

高知県観光振興部様

旅行プランサポートシステム

**Sutto**

内部設計書

株式会社 Flat

2019 年 12 月 22 日

# 目 次

<b>1</b>	<b>内部設計書作成方針</b>	<b>1</b>
1.1	開発環境	1
1.2	サーバ環境	1
1.3	動作環境	1
1.4	コーディング規約	2
1.4.1	命名規約	2
1.4.2	コーディングスタイル	2
1.5	設計書管理方針	3
<b>2</b>	<b>モジュール設計書</b>	<b>4</b>
2.1	ログイン機能	4
2.1.1	ログインモジュール	4
2.1.2	新規アカウント作成モジュール	5
2.2	ホーム機能	6
2.2.1	ホームモジュール	6
2.3	プラン作成機能	7
2.3.1	プラン作成モジュール	7
2.3.2	スポット検索モジュール	7
2.3.3	経路検索モジュール	8
2.3.4	メモ追加モジュール	8
2.4	MY プラン機能	10
2.4.1	MY プラン一覧モジュール	10
2.4.2	プラン確認モジュール	10
2.4.3	投稿モジュール	11
2.4.4	経路検索モジュール	12
2.5	みんなのプラン機能	13
2.5.1	みんなのプランモジュール	13
2.5.2	プラン表画面モジュール	13
2.5.3	検索モジュール	13
2.5.4	評価モジュール	14
2.5.5	経路検索モジュール	14
2.6	アカウント機能	16
2.6.1	アカウント情報変更モジュール	16
2.6.2	ユーザ情報変更モジュール	16
2.6.3	パスワード変更モジュール	17
2.7	広告機能	18
2.7.1	広告管理モジュール	18
<b>3</b>	<b>モジュール仕様</b>	<b>19</b>
3.1	Login.kt	19
3.2	NewAccountCreate.kt	19
3.3	PlanCreate.kt	20

3.4	SpotSearch.kt . . . . .	20
3.5	PlanTable.kt . . . . .	21
3.6	SpotInfo.kt . . . . .	21
3.7	Route.kt . . . . .	21
3.8	MyPlan.kt . . . . .	22
3.9	PlanPost.kt . . . . .	22
3.10	EveryonePlan.kt . . . . .	23
3.11	EveryoneSearch.kt . . . . .	23
3.12	PlanEvaluation.kt . . . . .	24
3.13	UserInfoChange.kt . . . . .	24
3.14	Advertisement.kt . . . . .	25
<b>4</b>	<b>データテーブル</b>	<b>26</b>
4.1	ユーザ管理システム . . . . .	26
4.1.1	ユーザ情報内部データー一覧 . . . . .	26
4.1.2	ユーザ情報データテーブル . . . . .	26
4.2	プラン作成システム . . . . .	27
4.2.1	プラン作成内部データー一覧 . . . . .	27
4.2.2	スポットテーブル . . . . .	27
4.2.3	プラン表示テーブル . . . . .	28
4.2.4	プラン作成テーブル . . . . .	29
4.2.5	スポットカテゴリテーブル . . . . .	30
4.2.6	交通カテゴリテーブル . . . . .	30
4.2.7	タイムテーブル . . . . .	31
4.2.8	料金テーブル . . . . .	32
4.2.9	都道府県テーブル . . . . .	32
4.2.10	スポット写真テーブル . . . . .	33
4.3	MY プランシステム . . . . .	34
4.3.1	MY プランシステム . . . . .	34
4.3.2	写真テーブル . . . . .	34
4.4	みんなのプランシステム . . . . .	35
4.4.1	みんなのプラン内部データー一覧 . . . . .	35
4.4.2	評価テーブル . . . . .	35
4.5	広告管理システム . . . . .	36
4.5.1	広告管理内部データー一覧 . . . . .	36
4.5.2	広告テーブル . . . . .	36
4.5.3	広告写真テーブル . . . . .	36

# 1 内部設計書作成方針

「Sutto」を開発するにあたり、使用する環境や作成方針を以下に示します。

## 1.1 開発環境

本システムの開発環境を表 1 に示します。

表 1: 開発環境

IDE	Android Studio 3.5.2
開発言語	Kotlin 1.3.5
画面設計	XML
バージョン管理	GitHub

## 1.2 サーバ環境

本システムのサーバ環境を表 2 に示します。

表 2: サーバ環境

OS	Ubuntu Server
サーバソフトウェア	Apache 2.4.41
データベース	mysql 8.0.18
vCPU	1
RAM	1.0

## 1.3 動作環境

本システムの動作環境を表 3 に示します。

表 3: 動作環境

OS	Android 8.1 以上
通信	Wi-Fi
CPU	MSM8940 1.4GHz + 1.1GHz
RAM	32GB
ROM	3GB
API	Google Maps API, 乗換案内オープン API
推奨端末	Android One X1

## 1.4 コーディング規約

本システムを開発する際のコーディング規約を示します。

### 1.4.1 命名規約

- ファイル・クラス名
  - － 英語のみとし、意味のある英単語
  - － 単語の頭文字は大文字を使用
  - － 後続する英単語の頭文字は大文字（アンダースコアは無し）  
例）UserDelete
- 変数・メソッド名
  - － 英字のみとし、意味のある英単語で構成
  - － 頭文字は小文字を使用
  - － 後続する英単語の頭文字は大文字（アンダースコアは無し）  
例）userLogin
- 定数
  - － 全て大文字の英単語を使用
  - － 複数単語の場合、単語間はアンダースコアで区切る  
例）HUNDRED
- テーブル・カラム名
  - － 全て小文字の英単語を使用
  - － 複数単語の場合、単語間はアンダースコアで区切る  
例）user\_id, user\_login\_flag

### 1.4.2 コーディングスタイル

- インデント
  - － 半角スペース 4 つ
- 括弧
  - － 中括弧 { は打った後に改行  
例)  
for(...){  
...  
}
  - － 小括弧の前後にスペースは無し
- 演算子
  - － 演算子の前後には半角スペース 1 文字を挿入

## 1.5 設計書管理方針

内部設計書の管理には Google ドライブを使用します。

## 2 モジュール設計書

モジュール構成において各図形・色が何を意味するか図 1 に示します。

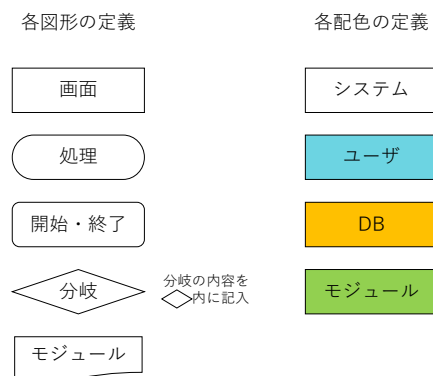


図 1: モジュール内の図形・色の定義

図 1 において、四角形は「表示される画面の名前」、楕円形は「処理内容」、角の丸い四角形は「モジュールの開始と終了」、ひし形は「分岐」、下が波線の四角形は「モジュール」を表します。また、図形の配色は白が「システムの動作」、水色が「ユーザの操作」、オレンジ色が「データベースの処理」、黄緑色が「モジュール」を表します。

各モジュールのすべての画面からホームボタンを押下することによりホーム画面に遷移します。また、広告についても同様にすべての画面にバナー広告が表示されており、バナーを押下することにより設定されている URL の画面に遷移します。

### 2.1 ログイン機能

#### 2.1.1 ログインモジュール

図 2 は、トップ画面からログインボタンを選択し、ユーザがログインをする際のシステムの流れを示しています。

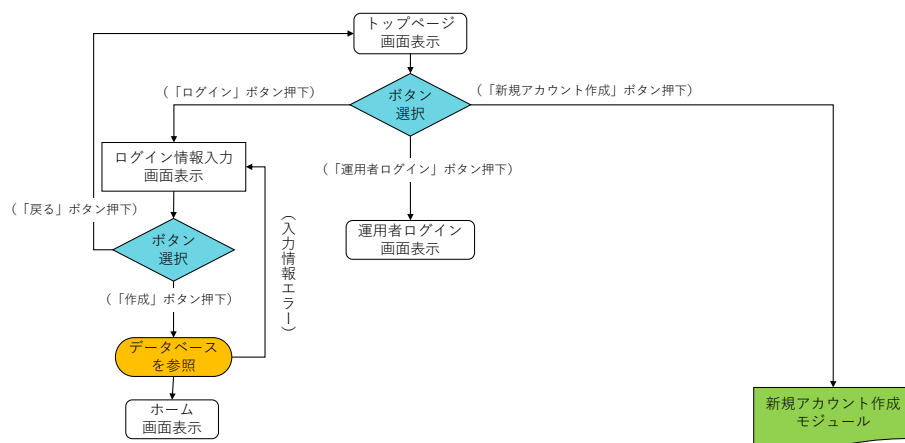


図 2: ログインモジュール

## 2.1.2 新規アカウント作成モジュール

図 3 は、新規アカウントを作成する際のシステムの流れを示しています。

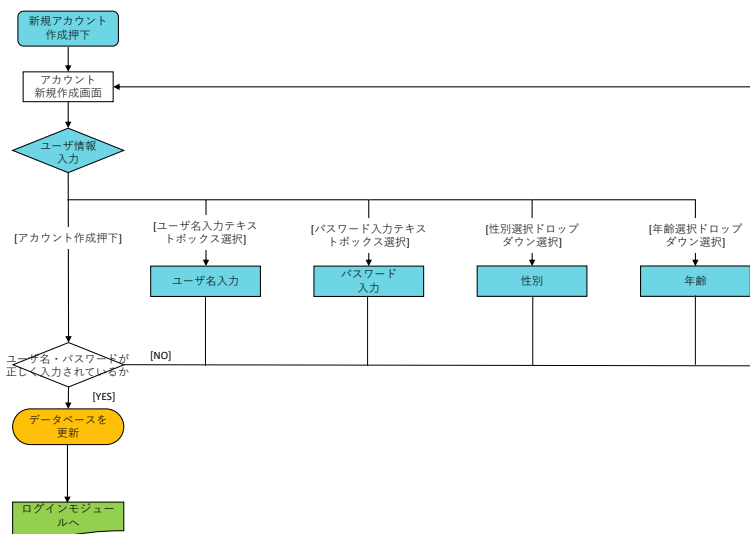


図 3: 新規アカウント作成モジュール



## 2.2 ホーム機能

### 2.2.1 ホームモジュール

図4は、ホーム画面を表示し、各サブシステムを利用するまでのシステムの流れを示しています。ピンクの破線で囲まれた部分が一般ユーザが扱うことができるモジュールです。

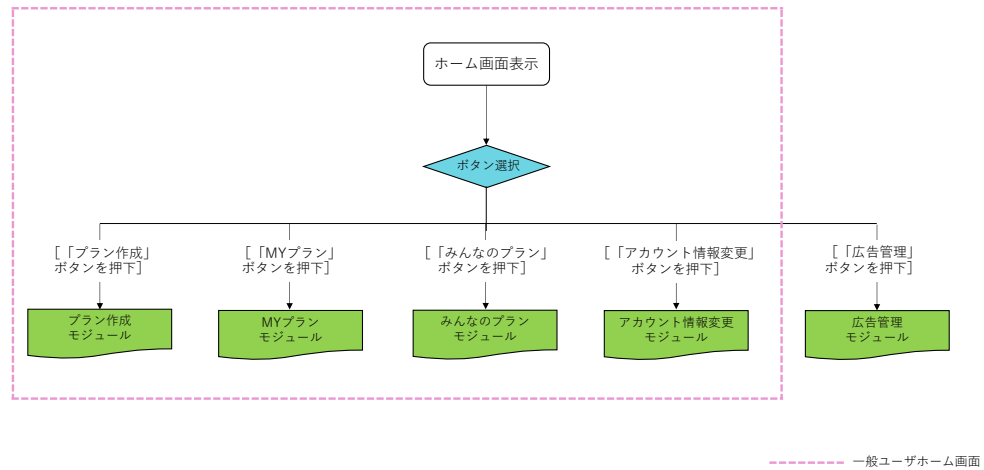


図 4: ホームモジュール

## 2.3 プラン作成機能

### 2.3.1 プラン作成モジュール

図 5 は、プランを作成する際のシステムの流れを示しています。

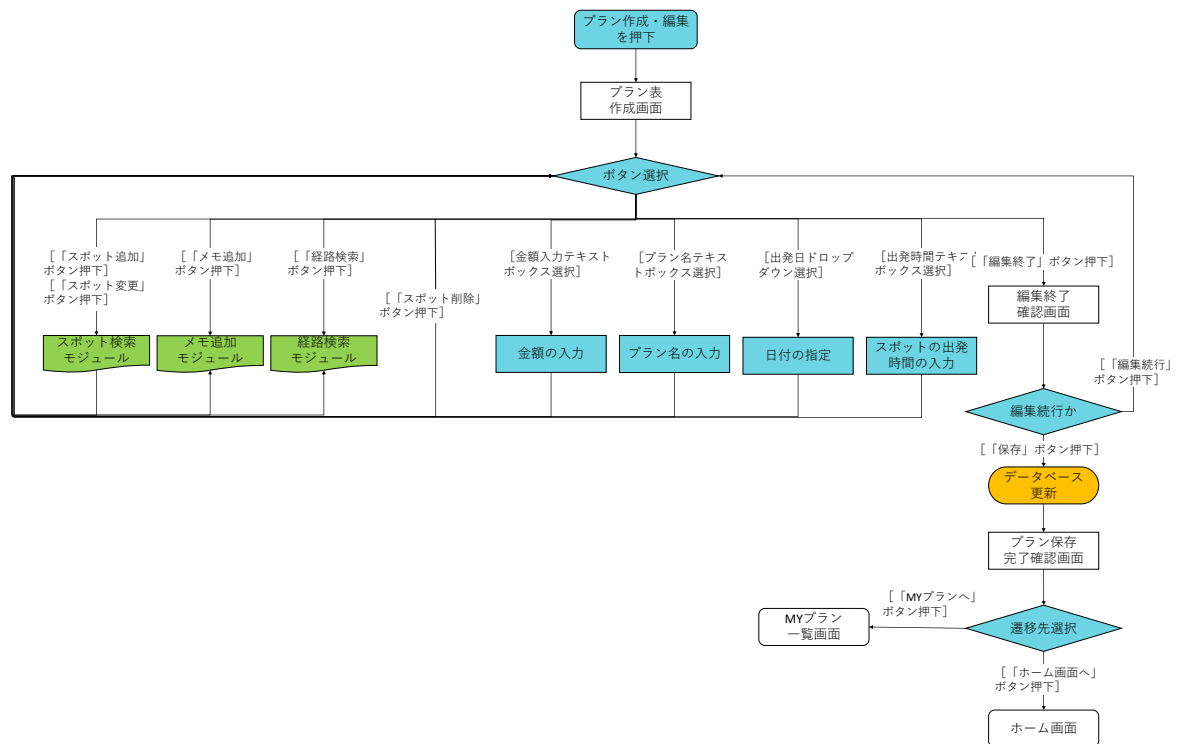


図 5: プラン作成モジュール

### 2.3.2 スポット検索モジュール

図 6 は、スポットを検索し、プラン表に追加する際のシステムの流れを示しています。

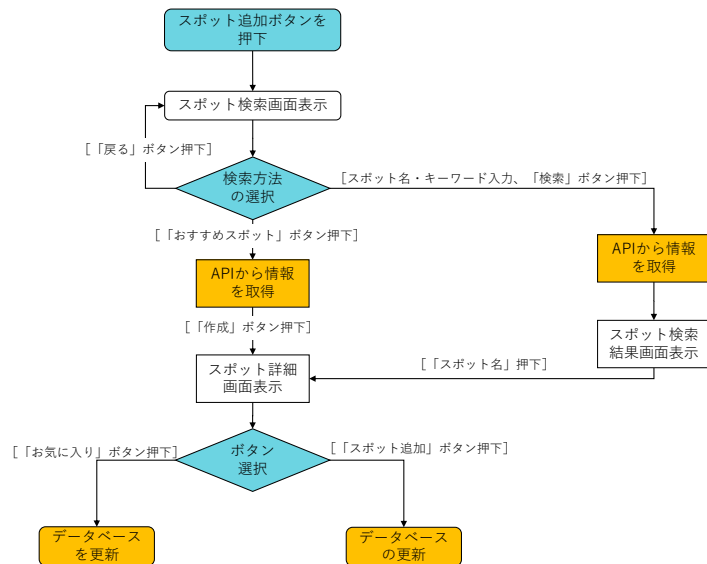


図 6: スポット検索モジュール

### 2.3.3 経路検索モジュール

図 7 は、スポット間の経路を検索し表示するシステムの流れを示しています。

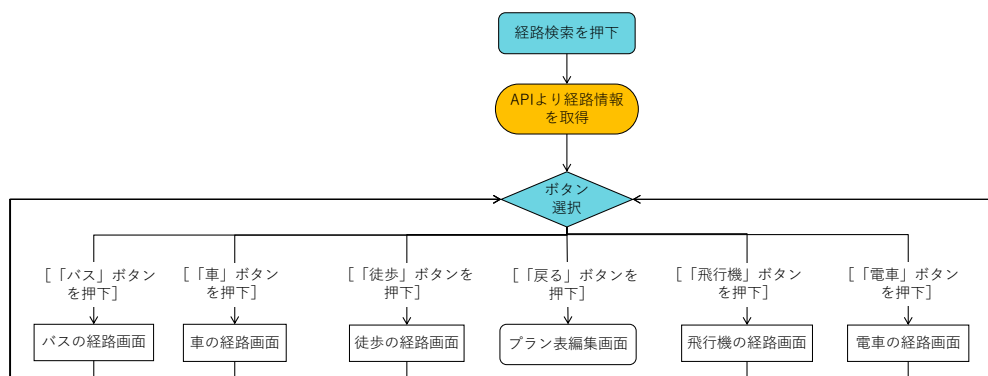


図 7: 経路検索モジュール

### 2.3.4 メモ追加モジュール

図 8 は、作成したプランにメモや写真を追加する際のシステムの流れを示しています。

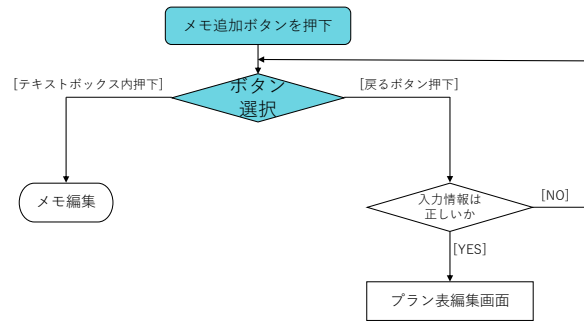


図 8: メモ追加モジュール

## 2.4 MY プラン機能

### 2.4.1 MY プラン一覧モジュール

図 9 は、保存したプラン表の一覧を表示するシステムの流れを示しています。

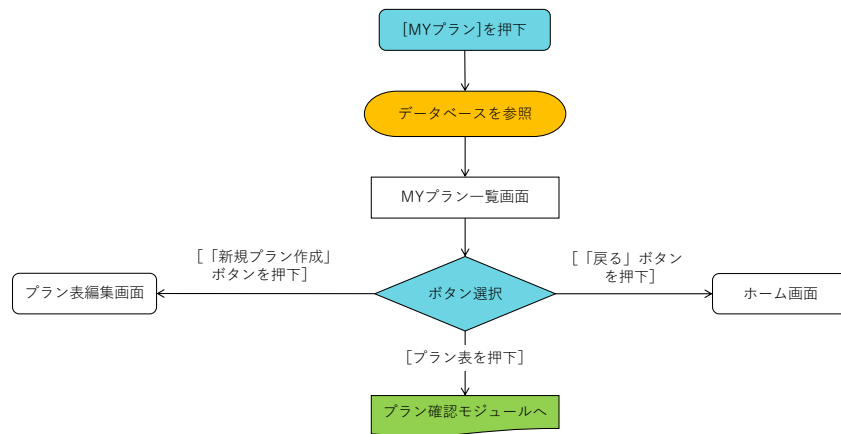
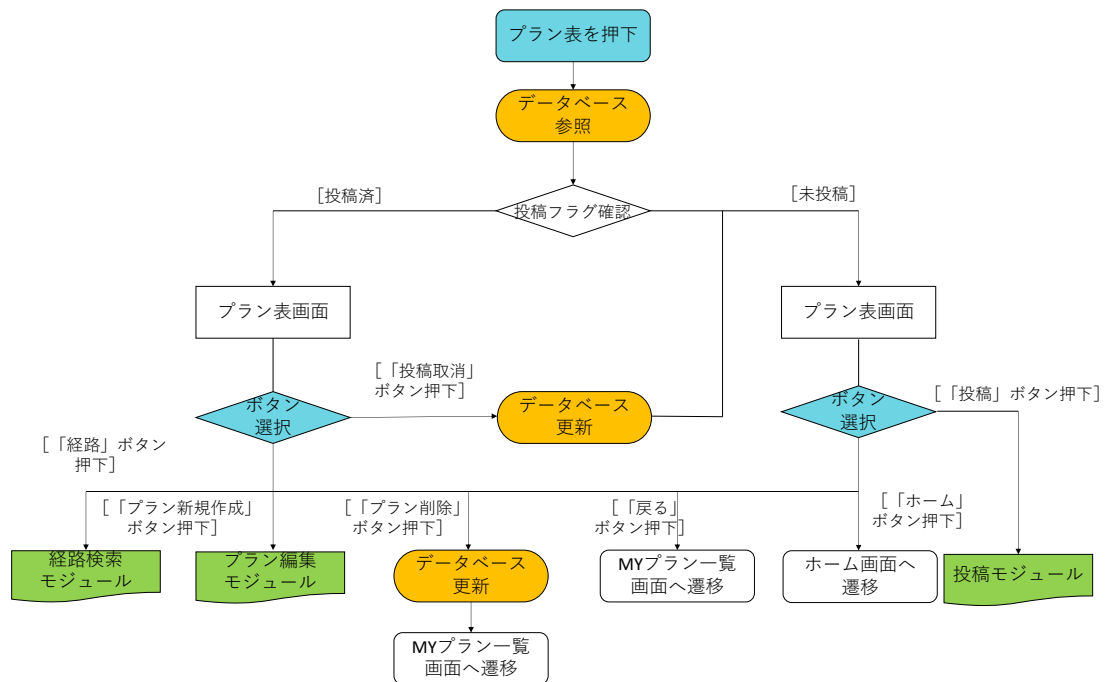


図 9: MY プラン一覧モジュール

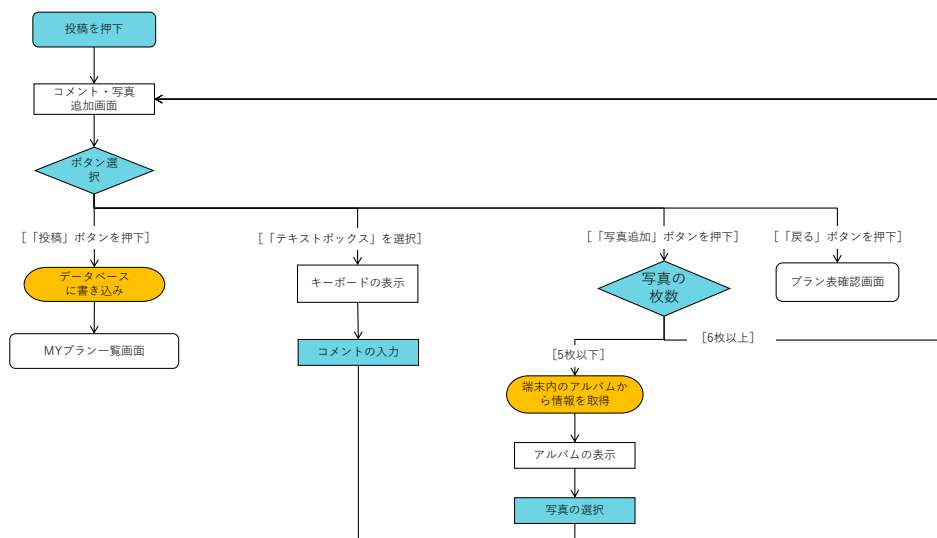
### 2.4.2 プラン確認モジュール

図 10 は、MY プラン一覧に保存されているプラン表の詳細を表示し、各サブシステムを利用するまでのシステムの流れを示しています。



### 2.4.3 投稿モジュール

図 11 は、選択されているプラン表をみんなのプランに投稿する際のシステムの流れを示しています。



#### 2.4.4 経路検索モジュール

図 12 は、スポット間の経路を検索し表示するシステムの流れを示しています。

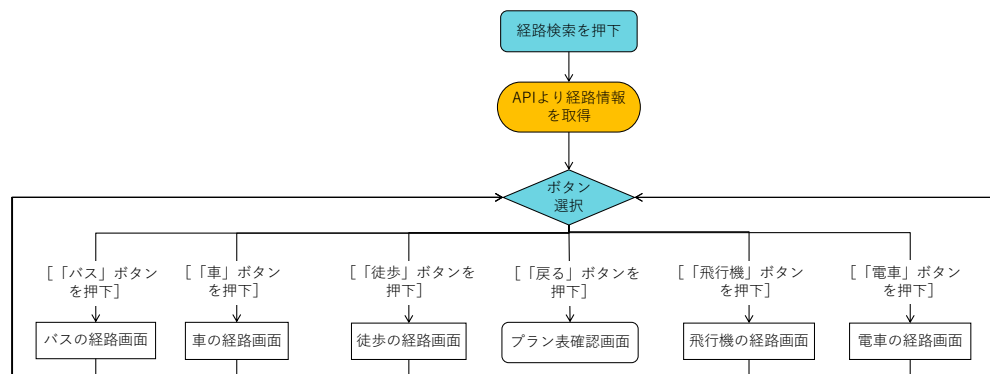


図 12: 経路検索モジュール

## 2.5 みんなのプラン機能

### 2.5.1 みんなのプランモジュール

図 13 は、みんなのプラン一覧を表示するシステムの流れを示しています。

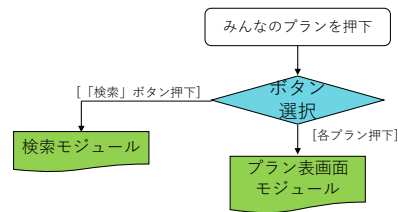


図 13: みんなのプランモジュール

### 2.5.2 プラン表画面モジュール

図 14 は、みんなのプランから選択して表示したプラン表に対する操作の流れを示しています。

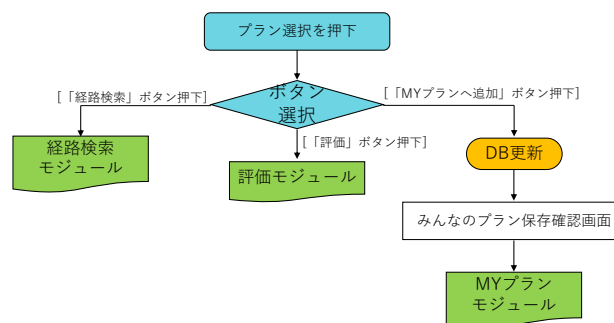


図 14: プラン表画面モジュール

### 2.5.3 検索モジュール

図 15 は、検索条件を指定するシステムの流れを示しています。



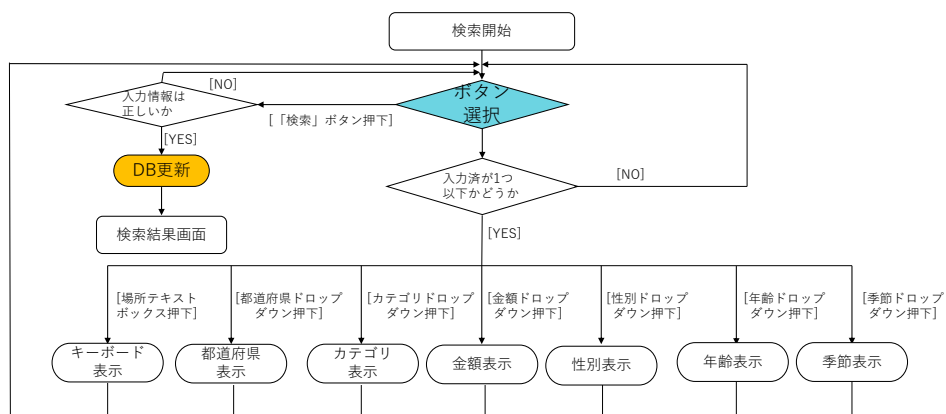


図 15: 検索モジュール

#### 2.5.4 評価モジュール

図 16 は、みんなのプランを評価するシステムの流れを示しています。

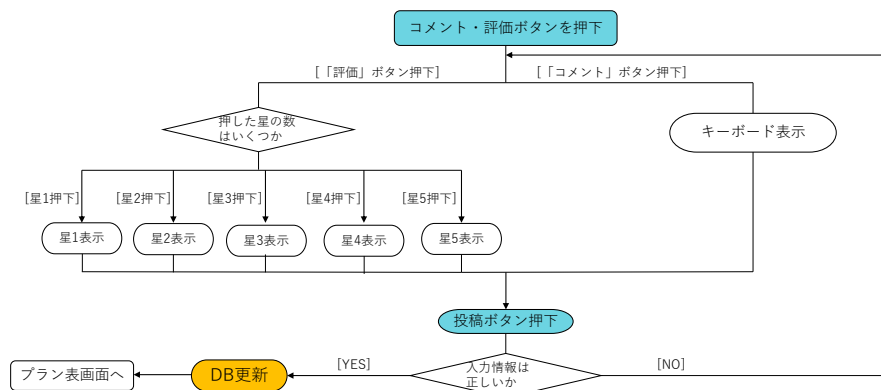


図 16: 評価モジュール

#### 2.5.5 経路検索モジュール

図 17 は、スポット間の経路を検索し表示するシステムの流れを示しています。

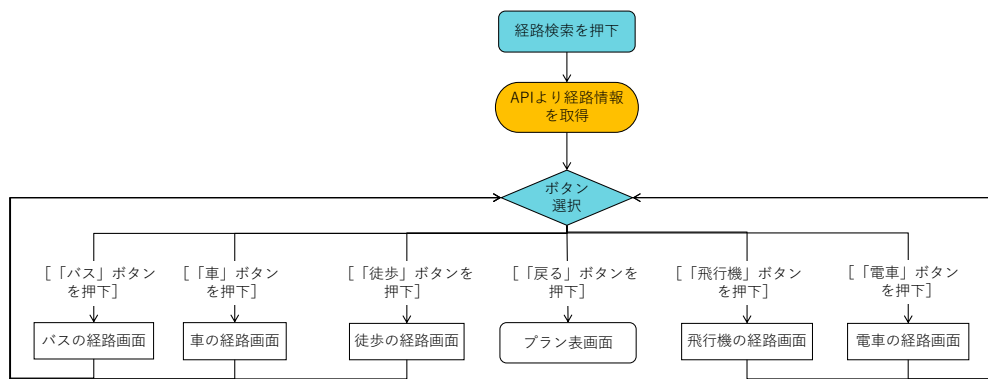


図 17: 経路検索モジュール

## 2.6 アカウント機能

### 2.6.1 アカウント情報変更モジュール

図 18 は、アカウント情報を表示し、変更する情報を選択するシステムの流れを示しています。

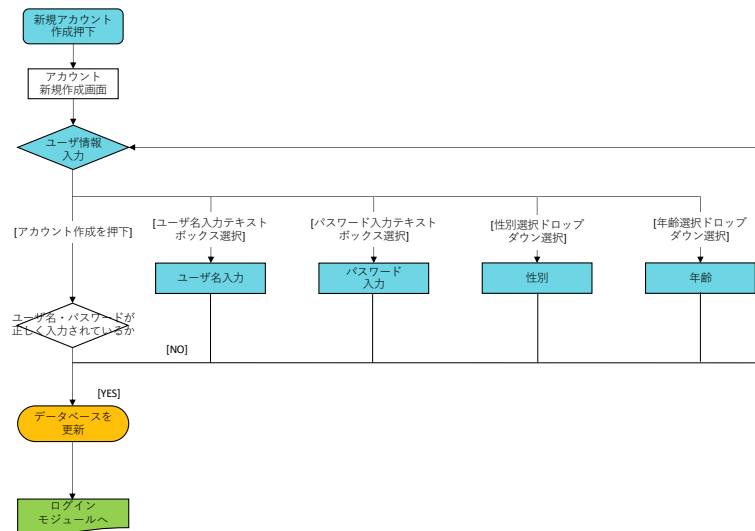


図 18: アカウント情報変更モジュール

### 2.6.2 ユーザ情報変更モジュール

図 19 は、登録しているユーザ情報を変更するシステムの流れを示しています。

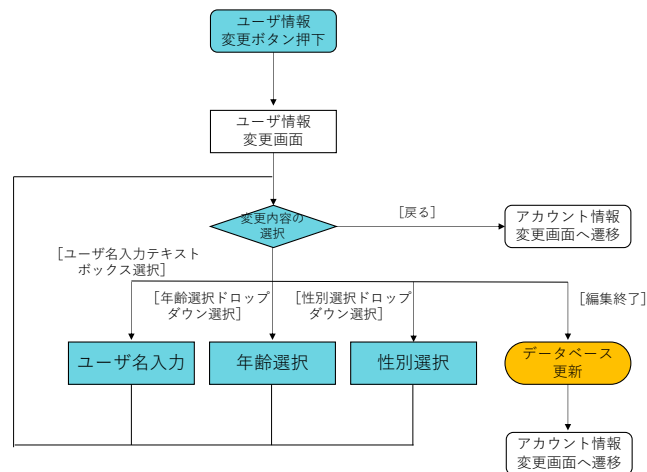


図 19: ユーザ情報変更モジュール

### 2.6.3 パスワード変更モジュール

図 20 は、パスワードを変更するシステムの流れを示しています。

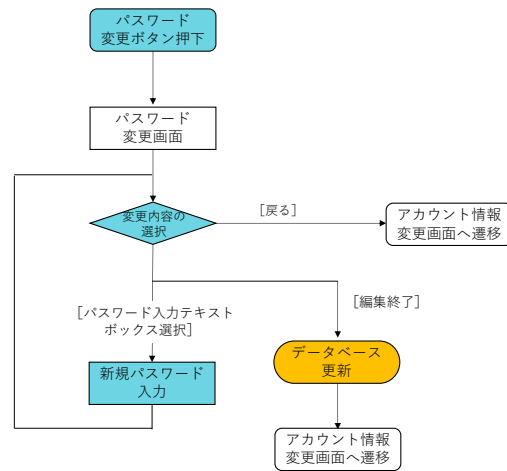


図 20: パスワード変更モジュール

## 2.7 広告機能

### 2.7.1 広告管理モジュール

図 21 は、運用ユーザがアプリ内に表示する広告を管理するシステムの流れを示しています。

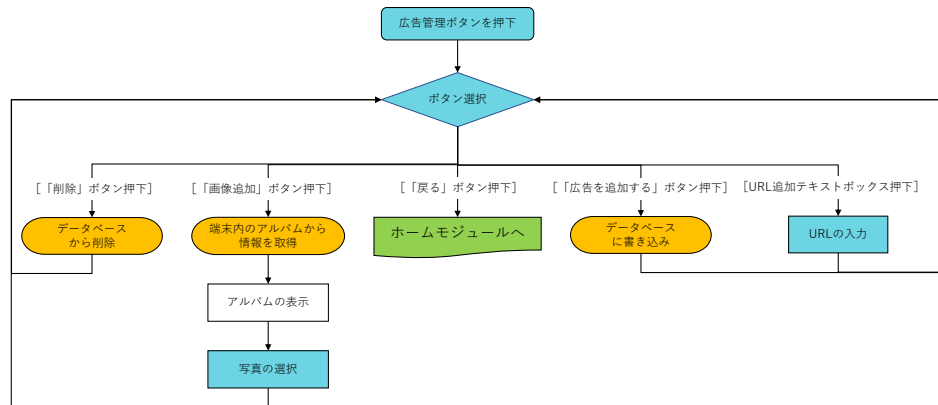


図 21: 広告管理モジュール

### 3 モジュール仕様

図 22 は、本システムのクラス図を示しています。

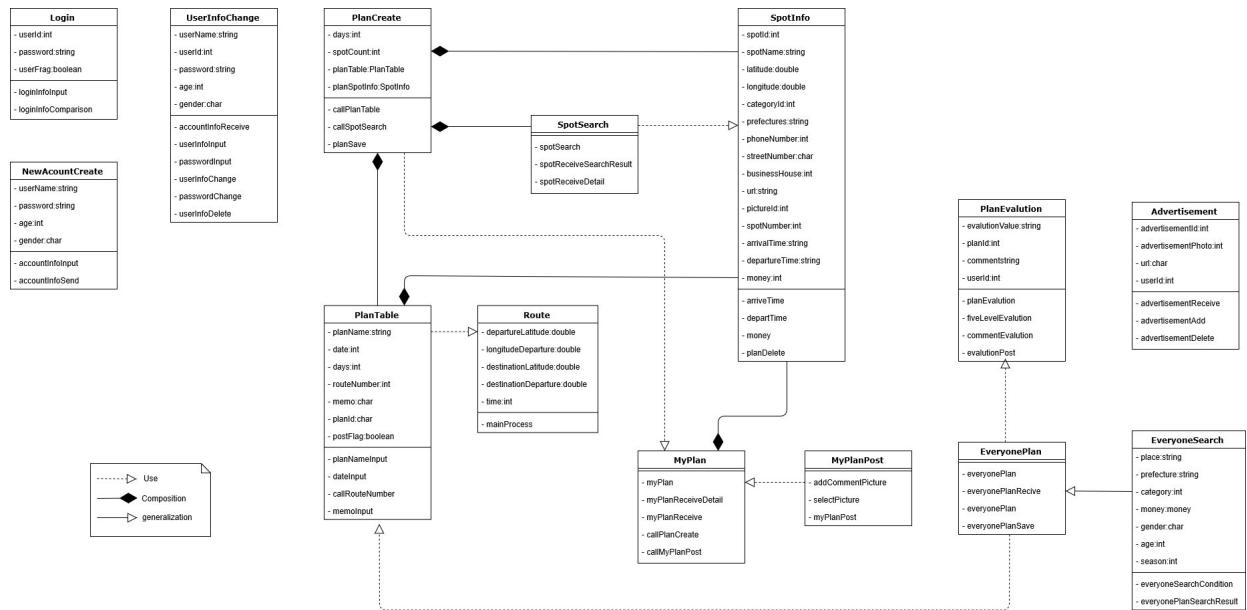


図 22: クラス図

#### 3.1 Login.kt

- クラス名 : Login

- － メソッド名 : loginInfoInput

概要 : ユーザにログイン情報を入力させる。

引数 : -

戻り値 : -

- － メソッド名 : loginInfoComparison

概要 : ログイン情報入力メソッドの情報を DB に送信し、入力した情報がテーブル上にあるか比較する。

引数 : userId(int 型)、password(string 型)

戻り値 : userFlag(boolean 型)

#### 3.2 NewAccountCreate.kt

- クラス名 : NewAccountCreate

- － メソッド名 : accountInfoInput

概要 : 新規で作成するアカウントの情報をユーザに入力させる。

引数 : -

戻り値 : -

- － メソッド名：accountInfoSend  
概要：アカウント情報入力メソッドで入力した情報を DB に送信し、ユーザ ID を受信しユーザに渡す。  
引数：userName(string 型)、password(string 型)、age(int 型)、gender(char 型)  
戻り値：userId(int 型)

### 3.3 PlanCreate.kt

- クラス名：PlanCreate
  - － メソッド名：callSpotSearch  
概要：SpotSearch を呼び出し、プラン表に追加するスポットの情報を取得する。  
引数：-  
戻り値：-
  - － メソッド名：callRoute  
概要：経路検索クラスを呼び出し、ルート検索を行う。  
引数：routeNumber(int 型)  
戻り値：-
  - － メソッド名：spotDelete  
概要：スポット削除の処理を受け付ける。  
引数：spotNumber(int 型)、dateNumber(int 型)  
戻り値：spotNumber(int 型)
  - － メソッド名：planSave  
概要：作成されたプラン表をデータベースに保存する。  
引数：-  
戻り値：-

### 3.4 SpotSearch.kt

- クラス名：SpotSearch
  - － メソッド名：spotSearchMain  
概要：スポットを検索し、プラン表にスポットを追加する。  
引数：-  
戻り値：-
  - － メソッド名：receiveSearchResult  
概要：引数の言葉を対にして一致する条件のスポットを DB から取得する。  
引数：searchWord(string 型)  
戻り値：planId(int[] 型), planName(string[] 型), pictureId(int[] 型)
  - － メソッド名：receiveDetail  
概要：スポットの詳細を DB から取得する。  
引数：planId(int 型)  
戻り値：searchSpotInfo(SpotInfo 型)

### 3.5 PlanTable.kt

- クラス名 : PlanTable
  - － メソッド名 : planNameInput  
概要 : ユーザのプラン名の入力を受け付ける。  
引数 : -  
戻り値 : -
  - － メソッド名 : dateInput  
概要 : ユーザの旅行の日付入力を受け付ける。  
引数 : -  
戻り値 : -
  - － メソッド名 : RouteSearch  
概要 : 経路検索クラスを呼び出し、ルート検索結果を表示する。  
引数 : departureLatitude(double 型)、departureLongitude(double 型)、destinationLatitude(double 型)、destinationLongitude(double 型)、time(int 型)  
戻り値 : -
  - － メソッド名 : memoInput  
概要 : ユーザのメモ入力を受け付ける。  
引数 : -  
戻り値 : -

### 3.6 SpotInfo.kt

- クラス名 : SpotInfo
  - － メソッド名 : arriveTime  
概要 : 到着時間の入力を受け付ける。  
引数 : -  
戻り値 : -
  - － メソッド名 : departTime  
概要 : 出発時間の入力を受け付ける。  
引数 : -  
戻り値 : -
  - － メソッド名 : fare  
概要 : 合計金額の入力を受け付ける。  
引数 : -  
戻り値 : -

### 3.7 Route.kt

- クラス名 : Route



- － メソッド名：routeMain  
概要：二地点の緯度と経度の情報を API に送り、経路の情報を得る。画面へ経路の表示を行う。  
引数：-  
戻り値：-

### 3.8 MyPlan.kt

- クラス名：MyPlan
  - － メソッド名：myPlanMain  
概要：MyPlan を動作させるメソッド。  
引数：-  
戻り値：-
  - － メソッド名：myPlanReceive  
概要：ログイン中のユーザのユーザ ID をもとに、そのユーザ ID のプランを DB から受け取る。  
引数：userId(int 型)  
戻り値：planNameArray(string[] 型)、planIdArray(int[] 型)
  - － メソッド名：myPlanReceiveDetail  
概要：選択したプランのプラン ID をもとにそのプランの情報を受信する。  
引数：planId(int 型)  
戻り値：planTableInstanceArray(PlanTable[] 型)、spotInfoInstanceArray(SpotInfo[] 型)
  - － メソッド名：callPlanCreate  
概要：プランを編集するを選択する場合に、プラン作成クラスを呼び出す。  
引数：planTableInstanceArray(PlanTable[] 型)、spotInfoInstanceArray(SpotInfo[] 型)  
戻り値：-
  - － メソッド名：callMyPlanPost  
概要：投稿ボタンを選択した際に、投稿を行うクラスを呼び出す。  
引数：-  
戻り値：-

### 3.9 PlanPost.kt

- クラス名：PlanPost
  - － メソッド名：planPostMain  
概要：PlanPost を動作させるメソッド。  
引数：-  
戻り値：-
  - － メソッド名：addCommentPicture  
概要：投稿した際に表示される写真とコメントをプランに追加する。  
引数：-  
戻り値：-

- － メソッド名：selectPicture
    - 概要：追加する写真を選択する。
    - 引数：-
    - 戻り値：-
  - － メソッド名：post
    - 概要：入力した内容を確定させ、評価・コメントを DB に保存する。
    - 引数：planId(int 型)、comment(string 型)、picture(int 型)
    - 戻り値：-

### 3.10 EveryonePlan.kt

- クラス名：EveryonePlan
  - － メソッド名：everyonePlanMain
      - 概要：EveryonePlan を動作させるメソッド。
      - 引数：-
      - 戻り値：-
    - － メソッド名：everyonePlanReceive
      - 概要：DB 内の投稿フラグのついたプランの情報を受信する。
      - 引数：-
      - 戻り値：planName(string[] 型)、planId(int[] 型)、averageEvaluationValue(double[] 型)
    - － メソッド名：everyonePlanReceiveDetail
      - 概要：選択したプランのプラン ID をもとにそのプランの情報を受信する。
      - 引数：planId(int 型)
      - 戻り値：planTableInstanceArray(PlanTable[] 型)、spotInfoInstanceArray(spotInfo[] 型)
    - － メソッド名：everyonePlanSave
      - 概要：プラン保存ボタンを選択した場合に、そのプラン情報を保存するユーザのプランとして保存する。
      - 引数：planId(int 型)、userId(int 型)
      - 戻り値：-

### 3.11 EveryoneSearch.kt

- クラス名：EveryoneSearch
  - － メソッド名：everyoneSearchMain
      - 概要：EveryoneSearch を動作させるメソッド。
      - 引数：-
      - 戻り値：-
    - － メソッド名：everyoneSearchCondition
      - 概要：ユーザが検索条件を入力する。
      - 引数：-
      - 戻り値：-

- － メソッド名：everyonePlanSearchResult  
概要：オーバーライドし、検索する内容を検索条件入力メソッドの内容に変更して DB に送信、一致する条件のものを受信する。  
引数：place(string 型)、prefecture(int 型)、category(int 型)、money(int 型)、age(int 型)、gender(char 型)、season(string 型)  
戻り値：planName(string[] 型)、planId(int[] 型)、evaluationAverage(double[] 型)

### 3.12 PlanEvaluation.kt

- クラス名：PlanEvaluation
  - － メソッド名：planEvaluationMain  
概要：PlanEvaluation を動作させるメソッド。  
引数：-  
戻り値：-
  - － メソッド名：fiveLevelEvaluation  
概要：投稿されているプランに 5 段階評価を入力する。  
引数：-  
戻り値：-
  - － メソッド名：commentEvaluation  
概要：投稿されているプランにコメントを入力する。  
引数：-  
戻り値：-
  - － メソッド名：evaluationPost  
概要：入力した 5 段階評価とコメント評価を DB に送信する。  
引数：planId(int 型)、userId(int 型)  
戻り値：-

### 3.13 UserInfoChange.kt

- クラス名：UserInfoChange
  - － メソッド名：accountInfoReceive  
概要：ユーザ ID をもとにユーザ情報を DB から受信する。  
引数：userId(int 型)  
戻り値：userName(string 型)、password(string 型)、age(int 型)、gender(char 型)
  - － メソッド名：userInfoInput  
概要：新しいユーザ情報を入力させる。  
引数：-  
戻り値：-
  - － メソッド名：passwordInput  
概要：新しいパスワードを入力させる。  
引数：-  
戻り値：-

- メソッド名：userInfoChange  
 概要：変更したユーザ情報を DB に送信する。  
 引数：userId(int 型)、userName(string 型)、age(int 型)、gender(char 型)  
 戻り値：-
- メソッド名：passwordChange  
 概要：変更したパスワードを DB に送信する。  
 引数：userId(int 型)、password(string 型)  
 戻り値：-
- メソッド名：userInfoDelete  
 概要：ユーザ情報を全て削除する。  
 引数：userId(int 型)  
 戻り値：-

### 3.14 Advertisement.kt

- クラス名：Advertisement
  - メソッド名：advertisementMain  
 概要：Advertisement を動作させるメソッド。  
 引数：-  
 戻り値：-
  - メソッド名：advertisementReceive  
 概要：運用ユーザが投稿した広告を受信する。  
 引数：userId(int 型)  
 戻り値：userId(int[] 型)、advertisementPhoto(int[] 型)、url(char[] 型)
  - メソッド名：advertisementAdd  
 概要：運用ユーザが広告を追加する。  
 引数：userId(int 型)  
 戻り値：-
  - メソッド名：advertisementDelete  
 概要：運用ユーザが追加した広告を削除する。  
 引数：advertisementId(int 型)  
 戻り値：-

## 4 データテーブル

データベースを構成するテーブルを示します。

### 4.1 ユーザ管理システム

#### 4.1.1 ユーザ情報内部データ一覧

表 4 には、ユーザ情報内部データの一覧を示します。

表 4: ユーザ情報内部データ一覧

システム名	プランサポートシステム「Sutto」	作成日	2019年11月30日
サブシステム	ユーザ管理システム	作成者	宮尾将史
論理名	物理名		
ユーザ情報テーブル	user		

#### 4.1.2 ユーザ情報データテーブル

表 5 には、ユーザ情報データテーブルの内容を示します。

表 5: ユーザ情報データテーブル

プランサポートシステム「Sutto」		作成日	2019年11月30日	
ユーザ管理システム		作成者	宮尾将史	
ユーザ情報テーブル		備考	一般・運用・管理フラグはいずれかを立てる必要がある	
論理名	物理名	データ柄	NOTNULL	PK/FK
ユーザID	user_id	int	○	PK
ユーザName	user_name	varchar	○	
パスワード	user_pass	varchar	○	
性別	user_gender	char		
年齢	user_age	int		
ログインフラグ	user_login_flag	Boolean		
一般ユーザフラグ	user_general_flag	Boolean	○	
運用ユーザフラグ	user_operation_flag	Boolean	○	
管理ユーザフラグ	user_management_flag	Boolean	○	

## 4.2 プラン作成システム

### 4.2.1 プラン作成内部データ一覧

表 6 には、プラン作成内部データの一覧を示します。

表 6: プラン作成内部データ一覧

システム名	プランサポートシステム「Sutto」	作成日	2019年12月16日
サブシステム	プラン作成システム	作成者	中村笙悟
論理名	物理名		
スポットテーブル	spot		
プラン表示テーブル	show_plan		
プラン作成テーブル	make_plan		
スポットカテゴリテーブル	spot_category		
交通カテゴリテーブル	traffic_category		
タイムテーブル	time		
料金テーブル	fare		
都道府県テーブル	prefecture		
スポット写真テーブル	spot_photo		

### 4.2.2 スポットテーブル

表 7 には、スポットに関する情報テーブルの内容を示します。

表 7: スポットテーブル

プランサポートシステム「Sutto」			作成日	2019年12月16日
プラン作成システム			作成者	中村笙悟
スポットテーブル			備考	
論理名	物理名	データ柄	NOTNULL	PK/FK
スポットID	spot_id	INT型	○	PK
スポットカテゴリID	spot_category_id	INT型	○	FK
スポットName	spot_name	VARCHAR型	○	
都道府県ID	prefecture_id	INT型	○	FK
緯度	spot_latitude	DECIMAL型	○	
経度	spot_longitude	DECIMAL型	○	
電話番号	spot_telephone	INT型		
住所	spot_address	VARCHAR型		
営業時間	spot_businesshours	CHAR型		
ホームページURL	spot_url	CHAR型		
お気に入りフラグ	spot_favorite_flag	BOOLEAN型	○	
スポット写真ID	spot_photo_id	INT型	○	FK

#### 4.2.3 プラン表示テーブル

表 8 には、プランを表示するために必要な情報テーブルの内容を示します。

表 8: プラン表示テーブル

プランサポートシステム「Sutto」			作成日	2019年12月16日
プラン作成システム			作成者	中村笙悟
プラン表示テーブル			備考	
論理名	物理名	データ柄	NOTNULL	PK/FK
プラン表示ID	show_plan_id	INT型	○	PK
ユーザID	user_id	INT型	○	FK
料金ID	fare_id	INT型	○	FK
プラン名	show_plan_name	VARCHAR型	○	
メモ	show_plan_memo	VARCHAR型		
旅行開始日	show_plan_start	DATE型	○	
旅行終了日	show_plan_end	DATE型	○	
公開フラグ	show_plan_flag	BOOLEAN型	○	

#### 4.2.4 プラン作成テーブル

表 9 には、プランを作成するために必要な情報を登録します。



表 9: プラン作成テーブル

プランサポートシステム「Sutto」			作成日	2019年12月16日
プラン作成システム			作成者	中村笙悟
プラン作成テーブル			備考	
論理名	物理名	データ柄	NOTNULL	PK/FK
プラン作成ID	make_plan_id	INT型	○	PK
スポットID	spot_id	INT型	○	FK
タイムID	time_id	INT型	○	FK
プラン表示ID	show_plan_id	INT型	○	FK
日付	make_plan_date	INT型	○	

#### 4.2.5 スポットカテゴリテーブル

表 10 には、スポットに関するカテゴリの情報を登録します。

表 10: スポットカテゴリテーブル

プランサポートシステム「Sutto」			作成日	2019年12月16日
プラン作成システム			作成者	中村笙悟
スポットカテゴリテーブル			備考	
論理名	物理名	データ柄	NOTNULL	PK/FK
スポットカテゴリID	spot_category_id	INT型	○	PK
スポットカテゴリName	spot_category_name	VARCHAR型	○	

#### 4.2.6 交通カテゴリテーブル

表 11 には、交通に関するカテゴリの情報を登録します。

表 11: 交通カテゴリテーブル

プランサポートシステム「Sutto」			作成日	2019年12月16日
プラン作成システム			作成者	中村笙悟
交通カテゴリテーブル			備考	
論理名	物理名	データ柄	NOTNULL	PK/FK
交通カテゴリID	traffic_category_id	INT型	○	PK
交通カテゴリName	traffic_category_name	VARCHAR型	○	

## 4.2.7 タイムテーブル

表 12 には、時間に関する情報を登録します。

表 12: タイムテーブル

プランサポートシステム「Sutto」			作成日	2019年12月16日
プラン作成システム			作成者	中村笙悟
タイムテーブル			備考	
論理名	物理名	データ柄	NOTNULL	PK/FK
タイムID	time_id	INT型	○	PK
プラン作成ID	make_plan_id	INT型	○	FK
ユーザID	user_id	VARCHAR型	○	FK
スポットID	spot_id	INT型	○	FK
出発時刻	time_start	DECIMAL型	○	
到着時刻	time_end	DECIMAL型	○	
交通カテゴリID	traffic_category_id	INT型	○	FK

#### 4.2.8 料金テーブル

表 13 には、料金に関する情報を登録します。

表 13: 料金テーブル

プランサポートシステム「Sutto」			作成日	2019年12月16日
プラン作成システム			作成者	中村笙悟
料金テーブル			備考	
論理名	物理名	データ柄	NOTNULL	PK/FK
料金ID	fare_id	INT型	○	PK
ユーザID	user_id	VARCHAR型	○	FK
お金1	fare_cash1	INT型		
お金2	fare_cash2	INT型		
お金3	fare_cash3	INT型		
お金4	fare_cash4	INT型		
お金5	fare_cash5	INT型		
お金6	fare_cash6	INT型		
お金7	fare_cash7	INT型		
お金8	fare_cash8	INT型		
お金9	fare_cash9	INT型		
お金10	fare_cash10	INT型		
お金11	fare_cash11	INT型		

#### 4.2.9 都道府県テーブル

表 14 には、都道府県の情報を登録します。

表 14: 都道府県テーブル

プランサポートシステム「Sutto」			作成日	2019年12月16日
プラン作成システム			作成者	中村笙悟
都道府県テーブル			備考	
論理名	物理名	データ柄	NOTNULL	PK/FK
都道府県ID	prefecture_id	INT型	○	PK
都道府県名	prefecture_name	CHAR型	○	

## 4.2.10 スポット写真テーブル

表 15 には、スポットの写真に関する情報を登録します。

表 15: スポット写真テーブル

プランサポートシステム「Sutto」			作成日	2019年12月16日
プラン作成システム			作成者	中村笙悟
スポット写真テーブル			備考	
論理名	物理名	データ柄	NOTNULL	PK/FK
スポット写真ID	spot_photo_id	INT型	○	PK
スポット写真名	spot_photo_name	CHAR型		
スポット写真URL	spot_photo_url	CHAR型	○	

### 4.3 MY プランシステム

#### 4.3.1 MY プランシステム

表 16 には、MY プラン内部データの一覧を示します。

表 16: MY プラン内部データ一覧

システム名	プランサポートシステム「Sutto」	作成日	2019年12月16日
サブシステム	MYプランシステム	作成者	中村笙悟
論理名	物理名		
写真テーブル	photo		

#### 4.3.2 写真テーブル

表 17 には、プランに追加した写真の情報を登録します。

表 17: 写真テーブル

プランサポートシステム「Sutto」			作成日	2019年12月16日
MYプランシステム			作成者	中村笙悟
写真テーブル			備考	
論理名	物理名	データ柄	NOTNULL	PK/FK
写真ID	photo_id	INT型	○	PK
プラン表示ID	show_plan_id	INT型	○	FK
写真名	photo_name	VARCHAR型		
写真URL	photo_url	CHAR型	○	

## 4.4 みんなのプランシステム

### 4.4.1 みんなのプラン内部データ一覧

表 18 には、みんなのプラン内部データの一覧を示します。

表 18: みんなのプラン内部データ一覧

システム名	プランサポートシステム「Sutto」	作成日	2019年11月30日
サブシステム	みんなのプランシステム	作成者	宮尾将史
論理名	物理名		
評価テーブル	evaluation		

### 4.4.2 評価テーブル

表 19 には、評価の情報を登録します。

表 19: 評価テーブル

プランサポートシステム「Sutto」		作成日	2019年11月30日	
みんなのプランシステム		作成者	宮尾将史	
評価テーブル		備考		
論理名	物理名	データ柄	NOTNULL	PK/FK
プラン表示ID	show_plan_id	INT型	○	PK
ユーザID	user_id	INT型	○	PK
評価	evaluation	varchar型		FK

## 4.5 広告管理システム

### 4.5.1 広告管理内部データ一覧

表 20 には、広告管理内部データの一覧を示します。

表 20: 広告管理内部データ一覧

システム名	プランサポートシステム「Sutto」	作成日	2019年11月30日
サブシステム	広告管理システム	作成者	宮尾将史
論理名	物理名		
広告テーブル	advertisement		
広告写真テーブル	advertisement_photo		

### 4.5.2 広告テーブル

表 21 には、広告に関する情報を登録します。

表 21: 広告テーブル

プランサポートシステム「Sutto」			作成日	2019年11月30日
広告管理システム			作成者	宮尾将史
広告テーブル			備考	
論理名	物理名	データ柄	NOTNULL	PK/FK
広告ID	advertisement_id	INT型	○	PK
ユーザID	user_id	INT型	○	FK
広告写真ID	advertisement_photo_id	INT型	○	FK
都道府県ID	prefectures_id	INT型	○	FK
広告名	advertisement_name	CHAR型	○	
URL	advertisement_url	CHAR型	○	
公開フラグ	advertisement_flag	boolean	○	

### 4.5.3 広告写真テーブル

表 22 には、広告写真の情報を登録します。

表 22: 広告写真テーブル

プランサポートシステム「Sutto」			作成日	2019年11月30日
広告管理システム			作成者	宮尾将史
広告写真テーブル			備考	
論理名	物理名	データ柄	NOTNULL	PK/FK
広告写真ID	advertisement_photo_id	INT型	○	PK
広告写真名	advertisement_phoro_name	CAHR型	○	
広告写真URL	advertisement_photo_url	CHAR型	○	