

# Docs first

from a client's perspective

if a feature is not documented, then it doesn't exist,  
if a feature is documented incorrectly, then it's broken

# Overview

- why good documentation matters more for apis and microservices
- how I've benefited from a docs first approach
- documentation technology (openapi/swagger)

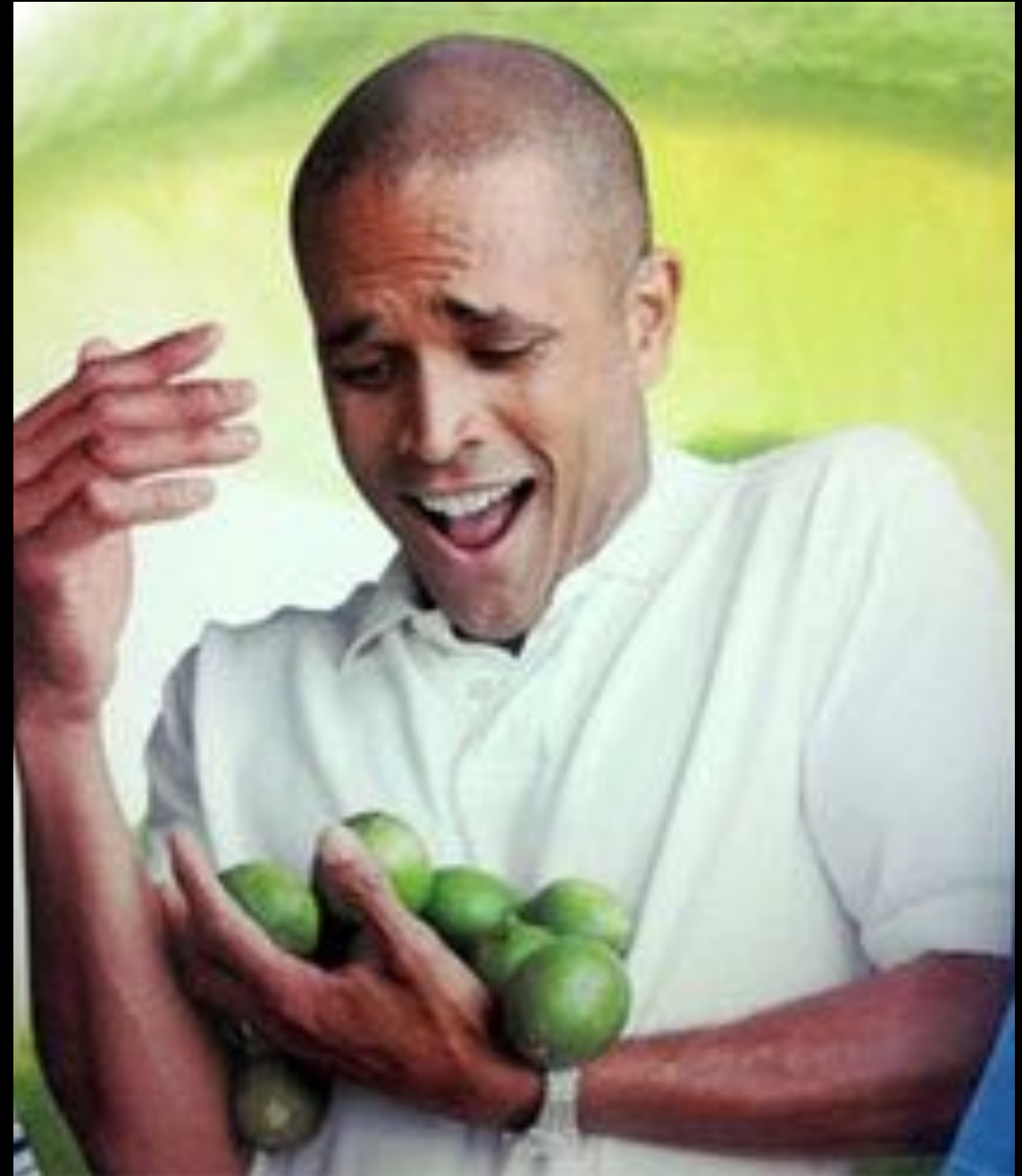


# Disclaimer: docs are boring

actual footage of a developer working on  
documentation

# All these clients

- user client
- admin client
- other micro services in ecosystem
- 3rd party clients



# When clients read the docs

- point out missing functionality
- complain about the number of requests
- complain about the structure of data
- show ambiguities
- “clarify” the “original” requirements

**timeline example**

# Documentation benefits

- Feedback before committing to a solution
- Thinking the whole project through
- Replace early specifications





# Machine readable documentation benefits

- OpenAPI (Swagger) ecosystem
- Spec validation
- UI rendering
- Automated testing
- Code generation





# Ui rendering

```
paths:
  /agencies:
    get:
      summary: List Agencies
      description: |
        Get a list of agencies on t
      tags:
        - 'Agencies'
      parameters:
        - $ref: '#/parameters/inclu
        - $ref: '../parameters/comm
        - $ref: '../parameters/comm
        - $ref: '../parameters/head
      responses:
        '200':
          description: Success
          content:
            application/json:
```

List Users

GET /users

List all users

QUERY PARAMETERS

page	integer	The page number of results to return
limit	integer	The max number of results to return per page, defaults to 24

HEADER PARAMETERS

X-Account-Id	string <b>Required</b>	Example: "1E35FD16-306B-11E7-A171-58A558409ABE" Account Id
--------------	------------------------	---------------------------------------------------------------

Responses

✓ 200
✗ default invalid input or action not permitted

Response samples

200 default

application/json

```
{
  "data": [
    {}
  ],
  "meta": {
    "pagination": {}
  }
}
```

# How it went

- larger OpenApi docs got hard to debug
- spec doesn't cover everything we need right now
- IDE support is lacking
- used machine readable docs with getting a machine to read it

# Next steps

- cli watch filesystem and validate docs
- json refs, allOf, oneOf, anyOf
- api scaffolding from spec (swagger -> laravel)
- help build OpenApi ecosystem