

## Project Design Phase-

### Solution Architecture

Date	9October2023
TeamID	NM2023TMID09503
ProjectName	Biometric Security System For Voting Platform
MaximumMarks	4Marks

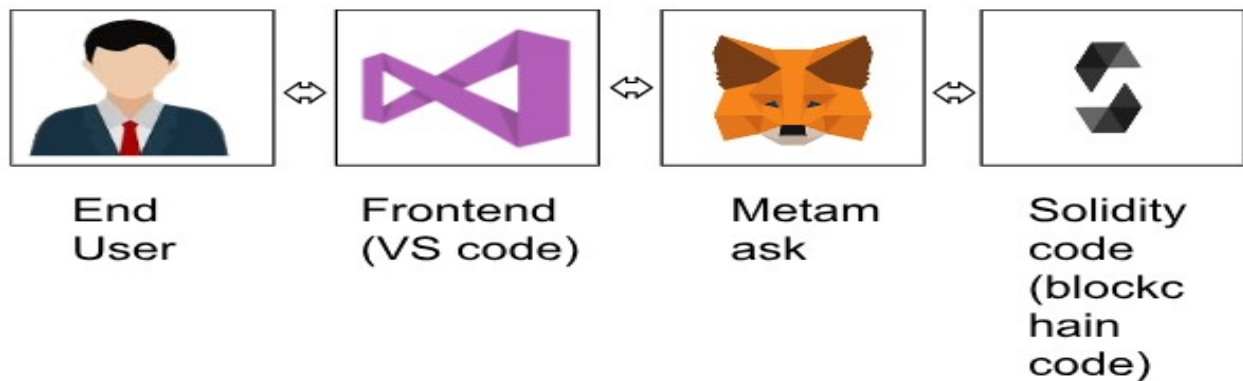
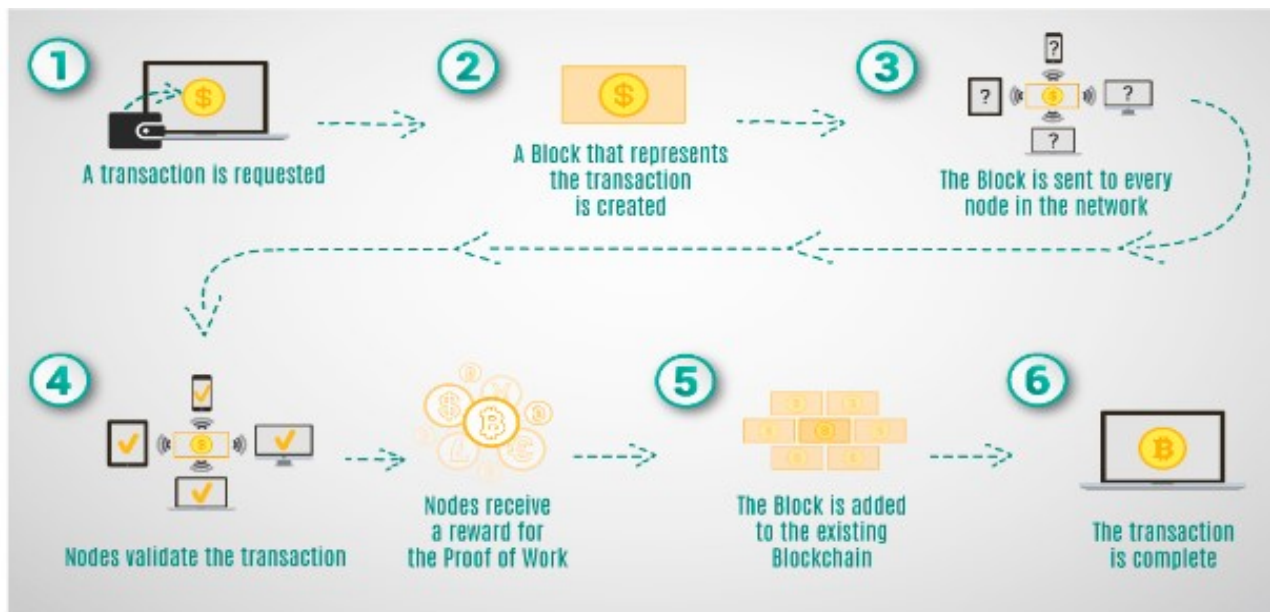
### Solution Architecture:

A biometric security system for a voting platform is a cutting-edge solution that leverages unique physiological or behavioral characteristics, such as fingerprints, irises, or facial features, to authenticate voters and safeguard the integrity of the electoral process. During registration, individuals' biometric data is securely stored, creating a binding link between their identity and their biometric template. On election day, voters undergo biometric authentication, ensuring that only eligible individuals cast their ballots. Privacy, data security, and accessibility considerations are paramount, along with the need for fallback mechanisms in case of authentication failures. This system not only enhances election security but also bolsters public trust and transparency, ushering in a new era of secure and reliable voting procedures.

#### Blockchain Biometric System

Integrating blockchain technology into biometric systems enhances security and privacy by storing biometric data in a tamper-proof and decentralized ledger. Blockchain's immutable records ensure transparent audit trails of data access and authentication events, reducing the risk of data breaches and enhancing accountability. Users can exercise greater control over their biometric data through smart contracts, specifying who can access it and under what conditions. This combination of biometrics and blockchain not only strengthens identity verification but also fosters trust in secure and reliable authentication processes.

## Example - Solution Architecture Diagram:



## Tech Stack:

- REMIX IDE
- VS CODE
- NODE JS
- METAMASK

## **Steps to complete the project :**

### **Step 1:**

Open the Zip file and download the zip file. Extract all zip files.

### **Step 2 :**

1. Open vs code in the left top select open folder. Select extracted file and open .
2. Select the projectname.sol file and copy the code.
3. Open the remix ide platform and create a new file by giving the name of projectname.sol and paste the code which you copied from vs code.
4. Click on solidity compiler and click compile the projectname.sol
5. Deploy the smart contract by clicking on the deploy and run transaction.
6. select injected provider - MetaMask. In environment
7. Click on deploy. Automatically MetaMask will open and give confirmation. You will get a pop up click on ok.
8. In the Deployed contract you can see one address copy the address.
9. Open vs code and search for the connector.js. In contract.js you can paste the address at the bottom of the code. In export const address.
10. Save the code.

### **Step 3:**

Open file explorer

1. Open the extracted file and click on the folder.
2. Open src, and search for utiles.
- 3 . You can see the frontend files. Select all the things at the top in the search bar by clicking alt+A. Search for cmd
4. Open cmd enter commands npm install npm bootstrap npm start
5. It will install all the packages and after completing it will open {LOCALHOST IP ADDRESS} copy the address and open it to chrome so you can see the frontend of your project.