

**Министерство науки и высшего образования Российской Федерации**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**  
**ITMO University**

**Лабораторная работа 3**

**По дисциплине** Фронтэнд-разработка

**Обучающиеся** Сокович Степан Витальевич

**Факультет** Факультет прикладной информатики

**Группа** К3344

**Направление подготовки** 45.03.04 Интеллектуальные системы в гуманитарной сфере

**Образовательная программа** Интеллектуальные системы в гуманитарной сфере

**Обучающиеся**

Сокович С.В.

**Преподаватель**

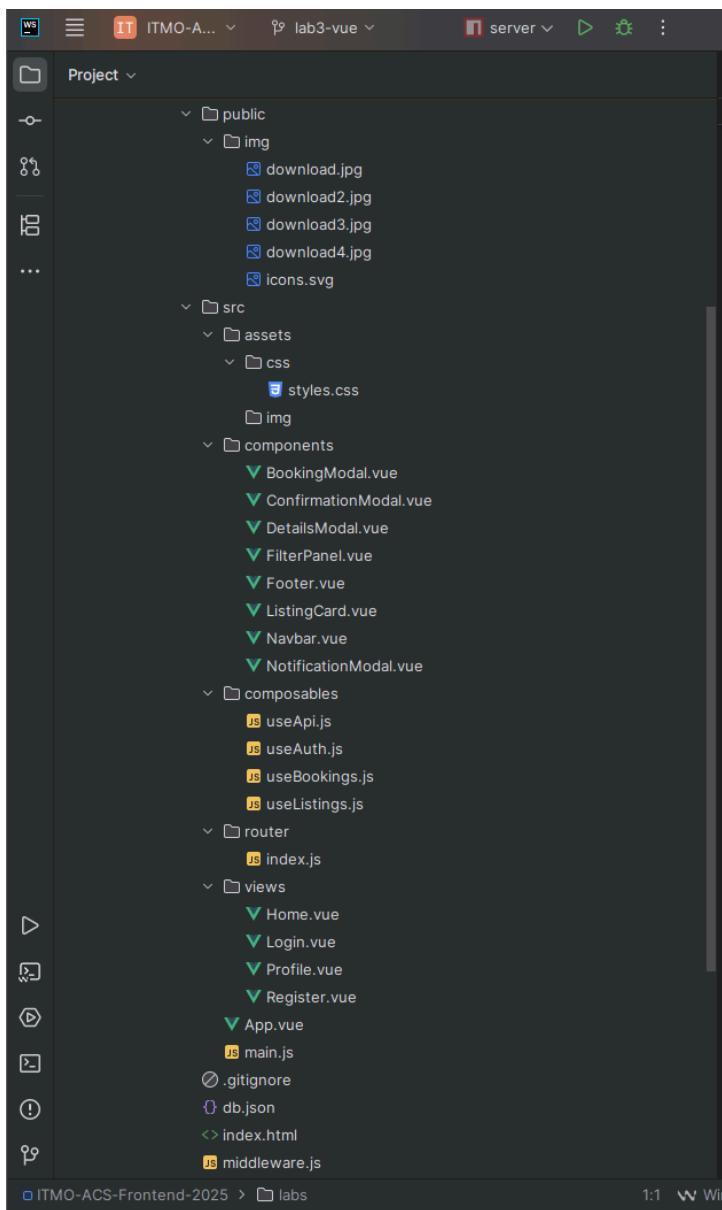
Добряков Д.И.

**Тема:** Разработка SPA на Vue 3 с маршрутизацией, внешним API (axios), компонентным подходом и composables.

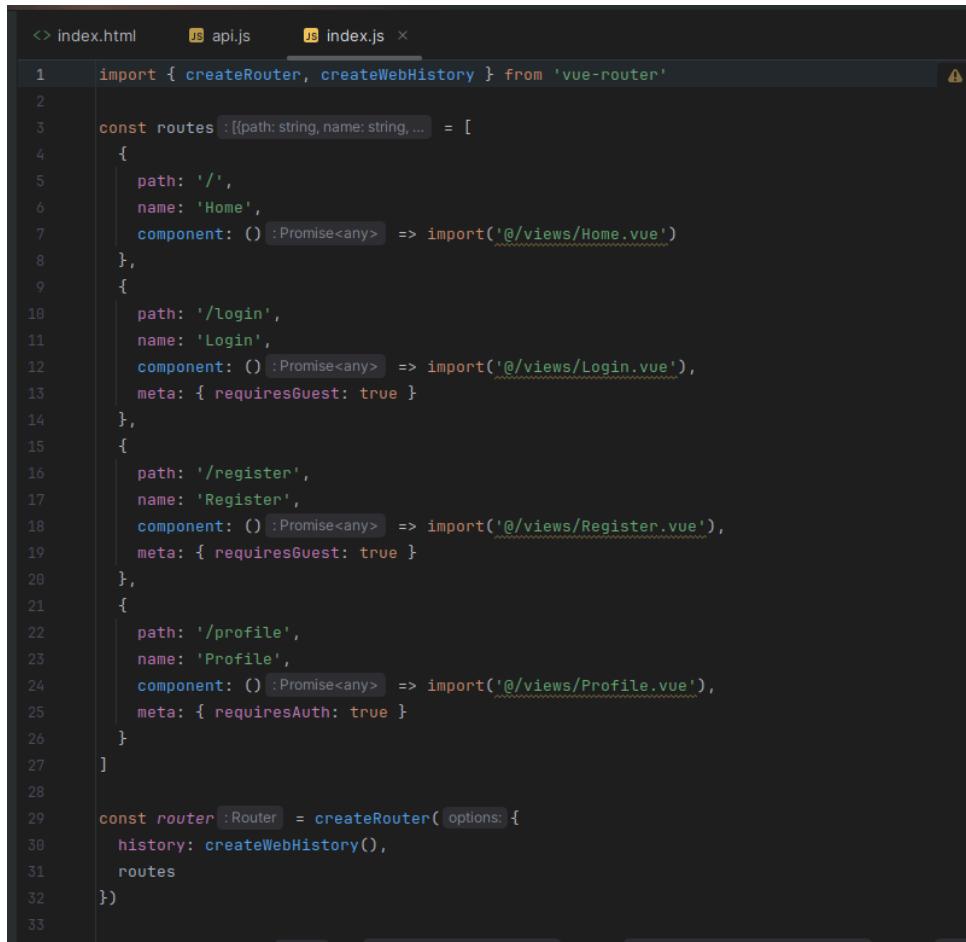
**Цель работы:** Мигрировать ранее созданный проект Разработать одностраничное веб-приложение на Vue 3, подключить Vue Router, реализовать получение данных из внешнего API (axios), разделить интерфейс на компоненты и вынести повторяющуюся логику в composables.

### Ход работы:

После инициализации проекта на vue и миграции на существующей версии проекта структура проекта выглядит следующим образом:

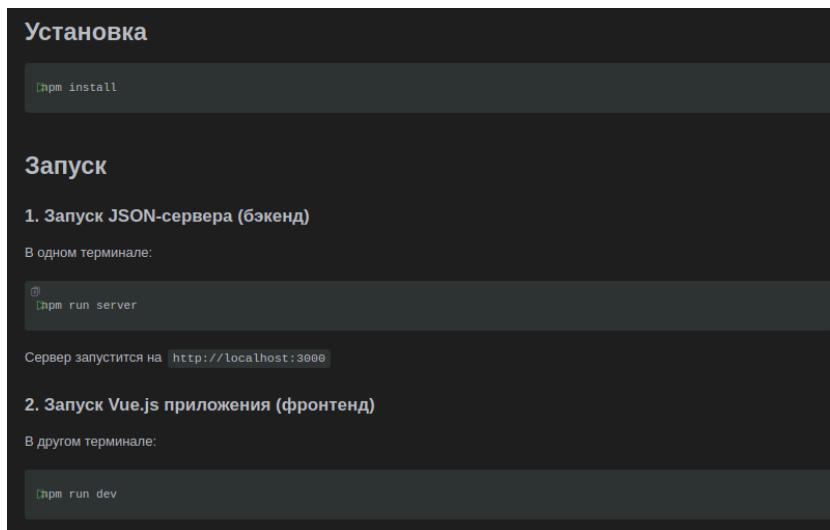


## Реализован роутер запросов



```
index.html      JS api.js      JS index.js ×
1 import { createRouter, createWebHistory } from 'vue-router'          ▲ 6
2
3 const routes :[path: string, name: string, ...] = [
4   {
5     path: '/',
6     name: 'Home',
7     component: () :Promise<any> => import('@/views/Home.vue')
8   },
9   {
10    path: '/login',
11    name: 'Login',
12    component: () :Promise<any> => import('@/views/Login.vue'),
13    meta: { requiresGuest: true }
14  },
15  {
16    path: '/register',
17    name: 'Register',
18    component: () :Promise<any> => import('@/views/Register.vue'),
19    meta: { requiresGuest: true }
20  },
21  {
22    path: '/profile',
23    name: 'Profile',
24    component: () :Promise<any> => import('@/views/Profile.vue'),
25    meta: { requiresAuth: true }
26  }
27 ]
28
29 const router :Router = createRouter( options: {
30   history: createWebHistory(),
31   routes
32 })
33
```

Запустим проект, запустив команды



Установка

```
npm install
```

Запуск

1. Запуск JSON-сервера (бэкенд)  
В одном терминале:  

```
npm run server
```

Сервер запустится на `http://localhost:3000`
2. Запуск Vue.js приложения (фронтенд)  
В другом терминале:  

```
npm run dev
```

На выходе получаем тот же самый рабочий сайт.

The screenshot shows a search results page for 'RentalSite'. On the left, there's a sidebar with filters for 'Type of real estate' (set to 'Any'), 'Max price' (set to 50000), and 'District' (set to 'Search by city/district'). Below these are 'Apply' and 'Reset' buttons. The main area is titled 'Search results' and shows four items:

- 2-bedroom apartment, Center**: Price 45 000 ₽ / month, District: Center. Includes a photo of a bedroom with a large window overlooking a building, a link to 'View details', and a 'More details' button.
- Studio near the metro**: Price 30 000 ₽ / month, District: North. Includes a photo of a living room with a sofa and a TV, a link to 'View details', and a 'More details' button.
- Private house, suburb**: Price 70 000 ₽ / month, District: Suburb. Includes a photo of a modern-looking house with a glass door, a link to 'View details', and a 'More details' button.
- 3-bedroom apartment, South**: Price 55 000 ₽ / month, District: South. Includes a photo of an open-plan living space with a kitchen and dining area, a link to 'View details', and a 'More details' button.

At the bottom right of the main area, it says '4 found'. At the very bottom center, it says '© RentalSite 2025'.

Вывод:

В ходе работы были выполнены следующие действия:

1. Создан новый проект на Vue 3 и перенесены файлы ранее реализованного интерфейса.
2. Подключён и настроен Vue Router для навигации между страницами без перезагрузки.
3. Настроено взаимодействие с внешним API через axios (создан инстанс, определены запросы на получение данных).
4. Интерфейс разделён на независимые компоненты для повышения читаемости и переиспользуемости.
5. Повторяющаяся логика (загрузка данных, состояние, фильтрация, авторизация) вынесена в composable.

После инициализации проекта на **Vue 3** и миграции ранее созданной версии приложения была сформирована базовая структура SPA, включающая основные директории для компонентов, страниц, маршрутизации и работы с API. Проект был адаптирован под компонентный подход и подготовлен для дальнейшего расширения функционала.