Weather Application Requirement Specification

Introduction

The Weather App delivers precise and current weather updates based on your location or search preferences. Enjoy features like real-time weather display, forecasts, location-specific conditions, historical data, interactive maps, alerts, and notifications. Designed with a user-friendly interface, customization options, responsiveness, cross-browser compatibility, and robust security, it ensures a seamless and reliable weather experience.

Key object

- **Current weather display-**
  - temperature, humidity, wind speed, weather description
- **Forecast display-**
  - temperatures, weather conditions, and chance of precipitation
- **Location-based weather-**
  - Allow users to view weather information for their current location or search for weather in specific cities, regions, or countries
- **Interactive maps-**

    - visualise weather patterns(කාලගුණ රටා), satellite imagery, or radar information
- **Alerts and notifications:-**
    - Display severe weather alerts or notifications for users based on their location or subscribed areas.
- **Multiple units and formats-**
    - C හා F යන දකෙනේම display වනේන ඕනෙ
- **Security-**
    - Implement secure protocols and practices to protect user data and ensure secure communication with the weather API
- **Responsive design-**
- **User customization-**
- **Performance optimization-**
- **Cross-browser compatibility-**
- **Historical weather data**-
    - Provide access to past weather data for a specified period, allowing users to view weather trends and historical records.

Functional

- Current weather display
- Forecast display
- Location-based weather
- Interactive maps
- Alerts and notifications
- Multiple units and formats
- Historical weather data

Non Functional

- **Usability**
    - Responsive design
    - User customization
    - User-Friendly Interface
- **Performance optimization**
- **Cross-browser compatibility**
- **Secure API Integration**
- **Security**

Dependencies

- The application is dependent on the availability and reliability of the "Weather API" service.
- Integration with third-party map services for interactive maps.

Technical Solutions

- Use a responsive front-end framework like Bootstrap for the user interface.
- Utilise asynchronous calls and caching mechanisms to optimise API data retrieval.
- Implement secure communication with the API using HTTPS

Priorities

1. API Integration

2. Current Weather Display
3. Forecast Display
4. Location-Based Weather
5. User Customization
6. Historical Weather Data
7. Alerts and Notifications
8. Multiple Units and Formats
9. Interactive Maps
10. Responsive Design
11. Performance Optimization
12. User-Friendly Interface
13. Cross-Browser Compatibility
14. Security