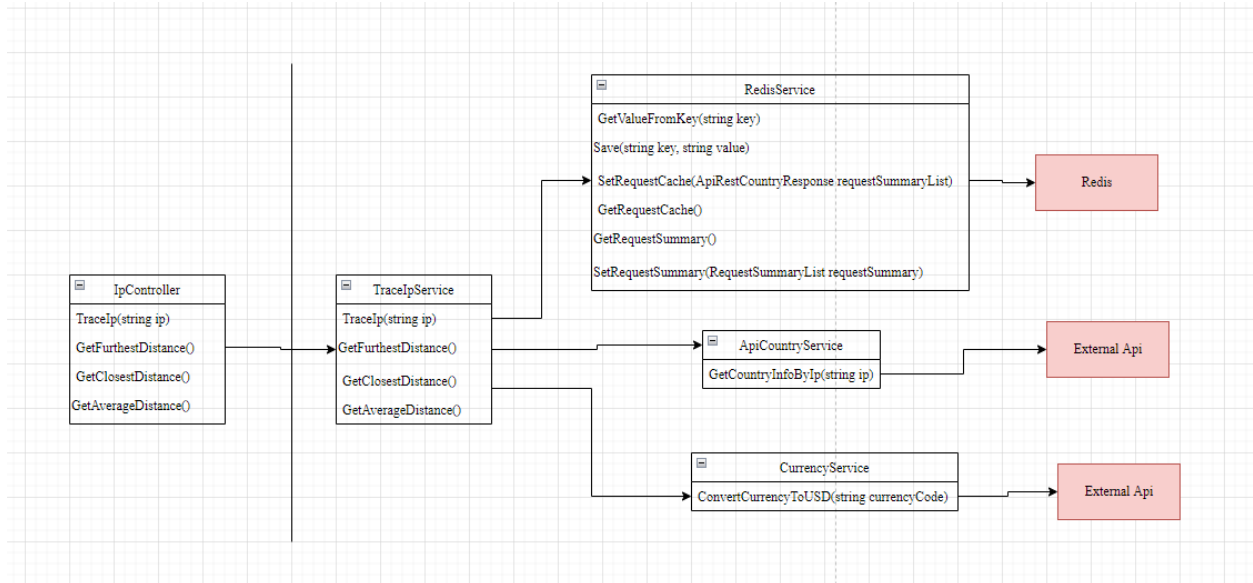


Documentation

Architecture



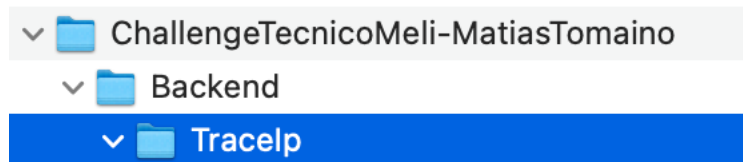
Turn-on the server

Requirements:

Docker.

Steps:

1. Download the repository (<https://github.com/MatiiTomaino/ChallengeTecnicoMeli-MatiasTomaino>)
2. Move to `ChallengeTecnicoMeli-MatiasTomaino/Backend/Tracelp`



3. Open a terminal in this path
4. Execute `sudo ./trace_ip_linux.sh`, here, the docker container is created and the server and redis instance will be running when the process finished.
 - a. **Note: if command doesn't work, run `chmod +x trace_ip_linux.sh`.**
5. Check that server and redis instance are running.

Containers
[Give Feedback](#)

A container packages up code and its dependencies so the application runs quickly and reliably from one computing environment to another. [Learn more](#)

Showing 1 of 1

| | NAME | STARTED | STATUS | |
|--------------------------|--|--------------|---------------|--|
| <input type="checkbox"/> | <div>traceip</div> <div>2 containers</div> <div> redis:6.2-alpine 25444f8eeef08 (traceip_cache_1) </div> <div> traceip-backend 611fd2df4c3a (traceip_traceip_1) </div> | | running (2/2) | Open |
| | | 1 minute ago | running | Open Logs Stop Refresh |
| | | 1 minute ago | running | Open Logs Stop Refresh |

Test API

APIs Available

- TraceIp
 - Required param: Ip
- GetFurthestDistance
- GetClosestDistance
- GetAverageDistance

1. Postman:

TraceIp : <http://localhost:5000/ip/traceip?ip=83.44.196.93>

GET TraceIp
+
...
No Environment

TraceIp / TraceIp
Save
...

GET
http://localhost:5000/ip/traceip?ip=83.44.196.93
Send

Params
Authorization
Headers (7)
Body
Pre-request Script
Tests
Settings
Cookies

Query Params

| KEY | VALUE | DESCRIPTION | ... | Bulk Edit |
|--|--------------|-------------|-----|-----------|
| <input checked="" type="checkbox"/> ip | 83.44.196.93 | | | |
| Key | Value | Description | | |

Body
Cookies
Headers (4)
Test Results
Status: 200 OK Time: 3.87 s Size: 715 B Save Response

Pretty
Raw
Preview
Visualize
JSON

```

1 {
2   "name": "Spain",
3   "isoCode": "ES",
4   "languages": [
5     {
6       "code": "es",
7       "name": "Spanish",
8       "native": "Español"
9     },
10    {
11      "code": "eu",
12      "name": "Basque",
13      "native": "Euskara"
14    },
15    {
16      "code": "ca",
17      "name": "Catalan",
18      "native": "Català"
19    },
20    {
21      "code": "gl",
22      "name": "Galician",

```

GetFurthestDistance: http://localhost:5000/lp/GetFurthestDistance

The screenshot shows the Tracelp API client interface. The top bar indicates the current request is a GET to `http://localhost:5000/lp/GetFurthestDistance`. The 'Params' tab is active, showing a table with columns 'KEY', 'VALUE', and 'DESCRIPTION'. Below this, the 'Body' tab is selected, displaying the response in JSON format. The status bar at the bottom indicates a successful response with status 200 OK, a time of 63 ms, and a size of 188 B.

| KEY | VALUE | DESCRIPTION |
|-----|-------|-------------|
| Key | Value | Description |

```
1 {  
2   "countryName": "Spain",  
3   "value": 10807.47  
4 }
```

Status: 200 OK Time: 63 ms Size: 188 B

GetClosestDistance: http://localhost:5000/lp/GetClosestDistance

The screenshot shows the Tracelp API client interface. The top bar indicates the current request is a GET to `http://localhost:5000/lp/GetClosestDistance`. The 'Params' tab is active, showing a table with columns 'KEY', 'VALUE', and 'DESCRIPTION'. Below this, the 'Body' tab is selected, displaying the response in JSON format. The status bar at the bottom indicates a successful response with status 200 OK, a time of 17 ms, and a size of 188 B.

| KEY | VALUE | DESCRIPTION |
|-----|-------|-------------|
| Key | Value | Description |

```
1 {  
2   "countryName": "Spain",  
3   "value": 10807.47  
4 }
```

Status: 200 OK Time: 17 ms Size: 188 B

GetAverageDistance: http://localhost:5000/lp/GetAverageDistance

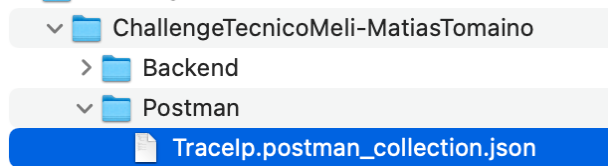
The screenshot shows the Tracelp API client interface. The top bar indicates the current request is a GET to `http://localhost:5000/lp/GetAverageDistance`. The 'Params' tab is active, showing a table with columns 'KEY', 'VALUE', and 'DESCRIPTION'. Below this, the 'Body' tab is selected, displaying the response in Text format. The status bar at the bottom indicates a successful response with status 200 OK, a time of 26 ms, and a size of 150 B.

| KEY | VALUE | DESCRIPTION |
|-----|-------|-------------|
| Key | Value | Description |

```
1 10807.47
```

Status: 200 OK Time: 26 ms Size: 150 B

There is available a collection example in repository to import in another computer:



2. Curl

In a terminal execute:

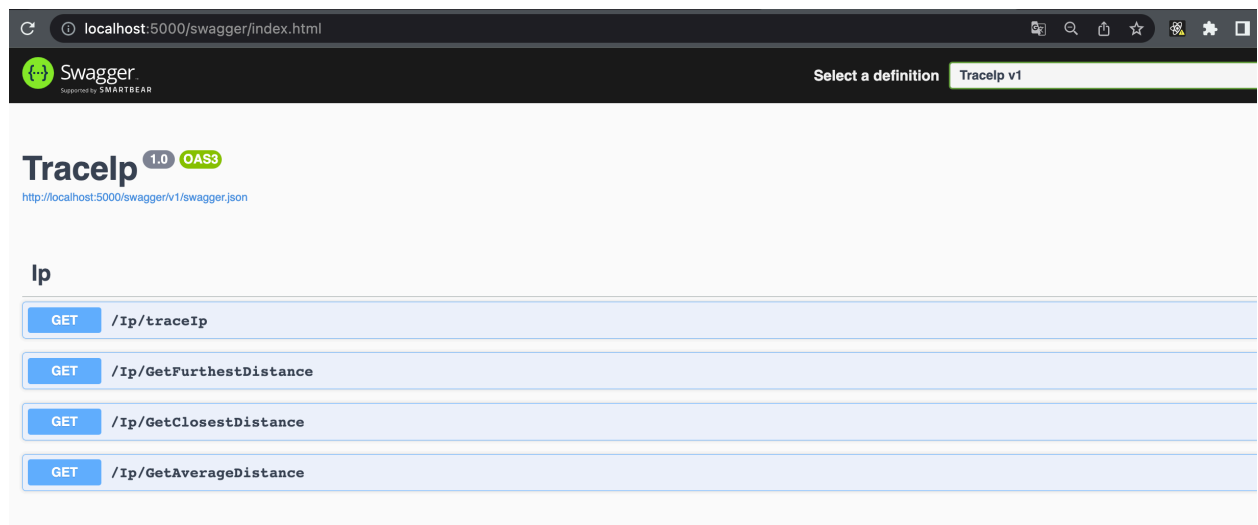
```
curl --location --request GET 'http://localhost:5000/Ip/traceip?ip=83.44.196.93'
```

```
MacBook-Pro-de-Matias-3:ChallengeTecnicoMeli-MatiasTomaino matii$ curl --location --request GET 'http://localhost:5000/Ip/traceip?ip=83.44.196.93'  
{  
  "name": "Spain",  
  "isoCode": "ES",  
  "languages": [  
    {  
      "code": "es",  
      "name": "Spanish",  
      "native": "Espa\u00f1ol",  
      "code": "eu",  
      "name": "Basque",  
      "native": "Euskara",  
      "code": "ca",  
      "name": "Catalan",  
      "native": "Catal\u00e0",  
      "code": "gl",  
      "name": "Galician",  
      "native": "Galego",  
      "code": "oc",  
      "name": "Occitan",  
      "native": "Occitan",  
      "actualTime": ["05:13 (UTC)", "06:13 (UTC+01:00)"],  
      "estimatedDistance": 10807.47,  
      "currency": {  
        "code": "EUR",  
        "name": "Euro",  
        "symbol": "\u20ac",  
        "quotation": null,  
        "currencyAndQuotation": "EUR (1 EUR = 1.05 USD)",  
        "dateTimeRequest": "2022-06-13T05:13:47.9793577+00:00",  
        "ip": "83.44.196.93"  
      }  
    }  
  ]  
}
```

3. Web

Swagger was enabled only for the challenge. Generally Swagger is available only for development environment, but I made an exception so that it can be tested more easily

Access : <http://localhost:5000/swagger/index.html>

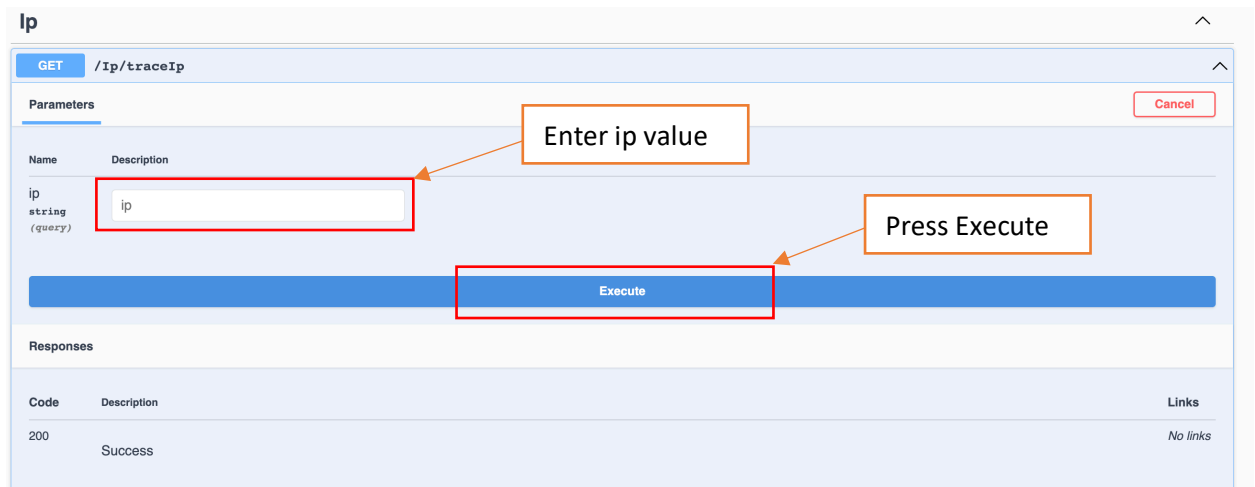


Test:

1. Enter to <http://localhost:5000/swagger/index.html>
2. Choice any Endpoint and click try it out



3. If is necessary, enter the params and press Execute



4. View Result

Curl

```
curl -X 'GET' \
'http://localhost:5000/Ip/traceIp?ip=83.44.196.93' \
-H 'accept: */*'
```

Request URL

```
http://localhost:5000/Ip/traceIp?ip=83.44.196.93
```

Server response

| Code | Details |
|------|---|
| 200 | <div><div>Response body</div><div><pre>{ "name": "Spain", "isoCode": "ES", "languages": [{ "code": "es", "name": "Spanish", "native": "Español" }, { "code": "eu", "name": "Basque", "native": "Euskara" }, { "code": "ca", "name": "Catalan", "native": "Català" }, { "code": "gl", "name": "Galician", "native": "Galego" }] }</pre></div><div> Download</div></div> |