

블록체인이론과 실습 과제1

# 블록체인 클라우드 펀딩 회사 BADIZ Project



컴퓨터정보통신대학원  
디지털융합금융학과 3조  
정낙현, 엄호천, 박진수, 김주원

# Contents

1. 블록체인 클라우드 펀딩 회사
2. BADIZ 소개
3. BADIZ 사업모델
4. BADIZ 토큰 생태계
5. BADIZ 플랫폼
6. BADIZ 로드맵
7. 부록





# 블록체인 클라우드 펀딩 회사

01

# 1-1. 블록체인 클라우드 펀딩 회사

## 1) 수익형 펀딩 : 엔터테인먼트 클라우드 펀딩

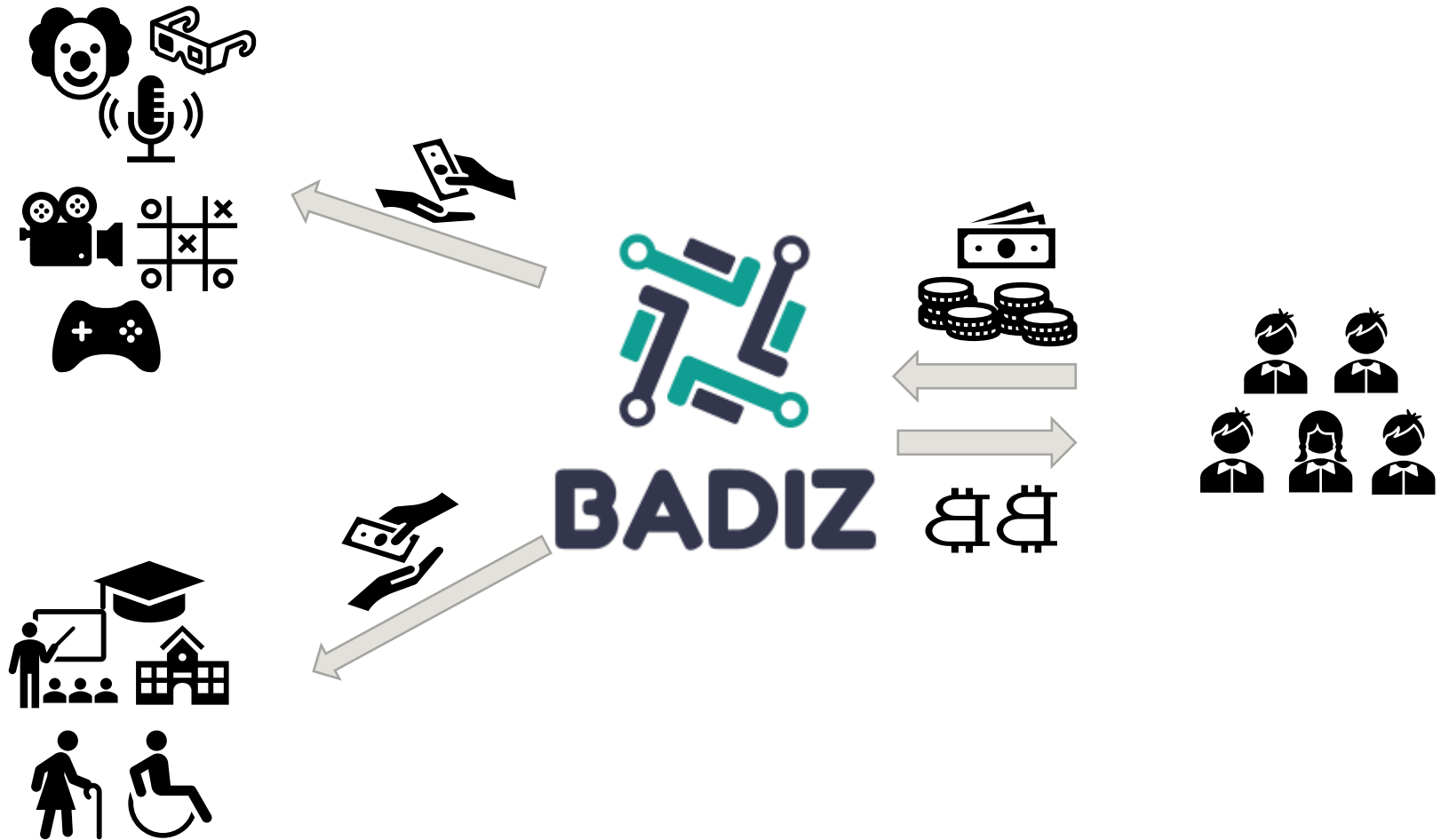
- 연예인 꿈나무(유튜버,인플루언서) 양성 투자 펀딩
  - 영화기획 투자 펀딩
  - 게임기획 투자 펀딩
- 클라우드 펀딩을 통하여 투자자 자금을 모집 후 투자자에게 토큰발행, 투자대상을 발굴하여 지분투자 및 대출계약을 하여 발생하는 수익을 투자자에게 배분

## 2) 비수익형 펀딩 : 사회복지 클라우드 펀딩

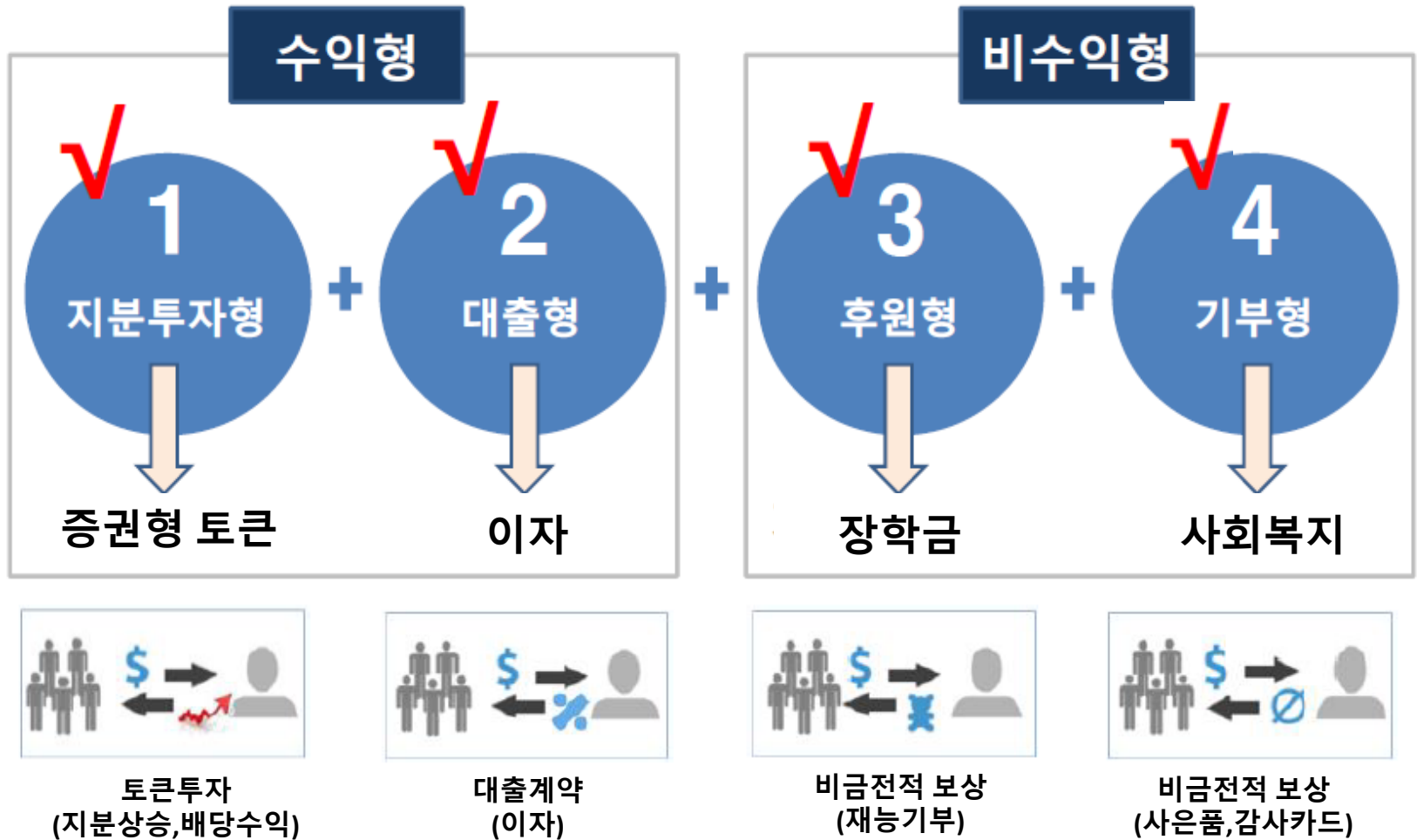
- 학자금/장학금 지원 펀딩
  - 기부금 펀딩
- 클라우드 펀딩으로 투자자 모집 , 펀딩대상 선정 후 투자금이나 수익을 사회복지사업에 투자



## 1-2. 블록체인 클라우드 펀딩 개요



# 1-3. 블록체인 클라우드 펀딩 모델





## BADIZ 소개

02

## 2-1. 회사 개요

사명	웹사이트	로고
BADIZ	<a href="http://nhjung74.cafe24.com/">http://nhjung74.cafe24.com/</a>	<b>BADIZ</b>

### 슬로건

- Speed : 빠른 클라우드 펀딩
- Simple : 단순한 클라우드 펀딩
- Security : 블록체인 기반 보안강화
- Scalable : 다양한 클라우드 펀딩 기회

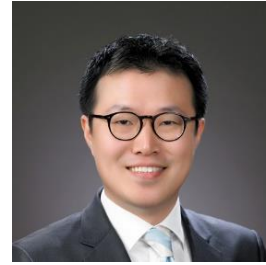




## 2-2. 주요인력



**정낙현 (CEO)**  
BADIZ 총괄 대표



**엄호천 (CFO)**  
BADIZ 총괄 재무기획



**박진수 (CTO)**  
BADIZ 총괄 기술



**김주원 (CMO)**  
BADIZ 총괄 마케팅



## 2-3. 회사 미션

세계에서 가장 빠르고 확장 가능하며 쉬운 블록체인 크라우드 펀딩 제공  
we aim to make Crowd Funding the fastest, most scalable, and  
easiest to use blockchain in the world.



### Fastest

- 빠른 크라우드 펀딩



### Scalable

- 다양한 크라우드 펀딩



### User-Freindly

- 사용자 친화적



## 2-4. 클라우드 펀딩 문제점

### 기존 클라우드 펀딩 문제점

- 법,규제에 의한 펀딩 절차 복잡
- 비싼 수수료
- 투자자 의사 결정 어려움

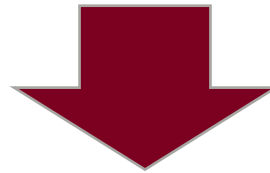
wadiz

CROWDY

tumblbug

OHMYCOMPANY

INDIEGOGO



BADIZ

- 빠른 클라우드 펀딩 투자
- 싼 수수료
- 투명한 투자 집행, 의사결정



## 2-5. 솔루션

### 블록체인 클라우드 펀딩

- 지분투자형
- 대출형
- 후원형
- 기부형



### 투자자

- 보안 강화
- 의사 결정
- 투자 기회



### 자금 수요자

- 자금 조달
- 기회 제공












### 엔터프라이즈

- 유연성향상
- 커뮤니티





# 2-6. Bussiness Model Canvas

<p><b>Key Partners</b> </p> <ul style="list-style-type: none"> <li>• 유튜버</li> <li>• 영화기획사</li> <li>• 방송종사자</li> <li>• 게임개발사</li> <li>• 학교, 장학재단</li> <li>• 복지재단</li> </ul>	<p><b>Key Activities</b> </p> <ul style="list-style-type: none"> <li>• 투자대상발굴 및 심사</li> <li>• 투자심사</li> <li>• 토큰발행</li> <li>• 홍보</li> <li>• 사후관리</li> <li>• 기부자금 활용 감시</li> <li>• 총회 투표</li> </ul> <p><b>Key Resources</b> </p> <ul style="list-style-type: none"> <li>• 양질의 투자대상</li> </ul>	<p><b>Value Propositions</b> </p> <ul style="list-style-type: none"> <li>• 다양한 투자기회</li> <li>• 성장 참여기회 제공</li> <li>• 원활한 자금조달</li> <li>• 홍보효과</li> <li>• 사회적 기부</li> <li>• 기부자금 모집, 집행의 투명성확보</li> <li>• 공동의사결정</li> </ul>	<p><b>Customer Relationships</b> </p> <ul style="list-style-type: none"> <li>• 게시판,커뮤니티</li> <li>• 커뮤니티 운영</li> <li>• 포인트,마일리지</li> <li>• 투자그룹(동호회)</li> </ul> <p><b>Channels</b> </p> <ul style="list-style-type: none"> <li>• 클라우드 펀딩 사이트</li> <li>• 어플리케이션(모바일)</li> <li>• 소셜미디어(SNS)</li> <li>• 투자교육</li> </ul>	<p><b>Customer Segments</b> </p> <ul style="list-style-type: none"> <li>• 인터넷,모바일 사용층</li> <li>• SNS 접속 층</li> <li>• 나이,지역 제한 없음</li> <li>• 객장,성별 제한 없음</li> </ul>
<p><b>Cost Structure</b> </p> <ul style="list-style-type: none"> <li>• 사이트구축</li> <li>• 서버비용</li> <li>• 마케팅비용</li> <li>• 광보비용</li> </ul>		<p><b>Revenue Streams</b> </p> <ul style="list-style-type: none"> <li>• 선취수수료</li> <li>• 광고수익</li> <li>• 성공보수</li> </ul>		



# 2-7. BADIZ 사이트

BADIZ



로그인

주문조회

마이쇼핑

게시판

이벤트

펀딩중



[후원형] 장학금을 후원해주세요  
1원

+ Cart



[대출형] 신용대출 클라우드펀딩  
1호  
1원

+ Cart



[기부형] 연탄나눔1호  
1원

+ Cart



[지분투자]영화제작 클라우드 펀딩 1호  
1원

+ Cart



[지분투자]테크유투버 클라우드 펀딩1호  
1원

+ Cart

신규펀딩중



[후원형] 장학금을 후원해주세요  
1원

+ Cart



[기부형] 연탄나눔1호  
1원

+ Cart



[지분투자]영화제작 클라우드 펀딩 1호  
1원

+ Cart

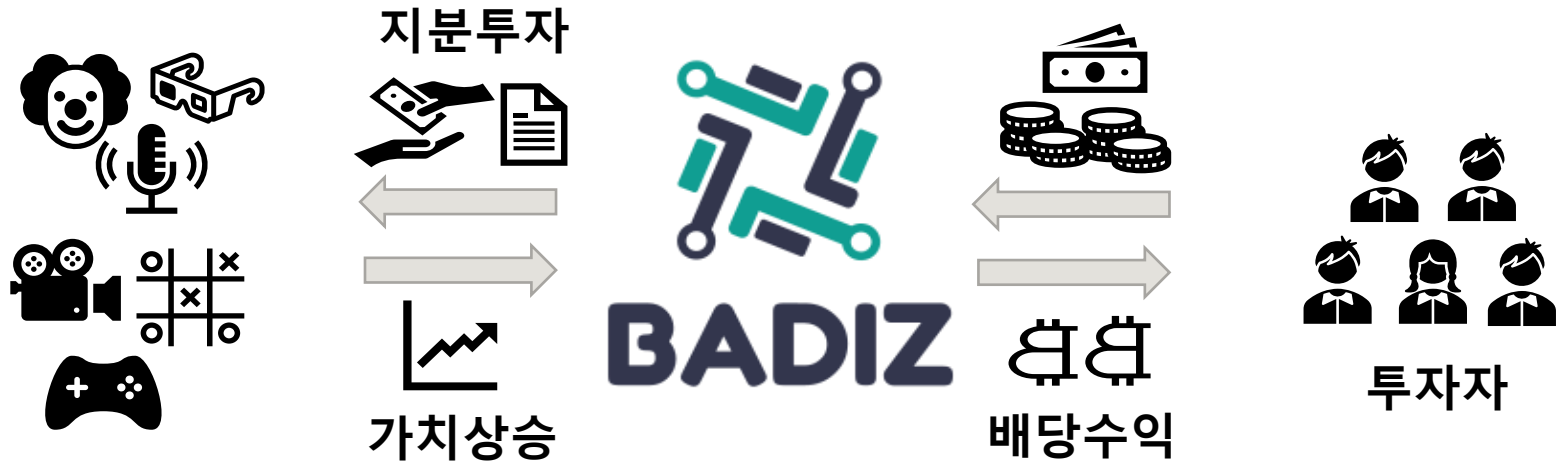




## BADIZ 사업모델

03

# 3-1. 지분투자형 블록체인 클라우드펀딩



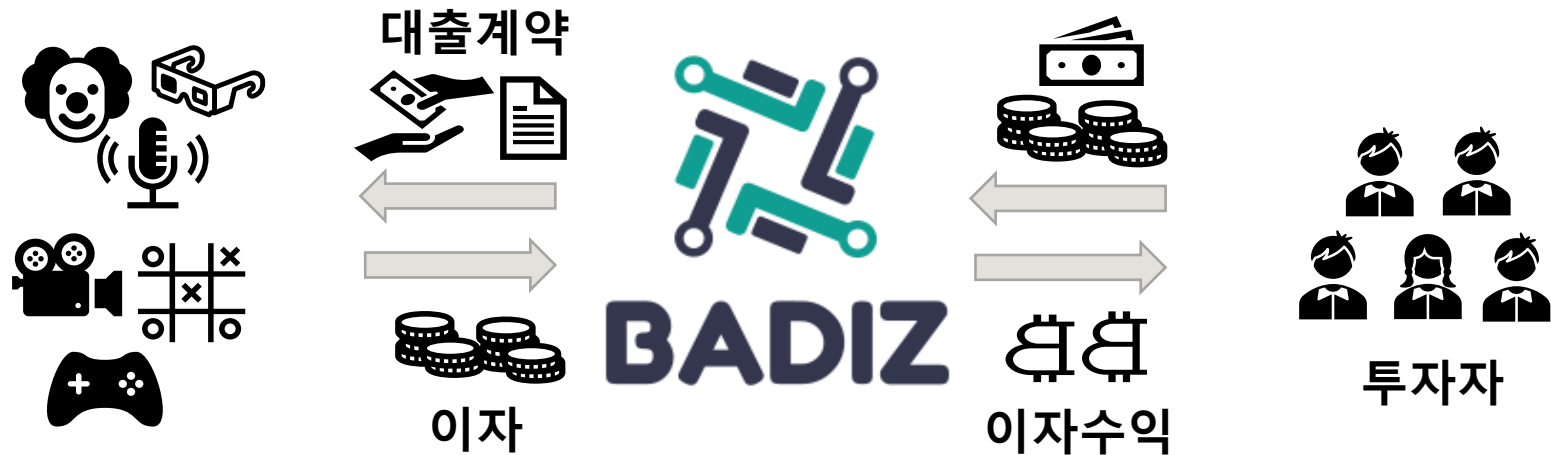
## 지분투자형 블록체인 클라우드펀딩

- ① 펀딩 개시  
투자자 모집
- ② 펀딩 완료  
투자자들 보유한 이더리움을 펀딩계좌로 입금, BAZ를 투자자계좌로 입금
- ③ 투자자금 집행  
펀딩계좌의 자금 집행 1차, 2차
- ④ 수익금 분배  
펀딩계좌로 수익금 입금되면 BAZ를 보유한 투자자에게 보유비율만큼 이더리움 입금
- ⑤ 펀딩 청산  
BAZ를 보유한 투자자에게 보유비율만큼 이더리움 입금, BAZ 소각, 펀딩 청산





## 3-2. 대출형 블록체인 크라우드펀딩

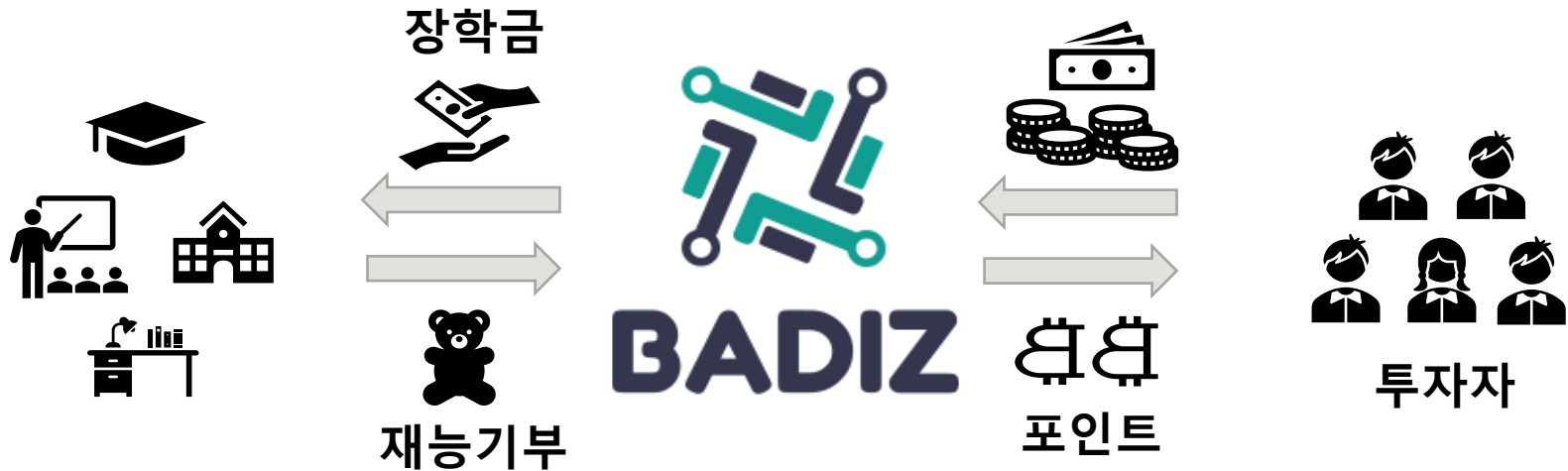


### 대출형 블록체인 크라우드펀딩

- ① 펀딩 개시  
투자자 모집
- ② 펀딩 완료  
투자자들 보유한 이더리움을 펀딩계좌로 입금, BAZ를 투자자계좌로 입금
- ③ 투자자금 집행  
펀딩계좌의 자금 집행 1차, 2차
- ④ 대출이자 분배  
펀딩계좌로 대출이자 입금되면 BAZ를 보유한 투자자에게 보유비율만큼 이더리움 입금
- ⑤ 펀딩 청산  
BAZ를 보유한 투자자에게 보유비율만큼 이더리움 입금, BAZ 소각, 펀딩 청산



### 3-3. 후원형 블록체인 크라우드펀딩

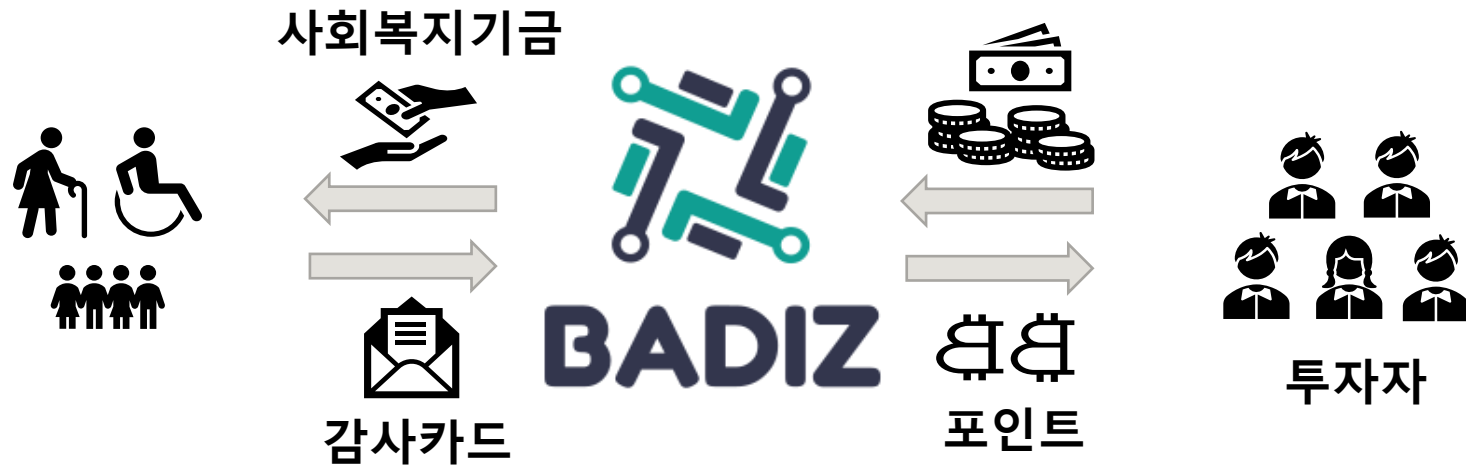


#### 후원형 블록체인 크라우드펀딩

- ① 펀딩 개시  
투자자 모집
- ② 펀딩 완료  
투자자들 보유한 이더리움을 펀딩계좌로 입금, BAZ를 투자자계좌로 입금
- ③ 투자자금 집행  
펀딩계좌의 자금 집행 1차, 2차  
집행자금 비율만큼 투자자 보유 BAZ 소각
- ④ 펀딩 종료  
최종자금 집행 후 투자자의 BAZ 소각, 펀딩 청산



## 3-4. 기부형 블록체인 크라우드펀딩



### 기부형 블록체인 크라우드펀딩

- ① 펀딩 개시  
투자자 모집
- ② 펀딩 완료  
투자자들 보유한 이더리움을 펀딩계좌로 입금, BAZ를 투자자계좌로 입금
- ③ 투자자금 집행  
펀딩계좌의 자금 집행 1차, 2차  
집행자금 비율만큼 투자자 보유 BAZ 소각
- ④ 펀딩 종료  
최종자금 집행 후 투자자의 BAZ 소각, 펀딩 청산



## 3-5. 환전,발행 개념



## 3-6. 클라우드 펀딩 수수료 체계

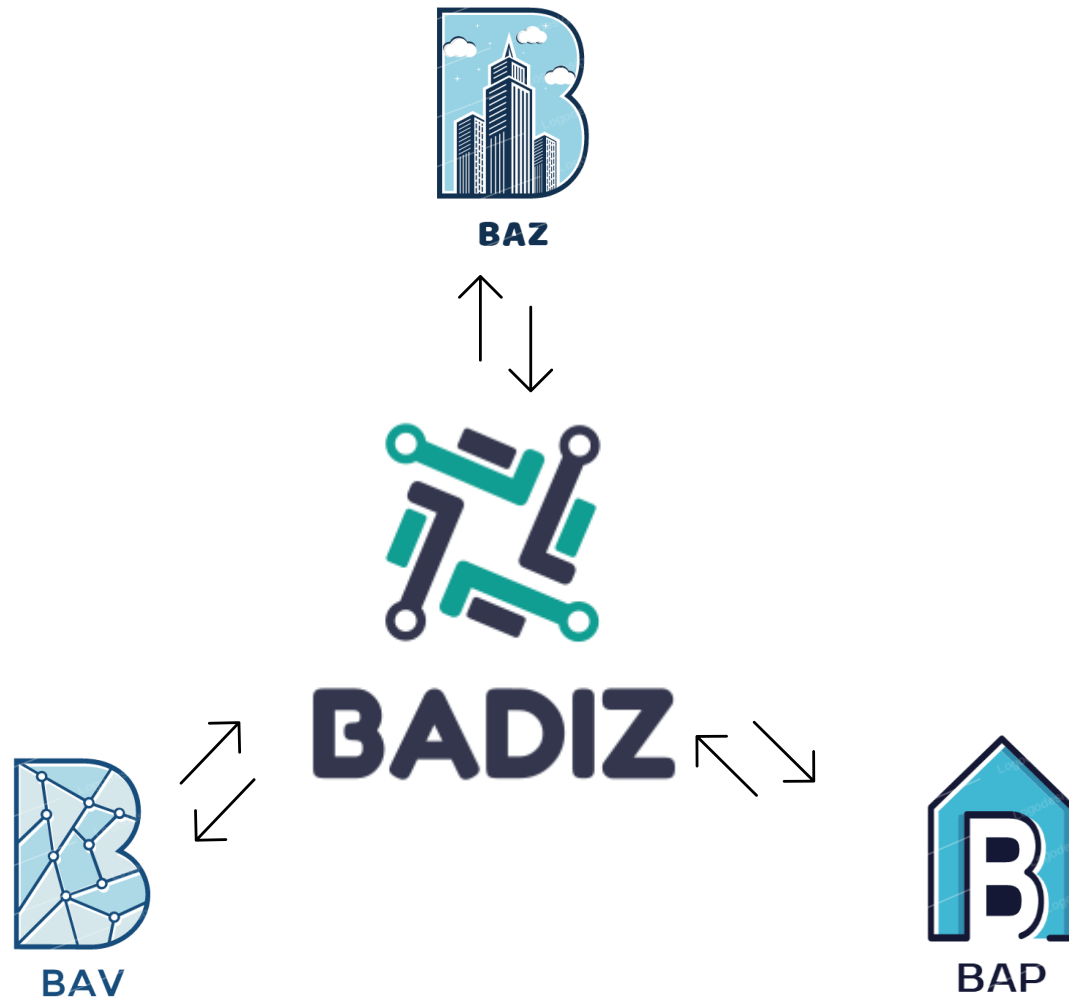




## BADIZ 토큰생태계

04

## 4-1. BADIZ 토큰 모델



## 4-2. BADIZ 토큰 종류



이름 : BAZ Token  
심볼 : BAZ  
단위 : 1  
종류 : 증권형 토큰 (DAI)  
설명 : 지분 소유권 , 수익금



이름 : BAV Token  
심볼 : BAV  
단위 : 1  
종류 : 거버넌스 토큰 (DAO)  
설명 : 최초 BADIZ 비즈니스 시작과 동시에 ICO  
투자 및 의사결정 투표



이름 : BAP Token  
심볼 : BAP  
단위 : 1  
종류 : 거래형 토큰  
설명 : 리워드, 포인트 , 클라우드펀딩시 발행,  
상품구매 , 클라우드 펀딩 자금 환전가능





## 4-3. BADIZ 토큰 할당



BAV

### BAV Token

투표	수량	비율	
채굴	1,500,000,000	50%	활동에 따른보상, Air Drop 등
판매	1,500,000,000	50%	Public/Private Sale , ICO 등

**최초 발행 및 할당 수량 : 3,000,000,000**

**BAV 소유자들이 투표를 통해 변경 , BADIZ 공식채널에 고지**

### 토큰 (BAV) 판매금 활용 계획 (안)

비율	수량	활용계획
20%	300,000,000	BADIZ 플랫폼 개발
50%	750,000,000	마케팅 , Air Drop 등
20%	300,000,000	연구소 설립 및 운영
10%	150,000,000	자체 FUNDING 자금



## 4-4. BADIZ 블록생성 보상 시스템

### 블록생성

- ① 블록생성 : 마스터노드
- ② 블록타임 : 3초
- ③ 1회 블록생성 수 : 5 BAZ
- ④ 펀딩별로 별도 블록 생성 ( 총 21 개 블록체인 가능 )

### 블록생성 후 보상

	마스터 노드(M-N)	서브노드(S-N)	F-ACCOUNT	B-ACCOUNT
1 회 블록생성수 연 5% 기준(BAZ)	5 BAZ			
블록타임(초)	3초			
보상비율	40%	40%	15%	5%
노드수	21	63	-	-

- ① 1회 블록 생성시 5 BAZ 발행하며, 블록 당 40% 보상비율로 보상 받는다.
- ② 서브노드는 1회 블록 생성시 발행된 5 BAZ 중에 블록 당 40% 보상비율로 보상을 받으며, 받은보상을 가지고 서브노드의 노드수로 1/n 나누어서 분배된다.



## 4-5. 노드

### 노드

- ① 마스터 노드는 서브 노드의 투표로 선출 , 플랫폼 유지시 의사결정권한
- ② 플랫폼에서 요구하는 권장사양의 서버를 운영
- ③ 코인 전체 물량의 0.2% BAZ 을 보유 (미달 시 블록보상 지급 안됨)
- ④ 블록생성을 통해서 BAZ 발행
- ⑤ BAZ 트랜잭션 처리
- ⑥ 투자진행시 투자결정 및 투자 관리 역할
- ⑦ 플랫폼 유지 위한 종합적인 의사결정

마스터 노드	서브 노드
블록생성	DApp 리소스 제공
트랜잭션 처리	트랜잭션 수집
투자를 위한 설정 및 기업 선정	투자리스트 선정
BADIZ 플랫폼 위한 의사결정	유사시 마스터 노드 대신 서버운영



# 4-6. ACCOUNT

## F-ACCOUNT

- ① 마스터노드에 의해 블록이 생성될 때 마다 블록 당 15% 보상비율로 BAZ 적립
- ② 투자에 사용되는 자금
- ③ 투자처에서 발생한 배당을 입금 및 투자자 분배시 사용

## B-ACCOUNT

- ① 마스터노드에 의해 블록이 생성될 때 마다 블록 당 5% 보상비율로 BAZ 적립
- ② 투자 진행후 투자자의 안전을 위해 사용
- ③ 배당 분배시 배당의 1%를 적립



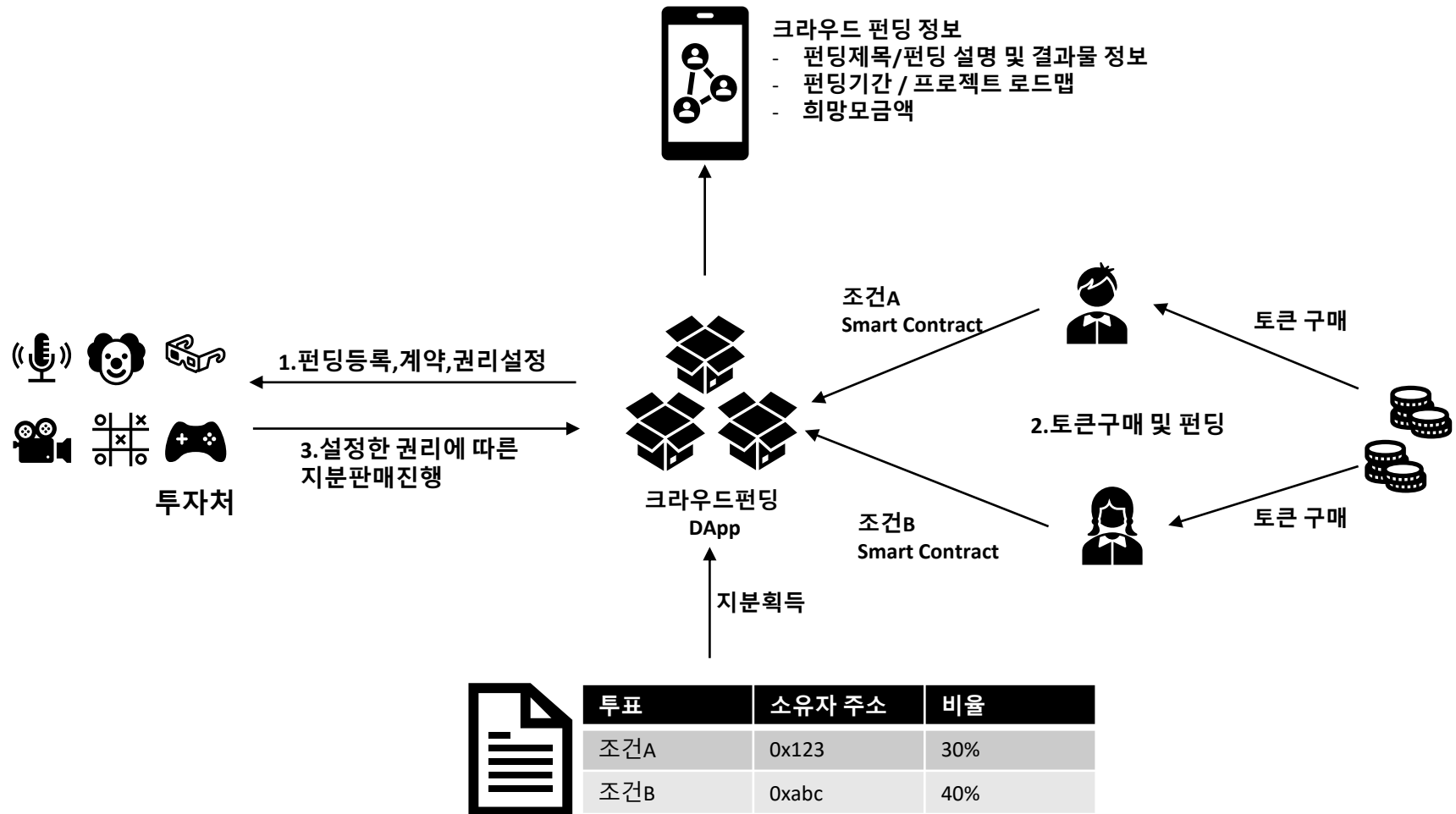


**BADIZ 플랫폼**

**05**

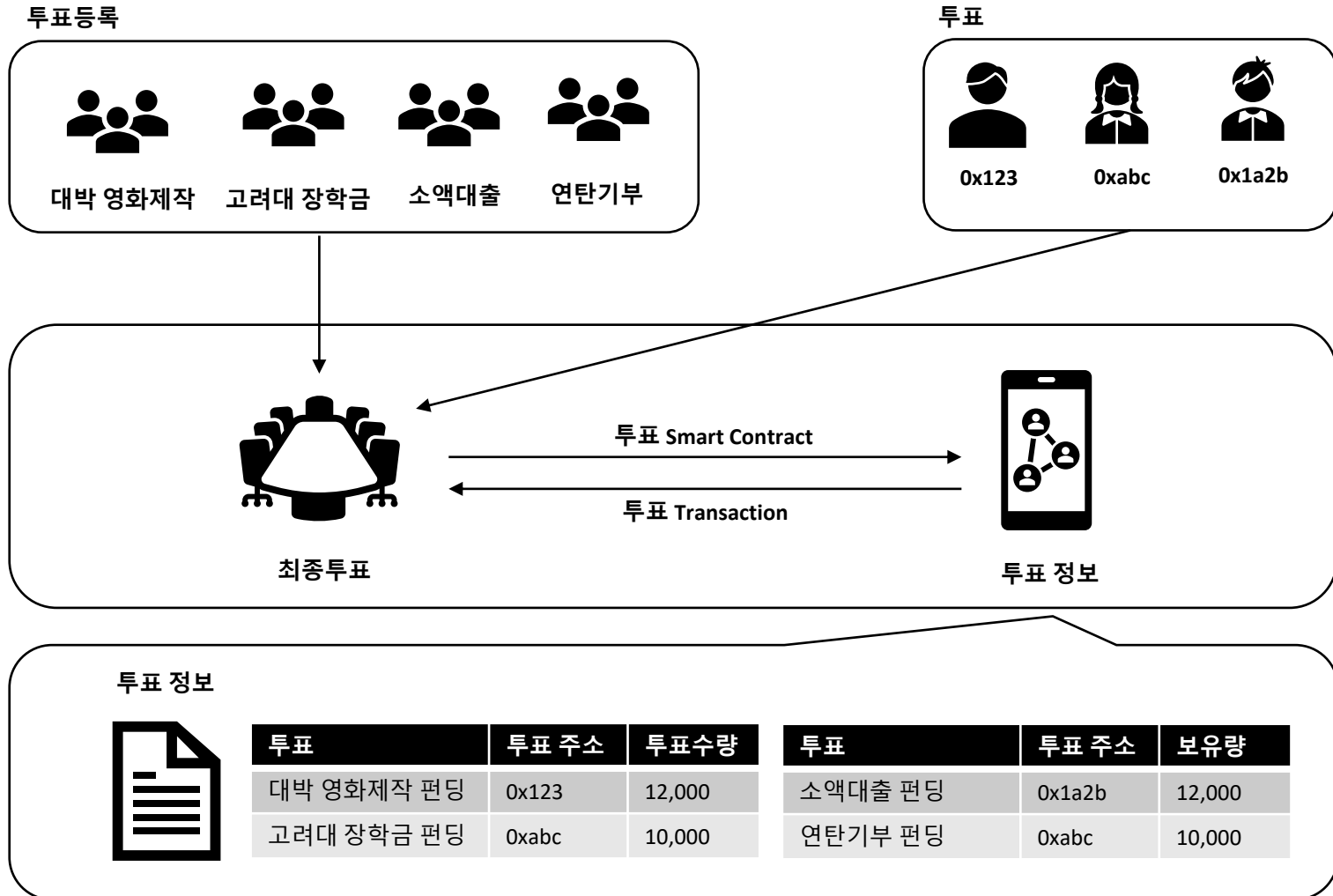
# 5-1. BADIZ Smart Contract

## 1) 클라우드 펀딩 Smart Contract

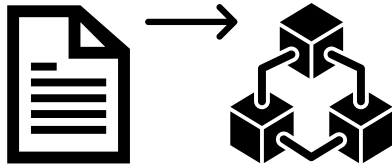


# 5-2. BADIZ Smart Contract

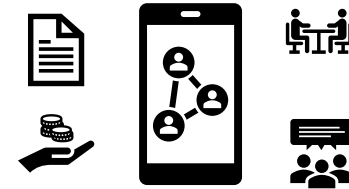
## 2) 투표 Smart Contract



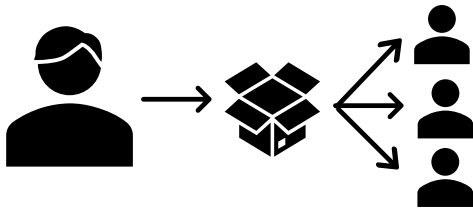
## 5-3. BADIZ Platform



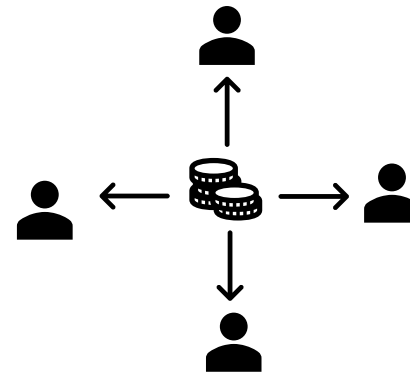
크라우드펀딩 등록



Smart contract



크라우드펀딩 과정 공개

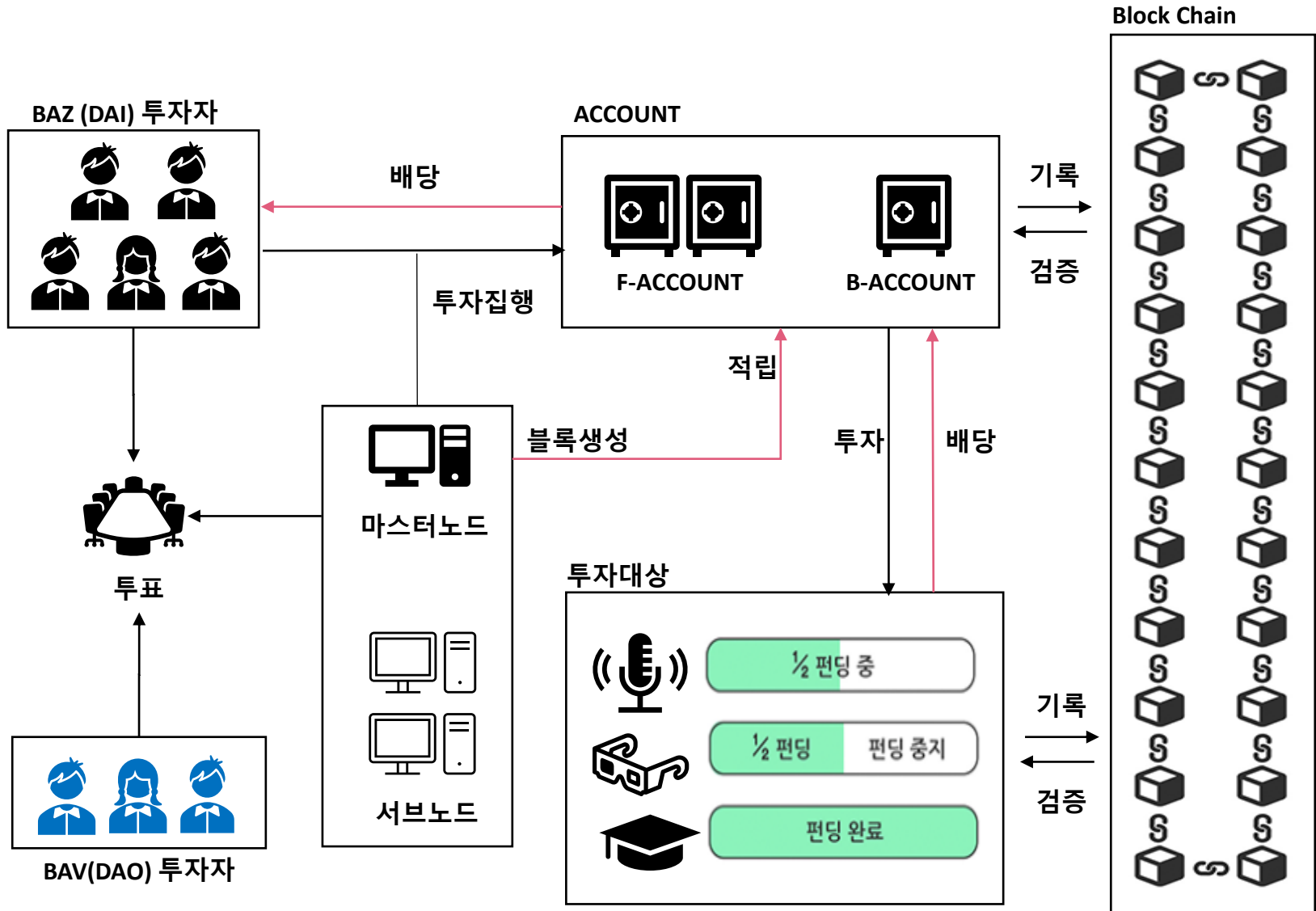


수익금 분배





# 5-4. BADIZ Platform Structure



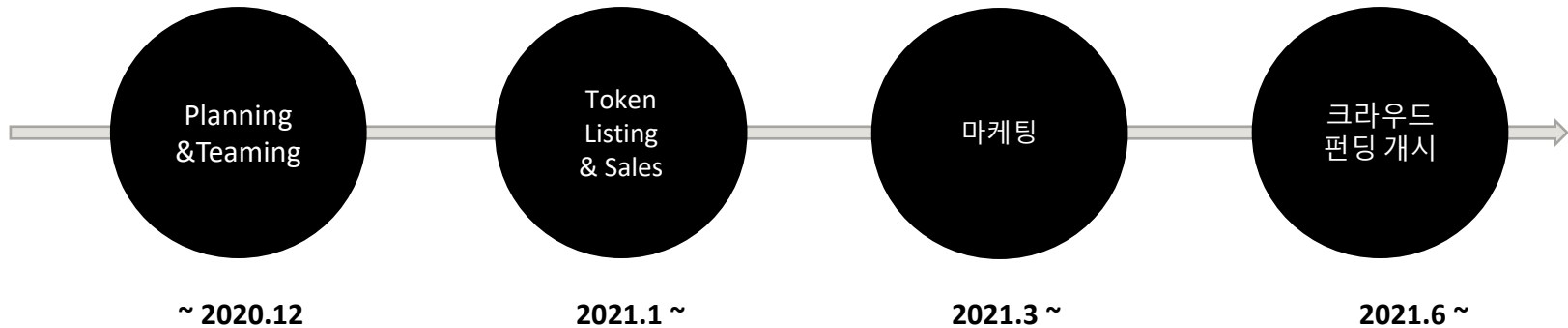


## BADIZ 로드맵

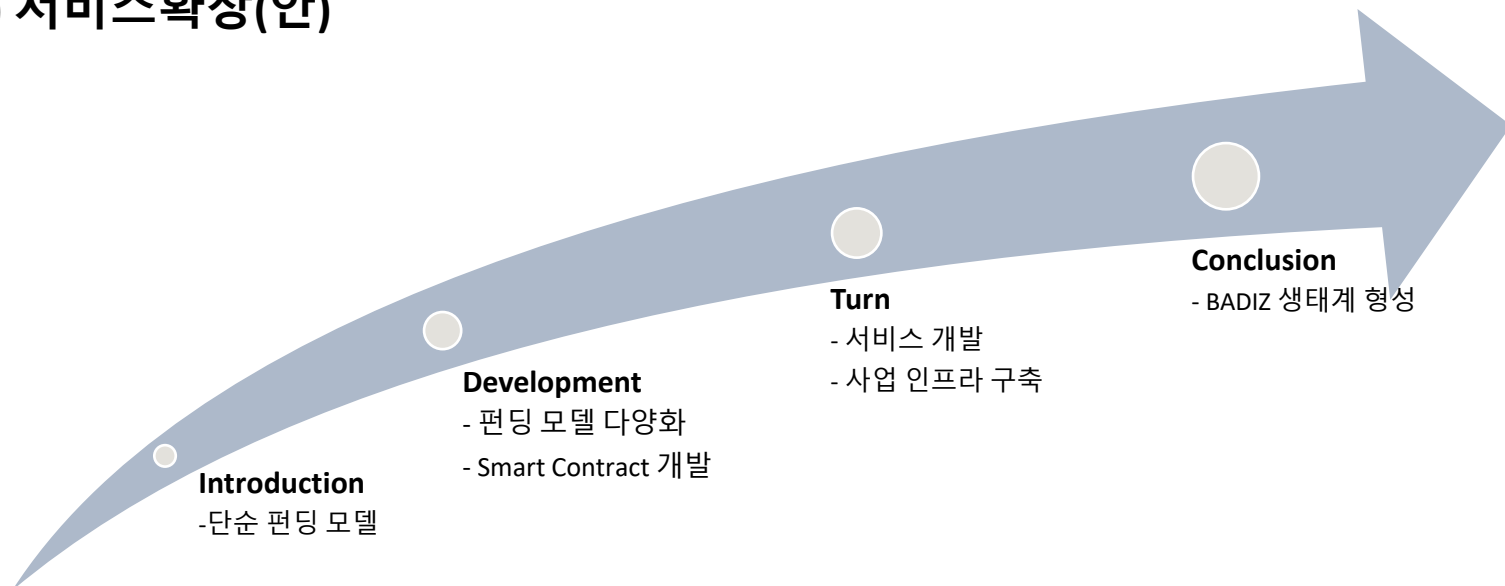
06

# 6. BADIZ 로드맵

## 1) 로드맵



## 2) 서비스확장(안)



부록

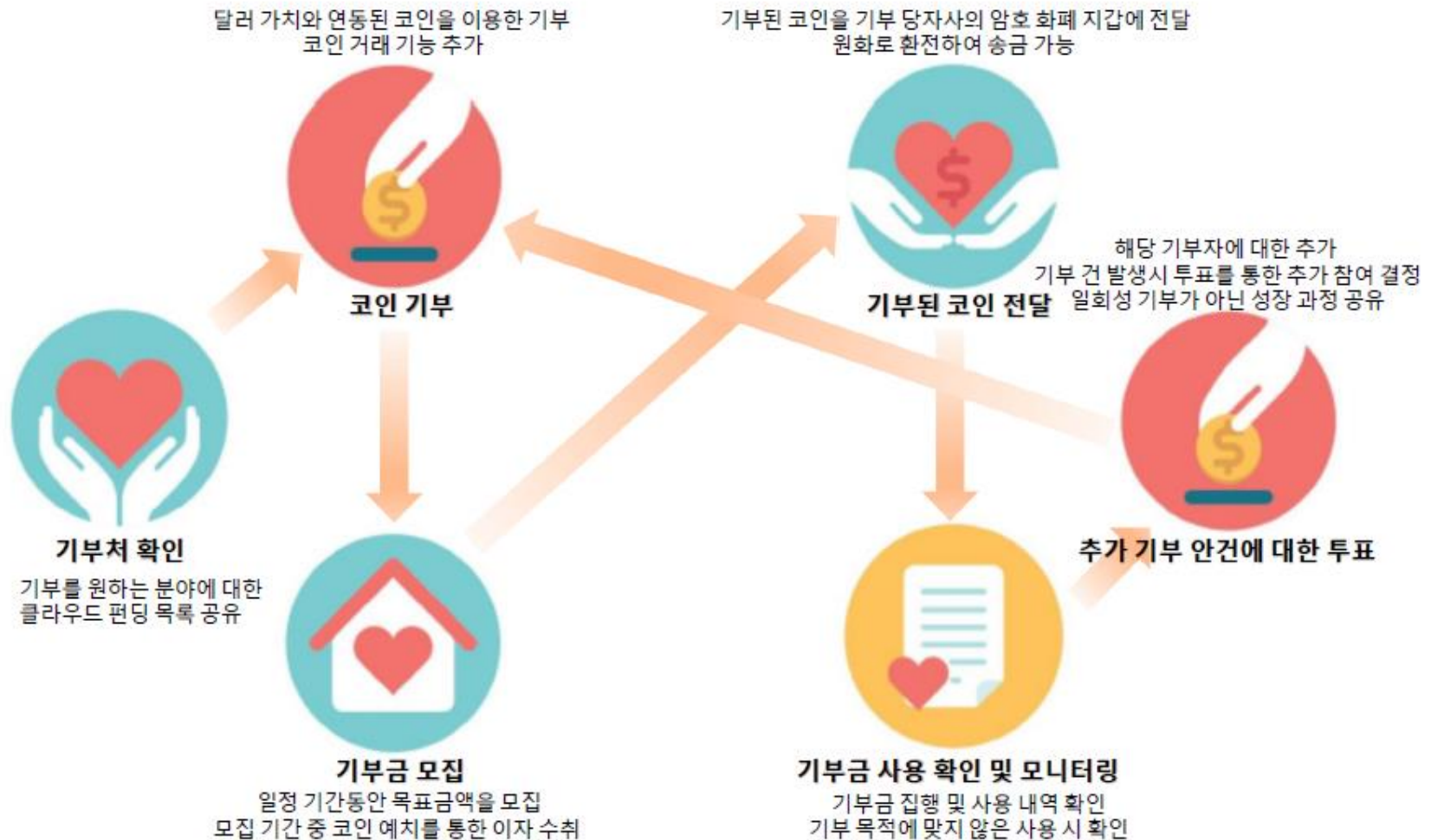
07

# 7-1. 장학금펀딩

	단순환급형(학생)	리워드환급형(학생)
순수기여형 (투자자)	<ul style="list-style-type: none"> <li>- 투자자는 거버넌스토큰 수취</li> <li>- 학생은 상환 의무 보유</li> </ul> <p>단순환급형(학생)은 일정 기간 이후 현금으로 상환</p>	<p>기부자는 거버넌스토큰 수취</p> <p>리워드환급형(학생)은 본인이 제공 가능한 용역, 서비스 등으로(토큰가치 환산) 상환</p> <p>플랫폼 내 제3의 기부자(거래형토큰 보유)가 보유 토큰을 활용하여 서비스 수취</p>
리워드기부형 (투자자)	<ul style="list-style-type: none"> <li>- 기부자는 거래형 토큰 수취 / 학생은 상환의무 보유</li> <li>- 기부자는 보유한 거래형 토큰으로 필요 서비스 수취</li> </ul>	<p>거래형 토큰(포인트,리워드)을 활용 직,간접 서비스 거래 가능</p>



## 7-2. 기부금펀딩



# 7-3. 시연(ICO)

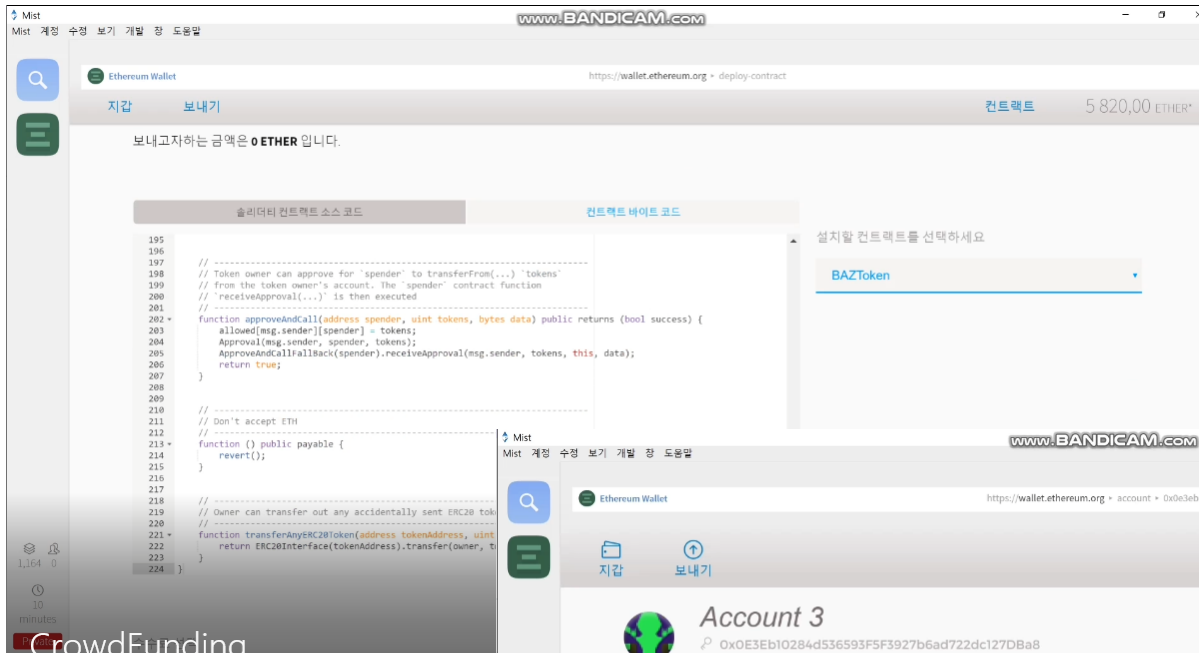
소스코드 : ICO.sol

```
1  pragma solidity ^0.4.18;
2
3
4  // -----
5  // Safe maths
6  // -----
7  library SafeMath {
8      function add(uint a, uint b) internal pure returns (uint c) {
9          c = a + b;
10         require(c >= a);
11     }
12     function sub(uint a, uint b) internal pure returns (uint c) {
13         require(b <= a);
14         c = a - b;
15     }
16     function mul(uint a, uint b) internal pure returns (uint c) {
17         c = a * b;
18         require(a == 0 || c / a == b);
19     }
20     function div(uint a, uint b) internal pure returns (uint c) {
21         require(b > 0);
22         c = a / b;
23     }
24 }
25
26
27 // -----
28 // ERC Token Standard #20 Interface
29 // https://github.com/ethereum/EIPs/blob/master/EIPS/eip-20.md
30 // -----
31 contract ERC20Interface {
32     function totalSupply() public constant returns (uint);
33     function balanceOf(address tokenOwner) public constant returns (uint balance);
34     function allowance(address tokenOwner, address spender) public constant returns (uint remaining);
35     function transfer(address to, uint tokens) public returns (bool success);
36     function approve(address spender, uint tokens) public returns (bool success);
37     function transferFrom(address from, address to, uint tokens) public returns (bool success);
38
39     event Transfer(address indexed from, address indexed to, uint tokens);
40     event Approval(address indexed tokenOwner, address indexed spender, uint tokens);
41 }
42
43
44 // -----
45 // Contract function to receive approval and execute function in one call
46 //
47 // Borrowed from MiniMeToken
48 // -----
49 contract ApproveAndCallFallBack {
50     function receiveApproval(address from, uint256 tokens, address token, bytes data) public;
51 }
52
53
54 // -----
55 // Owned contract
56 // -----
57 contract Owned {
```



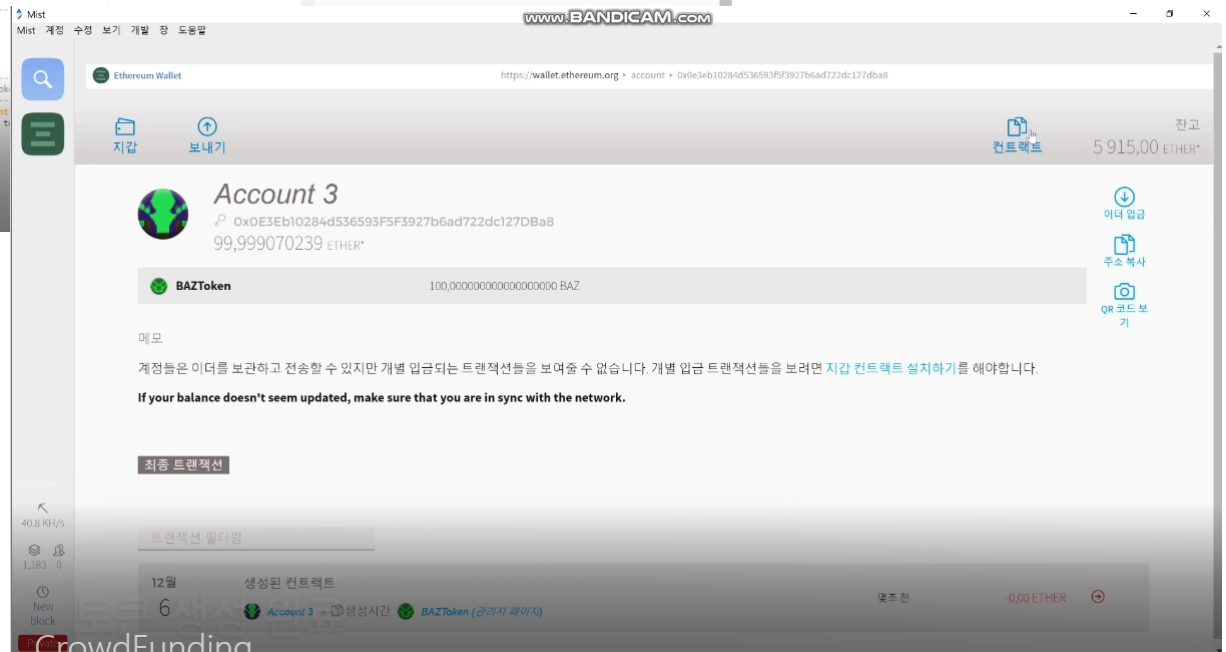
# 7-3. 시연(ICO)

## 1) BAZ 토큰생성



The screenshot shows the Mist Ethereum wallet interface. The top bar indicates the current balance is 5,820.00 ETH. The main area displays the 'Contracts' tab, showing the BAZToken contract code. The 'Deploy' button is highlighted, and the 'BAZToken' contract is selected in the dropdown menu. The code for the BAZToken contract is visible, including the 'approve' and 'transfer' functions.

## 2) BAZ 토큰생성완료



The screenshot shows the Mist Ethereum wallet interface after the BAZToken contract has been deployed. The 'Account 3' is displayed with the address 0x0E3Eb10284d536593F5F3927b6ad722dc127DBa8 and a balance of 99,999,070,239 BAZ tokens. The 'BAZToken' contract is also visible, showing a balance of 100,000,000,000,000,000 BAZ tokens. The interface includes a 'BAZToken' section with a balance of 100,000,000,000,000,000 BAZ tokens and a 'BAZToken' section with a balance of 100,000,000,000,000,000 BAZ tokens.





# 7-3. 시연(크라우드펀딩)

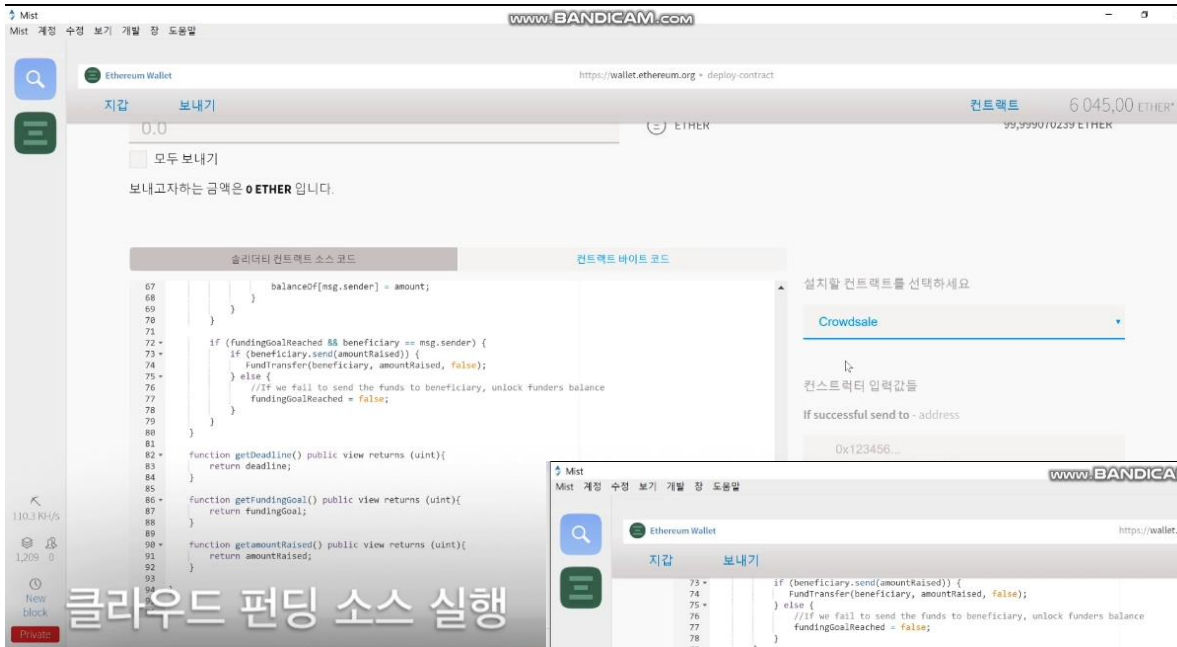
소스코드 : crowd\_funding.sol

```
1 pragma solidity ^0.4.18;
2
3 interface token {
4     function transfer(address receiver, uint amount) external;
5 }
6
7 contract Crowdsale {
8     address public beneficiary;
9     uint public fundingGoal;
10    uint public amountRaised;
11    uint public deadline;
12    uint public price;
13    token public tokenReward;
14    mapping(address => uint256) public balanceOf;
15    bool fundingGoalReached = false;
16    bool crowdsaleClosed = false;
17    uints public decimals;
18
19    event GoalReached(address recipient, uint totalAmountRaised);
20    event FundTransfer(address backer, uint amount, bool isContribution);
21
22    function Crowdsale(
23        address ifSuccessfulSendTo,
24        uint fundingGoalInEthers,
25        uint durationInMinutes,
26        uint etherCostOfEachToken,
27        address addressOfTokenUsedAsReward
28    ) public {
29        beneficiary = ifSuccessfulSendTo;
30        fundingGoal = fundingGoalInEthers * 1 ether;
31        deadline = now + durationInMinutes * 1 minutes;
32        price = etherCostOfEachToken * 1 ether;
33        tokenReward = token(addressOfTokenUsedAsReward);
34        decimals = 18;
35    }
36
37
38    function () payable public {
39        require(!crowdsaleClosed);
40        uint amount = msg.value;
41        balanceOf[msg.sender] += amount;
42        amountRaised += amount;
43        tokenReward.transfer(msg.sender, (amount / price) * 10**uint(decimals) );
44        FundTransfer(msg.sender, amount, true);
45    }
46
47    modifier afterDeadline() { if (now >= deadline) _; }
48
49
50    function checkGoalReached() public afterDeadline {
51        if (amountRaised >= fundingGoal){
52            fundingGoalReached = true;
53            GoalReached(beneficiary, amountRaised);
54        }
55        crowdsaleClosed = true;
56    }
57}
```

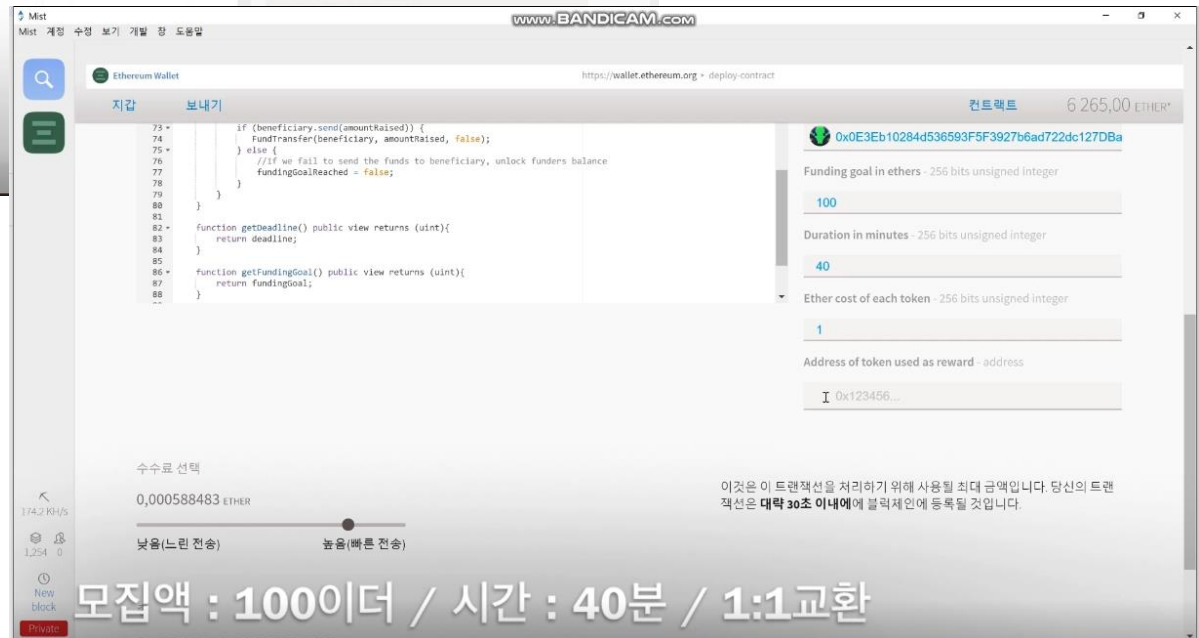


# 7-4. 시연(크라우드펀딩)

## 1) 크라우드펀딩 소스 실행

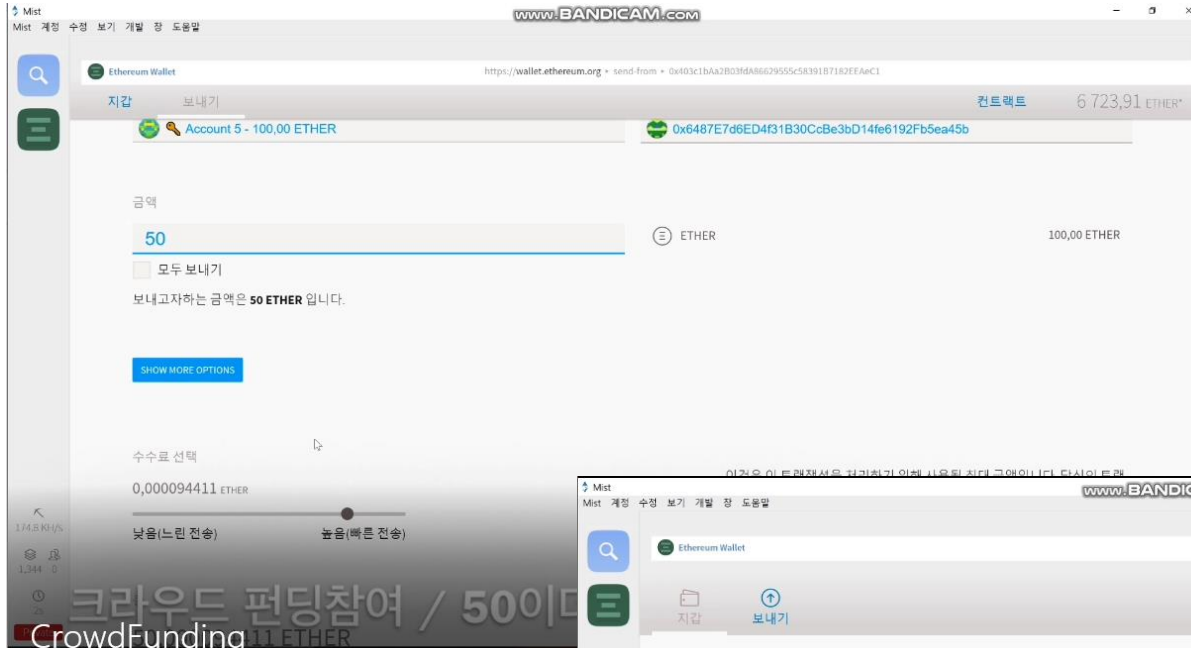


2) 크라우드펀드 모집  
모집액 : 100이더  
시간 : 40분

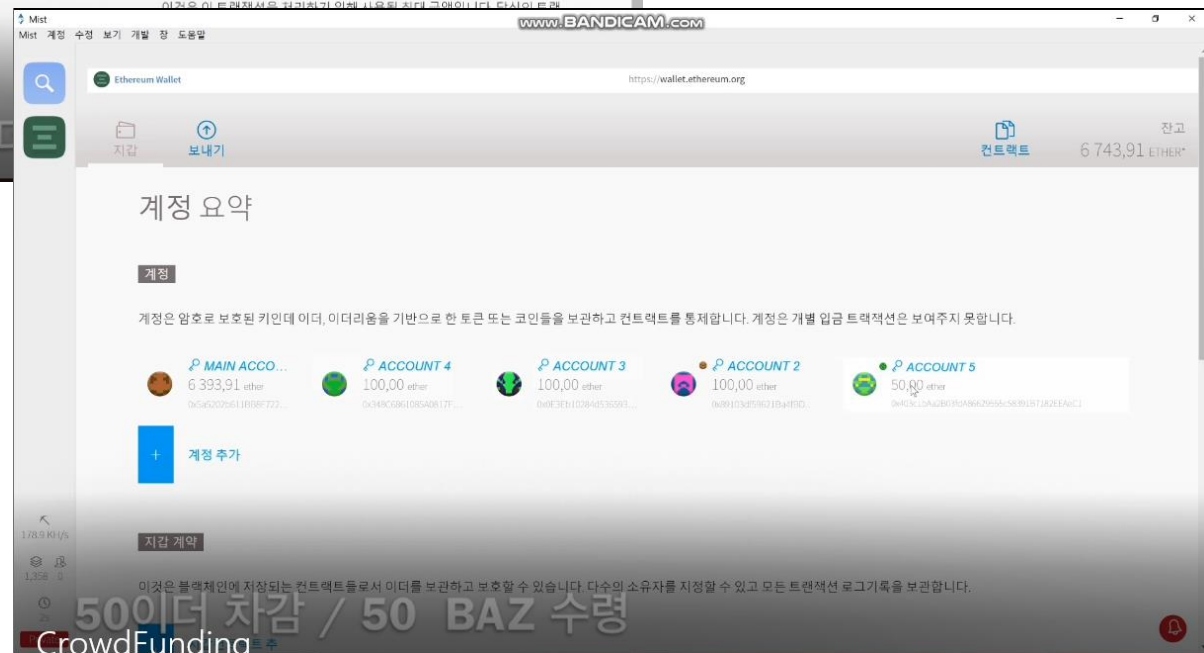


# 7-4. 시연(크라우드펀딩)

## 3) 크라우드펀딩 참여 ( 50 이더 )



## 4) 크라우드펀딩 완료 ( 50 이더 차감 50 BAZ수령 )



# 7-4. 시연(크라우드펀딩)

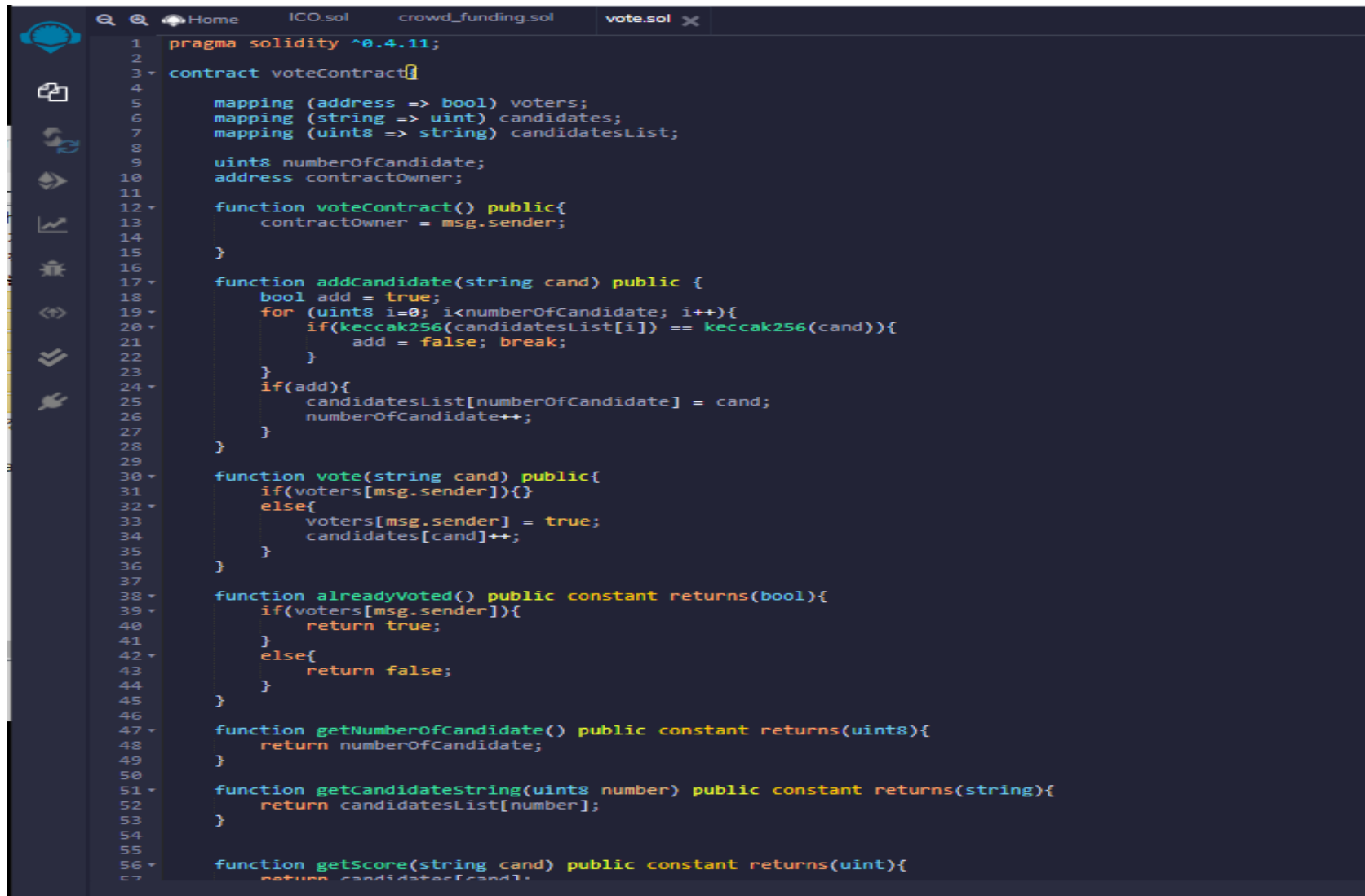
## 5) 크라우드펀드 모집완료

The screenshot displays the Ethereum Wallet interface within a web browser. The browser's address bar shows the URL `https://wallet.ethereum.org` and the account address `0x6487e7d6ed4f31b30cbe3bd14fe6192fb5ea45b`. The wallet's balance is shown as **6,988.91 ETH**. The main content area displays details for a crowdfunding campaign titled **CROWDSALE 6487**, which has a **100.00 ETH** goal. The interface includes sections for **Balance of**, **Address** (with a truncated address `0x123456...`), **Funding goal** (100,000,000,000,000,000), **Amount raised** (100,000,000,000,000,000), and **Price** (100,000,000,000,000,000). A large, semi-transparent watermark reading **펀딩 모집 완료** (Funding Completed) is overlaid on the bottom half of the screen. The bottom left corner features a **CrowdFunding** logo. The browser's status bar at the bottom shows network activity (182.6 KHz/s) and a **New block** notification.



# 7-5. 시연(투표)

소스코드 : vote.sol

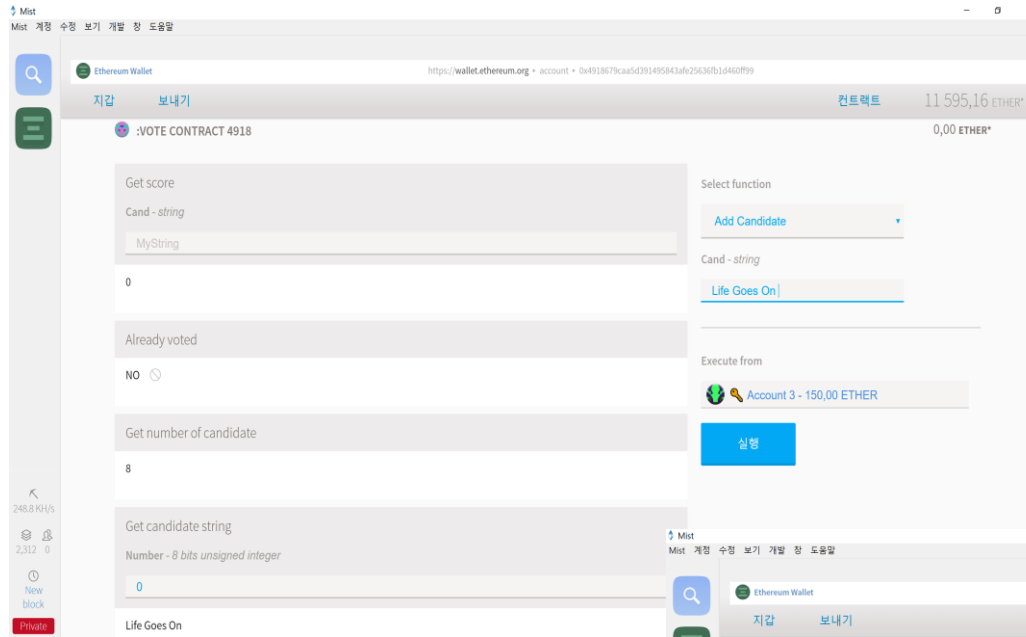


```
1 pragma solidity ^0.4.11;
2
3 contract voteContract{
4
5     mapping (address => bool) voters;
6     mapping (string => uint) candidates;
7     mapping (uint8 => string) candidatesList;
8
9     uint8 numberOfCandidate;
10    address contractOwner;
11
12    function voteContract() public{
13        contractOwner = msg.sender;
14    }
15
16    function addCandidate(string cand) public {
17        bool add = true;
18        for (uint8 i=0; i<numberOfCandidate; i++){
19            if(keccak256(candidatesList[i]) == keccak256(cand)){
20                add = false; break;
21            }
22        }
23        if(add){
24            candidatesList[numberOfCandidate] = cand;
25            numberOfCandidate++;
26        }
27    }
28
29    function vote(string cand) public{
30        if(voters[msg.sender]){
31            return;
32        }
33        voters[msg.sender] = true;
34        candidates[cand]++;
35    }
36
37    function alreadyVoted() public constant returns(bool){
38        if(voters[msg.sender]){
39            return true;
40        }
41        else{
42            return false;
43        }
44    }
45
46    function getNumberOfCandidate() public constant returns(uint8){
47        return numberOfCandidate;
48    }
49
50    function getCandidateString(uint8 number) public constant returns(string){
51        return candidatesList[number];
52    }
53
54    function getScore(string cand) public constant returns(uint){
55        return candidates[cand];
56    }
57 }
```

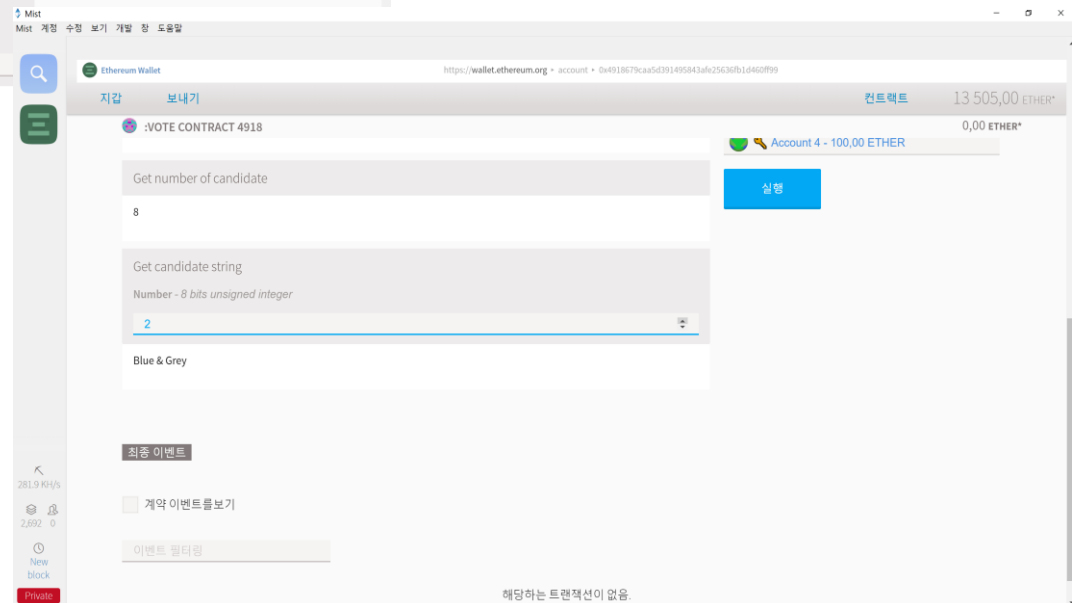


# 7-5. 시연(투표)

## 1) 투표(후보곡 추가)



## 1) 투표(후보곡 확인)





감사합니다