BLOCKCHAIN TECHNOLOGY LAB

(20CP406P)

LAB ASSIGNMENT - 9

B.Tech in Computer Science and Engineering Dept.,

Pandit Deendayal Energy University,

Gandhinagar

Name: Vrushank Ariwala

Roll No.: 19BCP141

Branch: CSE

❖ Aim:

Create your own Application of Blockchain Technology

❖ Implementation:

A blockchain-based Full stack Exit poll system for Indian elections. More specifically only registeredvoters can come and vote. For this project, we have four phases

Registration Phase: Phase is where users can register with their voter id card number and the admin can add parties to the blockchain.

Voting Phase: The phase where voters can actually vote for their favorite party.

Result Phase: Here the result is declared by the blockchain on the basis of which party has received the maximum number of votes.

Reset Phase: All the blockchain data is reset phase to conduct a new election in the future.

All the phase changes can only be done by the admin and other random voters can't do it.

Tech Stacks Used:

- React JS
- Redux Toolkit
- Node JS
- Solidity
- Ethers.js
- Express JS
- MongoDB

Video for knowing how the application works: https://www.youtube.com/watch?v=IFph8HVXUKg

Smart Contract Of the Application:

```
// SPDX-License-Identifier: MIT pragma
solidity >=0.4.22 <0.9.0;

contract Election { address
    public admin;

    enum PHASE {
        registration,
        voting, done,
        reset
    }

    PHASE public ElectionPhase;

    struct Candidate { string
        candidate_id;string
        partyName; string
        partyImage; uint256
        partyVotes;
    }

    mapping(string => Candidate) public candidates;string[]
    public   candidate_ids_list; Candidate[] candidatesList;
    Candidate[] resultCandidatesList;

    struct Voter { string
        voter_id;
        string votedCandidate_id;bool
        is_registerd;
        bool hasVoted;
    }

    mapping(string => Voter) public voters;string[]
    public voter_ids_list;
    Voter[] votersList;

    uint256 public candidatesCount;uint256
    public votersCount;
```

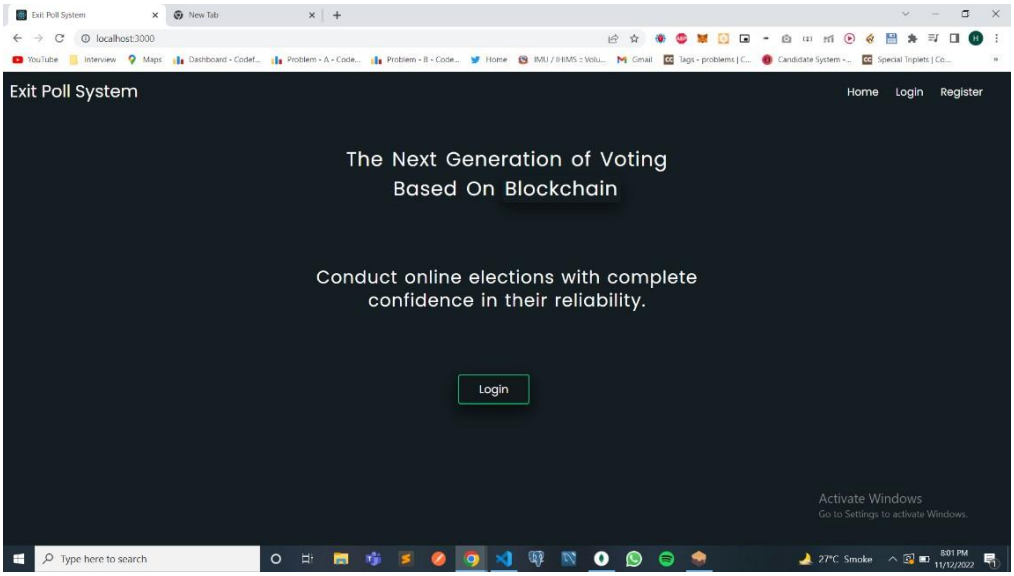
```
constructor() {
        admin = msg.sender;
        ElectionPhase = PHASE.registration;
}

function changeState(PHASE x) public {
        ElectionPhase = x;
}

function addCandidate( string
        memory _id, string
        memory _party,string
        memory _image
) public {
        require(
                ElectionPhase == PHASE.registration,
                "Registration phase is over"
        );
        candidates[_id] = Candidate(_id, _party, _image, 0);
        candidate_ids_list.push(_id); candidatesList.push(candidates[_id]);
        candidatesCount++;
}

function voterRegisteration(string memory _voter_id) public {require(
                ElectionPhase == PHASE.registration,
                "Registration phase is over"
        );
        require(
                voters[_voter_id].is_registerd == false,"You
                Already Registered"
        );
        voters[_voter_id] = Voter(_voter_id, "", true, false);
        voter_ids_list.push(_voter_id);
        votersCount++;
}

function vote(string memory _id, string memory _votedCandidate_id) public {require(
                ElectionPhase == PHASE.voting,
                "Currently Election is not started or Election is over"
        );
```

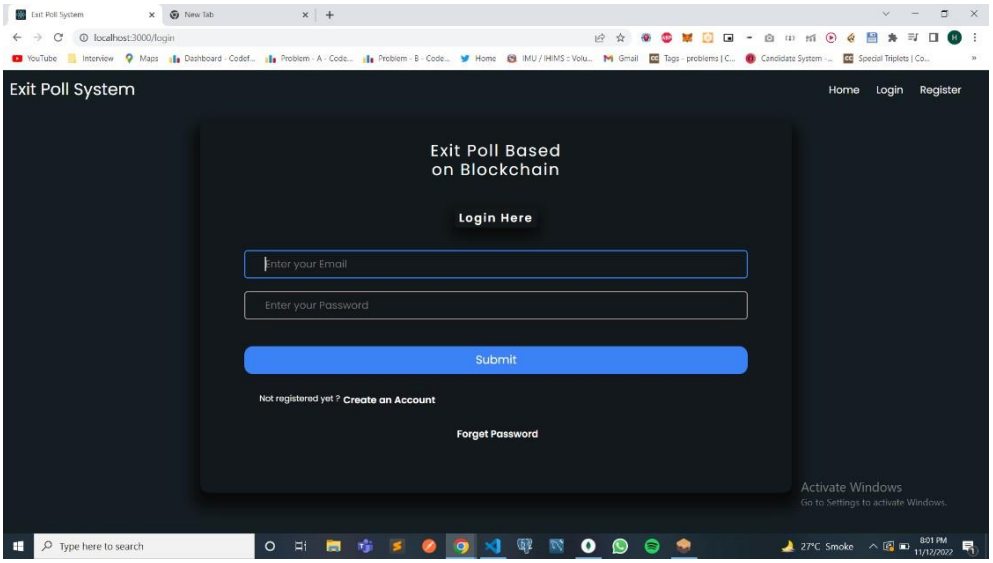
```
        require(voters[_id].is_registerd == true, "Register first"); require(voters[_id].hasVoted == false,
        "You have already voted");voters[_id].hasVoted = true;
        votersList.push(voters[_id]); voters[_id].votedCandidate_id =
        _votedCandidate_id;
        candidates[_votedCandidate_id].partyVotes++;
    }

    function getWinner() public view returns (string memory) { require(ElectionPhase ==
        PHASE.done, "Voting is not completed yet");uint256 maxVotes =
        candidates[candidate_ids_list[0]].partyVotes; string memory winner_id =
        candidates[candidate_ids_list[0]]
                .candidate_id;
        for (uint256 i = 1; i < candidate_ids_list.length; i++) {
                if (maxVotes < candidates[candidate_ids_list[i]].partyVotes) { maxVotes =
                        candidates[candidate_ids_list[i]].partyVotes; winner_id =
                        candidates[candidate_ids_list[i]].candidate_id;
                }
        }
        return winner_id;
    }

    function addCandidateToResultList() public {
        require(ElectionPhase == PHASE.done, "Result Phase Is Not Started");for (uint256 i = 0; i
        < candidate_ids_list.length; i++) {
                resultCandidatesList.push(candidates[candidate_ids_list[i]]);
        }
    }

    function getUpdatedCandidateList()public
        view
        returns (Candidate[] memory)
    {
        require(ElectionPhase == PHASE.done, "Result Phase Is Not Started");return
        resultCandidatesList;
    }

    function allCandidates() public view returns (Candidate[] memory) {return
        candidatesList;
    }

    function getCandidateIdList() public view returns (string[] memory) {return
        candidate_ids_list;
    }
```

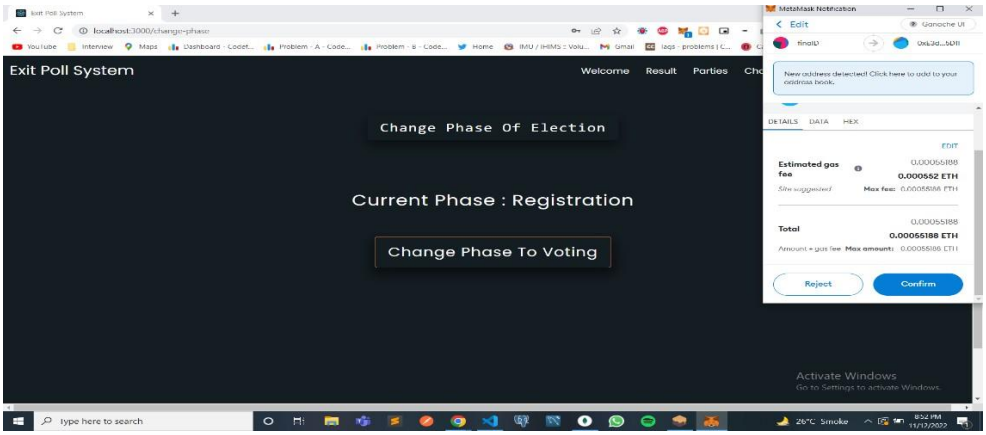
```
    function getVoterIdList() public view returns (string[] memory) {return
        voter_ids_list;
    }

    function allVoters() public view returns (Voter[] memory) {return votersList;
    }

    function resetElection() public {
        require(ElectionPhase == PHASE.reset, "Election is not Over yet");

        votersCount = 0;
        candidatesCount = 0; delete
        candidate_ids_list;delete
        candidatesList;
        delete resultCandidatesList;

        for (uint256 i = 0; i < voter_ids_list.length; i++) {
            voters[voter_ids_list[i]].is_registerd = false;
        }

        delete voter_ids_list;delete
        votersList;
    }
}
```

The Complete Application Code Can be Found at: https://github.com/har200105/ExitPollSystem

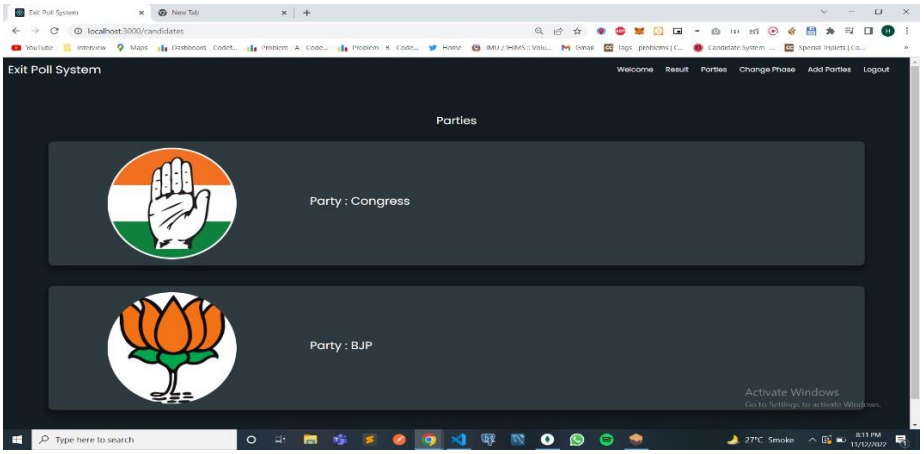
Screenshots:



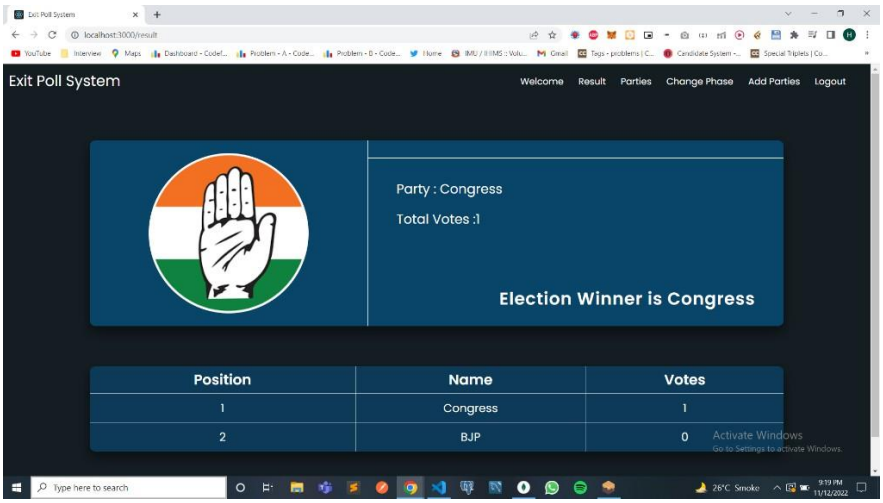





Exit Poll System
Welcome
Result
Parties
Candidates
Party :Congress
Vote
Party :BJP
Vote
MetaMask Notification
Edit
New address detected! Click here to add to your address book.
http://localhost:3000
DETAILS
DATA
HEX
Estimated gas fee
0.00297548
0.002975 ETH
Total
0.00297548
0.00297548 ETH
Activate Windows
Go to Settings to activate Windows.
Type here to search
26°C Smoke
8:52 PM
11/12/2022


Exit Poll System
Welcome
Result
Parties
Change Phase
Add Parties
Logout
Parties
Party : Congress
Party : BJP
Activate Windows
Go to Settings to activate Windows.
Type here to search
27°C Smoke
8:13 PM
11/12/2022


Exit Poll System
Welcome
Result
Parties
Voting
Voter Registration
Logout
Candidates
Party :Congress
Vote
Party :BJP
Vote
Activate Windows
Go to Settings to activate Windows.
Type here to search
26°C Smoke
9:18 PM
11/12/2022


Exit Poll System
Welcome
Result
Parties
Change Phase
Add Parties
Logout
Party : Congress
Total Votes :1
Election Winner is Congress
Position
Name
Votes
1
Congress
1
2
BJP
0
Activate Windows
Type here to search