SHREENIVASA ENGINEERING COLLEGE

ANNA UNIVERSITY : CHENNAI
OCT 2023

SUBMITTED BY

DHIVAGAR	S	622620104004
PANDIYAN	R	622620104018
RUPIKA	K	622620104021
THIRUMALAIVASAN	B	622620104031

Guided by

Mr.MAHAVISHNU

AP/CSE

PROJECT TITLE

CENTRAL BANKING SMART CONTRACT

INTRODUCTION

- A central bank smart contract is a self-executing contract with the terms of the agreement directly written into code.
- In the context of a central bank, this could refer to a smart contract governing various financial and monetary operations.
- For instance, a central bank smart contract might be designed to automate certain aspects of monetary policy, such as adjusting interest rates based on predefined economic indicators.
- It could also be used for issuing and managing digital currencies, facilitating secure and transparent transactions within the financial system.
- It's a fascinating intersection of traditional finance and cutting-edge technology

EXISTING SYSTEM

- **Digital Currency Issuance:** Central banks could create digital currencies as smart contracts, providing a more efficient and transparent way to issue and manage currency.
- **Monetary Policy Execution:** Smart contracts could automate the execution of monetary policies. For instance, adjusting interest rates based on predefined economic indicators or automatically implementing quantitative easing measures.
- **Transaction Settlement:** Smart contracts could facilitate faster and more secure settlement of transactions between financial institutions, reducing counterparty risks.

EXISTING SYSTEM DISADVANTAGES

- **Security Concerns:** Smart contracts, like any code, are susceptible to vulnerabilities. Bugs or security loopholes in the code could lead to exploits and financial losses.
- **Legal and Regulatory Challenges:** The legal status of smart contracts and their adherence to existing financial regulations can be uncertain.
- **Dependency on Technology:** Smart contracts are reliant on technology infrastructure, and any disruptions or failures in the underlying technology (e.g., blockchain networks) could impact the functioning of these contracts.
- **Resistance to Change:** The financial industry is traditionally conservative and resistant to rapid change

PROPOSED SYSTEM

- **Digital Currency Issuance:**

  Central banks could issue digital currencies as programmable tokens through smart contracts, allowing for precise control over monetary supply.

- **Programmable Monetary Policy:**

  Smart contracts could automate the execution of monetary policies based on predefined conditions. For example, adjusting interest rates or liquidity measures in response to economic indicators.

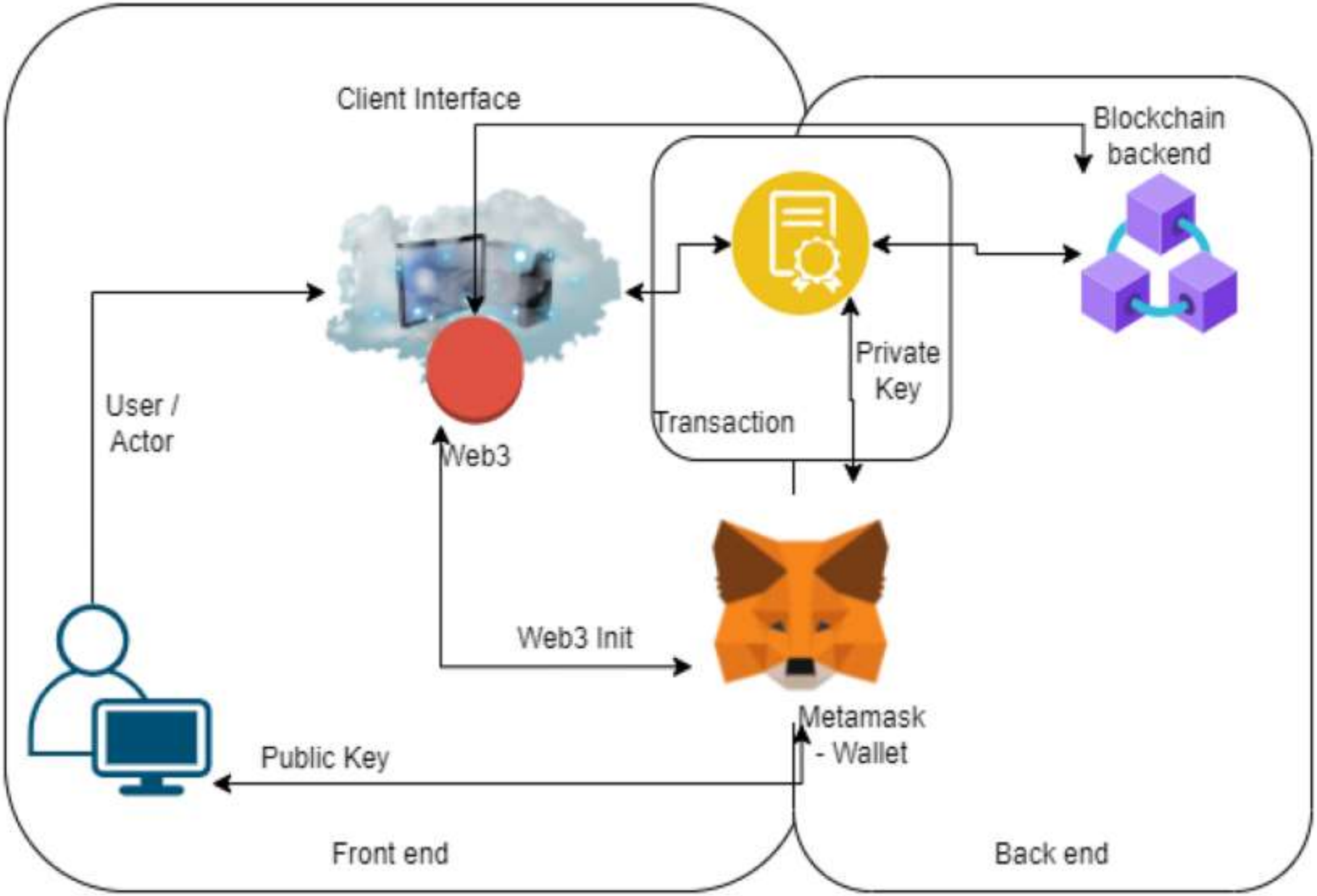
- **Transparent and Auditable Transactions:**

  All transactions could be recorded on a blockchain, providing a transparent and immutable ledger. This transparency could enhance accountability and facilitate audits.

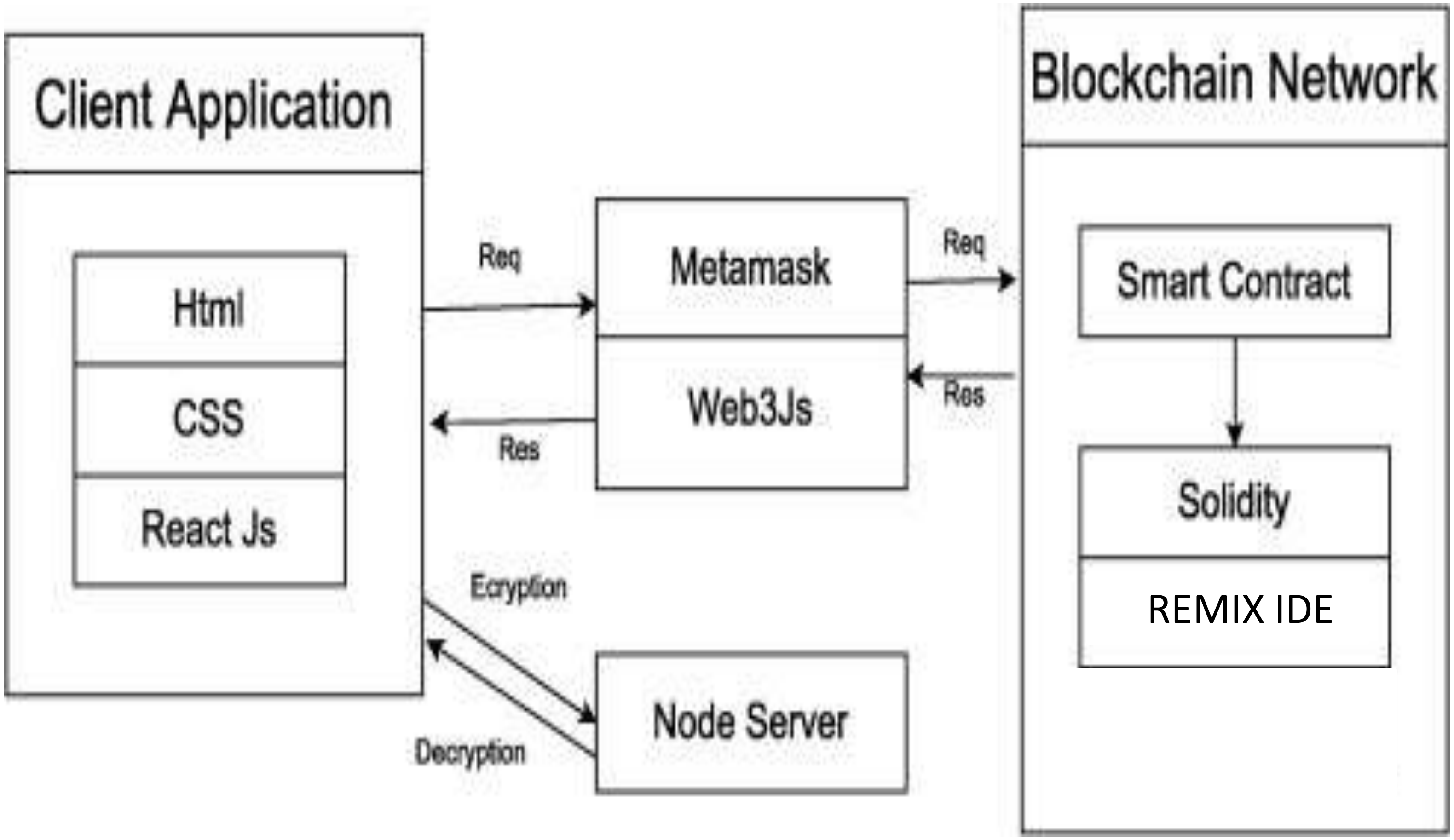
PROPOSED SYSTEM ADVANTAGES

- **Efficiency:** Sicker execution of transactions, policy implementations, and overall decision-making, streamlining the functioning of the central bank.
- **Transparency:** The decentralized nature of blockchain ensures transparency. All transactions and policy implementations are recorded on a public ledger, providing a clear and auditable history.
- **Reduced Fraud:** Smart contracts operate on secure blockchain technology, making it resistant to tampering.
- **Cost Savings:** Automation through smart contracts can lead to cost savings by eliminating the need for intermediaries, reducing paperwork, and increasing overall operational efficiency.

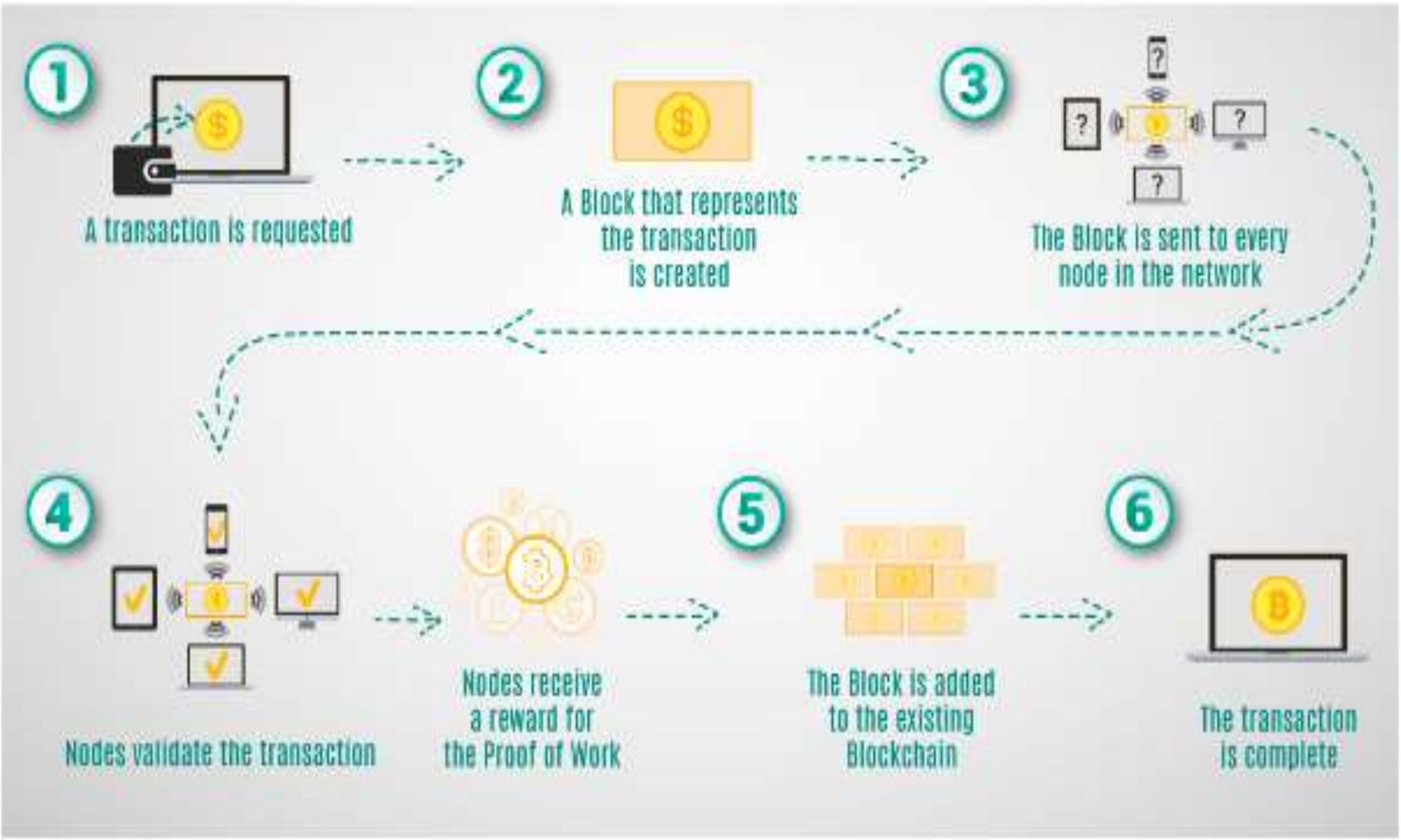
SOLUTION ARCHITECTURE



FLOW DIAGRAM



TECHNICAL ARCHITECTURE


1
$
A transaction is requested
2
$
A Block that represents the transaction is created
3
?
?
?
?
The Block is sent to every node in the network
4
Nodes validate the transaction
Nodes receive a reward for the Proof of Work
5
The Block is added to the existing Blockchain
6
B
The transaction is complete

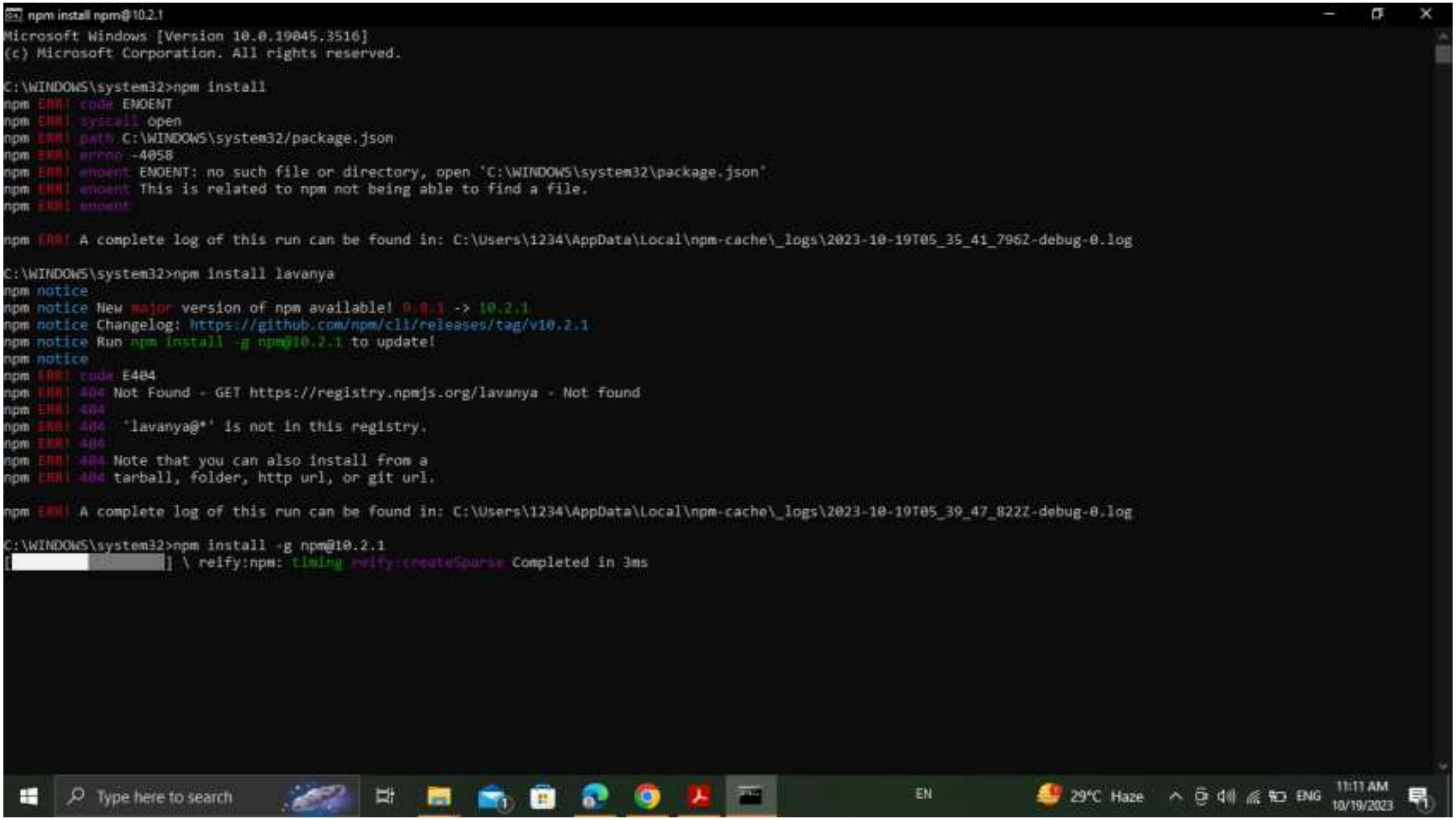
CONCLUSION

- In conclusion, integrating smart contracts into the functions of central banks presents a promising vision for the future of monetary systems.
- The automation, transparency, and efficiency offered by smart contracts could revolutionize the way central banks operate.
- Digital currency issuance, automated monetary policy execution, secure transaction settlements, regulatory compliance, data transparency, and streamlined cross-border transactions are among the potential benefits.
- However, this transition is not without its challenges. Security concerns, scalability issues, and the need for a robust legal and regulatory framework are significant hurdles that must be overcome.

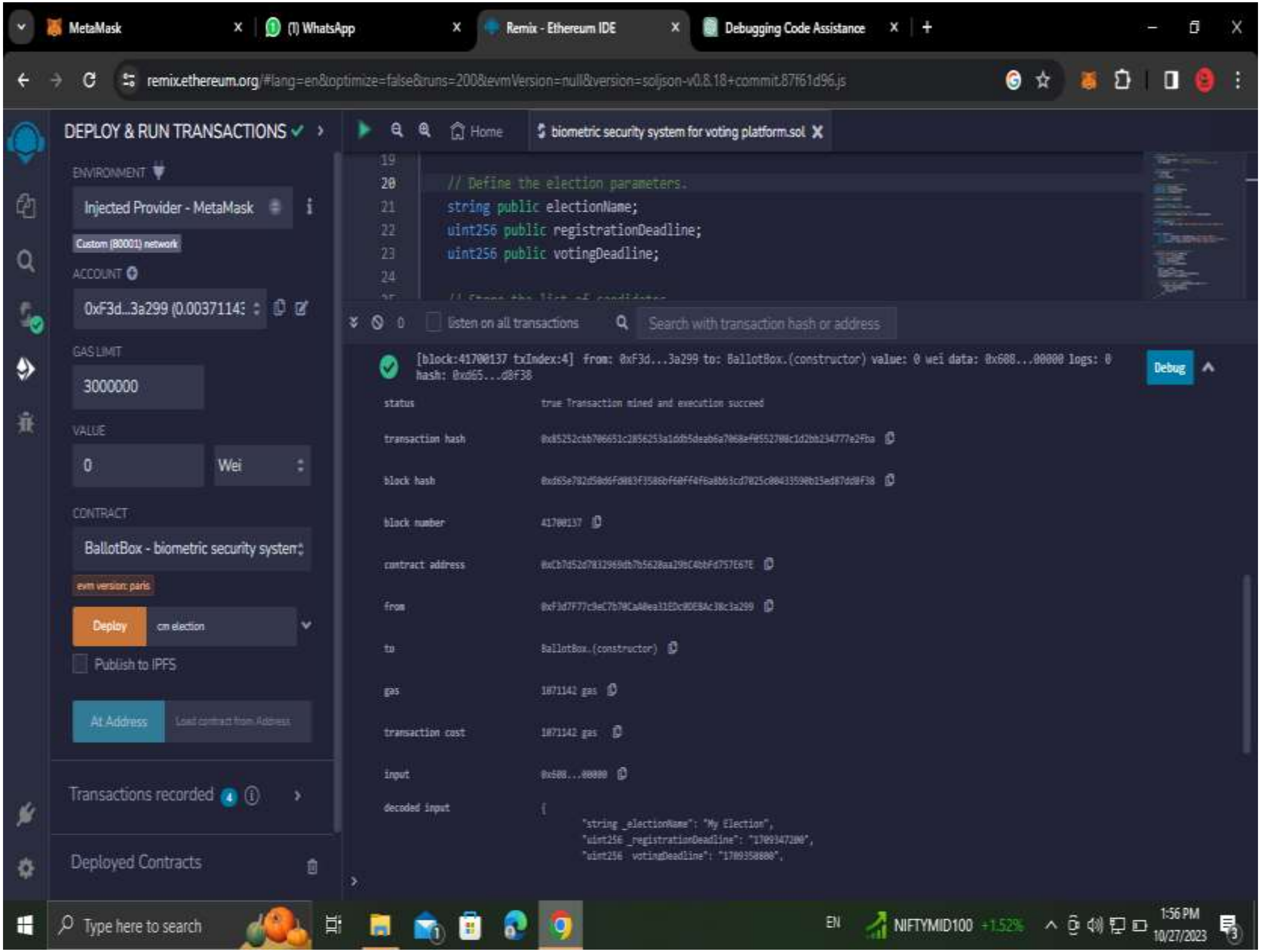
FUTURE SCOPE

- **Central Bank Digital Currencies (CBDCs):** Many central banks are exploring or actively developing CBDCs. Smart contracts could be integral to the design and operation of CBDCs, enabling programmable money and automating certain monetary policies.
- **Tokenization of Assets:** Central banks might tokenize traditional assets (such as government bonds) using smart contracts, making these assets more accessible and facilitating faster and more efficient trading on blockchain platforms.
- **Automated Monetary Policy:** Smart contracts could play a crucial role in automating various aspects of monetary policy, such as adjusting interest rates based on predefined economic indicators or automatically executing quantitative easing measures.

OUTPUT



```
Microsoft Windows [Version 10.0.19045.3516]
(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>npm install
npm ERR! code ENOENT
npm ERR! syscall open
npm ERR! path C:\WINDOWS\system32/package.json
npm ERR! errno -4058
npm ERR! enoent ENOENT: no such file or directory, open 'C:\WINDOWS\system32\package.json'
npm ERR! enoent This is related to npm not being able to find a file.
npm ERR! enoent

npm ERR! A complete log of this run can be found in: C:\Users\1234\AppData\Local\npm-cache\_logs\2023-10-19T05_35_41_796Z-debug-0.log

C:\WINDOWS\system32>npm install lavanya
npm notice
npm notice New major version of npm available! 9.8.1 -> 10.2.1
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.2.1
npm notice Run npm install -g npm@10.2.1 to update!
npm notice
npm ERR! code E404
npm ERR! 404 Not Found - GET https://registry.npmjs.org/lavanya - Not found
npm ERR! 404
npm ERR! 404  'lavanya@*' is not in this registry.
npm ERR! 404
npm ERR! 404 Note that you can also install from a
npm ERR! 404 tarball, folder, http url, or git url.

npm ERR! A complete log of this run can be found in: C:\Users\1234\AppData\Local\npm-cache\_logs\2023-10-19T05_39_47_822Z-debug-0.log

C:\WINDOWS\system32>npm install -g npm@10.2.1
[                   ] \ reify:npm: timing reify:createSparse Completed in 3ms
```



MetaMask
(1) WhatsApp
Remix - Ethereum IDE
Debugging Code Assistance
remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.18+commit.87f61d96.js
DEPLOY & RUN TRANSACTIONS ✓
Home
biometric security system for voting platform.sol

ENVIRONMENT
Injected Provider - MetaMask
Custom (80001) network

ACCOUNT
0xF3d...3a299 (0.00371143

GAS LIMIT
3000000

VALUE
0 Wei

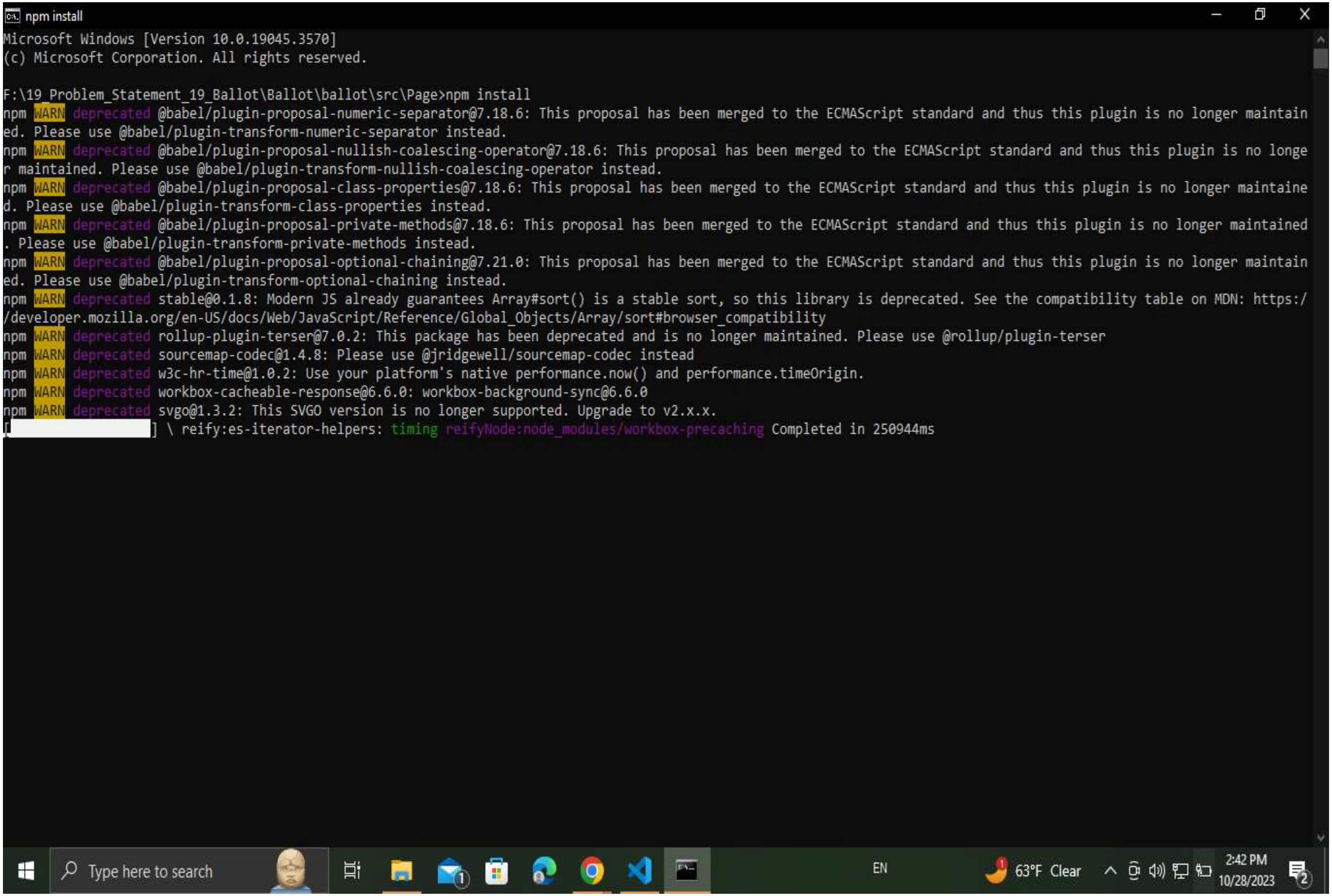
CONTRACT
BallotBox - biometric security system
evm version: paris

Deploy
cm election
☐ Publish to IPFS
At Address
Load contract from Address
Transactions recorded 4
Deployed Contracts

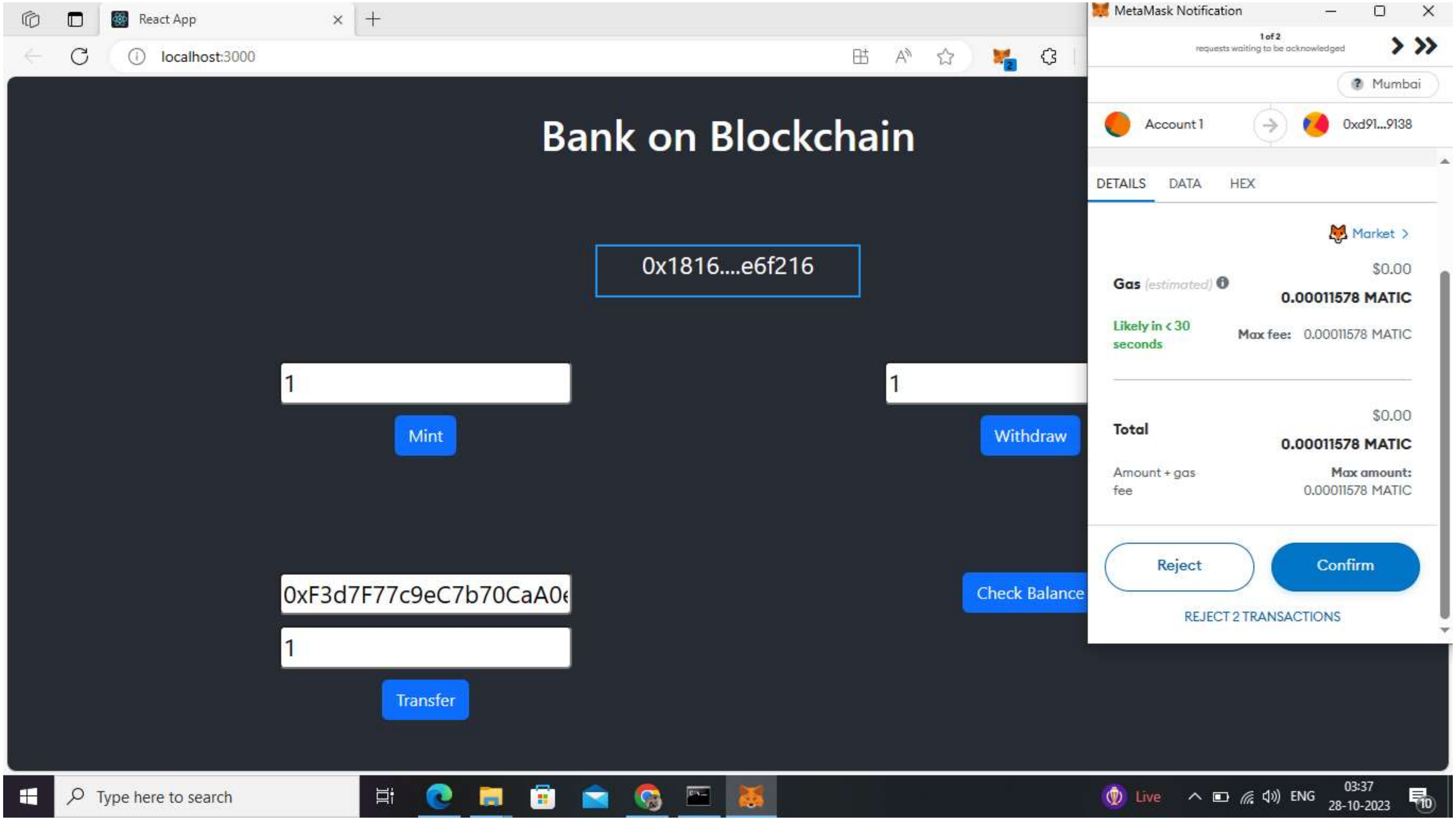
```
    // Define the election parameters.
    string public electionName;
    uint256 public registrationDeadline;
    uint256 public votingDeadline;

    // Store the list of candidates.
```

0
☐ listen on all transactions
Search with transaction hash or address
Debug

```
[block:41700137 txIndex:4] from: 0xF3d...3a299 to: BallotBox.(constructor) value: 0 wei data: 0x608...00000 logs: 0
hash: 0xd65...d8f38

status              true Transaction mined and execution succeed
transaction hash    0x85252cbb706651c2856253a1ddb5deab6a7068ef0552708c1d2bb234777e2fba
block hash          0xd65e782d50d6fd083f3586bf60ff4f6a8bb3cd7025c00433590b15ad87dd8f38
block number        41700137
contract address    0xCb7d52d7832969db7b5628aa29bC4bbFd757E67E
from                0xF3d7F77c9eC7b70CaA0ea31EDc0DEBAc38c3a299
to                  BallotBox.(constructor)
gas                 1071142 gas
transaction cost    1071142 gas
input               0x608...00000
decoded input       {
                            "string _electionName": "My Election",
                            "uint256 _registrationDeadline": "1709347200",
                            "uint256 _votingDeadline": "1709350800",
```



```
Microsoft Windows [Version 10.0.19045.3570]
(c) Microsoft Corporation. All rights reserved.

F:\19_Problem_Statement_19_Ballot\Ballot\ballot\src\Page>npm install
npm WARN deprecated @babel/plugin-proposal-numeric-separator@7.18.6: This proposal has been merged to the ECMAScript standard and thus this plugin is no longer maintain
ed. Please use @babel/plugin-transform-numeric-separator instead.
npm WARN deprecated @babel/plugin-proposal-nullish-coalescing-operator@7.18.6: This proposal has been merged to the ECMAScript standard and thus this plugin is no longe
r maintained. Please use @babel/plugin-transform-nullish-coalescing-operator instead.
npm WARN deprecated @babel/plugin-proposal-class-properties@7.18.6: This proposal has been merged to the ECMAScript standard and thus this plugin is no longer maintaine
d. Please use @babel/plugin-transform-class-properties instead.
npm WARN deprecated @babel/plugin-proposal-private-methods@7.18.6: This proposal has been merged to the ECMAScript standard and thus this plugin is no longer maintained
. Please use @babel/plugin-transform-private-methods instead.
npm WARN deprecated @babel/plugin-proposal-optional-chaining@7.21.0: This proposal has been merged to the ECMAScript standard and thus this plugin is no longer maintain
ed. Please use @babel/plugin-transform-optional-chaining instead.
npm WARN deprecated stable@0.1.8: Modern JS already guarantees Array#sort() is a stable sort, so this library is deprecated. See the compatibility table on MDN: https:/
/developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/sort#browser_compatibility
npm WARN deprecated rollup-plugin-terser@7.0.2: This package has been deprecated and is no longer maintained. Please use @rollup/plugin-terser
npm WARN deprecated sourcemap-codec@1.4.8: Please use @jridgewell/sourcemap-codec instead
npm WARN deprecated w3c-hr-time@1.0.2: Use your platform's native performance.now() and performance.timeOrigin.
npm WARN deprecated workbox-cacheable-response@6.6.0: workbox-background-sync@6.6.0
npm WARN deprecated svgo@1.3.2: This SVGO version is no longer supported. Upgrade to v2.x.x.
[██████████████████] \ reify:es-iterator-helpers: timing reifyNode:node_modules/workbox-precaching Completed in 250944ms
```



Bank on Blockchain

0x1816....e6f216

1

1

0xF3d7F77c9eC7b70CaA0e

1

1 of 2
requests waiting to be acknowledged

Mumbai

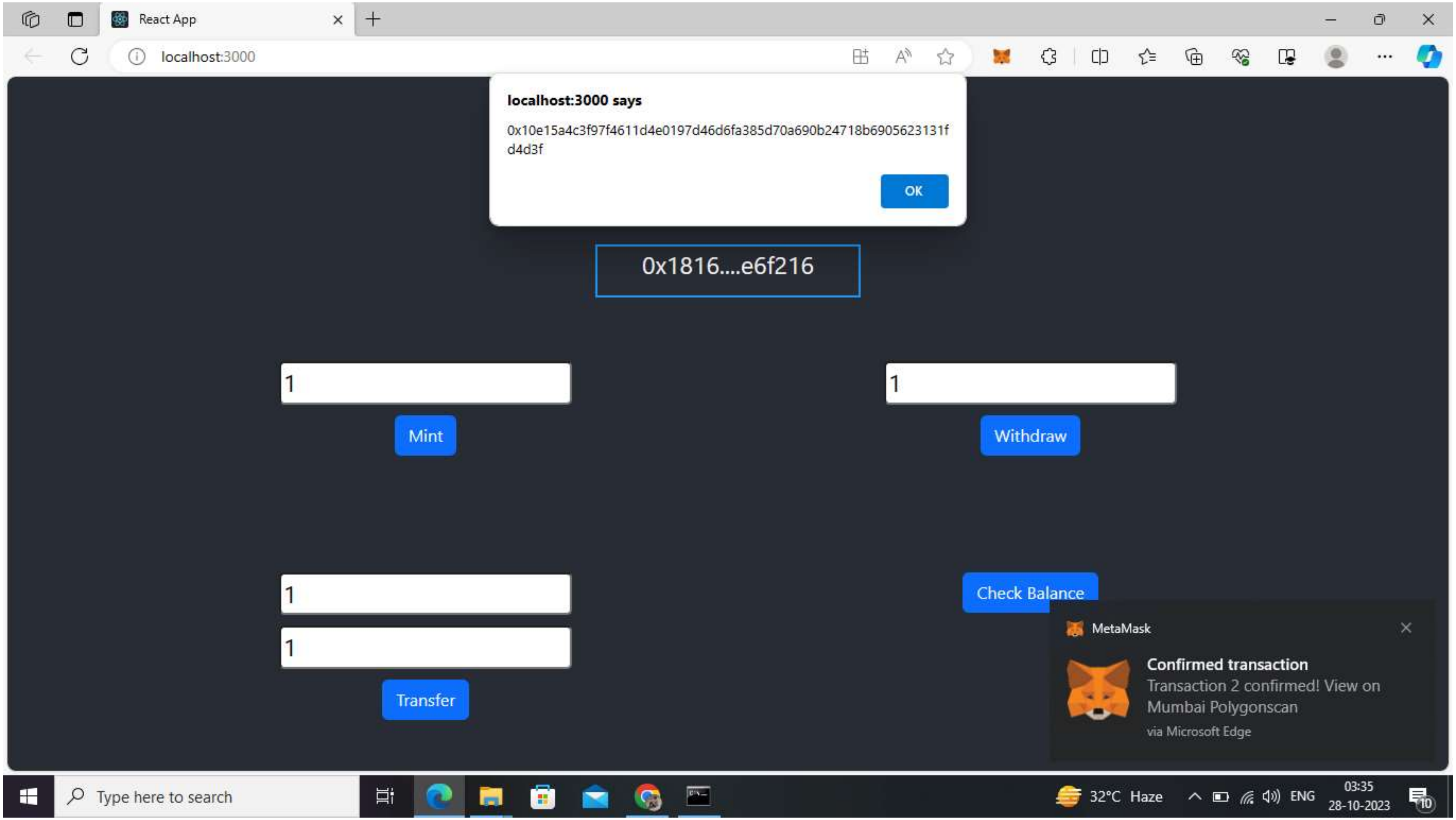
Account 1 → 0xd91...9138

Gas (estimated) $0.00
0.00011578 MATIC

Likely in < 30 seconds
Max fee: 0.00011578 MATIC

Total $0.00
0.00011578 MATIC

Amount + gas fee
Max amount: 0.00011578 MATIC



localhost:3000 says

```
0x10e15a4c3f97f4611d4e0197d46d6fa385d70a690b24718b6905623131f
d4d3f
```

MetaMask

Confirmed transaction

Transaction 2 confirmed! View on Mumbai Polygonscan

via Microsoft Edge

THANK YOU