API SPECIFICATION DOCUMENT

This document contains the API specification for Weather Service Version 1.0 23/11/2019

Weather API Service

Contents

[1.0 Weather Service API 2](#_Toc25536244)

[1.1 Web Service 2](#_Toc25536245)

[2.0 Web Methods 2](#_Toc25536246)

[2.1 GetWeather 2](#_Toc25536247)

[2.1.1 Request URL 2](#_Toc25536248)

[2.1.2 Parameters 2](#_Toc25536249)

[2.2 GetCitiesByCountry 3](#_Toc25536250)

[2.2.1 Request URL 3](#_Toc25536251)

[2.2.2 Parameters 3](#_Toc25536252)

[3.0 Glossary 3](#_Toc25536253)

[3.1 Conventions 3](#_Toc25536254)

[3.2 Status Codes 4](#_Toc25536255)

# Weather Service API

The purpose of this document is to provide a high level specification for the Weather Service API (Application Programming Interface). We expect the readers of this document will already have an understanding of basic API/Web Services concepts.

The Weather Service API will be used by clients to get weather details whenever they require. Once the API is called by the client, we will return weather information based on the city and country. The response will be in JSON format by default.

There will be only one web service URL be exposed and that particular web service will have various methods for clients to decide and call based on the data they require.

The web methods described below are some of the possible scenarios for the API calls and it can be extended to create as many web methods as required to service the customer needs.

## Web Service

When the client requires data from Weather Service API, the client will establish a connection to the API by calling the Weather Service API URL link and passing the parameters. Two methods are provided to check weather based on city and country. The Weather Service API will check the database to retrieve the temperature based on Celsius and return the information to client.

Client can use any technology such as Java, PHP and .NET to call and receive data from the web service as it will use the cross-platform standard JSON protocol.

# Web Methods

## GetWeather

This web method will return temperature based on city and country.

### Request URL

|  |  |
| --- | --- |
| Method | URL |
| GET | /GetWeather/{CityName}/{CountryName} |

### Parameters

Mandatory input parameters are city and country name. If the parameters not passed or values are not valid, error message will be provided.

INPUT Parameters for the request XML

|  |  |  |
| --- | --- | --- |
| Parameters | Values |  |
| CityName | String |  |
| CountryName | String |  |

Output Parameters for the response

|  |  |  |
| --- | --- | --- |
| Parameters | Values |  |
| Celcius | String |  |

## GetCitiesByCountry

This web method will return city and its temperature based on country.

### Request URL

|  |  |
| --- | --- |
| Method | URL |
| GET | /GetCitiesByCountry/{CountryName} |

### Parameters

Mandatory input parameter is country name. If the parameter is not passed or value is not valid, error message will be provided.

INPUT Parameters for the request XML

|  |  |  |
| --- | --- | --- |
| Parameters | Values |  |
| CountryName | String |  |

Output Parameters for the response

|  |  |  |
| --- | --- | --- |
| Parameters | Values |  |
| CityName | String |  |
| Celcius | String |  |

# Glossary

## Conventions

* Client: Client application
* Status – HTTP Status code of response
* All responses are in JSON format
* All request parameters are mandatory unless explicitly marked as optional.

## Status Codes

All status codes are standard HTTP status codes. The below ones are used in this API.

2XX - Success of some kind

4XX - Error occurred in client’s part

5XX - Error occurred in server’s part