



GlobalLogic[®]

Spring cloud contracts

Vinicius Antonio Gai

<https://www.linkedin.com/in/vinicius-antonio-gai-25361823/>

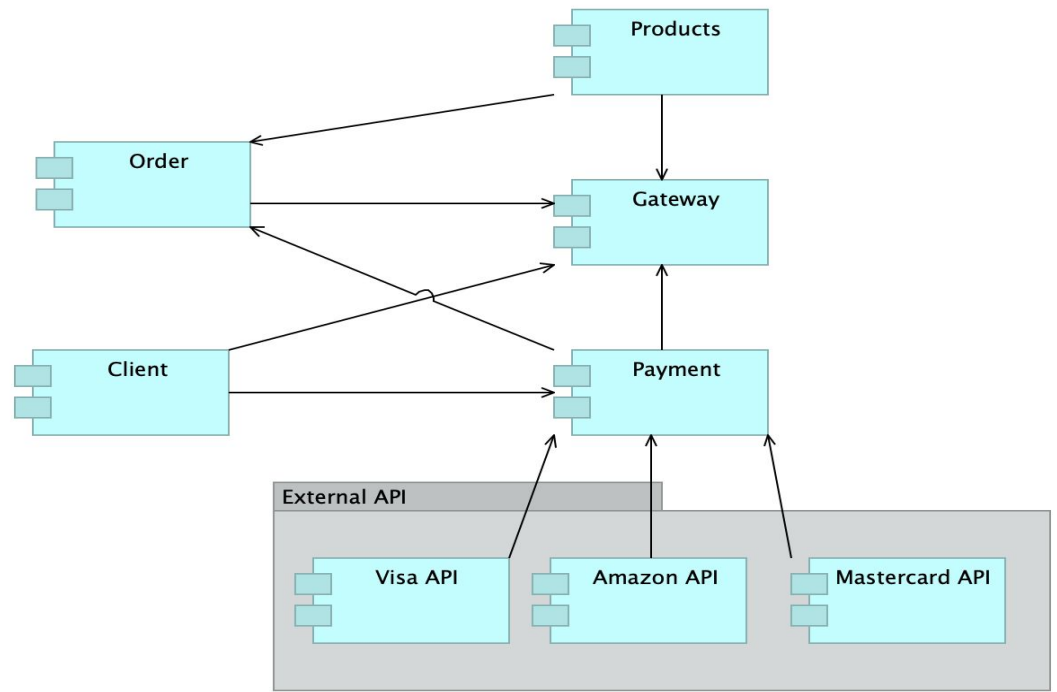
<https://github.com/viniciustoni>

Microservices everywhere

Microservices everywhere

- Microservices - almost everywhere;
- Scalability;
- Fast deploys;
- Isolate the deploy per application;
- Communication between apps:
 - REST Api
 - Streams
 - APIs

Microservices everywhere



Contracts - What is this?

Contracts

- URL
- Request/Response body
 - Structure;
 - Fields name;
 - Fields type;
- Some frameworks that can auto-generate it for us:
 - Sagger;
 - OpenAPI;

I have a contract,
so we are safe!!



Contracts

- Small changes, huge impact;
 - Removing an important field;
 - Changing the field type;
 - Moving things around;
- How to make sure that a simple change did not impact everything?
- Which consumer we will impact with one small change?

So.. how to validate/test it?

Spring cloud contracts

- Consumer drive contract;
- Validate Producer and consumer side;
- Can be written in YAML or Groovy;
- As a producer it'll be executed on verify life-cycle on Maven/gradle;
- As a Consumer it'll create a stub-runner, which will simulate an application;
- It can also be a standalone stub-runner application;
- The idea is to use same contract on producer and on consumer;

A hand holding a smartphone connected to a USB device, with a laptop keyboard visible in the background.

Producer side

Producer side

- Is who is providing the API
- Easy to identify if our changes will impact the consumers;
 - In case yes we could notify them about changes;
- Running as “JUnits” on our controllers;
- Almost an auto generated code for our tests;

Consumer side

Consumer side

- Who is consuming one “external” API
 - External can be another application inside your environment or third party application
- Validation if the contract is still valid as a consumer;
- Helping with tests - simulating rest calls with a stub instead of mocks;

Example - Code

Example - Code

- <https://github.com/viniciustoni/scc-meetup>
- Small app with 2 microservices
- Sales-product
 - Microservice that will contains the APIs for products and stock control
- Sales-order
 - Microservice that will contains the order APIs that will consume sales-products service to reserve products at stock

A person is holding a smartphone, and their hand is visible. The phone screen shows some text. A GlobalLogic logo is overlaid on the top left. The background is a blurred image of a person's face and hand holding the phone.

So.... Why should I use it?

So... Why should I use it?

- Validate your API contract;
- Make sure that your changes on the contract will be “validated” on the consumer side;
- Helping you with Integration tests;
 - Stub-runners instead of mock bean during API calls

Questions



References

- <https://spring.io/projects/spring-cloud-contract>



Thank you