ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN

LAB 3:

Solidity Smart Contract Development

Sinh viên thực hiện

19120659 – Phạm Văn Thành

20120356 – Lê Minh Quân

20120382 – Hoàng Thu Thủy

20120386 – Lê Phước Toàn

20120389 – Nguyễn Thị Bích Trâm

Giảng viên hướng dẫn:

Thầy Nguyễn Đình Thúc

Thầy Ngô Đình Hy

Bộ môn: Blockchain và Ứng dụng

Thành phố Hồ Chí Minh, ngày 26 tháng 12, 2023

THÔNG TIN NHÓM

Nhóm 3			
Họ và tên	**MSSV**	**Phân công**	**Đánh giá**
Phạm Văn Thành	19120659	Deploy smart contract	100%
Lê Minh Quân	20120356	Lập trình smart contract	100%
Hoàng Thu Thủy	20120382	Báo cáo	100%
Lê Phước Toàn	20120386	Tương tác với smart contract	100%
Nguyễn Thị Bích Trâm	20120389	Test smart contract	100%

MỤC LỤC

NỘI DUNG

1. Smart Contracts

1.1 Tổng quan

Hợp đồng thông minh (smart contract) là một thuật ngữ mô tả bộ giao thức đặc biệt có khả năng tự động thực hiện các điều khoản hay thỏa thuận giữa các bên (hệ thống máy tính) nhờ vào công nghệ blockchain. Các điều khoản được quy định trong hợp đồng thông minh tương đương với hợp đồng pháp lý truyền thống.

Nó sẽ tự động hoá hợp đồng mà không cần sự can thiệp từ bên ngoài. Điều này giúp người dùng tiện lợi và đảm bảo tính chính xác, minh bạch rất cao vì không có sự can thiệp lẫn đảo chiều. Ngoài ra nó cũng dễ dàng truy xuất khi cần thiết.

Một số đặc điểm nổi bật của smart contract bao gồm:

- Tự thực thi: Smart contract tự động thực thi mà không cần sự can thiệp của bên thứ ba. Khi điều kiện được đáp ứng, chúng thực hiện các hành động được xác định trước.
- Không phụ thuộc vào bên thứ ba: Smart contract hoạt động độc lập và không phụ thuộc vào bất kỳ bên thứ ba nào như ngân hàng, luật sư hoặc tổ chức trung gian khác. Điều này giúp giảm thiểu chi phí và thời gian giao dịch.
- Không thể xóa hoặc thay đổi: Một khi smart contract được triển khai, chúng không thể bị xóa và thay đổi mà không cần sự đồng thuận của tất cả các bên tham gia. Điều này đảm bảo tính vẹn toàn của giao dịch giữa các bên.
- Tính minh bạch: Hợp đồng thông minh luôn được lưu trữ trên blockchain, do đó tất cả mọi người đều có thể kiểm tra thông tin trên hợp đồng, bao gồm điều khoản, thời gian thực hiện…

1.2 Cơ chế hoạt động

B1: Người dùng khởi tạo Transaction từ ví blockchain của họ

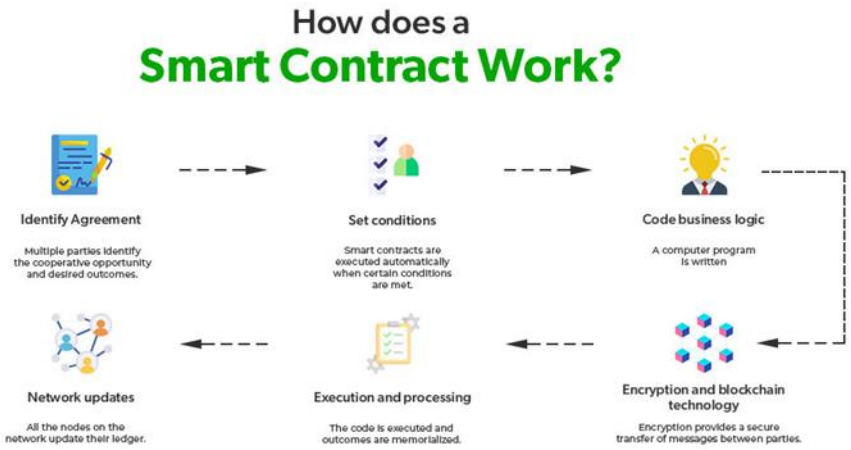
B2: Transaction được gửi tới cơ sở dữ liệu phân tán (Distributed database) , và được xác nhận danh tính

B3 Transactions được chấp thuận

B4: Giao dịch bao gồm mã code xác định loại giao dịch nào sẽ được thực thi

B5: Các transactions được thêm dưới dạng một khối (Block) trong chuỗi khối.

B6: Mọi thay đổi về trạng thái hợp đồng đều tuân theo quy trình tương tự để được cập nhật.



Hình 1-15: Cách thức hoạt động của smart contract

1.3 Ưu và nhược điểm của SmartContract

Ưu điểm:

- *Dữ liệu smart contract không thể thay đổi*: Hợp đồng thông minh không thể bị thay đổi và xâm nhập từ bên ngoài, khiến cam kết giữa nhiều bên luôn được đảm bảo toàn. Tuy nhiên, nếu như có lỗi trong smart contract từ đầu, nhà phát triển không thể sửa lỗi mà chỉ có thể phát triển một smart contract mới.
- *Phi tập trung và tự thực hiện (self executing)*: Hợp đồng thông minh không phụ thuộc vào bên thứ ba để xác thực các điều khoản trên hợp đồng, từ đó giảm thiểu chi phí của mạng lưới. Ngoài ra, smart contract có khả năng tự thực hiện, khi điều kiện được đáp ứng mà không cần sự can thiệp từ bên thứ ba như luật sư, ngân hàng…
- Mặc dù smart contract được sử dụng chủ yếu ở thị trường crypto, tuy nhiên đã có một số doanh nghiệp thực hiện nghiên cứu và triển khai áp dụng trong đời sống, nhằm tận dụng những ưu điểm như giảm thiểu chi phí, tăng hiệu suất công việc.

Nhược điểm:

- Dữ liệu không thể thay đổi khi phát sinh lỗi: Ưu điểm của smart contract hiện cũng đang là nhược điểm, một khi smart contract được triển khai, chúng sẽ không thể bị sửa đổi và can thiệp. Vì vậy, nếu smart contract xảy ra lỗi, nhà phát triển hoặc người tham gia chỉ có thể tạo một hợp đồng mới.

Ví dụ: Năm 2016, một tổ chức có tên là "The DAO" bị tấn công do có sai sót trong smart contract, khiến họ thiệt hại hàng triệu ETH. Khi đó, smart contract của The DAO không thể thay đổi, nên họ không thể sửa đổi code trong smart contract. Điều này cuối cùng đã dẫn đến một cuộc hard fork, tạo ra Ethereum Classic và Ethereum.

- Khó giải quyết tranh chấp: Hợp đồng thông minh không có cơ chế giải quyết tranh chấp khi có xảy ra những tình huống ngoài dự kiến hoặc không rõ ràng. Các bên tham gia có thể phải tìm đến các tổ chức hoặc cá nhân khác để giải quyết, nhưng điều này có thể mất nhiều thời gian và chi phí.
- Phụ thuộc vào công nghệ blockchain: Hợp đồng thông minh hoạt động dựa trên công nghệ blockchain, do đó nó cũng phải chịu những hạn chế của công nghệ này, ví dụ như: khả năng mở rộng, hiệu suất, tiêu thụ năng lượng, v.v.
- Chưa có bảo hộ pháp lý: Smart contract là sản phẩm chưa có những quy định và pháp lý rõ ràng. Vì vậy, nếu smart contract có lỗi xảy ra, người dùng cũng không được chính phủ bảo vệ quyền lợi. Theo Certik, các lỗi của smart contract có thể dẫn tới những cuộc tấn công như rug pull, exploit... Và những vụ tấn công này đã gây thiệt hại lên tới 3.7 tỷ USD trong năm 2022 (theo báo cáo của CertiK).

1.4 Ứng dụng của SmartContract

- Tài chính: Hợp đồng thông minh được sử dụng để thực hiện các giao dịch thanh toán, gửi tiết kiệm, vay mượn, đầu tư, bảo hiểm, v.v. một cách nhanh chóng, an toàn và tiết kiệm chi phí.
- Bất động sản: Hợp đồng thông minh thực hiện các giao dịch mua bán, thuê, cho thuê, quản lý, v.v. bất động sản một cách minh bạch, tự động và không cần đến các bên trung gian.
- Y tế: Hợp đồng thông minh có thể được sử dụng để quản lý và chia sẻ dữ liệu y tế của bệnh nhân, cung cấp các dịch vụ y tế từ xa, thanh toán các chi phí y tế, v.v. một cách an toàn và hiệu quả.
- Giáo dục: Hợp đồng thông minh có thể được sử dụng để cấp và xác nhận các bằng cấp, chứng chỉ, điểm số, v.v. của học sinh, sinh viên, giáo viên, v.v. một cách công bằng và không thể bị gian lận.
- Nhiều lĩnh vực khác như: năng lượng, giao thông, du lịch, nông nghiệp, v.v.

2. Decentralized Voting System

2.1 Mục đích áp dụng smart contract vào hệ thống

Mục đích của smart contract trong Decentralized Voting System là tạo ra một hệ thống bỏ phiếu phi tập trung, công khai và bảo mật trên nền tảng Ethereum. Smart contract cho phép người dùng bỏ phiếu cho các ứng cử viên và mỗi địa chỉ trên blockchain chỉ được bỏ phiếu một lần duy nhất. Người dùng có thể xem kết quả bỏ phiếu bằng cách truy vấn danh sách các ứng cử viên hoặc thông tin của một ứng cử viên bất kỳ. Sau khi thực hiện bỏ phiếu bầu, hành động đó sẽ không thể hoàn lại và do đó kết quả bỏ phiếu sẽ không thể bị thao túng và gian lận.

Smart contract tương tác với Ethereum blockchain (bao gồm các tài khoản và các smart contract khác) bằng cách công khai những phương thức để gọi thực thi. Danh sách các phương thức bao gồm:

- candidateCount: là getter của biến candidateCount cho biết số lượng ứng cử viên ở trong smart contract.
- getAllCandidates: có kết quả về là một mảng chứa tên các ứng cử viên và một mảng chứa số lượng phiếu bầu của các ứng cử viên đó.
- getCandidate: lấy thông tin (ID, tên và số lượng phiếu bầu) của một ứng cử viên bất kỳ.
- voterLookup: là getter của mapping voterLookup giúp cho biết một địa chỉ bất kỳ ở trên blockchain đã thực hiện bầu cử hay chưa.
- vote: phương thức chính dùng để bầu cử. Khi bầu cử, người dùng cần gửi kèm theo 1000 gwei (tương ứng với 0.000001 ether).

Smart contract giải quyết vấn đề của các hệ thống bỏ phiếu truyền thống, như sự thiếu minh bạch, gian lận, chi phí cao và khó kiểm soát.

2.2 Khó khăn và bài học kinh nghiệm rút ra được khi làm việc với smart contract

Khó khăn khi làm việc với smart contract, đặc biệt là smart contract sử dụng ngôn ngữ Solidity chính là việc cú pháp thay đổi nhanh. Cụ thể hơn, khi xem các bài hướng dẫn cũ, một số cách viết smart contract có thể không còn hoạt động với các phiên bản compiler mới hơn. Điều này gây ra sự bất tiện không ít khi viết smart contract.

Ngoài ra, do không thể biên dịch thành file thực thi và chạy ngay lập tức nên việc kiểm tra kết quả thực thi của smart contract có phần khó khăn hơn lập trình truyền thống.

Đặc biệt, do liên quan đến tương tác với blockchain, nếu như không có sự hỗ trợ của các framework phát triển smart contract chẳng hạn như Truffle hoặc Hardhat thì việc kiểm thử và deploy sẽ rất khó khăn. Cụ thể hơn, nếu như không dùng các framework thì lập trình viên phải tự chạy một blockchain node để tạo ra một giao dịch phục vụ cho việc kiểm thử và deploy smart contract.

2.3 Các tính năng dự định sẽ phát triển cho smart contract trong tương lai

Trong tương lai, nhóm dự định sẽ thêm vào những tính năng sau cho smart contract:

- Cho phép người dùng đăng ký làm ứng cử viên.
- Tạo ra một ERC-20 token đại diện cho phiếu bầu và ERC-721 token đại diện cho việc bầu cử của người dùng. Cụ thể hơn, người bầu cử sẽ sở hữu một ERC-721 token cho biết họ đã bầu cử. Với token này, họ sẽ được hưởng được nhiều quyền lợi thực khác.
- Cho phép tạo ra các cuộc bầu cử khác và tự động đóng cuộc bầu cử theo một thời gian nhất định nào đó.

Về khía cạnh hệ thống:

- Deploy một phần ứng dụng lên IPFS để đảm bảo tính minh bạch và tăng tốc độ truy cập tài nguyên.
- Sử dụng các Oracle để tích hợp thêm các real world API.
- Sử dụng TheGraph để đánh index cho các event xảy ra trong smart contract và xây dựng API để truy vấn thông tin từ các event đó.

Về khía cạnh bảo mật:

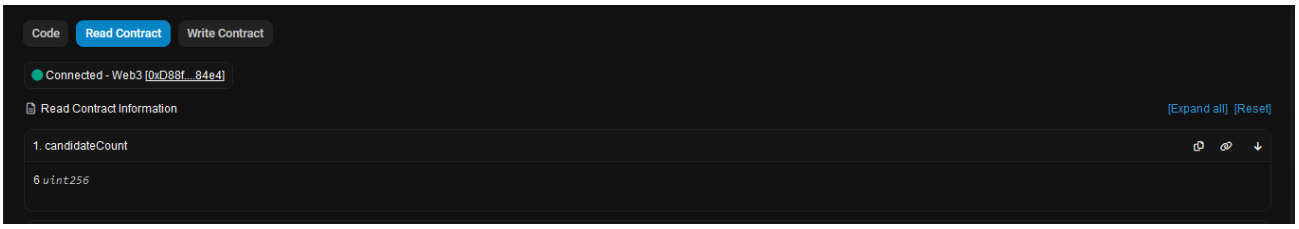
- Sử dụng contract Ownable để đảm bảo access control cho các phương thức quan trọng.

2.4 Mô tả và demo smart contract

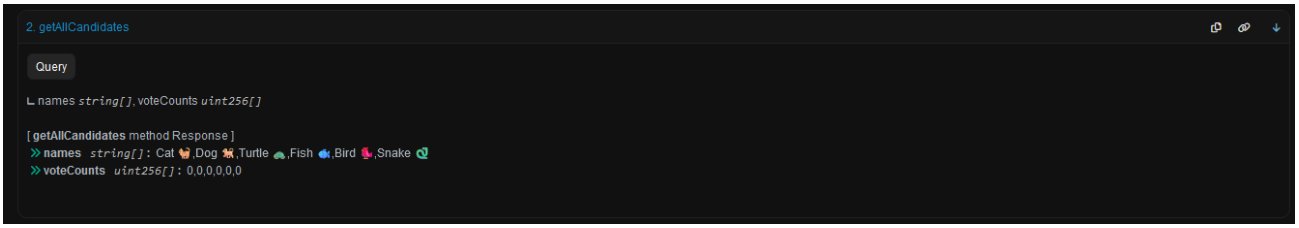
- Người dùng sẽ được bầu chọn các ứng cử viên từ danh sách sẵn có, mỗi người chỉ được vote một lần, và sau khi vote sẽ không thay đổi được quyết định của mình. Để vote thì người dùng cần phải kết nối với ví Metamask (được cài dưới dạng Extension của trình duyệt) để lần vote đó được xác thực và đảm bảo bởi hệ thống Blockchain. Hệ thống này sẽ sử dụng Ethereum để xác thực các lượt vote và dùng Solidity để tạo ra smart contract áp dụng vào hệ thống.

- Demo:

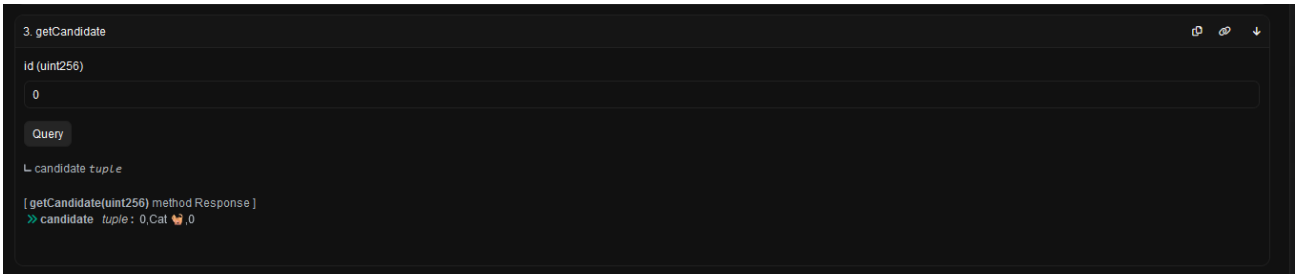
Số lượng ứng cử viên:



Danh sách các ứng cử viên:

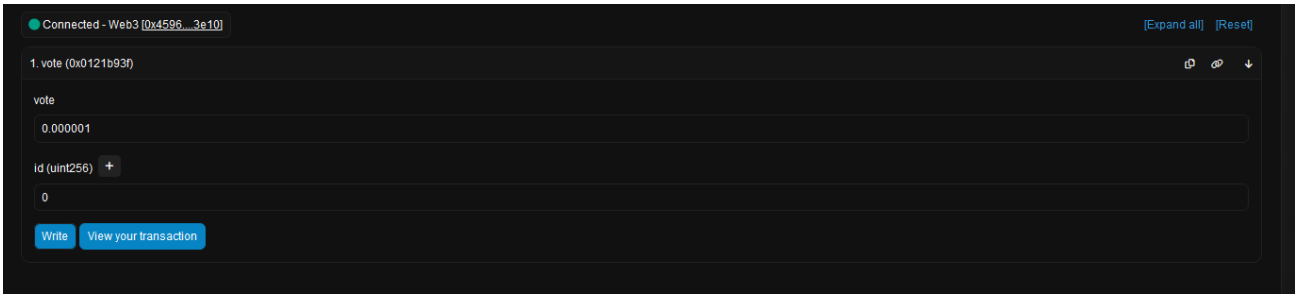


Truy vấn thông tin của một ứng cử viên bất kỳ bằng ID:

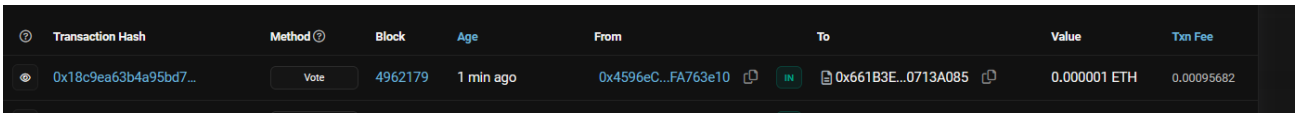


Nếu ID vượt quá số lượng ứng cử viên thì sẽ báo lỗi:

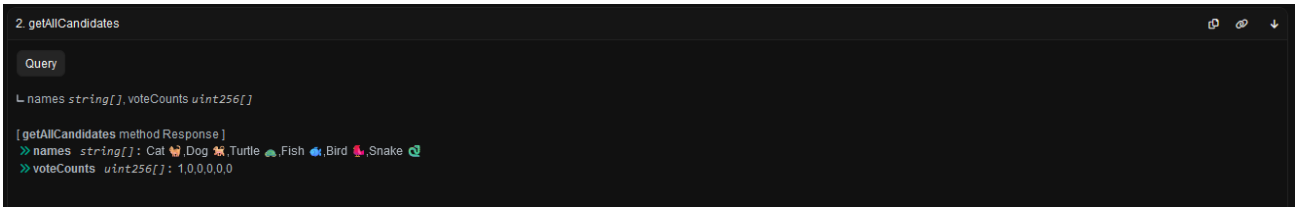
Vote cho ứng cử viên 0:



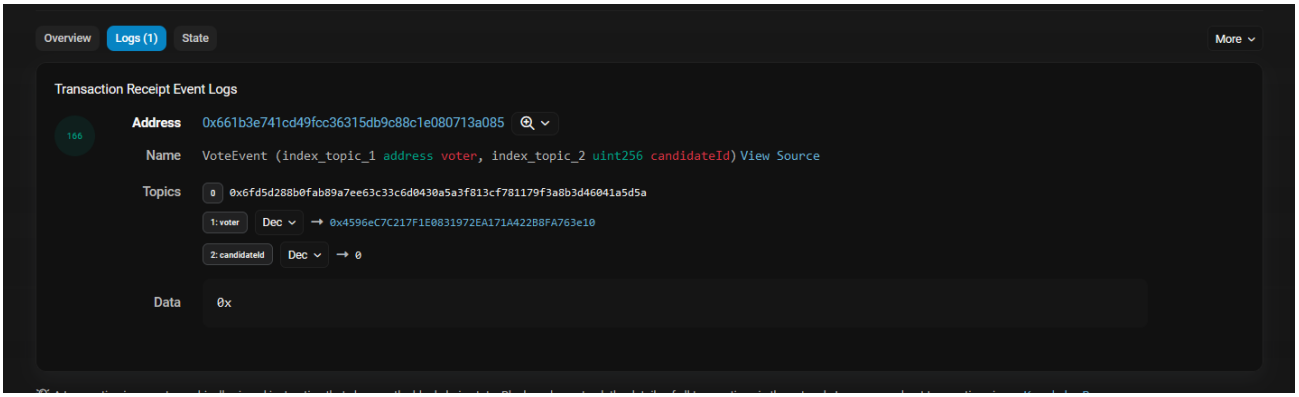
Sau khi vote thì xuất hiện một giao dịch như sau:



Kiểm tra lại danh sách các ứng cử viên thì thấy số phiếu bầu của ứng cử viên có ID là 0 đã tăng lên 1:



Transaction này cũng sinh ra một event như sau:



Link smart contract:

https://sepolia.etherscan.io/address/0x661b3e741cd49fcc36315db9c88c1e080713a085

TÀI LIỆU THAM KHẢO

https://www.iberdrola.com/innovation/smart-contracts

https://ethereum.org/en/learn/

https://trufflesuite.com/docs/truffle/

https://hardhat.org/tutorial

https://docs.metamask.io/wallet/