



04

Mapeando o vuex

Transcrição

Quando utilizamos o Vuex, é comum acessarmos algumas propriedades mais de uma vez, então a fim de facilitar o trabalho do desenvolvedor, podemos criar alguns assessores para propriedades específicas, chamados de **gatters**.

Definiremos um getter, por exemplo, para o usuário logado. Em `index.js` escreveremos um assessor para retornar um estado do usuário. Passaremos para nosso objeto de configuração o `getters`

```
const getters = {  
  usuarioEstaLogado: state => Boolean(state.token)  
}
```

```
export default new Vuex.Store({  
  state: estado,  
  mutations,  
  actions,  
  getters  
})
```

[COPIAR CÓDIGO](#)

Em `BarraNavegacao.vue` retornaremos o `this.$store.getters.usuarioEstaLogado`.

```
computed: {  
  usuarioestaLogado () {
```

```
    return this.$store.getters.usuarioEstaLogado
  }
}
```

[COPIAR CÓDIGO](#)

Ao testarmos a aplicação, verificaremos que tudo opera como o esperado. Temos uma outra vantagem oferecida pelo Vuex: ao invés de definir o assessor manualmente, podemos mapeá-lo para o interior de nossas propriedades computadas. Em `BarraNavegacao.vue` começaremos pela importação do `mapGetters` de `vuex`. Feito isso, espalharemos a função `mapGetters()` que recebe como parâmetro os assessores que queremos compartilhar.

```
computed: {
  ...mapGetters(['usuarioEstaLogado'])
}
```

[COPIAR CÓDIGO](#)

Ao testarmos a aplicação, verificaremos que tudo opera normalmente.