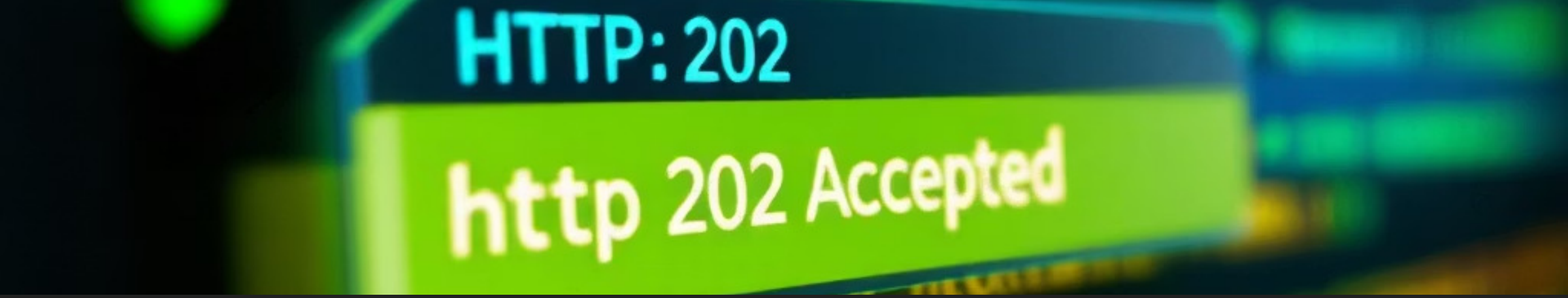


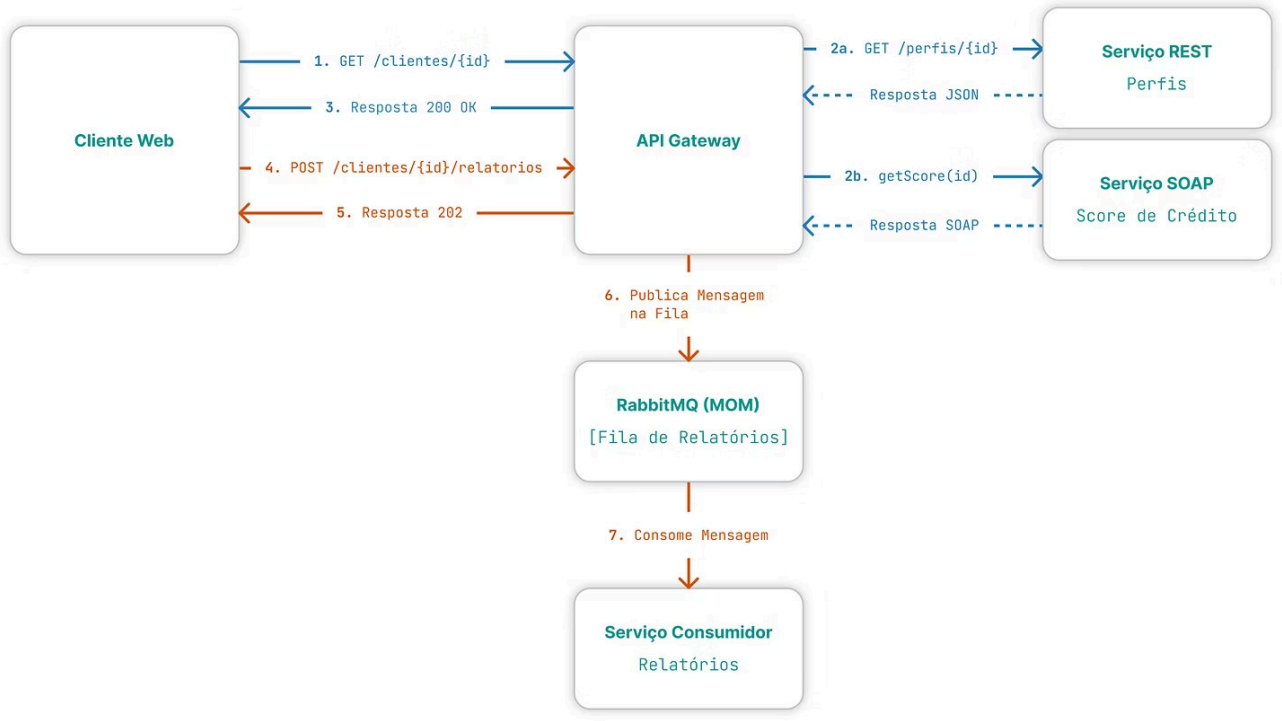

Desenvolvimento de Sistemas Distribuídos

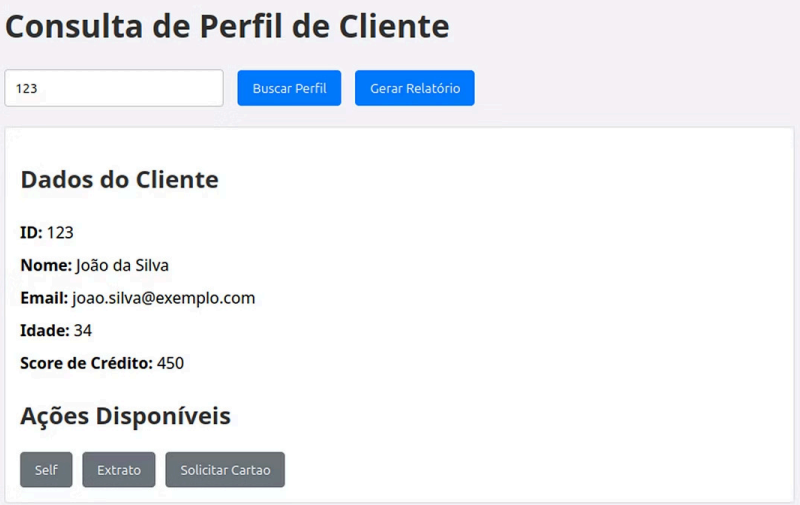
# Arquitetura Híbrida com API Gateway

## Integrando Serviços Síncronos (REST/SOAP) e Assíncronos (MOM)

 por Maycon Alexsander da Silva

# Arquitetura






# Fluxo Síncrono: Consulta Agregada de Dados

## Cliente

Envia **GET** /clientes/{id}

## Gateway

Orquestra chamadas REST e SOAP paralelas

## Serviços

REST (Node.js) e SOAP (Java/JAX-WS)

> As respostas são **agregadas** e retornadas em um único payload JSON com links HATEOAS.

# Fluxo Assíncrono: Processamento de Tarefas Demoradas

1

## Cliente

Envia POST /clientes/{id}/relatorios

2

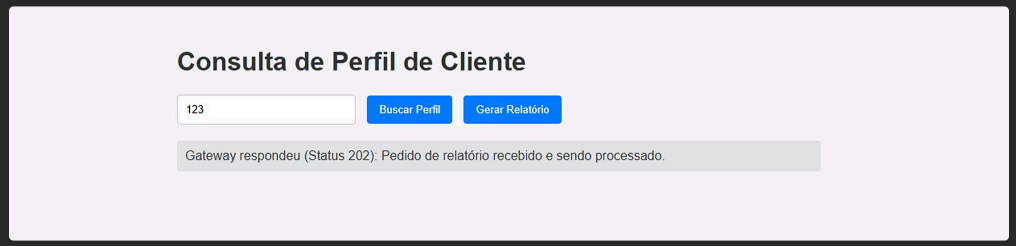
## Gateway (Produtor)

Publica mensagem na fila do RabbitMQ e responde com 202 Accepted

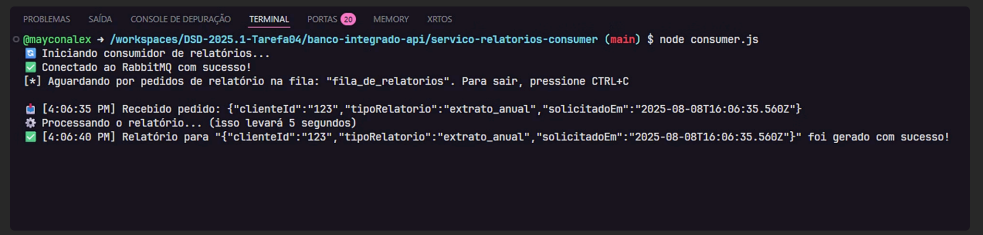
3

## Consumidor

Processa a mensagem da fila no seu próprio tempo




O cliente recebe a confirmação (HTTP 202) **instantaneamente**.




O Consumidor **retira** a mensagem da fila e **executa a tarefa demorada**.

# Ecossistema Tecnológico

### Frontend

HTML5, JavaScript (Fetch API)

### Gateway/Backend

Node.js (Express, Axios, amqplib)

### Serviço Legado

Java (JAX-WS, Maven)

### Mensageria

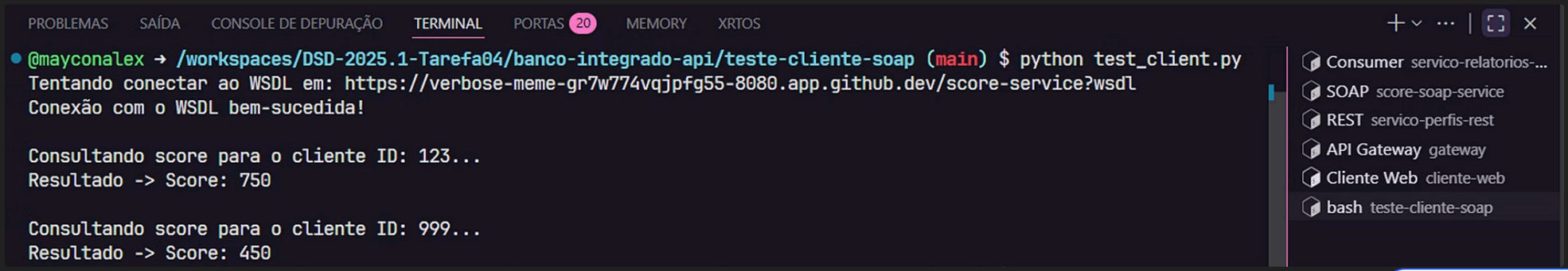
RabbitMQ (via Docker)

### Documentação

OpenAPI 3.0 (Swagger)

## Interoperabilidade SOAP:

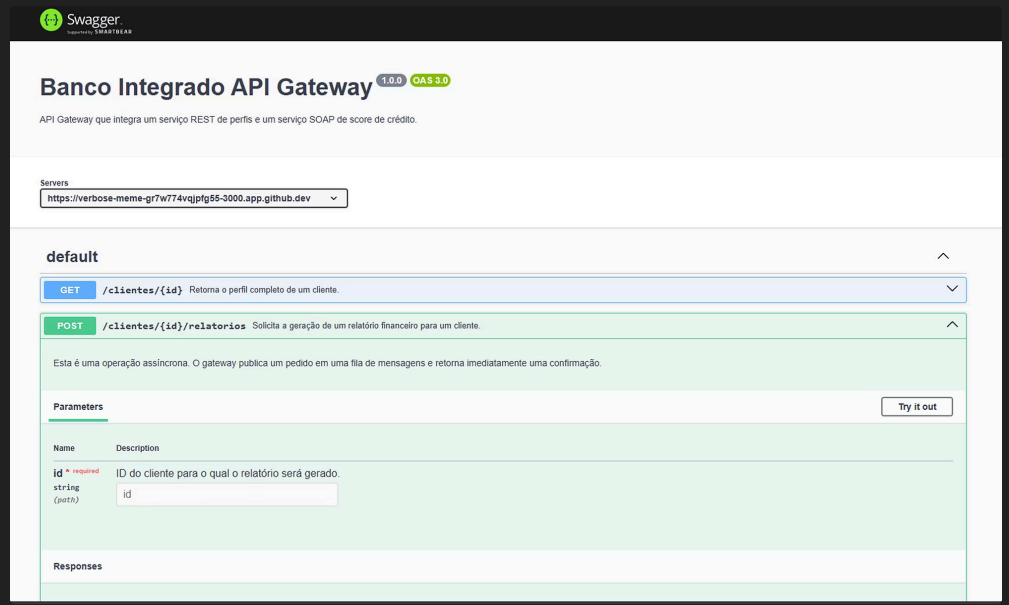
O contrato **WSDL** gerado pelo serviço Java foi consumido por um cliente **Python (Zeep)**.



# Validação da API e Documentação

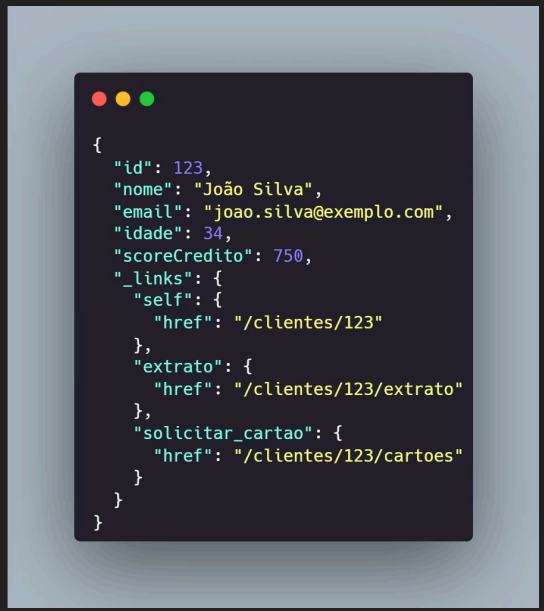
**Objetivo:** Apresentar os artefatos que comprovam a funcionalidade e a qualidade da API.

## Documentação da API:




Interface do **Swagger UI**, mostrando os dois endpoints (GET e POST) com suas descrições. A API é autodocumentada e interativa.

## Resposta HATEOAS:



```
{
  "id": 123,
  "nome": "João Silva",
  "email": "joao.silva@exemplo.com",
  "idade": 34,
  "scoreCredito": 750,
  "_links": {
    "self": {
      "href": "/clientes/123"
    },
    "extrato": {
      "href": "/clientes/123/extrato"
    },
    "solicitar_cartao": {
      "href": "/clientes/123/cartoes"
    }
  }
}
```

Trecho do JSON retornado pelo endpoint GET, destacando o bloco _links. Isso permite a descoberta dinâmica de ações pela API.

# Hands-On!

mayconalex/**DSD-2025.1-Tarefa04**

1 Contributor
0 Issues
0 Stars
0 Forks

GitHub

**GitHub - mayconalex/DSD-2025.1-Tarefa04**

Contribute to mayconalex/DSD-2025.1-Tarefa04 development by creating an account on GitHub.

**Acessar repositório**