

2441054 田中知来

2441095 横尾 実幸

2441070 橋田春樹

2441081 堀恵颯真

2441008 一原大起

グループ名「学生管理アプリ」

<https://github.com/g2441081/->

学生管理アプリケーション の構築

Flask / PostgreSQL / Docker による統合管理システム

グループ名：学生管理アプリ

1. 開発の背景と目的

- ・ 学生は複数の講義と課題を抱えており、情報の断片化が課題。
- ・ 出席数不足による単位失得のリスクを未然に防ぐ必要がある。
- ・ 「いつまでに何をすべきか」を一つの画面で完結させることを目指した。

2. システム概要

学生管理アプリ

+ 新規登録

授業名(必須)
例: 数学

課題内容
例: レポート

締切日時
年 / 月 / 日 --:--

追加

2026年1月

today < >

日	月	火	水	木	金	土
28日	29日	30日	31日	1日	2日	3日
4日	5日	6日	7日 ● 21:50 英語: レ...	8日	9日	10日
11日	12日	13日	14日	15日	16日	17日
18日 ● 23:50 数学: 課題	19日	20日	21日 ● 21:51 プログ...	22日	23日	24日
25日	26日	27日	28日	29日	30日	31日

授業・課題・出席リスト

プログラミング

出席: 3回

+ -

レポート

締切: 2026-01-21 21:51

完了

国語

出席: 1回

+ -

課題はありません

削除

数学

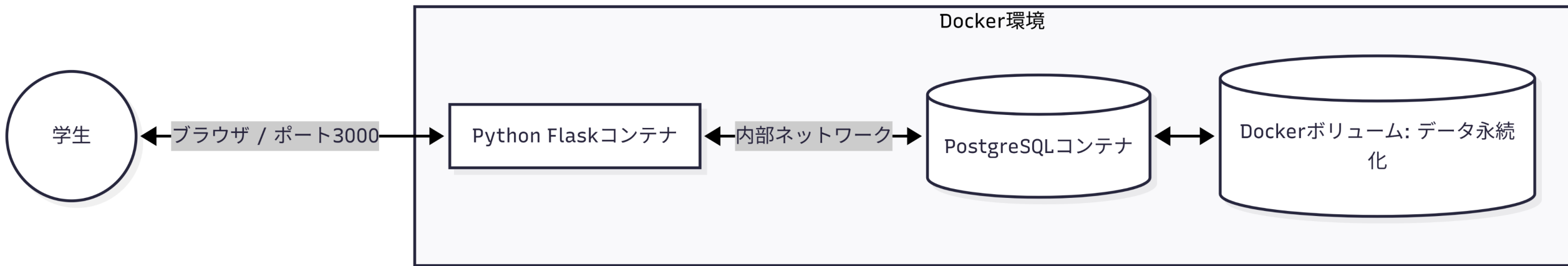
出席: 2回

+ -

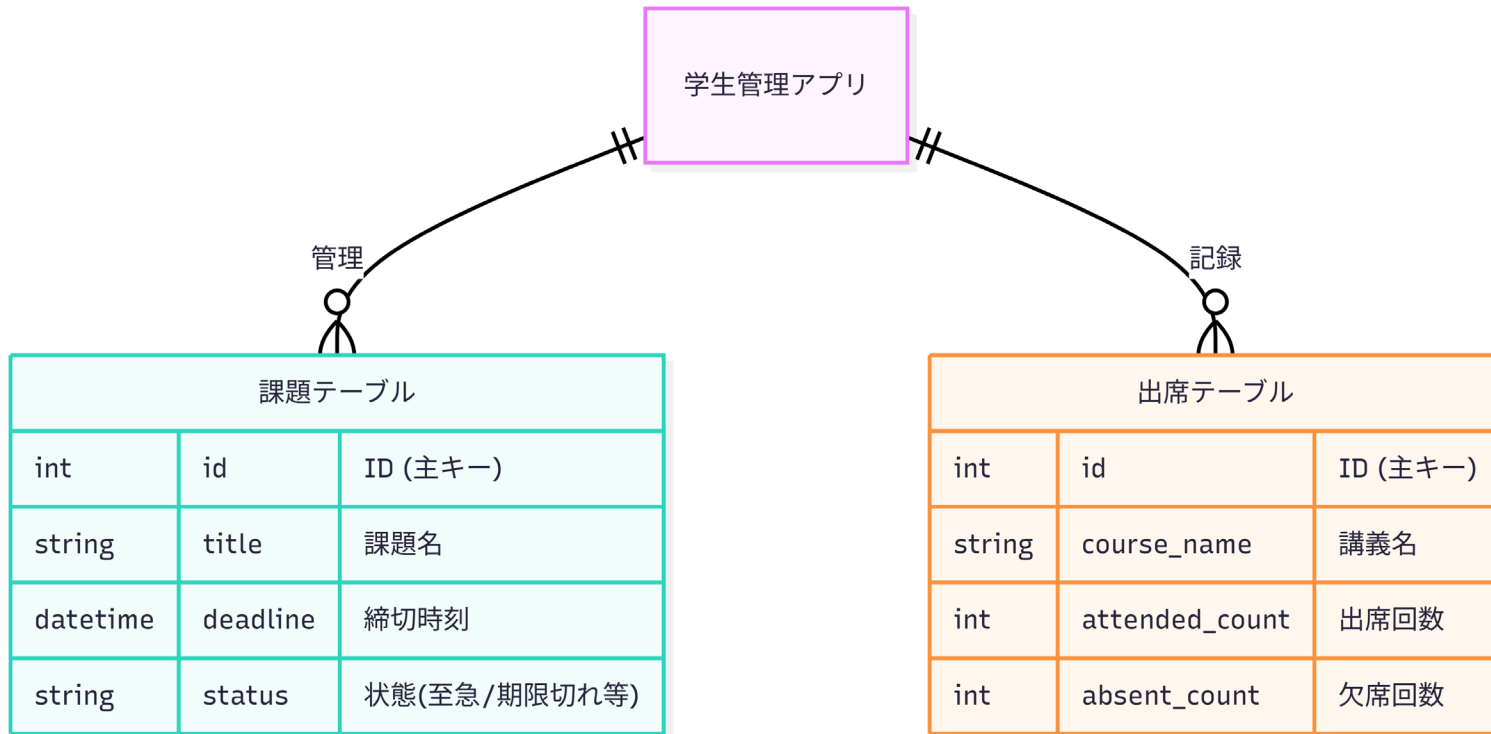
- ・ 課題管理：カレンダーUIを用いた締切の可視化と優先順位付け。
- ・ 出席記録：講義ごとの出席・欠席状況をリアルタイムに集計。
- ・ 一元化：複数の情報を一つのダッシュボードに統合し、確認漏れを防止。

3. システム構成 (Docker)

- ・ Docker Compose : 環境構築の自動化と再現性を確保。
- ・ Docker Network : WebサーバーとDBサーバーを仮想ネットワークで分離。
- ・ 永続化 : Volumes設定によりコンテナ破棄後もデータが残る設計。



4. データベース設計 (PostgreSQL)



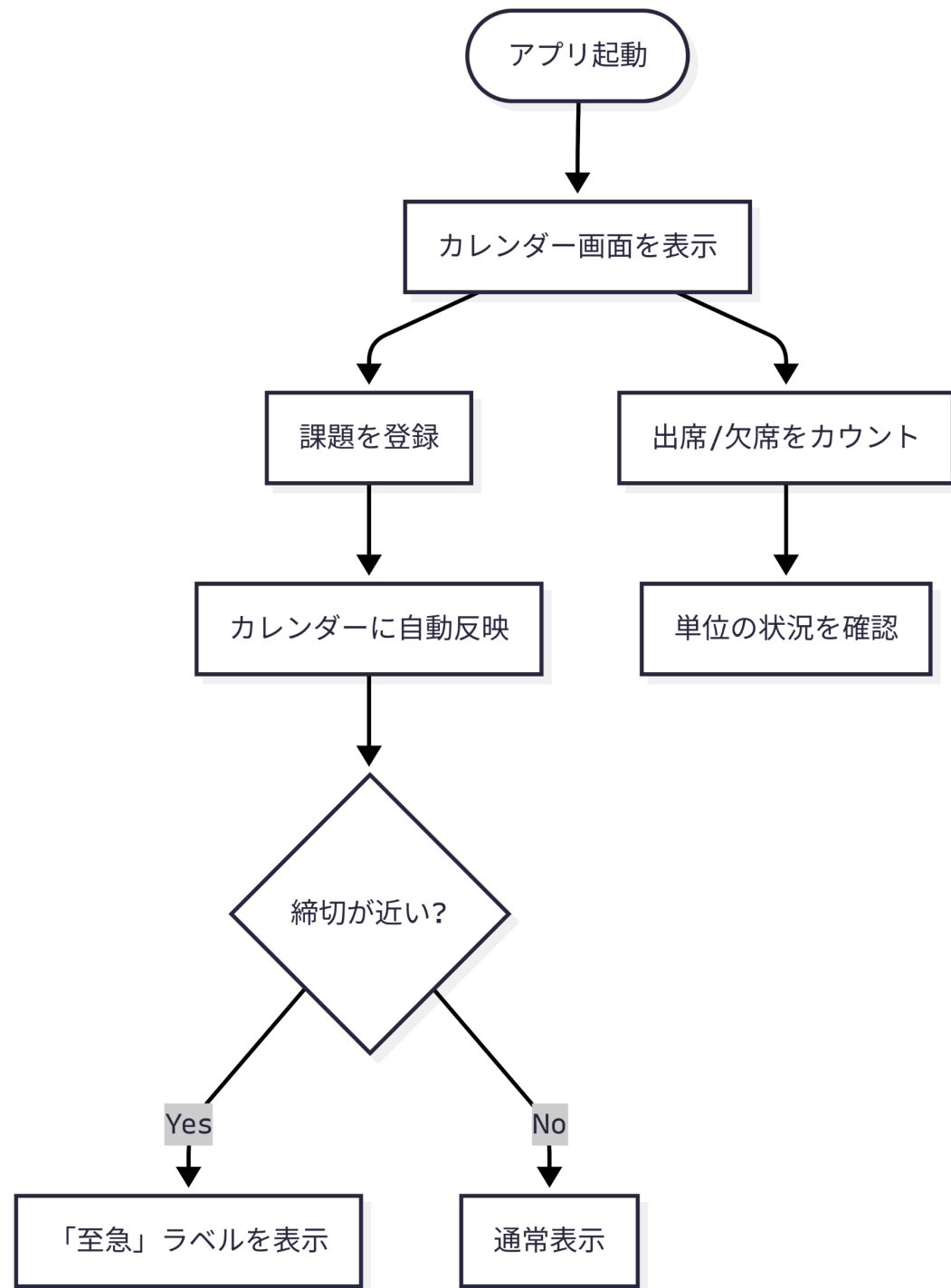
- ・ SQLによる厳密なデータリレーション管理。
- ・ 課題テーブル：タイトル、締切日、ステータスを保持。
- ・ 出席テーブル：講義ID、出席回数、欠席回数を保持。

5. 実装上の工夫

- ・ 動的ステータス管理：SQLクエリと現在時刻を比較し、ラベルを自動判別。
- ・ 非同期描画：FullCalendarを活用し、ページ遷移なしで課題を表示。
- ・ API設計：Flaskによる軽量かつ拡張性の高いサーバーサイド実装。

6. 動作デモンストレーション

- ・ 課題の追加とカレンダーへの即時反映。
- ・ 出席ボタンによるカウントアップ機能。
- ・ ※詳細は同梱の demo_video.mp4 を参照。



7. 結論と今後の展望

- ・フルスタックなWeb開発環境の構築に成功。
- ・今後の展望：スマートフォン通知機能の実装、通知APIとの連携。
- ・教員用ダッシュボードとの連携による双方向通信の実現。

8. 成果物およびリポジトリ

GitHub URL: <https://github.com/g2441081/->

- ・実装コード：/student-app
- ・デモ動画：demo_video.mp4
- ・本資料：学生管理アプリ.pdf