Web приложение “Bitcoin-кошелек”

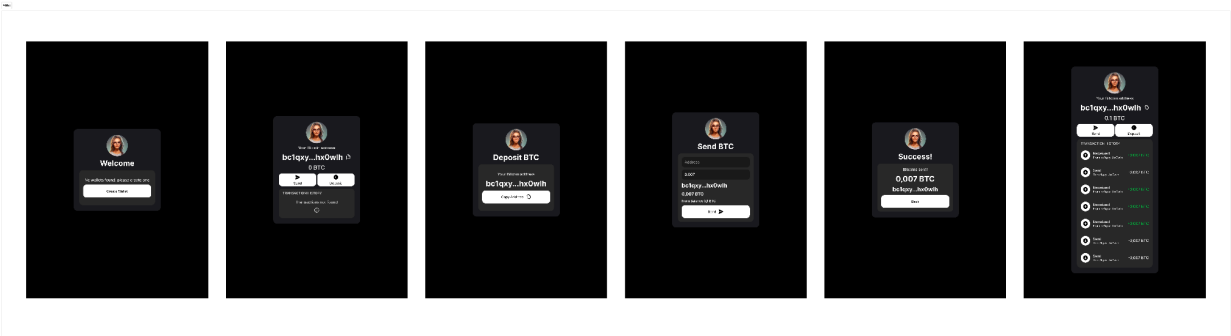
Описание задачи

Разработайте минималистичное Web приложение используя React.js, Next.js, Redux Toolkit для сети Bitcoin Testnet, с помощью которого пользователь сможет создать адрес, посмотреть список транзакций и отправить введенную сумму биткоинов на указанный биткоин-адрес

Основной используемый стек - React, Next.js, RTK Query, Tailwind/CSS Modules

Необходимо реализовать используя Mobile First подход Bitcoin Кошелек.
Дизайн
https://www.figma.com/file/yi9IptRTSWss5zF8ETC3u2/HH-Test-Bitcoin-Wallet?type=design&node-id=0-1&mode=design

Для иконок используйте https://heroicons.com/
Для скелета приложения можно воспользоваться `npx create-next-app@latest`



Верстка должна тянуться и адаптироваться под мобильные экраны (320px) и под широкие до максимального размера макета

1. **Welcome** экран с кнопкой создания, по нажатию создаем BTC Address и Public/Private Keys, которые можно хранить в localStorage, при повторном открытии кошелька сразу загружаем экран с транзакциями
2. **Transactions** список транзакций входящих и исходящих для адреса, если транзакций еще нет выводим текст. По клику на транзакцию должно вести на соответствующую транзакцию в (пример) https://blockstream.info/testnet/tx/2852f290c78b1a12494096dd02e41d692b176caf14a7e60b2be68d90e69df2f5 Зеленую галочку ставим только для транзакций которые имеют статус confirmed

3. **Deposit BTC** - экран на котором показываем данные адреса и кнопку копирования адреса. При успешном копировании меняем текст на кнопке на Copied на несколько секунд, после обратно на исходный текст.
4. **Send BTC (опционально, бонусные баллы)** - экран отправки средств по указанному адресу, по нажатию на Send формируется и отправляется транзакция через API (POST /tx). При успешной отправке показываем Success экран с кнопкой Back которая ведет на список транзакций кошелька. **В случае неуспеха отправки** пользователю должна выводиться соответствующая ошибка выше кнопки отправки. Обработка ошибок не регламентируется и реализуется на усмотрение кандидата. Перед отправкой средств приложение должно проверять, достаточно ли средств для отправки (включая комиссию майнерам). Комиссия майнерам должна составлять 0.000004 **tBTC**. (400 sats)

Для работы с Bitcon нужно использовать библиотеку https://github.com/bitcoinjs/bitcoinjs-lib

Так же вместо **tiny-secp256k1** лучше использовать
https://www.npmjs.com/package/@bitcoin-js/tiny-secp256k1-asmjs

Как создать кошелек -
https://github.com/bitcoinjs/bitcoinjs-lib/blob/master/test/integration/addresses.spec.ts#L109

Как создать транзакцию -
https://github.com/bitcoinjs/bitcoinjs-lib/blob/master/test/integration/transactions.spec.ts#L23

API для отправки транзакции и получения списка транзакций по адресу
https://github.com/Blockstream/esplora/blob/master/API.md

GET /address/:address/utxo
GET /address/:address
GET /address/:address/txs

POST /tx
GET /tx/:txid/hex
GET /tx/:txid

Результат выполнения ТЗ присылайте в виде ссылке на репозиторий Github/Gitlab.

Внешние ресурсы: “необходимо и достаточно”

Если понадобятся тестовые BTC, то их можно бесплатно получить набрав в Google “Bitcoin testnet faucets”.

При возникновении сложностей с получением tBTC следует сообщить об этом и мы пришлём 0.05 **tBTC** на указанный вами адрес.

Для тестирования транзакций в качестве **Bitcoin кошелька для сети Bitcoin Testnet** рекомендуем использовать Electrum, запустив его с флагом --testnet: https://electrum.org/#download

Для исследования hex транзакций можно использовать https://live.blockcypher.com/btc/decodetx/.

Опционально

Реализовать реалтайм получение транзакций через `pollingInterval`

Подсказки:

- Посмотрите, что такое UTXO (unspend transaction outputs). Проще всего сделать так, чтобы каждая очередная транзакция в качестве входов(vin) использовала *все* UTXO, соответствующие вашему адресу.
- Адрес разрабатываемого “кошелька” в любой момент может быть пополнен из другого кошелька. Транзакция пополнения в качестве *одного* из выходов(vout) будет содержать адрес, сгенерированный скриптом (другие выходы - “сдача”). Для создания расходной транзакции вам потребуется определить, какой выход каждой из транзакций пополнения ведёт к адресу вашего кошелька (см. документацию по ссылкам выше). В API blockstream есть способ получить выходные адреса bitcoin-транзакции.

Ликбез по блокчейну (опуская детали)

Блокчейн состоит из цепочки транзакций.

Для совершения операции “перевода” с одного адреса на другой создаётся транзакция, “входом” (vin) которой является первый адрес, а “выходом” второй. Транзакция может содержать несколько входов и несколько выходов.

Разница (в сумме “монет”) между всеми выходами и всеми входами определяет комиссию майнеров; если комиссия нулевая, транзакция не будет обработана майнерами - стоит оставить им некоторую плату.

Для того, чтобы потратить только часть “монет” с некоторого адреса, в качестве дополнительного выхода транзакции устанавливается адрес, находящийся под контролем отправителя. Иначе говоря, “сдача” с операции в явном виде должна быть перечислена себе.