

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Фронт-энд разработка

Отчет

Лабораторная работа 3

Выполнила:

Самчук Анита

К33402

Проверил:

Добряков Д. И.

Санкт-Петербург

2023 г.

Задача

Разработка одностраничного веб-приложения (SPA) с использованием фреймворка Vue.JS

Ход работы

Для работы с бутстрапом был установлен плагин bootstrap-vue

```
import { BootstrapVue } from 'bootstrap-vue'

import 'bootstrap/dist/css/bootstrap.css'
import 'bootstrap-vue/dist/bootstrap-vue.css'
import './assets/css/login.css'
import './assets/css/main.css'
import './assets/css/styles.css'
import './assets/css/light.css'

Vue.config.productionTip = false

new Vue({
  router,
  store,
  render: h => h(App)
}).$mount('#app')

Vue.use(BootstrapVue)
```

```
"axios": "^1.3.4",
"bootstrap": "^5.2.3",
"bootstrap-vue": "^2.23.1",
```

Некоторые элементы были переписаны, используя bootstrap-vue

```

<b-collapse class="collapse filters mt-2 ml-4" id="collapse">
  <h5>Filters</h5>
  <div class="row mt-3">
    <div class="col-4">
      <h6>Order By</h6>

```

Роутинг реализован с помощью vue-router

```

Vue.use(VueRouter)

const routes = [
  {
    path: '/',
    name: 'Dashboard',
    component: DashboardView
  },
  {
    path: '/login',
    name: 'Login',
    component: LoginView
  },
  {
    path: '/post/:id',
    name: 'Post',
    component: PostView
  },
  {
    path: '/profile',
    name: 'Profile',
    component: ProfileView
  },
  {
    path: '/signup',
    name: 'SignUpView',
    component: SignUpView
  }
]

const router = new VueRouter({
  mode: 'history',

```

Так же добавлен vuex, в сторе хранятся функции загрузки фотографий, и все состояния связанные с фотографиями

```

export default new Vuex.Store({
  state: {
    photos: null,
    postPhoto: null,
    likedPhotos: null
  },

  getters: {
    photos (state) {
      return state.photos
    },

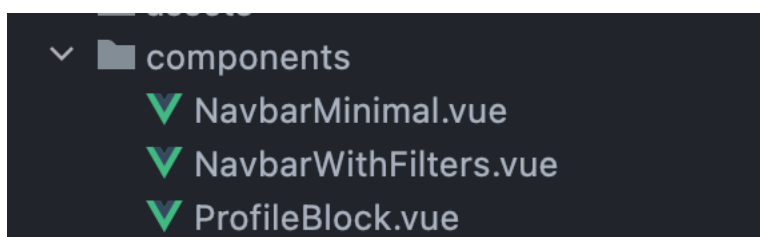
    postPhoto (state) {
      return state.postPhoto
    },

    likedPhotos (state) {
      return state.likedPhotos
    }
  },

  mutations: {
    setPhotos (state, payload) {
      state.photos = payload
    },
  },
})

```

Из повторяющихся элементов вынесены в компоненты разные навбары и блок профиля (логаут и вход на страницу профиля/логин)



На страницах логина, профиля и регистрации в хуке beforeMount проверяется авторизован ли пользователь и происходит редирект, если пользователь не соответствует условиям

```
},  
  
beforeMount () {  
  if (!localStorage.getItem('accessToken')) {  
    this.$router.push('/login')  
  }  
  
  this.loadProfile()  
  this.$store.dispatch('getLikedPhotos', { accessToken: localStorage.getItem('accessToken') })  
},
```

Вывод

Успешно реализована одностраничное приложения для поиска картинок с помощью vue.js