

## TESTE DE PRÁTICO LUMI – DESENVOLVEDOR(A) FULL STACK PLENO(A)

Olá, candidato(a) a Desenvolvedor Full Stack Pleno, estamos empolgados em ter você no processo de seleção da Lumi. Como parte crucial deste processo, apresentamos um desafio técnico que nos permitirá avaliar suas habilidades e competência em desenvolvimento Full Stack.

**Nota Importante:** O trabalho submetido para este teste não será utilizado para fins comerciais ou integrado em nossos produtos. O objetivo é puramente avaliativo.

### DESCRIÇÃO DO DESAFIO

Nós iremos fornecer algumas faturas de energia elétrica. Seu objetivo será desenvolver um código que seja capaz de:

- Extrair os dados relevantes dessas faturas.
- Organizar esses dados de maneira estruturada em um banco de dados PostgreSQL.
- Apresentar esses dados em uma aplicação web, por meio de uma API.

**Tecnologias a serem utilizadas:** Typescript/JavaScript, Node.js (Fastify/Express) e React.

### DETALHAMENTO DO DESAFIO

**1. Extração de Dados:** Você deve desenvolver um extrator de dados para capturar os dados das faturas de energia elétrica fornecidas (anexo FATURAS) e extrair as informações relevantes. Estas incluem, mas não estão limitadas a:

- “Nº DO CLIENTE”, por exemplo:

Nº DO CLIENTE  
**7202210726**

- “Mês de referência”, por exemplo:

Referente a  
**SET/2024**

- ‘Energia Elétrica’ – Quantidade (kWh) e Valor (R\$),
- ‘Energia SCEE s/ ICMS’ – Quantidade (kWh) e Valor (R\$),
- ‘Energia Compensada GD I’ – Quantidade (kWh) e Valor (R\$), por exemplo:

Itens da Fatura	Unid.	Quant.	Valores Faturados	
			Preço Unit	Valor (R\$)
Energia Elétrica	kWh	100	1,04841351	104,81
Energia SCEE s/ ICMS	kWh	1.860	0,58125187	1.081,12
Energia compensada GD I	kWh	1.860	0,56148931	-1.044,37

- ‘Contrib Ilum Publica Municipal’ – Valor (R\$), por exemplo:

Contrib Ilum Publica Municipal 47,57

**Observação:** Pode ser útil utilizar uma biblioteca como [pdf-lib](#) ou [pdf-parse](#) no Node.js para a extração eficiente dos dados dos PDFs fornecidos.

**2. Banco de Dados:** O Banco de dados deve estar em PostgreSQL. O uso de ORMs

(Sequelize/Prisma/Knex) será um diferencial.

**3. Aplicação:** Os dados armazenados no banco de dados devem ser acessíveis e visualizáveis por meio de uma aplicação web, utilizando React para o Front-end e Node.js para o back-end. O back-end pode ser elaborado utilizando os frameworks Fastify ou Express.

**3.1. Variáveis de Interesse:** abaixo estão elencadas as variáveis de interesse, algumas delas calculadas:

- **Consumo de Energia Elétrica (KWh):** corresponde ao somatório das variáveis 'Energia Elétrica kWh' + 'Energia SCEE s/ICMS kWh', por exemplo:

Exemplo (Abril/24): Consumo de Energia Elétrica = 50 kWh + 476 kWh = 526 kWh;

- **Energia Compensada (kWh):** corresponde à variável 'Energia Compensada GD I (kWh)';
- **Valor Total sem GD (R\$):** somatório dos valores faturados de 'Energia Elétrica (R\$)' + 'Energia SCEE s/ ICMS (R\$)' + 'Contrib Ilum Publica Municipal (R\$)';
- **Economia GD (R\$):** corresponde à 'Energia compensada GD I (R\$)'.

**3.2. Páginas da Aplicação:** a aplicação será composta por duas páginas, sendo elas:

**3.2.1. Dashboard:** esta página trata gráficos e cards representando os totais das variáveis, os gráficos de interesse são:

- Resultados de Energia(kWh) = Consumo de Energia Elétrica KWh vs Energia Compensada kWh;
- Resultados Financeiros (R\$) = Valor Total sem GDR\$ vs Economia GDR\$.

**3.2.2. Biblioteca de Faturas:** Página dedicada à disponibilização das faturas ao usuário final, permitindo que o usuário selecione um determinado **Nº DO CLIENTE** e realize o download de sua fatura de energia elétrica em um mês específico. Ex:

Consumidores													Distribuidoras													2018													2019													2020													2021													2022													2023													2024													Busca																																																			
Nome da UC													Número da UC													Distribuidora													Consumidor													Jan													Fev													Mar													Abr													Mai													Jun													Jul													Ago													Set												
CASA DONA COMERCIO VAREJISTA E SOLUÇ...													3002863513													CEMIG													CASA DONA COMERCIO VAR...																																																																																																																																	
Walter Boaventura da Silva													3003336712													CEMIG													Walter Boaventura da Silva																																																																																																																																	

**3.3. Detalhes adicionais da aplicação:**

**3.3.1. Filtros e Visualização:** A especificação exige filtros por número do cliente e período de análise. Garanta que o front-end com React tenha uma interface intuitiva para essa navegação.

**3.3.2. Gráficos:** Use bibliotecas como [Chart.js](#) ou [Recharts](#) para plotar gráficos claros e personalizáveis.

**3.3.3. Cards Resumo:** Crie cards no dashboard para exibir os totais acumulados por variável (Ex: total de energia consumida, valor compensado, etc.).

## 4. Testes Automatizados

**4.1. Testes unitários podem incluir validação de:**

- Parsing correto dos PDFs.
- Inserção no banco e retorno pela API.
- Cálculo correto dos valores agregados.

**5. Envio e Hospedagem:** A hospedagem da aplicação é opcional, mas pode ser um diferencial positivo. Você pode usar plataformas gratuitas como Vercel e Render para essa finalidade.

### ENTREGA

- **Código:** Disponibilizar o código em um repositório público no GitHub.
- **Documentação:** Fornecer um README detalhado com instruções de configuração, instalação, execução e uso da aplicação.
- **Demonstração:** Opcionalmente, hospede a aplicação em um servidor de sua escolha e nos envie o link para visualizarmos a aplicação web.

Envie o teste para os emails: [gabriel@labs-lumi.com.br](mailto:gabriel@labs-lumi.com.br), [victor@labs-lumi.com.br](mailto:victor@labs-lumi.com.br), [marcelo@labs-lumi.com.br](mailto:marcelo@labs-lumi.com.br), [nicolas@labs-lumi.com.br](mailto:nicolas@labs-lumi.com.br), contendo o código fonte em repositório Git (compartilhar o acesso com estes mesmos e-mails), juntamente com instruções no README do projeto sobre como configurar e executar a aplicação. Se possível, hospede a aplicação em um servidor de sua escolha e nos envie o link para visualizarmos a aplicação web, conforme explicitado no item **5. Envio e Hospedagem**.

Este desafio é projetado **exclusivamente** para avaliar sua experiência e competência com as tecnologias listadas, bem como suas habilidades de resolução de problemas, organização e atenção aos detalhes.

**Data limite para envio do teste prático: 20/10/2024**

Boa sorte! Estamos ansiosos para ver o que você irá criar!