Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiement :

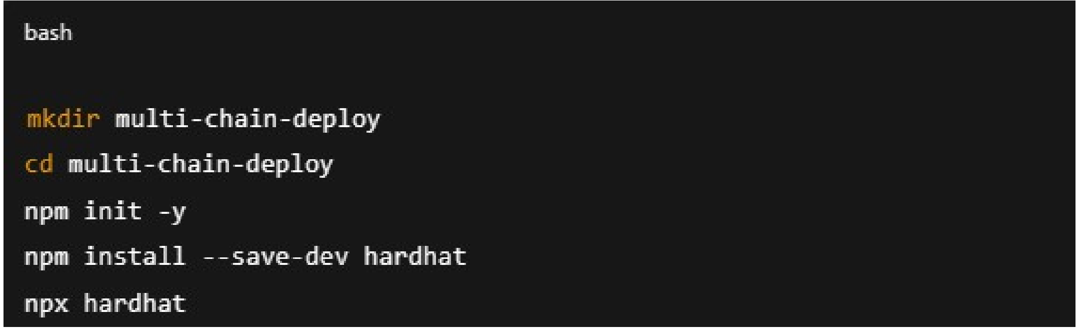
* Coding Phase: Pseudo Code / Flow Chart / Algorithm

Multi-Chain Deploy – BSC or Layer 2 Experience:

Objective

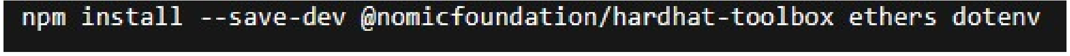
To deploy the same smart contract on multiple blockchains (e.g., Ethereum Sepolia, Binance Smart Chain Testnet, or a Layer 2 like Polygon, Arbitrum, or Optimism) and compare deployment, gas usage, and transaction behavior.

Step 1: Initialize Hardhat Project:



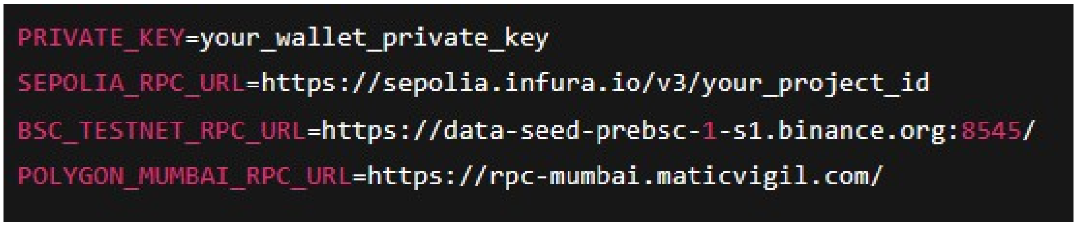
```
bash


mkdir multi-chain-deploy
cd multi-chain-deploy
npm init -y
npm install --save-dev hardhat
npx hardhat
```

Step 2: Install Dependencies :



```
npm install --save-dev @nomicfoundation/hardhat-toolbox ethers dotenv
```

Create a .env file for sensitive data:



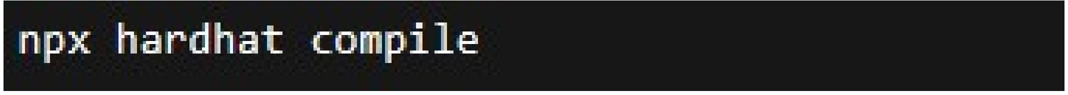
```
PRIVATE_KEY=your_wallet_private_key
SEPOLIA_RPC_URL=https://sepolia.infura.io/v3/your_project_id
BSC_TESTNET_RPC_URL=https://data-seed-prebsc-1-s1.binance.org:8545/
POLYGON_MUMBAI_RPC_URL=https://rpc-mumbai.maticvigil.com/
```

Step 3: Configure Hardhat for Multiple Networks

Step 4: Write a Simple Contract

Step 5: Deploy on Different Networks

(a) Compile

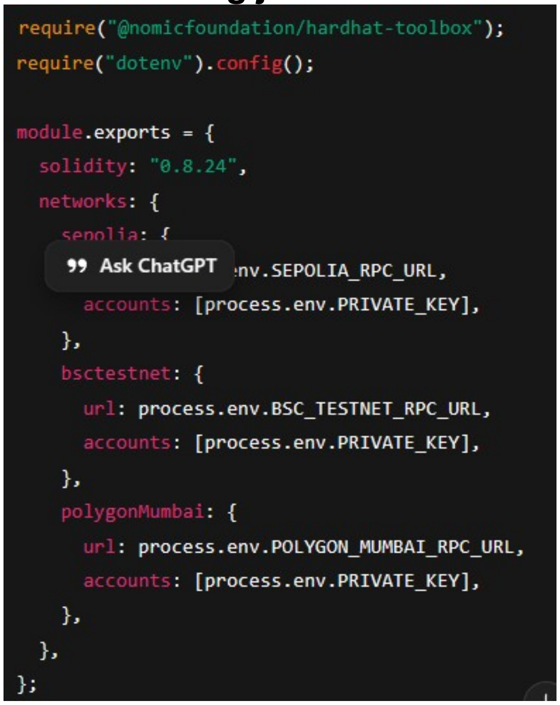


```
npx hardhat compile
```

*As applicable according to the experiment.
Two sheets per experiment (10-20) to be used.*

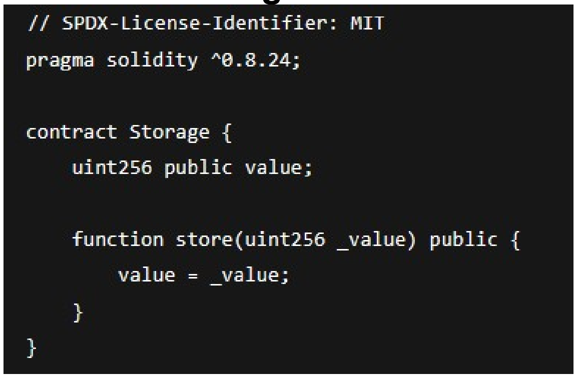
Coding Phase: Pseudo Code / Flow Chart / Algorithm

hardhat.config.js:



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

module.exports = {
  solidity: "0.8.24",
  networks: {
    sepolia: {
                  nv.SEPOLIA_RPC_URL,
      accounts: [process.env.PRIVATE_KEY],
    },
    bsctestnet: {
      url: process.env.BSC_TESTNET_RPC_URL,
      accounts: [process.env.PRIVATE_KEY],
    },
    polygonMumbai: {
      url: process.env.POLYGON_MUMBAI_RPC_URL,
      accounts: [process.env.PRIVATE_KEY],
    },
  },
};
```

contracts/Storage.sol :



```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.24;

contract Storage {
    uint256 public value;

    function store(uint256 _value) public {
        value = _value;
    }
}
```

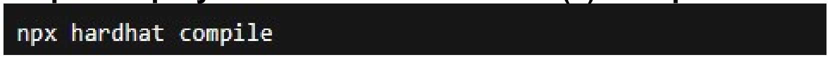
* Softwares used

- Solidity
- Hardhat
- Ether.js
- Metamask
- Infura
- Ethereum sepolia network
- Brave browser

**As applicable according to the experiment.*
Two sheets per experiment (10-20) to be used.

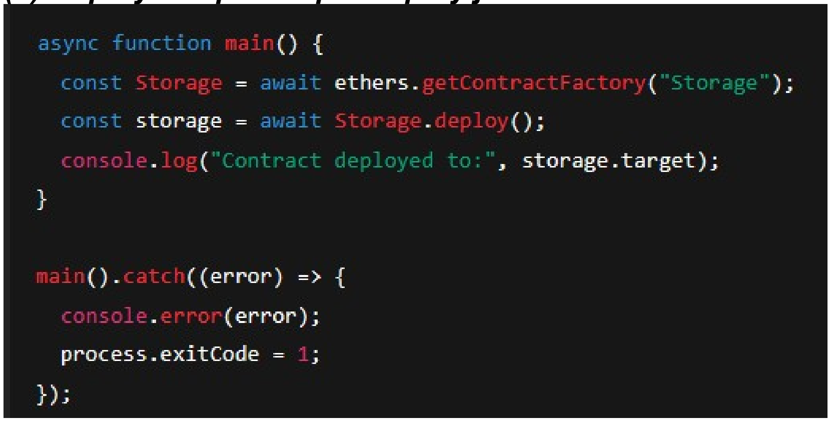
* Testing Phase: Compilation of Code (error detection)

Step 5: Deploy on Different Networks : (a) Compile



```
npx hardhat compile
```

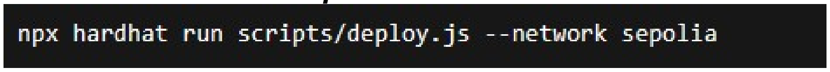
(b) Deploy Script: scripts/deploy.js



```
async function main() {
  const Storage = await ethers.getContractFactory("Storage");
  const storage = await Storage.deploy();
  console.log("Contract deployed to:", storage.target);
}

main().catch((error) => {
  console.error(error);
  process.exitCode = 1;
});
```

(c) Deploy Commands

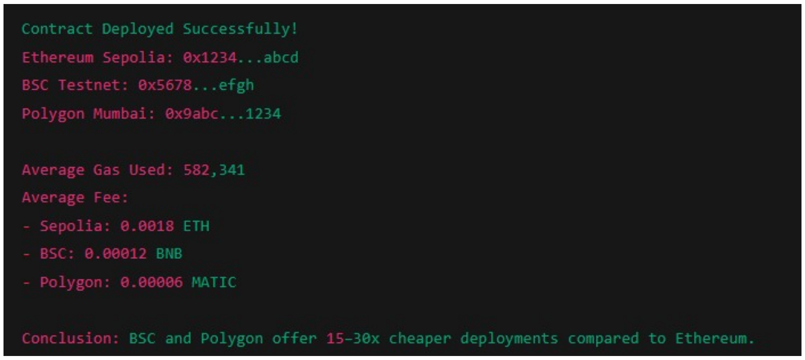
- ***To Ethereum Sepolia:***



```
npx hardhat run scripts/deploy.js --network sepolia
```

Step 6: Verify Deployment : Use Etherscan to view deployed contracts.

* Implementation Phase: Final Output (no error)



```
Contract Deployed Successfully!
Ethereum Sepolia: 0x1234...abcd
BSC Testnet: 0x5678...efgh
Polygon Mumbai: 0x9abc...1234

Average Gas Used: 582,341
Average Fee:
- Sepolia: 0.0018 ETH
- BSC: 0.00012 BNB
- Polygon: 0.00006 MATIC

Conclusion: BSC and Polygon offer 15–30x cheaper deployments compared to Ethereum.
```

* Observations

Deployment on BSC and Polygon Layer 2 was significantly cheaper than Ethereum, confirming Layer 2s and sidechains as cost-efficient deployment environments.

ASSESMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10		
Planning and Execution/ Practical Simulation/ Programming	10		
Result and Interpretation	10		
Record of Applied and Action Learning	10		
Viva	10		
Total	**50**		

Signature of the Student:

Name :

Regn. No. :

Signature of the Faculty:

****As applicable according to the experiment.***
Two sheets per experiment (10-20) to be used.