***Avaliação Técnica do Inatel 2 – OPEN SCOPE***

Versão 3.0 Página **1**

Este documento descreve a **segunda avaliação técnica** para os estagiários do IDP. O principal objetivo é avaliar as habilidades de criatividade e adaptabilidade, mantendo o escopo aberto. As orientações gerais serão fornecidas e o estagiário deve escolher um tema e desenvolvê-lo.

**1. Softwares:**

Os seguintes softwares são obrigatórios para serem utilizados durante o desenvolvimento da aplicação:

* • Kit de Desenvolvimento Java 11
* • Maven (última versão)
* • Docker Engine, WSL 2 e Ubuntu 20.04

Nota: O *Docker Desktop para Windows* deve ser evitado devido a restrições de licença.

Como ferramentas de desenvolvimento, você pode usar as mesmas usadas no primeiro projeto de avaliação.

**2.Orientações**

As seguintes orientações ajudarão o estagiário a desenvolver o seu projeto:

❖Procure projetos interessantes e escolha um. O projeto deve integrar-se a outros sistemas on-line;

❖O projeto consiste em um aplicativo back-end que fornece endpoints REST para seus clientes e consome endpoints remotos do projeto on-line existente;

❖Escrever uma documentação resumida (fornecida em README.md) descrevendo os requisitos funcionais em alto nível;

❖Evite um grande número de endpoints. Cerca de 4 e 6 endpoints já são suficientes;

❖Utilizar a plataforma Spring Boot e *Spring Initalizr* (start.spring.io) para configuração inicial;

❖O projeto deve ter a identificação do **estagiário** como prefixo. Ex: VitorFigueiredo\_MySolutionProject

❖Utilizar conceito de microsserviços baseado em conteinerizações. Uma sugestão é usar 2 contêineres diferentes: a) um contêiner para o aplicativo Spring Boot; b) outro contêiner para banco de dados MySQL;

❖

❖Use docker-compose para integrar os contêineres;

❖Você deve fornecer a documentação da API seguindo a especificação Swagger e OpenAPI 3;

❖Todos os métodos de serviço devem ter os respectivos testes unitários. Testes de sucesso e fracasso devem ser contemplados;

❖Postman pode ser usado para testes funcionais manuais durante a fase de desenvolvimento (entregar scripts automatizados para Postman é opcional). Para a demonstração, espera-se que haja alguns testes funcionais automatizados do endpoint usando qualquer estrutura de teste conhecida (por exemplo, Cucumber, Karate, Rest Assured ). Cenários positivos e negativos devem ser explorados e as asserções devem ser feitas no código de status da resposta, no cabeçalho (se houver) e na carga útil.

***Avaliação Técnica do Inatel 2 – OPEN SCOPE***

Versão 3.0 Página **2**

**3. Dicas importantes**

* • Na primeira fase do projeto, o H2 pode ser usado como banco de dados de arquivos local e não há necessidade de configuração de contêiner. Mas, na fase final, você deve configurar os arquivos de contêiner e alternar para o contêiner de banco de dados MySQL;
* • Trabalhe de forma incremental: Primeiro, faça o **diagrama de classes UML**, depois construa **classes de entidade**, depois a camada **de repositório**, depois a camada de serviço e, finalmente, a **camada de**  **controlador**. Lembre-se de projetar as **classes DTO**;
* • À medida que você desenvolve cada camada, escreva os testes de unidade e corrija as falhas à medida que forem encontradas;
* • Lembre-se de documentar as regras de negócio;
* • Você é encorajado a propor algoritmos, classes de utilidade ou arquiteturas "prontas para uso", desde que representem contribuições reais para a solução final;
* • A documentação da API não precisa permitir que o usuário envie solicitações e receba respostas.

**Produtos**

O projeto final deve ser entregue como um link de repositório do GitHub. O repositório deve ter os seguintes itens:

* • O projeto deve ser estruturado como projeto Maven e assim poderá ser compilado, testado, construído e empacotado por linha de comando (mvnw);
* • Os testes unitários abrangem todos os métodos e classes de negócio;
* • O arquivo README.md deve ter as instruções para executar o aplicativo com todos os contêineres;
* • A documentação da API baseada no Swagger e seguindo a Especificação OpenAPI 3. Exporte a documentação para um arquivo PDF. Crie a pasta de documentos e coloque todos os arquivos de documentação, incluindo o arquivo PDF exportado;
* • Criar a pasta de modelagem e colocar todas as imagens exportadas de diagramas de classe UML;
* • Criar a pasta de apresentação e colocar a apresentação PowerPoint do respectivo projeto.

*# Excelente projeto #*