

src/components/AppSidebar.vue

```
1 <script setup>
2 import { ref, computed, watch } from 'vue';
3 import LocationItem from './LocationItem.vue';
4
5 const props = defineProps({
6   isOpen: Boolean,
7   allLocais: Array,
8   floors: Array,
9   currentFloorId: String,
10  selectedLocalId: [Number, String, null],
11  loading: Boolean,
12  error: String
13 });
14
15 const emit = defineEmits(['update:currentFloorId', 'select-local', 'close']);
16
17 const searchTerm = ref('');
18
19 // Filtra locais baseado na busca E opcionalmente no andar atual
20 // Decisão: Mostrar sempre locais do andar atual + resultados da busca em outros andares
21 const filteredLocais = computed(() => {
22   const term = searchTerm.value.toLowerCase().trim();
23   if (!term) {
24     // Sem busca, mostra apenas os do andar atual
25     return props.allLocais.filter(local => local.andar === props.currentFloorId);
26   } else {
27     // Com busca, mostra os que batem com a busca (em qualquer andar)
28     return props.allLocais.filter(local =>
29       local.nome.toLowerCase().includes(term)
30     );
31   }
32 });
33
34 const internalCurrentFloor = ref(props.currentFloorId);
```

```
35
36 // Observa mudança externa no currentFloorId
37 watch(() => props.currentFloorId, (newVal) => {
38   internalCurrentFloor.value = newVal;
39 });
40
41 const changeFloor = () => {
42   //searchTerm.value = ''; // Limpa busca ao trocar de andar? Opcional.
43   emit('update:currentFloorId', internalCurrentFloor.value);
44 };
45
46 const handleSelectLocal = (local) => {
47   emit('select-local', local);
48   // Opcional: Fechar sidebar ao selecionar?
49   // emit('close');
50 };
51 </script>
52
53 <template>
54   <div class="sidebar" :class="{ 'is-open': isOpen }">
55     <h1>Locais</h1>
56
57     <div class="floor-selector">
58       <select v-model="internalCurrentFloor" @change="changeFloor">
59         <option v-for="floor in floors" :key="floor.id" :value="floor.id">
60           {{ floor.nome }}
61         </option>
62       </select>
63     </div>
64
65     <div class="search-container">
66       <input v-model="searchTerm" class="search-box" placeholder="Buscar local..." />
67     </div>
68
69     <div v-if="error" class="error">
70       {{ error }}
71     </div>
```

```

72
73 <div v-if="loading && !allLocais.length" class="status-message">Carregando locais...</div>
74
75 <div v-if="!loading && !filteredLocais.length && searchTerm" class="status-message">
76     Nenhum local encontrado para "{{ searchTerm }}".
77 </div>
78 <div v-if="!loading && !filteredLocais.length && !searchTerm" class="status-message">
79     Nenhum local neste andar.
80 </div>
81
82 <div class="locations-list">
83     <div
84         v-for="local in filteredLocais"
85         :key="local.id"
86         class="location-item"
87         :class="{ 'selected': selectedLocalId === local.id }"
88         @click="handleSelectLocal(local)"
89     >
90         {{ local.nome }}
91     </div>
92 </div>
93
94 </div>
95 </template>
96
97 <style scoped>
98 .sidebar {
99     width: var(--sidebar-width); /* Use CSS variable */
100     height: 100vh;
101     background-color: var(--sidebar-bg-color); /* Use CSS variable */
102     position: fixed;
103     left: 0;
104     top: 0;
105     transform: translateX(-100%);
106     transition: transform var(--transition-speed) ease-in-out; /* Use CSS variable */
107     z-index: 1000;
108     display: flex;

```

```
109     flex-direction: column;
110     overflow-y: auto;
111     box-sizing: border-box;
112     padding: 20px;
113     box-shadow: 2px 0 5px rgba(0, 0, 0, 0.1);
114 }
115
116 .sidebar.is-open {
117     transform: translateX(0);
118 }
119
120 .sidebar h1 {
121     font-size: 1.5rem;
122     margin-top: 40px;
123     margin-bottom: 20px;
124 }
125
126 .sidebar .floor-selector {
127     margin-bottom: 15px;
128 }
129
130 .sidebar .floor-selector select {
131     width: 100%;
132     padding: 8px;
133     border-radius: 4px;
134     border: 1px solid var(--border-color); /* Use CSS variable */
135     box-sizing: border-box;
136     background: white;
137 }
138
139 .sidebar .search-container {
140     margin-bottom: 15px;
141 }
142
143 .sidebar .search-box {
144     width: 100%;
145     padding: 10px;
```

```
146     border: 1px solid var(--border-color); /* Use CSS variable */
147     border-radius: 5px;
148     box-sizing: border-box;
149     margin-bottom: 20px;
150 }
151
152 .sidebar .locations-list {
153     max-height: calc(100vh - 200px);
154     overflow-y: auto;
155 }
156
157 .sidebar .location-item {
158     padding: 10px;
159     margin-bottom: 10px;
160     background-color: white;
161     border: 1px solid #eee;
162     border-radius: 5px;
163     cursor: pointer;
164     transition: background-color 0.2s ease, color 0.2s ease;
165     box-shadow: 0 1px 3px rgba(0, 0, 0, 0.05);
166 }
167
168 .sidebar .location-item:hover {
169     background-color: #e9ecef;
170 }
171
172 .sidebar .location-item.selected {
173     background-color: var(--primary-accent-color); /* Use CSS variable */
174     color: white;
175     font-weight: bold;
176     border-color: var(--primary-accent-color); /* Use CSS variable */
177 }
178
179 .sidebar .error {
180     color: var(--error-color); /* Use CSS variable */
181     background-color: var(--error-bg-color); /* Use CSS variable */
182     padding: 15px 20px;
```

```
183     border-radius: 5px;
184     text-align: center;
185     font-weight: bold;
186 }
187
188 .sidebar .status-message {
189     padding: 10px;
190     color: #6c757d;
191     text-align: center;
192     font-style: italic;
193 }
194
195 /* Media Queries (adjust as needed, consider main.css) */
196 @media screen and (max-width: 768px) {
197     .sidebar {
198         width: var(--sidebar-width-mobile); /* Use CSS variable */
199         padding: 15px;
200     }
201     .sidebar h1 {
202         font-size: 1.3rem;
203         margin-top: 55px;
204     }
205 }
206
207 @media screen and (max-width: 480px) {
208     .sidebar {
209         width: 90vw;
210         padding: 10px;
211     }
212     .sidebar h1 {
213         font-size: 1.2rem;
214         margin-top: 50px;
215     }
216 }
217 </style>
```