

스마트 컨트랙트 및 프라이빗 블록체인을 활용한 커피 물류 관리 시스템 구현



개요

커피는 더 이상 단순한 기호 식품이 아니라 일상의 필수적인 부분이 되어 우리 문화에 깊게 자리 잡고 있다. 현대인들에게는 커피를 마시는 것은 하루를 시작하는 일상이 되었으며, 식사 후에 졸음을 이기고 두뇌를 깨우는 데 필요한 것으로 인식되어 있다. 또한 커피를 마시면서 업무 혹은 학업을 하고, 사람들 간에 대화하는 등 습관처럼 일상에서 소비되고 있다.

이처럼, 커피는 한국뿐만 아니라 전 세계에서 상당히 큰 비중을 차지하고 있는 산업이다. 그러나 현재 커피 물류 시스템에서는 장시간 노동에 대한 적절한 임금을 받지 못하는 농부가 존재하며, 선호하는 원두를 소비하지 못하고 낮은 품질의 원두를 소비하는 소비자가 종종 존재한다. 이에 스마트 컨트랙트와 프라이빗 블록체인을 활용하여, 보안과 신뢰성, 투명성 그리고 효율성까지 확보하고자 한다. 또한 현재 택배 및 배송 업체들이 겪고 있는 원산지 확인, 물류 추적 등과 같은 문제를 해결하는 시스템을 구현하고자 한다.

목차

1. 연구 배경

- a) 배경
- b) 목표

2. 요구조건 및 제약 사항 분석에 대한 수정사항

- a) 프라이빗 네트워크 구축 문제

3. 설계 상세화 및 변경 내역

- a) 프라이빗 네트워크 문제로 인한 Hyperledger Besu 사용
- b) React 대신 Vue 사용

4. 과제 진척도

- a) 진척도
- b) 중간 결과
- c) 기대 효과

5. 참고문헌

1. 연구 배경

a) 배경

한국에서 커피 소비는 최근 몇 년 동안 상당한 증가세를 보이는 중이다. 유로모터스에 따르면 **2023년** 국내 1인당 연간 커피 소비량은 **405잔**으로 전 세계 1인당 연간 커피 소비량 **152잔** 대비 두 배 이상 높았다. 식품의약품안전처에 따르면, **2022년** 음료류 품목별 국내 판매액 중 전체 음료 시장에서 커피류가 차지하는 비중도 **30.8%**로 탄산음료 **25.5%**보다 높다. **2023년** 전 세계 커피 시장 규모는 **1,131억 5,260만 달러**로 전년 대비 **9.0%** 증가했고, **2027년까지** 연평균 **4.3%** 성장해 **1,331억 5,750만 달러**로 확대될 전망이다. 이처럼 전 세계적 커피 시장의 흐름은 확대될 것으로 예측된다.

커피 원두는 일반적으로 원산지를 기반으로 마케팅되며, 각 지역은 고유한 맛과 특성을 가진 커피를 생산하는 것으로 알려져 있다. 그러나 현재까지는 수작업을 통하여 원산지를 추적했기 때문에, 품질이 낮은 지역의 원두를 유명한 커피 재배 지역의 프리미엄 원두로 라벨링 되어 높은 가격을 책정하는 등 신뢰성이 부족하였다.

또한 농부에서부터 수출입업자, 도소매 업체 등 다분화된 공급망은 신뢰성 있는 원산지와 품질을 검증하는 것이 어려워지는 것에 동조한다. 그리고 커피의 최종 소비자뿐만 아니라 농부들 역시 투명성 및 시장 정보 부족으로 공정한 가격을 받지 못하여, 커피 농부들의 생계뿐만 아니라 커피 공급망의 전반에 품질과 지속 가능성에도 영향을 미친다.

b) 목표

이에 금융권에서 주로 사용하던 블록체인 기술을 커피 산업에서 활용하는 것은 커피 산업에 많은 이점을 줄 수 있다.

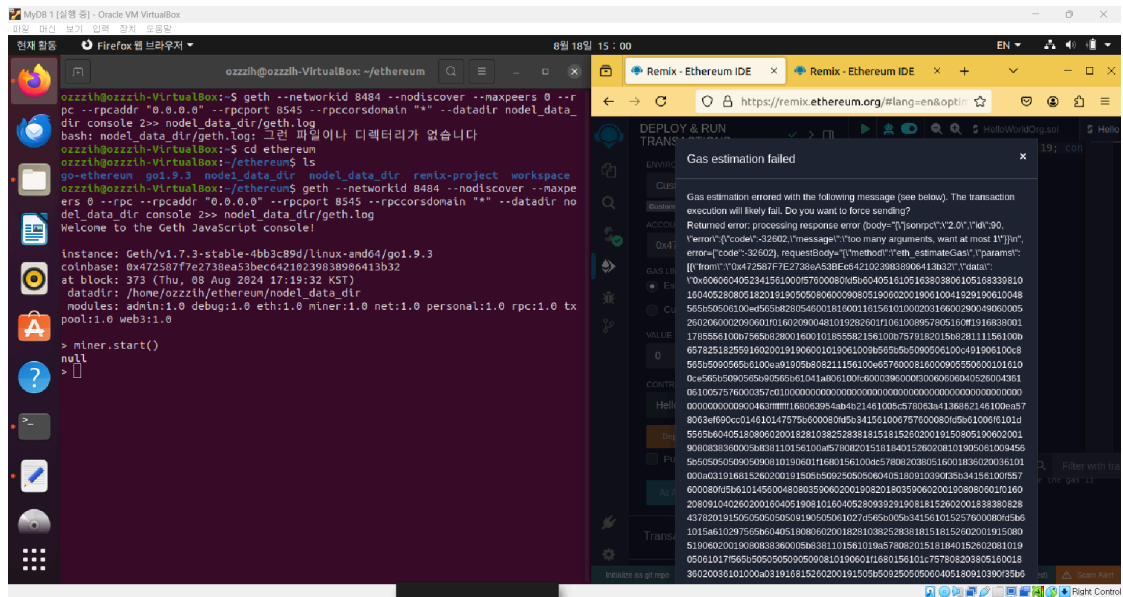
커피 공급망의 투명성과 추적 가능성을 개선하여 농부로부터 소매업체에 이르는 각 단계의 이해관계자가 커피 원두의 물류 과정을 정확히 파악함으로써, 도소매 업체가 원두의 품질에 맞는 합리적인 가격에 유통하도록 유도할 수 있다. 또한 최종 커피 소비자는 본인이 선호하는 원산지의 원두를 소비할 수 있으며, 가격에 상응하는 품질의 커피를 소비할 수 있게 유도할 수 있다.

본 연구는 스마트 컨트랙트 및 프라이빗 블록체인을 활용하여 복잡하고 불투명한 현재 물류 과정을 획일화하여, 투명성과 신뢰성을 부여하고자 한다. 추가로, 이 시스템을 통해, 실시간 물류 추적 및 오배송의 최소화를 목표로 한다.

2. 요구조건 및 제약 사항 분석에 대한 수정사항

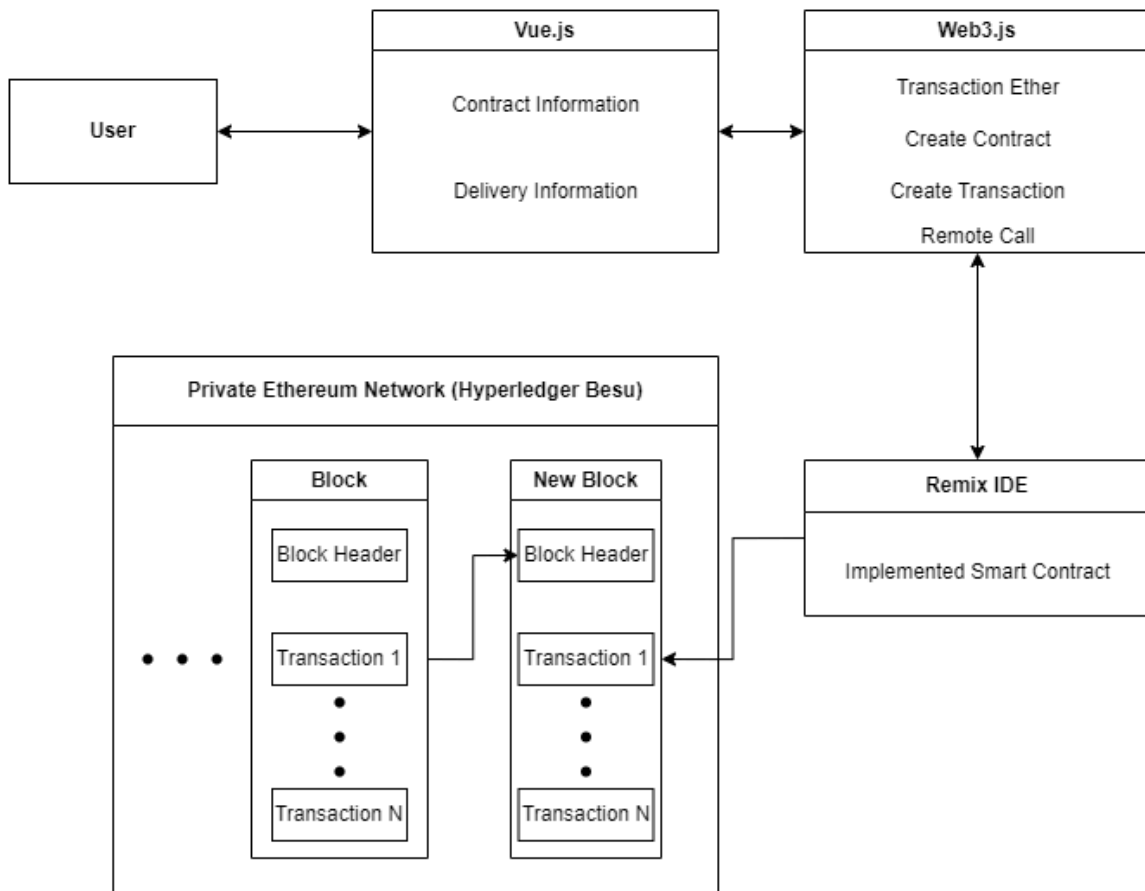
a) 프라이빗 네트워크 구축 문제

본 프로젝트의 초기 단계에서 **Geth**를 사용하여 프라이빗 이더리움 네트워크를 성공적으로 구축했다. 그러나 **Geth**의 특정 버전과 **Remix IDE** 간의 호환성 문제로 인해 **Solidity** 스마트 컨트랙트를 배포하는 과정에서 예기치 않은 장애가 발생했다. 구체적으로, **Geth**의 최신 버전에서 발생한 **API** 변경 사항 또는 **Remix IDE**의 업데이트된 요구 사항으로 인해 스마트 컨트랙트가 정상적으로 배포되지 않았다.



이에 따라, 우선적으로 테스트넷에서 스마트 컨트랙트를 테스트한 후, Hyperledger Besu를 활용하여 프라이빗 네트워크를 재구축하기로 하였다. 이러한 방식으로 문제를 해결하고, 향후 Hyperledger Besu를 통해 보다 안정적인 프라이빗 네트워크 환경을 구현하고 배포 할 계획이다.

3. 설계 상세화 및 변경 내역



a) 프라이빗 네트워크 문제로 인한 **Hyperledger Besu** 사용

초기 설계 단계에서는 **Geth**를 사용하여 프라이빗 이더리움 네트워크를 구축한다. 그러나 **Geth**와 관련된 버전 호환성 문제로 인해, 보다 안정적이고 효율적인 네트워크 구축을 위해 **Hyperledger Besu**로 전환하였다.

b) **React** 대신 **Vue** 사용

프론트엔드 개발 환경에서 **React**를 사용할 계획이었으나, 프로젝트 요구사항과 팀의 기술 스택을 고려하여 **Vue.js**로 전환한다. **Vue.js**는 **React**에 비해 상대적으로 가벼운 학습 곡선을 제공하며, 프로젝트의 특성상 빠르게 개발을 진행할 수 있는 장점을 가진다. 이에 따라, 설계에서 **Vue.js**를 사용하여 사용자 인터페이스를 구현하였다.

4. 과제 진척도

a) 진척도

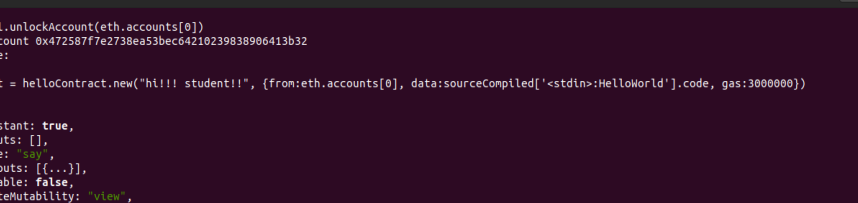
개발 구분	5					6				7				8					9			
	1	2	3	4	5	1	2	3	4	1	2	3	4	1	2	3	4	5	1	2	3	4
사전 자료 조사																						
블록체인 스터디																						
개발 프레임워크 스터디																						
블록체인 네트워크 구축																						
체인코드 개발																						
웹 UI 및 개발																						
테스트 및 수정																						
최종 점검 및 발표 준비																						

학번	이름	구성원별 진척상황
201924586	조주영	Vue.js로 플랫폼 개발 로그인 페이지, 메인 페이지, 원두 선택 페이지, 배송정보 페이지 등 기본적인 웹페이지 구성 완료 이더리움으로 결제 / 실시간으로 물류 배송 조회 기능 구현 중
201924436	김영목	Geth로 가상 네트워크 구축 채굴을 통한 스마트 컨트랙트 배포 커피 구매 및 물류 이동 컨트랙트 코드 개발 중
201941171	장영철	Hyperledger Besu로 가상 네트워크 구축 테스트넷 환경에서 스마트 컨트랙트 테스트 커피 구매 및 물류 이동 컨트랙트 코드 개발 중

b) 중간 결과

[illegible]

(1) Geth로 프라이빗 네트워크 생성 및 컨트랙트 소스 저장.



```
> personal_unlockAccount(eth.accounts[0])
Unlock account 0x4725877fe2738ea53bec64210239838906413b32
Passphrase:
true
> contract = helloContract.new("hi!!! student!!!", {from:eth.accounts[0], data:sourceCompiled['<stdin>:HelloWorld'].code, gas:3000000})
{
  abi: [{
    constant: true,
    inputs: [],
    name: "say",
    outputs: [{...}],
    payable: false,
    stateMutability: "view",
    type: "function"
  }, {
    constant: false,
    inputs: [{...}],
    name: "setGreeting",
    outputs: [],
    payable: false,
    stateMutability: "nonpayable",
    type: "function"
  }, {
    constant: true,
    inputs: [],
    name: "greeting",
    outputs: [{...}],
    payable: false,
    stateMutability: "view",
    type: "function"
  }, {
    inputs: [{...}],
    payable: false,
    stateMutability: "nonpayable",
    type: "constructor"
  }
  ],
  source: undefined
}
```

(2) Solidity로 작성된 컨트랙트를 컴파일하고, Geth 콘솔을 통해 배포.

```
> miner.start()
true
> eth.pendingTransactions
[]
> miner.stop()
true
> contract.say()
"hello student!!"
```

(3) 마이닝을 통해 트랜잭션을 처리하고 이더리움 생성.

- 위 사진은 **Geth** 환경에서 가상 네트워크를 설정하고 스마트 컨트랙트를 배포를 시도해 보았을때 사진이다.

- POW방식은 현재 최신 버전 **geth**에선 지원하지 않아 이전 버전 **geth**를 이용해야 하는데 이때 이전 버전 **geth**가 최신 **REMIX IDE**와 연동되지 않는 단점이 존재하였다. 이에 **Hyperledger Besu**를 통해 프라이빗 네트워크 구축하고 **Remix IDE**와 연동을 시도해보고자 한다.

```

jyc@BOOK-21K5BF6MCD:~$ docker run -d --name besu-node --user root -v $(pwd)/data:/besu/data -v $(pwd)/genesis.json:/besu/genesis.json hyperledger/besu:latest --genesis-file=/besu/genesis.json --network-id=1337 --data-path=/besu/data --rpc-http-enabled --rpc-http-api=ETH,NET,WEB3 --host-allowlist="*" --rpc-http-cors-origins="all" --rpc-http-port=8545
90bd8911dd35
jyc@BOOK-21K5BF6MCD:~$ docker ps
CONTAINER ID   IMAGE               COMMAND                  CREATED        STATUS        PORTS                               NAMES
90bd8911dd35   hyperledger/besu:latest   "besu --genesis-file="   6 seconds ago   Up 5 seconds (health: starting)   8545-8547/tcp, 8550-8551/tcp, 30303/tcp   besu-node

```

(1) 노드 실행

```

jyc@BOOK-21K5BF6MCD:~$ docker logs besu-node
2024-08-18 08:33:26.258+00:00 | main | INFO | Besu | Starting Besu
2024-08-18 08:33:26.727+00:00 | main | WARN | Besu | --rpc-ws-host has been ignored because --rpc-ws-enabled was not defined on the command line.
2024-08-18 08:33:26.737+00:00 | main | WARN | Besu | --graphql-http-host has been ignored because --graphql-http-enabled was not defined on the command line.
2024-08-18 08:33:26.877+00:00 | main | WARN | Besu | Forcing --bonsai-limit-trie-logs-enabled=false, since it cannot be enabled with --sync-mode=FULL and --data-storage-format=BONSAI.
2024-08-18 08:33:26.889+00:00 | main | INFO | Besu | Connecting to 0 static nodes.
2024-08-18 08:33:27.047+00:00 | main | INFO | Besu |
#
# Besu version 24.7.1
#
# Configuration:
# Network: Custom genesis file
# /besu/genesis.json
# Network Id: 1337
# Data storage: Bonsai
# Sync mode: Full
# RPC HTTP APIs: ETH,NET,WEB3
# RPC HTTP port: 8545
# Using LAYERED transaction pool implementation
# Using STACKED worldstate update mode
#
# Host:
# Java: openjdk-java-21
# Maximum heap size: 1.89 GB
# OS: linux-x86_64
# glibc: 2.39
# jemalloc: 5.3.0-g54eae1d8b56b1aa528be3bdd1877e59c56fa90c
# Total memory: 7.54 GB
# CPU cores: 16
#
# Plugin Registration Summary:
# No plugins have been registered.
# TOTAL = 0 of 0 plugins successfully registered.
#
#####

```

(2) Besu 로그 확인

The screenshot shows the Docker Desktop interface. On the left, the 'Containers' tab is selected. The main area shows a table of containers. One container, 'besu-node', is listed with the image 'hyperledger/besu:latest' and is in an 'Exited (2)' state. The 'Actions' column for this container shows a play button, a vertical ellipsis, and a trash icon. A message at the bottom right indicates 'Selected 1 of 1'.

Name	Image	Status	Port(s)	CPU (%)	Last state	Actions
besu-node	hyperledger/besu:latest	Exited (2)	30303:30303	N/A	7 seconds ago	Play, More options, Delete

- (3) Action이 바로 중지되는 문제가 발생하여 해결 과정에 있다.

c) 기대 효과

1. 투명한 거래 및 신뢰성 향상

블록체인 기술을 활용함으로써, 커피 구매 및 배송 과정에서 발생하는 모든 거래 기록을 투명하게 관리할 수 있다. 각 거래는 블록체인에 안전하게 기록되며, 변경 불가능한 특성을 지니므로 고객과 판매자 모두에게 신뢰성을 제공한다. 이는 고객이 제품을 구매할 때의 신뢰도를 높이고, 판매자는 정확한 거래 기록을 유지할 수 있게 한다.

2. 효율적인 배송 관리

Hyperledger Besu 프라이빗 네트워크를 통해 관리되는 배송 정보는 중앙 서버에 의존하지 않고, 블록체인 상에서 안전하게 저장 및 조회할 수 있다. 이를 통해 배송 과정의 효율성을 극대화하고, 데이터의 무결성을 보장할 수 있다.

3. 맞춤형 포인트 시스템 구축

테스트넷에서 조회된 ETH 잔액을 기반으로 맞춤형 포인트 시스템을 구현한다. 고객은 자신의 ETH 잔액을 손쉽게 확인할 수 있으며, 이를 통해 포인트를 사용하거나 적립할 수 있는 기능을 제공한다. 이로써 고객 충성도를 높이고, 반복 구매를 유도할 수 있다.

4. 기술적 유연성 및 확장성

Hyperledger Besu와 Vue.js를 결합함으로써, 프로젝트는 확장 가능한 구조를 갖추게 된다. 프라이빗 네트워크의 유연성을 활용해 다양한 비즈니스 요구사항을 수용할 수 있으며, Vue.js의 모듈화된 구조는 프론트엔드 기능 확장을 용이하게 한다. 이를 통해 향후 새로운 기능 추가 및 시스템 확장이 원활하게 이루어질 수 있다.

5. 참고문헌

- [초점] ‘대한커피민국’ 1인당 연간 405잔 마신다...세계 평균 2.5배
<http://www.newsian.co.kr/news/articleView.html?idxno=66185>
- GOPAX ACADEMY. “퍼블릭 블록체인과 프라이빗 블록체인이 무엇인가요?”
- BTCC. “스마트 컨트랙트란 무엇입니까?”
- 해시넷. “프라이빗 블록체인”
- 이은주, and 김진욱. "이더리움 기반 공공정보 소프트웨어 사업산출물관리 시스템 설계 및 구현." 정보처리학회논문지. 컴퓨터 및 통신시스템 11.6 (2022): 177.