**PROVA 1 – DESENVOLVIMENTO WEB**

N**ome: LUCAS DE PAULA GUIMARÃES Valor: 7 Nota: \_\_\_\_\_\_\_\_\_\_**

# QUESTÃO 1 – PRÁTICA (VALOR 5 PONTOS)

Você foi contrato por uma empresa de software para participar de um projeto. Nesse projeto você ficou responsável pela parte de backend.

O projeto consiste em criar um sistema para registro de doações. Sua API deve conter os seguintes métodos:

a) Método POST para enviar uma doação com os dados: valor doação, nome da instituição, CPF do doador, Nome do doador.

Sua API de conter as seguintes regras:

* Valores possíveis de doação: 10 reais, 20 reais, 50 reais e acima de 100 reais pode ser qualquer quantia.
* Nome instituição, nome do doador e CPF obrigatórios.
* CPF deve ser válido utilize o algoritmo que usamos em aula

O retorno da sua API deve conter nos casos de erro a mensagens de retornos a seguir:

Sendo:

# Mensagem: Contexto

“CPF Inválido” Para as situações que o cpf não for válido “Valor inválido de doação” Para as situações que o valor não for

satisfeito

“Campo nome obrigatório” Para campo nome vazio

“Campo nome instituição obrigatório” Para campo nome instituição vazio

“Campo cpf obrigatório” Para campo cpf vazio

“Obrigado pela sua

b) Para os casos de sucesso na doação (sem erro) sua API deve retornar a seguinte mensagem e brindes de acordo com sua doação.

{

“mensagem” : “Obrigado pela sua doação!”, “brinde” : “x”. }

Onde o brinde depende das regras:

Valor Doação Brinde

10 reais Você ganhou um vale desconto de 10% nas lojas xpto. Código do brinde de desconto é YYY

20 reais Você ganhou um vale desconto de 20% nas lojas xpto. Código do brinde de desconto é YYY.

50 reais Você ganhou um vale desconto de 30% nas lojas xpto. Código do brinde de desconto é YYY.

Acima 100 reais Você ganhou um vale desconto de 50% nas lojas xpto. Código do brinde de desconto é YYY.

Onde YYY é: um número aleatório que você vai gerar entre 1.000 e 9.999 + (concatenado) cpf + outro número aleatório que você vai gerar entre 10.000 e 99.999.

# CRITÉRIOS DE AVALIAÇÃO

|  |  |  |
| --- | --- | --- |
| **Critério** | **Valor** | **Nota** |
| Rota | 0,5 |  |
| Validações | 0,5 |  |
| Regra de negócio do valor de doação | 1 |  |
| Geração do brinde | 1 |  |
| Organização do projeto em pastas | 0,5 |  |
| Clean Code: nomes de variáveis, funções e organização de código (identado) | 0,5 |  |
| Teste no insominia funcionando (professor) | 1 |  |
| Total | 5 |  |

# QUESTÃO 2 – TEÓRICA (VALOR 2 PONTOS)

Sobre as tecnologias que utilizamos em sala e nos nossos projetos descreva o papel de cada uma no projeto da TDE utilizado destacando em qual parte você utilizou no projeto e as vantagens dessas escolhas.

Git Hub, Nodejs, visual studio code, mongodb, atlas, jest, git hub actions, heroku.

GITHUB: Ideal para postagens de projetos caso você queira um histórico completo de tudo que você já tenha feito em sua jornada, seria basicamente uma rede social de profissionais da tecnologia, com ele foi possível fazer o upload de todo o meu projeto final.

NODEJS: Se trata de um ambiente open-source e multiplataforma que utilizamos para criar nossa API, em JavaScript. Tem como vantagem o desenvolvimento mais rápido devido ao grande número de módulos e código reutilizáveis que se pode obter pelo meio do npm.

VISUAL STUDIO CODE: Principal plataforma para editarmos nosso código fonte, nele contém toda a estrutura do meu projeto.

MONGODB: É um banco de dados opensource, de alta performance e flexível, sendo considerado o principal banco de dados NoSQL. Como a própria introdução diz, nele este nosso banco de dados referente ao projeto.

GITHUB ACTIONS: É basicamente um orquestrador para o workflow. Através dele nós podemos construir um workflow com várias ações que vão descrever os passos necessários para compilar, testar, etc. Ele é essencial para a compilação desse projeto.

HEROKU: É uma plataforma nuvem que faz deploy de várias aplicações back-end seja para hospedagem, testes em produção ou escalar as suas aplicações.

ATLAS: serviço em nuvem do MongoDB que utilizamos para conectar com o nosso BD.

JEST: framework que utilizamos para fazer testes unitários em nossa API.