# Tool.vue

<template>  
 <div>  
 <div class="heading">  
 <h2>Total sick days per employee</h2>  
 <button class="btn" style="height: 50px; background: #1b4b72; color: white;" onclick="***location***.href='/sickdays/newday'">Add sickday</button>  
 </div>  
 <br>  
  
 <div class="py-6 px-8 w-1/4">  
 <select class="form-control form-select w-full"  
 name="ID" id="Employee" v-model="filter">  
 <option value="" selected>filter on employee</option>  
 <option v-for="(object, key) in tableData" :key="key" :value="object.userID">{{ object.name }}</option>  
 </select>  
 </div>  
  
 <br>  
  
  
 <div class="card">  
 <div class="flex items-center py-3 border-b border-50">  
 <div class="flex items-center"></div>  
 <div class="flex items-center ml-auto px-3 index-actions-container"></div>  
 </div>  
 <div class="relative">  
 <div class="overflow-hidden overflow-x-auto relative" style="">  
 <table cellpadding="0" cellspacing="0" data-testid="resource-table" class="table w-full table-default">  
 <thead>  
 <tr>  
 <th class="text-left" v-if="filter !== ''">  
 <span>ID</span>  
 </th>  
 <th class="text-left">  
 <span v-if="filter === ''">Employee</span>  
 <span v-else>start date</span>  
 </th>  
 <th class="text-left">  
 <span v-if="filter === ''">Day(s)</span>  
 <span v-else>End date</span>  
 </th>  
 <th class="text-left">  
 <span v-if="filter === ''">Do not count day(s</span>  
 <span v-else>Days count</span>  
 </th>  
 <th class="text-left" v-if="filter !== ''">  
 <span>Not counted</span>  
  
 </th>  
  
 <th class="text-left" v-if="filter !== ''">  
 <span>Description</span>  
 </th>  
 <th class="text-left" v-if="filter !== ''">  
 <span>Created date</span>  
 </th>  
 <th class="text-left" v-if="filter !== ''">  
 <span></span>  
 </th>  
  
  
 </tr>  
 </thead>  
 <tbody>  
 <tr dusk="1255-row" v-for="(object, key) in tableData" :key="key" v-if="filter === ''">  
 <td>  
 <div class="text-left text-left">  
 <span class="whitespace-no-wrap">{{ object.name }}</span>  
 </div>  
 </td>  
 <td>  
 <div class="text-left text-left">  
 <div>  
 <span class="whitespace-no-wrap" v-if="object.days\_count != null">{{ object.days\_count }}</span>  
 <span class="whitespace-no-wrap" v-else>0</span>  
 </div>  
 </div>  
 </td>  
  
 <td>  
 <div class="text-left text-left">  
 <span class="whitespace-no-wrap" v-if="object.do\_not\_count != null">{{ object.do\_not\_count }}</span>  
 <span class="whitespace-no-wrap" v-else>0</span>  
 </div>  
 </td>  
  
 </tr>  
  
 <tr dusk="1255-row" v-for="(object, key) in filteredRows" :key="key" v-if="filter !== ''">  
 <td>  
 <div class="text-left text-left">  
 <span class="whitespace-no-wrap">{{ object.start\_date }}</span>  
 </div>  
 </td>  
 <td>  
 <div class="text-left text-left">  
 <span class="whitespace-no-wrap">{{ object.end\_date }}</span>  
 </div>  
 </td>  
 <td>  
 <div class="text-left text-left">  
 <div>  
 <span class="whitespace-no-wrap" v-if="object.days\_count != null">{{ object.days\_count }}</span>  
 <span class="whitespace-no-wrap" v-else>0</span>  
 </div>  
 </div>  
 </td>  
  
 <td>  
 <div class="text-left text-left">  
 <span class="whitespace-no-wrap" v-if="object.do\_not\_count != null">{{ object.do\_not\_count }}</span>  
 <span class="whitespace-no-wrap" v-else>0</span>  
 </div>  
 </td>  
  
 <td>  
 <div class="text-left text-left">  
 <span class="whitespace-no-wrap">{{ object.description }}</span>  
 </div>  
 </td>  
 <td>  
 <div class="text-left text-left">  
 <span class="whitespace-no-wrap">{{ object.created\_datetime }}</span>  
 </div>  
 </td>  
 <td class="td-fit text-right pr-6 align-middle">  
 <div class="inline-flex items-center">  
 <span class="inline-flex">  
 <a href="/resources/clients/1255" class="cursor-pointer text-70 hover:text-primary mr-3 inline-flex items-center has-tooltip" data-testid="clients-items-0-destroy-button" dusk="delete-button" data-original-title="null">  
 <svg xmlns="http://www.w3.org/2000/svg" width="22" height="18" viewBox="0 0 22 16" aria-labelledby="destroy" role="presentation" class="fill-current">  
 <path d="M16.56 13.66a8 8 0 0 1-11.32 0L.3 8.7a1 1 0 0 1 0-1.42l4.95-4.95a8 8 0 0 1 11.32 0l4.95 4.95a1 1 0 0 1 0 1.42l-4.95 4.95-.01.01zm-9.9-1.42a6 6 0 0 0 8.48 0L19.38 8l-4.24-4.24a6 6 0 0 0-8.48 0L2.4 8l4.25 4.24h.01zM10.9 12a4 4 0 1 1 0-8 4 4 0 0 1 0 8zm0-2a2 2 0 1 0 0-4 2 2 0 0 0 0 4z"></path>  
 </svg>  
 </a>  
 </span>  
 <span class="inline-flex">  
 <a :href="'/sickdays/edit/'+object.ID" class="inline-flex cursor-pointer text-70 hover:text-primary mr-3 has-tooltip" dusk="edit-button" data-original-title="null">  
 <svg xmlns="http://www.w3.org/2000/svg" width="20" height="20" viewBox="0 0 20 20" aria-labelledby="edit" role="presentation" class="fill-current">  
 <path d="M4.3 10.3l10-10a1 1 0 0 1 1.4 0l4 4a1 1 0 0 1 0 1.4l-10 10a1 1 0 0 1-.7.3H5a1 1 0 0 1-1-1v-4a1 1 0 0 1 .3-.7zM6 14h2.59l9-9L15 2.41l-9 9V14zm10-2a1 1 0 0 1 2 0v6a2 2 0 0 1-2 2H2a2 2 0 0 1-2-2V4c0-1.1.9-2 2-2h6a1 1 0 1 1 0 2H2v14h14v-6z"></path>  
 </svg></a>  
 </span>  
 </div>  
 </td>  
 </tr>  
 </tbody>  
 </table>  
 <div v-if="filter !== ''">  
 <div class="flex border-b border-40">  
 <div class="flex border-b border-40 w-full">  
 <div class="w-1/5 py-6 px-8">  
 <label class="inline-block text-80 pt-2 leading-tight">Total days count: </label>  
 </div>  
 <div class="py-6 px-8 w-4/5">  
 <span>{{filteredRows.reduce((a, b) => a + parseFloat(b.days\_count),0)}}</span>  
 </div>  
 </div>  
 </div>  
 <div class="flex border-b border-40">  
 <div class="flex border-b border-40 w-full">  
 <div class="w-1/5 py-6 px-8">  
 <label class="inline-block text-80 pt-2 leading-tight">Total days not count: </label>  
 </div>  
 <div class="py-6 px-8 w-4/5">  
 <span>{{filteredRows.reduce((a, b) => a + parseFloat(b.do\_not\_count),0)}}</span>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
</template>  
  
<script>  
export default {  
 created() {  
 this.requestTableData();  
 this.requestSickData();  
  
 },  
  
 data() {  
 return {  
 filter: '',  
 tableData: [  
 {userID: ''},  
  
 ],  
 SickData: [  
 {employee\_id: ''},  
 ],  
  
 }  
 },  
  
  
 methods: {  
 requestTableData() {  
 Nova.request().get('/nova-vendor/sickdays/complete')  
 .then((response) => {  
 this.tableData = response.data;  
  
 })  
 },  
 requestSickData() {  
 Nova.request().get('/nova-vendor/sickdays/sickdays')  
 .then((response) => {  
 this.SickData = response.data;  
  
  
 })  
 },  
  
  
  
  
 },  
  
 computed:{  
 filteredRows() {  
 return this.SickData.filter(row => {  
 const employees = row.employee\_id.toString().toLowerCase();  
 const searchTerm = this.filter.toLowerCase();  
  
 return employees === searchTerm;  
 });  
 }  
  
 },  
 mounted() {  
 //  
 },  
}  
</script>  
  
<style>  
/\* Scoped Styles \*/  
</style>

NewDay.vue

<template>  
 <div>  
 <div><h2>Add sick day</h2></div>  
 <form @submit.prevent="save()" class="resource-form">  
 <div class="card">  
  
 <div class="flex border-b border-40">  
 <div class="flex border-b border-40 w-full">  
 <div class="w-1/5 py-6 px-8">  
 <label class="inline-block text-80 pt-2 leading-tight" for="Employee">Employee</label>  
 </div>  
 <div class="py-6 px-8 w-4/5">  
 <select class="form-control form-select w-full"  
 name="ID" id="Employee" v-model="fields.employee\_id">  
 <option v-for="(object, key) in EmployeeData" :key="key" :value="object.ID">{{ object.name }}  
 </option>  
 </select>  
 </div>  
 </div>  
 </div>  
  
 <div class="flex border-b border-40">  
 <div class="flex border-b border-40 w-full">  
 <div class="w-1/5 py-6 px-8">  
 <label class="inline-block text-80 pt-2 leading-tight" for="start\_date">Start date</label>  
 </div>  
 <div class="py-6 px-8 w-4/5">  
 <datepicker class="w-full form-control form-input form-input-bordered" id="start\_date" name="start\_date" v-model="fields.start\_date" required></datepicker>  
 </div>  
 </div>  
 </div>  
 <div class="flex border-b border-40">  
 <div class="flex border-b border-40 w-full">  
 <div class="w-1/5 py-6 px-8">  
 <label class="inline-block text-80 pt-2 leading-tight" for="end\_date">End date</label>  
 </div>  
 <div class="py-6 px-8 w-4/5">  
 <datepicker class="w-full form-control form-input form-input-bordered" id="end\_date" name="end\_date" v-model="fields.end\_date" required></datepicker>  
 </div>  
 </div>  
 </div>  
 <div class="flex border-b border-40">  
 <div class="flex border-b border-40 w-full">  
 <div class="w-1/5 py-6 px-8">  
 <label for="description" class="inline-block text-80 pt-2 leading-tight" >Description</label>  
 </div>  
 <div class="py-6 px-8 w-4/5">  
 <input type="text" id="description" name="description" v-model="fields.description" class="w-full form-control form-input form-input-bordered">  
 </div>  
 </div>  
 </div>  
 <div class="flex border-b border-40">  
 <div class="flex border-b border-40 w-full">  
 <div class="w-1/5 py-6 px-8">  
 <label class="inline-block text-80 pt-2 leading-tight" for="days\_count">Days</label>  
 </div>  
 <div class="py-6 px-8 w-4/5">  
 <input class="w-full form-control form-input form-input-bordered" name="days\_count" id="days\_count" type="number" min="0" required v-model="fields.days\_count">  
 </div>  
 </div>  
 </div>  
  
 <div class="flex border-b border-40 w-full">  
 <div class="w-1/5 py-6 px-8">  
 <label for="dont\_count" class="inline-block text-80 pt-2 leading-tight">Do not use in global count.</label>  
 </div>  
 <div class="py-6 px-8 w-1/2">  
 <input type="checkbox" class="checkbox mt-2" id="dont\_count" name="dont\_count" v-model="fields.dont\_count">  
 </div>  
 </div>  
 </div>  
 <div class="flex items-center form-footer mt-6">  
 <a tabindex="0" class="btn btn-link dim cursor-pointer text-80 ml-auto mr-6" v-on:click="cancel()">Cancel</a>  
 <button type="submit" class="btn btn-default btn-primary inline-flex items-center relative">  
 <span class="">Save</span>  
 </button>  
 </div>  
 </form>  
 </div>  
  
</template>  
  
<script>  
import ***Datepicker*** from 'vuejs-datepicker';  
  
  
export default {  
 name: "newday",  
  
  
  
  
  
 components: {  
 ***Datepicker***,  
 },  
  
 created() {  
 this.requestEmployeeData();  
 this.requestSickData();  
 // this.authUser;  
  
  
  
  
 },  
  
 data() {  
 return {  
 EmployeeData: {},  
 SickData: {},  
 fields: {  
 employee\_id: '',  
 start\_date: '',  
 end\_date: '',  
 description: '',  
 days\_count: '0',  
 do\_not\_count: '0',  
 dont\_count: false,  
 // authuser: '',  
  
 },  
  
  
 }  
 },  
  
 methods: {  
 requestEmployeeData() {  
 Nova.request().get('/nova-vendor/sickdays/employees')  
 .then((response) => {  
 this.EmployeeData = response.data;  
  
 })  
 },  
 requestSickData() {  
 Nova.request().get('/nova-vendor/sickdays/sickdays')  
 .then((response) => {  
 this.SickData = response.data;  
  
  
 })  
 },  
 save() {  
 ***console***.log(this.fields)  
 if (this.fields.dont\_count === false){  
 axios.post('/sickdays/save', this.fields).then(response => {  
 return ***window***.location.href = '/sickdays';  
 });  
 }else{  
 this.fields.do\_not\_count = this.fields.days\_count;  
 this.fields.days\_count = 0  
 axios.post('/sickdays/save', this.fields).then(response => {  
 return ***window***.location.href = '/sickdays';  
 });  
 }  
 },  
  
 cancel(){  
 return ***window***.location.href = '/sickdays';  
 },  
  
 },  
  
 mounted() {  
 //  
 },  
}  
</script>  
  
<style scoped>  
  
</style>

# Api.php

Route::get('/employees', function (Request $request) {  
 $employees = DB::table('employees')->select('ID','name')->get();  
  
 return $employees;  
});  
  
Route::get('/sickdays', function (Request $request) {  
 $sickdays = DB::table('hrm.illness\_absence')  
 ->select(DB::raw('ID,employee\_id,start\_date,end\_date,description,days\_count,created\_by,do\_not\_count,created\_datetime'))->get();  
  
 return $sickdays;  
});  
  
Route::get('/complete', function (Request $request){  
 $complete = DB::table('employees')  
 ->select(DB::raw('sum(days\_count) as days\_count,sum(do\_not\_count) as do\_not\_count,name,employees.ID as userID'))  
 ->leftJoin('hrm.illness\_absence','employees.ID', '=', 'employee\_id')  
 ->groupBy('name','employees.ID')->orderBy('name')->get();  
  
 return $complete;  
});

# Tool.js

{  
 name: 'sickdays',  
 path: '/sickdays/edit/:ID',  
 component: require('./components/edit'),  
},