

# 要件定義書

—出席簿アプリ—

5CS29 中作眞仁

## 目次

|                                     |   |
|-------------------------------------|---|
| 1. これまでの問題点                         | 3 |
| 2. ソフトウェアの概要（機能一覧）                  | 3 |
| 3. ソフトウェアの全体の機能構成（機能の説明、流れのフローチャート） | 3 |

## 1. これまでの問題点

顧客が抱える問題点を挙げると、以下のようなものがある。

- ・教員によるデータ入力ミス
- ・教員間でのデータずれ
- ・学生側の確認不足
- ・出席日数が少ない人の洗い出し困難
- ・集計が大変

これらの問題起因を分析すると、出席確認が複数の紙媒体で行われており、1 ページあたりの情報量が多いことであるとわかる。解決策として、データベースをバックエンドに導入し、データの連携を行い、1 ページごとの情報を削減する必要があると考えられる。そのため、シンプルな UI とバリデーション表示を組み合わせ、表の列を減らして最低限の情報表示を行うことで、見やすく利便性の高いシステムを構築することが重要であると考えられる。ただし、出席登録には時間をかける余裕がないため、タッチ回数を最小限に抑える方針を考えている。

## 2. ソフトウェアの概要（機能一覧）

以下に機能の一覧を示す

- (ア) 生徒の名前と番号を登録する
- (イ) 教科と先生の名前、前期後期を登録する
- (ウ) 日々の出欠席登録
- (エ) データの編集
- (オ) それぞれの生徒のデータ分析と集計

## 3. ソフトウェアの全体の機能構成（機能の説明、流れのフローチャート）

図 1 に生徒情報や教科の情報を入力する場面から出欠席登録を行うまでのフローチャートを示す。また、(ア) から (オ) は上記に 2 と同じである。

(ア) は教務課から提供される生徒情報を入力する場面である。

(イ) も (ア) と同様に、提供された予定表をデータとして入力する。これにより、日々の入力回数を減らすことができると考えている。

(ウ) は教員が日々のデータ入力を行う場面で、1 日分のみのデータ編集を行う。過去や先のデータに手を加えないようにするためである。また、登録の UI は表ではなく、on/off ボタンを利用してデータ入力ミスを減らすことができると考えている。

(エ) は管理者画面であり、ユーザーとパスワードで保護され、教員のみが編集できるようになる予定である。この画面ではすべてのデータを編集することができる。

(オ) この画面では登録後に移動する設計であり、登録をしなくてもアクセス可能である。授業ごとのデータを表示し、生徒情報の詳細な集計も表示する予定である。

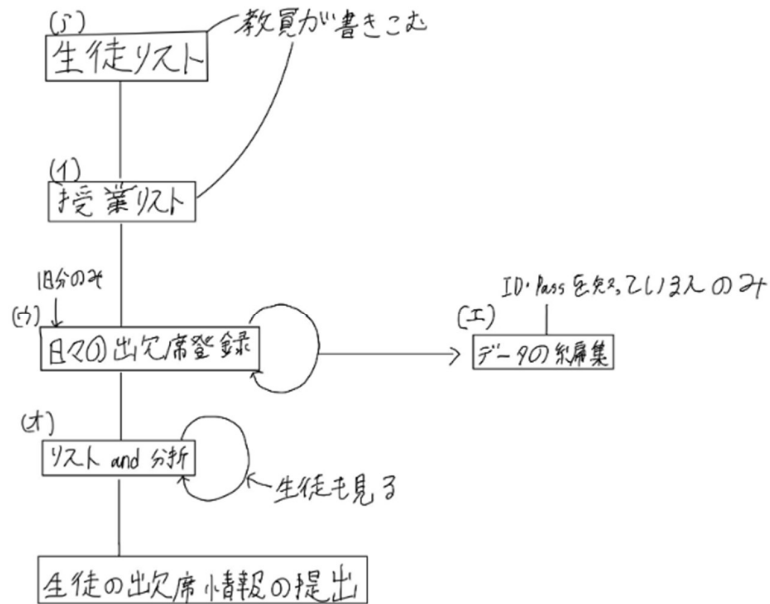


図1、フローチャート