

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Фронт-энд разработка

Отчет

Лабораторная работа №3

Выполнил:

Жигалова Анастасия Евгеньевна

K33392

Проверил:

Добряков Д. И

Санкт-Петербург

2023 г.

Задача

Мигрировать ранее написанный сайт на фреймворк Vue.JS.

Минимальные требования:

- Должен быть подключён роутер
- Должна быть реализована работа с внешним API
- Разумное деление на компоненты
- Использование composables

Ход работы:

Создаю App.vue – главный компонент, с его помощью я отображаю страницу. Header, Footer – постоянные, RouterView же меняется, в зависимости от страницы.

```
src > App.vue > ...
1  <script setup>
2  import { RouterLink, RouterView } from 'vue-router'
3  import Header from "@/components/Header.vue";
4  import Footer from "@/components/Footer.vue";
5
6  </script>
7
8  <template>
9    <header>
10      <nav>
11        <RouterLink to="/"></RouterLink>
12      </nav>
13    </header>
14    <Header />
15    <RouterView />
16    <Footer />
17  </template>
18
19
20 <style scoped>
21
22 </style>
23
```

Создаю компоненты, которые будут отображать каждую страницу.

Cart.vue – страница корзины

Catalog.vue – страница каталога

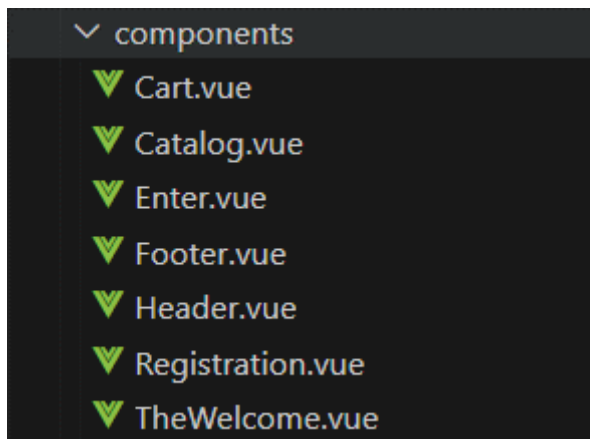
Enter.vue – страница входа

Footer.vue – подвал сайта

Header.vue – навигация сайта

Registrarion.vue – страница регистрации

TheWelcome.vue – главная страница



```
<!-- eslint-disable vue/multi-word-component-names -->
<template>
  <footer class="text-center">
    <div class="bg py-3">
      <div class="container">
        <div class="row">
          <div class="col-1">
            <h5>📍 нас</h5>
            <div>ТНЦ</div>
          </div>

          <div class="col-2">
            <h5>Время работы</h5>
            <ul class="list-unstyled">
              <li><div>Каждый день 🕒 9:00 до 18:00</div></li>
            </ul>
          </div>

          <div class="col-3">
            <h5>Мы в сети</h5>
            
            <a href="https://t.me/aanazhi"></a>
          </div>
        </div>
      </div>
    </div>
  </footer>
</template>

<script>

export default {
};
</script>
```

```

<!-- eslint-disable no-mixed-spaces-and-tabs -->
<script>
// If you are using PurgeCSS, make sure to whitelist the carousel CSS classes
import 'vue3-carousel/dist/carousel.css'
import { Carousel, Slide, Navigation, Pagination } from 'vue3-carousel'

export default {
  name: 'WelcomeView',
  components: {
    Carousel,
    Slide,
    Navigation,
    Pagination
  },
}
</script>

<template>
<div id="app">
  <Carousel>
    <Slide :index="1">
      
    </Slide>
    <Slide :index="2">
      
    </Slide>
    <Slide :index="3">
      
    </Slide>
    <Slide :index="4">
      
    </Slide>
  </Carousel>

  <template #addons>
    <Navigation />
    <Pagination />
  </template>
</div>
</template>

```

```

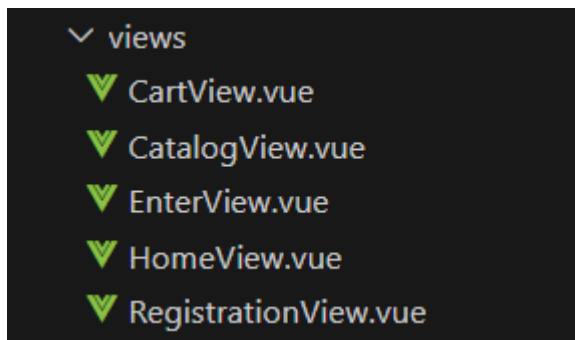
<!-- eslint-disable vue/multi-word-component-names -->
<script>

</script>

<template>
<nav class = "navigation">
  <nav class="navbar navbar-expand-lg navbar-light bg-white">
    <div class="container-fluid">
      <a class="navbar-brand" href="#">
        
      </a>
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" data-bs-keyboard="true">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse" id="navbarSupportedContent">
        <ul class="nav nav-tabs me-auto mb-2 mb-lg-0 top-menu">
          <li class="nav-item">
            <router-link class="dropdown-item" to="/home">Главная</router-link>
          </li>
          <li class="nav-item">
            <router-link class="dropdown-item" to="/catalog">Каталог</router-link>
          </li>
          <li class="nav-item">
            <router-link class="dropdown-item" to="/about">О нас</router-link>
          </li>
          <li class="nav-item">
            <router-link class="dropdown-item" to="/contact">Контакты</router-link>
          </li>
        </ul>
        <ul class="navbar-nav ms-auto mb-2 mb-lg-0">
          <li class="nav-item dropdown">
            <a class="nav-link dropdown-toggle" data-bs-toggle="dropdown" href="#">
              <span class="sr-only"></span>
            </a>
            <ul class="dropdown-menu" aria-labelledby="navbarDropdown">
              <li><a href="#">Профиль</a></li>
              <li><a href="#">Настройки</a></li>
              <li><a href="#">Выход</a></li>
            </ul>
          </li>
        </ul>
      </div>
    </div>
  </nav>
</nav>

```

Создаю views для всех страничек.



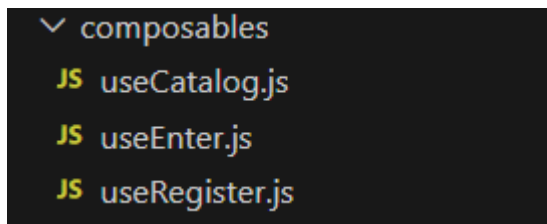
Структура у них одна.

```
1 <script setup>
2   import Cart from '../components/Cart.vue'
3
4 </script>
5
6 <template>
7   <main>
8     <Cart />
9
10  </main>
11 </template>
12
```

Создаю Роутер, добавляю взаимодействие с ним в index.js

```
src > router > JS index.js > routes
1 import { createRouter, createWebHistory } from 'vue-router'
2 import HomeView from '../views/HomeView.vue'
3 import EnterView from '../views/EnterView.vue'
4 import RegistrationView from '../views/RegistrationView.vue'
5 import CatalogView from '../views/CatalogView.vue'
6 import CartView from '../views/CartView.vue'
7
8
9 const router = createRouter({
10   history: createWebHistory(import.meta.env.BASE_URL),
11   routes: [
12     {
13       path: '/home',
14       name: 'home',
15       component: HomeView
16     },
17     {
18       path: '/enter',
19       name: 'enter',
20       component: EnterView
21     },
22     {
23       path: '/registration',
24       name: 'registration',
25       component: RegistrationView
26     },
27     {
28       path: '/catalog',
29       name: 'catalog',
30       component: CatalogView
31     },
32     {
33       path: '/cart',
34       name: 'cart',
35       component: CartView
36     },
37   ],
38 })
```

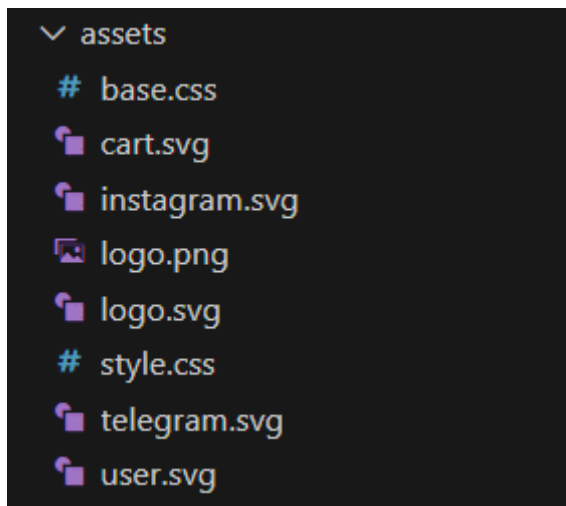
Для использования composables, создаю файлы js. Composable можно переиспользовать, а также он делает код более чистым.



```
src > components > ▼ Catalog.vue > {} style > ⚙️ .form-group-margin
40 |   </div>
41 | </div>
42 </template>
43
44 <script>
45 import { useCatalog } from "@composables/useCatalog";
46 import { onMounted } from 'vue';
47
48 export default {
49   setup() {
50     const { items, selectedCategory, searchKeyword, fetchItems, filteredItems, addToCart } = useCatalog();
51
52     // Вызов fetchItems при монтировании компонента
53     onMounted(fetchItems);
54
55     return { items, selectedCategory, searchKeyword, filteredItems, addToCart };
56   }
57 };
58 </script>
59
```

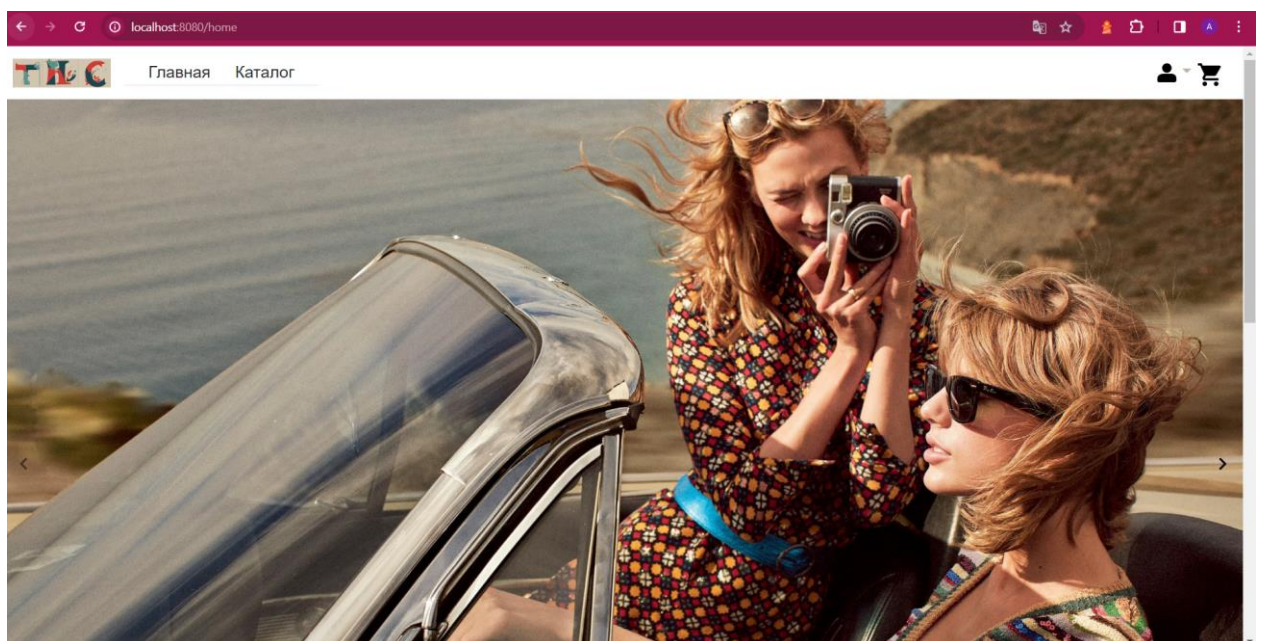
```
src > composables > JS useCatalog.js > useCatalog
1 /* eslint-disable no-undef */
2 import { ref, computed } from 'vue';
3
4 export function useCatalog() {
5   const items = ref([]);
6   const selectedCategory = ref('');
7   const searchKeyword = ref('');
8
9   const fetchItems = () => {
10     fetch('http://localhost:8090/api/collections/catalog/records')
11       .then(response => response.json())
12       .then(data => {
13         items.value = data.items;
14       })
15       .catch(error => console.error(error));
16   };
17
18   const filteredItems = computed(() => {
19     return items.value.filter(item => {
20       return (selectedCategory.value === '' || item.cardName.includes(selectedCategory.value)) &&
21         item.title.toLowerCase().includes(searchKeyword.value.toLowerCase());
22     });
23   });
24
25   const addToCart = (item) => {
26     fetch('http://localhost:8090/api/collections/cart/records', {
27       method: 'POST',
28       headers: {
29         'Content-Type': 'application/json'
30       },
31       body: JSON.stringify(item)
32     })
33       .then(response => response.json())
34       .then(data => {
35         console.log(data);
36         alert("Товар добавлен в корзину");
37       })
38   };
39 }
```

Добавляю svg-файлы для иконок.



В main.js добавляю бутстрап и стили из assets.

```
src > JS main.js > ...  
1  
2 import { createApp } from 'vue'  
3 import { createPinia } from 'pinia'  
4  
5 import App from './App.vue'  
6 import router from './router'  
7 import 'bootstrap/dist/css/bootstrap.min.css';  
8 import 'bootstrap/dist/js/bootstrap.bundle.min.js';  
9 import '@/assets/base.css';  
10 import '@/assets/style.css';  
11  
12 const app = createApp(App)  
13  
14 app.use(createPinia())  
15 app.use(router)  
16  
17 app.mount('#app')  
18
```





Категории:

Все

Поиск:



Юбка

Цена: 2300 руб.

[Добавить в корзину](#)



Джинсы

Цена: 1800 руб.

[Добавить в корзину](#)



Платье

Цена: 3000 руб.

[Добавить в корзину](#)



Рубашка

Цена: 2000 руб.

[Добавить в корзину](#)



Пройти регистрацию

Почта

Введите почту

Пароль

Введите пароль

[Зарегистрироваться](#)

О нас
TNC

Время работы
Каждый день с 9:00 до 18:00

Мы в сети



Войдите в свой аккаунт

Почта

Введите почту

Пароль

Введите пароль

[Войти](#)

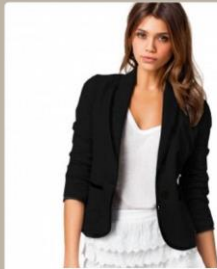
О нас
TNC

Время работы
Каждый день с 9:00 до 18:00

Мы в сети



В вашей корзине:



Пиджак

Цена: 4000 руб.

[Удалить из корзины](#)



Свитер

Цена: 2500 руб.

[Удалить из корзины](#)



Юбка

Цена: 2300 руб.

[Удалить из корзины](#)



Джинсы

Цена: 1800 руб.

[Удалить из корзины](#)

[Вернуться в каталог](#)

Вывод

В данной лабораторной работе я мигрировала свой сайт на Vue, подключив роутер, реализовав работу с внешним API, поделила на компоненты и использовала composable.