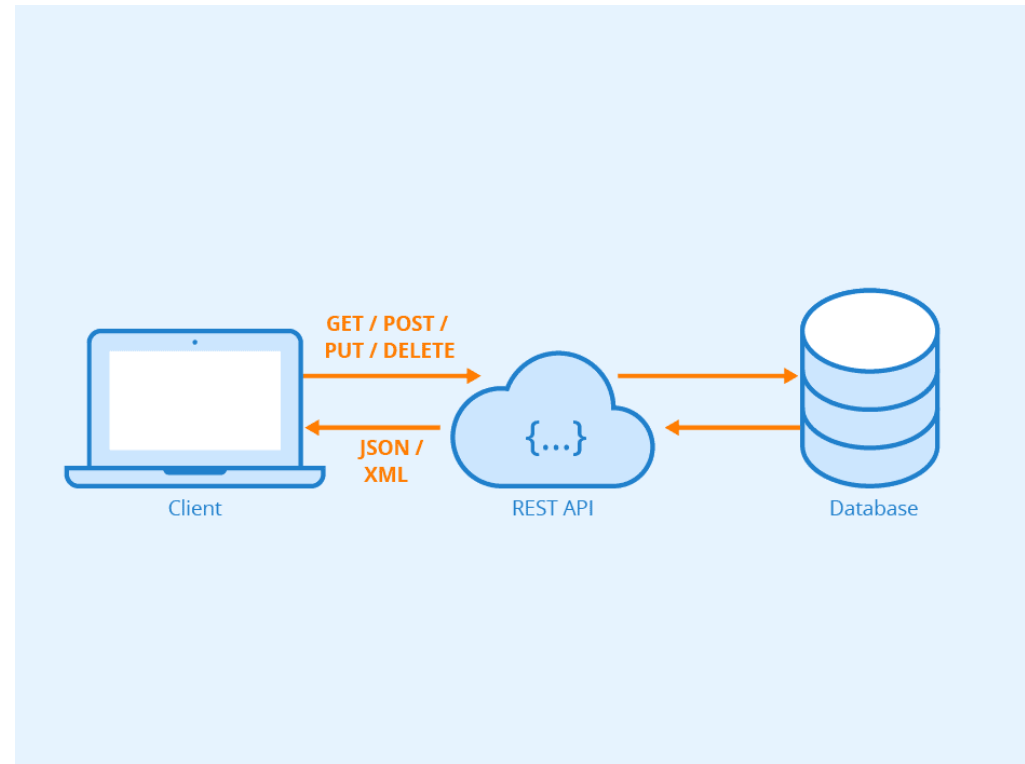
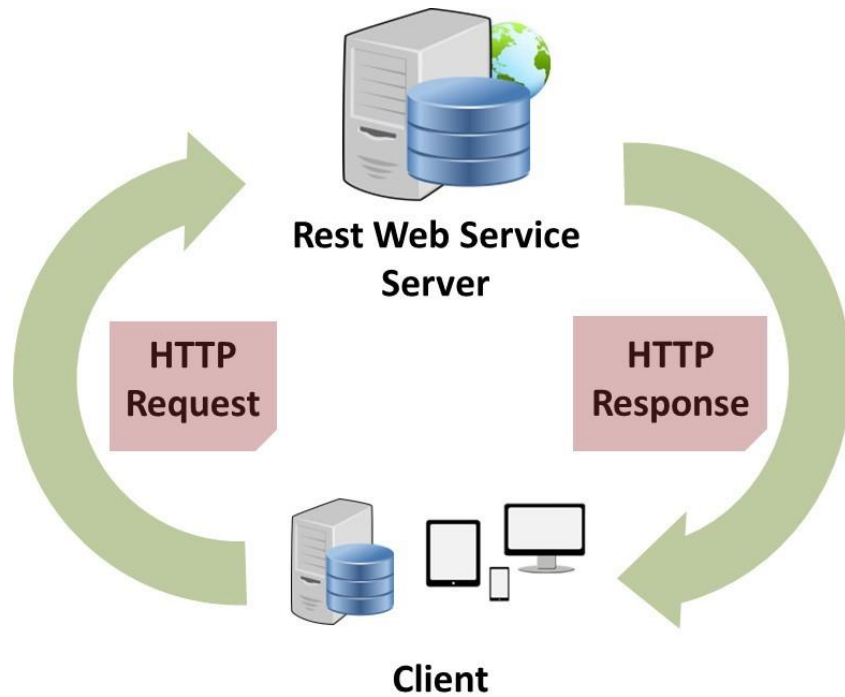


# What is an API?

[Quick introduction video.](#)

# What is a REST API?

- To simplify: a REST API is an API that links services together through the web (usually through HTTP)

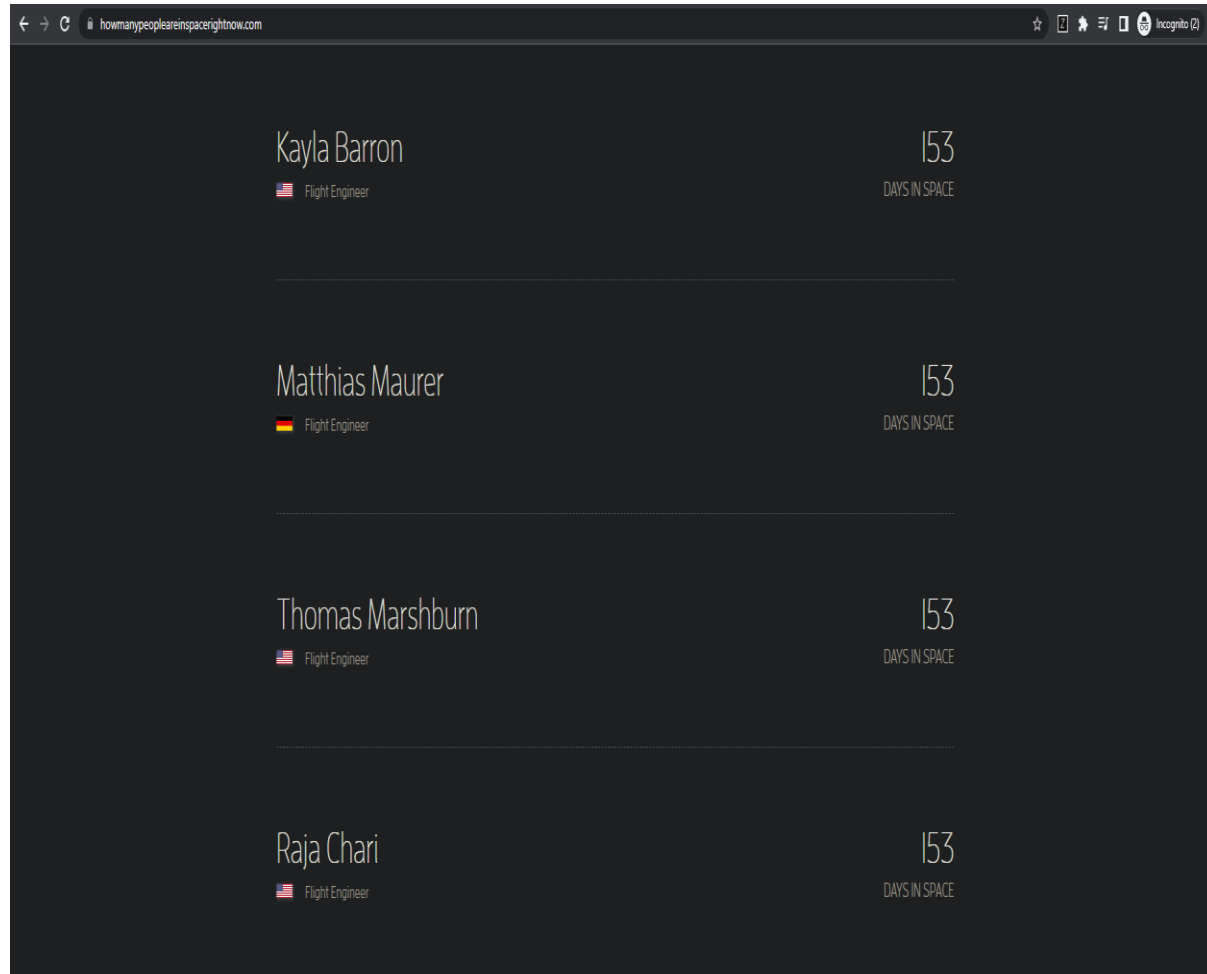


## *Nomenclature:*

- Representational State Transfer (REST)
  - An architectural style for designing the linkage between frontend and backend **web** applications.
- Application Programming Interface (API)
  - An API is a software medium that allows two different applications to talk to each other.

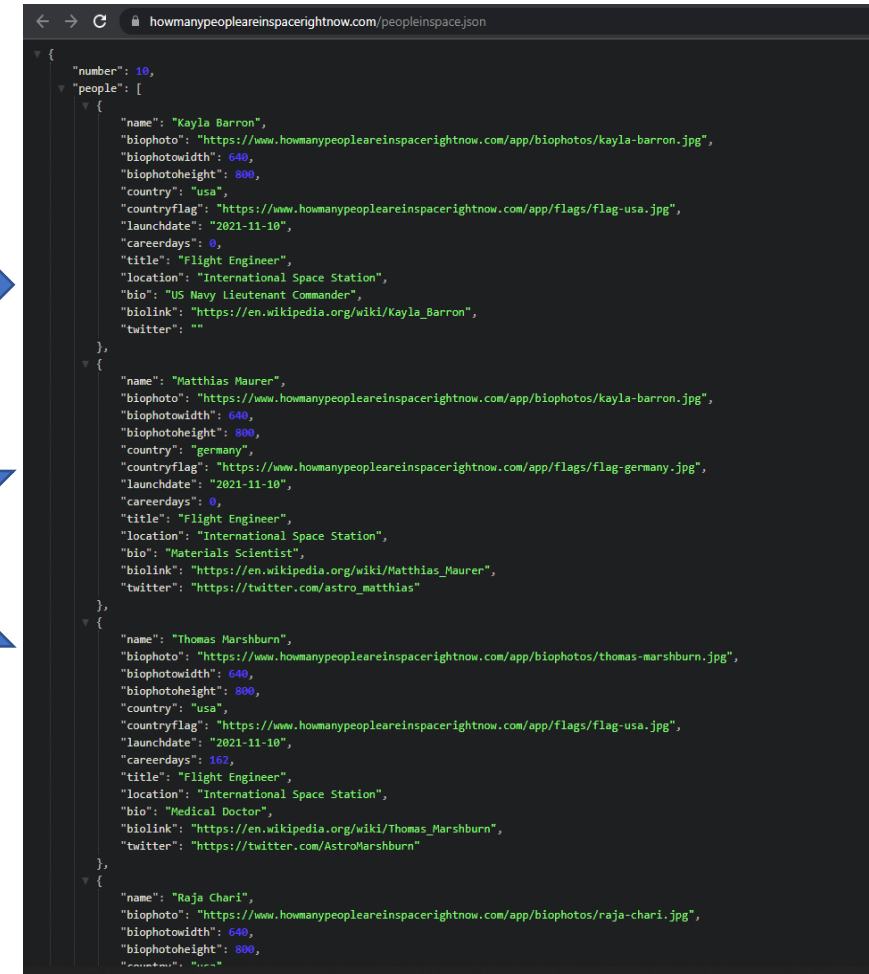
# Example:

GET - <https://www.howmanypeopleareinspacerightnow.com/>



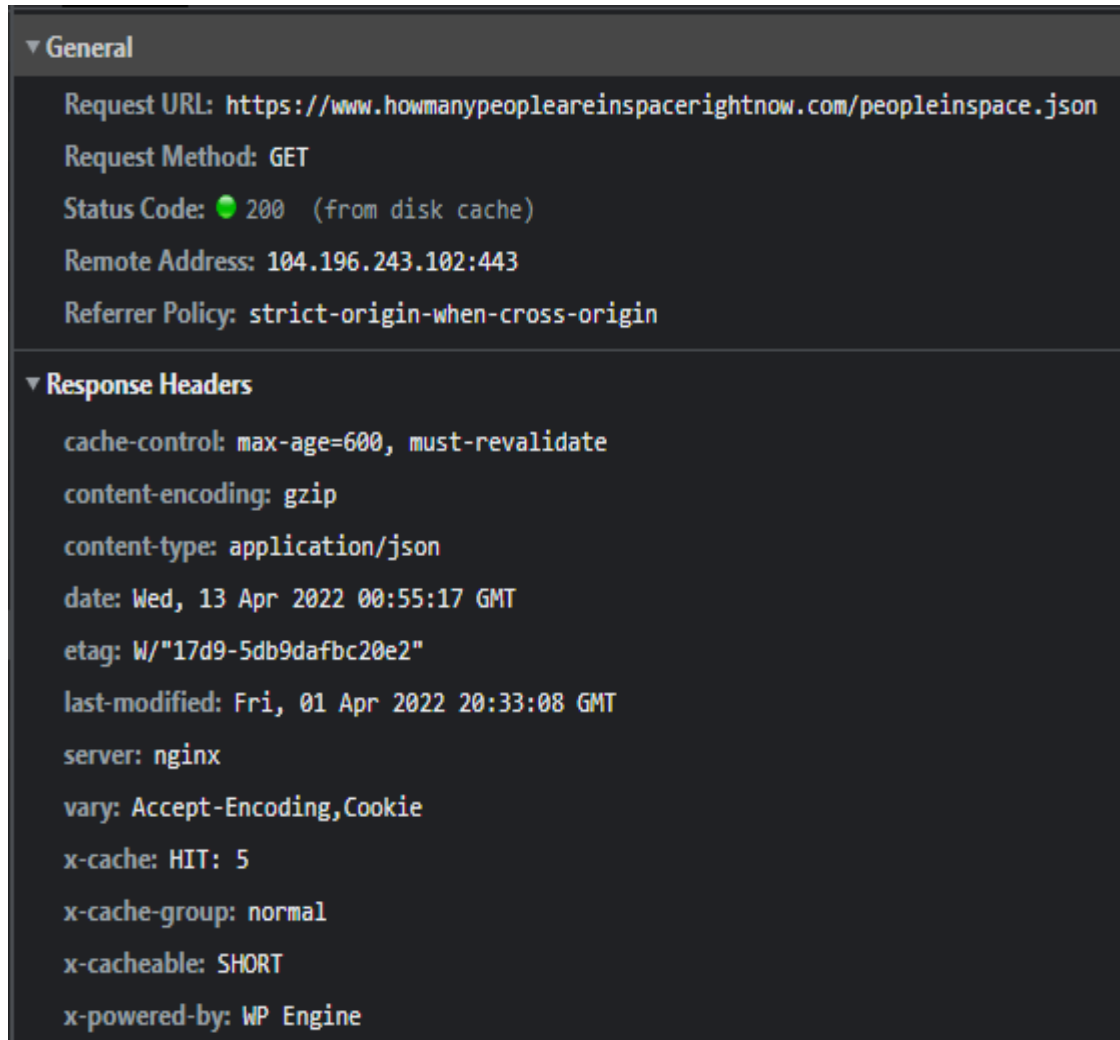
1) Client-side script calls **(request)** backend

2) Backend returns data **(response)** to client



GET - <https://www.howmanypeopleareinspacerightnow.com/peopleinspace.json>

# A dive deeper into the API request itself:



**General**

Request URL: `https://www.howmanypeopleareinspacerightnow.com/peopleinspace.json`

Request Method: GET

Status Code: ● 200 (from disk cache)

Remote Address: 104.196.243.102:443

Referrer Policy: strict-origin-when-cross-origin

**Response Headers**

cache-control: max-age=600, must-revalidate

content-encoding: gzip

content-type: application/json

date: Wed, 13 Apr 2022 00:55:17 GMT

etag: W/"17d9-5db9dafbc20e2"

last-modified: Fri, 01 Apr 2022 20:33:08 GMT

server: nginx

vary: Accept-Encoding, Cookie

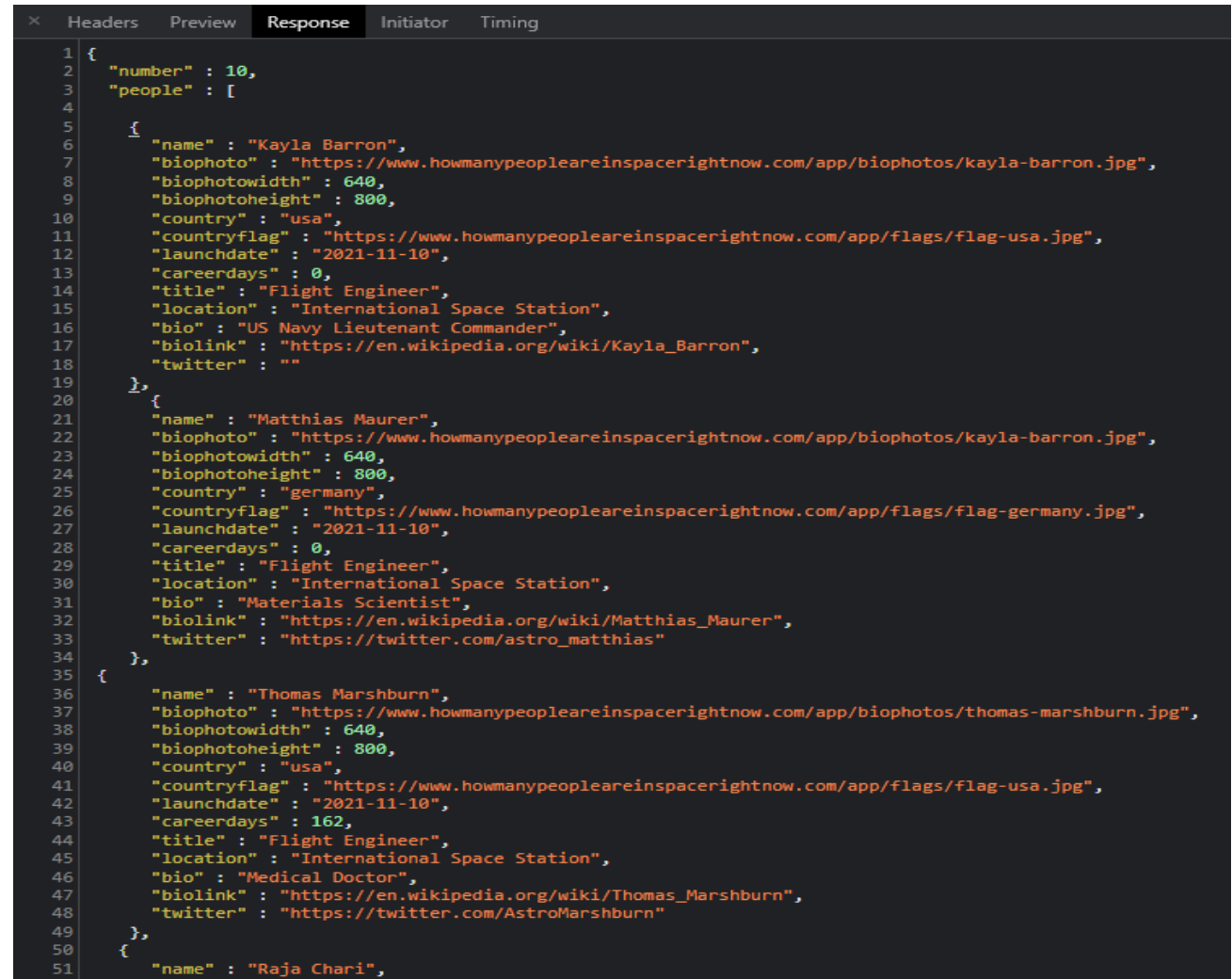
x-cache: HIT: 5

x-cache-group: normal

x-cacheable: SHORT

x-powered-by: WP Engine

Request ^



**Response**

```
1 {
2   "number" : 10,
3   "people" : [
4
5     {
6       "name" : "Kayla Barron",
7       "biophoto" : "https://www.howmanypeopleareinspacerightnow.com/app/biophotos/kayla-barron.jpg",
8       "biophotowidth" : 640,
9       "biophotoheight" : 800,
10      "country" : "usa",
11      "countryflag" : "https://www.howmanypeopleareinspacerightnow.com/app/flags/flag-usa.jpg",
12      "launchdate" : "2021-11-10",
13      "careerdays" : 0,
14      "title" : "Flight Engineer",
15      "location" : "International Space Station",
16      "bio" : "US Navy Lieutenant Commander",
17      "biolink" : "https://en.wikipedia.org/wiki/Kayla_Barron",
18      "twitter" : ""
19    },
20    {
21      "name" : "Matthias Maurer",
22      "biophoto" : "https://www.howmanypeopleareinspacerightnow.com/app/biophotos/kayla-barron.jpg",
23      "biophotowidth" : 640,
24      "biophotoheight" : 800,
25      "country" : "germany",
26      "countryflag" : "https://www.howmanypeopleareinspacerightnow.com/app/flags/flag-germany.jpg",
27      "launchdate" : "2021-11-10",
28      "careerdays" : 0,
29      "title" : "Flight Engineer",
30      "location" : "International Space Station",
31      "bio" : "Materials Scientist",
32      "biolink" : "https://en.wikipedia.org/wiki/Matthias_Maurer",
33      "twitter" : "https://twitter.com/astro_matthias"
34    },
35    {
36      "name" : "Thomas Marshburn",
37      "biophoto" : "https://www.howmanypeopleareinspacerightnow.com/app/biophotos/thomas-marshburn.jpg",
38      "biophotowidth" : 640,
39      "biophotoheight" : 800,
40      "country" : "usa",
41      "countryflag" : "https://www.howmanypeopleareinspacerightnow.com/app/flags/flag-usa.jpg",
42      "launchdate" : "2021-11-10",
43      "careerdays" : 162,
44      "title" : "Flight Engineer",
45      "location" : "International Space Station",
46      "bio" : "Medical Doctor",
47      "biolink" : "https://en.wikipedia.org/wiki/Thomas_Marshburn",
48      "twitter" : "https://twitter.com/AstroMarshburn"
49    },
50    {
51      "name" : "Raja Chari",
```

Response (JSON) ^

# How would be build one? - Backend



```
const express = require('express')
const cors = require("cors")
const app = express()
const port = 3000

app.use(cors()) // allow cross-origin request

app.get('/', (req, res) => {
  res.send('Data from backend')
})

app.listen(port, () => {
  console.log(`Backend listening on port ${port}`)
})
```

# How would we build one? - Frontend

```
<html>
<body>
  <div class="container">
    <div id="content"></div>
    <button onclick="onPress()">Get backend data</button>
  </div>
  <script>
    function onPress() {
      setText("Loading...")
      fetch("http://localhost:3000")
        .then((response) => response.text())
        .then(setText).catch(setText)
    }
    function setText(txt) {
      document.querySelector("#content").innerHTML = txt
    }
  </script>
</body>
</html>
```



# Demo:

Response when backend is offline

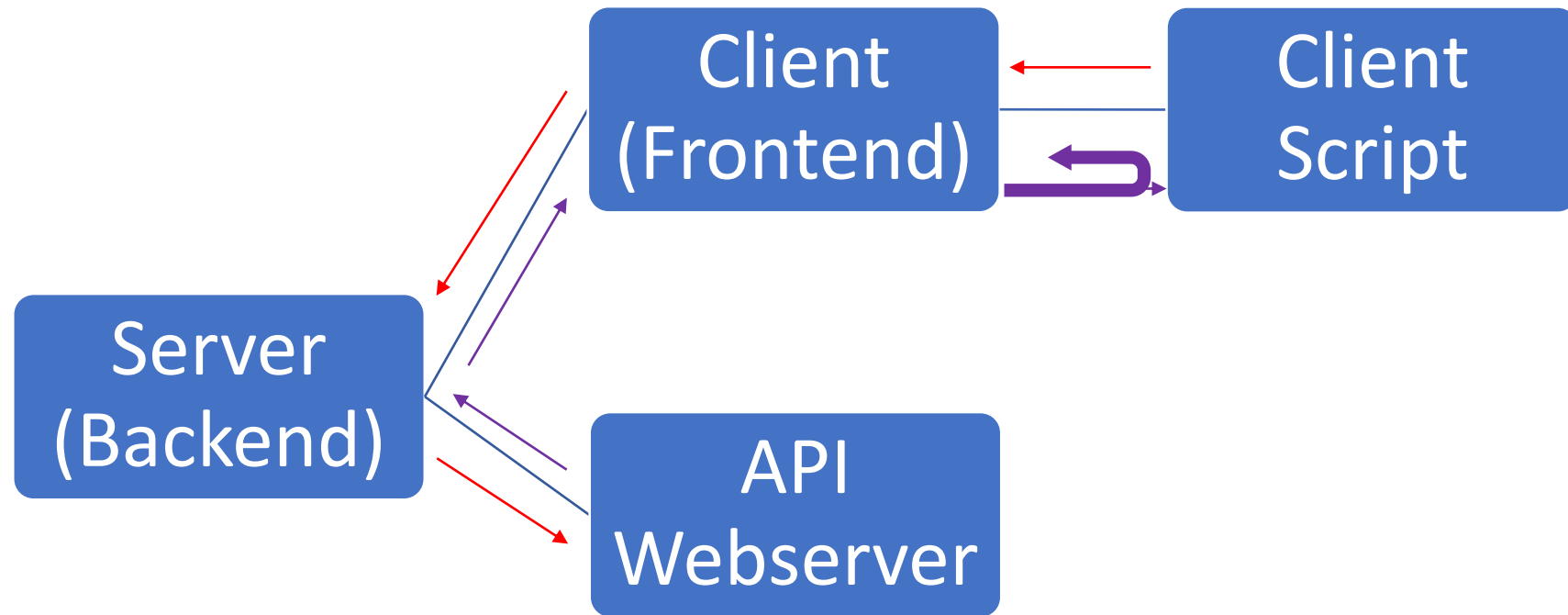
Get backend data

Response when backend is online

Get backend data



# What we just did:



■ - Response  
■ - Request

# What's next?

- In these slides, we only scratched the surface for one of the most integral pieces of the internet. The REST API workshop (this Friday) will go more in-depth in terms of technicalities and conventions for backend REST practices. We will not only build scalable REST APIs, but also dissect prebuilt ones.
- Thank you for listening