

# The changing API problem

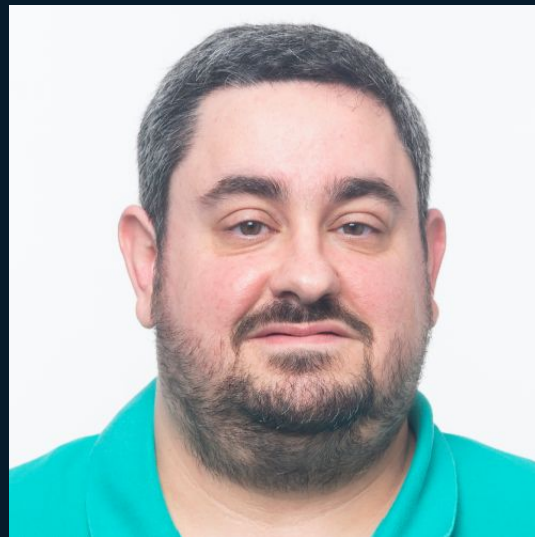
Conf42: JavaScript 2022



# Introduction

# Who I am

- Julio César Abalde Reyes
- Head frontend developer @ Sngular
- More than 19 years in web development
- Leader of frontend at OS<sup>3</sup> initiative.





# The pain

## Time to mock

The lack of API forces us to create our **own mocks**, which are usually **not aligned with the final API**. This increases the efforts of developers at integration time.

Another possibility is to **wait for the API team** to finish their work. But this can make us realize that we have missed something in the design when we include it in the front.

In addition, **we would block the entire front team** while the back team is performing its tasks.



## What's the difference?

Put on your thinking cap and find the subtle differences between these two photographs. As a hint, 7 items have been Photoshopped out of the second image. Answers are below, but don't cheat. Good luck!



Answers: 1. Hoop earring missing from woman wearing the striped blouse. 2. Missing round ball from the umbrella pull. 3. Cutting knife missing from table top. 4. Straw missing from glass. 5. Small amount of red wine in bottom of glass is gone. 6. Flower design in circle on front of woman's blouse is missing. 7. Medalion is missing from the necklace worn by the woman wearing the blue tank top in the back row.

## Find the difference

**Changes can remain hidden for a long time,** even until the time of integration

When this happens, **stress takes over the entire team,** until the change that caused the error is found and the pertinent modifications are made



# The solution

## API First

- **Both client and server side teams “negotiate” the API signature** until they reach an agreement
- **Development teams can work in parallel** both on client and server side, so we eliminate most of the current blocking time
- **Ensures good developer experiences:** well designed, well documented and consistent APIs
- **Reduces the risk of failure** because all teams must honor the contract. Differences appear at development time instead of at integration time





# OpenAPI specification file

Formerly **Swagger Specification**. Is an API description format for REST APIs. An OpenAPI file allows you to **describe your entire API**, including:

- **Available endpoints and methods** on each endpoint (GET, POST, PUT)
- **Operation parameters**
- **Input and output** for each operation
- **Authentication** methods
- Contact information, license, terms of use and other information.

```
openapi: 3.0.2
info:
  title: User Management
  version: "1.0"
servers:
  - url: https://localhost/v1
paths:
  /user:
    get:
      operationId: getAllUser
      responses:
        "200":
          description: OK
          content:
            application/json:
              schema:
                type: array
                items:
                  $ref: "user/components.yml#/components/user"
        "400":
          description: Bad Request
        "500":
          description: Server Error
```

## And there is nothing done already?

There are tools that allow us to use contracts in our development, but almost all of them require JAVA to work. Some examples are:

**Spring Cloud Contracts or PACT**

We don't want front developers to have to install JVM so we decided to make these tools using something they already have installed on their development machines: **NodeJS**.





# The tools

## nodejs-openapi-cli

- Allow us to create “automagically” the **scaffolding on both server and client side**, so that we save a lot of time that we can spend thinking
- Can create server and client code in **javascript and typescript**. Also, give us the possibility of **Angular** services files
- Is already in production as **@corunet/openapi-nodejs-cli** and can be installed from the global npm registry

## openapi-contracts-nodejs-cli

- Will allow us to launch a **mock server** as soon as we “sign” the contract with API team. We can interact with the server so we can check our GET, POST, PUT... requests even if the API team didn't complete them
- Will allow us to **automatically validate** an API server and its alignment with the contract
- This tool is currently **under development**

# Showtime





Thank **you**

twitter: @Julio\_Abalde

email: [julio.abalde@sngular.com](mailto:julio.abalde@sngular.com)

repository: <https://github.com/corunet>



USA | Spain | Mexico | Chile | Singapore  
[www.sngular.com](http://www.sngular.com)