

. Introduction

This document describes the design of the WeatherApp components as part of the WeatherApp software development project.

1.1. Document Overview

This document outlines the design of the WeatherApp components, focusing on their functionality, interactions, and overall architecture, which will be implemented using HTML, CSS, JavaScript, PHP and Oracle Database. The application will utilize the OpenWeatherMap API for fetching weather data.

1.2. References

1.2.1. Project References

Document Identifier	Document Title
----------------------------	-----------------------

[R1]	WeatherApp Project Plan
------	-------------------------

[R2]	User Requirements Document
------	----------------------------

1.2.2. Standard and Regulatory References

Document Identifier	Document Title
----------------------------	-----------------------

[STD1]	ISO 25010: Software Quality Requirements and Evaluation
--------	---

[STD2]	IEC 62304: Software Life Cycle Processes
--------	--

2. Software Architecture Overview

The WeatherApp consists of several top-level components that interact with each other. The software will run on XAMPP.



3. Software Design Description

3.1. Main Dashboard Component

3.1.1. Component Interfaces

- **Inputs:** User location, preferences.
- **Outputs:** Current weather data, forecast data.

3.1.2. Component Design Description

- The Main Dashboard displays current weather information, including temperature, humidity, wind speed, and UV index.
- It includes a login page and a registration page for user account management.
- The component fetches data from the OpenWeatherMap API.

3.1.3. Workflows and Algorithms

- User enters location → Dashboard fetches data from the OpenWeatherMap API → Data is displayed on the dashboard.

3.1.4. Software Requirements Mapping

- Maps to user requirements related to displaying weather data (REQ-101, REQ-102).
-

3.2. Forecast Component

3.2.1. Component Interfaces

- **Inputs:** Location, user preferences.
- **Outputs:** Hourly and daily forecast data.

3.2.2. Component Design Description

- Retrieves and displays hourly and daily weather forecasts from the OpenWeatherMap API.

3.2.3. Workflows and Algorithms

- Fetches forecast data from the OpenWeatherMap API based on user location → Displays data in an organized manner.

3.2.4. Software Requirements Mapping

- Related to user requirements for weather forecasting (REQ-201).
-

3.3. User Account Component

3.3.1. Component Interfaces

- **Inputs:** User registration information, login credentials.
- **Outputs:** User profile data, authentication status.

3.3.2. Component Design Description

- Features login and registration pages, user profile management, and password recovery options.

3.3.3. Workflows and Algorithms

- User registers → Data stored in the Oracle database → User logs in → Authentication is validated.

3.3.4. Software Requirements Mapping

- Relevant to user account management and authentication (REQ-301, REQ-302).
-

3.4. Alerts & Notifications Component

3.4.1. Component Interfaces

- **Inputs:** User preferences for alerts.
- **Outputs:** Severe weather alerts, notifications.

3.4.2. Component Design Description

- Manages notifications for severe weather alerts and custom user alerts sourced from the OpenWeatherMap API.

3.4.3. Workflows and Algorithms

- Monitor weather data from the OpenWeatherMap API → Trigger alerts based on predefined criteria → Notify users via push notifications.

3.4.4. Software Requirements Mapping

- Related to alert requirements (REQ-401).
-

3.5. Additional Features Component

3.5.1. Component Interfaces

- **Inputs:** User preferences for additional features.
- **Outputs:** Air quality index, sunrise/sunset times, health tips.

3.5.2. Component Design Description

- Offers interactive widgets, air quality data, weather-related health tips, and customization options, some of which may also utilize the OpenWeatherMap API.

3.5.3. Workflows and Algorithms

- Fetch additional data from external APIs including OpenWeatherMap → Display additional features on the dashboard.

3.5.4. Software Requirements Mapping

- Related to additional features requirements (REQ-501).
-

3.6. Help & Support Component

3.6.1. Component Interfaces

- **Inputs:** User queries, feedback.
- **Outputs:** FAQs, contact information for support.

3.6.2. Component Design Description

- Provides users with access to FAQs, user guides, and contact support options.

3.6.3. Workflows and Algorithms

- User submits a question → System searches FAQs → Provides relevant answers.

3.6.4. Software Requirements Mapping

- Related to support requirements (REQ-601).
-

4. SOUP Identification

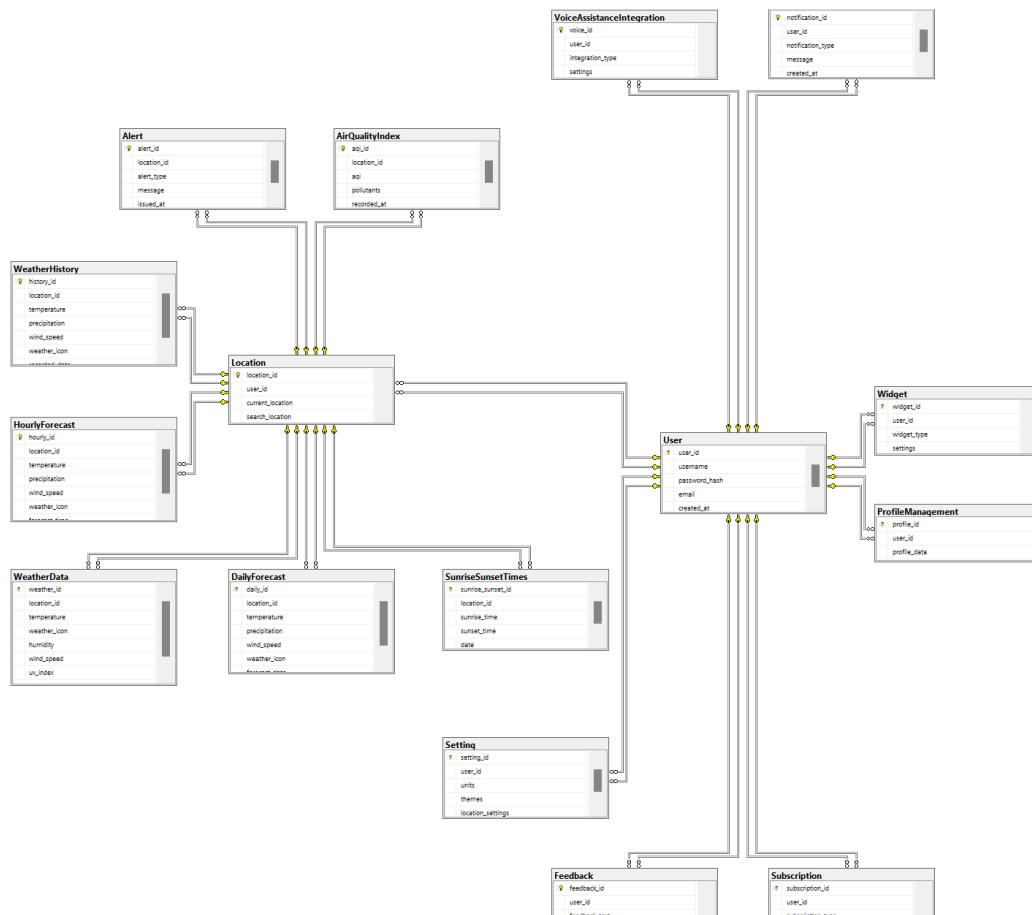
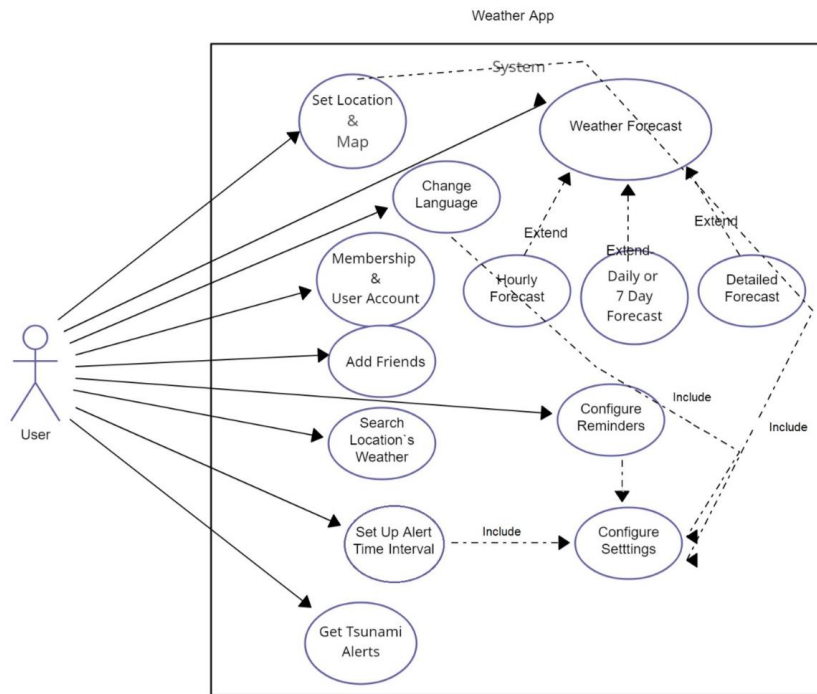
- **External Libraries:**
 - Newtonsoft.Json.dll, version 13.0.1, download URL: [URL], License: MIT, requirements traceability.
-

5. Critical Requirements

Requirement ID	Requirement Title	Component	Comment
REQ-001	Software shall have a login page	User Account Component	Login page implemented with validation.

Requirement ID	Requirement Title	Component	Comment
REQ-002	Software shall send alerts for severe weather	Alerts & Notifications Component	Notifications triggered based on API data.
REQ-003	Software shall provide hourly and daily forecasts	Forecast Component	Accurate data fetched from the OpenWeatherMap API.





ER DIAGRAM

