

01

## Exclusão através da API

### Transcrição

Começando deste ponto? Você pode fazer o [DOWNLOAD \(https://s3.amazonaws.com/caelum-online-public/vue/stages/10-alurapic.zip\)](https://s3.amazonaws.com/caelum-online-public/vue/stages/10-alurapic.zip) completo do projeto do capítulo anterior e continuar seus estudos a partir deste capítulo. Será necessário abrir seu terminal, entrar dentro da pasta alurapic e executar o comando `npm install` para baixar novamente todas as dependências da aplicação.

No capítulo anterior aprendemos a enviar dados para nossa API, mas tão importante quando o cadastro de novas fotos é sua exclusão. Já temos o método `remove` que já recebe a foto que desejamos excluir. Daí, basta usarmos `this.$http` para realizar uma requisição com o método `delete` que receberá uma URL que aponta para o endereço do recurso que desejamos apagar. Como as fotos vieram do servidor, todas elas possuem a informação `_id` que foi gerado automaticamente pela API.

Além dessa alteração, vamos também colocar um parágrafo centralizado que exibirá a informação de `mensagem`, propriedade que criaremos em nossa função `data`. No momento da remoção, vamos alterar a propriedade e através do mecanismo de data binding a mensagem será exibida automaticamente para o usuário.

Resumindo:

- criar a propriedade `mensagem` na função `data` iniciada com uma string vazia
- adicionar um parágrafo que exibe a mensagem usando a diretiva `v-show` para oculta-lo quando a mensagem estiver vazia
  - usar `$http` no método `remove` para solicitar a exclusão da foto pela nossa API

```
<!-- alurapic/src/components/home/Home.vue -->

<template>
  <div>
    <h1 class="centralizado">Alurapic</h1>

    <!-- novo elemento para exibir mensagens para o usuário -->
    <p v-show="mensagem" class="centralizado">{{ mensagem }}</p>

    <input type="search" class="filtro" @input="filtro = $event.target.value" placeholder="Filtrar...">
    <ul class="lista-fotos">
      <li class="lista-fotos-item" v-for="foto of fotosComFiltro">
        <meu-painel :titulo="foto.titulo">
          <imagem-responsiva :url="foto.url" :titulo="foto.titulo" v-meu-transform:scale(1.0)>
          <meu-botao
            rotulo="remover"
            tipo="button"
            estilo="perigo"
            :confirmacao="true"
            @botaoAtivado="remove(foto)"/>
        </meu-painel>
      </li>
    </ul>
  </div>
</template>
```

```
</div>
</template>

<script>

import Painel from '../shared/painel/Painel.vue';
import ImagemResponsiva from '../shared/imagem-responsiva/ImagemResponsiva.vue'
import Botao from '../shared/botao/Botao.vue';

export default {

components: {

'meu-painel': Painel,
'imagem-responsiva': ImagemResponsiva,
'meu-botao': Botao
},

data () {
return {

fotos: [],
filtro: '',
mensagem: ''
}
},
computed: {

fotosComFiltro() {

if (this.filtro) {
let exp = new RegExp(this.filtro.trim(), 'i');
return this.fotos.filter(foto => exp.test(foto.titulo));
} else {
return this.fotos;
}
}
},
methods: {

remove(foto) {
this.$http
.delete(`http://localhost:3000/v1/fotos/${foto._id}`)
.then(() => {
// assim que apagar, exibe a mensagem para o usuário
this.mensagem = 'Foto removida com sucesso'
}),
err => {
this.mensagem = 'Não foi possível remover a foto';
console.log(err);
}
)
}
}
}
```

```
},  
  
created() {  
  
  this.$http  
    .get('http://localhost:3000/v1/fotos')  
    .then(res => res.json())  
    .then(fotos => this.fotos = fotos, err => console.log(err));  
}  
}  
</script>  
<style>  
  
.centralizado {  
  text-align: center;  
}  
  
.lista-fotos {  
  list-style: none;  
}  
  
.lista-fotos .lista-fotos-item {  
  display: inline-block;  
}  
  
.filtro {  
  display: block;  
  width: 100%;  
}  
</style>
```

Excelente. Quando clicamos em remover, vemos a mensagem 'Foto removida com sucesso', mas a foto ainda continua! Mais estranho ainda é que se recarregarmos nossa aplicação a foto que apagarmos não é exibida. O que está acontecendo? É o que veremos no próximo vídeo.