

お天気アプリ 企画書

1. 企画書のタイトル

プロジェクト名: お天気アプリ

作成日: 2024年12月15日

作成者: [吉元美咲]

2. プロジェクト概要

目的:

天気情報を簡単に取得でき、傘の必要性を判断するアプリを開発する。

背景:

日常生活において、天気予報を手軽に確認し、傘が必要かどうかすぐに知りたいというニーズがある。外出前の判断時間を短縮し、快適な生活をサポートするために本アプリを開発する。

目標:

- ユーザーが都市名を入力すると、リアルタイムの天気情報を取得し表示する。
- 傘の必要性をわかりやすく通知する機能を提供する。

3. ターゲットユーザー

ユーザー層:

- 学生
- 社会人
- 外出の多い一般ユーザー

ユーザーのニーズ:

- 都市ごとの天気情報を簡単に知りたい。
- 雨天時に傘を持って行くか判断したい。

4. 機能要件

主な機能:

- 都市名入力による天気情報の取得・表示
- 天気に応じた傘の必要性を通知する機能
- 天気アイコンと気温、天気の説明表示

非機能要件:

- UI/UX: シンプルで直感的なデザイン
- パフォーマンス: APIレスポンスを高速に処理

5. システム構成

アーキテクチャ:

- クライアントサイドアプリケーション

技術スタック:

- フロントエンド: HTML、CSS、JavaScript
- API: OpenWeatherMap API
- データ処理: Fetch API (非同期通信)

6. 開発スケジュール (計3日 + 1日発表)

フェーズ	作業内容	期間
設計・準備	機能要件定義、デザイン設計	1日目
実装	天気情報取得機能、UIデザインの実装	2日目
テスト・デバッグ	アプリ全体の動作確認と修正	3日目
発表	プレゼンテーション準備と発表	4日目

進捗目標:

- 天気情報取得機能の実装: **1日目**
- 傘の必要性通知機能の追加: **2日目**
- テスト完了: **3日目**

7. リソース (時間と自己管理)

使用時間:

- 1日あたり **3時間**、合計 **9時間**

進捗管理方法:

- GitHubでバージョン管理
- 進捗をテキストファイルや日報で記録

8. リスクと対策

予測されるリスク:

- APIエラーや通信の不具合
- 開発時間の不足

対策:

- APIエラー処理を追加して通信失敗時の動作を保証
 - スケジュールを細分化し、毎日作業進捗を確認
-

9. 品質管理とテスト

テスト戦略:

- 単体テスト: 天気情報取得機能の動作確認
- 統合テスト: UIとAPIの連携テスト
- UIテスト: デザインの表示確認

品質基準:

- エラーの発生を最小限に抑え、ユーザーが快適に操作できるUIを提供する。
-

10. 納品物

納品物の概要:

- 完成版お天気アプリ
- 企画書、操作手順書、プレゼン資料、詳細設計書、テスト表

納品予定日:

2024年12月18日

11. 運用・保守計画

運用サポート:

- APIの動作確認と定期的なメンテナンス

保守計画:

- 将来的にレスポンシブデザインへの対応
 - 天気予報の詳細機能追加（降水量、風速、週間予報）
 - ユーザーフィードバックを収集し、改善を実施
-

12. レビューと反省

自己レビュー:

- 設計から実装までの時間管理の精度を高める
- UI/UXの改善点を洗い出し、次回開発に活かす