Blockchain Setup Lab

ODD SEMESTER 2023, 2024

Lab Instructor: Mrs. Lifna C S
Email: lifna.cs@ves.ac.in

Lab Outcome (LO's) :

On successful completion of the course, the learner will be able to:

LOs	Description
LO 1	To understand how blockchain systems (mainly Ethereum) work.
LO 2	To create the genesis block using Puppeth, a CLI tool, and an account using Smart Contract.
LO 3	To create mining blocks, check the account and PoW
LO 4	To use cryptocurrency exchanges and wallets safely.
LO5	To create Gateway to Blockchain Apps.
LO6	To use Blockchain on Mobile App and on Cloud.

No	List of Practical Experiments	LO
1	To install and set up an Ethereum network to create a private Ethereum blockchain for development and testing purposes.	LO1
2	Create a Blockchain Network using Python	LO1
3	Study on Solidity Programming for creating Smart Contracts	LO2
4	Creating a Smart contract in Remix IDE and managing the Smart Contract using Ganache	LO2
$$	Compare and analyze the simulation process of mining blocks and earning Ether (cryptocurrency) rewards in a local blockchain development environment using both (PoW) and (PoA) consensus mechanisms.	LO3
5	To develop a web-based gateway that serves as an entry point for users to access and interact with Ethereum Mainnet.	LO5
6	To develop a blockchain-powered web application using Solidity programming language on Remix IDE and embedding Metamask Wallet	LO5
7	To develop a simple crypto exchange and wallet system for users to trade and securely store cryptocurrencies.	LO4
8	To develop a web application that leverages blockchain technology through DApp development using **[Truffle]**	LO6
9	To deploy and configure a private blockchain network on a cloud platform (AWS or Azure) for testing and development purposes	LO6
10*	Hyperledger Fabric Demo GoLang / JavaScript	LO5, LO6

Note : Experiment **$$ is Removed** in 2024 and Experiment *** is added**

Text Books

1. Mastering Ethereum, Building Smart Contract and Dapps, Andreas M. Antonopoulos Dr. Gavin Wood, O'reilly.
2. Blockchain Technology, Chandramouli Subramanian, Asha A George, Abhillash K. A and Meena Karthikeyen, Universities press
3. Blockchain enabled Applications,Vikram Dhillon, Devid Metcalf, Max Hooper, Apress
4. Building Blockchain Projects, Narayan Prusty, Packt
5. Mastering Blockchain, Second Edition: Distributed ledger technology, decentralization, and smart contracts explained, 2nd Edition, Imran Bashir
6. Mastering Blockchain (Third Edition) - A deep dive into distributed ledgers, consensus protocols, smart contracts, DApps, cryptocurrencies, Ethereum, and more, Imran Bashir Packt Publication.

Sample Codes :

1. Blockchain Applications: A Hands-On Approach, by Arshdeep Bahga, Vijay Madisetti, Paperback – 31 January 2017.
2. Solidity Programming Essentials: A beginner's Guide to Build Smart Contracts for Ethereum and Blockchain, RiteshModi, Packt publication
3. Mastering Bitcoin, PROGRAMMING THE OPEN BLOCKCHAIN, 2nd Edition by Andreas M. Antonopoulos, June 2017, Publisher(s): O'Reilly Media, Inc. ISBN: 9781491954386

Online Resources

1. https://medium.com/@agrawalmanas09/how-to-setup-private-ethereum-blockchain-on-windows-10-machine-ab497e03d6b8
2. https://www.edureka.co/blog/ethereum-private-network-tutorial
3. https://medium.com/publicaio/a-complete-guide-to-using-metamask-updated-version-cd0d6f8c338f
4. https://docs.aws.amazon.com/blockchain-templates/latest/developerguide/blockchain-templates-create-stack.html

Implemented Blockchain Use Cases for Reference

1. DigiLocker - Jai Singhal

Lab - 1 : **Experiment No : 1** **Dated : 3rd & 4th week of July 2024**

Lab Objectives: To build and test a Blockchain using Ethereum in a private setup.

***Lab Outcomes (LO)*:** To understand how blockchain systems (mainly Etherum) work. (LO1)

Task to be performed :

- To install and set up an Ethereum network to create a private Ethereum blockchain for development and testing purposes.
- Follow the instructions in this manual to **set up a Private Ethereum Network**
    a. Choosing a network ID
    b. Choosing a consensus Algorithm
    c. Creating a Genesis Block
    d. Initializing the Geth Database
    e. Setting up Networking
    f. Running the member nodes
    g. Running a Signer (In Clique)

Note : Download the **genesis file** and edit the account details (ie. Public Keys of the peers in the network)

Instructions : (Write the Theory in the following format. Refer to Go Ethereum)

1. Significance of a Private Ethereum Network
2. Steps for creating a Private Ethereum Network

Outcome :

1. Understood the relevance of a Private Ethereum Network
2. Understood how to set up the Private Ethereum Network
3. Performed transactions between the nodes in the network
4. Check the status of transactions in the Transaction pool
5. Prepare a document with Aim, Tasks performed, Program, Output and Conclusion.
6. Submit the hardcopy **by the 1st week of August 2024**
   (As per the instructions, submit a hard copy of the same).

Lab - 2: **Experiment No: 2** **Dated: 1st week of August 2024**

AIM: Create a Blockchain using Python

Lab Objectives: To realize the basic techniques to build intelligent systems

***Lab Outcomes (LO)*:** Demonstrate the concept of Blockchain in real-world Applications (LO4)

Task to be performed :

Step - 1 : Construct Merkle Tree

1. Make a copy of this Google Colab Notebook
2. Try to solve the errors in each of the 4 Programs
3. In the 4th Program - Constructing a Merkle Tree Root Hash, modify the code as follows:
   - Update the transactions list with valid entries.
     eg : transactions = ['A -> B : $10', 'B -> C : $5,'C -> A : $2']
     Sample Transactions to be considered
     T1 : Alice → Bob : $200; T2 : Bob → Dave : $500; T3 : Dave → Eve : $100
     T4 : Eve → Alice : $300; T5 : Roo → Bob : $50
   - Hash the transactions before combining them in the for loop
   - Print all the intermediate hash during the construction of the Merkle Tree Root Hash

Step - 2 : Run a Blockchain with one node

1. Make a copy of this Google Colab Notebook
2. Try to solve the errors in given Program
3. After successful execution of the Program in Colab Notebook.
   - **Add a method, create_Transactions**
   - Mine the block only when the transaction list is not null.
   - Remove the transactions from the list of transactions before mining.
   - **Modify the method, proof_of_work() to search for the golden nonce**
   - Cryptographic Puzzle is to have **"000" leading zeros** in the Block Hash
4. Download the code - **blockchain.py**
5. **Update the code to incorporate the changes in step 3 to the code in step 4.**
6. Follow the steps in **Manual** to demonstrate the working of Blockchain using Flask and Postman.

Step - 3 : Run a Blockchain with 3 peer nodes

1. Download the code from folder, **Lab_2_Step_2**
2. Install requests in the virtual environment created in Step 1. (Follow the instructions)
3. Run the files - **hadcoin_node_5001.py**, **hadcoin_node_5002.py**, **hadcoin_node_5003.py** in 3

different terminals.

4. Open Postman, from each node - invoke **connect_node()** and pass the peers as POST requests.
5. Perform the following functions
   - Add Transactions - invoke add_transactions() as a POST request.
   - mining - mine_block(),
   - fetch the chain - get_chain(),
   - replace the longest chain - replace_chain()
6. Modify the code such that transactions are removed after they are added to the block.

Tools & Libraries used :

- Install ***Flask*** : pip install Flask
- Download ***Postman*** from https://www.postman.com/
- Python Libraries : **datetime, jsonify, hashlib, uuid4, urlparse, request**
- **Install requests :** pip install requests==2.18.4

Instructions : (Write theory for the following topics)

1. Cryptographic Hash functions in Blockchain
2. What is a Merkle Tree? How does a Merkle Tree work?
3. What is a Cryptographic Puzzle and explain the Golden Nonce
4. Benefits and Use cases of Merkle Tree
5. What is a Blockchain? Explain the process of Mining
6. How to check the validity of blocks in a Blockchain
7. Challenges in P2P networks
8. How transactions are performed on the network?
9. Explain the role of mempools
10. Write briefly about the libraries and the tools used during implementation.

Outcome :

1. Understood the challenges in P2P networks, how transactions are performed and how a miner mines a block to be added in a blockchain.
2. Implemented a Cryptocurrency in Python using Flask, Postman and Python libraries such as datetime, jsonify, hashlib, uuid4, urlparse, request.
3. Successfully mined the blocks among a P2P network with 3 nodes.
4. Performed transactions via the network.
5. Successfully updated the block across the network
6. Attach the Theory, Program, and Output **by the 2nd week of August 2024**
   (As per the instructions, submit a hard copy of the same).

Lab - 3: **Experiment No: 3** **Dated: 3rd week of August 2024**

AIM: Study on Solidity Programming for creating Smart Contracts

Lab Objectives: To explore Blockchain concepts.

***Lab Outcomes (LO)*:** Design Smart Contract using Solidity (LO2)

Task to be performed :

Step - 1 : **Hands on Solidity Programming Assignments for creating Smart Contracts**

1. Go to **LearnETH** Tutorials provided by **Remix IDE**
2. Explore through **Solidity Beginner Course**
3. Complete all the 19 Assignments provided with the Course

Step - 2 : **Deploying a Voting/Ballot Smart Contract**

1. Open **Remix IDE**
2. Under **Workspaces,** open **contracts** folder
3. Open **Ballot.sol**, contract.
4. Understand **Ballot.sol** contract.
5. Deploy the contract by changing the Proposal name from **bytes32 → string**

Refer : https://www.youtube.com/watch?v=GB3hiiNNDjk

Tools & Libraries used : Remix IDE, **Ethers**

Instructions : (Prepare for viva for the following topics)

1. Primitive Data Types, Variables, Functions - pure, view
2. Inputs and Outputs to Functions, Visibility, Modifiers and Constructors
3. Control Flow : if-else, loops
4. Data Structures : Arrays, Mappings, structs, enums, Data Locations
5. Transactions : Ether and wei, Gas and Gas Price, Sending Transactions
6. What is the relevance of require statements in the functions of Solidity Programs?
7. Understand the keywords mapping, storage and memory
8. Why bytes32 instead of string?

Outcome :

1. Understood the basics of Solidity Programming in writing Smart Contracts and Deploying them on the Remix VM. Successfully performed the Assignments given in the Tutorial.
2. Understood the logic of Ballot contract in Solidity and Successfully performed the deployment of the contract.
3. Prepare a document with Aim, Tasks performed, Program, Output and Conclusion.
4. Submit the hardcopy **by the 4th week of August 2024**
   (As per the instructions, submit a hard copy of the same).

Lab - 4: **Experiment No: 4** **Dated: 5th week of September 2024**

AIM: Implement the Blockchain platform Ganache

Lab Objectives: To explore Blockchain concepts.

***Lab Outcomes (LO)*:** Design Smart Contract using Solidity (LO2)

Task to be performed: **[Follow the GitHub repository for the experiment]**

1. Install Ganache
2. Connect Ganache Accounts with Metamask
3. Connect Remix IDE with Metamask
4. Create a Simple Solidity Smart Contract **based on the MiniPoject chosen**
5. Compile and Deploy the Smart Contract via Ganache Accounts added to Metamask.
6. Check the transaction details on the Ganache Environment
7. Interact with the smart contract

Instructions : (Prepare for viva for the following topics)

1. What is a Ganache?
2. List the steps involved in connecting Ganache Environment with a Metamask and Remix IDE for performing transactions.

Outcome :

1. Understood the steps for embedding the Metamask wallet with Remix IDE and perform transactions
2. Successfully performed the transactions on the Remix IDE via the account from Metamask Wallet
3. Prepare a document with the Aim, Tasks performed, Program, Output, and Conclusion.
4. Submit the hard copy **by the 1st week of October 2024**

(As per the instructions, submit a hard copy of the same)

Lab - 5 **Experiment No: 5** **Dated: 1st week of October 2023**

AIM: Compare and analyze the simulation process of mining blocks and earni4g Ether (cryptocurrency) rewards in a local blockchain development environment using both Proof of Work (PoW) and Proof of Authority (PoA) consensus mechanisms.

Lab Objectives: To explore Blockchain concepts.

***Lab Outcomes (LO)*:** To create mining blocks, check the account and PoW (LO3)

Task to be performed:

- Explain PoW and PoA with respective to the parameters:
    a. Mining process
    b. Earning Ether rewards
    c. Analysis

Instructions : (Prepare for viva for the following topics)

- Write a comparative study on PoW and PoA in tabular format.

Outcome :

1. Understood the concept of PoW and PoA
2. As per the understanding prepared the comparative study of PoW and PoA.
3. Prepare a document with the Aim, Tasks performed, Program, Output, and Conclusion.
4. Submit the hard copy **by the 2nd week of October 202**

Lab - 6: **Experiment No: 6** **Dated: 4th week of September 2023**

AIM: To develop a web-based gateway that serves as an entry point for users to access and interact with Ethereum Mainnet.

Lab Objectives: To understand and apply the concepts of keys, wallets.

***Lab Outcomes (LO)*:** To create Gateway to Blockchain Apps.(LO5)

Task to be performed :

1. Refer the **Google Colab Notebook**
2. Connect to the Ethereum Mainnet via Etherscan.io and Infura.io

Instructions : (Prepare for viva for the following topics)

1. What is Etherscan? Explain the working of Etherscan. Why do we need Etherscan?
2. Steps to fetch information from Etherscan.io
3. Steps to fetch information from Infura.io

Outcome :

1. Understood how to connect to the Etherscan.io and Infura.io to fetch information about Ethereum Mainnet.
2. Prepare a document with Aim, Tasks performed, Program, Output and Conclusion.
3. Submit the hardcopy **by the 5th week of September 2023**
   (As per the instructions, submit a hard copy of the same).

Lab - 7: **Experiment No: 7** **Dated: 4th week of September 2023**

AIM: To develop a blockchain-powered web application using Solidity programming language on Remix IDE and embedding Metamask Wallet

Lab Objectives: To understand and apply the concepts of keys, wallets.

***Lab Outcomes (LO)*:** To create Gateway to Blockchain Apps.(LO5)

Task to be performed: **[Follow the GitHub repository for the experiment]**

1. Set Up MetaMask:
   a. Install MetaMask
   b. Create or Import an Account:
   c. Fund Your Wallet : Sepolia Testnet (0.5 ETH per day) / RSK Testnet (0.05 RBTC per day)
2. Connect the Sepolia Testnet / RSK Testnet to Remix IDE
3. Create a Simple Solidity Smart Contract **based on the MiniPoject chosen**
4. Compile and Deploy the Smart Contract.
5. Check the transaction details on the RSK Explorer
6. Interact with the smart contract

Instructions : (Prepare for viva for the following topics)

1. What is a Metamask?
2. What is a test net?
3. List the steps to connect a Metamask with a Remix IDE for performing transactions.

Outcome :

1. Understood the steps for embedding the Metamask wallet with Remix IDE and perform transactions
2. Successfully performed the transactions on the Remix IDE via the account from Metamask Wallet
3. Prepare a document with the Aim, Tasks performed, Program, Output, and Conclusion.
4. Submit the hard copy **by the 5th week of September 2023**
   (As per the instructions, submit a hard copy of the same)