**Requisito de Software: Avaliação técnica de carga.**

1. **Descrição Geral**

Este documento especifica os requisitos para o desenvolvimento de um script que será responsável por importar dados de uma planilha e realizar a carga desses dados em um banco de dados.

1. **Escopo**

* Desenvolver um script em Python que seja capaz de.
  + Ler uma planilha excel (.xlsx).
  + Validar e transformar os dados antes de carregá-los no banco de dados.
  + Carregar os dados lidos para tabelas em um banco de dados PostsgreSQL.
  + Gerar logs de erros e logs de sucesso após a execução.

1. **Requisitos Funcionais**
   * **Importação de Planilha**

* **RFI001:** O script deverá ser capaz de importar todas as colunas da planilha **sales\_data\_with\_dates.xlsx** disponibilizada juntamente com esse documento.
* **RF002**: O script deverá validar o formato e a estrutura da planilha antes da importação.
* **RF003**: O script deverá converter os dados para formatos implícitos na planilha como datas e decimais. Registros fora do padrão deverão ser ignorados.
  + **Carga no Banco de Dados**
* **RF004**: Seja no script python ou em um arquivo de configuração separado deverá ser possível informar os parâmetros de conexão como host, porta, usuário, senha e nome do banco (**Atenção para o RNF003**).
* **RF007**: O script python deverá realizar a inserção dos dados importados para uma tabela do banco de dados.
* **RF008**: O script python deverá mapear as colunas da planilha para as colunas da tabela no banco de dados.
  + **Validação e Log**
* **RF009**: O script deve validar os dados da planilha quanto a valores nulos, tipos de dados e valores duplicados.
* **RF010**: O script deve gerar um log de erros contendo as linhas e colunas com problemas encontrados durante a validação.
* **RF011**: O script deve gerar um log de sucesso com a quantidade de registros importados e inseridos no banco de dados.
* **RF012**: A decisão sobre a melhor saída do log fica a cargo do desenvolvedor, podendo ser arquivo, tela, tabela de banco etc.

1. **Requisitos Não Funcionais**
   * **Duplicidade dos dados**
     + **RNF001**: Os dados disponibilizados no banco deverão refletir os mesmos dados da planilha, exceto dados incorretos que não deverão ser importados. O script deverá ter mecanismos para ser executado várias vezes seguidas sem que essa regra seja quebrada, isto é, sem duplicar os dados.
   * **Desempenho**

* **RNF002**: O script python deve ser capaz de processar a planilha **sales\_data\_with\_dates.xlsx**, que contém 1500 registros, em menos de 1 minuto.
  + **Segurança**
* **RNF003**: O script deve garantir a segurança das credenciais do banco de dados, não expondo essas informações em logs, mensagens de erros e nem as disponibilizando no código versionado.

1. **Requisitos de Implementação**

* **RI001**: O script deve ser desenvolvido em **Python (3.10 ou superior)**, utilizando bibliotecas como **Pandas** para manipulação de dados e **Psycopg2** para interações com o banco de dados.
* **RI002**: Todas as dependências necessárias devem estar disponíveis para instalação pelo **pip** e listadas no arquivo **requirements.txt**.
* **RI003**: Deve ser disponibilizado todos os scripts necessários para a criação dos objetos de banco de dados como tabelas, views etc. compatíveis com Postgress 16.
* **RI004**: Instruções de implantação adicionais deverão ser devidamente documentadas no README.md.

1. **Critérios de Aceitação**

* Para validação será executado o script .sql de implantação de banco em uma instância local na máquina do validador, seus dados de conexão serão informados no local indicado, posteriormente o script principal será executado pelo comando python.
* A execução do script deverá ocorrer sem erros não tratados.
* Será validado se todos os registros foram importados com sucesso no banco de dados.

1. **Entregáveis**
   * Ao menos um arquivo de script python (.py). Se houver mais de um então o primeiro a ser executado deverá ser identificado.
   * Ao menos um arquivo de deploy de banco (.sql).
   * Um arquivo com lista das dependências pip (requirements.txt).
   * Se for necessário, um documento com identificação do projeto e instruções adicionais que o desenvolvedor achar relevante (preferencialmente no README.md).
   * **Método de entrega:** Disponibilização da url do projeto no github.
2. **Prazos**

* **Data de Entrega**: 30/08/2024

1. **Responsáveis**
   * **Revisor e Aprovador**: Diego Bittencourt