

公立はこだて未来大学 2019 年度 システム情報科学実習 グループ報告書

Future University Hakodate 2019 System Information Science Practice
Group Report

プロジェクト名

ビーコン IoT で函館のまちをハックする - BEACON FUN Revolutions

Project Name

Leverage the Beacon IoT in Hakodate Real Downtown for Our Smarter Life

グループ名

去りし想ひを乗せゆきて

Group Name

Sarishi Omoi wo Nose Yukite

プロジェクト番号/Project No.

10-A

プロジェクトリーダー/Project Leader

渡邊慎太郎 Shintaro Watanabe

グループリーダー/Group Leader

松山穂乃夏 Honoka Matsuyama

グループメンバ/Group Member

松山穂乃夏 Honoka Matsuyama

宮越七菜 Nana Miyakoshi

石倉愛朗 Airu Ishikura

工藤大山 Daisen Kudo

指導教員

松原克弥 藤野雄一 鈴木恵二 鈴木昭二 奥野拓

Advisor

Katsuya Matsubara Yuichi Fujino Keiji Suzuki Sho'ji Suzuki Taku Okuno

提出日

2020 年 1 月 22 日

Date of Submission

January 22, 2020

概要

本プロジェクトは、ビーコンを函館の街の様々な場所に設置して、新たな体験や魅力を提供することを目的とする。ビーコンを用いることより、函館という地域に付加価値をもたらすことに重点をおいたサービスを提供する。どのようなサービスを提案するかアイデア出しを行うにあたり、フィールドワークによって函館の街の課題を洗い出した。洗い出した課題をもとに、函館らしさ・ビーコンである理由・新規性・継続したくなる魅力・必要性の5つの観点に重点をおいて7つのアイデアを創出した。さらに、それらをブラッシュアップすることによって4つのサービスを提案するに至った。

(※文責: 渡邊慎太郎)

函館には、長く保存されている歴史的建造物や200年以上の歴史を持つ路面電車がもたらすハイカラで美しい景観があり、趣のある街である。岩手県出身の歌人石川啄木(1886-1912)が函館を訪れた際、多くの歌を詠んだ。函館は、石川啄木が「死ぬのならば函館で」と手紙に残したほどの、特別な街である。しかし、函館市民に短歌を詠む習慣は浸透していないという現状があり、短歌を詠むという文化が廃れてしまう可能性がある。これらの背景を踏まえ、本グループでは函館の趣のある街並みを短歌として表現できるサービスを実施することで、函館市民にとって短歌をより身近なものに感じてもらえるようにする。サービス利用の一連の流れとして、まず、市電に乗車した人が、函館市電沿線の情景に対して短歌の第一句を詠む。次に他の人が短歌の第二句を詠む。この流れで、第三句、第四句、第五句まで詠むことで1つの短歌が完成する。前期の活動では、サービスの要件定義を実施した。後期ではこれを踏まえ、設計・実装を行った。その後、実際の成果発表の場でサービスを利用してもらう実証実験を実施。検証や意見を基に問題点や改善点を特定し修正を実施した。

キーワード 函館、路面電車、風景、広告、短歌、IoT、ビーコン

(※文責: 工藤大山)

Abstract

In this project, we aim to set up beacons in various places in the city of Hakodate to provide new experiences and attractions. By using beacons, we will provide services focused on bringing added value to the area of Hakodate. Before we came up with ideas for what kind of service to propose, we identified issues in the city of Hakodate by fieldwork. Based on the issues identified, 7 ideas were created with emphasis on the five viewpoints of the likeness of Hakodate, reasons for being beacons, novelty, attractiveness to be continued, and necessity. Furthermore, by brushing seven ideas up, we came to propose four services.

(※文責: 大野和也)

Hakodate is a quaint town with a beautiful landscape with a high color brought by a long preserved historical building and a tram with a history of over 200 years. When Iwaki Takuboku Ishikawa (1868-1912), who is from Iwate Prefecture, visited Hakodate, he wrote a lot of Tanka. Hakodate is a special city that Takuboku Ishikawa left behind in the letter “If I die, in Hakodate”. However, there is a problem that Hakodate’s citizens don’t have the habit of taking Tanka, and there is a possibility that the culture of putting Tanka will disappear. Based on these backgrounds, by implementing a service that can express the quaint cityscape of Hakodate as a Tanka, this group will make it possible for Hakodate citizens to be more familiar with Tanka. As a series of service usage, first, the person who got on the tram picks up the first phrase of the Tanka to the scenery of the Hakodate City Tram line. Then another person composes the second phrase of Tanka. In this flow, one Tanka is completed by crawling up to the third, fourth and fifth phrases. In the first semester, we decide defined of service requirements. In the second semester, we make design and implement. After that, a demonstration experiment was conducted in which the service was used in a place where actual results were announced. Problems and improvements were identified and corrected based on verification and opinions.

Keyword Hakodate, Tram, Landscape, Advertisement, Tanka, IoT, Beacon

(※文責: 工藤大山)

目次

第 1 章	本プロジェクトの活動と目的	1
1.1	背景	1
1.2	目的	1
1.3	ビーコンについて	1
第 2 章	グループ課題設定までのプロセス	2
2.1	プロセス概要	2
2.2	フィールドワーク	2
2.2.1	事前調査	2
2.2.2	フィールドワークレクチャー	2
2.2.3	実地調査	3
2.2.4	振り返り	3
2.3	サービスの考案	3
2.3.1	BS (ブレインストーミング) と KJ 法によるアイデア出し	3
2.3.2	OST (オープンスペーステクノロジー) によるアイデアの深掘り	4
2.3.3	アイデアのブラッシュアップ	4
2.3.4	新技術開発サロンとの情報交換会	4
2.3.5	テーマ決定	5
2.4	その他	6
2.4.1	ロゴ作成	6
2.4.2	ビーコンについての事前調査	8
2.4.3	Git/GitHub ワークショップ	8
2.4.4	昨年度のサービスの説明	9
第 3 章	提案するサービスについて	10
3.1	背景	10
3.2	目的	10
3.3	サービス概要	10
3.4	ビーコン使用の理由	11
3.5	利用方法	11
3.6	機能一覧	11
3.6.1	アカウントの作成・ログイン機能	11
3.6.2	ユーザページ	12
3.6.3	短歌の作成	12
3.6.4	短歌の閲覧	12
第 4 章	開発	13
4.1	技術習得	13

4.1.1	環境準備	13
4.1.2	夏季休業期間中の各メンバーの課題	15
4.2	開発手法	15
4.2.1	導入した手法	15
4.2.2	開発の流れ	16
4.2.3	効果, 課題, 解決策	16
第 5 章	システム仕様	19
5.1	システム構成	19
5.2	モバイルアプリケーション	19
5.2.1	iOS	19
5.2.2	Android	21
5.3	サーバサイド・アプリケーション	22
5.3.1	タイムラインに関する API	24
5.3.2	短歌投稿に関する API	24
5.3.3	ユーザに関する API	24
5.3.4	ビーコンに関する API	24
5.3.5	タグに関する API	24
5.4	デザイン	25
5.4.1	アプリデザイン	25
5.4.2	サービスロゴ	25
5.4.3	アプリアイコン	25
5.5	サーバサイド・インフラストラクチャー	26
第 6 章	成果発表会	27
6.1	中間発表	27
6.1.1	発表形式	27
6.1.2	発表技術の評価と反省	27
6.1.3	発表内容の評価と反省	27
6.2	公立はこだて未来大学オープンキャンパス	28
6.3	アカデミックリンク	28
6.4	成果発表会	29
6.4.1	発表形式	29
6.4.2	発表形式の評価と反省	30
6.4.3	発表内容の評価と反省	30
6.5	enPiT	31
6.5.1	発表形式	31
6.5.2	発表内容の評価と反省	31
第 7 章	各メンバーの振り返り	32
7.1	役割分担	32
7.2	松山穂乃夏の振り返り	32
7.3	宮越七菜の振り返り	34

7.4	石倉愛朗の振り返り	35
7.5	工藤大山の振り返り	36
第 8 章	まとめと展望	38
8.1	前期の振り返り	38
8.2	後期の振り返り	39
8.3	今後の展望	39
第 9 章	学び	40
9.1	情報をアウトプットする技術	40
9.2	サービス設計	40
9.3	振り返り	40
9.3.1	スケジュール管理	40
9.3.2	情報共有	41
付録 A	成果報告会で使用したプロジェクト概要のポスター	42
付録 B	成果報告会で使用した本グループのポスター	44
	参考文献	45

第 1 章 本プロジェクトの活動と目的

1.1 背景

近年, IoT (Internet of Things) の拡大に合わせて, 省電力無線通信である BLE (Bluetooth Low Energy) の利用が拡大している [1]. この BLE を利用したデバイスの 1 つに BLE ビーコンが存在する. BLE ビーコンは, BLE 対応デバイスを所持していれば誰でも利用可能であるため, 今後, ビーコンを活用したサービスの増加が期待できる.

(※文責: 瀧本恒平)

1.2 目的

本プロジェクトの目的は, ビーコンを使用して函館の街に新しい価値を創造することである. ビーコン IoT を用いたサービスの考案・開発・評価実験を行う.

(※文責: 瀧本恒平)

1.3 ビーコンについて

ビーコンとは, 電波を発して, それを受け取ることのできる機器で受信することにより, 位置をはじめとした情報を伝達するための機器である. このビーコンの 1 つに, BLE を用いて位置や情報を発信する機器がある. これを BLE ビーコンと呼ぶ. ビーコンは, 省電力・低コストで動作し, ビーコンの電波を受信する機器は, ビーコンそれぞれが所持する固有 ID や電波強度を取得することが可能である. また, BLE ビーコンにはいくつかの規格があり, 規格によって付加される情報が異なる. 例えば, 現在普及しているビーコンの規格として, 「iBeacon」・「Eddystone」・「LINE Beacon」等がある. iBeacon は, 固有 ID の送信や, 距離測定, 領域入出監視を行うことができる. Eddystone は, iBeacon でできることに加えて, URL の送信や, ビーコンのバッテリー電圧や温度等のデバイス自体が所持している情報を送信することができる. LINE Beacon は, BLE と iBeacon に対応し, LINE Beacon パケットをアダプタイズできるビーコン端末であり, コミュニケーションツールとして広く認知・活用されている「LINE」を通じて, 広告やプッシュ通知を送信することができる.

(※文責: 瀧本恒平)

第 2 章 グループ課題設定までのプロセス

2.1 プロセス概要

本プロジェクトのグループ課題設定までのプロセスは、大きく 2 つに分かれる。1 つ目は、フィールドワークに関わるプロセスである。実際に函館市内を見て回り、本プロジェクトで開発するサービスを考えるきっかけを探した。2 つ目は、サービスの考案に関わるプロセスである。フィールドワークや自身の体験を元に、アイデアを考えた。考えたアイデアについて、函館の地元企業の方、Tangerine 社、トランスコスモス社、担当教員、プロジェクトメンバーや OB・OG などに向けて、プレゼンを行った。そこで得られたコメントを自身のアイデアの改善の参考にした。その他にも、プロジェクトのロゴ制作や、開発に必要となる Git/GitHub のワークショップや、OB・OG からの昨年度のサービスの説明などが挙げられる。ここからは、フィールドワーク、サービスの考案、その他の 3 つに分けて、詳細を説明する。

(※文責: 吉川綾乃)

2.2 フィールドワーク

2.2.1 事前調査

函館における課題を調査するためにフィールドワークを計画した。まず、調査内容と調査地区を KJ 法によるアイデア出しによるによって挙げた。次に、挙げられたものから五稜郭地区・赤川地区・湯の川地区・西部地区についてフィールドワークを実施すると決定した。決定した調査地区を対して、五稜郭地区・赤川地区を担当するグループ、湯の川地区を担当するグループ、西部地区を担当するグループを決定し、それぞれのグループでおおまかな調査内容を決定した。

(※文責: 齊藤由樹)

2.2.2 フィールドワークレクチャー

フィールドワークの実施にあたり、5 月 10 日 (金)4 限の時間にて南部美砂子先生よりフィールドワークについてレクチャーを受けた。レクチャーでは、フィールドワークをする上での心得として単に見たものや聞いたものをメモするだけではなく、情報の検索、処理、相対化、生産、展開の 5 つの手続きを意識することが大切だと学んだ。また、本プロジェクトにおけるフィールドワークのアプローチとしては、問題そのものを探索する問題発見型であることを認識した。フィールドワーカーは訪問先にとってあくまで邪魔者であるので、謙虚さ、誠実さ、感謝を忘れないことが重要であるということを学んだ。

(※文責: 川谷知寛)

2.2.3 実地調査

実地調査は、2.2.1 節で述べた地区を 3 日の日程で実施した。五稜郭地区は、五稜郭公園、五稜郭タワー、シエスタ・ハコダテ、五稜郭交差点地下横断歩道、市電、千代台公園へ調査を行った。赤川地区は、四季の杜公園、笹流ダムへ調査を行った。湯の川地区は、湯倉神社、足湯、熱帯植物園、コーヒールームきくち、市民の森トラピスチヌ修道院、函館牛乳あいす 118、函館空港へ調査を行った。西部地区は、緑の島、北方民族資料館、八幡坂、元町公園、旧函館区公会堂、中華会館、旧イギリス領事館、赤レンガ倉庫、西埠頭、函館どつく、函館山、立待岬、函館公園へ調査を行った。実地調査中は、普段からある当たり前なことに疑問を持ち、細かなことにも注目して課題を発見することを心がけた。調査の記録や情報の共有には Slack を用いた。訪れた地区ごとのチャンネルを作成し、気づいたことや写真などを共有した。Slack での発言はどんな些細なことでもすることとし、より多くの情報を得られるようにした。

(※文責: 石川空人)

2.2.4 振り返り

調査の結果の共有を行うため、各々がフィールドワーク中に気付いた点を付箋に書き出し、模造紙に貼り付けた。その後、出た意見について KJ 法を使用してグループ分けを行い、そのグループ間の関連性を、矢印と理由で具体的に書き出した。書き出した気づきとその関連性のグラフ図を、その後のアイデア出しの材料とした。5 月 22 日 (水) には、2.2.3 節で記載した 8 箇所を訪れた。調査の結果、平日の昼間の市電は高齢者の乗車が多かったこと、本町という括りの中でも、場所によって人の多さがかなり異なっていたことに気付いた。各班滞りなくフィールドワークを進め、気づきをアウトプットする時間もしっかりと確保することが出来た。5 月 25 日 (土) には、2.2.3 節で記載した 7 箇所を訪れた。調査の結果、温泉や足湯が多くあること、教会や神社等の趣のある建造物の存在を確認できた。5 月 26 日 (日) には、2.2.3 節で記載した 13 箇所を訪れた。調査の結果、函館に縁のある作家の存在や、函館山登山道を通る人達のコミュニケーションがあることを確認した。全体を通して多くの気づきがあり、函館というまちについて改めて知ることができた。

(※文責: 松山穂乃夏)

2.3 サービスの考案

2.3.1 BS (ブレインストーミング) と KJ 法によるアイデア出し

フィールドワーク後のブレインストーミングを踏まえて、BS と KJ 法によるアイデア出しを行った。BS とは思いつく限りの多くの量のアイデアを出すことで、KJ 法とはそれらを付箋等にアウトプットして得られたアイデアを整理し、グルーピングを行う方法である。今回は、3 人を 1 組とするグループを 7 つ作り、各グループ内で 15 分間アイデア出しのブレインストーミングと KJ 法によるアイデアのグルーピングを行い、その後の 10 分間で他のグループのアイデアをレビューした。アウトプットの方法としては、7 つのグループに模造紙を 1 枚ずつ配布し、アイデアを付箋に書いて貼り付けるかたちで行った。

(※文責: 熊谷峻)

2.3.2 OST (オープンスペーステクノロジー) によるアイデアの深掘り

BS と KJ 法によるアイデア出しを行った後に、各プロジェクトメンバが興味のあるアイデアについて深めるために OST を行った。OST とは各々が関心を持ったテーマについて考え、より深い意見にするための方法である。時間を区切ることはせず、各々が関心を持ったアイデアについて考え、考え終わったら他のアイデアに移り各アイデアについて考えを深めていった。OST の結果、一人一人が好きなアイデアを自由に話し合うことができた。しかし、OST の短所として、話し合うことができるほどの人数が集まらなかったアイデアが自然消滅してしまうことがあったため、ほかの方法でのアイデア出しも必要であると感じた。最終的に、OST によって深められたアイデアを話し合いや投票によって 7 つに絞り込むことができた。

(※文責: 熊谷峻)

2.3.3 アイデアのブラッシュアップ

アイデア出しの後いくつかのプレゼンテーションや評価を行い、5 つのアイデアに絞った。またそれぞれのアイデアに興味のあるプロジェクトメンバーを偏りの少なくなるよう振り分け、各グループの詳細なサービス案についてブラッシュアップを行った。教員や学生間でのレビューを行いつつスライドに内容をまとめ、プロジェクトメンバー内でプレゼンテーションを行った。評価の方法として各アイデアに対して「函館らしさ」「ビーコンらしさ」「新規性」「継続性」「必要性」の観点について意見出しを行い、アイデアをさらに絞った、最終的なアイデアは 2 つでそれらの改善を行った。

(※文責: 海老原天紀)

2.3.4 新技術開発サロンとの情報交換会

6 月 19 日 (水) に函館周辺地域の企業の方々が開催する、新技術開発サロンとの情報交換会に他プロジェクトとともに参加した。まず参加企業の会社概要と事業内容の説明を受けた。ソフトウェア開発を行う会社から造船業まで幅広い会社の話を聞くことができる貴重な機会であり有意義であった。次に、プロジェクト内で考えたアイデアの発表・質疑応答を行った。発表ではビーコンプロジェクトの概要やビーコンの説明、実現するもの、利用例を紹介した。その後今年度の活動予定と、発表時点で構想段階であった 2 つのアイデアである、ぶれいるーむと函ライブについて発表を行った。すべての発表が終わった後、アイデアに関する質疑応答の時間を設けた。この中で、参加企業の方々から様々な質問や意見を頂くことが出来た。発表会自体は成功に収まったが、アイデアの決定にかなりの時間を要するなど、準備に大変手間取った。このため、発表会で十分な情報を提示することが出来ず、適切な意見をもらうことが出来なかった可能性があると考えている。これを踏まえ、これ以降の活動では意思決定をすばやく行うことを心がけるようにした。

(※文責: 工藤大山)

2.3.5 テーマ決定

新技術開発サロンとの情報交換会を終えて、新たにアイデア出しを行った。その後、担当教員、TA、函館市職員、Tangerine社、トランスコスモス社の方々に参加してもらい、アイデアコンテストを行った。アイデアコンテストでは「みみうち」「ゆまち」「山で仏に会ったよう」「Mystery Films」「函ライブ」「ふれいるーむ」「去りし想ひを乗せゆきて」の7個のアイデアをプレゼン、採点を行った。裏メニューの情報を提供するサービスである「みみうち」、温泉での待ち合わせを円滑に行うサービスである「ゆまち」、函館山の地蔵から情報を受け取るサービスである「山で仏に会ったよう」、函館市電の歴史を舞台とした脱出ゲーム「Mystery Films」、病院での複数人でのコミュニケーションを可能とする「ふれいるーむ」、市電で共同で短歌を詠む「去りし想ひを乗せゆきて」。各アイデアの採点には、「函館らしさ」「ビーコンらしさ」「新規性」「継続性」「必要性」の5つの観点で0~5点の6段階評価を行った。採点の結果は表2.1に示す。アイデアコンテストでの意見としては、以下が挙げられる。

- みみうち：裏メニューのためにアプリを入れたくなるのか疑問に思いました。ユーザが使いたくなるモチベーションがもう少し欲しいです。
- ゆまち：まずは待ち合わせに特化して具体的に必要な技術を調べてみて。
- 山で仏に会ったよう：案としては面白いが、あまり地蔵をピックアップしている意味がないように感じました。地蔵はあくまで、道筋の指標にしかならないのかなと感じました。
- Mystery Films：ゲームのコンテンツ内容がサービスの鍵になると思います。
- 函ライブ：個人的には面白そうだと思うのですが、一般的にはイメージしにくいので、やってみなければわからないタイプのアイデアだと思います。路上ライブの土壌をサイレントライブで作るってのは面白いと思います。このサービスを使ってくれる人とやりたい人の両方が揃ったら行ってみたいですね。
- ふれいるーむ：ビーコンが必要ないというかビーコンではできないアイデアだと思います。内容や対象自体は非常に悪くないと思います。
- 去りし想ひを乗せゆきて：アイデアはとても面白いので、ビーコンである意味をもう少し推せると良いと思った。

その後、アイデアコンテストでの意見を踏まえて、各アイデアでブラッシュアップを行った。その後、自分たちがどのアイデアの開発に携わりたいかという観点で投票を行い、「みみうち」「ゆまち」「函ライブ」「去りし想ひを乗せゆきて」の4個のアイデアを今年度のプロジェクト学習で開発することを決定した。

(※文責: 吉川綾乃)

表 2.1 各アイデアの評価の平均

アイデア名	函館らしさ	ビーコンらしさ	新規性	継続性	必要性
みみうち	2.6	3.0	3.3	3.3	2.7
ゆまち	3.3	3.6	4.0	3.2	3.3
山で仏に会ったよう	2.9	2.8	3.1	2.0	2.1
Mystery Films	3.4	3.2	2.9	2.8	2.3
函ライブ	2.2	3.5	3.0	2.3	2.3
ふれいるーむ	1.9	1.9	3.3	3.0	3.1
去りし想ひを乗せゆきて	3.6	3.3	3.0	3.6	3.0

2.4 その他

2.4.1 ログ作成

本プロジェクトでは、プロジェクトの特徴、目標、活動、呼び名などプロジェクトのアイデンティティについてチームで考え、今年度のプロジェクトを印象付けるロゴの製作を行った。ロゴ製作については、2ヶ月の時間を要した。初めに、各自で1案以上のロゴ案を持ち寄り、ロゴ検討会を行った。ここでは、それぞれのロゴ案に対し、良いと思った点だけを述べていくレビューを行った。レビューと他のメンバーのデザインも参考にして、各自でロゴの改善を行い、今度は改善点の指摘を含めた2回目のレビューを行った。その後、改善した17デザインを、投票により7つのデザインに絞り込んだ。(図 2.1, 図 2.2, 図 2.3, 図 2.4, 図 2.5, 図 2.6, 図 2.7)。7つのデザインそれぞれのコンセプトを把握し指導教員からのコメントも頂いた上で、2度目の投票を行い、2つのデザインに絞り込んだ。さらに改善を行い、函館らしさ、ビーコンらしさも考慮した上でメンバーとOB、OG、指導教員も加えた3度目の投票を行い、ロゴ原案を決定した。その後、ロゴのワーキンググループを結成し、デザイン原案の改善を行った。最後に、ロゴデザインの最終版とロゴ使用に関するガイドライン(図 2.8)、それをういたポスターのテンプレート(図 2.9)を作成した。



図 2.1 ログ案 1



図 2.2 ログ案 2



図 2.3 ログ案 3

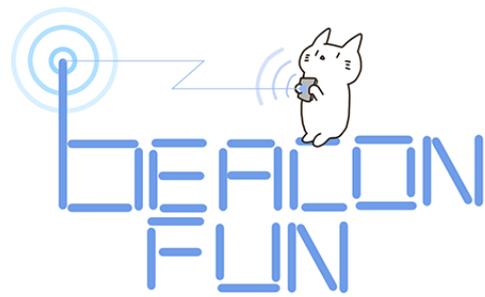


図 2.4 ログ案 4



図 2.5 ログ案 5



図 2.6 ログ案 6



図 2.7 ログ案 7



図 2.8 ログガイドライン

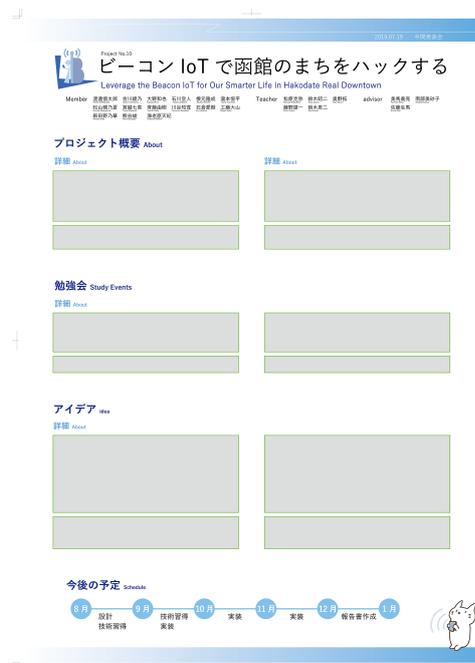


図 2.9 ポスターテンプレート

(※文責: 新田野乃華)

2.4.2 ビーコンについての事前調査

本プロジェクトでビーコンを使用するにあたり、ビーコンに関する知識を深く知るために、各自ビーコンについて調べた。さらに、5人1グループを3グループ作り、それぞれのグループでビーコンに関するスライドを作成し、全員の前で発表した。各グループのスライド発表を聞くことで、ビーコンの基本情報やビーコンの特徴、ビーコンを使用している既存サービス、ビーコンの利用場所、ビーコンの課題等を知ることができた。ビーコンの課題として、専用のアプリをインストールしないと機能しないことやユーザー側は電波を受信するために Bluetooth 機能をオンにしなければいけないこと、受信する端末の機種依存してしまうこと、仕切り・パソコンのディスプレイ等の遮蔽物や反射物で電波が遮られること、通知が煩わしいこと、盗難リスクがあることが挙げられた。ビーコンについて事前調査することにより、その後のアイデア出しではユーザー目線のアイデアやビーコンの特徴を活かしたアイデアを出すことができた。

(※文責: 石倉愛朗)

2.4.3 Git/GitHub ワークショップ

バージョン管理システムである Git と、Git ホスティングサービスである GitHub を用いたシステム開発・管理手法を本プロジェクトの TA から学んだ。まず、Git/GitHub とはどのようなサービスなのか説明があり、各 PC に Git のインストールを行った。続いて、GitHub のアカウント登録、Git で管理するディレクトリ・ファイルの作成、Git の初期設定、ブランチの作成、ブランチのチェックアウト、コメント付きでのコミット、プッシュ、プルリクエストの一連の流れを実行し

た。また、敢えてコンフリクトを起こすことでコンフリクトの修正方法や、新しいブランチを作成する際の良いブランチ名を学んだ。さらに、プロジェクトのアジェンダや議事録を GitHub のリポジトリを作成し、issue を立てることで、Markdown 形式の記法を学び、GitHub の機能をより深く知ることができた。

(※文責: 石倉愛朗)

2.4.4 昨年度のサービスの説明

今年度のプロジェクトでは、どのようなサービスを開発していくか、アイデア出しで出た案を選ぶ際の判断材料に利用するために昨年度のサービスの説明を受けた。昨年度のサービスの代表者に「@ハナセル」、「becoma」、「いさりび Graffiti」、「Telepath」の4つのサービスについて説明してもらった。「@ハナセル」は銅像を介して歴史的な偉人と会話できる新体験を提供できるサービスである。「becoma」はビーコンを活用し近くの人とマッチングを行い物を手放したい人と欲しい人をつなげるサービスである。「いさりび Graffiti」はビーコンを利用し列車自体をノートとしてメッセージや写真を投稿でき、列車毎の思い出ノートをみんなで作り上げるサービスである。「Telepath」は講義中や会議中で立場や周りが静かな時に気軽に質問できるサービスである。サービスの説明に加えて昨年度のアイデア決めにどのように行ったかなども説明を受けた。説明を受けたことで「いさりび Graffiti」のような、函館市内にビーコンを設置しサービスを運用した地域密着型のアイデアが出やすくなった。したがってプロジェクトのテーマに沿った評価の高いアイデアを選出することにつながった。

(※文責: 根元隆成)

第 3 章 提案するサービスについて

3.1 背景

1859 年に日本最初の対外貿易港として開かれたことをきっかけに、函館は異文化交流が盛んな街として発展してきた。現在の函館にも、長く保存されている歴史的建造物が数多くある。加えて、これらは 160 年以上の歴史を持つ路面電車の沿線に多く立ち並び、車窓からはハイカラで美しい情景を眺めることができる。このような趣のある街並みをもつ函館で、1907 年 5 月より 4 ヶ月間、歌人石川啄木は暮らしていた。石川啄木は先述した函館の雰囲気魅了され、多くの歌を詠んだ。そのことから、函館は石川啄木の「心のふるさと」と呼ばれている [2][3]。しかし現代の函館では、短歌を詠む文化に馴染みがなく、この文化が廃れてしまう可能性がある。そこで、函館市民に馴染みの深い市電と短歌をかけ合わせたら短歌を詠むきっかけになるのではないかと考える。

(※文責: 石倉愛朗)

3.2 目的

本グループの目的は、本サービスを通して、他の人とともに函館市内の 1 つの情景に向けて、1 つの短歌を複数人で詠み、想いや考えを共有すること、函館市民に短歌を楽しんでもらい、もっと身近に短歌や函館の情景に触れてもらうことである。さらに、自分が一部を詠んだ短歌が完成した際に、どのような短歌になるかなど、市電に乗車することを楽しみを付加する。

(※文責: 石倉愛朗)

3.3 サービス概要

本サービスは空間、景色を共有した人と、想いをつなぐ短歌リレーを提供するサービスである。函館のある 1 つの情景に対し、ある人が第一句を詠み、それに続き、複数の人で第五句まで詠み上げ、1 つの短歌を完成させる。手軽に言葉を投稿することができ、その後、短歌が出来上がるまで楽しむことができる。自分が新しい短歌を詠み出す、他の人が作った短歌の続きを詠む等の複数の楽しみ方がある。また、同じ情景を複数の人が同じ市電内から見て、短歌を詠み、閲覧するために、ビーコンを使用し、市電に乗車中かを判断する。さらに、市電の区間ごとに違う短歌を見られるため、その区間や区間沿線付近から見た情景の短歌のみを閲覧することができる。これらのサービスにより、気軽に短歌に触れること、他人の函館の情景に対する様々な感性を感じられること、自分が知らなかった函館の新しいスポット等を知ることなどが可能となる。

(※文責: 石倉愛朗)

3.4 ビーコン使用の理由

本サービスは市電から見た函館の街並みを共有し、想いを短歌として共に詠むサービスである。そのため、ユーザが市電に乗り込んでいる時のみ全てのアプリの機能が利用可能となる。それを実現するためには、ユーザが市電に乗り込んでいるか否かの判別が必要になる。そこで、ビーコンを市電内に設置し、乗車の判別を行った。ビーコンは位置情報を取得する GPS よりも検知の範囲が狭く、検知の精度も高い。また、屋内での使用も可能であり、市電のような動き続けるものに乗り込んでいる人が、市電という空間内にいるかどうかの判別に適しているからである。

(※文責: 松山穂乃夏)

3.5 利用方法

このサービスでは短歌を詠む際に、第一句を詠む場合と第二句以降を詠む場合が存在する。

まず、前者を想定してユーザストーリーを述べる。ユーザは市電に乗り込んで、車窓から函館の趣ある街並みを眺める。そこで本サービスを使用し、窓からの景色を見て感じたことを、短歌の第一句として投稿する。アプリケーションの使用を市電内に限定することで、同じ場所を通り、同じ景色を見た人と、短歌を通じたコミュニケーションが可能となる。この時、作成した短歌を表示する画面の背景色や文字色をカスタマイズすることで、自分の感情や作成した短歌のイメージの情報を追加することができる。また、作成した短歌に関連する単語のタグを付けることができる。タグを付けることで、その後の句を詠む人に自分の見た景色の情報をさらに伝えることができる。加えて、続きを詠む短歌を探している人が、今見ている景色に合致する短歌を見つけやすくなる。後日、市電に乗り込んだ際にアプリケーションを開くと、それまでに同じ情景を見た別ユーザによって続きが詠まれていれば、短歌が完成している。同じ情景を眺めた複数の人と共にひとつの短歌を完成させることで、共同作業や共感による喜びや、自分が思いもよらなかった短歌が出来上がる楽しみを得ることが可能となる。

次に、第二句以降を読む場合のユーザストーリーを述べる。ユーザは市電に乗り込んでアプリを開き、短歌がランダムに表示されるホーム画面や、タグの検索から続きを詠みたい未完成の短歌を探す。同じ情景を眺めた人と共に短歌を詠むことが目的であるため、この時続きを詠むことができる短歌は、現在ユーザがいる区間と同じ区間で第一句が詠まれた短歌のみである。続きを詠む未完成の短歌を選択すると、次の句を詠む画面に遷移し、続きを詠むことができる。

(※文責: 松山穂乃夏)

3.6 機能一覧

3.6.1 アカウントの作成・ログイン機能

ユーザがアプリケーションを使用する際、初回起動時には使用するユーザ名の登録を行う必要がある。ユーザ名の登録時にユーザ ID の発行が行われる。このユーザ名とユーザ ID を利用して短歌の投稿や閲覧を行う。

(※文責: 松山穂乃夏)

3.6.2 ユーザページ

ユーザページには、ユーザネームとアイコンの情報が表示される。また、自分が第一句から第五句のいずれかを詠んだ短歌を閲覧することができる。

(※文責: 松山穂乃夏)

3.6.3 短歌の作成

ビーコンの検知中、ユーザは市電の窓から見える景色に想いを馳せ、同じ景色を共有した他の乗客と共に短歌を詠むことができる。このサービスでは、一首全てを1人で詠むのではなく、短歌の第一句から第五句までを複数の人と詠む。

(※文責: 松山穂乃夏)

3.6.3.1 短歌の第一句を投稿

短歌を作成する際には、ユーザが短歌の第一句を詠む場合と、他のユーザが詠んだ第一句に続く句を詠む場合に分けられる。この節では前者の説明を行う。ユーザは車窓から見える景色に対して、感じた想いを自由に短歌の第一句として投稿する。投稿した短歌は、ユーザページから閲覧することが可能となる。短歌の投稿に加えて、短歌を表示する画面の背景色と文字色を、込めた想いに合わせて変更できる。また、詠んだ情景に合致する単語でタグを付けることで、短歌で詠んだ言葉だけでは伝えきれない想いや情景を追加できる。

(※文責: 松山穂乃夏)

3.6.3.2 短歌の第二句以降を投稿

ユーザがアプリケーションを起動すると、未完成の短歌や完成済みの短歌が表示される。その中から気に入った未完成の短歌を選び、その続きを詠むことができる。ユーザが詠んだ短歌は、ユーザページから閲覧することが可能となる。

(※文責: 松山穂乃夏)

3.6.4 短歌の閲覧

ユーザがいる場所に関わらず短歌を閲覧することができる。完成、未完成の短歌は分かれて表示されている。タイムライン上では第一句を詠んだユーザのユーザネームと、第二句が詠まれた年月日が表示されている。短歌をタップして詳細画面に遷移すると、先述した2つの情報に加え、第二句以降を詠んだユーザのユーザネームと、短歌につけられたタグ、短歌の詠まれた区間の情報が表示される。

(※文責: 松山穂乃夏)

第 4 章 開発

4.1 技術習得

本グループのメンバーは iOS アプリケーション開発と Android アプリケーション開発の経験を有していたため、本アプリケーションは iOS と Android に対応とした。以下に利用したツールを紹介する。

(※文責: 工藤大山)

4.1.1 環境準備

チーム開発では、チーム間でのコミュニケーションやアイデア・ソースコードの共有が不可欠である。以下に利用したツールを示す。

4.1.1.1 Slack

各メンバーや教員、TA との連絡にチームコミュニケーションツールである Slack を用いた。Slack には、他のアプリケーションとの連携手段が用意されていたり、ファイル共有やアップロードを容易に行うことができるという利点がある。

(※文責: 工藤大山)

4.1.1.2 Adobe Photoshop

アプリケーション開発にあたっては、アイコンやロゴ、ボタン等のパーツの作成が必要である。これらの作成にはアドビシステムズ株式会社が提供する Adobe Photoshop と、後述する Adobe Illustrator を使用した。Adobe Photoshop は主にラスター画像に対して編集を行うことができるビットマップ画像編集アプリケーションソフトウェアである。

(※文責: 工藤大山)

4.1.1.3 Adobe Illustrator

アプリケーション開発にあたっては、アイコンやロゴ、ボタン等のパーツの作成が必要である。これらの作成にはアドビシステムズ株式会社が提供する Adobe Illustrator と、先述した Adobe Photoshop を使用した。Adobe Illustrator は主にベクター画像に対して編集を行うことができるベクターイメージ編集アプリケーションソフトウェアである。

(※文責: 工藤大山)

4.1.1.4 Adobe XD

アプリケーション開発にあたって、プロトタイプと呼ばれるアプリケーション UI デザインを作成し、メンバー間で共有を行う必要がある。このプロトタイプ作成にはアドビシステムズ株式会社が提供する Adobe XD というプロトタイプ作成ツールを用いた。今回は Adobe XD を利用したプロトタイプ作成に長けたメンバーがおり、プロトタイプ作成に必要なコストが低いことから、Adobe XD を用いた。

(※文責: 工藤大山)

4.1.1.5 Zeplin

先述の Adobe XD で作成したプロトタイプの共有には Zeplin 社が提供する Zeplin を使用した。Zeplin はブラウザ上で動作するプロトタイプ閲覧ツールであるため、スマートフォンでプロトタイプの確認ができるという利点があるため、Zeplin を使用した。

(※文責: 工藤大山)

4.1.1.6 Git

ソースコードのバージョン管理ツールとして Git を使用した。Git は分散型のバージョン管理システムの 1 つであり、リポジトリ管理下である全てのファイルの変更履歴とともに完全なファイルのコピーを保存することができる。そのため、編集履歴を表示してどのような編集が行われたか確認したり、編集履歴を基にファイルを以前の状態に戻すことが可能である。リポジトリにはメンバーそれぞれのローカル環境に保存されているローカル環境とネットワーク上のストレージに保存されているリモートリポジトリの 2 種類がある。リモートリポジトリでは各メンバーがファイルに対して行った編集履歴が保存されており、編集履歴の確認や共有が可能である。また、Git には編集履歴を分岐させ、同時進行でファイルの編集ができる branch という概念が存在する。

(※文責: 工藤大山)

4.1.1.7 GitHub

GitHub とはリモートリポジトリホスティングサービスである。GitHub を使用することにより、複数のメンバー間でスムーズにファイルや編集履歴を共有することが可能である。GitHub の主要な機能の 1 つとして Pull Request がある。これは分岐した branch でファイルを編集した後、それを分岐元の branch に取り込む際に使用する。Pull Request を使用することで、いつ誰がどのファイルの編集をなぜ実施したかを把握でき、他の開発者がその編集に対してレビューを簡単に実施することができる。これによって本サービスでは、グループ間で編集履歴の可視化を行うことができたため、より品質の高いコードを作ることができた。また、GitHub には Issues という機能がある。これはプロジェクトやソースコードなどのファイルに対する課題などを管理することができる。Issues を用いることによって現状の課題やタスクを整理し、共有することができる。

(※文責: 工藤大山)

4.1.1.8 Docker

Docker とは OS レベルの仮想環境を提供するオープンソースソフトウェアである。それぞれの仮想環境を Docker コンテナと呼ぶ。サーバサイド・アプリケーション開発での環境統一と導入手順の簡易化のために使用した。

(※文責: 工藤大山)

4.1.2 夏季休業期間中の各メンバーの課題

本グループでは夏季休業中、後期の開発が円滑に進むよう、各自必要な技術習得を行った。互いの状況がわかるよう、3日に1度各々の行ったことを文章にまとめて共有した。夏季休業中に行った各々の課題は以下に示す。

松山穂乃夏

- XD のチュートリアルを行いプロトタイプの作成方法を学んだ
- サービスの UI フロー図とモックアップの作成を行った

宮越七菜

- Go のチュートリアルを用いて基礎的な学習を行った

石倉愛朗

- 「詳細！ Swift4 iPhone アプリ開発入門ノート」 [4] を読み知識を得た
- Swift のチュートリアルを通して詳しく学んだ

工藤大山

- Kotlin Bootcamp [5] のサイトを通して学習を行った。

(※文責: 松山穂乃夏)

4.2 開発手法

4.2.1 導入した手法

4.2.1.1 アジャイル開発

本プロジェクトでは、システムやソフトウェア開発の手法の1つである、アジャイル開発を導入した。アジャイル開発とは、大きな単位でシステムを区切ることなく、小単位で実装とテストを繰り返すことにより、臨機応変に対応することができるため、継続して動くソフトウェアを作り続けることができる。そのため本プロジェクトでは、この開発手法を採用した。

(※文責: 瀧本恒平)

4.2.1.2 スクラム

本プロジェクトでは、アジャイル開発手法の1つである、スクラムを採用した。スクラムとは、反復を繰り返す開発プロセスのことを指し、この反復の単位をスプリントと呼ぶ。スプリントで行う内容としては、スプリントプランニング、デイリースクラム、スプリントレビュー、スプリントレトロスペクティブ、開発作業などがある。スプリントプランニングとは、スプリントの開始に先立って行われる計画ミーティングを指す。このミーティングでは、プロダクトバックログと呼ばれるプロダクトへ追加する要求のリストからスプリントバックログを作成する。スプリントバックログとは、プロダクトバックログの項目を完了させるために必要なすべての作業をリスト化したものである。また、デイリースクラムとは、開発チームの全員が活動状況を共有し、前回のデイリースクラム以降に行った作業と、次回のデイリースクラムまでに行う作業を確認することをいう。これは、毎日決まった時間に決まった場所で15分程度行われる。また、スプリントレビューとは、スプリント終了時、チームの関係者に完成したプロダクトのデモンストレーションを行うことをいう。開発した機能が正しく動作しているか、方向性が間違っていないか等の確認をここで行う。また、スプリントレトロスペクティブとは、スプリントレビューの後に行われる、振り返りの機会のことをいう。ここでは、今回のスプリントでうまくいったこと、うまくいかなかったこと、どうすれば次のスプリントで改善できるかといったことを話し合う。本プロジェクトでは、1週間をスプリントのタイムボックスとして、スクラムを行った。

(※文責: 瀧本恒平)

4.2.2 開発の流れ

本プロジェクトでは、スプリント期間を1週間として開発を行った。毎週水曜日をスプリント開始日としてスプリントプランニングを行った。そして翌週の水曜日にスプリントレビューとスプリントレトロスペクティブを行った。本サービスでは、プロダクトバックログを付箋と模造紙で管理し、スプリントバックログはZenHubを用いて管理した。スプリント期間中では、金、日、火曜日に15分間のデイリースクラムを実施した。デイリースクラムでは、「前回のデイリースクラムから今日まで行ったこと」、「これから行うこと」、「困っていること」を共有し、解決策を議論した。スプリント中の開発ではGitHubを用いて開発した。スプリントレビューでは、他チームに開発した機能のデモを見せ、フィードバックを得ることで機能の品質を確認した。スプリントレトロスペクティブでは、KPTという振り返りの手法を用いて行った。KPTとはKeep, Problem, Tryの略である。Keepでは「今後も続けたいことや良かったこと」、Problemでは「うまくいかなかったことや課題」、Tryでは「今後行うこと」であり、この3つを付箋に書き出してチームの課題を発見して改善計画を立てることでスプリントの振り返りを行った。

(※文責: 宮越七菜)

4.2.3 効果, 課題, 解決策

4.2.3.1 効果

デイリースクラムを行い、定期的にチームメンバーの進捗状況を確認し、開発で詰まっているところや、困っているところを共有することができることで、チーム内の問題を早期に発見すること

ができた。例えば、アプリの UI デザインの作成に悩んでいるチームメンバーを発見して他のメンバーが支援したり、全体的に作業が遅れていることに早期に気づくことができたなどの成果があった。また、1 スプリントごとに実際に動かせるものを作ることで、チームメンバーのモチベーションを保つことができ、本当に必要な機能のみを実装することができた。開発の後半には、チームメンバーだけでなく、他のチームのメンバーにも 1 スプリント毎に作ったものをデモで紹介することでレビューをもらった。それによって、特にアプリの UI デザインを使いやすいデザインに改善し、より完成度の高いものを開発することができた。他にも、1 スプリントの最後に振り返り（スプリントレトロスペクティブ）を行うことで、良かったところと悪かったところをチーム内で共有、認識することができた。チームの課題や継続すべき点を浮き彫りにすることで次のスプリントをより効率的に行うことができるようになった。実際の例として、全員が進捗を上げることができたことが良かった点として、特に作業を楽しみながら行うことができず、モチベーションが低下していたことが悪かった点としてあげられ、良い点はそのままに、悪い点は新しく学んだ技術を自分で好きに応用して使ってみることで楽しみをつくって改善できたチームもあった。

全体として、チームの早期問題発見や短い期間での振り返りによる作業の効率化といった効果があげられた。

(※文責: 熊谷峻)

4.2.3.2 課題

アジャイル開発手法の一つであるスクラム開発を取り入れることで、数々の効果が出たが、同時に課題も見つかった。まず、デイリースクラムに関してだが、これは決められた日に決められた時間で必ず行わなければならないものだったが、メンバー全員が揃わなかったり、時間通りに始めることができなかつたりした。次に、本来は 1 週間で動くものを作り、デモを行うのが最善だったが、デモを行うことができたのが最後の 3 スプリントだけだったため、レビューをもらえる機会が少なくなってしまった。3 つめに、スプリントレトロスペクティブでは、何度も同じ問題が挙げられた時に、効果的な対策方法を探すことがなかなかできず、問題が長続きしてしまうことがあった。最後に、開発の前半はスクラム開発にまだ慣れていなかったため、進捗がうまくあがらず、最終的に、開発の後半にタスクが集中してしまった。

全体として、デイリースクラムへの遅刻、デモ披露の遅延、反省点の長続き、後半へのタスク集中が課題として挙げられた。

(※文責: 熊谷峻)

4.2.3.3 解決策

上記の 4.2.3.2 節で挙げられた課題を解決するために次のような対策を行った。まず、1 つ目の課題であるデイリースクラムへの遅刻に対する対策として、原因は、主にメンバーの寝坊やど忘れによるものが多かったため、デイリースクラムを行う日の前日や朝にリマインドを送ることで忘れられないようにしたり、時間を全員が集まりやすい時間に変更することで寝坊を防いだり、ビデオ通話を利用することで遠隔で参加できるようにした。結果として、デイリースクラムへの遅刻を完全に防ぐことはできなかったが、最小限に抑えることができた。次に、2 つ目の課題であるデモ披露の遅延に対する対策として、1 週間に一度の決められた時間だけでなく、空いている時間に、他のグループのメンバーに積極的に意見をもらいに行くことでレビューをもらう機会を増やした。結果

Leverage the Beacon IoT in Hakodate Real Downtown for Our Smarter Life

として、グループ間での交流が盛んになり、意見交換が行いやすい環境を作り出すことができた。そして、3つ目の課題である反省点の長続きに対する対策として、特にデイリースクラム時にメンバーのコンディションやモチベーションに問題がないかなどを注視し、デイリースクラム以外でも頻繁にメンバー間のコミュニケーションを行うことで、原因の発見を促した。結果として、根本的な原因を特定することができ、スプリントレトロスペクティブをより効率的に行うことができた。最後に、4つ目の課題である開発後半へのタスク集中に対する対策として、スプリントバックログをグループのベロシティに合わせて、消費しきれない範囲で無理なくタスクを割り振るようにし、主にスクラムマスターがデイリースクラムなどでチームの状況を確認し、開発がしやすい環境を整えることで、なるべく後半に負担がかからないようにした。結果として、ある程度負担軽減をすることができたが、最後にタスクが集中してしまうのは避けられなかった。

全体として、デモ披露の遅延と反省点の長続きの2点は解決することができたが、デイリースクラムへの遅刻と後半へのタスク集中の2点はある程度改善できたが、完全な解決をすることはできなかった。よって、この2点に関しては他の解決策を模索することが必要だと考える。

(※文責: 熊谷峻)

第 5 章 システム仕様

5.1 システム構成

本サービスは図 5.1 のように、ビーコン、モバイルアプリケーション、サーバーサイドアプリケーションの 3 つから構成される。ビーコンは、スマートフォンが特定の電波を検知する際の発信器としての役割を担う。本サービスでは、市電車両内にビーコンを設置し、市電に乗車した際にモバイルアプリケーションで電波を受信することで、市電車両への乗車確認を行うために利用する。モバイルアプリケーションはビーコンの電波の検出、短歌の閲覧、短歌を詠むなど、ユーザが行う一通りの機能の提供を行う。ビーコンの電波を受け取っている間、すなわち市電車両に乗車している間のみ、短歌を詠むことが可能となる。サーバーサイドアプリケーションは、初回起動時にモバイルアプリから送られてきたユーザ名をデータベースに登録し、それに対して採番されたユーザ ID をモバイルアプリケーションへ返却する。その後は作成された短歌やその情報の保存を行う。

(※文責: 松山穂乃夏)

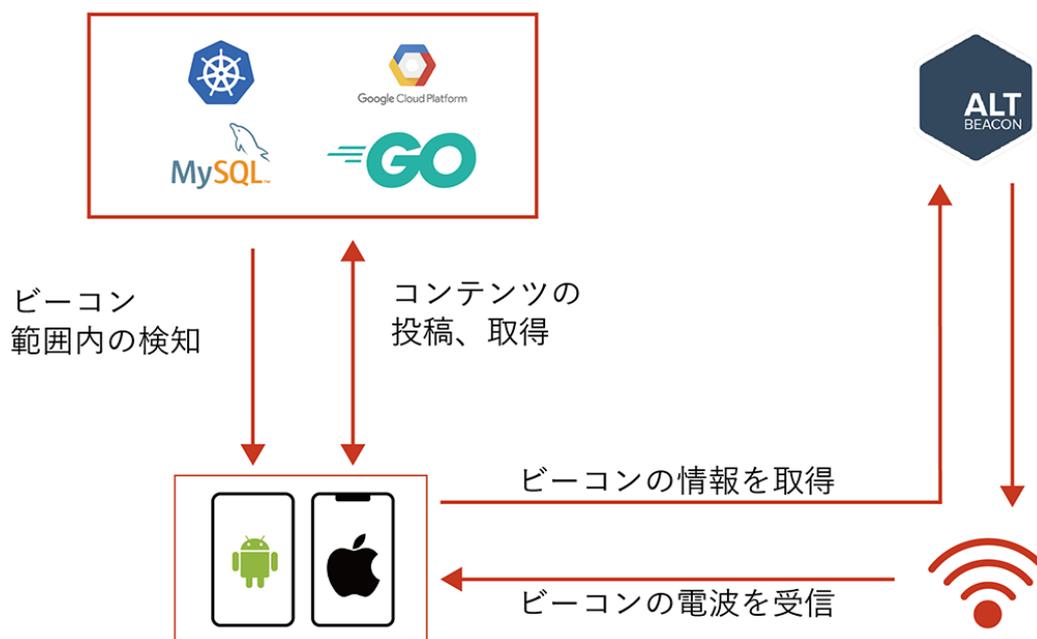


図 5.1 システム構成

5.2 モバイルアプリケーション

5.2.1 iOS

iOS アプリケーションは、Swift 5.0, Xcode 11.1 を用いて開発を行った。ライブラリ管理ツールには CocoaPods を用いた。

(※文責: 石倉愛朗)

5.2.1.1 開発環境

Xcode

本サービスのアプリケーション開発では、iOS アプリケーションと Android アプリケーションの両方の開発を実施した。iOS アプリケーション開発では、Apple 社が提供する Xcode という統合開発環境を使用して開発を実施した。バージョンは Xcode 11.1 である。

(※文責: 石倉愛朗)

5.2.1.2 ライブラリ導入

本アプリケーションの開発では5つのライブラリを導入した。以下に導入したライブラリとバージョン、概要を示す。

- Alamofire (4.9.1) : サーバーへの HTTP リクエスト機能を提供するライブラリである。本アプリケーションでは短歌の送受信、ユーザの登録・取得を行うために GET リクエストと POST リクエストの2つを使用した。
- SkeletonView (1.8.2) : 読み込み時の表示ができるライブラリである。
- SwiftyGif (5.2.0) : iOS で GIF アニメーションを表示するためのライブラリ
- Tabman (2.6.3) : ページングメニューやスワイプによるシームレスなコンテンツの切り替えの機能を提供するライブラリである。
- SwiftyJSON (5.0.0) : 少ない記述量で JSON データを取得でき、また直感的に JSON 内の欲しい値を取得することができるライブラリ

外部ライブラリを使用することにより、アプリに一定の品質の機能をアプリに実装できることや開発における負担を軽減出来るという利点がある。

(※文責: 石倉愛朗)

5.2.1.3 データモデル

本アプリケーションの実装にあたり、Tanka モデルを作成した。Tanka モデルは短歌の情報を保持する。以下にモデルの内容と概要を示す。

- tankaid(プライマキー) : 各短歌に割り振られた一意な ID
- sectionid : 市電の乗車区間を識別する ID
- phrase : 送信された短歌のテキスト
- userid : 短歌を送信したユーザの ID
- date : それぞれの句を投稿した時刻
- taglist : 短歌につけられたタグのリスト
- background : 短歌の背景色
- wordcolor : 短歌の文字色

(※文責: 石倉愛朗)

5.2.1.4 実装方法

1つのホーム画面に「フォロー」・「未完成短歌」・「完成短歌」の3つの画面を実装するために、Tabmanを使用した。Tabmanで実装したタブの色をテーマカラーに変更し、タブ間の移動のアニメーションをデザイナー指定のアニメーションへ変更した。「フォロー」・「未完成短歌」・「完成短歌」のそれぞれ短歌の一覧を表示するために、UIKitのUITableViewを使用した。取得した短歌分のセルを作成、表示し、それぞれのセルに「続きを詠む」・「詳細」ボタンを実装した。「続きを詠む」をタップすると、次に詠む句が何句目かを判断し、その句に対してUITextFieldを表示させ、続きの句を入力できるようにした。「詳細」をタップすると、その短歌の詳細画面へ遷移する。詳細画面では「短歌」・「タグ」・「区間」・「ユーザー名」・「投稿された日付」を表示するが、「ユーザー名」の取得に時間がかかるため、データを受信中はSkeletonViewを使用し、読み込み中であることをユーザにもわかりやすく表示した。また、新しく短歌が投稿された際にUITableViewに新しく反映するためにUIRefreshControlを使用した。これはUITableViewの上部を下方へスワイプすると読み込み中であることを示すサークルを表示する。データを更新し、反映し終わった際に、サークルの表示をなくす。これら短歌情報の送信、受信、ユーザー名の取得などをサーバーと照会するためにAlamofireを使用した。また、受信したデータはJSON形式であり、データに対してSwiftyJSONを使用し、受信したデータを簡単に扱うことができる。それらのデータを先述したTankaモデルに格納した。短歌を投稿する際には、ビーコンを検知する必要があり、TGRFrameworkを使用した。TGRFrameworkとは本サービスで使用するビーコンを提供していただいたTangerine株式会社が作成したフレームワークである。ビーコンを検知中の画面では検知中であることをユーザにわかりやすく表示するためにSwiftyGifを使用し、GIFアニメーションを表示した。また、アプリを起動する際にスプラッシュ画面を設定し、起動が早く応答性に優れているという印象を与えた。スプラッシュ画面には本サービスのロゴを配置し、起動中であることを意識させない起動画像をデザインした。

(※文責: 石倉愛朗)

5.2.2 Android

5.2.2.1 開発環境

Android Studio

本サービスのアプリケーション開発では、iOSアプリケーションとAndroidアプリケーションの両方の開発を実施した。Androidアプリケーション開発では、JetBrain社が開発しているIntelliJ IDEAをGoogle社がAndroidアプリケーション開発向けにカスタマイズしたAndroid Studioという統合開発環境を使用して開発を実施した。Android Studioのバージョンは開発開始当時は3.5.1であったが、セキュリティの観点と機能の差異が殆どないことから逐次アップデートを実施した。報告書作成時点での最新バージョンは3.5.3である。プログラミング言語はKotlinを使用した。選定理由は、Javaとの完全な互換性があるためJavaで記述されたライブラリを再利用できること、Javaと比べコードの記述量が少なくなり保守性に優れること、Javaよりも型の制約が強くコードの安全性が高まることなどがある。報告書作成時点での最新バージョンである1.3.50を使用した。ビルドツールにはGradleを使用した。報告書作成時点での最新バージョンである5.4.1を使用した。

(※文責: 工藤大山)

5.2.2.2 ライブラリ導入

本アプリケーションの開発では4つのライブラリを導入した。以下に導入したライブラリとバージョン、概要を示す。

- Retrofit(2.6.0) : でモルを利用した API 通信用のライブラリである。
- Gson(2.8.6) : Android で JSON を利用するためのライブラリである。
- AltBeacon(2.16.3) : Android で Bluetooth Low Energy 通信を行うためのライブラリである。
- glide(3.7.0) : Android で GIF アニメーションを表示するためのライブラリである。

(※文責: 工藤大山)

5.2.2.3 データモデル

本アプリケーションとサーバサイド・アプリケーションの通信の円滑化のため、Tanka モデルを作成した。以下にモデルの内容を示す。

- tankaid: 各短歌に割り振られるユニークな ID
- sectionid : 市電の乗車区間に割り振られる ID
- phrase : 短歌の各句
- userid : 短歌を作成したユーザ ID
- date : それぞれの句が投稿された時間
- taglist : 短歌に関連付けられたタグのリスト
- background : 短歌の背景色
- wordcolor : 短歌の文字色

(※文責: 工藤大山)

5.3 サーバサイド・アプリケーション

本サービスでは、短歌の情報やユーザ名をデータベースに保存するため、API サーバを開発した。開発には Go 言語を用いた。Go 言語を採択した理由は、習得が他言語より容易だということである。本サービスのデータベースには、短歌の詳細を保存する短歌テーブル、ユーザ名を保存するユーザテーブル、市電に設置する iBeacon の ID を格納するビーコンテーブル、タグを保存するタグテーブル、どの短歌にどのタグがつくのかを管理する短歌タグテーブルの5つのテーブルが存在する。それぞれのテーブルの内容は以下の通りである。

以降では、実装した API について、モデル別に章を分けて説明する。

(※文責: 宮越七菜)

表 5.1 tankatable

カラム名	型	概要
tankaid	int	MySQL による自動採番された ID
sectionid	int	市電の乗車区間を識別する ID
phrase1	string	短歌の第一句目
phrase2	string	短歌の第二句目
phrase3	string	短歌の第三句目
phrase4	string	短歌の第四句目
phrase5	string	短歌の第五句目
user1	int	短歌の第一句目を詠んだユーザ ID
user2	int	短歌の第二句目を詠んだユーザ ID
user3	int	短歌の第三句目を詠んだユーザ ID
user4	int	短歌の第四句目を詠んだユーザ ID
user5	int	短歌の第五句目を詠んだユーザ ID
date1	string	短歌の第一句目が詠まれた日付
date2	string	短歌の第二句目が詠まれた日付
date3	string	短歌の第三句目が詠まれた日付
date4	string	短歌の第四句目が詠まれた日付
date5	string	短歌の第五句目が詠まれた日付

表 5.2 usertable

カラム名	型	概要
userid	int	MySQL による自動採番された ID
username	string	ユーザが登録した名前

表 5.3 beacontable

カラム名	型	概要
uuid	string	市電に設置するビーコンの UUID
major	string	ビーコンの Major 値
minor	string	ビーコンの Minor 値

表 5.4 tagtable

カラム名	型	概要
tagid	int	MySQL による自動採番された ID
tag	string	短歌につけることができるタグ

表 5.5 tankatag

カラム名	型	概要
tankaid	int	タグ付けされた短歌の ID
tagid	int	タグの ID

5.3.1 タイムラインに関する API

- GET /timeline 投稿されたすべての短歌をリスト形式で表示する。
- GET /completeTL 短歌の 5 句すべてが詠まれ、完成された短歌をリスト形式で表示する。
- GET /NocompleteTL 途中の句まで詠まれた未完成の短歌をリスト形式で表示する。

(※文責: 宮越七菜)

5.3.2 短歌投稿に関する API

- POST /addPhrase データベースに投稿された短歌の 1 句を登録する。登録に必要な情報は区間 ID, 何句目かを表す正数, 登録するフレーズ, ユーザ ID, 日付, 背景色, 文字色である。

(※文責: 宮越七菜)

5.3.3 ユーザに関する API

- POST /adduser ユーザ名をデータベースに登録し, ユーザ名に対して採番されたユーザ ID を表示する。登録時に必要な情報はユーザ名のみである。アプリ起動時の初回のみ実行される。
- POST /getuser ユーザ ID に対応するユーザ名を照会し, 表示する。照会に必要な情報はユーザ ID である。

(※文責: 宮越七菜)

5.3.4 ビーコンに関する API

- POST /beacon 検知したビーコンが市電に設置されたものと一致するか照会する。照会に必要な情報はビーコンの UUID, ビーコンの Major 値, ビーコンの Minor 値である。

(※文責: 宮越七菜)

5.3.5 タグに関する API

- GET /gettag 短歌に付けることができるタグを表示する。
- POST /addtag 詠んだ短歌に対して登録したいタグをデータベースに登録する。登録に必要な情報は, 短歌 ID, つけたいタグのリストである。

(※文責: 宮越七菜)

5.4 デザイン

5.4.1 アプリデザイン

サービスのイメージカラーとしてチームメンバーで決定した、緋色 (#d3381c) を主に用いてデザインをした。昨今の SNS を参考に、ユーザが慣れ親しんだ UI を取り入れた。本サービスは他者と繋がる、自分の思いを言葉として投稿する等の点から SNS に近いサービスであると考えたため、このようなデザインを取り入れた。短歌を閲覧する画面では、短歌やその背景に設定する写真が映えるようにシンプルなデザインと配色にしている。市電乗車中の短い時間の中で手軽に言葉を投稿することができるように、少ない動線で投稿まで辿り着けるような UI になるよう心がけた。また、ユーザが第二句以降の続きを読みたいと感じる短歌に出会った時に、すぐ選択することのできる UI を採択した。続きを読みたい、同じ景色を見て感じた想いを綴りたい、と感じた時に、幾つかの手順を踏んでいてはその出会いの感動や瞬間的な想いを逃してしまうため、未完成の短歌から直接続きを詠む画面に遷移する UI を作成した。

(※文責: 松山穂乃夏)

5.4.2 サービスロゴ

ロゴを作成するにあたり、最初にサービスの特徴や要素を考えた。函館、ノスタルジックやレトロ、人との繋がりなどの要素が挙がり、それらを総括してロゴのラフスケッチを行った。いくつかの案をラフにて作成後、サービスの雰囲気似合うもの、先に挙げた要素が取り入れられているものを選び、Illustrator でプロトタイプを作成した。(図 5.2) 短歌の持つレトロや和風のイメージを含めるため、毛筆で書かれた青柳疎石フォントを使用した。このフォントは細く繊細かつ流れるような線が特徴的であり、サービスのイメージに合うと考えたためである。花卉は函館の市花であるつつじと、五稜郭公園に咲く桜をイメージしている。偶然同じ景色を見て、短歌の中での短い繋がりをもった他ユーザとの刹那的な出会いを、花の持つ瞬間的かつ美しいイメージと重ねている。花卉のイラストには水彩のテクスチャを加えて柔らかさを演出させた。最終調整として花卉の位置や数の微調整を繰り返したり、紅葉のイラストを加えたりしながらメンバーと話し合い、決定した。(図 5.3) (図 5.4) (図 5.5)

(※文責: 松山穂乃夏)

5.4.3 アプリアイコン

サービスロゴをベースに、アイコンのサイズでもサービスの概要が伝わるよう特徴や要素を再び考え、先述した要素に加えてビーコンの要素も取り入れた。ビーコンの電波をイメージした同心円と、短冊と筆を用いて短歌の要素を取り入れた。いくつかの配色パターンを作成し、メンバーと相談後決定した。サービスのイメージカラーである緋色 (#d3381c) をベースに、暖色を用いてまとまりのある配色にした。

(※文責: 松山穂乃夏)

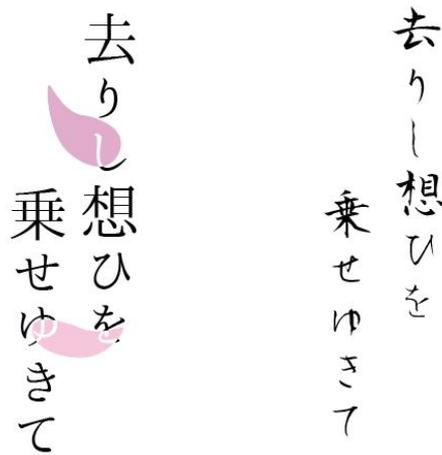


図 5.2 サービスロゴのプロトタイプ



図 5.3 サービスロゴ案 1

図 5.4 サービスロゴ案 2

図 5.5 サービスロゴ案 3

5.5 サーバサイド・インフラストラクチャー

本サービスの API サーバを運用するためのクラウドコンピューティングサービスとして Google Cloud Platform(以後, GCP) を採用した。GCP とは, Google 社が提供するクラウドサービス群のことである。GCP を用いることで仮想マシンを従量課金制で使用できる。本サービスでは GCP のサービスである Google Compute Engine (以降, GCE) と Google Kubernetes Engine (以降, GKE) を使用した。GCE とは, 仮想マシンと呼ばれる仮想的なコンピュータを提供するサービスである。GCE を使用することで, 要件に合わせた性能の仮想マシンを用意できる。GKE とは, コンテナ化されたアプリケーションを運用するためのサービスである。GKE を用いることでサービスの需要に合わせてオートスケーリングと呼ばれる CPU やメモリの量を自動で増やすシステムを完全に停止させずにアプリケーションを更新すること, 障害発生時の自動修復を行うことなどが可能になる。本サービスでは, GCE に MySQL をインストールし, Go で作成した API サーバを立てた。また, GKE の構築完了後, API サーバを GCE から GKE へ移行した。

(※文責: 宮越七菜)

第 6 章 成果発表会

6.1 中間発表

6.1.1 発表形式

2019年7月19日に、前期プロジェクト活動の成果を発表するため、中間発表を行った。中間発表では、プロジェクト全体のスライドとポスター1枚に加えて、各サービスのポスター4枚を用いた。最初の5分間で、全体スライドの発表を行った。全体スライドでは、プロジェクトの概要や、これまでに行った勉強会、フィールドワークでの様子、アイデア出しの方法、今後の予定、決定した4つのアイデアについて説明した。全体ポスターは、後から来た人へプロジェクトの概要を説明するために使用した。次に、各サービスごとに分かれポスターセッションと質疑応答を行った。ポスターセッションを行ったサービスは、「去りし想ひを乗せゆきて」・「みみうち」・「ゆまち」・「函ライブ」の4サービスであった。「去りし想ひを乗せゆきて」では、前半と後半に2人ずつに分かれ発表を行った。2人のうち1人は全体発表を1回とサービスの発表を1回、もう1人はサービスの発表を2回担当した。

(※文責: 宮越七菜)

6.1.2 発表技術の評価と反省

発表技術に関して、高評価な意見として

- 対話形式での発表で、とてもわかりやすかった
- ユーザストーリーや展望を繰り返して説明してくれたので、理解はできた

などがあった。低評価な意見としては、

- スマホを見ながらの発表が少し残念だった
- 原稿を読みながらの発表で伝わりにくかった
- 説明の仕方がイメージしづらかった

などがあり、平均評価は7.38であった。以上のことから、質疑応答でサービスを理解することができたが、原稿を読むという発表態度が印象の良いものではなかったことに加え、イメージしづらく分かりづらい発表だったということが分かった。改善点としては、原稿を読まずに発表することや、発表練習時に伝わりにくい部分がないか確認することがあげられる。

(※文責: 宮越七菜)

6.1.3 発表内容の評価と反省

発表内容に関しての高評価な意見として

- 短歌が知らない人によってどんどん成長していくのは面白い
- ビーコンは上手く使えている
- お店自慢の上の句に下の句をコメントできるのは良い

などがあつた。低評価な意見として

- 仕組だけの説明で「誰が」「なぜやりたいか」の説明が抜けている
- ターゲットに景色を楽しませたいのか、短歌を楽しませたいのかはつきりさせたいほうが良い
- 最初にユーザに興味を持ってもらえるような工夫が必要

などがあり、平均評価は 7.82 であつた。以上のことから、他の人によって短歌の続きが詠まれることは伝わつたが、対象ユーザやサービスを利用するモチベーションが曖昧だということが分かつた。よつて、対象ユーザやサービスを利用するモチベーションを再定義することが必要であることが分かつた。

(※文責: 宮越七菜)

6.2 公立はこだて未来大学オープンキャンパス

公立はこだて未来大学では 2019 年 8 月 4 日にオープンキャンパスが開催された。本プロジェクトでは、オープンキャンパスのイベントの 1 つであるインタラクティブ展示にて、「ここにいる人だけ」コミュニケーションとして出展した。出展にはプロジェクトのメンバー 8 人が参加し、中間発表時に使用したプロジェクト全体のポスター 1 枚と各サービスのポスター 4 枚、昨年度のプロジェクトのサービスのポスター 2 枚、その昨年度のサービスが体験できるデモを用意した。今年度のサービスは、オープンキャンパス開催時にはまだ実装ができていなかったため、昨年度のサービスを用いることで来場者にビーコンを用いたサービスを体験していただくことができた。今年度のサービスについては、興味を持った方に向けてサービスの概要や機能について簡単にポスターセッションを行った。

(※文責: 大野和也)

6.3 アカデミックリンク

2019 年 11 月 9 日に北海道教育大学函館校で行われた、はこだて高等教育機関合同研究発表会 HAKODATE アカデミックリンク 2019 に参加した。HAKODATE アカデミックリンクとは函館市内にある 8 つの大学・短大・高専の学生が一堂に集結し、ステージ発表やポスターセッションを通じて各々の研究を伝えるイベントである。函ライブ・みみうちと、ゆまち・去りし想ひを乗せゆきての 2 つに分かれてブースセッションを行った。最初に本プロジェクトの概要とビーコンについての説明を行い、各サービスの説明を順番に行った。サービスの説明では、本サービスの背景や目的、機能などについてポスターを用いて説明を行い、その後スマートフォンを使用してデモを行った。受け取つた意見として、

- 短歌という題材が面白い

- 他の人と一緒に詠む体験が新しい
- 市電で短歌を詠んだことを離れている家族が知ることが出来れば、単身赴任中などでも楽しめるそう

など、好意的な内容のものが多くみられた。発表の最中にはサービスの概要を上手く伝えきれず混乱させてしまう場面もあったが、効果的にサービスの内容を伝えることができたと言える。ゆまちの実際に使ってみてみたいと思わせる実用性と、去りし想ひを乗せゆきての新体験を提供するエンターテインメント性が評価され、審査員特別賞を受賞した(6.1)。



図 6.1 審査員特別賞

(※文責: 松山穂乃夏)

6.4 成果発表会

6.4.1 発表形式

成果発表会では3階のモールで発表を行った。はじめに本プロジェクトの概要や各アイデアのおおまかな説明を行うためにスライドを用い、5分ほど説明をした。その後、「去りし想ひを乗せゆきて」・「ゆまち」・「みみうち」・「函ライブ」の4サービスから興味を持っていただいたものを選んで頂き9分間で発表と質疑応答を行った。プロジェクト全体の展示として、ポスター1枚、ビデオ1つ、また、青森市学生ビジネスアイデアコンテストで受賞したグランプリを獲得した際のトロフィーも展示した。本グループはグループポスター、AndroidとiOSの両方の実機の画面を展示した。実機画面の展示に関しては来ていただいた人に、実際に操作画面や短歌投稿画面を見せながら説明した。

6.4.2 発表形式の評価と反省

発表形式に関して、高評価な意見として

- デモがとてもわかり易い
- 背景，ストーリー，内容がしっかりと考えられている

などが得られた。低評価な意見としては

- 声が聞きづらい

が得られた。平均評価は 10 点中、8.7 点だった。デモ端末を 2 台用意し、実際にデモを体験し、短歌を詠み合うことで、展示に来ていただいた人に本サービスの背景，ストーリーや内容をしっかりと理解してもらうことが出来た。成果発表会にデモを用意して、デモに関する高評価な意見が多く得られた点は良かった点だと思う。しかし、開発やポスター制作に多くの時間をかけてしまい、ポスター発表の練習に時間を割けなかったことや成果発表会のスペースの場がとても人が多かったことなどにより、声が聞きづらいという低評価な意見が見られた。開発やポスター制作には余裕を持って取り組む必要があることや予め声が聞きづらいことを想定し、発表形式を工夫することなど改善が必要な点が多くある。

(※文責: 石倉愛朗)

6.4.3 発表内容の評価と反省

発表内容に関して、高評価な意見として

- 新たなコミュニケーションとして利用できると感じた
- 操作性がよく、UI もわかりやすい
- 市電に乗っている時間をうまく利用できる

などが得られた。低評価な意見としては、

- アプリの UI をもう少し工夫できる

が得られた。アプリの UI・UX に関しては誰が使用してもわかりやすく、使いやすいようなものを作成しなければならない。機能的な面に関しては実装することができなかった機能があるので、アプリ開発をする際に計画的に機能を実装していく必要があると感じた。

(※文責: 石倉愛朗)

6.5 enPiT

6.5.1 発表形式

2019年12月14日にプロジェクトでの成果を発表するため、室蘭工業大学にて開催されたenPiT BizSysD 北海道・東北グループ合同発表会2019に参加した。参加校は公立ほこだて未来大学の他、室蘭工業大学・北海道情報大学・会津大学・岩手県立大学であった。発表では、成果発表会のときと同様に、プロジェクト全体のポスターとサービスのポスター各1枚の掲示とタブレットPC1台でプロジェクトの紹介動画の再生、実際に制作したアプリケーションのデモを実施した。発表時間は1時間交代制の2ターム制であった。発表内容は5分程度のポスターセッションを行いながら逐次質問を受け付ける形式であった。「去りし想いを乗せゆきて」は第2タームの発表であった。

(※文責: 工藤大山)

6.5.2 発表内容の評価と反省

発表内容に関する、高評価な意見として、

- シンプルなUIで使いやすい
- 市電と短歌という函館ならではのトピックを取り上げている点が面白い
- 短歌を詠むアプリは少ないのでアイデアとして面白そう

などが得られた。低評価な意見として、

- そもそも短歌を詠む人が多くないと思われる
- 収益化が難しそう

などが挙げられた。短歌を詠む層を増やすという点に関しては、広告などのアプリ以外も活用したアプローチを利用する必要もあると考えられる。収益化に関しては、アプリ内広告などの手法が考えられるが難しい課題であると考えられる。

(※文責: 工藤大山)

第 7 章 各メンバーの振り返り

7.1 役割分担

本グループでは、スクラム開発手法に則り、プロダクトオーナー 1 人、スクラムマスター 1 人を決定した上で、4 人全員がサービス開発に関わるように、4 人のうち、1 人がデザイン、2 人がモバイルアプリの開発、1 人がサーバサイドアプリの開発を行った。報告書の執筆はグループメンバー全員で分担して行った。以下にそれぞれが担当した内容を示す。

松山穂乃夏

- プロダクトオーナー
- モバイルアプリのデザインと UI 設計
- XD と Zeplin を用いたモックアップ作成
- サービスのロゴとアイコン作成
- ポスター作成

石倉愛朗

- iOS アプリ開発
- UI の実装

工藤大山

- テックリード
- Android アプリ開発
- UI の実装

宮越七菜

- スクラムマスター
- サーバサイドアプリの開発

(※文責: 宮越七菜)

7.2 松山穂乃夏の振り返り

プロジェクトを通してチームで実際の実開発を行う経験を得られたことは、自分にとって大きな糧になったと感じる。これまでの講義で、チームでのアイデア出しやプロダクトの制作を行ってきたことはあった。しかし、これだけの長い時間をかけ、各々の役割を決め、開発の手法を意識しながらサービスの制作に取り掛かってきたことは無かった。そのため、多くの学びを得ることができたと感じる。

前期の最初に行ったフィールドワークでは、行った先々での気付きや疑問を随時 Slack に投稿し、リアルタイムでの思考の言語化やメンバーとの共有を行った。そのため、フィールドワークでの発

見を逃すことなくその後の振り返りに繋ぐことができたと感じる。また、アイデア出しでは、ビーコンらしさと新規性を実現させるアイデアを提案することが難しく苦勞した。アイデア出しの手法一つとっても、KJ法やOSTなどの手法を知り、実際に活用することで、効率よく情報を整理する方法を学ぶことができた。しかしアイデア出しでは、その前に行ったフィールドワークでの情報を上手く活かしきることが出来なかったように思う。街を歩いて気付きを得ることは重要であったが、そこから一步踏み込んだ調査を行うことも必要だったのではないかと感じた。

後期に行った開発ではデザインを担当し、UIやロゴ、アプリアイコンのデザインを行った。アプリUIに関しては初めて行ったため無知な部分も多く、XDなどの初めて使うツールに手間取ったり、OSごとに異なるデザインのガイドラインに戸惑った。しかし、それも含めて経験と成長になった。アプリのデザインを担当するにあたり、UIに対する知識が不十分だと感じたため、まずは既存のアプリを使い、使いやすい、取り入れたいと感じるUI、反対に使いにくいと感じるUIについて考えた。その時間を設けたことで、アプリのUIについてのアウトプットの引き出しを増やすことに繋がった。アプリの全体像を考えるにあたり、ホームページを作成した講義の経験を活かしてUIフロー図を作成した。本サービスのアプリにおいて必要な機能は何か、それにおける画面の遷移はどのようになるかを最初にまとめることで、全体の見通しを立てることができた。実際の画面のプロトタイプを作成する際にはXDを使用した。初めての利用であったため手探りな部分ばかりで、機能を活かしきれていない場面も多くあった。プロトタイプの迅速な制作に適しているツールであり、恐らく今後利用する機会も多々あるため、今回の経験と反省を活かして使っていきたいと感じた。アプリUIを作成していく中では、他のメンバーとは先述のXDやZeplinを使用してデザインイメージを共有した。チームメンバーにUIデザインの提案を行い、実装の難易度や自分だけでは気付けない問題点を話し合い、実際のデザインを固めていった。相談する中で決まってくるデザインやアプリの動作も多々あり、コミュニケーションやアウトプットを増やすことの重要性を実感した。試行錯誤の途中で煮詰まった際などには、メンバーに相談をして解決したことも多々あったため、今後もそれを踏まえていきたい。アプリのデザインやアイコンを作成するために配色についての学びを得たり、フォントについての知見を得たりしたことは、今後の自身のアイデアの引き出しにもなった。

反省点として、プロダクトオーナーの役割を担っていたが、その役目を果たしきれなかった点が挙げられる。プロダクトオーナーとして作成しているプロダクトの方向性をしっかりメンバーに示すべきであったが、周りの意見を聞きすぎて決断力に欠ける場面が多くあった。メンバーの意見を聞くことも重要だが、それと同時に自身の持つサービス像の中で譲れない部分を持つておくことも大事であると感じた。今回スクラムの手法を取り入れて開発を進めてきたが、その開発手法のメリットを活かしきれていなかったようにも思える。プロダクトバックログの通りに開発を進めていくことが上手くできていなかった。最初から完璧なものや、デザインにこだわるものではなく、とにかく動くものを作るという意識が少々薄かったこと、それに対する危機感が持てなかったことが反省として挙げられる。今後同じように開発する機会があれば、今回の反省を踏まえていきたい。また、本サービスの開発にあたり、メンバー4人がそれぞれ別の言語や役割を担当しており、互いの進捗や現在取り組んでいることが不明瞭になりがちであったため、もっと積極的に情報交換やコミュニケーションをとることを促進すべきであった。プロジェクト全体を通して、自身のデザインの技術はもちろん、チームでの開発、その中でのデザイナーの役割について、良い経験が得られた。経験と反省点を踏まえ、今後活かしていきたい。

(※文責: 松山穂乃夏)

7.3 宮越七菜の振り返り

前期では、フィールドワーク、アイデア出し、アイデアのブラッシュアップ、アイデア発表をした。フィールドワークでは、フィールドワーク講習で言われた「とにかくメモを取ること」を大事にし、写真とともに思ったこと感じたことをとにかく Slack に投稿した。私が参加したフィールドワークの場所は本町方面と函館駅周辺であった。本町方面のフィールドワークでは、市電についてのフィールドワークを行ったことが印象に残っている。函館市電に乗ったことのないメンバーが居たため、五稜郭から千代台まで市電で移動した。時刻は平日午後 15 時ごろで、エコバッグを掲げた高齢者が多く乗車している印象を受けた。フィールドワーク実施前までは、函館市電は観光者向けの乗り物だと思っていた。しかし、地元の人が移動手段として市電を選択する文化が現代まで残っていることにより、函館は大正ロマンや昭和レトロを感じさせるのだということを実感した。市電に乗車した時刻は平日 15 時ごろだったため、他の時間帯の乗客はどのような人が多いのか、機会があれば検証しに行きたい。函館駅周辺のフィールドワークでは、Slack を用いて他の場所でフィールドワークを実施しているグループのリアルタイムな投稿を見ることに新鮮さを感じた。函館駅周辺のフィールドワークを実施する自分らの他に、函館山と港周辺のフィールドワークをするグループがあった。函館駅周辺には新鮮な海鮮を取り扱う朝市や、買い物が楽しめる赤レンガ倉庫がある。自分らがそのような場所を周り、観光客さながらに楽しんでいるのと同時に、函館山グループが必死に登山をし、苦しそうな写真を Slack 上に投稿していた。自分はその投稿に対して「頑張れ」などのスタンプを押した。スタンプを押すという形で投稿を見たときの気持ちを表せることは新鮮だと感じた。Slack のスタンプ機能は、コメントを書くだけでなくチャットに参加している人の気持ちを反映することができるため、チャットを盛り上げることや、発言回数を増やすことにつながると思った。アイデア出しでは、話し合いやアイデアとして論理的な形に捻出することが難しいと感じた。フィールドワーク時、写真とともに思ったこと感じたことを Slack に投稿した。しかし、投稿が非常に抽象的であり、「函館を歩いてみた感想」以上の発見につなげることが難しかった。プロジェクトの時間を利用し、模造紙と付箋を用いてアイデアについて話し合う場を設けた。しかし、アイデアを深めるということが何について話し合うことなのか分からず、あまり議論を発展させることができなかった。一方で、松原先生や TA の中田さんが話し合いに加わると、「誰が」・「何を」・「いつ」・「なぜ」などの 5W1H に着目し、アイデアと既存のものを組み合わせる新規性のあるアイデアを生み出す方向性により議論が進んだ。松原先生と中田さんの発想力に感服するとともに、その発想力はどこで養われるものなのか疑問に思った。今回のアイデア出しでは、ブラッシュアップによってアイデアを固めたというよりも、発表期日に間に合うように急いで発表用のスライドを作成し、発表し、フィードバックを得ることでアイデアを深めたような印象がある。今年度のメンバーにとっては、アイデアを固める上でアイデアの手法を用いてさらに深めるよりも、スライドを用いてプレゼンし、言語化していくというほうが向いていたのかもしれない。次にアイデア出しの機会があるとすれば、アイデア出しの手法を用いてみたいと思った。夏季休業期間では、各自で開発言語習得の時間を割いた。私はサーバサイドを引き受け、Go 言語の習得に向けてチュートリアルを行った。その中で、分からないことを質問することが難しいと感じた。分からない部分があっても、分からない部分がどこか自分では把握できないことや、把握していないまま何をどう言えば伝わるのかわからず、自分の中でも混乱した状態で質問をすることが辛かった。質問に対するハードルの高さは夏季休業明けの開発期間でも発揮され、人に聞けば早期解決したかもしれない実装部分に時間をかけてしまった。原因として、調べたことを忘れてしまうことが

挙げられる。忘れてしまうため、何をした結果どのような結果が得られたのかがわからなくなり、質問の仕方がわからなくなっていたと推測する。今後は、調べる前に何がしたいのか、何を調べたのか、何をどのように変更したのか、その結果どうなったのかをメモとして残していきたい。後期では、サービス策定をし、開発を行った。サービス策定後、夏季休業に入る前にチーム全体でスクラムの勉強をしておくべきだったと強く後悔している。スクラムに対する知識が不足したまま計画策定を行ったことで、毎週のスプリントレビューや計画ミーティングが曖昧になり、個人開発のようになったことが悔やまれる。来年度、本プロジェクトが存在しているとすれば、ビーコンレクチャーとアジャイル開発手法の勉強会は夏季休業よりも前に行うべきだと思う。合宿でプロダクト計画策定をしても楽しいかもしれない。開発を進めていく中で、チーム開発に有効なツールである GitHub や ZenHub, Zeplin の使い方を習得できた。ツールだけでなく、アイデアの企画から開発まで携わることができ、チーム開発やスクラムを体験できたのは貴重な体験だったと感じた。今後、チーム開発をする機会があれば、この1年間の体験を活かしたい。

(※文責: 宮越七菜)

7.4 石倉愛朗の振り返り

本プロジェクトはプロジェクト名の通り、ビーコンを使用し、函館に関連するサービスを提供するのが目的とされる。まず、始めにフィールドワークをし、その後、BS や KJ 法, OST などを行い、テーマを決定したが、どの作業、手法も実践するのが初めてで、本来の正しいアイデア出しが出来なかったと感じる。また、ビーコンに関する知識が足りない状態でアイデア出しを行っている状態であった。様々な作業方法や手法、また、IoT 機器であるビーコンに関して、詳しく知ることが出来たのはとても良い機会だった。一方、前期の多くの時間をアイデア出しのために使い、開発に着手するまでがとても遅かったことが反省点であった。開発においては iOS のアプリの開発を担当した。iOS のアプリを作成したことはあるが、1年を通して1つのアプリを作成することは初めての体験だった。故に、iOS をどのようなアーキテクチャで開発を進めていくか、また、どの機能から実装していけば良いか等、実際に開発をする上での知識が足りなかったと思う。設計方法を考えずに、開発を進めたことにより、クラスが肥大化し、コードを参考にしにくくなったこと、同じ処理を使い回せないことや機能の修正や追加が困難となったことが発生した。次回以降、個人やチームでアプリ開発をする時までに iOS アプリにおける様々な設計、アーキテクチャパターンを学ばなければならない。また、スクラム開発を取り入れるのも初めてであり、始めは慣れず、戸惑う場面が多かった。週に3回行うデイリースクラムは互いの進捗を報告し合うことで、互いの進捗具合などの様々な気づきがあったと思う。しかし、次のデイリースクラムまでに進捗を出すことが出来ない期間が度々あった。プロジェクト学習の時間だけでなく、時間外で多く作業をする必要があり、また、メリハリをつけ継続的に開発を続けていくことが重要であると改めて感じた。結果として、実装することが出来なかった機能があったことが大変悔やまれる。今後、普段から開発をより身近なものにし、出来ないことを出来るように努力していかなければいけないと思う。また、GitHub, ZenHub, Zeplin のツールを使うことでタスク管理やデザインの共有が容易に出来たが、もう少し、有効的に使うことが出来たと思う。タスク管理は曖昧になってしまっていた。チームメンバーがどれくらい進捗を出せていて、どこで躓いているのか等、ひと目で分かることが出来るツールであるので、互いにコミュニケーションを取り合い、情報を交換したり、分からない部分は先輩に聞いたりするなど、自分だけの問題にせず、自分以外の誰かに相談することが大切であった

と思う。プロジェクト学習全体を通して、個人での iOS アプリ開発の大変さ、チーム単位での開発の大変さやチームメンバー間のコミュニケーションの大変さ等、決して、1人では体験することが出来ないことをこの1年間を通して、学ぶことが出来た。今後のチーム開発や個人開発等にこの1年間の経験を活かしていきたい。

(※文責: 石倉愛朗)

7.5 工藤大山の振り返り

今回のシステム情報科学実習(以下、プロジェクト学習)では、一年間を通して様々な経験をする事ができた。プロジェクト学習のような PBL(ProjectBased-learning)形式を採用した実習は初めての経験で全くノウハウがなかった。そのため、そもそも PBL とはなんなのかから学習を開始し、PBL で学びを得る方法を考えたりする必要があったため大変であった。しかし、実際に社会に出てから仕事を始めると、個人で仕事をする事は少なく集団でチームを組むことが多いため、チームでの作業を行う経験ができるという点で、大変有意義であったと感じている。プロジェクト学習開始前までは、数人でチームを組みプログラム開発を行ったり、開発のためのアイデアを固めるといった経験はあった。しかし、1年を通して課題のリサーチから、要件定義、開発、検証まで自分たちでやる PBL 形式での学習というのはなかなか大変で骨が折れることであることを身に染みて感じる事ができた。これは大学生活の中でもなかなかできない体験であると感じている。話は変わるが、今回のプロジェクト学習ではアジャイル開発を採用した。アジャイル開発とは、1週間程度の短い期間で要件定義から開発までを一スプリントとして定義し、短い期間で要件定義から開発、レビューまでをこなすことで、早い段階で成果物が出来上がるため最終的に制作する成果物がイメージしやすく、想定と違った場合の修正が素早く行える開発方式である。私が所属したチームでは iOS アプリケーション開発担当と Android アプリケーション開発担当、サーバサイド・アプリケーション開発担当、デザイン担当がそれぞれ一人ずつの計4人で開発を行った。開発の途中で作業の手戻りが何回か発生したものの、アジャイル開発を採用していたことにより軌道修正を行いながらある程度アプリケーションとして成果物を作成する事ができた。もし、ウォーターフォールモデルを開発モデルとして採用していたとしたら、成果物の作成は困難となっていたと考えられる。

私の担当は Android アプリケーションの開発がメインであった。これのほかに、サーバサイド・アプリケーション開発環境の構築や保守、フロントエンド・アプリケーション(ここでは iOS アプリケーションと Android アプリケーションのこと)とサーバサイド・アプリケーション間で通信を行うためのデータセットや API の定義、先述した開発環境や Android アプリケーション(他チームのも含め)に不具合が発生した場合の解決・修正など技術面からチームメンバーやプロジェクトメンバーを支えるような役割をした。具体例として、AWS(Amazon Web Service) EC2 Instance(従量課金制のクラウドコンピューティングサービス)を利用した議事録作成者選出システム、Docker(OS レベルの仮想化環境を提供するオープンソースソフトウェア)による Golang + MySQL の開発環境の構築やフロントエンド・アプリケーションとサーバサイド・アプリケーションとの通信に使用した JSON データセットの定義、である。これらのツールでの開発経験や Docker を用いたツール作成の経験はほとんどなかったため手探りで進めていった分、時間がかかったり正常に動作しない場合があった。他のメンバーに迷惑をかけてしまう部分もあったことは反省しているが、ある程度は開発に貢献できたと考える。

Leverage the Beacon IoT in Hakodate Real Downtown for Our Smarter Life

最後に、プロジェクト学習は1人では決して行うことができない学習である。そのため、他のプロジェクトメンバーには迷惑をかけてしまう部分もあったが、快く受け止めてくれて本当に感謝している。この1年で得た経験は今後役立てていきたいと考える。

(※文責: 工藤大山)

第 8 章 まとめと展望

8.1 前期の振り返り

前期では始めに南部先生からフィールドワークに関する講義を行ってもらい、フィールドワークを行う際の注意点等を学習した。フィールドワークで訪れた場所は公立はこだて未来大学周辺、五稜郭公園周辺、本町周辺、湯の川地区、西部地区、函館駅周辺、函館山周辺である。フィールドワークの Slack のチャンネルを作成し、フィールドワークで気づいたこと、感じたことや写真などを投稿し、共有を行った。フィールドワーク後、BS と KJ 法を用いて、フィールドワークで気づいた点やアイデアなどを書き出して地区別ごとに場所やイメージなどでグループ分けをし、模造紙に貼り付け、振り分けた。次にビーコンについて調べた。主にビーコンの特徴、種類、活用事例や課題などに着目し、学習した。ビーコンについて調べ、理解することにより、その後に行うアイデア出しにおいて、ビーコンの特色を出せるようなアイデアを出し合うことができた。また、6月19日に公立はこだて未来大学において開催された「新技術開発サロンとの情報交換会」に参加し、様々な会社の概要と事業内容の説明を聞いた。そこで本プロジェクト内で考えたアイデアの発表・質疑応答を行った。そこで参加企業の方々から様々な意見をいただくことが出来た。学生からの視点ではなく、企業からの視点の意見をもらうことにより、新しい視点からアイデアを考えることが出来た。そして、それらの意見を参考に、アイデアの再考を行った。次に本年のビーコンプロジェクトのロゴを作成した。各メンバーがロゴを作成し、それに対し、各々がレビューをし、ロゴアイデアを改善することを数回繰り返した。その後、1人3票で投票を行い、投票が多かったロゴには、Slack 上でさらにレビューを行った。最終的に、投票が多かったロゴに対して、1人1票で投票を行い、ロゴを決定した。決定したロゴに対して、担当を決め、色・デザイン等の改善案を考えた。決定したロゴの色指定やサイズ、余白の最小値など、使用する際の注意点についてまとめたガイドラインを作成し、その後のスライドやポスター作成での利用を快適に行えるように下準備を行った。次に本プロジェクトの TA に Git/GitHub ワークショップを開催してもらい、開発に必要な Git/GitHub に関する知識を取得した。開発に関わるのが初めてだったプロジェクトメンバーは Git/GitHub の基礎的なシステムを学ぶことができ、既に開発に関わり、Git/GitHub を使用したことがあるプロジェクトメンバーは Git/GitHub に関する機能をより深く知ることが出来た。次に昨年度サービスの説明を昨年度サービスの代表者から聞いた。「@ハナセル」・「いさりび Graffiti」・「becoma」・「Telepath」の4つのサービスについて説明をもらい、質疑応答を行い、代表者より今年度サービスに対する様々な意見をもらうことが出来た。次にアイデア出しを行った。アイデア出しでは簡単なアイデアを200個程度提案し、BSKJ法・OST・ブラッシュアップを得て6つのアイデアに絞った。ここから「函館らしさ」・「ビーコンらしさ」・「新規性」・「継続性」・「必要性」の5つの観点で評価し、最終的に「去りし想ひを寄せゆきて」・「ゆまち」・「みみうち」・「函ライブ」の4つのサービスを後期に開発していくこととなった。

(※文責: 石倉愛朗)

8.2 後期の振り返り

後期の始めにスクラム開発についてグループで導入するかプロジェクトメンバーに確認を取り、全員の了承を得て導入することとなった。その後、スクラム開発について各々、勉強し知識を深めた。そして、プロダクトオーナー、スクラムマスターを決め、開発を開始した。また、開発を行う際に使用するツールやスクラム開発のデイリースクラムを週に何回、何時に行うかなど開発に必要なことを取り決めた。ツールは Slack, Adobe Photoshop, Adobe Illustrator, Adobe XD, Zeplin, Git, Docker を使用した。サービスは Git ホスティングサービスである GitHub を使用した。ツールやサービスを導入したことにより、円滑にコミュニケーションを取り合えたり、開発を進めたりすることができた。デイリースクラムはリモートで金、日、火曜日の 20 時に行った。2019 年 11 月に行われた HAKODATE アカデミックリンク 2019 で本サービスを評価してもらい、その評価をもとに成果発表会に向けて機能の修正を行った。そして、12 月 6 日の成果発表会において本サービスの最終発表を行った。

(※文責: 石倉愛朗)

8.3 今後の展望

今回実装したサービスでは、現段階で実現できていない機能が存在する。それは主に 2 つあり、1 つ目は通知機能である。自分がいずれかの句を詠んだ短歌が完成した時に、ビーコンの範囲外でも通知が来るという機能である。短歌の完成通知がビーコンの範囲外でも届き、自分の関わった短歌の完成した形を見たいと感じさせることによって、市電の乗車を促すことに繋がるのではないかと考える。また、次回以降の市電に乗車する楽しみを増やすことができる。2 つ目は写真の添付機能である。作成する短歌の背景として、短歌のテーマとなった函館の街の写真を 1 枚添付する機能である。現在は第一句を詠む人が、自分の詠んだ言葉の感情や想いに合わせて短歌を表示する画面の背景色や、文字色を変更できる。そこに写真という手段を加えることで、このサービスの目的である同じ景色の共有をより詳細に行うことができる。第一句を詠むだけでは伝えきれない、見た景色の美しさやその状況を詳細に伝えられるようになることで、より他の人と想いを繋ぐ短歌を詠むことができると考える。また、本サービスの発展的な利用方法として、市電に乗車するユーザー同士だけではなく、市電沿線の店とユーザーが共に短歌を詠む機能を展望として述べる。利用方法として、まず市電沿線の店が、自身の店の自慢の商品について短歌の上の句を詠んで投稿する。次にユーザーが、その商品に対する感想や口コミを続けて下の句として詠んで投稿する。1 つの上の句に対して、下の句は複数作成することができる。完成したそれぞれの短歌は、店とユーザーが共に詠んだ商品紹介の一首となり、それを見た他のユーザーが市電乗車のついでに立ち寄ってみたいくなるような宣伝効果を生むと考える。これらを実現し、今後本サービスが今以上に函館の街と人を繋ぐきっかけになることを願っている。

(※文責: 松山穂乃夏)

第9章 学び

9.1 情報をアウトプットする技術

前期では、ロゴ制作、フィールドワーク、サービス考案というアイデアをアウトプットする活動を行ってきた。ロゴ制作では、メンバー各自でプロジェクトのイメージを考えロゴを制作し、制作したロゴのイメージを他メンバーに伝えた。フィールドワークでは、函館市内6ヶ所にフィールドワークを行った。フィールドワークを行った場所は、西部地区・函館駅周辺・函館山・湯の川方面・五稜郭周辺・未来大周辺の6ヶ所である。この6ヶ所を回り、思ったことや気づいたことについて付箋に書き出し、ブレインストーミングでまとめた。その後、似ている意見をまとめてラベルを付け、KJ法で意見の関連を明らかにした。サービス考案では、フィールドワークのまとめや自身の経験からサービスのアイデアを考えた。小規模なOSTを用いて、自身が興味のあるアイデアに対してさらに議論を深めていく方式をとることにより、より活発な議論を行うことができた。また、アイデアについてプレゼンを行い、自身が考えるアイデアをメンバーに伝えた。最後にプロジェクトの中間発表では、スライドを用いたプロジェクト概要の説明に加え、考案したサービスのポスターセッションを行った。聞き手に伝わるようにスライドとポスターの文章を添削し、発表練習で聞き手がわかりにくいと感じた部分を修正することによって、アウトプットの技術を高めることができた。

(※文責: 宮越七菜)

9.2 サービス設計

前期のプロジェクトを通して、どのようにしてサービスを設計するのかを学ぶことができた。考案したサービスについて新規性、有用性、継続性、ビーコンらしさ、函館らしさを考え、教員やTA、他のグループメンバーによるレビューにより、問題点を繰り返し修正した。また、今回提案するサービスと類似した既存サービスとの相違について考えることで、サービスの新規性を深めることができた。これにより、サービスを考案・設計する技術を向上させることができた。

(※文責: 宮越七菜)

9.3 振り返り

9.3.1 スケジュール管理

プロジェクトリーダーを中心に年間スケジュールを作成し、それに従いプロジェクトの管理を行った。前期の活動では、スケジュールをプロジェクトのメンバー全員で共有したので、いつまで何を、これからは何をやるのかを把握することができた。また、グループ決定後はグループリーダーが活動内容や提出物の期限の管理を積極的に行い、スムーズに活動を進行することができた。しかし、当初の予定以上にアイデア出しやロゴ作成に割り当てる時間を多く取ってしまったの

で、スケジュール通りに進めることが出来なかった。加えて、アジェンダや今後の大まかな流れについて共有したにも関わらず、メンバー間で認識の差異があったことも課題である。後期の活動では、スケジュールを改めて見直し、メンバー同士で情報共有をこまめに行う必要がある。

(※文責: 大野和也)

9.3.2 情報共有

ロゴの作成やフィールドワークから得られたこと、アイデア出しでの意見交換、アイデア案の発表など、多くの場面で情報共有を行った。しかし、大人数で話し合うと意見をアウトプットしないメンバーが出てしまったり、同じ話題について話し合いを続けていると途中で意見が途絶えてしまうという問題があった。本プロジェクトでは、特にアイデア出しでの意見交換での場面ではどのような手法が良いかを考察した。最終的には、4つほどのグループを作り、最初は自分が議論に参加したいグループへと参加し、途中で他のグループの議論に参加したくなったら移動するという手法を用いた。その結果、メンバー全員が意見をアウトプットでき、議論が滞るということも無くなった。これらのことから、後期の活動でもグループ間の移動を積極的に行い、興味がある話題について議論する手法を用いることが重要である。

(※文責: 大野和也)

付録 A 成果報告会で使用したプロジェクト概要のポスター



図 A.1 プロジェクト概要 (最終成果報告会) (1/2 枚目)

開発 Software Development

開発手法 Development Method

- アジャイル開発の手法のひとつ、スクラムに挑戦
- 1週間で計画、実装、振り返りを行い、フィードバックを得て次週の開発に活かす
- 他の開発手法に比べ、仕様の変更や状況の変化に対して柔軟な対応ができる
- We used scrum one of agile processes.
- We repeat plan, implementation and review in a week.
- Scrum can respond to specification changes or situation changes.

チーム構成 Team Formation

- 3~4人のチームを構成
- 各チームに、開発するプロダクトの責任者であるプロダクトオーナー (PO)、スクラムを正しく実践できるよう責任を負うスクラムマスター (SM) が存在する
- We formed team has 3 or 4 members.
- Each team has product owner who has responsibility about a product and scrum master who has responsibility about scrum process.

開発フロー Development Flow

計画



1週間 (1スプリント) の間に実装する機能を計画する

We plan functions which implement programs during a week.

実装



週3~5日決めた時間に『やったこと、これからやること、困っていること』を共有

We shared "what I done", "what I will do", "what I'm worried" to group members between three to five times on every week.

振り返り



制作したプロダクトのデモを行い、互いに質問やアドバイスをした

We demonstrated product, asked questions and advised each other.



機能の実装を実現させるために必要なタスクを洗い出す

We identified tasks which need to implement functions.



共有した情報を元に、機能の実装を行った

We implement programs based on information we shared.



KPTやFDLを用いて一週間の振り返りを行った

We looked back a week using KPT(Keep, Problem, Try) or FDL(Fun, Done, Learn).

振り返りやフィードバックを活かし次週へ

学び Learning

- アイデア出しなどのサービスの根幹に関わることを妥協しないことによって、納得のいくサービスができる
- 実現性や、サービスの必要性を考慮した上でのアイデア出しや選考の手法
- 当たり前を疑い、今一度考えることが新たなサービスのきっかけになる
- We don't compromise brainstorming of idea which and we can get satisfied ideas.
- Decide the feasibility and necessity of the service, and consider how to select ideas.
- Suspect common sense and reconsider are become trigger of new service.

- 早期の行動と計画を立てる重要性
- 役割を分担し、各々がそれを全うすることの大切さ
- チーム内での認識の差をなくすための定期的な話し合い
- 目標を立て、気を引き締めることの重要性
- 新たな視点を得るため、些細なことでもアウトプットをする大切さ
- It's important to act early and make a plan.
- It's important to divide roles and complete.
- Regularly meeting for matching up recognition with in the team.
- It's important to set goals and brace oneself.
- It's important to output small things for getting new view points.



図 A.2 プロジェクト概要 (最終成果報告会) (2/2 枚目)

付録 B 成果報告会で使用した本グループのポスター

2019.12.06 プロジェクト学習 成果発表会



Project No.10 ビーコン IoT で函館のまちをハックする

去りし想ひを乗せゆきて

市電で感じるノスタルジア、貴方とつむぐ三十一文字

Make Our Nostalgia at the Tram into 31 Letters.

Member 松山穂乃夏 宮越七菜 石倉愛朗 工藤大山

背景

- 函館は歌人石川啄木の「心のふるさと」と言われているが、函館市民に短歌はあまり馴染みがない
- 短歌は心情、情景を詠むものである
- 市電から見えるノスタルジックな景色は、短歌を詠むきっかけになる

目的

- 他の人とともにひとつの情景に向けた短歌を詠むことで、同じ景色を見た人の思いを知る
- 自分の関わった短歌がどうなるかなど、市電の乗車を“楽しみ”に変える

サービス

概要

- 同じ景色を見ていた人と、一つの短歌を作り上げるサービス
- 一人一句ずつ言葉を紡ぎ、5人で一首の短歌を完成させる

UX

- ビーコンの利用により、市電内から見た同じ景色への想いを共有できる
- 言葉を繋ぎ、誰かと一つの短歌を作りあげていく楽しみ



ユーザストーリー



市電に乗車してビーコンの範囲内に入ること、アプリが使用可能



ノスタルジックな沿線の街並みに心を打たれ、初めの一節を投稿



後日、再び市電に乗車し、アプリを起動すると…



自分が詠んだ一節からは想像もしていなかった歌ができていた！

機能



システム構成



学び

- チーム内のコミュニケーションや作成するものに対する共通認識の大切さを学んだ
- チーム内で今どんなことを開発しているのかの共有が不十分な点があった
- 実際の開発を通してツールや言語、開発手法への知識が深まった
- チーム内での技術の共有の重要性
- 不明な点を教え合うことが困難だったため

展望

- 通知を実装し、短歌の完成をお知らせする
- お店の新たな広告方法としての利用 (利用例)
 - 市電沿線のお店が、自慢の商品をテーマに短歌の上の句を投稿
 - 市電に乗車した人が感想や紹介を交えた下の句を詠む
 - 出来た短歌を見た人が、乗車のついでに立ち寄ってみたいくなるような宣伝効果を生む



図 B.1 去りし想ひを乗せゆきて (最終成果報告会)

参考文献

- [1] 古館達也, 堀川三好, 工藤大希, 岡本東: Bluetooth Low Energy ビーコンを用いた屋内測位手法に関する研究, 情報科学技術フォーラム講演論文集, 第 14 巻, 4 号, pp. 311-312(2015)
- [2] 函館市, (2014), 保存地区の保存に関する基本計画,
<https://www.city.hakodate.hokkaido.jp/docs/2014012200141/files/hozonkeikaku.pdf> (last accessed 2019/7/23)
- [3] 福田 周, (2017), 石川啄木—短歌に見る生と死の表現—, pp127-131
- [4] 大重美幸, (2017), 詳細! Swift 4 iPhone アプリ開発 入門ノート
- [5] Kotlin Bootcamp Course,
<https://codelabs.developers.google.com/kotlin-bootcamp/> (last accessed 2019/12/21)