

Teste para Desenvolvedor Back End Sênior

Sistema de Receituário Digital

Contexto do Problema

Você deve criar uma API para processar uploads de arquivos CSV contendo dados de prescrições médicas e medicamentos. O sistema deve validar, processar e disponibilizar os dados através de endpoints REST.

Cenário de Negócio

Uma clínica precisa digitalizar suas prescrições históricas. Eles têm arquivos CSV com dados de medicamentos prescritos que precisam ser importados, validados e disponibilizados para consulta.

Estrutura do CSV de Entrada

```
id,date,patient_cpf,doctor_crm,doctor_uf,medication,controlled,dosage,days,duration,notes
RX001,2024-01-15,12345678901,123456,SP,Dipirona Sódica,false,500mg,8/8h,7,Tomar após as refeições
RX002,2024-01-16,98765432109,789012,RJ,Amoxicilina,false,875mg,12/12h,10,
RX003,2024-01-17,11122233344,123456,SP,Lorazepam,true,1mg,12/12h,30,1cp as 7h e as 19h
```

Requisitos Funcionais

1. Upload e Processamento de CSV

- Endpoint para upload de arquivo CSV
- Consistência dos dados conforme regras de validação
- Armazenamento dos dados válidos

2. Consultas de processamento

- Relatório de erros/sucessos

Regras de Validação

Campos Obrigatórios:

- **id**: único no sistema
- **date**: data válida
- **patient_cpf**: 11 dígitos
- **doctor_crm**: apenas números
- **doctor_uf**: UF válida (2 letras)
- **medication**: obrigatório
- **controlled**: boolean (quando vazio considerar false)
- **dosage**: obrigatório
- **frequency**: obrigatório
- **days**: número positivo

Teste para Desenvolvedor Back End Sênior

Sistema de Receituário Digital

Regras de Negócio:

- **frequency**: duração máxima de 90 dias
- **date**: não pode ser futura
- Medicamentos controlados (**controlled=true**) requerem observações
- Medicamentos controlados (**controlled=true**) têm **frequency** máxima de 60 dias

Endpoints Esperados

POST /api/prescriptions/upload Upload e processamento de CSV

GET /api/prescriptions/upload/:id Status do processamento

Estrutura de Resposta Esperada

Para ambos os endpoints:

```
{
  "upload_id": "uuid",
  "status": "processing|completed|failed",
  "total_records": 150,
  "processed_records": 145,
  "valid_records": 140,
  "errors": [
    {
      "line": 5,
      "field": "patient_cpf",
      "message": "CPF inválido",
      "value": "12345ABC900"
    }
  ]
}
```

Requisitos Técnicos

1. **Tecnologias**
 - Node.js (versão 20+)
 - Express.js
 - NestJS
2. **Banco de Dados**
 - À sua escolha (pode ser in-memory)
 - Modelagem adequada para consultas
3. **Processamento Assíncrono**
 - Upload deve responder imediatamente
 - Processamento em background
 - Status consultável via endpoint

Teste para Desenvolvedor Back End Sênior

Sistema de Receituário Digital

Entrega

1. **Código fonte** em repositório Git
2. **README** com instruções de instalação e execução

Instruções Importantes

- Priorize a qualidade das validações médicas
- Documente decisões técnicas importantes
- Considere cenários de erro (arquivo corrompido, dados inválidos)
- Implemente logs para auditoria
- Pense em escalabilidade para arquivos grandes

Dica: Foque na robustez das validações e no tratamento de erros. Em sistemas médicos, a qualidade dos dados é crítica.