

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Дисциплина:** Фронт-энд разработка

**Отчет**

**Лабораторная работа 3**

**Выполнил:**

**Фролова Кристина**

**К3439**

**Проверил:  
Добряков Д. И.**

**Санкт-Петербург**

**2025 г.**

## **Задача**

Мигрировать ранее разработанное приложение (в рамках ЛР1 и ЛР2) на фреймворк Vue.JS.

Каким требованиям ваше приложение должно соответствовать:

- Должен быть подключён роутер
- Должна быть реализована работа с внешним API (желательно посредством axios)
- Разумное деление на компоненты (продемонстрируйте понимание компонентного подхода)
- Использование composable для выделения повторяющиеся функционала в отдельные файлы

## **Ход работы**

На vue были перенесены все страницы приложения, многие страницы были разбиты покомпонентно, чтобы можно было их переиспользовать

VS vue-rent-service Version control

```

Project
  index.js
  stores/auth.js
  main.js
  jwt.js
  LoginPage.vue
  RegisterPage.vue
  OwnerC

  location.svg
  phone-svrepo-com.svg
  node_modules library root
  public
  src
    api
      advertisements.js
      auth.js
      index.js
      instance.auth.js
      instance.js
      messages.js
      rentals.js
      users.js
    components
      AdvertisementFilters.vue
      ChatModal.vue
      ContactModal.vue
      Footer.vue
      Navbar.vue
      OwnerCard.vue
      ProfileCard.vue
      ProfileTabs.vue
      PropertyCard.vue
      PropertyGrid.vue
      PropertyItem.vue
    composables
      useTheme.js
    router
      index.js
    stores
      advertisements.js
      auth.js
      index.js
    utils
      auth.storage.js
      datetime.js
      jwt.js
      mappers.js
    views
      AdvertisementPage.vue
      AdvertisementsPage.vue
      LoginPage.vue
      ProfilePage.vue
      RegisterPage.vue
      App.vue
      main.js
.editorconfig
.gitattributes
.gitignore
.eslintrc.json
index.html
.jsonconfig.json
.package.json

```

```

<script setup> Show component usages
import { reactive } from "vue";

const props = defineProps({ props: {
  modelValue: { type: Object, required: true },
});

const emit = defineEmits(emitOptions: ["update:modelValue", "apply"]);

const local = reactive(target: { ...props.modelValue });

function applyFilters() { Show usages
  emit(event: "update:modelValue", args: { ...local });
  emit(event: "apply");
}

</script>

<template>
  <form class="row g-2 mb-4">
    <div class="col-md-4">
      <input
        v-model="local.location"
        class="form-control"
        placeholder="Город"
      />
    </div>

    <div class="col-md-3">
      <select v-model="local.livingType" class="form-select">
        <option value="">Тип жилья</option>
        <option value="ROOM">Комната</option>
        <option value="FLAT">Квартира</option>
        <option value="COUNTRY_HOUSE">Дом</option>
      </select>
    </div>

    <div class="col-md-2">
      <input
        v-model="local.priceFrom"
        type="number"
        class="form-control"
        placeholder="Цена от"
      />
    </div>

    <div class="col-md-2">
      <input
        v-model.number="local.priceTo"
        type="number"
        class="form-control"
        placeholder="Цена до"
      />
    </div>

    <div class="col-md-1 d-grid">
      <button
        type="button"
        class="btn btn-primary"
        @click="applyFilters"
      >Apply</button>
    </div>
  </form>

```

Рисунок 1 - Пример компонента фильтров

```
api/index.js
import ...

export const advertisementsApi : AdvertisementsApi = new AdvertisementsApi(instance); Show usages
export const usersApi : UsersApi = new UsersApi(instance); Show usages
export const authApi : AuthApi = new AuthApi(apiAuth); Show usages
export const advertisementsAuthApi : AdvertisementsApi = new AdvertisementsApi(apiAuth); Show usages
export const messagesAuthApi : MessagesApi = new MessagesApi(apiAuth); Show usages
export const rentalsAuthApi : RentalsApi = new RentalsApi(apiAuth); Show usages
export const usersMeApi : UsersApi = new UsersApi(apiAuth); Show usages
```

Рисунок 2 - Используемые api

```
js router/index.js      ▼ AdvertisementsPage.vue      ▼ AdvertisementPage.vue      ▼ Advertiseme
1 import axios from "axios";
2 import { useAuthStore } from "@/stores/auth";
3 import { isTokenExpired } from "@/utils/jwt.js";
4
5 const apiAuth : AxiosInstance = axios.create({
6   baseURL: "http://localhost:8080/v1",
7 });
8
9 apiAuth.interceptors.request.use( onFulfilled: (config) :any  => {
10   const auth : Store<"auth", {...}, {...}, {...}> = useAuthStore();
11
12   if (auth.accessToken && isTokenExpired(auth.accessToken)) {
13     auth.logout();
14     return config;
15   }
16
17   if (auth.accessToken) {
18     const type = auth.tokenType || "Bearer";
19     config.headers.Authorization = `${type} ${auth.accessToken}`;
20   }
21   //↑L to Chat, ⇧I to Command
22   return config;
23 });
24
25 apiAuth.interceptors.response.use(
26   onFulfilled: (res) => res,
27   onRejected: (error) : Promise<never> => {
28     if (error?.response?.status === 401) {
29       const auth : Store<"auth", {...}, {...}, {...}> = useAuthStore();
30       auth.logout();
31     }
32     return Promise.reject(error);
33   }
34 );
35
36 export default apiAuth; Show usages
37
```

Рисунок 3 - Инстанс для авторизованных запросов

**Комната в коммуналке в центре Спб**

Санкт-Петербург

**60 000 ₽/день****Описание**

Уютная комната в историческом центре с видами на Мойку

**Условия аренды**

- Залог: 1073741824 ₽
- Комиссия: 1073741824 ₽

**Параметры**

- Тип объекта: Комната
- Комнат: 2
- Площадь: 46 м<sup>2</sup>
- Этаж: 5 из 12
- Площадь кухни: 12 м<sup>2</sup>
- Ремонт: string
- Интернет: есть
- Телевизор: есть
- Мебель: string
- Техника: string
- Общая площадь: 15 м<sup>2</sup>

**Владелец**

string string

Частное лицо

string

[Написать владельцу](#)**Рисунок 4 - Пример страницы объявления на vue****Вывод**

В ходе работы сайт был перенесён на vue.js