LESSON 8: HTML/JAVASCRIPT FUND ME

Reference: Patrick Collins

INTRODUCTION

In order for web3 to be approachable to the masses, we need to have user-friendly frontends and websites. There are a few challenges that full stack software engineers run into when approaching this problem in the blockchain space.

How do I connect Metamask to my UI? (Or Walletconnect, Phantom, etc.)

How do I execute a transaction with my smart contract from a website?

What are the tools the best of the best are doing?

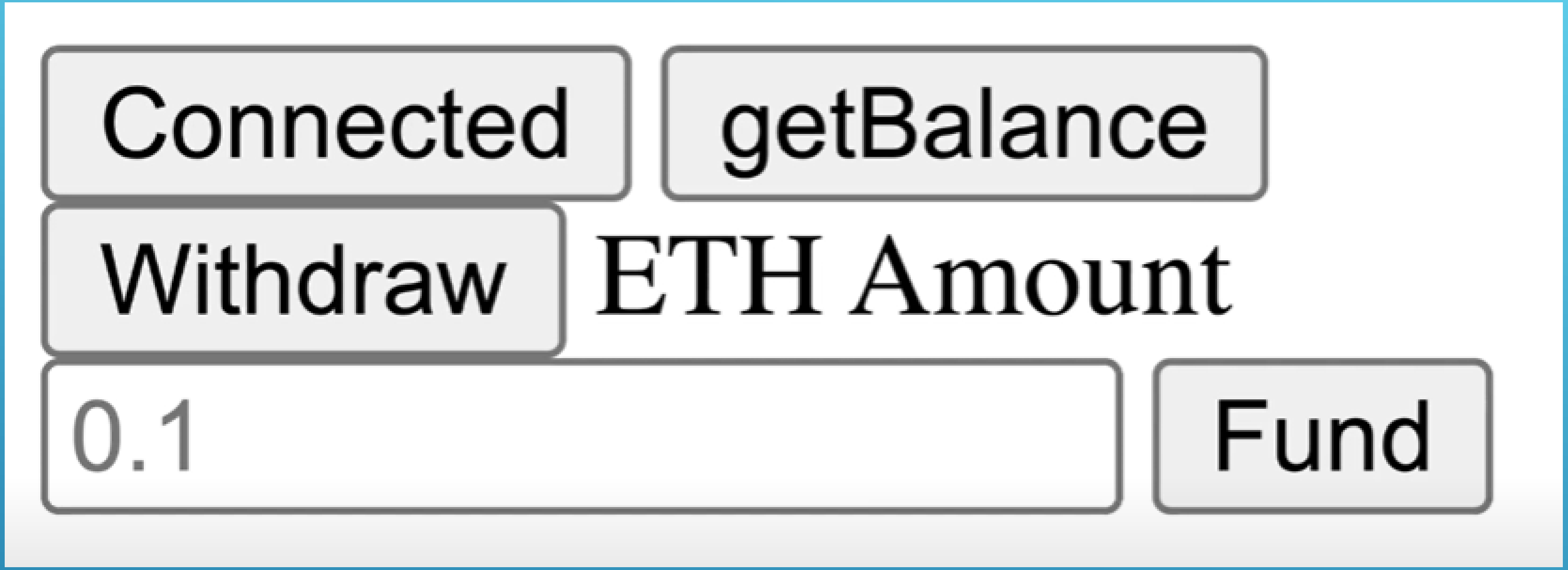
So, in asking myself this problem and trying to figure out what to recommend to developers, I ended up looking at nearly ALL the most popular solutions. So in this article, we are going to:

Understand what is going on in the browser when we want to interact with or send a transaction to a blockchain.

We look at six of the most popular methods to connect to our web3 applications

Give code examples and show what all the biggest players in the space use, so you can use the same tools!

If you'd like to see what some professional frontends look like right now, you can take a look at the Aave or Uniswap website.



ETH Amount

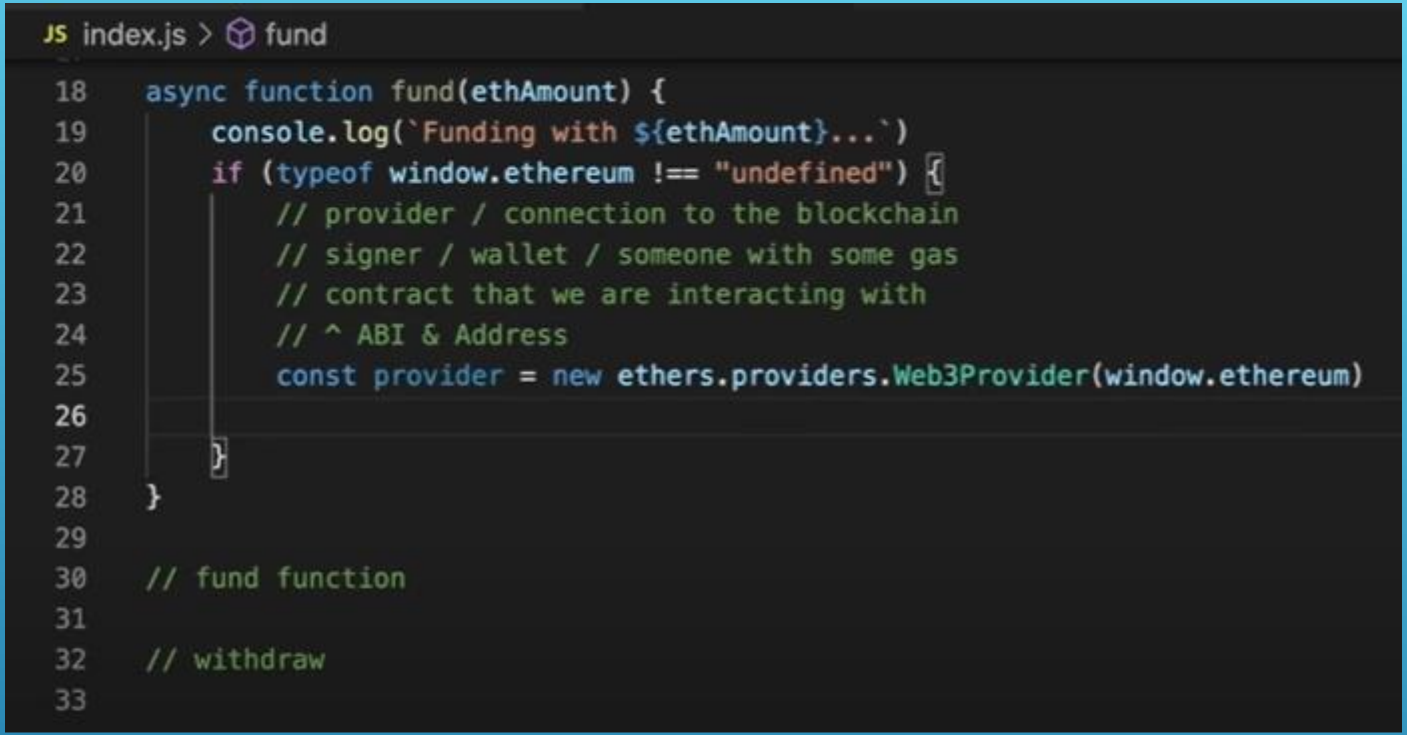
0.1

CONNECTING HTML TO METAMASK

- With a recent update (Mobile v5.3.0), we added a feature that allows you to add one of several popular networks without having to input any details.

- You can find this menu using the following steps:
  - Tap the hamburger icon in the top-left of the screen.
  - Go to 'Settings', and then to 'Networks'.
  - Tap the 'Add Network' button at the bottom of the screen.
  - You should now see a list of networks you can add under the 'Popular' tab. Tap one and follow the prompts to add it to MetaMask. All done!
- This feature will also be added to Extension soon.

SENDING A TRANSACTION FROM A WEBSITE



```javascript
async function fund(ethAmount) {
    console.log(`Funding with ${ethAmount}...`)
    if (typeof window.ethereum !== "undefined") {
        // provider / connection to the blockchain
        // signer / wallet / someone with some gas
        // contract that we are interacting with
        // ^ ABI & Address
        const provider = new ethers.providers.Web3Provider(window.ethereum)

    }
}

// fund function

// withdraw

```

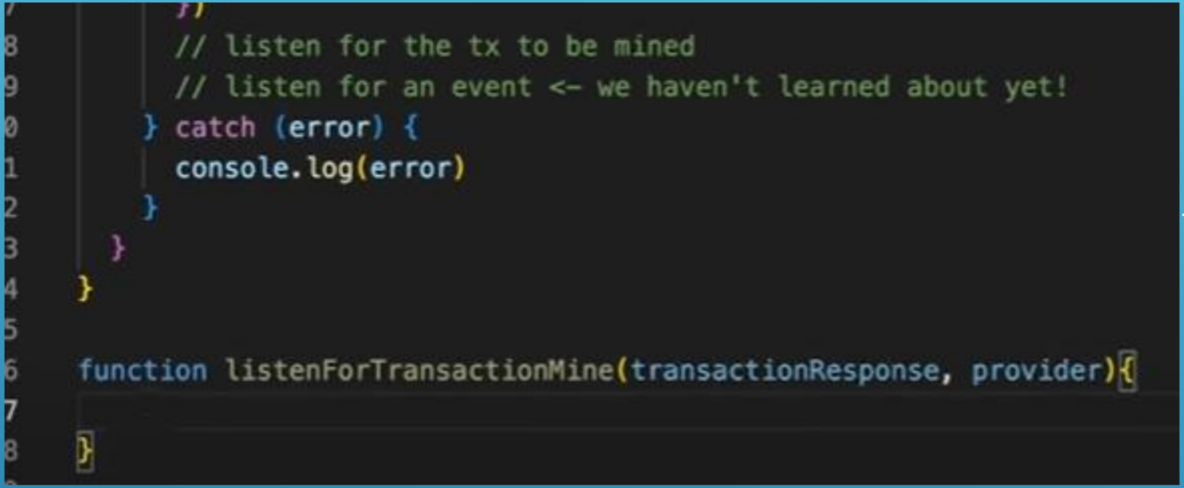
SENDING A TRANSACTION FROM A WEBSITE

MetaMask - RPC Error:

```
[ethjs-query] while formatting ouputs from RPC '{"value":{"code":-32603,"data":{"code":-32000,"message":"Nonce too high. Expected nonce to be 2 but got 4. Note that transactions can't be queued when automining."}}}'
```

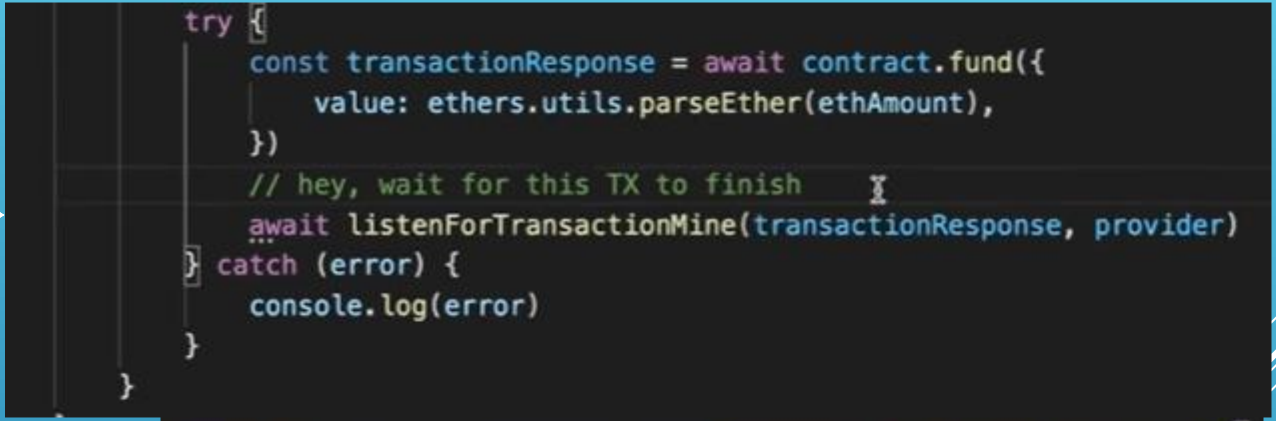
RESETTING AN ACCOUNT IN METAMASK

LISTENING FOR EVENTS AND COMPLETED TRANSACTIONS

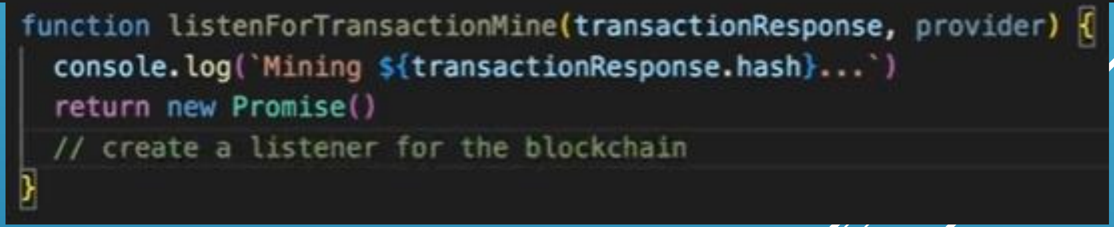


```
        })
        // listen for the tx to be mined
        // listen for an event <- we haven't learned about yet!
      } catch (error) {
        console.log(error)
      }
    }
}

function listenForTransactionMine(transactionResponse, provider){

}
```

→

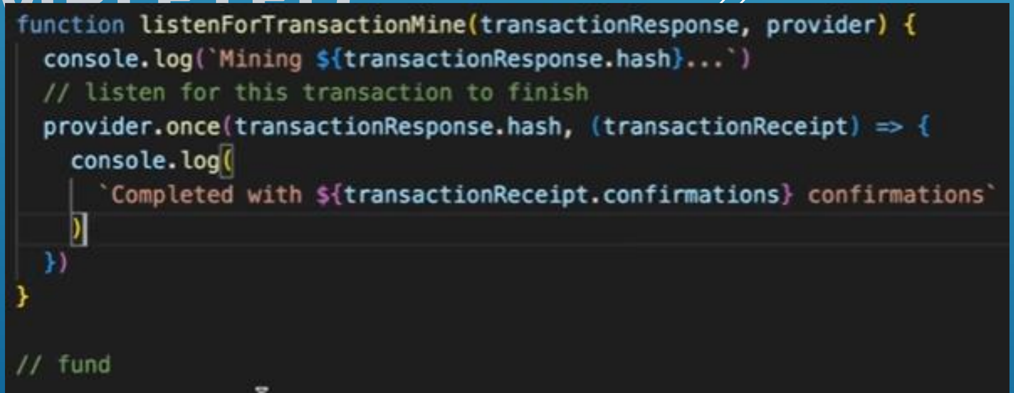


```
        try {
            const transactionResponse = await contract.fund({
                value: ethers.utils.parseEther(ethAmount),
            })
            // hey, wait for this TX to finish
            await listenForTransactionMine(transactionResponse, provider)
        } catch (error) {
            console.log(error)
        }
    }
```



```
function listenForTransactionMine(transactionResponse, provider) {
  console.log(`Mining ${transactionResponse.hash}...`)
  return new Promise()
  // create a listener for the blockchain
}
```

↓



```
function listenForTransactionMine(transactionResponse, provider) {
  console.log(`Mining ${transactionResponse.hash}...`)
  // listen for this transaction to finish
  provider.once(transactionResponse.hash, (transactionReceipt) => {
    console.log(
      `Completed with ${transactionReceipt.confirmations} confirmations`
    )
  })
}

// fund
```