Back-end test

After exploring the Open Weather API, the Postman tool was used to perform requests and assertions on responses using Javascript.
On the successful scenarios, standard tests were done to guarantee a quality pattern, such as:

- 200 response code for successful requests;
- Response time less than 500 ms.

For requested and edge cases, the following were considered and tested:

- **Current weather data by latitude and longitude:**
    - Response body contains defined longitude;
    - Response body contains defined latitude;
- **Edge cases:**
    - *Empty Latitude value*:
        - 400 response code;
        - Error message: Nothing to geocode;
    - *Empty Longitude value*:
        - 400 response code;
        - Error message: Nothing to geocode;
    - *Empty Latitude and Longitude value:*
        - 400 response code;
        - Error message: Nothing to geocode;
    - *Invalid Latitude value:*
        - 400 response code;
        - Error message: Wrong latitude;
    - *Invalid Longitude value:*
        - 400 response code;
        - Error message: Wrong longitude;
- **Current weather data by city name:**
    - Response body contains defined latitude;
- **Edge cases:**
    - *Non-existing city name;*
        - 400 response code;
        - Error message: City not found;
    - *Empty city name;*
        - 400 response code;
        - Error message: Nothing to geocode;
    - *City name with special characters;*
        - 200 response code;
- **Current weather with different units of measurement**
    - Response body contains temperature;

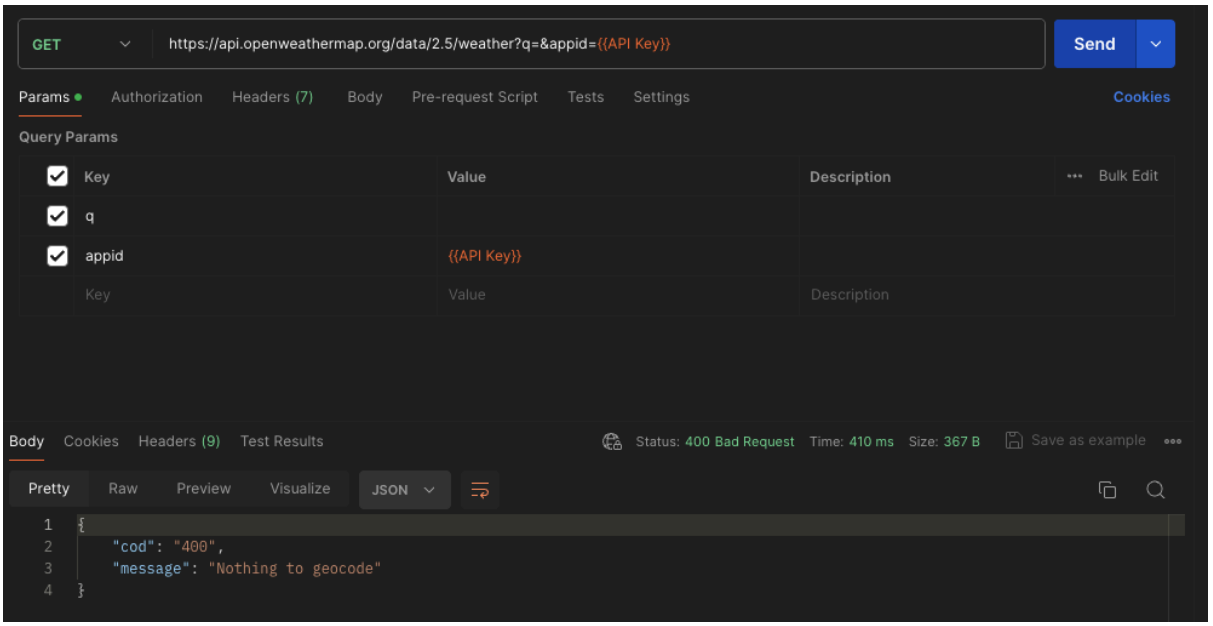
Depending on the acceptance criterias and on the business rules for the project, more tests can be done to guarantee that the response body is being covered, the minimum response time required, etc.

On the GitHub folder, within Documents > Postman, there will be a JSON file to import the created collection and also other cases that were mapped to this project.

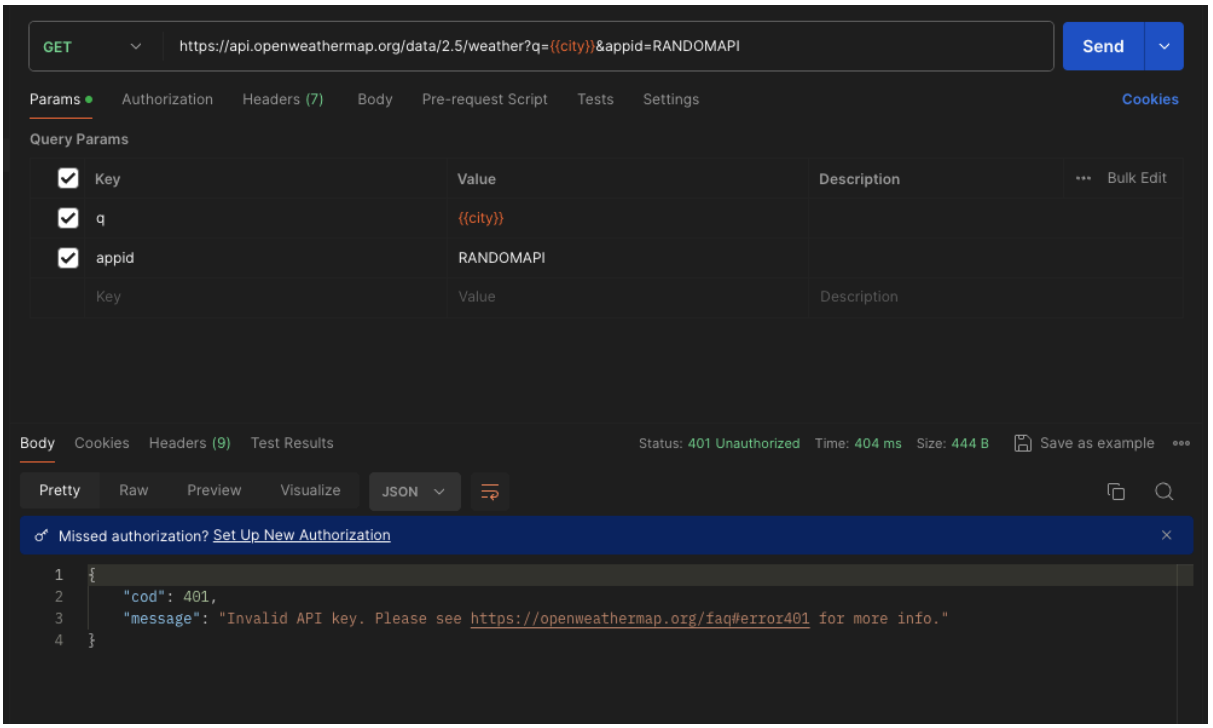
Status Code errors

Error status codes were asked to be reproduced on requests to the Open Weather API. These are the results obtained from the tests:

- **Error 400: Bad Request.**
  When there is no value for city, city id, zip code, or latitude and longitude coordinates, the application cannot return a local to provide information and the error status code is thrown.

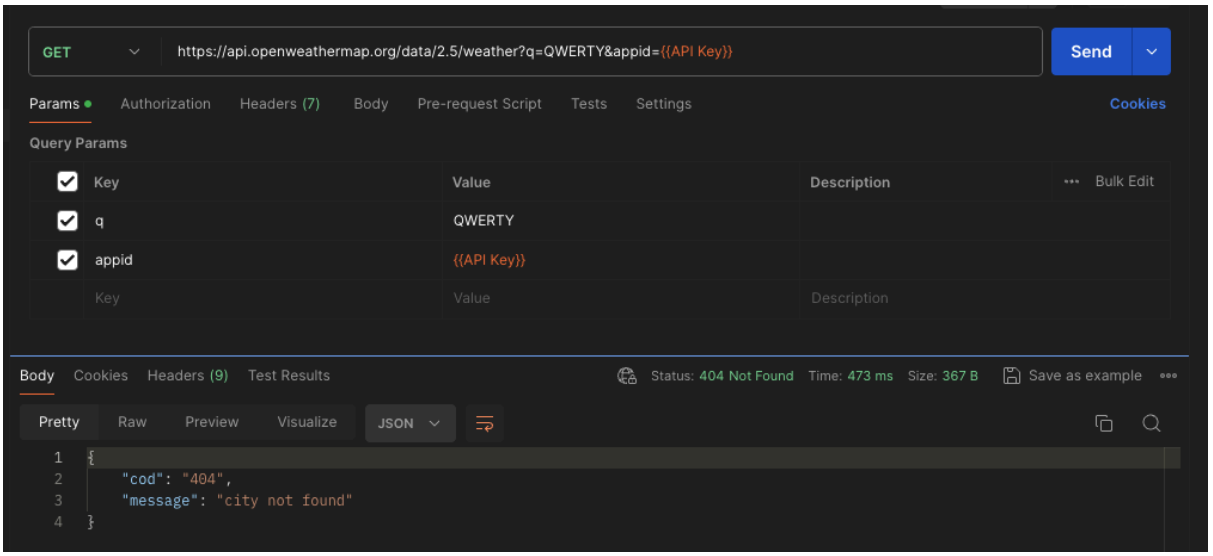


- **Error 401: Unauthorized**
  When there is no or a non-valid API Key, the error is thrown.



- **Error 404: Not found**

  When there is no result on the database for the selected input, such as a different city, the error is thrown.



- **Error 5xx: Server errors**

  There are 500 errors available to be thrown such as:

  - 500 error: Internal Server Error;
  - 502 error: Bad Gateway;
  - 503 error: Service Unavailable;
  - 504 error: Gateway Timeout.

  For the tests done, no 5xx error was able to be reproduced for this project.

- **Error 429: Too many requests**

  When there are multiple requests within a certain time, the 429 error is thrown.

I have tried to reach the limit of 60 requests per minute, which is the limit provided by Open Weather API, but it was not possible to reproduce only with multiple requests.
A possible future test to be done is parallelism, with multiple users.