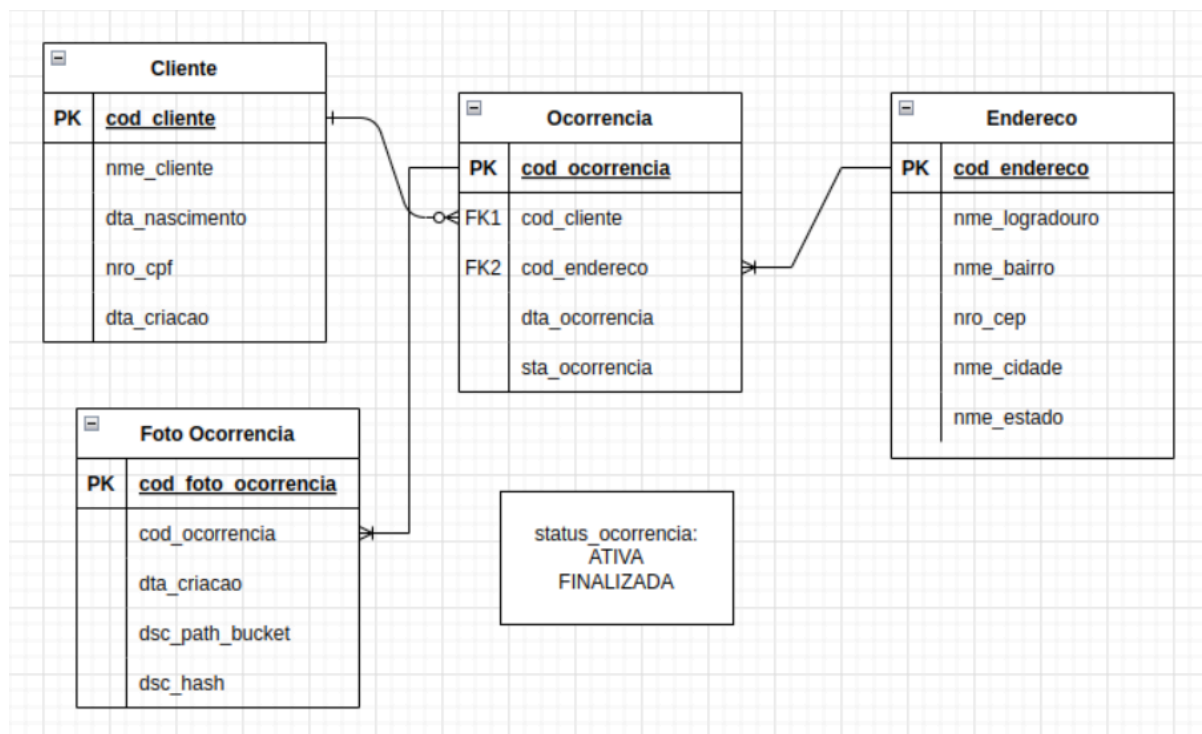


Avaliação de Capacidades Técnicas

Neste projeto você deverá implementar uma API REST disponibilizando o artefato final conforme orientações deste documento. A solução deverá basear-se no seguinte diagrama de banco de dados:

DIAGRAMA



Pré-requisitos:

Antes de iniciar a realização do teste prático realize a leitura deste documento até o final.

1. Utilizar linguagem de programação Java 11+ com framework Spring Boot
2. Executar em container um servidor Min.io para armazenamento de objetos S3 (<https://min.io/>);
3. Executar em container um servidor de banco de dados PostgreSQL em sua última versão;

Requisitos Gerais:

1. Implementar mecanismo de autorização e autenticação
2. A solução de autenticação deverá expirar a cada 30 minutos
3. Conter recursos de paginação em todas as consultas
4. Os dados produzidos deverão ser armazenados no servidor de banco de dados previamente criado em container
5. Armazenar arquivos e imagens no servidor de armazenamento de objetos Min.io (Storage) previamente criado em container
6. Orquestrar a solução final utilizando Docker Compose de modo que inclua todos os contêineres utilizados (Servidor Min.io (Storage), Banco de Dados, Artefato Java)
7. Utilizar Flyway Migrations para criar e popular tabelas do banco de dados

Requisitos Específicos:

Implementar uma API Rest para o diagrama de banco de dados acima tomando por base as seguintes orientações:

- Criar um CRUD para Cliente, Ocorrência, Endereço onde deverá ser contemplado a inclusão e edição dos dados das tabelas relacionadas
- Criar um endpoint para realizar cadastro de ocorrências onde deverão ser enviadas as informações de endereço da ocorrência, nome e CPF do cliente envolvido junto à evidência em imagem.
- Criar um endpoint para listar ocorrências apresentando também dados do cliente envolvido, endereço da ocorrência e link para acesso a evidência em imagem previamente cadastrada e disponibilizada pelo storage. O endpoint deverá permitir filtros por nome do cliente, cpf, data da ocorrência e cidade da ocorrência. Deve ser possível ordenar os registros pela data da ocorrência e cidade de forma ascendente ou decrescente.
- Criar um endpoint para finalização da ocorrência. Uma vez finalizada, a ocorrência não poderá ser alterada.
- Realizar o upload de uma ou mais imagens de evidências enviando-as para o Min.io.

Instruções

- O projeto deverá ser disponibilizado em um repositório Git. [_____](#)
No dia da entrega deverá ser compartilhado com os usuários do bitbucket o seu repositório com permissão de leitura para os seguintes usuários avaliadores da CasaDoKct.
 - QA#1 - qa_pela_saco@pqp.com.br
- Crie um arquivo README.md contendo orientações de como executar e testar a solução apresentada
- Decorrido o prazo de entrega, nenhum outro commit deverá ser enviado ao repositório do projeto.

- Saber usar esta ferramenta de versionamento é imprescindível para esta vaga, portanto, não será disponibilizado tutorial ou orientação de uso.
- Codifique como se fosse um sistema para entrar em produção, que poderia sofrer alterações e adição de novas features;
- Adicione as dependências que considerar necessárias;
- Deverá estar disponível no repositório de versionamento todos os arquivos e scripts utilizados para a solução.

O que esperamos?

- Implementação de todos os requisitos;
- Explicação detalhada de como executar a aplicação e executar os testes;
- Testes;
- Legibilidade;
- Escalabilidade;
- Commits pequenos;
- Ver sua experiência codificando;
- Clean Code;
- Soluções simples e práticas;

O que nós não esperamos?

- Descobrir que não foi você quem fez seu teste;
- Ver commits grandes, sem muita explicação nas mensagens em seu repositório;

O que avaliaremos?

- Requisitos alcançados;
- Histórico de commits do git;
- As instruções de como rodar o projeto;
- Organização, semântica, estrutura, legibilidade, manutenibilidade do seu código;
- Atendimento aos objetivos propostos;
- Escalabilidade da solução adotada; Não conseguiu fazer tudo? A sua forma de priorizar a entrega também vai ser considerada. Escreva no README.md o que você conseguiu e não conseguiu implementar, descrevendo porque você preferiu priorizar desta maneira.

Não conseguiu fazer tudo?

A sua forma de priorizar a entrega também vai ser considerada. Escreva no README.md o que você conseguiu e não conseguiu implementar, descrevendo porque você preferiu priorizar desta maneira.