**Comunicação entre componentes**

**Aula 119 – Comunicação entre componentes**

1. Baixe o conteúdo desta aula para podermos realizar as tarefas.
2. Instale as dependências digitando : npm i
3. Vemos que a aplicação possui três componentes Usuário, UsuárioEditar e UsuárioInfo, nós iremos realizar a comunicação entre estes componentes.
4. No componente Usuario vamos criar um botão abaixo do primeiro parágrafo chamando a função click que irá receber a função ‘Alterar Nome’.
5. Vamos dentro de export Default utilizar a propriedade data() criando dentro dela a variável nome inicialmento com o valor ‘Pedro’.
6. Vamos ainda adicionar a propriedade methods criando o método ‘alterarNome’ dentro do método iremos passar o valor ‘Ana’ para a variável ‘nome’.
7. <template>
8. <div *class*="container">
9. <h1>Componente Usuário</h1>
10. <p>Esse é um componente muito legal!</p>
11. <button @*click*='alterarNome'>Alterar Nonme</button>
12. <hr>
13. <div *class*="componentes">
14. <app-usuario-info />
15. <app-usuario-editar />
16. </div>
17. </div>
18. </template>
19. <script>
20. import AppUsuarioInfo from './UsuarioInfo'
21. import AppUsuarioEditar from './UsuarioEditar'
22. export default {
23. components: { AppUsuarioInfo, AppUsuarioEditar },
24. data() {
25. return {
26. nome: 'Pedro'
27. }
28. },
29. methods: {
30. alterarNome(){
31. *this*.nome = 'Ana'
32. }
33. },
34. }
35. </script>
36. <style *scoped*>
37. *.container* {
38. background-color: #27363b;
39. color: #fff;
40. padding: 10px;
41. }
42. *.container* hr {
43. margin: 20px 10px;
44. }
45. *.componentes* {
46. display: flex;
47. }
48. *.componentes* > \* {
49. margin: 10px;
50. }
51. </style>