

データベース
グループ名：赤べこ

2441062 永井 昊

2441034 小島寛斗

2441049 高橋昂聖

アプリテーマ

グループ作業を可視化できるアプリ

それぞれの役割や実際の仕事の貢献度を可視化し、グループワークなどで役割の偏りを防ぎ、円滑にグループワークをできるようにする。

背景・課題

授業・ゼミ・プロジェクト学習などでグループワークが増加

* オンラインツールの普及により作業状況が見えにくくなっている

* 無断欠席や作業不参加によるトラブルが起こりやすい

ターゲット

* 大学生・専門学校生

* グループワーク
を頻繁に行う授業・ゼミ

* 小規模プロジェクトを行う学生団体

ペルソナ

ペルソナA（一般学生）

* 年齢：20歳

* 属性：大学2年生

* 特徴：グループワークで「自分だけ頑張っている」と感じやすい

ペルソナB（リーダー役）

* 年齢：21歳

* 属性：ゼミ代表・班長

* 特徴：進捗確認に時間を取りられ、全体管理が負担

プロジェクト概要

仕事内容の難易度に応じた「貢献ポイント」の設定

タスク作成 リーダーが「資料作成：30pt」「資料整理：10pt」のように、大変さに応じてポイントを割り振り。

作業と報告 メンバーが担当タスクを完了すると、ボタン一つで報告。

見える化 蓄積されたポイントがグラフになり、「誰がどれだけ貢献したか」が全員に共有

システム構成

1. プrezentation層 (Web): ユーザーがスマホやPCで見る画面 (HTML/CSS)。

2. Application層 (App): ユーザーの操作を受けて、「ポイントを計算する」などの計算を行う脳 (Python)。

3. Data層 (DB): ユーザーの名前や獲得したポイントを保存しておく倉庫 (SQLite/PostgreSQL)。

Architectの定義: > * バックアップ: 万が一壊れても24時間前の状態には戻せるようにする。

性能: 10人同時にアクセスしても、1秒以内に画面が開くように設計。

データ設計

User (ユーザー) テーブル: 「名前」「メールアドレス」などの名簿。

Task (タスク) テーブル: 「タスク名」「割り振られたポイント」「誰の担当か」のリスト。

つながり: 1人の「ユーザー」には、複数の「タスク」が紐付きます。

これによって、「田中さんは、合計3個のタスク（計40点分）を完了した」という情報をコンピュータが正確に抽出できるようになります。