

requirements.txt

jornada_fast_api > backend > requirements.txt

```
1 fastapi
2 uvicorn
3 sqlalchemy
4 python-dotenv
5 pandas
6 asyncpg
7 python-multipart
8
```

Docker não abria corretamente.

Solução:

Adição de python-multipart

Inserir um arquivo que não é CSV

Ao inserir um arquivo que não é CSV,
o seguinte erro é apresentado

```
{  
  "detail": "O arquivo deve ser um CSV."  
}
```

Inserir um CSV com tabelas alternativas

Erro ao inserir um csv com tabelas alternativas

```
{  
  "detail": "400: CSV deve conter as colunas: {'solicitante', 'numserie', 'servico', 'valor', 'status', 'data'}"  
}
```

Para teste de performance foram inseridas 5000 ordens

```
{  
  "message": "5000 ordens de serviço inseridas com sucesso."  
}
```

Inserir uma linha com data errada

```
{  
  "message": "4 ordens de serviço inseridas com sucesso."  
}
```

Foram enviadas 4 ordens de serviço corretas e 1 com data errada. Nesse caso foram inseridas apenas 4, o código desconsiderou a errada.

Inserir uma tabela correta

Ao inserir o arquivo CSV corretamente, o seguinte erro foi apresentado:

```
{  
  "detail": "name 'OrdemServico' is not defined"  
}
```

```
# Inserir no banco de dados  
for _, row in df.iterrows():  
    os = OrdemServico(  
        solicitante=row["solicitante"],  
        data=row["data"].date() if isinstance(row["data"], datetime) else None,  
        servico=row["servico"],  
        valor=row["valor"],  
        status=row["status"],  
        numserie=row["numserie"]  
    )  
    db.add(os)  
db.commit()
```

backend/src/api/Analise.py

O erro é que o arquivo está buscando pelo nome OrdemServico, sendo que o nome atribuído no banco de dados é apenas OS

```
os = OS(  
    solicita  
    data=row
```

Para corrigir basta apenas alterar o nome para OS

```
{  
  "message": "5 ordens de serviço inseridas com sucesso."  
}
```

Depois de alterado o nome, as ordens de serviço são inseridas corretamente

Inserir valores errados ou nulos

```
{  
  "message": "5 ordens de serviço inseridas com sucesso."  
}
```

Após inserir valores errados, por exemplo o campo valor = cem reais ao invés de 100. A ordem é inserida com sucesso apesar do erro.

O mesmo ocorre com valores nulos, por exemplo o campo solicitante vazio.

No banco de dados, o solicitante é único. Mas ao inserir duas linhas com o mesmo solicitante o valor é adicionado com sucesso.