

Desafio | Front-end - Módulo 3

Após alguns meses trabalhando em projetos mais simples, você foi designado pelo seu Tech Lead para desenvolver uma aplicação que será apresentada a um cliente muito importante.

Para acessar o repositório da API que será consumida no front, acesse o seguinte [link](#).

Para o seu desenvolvimento foi liberado layout que você pode encontrar no seguinte [link](#).

Para acessar o pdf contendo as informações de id e class, acesse o seguinte [link](#).

O sistema trata-se de uma aplicação para controles de finanças pessoais, e as funcionalidades são:

Observação: Existem funcionalidades obrigatórias e opcionais!

- Cadastro de uma nova transação (**Obrigatório**)
- Edição de uma transação (**Obrigatório**)
- Exclusão de uma transação (**Obrigatório**)
- Listagem de transações (**Obrigatório**)
- Ordenação dos itens conforme clicado no header da tabela nas seguintes colunas: Data, Dia da semana e Valor (**Obrigatório**)
- Na parte de resumo, o valor de entradas, saídas e saldo é referente ao que está listado na tabela (**Obrigatório**)
- A parte de filtros não é toda obrigatória, veja abaixo:
 - Dia da semana (**Opcional**)
 - Categoria (**Opcional**)
 - Valor (**Obrigatório**)

Dicas importantes

Você receberá um *boilerplate* (um projeto com algumas configurações prontas).

Para começar a trabalhar você precisará executar alguns comandos, mas isso é somente na **primeira vez**.

1. Realizar o fork do repositório
2. Clonar para seu computador o fork feito
3. Executar o `npm install` (para instalar as dependências)
4. Executar o `npm start` (Para "rodar" o projeto! Esse passo deve ser executado todas as vezes que você for desenvolver a aplicação)

Quando você fizer todos esse procedimentos, o projeto em `React` irá executar e exibir no seu navegador uma aplicação básica.

Observação: Caso não abra automaticamente o seu navegador, você pode acessar o projeto usando o endereço <http://localhost:3000> .

Detalhamento de Requisitos:

Cadastro de uma nova transação:

Para cadastrar uma nova transação o usuário deverá clicar no botão `Adicionar Registro` , que ficará logo abaixo da área de `resumo` .



Adicionar Registro

Ao clicar no referido botão, um modal com a opção de adicionar informações de uma transação deve ser exibido:

Adicionar Registro

×

Entrada

Saída

Valor

Categoria

Data

Descrição

Confirmar

Nesse modal todas as informações devem ser preenchidas, lembrando que você pode adicionar uma entrada ou saída de dinheiro, por padrão o valor deve ser o de saída, caso o usuário queira adicionar um valor de entrada ele precisará clicar no botão **Entrada**.


*Todos os campos são obrigatórios!

Após realizar as validações, após o usuário clicar no botão **confirmar**, uma nova transação deve ser inserida e a tabela de listagem deve ser atualizada.

É importante lembrar que quando adicionarmos uma nova transação, a área de resumo deve refletir o que a tabela está exibindo, refazendo assim todos os cálculos após a inserção de um novo registro.

Editar uma transação:

Para editar uma transação o usuário deverá clicar no ícone do lápis, que se encontrará na tabela de listagem de transações:

01/09/21	Quarta	Venda dos brigadeiros	Pix	R\$ 100,00	 
----------	--------	-----------------------	-----	------------	---

Esse ícone =>



Ao clicar no ícone de editar uma transação, o modal (que foi utilizado para adicionar uma nova transação) deverá ser aberto e as informações da transação "clorada", deverão ser preenchidas automaticamente, assim como a imagem abaixo:

Editar Registro



Entrada

Saída

Valor

R\$ 100,00

Categoria

Pix

Data

12/09/2021

Descrição

Venda de Brigadeiros

Confirmar

*Novamente, todos os campos são obrigatórios!

Após validar os campos e o usuário clicar em confirmar, a transação deverá ser atualizada na API usando requisição do tipo `patch` ou `put` e a transação deverá ser atualizada na tabela de listagem de transações.

Excluir uma transação:

Para excluir uma transação o usuário deverá clicar no ícone da lixeira, que se encontrará na tabela de listagem de transações:

01/09/21	Quarta	Venda dos brigadeiros	Pix	R\$ 100,00	 
----------	--------	-----------------------	-----	------------	---

Esse ícone =>



Ao clicar nesse ícone, um "popup" irá aparecer para que o usuário confirme ou não a exclusão, fazendo com que não haja exclusões por engano, veja abaixo como aparece o "popup":



Listagem de transações:

As transações registradas por meio dos endpoints da `api`, deverão ser listadas numa tabela que ficará ao centro da página, nessa tabela teremos 6 colunas, sendo:

1. **Data** da transação no formato `dd/mm/yyyy`
2. **Dia da semana**, nessa coluna deveremos utilizar apenas os primeiros nomes dos dias da semana, ao invés de Segunda-Feira, deveremos utilizar o formato `Segunda`.
3. **Descrição**, nessa coluna listaremos as descrições informadas no cadastro de transação.
4. **Categoria**, aqui vamos mostrar as categorias inseridas em cada uma das transações cadastradas.

5. **Valor**, nessa coluna exibiremos os valores de cada uma das transações. Existem uma regra importante nas cores e nos sinais, para valores de **entrada de dinheiro (credit)** exibimos o número positivo e na cor **roxa**, já para **Saídas de dinheiro (debit)** exibimos o número na cor **laranja**.
6. Na última coluna nós não teremos um cabeçalho, nessa coluna ficarão os botões de editar e excluir.

Data ▾	Dia da semana ▾	Descrição	Categoria	Valor	
01/09/21	Quarta	Venda dos brigadeiros	Pix	R\$ 100,00	 
02/09/21	Quinta	-	Lazer	R\$ 58,50	 
03/09/21	Sexta	-	Alimentação	R\$ 12,00	 
06/09/21	Segunda	Venda dos casadinhos	Pix	R\$ 100,00	 

Cada linha da tabela representa uma transação. Portanto cada botão representa a ação para um registro.

Cabeçalho da tabela:

No cabeçalho da tabela deverá haver a opção de clicar e ordenar de forma **crescente** e **decrecente**, para isso basta o usuário clicar no nome da coluna, a cada clique a ordenação deve ser alterada entre **crescente** e **decrecente**.

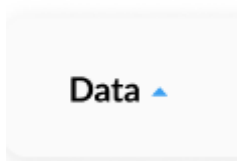
Cada vez que clicamos no nome da coluna, as ordenações feitas em outras colunas devem ser desfeitas, sendo assim, somente é possível ordenar uma coluna por vez.

As colunas que possuem ordenação são: **Data**, **Dia da semana** e **Valor**.

Veja abaixo o ícone que representa que a coluna está sendo ordenada:

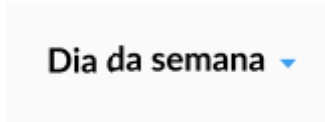
- Ordenando a coluna data de forma crescente (do menor para o maior)

◦



- Ordenando a coluna Dia da semana de forma decrescente (do maior para o menor)

◦



Importante: Somente a coluna em ordenação deve conter os ícones, resumidamente quando clicamos em uma coluna para ordenar, os outros ícones das demais colunas devem desaparecer.

Resumo das transações:

O resumo das transações devem ser exibidos numa "box", onde teremos apenas 3 informações:

- Entradas
- Saídas
- Saldo

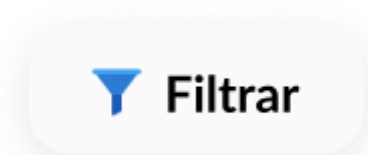
É importante ressaltar que os valores de entrada, saída e saldos são calculados com base nos elementos mostrados na tabela, ou seja, suponhamos que hajam 10 transações cadastradas na `api`, mas a tabela está recebendo um filtro por **dia da semana** e somente 3 transações estão sendo exibidas, basicamente o resumo deve ser calculado com base nessas 3 transações apenas.

Veja na imagem abaixo, como deve ser o resumo;

Resumo	
Entradas	R\$ 200,00
Saídas	R\$ 70,50
<hr/>	
Saldo	R\$ 129,50

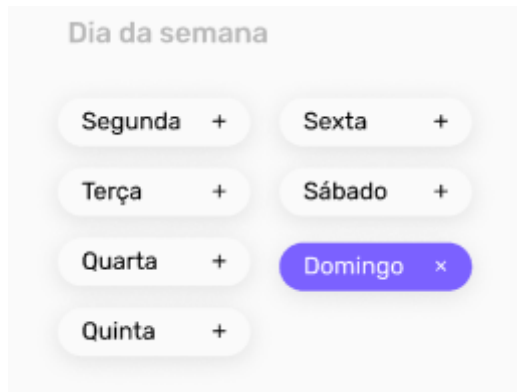
Filtros:

A área de filtros por padrão é oculta, por isso você deve implementar a lógica para que quando o usuário clique no botão **Filtrar** a área de filtro seja exibida e quando clicar novamente seja ocultada, veja abaixo o botão que exibe/oculta a área de filtros:



Os filtros servem para dar granularidade aos dados, ou seja, para haver a possibilidade de exibir as transações conforme selecionamos requisitos para tal. Por exemplo, se dissermos que deve-se

exibir apenas as transações realizadas aos domingos, basicamente deveríamos selecionar o domingo no filtro de **Dia da semana**, veja o exemplo abaixo:



Os filtros são cumulativos, ou seja, se você seleciona para filtrar **segunda** em **dia da semana**, depois seleciona **pix** em **categoria**, informa **valor mínimo 10** e **valor máximo 300**, todos os filtros devem ser levados em consideração. Resumidamente não devemos exibir na tabela transações com: Valores menores que 10, valores maiores que 300, realizadas em um dia diferente de segunda e nem outra categoria que não seja pix.

O funcionamento dos filtros segue a seguinte ordem:

1. Seleciona-se os filtros em dia da semana ou categoria
2. Informa-se valores mínimos e/ou valores máximos
3. Após selecionar os filtros desejados, clica-se no botão **aplicar filtros**.

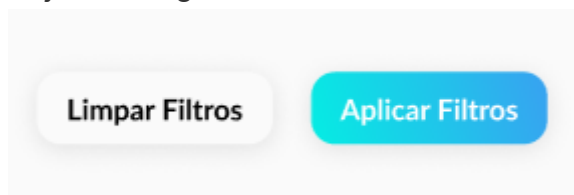
Importante: Não é obrigatório selecionar todos tipos de filtros (Dia da semana + Categoria + Valores Mínimos + Valores Máximo), pois todos eles são independentes.

Para limpar os filtros atuais, o usuário deverá clicar no botão **limpar filtros**

Importante:

1. Ao clicar em aplicar filtros sem nenhum filtro selecionado, o sistema deve exibir todas as transações disponíveis cadastradas.
2. Ao clicar em limpar filtros, o sistema deve exibir todas as transações disponíveis cadastradas.

Veja na imagem abaixo os botões:



Aulas úteis:

- [Layout flex, layouts internos e externos](#)
- [Fetch e assincronismo](#)
- [Props](#)
- [Props Children](#)
- [Quando criar components](#)
- [Estado](#)
- [Eventos no React](#)
- [Gerenciando multiplos inputs de formulário](#)

LEMBRE-SE: é melhor feito do que perfeito!!!

tags: front-end módulo 3 React CSS desafio