Project Design Phase-I

Solution Architecture

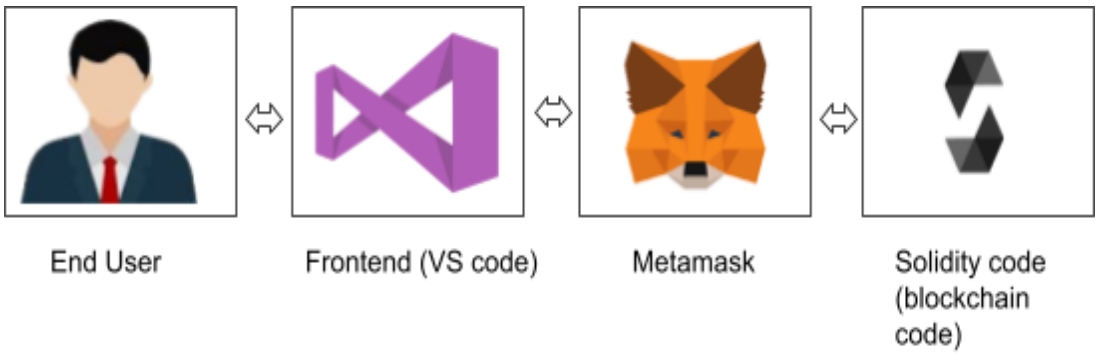
Date	25 NOVEMBER 2023
Team ID	NM2023TMID11945
Project Name	ELECTRONIC VOTING SYSTEM
Maximum Marks	4 Marks

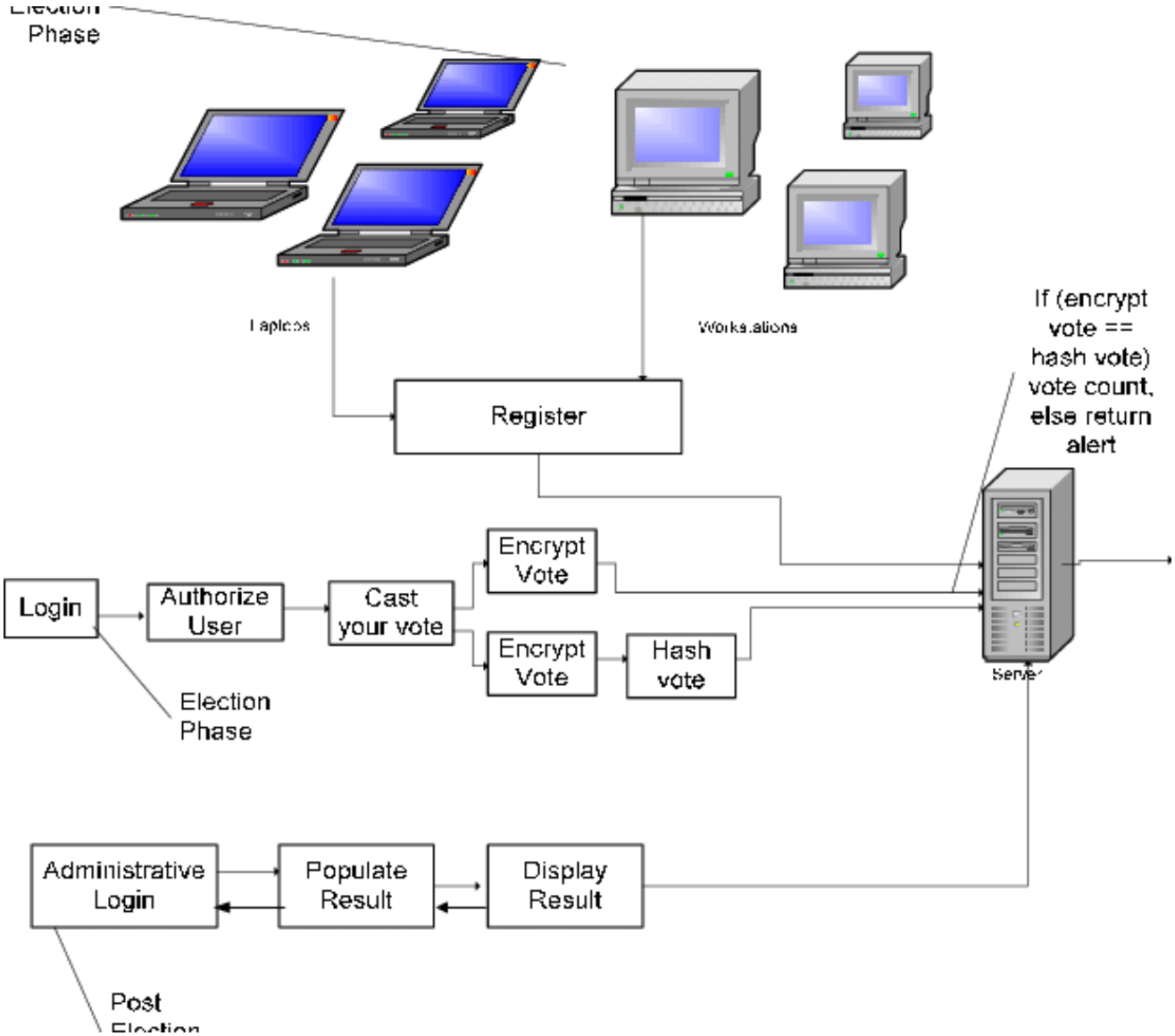
Solution Architecture:

- **Electronic voting** (also known as **e-voting**) is voting that uses electronic means to either aid or take care of casting and counting ballots.

- Depending on the particular implementation, e-voting may use standalone *electronic voting machines* (also called EVM) or computers connected to the Internet (**online voting**).
- It may encompass a range of Internet services, from basic transmission of tabulated results to full-function online voting through common connectable household devices.
- The degree of automation may be limited to marking a paper ballot, or may be a comprehensive system of vote input, vote recording, data encryption and transmission to servers, and consolidation and tabulation of election results.
- A worthy e-voting system must perform most of these tasks while complying with a set of standards established by regulatory bodies, and must also be capable to deal successfully with strong requirements associated with security, accuracy, integrity, swiftness, privacy, auditability, accessibility, cost-effectiveness, scalability and ecological sustainability.

Solution Architecture Diagram:





Prerequisite

1 download node.js : Node.js

2 download vs code: **Li4nk**

3 download metamask : https://metamask.io/

Steps to complete the project

Step 1:-

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.