

公立はこだて未来大学 2017 年度 システム情報科学実習  
グループ報告書

Future University Hakodate 2017 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

Project Name

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

グループ名

サプライズスナップ

Group Name

Surprise Snap

プロジェクト番号/Project No.

8-C

プロジェクトリーダー/Project Leader

1015253 橋場保鷹 Hodaka Hashiba

グループリーダー/Group Leader

1015062 古川晃也 Akiya Furukawa

グループメンバ/Group Member

1015062 古川晃也 Akiya Furukawa

1015176 荒駿一 Shunichi Ara

1015181 尾崎政亘 Masanobu Ozaki

1015208 雫石卓耶 Takuya Shizukuishi

指導教員

松原克弥 藤野雄一 鈴木恵二 奥野拓

Advisor

Katsuya Matsubara Yuichi Fujino Keiji Suzuki Taku Okuno

提出日

2018 年 1 月 19 日

Date of Submission

January 19, 2018

## 概要

本プロジェクトでは、ビーコンを使って街の状況を可視化することで、観光や街の課題を発掘し、その情報から新たな価値を創造して街に還元することを目的とする。函館の街中に多数のビーコンを配備することを想定して、街で暮らす人や観光客などを対象としたサービスの提案と実装を行う。サービスの提案にあたり、函館が持つ課題についての仮説を建てた上でフィールドワークを行い、観察やインタビューにより収集した情報をもとに複数のアイデアを提案した。さらに、提案したアイデアをブラッシュアップしてコンテスト形式のプレゼンテーションを行い、教員や外部協力企業からのフィードバックを参考にしつつ、交通、対話、観光体験の3つのテーマに関するサービスの開発を行うこととした。

( 文責: 橋場保鷹 )

函館は有名な観光地であり、函館山や八幡坂、五稜郭公園などの美しい撮影スポットが多数存在する。しかし、従来の写真撮影には、撮ることのできる写真の構図に限界があった。本グループでは固定位置からの突発的写真撮影により新たな観光体験を提供するサービス「サプライズスナップ」の提案、実装を行う。本サービスを利用することで、撮影スポットに適した構図で写真撮影を行うことが可能となる。また、サービス側から写真撮影を提案することで観光客に驚きの体験を付加し、さらに観光客同士のコミュニケーションを生み出すことで観光旅行の思い出をより濃いものとする。前期は、サプライズスナップ実装に向けて設計図や設計書の作成を行った。夏季休業は、実装に向けた学習とアプリケーションの大枠となるモックの作成を行い、後期は、サプライズスナップの全機能の実装と、利用者からのフィードバックを受け機能の修正を行った。

キーワード    ビーコン, 写真撮影, 観光, IoT

( 文責: 古川晃也 )

# Abstract

This project aims at creating brand-new values of Hakodate. To achieve that goal, we will discover problems by visualizing the current status of the city with beacons. To be specific, we proposed services for local residents and tourists and will develop them on the assumption that many beacons have been installed in Hakodate. We had made some hypotheses about problems of Hakodate prior to conduct fieldwork for creating service proposals. And we came up with some ideas based on results of observations and interviews in that research. We refined these ideas and presented them to our advisory professors and support companies for their feedback. Finally, we selected 3 services about transportation, communication, tourism.

( 文責: 橋場保鷹 )

Hakodate is one of the most famous sightseeing spots, there are many beautiful photography spots such as Hakodate Mountain, Hachiman saka Slope and Goryokaku Park. However, the composition of photographs which tourists take is limited. Therefore, this group proposes and implements the "Surprise Snap" service. It provides new tourism experience by sudden photographing from a ideal fixed position. By using this service, it is possible to take a picture with a good composition from a shooting spot. Suggesting photography from service makes surprising experience to tourists. Furthermore, this service make tourists' memories of sightseeing trips stronger by making communication among other tourists. In the first term, we created design drawings and design documents for the "Surprise Snap" implementation. In the summer holidays, we learned for implementation and creation of a mock which is the main body of the application, and in the latter term, we implemented implementation of all functions of surprise snap and received feedback from users to update it.

**Keyword** Beacon, Photography, Sightseeing, IoT

( 文責: 古川晃也 )

# 目次

<b>第 1 章</b>	<b>本プロジェクトの背景と目的</b>	<b>1</b>
1.1	背景 . . . . .	1
1.2	目的 . . . . .	1
1.3	ビーコンについて . . . . .	1
<b>第 2 章</b>	<b>グループ課題設定までのプロセス</b>	<b>2</b>
2.1	ロゴ作成 . . . . .	2
2.2	ビーコンに関する調査 . . . . .	2
2.3	フィールドワーク . . . . .	3
2.3.1	事前調査 . . . . .	3
2.3.2	地元の問題に関する調査 . . . . .	3
2.3.3	観光地の問題に関する調査 . . . . .	3
2.4	協力企業によるレクチャー . . . . .	3
2.5	開発するサービスの考案 . . . . .	5
2.5.1	アイデアソン . . . . .	5
2.5.2	グループによるブラッシュアップ . . . . .	5
2.5.3	アイデアコンテスト . . . . .	6
2.5.4	地元企業によるレビュー . . . . .	7
2.5.5	テーマ決定 . . . . .	7
<b>第 3 章</b>	<b>本グループについて</b>	<b>9</b>
3.1	背景 . . . . .	9
3.2	目的 . . . . .	9
<b>第 4 章</b>	<b>サブライズスナップについて</b>	<b>10</b>
4.1	概要 . . . . .	10
4.2	ユースケース . . . . .	10
4.3	機能 . . . . .	11
4.3.1	撮影スポットの検索 . . . . .	11
4.3.2	写真撮影の提案 . . . . .	11
4.3.3	写真撮影 . . . . .	11
4.3.4	履歴の表示 . . . . .	11
4.3.5	自発的な写真撮影の開始 . . . . .	11
<b>第 5 章</b>	<b>課題解決のプロセス</b>	<b>13</b>
5.1	到達目標の設定 . . . . .	13
5.1.1	本グループの目標 . . . . .	13
5.1.2	前期活動における目標 . . . . .	13
5.1.3	後期活動における目標 . . . . .	13

5.2	中間発表	14
5.2.1	発表内容	14
5.2.2	レビュー内容	14
5.2.3	改善に関する検討	15
5.3	札幌オープンキャンパス	15
5.4	仕様の作成	15
5.4.1	ユースケース図	15
5.4.2	アクティビティ図	16
5.4.3	機能一覧	16
5.4.4	画面遷移図	18
5.4.5	システム構成図	18
5.5	技術習得	19
5.5.1	環境準備	19
5.5.2	集中勉強会	21
5.5.3	各メンバーへ課した課題	21
5.6	実装	21
5.6.1	アプリケーション	21
5.6.2	API サーバ	23
5.6.3	カメラサーバ	27
5.7	ドキュメンテーション	28
5.7.1	Web サイト	29
5.7.2	リポジトリのドキュメント	29
5.8	デモ発表会	29
5.9	実装の改善	30
5.10	金沢工業大学教授に対しての発表	30
5.11	成果発表会	30
5.11.1	発表内容	30
5.11.2	レビュー内容	31
5.11.3	反省点	31
<b>第 6 章</b>	<b>各メンバーの役割と活動の振り返り</b>	<b>32</b>
6.1	役割分担	32
6.2	古川晃也の振り返り	32
6.3	雫石卓耶の振り返り	33
6.4	荒駿一の振り返り	33
6.5	尾崎政巨の振り返り	34
<b>第 7 章</b>	<b>まとめと展望</b>	<b>35</b>
7.1	前期の活動	35
7.2	後期の活動	35
7.3	活動の振り返り	35
7.3.1	本プロジェクトの目標	35
7.3.2	本グループの目標	35

7.4	サービスの展望 . . . . .	36
付録 A	中間発表会で使用した本グループのポスター	37
付録 B	成果報告会で使用した本グループのポスター	38

# 第 1 章 本プロジェクトの背景と目的

## 1.1 背景

近年、IoT やセンシングデバイス [1] といった技術の普及が進んでいる。総務省の調査によれば、世界におけるセンサーの出荷金額と出荷台数は年々増加し続けている。センシングデバイスの 1 つである BLE ビーコンは、Bluetooth Low Energy(以下、BLE) に対応したスマートデバイスがあれば誰でも活用が可能となるため、今後、ビーコンを活用した位置情報ベースのサービスやシステムの増加が期待できる。

( 文責: 橋場保鷹 )

## 1.2 目的

本プロジェクトは、ビーコンを用いて函館をより良くするサービスやシステムの提案、開発を行う。函館の街に約 1000 個のビーコンを配置すると想定して、それを活用したサービスを展開することによって、函館での暮らしを便利にしたり、新体験を創り出して函館の街を活性化したりすることが本プロジェクトの目的である。

( 文責: 橋場保鷹 )

## 1.3 ビーコンについて

ビーコンのひとつに、BLE を用いて情報を発信する機器がある。ビーコンの電波は、Bluetooth 4.0 以降に対応した端末で受信することが可能である。受信端末は、ビーコンの電波を通して、ビーコンまでの距離とビーコン ID や URL 等の少量の付加情報を取得することができる。ワイヤレスヘッドフォンなどの従来の Bluetooth デバイスとは異なり、ビーコンでは、ペアリングなしにデータを送受信することや、受信を契機としたユーザー通知の起動が可能である。現在普及している BLE ビーコンにはいくつかの規格があり、代表的な規格として、iBeacon、Eddystone と LINE Beacon がある。iBeacon は Apple 社が開発している規格であり、端末固有の UUID に加えて、任意に指定できるメジャー値とマイナー値、ビーコンと端末間の距離を図るための送信出力をデータとして発信することができる。Eddystone は Google 社が開発を行っている規格であり、iBeacon が持つ機能に加えて、URL 等の情報を付加することができる。LINE Beacon は、LINE 社が開発を行っている規格であり、スマートフォンアプリケーションである「LINE」へ広告を送信したり、プッシュ通知を送信したりすることが可能である。

( 文責: 橋場保鷹 )

## 第2章 グループ課題設定までのプロセス

### 2.1 ロゴ作成

本プロジェクトでは、プロジェクトの目的と対象の理解をすることとチームビルディングを目的としてロゴの作成を行った。ロゴの作成については、2週間の時間を要した。はじめに、本プロジェクトを説明するにあたって、どのような情報を伝える必要があるのかをブレインストーミングを用いて再確認した。再確認した情報をもとに、各自でロゴのデザイン候補を作成し、メンバー全員でレビューを行った。レビューでのコメントや他のメンバーのデザイン候補を参考に、各自でもう1つデザインを作成した。合計約30個のデザイン候補の中から、プレゼンテーションと多数決により、本プロジェクトのデザインの原案を決定した。その後、ロゴに関するワーキンググループを結成し、デザインの原案を改良しながらロゴデザインの最終版(図2.1)とそれを用いたポスターのテンプレート(図2.2)を作成した。

(文責: 橋場保鷹)

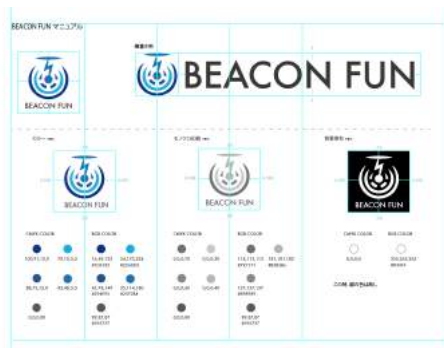


図 2.1 ロゴデザイン

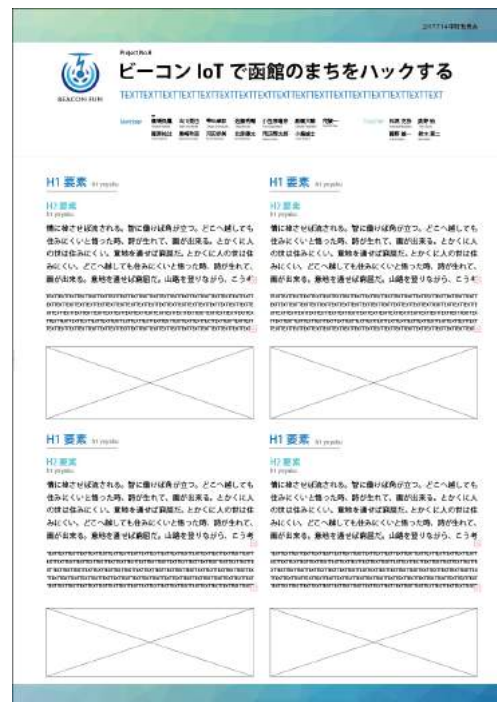


図 2.2 ポスターテンプレート

### 2.2 ビーコンに関する調査

本プロジェクトでは、ビーコンを用いた開発を行うため、ビーコンの仕様や事例について文献を用いて調査し、プレゼンテーション形式でメンバー間での共有を行った。文献調査には2日間の調査期間を設け、各自で書籍やインターネットから情報を収集して、発表を行うためのスライドを作成した。その結果、ビーコンでの開発環境、ビーコンの規格や詳細資料といった知識を得た。



## 2.3 フィールドワーク

### 2.3.1 事前調査

函館に存在する地域の課題や問題を調査するためにフィールドワークを計画した。問題を発見するためにどのような点に着目するべきかを3つのチームに分かれて話し合い、主にKJ法を用いて調査項目をまとめた。調査項目から、地元の問題を調査するグループと観光地としての問題を調査するグループに分担してフィールドワークを実施した。

( 文責: 橋場保鷹 )

### 2.3.2 地元の問題に関する調査

地元の問題を調査するグループは、主に公共施設へのインタビューを行い、地域が現状抱えている問題の洗い出しを行った。函館市青年センターとシエスタハコダテへ訪問を行い、インタビュー形式で地域が抱えている問題についての討論を行った。調査の結果(図 2.3)、函館は現在、土地が不足していて駐車スペースが足りないということがわかった。また、公共交通機関を積極的に利用してもらうことが駐車スペース不足の解消につながる、といった解決策を得ることができた。

( 文責: 河田歩美 )

### 2.3.3 観光地の問題に関する調査

観光地としての問題を調査するグループは、五稜郭やベイエリアといった有名な観光スポットへ訪問し、観光客へのインタビューや観察を行った。この時、五稜郭には日曜日と水曜日の2回訪れ、日曜日にはローカルなイベントが開催されていた。

調査の結果(図 2.4)、西部地区ではパンフレットや観光客向け情報誌の内容が分かりにくい、駅などにあるガイドの情報量が多すぎるといった問題を発見した。また、スマホを写真撮影に使っている観光客はいるが、他の用途で使っている人はあまり見かけなかった。対して五稜郭は、道路が入り組んでいるように見受けられ、慣れない人にとっては迷いやすく、観光に支障が出るのではないかと問題を発見した。加えて、観光客がアプリなどを使わずにインターネットで調べ物をしている様子が窺い知れた。また、日曜日はイベントがあったために人が多いのは当然と思われるが、イベントがない平日の水曜日にも観光客はおり、特にツアー客が多かった。西部地区、五稜郭の共通点としては、アジア人観光客やバス・市電の利用者が多いということが挙げられた。

( 文責: 小笠原瑠奈 )

## 2.4 協力企業によるレクチャー

ビーコンに関する開発やサービスの提供を実際に行っているトランスコスモス株式会社とTangerine 株式会社から、リモート会議サービスを介して遠隔でビーコンに関する2時間程度のレ

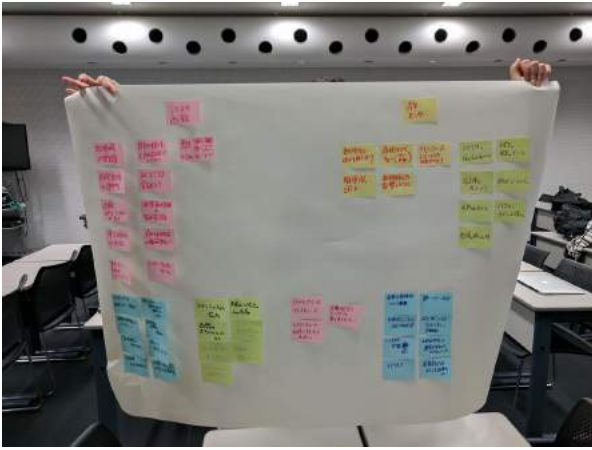


図 2.3 地元調査後のまとめ



図 2.4 観光地調査後のまとめ

クチャーを受けた。レクチャーは、「Introduction」,「Beacon History」,「Beacon Trend」,「Case Studies」,「Technical Challenges」,「Question & Free Discussion」から構成されていた。

「Introduction」では、企業の概要とプレゼンターのプロフィールの紹介を受けた。

「Beacon History」では、2013年にiBeaconがアナウンスされてから現在に至るまでの歴史についての説明を受けた。2015年にBluetoothをオンにしているユーザーが少なく、価値の低いサービスが増え、専用アプリのダウンロードが必要といった条件からビーコンが使われなくなった。しかし、2016年にBluetoothを用いたサービスの増加や、GPS等の位置情報サービスとは違ったエクスペリエンスを提供できるといった観点からビーコンが再注目されるようになった。

「Beacon Trend」では、ビーコンのこれからのトレンドについての説明を受けた。位置情報サービスの提供対象が「エリア」ではなく「人」へ遷移しているといわれており、主にリマーケティング用途でビーコンが注目されていることが明らかになった。

「Case Studies」では、実際にビーコンが導入された事例の紹介を受けた。アパレルショップにビーコンを設置し、来店を検知してノベルティを集めるサービスや、CDレンタルショップで、ユーザー自身のスマートフォンを用いた視聴サービスの提供が行われた。

「Technical Challenges」では、ビーコンの今後の課題についての説明を受けた。ビーコンは適正範囲で利用する必要があり、適正範囲を外れると、精度やエクスペリエンスにばらつきが生じる。また、アプリケーションを作成するにあたってAPIの制約があり、それが今後解消すべき課題である。

「Question & Free Discussion」では、プロジェクトメンバーからプレゼンターへ質問を投げかけた。グループ分けした質問内容と回答は以下の通りである。

江ノ島水族館公式アプリ「えのすい」について

- 回遊ルートが決まっているため、道案内というよりはスタンプラリー要素が大きい。
- 道案内は作る労力の割にはそんなに使わないだろうと考えた。先方のビジネス的な状況によっては作ることも視野に入る。
- 道案内をやっていた事例は他であるが、精度が悪く、案内サインでも十分代用できる。

ビーコンからの Advertise について

- iBeacon や Eddystone はあくまでも 1 つのサービスであり、GATT や独自プロトコルによるサービスを組み合わせることができる。

- iOS 側では iBeacon のパケットを Core Location が奪ってしまうが、それ以外であれば Core Bluetooth で拾うことができる。

#### LINE Beacon について

- 仕様は非公開である。
- LINE Beacon でサービスを提供することは、元のサービスである「LINE」の評価に直結してしまう。
- 電波強度と送信間隔の調整で、範囲を調整できる。

#### ビーコンのメンテナンスについて

- シンプルな構造のため、設置を行えば電池切れ以外はほぼメンテナンスが不要である。
- 設置の際は、外的要因による破損の対策をしなければならない。
- 該当等の公共設備に設置する場合には許可が必要である。

#### ビーコンの電力供給について

- ソーラー発電や振動発電を併用することができ、発電中のみ電波を発信する事もできる。
- シングルサービス以外ではチップ等のハードウェア的な改善が必要である。

#### 電波強度の調整について

- チップの仕様によって段階や最大強度が異なるが、基本的には SDK を通して操作が可能である。

( 文責: 橋場保鷹 )

## 2.5 開発するサービスの考案

### 2.5.1 アイデアソン

提案するサービスの決定に向けて、多くのアイデアをアウトプットするためにアイデアソンを 2 回実施した。アイデアソンの手法は、SPAJAM2017 札幌予選のアイスブレイクとして使われたものを参考とした。はじめに、1 チーム 3 人から 4 人のグループを 4 つ作成した。メンバー個人ごとに 8 分間でビーコンを用いたサービスの提案を行い、6 分間で他のグループメンバーと共有を行った。次に 5 分間でサービスの提案を行い、6 分間で共有を行った。最後に、3 分間でサービスの提案を行い、6 分間で共有を行った。徐々にアウトプットまでの時間を短くすることによってアウトプットが難しいことではないという意識付けと同時にテーマを設定する上で参考にできるアイデアのアウトプットを行った。2 回のアイデアソンを通して合計 96 個のアイデアを出すことができた。

( 文責: 橋場保鷹 )

### 2.5.2 グループによるブラッシュアップ

アイデアソンでアウトプットした 96 個のアイデアをもとに、グループワークを通してアイデアのブラッシュアップを行った。4 人 1 グループで 3 つのグループを作り、予め提案例としてプロ

ジェクトリーダーとティーチングアシスタントが作成したプレゼンテーションを参考にしながら、各チームごとに2つのアイデアのブラッシュアップを行った。ブラッシュアップ期間は1週間とし、途中で教員レビューや学生同士のレビューを行うことで作業を効率的に進めることができた。

( 文責: 橋場保鷹)

### 2.5.3 アイデアコンテスト

アイデアコンテストでは、ブラッシュアップした8つのアイデアと、提案例として作成したアイデア1つの合計9つあるアイデアについて、発表と評価を行った。発表は、1アイデアあたりプレゼンテーション時間が10分、コメント時間が5分に加えて、アイデアコンテストにあたり担当教員を招いた他、トランスコスモス株式会社と Tangerine 株式会社の方とリモート会議サービスを介して遠隔から発表を行った。また、オンラインの投票フォームを用いてアイデアに対する評価を収集し、「新しいUX/新体験」、「オリジナリティ」、「函館との親和性」、「ビーコンの活用」、「サービスの継続性」の5項目の観点で評価した。ユーザーの新体験を重視したサービスの提案を行うため、「新しいUX/新体験」を10段階、残りの項目を5段階として、評価項目に重みを付けた。

発表では「サプライズスナップ」、「Music Travel」、「MindEcho」、「きょう you」、「購入シェア」、「Hako-B」、「beacomm」、「函館ディスリガーデン」、「Contextual-指さし会話帳」の合計9つのアイデアを発表した。

「サプライズスナップ」は、観光客向けの写真撮影サービスである。カメラをビーコンとともに函館の街各所に設置しておき、ビーコンで観光客が通り過ぎるのを検知して自動で写真を撮影する。

「Music Travel」は、観光客向けの音楽共有サービスである。ビーコンとスピーカーを観光名所に設置し、そこにいる人の数や性質によって様々な音楽を提供する。

「MindEcho」は、観光客同士の感情を共有するサービスである。スマートフォンでジェスチャー入力を行うことによって、同じビーコンの周囲にいる人へ向かって感情をアウトプットする。

「きょう you」は、観光客同士の体験を共有するためのサービスである。他のSNSサービスとビーコンの情報を連動することで、自分の足跡や体験を他のユーザーに共有する。

「購入シェア」は、地元の買い物客同士を繋げることで商品を知ってもらうためのサービスである。ビーコンに登録された商品を宣伝することで、その商品を購入する際に割引が入る。後続の買い物客に宣伝を見せることでその商品に興味を持ってもらう。

「Hako-B」は、函館のバスを観光客や地元の人にわかりやすく利用してもらうためのサービスである。バスやバス停にビーコンを設置し、自分の行きたい場所によって適切にバスやバス停の位置を知らせてくれる。

「beacomm」は、観光客と地元の住民を繋げるためのサービスである。ビーコンの電波を用いて店の中でのみ参加ができるローカルなSNSを作ることで、顔を合わせずに地元の住民と観光客が意思疎通ができる。

「函館ディスリガーデン」は、函館に不満がある人と、函館をより良くしたい人をマッチングさせるためのサービス。函館に不満のあるユーザーがビーコンに問題を投稿し、それを見つけた函館をより良くしたい人が解決するといったきっかけづくりを行う。

「Contextual-指さし会話帳」は、外国人観光客のコミュニケーションを支援するサービスである。ビーコンの電波を検知し、今いる場所や経路からその場にふさわしいフレーズと翻訳を提供することで、店や交通機関等での会話を支援する。

表 2.1 が示すとおり、総合的に評価点が高かったのは「サプライズスナップ」であった。新体験

の点数が高く、今までにないサービスだったことが高得点の要因だといえる。また、Hako-B が函館との親和性やビーコンの特徴を活かしているか、継続性があるかといった評価が高かった。理由として、函館固有の悩みであるバスの事情を解消し、ビーコンならではのアプローチで解決ができると行った点が挙げられる。

( 文責: 橋場保鷹)

表 2.1 各アイデアと評価の平均

	新体験	オリジナリティ	親和性	ビーコンの特徴	継続性
サプライズスナップ	9.2	4.9	4.7	4.4	3.5
Music Travel	8.4	4.1	3.4	3.7	3.2
beacomm	7.2	3.7	3.2	4.0	3.3
きょう you	7.2	3.7	3.8	4.0	3.8
購入シェア	7.9	4.1	3.9	4.1	4.2
Hako-B	7.1	3.8	4.7	4.7	4.4
MindEcho	8.1	4.3	3.2	2.8	3.1
函館ディスリガーデン	8.6	4.8	3.9	3.9	3.7
Contextual-指さし会話帳	7.4	3.7	4.5	4.2	4.0

#### 2.5.4 地元企業によるレビュー

函館でもものづくり関係の業務を行っている企業を招いた「函館市異業種交流会」へ参加し、アイデアコンテストで評価が上位であった「サプライズスナップ」、「Hako-B」、「函館ディスリガーデン」の3つのアイデアを発表した。発表に用いたスライドは、アイデアコンテストで使用したスライドを3つ合わせて10分に収まるように簡略化したものを用いた。参加した目的は、函館の住民やIT技術者ではない人の視点から、テーマに対しての客観的な評価を得るためである。「サプライズスナップ」の改善につながると判断したレビューには「シャッターを自分で押せると良い」という意見があった。この意見はアイデアコンテストで言われた肖像権の問題を解決する一つの手法として参考にした。「Hako-B」の改善につながると判断したレビューに「バス停までの経路案内もしてほしい」という意見があった。「Hako-B」はバス停のわかりにくさの改善する目的で考案されたため、必要な機能であることから実装を検討した。「函館ディスリガーデン」の改善につながると判断したレビューに「クーポンや粗品などの用途があるものを報酬として用意すべき」という意見があった。報酬として、賞状を渡すなどの考案はあったが、それでは継続性の要素として弱いことがわかった。全体として、質問が途切れることなく寄せられ、ぜひ作ってほしいなど前向きな意見が得られた。

( 文責: 荒田啓太郎)

#### 2.5.5 テーマ決定

アイデアコンテストを通して得られたフィードバックをもとに、実際に開発を行うテーマの決定を行った。テーマの選択では、フィードバックで得られた評価を加味した上で、多数決により

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

「Hako-B」、「サプライズスナック」、「Contextual-指さし会話帳」をテーマとして決定した。開発を行うサービスの数は、人数や開発期間を考えた上で3つが妥当であるという意見で一致した。

( 文責: 橋場保鷹 )

## 第3章 本グループについて

### 3.1 背景

函館は有名な観光地であり、函館山や八幡坂、トラピスチヌ修道院などの美しい撮影スポットが多く存在する。その景観を楽しみに、多くの観光客が国内外から訪れ写真撮影を行う。

しかし、従来の写真撮影では、撮ることのできる写真の構図に限界があった。例えば、八幡坂での記念撮影を例に挙げて説明する。多くの場合、観光客は坂を背景に自分自身を撮影しようとする。手持ちのスマートフォンやカメラ、自撮り棒を用いて撮影した場合に坂を上手く撮影できない。地面から低い位置で坂と平行に撮影することになり、平地で撮影しているような写真となりがちである。坂であるということ表現するためには、ある程度離れた、地面から高い位置で撮影を行う必要がある。函館公式の画像ライブラリにもそのような写真が多い。また、観光地の写真撮影は単調でマンネリ化しやすいことも課題であると考えた。観光客は観光先各所で同様の写真撮影を行うため、撮影という体験の新鮮味は薄れていく。

( 文責: 古川晃也 )

### 3.2 目的

本グループの目的は、突発的な写真撮影により新たな観光体験を生み出すサービス「サプライズスナップ」の提案、開発を行うことである。

本サービスは、観光スポットの美しい構図の写真の撮影ができる位置に設置されたカメラを用いて撮影を行うことで、それぞれの撮影スポットに適した構図で写真撮影を行う。また、ユーザーに突発的な写真撮影を提案することで撮影に適した位置での記念撮影を逃さないようにすると同時に、従来と異なった新鮮な体験を生み出す。本サービスでは、観光客・団体の枠を超えた写真撮影を提案する。ユーザーは合同の写真撮影という体験を共有することで、観光客の間にコミュニケーションを行う。これにより、観光客はその時々写真撮影で新鮮な体験を得ることができる。

( 文責: 古川晃也 )

## 第 4 章 サプライズスナップについて

### 4.1 概要

サプライズスナップは、新しい形の写真撮影サービスである。函館を訪れた観光客が、観光地で記念撮影をする際に本サービスを利用する。従来の写真撮影の品質を高め、サプライズとコミュニケーションの体験を付加する。アプリケーションを通して、不特定多数のユーザーへ突発的の合同写真撮影を提案する。

本サービスでは、撮影スポットごとに設置された専用のカメラを通して、撮影を行う。そのため、その場所に適した構図で撮影を行うことができ、写真自体の品質の向上が見込まれる。

また、本サービスでは、ユーザーに突発的写真撮影を提案することで、驚きという体験を付加する。これにより、自主的に行っていた写真撮影という体験を、より濃いものにできる。

従来の写真撮影には、観光客・団体を越えたコミュニケーションは無かった。本サービスでは、観光客・団体の枠を超えた写真撮影を提案する。つまり、その場にいる全ての人を対象とした、合同の写真撮影を提案するということである。合同の写真撮影という体験を共有することで、観光客の間にコミュニケーションが生まれることが期待できる。これにより、観光自体の質の向上が見込まれる。

( 文責: 零石卓耶 )

### 4.2 ユースケース

始めに、ユーザーはアプリケーションをインストールし、Bluetooth を有効にしたまま観光を行う。ユーザーが撮影スポットに入ると、写真撮影を提案する通知がユーザーに送られる。ユーザーが通知に反応するとアプリケーションが起動し、写真撮影を行うための画面が表示される。ユーザーは写真撮影に参加するか否かを選択する。ユーザーが参加を選択すると、アプリケーションはユーザーへ「カメラの前に移動せよ」等の指示を出す(図 4.1)。その後、一定時間の経過後に撮影が行われる。

写真撮影が完了すると、アプリケーションの画面が撮影した写真の画像ダイアログ画面(図 4.2)へ遷移する。この画面では、端末上に写真を保存したり、QR コード画面(図 4.3)へ遷移したりすることが出来る。QR コード画面では、異なる端末間で画像を共有するための QR コードが表示される。

また、撮影スポットにいないときや、写真撮影が行われていないときにアプリケーションを起動すると、撮影履歴画面(図 4.4)が表示される。この画面では、各画像のダイアログ画面に遷移することが出来る。これによりユーザーは、任意のタイミングで写真のダウンロードや共有ができる。

また、突発的写真撮影は、同一のユーザーに対して撮影スポットごとに1度しか提案しない。つまり、ある場所で1度突発的な写真撮影を行った後は、もう1度その場所を訪れても突発的な写真撮影が提案されないということである。その代わりに、突発的な写真撮影を自発的に起動することの出来る機能(図 4.4 の左下のボタン)が有効になる。写真撮影を2回以上希望するユーザーは、この機能を用いることができる。



## 4.3 機能

### 4.3.1 撮影スポットの検索

一度起動されると、アプリケーションはバックグラウンドで撮影スポットの検索を行う。撮影スポットとは、本サービス専用のカメラが設置された場所のことである。端末の Bluetooth 機能を用いて、撮影スポットに設置されているビーコンを定期的に探す。

### 4.3.2 写真撮影の提案

アプリケーションが撮影スポットのビーコンを検出すると、サーバと連携して写真撮影を計画する。サプライズスナップの予定が確定すると、通知機能を用いて、ユーザーに写真撮影を提案する。ユーザーは写真撮影を了承するか、拒否できる。

### 4.3.3 写真撮影

ユーザーが写真撮影を了承すると、アプリケーションはカメラの前へのユーザーの誘導やカウントダウン、カメラに実際に写っている画像の表示等を行う。一定時間が経つと写真が撮影され、写真撮影に参加したユーザーへ配信される。

### 4.3.4 履歴の表示

サプライズスナップ時以外には、撮影履歴が表示される。いつ、どこで、どのような写真を撮影したか閲覧できる。また、任意の写真を端末上にダウンロードしたり、他の端末に共有するための QR コードを表示したり出来る。

### 4.3.5 自発的な写真撮影の開始

過去、その場所でサプライズスナップをしたことのあるユーザーは、その場所で自発的にサプライズスナップを起動できる。



図 4.1 サプライズスナップ画面.

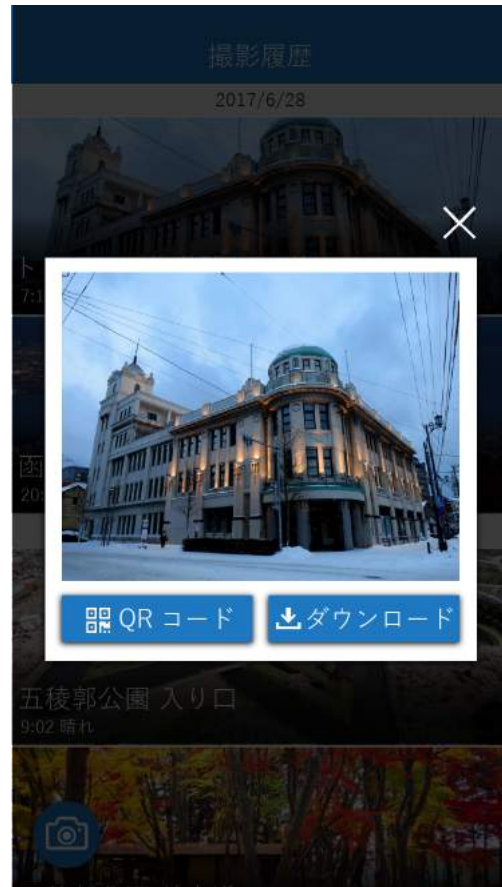


図 4.2 画像ダイアログ画面.



図 4.3 QRコード画面.



図 4.4 履歴画面.

## 第 5 章 課題解決のプロセス

### 5.1 到達目標の設定

本章では、本グループにおける通年、前期、後期それぞれの到達目標を述べる。

( 文責: 古川晃也 )

#### 5.1.1 本グループの目標

本プロジェクトの目標は、ビーコンを活用して函館の街を盛り上げられるようなサービスのプロトタイプを作成することである。本グループにおいては、「サプライズスナップ」のサービスのシステムを構築することと、誰もがこのサービスを再現できるようにドキュメント付きのソースコードを Web 上に公開することを通年の目標とする。

( 文責: 古川晃也 )

#### 5.1.2 前期活動における目標

本グループの前期の目標は、作成・公開するサービスを考案しコンセプトを考え、後期の開発に向けて設計を行うことである。まず、解決する問題を見つけ出し、どのようなサービスを作成すればその問題を解決できるかをいくつか考え出して実装するサービスを決定する。そして、そのサービスの仕様や利用する技術について明瞭にするため、システム設計図や画面遷移図、ユーザーの利用ワークフロー図を作成する。

( 文責: 古川晃也 )

#### 5.1.3 後期活動における目標

後期活動における目標は、考案したサービス「サプライズスナップ」を実装・公開することである。具体的には以下のことをすべて達成することが目標である。

1. サプライズスナップの設計書にある機能をすべて実装し、動作可能な状態とする。
2. サプライズスナップを学内や函館の観光地に展開し、ユーザーから得たフィードバックを参考にサービスの質の向上を図る。
3. サプライズスナップについての情報をまとめた Web サイトを公開し、誰でもサービスを再現できるようにする。
4. サプライズスナップの公開可能なソースコードをすべて Web 上に公開する。
5. 公開したりポジットリすべてにドキュメントを書き、誰でも修正が行えるようにする。

( 文責: 古川晃也 )

## 5.2 中間発表

### 5.2.1 発表内容

本プロジェクトではスライドを使用せずにポスターのみを用いて発表を行った。内容は以下の6項目であった。ポスターは、以下内容を含む構成とした。

1. 課題  
自撮りの構図では限界があり理想の構図で写真を撮ることが困難であり、また観光スポットごとに記念撮影を行うと段々とそれが作業のようになりマンネリ化してしまうという説明
2. 提案  
最良の位置から突発的の写真撮影を行うことで、観光客に構図が美しい写真と驚きの体験を提供するという説明
3. アプリケーションの概要  
「設置してあるカメラの撮影スポットに入るとアプリケーションがビーコンの電波を検知する」、「複数人で最高の構図で写真を撮影する」、「その場でコミュニケーションをしながら写真を共有する」というアプリケーションの利用の流れについての説明
4. 画面遷移  
アプリケーションを構成する主な画面である、デフォルトの状態である履歴画面。撮影スポットに入った場合に切り替わる撮影画面。撮影が終了したときに切り替わる写真詳細画面。QRコードボタンを押下したときのQRコード画面の説明
5. システム  
サービスを構成するサプライズスナップのアプリケーション。撮影するためのカメラとそれを制御する Raspberry Pi 2。ビーコンを監査、制御するための TangerineHub。それら3つを統制するクラウド上のサーバの4つのサブシステムについての説明
6. 今後の予定  
10月に第1バージョンの作成を行い、11月に実際に学内などに設置しフィードバックをもらいながらアプリケーションの改善を行うという説明

( 文責: 尾崎政亘 )

### 5.2.2 レビュー内容

以下に中間発表で行った「発表技術」と「発表内容」に対して受けたコメントを抜粋して紹介する。

- 声がかぶっていて聞きづらかった
- 発表を3グループ全て見せるようにするといいと感じた
- ポスターだと見づらかったので、スライドを作成したほうが良かった
- サーバと携帯間に時差が生じるため考慮したほうがいい
- 人数や季節によって「最高の構図」がかわるのではないかと

点数による評価では、「発表技術」の平均評価が7.3点、「発表内容」の平均評価は7.25点であった。

### 5.2.3 改善に関する検討

前節で示したコメントから、発表する方法と内容について改善が必要であると考え。発表方法は、全体の説明を行ってから3つのグループに分かれて発表を行うと声が重なってしまうために聞こえにくくなることがわかった。またポスターだとプロジェクトの説明を1枚にまとめるため見辛くなってしまいが、スライドでは多くの枚数に切り分けて説明が可能になるため、ポスターだけではなくスライドを使用したほうが良いと気づいた。最終成果発表の時には、3つ同時に発表を行うのではなく全てを順番にまとめ、スライドの使用を検討する。発表内容について2つの点を指摘された。1つ目はサーバと携帯間に時差が生じ写真をダウンロードするときに時間がかかってしまうのではないかというコメントである。この指摘に対しては、プロトタイプを作成し評価することとする。2つ目は本システムが想定する「最高の構図」は状況によって変化するという指摘である。季節や人数によっても「最高の構図」が変化していくため、ディープラーニングなどを用いて機械学習を行い人数や季節によって構図を変化させることを検討することとする。

また、「アイデアは面白い」や「記念撮影の問題点を解決できる」などアイデアを評価するコメントもあった。これからは、これらの意見を参考に評価の高い特徴を維持しつつ、指摘を受けた点を改善できるように開発を進める。

( 文責: 尾崎政巨 )

## 5.3 札幌オープンキャンパス

9月8日・9日の2日にかけて、札幌で行われた「企業交流会およびオープンキャンパス in 札幌」(以下札幌オープンキャンパス)にて、ポスターセッションを行った。ポスターは中間成果発表会にて用いた全体ポスターを使い、本プロジェクトの概要を伝えた。8日は、企業向けにホテルさっぽろ芸文館にて企業に対してポスターセッションを行った。企業の方々にプロジェクトの活動を伝えることができた。9日には、高校生などを含んだ一般人に対して、プロジェクトの紹介をおこなった。未来大を希望している高校生や、活動に興味のある方などとの交流ができた。

( 文責: 荒駿一 )

## 5.4 仕様の作成

### 5.4.1 ユースケース図

開発者がユーザーに機能的要求を確認するためにユースケース図(図5.1)を作成した。アクターである観光客はアプリケーションを起動してスポットの検索を行う。スポットに入ると、ユーザーは撮影の提案を受ける。ユーザーは、撮影が終了すると写真を保存するかを選択できる。また、同じスポットに入る場合ユーザーが自発的に写真を撮ることも可能となる。

( 文責: 尾崎政巨 )

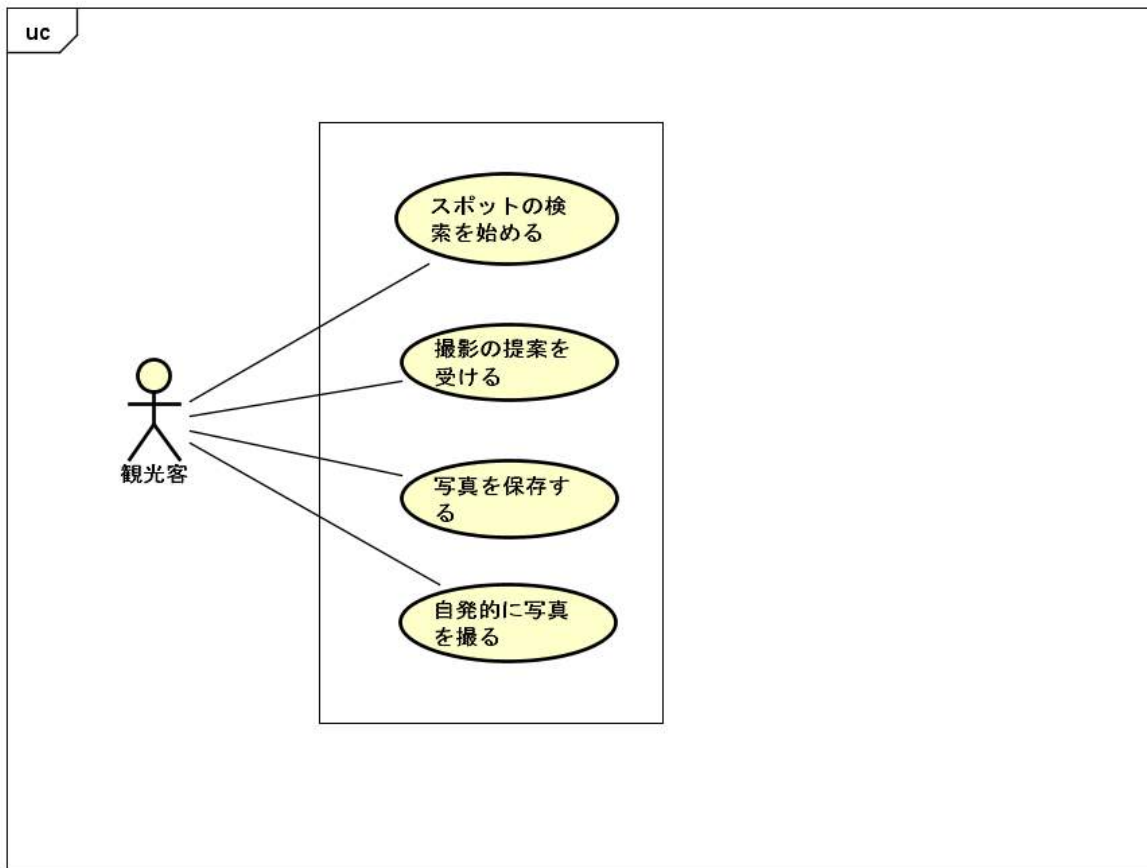


図 5.1 ユースケース図

### 5.4.2 アクティビティ図

ユーザーがアプリケーションを使用した際に、それぞれの機能が実行される順を理解し開発を進められるようにするため、アクティビティ図 (図 5.2) を作成した。ユーザーがアプリケーションを起動するとアプリケーションは端末の Bluetooth が有効か調べ、有効ではなかった場合にはユーザーに有効するよう通知する。アプリケーションは撮影スポットに入ったことを確認したときに、撮影画面に切り替わりユーザーが撮影するか判断し、撮影を許可した場合には撮影が開始される。撮影が終わるとユーザーは写真を保存するか QR コード画面で共有するか判断できる。

( 文責: 尾崎政巨 )

### 5.4.3 機能一覧

サービスの機能と概要を表すために実装機能を作成した。

#### 1. Bluetooth 通知機能

アプリケーション起動時、Bluetooth 機能がオンになっているか確認する。オンでなかったとき、ユーザーへオンにするようにアプリケーションがユーザーへ促す。

#### 2. スポット検索機能

アプリケーションがバックグラウンドで撮影スポットの検索を行う。撮影スポットとは、本

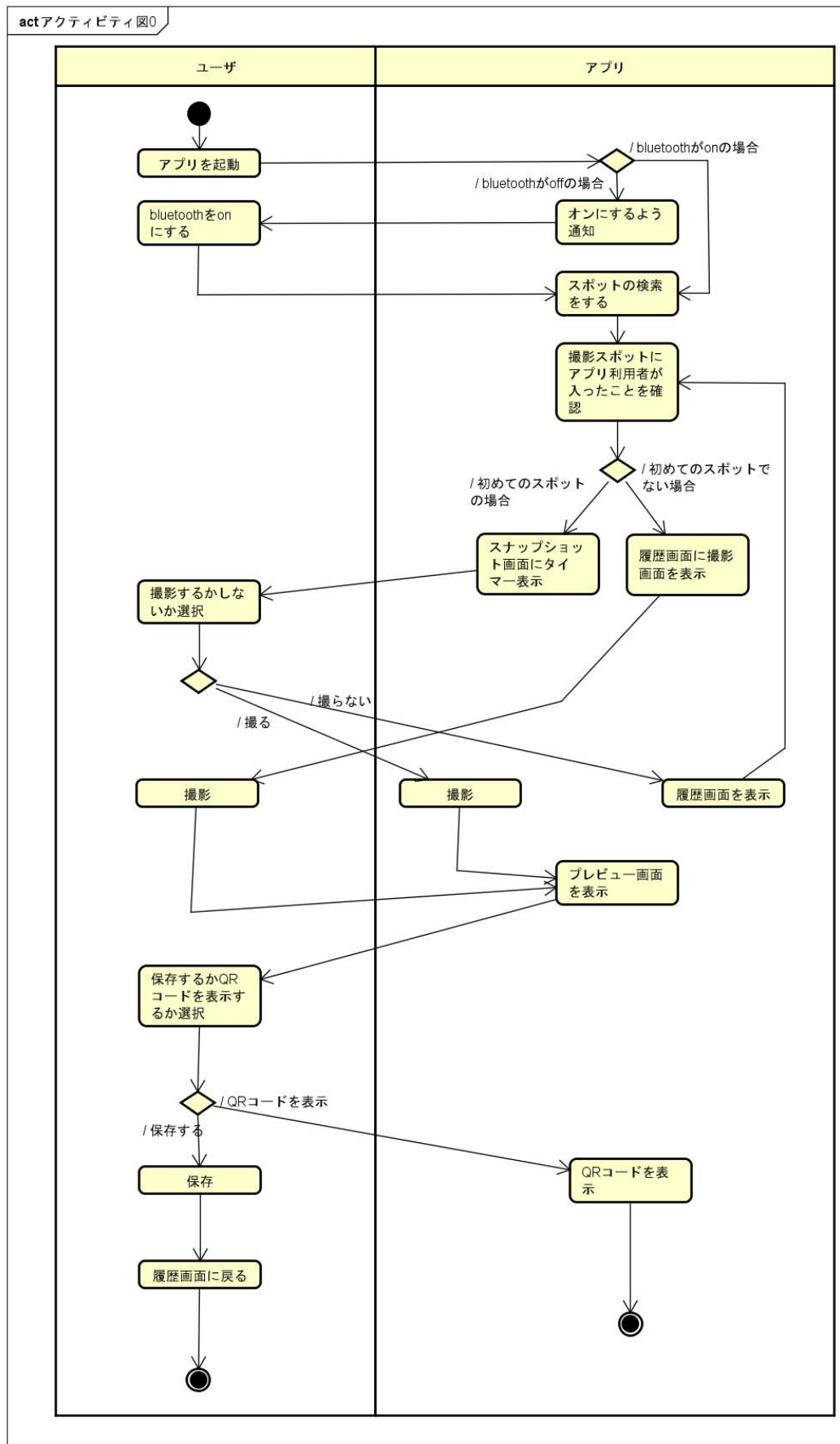


図 5.2 アクティビティ図



サービス専用のカメラが設置された場所のことを示す。

3. スポット検出機能

撮影スポットに設置してあるビーコンから発せられる電波を端末の Bluetooth 機能を用いて探し出す。

4. 撮影許可機能

ユーザーが撮影スポットへ入ったときに、突発的の写真撮影の提案が行われ、許可した場合は撮影が開始される。

5. カウントダウン機能

写真撮影が行われるまで端末にカウントダウンを表示し、ユーザーにあと何秒で撮影が開始されるか伝える。

6. 撮影機能

カウントダウンが終了すると撮影が開始される。

7. 保存機能

撮った写真を端末上に保存できる。

8. QR コード表示機能

QR コードを表示させ、アプリケーションを持っていない人とでも共有できるようにする。

9. 履歴表示機能

撮った写真を履歴として表示させ、いつ、どこで撮ったのかを振り返ることができる。

( 文責: 尾崎政巨 )

#### 5.4.4 画面遷移図

画面がどのような順序で表示されるか、あるいは画面同士がどのような関連性を持っているのかを示す画面遷移図 (図 5.3) を作成した。ユーザーが撮影範囲内に入っていないときは履歴画面を表示する。履歴画面では、自分の今までに撮ってきた写真が一覧され、思い出を振り返ることができる。撮影範囲内に入ると撮影画面に切り替える。この画面では、撮影までのカウントダウンと自分がカメラからどのように写っているのか確認できる。撮影が終了すると、写真詳細画面に切り替え、撮った写真をダウンロードと QR コード画面への遷移を可能にする。QR コード画面では、写真のダウンロードリンクへの QR コードを表示して、他の人との写真の共有が行える。

( 文責: 尾崎政巨 )

#### 5.4.5 システム構成図

本サービスを実現するためにどのような技術が必要かを明確にするため、システム構成図 (図 5.4) を作成した。システムは、大きく分けて 3 つのサブシステムで構成する。

1. アプリケーション

ユーザーが自身のスマートフォンにインストールして利用するアプリケーションである。設置してあるカメラの撮影範囲に入ったことをビーコンから発信される BLE 電波を検知し、サーバに送信する。ユーザーが撮影範囲に入ったことをクラウドから通知されると写真撮影を行い、その撮影した写真のサーバへのアップロードも行う。



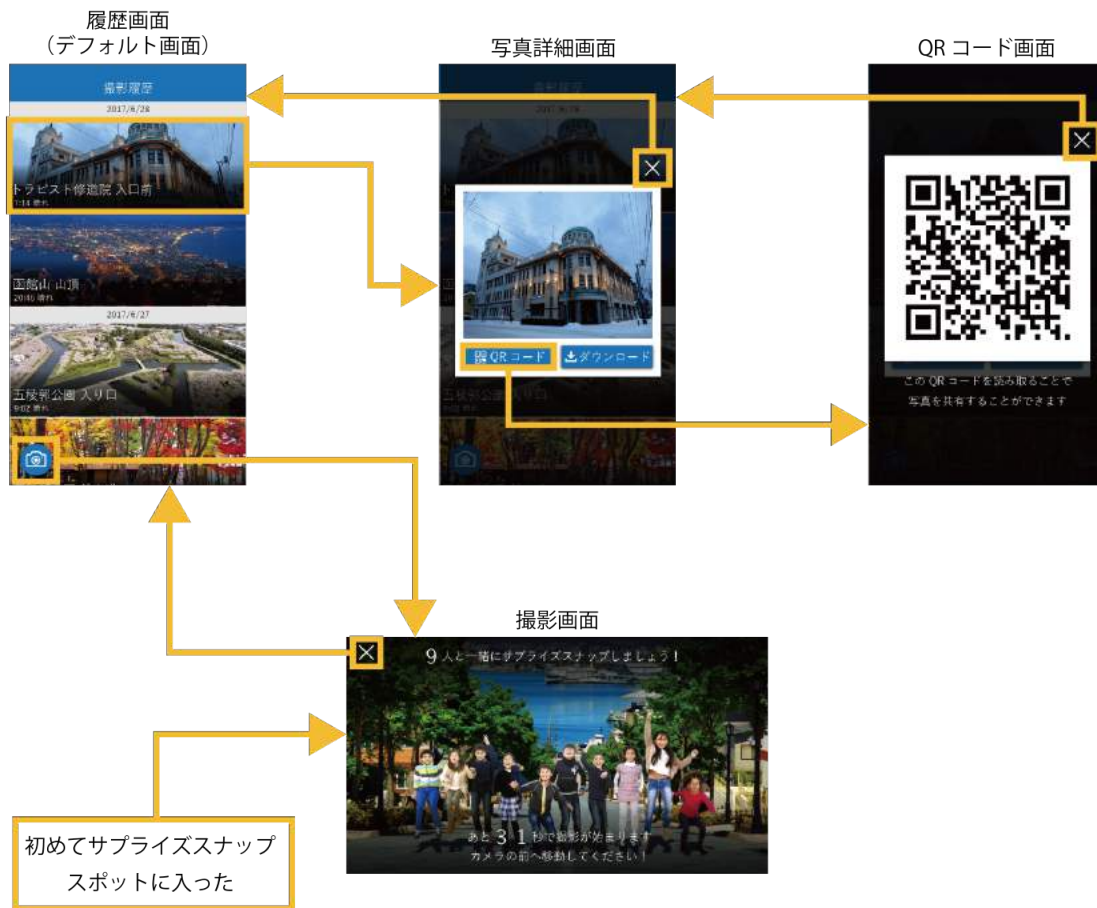


図 5.3 画面遷移図

## 2. API サーバ

クラウド上に構築するサーバである。プログラミング言語 Python と、マイクロ Web フレームワーク Flask と、データベース MySQL とで構成される。このサーバが、次に説明する 2 つのサブシステム (アプリケーション、カメラサーバ) とやりとりし、制御・同期を行う。また、設置しているカメラのプレビュー映像をリアルタイムで表示し、撮影した写真のダウンロードと保存も行う。

## 3. カメラサーバ

実際に写真撮影を行うカメラと、そのカメラを制御する Raspberry Pi 3 からなるシステムである。API サーバからの指示に従い、写真の撮影を行う。

( 文責: 古川晃也 )

## 5.5 技術習得

### 5.5.1 環境準備

- Slack

Slack は Slack Technologies, Inc. が運営しているチームコミュニケーションツールである。グループメンバーとのリアルタイムなチャットや通話が可能であること、後述する Asana

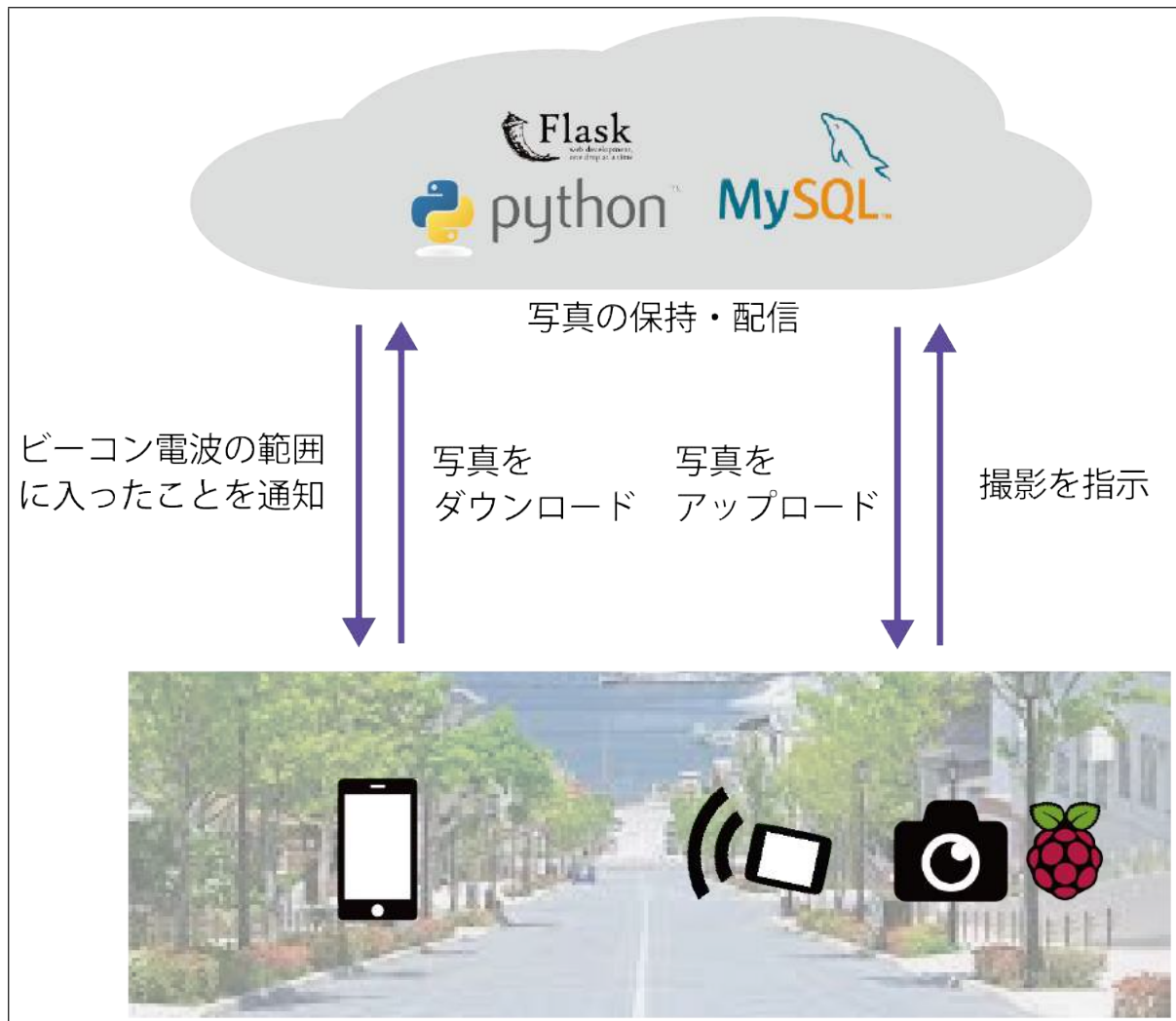


図 5.4 システム構成図

や Instagantt、GitHub との連携が可能であることから利用した。

- GitHub

GitHub は GitHub Inc. が運営しているソフトウェア開発プロジェクトのための共有ウェブサービスである。プロジェクトで開発するプログラムや報告書などのリポジトリの共有やタスクの管理に用いた。Slack や Asana と連携して利用した。

- Asana

Asana は Asana Inc が運営しているプロジェクト管理アプリケーションである。GitHub や Instagantt との連携を行うことができ [2]、GitHub に登録されたタスクが自動的に Asana に追加され、Instagantt にも反映されるように設定して利用した。

- Instagantt

Daniel Guajardo Kushner によって作成された Asana と連携して用いるガントチャートサービスである。Asana と連携させ、タスクの進行状況や期限までの日程を可視化するために利用した。

( 文責: 荒駿一 )

### 5.5.2 集中勉強会

夏季休暇前に、基本的な知識の獲得と学習すべき項目の洗い出しのために7月4日、7月5日の二日に渡って集中勉強会を行った。アプリケーションの開発を行う古川・荒と、APIサーバ・カメラサーバを開発する零石・尾崎の2班に別れ、それぞれ担当する技術についての学習を行った。具体的には、アプリケーションサイドはAndroidについての学習を行うため、Androidについての書籍を読みながら簡単なTodoアプリケーションを作成した。Todoアプリケーションとは、項目の登録・一覧・削除が行えるアプリケーションである。APIサーバ・カメラサーバサイド班は、Flaskのドキュメントを読みながら、Todo Webアプリケーションを作成した。

( 文責: 荒駿一 )

### 5.5.3 各メンバーへ課した課題

各自が夏季休暇終了までに行ってくる課題を設定した。

古川

Androidの入門書である「TECHNICAL MASTER はじめてのAndroidアプリ開発」[3]の読了とアプリケーションのモックの作成

荒

Android開発勉強のため、古川と同じAndroid入門書のサンプルアプリケーションの開発

尾崎

Pythonの入門書の読了

零石

APIサーバのモックの作成

( 文責: 荒駿一 )

## 5.6 実装

サブライズスナップのサービスは、5.4.5で述べたアプリケーション、APIサーバ、カメラサーバから構成される。以下で、それぞれのシステムの実装について説明する。

( 文責: 古川晃也 )

### 5.6.1 アプリケーション

概要

アプリケーションは、本サービスを利用するためにユーザーがAndroid端末にインストールして用いるAndroidアプリケーションである。ユーザーはこのアプリケーションを通して撮影の開

始、撮影画面のプレビュー、撮影した写真の保存、撮影した写真の共有を行う。

#### 1. 写真の撮影

アプリケーションをインストールすると、常にバックグラウンドで周囲のビーコンの情報を探る。ビーコンの電波を検出すると、ビーコンの UUID を取得し本サービスの利用 UUID と一致するか確認する。一致した場合はメジャー値とマイナー値を API サーバの /get\_place エンドポイントに問い合わせで撮影場所を特定する。それと同時に、/get\_visitor にて撮影場所での撮影の開始をサーバに指示し、撮影画面のプレビューに移る。

#### 2. 撮影画面のプレビュー

撮影開始から撮影直前までの間、カメラから見えている映像をユーザーに写す。その瞬間にカメラに写っている画像を返す API サーバの /get\_live\_view に 1 秒間に 10 回リクエストを送り、取得した画像を即時に画面に反映させることで、リアルタイムに近いプレビューを実現している。

#### 3. 撮影した写真の保存

撮影された写真は API サーバ上に保存される。撮影画面のプレビューが終了し次第、API サーバの /get\_snap\_state に問い合わせを繰り返す。API サーバから撮影の終了が確認されたら、/get\_thum に問い合わせを行い圧縮されたサムネイル画像を取得し、端末のデータベースにバイナリで保存する。

#### 4. 撮影した写真の共有画像ダイアログ画面に表示されている QR コードボタンを押すと、サーバ上に保存されている高画質な画像への URL を QR コードに変換した画像を生成し、表示する。

( 文責: 古川晃也 )

### 利用した技術

#### 1. Android

Android は、Linux カーネルをベースとした携帯端末用の OS である。他の携帯端末のプラットフォームには、アップル社の iOS、マイクロソフト社の Windows 10 Mobile、Mozilla の Firefox OS などがある。Android の大きな特徴として OS がオープンソースであるという点がある。本プロジェクトで用いるビーコンは、BLE を利用して端末の物理的な機能を用いるため、OS のソースコードを含めて情報がより多く公開されていてトラブルシューティングをしやすいと考えた。また、古川と雫石が Android 開発をした経験があること、大学の講義でグループメンバー全員が Java 言語を習得している点も加味した。

#### 2. iBeacon

iBeacon はアップル社が開発している BLE ビーコンの一種である。同様な規格には Google 社が開発を行っている Eddystone がある。Eddystone は iBeacon の持つ情報に加え URL などの情報も発信することができる。本サービスにおいては、iBeacon の発信する三種類の値があれば成立するため、シンプルな iBeacon を用いて開発を行った。ビーコンに UUID、メジャー値、マイナー値を設定でき、BLE の電波にこれらの情報を乗せることができる。本サービスでは UUID をサービスに利用しているアプリケーションかどうかの判別に用いて、メジャー値とマイナー値の組み合わせに場所を対応させることで撮影範囲に入ったかと、どの撮影箇所なのかを iBeacon を用いて判別できるようにしている。

#### 3. Android Beacon Library

Android でビーコンを検出するための代表的なライブラリである。周囲の iBeacon の BLE を検知し、そのビーコンのメジャー値・マイナー値を取得するのに利用した。

#### 4. OkHttp

OkHttp は Java の HTTP クライアントライブラリである。応答キャッシングによって、同一のリクエストをキャッシュして高速化できることや、ネットワークが一時的な障害に陥った際に障害回復まで自動的に待つといった特徴がある。開発当初は利用していた Android についての書籍で利用されていた HTTP 通信ライブラリ Volley を利用していたが、OkHttp では同じ処理を行うコードの行数が半減したことから、OkHttp を利用するに至った。

#### 5. ZXing

1次元および2次元バーコードのエンコードやデコードをサポートする Java ライブラリである。商品用バーコードや QR コードをはじめとする様々な規格に対応している。本サービスにおいては、撮影した写真データの URL を QR コード画像に変換するため利用した。

( 文責: 荒駿一 )

## 5.6.2 API サーバ

### 概要

API サーバには、大きく分けて2つの役割がある。

1つ目の役割は、ユーザーやサプライズスナップ、撮影スポット、ビーコン、カメラサーバ等の情報を管理することである。サプライズスナップのサービスを実現するためには、例えば、「だれが、いつ、どこで、サプライズスナップを行ったか」や、「どのビーコンとカメラサーバが、どの撮影スポットに対応しているか」といった情報が必要である。API サーバは、これらの情報を一元管理している。管理するとは、これらの情報をデータベースに記録し、適切なタイミングで追加、更新、削除して行くことである。データベースの構造(図 5.5)と各エンティティは次の通りである。

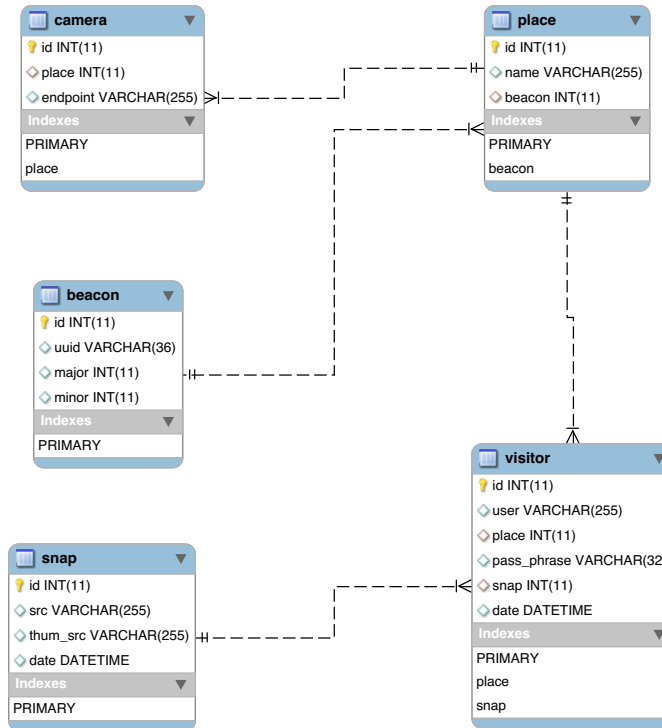


図 5.5 API サーバのデータモデル.

- beacon エンティティ  
ビーコンの情報を表す。ビーコンの UUID やメジャー値、マイナー値を持つ。
- camera エンティティ  
カメラ、カメラサーバの情報を表す。そのカメラが属する撮影スポットの情報や、カメラサーバのエンドポイントのアドレスを持つ。
- place エンティティ  
撮影スポットの情報を表す。その名前と、属しているビーコンの情報を持つ。
- snap エンティティ  
サプライズスナップの情報を表す。写真やサムネイル画像の内部パス、サプライズスナップが行われた時刻の情報を持つ。
- visitor エンティティ  
撮影スポットへ訪れたユーザーの情報を表す。また、ユーザーが撮影スポットへ確かに訪れたという証明書としても機能する。ユーザーの名前、訪れた場所、visitor 毎に推測不可能なデータ、ユーザーが参加したサプライズスナップ、訪れた時刻の情報を持つ。

2 つ目の役割は、スマートフォンアプリケーションやカメラサーバと通信しながら、サプライズスナップのサービスを実現することである。具体的には、次のような役割、機能がある。

- サプライズスナップの計画
- カメラサーバへの撮影指示
- 写真の保持・配信

- ライブビューの配信

これらの機能を外部から利用するために、API サーバは、複数のエンドポイントを公開している。エンドポイントとは、ある情報を与えると、各エンドポイントに対応した処理が実行され、その結果を受け取ることができるものである。API サーバは、次の 13 個のエンドポイントを持つ。

- /get\_visitor  
ユーザーとビーコンの情報を与えると、API サーバはサプライズスナップを計画し visitor (前述) を返す。ユーザーが撮影スポットに入り、それを API サーバに通知するために用いる。
- /delete\_visitor  
visitor を与えると、API サーバはその削除を試み、それが成功したか、失敗したかを返す。ユーザーがサプライズスナップを拒否し、それを API サーバに通知するため等に用いる。
- /get\_snap\_state  
visitor を与えると、API サーバはサプライズスナップの状況を返す。具体的には、サプライズスナップが行われる時刻、それが進行中なのか、完了しているのかという情報、それに参加している人数等を返す。スマートフォンアプリケーションが、サプライズスナップの情報を表示したり、その完了をポーリングしたりするために用いる。
- /get\_place  
ビーコンの情報を与えると、API サーバは、それに対応する撮影スポットの名前を返す。スマートフォンアプリケーションが、写真一覧画面等において、撮影スポットの名前を取得するため等に用いる。
- /get\_image  
visitor を与えると、API サーバは、それに対応するサプライズスナップの写真を返す。もし visitor が不正なものなら、HTTP 404 を返す。ユーザーが写真のダウンロード機能を利用するために用いる。
- /get\_thum  
visitor を与えると、API サーバは、対応するサプライズスナップのサムネイル画像を返す。もし visitor が不正なものなら、HTTP 404 を返す。スマートフォンアプリケーションが写真一覧画面等で、サムネイル画像を表示するため等に用いる。
- /add\_camera  
カメラサーバの情報を与えると、API サーバはその情報を記録し、それが成功したか、失敗したかを返す。カメラサーバが起動し、自身を API サーバに登録するために用いる。
- /post\_image  
サプライズスナップの写真やサムネイル画像等を与えると、API サーバはそれを記録し、それが成功したか、失敗したかを返す。カメラサーバが撮影を完了し、それを API サーバに送信するために用いる。
- /post\_live\_view  
ライブビューの画像を与えると、API サーバはその情報を保持し、それが成功したか、失敗したかを返す。カメラサーバがライブビュー画像の配信を行うために用いる。
- /get\_live\_view  
visitor を与えると、API サーバは、それに対応した撮影場所のライブビュー画像を返す。スマートフォンアプリケーションが、ライブビュー画像を取得するために用いる

以下のエンドポイントは開発用のものである。開発者が上記のエンドポイントのテストやデバッグのために使用する。開発時にのみ有効であり、本番環境では HTTP 404 を返す。

- /  
アクセスすると、各 API をテストするためのツールや、データベース内の情報を閲覧するためのリンクなどが表示される。
- /models/<model>  
アクセスすると、DB にある全ての<model>を返す。
- /models/<model>/<id>  
アクセスすると、<id>を持つ、<model>を返す。

### 利用した技術

プログラミング言語に Python を利用した。なぜなら、習得の容易さや後述する Flask の使用を想定したからである。様々な事例から Python は、Web アプリケーション開発の入門に適しており、習得が容易であると考えた。技術の習得にかかる時間を短縮することは、限られた期間で開発する上で重要だと考えたため、Python を採用した。

Web アプリケーションフレームワークに Flask を利用した。その理由は、習得が容易で、かつ十分な機能を持っていることや、WebSocket の利用に関する多くの採用事例があること等である。Flask は今回の規模に最も適した Web アプリケーションフレームワークである。競合するものに、django、Bottle、Tornado 等があるが、これらは適切ではないと考えた。django や Tornado は機能が多く、学習に時間がかかってしまい、Bottle は機能が逆に少なすぎるからである。今回、WebSocket を用いることは、開発期間の関係で出来なかったが、将来的に必要となる可能性が高い。例えば、サービスのリアルタイム性を高め、通信の無駄を排除する場合に必要である。

データベースに MySQL、これを Python から操作するための ORM に SQLAlchemy を利用した。その理由は、多くの採用事例があることである。そのため、Web 上にたくさんの技術的情報が存在しており、限られた期間で効率よく開発を行うことができた。

API サーバからカメラサーバの通信経路の確立に、OpenSSH による SSH リモート・ポートフォワーディングを利用した。SSH リモート・ポートフォワーディングとは、クライアントからサーバに SSH 接続を行い、サーバからの所定の packets をクライアントに転送する事である。提案サービスの実装には、API サーバからカメラサーバへの通信が必要である。しかし API サーバが、カメラサーバに直接アクセスすることはない。なぜなら、これを可能にするには、カメラサーバをインターネット上に公開しなければならないからである。そのためには、カメラサーバが設置される撮影スポットのネットワークに関する、高度な設定が必要になり、提案サービスの導入の敷居を上げてしまう。SSH リモート・ポートフォワーディングを用いることで、カメラサーバをインターネット上に公開しなくとも、API サーバからカメラサーバへ通信することが可能になる。カメラサーバが SSH で API サーバに接続し、API サーバはこの接続を用いてカメラサーバに接続を行う。OpenSSH を用いた理由は、手軽に利用できることである。OpenSSH はほとんどの Linux ディストリビューションに標準で含まれており、特別なセットアップをしなくても利用できた。

アプリケーションの開発とデプロイのために Docker を利用した。その理由は、開発と本番の環境が多様であったからである。開発環境は、Windows と macOS、本番環境は Linux であった。Docker を用いて、仮想化された Linux コンテナ上で開発することで、これらの環境の違いを吸収できた。また環境が記述された Dockerfile 等を、Git を用いてソースコードと共に管理できた。そ



のため、環境の違いによるアプリケーションの不具合を解消することができ、効率の良い開発が可能だった。

ログの解析、可視化に fluentd、Elasticsearch、kibana を利用した。その理由は、無償であることや、採用事例が多いことである。これらの技術を利用するまでは、Docker のログ表示機能を用いてログを確認していた。しかしこれでは、ログを閲覧するためにサーバにログインしなくてはならない。そのため、これらの技術を採用し、Web 上からログを閲覧できるようにした。fluentd でログを収集し、Elasticsearch でログの保存、検索を行い、kibana でログの可視化を行った。競合する技術はいくつかあったが、有償であったり、採用事例と情報が少なかつたりしたので、これらの技術を採用した。

( 文責: 雫石卓耶 )

### 5.6.3 カメラサーバ

#### 概要

カメラサーバの役割は、カメラを操作し、写真の撮影やライブビューの配信を行うことである。API サーバから撮影指示を受けると、一定時間撮影を待機して、カメラへ撮影指示を出す。その後、写真を取得して API サーバへ送信する。撮影を待機している間は、カメラからライブビューを受信し、API サーバへ送信する。カメラサーバも API サーバと同様に、次の 2 つのエンドポイントを持つ。

- /set\_supsnap  
撮影待機時間を与えると、カメラサーバが撮影開始の処理を行い、それが成功したか、失敗したかを返す。API サーバがカメラサーバへ撮影を指示するために用いる。
- /get\_image  
カメラ内部のファイル名を与えると、それに対応した画像を返す。本番環境では利用されず、開発者がカメラのテストやデバッグをする際に用いる。

#### 利用した技術

プラットフォームとして Raspberry Pi 2 を用いた。その理由は、多くの採用事例があることである。そのため、Web 上にたくさんの技術的情報が存在しており、効率良く開発を行うことができた。また、第 3 者がこのサービスを導入する際、このプラットフォームのセットアップを行う必要がある。このサービスをオープンソースとして公開するということを考えると、導入の敷居を下げることは非常に重要なことだと考える。

Web アプリケーションの構築、カメラ制御のためのプログラミング言語に Python、Web アプリケーションフレームワークに Flask を用いた。その理由は、前述した API サーバにおける理由と同様である。

カメラに OLYMPUS AIR A01 を用いた。その理由は、画質が良いこと、オートフォーカスや、ライブビューのストリーミング等の機能が十分なこと等である。カメラの選定にあたって、当カメラと競合したのは、Raspberry Pi Camera であった。Raspberry Pi Camera は、Raspberry Pi と組み合わせて使用することが想定されているカメラであり、採用事例や情報が豊富である。そのため、技術的不安がないという利点があるが、当カメラに比べて画質が劣り、カメラとしての機能が不足しているという欠点がある。当サービスが、写真撮影というテーマを扱っている以上、その

部分で妥協すべきではないと判断し、当カメラを採用した。

カメラの制御のために、Python の Requests ライブラリを用いた。Requests は HTTP ライブラリである。その理由は、習得が容易であると考えたからである。前述のカメラは、起動すると Wi-Fi アクセスポイントとなり、HTTP サーバとして動作する。その API を利用するためには、Python から HTTP リクエストを発行する必要がある。しかし Python 標準の urllib2 を用いると、ソースコードが冗長になってしまい、開発が遅れてしまう懸念があった。そのため urllib2 を抽象化して、単純に記述できる当ライブラリを用いた。事前に調査を行ったが、競合するものは確認されなかった。

カメラのライブビューを解析するために Node.js を用いた。その理由は、前述したカメラを Node.js から扱うための Nodejs ライブラリの存在である。前述したカメラは、ライブビューを RTP/Dynamic を用いて配信する。RTP はネットワークにおいて音声や動画をリアルタイムに伝送するためのプロトコルである。RTP は主要なデータ形式をサポートしており、例えば Motion JPEG を送受信するための RTP/JPEG がある。通常、Motion JPEG を伝送する場合はこちらのプロトコルを利用するのが一般的である。しかし当カメラは、Motion JPEG を伝送するにもかかわらず、RTP/Dynamic を用いる。フォーカスの状況や、顔認識の結果等の情報も同時に伝送するためである。RTP/Dynamic は、事前に伝送するデータ形式を規定せず、配信側がデータ形式を自由に決定できるプロトコルである。そのため、ライブビュー画像を配信するためには、当カメラ独自のプロトコルにもとづいて、パケットを解析しなくてはならなかった。これを Node.js 上で実装したライブラリが opc-js である。パケット解析の際は、このライブラリのソースコード参考にしながら、独自に機能を実装した。

カメラのビュー画像をストリーミング配信するためのマルチメディアフレームワークに、GStreamer を用いた。その理由は、多くの採用事例があることである。前述した方法でカメラのライブビューを、RTSP などで配信する場合は、画像のエンコードや RTSP サーバの構築が必要になる。これらを比較的容易に実現するソフトウェアが GStreamer である。パイプライン・アーキテクチャを採用し、簡単な動作であればコマンドだけで実現できる。実際コマンドのみで、前述した方法で解析した JPEG 画像を RTSP で Youtube Live に配信することに成功している。今回ライブビュー画像をストリーミング配信することは、開発期間の関係で実現できず、その一部を実装するに留まった。しかしこの技術は、将来的に必要となる可能性が高い。現状の JPEG 画像を HTTP を用いて配信する方法では、通信に無駄が生じ、なめらかな表示が困難である。ライブビューのストリーミング配信は、このような問題を解決することができる。

( 文責: 栗石卓耶 )

## 5.7 ドキュメンテーション

当サービスの利用方法や修正方法などを誰でも行えるよう、サブライズスナップについて説明する Web サイトの作成とソースコードのリポジトリヘドキュメントの追加を行った。

( 文責: 古川晃也 )

### 5.7.1 Web サイト

Web サイトは当サービスの説明と、サービスの構築に必要なリポジトリへのリンクを公開した。Web サイトは、当サービスの概要を利用者が理解できるようにし、サービスを利用・拡張したい人がそれを行えるようにすることを目的としている。Web サイトの構築には Jekyll を利用し、公開には GitHub Pages を利用した。Jekyll はブログを意識したシンプルな静的サイトジェネレーターである。Jekyll は様々なフォーマットのテンプレートディレクトリを、ローカルのコンピュータ上で実行するだけで Web サイトに必要なファイルを一気に生成できる。GitHub Pages は、無料で静的な Web ページの公開を行えるサービスである。Jekyll を利用してマークダウンファイルから Web ページを生成して公開する機能があり、今回はそれを利用した。そうすることで、少ない手数で見やすい Web ページを公開できた。

( 文責: 古川晃也 )

### 5.7.2 リポジトリのドキュメント

リポジトリのドキュメントには、それぞれのリポジトリのビルドやデプロイ方法を記述した。サーバのエンドポイントやその仕様も記述し、ドキュメントを読めば誰でもそのリポジトリの利用方法がわかるようにした。

( 文責: 古川晃也 )

## 5.8 デモ発表会

11月10日金曜日にプロジェクト担当教員やTAに対して、アプリケーションとポスターを使用したデモ発表会を行った。デモ発表会は最終発表報告会を想定した発表を行い、アプリケーションの挙動やポスターについての指摘を受けた。「デモの完成度は高かったか」、「ポスターはわかりやすかったか」を1「低かった」から5「高かった」までの数字で15人に評価してもらい、それとともにコメントを受けた。評価結果では、「デモの完成度は高かったか」に対して平均評価が4.26と高く「ストリーミングが素晴らしい」や「カメラの実装が出来ているのはすごい」など、アプリケーションの機能に関しては評価が高かったが、「写真をとった瞬間がわかりにくかった」などの改善点があることもコメントにより示された。また、「もう少しストーリー性があると思った」や「もっと便利だということが伝わるデモにすればよかった」などデモの仕方に改善点を見つけた。「ポスターはわかりやすかったか」に対しては平均評価が3.6であり、「画面の説明をもう少し詳しく入れてもいいと思った」「アプリケーションの使い方は文字が小さくてわかりにくかった」、「いらぬ説明があった」などのコメントがあり、ポスターの改善点が多く見つかった。

( 文責: 尾崎政亘 )

## 5.9 実装の改善

デモ発表会で受けた「写真をとった瞬間がわかりにくかった」という指摘をもとに、アプリケーションのプレビュー画面の修正を行った。まず、プレビュー映像の表示領域を広くした。そうすることによってプレビュー画面自体の見やすさを向上させた。また、カウントダウンの数字を見やすくした。具体的には、これまでのプレビュー画面に表示される数字が黒く、映像に混ざってしまいやすかったところを、数字を白色で表示するように修正した。これら2点の修正を行うことで、カウントダウンの数値が見やすくなり、写真がいつ撮られるのかがわかりやすくなった。

( 文責: 荒駿一 )

## 5.10 金沢工業大学教授に対しての発表

来訪された教授達を相手にポスターによるプロダクトの説明を3分間で行った。サプライズスナップのサービスの背景とそのサービス内容を説明した。「函館のみに限らず、他の観光地にも展開できるだろう」といった意見を得られた。

( 文責: 古川晃也 )

## 5.11 成果発表会

### 5.11.1 発表内容

本プロジェクトではポスターと実際に作成したアプリケーションを用いて発表を行った。内容は以下の7項目であり、以下の構成で発表した。

#### 1. 課題

自撮りでは構図に限界が出てしまい理想の構図では写真が撮れない、また観光旅行では記念撮影を複数行うことでマンネリ化してしまうという説明

#### 2. 提案

最良の位置から突発的写真撮影を行うことによって観光客に美しい写真と驚きの体験を提供するという説明

#### 3. サービスの概要

「設置してあるカメラの撮影範囲に入るとアプリケーションがビーコンの電波を検知する」、「複数人で最高の構図で写真を撮影する」、「その場でコミュニケーションをしながら写真を共有する」というサプライズスナップのサービスの流れを説明

#### 4. 画面遷移

アプリケーションを起動したときに出るデフォルト画面。撮影を開始したときに表示される撮影画面。デフォルト画面に表示された画像を押下したときに出る写真詳細画面。QRコードボタンを押下したときに表示されるQRコード画面についての説明

#### 5. システム

サービスを構成するサプライズスナップのアプリケーション、撮影するためのカメラとそれを制御する Raspberry Pi2、写真の保持や配信を行うクラウド上のサーバの3つのサブシス

#### テムについての説明

##### 6. 学び

技術的課題が発覚したことで、工期に大幅な遅れを生じたことから事前調査を行うことの大切さを学んだことの説明。またグループ内の情報が整理されていないため作業の効率が悪くなってしまったことから、情報管理の大切さを理解したと説明

##### 7. 展望

現状では撮影画面に表示されるプレビューがカクついてしまっているために、より滑らかに表示できるようにする。また現在は日本語のみとなっているため多言語対応を行うようにすると説明

また最終発表では実際にアプリケーションを動かしながら上記の説明を行った。デモ発表会での反省を活かしてカメラの前の壁に 148x98cm の八幡坂の写真を貼り、八幡坂で利用しているような空間を演出することによって有用性をアピールした。

( 文責: 尾崎政巨 )

### 5.11.2 レビュー内容

以下に最終発表で行った「発表技術」と「発表内容」に対して受けたコメントを抜粋して紹介する。

- シャッターチャンスを見逃さず写真が撮れるところは面白いと思いました
- “ 驚きの体験 ” という理由が薄い、カメラの設置が難しいのでは
- 声が小さい、表示が遅いのでこれから期待したい
- どのようにして周りを巻き込んだりしているのかわからない
- デモがわかりやすくよかった
- 旅行などの際に非常にいいと思う

( 文責: 尾崎政巨 )

### 5.11.3 反省点

前節で示したコメントから、発表する方法は改善されたが内容についてまだ改善があると考えられる。中間発表を行った際に受けたコメントでは「ポスターだと見づらかったので、スライドを作成したほうがよかった」というコメントを受けた。成果発表会ではポスターだけではなくスライドを用いたことによって、前述のようなコメントは受けなかった。「発表内容」に関して、「驚きの体験」という理由が薄い、カメラの設置が難しいのでは」や「どのようにして周りを巻き込んだりしているのかわからない」などのコメントを受けた。アプリケーションを動かしながら発表を行ったが実機 1 台のみで行ったため驚きの体験やどのように周りを巻き込み撮影を行うのかがわかりづらくなってしまった。

( 文責: 尾崎政巨 )

## 第 6 章 各メンバーの役割と活動の振り返り

### 6.1 役割分担

本グループは、API サーバとカメラサーバの開発を行う雫石と、アプリケーションの開発を行う古川、荒、尾崎三人のチームに別れて開発を行った。API サーバとカメラサーバについては、雫石が設計から実装まで全て行った。アプリケーション開発では古川が設計とモックの作成を行い、その後の必要な機能追加のタスク作成、その割り振りと進捗管理を行った。荒と尾崎はモックを作成している間に開発に必要な知識の学習に専念し、モックが完成してからは開発タスクの消化に専念した。

- 古川
  - グループのタスクの管理
  - 進捗管理
  - アプリケーションの詳細設計
  - アプリケーションのモック作成
  - アプリケーションのタスク作成
- 雫石
  - ポスターの作成
  - アプリケーションの画面デザインの作成
  - API サーバの設計
  - API サーバの開発
  - カメラサーバの利用技術の調査
  - カメラサーバの設計
  - カメラサーバの開発
- 荒
  - アプリケーションのデザインの修正
  - アプリケーションの機能実装
- 尾崎
  - アプリケーションのタスクの作成
  - アプリケーションの利用ライブラリの調査
  - アプリケーションの利用ライブラリの選定
  - アプリケーションの機能実装

( 文責: 古川晃也 )

### 6.2 古川晃也の振り返り

Android 開発の経験がほとんどない中で、機能の追加方法、利用するライブラリやモジュールの選定に時間がかかった。手探りでドキュメントや入門書を読み選定した。しかし、利用するライブ

ラリを途中で変えたりすることや、一度実装した機能がなくなってしまうたりした。そこで、各ソフトウェアに詳しい人に相談したところ、便利な機能や適切なライブラリなどをすぐに知ることができ、失敗も少なく開発をスムーズに進めることができた。開発を行うときは、開発に必要な技術についてよく知っている人に相談できる状況を作ることがとても大切だと学んだ。

( 文責: 古川晃也 )

### 6.3 雫石卓耶の振り返り

プロジェクト学習を通して様々な活動があり、様々な技術や教訓を学んだ。

API サーバの開発、カメラサーバの開発を行った。API サーバの開発では、Python の Flask という Web アプリケーションフレームワークを用いた開発を行った。初めに、API サーバの設計を行った。必要なエンドポイントや、DB のデータモデルについて決定した。Python を用いた開発経験が無かったので、公式ドキュメントや、Web 上にある各種チュートリアルを用いて言語の習得を行った。その後、Flask に関するチュートリアルを行い、簡単なテストアプリケーションを作成した。これをインクリメンタルに拡張していくことで、実際の API サーバを構築していった。夏休み期間を利用して集中的に開発することで、3 日間で API サーバを実装し終わることができた。利用した技術はどれも開発経験の無いものだったが、順を追って学習や実装していくことで着実に開発を進めることができた。この活動を通して、Python を用いた API サーバの開発に関する技術を学んだ。

カメラサーバの開発では、主に Node.js や GStreamer を用いた開発を行った。カメラの制御や撮影した画像の配信は、API サーバと同じ技術で解決できた。この開発で一番苦労したのは、カメラのライブビュー画像の配信である。カメラの配信するストリーミングプロトコルを解析し、API サーバを通じてアプリケーションに表示しなければならない。これには GStreamer を用いる予定であったが、カメラの生成するストリーミング形式が特殊であったため、利用できなかった。このことが判明するまでに、多くの期間がかかってしまった。この原因は、ストリーミングに関する知識不足と、カメラの仕様書を熟読しなかったことであった。自分の都合の良いように、カメラの動作を勘違いしていたため、問題の解決に時間がかかってしまった。この活動を通して、技術に関する知識が不十分な間は、実装を始めるべきではないという教訓を得た。

( 文責: 雫石卓耶 )

### 6.4 荒駿一の振り返り

プロジェクト学習を通して様々な活動をした。

前期は主にサービスの提案と開発の準備であった。サービスの提案をするにあたりフィールドワークを行い、まちにある問題点を発見することに努め、その問題を解決する方法を考えた。また、サービスをビーコン利用により実現するため、ビーコンに関する知識の取得をした。開発の準備では、チーム開発手法の学習、アンドロイドアプリケーション開発環境の構築、アンドロイドアプリケーションの開発、Java 言語の学習をした。実際にアンドロイドアプリケーションを製作したときに、サンプルアプリケーションに自分で考えた追加機能を実装する試みをした。その際、Web サイトを参照することが多くなり様々な知識を得ることができた。

後期は主に開発をした。本格的なチーム開発を体験したのは今回のプロジェクト学習が初めての経験であったため、GitHub、SourceTree の使用に苦戦した。自分の手違いから開発の出戻りを発生させてしまったことがあったが、そのような失敗から徐々にチーム作業に慣れていくことができた。開発に慣れてきてからは、各機能を実装するタスクを引き受けた。主に UI 面タスクが多く、サーバのデータを取得・加工しアプリケーションに表示させたり、ダイアログ画面を作成しそこから遷移される画面との連携などの機能を実装した。経験や知識の不足から簡単に実装することはできなかったが、書籍や公式ドキュメントを参照しながら課題を解決していくことができた。また、開発中に他のメンバーのタスクを手伝う機会が多くあり、非同期処理の仕組みや、QR コード生成の仕組みなどを学習し、実装が達成できるように議論をすることもできた。

( 文責: 荒駿一 )

## 6.5 尾崎政巨の振り返り

1 年間のプロジェクトを通して、様々なことを学ぶことができた。前期は主に、フィールドワークを行うことで函館の問題点を見つけ出し、それを解決するための方法を考えた。また、ビーコンについての情報を知らなかったため、3 つのグループに分かれて iBeacon、LINE Beacon、Eddystone についてそれぞれ調べた。調べたことでそれぞれのビーコンの特性を学ぶことができた。また後期から始まるチーム開発のために、チーム開発手法、Android アプリケーション開発のための環境構築、Java の学習を行った。そこから、チーム開発を行うために必要な知識を学ぶことができた。後期は主に、開発を行った。開発を効率よく行うために GitHub や SourceTree、Instagantt、Asana などを使用した。これらのツールを今までに使用したことが無かったために苦労した。また、開発が進んでいくとあらゆるタスクが発生した。その中で私が担当したタスクが、画像の情報が入った QR コード生成の機能、非同期で画像をアプリケーションに取り込む、サーバにある情報の獲得、サーバにある画像をアプリケーションで表示、アプリケーションの UI 設計などであった。参考書やウェブサイトなどで学習しながら開発することで、これらを実装できるようになった。

( 文責: 尾崎政巨 )



## 第7章 まとめと展望

### 7.1 前期の活動

前期では、大きく分けて5つのことを行った。まず第1に、ビーコンの仕様や特徴について各自調べたうえでプレゼンテーションを行うことによって、プロジェクトメンバー全体の共有を行った。第2に、フィールドワークを通して、函館の特徴や課題を洗い出した。第3に、ビーコンを利用したサービスのアイデア出しとその選定を行い、実際に開発を行う3つのサービスを決定し、グループ分けを行った。第4に、サービスのコンセプトやユースケースを考え、画面設計やシステム設計を行った。最後に、これらの成果を中間発表会で発表し、提案サービスについてのフィードバックを受けた。

( 文責: 荒駿一 )

### 7.2 後期の活動

後期では、「サプライズスナップ」の実装を行い、その利用方法などのドキュメントを書いた。その後、デモ発表会にて実際に利用する状況を再現して発表し、教員やTA、他グループのメンバーからのフィードバックを受けた。そのフィードバックをもとにサービスを修正し、サービスの質を向上させた。

( 文責: 荒駿一 )

### 7.3 活動の振り返り

本プロジェクトの目標と、本グループの目標について、それぞれ達成できたかどうかを振り返る。

#### 7.3.1 本プロジェクトの目標

本プロジェクトの目標は、ビーコンを活用して函館の街を盛り上げられるようなサービスのプロトタイプを作成することであった。この目標は、実際に稼働するアプリケーションを作成することができたので、達成することができた。

#### 7.3.2 本グループの目標

本グループの前期の目標

本グループの前期の目標は、作成・公開するサービスを考案しコンセプトを考え、後期の開発に向けて設計を行うことであった。前期においてはビーコンについての技術調査とサービスのアイデアを練った。そして、サービスのユースケース図やシステム設計図、画面遷移図を作成し、利用ワークフロー図を作成した。これらのドキュメントは後期でも参照し、非常に役に立った。以上より、前期の目標は達成した。

## 本グループの後期の目標

後期の目標は下記の5つであった。それぞれについて振り返りを行う。

1. サプライズスナップの設計書にある機能をすべて実装し、動作可能な状態とする。  
前期に作成した各種設計内容を全て実装することができたので、この目標は達成することができた。
2. サプライズスナップを学内や函館の観光地に展開し、ユーザーから得たフィードバックを参考にサービスの質の向上を図る。  
観光地に展開することはできなかったが、学内のデモ発表会にてサービスの動作を教員やプロジェクトメンバーに見てもらい、そのフィードバックを受けてサービスの修正を行うことができた。そのため、この目標も達成できた。
3. サプライズスナップについての情報をまとめた Web サイトを公開し、誰でもサービスを再現できるようにする。  
サプライズスナップについての Web サイトを作成し、サービスが再現できるようにした。そのため、この目標も達成できた。
4. サプライズスナップの公開可能なソースコードをすべて Web 上に公開する。  
サプライズスナップのアプリケーション・API サーバ・カメラサーバすべてのソースコードを GitHub 上に公開した。そのため、この目標も達成できた。
5. 公開したりポジトリすべてにドキュメントを書き、誰でも修正が行えるようにする。  
すべてのリポジトリにドキュメントを書いた。そのためこの目標も達成できた。

( 文責: 荒駿一 )

## 7.4 サービスの展望

サービスの展望として、観光地への展開を考えた修正を行うことが考えられる。まず、不安定なネットワークでも安定に稼働するようなシステムにすることがあげられる。現時点では安定したネットワークがあることを前提にシステムが作られていて、ネットワークが切断された際に自動的に再接続する仕組みなどが整っていない。また、雨天などでの利用も考えていないため、それらにも対応することも検討したい。

( 文責: 荒駿一 )

# 付録 A 中間発表会で使用した本グループのポスター

2017.7.14 中間発表会



Project No.8 ビーコンを用いて函館のまちをより良くするサービスの提案・開発

## サプライズスナップ

Surprise Snap

新たな体験を生み出す突発的な写真撮影

Member <sup>GL</sup> 古川晃也 零石卓耶 荒駿一 尾崎政臣  
Alaya Hironaka Takashi Shiroishi Shunichi Arai Masamichi Ozaki

### 課題 Background

- ・自撮りの構図には限界があり、理想の構図では写真が撮れない
- ・記念撮影は観光旅行では連続するのでマンネリ化する
- ・There is a limit to the composition of self-shooting, and it is impossible to take pictures with the ideal composition
- ・Taking commemorative photo continues in sightseeing, so it gets stuck in a rut

### 提案 Suggestion

最良の位置からの突発的な写真撮影を行うことで、観光客に美しい写真と驚きの体験を提供する

Provide tourists with beautiful pictures and surprise experience by performing unexpected photographs from the best position

---

### アプリの概要 App



設置してあるカメラの撮影範囲に入るとアプリがビーコンの電波を検知する  
When a device enter the shooting range of the installed camera, this App detect the radio wave from beacon



複数人で最高の構図で写真を撮影する  
Taking pictures with the best composition by multiple people



その場でコミュニケーションをしながら写真を共有する  
Communication is born while sharing photos

### 画面遷移 Screen Transition



撮影スポットに近づくと、撮影画面が表示されます。撮影が終了すると、写真詳細画面が表示されます。QRコード画面は、写真の共有に使用されます。

### システム構成 System Structure



ビーコンの電波が、カメラの検知範囲に入ると、カメラが撮影を行います。撮影された写真は、Raspberry Pi 3を経由してクラウドサーバーにアップロードされます。クラウドサーバーは、Flask、python、MySQLを使用しています。クラウドサーバーは、アプリと通信し、写真の共有を行います。

---

### 今後の予定 Future Plans

10月

設計を基に  
第一バージョンの作成を行う

11月

学内などに設置し  
フィードバックを得る

与えられたフィードバックを  
基に繰り返し改善を行う

図 A.1 サプライズスナップポスター（中間発表会）

# 付録 B 成果報告会で使用した本グループのポスター

2017.12.8 最終成果報告発表会



Project No.8 ビーコン IoT でまちより良くするサービスの提案・開発

**サプライズスナップ** Surprise Snap

新たな体験を生み出す突発的な写真撮影  
Sudden photography to produce new experiences

Member 古川晃也 Akira Furukawa 幸石卓耶 Takuya Shoukuzumi 荒駿一 Shunichi Arai 尾崎政巨 Masamichi Ozaki

### 課題

- ・自撮りの構図には限界があり、理想の構図では写真が撮れない
- ・記念撮影は観光旅行では連続するのでマンネリ化する

### 提案

最良の位置からの突発的な写真撮影を行うことで、観光客に美しい写真と驚きの体験を提供する

### サービスの概要



設置してあるカメラの撮影範囲に入るとアプリがビーコンの電波を検知する

複数人で最高の構図で写真を撮影する

その場でコミュニケーションをしながら写真を共有する

### システム構成



写真の保持・配信

ビーコン電波の範囲に入ったことを通知

写真をダウンロード

写真をアップロード

撮影を指示

### 画面遷移



画像を選択

履歴画面 (デフォルト画面)

撮影スポットに入ったとき

撮影画面

写真詳細画面

QRコード画面

### 学び

- ・事前調査を行うことの大切さを学んだ  
→ 技術的な課題が後から発覚し、工期が大幅に遅れてしまった
- ・情報管理の大切さを理解した  
→ グループ内の情報が整理されておらず、効率が悪かった

### 展望

- ・よりなめらかにプレビューを表示する  
→ 現状では、表示が時折カクついてしまう
- ・多言語対応を行う  
→ 海外観光客でも利用可能にする

図 B.1 サプライズスナップポスター（成果報告会）

## 参考文献

- [1] 総務省, 情報通信白書, *ICT* 白書, IoT・ビッグデータ・AI: ネットワークとデータが創造する新たな価値, 平成 28 年版. 日経印刷, 全国官報販売協同組合 (発売), 2016, ISBN: 978-4-86579-060-3.
- [2] A. Inc. (2017). Asana apps and integrations, [Online]. Available: <https://asana.com/apps>.
- [3] *TECHNICAL MASTER* はじめての *Android* アプリ開発, 第 2 版. 秀和システム, 2016, ISBN: 978-4798048536.