

災害時を想定した複数人での活動を支援するシステムの構築

廣瀬研究室 4年 c1151073 鈴木光明

平成30年1月15日

概要

近年、GPS 内蔵型携帯電話等の位置情報を計測できる情報端末が広く普及している。しかし、近年の熊本地震や北海道胆振地震をはじめとする災害時にその機能が有効に活用されていないのが現状である。現在位置情報を利用したガイドシステムは存在するが、利用者同士のコミュニケーションや情報共有が念頭に置かれていない場合が多い。また、従来災害時に用いられている伝言板等のシステムでは携帯電話と PC との連携が取れず、避難者の立場が考慮されていないことが問題視されている。

そこで、個人単位で位置情報を発信できる点に着目し、災害時における避難者の位置情報の共有の簡易化を図る Web アプリケーションを構築する。また、外部からの匿名性を保持した上で、設定したグループ内における個人間での情報の相互共有を可能にし、災害時に家族等の少人数のコミュニティ内で活用できるような機能も構築する。また、その機能を応用し、フィールドワーク等複数人の位置情報を相互に確認したいような場面でも利用できるようなシステムを構築する。(441 文字)

目次

| | | |
|--------------|----------------------------|-----------|
| 第 1 章 | 目的・テーマ設定 | 5 |
| 1.1 | テーマ設定の背景 | 5 |
| 1.2 | 過去の類似研究の例 | 5 |
| 第 2 章 | 提案 | 7 |
| 2.1 | 現状 | 7 |
| 2.1.1 | 災害時の逃げ遅れによる被害 | 7 |
| 2.1.2 | 避難後の安否確認 | 7 |
| 2.1.3 | 既存のアプリケーション | 7 |
| 2.2 | 提案するシステム | 8 |
| 第 3 章 | システムの構築環境 | 9 |
| 3.1 | データの正規化 | 9 |
| 3.2 | 作業環境 | 10 |
| 第 4 章 | システムの設計 | 13 |
| 4.1 | RDBMS によるデータ管理 | 13 |
| 4.2 | 個別アカウントの管理 | 13 |
| 4.2.1 | アカウントの制約に関して | 13 |
| 4.2.2 | アカウントの認証期限に関して | 14 |
| 4.3 | JavaScript を用いた動的な Web の作成 | 14 |
| 4.3.1 | Leaflet.js を用いたマップ作成 | 14 |
| 4.3.2 | Ajax による非同期通信 | 14 |
| 第 5 章 | システムの概観と実行 | 15 |
| 5.1 | 本システムの概観 | 15 |
| 5.2 | 個人アカウントの管理 | 15 |
| 5.2.1 | 個人アカウント作成 | 15 |
| 5.2.2 | ログイン | 16 |
| 5.3 | 位置情報を利用した情報共有 | 16 |
| 5.3.1 | 個人の位置情報の送信 | 16 |
| 5.3.2 | 少人数グループでの利用 | 17 |
| 5.3.3 | 設定地域内の分布表示 | 17 |
| 第 6 章 | 結論とシステムへの評価 | 21 |
| 6.1 | 結論 | 21 |
| 6.2 | システムへの評価と展望 | 21 |
| 6.2.1 | 有事の際の情報リセット | 21 |
| 6.2.2 | グループ参加の有無と個人の特典 | 21 |

| | | |
|-------|---------------------------|----|
| 6.2.3 | グループ管理のセキュリティ観点 | 21 |
| 6.2.4 | 同時利用時のキャパシティ | 22 |

第1章 目的・テーマ設定

本研究の設定する目的とテーマ設定に至るまでの背景について記述する。

1.1 テーマ設定の背景

日本では近年に見られる熊本地震や北海道胆振東部地震をはじめとした、地震や大雨による洪水等、多数の被害者が出るような災害が頻繁に起こっている [1]。その中でも高齢者をはじめとする災害弱者の逃げおくれによる被害が問題となっている [2]。他にも、電話や交通等のライフラインが混雑するため避難後の安否確認が困難になる [3]。そこで近年では避難経路の確保や把握等の日常的な活動を行なうことが促進されている。しかし、事前の準備のみならず実際に災害時の状況に陥った際の対応策が必要であると考えた。そこで、各自が持っている位置情報を発信できる情報端末を利用することで被災地の人間の逃げ遅れの確認や、家族等少人数でのコミュニティ内での情報の相互共有をする手段の必要性を感じた。

また、その他にも野外でのフィールドワークを複数人で行う際や集団のツアー客の自由行動等、予め時間や集合場所を決めていたとしても集団の動向を一目で把握できないのは不便な状況が存在する。宗森らによると既存の位置情報を使ったガイドシステムでは、その場での交換やチャットなど、利用者同士のコミュニケーションや情報共有は念頭に置かれていない場合が多いとされている [4, 5]。また、位置情報を扱うため、Apple Store や Google Play から配布されているアプリケーションであっても、位置情報を不正に取得している例が確認されている [6]。

更に、Google Play で nekonotesoft から配信されている情報共有ソフト “いまどこ?位置検索” 等をはじめとした無料のアプリケーションでは共有できる数に制限があったり全ての機能を利用するには有料であったりといったケースが存在する。例として上述の “いまどこ?位置検索” では14日間の試用期間では「自動測位機能」および「見守り機能」を利用することができず、また14日目以降の利用にはライセンスの購入と月額料の支払いが必要になる [7]。

そこで、特別な外部のアカウントや電話番号等の登録情報を必要としない、参加者の状態を個人単位で制限なくリアルタイムで相互に把握することができるアプリケーションが必要だと考えた。

1.2 過去の類似研究の例

本研究では災害時に複数人の位置情報を取り扱うことに重点を置くため、過去の研究の中から災害時の位置情報や複数人の位置情報を扱う取り組みが行われた文献について記述する。

松崎らは、スマートホームシステムのホームサーバ¹を利用して地震災害時における被災状況の確認や被災者の救助支援に応用するシステムを提案している [8]。モバイル端末とホームサーバの通信によりユーザとホームサーバとの距離を観測することで、在宅情報から生き埋め等の

¹家庭内に存在する家電製品や防犯システム等を一律に制御するシステムである。ホームサーバとはその各種センサを制御されているサーバのことである。

被災者を特定し、救助要請マップを作成してシミュレーション実験を行い、研究の有効性を示した。しかし、ここではホームサーバごとに管理しているため、人数に誤差が出ることや使用する多数の機器に対し電力を供給し続けなければならないことが問題となっている。

また、複数人の位置情報を用いた遠隔地での情報共有に関して、宗森らはPDA(携帯情報端末)とGPSと携帯電話を使用し、位置情報を用いて遠隔地間で行う鬼ごっこをはじめ複数種の電子鬼ごっこの実験を行っている [4]。位置情報を変換することで離れた場所でも問題なく電子鬼ごっこを行い、実験の結果では位置情報を柔軟に扱ったサービスが受け入れられている。

第2章 提案

現状抱えている問題点を踏まえて、本研究で提案するシステムについて記述する。

2.1 現状

災害時等を想定した際の離れている人同士の安否確認等の情報の相互確認についての現状を記述する。

2.1.1 災害時の逃げ遅れによる被害

渡邊らは、災害時に住民の孤立や逃げ遅れの二次災害による犠牲者の発生を問題とし、最短避難経路提示システムを開発した [9]。そこでは、従来の災害時の伝言板等のシステムでは携帯電話と PC との連携がとれず、避難者の立場が考慮されていないことが問題視されていた。そこで、避難所や交差点に ID とグループ番号を付加することで、携帯電話を活用し最短の避難経路に誘導するシステムを実現した。また、GPS 内蔵型携帯電話が使用できない状況では Wi-Fi 付きノート PC を用い、Google 社から Wi-Fi のアクセスポイントと IP アドレスを基に測位された現在地を利用することで実測値との誤差を 5% 以下で実現した。ここから、災害時であっても正確に位置情報を発信・共有可能であることが示されている。

2.1.2 避難後の安否確認

2016 年の熊本地震におけるライフラインの復旧に関して能島は阪神淡路大震災や東日本大震災と比べ大幅に短縮され、迅速さにおいてはほぼ飽和状態に達していると述べている [3]。しかし、ライフラインの完全な復旧には 2 週間を要しているのが現状であり、その間は十分に外部との連絡を取れない人がいるのが事実である。

2.1.3 既存のアプリケーション

現在、Apple 社が提供するクラウドサービス “iCloud” の機能である “友だちを探す” や Google が提供している地図・ローカル検索サービス “Google マップ” の機能である “現在地を共有” 等の位置情報の共有を目的としたアプリケーションが存在する。しかし、これらのアプリケーションは利用する度に共有相手の設定や時間の指定や設定の変更が必要になる。また、個人間の共有を目的にしているため、複数人のグループ等で共有をする際はグループ内の全員がお互いに個人間の設定をする必要がある。そのため、表 2.1 のように人数が増えるほど必要な設定数が増えていくことになる。

また、これらのアプリケーションではお互いに iCloud や Google アカウント等の外部の登録が必要な固有アカウントを把握している利用者の情報しか得ることができないため、情報の共有に制限がある。

2.2 提案するシステム

そこで、本研究では上述の問題点を踏まえて以下のような機能を導入したシステムを提案する。

1. システム利用者の位置情報を把握できる機能

本システムの根本的な機能として、利用者がシステムに送信した位置情報のデータをブラウザ上のマップにユーザの設定する分単位で更新して表示する。

2. 一定範囲の地域内の分布を表示するマップ作成アプリケーション

災害時に利用者の避難状況を把握できるようにするため、設定した地域内の利用者の分布をブラウザ上のマップに表示する。ここでは個人は特定せず、あくまで分布のみの表示とする。

3. 個別アカウントによるグループ管理

利用者個人を判別するためのアカウントを用いて、複数人単位で相互の情報管理を可能にする。既存のアプリケーションにおける問題点を踏まえ、複数人での共有を簡潔な作業で行えるようにする。

4. 複数人グループで情報共有する利用者のマップ作成アプリケーション

上記の複数人グループ内で、個人を判別した上で位置情報を相互に設定した数分単位で更新・管理できるマップをブラウザ上で実現する。

| 利用人数 | 既存のシステム | 本システム |
|------|------------|-------|
| 2 | 2回 | 2回 |
| 3 | 6回 | 3回 |
| 4 | 12回 | 4回 |
| n | $n(n-1)$ 回 | n回 |

表 2.1: システム利用時に必要な設定数の差

第3章 システムの構築環境

本研究で開発するシステムの環境について記述する。

3.1 データの正規化

今回は関係データベース (DBMS) でデータを管理するため、データを正規化する必要がある。正規化とは、データを効率良く扱うために特定のルールにしたがって整理することである。以下に正規化の手順について本システムで利用したデータの一部を用いて記述する。

非正規形

整理されていない生の状態のデータのことを非正規形と言う。データベースでデータを管理する際、このように一つの項目 (フィールド) 内に複数のデータが格納されていても複数のデータと認識することはできない。これらのデータをデータベースが計算機で処理しやすい形式にすることが正規化である。

| | | | |
|----|-----------|----|---|
| 太郎 | hogehoge | 鶴岡 | A グループ 38.4, 138.5 12:00 B グループ 38.6, 138.1 13:00 |
| 花子 | gehogeho | 酒田 | A グループ 38.1, 138.1 13:05 |
| 鈴木 | pasuwa-do | 鶴岡 | B グループ 38.5, 138.5 14:30 |

表 3.1: 生のデータ

第1正規化

非正規形において繰り返していたり、フィールド内に複数格納されていたりするデータを別の行 (レコード) に分離し、繰り返しを排除する。

| name | pass | group | local | lat | lng | time |
|------|-----------|--------|-------|------|-------|-------|
| 太郎 | hogehoge | A グループ | 鶴岡 | 38.4 | 138.5 | 12:00 |
| 太郎 | hogehoge | B グループ | 鶴岡 | 38.6 | 138.1 | 13:00 |
| 花子 | gehogeho | B グループ | 酒田 | 38.1 | 138.1 | 13:05 |
| 鈴木 | pasuwa-do | B グループ | 鶴岡 | 38.5 | 138.5 | 14:30 |

表 3.2: 繰り返しを排除した形

更に、テーブルごとにデータを一意に識別するための項目を設定する。このようなコードのことを主キーという。主キーは複数カラムから構成されることもあり、その場合は複合主キーと言う。

第2正規化

表3.2の第1正規形から部分関数従属性を排除する。関数従属とは、ある属性 X を決めると他の属性 Y が一意に決まるような状態のことである。この時、X を決定項、Y を被決定項呼ぶ。部分関数従属とは、このときの被決定項が複数の決定項を持っている状態のことである。表3.2の場合、name が決まれば pass が決まるが、他の項目が必ずしも決定しない。そこで、例えば name を主キーとする pass だけをまとめたテーブルを新規に作成する。このように第2正規化をすることでテーブルごとの列の数を減らし、データの管理をより単純化する。

| name | pass |
|------|-----------|
| 太郎 | hogehoge |
| 花子 | gehogeho |
| 鈴木 | pasuwa-do |

表 3.3: パスワードテーブル

| name | local |
|------|-------|
| 太郎 | 鶴岡 |
| 花子 | 酒田 |
| 鈴木 | 鶴岡 |

表 3.4: 地方テーブル

| name | group | lat | lng | time |
|------|--------|------|-------|-------|
| 太郎 | A グループ | 38.4 | 138.5 | 12:00 |
| 太郎 | B グループ | 38.6 | 138.1 | 13:00 |
| 花子 | B グループ | 38.1 | 138.1 | 13:05 |
| 鈴木 | A グループ | 38.5 | 138.5 | 14:30 |

表 3.5: ユーザ情報テーブル

第3正規化

第2正規形から推移的関数従属性を排除する。推移的関数従属性とは、被決定項または決定項と被決定項の組み合わせによって他の被決定項が一意に決まることを指す。今回の場合、表3.1において、name と group が決まればその他のデータが決まるため、別のテーブルを新たに作成する。検索の効率を考慮して部分的に正規化の程度を敢えて落とした状態のままにする場合もある。最終的には表3.6 3.9の状態になる。

3.2 作業環境

開発は以下の環境で行う。

- Ruby 2.3.0
- Leaflet.js 1.3.4
- SQLite3

| name | pass |
|------|-----------|
| 太朗 | hogehoge |
| 花子 | gehogeho |
| 鈴木 | pasuwa-do |

表 3.6: パスワードテーブル

| name | local |
|------|-------|
| 太朗 | 鶴岡 |
| 花子 | 酒田 |
| 鈴木 | 鶴岡 |

表 3.7: 地方テーブル

| name | group | lat | lng |
|------|--------|------|-------|
| 太朗 | A グループ | 38.4 | 138.5 |
| 太朗 | B グループ | 38.6 | 138.1 |
| 花子 | B グループ | 38.1 | 138.1 |
| 鈴木 | A グループ | 38.5 | 138.5 |

表 3.8: 位置情報テーブル

| name | group | time |
|------|--------|-------|
| 太朗 | A グループ | 12:00 |
| 太朗 | B グループ | 13:00 |
| 花子 | B グループ | 13:05 |
| 鈴木 | A グループ | 14:30 |

表 3.9: 利用時刻テーブル

第4章 システムの設計

本システムの設計について記述する。

4.1 RDBMS によるデータ管理

前章で述べた通り本システムではRDBMSで各データを管理する。そこからRubyプログラムやCGIからSQL文を実行することで必要なデータを取り出す形式である。本システムで扱うデータのER図は図4.1の通りである。

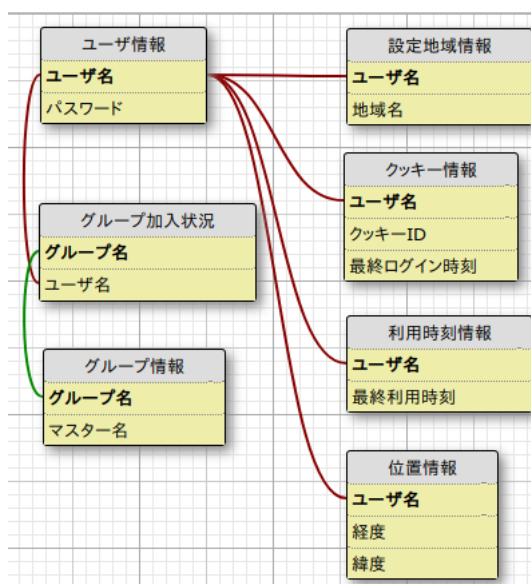


図 4.1: 本システムで扱うデータのER図

4.2 個別アカウントの管理

本システムを利用するために必要なアカウント周辺について記述する。

4.2.1 アカウントの制約に関して

本システムは外部の固有アカウントを必要としない設計にしている。そのため、システムを利用するためのアカウントを最初に作成する必要がある。メールアドレス等の登録を用いないため、ユーザ名を直接ユニークキーとする。

4.2.2 アカウントの認証期限に関して

本システムでは後述の Ajax による非同期通信を行う以外に、複数ページを経由したり、システムから離れたりした際にもある程度の間ログイン情報を保持する必要がある。そこで、Cookie を利用することでログイン情報を保持する。セキュリティ上の問題を踏まえて、永久に保持するのではなく情報は 24 時間で破棄する。

4.3 JavaScript を用いた動的な Web の作成

本システムでは主に JavaScript を用いて動的な Web ページを作成する。

4.3.1 Leaflet.js を用いたマップ作成

Leaflet.js とは Web 上で地図を作成・表示するためのオープンソースの JavaScript ライブラリである。GeoJSON からデータを読み込み、ポップアップ等の機能を利用したマップを作成することができる。本システムではこれを用いて OpenStreetMap の地図をベースにしたマップを作成する。

4.3.2 Ajax による非同期通信

本システムで利用時に更新する情報は一部であるため、非同期通信で更新する必要がある情報のみをサーバにリクエストし、更新する。本システムでは以下のような流れで利用している。

1. ユーザ自身の位置情報を送信する。
2. CGI で SQL 文を実行し、データベースからその時の利用者に必要なデータを JSON 形式で返す。
3. 受け取ったデータをもとにマップを作成する。

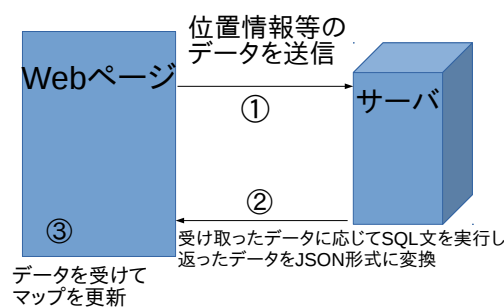


図 4.2: 本システムでの非同期通信の例

第5章 システムの概観と実行

実際に構築したシステムの概観と実行する行程について記述する。

5.1 本システムの概観

本システムを利用する際の流れは以下の通りである。

1. アカウントの作成・ログイン

本システムを利用するための情報を入力しアカウントを作成し、ログインする。

2. グループの作成・加入・選択

指定したユーザ同士で構成されるグループ周辺の設定をする。

3. 利用端末から情報を送信・取得

ユーザ自身の位置情報・利用時間等の情報を DBMS に送信すると同時に最新の情報を取得する。

4. 取得したデータが反映されたマップの閲覧

取得したデータを元に目的に応じて作成されたマップを閲覧する。

なお、目的がユーザ自身の情報の更新に限る場合グループを選択する必要はなく、自身の情報のみを更新する機能を利用することができる。利用時の概念図は図 5.1 のようになる。各行程の詳細な手順は次章以降に記述する。

5.2 個人アカウントの管理

本システムを利用する際、利用者を特定するための個人アカウント周辺の機能に関して記述する。

5.2.1 個人アカウント作成

ユーザ名とパスワードと地域の情報を設定し、利用者個人を判別するためのアカウントを作成する機能である。アカウント作成時にはユーザ名、パスワード、地域の3項目を入力する。ユーザ名を個人を判別するためのデータとするため、ユーザが既存のアカウントと重複していた際にはエラーを表示する。

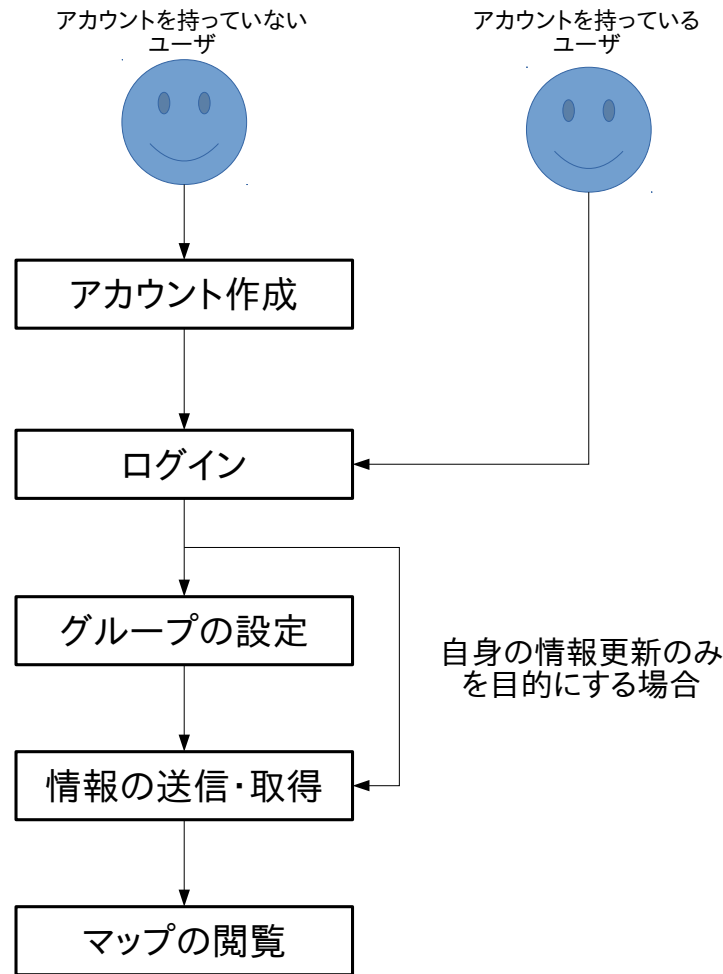


図 5.1: システム利用時の概念図

5.2.2 ログイン

作成したアカウントのユーザ名とパスワードを入力し、システムにログインするための機能である。ユーザのログイン情報を保持するため、ログインと同時にクッキー ID を取得する。認証期限は 24 時間に設定している。

5.3 位置情報を利用した情報共有

本システムのメインとなる機能である。利用者が意図的に情報を送信したり、送信情報をマップ上に表示して共有したりする機能と、そこで得た情報を元に地域ごとに利用者の分布を表示する機能の大きく 2 つに分かれる。

5.3.1 個人の位置情報の送信

アカウントを所有するユーザがデータベースに自分の利用情報をデータベースに送信する機能である。ここで得たデータを元に後述の地域ごとの分布マップを作成する (5.3.3 章)。



CREATE NEW USER'S DATA

名前を入力:

パスワードを入力:

出身地:

図 5.2: ログイン画面

5.3.2 少人数グループでの利用

特に少人数単位のグループ内で、個人を特定した上で相互に情報を共有するシステムである。災害時に家族等少人数のコミュニティ内でお互いの情報を把握するための機能である。この機能は、災害時のみならず、グループワーク等複数人の位置情報を相互に把握したい状況で利用することも目的とする。この機能を利用する際に利用者が送信したデータも 5.3.3 章のマップ作成に使用する。

グループ作成・管理

少人数グループでの利用をするためのグループを作成・管理する機能である。以下の3つの機能からなる。

- 新規グループの作成
- グループへの加入
- 加入済みグループからの脱退

グループ単位のマップの作成

5.3.2 章の機能を用いて作成したグループ単位で利用するシステムである。グループメンバーが最後に利用した位置情報、名前、時刻をマップ上のピンで表示する。

5.3.3 設定地域内の分布表示

利用者が 5.3.1 章や 5.3.2 章のシステムを利用した際に発信される情報をもとに各利用者が設定した地域分布マップを作成するシステムである。ここでは利用者にプライバシーやセキュリティ面への考慮からユーザ名等は表示せず、地域内での分布のみを表示する。図 5.6 は山形県酒田市の人口密度をもとにマップを擬似的に作成したものである。

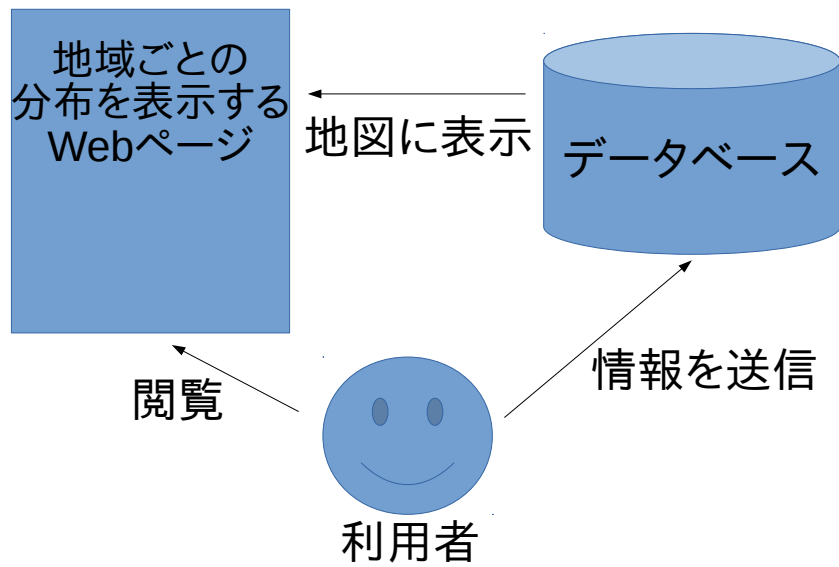


図 5.3: 個人利用時の概念図

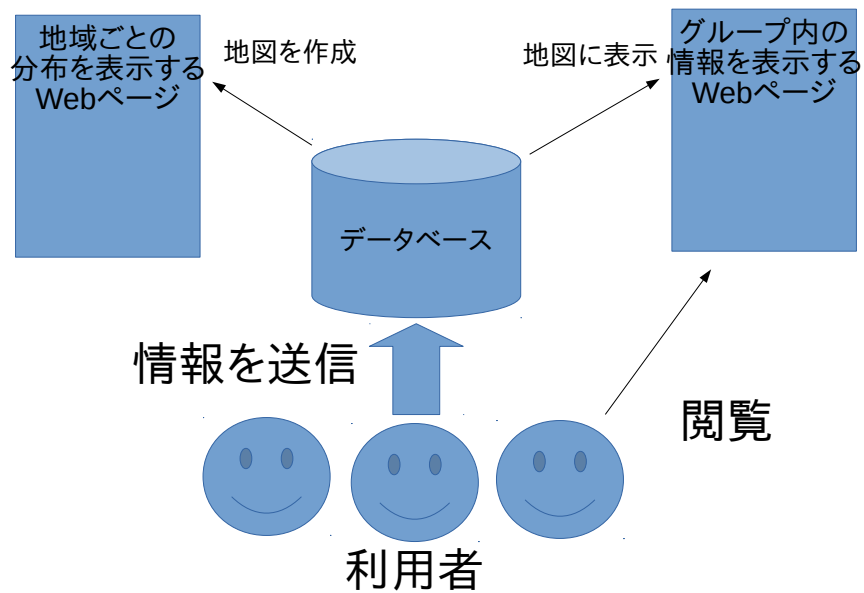


図 5.4: グループ利用時の概念図



図 5.5: グループ利用時の画面例

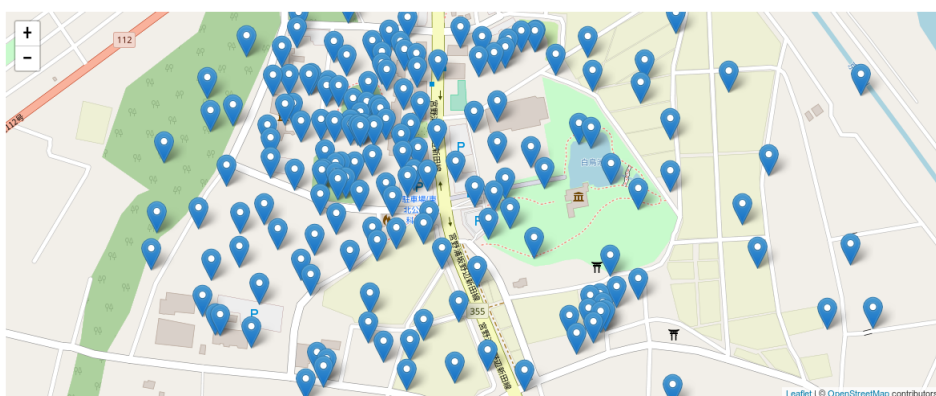


図 5.6: 地域の分布表示の例

第6章 結論とシステムへの評価

本システムへの結論・評価としては以下のようなものが挙げられる。

6.1 結論

アカウントを作成した利用者から発信された情報を元に地域の住民の避難状況等をリアルタイムで把握するためのマップを作成する Web アプリケーションシステムを構築することができた。また、それにともない複数人のグループ単位で情報を共有できるシステムを構築した。

6.2 システムへの評価と展望

本システムへの評価と展望について記述する。

6.2.1 有事の際の情報リセット

本システムは災害時のみならず、平常時からの利用も考慮している。それに伴い、地震等が発生する以前のデータが残っていると地震後に全てのユーザが情報を更新しなければ正確な避難情報を得ることができなくなるという事態に陥る。そこで、Web スクレイピング¹を用いて常時気象庁からデータを取得し、災害発生時にその時点よりも前のデータを消去する等の処理を加えることで災害時に対応可能にする必要がある [10]。

6.2.2 グループ参加の有無と個人の特定

本システムを個人でのみ利用しているユーザに関して、個人を特定する方法が完全でない状態になっている。匿名性の保持と言う点では問題はないが、災害時を考えた際に個人の特定の必要性に関して懸念される。

6.2.3 グループ管理のセキュリティ観点

現段階の本システムは、ユーザであれば誰でも存在する全てのグループに自由に入出力できるようになっている。グループに ID を割り振る、特定のメンバーに承認周りの権限を付与する等してグループ管理周辺のセキュリティについて整えていく必要がある。

¹Web サイトから Web ページの HTML データを収集して、特定のデータを抽出、整形し直すこと。

6.2.4 同時利用時のキャパシティ

5.3.3 章から、個人の特定を目的としない場合の避難所から離れている利用者のばらつき等の視認性においては問題なく見えるが、災害時に多くの利用者が同時に利用した場合、またより人口の多い場合などの動作に関してプログラムを用いて模擬的に検証する必要がある。

参考文献

- [1] 内閣府. 災害情報・防災情報のページ - 内閣府, 2018-11-10. <http://www.bousai.go.jp/updates/>.
- [2] 金野達也, 戸羽開, 柴田義孝, 橋本浩二. 津波などの2次災害を考慮した災害時避難経路提示システム. 第76回全国大会講演論文集, Vol. 2014, No. 1, pp. 613–614, mar 2014.
- [3] 能島暢呂. 熊本地震における供給系ライフラインの被害と復旧: 震災から得られた教訓と残された課題 (特集 平成28年熊本地震(2) 住民生活). 消防防災の科学, No. 127, pp. 30–34, 2017.
- [4] 宗森純, 宮内絵美, 牟田智宏, 吉野孝, 湯ノ口万友. 電子鬼ごっこ支援グループウェアの開発と適用. 情報処理学会論文誌, Vol. 42, No. 11, pp. 2584–2594, nov 2001.
- [5] 宗森純, 上坂大輔, タイミンチー, 吉野孝. 位置情報を用いた汎用双方向ガイドシステム xexplorer の開発と適用. 情報処理学会論文誌, Vol. 47, No. 1, pp. 28–40, jan 2006.
- [6] 坪田大吾, 花田経子. ios アプリケーションにおける個人情報の取り扱いに関する調査と考察. 第76回全国大会講演論文集, Vol. 2014, No. 1, pp. 605–606, mar 2014.
- [7] DailyTimer.net. いまどこ?オンラインマニュアル, 2018-12-20. <http://file.blog.fc2.com/nekonotesoft/manual/imadoco-jp.html>.
- [8] 松崎頼人, 榎原博之. 地震時におけるスマートホームを利用したアドホックネットワークー生き埋め被災者のための救助要請 map データの配信. 情報処理学会論文誌数理モデル化と応用 (TOM) , Vol. 6, No. 1, pp. 64–78, mar 2013.
- [9] 渡邊博之, 成田祐一, 大山勝徳, 加瀬澤正, 武内惇, 竹中豊文. モバイル端末を活用した災害時最短避難経路提示システムの開発. 情報処理学会論文誌, Vol. 53, No. 7, pp. 1768–1773, jul 2012.
- [10] 一恭小島, 高広奥村. 2115 web スクレイピングによる温熱データと居住者の温冷感申告との関係性 (福祉機械・ヒューマンインターフェース (2)). 機素潤滑設計部門講演会講演論文集, Vol. 2014, pp. 129–130, 2014.