**Test plan for “Weather dev challenge”**

1. **Test environment:**

Postman version 6.1.3 (6.1.3)

1. **Test approach:**

Automated API testing via Postman:

* black box testing;
* functional testing;
* positive testing;
* negative testing.

**3. Test items**

**Features To Be Tested**

**Current weather data**

**1.Current weather data by city name:**

a) check that API responds with a list of results that match a searching word (city name);

b) check that API responds with a list of results that match a searching words (city name and country code);

c) check the possibility to receive a central district of the city/town with its own parameters (geographic coordinates/id/name) in API response;

d) check that API response with an error if the city (or country) name doesn’t exist (invalid data).

**2. Current weather data by** **city ID:**

a) check that API responds with exact result that match a searching ID;

b) check that API response with an error if the ID doesn’t exist (invalid data).

**3. Current weather data by geographic coordinates:**

a) check that API responds with exact result that match a searching coordinates;

b) check that API response with an error if the coordinates don’t exist (invalid data).

**4 Current weather data by ZIP code:**a) check that API responds with exact result that match a searching ZIP code;

b) check that API response with an error if the ZIP code doesn’t exist (invalid data).

**5. Current weather data for several cities:**

a) check that API responds with exact result that match a searching multiple IDs;

b) check that API response with an error if IDs don’t exist (invalid data).

**6 Check the response mode:**

Check that API responds with data in JSON (by default), XML, or HTML format (if mansi )

**Features not to be tested**

* 5 day / 3 hour forecast
* 16 day / daily forecast
* Historical data
* History Bulk
* Weather map layers
* UV Index BETA
* Weather stations
* Bulk downloading
* Weather alerts
* Air pollution BETA