

## Teste Técnico

Para que possamos avaliar o seu desempenho técnico, desenvolva o teste descrito neste documento da melhor maneira possível, aplicando técnicas de POO, MVC, Clean Code, Multi-Tier e utilizando ao máximo o seu potencial.

### 1. Criar banco de dados com estrutura do arquivo abaixo (preferencialmente em PostgreSQL)

```
CREATE TABLE pessoa (  
    idpessoa bigserial NOT NULL,  
    flnatureza int2 NOT NULL,  
    dsdocumento varchar(20) NOT NULL,  
    nmprimeiro varchar(100) NOT NULL,  
    nmsegundo varchar(100) NOT NULL,  
    dtregistro date NULL,  
    CONSTRAINT pessoa_pk PRIMARY KEY (idpessoa)  
);  
  
CREATE TABLE endereco (  
    idendereco bigserial NOT NULL,  
    idpessoa int8 NOT NULL,  
    dscep varchar(15) NULL,  
    CONSTRAINT endereco_pk PRIMARY KEY (idendereco),  
    CONSTRAINT endereco_fk_pessoa FOREIGN KEY (idpessoa) REFERENCES pessoa(idpessoa) ON DELETE  
cascade  
);  
CREATE INDEX endereco_idpessoa ON endereco (idpessoa);  
  
CREATE TABLE endereco_integracao (  
    idendereco bigint NOT null,  
    dsuf varchar(50) NULL,  
    nmcidade varchar(100) NULL,  
    nmbairro varchar(50) NULL,  
    nmlogradouro varchar(100) NULL,  
    dscomplemento varchar(100) NULL,  
    CONSTRAINT enderecointegracao_pk PRIMARY KEY (idendereco),  
    CONSTRAINT enderecointegracao_fk_endereco FOREIGN KEY (idendereco) REFERENCES  
endereco(idendereco) ON DELETE cascade  
);
```

### 2. Definir arquitetura do sistema em três camadas

- Comunicação Rest com JSON entre aplicação Cliente / Servidor
- Aplicar Clean Code
- Orientação a objetos
- Padrões de projeto
- Garantir integridade entre registros (não ter pessoa sem endereço)
- Camada de persistência, utilizar Firedac

### **3. Desenvolver um cadastro de pessoas**

- Objetivo é fazer cadastro simplificado com os dados da pessoa e o CEP (no item 4 a tabela *endereco\_integracao* será atualizada com base no CEP informado)
- Tabelas
  - Pessoa e Endereco
- Métodos
  - Insert
  - Update
  - Delete
  - Insert em lote (novo método): recebe uma lista de pessoas (considerando que essa lista poderá ter 50.000 registros. Adotar uma estratégia para que a inserção desses registros seja performática)

### **4. Desenvolver nova rotina utilizando Threads**

- Objetivo é atualizar os endereços das pessoas cadastradas no item 3
- Para cada registro da tabela endereco, ler campo CEP e fazer a integração com a “API via cep” através da URL [viacep.com.br/ws/\\_numero\\_CEP/json/](http://viacep.com.br/ws/_numero_CEP/json/)
  - Utilizar campo CEP da tabela endereco
- Atualizar os campos da tabela *endereco\_integracao* com os dados do JSON de retorno

### **Critérios de Avaliação**

1. Utilize PostgreSQL como banco de dados
2. Utilize FireDAC para conexão ao banco de dados
3. Utilize conceitos de transação e tratamento de exceções na gravação dos dados
4. Capriche na escrita do seu código, pois, a formatação está sendo avaliada
5. Utilize conceitos de orientação a objetos, criando classes por exemplo
6. Não utilize componentes de terceiros, use sempre o que é nativo da IDE
7. Publique seu teste no GitHub, ou em outro repositório de código, e deixe o repositório público, enviando o link para o departamento de RH da WK Technology