САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Дисциплина: Фронт-энд разработка

Отчет

Домашняя работа №5

Выполнил:

Русинов Василий

Группа К3440

Проверил:
Добряков Д. И.

Санкт-Петербург

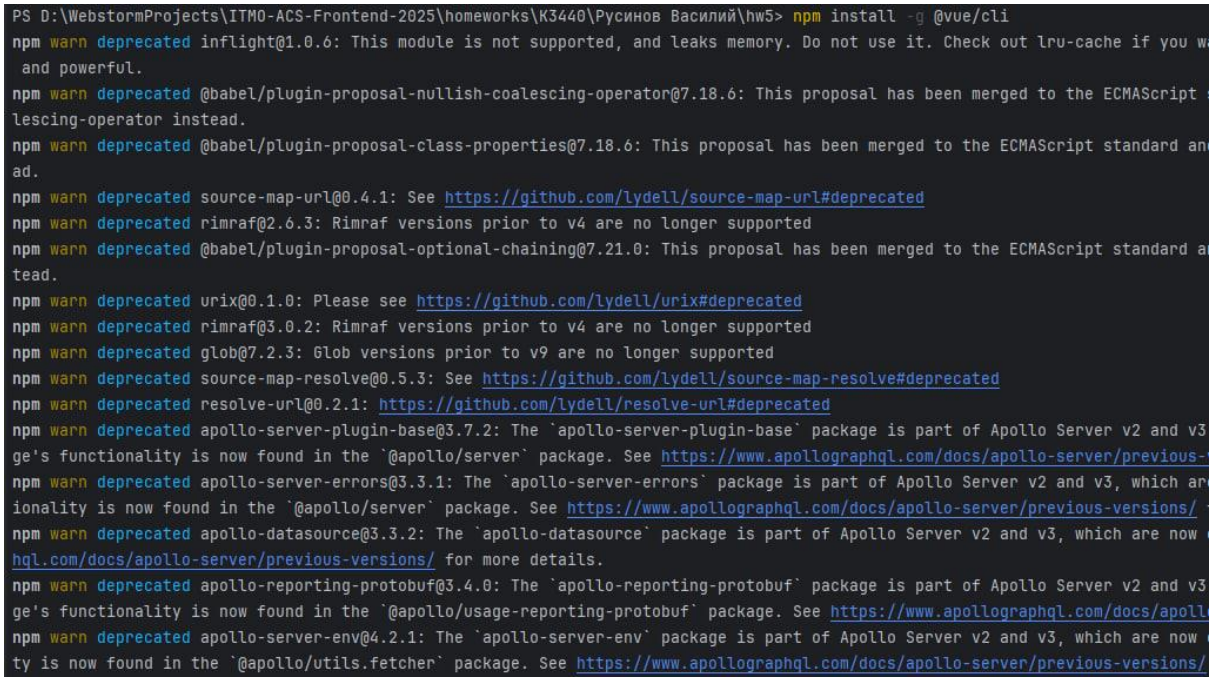
2025 г.

Задача

В рамках данной работы Вам предстоит изучить основные команды пакетного менеджера NPM и научиться стартовать проект на Vue. Научиться работать с npm и vue на основе мануала: https://docs.google.com/document/d/187UkgGNrcWqkb2aCGpkHTLgeozoElMqdVgVGMBOC9gk/edit?usp=sharing.

Ход работы

Установка vue:



Инициализация проекта:

```
PS D:\WebstormProjects\ITMO-ACS-Frontend-2025\homeworks\K3440\Русинов Василий\hw5> npm init vue@latest

> npx
> create-vue

┌  Vue.js - The Progressive JavaScript Framework
│
◇  Project name (target directory):
│  vue-project
│
◇  Select features to include in your project: (↑/↓ to navigate, space to select, a to toggle all, enter to confirm)
│  TypeScript, Router (SPA development), Pinia (state management), ESLint (error prevention)
│
◇  Select experimental features to include in your project: (↑/↓ to navigate, space to select, a to toggle all, enter to
confirm)
│  none
│
◇  Skip all example code and start with a blank Vue project?
│  Yes

Scaffolding project in D:\WebstormProjects\ITMO-ACS-Frontend-2025\homeworks\K3440\Русинов Василий\hw5\vue-project...
│
└  Done. Now run:
```

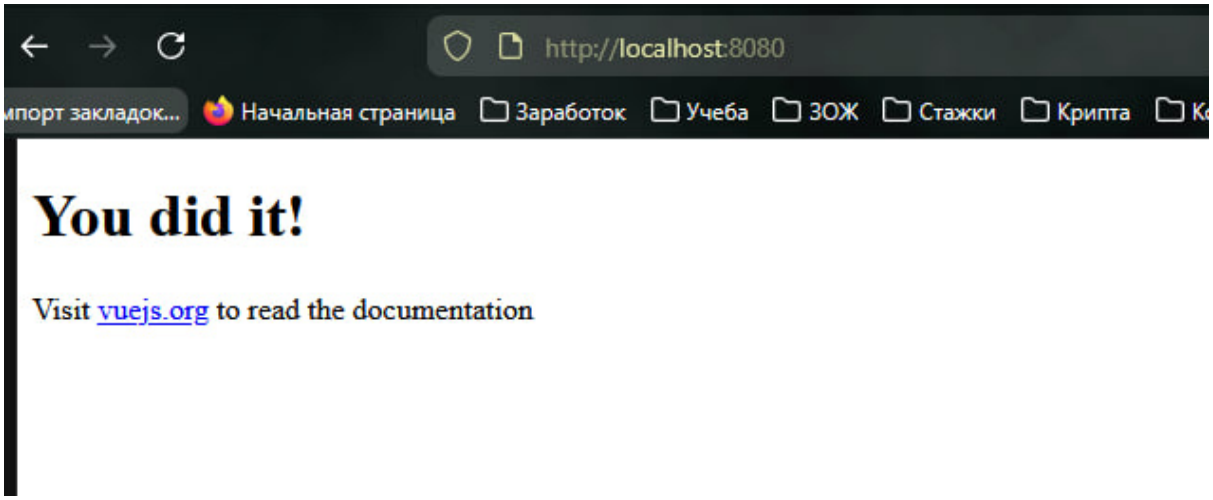
Установка зависимостей:

```
PS D:\WebstormProjects\ITMO-ACS-Frontend-2025\homeworks\K3440\Русинов Василий\hw5> cd vue-project
PS D:\WebstormProjects\ITMO-ACS-Frontend-2025\homeworks\K3440\Русинов Василий\hw5\vue-project> npm install

added 296 packages, and audited 297 packages in 45s

73 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS D:\WebstormProjects\ITMO-ACS-Frontend-2025\homeworks\K3440\Русинов Василий\hw5\vue-project> npm start

> vue-project@0.0.0 start
> vite --port 8080


  VITE v7.3.0  ready in 1951 ms

  ➜  Local:   http://localhost:8080/
  ➜  Network: use --host to expose
  ➜  Vue DevTools: Open http://localhost:8080/__devtools__/ as a separate window
  ➜  Vue DevTools: Press Alt(⌥)+Shift(⇧)+D in App to toggle the Vue DevTools
  ➜  press h + enter to show help
```

Изменения в файле package.json:

```
"scripts": {
  "start": "vite --port 8080",
  "dev": "vite",
  "build": "vite build",
  "preview": "vite preview",
  "build-only": "vite build",
  "type-check": "vue-tsc --build",
  "lint": "eslint . --ext .vue,.js,.jsx,.cjs,.mjs --fix --ignore-path .gitignore"
},
```

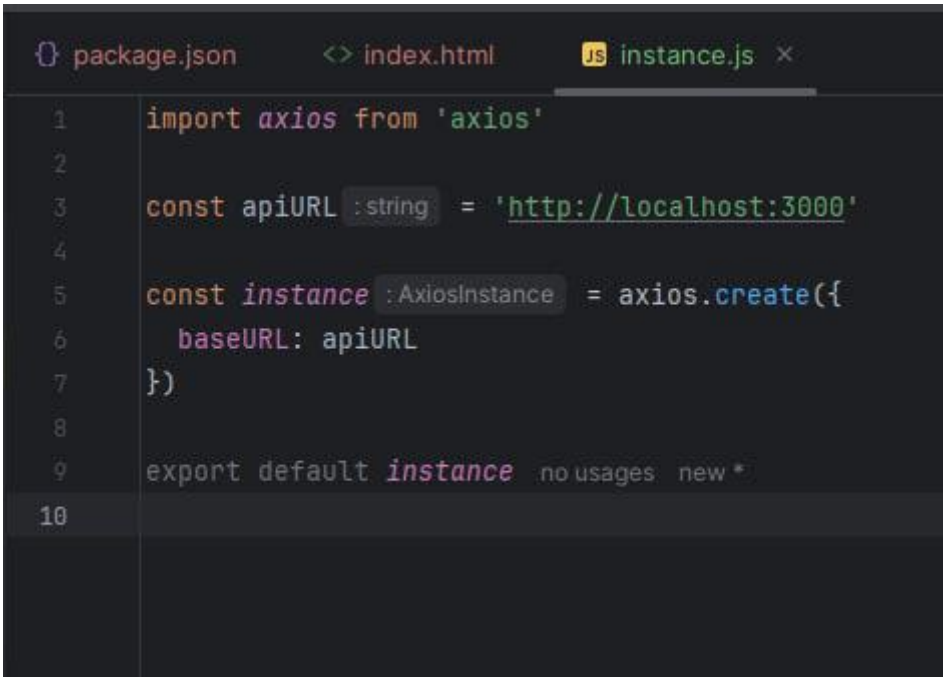
Результат запуска проекта:



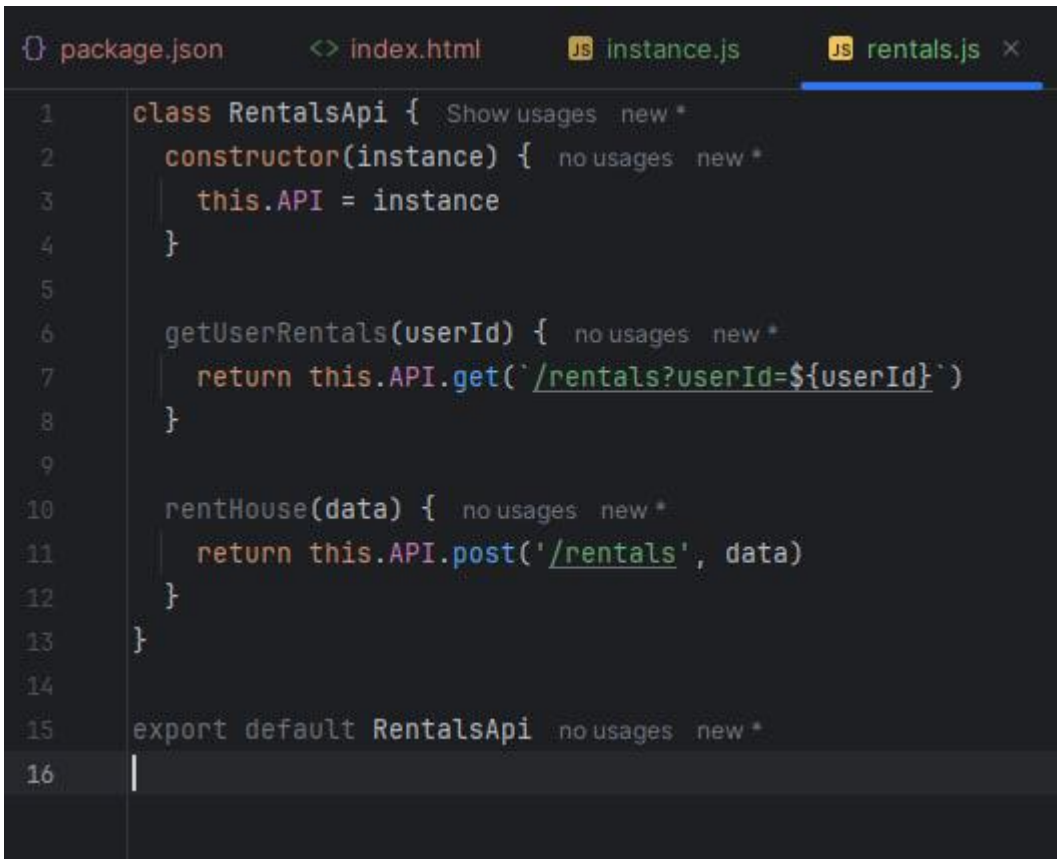
Установка утилит:

```
PS D:\WebstormProjects\ITMO-ACS-Frontend-2025\homeworks\K3440\Русинов Василий\hw5\vue-project> npm i axios pinia-persists bootstrap -S

added 26 packages, and audited 323 packages in 4s

81 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

Файл api/instance.js:



Файл api/rentals.js:



Файл api/index.js:

package.json | index.html | instance.js | rentals.js | index.js

```
import instance from "@/api/instance"
import RentalsApi from "@/api/rentals"

const rentalsApi :RentalsApi = new RentalsApi(instance)  no usages  new *

export {
  rentalsApi
}
```

Файл stores/index.js:

index.js

```
import { persist } from 'pinia-persists'
import { createPinia } from 'pinia'

const pinia :Pinia = createPinia()

pinia.use(persist())

export default pinia  no usages  new *
```

Файл main.js:

index.js main.js

```
import { createApp } from 'vue'

import App from '@/App.vue'
import router from '@/router'
import store from '@/stores'

import 'bootstrap/dist/css/bootstrap.min.css'
import 'bootstrap'
import '@/assets/main.css'

const app : App<Element> = createApp(App)

app.use(store)
app.use(router)

app.mount( rootContainer: '#app')
```

Файл App.vue:

index.js main.js App.vue

```
<template>  Show component usages
  <router-view />
</template>
```

Файл stores/rentals.js:

stores\rentals.js

```
import { defineStore } from 'pinia'
import { rentalsApi } from '@/api'

export const useRentalsStore : StoreDefinition<"rentals", {...}, {...}, {...}> = defineStore( id: 'rentals', options: { Show usages  new *
  state: () : {rentals: []} => ({
    rentals: []
  }),

  actions: {
    async loadRentals(userId) : Promise<void> {
      const res = await rentalsApi.getUserRentals(userId)
      this.rentals = res.data
    },
    async rentHouse(data) : Promise<void> {
      await rentalsApi.rentHouse(data)
      await this.loadRentals(data.userId)
    }
  },

  persist: true
})
```

Файл router/index.js:

stores\rentals.js index.js

```
import { createRouter, createWebHistory } from 'vue-router'

const routes : [{path: string, name: string, ... = [
  {
    path: '/',
    name: 'rentals',
    component: () : Promise<{...}> => import('@/views/RentalsPage.vue')
  }
]


export default createRouter( options: { Show usages
  history: createWebHistory(),
  routes
})
```

Компонента RentalCard.vue:

stores\rentals.js | index.js | RentalCard.vue

```vue
<template>  Show component usages
  <div class="card">
    <img :src="img" class="card-img-top">
    <div class="card-body">
      <h5>{{ title }}</h5>
      <p>{{ price }} ₽/мес</p>
    </div>
  </div>
</template>

<script>
export default {  Show usages  new *
  props: {
    title: String,
    price: Number,
    img: String
  }
}
</script>
```

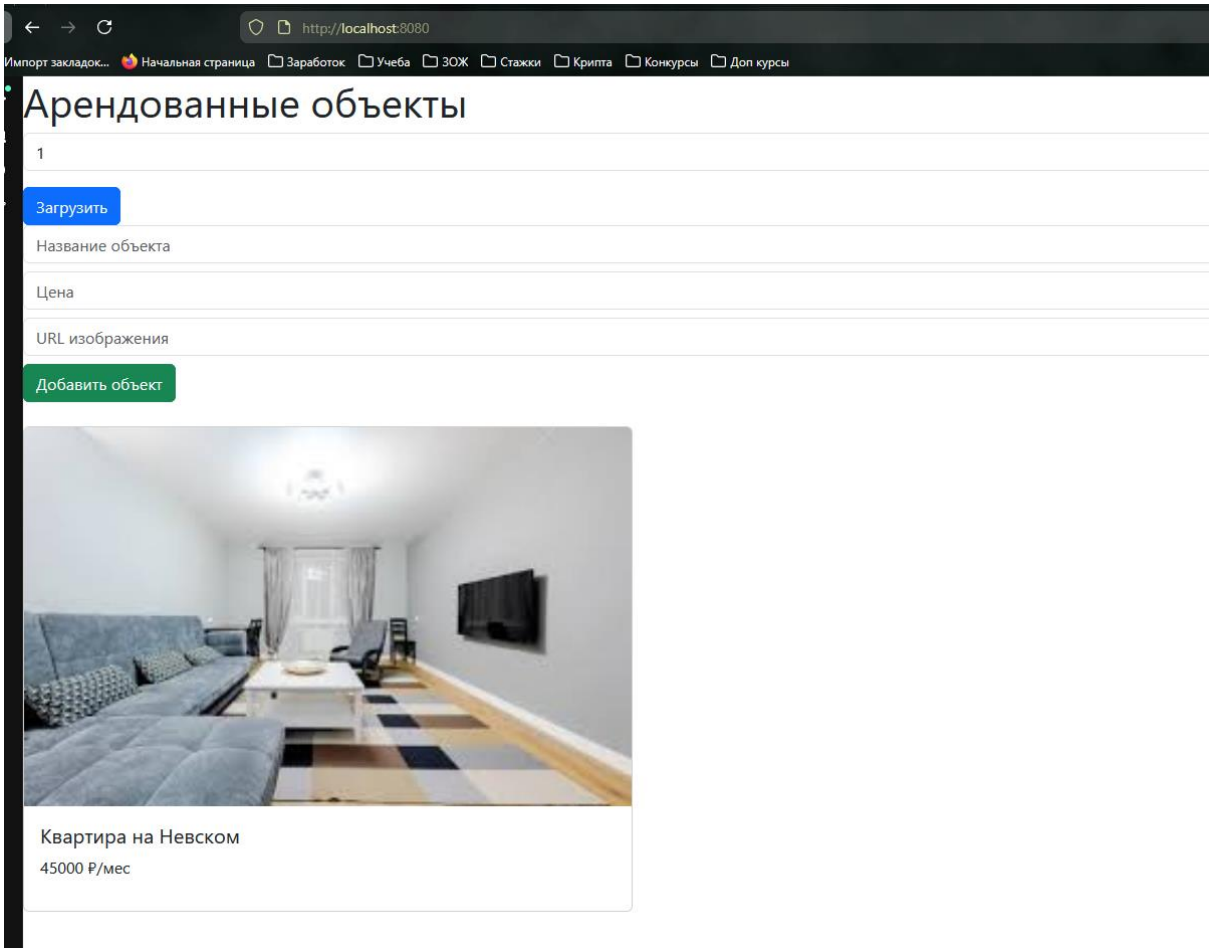
Представление RentalsPage:

```vue
<template>
  <h1>Арендованные объекты</h1>

  <form @submit.prevent="load">
    <input
      type="number"
      v-model="userId"
      placeholder="ID пользователя"
      class="form-control mb-3"
    >
    <button class="btn btn-primary">Загрузить</button>
  </form>
  <form @submit.prevent="createRental" class="mb-4">
    <input
      v-model="form.title"
      placeholder="Название объекта"
      class="form-control mb-2"
    >

    <input
      type="number"
      v-model="form.price"
      placeholder="Цена"
      class="form-control mb-2"
```

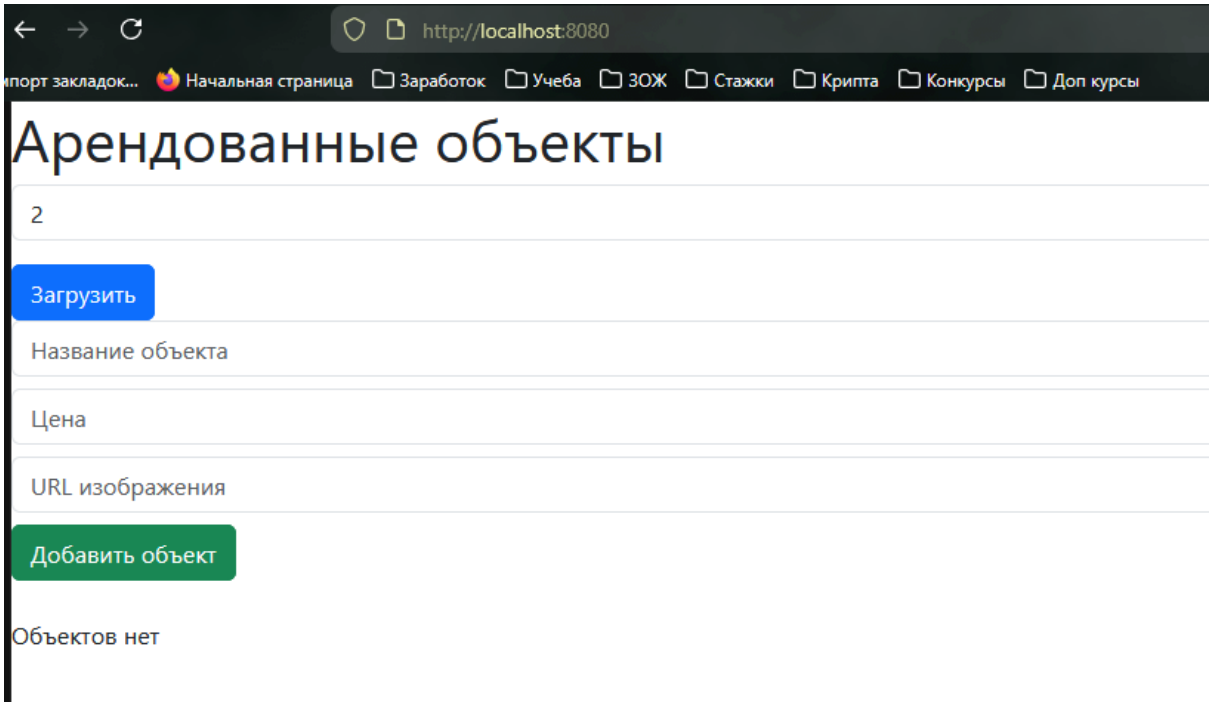
```
    >

    <input
      v-model="form.img"
      placeholder="URL изображения"
      class="form-control mb-2"
    >

    <button class="btn btn-success">
      Добавить объект
    </button>
  </form>

  <p v-if="rentals.length === 0">
    Объектов нет
  </p>

  <div class="row">
    <div class="col-md-4" v-for="item in rentals" :key="item.id">
      <RentalCard
        :title="item.title"
        :price="item.price"
        :img="item.img"
      />
    </div>
  </div>
</template>

<script>
import { mapState, mapActions } from 'pinia'
import { useRentalsStore } from '@/stores/rentals'

import RentalCard from '@/components/RentalCard.vue'

export default {
  components: { RentalCard },

  data() {
    return {
      userId: localStorage.getItem('userId') || 1,
      form: {
        title: '',
        price: '',
        img: '',
        userId: 1
      }
    }
  },

  computed: {
    ...mapState(useRentalsStore, ['rentals'])
  },

  methods: {
    ...mapActions(useRentalsStore, ['loadRentals', 'rentHouse']),
    load() {
      this.loadRentals(this.userId)
    },
    async createRental() {
      await this.rentHouse(this.form)
```

```
      this.form.title = ''
      this.form.price = ''
      this.form.img = ''
    }
  },

  mounted() {
    this.loadRentals(1)
  }
}
</script>
```

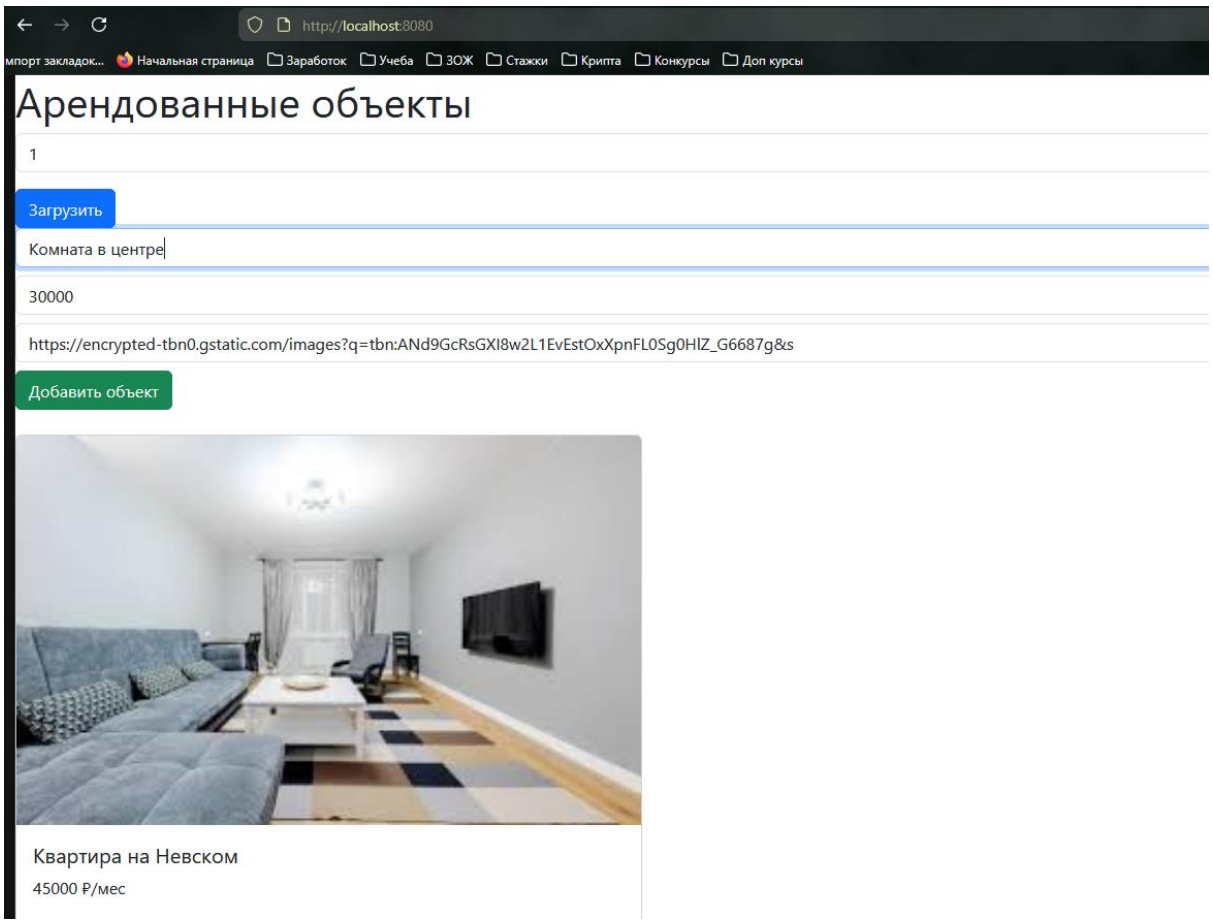
Финальная версия:



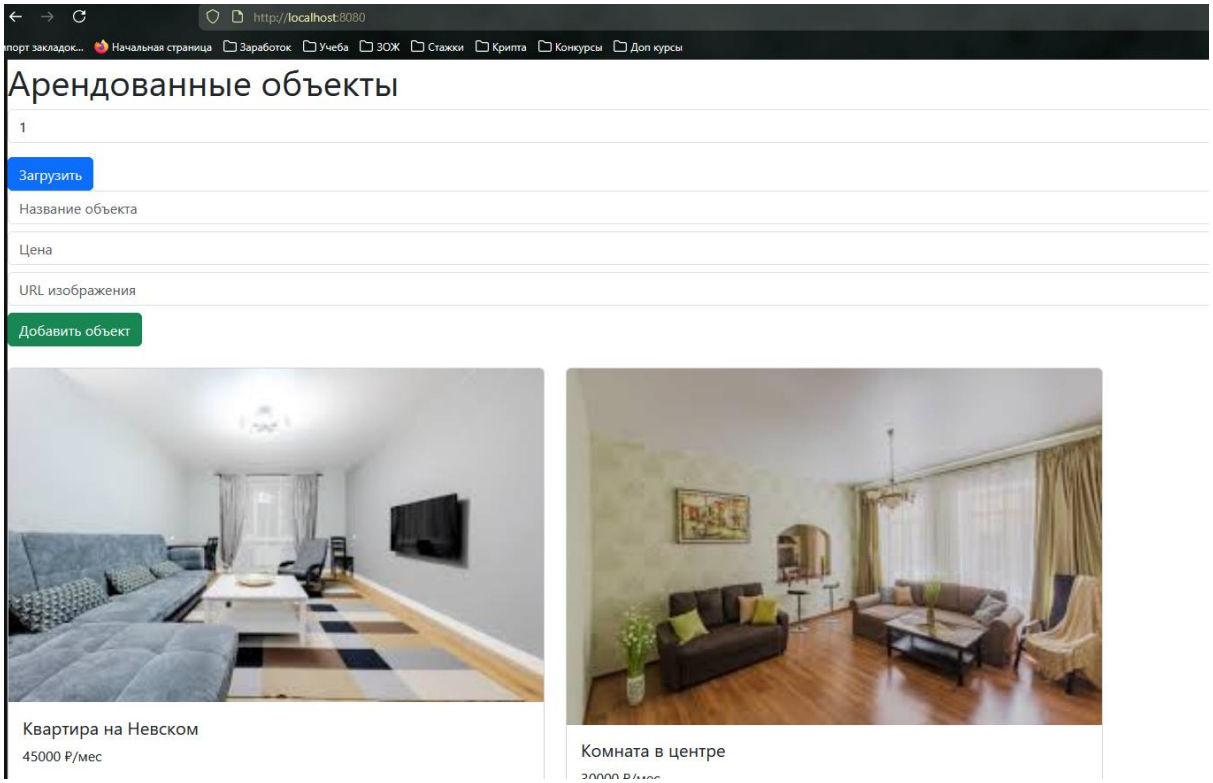
При попытке загрузить информацию для второго юзера:



Добавление объекта:



Объект добавлен, форма чиста:



Вывод

В результате выполнения работы, были изучены основы работы с npm и vue на основе мануала.