

2022 Block Chain Project

BloCar

박수권 김민상 박준규

oAo

CONTENTS

01 개요

- BloCar 소개
- 기존 중고차 거래 문제 현황
- 개선점 파악

02 사업 계획

- 블록체인 접목 방식 설계

03 제작 구조

- Used

04 결과 보고

- Risk & Issues

01

개요

시장조사

: 기존 중고차 시장 문제점 파악

BloCar 블록체인 도입

시장 조사

□ 성능상태 점검내용과 실제 차량상태가 다른 소비자피해가 79.7%

중고차 성능·상태점검기록부, 실제 차량과 달라 피해 빈발

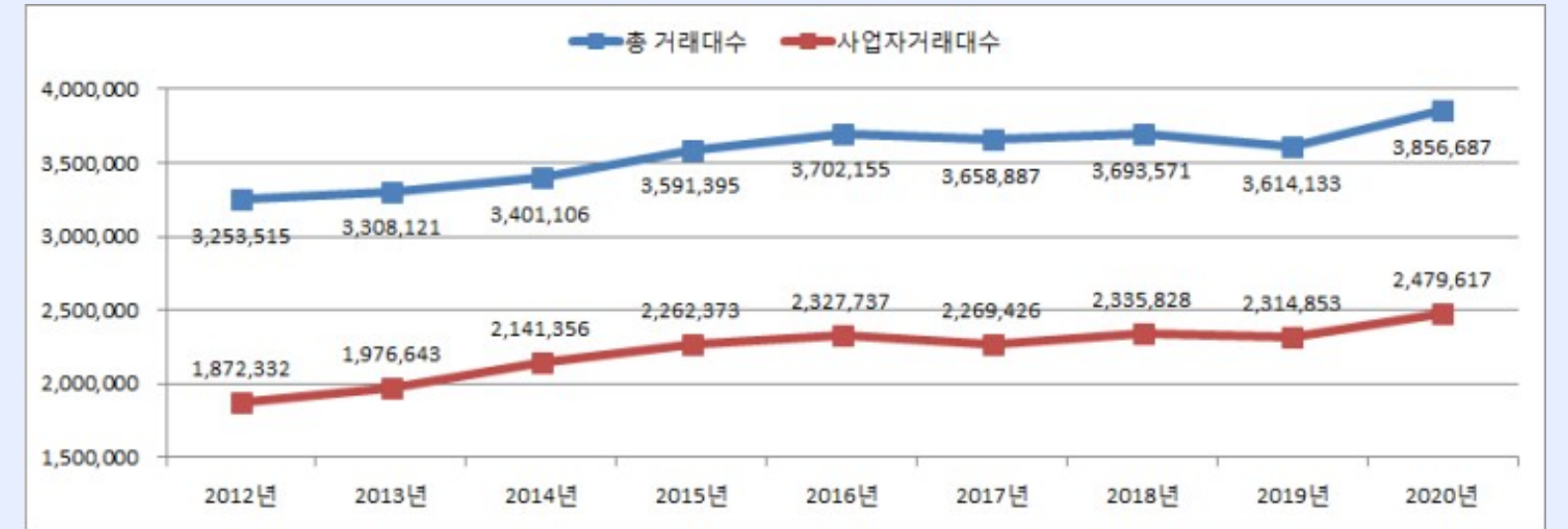
[연도별 지역별 피해구제 접수 현황]

(단위 : 건, %)

	2016년		2017년		2018년		2019년 6월		계	
	건수	비율	건수	비율	건수	비율	건수	비율	건수	비율
경기도	76	(25.3)	68	(27.9)	62	(36.0)	35	(45.5)	241	(30.4)
서울특별시	63	(21.0)	46	(18.9)	31	(18.0)	7	(9.1)	147	(18.5)
인천광역시	24	(8.0)	18	(7.4)	9	(5.2)	8	(10.4)	59	(7.4)
부산광역시	20	(6.7)	13	(5.3)	9	(5.2)	4	(5.2)	46	(5.8)
경상남도	17	(5.7)	4	(1.6)	8	(4.7)	3	(3.9)	32	(4.0)
대구광역시	12	(4.0)	16	(6.6)	3	(1.7)	1	(1.3)	32	(4.0)
기타	88	(29.3)	79	(32.4)	50	(29.1)	19	(24.7)	236	(29.8)
전체	300	(100.0)	244	(100.0)	172	(100.0)	77	(100.0)	793	(100.0)

* 기타 : 경북 32건, 대전 31건, 전북 29건, 충남 29건, 충북 27건, 전남 25건, 광주 22건 등

중고차 거래량 ↑ → 중고차 거래 피해 ↑



○ 중고차 소비자피해 유형을 살펴보면 '성능상태 점검내용과 실제 차량상태가 다른 경우'가 632건(79.7%)으로 가장 많았고, 다음으로 매매업자가 양수인(구입자)으로부터 미리 받은 등록비 등의 차액을 반환해주지 않는 '제세공과금 미정산' 34건(4.3%), 계약위반 등으로 인한 '계약금 환급 지연·거절' 17건(2.1%) 등이었음.

- '성능·상태 점검내용과 실제 차량상태가 다른 경우'(632건, 79.7%) 중에는 '성능·상태 불량'(572건, 전체 신청사건의 72.1%)이 가장 많았고, 다음으로 '주행거리 상이'(25건, 3.2%), '침수차량 미고지'(24건, 3.0%) 등이었음.

성능 상태 점검내용과 차이

성능 상태 · 불량

주행거리 상이

침수차량 미고지

제세공과금 미정산

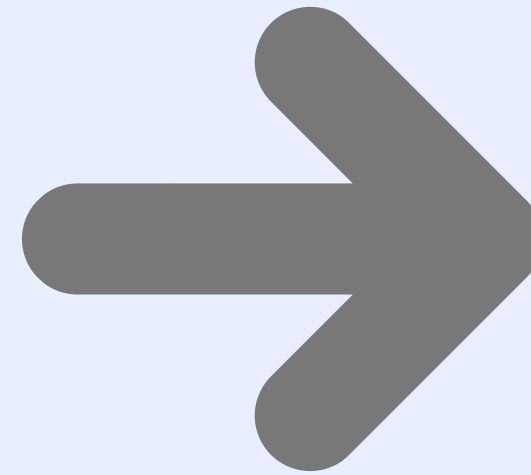
시장조사

기존 문제점

-개인간의 거래



Seller



Buyer

중고차에 대한 품질 보증이 어려움

**CREATIVITY
INSPIRED**

Knowledge is Power.

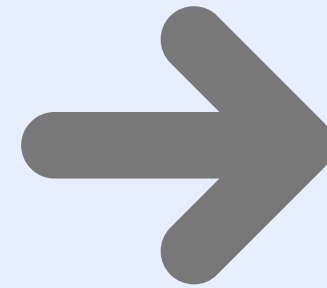
시장조사

기존 문제점

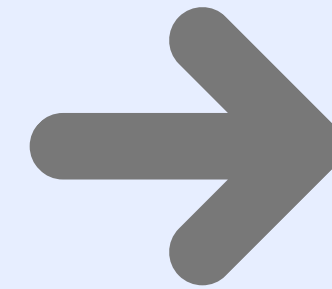
-딜러를 통한 거래



Seller



Dealer



Buyer

매입에 따른 중개 수수료 과다 부여

**CREATIVITY
INSPIRED**

Knowledge is Power.

시장조사

Features



기존에 국내에서 중고차를 운영하고 있는 업체들의 문제점



허위매물과
차량 정보에 대한 조작

유용한 이점 파악

1. 신뢰있는 데이터

자동차에 성능에 대해서 전문적인 인력을 통해 쉽게 확인할수있다

2. 위변조 불가

차량 거래전에 차량에대한 전반적인 상태를 알아볼수있다.

3. 수수료 완화

딜러를 거쳐서 사용되는 딜러 수수료를 줄일수있다.

4. 안전한 거래

암호화폐화된 결제로 판매자와 구매자에게 안전한 거래가 이루어지게 해줄 수있다

CREATIVITY
INSPIRED

Knowledge is Power.

02

사업 계획

수익구조
플랫폼 구조

사업 계획

토큰 생태계 구조

인센티브 토큰 지급



Buyer

상품에 대한 감정된 정보를 보고
상품의 구매를 결정한다.



Seller

물건상품을 등록하고 물품에 대한
정보를 제공한다.



Appraiser

판매자가 판매하려는 차량에 대한
성능이나 상태를 확인해준다.

사업 소개

중고차 판매 -> 블록체인으로 관리

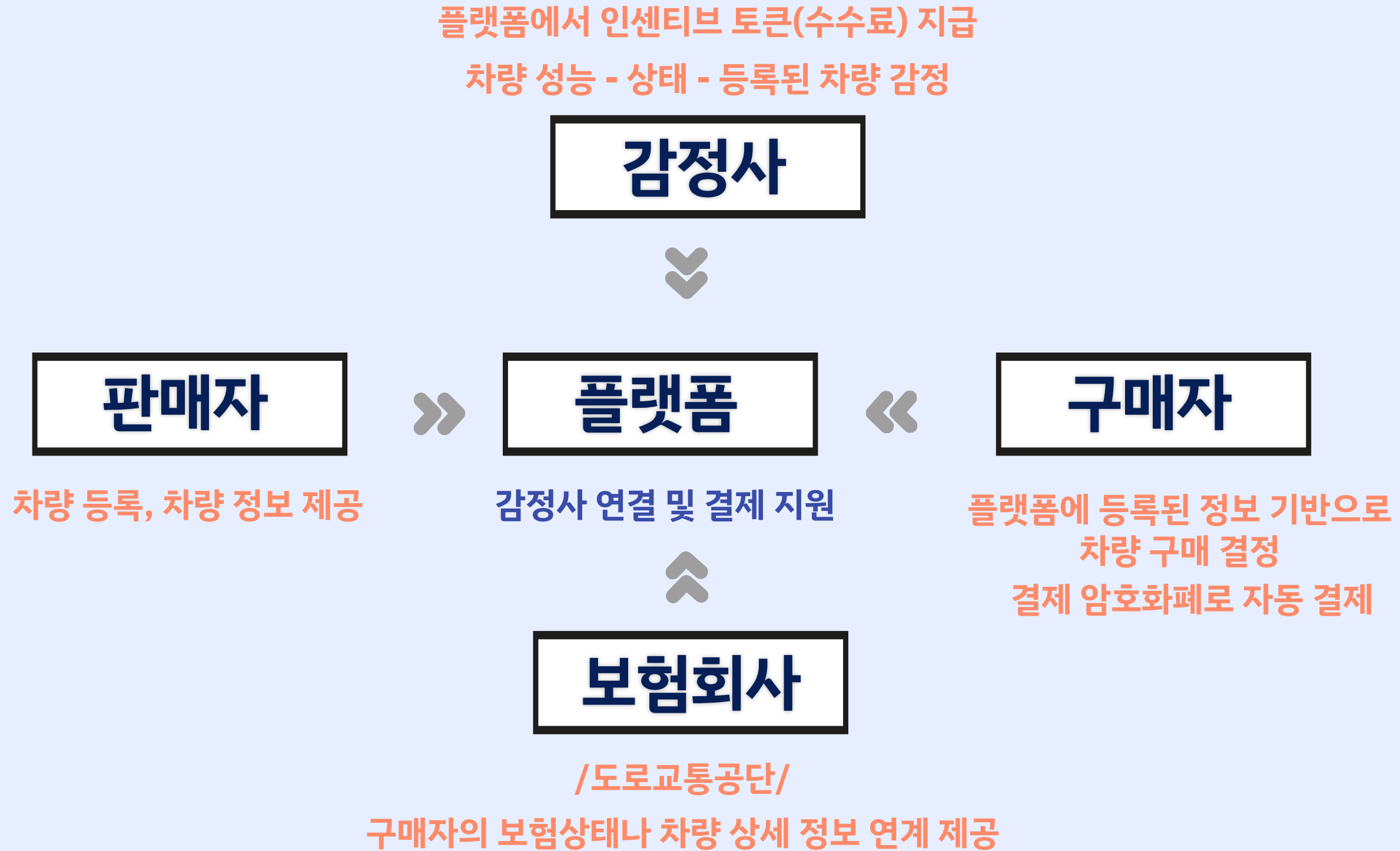
자동차 중고거래 시장을 블록체인으로 투명하게 관리하는 플랫폼

- 블록체인 기반 차량 정보 및 매물 정보 관리 기술 데이터 제공
- 토큰 이코노미를 활용한 멤버십 제공

<절차>

1. 매물 등록 시 감정사가 매물 검증과 차량 검수 완료
2. 등록 담당자가 차량의 주요 정보와 매물정보, 인증정보를 플랫폼에 입력
3. 구매자는 다양한 매물 확인 가능 (딜러 수수료 절감)

BlockChain Route (B2C)



Project Final Result



03

제작 구조

SRC 및 SmartContract

<https://remix.ethereum.org/#optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.0+commit.c7dfd78e.js>

The image shows the Remix IDE interface with a Solidity smart contract for a car market and its execution logs.

Contract Code (Left Panel):

```
1 // SPDX-License-Identifier: GPL-3.0
2 pragma solidity ^0.8.0;
3
4 import "@openzeppelin/contracts/token/ERC20/ERC20.sol";
5
6 contract CarMarket is ERC20{
7     address payable public owner;
8     address payable public seller;
9     address payable public appraiser;
10    enum Statuses{Sale, Sold}
11    Statuses currentStatus;
12    event Sold(address _buyer, uint _value);
13    uint public seller_value; // 판매자 이더 잔고
14    uint public owner_value; // 종개자(owner) 이더 잔고 / 수수료 3%
15    uint public appraiser_value; // 감정사 이더 잔고 / 수수료 5%
16
17    constructor(string memory name, string memory symbol) ERC20(name,symbol){
18        owner = payable(msg.sender);
19        currentStatus = Statuses.Sale;
20    }
21    modifier onlyOwner{
22        require(msg.sender==owner,"only owner can reg");
23        _;
24    }
25    // Seller address 등록
26    function RegSeller(address payable _seller) public onlyOwner{
27        seller = _seller;
28    }
29
30    // Appraiser address 등록
```

Contract Code (Right Panel):

```
function RegAppraiser(address payable _appraiser) public onlyOwner{
    appraiser = _appraiser;
}
/*
modifier onlyWhileVacant{
    require(currentStatus == Statuses.Sale, "sold out");
    _;
}
*/
modifier costs (uint _amount){
    require(msg.value >=_amount,"Not Enough provided");
    _;
}
receive() external payable costs(0 ether){
    currentStatus = Statuses.Sold;
    // 구매 금액의 3%를 owner에게 전송
    owner_value = (msg.value * 3) / 100;
    // 구매 금액의 92%를 seller에게 전송
    seller_value = (msg.value * 92) / 100;
    // 구매 금액의 5%를 appraiser에게 전송
    appraiser_value = (msg.value * 5) / 100;

    // 구매 금액이 아닌 수수료 토큰 지급
    _mint(appraiser,1000);

    // owner, seller에게 이더 전송.
    owner.transfer(owner_value);
    seller.transfer(seller_value);
    appraiser.transfer(appraiser_value);

    emit Sold(msg.sender,msg.value);
}
```

Transaction Log (Bottom Panel):

- CALL [call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: CarMarket.seller() data: 0x085...51a53
call to CarMarket.seller_value
- CALL [call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: CarMarket.seller_value() data: 0xc20...t
call to CarMarket.seller_value
- CALL [call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: CarMarket.seller_value() data: 0xc20...t
creation of CarMarket pending... [call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: CarMarket.seller() data: 0x085...51a53
o CarMarket.seller_value
- CALL [call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: CarMarket.seller_value() data: 0xc20...t
o CarMarket.seller_value

Deployed Contracts (Left Panel):

- CARMARKET AT 0XD91...39138 (MEMORY)
- approve
- decreaseAllow...
- increaseAllow...
- RegAppraiser
- RegSeller
- transfer
- transferFrom
- allowance
- appraiser
- appraiser_value
- balanceOf
- decimals
- name
- owner
- owner_value

SRC 및 SmartContract

DEPLOY & RUN TRANSACTIONS

Transactions recorded 4

Deployed Contracts

- CARMARKET AT 0XD91...39138 (MEMORY)

approve address spender, uint256 amount

decreaseAllow... address spender, uint256 subtra

increaseAllow... address spender, uint256 added

RegAppraiser 0x4B20993Bc481177ec7E8f57

RegSeller 0xAb8483F64d9C6d1EcF9b845

transfer address to, uint256 amount

transferFrom address from, address to, uint256 amount

allowance address owner, address spender

appraiser

0: address: 0x4B20993Bc481177ec7E8f571ceCaE8A9e22C02db

appraiser_value

balanceOf address account

decimals

name

owner

0: address: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4

owner_value

```
1 // SPDX-License-Identifier: GPL-3.0
2 pragma solidity ^0.8.0;
3
4 import "@openzeppelin/contracts/token/ERC20/ERC20.sol";
5
6 contract CarMarket is ERC20{
7     address payable public owner;
8     address payable public seller;
9     address payable public appraiser;
10    enum Statuses{Sale, Sold}
11    Statuses currentStatus;
12    event Sold(address _buyer, uint _value);
13    uint public seller_value; // 판매자 이더 잔고
14    uint public owner_value; // 중개자(owner) 이더 잔고 / 수수료 3%
15    uint public appraiser_value; // 감정사 이더 잔고 / 수수료 5%
16
17    constructor(string memory name, string memory symbol) ERC20(name,symbol)
18    {
19        owner = payable(msg.sender);
20        currentStatus = Statuses.Sale;
21    }
22    modifier onlyOwner{
23        require(msg.sender==owner,"only owner can reg");
24        _;
25    }
26    // Seller address 등록
27    function RegSeller(address payable _seller) public onlyOwner{
28        seller = _seller;
29    }
30    // Appraiser address 등록
```

ACCOUNT +

- 0x4B2...C02db (100.3 ether)
- 0x5B3...eddC4 (100.179999999998206394 ether)
- 0xAb8...35cb2 (105.52 ether)
- 0x4B2...C02db (100.3 ether)
- 0x787...cabaB (93.999999999999824778 ether)
- 0x617...5E7f2 (100 ether)
- 0x17F...8c372 (100 ether)

=> Owner
=> Seller
=> Appraiser
=> Buyer

CALL [call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: CarMarket.seller() data: 0x085...51a53
call to CarMarket.seller_value

CALL [call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: CarMarket.seller_value() data: 0xc20...b4c3
call to CarMarket.seller_value

CALL [call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: CarMarket.seller_value() data: 0xc20...b4c39
creation of CarMarket pending...

balanceOf 0x4B20993Bc481177ec7E8f57

0: uint256: 1000

감정사가 1대 당 받는 인센티브 토큰

이더스캔 분배 현황

이더스캔 주소 : <https://ropsten.etherscan.io/tx/0xb4fc4f058db2d26d54ce4e8751de24957970357d344325d375f5baac9d15fd83>

스마트 컨트랙트 주소 : 0x96abfdaa946ca6405363d228571075acc9a931dc

The screenshot shows the Etherscan interface for a contract on the Ropsten Testnet. The contract address is 0x96abfdaa946ca6405363d228571075acc9a931dc. The contract overview shows a balance of 0 Ether. The more info section includes a name tag (Not Available), contract creator (0xe1ce9fe03665763824...), and a token tracker for 'car (car)'. The transactions section shows a list of 7 transactions, with the second transaction (0x7f720075c010974eb5...) highlighted in pink. The right sidebar shows a detailed view of the highlighted transaction, including the contract address and the amount of Ether transferred (0.005 Ether).

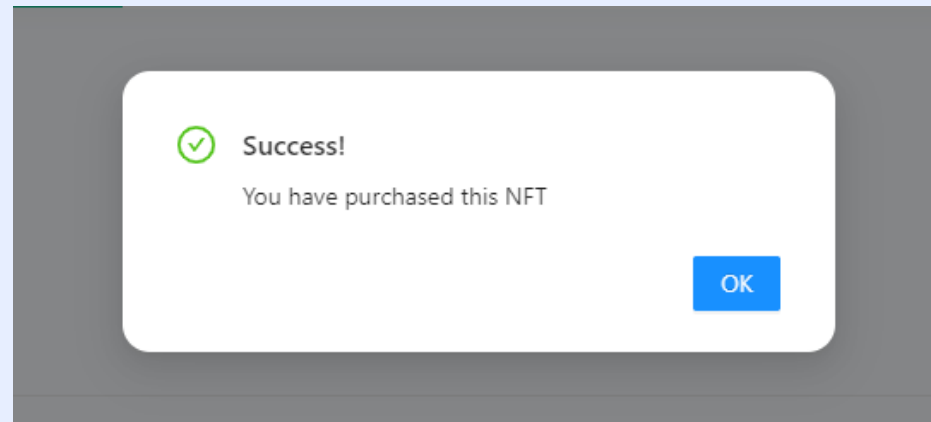
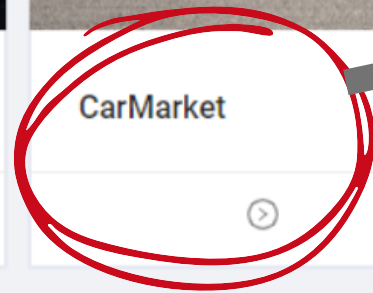
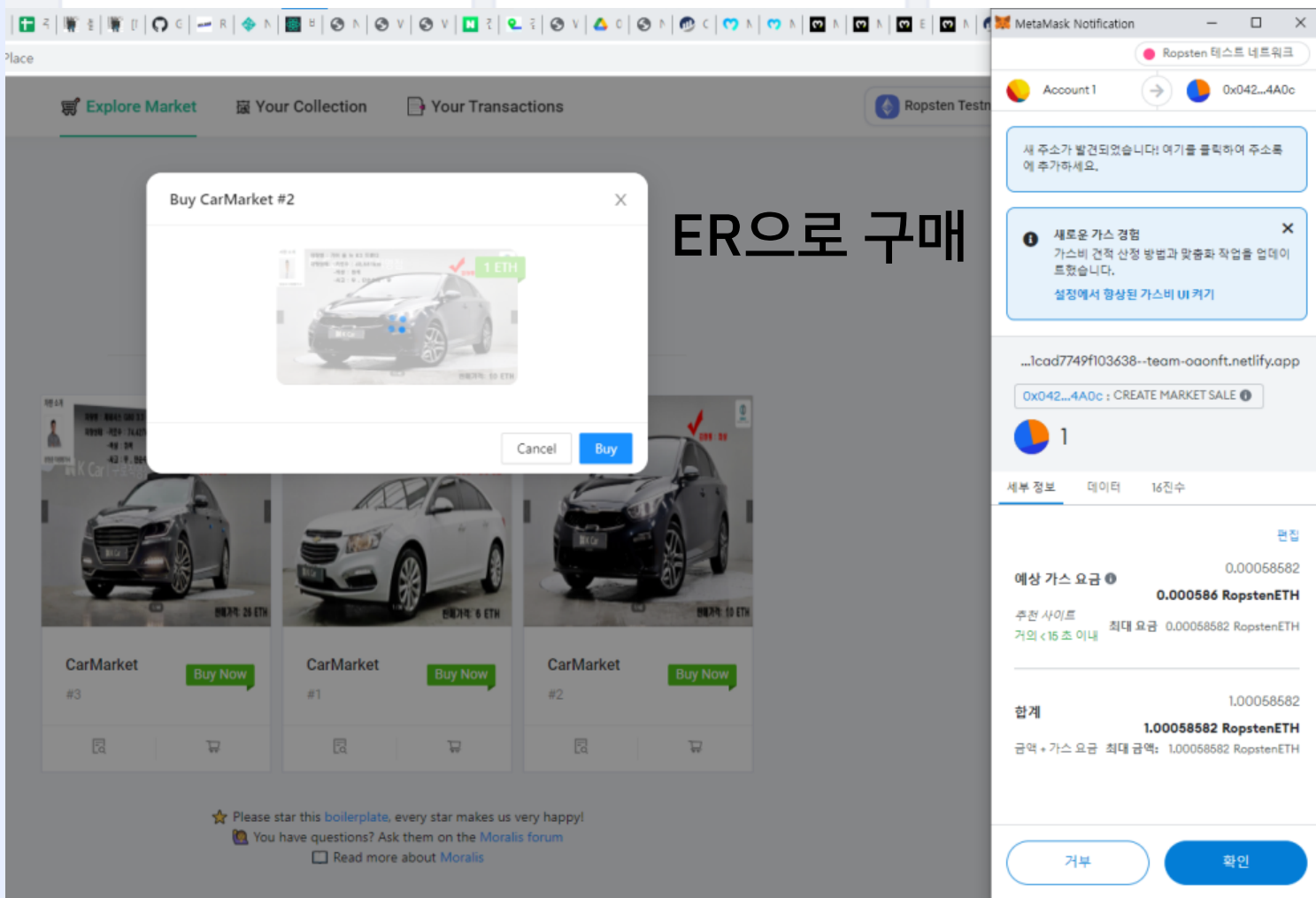
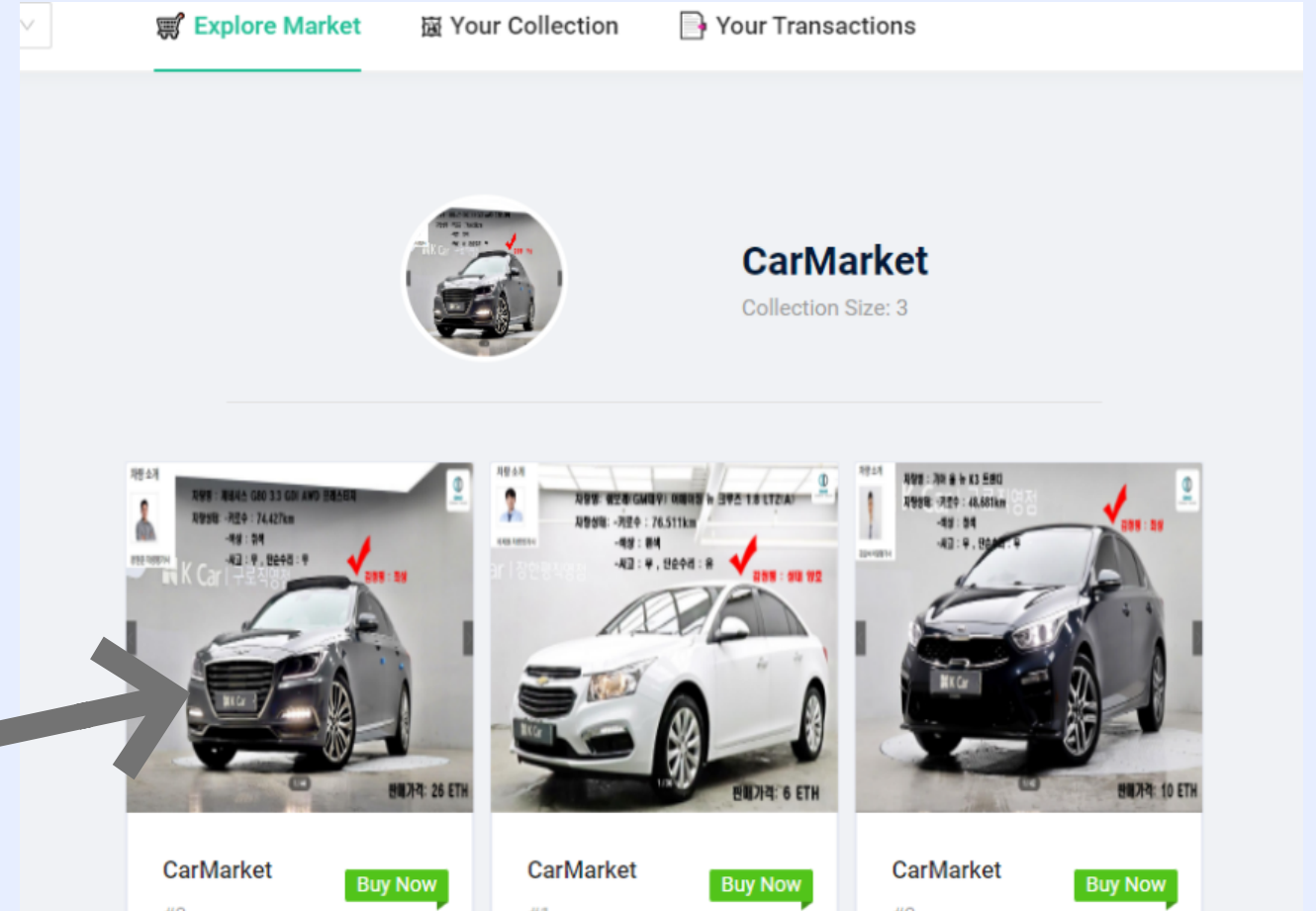
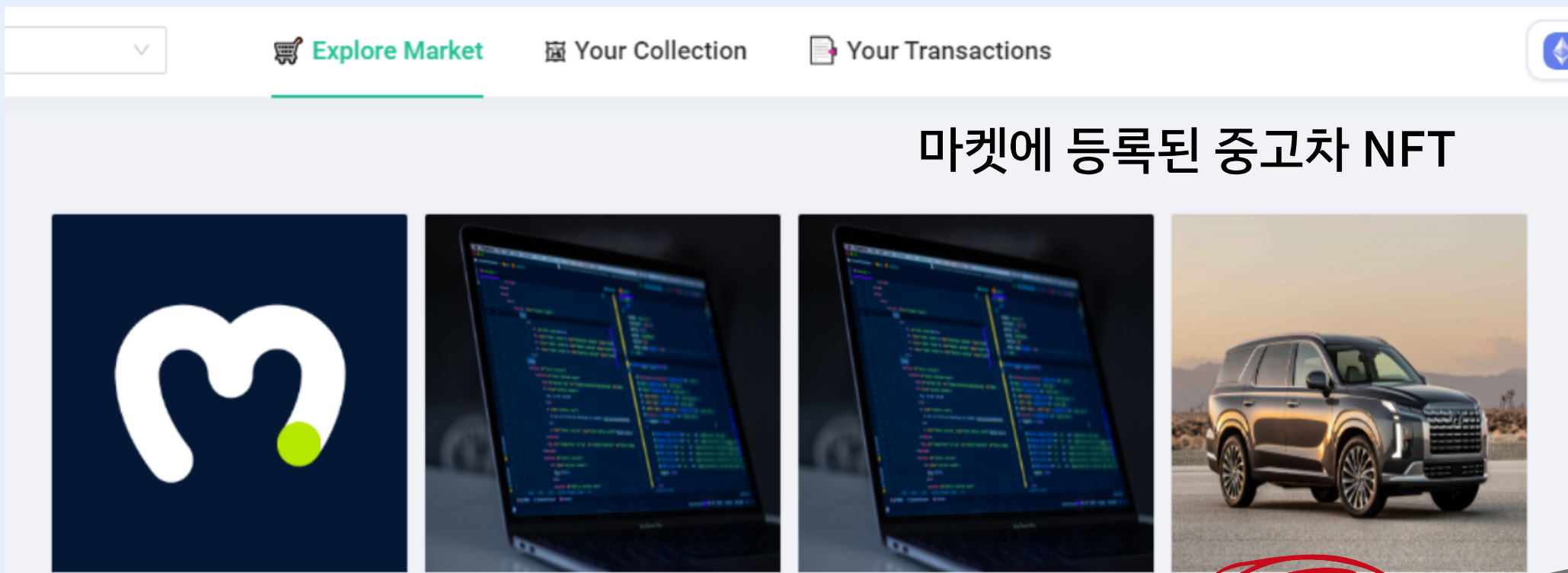
Txn Hash	Method	Block	Age	From	To
0xb4fc4f058db2d26d54c...	Transfer	12582859	1 min ago	0xd008e081e97d6a78dc...	0x96abfdaa946ca64053...
0x7f720075c010974eb5...	0xc8ee0a56	12582855	2 mins ago	0xe1ce9fe03665763824...	0x96abfdaa946ca64053...
0x981404a4d9e9bb6305...	Transfer	12582841	5 mins ago	0xd008e081e97d6a78dc...	0x96abfdaa946ca64053...
0x0d09251b77de1af6dcf...	Transfer	12582793	17 mins ago	0xd008e081e97d6a78dc...	0x96abfdaa946ca64053...
0xfbda4fa9c40c9bb0cda...	0xc8ee0a56	12582791	17 mins ago	0xe1ce9fe03665763824...	0x96abfdaa946ca64053...
0xc9d61bf4660b0792d2...	0x3c268d88	12582788	18 mins ago	0xe1ce9fe03665763824...	0x96abfdaa946ca64053...
0x77e9f653d8142006ea...	0x60806040	12582780	20 mins ago	0xe1ce9fe03665763824...	Contract Creation

Owner : 3 %
Seller : 92 %
Appraiser : 5 %

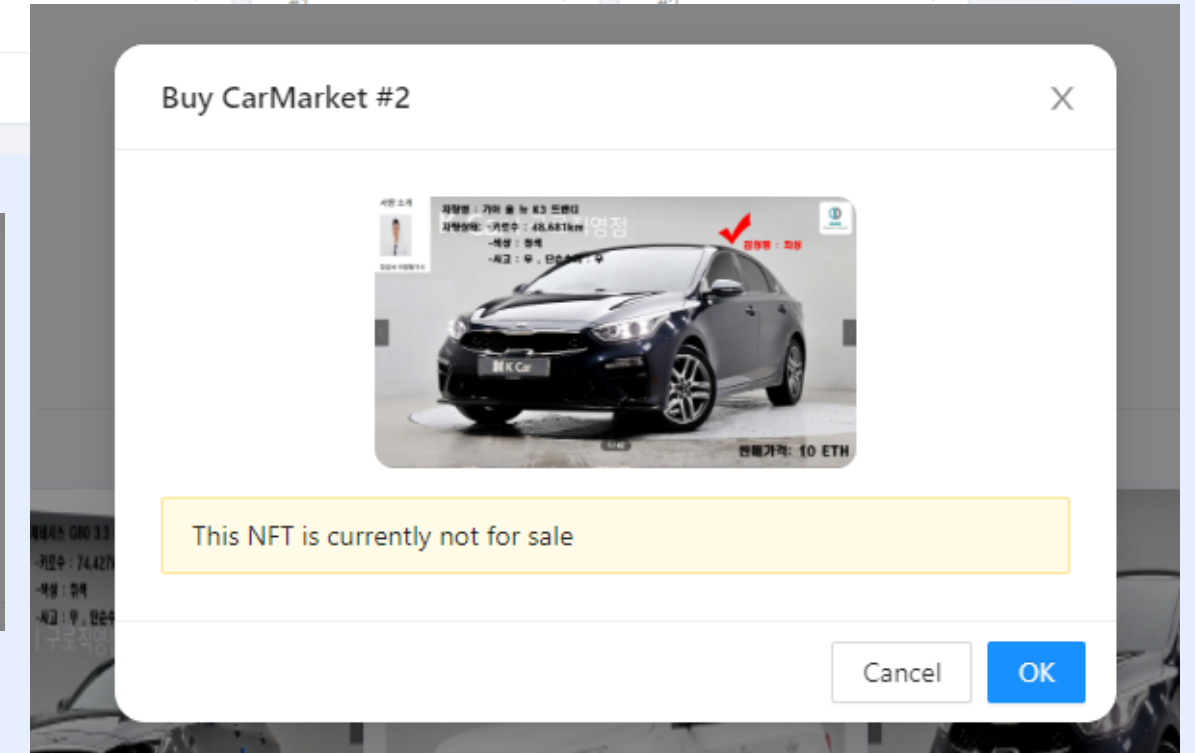
비율로 분배

NFT 등록 및 구매

<https://62cf805f01cad7749f103638--team-oaonft.netlify.app/NFTMarketPlace>



구매 성공



구매 시 차량 판매 완료 (구매불가)

NFT 등록 방법

```

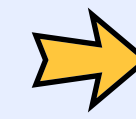
metadata - Windows 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
{
  "description": "abc ",
  "image": "ipfs://QmVapTTx9qhmzXqc53gzCLx2r7qaPBcZw7WpZioVi5xnt4",
  "name": "abc"
}
    
```

metadata.json image 주소 변경



Name	CID
metadata.json	Qmc7iZaa3Ma1MpDjx9aVGy9zpv1QnJGSS8vsUF7Y3wXeHL
metadata.json	QmPgd8NTA972yZ7FyA385Wg7y2Snie6ehivHuu8z4uwVJv
metadata.json	QmPGFaCar3w5GB88ocNce7AZEXtXxnz78TzwStg8LmXubv
Car3.jpg	QmVapTTx9qhmzXqc53gzCLx2r7qaPBcZw7WpZioVi5xnt4
	QmdrnEfAZZY3jkpn1HZY2mDZsfPrW9uG4QP3t92aWEE4xa
	QmYwnMP74Uebqj91RQFxmHvGTFsdMFQgMpGnLbB6K7jh4k
	QmPYUA11hJ17LaoAey1vjjjuQCYvpDrKeTaaX1SBcixjb

metadata.json pinata 업로드



Ropsten 테스트 네트워크

Account 1 → 0x699...5E5e

새 주소가 발견되었습니다! 여기를 클릭하여 주소록에 추가하세요.

새로운 가스 경험
가스비 견적 산정 방법과 맞춤형 작업을 업데이트했습니다.
설정에서 향상된 가스비 비켜기

세부 정보 데이터 16진수

거의 < 15 초 이내

합계 0.00032012
0.00032012 RopstenETH

금액 + 가스 요금 최대 금액:
0.00032012 RopstenETH

거부 확인

metamask 연결

DEPLOY & RUN TRANSACTIONS

Transactions recorded 5

Deployed Contracts

NFT AT 0X337...AAD26 (BLOCKCHAIN)

NFT AT 0X699...5E5E (BLOCKCHAIN)

approve address to, uint256 tokenId

mintNFT ipfs://Qmc7iZaa3Ma1MpDj

renounceOwn...

safeTransferFr... address from, address to, u

safeTransferFr... address from, address to, u

setApprovalFo... address operator, bool app

transferFrom address from, address to, u

transferOwner... address newOwner

balanceOf address owner

getApproved uint256 tokenId

isApprovedFor... address owner, address op

name

owner

ownerOf uint256 tokenId

supportsInterf... bytes4 interfaced

symbol

```

1 // SPDX-License-Identifier: GPL-3.0
2 pragma solidity ^0.8.0;
3
4 import "@openzeppelin/contracts/token/ERC721/ERC721.sol";
5 import "@openzeppelin/contracts/utils/Counters.sol";
6 import "@openzeppelin/contracts/access/Ownable.sol";
7 import "@openzeppelin/contracts/token/ERC721/extensions/ERC721URIStorage.sol";
8 import "@openzeppelin/contracts/utils/introspection/IERC165.sol";
9
10 contract NFT is ERC721URIStorage, Ownable{
11     using Counters for Counters.Counter;
12     Counters.Counter private tokenId;
13
14     constructor() ERC721("CarMarket", "OAO"){}
15
16     function mintNFT(string memory tokenURI) public onlyOwner returns (uint256){
17         tokenId.increment();
18         uint256 newId = tokenId.current();
19         _mint(msg.sender, newId);
20         _setTokenURI(newId, tokenURI);
21         return newId;
22     }
23 }
    
```

ContractDefinition NFT 1 reference(s)

view on etherscan

[block:12587722 txIndex:3] from: 0xe1C...e0503 to: NFT.(constructor) value: 0 wei data: 0x608...00033
transact to NFT.mintNFT pending ...

view on etherscan

[block:12587732 txIndex:1] from: 0xe1C...e0503 to: NFT.mintNFT(string) 0x699...5E5e value: 0 wei data:
transact to NFT.mintNFT pending ...

view on etherscan

[block:12587755 txIndex:0] from: 0xe1C...e0503 to: NFT.mintNFT(string) 0x699...5E5e value: 0 wei data:
transact to NFT.mintNFT pending ...

compileAndRun is requiring to compile contracts/M

remix 업로드 주소 mint

CarMarket #1

CarMarket #2

판매가격: 6 ETH

판매가격: 10 ETH



CarMarket #3

CarMarket #1

CarMarket #2

판매가격: 20 ETH

판매가격: 6 ETH

판매가격: 10 ETH

NFT 마켓 등록

04

결과 보고

Issue

1. 공정한 수익분배(부당이익 취득 불가)
2. 수수료에서 상대적으로 높은 보완성 => 사기 가능성 낮춤
3. 정보 왜곡이 어려워 신뢰도가 높음
4. 대량으로 축적된 거래 데이터를 가공하여 2차 정보로써 수익 창출

Risk

1. 원활한 정보 삭제 불가능
2. 구매 원화의 불확실성

oAo 블록체인 프로젝트 계획서

Thank you

oAo Company

박수권
김민상
박준규