



PROJETO EM GRUPO

Módulo 3 – Sendo
servido



Todos os direitos reservados
©2022 Resilia Educação

CONTEXTO



Chegou o grande dia! Esse é o projeto que encerra o ciclo de Front-end e para isso nada melhor do que **desenvolver seu primeiro site!**

E para isso vocês deverão utilizar o Json-server construído no Projeto Individual.

Caprichem nas telinhas bonitas!





O QUE É PARA FAZER?

Criar um site utilizando o json.server

O projeto deverá ser construído usando o framework React aprendido em aula, nele iremos desenvolver um site que deverá utilizar um servidor Json construído anteriormente. Este projeto deverá seguir um tema dos utilizados pelos integrantes no projeto 1.

COMO FAZER?



Requisitos

- ⇒ Utilize endpoints do Json-server que foi criado no Projeto 1;
- ⇒ Utilizar o README.md do repositório para documentação do projeto;
- Z ⇒ Utilizar a biblioteca react-router-dom para roteamento das páginas;
- O site deve realizar todos os métodos HTTP:
 - **GET** - Página para listar os cadastros ou produtos com opção de busca;
 - **POST** - Página para incluir um novo registro ou produto;
 - **PUT** - Página para editar os cadastros ou produtos;
 - **DELETE** - Opção de excluir itens cadastrados.

EXTRAS



Requisitos Extras

- ⇒ Utilizar CSS Grid para definição do layout das páginas;
- ⇒ Tenha uma animação CSS para carregamento inicial do site (sem biblioteca).
- ⇒ Animações extra
- ⇒ Validação dos campos do formulário além do simples

COMO FAZER?



Dicas

- ⇒ Utilize o axios para realizar as requisições.
- ⇒ Lembre-se de verificar se a requisição terminou para poder consumir os dados na tela, você pode facilmente utilizar isso utilizando o [encadeamento opcional](#).

COMO FAZER?



Documentação e referências

- ⇒ Encadeamento opcional - JavaScript
<https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Operators/Optional_chaining#:~:text=O%20operador%20de%20encadeamento%20opcional%20prov%C3%AA%20uma%20forma,um%20objeto%20obj%20que%20possui%20uma%20estrutura%20aninhada.>>
- ⇒ React Tutorial <<https://react-tutorial.app/>>
- ⇒ Tutorial: Intro to React <<https://reactjs.org/tutorial/tutorial.html>>
- ⇒ Como usar o Axios com o React: o guia definitivo
<<https://www.freecodecamp.org/portuguese/news/como-usar-o-axios-com-o-react-o-guia-definitivo-2021/#:~:text=Conte%C3%BAdo%20O%20que%20%C3%A9%20o%20Axios%3F%202,Como%20tratar%20erros%20com%20o%20Axios%20Mais%20itens>>>



F.A.Q.

Neste projeto eu posso utilizar alguma biblioteca CSS?

Pode sim! Na realidade é até recomendado já que realmente o projeto terá que ser criado em curto tempo e estas bibliotecas poderão ajudar bastante em questão de tempo.

Posso criar outro Json-server específico para este projeto?

Pode, porém, este novo Json-server deve se encaixar em alguns dos temas exigidos no Projeto 1, e esta troca deve ser informada aos facilitadores.

Onde devo utilizar o grid?

O grid deve ser utilizado para o todo da página, ou seja, para criar o layout e posicionar todos os elementos, dentro dos elementos pode ser utilizado o flex para alinhamentos de textos e subelementos

Qual abordagem de componente devo usar?

Pode-se usar a abordagem de componentes de classes ou componentes funcionais, como preferir.

MÃO NA MASSA

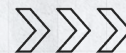


Momento 1 - Início

Neste primeiro momento comece identificando suas dores e dificuldades com o react;

Em seguida identifiquem tudo o que deverá ser realizado no projeto, comece dividindo em grandes grupos, depois isolem cada grupo e identifiquem subtarefas até ter todo o escopo do que o grupo deverá fazer.

Esse primeiro momento é crucial para a organização do projeto e todos devem compreender quais serão as ações dos grupos nos próximos momentos.



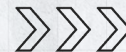


Momento 2 - Desenvolvimento

Agora é a hora de iniciar o código do projeto, para esse primeiro momento vocês podem realizar as seguintes etapas:

- Criar o projeto usando vite.
- Instalar as dependências do projeto com o npm
- Iniciar o git na pasta
- Criar um .gitignore (gitignore.io) para projetos node

Salve e Commite todas as alterações realizadas no projeto, lembrem-se de trabalhar utilizando o Github durante todas as etapas para garantir que todos estão contribuindo com o repositório.



MÃO NA MASSA

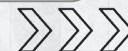


Momento 3 - Desenvolvimento

Momento do grupo focar em avançar com a criação do projeto:

- Criar a estrutura das pastas e realizar a limpeza dos arquivos padrões como App.jsx & main.jsx.
- Criar o repositório no github / Conectar com o projeto local
- Realizar o “git push”
- Começar a desenvolver a tela inicial.

A divisão das tarefas pode ser reajustada em cada etapa de acordo com a necessidade do grupo, lembrem-se de reservar alguns minutos ao final de cada momento de trabalho para manter todos os integrantes atualizados.



MÃO NA MASSA



Momento 4 - Andamento

Esta é a reta final do desenvolvimento, agora é hora de focar em finalizar todas as telas e os eventuais ajustes necessários no projeto.

Mantenham o repositório sempre atualizado com as alterações que foram realizadas no projeto.

Comecem a realizar os testes e a integrar o código que foi criado por cada membro do grupo, verifiquem se ocorreram conflitos durante essas integrações e preparem tudo para a publicação do projeto.



MÃO NA MASSA



Momento 5 - Finalização

Agora vamos realizar os ajustes finais em nosso projeto e realizar o deploy (hospedagem) do nosso projeto

Lembre-se de sempre ir testando o projeto para ver se tudo está funcionando.

Lembre-se de sempre reservar de 10 a 15 minutos juntamente com seu grupo para realizar uma reflexão de como foi construir o projeto e de como foi a experiência de cada um durante esse período.

Teste detalhadamente o projeto depois do deploy para garantir que esteja tudo funcionando.

Entregue o projeto na plataforma.



RUBRICA



Conteúdo	Habilidades
Organização do Projeto	<ol style="list-style-type: none">1. O projeto está usando a estrutura MVC;2. O programa está estruturado em pastas de Componentes, camada de serviço e modelo/hooks;3. Os arquivos/componentes/lógica/hooks estão correspondendo as funcionalidades do padrão;4. O código está indentado e organizado;5. O projeto está usando as boas práticas para componentizar, mantendo o código limpo.
React	<ol style="list-style-type: none">1. O projeto está funcionando normalmente;2. Não existe nenhum problema de sintaxe no projeto;3. Está sendo usada a biblioteca react-router-dom para o roteamento de páginas;4. Existem componentes reutilizáveis;5. Existe pelo menos uma utilização do hook useEffect.
Funcionalidades	<ol style="list-style-type: none">1. Apresentou todas as páginas pedidas;2. Todas as requisições estão funcionando (GET, POST, PUT, DELETE);3. Existe uma validação além do padrão no formulário;4. As requisições estão armazenadas em um arquivo diferente (podendo estar em custom hooks);5. Existe animação de carregamento do site e animações extras, por exemplo em botões.



RUBRICA



Conteúdo	Habilidades
Documentação	<ol style="list-style-type: none">1. O projeto está no GitHub e contém um README.md;2. Readme contém passo a passo para o clone do projeto;3. Readme contém passo a passo para a execução do projeto;4. Readme listando todos os endpoints utilizados e link para o projeto do Json-server utilizado;5. Readme contém informações como imagens, gráficos, links externos.
Git/GitHub	<ol style="list-style-type: none">1. Entregou o link do repositório no GitHub;2. O código está completo/funcionando no GitHub;3. Além do código, foi colocado um arquivo README explicando do que se trata e como pode ser executado o projeto;4. O código foi enviado em commits por etapas;5. As descrições dos commits/PRs estão bem redigidas e apresentam bem as mudanças realizadas.





**Até a próxima e
#confianoprocesso**



Todos os direitos reservados
©2022 Resilia Educação

RESILIA |  **Senac**