



Lesson 10

NextJS Smart Contract Lottery

Hikmah Nisya - 1103184094

Radzis Araaf Jaya Jamaludin - 1103184234

Raudhatul Rafiqah Assyahiddini - 1103180225

Next JS for smart contract

Dimana NEXT JS untuk SmartContract adalah aplikasi yang desentralisasi yang tersedia berbagai bentuk dan ukuran. Fitur dari aplikasi ini pasar terdesentralisasi, pinjaman kilat, permainan kripto, dan situs NFT sederhana.



Set up

```
npm i web3 nest react react-dom
```

Web3

```
const [web3, setWeb3] = useState(null)
const [address, setAddress] = useState(null)

useEffect() => {
  window.ethereum ?
  ethereum.request ({ method: "eth_requestAccounts" }).then((accounts) => (
    setAddress(requestAccounts[0])
    let w3 = new web3(ethereum)
    setWeb3(w3)
  )).catch((err) => console.log(err))
: console.log("Please install Metamask")
}, [])
```

Abi dan pemanggilan fungsi contract

Setelah set up agar kita bisa memanggil fungsi atau menggunakan fungsi web3 yang sudah di buat, kita perlu menambahkan ABI smartcontract ke situs yang mana dapat memanggil fungsi. Menggunakan REMIX, IDE solidity online selanjutnya menambahkan file baru dan tempel pada smartcontract ke editor. Selanjutnya compile dan akan muncul variabel seperti di samping ini.



CONTRACT

VRFConsumerBaseV2 (VRFConsumerBaseV2)

Publish on Ipfs 

Publish on Swarm 

Compilation Details

 ABI  Bytecode

Memanggil contract view function

Memanggil fungsi kontrak pada Web3 kita harus tahu jenis fungsi yang digunakan agar mengetahui mana saja yang akan digunakan. Untuk fungsi tambahan dan fungsi yang tidak memerlukan gas fee, dapat menggunakan status kontrak baru

```
function mint(){  
  let _price = web3.utils.toWei("1");  
  let encoded = contract.methods.safeMint().encodeABI()  
  
  let tx = {  
    from: address:  
    to : YOUR_CONTRACT_ADDRESS,  
    data : encoded,  
    nonce : "0x00",  
    value : web3.utils.numberToHex(_price)  
  }  
  
  let txHash = ethereum.request({  
    method : 'eth_sendTransaction',  
    params : [tx],  
  }). then((hash) => {  
    alert("you can view your transaction with hash ")  
  }).catch((err) => console.log(err))  
  return txHash  
}
```