

Desafio Radix

Você é um desenvolvedor em uma grande empresa do setor de óleo e gás. Uma das plantas dessa empresa instalou sensores em seus 2.000 equipamentos e você foi encarregado de criar a infraestrutura para receber os dados desses sensores em tempo real.

Os equipamentos são capazes de enviar dados no formato JSON para um endpoint. Um exemplo do payload é:

```
{
  "equipmentId": "EQ-12495",
  "timestamp": "2023-02-15T01:30:00.000-05:00",
  "value": 78.42
}
```

- **equipmentId** é o identificador do equipamento;
- **timestamp** é a data e hora que o evento ocorreu, incluindo o fuso horário.
- **value** é o valor do sensor com a precisão de duas casas decimais.

Suas tarefas:

- Modele um banco de dados da sua escolha para o caso de uso apresentado;
- Crie uma API com um endpoint que receba as requisições em tempo real e armazene no banco de dados;
- Alguns dos sensores da planta podem apresentar falhas técnicas, resultando em lacunas nos dados. Para lidar com isso, o fornecedor pode enviar arquivos CSV com os dados perdidos. Adicione na API um endpoint que receba um arquivo CSV, realize o parser dos dados e salve os valores no banco de dados. O formato do csv pode ser encontrado na tabela 1;
- Crie uma tela que exiba o valor médio de cada sensor nas últimas 24 horas, 48 horas, 1 semana ou 1 mês. Sua tela deve possuir gráficos para facilitar a análise;
- Crie uma documentação para a sua solução.

equipmentId	timestamp	value
EQ-12495	2023-02-12T01:30:00.000-05:00	78.8
EQ-12492	2023-01-12T01:30:00.000-05:00	8.8

Tabela 1 - Exemplo do csv gerado manualmente.

Bônus:

- Crie testes de integração e unitários para a solução;
- Crie scripts que automatizem o deploy e execução das aplicações;
- Implemente um mecanismo de autenticação no sistema;
- Crie um relatório de teste de carga para a solução num cenário de 500, 1000, 5000 e 10000 requisições simultâneas.
- Sua solução foi um sucesso e a empresa deseja expandi-la globalmente. Quais componentes extras você adicionaria na solução existente? Desenhe um diagrama para exemplificar a sua proposta de mudança.

Observações:

- O desafio foi estruturado de forma a abranger todos os níveis de senioridade, tente entregar o máximo de features que conseguir, mas entenda que não existe certo ou errado nesse caso. O bônus não é obrigatório, mas a realização dessas tarefas ajuda no processo de nivelamento de senioridades mais altas.
- Recomendamos que a solução seja feita em JavaScript/TypeScript e/ou Python. Mas sinta-se à vontade para utilizar outras tecnologias.
- O projeto precisa estar publicado no seu GitHub, com visibilidade pública. O link deve ser disponibilizado para o RH.