

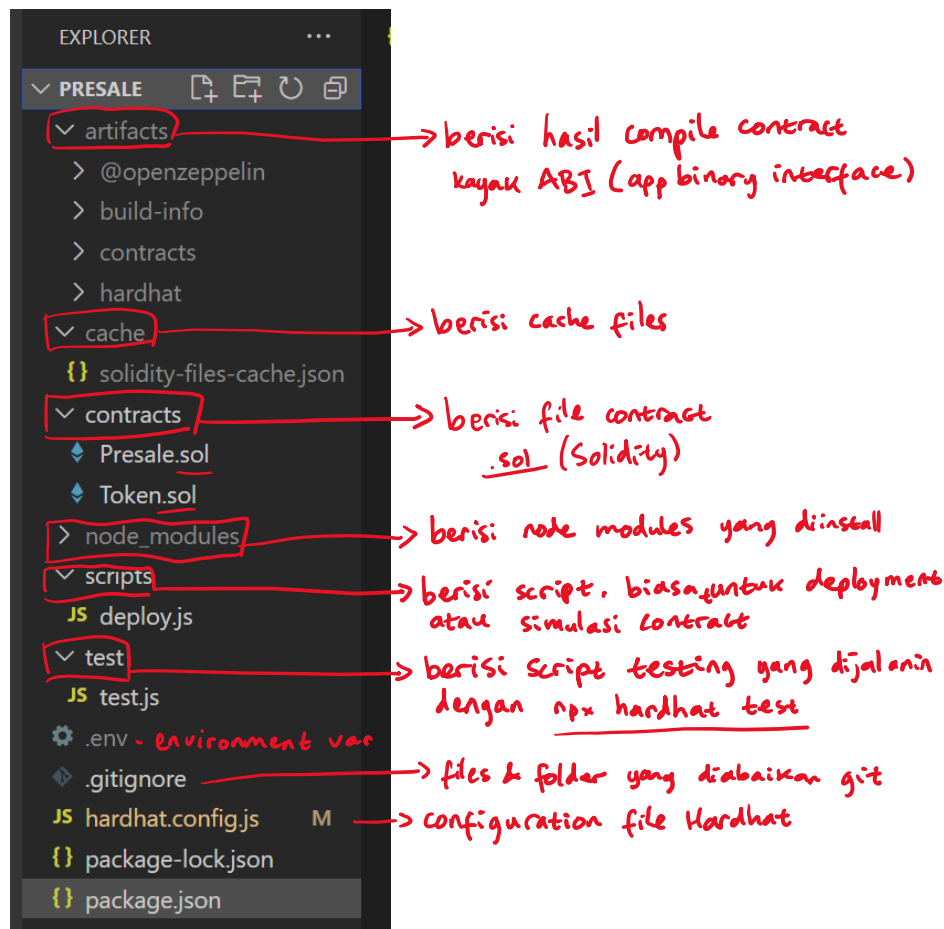
# Understanding Your Project Architecture

Buat yang baru mulai development smart contract di lokal, di sini akan dijelaskan terkait file architecture dalam sebuah project, karena biasa pakai Hardhat untuk devtool, fokusnya akan ke Hardhat project, tapi walaupun pakai devtool lain untuk initiate project, struktur file-nya kurang lebih sama.

## Contents

Files and Folders.....	3
Why <b>.gitignore</b> is crucial?.....	3
.env.....	3
node_modules .....	3
artifacts and cache .....	3
What is <b>package.json</b> ? .....	4
The Hardhat Config .....	4

## Files and Folders



## Why .gitignore is crucial?

.gitignore itu file yang penting banget kalau kalian mau naikin project kalian ke GitHub atau Git lainnya. Karena gunanya untuk meng-exclude atau mengabaikan folder atau file. Kenapa kita ngga mau upload semuanya ke Git? Karena ngga semua file atau folder project kita itu penting atau aman buat di upload.

Biasa .gitignore cukup **.env**, **node\_modules**, **artifacts**, **cache**.

### .env

Ini perlu diabaikan karena masalah keamanan. .env itu isinya environment variables yang biasanya diisi **private key** dan **API keys**. Kalau di upload ke Git, ya tersebar dan wallet-mu bisa diambil orang lain.

```
ENV_VAR=VALUE
```

### node\_modules

Ini perlu diabaikan karena isinya berupa node modules yang kita install di project. Kalau ngga diabaikan, size yang di-upload jadi terlalu besar. Lagian udah ada package.json yang ngerangkum semua modules yang dipakai.

### artifacts and cache

Ini perlu diabaikan karena emang ngga penting aja. Isi artifacts hanya hasil dari compilation smart contract kayak ABI. Bisa di generate saat contract di-compile lagi dengan **npx hardhat compile**. Cache gunanya cuman biar pas rerun script atau recompile bisa lebih cepet. Ini juga bakal di-generate pas contract di-compile.

## What is package.json?

Isinya adalah semua node modules yang diperlukan project beserta versi-nya atau dependencies. Ini penting biar kalau orang lain mau run project, bisa running dengan mudah karena menggunakan modules dengan versi yang sama. Terkadang beda versi, syntax module-nya udah beda. Semua yang ada di **package.json** bisa dengan mudah di-install dengan command **npm i** atau **npm install**.

Dependencies ini ada 2 jenis, dependencies dan devDependencies. Dependencies adalah module yang diperluin saat project udah live dan devDependencies untuk module yang diperlukan saat development saja.

```
{
  "name": "hardhat-project",
  "dependencies": {
    "@nomicfoundation/hardhat-toolbox": "^5.0.0",
    "@openzeppelin/contracts": "^4.9.6"
  },
  "devDependencies": {
    "chai": "^4.3.7",
    "hardhat": "^2.22.3"
  }
}
```

## The Hardhat Config

File **hardhat.config.js** atau **hardhat.config.ts** (tergantung pakai JS atau TS) akan auto-generate kalau run command **npm run hardhat init** atau initialize Hardhat project. Isinya berupa konfigurasi seperti network yang tersedia beserta akunnya, etherscan API key untuk verify contract, gas reporter untuk tahu pemakaian gas, dan lainnya. Berikut contoh config.

```
require("@nomicfoundation/hardhat-toolbox");
require("@nomicfoundation/hardhat-chai-matchers");
require('dotenv').config();

const PRIVATE_KEY = process.env.PRIVATE_KEY ?? "";
const BASE_API_KEY = process.env.BASE_API_KEY ?? "";
const SEPOLIA_API_KEY = process.env.SEPOLIA_API_KEY ?? "";
const ETHERSCAN_API_KEY = process.env.ETHERSCAN_API_KEY ?? "";
const BASESCAN_API_KEY = process.env.BASESCAN_API_KEY ?? "";

module.exports = {
  solidity: {
    version: "0.8.19",
    settings: {
      optimizer: {
        enabled: true,
        runs: 2,
      },
    },
  },
  gasReporter: {
    enabled: false
  },
  networks: {
    hardhat: {
    },
    // local: {
    //   url: "https://rpc-sepolia.rockx.com",
```

```
// accounts: ["0x7971b562d945ff237941e107533807c4ca9218743ee68b78b5d2b735817d140b"]
// }
local: {
  url: "http://127.0.0.1:8545"
},
base: {
  url: BASE_API_KEY,
  accounts: [PRIVATE_KEY]
},
sepolia: {
  url: SEPOLIA_API_KEY,
  accounts: [PRIVATE_KEY]
}
},
etherscan: {
  apiUrl: "https://sepolia.basescan.org/",
  apiKey: BASESCAN_API_KEY
}
};
```