

Национальный исследовательский университет ИТМО



Лабораторная работа №3  
«Разработка одностраничного веб-приложения (SPA) с использованием  
фреймворка Vue.JS»

По дисциплине  
«Фронт-энд разработка»

Выполнил:  
Шевченко Максим  
Группа:  
K33401  
Преподаватель:  
Добряков Д.И.

Санкт-Петербург  
2023 г

## ЗАДАНИЕ

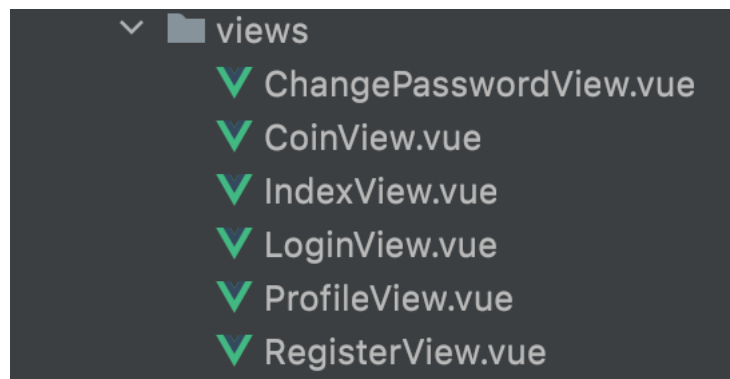
Мигрировать ранее написанный сайт на фреймворк Vue.JS.

Минимальные требования:

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

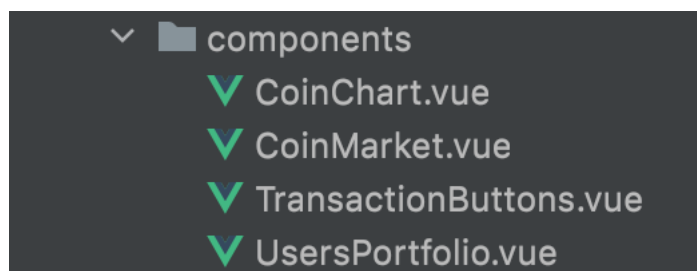
## ВЫПОЛНЕНИЕ

В процессе работы были реализованы следующие представления:



1. Страница смены пароля
2. Страница подробной информации о валюте
3. Приветственная страница и рынок валюты
4. Страница авторизации
5. Портфель пользователя
6. Страница регистрации

И следующие компоненты:



1. График роста валюты
2. Компонент для отображения списка валют и фильтров
3. Кнопки продажи и покупки валюты
4. Список валют пользователя

Роутер:

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

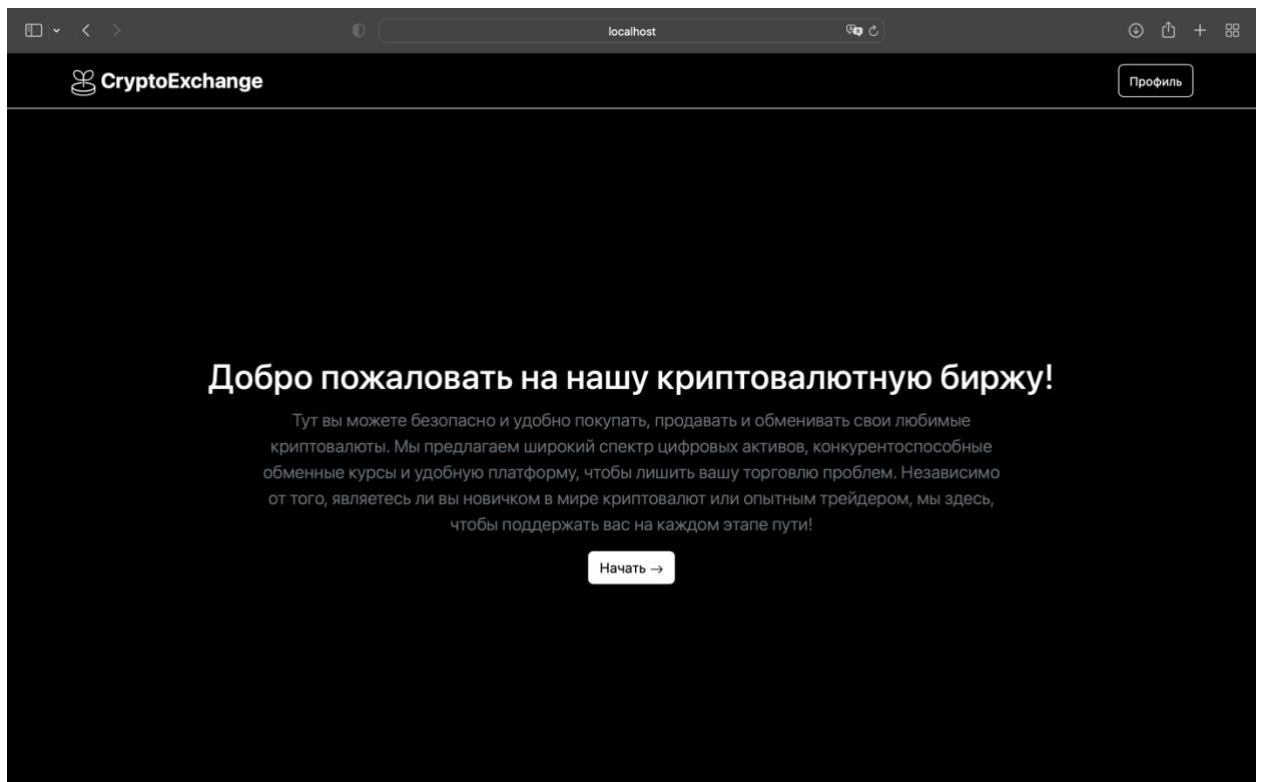
const router = createRouter({
  history: createWebHistory(import.meta.env.BASE_URL),
  routes: [
    {
      path: '/',
      name: 'index',
      component: () => import('@views/IndexView.vue')
    },
    {
      path: '/coin/:id',
      name: 'coin',
      component: () => import('@views/CoinView.vue')
    },
    {
      path: '/login',
      name: 'login',
      component: () => import('@views/LoginView.vue')
    },
    {
      path: '/register',
      name: 'register',
      component: () => import('@views/RegisterView.vue')
    },
    {
      path: '/profile',
      name: 'profile',
      component: () => import('@views/ProfileView.vue')
    },
    {
      path: '/change_password',
      name: 'change_password',
      component: () => import('@views/ChangePasswordView.vue')
    }
  ]
})

export default router
```

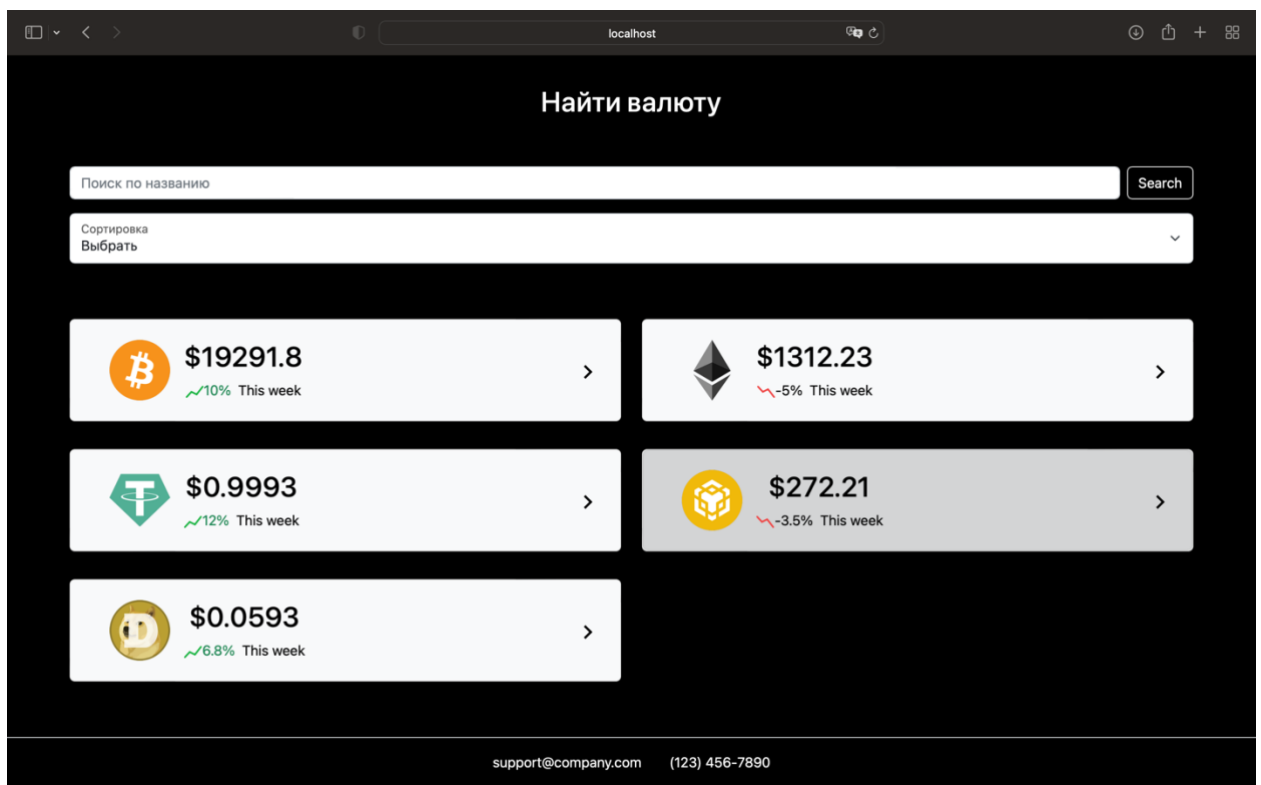
**Представления:**

1. IndexView

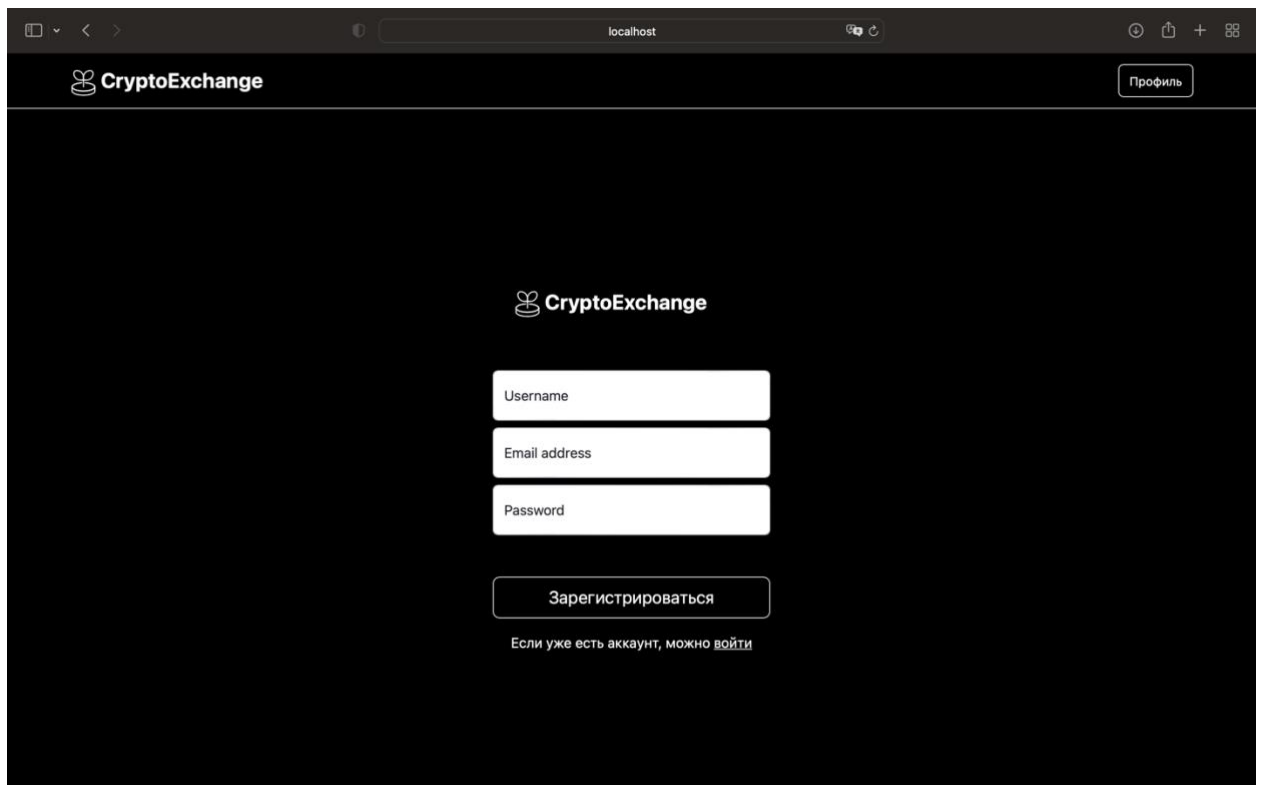
Содержит компонент CoinMarket



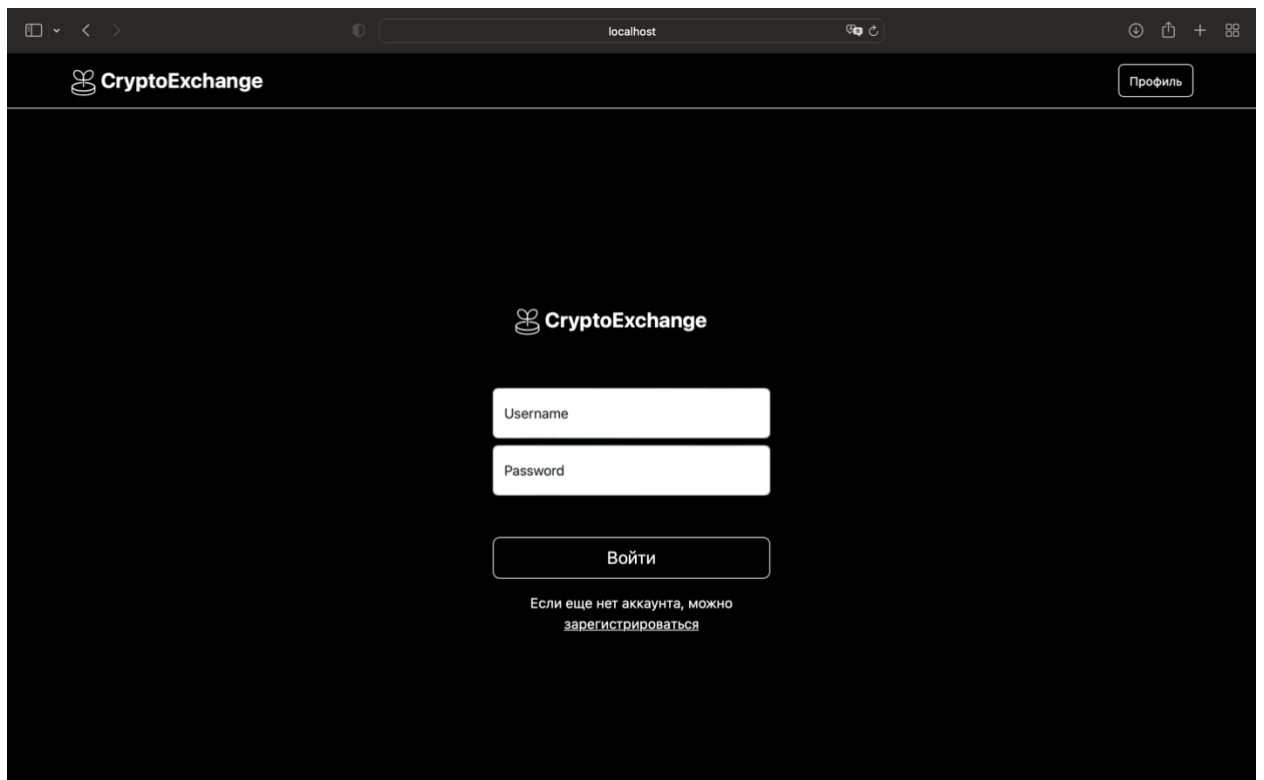
## CoinMarket:



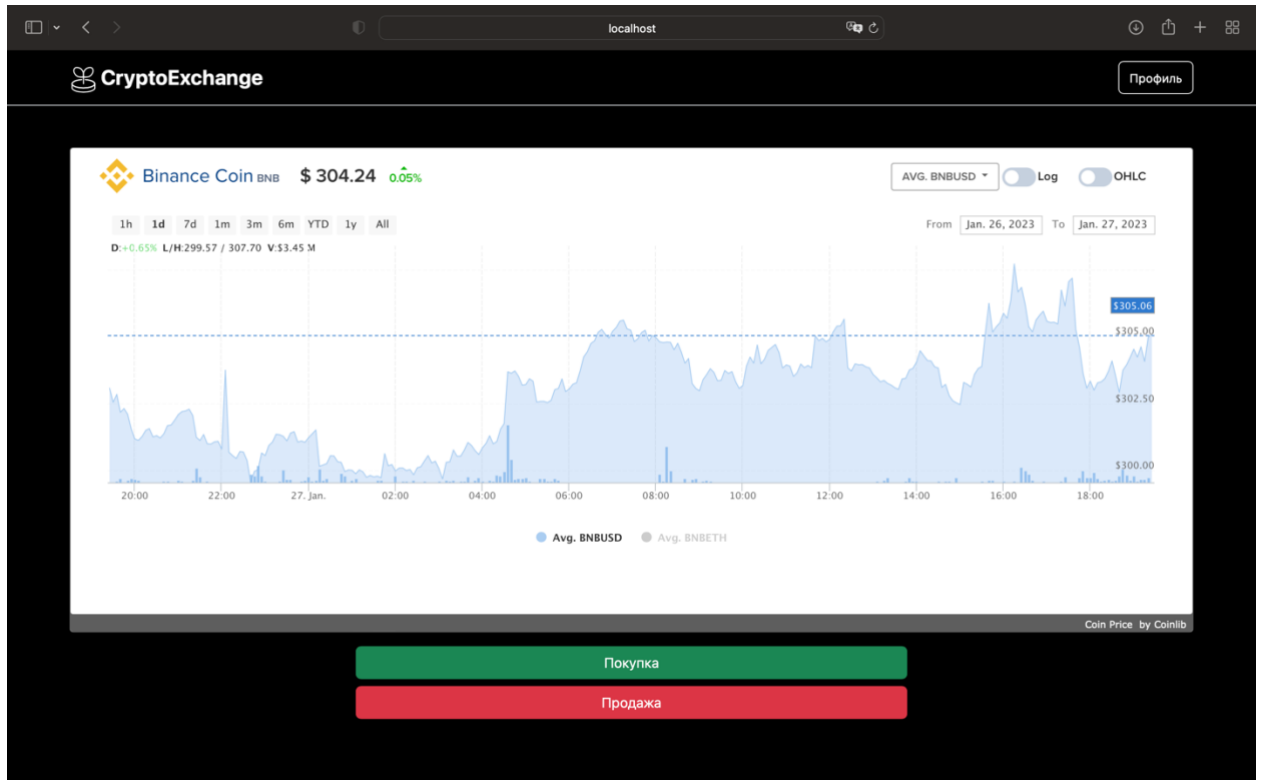
## 2. RegisterView – без деления на компоненты



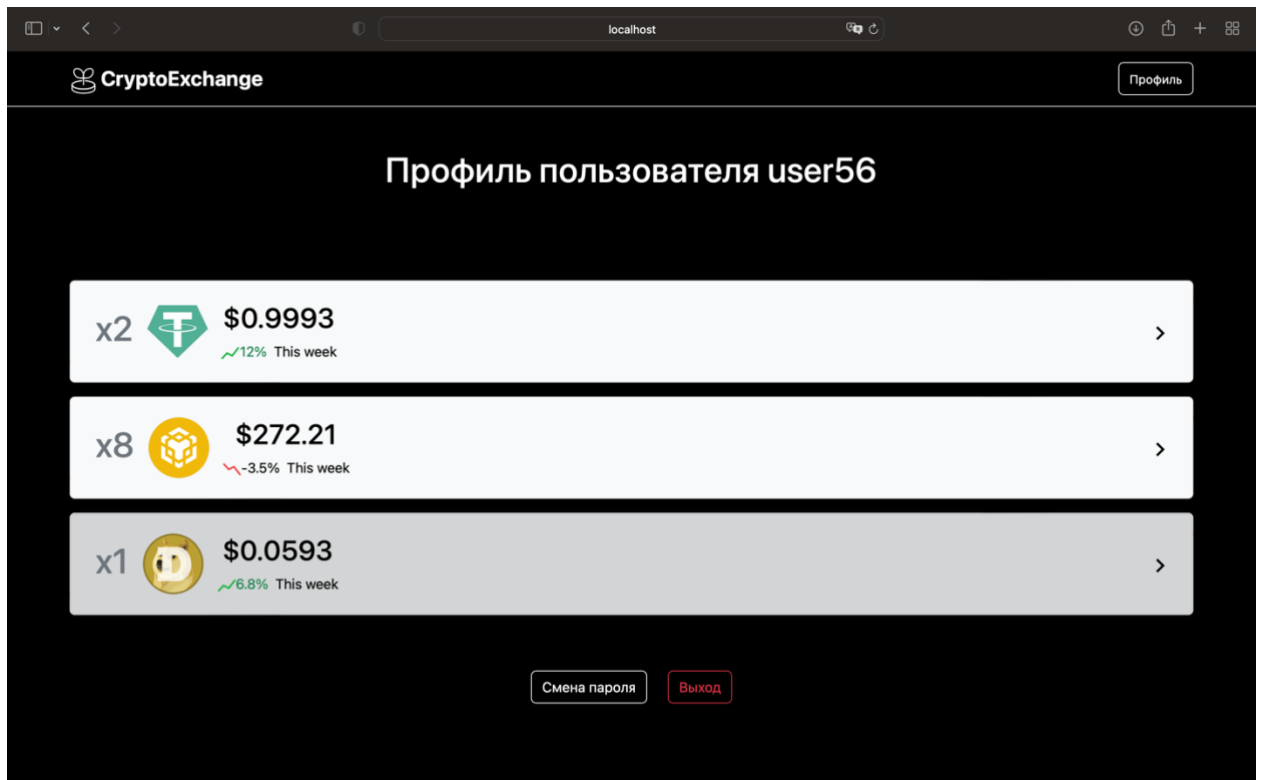
### 3. LoginView – без деления на компоненты



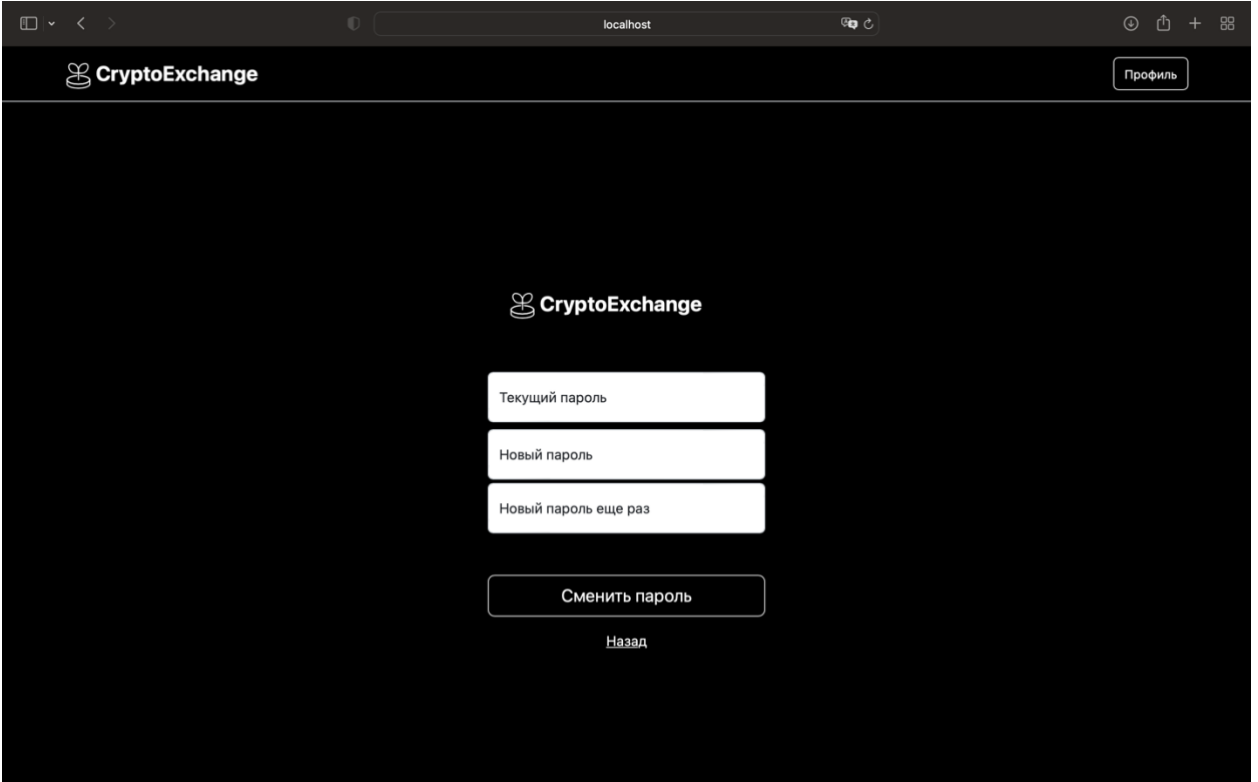
4. CoinView – информация о криптовалюте – содержит компоненты CoinChart и TransactionButtons



5. ProfileView – содержит компонент UsersPortfolio



6. ChangePasswordView



## ВЫВОДЫ

В процессе работы была реализована клиентская часть криптобиржи с помощью фреймворка `vue`. Для запросов я использовал `axios`, также был подключен роутер и, как я думаю, я разумно поделил представления на компоненты.