

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Дисциплина:** Фронт-энд разработка

Отчет

Лабораторная работа №2

Выполнил:

Русинов Василий

Группа К3440

Проверил:

Добряков Д. И.

Санкт-Петербург

2025 г.

## Задача

Варианты остаются прежними. Теперь Вам нужно привязать то, что Вы делали в ЛР1 к внешнему API средствами fetch/axios/xhr. Реализуйте моковое API средствами JSON-сервера и подключите к нему авторизацию, как в примерах, которые мы рассматривали в рамках тем "Имитация работы с API".

## Ход работы

Настроен локальный REST API с помощью json-server, использующий файл db.json (рисунок 1). Также реализована регистрация пользователей с сохранением данных в db.json.

```
1  {
2    "users": [
3      {
4        "id": "1",
5        "name": "Иван",
6        "email": "test@test.ru",
7        "password": "12345",
8        "token": "token-ivan"
9      },
10     {
11       "id": "2c3d",
12       "name": "ya",
13       "email": "ya@mail.ru",
14       "password": "ya@mail.ru",
15       "token": "1765923741846"
16     },
17     {
18       "id": "6790",
19       "name": "ya@mail.ru",
20       "email": "ya@mail.ru",
21       "password": "ya@mail.ru",
22       "token": "1765992338192"
```

Рисунок 1 – Пример данных в db.json

Реализована логика аренды объекта недвижимости:

- при нажатии кнопки «Арендовать» создаётся запись в rentals (рисунок 2);
- объект привязывается к текущему пользователю по userId.

```
"rentals": [  
  {  
    "id": "8ec3",  
    "userId": "470f",  
    "title": "Квартира в Москве",  
    "price": 45000,  
    "img": "http://localhost:5500/img/house1.jpg"  
  }  
]
```

Рисунок 2 – Запись в rentals

Реализовано отображение арендованных объектов в профиле (рисунки 3 и 4).

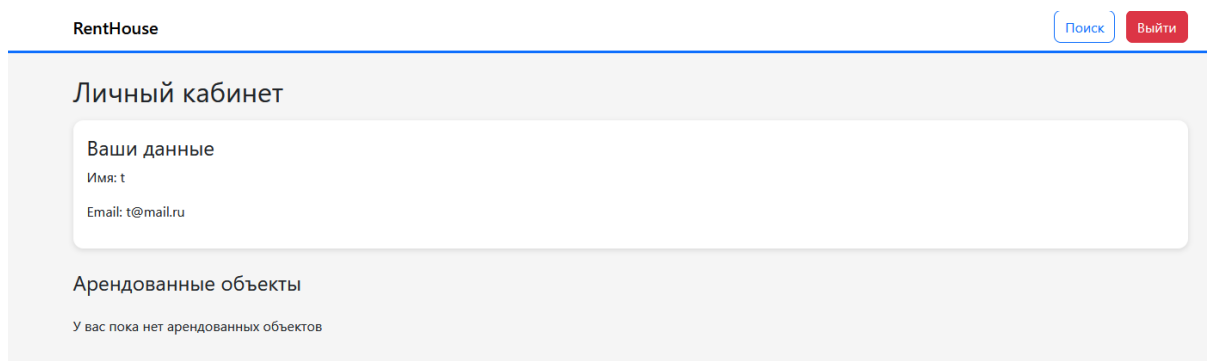


Рисунок 3 – Профиль до аренды

## Личный кабинет

## Ваши данные

Имя: t

Email: t@mail.ru

## Арендованные объекты



Квартира в Москве

45 000 Р/мес

Рисунок 4 – Профиль после аренды

**Вывод**

В ходе выполнения лабораторной работы были получены практические навыки разработки клиентских веб-приложений, работы с REST API, управления состоянием пользователя и взаимодействия с сервером без использования серверной логики.