Blockchain's impact on the sports industry and fan engagement

Drishti Samvedi - D17B/63

Introduction

- Blockchain is revolutionizing the sports industry by enhancing transparency, security, and efficiency across multiple facets.
- It has introduced innovative concepts like fan tokenization, allowing fans to engage actively with their favorite teams.
- Despite facing challenges, including regulatory hurdles, blockchain's potential to transform how sports organizations operate and interact with fans is increasingly evident, promising a more decentralized, transparent, and fan-centric future for the sports industry

Fan Engagement in Sports

Parameter	Traditional Approach	Using Blockchain
Merchandise	Purchase physical items (eg: jerseys)	Digital collectibles
Ticketing	Traditional ticketing systems	Transparent and secure ticketing
Fan Tokens	Not applicable	Voting rights, access to exclusive content/events
Smart Contracts	Not applicable	Automated for fair athlete payments
Global Reach	Local and regional focus	Global reach, breaking geographical barriers
Interactivity	Limited direct interaction	Direct and interactive fan-team relationships

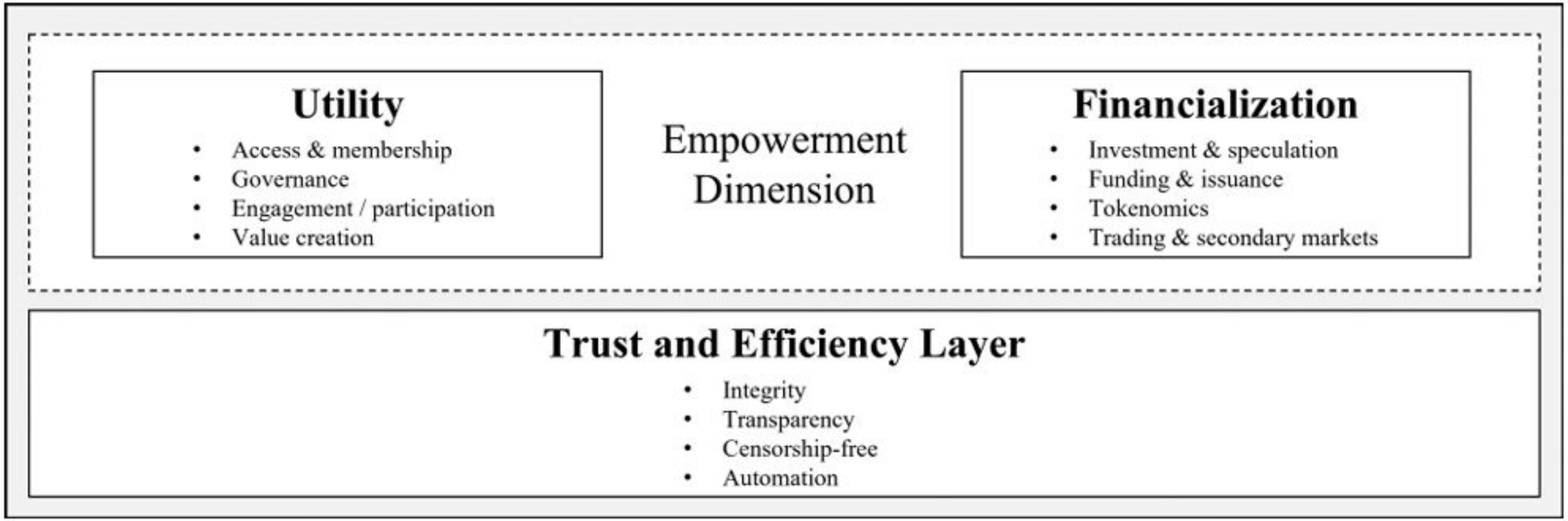
Fan Tokens

- Type of cryptocurrency that entitles owners to many fan-related membership benefits.
- Ownership and Voting Rights on various club decisions.
- Exclusive content, merchandise, events, and experiences related to the sports club.
- Fan token markets have grown rapidly, with significant market capitalization.
- Fan-centric approach, allowing fans to actively participate in club-related decisions and activities.
- Enable clubs to connect with their global fan base, irrespective of geographical location.

How do Fan Tokens work?

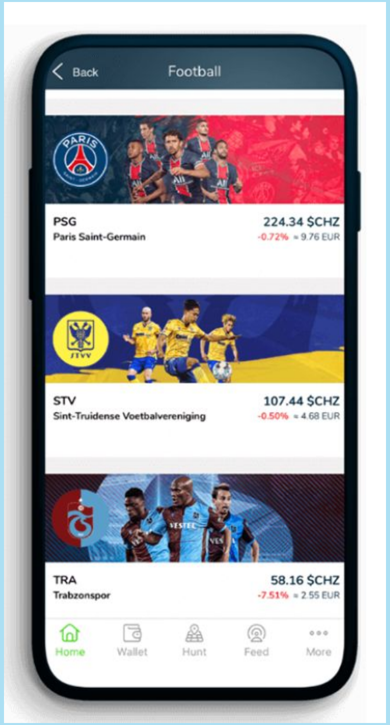
1. Issuance by Sports Clubs
2. Token Sale
3. Ownership and Wallets
4. Blockchain Technology
5. Voting Rights, Exclusive Content and Benefits
6. Token Trading
7. Regulatory Considerations

Blockchain-based Fan Token Framework



Socios Fan Token Platform

How Socios works?

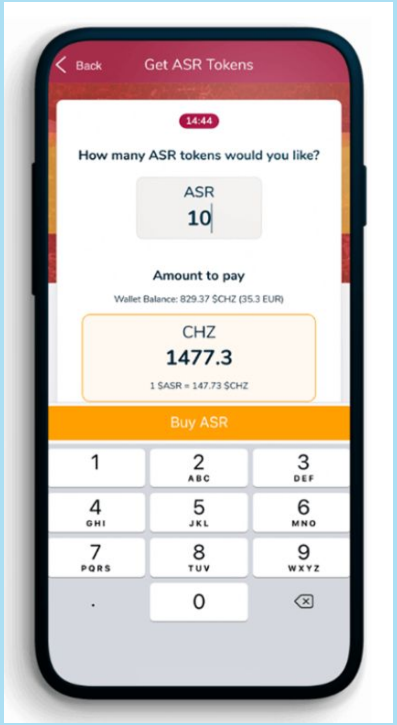

Back Football

PSG 224.34 $CHZ
Paris Saint-Germain -0.72% ≈ 9.76 EUR

STV 107.44 $CHZ
Sint-Truidense Voetbalvereniging -0.50% ≈ 4.68 EUR

TRA 58.16 $CHZ
Trabzonspor -7.51% ≈ 2.55 EUR

STEP 1
Sign up to Socios.com


Back Get ASR Tokens

14:44

How many ASR tokens would you like?

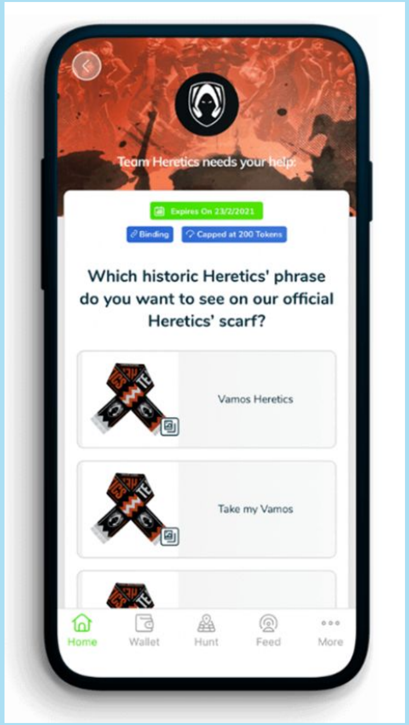
ASR
10

Amount to pay

Wallet Balance: 829.37 $CHZ (35.3 EUR)

CHZ
1477.3
1 $ASR = 147.73 $CHZ

STEP 2
Buy Fan Tokens & hunt free Tokens


Team Heretics needs your help

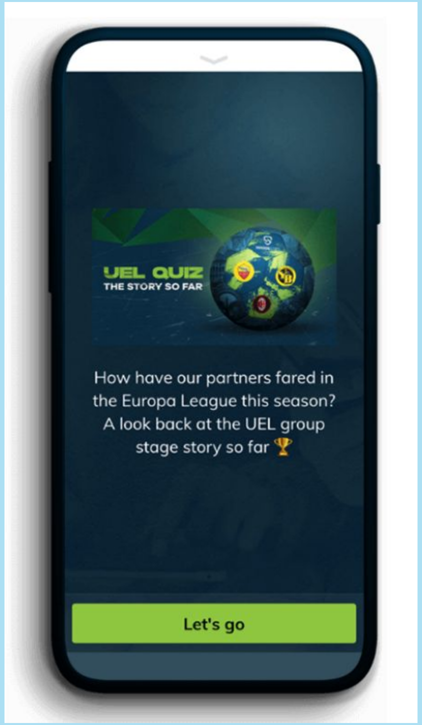
Expires On 23/2/2021
Binding
Capped at 200 Tokens

Which historic Heretics' phrase do you want to see on our official Heretics' scarf?

Vamos Heretics

Take my Vamos

STEP 3
Vote on team polls



How have our partners fared in the Europa League this season? A look back at the UEL group stage story so far 🏆

STEP 4
Engage & score rewards

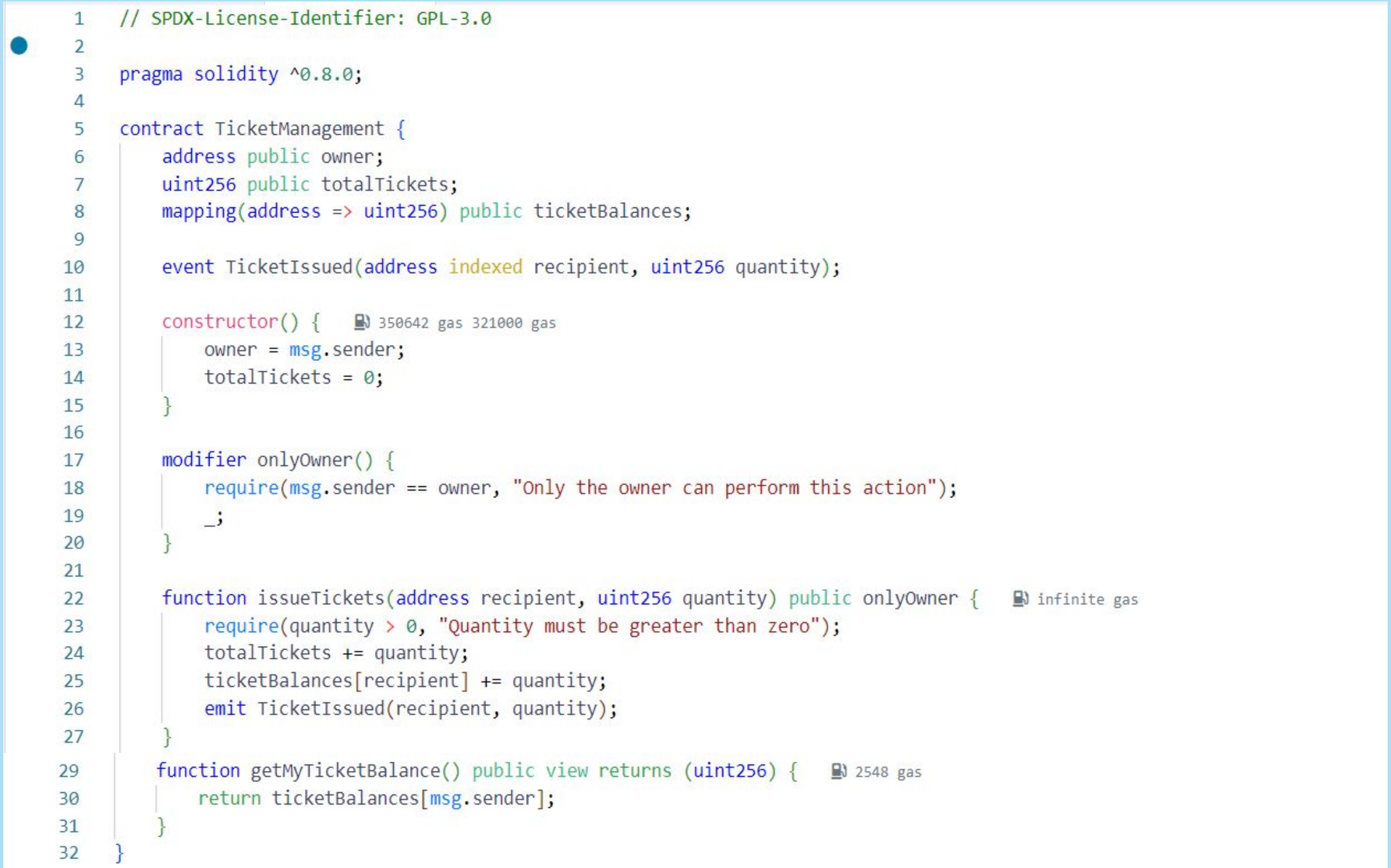
Literature Review

Enhancing Trust, Efficiency, and Empowerment in Sports: Developing a Blockchain-Based Fan Token Framework

Conclusion

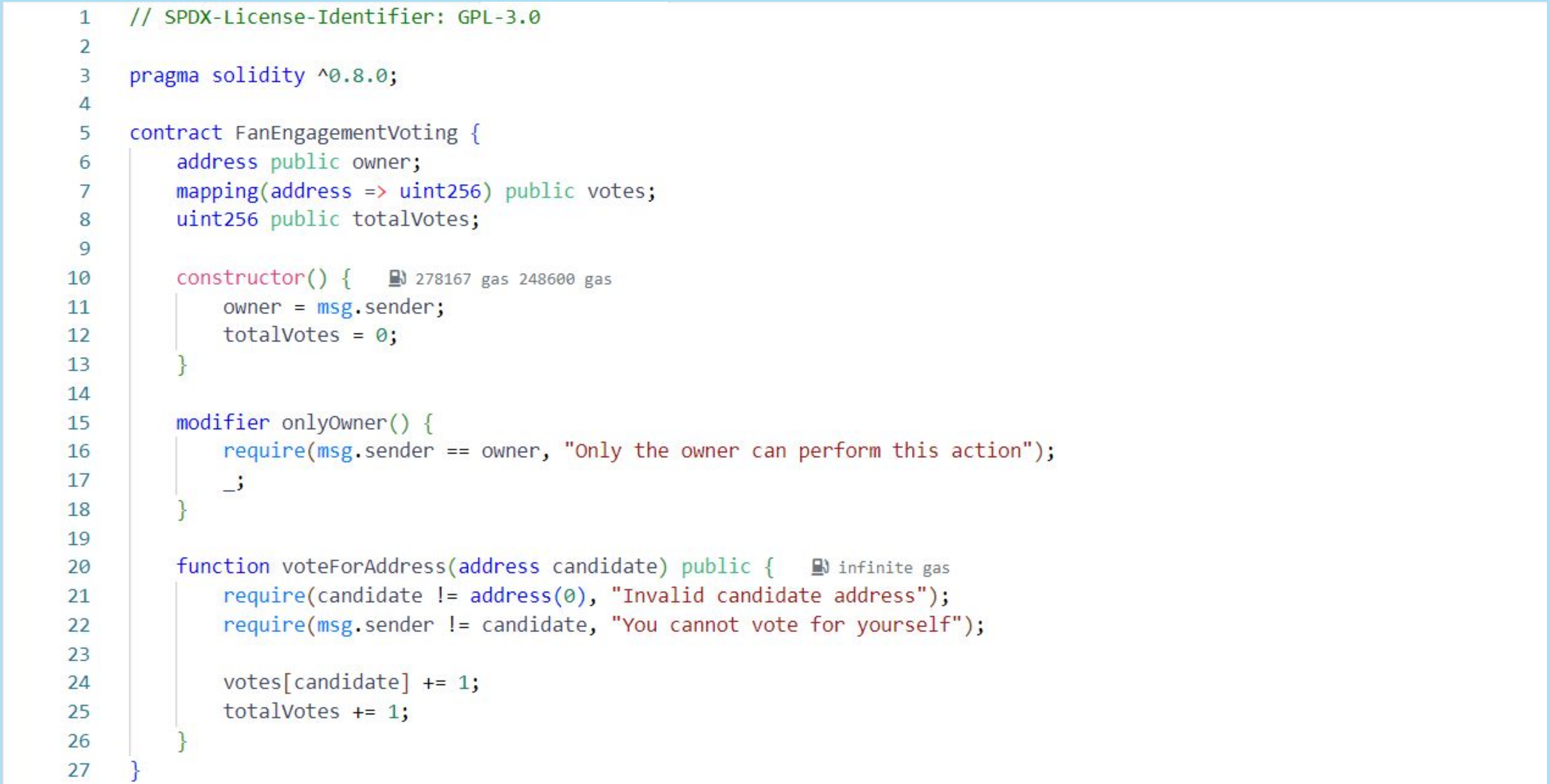
- Fan tokens represent a remarkable convergence of blockchain technology and sports fandom, offering an innovative and engaging way for fans to connect with their favorite teams and organizations.
- While fan tokens have gained popularity primarily in football (soccer) clubs, their potential extends across various sports and industries.
- They signify a shift towards more interactive and inclusive fan experiences, underlining the ever-evolving relationship between technology and sports fandom.

Experiment 6: Smart Contract 1



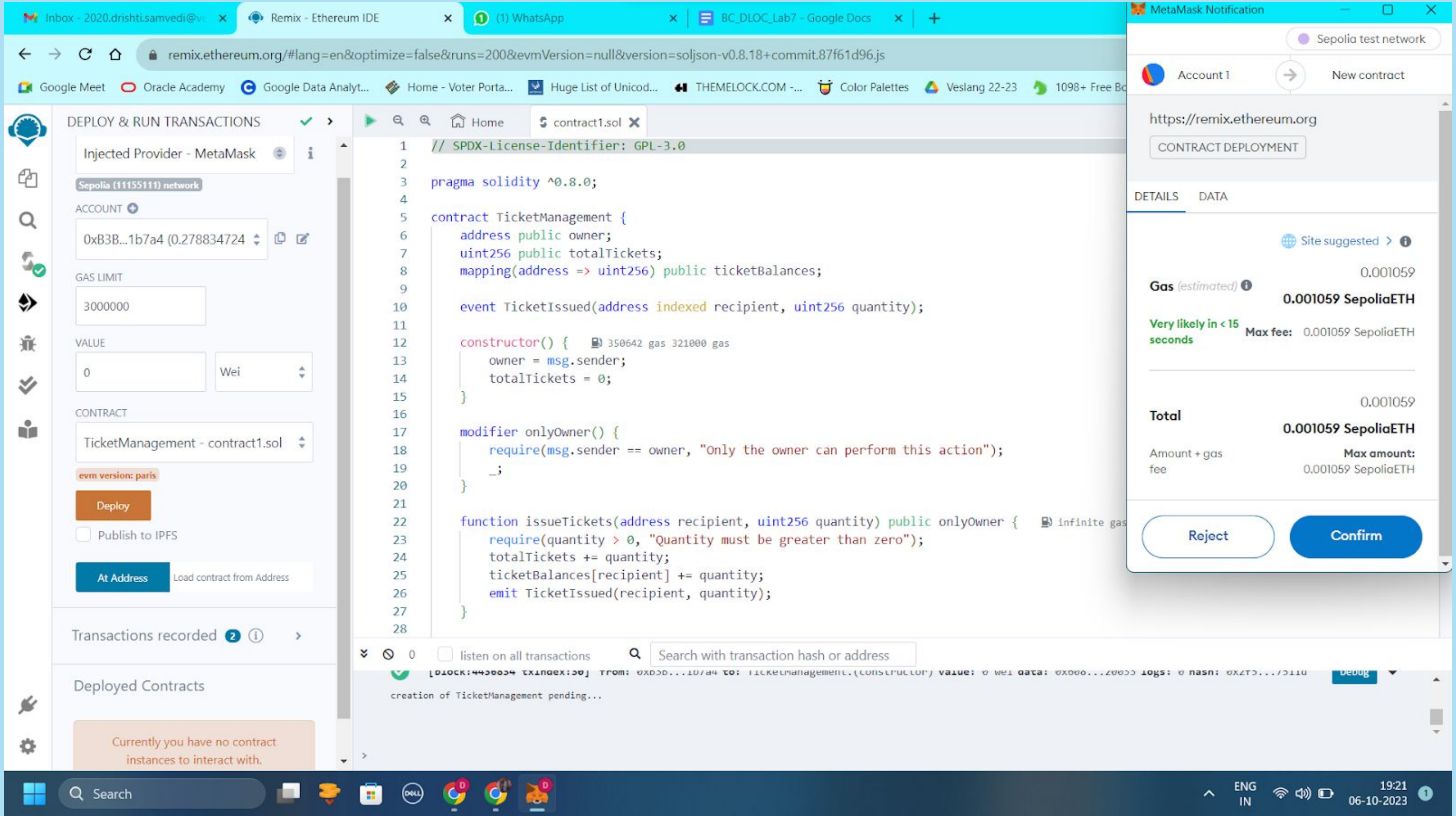
```
// SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.0;

contract TicketManagement {
    address public owner;
    uint256 public totalTickets;
    mapping(address => uint256) public ticketBalances;

    event TicketIssued(address indexed recipient, uint256 quantity);

    constructor() {    350642 gas 321000 gas
        owner = msg.sender;
        totalTickets = 0;
    }

    modifier onlyOwner() {
        require(msg.sender == owner, "Only the owner can perform this action");
        _;
    }

    function issueTickets(address recipient, uint256 quantity) public onlyOwner {    infinite gas
        require(quantity > 0, "Quantity must be greater than zero");
        totalTickets += quantity;
        ticketBalances[recipient] += quantity;
        emit TicketIssued(recipient, quantity);
    }
    function getMyTicketBalance() public view returns (uint256) {    2548 gas
        return ticketBalances[msg.sender];
    }
}
```

Experiment 6: Smart Contract 2



```
// SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.0;

contract FanEngagementVoting {
    address public owner;
    mapping(address => uint256) public votes;
    uint256 public totalVotes;

    constructor() {    278167 gas 248600 gas
        owner = msg.sender;
        totalVotes = 0;
    }

    modifier onlyOwner() {
        require(msg.sender == owner, "Only the owner can perform this action");
        _;
    }

    function voteForAddress(address candidate) public {    infinite gas
        require(candidate != address(0), "Invalid candidate address");
        require(msg.sender != candidate, "You cannot vote for yourself");

        votes[candidate] += 1;
        totalVotes += 1;
    }
}
```

Experiment 7: Integration with Metamask



Injected Provider - MetaMask

Sepolia (11155111) network

ACCOUNT: 0xB3B...1b7a4 (0.278834724

GAS LIMIT: 3000000

VALUE: 0 Wei

CONTRACT: TicketManagement - contract1.sol

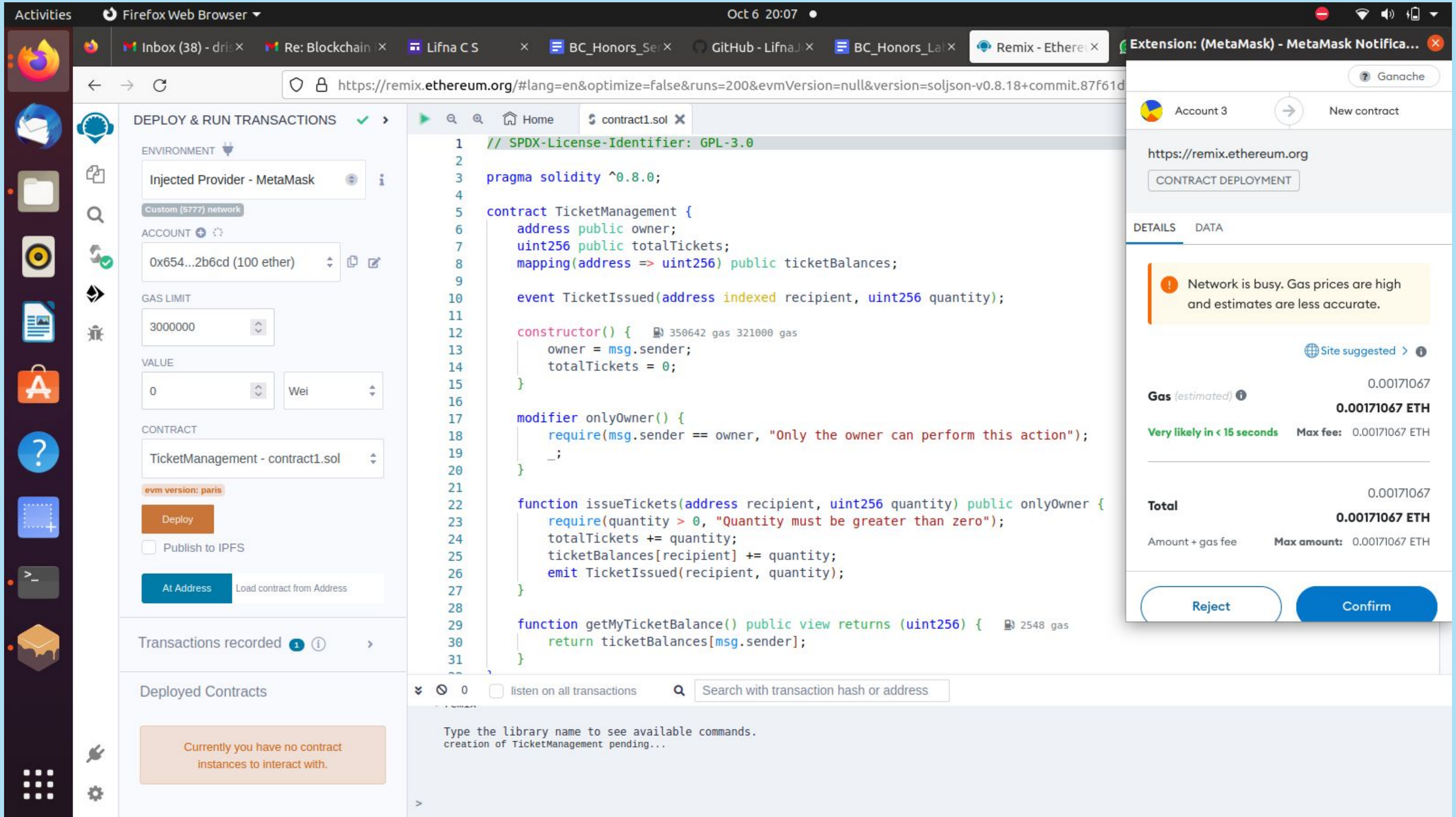
evm version: paris

```
// SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.0;

contract TicketManagement {
    address public owner;
    uint256 public totalTickets;
    mapping(address => uint256) public ticketBalances;

    event TicketIssued(address indexed recipient, uint256 quantity);

    constructor() {    350642 gas 321000 gas
        owner = msg.sender;
        totalTickets = 0;
    }

    modifier onlyOwner() {
        require(msg.sender == owner, "Only the owner can perform this action");
        _;
    }

    function issueTickets(address recipient, uint256 quantity) public onlyOwner {    infinite gas
        require(quantity > 0, "Quantity must be greater than zero");
        totalTickets += quantity;
        ticketBalances[recipient] += quantity;
        emit TicketIssued(recipient, quantity);
    }
```

```
[block:4436854 txIndex:30] from: 0xb3b...1b7a4 to: TicketManagement.(constructor) value: 0 wei data: 0x608...20033 logs: 0 hash: 0x2f5...7511d
creation of TicketManagement pending...
```

Gas (estimated)	0.001059
	0.001059 SepoliaETH
Very likely in < 15 seconds	Max fee: 0.001059 SepoliaETH
Total	0.001059
	0.001059 SepoliaETH
Amount + gas fee	Max amount: 0.001059 SepoliaETH

Experiment 8: Deploying with Ganache Account



https://remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.18+commit.87f61d

DEPLOY & RUN TRANSACTIONS

ENVIRONMENT: Injected Provider - MetaMask
Custom (5777) network

ACCOUNT: 0x654...2b6cd (100 ether)

GAS LIMIT: 3000000

VALUE: 0 Wei

CONTRACT: TicketManagement - contract1.sol
evm version: paris

Transactions recorded 1

Deployed Contracts

Currently you have no contract instances to interact with.

```
// SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.0;

contract TicketManagement {
    address public owner;
    uint256 public totalTickets;
    mapping(address => uint256) public ticketBalances;

    event TicketIssued(address indexed recipient, uint256 quantity);

    constructor() {    350642 gas 321000 gas
        owner = msg.sender;
        totalTickets = 0;
    }

    modifier onlyOwner() {
        require(msg.sender == owner, "Only the owner can perform this action");
        _;
    }

    function issueTickets(address recipient, uint256 quantity) public onlyOwner {
        require(quantity > 0, "Quantity must be greater than zero");
        totalTickets += quantity;
        ticketBalances[recipient] += quantity;
        emit TicketIssued(recipient, quantity);
    }

    function getMyTicketBalance() public view returns (uint256) {    2548 gas
        return ticketBalances[msg.sender];
    }
```

0
listen on all transactions
Search with transaction hash or address

```
Type the library name to see available commands.
creation of TicketManagement pending...
```

Extension: (MetaMask) - MetaMask Notifica...

Ganache

Account 3 → New contract

https://remix.ethereum.org

CONTRACT DEPLOYMENT

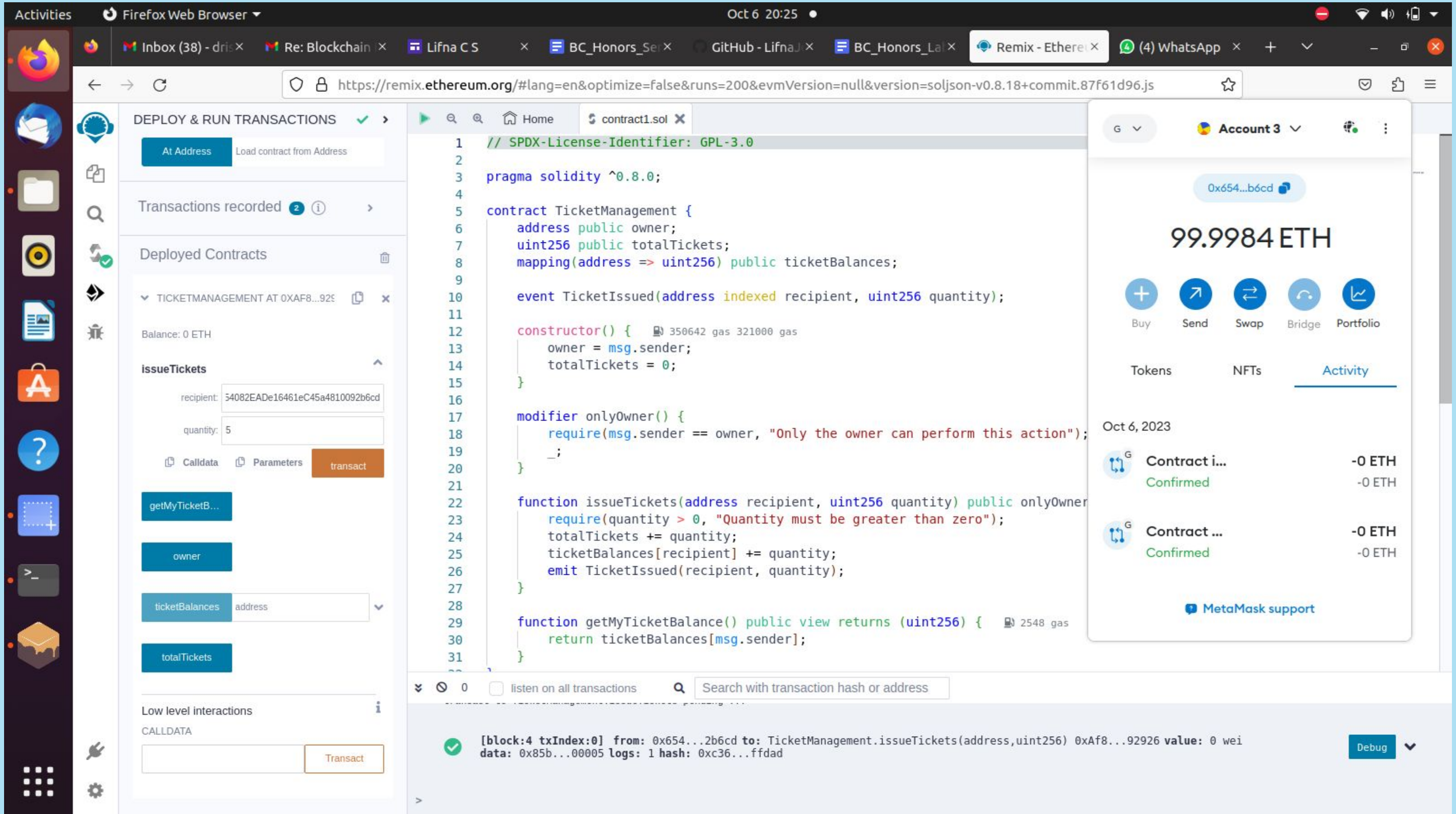
Network is busy. Gas prices are high and estimates are less accurate.

Site suggested

Gas (estimated) 0.00171067
0.00171067 ETH
Very likely in < 15 seconds   Max fee: 0.00171067 ETH

Total 0.00171067
0.00171067 ETH
Amount + gas fee   Max amount: 0.00171067 ETH

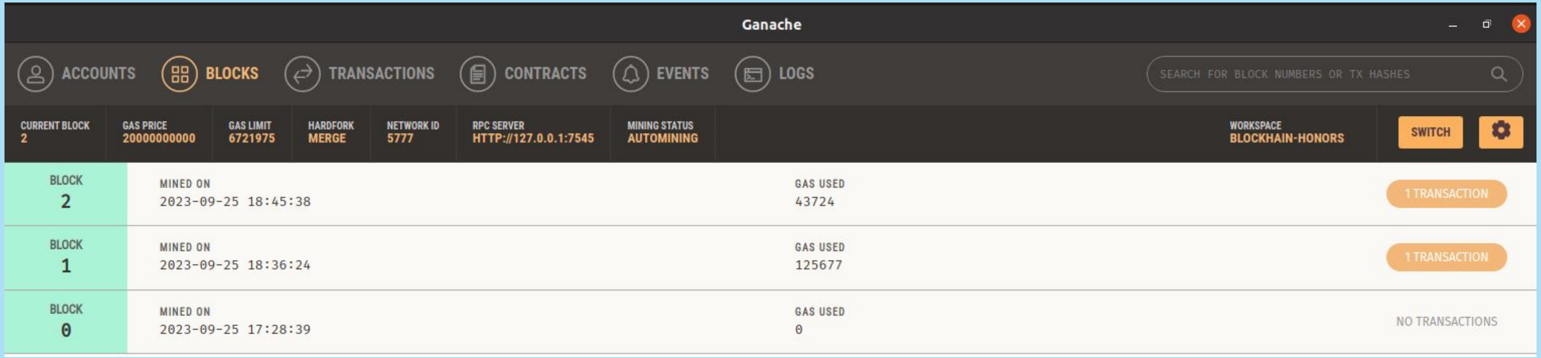
Experiment 8: Deploying with Ganache Account



```
// SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.0;

contract TicketManagement {
    address public owner;
    uint256 public totalTickets;
    mapping(address => uint256) public ticketBalances;

    event TicketIssued(address indexed recipient, uint256 quantity);

    constructor() {   350642 gas 321000 gas
        owner = msg.sender;
        totalTickets = 0;
    }

    modifier onlyOwner() {
        require(msg.sender == owner, "Only the owner can perform this action");
        _;
    }

    function issueTickets(address recipient, uint256 quantity) public onlyOwner
        require(quantity > 0, "Quantity must be greater than zero");
        totalTickets += quantity;
        ticketBalances[recipient] += quantity;
        emit TicketIssued(recipient, quantity);
    }

    function getMyTicketBalance() public view returns (uint256) {   2548 gas
        return ticketBalances[msg.sender];
    }
```

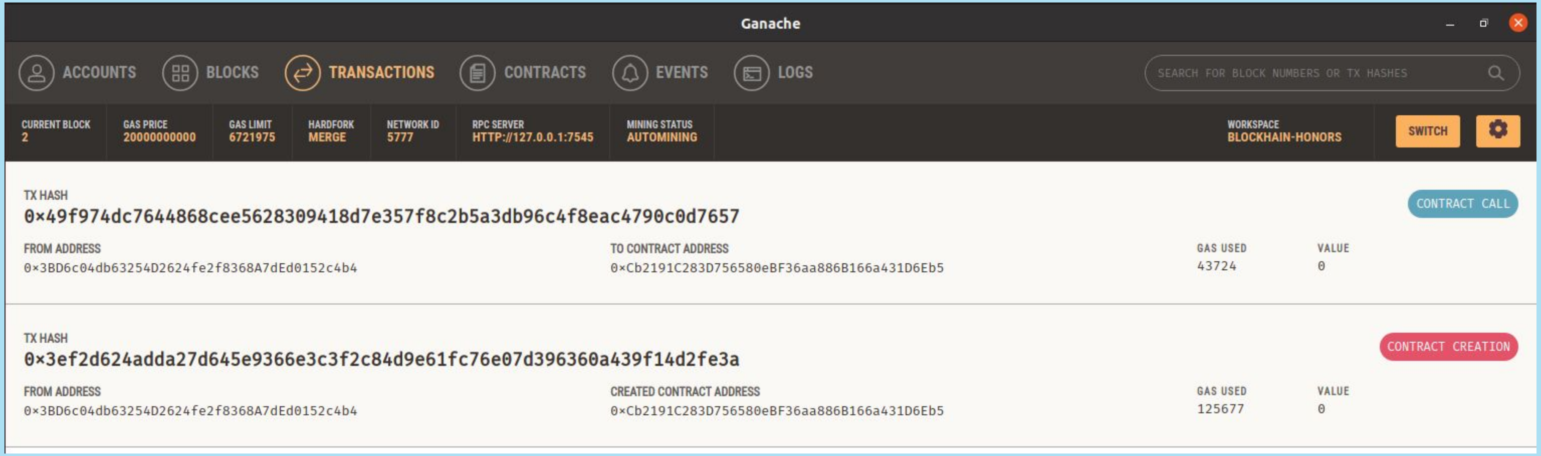
```
[block:4 txIndex:0] from: 0x654...2b6cd to: TicketManagement.issueTickets(address,uint256) 0xAf8...92926 value: 0 wei
data: 0x85b...00005 logs: 1 hash: 0xc36...ffdad
```

Experiment 8: Deploying with Ganache Account



CURRENT BLOCK	GAS PRICE	GAS LIMIT	HARDFORK	NETWORK ID	RPC SERVER	MINING STATUS	WORKSPACE
2	20000000000	6721975	MERGE	5777	HTTP://127.0.0.1:7545	AUTOMINING	BLOCKHAIN-HONORS

BLOCK	MINED ON	GAS USED	
2	2023-09-25 18:45:38	43724	1 TRANSACTION
1	2023-09-25 18:36:24	125677	1 TRANSACTION
0	2023-09-25 17:28:39	0	NO TRANSACTIONS



CURRENT BLOCK	GAS PRICE	GAS LIMIT	HARDFORK	NETWORK ID	RPC SERVER	MINING STATUS	WORKSPACE
2	20000000000	6721975	MERGE	5777	HTTP://127.0.0.1:7545	AUTOMINING	BLOCKHAIN-HONORS

TX HASH
0×49f974dc7644868cee5628309418d7e357f8c2b5a3db96c4f8eac4790c0d7657
CONTRACT CALL

FROM ADDRESS	TO CONTRACT ADDRESS	GAS USED	VALUE
0×3BD6c04db63254D2624fe2f8368A7dEd0152c4b4	0×Cb2191C283D756580eBF36aa886B166a431D6Eb5	43724	0

TX HASH
0×3ef2d624adda27d645e9366e3c3f2c84d9e61fc76e07d396360a439f14d2fe3a
CONTRACT CREATION

FROM ADDRESS	CREATED CONTRACT ADDRESS	GAS USED	VALUE
0×3BD6c04db63254D2624fe2f8368A7dEd0152c4b4	0×Cb2191C283D756580eBF36aa886B166a431D6Eb5	125677	0