

# Arquitetura

A arquitetura escolhida foi **SOFEA “Service Oriented Front End Architecture”**

## Princípios do SOFEA

- Download da aplicação, troca de dados e fluxo de apresentação devem ser desacoplados do servidor
- Fluxo de apresentação é de responsabilidade do cliente
- Todas as comunicações com o servidor devem ser via web services (REST, SOAP, etc.)
- Componentes do servidor devem focar na lógica do negócio e serem expostos em forma de serviços (SOA).

## Benefícios do SOFEA

- Escalabilidade (processing, stateless, caching)
- Interoperabilidade (BaaS - Back-end as a Service)
- Se encaixa em ambientes SOA e Cloud
- Melhor organização e separação da lógica
- Maior responsividade para o usuário
- Desenvolvimento assíncrono do front-end e back-end

## Back End avaliacao-api

O projeto foi desenvolvido em java 8 utilizando framework Spring boot que facilita o processo de configuração e publicação das aplicações. A intenção é ter o seu projeto rodando o mais rápido possível e sem complicação.

Também utilizei Spring Data JPA, trata-se de um framework que nasceu para facilitar a criação dos nossos repositórios. Ele faz isso nos liberando de ter que implementar as interfaces referentes aos nossos repositórios (ou DAOs), e também já deixando pré-implementado algumas funcionalidades como, por exemplo, de ordenação das consultas e de paginação de registros.

## Front End avaliacao-ui

O projeto foi desenvolvido em Angular 4, que é um framework completo, possui muitos componentes disponíveis para diversas necessidades e já está pronto! Além disso temos uma certa segurança que é o Google por trás disto tudo.

Também utilizei alguns componentes do PrimeNg e do Bootstrap para dar visual mais rico e responsivo para a aplicação.

Utilizei Um serviço rest para carregar dinamicamente os estados e municípios:

<http://www.geonames.org/>

## Ides:

Spring Tool Suite: <https://spring.io/tools/sts> (Eclipse com Spring Tools)

Visual Studio Code: <https://code.visualstudio.com/>

## DB:

MySQL: <https://dev.mysql.com/downloads/mysql/>

## Servers:

Spring Boot usa um container embarcado (por padrão é o **Tomcat**) para facilitar o desenvolvimento.

Node: <https://nodejs.org/en/download/>

## Cloud:

<http://avaliacao-join-ui.herokuapp.com/>