Snack ver1.0

内部設計書

# 内部設計

外部設計に従い、各画面の内部動作を細かく記述する。

コーディングを外注する場合は、処理単位にクラスや関数名などを明確にするとよい。

## Snackアプリ

### 新規登録処理

Post (register)

**新規登録画面**

成功

**登録成功画面**

フォーム入力された新規メンバー情報のエラーCheck

DB

(members)

に追加

失敗

## ログイン処理

Post

(login)

**ログイン画面**

成功

**マイページ面**

DB (members)　に一致するログイン情報があるかを Check

失敗

DB

(members)

から取得

### アカウント削除画面処理

Post

(member\_delete)

**アカウント**

**削除成功画面**

**アカウント**

**削除確認画面**

DB

(members)

から削除

### アカウント編集画面処理

**アカウント**

**編集画面**

Post

(member\_edit)

**アカウント**

**編集成功画面**

・nameがあるか？

成功

DB

(members)

の更新

失敗

### パスワード変更画面処理

**アカウント**

**編集画面(pass)**

**アカウント**

**編集成功画面**

**(pass)**

Post

(member\_pass\_change)

・既存passが正しいか

・新passが5文字以上か

・確認用passと一致するか

成功

DB

(members)

の更新

失敗

## おすすめする画面処理

**おすすめする**

**画面**

Post

(Recomend)

**おすすめ**

**成功画面**

フォームに入力された

スナック情報にエラーがないかcheck

成功

DB

(snacks)

に追加

失敗

## 検索処理 (条件,★,R)

Post

・search

・likes\_search

・recomender\_search

・search

:フォームに入力された条件で検索。

・likes\_search

：ユーザーがイイネしたスナックを検索。

・recomender\_search

：recomenderがおすすめしたスナックだけを検索。

**マイページ**

取得したスナックを表示

DB

(snacks)

から取得

## イイネ処理

DB

(likes)

の除去

DB

(snacks)

の更新

イイネ

削除

イイネされている

**マイページ**

Get(非同期通信)

(like\_unlike)

すでにイイネされているかcheck

DB

(likes)

から取得

イイネ

追加

イイネされていない

DB

(likes)

に追加

DB

(snacks)

の更新

## Suggest処理

Middleware

SnackSuggest

・そのメンバーのイイネの傾向から、まだイイネしていないスナックを取り出しsuggestする

**マイページ**

マイページ

にアクセス

取得した

スナックを表示

DB

(snacks)

から取得

DB

(likes)

から取得

## スナック削除画面処理

＊そのメンバーがおすすめしたものだけアクセスできる

Post

(snack\_delete)

**スナック**

**削除成功画面**

**スナック**

**削除確認画面**

DB

(snacks)

から削除

## スナック編集画面処理

＊そのメンバーがおすすめしたものだけアクセスできる

**スナック**

**編集画面**

Post

(snack\_edit)

**スナック**

**編集成功画面**

・入力されたスナック情報のエラーcheck

成功

DB

(snacks)

の更新

失敗

## 管理者ログイン画面処理

＊id=1のログインメンバーのみアクセスできる

。

**管理者**

**ログイン画面**

**スナック**

**管理画面**

Post

(administrator\_login)

入力された管理者ログイン情報が正しいかcheck

成功

失敗

## スナック管理処理

Post 非同期通信

(search)

**スナック**

**管理画面**

**(search後)**

**スナック**

**管理画面**

＊deletion=1のスナックは、マイページで検索しても表示されない。

。

DB

(snacks)

から取得

・snacksのdeletionが0であれば、「非表示にする」ボタン。1なら「表示する」ボタン。

Post

(表示する)

Post

(非表示にする)

そのスナックのdeletionを1にする。

そのスナックのdeletionを0にする。

DB

(snacks)

を更新

DB

(snacks)

を更新

## メンバー管理処理

Post非同期通信

(search)

**メンバー**

**管理画面**

**(search後)**

**メンバー**

**管理画面**

＊deletion=1のメンバーはログインができない。

。

DB

(members)

から取得

・membersのdeletionが0であれば、「非表示にする」ボタン。1なら「表示する」ボタン。

Post

(表示する)

Post

(非表示にする)

そのメンバーのdeletionを1にする。

そのメンバーのdeletionを0にする。

DB

(members)

を更新

DB

(members)

を更新

English ver

## Snack ver1.0

### New Registration processing

**New**

**Registration**

**Screen**

**Registration Success Screen**

Post (register)

Error Check for new member information entered in the forms.

success

DB

(members)

add

failure

## 1.2.2 Login processing

Check if there is matching login information in

DB (members).

Post

(login)

**Login Screen**

success

**Mypage/home**

failure

DB

(members)

get

### 1.2.3 Account Deletion processing

**Account Deletion Confirmation Screen**

Post

(member\_delete)

**Account Deletion**

**Success Screen**

DB

(members)

delete

### 1.2.4 Account Edit Screen processing

**Account Edit Screen**

Post

(member\_edit)

**Account Edit Success Screen**

・Is there a 「name」?

success

DB

(members)

update

failure

### 1.2.5 Password Change Screen processing

・Is the current pass correct?

・Is the new pass more than 5 characters?

・Does it match the confirmation pass?

**Account Edit Screen (pass)**

**Account Edit Success Screen (pass)**

Post

(member\_pass\_change)

success

DB

(members)

update

failure

## 1.2.6 Recommend Screen processing

**Recommend**

**Screen**

Post

(Recommend)

success

**Recommend Success Screen**

Error Check for new snack information entered in

the forms.

DB

(snacks)

add

failure

## 1.2.7 Search processing (condition,★,R)

Post

・search

・likes\_search

・recomender\_search

・search

: Search by the criteria entered in the form.

・likes\_search

：Search for snacks that users have liked.

・recommender\_search

：Search only for snacks recommended by recommender.

**Mypage/home**

Show acquired snacks.

DB

(snacks)

get

## 1.2.8 Like processing

Like

Delete

DB

(likes)

delete

DB

(snacks)

update

Already Liked

**Mypage/home**

Get

(like\_unlike)

Check if you already have a "like".

DB

(likes)

get

Like

Add

Unliked yet

DB

(likes)

add

DB

(snacks)

update

## Suggest processing

Middleware

SnackSuggest

・Select snacks that have not yet been "liked" and suggest them based on the "like" trend of the member.

Access

Mypage

**Mypage**

Show acquired snacks.

DB

(snacks)

get

DB

(likes)

get

## 1.2.10 Snack Deletion Screen processing

＊Access only what that member recommends.

Post

(snack\_delete)

**Snack Deletion Success Screen**

**Snack Deletion Confirmation Screen**

DB

(snacks)

delete

## 1.2.11 Snack Edit Screen processing

＊Access only what that member recommends.

**Snack Edit**

**Screen**

Post

(snack\_edit)

**Snack Edit**

**Success Screen**

・Error check of snack information entered in the forms.

success

DB

(snacks)

update

failure

## 1.2.12 Administrator Login Screen processing

＊Only logged-in members with id=1 have access.

**Administrator Login Screen**

**Snack Administration Screen**

Post

(administrator\_login)

・Check if the entered administrator login information is correct.

success

failure

## 1.2.13 Snack Administration processing

**Snack Administration Screen (after search)**

**Snack Administration Screen**

Post

(search)

＊Snacks with deletion=1 are not displayed when searching in My Page.

DB

(snacks)

get

・If the deletion of the snack is 0, it is the "非表示にする" button; if it is 1, it is the "表示する" button.

Post

(表示する)

Post

(非表示にする)

Set the deletion of that snack to 1.

Set the deletion of that snack to 0.

DB

(snacks)

update

DB

(snacks)

update

## 1.2.14 Member Administration processing

**Member Administration Screen (after search)**

**Member Administration Screen**

Post

(search)

＊Members with deletion=1 cannot login.

。

DB

(members)

get

If the deletion of members is 0, the button is "非表示にする"; if it is 1, the button is "表示する".

Post

(表示する)

Post

(非表示にする)

Set the deletion of that member to 1.

Set the deletion of that member to 0.

DB

(members)

update

DB

(members)

update