# An API-FIRST Approach

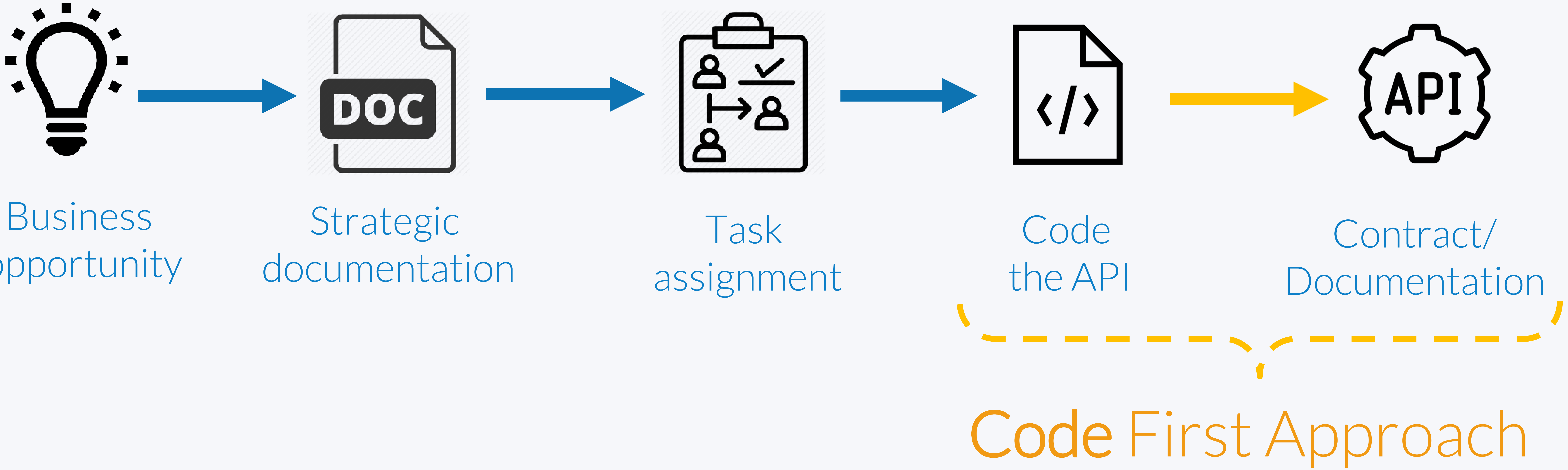
The Best Approach for API Development

1

## Problems in API development

1. **Code-first** approach (current approach)
2. The **problem** with code-first approach

## Code First Approach



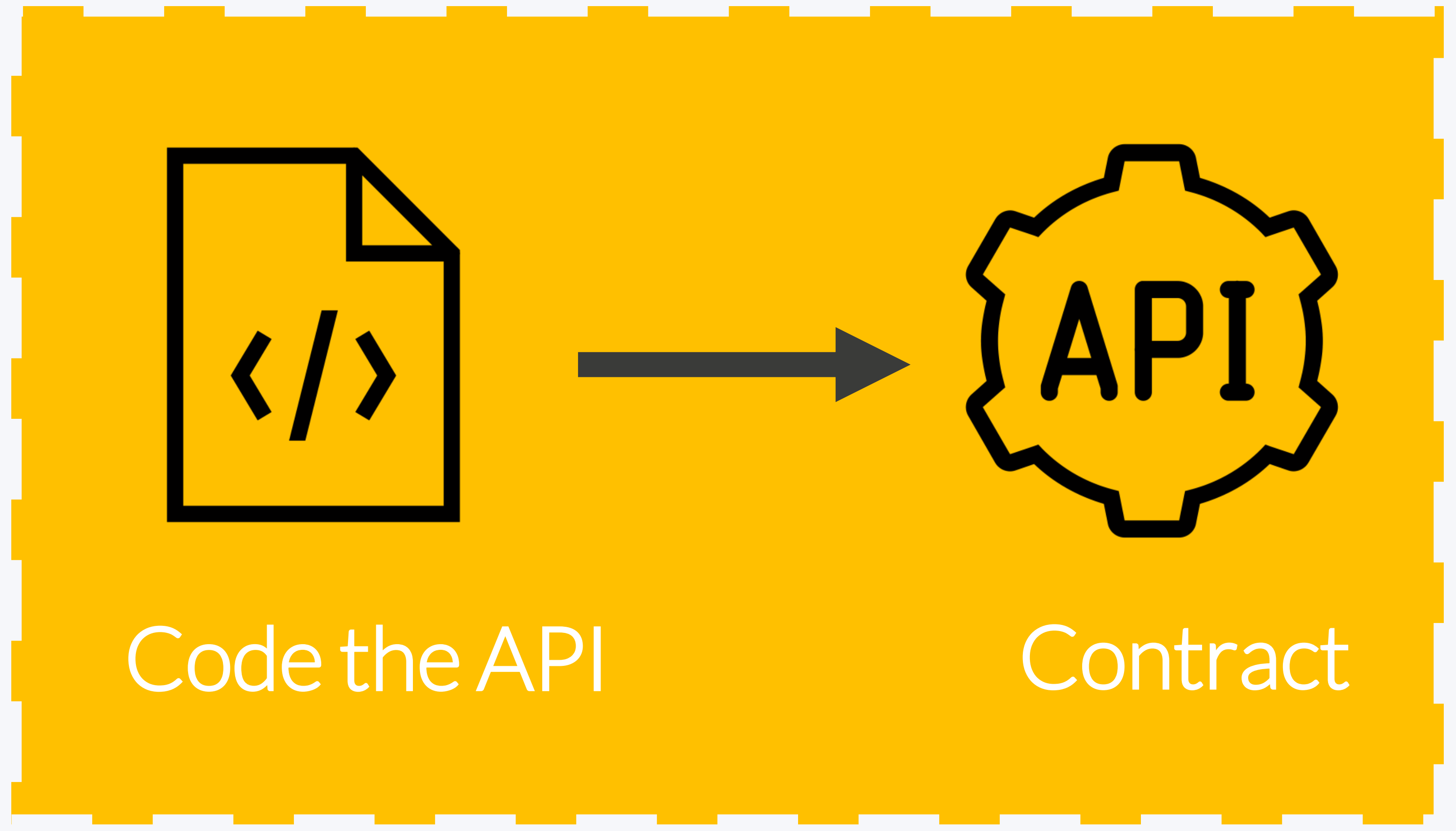
Current problems in API development?

## The Problem with Code First approach

Situation

Implementation of New Features

Code First Approach



Issues

- Team interdependence
- Late detection of issue
- Inconsistency in design
- Error-prone
- Labor-intensive
- Breaking capacity

# Lab

- Use your LLM and try to find out what http://ec2-54-188-50-153.us-west-2.compute.amazonaws.com:9966/petclinic/v3/api-docs is doing.

- Ask for the functionality and the overall purpose of that REST API.
- Look at http://ec2-54-188-50-153.us-west-2.compute.amazonaws.com:9966/petclinic for visualization.

# Lab

- Use lab-s3 for further labs.

- Open the lab and use Copilot for understand what the software solution is doing.
- Let Copilot create a step-by-step guide of how to get the system up and running and execute the test suite.

- When a test fails,
  1. ask Copilot why the test is failing.
  2. After that ask copilot to create a bug report and simple reproduction scenario.
     - If you have an existing report template, use it. Modify if not suitable.
  3. Finally ask for a potential fix for the problem (don’t fix it, just evaluate what has been proposed and if its useful or not)

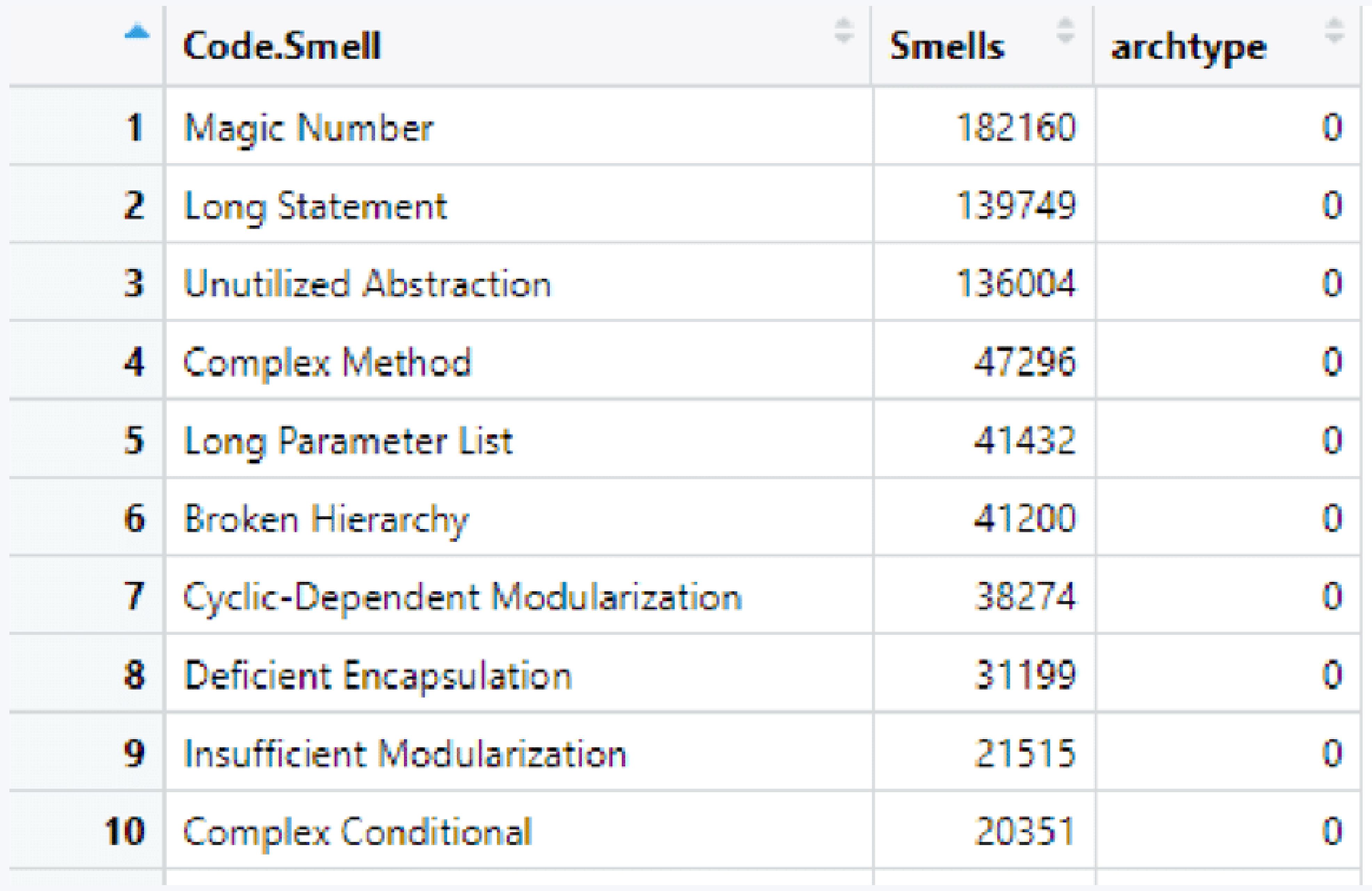
# Lab

- Select one tag (endpoint group) of the API and ask copilot to provide 30 edge cases / test boundaries for it.

- Review what has been created.
- Understand the quality and get a personal meaning of the code that has been generated.

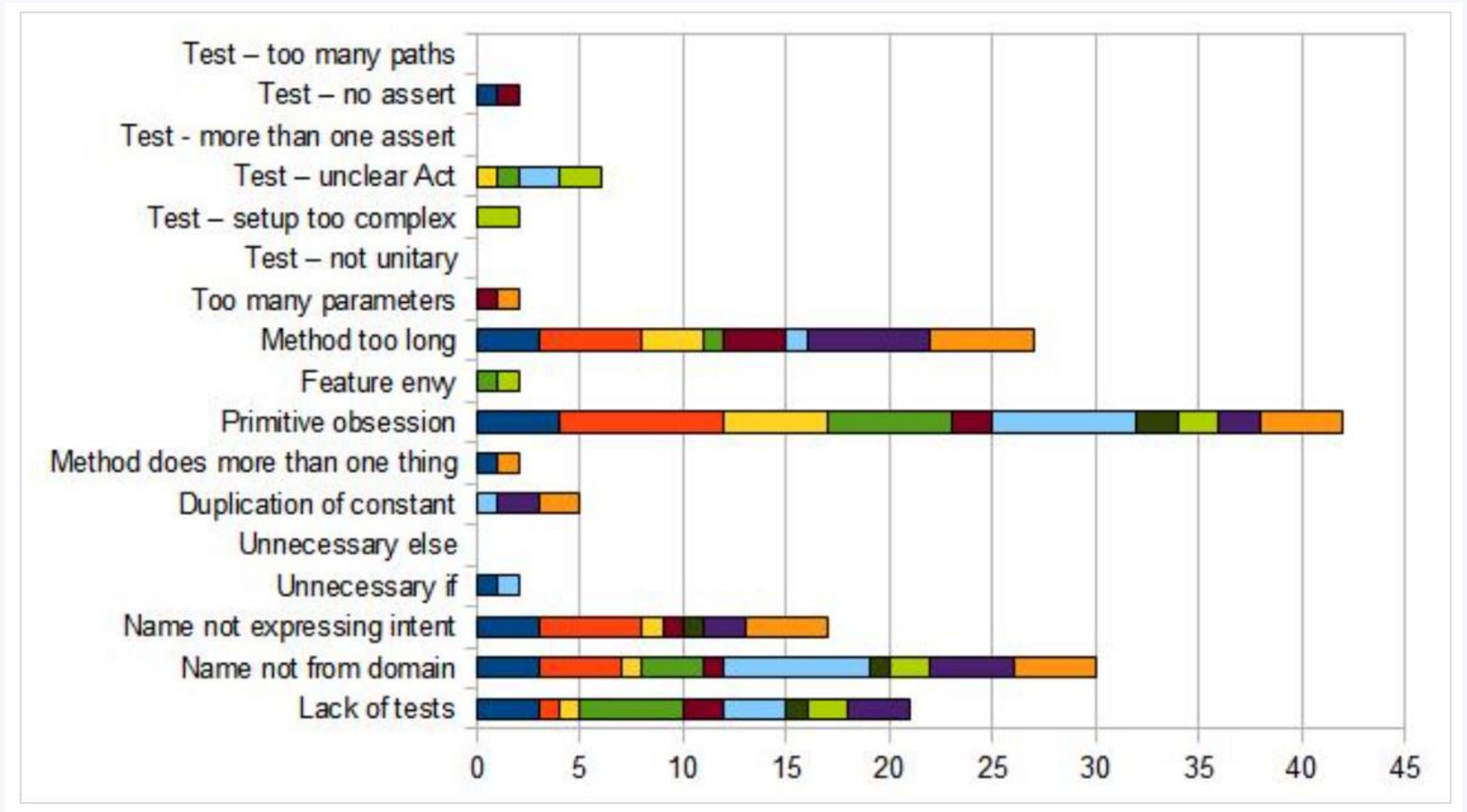
- Ask Github copilot to refactor the existing test suites following fundamental and basic coding principals.

## Identify Code Smells



| | Code.Smell | Smells | archtype |
|---|---|---|---|
| 1 | Magic Number | 182160 | 0 |
| 2 | Long Statement | 139749 | 0 |
| 3 | Unutilized Abstraction | 136004 | 0 |
| 4 | Complex Method | 47296 | 0 |
| 5 | Long Parameter List | 41432 | 0 |
| 6 | Broken Hierarchy | 41200 | 0 |
| 7 | Cyclic-Dependent Modularization | 38274 | 0 |
| 8 | Deficient Encapsulation | 31199 | 0 |
| 9 | Insufficient Modularization | 21515 | 0 |
| 10 | Complex Conditional | 20351 | 0 |

General code smells



Testing related code smells

# Code Smells

- Ask Copilot to list the top 10 common code smells for unit-testing.

- Tell Copilot to compare the top 10 smells with your current code base (from lab-s3) and list the findings in an prioritized order.

- Ask Copilot how to fix the two most critical findings.
- Tell Copilot to fix it.

- After that, ask for a list of best-practices and further improvements and repeat the previous steps.

2

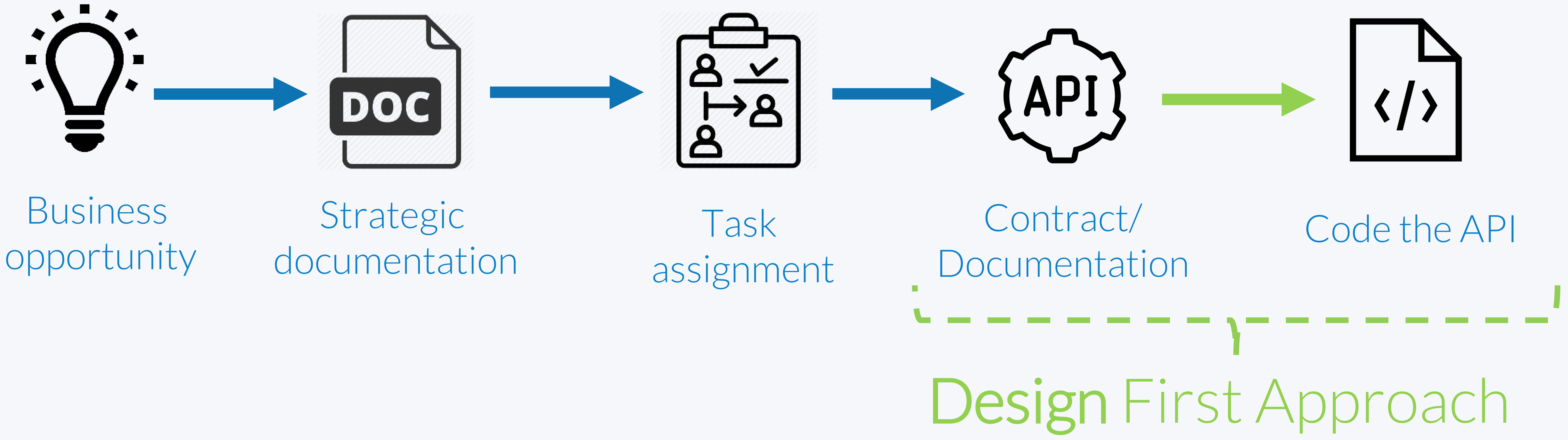
## How to Resolve the problem?

1. Change the current approach
2. Use new technologies

## 2.1 Change the current approach

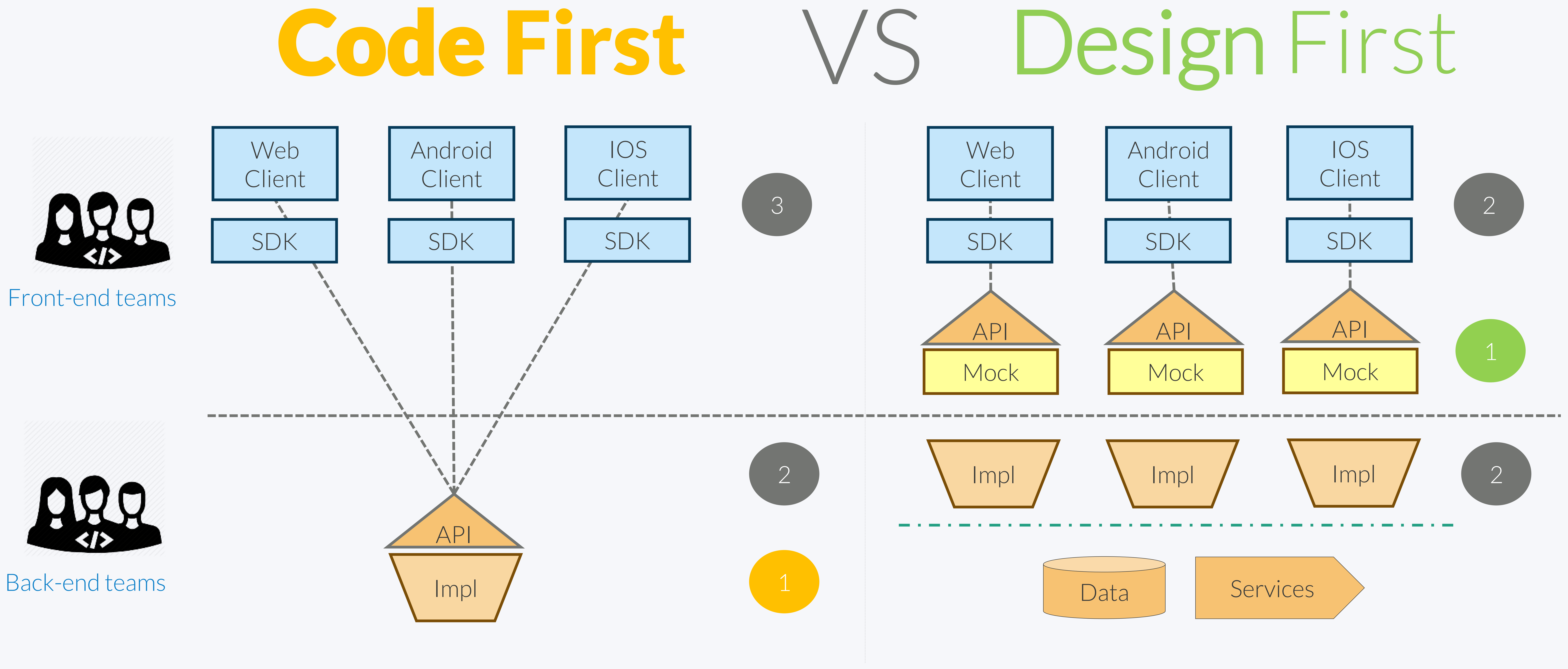
- Using **design-first** approach
- Code-first vs **design-first**
- The **benefits** of design-first approach

## Using **design-first** approach



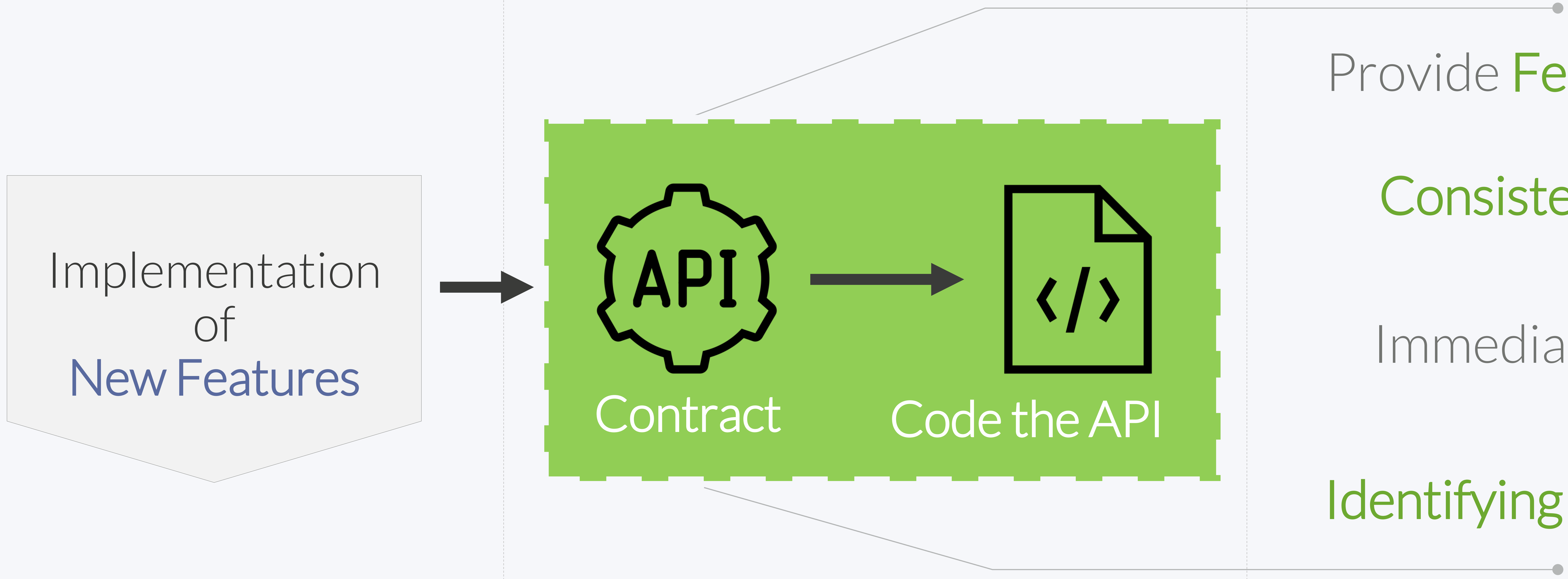
Change the current approach

## Code First vs Design First



## The Benefits of Design First Approach

| Situation | Design First Approach | Issues |
|---|---|---|
| Implementation of New Features | Contract → Code the API | Provide Feedback<br>Consistent Design<br>Immediately Interact<br>Identifying Issues |



## 2.2 Use new technologies

- ❖ Swagger Editor
- ❖ Swagger Codegen
- ❖ Swagger UI
- ❖ Swagger Inspector

Use new technologies

## Swagger Tools

Design
Swagger Editor

Build
Swagger Codegen

Document
Swagger UI

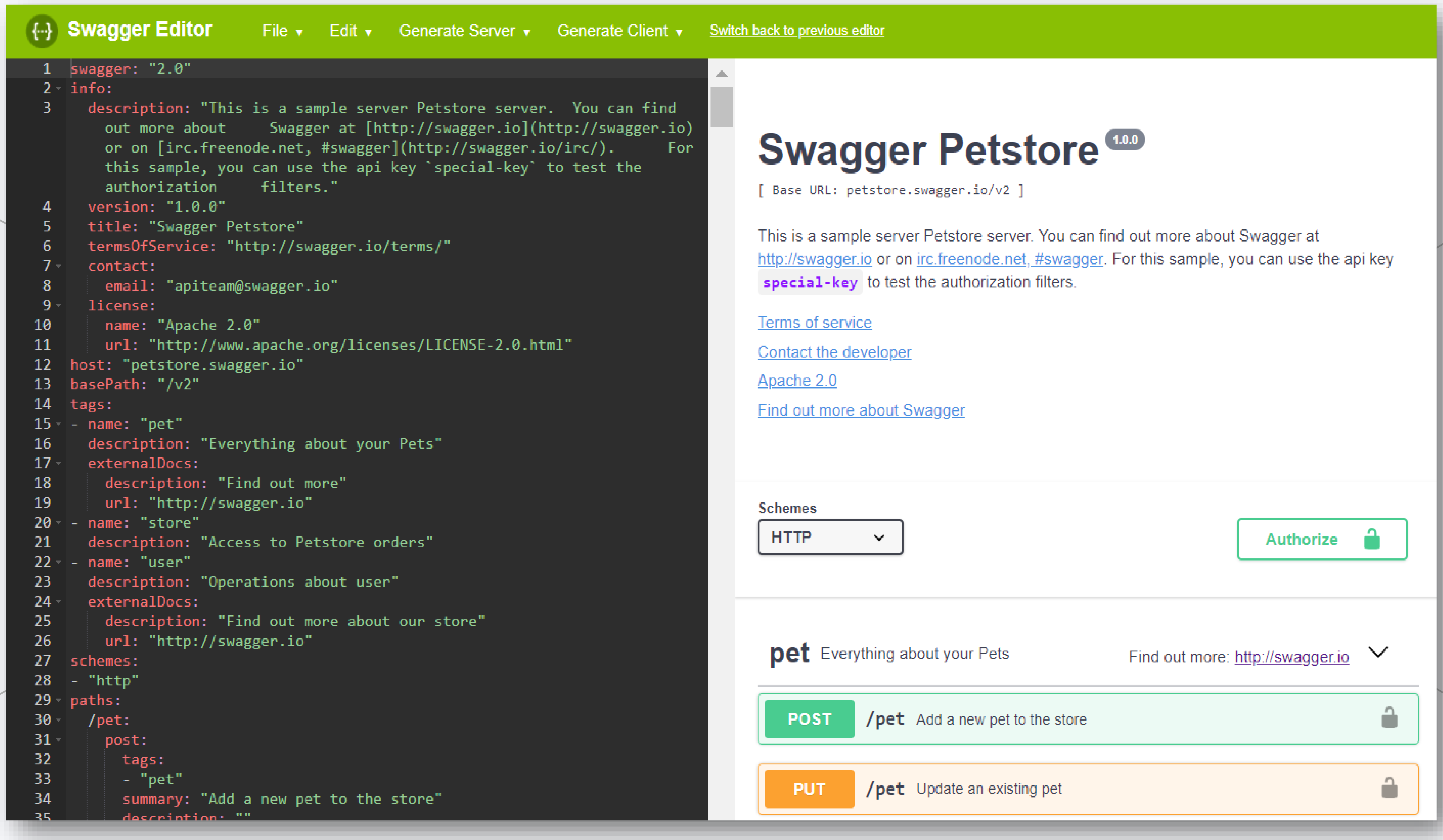
Test
Swagger Inspector

Design & model APIs → Generate client & server code → Visualize & Interact with the API's → API Testing & Documentation Tool

## Swagger Editor

Smart Feedback

Easy-to-use



```yaml
swagger: "2.0"
info:
  description: "This is a sample server Petstore server.  You can find
    out more about     Swagger at [http://swagger.io](http://swagger.io)
    or on [irc.freenode.net, #swagger](http://swagger.io/irc/).      For
    this sample, you can use the api key `special-key` to test the
    authorization     filters."
  version: "1.0.0"
  title: "Swagger Petstore"
  termsOfService: "http://swagger.io/terms/"
  contact:
    email: "apiteam@swagger.io"
  license:
    name: "Apache 2.0"
    url: "http://www.apache.org/licenses/LICENSE-2.0.html"
host: "petstore.swagger.io"
basePath: "/v2"
tags:
- name: "pet"
  description: "Everything about your Pets"
  externalDocs:
    description: "Find out more"
    url: "http://swagger.io"
- name: "store"
  description: "Access to Petstore orders"
- name: "user"
  description: "Operations about user"
  externalDocs:
    description: "Find out more about our store"
    url: "http://swagger.io"
schemes:
- "http"
paths:
  /pet:
    post:
      tags:
      - "pet"
      summary: "Add a new pet to the store"
      description: ""
```

**Swagger Petstore** 1.0.0

[ Base URL: petstore.swagger.io/v2 ]

This is a sample server Petstore server. You can find out more about Swagger at http://swagger.io or on irc.freenode.net, #swagger. For this sample, you can use the api key `special-key` to test the authorization filters.

Terms of service

Contact the developer

Apache 2.0

Find out more about Swagger

Schemes: HTTP

Authorize

**pet** Everything about your Pets — Find out more: http://swagger.io

POST /pet Add a new pet to the store

PUT /pet Update an existing pet

Runs Anywhere

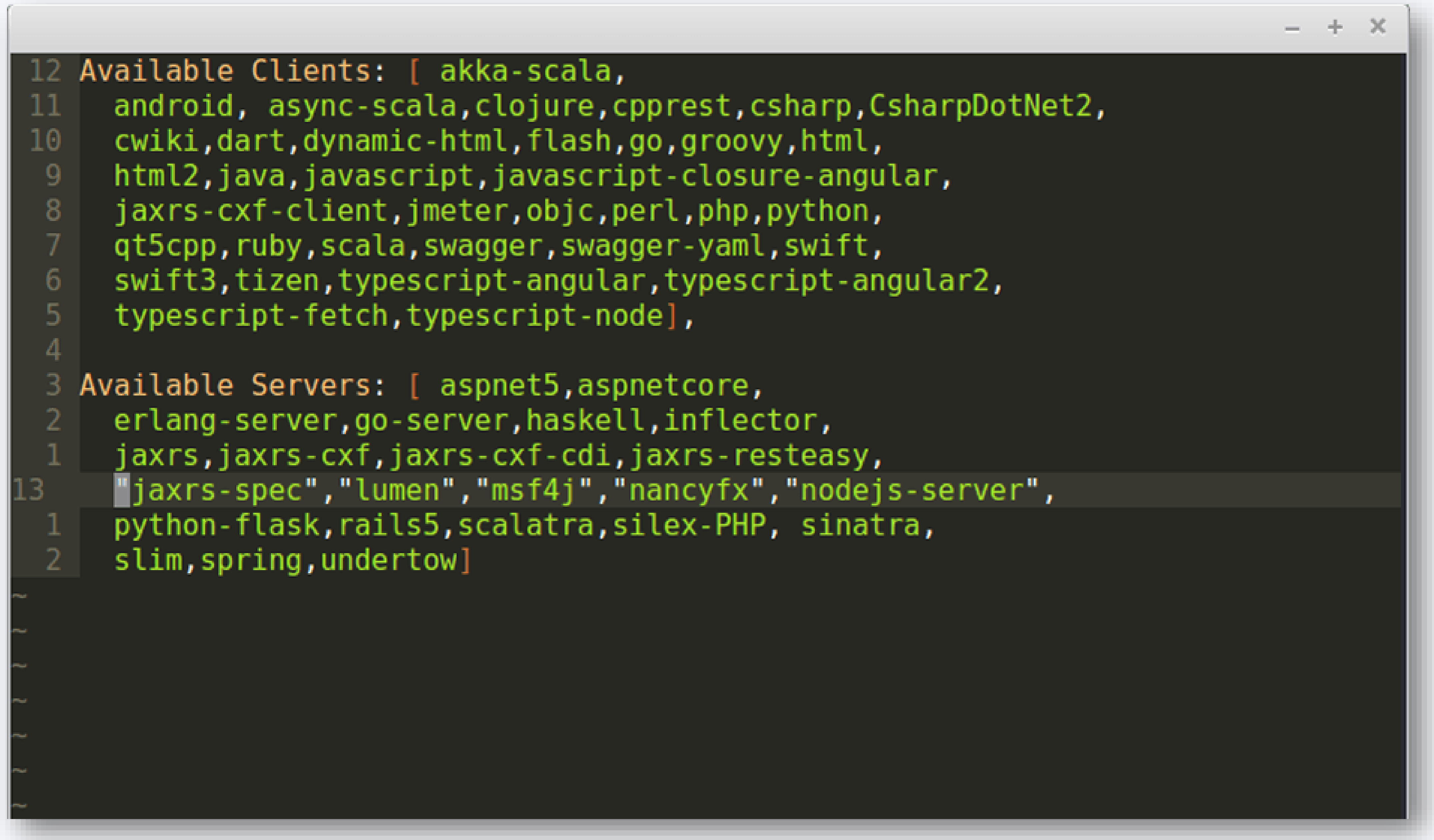
Intelligent
Auto-completion

Visually design your API without coding knowledge

# Swagger Codegen

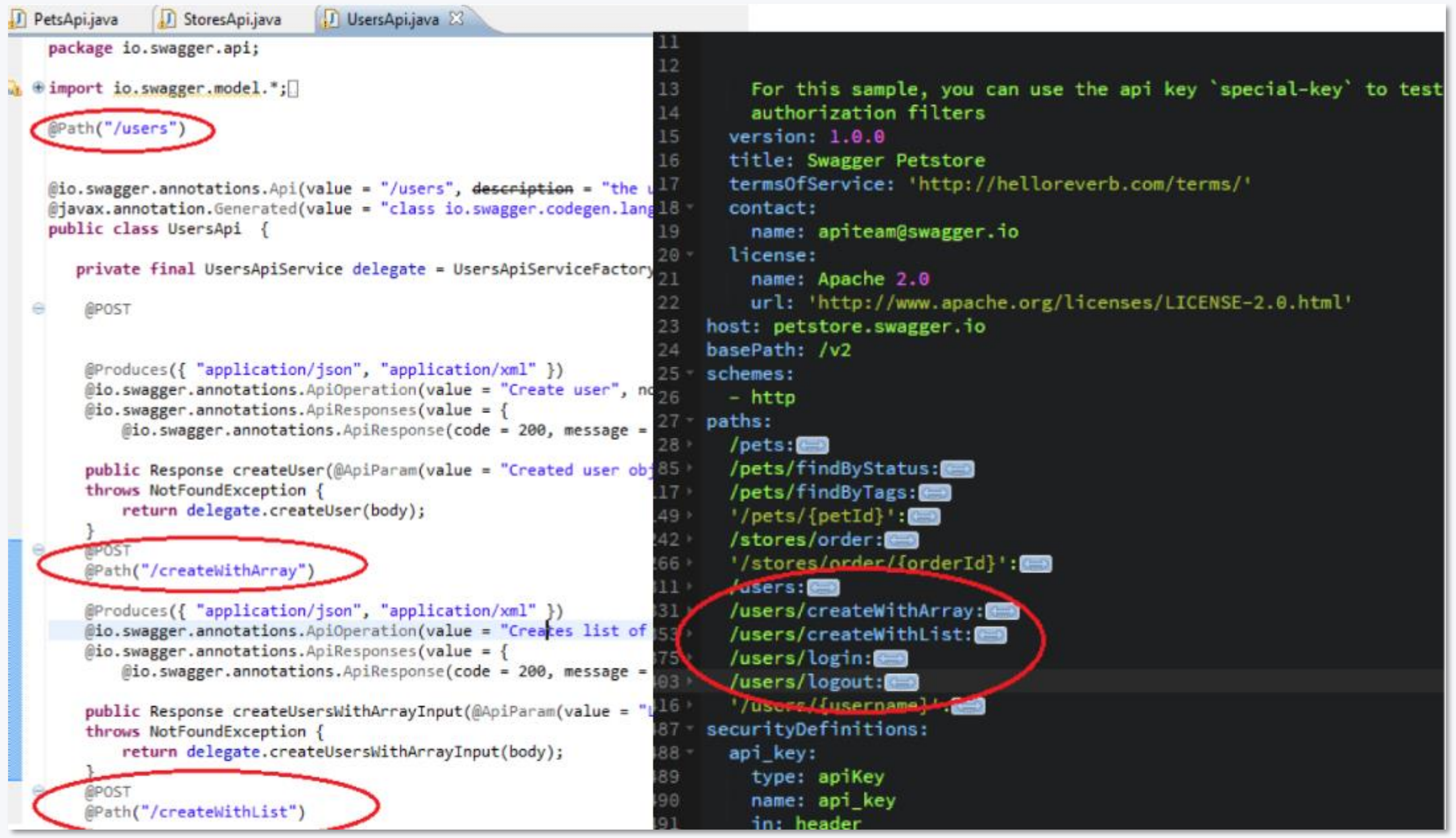
1 Generate **Client** SDKs

2 Generate **Servers**



```
Available Clients: [ akka-scala,
  android, async-scala,clojure,cpprest,csharp,CsharpDotNet2,
  cwiki,dart,dynamic-html,flash,go,groovy,html,
  html2,java,javascript,javascript-closure-angular,
  jaxrs-cxf-client,jmeter,objc,perl,php,python,
  qt5cpp,ruby,scala,swagger,swagger-yaml,swift,
  swift3,tizen,typescript-angular,typescript-angular2,
  typescript-fetch,typescript-node],

Available Servers: [ aspnet5,aspnetcore,
  erlang-server,go-server,haskell,inflector,
  jaxrs,jaxrs-cxf,jaxrs-cxf-cdi,jaxrs-resteasy,
  "jaxrs-spec","lumen","msf4j","nancyfx","nodejs-server",
  python-flask,rails5,scalatra,silex-PHP, sinatra,
  slim,spring,undertow]
~
~
~
~
~
~
~
```



```
package io.swagger.api;

import io.swagger.model.*;

@Path("/users")


@io.swagger.annotations.Api(value = "/users", ~~description~~ = "the u
@javax.annotation.Generated(value = "class io.swagger.codegen.lang
public class UsersApi  {

   private final UsersApiService delegate = UsersApiServiceFactory

    @POST


    @Produces({ "application/json", "application/xml" })
    @io.swagger.annotations.ApiOperation(value = "Create user", nc
    @io.swagger.annotations.ApiResponses(value = {
        @io.swagger.annotations.ApiResponse(code = 200, message =

    public Response createUser(@ApiParam(value = "Created user obj
    throws NotFoundException {
        return delegate.createUser(body);
    }
    @POST
    @Path("/createWithArray")

    @Produces({ "application/json", "application/xml" })
    @io.swagger.annotations.ApiOperation(value = "Creates list of
    @io.swagger.annotations.ApiResponses(value = {
        @io.swagger.annotations.ApiResponse(code = 200, message =

    public Response createUsersWithArrayInput(@ApiParam(value = "L
    throws NotFoundException {
        return delegate.createUsersWithArrayInput(body);
    }
    @POST
    @Path("/createWithList")
```

```
    For this sample, you can use the api key `special-key` to test
    authorization filters
  version: 1.0.0
  title: Swagger Petstore
  termsOfService: 'http://helloreverb.com/terms/'
  contact:
    name: apiteam@swagger.io
  license:
    name: Apache 2.0
    url: 'http://www.apache.org/licenses/LICENSE-2.0.html'
host: petstore.swagger.io
basePath: /v2
schemes:
  - http
paths:
  /pets:
  /pets/findByStatus:
  /pets/findByTags:
  '/pets/{petId}':
  /stores/order:
  '/stores/order/{orderId}':
  /users:
  /users/createWithArray:
  /users/createWithList:
  /users/login:
  /users/logout:
  '/users/{username}':
securityDefinitions:
  api_key:
    type: apiKey
    name: api_key
    in: header
```

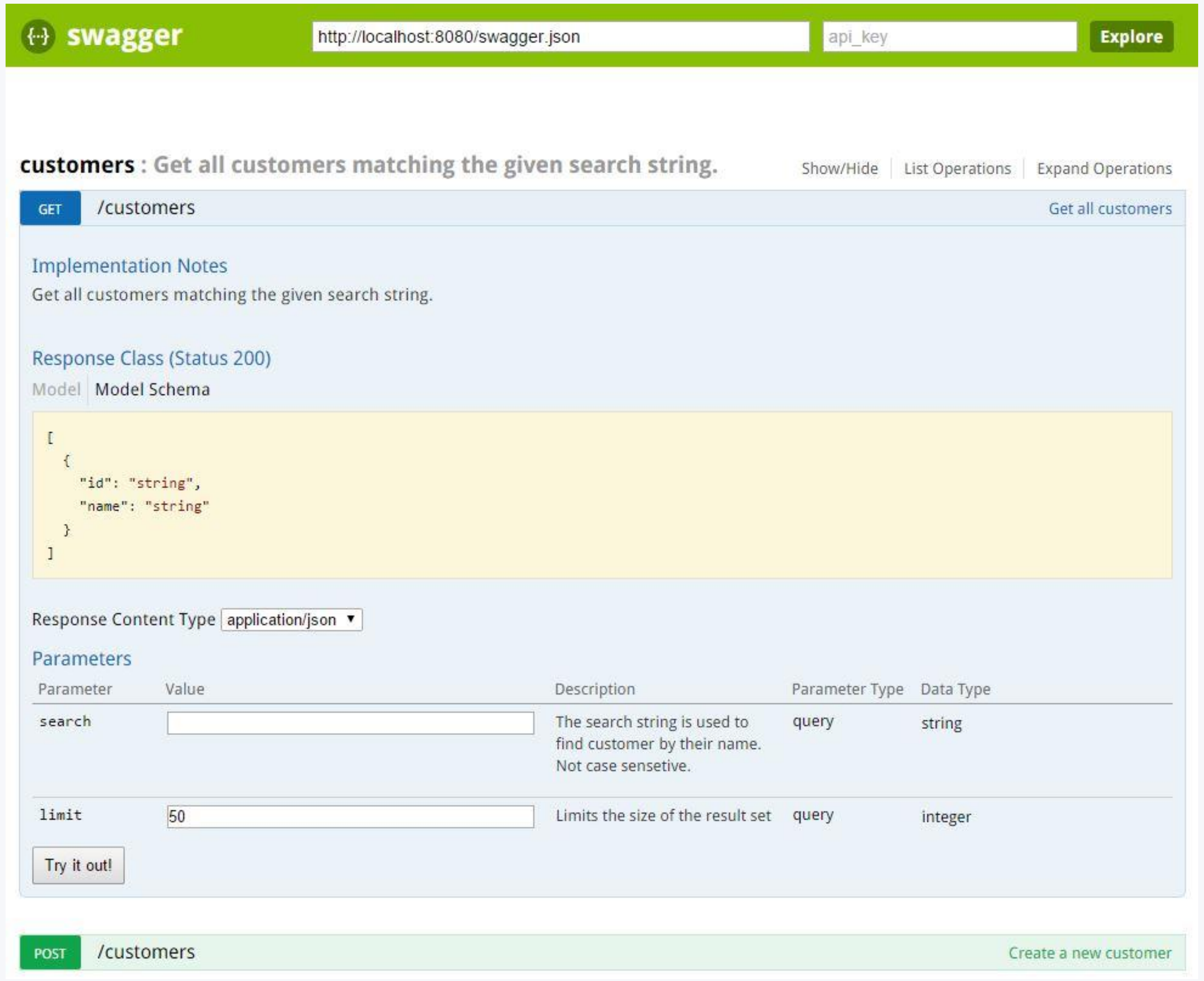
Generating **Server's** and **Client's** code form spec

Technologies

## Swagger UI

Human Friendly

Easy to Navigate



customers : Get all customers matching the given search string.

Implementation Notes

Get all customers matching the given search string.

Response Class (Status 200)

```
[
  {
    "id": "string",
    "name": "string"
  }
]
```

Response Content Type application/json

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|---|---|---|---|---|
| search | | The search string is used to find customer by their name. Not case sensetive. | query | string |
| limit | 50 | Limits the size of the result set | query | integer |

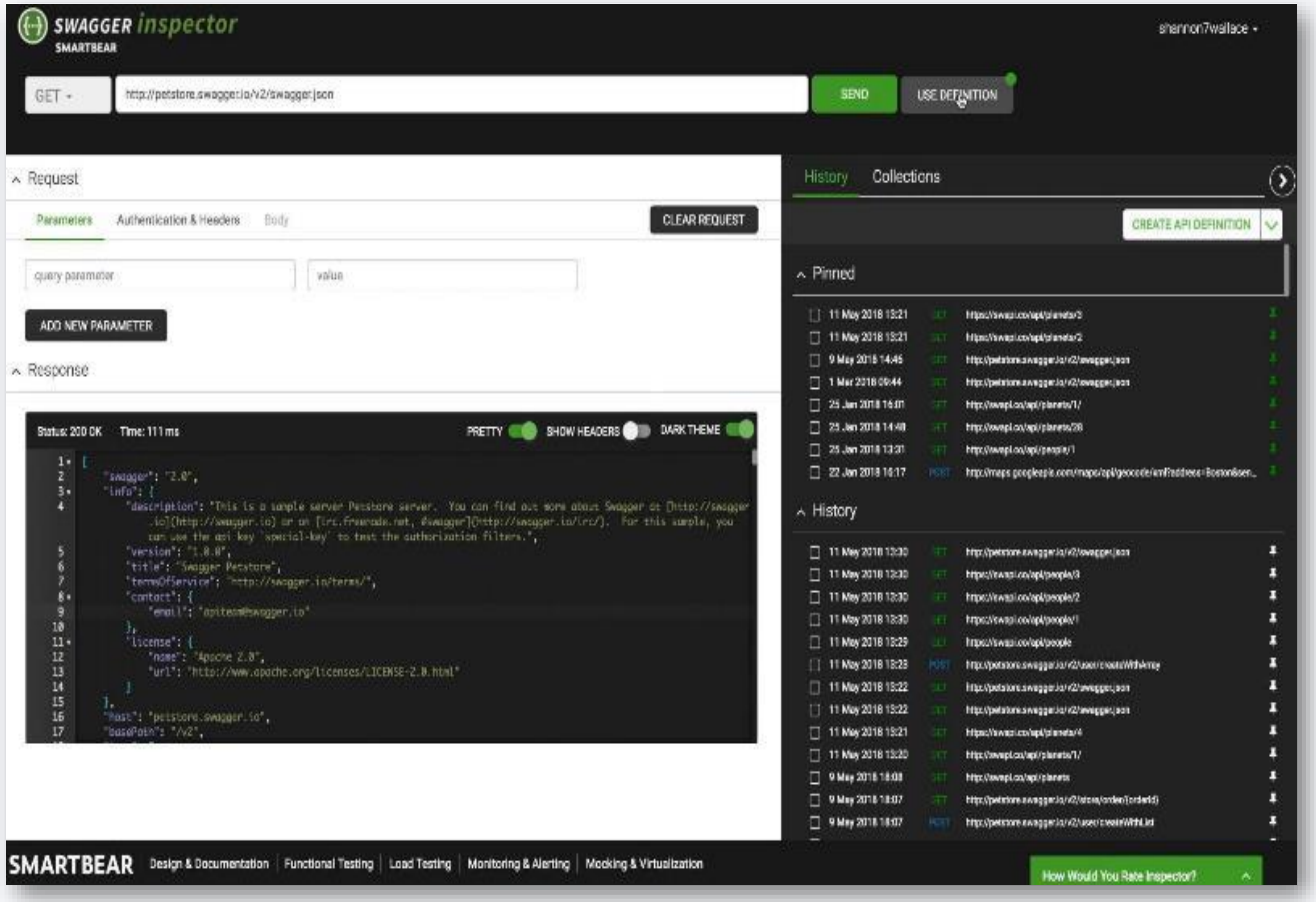
Dependency Free

All Browser Support

Automatically generated from your API Specification

Technologies

## Swagger Inspector



```
http://petstore.swagger.io/v2/swagger.json
```

```
Status: 200 OK    Time: 111 ms
```

```
{
    "swagger": "2.0",
    "info": {
        "description": "This is a sample server Petstore server.  You can find out more about Swagger at [http://swagger
            .io](http://swagger.io) or on [irc.freenode.net, #swagger](http://swagger.io/irc/).  For this sample, you
            can use the api key `special-key` to test the authorization filters.",
        "version": "1.0.0",
        "title": "Swagger Petstore",
        "termsOfService": "http://swagger.io/terms/",
        "contact": {
            "email": "apiteam@swagger.io"
        },
        "license": {
            "name": "Apache 2.0",
            "url": "http://www.apache.org/licenses/LICENSE-2.0.html"
        }
    },
    "host": "petstore.swagger.io",
    "basePath": "/v2",
```

Pinned

```
11 May 2018 13:21   GET    https://swapi.co/api/planets/3
11 May 2018 13:21   GET    https://swapi.co/api/planets/2
9 May 2018 14:46    GET    http://petstore.swagger.io/v2/swagger.json
1 Mar 2018 09:44    GET    http://petstore.swagger.io/v2/swagger.json
25 Jan 2018 16:01   GET    http://swapi.co/api/planets/1/
25 Jan 2018 14:48   GET    http://swapi.co/api/planets/28
25 Jan 2018 13:31   GET    http://swapi.co/api/people/1
22 Jan 2018 16:17   POST   http://maps.googleapis.com/maps/api/geocode/xml?address=Boston&sen...
```

History

```
11 May 2018 13:30   GET    http://petstore.swagger.io/v2/swagger.json
11 May 2018 13:30   GET    https://swapi.co/api/people/3
11 May 2018 13:30   GET    https://swapi.co/api/people/2
11 May 2018 13:30   GET    https://swapi.co/api/people/1
11 May 2018 13:29   GET    https://swapi.co/api/people
11 May 2018 13:23   POST   http://petstore.swagger.io/v2/user/createWithArray
11 May 2018 13:22   GET    http://petstore.swagger.io/v2/swagger.json
11 May 2018 13:22   GET    http://petstore.swagger.io/v2/swagger.json
11 May 2018 13:21   GET    https://swapi.co/api/planets/4
11 May 2018 13:20   GET    http://swapi.co/api/planets/1/
9 May 2018 18:08    GET    http://swapi.co/api/planets
9 May 2018 18:07    GET    http://petstore.swagger.io/v2/store/order/{orderId}
9 May 2018 18:07    POST   http://petstore.swagger.io/v2/user/createWithList
```

1 **Validate** Functionality During Development

2 **Explore** API Capabilities

3 **Automating** API Testing In SoapUI

**Test** without testing your patience