Додаток Д

Код головної сторінки

<template>

<v-layout

wrap

>

<!--Login dialog--->

<v-dialog v-model="dialog" persistent max-width="600px">

<v-card>

<v-card-title>

<span class="headline">Профіль користувача</span>

</v-card-title>

<v-card-text>

<v-container grid-list-md>

<v-layout wrap>

<v-flex xs12>

<v-text-field label="Ім'я користувача\*" v-model = "username" required></v-text-field>

</v-flex>

<v-flex xs12>

<v-text-field label="Пароль\*" v-model = "password" type="password" required></v-text-field>

</v-flex>

</v-layout>

</v-container>

</v-card-text>

<v-card-actions>

<v-spacer></v-spacer>

<v-btn color="blue darken-1" flat @click="dialog = false">Закрити</v-btn>

<v-btn color="blue darken-1" flat @click="execLogin">Ввійти</v-btn>

</v-card-actions>

</v-card>

</v-dialog>

<!--Edit dialog-->

<v-dialog v-model="editDialog" persistent max-width="600px">

<v-card>

<v-card-title>

<span class="headline">Edit book</span>

</v-card-title>

<v-card-text>

<v-container grid-list-md>

<v-layout wrap>

<p>{{editDialogData.cardId}}</p>

<v-flex xs12>

<v-text-field label="bookNm\*" v-model = "editDialogData.bookNm"></v-text-field>

</v-flex>

<v-flex xs12>

<v-text-field label="author\*" v-model = "editDialogData.author"></v-text-field>

</v-flex>

<v-flex xs12>

<v-text-field label="bklang\*" v-model = "editDialogData.bklang"></v-text-field>

</v-flex>

<v-flex xs12>

<v-text-field label="tomeNm\*" v-model = "editDialogData.tomeNm"></v-text-field>

</v-flex>

<v-flex xs12>

<v-text-field label="quants\*" v-model = "editDialogData.quants"></v-text-field>

</v-flex>

<v-flex xs12>

<v-text-field label="bk\_MON\*" v-model = "editDialogData.bk\_MON"></v-text-field>

</v-flex>

<v-flex xs12>

<v-text-field label="speclt\*" v-model = "editDialogData.speclt"></v-text-field>

</v-flex>

<v-flex xs12>

<v-text-field label="depart\*" v-model = "editDialogData.depart"></v-text-field>

</v-flex>

<v-flex xs12>

<v-text-field label="bKyear\*" v-model = "editDialogData.bKyear"></v-text-field>

</v-flex>

<v-flex xs12>

<v-text-field label="prices\*" v-model = "editDialogData.prices"></v-text-field>

</v-flex>

</v-layout>

</v-container>

</v-card-text>

<v-card-actions>

<v-spacer></v-spacer>

<v-btn color="blue darken-1" flat @click="editDialog = false">Close</v-btn>

<v-btn color="blue darken-1" flat @click="editBook(id)">Save</v-btn>

</v-card-actions>

</v-card>

</v-dialog>

<!--tabs container-->

<v-container>

<v-toolbar tabs color="green darken-2">

<v-toolbar-side-icon @click.stop="drawer = !drawer"></v-toolbar-side-icon>

<v-toolbar-title>Book library</v-toolbar-title>

<v-spacer></v-spacer>

<v-btn @click="updateBooks"> Оновити <v-icon>refresh</v-icon> </v-btn>

<template v-slot:extension>

<v-tabs

v-model="tab"

align-with-title

color="green darken-2"

>

<v-tabs-slider color="white"></v-tabs-slider>

<v-tab v-for="item in tabitems" :key="item.title">

{{ item.title }}

</v-tab>

</v-tabs>

</template>

</v-toolbar>

<v-tabs-items v-model="tab">

<v-tab-item v-for="item in tabitems" :key="item.title">

<v-card flat color="green lighten-1">

<!--SEttings--->

<v-form

ref="form"

v-model="valid"

lazy-validation

v-if="item.title=='Налаштування'"

>

<v-text-field

v-model="name"

:counter="10"

:rules="nameRules"

label="Name"

required

></v-text-field>

<v-text-field

v-model="email"

:rules="emailRules"

label="E-mail"

required

></v-text-field>

<v-checkbox

v-model="checkbox"

:rules="[v => !!v || 'You must agree to continue!']"

label="Do you agree?"

required

></v-checkbox>

<v-btn

:disabled="!valid"

color="success"

@click="validate"

>

Validate

</v-btn>

<v-btn

color="error"

@click="reset"

>

Reset Form

</v-btn>

<v-btn

color="warning"

@click="resetValidation"

>

Reset Validation

</v-btn>

</v-form>

<!--End settings--->

<!--Book list--->

<v-expansion-panel

expand

v-if="item.title=='Список книг'"

>

<v-expansion-panel-content

v-for="(item, index) in pagedBooks[currentPage]"

:key="item.cardId+index"

color="green lighten-1"

>

<v-flex xs12>

<v-card color="green lighten-1" class="white--text">

<v-layout row>

<v-flex xs5>

<v-img

src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcSZ2wROYyJwi8EDkr9F6BfEyEJ5j0pzybXsk4Jxw3-IeiSDbJAn1Q"

aspect-ratio="3"

contain

></v-img>

</v-flex>

<v-flex xs7>

<v-card-title primary-title>

<div>

<div class="headline">{{item.bookNm}} - {{item.author}}</div>

<div>Номер тому: {{item.tomeNm}}</div>

<div>Кількість книг: {{item.quants}}</div>

<div>Ціна: {{item.prices}}</div>

<div>Рік випуску: {{item.bKyear}}</div>

<div>Кількість: {{item.quants}}</div>

<div>Мова: {{item.bklang}}</div>

<div>МОН: {{item.bk\_MON}}</div>

<div>Спеціальність: {{item.speclt}}</div>

<div>Відділ: {{item.depart}}</div>

<v-btn v-if=loggedIn :key="item.cardId+index" @click="addBook(item.cardId)">Додати до видачі</v-btn>

<v-btn v-if=loggedIn :key="item.cardId+index+index" @click.stop="editDialog = true;

editDialogData.slug = item.slug,

editDialogData.bookNm= item.bookNm,

editDialogData.author= item.author,

editDialogData.bklang= item.bklang,

editDialogData.tomeNm= item.tomeNm,

editDialogData.quants= item.quants,

editDialogData.bk\_MON= item.bk\_MON,

editDialogData.speclt= item.speclt,

editDialogData.depart= item.depart,

editDialogData.bKyear= item.bKyear,

editDialogData.prices= item.prices

">Редагувати книгу</v-btn>

</div>

</v-card-title>

</v-flex>

</v-layout>

<v-divider light></v-divider>

</v-card>

</v-flex>

<template v-slot:header>

<div>{{item.bookNm}} - {{item.author}}</div>

</template>

</v-expansion-panel-content>

<v-pagination

v-model="currentPage"

:length="pages"

:total-visible="7"

>

</v-pagination>

</v-expansion-panel>

<!--Book upload--->

<v-card v-if="item.title=='Завантаження книг'">

<v-container

fluid

grid-list-md

>

<form action="http://localhost:3000/api/uploads" enctype="multipart/form-data" method="post">

<input type="file" name="upload" multiple>

<input type="submit" value="Завантажити" style = "background: #016ABC;

color: #fff;

border: 1px solid #eee;

border-radius: 20px;

box-shadow: 5px 5px 5px #eee;text-shadow:none;">

</form>

<v-layout row wrap>

<v-flex

v-for="(upload,index) in uploads"

:key="upload.bookNm+index"

v-bind="{ [`xs4`]: true }"

>

<v-card>

<v-container

fill-height

fluid

pa-2

>

<v-layout fill-height>

<v-flex xs12 align-end flexbox>

<v-img

src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcSZ2wROYyJwi8EDkr9F6BfEyEJ5j0pzybXsk4Jxw3-IeiSDbJAn1Q"

aspect-ratio="3"

contain

></v-img>

<span class="text-truncate headline " >{{upload.bookNm}}</span>

<div v-if="loggedIn" class="text-truncate "><a v-bind:href="downloadLink+upload.bookNm">Завантажити</a></div>

</v-flex>

</v-layout>

</v-container>

<v-card-actions>

</v-card-actions>

</v-card>

</v-flex>

</v-layout>

</v-container>

</v-card>

<!--Vydacha--->

<v-expansion-panel

expand

v-if="item.title=='Видача книг'"

>

<v-expansion-panel-content

v-for="(item,index) in gaveBooks"

:key="item.slug"

>

<template v-slot:header>

<div>{{item.bookNm}} - {{item.author}}</div>

</template>

<v-card color="green lighten-1" class="white--text">

<v-layout row>

<v-flex xs5>

<v-img

src="https://cdn.vuetifyjs.com/images/cards/halcyon.png"

height="125px"

contain

></v-img>

</v-flex>

<v-flex xs7>

<v-card-title primary-title>

<div>

<div class="headline">{{item.bookNm}} - {{item.author}}</div>

<div>Номер тому: {{item.tomeNm}}</div>

<div>Кількість книг: {{item.quants}}</div>

<div>Ціна: {{item.prices}}</div>

<div>Рік випуску: {{item.bKyear}}</div>

<div>Кількість: {{item.quants}}</div>

<div>Мова: {{item.bklang}}</div>

<div>МОН: {{item.bk\_MON}}</div>

<div>Спеціальність: {{item.speclt}}</div>

<div>Відділ: {{item.depart}}</div>

<v-btn :key="item.cardId+index" @click="removeBook(item.cardId)">Віддати книгу</v-btn>

</div>

</v-card-title>

</v-flex>

</v-layout>

<v-divider light></v-divider>

</v-card>

</v-expansion-panel-content>

</v-expansion-panel>

</v-card>

</v-tab-item>

</v-tabs-items>

</v-container>

<!--NAVIGATION--->

<v-navigation-drawer v-model="drawer" absolute temporary >

<!--<v-btn href="http://pornhub.com" target="\_blank" flat dark small color="white">

<v-icon dark>favorite</v-icon>

</v-btn> -->

<v-list class="pa-1">

<v-list-tile avatar>

<v-list-tile-avatar>

<img v-if="loggedIn" src="https://randomuser.me/api/portraits/men/85.jpg">

</v-list-tile-avatar>

<v-list-tile-content>

<v-list-tile-title>{{username}}</v-list-tile-title>

</v-list-tile-content>

</v-list-tile>

</v-list>

<v-list class="pt-0" dense>

<v-divider></v-divider>

<v-list-tile

v-for="item in items"

:key="item.title"

>

<v-list-tile-action>

<v-icon>{{ item.icon }}</v-icon>

</v-list-tile-action>

<v-list-tile-content>

<v-list-tile-title>{{ item.title }}</v-list-tile-title>

</v-list-tile-content>

</v-list-tile>

<v-btn

color="primary"

dark

@click.stop="dialog = true"

>

Ввійти

</v-btn>

</v-list>

</v-navigation-drawer>

</v-layout>

</template>

<script>

import axios from 'axios'

import { constants } from 'crypto';

export default {

data () {

return {

editBookDialog:false,

searchText: '',

downloadLink: 'http://localhost:3000/api/uploads/download/',

apiLink: 'http://localhost:3000/api',

drawer: null,

text: '',

kindaFile: '',

items: [

{ title: 'Додому', icon: 'dashboard' },

{ title: 'Про автора', icon: 'question\_answer' }

],

tab: null,

tabitems: [

{title:'Список книг',content: '1'}

,{title: 'Завантаження книг',content: '2'},

],

dialog: false,

editDialog: false,

editDialogData:{

slug: null,

bookNm: null,

author: null,

bklang: null,

tomeNm: null,

quants: null,

bk\_MON: null,

speclt: null,

depart: null,

bKyear: null,

prices: null,

},

password: '',

valid: true,

name: '',

nameRules: [

v => !!v || 'Name is required',

v => (v && v.length <= 10) || 'Name must be less than 10 characters'

],

email: '',

username:'',

loggedIn:false,

emailRules: [

v => !!v || 'E-mail is required',

v => /.+@.+/.test(v) || 'E-mail must be valid'

],

select: null,

checkbox: false,

info: null,

gaveBooks: [],

test: null,

books: null,

currentPage: 1,

pages: 1,

booksPerPage: 10,

pagedBooks: [],

uploads: null,

editData:{

bookNm: null,

author: null,

bklang: null,

tomeNm: null,

quants: null,

bk\_MON: null,

speclt: null,

depart: null,

bKyear: null,

prices: null,

}

}

},

mounted() {

axios

.get(this.apiLink+'/books',)

.then(response => {this.books = response.data.books

var i,j,temparray,chunk = 15;

for (i=0,j=this.books.length; i<j; i+=chunk) {

temparray = this.books.slice(i,i+chunk);

this.pagedBooks.push(temparray)

}

this.pages = this.pagedBooks.length-1;

});

axios

.get(this.apiLink+'/uploads',)

.then(response => {this.uploads = response.data.uploads

});

},

methods: {

search(searchText){

axios

.post(this.apiLink+'/books/search',{"search":searchText}).then(

response => {this.books = response.data.books;

var i,j,temparray,chunk = 15;

this.pagedBooks = [];

for (i=0,j=this.books.length; i<j; i+=chunk) {

temparray = this.books.slice(i,i+chunk);

this.pagedBooks.push(temparray)

}

this.pages = this.pagedBooks.length-1;

}

)

},

editBook(cardId){

axios

.post(this.apiLink+'/books/'+slug,{

bookNm:this.editData.bookNm,

author:this.editData.author,

bklang:this.editData.bklang,

tomeNm:this.editData.tomeNm,

quants:this.editData.quants,

bk\_MON:this.editData.bk\_MON,

speclt:this.editData.speclt,

depart:this.editData.depart,

bKyear:this.editData.bKyear,

prices:this.editData.prices

})

},

setLink(){

this.link = "http://pornhub.com"

},

saveFile(){

axios

.post(this.apiLink+'/uploads/upload',{"file":this.kindaFile})

},

processFile(event) {

this.kindaFile = event.target.files[0]

},

nextPage(){

this.currentPage++;

},

prevPage(){

this.currentPage--;

},

execLogin(){

this.dialog = false;

this.loggedIn = true;

this.tabitems = [

{title:'Список книг',content: '1'}

,{title: 'Завантаження книг',content: '2'}, {title:'Видача книг',content: '3'},

]

axios

.post(this.apiLink+'/gavebooks',{"username":this.username})

.then(response => {this.gaveBooks = response.data.books;

var i,j,temparray,chunk = 15;

this.pagedBooks = [];

for (i=0,j=this.books.length; i<j; i+=chunk) {

temparray = this.books.slice(i,i+chunk);

this.pagedBooks.push(temparray)

}

this.pages = this.pagedBooks.length-1;

});

},

addBook(id){

var url = this.apiLink+'/gavebooks/'+id;

var deleteFromRegularList = this.apiLink+'/books/'+id;

axios

.post(url,{"author":this.username})

.then(response => {if(response.data.books){

axios

.post(this.apiLink+'/gavebooks',{"username":this.username})

.then(response => {this.gaveBooks = response.data.books;

axios

.delete(deleteFromRegularList).then(response=>{

axios

.get(this.apiLink+'/books',)

.then(response => {this.books = response.data.books;

var i,j,temparray,chunk = 15;

this.pagedBooks = [];

for (i=0,j=this.books.length; i<j; i+=chunk) {

temparray = this.books.slice(i,i+chunk);

this.pagedBooks.push(temparray)

}

this.pages = this.pagedBooks.length-1;

});

})

});

}});

},

removeBook(id){

var removeGaveBookUlr = this.apiLink+'/gavebooks/'+id+'/remove';

var deleteGaveBookUlr = this.apiLink+'/gavebooks/'+id+'/delete';

axios

.get(removeGaveBookUlr,{"username":this.username})

.then(response => {if(response){

axios

.delete(deleteGaveBookUlr,{"username":this.username}).then(response=>{

axios

.post(this.apiLink+'/gavebooks',{"username":this.username})

.then(response => {this.gaveBooks = response.data.books;

axios

.get(this.apiLink+'/books',)

.then(response => (this.books = response.data.books));

var i,j,temparray,chunk = 15;

this.pagedBooks = [];

for (i=0,j=this.books.length; i<j; i+=chunk) {

temparray = this.books.slice(i,i+chunk);

this.pagedBooks.push(temparray)

}

this.pages = this.pagedBooks.length-1;})

})

}});

},

updateBooks(){

axios

.post(this.apiLink+'/gavebooks',{"username":this.username})

.then(response => {this.gaveBooks = response.data.books;

axios

.get(this.apiLink+'/books',)

.then(response => {this.books = response.data.books;

var i,j,temparray,chunk = 15;

this.pagedBooks = [];

for (i=0,j=this.books.length; i<j; i+=chunk) {

temparray = this.books.slice(i,i+chunk);

this.pagedBooks.push(temparray)

}

this.pages = this.pagedBooks.length-1;});

})

},

validate () {

if (this.$refs.form.validate()) {

this.snackbar = true

}

},

reset () {

this.$refs.form.reset()

},

resetValidation () {

this.$refs.form.resetValidation()

}

}

}

</script>

<!-- Add "scoped" attribute to limit CSS to this component only -->

<style scoped>

</style>