

Bootcamp IGTI

Trabalho Prático

Módulo 3

React II

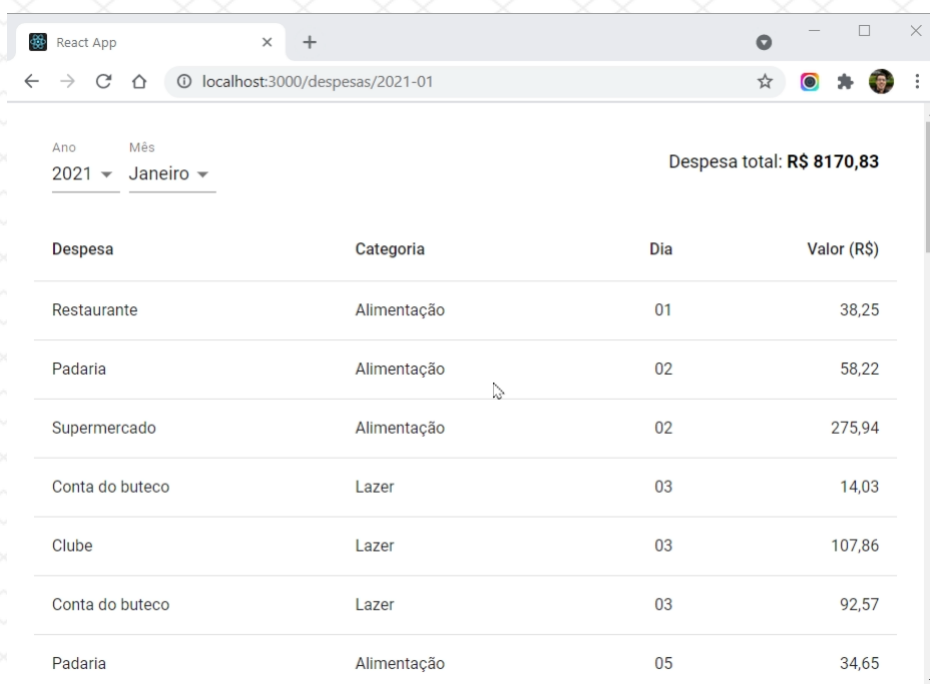
Objetivos

Exercitar os seguintes conceitos trabalhados no Módulo:

- ✓ Criar aplicações React em TypeScript.
- ✓ Utilizar Material-UI.
- ✓ Implementar roteamento com React Router.

Enunciado

Construa, utilizando React, TypeScript Material-UI e React Router, uma aplicação simplificada de gestão de despesas pessoais. Neste trabalho, a aplicação terá uma única funcionalidade, **exibir a lista de todas as despesas** de um determinado ano/mês selecionado e a **soma das despesas**. A seguir, temos um exemplo da possível interface.



The screenshot shows a web browser window with the address bar displaying 'localhost:3000/despesas/2021-01'. The application interface includes a header with 'React App' and a navigation bar. Below the navigation bar, there are two dropdown menus for 'Ano' (Year) and 'Mês' (Month), currently set to '2021' and 'Janeiro' (January). To the right of these dropdowns, the text 'Despesa total: R\$ 8170,83' is displayed. Below this, a table lists expenses with columns for 'Despesa' (Expense), 'Categoria' (Category), 'Dia' (Day), and 'Valor (R\$)' (Value in R\$). The table contains eight rows of data.

Despesa	Categoria	Dia	Valor (R\$)
Restaurante	Alimentação	01	38,25
Padaria	Alimentação	02	58,22
Supermercado	Alimentação	02	275,94
Conta do buteco	Lazer	03	14,03
Clube	Lazer	03	107,86
Conta do buteco	Lazer	03	92,57
Padaria	Alimentação	05	34,65

Os dados exibidos serão buscados de um Back End fornecido pelo professor, que **será postado no fórum de avisos**, com instruções de execução. O Back End oferece um único *endpoint* `/despesas` que retorna um array de objetos, como exemplificado a seguir:

```
[{
  "id": 1,
  "descricao": "Consulta médica",
  "categoria": "Saúde",
  "valor": 197.23,
  "mes": "2020-10",
  "dia": "25"
},
...
]
```

Adicionalmente, podemos filtrar por mês e ordenar os resultados por dia passando os seguintes parâmetros na requisição:

```
http://localhost:3001/despesas?mes=2021-01&_sort=dia
```

Sua aplicação não precisa ter a mesma aparência do exemplo dado, mas precisa atender aos seguintes requisitos:

1. Utilize TypeScript. Defina uma interface para descrever o objeto que representa uma despesa e use-o na aplicação.
2. Deve existir dois campos para selecionar Ano e Mês na interface, usando o componente Select do Material-UI. Ao selecionar, a URL deve mudar para refletir o ano/mês selecionado e os dados devem ser exibidos. Ou seja, você deverá receber ano e mês no formato que achar adequado como um parâmetro da rota, usando React Router.
3. Exiba a soma total das despesas, além da lista de despesas, mostrando os campos **descricao**, **categoria**, **dia** e **valor**.
4. Utilize componentes do Material-UI, quando possível.

Atividades

Os alunos deverão desempenhar as seguintes atividades:

1. Crie a aplicação, usando TypeScript.
2. Adicione as bibliotecas Material-UI e React Router.
3. Busque os dados do Back End para um mês fixo.
4. Construa sua interface e exiba os dados na aplicação.
5. Implemente a mudança de ano/mês, usando React Router.
6. Com base no código fonte da sua aplicação e nos dados exibidos por ela, responda ao questionário de entrega do Trabalho Prático (a aplicação não é entregue, basta responder o questionário).

Dicas

- Comece com um HTML estático no *template* do seu componente e vá adicionando comportamentos dinâmicos aos poucos, testando a cada etapa.
- Consulte o código fonte do projeto construído nas videoaulas se precisar, ele possui muitos exemplos úteis para esta aplicação.
- Para implementar a mudança de ano/mês, não será possível usar o componente Link, mas você pode usar o *hook* **useHistory** do React Router para obter um objeto que permite navegar para outra URL, usando a função **push**.