

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего образования

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Факультет инфокоммуникационных технологий
Направление подготовки: 09.03.03 Мобильные и сетевые технологии»

Отчет

по лабораторной работе № 3 «Разработка одностраничного веб-приложения
(SPA) с использованием фреймворка Vue.JS»
по дисциплине «**Фронт-энд разработка**»

Автор: Секен Бексултан [288756]

Группа №К34402

Преподаватель: Добряков Д. И.



Санкт-Петербург, 2023 г.

Задача: перенести написанный ранее сайт на фреймворк Vue.JS.

Ход работы:

1. Роутер

```
1  import Vue from 'vue'
2  import VueRouter from 'vue-router'
3  import Home from '@/views/Home.vue'
4  import Login from '@/views/Log.vue'
5  import Acc from '@/views/Account.vue'
6  import Weather from '@/views/Search.vue'
7
8  Vue.use(VueRouter)
9
10 const routes = [
11   {
12     path: '/',
13     name: 'Home',
14     component: Home
15   },
16   {
17     path: '/login',
18     name: 'Login',
19     component: Login
20   },
21   {
22     path: '/account',
23     name: 'Account',
24     component: Acc
25   },
26   {
27     path: '/search',
28     name: 'Search',
29     component: Weather
30   }
31 ]
```

2. Работа с внешним API

```
46 <script>
47 import axios from 'axios'
48 export default {
49   name: 'Weather',
50   data: () => ({
51     city: '',
52     savedCity: '',
53     info: {
54       main: {
55         temp: '',
56         feels_like: ''
57       },
58       wind: {
59         speed: ''
60       }
61     },
62     location: {
63       coords: {
64         latitude: '',
65         longitude: ''
66       }
67     }
68   }),
69   methods: {
70     submitCity () {
71       localStorage.setItem('city', this.city)
72       this.savedCity = this.city
73       this.$refs.form.reset()
74     },
75     getWeather () {
76       axios
77         .get('https://api.openweathermap.org/data/2.5/weather?q=' + this.city + '&appid=62f76307202d2bbb00f83a4de8ac7393&lang=ru&units=metric')
78         .then(response => (this.info = response.data))
79     },
80     async getLocation () {
81       navigator.geolocation.getCurrentPosition(pos => {
82         this.location = pos
83       })
84       axios
85         .get('https://api.openweathermap.org/data/2.5/weather?lat=' + this.location.coords.latitude + '&lon=' + this.location.coords.longitude + '&appid=62f76307202d2bbb00f83a4de8ac7393&lang=ru&units=metric')
86         .then(response => (this.info = response.data))
87     }
88   }
89 }
90 </script>
```

3. Компоненты

```
1 <template>
2 <header>
3   <nav class="navbar navbar-expand-md navbar-dark">
4     <a class="navbar-brand" href="/">Прогноз погоды</a>
5     <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" data-cy="1" data-ec="1">
6       <span class="navbar-toggler-icon"></span>
7     </button>
8     <div class="collapse navbar-collapse" id="navbarsExampleDefault">
9       <ul class="navbar-nav mr-auto">
10        <li class="nav-item">
11          <a class="nav-link" href="/account">Личный кабинет</a>
12        </li>
13        <li class="nav-item">
14          <a class="nav-link" href="/search">
15            <svg xmlns="http://www.w3.org/2000/svg" width="16" height="16" fill="currentColor" class="bi bi-search" v
16              <path d="M11.742 10.344a6.5 6.5 0 1 0-1.397 1.398h-.001c.03.04.062.078.098.115l3.85 3.85a1 1 0 0 1.
17            </svg>
18            Поиск
19          </a>
20        </li>
21      </ul>
22      <ul class="navbar-nav ml-auto">
23        <li class="nav-item">
24          <a class="nav-link" href="/login">Вход/Регистрация</a>
25        </li>
26      </ul>
27    </div>
28  </nav>
29 </header>
30 </template>
31
32 <script>
33 export default {
34   name: 'Header'
35 }
36 </script>
37
38 <style>
39 .navbar{
40   background-color: #343434;
41 }
42 </style>
43
44 <template>
45   <div class="body">
46     <div class="container form" >
47       <h3>Вход</h3>
48       <div class="mb-3">
49         <label for="exampleInputEmail1" class="form-label">Email</label>
50         <input type="email" class="form-control" id="exampleInputEmail1" aria-describedby="emailHelp">
51       </div>
52       <div class="mb-3">
53         <label for="exampleInputPassword1" class="form-label">Пароль</label>
54         <input type="password" class="form-control" id="exampleInputPassword1">
55       </div>
56       <a href="/account" class="link me-3"><button type="submit" class="btn btn-success">Войти</button></a>
57       <a href="/account" class="link"><button type="submit" class="btn btn-success">Регистрация</button></a>
58     </div>
59   </div>
60 </template>
```

4. Представление

```
1  <template>
2    <div>
3      <Header/>
4      <Login/>
5    </div>
6  </template>
7
8  <script>
9    import Header from '@/components/Header'
10   import Login from '@/components/Login'
11   export default {
12     name: 'Log',
13     components: {
14       Login,
15       Header
16     }
17   }
18  </script>
```

Вывод: в ходе работы, ранее написанный сайт с помощью HTML, CSS, и JS был перенесен на фреймворк Vue.JS. Для работы с внешним API была использована библиотека axios. Поскольку фреймворк Vue.JS использует компоненты подход – страницы сайта были разделены на более мелкие компоненты, которые можно использовать при создании новых страниц.