

筑波大学大学院博士課程

システム情報工学研究科特定課題研究報告書

M2Mを用いた大規模データ収集システムの
構築に関する研究開発
ーセンサデータ収集機能を持つスマートフォン向け
クライアントアプリケーションの設計と開発ー

枝松 ちさと

修士（工学）

（コンピュータサイエンス専攻）

指導教員 高橋 伸

2014年3月

概要

本プロジェクトは、筑波大学大学院「高度 IT 人材育成のための実践的ソフトウェア開発専修プログラム」における特定課題研究として、「M2M を用いた大規模データ収集システムの構築に関する研究開発」というテーマのもと、著者を含む学生 4 名のチームと株式会社富士通研究所の協同で遂行したものである。近年、新たなビジネスの開拓や既存の問題の解決のために、大規模データの利用に注目が集まっている。同時に、それらのデータを自動的に収集・処理を行うためのアーキテクチャとして、人間の仲介なく機械が自律的に情報交換を行う M2M (Machine-to-Machine) の仕組みを適用する動きが高まっている。しかしながら、現状のシステムでは人間が仲介していないとはいいがたく、また大規模なデータを効率的に収集するためのシステム最適化についてはまだ議論の余地があると考えられる。そこで本特定課題研究では、スマートフォンとサーバを用いた“人間の仲介のないシステム”を開発し、この開発を通して M2M システムや大規模データの収集における課題を抽出した。そして、それらの課題について実験を行い、システムの最適化について議論・検討を行った。具体的には、「道路の路面異常の検出」という問題に対して M2M ネットワークシステムを適用して設計・開発を進めた。本報告書では、本プロジェクトの全体像について述べるとともに、著者が担当したスマートフォン向けクライアントアプリケーションと Web アプリケーションの開発、開発を通して抽出した課題とその考察、およびプロジェクトの実績について報告する。

目次

第1章	はじめに	1
1.1	プロジェクトの背景	1
1.2	プロジェクトの目的とアプローチ	2
1.3	プロジェクト体制	2
1.4	本報告書の構成	3
第2章	本プロジェクトで取り扱う例題について	4
2.1	例題の設定プロセス	4
2.1.1	ブレインストーミングによる例題候補の列举	4
2.1.2	例題の設定基準	6
2.1.3	設定した例題	7
2.2	道路の維持管理に関する問題	7
2.2.1	現状	7
2.2.2	問題点	8
2.2.3	解決策	8
2.3	関連研究	8
2.3.1	BumpRecorder	9
2.3.2	FUJITSU Intelligent Society Solution 道路パトロール支援サービス	9
2.3.3	開発システムの位置付け	9
2.4	議論	10
第3章	開発システム	11
3.1	システム概要	11
3.2	提供する機能	12
3.3	システム構成	12
3.3.1	ハードウェア構成	13
3.3.2	ソフトウェア構成	15
3.4	開発の担当	17
3.4.1	メンバの開発担当	17
3.4.2	著者の担当範囲	17
第4章	クライアントアプリケーションの開発	19

4.1	スマートフォン向けクライアントアプリケーション	19
4.1.1	概要	19
4.1.2	実装環境	19
4.1.3	実装した機能	20
4.2	地図閲覧のための Web ブラウザアプリケーション	23
4.2.1	概要	23
4.2.2	実装環境	23
4.2.3	実装した機能	23
第 5 章	開発アプリケーションの評価実験	27
5.1	スマートフォン向けアプリケーションにおける検討すべき課題	27
5.1.1	低負荷処理	27
5.1.2	省パケット通信	27
5.2	低負荷処理に関する実験	28
5.2.1	仮説	28
5.2.2	検証	28
5.2.3	結果	31
5.3	省パケット処理に関する実験	33
5.3.1	期待される結果	33
5.3.2	実験による評価	33
5.3.3	結果	38
5.4	考察	39
第 6 章	プロジェクトの振り返り	41
6.1	プロジェクトの基本方針	41
6.1.1	開発の方針	41
6.1.2	コミュニケーション計画	41
6.1.3	役割分担	42
6.2	スケジュールの計画と実績	42
6.3	考察	45
6.3.1	スケジュールの遅延	45
6.3.2	運用の一部形骸化	46
6.4	リーダーの立場からの振り返り	46
第 7 章	まとめ	47
	謝辞	49
	参考文献	50

図目次

2.1	作成した 4 象限マトリクスの一部	5
2.2	路面損傷している道路	7
3.1	システムの全体イメージ	11
3.2	ハードウェア構成図	13
3.3	ソフトウェア構成図	16
4.1	Android アプリケーションの処理フロー	21
4.2	Web アプリケーションの処理フロー	24
4.3	出力される XML データの例	24
4.4	取り出し範囲の半径	25
4.5	つくば市周辺の路面情報マップ	26
5.1	要素ごとの消費電力の割合	31
5.2	端末の状態別の消費電力	32
5.3	サンプリング周波数別の消費電力	33
5.4	バイナリファイルへの書き込み	37
5.5	CSV ファイルのサイズを 1 としたときの各ファイルのサイズ比	39
6.1	本プロジェクトのスケジュール計画と実績	43
6.2	スコープ 1 の開発範囲	44
6.3	スコープ 2 の開発範囲	44

表 目 次

2.1	市場でリリースされている IT サービス (抜粋)	4
2.2	例題として本プロジェクトが提案したサービス (抜粋)	6
3.1	クライアントサイドのハードウェア構成	13
3.2	サーバサイドのハードウェア構成	14
3.3	クライアント PC のソフトウェア構成	17
3.4	スマートフォンのソフトウェア構成	17
3.5	サーバサイドのソフトウェア構成	17
3.6	開発担当の一覧	17
4.1	センサから計測する情報	21
4.2	1つのレコードに記録される項目	22
4.3	XML データの属性	25
5.1	各実験条件のパラメータ設定の一覧	30
5.2	実験端末の一覧	31
5.3	消費電力量の平均値	32
5.4	サンプリング周波数ごとの走行情報ファイルのサイズ	32
5.5	各種センサ情報の上限, 下限, およびビット数	35
5.6	実験端末の一覧	38
5.7	各ファイル形式の 1 時間あたりのサイズ [MB]	38
6.1	チームメンバの開発担当	42

第1章 はじめに

本プロジェクトは、筑波大学大学院「高度 IT 人材育成のための実践的ソフトウェア開発専修プログラム（専修プログラム）」における特定課題研究として、「M2M を用いた大規模データ収集システムの構築に関する研究開発」というテーマのもと、システムの開発を行ったものである。プロジェクトは、著者を含む専修プログラム所属の学生 4 人のチームと株式会社富士通研究所の協同で進行した。本報告書はこのプロジェクトの実施内容、および著者のプロジェクトへの貢献についてまとめたものである。

1.1 プロジェクトの背景

近年、新たなビジネスの開拓や現状業務への付加価値向上、あるいは既存の問題の解決などのために、大規模データを重要な資産として取り扱う動きが高まっている [1][2][3]。例えば、Amazon¹ や楽天² などといった EC ショップでは、利用者の購買履歴や Web サイト内での顧客の動きなどのデータを集約し、利用者の動向などを分析している。そしてその分析結果を使って利用者個々にあったおすすめ商品を提示するといった、利用者の購買意欲を高める情報提供を行っている。また大阪ガスでは、400 万件以上のガス器具の修理履歴を分析しデータベース化している。修理依頼を受けた際、このデータベースを利用して故障が疑われる部品などを絞り込み、修理作業員の業務効率化を図っている。

同時に、それらのデータを大量に収集・処理するために機械同士が通信ネットワークを介し、人間の仲介なく自律的に情報交換を行う M2M (Machine-to-Machine) の仕組み [4] を利用したシステムにも注目が集まっている [5][6][7]。JR 東日本ウォータービジネスが開発した次世代自動販売機 [8] は、搭載したカメラで利用者の年代や性別などといった顧客の属性を判定し、その属性に合ったおすすめ商品をディスプレイ上に表示するというものである。さらに、センサで取得した顧客の属性や購入時の情報 (POS 情報) は自動的にサーバに送信・集約されており、これらの情報は売上拡大のために分析されている。このとき、顧客の属性判断や POS 情報の集約などは自動で行われており、この機械は人間を仲介していない M2M システムであると言える。

しかしながら、現在利用されている“M2M システム”と呼ばれるものの多くは人間が仲介しており、M2M システムであるとは言いがたい。また、大規模データの処理には非常に負荷がかかるため、システム構築の際には、データの取り扱い方やリソースの配分などといった

¹Amazon (URL : <http://www.amazon.co.jp/>)

²楽天市場 (URL : <http://www.rakuten.co.jp/>)

最適化について十分に検討する必要がある。さらに，サーバ・クライアントといった構成要素の間には負荷のトレードオフがあり，システムが効率的に稼働するためには，システム全体で負荷について考慮しなければならない。それにも関わらず，サーバ単体・クライアント単体といった構成要素ごとの局所的な最適化についてのみ議論されることが多く [9][10]，システム全体としての負荷バランスを考慮した最適化についての議論は少ない。

1.2 プロジェクトの目的とアプローチ

本プロジェクトでは，人間が仲介することのない M2M ネットワークを利用した大規模データの収集システムを開発すること，および開発を通して課題を抽出し，この課題について考察・検討することを目的とする。

この目的を達成するためのアプローチとして，本プロジェクトでは世の中で問題となっている事象を例題として取り上げる。そして，その例題を解決するためにクライアント・サーバモデルによる M2M システムの開発を行い，大規模にデータを収集・処理する仕組みを構築する。またこの開発を通じて，大規模データ収集システムの構築における課題を抽出する。開発システムに対し構成要素ごとに負荷テストなどの実験を実施し，得られた結果から抽出した課題への対策を考察するとともに，システムの全体最適について議論する。

1.3 プロジェクト体制

本プロジェクトの実施体制は次の通りである。

顧客

山下 浩一郎 氏³

課題担当教員

天笠 俊之 准教授⁴

学生メンバ

枝松 ちさと⁵

林 貴哉⁵

村林 竜司⁵

³株式会社 富士通研究所

⁴筑波大学 システム情報系 情報工学域

⁵筑波大学大学院 システム情報工学研究科 コンピュータサイエンス専攻 高度 IT 人材育成のための実践的開発専修プログラム

胥 徳文⁵ (12 月まで従事)

1.4 本報告書の構成

本報告書は本章を含め 7 章で構成されている。第 2 章では、本プロジェクトで取り扱う例題をどのように設定したのか述べたうえで、取り扱う例題が抱えている問題とその解決策について論じる。第 3 章では、本プロジェクトで実際に開発したシステムについて説明する。第 4 章では、著者が担当したクライアントアプリケーションの開発について述べる。第 5 章では、クライアントアプリケーションの開発を通して抽出した課題について述べたうえで、これらの課題について検討するために実施した評価実験とその結果、および考察について論じる。第 6 章では、プロジェクトの実施状況について説明し、プロジェクトリーダーとしての観点からプロジェクトを振り返ったうえで考察する。最後に、第 7 章で本プロジェクトについての成果についてまとめ、今後の課題について述べる。

第2章 本プロジェクトで取り扱う例題について

本章では、本プロジェクトで取り扱う例題を設定するまでのプロセスを述べ、設定した例題に関する現状および抱えている問題について論じる。

2.1 例題の設定プロセス

2.1.1 ブレインストーミングによる例題候補の列挙

例題を設定するにあたり、本プロジェクトではまず世の中における問題の抽出を行った。可能な限りの問題を列挙するために、ブレインストーミングによる洗い出しを行い、抽出した問題の分析・整理を行った。以下で、本プロジェクトでの例題列挙のプロセスについて説明する。なお、この例題設定のプロセスで洗い出した詳細なアウトプットについては、付録に掲載する。

既存サービスの調査

まず事前調査として、市場にリリースされている IT サービスや、現在リリースに向けて研究を行っているプロジェクトなどの洗い出しを自由に行い、それらが世の中のどのような問題を解決するために開発されたのかを列挙した。列挙した一部のサービスを表 2.1 に示す。

表 2.1: 市場でリリースされている IT サービス (抜粋)

列挙した IT サービス
・クラウド型電子カルテ
・IC タグを使ったサプライチェーン・マネジメント
・衛星データやセンサを用いた漁場予測システム
・スマートグリッド
⋮

そして、先の事前調査で洗い出したサービスがどのような分野で利用されているのかを分類したり、様々な視点から 2 つの軸に分けて 4 象限マトリクスに落とし込む作業を行った。この結果をもとに、市場としてまだ未開拓の領域があるかどうかを議論した。

このとき作成した4象限マトリクスの一部を、図2.1に示す。図左側の4象限マトリクスは、縦軸に自動で入力が行われるのか、手動で入力が行われるのかを、横軸にデータを蓄積して利用するのか、あるいはその場で入力されたデータだけで利用できるのかを取ってITサービスをマッピングしたものである。「クラウド型電子カルテ」を例にとると、このサービスは医療分野で活用されるものである。また、情報の入力はいずれも医師が行い、データは患者ごとに蓄積されていくものであるため、手動で入力が行われ、データは蓄積して利用されるという第3象限にマッピングされる。図右側のマトリクスは、縦軸にサービスで得た情報が外部に発信されるのか、社内など閉じられた環境でのみ利用されるのかを、横軸にそのサービスは法人向け（Business）であるのか、個人向け（Consumer）であるのかを取ってITサービスをマッピングしたものである。「ICタグを使ったサプライチェーン・マネジメント」システムを例にとると、このサービスは経営分野で活用される。また、この情報は基本的に社内で活用され、法人向けに開発されているものであるため、閉じられた環境で利用され、法人向けのサービスという第3象限にマッピングされる。

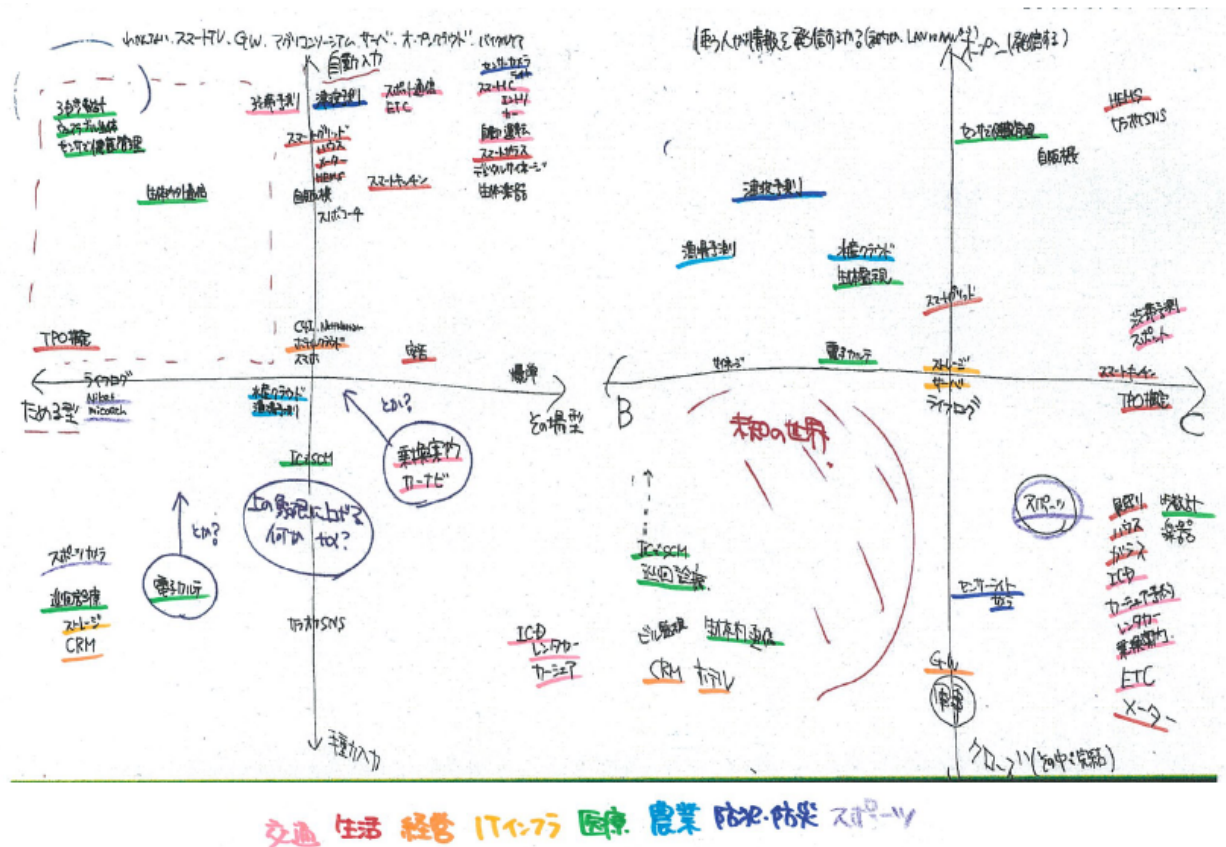


図 2.1: 作成した 4 象限マトリクスの一部

例題の列举

事前調査および作成した4象限マトリクスの結果をふまえながら、本プロジェクトでどのような例題を取り扱うかについて、世の中における問題や要望を考慮しながら議論を行った。列举した一部の例題案を、表 2.2 に示す。

表 2.2: 例題として本プロジェクトが提案したサービス（抜粋）

世の中の問題・要望	サービス	概要
家屋の劣化がすぐに把握できない	家屋の補修箇所検知システム	家屋の壁や天井にセンサを取り付け、故障しそうな箇所を検知・連絡するサービス
店舗の集客率を向上させたい	店舗配置支援システム	どこから来る客が多いのかなどの情報をスマートフォンを用いて集め、地図に可視化して店舗開設時に役立てるサービス
道路の維持管理に人手を要している	道路の維持管理支援システム	路面の異常をスマートフォンのセンサを使って把握し、道路の維持管理を支援するサービス
自分の健康状態をその場で知りたい	セルフ健康診断システム	日常の行動ログなどを用いて、健康診断を行うサービス

2.1.2 例題の設定基準

例題を絞り込むために、例題設定の指標として以下の5点を考慮することにした。

- 新規性
他の既存サービスに比べて、そのサービスに新規性があるか。
- 実現可能性
そのサービスを実現するために、技術的な面・期間的な面でどれだけ課題があるか。
- 有効性
サービスが、その問題の解決に効果的であるか。
- 有用性
サービスが実用に耐えうるか。
- 市場性
市場にその問題を解決したいというニーズがあるか。

2.1.3 設定した例題

2.1.2 項で挙げた基準をふまえ、本プロジェクトでは、例題として「道路の維持管理を支援するサービス」を取り扱うこととした。この例題における現状や問題については、次節で論じる。

2.2 道路の維持管理に関する問題

本節では、2.1.3 項で例題として決定した道路の維持管理に関して、現状とその問題点について論じる。そして、列挙した問題点に対して本プロジェクトでどのように解決するか述べる。

2.2.1 現状

昨今、図 2.2 に示すような路面損傷および落下物・路上障害物が原因となって起きる交通事故が後を絶たない。実際、路面異常や落下物などの道路瑕疵件数・報告件数は増加傾向にあるというデータもある [11]。この状況に対し、国などの道路管理者はパトロールカーで道路の巡回、点検を行うなどの措置を講じている。例えば、国道を管理している国土交通省は、路線の交通量に応じて巡回頻度を見直すという対応をとっている。



図 2.2: 路面損傷している道路

2.2.2 問題点

2.2.1 節で道路の維持管理における現状について述べたが、これには次のような問題が存在している。

1. 即時性が欠如している

巡回の頻度を見直すという対応を講じてはいるものの、維持管理の予算に制約があるために実際は巡回の頻度が制限されており、リアルタイムに路面の異常を発見することが難しくなっている。

2. 情報量が少なく、偏りがある

道路管理者のみで路面異常などを発見するには、情報の収集量に限界がある。また、路線によって巡回頻度が異なっているため、路面の異常や障害を発見する確率が路線によってばらついている [11][12]。

3. 作業負担が大きい

現在の道路管理における作業は、路面を目視で確認するといったように基本的には人手で行われている。このため、道路管理作業者の負担が大きくなるとともにコストも多大に掛かっている。

これらの問題の根本的な原因は、人的資源の制約などといった人手によるものが大きい。これらの問題に対処するためには、人手を介さない維持管理の仕組みを実現する必要がある。

2.2.3 解決策

2.2.2 項で挙げた問題に対し、我々は本プロジェクトのテーマである「M2M ネットワークを利用したデータ収集システム」を適用することで解決を図る。

具体的には、スマートフォンを利用して自動車の走行時の情報を計測し、これを自動的に集約したうえで路面の異常を検出するシステムを構築する。スマートフォンを用いる主な理由としては、スマートフォンが急速に普及しており、多くのユーザから情報を収集できること、またスマートフォンには多彩なセンサが搭載されており、走行時の情報を様々な視点から一度に計測できることが挙げられる。

本システムによって、2.2.2 項で挙げた問題点を解決することができる。

2.3 関連研究

本節では、本プロジェクトで取り扱う例題について、現在リリースされているサービスなどについて述べたうえで、本プロジェクトの位置付けについて論じる。

2.3.1 BumpRecorder

バンプレコーダー株式会社が開発・提供している段差計測アプリケーション“BumpRecorder”は、自動車の路面走行時の振動から段差を計測するものである [13]。このアプリケーションでは、スマートフォンのセンサを用いて路面走行時の振動を観測・収集し、文献 [14] で提案されている手法を用いて路面段差を検出する。そして、その分布状況は Google マップ¹ 上で確認することができる。

しかしながら、このアプリケーションで用いられている手法では、1 回の解析に使うデータが入力された 1 件の走行情報のみとなっている。すなわち、解析時に他の走行情報を参考していない。また、このアプリケーションでは 1 件の走行情報で高精度の解析結果を得るために、スマートフォン側で段差の有無の判定や高さの計算を行うなど、様々な処理が行われている。このため、スマートフォン側の処理負荷が非常に大きく、さらにバッテリー消耗が激しくなっているのが見受けられる。

2.3.2 FUJITSU Intelligent Society Solution 道路パトロール支援サービス

富士通株式会社からリリースされている“FUJITSU Intelligent Society Solution 道路パトロール支援サービス”[15] は、自治体などの道路管理者をターゲットにしたサービスである。

このサービスは、道路管理者が所有するパトロールカーなどの巡回車にスマートフォンを搭載し、現行の業務と同じように道路巡回を行い、スマートフォンに搭載されている各種センサを用いて巡回時の情報を計測・記録する。そして、記録したセンサの情報をもとに、舗装道路の凹凸情報を推定し、地図上に表示するというものである。このサービスには、道路管理者の現行の業務負担を増加させることなく道路の維持管理を支援できるという特徴がある。

しかしながら、先にも述べたように道路管理者向けのものであるため、収集できるデータ数や収集範囲が限られてしまう。また、精度の良い結果を得るため、巡回車ごとにスマートフォンのキャリブレーションを行う必要があり、設定に手間を要してしまう。

2.3.3 開発システムの位置付け

本プロジェクトで開発するシステムでは、走行情報を収集する機器には一般に普及しているスマートフォンを用い、ユーザに特別な準備を必要とさせない。加えて、走行情報の収集者は一般の道路利用者を想定している。このため、[15] のサービスと比べて広範囲に渡って道路の維持管理をサポートすることができ、収集できるデータ数も多くなることが期待される。

また、路面異常の検出手法には統計的な処理を採用する。すなわち、大量のデータで路面異常の検出精度を向上させる手法を取るため、他のユーザの走行情報も考慮することができる。粗悪なデータも許容することも可能となるため、キャリブレーション処理やスマートフォン側での解析処理が不要となり、設定の手間やスマートフォンの処理負荷を軽減できる。

¹Google マップ (URL: <https://maps.google.co.jp/>)

2.4 議論

本節では、今回取り扱う例題が本プロジェクトの目的に適しているかについて議論する。

まず、この例題が M2M の仕組みを適用するのに適していると判断した理由は、現在の道路維持管理の仕組みが基本的に人的資源を要しているものであるが、路面異常の検出といった初期対応についてはセンサを利用すれば実現できるものであり、自動的に管理することが可能な事案であると考えたからである。

そして、今回の例題が本プロジェクトの目的に沿っていると判断した理由は、提案システムが独立で動く仕組みを複数組み合わせることによって実現するものだからである。2.2.3 項でも述べたように、提案システムでは一般に普及しているスマートフォンを利用するというアプローチを取っており、容易に多くの機器をネットワークに参加させることができる。これによって、走行時の情報が広範囲に渡って大量に収集できる仕組みと、ネットワークを介してそれらの情報が集約される仕組みが構築される。また提案システムでは、その集約した情報から路面異常を検出するという、大量の情報を操作する仕組みも同時に確立される。今回取り扱う例題では、これらの仕組みを組み合わせることで現状の問題を解決するため、それぞれ単独で最適化を行うのではなく、全体を俯瞰したうえでバランスを議論する必要があると考えられる。

以上より、今回の例題は M2M システムを適用するに値し、なおかつ全体の最適化について議論・検討するのに適していると考えられる。

第3章 開発システム

本章では、プロジェクトで実際に開発したシステムの概要、およびシステムの構成について述べる。

3.1 システム概要

本プロジェクトで開発したシステムの全体イメージを図 3.1 に示す。

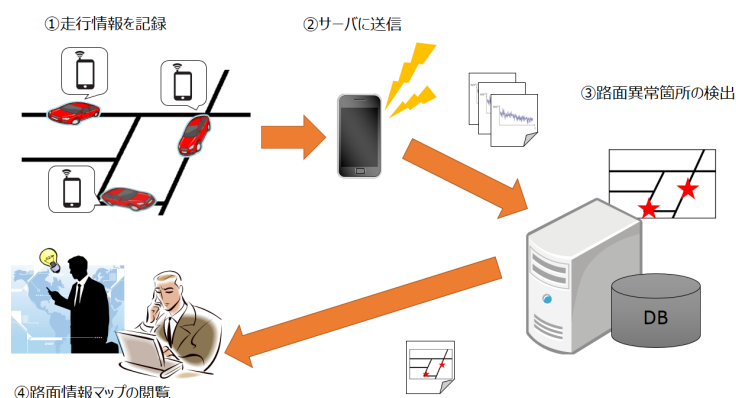


図 3.1: システムの全体イメージ

このシステムは、クライアント・サーバモデルで構成する。クライアント側では主にデータの収集を、サーバ側ではデータの加工処理を行う。

大まかな流れとしては、まずスマートフォンにインストールしたアプリケーションを利用し、端末に搭載されているセンサで一般の道路利用者が路面を走行した際の情報を計測する。そしてこれらの情報を“走行情報”としてデータを記録する。走行情報は一定のタイミングでサーバに送信される。サーバ側では、送られてきた走行情報を解析することで路面段差やひび割れ、凹凸などといった路面の異常（以下、路面異常と称する）を検出し、異常と判断された位置や方向などの情報を“路面異常箇所”としてサーバに保存する。

また、クライアントアプリケーションを介してユーザからサーバに路面異常箇所の閲覧要求があった場合、サーバはその要求に応じた路面異常箇所の情報を返す。クライアントアプリケーション側では、返された路面異常箇所を地図にプロットした上で“路面情報マップ”としてユーザに表示する。

このシステムでは、走行情報の計測に特別な専用機器ではなくスマートフォンを用いるため、クライアントサイドの導入は容易に行うことができる。さらに、スマートフォンは汎用機器として広く普及しているので、多くのユーザから情報を集約するのに適しており、即時性の問題と情報量の少なさの問題を解消できると考えられる。また、走行情報の収集から路面異常の検出まではシステムが自動的に行う。これにより、2.2.2 節で述べた、作業負担が大きいという問題を解決することができる。

3.2 提供する機能

開発システムは、大きく「走行情報収集機能」、「解析機能」および「路面情報マップ閲覧機能」の3機能を有する。本節では、各機能の概要を述べる。

走行情報収集機能

振動や位置など、走行時の情報をスマートフォンのセンサから読み取り、これを“走行情報”としてファイルに記録する機能である。主に収集する走行情報は計測時刻、振動、速度、進行の方角の4種類である。この機能はスマートフォンアプリケーションが有するものである。

解析機能

スマートフォンで収集した走行情報を解析し、路面異常を検出する機能である。解析には統計的な手法を用いており、他のユーザの走行情報も考慮している。この機能はサーバが有する。

路面情報マップ閲覧機能

解析機能で検出した路面異常の情報を、地図に重ねあわせて“路面情報マップ”として閲覧可能にする機能である。ユーザの指定範囲に応じて路面異常情報を取り出し、地図にマッピングして表示する。この機能は主に路面異常情報の操作と地図表示の2つで構成され、路面異常情報の操作はサーバが、地図表示はクライアントが実行する。

3.3 システム構成

本節では、開発システムのハードウェア構成、およびソフトウェア構成について説明する。

3.3.1 ハードウェア構成

本システムにおけるハードウェア構成を図 3.2 に、仕様を表 3.1、表 3.2 にまとめる。なお、クライアント PC およびスマートフォンは汎用機器を想定しているため、ここでは著者が所有している端末で、なおかつ動作を確認できたものの仕様を記載する。

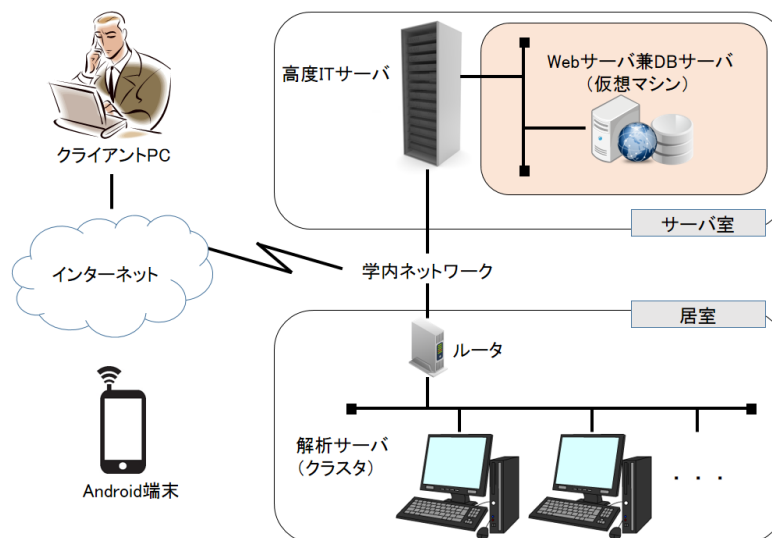


図 3.2: ハードウェア構成図

表 3.1: クライアントサイドのハードウェア構成

	クライアント PC	スマートフォン
CPU	Intel Core i5 2.6GHz	Samsung S5PC210 1.2GHz
RAM	8GB	1GB
ストレージ容量	150GB	16GB

本システムは、クライアント PC、Android 端末、およびサーバ群から構成される。サーバ群では、Web サーバ、データベースサーバ、解析サーバが構築されている。各構成要素における役割を以下に説明する。

クライアント PC

路面情報マップを閲覧するための端末としての役割を果たす。端末にインストールされている Web ブラウザを介して Web サーバへアクセスし、任意の場所における路面異常箇所を地図上に表示する。

表 3.2: サーバサイドのハードウェア構成

	仮想サーバ (Web サーバ兼データベースサーバ)	解析サーバ
CPU	Intel Xeon 2.5GHz	Intel Core 2 Duo 2.66GHz
RAM	2GB	2GB
ストレージ容量	20GB	200GB

Android 端末

路面走行時の情報を計測する役割を持つ。路面走行時の振動や進行方向などを Android 端末に搭載されているセンサを用いて計測し、走行情報ファイルとして記録する。計測終了時にはファイルを圧縮し、Web サーバへ送信する。また、クライアント PC と同様に路面情報マップを閲覧することも可能である。

Web サーバ

Web サーバは主にクライアントサイドとのデータ送受信の役割を果たす。Android 端末から圧縮された走行情報ファイルを受信すると、そのファイルの解凍処理を実行し、解凍ファイルを Web サーバ自身のストレージへ一時保存する。また、クライアント PC などから路面情報マップの表示要求を受け取ると、Web サーバはデータベースサーバへ必要な情報を問い合わせ取得し、地図を表示するためのデータを生成してクライアントサイドへ送信する。

データベースサーバ

データベースサーバは走行情報および解析結果である路面異常箇所の情報を保持しておく役割を持つ。また、問い合わせに応じて情報を出力する役割も果たす。データの操作はデータベース管理システム (DBMS) を介して実行される。

解析サーバ

走行情報を解析し、路面異常箇所を検出する役割を担う。Web サーバのストレージに保存されたファイルを取り出し、統計的な解析手法を用いて路面異常箇所を検出する。解析サーバはスケーラビリティを確保するためにクラスタ化し、複数のノードで構成されている。なお、走行情報および路面異常情報のデータベースへの格納は、解析サーバが行う。

このうち、Web サーバおよびデータベースサーバについては、専修プログラム内で利用されているサーバで 1 台のサーバとして仮想化され、稼働している。仮想化されたサーバにはグローバル IP アドレスが割り当てられており、haruna.sit.cs.tsukuba.ac.jp というドメインを有

する（以降、Web サーバ・データベースサーバを総称して haruna と呼称する）。学内からのアクセスに関しては特に制限を設けていないが、学外からのアクセスには HTTP のみ許可するものとし、80 番ポートのみを開放している。データベースサーバへは Web サーバ、解析サーバ、および学内ネットワーク内にある計算機端末からのみアクセス可能である。

解析サーバについては、学生チームが居室としている工学系学系 F 棟 3F811 室に設置し、学内のネットワークに接続して運用している。なお、学内ネットワークの制約上、解析サーバから haruna へアクセスすることはできるが、haruna から解析サーバへアクセスすることは出来ない。また、解析サーバはクラスタ化していると述べたが、現在は解析処理を行う複数のスレーブノードと、そのスレーブノードおよび Web サーバを監視し制御を行うマスタノードの構成で運用している。マスタノード、スレーブノードが持つそれぞれの役割については次節で説明する。

3.3.2 ソフトウェア構成

本システムにおけるソフトウェア構成を図 3.3、および表 3.3、表 3.4、表 3.5 にまとめる。また、クライアント PC およびスマートフォンについては、汎用機器を想定しているため、本システムがサポートするソフトウェア仕様を記載している。なお、図 3.3 において、本プロジェクトで開発した部分は青枠線で囲った部分である。以下で、それぞれのソフトウェアについて概要を述べる。

Android アプリケーション

走行情報の計測・圧縮、サーバへのデータ送信、および路面情報マップの閲覧ができるソフトウェアである。詳細については第 4 章で説明する。

地図表示（Map 表示）

路面情報マップの表示要求を受け取り、要求に応じた情報をデータベースから取得して整形したうえで要求元に返す機能を持ったものである。このモジュールは Web サーバ上に配置されている。詳細については第 4 章で説明する。

ファイル受信・解凍

Android アプリケーションから送信された走行情報の圧縮ファイルを解凍し、その解凍ファイルを Web サーバのストレージに保存する機能を持っている。このモジュールは Web サーバに配置されている。

ファイル通知

新たな走行情報ファイルが Web サーバのストレージに追加・保存されたことを解析サーバのマスタノードに通知するものである。このモジュールは Web サーバに配置されている。

解析制御

Web サーバからファイル追加の通知を受けて、解析処理をどのスレーブノードに実行させるかを制御するものである。このモジュールは解析サーバのマスタノードに配置されている。マスタノードにはどのスレーブノードが処理可能であるかが通知されており、通知された順に解析処理を実行するようスレーブノードに要求を送る。

走行情報解析

Web サーバにアクセスし、ストレージに保存されている走行情報を取得して路面異常箇所の検出処理を行うものである。検出処理が完了すると、走行情報と路面異常箇所情報をデータベースに格納したうえで、マスタノードへ自身が待機状態に遷移したことを通知する。このモジュールは解析サーバのスレーブノードに配置されており、マスタノードから処理要求を受け取った時に実行される。

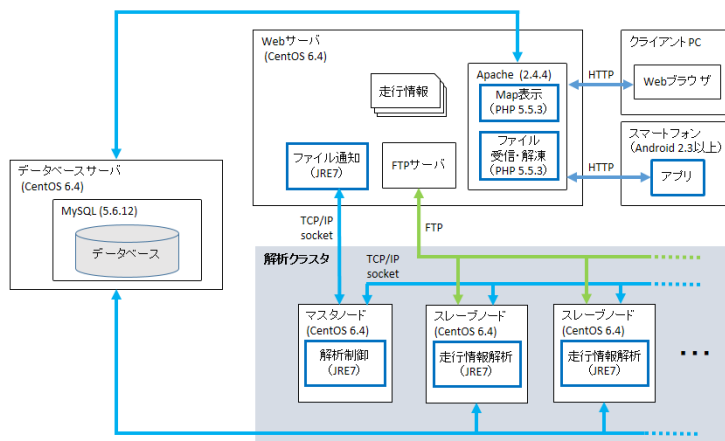


図 3.3: ソフトウェア構成図

表 3.3: クライアント PC のソフトウェア構成

ソフトウェア	動作確認バージョン
OS	Windows8 (64bit)
Web ブラウザ	Google Chrome

表 3.4: スマートフォンのソフトウェア構成

ソフトウェア	名称とバージョン
OS	Android 2.3 以降
プログラミング言語	Java

表 3.5: サーバサイドのソフトウェア構成

ソフトウェア	名称とバージョン
OS	CentOS 6.4 (64bit)
DBMS	MySQL 5.6.12
WEB サーバ	Apache 2.4.4
プログラミング言語	PHP 5.5.3
	Java
	JavaScript

3.4 開発の担当

3.4.1 メンバの開発担当

このシステムの開発において、本プロジェクトでは機能ごとに開発担当を分け、それぞれで設計・実装を進めることとした。表 3.6 にそれぞれの機能の主な開発担当をまとめる。

表 3.6: 開発担当の一覧

ハードウェア	ソフトウェア	担当メンバ
Android	Android アプリケーション	枝松
Web サーバ	地図表示	胥
	ファイル受信・解凍	
	ファイル通知	
解析サーバ	解析制御	村林
	走行情報解析	
データベースサーバ	データベース設計	林

3.4.2 著者の担当範囲

本プロジェクトにおいて、著者は主にスマートフォン向けクライアントアプリケーションの設計と実装および路面情報マップ閲覧のための Web アプリケーションの設計と実装を担当した。スマートフォン向けクライアントアプリケーションでは主に、スマートフォンに搭載

された各種センサを利用して走行時の情報を収集する機能，およびその情報を圧縮したうえでサーバ側へ送信する機能を提供する．路面情報マップ閲覧のための **Web** アプリケーションでは，解析の結果異常と判断された箇所を地図にプロットし，表示する機能を提供する．第 4 章で，クライアントアプリケーションで提供する機能や処理内容について述べる．

第4章 クライアントアプリケーションの開発

本章では、プロジェクトで著者が担当したセンサ情報収集機能を持つスマートフォン向けのクライアントアプリケーションのプロトタイプ設計と実装、および路面情報マップ閲覧のための Web アプリケーションの設計と実装について述べる。

4.1 スマートフォン向けクライアントアプリケーション

4.1.1 概要

本アプリケーションは、路面走行時の振動や進行方向などをスマートフォンに搭載されているセンサを用いて計測し、ファイルに書き出すというものである。

具体的な利用フローを説明する。本アプリケーションでユーザに必要とする操作は、アプリケーションの起動・終了を除き走行情報の計測を開始する操作、および停止する操作のみである。まず、ユーザは GPS の計測を有効にした状態でアプリケーションを起動する。次に、スマートフォンをダッシュボードや助手席などといった任意の場所に設置したうえで、走行情報の計測を開始させる。なお、設置にあたってはスマートフォンの姿勢は指定しない。そのままユーザは通常通り自動車を運転する。その際、低速で走るなどといった計測のための特別な行動はしなくてよい。この間に、スマートフォン内では自動的にセンサ情報を取得し、走行情報ファイルに書き出しを行っている。そして、目的地に到着するなど走行を終えた時点で、ユーザは走行情報の計測を停止させる。このときスマートフォン内部では、走行情報ファイルを圧縮し、解析のためにサーバへデータを送信している。以上が一連の利用の流れである。

また、段差の分布状況を確認したい場合には、ブラウザ上で路面情報マップとして閲覧することも可能である。

4.1.2 実装環境

スマートフォン向けクライアントアプリケーションにおける実装環境は以下の通りである。

- 開発マシン：TOSHIBA dynabook R732（OS:Windows 8 Pro（64bit）, CPU：Intel Core i5 2.6GHz, RAM：8GB）
- 開発環境：Eclipse（追加プラグイン：Android Development Tools¹）
- 利用ライブラリ：Android SDK²
- 開発言語：Java
- 開発に使用した Android 端末：
 - Nexus 7³（OS：Android4.2.2, CPU：NVIDIA Tegra3 1.3GHz, RAM：1GB）
 - GALAXY S II SC-02C⁴（OS:Android2.3.6, CPU:Samsung S5PC210 1.2GHz, RAM: 1GB）

次項以降、開発したスマートフォン向けクライアントアプリケーションを Android アプリケーションと呼称する。

4.1.3 実装した機能

実装した Android アプリケーションの大まかな処理を図 4.1 に示す。

なお、詳細なフローについては概念シーケンス図として付録に掲載する。以下でそれぞれの機能について詳しく述べる。

記録ファイルの作成

走行時に取得したセンサ情報を記録するためのファイルを作成する機能である。記録ファイルはアプリケーションを起動し、走行情報の計測を開始してセンサ情報を計測し始めるときに作成される。ファイル形式は CSV を採用した。理由としては、CSV ファイルの汎用性と可読性が高いこと、サーバ側での解析操作が行いやすいことが挙げられる。

センサ情報の計測

走行時の振動、走行位置などの情報をセンサを使って計測する機能である。本アプリケーションでは、走行情報を計測するためにスマートフォンに搭載されている加速度センサ、地磁気センサ、GPS センサを利用する。計測は一定間隔で行われ、走行情報の計測を停止するまで繰り返し実行される。アプリケーションで計測する情報、およびその取得間隔を表 4.1 にまとめる。なお、3 軸傾きは加速度センサ、地磁気センサの両方を用いて計測している。

¹ADT Plugin — Android Developers (URL: <http://developer.android.com/tools/sdk/eclipse-adt.html>)

²Android SDK — Android Developers (URL: <https://developer.android.com/sdk/index.html>)

³<https://support.google.com/nexus/answer/2667663?hl=ja>

⁴<http://www.samsung.com/jp/consumer/mobilephone/smartphone/docomo/SGH-N033LKNDKM>

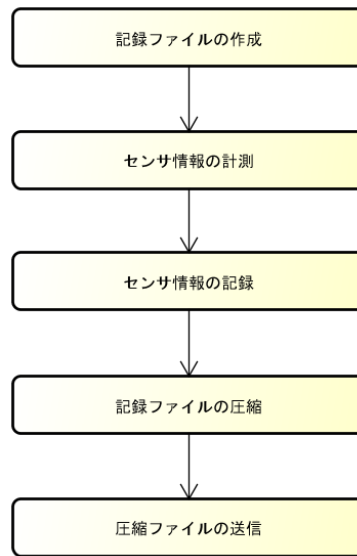


図 4.1: Android アプリケーションの処理フロー

表 4.1: センサから計測する情報

利用センサ	計測する情報	取得間隔
加速度センサ	3 軸加速度 (x, y, z), 3 軸傾き (方位, 仰角, ロール角)	40Hz
地磁気センサ	3 軸傾き (方位, 仰角, ロール角)	
GPS センサ	位置 (緯度・経度), 速度 (速さ, 進行の方角)	1Hz, または 1m の移動が発生した時

これらの情報はサーバ側で異常検出の解析処理に利用される．具体的には，加速度と傾きから鉛直方向の加速度値を計算し，これを解析することで路面の異常を検出している．位置や速度は，その異常に関する付加情報として用いられる．

センサ情報の記録

計測したセンサ情報を走行情報としてファイルに書き込む機能である．ファイルへの書き込みはセンサ情報を計測した都度行われ，走行情報の計測を停止するまで繰り返し実行される．1つのレコードには，6種類13項目が記録される．1レコードの記録項目についてまとめたものを表 4.2 に示す．

ここで，yyyy は西暦を，MM は月を，dd は日付，HH は 24 時間表記における時間，mm は分，ss は秒，SSS はミリ秒を表す．また，ファイル名は CSV のファイル名を指し，このファイル名は走行情報の計測開始時刻を付される．例えば，“2013 年 12 月 8 日 20 時 9 分 42 秒 136”

表 4.2: 1つのレコードに記録される項目

記録項目	単位・形式	種類	データ型
ファイル名	yyyyMMdd_HH:mm:ss.SSS	ファイル名	文字列型
日付	yyyy/MM/dd	センサ情報取得時刻	
時間	HH:mm:ss.SSS		
緯度	度	位置	実数型
経度			
進行の方角	度	速度	整数型
速さ	km/h		
x 軸加速度	m/s^2	加速度値	実数型
y 軸加速度			
z 軸加速度			
方位	rad	傾き	
仰角			
ロール角			

という時点で走行情報の計測を開始し，“2013年12月8日20時45分12秒77”という時点センサ取得時刻としてレコードに記録する場合，ファイル名は“20131208.200942136”，日付は“2013/12/08”，時間は“20:45:12.077”と表記される．

記録ファイルの圧縮

センサ情報を記録したファイルを圧縮する機能である．これは，走行情報の計測を停止し，センサ情報の計測・記録を終了したときに実行される．圧縮形式には **Gzip** を採用している．圧縮機能を実装した理由としては，1件あたりの走行情報ファイルが数十 MB⁵ と非常に大きく，送信時に端末・サーバ双方への負荷がかかると考えたからである．この機能により，先ほどのファイルは数 MB⁶ に圧縮される．

圧縮ファイルの送信

圧縮した走行情報ファイルを **Web** サーバへ送信する機能である．ファイルを送信するタイミングは，走行情報ファイルの圧縮完了時である．また，1回の走行情報の計測で1件のファイルをサーバへ送信する．送信プロトコルには **HTTP** を用いている．

また，オプションとして路面の異常箇所を地図上で確認できる機能も備えている．以下で詳細を述べる．

⁵1時間あたりのデータサイズはおおよそ 20MB であった．

⁶圧縮後，データサイズはおおよそ 4MB になった．

路面異常箇所の地図表示

任意の場所周辺における路面異常箇所を Google Map にプロットし、路面情報マップとして表示を行う機能である。アプリケーション起動時の初期表示ではスマートフォンの現在地を取得し、その周辺の路面異常箇所を表示するようになっている。当初、Web ブラウザを Android アプリケーションに埋め込む WebView クラスを利用して実装する予定であったが、プロジェクト進行の都合上、実装工数が不足することが判明したため、Web ブラウザから直接確認する形式に変更している⁷。Web ブラウザで閲覧する機能の詳細については、次節の実装に従っている。

4.2 地図閲覧のための Web ブラウザアプリケーション

4.2.1 概要

本アプリケーションは、端末にインストールされている Web ブラウザで任意の場所周辺における路面異常箇所を Google Map にプロットして表示を行うものである。次節以降、地図閲覧のための Web ブラウザアプリケーションを Web アプリケーションと呼称する。

4.2.2 実装環境

Web アプリケーションにおける実装環境は以下の通りである。

- 開発マシン：HP ENVY h8-1560jp（OS：Windows 8 Pro（64bit）、CPU：Intel Core i7 3.5GHz、RAM：16GB）
- 利用 API：Google Maps JavaScript API v3⁸
- 開発言語：PHP、JavaScript、HTML
- 動作確認ブラウザ：Google Chrome

4.2.3 実装した機能

実装した Web アプリケーションは、図 4.2 に示した流れで処理を行う。なお、詳細フローについては概念シーケンス図として付録に掲載する。以下で、それぞれについて詳しく述べる。

⁷この変更について、顧客とは合意済みである。

⁸<https://developers.google.com/maps/documentation/javascript/?hl=ja>

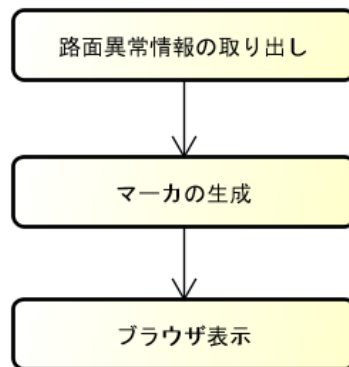


図 4.2: Web アプリケーションの処理フロー

路面異常情報の取り出し

任意の位置周辺の路面異常情報を取り出す機能である。この機能は Web ブラウザ上で地図の移動や拡大・縮小のイベントが発生した際に実行される。クライアントからは地図の中心の緯度・経度，中心からブラウザ画面右上隅までの距離の 3 情報が渡される。路面異常情報はデータベースサーバに格納されているものを取得し，XML 形式で出力する。取得する情報は，路面異常の検出日，検出位置（緯度・経度），評価値⁹，および地図の中心位置からの距離の 4 種類である。なお，次のフローで“異常”である箇所を表示するので，出力する異常情報は評価値が一定以上を超えているもののみとした。出力される XML データの例を図 4.3 に示す。加えて，XML データのフォーマットを表 4.3 に示す。

```

<markers>
  <marker detection_date="2013-12-03" latitude="36.046126582254495" longitude="140.1233451522046"
    anomaly_level="0.6355000000000001" distance="12.221098149927927"/>
  <marker detection_date="2013-12-03" latitude="36.04650071217099" longitude="140.12328747734213"
    anomaly_level="0.6342795" distance="12.233891111342345"/>
  <marker detection_date="2013-12-03" latitude="36.05345495212134" longitude="140.12169397320116"
    anomaly_level="0.7077074499999999" distance="12.453708120204112"/>
  .
  .
  .
</markers>
  
```

図 4.3: 出力される XML データの例

また，Web ブラウザに表示している範囲における全ての路面異常情報を表示するために，図 4.4 に示すように地図の中心から最も距離が離れた位置，すなわち中心から画面の右上隅の距離を計算し，この距離を半径とした円内の異常情報の一覧を取得するよう実装した。

⁹解析時に付与される本システム独自の指標である。この評価値は 0～1 の実数値を持ち，この値が高ければ高いほどその位置に異常箇所があるという可能性が高いことを表す。

属性名	内容
detection_data	異常検出日
latitude	緯度
longitude	経度
anomaly_level	評価値
distance	ブラウザ画面中心からの距離

表 4.3: XML データの属性

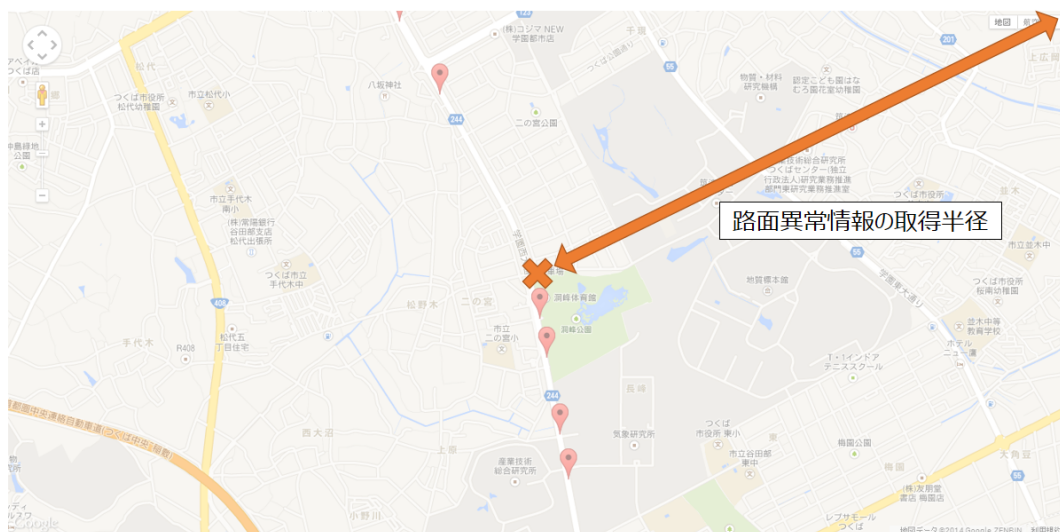


図 4.4: 取り出し範囲の半径

マーカの生成

先の「路面異常情報の取り出し」機能で取得した情報をもとにマーカを生成する機能である。この機能は前のフローで出力された **XML** データを利用するため、**XML** 形式のデータが生成されてから実行される。**XML** データから緯度と経度を取り出し、これを付加情報としてマーカに持たせる。マーカは取得した異常情報のレコード数だけ生成される。

ブラウザ表示

先のフローで生成したマーカに付された位置情報に基づき、**Google Map** に重ね合わせて表示する機能である。実際に、つくば市周辺の路面情報マップを図 4.5 として示す。



図 4.5: つくば市周辺の路面情報マップ

第5章 開発アプリケーションの評価実験

本章では、開発した Android アプリケーションにおいて考慮すべき課題を述べたうえで、それらの課題を議論するために行った実験とその結果を示す。そして、結果をもとに考察を行う。

5.1 スマートフォン向けアプリケーションにおける検討すべき課題

本プロジェクトでは、スマートフォン向けアプリケーションで検討すべき主な課題として、低負荷処理と省パケット通信の2点を抽出した。以下でそれぞれ課題の詳細を述べる。

5.1.1 低負荷処理

開発アプリケーションでは、走行情報の計測・記録、およびサーバへの送信を行っている。アプリケーションのターゲットは一般に道路を利用しているユーザを想定している。したがって、本システムをサービスとして提供するにあたっては、ユーザビリティを十分に考慮する必要がある。スマートフォンアプリケーションの利用者動向に関する調査 [16] によれば、アプリケーションの利用に関して、“電池の消費速度”や“端末動作速度”について不安を感じるユーザが多いことが明らかになっている。これより、電池の消耗や端末の動作速度に影響を及ぼすようなアプリケーションは、敬遠される可能性があると考えられる。

本システムでは多くのユーザから走行情報を集めて精度を向上させるという手法であるため、可能な限りユーザから走行情報を収集できることが望ましい。我々はこの課題に対処し、電力消費を抑えるための設計について議論すべきだと考えた。

5.1.2 省パケット通信

先にも述べたように、開発アプリケーションは走行情報をサーバへ送信するという機能を持つ。本システムでは大量にデータを取り扱うことを想定しており、何らかの対策を講じなければネットワークのトラフィック増大やサーバ処理の高負荷化などが起きる。

またデータを送信するにあたって、パケット量や通信料などは基本的にアプリケーションユーザが負担することになる。本システムをサービスとして稼働する場合、低負荷処理の課題と同様にユーザビリティの観点から通信量を抑える必要があると考えられる。

以上より，本プロジェクトではどのようにパケット通信を低減させるか，すなわち走行情報のデータサイズをどのように抑えたらよいか検討すべきだと考えた。

これらの課題を検討するために，本プロジェクトでは評価実験を行うことにした．低負荷処理の課題に対しては，スマートフォンの消費電力の観点から議論する．省パケット通信に関する課題では，データの省サイズ化を試みることで検討を行う．5.2 節，5.3 節で，それぞれの実験内容と結果について述べる．

5.2 低負荷処理に関する実験

低負荷処理に関して，我々はスマートフォンの消費電力にフォーカスを当てることにした．電力消費の要因としていくつかの仮説を立て，これを検証するための実験を実施した．

5.2.1 仮説

我々は，スマートフォンの電力を消費する要因として次の3つがあると考えた．

- GPS の測位
一般に，地図アプリケーションなどで GPS を利用していると電力の消費が激しくなると考えられている．
- センサのサンプリング周波数
センサのサンプリング周波数が高いと，それだけセンサデータを計測する頻度が高くなるということになり，消費電力が増大するものと予想される．
- 画面バックライト
一般に，スマートフォンの消費電力が大きいと考えられている要因である．

5.2.2 検証

5.2.1 項で述べた要因がスマートフォンにおける電力消費の原因となっているかを検討するために，アプリケーション利用時の消費電力量を計測し，比較することにした．

実験内容

本実験では，様々な条件を組み合わせで Android アプリケーションを動作させる．そして，一定時間経過後のバッテリー残量を調べて消費電力量の計算を行う．

アプリケーション要件

実験対象のアプリケーションでは、以下の要件が実装されている。

- 走行情報 CSV ファイルを生成する
- センサ情報を一定間隔で取得し、それを CSV ファイルに記録する
- CSV ファイルを Gzip 形式で圧縮する

このアプリケーションで記録するセンサ情報は、ファイル名、センサ情報取得時刻（日付・時間）、位置（緯度・経度）、速度（速さ・進行の方角）、3 軸加速度 (x, y, z)、傾き（方位・仰角・ロール角）、3 軸地磁気 (x, y, z)、照度の計 8 種類 17 項目である。位置、速度のデータのみ、サンプリング周波数を 1Hz に固定する。また、WiFi 機能は無効にし、3G 回線もしくは LTE 回線下で本実験を行うこととした。

実験手順

本実験は、以下の手順で行った。

1. アプリケーション開始前の Android 端末のバッテリー残量 [%] を記録する。
2. 各実験条件下で Android アプリケーションを実行してセンサ情報の記録を開始する。
3. 1 時間経過後、アプリケーションを終了する。このときのバッテリー残量 [%] を記録する。

各実験条件を以下に記す。

実験条件

- 条件 1
画面バックライトを 常時点灯 させた状態でアプリケーションを実行する。GPS 機能は 無効 にし、加速度などのセンサデータのサンプリング周波数は 100Hz に設定する。端末は固定し、静止 させておく。
- 条件 2
画面バックライトを 常時点灯 させた状態でアプリケーションを実行する。GPS 機能は 有効 にし、加速度などのセンサデータのサンプリング周波数は 100Hz に設定する。端末は次の 2 通りの条件で設置する。
 - － 条件 2-a : 位置が取得できた時点で端末を固定し、静止 させておく。
 - － 条件 2-b : 端末を常に 移動 させておく。

- 条件 3
アプリケーションの実行開始とともに画面バックライトを 消灯 する。GPS 機能は 有効 にし、加速度などのセンサデータのサンプリング周波数は 100Hz に設定する。端末は固定し、静止 させておく。
- 条件 4
アプリケーションの実行開始とともに画面バックライトを 消灯 する。GPS 機能は 有効 にし、加速度などのセンサデータのサンプリング周波数は以下の 4 通りに設定する。端末は固定し、静止 させておく。
 - － 条件 4-a : 20Hz に設定して計測する。
 - － 条件 4-b : 40Hz に設定して計測する。
 - － 条件 4-c : 60Hz に設定して計測する。
 - － 条件 4-d : 80Hz に設定して計測する。

条件の各要素をまとめたものを表 5.1 に示す。条件 1 と条件 2-a を比較することで GPS の消費電力を、条件 2-a と条件 3 を比較することで画面バックライトの消費電力を計測できる。センサデータのサンプリング周波数の違いにおける消費電力の差を観察するには、条件 3 と条件 4 を比較すればよい。また、条件 2-a、2-b で移動時と静止時の消費電力量の差がないか検証する。

表 5.1: 各実験条件のパラメータ設定の一覧

	画面バックライト	GPS 機能	サンプリング周波数 [Hz]	端末の状態
条件 1	○	×	100	静止
条件 2-a	○	○	100	静止
2-b				移動
条件 3	×	○	100	静止
条件 4-a	×	○	20	静止
4-b			40	
4-c			60	
4-d			80	

実験端末

実験端末には、表 5.2 に示した 3 台を利用した。

表 5.2: 実験端末の一覧

端末機種名	OS バージョン	バッテリー容量 [mAh]	バッテリー電圧 [V]
GALAXY S II SC-02C	Android 2.3.6	1650	3.7
Xperia A SO-04E	Android 4.2.2	2300	3.6
ARROWS X F-10D	Android 4.0.3	1800	3.7

5.2.3 結果

本項では、各実験を実施しそれぞれの条件を比較した結果について述べる。なお、以降の結果で示す消費電力量 $P[W]$ は、次式に基づいて算出している。

$$P = \frac{(J_{before} - J_{after})}{100} * C * E * 1000$$

ここで J_{before} は実験前のバッテリー残量 [%], J_{after} は実験後のバッテリー残量 [%], C は実験端末のバッテリー容量 [mAh], E は実験端末のバッテリー電圧 [V] を表す。

要素ごとの消費電力の割合

条件 1, 条件 2-a および条件 3 を比較し、GPS と画面バックライトの要素ごとの消費電力量の割合を示したグラフを図 5.1 として示す。アプリケーション全体の消費電力量は 1.35W であった。このうち、全体に占める GPS の消費電力量は 1 時間で平均 0.09W であった。これは、条件 2-a におけるアプリケーション全体の消費電力のうち、約 7 % を占めていることになる。同様に、全体に占める画面バックライトの消費電力量は 1 時間で平均 0.22W であった。これは、条件 2-a におけるアプリケーション全体の消費電力のうち、およそ 17 % を占めていることになる。

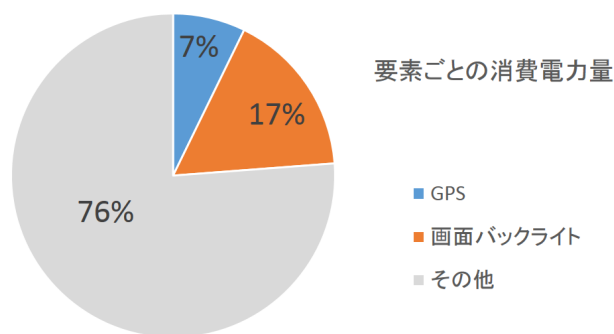
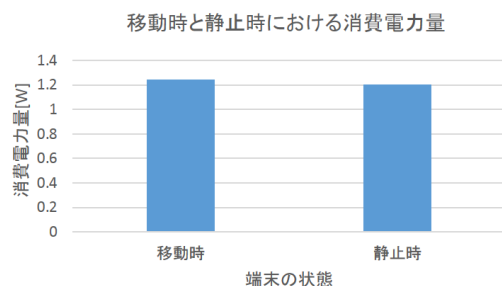


図 5.1: 要素ごとの消費電力の割合



端末の状態	消費電力量 [W]
移動時	1.24
静止時	1.20

表 5.3: 消費電力量の平均値

図 5.2: 端末の状態別の消費電力

移動時と静止時における消費電力量の違い

条件 2-a, 条件 2-b を比較し, 端末の移動時と静止時における消費電力の差をグラフ化したものを図 5.2 として示す. なおこの比較にあたっては, 条件 2-b の実験を実施できた端末が GALAXY S II のみであったため, この端末に限定した結果を示している. 表 5.3 にも示しているように, 端末の状態別における平均消費電力量は移動時が 1.24W, 静止時が 1.20W であった.

サンプリング周波数別の消費電力

条件 3, 条件 4 を比較し, サンプリング周波数の違いによる消費電力の差をグラフ化したものを図 5.3 として示す. このグラフでは, 縦軸に端末の消費電力量を, 横軸にサンプリング周波数をとっている.

3 つの端末における平均の消費電力量は, 20Hz では 0.81W, 40Hz では 1.09W, 60Hz では 0.98W, 80Hz では 0.97W, 100Hz では 1.03W であった.

本実験において, ある 1 回の試行での走行情報ファイルのサイズは表 5.4 に示したようになった. なお, この端末は GALAXY S II である.

サンプリング周波数 [Hz]	走行情報ファイル (CSV 形式) [MB]	圧縮ファイル (Gzip 形式) [MB]
20	11.40	2.28
40	28.67	5.07
60	36.85	6.41
80	31.49	6.21
100	55.78	9.77

表 5.4: サンプリング周波数ごとの走行情報ファイルのサイズ

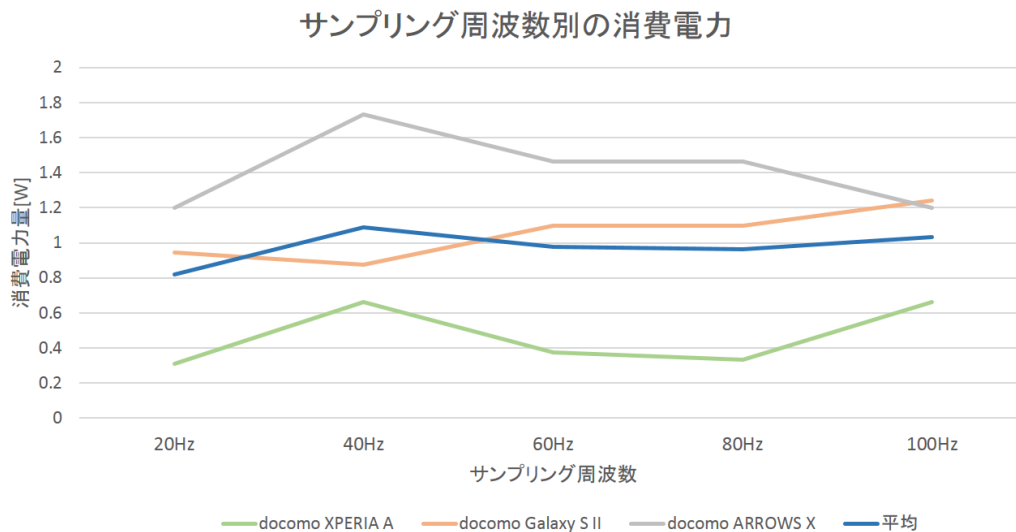


図 5.3: サンプリング周波数別の消費電力

5.3 省パケット処理に関する実験

省パケット処理に関して、我々は走行情報ファイルの省サイズ化にフォーカスを当てることにした。ファイルのサイズが大きくなってしまいう要因について仮説を立て、これを検証するための実験を実施した。

5.3.1 期待される結果

現行のアプリケーションでは、センサ情報を CSV ファイルに書き出している。CSV ファイルは文字列で記録されているため、ファイルのサイズが大きくなっていると考えられる。そこで本プロジェクトでは、センサ情報を文字列ではなくバイナリデータに変換して書き出しを行えばファイルのサイズを抑えることができるのではないかと仮説を立てた。

5.3.2 実験による評価

5.3.1 項で述べた仮説を検証するために、センサ情報の記録機能を拡張したうえで走行情報の計測を行い、走行情報ファイルのサイズを比較することにした。

実験内容

本実験では、アプリケーションを動作させてセンサ情報を CSV ファイルとバイナリファイルの二形式で書き出しを行う。そして、一定時間経過後のファイルサイズを調べて比較する。

アプリケーション要件

実験対象のアプリケーションでは、以下の要件が実装されている。

- 走行情報 CSV ファイルを生成する
- 走行情報バイナリファイルを生成する
- センサ情報を一定間隔で取得し、これを CSV ファイルに記録する
- センサ情報を一定間隔で取得し、これをバイナリファイルに記録する
(なお、記録のタイミングは CSV ファイルの記録と同時に行う)
- CSV ファイルを Gzip 形式で圧縮する
- バイナリファイルを Gzip 形式で圧縮する
- CSV ファイルを圧縮したファイルをサーバに送信する

このアプリケーションで記録するセンサ情報は、ファイル名、センサ情報取得時刻（日付・時間）、位置（緯度・経度）、速度（速さ・進行の方角）、3 軸加速度 (x, y, z)、傾き（方位・仰角・ロール角）の計 6 種類 13 項目である。サンプリング周波数については、位置および速度は 1Hz、加速度および傾きは 40Hz に設定する。通信の形態については 3G 回線や LTE 回線を持たない端末についてのみ WiFi 機能を有効にしてもよいものとし、基本的には 3G 回線もしくは LTE 回線下で本実験を行うこととした。

次で、実験のために追加で実装したバイナリファイルの作成機能の設計について説明する。

アプリケーションの機能拡張

本実験を実施するにあたり、バイナリファイル作成の機能を追加で実装した。センサ情報をバイナリ化するにあたり、本実装も視野に入れ、解析サーバ側で復元できることが必要であると考えた。そこで本プロジェクトでは、バイナリ化の前準備として計測値の上限値と下限値を定めて、センサ情報の種類ごとにビット数を固定することとした。各センサ情報の元データの上限値、下限値、およびビット数をまとめたものを表 5.5 に示す。

本実装では、実数型における小数部分の有効桁数を 6 桁としてデータの簡略化を図っている。なお、この桁数は解析精度を維持するのに必要な桁数として定めている。

各センサ情報におけるビット数は次の仕様で決められている。

仕様 1

上限、下限がともに整数型であり、なおかつ負数を取らない場合は、上限値を含むようなビット数とする。

表 5.5: 各種センサ情報の上限, 下限, およびビット数

センサ情報	上限値	下限値	ビット数
計測開始時刻	2^{64}	0	64
緯度	90.000000	-90.000000	28
経度	180.000000	-180.000000	29
3 軸加速度 (x, y, z)	64.999999	-64.999999	27
進行の方角	360	0	9
速さ	256	0	8
傾き (方位, 仰角, ロール角)	3.141592	-3.141592	23
現在時刻におけるミリ秒	999	0	10

仕様 2

上限, 下限が実数型であり, なおかつ負数をとる場合は絶対値を取った時の最大値について整数部分, 小数部分に分け, それぞれの部分に対して仕様 1 を適用する. その後, 整数部分, 小数部分のビット数を足しあわせ, 更に符号部分の 1 ビットを加えたものをその情報が持てるビット数とする.

例えば, 進行の方角は 0 度~360 度までの整数値を取るなので, 仕様 1 を適用する. 360 は 256 ($= 2^8$) より大きく 512 ($= 2^9$) より小さいので, この情報は 9 ビットあれば良いということになる. また, 3 軸加速度を例にとると, これらは -64.999999~64.999999 までの実数値を取るなので, 仕様 2 を適用することになる. 絶対値の最大値について整数部分, 小数部分に数値を分けると, 64 と 999999 となる. それぞれに仕様 1 を適用すると, 整数部分のビット数は 6, 小数部分のビット数は 20 と求められる. 符号は正負で 1 ビットで表せる. これを全て足し合わせると $6 + 20 + 1 = 27$ となり, 3 軸加速度の情報は 27 ビットあれば良いことになる.

ただし, 計測開始時刻については Java の `System.currentTimeMillis()` メソッドを利用しており, 1970 年 1 月 1 日の 00:00:00.000 からの経過時間をミリ秒 (long 型 64 ビット) で取得するように実装しているため, これらの仕様は適用せずに 64 ビットと定めている. CSV ファイルでは “2013/12/20” のように日付を文字列で記録していたが, この実装では計測開始時刻をある起点からの経過時間として数値で取得したのちにバイナリ化を行うので, データサイズを小さくすることが可能である.

続いて, センサ情報をバイナリファイルに書き込む手順について説明する. バイナリファイルはヘッダ部分とボディ部分を持つ. ヘッダ部分に持たせる情報, およびボディ部分に持たせる情報は次の通りである.

- ヘッダ部分 (288 ビット)

- 計測開始時刻
- 緯度・経度
- 3 軸加速度
- 進行の方角・速さ
- 3 軸傾き

- ボディ部分 (1 ブロック : 234 ビット)

- 現在時刻におけるミリ秒
- 緯度・経度
- 3 軸加速度
- 進行の方角・速さ
- 3 軸傾き

数値のバイナリ化は、先の表 5.5 で定めたビット数で行われる。桁数が足りない場合には、右詰めゼロパディングを行って固定長サイズにする。各センサ情報が表 5.5 で定めた上限、下限の範囲を超えた場合には、その情報はエラーとし、全てのビットを 1 でパディングする。

バイナリファイルの書き込みは図 5.4 のような流れで実行される。ヘッダの情報は一度のみ、ファイルの先頭に書き込まれる。ボディの情報はヘッダに続いて記録され、センサが一定間隔で取得されるたびに繰り返し記録される。つまり、1 レコードを 1 つのブロックとして扱い、そのブロックを繰り返し記録するという処理を行っていることになる。

また計測終了時において、バイナリファイルに書き込んだボディのブロック数によっては、ファイル全体としてビット数が足りない、すなわち中途半端なビット数で終わってしまうためにバイト単位で取り扱えない場合がある。この場合、バイト単位で扱えるようになるまでゼロパディングを行うよう実装した。

ファイルへの書き込みにおいて、CSV ファイルにおけるフォーマットとバイナリファイルにおけるフォーマットが異なるので、その違いについて述べる。CSV ファイルでは、可読性およびデータベース操作の容易さという観点から、各種センサ情報をそのまま文字列として扱い、カンマ区切りで記録するよう実装していた。これに対しバイナリファイルでは、サイズを抑えるという観点から、重複するデータを省いて記録するよう実装している。具体的には、CSV ファイルに持たせていた“ファイル名”という項目は、既に生成したファイルに付されている名前で省略可能であるので、バイナリファイルにおいてこの項目の書き込みは行っていない。またセンサデータの取得時刻についても、バイナリ化では取得時刻のミリ秒のみを書き出している。これは、センサ情報がある時刻を起点として時系列的にセンサリングしているという仕様に基づき、取得時刻におけるミリ秒さえ分かれば、計測開始時刻との差分

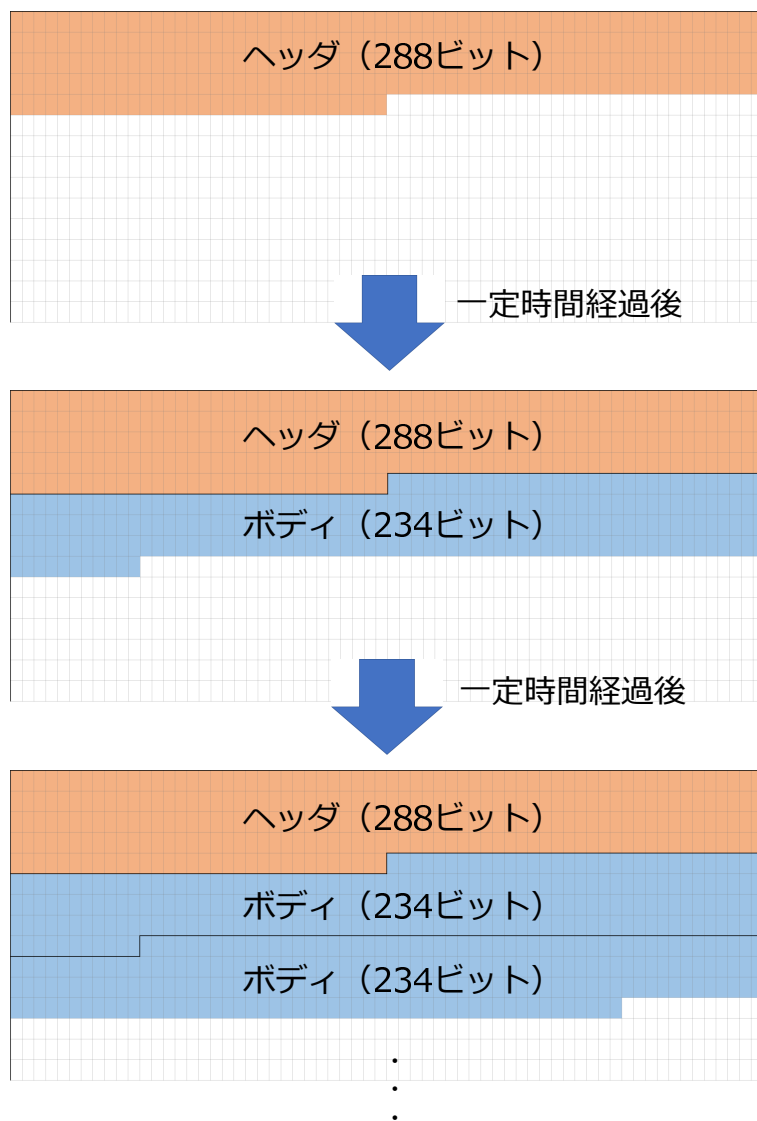


図 5.4: バイナリファイルへの書き込み

を取ることで取得時刻を導出できると考えたからである¹。

ここまで、検証のために追加実装したバイナリファイルの作成機能について説明した。次で、このアプリケーションを使ってどのような手順で実験を実施したのかについて述べる。

実験手順

本実験は、以下の手順で実施した。

1. 機能拡張後の **Android** アプリケーションを実行し、センサ情報の記録を開始する。端末は移動している状態にする。
2. 1 時間経過後、アプリケーションを終了する。
3. この試行で生成された **CSV** ファイル、**CSV** ファイルを圧縮した **Gzip** ファイル、バイナリファイル、およびバイナリファイルを圧縮した **Gzip** ファイルにおけるそれぞれのファイルサイズ [MB] を記録する。

実験端末

実験端末には、表 5.6 に示した 2 台を利用した。

表 5.6: 実験端末の一覧

端末機種名	OS バージョン	CPU
Optimus it L-05D	Android 4.0.4	Qualcomm Snapdragon S4 1.5GHz
Nexus 7	Android 4.2.2	NVIDIA Tegra3 1.3GHz

5.3.3 結果

表 5.7: 各ファイル形式の 1 時間あたりのサイズ [MB]

	CSV 形式	CSV 圧縮	バイナリ形式	バイナリ圧縮
#1	16.0	4.49	3.96	3.06
#2	17.89	4.67	4.02	3.13
#3	28.66	6.25	6.43	4.29
平均	20.85	5.14	4.80	3.49

¹バイナリファイルの復元はサーバ側の処理であり、著者の担当外であるため本報告書では説明しない。

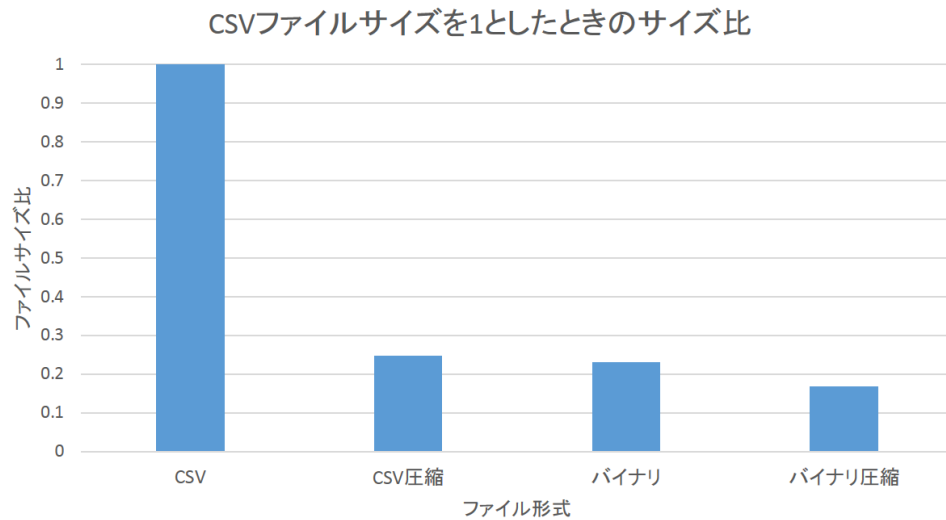


図 5.5: CSV ファイルのサイズを 1 としたときの各ファイルのサイズ比

実験の結果を表 5.7 にまとめる．また，CSV ファイルのサイズを 1 とした場合における各ファイルのサイズ比を表したものを図 5.5 に示す．このサイズ比は表 5.7 に示した平均値から算出している．CSV 形式のファイルサイズを 1 としたとき，CSV を圧縮したファイルのサイズ比は 0.24 であった．また，バイナリ形式のサイズ比は 0.23，バイナリを圧縮したファイルのサイズ比は 0.16 であった．

5.4 考察

5.2 節，5.3 節の結果について考察を行うとともに，抽出した課題について議論する．

まず，GPS 測位における移動時と静止時の消費電力量の差についてであるが，図 5.2 に示したように，状態の違いで電力の消費量に大きな差は見られなかった．この結果より，他の条件において端末を固定して静止状態にしている場合でも，移動時と同様の結果が得られると考えられる．

一般に消費電力が大きいと考えられている GPS については，全体の消費電力量に対して 1 割を切っており，バッテリー消耗の大きな要因とはなっていないことが分かった．したがって GPS の測位に関しては，本アプリケーションでは現行のサンプリング周波数のままで良く，その他の省電力化の対応を講じる必要はないと考えられる．

画面バックライトは，結果から全体のおよそ 2 割を占めていることが読み取れる．このことから，スマートフォン向けアプリケーションにおいて電力の消費を抑えるには，画面を消灯した状態で利用できるように設計にすべきであると言える．M2M システムという観点から

見れば、利用者が意識せずに使えるシステムであることが望ましいため、画面のないアプリケーション、すなわち常にバックグラウンドで動作するというアプリケーション設計にするのが良いと考えられる。

GPS と画面のバックライトの両方を利用したアプリケーションに、地図アプリケーションがある。一般に、地図アプリケーションを利用していると GPS の測位が行われるために消費電力が高くなってしまうと言われている。しかしながら、この実験結果を考慮すると地図アプリケーションにおいて実際に電力を消費している大きな要因は画面バックライトであると推測される。

センサ情報のサンプリング周波数における消費電力量については、端末ごとにばらつきはあるものの、周波数に応じて消費電力量が極端に増加するという傾向は見られなかった。この結果より、センサ情報の取得において、サンプリング周波数をある程度高く設定しても電力の消費には大きく影響しないと考えられる。つまり、サンプリング周波数を高く設定して詳細なセンサデータを記録することによって、Android 端末側の負荷を抑えたまま路面異常解析の精度を向上させることができ、システムの質を上げられる可能性があるということになる。しかしながら、サンプリング周波数とデータサイズにはトレードオフの関係がある。開発したシステムでは、サーバで大量のデータを取り扱うことを想定しており、サーバ側の負荷も考慮する必要がある。本システムにおいて、路面の異常を検出するのに必要なサンプリング周波数は最低でも 40Hz であることが分かっている [17]。また、文献 [18] の実験結果によれば、路面異常と判断できる最低の走行回数はサンプリング周波数によって異なっており、40Hz では 164 回、60Hz では 54 回、80Hz では 40 回という結果であった。すなわち、アプリケーション側のサンプリング周波数を 40Hz に設定してしまうと、他の周波数に比べて総データ量が多くなり、データベースサーバに負荷がかかってしまうということである。これらの結果と表 5.4 の圧縮後のファイルサイズを総合して考えると、本アプリケーションではサンプリング周波数を 80Hz に設定するのが望ましいと考察される。

省パケット処理に関しては、図 5.5 から現在のアプリケーション仕様で用いている Gzip 圧縮の CSV ファイルとバイナリ形式ファイルにおいて、データサイズに極端な差は見られないことが分かった。今回は実装のしやすさから CSV 形式で走行情報を記録し、Gzip 形式で圧縮してサーバに送信するという手法を採用したが、サーバ側の処理負荷を考慮すれば、走行情報はバイナリ形式で記録して送信する方が良いと考えられる。ただし、現在本アプリケーションは取得したセンサ情報のほとんどが何も加工を行わずに記録する仕様となっている。したがって、前回取得したセンサ情報との差分を取るなどといった、センサ情報自体の加工を行う工夫をすればデータのサイズを軽量化できると推測される。この点については、クライアントとサーバの処理配分にも関わってくる部分であるため、どのようにセンサ情報を加工するかも含めて今後調査・検討を行う必要がある。

第6章 プロジェクトの振り返り

著者は本プロジェクトにリーダーとして参加し、プロジェクトの進行に関与した。本章では、プロジェクト・マネジメントの観点から本プロジェクトについて振り返る。

6.1 プロジェクトの基本方針

6.1.1 開発の方針

本プロジェクトの開発プロセスには、インクリメンタルモデルを採用した。採用の理由としては、インクリメンタルモデルが開発途中における要件追加や変更に対応しやすいモデルであり、本プロジェクトにおいては、システムの課題を検討するために変更が多く発生する可能性があると判断したからである。これに伴い、本プロジェクトでは要件定義フェーズ時に開発範囲を複数のスコープに分割することとし、スコープごとに実装を行うこととした。

また、開発手法には Redmine¹ を用いたチケット駆動型開発を実施することとした。この手法を利用し作業をチケットとして記録することで、各個人の作業時間やプロジェクトの進捗状況を定量的に把握することができると考えたからである。

6.1.2 コミュニケーション計画

本プロジェクトでは、学生チーム内でのコミュニケーションのほか、顧客とのコミュニケーション、および課題担当教員とのコミュニケーションをとることとした。それぞれの会議体についての計画は次の通りである。

チームミーティング

学生メンバ全員が参加し、各個人の進捗状況および短期的な計画についての報告と共有を行う。

顧客ミーティング

学生メンバ全員と顧客、および課題担当教員が参加し、対面形式で打ち合わせを行う。システム開発に関する議論やプロジェクトの進行状況などを報告する。

¹Redmine (URL: <http://www.redmine.org/>)

進捗報告

学生メンバ全員と課題担当教員が参加し、学生メンバが課題担当教員へプロジェクトの進行状況を報告する。報告は対面形式、またはメールで行うこととした。

6.1.3 役割分担

本プロジェクトでは、メンバ間で開発の役割分担を行い、その役割に基づいて課題を抽出・検討していくこととした。チームメンバの開発担当を大まかにまとめたものを表 6.1 に示す²。

表 6.1: チームメンバの開発担当

メンバ	開発担当
枝松ちさと	クライアントアプリケーション
林 貴哉	データベース
村林 竜司	解析サーバ
胥 徳文	Web サーバ

6.2 スケジュールの計画と実績

本プロジェクトの実施期間は、2013 年 5 月より 2014 年 1 月までであった。プロジェクトの計画および実績を図 6.1 に示す。

以下で、各フェーズの概要を述べる。

例題設定フェーズ

世の中における問題をブレインストーミングを通じて洗い出し、これらを例題として本プロジェクトのテーマを適用できるか検討した³。このフェーズにおける成果物は、ブレインストーミングにおいて作成した 4 象限マトリクスである。

要件定義フェーズ

例題設定フェーズにおいて決定した例題について、解決すべき問題点の整理とそれを満たすためのシステム要件の定義を行った。要件の整理がまとまった時点で、6.1 節で述べたプロジェクトの方針に基づき、開発スコープを 2 つに分割した。分割後のスコープを図 6.2、図 6.3 に示す。

²詳細な開発担当については 3.4 節を参照されたい。

³詳しくは第 2 章を参照されたい。

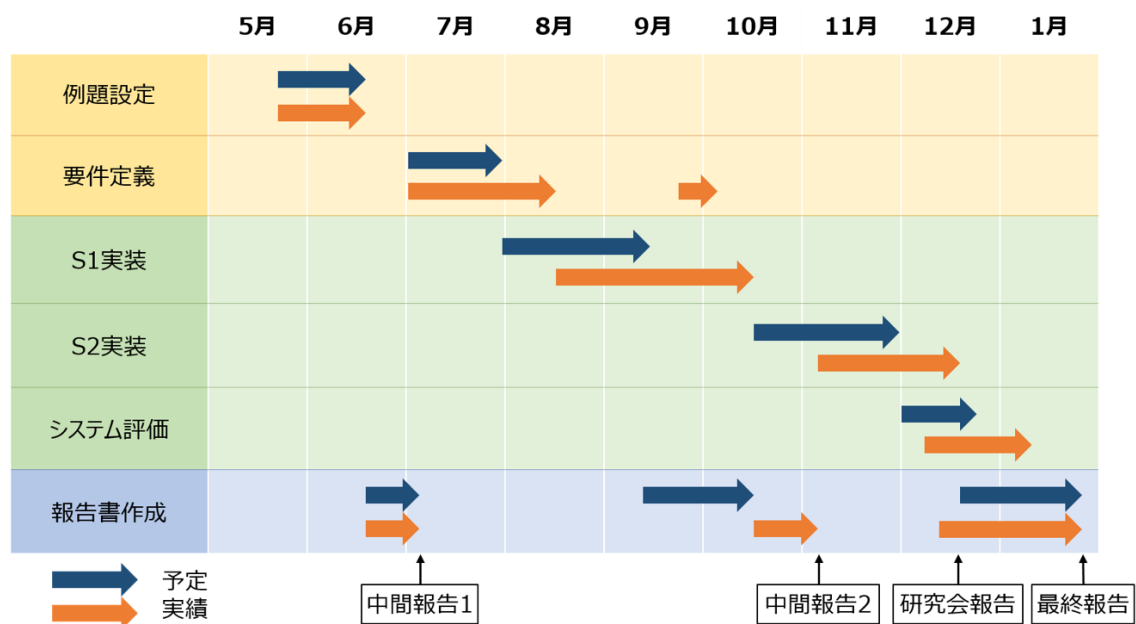


図 6.1: 本プロジェクトのスケジュール計画と実績

スコープ 1 (S_1) では、走行情報の収集や、路面異常箇所検出のための解析アルゴリズムの実装などといったシステムが稼働できる最低限の機能を開発することとした。スコープ 2 (S_2) では、ユーザがシステムをサービスとして利用するために必要な機能を開発することとした。具体的には、路面異常箇所を地図に重ね合わせて表示する機能や、その地図を閲覧するための機能である。その他に、路面異常の検出精度を向上させるための解析手法の改善を行うこととした。

このフェーズにおける成果物は、システムの開発構想についてまとめたシステム構想書、および開発システムの要件についてまとめた要件定義書である。

実装フェーズ（スコープ 1）

要件定義フェーズで抽出したシステム要件のうち、 S_1 の設計および実装を行った。ここで、サーバサイド、クライアントサイドでそれぞれ開発の担当を決めてそれぞれで開発を進めた。

実装フェーズ（スコープ 2）

要件定義フェーズで抽出したシステム要件のうち、 S_2 の設計および実装を行った。またこの間に、研究会⁴の準備と S_1 の開発において発見された課題についての予備検討を始めた。

⁴情報処理学会 第 199 回計算機アーキテクチャ・第 142 回ハイパフォーマンスコンピューティング合同研究発表会にて発表。

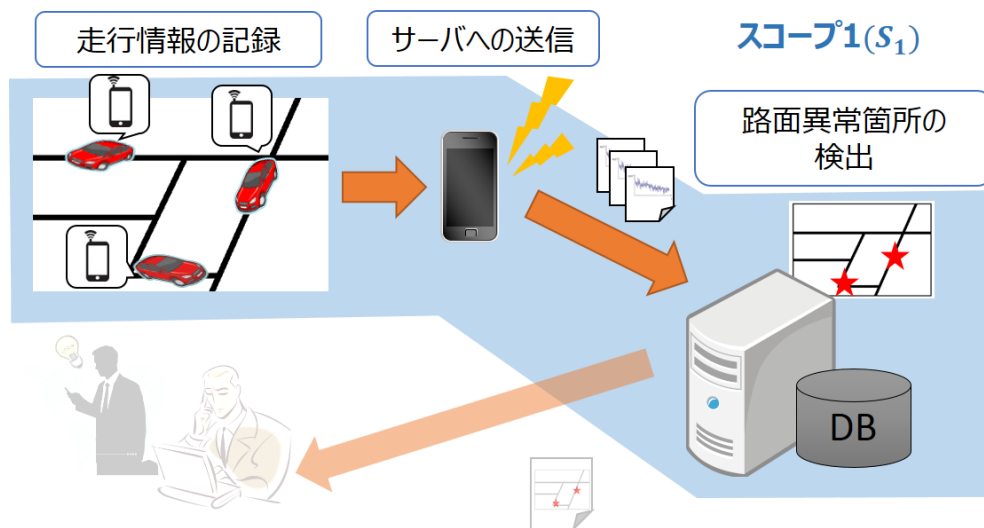


図 6.2: スコープ 1 の開発範囲

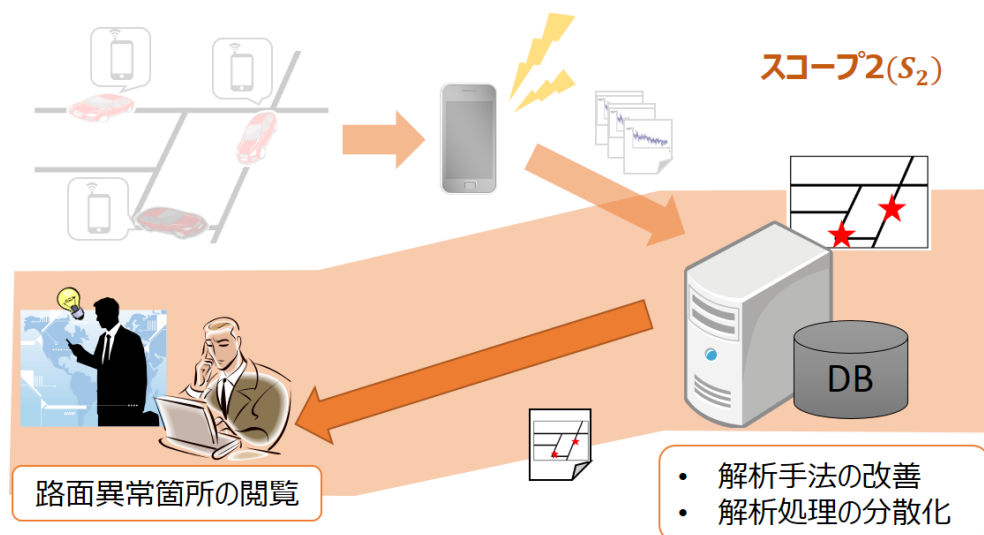


図 6.3: スコープ 2 の開発範囲

このフェーズにおいて、メンバ1名が個人の都合によりチームから外れたため、そのメンバが担当していたタスクを残りの3名で担当することとなった。

システム評価フェーズ

開発したシステムの評価を行った。具体的には、要件定義フェーズ、実装フェーズで発見された課題について議論するための負荷実験や、解析アルゴリズムの評価実験などである。実験はサーバサイド・クライアントサイドに分けて行った。

6.3 考察

本節では、プロジェクト遂行上に起こった問題について述べ、その原因について考察する。

6.3.1 スケジュールの遅延

図 6.1 に示したように、プロジェクトのスケジュールに遅延が発生した。これにより、テストに十分な時間を割くことが出来なかった。遅延発生の原因としては、以下の項目が要因であると考えられる。

- 要件定義フェーズの長期化
要件定義フェーズにおいて、要件の洗い出し・整理に予想よりも時間を要した。これに派生して、関連ドキュメントの修正が繰り返し発生したためさらなる遅延が生じ、工程完了が遅れた。
- 学習コストの肥大化
Android におけるクライアントアプリケーション、PHP でのサーバサイドアプリケーションの開発を行ったことのあるメンバが少なかった。これにより、機能の実現可能性や必要技術などについて追加の調査を要することとなり、さらにこの調査期間に多くの時間を費やしていたため、遅延が生じたと考えられる。
- 要件定義フェーズ・実装フェーズで抽出した課題の検証の長期化
要件定義フェーズにおいて、要件を整理した段階でシステムの開発に関する課題が発見された。同様に、スコープ 1 における実装フェーズの途中でもいくつか発見された。これらの課題検討を実装と並行して進行させたため、実装フェーズに遅れが生じた。
- 制約による設計変更
当初の設計では、Web サーバが直接解析サーバにアクセスするという仕様になっていた⁵。しかし大学においてアクセスに関する制約があり、Web サーバから解析サーバへ

⁵現在のアーキテクチャ構成は第 3 章を参照。

のアクセスは不可であることが判明したため、追加で解析サーバから Web サーバへアクセスするための機能を追加するなどといった大幅な設計変更が発生した。このため設計と実装の工数が増加し、スケジュールが遅れた。

6.3.2 運用の一部形骸化

6.1 節でも触れたが、当初、本プロジェクトでは定量的な評価を可能にするためにチケット駆動開発を採用し、Redmine の導入・活用を開発方針として掲げた。しかしながら、チケット発行・進捗の記録方法のルールが曖昧であったことや、作業に集中してしまい、チケットの発行を忘れてしまうということがあったために、Redmine の運用が形骸化してしまった。このため、進捗状況や工数の記録がまばらとなり、それぞれのフェーズでどれくらいの工数を要していたのかなどといった定量的な測定が出来ていない。これに伴って、各メンバーの進捗を把握しづらい状況が発生し、フェーズの完了が遅れるという影響もあった。

6.4 リーダの立場からの振り返り

著者は本プロジェクトにリーダーとして参加し、進行に関与した。本節では、リーダーとしての立場からプロジェクトを振り返る。

著者がプロジェクトを円滑に進行させるうえで重要な要因であると考えていたことは、メンバー間の意識共有を十分に行うことであった。そのために、チームミーティングは可能な限り全員を参加すること、会議体ごとにしっかり議事録をとっておくこと、作成したドキュメントについてはレビューを綿密に行うなどといった施策を講じた。これらの施策によって、上流工程での意識合わせや生産的な議論を交わすことができたと考えられる。その半面、理解できるドキュメントの作成などといった可読性を向上させる取り組みに注力してしまうこととなり、下流工程に遅れが生じてしまうという問題が発生した。この遅れを取り戻すために、本プロジェクトでは開発を分担し属人的に実装を進めることで対応した。結果として、開発スケジュールの遅れは最小限に留められたものの、メンバー間の情報共有が十分に行えず、個人の進捗が不透明化するという弊害が発生した。今回の実績をふまえると、プロジェクトを進めるには単純に意識共有を行うだけでなく、工程別にコミュニケーションと作業のバランスを適宜見直す必要性があったものと考えられる。

第7章 まとめ

本プロジェクトでは、「M2M を用いた大規模データ収集システムの構築に関する研究開発」というテーマのもと、路面の維持管理についての問題を例題として取り上げ、問題を解決するシステムの開発を行うとともに、M2M システム、大規模データシステムを構築する上での課題についての考察を行った。スマートフォンが有するセンサを用いて路面走行時の振動を計測し、その情報をもとに路面の異常箇所を自動で検出するよう設計することで、人間の仲介なく稼働できるシステムを開発した。クライアント・サーバモデルでのシステム開発を通し、クライアント側の負荷、およびサーバの解析精度、データ入出力に関する課題を抽出し、検討を行った。その結果、クライアントのみ、サーバのみといった個々の要素での課題検討では他の要素の負荷を増大させるなどの影響を与えうるため、システム全体を俯瞰した上で最適化を図る必要があることが分かった。

今後の課題としては、本システムをサービスとして実際に運用するにあたっての課題を調査することである。今回のプロジェクトではシステムの負荷バランスに重点を置いて進めてきたが、実際に大規模にデータを収集するためには、サービスとして成立させる必要があると考える。

運用に際してまず挙げられることは、Android アプリケーションの利用者を増加させる仕組みを検討することである。なぜなら、広範囲かつ効率的に路面状況を把握するためには、多くの利用者の存在が不可欠だからである。この仕組みを確立させる一つの方法に、利用者にインセンティブを付与することが挙げられる。例えば、有料道路の管理者が効率的に路面異常を発見するために本システムを利用したい場合を考える。本システムで収集した位置情報をもとにアプリケーション利用者の走行距離を算出し、その距離に応じた割引特典を与えるといったサービスを成立させれば、道路利用者の積極的な情報提供と効率的な道路管理の双方を満たすことができると考えられる。一例として有料道路の管理に対するサービス案を提示したが、実際にどのようなサービスとして成立させるかは、十分検討する必要がある。また、サービスとして成立した際には収集するデータ量が増加することが想定される。このとき開発システムがどのような挙動を示すのか、あるいはデータの増大に対してどう対処し最適化を図るのかについても同時に調査・検討する必要があると考えられる。

もう一つの課題には、情報の取り扱いへの配慮である。本システムでは路面の異常箇所を特定するために位置情報と時間情報を収集している。これらは、場合によっては個人の特定につながりかねない情報であるため、データの送信に懸念を示すユーザーがいることも十分に有り得る。したがって、プライバシーの観点から収集する情報については注意して取り扱う必

要があり，十分な議論を行うべきであると考えられる．

謝辞

本報告書を執筆するまでに、多くの人にご指導、ご協力をいただきました。

株式会社富士通研究所の山下浩一郎様、株式会社 FUJITSU ユニバーシティの川高美由紀様には、テーマの提供、およびプロジェクト全体を通して活発な議論、多くの学ぶ機会を与えてくださいましたことを深く感謝いたします。

本プロジェクトの課題担当教員である天笠俊之准教授には、日頃から多くのご指導をいただきました。プロジェクトを進めていく上で困っている際には相談にも乗っていただき、多くのことを学ばせていただきました。ここで厚く御礼申し上げます。

指導教員である高橋伸准教授には、報告書の執筆や発表などにおいて大変丁寧なご指導とご助言をいただきました。心より感謝申し上げます。また、インタラクティブプログラミング研究室の田中二郎教授、三末和男准教授、志築文太郎准教授、嵯峨智准教授、Shimona Vasilache 助教には、私が研究室に配属されて以来ゼミやミーティングを通して貴重なご意見をいただきました。深く感謝申し上げます。

本プロジェクトのチームメンバである村林竜司氏、林貴哉氏、胥徳文氏にはプロジェクトを遂行するにあたり、様々な面でお世話になりました。ここに感謝の意を表します。

高度 IT 人材育成のための実践的ソフトウェア開発専修プログラムの 6 期生のみなさまには、プロジェクト活動から日常生活まで数多くのご意見やご指摘をいただき、充実した学生生活を送ることが出来ました。ここまで来ることが出来たのは、みなさまから常に刺激を受けることが出来たからだと思っています。この場を借りて心より御礼申し上げます。

最後に、これまで私を支えてくれた家族や友人、大学院生活でお世話になったすべての方々に、心より感謝いたします。本当にありがとうございました。

参考文献

- [1] W. Fan and A. Bifet. Mining big data: current status, and forecast to the future. SIGKDD Explorations Newsletter, 2013, 14.2:1-5.
- [2] D. Howe, M. Costanzo, P. Fey, T. Gojobori, L. Hannick, et al. Big data: The future of biocuration. Nature, 2008, 455.7209: 47-50.
- [3] 喜連川優. ビッグデータの潮流とデータエコシステム. 情報管理, 2013, 55.10: 705-711.
- [4] K. Chang, A. Soong, M. Tseng, and X. Zhixian. Global wireless machine-to-machine standardization. Internet Computing, IEEE, 2011, 15.2: 64-69.
- [5] G. Wu, S. Talwar, K. Johnsson, et al. M2M: From mobile to embedded internet. Communications Magazine, IEEE, 2011, 49.4: 36-43.
- [6] 渡辺 尚. 活用が始まる M2M の現状と展望. IEEJ Transactions on Electronics, Information and Systems, 2012, 132.5: 626-631.
- [7] H. Morikawa. Big data and M2M for Designing a Digital Future. International Smart Infrastructure Symposium, Cambridge, UK, Nov. 2012.
- [8] 株式会社 JR 東日本ウォータービジネス. 夢の飲料自販機 エキナカ本格展開へ ～マーケティング頭脳を搭載した次世代自販機～. <http://www.jre-water.com/pdf/100810jisedai-jihanki.pdf>
- [9] J. Cohen, B. Dolan, M. Dunlap, et al. MAD skills: new analysis practices for big data. Proceedings of the VLDB Endowment, 2009, 2.2: 1481-1492.
- [10] 間瀬憲一, 栗林孝行, 津野昭彦. QoS 統計データを利用した動的サーバ選択法. 電子情報通信学会論文誌. 2003, B, 通信 J86-B.3: 499-510.
- [11] 国土交通省. 国道 (国管理) の維持管理等の現状と課題について. 国道 (国管理) の維持管理等に関する検討会 (第 1 回), 2012. http://www.mlit.go.jp/road/ir/ir-council/road_maintenance/pdf/4.pdf
- [12] 国土交通省. 国道 (国管理) の維持管理等に関する検討会とりまとめ 参考資料. 国道 (国管理) の維持管理等に関する検討会. http://www.mlit.go.jp/road/ir/ir-council/road_maintenance/pdf/report02.pdf

- [13] バンプレコーダー株式会社. BumpRecorder. <http://www.bumprecorder.com/>
- [14] 八木浩一. スマートフォンの加速度センサを用いた路面段差検出手法. 第9回 ITS シンポジウム, 2010.
- [15] 富士通株式会社. 道路パトロール支援サービス: 富士通. <http://jp.fujitsu.com/solutions/cloud/saas/application/road-net/>
- [16] 総務省. スマートフォンアプリケーションに係る利用者の動向. 利用者視点を踏まえた ICT サービスに係る諸問題に関する研究会, スマートフォンを經由した利用者情報の取扱いに関する WG, 2012. http://www.soumu.go.jp/main_content/00150948.pdf
- [17] 村林竜司. M2M を用いた大規模データ収集システムの構築に関する研究開発 -路面段差検出手法の開発と性能評価-. 筑波大学大学院博士課程 システム情報工学研究科特定課題研究報告書, 2014.
- [18] 林 貴哉. M2M を用いた大規模データ収集システムの構築に関する研究開発 -データベースの設計および負荷実験-. 筑波大学大学院博士課程 システム情報工学研究科特定課題研究報告書, 2014.

付録

本プロジェクトにおける成果物として，以下を添付する．

- 例題設定フェーズにおける成果物（付録 A）
 - － 事前調査
 - － 4 象限マトリクス
 - － 例題候補
- 要件定義フェーズにおける成果物（付録 B）
 - － システム構想書
 - － 要件定義書
- 実装フェーズにおける成果物（付録 C）
 - － 基本設計書
- 評価フェーズにおける成果物（付録 D）
 - － 消費電力実験の実験計画書

付録A 例題設定フェーズにおける成果物

A.1 事前調査

1. IaaS, PaaS, SaaS

- <http://itpro.nikkeibp.co.jp/article/Keyword/20110216/357282/>
- IaaS
 - ハード、OSまでを提供
 - 仮想化技術を用いて必要な分だけを提供
 - サービス例
 - Amazon EC2
- PaaS
 - IaaS+ミドルウェアまでを提供
 - サービス例
 - Google App Engine、Microsoft Windows Azure
- SaaS
 - PaaS+アプリケーションまでを提供
 - サービス例
 - CRMとか

2. クラウド型 HEMS

- 家庭内の消費電力や電気料金を蓄積・可視化
 - <http://jpn.nec.com/energy/house/hems/details.html>
 - しかし、現在のHEMSは可視化くらいまでしか？
- 国内普及率
 - 2011年度時点で1%未満、2010年度の普及率は17%（？）
 - <http://www.env.go.jp/council/06earth/y0613-14/mat04-2.pdf>
 - <http://www.meti.go.jp/committee/downloadfiles/g40520a72j.pdf>
 - <http://techon.nikkeibp.co.jp/article/WORD/20130203/263911/>
 - <http://www.kankyo-business.jp/news/003263.php>
 - <http://www.meti.go.jp/committee/downloadfiles/g40520a72j.pdf>

3. クラウド型 CRM

- 顧客管理をクラウドで行う。
- サービス
 - Microsoft Dynamics CRM Online
 - <http://crm.dynamics.com/ja-jp/on-demand>
 - Salesforce
 - http://www.use-ebisu.co.jp/corporate_service/solution/salesforce01.html
 - <http://www.crmstyle.com/>
- 普及率：？
 - <http://japan.internet.com/wmnews/20130311/1.html>

4. クラウド型 ストレージ

- サービス（個人向け）
 - Dropbox
 - SugarSync
 - Googleドライブ
 - iCloud
 - SkyDrive etc...
 - <http://itlifehack.jp/archives/6591480.html>
- 法人向け
 - <http://matome.naver.jp/odai/2134734993004490201>
- 成長状況（？）2011時点
 - <http://japan.zdnet.com/cloud/analysis/35000296/>

4-1. SecureCube Secret Share (NRI セキュア)

“秘密分散技術を活用したセキュアなクラウドストレージサービス”

- 特長
 - ファイルを分割、複数のクラウドDCIに遠隔地保存
 - 重要情報だから漏洩が心配→その情報を重要でなくすればいい！
→発想の転換。
- 導入事例（1例）
 - 松山商工会議所

<http://www.nri-secure.co.jp/service/cube/secretshare.html>

5. クラウド型 電子カルテ

- 診療情報の外部保存の解禁にともなって普及していく？
- サービス
 - <http://www.media-inc.co.jp/product/ecwe/>
- 記事
 - <http://www.itmedia.co.jp/enterprise/articles/1304/09/news012.html>
- 電子カルテ自体の普及率は2割程度（2013年時点）
 - <http://popnori.main.jp/pop/?p=231>

6.クラウド型サーバ

- IaaS
- サービス
 - Amazon EC2
 - さくらのクラウド
 - <http://cloud.sakura.ad.jp/>
- 調査
 - http://download.parallels.com/doc/SMBBBrochure_Japan_localized.pdf

7.クラウド型グループウェア

- サービス例
 - Microsoft Office 365
 - サイボウズLive
- オンプレミス型vsクラウド型・・・出荷本数はオンプレミス型の5倍
 - サイボウズのグループウェア(<http://officetimeline.cybozu.co.jp/?p=1156>)

8.オープンクラウド

- オープンソーススペースのクラウド関連技術
 - <http://enterprisezone.jp/iti/detail/4427>
- オープンPaaS
 - Cloud Foundry, OpenShift
- オープンIaaS
 - OpenStack, CloudStack, Eucalyptus, Wakame
- オープンネットワーク (Software Defined Networking: SDN)
 - OpenFlow
- オープンDC
 - Open Compute Project (FBが中心となって推進)

9.クラウド型シンクライアント

- 仮想デスクトップの利用
- サービス
 - <http://smartdesktop.ize.co.jp/>
- クライアント仮想化ソリューション市場の導入率
 - <http://japan.zdnet.com/virtualization/analysis/35024902/>
 - <http://itpro.nikkeibp.co.jp/article/ActiveSP/20130219/457026/>
 - http://www.hitachi.co.jp/Prod/comp/soft1/spcon/itbnavi_1303/

クラウドサービスの利用状況

- <http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h24/html/nc244110.html>
- <http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h24/summary/summary01.pdf>

スマート〇〇

- **スマートインターチェンジ** 高速道路の本線やサービスエリア、パーキングエリア、バスストップから乗り降りできるように設置されるインターチェンジ
- **スマートエントリ** 鍵なしで施設、駅、自動車などのエンジン始動ができる。セキュリティが課題。
- **スマートカー** ITSやセンサーで車を高度に快適、安全にしたもの。車線逸脱防止、自動ブレーキ、カーナビなど
- **スマートガラス** 電氣的に透明度を切り替えられるガラス製品
- **スマートグリッド** 電力の流れを需要と供給から最適化する。
- **スマートハウス** 使用電力の見える化、電力利用の最適化、地域の電力状況による最適化
- **スマートメーター** 企業や家庭の電力使用量を自動的に検針して従来の検針サービスを提供
- **スマート爆弾** 航空機から投下されたのち、レーザー光線やテレビカメラで目標に誘導される爆弾。

スマートアグリコンソーシアム

- 富士通など「スマートアグリコンソーシアム」設立
ICT活用農業の成功モデル確立へ
- <http://www.computerworld.jp/topics/610/204249>

スマートボード

- <http://www.aec.co.jp/mm/products/smartboard.htm>

スマートスクール

- スマートスクールがつくる次世代地域社会の基盤
- <http://hyocom.jp/blog/blog.php?key=85875>

スマートテレビ

- スマートテレビの定義
<http://ja.wikipedia.org/wiki/%E3%82%B9%E3%83%9E%E3%83%BC%E3%83%88%E3%83%86%E3%83%AC%E3%83%93>

スマートキッチン

- スマートキッチン(ユーザ主導型調理支援システム)
- <http://www.mm.media.kyoto-u.ac.jp/research/skitchen.html>

センサーカメラ

- URL : <http://panasonic.jp/sensor/>
- 人物や動物を検知して、何か変化があったときの動画画像を保存する
 - マイク付のものもあり、音声も記録することが可能
- 利用例
 - 留守番している子どもやペットの様子を観察
 - 防犯

センサーライト

- URL : http://www.ring-g.co.jp/sub_hinmei/0008.htm
- 赤外線等で人物等を感じ、一定時間ライトを点灯させる
- 使用例
 - ・ 玄関の証明
 - ・ トイレや廊下の節電
 - ・ 防犯

3Dセンサー搭載歩数計

- URL : <http://www.tanita.co.jp/products/training/3Dsensor/>
- これまでの歩数計とは違い、全方向に対する運動を感知
 - ・ より正確な歩数カウントが可能
- 使用例
 - ・ ウォーキングやダイエット

ウェアラブルセンサを用いた利用者のTPO推定と
情報提供サービスへの利用

- URL : http://www.hm.indsys.chuo-u.ac.jp/wp-content/uploads/2011/10/JSKE_2008_sudo.pdf
- GPSや生体情報、時刻等を集計し、TPOに合わせた情報を提供
- 例えば...
 - ・ お昼時、空腹(朝食から4時間程度経過)で昨日ラーメンを食べたなら
⇒ 近くのラーメン以外レストラン情報を提供
 - ・ アクティビティが好きな人が観光地にいるなら
⇒ 近くのアクティビティ情報や混雑状況の提供

環境センサーネットワークを活用した「防災テレメータサービス」

- URL : http://www.nttdocomo.co.jp/info/news_release/2012/11/30_01.html
- 自治体向けの災害監視支援サービス
 - ・ 全国各地に設置した環境センサーより気象情報を観測・蓄積
- 使用例
 - ・ 花粉情報の提供
 - ・ ゲリラ豪雨の観測から河川の氾濫や土砂崩れ等の警告

高齢者見守り・安否確認サービス一覧

- URL: <http://www.pref.kanagawa.jp/cnt/f470004/>
- 一人暮らしの高齢者の生活状態の確認
 - ・ ガスや電気の使用状況、安否確認用センサ等より情報を収集
 - ・ 異常がないか監視
- 使用例
 - ・ 遠隔地の高齢の親が連絡がつかず、電気もガスも使用されていない場合、警察や介護サービスに安否確認の依頼
 - ・ 長期にわたって安否確認センサに変化がない場合、異常通知

センサの種類

- URL : <http://www3.jeod.or.jp/niigata/college/kouza/H20/Jul15.pdf>

海 × クラウド

- <http://www.japan-cloud.org/consortium/wg/wg7.html>
- <http://monoist.atmarkit.co.jp/mn/articles/1204/18/news070.html>
 - 漁業関係者と消費者
- 実証研究段階だったり、WGの開催だったりとどまっている？

海 × センサ

- <https://www.semiconportal.com/archive/editorial/technology/design/120215-uksensor.html>
 - 技術記事
- <http://www.ntt.co.jp/journal/0709/files/jn200709052.pdf>
 - 開発段階、普及までには至ってなさそう？
 - 光ファイバセンシングを利用、M2Mっぽい？

空 × 海

- 衛星データの利用
 - 漁場特定・予測
 - マイクロ波センサによる海水温の測定、光学センサの利用
- http://www.mext.go.jp/a_menu/kaihatu/space/application/detail/1291194.htm
 - 漁業関係者には配信しているもよう

スポーツ × クラウド

- スポーツビデオカメラ
 - http://k-tai.impress.co.jp/docs/news/20120322_520602.html
- 世界的な販売台数は増加していくと予測されている
 - http://av.watch.impress.co.jp/docs/news/20120322_520614.html
 - <http://matome.naver.jp/odai/2134676127352749601>

スポーツ × スマート

- au Smart Sports Run&Walk
 - ケータイが専属トレーナーに
 - SNS連携つき
 - <http://www.au.kddi.com/mobile/service/featurephone/life-support/run-walk/>
- 会員は300万人超(2011)
 - http://www.kddi.com/corporate/news_release/2011/1102/
 - 2013/4 時点でのau契約数は約3800万人
 - <http://www.kddi.com/corporate/ir/finance/keiyakusu/>

スポーツ × センサ

- Nike+
- Adidas micoach
 - センサ利用数はいずれもわからず・・・
 - 実用化はされている

ライフログ

- サービスとして実現している

デジタルサイネージ

- 鉄道、駅、空港などの交通機関、商業施設、屋外、銀行、オフィスなど多岐にわたる
 - <http://www.yanoict.com/yzsurvey/2012/2.html>

次世代型自販機

- マーケティング自販機
 - <http://www.logi-solu.co.jp/article/14543461.html>
 - JR東 acure
 - 売上が確か上がった・・・はず
 - 設置台数は増えている

ホテル×クラウド

- 製品としてはある(富士通、NECなど)
 - NEC: 基幹業務、デジタルサイネージ、IP電話をサポート
 - http://jpn.nec.com/hotel/panel_director/index.html?
 - 富士通: 基幹業務が主?
 - http://glovvia.fujitsu.com/glovvia_smart/hotel/cloud/

- クラウド利用率は増加している
 - <http://bizmakoto.jp/makoto/articles/1302/04/news051.html>

SmarterPlanet (IBM)

- <http://www-06.ibm.com/innovation/jp/smarterplanet/>

軍事

- [C4I](#) 近くの戦車が互いに情報をやりとりしたり、基幹連隊指揮統制システムと接続したりして、敵や味方に関する情報を共有するほか、指揮統制も可能になる。敵味方の位置、各車両の残弾数、残燃料などの状態が一目で分かり、戦車部隊と普通科部隊が一体化した作戦行動が取れるようになることを目的としている。
- [Nett Warrior](#) 映像を戦場ネットワークを通じて各級司令部が入手して状況判断に役立てたり、敵味方の位置・状況、地図、個人の生理的状态、携行火器の残弾数などが広く共有されるようになる。これによって、指揮官だけでなく兵士にとっても、主体的に適切な判断と行動をとる手助けとなる。敵味方の判断。

交通

スマートIC
スマートエントリ
スマートカー
カーシェアリング予約

レンタカー

乗り換え案内
渋滞予測

インテリジェンスコントロールディスプレイ
スポット通信サービス
ETC

自動走行(支援)

生活

スマートガラス
スマートグリッド
スマートハウス
スマートメーター
スマートTV
スマートキッチン
クラウド型HEMS
ウェアラブルセンサ・TPO推定
高齢者見守り・安否確認

通信

スマートフォン

経営

クラウド型CRM
クラウド型グループウェア

農業

スマートアグリコンソーシアム
水産クラウド
衛星データ・センサで漁場予測

ITインフラ

クラウド型ストレージ
クラウド型サーバー
オープンクラウド

医療

クラウド型電子カルテ
3Dセンサ歩数計
ウェアラブル生体情報監視
生体内外無線通信
バイタルケアネットワーク
巡回診療支援システム
ICタグによるSCM
健康機器センサで健康管理

ライフログ

食べ物ログ
睡眠ログ
衣服ログ
読書メーター

宿泊

ホテル総合クラウドサービス(NEC,富士通)

娯楽

カラオケSNS
生体センサで楽器

軍事

スマート爆弾
C4I
Nett Warrior

防犯・防災

センサーライト
センサーカメラ
津波予測

スポーツ

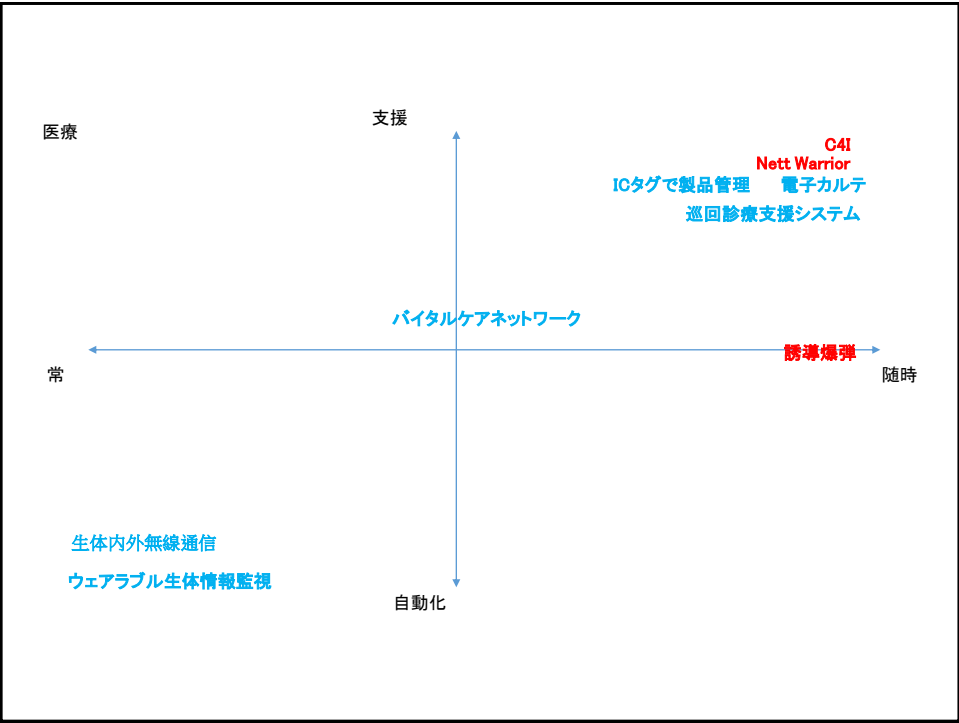
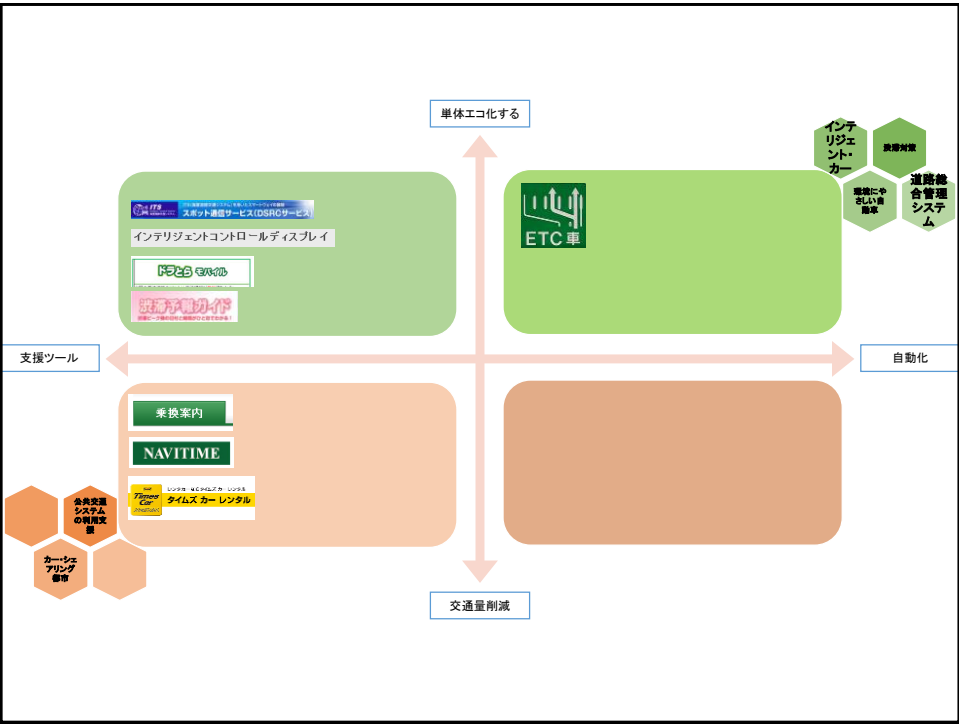
スポーツビデオカメラ
Au SmartSports
Nike+, adidas microach
スノーボード自動コーチング

?

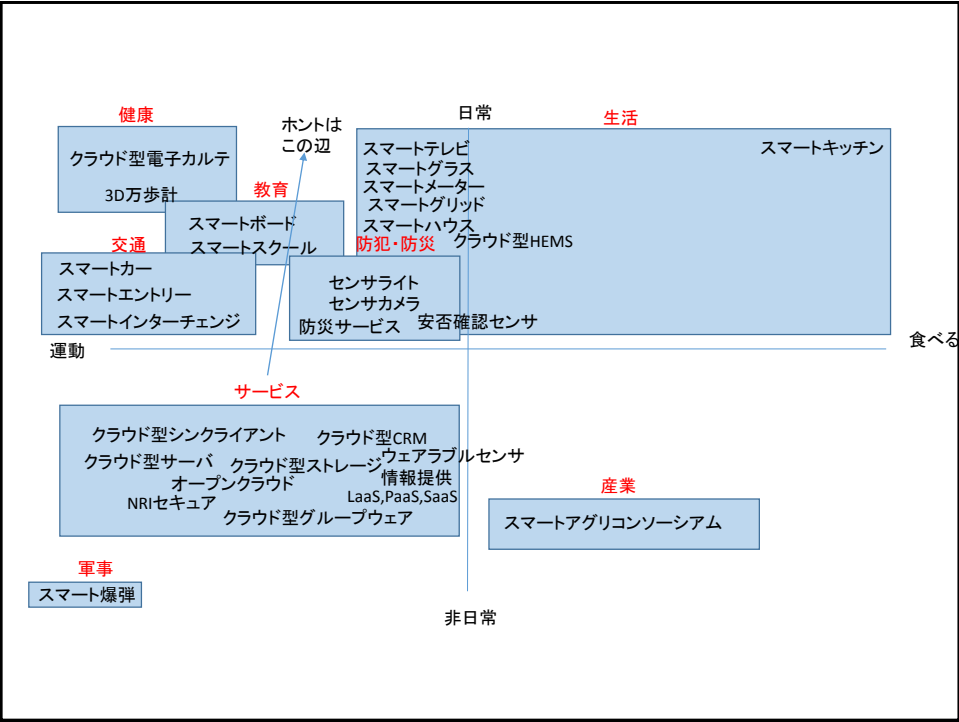
デジタルサイネージ
サジェスト型(?)自販機

A.2 4象限マトリクス

4象限マトリクス



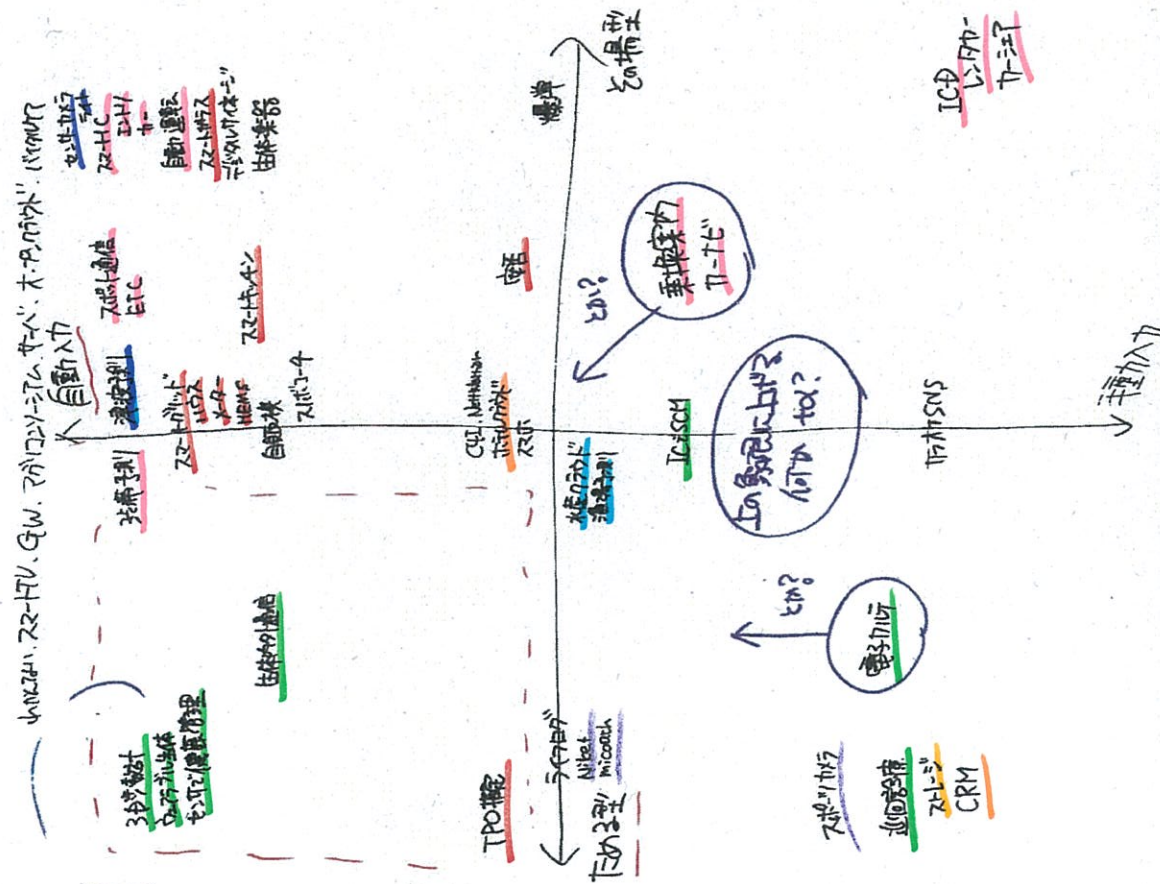
4象限マトリクス



德

2013/5/31 16:24

（車）人が情報を発信する？（会社、LAN、WWW、FAX、郵便局）



A.3 例題候補

めんどうなこと、困ること	解決策
電話(決まった場所ではしか使えない)	無線(どこでも、携帯、子機)
自動車のドア、窓(荷物持つと開けづらい)	パワードア、ウィンドウ
カメラ(撮った写真をその場でチェックできない)	デジタルカメラ
お金の引き出し(銀行じゃないとできない)	ATM(銀行以外でも)
バスがいつ来るかわからない(遅れる)	電車みたいどこにいたかわかれば楽
冬歩くのが寒い、大変	地下道
写真撮るとき表情がわるい	笑った瞬間写真撮る技術
著作権侵害	電子透かし
音楽ファイルの曲がつながっている	無音部分で自動カット
ビデオ返却めんどくさい	ポストで返却
注射痛い	痛くない注射針
この音楽なんやったっけ？	音を解析して曲名出してくれる
あの芸能人の名前忘れた！	はい、いいえで名前あてる
時計の時刻ずれる	電波時計
地震が	地震カウントダウン
URL入力するの大変	QRコード
バック駐車がうまくいかない	センサ、カメラ、補助線の導入

めんどうなこと、困ること	解決策
前との車間距離がわからない	センサ
右利きように作られたインターフェースが多い	？
バスの料金支払い制度	Suicaなどの導入
書類の持ち運びが不便	電子化
人身事故、人立入	高架化、地下化
病気になる	遺伝子検査
新生児医療	測定データをリアルタイムに把握・分析・活用
省エネへのモチベーション維持	HEMSによる「見える化」
若者が駅にたむろする	モスキート音で撃退
人に見られたくない、盗聴されたくない	VPN技術、暗号化
エレベーターが気に食わない	エレベーター同士が相互に通信しあう
電子マネーの扱い方がコンビニによってまちまち	支払い方法の統一
トイレの洗面台を化粧する人が占領する	洗面台と化粧台の分離(問題点の切り離し？)
花粉症とかハウスダストとか	空気清浄機、マスク(フィルタリング)

<起承転結案 6/10>

家+診察⇒故障しそうな箇所を検知、連絡

壁や天井に何かくっつけたら面白い？

警察・裁判+3D⇒逆転裁判的な

生活をセンシング+寿命⇒寿命診断+健康管理

家+見えない⇒ハウスダスト等の空気環境の改善

これまでのお金の使い方や生活リズム等の情報からこれからどれだけ何にお金がかかるか判断

病院+どこでも⇒健康診断を日常のデータからできないか

事前検知+消防⇒火事発生場所の事前特定により通報の短縮

エコ+大量消費⇒使用状況を検知し、その人の必要/不必要の通知（カメラとかで画像認識して？）

蛇口+凍結⇒凍結しそうな温度や箇所を検知し、自動で凍結しないように水流調整

観光+表情検出⇒幸せそうな人が多い場所がわかる

ティッシュ+表情検出⇒くしゃみしそうなひとにティッシュを差し出す

出店+スマホ⇒どこから来る人が多いか地図上に可視化して店舗配置に役立てる

テーマパーク+笑顔検出⇒どのアトラクションがウケてるかわかる！

表情が悪い+笑顔の時撮影⇒表情を分析して気分予測、その場で気分転換する案を提供

カメラ+デジカメ⇒見えないものを見えるように、子供が絵を書く教育の支援システム

芸能人の名前を知る+2択⇒勉強したものを忘れたときに、記憶支援ツールを提供

付録B 要件定義フェーズにおける成果物

B.1 システム構想書

システム構想書

1.0 版

チームシュークリーム
枝松ちさと
胥徳文
林貴哉
村林竜司

改訂履歴

[illegible]

目次

1. はじめに.....	3
1.1 本書の目的	3
1.2 構成	3
2. 背景.....	4
2.1 プロジェクト背景	4
2.2 テーマ背景	4
3. 現状と問題.....	5
3.1 現状の路面監視.....	5
3.2 既存の路面異常検出システム	5
3.3 路面監視と既存の路面異常検出システムの問題点	5
4. 解決策の提案	6
5. 開発システム	7
5.1 システム概要	7
5.2 システムの対象.....	7
5.3 システム構成	8
5.4 システム評価実験	9
5.5 システム完成要件	9
6. 実現不確定事項	10
7. 開発スケジュール	11
A. 参考文献	12

1. はじめに

1.1 本書の目的

本書は顧客、開発チームに対してチームシュークリームが開発するシステムの構想を明確にしたものである。

1.2 構成

本書の構成は以下の通りである。

- 1. はじめに
本書の目的や構成について述べる。
- 2. 背景
本プロジェクト発足の背景やプロジェクトで焦点をあてる路面監視における背景について述べる。
- 3. 現状と問題
路面監視や現在公開されているシステムの現状を分析し、それらが抱える問題について述べる。
- 4. 解決策の提案
3章の問題を踏まえて、本プロジェクトにおける解決策について述べる。
- 5. 開発システム
本プロジェクトで開発するシステムの概要や評価方法などについて述べる。
- 6. 実現不確定事項
研究開発プロジェクトにおいて、本チームが実現できるか確定していない事柄について述べる。
- 7. 開発スケジュール
本プロジェクトの開発スケジュールについて述べる。

2. 背景

2.1 プロジェクト背景

現在ビッグデータやクラウドといった技術に注目が集まっている。しかし、これらの名称には明確な定義がなく、バズワードとして認識されている。そこで株式会社富士通研究所様より、こういったバズワードの定義を明確にするためにビッグデータやクラウドを用いたシステム開発を研究開発プロジェクトで行ってほしいと依頼を受けた。

本プロジェクトは、マシンが自律的に稼働すること、および大量の情報を処理することを前提とする。この前提のもとにシステム開発の詳細なテーマを設定する。

テーマ設定は、日常の問題をピックアップし、既存の技術等と関連付けて解決策を考案していくなどブレインストーミングをメインに進めていった。

2.2 テーマ背景

道路の段差や穴、落下物等の路面異常は交通事故の要因となる場合がある（A. 参考文献 ※1）。未然に事故を防ぐために、道路の維持管理をしている組織などは定期的に路面監視を行い、事故の要因となりうるものを迅速に取り除くなどの対策を講じている。しかし、平成 22 年から巡回頻度の基準は緩和されているため、網羅性やリアルタイム性も低下していると考えられる。交通量が多いほど路面損傷の発見数が多いという現状を踏まえ、維持管理を行う組織は交通量に応じて巡回頻度を上げるという対策を行うことで、効果的な路面監視を目指している。

効率的な路面監視を行うために巡回支援を目的とした製品もある。例えば、巡回を行う車にスマートフォンを設置し、スマートフォンの加速度センサで道路の凹凸情報を収集し、その情報から道路の補修箇所を地図上に表示する製品が富士通株式会社からリリースされている（※2）。他にもスマートフォンを用いて路面状況を推定する研究も行われている（※3）。これらは特殊な装置を用いないため、導入が容易である。しかし、巡回の際に路面異常を発見しやすくなるものの、巡回地域の拡大や巡回頻度の増加にはつながらないため、網羅性やリアルタイム性の向上にはつながらない。また、巡回車両の特性による差異を考慮してキャリブレーション処理を行わなければならない。さらにその設定は人が行わなければならない。これらのシステムは、共通して路面異常を検出するために走行が必要であるため、現行の巡回と変わらず、コスト削減にはつながらない。

本プロジェクトでは、巡回時の支援ではなく、巡回の前段階の支援を考える。多数の一般の道路利用者から走行情報を収集して路面監視に利用できれば、路面監視のための巡回の前に、あらかじめ路面異常箇所を知ることができる。また、大量に走行情報を収集できれば、網羅性やリアルタイム性が向上するだけでなく、統計処理等を行うことによって個々のキャリブレーションが不要となる。

本プロジェクトでは路面異常検出を行うシステムを開発する。開発システムでは路面異常検出のために一般の道路利用者から走行情報を収集する。一般の道路利用者から走行情報を収集するため、走行情報の測定や送信には一般に普及しているスマートフォンが適していると考えた。これにより、走行情報は広範囲にわたって常時収集できるようになり、網羅性やリアルタイム性の高い路面異常検出が可能となる。また、多くの走行情報を解析するため、車両による差異やスマートフォンの設置位置に関する差異のためのキャリブレーションを必要としない。検出した路面異常は地図に付加し、ネットワークを介して提示することで、どこでも容易に路面異常箇所を閲覧できるようにする。

3. 現状と問題

3.1 現状の路面監視

現在、路面監視は対象道路の維持管理を行っている組織や人が目視で行っており、その場で対処できる落下物等の路面異常は即時対応している。近年、路面異常の発見件数は増加傾向にあるため、交通量の多い道路では巡回頻度を他の道路より多く設定するなどの対策を行っている。

3.2 既存の路面異常検出システム

現在、路面異常を検出するシステムやサービスがいくつか公開されている。例えば、富士通株式会社でリリースされたサービス（A. 参考文献 ※2）や路面段差を検出するスマートフォン向けアプリケーション（A. 参考文献 ※3）がある。

富士通株式会社でリリースされた“道路パトロール支援サービス”は巡回車両にスマートフォンを設置し、加速度センサを用いて走行時の凹凸情報を収集する。そして、収集した情報を解析し、補修が必要な箇所の特定を行う。

路面段差を検出するスマートフォン向けアプリケーションは、スマートフォンセンサから取得した情報を解析し、端末に表示する。また、走行情報は共有サーバにアップロードすることができ、web サイトにアクセスすることで共有されている走行情報を閲覧できる。計測・解析の前処理には、スマートフォンの特性を計測するため、測定開始から 1km 程度キャリブレーションを行う必要がある。さらに、車内に設置する際には路面に対して垂直、または水平にスマートフォンを固定することが推奨されている。

3.3 路面監視と既存の路面異常検出システムの問題点

現在路面監視は平成 21 年以前に比べて巡回頻度が緩和されており、作業量が減少しているために網羅性やリアルタイム性が低下している。作業員増員などのリソースを追加することである程度解決できるが、現状でそれは困難である。国内だけでも膨大な数の道路があるため、それをすべて目視で監視するにはかなりのコストがかかることが原因として挙げられる。

既存の路面異常検出システムは、コスト削減につながるように思える。しかし、路面状況の解析を 1 件の走行情報ごとに行っているため、走行中のスマートフォンの操作や子どもの飛び出しなどの突発的な事象を取り除くことはできない。また、路面監視のために巡回しなくてはならず、巡回にかかるコストはさほど削減されない。さらにキャリブレーションが必要なため、人の作業が増えてしまうこともコスト削減の観点から問題と言える。

表 1 に問題点をまとめたものを示す。

表 1：問題点

路面監視の問題点	既存システムの問題点
問題 1：網羅性・リアルタイム性が低い 問題 2：すべて目視で行っている	問題 3：突発的な異常を取り除けない 問題 4：コストを削減できていない 問題 5：キャリブレーション操作が必要

4. 解決策の提案

本章では、3.3 路面監視と既存の路面異常検出システムの問題点 3.3 節で示した問題についてそれぞれ解決策を述べる。

問題 1：網羅性・リアルタイム性が低い

路面異常検出に用いる走行情報を一般の道路利用者から収集することで、解決を図る。各地の道路について、広範囲にわたって様々な時間の情報を蓄積できるようになるため、その情報を解析して路面異常検出を行えば、網羅性・リアルタイム性の高い検出結果が望める。

問題 2：すべて目視で行っている

路面異常検出結果を地図に付加して表示することで、解決を図る。巡回の前に地図を閲覧することで、あらかじめ路面異常箇所を知ることができる。事前に路面異常箇所を知ることができれば、提示された地点を確認するだけでよいため、効率的な路面監視が行える。

問題 3：突発的な異常を取り除けない

問題 4：あまりコストを削減できていない

一般の運転者から走行情報を収集するので、大量の走行情報が蓄積できるようになる。この大量のデータを用いることで、解決を図る。統計的な処理を行うことにより突発的な事象は取り除くことができる。

問題 5：キャリブレーション操作が必要

問題 3 の解決策と同様、大量の走行情報が蓄積されているため、統計的な処理を行うことで路面異常以外の外れ値を取り除くことができる。

5. 開発システム

5.1 システム概要

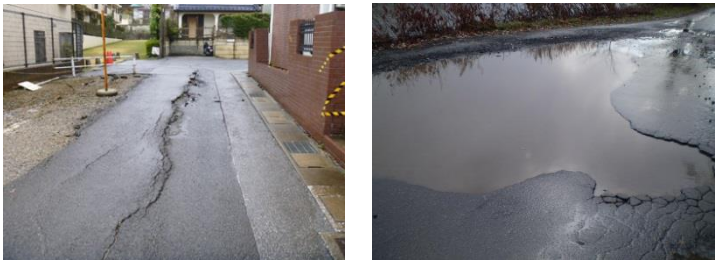
本プロジェクトでは、路面異常を検出するシステムを開発する。検出結果は地図に付加することで路面状態マップとして提示する。走行情報の測定には一般に普及しているスマートフォンを用いる。一般の道路利用者が走行情報を収集する際の操作は、アプリケーションの起動と終了以外不要とする。スマートフォンでセンシングした走行情報はサーバへ送信する。サーバでは走行情報の解析・統計処理を行い、結果を地図に重ね合わせて路面状態マップとして提供する。走行情報提供者である一般の道路利用者や閲覧者はそのマップから路面異常箇所を判断できる。路面異常を付加したマップの閲覧による効果として、路面監視の効率化が期待できる。

5.2 システムの対象

- 検出対象

本システムの検出対象は路面の異常である。路面異常とは路面の損傷による凹凸やひび、段差、轍等を指す。例を表2に示す。本プロジェクトで扱う路面異常は表2の区分2と3である。

表2：路面異常

区分	状態	内容
1	走行不可能・困難	道路の全面的な損壊、一時的な通行止め、冠水、落下物 
2	機能障害	ひび割れや轍による段差、道路勾配の不良（水たまり） 
3	快適性の低下	補修による段差、ハンプ 

● 対象ユーザ

本システムの利用者は走行情報を提供する一般の道路利用者と路面状態マップを閲覧者である。走行情報の収集にスマートフォンを用いることから走行情報提供者はスマートフォンを所有している一般の運転者とする。路面状態マップ閲覧者は道路の維持管理を行う組織/人、走行情報提供者を想定している。

表 3 に本システムの対象をまとめる。

表 3：システムの対象

	対象		詳細
検出	路面異常		表 2 の区分 2 と 3
システム利用者	システム利用者	走行情報提供者	スマートフォンを所有している一般の運転者
		路面状態マップ閲覧者	道路管理を行う人/組織、走行情報提供者

5.3 システム構成

本システムは走行情報の収集・送信を行うスマートフォン、処理や情報の蓄積等を行うサーバ、閲覧を行うクライアントPCから構成されている。また利用者は走行情報提供者と路面状態マップの閲覧者である。以下にその簡略図を示す。

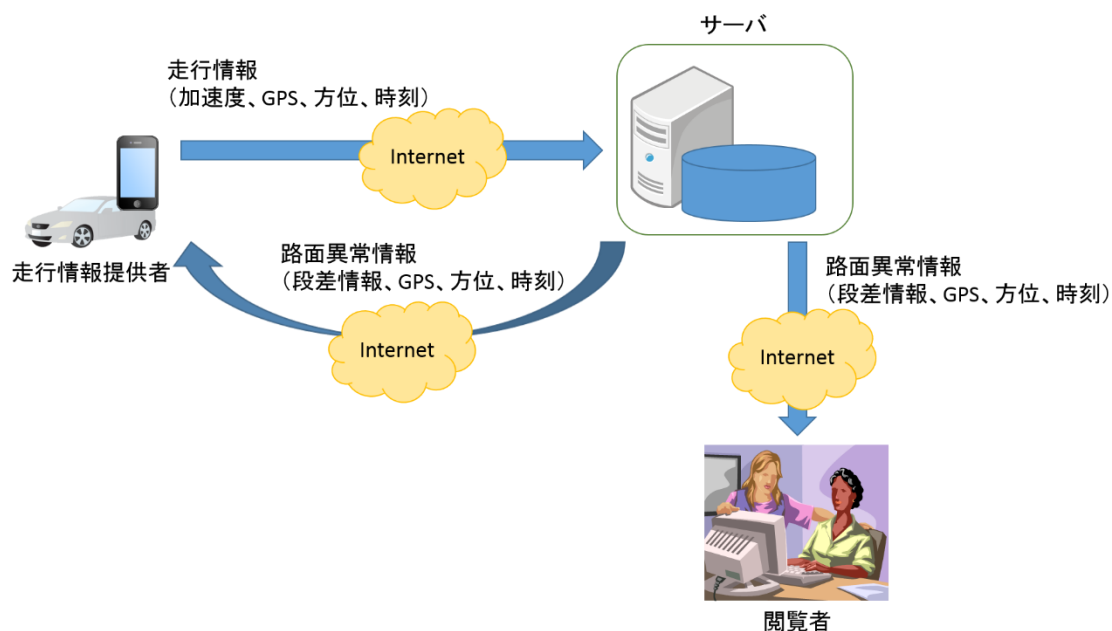


図 1：システム構成図

5.4 システム評価実験

開発システムの評価実験は筑波大学内の道路で行う。走行情報を収集し、路面異常を正しく検出できているか、目視に比べて検出精度やコストにどれだけに差異があるのかを評価する予定である。

5.5 システム完成要件

本システムの完成は、ユーザ数 20 人から集めた走行情報から路面異常を正しく検出でき、それを正しく地図に表示できることが確認できた時点とする。

6. 実現不確定事項

ここでは、研究開発の期間内で実現できるか不明なものを記述する。

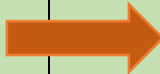
以下に、実現不確定事項をまとめる。

No	実現不確定事項	詳細	現状
01	利用者増加の仕組み	利用者が増えるためのインセンティブ	未だ考えられていない
02	データ自動増加の仕組み	走行情報を自動的に増加させていく仕組み	取り組みそうにない
03	GPS 取得タイミング	省電力のために GPS の取得間隔を空けることを考える	取り組みそうにない
04	端末間通信	GPS 取得やデータ送受信に端末間通信を用いて、より省電力化を試みる	取り組みそうにない
05	省電力化	スマートフォンにもたらず電力影響を極力削減する	簡単な省電力化は試みる

7. 開発スケジュール

以下に開発スケジュールを示す。本プロジェクトでは、2013 年 12 月中にシステム完成を目指す。9 月中旬から 10 月の中旬は、後述の報告会のため開発作業を行わない。第 1 イテレーションでは異常検出以外の機能、第 2 イテレーションでは異常検出機能を開発する。

また、10 月や 2 月は研究開発プロジェクトにおける中間報告会があるため、その 2,3 週間前には資料の準備等を開始する。

		7 月	8 月	9 月	10 月	11 月	12 月	1 月	2 月
要件定義									
開発	第 1 イテレーション								
	第 2 イテレーション								
システム評価									
報告会（準備含む）									

A. 参考文献

※1 : http://www.mlit.go.jp/road/ir/ir-council/road_maintenance/pdf/4.pdf

※2 : <http://pr.fujitsu.com/jp/news/2013/06/25.html>

※3 : <http://itpro.nikkeibp.co.jp/article/COLUMN/20111109/373621/>

B.2 要件定義書

要件定義書

1.0 版

チームシュークリーム
枝松ちさと
胥徳文
林貴哉
村林竜司

改訂履歴

[illegible]

目次

1	はじめに	4
1.1	本書の目的	4
1.2	本書の構成	4
1.3	用語定義	5
2	システム概要	6
2.1	システムについて	6
2.2	システム全体構成	6
2.3	本システムでの処理の流れ	7
2.3.1	走行情報の収集・解析を行う処理の流れ	7
2.3.2	路面状態マップを閲覧する際の処理の流れ	8
3	機能要件	9
3.1	ユースケース図およびユースケース記述	9
3.2	本システムで実装する機能	16
3.2.1	走行情報収集機能	16
3.2.2	データ送信機能	16
3.2.3	路面状態マップ閲覧機能（スマートフォンサイド）	17
3.2.4	路面状態マップ閲覧機能（クライアント PC サイド）	17
3.2.5	データ受信機能	17
3.2.6	データ解析機能	17
3.2.7	路面状態マップ生成機能	18
3.2.8	データ送信機能	18
3.2.9	データ管理機能	18
3.2.10	異常推定箇所検出機能	18
4	画面イメージ	19
4.1	アプリケーションから路面状態マップを閲覧する場合の画面イメージ	19
4.2	クライアント PC から異常状態マップを閲覧する場合の画面イメージ	20
5	データベース仕様	21
6	システム仕様	22
6.1	スマートフォンサイドのシステム仕様	22
6.2	クライアント PC のシステム仕様	23
6.3	サーバサイドのシステム仕様	24
7	未確定事項	25
7.1	検討事項	25

表目次

表 1：用語定義	5
----------------	---

図目次

図 1：システム全体構成図	6
図 2：走行情報の収集・解析を行う処理の流れ	7
図 3：路面状態マップを閲覧する際の処理の流れ.....	8
図 4：ユースケース一覧.....	9
図 5：データベースに格納するエンティティ	21

1 はじめに

1.1 本書の目的

本書は、顧客および開発チームに対して、本プロジェクトで開発するシステムの概要や機能、外部仕様を明確化することを目的としている。

1.2 本書の構成

本書は以下の構成となっている。

- 1 はじめに
本書の目的や、本書を読み進めるうえで必要な用語定義などについて述べる。
- 2 システム概要
本プロジェクトで開発するシステムの概要について述べる。
- 3 機能要件
開発するシステムに実装する要件について述べる。
- 4 画面イメージ
開発するシステムにおいて、画面出力のモックアップを示す。
- 5 データベース仕様
開発するシステムにおいて、システムに記録するデータの内訳について示す。
- 6 システム仕様
開発するシステムにおける仕様について述べる。
- 7 未確定事項
本書作成時に確定しなかった事項、および確定時期について述べる。

1.3 用語定義

本書で使用する用語は表 1 に示す定義に基づく。

表 1：用語定義

ID	用語	定義
WD-01	路面	主に車道の路面を指す。歩道は含まない
WD-02	センサ情報	スマートフォンのセンサから得られる加速度や GPS などの情報
WD-03	走行情報	時刻、センサ情報の集合
WD-04	路面異常	路面の損傷による段差のような、快適な運転走行を妨げる段差などを指す
WD-05	路面異常情報	路面異常の緯度・経度などの情報
WD-06	路面状態マップ	路面異常情報を付加した地図
WD-07	道路管理者	路面の維持管理を行っている人・組織
WD-08	走行情報提供者	走行情報をシステムに提供する人
WD-09	路面状態マップ閲覧者	走行情報提供者と道路管理者
WD-10	路面監視	路面異常がないか巡回して行う監視

2 システム概要

2.1 システムについて

本システムは路面異常を検出し、その異常箇所を地図に付加して提示する。異常の検出には走行情報を用い、この情報の収集にはスマートフォンのセンサ情報を用いる。走行情報収集の際に操作が多いと走行情報提供者の負担となってしまう。そこで、本システムでは走行情報提供者の操作をアプリケーションの起動・終了に限定する。多数の走行情報提供者より収集した大量の走行情報をサーバで解析・統計処理を行うことにより、スマートフォンごと、車ごとのキャリブレーションを必要としない路面異常検出を実現する。

2.2 システム全体構成

本プロジェクトで開発するシステムの簡略な全体構成図を図 1 に示す。

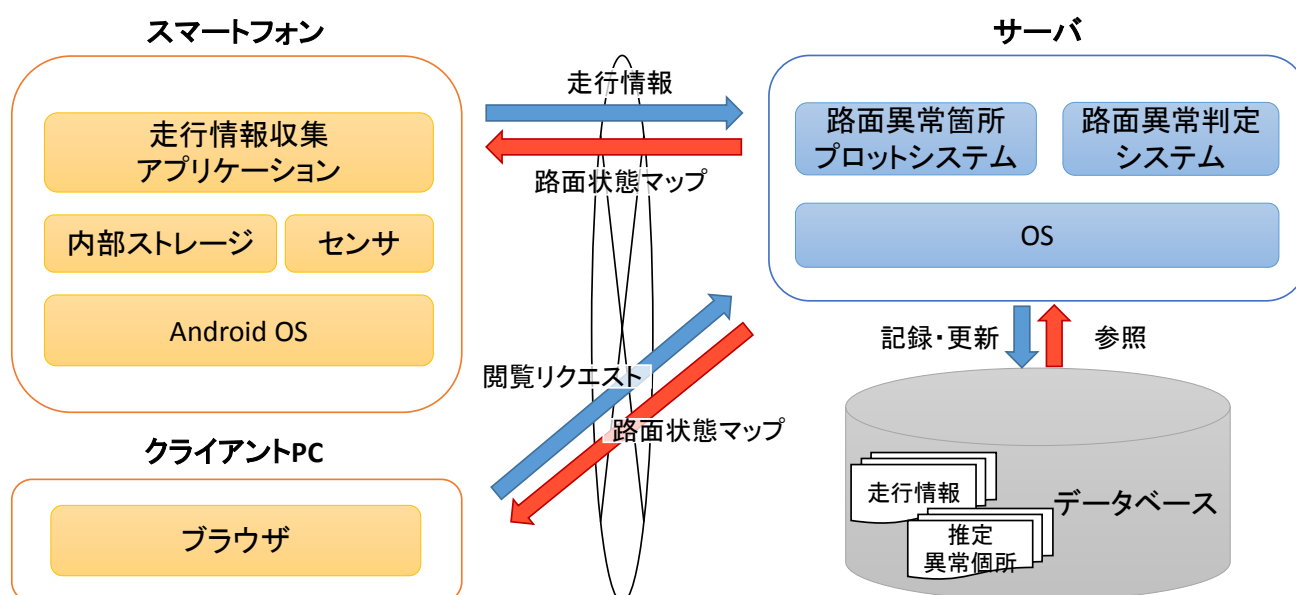


図 1：システム全体構成図

2.3 本システムでの処理の流れ

2.3.1 走行情報の収集・解析を行う処理の流れ

図 2 に、走行情報を収集・解析するまでの大まかな処理を示す。

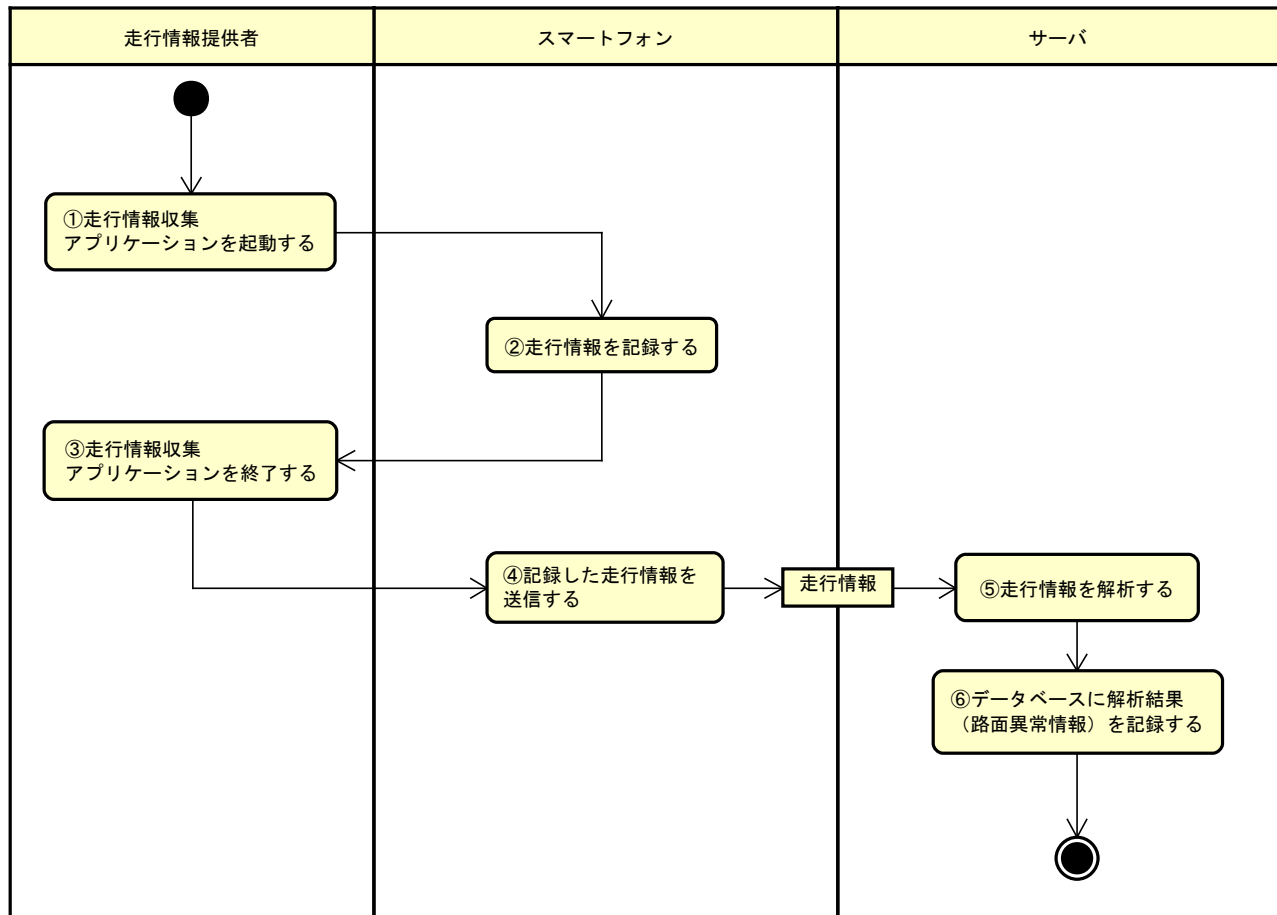


図 2：走行情報の収集・解析を行う処理の流れ

- ① 走行情報提供者は、スマートフォンにインストールされている走行情報収集アプリケーションを起動し、自動車で路面を走行する。
- ② スマートフォンアプリケーションは、スマートフォンに搭載されているセンサを用いて、走行時の振動や位置などの走行情報を取得し、記録する。
- ③ 走行を終えたら、走行情報提供者は走行情報収集アプリケーションを終了する。
- ④ スマートフォンアプリケーションは、終了時に走行情報データをサーバに送信する。
- ⑤ サーバは、スマートフォンから送られてきた走行情報を解析し、路面異常の検出を行う。
検出した異常は、路面異常情報としてデータベースに記録する。

2.3.2 路面状態マップを閲覧する際の処理の流れ

図 3 に、路面状態マップを閲覧する際の大まかな処理を示す。

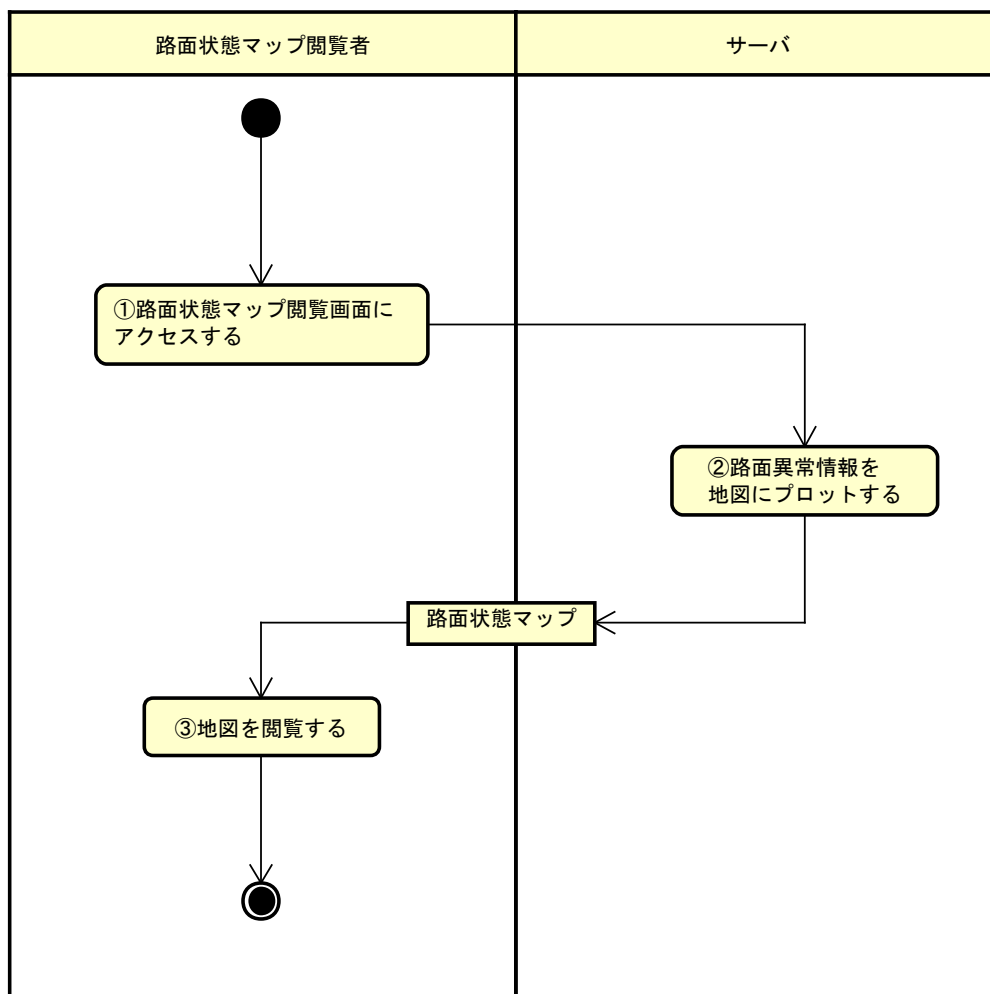


図 3：路面状態マップを閲覧する際の処理の流れ

路面状態マップの閲覧は、クライアント PC 上の Web ブラウザから閲覧する方法と、スマートフォンのアプリケーションから閲覧する方法の 2 種類を想定している。

- ① 路面状態マップ閲覧者は、クライアント PC の Web ブラウザ（またはスマートフォンアプリケーション）を通じて、路面状態マップ閲覧画面にアクセスする。
- ② リクエストを受け取ったサーバは要求されている範囲に対して、記録されている路面異常情報を地図にプロットしたうえで、これをクライアント PC（またはスマートフォン）に返す。
- ③ クライアント PC の Web ブラウザ上（またはスマートフォンアプリケーション上）に路面状態マップが表示され、路面状態マップ閲覧者はそれを閲覧する。

詳細な処理機能などについては、3.2 節で述べる。

3 機能要件

3.1 ユースケース図およびユースケース記述

本システムにおけるユースケース一覧を図 4 に示す。

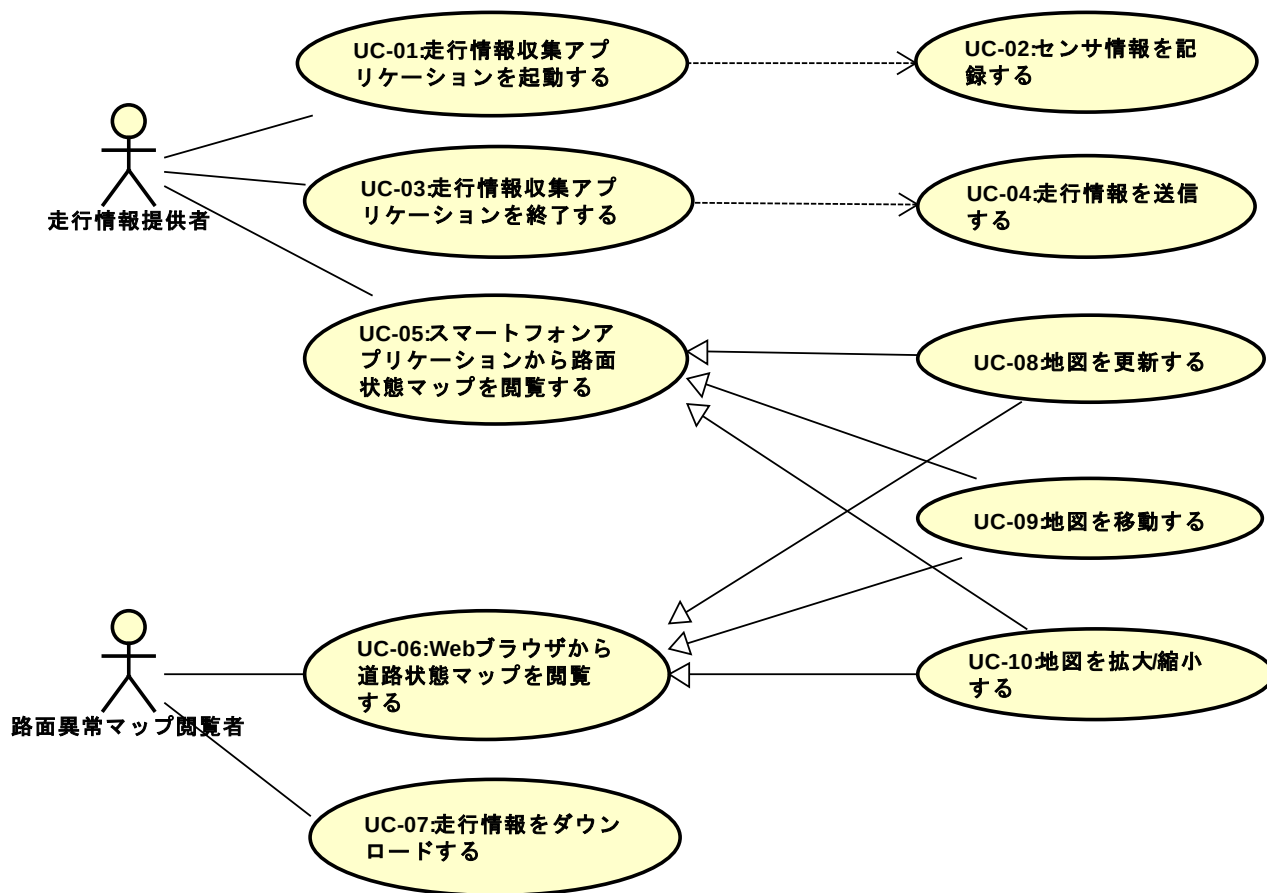


図 4：ユースケース一覧

図 4 で示したユースケースについて、それぞれ詳細を記述したユースケース記述を次ページより示す。

ユースケース ID	UC-01
ユースケース名	走行情報収集アプリケーションを起動する
概要	走行情報収集アプリケーションを起動し、センサ情報の取得を開始する
事前条件	
基本系列	1. 走行情報提供者はスマートフォンの走行情報収集アプリケーションをタップする 2. アプリケーションはセンサ情報の取得を開始する
代替系列	2a. 基本系列 2 において、センサ情報の取得が開始できない場合、 1. アプリケーションはセンサ情報の取得が開始できない旨をメッセージ画面に表示する 2. 走行情報提供者はメッセージ画面の「終了」ボタンを押下する 3. アプリケーションを終了し、本ユースケースを終了する
事後条件	スマートフォンのセンサを利用してセンサ情報の取得が開始できていること
備考	起動後はバックグラウンドで動作する

ユースケース ID	UC-02
ユースケース名	センサ情報を記録する
概要	スマートフォンから取得したセンサ情報を記録する
事前条件	・ 走行情報収集アプリケーションが起動していること ・ スマートフォンのセンサを利用してセンサ情報の取得ができていること
基本系列	1. 走行情報提供者はスマートフォンを車内に置いた状態で走行する 2. アプリケーションは走行中のセンサ情報を取得し、スマートフォンに走行情報として記録する
代替系列	2a. 基本系列 2 において、センサ情報をスマートフォンに記録ができない場合、 1. アプリケーションは記録不可・センサ情報を破棄する旨をメッセージ画面に表示する 2. 走行情報提供者はメッセージ画面の「OK」ボタンを押下する 3. アプリケーションはセンサ情報を破棄し、スマートフォンに走行情報を記録せずに終了し、本ユースケースを終了する
事後条件	走行情報が記録されたファイルがスマートフォンに作成されていること
備考	UC-01 と依存関係にある

ユースケース ID	UC-03
ユースケース名	走行情報収集アプリケーションを終了する
概要	走行情報収集アプリケーションを終了する スマートフォンに記録した走行情報をサーバに送信する
事前条件	・アプリケーションが起動していること ・走行情報がスマートフォンに記録されていること ・路面走行が終了していること
基本系列	1. 走行情報提供者はアプリケーション画面より終了ボタンをタップする 2. アプリケーションはセンサ情報の取得、および記録を停止する 3. アプリケーションは記録した走行情報をサーバに送信する 4. 送信が完了したら、アプリケーションを終了する
代替系列	2a. 基本系列 2 において、走行情報がサーバに送信できない場合、 1. 一定時間経過後、再度サーバに走行情報を送信する 2. 基本系列 4 へ進む
事後条件	・アプリケーションが終了していること ・走行情報がスマートフォンから削除されていること
備考	

ユースケース ID	UC-04
ユースケース名	走行情報を送信する
概要	アプリケーション終了時にシステムに走行情報を送信する
事前条件	
基本系列	1. 走行情報提供者は走行情報収集アプリケーションを終了する 2. アプリケーションは記録した走行情報をサーバに送信する 3. スマートフォンから走行情報を削除する
代替系列	2a. 基本系列 2 において、走行情報の送信に失敗した場合、 1. 一定時間経過後、再度サーバに走行情報を送信する 2. 基本系列 3 へ進む
事後条件	・走行情報がスマートフォンから削除されていること ・サーバに走行情報が記録されていること
備考	UC-03 と依存関係にある

ユースケース ID	UC-05
ユースケース名	スマートフォンアプリケーションから路面状態マップを閲覧する
概要	スマートフォンアプリケーションを利用して路面状態マップを閲覧する
事前条件	・路面異常情報がサーバに記録されていること ・アプリケーションが起動していること
基本系列	1. アプリケーションはサーバに路面状態マップを要求する 2. サーバはデータベースに記録されている路面異常情報を地図にプロットし、この情報をアプリケーションに送信する 3. アプリケーションは画面に路面状態マップを提示する
代替系列	
事後条件	路面状態マップがアプリケーション上に表示されていること
備考	地図は既存の地図サービスを利用する

ユースケース ID	UC-06
ユースケース名	Web ブラウザから路面状態マップを閲覧する
概要	Web ブラウザを利用して路面状態マップを閲覧する
事前条件	・路面異常情報がシステムに記録されていること ・Web ブラウザが起動していること
基本系列	1. 路面状態マップ閲覧者は Web ブラウザから路面状態マップ閲覧画面にアクセスし、路面状態マップの表示をサーバに要求する 2. サーバはデータベースに記録されている路面異常情報を地図にプロットし、この情報を Web ブラウザに送信する 3. Web ブラウザは路面状態マップを表示する 4. 路面状態マップ閲覧者は、表示された地図を閲覧し、本ユースケースを終了する
代替系列	
事後条件	路面状態マップが Web ブラウザに表示されていること
備考	地図は既存の地図サービスを利用する

ユースケース ID	UC-07
ユースケース名	走行情報をダウンロードする
概要	Web ブラウザを利用して走行情報をダウンロードする
事前条件	・路面異常情報がシステムに記録されていること ・Web ブラウザが起動しており、「路面状態マップ画面」が表示されていること
基本系列	1. 路面状態マップ閲覧者は路面状態マップ閲覧画面から「Download」リンクを押下し、路面状態マップ閲覧画面に表示されている範囲の走行情報、および路面異常情報の一覧を表示するようサーバに要求する 2. サーバはデータベースに記録されている走行情報のうち、指定範囲の走行情報および路面異常情報の一覧を取得し、この情報を Web ブラウザに送信する 3. Web ブラウザは「走行情報および路面異常情報ダウンロード画面」に走行情報および路面異常情報のファイル名一覧を表示する 4. 路面状態マップ閲覧者は、表示された一覧のうち、任意の走行情報・路面異常情報のファイルにチェックを入れ、[ダウンロード]ボタンを押下する 5. サーバは、要求にしたがってチェックされた走行情報（、路面異常情報）をデータベースから抽出し、CSV ファイルにしてブラウザへ送信する 6. 閲覧している PC にチェックを入れたファイルが保存され、本ユースケースを終了する
代替系列	4a. 基本系列 4 において、複数の走行情報・路面異常情報のチェックが入った場合、 1. サーバは、複数の CSV ファイルを生成後、1 つのファイルにアーカイブして Web ブラウザへ送信する 2. 基本系列 6 へ進む
事後条件	・走行情報が閲覧している PC に保存されていること ・「走行情報および路面異常情報ダウンロード画面」が表示されていること
備考	

ユースケース ID	UC-08
ユースケース名	地図を更新する
概要	路面状態マップを最新の情報に更新する
事前条件	・アプリケーション（または Web ブラウザ）が起動していること ・路面状態マップが表示されていること
基本系列	1. 路面状態マップ閲覧者はアプリケーション（または Web ブラウザ）に表示されている更新ボタンを押下する 2. サーバは、最新の路面状態情報を地図に付加した路面状態マップをアプリケーション（または Web ブラウザ）に送信する 3. アプリケーション（または Web ブラウザ）は画面を更新し、最新の路面状態マップを表示する
代替系列	
事後条件	路面状態マップと更新日時が最新の情報に更新されていること
備考	UC-05、UC-06 と汎化関係にある

ユースケース ID	UC-09
ユースケース名	地図を移動する
概要	路面状態マップを移動させ、任意の地点を閲覧する
事前条件	・アプリケーション（または Web ブラウザ）が起動していること ・路面状態マップが表示されていること
基本系列	1. 路面状態マップ閲覧者は目的の位置に地図の表示を移動させる 2. サーバは移動後の表示範囲内の路面異常情報を地図に付加した路面状態マップをアプリケーション（または Web ブラウザ）に送信する 3. アプリケーション（または Web ブラウザ）は画面を更新し、移動後の路面状態マップを表示する
代替系列	
事後条件	移動後の路面状態マップが表示されていること
備考	・移動に関する基本的な操作などは、本システムで利用する地図サービスに従う ・ UC-05、UC-06 と汎化関係にある

ユースケース ID	UC-10
ユースケース名	地図を拡大/縮小する
概要	路面状態マップの任意の場所を拡大/縮小する
事前条件	・ アプリケーション（または Web ブラウザ）が起動していること ・ 路面状態マップが表示されていること
基本系列	1. 路面状態マップ閲覧者はアプリケーション（または Web ブラウザ）上で、目的の場所で拡大/縮小操作を行う 2. サーバは拡大/縮小後の表示範囲内の路面異常情報を付加した路面状態マップをアプリケーション（または Web ブラウザ）に送信する 3. アプリケーション（または Web ブラウザ）は画面を更新し、拡大/縮小後の路面状態マップを表示する
代替系列	
事後条件	拡大/縮小後の表路面状態マップが表示されていること
備考	・ 移動に関する基本的な操作などは、本システムで利用する地図サービスに従う ・ UC-05、UC-06 と汎化関係にある

3.2 本システムで実装する機能

本システムはスマートフォン、クライアント PC、サーバで構成される。それぞれの構成要素において、本システムがサポートする機能を以下に示す。

- スマートフォンでサポートする機能
 - ・ 走行情報収集機能
 - ・ データ送信機能
 - ・ 路面状態マップ閲覧機能
- クライアント PC でサポートする機能
 - ・ 路面状態マップ閲覧機能
 - ・ 走行情報・路面異常情報ダウンロード機能
- サーバでサポートする機能
 - ・ データ受信機能
 - ・ データ解析機能
 - ・ 路面状態マップ生成機能
 - ・ データ送信機能
 - ・ データ管理機能
 - ・ 異常推定箇所検出機能

次項より、各機能の詳細について説明する。

3.2.1 走行情報収集機能

機能 ID	機能名称	概要
FN-01	加速度取得	3 軸加速度の値を取得する
FN-02	位置取得	緯度・経度を取得する
FN-03	方角取得	走行方向の方角を取得する
FN-04	速度取得	大まかな走行速度を取得する
FN-05	自動記録	取得した走行情報を一時的にファイルに記録する

3.2.2 データ送信機能

機能 ID	機能名称	概要
FN-06	圧縮	走行情報を記録したファイルを圧縮する
FN-07	ファイル送信	圧縮したファイルをサーバに送信する
FN-08	自動削除	走行情報の記録ファイルおよび圧縮ファイルを自動的に削除する

3.2.3 路面状態マップ閲覧機能（スマートフォンサイド）

機能 ID	機能名称	概要
FN-09	地図の表示	ユーザの端末に路面状態マップを表示する
FN-10	地図の移動	任意に路面状態マップの移動を行う
FN-11	地図の拡大/縮小	任意に路面状態マップの拡大/縮小を行う
FN-12	地図の更新	路面状態マップを最新の状態に更新する

3.2.4 路面状態マップ閲覧機能（クライアント PC サイド）

機能 ID	機能名称	概要
FN-13	地図の表示	路面状態マップを Web ブラウザに表示する
FN-14	地図の移動	Web ブラウザ上で任意に路面状態マップの移動を行う
FN-15	地図の拡大/縮小	Web ブラウザ上で任意に路面状態マップの拡大/縮小を行う
FN-16	地図の更新	路面状態マップを最新の状態に更新する
FN-17	データのダウンロード	任意の走行情報、路面異常情報を Web ブラウザからダウンロードする

3.2.5 データ受信機能

機能 ID	機能名称	概要
FN-18	ファイル受信	スマートフォンから送信される圧縮ファイルを受信する
FN-19	解凍	圧縮ファイルを解凍し、走行情報を取り出す

3.2.6 データ解析機能

機能 ID	機能名称	概要
FN-20	異常推定箇所取り出し	データベースサーバから異常推定箇所を取り出す
FN-21	走行情報取り出し	データベースサーバから走行情報を取り出す
FN-22	ノイズ除去	走行情報の加速度データからノイズを除去する
FN-23	統計処理	取り出した加速度データから路面の特徴量を得る
FN-24	異常有無判定	統計処理によって得られた特徴量から路面異常の有無を判定する
FN-25	路面異常データ格納	路面異常が存在する位置をデータベースに格納する
FN-26	データベース修正	異常推定箇所をデータベースから削除する

3.2.7 路面状態マップ生成機能

機能 ID	機能名称	概要
FN-27	路面状態マップの生成	既存の地図サービスに路面異常情報をアイコンとして付加し、路面状態マップを生成する

3.2.8 データ送信機能

機能 ID	機能名称	概要
FN-28	データ送信	FN-27 で生成した路面異常マップをクライアントに送信する

3.2.9 データ管理機能

機能 ID	機能名称	概要
FN-29	走行情報の記録	FN-19 で解凍した走行情報をデータベースに記録する
FN-30	走行情報の検索	条件絞り込みによる走行情報の検索を行う
FN-31	走行情報の参照	データベースに記録されている走行情報を参照する
FN-32	走行情報の削除	データベースに記録されている走行情報を削除する
FN-33	異常候補点の記録	FN-24 で検出した異常候補点をデータベースに記録する
FN-34	異常候補点の検索	条件絞り込みによる異常候補点の検索を行う
FN-35	異常候補点の参照	データベースに記録されている異常候補点を参照する
FN-36	異常候補点の更新	データベースに記録されている異常候補点を更新する
FN-37	異常候補点の削除	データベースに記録されている異常候補点を削除する

3.2.10 異常推定箇所検出機能

機能 ID	機能名称	概要
FN-38	走行情報圧縮ファイル取り出し	データ解析サーバに保存されている走行情報圧縮ファイルを取り出す
FN-39	走行情報圧縮ファイル解凍	取り出した走行情報圧縮ファイルを解凍し、元の圧縮ファイルを削除する
FN-40	異常推定箇所検出	解凍した走行情報から路面の異常推定箇所を検出する
FN-41	異常推定箇所格納	検出された異常推定箇所をデータベースサーバに格納する
FN-42	走行情報格納	異常推定箇所検出後の走行情報をデータベースに格納する

4 画面イメージ

本章では、スマートフォンアプリケーションおよび Web ブラウザにおける路面状態マップの閲覧画面イメージについてそれぞれ記述する。

4.1 アプリケーションから路面状態マップを閲覧する場合の画面イメージ

画面 ID	SC-01
画面名	路面状態マップ閲覧画面（アプリケーション）

① 2013/07/15(Mon) 15:00 更新

②

2013/07/15(Mon) 12:00

③ 更新

④ 測定中...

⑤ 終了

要素 ID	パラメータ名 (和)	パラメータ名 (英)	型	送受信属性		備考
				送信	受信	
1	更新時刻	update-time	ラベル	0	1	更新時刻が表示される
2	路面状態マップ	map	地図	0	1	路面状態マップが表示される
3	更新	update	ボタン	1	0	路面状態マップと更新時刻が更新される
4	稼働通知	notification	ラベル	0	1	稼働中の旨を表示する
5	終了	finish	ボタン	1	0	閲覧・測定を終了し、アプリケーションを終了する

4.2 クライアント PC から異常状態マップを閲覧する場合の画面イメージ

画面ID	SC-02
画面名	路面状態マップ閲覧画面（ブラウザ）

The image is a screenshot of a web browser displaying a road status map. The map shows a network of roads with various labels in Japanese, including '台坪入口', '筑波大学 附属図書館', 'つくばビジネスカレッジ専門学校', '鹿島神社', '1丁目', '巨町の森公園', '大学公園南', and '筑波大通り'. A red box labeled '2' encloses the map area. A callout bubble points to a specific location on the map, labeled '2013/07/15(Mon) 12:00'. Another red box labeled '3' encloses an '更新' (Update) button at the bottom left. A red box labeled '1' encloses the text '2013/07/15(Mon) 15:00 更新' at the top right. The map includes a scale bar (0 to 200 meters) and a compass.

要素ID	パラメータ名 (和)	パラメータ名 (英)	型	送受信属性		備考
				送信	受信	
1	更新時刻	update-time	ラベル	0	1	更新時刻が表示される
2	路面状態マップ	map	地図	0	1	路面状態マップが表示される
3	更新	update	ボタン	1	0	路面状態マップと更新時刻が更新される

5 データベース仕様

図 5 に、本システムにおけるデータベースの仕様を示す。

走行情報	路面異常箇所
記録ファイル名 記録日 記録時刻 ID 緯度 経度 加速度X 加速度Y 加速度Z 方角 速度	検出日 検出時刻 緯度 経度 方角 異常レベル

図 5：データベースに格納するエンティティ

6 システム仕様

本章では、本システムでサポートする機能の仕様について記述する。

6.1 スマートフォンサイドのシステム仕様

仕様 ID	仕様	関連機能 ID
SP-01	加速度の取得については、スマートフォンに搭載されている加速度センサを利用して取得する。また、加速度値の取得は 0.03 秒ごとに行う。	FN-01
SP-02	位置の取得については、スマートフォンに搭載されている GPS センサを利用して緯度・経度を取得する。また、位置の取得は 1 秒ごと、もしくは 1 メートルの移動が発生した際に行う。	FN-02
SP-03	方角の取得については、スマートフォンに搭載されている GPS センサをもとに算出されるものを利用し、北を 0 度として 0～360 度の範囲で記録するものとする。また、方角の取得は 1 秒ごと、もしくは 1 メートルの移動が発生した際に行う。	FN-03
SP-04	速度の取得については、スマートフォンに搭載されている GPS センサをもとに算出されるものを利用して記録し、記録単位は km/h とする。また、速度の取得は 1 秒ごと、もしくは 1 メートルの移動が発生した際に行う。	FN-04
SP-05	端末に一時的に記録するデータの形式は CSV ファイルとする。 ファイルはアプリケーションの起動時に毎回新規作成される。なお、この時のファイル名は、アプリケーション起動時刻を付加した“YYYYMMDD_hhmmssSSS.csv”とする。 ファイルに書き込むデータとしては、FN-01、FN-02、FN-03、FN-04 での取得時刻および計測値とする。	FN-05
SP-06	記録したデータは、gzip 形式で圧縮する。 また、このときの拡張子を除いたファイル名は FN-04 と同じものとする。	FN-06
SP-07	スマートフォン－サーバ間のデータ通信におけるプロトコルは HTTP を用いるものとする。	FN-07 FN-18
SP-08	ファイルを送信するタイミングは、アプリケーションの終了時とする。	FN-07
SP-09	FN-05 で作成されたファイルを削除するタイミングは、当該ファイルが FN-07 の機能によりサーバにデータが送信完了したときとする。	FN-08
SP-10	コンテンツの更新は手動で行うものとする。	FN-12
SP-11	路面異常マップに表示する推定異常箇所は、推定の確かさ、路面異常のレベルなどに応じて表示方法を変えるものとする。	FN-09 FN-24
SP-12	路面異常マップの閲覧操作（移動・拡大/縮小）については、本システムで利用する地図サービスに基づくものとする。	FN-10～ FN-12

6.2 クライアント PC のシステム仕様

仕様 ID	仕様	関連機能 ID
SP-13	路面状態マップの表示について、サーバからのレスポンス時間は、平常時 3 秒、ピーク時 6 秒を目標とする。	FN-13
SP-14	路面異常マップに表示する推定異常箇所は、推定の確かさ、路面異常のレベルなどに応じて表示方法を変えるものとする。	FN-13 FN-24
SP-15	路面異常マップの閲覧操作（移動・拡大/縮小）については、本システムで利用する地図サービスに基づくものとする。	FN-14～ FN-16
SP-16	ダウンロードする走行情報には、①記録時刻、②緯度・経度、③3 軸加速度④方角の値を含めるものとする。路面異常情報には、①検出時刻、②緯度・経度、③推定異常レベルの情報を含めるものとする。	FN-17

6.3 サーバサイドのシステム仕様

仕様 ID	仕様	関連機能 ID
SP-17	サーバに同時に接続できる最大のクライアント数は 100 とする。	FN-07 FN-18
SP-18	データの受信については、認証を行った上で実行するものとする (詳細は 7.1 節を参照)	FN-18
SP-19	1 件のデータ受信におけるレスポンス時間は、平常時 10 秒、ピーク時 20 秒を目標とする。	FN-18
SP-20	サーバに搭載するシステム、および格納データの総容量は 10GB までとする。	—
SP-21	走行情報データベースに記録できる最大件数は 2000 件とする。	FN-29
SP-22	推定異常箇所データベースに記録できる最大件数は 10000 件とする	FN-33
SP-23	データ解析における処理時間は、1 回あたり 1 時間以内を目標とする。	FN-20～ FN-26
SP-24	推定異常箇所データベースの更新頻度は、1 時間に 1 回とする。	FN-36
SP-25	路面異常マップの生成方法は、既存の地図サービスを利用する。緯度・経度、および路面異常のレベルをアイコンとして地図に重ね合わせ、これを路面異常マップとして提供するものとする。	FN-27
SP-26	データベースは、本システムのサーバのみから接続を受け付けるものとし、走行情報提供者など一般ユーザからの接続は受け付けない。	FN-29～ FN-42
SP-27	ノイズ除去は、ローパスフィルタを用いて行うものとする。	FN-20
SP-28	異常候補点検出のロジックは、現在仕様未定である。 (詳細は 7.1 節を参照)	FN-21
SP-29	統計処理には、現在仕様未定である。 (詳細は 7.1 節を参照)	FN-22
SP-30	異常有無の判定ロジックは、現在仕様未定である。 (詳細は 7.1 節を参照)	FN-23

7 未確定事項

7.1 検討事項

ここでは、確定はしていないものの、重要と認識している事案を記述する。特に異常検出の精度のように実際にデータを収集し、異常検出を行ってから確認するため検討事項となっているものもある。

以下に、検討事項をまとめる。

No.	検討事項	現状の想定	確定工程
01	データ受信ロジック	ユーザ認証を行った上で受信を行う	実装工程
02	異常検出タイミング	一定数の走行情報がサーバに蓄積された時、 もしくは一定時間経過した時	実装工程
03	路面異常更新頻度	路面異常を検出すると同時に既存の路面異常を 更新する	実装工程
04	異常検出ロジック	分散値から閾値で判断	実装工程
05	統計処理	恒常的な異常と突発的な異常を判断する	実装工程
06	異常レベル	路面異常の確かさを数段階に分けて表示する	実装工程
07	異常種類分類	異常かどうかだけを表示する	実装工程
08	最大データ蓄積量	走行情報、路面異常情報で 1TB 確保する	テスト工程

付録C 実装フェーズにおける成果物

C.1 基本設計書

基本設計書

1.0 版

チームシュークリーム
枝松ちさと
胥徳文
林貴哉
村林竜司

改訂履歴

[illegible]

目次

1 はじめに.....	5
1.1 本書の目的	5
1.2 本書の構成	5
2 ハードウェア構成.....	6
2.1 ハードウェア全体構成	6
3 ソフトウェア構成.....	7
3.1 スマートフォン側.....	7
3.2 サーバ側	8
3.2.1 WEB サーバ.....	8
3.2.2 データベースサーバ.....	9
3.2.3 データ解析サーバ	10
3.3 クライアント PC 側	10
4 機能設計.....	11
4.1 スマートフォン側.....	11
4.1.1 分析クラス図	11
4.1.2 分析シーケンス図	12
4.2 サーバ側	15
4.2.1 WEB サーバ.....	15
4.2.1.1 ロバストネス図	15
4.2.1.2 分析クラス図	15
4.2.1.3 分析シーケンス図.....	16
4.2.2 データベースサーバ.....	16
4.2.3 データ解析サーバ	17
4.2.3.1 機能一覧	17
4.2.3.2 モジュール一覧	18
4.2.3.3 分析クラス図	19
4.2.3.4 分析シーケンス図.....	19
4.3 クライアント PC 側	22
4.3.1 ロバストネス図.....	22
4.3.2 分析クラス図	22
4.3.3 分析シーケンス図	23
5 画面設計.....	25
5.1 スマートフォン側.....	25
5.1.1 画面一覧.....	25
5.1.2 画面詳細.....	25
5.1.3 画面遷移.....	29

5.2 クライアント PC 側	30
5.2.1 画面一覧	30
5.2.2 画面詳細	30
5.2.3 画面遷移	35
6 ファイル定義	36
6.1 ファイル一覧	36
6.2 ファイル詳細	36
7 データベース設計	37
7.1 テーブル一覧	37
7.2 テーブル定義	37

図目次

図 1：ハードウェア全体構成図.....	6
図 2：スマートフォン側 ソフトウェア構成図	7
図 3：WEB サーバ側 ソフトウェア構成図.....	8
図 4：データベースサーバ側 ソフトウェア構成図	9
図 5：データ解析サーバ側 ソフトウェア構成図.....	10
図 6：スマートフォン側 分析クラス図	11
図 7：走行情報収集機能 分析シーケンス図	12
図 8：データ送信機能 分析シーケンス図.....	13
図 9：路面状態マップ閲覧機能 分析シーケンス図	14
図 10：データ受信機能 ロバストネス図	15
図 11：データ受信機能 分析クラス図.....	15
図 12：データ受信機能 分析シーケンス図	16
図 13：異常推定箇所検出機能 処理フロー	17
図 14：データ解析機能 処理フロー	18
図 15：データ解析サーバ側 分析クラス図	19
図 16：データ解析サーバ実行制御 シーケンス図.....	19
図 17：異常推定箇所検出機能 分析シーケンス図.....	20
図 18：データ解析機能 分析シーケンス図	21
図 19：路面状態マップ閲覧機能 ロバストネス図.....	22
図 20：走行情報・路面異常情報ダウンロード機能 ロバストネス図.....	22
図 21：クライアント PC 側 分析クラス図.....	22
図 22：路面状態マップ閲覧機能 分析シーケンス図	23
図 23：走行情報・路面異常情報ダウンロード機能 分析シーケンス図	24
図 24：走行情報ファイルのフォーマット	36

1 はじめに

1.1 本書の目的

本書は、本プロジェクトで開発するシステムの基本設計について記述したものである。このドキュメントの対象者は顧客、および開発チームである。

1.2 本書の構成

本書は以下の構成となっている。

- 2 ハードウェア構成
システムのハードウェア構成を図で示す。
- 3 ソフトウェア構成
システムのソフトウェア構成を、ハードウェアごとに図で示す。
- 4 機能設計
UML を用いてシステムの機能設計を記述する。
- 5 画面設計
各端末における画面設計を記述する。
- 6 ファイル定義
本システムで扱うファイル一覧およびその形式について定義する。
- 7 データベース設計
システムのデータベースおよびテーブルについて定義する。

2 ハードウェア構成

2.1 ハードウェア全体構成

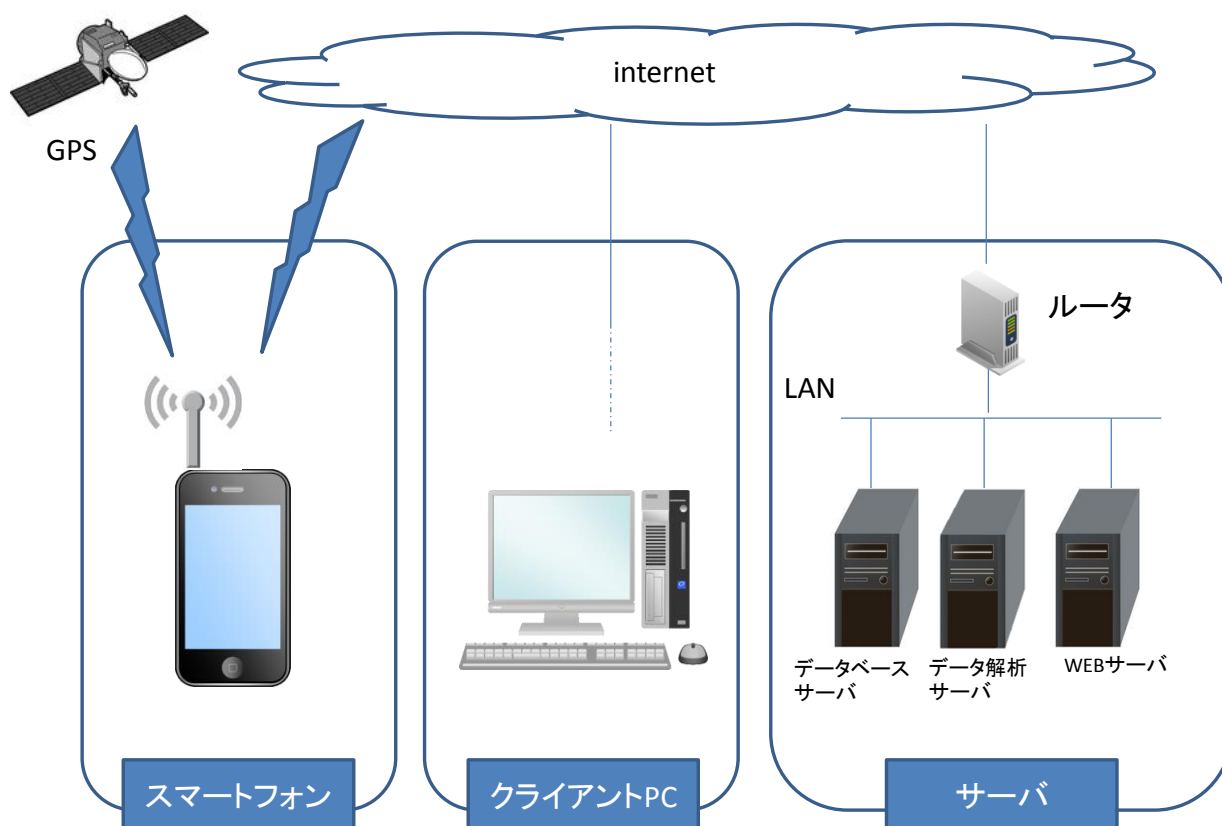


図 1：ハードウェア全体構成図

3 ソフトウェア構成

本章では、以下の表記方法に従ってソフトウェア構成を記述する。

ハードウェアを示す	 : 赤いブロック
既存のソフトウェア（部品）を示す	 : 青いブロック
本プロジェクトで開発するソフトウェアを示す	 : 白いブロック

3.1 スマートフォン側

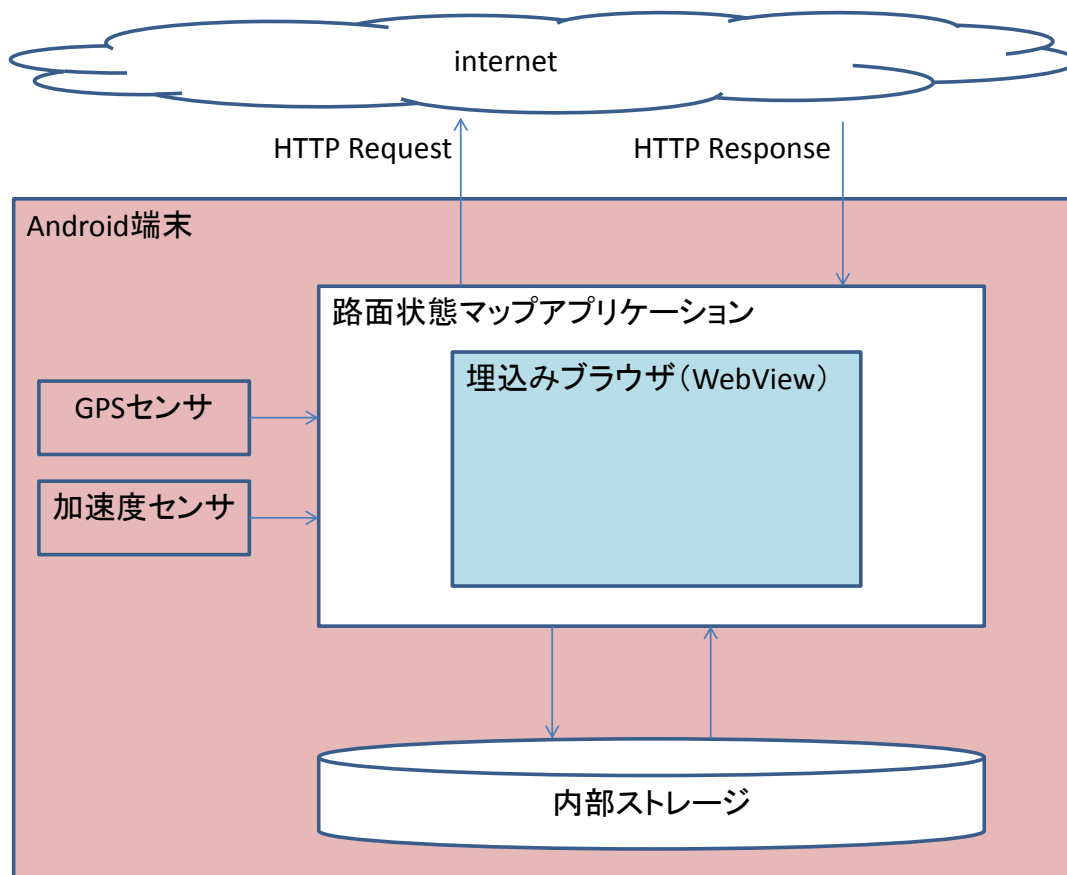


図 2：スマートフォン側 ソフトウェア構成図

保証ソフトウェアバージョン

ターゲット OS	Android 4.2 (最小ターゲットは 2.3)
----------	-------------------------------

3.2 サーバ側

3.2.1 WEB サーバ

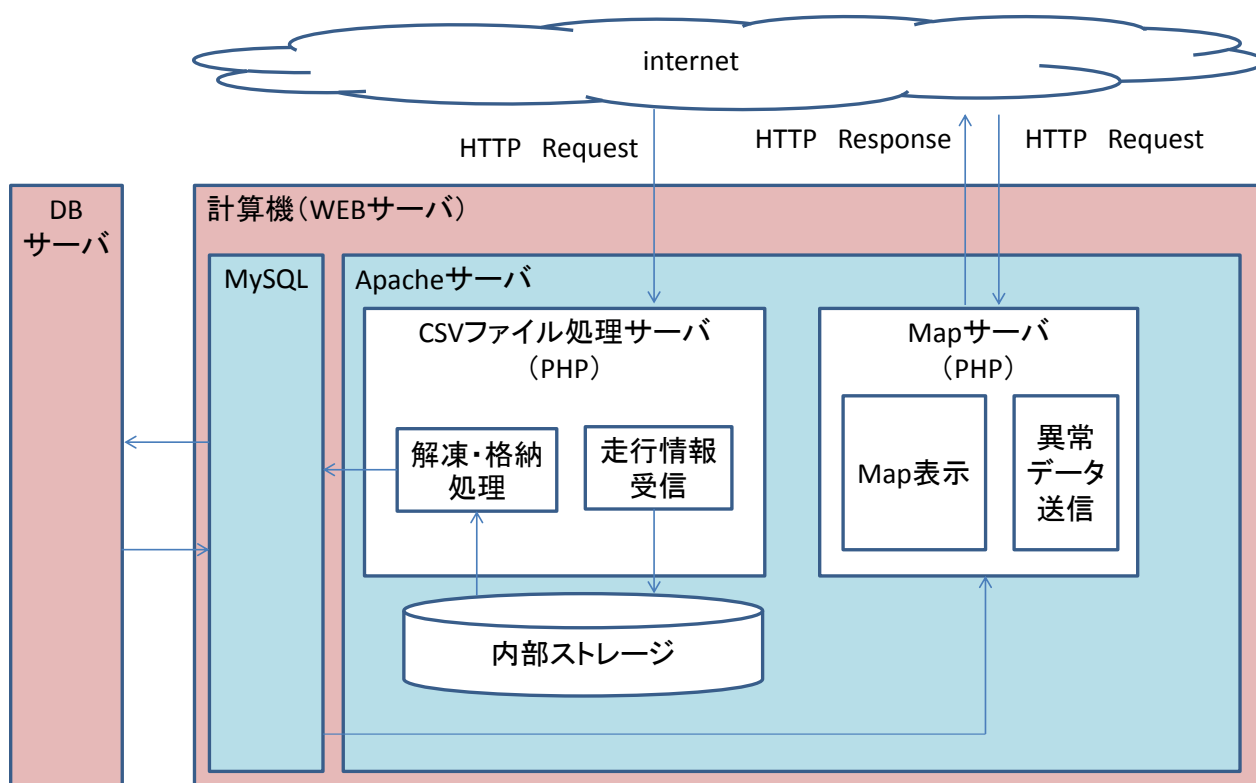


図 3：WEB サーバ側 ソフトウェア構成図

ソフトウェアバージョン情報

OS	CentOS6.4
Web サーバソフトウェア	Apache 2.4.4 (Unix)
PHP	5.4.19

3.2.2 データベースサーバ

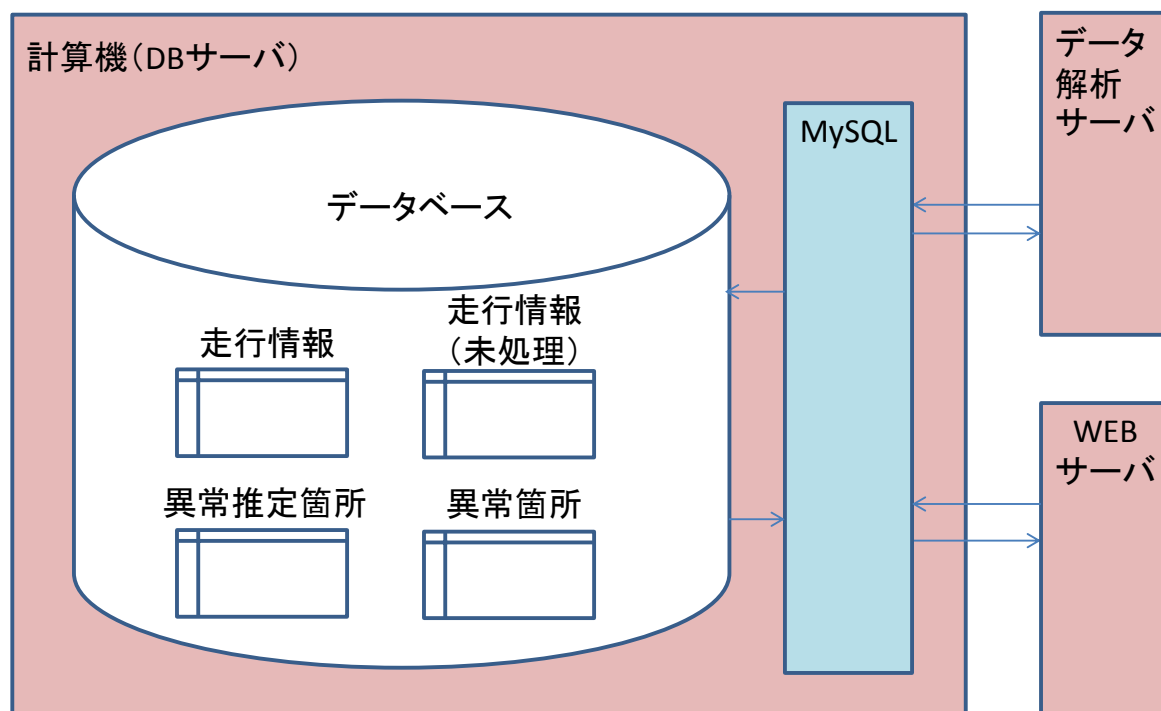


図 4：データベースサーバ側 ソフトウェア構成図

ソフトウェアバージョン情報

OS	CentOS6.4
データベース	MySQL 5.6.12

3.2.3 データ解析サーバ

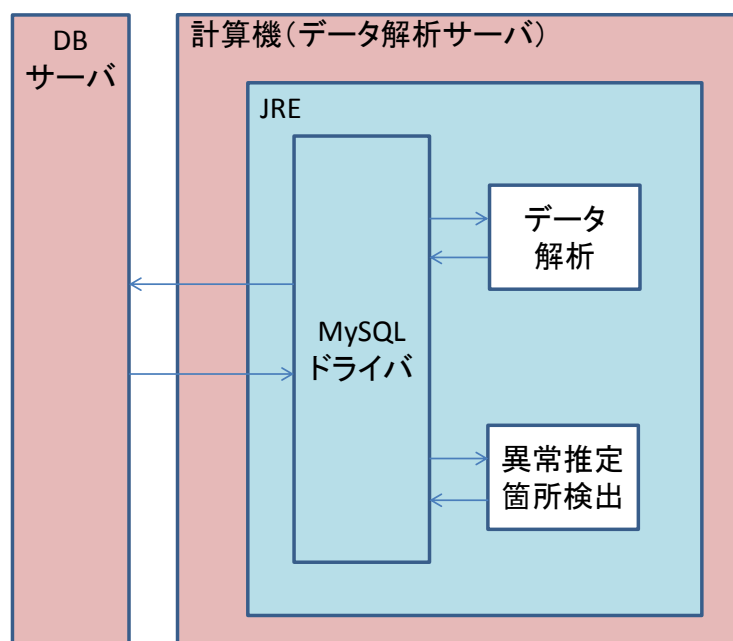


図 5：データ解析サーバ側 ソフトウェア構成図

ソフトウェアバージョン情報

OS	CentOS6.4(仮)
Java	JRE7
MySQL ドライバ	Connector/J 5.1.26

3.3 クライアント PC 側

クライアント PC 側ではブラウザをインストールする。動作保証するブラウザを以下にまとめる。

動作保証ブラウザおよびバージョン一覧

Internet Explorer	10
Chrome	29

4 機能設計

4.1 スマートフォン側

4.1.1 分析クラス図

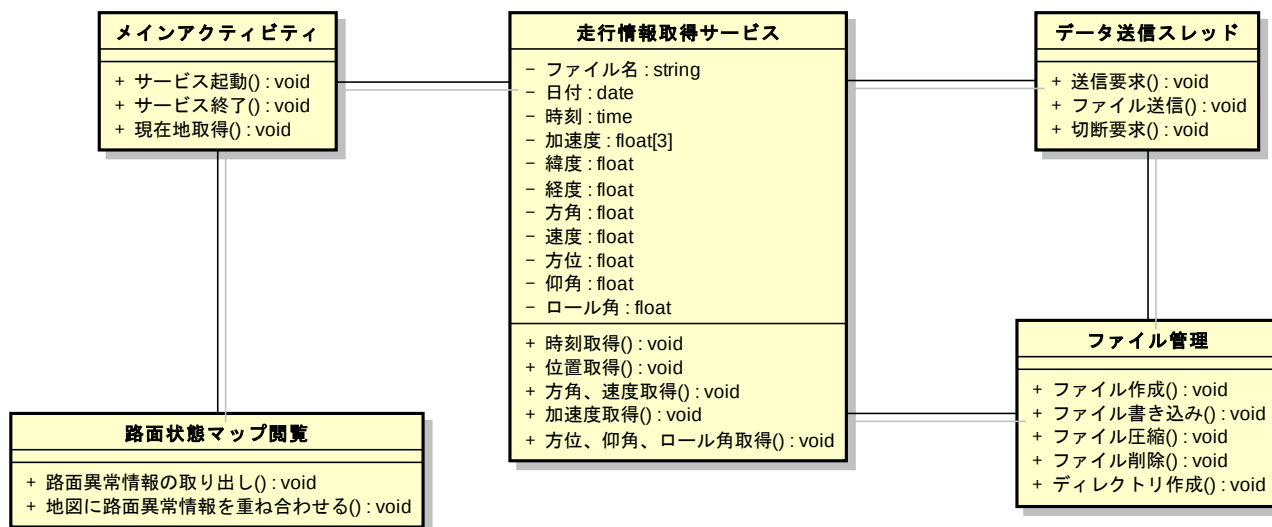


図 6 : スマートフォン側 分析クラス図

4.1.2 分析シーケンス図

走行情報収集機能

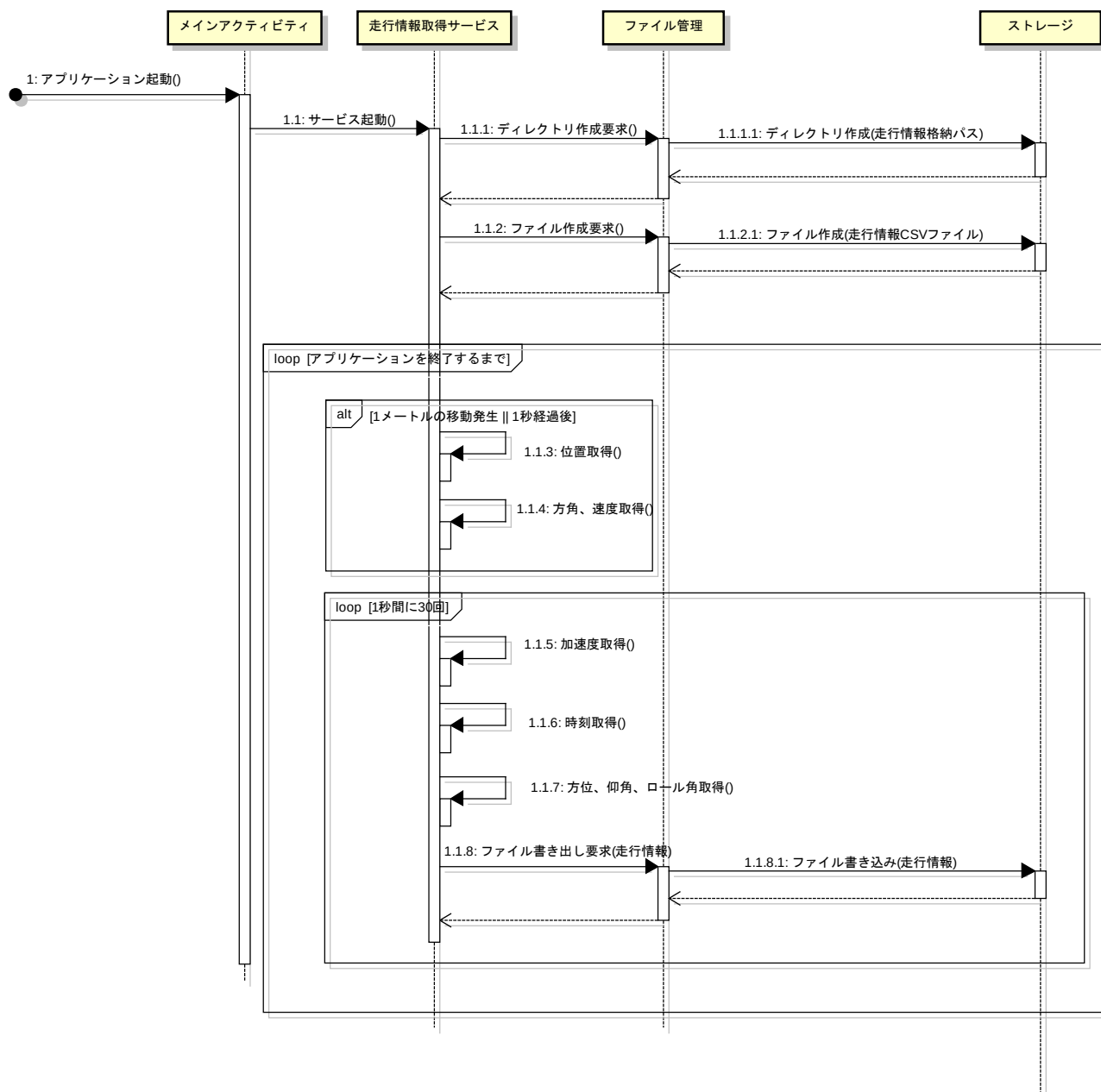


図 7：走行情報収集機能 分析シーケンス図

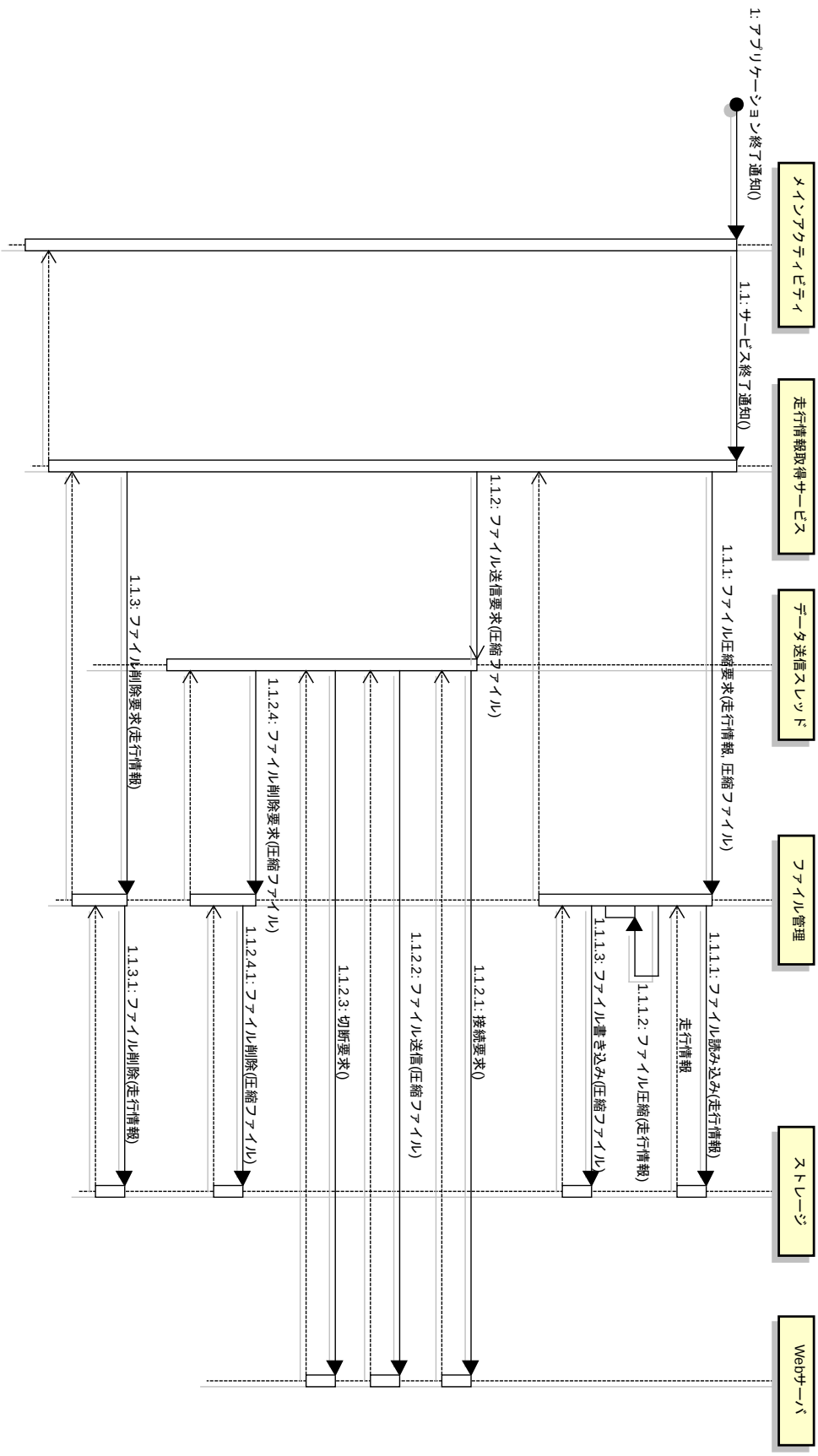


図 8：データ送信機能 分析シーケンス図

路面状態マップ閲覧機能

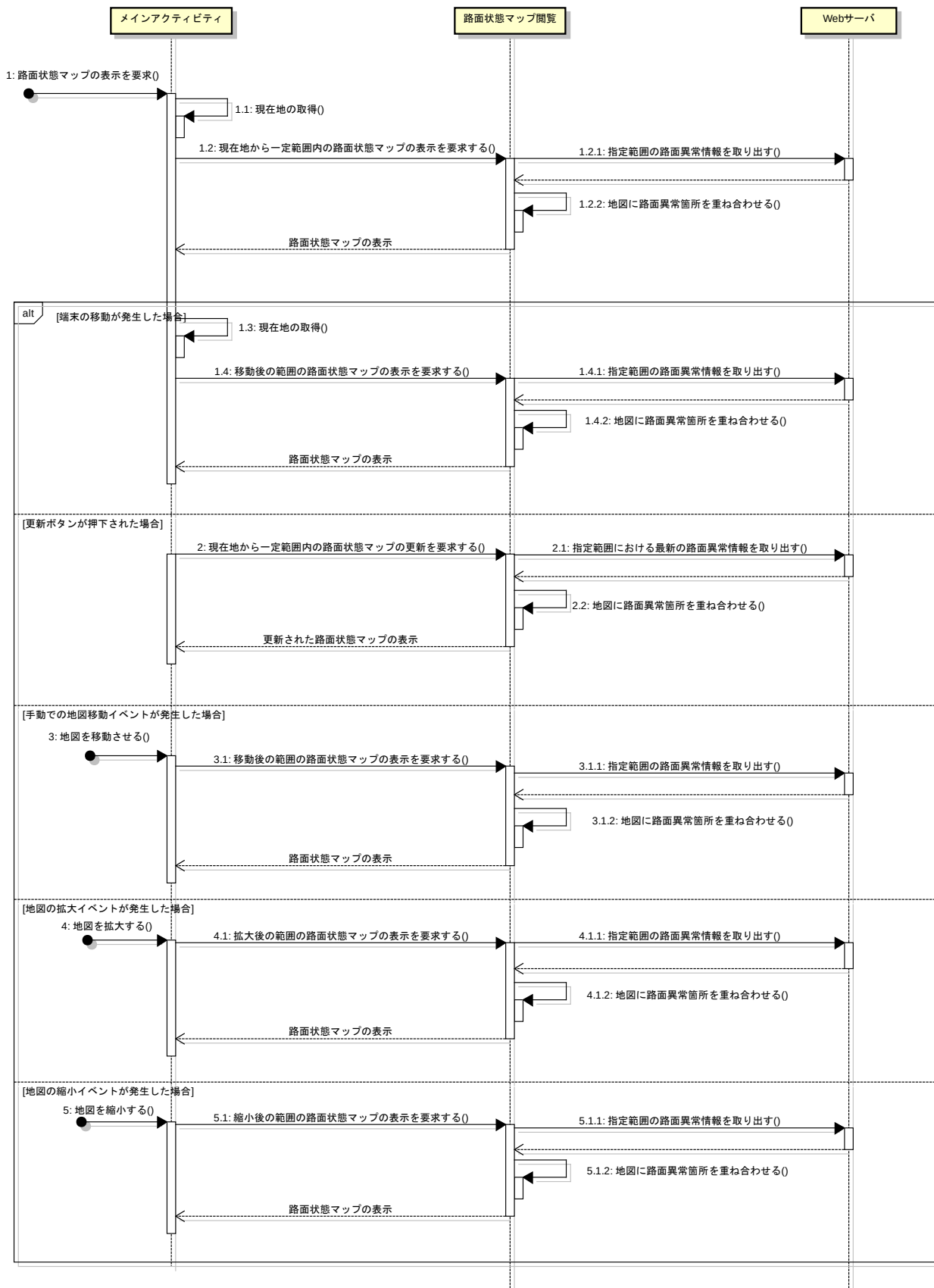


図 9：路面状態マップ閲覧機能 分析シーケンス図

4.2 サーバ側

4.2.1 WEB サーバ

4.2.1.1 ロバストネス図

データ受信機能

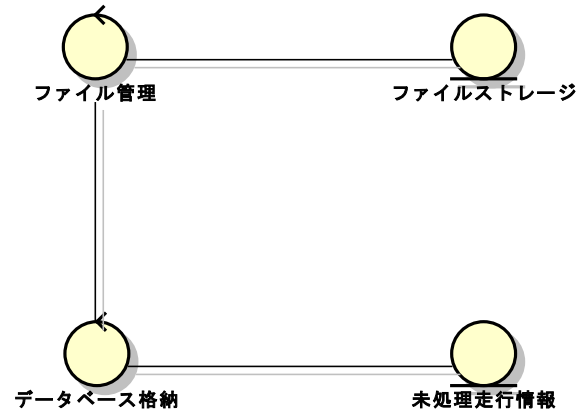


図 10：データ受信機能 ロバストネス図

4.2.1.2 分析クラス図

データ受信機能

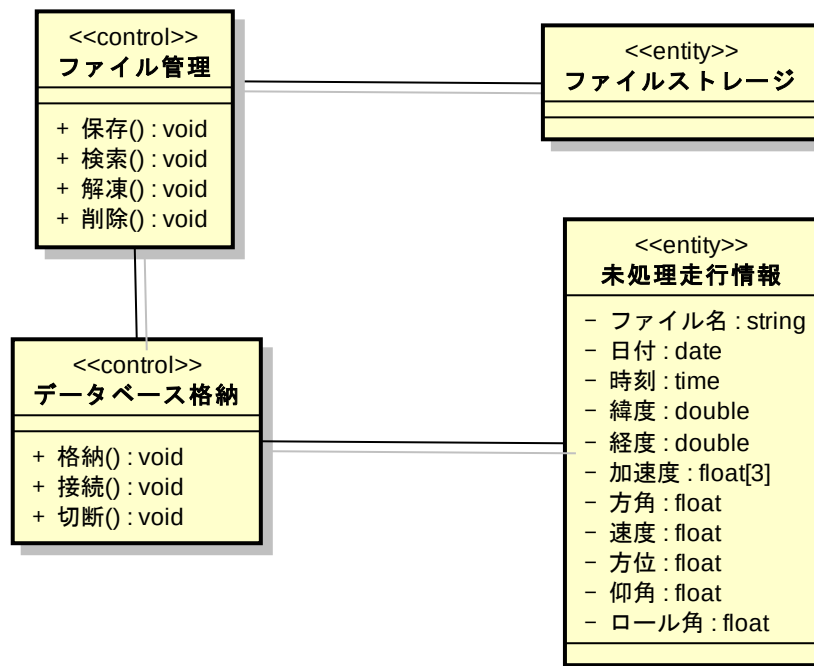


図 11：データ受信機能 分析クラス図

データ受信機能



データベースサーバでは既存のデータベース管理システムを導入し、独自のプログラムは実装しない。データベース設計については7章を参照。

4.2.3 データ解析サーバ

4.2.3.1 機能一覧

異常推定箇所検出機能

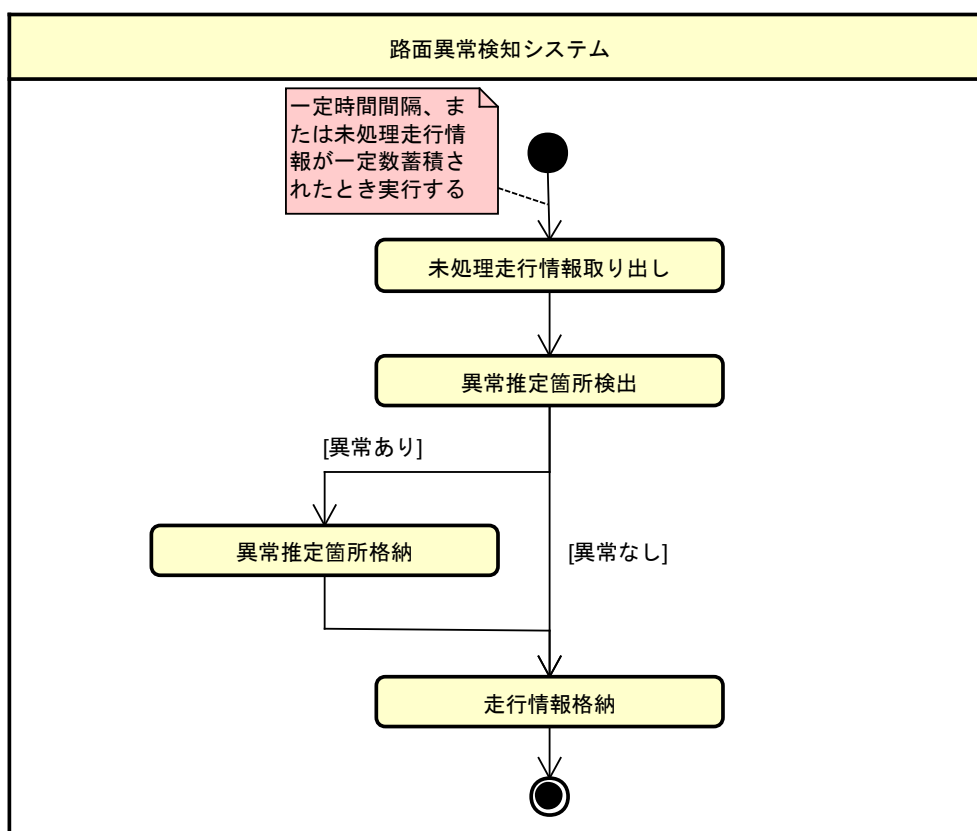


図 13：異常推定箇所検出機能 処理フロー

データ解析機能

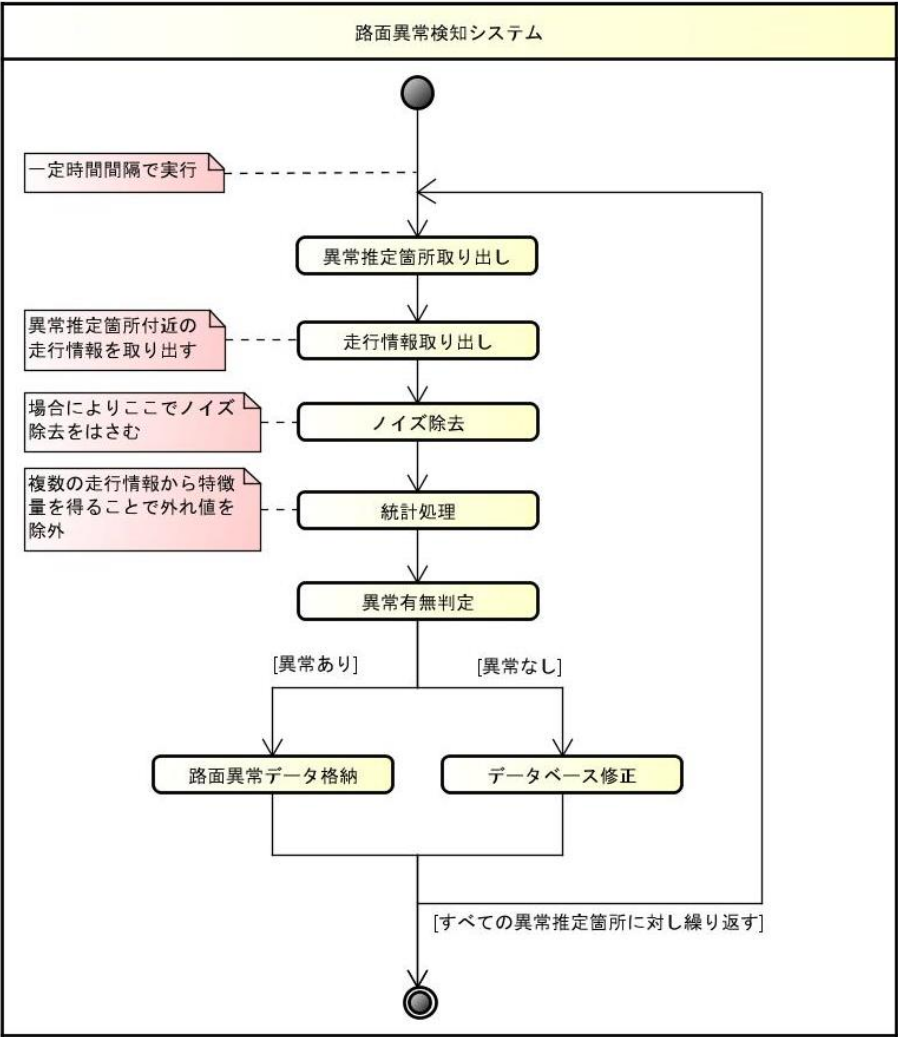


図 14：データ解析機能 処理フロー

4.2.3.2 モジュール一覧

モジュール ID	モジュール名称	概要	入力	出力	関連機能
MD-01	データベース管理	データベースに対し、接続、切断、データの取り出し、格納、削除、更新等を行う	各データ、データベース操作に必要な情報	各データ	FN-25, FN-26, FN-33, FN-34, FN-35, FN-36, FN-37, FN-38, FN-41,

					FN-42
MD-02	異常検出	単数または複数時系列データの変化点を検出する 複数時系列データの場合は統計処理を行った後、変化点検出を行う	時系列データ	変化点データ	FN-23, FN-24, FN-40

4.2.3.3 分析クラス図

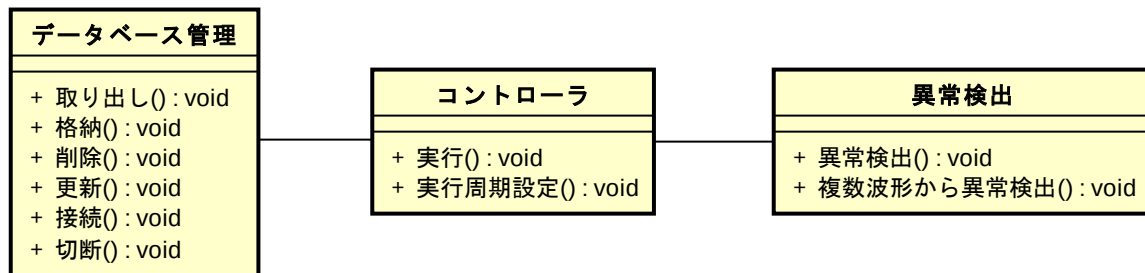


図 15：データ解析サーバ側 分析クラス図

4.2.3.4 分析シーケンス図

データ解析サーバ実行制御

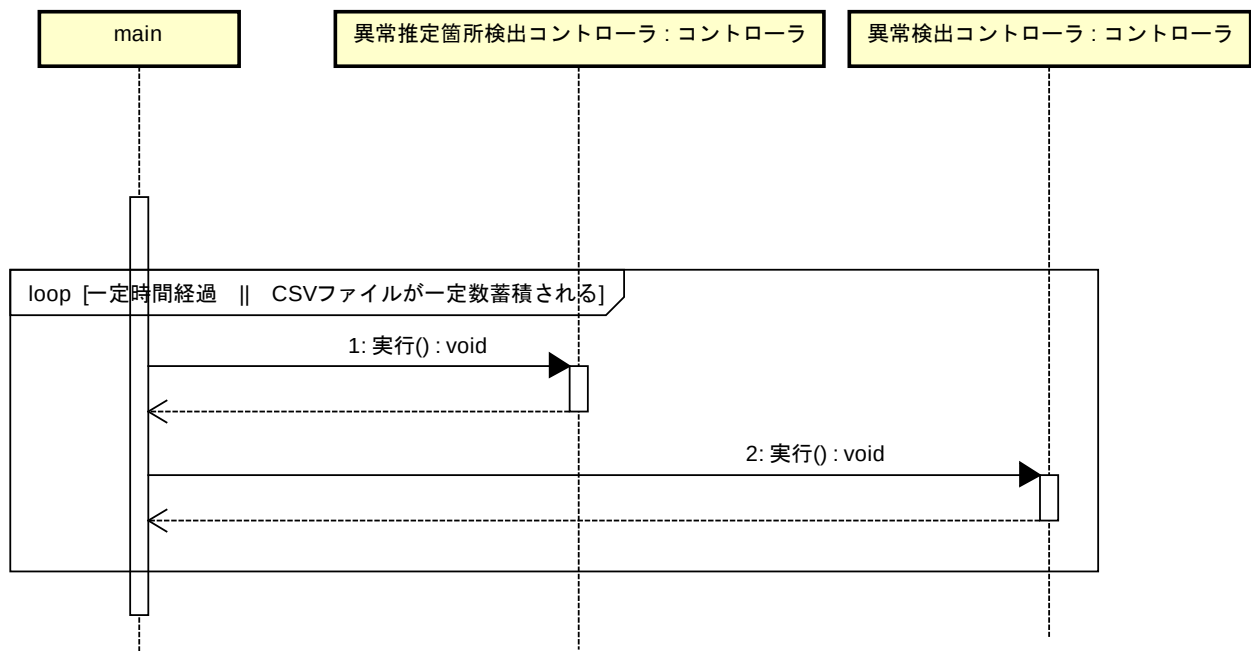


図 16：データ解析サーバ実行制御 シーケンス図

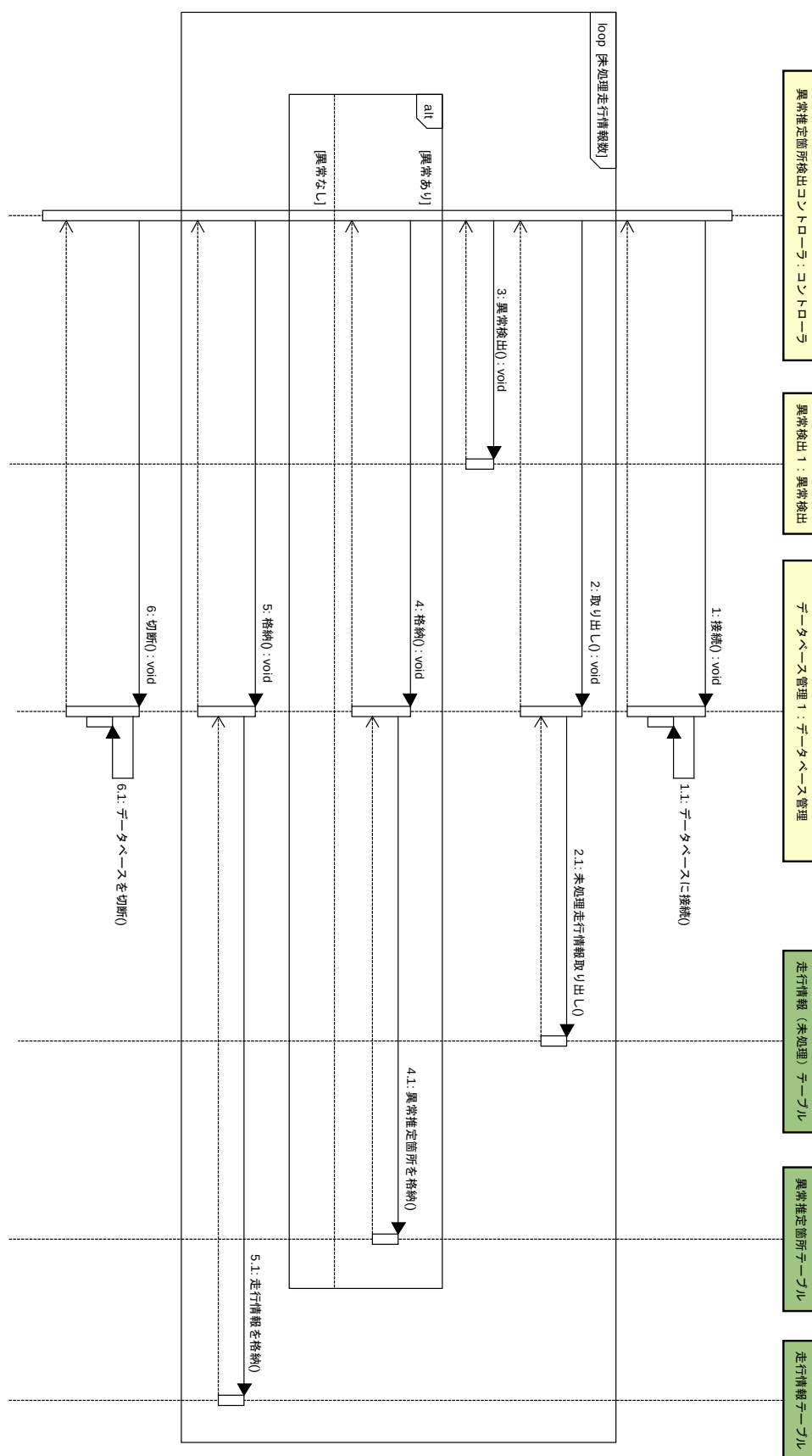


図 17：異常推定箇所検出機能 分析シーケンス図

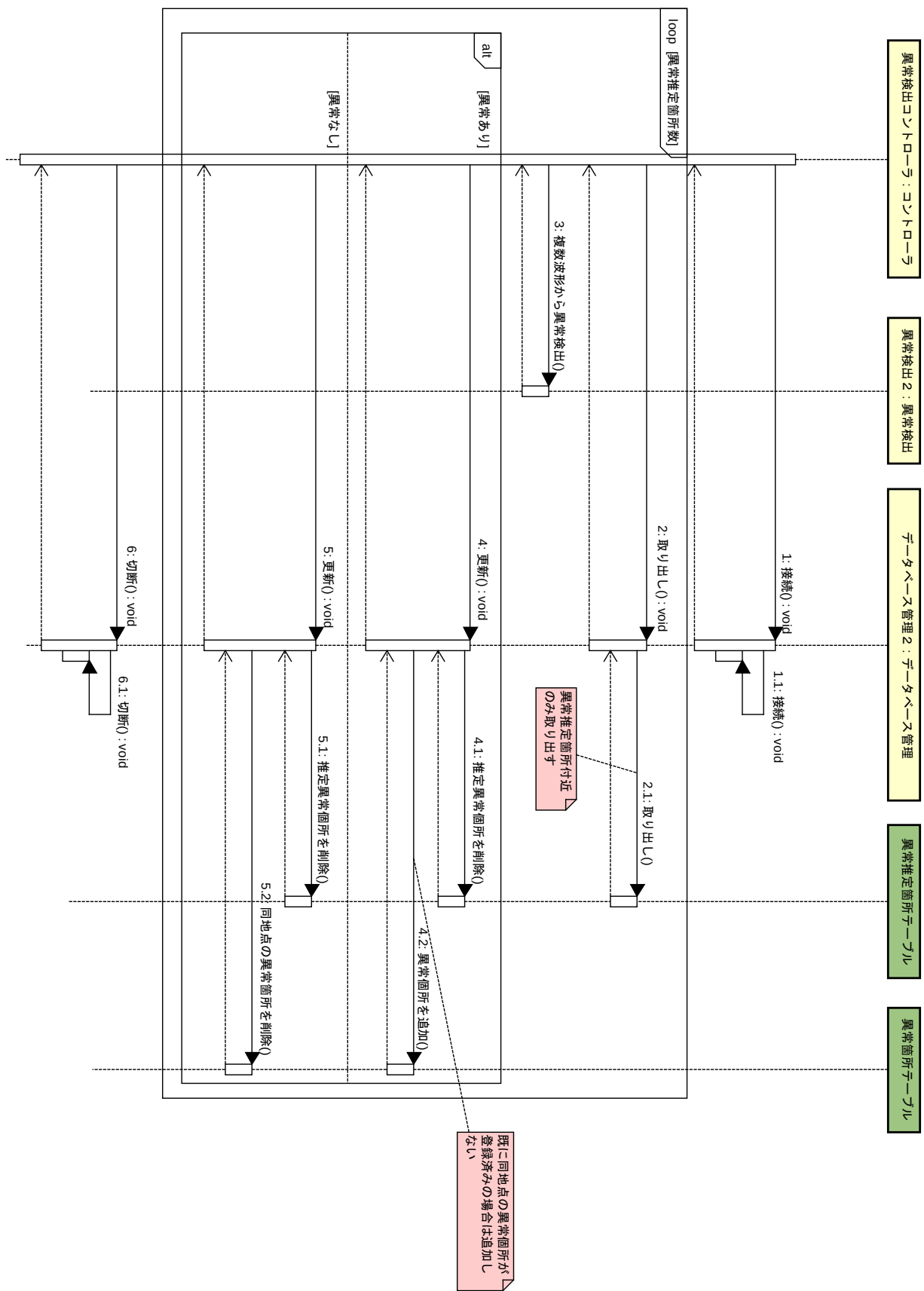


図 18：データ解析機能 分析シーケンス図

4.3 クライアント PC 側

4.3.1 ロバストネス図

路面状態マップ閲覧機能

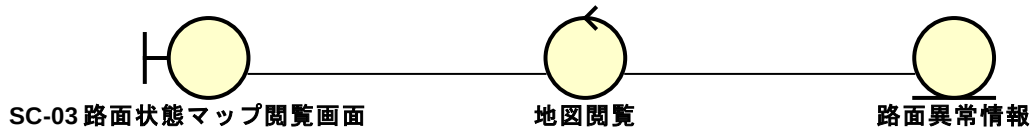


図 19：路面状態マップ閲覧機能 ロバストネス図

走行情報・路面異常情報ダウンロード機能

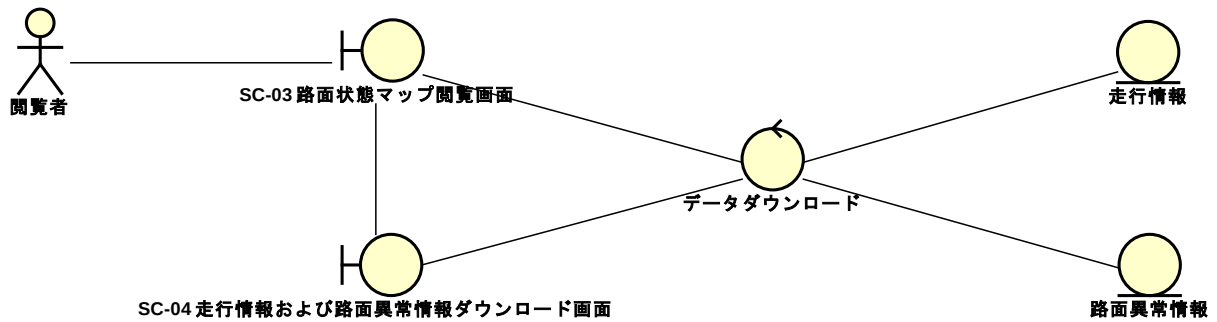


図 20：走行情報・路面異常情報ダウンロード機能 ロバストネス図

4.3.2 分析クラス図

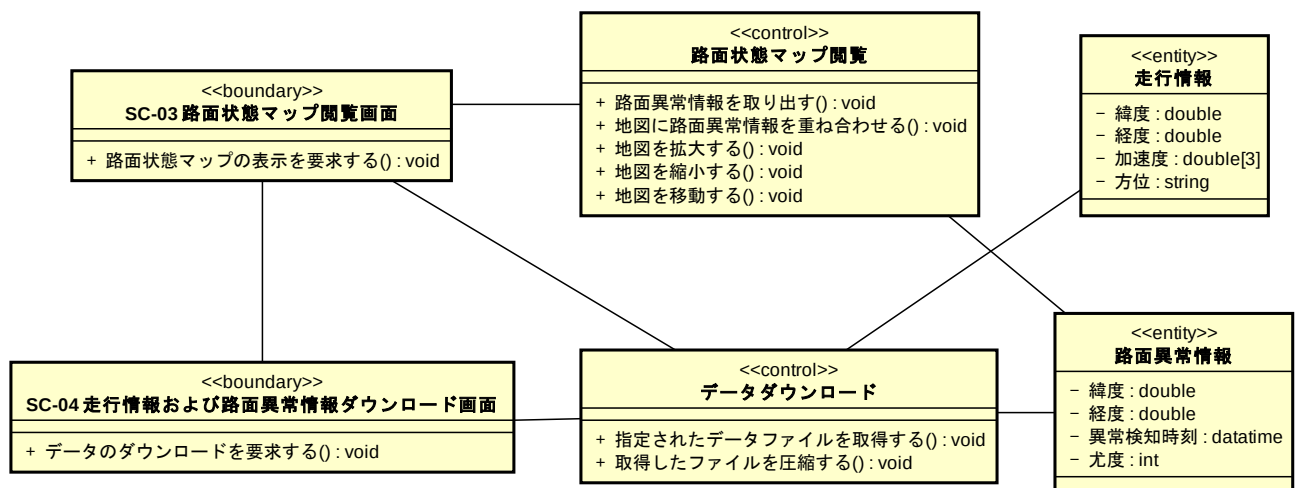


図 21：クライアント PC 側 分析クラス図

4.3.3 分析シーケンス図

路面状態マップ閲覧機能

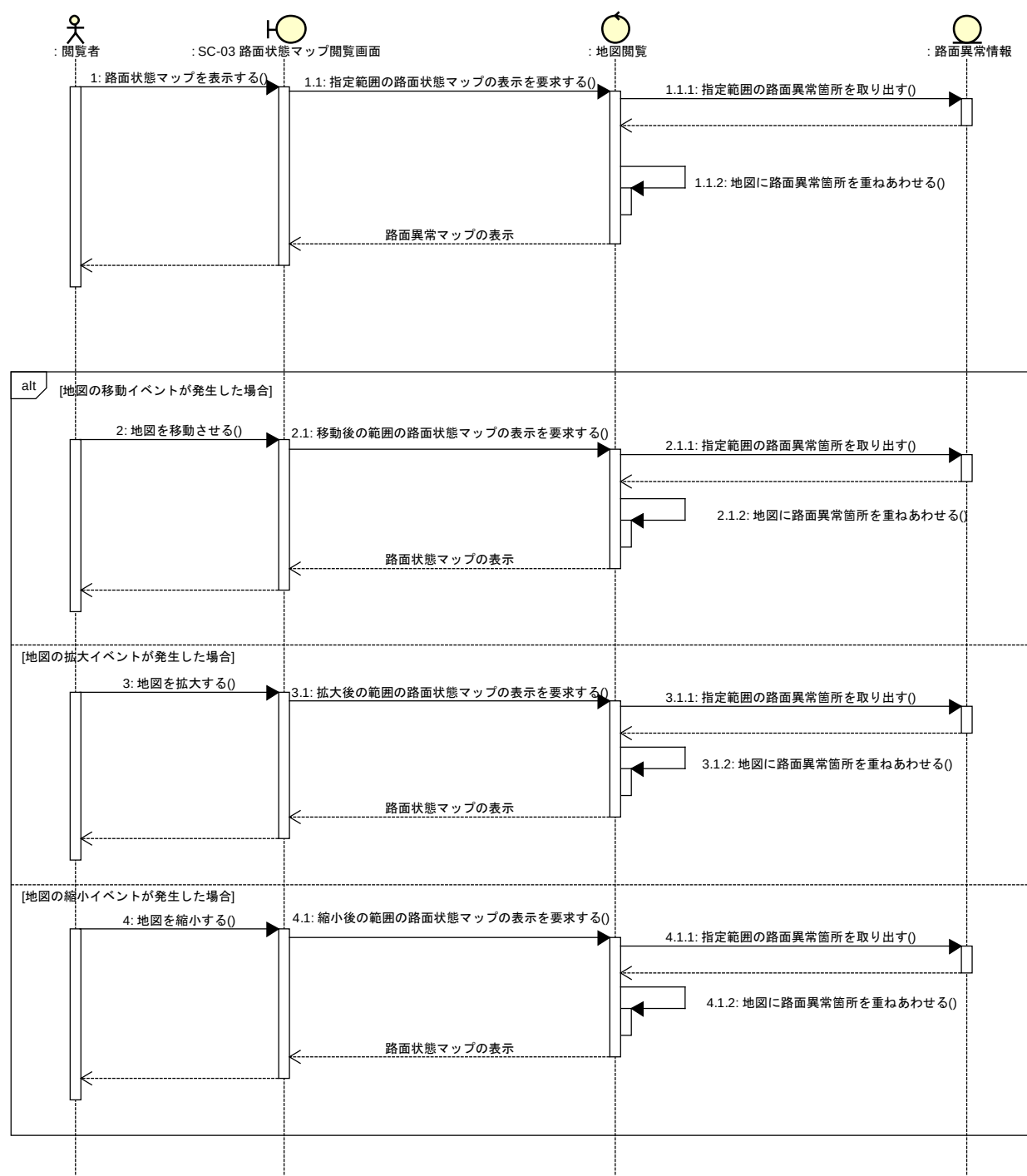


図 22：路面状態マップ閲覧機能 分析シーケンス図

走行情報・路面異常情報ダウンロード機能

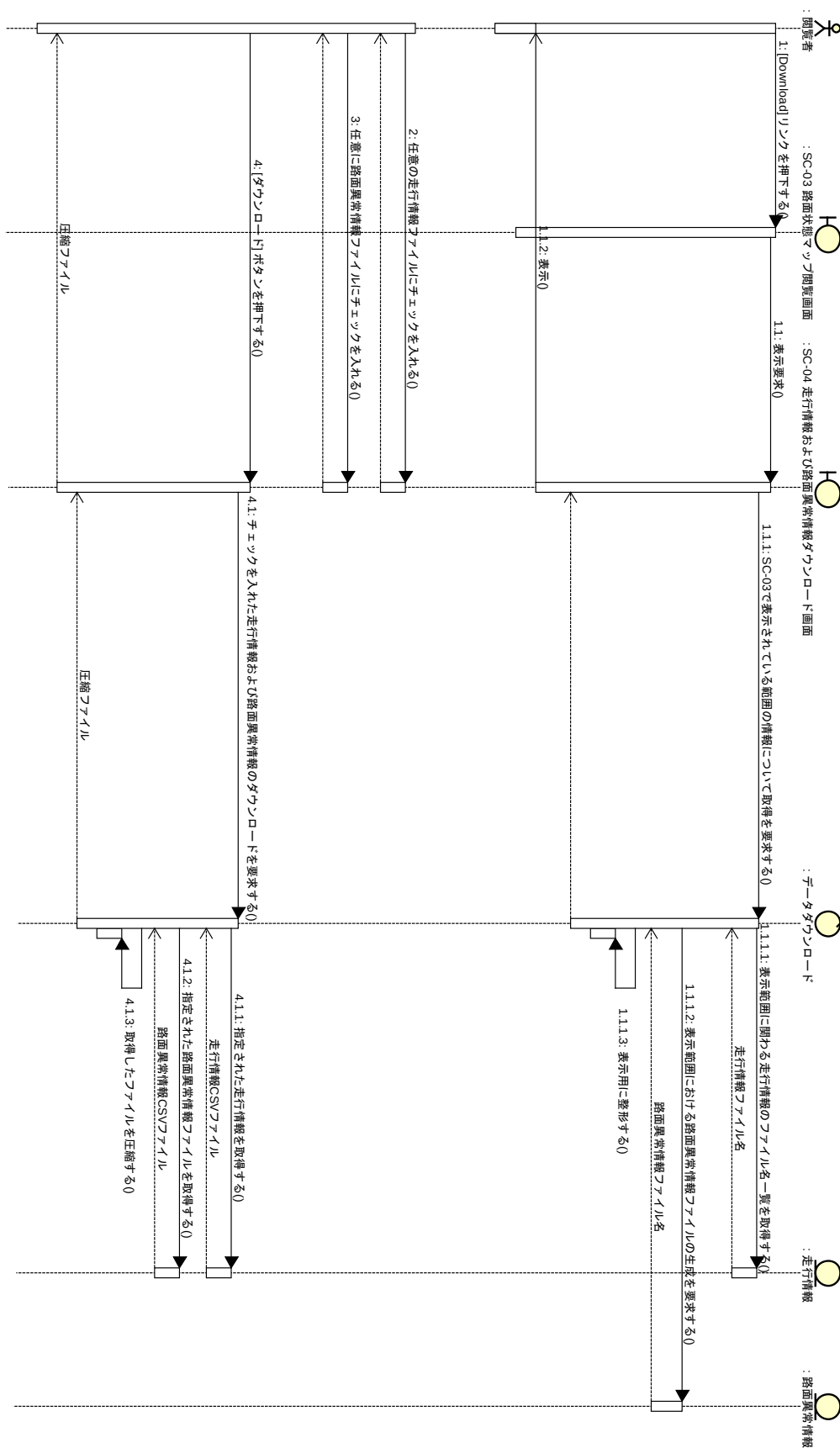


図 23：走行情報・路面異常情報ダウンロード機能 分析シーケンス図

5 画面設計

5.1 スマートフォン側

5.1.1 画面一覧

以下に、スマートフォンの画面一覧を示す。

- SC-01：路面状態マップ表示画面
- SC-02：メニュー画面
- SC-03：アプリケーション終了確認ポップアップ

5.1.2 画面詳細

路面状態マップ表示画面の詳細を次ページより示す。

画面 ID	SC-01
画面名	路面状態マップ表示画面



要素 ID	パラメータ名 (和)	パラメータ名 (英)	型	送受信属性		備考
				送信	受信	
1	路面状態マップ	map	地図	0	1	路面状態マップが表示される
2	路面異常点	anomaly-point	ポイント	0	1	路面異常箇所にプロットされる異常点
3	路面異常情報	anomaly-info	テキスト	0	1	選択された路面異常の詳細情報を表示する
4	地図拡大ボタン	expansion	ボタン	1	0	GoogleMap の地図を拡大する機能
5	地図縮小ボタン	reduction	ボタン	1	0	GoogleMap の地図を縮小する機能
6	稼働通知	notification	ラベル	0	1	稼働中の旨を表示する
7	メニュー	menu	ボタン	1	0	メニューを表示する
8	更新	update	メニュー	1	0	路面状態マップと更新時刻が更新される
9	終了	close	メニュー	1	0	閲覧・測定を終了し、アプリケーションを終了する
10	更新時刻	update-time	ラベル	0	1	最終更新時刻が表示される

端末のメニューボタンを押下した時の画面の詳細を以下に示す。

画面 ID	SC-02					
画面名	メニュー画面					
						
要素 ID	パラメータ名 (和)	パラメータ名 (英)	型	送受信属性		備考
				送信	受信	
1	更新ボタン	map	ボタン	1	0	路面状態マップが更新される
2	終了ボタン	anomaly-point	ボタン	1	0	測定を終了し、アプリを終了する

アプリケーション終了確認ポップアップの詳細を以下に示す。

画面 ID	SC-03
画面名	アプリケーション終了確認ポップアップ

要素 ID	パラメータ名 (和)	パラメータ名 (英)	型	送受信属性		備考
				送信	受信	
1	終了確認通知	close-notification	ラベル	0	1	
2	はい	yes	ボタン	1	0	ボタンを押下すると、走行情報をサーバに送信、ストレージから削除、アプリケーションの終了処理が行われる
3	いいえ	no	ボタン	1	0	画面 SC-01 に戻る

5.1.3 画面遷移

図 1 にアプリケーションの画面遷移図を示す。画面遷移図はアプリ起動から終了までを表している。

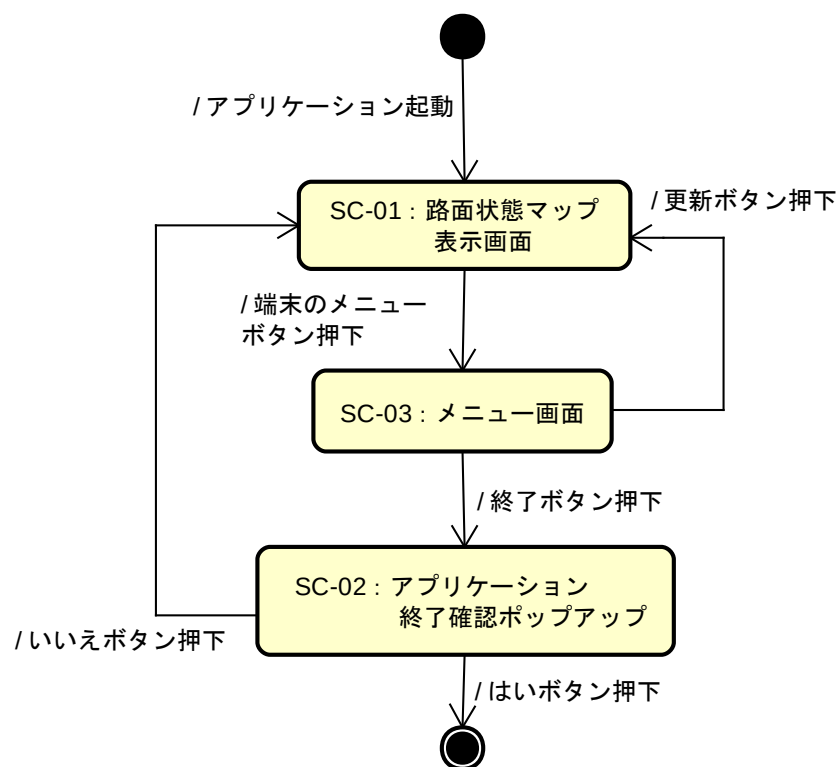


図 1：アプリケーション画面遷移図

5.2 クライアント PC 側

5.2.1 画面一覧

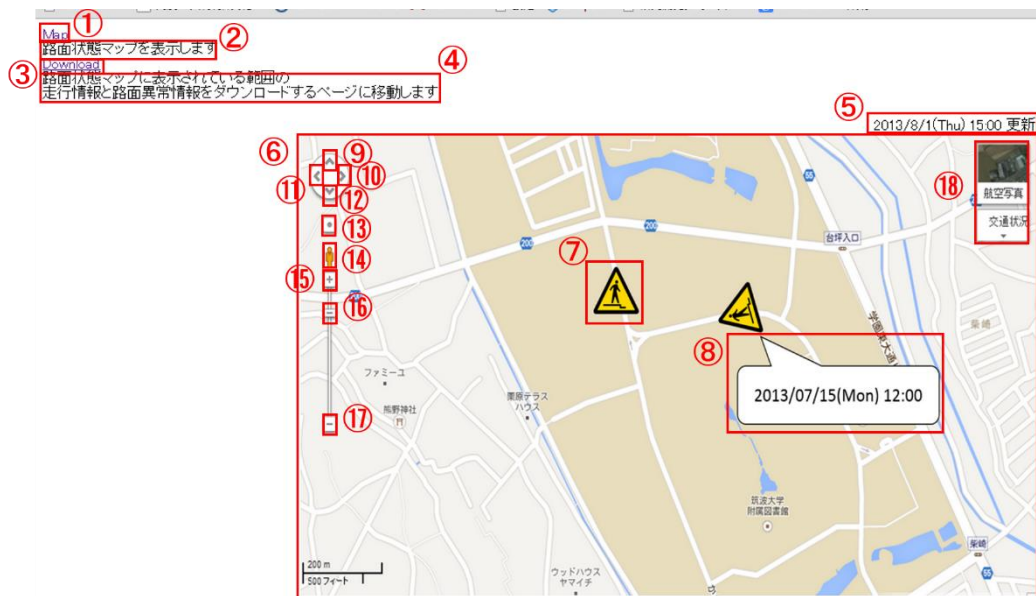
以下に、クライアント P C の画面一覧を示す。

- SC-04：路面状態マップ表示画面
- SC-05：走行情報および路面異常情報ダウンロード画面

5.2.2 画面詳細

路面状態マップ表示画面の詳細を次ページより示す。

画面 ID	SC-04
画面名	路面状態マップ閲覧画面



要素 ID	パラメータ名 (和)	パラメータ名 (英)	型	送受信属性		備考
				送信	受信	
1	マップ	map-link	リンク	1	0	SC-03 に遷移する（現在表示されている範囲を保持する）
2	マップ説明	map-detail	ラベル	0	0	
3	ダウンロード	download-link	リンク	1	0	SC-04 に遷移する（表示されている範囲を保持する）
4	ダウンロード説明	download-detail	ラベル	0	0	
5	更新時刻	update-time	ラベル	0	1	更新時刻が表示される
6	路面状態マップ	map	地図	0	1	路面状態マップが表示される
7	路面異常点	anomaly-point	ポイント	0	1	路面異常箇所にプロットされる異常点
8	路面異常情報	anomaly-info	ラベル	0	1	選択された路面異常の詳細情報を表示する
9	上ボタン	up-button	ボタン	1	0	GoogleMap の機能に準ずる
10	右ボタン	right-button	ボタン	1	0	GoogleMap の機能に準ずる
11	左ボタン	left-button	ボタン	1	0	GoogleMap の機能に準ずる
12	下ボタン	down-button	ボタン	1	0	GoogleMap の機能に準ずる
13	現在地表示	current-location	ボタン	1	0	GoogleMap の機能に準ずる
14	ストリートビュー	street-view	ボタン	1	0	GoogleMap の機能に準ずる
15	拡大ボタン	expansion	ボタン	1	0	GoogleMap の機能に準ずる

16	スクロール	scroll	ボタン	1	0	GoogleMap の機能に準ずる
17	縮小ボタン	reduction	ボタン	1	0	GoogleMap の機能に準ずる
18	地図種類変更	map-change	ボタン	1	0	GoogleMap の機能に準ずる

走行情報および路面異常情報ダウンロード画面の詳細を以下に示す。

画面 ID	SC-05
画面名	走行情報および路面異常情報ダウンロード画面

①

Map

②

③

Download

④

路面状態マップを表示します

路面状態マップに表示されている範囲の走行情報と路面異常情報をダウンロードするページに移動します

⑤	⑥ 走行情報	路面異常情報
☐	yyyyymmddTTMMSSsms_no.csv	☐ unusualData_yyyymmdd.csv
☐	yyyyymmddTTMMSSsms_no.csv	
☐	yyyyymmddTTMMSSsms_no.csv	
☐	yyyyymmddTTMMSSsms_no.csv	
☐	yyyyymmddTTMMSSsms_no.csv	
☐	yyyyymmddTTMMSSsms_no.csv	
☐	yyyyymmddTTMMSSsms_no.csv	
☐	yyyyymmddTTMMSSsms_no.csv	
☐	yyyyymmddTTMMSSsms_no.csv	
☐	yyyyymmddTTMMSSsms_no.csv	

⑦

⑧

選択したファイルをダウンロード

⑨

ダウンロード

要素 ID	パラメータ名 (和)	パラメータ名 (英)	型	送受信属性		備考
				送信	受信	
1	マップ	map-link	ボタン	1	0	SC-03 に遷移する（現在保持している範囲を保持する）
2	マップ説明	map-detail	ラベル	0	0	
3	ダウンロード	download-link	ボタン	1	0	SC-04 に遷移する（現在保持している範囲を保持する）
4	ダウンロード説明	download-detail	ラベル	0	0	
5	走行情報ダウンロードチェックボックス	drive-info-check	チェックボックス	1	0	
6	走行情報ファイル名	drive-info-filename	ラベル	0	1	地図に表示されている範囲を含む走行情報のファイル名
7	路面異常情報ダウンロードチェックボックス	anomaly-info-check	チェックボックス	1	0	
8	路面異常情報ファイル名	anomaly-info-filename	ラベル	0	1	地図上の路面異常情報のファイル名
9	一括ダウンロード	file-download	ボタン	1	0	チェックされている走行情報、路面異常情報がダウンロードされる

※ブラウザの戻る・進む・更新ボタンについて

ボタンの機能は使用できる。現在の範囲の情報を保持したまま画面遷移・更新を行う。また、SC-04 へのアクセス時は地球全体の地図を表示する（仮）。SC-05 に直接アクセスした場合は要素 ID5～8 は何も表示しない。

5.2.3 画面遷移

図2にクライアントPCの画面遷移図を示す。

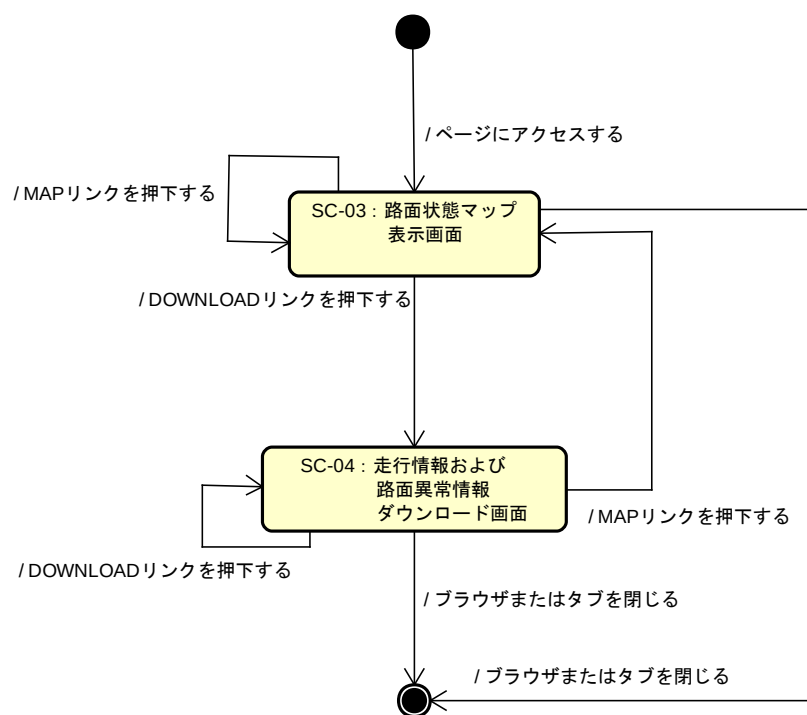


図2：クライアントPC画面遷移

6 ファイル定義

6.1 ファイル一覧

ファイル ID	ファイル名	説明	ファイル形式	存在場所
FL-01	走行情報ファイル	スマートフォンで取得した走行情報が記録されている。	csv	スマートフォン、WEB サーバ、データ解析サーバ

6.2 ファイル詳細

FL-01 走行情報ファイル

ファイル名

[西暦][月][日][時][分][秒][ミリ秒].csv

2013 年 9 月 1 日 12 時 34 分 56.012 秒に作成された走行情報ファイルの場合:20130901_123456012.csv

フォーマット

走行情報は csv 形式で記録し、上の行から時系列順に各データを格納する。データの並び順を図 24：走行情報ファイルのフォーマットに示す。

20130930_102229901	2013/9/30	12:23:06.135	36.0791	140.1087707	0.3405087	3.091819	9.752169	330	38.05542	1.3312019	1.2551605	-0.17909479
20130930_102229901	2013/9/30	12:23:06.160	36.0791	140.1087707	0.38136974	2.9964764	10.065437	330	38.05542	1.2566848	1.2620406	-0.1096901
20130930_102229901	2013/9/30	12:23:06.188	36.0791	140.1087707	0.46309182	2.9283748	9.384419	330	38.05542	1.2693121	1.2792472	-0.12659213
20130930_102229901	2013/9/30	12:23:06.211	36.0791	140.1087707	0.25878662	2.8875136	9.152874	330	38.05542	1.3009325	1.2647951	-0.15684071
20130930_102229901	2013/9/30	12:23:06.236	36.0791	140.1087707	0.24516626	3.050958	9.997335	330	38.05542	1.2318332	1.264053	-0.08938384
↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
ファイル名	日付	時刻	緯度	経度	X軸加速度	Y軸加速度	Z軸加速度	方角	速度	方位	仰角	ロール角

図 24：走行情報ファイルのフォーマット

データ名	数値範囲	特記事項など
ファイル名	00000000_0000000000~ 99991231_235959999	
日付	1/1/1~9999/12/31	
時刻	00:00:00.000~23:59:59.999	
緯度	-90.00000~90.00000	
経度	-180.00000000~180.00000000	
X 軸加速度	-100.000000000~ 100.000000000	変更の可能性あり
Y 軸加速度	-100.000000000~ 100.000000000	変更の可能性あり
Z 軸加速度	-100.000000000~ 100.000000000	変更の可能性あり
方角	0.0~360.0	北を 0 度として時計回りに 360 度
速度	0.0~999.9	単位は km/h 数値範囲は変更の可能性あり
方位	-180.00000000~180.00000000	
仰角	-180.00000000~180.00000000	
ロール角	-180.00000000~180.00000000	

7 データベース設計

7.1 テーブル一覧

- 走行情報テーブル
- 路面異常候補テーブル
- 路面異常テーブル
- 未処理走行情報テーブル

7.2 テーブル定義

別途資料”テーブル定義書”を参照。

テーブル番号	Table-01
論理テーブル名	走行情報
物理テーブル名	drive_data

No.	論理フィールド名	物理フィールド名	データ型	サイズ	初期値	主キー	NULL	備考
1	ファイル名	filename	varchar	18			not	yyyymmdd_hhmmssMMM
2	記録日	rec.date	date				not	
3	記録時間	rec.time	time(3)	3			not	
4	緯度	latitude	double				not	
5	経度	longitude	double				not	
6	加速度x	acc.x	float				not	
7	加速度y	acc.y	float				not	
8	加速度z	acc.z	float				not	
9	方向角	bearing	int	3			not	0～360の値が格納される
10	速度	speed	float				not	
11	方位	azimuth	float				not	
12	仰角	pitch	float				not	
13	ロール角	roll	float				not	

テーブル番号		Table-02
論理テーブル名	路面異常候補	物理テーブル名
		road_anomaly_con

No.	論理フィールド名	物理フィールド名	データ型	サイズ	初期値	主キー	NULL	備考
1	推定日	estimate_date	date				not	
2	推定時間	estimate_time	time(3)	3			not	
3	緯度	latitude	double				not	
4	経度	longitude	double				not	
5	方角	bearing	int	3			not	0～360の値が格納される
6	異常レベル	anomaly_level	int	2			not	

テーブル番号		Table-03
論理テーブル名	路面異常情報	物理テーブル名
		road_anomaly_data

No.	論理フィールド名	物理フィールド名	データ型	サイズ	初期値	主キー	NULL	備考
1	検出日	detection_date	date				not	
2	検出時間	detection_time	time(3)	3			not	
3	緯度インデックス	latitude_index	double				not	
4	経度インデックス	longitude_index	double				not	
5	緯度	latitude	double				not	
6	経度	longitude	double				not	
7	方角	bearing	int	3			not	0～360の値が格納される
8	異常レベル	anomaly_level	int	2			not	

テーブル番号	Table-04
論理テーブル名	未処理走行情報
物理テーブル名	unprocessed_data

No.	論理フィールド名	物理フィールド名	データ型	サイズ	初期値	主キー	NULL	備考
1	ファイル名	filename	varchar	18			not	yyyymmdd_hhmmssMMM
2	記録日	rec.date	date				not	
3	記録時間	rec.time	time(3)	3			not	
4	緯度	latitude	double				not	
5	経度	longitude	double				not	
6	加速度x	acc.x	float				not	
7	加速度y	acc.y	float				not	
8	加速度z	acc.z	float				not	
9	方向角	bearing	int	3			not	0～360の値が格納される
10	速度	speed	float				not	
11	方位	azimuth	float				not	
12	仰角	pitch	float				not	
13	ロール角	roll	float				not	

付録D システム評価フェーズにおける成果物

D.1 消費電力実験の実験計画書

実験計画書

2013/10/4

チームシュークリーム
枝松ちさと
胥徳文
林貴哉
村林竜司

概要

本書は、本プロジェクトがシステムを開発するにあたり、懸念事項となっているスマートフォンの消費電力についての実験を行うための計画書である。本計画書は、チームシュークリームが実験を行うために参照するものである。

実験の対象とするアプリケーション

本実験で用いるアプリケーションは、本プロジェクトで開発予定の走行情報収集アプリケーションとする。

アプリケーション要件

実験対象のアプリケーションでは、以下の要件が実装されている。

- 走行情報 CSV ファイルを作成する
- 走行情報を記録する
 - 「走行情報」として記録する項目：
ファイル名、取得日、取得時刻、緯度・経度、3 軸加速度、方角、速度、方位、仰角、ロール角、
3 軸地磁気、照度
- 走行情報 CSV ファイルを圧縮する

事前条件

本実験を行う際の事前条件は、下記の通りである。

- WiFi は OFF にし、3G 回線、もしくは LTE 回線下で実験を行うこと
- 実験と関係のないアプリケーションは可能な限り終了しておくこと

実験 1

実験概要

バックライトを常時点灯させた状態、かつ GPS 機能を用いないで走行情報を記録する。

実験条件

実験 1 における条件は下記の通りである。

- センシングの頻度は 100Hz とする
- GPS の機能は OFF にしておくものとする
- アプリケーション起動中はバックライトを 常時点灯 させ、意図的に画面を消灯させるなどの操作は行わないものとする

実験結果として記録する内容

実験結果としては、下記の内容を記録すること。

- 実験端末
- 実験前の電池残量 (%)
- 実験開始時刻
- 実験終了時刻
- 実験後の電池残量 (%)

なお、特記事項がある場合には適宜記しても良い。

実験手順

1. アプリケーション起動前にスマートフォンの電池残量を確認し、実験結果記録用紙に記録する
2. アプリケーションを起動し、走行情報の記録を開始する
3. スマートフォンを静止させた状態で 1 時間置いておく
4. 1 時間経過後、「Stop」ボタンを押下し走行情報の記録を停止する
5. アプリケーションを終了し、スマートフォンの電池残量を実験結果記録用紙に記録する

実験 2

実験概要

バックライトを常時点灯させた状態、かつ GPS を用いて走行情報を記録する。

実験条件

実験 2 における条件は下記の通りである。

- センシングの頻度は 100Hz とする
- GPS の機能は ON にしておくものとする
- アプリケーション起動中はバックライトを 常時点灯 させ、意図的に画面を消灯させるなどの操作は行わないものとする

実験結果として記録する内容

実験結果としては、下記の内容を記録すること。

- 実験端末
- 実験前の電池残量 (%)
- 実験開始時刻
- 実験終了時刻
- 実験後の電池残量 (%)

なお、特記事項がある場合には適宜記しても良い。

実験手順

1. アプリケーション起動前にスマートフォンの電池残量を確認し、実験結果記録用紙に記録する
2. アプリケーションを起動し、走行情報の記録を開始する
3. スマートフォンを静止させた状態で 1 時間置いておく
4. 1 時間経過後、「Stop」ボタンを押下し走行情報の記録を停止する
5. アプリケーションを終了し、スマートフォンの電池残量を実験結果記録用紙に記録する

実験 3

実験概要

バックライトを消灯させた状態、かつ GPS を用いて走行情報を記録する。

実験条件

実験 3 における条件は下記の通りである。

- センシングの頻度は 100Hz とする
- GPS の機能は ON にしておくものとする
- アプリケーション起動中はバックライトを 消灯 させ、意図的に画面を点灯させるなどの操作は行わないものとする

実験結果として記録する内容

実験結果としては、下記の内容を記録すること。

- 実験端末
- 実験前の電池残量 (%)
- 実験開始時刻
- 実験終了時刻
- 実験後の電池残量 (%)

なお、特記事項がある場合には適宜記しても良い。

実験手順

1. アプリケーション起動前にスマートフォンの電池残量を実験結果記録用紙に記録する
2. アプリケーションを起動し、スピナーから「100」を選択する
3. 「計測開始」ボタンを押下し、走行情報の記録を開始する
4. 画面のバックライトを消灯させ、スマートフォンを静止させた状態で 1 時間置いておく
5. 1 時間経過後、画面を点灯させ、「計測停止」ボタンを押下して走行情報の記録を停止する
6. アプリケーションを終了し、スマートフォンの電池残量を実験結果記録用紙に記録する

実験 4

実験概要

バックライトを消灯させた状態、かつ GPS を用いて走行情報を記録する。

実験条件

実験 4 における条件は下記の通りである。

- センシングの頻度は 40Hz とする
- GPS の機能は ON にしておくものとする
- アプリケーション起動中はバックライトを 消灯 させ、意図的に画面を点灯させるなどの操作は行わないものとする

実験結果として記録する内容

実験結果としては、下記の内容を記録すること。

- 実験端末
- 実験前の電池残量 (%)
- 実験開始時刻
- 実験終了時刻
- 実験後の電池残量 (%)

なお、特記事項がある場合には適宜記しても良い。

実験手順

1. アプリケーション起動前にスマートフォンの電池残量実験結果記録用紙に記録する
2. アプリケーションを起動し、スピナーから「40」を選択する
3. 「計測開始」ボタンを押下し、走行情報の記録を開始する
4. 画面のバックライトを消灯させ、スマートフォンを静止させた状態で 1 時間置いておく
5. 1 時間経過後、画面を点灯させ、「計測停止」ボタンを押下して走行情報の記録を停止する
6. アプリケーションを終了し、スマートフォンの電池残量を実験結果記録用紙に記録する

補助実験 2-1

実験概要

バックライトを常時点灯させた状態、かつ GPS を用いて走行情報を記録する。

実験条件

実験 2 における条件は下記の通りである。

- センシングの頻度は 100Hz とする
- GPS の機能は ON にしておくものとする
- アプリケーション起動中はバックライトを 常時点灯 させ、意図的に画面を消灯させるなどの操作は行わないものとする

実験結果として記録する内容

実験結果としては、下記の内容を記録すること。

- 実験前の電池残量 (%)
- 実験開始時刻
- 実験終了時刻
- 実験後の電池残量 (%)

なお、特記事項がある場合には適宜記しても良い。

実験手順

6. アプリケーション起動前にスマートフォンの電池残量を確認し、実験結果記録用紙に記録する
7. アプリケーションを起動し、走行情報の記録を開始する
8. スマートフォンを移動させる状態で 1 時間置いておく
9. 1 時間経過後、「Stop」ボタンを押下し走行情報の記録を停止する
10. アプリケーションを終了し、スマートフォンの電池残量を実験結果記録用紙に記録する

補助実験 4-1

実験概要

バックライトを消灯させた状態、かつ GPS を用いて走行情報を記録する。

実験条件

実験 4 における条件は下記の通りである。

- センシングの頻度は 20Hz とする
- GPS の機能は ON にしておくものとする
- アプリケーション起動中はバックライトを 消灯 させ、意図的に画面を点灯させるなどの操作は行わないものとする

実験結果として記録する内容

実験結果としては、下記の内容を記録すること。

- 実験端末
- 実験前の電池残量 (%)
- 実験開始時刻
- 実験終了時刻
- 実験後の電池残量 (%)

なお、特記事項がある場合には適宜記しても良い。

実験手順

1. アプリケーション起動前にスマートフォンの電池残量を確認し、実験結果記録用紙に記録する
2. アプリケーションを起動し、スピナーから「20」を選択する
3. 「計測開始」ボタンを押下し、走行情報の記録を開始する
4. 画面のバックライトを消灯させ、スマートフォンを静止させた状態で 1 時間置いておく
5. 1 時間経過後、画面を点灯させ、「計測停止」ボタンを押下して走行情報の記録を停止する
6. アプリケーションを終了し、スマートフォンの電池残量を実験結果記録用紙に記録する

補助実験 4-2

実験概要

バックライトを消灯させた状態、かつ GPS を用いて走行情報を記録する。

実験条件

実験 4 における条件は下記の通りである。

- センシングの頻度は 60Hz とする
- GPS の機能は ON にしておくものとする
- アプリケーション起動中はバックライトを 消灯 させ、意図的に画面を点灯させるなどの操作は行わないものとする

実験結果として記録する内容

実験結果としては、下記の内容を記録すること。

- 実験端末
- 実験前の電池残量 (%)
- 実験開始時刻
- 実験終了時刻
- 実験後の電池残量 (%)

なお、特記事項がある場合には適宜記しても良い。

実験手順

1. アプリケーション起動前にスマートフォンの電池残量を確認し、実験結果記録用紙に記録する
2. アプリケーションを起動し、スピナーから「60」を選択する
3. 「計測開始」ボタンを押下し、走行情報の記録を開始する
4. 画面のバックライトを消灯させ、スマートフォンを静止させた状態で 1 時間置いておく
5. 1 時間経過後、画面を点灯させ、「計測停止」ボタンを押下して走行情報の記録を停止する
6. アプリケーションを終了し、スマートフォンの電池残量を実験結果記録用紙に記録する

補助実験 4-3

実験概要

バックライトを消灯させた状態、かつ GPS を用いて走行情報を記録する。

実験条件

実験 4 における条件は下記の通りである。

- センシングの頻度は 80Hz とする
- GPS の機能は ON にしておくものとする
- アプリケーション起動中はバックライトを 消灯 させ、意図的に画面を点灯させるなどの操作は行わないものとする

実験結果として記録する内容

実験結果としては、下記の内容を記録すること。

- 実験端末
- 実験前の電池残量 (%)
- 実験開始時刻
- 実験終了時刻
- 実験後の電池残量 (%)

なお、特記事項がある場合には適宜記しても良い。

実験手順

1. アプリケーション起動前にスマートフォンの電池残量を確認し、実験結果記録用紙に記録する
2. アプリケーションを起動し、スピナーから「80」を選択する
3. 「計測開始」ボタンを押下し、走行情報の記録を開始する
4. 画面のバックライトを消灯させ、スマートフォンを静止させた状態で 1 時間置いておく
5. 1 時間経過後、画面を点灯させ、「計測停止」ボタンを押下して走行情報の記録を停止する
6. アプリケーションを終了し、スマートフォンの電池残量を実験結果記録用紙に記録する