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

情報通信技術を利活用した
農作物向け獣害防止システムの開発
―Webアプリケーションのテストと
ユーザビリティの向上―

浜野悠介

修士（工学）

（コンピュータサイエンス専攻）

指導教員　高橋伸

2015年3月

概要

獣害は近年、全国的な問題となっている。三重県鈴鹿市においても例外ではなく、毎年多くの獣害を被っている。JA 鈴鹿ではこれら獣害の対策を行っており、その中でも発信機を活用したサルの生息調査と組織的な追い払いが最も効果的な方法とされている。しかし、受信機の運用者が必要であること、受信機の使用は経験則に基づいて行われており属人性が高いこと、サルの位置情報の即時的な通知方法がないことから効果的に追い払いが実施できていない現状がある。そこで本プロジェクトでは JA 鈴鹿を顧客とし、これらの問題を解決する獣害防止システムの開発を行った。筆者はシステムのソリューションの一つである Web アプリケーションのテストとユーザビリティの向上を担当した。Web アプリケーションのテストでは、テスト計画を立案するとともに、テストの十分性を確保するための方法を検討した。ユーザビリティの向上では、ユーザビリティの正確な評価だけでなく、その評価から問題点を抽出し、改善を行うことでユーザビリティの向上を実現した。

目 次

第 1 章　はじめに　1

第 2 章　プロジェクトの概要　3
2.1　獣害の現状 3
2.2　既存の対策とその問題点 4
2.3　プロジェクトの目的 6
2.4　提案内容 7
2.5　システム概要 7
2.5.1　システム全体の流れ 7
2.5.2　観測機器 8
2.5.3　Web アプリケーション 11
2.6　システム導入による効果 12
2.7　プロジェクト体制 13

第 3 章　Web アプリケーションのテスト　16
3.1　テスト計画の立案 16
3.1.1　単体テスト 16
3.1.2　結合テスト 18
3.1.3　受け入れテスト 19
3.1.4　バグ管理方法 19
3.2　テスト自動化ツールの導入方法 20
3.2.1　PHPUnit 20
3.2.2　Selenium 21
3.3　テスト十分性 21
3.3.1　単体テストの十分性確保 22
3.3.2　結合テストの十分性確保 23

第 4 章　ユーザビリティの向上　26
4.1　品質とユーザビリティの関係 26
4.2　ユーザビリティ評価手法 27
4.2.1　ユーザビリティ・インスペクション 27
4.2.2　ユーザビリティ・テスティング 29

第 5 章　Web アプリケーションのテストとユーザビリティの向上における評価と考察　32
5.1　Web アプリケーションのテストにおける評価 32
5.1.1　結果 32
5.1.2　考察 33
5.2　ユーザビリティ向上における評価 34
5.2.1　結果 34
5.2.2　考察 38

第 6 章　おわりに　41

謝辞　43

参考文献　44

図 目 次

2.1 野生鳥獣による農作物被害金額の推移 (農林水産省より引用)[1] 3
2.2 年齢別狩猟免許所持者数の推移 [3] 4
2.3 サルの位置捕捉から追い払いまでの流れ 6
2.4 システム全体の流れ 8
2.5 観測機器の処理の流れ 9
2.6 Web アプリケーションの処理の流れ 12
2.7 ステークホルダ 13
2.8 プロジェクトスケジュール 15

3.1 テストの流れ 17
3.2 Selenium IDE の動作例 21
3.3 カバレッジ測定結果の一例 23
3.4 FV 表の一部 24
3.5 利用者登録・編集画面のデシジョンテーブル 24

5.1 修正した観測箱登録画面のスクリーンショット 38
5.2 修正したエラーメッセージ表示のスクリーンショット 39
5.3 修正したメニューバーのスクリーンショット 39
5.4 掲示板投稿画面のスクリーンショット 40

第1章　はじめに

本書は、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ソフトウェア開発専修プログラムにおける研究開発プロジェクトの成果をまとめた特定課題研究報告書である。

獣害は近年、全国的な問題となっている。農林水産省による調査では平成 24 年度の全国の獣類による農作物被害額は約 230 億円である。調査を始めた平成 11 年度から農作物被害額は年々増加しており、今後も増加することが予想される。被害の中でもサル、イノシシ、シカによるものが全体の 7 割を占めている。こういった獣害が深刻化している要因として、獣類の生息域の拡大、猟師人口の減少や高齢化などが考えられる。さらに、獣害は営農意欲の減退、耕作放棄地の増加などをもたらし、報告されている被害額以上の損失がある。農林水産省ではこういった獣害の防止対策を行っている地域に対して鳥獣被害防止総合対策交付金を割り当てることによって支援を行っている。平成 26 年度の交付金の予算は 95 億円である。具体的な支援内容としては捕獲機材の導入、捕獲に関する専門家の育成支援、捕獲技術高度化施設の建設などがある。

三重県鈴鹿市の獣害調査を行っている JA 鈴鹿においても様々な獣害対策を行っており、防護柵設置による獣類の侵入防止、捕獲による個体数の管理と調整、発信機を活用したサルの生息調査と組織的な追い払いなどの対策が実施されている。この中で、最も効果的な方法は発信機を活用したサルの生息調査と組織的な追い払いである。これはサルに取り付けられた首輪型のテレメトリ発信機からの電波を受信することでサルの場所を捕捉し、生息調査と組織的な追い払いを可能にする。生息調査を行うことによりサルの行動域を把握し、群れの行動特性をつかむことができる。行動特性が分かることでサルの高い学習能力を逆手に取った組織的な追い払いが可能となり、農作物被害の減少に有効とされている。しかし、この対策においても JA 鈴鹿では以下のような問題を抱えている。

- 受信機の運用者が必要である

  発信機の電波を受信して、サルの位置を捕捉するために受信機の運用者はサルに狙われやすい農地周辺を巡回する必要がある。運用者は主に NPO 法人「サルどこネット」の会員であり、平均年齢が高い。よって受信機を携帯して長時間巡回することは運用者の体への負担が大きい。

- 受信機の使用は属人性が高い

  サルの位置は受信機を向けている方向と受信機から聞こえてくる発信機からの信号を受信した音の大きさから捕捉している。また、異なる場所にある複数の受信機の方向や音

の連携により捕捉の精度を向上させている。しかし、このように受信機の使用には訓練が必要であり属人性が高い。

上記の問題を解決するために、本プロジェクトでは以下の 2 つのソリューションを併用した、農作物向け獣害防止システムを開発した。

- 一定距離内へのサルの侵入を自動で検知する観測機器

  発信機からの電波を受信機が検知したかどうかを自動判別できる。また、観測機器データを定期的にサーバへ送信する。

- サルの位置情報を地図上に表示し、サルの検知情報を管理する Web アプリケーション

  観測機器から送信されたデータを用いることで、サルの場所を捕捉し次にサルが出没する確率が高い場所を表示する。また、農家や観測機器の情報を管理することができる。

このシステムを用いることでサルの場所を即時的に知ることができる。さらに、サルの農地への接近を電子メールで警告することで、より効率的に組織的な追い払いが可能となる。

本プロジェクトの最終目的はサルによる農作物被害を上記システムを用いることで減少させることである。しかし、これらを確認するためには長期間のシステム運用が必要となる。そのため、本年度では長期間のシステム運用が可能となるような機能の実装を目的とする。

筆者による本プロジェクトへの貢献の一つは Web アプリケーションのテストである。Web アプリケーションのテスト、つまりソフトウェアテストは仕様書に沿った動作をすることの検証を行う。テストを行うことにより、実装した機能が意図した挙動であるか、システムが予期せぬ動作をしないかなどの確認をすることができる。こうすることで、仕様漏れの防止やバグが存在しないシステムを顧客に納品することができる。このようなソフトウェアテストを効率化するために様々なテスト手法が考案されている。本プロジェクトのテスト計画にいくつかのテスト手法を導入し、実施する。導入するテスト手法はテスト自動化ツールによるテストの効率化、ホワイトボックステスト及びブラックボックステストによるテスト十分性の確保である。

もう一つの筆者による本プロジェクトへの貢献はユーザビリティの向上である。近年の情報処理機器の急速な普及に伴い、ユーザビリティの重要性が高まっている。本プロジェクトにおいても Web アプリケーションの開発を行うため、ユーザビリティの向上は必要不可欠である。本プロジェクトではユーザビリティ・インスペクション及びユーザビリティ・テスティングを用いることでユーザビリティの評価を行う。このような手法を用いてユーザビリティを正確に評価することにより、的確に改善すべき部分を発見することができる。よって Web アプリケーション全体のユーザビリティの向上が見込める。

本書は全 6 章で構成されている。第 2 章では本プロジェクトの概要について述べる。第 3 章では Web アプリケーションのテストについて述べる。第 4 章ではユーザビリティの向上について述べる。第 5 章では第 3 章、第 4 章で述べた部分の評価と考察を述べる。第 6 章ではまとめを述べる。また、付録として本システムの設計資料やプロジェクト管理の資料に加えて、Web アプリケーションのテストとユーザビリティ評価に関する資料を添付する。

第2章　プロジェクトの概要

2.1　獣害の現状

近年、全国的に獣害が深刻化している。農林水産省の調査によると、野生鳥獣による全国の農作物被害額は調査を始めた平成 11 年度から年々増加し、平成 24 年度には約 230 億円に及ぶ。図 2.1 に平成 18 年度から 24 年度までの野生鳥獣による農作物被害額の推移を示す [1]。

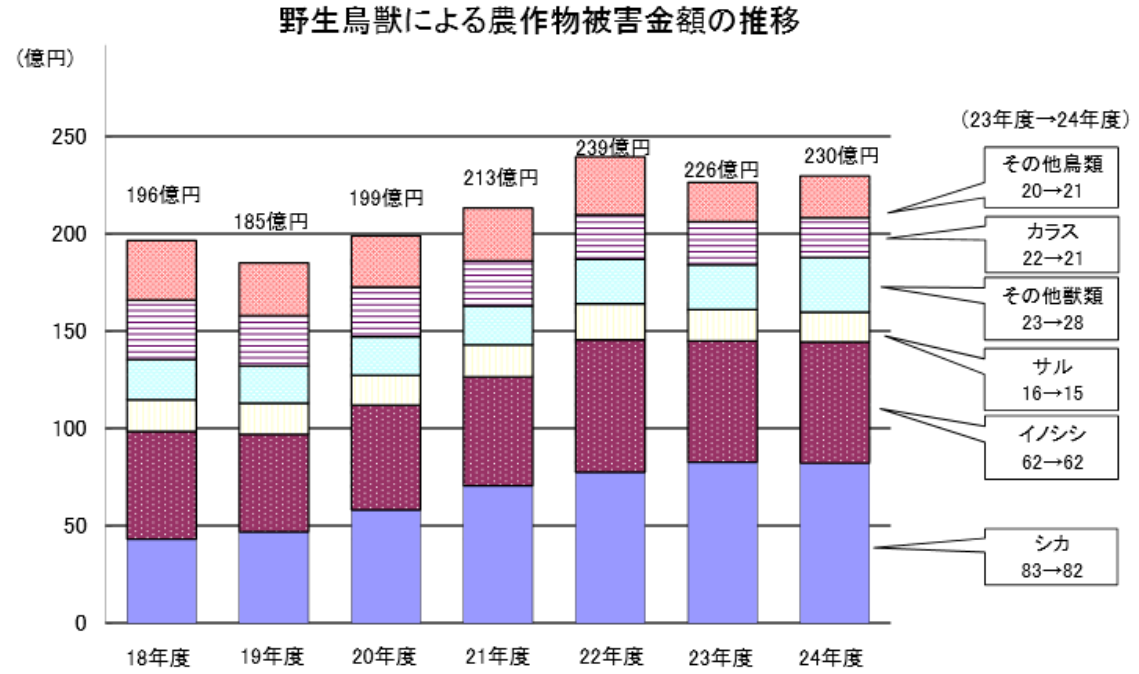


図 2.1: 野生鳥獣による農作物被害金額の推移 (農林水産省より引用)[1]

さらに、獣害は営農意欲の減退、耕作放棄地の増加等をもたらし、報告されている被害額以上の損失がある。このように獣害が深刻化している要因としては、獣類の生息域の拡大 [2]、図 2.2 に示した猟師人口の減少と高齢化 [3] などが考えられる。これらの獣害のうち、全体の 7 割がサル、シカ、イノシシによるものである。

本プロジェクトの顧客である JA 鈴鹿は三重県鈴鹿市の獣害調査を行っている。鈴鹿市においても獣害は深刻化しており、平成 22 年度にはサル、シカ、イノシシによる被害が約 3000 万円にも及んでいる [4]。これらの獣害を軽減させるために様々な獣害防止対策を行っている

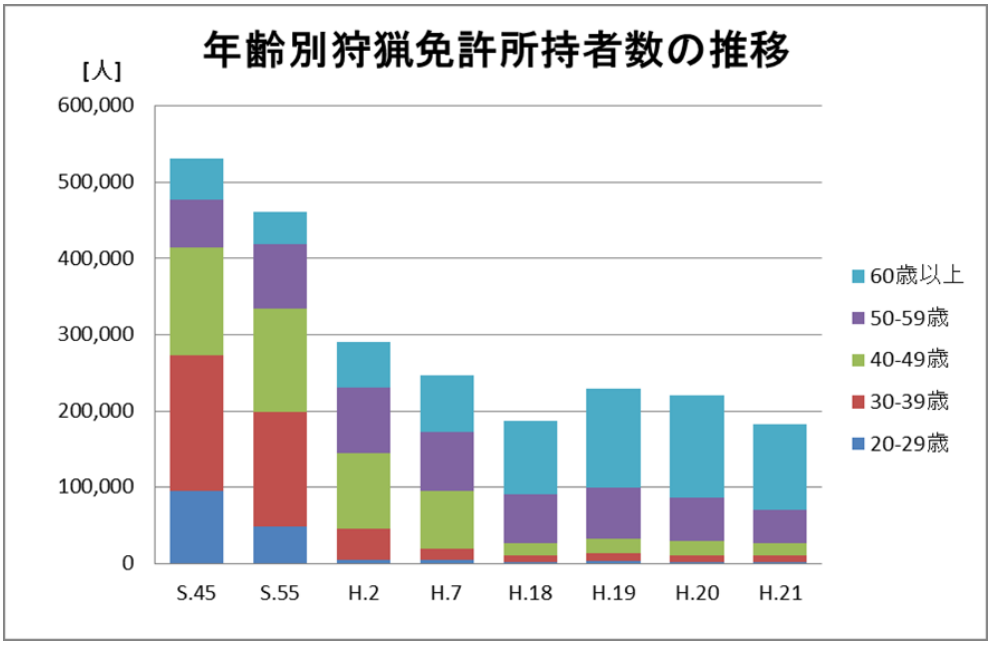


図 2.2: 年齢別狩猟免許所持者数の推移 [3]

が、効果が薄いというのが現状である。

2.2　既存の対策とその問題点

農林水産省では獣害防止対策を行っている地域に対して鳥獣被害防止総合対策交付金を割り当てることによって支援を行っている。平成 26 年度の交付金の予算は 95 億円である [5]。さらに、効果的な防止対策を周知させるために様々なガイドラインを掲示している [6][7]。また、三重県内の農業技術の中枢機関である三重県農業研究所においても獣害対策研究がなされており、獣害対策のガイドラインを掲示している [8][9][10]。

JA 鈴鹿においてもこれらのガイドラインを用いて以下のような獣害対策を実施している。

- 防護柵設置による獣類の侵入防止
- 捕獲による個体数の管理と調整
- 発信機を活用したサルの生息調査と組織的な追い払い

(1) 防護柵設置による獣類の侵入防止

農地の周囲を防護柵で囲うことによって獣類を農地へ侵入させないようにすることができる。防護柵の種類は大きく物理柵と心理柵に分けることができる。物理柵は強度を持たせた

柵の高さにより、動物の出入りをコントロールする柵である。心理柵は学習によって柵を回避するようになった獣類の心理を利用した防護策であり、電気柵が一般的である。しかし、防護柵は設置に多額の費用が掛かり、設置後の定期的なメンテナンスが必要である。また、防護柵を設置したとしても獣類が防護柵で保護された農地を避け、防護柵がない農地へ向かうことで農作物被害が生じてしまう。以上のことからこの対策は根本的な獣害対策にはならない。

(2) 捕獲による個体数の管理と調整

獣類の捕獲による個体数管理を行うことで、獣害の軽減と地域個体群の長期にわたる安定的な維持を図ることができ、個体数、生息密度、分布域又は群れの構造などを適切に管理することができる [11]。こういった個体数管理を効果的に行うことにより、獣害防止につながる。しかし、2.1 節で述べたように捕獲を行う猟師人口の減少と高齢化が進んでいることに加えて、獣類の個体数の増加などの理由により、猟師への負担が非常に大きくなっており、個体数管理を効果的に行うことは現状難しい。

(3) 発信機を活用したサルの生息調査と組織的な追い払い

獣害を多く引き起こしている獣類の内の一種であるサルは以下のような特徴がある。

- 高い学習能力をもつ
- 数十頭の群れで活動する

これらの特徴を考慮して多くの獣害対策ガイドラインでは、サルに対して生息調査と組織的な追い払いを行うことが推奨されている。生息調査を行う理由はサルの行動域を把握し、群れの行動特性をつかむことができるからである。さらに、行動特性をつかむことでサルに対して繰り返して追い払いすることが可能となり、サルに人間の恐怖を覚え込ませることができる。これらの結果として、サルの行動域を集落から山へ追い上げることが可能となる。

こういった生息調査と組織的な追い払いを効果的に行うためには、サルの位置を捕捉する必要がある。本プロジェクトの顧客である JA 鈴鹿では、NPO 法人「サルどこネット」[12] と協力してサルの位置捕捉を行っている。図 2.3 にサルの位置捕捉から追い払いまでの流れを示す。サルはエサを求めて山から人里に降りてくる (図 2.3(1))。サルどこネット会員は定期的にサルに狙われやすい農地周辺を巡回しサルの位置を捕捉する。具体的には、サルが群れで行動していることを利用し、群れの中心的なサルに首輪型のテレメトリ発信機を取り付け、発信されている電波を受信機を用いて探ることによって捕捉している (図 2.3(2))。サルどこネット会員は捕捉情報をサルどこネットが運営している Web サイト「サルどこ位置情報掲示板」に投稿する (図 2.3(3))。一般の農家や JA 職員は投稿された情報をスマートフォンや PC から確認し、自分の担当区域の近くにサルが現れたことを確認したら、現地に出向いて追い払いを行う。

しかし、この対策においても鈴鹿市では以下のような問題を抱えている。

図 2.3: サルの位置捕捉から追い払いまでの流れ

1. 受信機の運用者が必要であり、サルの位置捕捉するプロセスが自動化できていない

上で述べたように、サルの位置を捕捉するために受信機の運用者は定期的にサルに狙われやすい農地周辺を巡回する必要がある。そのため、24 時間常にサルに取り付けられたテレメトリ発信機の電波を監視することができない点と運用者の高齢化に伴い受信機を持った巡回が運用者にとって大きな負担になる点について問題がある。

2. 受信機を用いたサルの位置捕捉は属人性が高い

サルの位置は受信機を向けている方向と受信機から聞こえてくる発信機からの信号を受信した音の大きさから捕捉している。また、異なる場所にある複数の受信機の方向や音の連携により位置捕捉の精度を向上させている。このように受信機の使用には訓練が必要であり属人性が高いことが問題である。

このように、様々な獣害対策が行われているが、上で述べたような問題が多数存在しているのが現状である。

2.3 プロジェクトの目的

本プロジェクトは三重県鈴鹿市の農業を管轄している JA 鈴鹿を顧客として、鈴鹿市の獣害の削減を支援することを目的としている。鈴鹿市では 2.1 節で述べたように獣害被害が深刻化

しており、2.2 節で述べたような対策を行っているが、それぞれ問題を抱えており、効率的に対策を実施できていない。本プロジェクトでは、その中でも「発信機を活用したサルの生息調査と組織的な追い払い」に存在する問題を解決する ICT ソリューションを提供し、生息調査と組織的な追い払いの効率的な実施を本年度の目的とする。

2.4 提案内容

サルの生息調査と組織的な追い払いを支援するために、以下の 4 つの方法を提案した。

1. サルの位置捕捉プロセスの自動化

サルに取り付けられている発信機からの電波を受信する観測機器を山や農地周辺に設置する。観測機器は受信音を音解析することによって、サルが観測機器周辺にいるかどうかを自動判別する。また、検知情報を定期的にサーバに送信することで、サルの捕捉情報を管理することができる。これにより、受信機を持った運用者の巡回による負担を軽減させることができる。

2. 複数の観測機器連携

複数の観測機器を観測範囲が重なるように配置することによって、サルの位置捕捉の精度を向上させる。これにより、受信機を用いたサルの位置捕捉の属人性が高いことから位置捕捉の精度の向上が難しいという問題を解決することができる。

3. サルの位置捕捉情報を用いた生息調査

観測機器からサーバに送られたサルの位置捕捉情報を蓄積することにより、その蓄積データとサルの行動特性 [13] からサルの動向を予測する。これにより、農地が獣害に遭う前の追い払いや効率的な生息調査が行える。

4. サルの位置捕捉情報の即時的な通知

観測機器がサルに取り付けられている発信機からの電波を検知した場合に、検知地域周辺の農地の所有者や猟師に即時的に通知を行う。これにより、サルの検知とほぼ同時に関係者へ連絡が可能となるため、従来よりも早い段階で追い払いを始めることができる。

2.5 システム概要

2.5.1 システム全体の流れ

図 2.4 にシステム全体の流れを示す。サルに取り付けられている発信機からの電波を周辺にある観測機器が検知した場合、検知情報をサーバに送信する。また、検知していない状態でも、観測機器が正常に動作しているかを管理するために定期的に動作情報を送信する (図 2.4(1))。送信された情報はサーバ内で処理され、Web アプリケーション上で確認することができる。さらに、サルを観測機器が検知した場合、検知地域周辺の農地の所有者などに電子

メールを送付することにより、警告する (図 2.4(2))。これらのシステムを利用するためにシステムの管理者である JA 鈴鹿の職員は、事前に Web アプリケーションから農地所有者の情報や農作物の情報、サル群の情報などを登録する (図 2.4(3))。

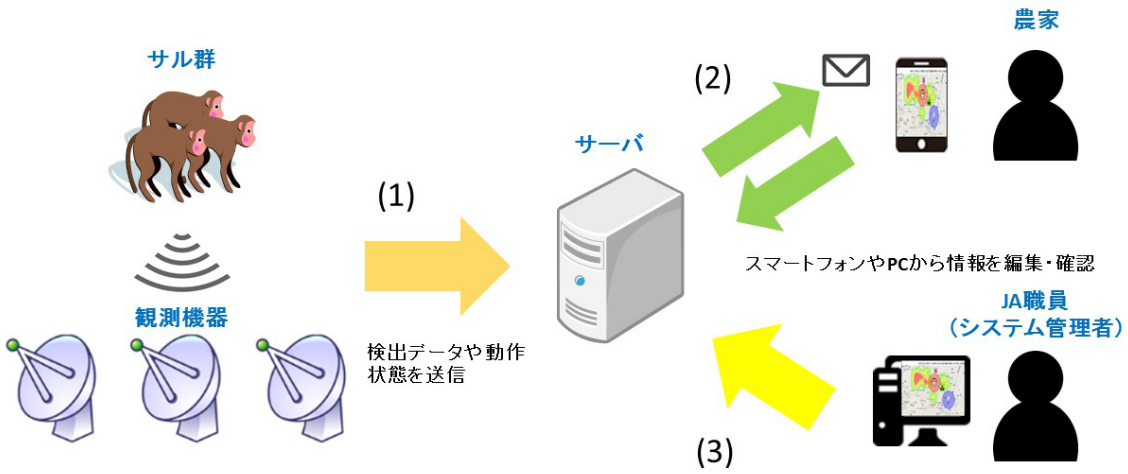


図 2.4: システム全体の流れ

2.5.2 観測機器

図 2.5 に観測機器の処理の流れを示す。観測機器は、受信モジュール、解析モジュール、データ通信端末によって構成されている。受信モジュールはアンテナと受信機から構成されており、アンテナから受信した信号を受信機によって音信号に変換している (図 2.5(1))。シングルボードコンピュータでは受信モジュールからの音信号を解析し、音信号の中に発信機からの信号が含まれているかを判別している (図 2.5(2))。その判別結果を定期的にデータ通信端末によってサーバへ送信する (図 2.5(3))。

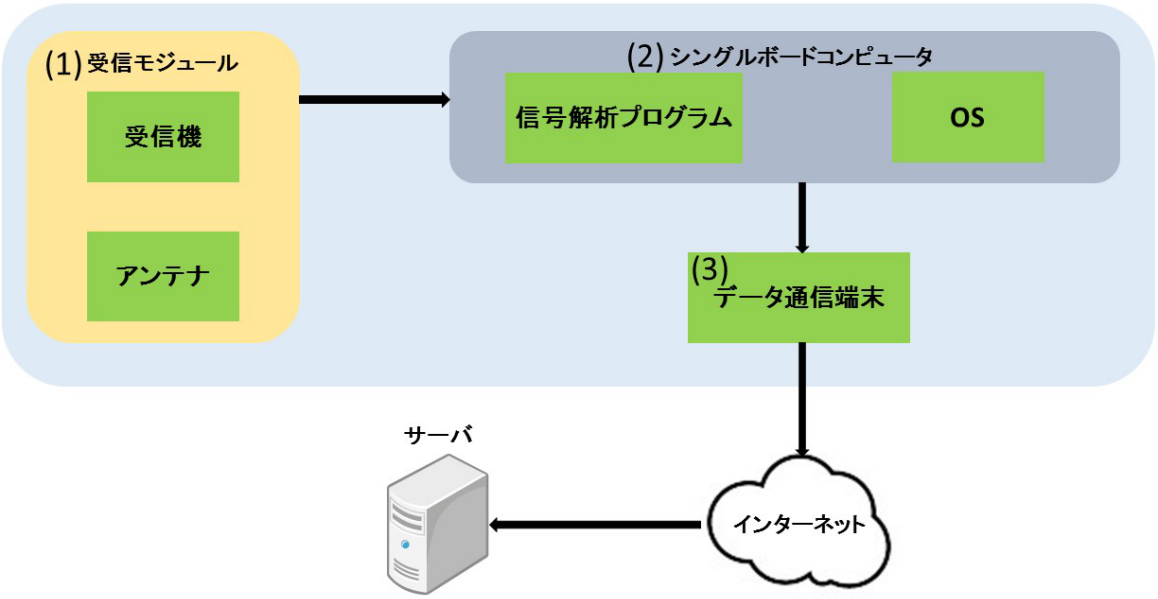


図 2.5: 観測機器の処理の流れ

表 2.1 に観測機器からサーバへ送信するパラメータを示す。

表 2.1: 観測機器からサーバへ送信するパラメータ

変数名	型	概要
box_id	int	観測機器の識別 ID
monkey_id	int	検知したサル群の識別 ID

表 2.2 に観測機器に用いたハードウェア構成を示し、表 2.3 にシングルボードコンピュータとして用いた RaspberryPi のソフトウェア構成を示す。

表 2.2: 観測機器のハードウェア構成

機器	型番	メーカー
広域帯ハンディレシーバー (受信機)	IC-R6	アイコム株式会社
八木アンテナ	NY144Z	株式会社ナテック
Wi-Fi USB アダプタ	GW-USNano2-G	プラネックスコミュニケーションズ株式会社
モバイルルータ	GL06P	HUAWEI
モバイルバッテリー	G-B201W	GSK
Raspberry Pi	Type B	Raspberry Pi
USB オーディオ変換ケーブル	BSHSAU01BK	株式会社バッファロー

表 2.3: RaspberryPi のソフトウェア構成

種類	名称
信号解析プログラム	C 言語
OS	Raspbian

観測機器は単機で運用するだけでなく、複数機連携での運用が可能である。本システムでは次の 3 通りの運用方法を想定している。

1. メッシュ型配置

   サルの生息域に, 観測機器を検知範囲が互いに重なり合うよう、メッシュ状に設置する運用方法である。これにより、サルの位置を常時監視できるようになる。よって、組織的な追い払いはもちろん、生息調査や捕獲による個体数の管理と調整を行う際にも有効である。しかし、この運用方法を実現するためには観測機器を相当数設置する必要があるため、初期の設置費用と維持費用が高いという問題がある。

2. ライン型配置

   サルの生息域と農地の境界線上に一定間隔で観測機器を設置する運用方法である。これにより、山から人里に降りてきたサルを即時に検知できるため、農地へ近づく前に追い払いを行うことが可能となる。しかし、サルの位置を常時監視できないため、生息調査に用いるためにはデータが不足する恐れがある。

3. 個人型配置

   農家が所有する農地ごとに観測機器を設置する運用方法である。これにより、農家は農地周辺にサルが接近したことを即時的に知ることができる。しかし、観測機器を規則的に設置していないため、利用は即時的な追い払いのみに限定されてしまう。

これらの運用方法は、システムを運用する地域に応じて変更することができるため、一番地域に適した運用方法を実施することができる。

2.5.3 Web アプリケーション

図 2.6 に Web アプリケーションの処理の流れを示す。Web アプリケーションはクライアントとサーバ間の通信、クライアントと Google サーバ間の通信によって構成される。サーバにはデータベース (以下 DB とする) があり、Web ブラウザから送信された HTTP リクエスト (図 2.6(1)) に応じて、ログイン認証、DB 内データの取得と更新などの処理を行い、HTML を返す (図 2.6(2))。これにより、獣害対策に必要な情報を入出力するための UI を実現している。Google サーバには Google Maps API があり、Web ブラウザから送信された HTTP リクエスト (図 2.6(3)) に応じて、サルの位置やサルの動向予測を Google maps 上に描画する処理を行い、JSON 形式のデータを返す (図 2.6(4))。これにより、農地情報とサルの検知履歴を基にサルの位置やサルの動向予測を可視化する機能を実現している。また、サーバが観測機器から送信されたサルの検知情報を受け取った場合 (図 2.6(5)) は、サルが出没したと判断し、農作物被害に遭う可能性が高い農地の所有者へ電子メールを自動送信する。

表 2.4 に WEB アプリケーションの実行環境を示す。

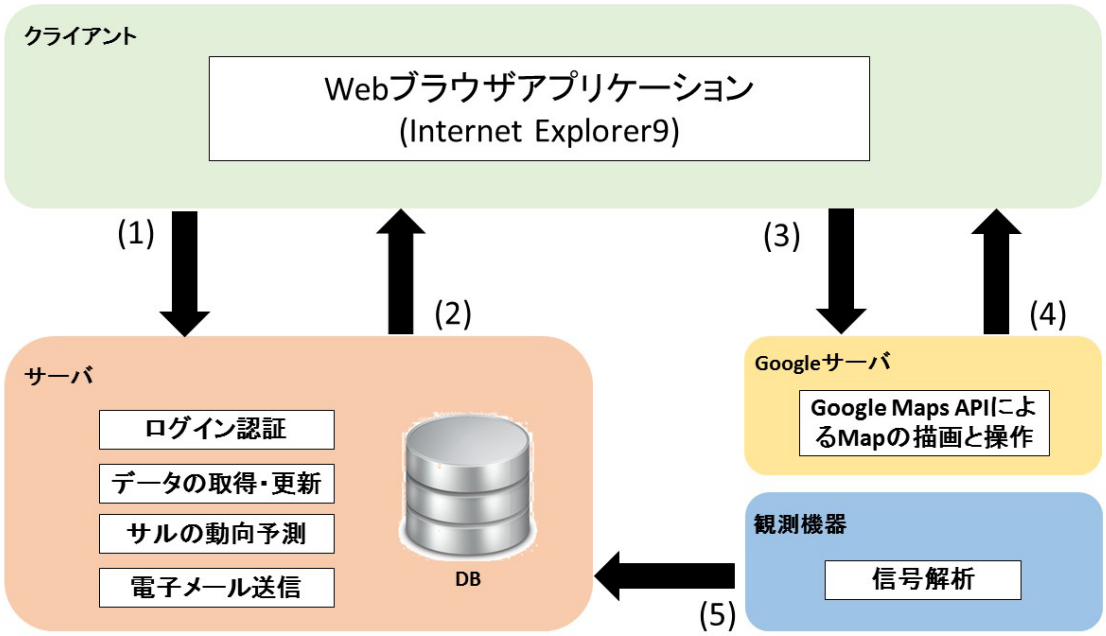


図 2.6: Web アプリケーションの処理の流れ

表 2.4: Web アプリケーションの実行環境

種類	名称	バージョン
OS	Ubuntu	14.04
Web サーバ	Apache	2.4.7
Web アプリケーションフレームワーク	cakePHP	2.4.10
プログラミング言語	PHP	5.3.10
データベース	MySQL	5.5.37
CSS フレームワーク	Twitter Bootstrap	2.2.2
JavaScript ライブラリ	jQuery	1.8.3
単体テストツール	PHPUnit	3.7
結合テストツール	Selenium	2.5.0

2.6　システム導入による効果

本システムを導入することで 2.2 節の「発信機を活用したサルの生息調査と組織的な追い払い」に存在する問題を解決することができる。観測機器は発信機からの電波を受信しているかを自動判別し、その情報をサーバへ送信する機能を備えている。これにより、サルの位置捕捉プロセスの自動化を実現する。さらに、複数の観測機器連携によりサルの位置捕捉の精

度を向上させている。Web アプリケーションでは観測機器からのデータを活用し、サルの動向予測を表示させることでサルの生息調査を効率的に行うことができる。さらに、サルの位置捕捉情報を即時にサーバから電子メールで送付することで従来よりも早い段階での追い払いを可能にする。

2.7　プロジェクト体制

図 2.7 に本プロジェクトのステークホルダを示す。開発メンバ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の学生 5 名である。顧客は JA 鈴鹿営農指導課の川出洋正様である。川出様を通じて、JA 鈴鹿の職員や三重県議会議員、猟師、農家の方々にもご協力していただいた。また課題担当教員として、山際伸一准教授と和田耕一教授に指導をいただいた。

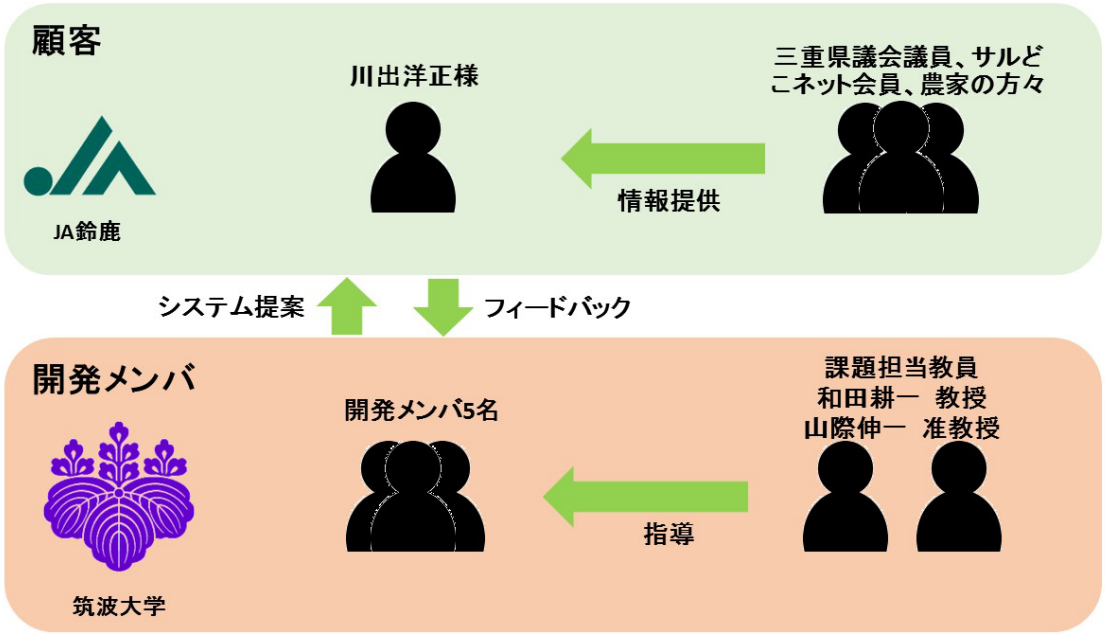


図 2.7: ステークホルダ

次に開発メンバの役割分担と開発を行うために用いたツールについて述べる。表 2.5 に役割分担の一覧を示す。開発メンバはシステム開発、マネジメント、評価を分担し、プロジェクトを進めた。表 2.6 に開発をするために用いたツールの一覧を示す。ソースコードの管理や開発メンバによる開発環境の統一などのためにツールを用いた。

次に開発スコープとスケジュールについて述べる。本プロジェクトではインクリメンタル開発によって開発を進めた。以下に設定した開発したスコープと筆者の担当部分を示し、図

表 2.5: 開発メンバの役割分担

メンバ	役割
天本涼太	Web アプリケーションの開発
	プロジェクトマネジメント
浜野悠介	Web アプリケーションのテスト
	ユーザビリティの向上
森恒成	観測機器回路の設計と実装
	観測機器の評価実験
光岡遼	観測機器の組み込みシステム (音解析と通信) に関する処理
岡部健信	害獣の位置検出アルゴリズムの開発

表 2.6: 開発をするために用いたツール

種類	名称	バージョン
ソースコードバージョン管理システム	Git	1.9.1
開発ドキュメント共有ツール	Dropbox	2.8.2
開発環境作成ツール	Vagrant	1.3.5

2.8 に本プロジェクトのスケジュールを示す。

- S0：顧客から合意を得た要件を満たす機能の開発

  顧客から合意を得た要件を満たす機能の開発を行う。筆者はテスト手法とユーザビリティ評価手法について学習し、本イテレーションで開発した部分のテストを担当する。

- S1：顧客から要望のあった機能や開発メンバ内で新たに挙がった機能の開発

  S0 で開発したシステムを顧客にレビューしていただき、顧客からあった要望や開発メンバ内で新たに挙がった機能をシステムに反映させる。筆者は本イテレーションで開発した部分のテストとユーザビリティの向上としてユーザビリティ・インスペクションを行う。

- S2：システムの品質向上

  システムの品質向上を行うために、さらなるユーザビリティとセキュリティの向上を行う。筆者はユーザビリティの向上としてユーザビリティ・テスティングを行う。

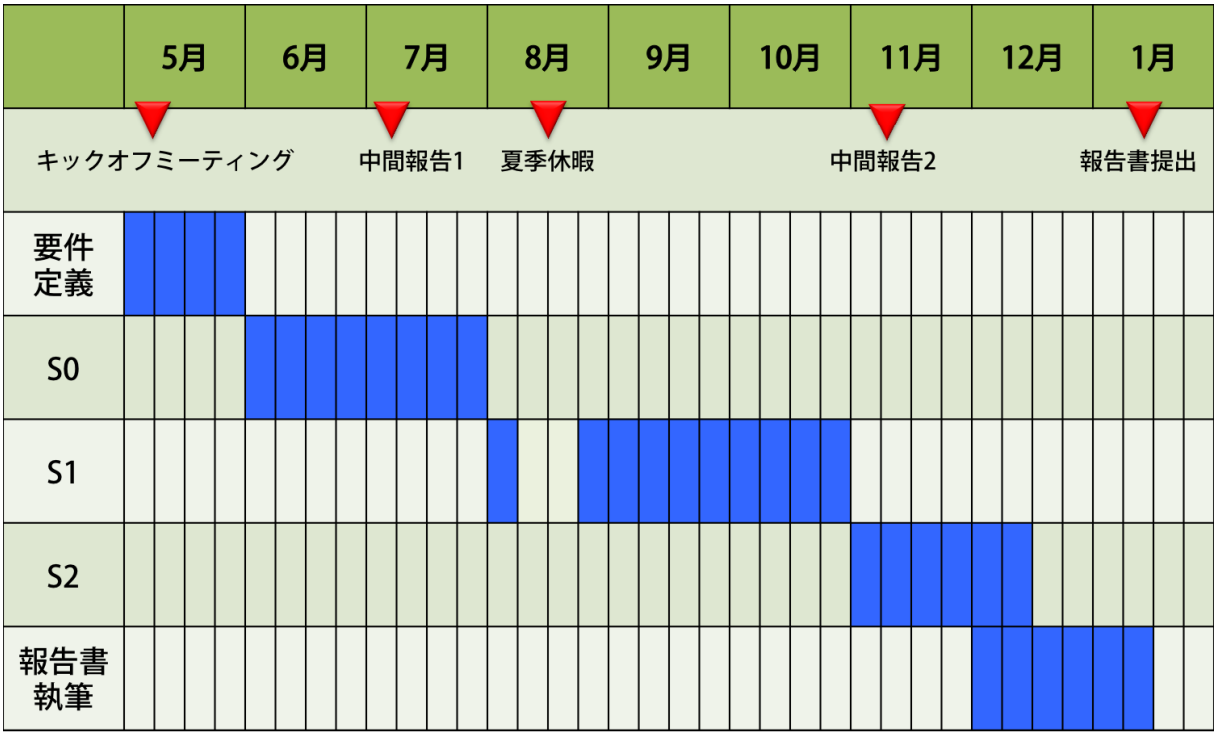


図 2.8: プロジェクトスケジュール

第3章　Webアプリケーションのテスト

本プロジェクトにおいて筆者はWebアプリケーションのテストを担当した。Webアプリケーションのテスト、つまりソフトウェアテストは開発したシステムが仕様通りに動作するかどうかを確認する作業である。近年、ソフトウェアテストの重大性が増しているが、それに比例してソフトウェアテストの困難さも増している [14]。この課題を解決するために、様々なソフトウェアテスト手法が考案されている [15]。本プロジェクトにおいてもWebアプリケーションが仕様通りに動作するかどうかを確認するためにテストを実施した。本章ではテスト計画の立案とテスト十分性を確保するために行うことについて述べる。なお、テスト実施の結果や考察は第5章で述べるため、本章では省略する。

3.1　テスト計画の立案

図3.1にテストの流れを示す。本プロジェクトでは筆者を中心として、テストを「単体テスト」「結合テスト」「受け入れテスト」の3つに分け実施する。単体テストは各スプリントの実装完了後に行い、結合テストは単体テスト完了後に行う。受け入れテストはスプリントを全て終えた後、納品と同時に行う。また、各テストの工程においてテストケースの作成、テスト自動化のためのテストコードの実装を行う。

テスト自動化は多くのシステム開発現場で導入されており、テスト自動化における研究も多い [16][17]。テスト自動化を導入するとテストコードを記述する手間が増える。しかし、同じ仕様のままソースコードが変更されたとしても、記述したテストコードを実行するだけでテストを再実行できる。また、本プロジェクトではインクリメント開発を行うことから、ソースコードの変更が多く発生することが予想され、テスト自動化の導入はテスト実施において大きなメリットになると判断した。テスト実施の際に発見されたバグは開発メンバと共有し、担当を分担してデバッグ作業を行うこととする。次にそれぞれのテスト工程の詳細とバグ管理方法について述べる。

3.1.1　単体テスト

表3.1に単体テストの概要を示す。単体テストはプログラム単体の機能を構成する関数やメソッドを中心に行う [18]。本プロジェクトで開発したWebアプリケーションはフレームワークとしてcakePHPを用いており、MVCモデルを採用している。MVCモデルとはアプリケーションの機能をユーザインタフェース (View) とその他のロジカルな処理 (Model) とViewと

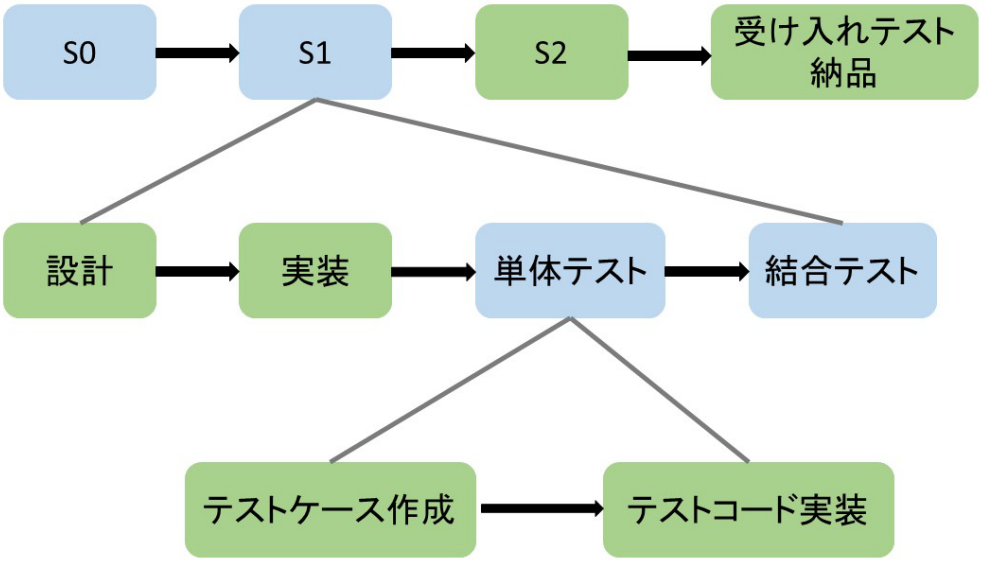


図 3.1: テストの流れ

Model の制御 (Controller) の 3 つに分割することで設計や実装を容易にするプログラミングモデルである [19]。本プロジェクトにおける単体テストの対象は MVC モデルの中においてコントローラクラスとモデルクラスの関数に限定する。ビュークラスのテストは結合テストで行うこととする。なぜならば、ビュークラスは Google Maps API を用いており、実際に Web ブラウザ上に表示されているものを目視で確認する必要があるからである。本プロジェクトの単体テストでは仕様通りに関数が動作するかをテストする。具体的には入力値と期待される出力値を用意し、実際の出力値と期待される出力値を比較する。この入力値と期待される出力値の準備と、実際の出力値と期待される出力値の比較をテストコードとして記述し、自動化する。自動化のために単体テストでは PHPUnit というテスティングフレームワークを用いる。以下に PHPUnit を用いた理由を 2 点示す。

1. 他の言語のテスティングフレームワークと共通部分が多く、学習しやすい

   テスティングフレームワークは多くの言語で提供されており、Java の JUnit や Ruby の RubyUnit などがあある。PHPUnit を含めたこれらのテスティングフレームワークは総称して xUnit と呼ばれ、SmallTalk 用の SUnit が基になっている [20]。つまり、水準が同じテスティングフレームワークが多く存在し、学習教材が豊富であると言える。

2. コードカバレッジ測定機能が備わっている

   本プロジェクトではコードカバレッジ測定を行うことで、単体テストの品質を測る。

PHPUnit にはコードカバレッジ測定機能が組み込まれており、PHPUnit を導入すれば、コードカバレッジ測定を行うことができる。コードカバレッジ測定については 3.3 節で説明する。

表 3.1: 単体テストの概要

対象	コントローラクラス、モデルクラス
確認項目	仕様通りに関数が動作するか
使用ツール	PHPUnit

3.1.2 結合テスト

表 3.2 に結合テストの概要を示す。結合テストは複数のモジュールを組み合わせて行う。本プロジェクトの結合テストは MVC モデルのモジュールをそれぞれ組み合わせて、仕様通りに動作するかをテストする。具体的には Web ブラウザ上での画面遷移やクリックといった GUI 操作が正常に動作するか、Google Map に表示されている内容が正しいか、などを確認する。これにより、ユーザ視点での機能の振る舞いや、UI を検証することができる。本プロジェクトの結合テストでは Selenium というテストツールを用いる。Selenium の特徴として、Web ブラウザ上での GUI 操作を自動で行える点がある。また、どの画面に遷移したか、特定の文字列が含まれているか、などを自動で検証することも可能である。例えば、以下のような動作を想定する。

1. 管理者一覧画面の表示
2. 編集する管理者情報を選択
3. 編集画面へ遷移
4. 選択した管理者情報を編集
5. 確認画面へ遷移
6. 編集完了画面へ遷移

こういった動作確認をテスト自動化を行わずにテストする場合、ソースコードが変更する度にテスト担当者が実際に Web ブラウザ上で操作を行う必要があり、大変手間である。テストコードを記述すれば、こういった動作確認を短時間で行うことができる。さらに、こういった自動化を行う場合、通常テストコードを記述する必要があるが、Selenium IDE というプラグインを用いることにより、ユーザが操作した GUI 操作を記録し、テストコードを自動生成する。これにより、テストコードを記述する技術が無かったとしても実装部分を知らない開発メンバでもテストを行うことができる。これらの特徴から Selenium の導入は結合テストに

とってメリットがあると判断できる。しかし、本プロジェクトでは Selenium では自動化できない動作確認がいくつか存在する。それは、主に Google Map 表示部分であり、Google Map 上に配置しているアイコンの確認、編集はテスト担当者の手動での確認となる。

表 3.2: 結合テストの概要

対象	MVC モデルの組み合わせ
確認項目	仕様通りに機能が動作するか
使用ツール	Selenium

3.1.3 受け入れテスト

表 3.3 に受け入れテストの概要を示す。受け入れテストは納品時に顧客がシステムの機能や性能が要件と合致しているかの確認を行う。本プロジェクトにおいても顧客である JA 鈴鹿に受け入れテストを担当していただく。受け入れテストの場合、行う内容は結合テストと同じであるが、全ての結合テストのテストケースを顧客に実行していただくのは大きな負担となってしまう。よって、本プロジェクトでは受け入れテストは機能毎の正常系の確認とシステムの評価アンケートの実施とする。また、それと同時に納品物である設計書やマニュアルの評価もしていただく。

表 3.3: 受け入れテストの概要

対象	システム全体の機能
確認項目	顧客が期待していた仕様通りに機能が動作するか
使用ツール	なし

3.1.4 バグ管理方法

本プロジェクトではバグを発見した場合、バグ管理表に内容を記述し、担当者にバグ修正を依頼することとした。表 3.4 に本プロジェクトで定めたバグの定義を示す。

表 3.4: バグの定義

設計のバグ	仕様として決定していない項目がある テストケースが仕様に反している
コーディングのバグ	仕様と異なる機能を実装する イテレーション終了時に実装すべき機能が未実装である

3.2　テスト自動化ツールの導入方法

本プロジェクトではテスト自動化のために PHPUnit と Selenium を導入する。以下に各ツールの導入方法について述べる。

3.2.1　PHPUnit

PHPUnit は PHP の PEAR パッケージの一部であり、PEAR コマンドを用いることで簡単にインストールすることができる。また、PHPUnit は以下の規約に従う必要がある。

- テストファイルは app/Test/Case/というディレクトリに配置する
- テストファイル名は***Test.PHP とする
- テストのクラスは必ず ControllerTestCase を継承する
- テストファイル名とテストクラス名は同じにする
- テストのメソッド名は test***とする

これらの規則に従うことで実装したコードが期待した通りに動作するかの確認を行うことができる。これに伴い、期待した値と実際の出力値を比較するためにアサーションを用いる。アサーションには複数のメソッドが用意されており、数値や真偽値、文字列、配列などの比較を行うことができる。テストコードの実装が完了したら、テストの実行をする必要がある。cakePHP はテストを実行するために Web ベースのインタフェースを提供しており、Web ブラウザを通してテストを実行することができる。テストの実行と並行してコードカバレッジの測定も可能である。

テストコードの挙動がデータベースに依存する場合、フィクスチャを用いることでテスト用のデータをデータベースに投入することができる。そのため実際のアプリケーションに使われているデータに惑わされることなくテストを行うことができる。フィクスチャを作成する場合は app/Test/Fixture というディレクトリにファイルを設置する。また、どのようなフィールドを持ったテーブルを作成するか、初期状態でどのようなレコードをテーブルに配置するかを定義する。フィクスチャを用いてテストする場合、以下の動作をする。

1. フィクスチャで定義されたテーブルを作成する
2. フィクスチャで定義されたデータをテーブルに投入する
3. テストを実行する
4. フィクスチャで作成したテーブルを空にする
5. フィクスチャで作成したテーブルを削除する

3.2.2 Selenium

本プロジェクトでは Selenium を用いるために Mozila Firefox のプラグインである Selenium IDE を導入した。3.1.2 節で述べたように、これを導入することでブラウザ操作を記録して、そこから Selenium のテストコードを自動生成するためテストコードの作成がより簡単になる。図 3.2 に Selenium IDE の動作例を示す。ブラウザの操作を記録状態にすることでテストコードを自動生成する (3.2(1))。記録することができる操作は指定 URL の表示 (3.2(2)) やテキストの入力、ボタンのクリック (3.2(3)) などがある。さらに、アサーションを用いて値の比較や表示している画面の URL の比較 (3.2(4)) を行うことができる。

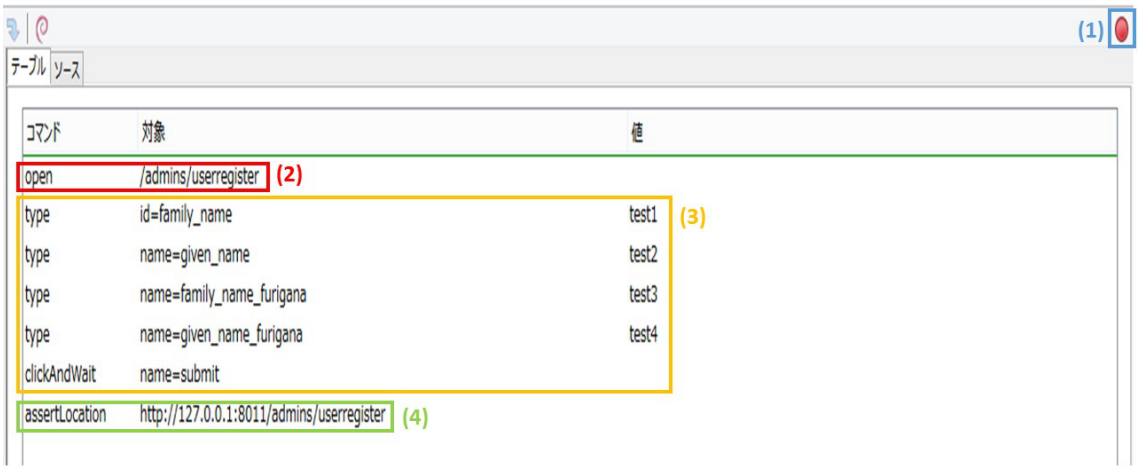


図 3.2: Selenium IDE の動作例

3.3 テスト十分性

テストをどこまでやるかを決めることは非常に難しい。なぜなら、テストはバグがあることを示すことはできても、バグがないことを示すことはできないからである。だからといって、できるだけバグがないことを示すために、ソフトウェアが取り得る全ての状態、入力条件の組み合わせをテストすることは、現在のコンピュータの計算性能では時間的に不可能である。そのため、多くのシステム開発の現場では以下のような基準を設けて、テスト終了を判断している [21]。

- カバレッジなどの技術面からみたプロダクトリスクの度合い

- 開発期間や予算などプロジェクトの制限

本プロジェクトにおいてもテスト十分性の確保を行う。以下に単体テストと結合テストにおいて筆者が行ったテスト十分性の確保の方法について述べる。

3.3.1 単体テストの十分性確保

単体テストにおいてはホワイトボックステストというテスト手法を用いて、テストケースを作成することで十分性を確保する。ホワイトボックステストとはソフトウェア内部の構造を分析し、実装した機能が意図した通りに動作しているかを検証する方法である。具体的には、プログラム中の命令や条件、分岐毎に網羅するようにテストする。このプログラムのソースコードがテストされた割合をカバレッジと呼ぶ。ホワイトボックステストではカバレッジを測定することで、テストが実施されていないソースコードを確認し、テストの十分性を確保できる。カバレッジには複数のカバレッジ基準が設けられている。以下に代表的なカバレッジ基準を示す [22]。

- 命令網羅 (C0)

  命令を全て実行するようにテストする

- 分岐網羅 (C1)

  分岐命令の真偽値の両方を全て実行するようにテストする

- 複合条件網羅 (C2)

  条件文の真偽値の組み合わせを全て実行するようにテストする

これらのカバレッジ基準には包含関係が成り立つことが知られている [23]。つまり複合条件網羅は分岐網羅を包含し、分岐網羅は命令網羅を包含する。つまり、この 3 つのカバレッジ基準であれば、複合条件網羅が最もテストケースが多く、信頼性が高い基準である。しかし、複合条件網羅は条件数が増えるにしたがって指数関数的に組み合わせ数が増加するため、現実的には実施が困難な基準である。

本プロジェクトでは命令網羅によるカバレッジ測定を行う。本プロジェクトのテスト担当者が筆者のみであることから、テストの工数が足りなくなった場合に補うことが難しいと考え、最も作成するテストケース数が少なく、工数を抑えることができる命令網羅を採用する。カバレッジ測定は PHPUnit の機能を用いて行う。図 3.3 にカバレッジ測定結果の一例を示す。コードが青く表示されている部分は少なくとも一回はテストによって実行されている命令であり、赤く表示されている部分は一回もテストによって実行されていない命令である。実行されていない命令を無くすようにテストを実行し、カバレッジをできるだけ 100%になるようにすることで単体テストの十分性を確保する。

しかし、ホワイトボックステストでのカバレッジ測定には欠点が指摘されている [24]。カバレッジ測定の場合、要件と違った機能がプログラムに組み込まれていたとしても判明しな

```
998         public function boxconfirm(){
999                 $this->set('userinfo', $this->Auth->user());
100                 $this->set('request',$this->request->data);
100                 $geos = $this->Geo->find('all');
100                 $this->set('geos',$geos);
100                 $boxes = $this->Box->find('all');
100         $this->set('boxes',$boxes);
1005
1006        // 編集から遷移してきたとき、かつ、観測箱の削除を行っているとき
100         if(!empty($this->request->data['id']) && ($this->request->data['is_torikeshi'] != 0) ){
100                 $pre_box_data = $this->Box->find(
100                                 'first',array(
1010                                                 'conditions' => array(
101                                                                 'Box.id' => $this->request->data['id']
101                                                 )
101                                 )
101                 );
101                 $this->set('pre_box_data',$pre_box_data);
101         }
1017
101                 if(!empty($this->request->data)){
101                         if(empty($this->request->data['name'])){
1020                                    //$this->Session->setFlash(__('名前を入力してください。'));
1021                                    //if($this->Session->check('AlertBoxName.posted')){
1022                                    //      $this->Session->delete('AlertBoxName.posted');
1023                                    //}
102                                     $this->Session->write('AlertBoxName.posted',1);
102                                     $this->Session->write('Register.posted', $this->request->data);
102                                     $this->redirect($this->referer());
102                                 } else if(empty($this->request->data['location_name'])){
1028                 //$this->Session->setFlash(__('観測箱の位置を入力してください。'));
1029                                    //if($this->Session->check('AlertBoxLocation.posted')){
```

図 3.3: カバレッジ測定結果の一例

いという欠点がある。そのため本プロジェクトではこの欠点を補うために結合テストにおいて外部構造からテストケースを作成するブラックボックステストを行う。

3.3.2 結合テストの十分性確保

結合テストにおいてはブラックボックステストというテスト手法を用いて、テストケースを作成することで十分性を確保する。ブラックボックステストとはシステムの内部構造とは無関係に、外部から見た機能を検証する方法である。本プロジェクトでは以下に示すドキュメントを作成し、十分性を確保した上でブラックボックステストを行う。

- FV 表

  FV 表 (Function Verification Table)[25] は、各画面について機能と検証内容を記述する表である。FV 表を用いることで、テストケースを作成する観点を統一させることができる。図 3.4 に作成した FV 表の一部を示す。記述する内容は、テスト対象とする画面の名称、各画面にどのような機能があるか、機能に対して検証すべき項目、検証項目に合わせて作成するテスト仕様書である。

- テスト仕様書

  本プロジェクトでは作成するテスト仕様書をデシジョンテーブルと要素別動作確認表とする。

画面名	目的機能	検証内容	作成テスト仕様書
トップ	・現在時刻の表示 ・サルの出現率(ヒートマップ)表示 ・観測箱、農地の表示 ・農地アイコン表示の切り替え ・マニュアルへのリンク ・1分毎のページ更新	・観測箱と農地の表示が正しいか ・サルの出現率表示(ヒートマップ)が正しいか ・農地アイコン表示の切り替えが正しいか ・マニュアルへのリンクが正しいか ・1分毎にページが更新されているか	要素別動作確認書
管理者ログイン	・管理者ページへのログイン機能	・登録されているログインIDとパスワードでログインできるか	デシジョンテーブル
メニューバー	・各一覧ページへのリンク ・ログインユーザ名がログインしている管理者のログインIDを表示する	・各一覧ページへのリンクが正しいか ・ログインユーザ名がログインしている管理者のログインIDが表示されているか	要素別動作確認書
管理者トップ	・現在時刻の表示 ・サルの出現率(ヒートマップ)表示 ・観測箱、農地の表示 ・農地アイコン表示の切り替え ・マニュアルへのリンク ・1分毎のページ更新	・観測箱と農地の表示が正しいか ・サルの出現率表示(ヒートマップ)が正しいか ・農地アイコン表示の切り替えが正しいか ・マニュアルへのリンクが正しいか ・1分毎にページが更新されているか	要素別動作確認書
利用者一覧	・利用者情報の表示 ・各利用者情報の操作リンク ・利用者の名前検索 ・特定のカラムのソート機能	・利用者情報の表示内容は正しいか ・各利用者情報の操作リンクは正しいか ・利用者の名前検索機能は正しいか ・カラムのソート機能が正しいか	要素別動作確認書
利用者登録・編集	・利用者情報を新規登録 ・利用者情報を編集 ・利用者情報登録・編集確認ページへのリンク	・各入力欄に記述されている内容は正しいか ・セッションが存在した場合、正しく表示されているか ・JavaScriptの動作は正しいか ・利用者登録・編集確認ページへのリンクは正しいか	デシジョンテーブル

図 3.4: FV 表の一部

デシジョンテーブルは複数の入力や条件に対する出力や動作を表にしたものである [26]。図 3.5 に利用者登録・編集画面のデシジョンテーブルを示す。これを用いることで複数の入力項目の組み合わせが存在する画面に対して、判定処理を網羅するようにテストケースを作成することができる。

条件	検知メールの受け取り	受け取らない	－	－	－	－	－	○			○	○	○	○	○	○	○	○	○
		自分の農地のみ	－	－	－	－	－		○										
		全ての農地	－	－	－	－	－			○									
	農地個数	0	－	－	－	－	－	－	－	－	○	○	○						
		1	－	－	－	－	－	－	－	－				○	○	○			
		2以上	－	－	－	－	－	－	－	－							○	○	○
	農地場所	観測箱の検知範囲円内	－	－	－	－	－	－	－	－				－	○		○		○
		観測箱の検知範囲円外	－	－	－	－	－	－	－	－				－		○		○	○
	農作物	代表的な農作物を選択	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－
		未設定	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－
		それら以外	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－
	「ここに農地を追加」	なし	－	－	－	－	－	－	－	－	○			○					
		あり	－	－	－	－	－	－	－	－		○	○		○	○	○	○	○
	「操作を取り消し」	なし	－	－	－	－	－	－	－	－				－	－	－	－	－	－
		あり	－	－	－	－	－	－	－	－	○		○	－	－	－	－	－	－
	農地クリックによる削除	なし	－	－	－	－	－	－	－	－				－	－	－	－	－	－
		あり	－	－	－	－	－	－	－	－		○	○	－	－	－	－	－	－
トリガ	農地確認	農地確認ボタン押下	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
動作		利用者登録・編集確認ページへ遷移						○	○	○	○	○	○		○	○	○	○	○
	必須	必須チェックエラー	○	○	○	○	○												
		農地追加途中エラー												○					
確認方法			自動	自動	自動	自動	自動	自動	自動	自動	手動	手動	手動	手動	手動	手動	手動	手動	手動
結果		11月6日									○	○	○	×	○	○	×	○	×
		11月20日	○	○	○	○	○	○	○	○				○			○		○

図 3.5: 利用者登録・編集画面のデシジョンテーブル

要素別動作確認表は各要素を個別に動作確認すれば十分な画面に対して、全要素を確認する表である。表 3.5 に利用者一覧画面の要素別動作確認表を示す。

表 3.5: 利用者一覧画面の要素別動作確認表

要素名	論理項目種別	確認内容
氏名	ラベル	氏名が正しく表示されている
氏名（よみ）	ラベル	氏名 (よみ) が正しく表示されている
氏名 (よみ)	ラベル	ソートする
E-mail	ラベル	E-mail が正しく表示されている
検知メールの受け取り	ラベル	検知メールの受け取りが正しく表示されている
農地位置	ラベル	農地位置が正しく表示されている
編集操作	ボタン	ボタンに対応した利用者の利用者編集ページへ遷移
削除操作	ボタン	ボタンに対応した利用者の削除モーダルを展開
削除モーダル	ボタン	削除ボタンを押下すると利用者情報の削除、キャンセルボタンを押下すると処理を何もせずにモーダルを閉じる
新規登録	ボタン	利用者登録ページへ遷移
新規登録	ボタン	観測箱が一つも動作していない場合ボタンを表示させない
検索	テキストボックス	利用者の氏名の全部もしくは一部が一致する利用者を検索する
氏名 (よみ)	ラベル	検索した後にソートを実行すると検索した中でソートする
削除操作	その他	農地を所有している利用者を削除した時、所有している農地も削除する

本プロジェクトでは以下に示す手順で結合テストを行う。

1. FV 表を作成し、各画面の検証内容と作成するテスト仕様書を列挙する。
2. 各画面に対して、テスト仕様書を作成する。
3. テスト仕様書を基にテストを実施する。
4. バグを発見した場合はバグ管理表に内容を記述し、担当者にバグ修正を依頼する。

第4章　ユーザビリティの向上

本プロジェクトにおいて筆者はWebアプリケーションのユーザビリティの向上を担当する。ユーザビリティとは一般的に使いやすさや操作性を意味する。近年のPCやスマートフォンなどの情報処理機器の急速な普及とユーザ層の拡大に伴い、ユーザビリティの重要性が高まっている [27][28][29]。本章ではソフトウェア品質とユーザビリティの関係、及びユーザビリティを向上させるために筆者が行ったユーザビリティ評価について述べる。なお、ユーザビリティ評価の結果や考察は第5章で述べるため、本章では省略する。

4.1　品質とユーザビリティの関係

ソフトウェア品質とはソフトウェアに本来備わっている特性、もしくは属性をいう。ISO/IEC9126はこのソフトウェア品質の評価に関する国際規格である [30]。表 4.1 に ISO/IEC9126 で示されているソフトウェア品質特性を示す。品質特性は6種類存在する。

表 4.1: ソフトウェア品質特性

品質特性	説明
機能性	ソフトウェアに明示的、及び暗示的に要求されている機能の実装度合い
信頼性	指定された条件の下で機能要件を維持し続けることができる度合い
使用性	ソフトウェア使用の際の理解しやすさ、習得しやすさ、使いやすさの度合い
効率性	明示された条件の下で目的達成のために使用する資源の度合い
保守性	ソフトウェア修正の際の労力の度合い
移植性	ある環境から他の環境への移行の際の労力の度合い

ユーザビリティはこの品質特性の内の一つである使用性のことを示す。さらに使用性にはブレークダウンした品質副特性が5種類存在する。表 4.2 に使用性の品質副特性を示す。

表 4.2: 使用性の品質副特性

使用性の品質副特性	説明
理解性	ソフトウェアの機能や使い方をどれ程理解できるかの度合い
習得性	ソフトウェアを使用する上で参照するマニュアルの読みやすさや、運用方法を覚えるまでの度合い
運用性	ソフトウェアの使用時に感じるストレスの度合い
魅力性	画面のデザインや UI が感性に訴求するかの度合い
使用性標準適合性	国際規格 ISO9241 シリーズなどの使用性に関連する規格や規約に準拠している度合い

本プロジェクトではこれら使用性の品質副特性を考慮しながら、全体のユーザビリティ向上を目指す。

4.2 ユーザビリティ評価手法

本節では、本プロジェクトで行うユーザビリティ評価手法であるユーザビリティ・インスペクションとユーザビリティ・テスティングについて述べる。

4.2.1 ユーザビリティ・インスペクション

ユーザビリティ・インスペクションは評価者自らの知識や経験に基づいて評価を行い問題点を発見する手法である [31]。しかし、評価者毎に評価の基準が定まっていないと意味がない。そこで、評価に客観性を持たせるために様々なガイドラインが考案されている。その中でも Jakob Nielsen が考案した 10 ヒューリスティックスは代表的である [32]。表 4.3 に 10 ヒューリスティックスを示す。

本プロジェクトではこの 10 ヒューリスティックスをガイドラインとして用いて、評価対象が犯しているルール違反を探索する手法であるヒューリスティックス評価を行う。ヒューリスティックス評価は精度と客観性に優れており、代表的な手法として世界中で用いられるている。しかし、ヒューリスティックス評価にもいくつか問題点が存在する。以下にその問題点と本プロジェクトで行う対策について述べる。

- 問題点の検出過多

  問題点の一つは評価対象の問題点を見つけすぎることである。問題点を見つけすぎてしまうと、全ての問題点の対策を実施するとなると、納期などの関係上難しいことが多い。本プロジェクトにおいては、発見した全ての問題点を修正せずに、各問題点に重要

表 4.3: 10 ヒューリスティックス

10 ヒューリスティックス	説明
システム状態の視認性	システムは今、何を実行しているのかを常にユーザに知らせなくてはならない
システムと実世界の調和	ユーザに馴染みのある用語、フレーズ、コンセプトを用いる
ユーザコントロールと自由度	不測の状態から抜け出すための手段を用意する
一貫性と標準化	異なる用語、状況、行動が同じことを意味してはいけない
エラーの防止	問題の発生を防止するようなデザイン
記憶しなくても、見ればわかるように	システム利用のための説明は可視化するか、いつでも引き出せるようにする
柔軟性と効率性	初心者と経験者の両方の要求を満たす
美的で最小限のデザイン	余分な情報は相対的に視認性を減少させる
ユーザによるエラー認識、診断、回復をサポートする	エラーメッセージは平易な言葉で表現し、問題を的確に指示する
ヘルプとマニュアル	ヘルプやマニュアルを提供する

度を評価者ミーティング時に設定し、重要度が高い問題点から修正していくという対策を講じる。

- 実施コスト

  もう一つの問題点は実施コストがかさむことである。評価対象にもよるが、通常、ヒューリスティックス評価を実施する場合、評価者として複数の専門家を雇う必要があり、人件費がかかる。本プロジェクトでは開発メンバを評価者とすることで実施コストを抑える。

- 認知度

  そもそも評価のガイドラインとなる 10 ヒューリスティックスがあまり知られていないという問題点がある。一般の人はもちろん、開発者やデザイナなどでも十分に理解している人は少ない。実際に本プロジェクトの開発メンバにおいても認知しているメンバは存在しなかった。そのため、筆者が始めにヒューリスティックス評価を学習し、ヒューリスティックス評価マニュアルを作成することで、開発メンバにヒューリスティックス評価を周知させることとする。

本プロジェクトではヒューリスティックス評価を用いたユーザビリティ・インスペクションを以下に示す手順で行う。

1. 評価者に 10 ヒューリスティックスを周知させる。

2. 評価者個人が 10 ヒューリスティックスを基にシステムを評価する。

3. 評価者全員の評価が終わり次第、開発メンバを集め、ミーティングを実施する。

4. ミーティング内で発見した問題点を開発メンバ内で共有し、設定した重要度から修正する問題点を決定する。

5. 修正する問題点に対して、対策方法、修正担当者、期限を決定する。

4.2.2 ユーザビリティ・テスティング

ユーザビリティ・テスティングは被験者にシステムを操作してもらい、問題点を発見する方法である [33]。これはユーザが実際に操作し、ユーザ視点からしか得られない重要な問題点を発見することができる。本プロジェクトにおいても被験者に協力をお願いし、ユーザビリティ・テスティングを実施する。なお、ユーザビリティ・テスティングは 4.2.1 節で述べたユーザビリティ・インスペクションを実施した後に実施する。理由は、ユーザビリティ・インスペクションであるヒューリスティックス評価を行い、修正を行うことで最低限のユーザビリティは確保していることから、ユーザビリティ・テスティング時の被験者への負担を軽減させることができるからである。なお Jakob Nielsen が提唱した被験者数と発見できるユーザビリティ問題の数に関する公式 4.1 から被験者の数は 5 人程度とする [34]。

$$N(1-(1-L)^n) \tag{4.1}$$

- N：潜在的なものを含むユーザビリティ問題の数
- L : 1 人の被験者がテストして発見できるユーザビリティ問題が全体に占める割合（Jakob Nielsen は 0.31 を提示している)
- n：被験者数

この公式に例えば L=0.31、n=5 を代入すると A=0.8436N になる。仮に 100 個のユーザビリティ問題が潜在的に存在していたとすると 5 人の被験者によってユーザビリティ・テスティングを行った場合、約 84 個、つまり全体の約 84%のユーザビリティ問題を発見することができる。本プロジェクトでは全体の 80%程度のユーザビリティ問題を発見することができれば、十分であると判断する。

次に本プロジェクトで用いるユーザビリティ・テスティング手法について述べる。

思考発話法

本プロジェクトのユーザビリティ・テスティングは被験者にタスク (作業) を提示し、それを実行するように依頼する。その際に、タスクの達成・未達成の記録だけでなく、被験者に

考えていることを常に話しながらタスクを実行していただき、その内容を記録する。この手法を思考発話法という [35]。思考発話法を用いることにより、操作の「失敗」「戸惑い」「不満」の事実だけでなく、原因や理由を明らかにすることができる。

System Usability Scale(SUS)

SUS はユーザビリティに関する 10 個の質問があるアンケートである [36][37]。SUS を用いることによりユーザビリティを定量的に計測することができる。表 4.4 に SUS のアンケート項目を示す。

表 4.4: SUS のアンケート項目

項目番号	質問内容
1	このシステムを今後も繰り返し使いたい
2	このシステムは必要以上に複雑だった
3	このシステムは簡単に使えた
4	このシステムを使えるようになるには、技術に詳しい人のサポートが必要だ
5	このシステムは機能がよくまとまっていた
6	このシステムは一貫性に欠けるところが多かった
7	たいていの人は、このシステムをすぐ使えるようになるだろう
8	このシステムはとても使いづらかった
9	このシステムを使っていて不安を感じることはなかった
10	このシステムを使い始める前に、多くのことを学ぶ必要があった

これらの項目に対して 5 段階のリッカート尺度を設け、回答を得る。表 4.5 に設けたリッカート尺度について示す。

表 4.5: リッカート尺度

回答番号	評価内容
1	全くそう思わない
2	そう思わない
3	どちらともいえない
4	そう思う
5	非常にそう思う

SUS は 2 点の特徴がある。 l つは 10 個のアンケート項目のみであり、被験者へのアンケート回答への負担が小さいという点である。そしてもう 1 つの大きな特徴は式 4.2 を用いて、0 から 100 までのスケールへ変換し、ユーザビリティを評価できる点である。

$$2.5\{\sum_{i=1}^{5}(S[2i-1]-1)+\sum_{i=1}^{5}(5-S[2i])\} \tag{4.2}$$

- S[i]：アンケート項目 i 番目の回答番号

このスケールが 100 に近い程、ユーザビリティが高いということが言える。Jeff Sauro は調査からこのスケールの平均値を 68 としている [38]。本プロジェクトではこのスケールの目標値を 75 と設定し、ユーザビリティ向上を目指す。

本プロジェクトでは思考発話法と SUS を用いた以下に示す手順でユーザビリティ・テスティングを行う。

1. 被験者にタスクを提示し、そのタスクを達成するようにシステムを操作してもらう。
2. 被験者がシステムを操作している時は思考発話法を用いて、被験者の意見を聞き出す。
3. タスクを全て終えたら、SUS のアンケートに回答していただく。
4. 被験者全員の評価が終わり次第、開発メンバを集め、ミーティングを実施する。
5. ミーティング内で発見した問題点を開発メンバ内で共有し、設定した重要度から修正する問題点を決定する。
6. 修正する問題点に対して、対策方法、修正担当者、期限を決定する。

第5章　Webアプリケーションのテストとユーザビリティの向上における評価と考察

5.1　Webアプリケーションのテストにおける評価

本節では第3章で述べた手法を本プロジェクトに適用した場合の結果と考察を述べる。本プロジェクトでは、単体テストと結合テストを行い、同時にバグ管理を行った。

5.1.1　結果

表5.1に単体テスト結果を示し、表5.2に結合テスト結果を示す。また、表5.3に発見したバグの分析結果を示す。

表5.1: 単体テスト結果

テスト対象コードのステップ数	テストケース数	テストコードステップ数
1642	110	4943

表5.2: 結合テスト結果

テスト対象のViewファイル数	テストケース数
23	210

表5.3: バグの分析結果

バグの種類	バグ数
設計のバグ	36
コーディングのバグ	29

受け入れテストとしてプロジェクト評価アンケートを行った。当初の計画ではこれに加えて、手動による結合テストで行ったテストケースの正常系の確認と運用マニュアル評価アンケートを実施する予定だったが、システム評価を行う予定の担当者が不在になり、実施することができなかった。表5.4にプロジェクト評価アンケート結果を示す。回答に示している数値は表4.5で示したリッカート尺度と同じものである。

表 5.4: プロジェクト評価アンケート結果

質問内容	回答
納品したシステムは抱えている問題を解決することができる	4
納品したシステムはプロジェクト発足時に定義した要件を満たしている	5
納品したドキュメント (書類) に不備はなかった	3
プロジェクトの進捗具合を定期的に報告していてた	5
ミーティングの開催頻度は適切だった	4
ご意見、ご要望、ご感想がありましたら、ご記載ください (自由記述)	特になし

5.1.2 考察

本プロジェクトでは単体テストにおいてはカバレッジ率 100%、結合テストにおいては FV 表とテスト仕様書の作成を実施することでテストの十分性を確保することとした。

単体テストはテスト対象コードのステップ数 1642 に対してテストケース数は 110(表 5.1) となり、カバレッジ率を 100%でテストを終了させることができた。本プロジェクトのソフトウェアの規模から 110 というテストケース数は少ないが、単体テストにおいては最もテストの工数が少ない命令網羅を採用したため、このような結果となった。

結合テストはテスト仕様書として作成したデシジョンテーブルと要素別動作確認表から合計で 210(表 5.2) のテストケースを導出した。210 のテストケースの内、127 が Selenium を用いて自動化したテスト、83 が手動で確認したテストとなる。手動によるテストが多くなってしまった理由は Google Maps API を用いた機能をテストする部分が多く、Selenium ではその部分を自動化することができなかったためである。

受け入れテストはシステム評価担当者不在のためプロジェクト評価アンケートの実施のみになってしまった。プロジェクト評価アンケートの結果は概ね良い回答を得られたが、「納品したドキュメント (書類) に不備はなかった」という質問に対する回答が他と比べて低かった (表 5.4)。原因はシステム評価担当者が不在のため、納品したドキュメントの確認が困難であったことである。

本プロジェクトのテスト計画の立案について適切だった点はテストの十分性の確保について最初に計画できたことである。単体テストにおいてはカバレッジ率 100%を終了条件として設定していたため、これを満たすようにテストケースを作成することにより、十分性を確保できた。また、結合テストにおいてはテスト仕様書を作成することにより、テストケースの漏れを無くし、十分性を確保できた。不適切だった点は主に 2 つある。一つはテスト担当者を筆者のみとし、テストの作業も全て筆者一人で行った点である。筆者が実装を担当しなかった部分においても筆者がテストを担当したため、機能の詳細だけでなくコードレベルで理解する必要があった。そのため、コードの読み込みや実装者への問い合わせに時間を費やすことになり、多くの工数を割り当てることとなってしまった。テスト担当者はテストの取りまとめを行い、テスト実施は開発メンバ内で行うべきだった。もう一つは多くの仕様変更に柔軟に対応してテストを実施できなかった点である。表 5.3 に示した通り、設計のバグが 36 個見

つかった。その多くは仕様として決定していない項目だった。さらに、開発メンバにバグの計上をするように指示していたが、多くのメンバがバグを計上せずに変更作業を行っていたため、実際は記録していたバグよりも多くのバグが存在していたことが言える。これらがテスト実施中や終了後に発見され、仕様が変更される度にテストケースも変更する必要があった。そのため、その都度大きな手戻りが発生してしまい、多くの時間を費やしてしまった。原因は設計をあいまいにしてしまったことにある。設計時に開発メンバ内でレビューを徹底して行う必要があった。

以上のことからソフトウェアテスト手法とテスト自動化ツールを導入することで本プロジェクトのテストの十分性の確保とテスト効率の向上を実現した。先に述べたようにソフトウェアテスト手法を導入したことにより、テストの十分性の確保を行うことはできたが、当初計画していた工数よりも多くの工数を費やしてしまった。この原因は開発メンバと密にコミュニケーションや連携をとってテストを実施しなかったことが考えられる。テストもシステム開発において重要な工程であるということを開発メンバに認識させ、協力して行うべきであると言える。

5.2 ユーザビリティ向上における評価

本節では 4.2 節で述べた手法を本プロジェクトに適用した場合の結果と考察を述べる。本プロジェクトでは、ユーザビリティ・インスペクションとユーザビリティ・テスティングを行うことにより、ユーザビリティの評価と向上を行った。

5.2.1 結果

ユーザビリティ・インスペクション

ヒューリスティックス評価を行う評価者は開発メンバ 5 人とし、S0 終了後に実施した。表 5.5 に利用者登録・編集画面のヒューリスティックス評価結果を示す。また表 5.6 に全画面に対するヒューリスティックス評価の分析結果示す。合計 80 項目の問題点を発見した。発見した問題点に対して重要度を設定し、重要度が高い問題点を優先して修正することとしたため、結果として 26 項目の問題点に対して修正を行った。

表 5.5: 利用者登録・編集画面のヒューリスティックス評価結果

問題点	違反しているヒューリスティックス
ただ地図が描かれているだけで農地登録の説明がない	ヘルプとマニュアル
管理者情報登録ページには項目のラベルがあるがこちらにはない	一貫性と標準化
ラジオボタンとラベルが一体化されていない。若干クリックしにくい	柔軟性と効率性
利用者がちゃんと登録できたのかが分かりにくいため、登録時に登録完了メールを送るべき	システム状態の視認性
氏名などは必須だが農地登録は必須でないなど、必須かそうでないかがわからない	エラーの防止
利用者一覧ページでは「すべての農地」だが、登録ページでは「すべての検知範囲」になっている	一貫性と標準化

表 5.6: ヒューリスティックス評価の分析結果

10 ヒューリスティックス	合計
システム状態の視認性	16
システムと実世界の調和	3
ユーザコントロールと自由度	4
一貫性と標準化	20
エラーの防止	2
記憶しなくても、見ればわかるように	6
柔軟性と効率性	6
美的で最小限のデザイン	13
ユーザによるエラー認識、診断、回復をサポートする	7
ヘルプとマニュアル	3
	80

ユーザビリティ・テスティング

被験者に提示したタスクの一覧を表 5.7 に示す。また、思考発話法を用いたことにより、被験者から得ることのできた管理者管理機能部分のフィードバックを表 5.8 に示し、フィードバックの分析結果を表 5.9 に示す。合計 39 項目のフィードバックを得た。その中で 16 項目のフィードバックを参考にして修正を行った。被験者は大学院生 6 名と JA 鈴鹿の職員 1 名に協力していただいた。

表 5.7: タスク一覧

項目番号	タスク内容
1	管理者向けページにログインする
2	新規に管理者を登録する
3	項目番号 2 で登録した管理者のパスワードを変更する
4	項目番号 2 で登録した管理者として再度ログインする
5	新規に利用者を登録する
6	利用者の登録されている農地を一つ削除する
7	新規に観測箱を登録する
8	項目番号 7 で登録した観測箱の位置を適当な場所に変更する
9	新規に観測動物を登録する
10	項目番号 9 で登録した観測動物の「備考」を変更する
11	新規に農作物を登録する
12	項目番号 11 で登録した農作物の「被害終了日」を変更する
13	「亀山猿群 A」を観測した検知履歴のみを表示する
14	受信した検知警告メールからシステムにアクセスし、サルの出没確率が高い地域を確認する
15	掲示板にメッセージを投稿する
16	項目番号 15 で登録したメッセージを削除する
17	マニュアルを表示する
18	農地アイコンを非表示にする

表 5.10 に表 4.4 と表 4.5 を用いた SUS の分析結果を示す。また式 4.2 を用いて算出したスケールの平均値は 69.6 であった。

表 5.8: 管理者管理機能のフィードバック

管理者のパスワード編集時に文字が隠されているため表示して確認したい
管理者が他の管理者を編集できるのはおかしい。スーパーユーザーを用意すべき
管理者番号 1 番が削除できない表示はいらない
「よみ」ではなく「ふりがな」
「氏名」と「管理者氏名」が表記ゆれ
管理者ごとに観測箱や農作物、観測動物を管理すべき
管理者トップ画面に管轄地点の住所を表示する
ログインユーザの情報が名前しかない。ログイン ID にすべき

表 5.9: フィードバックの分析結果

UI	機能	合計
30	9	39

表 5.10: SUS の分析結果

項目番号	質問内容	平均値
1	このシステムを今後も繰り返し使いたい	4.3
2	このシステムは必要以上に複雑だった	2.4
3	このシステムは簡単に使えた	3.6
4	このシステムを使えるようになるには、技術に詳しい人のサポートが必要だ	2.1
5	このシステムは機能がよくまとまっていた	4
6	このシステムは一貫性に欠けるところが多かった	2.3
7	たいていの人は、このシステムをすぐ使えるようになるだろう	3.6
8	このシステムはとても使いづらかった	2
9	このシステムを使っていて不安を感じることはなかった	3.7
10	このシステムを使い始める前に、多くのことを学ぶ必要があった	2.4

5.2.2　考察

- ユーザビリティ・インスペクション

  ユーザビリティ・インスペクションとしてヒューリスティックス評価を行い、発見した問題点の分析結果 (表 5.6) から「システム状態の視認性」、「一貫性と標準化」、「美的で最小限のデザイン」に反している問題点が多いことが分かる。しかし、評価は各画面毎に行ったため、同じ問題であっても画面が異なれば別の問題として計上されている。そのため、上で述べた 3 つの基準に対する問題点が多いが、問題点の種類の数としては他の基準と大きな差はない。本プロジェクトにおいてヒューリスティックス評価を行った結果、合計 80 項目という多くの問題点を発見した理由は画面設計の段階でガイドラインとして 10 ヒューリスティックスを用いていなかったことが考えられる。設計時に問題点を発見することができれば一度実装した後に修正を行うという工数を削減することができたと言える。しかし、実装後に確認してみないと発見できない問題点なども存在する。よって、ヒューリスティックス評価は画面設計終了後と各イテレーション開発終了後に行うべきだと言える。本プロジェクトにおける、画面設計時に発見できたと考えられる問題点として「観測箱を設置した後、クリックして削除できる旨の文が表示されない」が挙げられる。観測箱登録機能において、観測箱の位置を登録するために Google Maps 上にて観測箱アイコンを設置する。その際、観測箱アイコンを変更する場合、一度設置した観測箱アイコンをクリックして削除した後に変更する位置に新たに観測箱アイコンを設置する必要がある。設置した観測箱アイコンをクリックして削除できる旨の表示が設計時に抜けていたため、そのまま実装してしまった。そのため、設置した観測箱アイコンをマウスオーバーするとクリックして削除できる旨の文を表示するように修正した。修正した観測箱登録画面のスクリーンショットを図 5.1 に示す。

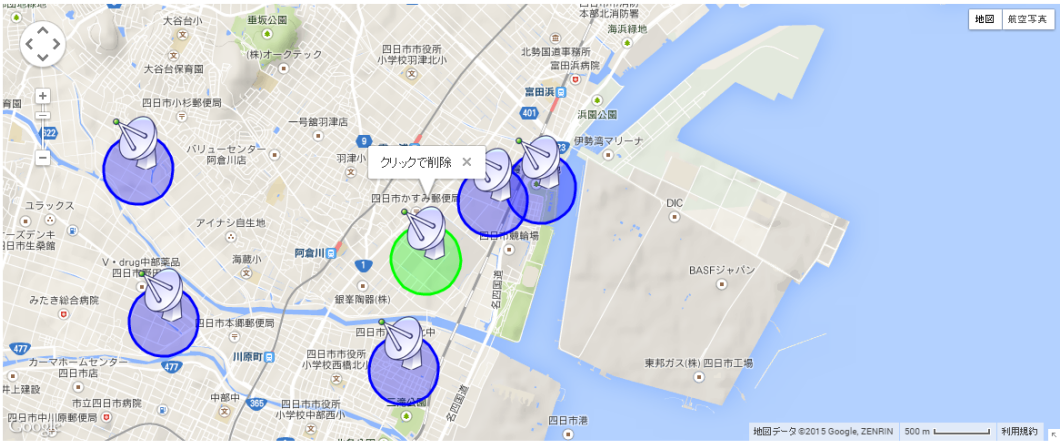


図 5.1: 修正した観測箱登録画面のスクリーンショット

また、実装後の確認が必要だったと考えられる問題点は「エラーメッセージが小さく

て分かりづらい」が挙げられる。画面設計時にはエラーメッセージは cakePHP のバリデーションチェックエラーでの表示だったが、実際にシステムを操作して確認したところ、エラーメッセージの表示が小さく分かりづらいという問題点を発見した。そのため jQuery を用いてポップアップ画面上にエラーメッセージを表示させるという修正を行った。修正したエラーメッセージ表示のスクリーンショットを図 5.2 に示す。

図 5.2: 修正したエラーメッセージ表示のスクリーンショット

- ユーザビリティ・テスティング

  ユーザビリティ・テスティングである思考発話法を用いて得られたフィードバックの分析結果 (表 5.9) から UI に関する意見が多いことが分かる。例えば、UI に関する意見として「マニュアルや掲示板の表示方法が分かりづらい」が挙げられる。ユーザビリティ・テスティングを行った時点においてマニュアルと掲示板はトップ画面にあるボタンをクリックすることで表示することができた。しかし、マニュアルと掲示板の表示をタスクとして提示すると、多くのユーザが表示方法が分かりづらいと回答した。そのため、トップ画面ではなくメニューバーの部分に表示するためのプルダウンを設置した。修正したメニューバーのスクリーンショットを図 5.3 に示す。

図 5.3: 修正したメニューバーのスクリーンショット

機能毎に分析してみると掲示板機能に関する意見が多かった。多かった理由は 2 点考えられる。一つは設計を開発メンバ内でレビューせずに実装に移ってしまった点である。

掲示板機能の担当者が開発メンバ内で設計をレビューせずに実装に移ったため、担当者の独断でレイアウトや動作を決定してしまった。もう一つは掲示板機能に対してヒューリスティックス評価を行わなかった点である。ヒューリスティックス評価は S0 終了後に行ったため S1 で開発した機能に対してはヒューリスティックス評価を行っていない。例えば、掲示板機能に関する機能の意見として「掲示板とアカウント管理を紐づけて簡単に投稿できた方が良い」が挙げられる。掲示板投稿画面のスクリーンショットを図 5.4 に示す。図 5.4 からも分かるように掲示板に投稿する為には投稿者名やメールアドレスを記入する必要がある。そのため、アカウント機能と連携して投稿者ごとにアカウント情報を管理すべきという意見が多かった。しかし、修正を行うための工数が足りないと開発メンバと判断し、修正は断念した。このような重要な問題点は開発メンバ内での設計のレビューの徹底と各イテレーション終了後のヒューリスティックス評価を行うことによる早い段階での発見が重要だと言える。

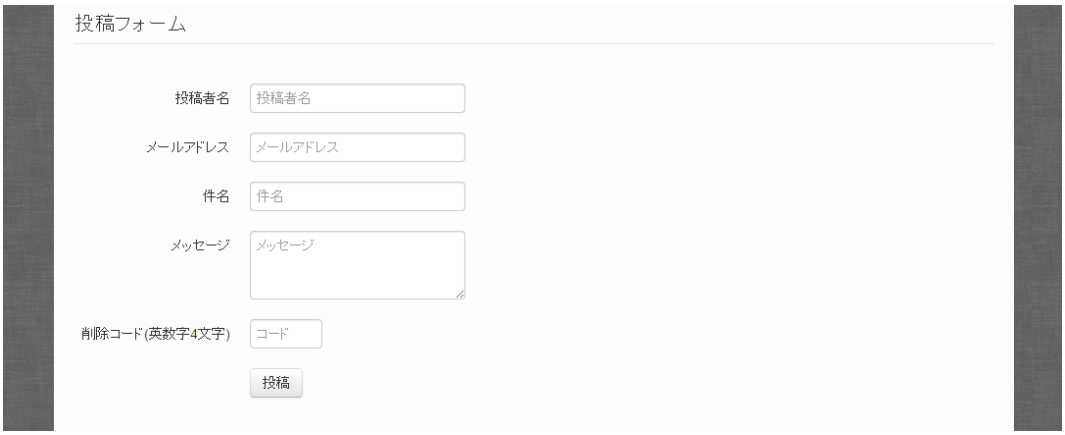


図 5.4: 掲示板投稿画面のスクリーンショット

また、SUS の分析結果 (表 5.10) からスケールの平均値が 69.6 であった。これは JeffSauro が求めたスケールの平均値である 68 を上回っているが、本プロジェクトで設定した目標である 75 を下回っている。しかし、被験者からのフィードバックを得た後に、それらの項目について修正を行ったため、得られたスケールよりもユーザビリティを向上させることができたと考えられる。修正後にもう一度ユーザビリティ・テスティングを行いたかったが、被験者への協力依頼や準備のコストや工数が足りないと判断し、断念した。

以上のことから、ユーザビリティ評価を行うことにより問題点を発見し、修正することでユーザビリティの向上を実現した。本プロジェクトにおいて実施したユーザビリティ・インスペクションとユーザビリティ・テスティングによってユーザビリティを正確に評価することができた。また、評価結果から改善すべき問題点の抽出が可能となり、ユーザビリティの向上につなげることができた。本プロジェクトにおいてこれらの評価手法を用いたことは妥当であったと言える。

第6章　おわりに

本プロジェクトではサルによる農作物被害を軽減するため、情報通信技術を利活用した獣害防止システムの開発を行った。

近年、全国的に獣害が深刻化している。顧客である JA 鈴鹿においても様々な獣害対策を行っており、防護柵設置による獣類の侵入防止、捕獲による個体数の管理と調整、発信機を活用したサルの生息調査と組織的な追い払いなどの対策が実施されている。この中で、最も効果的な方法は発信機を活用したサルの生息調査と組織的な追い払いである。現在 JA 鈴鹿は NPO 法人や猟師と連携して、サルに発信機付きの首輪を取り付け、その発信機の信号を受信することでサルの位置を捕捉している。しかし、この対策において JA 鈴鹿では 2 つの問題点を抱えている。1 つ目は受信機の運用者が必要であり、サルの位置捕捉するプロセスが自動化できていないこと、2 つ目は受信機を用いたサルの位置捕捉は属人性が高いことである。本プロジェクトではこれらの問題を解決する ICT ソリューションを提供した。

提供した ICT ソリューションは一定距離内へのサルの侵入を自動で検知する「観測機器」と、サルの位置情報を地図上に表示し、サルの検知情報を管理する「Web アプリケーション」から構成される。これらを用いることで以下の機能を実現した。

- サルの位置捕捉プロセスの自動化

  観測機器を山や農地周辺に設置することで、サルが観測機器周辺にいるかどうかを自動判別する。

- 複数の観測機器連携

  複数の観測機器を観測範囲が重なるように配置することによって、サルの位置捕捉の精度を向上させる。

- サルの位置捕捉情報を用いた生息調査

  観測機器からサーバに送られたサルの位置捕捉情報を蓄積することにより、その蓄積データとサルの行動特性からサルの動向を予測する。

- サルの位置捕捉情報の即時的な通知

  サルの位置捕捉情報を検知地域周辺の農地の所有者や猟師に電子メールを用いて通知する。

これらの機能の実現によって、上で述べた顧客が抱えている発信機を活用したサルの生息調査と組織的な追い払いの問題を解決することができる。

筆者による本プロジェクトへの貢献の一つは Web アプリケーションのテストである。テストを行うことにより、開発したシステムが仕様通りに動作するかどうかを確認することができる。本プロジェクトにおいては単体テスト、結合テスト、受け入れテストを行った。また、テストの十分性を確保するためにホワイトボックステスト手法とブラックボックステスト手法を用いた。しかし、テスト担当者を筆者のみとしたこと、多くの仕様変更に柔軟に対応してテストを実施できなかったことなどの問題を抱えてしまった。このことから、テストは実装者と協力して行うべきであること、仕様変更の発生をなるべく少なくするために設計の段階で細かく仕様を決定し、定期的にチーム内レビューをすることで仕様漏れなどを早期の段階で発見できるようにする必要があると分かった。

もう一つの筆者による本プロジェクトへの貢献はユーザビリティの向上である。近年の情報処理機器の急速な普及に伴い、ユーザビリティの重要性が高まっている。本プロジェクトにおいてはユーザビリティ・インスペクション及びユーザビリティ・テスティングによりユーザビリティを評価し、発見した問題を改善した。評価結果として多くのユーザビリティの問題を発見することができ、これらを改善することでユーザビリティの向上を実現できた。なお、ユーザビリティを数値化する SUS を行った結果、目標値を満たすことが出来なかった。しかし、SUS を実施した時に発見された問題点を改善することができたため、ユーザビリティの水準は最低限保つことができた。

今後の展望として、サルの追い払いの自動化が挙げられる。本システムではサルの位置を推定し、その情報を農地関係者に通知する機能のみである。追い払いを自動化することができれば、農地関係者への負担をさらに軽減させることができる。これを実現させるためにはさらなる位置推定の精度向上とマルチコプターなどを用いた追い払いが必要となる。また、獣害はサルだけではなく、シカやイノシシによっても発生しているため、これらの獣類への対応も検討する必要がある。これを実現させるためにセンシング技術や画像処理技術が有効であると考えている。以上のような改良を行うためには複数年に渡って産官学が連携し、プロジェクト体制を整えることが重要であると考えられる。

謝辞

JA 鈴鹿営農指導課の川出洋正様には獣害対策関係者各位への御声掛けや現地実験の提供など多大な支援を頂きました。心より感謝致します。本プロジェクトを進めるにあたり、指導教員の高橋伸准教授、並びに田中二郎教授をはじめとしたインタラクティブプログラミング研究室の先生方には多くの御指導や出張の支援を頂きました。厚く御礼申し上げます。また、課題担当教員である和田耕一教授と山際伸一准教授には、ミーティングにて熱心に御指導頂き、報告書や論文の執筆で大変お世話になりました。深く感謝致します。また、ユーザビリティ・テスティングに協力して頂いた筑波大学大学院の学生の皆様へ心から感謝致します。プロジェクトメンバである、天本涼太氏、森恒成氏、岡部健信氏、光岡遼氏とは共にプロジェクトを進め、多くの意見や助言をして頂きました。ありがとうございました。最後にこれまで私を支えて下さった、家族、友人に心より感謝申し上げます。

参考文献

[1] 農林水産省. 鳥獣被害対策の現状と課題. http://www.maff.go.jp/j/seisan/tyozyu/higai/pdf/261202_meguji.pdf (閲覧日：2014 年 12 月 17 日)

[2] 農林水産省. 鳥獣被害の現状と要因. http://www.maff.go.jp/j/seisan/tyozyu/higai/h_manual/h18_03/pdf/data1.pdf (閲覧日：2014 年 12 月 17 日)

[3] 総務省. 捕獲に関する統計資料集計. https://www.env.go.jp/nature/choju/capture/pdf/d1.pdf (閲覧日：2014 年 12 月 17 日)

[4] 鈴鹿市. 鈴鹿市鳥獣被害防止計画. http://www.city.suzuka.lg.jp/kouhou/gyosei/plan/keikaku/kakusyu/data/chojyu/plan_01.pdf (閲覧日 : 2014 年 12 月 17 日)

[5] 農林水産省. 鳥獣被害防止総合対策交付金. http://www.maff.go.jp/j/seisan/tyozyu/higai/pdf/h26_tousho_pr.pdf (閲覧日：2014 年 12 月 18 日)

[6] 農林水産省. 抜本的な鳥獣捕獲強化対策. http://www.maff.go.jp/j/seisan/tyozyu/higai/pdf/kyouka.pdf (閲覧日：2014 年 12 月 18 日)

[7] 農林水産省. 野生鳥獣被害防止マニュアル. http://www.maff.go.jp/j/seisan/tyozyu/higai/h_manual/h26_03/pdf/data0_6.pdf (閲覧日：2014 年 12 月 18 日)

[8] 三重県農業研究所. ニホンザルの被害対策. http://www.mate.pref.mie.lg.jp/marc/jyugai/pdf/sarunohigaitaisaku.pdf (閲覧日：2014 年 12 月 18 日)

[9] 三重県農業研究所. イノシシの被害対策. http://www.mate.pref.mie.lg.jp/marc/jyugai/pdf/inosisinohigaitaisaku.pdf (閲覧日：2014 年 12 月 18 日)

[10] 三重県農業研究所. シカの被害対策. http://www.mate.pref.mie.lg.jp/marc/jyugai/pdf/sikahigaitaisaku.pdf (閲覧日：2014 年 12 月 18 日)

[11] 農林水産省. 被害防止対策. http://www.maff.go.jp/j/seisan/tyozyu/higai/h_manual/h21_03/pdf/data1.pdf (閲覧日：2014 年 12 月 18 日)

[12] NPO 法人サルどこネット. http://www.sarudoko.net (閲覧日 : 2014 年 12 月 18 日)

[13] 戸田春香. 三重県亀山丘陵におけるニホンザルの行動特性と猿害. 地理学評論, 80-11, 614-634, 2007

[14] 黒川利明, 品川萬里. ソフトウェア・テストの技術動向と課題. 科学技術動向, 085, 8-18, 2008

[15] 深谷直彦, 古川善吾, 西康晴編. ソフトウェアテストの最新動向. 情報処理, Vol.49, No.2, 126-173, 2008

[16] 八戸文仁, 伊藤恵. ソフトウェアテスト自動化を改善する為のテスト手法. 電気・情報関係学会北海道支部連合大会, 112, 158-159, 2006

[17] David Saff, Shay Artzi, Jeff H. Perkins,Michael D. Ernst. Automatic test factoring for Java. ASE 2005: Proceedings of the 20th Annual International Conference on Automated Software Engineering, 114-123, 2008

[18] 岡村正司. ソフトウェア開発における単体テストの生産性向上 -C1 テストケースの有効性-. プロジェクトマネジメント学会 2001 年度春季研究発表大会, 2-8, 124-129, 2001

[19] Shoji Yuen, Keishi Kato, Daiju Kato, Kiyoshi Agusa. Web Automata: A Behavioral Model of Web applications based on the MVC model. Information and Media Technologies, Vol.1, No.1, 66-79, 2006

[20] Gerard Meszaros. xUnit Test Patterns:Refactoring Test Code. Pearson Education, 2007

[21] 湯本剛, 大西建児, 勝亦匡秀 , 佐々木方規, 鈴木三紀夫, 町田欣史, 吉澤智美, 中野直樹. ソフトウェアテスト教科書 JSTQB Foundation 第 3 版. 翔泳社, 2012

[22] 丸地康平, 酒井政裕, 進博正. シーケンス制御プログラムのテストに適した新しいカバレッジ基準の提案. 情報処理学会研究報告, 2013-SE-179, 17, 1-8, 2013

[23] John J.Chilenski, Steven P.Miller. Applicability of modified condition/decision coverage to software testing. Software Engineering Journal Vol.9,1994

[24] 飯田志津夫. 基幹システムにおけるテスト十分性の確保. Unisys 技法, Vol.28, No.4, 513-525, 2009

[25] 詳説 FV 表. `http://www.hayst.com/Documents/FV-table.pdf` (閲覧日：2014 年 12 月 25 日)

[26] 守屋慎次, 斉藤剛, 平松啓二. デシジョンテーブルの完備性と重複性. 情報処理学会論文誌, Vol.21, No.5, 1980

[27] 岡田英彦. ユーザビリティとその評価手法. システム制御情報学会, Vol.45, No.5, 269-276, 2001

[28] 業務アプリケーションについてのアンケート. メイド・イン・ジャパン・ソフトウェアコンソーシアム, 2007

[29] 鹿島泰介. クラウド時代のユーザインタフェースの方向性-サービスサイエンスで UI の方向性を検証する. 情報処理学会デジタルプラクティス, Vol.1, No.1, 13-20, 2010

[30] ISO/IEC 9126-1:Software engineering - Product quality - Part 1:Quality model. 2001

[31] 樽本徹也. ユーザビリティエンジニアリング -ユーザエクスペリエンスのための調査、設計、評価手法-. オーム社, 2005

[32] Nielsen Norman Group. Usability Heuristics for User Interface Design. http:www.nngroup.com/articles/ten-usability-heuristics/ (閲覧日：2014 年 12 月 25 日)

[33] 阪井誠, 中道上, 島和之, 中村匡秀, 松本健一 WebTrace:視線を利用した Web ユーザビリティ評価環境. 情報処理学会論文誌, Vol.44, No.11, 2575-2586, 2003

[34] Nielsen Norman Group. Why You Only Need to Test with 5 Users. http://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/ (閲覧日：2014 年 12 月 25 日)

[35] 松永志野. ライティング・プロセス探索を目的とする思考発話法による実験方法に関する研究. 熊本大学社会文化研究, 249-260, 2011

[36] Jeff Sauro, James R.Lewis. Quantifying the User Experience: Practical Statistics for User Research. MORGAN KAUFMANN PUBL INC, 2012

[37] John Brooke. SUS - A quick and dirty usability scale. http:www.itu.dk/courses/U/E2005/litteratur/sus.pdf (閲覧日：2014 年 12 月 25 日)

[38] Jeff Sauro. Measuring Usability With The System Usability Scale. http://www.measuringu.com/sus.php (閲覧日：2014 年 12 月 25 日)

付録

目次

A. 要件定義書 49
B. 設計書類 67
1.画面設計書 67
2. 観測箱シーケンス図 91
3. ER 図 92
C. プロジェクト文書 93
1. WBS 93
2. ガントチャート 96
3. 作業管理表 99
4. リスク管理表 101
D. マニュアル 107
E. アンケート 137
1. 中間顧客アンケート 137
2. マニュアルとプロジェクト評価のアンケート 139
3 顧客満足度調査 141
F. 評価実験 145
G. テスト 159
1. FV 表 159
2. デシジョンテーブル 160
3. 要素別動作確認表 170
H. ユーザビリティテスト 183
1. ヒューリスティック評価マニュアル 183
2. ヒューリスティック評価結果 185
3. ユーザビリティテスティング 186
4. ユーザビリティテスティングマニュアル 192
5. 同意書 195
I. 検収書 196

2014 年
6 月 6 日
第 1.5 版

JA 鈴鹿 - 筑波大学大学院
「研究開発プロジェクト」
要件定義書 兼 作業範囲記述書
（要件定義～必須機能開発フェーズ）

情報通信技術を利活用した
農作物向け獣害防止システムの開発

担当教員：和田耕一、山際伸一

チーム：A=M+H
天本涼太
岡部健信
浜野悠介
光岡　遼
森　恒成

目次

1. 本書類について 1

2. プロジェクトについて 2

2.1. プロジェクトの概要 2

2.2. プロジェクトの目標 3

2.3. 関係者一覧とその役割 4

3. プロジェクトの背景 5

3.1. 農作物への被害の現状 5

3.2. 獣害防止策の現状と課題 6

3.3. システム化の目的 7

4. システムの概要 8

4.1. システムの構成 8

4.2. システムの運用イメージ 9

4.3. システム化の対象範囲 10

4.4. WEB サービスの概要 12

4.5. システムの費用と効果 13

4.6. 概略スケジュール 14

4.7. システムの実験と評価・改良 14

5. 関係者間の連絡方法 15

6. 参考資料 16

▲目次に戻る

1. 本書類について

本書類は、「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ソフトウェア開発専修プログラム」における、研究開発プロジェクトのテーマ、「情報通信技術を利活用した農作物向け獣害防止システムの開発」についての要件定義書、兼、作業範囲記述書です。

本書類中では、JA 鈴鹿をお客様として、筑波大学院の学生 5 名が主体で実施するプロジェクトの概要や目標、システム化の背景やシステムが対象とする範囲、そのスケジュールなどについて詳細を記載いたします。

なお、本書類は以下の 3 つの目的のために、作成・配布しています。

①　プロジェクトに対する、関係者間の情報共有のため
②　開発するシステムについて、関係者皆様からの貴重なご意見をいただくため
③　プロジェクトの成果物や評価方法について、JA 鈴鹿からの合意を得るため

本書類に記載されていることに関してご意見・ご要望・修正依頼がある場合には、6. 関係者間の連絡方法をご参照のうえ、「筑波大学大学院　天本涼太」までご連絡ください。

以上について、関係者皆様のご理解とご協力をどうぞ宜しくお願いいたします。

2. プロジェクトについて

2.1. プロジェクトの概要

本プロジェクトでは､JA 鈴鹿をお客様として､筑波大学大学院の学生 5 名が主体となって、以下の要件の観測箱システム・WEB サービスを開発いたします。

【観測箱システムについて】

① テレメトリ発信機が取り付けられた野生動物の出没区域に設置することで、設置位置から半径 500m 程度の野生動物の現在位置を、リアルタイムに可視化することができる観測箱システムを開発します。

② 観測箱システムは、複数機連携して設置・動作設定を行うことができます。この複数機連携により、猿の位置情報の推定をより正確に割り出すことができます。

【WEB サービスについて】

① 野生動物が観測箱システム（＝農地の近く）に近付いたことを関係者に E メールで自動的にお知らせすることができます。このお知らせのために、あらかじめサービスの利用者を登録することができる WEB サービスを開発します。

② この WEB サービス上で､観測箱システムの設置位置の近くにいる野生動物の現在位置を地図上で確認することができます。

③ この WEB サービス上で、観測箱システムの動作記録（野生動物の観測域への侵入履歴）を閲覧することができます。

□ 本プロジェクトでは、野生動物の「位置検出とそのデータ化」に焦点を当てており、野生動物の「自動撃退」は視野に入れていません。機能の詳細は 4.システムの概要でご確認ください。

2.2. プロジェクトの目標

本プロジェクトでは、次の 3 つの要件を満たすことを目標としています。

① 成果物（現行では、2.1.プロジェクトの概要、および、4.システムの概要で記載しているシステム）について、JA 鈴鹿のご担当者様から合意をいただくこと

② 合意をいただいた成果物を、プロジェクトメンバが期限内・予算内に完成させること

③ 完成した成果物について実験を行い、その有用性や使い勝手などについて、JA 鈴鹿のご担当者様から評価をいただくこと

本プロジェクトの成果物の位置付けは「将来的に、野生動物による農作物の被害を減らすためのシステムの実験機」となっています。従って、本プロジェクトの目標は「実験機の開発と評価」に焦点を当てています。この評価が良い結果になった（野生動物による農作物の被害を減らすのに役立つことが証明された）場合に、「野生動物による農作物の被害を減らすこと」と「システムの量産化」を新たな目標として、システムを改良していくことになります。

図 1 に、本プロジェクトの位置付けを示します。

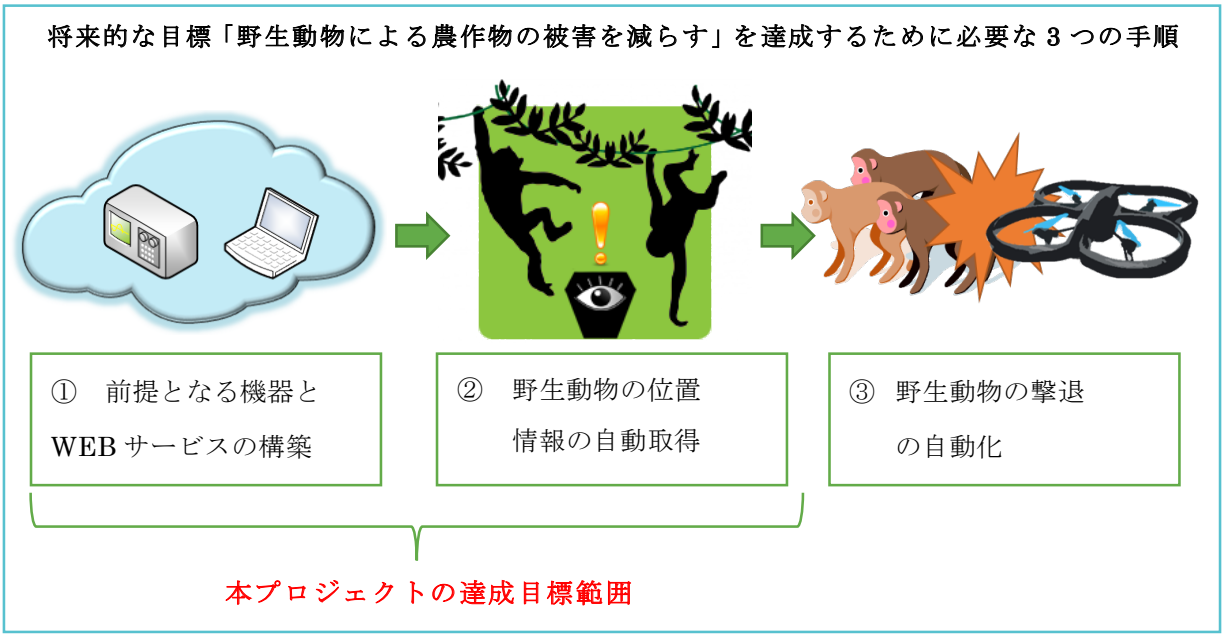


図 1. 「野生動物による農作物の被害を減らす」ために必要な手順と、本プロジェクトの位置付け

2.3. 関係者一覧とその役割

以下の表 1 に、本プロジェクトの関係者一覧（順不同敬称略）を示します。表 1 には所属と氏名、そして、本プロジェクトにおいて期待されると思われる役割を記載しています。

表 1　関係者一覧

所属	氏名	役割
鈴鹿市農林水産課	服部　慎也	情報のご提供【獣害被害・獣害対策について】
鈴鹿市農林水産課	田中　瑛	情報のご提供【獣害被害・獣害対策について】
四日市鈴鹿地域農業改良普及センター	辻　慎司	情報のご提供【獣害被害・獣害対策について】
亀山市環境産業部農政室	宮崎　哲二	情報のご提供【獣害被害・獣害対策について】
亀山市環境産業部農政室	宮村　一郎	情報・機材ご提供【テレメトリ発信機について】
四日市農林事務所森林・林業室	喜多村　雅夫	情報のご提供【林業における獣害について】
杣 SomaPlanning	岡安　保郎	情報のご提供【林業における獣害について】
鈴鹿工業高等専門学校	大津　孝佳	技術、知的財産に関するアドバイス
JA 鈴鹿　営業部	丸市　弘和	成果物についてのご意見、評価などの提供
JA 鈴鹿　営農指導課	長谷川　清	成果物についてのご意見、評価などの提供
JA 鈴鹿　営農指導課	川出　洋正	成果物についてのご意見、評価などの提供
JA 鈴鹿　営農指導課	栗田　幸江	JA 鈴鹿 － 筑波大学間のご連絡の担当
三重県議	藤田　宣三	成果物についてのご意見、評価などの提供
NPO　サルどこネット	明石　武美	情報・機材ご提供【テレメトリ発信機について】
筑波大学	和田　耕一	プロジェクトの担当教員【技術指導など】
筑波大学	山際　伸一	プロジェクトの担当教員【技術指導など】
筑波大学	天本　涼太	プロジェクトメンバ【プロジェクトリーダ】
筑波大学	岡部　健信	プロジェクトメンバ
筑波大学	浜野　悠介	プロジェクトメンバ
筑波大学	光岡　遼	プロジェクトメンバ【プロジェクトサブリーダ】
筑波大学	森　恒成	プロジェクトメンバ

3. プロジェクトの背景

3.1. 農作物への被害の現状

野生動物（獣類・鳥類）による農作物への被害の増加は全国的な課題であり、三重県全体では、平成 24 年度で約 3 億 9 千万円の被害が報告されています[i]。野生動物による農作物被害のうち、最も大きい動物 3 種類を次に示します。

① 猪による被害：約 1 億 5 千万円（39%）

② 猿による被害：約 1 億 3 千万円（32%）

③ 鹿による被害：約 0 億 8 千万円（22%）

鈴鹿市では、平成 24 年度でそれぞれ、

① 鹿による被害：581 万円

② 猿による被害：433 万円

③ 猪による被害：428 万円

の被害が出ています[ii]。また、野生動物による被害は農作物だけではなく、次のような被害が報告されています。

A) 猿：民家や倉庫の雨どい、かわらの破損、人への被害

B) 鹿：杉、ヒノキの食害といった林業への被害、車との衝突

C) 猪：ゴルフ場の芝の掘り起こし、水田の踏み倒し、（電気）柵の破壊

従って、野生動物が人里へ降りてこないようにすることで、これらの被害を防ぐことが可能だと考えられています。なお、野生動物による被害が増えている原因としては、①都市開発などにより野生動物の生息環境が変化したこと、②里山での人間活動の低下、③耕作放棄地の増加などにより、人里周辺が生息しやすい場所となったことが考えられています。そうして農作物に被害が発生すると、農業者の営農意欲が低下し、耕作放棄地の増加に繋がるという悪循環が発生しています。

3.2. 獣害防止策の現状と課題

【現状】

鈴鹿市では、平成25年度の取り組みとして、国の交付金を含めた予算200万円～を計上し、以下のような取り組みを行っています。

① 猟友会との連携による野生動物の個体数調整

② 電波発信機を用いた猿の位置情報確認・生息分布域調査

③ 地域住民の獣害対策に関する意識の啓発を行う講習会などの開催

【課題】

野生動物からの獣害を減らすためには、町ぐるみの野生動物追い払いや、野生動物の捕獲・監視による個体数の調整が有効です。しかし、これらの対策を実施するためには野生動物の居場所をリアルタイム、かつ、正確に知る必要があります。そのためには上記②の「電波発信機を用いた猿の位置情報確認」が有効ですが、電波発信機はいくつかの課題を抱えています。この課題について、類似のシステムである「猿人善快 - 猿の接近警戒システム」の特許から課題を引用します。

> 従来の鳥獣被害対策は駆除中心に行われていたが、その被害は年々増加し、一方では自然保護、種の保存という観点で保護対象とされる鳥獣類も多く、決定的な被害対策が見いだせずにいた。一部に、テレメトリ発信器を使用して、その発信電波を利用することで、その接近を知ろうとする受信機の利用もあったが、その受信機の利用については、ある程度の熟練、すなわち運用技術を必要とし、また常に受信状態を気にしていないとテレメトリ信号の受信に気づかないと言う問題があり、誰でもが有効に発信電波が利用できる状況ではなかった。テレメトリ発信器の電波受信に際しては、雑音などの影響も大きく、また住民が日頃の生活空間のなかで、接近を予知して対応できるほどの受信感度（感知距離）が得られていなかった。そして、接近を知ったときの対応も、行政窓口への連絡で終わってしまう場合がほとんどで、その場での対応がとられていない、すなわち野生動物の接近に対して即効的な対処ができていなかった。

課題①

課題②

課題③

これらの課題を解決することができれば、野生動物の生息分布域の調査が容易に進み、既存の電気柵といった獣害対策を効果的に導入することができるようになります。また、人里に近付いてくる野生動物を自動的に追い払う、といった新しい解決方法も導入が可能になります。

3.3. システム化の目的

これまでの情報を踏まえて、私たちは「電波発信機を用いた猿の位置情報確認」を発展させたシステムを開発します。具体的には、システムに事前登録を行った誰もが猿の位置をリアルタイムに、かつ正確に確認でき、猿が農地に接近していることを関係者に連絡することで即時的な行動を促すシステム（観測箱システム＋WEB サービス）を開発します。またこのシステムは、猿の位置情報を継続的に監視することで、猿の行動履歴を閲覧することもできます。

以下に、本プロジェクトによるシステム化の目的を示します。

① 農家が、自分の農地を自分で守ることができるようにするため。「自分の所有する農地に近付いた」猿の位置をリアルタイム、かつ、自動的に知ることで、農作物被害を未然に防ぐことができるようになります。
② 猿による獣害を減らし、農産業の発展を促すため。観測箱近辺に出没した猿の行動記録を確認することで、既存の獣害対策をより効果的に実施することができるようになります。

4. システムの概要

4.1. システムの構成

図 2 にシステムの構成図を示します。

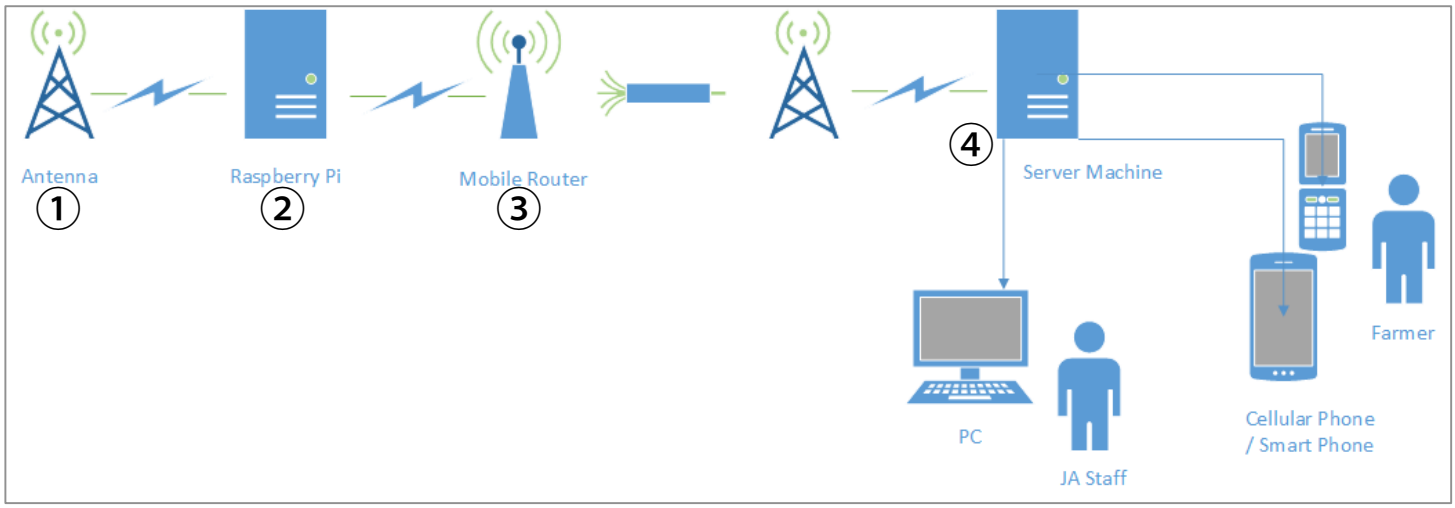


図 2. システムの構成図

① 【Antenna】猿につけられているテレメトリ発信機の電波を受信するためのアンテナ（＋電波受信機）です。今回使用するテレメトリ発信機の型番は LT-01、電波受信機の型番は STD-T99 で、仕様はこちらに記載されています。

② 【RaspberryPi】観測箱システムです。電波の解析部、信号制御部、電源部から構成されます。

③ 【Mobile Router】観測箱システムから猿の検知情報がインターネットに送信されます。

④ 【Server Machine】観測箱システムから送付された猿の検知情報を用い、猿の位置情報を計算します。そして、WEB サービスを通じて PC や携帯端末に猿の位置情報を表示させたり、関係者に E メールを送付したりします。

4.2. システムの運用イメージ

図 3 にシステムの運用イメージ図を示します。

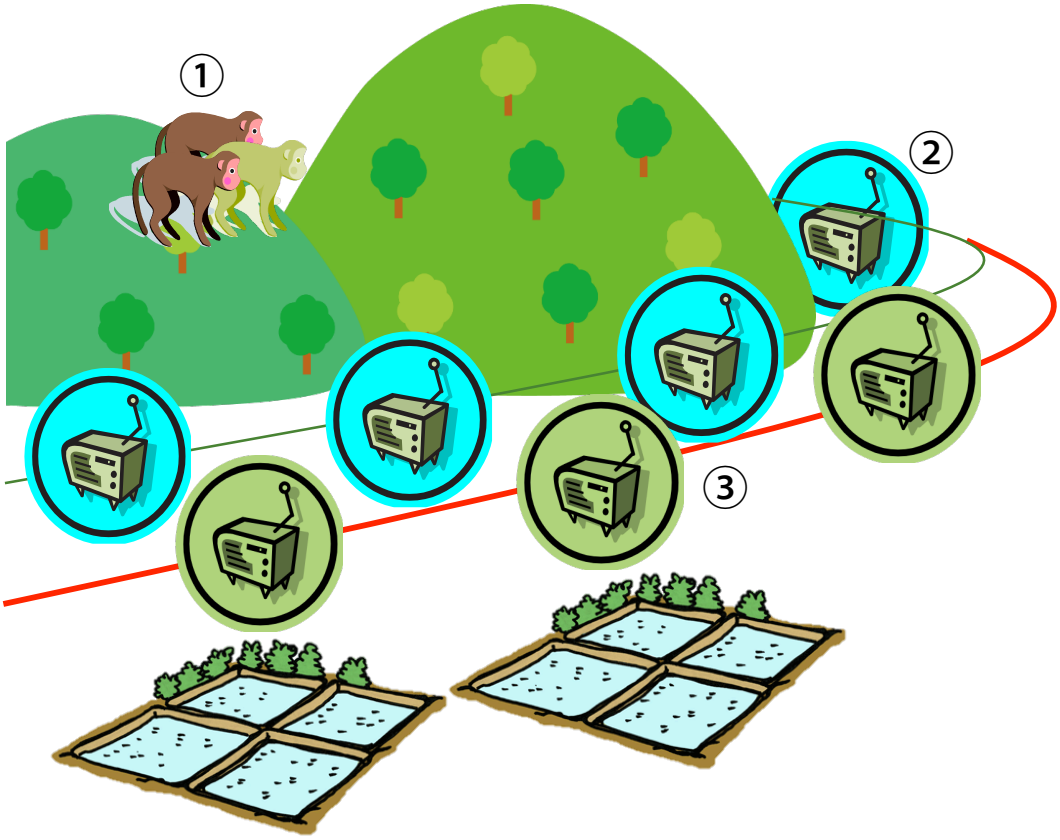


図 3. システムの運用イメージ図

① 【猿の群れ】今回のシステムで撃退の対象となる野生動物です。三重県鈴鹿市においては、群れの中心になっている猿に、電波を送信するテレメトリ発信機がついています。今回のシステムでは、このテレメトリ発信機の電波を利用します。

② 【観測箱システム】観測箱システムは複数機での連携動作を想定しています。本プロジェクトでは、図 3 のように、猿の出没区域と農地の境界線に複数列並べて動作させることを想定しています。将来的には、山の中に網目状に観測箱システムを設置することで、猿の位置を常時観測できるようにすることも視野に入れています。なお、本プロジェクトでは実験機として、観測箱システムを 3 つ製作し、動作確認・評価を行う予定です。

③ 【観測箱システム：警告ライン】農地側の観測箱システムのラインです。このラインの観測箱システムが猿の侵入を受信した場合に、利用者に E メールを送信します。

4.3. システム化の対象範囲

この節では、本プロジェクトにおける、システム化の対象範囲を示します。最初に、どういった前提条件のもとにシステムを開発するのかを述べます。次に、農家の皆様や JA の管理者様がそのシステムで何ができるのか、どんな意味があるのかを記載します。最後に、このプロジェクトでシステム化を対象外としていることも記載します。

【前提条件／制約条件】

① 観測箱システムの設置場所では、電源が確保できるものとする

② 観測箱システムの設置場所では、3G ネットワークに接続できるものとする

③ 観測箱システムの運用は、晴天、および、曇の天候状態であるものとする

【システム化の対象内】

◆ **農家ができること**

1 WEB サービスに登録した自分の農地に近付いた猿の位置をリアルタイム、かつ、自動的に知ることができる。それは農作物を猿から守るためである

1.1 登録したメールアドレスで、自分が所有する農地に近づいている猿の現在位置が記載されたメールを受信できる

1.2 WEB にアクセスすることで、猿の現在位置を地図で確認することができる

- パーソナルコンピュータの WEB ブラウザから確認することができる
- モバイル端末(スマートフォン)から確認することができる

◆ **猟友会といった猿の位置を知りたい利用者ができること**

1. 観測箱システムの近辺にいる猿の位置をリアルタイム、かつ、自動的に知ることができる。それは猿の発見・捕獲といった作業をより効率的に実施するためである

1.1. 登録したメールアドレスで、設置されている全ての観測箱システムのいずれかに近づいている猿の現在位置が記載されたメールを受信できる

1.2. WEB にアクセスすることで、猿の現在位置を地図で確認することができる

- パーソナルコンピュータの WEB ブラウザから確認することができる
- モバイル端末(スマートフォン)から確認することができる

◆ **JA のシステム管理者ができること**

1　観測箱システムの近辺にいる猿の位置をリアルタイムに確認することができる。それは猿による獣害を減らし、農産業の発展を促すためである

1.1　猿についているテレメトリ発信機の周波数の帯域別に位置情報を確認することができる

1.2　猿の群れがいる確率が高い場所を視覚的に確認することができる

1.3　観測箱のシステムが稼働しているかどうかを確認することができる

2　猿の接近お知らせサービスを利用する農家や一般利用者をシステムに登録することができる。それは農家の所有する農地への猿の接近のお知らせを自動化するためである

2.1　ログイン ID とパスワードを用いることで管理者画面へログインできる

2.2　管理者情報（アカウント名、パスワード）を編集することができる

- 管理者を追加, 変更, 削除することができる

2.3　農家情報（姓名、農地の位置、メールアドレス、お知らせの有無、観測箱システムとの結びつき）を編集することができる

- 農家情報を追加, 変更, 削除することができる

2.4　観測箱情報（観測箱の番号、観測箱の設置位置）を編集することができる

- 観測箱情報を追加, 変更, 削除することができる

3　猿の出現頻度や出現位置の記録を参照することができる。それは猿の行動パターンを体系的に把握することで、今後の獣害対策に活かすためである

3.1　システムの動作記録を閲覧することができる

- 特定の観測箱システムの動作記録を絞り込んで閲覧することができる
- 猿の検出記録を絞り込んで閲覧することができる

【システム化の対象外】

◆ 農地に近づいてくる猿の自動撃退

◆ 観測箱システムを網目状に設置することによる、猿の位置の常時可視化

◆ 長期的なハードウェアの運用機能（商用電源の引き込み、防塵防水ケース等）

4.4. WEB サービスの概要

図 4 に開発予定の WEB サービスの画面遷移図を示します。この図は「どんな画面があって、どういった条件で画面が遷移し、各画面ではどのようなことができるのか」を示しています。WEB サービスのイメージを深めるために、もし宜しければご参照ください。

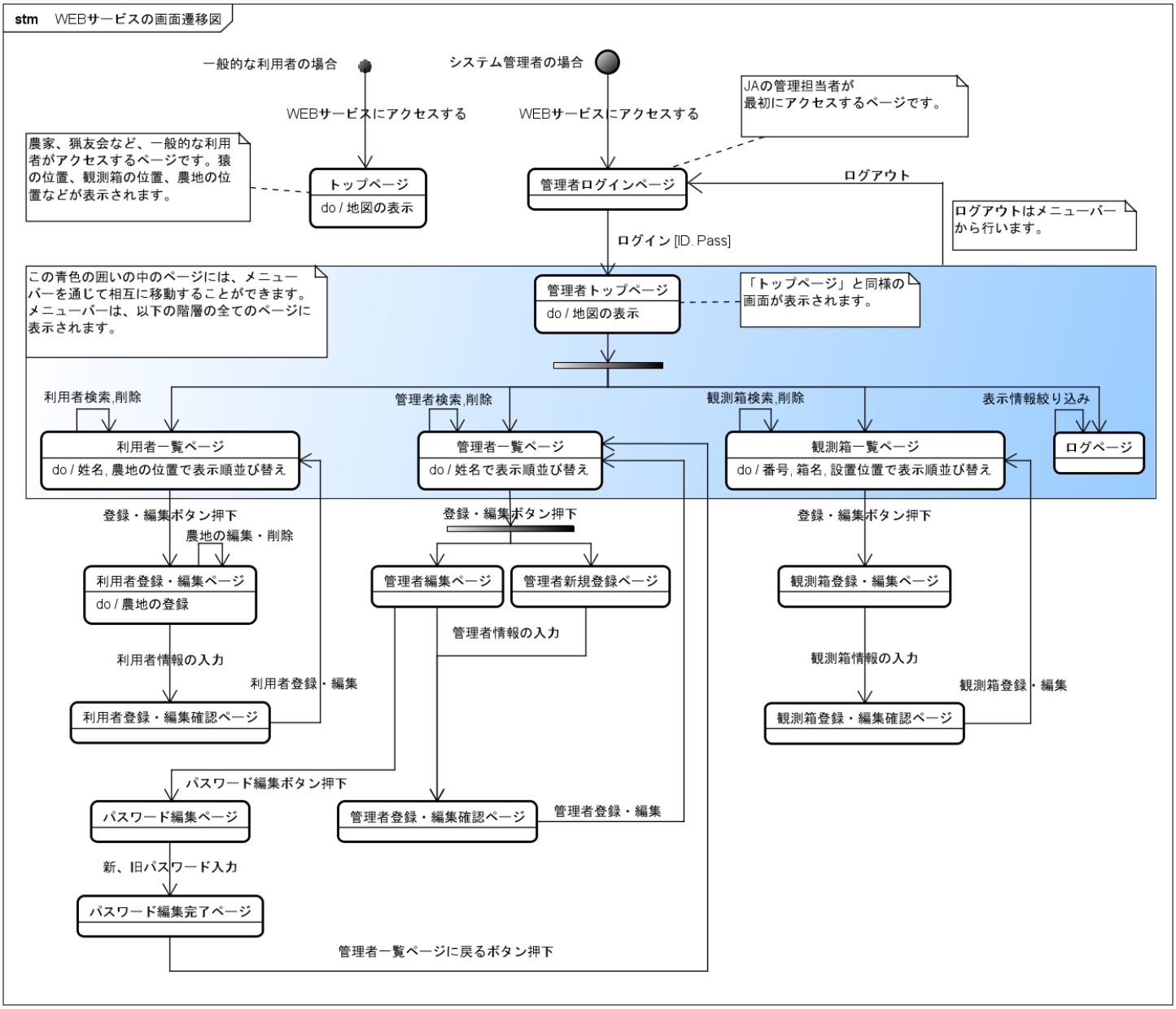


図 4. WEB サービスの画面遷移図

4.5. システムの費用と効果

以下に、本プロジェクトのシステム開発にかかる費用を記載いたします。

【システム構築に掛かる費用】

- 観測箱システム：約 59,000 円／実験機 1 つ当たり
  - 信号の受信機(型番が DJ-X11 の場合)：約 41,000 円
  - ダイポールアンテナ：約 5,000 円
  - RaspberryPi（ワンボードコンピュータ）：約 4,000 円
  - モバイルルーター：（月額）2,000 円
  - モバイルバッテリー：約 5,000 円
  - その他の周辺機器：約 2,000 円
- WEB サービス：0 円
- 合計（観測箱システム 3 機＋WEB サービス）：約 177,000 円

なお、上記の金額はあくまで実験機にかかる費用です。したがって、業者等と連携して本格的にシステムを量産することになった場合には、費用が大きく変動します。たとえば、信号の受信機は、回路を自作することで費用を数千円程度にまで削減できる見込みです。

【システム導入によって期待される効果】

1. 観測箱システムを設置した近辺（半径 500m 程度）に猿が侵入したことを、自動的に検知できるようになります。また、その情報を E メールや WEB サービス上で確認することができます。これによって、農家が自分の農地の農作物を守りやすくなります。
2. 複数の観測箱システムの検知範囲内に侵入した猿の位置情報は、観測箱ひとつで検知した場合よりも正確に取得することができます。これによって、猟友会や JA の担当者が猿を発見・追跡することが簡単になります。

4.6. 概略スケジュール

表 2 に概略スケジュールを示します。

作業名	5 月	6 月	7 月	8 月	9 月	10 月	11 月
要件定義	→				夏季休暇		
システム開発		→					
実験と評価			→	→			
システム改良				→		→	
納品／運用							→

4.7. システムの実験と評価・改良

システムの第 1 回実験は、7 月初頭に筑波大学内での実施を予定しています。

その実験結果は JA 鈴鹿に送付し、実験結果、およびシステムについてレビュー（フィードバック）をいただく予定です。このレビュー期間は 2 週間程度を想定しています。そのレビュー結果を元に、追加・修正する機能を決定し、8 月, 10 月に渡ってシステムの改良を重ねていきます。そして 11 月頃、三重県鈴鹿市に実験機 3 機を設置し、システムの第 2 回実験（実地運用試験）を行う予定です。JA 鈴鹿の皆様には、その運用試験にご協力をいただきたいと考えています。お忙しいところ恐縮ですが、システムの実験と評価についてのご協力をどうぞよろしくお願いいたします。

※実験については、詳細が決まり次第またご連絡差し上げます。

5. 関係者間の連絡方法

「表 1. 関係者の一覧」で示した皆様の連絡先を登録したメーリングリストを作成いたしました。

[redacted]

にメールを送信していただければ、関係者の皆様全員に同じ文面のメールを届けることができます。今後、本書類や本プロジェクトについて何かご意見がありましたら、必要に応じてこのメーリングリストをご利用ください。

なお、このメーリングリストから登録を外していただきたい方がいらっしゃいましたら、筑波大学大学院　天本涼太：[redacted]までご一報ください。

6. 参考資料

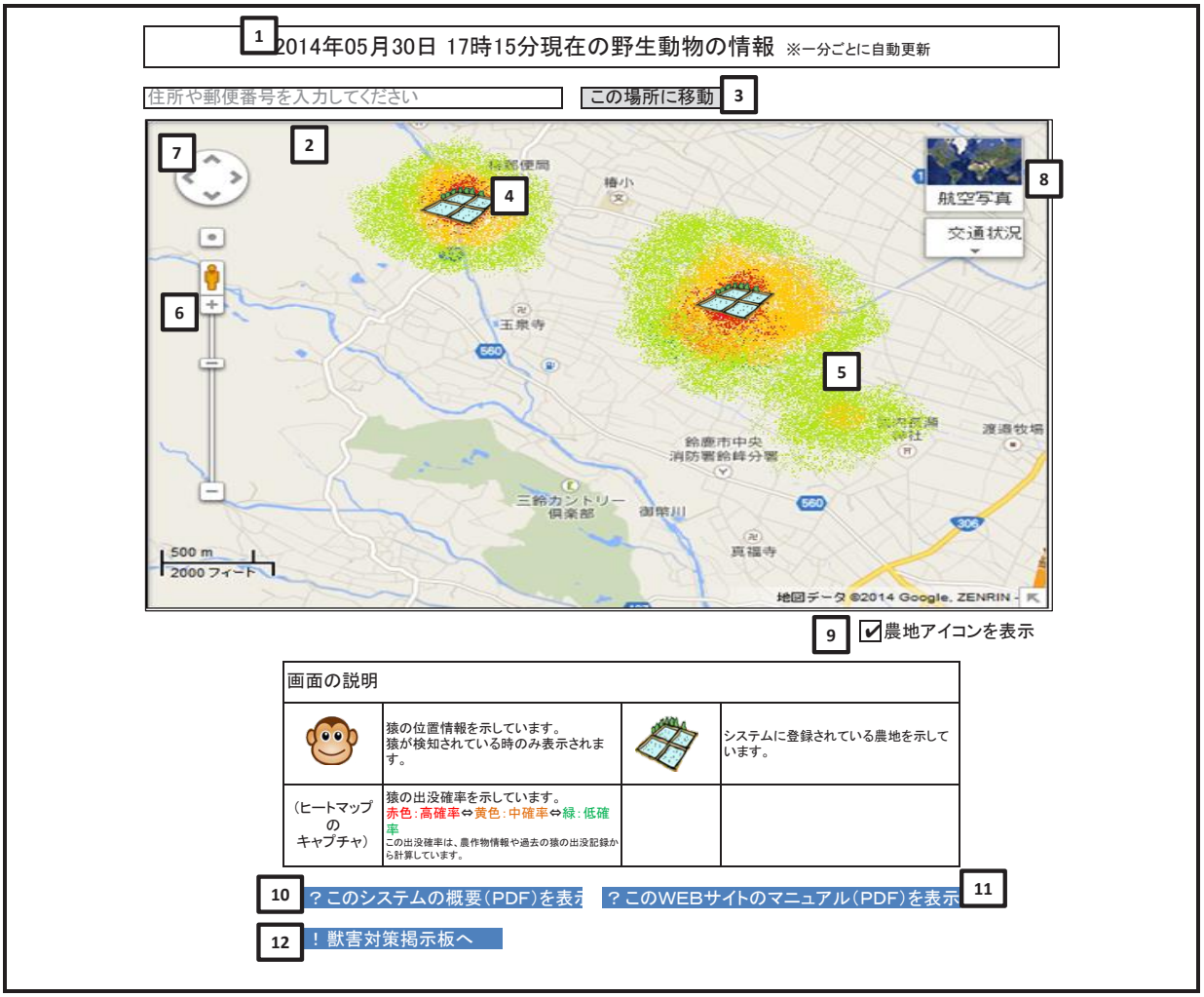
1. 農林水産省 - 鳥獣被害対策の現状と課題（PDF）
2. 三重県農業研究所 - 三重県における獣害の状況（PDF）
3. 東海農政局 - 農作物被害状況
4. 戸田 春華.”三重県亀山丘陵におけるニホンザルの行動特性と猿害”. 日本地理学会. 2007

以上

[i] 参考資料 1, 2

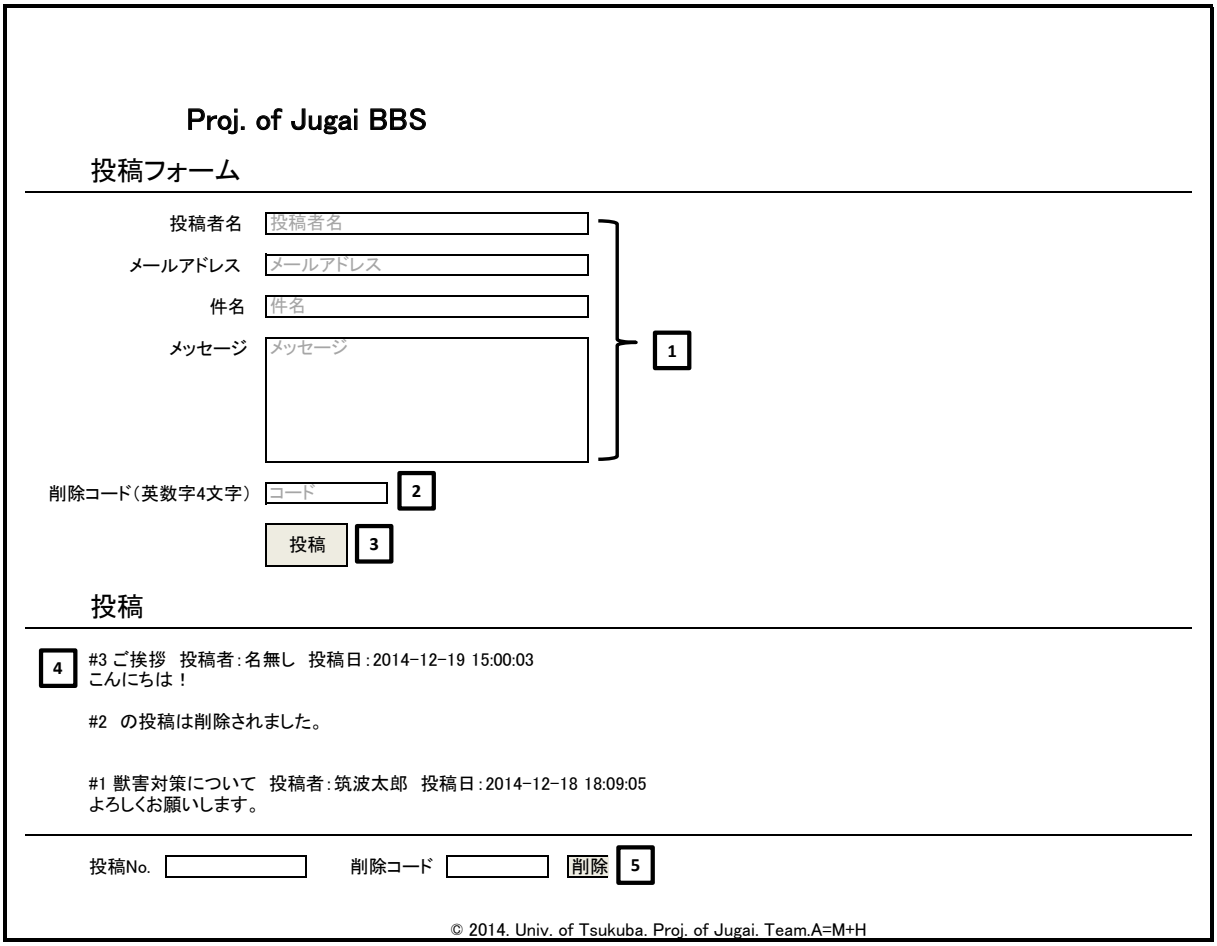
[ii] 5/2 会議時配布資料 - 獣害対策について（鈴鹿市農林水産課）

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
U0010	トップページ	/users/index	2014/5/30	2014/9/9	天本	光岡



1.日時表示
　マップの上に現在の日時を表示する。
2.マップ表示
　GoogleAPIを使用し、デフォルト（縮尺www、地点xxx,yyy）のマップを表示する。クエリストリングやセッションにデフォルト値が設定されている場合、その値に
3.マップの中心移動
　左のテキストボックスに入力されたキーワードを基にマップの描画中心点を移動する。
4.農地表示
　DBと接続して、geosテーブルに入っている農地の位置情報とステータスを取得し、MAP上にアイコンを表示する。
5.ヒートマップ表示
　サルの検出結果から次にサルが向かいやすい場所を色で表示する。
　赤＞黄＞緑で出現確率が高い
6.縮尺変更
　マップのデフォルトの縮尺を変更し、【マップ表示】する。その後、【観測箱表示】を行う。
7.位置移動
　マップのデフォルトの位置を変更し、【マップ表示】する。その後、【観測箱表示】を行う。
8.地図変更
　マップの地図を変更（図形<=>航空写真）し、【マップ表示】する。その後、【観測箱表示】を行う。
9.農地アイコンの表示切替
　農地アイコンを表示する/しないをトグルで切り替える。
10.システム概要の表示
　システム概要を説明するPDFファイルを開く。
11.マニュアルの表示
　このシステムの利用方法を説明するPDFファイルを開く。
12.獣害対策掲示板
　【U0020意見交換ページ】を新しいウインドウで開く。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
U0020	意見交換ページ	/bbs/index	2014/9/25	2014/9/28	森	光岡



1.投稿入力フォーム
投稿者名、メールアドレス、件名、メッセージを入力する。

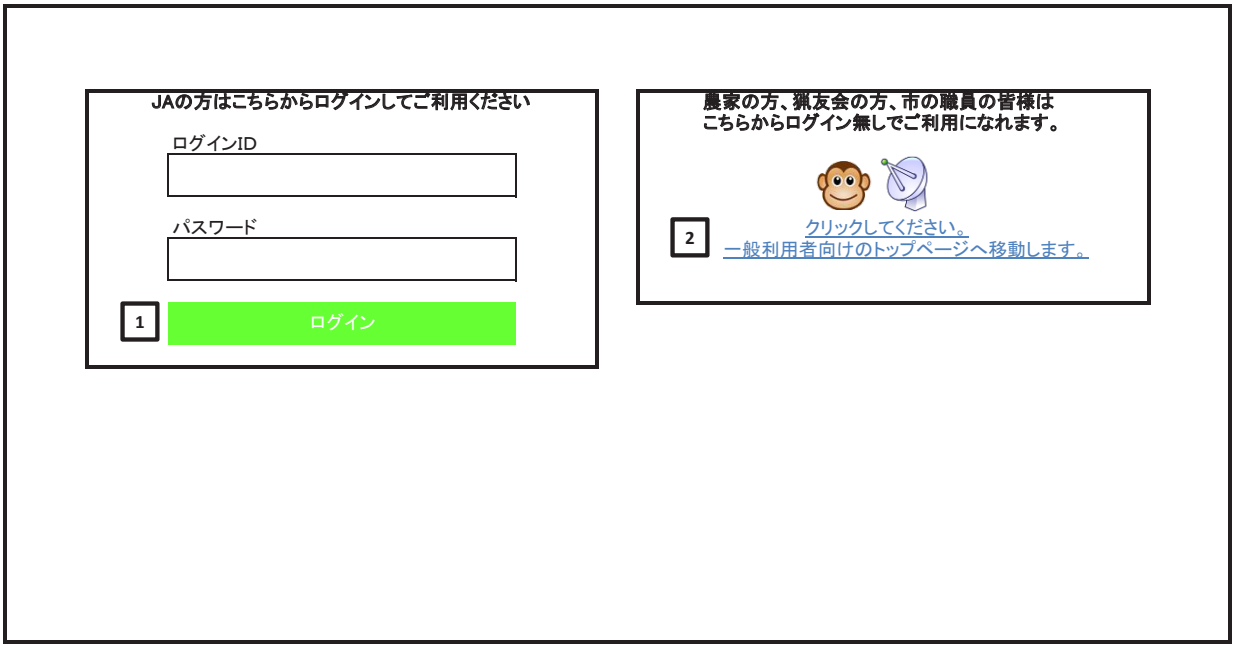
2.削除コード
後から投稿を削除する際に必要となる4ケタの英数字を入力する。

3.投稿ボタン
【1.投稿入力フォーム】にて入力された内容を投稿する。

4.投稿内容
今までに投稿されたメッセージが時系列降順に表示される。

5.投稿削除ボタン
左のフォームに削除したい投稿のNo.と投稿時に入力した削除コードを入力することで投稿されたメッセージを削除する。

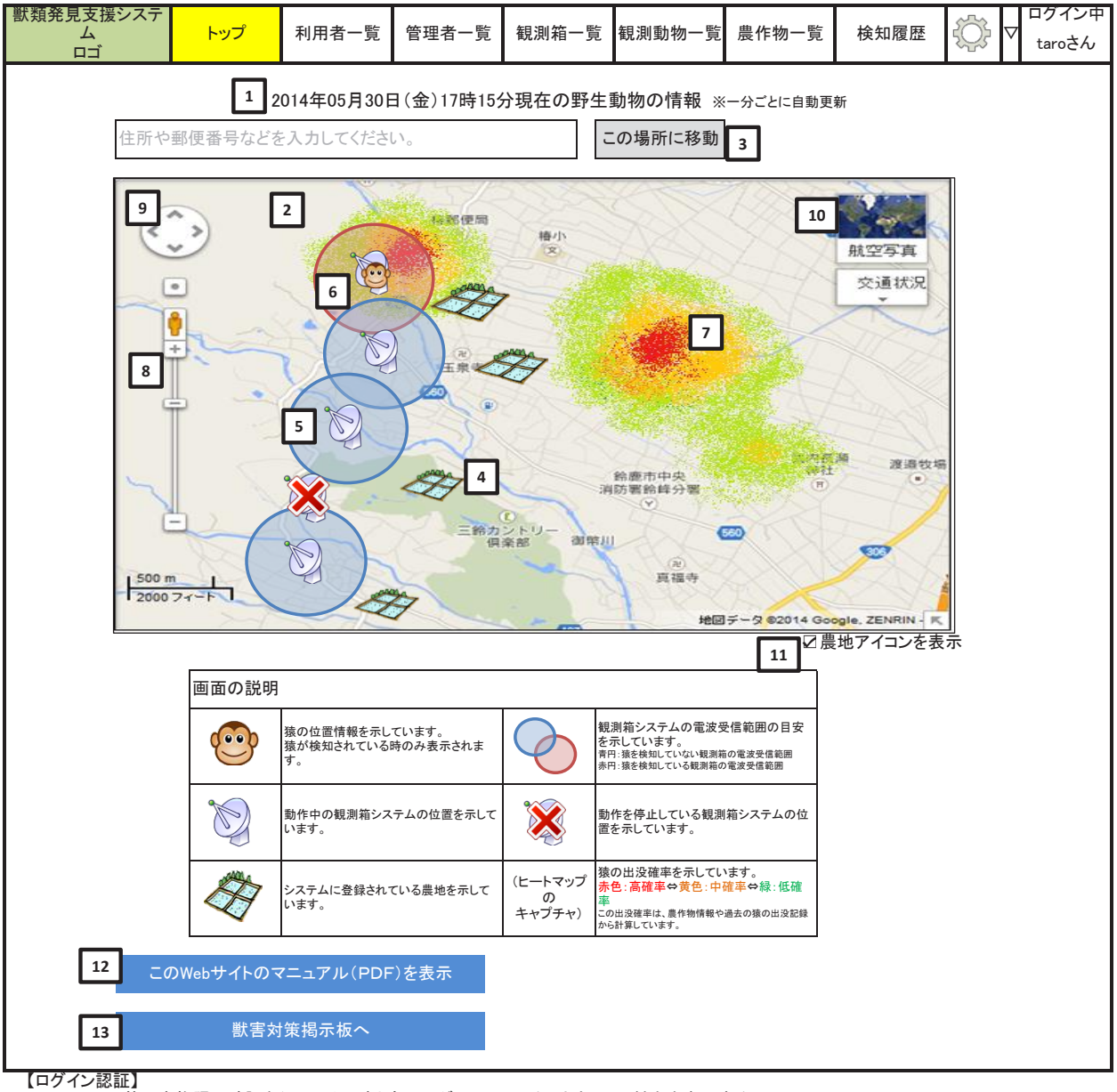
画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0010	管理者ログインページ	/admins/login	2014/5/30	2014/6/11	天本	光岡



1.管理者としてログイン
CakeのAuth機能より、認証を行う。認証成功の場合は、【A0020.管理者トップページ】へ、
失敗の場合は、IDとパスワードが一致しませんとメッセージを出し、このページへ遷移させる。

2.利用者ページへのリンク
【U0010.利用者ページ】へ遷移する。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0020	管理者トップページ	/admins/index	2014/5/30	2014/9/11	天本	光岡



【ログイン認証】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

1.日時表示
マップの上に現在の日時を表示する。

2.マップ表示
GoogleAPIを使用し、デフォルト（縮尺www、地点xxx,yyy）のマップを表示する。
クエリストリングやセッションにデフォルト値が設定されている場合、その値に従う。

3.マップの中心移動
左のテキストボックスに入力されたキーワードを基にマップの描画中心点を移動する。

4.農地表示
DBと接続して、geosテーブルに入っている農地の位置情報とステータスを取得し、MAP上にアイコンを表示する。

5.観測箱表示
DBと接続して、boxesテーブルに入っている観測箱の位置情報とステータスを取得し、MAP上にアイコンを表示する。
状態が【異常】のものは×印が表示され、サルを検知しているものは円の色が赤で表示される。

6.検知中のサルの表示
観測箱の検知状態から計算されたサルの位置を表示する。

7.ヒートマップ表示
サルの検出結果から次にサルが向かいやすい場所を色で表示する。
赤＞黄＞緑で出現確率が高い

8.縮尺変更
マップのデフォルトの縮尺を変更し、【マップ表示】する。その後、【観測箱表示】を行う。

9.位置移動
マップのデフォルトの位置を変更し、【マップ表示】する。その後、【観測箱表示】を行う。

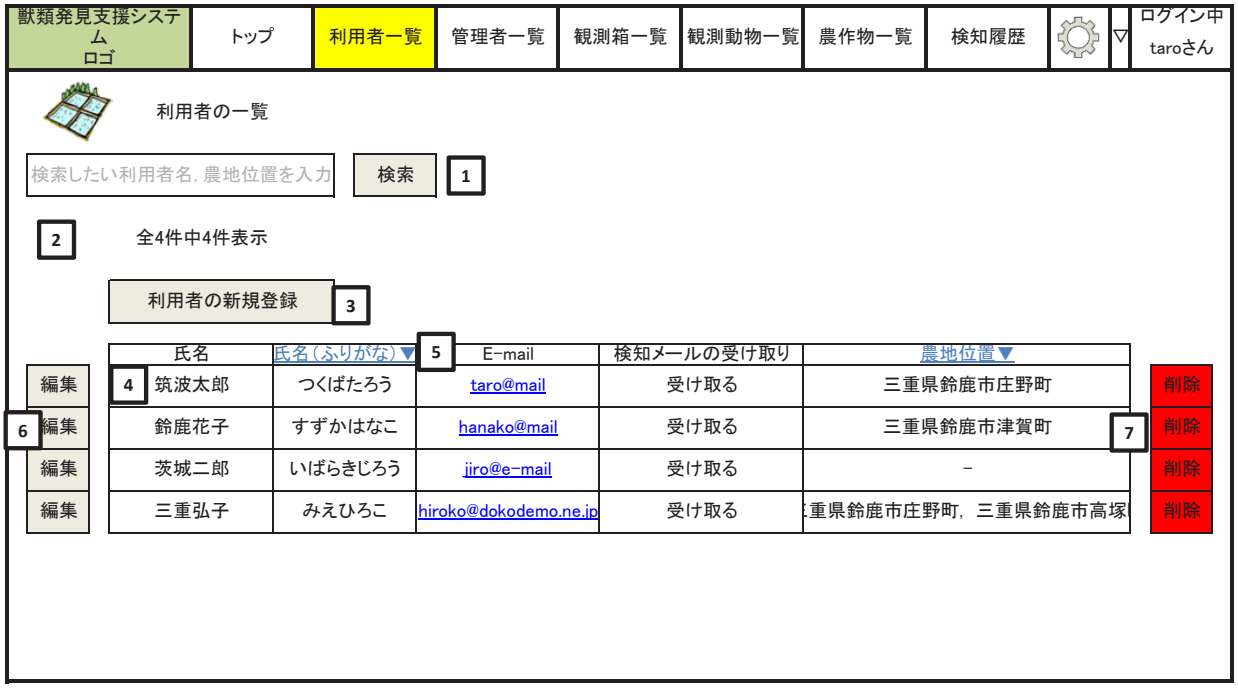
10.地図変更
マップの地図を変更（図形<=>航空写真）し、【マップ表示】する。その後、【観測箱表示】を行う。

11.農地アイコンの表示切替
農地アイコンを表示する/しないをトグルで切り替える。

12.マニュアルの表示
このシステムの利用方法を説明するPDFファイルを開く。

13.獣害対策掲示板
【U0020意見交換ページ】を新しいウインドウで開く。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0030	利用者一覧ページ	/admins/userlist	2014/5/30	2014/6/5	天本	光岡



削除確認画面

【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

1.検索機能
指定された条件でDBに検索をかけ、ページをリロードして、条件に合うデータのみを表示する。

2.表示件数の表示
検索の条件に適したものの件数を表示する。デフォルトでは全件表示する。

3.新規登録ボタン
【A0040利用者登録・編集ページ】に遷移する。

4.利用者表示
DBと接続して、usersテーブルに入っている利用者の情報を取得し、表示する。

5.ソート機能
・いま昇順である項目が押下された場合は、その項目を降順に切り替える。
いま表示している条件をセッションに登録して、逆順に表示する。
・昇順でも降順でもない項目が押下された場合、その項目を昇順に切り替える。
・デフォルトはIDで降順のソート。

6.編集ボタン
その行の利用者の情報を保持して【A0040.利用者登録・編集ページ】に遷移する。

7.削除ボタン
クリックされた際に、削除確認の画面（下図）を表示する。

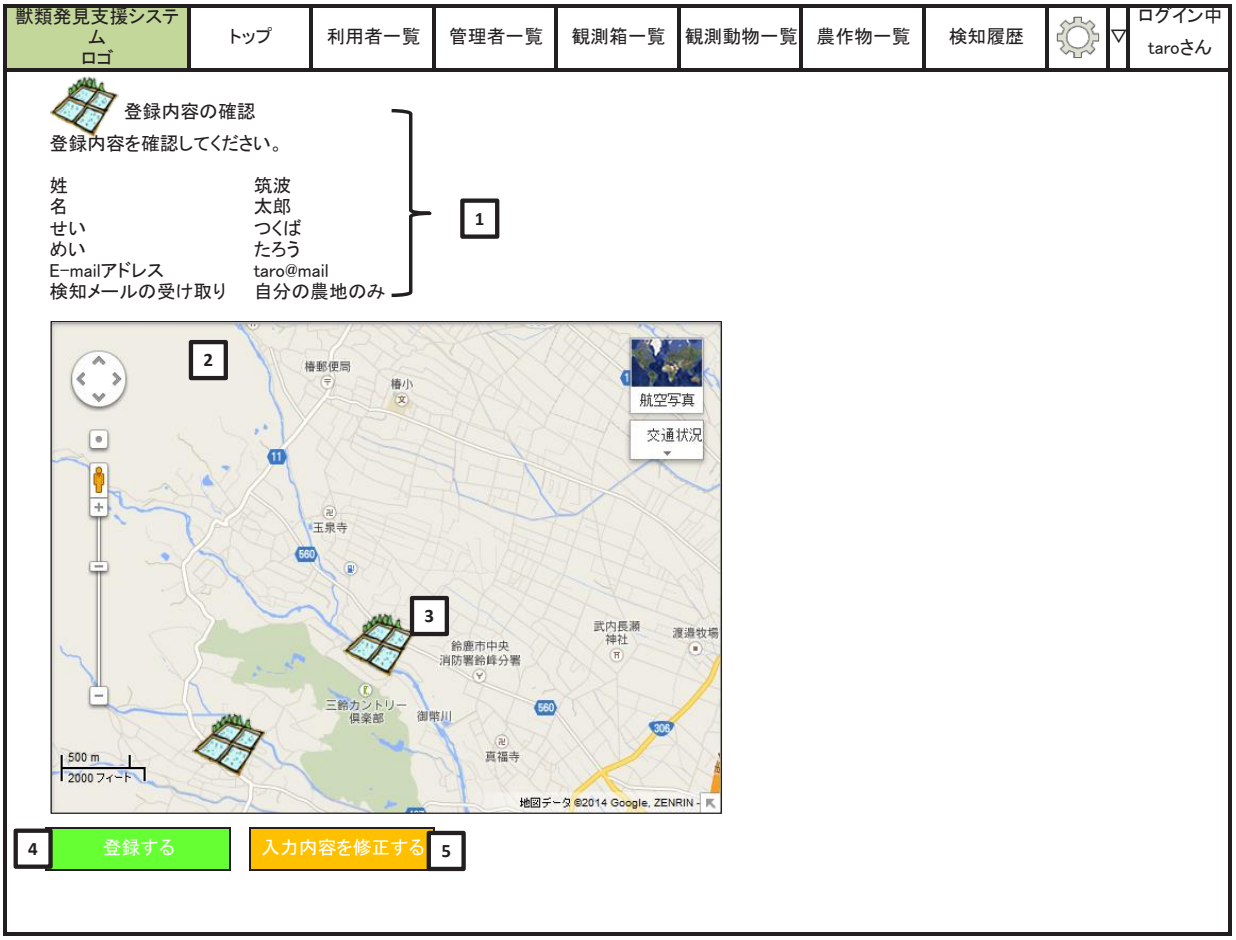
8.削除実行ボタン
該当のレコードをDBから削除する。

9.削除キャンセルボタン
元のページに戻る。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0040	利用者登録・編集ページ	/admins/userregister	2014/5/30	2014/10/25	天本	光岡

【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。
【データ取得】
Postされてきたユーザーの IDの情報をDBから取得し、フォームに表示する。
1.登録情報入力フォーム
登録する利用者の姓名、姓名のよみがな、E-mailアドレス、検知情報の受け取りの可否を入力する。
2.マップ表示
GoogleAPIを使用し、デフォルト(縮尺www、地点xxx,yyy)のマップを表示する。
クエリストリングやセッションにデフォルト値が設定されている場合、その値に従う。
3.マップの中心移動
左のテキストボックスに入力されたキーワードを基にマップの描画中心点を移動する。
4.農地表示
DBと接続して、geosテーブルに入っている該当ユーザーの農地の位置情報を取得し、MAP上にアイコンを表示する。
5.農地情報登録
地図をクリックされた場合、ピンをドロップし、農地を登録するかを確認する。
耕作している作物を選択する。
追加をクリックされた場合、セッションに位置情報を保存する。
中止をクリックされた場合、ピンを削除する。ピンをクリックされた場合、削除と移動、中止を確認し、
削除がクリックされた場合、ピンを削除し、セッションのデータも削除する。
移動がクリックされた場合、ピンが抜け移動できるようになる。クリックすると位置を確定する。
中止がクリックされた場合、ピンの編集を中止する。
6.登録確認ボタン
【A0050.利用者編集確認ページ】へ遷移する。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0050	利用者編集確認ページ	/admins/userconfirm	2014/6/1	2014/6/5	天本	光岡



【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

1.登録情報表示
登録ページからPOSTされてきたデータを表示する。

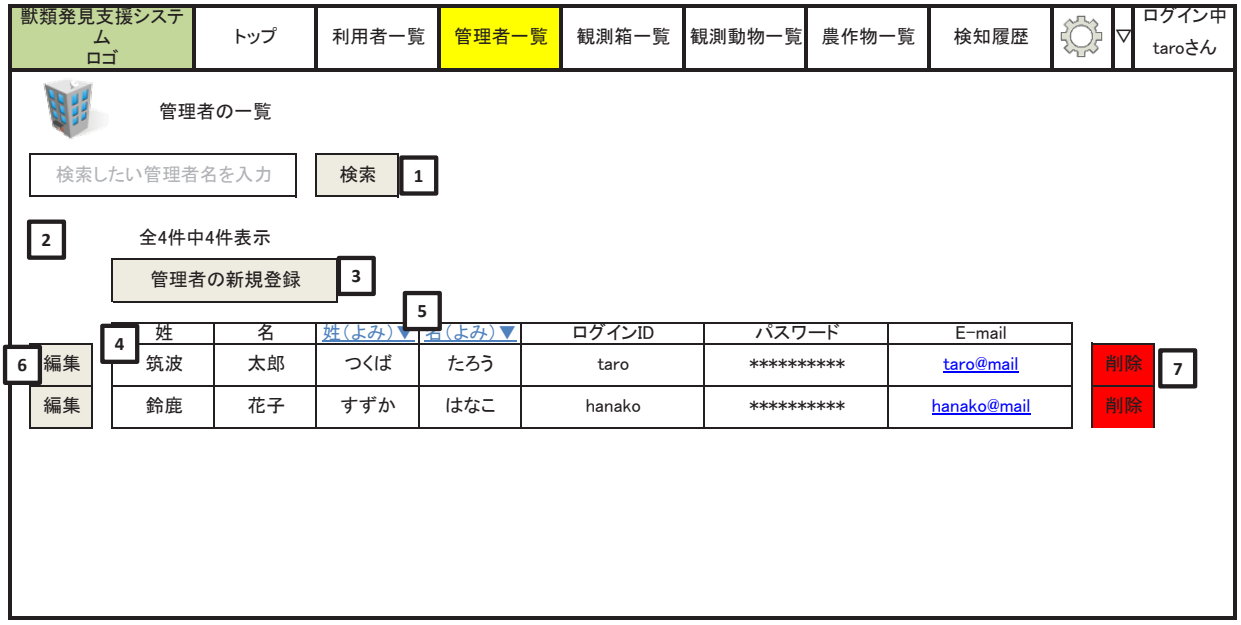
2.マップ表示
GoogleAPIを使用し、デフォルト（縮尺www、地点xxx,yyy）のマップを表示する。
クエリストリングやセッションにデフォルト値が設定されている場合、その値に従う。

3.登録する農地の表示
登録する全ての農地をマップ上に表示する。

4.利用者登録
登録がクリックされた場合、DBにデータを保存する。

5.再編集
編集し直すがクリックされた場合、データをセッションに保存し、利用者登録編集ページに戻る。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0060	管理者一覧ページ	/admins/adminlist	2014/5/30	2014/6/5	天本	光岡



削除確認画面

【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

1.検索機能
指定された条件でDBに検索をかけ、ページをリロードして、条件に合うデータのみを表示する。

2.表示件数の表示
検索の条件に適したものの件数を表示する。デフォルトでは全件表示する。

3.新規登録ボタン
【A0070管理者新規登録ページ】に遷移する。

4.管理者表示
DBと接続して、adminsテーブルに入っている管理者の情報を取得し、表示する。

5.ソート機能
・いま昇順である項目が押下された場合は、その項目を降順に切り替える。
いま表示している条件をセッションに登録して、逆順に表示する。
・昇順でも降順でもない項目が押下された場合、その項目を昇順に切り替える。
・デフォルトはIDで降順のソート。

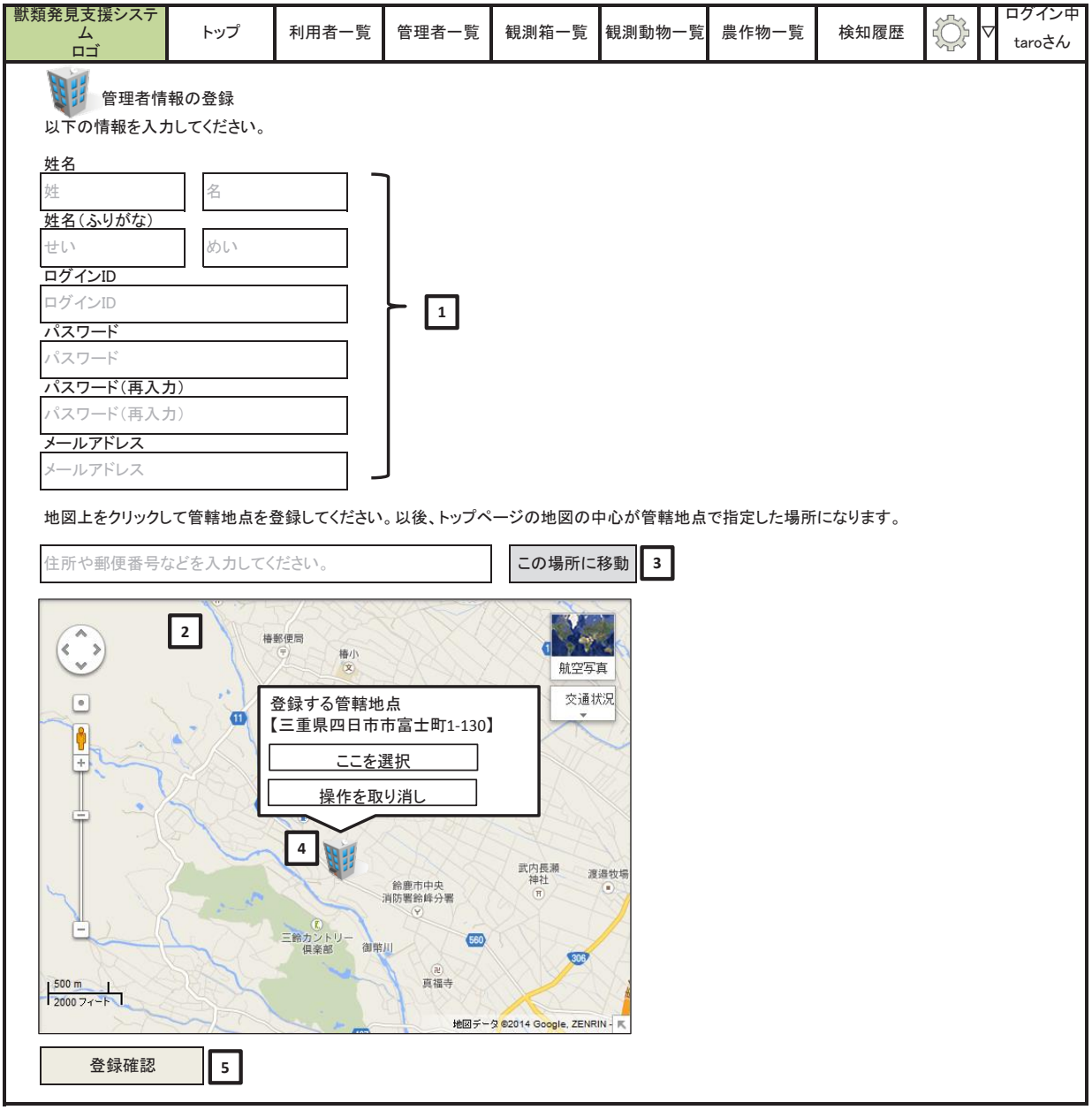
6..編集ボタン
その行の管理者の情報を保持して【A0080.管理者編集ページ】に遷移する。

7.削除ボタン
クリックされた際に、削除確認の画面(下図)を表示する。

8.削除実行ボタン
該当のレコードをDBから削除する。

9.削除キャンセルボタン
元のページに戻る。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0070	管理者新規登録ページ	/admins/adminregister	2014/5/30	2014/6/5	天本	光岡



【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

1.登録情報入力フォーム
登録する管理者の姓名、姓名のよみがな、ログインID、パスワード、パスワードの再入力、E-mailアドレスを入力する。

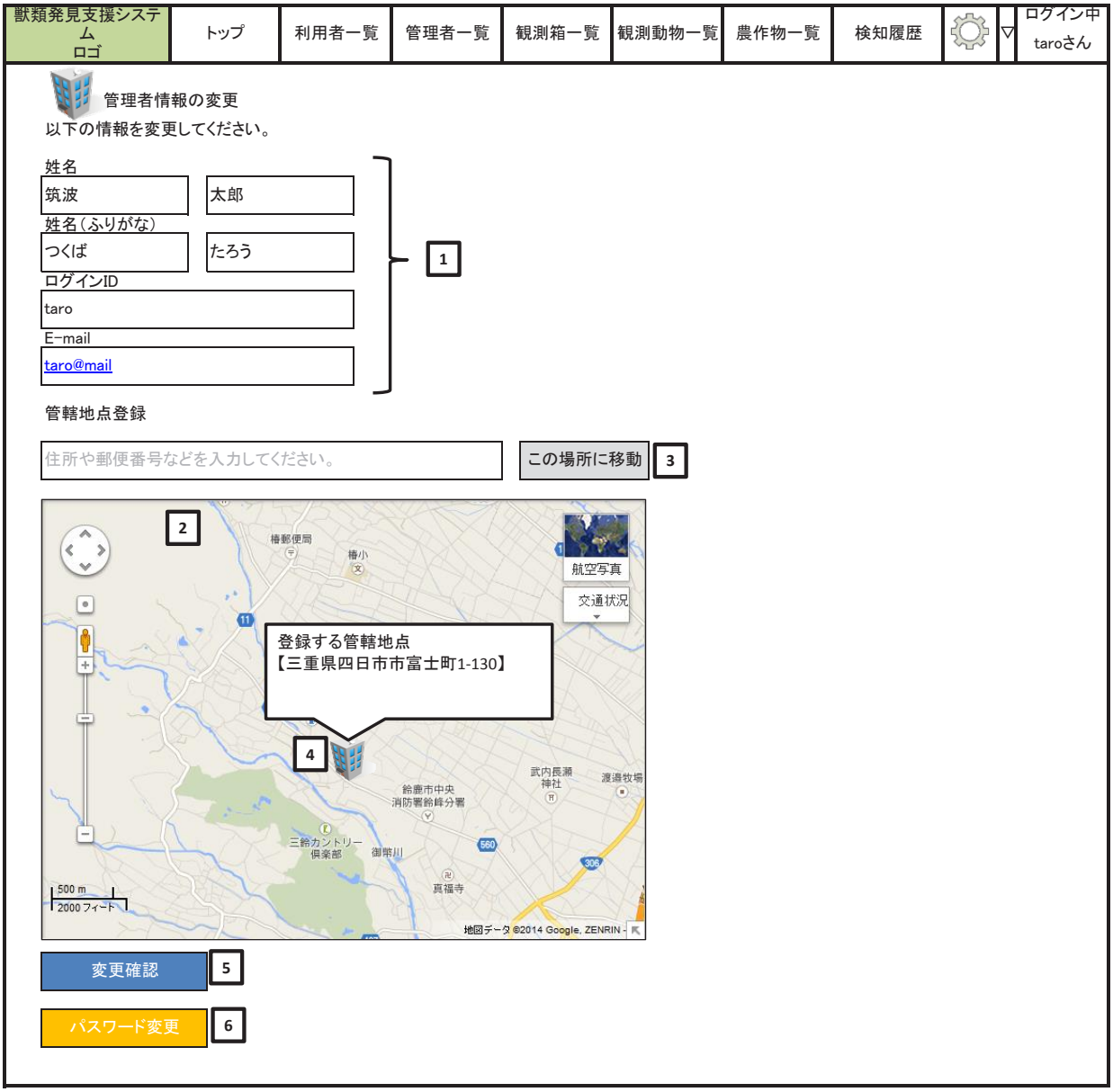
2.マップ表示
GoogleAPIを使用し、デフォルト（縮尺www、地点xxx,yyy）のマップを表示する。
クエリストリングやセッションにデフォルト値が設定されている場合、その値に従う。

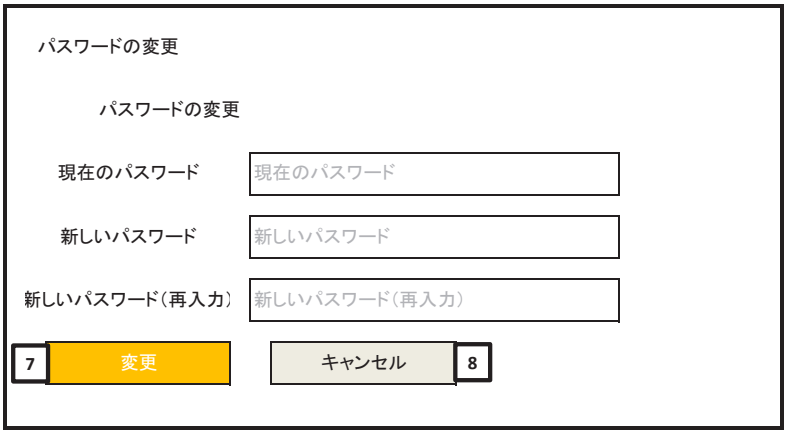
3.マップの中心移動
左のテキストボックスに入力されたキーワードを基にマップの描画中心点を移動する。

4.管轄地点登録
地図をクリックされた場合、ピンをドロップし、管轄地点を登録するかを確認する。
追加をクリックされた場合、セッションに位置情報を保存する。
中止をクリックされた場合、ピンを削除する。ピンをクリックされた場合、削除と移動、中止を確認し、
削除がクリックされた場合、ピンを削除し、セッションのデータも削除する。
移動がクリックされた場合、ピンが抜け移動できるようになる。クリックすると位置を確定する。
中止がクリックされた場合、ピンの編集を中止する。

5.登録確認ボタン
【A0090.管理者編集確認ページ】へ遷移する。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0080	管理者編集ページ	/admins/adminedit	2014/5/30	2014/9/5	天本	光岡





【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

【データ取得】
Postされてきた管理者のIDの情報をDBから取得し、フォームに表示する。

1.登録情報入力フォーム
編集する管理者の姓名、姓名のよみがな、ログインID、E-mailアドレスを入力する。

2.マップ表示
GoogleAPIを使用し、デフォルト（縮尺www、地点xxx,yyy）のマップを表示する。
クエリストリングやセッションにデフォルト値が設定されている場合、その値に従う。

3.マップの中心移動
左のテキストボックスに入力されたキーワードを基にマップの描画中心点を移動する。

4.管轄地点登録
POSTされてきたユーザの情報から登録されている管轄地点をMAPに表示する。
地図をクリックされた場合、ピンをドロップし、管轄地点を登録するかを確認する。
追加をクリックされた場合、セッションに位置情報を保存する。
中止をクリックされた場合、ピンを削除する。ピンをクリックされた場合、削除と移動、中止を確認し、
削除がクリックされた場合、ピンを削除し、セッションのデータも削除する。
移動がクリックされた場合、ピンが抜け移動できるようになる。クリックすると位置を確定する。
中止がクリックされた場合、ピンの編集を中止する。

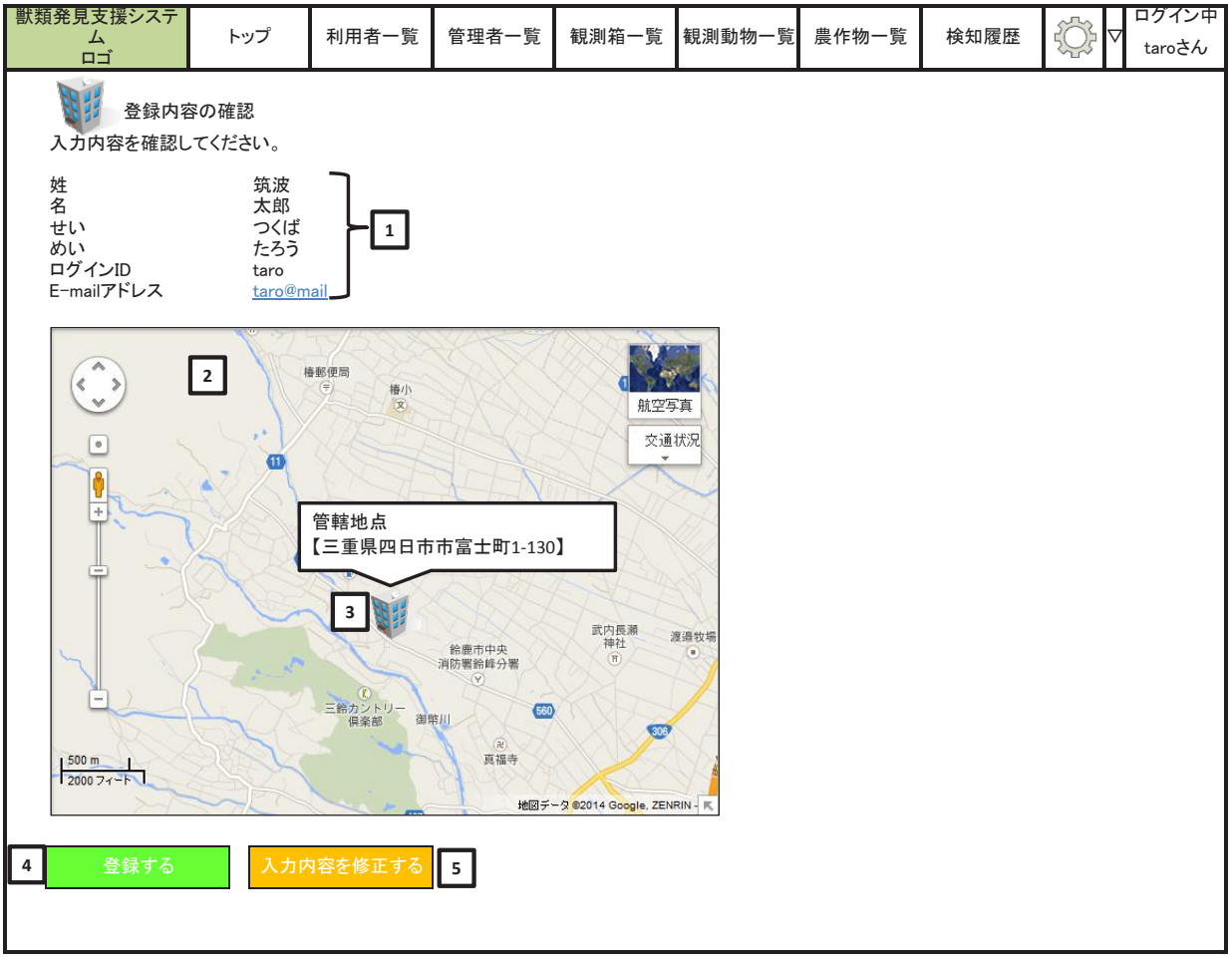
5.変更確認ボタン
【A0090.管理者編集確認ページ】へ遷移する。

6.パスワード変更モーダル表示
パスワード変更がクリックされた場合、パスワード変更のフォームをモーダルで表示する。

7.パスワード変更ボタン
フォームに入力された現在のパスワードの一致を確認し、新しいパスワードにDBを更新する。
パスワード変更完了のページを表示し、モーダルを閉じる。

8.変更キャンセルボタン
モーダルを閉じる。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0090	管理者編集確認ページ	/admins/adminconfirm	2014/5/30	2014/6/2	天本	光岡



【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

1.登録する管理者情報の表示
セッションに保存されている登録する管理者の情報を表示する。
・新規登録の場合は姓名、姓名（よみ）、ログインID、パスワード（*印で表示）、E-mailを表示する。
・編集の場合は姓名、姓名（よみ）、ログインID、E-mailを表示する。

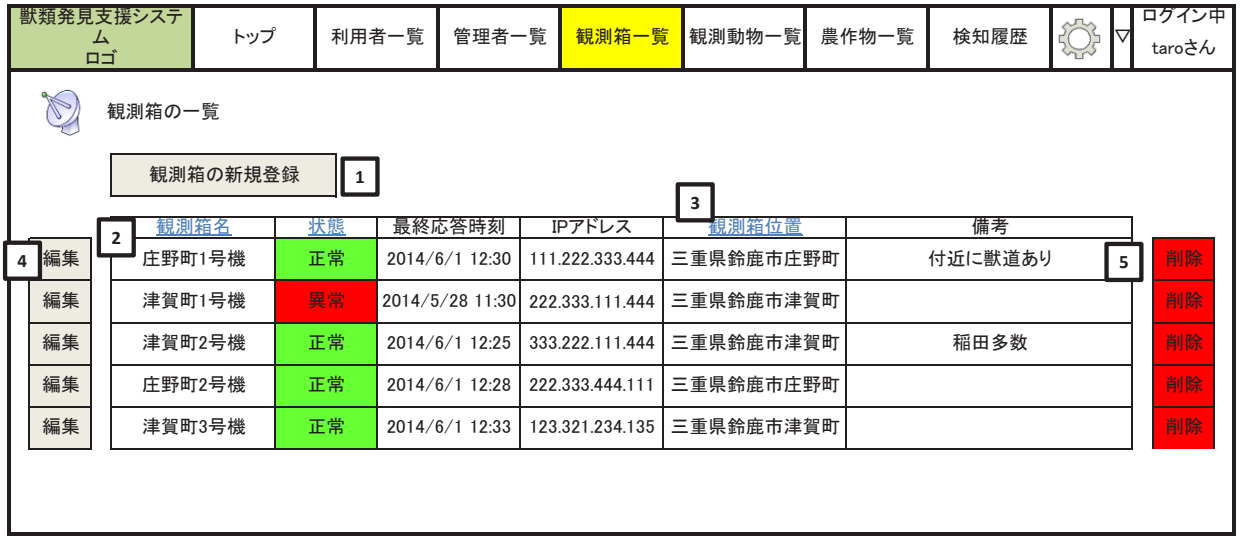
2.マップ表示
GoogleAPIを使用し、デフォルト（縮尺www、地点xxx,yyy）のマップを表示する。
クエリストリングやセッションにデフォルト値が設定されている場合、その値に従う。

3.登録する管轄地点の表示
登録する管轄地点をマップ上に表示する。

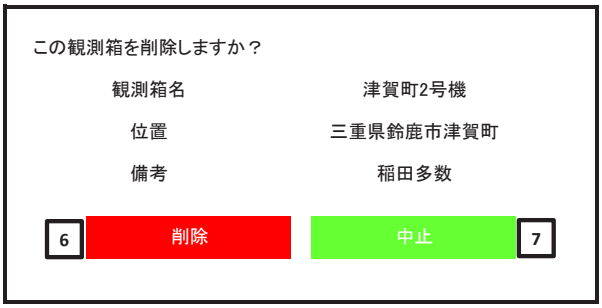
4.登録ボタン
・DBにセッションに保存されている姓、名、姓（よみ）、名（よみ）、ログインID、パスワード（暗号化済み）、E-mailを保存する。
・登録完了をユーザに通知し、【A0060.管理者一覧ページ】に遷移する。

5.編集に戻るボタン押下
【A0080.管理者編集ページ】に遷移する。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0100	観測箱一覧ページ	/admin/boxlist	2014/5/30	2014/6/5	天本	光岡



削除確認画面



【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

1.新規登録ボタン
【A0110観測箱編集ページ】に遷移する。

2.観測箱表示
DBと接続して、boxesテーブルに入っている観測箱の情報を取得し、表示する。

3.ソート機能
・いま昇順である項目が押下された場合は、その項目を降順に切り替える。
いま表示している条件をセッションに登録して、逆順に表示する。
・昇順でも降順でもない項目が押下された場合、その項目を昇順に切り替える。
・デフォルトはIDで降順のソート。

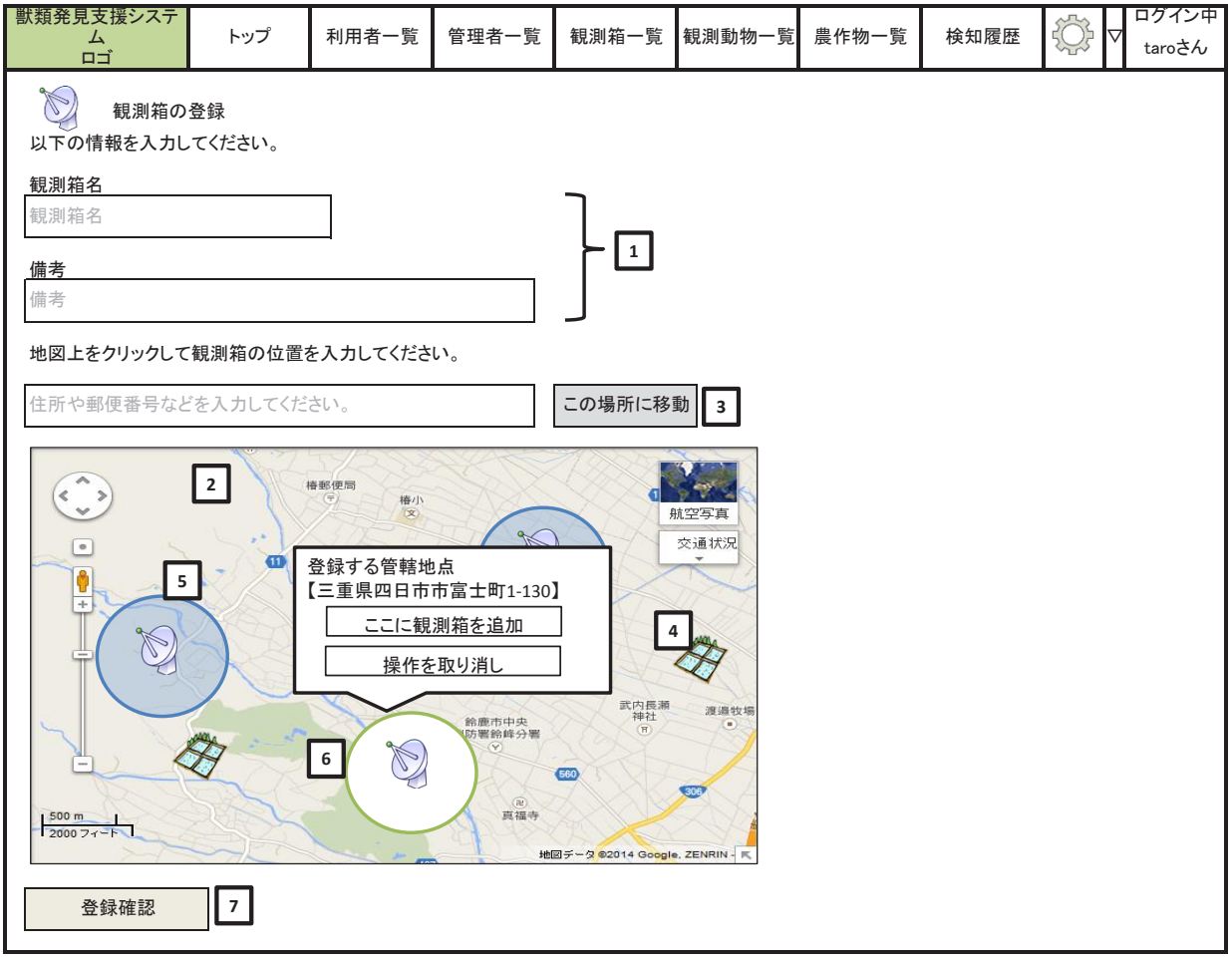
4..編集ボタン
その行の観測箱の情報を保持して【A0110.観測箱編集ページ】に遷移する。

5.削除ボタン
クリックされた際に、削除確認の画面（下図）を表示する。

6.削除実行ボタン
該当のレコードをDBから削除する。

7.削除キャンセルボタン
元のページに戻る。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0110	観測箱編集ページ	/admins/boxregister	2014/5/30	2014/6/5	天本	光岡



【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

【データ取得】
Postされてきた観測箱のIDの情報をDBから取得し、フォームに表示する。

1.登録情報入力フォーム
登録・編集する観測箱の名前、備考を入力する。

2.マップ表示
GoogleAPIを使用し、デフォルト（縮尺www、地点xxx,yyy）のマップを表示する。
クエリストリングやセッションにデフォルト値が設定されている場合、その値に従う。

3.マップの中心移動
左のテキストボックスに入力されたキーワードを基にマップの描画中心点を移動する。

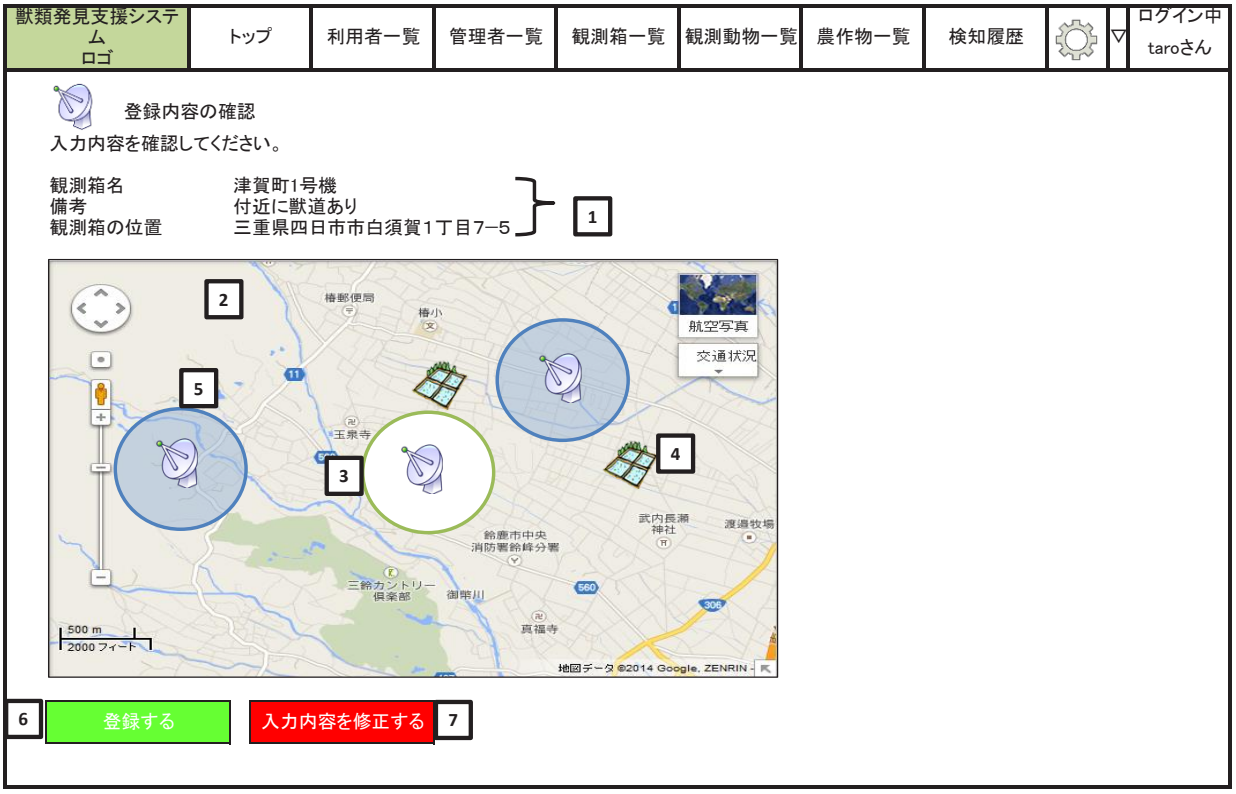
4.農地表示
DBと接続して、geosテーブルに入っている農地の位置情報とステータスを取得し、MAP上にアイコンを表示する。

5.観測箱表示
DBと接続して、boxesテーブルに入っている観測箱の位置情報とステータスを取得し、MAP上にアイコンを表示する。
状態が【異常】のものは×印が表示され、サルを検知しているものは円の色が赤で表示される。

6.観測箱の位置登録
POSTされてきた観測箱の情報から登録されている位置にアイコンを表示する。
地図をクリックされた場合、ピンをドロップし、観測箱を登録するかを確認する。
追加をクリックされた場合、セッションに位置情報を保存する。
中止をクリックされた場合、ピンを削除する。ピンをクリックされた場合、削除と移動、中止を確認し、
削除がクリックされた場合、ピンを削除し、セッションのデータも削除する。
移動がクリックされた場合、ピンが抜け移動できるようになる。クリックすると位置を確定する。
中止がクリックされた場合、ピンの編集を中止する。

7.登録確認ボタン
【A0120.観測箱編集確認ページ】へ遷移する。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0120	観測箱編集確認ページ	/admins/boxconfirm	2014/6/2	2014/6/5	天本	光岡



【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

1.登録する観測箱情報の表示
セッションに保存されている登録する管理者の情報を表示する。

2.マップ表示
GoogleAPIを使用し、デフォルト（縮尺www、地点xxx,yyy）のマップを表示する。
クエリストリングやセッションにデフォルト値が設定されている場合、その値に従う。

3.登録する管轄地点の表示
登録する管轄地点をマップ上に表示する。円の色は緑とする。

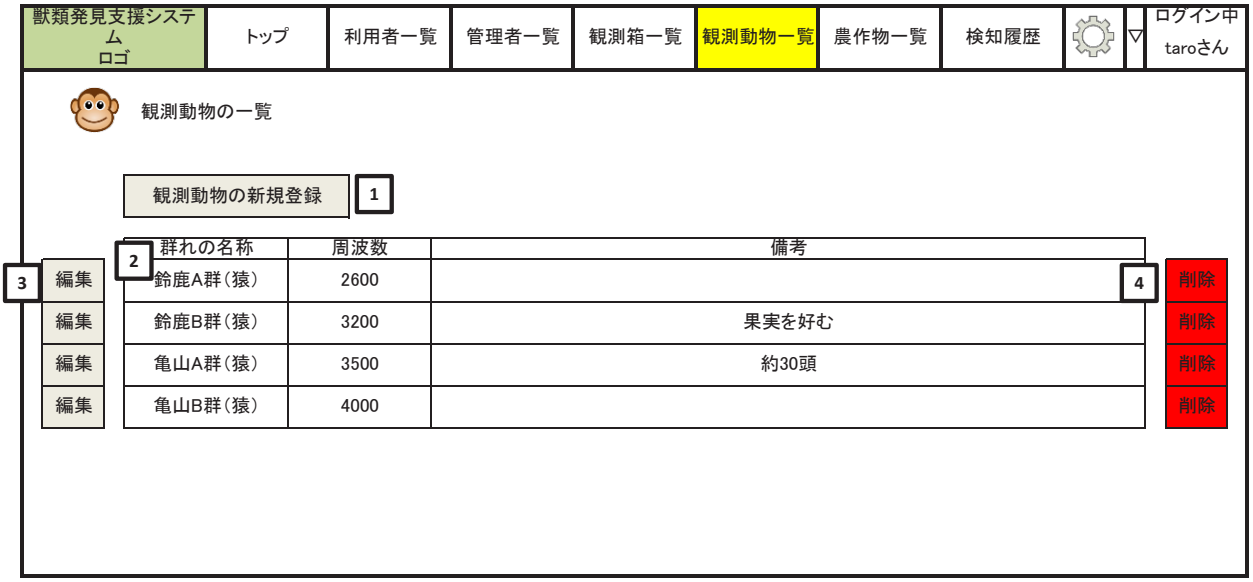
4.農地表示
DBと接続して、geosテーブルに入っている農地の位置情報とステータスを取得し、MAP上にアイコンを表示する。

5.観測箱表示
DBと接続して、boxesテーブルに入っている観測箱の位置情報とステータスを取得し、MAP上にアイコンを表示する。
状態が【異常】のものは×印が表示され、サルを検知しているものは円の色が赤で表示される。

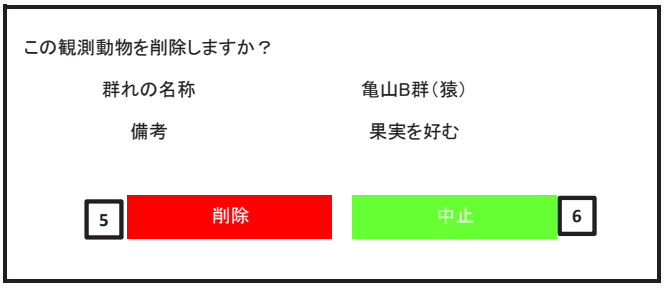
6.登録ボタン
・DBにセッションに保存されている観測箱名、備考、観測箱の位置（緯度・経度）を保存する。
・登録完了をユーザに通知し、【A0100.観測箱一覧ページ】に遷移する。

7.編集に戻るボタン押下
【A0110.観測箱編集ページ】に遷移する。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0130	観測動物一覧ページ	/admins/animallist	2014/9/12	2014/9/12	天本	光岡



削除確認画面



【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

1.新規登録ボタン
【A0140観測動物登録ページ】に遷移する。

2.観測動物表示
DBと接続して、animalsテーブルに入っている観測動物の情報を取得し、表示する。

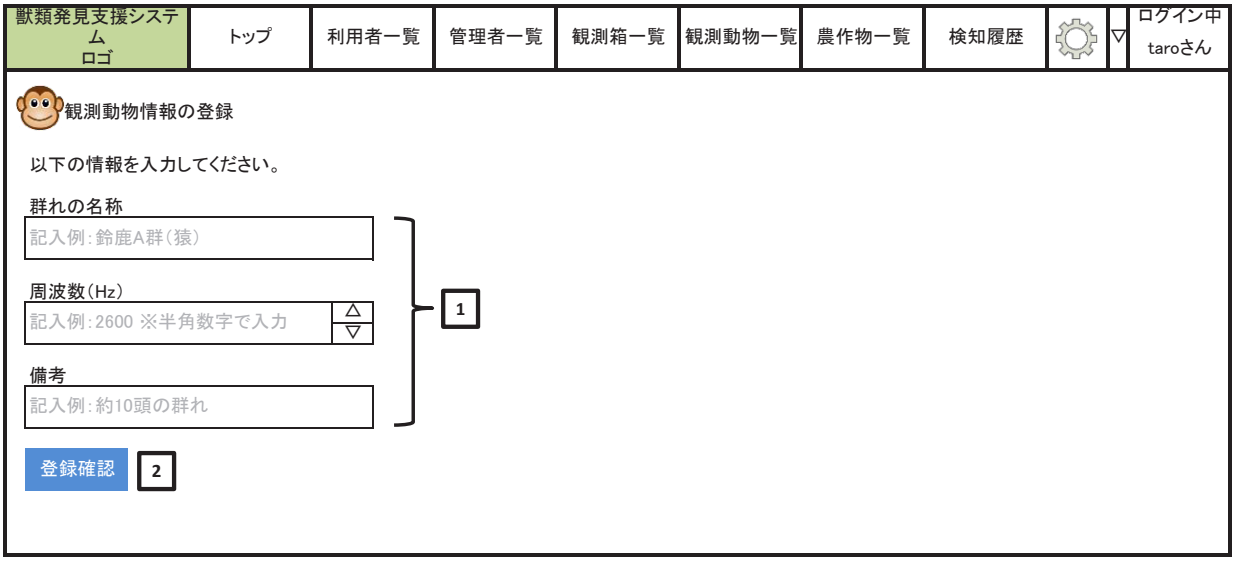
3..編集ボタン
その行の観測動物の情報を保持して【A0140.観測動物登録ページ】に遷移する。

4.削除ボタン
クリックされた際に、削除確認の画面（下図）を表示する。

5.削除実行ボタン
該当のレコードをDBから削除する。

6.削除キャンセルボタン
元のページに戻る。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0140	観測動物登録ページ	/admins/animalregister	2014/9/12	2014/9/12	天本	光岡



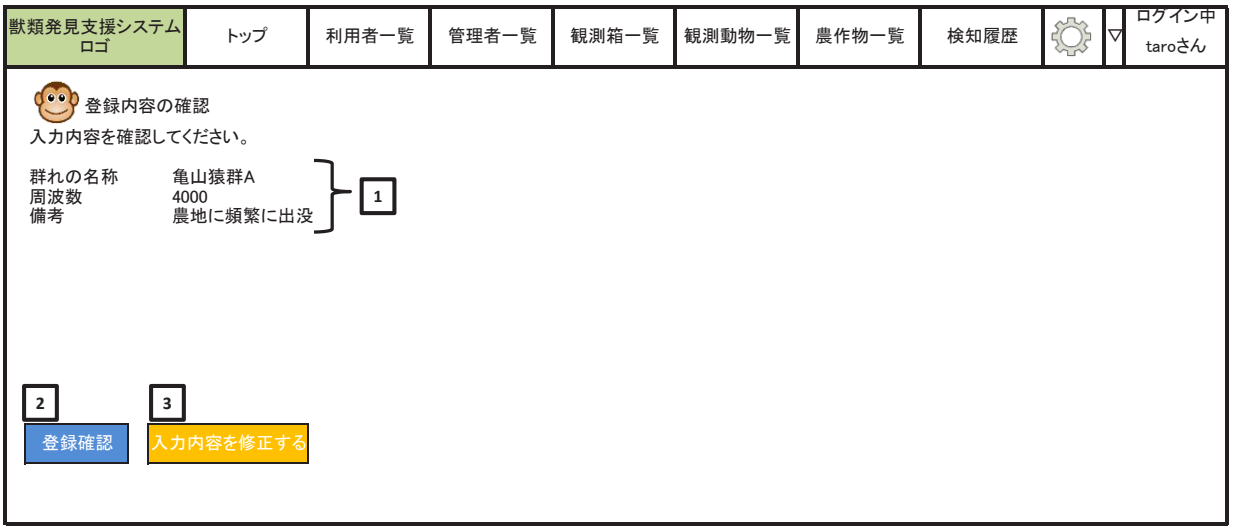
【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

【データ取得】
Postされてきた観測動物のIDの情報をDBから取得し、フォームに表示する。

1.登録情報入力フォーム
登録・編集する観測動物の群れの名称、周波数、備考を入力する。

2.登録確認ボタン
【A0150.観測動物登録確認ページ】へ遷移する。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0150	観測動物登録確認ページ	/admins/animalconfirm	2014/9/12	2014/9/12	天本	光岡



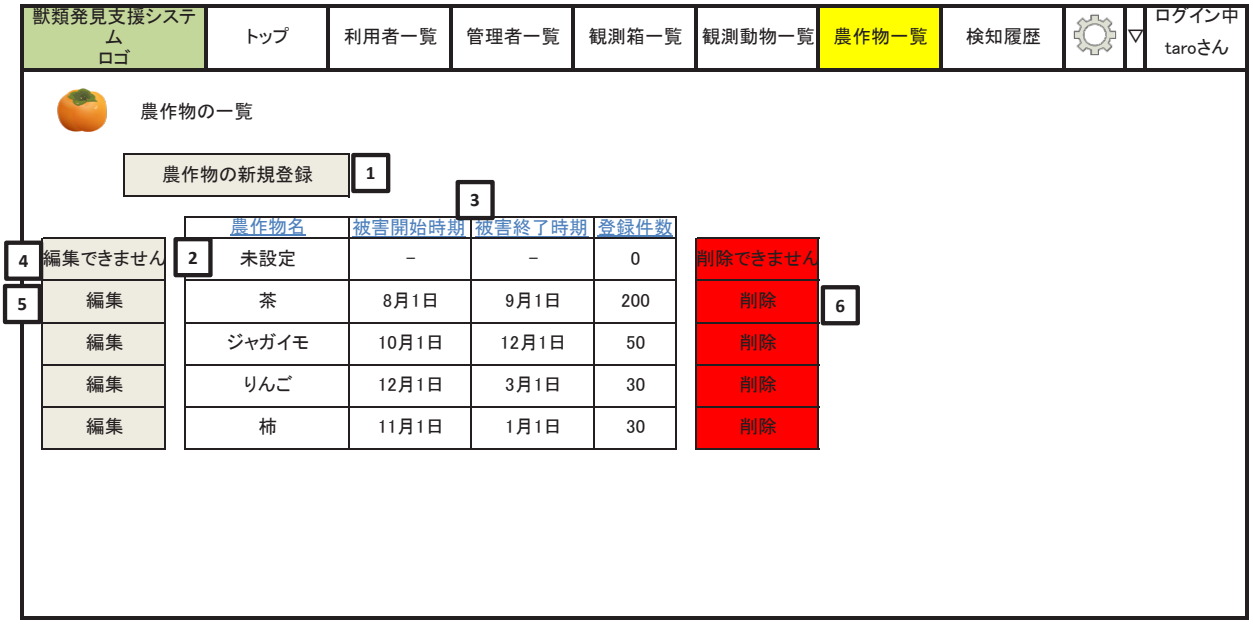
【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

1.登録する観測動物情報の表示
セッションに保存されている登録する観測動物の情報を表示する。

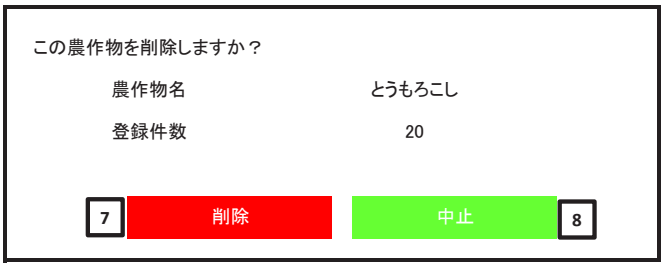
2.登録ボタン
・DBにセッションに保存されている観測動物の群れの名称、周波数、備考を保存する。
・登録完了をユーザに通知し、【A0130.観測動物一覧ページ】に遷移する。

3.編集に戻るボタン押下
【A0140.観測動物登録ページ】に遷移する。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0160	観測箱一覧ページ	/admin/croplist	2014/9/9	2014/9/9	天本	光岡



削除確認画面



【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

1.新規登録ボタン
【A0170農作物編集ページ】に遷移する。

2.農作物表示
DBと接続して、cropsテーブルに入っている農作物の情報を取得し、表示する。

3.ソート機能
・いま昇順である項目が押下された場合は、その項目を降順に切り替える。
いま表示している条件をセッションに登録して、逆順に表示する。
・昇順でも降順でもない項目が押下された場合、その項目を昇順に切り替える。
・デフォルトはIDで降順のソート。

4.未設定農作物の編集削除
未設定農作物の編集削除はできないものとする。

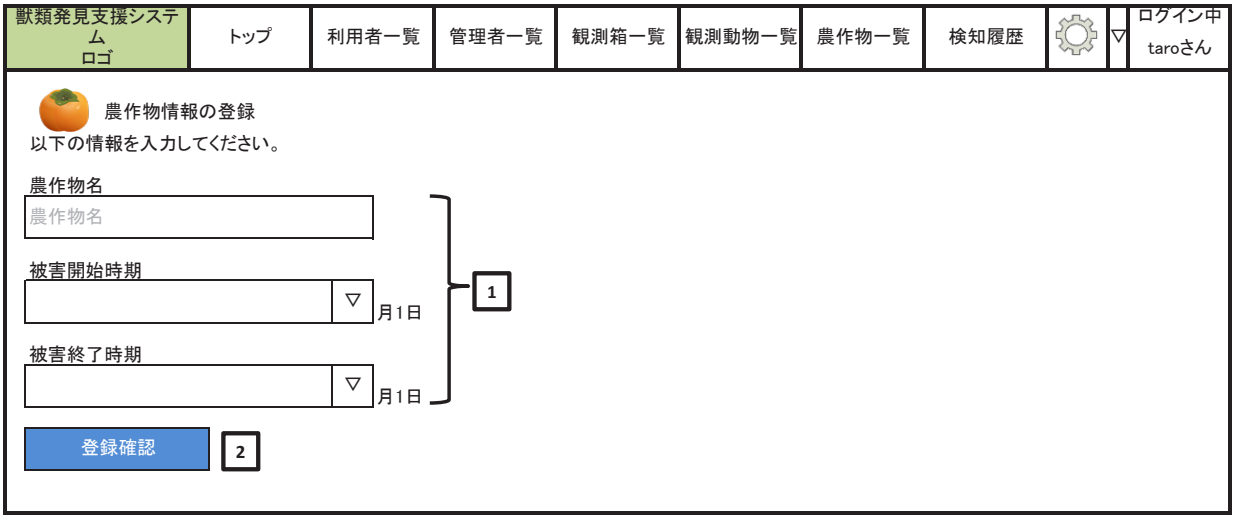
5.編集ボタン
その行の農作物の情報を保持して【A0170.農作物編集ページ】に遷移する。

6.削除ボタン
クリックされた際に、削除確認の画面（下図）を表示する。

7.削除実行ボタン
該当のレコードをDBから削除する。

8.削除キャンセルボタン
元のページに戻る。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0170	農作物編集ページ	/admins/cropregister	2014/9/9	2014/9/9	天本	光岡



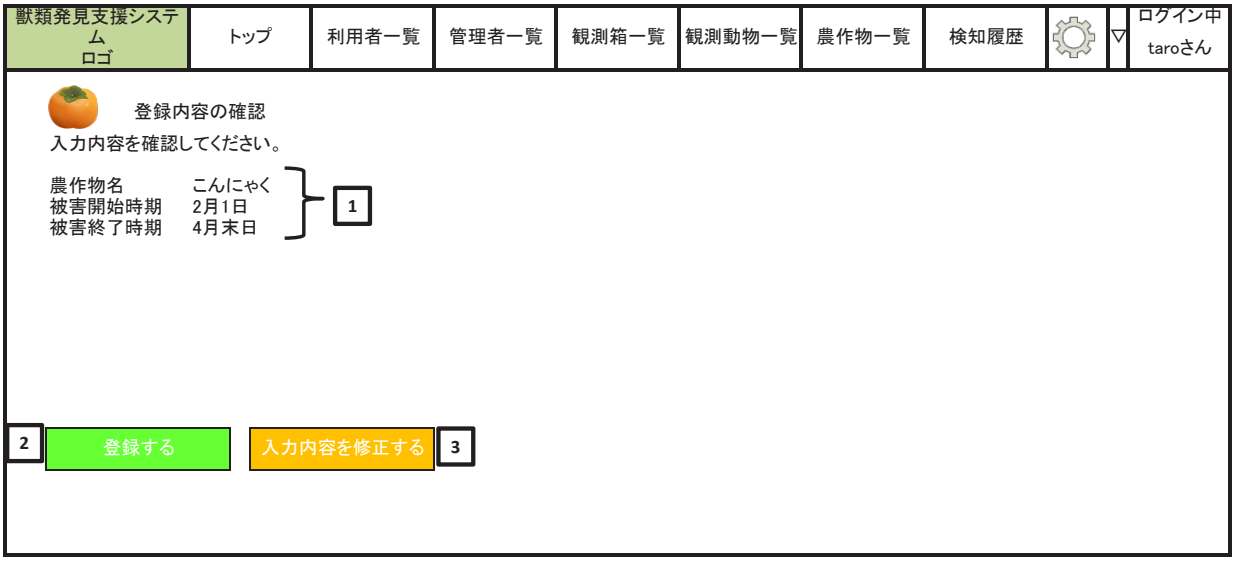
【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

【データ取得】
Postされてきた農作物のIDの情報をDBから取得し、フォームに表示する。

1.登録情報入力フォーム
登録・編集する農作物の名称、被害開始時期、被害終了時期を入力する。
なお、被害開始時期・被害終了時期はドロップリストから当該月を選択する。

2.登録確認ボタン
【A0180.農作物登録確認ページ】へ遷移する。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0180	農作物編集確認ページ	/admins/cropconfirm	2014/9/9	2014/9/9	天本	光岡



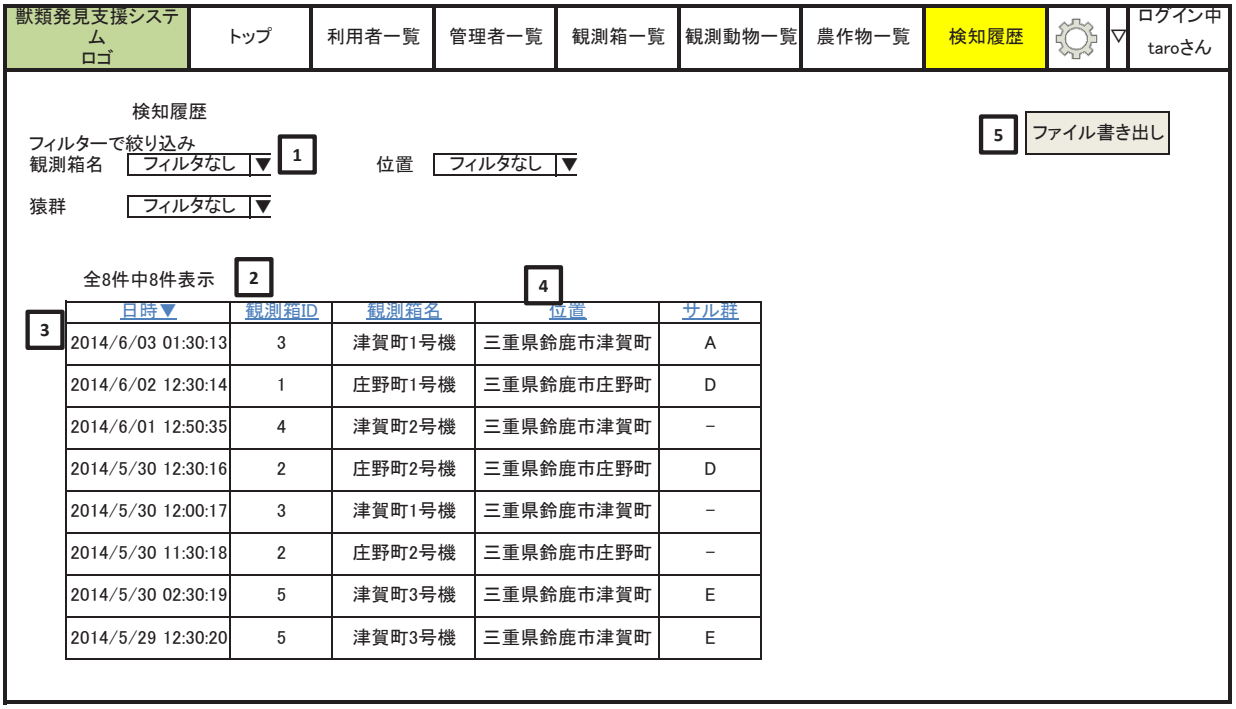
【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

1.登録する観測動物情報の表示
セッションに保存されている登録する農作物の情報を表示する。

2.登録ボタン
・DBにセッションに保存されている農作物の名称、被害開始時期、被害終了時期を保存する。
・登録完了をユーザに通知し、【A0160.農作物一覧ページ】に遷移する。

3.編集に戻るボタン押下
【A0170.農作物編集ページ】に遷移する。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
A0190	検知履歴ページ	/admins/loglist	2014/5/30	2014/6/5	光岡	光岡



【ログイン確認】
セッションで管理者権限の確認を行い、ページ上部にログインしているアカウントの姓名を表示する。

1.フィルタリング機能
観測箱名、位置、サル群でDBに検索をかけ、ページをリロードして、条件に合うデータのみを表示する。

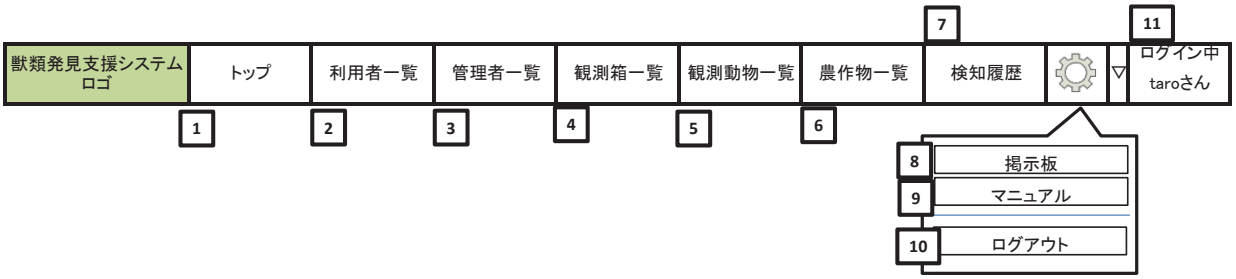
2.表示件数の表示
検索の条件に適したものの件数を表示する。デフォルトでは全件表示する。

3.ログ表示
DBと接続して、logsテーブルに入っている全てのレコードを取得し、表示する。
レコードの初期表示順は日時(ID)の降順である。

4.ソート
・今昇順である項目が押下された場合は、その項目を降順に切り替える。
今表示している条件をセッションに登録して、逆順に表示する。
・昇順でも降順でもない項目が押下された場合、その項目を昇順に切り替える。
日時以外でソートした場合は、第二の検索条件がIDの降順になる。

5.ファイル書き出し
今表示されているレコードをcsvファイルに書き出す。

画面ID	画面名	アクション	作成日	最終更新日	作成者	最終更新者
-	メニューバー	-	2014/5/30	2014/6/5	天本	光岡



1.トップページ
【A0020.管理者トップページ】へ遷移する。

2.利用者一覧
【A0030.利用者一覧ページ】へ遷移する。

3.管理者一覧
【A0060.管理者一覧ページ】へ遷移する。

4.観測箱一覧
【A0100.観測箱一覧ページ】へ遷移する。

5.観測動物一覧
【A0130.観測動物一覧ページ】へ遷移する。

6.農作物一覧
【A0160.農作物一覧ページ】へ遷移する。

7.検知履歴
【A0190.検知履歴ページ】へ遷移する。

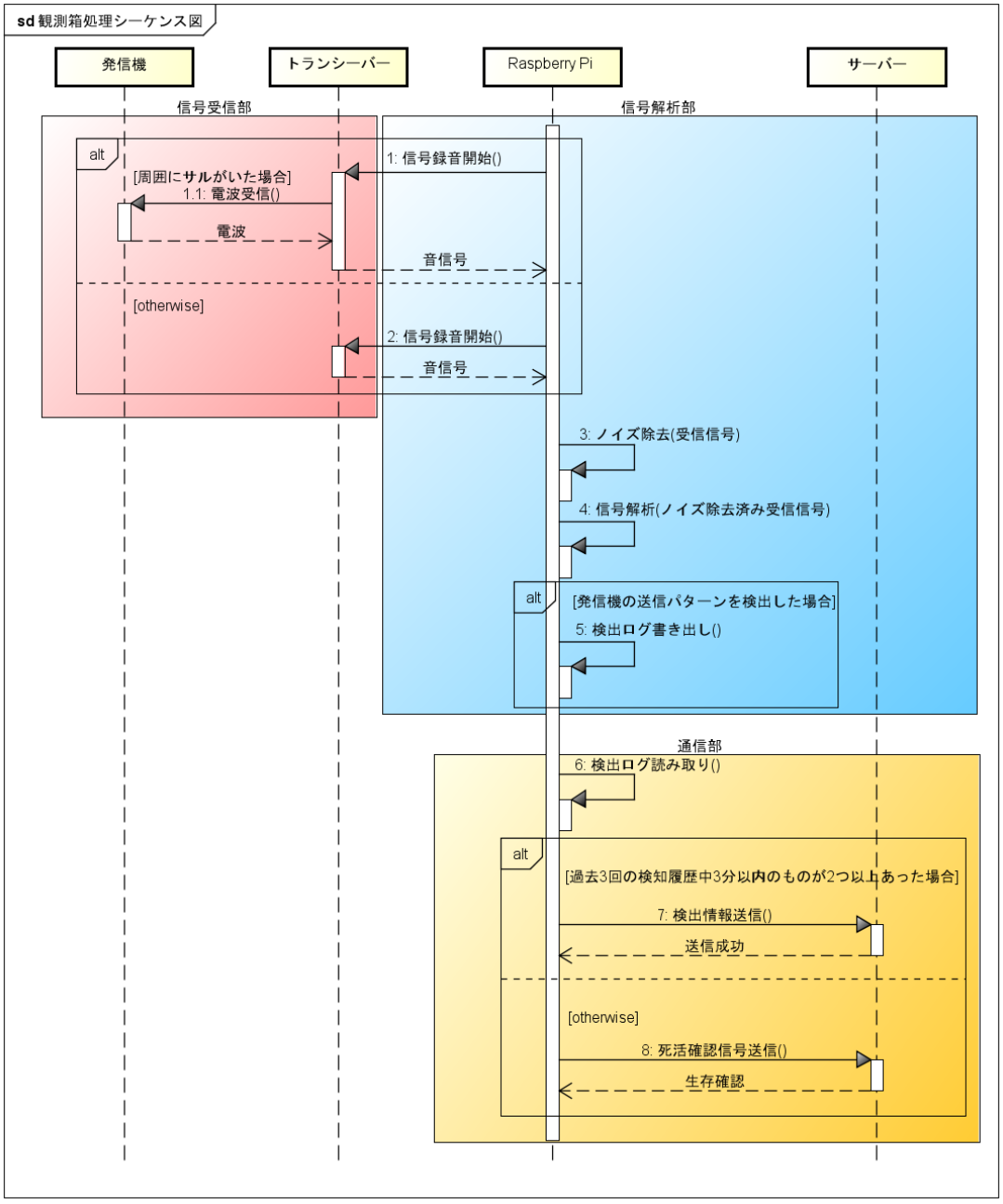
8.掲示板
【U0020.意見交換ページ】を新しいウインドウで開く

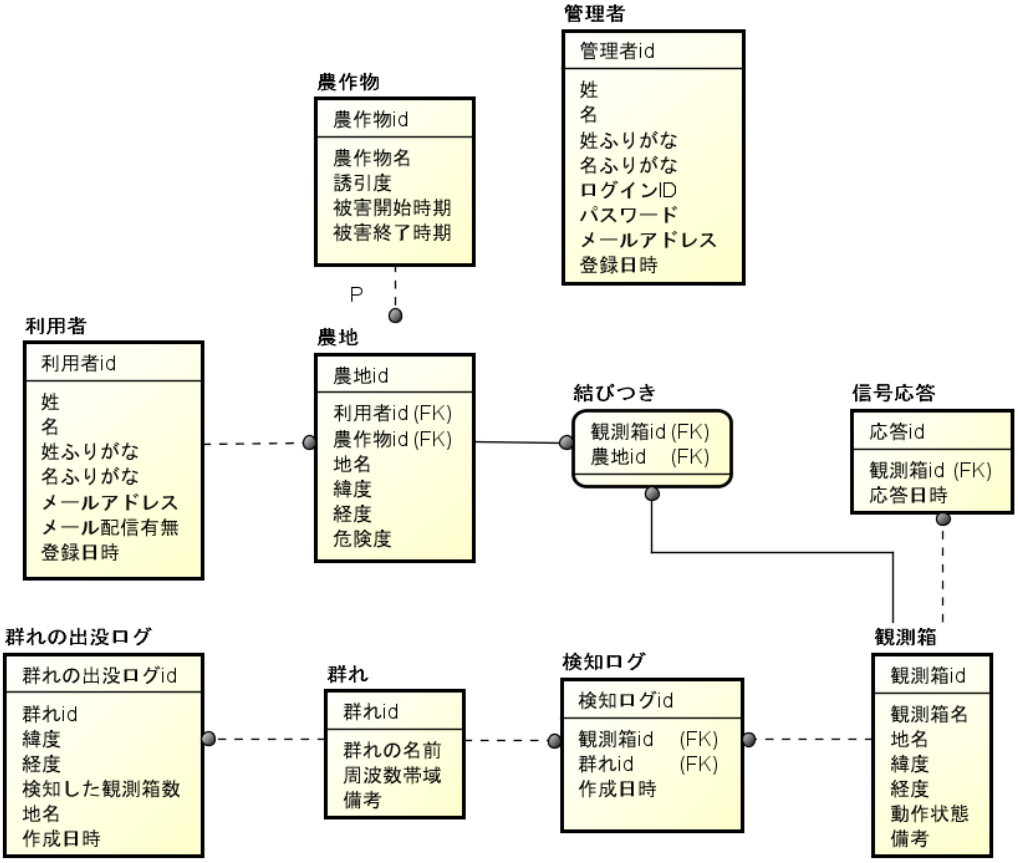
9.マニュアル
このシステムの利用方法を説明するPDFファイルを開く。

10.ログアウト
セッションに保存されているログイン情報を破棄して【A0010.管理者ログインページ】へ遷移する。

11.ログイン中の管理者
現在ログインしている管理者のログインIDを表示する。

B. 設計書類 2. 観測箱シーケンス図




管理者
管理者id
姓
名
姓ふりがな
名ふりがな
ログインID
パスワード
メールアドレス
登録日時
農作物
農作物id
農作物名
誘引度
被害開始時期
被害終了時期
P
利用者
利用者id
姓
名
姓ふりがな
名ふりがな
メールアドレス
メール配信有無
登録日時
農地
農地id
利用者id (FK)
農作物id (FK)
地名
緯度
経度
危険度
結びつき
観測箱id (FK)
農地id (FK)
信号応答
応答id
観測箱id (FK)
応答日時
群れの出没ログ
群れの出没ログid
群れid
緯度
経度
検知した観測箱数
地名
作成日時
群れ
群れid
群れの名前
周波数帯域
備考
検知ログ
検知ログid
観測箱id (FK)
群れid (FK)
作成日時
観測箱
観測箱id
観測箱名
地名
緯度
経度
動作状態
備考

※ 括弧内は見積り工数 (h)
※ 要件定義の期間: 1人当たり15h/週で計算 → 5人で75h/週 → 300h/4週 (1人あたり60hまで)

要件定義期間(合計300h, 5/2 ~ 5/28コアタイム終了まで)

1.0. WBSを作成する(5h : 全員, 合計25h)
2.0. 要件定義書 (作業範囲記述書) を作成する(25h : 天本)
3.0. 品質計画書を作成する(15h : 浜野)
4.0. 技術調査を行う(15h : 天本, 40h : 岡部, 30h : 光岡, 15h : 浜野, 25h : 森, 合計120h)
5.0. ハードウェアのアーキテクチャを決定する(10 : 光岡, 15 : 森, 合計30h)
6.0. JA鈴鹿でキックオフ・ ミーティングを行う(10h : 全員, 合計50h)
7.0. 担当教員ミーティングを行う(5h : 全員, 合計25h)

【総工数】
天本 : 60
岡部 : 60
浜野 : 60
光岡 : 60
森 : 60
合計 : 300

※ 括弧内は見積り工数 (h)
※ S0の期間: 1人当たり23h/週で計算 → 5人で115h/週 → 1035h/9週 (1人あたり207hまで)

S0 (合計900h, 6/6 ~ 8/1コアタイム終了まで)

== プロジェクトマネジメント, ドキュメント関連 == (天本88h, その他のメンバ35h)
1.0 WBSとガントチャートを作成する(10h : 天本)
2.0 リスク管理表を作成する(8h : 天本)
3.0 ミーティング等の進捗管理資料を作成する(10h : 天本)
4.0 中間報告会の資料を作成する(25h : 天本, 5h : 全員, 合計 50h)
5.0 中間報告書を作成する(15h : 全員, 合計75h)
6.0 ミーティングや週間報告を実施する(15h : 全員)

== 品質保証関連 == (浜野85h, その他のメンバ20h)
1.0 テストケースを作成する(40h : 浜野)
2.0 品質保証関連書類を作成する(15 : 浜野)
3.0 品質管理のためのツールを導入する(10 : 浜野)
4.0 テストを実施する(10h : 全員)
5.0 テストで見つかった不具合を修正する(10h : 全員)

== 組み込み機器関連 == (光岡120h, 森120h)
1.0 機材の手配と初期設定を行う(20h : 光岡, 30h : 森)
2.0 回路を構築する(25h : 森)
3.0 アンテナを構築する(25h : 森)
4.0 音声処理を実装する(40h : 光岡)
5.0 RaspberryPi - Server間 通信を実装する(30h : 光岡)
6.0 評価実験の計画書を作成する(10h : 森)
7.0 評価実験を行う(30h : 光岡, 30h : 森)

== 開発, サルの検知アルゴリズム関連 == (天本105h, 岡部120h, 浜野70h, 光岡40h, 森35h)
1.0 開発環境を構築する(15h : 天本, 15h : 岡部, 15h : 浜野, 合計45h)
2.0 猿の位置検出アルゴリズムを実装する(45h : 岡部)
3.0 ベースレイアウトを設計する(25h : 天本, 10h : 岡部, 10h : 浜野, 合計45h)
4.0 データベースを設計する(10h : 天本, 10h : 岡部, 10h : 光岡)
5.0 ログインページを実装する(15h : 浜野, 15h : 岡部, 合計30h)
6.0 管理者トップページを実装する(15h : 天本)
7.0 観測箱管理機能を実装する(10h : 天本, 10h : 光岡, 合計20h)
8.0 利用者管理機能を実装する(10h : 天本, 10h : 浜野, 合計20h)
9.0 管理者管理機能を実装する(10h : 天本, 10h : 岡部, 合計20h)
10.0 マップの表示機能を実装する(10h : 天本, 10h : 光岡)
11.0 ログの管理機能を実装する(15h : 浜野, 15h : 森, 合計30h)
12.0 メール送信機能を実装する(15h : 岡部, 10h : 森, 合計25h)
13.0 観測箱の死活監視機能を実装する(10h : 光岡, 10h : 森)

== その他 == (天本17h, 岡部45h, 浜野20h, 光岡15h, 森20h)
1.0 納品用マニュアルを作成する(15h : 岡部)
2.0 納品時のレビュー用アンケートを作成する(5h : 天本, 5h : 浜野)
3.0 フロート ※予備時間(0h : 天本, 10h : 岡部, 0h : 浜野, 0h : 光岡, 0h : 森)

【総工数】
天本 : 218
岡部 : 200
浜野 : 195
光岡 : 215
森 : 210
合計 : 1038

※ 括弧内は見積り工数 (h)
※ S1の期間 ： 1人当たり23h/週で計算→5人で115h/週→1265h/11週 (工数は1人あたり253hが目安)

S1(合計1265h, 一人あたり253h, 8/15 ~ 11/7コアタイム終了まで)

1.0. プロジェクトマネジメント関連(天本56h, その他のメンバ24h)
- 1.1. 作業進捗報告書を作成する(6h : 天本)
- 1.2. リスク管理表を更新する(4h : 天本)
- 1.3. 担当教員向けミーティング資料を作成する(2h * 6回 = 12h : 天本)
- 1.4. 担当教員向けミーティングを実施する(1h * 全員 * 6回 = 30h)
- 1.5. チームミーティングを実施する(1h * 全員 * 10回 = 50h)
- 1.6. WBS・ガントチャートを作成・更新する(10h : 天本)
- 1.7. 第2回中間報告に参加する(8h * 全員 = 40h)

2.0. 品質計画関連(天本61, 浜野103, その他のメンバ31h)
- 2.1. ユーザビリティ評価のガイドラインを作成する(5h : 浜野)
- 2.2. チームメンバーによるシステムのユーザビリティ評価を行う(3h * 全員 = 15h)
- 2.3. ユーザビリティ改良検討会を実施する(2h * 全員 = 10h)
- 2.4. システムのユーザビリティを改良する(15h : 天本, 25h : 浜野, 合計40h)
- 2.5. ユーザーテストマニュアルを作成する(10h : 浜野)
- 2.6. ユーザーテストを実施する(0.5h * 4回分 = 2h : 浜野)
- 2.7. ユーザーテストの結果の検討会を実施する(1h * 全員 = 5h)
- 2.8. ユーザーテストの結果をシステムに反映する(15h : 天本, 15h : 浜野, 合計30h)
- 2.9. 新規追加分のページ・機能のテストケースを作成する(15h : 浜野)
- 2.10. システムテストを実施する(10h * 全員 = 50h)
- 2.11. テストで見つかった不具合を修正する(15h * 全員 = 75h)

3.0. 組み込み機器関連(110h : 光岡, 85h : 森, 10 : 岡部)
- 3.1. バンドパスフィルタを設計・作成する(30h : 光岡)
- 3.2. 信号とノイズの識別アルゴリズムを設計・作成する(40h : 光岡)
- 3.3. ノイズ除去フィルタの調査と比較実験を行う(10h : 光岡)
- 3.4. 実験の準備を行う(25h : 森)
- 3.5. 観測箱システムの検知範囲の学内実験を行う(30h : 光岡, 30h : 森)
- 3.6. 猿の検知メールの来る頻度や文面 (IN / OUT) などを改良する(10h : 岡部)
- 3.7. 観測箱が猿の検知時に音を出すように改良する(15h : 森)
- 3.8. 観測箱が猿の検知時に光を出すように改良する(15h : 森)

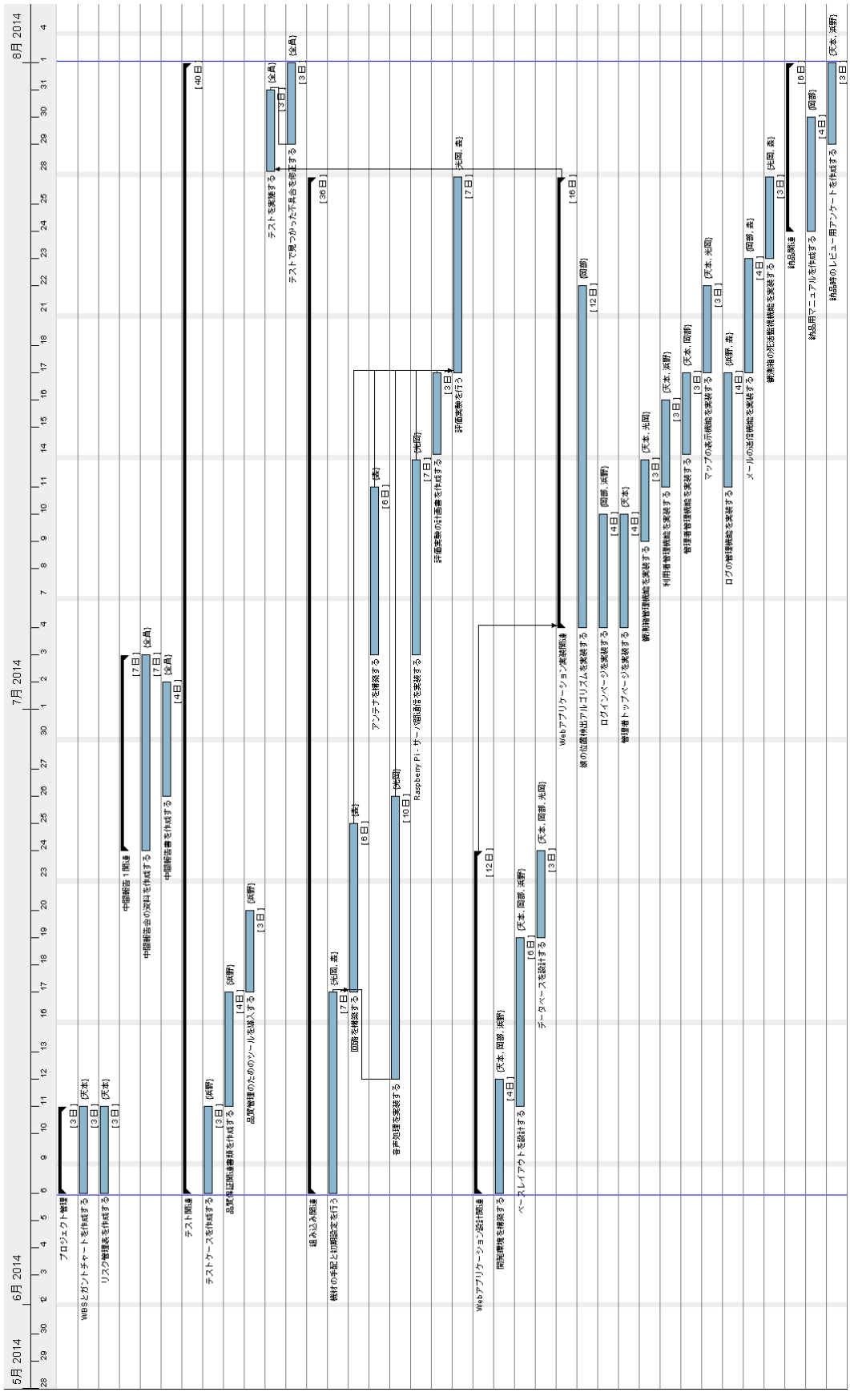
4.0. 開発, サルの検知アルゴリズム関連(40h : 天本, 95h : 岡部, 20h : 浜野)
- 4.1. 農作物の危険度表示のためのアルゴリズムを実装する(20h : 岡部)
- 4.2. 農作物管理機能を実装する(15h : 天本, 10h : 岡部)
- 4.3. 農作物情報を格納できるようにシステムとDBを更新する(10h : 天本, 20h : 岡部, 20h : 浜野)
- 4.4. ヒートマップを実装する(15h : 天本, 15h : 岡部)
- 4.5. 土地情報利用・登録の技術調査を行う(10h : 岡部)
- 4.6. 観測箱の配置に関する調査と計算を行う(20h : 岡部)

5.0. ドキュメント関連(全員100h)
- 5.1. 第2回中間報告の論文を書く(70h * 全員 = 350h)
- 5.2. 第2回中間報告の発表資料を作成する(全員15h, 合計75h)
- 5.3. 特定課題研究報告書に必要なドキュメントを作成する(15h * 全員 = 75h)

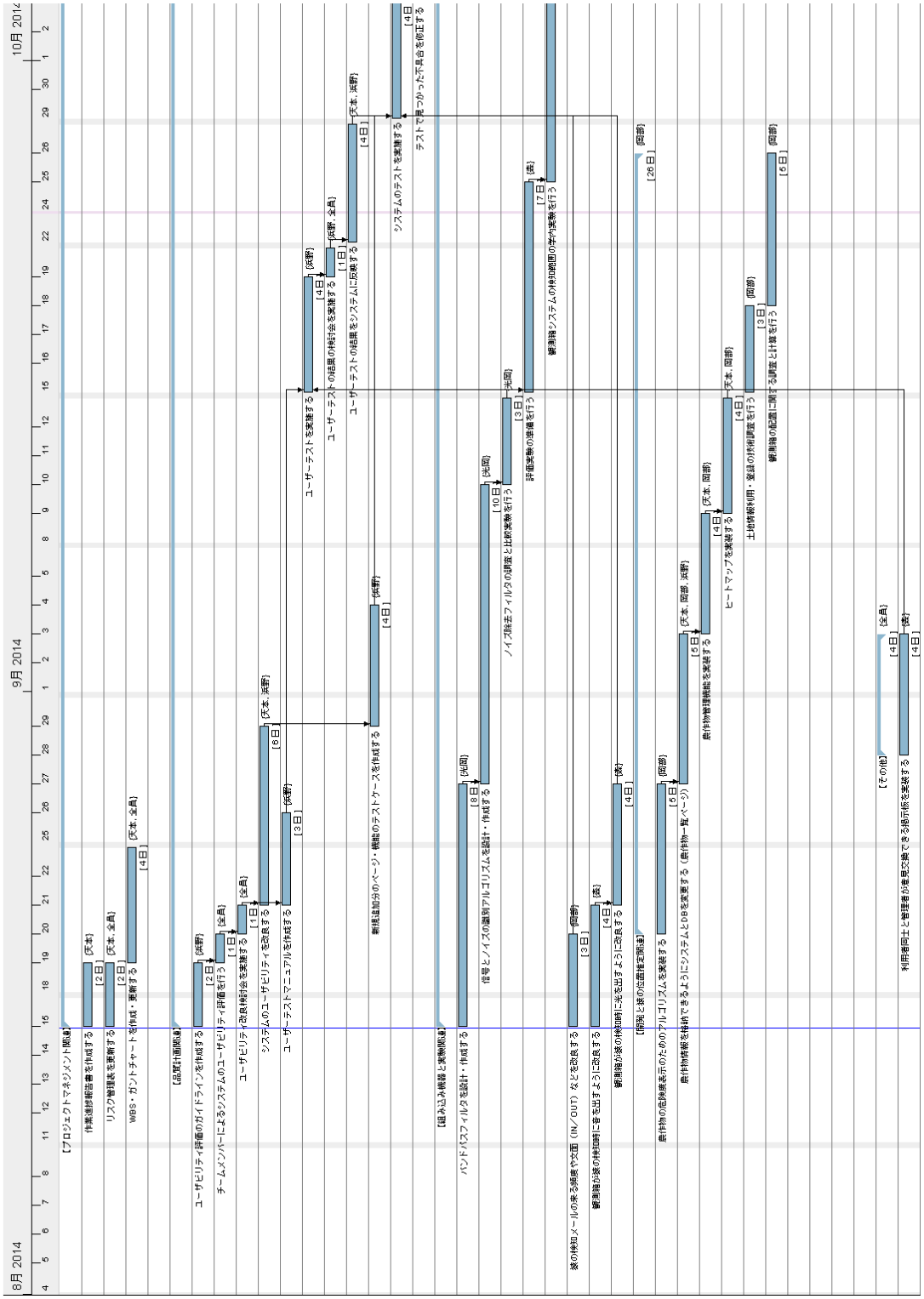
6.0. その他 (15h : 森)
- 6.1. 利用者同士と管理者が意見交換できる掲示板を実装する(15h : 森)

7.0. フロート ※予備時間(0h : 天本, 0h : 岡部, 0h : 浜野, 0h : 光岡, 0h : 森)

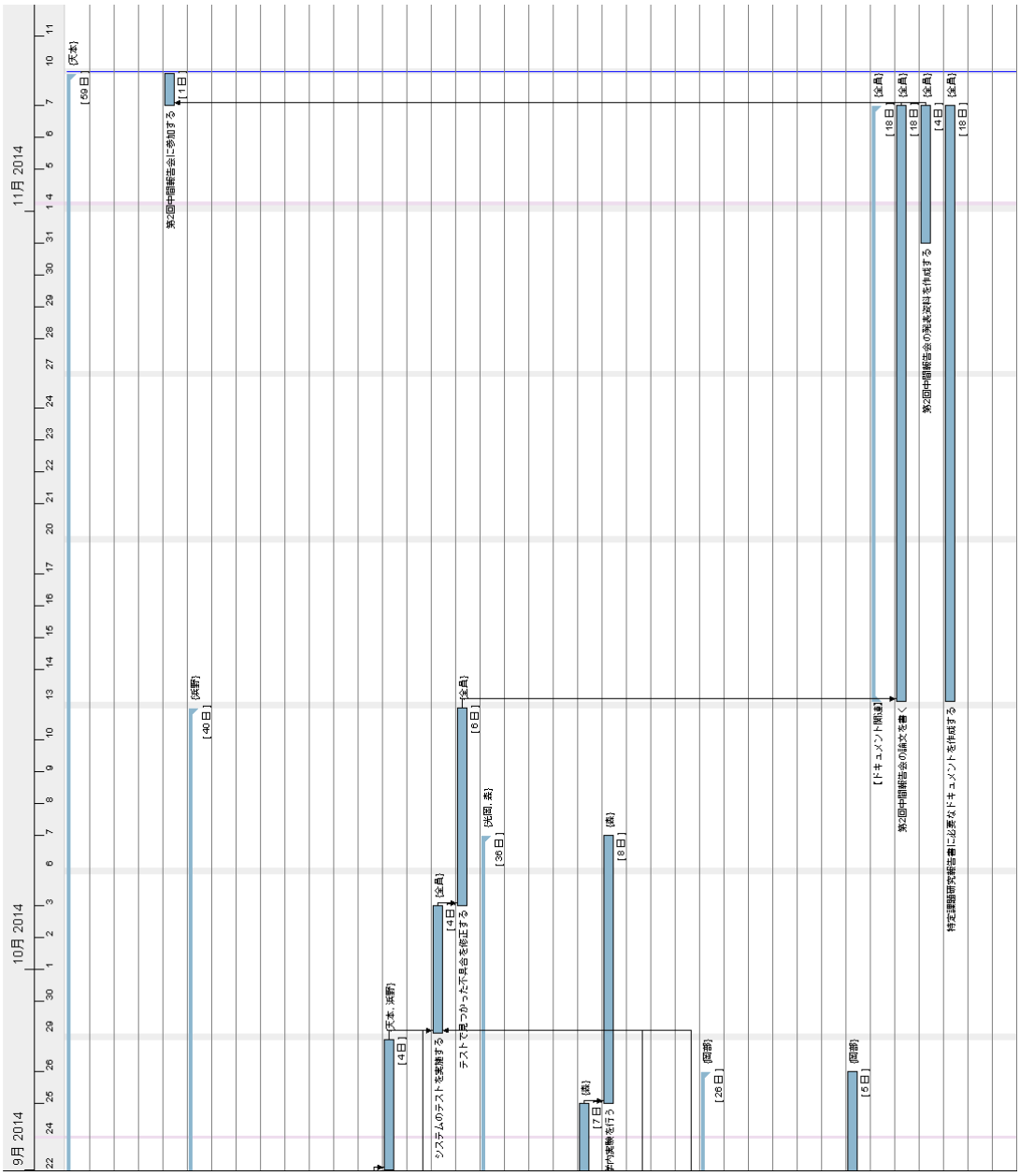
【総工数】
天本 ： 257h
岡部 ： 260h
浜野 ： 247h
光岡 ： 265h
森 ： 255h
合計 ： 1284h


5月 2014
6月 2014
7月 2014
8月 2014
プロジェクト管理 [3日]
WBSとガントチャートを作成する [3日] {伝本}
リスク管理表を作成する [3日] {伝本}
中間報告 1開始
中間報告の資料を作成する [7日] {全員}
中間報告書を作成する [4日] {全員}
テスト開始 [40日]
テストケースを作成する [3日] {浜野}
品質保証関連書類を作成する [4日] {浜野}
品質管理のためのツールを導入する [3日] {浜野}
組み込み開始 [36日]
機材の手配と切断設定を行う [7日] {光岡, 表}
回路を構築する [6日] {表}
アンテナを構築する [6日] {表}
音声処理を実装する [10日] {光岡}
Raspberry Pi・サーバ間通信を実装する [7日] {光岡}
評価実験の計画書を作成する [3日]
評価実験を行う [7日] {光岡, 表}
Webアプリケーション設計開始 [12日]
開発環境を構築する [4日] {伝本, 岡部, 浜野}
ベースレイアウトを設計する [6日] {伝本, 岡部, 浜野}
データベースを設計する [3日] {伝本, 岡部, 光岡}
Webアプリケーション実装開始 [16日]
鉢の位置検出アルゴリズムを実装する [12日] {岡部}
ログインページを実装する [4日] {岡部, 浜野}
管理者トップページを実装する [4日] {伝本}
鉢削除管理機能を実装する [4日] {伝本, 光岡}
利用者管理機能を実装する [3日] {伝本, 浜野}
管理者管理機能を実装する [3日] {伝本, 岡部}
マップの表示機能を実装する [3日] {伝本, 光岡}
ログの管理機能を実装する [4日] {浜野, 表}
メールの送信機能を実装する [4日] {岡部, 表}
鉢削除の完成機能を実装する [4日] {岡部, 表}
納品開始 [3日] {光岡, 表}
納品用マニュアルを作成する [6日]
納品時のレビュー用アンケートを作成する [4日] {岡部}
[3日] {伝本, 浜野}

S1 (1/2)


8月 2014
9月 2014
10月 2014
【プロジェクトマネジメント関連】
作業連絡報告書を作成する
リスク管理表を更新する
WBS・ガントチャートを作成・更新する
【品質計画関連】
ユーザビリティ評価のガイドラインを作成する
チームメンバーによるシステムのユーザビリティ評価を行う
ユーザビリティ改良検討会を実施する
システムのユーザビリティを改良する
ユーザーテストマニュアルを作成する
新規追加分のページ・機能のテストケースを作成する
ユーザーテストを実施する
ユーザーテストの結果の検討会を実施する
ユーザーテストの結果をシステムに反映する
システムのテストを実施する
テストで見つかった不具合を修正する
【組み込み機器と実験関連】
バンドパスフィルタを設計・作成する
信号とノイズの識別アルゴリズムを設計・作成する
ノイズ除去フィルタの調査と比較実験を行う
評価実験の準備を行う
観測箱システムの検知範囲の学内実験を行う
焼の検知メールの来る頻度や文面（IN／OUT）などを改良する
観測箱が焼の検知時に音を出すように改良する
観測箱が焼の検知時に光を出すように改良する
【開発と焼の位置推定関連】
農作物の危険度表示のためのアルゴリズムを実装する
農作物情報を格納できるようにシステムとDBを実装する（農作物一覧ページ）
農作物管理機能を実装する
ヒートマップを実装する
土地情報利用・登録の技術調査を行う
観測箱の配置に関する調査と計算を行う
【その他】
利用者同士と管理者が意見交換できる掲示板を実装する

S1 (2/2)


9月 2014
10月 2014
11月 2014
[59日] [天本]
[1日] 第2回中間報告会に参加する
[40日] [浜野]
[4日] [天本, 浜野]
[4日] システムのテストを実施する [全員]
[6日] テストで見つかった不具合を修正する [全員]
[38日] [光岡, 森]
[7日] [森]
[8日] 外部実験を行う [森]
[26日] [岡部]
[5日] [岡部]
[18日] [ドキュメント関連] [全員]
[18日] 第2回中間報告会の論文を書く [全員]
[4日] 第2回中間報告会の発表資料を作成する [全員]
[18日] 特定課題研究報告書に必要なドキュメントを作成する [全員]

クリティカルタスク	番号	【作業の分類】 or タスク名 ●印：データ管理用の欄	責任者	計画								実績								備考
				開始日	終了日	天本	岡部	浜野	光岡	森	総工数	開始日	終了日	天本	岡部	浜野	光岡	森	総工	
	1.0	【プロジェクトマネジメント関連】	天本	08/15	11/07	56.0	24.0	24.0	24.0	24.0	152.0	08/15	11/07	66.0	31.5	31.5	31.5	31.5	192.0	
	1.1	作業進捗報告書を作成する	天本	08/15	08/18	6.0					6.0	08/15	08/18	7.0					7.0	
	1.2	リスク管理表を更新する	天本	08/15	08/18	4.0					4.0	08/15	08/18	3.0	0.5	0.5	0.5	0.5	5.0	
	1.3	担当教員向けミーティング資料を作成する(2h * 6回)	天本	08/15	11/07	12.0					12.0	08/15	11/07	10.0					10.0	
	1.4	担当教員向けミーティングを実施する(1h * 全員 * 6回)	全員	08/15	11/07	6.0	6.0	6.0	6.0	6.0	30.0	08/15	11/07	6.0	6.0	6.0	6.0	6.0	30.0	
	1.5	チームミーティングを実施する(1h * 全員 * 10回)	天本	08/15	11/07	10.0	10.0	10.0	10.0	10.0	50.0	08/15	11/07	16.0	16.0	16.0	16.0	16.0	80.0	
	1.6	WBS・ガントチャートを作成・更新する	天本	08/19	08/22	10.0					10.0	08/19	08/22	16.0	1.0	1.0	1.0	1.0	20.0	
✓	1.7	第2回中間報告会に参加する	全員	11/07	11/07	8.0	8.0	8.0	8.0	8.0	40.0	11/07	11/07	8.0	8.0	8.0	8.0	8.0	40.0	
	2.0	【品質計画関連】	浜野	08/15	10/10	61.0	31.0	103.0	31.0	31.0	257.0	08/15	11/10	91.0	66.0	126.0	33.0	40.0	356.0	
	2.1	ユーザビリティ評価のガイドラインを作成する	浜野	08/15	08/18			5.0			5.0	08/15	08/18			1.0			1.0	
	2.2	チームメンバーによるシステムのユーザビリティ評価を	全員	08/19	08/19	3.0	3.0	3.0	3.0	3.0	15.0	08/19	08/19	3.0	3.0	5.0	3.0	3.0	17.0	
	2.3	ユーザビリティ改良検討会を実施する	全員	08/20	08/20	2.0	2.0	2.0	2.0	2.0	10.0	08/20	08/20	1.0	1.0	1.0	1.0	1.0	5.0	
	2.4	システムのユーザビリティを改良する	浜野	08/21	08/28	15.0		25.0			40.0	08/21	09/10	32.0	12.0	38.0			82.0	IE対応のために遅延
	2.5	ユーザーテストマニュアルを作成する	浜野	08/21	08/25			10.0			10.0	08/21	08/25			6.0			6.0	
✓	2.6	ユーザーテストを実施する(0.5h * 4回分)	浜野	09/15	09/18			2.0			2.0	09/15	10/02			5.0			5.0	被験者の都合により遅延
✓	2.7	ユーザーテストの結果の検討会を実施する	全員	09/19	09/19	1.0	1.0	1.0	1.0	1.0	5.0	10/02	10/02	1.0	1.0	1.0	1.0	1.0	5.0	
✓	2.8	ユーザーテストの結果をシステムに反映する	浜野	09/22	09/26	15.0		15.0			30.0	10/02	10/22	22.0	8.0	29.0	6.0	6.0	71.0	IE対応のために遅延
	2.9	新規追加分のページ・機能のテストケースを作成する	浜野	08/29	09/03			15.0			15.0	08/29	09/03			7.0			7.0	
✓	2.10	システムのテストを実施する	全員	09/29	10/03	10.0	10.0	10.0	10.0	10.0	50.0	10/22	10/24	8.0	8.0	8.0	8.0	8.0	40.0	
✓	2.11	テストで見つかった不具合を修正する	全員	10/03	10/10	15.0	15.0	15.0	15.0	15.0	75.0	10/24	11/10	24.0	33.0	25.0	14.0	21.0	117.0	
	3.0	【組み込み機器と実験関連】	光岡	08/15	10/07	0.0	0.0	10.0	90.0	85.0	185.0	08/15	11/05	0.0	9.0	3.0	117.0	101.0	230.0	
	3.1	バンドパスフィルタを設計・作成する	光岡	08/15	08/26				30.0		30.0	08/15	08/26				32.0		32.0	
	3.2	信号とノイズの識別アルゴリズムを設計・作成する	光岡	08/26	09/09				20.0		20.0	08/26	09/09				14.0		14.0	
	3.3	ノイズ除去フィルタの調査と比較実験を行う	光岡	09/09	09/12				10.0		10.0	09/09	09/12				12.0		12.0	
	3.4	評価実験の準備を行う	森	09/15	09/24					25.0	25.0	09/15	10/08				26.0	32.0	58.0	機器の不調により遅延
	3.5	観測箱システムの検知範囲の学内実験を行う	森	09/24	10/07				30.0	30.0	60.0	10/08	11/05		3.0	3.0	33.0	45.0	84.0	機器の不調により遅延
	3.6	猿の検知メールの来る頻度や文面 (IN / OUT) などを改	岡部	08/15	08/19			10.0			10.0	08/15	08/19		6.0				6.0	
	3.7	観測箱が猿の検知時に音を出すように改良する	森	08/15	08/20					15.0	15.0	08/15	08/20					12.0	12.0	
	3.8	観測箱が猿の検知時に光を出すように改良する	森	08/20	08/26					15.0	15.0	08/20	08/26					12.0	12.0	

	4.0	【猿の位置推定関連】	岡部	08/20	09/26	40.0	95.0	20.0	0.0	0.0	155.0	08/20	09/26	39.0	87.0	14.0	0.0	0.0	140.0	
✓	4.1	農作物の危険度表示のためのアルゴリズムを実装する	岡部	08/20	08/26		20.0				20.0	08/20	08/29		42.0				42.0	
✓	4.2	農作物情報を格納できるようにシステムとDBを変更する	天本	08/26	09/02	10.0	20.0	20.0			50.0	08/29	09/08	16.0	20.0	14.0			50.0	
✓	4.3	農作物管理機能を実装する	天本	09/03	09/08	15.0	10.0				25.0	09/08	09/12	17.0	7.0				24.0	
✓	4.4	ヒートマップを実装する	天本	09/09	09/12	15.0	15.0				30.0	09/12	09/15	6.0	6.0				12.0	
	4.5	土地情報利用・登録の技術調査を行う	岡部	09/12	09/17		10.0				10.0	未開始	未終了						0.0	このタスクを中止して他の遅れているタスクへ工数を回す
	4.6	観測箱の配置に関する調査と計算を行う	岡部	09/18	09/26		20.0				20.0	09/18	09/26		12.0				12.0	
	5.0	【ドキュメント関連】		10/13	11/06	100.0	100.0	100.0	100.0	100.0	500.0	10/13	11/06	81.0	50.0	70.0	72.0	84.0	357.0	
✓	5.1	第2回中間報告の論文を書く	全員	10/13	11/06	70.0	70.0	70.0	70.0	70.0	350.0	10/13	11/06	55.0	40.0	62.0	58.0	73.0	288.0	
✓	5.2	第2回中間報告会の発表資料を作成する	全員	10/31	11/06	15.0	15.0	15.0	15.0	15.0	75.0	10/31	11/06	23.0	4.0	4.0	4.0	4.0	39.0	天本が雛形を作ることで他のメンバの工数を節約
	5.3	特定課題研究報告書に必要なドキュメントを作成する	全員	10/13	11/06	15.0	15.0	15.0	15.0	15.0	75.0	10/13	11/06	3.0	6.0	4.0	10.0	7.0	30.0	優先度は高くないので、他のタスクへ工数を回す
	6.0	【その他】		08/28	09/02	0.0	0.0	0.0	0.0	15.0	15.0	08/28	09/05	0.0	0.0	0.0	0.0	26.0	26.0	
	6.1	利用者同士と管理者が意見交換できる掲示板を実装する	森	08/28	09/02					15	15.0	08/28	09/05					26.0	26.0	
	6.2	フロート ※自由に使える作業時間	全員			0	0	0	0	0	0.0			0.0		0.0	0.0	0.0	0.0	
	•	【合計】 ※目安は一人あたり253h, 総工数1265h				257	250	257	245	255	1264			277	244	245	254	283	1,301	

リスク評価 SO

番号	要因の大分類	要因の小分類	事象	影響度						発生確率						リスクスコア
				天本	岡部	浜野	光岡	森	AVE	天本	岡部	浜野	光岡	森	AVE	
1	学生	−	授業の課題で手一杯になり作業に取り組めない	1	1	2	2	3	1.8	2	3	2	3	2	2.4	4.32
2	学生	−	夏季休暇中に集まって作業できない	2	1	1	1	1	1.2	3	3	3	2	3	2.8	3.36
3	学生	−	メンバーが離脱する（病気、怪我、不登校）	3	3	1	3	3	2.6	1	1	1	1	1	1	2.6
4	学生	−	作業の進捗を正確に報告しない（できない）	2	2	1	2	3	2	3	2	2	2	2	2.2	4.4
5	学生	−	作業の遅れを容認してしまう	3	1	2	2	2	2	3	3	2	3	3	2.8	5.6
6	学生	−	困っていることをメンバ間で共有（解決）できない	2	2	2	2	2	2	3	1	3	1	2	2	4
7	プロジェクト	ステークホルダ	JAの担当者との連絡に時間がかかる	1	2	2	1	1	1.4	3	2	1	3	3	2.4	3.36
8	プロジェクト	ステークホルダ	JAの担当者が仕様変更を要求する	2	3	1	3	3	2.4	2	1	2	2	2	1.8	4.32
9	プロジェクト	ステークホルダ	JAの担当者のITリテラシが不足しており、 システムを運用（評価）できない	3	1	2	2	2	2	2	2	1	3	3	2.2	4.4
10	プロジェクト	ステークホルダ	JAの担当者が不在になる	3	1	1	2	3	2	1	1	1	1	1	1	2
11	プロジェクト	ステークホルダ	知的財産（特許）に関してのいざこざが発生する	2	1	1	3	3	2	2	1	2	1	1	1.4	2.8
12	プロジェクト	ステークホルダ	システムと、関係者の実業務との整合性が取れない	1	2	3	3	3	2.4	1	3	2	2	1	1.8	4.32
13	プロジェクト	スケジュール遅延	作業の工数見積もりが甘い	3	3	3	2	2	2.6	3	3	2	3	3	2.8	7.28
14	プロジェクト	スケジュール遅延	仕様の検討漏れがある	2	2	3	2	3	2.4	3	3	2	3	3	2.8	6.72
15	プロジェクト	スケジュール遅延	学校の作業〆切が変更になる	2	3	1	2	1	1.8	2	2	1	3	3	2.2	3.96
16	プロジェクト	スケジュール遅延	予期せぬ作業が増える（作業の洗い出しが不十分）	2	2	3	2	2	2.2	3	3	2	3	3	2.8	6.16
17	開発	−	開発環境に不具合が出る	3	3	2	3	3	2.8	2	2	2	1	2	1.8	5.04
18	開発	−	ソースコードの保守（改良）が難しくなる	1	2	1	1	2	1.4	3	1	1	1	1	1.4	1.96
19	開発	−	システムとその関連技術に脆弱性が発見される	3	3	2	1	3	2.4	2	1	1	1	1	1.2	2.88
20	開発	−	システムが納品先（JA鈴鹿）の環境で動かない	3	3	3	3	3	3	2	3	2	3	1	2.2	6.6
21	開発	−	技術調査不足によって開発が滞る	2	2	3	2	3	2.4	3	2	3	3	1	2.4	5.76
22	開発	−	専門用語、業務用語が統一されていない	1	1	2	1	1	1.2	3	3	1	2	2	2.2	2.64
23	開発	−	非機能用件が明確化できていない	2	2	2	2	2	2	3	3	2	2	2	2.4	4.8
24	開発	−	バグ発見時の対処法が統一されていない	1	1	1	1	1	1	2	1	1	1	1	1.2	1.2
25	開発	−	ドキュメントの作成・更新管理ができていない	1	1	1	1	2	1.2	3	3	2	2	2	2.4	2.88
26	外部	−	システムの機材・学生のPCが壊れる	2	2	2	1	3	2	2	1	1	2	1	1.4	2.8
27	外部	−	実験場の確保が難しい（システムの設置場所がない）	2	2	2	2	3	2.2	2	1	1	1	1	1.2	2.64
28	外部	−	天候の問題で実験が実施できない	3	2	3	3	3	2.8	1	1	1	1	1	1	2.8
29	外部	−	予算が足りない	3	3	2	3	3	2.8	2	1	1	1	1	1.2	3.36

リスク対応策 S0

番号	事象	リスクスコア	リスクレベル	予防策	発生時の対応策	担当者
13	作業の工数見積もりが甘い	7.28	高	週間報告を実施し、作業の進捗をできるだけ正確に把握する	WBS, ガントチャートの見直しを行う	天本
14	仕様の検討漏れがある	6.72	高	設計などのドキュメントは必ず3人以上でのレビューを実施する	翌コアタイム時に、3人以上で仕様を検討し直す	天本
20	システムが納品先（JA鈴鹿）の環境で動かない	6.6	高	JA鈴鹿の運用環境（OS、ブラウザなど）をドキュメントに残し、共有する	各機能の担当者が、動作環境に合わせて開発し直す	浜野
16	予期せぬ作業が増える（作業の洗い出しが不十分）	6.16	高	2週間毎にWBSとスケジュールを見直し、チームレビューを実施する	作業の担当者が責任を持ち、手が空いているメンバーと対処する	天本
21	技術調査不足によって開発が滞る	5.76	中	技術調査担当者は、コアタイム毎に、調査結果をメンバと共有する		全員
5	作業の遅れを容認してしまう	5.6	中	週間報告時に作業の進捗を確認する		全員
17	開発環境に不具合が出る	5.04	中	なるべくノウハウのある開発基盤を採用する		森
23	非機能用件が明確化できていない	4.8	中	予め必要と思われる非機能要件を洗い出し、JA鈴鹿に提示・確認する		浜野
4	作業の進捗を正確に報告しない（できない）	4.4	中	週間報告を実施し、作業の進捗をできるだけ正確に把握する		天本
9	JAの担当者のITリテラシが不足しており、 システムを運用（評価）できない	4.4	中	PC初心者でも運用できるように注意して運用マニュアルを作成する		天本
1	授業の課題で手一杯になり作業に取り組めない	4.32	中	課題は早め早めに終わらせておく		全員
8	JAの担当者が仕様変更を要求する	4.32	中	要件定義書をJA鈴鹿に提示し、合意を得ておく		天本
12	システムと、関係者の実業務との整合性が取れない	4.32	中	システムの運用イメージをJA鈴鹿に提示し、問題がないことを確認する		天本
6	困っていることをメンバ間で共有（解決）できない	4	中	週間報告を実施し、困っていることをチーム内で共有する		天本
15	学校の作業〆切が変更になる	3.96	中	常に最悪の場合を想定してスケジュールを組んでおく		天本
2	夏季休暇中に集まって作業できない	3.36	中	遠隔会議が行える環境を整える		天本
7	JAの担当者との連絡に時間がかかる	3.36	中	知りたい情報は、早めに担当者に聞いておく		浜野
29	予算が足りない	3.36	中	予算状況を可視化し、効率的に運用する		天本
19	システムとその関連技術に脆弱性が発見される	2.88	低			
25	ドキュメントの作成・更新管理ができていない	2.88	低			
11	知的財産（特許）に関してのいざこざが発生する	2.8	低			
26	システムの機材・学生のPCが壊れる	2.8	低			
28	天候の問題で実験が実施できない	2.8	低			
22	専門用語、業務用語が統一されていない	2.64	低			
27	実験場の確保が難しい（システムの設置場所がない）	2.64	低			
3	メンバーが離脱する（病気、怪我、不登校）	2.6	低			
10	JAの担当者が不在になる	2	低			
18	ソースコードの保守（改良）が難しくなる	1.96	低			
24	バグ発見時の対処法が統一されていない	1.2	低			

リスク評価S1

番号	要因の大分類	要因の小分類	事象	影響度						発生確率						リスクスコア
				天本	岡部	浜野	光岡	森	AVE	天本	岡部	浜野	光岡	森	AVE	
1	学生	－	授業の課題で手一杯になり作業に取り組めない	2	2	2	1	3	2	1	1	1	1	1	1	2
2	学生	－	夏季休暇中に集まって作業できない	2	2	2	3	1	2	3	3	3	3	3	3	6
3	学生	－	メンバーが離脱する（病気、怪我、不登校）	3	3	3	2	3	2.8	1	1	1	2	1	1.2	3.36
4	学生	－	作業の進捗を正確に報告しない（できない）	3	2	2	2	2	2.2	3	2	3	3	2	2.6	5.72
5	学生	－	作業の遅れを容認してしまう	3	2	2	2	1	2	3	2	3	3	3	2.8	5.6
6	学生	－	困っていることをメンバ間で共有（解決）できない	2	2	2	2	3	2.2	2	2	3	3	2	2.4	5.28
7	プロジェクト	ステークホルダ	JAの担当者との連絡に時間がかかる	2	2	1	1	3	1.8	2	2	2	1	2	1.8	3.24
8	プロジェクト	ステークホルダ	JAの担当者が仕様変更を要求する	2	2	1	1	3	1.8	2	3	1	1	1	1.6	2.88
9	プロジェクト	ステークホルダ	JAの担当者のITリテラシが不足しており、 システムを運用（評価）できない	3	3	3	2	3	2.8	3	2	2	3	3	2.6	7.28
10	プロジェクト	ステークホルダ	JAの担当者が不在になる	2	2	2	1	3	2	1	1	1	1	1	1	2
11	プロジェクト	ステークホルダ	知的財産（特許）に関してのいざこざが発生する	2	3	1	1	3	2	2	1	1	1	1	1.2	2.4
12	プロジェクト	ステークホルダ	システムと、関係者の実業務との整合性が取れない	2	2	2	2	2	2	2	1	2	1	1	1.4	2.8
13	プロジェクト	スケジュール遅延	作業の工数見積もりが甘い	2	3	2	2	2	2.2	2	2	2	2	2	2	4.4
14	プロジェクト	スケジュール遅延	仕様の検討漏れがある	3	3	2	3	2	2.6	3	2	3	3	2	2.6	6.76
15	プロジェクト	スケジュール遅延	学校の作業〆切が変更になる	2	3	2	2	2	2.2	2	1	1	1	2	1.4	3.08
16	プロジェクト	スケジュール遅延	予期せぬ作業が増える（作業の洗い出しが不十分）	2	3	2	2	2	2.2	3	2	2	2	3	2.4	5.28
17	開発	－	開発環境に不具合が出る	1	2	3	1	3	2	1	1	3	1	1	1.4	2.8
18	開発	－	ソースコードの保守（改良）が難しくなる	2	2	3	3	2	2.4	2	2	3	3	1	2.2	5.28
19	開発	－	システムとその関連技術に脆弱性が発見される	2	2	2	2	3	2.2	1	2	2	1	1	1.4	3.08
20	開発	－	システムが納品先（JA鈴鹿）の環境で動かない	3	3	3	2	3	2.8	1	2	2	1	1	1.4	3.92
21	開発	－	技術調査不足によって開発が滞る	2	2	2	2	2	2	2	2	2	2	2	2	4
22	開発	－	専門用語、業務用語が統一されていない	1	1	1	2	2	1.4	1	1	1	1	1	1	1.4
23	開発	－	非機能用件が明確化できていない	2	3	3	3	2	2.6	2	2	2	3	2	2.2	5.72
24	開発	－	バグ発見時の対処法が統一されていない	1	1	2	1	2	1.4	1	1	1	1	1	1	1.4
25	開発	－	ドキュメントの作成・更新管理ができていない	2	2	2	1	2	1.8	1	1	2	2	1	1.4	2.52
26	外部	－	システムの機材・学生のPCが壊れる	2	3	3	3	3	2.8	1	1	1	2	2	1.4	3.92
27	外部	－	実験場の確保が難しい（システムの設置場所がない）	2	2	2	2	3	2.2	1	2	1	2	2	1.6	3.52
28	外部	－	天候の問題で実験が実施できない	2	1	1	2	3	1.8	1	1	1	1	1	1	1.8
29	外部	－	予算が足りない	2	2	2	2	3	2.2	2	2	1	1	1	1.4	3.08

リスク対応策 S1

番号	事象	リスクスコア	リスクレベル	予防策	発生時の対応策	担当者
9	JAの担当者のITリテラシが不足しており、 システムを運用（評価）できない	7.28	高	PC初心者でも運用できるように注意して運用マニュアルを作成する	システムを評価できる別の指標を提示し、 それを元に評価してもらう	天本
14	仕様の検討漏れがある	6.76	高	設計などのドキュメントは必ず3人以上でのレビューを実施する	翌コアタイム時に、3人以上で仕様を検討し直す	天本
2	夏季休暇中に集まって作業できない	6	高	遠隔会議が行える環境を整える	週間報告会を普段より密に実施する	天本
4	作業の進捗を正確に報告しない（できない）	5.72	中	週間報告を実施し、作業の進捗をできるだけ正確に把握する		天本
23	非機能用件が明確化できていない	5.72	中	予め必要と思われる非機能要件を洗い出し、JA鈴鹿に提示・確認する		天本
5	作業の遅れを容認してしまう	5.6	中	週間報告時に作業の進捗を確認する		全員
6	困っていることをメンバ間で共有（解決）できない	5.28	中	週間報告を実施し、困っていることをチーム内で共有する		全員
16	予期せぬ作業が増える（作業の洗い出しが不十分）	5.28	中	2週間毎にWBSとスケジュールを見直し、チームレビューを実施する		天本
18	ソースコードの保守（改良）が難しくなる	5.28	中	どこにどんなファイルがあるかをまとめた書類をつくる。レビューを実施		森
13	作業の工数見積もりが甘い	4.4	中	週間報告を実施し、作業の進捗をできるだけ正確に把握する		天本
21	技術調査不足によって開発が滞る	4	中	技術調査担当者は、コアタイム毎に、調査結果をメンバと共有する		全員
20	システムが納品先（JA鈴鹿）の環境で動かない	3.92	中	JA鈴鹿の運用環境（OS、ブラウザなど）をドキュメントに残し、共有する		浜野
26	システムの機材・学生のPCが壊れる	3.92	中	バックアップを用意しておく		全員
27	実験場の確保が難しい（システムの設置場所がない）	3.52	中	早め早めに顧客と連絡を取り合い、準備を進めておく		全員
3	メンバーが離脱する（病気、怪我、不登校）	3.36	中	知識の共有を行い、作業を分担できるようにしておく		全員
7	JAの担当者との連絡に時間がかかる	3.24	中	知りたい情報は、早めに担当者に聞いておく		全員
15	学校の作業〆切が変更になる	3.08	中	常に最悪の場合を想定してスケジュールを組んでおく		天本
19	システムとその関連技術に脆弱性が発見される	3.08	中	脆弱性のチェックを定期的に実施する		天本
29	予算が足りない	3.08	中	予算状況を可視化し、効率的に運用する		天本
8	JAの担当者が仕様変更を要求する	2.88	低			
12	システムと、関係者の実業務との整合性が取れない	2.8	低			
17	開発環境に不具合が出る	2.8	低			
25	ドキュメントの作成・更新管理ができていない	2.52	低			
11	知的財産（特許）に関してのいざこざが発生する	2.4	低			
1	授業の課題で手一杯になり作業に取り組めない	2	低			
10	JAの担当者が不在になる	2	低			
28	天候の問題で実験が実施できない	1.8	低			
22	専門用語、業務用語が統一されていない	1.4	低			
24	バグ発見時の対処法が統一されていない	1.4	低			

リスク評価 S2

番号	要因の大分類	要因の小分類	事象	影響度						発生確率						リスクスコア
				天本	岡部	浜野	光岡	森	AVE	天本	岡部	浜野	光岡	森	AVE	
1	学生	−	授業の課題で手一杯になり作業に取り組めない	1	2	2	1	1	1.4	1	1	1	1	1	1	1.4
2	学生	−	夏季休暇中に集まって作業できない	1	2	2	1	1	1.4	1	1	2	1	1	1.2	1.68
3	学生	−	メンバーが離脱する（病気、怪我、不登校）	3	3	3	2	3	2.8	1	1	1	1	1	1	2.8
4	学生	−	作業の進捗を正確に報告しない（できない）	2	2	2	2	3	2.2	1	2	2	2	1	1.6	3.52
5	学生	−	作業の遅れを容認してしまう	2	2	2	2	3	2.2	2	2	2	3	3	2.4	5.28
6	学生	−	困っていることをメンバ間で共有（解決）できない	1	2	2	1	3	1.8	1	1	3	1	1	1.4	2.52
7	プロジェクト	ステークホルダ	JAの担当者との連絡に時間がかかる	1	1	1	1	1	1	2	3	1	2	3	2.2	2.2
8	プロジェクト	ステークホルダ	JAの担当者が仕様変更を要求する	2	3	3	2	3	2.6	1	2	2	1	1	1.4	3.64
9	プロジェクト	ステークホルダ	JAの担当者のITリテラシが不足しており、 システムを運用（評価）できない	2	2	2	2	3	2.2	1	2	1	2	3	1.8	3.96
10	プロジェクト	ステークホルダ	JAの担当者が不在になる	2	2	2	1	3	2	1	2	1	1	1	1.2	2.4
11	プロジェクト	ステークホルダ	知的財産（特許）に関してのいざこざが発生する	1	2	1	1	3	1.6	1	1	1	1	1	1	1.6
12	プロジェクト	ステークホルダ	システムと、関係者の実業務との整合性が取れない	1	3	1	1	3	1.8	1	1	1	1	3	1.4	2.52
13	プロジェクト	スケジュール遅延	作業の工数見積もりが甘い	1	2	2	3	3	2.2	1	2	2	2	3	2	4.4
14	プロジェクト	スケジュール遅延	仕様の検討漏れがある	2	2	3	2	3	2.4	1	2	1	2	3	1.8	4.32
15	プロジェクト	スケジュール遅延	学校の作業〆切が変更になる	1	3	2	1	3	2	3	1	2	1	3	2	4
16	プロジェクト	スケジュール遅延	予期せぬ作業が増える（作業の洗い出しが不十分）	2	2	2	2	3	2.2	1	2	2	3	3	2.2	4.84
17	開発	−	開発環境に不具合が出る	3	3	2	3	3	2.8	1	2	1	2	3	1.8	5.04
18	開発	−	ソースコードの保守（改良）が難しくなる	1	1	1	2	3	1.6	2	1	1	2	3	1.8	2.88
19	開発	−	システムとその関連技術に脆弱性が発見される	2	2	2	2	3	2.2	1	1	1	1	1	1	2.2
20	開発	−	システムが納品先（JA鈴鹿）の環境で動かない	3	3	3	1	3	2.6	1	1	1	2	1	1.2	3.12
21	開発	−	技術調査不足によって開発が滞る	1	2	2	1	3	1.8	1	1	2	1	1	1.2	2.16
22	開発	−	専門用語、業務用語が統一されていない	1	1	1	1	1	1	1	1	1	1	1	1	1
23	開発	−	非機能用件が明確化できていない	2	1	2	2	1	1.6	1	1	1	2	3	1.6	2.56
24	開発	−	バグ発見時の対処法が統一されていない	1	1	1	2	1	1.2	1	2	2	1	3	1.8	2.16
25	開発	−	ドキュメントの作成・更新管理ができていない	2	1	2	2	1	1.6	2	2	2	2	3	2.2	3.52
26	外部	−	システムの機材・学生のPCが壊れる	2	2	2	2	3	2.2	1	2	1	2	1	1.4	3.08
27	外部	−	実験場の確保が難しい（システムの設置場所がない）	2	2	2	2	3	2.2	2	2	1	1	1	1.4	3.08
28	外部	−	天候の問題で実験が実施できない	1	1	1	2	3	1.6	1	2	1	2	1	1.4	2.24
29	外部	−	予算が足りない	1	1	1	1	3	1.4	1	2	1	1	1	1.2	1.68

リスク対応策 S2

番号	事象	リスクスコア	リスクレベル	予防策	発生時の対応策	担当者
5	作業の遅れを容認してしまう	5.28	中	週間報告時に作業の進捗を確認する		全員
17	開発環境に不具合が出る	5.04	中	なるべくノウハウのある開発基盤を採用する		全員
16	予期せぬ作業が増える(作業の洗い出しが不十分)	4.84	中	2週間毎にWBSとスケジュールを見直し、チームレビューを実施する		天本
13	作業の工数見積もりが甘い	4.4	中	週間報告を実施し、作業の進捗をできるだけ正確に把握する		全員
14	仕様の検討漏れがある	4.32	中	設計などのドキュメントは必ず3人以上でのレビューを実施する		全員
15	学校の作業〆切が変更になる	4	中	常に最悪の場合を想定してスケジュールを組んでおく		天本
9	JAの担当者のITリテラシが不足しており、 システムを運用(評価)できない	3.96	中	PC初心者でも運用できるように注意して運用マニュアルを作成する		岡部
8	JAの担当者が仕様変更を要求する	3.64	中	要件定義書をJA鈴鹿に提示し、合意を得ておく		天本
4	作業の進捗を正確に報告しない(できない)	3.52	中	週間報告を実施し、作業の進捗をできるだけ正確に把握する		全員
25	ドキュメントの作成・更新管理ができていない	3.52	中	定期的にドキュメントレビューを実施する		全員
20	システムが納品先(JA鈴鹿)の環境で動かない	3.12	中	JA鈴鹿の運用環境(OS、ブラウザなど)をドキュメントに残し、共有する		浜野
26	システムの機材・学生のPCが壊れる	3.08	中	定期的にバックアップを取っておく		全員
27	実験場の確保が難しい(システムの設置場所がない)	3.08	中	早め早めに顧客と連絡を取り合い、準備を進めておく		天本
18	ソースコードの保守(改良)が難しくなる	2.88	低			
3	メンバーが離脱する(病気、怪我、不登校)	2.8	低			
23	非機能用件が明確化できていない	2.56	低			
6	困っていることをメンバ間で共有(解決)できない	2.52	低			
12	システムと、関係者の実業務との整合性が取れない	2.52	低			
10	JAの担当者が不在になる	2.4	低			
28	天候の問題で実験が実施できない	2.24	低			
7	JAの担当者との連絡に時間がかかる	2.2	低			
19	システムとその関連技術に脆弱性が発見される	2.2	低			
21	技術調査不足によって開発が滞る	2.16	低			
24	バグ発見時の対処法が統一されていない	2.16	低			
2	夏季休暇中に集まって作業できない	1.68	低			
29	予算が足りない	1.68	低			
11	知的財産(特許)に関してのいざこざが発生する	1.6	低			
1	授業の課題で手一杯になり作業に取り組めない	1.4	低			
22	専門用語、業務用語が統一されていない	1	低			

獣類発見支援システム

JA 鈴鹿　＋　筑波大学

目次

本マニュアルについて 3

獣類発見支援システムの概要 4

このシステムでできること 4

サルの監視の自動化 4

獣害に遭う可能性の高い農地の自動判別・警告 4

近隣住民への直感的通知 4

蓄積された検知データの活用 4

画面と機能の解説 6

ログインページ 6

利用者画面 7

利用者トップページ 7

管理者画面 8

メニューバー 8

管理者トップページ 9

利用者管理画面 10

管理者管理画面 13

観測箱管理画面 16

観測動物管理画面 19

農作物管理画面 22

検知履歴 25

掲示板 26

用語 28

利用者 28

検知メールの受け取り 28

管理者 28

観測箱 28

最終応答時間 28

観測動物 28

周波数 28

農作物 29

獣害被害開始想定月、終了想定月 29

本マニュアルについて

本マニュアルは、以下の情報について記載しています。必要に応じてご参照ください。

本マニュアルはシステムからダウンロード可能です。

1. 獣類発見支援システムの概要

   「このシステムでできること」を簡単に解説しています。

2. 画面と機能の解説

   システムの機能を画面ごとに解説しています。

3. 用語

   本マニュアルで用いている用語について解説しています。

獣類発見支援システムの概要

このシステムでできること

サルの監視の自動化

サルの現在位置を，Web ブラウザを用いて，いつでも地図で確認することができます。

獣害に遭う可能性の高い農地の自動判別・警告

農地の位置情報や作物情報を元に，サルの次の移動先を事前に予測することができます。そして獣害に遭う可能性の高い農地の所有者に警告メールを自動で送付することができます。

近隣住民への通知

農地の近くにサルが現れた場合，システムがランプや警告音で近隣の住民にサルの出没を通知することができます。

蓄積された検知データの活用

システムに蓄積された検知データを分析して，今後の獣害対策に利用できます。

獣害対策に関する情報の集中管理できます。

被害状況や，関係者の意見をシステム上に記録することができます。

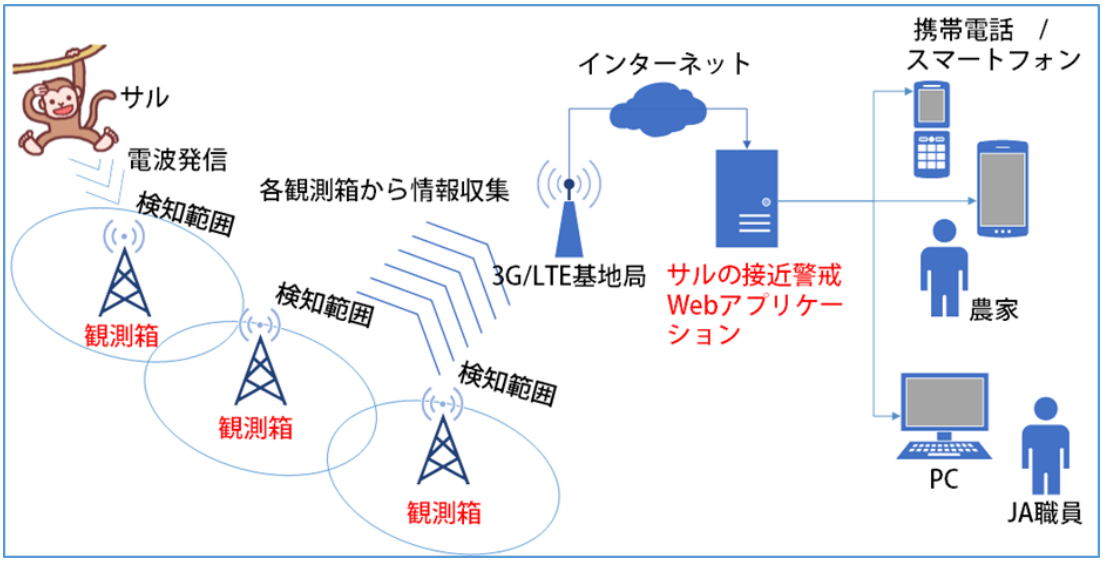


図 1　システムのイメージ図

画面と機能の解説

ログインページ

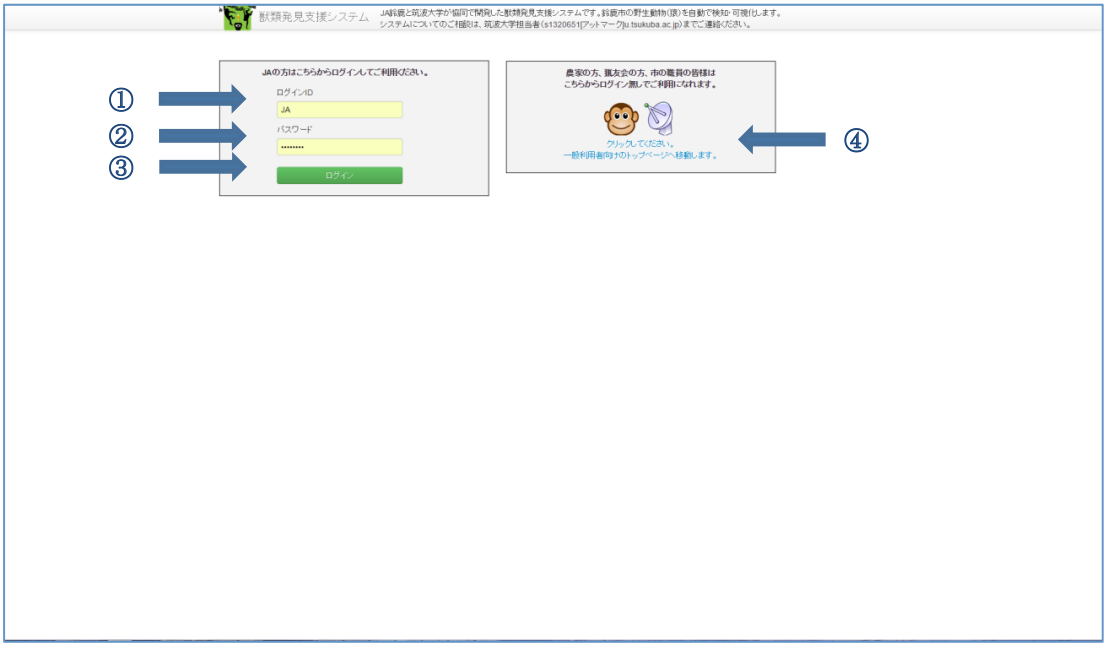


図 2　ログインページ

システムにアクセスすると最初に表示される画面です。

管理者の方はログインすることで、管理ページに移動することができます。

利用者の方は利用者トップページに移動することができます。

番号	名称	備考
①	ログイン ID 入力フォーム	管理者ページに移動するためのログイン ID を入力します。
②	パスワード入力フォーム	管理者ページに移動するためのパスワードを入力します。
③	ログインボタン	ログインを行います。
④	利用者トップページへのリンク	利用者トップページに移動します。

利用者画面

利用者トップページ

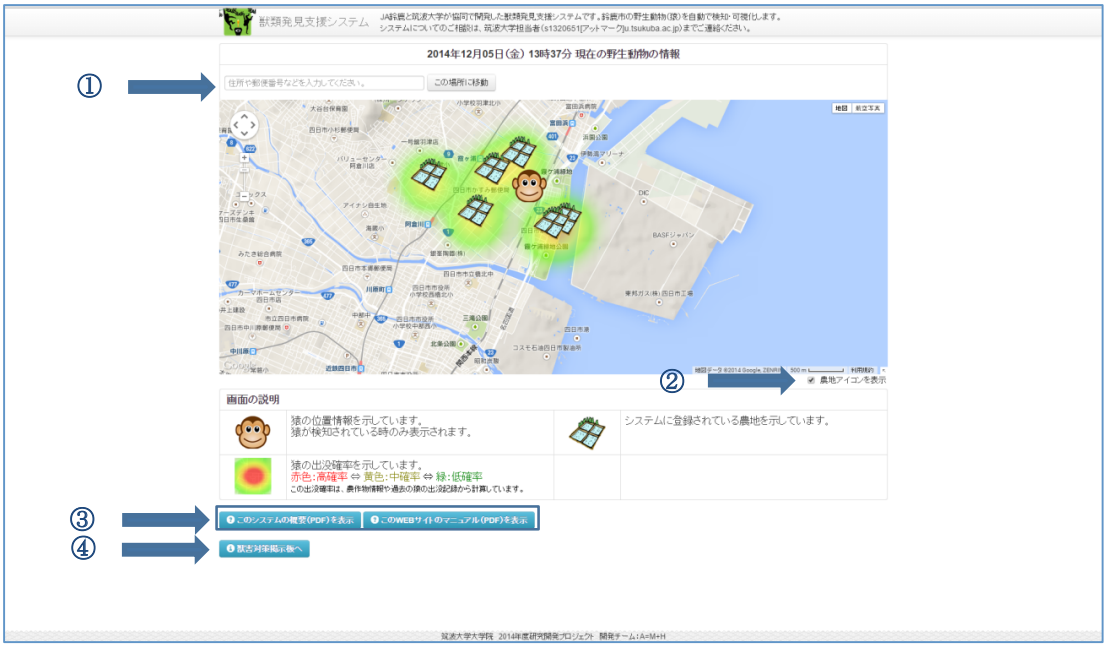


図 3　利用者トップページ

利用者トップページです。

農地の位置、検知された害獣の位置、獣害被害が予想される農地の位置が表示されます。

番号	名称	備考
①	位置検索フォーム	マップの表示中心を変更します。
②	アイコン表示 ON/OFF チェックボックス	マップ上の農地アイコンの表示 ON/OFF を切り替えます。
③	ドキュメント表示ボタン	各種ドキュメントを表示します。
④	掲示板移動ボタン	獣害対策掲示板へ移動します。

管理者画面

メニューバー

図 4　メニューバー

各管理者機能へアクセスができます。

番号	名称	備考
①	トップページ移動リンク	管理者トップページに移動します。
②	トップページ移動リンク	管理者トップページに移動します。
③	利用者一覧ページ移動リンク	利用者一覧ページに移動します。
④	管理者一覧ページ移動リンク	管理者一覧ページに移動します。
⑤	観測箱一覧ページ移動リンク	観測箱一覧ページに移動します。
⑥	観測動物一覧ページ移動リンク	観測動物一覧ページに移動します。
⑦	農作物一覧ページ移動リンク	農作物一覧ページに移動します。
⑧	検知履歴ページ移動リンク	検知履歴ページに移動します。
⑨	ユーティリティプルダウン	掲示板へのリンク、マニュアルの表示、ログアウト機能を表示します。

管理者トップページ

図 5　管理者トップページ

管理者トップページです。

農地の位置、検知された害獣の位置、獣害被害が予想される農地の位置が表示されます。

番号	名称	備考
①	位置検索フォーム	マップの表示中心を変更します。
②	アイコン表示 ON/OFF チェックボックス	マップ上の農地アイコンの表示 ON/OFF を切り替えます。
③	マニュアル表示ボタン	マニュアルを表示します。
④	掲示板移動ボタン	獣害対策掲示板へ移動します。

利用者管理画面

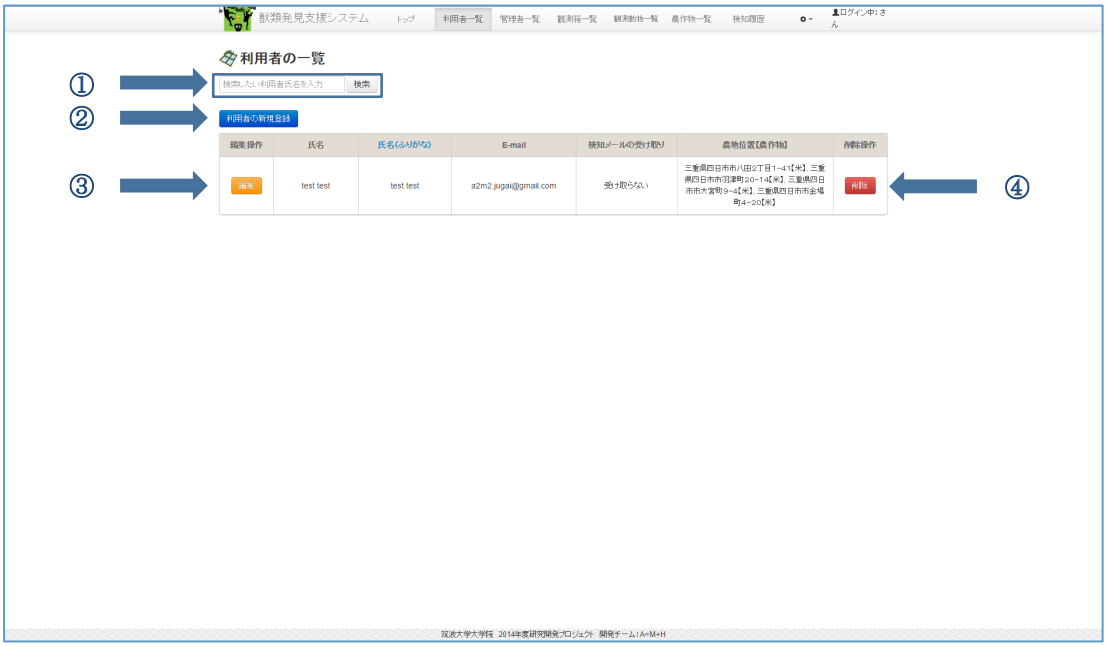


図 6　利用者一覧画面

登録済みの利用者一覧を確認することができます。

また、利用者情報の新規登録、編集、削除を行うことができます。

番号	名称	備考
①	検索フォーム	氏名での検索を行います。
②	利用者新規登録ボタン	利用者情報登録画面に移動します。
③	編集ボタン	利用者情報登録画面に移動し利用者情報を変更します。
④	削除ボタン	利用者情報を削除します。

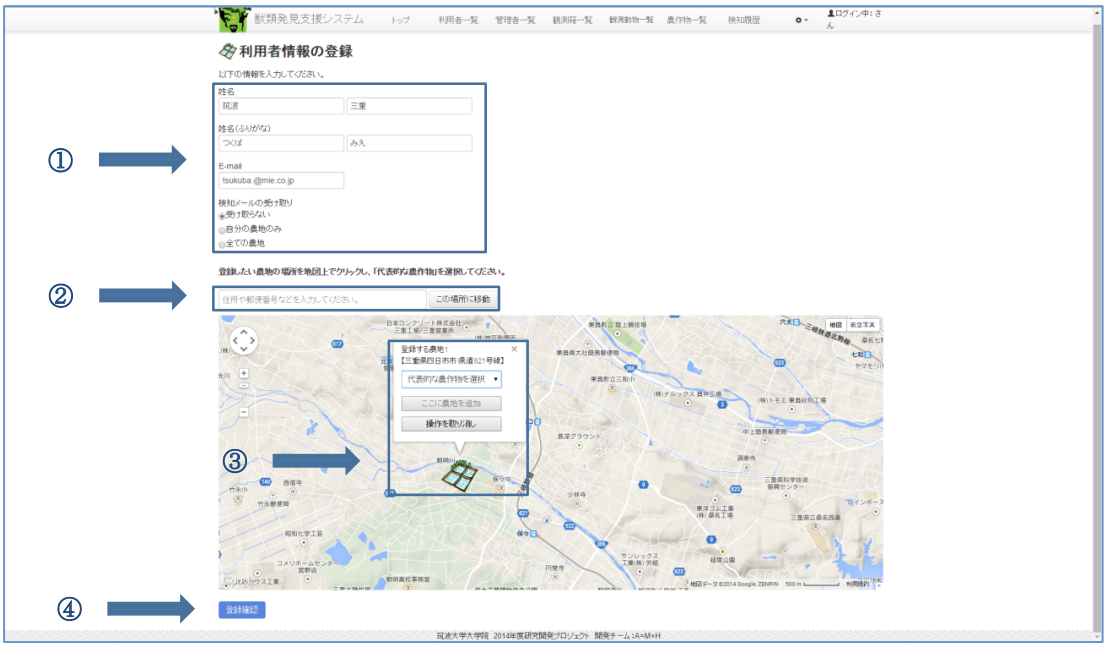


図 7　利用者情報登録画面

利用者情報の登録、変更を行うことができます。

番号	名称	備考
①	利用者情報入力欄	氏名、ふりがな、E メールアドレス、検知メールの受け取り設定を入力します。
②	位置検索フォーム	マップの表示中心を変更します。
③	農地位置入力アイコン	MAP 上をクリックし、農作物を選択することで利用者の持つ農地位置を登録します。
④	登録確認ボタン	登録利用者情報確認画面に移動します。

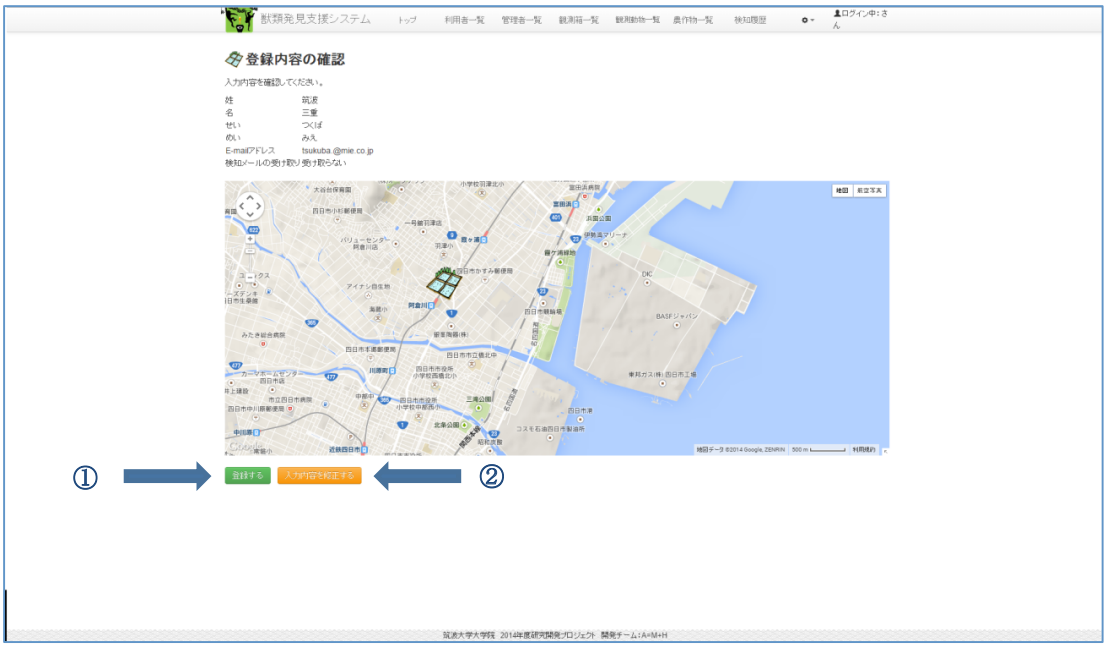


図 8　登録利用者情報確認画面

利用者情報登録画面で入力した内容を保存前に確認します。

番号	名称	備考
①	登録ボタン	利用者情報を登録し、利用者一覧画面に移動します。
②	入力内容修正ボタン	利用者情報登録画面に戻ります。

管理者管理画面

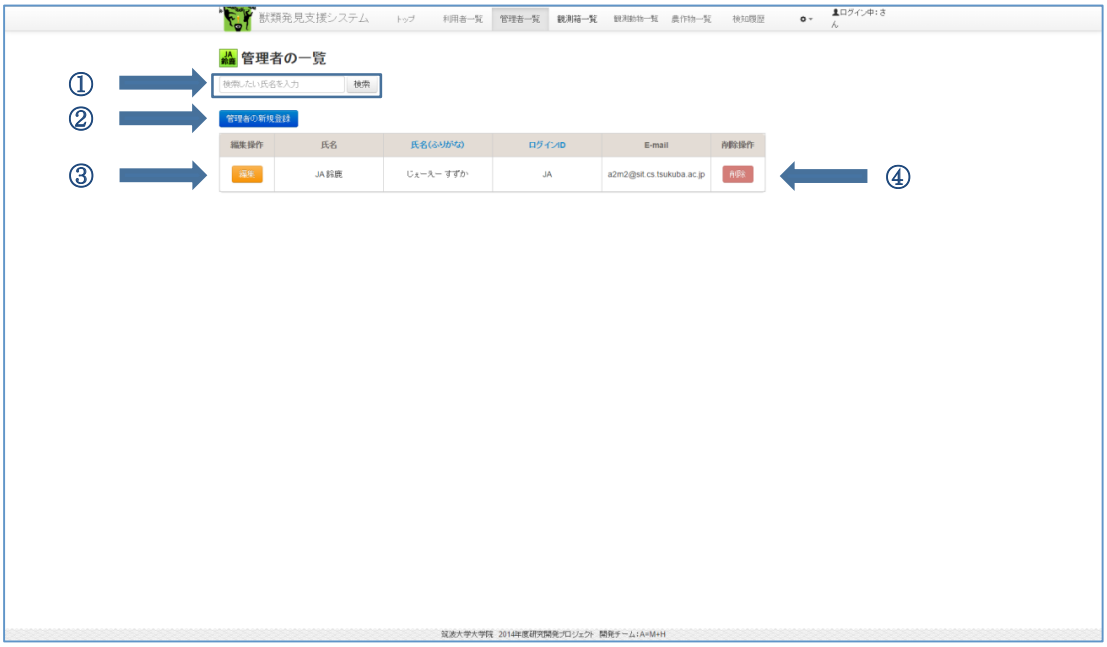


図 9　管理者一覧画面

登録済みの管理者一覧を確認することができます。

また、管理者情報の新規登録、編集、削除を行うことができます。

番号	名称	備考
①	検索フォーム	氏名での検索を行います。
②	管理者新規登録ボタン	管理者情報登録画面に移動します。
③	編集ボタン	管理者情報登録画面に移動し管理者情報を変更します。
④	削除ボタン	管理者情報を削除します。

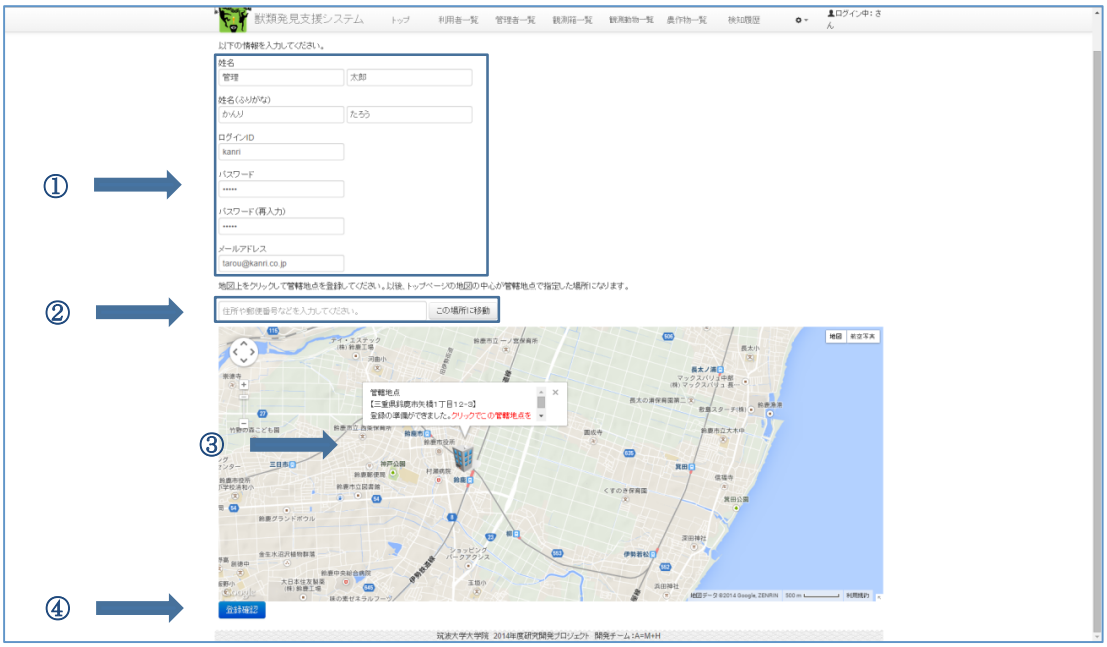


図 10　管理者情報登録画面

管理者情報の登録、変更を行うことができます。

番号	名称	備考
①	管理者情報入力欄	氏名、ふりがな、ログイン ID、パスワード、E メールアドレスを入力します。
②	位置検索フォーム	マップの表示中心を変更します。
③	管轄地域入力アイコン	MAP 上をクリックし、地点を選択することで毎回表示される中心地点を登録します。
④	登録確認ボタン	登録利用者情報確認画面に移動します。

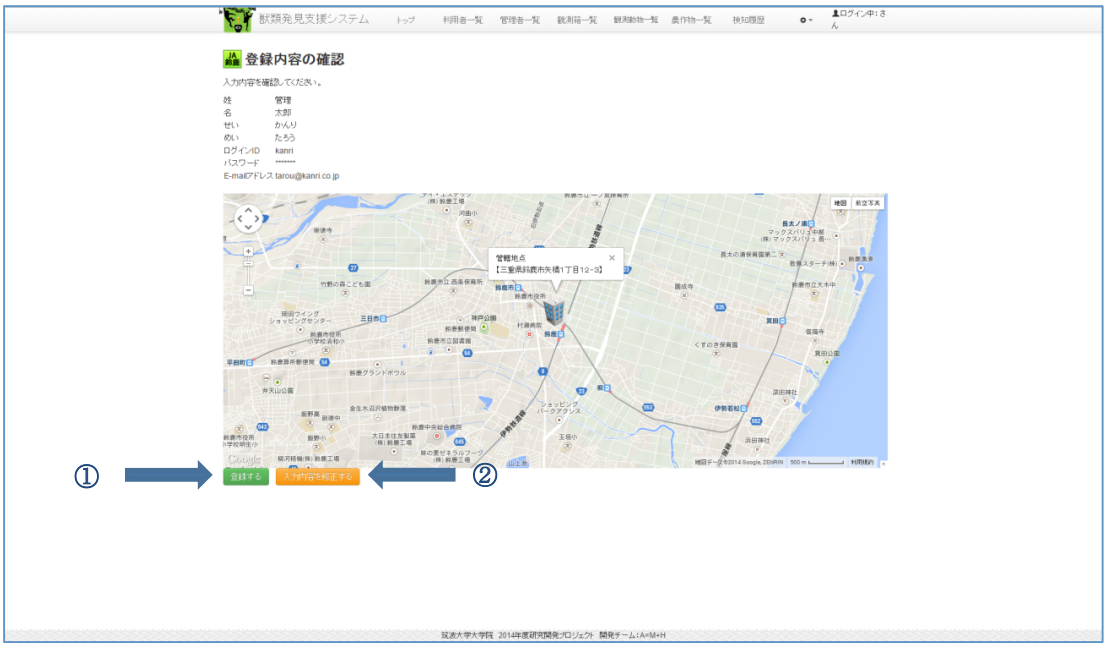


図 11　登録利用者情報確認画面

管理者情報登録画面で入力した内容を保存前に確認します。

番号	名称	備考
①	登録ボタン	管理者情報を登録し、利用者一覧画面に移動します。
②	入力内容修正ボタン	管理者情報登録画面に戻ります。

観測箱管理画面

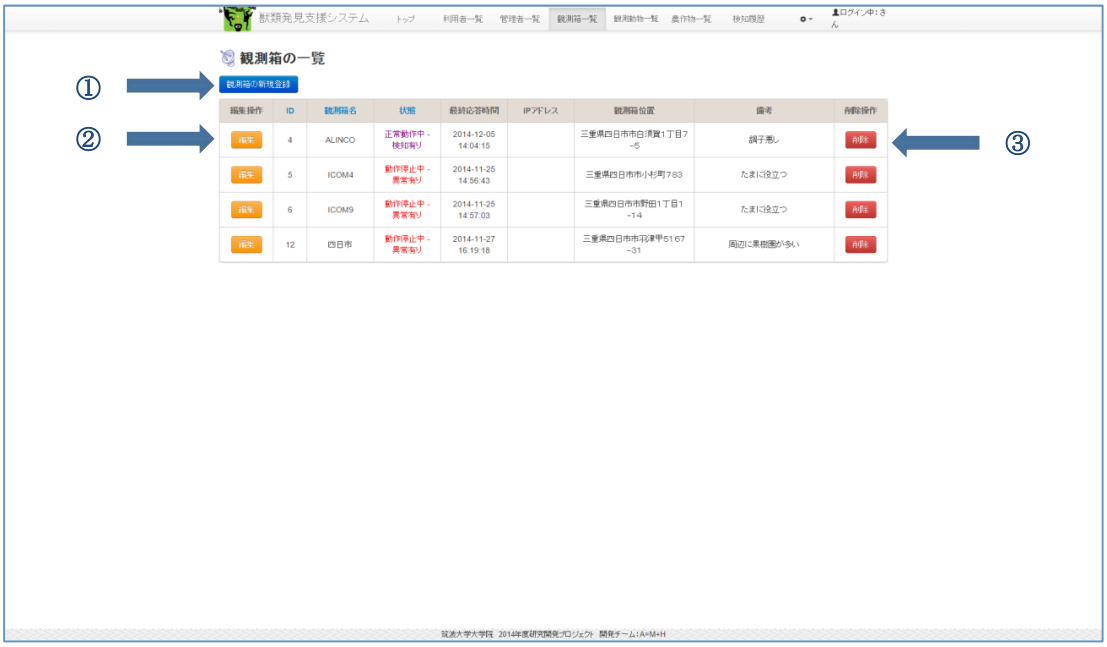


図 12　観測箱一覧画面

登録済みの観測箱一覧を確認することができます。

また、観測箱情報の新規登録、編集、削除を行うことができます。

番号	名称	備考
①	観測箱新規登録ボタン	観測箱情報登録画面に移動します。
②	編集ボタン	観測箱情報登録画面に移動し管理者情報を変更します。
③	削除ボタン	観測箱情報を削除します。

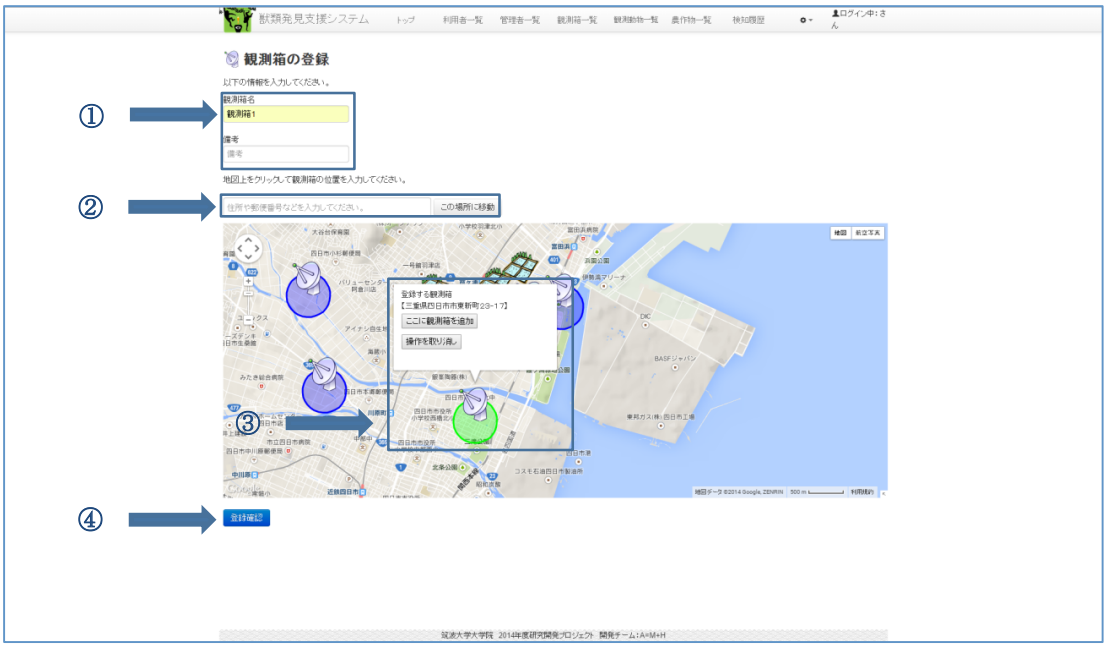


図 13　観測箱登録画面

観測箱情報の登録、変更を行うことができます。

番号	名称	備考
①	観測箱情報入力欄	観測箱の名前、備考を入力します。
②	位置検索フォーム	マップの表示中心を変更します。
③	観測箱位置入力アイコン	MAP 上をクリックすることで登録する観測箱の位置を指定します。
④	登録確認ボタン	登録観測箱情報確認画面に移動します。

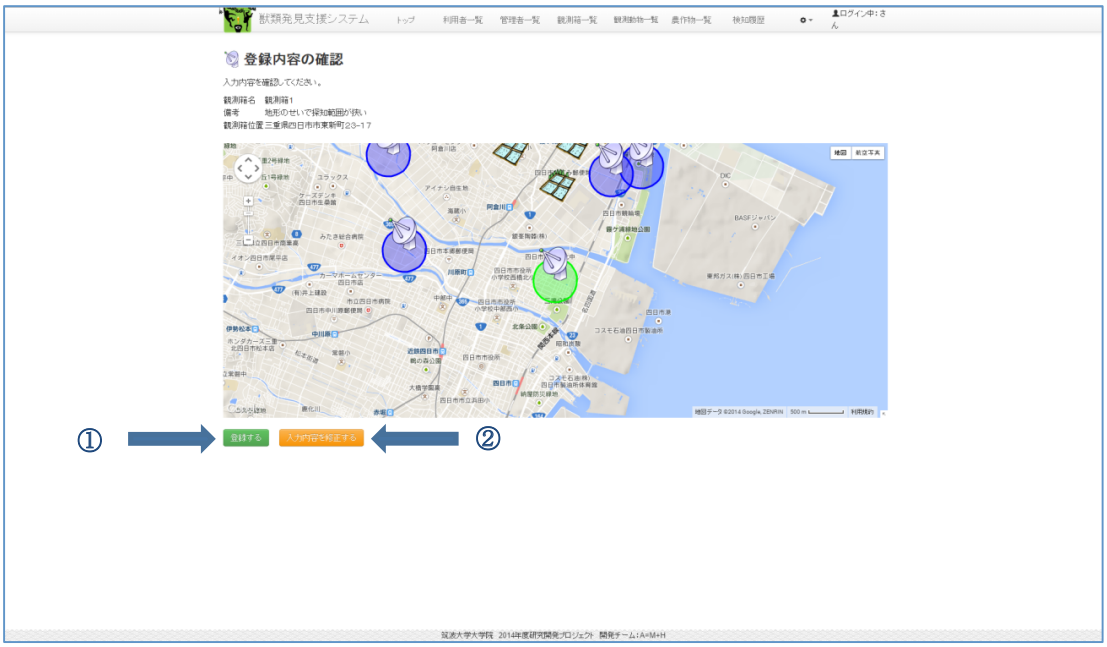


図 14　登録観測箱情報確認画面

利用者情報登録画面で入力した内容を保存前に確認します。

番号	名称	備考
①	登録ボタン	観測箱情報を登録し、観測箱一覧画面に移動します。
②	入力内容修正ボタン	観測箱情報登録画面に戻ります。

観測動物管理画面

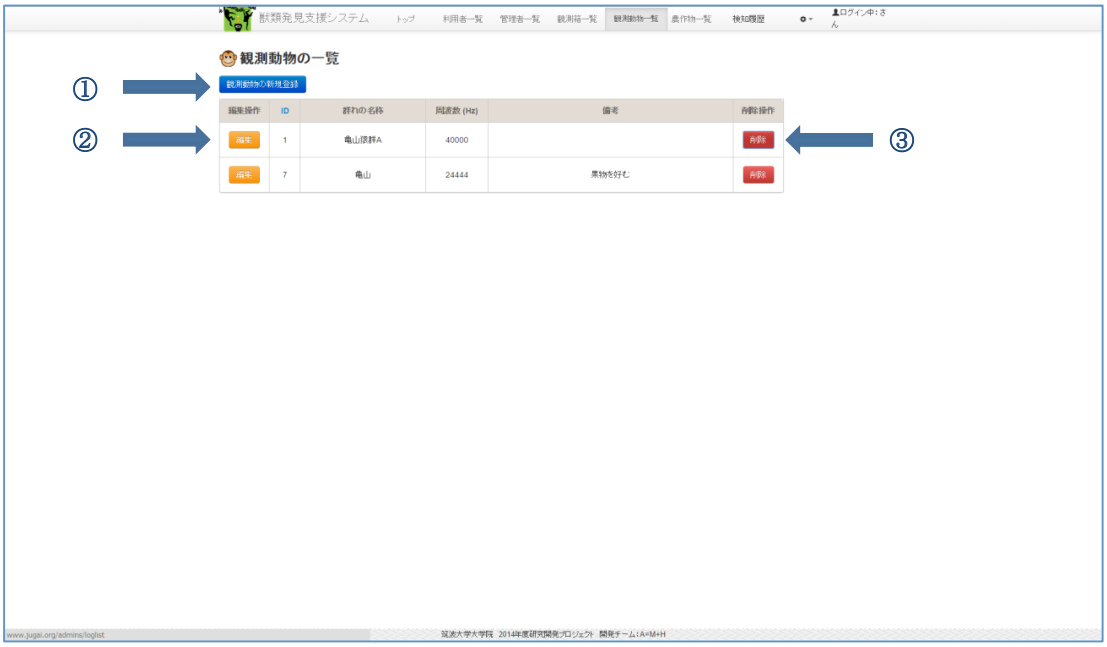


図 15　観測動物一覧画面

登録済みの観測動物一覧を確認することができます。

また、観測動物情報の新規登録、編集、削除を行うことができます。

番号	名称	備考
①	観測動物新規登録ボタン	観測動物情報登録画面に移動します。
②	編集ボタン	観測動物情報登録画面に移動し観測動物情報を変更します。
③	削除ボタン	観測動物情報を削除します。

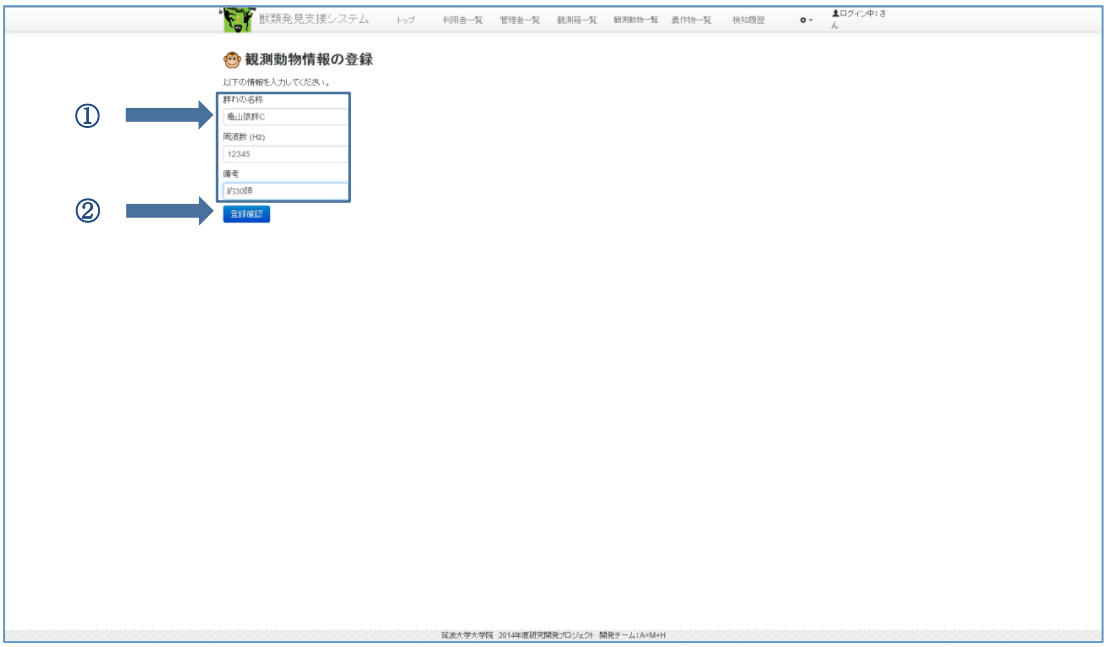


図 16　観測動物登録画面

観測動物情報の登録、変更を行うことができます。

番号	名称	備考
①	観測動物情報入力欄	登録する観測動物の群れの名前、発信機の周波数、備考を入力します。
②	登録確認ボタン	観測動物情報の確認画面に移動します。

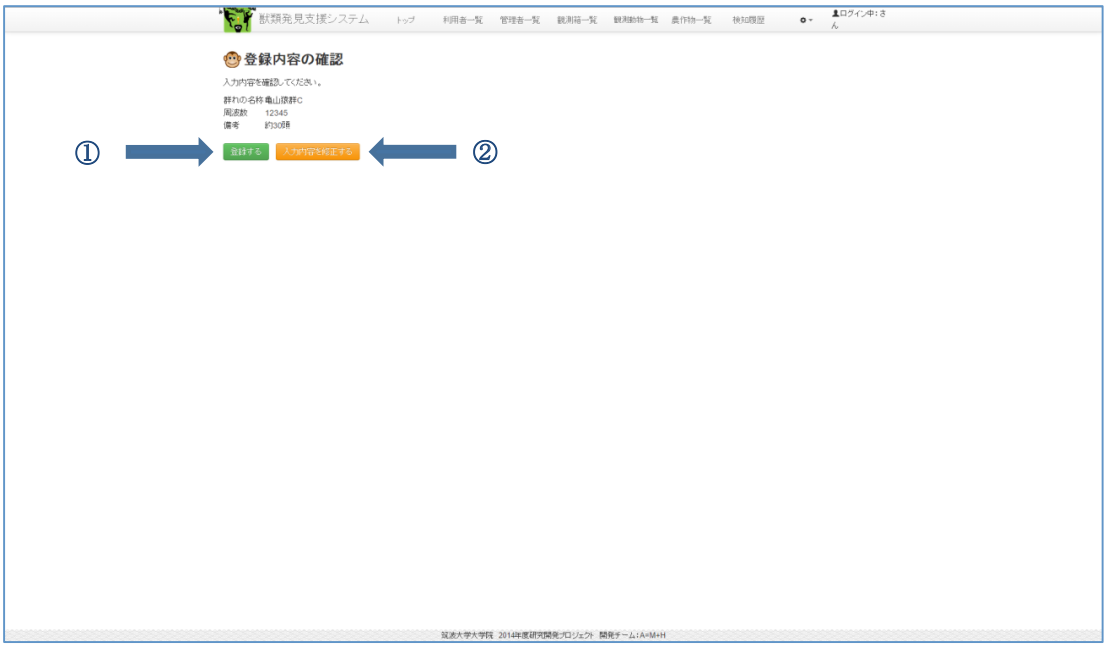


図 17　登録観測動物情報確認画面

観測動物情報登録画面で入力した内容を保存前に確認します。

番号	名称	備考
①	登録ボタン	観測動物情報を登録し、観測動物一覧画面に移動します。
②	入力内容修正ボタン	観測動物情報の登録画面に戻ります。

農作物管理画面

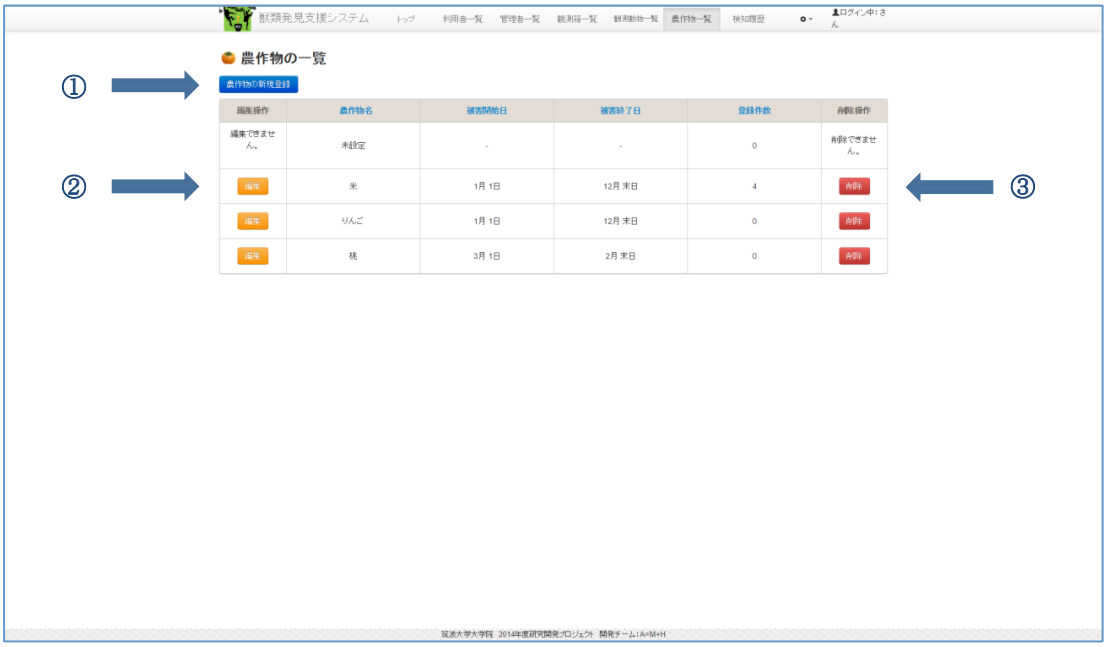


図 18　農作物一覧画面

登録済みの農作物一覧を確認することができます。

また、農作物情報の新規登録、編集、削除を行うことができます。

番号	名称	備考
①	農作物新規登録ボタン	農作物情報登録画面に移動します。
②	編集ボタン	農作物情報登録画面に移動し農作物情報を変更します。
③	削除ボタン	農作物情報を削除します。

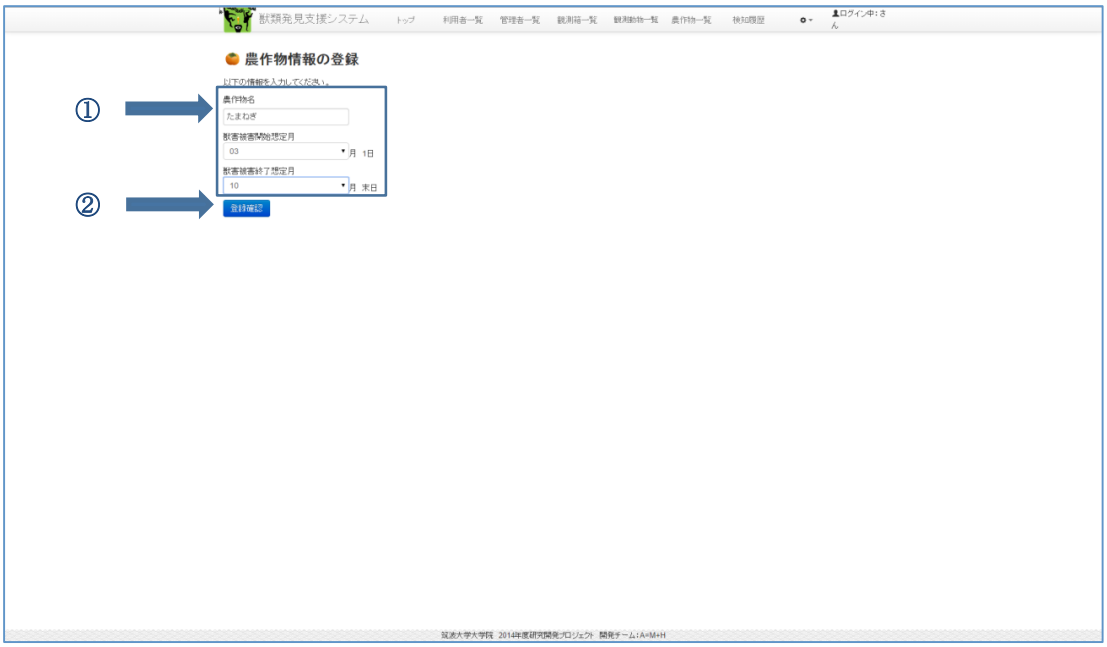


図 19　農作物情報登録画面

農作物情報の登録、変更を行うことができます。

番号	名称	備考
①	農作物情報入力欄	農作物名、被害開始想定月、被害終了想定月を入力します。
②	登録確認ボタン	登録農作物情報の確認画面に移動します。

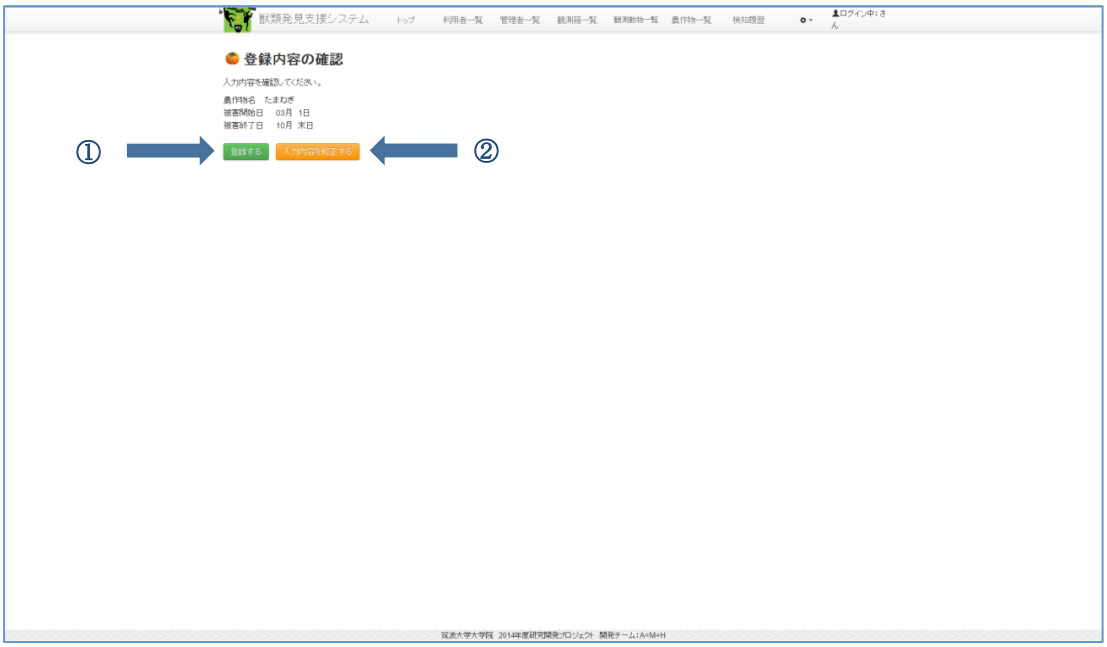


図 20　登録農作物情報確認画面

農作物情報登録画面で入力した内容を保存前に確認します。

番号	名称	備考
①	登録ボタン	農作物情報を登録し、農作物一覧画面に移動します。
②	入力内容修正ボタン	農作物情報の登録画面に戻ります。

検知履歴

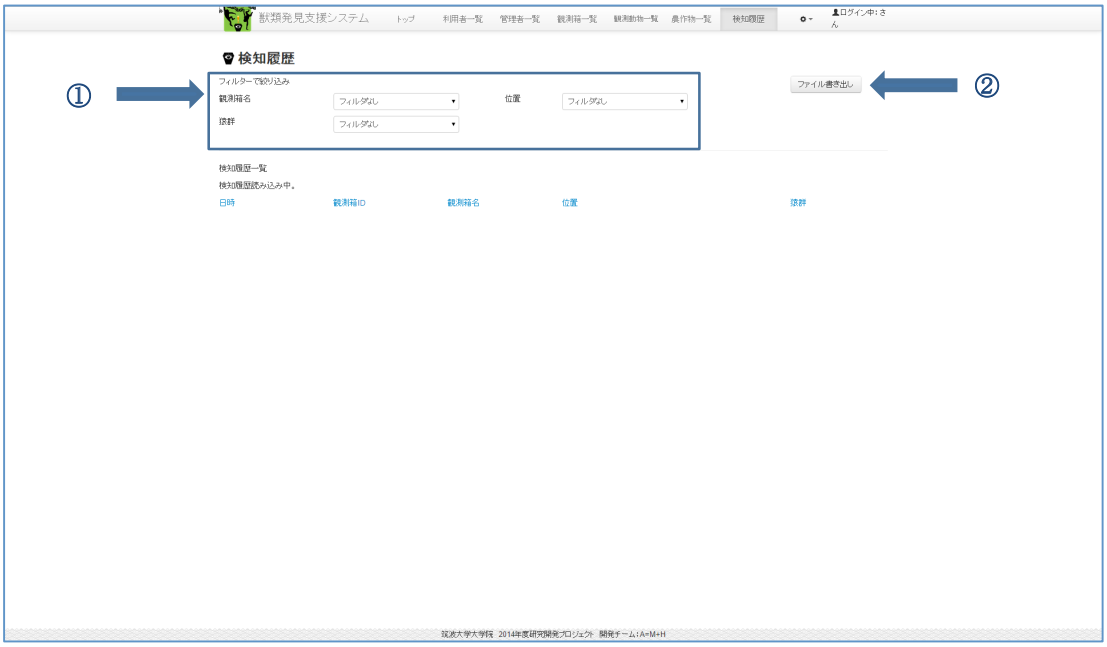


図 21　検知履歴画面

観測動物が検知された履歴を閲覧できます。

番号	名称	備考
①	フィルター	履歴情報を絞り込みます。
②	ファイル書き出しボタン	履歴情報をファイルとしてダウンロードします。

掲示板

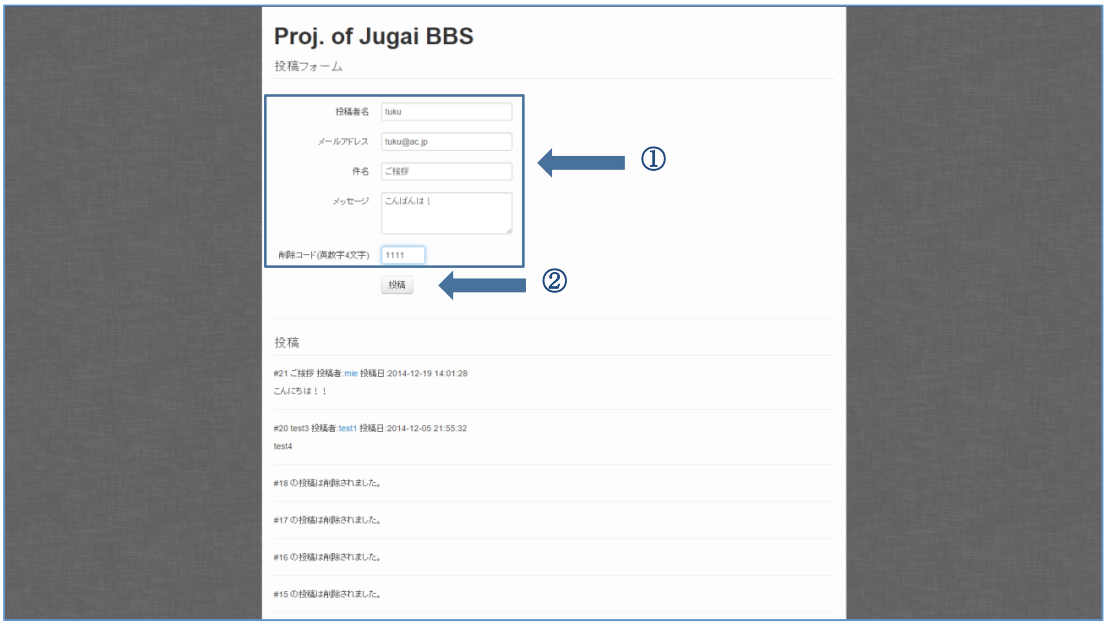


図 22　掲示板上部

図 23　掲示板下部

意見交換を行う掲示板です。ほかの利用者とネット上で話し合うことができます。

番号	名称	備考
①	入力欄	掲示板に投稿する投稿者名、E メールアドレス、記事の件名、メッセージ、削除する際に使う削除コードを入力します。
②	投稿ボタン	入力欄に入力した情報を投稿します。 投稿メッセージ確認画面に移動します。
③	削除フォーム	記事の投稿 No と削除コードを入力し、削除ボタンを押すことで投稿した記事を削除することができます。

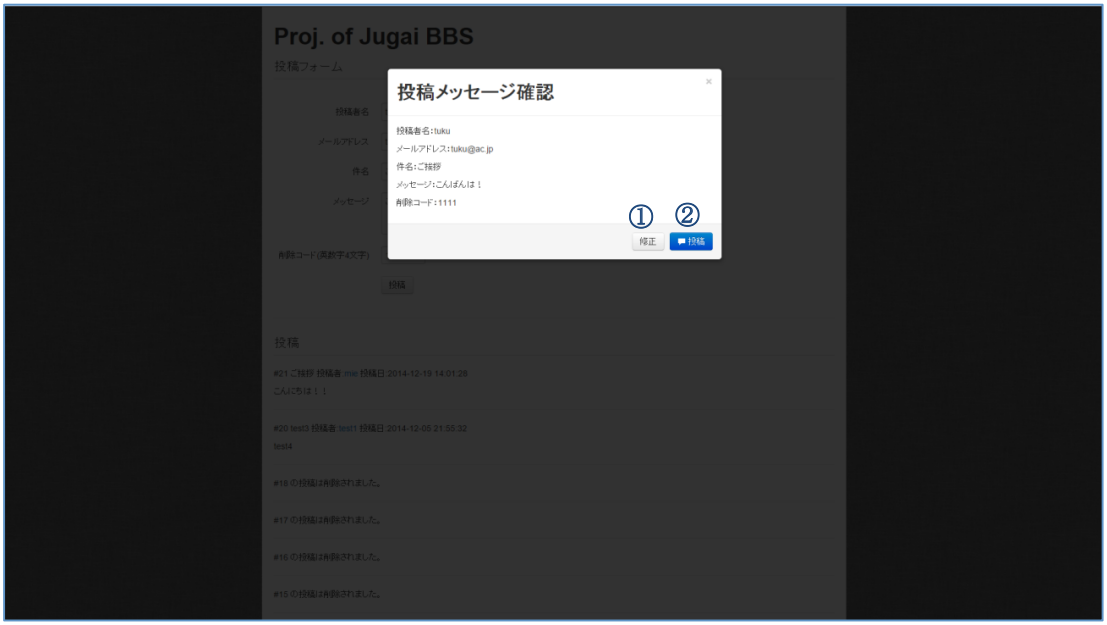


図 24　投稿確認画面

投稿するメッセージの内容を投稿する前に確認します。

番号	名称	備考
①	修正ボタン	投稿内容を修正します。
②	投稿ボタン	内容を確認し投稿します。

用語

利用者

農地を持っている人のことです。

検知メールの受け取り

システムに登録された農地で獣害が発生する確率が高いと判断した場合にメールを送信します。

- 受け取らない…検知メールを受信しません。
- 自分の農地のみ…自分の農地のみ受信します。
- 全ての農地…全国のメールを受信します（非常に多数のメールが来るため停止しています）

管理者

各種情報の閲覧、登録、編集、削除ができます。主に JA の担当者の方を想定しています。

観測箱

保護対象の地域に配置された観測動物から発信される電波の受信を行う観測機器のことです。

最終応答時間

観測箱が最後に正常に稼働したことを確認できた時刻のことです。

観測動物

発信機を付けた害獣のことです。

周波数

観測動物の発信機が発する電波の周波数のことです。

農作物

獣害被害開始想定月、終了想定月

農作物が獣害被害にあうと想定される期間を月単位で指定します。

項目番号	対象画面名	対象機能名	レビュー希望観点	備考	ご回答
1	ログイン画面	（全般）	（機能やレイアウトについて、何かご意見があればご記載ください）		特になし
2	管理者トップ画面	（全般）	（機能やレイアウトについて、何かご意見があればご記載ください）		特になし
3		地図	表示されている内容（観測箱、農地の位置、動作状態、結びつきなど）に不足はないか。操作方法や画面の説明に過不足はないか。	現在、操作方法はマニュアルに記載しています。	特になし
4			縮尺の範囲は適切であるか。		適切
5	利用者一覧画面	（全般）	（機能やレイアウトについて、何かご意見があればご記載ください）		特になし
6		利用者	登録／表示する利用者の情報に過不足はないか。	農地の種類（水田、果樹園、……など）も設定できるようにするか検討中です。	特になし
7			メールの送付の対象者の選別は適切か。	「自分の農地のみ」、「全ての検知情報」という設定のことです。	適切
8		検索	検索機能、並び替え機能に過不足はないか。		特になし
9	利用者登録・編集画面	（全般）	（機能やレイアウトについて、何かご意見があればご記載ください）		特になし
10		地図	利用者の農地位置を適切に、不自由なく登録できるか。もし改善点があれば合わせてご記載ください。		特になし
11	利用者登録確認画面	（全般）	（機能やレイアウトについて、何かご意見があればご記載ください）		特になし
12	管理者一覧画面	（全般）	（機能やレイアウトについて、何かご意見があればご記載ください）		特になし
13		管理者	登録／表示する管理者の情報に過不足はないか。		特になし
14	管理者新規登録画面	（全般）	（機能やレイアウトについて、何かご意見があればご記載ください）		特になし
15	管理者編集画面	（全般）	（機能やレイアウトについて、何かご意見があればご記載ください）		特になし
16	管理者編集確認画面	（全般）	（機能やレイアウトについて、何かご意見があればご記載ください）		特になし
17	観測箱一覧画面	（全般）	（機能やレイアウトについて、何かご意見があればご記載ください）		特になし
18		観測箱	登録／表示する観測箱の情報に過不足はないか。		特になし
19	観測箱編集画面	（全般）	（機能やレイアウトについて、何かご意見があればご記載ください）		特になし
20		地図	観測箱を適切に、不自由なく登録できるか。もし改善点があれば合わせてご記載ください。		特になし
21	観測箱編集確認画面	（全般）	（機能やレイアウトについて、何かご意見があればご記載ください）		特になし
22	ログ画面	（全般）	（機能やレイアウトについて、何かご意見があればご記載ください）		特になし
24	その他	－	その他、全体として何かご意見、ご要望、改善点などがあればご記載ください。		利用者同士と管理者が意見交換できる掲示板を付けるとよいと思います。

項目番号	追加する機能の候補	機能の説明	JA側 実装希望順位	なにかコメントがあればご記載ください
1	猿の群れの登録ページを追加する。農地に「農作物」情報を追加し、猿の移動予測を行う。 ※この機能はほぼ必須であるため、レビュー結果をご返答いただくまでの間にも開発を進めていきます。	猿の群れに名前を付けて登録することができるページを追加します。これによって複数の発信器周波数に対応できるようにします。また、発展的要素として、「猿の現在位置を推測して地図上に表示する」といった機能の追加を予定しています。	-	
2	猿の位置検出の精度を向上させる。	現時点では、猿は観測箱の中心からおおよそ500m以内にいることしか分かりません。これを発展させ、「だいたいこの位置にいるだろう」と言える程度にまで、検出精度を向上させます。	4番目	猿は、通り道を通るとき以外は、短時間でそれほど遠くに移動することはないそうです。もちろん検出精度は良いに越したことはありませんが、他3項目の希望と比較すると、半径500m以内での認識でもよいかと思います。
3	猿が検知された場合に、ランプを灯したり、スピーカーで警告音を発するなど、即時的な対応を行う。	現在のシステムでは、メールを受信した人が任意で猿を追い払う仕様になっています。これを改善し、他の人も追い払えるようにするため、「そこに猿がいるんだな！」ということが遠目にでも分かるような仕組みを構築します。	2番目	夜はランプが灯る、昼は警告音が鳴るというのが理想ですが、両方付けるのが難しければ、警告音の方がよいかと思います。ただし、音量を調整する等、近隣住民に対する配慮は必要かと思います。
4	猿以外の動物にも対応可能にする。	観測箱にカメラを取り付け、画像認識を行うことで、発信器がついていない鹿や猪といった動物も検知できるように改良します。ただし、画像認識なので誤検出が発生する可能性があります。また、高性能なカメラが必要になるという欠点があります。	1番目	鹿や猪にも対応することは、大変強い希望です。誤検出と高性能カメラのことを考えると悩むところですが、是非実現してほしいです。
5	任意の観測箱と農地を結び付ける。	現在は、農地と、その農地に最も近い観測箱とが自動で結び付く(最も近い観測箱に猿が接近した際にメールを送信する)ようになっています。この改良により農家の人やJAの人が任意で観測箱と農地の結びつきを決めることができるようになります。	5番目	
6	盗難、いたずら防止機能。		3番目	観測箱そのものを猿のいたずら、人間の盗難から守るための機能が必要かと思います。
7	(その他、何か希望する追加機能がありましたらご記載ください)	(その他、何か希望する追加機能がありましたらご記載ください)		

運用マニュアル評価アンケート	
実施日：2014年12月29日	

獣類発見支援システムの「運用マニュアル」を読んで、各質問に対して5段階から選択してで評価してください。

1:全くそう思わない
2:そう思わない
3:どちらともいえない
4:そう思う
5:非常にそう思う

1. このマニュアルを読めば、今後もシステムを運用することができる	
2. マニュアルの内容は簡単に理解することができる	
3. マニュアルに書かれている用語を理解することができる、 もしくは用語の説明がしっかりと書かれている	
4. 誤字・脱字が無く、表記や表現に一貫性がある	
5. 知りたい情報をマニュアルから素早く探し出すことができる	
6. システムに対する問い合わせ先など困った時の連絡先が記載されている	

担当者が不在で評価ができません
山洋正

ご署名：

プロジェクト評価アンケート	
実施日：2014年12月27日	
筑波大学の学生が取り組んだ本プロジェクト(獣類発見支援システムの開発)の活動について、 5段階で評価してください	

1:全くそう思わない
2:そう思わない
3:どちらともいえない
4:そう思う
5:非常にそう思う

1. 納品したシステムは抱えている問題を解決することができる。	4
2. 納品したシステムはプロジェクト発足時に定義した要件を満たしている。	5
3. 納品したドキュメント(書類)に不備はなかった。	3
4. プロジェクトの進捗具合を定期的に報告していた。	5
5. ミーティングの開催頻度は適切だった。	4
6. なにかご意見、ご要望、ご感想がありましたら、ご記載ください。	

3.2

ご署名：川出洋正

研究開発プロジェクト－顧客アンケート　　**回答日：** 2014 **年** 12 **月** 27 **日**

本プロジェクトの要件定義、開発体制、成果物などについてのアンケート調査にご協力ください。
質問はQ1-Q12まであり、所要時間は約15分です。
*必須の回答

要件定義について

Q1. 要件定義において、解決すべき課題を適切に捉えていましたか？ *

	1	2	3	4	5	
捉えていなかった	○	○	○	✓	○	捉えていた

Q2. 要件定義の満足度はいかがでしたか？ *

	1	2	3	4	5	
満足していない	○	○	○	○	✓	満足している

もしよろしければQ1-Q2の回答理由をご記載ください。

Q1について. 実際に~~対~~獣害対策的なものがほしかった

要件定義について、改善すべき点はありましたか？　よろしければご記載ください。

絵で見てもらい。
多くの人に

開発体制について

Q3. ミーティングの開催頻度、開催時間は適切でしたか？ *

	1	2	3	4	5	
適切ではなかった	○	○	✓	○	○	適切であった

Q4. 開発の進捗が適切に伝わっていましたか？ *

	1	2	3	4	5	
伝わっていなかった	○	✓	○	○	○	伝わっていた

Q5. 開発体制の満足度はいかがでしたか？ *

	1	2	3	4	5	
満足していない	○	○	○	✓	○	満足している

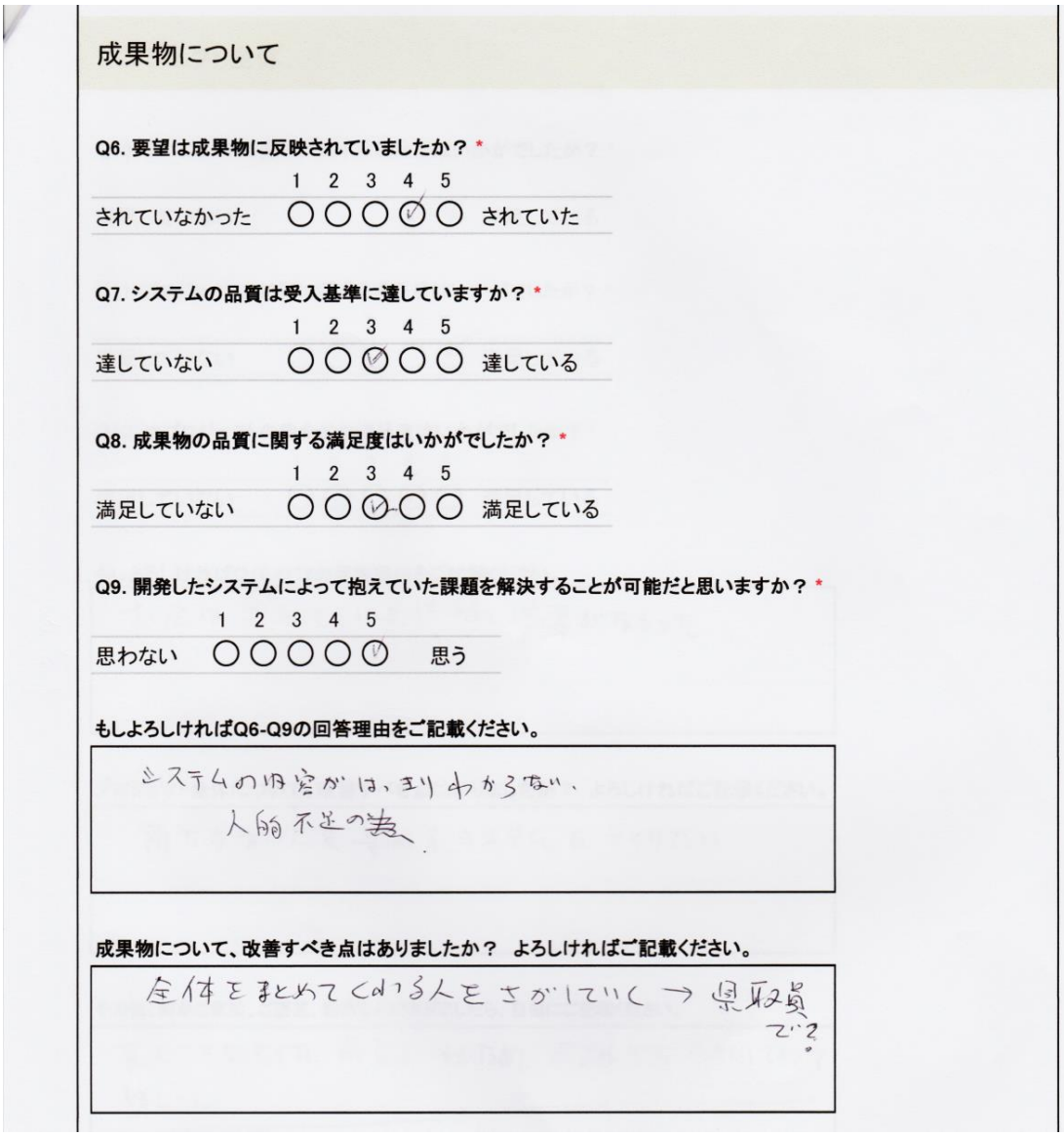
もしよろしければQ3-Q5の回答理由をご記載ください。

Q3
Q4

パソコン利用がうまくできなかったため

担当者不在の為。

開発体制について、改善すべき点はありましたか？ よろしければご記載ください。



成果物について

Q6. 要望は成果物に反映されていましたか？ *

	1	2	3	4	5	
されていなかった	○	○	○	✓	○	されていた

Q7. システムの品質は受入基準に達していますか？ *

	1	2	3	4	5	
達していない	○	○	✓	○	○	達している

Q8. 成果物の品質に関する満足度はいかがでしたか？ *

	1	2	3	4	5	
満足していない	○	○	✓	○	○	満足している

Q9. 開発したシステムによって抱えていた課題を解決することが可能だと思いますか？ *

	1	2	3	4	5	
思わない	○	○	○	○	✓	思う

もしよろしければQ6-Q9の回答理由をご記載ください。

システムの内容がはっきりわからない。
人的不足の為。

成果物について、改善すべき点はありましたか？ よろしければご記載ください。

全体をまとめてくれる人をさがしていく → 県職員で？

プロジェクト全体

Q10. プロジェクトの費用に関する満足度はいかがでしたか？ *

	1	2	3	4	5	
満足していない	○	○	○	○	✓	満足している

Q11. プロジェクトの納期に関する満足度はいかがでしたか？ *

	1	2	3	4	5	
満足していない	○	○	○	○	✓	満足している

Q12. 本プロジェクトの総合的な満足度はいかがでしたか？ *

	1	2	3	4	5	
満足していない	○	○	○	✓	○	満足している

もしよろしければQ10-Q12の回答理由をご記載ください。

お金は農協とは具体的に必要がなかった

プロジェクト全体について、改善すべき点はありましたか？　よろしければご記載ください。

新たなお金を集める、システムをつくりたい

その他、何かご意見、ご要望、感想などがありましたら、自由にご記載ください。

長くきちっとした品質にするため、引継ぎをうまくしていってほしい。

農家がすぐに効果がわかるものを早く開発してもらうとたすかる。

署名： 川出洋正

アンケートは以上です。お忙しい中でのご協力、誠にありがとうございました。

初版：16 December, 2014

評価実験 最終報告書

目次

1. 基本動作 …… 1

2. 検知性能 …… 5

3. 予行実験 …… 7

4. 現地実験 …… 10

1. 基本動作

実施日：28 May, 2014

参加者：天本、岡部、浜野、光岡、森

目的

１．発信器、レシーバー、および、試作したアンテナの動作確認

２．試作したアンテナによる発信器の受信範囲計測

概要

筑波大学かえで通りの見通しのよい直線道路を利用し、遮蔽物がない状態において、試作したアンテナの最大受信範囲を計測する。

計測方法

１．２チームに別れ、１チームがアンテナを持ち、レシーバーより聞こえる音を確認する。他の１チームは、発信器を持ち、アンテナから離れるように移動する。

２．レシーバーより音が確認できなくなった時点で、GPSを使用し、それぞれのチームの位置情報を記録する。

３．記録した位置情報をGoogle Inc.のGoogleマップ[1]に入力し、２点間の距離を求める。

使用機器

・レシーバー

　　ALINCO　DJ-X11

・アンテナ

　　試作したダイポールアンテナ

・テレメトリー発信器

　　型番不明（現在鈴鹿市で使用しているもの）　周波数：149.243MHz

[1] Googleマップ

- https://www.google.co.jp/maps/

結果

保存した２つの位置情報より２点間距離を求めた。その結果を図1に示す。図1より、半径約100m程度が最大受信範囲であることがわかった。

図1　２点間距離計算結果

1. 初期動作（再実験）

実施日：18 June, 2014

参加者：天本、岡部、浜野、光岡、森

目的

１．購入したアンテナによる発信器の受信範囲計測

概要

筑波大学かえで通りの見通しのよい直線道路を利用し、遮蔽物がない状態において、購入したアンテナの最大受信範囲を計測する。

計測方法

１．２チームに別れ、１チームがアンテナを持ち、レシーバーより聞こえる音を確認する。他の１チームは、発信器を持ち、アンテナから離れるように移動する。

２．レシーバーより音が確認できなくなった時点で、GPSを使用し、それぞれのチームの位置情報を記録する。

３．記録した位置情報をGoogle Inc.のGoogleマップに入力し、２点間の距離を求める。

使用機器

・レシーバー

ICOM IC-R6

・アンテナ

NATEC NY144Z

・テレメトリー発信器

型番不明（現在鈴鹿市で使用しているもの）　周波数：149.243MHz

結果

保存した２つの位置情報より２点間距離を求めた。その結果を図1に示す。図2より、半径約600m程度が最大受信範囲であることがわかった。

図2　２点間距離計算結果

2. 検知性能

実施日：11 July, 2014

参加者：天本、岡部、浜野、光岡、森

目的

１．開発した観測箱の検知範囲、および、精度の計測

概要

筑波大学かえで通りの見通しのよい直線道路を利用し、遮蔽物がない状態において、開発した観測箱の検知範囲、および、精度を計測する。

計測方法

1. 観測箱をある定点に設置し、その地点から発信器を持ち、観測箱から離れるように移動する。
2. 50m毎に10分移動を停止し、精度を計測する。
3. 検知の精度が0%となった地点より観測箱からさらに50m離れた地点まで上記を繰り返す。
4. 同時に人による受信器を耳で聞くことによる検知を行う。

使用機器

・開発した観測箱

・テレメトリー発信器

　　型番不明（現在鈴鹿市で使用しているもの）　周波数：149.243MHz

結果

実験を行った場所を図3に示す。また、この実験の結果をまとめたグラフを図4に示す。

図3　実験を行った場所

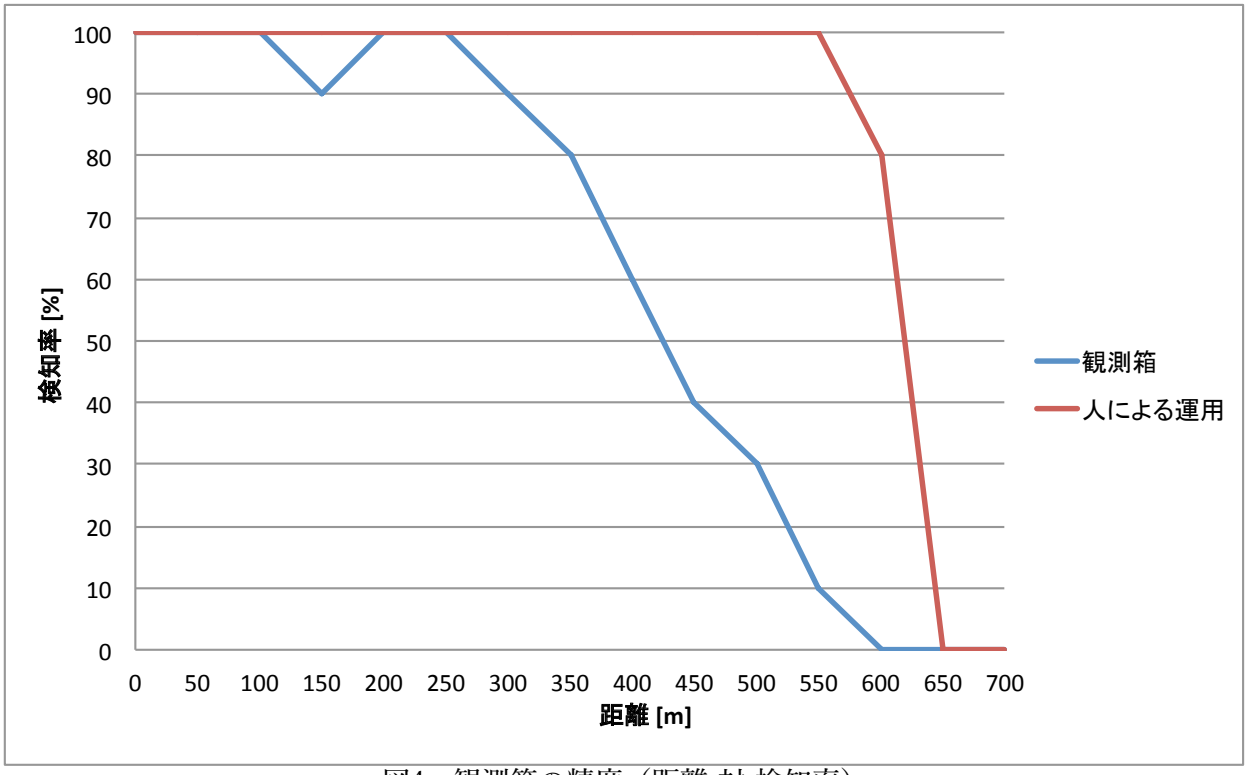


図4　観測箱の精度（距離 対 検知率）

3. 予行実験

実施日：9 November, 2014

参加者：天本、岡部、浜野、光岡、森

目的

1. 現地実験と同じ実験を行うことで当日発生しうるリスクを確認する
2. 現地での協力者に必要な時間や機器を事前に連絡するため
3. 多くの状況における観測箱の性能を評価するため

概要

筑波大学付近数地点を利用し、さまざまな状態において、開発した観測箱の検知範囲、および、精度を計測する。

計測方法

1. 観測箱をある定点に設置し、その地点から発信器を持ち、観測箱から離れるように移動する。
2. 50m毎に10分移動を停止し、精度を計測する。
3. 検知の精度が0%となった地点より観測箱からさらに50m離れた地点まで上記を繰り返す。

使用機器

・開発した観測箱

・テレメトリー発信器

　　型番不明（現在鈴鹿市で使用しているもの）　周波数：149.243MHz

結果

この評価実験の実施地点を図5に示す。また、この実験の結果についてまとめたグラフを図6に示す。

図5　実験を行った場所

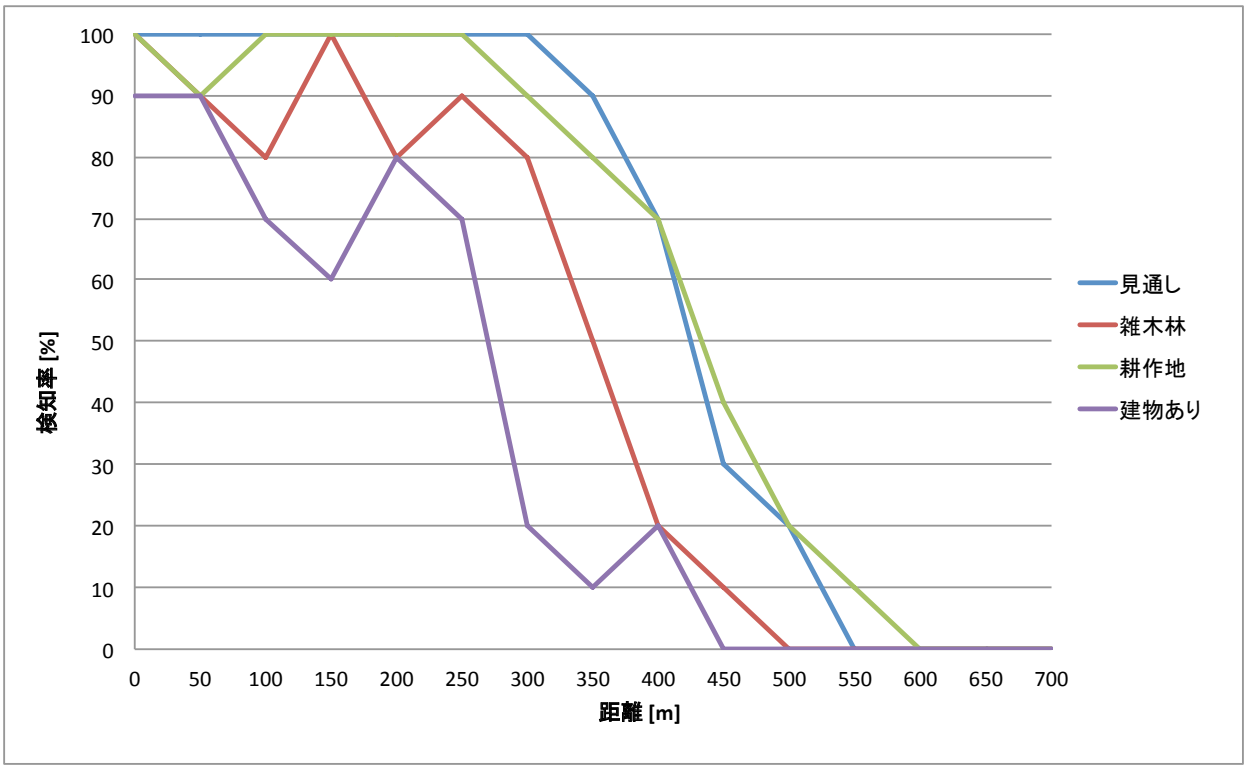


図6　観測箱の精度（距離 対 検知率）

4. 現地実験

1. 評価実験日時・概要スケジュール

2014年11月13日 木曜日

- 12:00　三重県鈴鹿市駅到着・移動開始（JA鈴鹿車両）
- 12:10　JA鈴鹿着　ヒアリング・ユーザーテスト
- 12:30　亀山市　ヒアリング
- 13:00　実験場所到着・実験準備（観測箱設置）
- 13:30　実験開始（スケジュール別途・2.5時間）
- 16:00　実験終了・撤収作業開始（観測箱回収）
- 16:30　撤収作業完了・移動開始（JA鈴鹿車両）
- 17:00　三重県鈴鹿市駅到着・解散

2. 評価実験地

実験を行った場所（三重県　鈴鹿市　（〒519-0221付近））を図7に示す。

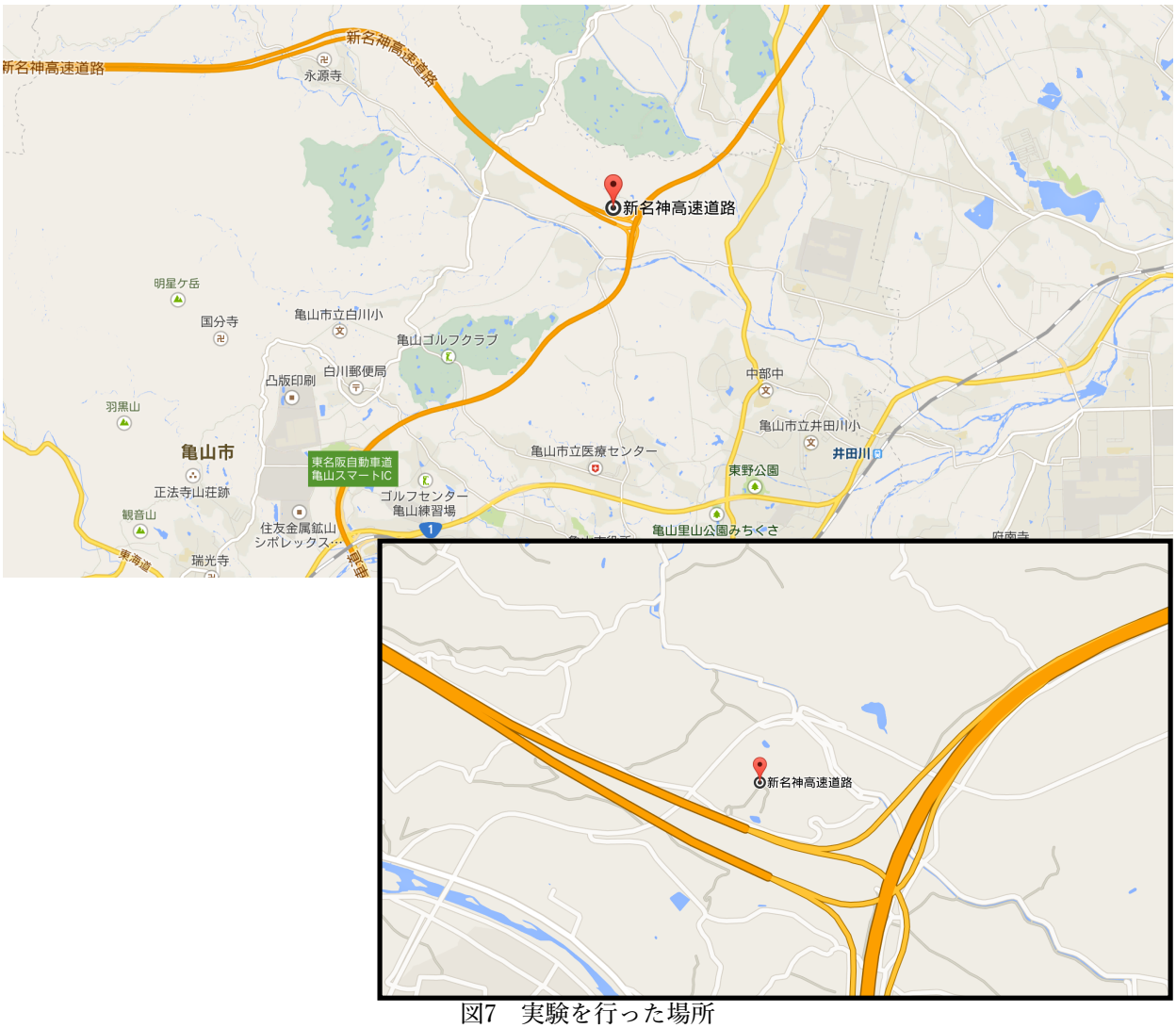


図7　実験を行った場所

3. 使用機器

・観測箱　3台

・Raspberry Pi + プログラムインストール済みSDカード

・Wi-Fiアダプター

・モバイルルーター

・モバイルバッテリー

・USBオーディオ変換器

・レシーバー

・アンテナ

・BNC同軸ケーブル

・BNC-BNC変換プラグ

・SMA-BNC変換プラグ

・USBケーブル

・オーディオケーブル

・発信器　1台

・コンピューター　1台

4. 携行品

・充電器（モバイルルーター 3台、レシーバー 3台、モバイルバッテリー 3台）

・スピーカー + USB電源

・警告灯（LED-Cube） + Mini-B USBケーブル + USB電源

5. 目的

実際に使用が想定される地点での観測箱の性能評価を行うこと。

6. 概要

実際に使用が想定される地点での観測箱の検知範囲、および、精度を計測する。

7. 計測方法

1. 観測箱をある定点に設置し、そこから発信器を持ち観測箱から離れるように移動する。
2. 50m毎に10分移動を停止し、精度を計測する。
3. 検知の精度が0%となった地点より観測箱からさらに50m離れた地点まで上記を繰り返す。
4. 複数の観測箱を検知範囲が重なるように設置し、重なった検知範囲に発信器を置き、複数機が連携した際の、検知精度を計測する。

結果

この実験の距離対精度についてまとめた結果を図8に示す。また、複数機で検知を行った際の検知回数を図9に示す。

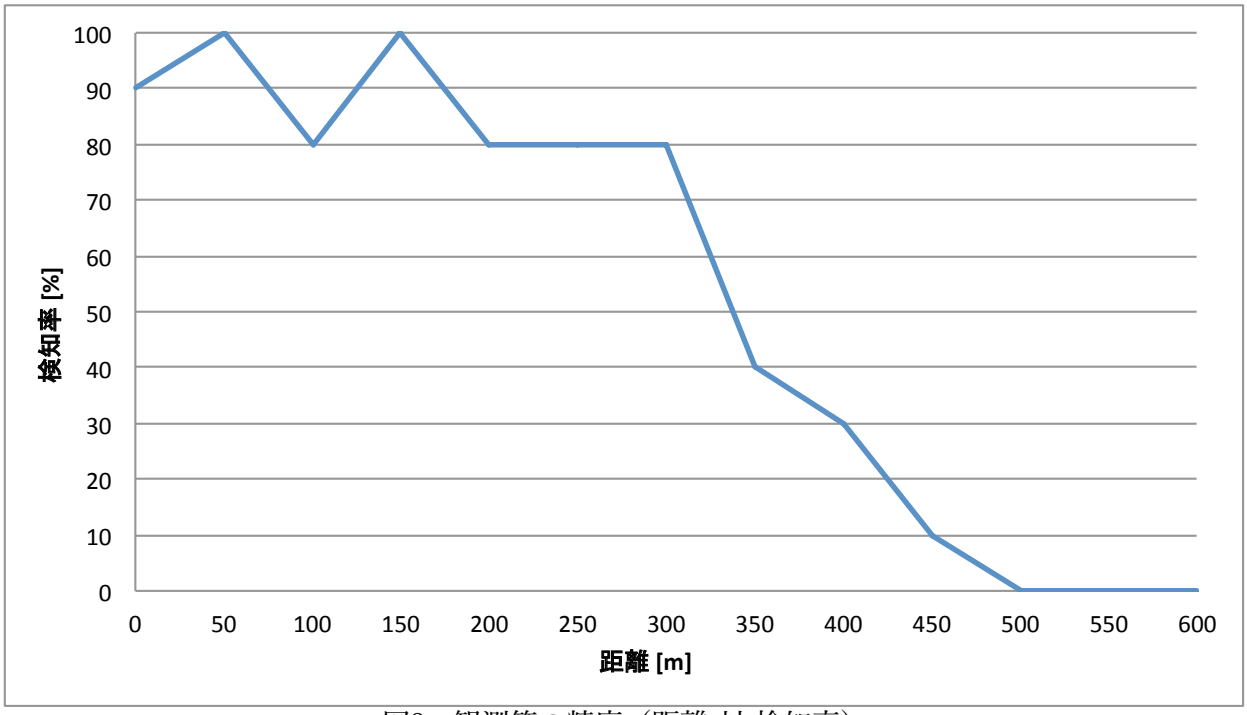


図8　観測箱の精度（距離 対 検知率）

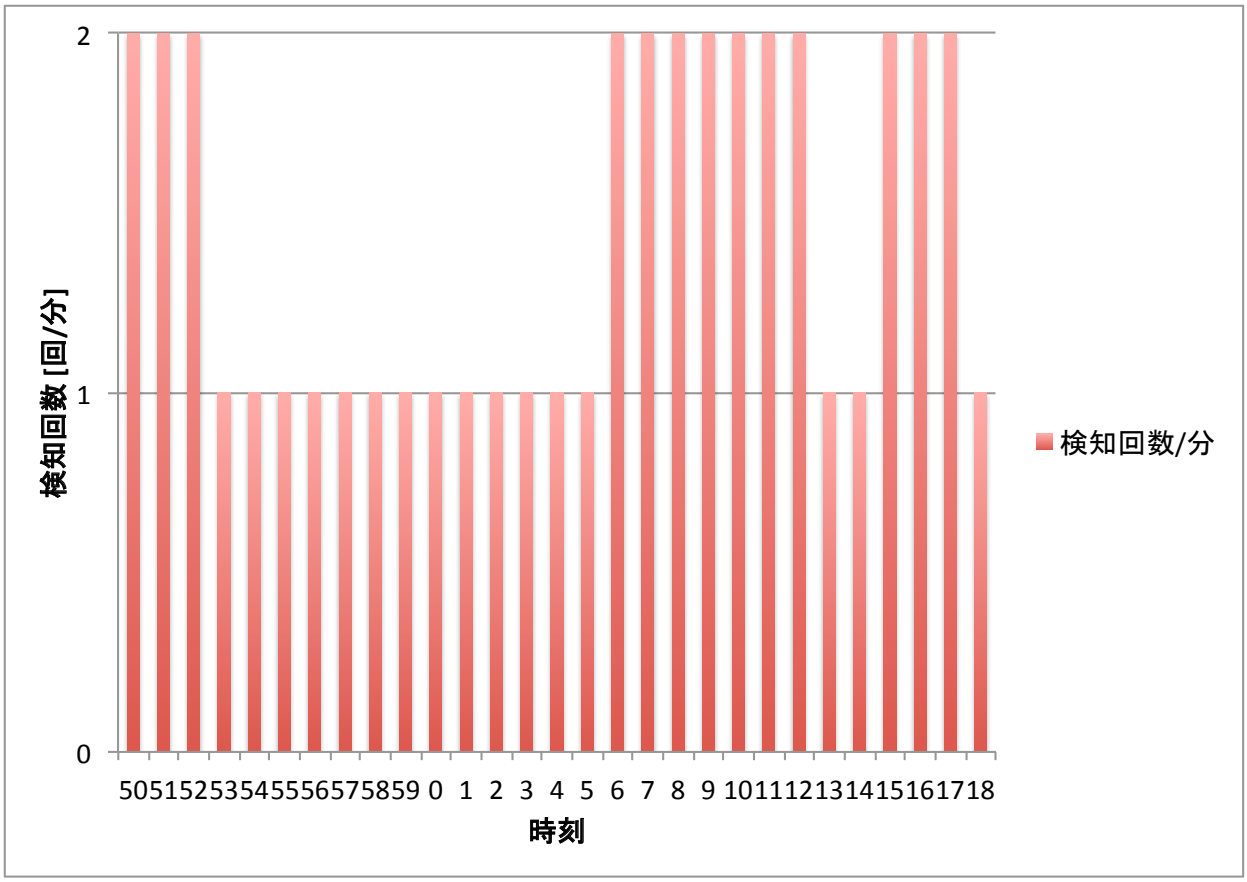


図9　時刻毎の検知回数

G. テスト　1. FV 表

画面名	目的機能	検証内容	作成テスト仕様書
トップ	・現在時刻の表示 ・サルの出現率(ヒートマップ)表示 ・観測箱、農地の表示 ・農地アイコン表示の切り替え ・マニュアルへのリンク ・1分毎のページ更新	・観測箱と農地の表示が正しいか ・サルの出現率表示(ヒートマップ)が正しいか ・農地アイコン表示の切り替えが正しいか ・マニュアルへのリンクが正しいか ・1分毎にページが更新されているか	要素別動作確認書
管理者ログイン	・管理者ページへのログイン機能	・登録されているログインIDとパスワードでログインできるか	デシジョンテーブル
メニューバー	・各一覧ページへのリンク ・ログインユーザ名がログインしている管理者のログインIDを表示する	・各一覧ページへのリンクが正しいか ・ログインユーザ名がログインしている管理者のログインIDが表示されているか	要素別動作確認書
管理者トップ	・現在時刻の表示 ・サルの出現率(ヒートマップ)表示 ・観測箱、農地の表示 ・農地アイコン表示の切り替え ・マニュアルへのリンク ・1分毎のページ更新	・観測箱と農地の表示が正しいか ・サルの出現率表示(ヒートマップ)が正しいか ・農地アイコン表示の切り替えが正しいか ・マニュアルへのリンクが正しいか ・1分毎にページが更新されているか	要素別動作確認書
利用者一覧	・利用者情報の表示 ・各利用者情報の操作リンク ・利用者の名前検索 ・特定のカラムのソート機能	・利用者情報の表示内容は正しいか ・各利用者情報の操作リンクは正しいか ・利用者の名前検索機能は正しいか ・カラムのソート機能が正しいか	要素別動作確認書
利用者登録・編集	・利用者情報を新規登録 ・利用者情報を編集 ・利用者情報登録・編集確認ページへのリンク	・各入力欄に記述されている内容は正しいか ・セッションが存在した場合、正しく表示されているか ・JavaScriptの動作は正しいか ・利用者登録・編集確認ページへのリンクは正しいか	デシジョンテーブル
利用者登録・編集確認	・利用者情報の確認 ・登録処理 ・修正処理	・利用者情報が正しく表示されているか ・登録が正しくされているか ・修正ページに表示されている内容が正しいか	要素別動作確認書
管理者一覧	・管理者情報の表示 ・各管理者情報の操作リンク ・管理者の名前検索 ・特定のカラムのソート機能	・管理者情報の表示内容は正しいか ・各管理者情報の操作リンクは正しいか ・管理者の名前検索機能は正しいか ・カラムのソート機能が正しいか	要素別動作確認書
管理者登録・編集	・管理者情報を新規登録 ・管理者情報を編集 ・管理者情報登録・編集確認ページへのリンク	・各入力欄に記述されている内容は正しいか ・セッションが存在した場合、正しく表示されているか ・利用者登録・編集確認ページへのリンクは正しいか	デシジョンテーブル
パスワード編集	・管理者のパスワード変更	・各入力欄に記述されている内容は正しいか	デシジョンテーブル
管理者登録・編集確認	・管理者情報の確認 ・登録処理 ・修正処理	・管理者情報が正しく表示されているか ・登録が正しくされているか ・修正ページに表示されている内容が正しいか	要素別動作確認書
観測箱一覧	・観測箱情報の表示 ・各観測箱情報の操作リンク ・特定のカラムのソート機能	・観測箱情報の表示内容は正しいか ・各観測箱情報の操作リンクは正しいか ・カラムのソート機能が正しいか	要素別動作確認書
観測箱登録・編集	・観測箱情報を新規登録 ・観測箱情報を編集 ・観測箱登録・編集確認ページへのリンク	・各入力欄に記述されている内容は正しいか ・セッションが存在した場合、正しく表示されているか ・JavaScriptの動作は正しいか ・観測箱登録・編集確認ページへのリンクは正しいか	デシジョンテーブル
観測箱登録・編集確認	・観測箱情報の確認 ・登録処理 ・修正処理	・観測箱情報が正しく表示されているか ・登録が正しくされているか ・修正ページに表示されている内容が正しいか	要素別動作確認書
観測動物一覧	・観測動物情報の表示 ・各観測動物情報の操作リンク ・特定のカラムのソート機能	・観測動物情報の表示内容は正しいか ・各観測動物情報の操作リンクは正しいか ・カラムのソート機能が正しいか	要素別動作確認書
観測動物登録・編集	・観測動物情報を新規登録 ・観測動物情報を編集 ・観測動物登録・編集確認ページへのリンク	・各入力欄に記述されている内容は正しいか ・セッションが存在した場合、正しく表示されているか ・観測動物登録・編集確認ページへのリンクは正しいか	デシジョンテーブル
観測動物登録・編集確認	・観測動物情報の確認 ・登録処理 ・修正処理	・観測動物情報が正しく表示されているか ・登録が正しくされているか ・修正ページに表示されている内容が正しいか	要素別動作確認書
農作物一覧	・農作物情報の表示 ・各農作物情報の操作リンク ・特定のカラムのソート機能	・農作物情報の表示内容は正しいか ・各農作物情報の操作リンクは正しいか ・カラムのソート機能が正しいか	要素別動作確認書
農作物登録・編集	・農作物情報を新規登録 ・農作物情報を編集 ・農作物登録・編集確認ページへのリンク	・各入力欄に記述されている内容は正しいか ・セッションが存在した場合、正しく表示されているか ・農作物登録・編集確認ページへのリンクは正しいか	デシジョンテーブル
農作物登録・編集確認	・農作物情報の確認 ・登録処理 ・修正処理	・農作物情報が正しく表示されているか ・登録が正しくされているか ・修正ページに表示されている内容が正しいか	要素別動作確認書
ログ閲覧	・フィルター絞り込み ・各項目のソート機能 ・CSVファイルの書き出し	・フィルター絞り込み機能が正しいか ・ソート機能が正しいか ・CSVファイルが正しく書き出されているか	デシジョンテーブル
意見交換掲示板	・BBSへの書き込み ・書き込みの削除	・投稿者情報の内容は正しいか ・投稿削除は正しく行われるか	デシジョンテーブル

管理者ログイン画面

	項目名	入力	1	2	3	4	5	6	7
条件	ログインID	入力なし	○						
		登録されていないログインID		○	○				
		登録されているログインID				○	○	○	
	パスワード	入力なし	－	○		○			
		入力されているログインIDと対応していないパスワード	－		－		○		
		入力されているログインIDと対応しているパスワード	－		－			○	
トリガ	ログイン	ログインボタン押下	○	○	○	○	○	○	
	一般者向け	一般者向けリンク押下							○
動作		管理者トップページへ遷移						○	
		トップページへ遷移							○
	必須	必須チェックエラー	○	○		○			
	参照	参照チェックエラー			○		○		
確認方法			自動	自動	自動	自動	自動	自動	自動
結果		11月20日	○	○	○	○	○	○	○

利用者登録・編集画面

	項目名	入力	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
条件	姓	入力なし	○																
		入力あり	－	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
	名	入力なし	－	○															
		入力あり	－		○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
	せい	入力なし	－	－	○														
		入力あり	－	－		○	○	○	○	○	○	○	○	○	○	○	○	○	○
	めい	入力なし	－	－	－	○													
		入力あり	－	－	－		○	○	○	○	○	○	○	○	○	○	○	○	○
	E-mail	入力なし	－	－	－	－	○												
		入力あり	－	－	－	－		○	○	○	○	○	○	○	○	○	○	○	○
	検知メールの受け取り	受け取らない	－	－	－	－	－	○			○	○	○	○	○	○	○	○	○
		自分の農地のみ	－	－	－	－	－		○										
		全ての農地	－	－	－	－	－			○									
	農地個数	0	－	－	－	－	－	－	－	－	○	○	○						
		1	－	－	－	－	－	－	－	－				○	○	○			
		2以上	－	－	－	－	－	－	－	－							○	○	○
	農地場所	観測箱の検知範囲円内	－	－	－	－	－	－	－	－				－	○		○		○
		観測箱の検知範囲円外	－	－	－	－	－	－	－	－				－		○		○	○
	農作物	代表的な農作物を選択	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－
		未設定	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－
		それら以外	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－	－
	「ここに農地を追加」	なし	－	－	－	－	－	－	－	－	○			○					
		あり	－	－	－	－	－	－	－	－		○	○		○	○	○	○	○
	「操作を取り消し」	なし	－	－	－	－	－	－	－	－				－	－	－	－	－	－
		あり	－	－	－	－	－	－	－	－	○		○	－	－	－	－	－	－
	農地クリックによる削除	なし	－	－	－	－	－	－	－	－				－	－	－	－	－	－
		あり	－	－	－	－	－	－	－	－		○	○	－	－	－	－	－	－
トリガ	農地確認	農地確認ボタン押下	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
動作		利用者登録・編集確認ページへ遷移						○	○	○	○	○	○		○	○	○	○	○
	必須	必須チェックエラー	○	○	○	○	○												
		農地追加途中エラー												○					
確認方法			自動	自動	自動	自動	自動	自動	自動	自動	手動	手動	手動	手動	手動	手動	手動	手動	手動
結果		11月6日									○	○	○	×	○	○	×	○	×
		11月20日	○	○	○	○	○	○	○	○				○			○		○

管理者新規登録画面

	項目名	入力	1	2	3	4	5	6	7	8	9	10	11
条件	姓	入力なし	○										
		入力あり		○	○	○	○	○	○	○	○	○	○
	名	入力なし	－	○									
		入力あり	－		○	○	○	○	○	○	○	○	○
	せい	入力なし	－	－	○								
		入力あり	－	－		○	○	○	○	○	○	○	○
	めい	入力なし	－	－	－	○							
		入力あり	－	－	－		○	○	○	○	○	○	○
	ログインID	入力なし	－	－	－	－	○						
		使用されているログインID	－	－	－	－		○					
		使用されていないログインID	－	－	－	－			○	○	○	○	○
	パスワード	入力なし	－	－	－	－	－	－	○				
		入力あり	－	－	－	－	－	－		○	○	○	○
	パスワード（再入力）	入力なし	－	－	－	－	－	－	－	○			
		パスワード項目と異なる	－	－	－	－	－	－	－		○		
		パスワード項目と一致	－	－	－	－	－	－	－			○	○
	E-mail	入力なし	－	－	－	－	－	－	－	－	－	○	
		入力あり	－	－	－	－	－	－	－	－	－		○
トリガ	登録確認	「登録確認」ボタンを押下する	○	○	○	○	○	○	○	○	○	○	○
動作		管理者登録確認ページへ遷移											○
	必須	必須チェックエラー	○	○	○	○	○		○	○		○	
	参照	ログインIDチェックエラー						○					
	参照	パスワードチェックエラー									○		
確認方法			自動	自動	自動	自動	自動	自動	自動	自動	自動	自動	自動
結果		11月20日	○	○	○	○	○	○	○	○	○	○	○

管理者編集画面

	項目名	入力	1	2	3	4	5	6	7	8
条件	姓	入力なし	○							
		入力あり		○	○	○	○	○	○	○
	名	入力なし	–	○						
		入力あり	–		○	○	○	○	○	○
	せい	入力なし	–	–	○					
		入力あり	–	–		○	○	○	○	○
	めい	入力なし	–	–	–	○				
		入力あり	–	–	–		○	○	○	○
	ログインID	入力なし	–	–	–	–	○			
		使用されているログインID	–	–	–	–		○		
		使用されていないログインID	–	–	–	–			○	○
	E-mail	入力なし	–	–	–	–	–	–	○	
		入力あり	–	–	–	–	–	–		○
トリガ	登録確認	「登録確認」ボタンを押下する	○	○	○	○	○	○	○	○
動作		管理者登録確認ページへ遷移								○
	必須	必須チェックエラー	○	○	○	○	○		○	
	参照	ログインIDチェックエラー						○		
確認方法			自動	自動	自動	自動	自動	自動	自動	自動
結果		11月20日	○	○	○	○	○	○	○	○

パスワード編集画面

	項目名	入力	1	2	3	4	5	6	7
条件	現在のパスワード	入力なし	○	−	−				−
		登録したパスワードと一致しない		−	−	○			−
		登録したパスワードと一致する		−	−		○	○	−
	新しいパスワード	入力なし		○	−				−
		入力あり	○		−	○	○	○	−
	新しいパスワード(再入力)	入力なし			○				−
		新しいパスワード項目と一致しない	−	−		−	○		−
		新しいパスワード項目と一致する	−	−		−		○	−
トリガ	変更	「変更」ボタンを押下する	○	○	○	○	○	○	
	キャンセル	「キャンセル」ボタンを押下する							○
動作		管理者一覧ページへ遷移						○	
		モーダルを閉じる							○
	必須	必須チェックエラー	○	○	○				
	参照	現在のパスワードチェックエラー				○			
	参照	新しいパスワードチェックエラー					○		
確認方法			自動	自動	自動	自動	自動	自動	自動
結果		11月20日	○	○	○	○	○	○	○

観測箱登録・編集画面

	項目名	入力	1	2	3	4	5	6	7	8	9
条件	観測箱名	入力なし	○								
		入力あり		○	○	○	○	○	○	○	○
	備考	入力なし	–	○		–	–	–	–	–	–
		入力あり	–		○	–	–	–	–	–	–
	新規観測箱数	0	–	○	○	○	○	○			
		1	–						○	○	○
	「ここに観測箱を追加」	なし	–	○	○	○					
		あり	–				○	○	○	○	○
	「操作を取り消し」	なし	–	○	○		○		○		
		あり	–			○		○		○	○
	クリックで観測箱を削除	なし	–	○	○	○			○	○	
		あり	–				○	○			○
トリガ	登録確認	登録確認ボタン押下	○	○	○	○	○	○	○	○	○
動作		観測箱確認ページへ遷移							○	○	○
	必須	必須観測箱名チェックエラー	○								
	必須	必須観測箱設置チェックエラー		○	○	○	○	○			
確認方法			自動	手動	手動	手動	手動	手動	手動	手動	手動
結果		11月6日		○	○	○	○	○	○	○	○
		11月20日	○								

観測動物登録・編集画面

	項目名	入力	1	2	3	4	5
条件	群れの名称	入力なし	○				
		入力あり		○	○	○	○
	周波数(Hz)	入力なし	–	○			
		半角数字で1以上ではない	–		○		
		半角数字で1以上である	–			○	○
	備考	入力なし	–	–	–	○	
		入力あり	–	–	–		○
トリガ	登録確認	「登録確認」ボタンを押下する	○	○	○	○	○
動作		観測動物確認ページへ遷移				○	○
	必須	必須チェックエラー	○	○	○		
確認方法			自動	自動	自動	自動	自動
結果		11月20日	○	○	○(許容する)	○	○

農作物登録・編集画面

	項目名	入力	1	2	3	4
条件	農作物名	入力なし	○			
		入力あり		○	○	○
	被害開始日	入力なし	-	○		
		入力あり	-		○	○
	被害終了日	入力なし	-	-	○	
		入力あり	-	-		○
トリガ	登録確認	「登録確認」ボタンを押下する	○	○	○	○
動作		農作物確認ページへ遷移				
	必須	必須チェックエラー	○	○		
	参照	日付けチェックエラー			○	○
確認方法			自動	自動	自動	自動
結果		11月20日	○	○	○	○

ログ閲覧画面

	項目名	入力	1	2	3	4	5	6	7	8	9	10	11	12	13	14
条件	観測箱名	フィルタなし	○				○	○	○	○	○	○				
		それ以外の項目		○	○	○							○	○	○	○
	位置	フィルタなし	○	○			○	○	○	○	○	○	○			
		それ以外の項目			○	○								○	○	○
	猿群	フィルタなし	○	○	○		○	○	○	○	○	○	○	○		
		それ以外の項目				○									○	○
	ソート項目クリック	日時					○						○	○	○	○
		観測箱ID						○								
		観測箱名							○							
		位置								○						
		猿群									○					
		タイプ										○				
トリガ	ファイル書き出し	「ファイル書き出し」ボタン押下														○
動作		フィルタリングされる		○	○	○							○	○	○	○
		ソートされる					○	○	○	○	○	○	○	○	○	○
		ファイル書き出しされる														○
確認方法			手動	手動	手動	手動	手動	手動	手動	手動	手動	手動	手動	手動	手動	手動
結果		11月6日	○	○	○	○	○	○	○	○	○	○	○	○	○	○

掲示板画面

	項目名	入力	1	2	3	4	5	6	7	8	9	10	11	12	13
条件	投稿者名	入力なし	○								－	－	－	－	－
		入力あり		○	○	○	○	○	○	○	－	－	－	－	－
	メールアドレス	入力なし	－	○							－	－	－	－	－
		正規表現にそぐわない入力	－		○						－	－	－	－	－
		正規表現にそった入力	－			○	○	○	○	○	－	－	－	－	－
	件名	入力なし	－	－	－	○					－	－	－	－	－
		入力あり	－	－	－		○	○	○	○	－	－	－	－	－
	メッセージ	入力なし	－	－	－	－	○				－	－	－	－	－
		入力あり	－	－	－	－		○	○	○	－	－	－	－	－
	削除コード	入力なし	－	－	－	－	－	○			－	－	－	－	－
		英数字4文字ではない	－	－	－	－	－		○		－	－	－	－	－
		英数字4文字である	－	－	－	－	－			○	－	－	－	－	－
	投稿No.	入力なし	－	－	－	－	－	－	－	－	○				
		正しくない投稿No.	－	－	－	－	－	－	－	－		○			
		正しい投稿No.	－	－	－	－	－	－	－	－			○	○	○
	削除コード(削除時)	入力なし	－	－	－	－	－	－	－	－	－	－	○		
		投稿No.に対応していない	－	－	－	－	－	－	－	－	－	－		○	
		投稿No.に対応している	－	－	－	－	－	－	－	－	－	－			○
トリガ	投稿	「投稿」ボタンを押下	○	○	○	○	○	○	○	○					
	削除	「削除」ボタンを押下									○	○	○	○	○
動作		投稿完了								○					
		削除完了													○
	必須	投稿必須チェックエラー	○	○		○	○	○							
	必須	削除必須チェックエラー									○				
	参照	メールアドレス参照チェックエラー			○										
	参照	投稿No.参照チェックエラー										○			
	参照	削除コード参照チェックエラー							○					○	
	参照	削除時削除コード参照チェックエラー											○		
確認方法			自動	自動	自動	自動	自動	自動	自動	自動	自動	自動	自動	手動	手動
結果		11月20日	○	○	○	○	○	○	○	○	○	○	○	○	○

G. テスト　3. 要素別動作確認表

利用者トップ画面

項目番号	要素名	論理項目種別	確認内容	確認方法	確認結果	修正日
1	日時	ラベル	現在時刻がずれなく表示されている	手動	○	
2	google map	その他	サルの出現率が正しく表示されている	手動	○	
3	このWEBサイトのマニュアル(PDF)を表示	ボタン	マニュアルを表示する	自動	○	
4		その他	1分毎にページが更新されている	手動	○	
5	獣害対策掲示板へ	ボタン	獣害対策掲示板画面へ遷移する	自動	○	
6	このシステムの概要(PDF)を表示	ボタン	ポスター画面へ遷移する	自動	○	

メニューバー

項目番号	要素名	論理項目種別	確認内容	確認方法	確認結果	修正日
1	ログイン中	ラベル	ログインしている管理者の姓が表示されている	自動	○	
2	トップ	リンク	管理者トップページへ遷移	自動	○	
3	利用者一覧	リンク	利用者一覧ページへ遷移	自動	○	
4	管理者一覧	リンク	管理者一覧ページへ遷移	自動	○	
5	観測箱一覧	リンク	観測箱一覧ページへ遷移	自動	○	
6	野生動物一覧	リンク	野生動物一覧ページへ遷移	自動	○	
7	農作物一覧	リンク	農作物一覧ページへ遷移	自動	○	
8	ログ閲覧	リンク	ログページへ遷移	自動	○	
9	ログアウト	リンク	ログアウトして管理者ログインページへ遷移	自動	○	

G. テスト 3. 要素別動作確認表

管理者トップ画面

項目番号	要素名	論理項目種別	確認内容	確認方法	確認結果	修正日
1	日時	ラベル	現在時刻がずれなく表示されている	手動	○	
2	google map	その他	サルの出現率が正しく表示されている	手動	○	
3	農地アイコンを表示	その他	農地アイコン表示の切り替えが可能	手動	○	
4	このWEBサイトのマニュアル(PDF)を表示	ボタン	マニュアルを表示する	自動	○	
5		その他	1分毎にページが更新されている	手動	○	
6	獣害対策掲示板へ	ボタン	獣害対策掲示板画面へ遷移する	自動	○	

G. テスト 3. 要素別動作確認表

利用者一覧画面

項目番号	要素名	論理項目種別	確認内容	確認方法	確認結果	修正日
1	氏名	ラベル	氏名が正しく表示されている	自動	○	
2	氏名(よみ)	ラベル	氏名(よみ)が正しく表示されている	自動	○	
3	氏名(よみ)	ラベル	ソートする	手動	○	
4	E-mail	ラベル	E-mailが正しく表示されている	自動	○	
5	検知メールの受け取り	ラベル	検知メールの受け取りが正しく表示されている	自動	○	
6	農地位置(農作物)	ラベル	農地位置が正しく表示されている	自動	○	
7	編集操作	ボタン	ボタンに対応した利用者の利用者編集ページへ遷移	自動	○	
8	削除操作	ボタン	ボタンに対応した利用者の削除モーダルを展開	自動	○	
9	削除モーダル	ボタン	削除ボタンを押下すると利用者情報の削除、 キャンセルボタンを押下すると処理を何もせずにモーダルを閉じる	手動	○	
10	新規登録	ボタン	利用者登録ページへ遷移	自動	○	
11	新規登録	ボタン	観測箱が一つも動作していない場合ボタンを表示させない	手動	○	
12	検索	テキストボックス	利用者の氏名の全部もしくは一部が一致する利用者を検索する	手動	○	
13	氏名(よみ)	ラベル	検索した後にソートを実行すると検索した中でソートする	手動	×	10月25日
14		その他	農地を所有している利用者を削除した時、 所有している農地も削除する	手動	○	

利用者登録・編集確認画面

項目番号	要素名	論理項目種別	確認内容	確認方法	確認結果	修正日
1	氏名	ラベル	氏名が正しく表示されている	自動	○	
2	氏名(よみ)	ラベル	氏名(よみ)が正しく表示されている	自動	○	
3	E-mail	ラベル	E-mailが正しく表示されている	自動	○	
4	検知メールの受け取り	ラベル	検知メールの受け取りが正しく表示されている	自動	○	
5	google map	その他	登録する農地の位置と関係が正しいか	手動	○	
6	登録する	ボタン	利用者情報の登録処理を行い、利用者一覧ページへ遷移	自動	○	
7	入力内容を修正する	ボタン	入力内容を保持したまま利用者登録・編集ページへ遷移	自動	○	

管理者一覧画面

項目番号	要素名	論理項目種別	確認内容	確認方法	確認結果	修正日
1	氏名	ラベル	氏名が正しく表示されている	手動	○	
2	氏名(よみ)	ラベル	氏名(よみ)が正しく表示されている	手動	○	
3	氏名(よみ)	ラベル	ソートする	手動	×	10月26日
4	ログインID	ラベル	ログインIDが正しく表示されている	手動	○	
5	ログインID	ラベル	ソートする	手動	×	10月26日
6	E-mail	ラベル	E-mailが正しく表示されている	手動	○	
7	編集操作	ボタン	ボタンに対応した管理者の管理者編集ページへ遷移	自動	○	
8	削除操作	ボタン	ボタンに対応した管理者の削除モーダルを展開	自動	○	
9	削除操作	ラベル	管理者番号が先頭の管理者は削除項目を生成しない	手動	○	
10	削除モーダル	ボタン	削除ボタンを押下すると管理者情報の削除、 キャンセルボタンを押下すると処理を何もせずにモーダルを閉じる	手動	○	
11	氏名(よみ)	ラベル	検索した後にソートを実行すると検索した中でソートする	手動	×	10月26日
12	ログインID	ラベル	検索した後にソートを実行すると検索した中でソートする	手動	×	10月26日
13	新規登録	ボタン	管理者登録ページへ遷移	自動	○	
14	検索	テキストボックス	管理者の氏名の全部もしくは一部が一致する管理者を検索する	手動	○	

管理者登録・編集確認画面

項目番号	要素名	論理項目種別	確認内容	確認方法	確認結果	修正日
1	氏名	ラベル	氏名が正しく表示されている	自動	○	
2	氏名（よみ）	ラベル	氏名（よみ）が正しく表示されている	自動	○	
3	ログインID	ラベル	ログインIDが正しく表示されている	自動	○	
4	E-mail	ラベル	E-mailが正しく表示されている	自動	○	
5	パスワード	ラベル	パスワードが隠されて表示されている	自動	○	
6	登録する	ボタン	管理者情報の登録処理を行い、管理者一覧ページへ遷移	自動	○	
7	入力内容を修正する	ボタン	入力内容を保持したまま管理者登録・編集ページへ遷移	自動	○	

観測箱一覧画面

項目番号	要素名	論理項目種別	確認内容	確認方法	確認結果	修正日
1	ID	ラベル	IDが正しく表示されている	手動	○	
2	ID	ラベル	ソートする	手動	○	
3	観測箱名	ラベル	観測箱名が正しく表示されている	手動	○	
4	観測箱名	ラベル	ソートする	手動	○	
5	状態	ラベル	状態が正しく表示されている	手動	○	
6	状態	ラベル	ソートする	手動	○	
7	最終応答時間	ラベル	最終応答時間が正しく表示されている	手動	○	
8	IPアドレス	ラベル	IPアドレスが正しく表示されている	手動	○	
9	観測箱位置	ラベル	観測箱位置が正しく表示されている	手動	○	
10	備考	ラベル	備考が正しく表示されている	手動	○	
11	編集操作	ボタン	ボタンに対応した観測箱の観測箱編集ページへ遷移	自動	○	
12	削除操作	ボタン	ボタンに対応した観測箱の削除モーダルを展開	自動	○	
13	削除モーダル	ボタン	削除ボタンを押下すると観測箱情報の削除、 キャンセルボタンを押下すると処理を何もせずにモーダルを閉じる	手動	○	
14	新規登録	ボタン	観測箱登録ページへ遷移	自動	○	

観測箱登録・編集確認画面

項目番号	要素名	論理項目種別	確認内容	確認方法	確認結果	修正日
1	観測箱名	ラベル	観測箱名が正しく表示されている	手動	○	
2	備考	ラベル	備考が正しく表示されている	手動	○	
3	観測箱位置	ラベル	観測箱位置が正しく表示されている	手動	○	
4	google map	その他	登録する観測箱の位置と関係が正しいか	手動	○	
5	登録する	ボタン	観測箱情報の登録処理を行い、観測箱一覧ページへ遷移	手動	○	
6	入力内容を修正する	ボタン	入力内容を保持したまま観測箱登録・編集ページへ遷移	手動	○	

観測動物一覧画面

項目番号	要素名	論理項目種	確認内容	確認方法	確認結果	修正日
1	ID	ラベル	IDが正しく表示されている	手動	○	
2	ID	ラベル	ソートする	手動	○	
3	群れの名称	ラベル	群れの名称が正しく表示されている	手動	○	
4	周波数	ラベル	周波数が正しく表示されている	手動	○	
5	備考	ラベル	備考が正しく表示されている	手動	○	
6	編集操作	ボタン	ボタンに対応した観測動物の観測動物編集ページへ遷移	自動	○	
7	削除操作	ボタン	ボタンに対応した観測動物の削除モーダルを展開	自動	○	
8	削除モーダル	ボタン	削除ボタンを押下すると観測動物情報の削除、 キャンセルボタンを押下すると処理を何もせずにモーダルを閉じる	手動	○	
9	新規登録	ボタン	観測動物登録ページへ遷移	自動	○	

観測動物登録・編集確認画面

項目番号	要素名	論理項目種別	確認内容	確認方法	確認結果	修正日
1	群れの名称	ラベル	群れの名称が正しく表示されている	自動	○	
2	備考	ラベル	備考が正しく表示されている	自動	○	
3	周波数	ラベル	周波数が正しく表示されている	自動	○	
4	登録する	ボタン	観測動物情報の登録処理を行い、観測動物一覧ページへ遷移	自動	○	
5	入力内容を修正する	ボタン	入力内容を保持したまま観測動物登録・編集ページへ遷移	自動	○	

農作物一覧画面

項目番号	要素名	論理項目種	確認内容	確認方法	確認結果	修正日
1	農作物名	ラベル	農作物名が正しく表示されている	手動	○	
2	農作物名	ラベル	ソートする	手動	○	
3	被害開始日	ラベル	被害開始日が正しく表示されている	手動	○	
4	被害開始日	ラベル	ソートする	手動	○	
5	被害終了日	ラベル	被害終了日が正しく表示されている	手動	○	
6	被害終了日	ラベル	ソートする	手動	○	
7	登録件数	ラベル	登録件数が正しく表示されている	手動	○	
8	登録件数	ラベル	ソートする	手動	○	
9	編集操作	ボタン	ボタンに対応した農作物の農作物編集ページへ遷移	自動	○	
10	削除操作	ボタン	ボタンに対応した農作物の削除モーダルを展開	自動	○	
11	削除モーダル	ボタン	削除ボタンを押下すると農作物情報の削除、 キャンセルボタンを押下すると処理を何もせずにモーダルを閉じる	手動	○	
12	新規登録	ボタン	農作物登録ページへ遷移	自動	○	
13	削除操作	その他	削除した農作物に対応していた農地があった場合、その農地の農作物を未設定にする	手動	○	

農作物登録・編集確認画面

項目番号	要素名	論理項目種別	確認内容	確認方法	確認結果	修正日
1	農作物名	ラベル	農作物名が正しく表示されている	自動	○	
2	被害開始日	ラベル	被害開始日が正しく表示されている	自動	○	
3	被害終了日	ラベル	被害終了日が正しく表示されている	自動	○	
4	登録する	ボタン	農作物情報の登録処理を行い、農作物一覧ページへ遷移	自動	○	
5	入力内容を修正する	ボタン	入力内容を保持したまま農作物登録・編集ページへ遷移	自動	○	

ヒューリスティックス評価マニュアル

ユーザビリティ評価においてユーザビリティ・インスペクションという手法がある。これは評価者自らの知識や経験に基づいて行うものである。そこで評価に客観性を持たせるためにヤコブ・ニールセンが考案した 10 ヒューリスティックスをガイドラインとして用いる。評価する時はこれらのヒューリスティックスの観点から評価する

1.　システム状態の視認性

システムは常に、適切なフィードバックを通じて、妥当な時間内で、ユーザーに対して「何が起こっているか」を知らせなければならない。

- 待ち状態を示す砂時計型カーソル
- 完了を知らせる確認画面

2.　システムと実世界の調和

システムは、システム指向の言語ではなく、ユーザーの言語（ユーザーに馴染みのある言葉や概念）を使って、ユーザーとコミュニケートしなければならない。利用される環境のしきたりに従い、自然で理にかなった形で情報を提示するようにシステムを作る。

- ゴミ箱のアイコン
- 買い物かごアイコン
- 日本語のウェブサイトは日本語を用いる

3.　ユーザコントロールと自由度

ユーザーはしばしば使うべき機能の選択を誤るので、明確な「非常口」（長ったらしいダイアログ対話をしなくても、望まない状態から抜けれる仕組み）が必要である。アンドゥとリドゥを提供する。

- 戻るボタンがある
- ウェブブラウザのサイズを変更してもデザインが変わらない

4.　一貫性と標準化

他のアプリケーションと異なる言葉、状態、作用を UI に実装して、ユーザーに「これはつまり～と同じことだろうか？」といった混乱を与えてはならない。プラットフォームのしきたりに従え。

- ページデザインを統一する
- ページタイトルを一致させる

5.　エラーの防止

優れたエラーメッセージを用意するよりも、はじめから注意深くデザインして問題発生を防止するほうが良い。誤操作を引き起こしそうな条件を除去またはチェックしておいて、ユーザーに対しては操作を実行するかどうかを事前に確認する。

- 入力フォームの必須項目には印を付けて目立たせる

6.　記憶しなくても、見れば分かるように

オブジェクトや作用や選択肢を可視化することによって、ユーザーの記憶負荷を最小限にしよう。ユーザーはシステムとのやりとりを続ける中で、情報を記憶しなくても済むようにしなければならない。システムを使うための説明/指南は視覚的で、いつでも容易に見れるようにすべきである。

- アイコンのヘルプ

7.　柔軟性と効率性

システム利用を効率化できる仕組みは、初心者ユーザーには見られないが、熟練ユーザーの作業を高速化し、ひいては未経験者/経験者問わずユーザーの要求を満たすことにつながるだろう。ユーザーがたびたび実行する機能については、ユーザー自身が効率化を調整できるようにする。

- キーボード・ショートカットキーの設定

8.　美的で最小限のデザイン

システムとユーザー間の対話においては、見当違いな不必要な情報を含んではならない。あらゆる余計な情報群は、関連する情報群と競合してしまうので、相対的に、必要な情報の露出を低下させてしまう。

- テキストだけでなく、効果的に画像を使用する
- ページには見出しを付けて、空白や行間を用意する

9.　ユーザによるエラー認識、診断、回復をサポートする

エラーメッセージは自然言語で表現されるべきであって、プログラムコードで表現されるべきではない。メッセージは問題点を正確に示し、解決策を建設的に提言しなければならない。

- カスタムメッセージを表示する
- 誤入力の項目がある場合は、メッセージを表示した上で、さらに項目名に印を付けて目立たせる

10. ヘルプとマニュアル

システムが説明文書なしでも使えるほうが良いとしても、ヘルプや説明文書は必要だろう。この手の情報はすべて、容易に検索でき、ユーザーの行為に焦点を当てることができ、なすべき具体的な手順がリストアップされ、分量が大きすぎないようにすべきである。

- 機能の説明だけでなく、タスクの実行手順を示す
- 文字だけでなくイラストやスクリーンショット、動画などを併用する

画面	問題点	解決案	優先度	担当者	修正開始日	修正終了日
全体	エラーメッセージの表示が小さく、分かりづらい	アラート表示に変更する	3	浜野	9月17日	9月24日
利用者一覧	氏名検索におけるラベルが「利用者名」となっており「氏名」と揺れがある	「氏名」に統一する	3	浜野	9月17日	9月17日
管理者一覧	氏名検索におけるラベルが「管理者名」となっており「氏名」と揺れがある	「氏名」に統一する	3	浜野	9月17日	9月17日
観測箱登録	観測箱を設置した後クリックして削除できる旨の文が表示されない	カーソルを重ねたときにメッセージを表示する	3	天本	9月17日	9月17日
利用者一覧	「情報の受け取り範囲」という表現はわかりづらい	表現を変更する	3	浜野	9月19日	9月19日
利用者登録	ただ地図が描かれているだけで農地登録の説明がない。	説明文を表示する	3	浜野	9月19日	9月19日
利用者登録	管理者情報登録ページには項目のラベルがあるがこちらにはない	管理者に統一	3	浜野	9月19日	9月19日
全体	ページタイトルがおかしい。トップと管理者一覧のタイトルが「Admins」、 利用者一覧と観測箱一覧が「userslist」になっている。	ページタイトルを変更する	3	天本	9月17日	9月17日
ログ	フィルターの「情報タイプ」が何を示しているかわからない	「情報タイプ」の仕様を変更する	3	森	9月17日	9月17日
ログ	フィルターの解除（全部表示）をしたい場合、 わかりにくい（現在はフィルターを空白にすればよいようになっている）	フィルター解除方法の変更	3	森	10月2日	10月2日
利用者一覧	観測箱との結びつきがわかりづらい	結びつきの仕様検討	3	全員	9月17日	9月17日
利用者登録	ラジオボタンとラベルが一体化されていない。若干クリックしにくい	ラジオボタンのラベルを選択しても反応するようにする	3	浜野	9月19日	9月19日
利用者登録	利用者がちゃんと登録できたのかが分かりにくい。登録時に、登録完了メールを送るべき	登録完了メールを送る	3	森	10月7日	10月7日
利用者変更 観測箱変更 観測動物変更 農作物変更	編集時には、誤解を防止するために「～情報の編集」と表示すべき	「登録」表示から「編集」表示に変更する	3	浜野	9月19日	9月19日
各一覧	削除モーダルの表記ゆれ	表記ゆれの改善	3	浜野	9月19日	9月19日
全体	WEBサイトのアイコンがCakePHPのまま	システムのアイコンに変更する	3	天本	9月17日	9月17日
各一覧	編集と削除だけタブに名前がない	「編集操作」「削除操作」と記載する	2	天本	9月26日	9月26日
農地登録確認 観測箱登録確認	確認画面にマップ表示がない	登録する農地や観測箱のマップ情報を表示する	2	森	9月26日	9月26日
メール	本文に情報がほとんどない	観測箱や農地の住所を表示する	2	森	9月26日	9月26日
ログ	ダウンロードしたファイルの拡張子がなく、 ファイル名も意味のない英数字の組み合わせのためわかりにくい。	ファイル名を適切にして、拡張子をつける	2	森	2014/10/17	2014/10/17
利用者登録	氏名などは必須だが農地登録は必須でないなど、必須かそうでないかがわからない	必須項目には印を付ける	2	浜野	9月26日	9月26日
利用者登録	利用者一覧ページでは「すべての農地」だが、登録ページでは「すべての検知範囲」になっている	すべての検知範囲で統一する	2	天本	9月26日	9月26日
トップ	農地や観測箱にマウスを合わせるとクリックできそうなカーソルに代わるが、 実際はクリックしても何も起きない	カーソルを変更させない	2	天本	9月26日	9月26日
利用者トップ	このサイトの説明ボタンを隠すボタンが機能していない	画面遷移で違う画面で説明を表示する	2	天本	9月26日	9月26日
利用者トップ	サイトの説明の表示状態が保持されない	画面遷移で違う画面で説明を表示する	2	浜野	9月26日	9月26日
管理者変更	パスワード変更が黄色なのが違和感がある	パスワード変更ボタンの色を変更する	2	浜野	9月26日	9月26日
全体	WEBサイトからマニュアルにアクセスすることができない	マニュアルのPDFファイルを閲覧できるようにする	1	浜野		
全体	ソートできる部分を青字で強調しているがクリックしてみるまでソート機能なのか分からない	ソートできる部分に「▲▼」などの記号をつける	1	浜野		
利用者一覧 観測箱一覧	農地情報に郵便番号を含んでいる場合と含んでいない場合がある	郵便番号は除外する	1	天本		
利用者一覧	初期のソート向きが表示されていない	どうソートされているのか分かるように表示する	1	浜野		
トップ	地図上でまれにストリートビューを開いてしまうことがある。これは誤操作を招く可能性がある	ストリートビューを開けないようにする	1	天本		
トップ	地図上のアイコンでクリックできるものとできないものがある。統一すべき	クリックしてメッセージを表示できるようにする	1	天本		
各一覧	利用者、管理者番号がないので人数の管理がしにくい	番号を表示する	1	浜野		
全体	ログアウトしなければ利用者画面が見れない	ログイン中からも利用者画面を表示する	1	浜野		

管理者管理機能

	指摘内容	修正	担当	期日	修正完了	分類
1	管理者のパスワード管理時に文字が隠されているため表示して確認したい	×				UI
2	管理者が他の管理者を編集・削除できるのはおかしい。スーパーユーザーを用意すべき	自分は自分のアカウントを削除できない ログイン情報を常に更新する	浜野	12月3日	12月3日	機能
3	管理者で1番が削除できない表示はいらない	表示方法を変える（グレーとかにする）	浜野	12月3日	12月3日	UI
4	管理者で「よみ」ではなく「ふりがな」	やる	浜野	12月3日	12月3日	UI
5	「氏名」と「管理者氏名」で表記ゆれ	やる	浜野	12月3日	12月3日	UI
6	管理者ごとに観測箱や農作物、観測動物を管理	今年はJA鈴鹿だけだから対応しない				機能
7	管理者トップに管轄地点の住所を表示する	×				UI
8	ログインユーザの情報が名前しかない。ログインIDにすべき	×				UI

利用者管理機能

	指摘内容	修正	担当	期日	分類
1	農地の編集はどうすればいいの	×			機能
2	利用者登録で自分の農地をまだ設置していないのにメール受取に自分の農地があるのはおかしい	×			UI

検知履歴管理機能

	指摘内容	修正	担当	期日	修正完了	分類
1	検知履歴を全て出すのはおかしい。ページ送りとかにすべき	×				UI
2	検知履歴のタイプはいらない	やる	森	2014/12/3	2014/12/3	UI

掲示板管理機能

	指摘内容	修正	担当	期日	修正完了日	分類
1	掲示板の名前にあるリンクがいらない	×				UI
2	掲示板の削除方法が分かりづらい。せめて各投稿の横に削除ボタンを設置	×				UI
3	掲示板の削除コードを忘れたらどうやって削除するの？	×				機能
4	掲示板は管理者のみ全ての投稿を削除できるようにする	_php_MyAdminの案内を設置	森	2014/12/3	2014/12/3	機能
5	掲示板とアカウント管理を紐づけて簡単に投稿できるようにする	×				機能
6	掲示板のメッセージ枠が小さい	×				UI
7	掲示板に投稿があったら管理者に通知などをする	×				機能
8	掲示板を管理するアカウントが必要。悪口とかあったらどうするのか	_php_MyAdminの案内を設置	森	2014/12/3	2014/12/3	機能

観測動物管理機能

	指摘内容	修正	担当	期日	修正完了日	分類
1	観測動物の周波数が半角数字のみだとエラーがほしい	プレイスフォルダーで指示す	浜野	12月3日	12月3日	UI

	複数画面に関わる機能 指摘内容	修正	担当	期日	修正完了日	分類
1	新規登録や編集時にエンターキーを押下した時に入力途中で情報が postされてしまいなぜエラーメッセージが 表示されているか分からない（特にマップ上にアイコンを設置する画面）	formをマップ検索と別にする	光岡	2014/12/5	2014/12/5	UI
2	アラート表示の表示元が「Webページ」からになっており、本システム名にすべき	調査してできそうだったらやる	浜野	2014/12/3	2014/12/3	UI
3	マップ上のアイコンの操作でドラッグできると思ってしまう	×				UI
4	登録一覧メニューから新規登録したい	×				UI
5	完了確認はアラート表示ではなくインフォメーション表示とかにすべき	調査してできそうだったらやる	浜野	2014/12/3	2014/12/3	UI
6	マップ上のアイコンの削除とかは確認表示すべき	やる	光岡	2014/12/5	2014/12/5	UI
7	サルアイコンが可愛くて憎しみさが薄れる	×				UI
8	農地アイコンが一種類だと全て米だと認識してしまう	×				UI
9	テキスト欄がpostすると小さくなってplaceholderが見えづらくなる	×				UI
10	マップの倍率が高い時のみアイコンを設置できるようにする。農地がすごく隣接してたら登録できない	高い精度が求められないためあいまいでよい				機能
11	農地アイコン非表示のチェックボックスの場所が分かりづらい	改行しているのは直す	天本	2014/12/5	2014/12/5	UI
12	1つの枠内に編集と削除を入れる	×				UI
13	マップをテキストから移動した時にその場所に目印などを置く	×				UI
14	infowindowを大きくする	×				UI
15	メニューにマニュアル表示や掲示板表示があるべき	ツールボックスをメニューに作る	森	2014/12/3	2014/12/3	UI
16	マウスのホイールでマップの拡大縮小	×				UI
17	confirm画面でinfowindowを再表示すべき	マウスオーバーで表示できるようにする	天本	2014/12/5	2014/12/5	UI
18	マップのアイコンの「クリックで削除できます。」が黒字だったり赤字だったり	やる	天本	2014/12/5	2014/12/5	UI

獣類発見支援システムのユーザテスト

- 以下のタスクをそれぞれ実行してください。

1. 管理者(JA)向けページにログインしてください。
   ログイン ID：test
   パスワード：test

2. 新規に管理者(JA 職員)を登録してください。
   入力例
   姓：筑波
   名：太郎
   せい：つくば
   めい：たろう
   ログイン ID：taro
   パスワード：tsukuba
   パスワード（再入力）：tsukuba
   E-mail：taro@tsukuba.jp
   管轄地域登録：筑波大学敷地内

3. 2.で登録した管理者のパスワードを変更してください。
   新パスワード：JAsuzuka

4. ログアウトして 2.で登録した管理者として再度ログインしてください。

5. 新規に利用者(一般の農家)を登録してください。また、米とりんご両方の農地をそれぞれ一つ、三重県四日市市の四日市ドーム周辺に登録してください。
   入力例
   姓：獣害
   名：次郎
   せい：じゅうがい
   めい：じろう
   E-mail：a2m2.jugai@gmail.com
   検知メールの受け取り：自分の農地のみ

6. 5.で登録した利用者の米の農地のみを削除してください。

7. 新規に観測箱を四日市ドーム周辺に登録してください。
   入力例
   観測箱名：四日市
   備考：周辺に果樹園が多い

8. 7.で登録した観測箱の位置を適当な位置に変更してください。

9. 新規に観測動物を登録してください。
   入力例
   群れの名称：亀山猿群 D
   周波数：20000
   備考：50 頭程度の群

10. 9.で登録した観測動物の情報である「備考」を「果物を好む」に変更してください。

11. 新規に農作物を登録してください。
    入力例
    農作物名：桃
    被害開始日：2 月 1 日
    被害終了日：4 月末日

12. 11.で登録した農作物の情報である「被害終了日」を「5 月末日」に変更してください。

13. 「亀山猿群 A」を観測した検知履歴のみを表示してください。

14. 受信した検知警告メールからシステムにアクセスして、サルの出没確率が高い地域を教えてください。

15. トップ画面から掲示板画面を表示し、メッセージを投稿してください。
    入力例
    投稿者名：三重三郎
    メールアドレス：mie@mie.jp
    件名：獣害対策
    メッセージ：獣害対策として筑波大学が開発したシステムの導入を提案します。
    削除コード：3333

16. 15.で投稿したメッセージを削除してください。

    以下の質問はトップページを表示してお答えください。

17. マニュアル(PDF ファイル)を表示してください。

18. 農地アイコンを非表示にしてください。

19. マップ表示について何か疑問に思った点、質問したいことはありますか

獣類発見支援システムの調査のお願い

この度は調査にご協力いただき、ありがとうございます。

- 本調査の目的は、「獣類発見支援システム」の WEB アプリケーション部分を使用していただき、その使用性を調査することです。
- いくつかタスクを提示するので、そのタスクを達成するためにシステムを操作してください。
- 操作中に感じたこと、思ったこと、考えていることは操作に支障がでない程度に発話してください。
- 操作中にこちらから質問する場合がありますが、あなたが答えられる範囲で答えてください。
- 全てのタスクを終えた後、簡単なアンケートに答えてもらいます。
- 調査の参加予定時間は 30 分程度です。
- 調査への参加は協力者の自由意志によるものであり、調査への参加を随時拒否・撤回することができます。
- この調査によって得られたデータは、個人が特定できないように処理します。
- 学内外にて発表する論文に調査内容を利用することがありますが、いかなる場合においても協力者のプライバシーは保全されます。

調査に関して、上記の内容を十分に理解し同意して頂けましたら、下部の署名欄に署名をお願いします。

平成　　年　　月　　日

所属＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿

氏名＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿＿

検収日 2014 年 12 月 27 日

検収書

検収項目	捺印または署名
Webアプリケーション	川出
運用マニュアル	川出

上記の成果物を検収しました。

署名 川出洋正