Приложение 1

Листинг – NewsCreateModal.vue

|  |
| --- |
| <template> |
|  | <b-modal |
|  | id="modal-create-news" |
|  | ref="modal" |
|  | title="Создание новости" |
|  | centered |
|  | @show="resetModal" |
|  | @shown="focusMyElement" |
|  | @hidden="resetModal" |
|  | @ok="handleOk" |
|  | @change="$emit('input', $event)" |
|  | v-model="value" |
|  | > |
|  | <form |
|  | ref="form" |
|  | @submit.stop.prevent="handleSubmit" |
|  | > |
|  | <b-form-group |
|  | label="Заголовок новости" |
|  | label-for="input-1" |
|  | :invalid-feedback="titleState" |
|  | > |
|  | <b-form-input |
|  | id="input-1" |
|  | ref="focusThis" |
|  | v-model="title" |
|  | :state="!titleState" |
|  | required |
|  | placeholder="Введите заголовок" |
|  | /> |
|  | </b-form-group> |
|  | <b-form-group |
|  | label="Текст новости" |
|  | label-for="input-2" |
|  | :invalid-feedback="textState" |
|  | > |
|  | <b-form-input |
|  | id="input-2" |
|  | v-model="text" |
|  | :state="!textState" |
|  | required |
|  | placeholder="Введите текст новости" |
|  | /> |
|  | </b-form-group> |
|  | <b-form-group |
|  | label="Дата новости" |
|  | label-for="input-3" |
|  | :invalid-feedback="dateState" |
|  | > |
|  | <b-form-datepicker |
|  | class="mb-2" |
|  | id="input-3" |
|  | v-model="date" |
|  | :state="!dateState" |
|  | /> |
|  | </b-form-group> |
|  | <b-form-group |
|  | label="Дата новости" |
|  | label-for="input-3" |
|  | :invalid-feedback="dateState" |
|  | > |
|  | <b-form-file |
|  | v-model="image" |
|  | :state="Boolean(image)" |
|  | accept=".jpg, .png" |
|  | placeholder="Выберите фаил или перенесите его" |
|  | drop-placeholder="Перенесите его фаил сюда..." |
|  | /> |
|  | </b-form-group> |
|  | </form> |
|  | <template v-slot:modal-footer> |
|  | <div class="w-100"> |
|  | <b-button |
|  | variant="success" |
|  | class="float-right" |
|  | type="submit" |
|  | :disabled="isWaitingServerResponse" |
|  | @click="handleSubmit" |
|  | > |
|  | <b-spinner v-if="isWaitingServerResponse" small type="grow"></b-spinner> |
|  | Сохранить |
|  | </b-button> |
|  | <b-button |
|  | variant="secondary" |
|  | class="float-right mr-2" |
|  | :disabled="isWaitingServerResponse" |
|  | @click="$root.$emit('bv::hide::modal', 'modal-create-news', '#btn-show-create-news')" |
|  | > |
|  | Закрыть |
|  | </b-button> |
|  | </div> |
|  | </template> |
|  | </b-modal> |
|  | </template> |
|  | <script> |
|  | import { uploadArticle } from '@/api/news'; |
|  | export default { |
|  | props: { |
|  | value: { |
|  | type: Boolean, |
|  | required: true, |
|  | }, |
|  | }, |
|  | data: () => ({ |
|  | title: '', |
|  | image: null, |
|  | text: '', |
|  | date: new Date(), |
|  | isWaitingServerResponse: false, |
|  | }), |
|  | computed: { |
|  | titleState() { |
|  | if (this.title.length < 3) { |
|  | return 'Длина заголовка не может быть меньше 3 символов'; |
|  | } |
|  | return ''; |
|  | }, |
|  | textState() { |
|  | if (this.text.length < 3) { |
|  | return 'Длина контента не может быть меньше 10 символов'; |
|  | } |
|  | return ''; |
|  | }, |
|  | dateState() { |
|  | if (!this.date) { |
|  | return 'Вы забыли выбрать дату'; |
|  | } |
|  | return ''; |
|  | }, |
|  | }, |
|  | methods: { |
|  | resetModal() { |
|  | this.title = ''; |
|  | this.image = null; |
|  | this.text = ''; |
|  | this.date = new Date(); |
|  | }, |
|  | handleOk(bvModalEvt) { |
|  | // Prevent modal from closing |
|  | bvModalEvt.preventDefault(); |
|  | // Trigger submit handler |
|  | this.handleSubmit(); |
|  | }, |
|  | handleSubmit() { |
|  | // Exit when the form isn't valid |
|  | if (!this.$refs.form.checkValidity()) { |
|  | return; |
|  | } |
|  | this.isWaitingServerResponse = true; |
|  | uploadArticle({ |
|  | title: this.title, |
|  | text: this.text, |
|  | date: this.date, |
|  | image: this.image, |
|  | }) |
|  | .then(({ data }) => { |
|  | this.$emit('created', data); |
|  | }) |
|  | .catch((e) => { |
|  | alert('Произошла ошибка при создании новой новости'); |
|  | console.log(e); |
|  | }) |
|  | .finally(() => { |
|  | this.isWaitingServerResponse = false; |
|  | }); |
|  | }, |
|  | focusMyElement() { |
|  | this.$refs.focusThis.focus(); |
|  | }, |
|  | }, |
|  | }; |
|  | </script> |

Приложение 2

Листинг – news.js

|  |
| --- |
| import Router from 'koa-router'; |
| import { uploadFileToFirebase, removeFileFromFirebase } from '../../utils/Firebase/Storage'; |
| import News from '../../db/models/News'; |
| import permission from '../../middlwares/permission'; |
| const router = new Router(); |
| export default router |
| .put('/upload/image', permission(['superadmin', 'admin', 'moderator']), async (ctx) => { |
| const { joiValidValues } = ctx.state; |
| const article = await News.findById(joiValidValues.\_id); |
| if (article.image.Key) { |
| await removeFileFromFirebase(article.image.Key); |
| } |
| const image = await uploadFileToFirebase(ctx.request.files.image); |
| ctx.body = await News.findByIdAndUpdate(joiValidValues.\_id, { image }, { new: true }); |
| }) |
| /\*\* |
| \* EDIT article of news |
| \* @body { ?AWSS3Image:oldImage, String:title, String:text, Date:date, Boolean:isEditedImage, ?FormData:image } |
| \*/ |
| .put('/upload/text', permission(['superadmin', 'admin', 'moderator']), async (ctx) => { |
| ctx.body = await News.findOneAndUpdate({ \_id: ctx.state.joiValidValues.\_id }, { |
| title: ctx.state.joiValidValues.title, |
| text: ctx.state.joiValidValues.text, |
| date: ctx.state.joiValidValues.date, |
| }); |
| }) |
| /\*\* |
| \* ADD new article of news |
| \* @body { ?FormData:image, String:title, String:text, ?Date:date } |
| \* |
| \* @return |
| \* 200 - <news> |
| \* xxx - |
| \*/ |
| .post('/upload/text', permission(['superadmin', 'admin', 'moderator']), async (ctx) => { |
| ctx.body = await News.create({ |
| title: ctx.state.joiValidValues.title, |
| text: ctx.state.joiValidValues.text, |
| date: ctx.state.joiValidValues.date, |
| }); |
| }) |
| /\*\* |
| \* DELETE article from news |
| \* @body { Number: id } |
| \* |
| \* @return 200|404 |
| \*/ |
| .delete('/', permission(['superadmin', 'admin', 'moderator']), async (ctx) => { |
| const { \_id } = ctx.state.joiValidValues; |
| const article = await News.findById(\_id); |
| if (!article) { |
| return ctx.throw(404); |
| } |
| /\* remove image from AWS \*/ |
| await removeFileFromFirebase(article.image.Key); |
| await News.deleteOne({ |
| \_id, |
| }); |
| }); |