

Mobi7 / Localiza – Recrutamento e Seleção de Desenvolvedores

Preparativo para Code Interview – BackEnd – V1.0

Descrição do desafio

Determinado cliente possui interesse em saber quanto tempos seus veículos ficaram em determinadas regiões. Para isso, esse cliente cadastrou seus pontos de interesse (POIs) no sistema e agora deseja visualizar numa interface web o tempo que cada veículo permaneceu nos pontos cadastrados. Você será responsável por efetuar essa entrega* para ele.

*A entrega desta prova refere-se apenas as APIs de backend para consumo do frontEnd. Não contemplando o frontEnd em si.

O que você possui previamente

- Base de Dados de Posição posicoes.csv
 - Formato do arquivo – colunas separadas por virgula

placa	Data_posicao	velocidade	longitude	latitude	ignicao
Placa do veículo	Data em que a posição foi enviada	Velocidade do veículo em km/h	Longitude em formato decimal	Latitude em formato decimal	True – veículo ligado False – Veículo Desligado

Exemplo:

TESTE001,Wed Dec 19 2018 15:07:19 GMT-0200 (Hora oficial do Brasil),10,-51.549662,-25.5244493,true

Explicação do exemplo acima:

Veículo de placa TESTE001 estava no dia 19/12/18 às 15h07m19s (Horario de Brasilia) com velocidade de 10 km/h na geolocalização (Longitude: -51.549662, Latitude :-25.5244493) com a sua ignição ligada.

Para ver onde é a posição no google maps por exemplo:

<http://maps.google.com/maps?q=-25.5244493,-51.549662>

- Base de Dados de POI
 - Formato do Arquivo

nome	raio	latitude	longitude
Nome do POI	Raio em metros do círculo em que latitude e longitude são o centro	Longitude em formato decimal	Latitude em formato decimal

Exemplo:

PONTO 24,350,-25.36496636999715,-51.46980205405271



O que você vai entregar

Requisitos Funcionais Obrigatórios:

- APIs de Backend de consulta retornara a **quantidade de tempo que os veículos passaram dentro de cada POI**. Ou seja, deseja-se saber, quanto tempo os veículos da lista de posições permaneceram dentro de cada um dos Pontos de Interesse (POI) da lista de POIs cadastradas pelo cliente.
 - Esta API será consumida pelo FrontEnd, sendo possível exibir ao usuário uma tabela com as informações resumidas.
 - Usuário poderá filtrar por data e/ou por placa do veículo no FrontEnd. A API do backend deve entregar a informação filtrada para o front (não haverá processamento no frontEnd);
- APIs de Backend para inserção de novos dados de posicionamento de veículos (referentes a tabela Dados de Posição posicoes.csv).
- APIs de Backend para inserção de novos Pontos de Interesse (dados similares a tabela Dados de Posição posicoes.csv)
- Por simplicidade, não é necessário realizar autenticação de usuário.

Para candidatos de Nível Sênior e Especialista:

- A solução deve ser minimamente escalável, sendo possível realizar a ingestão de novos dados de posição de veículos através de API própria. Além disso, a consulta dos dados deve ser otimizada, tendo capacidade para suportar um nível elevado de TPS (*transactions per second*). Ter em mente que, para um mesmo cliente, o FrontEnd irá realizar diversas requisições idênticas de consulta de dados na API do Backend a ser desenvolvida.

Requisitos Não-Funcionais:

- Sobre os fontes:

- O código fonte deve ser publicado em repositório público (por exemplo github através de uma conta free) até a data e horário combinado em e-mail;
- Todo artefato usado em desenvolvimento (Scripts, códigos, etc) deve ser salvo no repositório
- Sobre as tecnologias permitidas
 - Banco de Dados: escolha livre por parte do candidato desde que Open-Source;
 - Back-End: Java 7/8, Spring Boot, Hibernate, JPA, Kotlin
 - Web-Server: TomCat, nginx, Undertow, Glassfish;
 - Documentação mínima: Swagger dos endpoints
 - Containerização da aplicação (docker) e uso de cache (redis, memcache, etc) são opcionais, porém são pontos de destaque. Testes (unitários e integração) também são opcionais.

O que será avaliado

- Explicações na Code Interview sobre trade-offs técnicos efetuados na elaboração;
- Domínio técnico das tecnologias e frameworks empregados;
- Legibilidade/Modularização do código;
- Design Patterns;

Considera-se que os artefatos gerados estarão disponíveis sob domínio público sob a licença

