

Estilizando o Header.

Crie o arquivo `src/components/Header.css` conforme print abaixo.

```
# Header.css x
1  header#main-header {
2      background: #fff;
3      height: 72px;
4      border-bottom: 1px solid #DDD;
5  }
6
7  header#main-header .header-content {
8      width: 100%;
9      max-width: 980px;
10     margin: 0 auto;
11     height: 72px;
12     padding: 0 30px;
13
14     display: flex;
15     align-items: center;
16     justify-content: space-between;
17 }
```

Importando o arquivo de estilização `src/components/Header.css` no arquivo de cabeçalho.

Edite o arquivo `src/components/Header.js`

Insira a linha 2 neste arquivo para importar o arquivo de estilização.

```
JS Header.js x
1  import React from 'react'
2  import './Header.css'
3  import logo from '../assets/logo.svg'
4  import camera from '../assets/camera.svg'
5
6  export default function Header() {
```

Abra o navegador e será exibida a página abaixo.



Criando um link no Header

A logo do instagram será um link com a página inicial (componente Feed) e a câmera com uma nova postagem (componente New).

Edite o arquivo src/componentes/Header.js

- Insira a linha 2 para importar o Link do react-router-dom;
- Insira as linhas 11 e 13 para que ao clicar no logo do instagram seja exibida a página inicial, componente Feed
- Insira as linhas 14 e 16 para que ao clicar na câmera seja exibida a página para inserir uma nova publicação (new).

```
JS Header.js x
1 import React from 'react'
2 import { Link } from 'react-router-dom'
3 import './Header.css'
4 import logo from '../assets/logo.svg'
5 import camera from '../assets/camera.svg'
6
7 export default function Header() {
8   return (
9     <header id='main-header'>
10       <div className='header-content'>
11         <Link to='/'>
12           <img src={logo} alt='InstaFoa' />
13         </Link>
14         <Link to='/new'>
15           <img src={camera} alt='Enviar publicação' />
16         </Link>
17       </div>
18     </header>
19   )
20 }
```

Abra a página e faça a navegação, clique no ícone do instagram para exibir Feed e na câmera para exibir New.

Trabalhando com o Feed para exibir as últimas postagens

Criando a estrutura dos Feeds

Edite o arquivo `src/pages/Feed.js` e configure-o conforme print a seguir.

```
JS Feed.js x
1  import React, { Component } from 'react'
2
3  import more from '../assets/more.svg'
4  import like from '../assets/like.svg'
5  import comment from '../assets/comment.svg'
6  import send from '../assets/send.svg'
7
8  class Feed extends Component {
9    render() {
10     return (
11       <section id='post-list'>
12         <article>
13           <header>
14             <div className='user-info'>
15               <span>Thiaguinho</span>
16               <span className='place'>Volta Redonda</span>
17             </div>
18             <img src={more} alt='Mais' />
19           </header>
20           <img src='http://localhost:3333/files/Thiaguinho.jpg'
21             alt='' />
22           <footer>
23             <div className='actions'>
24               <img src={like} alt='' />
25               <img src={comment} alt='' />
26               <img src={send} alt='' />
27             </div>
28           </footer>
29         </article>
30       </section>
31     )
32   }
33 }
```

```

29         <strong>900 curtidas</strong>
30         <p>
31             Um show muito bom
32             <span>
33                 #alegria alegria #caraca moleque
34             </span>
35         </p>
36     </footer>
37 </article>
38 </section>
39 )
40 }
41 }
42
43 export default Feed
    
```

Selecione todo o article, copie e cole para baixo para gerar uma nova postagem no navegador.

Abra o navegador e duas postagens serão exibidas.



← → ↻ ⓘ localhost:3000



ThiaguinhoVolta Redonda



900 curtidas

Um show muito bom#alegria alegria #caraca moleque

Criando a estilização da página Feed

Crie o arquivo `src/pages/Feed.css` com o conteúdo abaixo.

```
# Feed.css x
1  section#post-list {
2      width: 100%;
3      max-width: 580px;
4      margin: 0 auto;
5      padding: 0 30px;
6  }
7
8  section#post-list article {
9      background: #FFF;
10     border: 1px solid #DDD;
11     margin-top: 30px;
12 }
13
```

```
# Feed.css x
14 section#post-list article header {
15     padding: 20px;
16
17     display: flex;
18     align-items: center;
19     justify-content: space-between;
20 }
21
22 section#post-list article header .user-info {
23     display: flex;
24     flex-direction: column;
25 }
26
```

```
# Feed.css x
27 section#post-list article header .user-info span {
28   font-size: 13px;
29 }
30
31 section#post-list article header .user-info span.place {
32   font-size: 11px;
33   color: #666;
34   margin-top: 3px;
35 }
36
37 section#post-list article > img {
38   width: 100%
39 }
40
```

```
# Feed.css x
41 section#post-list article footer {
42   padding: 20px;
43 }
44
45 section#post-list article footer .actions {
46   margin-bottom: 10px;
47 }
48
49 section#post-list article footer .actions img {
50   height: 20px;
51   margin-right: 10px;
52 }
53
```



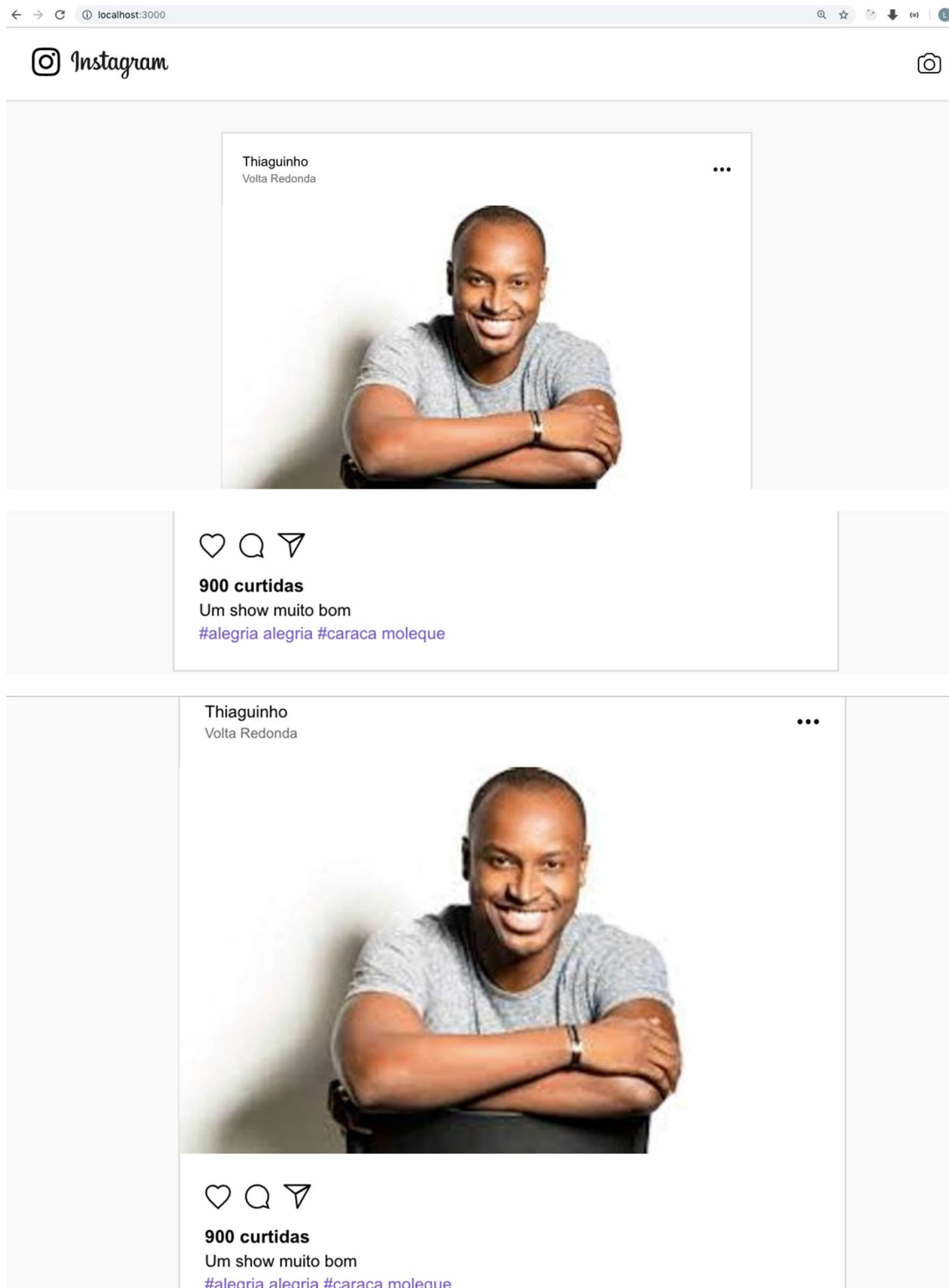
```
# Feed.css x
54 section#post-list article footer p {
55   font-size: 13px;
56   margin-top: 2px;
57   line-height: 18px;
58 }
59
60 section#post-list article footer p span {
61   color: #7159c1;
62   display: block;
63 }
```

Importando o arquivo `src/pages/Feed.css` no arquivo `src/pages/Feed.js`

Edite o arquivo `src/pages/Feed.js` e insira a linha 3

```
JS Feed.js x
1  import React, { Component } from 'react'
2
3  import './Feed.css'
4
5  import more from '../assets/more.svg'
6  import like from '../assets/like.svg'
```


Abra o navegador e verifique se a página será estilizada conforme print a seguir.



Buscando os dados da API

Para buscar os dados da API que está no NodeJS e no banco de dados MongoDB, será utilizada a lib axios. Para instalar digite o comando **yarn add axios**.

- Crie a pasta **services** dentro da **src**

Crie o arquivo **src/services/api.js** e digite o código abaixo.

- Na linha 1 importa a lib do axios;
- Na linha 3 e 4 crie uma variável **api** que armazena a **baseUrl**, ou seja, o caminho de acesso do backend;
- Na linha 7 exporta a variável **api**

```
1 import axios from 'axios'
2
3 const api = axios.create({
4   |   baseUrl: 'http://localhost:3333'
5 })
6
7 export default api;
```

Edite o arquivo **src/pages/Feed.js**

Insira a linha 2 para importa a **api**

```
1 import React, { Component } from 'react'
2 import api from '../services/api'
3 import './Feed.css'
```

- Insira da linha 10 até 18.
- Na linha 11 cria um estado com o atributo feed como array.
- Na linha 15 cria o método componentDidMount() como assíncrono, ele será executado antes do componente ser montado na tela.
- Na linha 16 faz uma requisição para a rota posts com o método get para obter todos os posts
- Na linha 17 alterar o estados do atributo feed utilizando o setState com o conteúdo da variável response.data, automaticamente o método render, será executado

```

10  class Feed extends Component {
11    state = {
12      feed: [],
13    }
14
15    async componentDidMount() {
16      const response = await api.get('/posts')
17      this.setState({ feed: response.data })
18    }
19
20    render() {
21      return (
22        <section id='post-list'>

```

- Insira a linha 23
- Altere a linha 27 para exibir o autor do post
- Altere a linha 28 para exibir o local do post
- Altere a linha 32 para exibir a imagem do post, observe que foi utilizado o template string

```

20    render() {
21      return (
22        <section id='post-list'>
23          { this.state.feed.map(post => (
24            <article key={post._id}>
25              <header>
26                <div className='user-info'>
27                  <span>{post.author}</span>
28                  <span className='place'>{post.place}</span>
29                </div>
30                <img src={more} alt='Mais' />
31              </header>
32              <img src={`http://localhost:3333/files/${post.image}`}
33                alt='' />
34              <footer>
35                <div className='actions'>
36                  <img src={like} alt='' />

```

- Altere a linha 41 para exibir o número de likes do post (evento)
- Altere a linha 43 para exibir a descrição do evento

- Altere a linha 45 para exibir as hashtags do evento

```
37         <img src={comment} alt='' />
38         <img src={send} alt='' />
39     </div>
40
41     <strong>{post.likes} curtidas</strong>
42     <p>
43         {post.description}
44         <span>
45             {post.hashtags}
46         </span>
47     </p>
48 </footer>
49 </article>
50   )}
51 </section>
52
53   }
54 }
55
56 export default Feed
```

Abra o navegador e pressione F5 para atualizar a página, verifique se a aplicação foi exibida com todos os posts (eventos) que foram cadastrados no backend do NodeJS utilizando o insomnia.

Crie um novo post no insomnia de um novo evento, abra a aplicação, atualize a página e verifique se o mesmo foi exibido.