This guide will take you through the process of deploying a smart contract to the Polygon Mainnet using Hardhat, a popular Ethereum development environment. Hardhat provides powerful tools for compiling, testing, and deploying smart contracts efficiently.

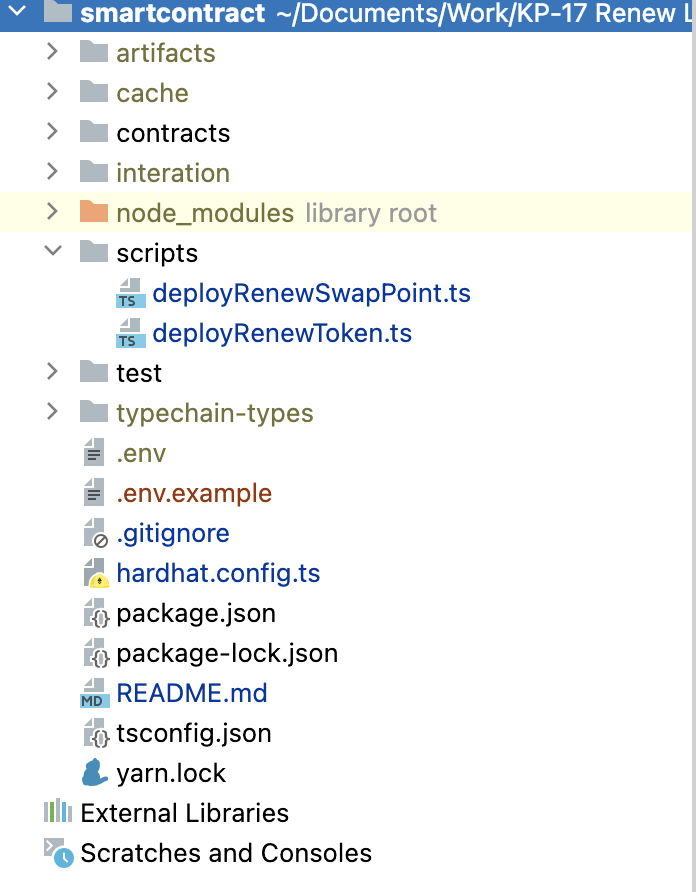
### **Prerequisites**

* Polygon Wallet: You need a Polygon-compatible wallet such as MetaMask.
* Node.js and npm: Ensure you have Node.js and npm installed on your machine.
* Hardhat: Install Hardhat globally on your machine by running
* npm install -g hardhat.
* Smart Contract Code: Have the Solidity code for your smart contract ready.

### **Steps to Deploy**

#### Initialize Your Hardhat Project

Clone source code, and you will see the structure below:



Run those script to setup project:

npm i

cp .env.example .env

Fill all parameters in .env file,

DEPLOY\_PRIVATE\_KEY=430a55b479363317675accb443822cc1c749aecfb215ae44873fe88a9449737f

RENEW\_SWAP\_OWNER\_ADDRESS=

RENEW\_ADDRESS=

RENEW\_SWAP\_RATIO=0.01

RENEW\_TOKEN\_OWNER\_ADDRESS=0x48B8D05d0d2ce8F4059eC7E5097eabC6308E4c9f

* DEPLOY\_PRIVATE\_KEY: Private key of wallet, account which be used to deploy smartcontract, ensure that account has native coin in it (MATIC or ETH)
* RENEW\_SWAP\_OWNER\_ADDRESS: In smart contract RenewTokSwapPoint, setup owner of contract RenewTokSwapPoint.
* RENEW\_ADDRESS: In smartcontract RenewTokSwapPoin, config coin to convert point to Token.
* RENEW\_SWAP\_RATIO: In smartcontract RenewTokSwapPoint, config ratio swap between 1 point to token (1point = **ratio** Token)
* RENEW\_TOKEN\_OWNER\_ADDRESS: In smartcontract RENEWTOKEN, set owner of this smart contract which has admin role

#### Deploy

* Smartcontract RenewTokSwapPoint:
  + Deploy:

npx hardhat deploy --network <network> script/deployRenewSwapPoint.ts

* Smartcontract RenewToken:

Deploy:

npx hardhat deploy --network <network> script/deployRenewToken.ts

1. Verify contract:

Follow this guide:

https://hardhat.org/hardhat-runner/docs/guides/verifying