**GURU GOBIND SINGH COLLEGE OF ENGINEERING & RESEARCH CENTRE, NASHIK**

**MINI PROJECT REPORT**

**Academic year: 2024-25**

**“E-Voting System**

**Bachelor of Engineering (Computer Engineering)**

**Course**: Laboratory Practice-III (BT)

**Course code: 410246**

**By**

**Name(s): Aman H. Bagga**

Under the Guidance of

**Mr. S.G. Shukla**

**Guru Gobind Singh College of Engineering & Research Centre, Nashik**

**Mini-Project Report**

**Name of Programme**: Computer Engineering **Academic Year:** 2024-25

**Semester:** BECO-Sem 1 C**ourse code:** 410246

**Name of Course**: Laboratory Practice-III (BT)

-------------------------------------------------------------------------------------------------------------------------------

**Title of Mini-Project:** “E-Voting System

**1.0 Rationale:**

In this mini project we have developed a smart contract for e-voting system on Ethereum blockchain. Voting system are the most important and valuable system for our country. A blockchain-based smart contract for an e-voting system offers a secure, transparent, and decentralized solution to traditional voting. This project will explore the potential of blockchain technology to revolutionize the way elections are conducted, providing a more secure, efficient, and democratic voting experience.

**2.0 Aim /Benefits of Mini-Project:**

The mini project on an E-voting system aims aims to enhance voter confidence, reduce fraud, and promote democratic participation. E-voting systems leveraging blockchain technology offer numerous benefits, including enhanced security, transparency, immutability, decentralization, verifiability, accessibility, efficiency, cost-effectiveness, increased voter engagement, and auditability. These advantages make blockchain-based e-voting systems a promising solution for improving the democratic process and ensuring fair and reliable elections.

**3.0 Course Outcomes achieved (COs):**

1. Illustrate the Ethereum public block chain platform (CO5)
2. Identify relative application where block chain technology can be effectively used and implemented (CO6)
3. Apply and evaluate classification and clustering techniques (CO3)

**4.0 Literature Review:** -

Traditional e-voting systems have evolved significantly since their early attempts, addressing security concerns, usability challenges, and voter confidence. While centralized systems have been used, decentralized approaches leveraging blockchain technology offer enhanced security and transparency. Cryptographic techniques, authentication, and user-friendly interfaces are essential for robust e-voting. Hybrid models combining traditional and electronic voting are also explored. Future trends include further advancements in blockchain, emerging technologies, and regulatory frameworks to ensure secure, reliable, and accessible e-voting systems.

**5.0 Actual Methodology followed:**

The central authority has to assign the name of all the voters and deploy the contract. Once deployed the user has to select his has and with the help of index ID allocated, he/she has to vote. All the process done by the user is dine using the special index id allocated to each and individual voter.

**1.Algorithm:**

1. Start
2. Create 2 structure, first Voter to store if the person has voted or not and the address of voter and second candidate to store name of the voter and the vote, he/she has given.
3. Use constructor to store all the names of the candidate into the candidate structure.
4. Use function vote to check if the voter has voted or not
5. If voted show message already vote else accept vote of the voter.
6. Use function getCandidateVote to get index of voter and show if the voter with that index has voted or not
7. Use getCandidateName to get index and display the name of the voter at that index
8. Stop

**6.0 Actual Code of Program:**

//SPDX-License-Identifier: MIT

//https://betterprogramming.pub/developing-a-smart-contract-by-using-re mix-ide-81ff6f44ba2f

pragma solidity >=0.7.0 <0.9.0;

contract EVoting

{

    struct Voter

    {

        bool hasVoted;

        address voterAddress;

    }

    struct Candidate

    {

        string name;

        uint votes;

    }

    Candidate[] public candidates;

    mapping(address => Voter) public voters;

    constructor(string[] memory \_candidateNames)

    {

        for (uint i = 0; i < \_candidateNames.length; i++)

        {

            candidates.push(Candidate(\_candidateNames[i], 0));

        }

    }

    function vote(uint \_candidateIndex) public

    {

        require(!voters[msg.sender].hasVoted, "Already voted");

        require(\_candidateIndex < candidates.length, "Invalid candidate index");

        voters[msg.sender].hasVoted = true;

        voters[msg.sender].voterAddress = msg.sender;

        candidates[\_candidateIndex].votes++;

    }

    function getCandidateVotes(uint \_candidateIndex) public view returns (uint)

    {

        return candidates[\_candidateIndex].votes;

    }

    function getCandidateName(uint \_candidateIndex) public view returns (string memory)

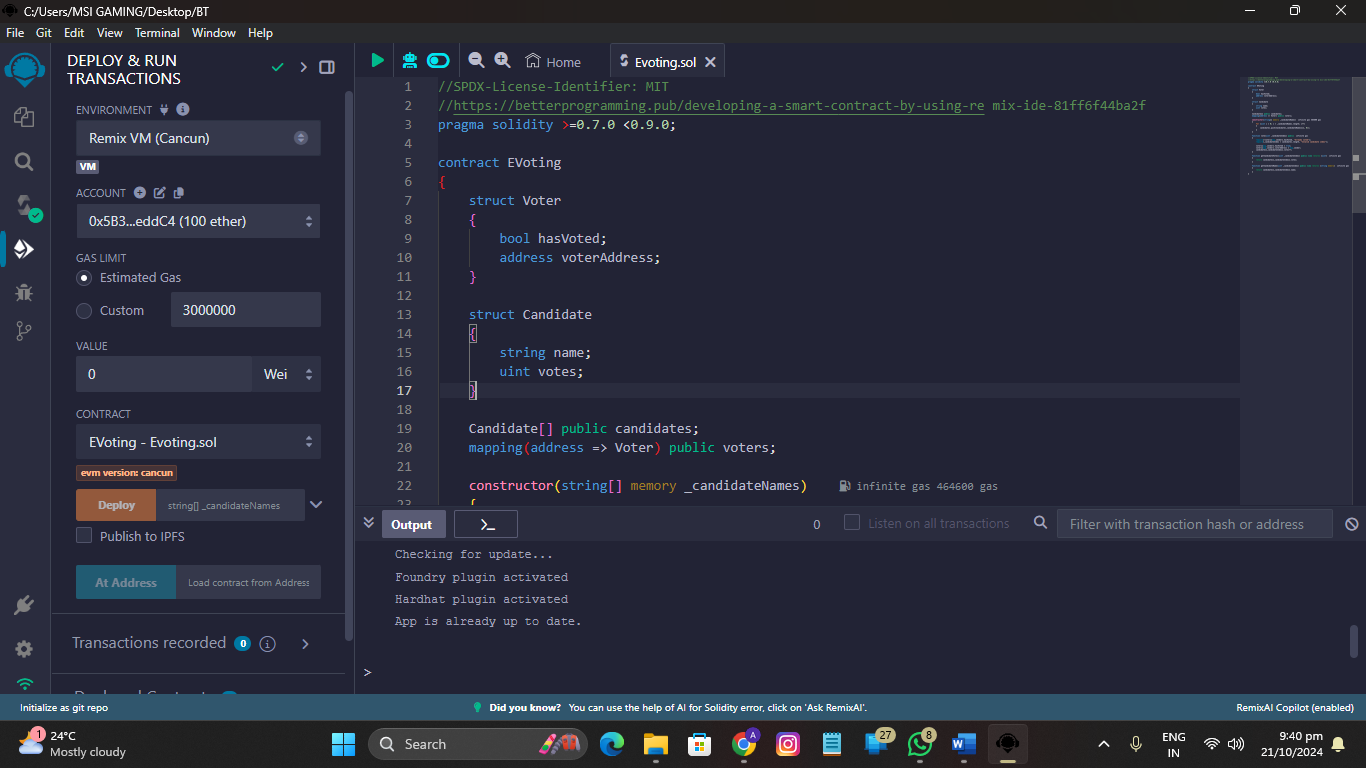
    {

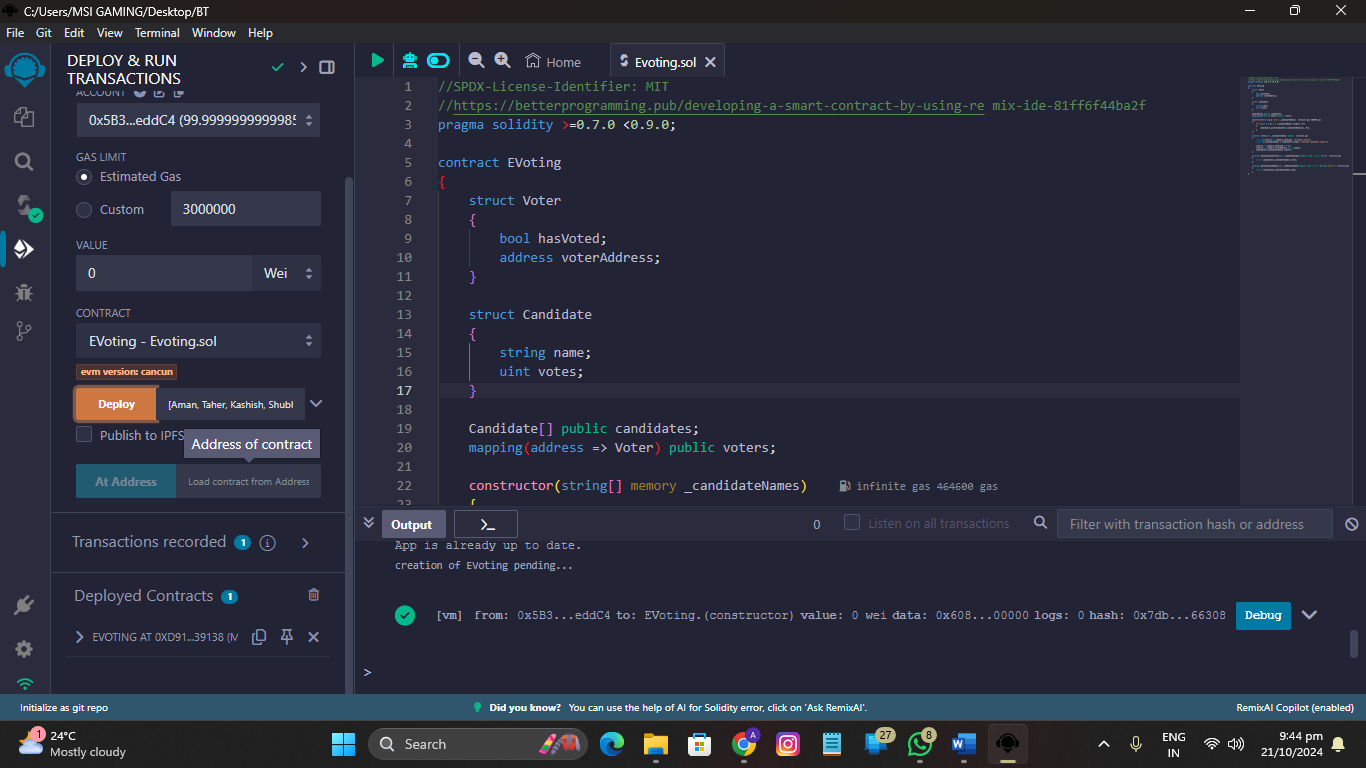
        return candidates[\_candidateIndex].name;

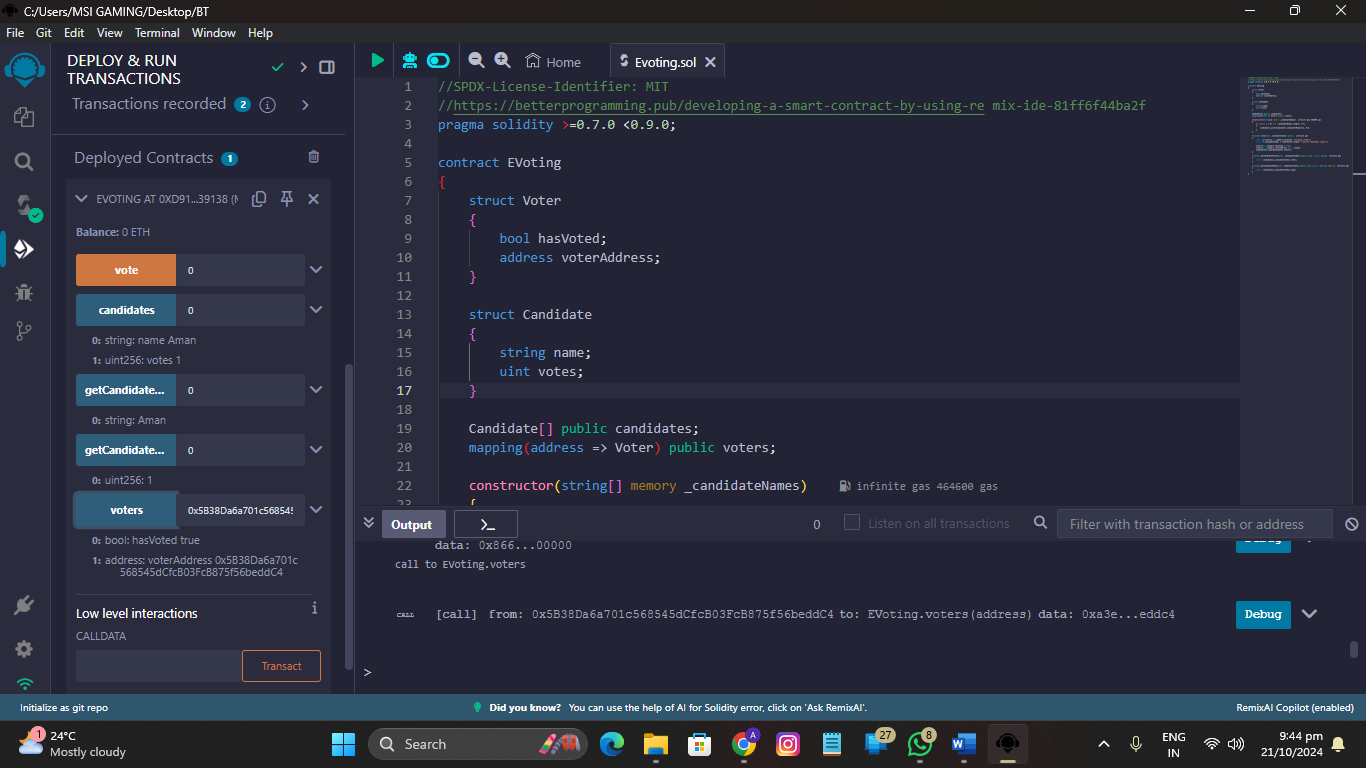
    }

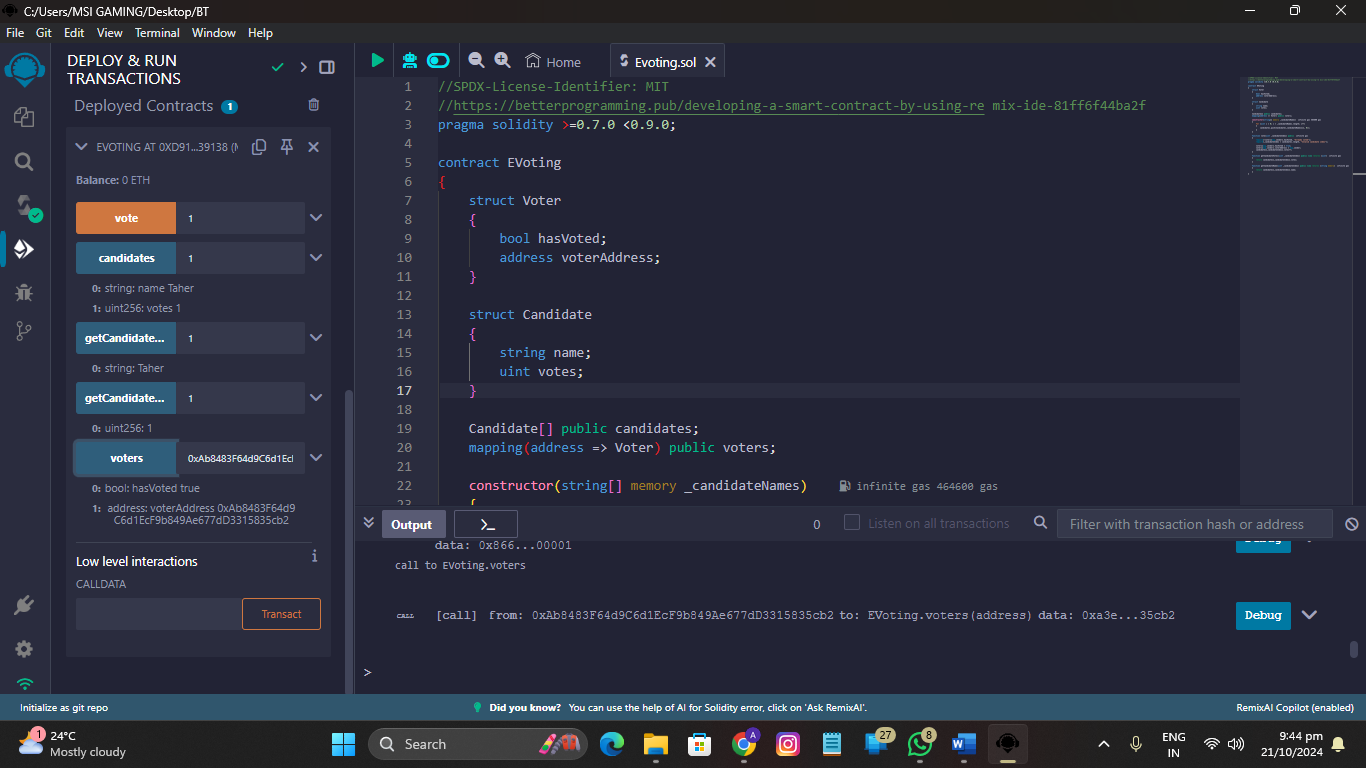
}

**7.0 Output**









**8.0 Actual Resources used:**

| **S. No.** | **Name of Resource/material** | **Specifications** | **Qty** | **Remarks** |
| --- | --- | --- | --- | --- |
| 1 | Computer System | Windows OS,i3 processor ,2GB RAM | 01 | - |
| 2 | Software | Remix IDE | 01 | - |

**9.0 Skill Developed / Learning outcome from this Mini-Project:**

1. Understanding the concept of smart contract

2. How Ethereum blockchain works

**10.0 Applications of Mini Project:**

1. To enhance the security, transparency, and efficiency of national elections, reducing the risk of fraud and increasing voter confidence.

**Evaluated by: Mr. S.G. Shukla**

**Date: Name & Signature of Guide**