School: ..... Campus: .....

Academic Year: ..... Subject Name: ..... Subject Code: .....

Semester: ..... Program: ..... Branch: ..... Specialization: .....

Date: .....

Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiement : Build DeFi – Using Uniswap or Aave Testnets

Objective/Aim:

To understand the working of Decentralized Finance (DeFi) protocols by building and interacting with liquidity pools and lending mechanisms on Uniswap or Aave test networks.

Apparatus/Software Used:

- Laptop

- Node.js and npm
- Visual Studio Code (VS Code)
- MetaMask Wallet (connected to Ethereum Testnet like Sepolia or Goerli)
- Web Browser (Chrome / Brave)
- Web3.js / Ethers.js Library
- Uniswap / Aave Testnet Interface
- Test Tokens (ETH, DAI, USDC) via Faucets

Theory/Concept:

- Decentralized Finance (DeFi) is a financial system built on blockchain technology that removes intermediaries like banks by using smart contracts.
  Two major DeFi platforms are:
- Uniswap:
  A decentralized exchange (DEX) that enables users to swap tokens using automated liquidity pools instead of centralized order books.
- Aave:
  A decentralized lending and borrowing protocol allowing users to lend crypto assets and earn interest or borrow by providing collateral.

Procedure:

1. Install and set up MetaMask with a test network like Sepolia or Goerli.
2. Use a faucet to obtain test ETH and tokens such as DAI or USDC.
3. Open Uniswap Testnet Interface or Aave Testnet Interface in your browser.
4. Connect MetaMask wallet to the testnet DApp.
5. For Uniswap:
6. Select token pairs (e.g., ETH/DAI).
7. Add liquidity by supplying both tokens.
8. Perform a test token swap and confirm the transaction in MetaMask.
9. Observe liquidity pool share and swap rate updates.
10. For Aave:
11. Supply a token (e.g., DAI) to the lending pool.
12. Borrow another token (e.g., USDC) using supplied assets as collateral.
13. Observe interest rate and health factor changes.
14. Interact with smart contracts using Web3.js/Ethers.js scripts to fetch pool or lending data.
15. Record transaction hashes and analyze transaction confirmation on Etherscan Testnet.
16. Compare gas usage and transaction speed between swap and lending actions.

Observation Table:

- MetaMask successfully connected to the Uniswap and Aave testnets.
- Token swaps and liquidity operations were completed on Uniswap.
- Lending and borrowing operations were performed successfully on Aave.
- Smart contract interactions were verified on Etherscan testnet with correct state changes.
- Gas fees varied based on network congestion and action type.

ASSESSMENT

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

Signature of the Student:

Name :

Regn. No.

Signature of the Faculty: