INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

Voting on blockchain DApp

Cryptography & Blockchain class

INSA Lyon

Master MINDS

Participants: Loe Louis-Marie, Maïa Jouenne, Lancelot Tariot Camille

Introduction

- **Decentralized Application (DApp)**
  A web application that interacts with smart contracts on a blockchain.
- **Why a Voting DApp?**
  - **Transparency:** Votes are recorded on-chain and cannot be altered.
  - **Security:** Only valid voters can vote, enforced by smart contracts.
  - **Trustlessness:** No central authority controls the results.
  - **Auditability:** Anyone can verify results from the blockchain.
  - **Resilience:** No single point of failure; the network continues even if one node fails.
  - **Tamper-Proof Results:** Immutable vote records prevent manipulation.
  - **Vote integrity:** No fake vote, no vote tampering, no double voting

Why does Blockchain make sense in voting

Regular elections are known to be subjected to rigging.

In blockchain Voting:

- Elections cannot be rigged
- No one can vote twice
- Lack of central authority eliminates any kind of malfeasance in the election
- Elections are not the only use cases. Voting in general can be performed with integrity on the blockchain.

High-Level Architecture

The Voting DApp uses MetaMask as a secure wallet and transaction signer.

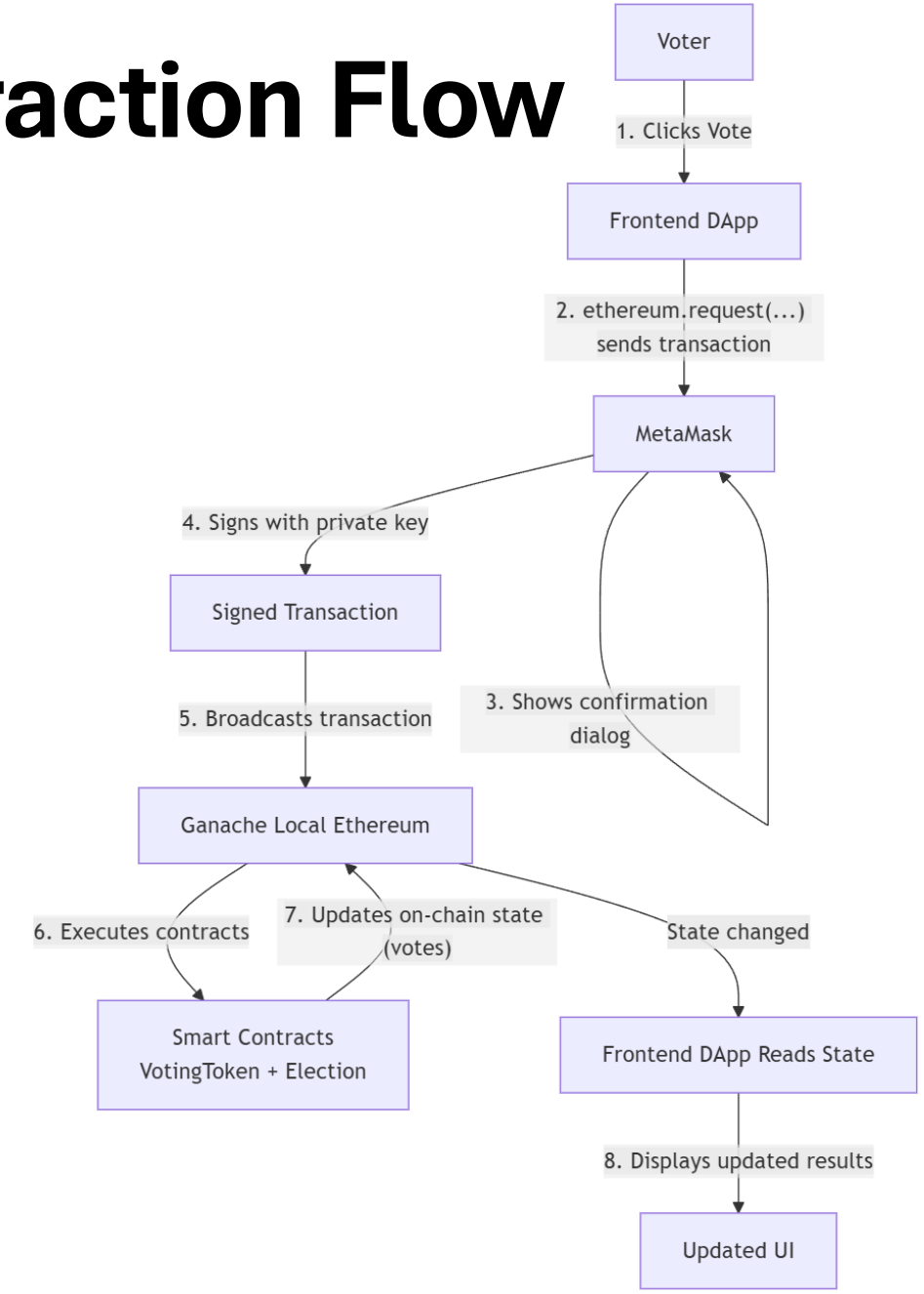
MetaMask Role:

- Stores private keys securely
- Requests transaction approval
- Signs transactions
- Sends signed transactions to Ganache

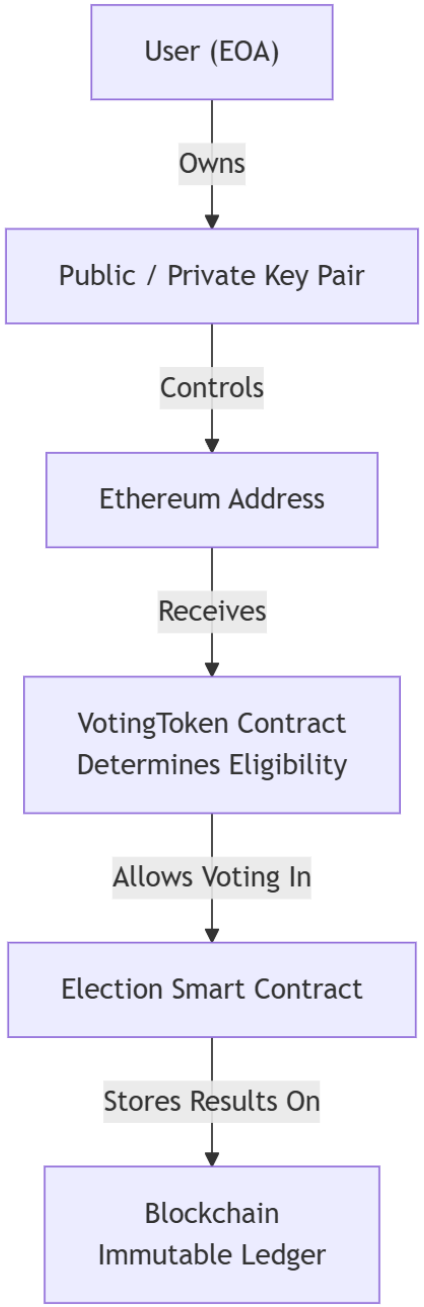
Components:

- **Frontend** (Web UI)
  HTML + JavaScript served by Parcel
- **MetaMask** (Wallet Provider)
  Injects window.ethereum, manages accounts, signs transactions
- **Ethereum Blockchain** (Ganache – Local Network)
  Executes smart contracts and stores immutable state
- **IPFS and ERC-20 token**
- **Smart Contracts** (Solidity)
  - VotingToken (voter eligibility)
  - Election (candidates, voting, results)

End-to-End Interaction Flow



Another Flow



Key points

- Only voters need blockchain accounts
- Candidates are stored as on-chain data (no wallet needed)
- MetaMask ensures security by signing transactions locally
- Blockchain enforces voting rules, not the frontend

Why IPFS in our Voting DApp?

- Blockchain is not designed for large data storage
- Storing detailed election results on-chain is:
  - expensive (gas costs)
  - inefficient
- Voting systems require:
  - transparency
  - integrity
  - auditability
- IPFS provides decentralized, content-addressed storage

Blockchain + IPFS: Separation of Responsibilities

Blockchain (On-chain)	IPFS (Off-chain)
Voting rules	Detailed results
Voter eligibility	Candidates list
Election lifecycle	Vote receipts (hashed)
Proof of integrity	JSON election archive
IPFS CID reference	Immutable file storage

How IPFS is used in our system ?

- At the end of an election:
  - Results are aggregated by the smart contract
  - A JSON file is generated by the frontend
- This file is uploaded to IPFS:
  - Content-addressed (CID = file hash)
  - Immutable by design
- The CID is stored on-chain in the Election smart contract
- Anyone can retrieve and verify the results via the IPFS gateway

Integrity, Transparency and Auditability

- IPFS uses content-based addressing:
  - Any change in the file changes its CID
- The smart contract stores only the CID:
  - guarantees integrity
  - prevents result tampering
- Results are publicly auditable without revealing voter identities

Presentation of the DApp frontend

Token-Gated Election

Voting DApp

Deploy contracts locally, distribute VOTE tokens, and run an on-chain election.

Account

Address: **Not connected**
Network: **N/A**
Network ID: **N/A**
Admin: **N/A** (**No**)
Token Balance: **0** VOTE
Has Voted: **N/A**
Election Active: **N/A**

Connect MetaMask | Refresh State

Connect MetaMask to continue.

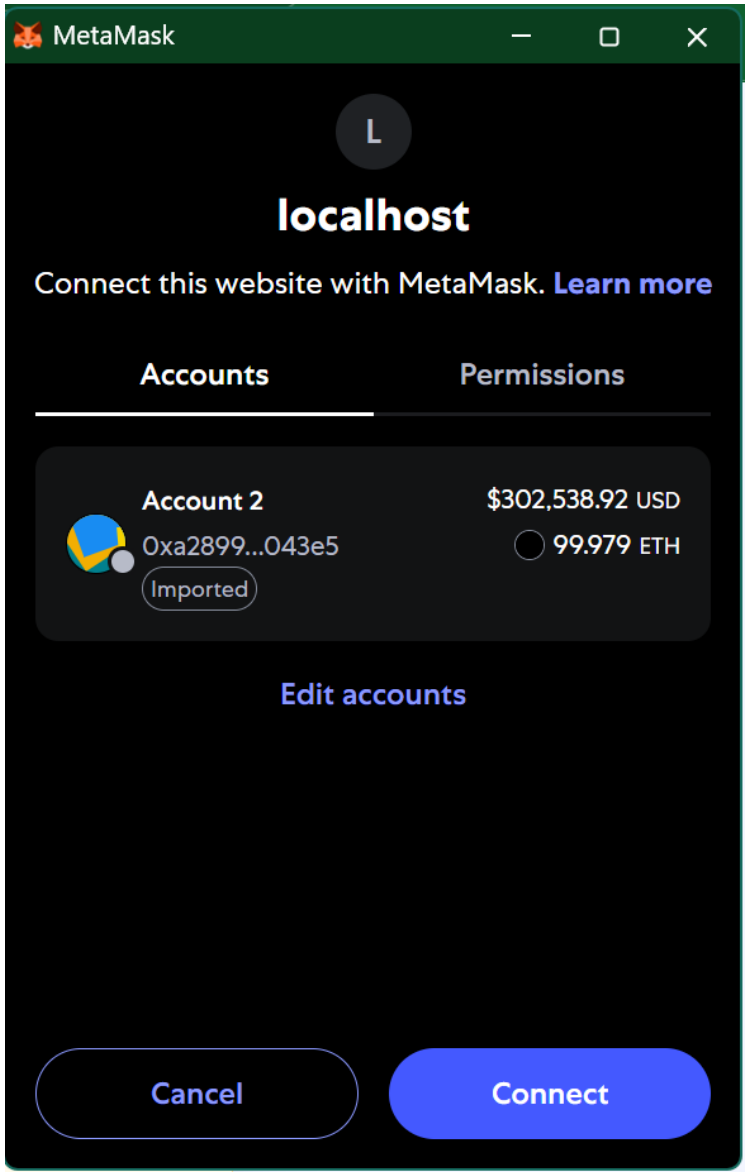
Connect MetaMask to continue.

Results

Show Live Results

Activity

```
Ready.
```


MetaMask
L
localhost
Connect this website with MetaMask. Learn more
Accounts
Permissions
Account 2
0xa2899...043e5
Imported
$302,538.92 USD
99.979 ETH
Edit accounts
Cancel
Connect

Account

Address: **0xa2899a180a559fb9cb1508c576b44294c18043e5**

Network: **1337 (0x539)**

Network ID: **5777**

Admin: **0xa2899A180a559fb9cB1508C576b44294c18043e5** (Yes)

Token Balance: **91** VOTE

Has Voted: **No**

Election Active: **Inactive**

Connect MetaMask | Refresh State

Admin Controls

1 **Add candidates**

Add candidates to define the election.

Candidate Name

Add Candidate

2 **Run election**

Add candidates before starting the election.

3 **Reset election**

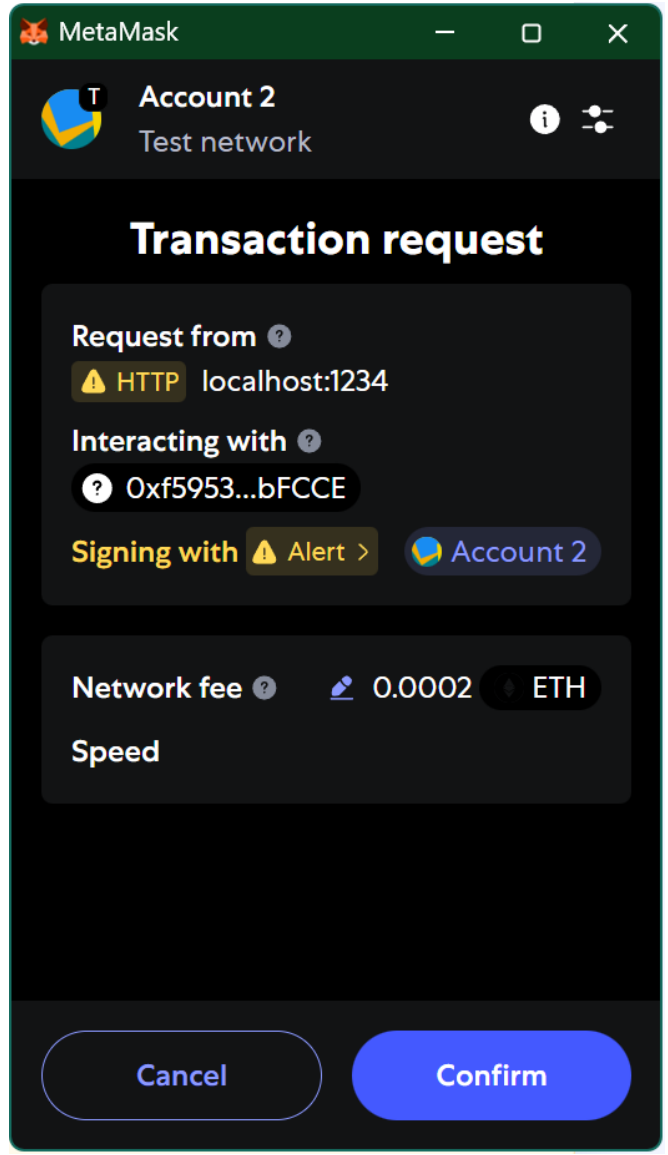
No active election. Add candidates to begin a new one.

Voting is not available. Election is not active.

Results

Show Live Results

No candidates yet.



Admin Controls

1

Add candidates

Candidates ready (3 total).

Candidate Name

Add Candidate

2

Run election

Start the election when you are ready.

Start Election

3

Reset election

Clear the current election and issue new voting tokens.

Reset Election

Voting is not available. Election is not active.

Results

Show Live Results

```
ID: 1, Name: Maïa, Votes: 0
ID: 2, Name: Louis, Votes: 0
ID: 3, Name: Lancelot, Votes: 0
```

Admin Controls

1

Add candidates

Election active (3 candidates).

Candidate Name

Add Candidate

2

Run election

Election is active. End it when ready.

End Election

3

Reset election

Reset will end the election, clear candidates, and issue new tokens.

Reset Election

Vote

#1 - Maïa

Candidate ID (optional)

Vote

Results

Show Live Results

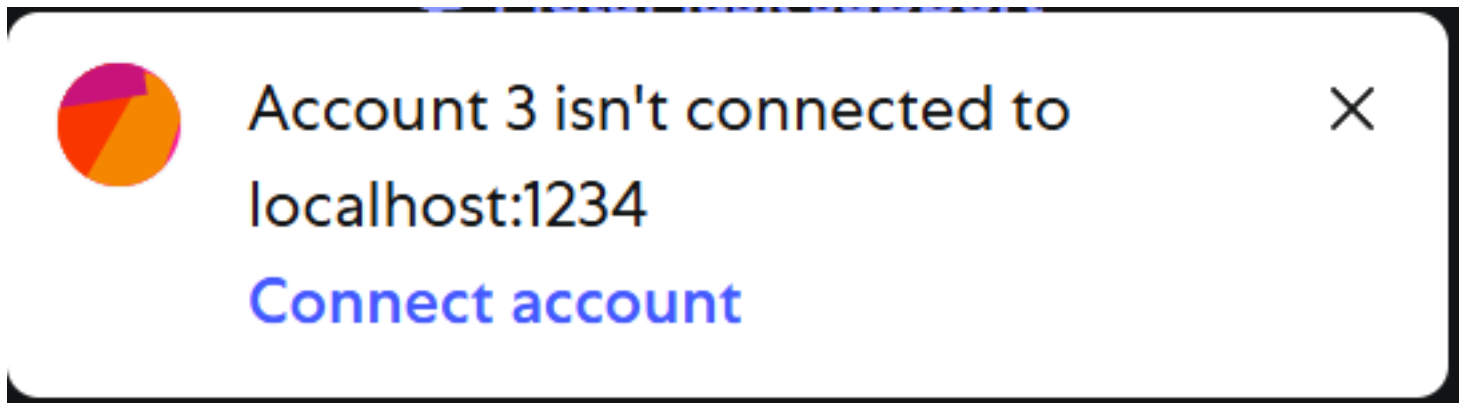
```
ID: 1, Name: Maïa, Votes: 0
ID: 2, Name: Louis, Votes: 0
ID: 3, Name: Lancelot, Votes: 0
```

Voting is not available. You have already voted.

Results

Show Live Results

```
ID: 1, Name: Maïa, Votes: 1
ID: 2, Name: Louis, Votes: 0
ID: 3, Name: Lancelot, Votes: 0
```


Account 3 isn't connected to
localhost:1234
Connect account

Admin controls are only available to the admin account.

Vote

#1 - Maïa

Candidate ID (optional)

Vote

Results

Show Live Results

```
ID: 1, Name: Maïa, Votes: 1
ID: 2, Name: Louis, Votes: 0
ID: 3, Name: Lancelot, Votes: 0
```

Admin Controls

1

Add candidates

Election active (3 candidates).

Candidate Name

Add Candidate

2

Run election

Election is active. End it when ready.

End Election

3

Reset election

Reset will end the election, clear candidates, and issue new tokens.

Reset Election

Voting is not available. Election is not active. You have already voted.

Results

Show Live Results

```
ID: 1, Name: Maïa, Votes: 2
ID: 2, Name: Louis, Votes: 1
ID: 3, Name: Lancelot, Votes: 0
```

Activity

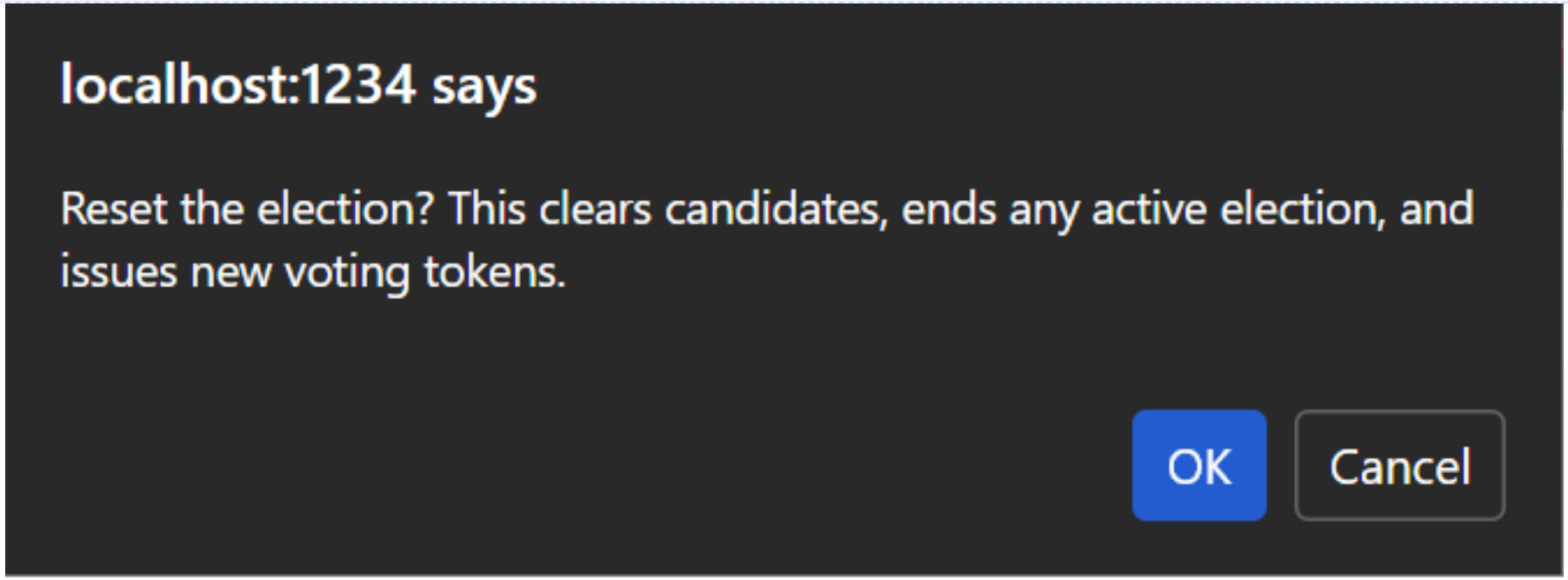
```
[13:38:43] Election ended.
[13:37:19] Vote cast for candidate ID: 2.
[13:36:57] Vote cast for candidate ID: 1.
[13:35:28] Vote cast for candidate ID: 1.
[13:31:44] Election started.
[13:31:07] Candidate added: Lancelot
[13:31:02] Candidate added: Louis
[13:30:52] Candidate added: Maïa
[13:29:25] Connected successfully.
Ready.
```

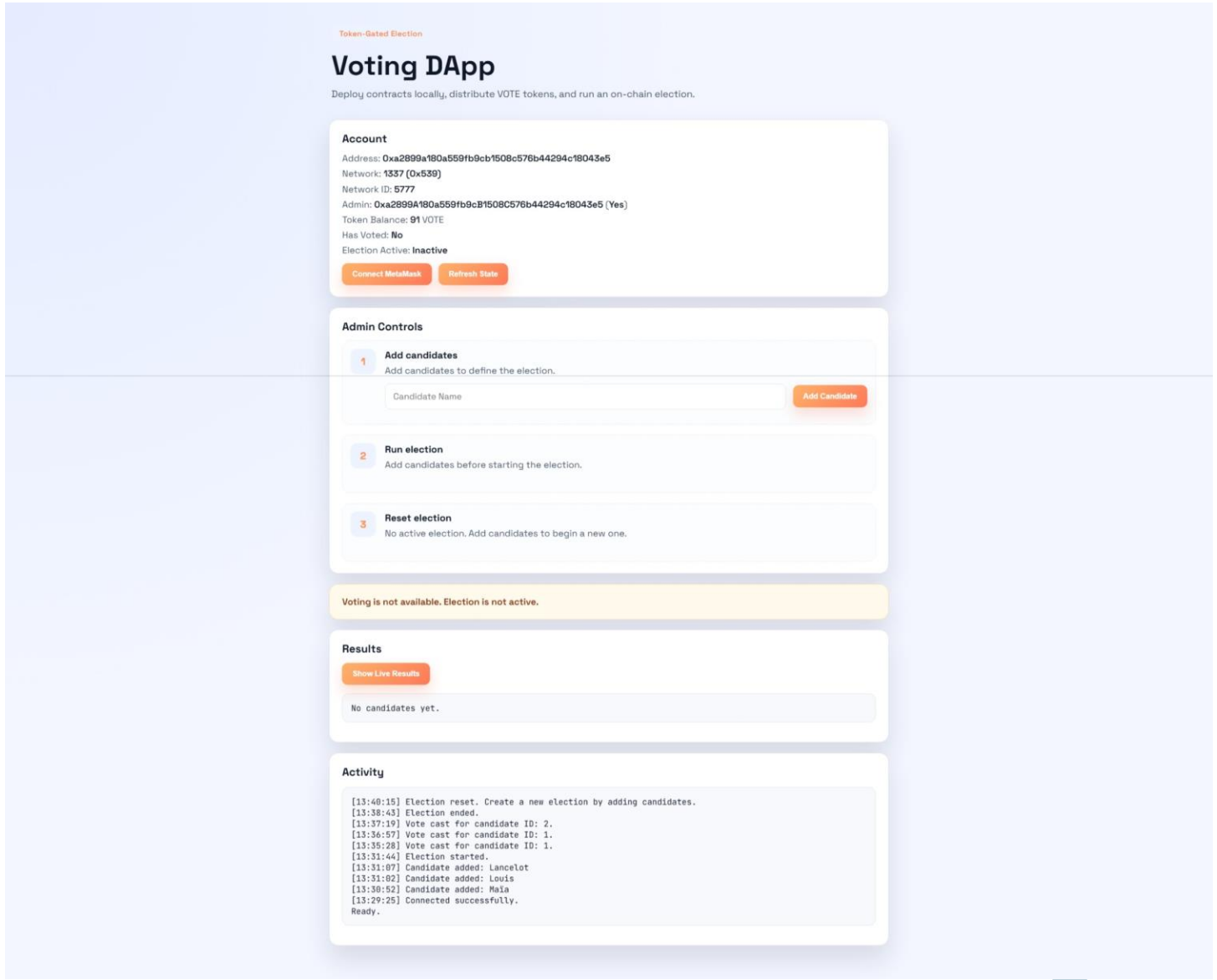
3

Reset election

Clear the current election and issue new voting tokens.

Reset Election




Token-Gated Election
Voting DApp
Deploy contracts locally, distribute VOTE tokens, and run an on-chain election.
Account
Address: 0xa2899a180a559fb9cb1508c576b44294c18043e5
Network: 1337 (0x539)
Network ID: 5777
Admin: 0xa2899A180a559fb9cB1508C576b44294c18043e5 (Yes)
Token Balance: 91 VOTE
Has Voted: No
Election Active: Inactive
Connect MetaMask
Refresh State
Admin Controls
1
Add candidates
Add candidates to define the election.
Candidate Name
Add Candidate
2
Run election
Add candidates before starting the election.
3
Reset election
No active election. Add candidates to begin a new one.
Voting is not available. Election is not active.
Results
Show Live Results
No candidates yet.
Activity
[13:40:15] Election reset. Create a new election by adding candidates.
[13:38:43] Election ended.
[13:37:19] Vote cast for candidate ID: 2.
[13:36:57] Vote cast for candidate ID: 1.
[13:35:28] Vote cast for candidate ID: 1.
[13:31:44] Election started.
[13:31:07] Candidate added: Lancelot
[13:31:02] Candidate added: Louis
[13:30:52] Candidate added: Maïa
[13:29:25] Connected successfully.
Ready.

Demo

Further possible improvements

1. Use **Oracle** to connect to the national voting register
2. Use **biometrics** (fingerprint, iris, face) to add new functionality
3. Handle **multiple elections** simultaneously
4. Handle voting secrecy as required in a real world election

Question