# CONTRACT FIRST WITH OPENAPI

March 19, 2021

# ABOUT ME

Lead Software Engineer in Exadel

10+ years in Java-development

**ARTEM VEGERA**

# AGENDA

- Difference between Contract First and Code First
- How to do Contract First with OpenAPI Specification
- Pitfalls of Contract First approach
- OpenAPI tooling landscape

# THE INITIAL PROBLEM

## TYPICAL API DEVELOPMENT

- Feature request
- Ping-pong development between BE and UI devs
- QA can write API-tests after BE-implementation
- Feature in production

# THE INITIAL PROBLEM

TYPICAL API DEVELOPMENT

**WHAT WE WANT?**

- Be able to specify our exposed API
- Parallelize API-implementation and API-consumers work like

  BE, UI, Mobile, QA Automation, Integration Apps, etc.
- (optional) Have separate API design phase

# WITHOUT DOCUMENTATION

## PROS

- Just write what's on your mind
- Easy to make changes and break things*
- Boost your personal productivity*

* before sharing your API with other people

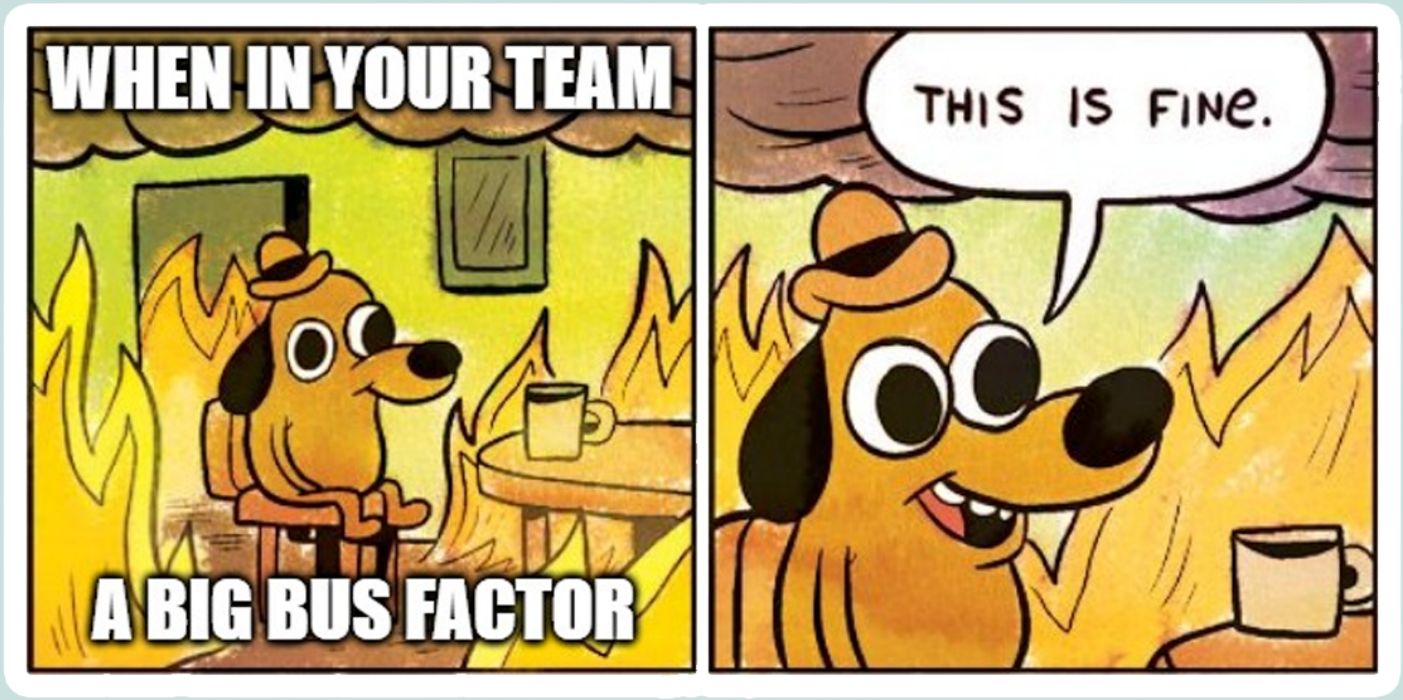
# WITHOUT DOCUMENTATION

PROS

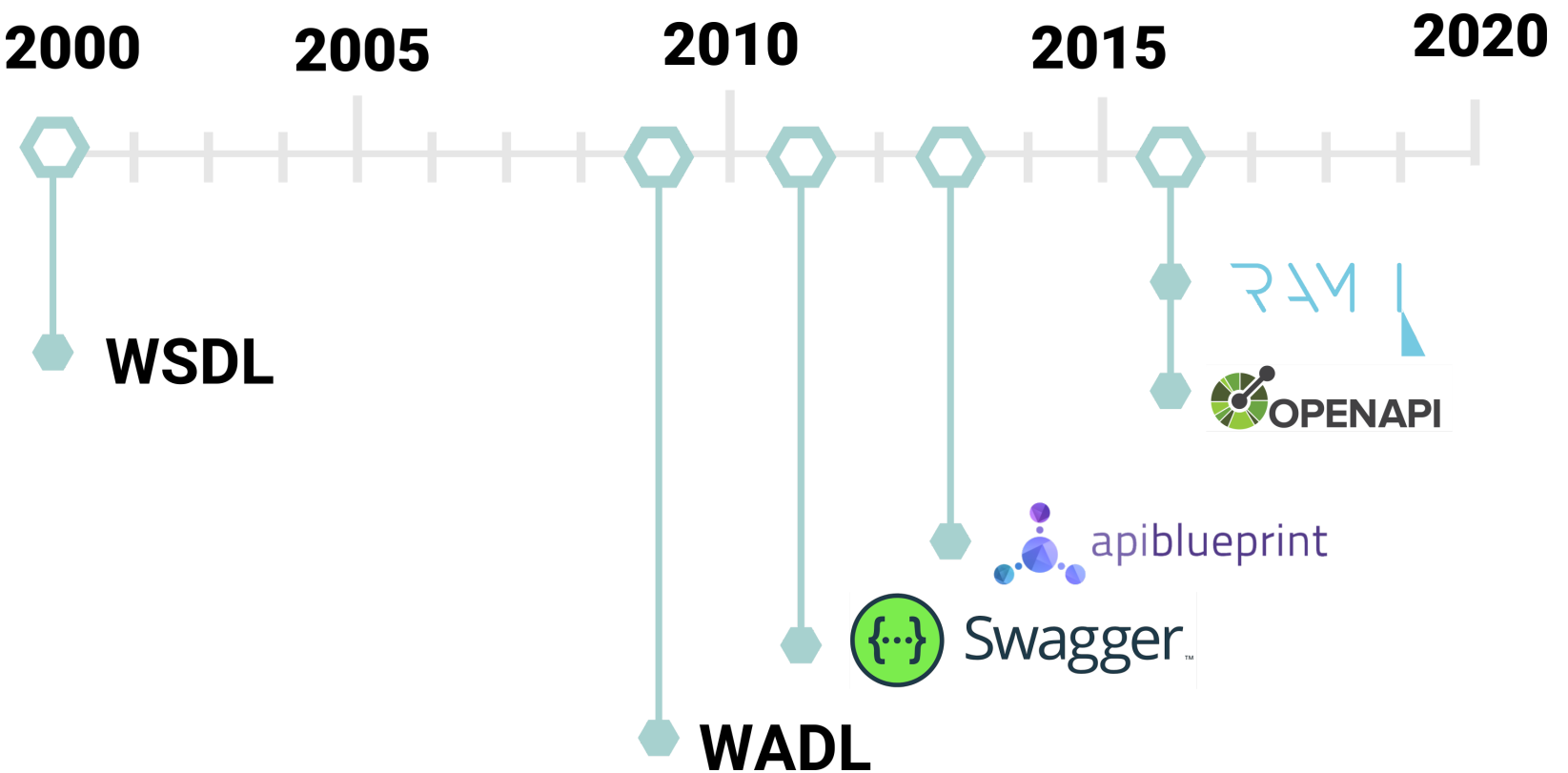
**CONS**

- Difficult collaboration
- Risk of bus factor
- No mechanism of notification about API changes

# WE SHOULD DO SOMETHING

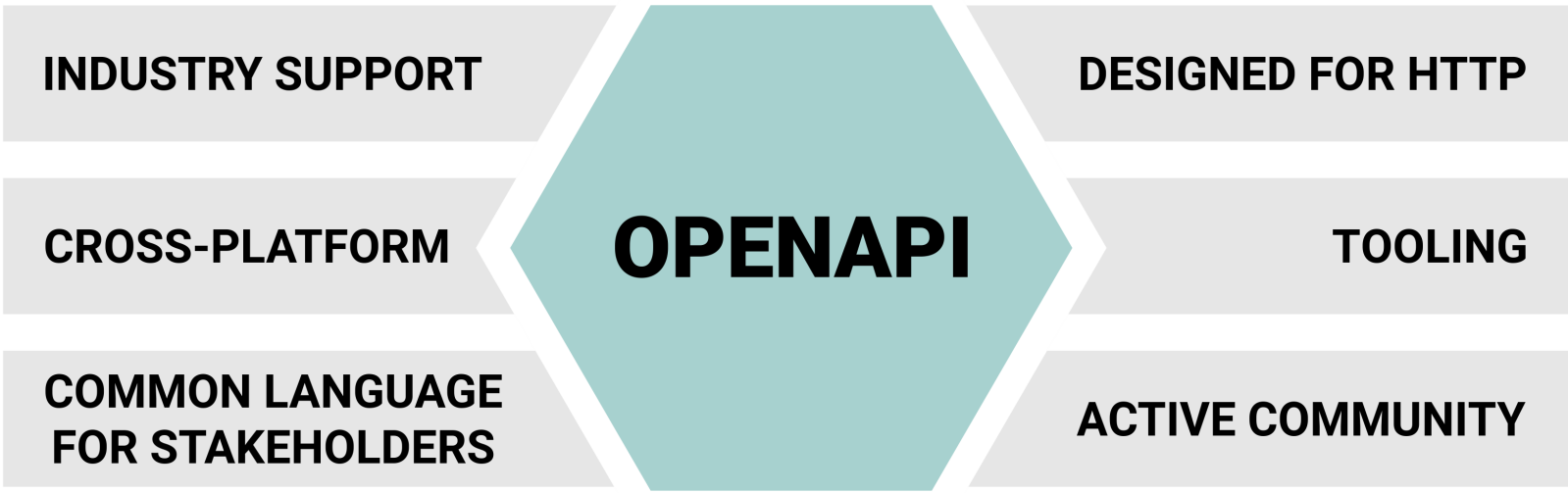


# API DESCRIPTION LANGUAGES



# WHICH ONE IS BETTER?

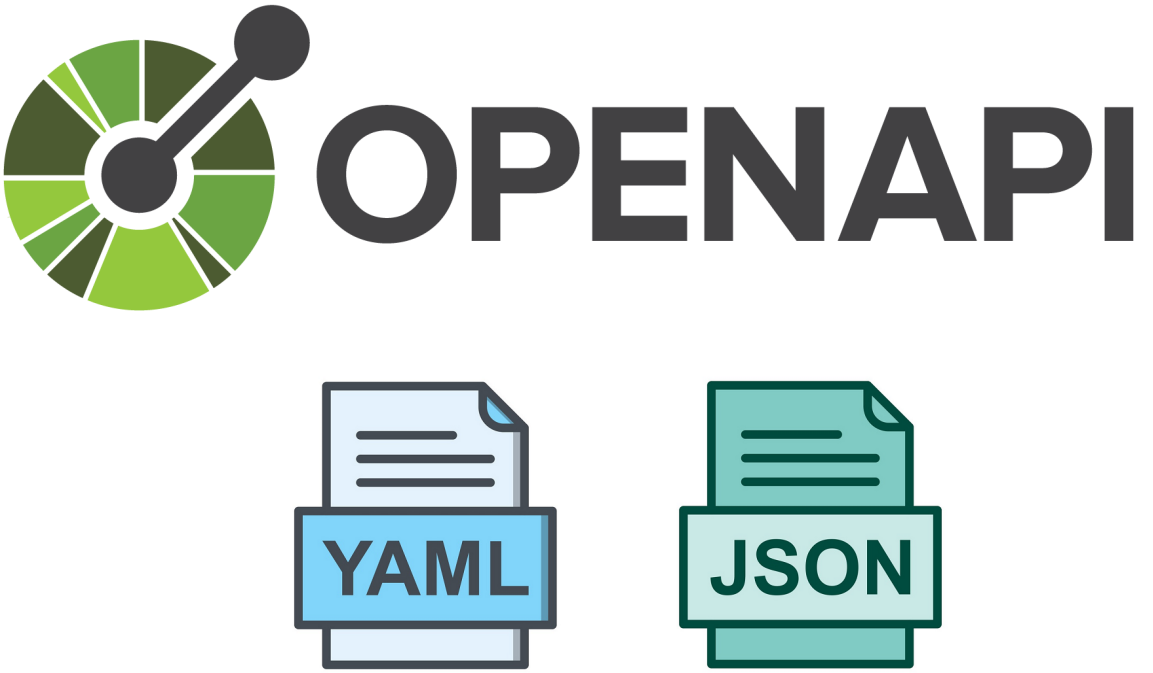
# OPENAPI PITCH



# TELL ME MORE ABOUT OPENAPI!

# OPENAPI SPECIFICATION

## OVERVIEW



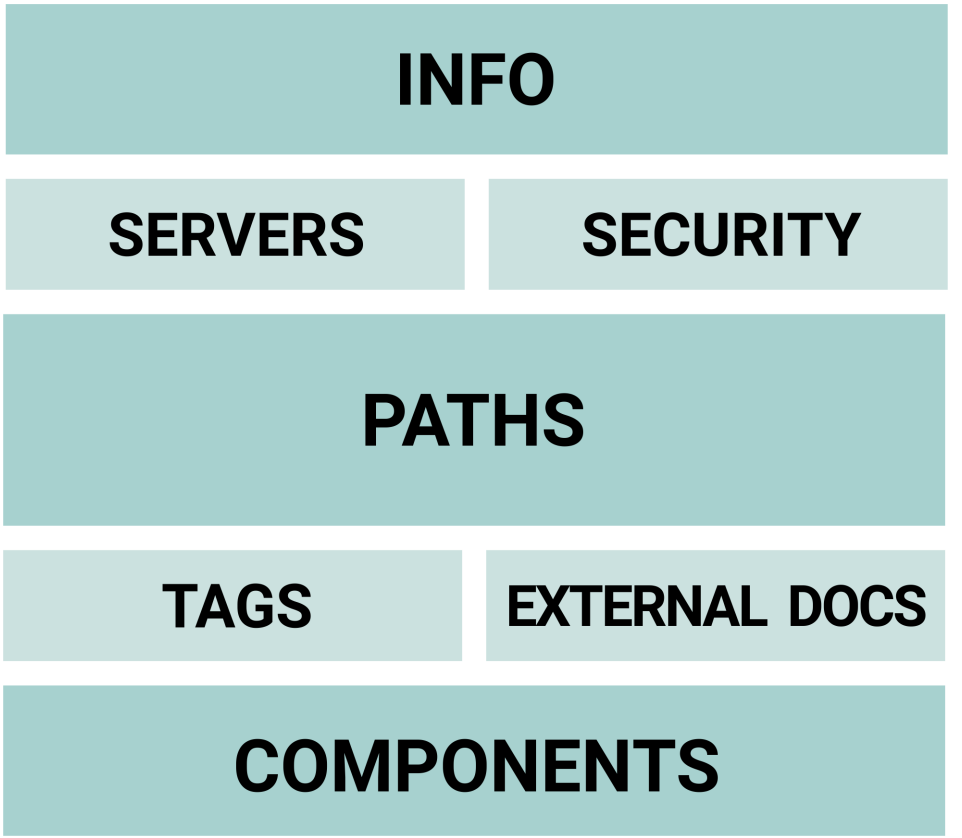
# OPENAPI SPECIFICATION



| | VERSION | RELEASE DATE |
|---|---|---|
| SWAGGER | 1.0 | Aug 10, 2011 |
| | 1.1 | Aug 22, 2012 |
| | 2.0 | Sep 08, 2014 |
| OPENAPI | 3.0.0 | Jul 26, 2017 |
| | 3.0.1 | Dec 06, 2017 |
| | 3.0.2 | Oct 08, 2018 |
| | 3.0.3 | Feb 02, 2020 |
| | 3.1 | Feb 20, 2021 |

# OPENAPI SPECIFICATION





# OPENAPI SPECIFICATION

OVERVIEW

HISTORY

**STRUCTURE**

```yaml
openapi: 3.0.2
info:
  title: Swagger Petstore - OpenAPI 3.0
  description: This is a…s in OAS3.
  termsOfService: http://swagger.io/terms/
  license:
    name: Apache 2.0
    url: http://www.apache.org/licenses/LICENSE-2.0.html
  version: 1.0.5
externalDocs: <2 keys>
servers: <1 item>
tags:
- name: pet
  description: Everything about your Pets
- name: store
  description: Everything about your Store
paths:
  /pet:
    put: <7 keys>
    post: <7 keys>
  /pet/findByStatus:
    get: <7 keys>
  /pet/findByTags:
    get: <7 keys>
  /pet/{petId}:
    get: <7 keys>
    post: <7 keys>
    delete: <7 keys>
```

# OPENAPI SPECIFICATION

OVERVIEW

HISTORY

**STRUCTURE**

```yaml
paths:
  /pet/{petId}:
    get:
      tags:
        - pet
      summary: Find pet by Id
      description: Returns a single pet
      operationId: getPetById
      parameters:
        - name: petId
          in: path
          description: Id of pet to return
          required: true
          schema:
            type: integer
            format: int64
      responses:
        "200":
          description: successful operation
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/Pet'
```

# OPENAPI SPECIFICATION

OVERVIEW

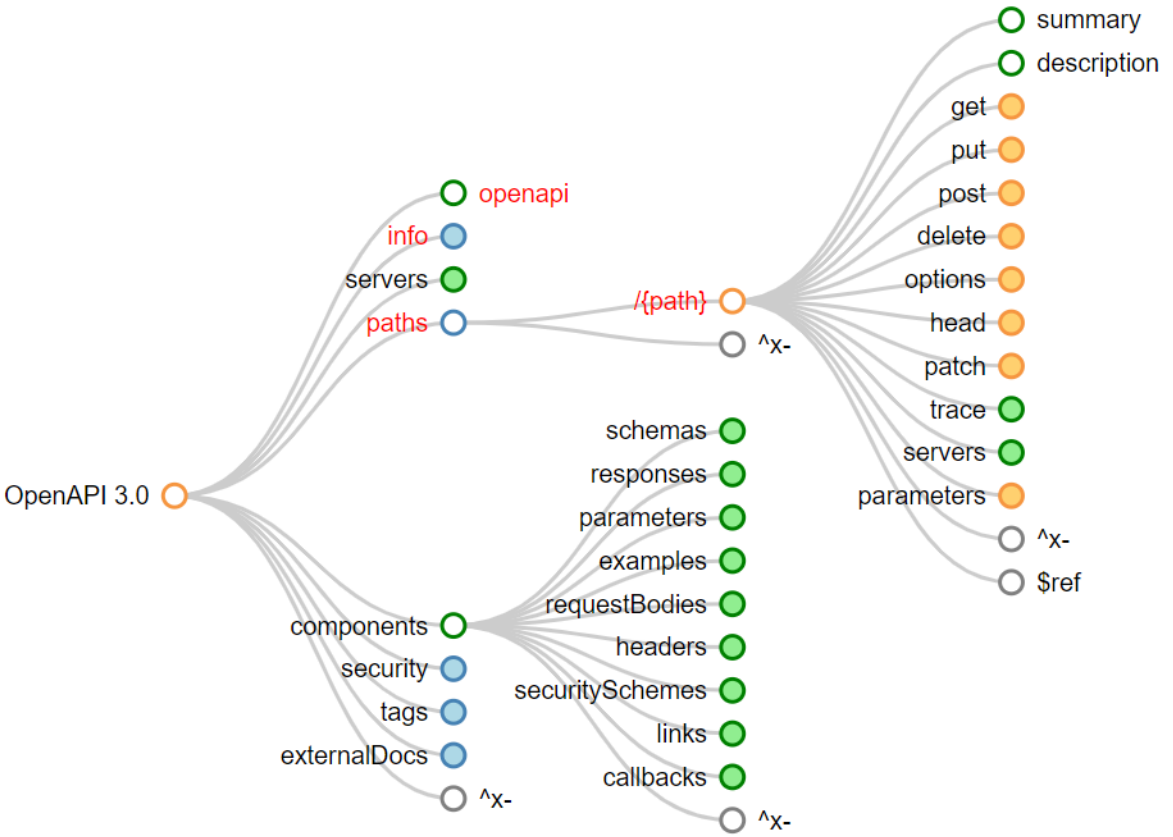
HISTORY

**STRUCTURE**

```yaml
schemas:
  Address:
    type: object
    properties:
      street:
        type: string
        example: 437 Lytton
      city:
        type: string
        example: Palo Alto
      state:
        type: string
        example: CA
      zip:
        type: string
        example: "94301"
```

# OPENAPI SPECIFICATION






openapi-map.apihandyman.io

# OPENAPI SPECIFICATION



**LAST VERSION**

- spec.openapis.org/oas/v3.1.0

**EXAMPLES**

- petstore3.swagger.io
- github.com/github/rest-api-description
- github.com/azure/azure-rest-api-specs

# OPENAPI INITIATIVE

## OVERVIEW

# OPENAPI INITIATIVE

OVERVIEW

**MEMBERS**

# OPENAPI INITIATIVE




=

**STANDARD**


=

**TOOLING**

FROM SMARTBEAR COMPANY

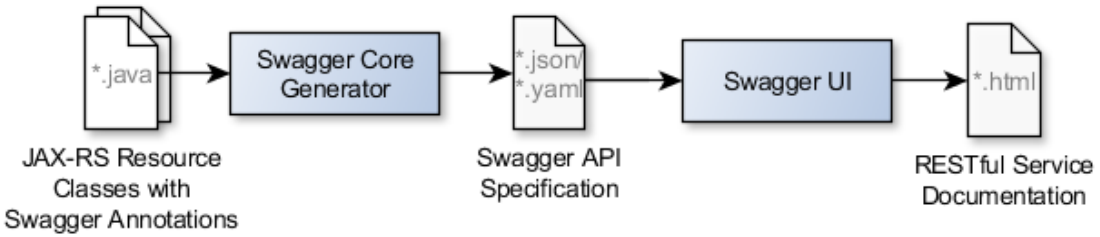
# OPENAPI INITIATIVE

- openapis.org
- github.com/oai
- twitter.com/openapispec
- youtube.com/channel/UCaw7jsGhrsaoSIYPPUF5q2w

# AAAAAAAAAAAAAAAAA

# OAS GENERATION FROM SOURCE CODE

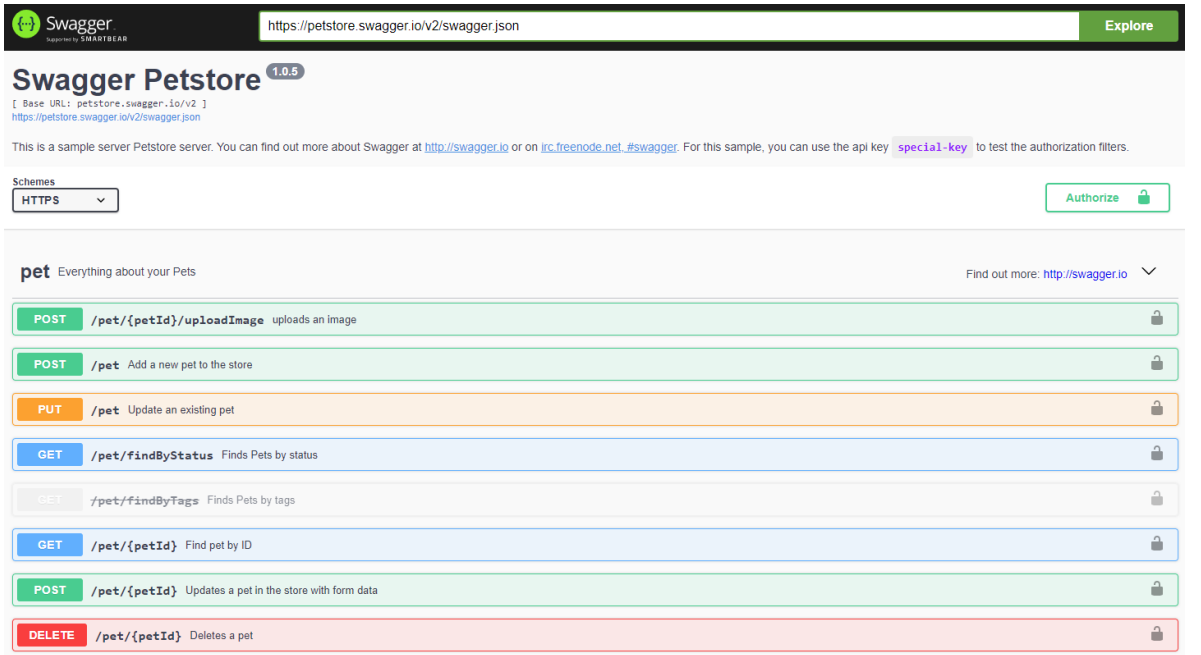
## APPROACH



# OAS GENERATION FROM SOURCE CODE

APPROACH

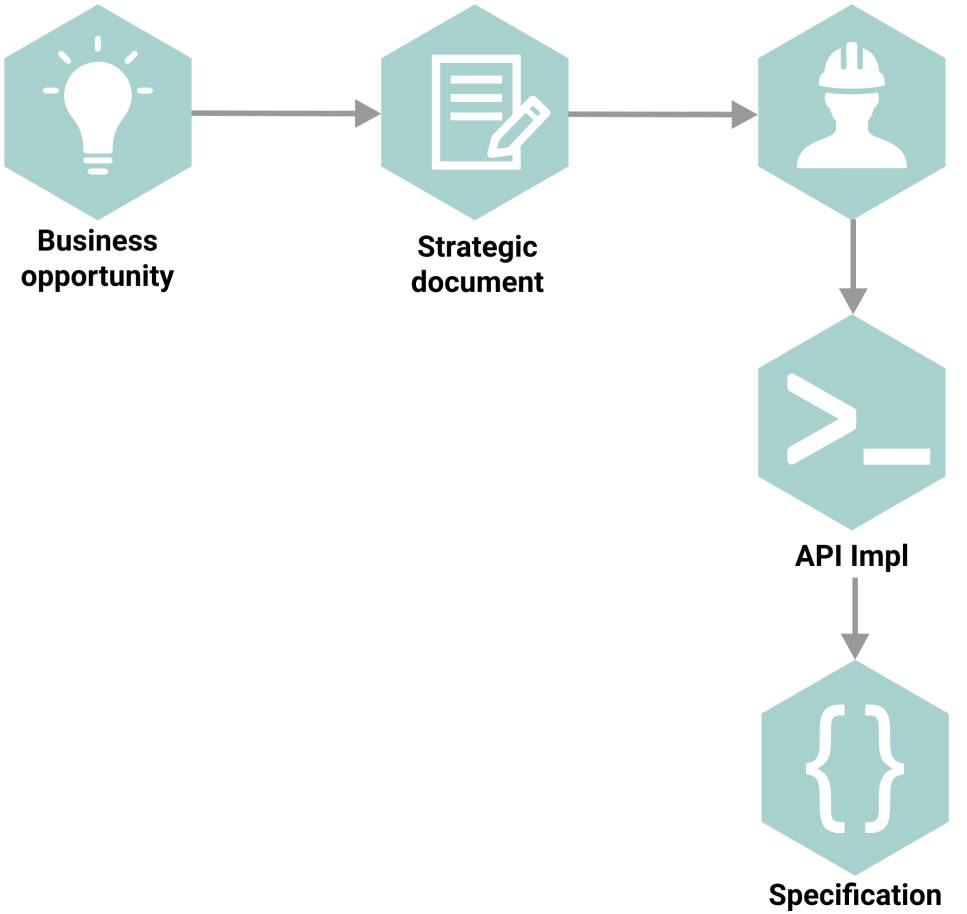
**EXAMPLE**



# GENERATE DOCS USING ONE BUTTON?

# CODE FIRST

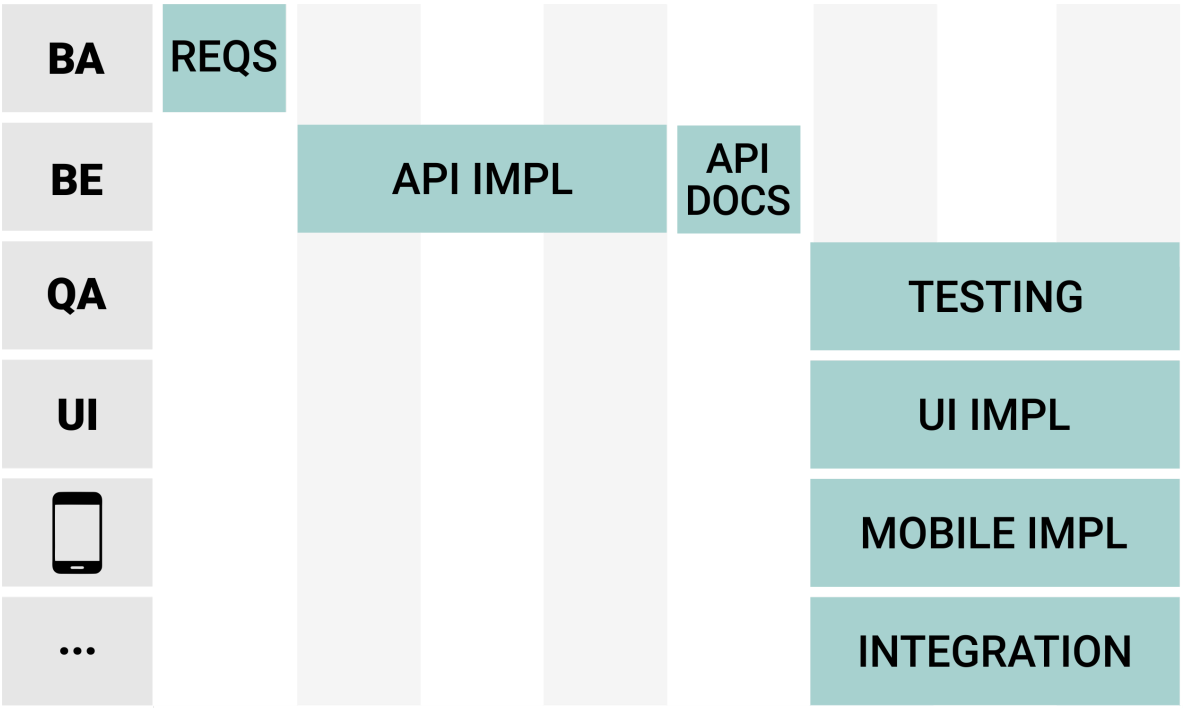
## OVERVIEW



# CODE FIRST

OVERVIEW

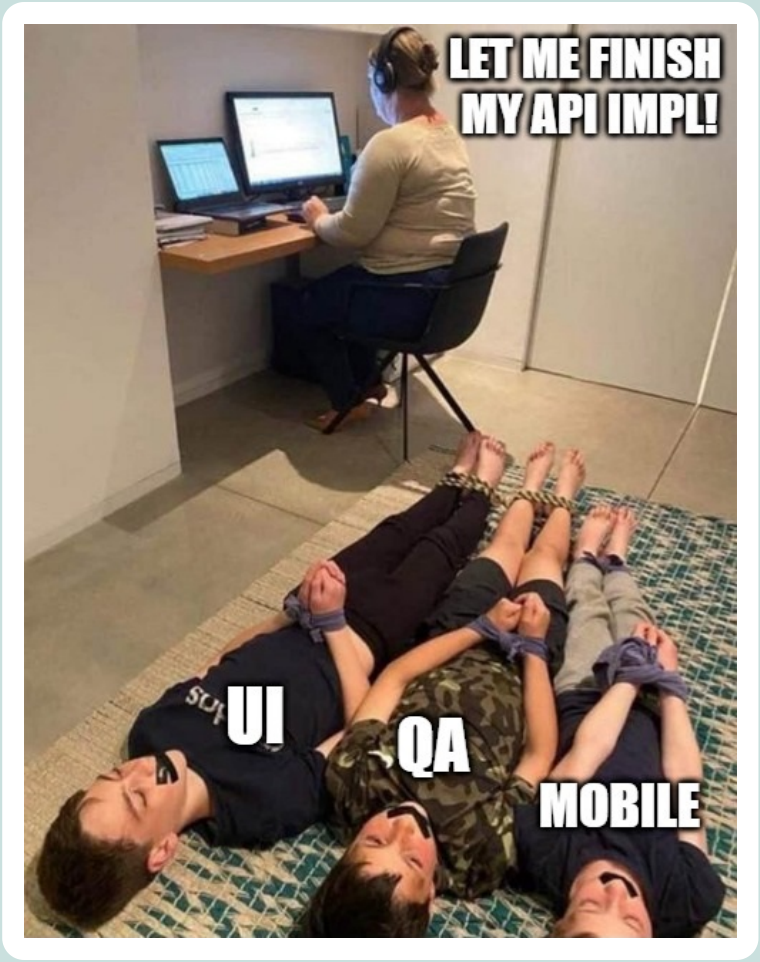
**WORKFLOW**



# CODE FIRST

- Easy to implement and use
- Documentation is a part of process out of the box
- Documentation is up to date always
- Code is single source of truth

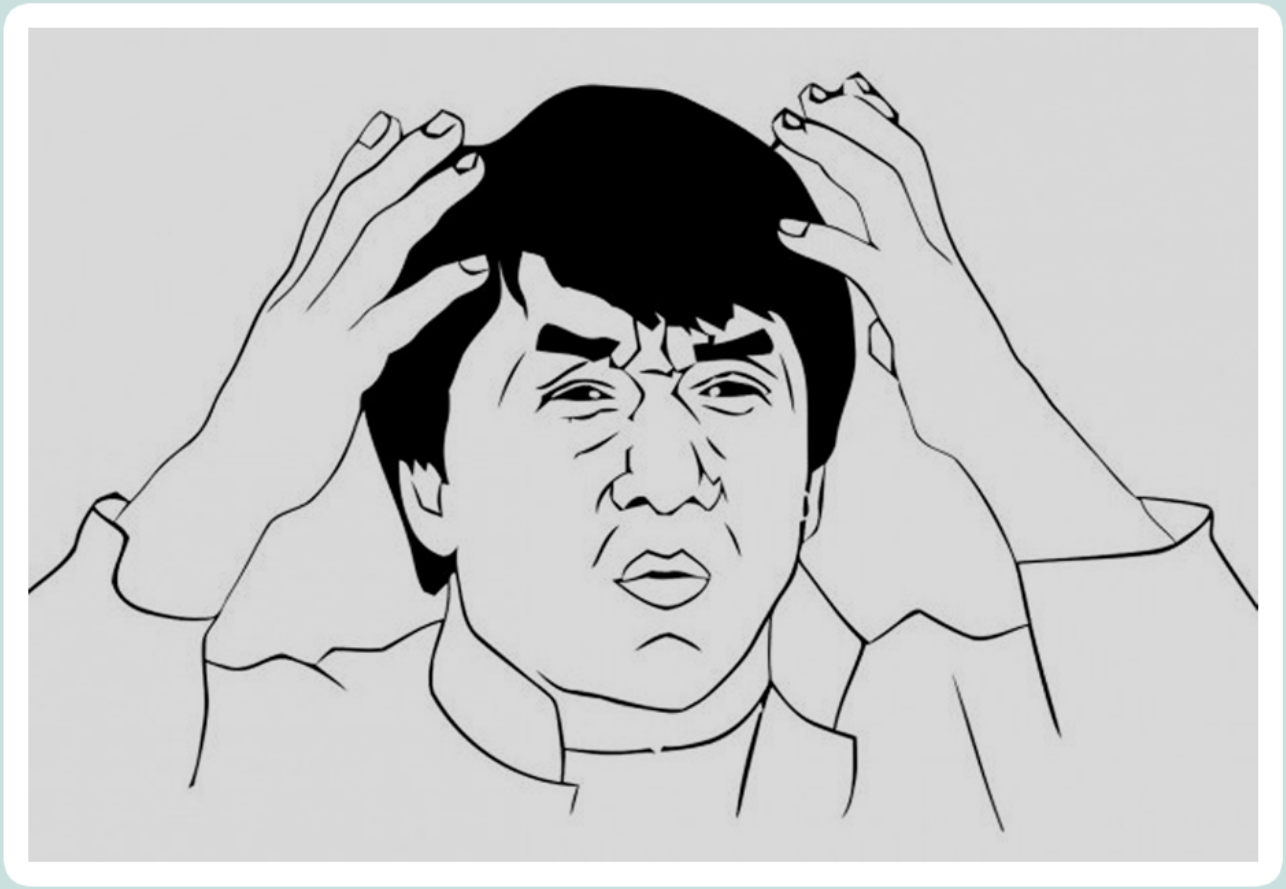
# CODE FIRST IN REAL LIFE



# CODE FIRST

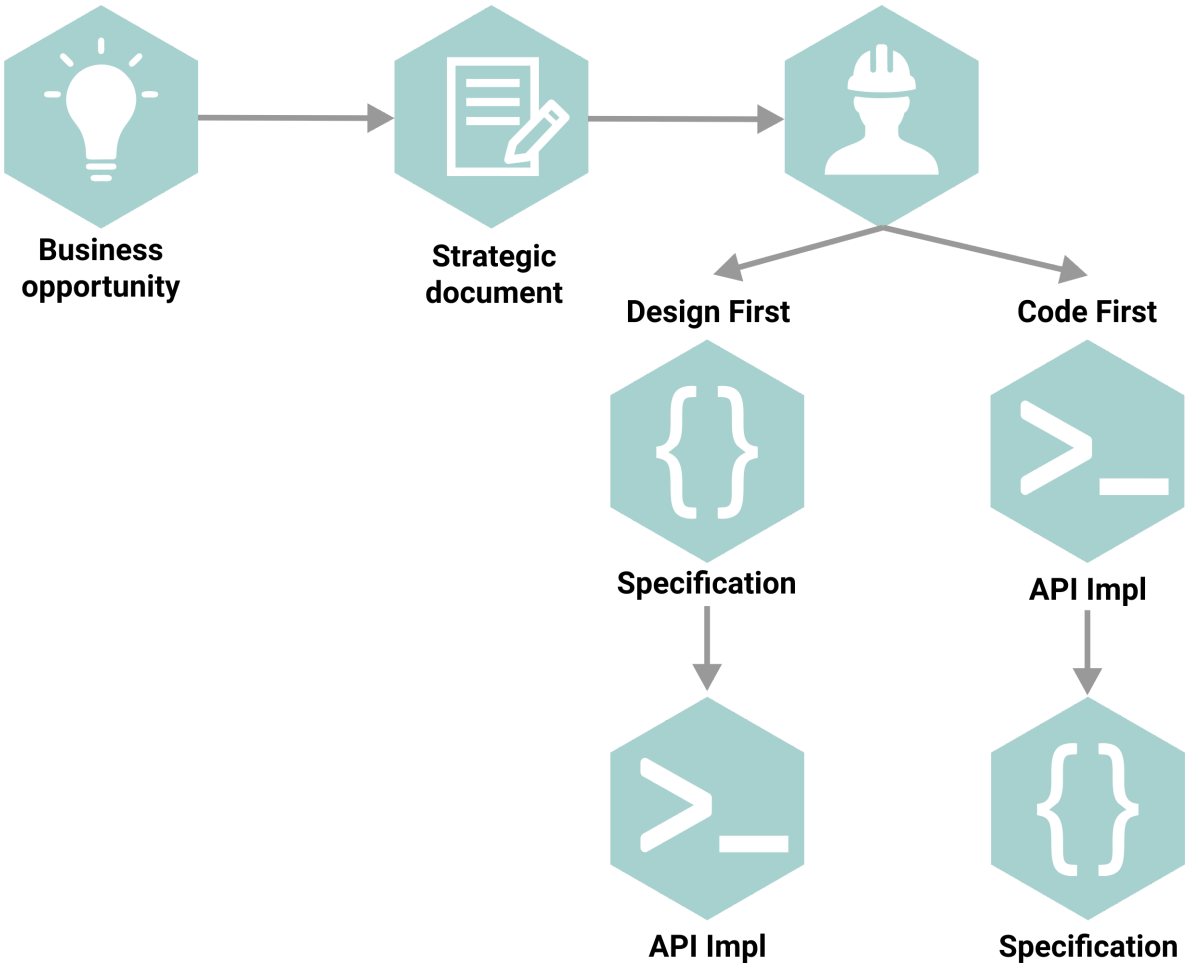
- API stakeholders should wait the implementation part
- Documentation is without business details by default
- Impossible to change documentation without developers

# WRITE OAS MANUALLY, REALLY?



# CONTRACT FIRST

## OVERVIEW



# CONTRACT FIRST

OVERVIEW

**SYNONYMS**

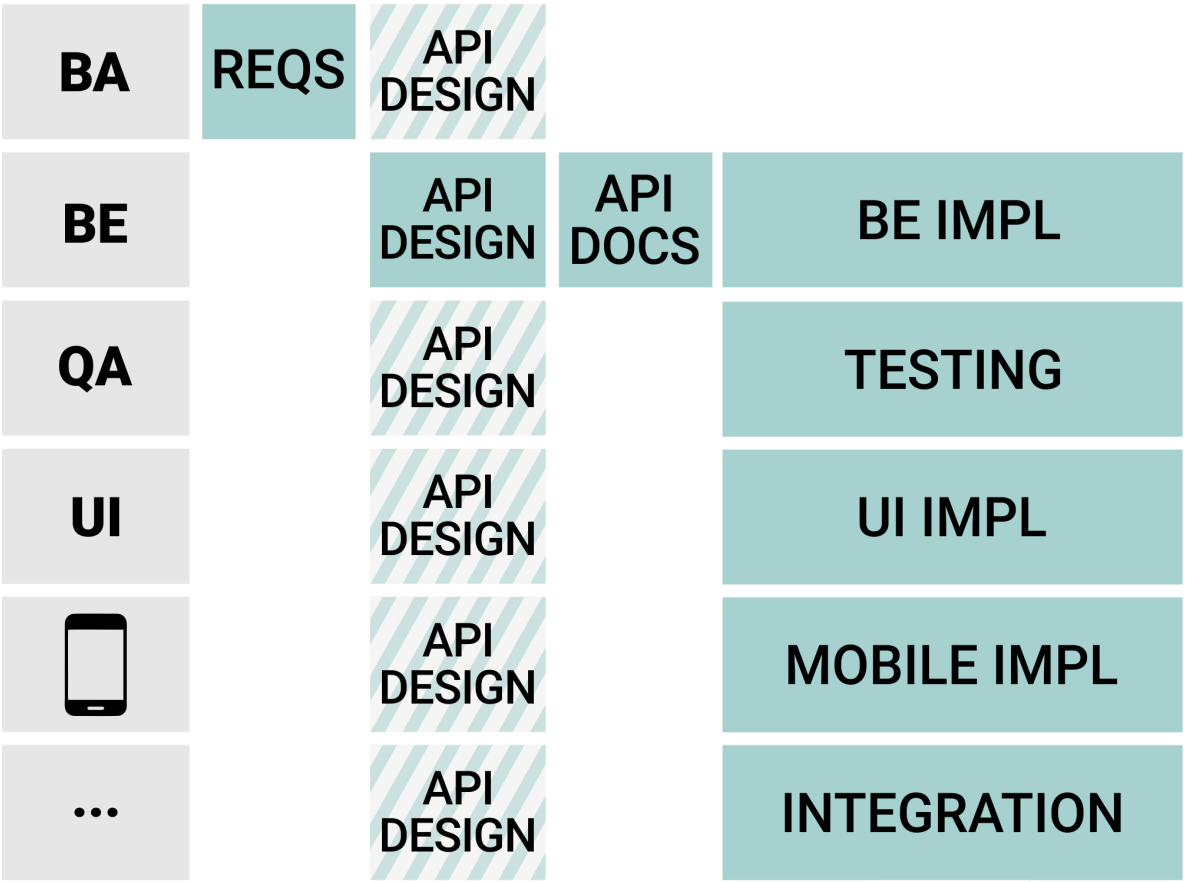
API

CONTRACT
DESIGN
SPECIFICATION

FIRST

# CONTRACT FIRST





# CONTRACT FIRST



- Contract is single source of truth
- Parallel API-implementation and API-consumers work
- Design phase with involvement of any stakeholder(s)
- Not only developers can change documentation
- (potentially) You can re-implement your application easily.
- Complexity is immediately visible

# WHY SHOULD WE WAIT?

# CONTRACT FIRST

OVERVIEW

SYNONYMS

WORKFLOW

PROS

**CONS**

- Writing and review of specification is bottleneck of your process
- You should change your processes (not just development)
- You should be great in writing texts and soft words
- Specification review may take time to agreed on

# MAY BE TOOLING IS COOL?

# OPENAPI TOOLING

## OVERVIEW

- Converters
- Validators
- Documentation
- DSL
- Text & GUI Editors
- Mock Servers
- Parsers
- Generators
- Security
- Testing

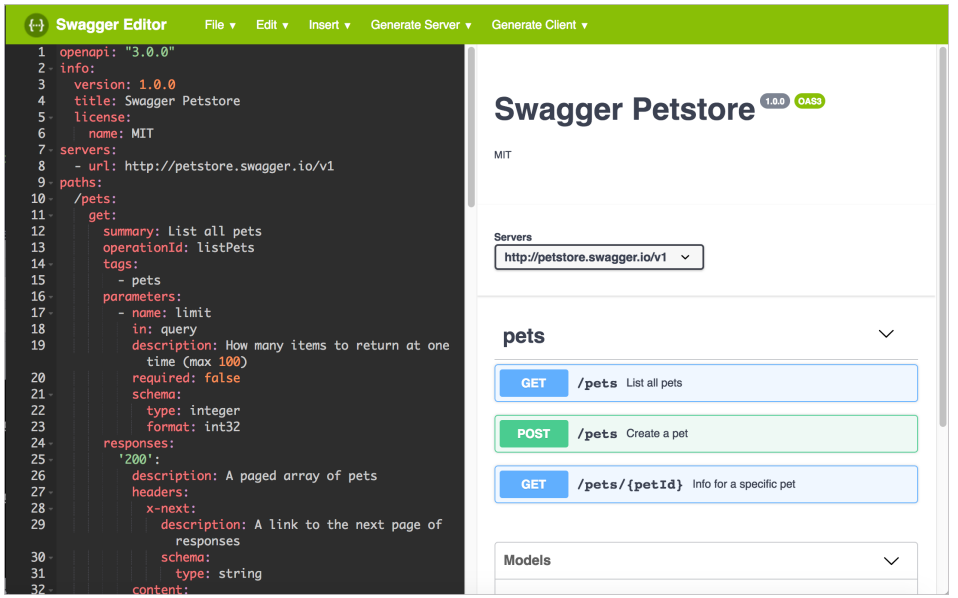
See openapi.tools

# OPENAPI TOOLING

OVERVIEW

**TEXT EDITORS**

- Intellij Idea
- Visual Studio
- Stoplight Desktop
- Swagger Editor



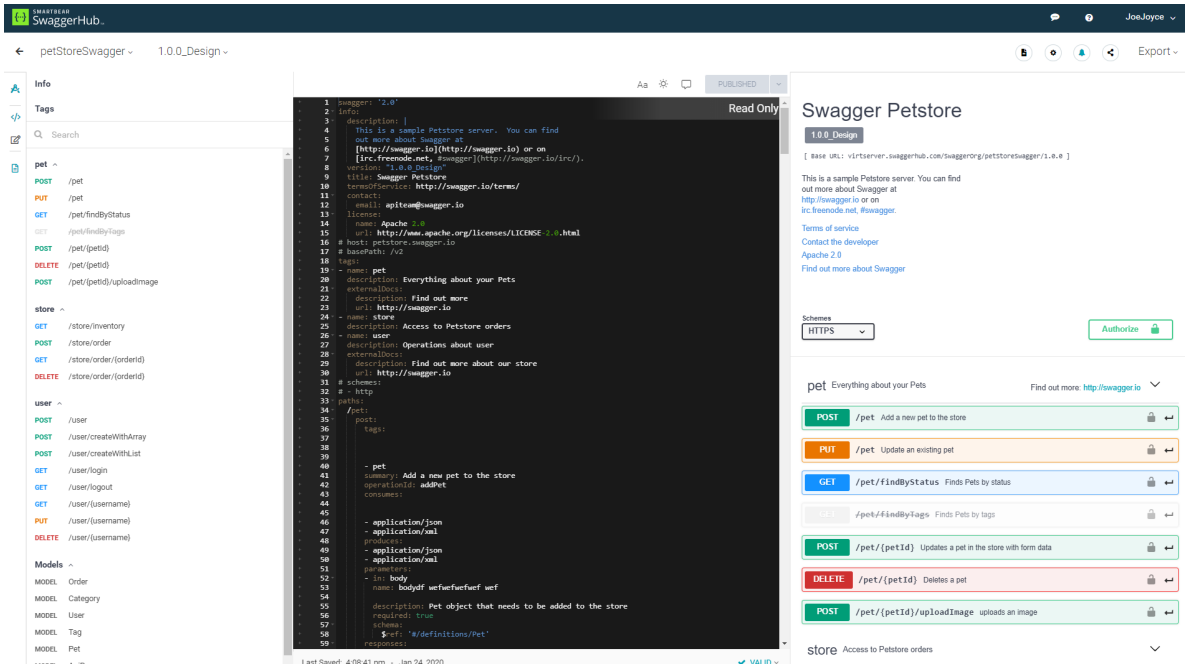
# OPENAPI TOOLING

OVERVIEW

TEXT EDITORS

**GUI EDITORS**

swagger.io/tools/swaggerhub



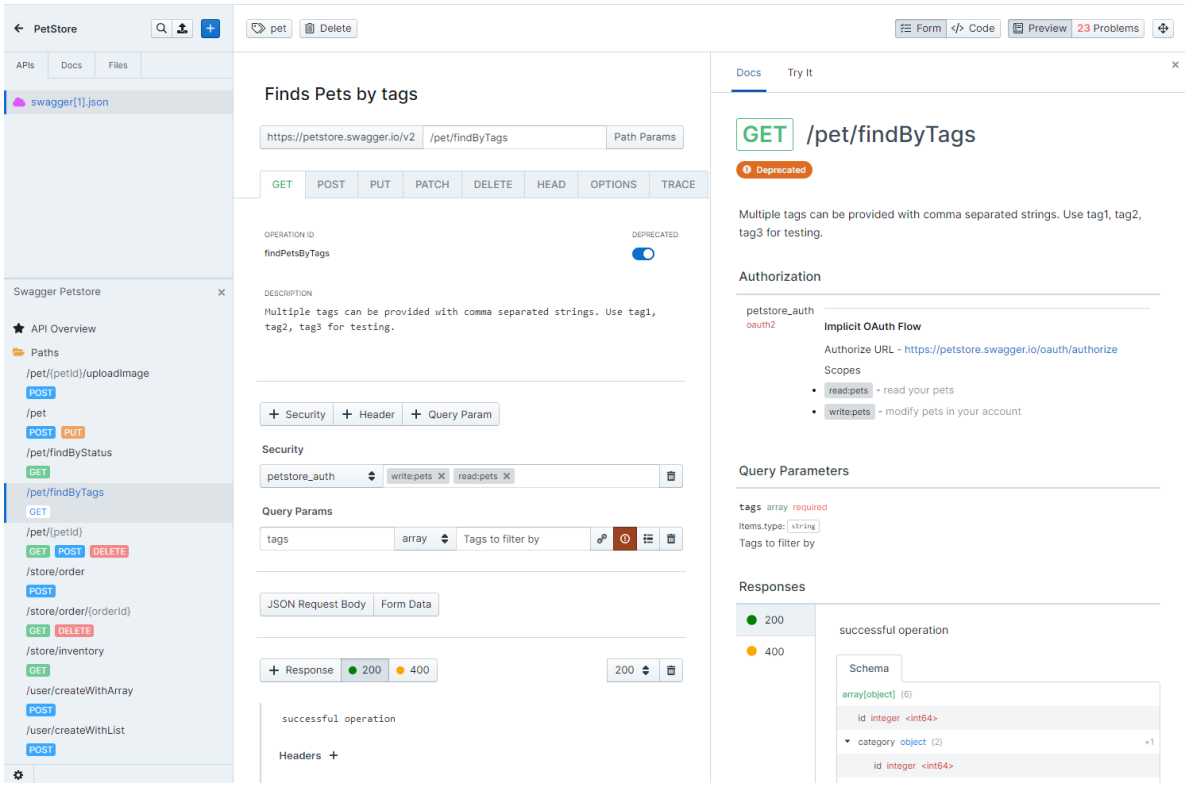
# OPENAPI TOOLING

OVERVIEW

TEXT EDITORS

**GUI EDITORS**

stoplight.io



# OPENAPI TOOLING



- github.com/openapitools/openapi-style-validator
- stoplight.io/open-source/spectral

```
▶ spectral lint petstore.yaml
Adding OpenAPI 3.x functions
OpenAPI 3.x detected

petstore.yaml
  2:6    warning   info-contact             Info object should contain `contact` object.
  2:6    warning   info-description         OpenAPI object info `description` must be present and non-empty string.
  11:9   warning   operation-description    Operation `description` must be present and non-empty string.
 42:10   warning   operation-description    Operation `description` must be present and non-empty string.
  57:9   warning   operation-description    Operation `description` must be present and non-empty string.

✖ 5 problems (0 errors, 5 warnings, 0 infos)
```

# OPENAPI TOOLING

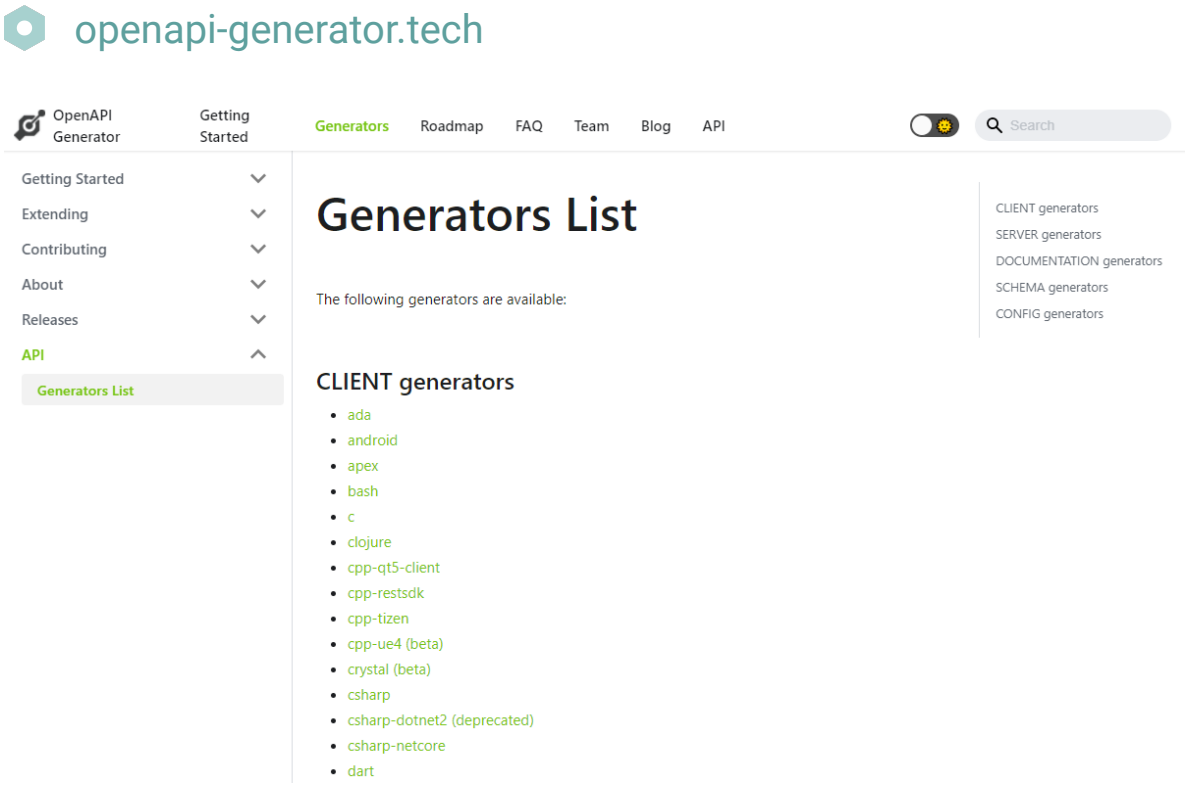
OVERVIEW

TEXT EDITORS

GUI EDITORS

VALIDATORS

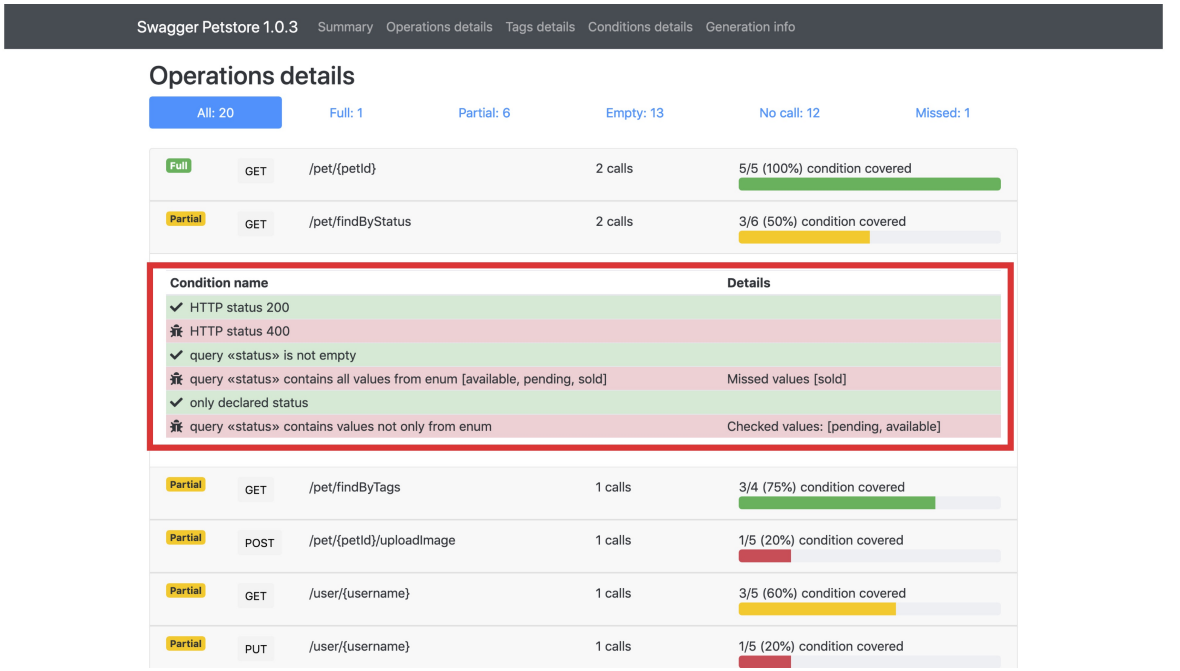
**GENERATORS**



# OPENAPI TOOLING

- OVERVIEW
- TEXT EDITORS
- GUI EDITORS
- VALIDATORS
- GENERATORS
- **COVERAGE**

github.com/viclovsky/swagger-coverage



# OPENAPI TOOLING

- github.com/danielgtaylor/apisprout
- stoplight.io/open-source/prism

```
[➜ Stoplight prism mock --help
Start a mock server with the given spec file

USAGE
  $ prism mock SPEC

ARGUMENTS
  SPEC  Path to a spec file. Can be both a file or a fetchable resource on the web

OPTIONS
  -d, --dynamic        Dynamically generate examples.
  -h, --host=host      [default: 127.0.0.1] Host that Prism will listen to.
  -m, --multiprocess   Fork the http server from the CLI
  -p, --port=port      (required) [default: 4010] Port that Prism will run on.
```

# OPENAPI TOOLING

- OVERVIEW
- TEXT EDITORS
- GUI EDITORS
- VALIDATORS
- GENERATORS
- COVERAGE
- MOCKING
- **VERSIONING**

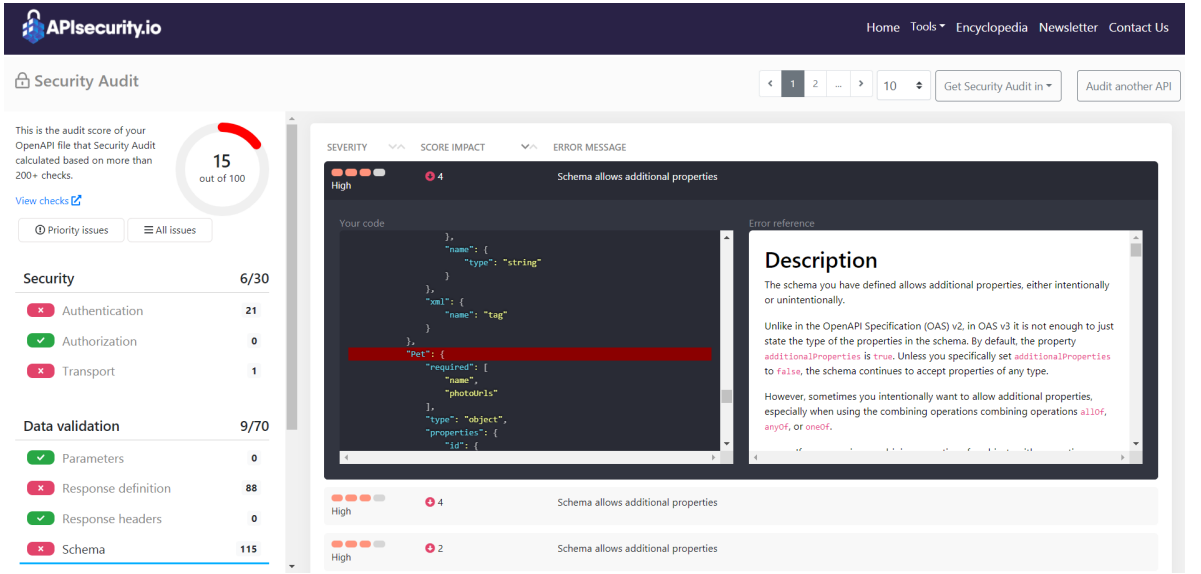
github.com/azure/openapi-diff

```
vishrut@visshamac openapi-diff $ oad compare --help
Commands:
  compare <old-spec> <new-spec>  Compares old and new open api specification for
                                 breaking changes.

Options:
  --version          Show version number                                [boolean]
  -l, --logLevel     Set the logging level for console.
  [choices: "off", "json", "error", "warn", "info", "verbose", "debug", "silly"]
                                                                [default: "warn"]
  -f, --logFilepath  Set the log file path. It must be an absolute filepath. By
                     default the logs will stored in a timestamp based log file
                     at "/Users/vishrut/oad_output".
  -j, --inJson       A boolean flag indicating whether output format of the
                     messages is json.                [boolean] [default: true]
  -h, --help         Show help                                          [boolean]
  -o, --oldTagName   The tag name for the old specification file.  If include it
                     indicates that the old spec file is a readme file
  -n, --newTagName   The tag name for the new specification file.  If include it
                     indicates that the new spec file is a readme file
```

# OPENAPI TOOLING

- 42crunch.com/api-security-audit
- zaproxy.org/docs/desktop/addons/openapi-support
- apisecurity.io/tools/audit



# PITFALLS OF STRICT CONTRACT FIRST

- You should be ready for changes in your DEV/QA/BA/Workflow/etc. processes
- Text writing skill is really important
- Our design mistakes become more transparent
- Focus only on API design in OAS and don't expose impl details
- Background scheduled jobs are not supported
- OAS doesn't cover async messaging between apps
- GraphQL API is not supported out of the box
- Tooling are still in development

# CHOOSING THE RIGHT APPROACH

**CONTRACT FIRST WHEN**

- Developer Experience Matters
- Delivering Mission-Critical APIs
- Ensuring Good Communication

**CODE FIRST WHEN**

- Delivery Speedy Matters
- Developing Internal APIs

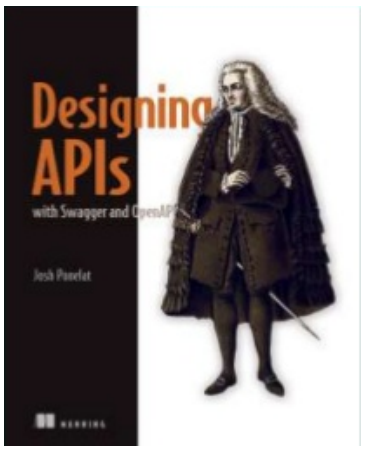
# CONCLUSION

- No documentation ⇒ Code First ⇒ Contract First
- Initial problem was solved and we can specify, design and parallelize API-implementation and API-consumers work
- It's still possible to use hybrid approach

# BONUS LINKS

## USEFUL LINKS

- idratherbewriting.com/learnapidoc
- apisyouwonthate.com/blog
- apihandyman.io/blog
- apis.guru/awesome-openapi3
- API the Docs Youtube Channel




## WHAT ELSE?

- AsyncAPI - asyncapi.com/docs/specifications/2.0.0
- SLA for OpenAPI - github.com/isa-group/SLA4OAI-Specification