

Project Dependencies:

1. node.js, npm
2. npx
3. truffle
4. @truffle/hdwallet-provider
5. @openzeppelin/contracts
6. ethers
7. axios
8. chai, chai-as-promised
9. ipfs-http-client
10. web3
11. web3modal
12. tailwindcss, postcss
13. autoprefixer

Steps:

1. Download node.js and npm from <https://nodejs.org/>
2. Download npx from command prompt: npm i npx
3. Create next.js app using npx:
 - a. Command: npx create-next-app nft-polygon-marketplace
4. Install truffle framework: npm install -g truffle
5. Install dependencies:
 - a. Command: npm install @truffle/hdwallet-provider @openzeppelin/contracts ethers axios chai chai-as-promised ipfs-http-client@50.1.2 web3 web3modal
 - b. Command: npm install --save-dev tailwindcss@latest postcss@latest autoprefixer@latest
6. Configure truffle:
 - a. Create a truffle project structure: truffle init
 - b. Setup Ganache, Metamask
 - c. Create a .secret file to hold wallet seed phrase
 - d. Changes in truffle-config.js file:
 - i. Set up @truffle/hdwallet-provider
 - ii. Set up development network:

```
development: {  
  host: "127.0.0.1",    // Localhost (default: none)  
  port: 7545,          // Standard Ethereum port (default: none)  
  network_id: "**",     // Any network (default: none)  
}
```
 - iii. Set up polygon test network:

- ```
maticmumbai: {
 provider: () => new HDWalletProvider(mnemonic, <rpc url>),
 network_id: 80001,
 confirmations: 2,
 timeoutBlocks: 200,
 skipDryRun: true
}
iv. Set up Polygon main net:
maticmainnet: {
 provider: () => new HDWalletProvider(mnemonic, <rpc url>),
 network_id: '137'
}
e. Set Solidity compiler version: 0.8.3
```

7. Tailwind setup:

- a. In the command prompt run: `npx tailwind init -p`
- b. Modify `global.css` file to:  
    `@tailwind base;`  
    `@tailwind components;`  
    `@tailwind utilities;`

8. Creating `NFT.sol` smart contract

9. Creating `NFTMarket.sol` smart contract

10. Compiling contracts: `truffle compile`

11. Testing contracts: `truffle test`

12. Migrating contracts: `truffle migrate --network <give network name here>`

13. To run front end server: `npm run dev`