

CSV Formatter Project Summary / App PRD

[Overview](#)

[Background](#)

[Requirements](#)

[UI & Screens](#)

[CSV Format & Samples](#)

[User Story with Screen Images & Acceptance Criteria](#)

[Tickets](#)

Overview

Manabie 塾ERP（カスタムSalesforce）で使用する、生徒の一括クラス配置のためのCSV作成を簡単に行うシンプルなウェブアプリ。

☒ Write (this) document

☒ Make app & test

☒ Deploy on GitHub

☐ Feedback & Improve

Background

ERPシステムでは、各「拠点コース」オブジェクトの下にユニークな「クラス」オブジェクトが作られている。「クラス」は同じ名前であっても親の「拠点コース」が違えば別オブジェクトなので、システムの内部的には名前に関わらずユニークなIDが割り振られている。基本的にシステムをユーザーが使う際にはこの「クラス」を選択時に親の「拠点コース」が既に指定されている状態で、従ってシステム上で選択できるクラスはその「拠点コース」の「クラス」のみとなる。

現在ERP上の業務フローとして、「ある拠点コースを取る生徒に、特定のクラスを割り当てる」という作業がある。UI上では「違う拠点コースを取る複数の生徒に対して、クラスの割り当てを一括で行う」処理ができないため、CSVインポートを使っている。しかし、このCSVが上記の出力ファイルにあたり、生徒に割り当てたいクラスの指定をIDで行わなくてはならず（親拠点コースの違うクラス名の重複をを許可しているため）、「欲しい拠点コースのクラスIDを調べて、それをインポートするCSVに貼り付ける」という人間のユーザーにとってはミスの発生しやすい作業を手作業で行うことになってしまっている。

そこで、アプリ上では「ユーザーは生徒情報（生徒名、拠点コース名、Lesson Allocation ID）のテーブルを見ながら、各行（生徒×拠点コース情報）に対しクラス名を選択するだけで、自動的にアプリ内でその拠点コースのクラスIDを挿入したCSVを作成する」という処理を行いたい。

Requirements

利便性と開発の簡単さを考えて、Next.jsフレームワークでのアプリ開発を行う。

1. 2つの異なるCSVファイルをユーザーにアップロードさせる。
2. 内容のデータを読み取り、必要な項目を表示・取得。
3. テーブルをユーザーに編集させ、完了したらCSVとしてエクスポートできるようにする。

UI & Screens

<https://www.figma.com/design/SsU8az6hzpmOxdgSf1uoi8/%E3%82%AF%E3%83%A9%E3%82%B9%E3%83%A1%E3%83%B3%E3%83%90%E3%83%BCCSV%E3%82%A2%E3%83%97%E3%83%AA?node-id=0-40454&t=MVj8zRjaMfexPUxq-1> [Connect your Figma account](#)

CSV Format & Samples

入力ファイル1：生徒名、拠点コース名、Lesson Allocation ID

生徒名	Location Course: 拠点コース	授業の割当て: 授業の割当ID
山田 太郎	コース1 - 拠点1	104853
川口 一郎	コース1 - 拠点2	193148
山田 太郎	コース2 - 拠点1	602954
鈴木 花子	コース1 - 拠点1	349584

入力ファイル2：拠点コース名、クラス名、クラスID

拠点コース: 拠点コース	クラス: クラス	クラス: ID
コース1 - 拠点1	クラス1	a0IGA00001zXcbb
コース1 - 拠点2	クラス1	a0IGA00001zXtYw
コース2 - 拠点1	クラス1	a0IGA00001zXtbn
コース1 - 拠点1	クラス2	a0IGA00e01zXtbn


出力ファイルフォーマット

生徒名	Lesson Allocation	拠点コース名	クラス	開始日
山田 太郎	104853	コース1 - 拠点1	a0IGA00001 zXcbb	2025-04-30T14:00:00.000+0900

User Story with Screen Images & Acceptance Criteria

まずはコアシステムの完成が優先。入力CSVは基本的にSFのエクスポートで正しいデータをそのまま取り出しているはずなので、ミスは多くはないものとして想定する。よって複雑なバリデーションチェックやエラーハンドリングによるユーザーフレンドリーな警告などの実装は、後回しにする。

US	Detailed US	Acceptance Criteria	Error Handling
1	<p>1. 1つ目のCSVファイル（入力ファイル1）がアップロードができる</p>  <p>2. 入力ファイル1の内容のプレビュー、「戻る」「次へ」ボタンが表示される</p>  <p>3. 「次へ」をクリックするとUS2へ進む</p>	<ul style="list-style-type: none"> 入力ファイル1のアップロード画面のプログレスバーは1 入力ファイル1のバリデーションチェック： <ul style="list-style-type: none"> ラベルが「生徒名」「Location Course: 拠点コース」「授業の割当て: 授業の割当てID」かどうか (Low Priority: SF自体にバリデーションチェックがあるので、このアプリでそこまで厳しくしなくてもいいかも) データ抜け漏れのあるレコード、重複データのチェック US1-1のバリデーションチェックが成功したらUS1-2へ プレビューでは、入力ファイル1の最初5行を表示して、ユーザーがアップロードしたファイルが間違っていない 	<ul style="list-style-type: none"> CSV以外は受け付けない バリデーションエラーが出たら処理を中断、エラーメッセージ「不正なファイルです」を表示して再アップロードを促す（プレビュー画面へ移行しない） (Low Priority) バリデーションエラー時に「何が問題か」を検出し表示する

		<p>いか確認できるようにする。</p> <ul style="list-style-type: none"> ・プレビュー画面で「戻る」をクリックすると入力ファイル1のアップロード画面へ戻る。 ・入力ファイル1プレビュー画面のプログレスバーは1.5 	
2	<p>明記しない仕様はUS1と同様</p> <ol style="list-style-type: none"> 1. 2つ目のCSVファイル（入力ファイル2）がアップロードできる 2. 入力ファイル2の内容のプレビュー、「戻る」「次へ」ボタンが表示される 3. 「次へ」をクリックするとUS3へ進む 	<ul style="list-style-type: none"> ・ファイル1と同じく各種チェックするが、ラベルは「拠点コース」「クラス」「クラスID」 	
3	<p>クラス配置CSV作成くん</p>  <ol style="list-style-type: none"> 1. テーブル上の生徒をフィルタリングできる <ol style="list-style-type: none"> a. 生徒ごと b. 拠点コースごと 2. テーブル上の生徒をチェックボックスで選択/選択解除できる 3. 選択した生徒に対して日付をカレンダーから一括入力できる 4. 生徒のクラスプルダウンからその生徒のクラス選択ができる 5. 「入力値を全てクリア」ボタンを押すと、確認画面が出て了承する 	<ul style="list-style-type: none"> ・テーブル画面のプログレスバーは3 ・入力ファイル2から拠点コースごとにクラス名とクラスIDを取得し、テーブルのその拠点コースに所属する生徒のクラス行ではプルダウンで存在するクラスを選択できるようにする。その際、システム内では適切なクラスIDを当てる。 ・テーブル表示生徒のフィルタリング <ul style="list-style-type: none"> ◦ 取得した生徒でプルダウンリストを作成。選択した生徒で表示生徒をフィルタリング <p>取得した拠点コースでプルダウンリストを作成。</p>	

とそこまで入力したクラスと日付を全て未選択状態に戻せる

6. エクスポートボタンを表示して、押すとその状態のテーブルをCSVとしてDLできるようにする

選択した拠点コースで表示生徒をフィルタリング

- 両方のフィルタリングが同時に使用されている場合、ANDフィルタリングする
- 生徒選択チェックボックス
 - 一人以上同時に選択できる
 - 全選択チェックボックスはフィルタリングで表示されている生徒のみ選択
 - 全選択解除チェックボックスはフィルタリングで表示されていない生徒も選択解除
- クラスプルダウン
 - 一番上は常に「▼ 選択してください」（未選択用オプション）
 - プルダウンに表示するクラス名を正しく親拠点コースでフィルタリングする。
 - マウスポインタの選択中、プルダウンの項目の色を変える
 - 一括変更しなくてよい。そのプルダウンのある行の生徒のみ対象
- 日付カレンダー：
 - マウスポインタの選択中、プルダウンの項目の色を変える
 - 生徒が選択されていない場合、その行の生徒のみに変更を適用

		<ul style="list-style-type: none"> 。生徒が選択されている場合、選択中の生徒全てに変更を適用 。カレンダーで年月日まで入力可能とし、日付フォーマットの残りの部分はシステムで固定で挿入 (T14:00:00.000+0900) 。 「入力値を全てクリア」 ボタンを押すと、確認ポップアップが表示。「キャンセル」で何もせずにポップアップを閉じる。「消去する」で入力したクラスと日付を全て未選択状態に戻す 。 エクスポートファイルにテーブル内容を反映し、各種レコードを適切なフォーマットに変換してCSVとして生成する。 	
--	--	---	--

Tickets

Type	Summary	Assignee	Priority	Status	Due date
✖	全選択解除ボックス		Medium	TO DO	Oct 8, 2025
✖	戻るボタン		Medium	TO DO	Oct 8, 2025

 Synced just now • [2 items](#)