Project Design Phase-I

Solution Architecture

Date	09 October 2023
TeamID	NM2023TMID09491
Project Name	Electronic Voting System
Maximum Marks	4 Marks

Solution Architecture:

1. **Blockchain Infrastructure:**
   - Select a suitable blockchain platform (e.g., Ethereum or a purpose-built blockchain).
   - Implement a decentralized network for transparency and immutability.
2. **Identity Verification:**
   - Employ secure and private identity verification methods (e.g., cryptographic keys) for eligible voters.
   - Store verified identities off-chain, ensuring privacy.
3. **Smart Contracts:**
   - Utilize smart contracts for voter registration, vote casting, and tallying.
   - Deploy tamper-resistant, auditable code for these contracts.
4. **Voter Registration:**
   - Implement a secure and private voter registration process.
   - Generate unique cryptographic identifiers for each eligible voter.

5. **Decentralized Validation:**
Engage trusted validators to verify and validate votes while maintaining voter anonymity.

6. **Security Measures:**
Implement strong encryption and access controls to protect the blockchain network from cyber threats.

7. **Transparency & Auditability:**
Leverage blockchain's inherent transparency for independent verification of the vote tally.

8. **Scalability Solutions:**
Address scalability challenges through sharding, layer 2 solutions, or a hybrid blockchain setup to accommodate a large number of voters.

9. **User-Friendly Interfaces:**
Develop intuitive web and mobile applications for voters, candidates, and administrators.

10. **Legal & Regulatory Compliance:**
Ensure adherence to local election laws and regulations to guarantee the system's legality.

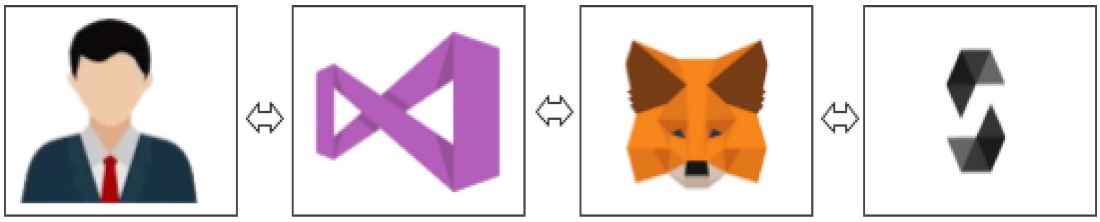
11. **Privacy Protection:**
Implement privacy-enhancing technologies to shield individual votes while maintaining the blockchain's transparency.

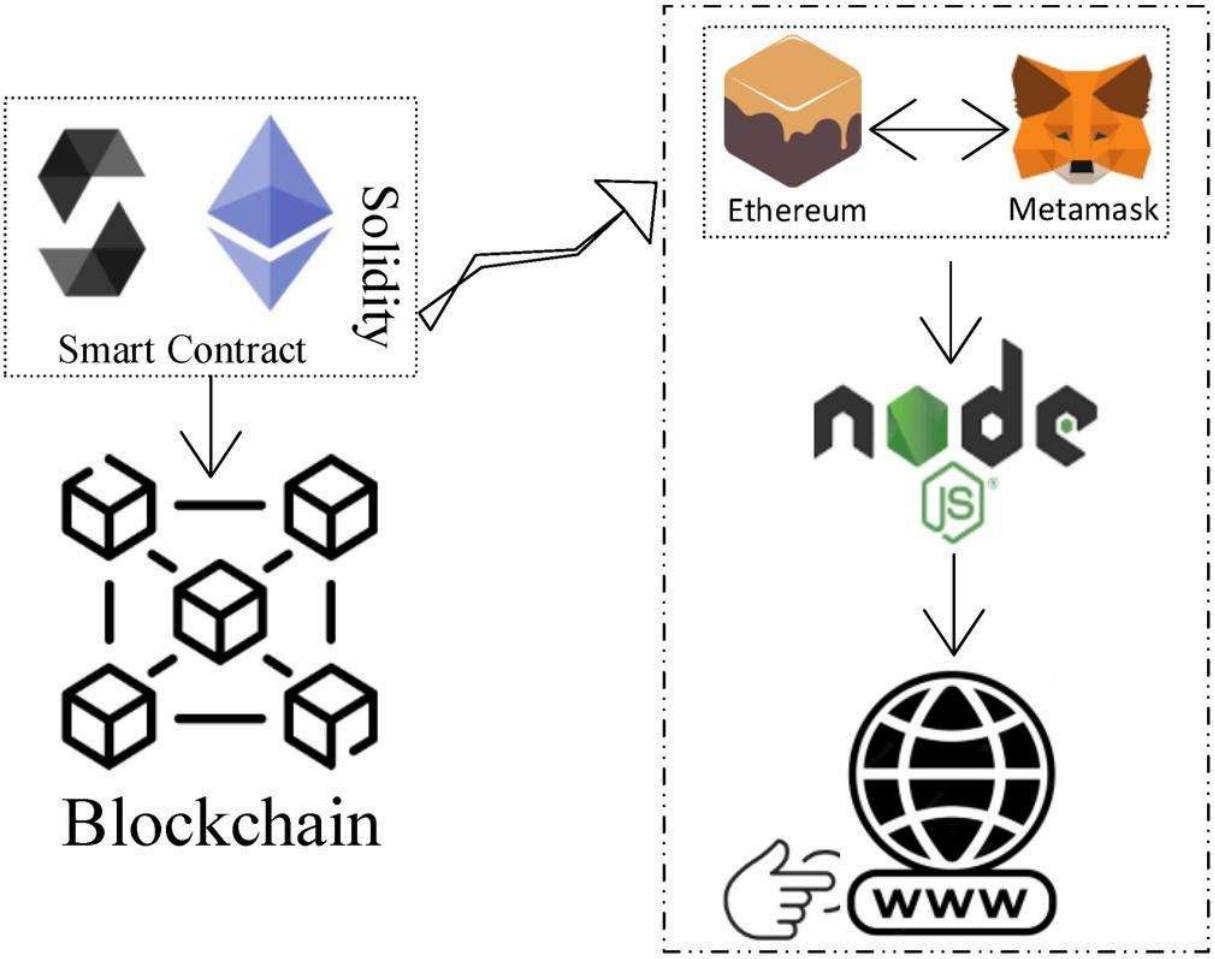
12. **Results & Tallying:**
Deploy smart contracts for automatic, tamper-resistant vote tallying.
Ensure that results are immediately visible and verifiable.

13. **Backup & Recovery:**
Set up robust backup and recovery mechanisms to safeguard against data loss or system failures.

14. **Voter Education:**
Provide comprehensive voter education and training on how to use the electronic voting system.

15. **Contingency Plans:**
Develop contingency plans for handling system failures, cyberattacks, and other unforeseen disruptions.


End User
Frontend (VS code)
Metamask
Solidity code (blockchain code)


Solidity
Smart Contract
Blockchain
Ethereum
Metamask
node
JS
WWW

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.