

Arquitetura JavaScript (React I)

CAPÍTULO 5 – INTEGRAÇÃO COM O BACK END

PROF. RAPHAEL GOMIDE

React I

CAPÍTULO 5 – INTEGRAÇÃO COM O BACK END

PROF. RAPHAEL GOMIDE

Nesta aula



- ☐ Integração do React com o Back End.
- ☐ Criação do projeto **react-flash-cards-v2**.

Integração do React com o Back End



- Em regra, um Back End consiste em uma API que recebe e devolve dados no formato JSON.
- O sistema que trata esses dados pode ser construído com diversas tecnologias (Java, .NET, Node.js, PHP, Python etc.).
- O Back End pode ser considerado como “estado do mundo” (*state of the world*).
- Sendo assim, devemos sincronizar os dados com `useEffect`.
- É interessante, também, dar um *feedback* visual ao usuário enquanto os dados estão sendo carregados.



Integração do React com o Back End



- [Projeto de referência.](#)
- Em **useEffect**, há uma função para a obtenção dos dados com o comando **fetch**.
- O App inicia sem dados e com **loading = true**.
- Após obter os dados, definimos os mesmos e **loading = false**.
- O valor de **loading** é utilizado para definir o que é renderizado.

```
const [loading, setLoading] = useState(true);
const [todos, setTodos] = useState([]);

useEffect(() => {
  async function getTodos() {
    const resource = await fetch(
      "https://jsonplaceholder.typicode.com/todos"
    );
    const json = await resource.json();
    setTodos(json);

    // Forçando a visualização do texto "Carregando..."
    setTimeout(() => {
      setLoading(false);
    }, 500);
  }

  getTodos();
});

if (loading) {
  return <p>Carregando...</p>;
}

return (
  <div>
    <h1>Todo's</h1>
    // restante do código
  </div>
);
```



Prática



- Acompanhe o professor:
 - Criação do projeto **react-flash-cards-v2**.

Próxima aula



- Projeto **react-flash-cards-v3**.