

13 情報処理試験対策アプリの作成

下川原 祐太, 出堀 迅

指導教員 石舘 勝好

1. はじめに

現在配信されている情報処理試験対策アプリは、既存の過去問を解いていくアプリが多いのが特徴である。そこで私たちは単語の暗記と練習問題の両方を行え、単語の登録・編集を自分で行い練習問題の作成も自分で登録した単語のデータから自動で問題を作るアプリの開発をする。

2. 動作環境

本アプリは Android4.0 以上のスマホ,タブレットに対応している。
開発環境を以下に示す。

図 1 開発環境

| 開発環境 | |
|---------|------------------|
| OS | Windows7 |
| IDE | Android studio |
| 言語 | Java8 |
| SDK | Android SDK |
| JDK | JDK1.8.0_92 |
| デバック用端末 | Androidver 5.1.1 |

3. システム概要

3.1 アプリの内容

- 基本情報処理試験, 情報セキュリティマネジメントに対応したアプリになっている。
- 単語の登録・編集・削除が行える
- 登録した単語は CSV ファイルとして保存している。
- 単語の登録はスマホからの入力で行う。コピー&ペーストを使用できる。
- 登録した単語で練習問題を解くことが出来る。

3.2 画面レイアウトおよび遷移

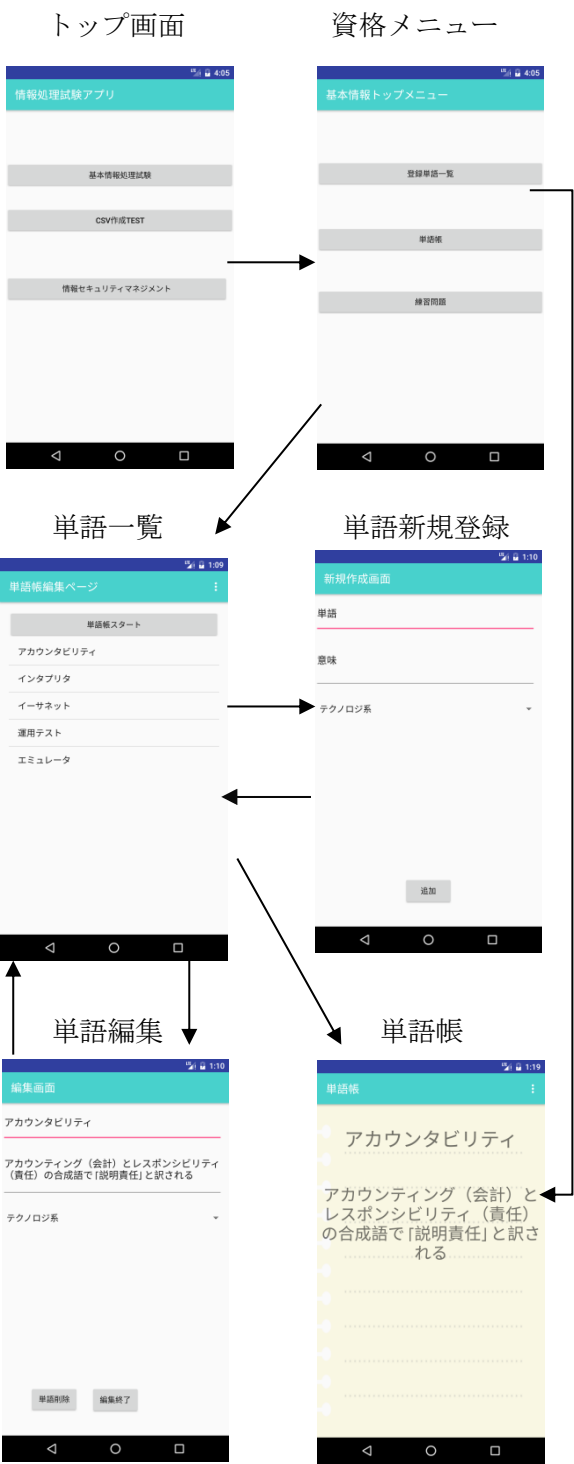
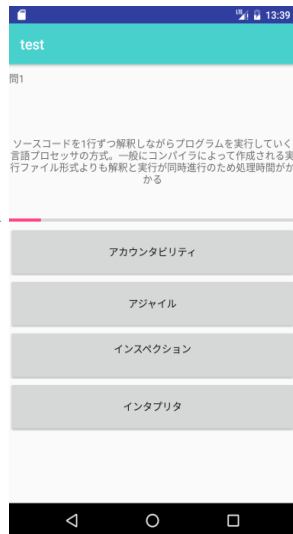


図 2-1 画面レイアウトおよび遷移

練習問題トップ



4 択出題画面



結果画面



図 2-2 画面レイアウトおよび遷移

3.4 単語の管理について

アプリ起動時に CSV ファイルを作り, そのファイルを上書きしていくことで単語を登録していく。

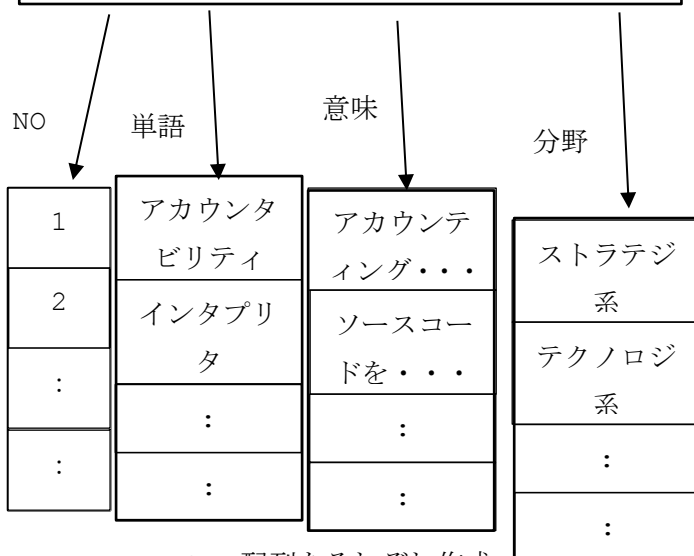
データを呼び出すときは自作クラスで CSV ファイルの各列ごとに ArrayList 型のフィールドを用意しアプリ初期化時に CSV ファイルから読み込む。

以下にイメージ図を示す。

NO, 単語, 意味, 分野

- 1, アカウンタビリティ, アカウンティング・・・, ストラテジ系
- 2, インタプリタ, ソースコードを・・・, テクノロジ系

：, ：, ：, ：
：, ：, ：, ：



ArrayList 配列をそれぞれ作成

図 4 単語の管理

3.3 アプリの流れ

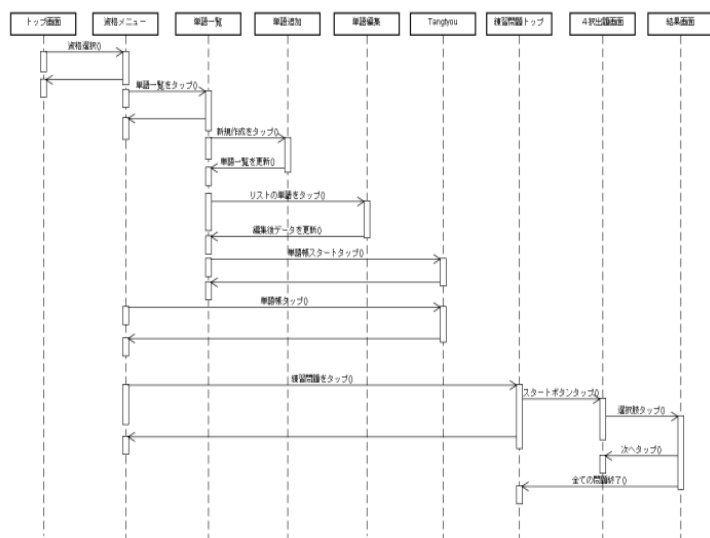


図 3 アプリの流れ

3.5 各クラスについて

TopActivity: アプリ起動時のトップ画面. 資格の選択.

Shikakumenu: 資格選択後の画面. 単語の登録や練習問題の画面へ遷移.

MainActivity: 登録済みの単語一覧を表示. 単語の新規登録や編集画面へ遷移. ボタンタップで単語帳画面へ遷移. メ

ニューから新規作成, ソートが可能.

addData: 単語の新規登録. ブラウザからのコピー&ペーストも可能. 分野はボックスからの選択.

editdata: 登録されている単語を編集.

機能は addData とほぼ同じ.

Tangotyou: 登録されている単語を単語帳のように表示. スワイプで次の単語へ. 単語を検索することでその単語のページへとぶこともできる.

Select_time: 練習問題トップ画面.

問題数, 回答時間, ジャンルの選択をしボタンタップで問題がスタートする.

Test: 4択形式の問題を出題. 問題のテキストは登録されている意味を表示してその意味に合う単語を回答していく形式となる. タイムバーは最初に設定された制限時間で進み具合が変わる.

Result: 最初のテキストには答えの単語を表示. その下には意味を表示. 問題終了ボタンタップで途中で問題を終了できる. 矢印ボタンタップで次の問題へ. Test ページと Result ページを行き来するのようなイメージ.

クラス図を図 5 に示す.

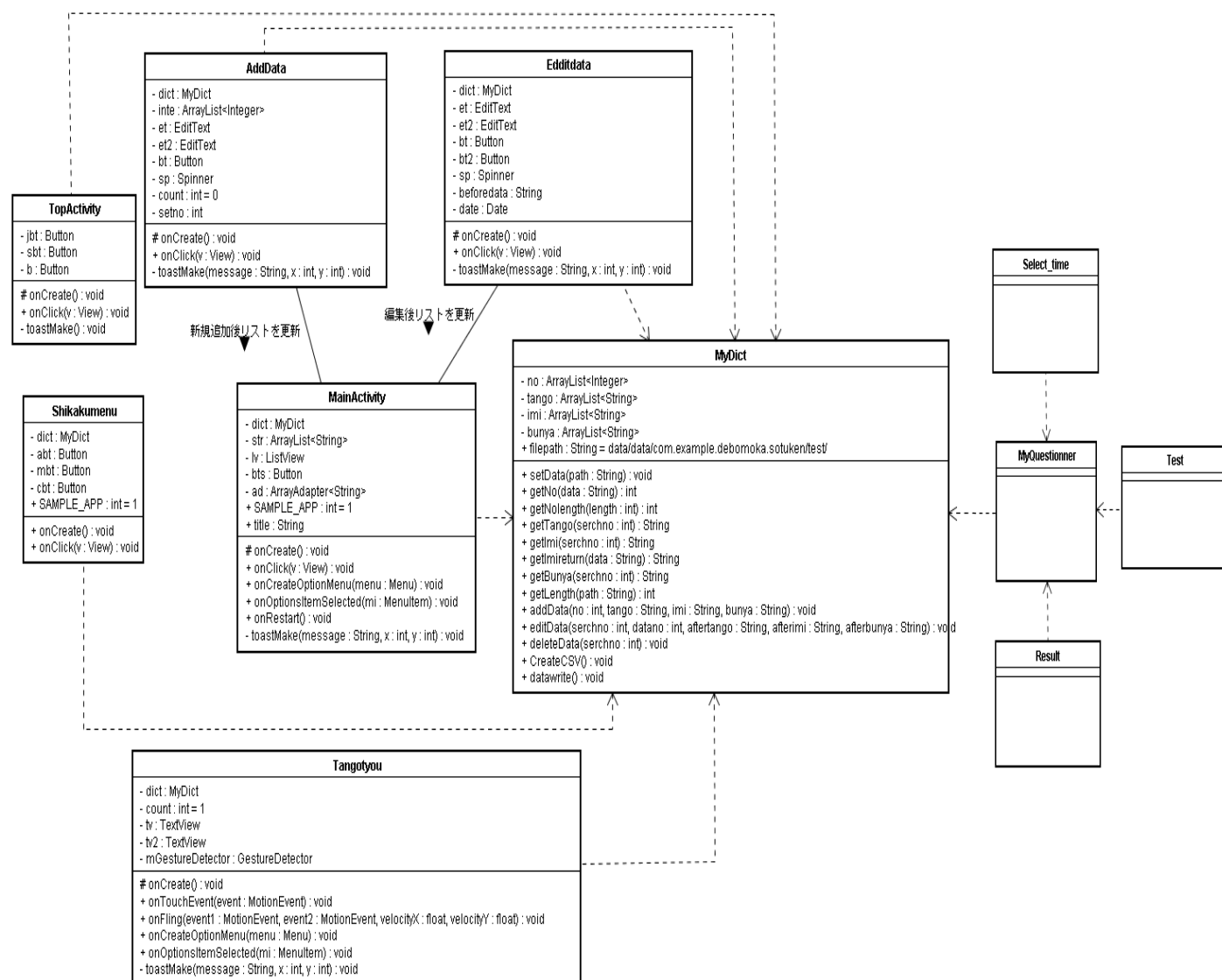


図 5 クラス図

3.6 MyDict・MyQuestionner クラスについて

(1)MyDict

単語を管理するクラス.

ほかのクラスで MyDict クラスの get・set 等のメソッドを呼び出すことで単語の登録・編集・削除が容易に行える.

コンストラクタで単語のデータを管理する ArrayList 配列を用意する.

アプリ起動時・終了時に CSV ファイルを読み書きする.

(2)MyQuestionner

出題する問題表を作成するクラス. MyDict に登録されているデータから問題用紙のようなものを作成する.

練習問題トップ画面で選択された出題パターンをもとに図のような問題の表を作る. 4 択出題画面ではこの表を 1 問目から順に読み込んでいく. 正誤の欄には実際に解いたときに正解であれば”○”, 不正解であれば”×”を入れていく.

| No | 出題(意味) | 正答 | ダミーの解答 | | | 正誤 |
|----|------------|-----------|--------|---------|-------|----|
| 1 | アカウント・グ... | アカウントビリティ | CPU | アジャイル | アセンブラ | |
| 2 | ソースコードを... | インタプリタ | TCP/IP | エミューレータ | WEP | |
| : | : | : | : | : | : | |

図 6 出題表イメージ図

4. 進捗状況

4.1 実装済み機能

- 単語の管理(登録・編集・削除)
- 単語帳の表示
- 出題パターンの選択

4.2 未実装の機能

- 問題の出題機能
- 回答後の結果画面

4.3 実機テスト

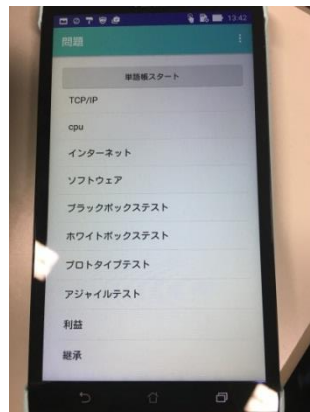


図 7 スマートフォンでの動作画像

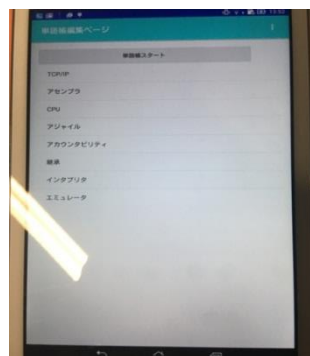


図 8 タブレットでの動作画像

5. 今後の予定

問題出題機能の完成, レイアウトを決める, 細部の機能の調整が主な課題となっている.

問題の出題方法が今はランダムで 4 択の選択肢が被らないように表示しているが, 分野を指定したりある程度の規則性をもってランダムに出題できるようにするなどの拡張性を持っているので機能を増やしていきたい.

5. 参考文献

- 黒帯エンジニアが教えるプロの技術 Android 開発の教科書
著者 筒井 俊祐 他
出版社 SB クリエイティブ
- やさしい Android プログラミング第 3 版
著者 高橋麻奈
出版社 SB クリエイティブ