

内部設計書

ドローン宅配事業者支援システム
Aero Net

土佐山田IT株式会社

久保田 天治
寺内 俊輔

塩澤 康志
林 晃太郎

蟬 祐介
松本 吏司

目次

1	はじめに	2
2	システム概要	2
3	動作環境	2
4	開発環境	3
4.1	ファイル構成	3
4.2	コーディング規約	3
5	機能一覧	4
5.1	管理者向け機能	5
5.2	事業者向け機能	5
5.3	利用者向け機能	6
6	遷移図の定義	6
7	モジュール一覧	7
8	各モジュール定義	8
9	コンポーネント一覧	17
10	各コンポーネント定義	18
11	貢献度	30

1 はじめに

本書は、ドローン宅配事業者支援システム Aero Net の内部設計書である。本書では、動作環境及び開発環境を示し、本システムの機能におけるモジュール及びコンポーネントの詳細を示す。

2 システム概要

本システムでは、図 1 及び以下に示すドローン宅配を支援する。

- 送り主である利用者が、管理者に対して宅配依頼を行う。
- 管理者が事業者に対し、集荷・配送依頼を行う。
- 事業者がドローンを用い、依頼を受けた荷物の集荷を行う。
- 受け取り主である利用者が事業者から離れている場合、トラックを用いて荷物を受け取り主の最寄りの事業者へ配送する。
- 事業者がドローンを用い、荷物を受け取り主へ配達する。

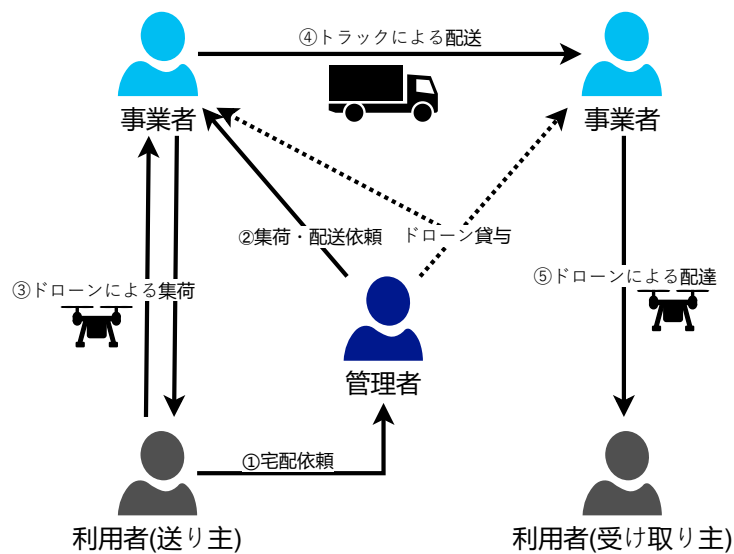


図 1 ドローン宅配の流れ

3 動作環境

本システムを使用するためには、以下の表 1 に示す動作環境が必要である。また、サーバの動作環境を表 2 に示す。

表 1 動作環境

OS	Windows10 以上
CPU	1GHz 以上
メモリ	4GB 以上
ブラウザ	Firefox

表 2 サーバの動作環境

サーバ	AWS(EC2 t2.micro)
-----	-------------------

4 開発環境

本システムの開発環境を、表 3 に示す。

表 3 開発環境

プラットフォーム	Docker
フレームワーク	Laravel
バージョン管理	Git
使用言語	PHP HTML CSS JavaScript

4.1 ファイル構成

4.2 コーディング規約

本システムでは、PHP 標準勧告である PSR-12 に従う。

- 命名規則
 - － 変数名
 - * スネークケースを採用
 - － 定数名
 - * 全部大文字
 - * スネークケースを採用
 - － クラス名
 - * StudlyCaps 記法 (先頭と単語の区切りを大文字) を採用
 - － メソッド名
 - * camelCase 記法 (単語の区切りを大文字) を採用
 - － 予約語
 - * 小文字で書く
- コーディングスタイル
 - － インデント

- * スペース 4 つとする
- * タブによるインデントは行わない
- 括弧
 - * クラスやメソッドの波括弧は新しい行に配置する
 - * メソッドや関数や制御構造の開き括弧は同じ行にスペースを一つ入れた後配置する
- 行
 - * 名前空間定義の後に空行を挟む
 - * use 定義ブロックの後に空行を挟む
 - * ファイルの最終行には空行をいれてはならない
 - * 1 行の長さの目安は 80 文字以内
 - * 1 行の長さは 120 文字を超えてはならない
 - * 行の最後には空白を入れてはならない
 - * 可読性を上げるための空行は許可する
 - * 1 行に複数のステートメントがあってはならない
- 文字列
 - * シングルクォートで文字列は囲む
- コメント
 - * 文章を使う
 - * 命令形を使う
 - * クラス、メソッド、プロパティに対しては、ドキュメンテーションコメント (`/** ... */`で囲まれた説明) を付ける
- クラス
 - * StudlyCaps 記法 (先頭と単語の区切りを大文字) を採用
- 演算子
 - * 演算子の周囲にスペースを何個使用しても構わない
 - * 単項演算子：インクリメント/デクリメント演算子は演算子とオペランドの間にスペースを挟んではない
 - * 単項演算子：型キャスト演算子は括弧の内側にスペースがあってはならない
 - * 二項演算子：代数演算子、比較演算子、代入演算子、ビット演算子、論理演算子、文字列演算子、型演算子は前後に 1 個以上のスペースを置く
 - * 三項演算子：`?:` それぞれの前後に 1 個以上のスペースを置く
 - * 三項演算子：三項演算子の真ん中のオペランドが省略されている場合、演算子は二項の比較演算子と同様のスタイルルールを適用する

5 機能一覧

本システムには、管理者向け機能、事業者向け機能、利用者向け機能が存在しており、それぞれの機能を以下に示す。

5.1 管理者向け機能

管理者向け機能を以下に示す。

- ログイン機能
- ログアウト機能
- 事業者一覧閲覧機能
- 請求書送付機能
- 事業者情報詳細閲覧機能
- 事業者情報編集機能
- 事業者支払い情報詳細閲覧機能
- 事業者支払い情報詳細編集機能
- 事業者ドローン情報詳細機能
- 事業者ドローン情報編集機能
- 利用者一覧閲覧機能
- 請求書送付機能
- 利用者情報詳細閲覧機能
- 利用者情報編集機能
- 利用者支払い情報詳細閲覧機能
- 利用者支払い情報詳細編集機能
- 事業者統計情報表示機能
- 利用者統計情報表示機能
- 宅配依頼一覧表示機能
- 宅配仕事割り振り機能
- 事業者ドローン情報編集
- ドローン登録機能

5.2 事業者向け機能

事業者向け機能を以下に示す。

- ログイン機能
- ログアウト機能
- 事業者情報編集機能
- 配達完了通知機能
- 使用ドローン登録機能
- 子アカウント一覧表示機能
- 子アカウント発行機能
- 子アカウント削除機能
- 子アカウント編集機能

- ドローン種類一覧機能
- ドローンの修理依頼機能
- ドローンの機体トラブル報告
- 退会機能
- 所持ドローン一覧機能

5.3 利用者向け機能

利用者向け機能を以下に示す。

- ログイン機能
- ログアウト機能
- 利用者会員登録機能
- 利用者会員情報編集機能
- 宅配場所登録機能
- 宅配依頼機能
- お気に入り一覧表示機能
- お気に入りからデータ参照機能
- 受け取り完了通知機能
- お気に入り登録機能
- 退会機能

6 遷移図の定義

図 2 に、コンポーネント及びモジュールの処理順序を示すダイアグラムで用いる記法を示す。

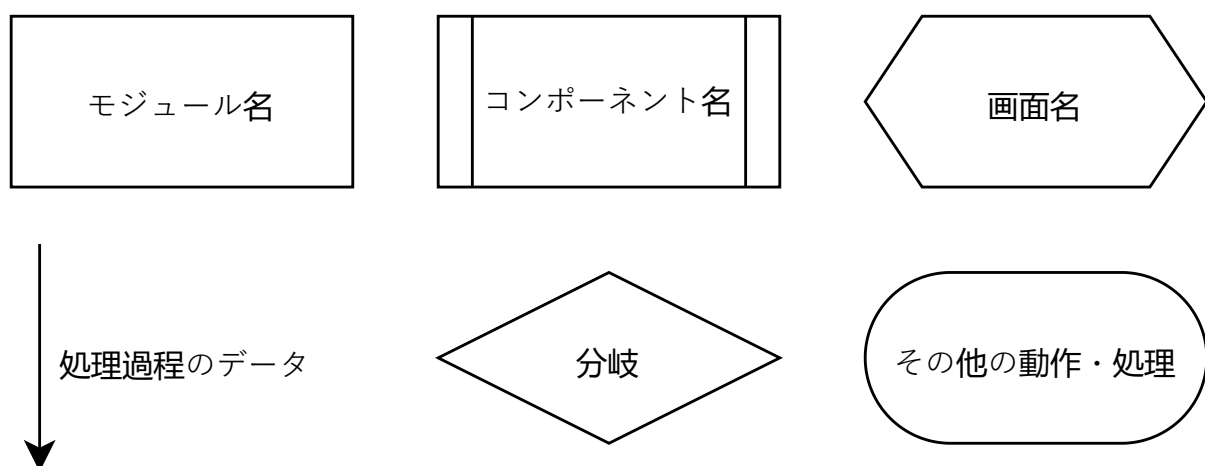


図 2 ダイアグラムで用いる図形とその意味

処理の流れは矢印で示し、処理に必要なデータが存在する場合は矢印付近に表記を行う。各ダイアグラム

は、該当コンポーネント又はモジュールの呼び出しから始まり、処理終了で終わるものとする。

7 モジュール一覧

- ログインモジュール
- ログアウトモジュール
- 一覧表示モジュール
- 詳細表示モジュール
- 登録モジュール
- 編集モジュール
- 分析モジュール
- 退会モジュール

8 各モジュール定義

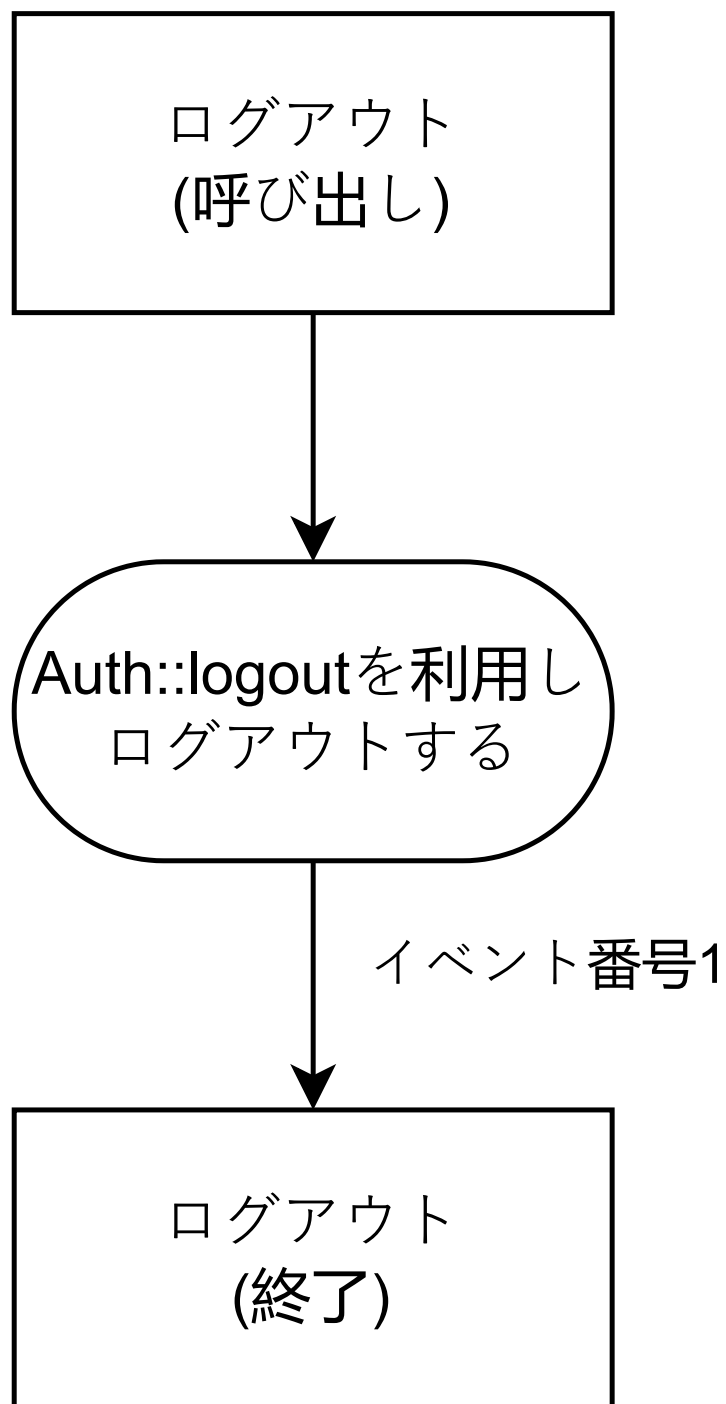


図3 ログインモジュール (バック)

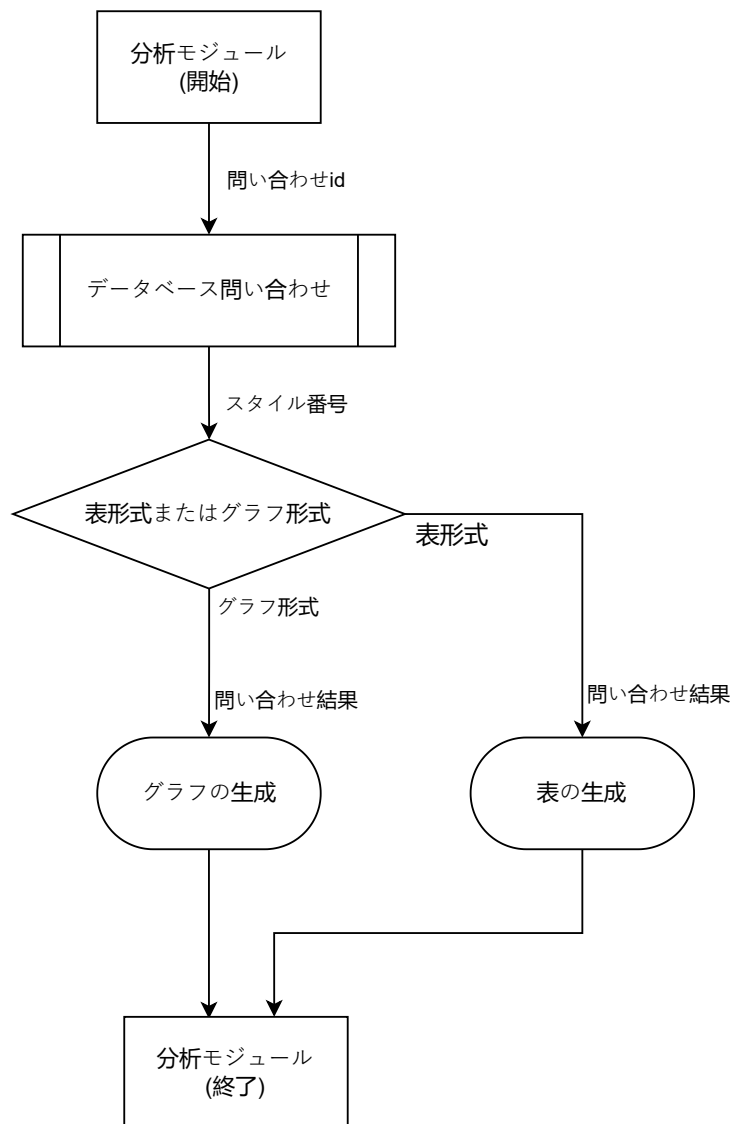


図4 ログアウトモジュール (バック)

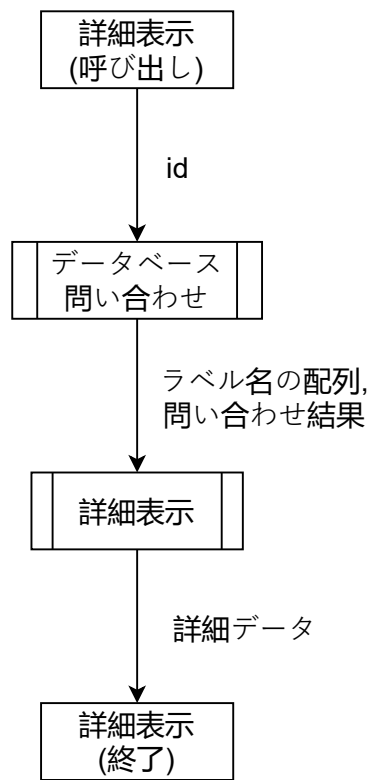


図 5 一覧表示モジュール

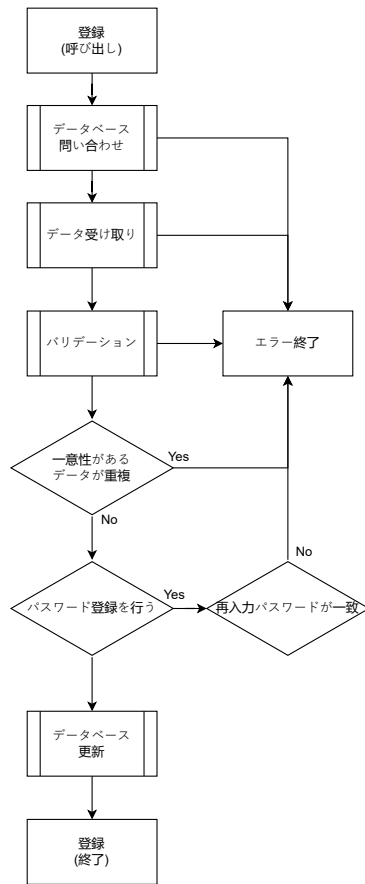


図 6 詳細表示モジュール

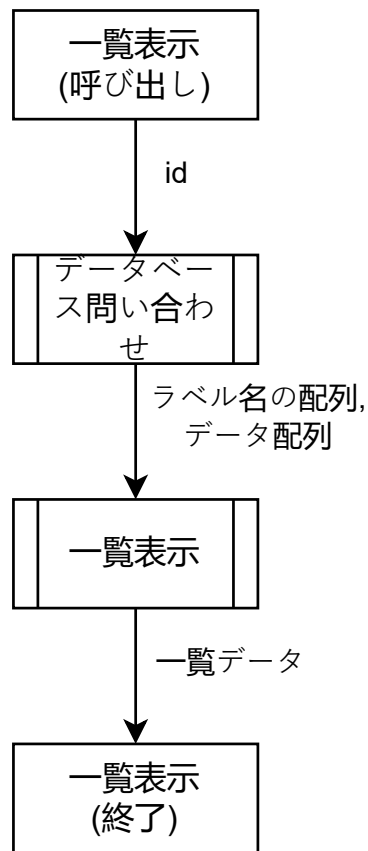


図 7 登録モジュール

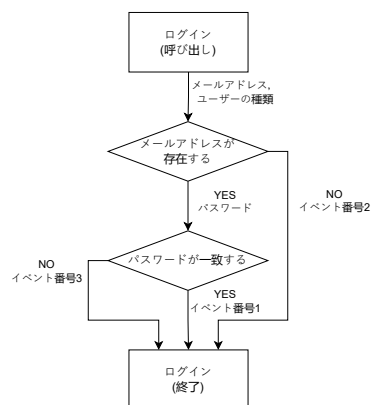


図 8 編集モジュール

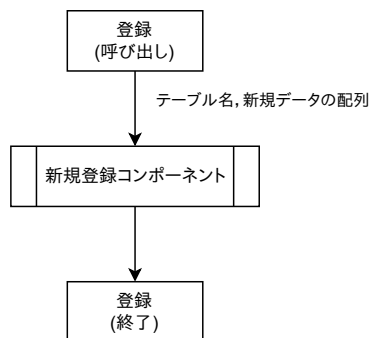


図 9 分析モジュール

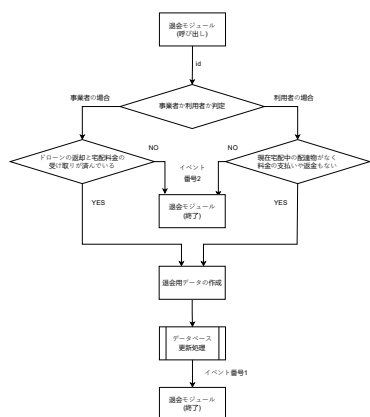


図 10 退会モジュール

モジュール定義				
共通情報	システム名	Aero Net	バージョン	1.4
	工程名	内部設計	作成日付	2023/12/08
	作成者	林晃太郎	更新日付	2023/12/15
書誌情報	モジュールID	2		
	名称	ログアウトモジュール		
	概要	本システムからログアウトする		
入力				
なし				
出力				
処理終了イベント				
正常終了 -> イベント番号1				
異常終了 -> イベント番号-1				
処理手順				
1. LaravelのAuth::logout関数を利用し、 ログアウトを行う。				
2. 正常終了の場合、 イベント番号1を返す。				
3. 異常終了の場合、 イベント番号-1を返す。				

図 11 ログインモジュール

モジュール定義				
共通情報	システム名	Aero Net	バージョン	1.4
	工程名	内部設計	作成日付	2023/12/08
	作成者	松本吏司	更新日付	2023/12/15
書誌情報	モジュールID	7		
	名称	分析モジュール		
	概要	データの可視化をグラフ等で行う		
入力				
第1引数：分析するデータ名(表示用)				
第2引数：スタイル番号				
第3引数：データベース問い合わせ用id				
出力				
処理終了イベント				
正常終了 -> イベント番号1 及び 生成物				
異常終了 -> イベント番号-1				
処理手順				
1. 第3引数を用い、データベース問い合わせコンポーネントを呼び出す。				
2a. 入力されたスタイル番号が表形式のものを示す場合、3へ進む。				
2b. 入力されたスタイル番号がグラフ形式を示す場合、5へ進む。				
3. 分析するデータを表形式で生成する。キャプションを第1引数とする。				
4. 生成した表とイベント番号1を返し、終了する。				
5. 分析するデータをグラフ形式で生成する。キャプションを第1引数とする。				
6a. 生成したグラフとイベント番号 1 を返し、終了する。				
6b. 表またはグラフの生成に失敗した場合はイベント番号 -1 を返し、終了する。				

図 12 ログアウトモジュール

モジュール定義				
共通情報	システム名	Aero Net	バージョン	1.4
	工程名	内部設計	作成日付	2023/12/08
	作成者	塩澤康志	更新日付	2023/12/15
書誌情報	モジュールID	4		
	名称	詳細表示モジュール		
	概要	詳細情報を表示する		
入力				
第1引数：データのラベル(カラム名)の入った配列				
第2引数：id				
出力				
処理終了イベント				
正常終了 -> イベント番号1及び詳細データ				
DBアクセスエラー -> イベント番号-1				
処理手順				
1. idを引数とし、データベース問い合わせコンポーネントを呼び出す。				
2. ラベル名の配列と1の結果を引数とし、詳細表示コンポーネントを呼び出す。				
3. 正常に終了すればイベント番号1と詳細データを返し、エラーが起こればイベント番号-1を返す。				

図 13 一覧表示モジュール

モジュール定義				
共通情報	システム名	Aero Net	バージョン	1.3
	工程名	内部設計	作成日付	2023/12/08
	作成者	久保田天治	更新日付	2023/12/15
書誌情報	モジュールID	6		
	名称	編集モジュール		
	概要	DBに登録されたデータを編集する		
入力				
なし				
出力				
処理終了イベント 正常終了 -> イベント番号1 DBアクセスエラー -> イベント番号-1 入力規則エラー -> イベント番号-2 パスワード再入力エラー -> イベント番号-3				
処理手順				
1.登録するデータのバリデーションチェックを行う 1.1.バリデーションチェックでエラーを返した際、イベント番号-2を呼び終了 2.引数として与えられたデータをデータベースに登録する 2.1.データベースへのアクセスに失敗した場合、イベント番号-1を呼び終了 2.2.一意性のあるデータを登録する際には、データベースに同一のデータがないか調べる 2.2.1.同一のデータがある場合、イベント番号-4を呼び終了 2.3.パスワードを登録する際には、再入力データと比較を行う 2.3.1.入力されたパスワードと再入力されたパスワードが異なる場合、イベント番号-3を呼び終了 3.イベント番号1を呼び終了				

図 14 詳細表示モジュール

モジュール定義				
共通情報	システム名	Aero Net	バージョン	1.4
	工程名	内部設計	作成日付	2023/12/08
	作成者	蟬 祐介	更新日付	2023/12/15
書誌情報	モジュールID	3		
	名称	一覧表示モジュール		
	概要	一覧画面を表示する		
入力				
第1引数：id				
第2引数：ラベル名の配列				
出力				
処理終了イベント				
正常終了 -> イベント番号1と一覧データ				
DBアクセスエラー -> イベント番号-1				
処理手順				
1. idをつくる。				
2. idに応じてデータベース問い合わせコンポーネントを呼び出す。				
3. ラベル名の配列から、一覧表示コンポーネントを呼び出す。				
4. イベント番号1と一覧データを返す。				

図 15 登録モジュール

モジュール定義				
共通情報	システム名	Aero Net	バージョン	1.4
	工程名	内部設計	作成日付	2023/12/08
	作成者	林晃太郎	更新日付	2023/12/15
書誌情報	モジュールID	1		
	名称	ログインモジュール		
	概要	本システムへのログインを認証する		
入力				
第1引数：メールアドレス				
第2引数：パスワード				
第3引数：ユーザーの種類				
出力				
処理終了イベント				
正常終了 -> イベント番号1				
メールアドレスが登録されていない-> イベント番号2				
パスワードの不一致 -> イベント番号3				
異常終了-> イベント番号-1				
処理手順				
1.DBにアクセスし、メールアドレスとパスワードを取得する。				
2.Laravelの機能でメールアドレスとユーザーの種類で絞り込む。				
2a.メールアドレスが存在しないならイベント番号2を返し、終了する。				
3.入力のパスワードとDBから取得したパスワードを比較する。				
3a.等価であればイベント番号1を返し、終了する。				
3b.等価でなければ、イベント番号3を返し、終了する。				
4. 正常終了の場合、イベント番号1を返す。				
5. 異常終了の場合、イベント番号-1を返す。				

図 16 編集モジュール

モジュール定義				
共通情報	システム名	Aero Net	バージョン	1.4
	工程名	内部設計	作成日付	2023/12/08
	作成者	久保田天治	更新日付	2023/12/15
書誌情報	モジュールID	5		
	名称	登録モジュール		
	概要	引き数をDBに登録する		
入力				
第1引数：テーブル名				
第2引数：新規データの配列				
出力				
処理終了イベント				
正常終了 -> イベント番号1				
異常終了 -> イベント番号-1				
処理手順				
1. テーブル名，新規データの配列を受け取る。				
2. 新規登録コンポーネントを呼び出す。				
3a. 新規登録に成功した場合はイベント番号1を返し，終了する。				
3b. 新規登録に失敗した場合はイベント番号-1を返し，終了する。				

図 17 分析モジュール

モジュール定義				
共通情報	システム名	Aero Net	バージョン	1.4
	工程名	内部設計	作成日付	2023/12/08
	作成者	寺内俊輔	更新日付	2023/12/15
書誌情報	モジュールID	8		
	名称	退会モジュール		
	概要	事業者と利用者が退会するためのモジュール		
入力				
第1引数：事業者idもしくは利用者id				
出力				
処理終了イベント 退会処理完了 -> イベント番号1 退会処理不可 -> イベント番号2 エラー -> イベント番号-1				
処理手順				
1. idを受け取りそれが事業者のものか利用者のものか判定する 2a. 事業者のものである場合,ドローンの返却と宅配料金の受け取りが済んでいるかを確認する.もしあった場合退会処理不可を行う. 2b. 利用者のものである場合,現在宅配中の配達物がないか,料金の支払いや返金がないか確認する.もしあった場合退会処理不可を行う. 3. 退会完了処理を行う.				

図 18 退会モジュール

9 コンポーネント一覧

- データの入力フォーム
- データの送信 (フロントエンド)
- ボタン表示
- データの一覧表示
- データの詳細表示
- ヘッダーの表示
- サイドバーの表示
- 絞り込み
- グラフ表示
- データの受け取り
- データ送信 (バックエンド)
- データベース問い合わせ
- データベース更新
- データベースに新規追加
- バリデーション (入力規則確認)
- ページ遷移
- データ並び替え

10 各コンポーネント定義

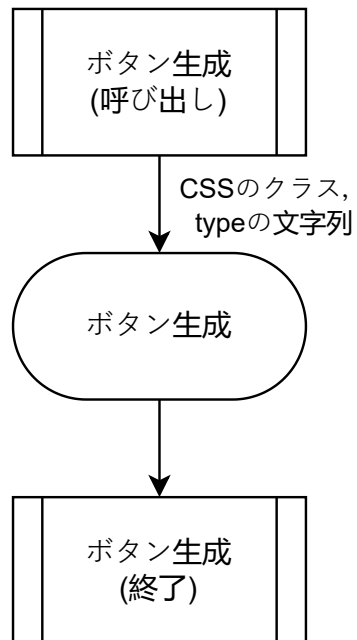


図 19 ボタン生成 (フロント)

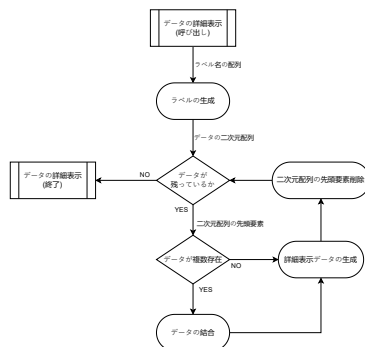


図 20 データの詳細表示 (フロント)

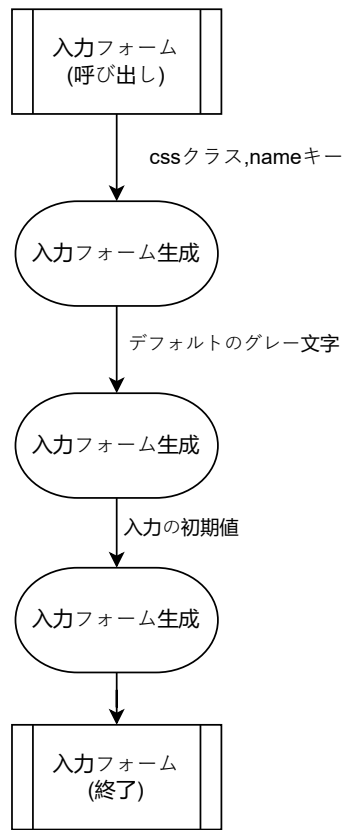


図 21 データの入力フォーム (フロント)

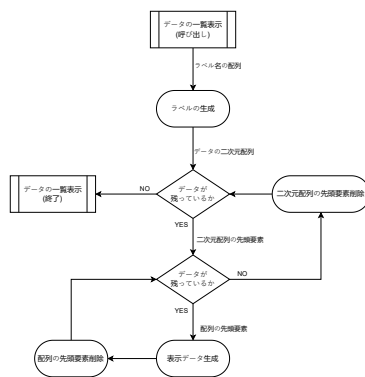


図 22 データの一覧表示 (フロント)

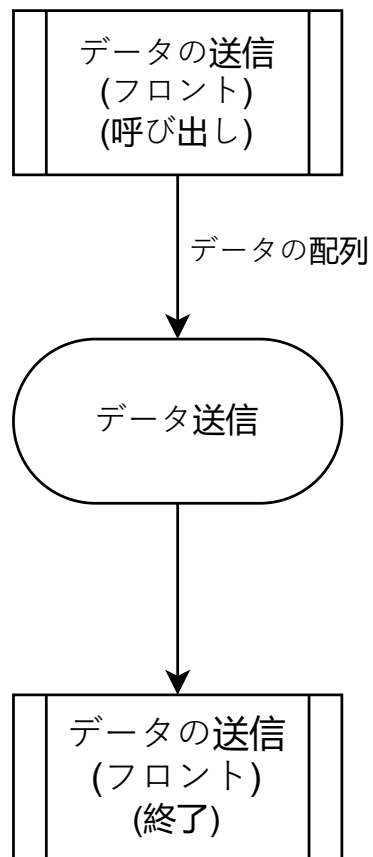


図 23 データ送信 (フロント)

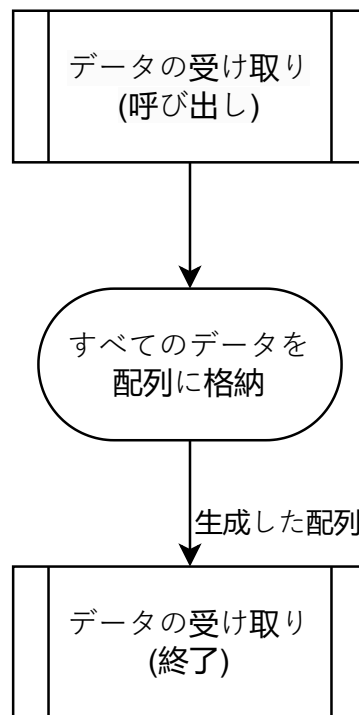


図 24 データの受け取り (バック)

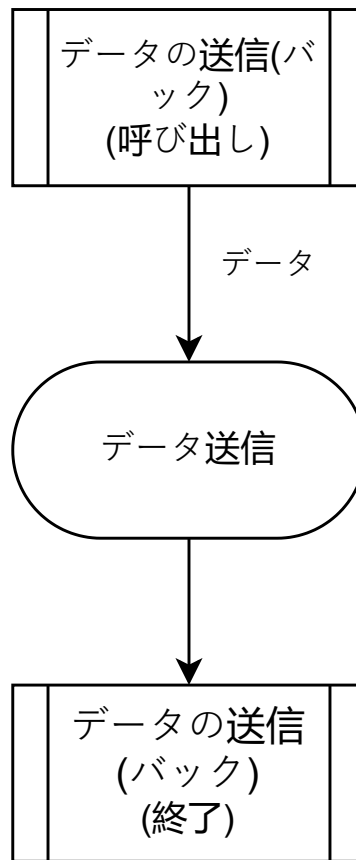


図 25 データ送信 (バック)

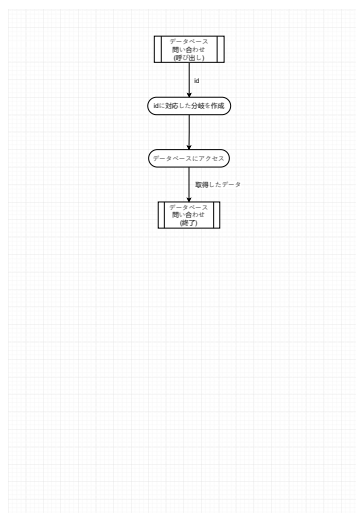


図 26 データベース問い合わせ (バック)

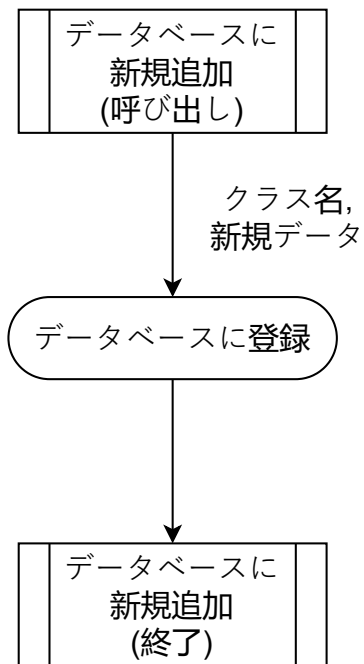


図 27 データベース新規登録 (バック)

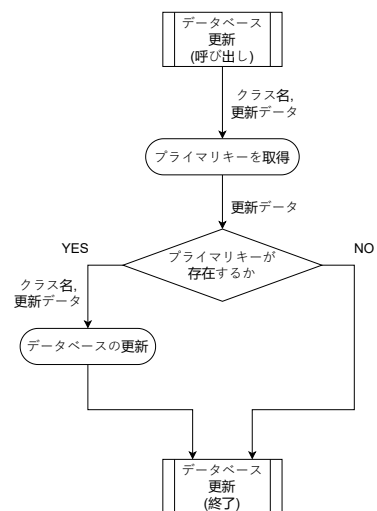


図 28 データベース更新 (バック)

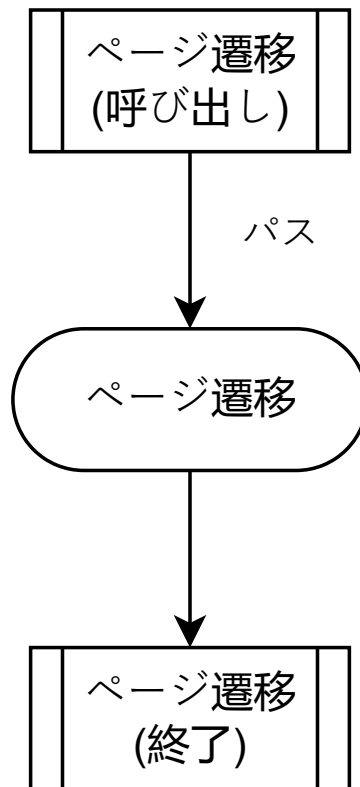


図 29 ページ遷移 (バック)

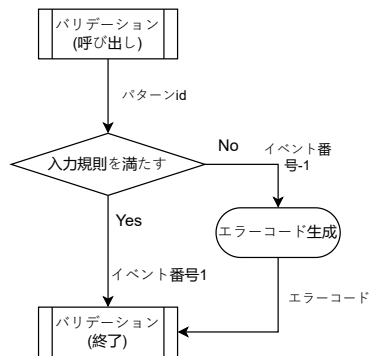


図 30 バリデーション (バック)

コンポーネント定義				
共通情報	システム名	Aero Net	バージョン	1.3
	工程名	内部設計	作成日付	2023/12/11
	作成者	林晃太郎	更新日付	2023/12/15
書誌情報	コンポーネントID	3		
	名称	ボタン生成		
	概要	入力された形式に応じてボタンを生成する		
入力				
第1引数：typeの文字列				
第2引数：cssのクラス				
出力				
処理終了イベント				
正常終了 -> イベント番号1				
エラー -> イベント番号-1				
処理手順				
1. typeの文字列とcssのクラスに従ってボタンを生成する。				
2. 正常に動作すればイベント番号1を返す。				
3. エラーが発生した場合、イベント番号-1を返す。				

図 31 ボタン生成 (フロント)

コンポーネント定義				
共通情報	システム名	Aero Net	バージョン	1.3
	工程名	内部設計	作成日付	2023/12/11
	作成者	松本吏司	更新日付	2023/12/15
書誌情報	コンポーネントID	5		
	名称	データの詳細表示コンポーネント		
	概要	受け取ったデータを詳細表示する		
入力				
第1引数：データのラベル(カラム名)の入った配列				
第2引数：データの2次元配列				
出力				
処理終了イベント				
正常終了 -> イベント番号 1 及び 生成された画面データ				
異常終了 -> イベント番号 -1				
処理手順				
1. 表示領域の一列目にラベルが来るように画面データの生成を行う。				
2. 二列目に各ラベルに対応するデータを表示するように、画面データの生成を行う。 二列目に対応するデータは一つにまとめて生成を行う。(姓・名などを空白で結合)				
3. 生成が終了した場合、イベント番号 1 と 生成した画面データを返す。				
4. 何らかのエラーの際は、イベント番号 -1 を返す。				

図 32 データの詳細表示 (フロント)

コンポーネント定義				
共通情報	システム名	Aero Net	バージョン	1.4
	工程名	内部設計	作成日付	2023/12/11
	作成者	久保田天治	更新日付	2023/12/15
書誌情報	コンポーネントID	1		
	名称	入力フォーム		
	概要	入力フォームを作成する		
入力				
第1引数：cssクラス				
第2引数：name属性のキー				
第3引数：デフォルトのグレー文字(必要のない場合空にする)				
第4引数：入力の初期値(必要のない場合空にする)				
出力				
処理終了イベント				
正常終了 -> イベント番号1				
エラー -> イベント番号-1				
処理手順				
1. cssのクラスを埋め込んだtext形式の入力フォームを生成する				
2. placeholder属性の値が第3引数にする				
3. value属性の初期値が第4引数にする				

図 33 データの入力フォーム (フロント)

コンポーネント定義				
共通情報	システム名	Aero Net	バージョン	1.3
	工程名	内部設計	作成日付	2023/12/11
	作成者	松本吏司	更新日付	2023/12/15
書誌情報	コンポーネントID	4		
	名称	データの一覧表示コンポーネント		
	概要	受け取ったデータを一覧表示する		
入力				
第1引数: データのラベル(カラム名)の入った配列				
第2引数: データの2次元配列				
出力				
処理終了イベント				
正常終了 -> イベント番号 1 及び 生成された画面データ				
異常終了 -> イベント番号 -1				
処理手順				
1. 表示領域の一行目にラベルが来るように画面データの生成を行う。				
2. 二行目以降にデータを表示するように画面データの生成を行う。				
3. 生成が終了した場合、イベント番号 1 と生成した画面データを返す。				
4. 何らかのエラーの際は、イベント番号 -1 を返す。				

図 34 データの一覧表示 (フロント)

コンポーネント定義				
共通情報	システム名	Aero Net	バージョン	1.3
	工程名	内部設計	作成日付	2023/12/11
	作成者	林晃太郎	更新日付	2023/12/15
書誌情報	コンポーネントID	2		
	名称	データの送信(フロント)		
	概要	データをバックエンド側に送信する		
入力				
第1引数：データの配列				
出力				
処理終了イベント 正常終了 -> イベント番号1 エラー -> イベント番号-1				
処理手順				
1. データをバックエンド側に送信する。 2. 送信が正常に完了すればイベント番号1を返す。 3. エラーが発生した場合、イベント番号2を返す。				

図 35 データ送信 (フロント)

コンポーネント定義				
共通情報	システム名	Aero Net	バージョン	1.3
	工程名	内部設計	作成日付	2023/12/11
	作成者	松本吏司	更新日付	2023/12/15
書誌情報	コンポーネントID	7		
	名称	データの受け取りコンポーネント		
	概要	バックエンドのデータ受け取りを担う		
入力				
第1引数：データの配列				
出力				
処理終了イベント 正常終了 -> イベント番号1				
処理手順				
1. 引数として渡されたデータ配列を受け取り、それを返す				

図 36 データの受け取り (バック)

コンポーネント定義				
共通情報	システム名	Aero Net	バージョン	1.3
	工程名	内部設計	作成日付	2023/12/11
	作成者	林晃太郎	更新日付	2023/12/15
書誌情報	コンポーネントID	8		
	名称	データの送信(バック)		
	概要	データをフロントエンド側に送信する		
入力				
第1引数：データの配列				
出力				
処理終了イベント 正常終了 -> イベント番号1 エラー -> イベント番号-1				
処理手順				
1. データをフロントエンド側に送信する。 2. 送信が正常に完了すればイベント番号1を返す。 3. エラーが発生した場合、イベント番号2を返す。				

図 37 データ送信 (バック)

コンポーネント定義				
共通情報	システム名	Aero Net	バージョン	1.4
	工程名	内部設計	作成日付	2023/12/11
	作成者	塩澤康志	更新日付	2023/12/15
書誌情報	コンポーネントID	9		
	名称	データベース問い合わせ		
	概要	データベースに問い合わせる		
入力				
第1引数 : id				
出力				
処理終了イベント				
正常終了-> 取得したデータ及びイベント番号1				
DBアクセスエラー-> イベント番号-1				
処理手順				
1. idに対応した分岐を作成する.				
2. データベースにアクセスしてデータを取得する.				
3. 正常に終了すればイベント番号1と取得したデータを返し、エラーが起こればイベント番号-1を返す.				

図 38 データベース問い合わせ (バック)

コンポーネント定義				
共通情報	システム名	Aero Net	バージョン	1.4
	工程名	内部設計	作成日付	2023/12/11
	作成者	塩澤康志	更新日付	2023/12/15
書誌情報	コンポーネントID	11		
	名称	データベース新規追加		
	概要	データベースにデータを新規追加する		
入力				
第1引数：テーブル名				
第2引数：新規データの配列				
出力				
処理終了イベント				
正常終了 -> イベント番号1				
DBアクセスエラー -> イベント番号-1				
処理手順				
1. データを追加する。				
2. 正常に終了すればイベント番号1, エラーが起こればイベント番号-1を返す。				

図 39 データベース新規登録 (バック)

コンポーネント定義				
共通情報	システム名	Aero Net	バージョン	1.4
	工程名	内部設計	作成日付	2023/12/11
	作成者	塩澤康志	更新日付	2023/12/15
書誌情報	コンポーネントID	10		
	名称	データベース更新		
	概要	データベースのデータを更新する		
入力				
第1引数：テーブル名				
第2引数：更新データの配列				
出力				
処理終了イベント				
正常終了 -> イベント番号1				
DBアクセスエラー -> イベント番号-1				
処理手順				
1. プライマリキーを確認する。				
2. データを更新する。				
3. 正常に終了すればイベント番号1，エラーが起こればイベント番号-1を返す。				

図 40 データベース更新 (バック)

コンポーネント定義				
共通情報	システム名	Aero Net	バージョン	1.3
	工程名	内部設計	作成日付	2023/12/11
	作成者	蟬 祐介	更新日付	2023/12/15
書誌情報	コンポーネントID	13		
	名称	ページ遷移		
	概要	ページを遷移する		
入力				
第1引数: パス				
出力				
ファイル				
処理手順				
1. パスを指定する				
2. 指定されたパスからファイルを返す				

図 41 ページ遷移 (バック)

コンポーネント定義				
共通情報	システム名	Aero Net	バージョン	1.3
	工程名	内部設計	作成日付	2023/12/11
	作成者	蟬 祐介	更新日付	2023/12/15
書誌情報	コンポーネントID	12		
	名称	バリデーション		
	概要	入力規則を満たすか確認する		
入力				
第1引数: パターンid				
第2引数: 配列				
出力				
処理終了イベント				
正常終了 -> イベント番号1				
異常終了 -> イベント番号-1				
処理手順				
1. パターンid, 配列を受け取る.				
2. 配列がパターンidの入力規則を満たすかどうか判別する.				
3. 入力規則を満たした場合は1を返し, 満たしていなければ-1とエラーコードを返す.				

図 42 バリデーション (バック)

11 貢献度

- 1240312 久保田 天治
 - － モジュール定義書作成
- 1250329 塩澤 康志
 - － モジュール定義書作成
 - － コンポーネント遷移図作成
 - － システム概要の記述
- 1250333 蟬 祐介
 - － 会議のリーダー及び書記を担当
 - － 動作環境と開発環境の記述
 - － コンポーネントの遷移図とモジュールの定義を作成
- 1250348 寺内 俊輔
 - － モジュールとコンポーネントのリストリストアップ
 - － 各コンポーネントの使用モジュールの決定
 - － モジュール定義の作成
- 1250358 林 晃太郎
 - － 会議のリーダーと書記, タイムキーパー
 - － フロー図と定義書のテンプレートの作成
 - － KPT の改善案の提示
 - － コーディング規約の選定と内部設計書への記載 (未完成)
- 1250371 松本 史司
 - － 遷移図表記のルール決めの記述
 - － 遷移図の作成
 - － 定義書の作成