

Atividade 6

React – Componentes, Roteamento, Hooks e Axios



Peso 2

V1.0

Universidade Federal do Ceará

Projeto de Interfaces Web

Prof. Victor Farias

- **Data de entrega:** 31/03/21
- **Modo de entrega**
 - Código + vídeo de apresentação zipado pelo moodle2
 - O vídeo também pode ser subido no YouTube e envia o link no arquivo zip
- **Instruções para o código**
 - Excluir node_modules e adicionar os outros arquivos ao zip final
- **Instruções para o vídeo de apresentação:**
 - Somente as funcionalidades que foram mostradas no vídeo receberam nota
 - Lembrar de usar uma qualidade um pouco mais baixa, caso envie o vídeo pelo moodle, pois o moodle só aceita arquivos de até 20mb
 - Pode usar algum software como o obs ou gravar a tela do computador ou celular
 - Tempo máximo do vídeo: 3 minutos

Esboço PaginaFeed logado 700px

LOGO | LOGOUT | LIMPAR HISTÓRICO | POSTAR | VOLTAR

JOÃO
"MENTIRO, TÁ QUENTE DEMAIS HOJE"
27 LIKES | CORTAR

ROBSON
"NUM É QUE TÁ?"

LUA VA
"A RMARIA! NEM ME DIGA"

ESCREVA SUA MENSAGEM

700px

Esboço PaginaPostar logado

LOGO | LOGOUT | LIMPAR HISTÓRICO | POSTAR | VOLTAR

ESCREVA SUA MENSAGEM

SUBMETER

FORMPOSTAR

700px

Roteiro do vídeo

1. Estando deslogado, entre na rota "/" e mostrar redirecionando para login.
2. Entre na página de cadastro "/cadastro", cadastre um usuário e mostre redirecionando para "/login".
3. Na página de login, use email e senha cadastradas para fazer login e mostrar redirecionando para "/" (Página de feed)
 - a. Abra a aba Application no devtools para mostrar que nome e token foram armazenados no localStorage.
 - b. Com o plugin React Developer Tools, vá na aba Components no devtools e mostre token e nome armazenados no contexto.
4. Agora devemos estar na página de feed. Nesse passo, abra a aba de network no devtools para mostrar as requisições sendo feitas.
5. Crie dois posts usando o formulário e mostrar feed recarregando. Deve haver uma requisição POST para inserir o post e um GET para recarregar as mensagens para cada post inserido.
6. Clique em logout. Deve ser redirecionado para "/login"
 - a. Abra a aba Application no devtools para mostrar que nome e token estão null no localStorage.
 - b. Com o plugin React Developer Tools, vá na aba Components no devtools e mostre token e nome estão null no contexto.
7. Cadastre outro usuário
8. Faça login com esse novo usuário
9. Abra a aba network do devtools. Comente no post do primeiro usuário e mostre que o post foi atualizado com o novo post. Deve haver uma requisição POST para inserir novo comentário e uma requisição GET para pegar o comentários do post.

Questões

1. Essa atividade é continuação da atividade anterior.
2. Todas as requisições do axios devem estar em arquivos dentro da pasta api como mostrado nas aulas.

3. Preferencialmente, use o backend que você programou na primeira parte da disciplina. Mas o professor vai fornecer um backend para quem não conseguiu fazer.

4. Componente PáginaCadastro

- a. Página deve estar na rota /cadastro.
- b. Deve conter formulário com campos de nome, email e senha e um botão de submeter.
- c. Ao submeter o formulário, uma requisição deve ser enviada para a api na rota POST /api/usuarios.
- d. Após requisição, redirecionar página para /login.

5. Componente PáginaLogin

- a. Página deve estar na rota /login.
- b. Deve conter formulário com campos de email e senha e um botão de submeter.
- c. Ao submeter o formulário, uma requisição deve ser enviada para a api na rota POST /api/usuarios/signin.
- d. O token retornado deve ser guardado como estado do componente App e deve ser distribuído para aplicação inteira via contexto.
- e. O token também deve ser guardado no localStorage para persistir ao longo dos acessos ao sistema. Ao iniciar o sistema, o token deve ser carregado no estado de App

6. Componente Navegador

- a. Seu navegador deve ter dois modos, o modo logado e o não logado baseado na existência ou não do token
- b. O token deve ser obtido via useContext
- c. Modo logado
 - i. Siga o esboço. O logo deve ficar a esquerda e os botões de linha do tempo, postar e logout ficam a direita juntamente com o nome do usuário logado
 - ii. Ao clicar no botão de logout, o token deve ser setado como null e o navegador deve ir para o modo não logado
- d. Modo não logado
 - i. O logo continua a esquerda. Mas a direita teremos apenas o botão de cadastro e login. O botão de

cadastro leva para a rota /cadastro e o de login para /login.

7. Página PaginaFeed

- a.** A PaginaFeed tem que requisitar os posts para o backend e guardar como seu estado.
- b.** Os posts devem ser renderizados a partir dos posts do backend que estão no estado.
- c.** Ao adicionar um novo post, deve ser feita requisição de adicionar post para o backend e atualizar os posts disparando a requisição de listar posts no backend.
- d.** Se usuário entrar na rota “/” e não houver token, redirecionar para “/login”

8. Página PaginaFeed – Componente Post

- a.** O componente Post deve requisitar ao backend todos os comentários do post que ele recebe via props
- b.** Os comentários ficam armazenados como estado do componente Post
- c.** Ao adicionar um comentário, deve-se fazer uma requisição para inserir o comentário para esse post no backend e atualizar os comentários