

データベース講義 最終課題発表

チームの「SOS」を見逃さない日報共有システム

チームメンバー：渡邊諒（2442102）・田中誠真（2442054）

使用技術

開発言語: Python 3.11

フレームワーク: Django 5.0

データベース: PostgreSQL 15

インフラ: Docker

フロントエンド: HTML5/CSS3

背景と課題

開発の動機:

- ・リモートワークや多忙な業務において、メンバーの不調(SOS)や貢献が見えづらい。
- ・単なる業務報告だけでは、チームの雰囲気分からない。

解決策:

- ・日報に「コンディション(調子)」という定量データを付与。
- ・「書く」だけでなく「データとして集計・可視化」するシステムを構築する。

システム概要

アプリ名: DailyReportApp

主な機能:

- CRUD機能: 日報の作成・閲覧・編集・削除
- SOS検知: コンディション「悪い」を自動抽出し、アラート表示
- ランキング: 投稿数を集計し、貢献度をランキング化
- 画像投稿: 現場の状況を視覚的に共有

特徴: レスポンシブデザイン(スマホ対応)による場所を選ばない利用。

システム構成

Web 3層構造の採用:

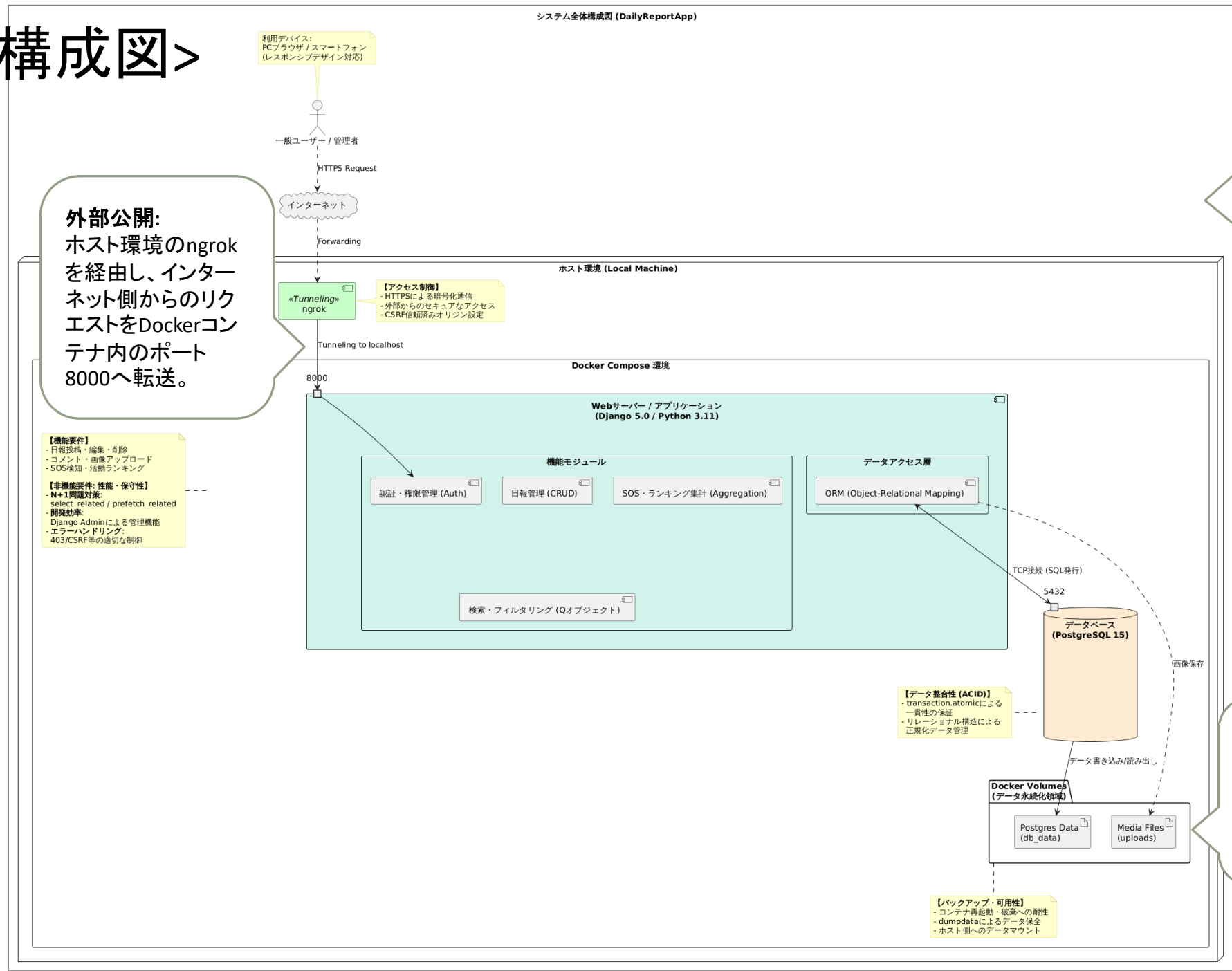
Web/App層: Django (Python) コンテナ

DB層: PostgreSQL コンテナ

データ永続化にはDocker Volumeを使用。

開発環境: Docker Composeによる完全コンテナ化(IaC)。

<システム構成図>



外部公開:
ホスト環境のngrok
を経由し、インター
ネット側からのリク
エストをDockerコン
テナ内のポート
8000へ転送。

Web3層構造:
「Webサーバー/ア
プリ(Django)」 「DB
サーバー
(PostgreSQL)」を分
離・連携。

データ永続化:
Docker Volumesを使用
し、コンテナを再起動し
てもデータが消えない
構成を構築。

データベース設計

ER図の概要:

ユーザー (Users)、日報 (Reports)、カテゴリー (Categories) を中心に設計。

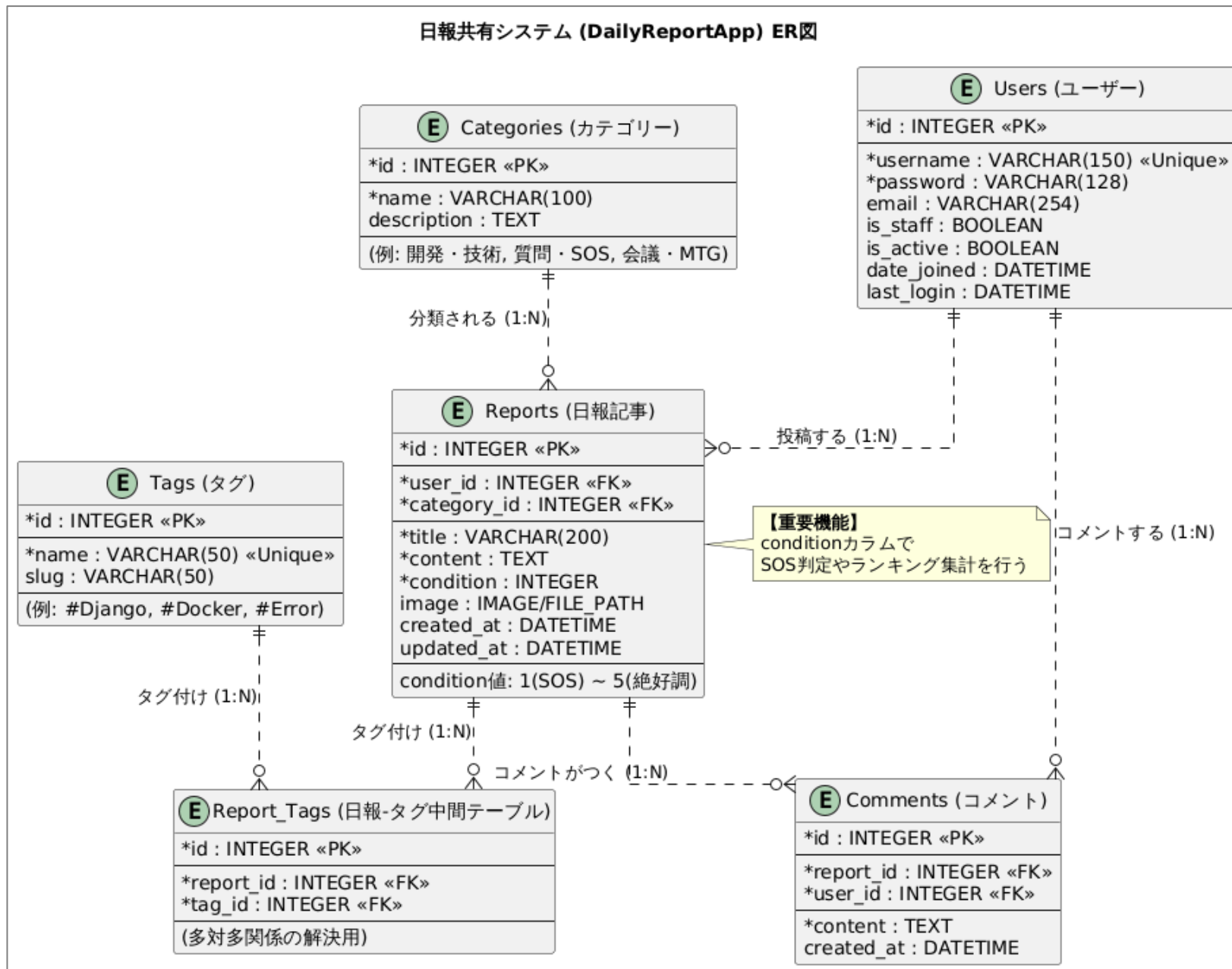
設計のポイント:

正規化: データの重複を防ぐため、カテゴリーやタグを別テーブル化。

多対多の解決:

日報とタグの関係 (Reports \leftrightarrow Tags) に中間テーブル (Report_Tags) を導入し、柔軟なタグ付けを実現。

<ER図>



DB技術の活用① 集計と可視化

課題: Pythonのループ処理で集計すると遅い。

解決策:

- ・データベースエンジン側での集計処理を採用。
- ・Djangoの `annotate` と `Count` 関数を使用し、SQLの `GROUP BY` に相当する処理を実行。

実装機能:

- ・ランキング機能:
 - ユーザーごとの投稿数をDB側でカウントし、降順ソートして取得。
- ・SOS抽出: `filter(condition=1)` でDB側でフィルタリングし、高速に抽出。

DB技術の活用② パフォーマンス最適化

直面した問題 (N+1問題):

一覧画面表示時に、記事数分のSQL(ユーザー情報取得など)が発行され、動作が重くなる可能性があった。

解決策:

- ・Eager Loading (事前読み込み) の実装。
- ・select_related(外部キー結合)と prefetch_related(多対多結合)を使用。

成果:

SQLの発行回数を劇的に削減し、レスポンス速度を向上。

DB技術の活用③ トランザクション管理

活用シーン: 日報を新規投稿する処理。

内容:

本アプリでは正規化の観点から、「日報本体(タイトル・よもやま話)」と「業務トピック(箇条書き部分)」を別テーブルに分割しています。投稿時にはこれらへの INSERT 処理が連続して発生しますが、これを単一のトランザクションとして実行しています。

DB技術の活用④ 正規化とリレーション

活用シーン: 日報データの構造化。

内容:

1つの日報に対して複数のトピックが存在するため、Reports テーブルと ReportTopics テーブルを分離し、1対多 (One-to-Many) のリレーションを構築しました。

検索機能の実装

要件: タイトルだけでなく、本文も含めたあいまい検索。

実装:

Qオブジェクト(SQLの OR 演算子)を使用。

WHERE (title LIKE '%keyword%' OR content LIKE '%keyword%') に相当するクエリを発行。

ユーザビリティ:

カテゴリー絞り込みとのAND検索も併用可能にし、過去のナレッジ検索を容易にした。

デモ / 実際の画面

