

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Фронт-энд разработка

Отчет

Лабораторная работа №3

Выполнил:
Саунин Антон
К33402

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 composable

Ход работы:

Создаю App.vue – главный компонент, с его помощью я отображаю страницу.

Header постоянны1, RouterView же меняется, в зависимости от страницы.

```
src > App.vue
1  <script setup>
2  import Header from './components/Header.vue'
3  import modalCart from './components/modalCart.vue'
4  </script>
5
6  <template>
7  <div>
8    <Header />
9    <router-view></router-view>
10   <modalCart />
11  </div>
12 </template>
13
14 <style scoped></style>
15
```

Создаю компоненты, которые будут отображать каждую страницу.

Header - шапка сайта

productCard - конструктор карточки товара

productCardList - Вывод всех товаров на страницу

CartProduct – конструктор карточки товара для корзины

cartList – вывод всех товаров в корзину

OrderCard - конструктор карточки заказа

OrderList - Вывод всех карточек заказов в профиль

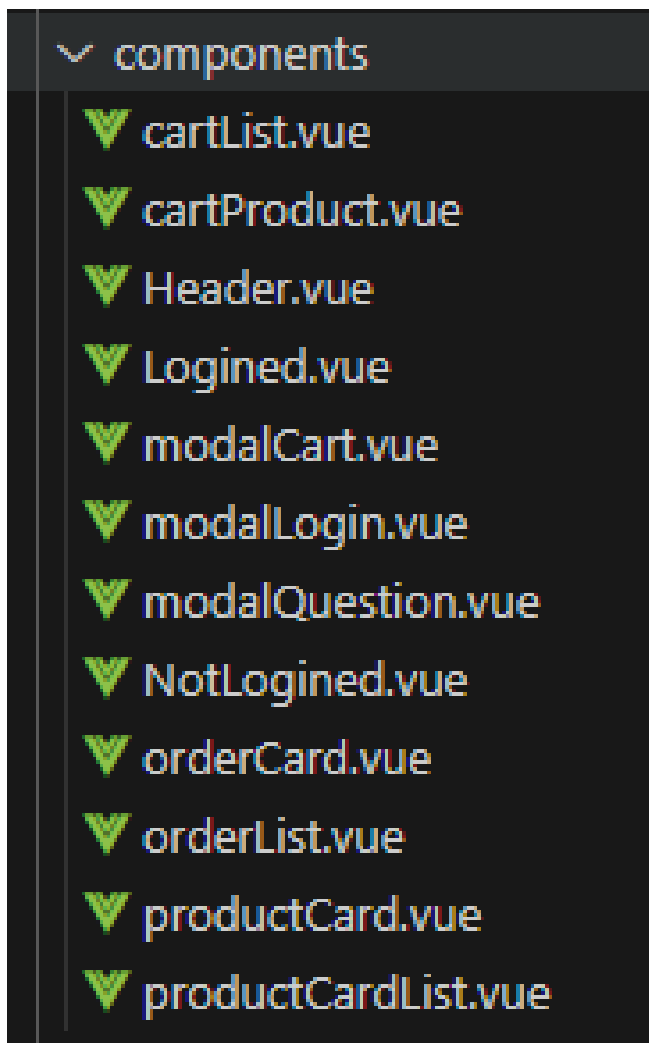
LoggedIn - компонент шапки, отображаемый, когда пользователь
залогинен

NotLoggedIn - компонент шапки, отображаемый, когда пользователь
не залогинен

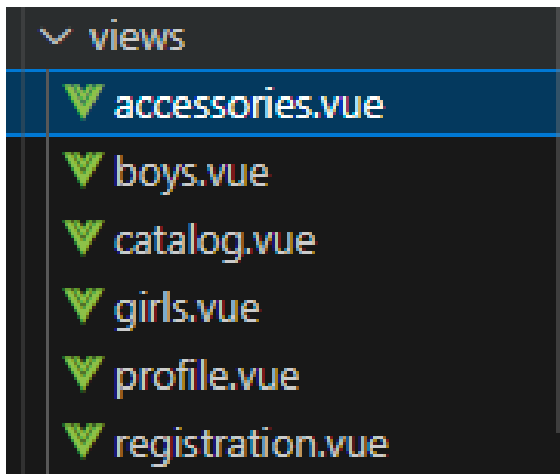
ModalQuestion - модальное окно подтверждения добавления товара в
корзину

modalLogin – модальное окно входа

modalCart – модальное окно корзины



Создаю views для всех страниц.



catalog.vue, boys.vue, girls.vue и accessories.vue отличаются лишь в подключенных скриптах:

```
<div>
  <input
    type="checkbox"
    id="checkbox3"
    style="accent-color: #077015"
    checked="checked"
    value="3"
    v-model="filters"
  />
  <label for="checkbox3">Topwear</label>
</div>
<div class="search-item">
  <input
    @input="onChangeSearchInput"
    type="search"
    id="searchbox"
    style="color: #4a4a4a; width: 70%; border-radius: 10px 0px 0px 10px"
  />
  <button
    style="
      background-color: #077015;
      color: #ffd700;
      width: 25%;
      border-radius: 0px 10px 10px 0px;
    "
  >
    search
  </button>
</div>
</div>
<productCardList :items="items" />
```

profile.vue:

```
<template>
  <div class="container-fluid" style="min-height: 100vh">
    <h2>Welcome!</h2>
    <h3>Here you can see your orders!</h3>
    <div class="d-flex flex-row flex-fill justify-content-start" id="Orders">
      <orderList :orders="orders" />
    </div>
  </div>
</template>

<style scoped>
1 reference
.container-fluid {
  margin: 3%;
}
</style>
```

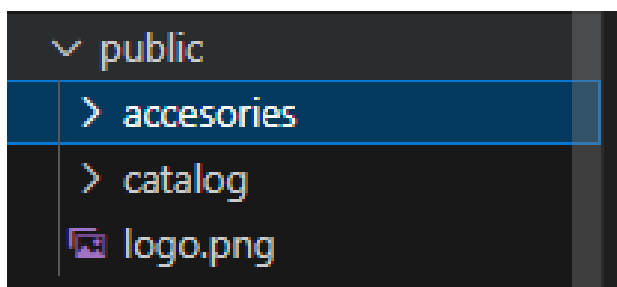
Registration.vue

```
<template>
  <div class="container" style="height: 95vh; margin-top: 5%">
    <form @submit="registerApi">
      <div class="d-flex flex-column justify-content-center" style="padding-left: 40%">
        <div class="d-flex flex-column RegObj">
          <label for="regusername">Username</label>
          <input
            type="text"
            name="id"
            id="regusername"
            style="width: 33%; min-width: 150px"
            autocomplete="off"
            required
          />
        </div>
        <div class="d-flex flex-column RegObj">
          <label for="regphonenumber">Phone number</label>
          <input
            type="tel"
            value="+79"
            pattern="\+7[0-9]{10}"
            name="phonenumber"
            id="regphonenumber"
            style="width: 33%; min-width: 150px"
            autocomplete="off"
            required
          />
        </div>
        <div class="d-flex flex-column RegObj">
          <label for="regemail">Email</label>
          <input
            type="email"
            name="email"
            id="regemail"
            style="width: 33%; min-width: 150px"
            autocomplete="on"
            required
          />
        </div>
        <div class="d-flex flex-column RegObj">
          <label for="regpass">Password</label>
          <input
            type="password"
            name="password"
            id="regpass"
            style="width: 33%; min-width: 150px"
            autocomplete="on"
            required
          />
        </div>
      </div>
    </form>
  </div>
</template>
```

Создаю Роутер, добавляю взаимодействие с ним в index.js

```
const routes = [  
  { path: '/catalog', name: 'catalog', component: catalog },  
  { path: '/boys', name: 'boys', component: boys },  
  { path: '/girls', name: 'girls', component: girls },  
  { path: '/accessories', name: 'accessories', component: accessories },  
  { path: '/registration', name: 'registration', component: registration },  
  { path: '/profile', name: 'profile', component: profile }  
]  
  
const router = createRouter({  
  history: createWebHistory(),  
  routes  
})  
export { router }  
const app = createApp(App)  
app.use(router)
```

Добавляю картинки в public:



Также в логике сайта был использован composable: useApi.js:

```
import axios from 'axios'
import { ref } from 'vue'
const orders = ref([])
const items = ref([])
const SearchQuery = ref('')
const filters = ref([])
export default function useApi() {
  const loginApi = (loginData, username) => {
    axios
      .post('http://localhost:3000/login', loginData)
      .then(function (response) {
        const { data } = response
        const { accessToken, user } = data
        localStorage.accessToken = accessToken
        localStorage.user = JSON.stringify(user)

        username.value = user
        const btnClose = document.getElementById('btnClose')
        btnClose.click()
      })
      .catch(function (error) {
        console.log(error)
        alert('email or password is incorrect')
      })
  }

  const ordersApi = async () => {
    const username = JSON.parse(localStorage.getItem('user')).id
    let url = `http://localhost:3000/orders?username=${username}`
    const { data } = await axios.get(url)
    orders.value = data
  }

  const catalogApi = async (sex = '') => {
    try {
      let checks = []
      let url = `http://localhost:3000/products?q=${SearchQuery.value}`
      if (sex == 'boys') {
        url += `&sex=boys`
      }
      if (sex == 'girls') {
        url += `&sex=girls`
      }
      if (sex == 'acs') {
        url += `&sex=acs`
      }
      for (let check of filters.value) {
        checks.push(check)
        url += `&type=${check}`
      }
    }
  }
}
```

useAuth.js:

```
1 import { ref } from 'vue'
2 import { router } from '@/main'
3 import useApi from '@/composable/useApi'
4
5 const { loginApi } = useApi()
6 const username = ref(localStorage.getItem('user'))
7 export default function useAuth() {
8   const login = async (event) => {
9     event.preventDefault()
10    const inputs = Array.from(event.target.querySelectorAll('input'))
11    const loginData = {}
12
13    for (const input of inputs) {
14      loginData[input.name] = input.value
15    }
16    loginApi(loginData, username)
17  }
18  const logout = () => {
19    localStorage.clear()
20    username.value = ''
21  }
22  const OpenProfile = () => {
23    router.push('/profile')
24  }
25  return {
26    username,
27    login,
28    logout,
29    OpenProfile
30  }
31 }
```


useCart.js:

```
import { ref } from 'vue'
import axios from 'axios'
import 'bootstrap/dist/css/bootstrap.css'
import bootstrap from 'bootstrap/dist/js/bootstrap.bundle.min.js'
const cartProducts = ref([])
let totalPrice = ref(0)
export default function useCart() {
  const addToCart = async (name) => {
    try {
      let url = `http://localhost:3000/products?name_like=${name.replaceAll(' ', '%20')}`
      const { data } = await axios.get(url)
      cartProducts.value.push(data[0])
      console.log(totalPrice.value)
      totalPrice.value += Number(data[0].price.replace('$', ''))
      console.log(totalPrice.value)
    } catch (err) {
      console.log(err)
    }
  }

  const MakeOrder = async (event) => {
    event.preventDefault()
    if (localStorage.getItem('accessToken')) {
      if (Object.keys(cartProducts.value).length !== 0) {
        var user = localStorage.getItem('user')
        var profileU (local var) profileUsername: any
        console.log(profileUsername)
        var date = new Date()
        for (let product of cartProducts.value) {
          console.log(product)
          await axios.post('http://localhost:3000/orders', {
            name: product.name,
            date: date.toString(),
            price: product.price,
            imgSrc: product.imgSrc,
            username: profileUsername
          })
        }
        cartProducts.value = []
        document.getElementById('cartClose').click()
      } else {
        document.getElementById('cartClose').click()
        alert('First add something to cart!')
      }
    } else {
      document.getElementById('cartClose').click()
      new bootstrap.Modal(document.getElementById('modal-reg')).show()
    }
  }

  return {
    addToCart,
    MakeOrder
  }
}
```

Вывод

В данной лабораторной работе я мигрировал свой сайт на Vue, подключив роутер, реализовал работу с внешним API, разумно поделил на компоненты и использовал composable.

