



Centurion  
UNIVERSITY

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 Experiment : Smart Libraries – Libraries and Proxy Contracts**

### \* **Coding Phase: Pseudo Code / Flow Chart / Algorithm**

- | **Define a Library** with reusable logic (e.g., math operations).
- | **Deploy the Library** (if external).
- | **Create a Proxy Contract** that holds state variables.
- | **Implement a Logic Contract** (the actual contract with functions).
- | **Proxy Delegates Calls** from users to the logic contract using delegatecall.
- | **User Interacts** only with the proxy contract address, ensuring upgradeability.

### \* **Software used**

1. MetaMask Wallet
2. Remix IDE.
3. MS Word.
4. Brave for researching.

Page No.....

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

## \* Implementation Phase: Final Output (no error)

Deploy the **Library** (e.g., SmartLibrary.sol).

Deploy the **Logic Contract** using the library functions.

Deploy the **Proxy Contract** that points to the logic contract's address.

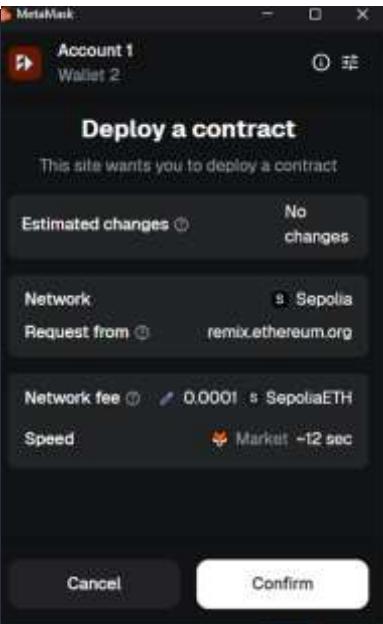
Call a function (e.g., addNumbers(2,3)) from the proxy.

Proxy forwards the call via delegatecall → executes in the logic contract.

Output (e.g., 5) is returned to the user, but the state is stored in the proxy.

If logic needs updating, deploy a new **Logic V2 contract** and update the proxy to point to it.

Users continue using the same proxy address, but new logic executes.



The screenshot shows the MetaMask extension interface. A dialog box titled "Deploy a contract" is open, prompting the user to confirm the deployment of a contract. It displays the following information:

- Estimated changes:** No changes
- Network:** Sepolia
- Request from:** remix.ethereum.org
- Network fee:** 0.0001 SepoliaETH
- Speed:** Market (~12 sec)

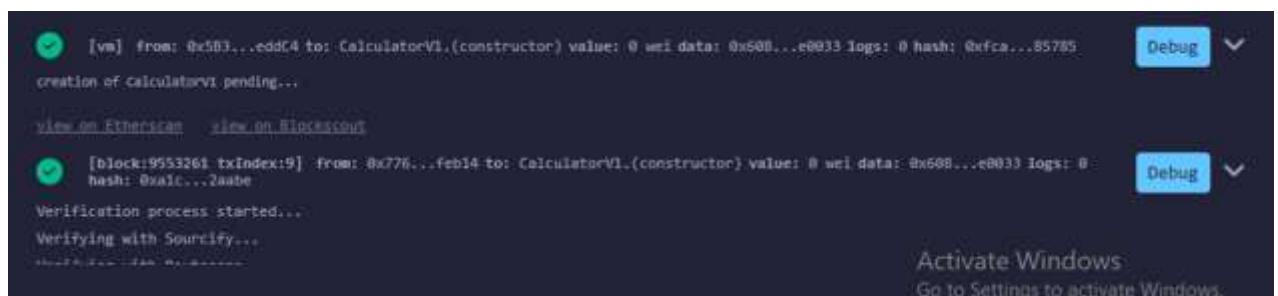
At the bottom of the dialog are "Cancel" and "Confirm" buttons.

On the left side of the image, there is a code editor window showing Solidity code for a library and a logic contract. The library, named MathLib, contains two functions: add and sub. The logic contract, named CalculatorV1, uses MathLib and defines its own addNumbers function.

```

1 // SPDX-License-Identifier: MIT
2 pragma solidity ^0.8.7;
3
4 /**
5  * SMART LIBRARY
6  */
7 library MathLib {
8     function add(uint256 a, uint256 b) internal pure returns (uint256) {    // infinite gas
9         return a + b;
10    }
11
12    function sub(uint256 a, uint256 b) internal pure returns (uint256) {    // infinite gas
13        require(b <= a, "Subtraction underflow");
14        return a - b;
15    }
16 }
17
18 /**
19  * LOGIC CONTRACT V1
20 */
21 contract CalculatorV1 {
22     using MathLib for uint256;
23     uint256 public result;
24
25     function addNumbers(uint256 _a, uint256 _b) public {    // infinite gas
26         result = add(_a, _b);
27     }
28 }

```



The screenshot shows the Ethereum Remix IDE interface during the deployment process of the CalculatorV1 contract. The terminal-like output pane displays the following logs:

- [vm] from: 0x503...edc4 to: CalculatorV1.(constructor) value: 0 wei data: 0x500...e0033 logs: 0 hash: 0xfc...85785 creation of calculatorv1 pending...
- [block:9553261 txIndex:9] from: 0x726...feb14 to: CalculatorV1.(constructor) value: 0 wei data: 0x608...e0033 logs: 0 hash: 0x1c...2aafe Verification process started...
- Verifying with Sourcify...

At the bottom right, there is an "Activate Windows" message: "Go to Settings to activate Windows".

## \* Observations:

- Marketplace requires **approval** before listing NFTs.
- Each NFT listing is tracked by a unique listingId.
- Buyers must pay **exact ETH** price, otherwise transaction fails.
- Smart contract prevents **self-purchase** by the seller.
- Sellers can cancel or update their NFT listings anytime.

## ASSESSMENT

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		
<b>Total</b>	<b>50</b>		

***Signature of the Student:***

Name :

Regn. No. :

***Signature of the Faculty:***

Page No.....

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