

Desafio Técnico – Desenvolvedor(a) Fullstack

Contextualização

A SmartMart Solutions é uma empresa varejista digital em expansão que atua com múltiplos produtos e categorias em um ambiente altamente competitivo. O time de operações comerciais busca um sistema interno para cadastro, visualização e análise de produtos e vendas, com um painel visual simples e eficiente para auxiliar na tomada de decisões.

Você foi contratado para construir o primeiro protótipo funcional dessa aplicação, unindo a construção de APIs em Python e uma interface visual em React. O objetivo é criar uma base sólida para visualização de dados de vendas e permitir a inserção de produtos manualmente ou via arquivos CSV, com filtros e edição.

Arquivos Disponíveis

1. `products.csv`

Contém a lista inicial de produtos com colunas como `id`, `name`, `category_id`, `price`, entre outras.

2. `categories.csv`

Traz as categorias de produtos com seus respectivos `id` e `name`.

3. `sales.csv`

Histórico de vendas ao longo do ano, com colunas como `product_id`, `month`, `quantity`, `total_price`.

Entregáveis

Código Python (Back-end)

- Forneça os arquivos `.py` da sua implementação, utilizando o framework de sua escolha (ex: FastAPI, Flask ou Django).
- A base de dados pode ser relacional (PostgreSQL, SQLite, etc.) ou em memória, conforme preferir.

Frontend (React)

- Código em React, Vite ou Next.js.
- O estilo deve ser feito utilizando **Tailwind CSS** com bibliotecas como **Shadcn/Radix** ou **Ant Design**.

Documentação (README.md)

- Instruções claras de como instalar, rodar e testar o projeto localmente.
- Se tiver deploy em produção, inclua o link público.
- Adicione prints ou GIFs demonstrando a interface (opcional).

Submissão

Envie sua solução para hanielly.rodrigues@apollosolutions.dev em **até 4 dias após o recebimento deste desafio.**

Inclua no corpo do e-mail:

- Seu nome completo
- Uma mensagem breve sobre sua submissão
- Os arquivos organizados (ex: em um [.zip](#), com estrutura clara de [frontend/](#) e [backend/](#)) **OU** um link para um repositório no github.
- PDF com o título “Questions - seu nome” contendo as 3 questões dissertativas.

Segue o Enunciado abaixo.

Interface Requirements:

- An interface to insert products based on the provided CSV.
- All the data should be in a database of your choice.
- A small and simple dashboard showing charts for sales, including profit by total_price variation, over a year.

For backend:

Note: this needs to be in Python; the framework is up to you.

- [] A POST endpoint for inserting new products;
- [] A GET endpoint for listing products, sales, including profit.
- [] A GET endpoint for listing categories.
- [] A POST endpoint for inserting products from a CSV file

For frontend:

Note: Framework needs to be a React or a React variant, for example (Vite, Nextjs, Quik), the libraries are up to you, but make sure you are using Tailwind CSS for styling (shadcn/radix) or Ant Design as a component library.

- [] Dashboard with a minimum of two charts, with sales representing quantity and sales profit.

You can use bar charts or line charts, it's up to you.

- [] A form for product registration
- [] The possibility to upload a CSV file and insert new products from it
- [] Filters by product category

Extras:

- [] Allow the user to edit the values, like the number of sales and prices, for each month of the year.
- [] Insertion of new categories.
- [] The possibility to download a CSV from the database products/sales.

Important

Write a documentation on a README.md file with instructions on installing and running your project locally. If you have deployed and have a public URL, please add that to the readme file too.

Questions

1. What would be your first improvements if you had more implementation time?

2. What approach would you use to insert a price history into the database and chart for the dashboard?
3. What changes would need to be made to support updates in the product categories to have a discount percentage so that whenever the discount percentage is changed, the new price would be reflected in all products of the same category?