

## Avaliação Desenvolvedor Back-end Attornatus

O objetivo deste documento é identificar seus conhecimentos quanto às tecnologias utilizadas no cotidiano de desenvolvimento da equipe de Back-end na Attornatus Procuradoria Digital.

Esta análise propõe avaliar os seguintes temas:

- Qualidade de código
- Java, Spring boot
- API REST
- Testes

A entrega deverá ser feita da seguinte forma:

- O prazo para entrega da avaliação será de até 7 dias após envio da mesma
- Encaminhar este documento com as perguntas respondidas e com o link do código público em sua conta do GitHub
- Opcionalmente, caso você consiga fazer o build da aplicação, poderá também informar o link de acesso

### Qualidade de código

#### 1. Durante a implementação de uma nova funcionalidade de software solicitada, quais critérios você avalia e implementa para garantia de qualidade de software?

Eu avalio serem várias etapas para definir uma qualidade no software ou na nova funcionalidade, mas todas essas etapas começam com a definição do processo de software, logo mais a definição de requisitos, nesse caso o MPS.BR (Melhoria do Processo de Software Brasileiro) que avalia níveis de qualidade de processo de software diz que essa parte deverá ser com o cliente assim como a sua aprovação feito em um documento como ata de reunião.

Em Seguida, segue pela etapa planejamento do produto já que os requisitos foram aprovados pelo cliente com seus artefatos, com isso inicia a criação de artefatos antes da codificação, pois essa etapa é altamente importante para a implementação do produto, devido à sua ajuda no evitamento de erros e retrabalhos na próxima etapa, como os "*Wireframes*" que podem ser feitos durante as etapas dos requisitos e nessa fase se torna o principal artefato do "*Front-end*", contudo no "*Back-end*" são os diagramas de classe e o de entidade relacionamento, porém o principal é o documento de arquitetura de software fornecendo um modelo de arquitetura proposta e tendo o mapeamento do produto por completo, sua finalidade assim como os outros diagramas são para o bom entendimento do produto na fase de construção do dele.

Em adiante tem etapa principal e mais conhecida, que parte de “mão na massa” é a codificação do produto, nessa etapa é boa para avaliar a otimização do

produto em quanto tempo ele consegue entregar os seus serviços, assim como a aplicação de testes automatizados e revisão de código para garantir que novas funcionalidades tenham qualidade nessa etapa, como também é crucial a segurança, escalabilidade e fácil manutenibilidade do código durante a execução dela.

E por fim a última fase, que de manutenção do código, que surge para verificar a usabilidade do produto, nela é lançado a primeira versão do produto para teste, nela conseguirá identificar a possibilidade bugs ou falhas que podem que estão presentes ou pode surgir após adição de novas funcionalidades, também se torna crucial que a equipe de manutenção tenha sempre o feedback dos usuários, para garantir a adição de novas funcionalidades para garantir o ciclo de vida do produto.

Seguindo por todas essas etapas, acredito que o produto que se desenvolva ou uma nova funcionalidade a ser adicionada terá uma grande qualidade e assim como o seguimento das normas que garante isso.

## **2. Em qual etapa da implementação você considera a qualidade de software?**

Acredito que todas as etapas são necessárias para garantir a qualidade de software, sendo que cada uma tem a sua função no ciclo de vida dele, desde elicitar requisitos e garantir que são completos e entendíveis, seguindo na fase de planejamento que nela será garantida as criações de artefatos que serão de extrema importância para um bom entendimento para a construção do software que é a próxima etapa, nela se garante a qualidade do produto por testes, revisão e seguir boas práticas e por fim na última etapa que é da manutenção nela se baseia a qualidade pela correção do código sendo bugs ou falhas e incorporação de feedback dos usuários que garante novas funcionalidades para o produto garantindo seu ciclo de vida.

Link do repositório do GitHub:

<https://github.com/samuelfilho-dev/api-pessoa>

## Desafio Java

Usando Spring boot, crie uma API simples para gerenciar Pessoas. Esta API deve permitir:

- Criar uma pessoa
- Editar uma pessoa
- Consultar uma pessoa
- Listar pessoas
- Criar endereço para pessoa
- Listar endereços da pessoa
- Poder informar qual endereço é o principal da pessoa

Uma Pessoa deve ter os seguintes campos:

- Nome
- Data de nascimento
- Endereço:
  - Logradouro
  - CEP
  - Número
  - Cidade

Requisitos

- Todas as respostas da API devem ser JSON
- Banco de dados H2

Diferencial

- Testes
- Clean Code

Será levado em avaliação

- Estrutura, arquitetura e organização do projeto
- Boas práticas de programação
- Alcance dos objetivos propostos.