

システムテストの補足

1. 各区分の詳細内容

機能テスト内容

機能テスト

機能が要件通り達成できるか

項目一覧

- | | |
|----------|--------------|
| ・インストール | ・フードバンク |
| ・アカウント作成 | ・食品登録(文字入力) |
| ・ログイン | ・食品登録(バーコード) |
| ・ログアウト | ・食品一覧の閲覧 |
| ・設定 | ・食品の削除 |
| | ・通知 |
| | ・レシピ |
| | ・再ログイン |

性能テスト内容

性能テスト

読み込みなどの時間が短いか、アプリやサーバー性能

項目一覧

- ・インストール
- ・アカウント作成
- ・ログイン
- ・ログアウト
- ・設定
- ・フードバンク
- ・食品登録(文字入力)
- ・食品登録(バーコード)
- ・食品一覧の閲覧
- ・食品の削除
- ・通知
- ・レシピ

操作性テスト内容

操作性テスト

その機能が使いやすいか、目的を達成することが簡単か

項目一覧

- ・アカウント作成
- ・ログイン
- ・ログアウト
- ・設定
- ・フードバンク
- ・食品登録(文字入力)
- ・食品登録(バーコード)
- ・食品一覧の閲覧
- ・食品の削除
- ・通知
- ・レシピ

例外処理テスト内容

例外処理

通常起こりにくい処理に対応できるか

項目一覧

- ・アカウント作成時
 - ・id名に特殊文字と長文
 - ・パスワードに特殊文字と長文
 - ・郵便番号に数字以外や存在しない郵便番号を含める
- ・不一致のidパスワードでログイン
- ・設定時
 - ・パスワードを特殊文字や長文に変更
 - ・郵便番号に数字以外や存在しない郵便番号を含める
- ・食品登録時
 - ・期限を数年後や未記入
 - ・食品名の長文入力と未記入
- ・レシピ閲覧時
 - ・長文や意味不明な食品
- ・通信エラー時
 - ・アプリ立ち上げ前から
 - ・アプリ立ち上げ中に

負荷テスト内容

負荷テスト

アプリに負荷を与えたときに正常に動作するか

項目一覧

- ・短時間に食品の削除
- ・ボタンの連打
- ・連続でパスワード変更
- ・アプリの立ち上げと終了を連続で実行

2. エラーポイントの算出方法

エラーポイントの算出方法

各検証項目に対して【大】、【中】、【小】、【無し】の4段階で課題の大きさをテスト実施者が判断する。

課題の大きさ評価例として以下を示した。

- ・【大】→利用できない
- ・【中】→利用できるが不満を感じる
- ・【小】→利用には及第点であるが、既存の他アプリに比べ劣る
- ・【無し】→他のアプリと遜色なく、十分に利用できる

また、はデバイス非対応時、前提機能が利用できないときには【検証不可】と判断する。
【大】は1点、【中】は2/3点、【小】は1/3点、【無し】は0点とし、テスト実施者全員(3人)の平均を各項目のエラーポイントとした。

テスト実施者全員が【検証不可】とした場合、その項目は「検証なし」とした。

3. 品質の評価方法

品質の評価方法

各検証項目に対して「良」、「可」、「不可」の3段階で品質を評価した。

品質の目安

- ・「良」→問題なし
- ・「可」→概ね問題ないが改善が必要
- ・「不可」→改善が早急に必要

品質の評価基準

- ・「良」→エラーポイントが0.167未満かつ
テスト実施者の課題大きさの回答に【大】または【中】がない場合
- ・「可」→エラーポイントが0.66未満かつ
テスト実施者の課題大きさの回答に【大】がない場合
- ・「不可」→「良」又は「可」でない場合

4. テスト01と02での項目の違い

テスト01と02の項目の違い

テスト01実施後、改善が必要な機能を絞った

削除した機能と理由

- ・idの変更 → あったら便利だが絶対必要ではないから
- ・食品カテゴリーの設定 → あったら便利だが絶対必要ではないから
- ・iPadの画面分割 → iPad利用者や分割画面は優先順位として低いから