REST (in Firebase) + HTTP

Some of examples and definitions are from Firebase docs - https://firebase.google.com/docs, or great web docs https://developer.mozilla.org/ or Wikipedia.

Hello

I am Mateusz Choma

I am a scientific mind, passionate of technology, an engineer "squared" - a graduate of two universities in Lublin :)
As well, I am a JS developer, entrepreneur and owner of small software house - Amazing Design.

1.
What is HTTP?

What is HTTP?

Hypertext Transfer Protocol

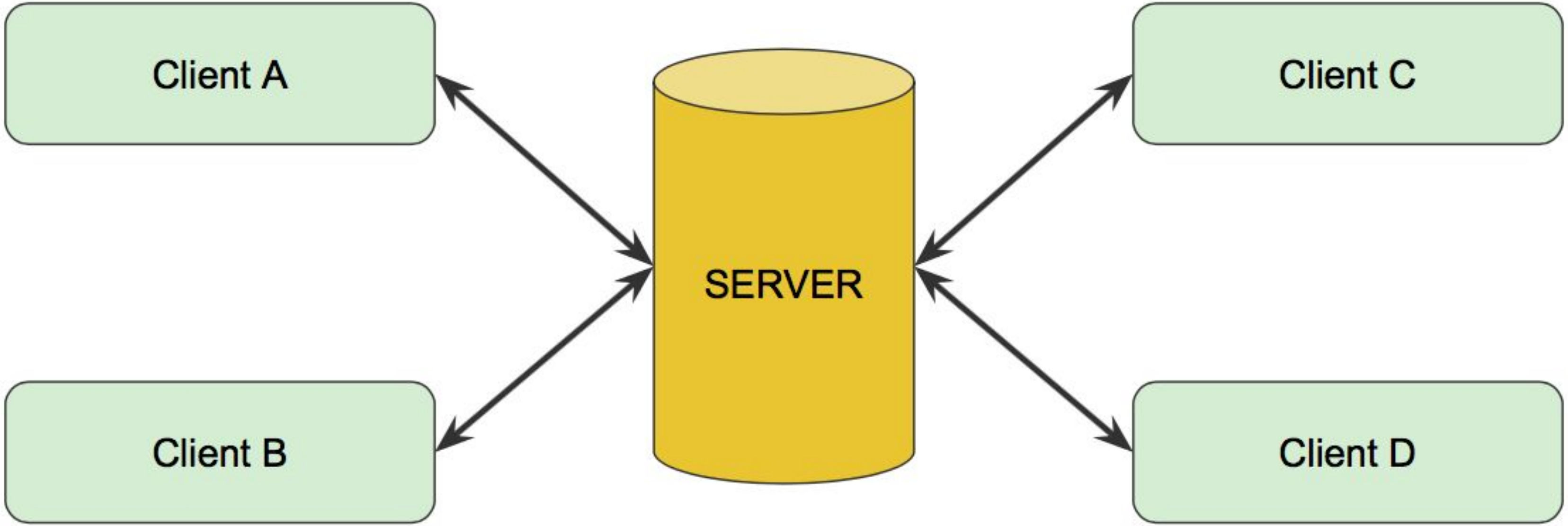
HTTP is an protocol that is a high-level interface that we use to make request over the Internet.
Its definition presumes an underlying and reliable transport layer protocol - Transmission Control Protocol (TCP) and physical infrastructure..

HTTP is:

- text
- stateless
- client-server architecture

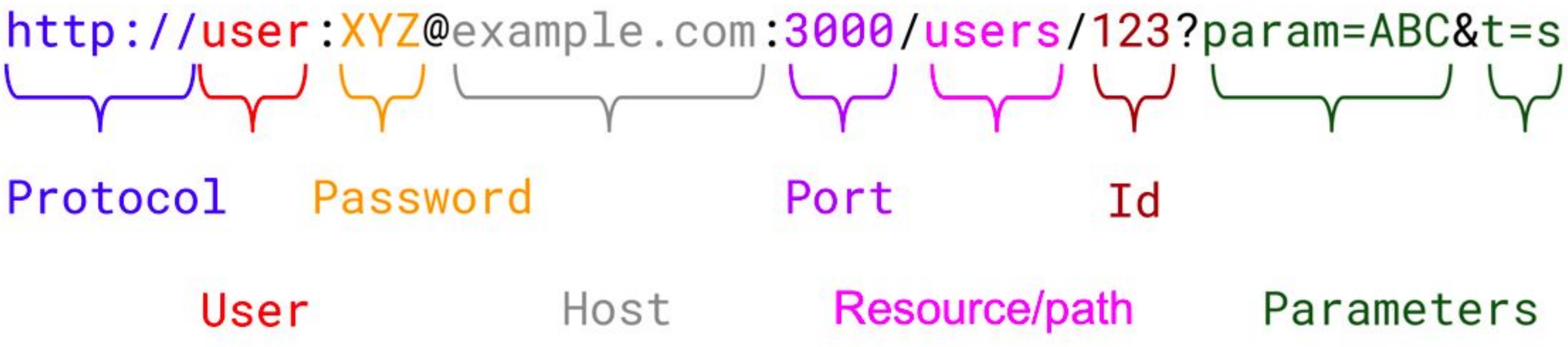
What is HTTP?

Client-server architecture



What is HTTP?

URL



http://user:XYZ@example.com:3000/users/123?param=ABC&t=s

Protocol Password Port Id

User Host Resource/path Parameters

What is HTTP?
HTTP request

The client and server communicate by **plain-text** messages.

The **client sends requests** to the server and the **server sends response.**

Request contains:

1. A request line (e.g. GET users/123 HTTP/1.1)
2. Request header fields
3. An empty line
4. An optional message body

What is HTTP?

HTTP status codes

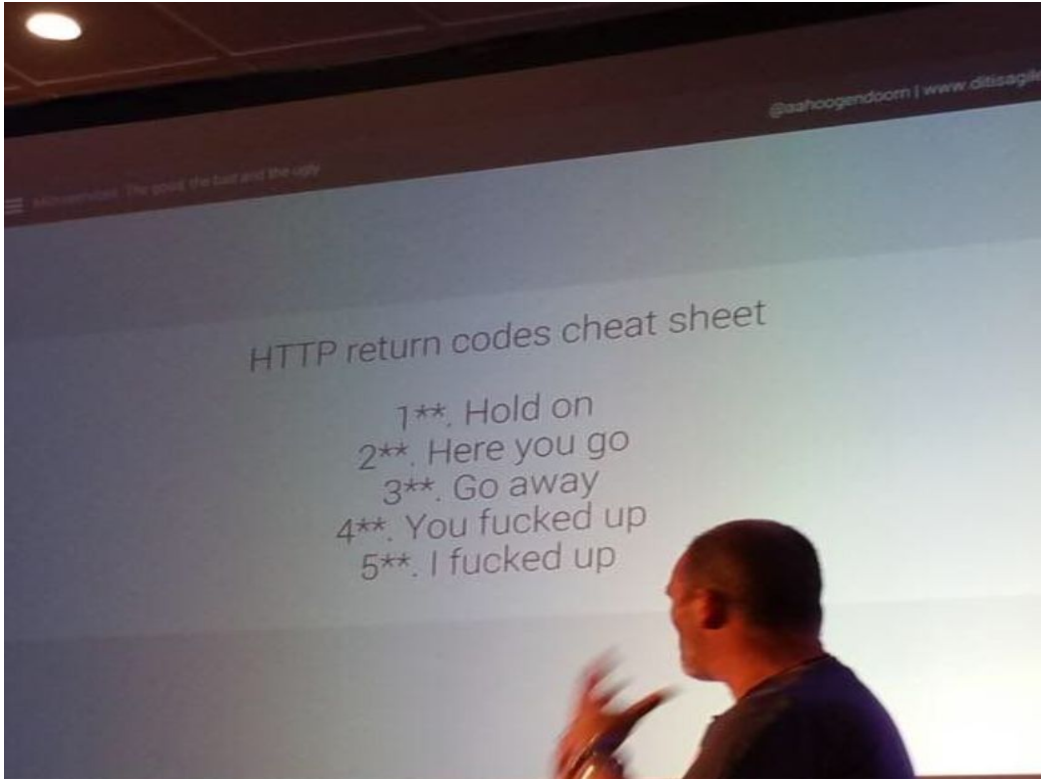
Status codes are issued by a server in response to a browser's request made to the server.

1xx - informational
2xx - success
3xx - redirects
4xx - client error
5xx - server error

https://en.wikipedia.org/wiki/List_of_HTTP_status_codes

What is HTTP?

HTTP status codes cheat sheet



Microservices. The good, the bad and the ugly

@aahoogendoorn | www.ditisagil

HTTP return codes cheat sheet

1**. Hold on
2**. Here you go
3**. Go away
4**. You fucked up
5**. I fucked up

What is HTTP?

Every enterprise-grade software has cats inside ;)

https://http.cat/

https://httpstatusdogs.com/

What is HTTP?
Headers

HTTP header fields are components of the header section of HTTP request and response.

They define the operating parameters of an HTTP transaction.

Headers can be checked here -> https://en.wikipedia.org/wiki/List_of_HTTP_header_fields

What is HTTP?
Message body

HTTP Message Body is the data bytes transmitted in an HTTP transaction message immediately following the headers.

What is HTTP?

Sample HTTP response

```
HTTP/1.1 200 OK
Date: Sun, 10 Oct 2010 23:26:07 GMT
Server: Apache/2.2.8 (Ubuntu) mod_ssl/2.2.8 OpenSSL/0.9.8g
Last-Modified: Sun, 26 Sep 2010 22:04:35 GMT
Accept-Ranges: bytes
Content-Length: 13
Connection: close
Content-Type: text/html

Hello world!
```

2.
REST API

REST API

API - application programming interface

In general terms, it is a set of clearly defined methods of communication between various software components. A good API makes it easier to develop a computer program by providing all the building blocks, which are then put together by the programmer.

REST API
REST - Representational State Transfer

A way of providing interoperability between computer systems on the Internet. REST-compliant Web services allow requesting systems to access and manipulate textual representations of Web resources using a uniform and predefined set of stateless operations.

REST has:

- Unified interface
- Stateless communication
- Client-server architecture
- HATEOAS

“

Task 0

Find an install Chrome plugin called JSON Viewer

REST API

Checking weather for Lublin through REST API

HOST: http://api.openweathermap.org

RESOURCE: forecast

VERSION: 2.5

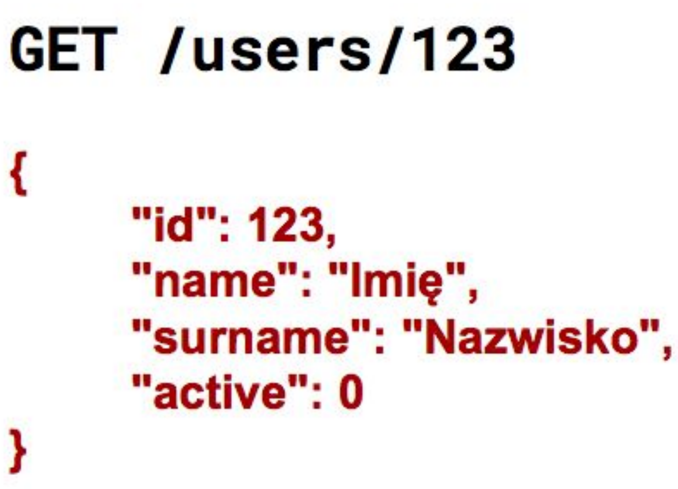
CITY ID: 765876

API Key: 0b3d75e5a49f2a267f054a0a60bed6f3

http://api.openweathermap.org/data/2.5/forecast?id=765876&APPID=0b3d75e5a49f2a267f054a0a60bed6f3

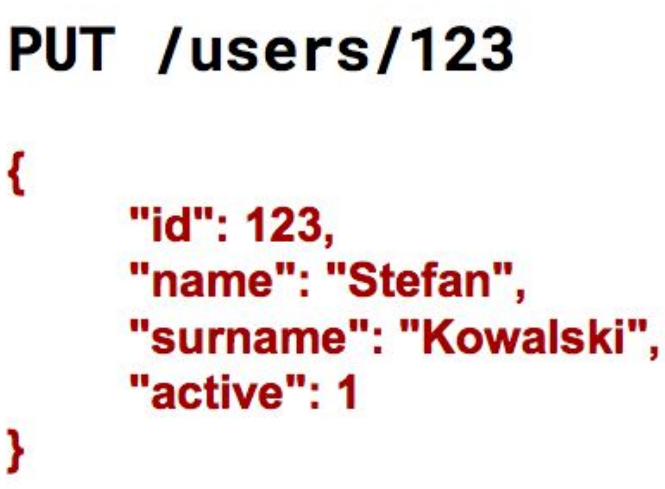
REST API

Resources representation



```
GET /users/123

{
        "id": 123,
        "name": "Imię",
        "surname": "Nazwisko",
        "active": 0
}
```

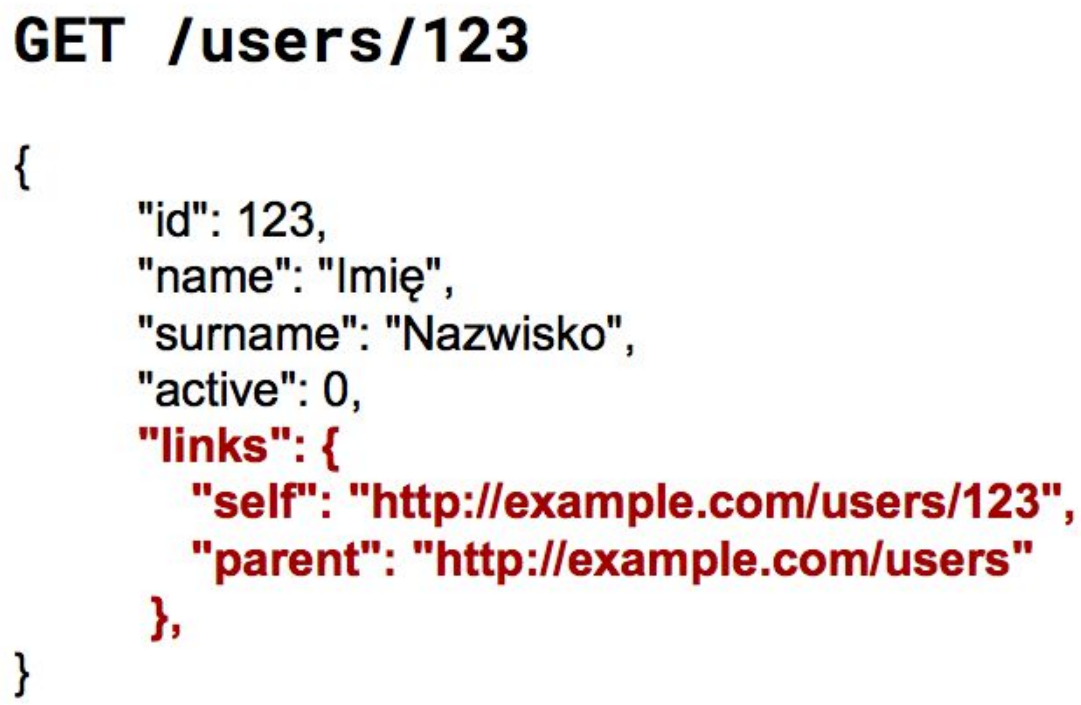


```
PUT /users/123

{
        "id": 123,
        "name": "Stefan",
        "surname": "Kowalski",
        "active": 1
}
```

REST API

Hypermedia As The Engine Of Application State



```
GET /users/123

{
        "id": 123,
        "name": "Imię",
        "surname": "Nazwisko",
        "active": 0,
        "links": {
           "self": "http://example.com/users/123",
           "parent": "http://example.com/users"
         },
}
```

REST API

JSON - JavaScript Object Notation

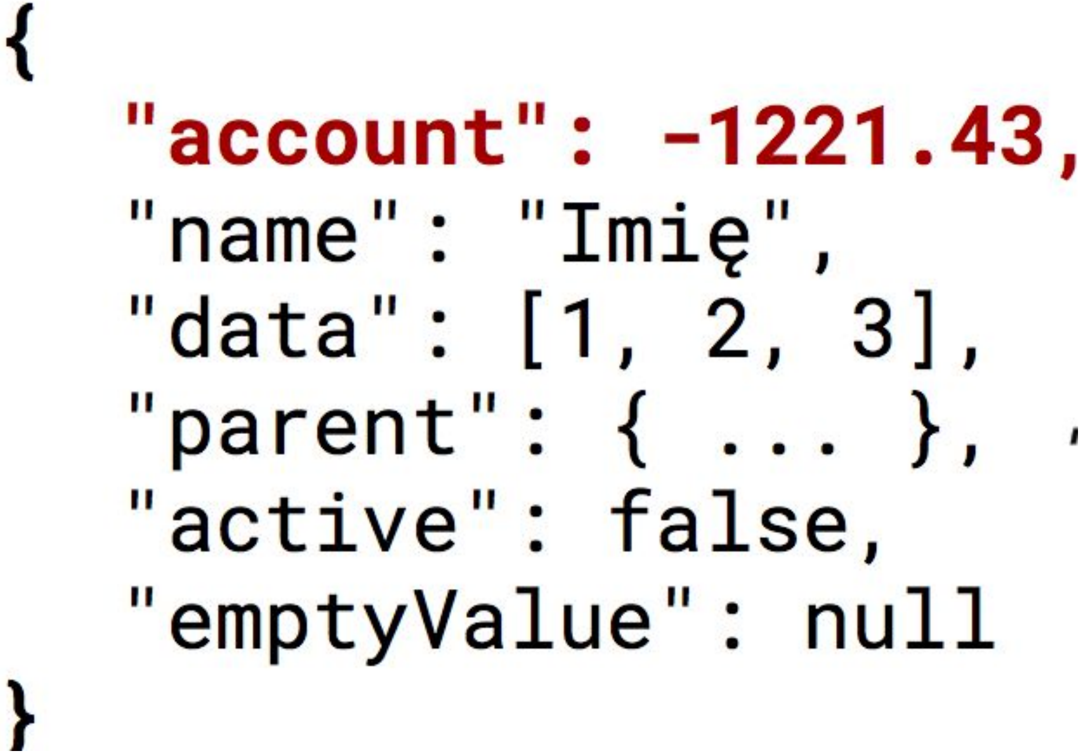
JSON is an standard file format that uses human-readable text to transmit data objects consisting of attribute–value pairs and array data types. It is a very common data format used for asynchronous browser–server communication.

JSON data is JS object literal, with keys surrounded by double quotes and it allows data types (**FUNCTION IS NOT THERE**):

- number
- string
- boolean
- array
- object
- null

REST API

JSON - JavaScript Object Notation



```
{
    "account": -1221.43,
    "name": "Imię",
    "data": [1, 2, 3],
    "parent": { ... },
    "active": false,
    "emptyValue": null
}
```

“ ***Task 1***

Find an install Chrome plugin called Advanced REST Client

Task 2

Using http://openweathermap.org/ check actual temperature in Gdańsk in Celsius degrees

0b3d75e5a49f2a267f054a0a60bed6f3

REST API

Example API

http://omdbapi.com/

https://randomuser.me/

Task 3

Using GET check what is under the endpoint https://ad-snadbox.firebaseio.com/get-endpoint.json

REST API

Basic methods

GET - getting resources
POST - creating new resources
PUT - updating whole resource
PATCH - updating part of resource
DELETE - delete resource

REST API

Basic methods on Firebase REST API example

https://firebase.google.com/docs/reference/rest/database/

REST API

Basic methods

1. **GET** /users/YOUR_NAME.json

2. **POST** /users/YOUR_NAME.json

```
{ "name": "Mateusz", "surname": "Choma", "phone": "+48" }
```

3. **PUT** /users/YOUR_NAME/**_ID_**.json

```
{ "name": "Mateusz", "surname": "Choma", "active": 0,
"phone": "+48" }
```

4. **PATCH** /users/YOUR_NAME/**_ID_**.json

```
{ "active": 1 }
```

5. **DELETE** /users/YOUR_NAME/**_ID_**.json

“

Task 4

Try to use methods from previous slide on this API https://ad-snadbox.firebaseio.com/

“

Task 5

Make all this requests using FETCH in REACT.

** triggered by button clicks*
*** editable body in textarea*

Task 6 CONTEST!

This API sends mails - https://emaillabs-sender.herokuapp.com/

Use POST to send mail to mateusz.choma@infoshareacademy.com

Fastest WIN!

Task 6 CONTEST!

“

API require POST body to be a JSON object that has properties:

emails - array of to emails [strings]
message - message [string]
subject - subject [string]
from - email of sender [string]
fromName - name of sender [string]