📄 Beispielhafter #app-Inhalt:

<div id="app" class="container py-4">

<h1 class="mb-4 text-center">🗂️ Kanban Board mit Nutzer, Rollen & Farben</h1>

<!-- Nutzer hinzufügen -->

<div class="mb-3 d-flex gap-2">

<input v-model="state.newUser" placeholder="Nutzername" class="form-control" />

<select v-model="state.selectedRole" class="form-select">

<option disabled value="">Rolle wählen</option>

<option>Product-Owner</option>

<option>Entwickler</option>

<option>Scrum-Master</option>

</select>

<button @click="addUser" class="btn btn-primary">Nutzer hinzufügen</button>

</div>

**🔍 Erklärung:**

* v-model="state.newUser": Zweibindung – das Inputfeld ist mit state.newUser verknüpft.
* v-model="state.selectedRole": Dropdown zur Rollenwahl, gebunden an state.selectedRole.
* @click="addUser": Beim Klick auf den Button wird die Methode addUser() ausgeführt.

<!-- Neue Aufgabe hinzufügen -->

<div class="mb-4 d-flex gap-2">

<input v-model="state.newTask" placeholder="Aufgabe" class="form-control" />

<select v-model="state.selectedUser" class="form-select">

<option disabled value="">Verantwortlich</option>

<option v-for="user in state.users" :key="user.name" :value="user.name">{{ user.name }}</option>

</select>

<input type="date" v-model="state.selectedDate" class="form-control" />

<button @click="addTask('backlog')" class="btn btn-success">Zu Backlog</button>

</div>

**🔍 Erklärung:**

* v-model="state.newTask": Aufgabenbeschreibung wird gebunden.
* v-model="state.selectedUser": Nutzer aus Dropdown wählen.
* v-model="state.selectedDate": Deadline auswählen.
* @click="addTask('backlog')": Aufgabe wird der Spalte „Backlog“ zugewiesen.

<!-- Kanban-Spalten -->

<div class="kanban-board">

<div class="kanban-column" v-for="(tasks, column) in state.columns" :key="column">

<h3>{{ columnNames[column] }}</h3>

<ul class="kanban-list">

<li v-for="(task, index) in tasks" :key="index" class="kanban-item">

<span>

<span :style="{ backgroundColor: getRoleColor(findUser(task.user)?.role) }" class="role-badge"></span>

{{ task.text }} – {{ task.user }} ({{ task.deadline }})

</span>

<div>

<button class="btn btn-sm btn-outline-primary" @click="moveTask(column, nextColumn(column), index)">➡️</button>

<button class="btn btn-sm btn-outline-danger" @click="deleteTask(column, index)">🗑️</button>

</div>

</li>

</ul>

</div>

</div>

</div>

**🔍 Erklärung:**

* v-for="(tasks, column) in state.columns": Für jede Spalte (z. B. "todo", "inprogress") eine eigene Spalte anzeigen.
* columnNames[column]: Lesbarer Name (z. B. „In Progress“ statt „inprogress“).
* v-for="(task, index) in tasks": Jede Aufgabe in der Spalte anzeigen.
* getRoleColor(...): Farbe je nach Rolle bestimmen.
* moveTask(...): Schiebt die Aufgabe in die nächste Spalte.
* deleteTask(...): Löscht die Aufgabe.