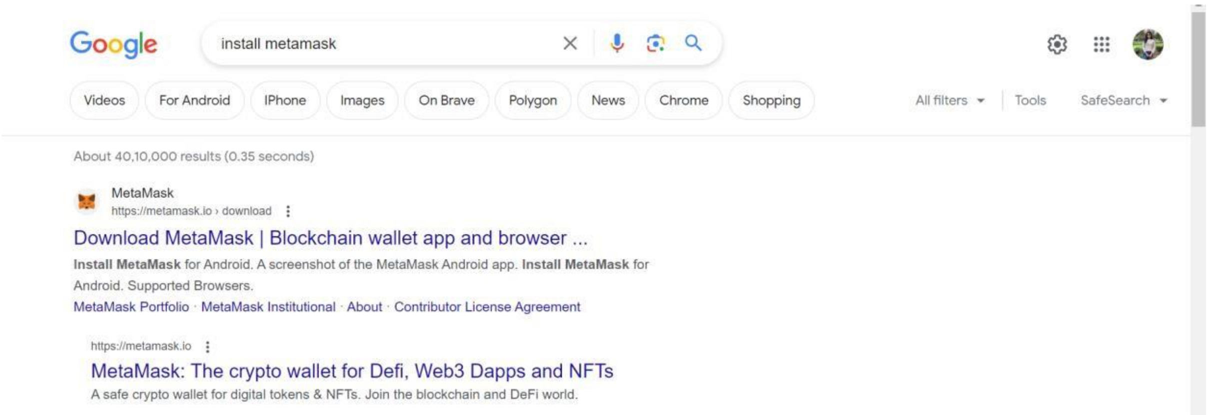
1.Assignment-1

Installation of MetaMask and study spending Ether per transaction.



About 40,10,000 results (0.35 seconds)

MetaMask
https://metamask.io › download

Download MetaMask | Blockchain wallet app and browser ...

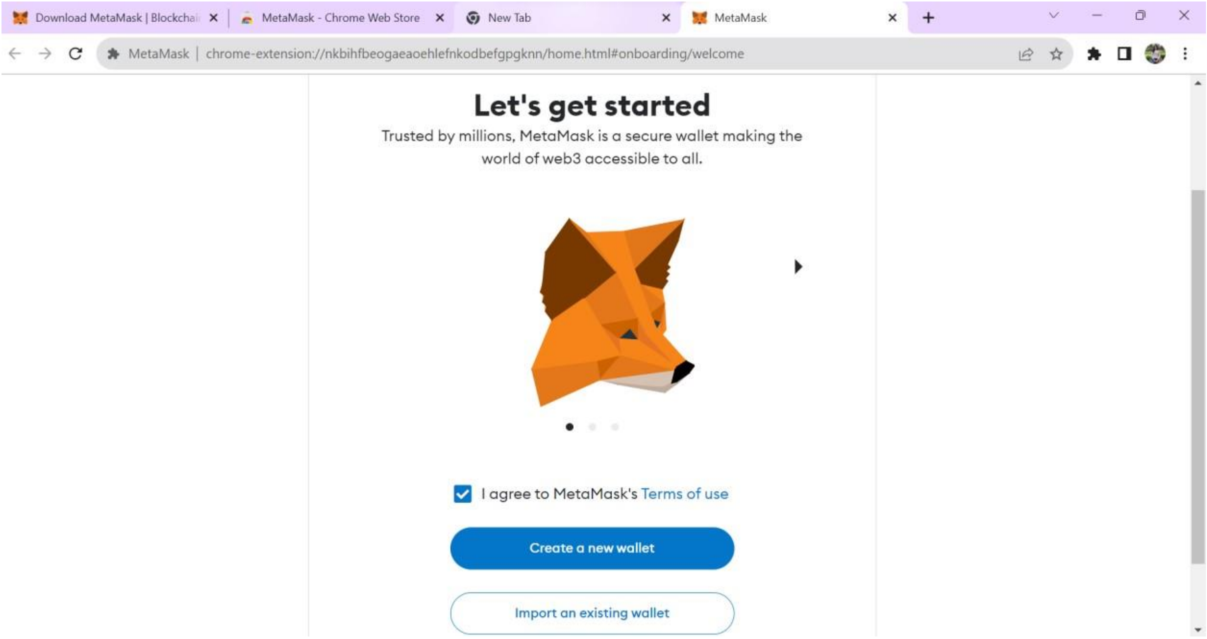
Install MetaMask for Android. A screenshot of the MetaMask Android app. **Install MetaMask** for Android. Supported Browsers.

MetaMask Portfolio · MetaMask Institutional · About · Contributor License Agreement

https://metamask.io

MetaMask: The crypto wallet for Defi, Web3 Dapps and NFTs

A safe crypto wallet for digital tokens & NFTs. Join the blockchain and DeFi world.



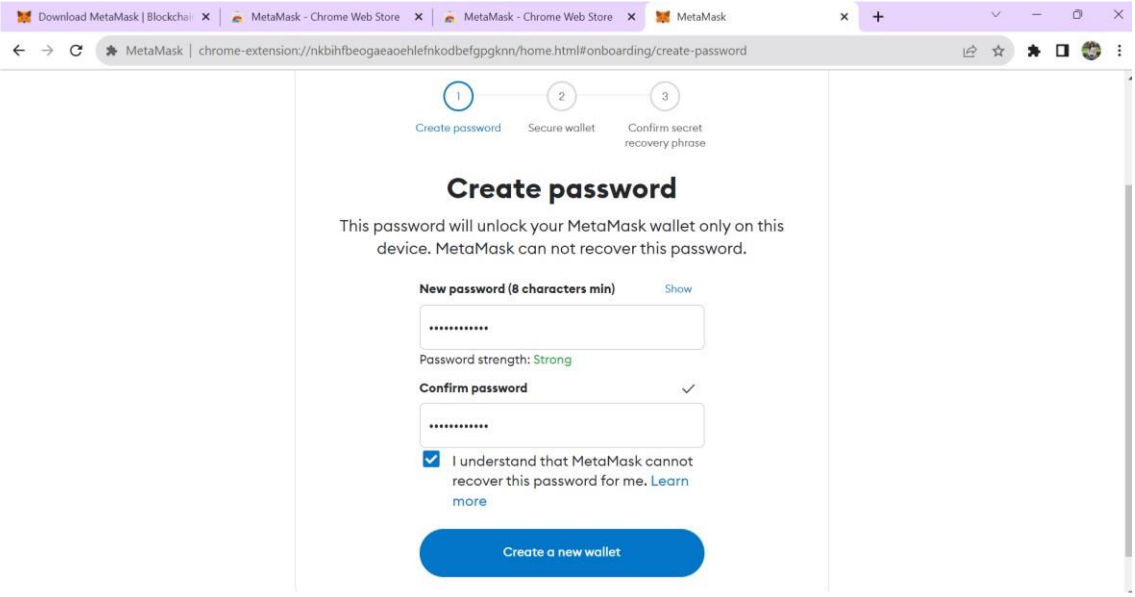
Download MetaMask | Blockchai
MetaMask - Chrome Web Store
New Tab
MetaMask
MetaMask | chrome-extension://nkbihfbeogaeaoehlefnkodbefgpgknn/home.html#onboarding/welcome

Let's get started

Trusted by millions, MetaMask is a secure wallet making the world of web3 accessible to all.

☑ I agree to MetaMask's Terms of use

Create a new wallet
Import an existing wallet


Download MetaMask | Blockchai
MetaMask - Chrome Web Store
MetaMask - Chrome Web Store
MetaMask
MetaMask | chrome-extension://nkbihfbeogaeaoehlefnkodbefgpgknn/home.html#onboarding/create-password
1
Create password
2
Secure wallet
3
Confirm secret recovery phrase

Create password

This password will unlock your MetaMask wallet only on this device. MetaMask can not recover this password.

New password (8 characters min) Show

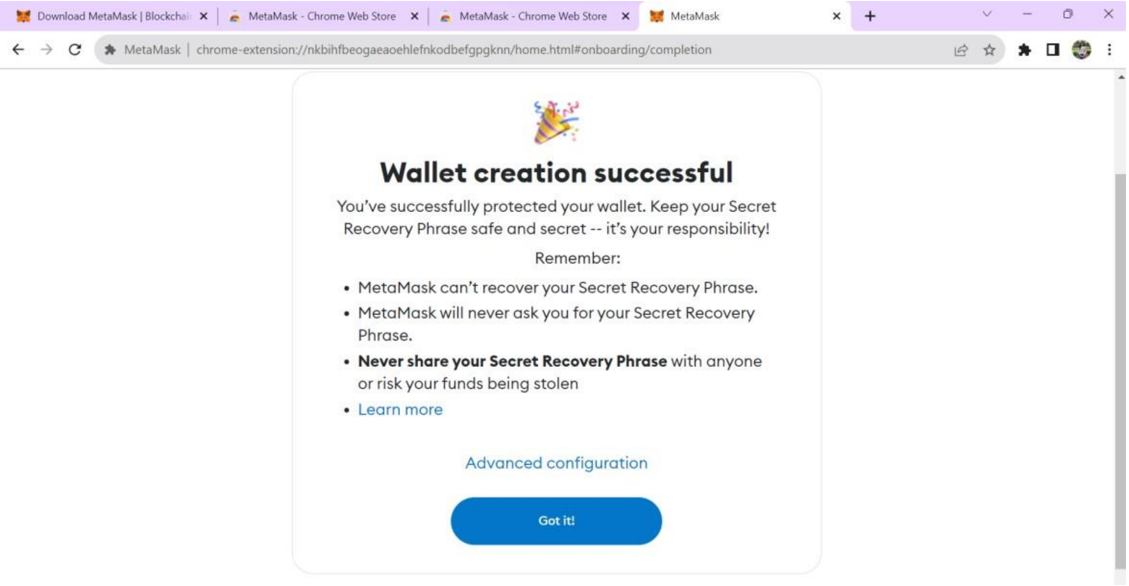
••••••••••••

Password strength: Strong

Confirm password ✓

••••••••••••

☑ I understand that MetaMask cannot recover this password for me. Learn more


Download MetaMask | Blockchai
MetaMask - Chrome Web Store
MetaMask - Chrome Web Store
MetaMask
MetaMask | chrome-extension://nkbihfbeogaeaoehlefnkodbefgpgknn/home.html#onboarding/completion

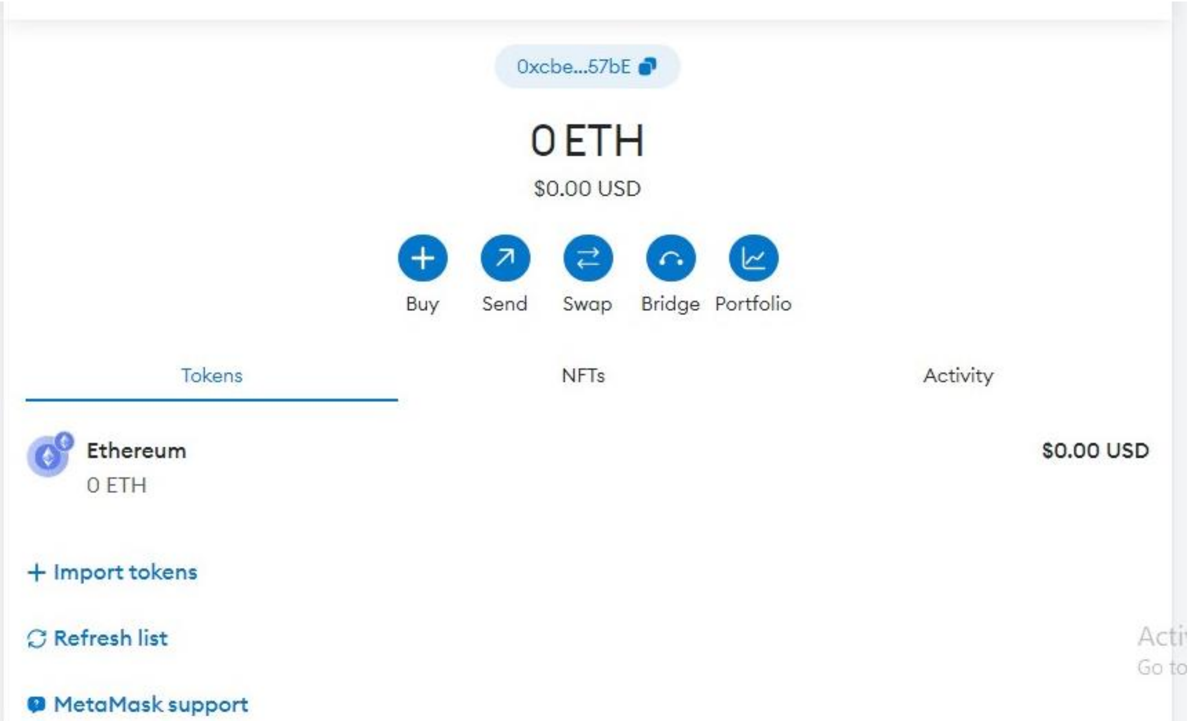
Wallet creation successful

You've successfully protected your wallet. Keep your Secret Recovery Phrase safe and secret -- it's your responsibility!

Remember:

- MetaMask can't recover your Secret Recovery Phrase.
- MetaMask will never ask you for your Secret Recovery Phrase.
- **Never share your Secret Recovery Phrase** with anyone or risk your funds being stolen
- Learn more

Advanced configuration



0xcbe...57bE

0 ETH

$0.00 USD

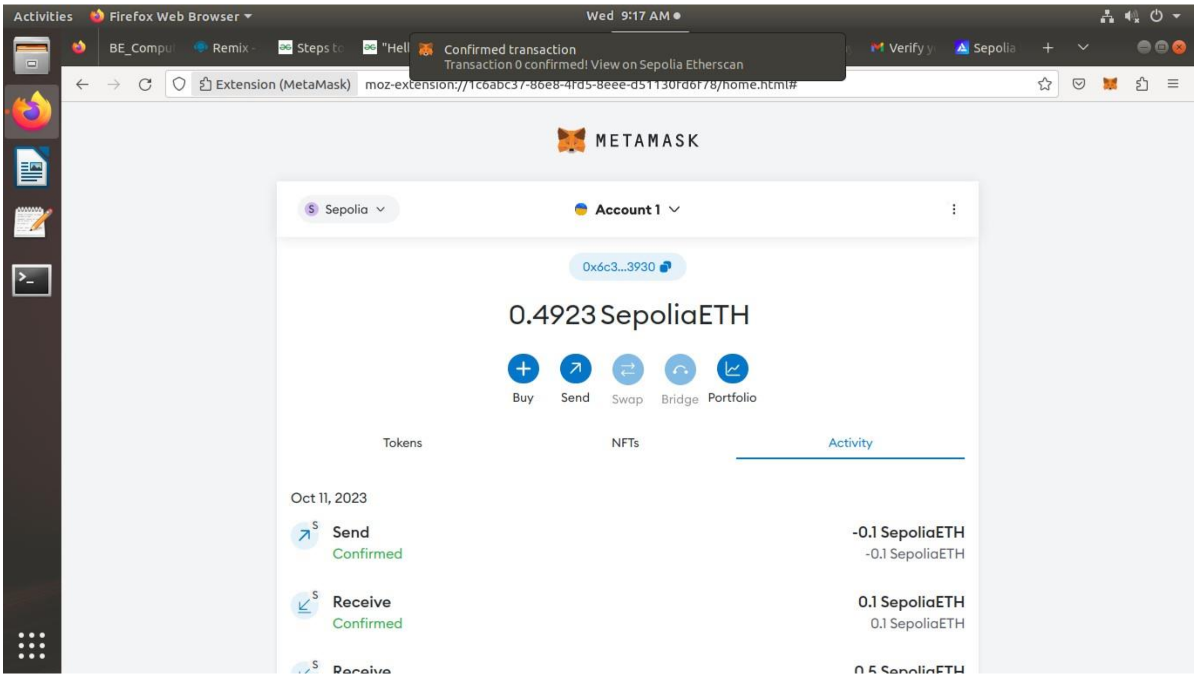
Ethereum

0 ETH

$0.00 USD

2.Assignment-2

Create your own wallet using Metamask for crypto transactions.



Confirmed transaction
Transaction 0 confirmed! View on Sepolia Etherscan

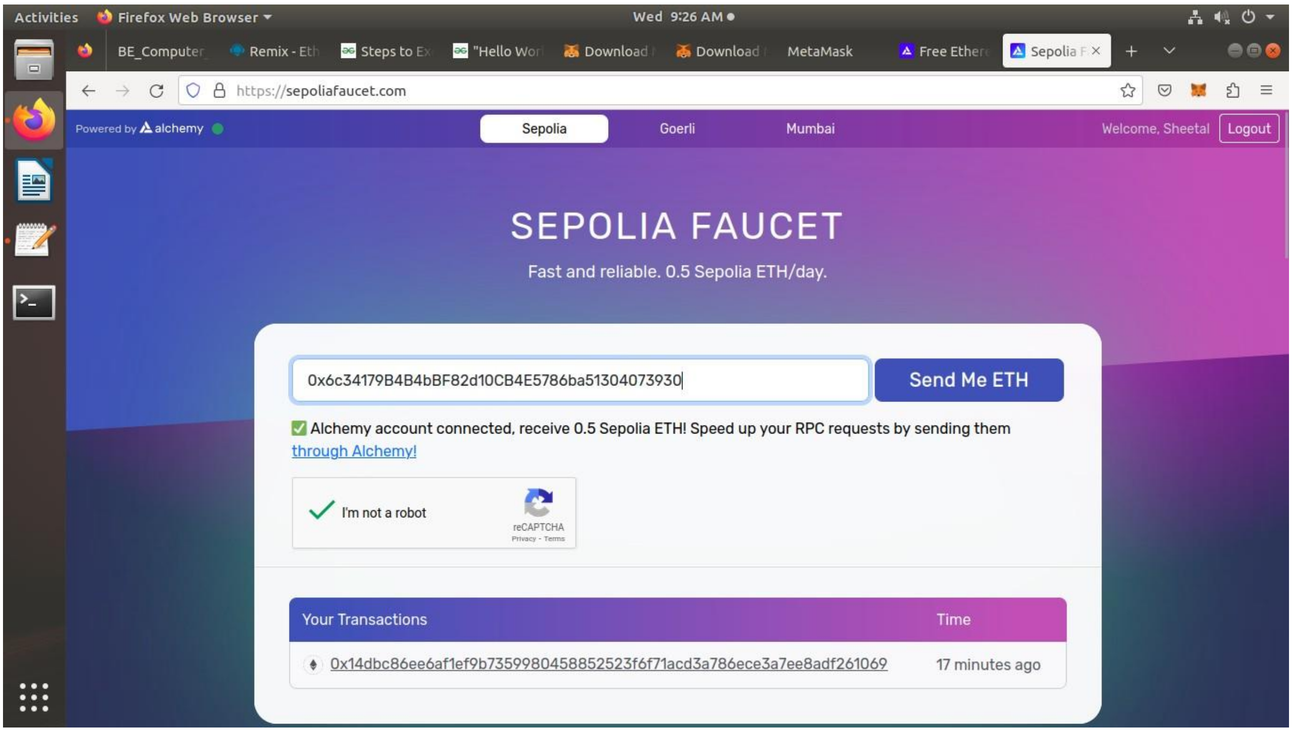
moz-extension://1c6abc37-86e8-4fd5-8eee-d51130fd6f78/home.html#

0x6c3...3930

0.4923 SepoliaETH

Oct 11, 2023

Send Confirmed	-0.1 SepoliaETH -0.1 SepoliaETH
Receive Confirmed	0.1 SepoliaETH 0.1 SepoliaETH
Receive	0.5 SepoliaETH



https://sepoliafaucet.com

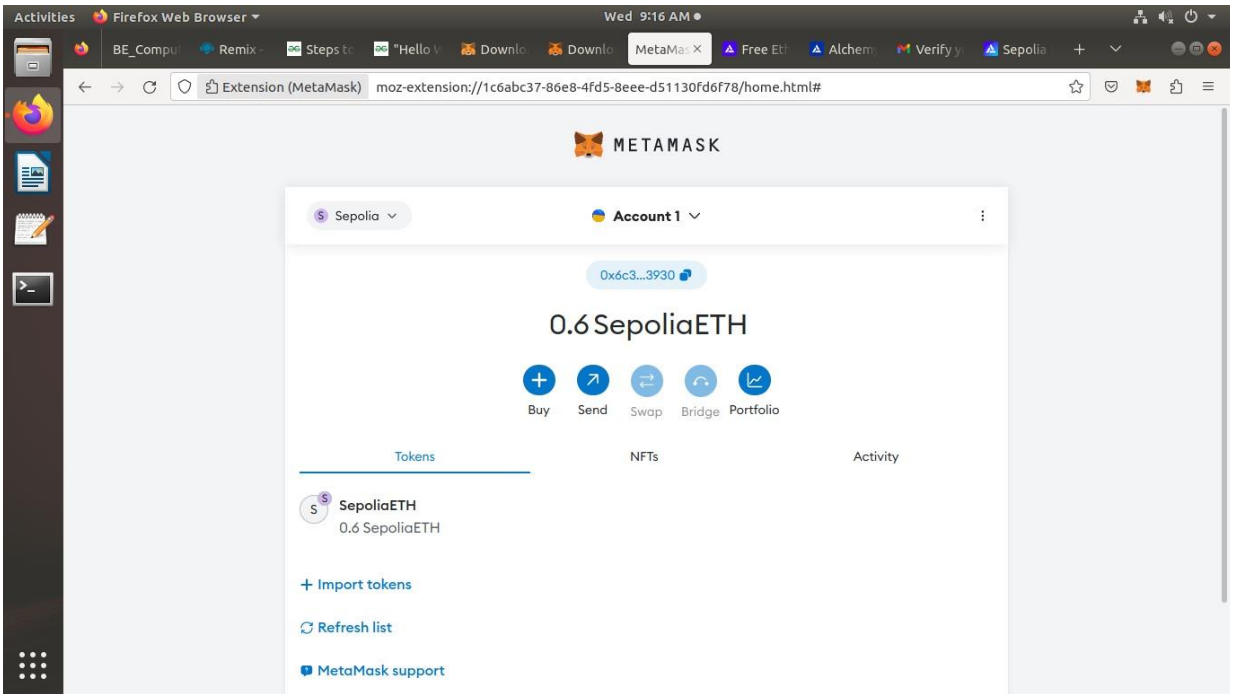
SEPOLIA FAUCET

Fast and reliable. 0.5 Sepolia ETH/day.

0x6c34179B4B4bBF82d10CB4E5786ba51304073930

✅ Alchemy account connected, receive 0.5 Sepolia ETH! Speed up your RPC requests by sending them through Alchemy!

Your Transactions	Time
0x14dbc86ee6af1ef9b7359980458852523f6f71acd3a786ece3a7ee8adf261069	17 minutes ago


Activities
Firefox Web Browser
Wed 9:16 AM
BE_Compu
Remix -
Steps to
"Hello W
Downlo
Downlo
MetaMa
Free Et
Alchem
Verify y
Sepolia
Extension (MetaMask)
moz-extension://1c6abc37-86e8-4fd5-8eee-d51130fd6f78/home.html#
METAMASK
Sepolia
Account 1
0x6c3...3930
0.6 SepoliaETH
Buy
Send
Swap
Bridge
Portfolio
Tokens
NFTs
Activity
SepoliaETH
0.6 SepoliaETH
+ Import tokens
Refresh list
MetaMask support

3.Assignment 3

```
pragma    solidity    >=    0.8.18;
contract Bank{
mapping(address => uint) public user_account; mapping(address
=> bool) public user_exist;
function create_account() public payable returns(string memory){
require(user_exist[msg.sender] == false, "Account Already created!");
user_account[msg.sender] = msg.value; user_exist[msg.sender] =
true; return "Account created";
}
function deposit(uint amount) public payable returns(string memory){
require(user_exist[msg.sender] == true, "Account not created!");
require(amount > 0, "Amount should be greater than 0");
user_account[msg.sender] += amount; return "Amount deposisted
sucessfully";
}
function withdraw(uint amount) public payable returns(string memory){
require(user_exist[msg.sender] == true, "Account not created!"); require(amount
> 0, "Amount should be greater than 0");
require(user_account[msg.sender] >= amount, "Amount is greater than money deposisted");
user_account[msg.sender] -= amount; return "Amount withdrawn sucessfully";} function
account_balance() public view returns(uint){ return user_account[msg.sender];}
function account_exists() public view returns(bool){ return
user_exist[msg.sender];
}
}
```

4.Assignment 4

```
pragma solidity >= 0.8.18; contract MarksManagmtSys
{
struct Student
{ int
ID;
string fName; string lName; int marks;
}
address owner; int public stdCount = 0;
mapping(int => Student) public stdRecords;

modifier onlyOwner
{
require(owner == msg.sender);
_;
}
constructor()
{
owner=msg.sender;
}


function addNewRecords(int _ID, string
memory _fName, string memory _lName, int
_marks) public onlyOwner
{

stdCount = stdCount + 1;
stdRecords[stdCount] = Student(_ID, _fName, _lName, _marks);
}
function bonusMarks(int _bonus) public onlyOwner
{
stdRecords[stdCount].marks =

stdRecords[stdCount].marks + _bonus;
}
}
```

SCTR's Pune Institute of Computer Technology Dhankawadi,
Pune

A MINI-PROJECT REPORT ON

Decentralized Voting System on Blockchain

SUBMITTED BY

Aayush Goyal 41302 Divya
Khandare 41321
Arnav Firke 41322

Under the guidance of
Prof. V.S. Gaikwad

DEPARTMENT OF COMPUTER ENGINEERING
ACADEMIC YEAR 2023-24

Contents

1 Title

Decentralized Voting System on Blockchain

2 Introduction

The rapid advancement of blockchain technology has created opportunities for improving various sectors, including voting systems. Traditional voting methods are often plagued by issues such as fraud, lack of transparency, and inefficiencies. This project aims to develop a decentralized voting system on the Ethereum blockchain using smart contracts, which will enhance the integrity and security of the electoral process.

The system allows for candidate registration, voter authorization, and secure voting, ensuring each voter's choice is recorded immutably. By utilizing smart contracts, the voting process is automated, enabling real-time results and minimizing the potential for electoral fraud.

This report provides an overview of the design and implementation of the voting system, emphasizing the importance of security and transparency in electoral processes.

3 Problem Statement

The primary objective of this project is to create a decentralized voting system that enhances the integrity and security of the electoral process. The system aims to ensure that only authorized voters can participate, that votes are accurately counted, and that the results are transparent and tamper-proof. Specifically, it addresses issues such as:

- Voter impersonation and identity theft.
- Vote tampering and alteration.
- Lack of transparency in the voting process.
- Delays in vote counting and result announcement.

4 Objectives and Scope

4.1 Objectives of Project:

The main objectives of the decentralized voting system are:

- To implement a secure and transparent voting mechanism using blockchain technology.
- To automate the election process, allowing for real-time counting of votes and quick determination of winners.
- To minimize the risks associated with electoral fraud by ensuring an immutable record of votes.
- To provide a user-friendly interface for voters and election administrators, improving accessibility.

4.2 Scope of Project:

- **Smart Contract Development:** Create and deploy the voting smart contract on the Ethereum blockchain, incorporating best practices for security and performance.
- **User Interface Development:** Design a user-friendly interface for voters to interact with the smart contract, ensuring ease of use and accessibility for all demographic groups.
- **Testing and Validation:** Conduct rigorous testing to ensure the system operates correctly and securely, including unit tests, integration tests, and user acceptance testing.
- **Deployment and Monitoring:** After successful testing, deploy the system on the Ethereum mainnet and monitor its performance during elections.

5 Theoretical Background

5.1 Blockchain Technology

Blockchain is a distributed ledger technology that allows multiple parties to maintain a shared database in a secure and transparent manner. Each transaction or vote is recorded in a block, and once the block is filled, it is added to a chain of previous blocks, creating a chronological and immutable record. This ensures that no single party can alter the history of transactions without consensus from the network.

Key characteristics of blockchain technology include:

- **Decentralization:** Unlike traditional systems that rely on a central authority, blockchain operates on a peer-to-peer network, distributing control among all participants.
- **Immutability:** Once data is written to the blockchain, it cannot be altered or deleted, ensuring the integrity of the records.
- **Transparency:** All participants can view the entire transaction history, enhancing trust among users.

- **Security:** Cryptographic techniques are used to secure data, making it resistant to tampering and fraud.

- **Consensus Mechanisms:** Various algorithms like Proof of Work (PoW) and Proof of Stake (PoS) ensure agreement on the blockchain's state among participants.

5.2 Smart Contracts

Smart contracts are self-executing contracts with the terms of the agreement directly written into code. They operate on blockchain networks and automatically enforce the rules and conditions stipulated within them. Key advantages include:

- **Automation:** Smart contracts automate processes, reducing the need for intermediaries and minimizing human error.

- **Trust:** Since they run on a blockchain, all parties can trust that the contract will execute as intended without manipulation.

- **Efficiency:** Transactions are processed quickly, allowing for faster results and reduced administrative overhead.

- **Auditability:** All actions performed by the contract can be audited and verified by all stakeholders.

5.3 Applications of Decentralized Voting Systems

Decentralized voting systems leverage blockchain technology to address common challenges in traditional voting methods. These systems:

- Enhance security by preventing tampering and ensuring votes are counted as cast. •

Improve accessibility, allowing remote and online voting options for eligible voters.

- Foster trust in the electoral process through transparency and auditability.

- Enable real-time result tracking, increasing voter engagement and confidence in outcomes.

6 About the Smart Contract

The smart contract facilitates the voting process with the following features:

- **Candidate Registration:** Only the election owner can add candidates, ensuring control over the electoral slate.

- **Voter Authorization:** The election owner can authorize voters through a secure mechanism, ensuring only permitted individuals can vote.

- **Voting Process:** Authorized voters can cast votes for candidates, with restrictions to prevent double voting and ensure anonymity.

- **Results Calculation:** The contract can determine the winner based on the votes received, providing immediate feedback after the voting period.

- **Audit Trail:** Each transaction is recorded on the blockchain, providing a complete history of the election process that can be audited by stakeholders.

7 Methodological Details

The development of the decentralized voting system involved the following steps:

- **Requirement Analysis:** Identify the functional and non-functional requirements for the voting system through stakeholder consultations and analysis of existing systems.
- **Smart Contract Design:** Design the data structures and functions necessary for the voting process, ensuring that security and efficiency are prioritized.
- **Implementation:** Code the smart contract in Solidity and deploy it on the Ethereum test network, allowing for thorough testing in a simulated environment.
- **Testing:** Perform extensive testing to ensure the contract is secure, all functionalities are working as intended, and the user interface is intuitive.
- **Documentation:** Create comprehensive documentation for both developers and endusers, covering system architecture, user manuals, and maintenance guides.

8 Modern Engineering Tools Used

The development of the decentralized voting system utilized several engineering tools and technologies:

1. Solidity:

The programming language used for writing smart contracts on the Ethereum blockchain, designed for security and efficiency. Solidity allows developers to create robust applications with complex logic.

2. Remix IDE:

An integrated development environment that facilitates writing, testing, and deploying smart contracts, featuring a user-friendly interface and built-in tools for debugging.

3. Ethereum Blockchain:

The decentralized platform on which the voting system operates, providing security and immutability through its consensus mechanism. The Ethereum network allows for the deployment of decentralized applications (DApps) and smart contracts.

4. Ganache:

A personal blockchain for Ethereum development that allows testing of smart contracts in a controlled environment, enabling developers to simulate blockchain behavior without incurring transaction costs.

9 Future Work and Considerations

While the current implementation demonstrates the feasibility of a decentralized voting system, future enhancements could include:

- **Multi-Election Support:** Expanding the system to handle multiple elections simultaneously, accommodating different voter demographics and election types.

- **Enhanced User Interface:** Developing a more interactive and visually appealing user interface to improve user experience and accessibility.
- **Mobile Compatibility:** Ensuring that the voting application is fully functional on mobile devices to increase accessibility for voters.
- **Integration of Additional Features:** Considering the addition of features such as blockchain-based identity verification to enhance voter authentication.
- **Regulatory Compliance:** Ensuring that the voting system adheres to local laws and regulations governing elections and data protection.

10 Conclusion

In conclusion, this project successfully demonstrates the potential of blockchain technology to create a secure and transparent voting system. By implementing a decentralized approach, the system minimizes the risks of fraud and enhances the integrity of elections.

The smart contract developed allows for efficient management of candidates and voters, ensuring the voting process is smooth and reliable. Future work may include expanding the system to support multiple elections simultaneously and enhancing the user interface for better accessibility.

This project not only contributes to the field of blockchain applications but also serves as a foundation for developing more secure electoral processes in the future.