすることである。前期では、Web アプリケーションとデータベースがそれぞれ独立しており、データベース上に漁船の航跡や出港時刻、帰港時刻といったデータは存在しているものの、それらのデータを Web アプリケーションに反映させることができていない。後期では、漁業者がリアルタイムの情報を得られるよう、データベース上に存在する最新のデータを常に Web アプリケーションに反映させ、次に行う漁で活かしていただけるようにしたいと考えている。

3 つ目は、Web アプリケーションのマルチデバイス化である。前期では、Web アプリケーションのモックアップがパソコン画面のみにしか対応していない。パソコンであると、画面を開いて操作するまで時間がかかるため、漁船の上や出港する前のタイミングで素早く確認することができない。後期では、スマートフォンやタブレット端末の画面にも対応できるようにし、漁業者が漁のデータや漁船の航跡データを手軽に見られる環境を作りたいと考えている。

4 つ目は、湖（うみ）レコとの連携を行うことによる漁獲量データの見える化である。前期は、湖（うみ）レコの漁獲量データを利用することができず、データを分析することはできなかった。後期は、湖（うみ）レコのデータを使用する許可が下りた場合、過去の漁獲量や漁場のデータから、漁獲量が増加すると予想される漁場を見える化し、今後漁業者が効率的に漁を行えるよう提案していきたいと考えている。

また、これらの 4 つの課題だけではなく、中間発表でいただいた質疑、指導教員からのアドバイスや沖島漁業協同組合の漁業者が抱えている課題をもとに、熟練漁業者だけでなく新規の漁業者にも効率的に漁を行ってもらえるよう、データの見える化を進めていきたいと考えている。

(※文責: 古明地颯天)

第 4 章 前期結果

4.1 前期プロジェクトの結果

1. プロジェクト目標の設定、方針を決定
 - 「漁業の見える化を行い、それを分かりやすく漁業者に伝え共有できる Web アプリケーションの作成」を目標として設定出来たため、達成できた。
 - このプロジェクトの計画を立てられたため、達成できた。
2. 琵琶湖に関する情報収集
 - 琵琶湖の現状・行われている先行事例に調べ理解したため、達成できた。
3. データベースの構築
 - 前処理（時刻を JST に変更）することができた。
 - 我々のデータベースにデータを保存することができた。
 - データベースの自動更新することが出来ていない。
4. 天気を表示する機能
 - 複数サイトから天気データを取得することは出来なかったが、1つのサイトから天気データを取得することができた。
 - csv 形式でデータを保存することが出来た。
5. 航跡のプロットを行う機能
 - 航跡を可視化することができた。
 - 可視化した航跡を HTML 形式で保存できた。
6. 面積を求める機能
 - すべての漁場の面積を求めることはできなかったが、1箇所分の面積を求めることができた。
7. 漁獲量を出力する機能
 - 湖（うみ）レコに漁獲量のデータを使わせてもらうことを依頼している段階である。
8. Web アプリケーションの作成
 - モックアップの作成は達成した。
 - 我々のデータベース上にある可視化した航跡・漁場の面積の情報を表示することができなかった。

(※文責: 関崎証)

4.2 成果の評価

4.2.1 データベースの構築

データベースは直接は成果に影響しないが、早めに構築することができチーム全体の進行に影響を及ぼさなかった点で評価できると考える。また、データベースのアップデートをするプログラムを Python で書いた。リアルタイムに更新できるシステムの構築に必要であり、後期の Web アプリ

ケーションを作成する際に後からリアルタイム更新のための変更をする必要無いため、前期中に書くことができ、グループの進行を止めずに済んだ。

(※文責: 梶島司遠)

4.2.2 天気を表示する機能

天気についてはデータの取得先を今のところ tenki.jp にしているが、今後は漁業者に普段どこから天気予報の情報を取得しているかを聞き、その天気予報を反映できるようにする。また、漁業者が特に決まった天気予報を見ていない場合やそれぞれ違う天気予報を参照している場合、複数の天気予報サイトから情報を取得する。

(※文責: 松本葉太)

4.2.3 航跡のプロットを行う機能

Python のライブラリにある folium を用いて、漁船の座標をマップ上にプロットすることができた。しかし、データの欠損や、日をまたぐ漁によって、途切れている航跡もあった。後期では、航跡を途切れさせないように取り組みたいと考えている。

(※文責: 古明地颯天)

4.2.4 面積を求める機能

前期では、主に座標のみを用いて漁場の特定を行ったが、漁船の航行速度や進行方向などの情報を使う方法も模索した。座標のみを使う場合、精度に不安が残るため後期では後者の方法も取り入れていく。

(※文責: 松本葉太)

4.2.5 Web アプリケーションの作成

Web アプリケーションで、天気・航跡・面積・漁獲量・出港時刻および帰港時刻の 5 つのページのモックアップの作成を行うことができた。しかし、中間発表会において「現在のモックアップでは誰の航跡なのかが分かりづらい」という意見をいただいた。モックアップに関しては修正を行う必要があると考える。また、データベース上にある可視化した航跡・漁場の面積の情報を表示することができなかった。さらに後期では、マルチデバイス対応させることおよび、湖（うみ）レコの漁獲量を表示させていきたい。

(※文責: 関崎証)

4.3 担当分担課題の評価

4.3.1 関崎証

グループリーダー業務

メンバーの仕事分担・スケジュール管理・議事録作成を行った。大きな問題を起こすことなく進めることができた。

Web アプリケーションの作成

Figma を用いて作成された Vue.js の Vue Router 追加し、ページ遷移を実装した。しかし、作成したモックアップの UI の一部が見つらいという指摘を受けた。後期では、UI の改善・データベース上にある可視化した航跡等の情報を表示させたい。また、湖（うみ）レコや気象情報の外部サイトからの情報も表示させていきたい。

ポスター作成

ポスターの「背景・目的」を担当した。また、全体の文章の確認を行った。

発表スライド作成

背景・先行事例・目的のスライド作成及び、発表を行った。

(※文責: 関崎証)

4.3.2 椛島司遠

航跡のプロット

航跡のプロットをゴールデンウィーク中に完成させることができた。結果的に面積を求めるプログラムの作成に重要な機能になったため早めの完成でチームに貢献できたと考える。

データベースの構築

データベースの構築は他のすべての作業の前提となるため早急に構築する必要があった。プログラムの検証のために6月上旬にデータベースを構築できた。また、Python でデータベースの差分をアップデートするプログラムを書いた。

(※文責: 椛島司遠)

4.3.3 古明地颯天

漁場の面積計算

漁船の航跡から漁を行った場所である楕円の面積を全て求めようとした。前期では、プログラムで楕円と判別できていない場所があり、一部の楕円の面積しか求めることができず、分析に至らなかった。また、プログラムもループ文が多く、実行が完了するまで30秒ほどかかっていた。後期では、他のメンバーと協力し、楕円を識別できるようにプログラムを改善し、精密で迅速な処理が行えるようにする。また、楕円の面積を分析し、効率的な漁を行えるようにしたいと考えている。

4.3.4 松本葉太

漁場の面積計算

座標データをもとに漁場を特定し、扇形の一部である楕円型の漁場の面積を求めた。しかし、この面積は緯度経度をもとに計算されたものであり、メートル換算して見やすくする必要がある。また、地球が球体であることを考慮していないため、正確に漁場の規模を計算できてない可能性がある。

天気情報の取得

個人的に作った天気情報を取得し Twitter に投稿するプログラムを流用し、csv 形式で出力できるようにしたため、システム自体は早期に完成させることができた。

(※文責: 松本葉太)

4.3.5 横田凌也

Web アプリケーションの作成

Figma を用いて、UI デザインを作成した。その後、Figma のプラグインを用いて Vue.js 使用できるコードに変換した。後期では、デザインの操作性の向上を目指し、現在画面遷移しか出来ていないところを Web アプリケーションから航跡や天気等の情報を直接見れるようにする。

ポスター作成

Figma を用いて発表ポスターの作成を行った。プロジェクトメンバーに「背景・目的」「アプローチ」「結果」の各項目について原稿を書いてもらった。その後、英語に翻訳した各項目の校正を行った。過去のポスターのデザインを参考にしながらデザインの構成を考え作成した。

(※文責: 横田凌也)

第 5 章 後期活動内容

5.1 活動目標

後期の活動目標については以下に示す。

- プロジェクトの後期目標の設定
課題: 前期の活動の成果及び達成度を鑑みて、琵琶湖チームのメンバー全員で話し合いを行う。そして、プロジェクトの到達目標の再設定を行う。また、ここで設定した到達目標を達成するためのスケジュールの設定を行う。
- 航跡データの分析
課題: 前期に達成できなかった我々のデータベースから日付、時刻 (UTC)、漁船 id、緯度、経度、航行速度、漁船の進路方向のデータを取得し、漁を行った面積を求めるシステムを完成させる。
- Raspberry Pi に本番環境を構築
課題: 本番環境として、Raspberry Pi にデータベースサーバー、API サーバー、Web サーバーを構築する。
- データベースの更新
課題: 航跡データの分析（漁を行った面積及び距離）の結果を格納するテーブルの追加を行う。また、湖（うみ）レコで収集している漁獲量データの格納を行う。
加えて優先度の低い目標として、和田教授が収集している日付、時刻 (UTC)、漁船 id、緯度、経度、航行速度、漁船の進路方向のデータを格納しているデータベースから、我々のグループのデータベースにデータを格納すること、時刻を JST に変換する前処理、前処理を行った航跡データの分析を自動で行う処理を追加する。
- API の作成
課題: 我々のデータベース上に格納されている航跡の表示結果、航跡データの分析結果、湖（うみ）レコで収集している漁獲量データを、湖マップに渡すために API の設計及び実装を行う。
- Web アプリケーションの作成
課題: 前期に Figma で作成したモックアップをもとに、Vue.js を用いて Web アプリケーションを作成する。
- Web サーバーの構築
課題: Apache を使い、Web サーバーを構築する。また、最低限 Basic 認証の導入を行う。加えて努力目標として、Basic 認証とは別の認証を実装し、アカウントごとに権限を切り替えることを目指す。
- Web サーバーのアクセスログ取得
課題: Apache のアクセスログを取得し、Discord で通知を行うシステムを実装する。
- ヒアリングの実施
課題: データの提供をしていただいている琵琶湖の漁業者、滋賀県水産業普及指導員へのヒアリング及び作成した Web アプリケーションに対するフィードバックを頂く。

5.2 役割分担

5.2.1 フロントエンド班

Web アプリケーションの実装

我々は Vue.js を用いて Web アプリケーションを作成する。

- ヘッダー

機能: ヘッダーでは、湖マップのロゴ、湖マップのタイトル及びホーム、航跡、漁の面積、漁獲量のページに移動するためのタブを表示する。

- ホーム画面

機能: このページでは、3つの要素から構成される。1つ目は、湖マップの作成目的及び概要を表示する。2つ目は、各ページの操作方法についての説明を表示する。3つ目は、各ページの簡単な説明を表示する。

(※文責: 横田凌也)

- 航跡画面

機能: このページでは、日付と漁船 id を選択することができる。そして、選択されている日付に操業していたすべての漁船の航跡と、選択された漁船 id に注目した航跡を見ることができる画面を実装する。

- 漁の面積画面

機能: このページでも「航跡画面」同様、日付と漁船 id を選択することができる。そして、選択された漁船 id に注目した航跡、漁ごとの面積・距離の一覧、漁の面積と漁獲量の図、漁の距離と漁獲量の図を見ることができる画面を実装する。

- 漁獲量画面

機能: このページでも「航跡画面」同様、日付と漁船 id を選択することができる。そして、選択されている日付が含まれる一週間分の漁獲量を日付・魚種ごとに表示する図、選択されている漁船 id の合計漁獲量を表示する図を見ることができる画面を実装する。このページで見ることができる漁獲量データは、湖（うみ）レコから提供していただいたデータであり、漁業者自身が報告したものである。

(※文責: 関崎証)

5.2.2 バックエンド班

面積の求め方の変更

前期の面積の求め方では、漁場の一部しか判定、計算することができなかった。そこで、後期では別の方法を試す。手順は3段階ある。

I. 航行速度が0になっている座標データから漁場の候補データを集める

前期の時点で作成した図 3.5 をもとに航行速度が0になるタイミングに着目した。我々は、航行速度が0になっている2点で囲まれた航跡が漁場になっているのではないかと予測した。そこでデータを時系列順に並べ、時刻が早い順から2点1組を作り、その間の時刻の座標データと2点を含む座標データの集まりを一つの楕円型の漁場の候補データとして扱う。実際にその航跡を可視化した結果が以下の図 5.1 である。また、図 5.2、5.3 は図 5.1 の一部を拡大したものである。近くの同じ色同士のマーカーは、同じ一つの楕円形の漁場に属していると判定されている。これらの図を見ると、ほとんどの漁場が特定できていることが分かる。

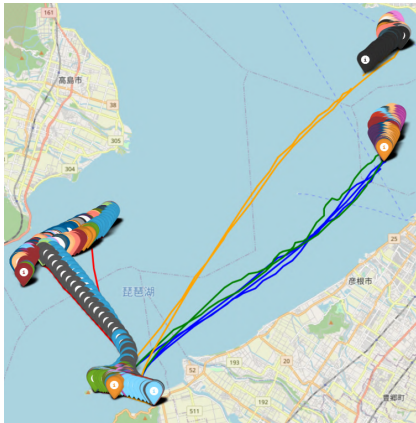


図 5.1 航行速度0で囲まれた点の全体図

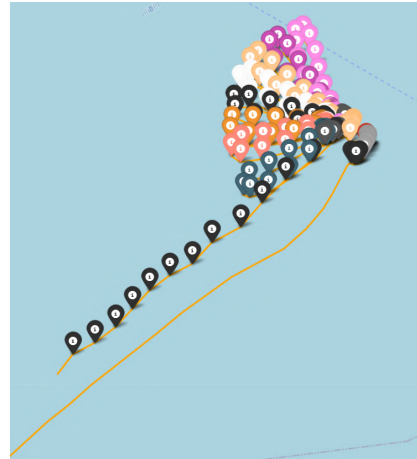


図 5.2 漁港から遠い漁場

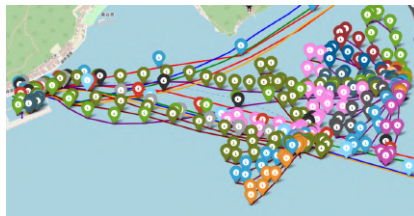


図 5.3 漁港から近い漁場

II. 制約で余分な候補データを除く

これら3つの図を解析した結果、それぞれ候補データから除かなければならないデータが3種類あることが分かった。最初に図 5.1 では、左下の方で別の漁法が行われているが、候補データとして捉えられてしまっている。別の漁法の排除については難しく、同じ漁船の id でも違う漁法を季節などによって使い分けるため、漁船 id で判断することができないのである。つまり、航跡の特徴から判断する必要がある。次に図 5.2 では、黒いマーカーの部分は実際には漁場ではない。最後に図 5.3 では、漁港付近で漁船が止まっていることが多く、それらが候補データとしてとらえられてしまっている。これを排除する一番簡単な方法は、漁港は動かないため、緯度経度を固定値で指定し、特定の範囲のデータについては候補データの判定プログラムに入力しないようにすることである。

別の漁法、楕円型ではない漁場、漁港のこれら3つの要素を除くため、前手順で求めた2点1組の座標データから道のりと距離を求めて、道のりと距離との比で判断する方法を考えた。検知した

い漁法の特徴は楕円型であり、2点間の距離が短くなり、それに対して道のりが明らかに長くなるはずである（図 5.4）。

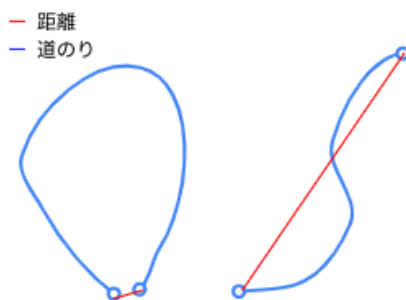


図 5.4 2点間の道のりと距離の比較

この方法を用いて、3種類の排除しなければならないデータを排除できると考えた。漁港は比較的静的なデータであり別途対応もできるが、新しく漁港ができる、減ることなどを考慮して動的に対応することが望ましい。今回は、距離に対して道のりの比が20%以上であれば漁場ではないと判断し除外することとした。以下の図 5.5、5.6、5.7 に制約を施した後の結果を示す。

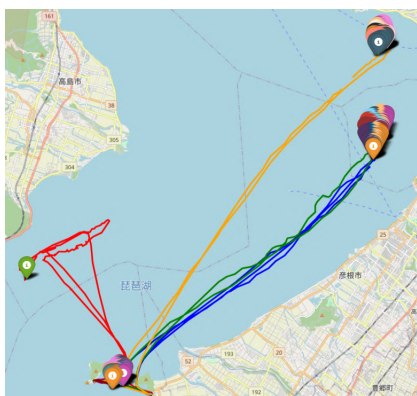


図 5.5 全体図

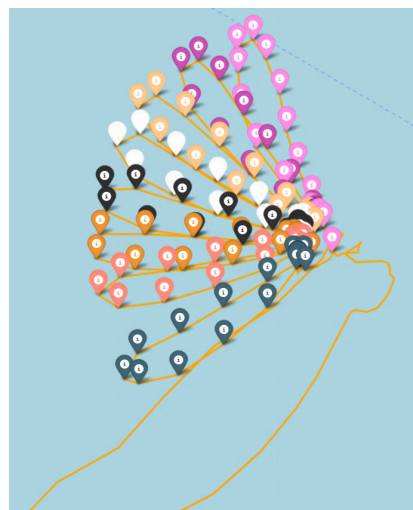


図 5.6 漁港から遠い漁場

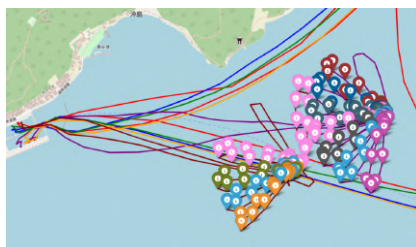


図 5.7 漁港から近い漁場

制約の効果があつたことが図 5.5 からわかる。しかし、すべてがうまくいったというわけではない。赤色の航路は今回判定したくない別漁法の航跡だが一つだけ漁場と判定してしまっている。これは、制約前の段階でごく短い曲線であつたと考えられる。そのため今回設定した制約をすり抜け

てしまった。これに関してはデータを集めて、他の日でも同じような傾向があるなら制約前の曲線の長さなどによりさらに制約をかける必要がある。漁港から遠い位置の漁場は図 5.6 から正しく判定できているとわかる。青と緑の航跡でも同じような結果となった。しかし、図 5.7 の漁港から近い二つの航跡は楕円型の漁場をそれぞれ一つずつ取り逃してしまっている。制約をかける前の図 5.3 と比較すると取り逃してしまっている漁場は制約前に漁港を巻き込んでいたものだとわかる。このことから、漁港付近の航跡の航行速度が 0 になるパターンは私たちが予測したパターンになるとは限らない。これを解決する糸口は、漁港から出た漁船、もしくは戻る漁船は航行速度を 0 にすることなく直接漁場に入るまたは、出るなどして漁を行っている可能性がある。つまり、漁の最初か最後の楕円形の漁場だけを特別扱いして漁場と判定するということである。この場合、特別な漁場は楕円の図形に曲線が尾のようについている形になるため、正確な面積を求めるためには、この尾を取り除くような処理が別途必要になる。

III. 求めた漁場の座標データから面積を求める

面積の計算方法について説明する。前期と変わって面積を求めるにあたって geopandas というライブラリを使った。こちらでは、地球が球体であることを考慮して、球体の表面積として計算できる点で前期よりも進化した。まず、地理座標系は緯度経度であるため、座標間の距離をメートルで表せるようにしたい。今回は投影座標系の一つの UTM 座標系（ユニバーサル横メルカルト図法）に変換する。これは、球体（3次元）を2次元表面上に投影する座標系であり、国土地理院などでも使われている座標系である。変換する際に緯度経度の EPSG から UTM ゾーンがどこなのかを調べる。結果、EPSG が 4326、UTM ゾーンは 53、であった。日本国内でも変わるため注意が必要である。

これらの情報をもとに求めた漁場の座標データを UTM 座標系に変換した。これには、pyproj ライブラリの Transformer を用いた。次に、変換した座標データをポリゴンにする。漁場の航跡データはまだ点の集まりであり、それらの点を結ぶ必要がある。そして、面積を求められる図形とするために航跡データの最初の点と最後の点を結ぶ必要がある。これがポリゴンに変換する理由である。変換には、shapely.geometry にある polygon ライブラリを用いた。入力した座標データを順々に線で結んでいくので、最後に最初と同じ座標を入力する工夫が必要である。

最後に、作成したポリゴンの面積を求める。そこで、geopandas の area 関数を使うことで面積を求める。面積は漁船ごとと、楕円上の漁場ごとに求めた。

（※文責: 松本葉太）

API

データベース上にある漁船の航跡や漁場の面積、漁獲量、年月日といったデータを特定の URL にアクセスすることで Web アプリケーションが取得できるよう作成する。データベース上にある年月日のデータは、それぞれ別のカラムで保存されているが、湖マップでは年月日をつなげた 1 つの文字列として出力することから、1 つの文字列に型を変換するプログラムを書く。また、datetime を用いて、年月日から週番号を求める。加えて、データベース上のデータをフロントエンド班が扱いやすい形式にするため、JSON のフォーマットに変更して Web アプリケーションに返すようにする。

ログの取得と出力

Web アプリケーションにアクセスしたユーザーの情報を取得するために Apache のアクセスログを取得する。具体的には、ユーザーのアクセスした日時やページの URL、OS、使用したブラウザを取得する。また、Discord.py を用いて Discord Bot を作成し、アクセスしたユーザーの情報を自動的に出力するプログラムを書く。

(※文責: 古明地颯天)

5.2.3 データベース班

Raspberry Pi のセットアップ

本プロジェクトではサーバーとして Raspberry Pi を用いることにした。そのため OS のインストール及びネットワーク接続の設定を行う。

PostgreSQL のインストール

データベースとして PostgreSQL を利用するため、インストール及びユーザー作成などの設定を行う。また、テーブルには個人の端末上でテスト用に動かしていたデータベースをダンプし、それを読み込ませる。

Apache のセットアップ

Web サーバーとして Apache をセットアップし、公開のための設定を行う。

mod_wsgi のセットアップ

Flask で書いた API のコードを Apache 経由でアクセスできるようにするために mod_wsgi を利用した。そのための連携の設定を行う。

(※文責: 椛島司遠)

5.3 後期の成果

5.3.1 フロントエンド班

Web アプリケーションの実装

我々は Vue.js を用いて後期で改めて定めた機能をもつ Web アプリケーションの作成を目指し実装を行った。

以下、行った実装とその結果をまとめた。

- ヘッダー

実装: ヘッダー及びホーム画面は、Vue.js フレームワーク向けのマテリアルデザインコンポーネントフレームワークである Vuetify を使用した。アプリケーションのトップバーを表す v-app-bar を使用した。color プロパティを使用し、ヘッダーの色を黒に設定した。v-toolbar-title コンポーネントを使用し、湖マップのロゴを設置した。v-tabs 及び v-for を使用して menuItems からメニューアイテムを動的に生成した。menuItems で

は各タブへのリンクを設置した。サイドメニューを開くためのアイコンを作成するために、v-app-bar-nav-icon コンポーネントを使用した。クリックイベントが発生したときに、表示/非表示を管理する変数の値を切り替え、サイドメニューを表示/非表示にする機能を実装した。サイドメニューを表す v-navigation-drawer コンポーネント内で、v-model を使用して表示/非表示を管理する変数の状態にバインドする。v-list 内にはサイドメニューのアイテムが動的に生成されるように実装した。

結果: 以下は、実際に作成した湖マップのロゴや各ページへの移動のためのタブ (図 5.8) を表示した画面である。



図 5.8 各タブの表示

● ホーム画面

実装: 1つ目の作成目的及び概要のセクションは、セクションのタイトルを表示し、v-container コンポーネントの中では、6:5 の比率で湖マップのロゴと湖マップについての文章を表示した。2つ目の各ページの操作方法についてのセクションは、v-carousel コンポーネントを用いて、日付と漁船 id の選択方法を横にスライドして見ることができるように表示した。3つ目の各ページの簡単な説明についてのセクションは、v-card コンポーネントを用いて、クリックすることで指定のページに移動することが出来るようにした。各要素では、航跡、漁の面積、漁獲量の3つを対象に作成した。対象のページのメイン画像と簡単な説明を表示した。

結果: 3つのセクションを実装した。1つ目は、湖マップの作成目的及び概要 (図 5.9) を表示した画面である。2つ目は、各ページで使用する日付と漁船 id を選択する操作方法 (図 5.10) を表示した画面である。3つ目は、各ページの簡単な説明 (図 5.11) を表示した画面である。

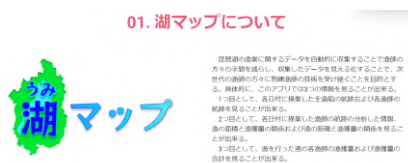


図 5.9 湖マップの作成目的及び概要



図 5.10 湖マップの操作方法

図 5.11 各ページの簡単な説明

(※文責: 横田凌也)

- 航跡画面

実装: 日付の選択は、HTML の input 要素の type 属性に date を指定し実装した。また、漁船 id の選択は、select 要素に含まれる option 要素に存在する漁船 id を追加することで実装した。ただし、航跡の漁船 id と湖（うみ）レコの漁獲量データにおける id が違うため変換を行った。漁船の航跡は API によって HTML ファイルを得ることができるため、そのファイルを iframe 要素を使い表示した。この HTML ファイルを取得し表示する一連の動作は、ページ遷移時、検索ボタンをクリックされるたびに行った。

結果: 航跡表示画面を定義通りに実装することができた。以下は、選択されている日付に操業していたすべての漁船の航跡（図 5.12）、選択された漁船 id に注目した航跡（図 5.13）を表示した画面である。



図 5.12 選択されている日付に操業していたすべての漁船の航跡

shiga13さんの漁船別航路

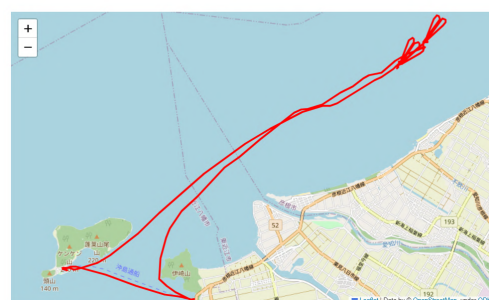


図 5.13 選択された漁船 id に注目した航跡

- 漁の面積画面

実装: 漁船 id は、「航跡画面」と同様の航跡の漁船 id と湖（うみ）レコの漁獲量データの id の変換処理を行った。選択された漁船 id に注目した航跡の実装は、上記と同様である。漁ごとの面積・距離の一覧の実装は、API で受け取った値をリストに格納し、Vue.js の v-for ディレクティブでレンダリングした。漁の面積と漁獲量のヒストグラム、漁の距離と漁獲量のヒストグラムでは、始めに漁の面積・距離の値を取得する API から計算結果を取得した。その後、その結果に含まれる漁獲量データの id の漁獲量を取得する API の結果を取得した。そして、漁獲量データの id、日付が完全一致するデータのみを抽出することで、漁の面積に対する漁獲量、距離に対する漁獲量を得ることができた。最後に、それらの結果を Chart.js を用いて図を作成した。これらの選択された漁船 id に注目した航跡、漁ごとの面積・距離の一覧、漁の面積と漁獲量の図、漁の距離と漁獲量の図を作成する一連の動作は、ページ遷移時、検索ボタンをクリックされるたびに行うように実装した。

結果: 漁の面積画面を定義通りに実装することができた。以下は、漁ごとの面積・距離の一覧（図 5.14）、漁の面積と漁獲量のヒストグラム（図 5.15）、漁の距離と漁獲量のヒストグラム（図 5.16）を表示した画面である。

- 漁獲量画面

実装: 漁船 id は、「航跡画面」と同様の航跡の漁船 id と湖（うみ）レコの漁獲量データの id の変換処理を行った。選択されている日付が含まれる一週間分の漁獲量を日付・魚種ごとに表示する図の実装では、選択された日付の漁獲量、選択された日付の漁船の漁獲量

航跡分析

Index	ID	年	月	日	漁を行った面積	漁を行った距離
0	shiga12	2022	9	13	4075430.1573706325 m ²	9091.0792579807 m
1	shiga12	2022	9	13	19.6096289642 m ²	25.1541256149 m
2	shiga12	2022	9	13	7.0035188043 m ²	29.7044149018 m
3	shiga12	2022	9	13	768.8919118365 m ²	278.9354535789 m
4	shiga12	2022	9	13	2529.5552384515 m ²	222.4201755598 m
5	shiga12	2022	9	13	12630.2139988773 m ²	711.5639479957 m
6	shiga12	2022	9	13	988090.3443783512 m ²	5637.7419970834 m
7	shiga12	2022	9	13	4281.4577495449 m ²	482.9990881244 m

図 5.14 漁ごとの面積・距離の一覧

漁の面積と漁獲量

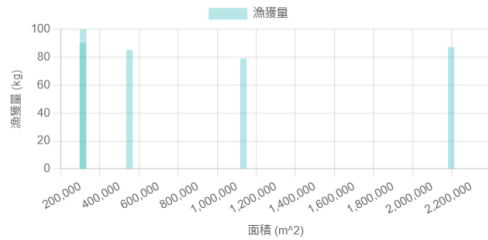


図 5.15 漁の面積と漁獲量のヒストグラム

漁の距離と漁獲量

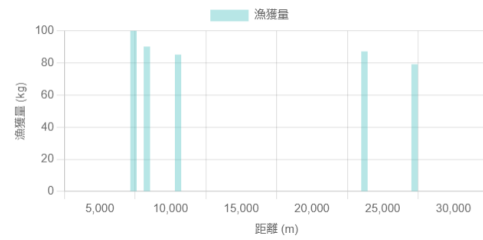


図 5.16 漁の距離と漁獲量のヒストグラム

を取得する API から値を取得した。ただし、この API から取得したできる JSON 形式の結果の順番は担保されない。そのため、日付順にソートする必要があった。ソートの処理では、始めに日付とその日付の順番を別変数に格納した。そして、その変数に対して日付順にソートを行い、適切な順番に並べ直した。最後に、ソートされた順に、元データの中身をそれぞれ入れ替えた。このソート処理を終えたデータを Chart.js を用いて図を作成した。選択されている漁船 id の合計漁獲量を表示する図では、ソートの処理は必要なく、取得した値を Chart.js を用いて図を作成した。これら、選択されている日付が含まれる一週間分の漁獲量を日付・魚種ごとに表示するヒストグラム、選択されている漁船 id の合計漁獲量を表示するヒストグラムを作成する一連の動作は、ページ遷移時、検索ボタンをクリックされるたびに行うように実装した。

結果: 漁獲量画面を要件通りに実装することができた。以下は、選択されている日付が含まれる一週間分の漁獲量を日付・魚種ごとに表示するヒストグラム (図 5.17)、選択されている漁船 id の合計漁獲量を表示するヒストグラム (図 5.18) を表示した画面である。

shiga13さんの 2022/9/12 0:00 から 2022/9/16 0:00 までの日にちごとの漁獲報告情報

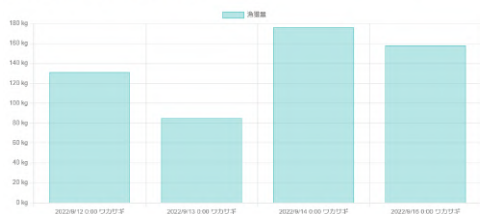


図 5.17 選択されている日付が含まれる一週間分の漁獲量を日付・魚種ごとに表示するヒストグラム

shiga19さんの 2022年 第37週 漁獲報告情報の集計



図 5.18 選択されている漁船 id の合計漁獲量を表示するヒストグラム

(※文責: 関崎証)

5.3.2 バックエンド班

システムの構成

制作していく途中で前期のままではシステムの設計に問題があったため図 5.19 のようにした。

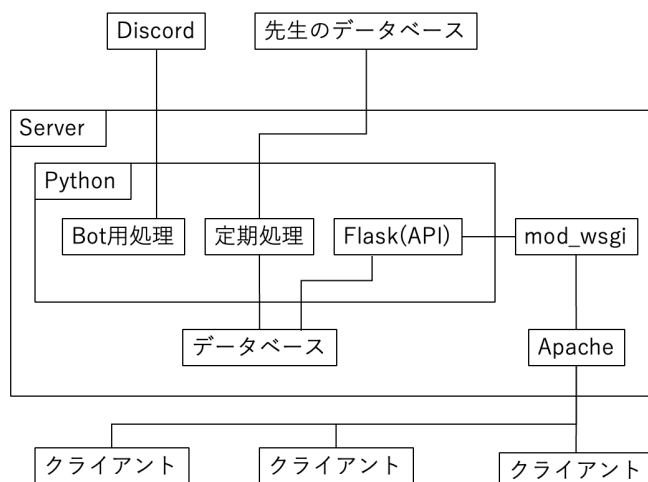


図 5.19 システム構成図

前期からの変更点としては大きく2つある。1つ目は Apache と Python の間に mod_wsgi という Apache のモジュールを追加したことである。これは Flask のみでも build-in サーバー機能で動かすことはできるが、ドキュメントによると「製品用に使用するべきではない」と書いてあるため build-in サーバーの使用をとりやめたためである。2つ目は Discord への通知を送るための Bot 用の処理を追加したことである。これは前期の段階では通知用処理の機能が十分に考慮できていなかったためである。

(※文責: 梶島司遠)

求めた面積の精度

求めた面積の精度の求め方と比較について説明する。今回は簡易的に QGIS で求めたデータを使って比較する。我々が求めた漁場の座標データを QGIS に読み込みポリゴンを作成する。実際の航跡と見比べて、正しい形のポリゴンが生成されるように適宜座標を追加・削除する。そのポリゴンの面積を QGIS 内の機能を使って面積を求める。また、この時座標系を UTM 座標系に合わせる。その結果を以下に示す。

loop名	求めた面積 (㎡)	QGISの面積 (㎡)	QGISとのずれ
shiga14loop136	52765	52972	-0.39%
shiga14loop137	37283	37302	-0.05%
shiga14loop138	45560	45583	-0.05%
shiga14loop139	45560	47695	-4.48%
shiga14loop140	39162	39182	-0.05%
shiga14loop141	31400	31416	-0.05%
合計	251730	254150	-0.95%

図 5.20 漁港から遠い漁場

loop名	求めた面積 (㎡)	QGISの面積 (㎡)	QGISとのずれ
shiga15loop160	0	11276	-100.00%
shiga15loop164	9638	9644	-0.06%
shiga15loop171	10740	10746	-0.06%
shiga15loop174	6877	6882	-0.07%
shiga15loop179	12195	12203	-0.07%
shiga15loop183	16007	13519	18.40%
合計	55457	64270	-13.71%

図 5.21 漁港から近い漁場

漁港から遠い漁場では、漁を行った面積の計算精度の平均が全体で 0.95% 過少に求められてい

る。おそらくポリゴンにする過程か面積を求める計算式などでの誤差しかない。一方、漁港から近い漁場の場合では、漁を行った面積の計算精度の平均が全体で 13.71% 過少に求められている。これは、先ほど述べた通り楕円型の漁場を一つ取り逃がし、loop160 の QGIS とのずれが 100.00% 過少になっていることが影響している。また、loop183 は漁場を約 18.40% 過大になっていることにも注意が必要である。これらは、漁船が網を引き終わった後にもしばらく航行を続けた後に止まったことが原因だと考える。そのため図 5.22 のように本来の楕円から少し崩れた形のポリゴンが生成された。

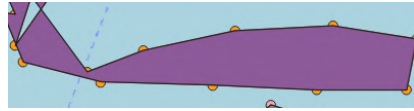


図 5.22 崩れた形の楕円形

このように、漁港付近と遠方で漁船の動かし方や漁場の大きさが異なる。これらは熟練の漁業者が無意識に長年の感覚で効率的に漁を行っている結果だと考察できる。この結果をさらに分析すれば若手漁業者が効率の良い漁法を学ぶ鍵になるかもしれない。

(※文責: 松本葉太)

API

始めに API を Web アプリケーションが取得できるよう、Flask を用いてルーティングを行い、URL にアクセスしたとき、その URL に対応する関数が実行されるようにした。また、データベース上からデータを取得するとき、PostgreSQL を扱うことから、Psycopg2 を導入した。

次に、関数を作成した。Web アプリケーションが取得するデータは、下記の 7 種類である。

- 指定された年月日における全ての漁船の航跡データ
- 指定された年月日における全ての漁船の航跡データを HTML の形式で取得するもの
- 指定された漁船 id 及び年月日に対応する漁船の航跡データ
- 指定された漁船 id 及び年月日に対応する漁船の航跡データを HTML の形式で取得するもの
- 指定された漁船 id 及び年月日に対応する週ごとの全ての魚種を合計した漁獲量データ
- 指定された漁船 id 及び年月日に対応する魚種ごとの漁獲量データ
- 指定された年月日に対応する漁船が漁を行った面積とその距離に関するデータ

URL にアクセスするとき、日付を指定する場合は年月日、漁船 id 及び日付を指定する場合は漁船 id と年月日に対応する部分の URL をそれぞれ変更することで、指定したデータを Web アプリケーションが取得できるようにした。年月日を取得するとき、Web アプリケーションでは年月日を 1 つの文字列で扱うが、データベース上では年月日が別々に保存されているので、型を変換した。加えて、datetime を用いて週番号を求めた。

最後に、データベース上のデータをフロントエンド班が扱いやすい形式にするために、JSON のフォーマットに変更して Web アプリケーションが取得できるようにした。

ログの取得と出力

アクセスログファイルを取得できるように、scp を用いてアクセスログファイルをプロジェクト学習のファイルにコピーした。また、ネットワーク上にある Web サーバーを管理するために Paramiko を導入した。ホスト、ポート番号、ユーザー、パスワードをあらかじめ入力しておくことで遠隔でも管理できるようにした。アクセスログファイルを取得した後、データを見やすくするため、データをパースした。パースしたデータから今回必要なユーザーのアクセスした日時、URL、ホスト、OS、利用したブラウザのデータをデータフレームに保存し、カラムごとにユーザーの情報を分類した。

ログの出力には Discord Bot を用いた。プロジェクト学習で用いている Discord サーバーにログを出力するチャンネルを作り、チャットで「/log」と入力することで Discord Bot がパースしたログを出力するようにした。アクセスログファイルはプログラムファイルを起動することで、自動的に最新のファイルを取得することができる。

(※文責: 古明地颯天)

5.3.3 データベース班

Raspberry Pi のセットアップ

最初に OS のインストールを行った。OS には Ubuntu Server 22.04 を利用し、SD カードへのインストールは Raspberry Pi Imager を利用した。ネットワークの設定は最初 free-wifi へ接続し利用していたが安定しなかった。そこで、フォーラムの投稿 [4] を参考にしてドライバの一部の機能を無効化したところ接続が安定した。

PostgreSQL のセットアップ

データベースとして PostgreSQL を利用するためインストール及びデータベース作成、ユーザー作成などの設定を行った。また、テーブルには個人の端末上でテスト用に動かしていた物をダンプし、それを読み込ませた。

Apache のセットアップ

Web サーバーとして Apache を使用した。LAN 内のみで利用するため Basic 認証のみをかけた。Web サイト [5] を参考にし、以下のような設定にした。

```
<Directory /var/www/html>
AuthType Basic
AuthName "Basic Authentication"
AuthUserFile /etc/apache2/.htpasswd
require valid-user
</Directory>
```

mod_wsgi のセットアップ

Flask で書いた API のコードを Apache 経由でアクセスできるようにするために mod_wsgi を利用した。そのための連携の設定を行った。

5.4 琵琶湖ヒアリング結果

琵琶湖で行ったヒアリングの準備、行ったこと、結果について書く。

5.4.1 スケジュール

日時：令和5年11月25日（土）12時30分～16時00分

目的：以下の内容についてのヒアリングを実施する

- － 琵琶湖漁業者について
- － 琵琶湖漁業について
- － 湖マップについて

参加者：久米氏（滋賀県）、奥村組合長・漁業者3名（沖島漁業協同組合）、和田雅昭（教員）、関崎証・椛島司遠（B3）

5.4.2 準備

久米氏、奥村組合長、沖島漁業協同組合の漁業者3名にヒアリングを行うため、2種類のスライドを作成した。久米氏には処理の説明を含めた具体的なスライドを作成した。奥村組合長、沖島漁業協同組合の漁業者3名には作成したWebアプリケーションである湖マップを体験してもらうための説明を行うスライドを作成した。琵琶湖の沖島漁業協同組合にある建物のローカルネットワークをお借りして行う想定でRaspberry Pi及び周辺機器を持ち込んだ。

5.4.3 行ったこと

久米氏には処理の説明を行い湖マップを体験していただいた。奥村組合長、沖島漁業協同組合の漁業者3名には簡単な処理の説明を行い、湖マップ画面を見ていただき評価をいただいた。また、漁に関する知識も教えていただいた。

5.4.4 結果

久米氏（滋賀県）、奥村組合長・漁業者3名（沖島漁業協同組合）に対して実施したヒアリングの結果を以下にまとめる。

琵琶湖漁業者について

- 担い手不足の課題を抱えている
- 湖（うみ）レコの報告の他に、自身で操業日誌を付けている
- 漁業者間で漁に関しての情報交換を行っている
 - － その日に魚が取れたか取れなかったか
 - － 琵琶湖のどこで漁を行ったのか
 - － ただし、上記の情報は秘密にすることもある

琵琶湖の水産資源について

- 琵琶湖の水産資源を管理するため、湖（うみ）レコが運用されている
 - － 湖（うみ）レコの利用人数・報告事項・機能は 1.2.1 の内容と同様である
- 琵琶湖には固有種が多く生息している
- 年々漁獲量が減っている

沖びき網漁について

- 主な漁獲対象は、ゴリ・モロコ・ワカサギ・イサザ・エビなどである
- アンカーを起点に、長いロープのついた袋状の網を仕掛け、漁船を固定し、ロープを巻いて網を上げる
 - － 網を上げ魚が取れていた場合、その周辺で起点からの角度を変えて網を仕掛ける
 - － 魚が取れていなかった場合、漁業者の勘や経験をもとに起点からの角度を大きく変えて網を仕掛ける
- 網を仕掛ける面積は基本的に一定である
 - － 水深が浅い場合などには、小さくなる場合もある
 - － 風などの影響で、面積が小さくなる場合も考えられる
- 連続で同じ場所で漁を行うことが無い
 - － 湖底の泥が巻き上がることで、魚が逃げていくため
- アンカーを沈める場所の決め方は以下である
 - － 前日良く取れた場合は、前日の場所付近にアンカーを沈める
 - － 悪かった場合は、漁業者間の情報交換の結果をもとに新しい場所を開拓する
- 現在の網は化学繊維を使っているため、比重が重く沈みやすい
 - － 従来よりも湖底の泥を巻き込みやすく、魚が逃げやすくなってしまった
 - － 網に泥が溜まりやすく、穴が開きやすくなった
 - － 網に穴が開いた場合には漁ができなくなるため、漁を終了する必要がある
- 網に穴が開くこと、漁船のプロペラに網が絡むことに苦労している
- 魚が取れない場合、網に穴が開いた場合は、燃料代が無駄になる

湖マップについて

- 航跡の可視化は役に立つと感じる
 - － 地図を見ることで漁のイメージが付くので良い
 - － 若手育成対して役立つ
 - － 分かりにくい配色はやめて欲しい
- ヒアリングを実施した熟練漁業者には、漁の面積はあまり魅力を感じない
 - － 若手育成対しては使える可能性もある
- 現在の日付と漁船 id を選択するアプリの操作は難しいと感じる
- 前日や前々日の航跡も同時に見られるようにしてほしい

(※文責: 椛島司遠)

5.5 成果発表会まとめ

5.5.1 発表内容

最終発表会は12月8日金曜日の15:20~17:30に本学で行われた。発表は前半3回と後半3回に分けられており、前半と後半共に、函館水産チームと琵琶湖チームの活動を要点をまとめて2分で説明し、その後チームごとに分かれた場所で発表を行った。聴衆者には、2分の説明の後、興味を引くチームの発表場所へ移動をお願いした。各チームの発表では活動の詳細を8分で説明し、最後に質疑応答とアンケート回答の時間として5分設けた。また前半と後半の間に10分の休憩を挟んだ。また、指定された場所に事前に作成したポスターを設置し、モニターを用いてプレゼンテーションを表示しながら行った。また、発表に関するアンケートをGoogleフォームに入力してもらい、発表技術と発表内容に関して評価を受けた。我々の発表は、フロントエンド班、バックエンド班が担当した活動のプレゼンテーションのスライドを確認しながら作成し、発表を行うことで内容に誤りがないようにした。また、質疑応答の時間を5分間設けることで、スライドだけではわからない内容に関する質疑に対して、具体的に説明できるようにした。発表の内容としては、研究背景、先行事例、現在の課題を始めに説明することで、本プロジェクトの目的を決定した過程がわかりやすいようにした。その後、目的を達成するために見える化したものと、Webアプリケーションを操作しながら説明する時間に十分な時間を確保することで、専門的な内容やWebアプリケーションについて具体的に説明できるようにした。最後に、見える化したときに生じた問題と琵琶湖でのヒアリング結果をもとに今後の展望として説明した。

(※文責: 古明地颯天)

5.5.2 いただいた意見・質問・回答

いただいた意見、質問に関しては以下にまとめる。

- 最初の方の説明は難しいプログラムの説明でよくわからなかった。しかしアプリの使用方法や今後の展望などは非常に分かりやすかった。
- 実際に作ったサイトを表示していてわかりやすかった。
- 実際の可視化例や挙動が紹介されていてイメージが持ちやすかった。
- 水産系のことについては詳しくないのですが、実際の漁師の立場に立って開発を行っているため、ユーザーが使いやすい形に仕上がっていることが発表を聞いて伝わってきました。ヒアリングからの問題発見→開発の流れがよく伝わってきました。
- 過程や改善点などわかりやすかった。
- 可視化自体はすごいと思うが、それが琵琶湖の問題をどう解決できるのか不明瞭だったが、質問すれば丁寧に答えてくれたし納得できた。
- 目的とそれに対するアプローチがわかりやすく説明してて良かった。
- ヒアリングでフィードバックをもらうのが目的に適していて良いと思った。
- はじめて話を聞く人には、航跡の特徴が伝わっていなかったかもしれない。
- 実ユーザーによる評価ができていないプロジェクトがほぼない中で、ちゃんとできたのは良かった。

- 実用性の高い Web ページを作っていた。
- 網を用いた面積の具体的な方法の説明が少し早いと感じてわかりづらい部分があった。
- アプリのフィードバックをしっかりと受けているのが良い。
- 精度の高いアプリを制作していた。
- 実際に行ったことから目的、結果までをしっかりと網羅していたので、不明点などがない発表でよかった。
- 中間発表時の目標のように、実際に動作して使用しやすい感じに作られていたため、当初の目標である後継者作りに貢献できそうだったので、非常に良い取り組みであったと思う。
- 目的に対して結果がしっかり対応していました。
- 技術継承問題の掘り下げがあったらもっと良かった。
- 後継者にどのような支援ができるのかの話を強調できるとより提案システムの有効性が言えたと思います。
- プレゼンのわりと早い段階で今年度の取り組みに入って、活動内容もバラエティがあったのでその疑問はあまり感じなかった。今年度の成果をもう少し俯瞰したマリン IT への貢献を強調できたらさらに良かったか。

質問とその回答

中間発表で質問された内容と回答を以下にまとめる。

質問：テーマとして琵琶湖を選んだ理由は？

回答：和田先生が持っているデータの中で一番データ量が多かったため。

質問：面積を求める理由は？実際にそれがどのように後継者育成に役立つのか？

回答：漁獲量と比較して漁の効率がよかった航跡を真似して漁を行えば若手漁業者でも熟練者と同じ結果を期待できます。

質問：湖マップの表示が重い理由は？今後データが増えれば更に重くなりそうな気がするけど、大丈夫？

回答：発表会当日は Wi-Fi が混んでいたため、動作が遅かったのであり、普段は快適に動いています。

質問：だれがデータを取得しているのか？

回答：漁業者の方々に漁船にセンサーをつけてもらい、和田先生にデータを送ってもらっています。

質問：プロジェクトは依頼されて行ったのか、自発的に目的を決めて行ったのか？

回答：依頼されています。

質問：QGIS のデータは正しいデータなのか？

回答：正解データとしている QGIS のポリゴンは人の目で漁場と判断した座標データから生成している為、完全な正解データとは言えません。

(※文責: 横田凌也)

5.6 今後の展望

5.6.1 面積の計算精度の向上

漁港から遠い場合、漁を行った面積の計算精度の平均が全体で 0.95% 過少であった。しかし、港から近い場合、漁を行った場所を一部判別できていないため、面積が 100.00% 過少となっている場合がある。また、実際に漁を行った範囲よりも 18.40% 過大に漁を行っているとして判別している場所があったため、面積の計算精度の平均が全体で 13.71% 過少となった。漁場の遠近で計算精度のばらつきがあると言える。現在、漁船の航行速度を用いて漁を行った場所を判別していることから、航行速度に関連するプログラムを改善することで、漁港から近い場合における漁場の面積の計算精度を向上させることができると考える。面積の計算精度の平均が全体で 100% に近づけるようにする。

5.6.2 UI や UX の向上

我々が作成した Web アプリケーションは、年月日や漁船 id を変更するときに細かい操作が必要であったり、指定の日付や漁船番号に関連する航跡や漁の面積、漁獲量全てのデータを 1 つのページで確認できないため、ユーザーに複数回の操作を必要としている。しかし、ヒアリングでは琵琶湖漁業者から、日付を選択する細かい操作や何度も入力しなければならない複雑な操作は難しいという評価を受けた。このことから、Web アプリケーションへの入力回数を減らすために、1 度の操作ですべての情報がわかるようなデザインにすることや、印刷ボタンを設けることで、クリックするだけで全てのデータを印刷できるよう改善する。

5.6.3 Web アプリケーションの表示機能の追加

既存の Web アプリケーションでは、指定した年月日における全ての漁船の航跡や指定した漁船 id の航跡を表示していた。しかし、琵琶湖のヒアリングで漁業者から、漁獲量が増加したとき、翌日も同じ場所で漁を行うことを教えていただいた。そのため、指定した年月日だけでなく、指定した年月日の前日や前々日の航跡も同時に見られるようにしてほしいとの要望があった。このことから、前日や前々日の航跡を並列して表示することや、1 つのマップ上に重ねて表示できるようにする。

(※文責: 古明地颯天)

第 6 章 後期結果

6.1 後期プロジェクトの結果

1. プロジェクトの後期目標の設定
 - 前期の活動の成果及び達成度の確認を行った上で、適切なプロジェクトの到達目標の再設定を行うことができたため、達成できた。
 - プロジェクトの到達目標の再設定に伴うスケジュールの調整を適切に行うことができたため、達成できた。
2. 航跡データの分析
 - 前期に達成できなかった漁を行った面積を求めるシステムを実装することができた。漁港から遠い場合では、漁を行った計算精度の平均が全体で 0.95% 過少。漁港から近い場合では、漁を行った計算精度の平均が全体で 13.71% 過少であった。ただし、近い場合では、漁場の取り逃しや、過大に取っていることには注意が必要である。
 - 加えて、漁を行った距離も求めることができたため、達成できた。
3. Raspberry Pi に本番環境を構築
 - 本番環境として、Raspberry Pi にデータベースサーバー、API サーバー、Web サーバーを構築することができたため、達成できた。
4. データベースの更新
 - 航跡データの分析（漁を行った面積及び距離）の結果を格納するテーブルの追加を行うことができた。また、湖レコで収集している漁獲量データの格納を行うことができたため、達成できた。
 - 和田教授が収集している日付、時刻（UTC）、漁船 id、緯度、経度、航行速度、漁船の進路方向のデータを格納しているデータベースから、我々のグループのデータベースにデータを格納すること、時刻を JST に変換する前処理、前処理を行った航跡データの分析を自動で行う処理は実装できなかった。
5. API の作成
 - 7 種類の API に対して、統一されたルールに沿って設計及び実装を行うことができたため、達成できた。
6. Web アプリケーションの作成
 - 前期に Figma で作成したモックアップをもとに、Vue.js を用いて Web アプリケーションを要件通りに実装することができたため、達成できた。
7. Web サーバーの構築
 - Apache を使った Web サーバーを構築と Basic 認証の導入を行うことができたため、最低限の目標は達成できた。
 - 上記は他の目標の方が努力目標よりも優先順位が高いと判断したためである。
8. Web サーバーのログ取得
 - Apache のアクセスログを取得し、Discord で通知を行うシステムを実装することができたため、達成できた。

9. ヒアリングの実施

- 久米氏、奥村組合長、沖島漁業協同組合の漁業者3名へヒアリングを行うことができた。また、フィードバックや貴重なお話を聞くことができたため、達成できた。

(※文責: 古明地颯天)

6.2 成果の評価

6.2.1 Web アプリケーションの作成

前期に Figma で作成したモックアップをもとに、後期で定めた要件通りに航跡の可視化、漁の面積の可視化、漁獲量の可視化を行うことができた。前期のモックアップでは表示していなかった、湖マップの概要、操作方法、各ページの紹介などユーザーが使いやすいように情報を追加することができた。さらに各ページで見られる情報を減らし、一括で見やすいように湖マップのデザインを作成した。

そして、琵琶湖でのヒアリングにおいて「航跡の可視化は若手漁師の育成の役に立つ」という旨の肯定的なご感想をいただいた。しかし、同時にヒアリングを行った熟練漁業者にとっては「現在のアプリでは操作を覚えるのは難しい」という旨の意見をいただいた。

まとめると、後期で定めた要件を満たす Web アプリケーションを作成することができた。今後は、ご高齢な場合でも使いやすい UI/UX を目指すとともに、ヒアリングの中で要望のあった「前日までの航跡を重ねて表示する機能」を追加していきたい。

(※文責: 関崎証)

6.2.2 漁場の面積計算

前期に比べて後期は大きく進捗した点は二つ。楕円形の漁場をすべて判別できたことと、より精度の高い面積計算方法を用いることができたこと。また、QGIS を用いて簡易的ではあるが精度を求めて、自分たちの求めた計算結果がどのような時によいか、そしてどのような時精度が悪くなるかがわかったことで、今後の展望へとつながった。

一方で QGIS で精度を測る方法は完全だとは言えない。私たちの目で航跡を見て漁場だと判断した座標のみを切り取って正解データとして比較している為である。つまり、私たちが漁場の場所を誤って捉えれば、正解データとしての信用がなくなってしまう。また、人の手で正解データを作っている以上は精度を確かめられるデータ数もそう多くできない。ならば、現地の漁業者に漁をしているときに何らかの合図をしてもらう方法もあるが、それができるならそもそも自動で漁場を判別するシステム自体がいらなくなってしまうため本末転倒である。従って、現状では我々が行った方法でよいと結論になった。QGIS を用いた方法で、我々の面積を求めるシステムを評価すると漁港から遠い場合では、漁を行った計算精度の平均が全体で 0.95% 過少。漁港から近い場合では、漁を行った計算精度の平均が全体で 13.71% 過少であった。ただし、近い場合では、100.00% 過少と判断される漁場や、18.40% 過大に判断される漁場があることには注意が必要である。

(※文責: 松本葉太)

6.2.3 Web アプリケーションをホストする作業

サーバーとして Raspberry Pi を利用したため OS のインストールからソフトウェアのを動かすところまで全てを行った。難しい作業だったが今までの学びを活かし問題が起きても対処することができた点がよかったと考える。

最低限のセキュリティ対策として Basic 認証を設定した。時間はかかってしまったがすり抜けなどの問題は起こさなかったのがよかったと思う。

バックエンド班が作成した Python で書かれた Flask のコードを mod_wsgi を使用して Apache にホストすることを行った。最終的に実行できるようにはなったが、知識不足から設定がすぐにできずグループのメンバーに迷惑をかけてしまったのが良くない点であった。

フロントエンド班が作成したフロントのコードを Apache でホストした。最初シングルページアプリケーション向けの設定ができておらず再読み込みなどがうまく行かなかったが設定の見直しで対応することができた点がよかった点だと考える。

(※文責: 梶島司遠)

6.2.4 データベースの構築

データベースの構築は初めての作業であり学ぶことが多く大変な作業だった。作成したテーブルは完璧な状態とは言えないが今回のプロジェクトで実装した機能に限れば十分であったと考える。

(※文責: 梶島司遠)

6.3 担当分担課題の評価

6.3.1 関崎証

グループリーダー業務

メンバーの仕事分担・スケジュール管理・進捗管理を行った。ヒアリングに伺う日程までに開発と説明資料作成を終わらせることができた。

問題が起きた場合でも、柔軟に対応することができた。

Web アプリケーションの作成

前期に Figma で作成したモックアップをもとに、Vue.js を用いて Web アプリケーションの作成を行った。担当したのは航跡画面の実装、漁の面積画面の実装、漁獲量画面の実装である。良かった点として、機能の要件を定めた上でバックエンド班とコミュニケーションをとり、開発を進めることができたことがある。また、期限内に Web アプリケーションを完成させることができた。今後は、ヒアリングの結果である UI/UX の向上と「前日までの航跡を重ねて表示する機能」を追加していきたい。

(※文責: 関崎証)

6.3.2 椛島司遠

航跡のプロット

航跡のプロットをゴールデンウィーク中に完成させることができた。結果的に面積を求めるプログラムの作成に重要な機能になったため早めの完成でチームに貢献できたと考える。後期はサーバー全般に関する設定を行った。

データベースの構築

データベースの構築は他のすべての作業の前提となるため早急に構築する必要がある。プログラムの検証のために6月上旬にデータベースを構築できた。また、Pythonでデータベースの差分をアップデートするプログラムを書いた。

Raspberry Pi の設定

今まで個人的に行ったサーバー構築の知見を活かした点がよかった。

Apache の設定

今まで Apache を触ったことがなかったため作業に時間がかかってしまったが、無事に設定を終わらせることができたのでよかった。

API の設定

本番用に Flask 標準の機能ではなく Apache と mod_wsgi を利用して API をホストできてよかった。

(※文責: 椛島司遠)

6.3.3 古明地颯天

API

特定の URL にアクセスすることで、Web アプリケーションがデータを取得できるプログラムを書いた。反省点として、参照するテーブルを間違えていたため、Web アプリケーションでエラーが発生してしまったので、今後はプログラムを書いた後、見直してから Git に commit を作成するようにする。

ログの取得と出力

アクセスログを取得し Discord Bot を用いて出力できるようにした。全てを自動化することはできなかったが、プログラムを実行すればログファイルの再読み込みを行い、最新のデータを取得できるようにした。反省点として、エラーを解消できず1人で抱え込んでしまい、時間がかかったため、今後はサイトを参照するだけでなく、バックエンド班のメンバーにプログラムを見てもらうようにする。

成果発表会のスライド作成

琵琶湖漁業者へのヒアリングを滋賀県で行うときに用いたスライドをもとに作成した。図や表を取り入れることで視覚的にわかりやすくなるよう心がけた。反省点として、スライドに含まれている文字数が多かったので、今後は図や表を中心としたデザインかつ知識のない人でもわかる内容になるよう心がける。

(※文責: 古明地颯天)

6.3.4 松本葉太

漁場の面積計算

後期は面積の計算とその精度について求めた。面積を求める方法については前期末に構想を他メンバーが練っており、予定通り進めることができた。QGISによるシステムの精度の調査では、正解データとするには、漁場の形を手作業で入力しなければならないと気づくことができた。その過程で漁船の動きに特徴があるとわかった。

資料作成

後期は前期にはほとんどかわらなかつた最終成果発表用のポスターの一部制作や琵琶湖ヒアリング時の現地の人たちへの説明用資料の制作にも携わった。特に漁業協同組合の漁業者に向けた資料作成は難しく、安易に外来語を使わないなどの今までにない工夫をした。私たちが考えた面積計算の方法などを説明する難しさを実感した。最終成果発表のコメントから図などが多くてわかりやすかったと好評であったので成果は上げられたといえる。

(※文責: 松本葉太)

6.3.5 横田凌也

Web アプリケーションの作成

前期に Figma で作成したモックアップをもとに、Vue.js を用いて Web アプリケーションの作成を行った。担当したのは、各ページのデザイン全般および各デバイスへのレスポンス対応である。良かった点としては、各デバイスへのレスポンス対応を作成できたことである。また、期限内に Web アプリケーションを完成させることができた。今後は、ヒアリングの結果を元に UI/UX を向上し、どんな方でも難が無く使用できるデザインを設計し作成する。

ポスター作成

Figma を用いて発表ポスターの作成を行った。後期では、サブポスターの作成をメインに行った。前期のポスターの可読性を向上させるために文字サイズを大きくするなどデザイン関係を担当した。過去のポスターのデザインを参考にしながらデザインの構成を考え作成した。

(※文責: 横田凌也)

付録 A 新規習得技術

プロダクトの開発をするにあたって必要であると考えた言語と技術の習得することを目指した。技術は下記の5つである。

1. Vue.js

私たちは、航跡・漁場の面積・漁獲量を見ることが出来る Web アプリケーション「湖マップ」を開発するため JavaScript のフレームワークである Vue.js を習得した。加えて、Vuetify や vue-chartjs などの必要なライブラリも習得した。前期では Vue.js の環境構築、コンポーネントの理解、構文の理解、Vuetify を用いたコンポーネントの切り替えを実践することが出来た。後期では、vue-chartjs など必要なライブラリを使い、アプリケーションの機能実装を実践することが出来た。

2. Python

チームメンバー全員が Python を使用したプログラミングの経験があったため、バックエンドの処理に Python を選定した。具体的には航跡の描画、面積の計算、API に使用している。Python は様々な用途のライブラリが用意されており、今回のプロジェクトでは psycopg2、Flask、folium、geopandas などを新規で使用した。これらを通して複数人でプログラムを書く際の問題を知ることができた。また、ドキュメントを参考にしながらプログラムを書く能力を身に付けることが出来たと考えている。

3. PostgreSQL

授業で使用したことがあるため選定され、航跡データを保存及び読み込みするために利用した。航跡データは 100 万行を超える巨大なデータである。そのため毎回すべてを読み込むことは難しく、ある程度絞り込んで読み込みが出来るものが求められた。そのためデータベースを使用することにした。今回のプロジェクトを通して、テーブル設計の作りこみの大事さなどを学ぶことが出来た。

4. Git/Github

私たちは、ファイルのバージョン管理を行うためのツールである Git/Github の使い方を習得した。具体的には、commit、push、pull などの必要なコマンドを習得した。Git/Github は、実際の開発現場でも使われているツールである。そのため、Git/Github を習得できたことは価値があったと考えている。

5. LaTeX

私たちは、文書作成ツールである LaTeX を習得した。LaTeX の書き方や、数式の書き方を調べ、実践することができた。LaTeX は、今後卒業論文を書く場合にも使うことができるツールである。そのため、LaTeX を習得できたことは価値があったと考えている。

付録 B 活用した講義

1. データベース工学

データベースの基礎知識と PostgreSQL の使用方法について学んだ。本プロジェクトのデータベースは、この知識を元に作成した。

2. システム管理方法論

Apache と Basic 認証の導入について学んだ。本プロジェクトの Web サーバーの構築は、この知識を元に作成した。

参考文献

- [1] 関慎介 (2022) 漁業の見える化 技術継承と漁家営業の安定を目指した漁労技術の数値化. 和田雅昭 (編) スマート水産業入門. 緑書房, p60-61.
- [2] 滋賀県 (2023) 漁獲報告 WEB アプリ「湖 (うみ) レコ」の部屋. <https://www.pref.shiga.lg.jp/ippan/shigotosangyou/suisan/329441.html> (2023/06/30 アクセス)
- [3] 米川明彦 (2019) 平成の新語・流行語辞典. 東京堂, p183.
- [4] kcajminer2312, kerry_s (2023) WI-FI Issues. <https://forums.raspberrypi.com/viewtopic.php?t=349810> (2024/1/10 アクセス)
- [5] arkgame (2023) Ubuntu 23.04 Apache2 Basic 認証の設定方法. <https://arkgame.com/2023/10/01/post-323121/> (2024/1/10 アクセス)