Aaahh!!! Real Endpoints!

Making sense of that gibberish with Loren Velasquez
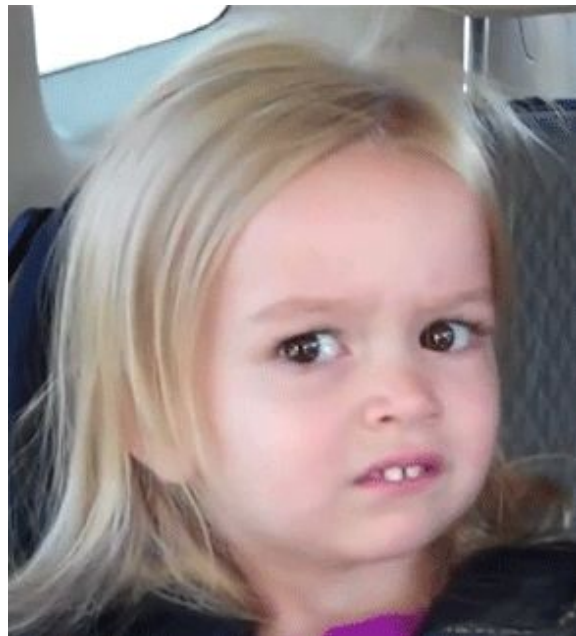
Have you heard of words like....

- Endpoints
- API
- RESTful
- CRUD
- GET
- POST
- PATCH
- DELETE
- verbs

And the immediate thought was...

> Que?

> lolwut

> Plz speak human/decipher yourself

> do you mean IPA?

> Exit, pursued by a bear

Odds are ever in your favor!

Today, is the dawning of a new era.

Gibberish will be translated so you can talk tech back and know what the heck is going on!

But first, DATABASE… think of a closet

- ADD CLOTHES
- SELECT CLOTHES
- BEDAZZLE CLOTHES
- REMOVE CLOTHES

> Better excel/spreadsheets

Translation

CLOSET - TABLE (other rooms hold different stuff, house = database)

ADD CLOTHES - INSERT data

SELECT CLOTHES - SELECT data

BEDAZZLE CLOTHES - UPDATE data

REMOVE CLOTHES - DELETE data

API: Application Programming Interface

API

- way we get information from online services
- Docs tells you how to make requests from API
- SOAP vs REST

REST - type of API

- Representational State Transfer - think web services (using HTTP protocol)
- standards/rules for how to build a web API so everyone is on the same page

https://stackoverflow.com/questions/41189842/what-is-difference-between-rest-and-api

What's the skinny on RESTful APIs?

- set of URLs that return data responses
- makes sharing data easier
- use JSON (python dictionaries) for a data format
- follows guidelines
- docs tell you what you want to know about how to interact with API

Terms

Resource: object/representation of something, has associated data with it

- Schools, Customers, Employees = resources (plural nouns)
- Add/Delete/Update/Select = actions/operations performed on resources

URL: sentence where resources are nouns and HTTP methods are verbs

- should only contain resources (nouns) not actions or verbs
- Uniform Resource Locator, path resource can be located and actions can be performed

Source: https://hackernoon.com/restful-api-designing-guidelines-the-best-practices-60e1d954e7c9

HTTP methods aka verbs: GET & POST

Indicates type of action to be performed on a resource

GET - requests data from resource

- **GET /foods** - list of all the foods
- **GET /foods/1** - food with id=1 info

POST - requests server to create a resource in database

- **POST /foods** - create a food
  - Data provided

Source: https://hackernoon.com/restful-api-designing-guidelines-the-best-practices-60e1d954e7c9

HTTP methods aka verbs: PATCH & DELETE

Indicates type of action to be performed on a resource

PATCH - requests data from a resource to be updated in database

- **PATCH /foods/1** - Update data for food id=1
  - Data provided

DELETE - requests that the resource be removed in database

- **DELETE /foods/1** - delete food id=1

Source: https://hackernoon.com/restful-api-designing-guidelines-the-best-practices-60e1d954e7c9

HTTP Status Codes

This page is created from HTTP status code information found at ietf.org and Wikipedia. Click on the **category heading** or the **status code** link to read more.

1xx Informational

100 Continue   101 Switching Protocols   102 Processing (WebDAV)

2xx Success

★ 200 OK   ★ 201 Created   202 Accepted
203 Non-Authoritative Information   ★ 204 No Content   205 Reset Content
206 Partial Content   207 Multi-Status (WebDAV)   208 Already Reported (WebDAV)
226 IM Used

3xx Redirection

300 Multiple Choices   301 Moved Permanently   302 Found
303 See Other   ★ 304 Not Modified   305 Use Proxy
306 (Unused)   307 Temporary Redirect   308 Permanent Redirect (experimental)

4xx Client Error

★ 400 Bad Request   ★ 401 Unauthorized   402 Payment Required
★ 403 Forbidden   ★ 404 Not Found   405 Method Not Allowed
406 Not Acceptable   407 Proxy Authentication Required   408 Request Timeout
★ 409 Conflict   410 Gone   411 Length Required
412 Precondition Failed   413 Request Entity Too Large   414 Request-URI Too Long
415 Unsupported Media Type   416 Requested Range Not Satisfiable 417 Expectation Failed
418 I'm a teapot (RFC 2324)   420 Enhance Your Calm (Twitter)   422 Unprocessable Entity (WebDAV)
423 Locked (WebDAV)   424 Failed Dependency (WebDAV)   425 Reserved for WebDAV
426 Upgrade Required   428 Precondition Required   429 Too Many Requests
431 Request Header Fields Too Large 444 No Response (Nginx)   449 Retry With (Microsoft)
450 Blocked by Windows Parental Controls (Microsoft)   451 Unavailable For Legal Reasons   499 Client Closed Request (Nginx)

5xx Server Error

★ 500 Internal Server Error   501 Not Implemented   502 Bad Gateway
503 Service Unavailable   504 Gateway Timeout   505 HTTP Version Not Supported
506 Variant Also Negotiates (Experimental) 507 Insufficient Storage (WebDAV) 508 Loop Detected (WebDAV)
509 Bandwidth Limit Exceeded (Apache)   510 Not Extended   511 Network Authentication Required
598 Network read timeout error   599 Network connect timeout error

Apply the fresh knowledge...

https://developer.github.com/v3/#http-verbs

https://developer.github.com/v3/users/

https://developer.github.com/v3/projects/

Enough Chit chat

Show me the code! DEMO! DEMO! DEMO!