



# 스마트 컨트랙트를 활용한 아동 급식 결제와 토큰 기반 기부 시스템 제안

# 아동 급식 카드



결식아동 급식사업의 일환  
결식이 우려되는 아이들에게 발급하는 IC카드

표 VII-49 전자카드 지속적 이용 의향

구 분		사례수	예	아니오
		N	%	%
전 체		(248)	95.2	4.8
학교급	초등학생	(84)	92.9	7.1
	중학생	(97)	96.9	3.1
	고등학생	(67)	95.5	4.5
성별	남학생	(128)	96.1	3.9
	여학생	(120)	94.2	5.8
성별*학교급	초등학교 남학생	(38)	94.7	5.3
	초등학교 여학생	(46)	91.3	8.7
	중학교 남학생	(55)	96.4	3.6
	중학교 여학생	(42)	97.6	2.4
	고등학교 남학생	(35)	97.1	2.9
	고등학교 여학생	(32)	93.8	6.3
지역아동센터 이용	이용	(128)	91.4	8.6
	미이용	(120)	99.2	0.8

서울시 아동급식평가 및 전달체계 구축방안연구 (2010)

지속적인 이용  
'원한다'는 응답 **95.2%**



사용하는 아동들에게  
**긍정적인 인식**

# 아이디어 동기



담당 공무원이  
허위로 아동을 등록  
1억 5천만원을 횡령



지자체별로  
위탁 업체를 통해 운영  
폐쇄적인 구조

## 해결 방안

블록체인 스마트 컨트랙트를 활용하여 거래에 대한 정보를  
통합된 DB에 저장하여 투명성을 보장

# 근본적인 문제점

## 해결 방안



부족한 지원 금액

적은 가맹점

타인의 사용이 가능

심리적 낙인감

전용 리더기  
설치 필요

✓ 블록체인 스마트 계약을 사용하여  
신뢰적인 기부 시스템 적용

**할인된 금액**으로 아동에게 상품을 제공

**우수한 가맹점 혜택**을 통해 더 많은  
가맹점 유치

✓ 스마트폰 앱을 통해 인증과 QR 코드 형식의  
결제 방식을 사용

가맹점의 정보와 최신화 등  
**추가적인 서비스를 제공이 용이**

# 개발 환경

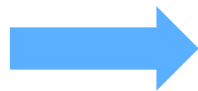
## HyperLedger Fabric



허가형 네트워크

거래의 기밀 유지

암호화폐 불필요



제안하는 아동급식카드에 적합

# 개발 환경



{ **REST:API** }

## 블록체인 네트워크를 노출

- 멀티플랫폼 지원 및 연동



## 가맹점을 가정한 웹

- 상품등록, QR코드 생성



## 아동이 사용하는 앱

- QR코드를 사용한 결제
- 잔여 포인트와 구매 내역 조회

# 프로토타입 구현



1. 개발환경을 제공하는 HyperLedger Composer 사용



2. 비즈니스 네트워크 작성 - 모델링, 액세스 제어, 체인코드



3. HyperLedger Fabric 배포 및 REST 서버 실행



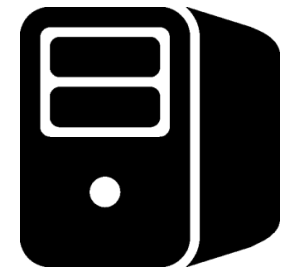
4. 웹과 앱 연동

# 동작 구성



아동

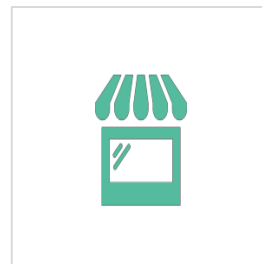
이메일 인증을 통한  
회원가입 승인



서버

1. 상품의 QR코드를 스캔
2. 상품, 금액 확인 후 지문 결제

3. 상품 지급



가맹점

1. 상품 등록

2. QR 코드 생성



# 구매 동작 구성



## 구매 시 체인코드

```
async function buyCommodity(buy){
  var buyer = buy.member
  var seller = buy.commodity.seller;
  var commodityPrice = buy.commodity.price;
  var fundraising = buy.fundraising;
  var discount = 1;

  var participantRegistry = await getParticipantRegistry('org.sample' + '.Member');
  var participants = await participantRegistry.getAll();
  if(buy.member.balance < buy.commodity.price)
    throw new Error('Insufficient balance');

  if(seller.token/participants.length >= 0.2)
    discount = 0.8;
  if(fundraising.dpoint > 5000){
    buyer.balance -= commodityPrice * discount;
    fundraising.dpoint -= commodityPrice * (1 - discount);
  } else {
    buyer.balance -= commodityPrice;
  }
  seller.balance += commodityPrice;

  return getAssetRegistry('org.sample' + '.Commodity').then(function(assetRegistry) {
    return assetRegistry.update(buy.commodity);
  }).then(function() {
    return getParticipantRegistry('org.sample' + '.Member').then(function(memberRegistry) {
      return memberRegistry.update(buyer);
    }).then(function() {
      return getParticipantRegistry('org.sample' + '.Seller').then(function(memberRegistry) {
        return memberRegistry.update(seller);
      }).then(function() {
        return getAssetRegistry('org.sample' + '.Fundraising').then(function(assetRegistry) {
          return assetRegistry.update(fundraising);
        });
      });
    });
  });
}
```

1. 아동이 월마다 만족하는 가맹점에 토큰 부여
2. 기부자의 기부
3. 가맹점 토큰 보유가 상위 20% 이상일 경우 아동이 구매 시 20%의 할인 적용되고 할인된 금액은 기부금에서 차감

# 테스트 이미지



Participant registry for org.sample.Member

ID	Data
Member1	{ "\$class": "org.sample.Member", "name": "", "region": "a", }
Member2	{ "\$class": "org.sample.Member", "name": "Member", "region": "a", "limit": 12000, "supports": 2, "email": "Member2", "balance": 8800, "token": 1 }

Participant registry for org.sample.Seller

ID	Data
Seller1	{ "\$class": "org.sample.Seller", "organisation": "a", "email": "Seller1", "balance": 28000, "token": 1 }

구매자 잔액 : 8800  
판매자 잔액 : 28000  
모금 금액 : 9800

Asset registry for org.sample.Fundraising

ID	Data
fundraising	{ "\$class": "org.sample.Fundraising", "fundraisingId": "fundraising", "description": "", "dpoint": 9800 }

Collapse

# 테스트 이미지



Asset registry for org.sample.Commodity

ID	Data
commodity	<pre>{   "\$class": "org.sample.Commodity",   "commodityId": "commodity",   "description": "",   "price": 1000,   "qr": "",   "seller": "resource:org.sample.Seller#Seller1" }</pre>

Collapse

1000원짜리 물품을 선택

Submit Transaction

Transaction Type: Buy

JSON Data Preview

```
1 {
2   "$class": "org.sample.Buy",
3   "commodity": "resource:org.sample.Commodity#commodity",
4   "member": "resource:org.sample.Member#Member2",
5   "fundraising": "resource:org.sample.Fundraising#fundraising"
6 }
```

☐ Optional Properties

Just need quick test data? [Generate Random Data](#) Cancel Submit

구매 트랜잭션 실행

# 테스트 이미지



<div>Member2</div> <pre>{   "\$class": "org.sample.Member",   "name": "Member",   "region": "a",   "limit": 12000,   "supports": 2,   "email": "Member2",   "balance": 8000,   "token": 1 }</pre> <div>Collapse</div>	<div>Participant registry for org.sample.Seller</div> <table><thead><tr><th>ID</th><th>Data</th></tr></thead><tbody><tr><td>Seller1</td><td><pre>{   "\$class": "org.sample.Seller",   "organisation": "a",   "email": "Seller1",   "balance": 29000,   "token": 1 }</pre></td></tr></tbody></table>	ID	Data	Seller1	<pre>{   "\$class": "org.sample.Seller",   "organisation": "a",   "email": "Seller1",   "balance": 29000,   "token": 1 }</pre>
ID	Data				
Seller1	<pre>{   "\$class": "org.sample.Seller",   "organisation": "a",   "email": "Seller1",   "balance": 29000,   "token": 1 }</pre>				
<div>구매자 잔액 : 8800 -&gt; 8000 판매자 잔액 : 28000 -&gt; 29000 모금 금액 : 9800 -&gt; 9600</div>	<div>Asset registry for org.sample.Fundraising</div> <table><thead><tr><th>ID</th><th>Data</th></tr></thead><tbody><tr><td>fundraising</td><td><pre>{   "\$class": "org.sample.Fundraising",   "fundraisingId": "fundraising",   "description": "",   "dpoint": 9600 }</pre></td></tr></tbody></table> <div>Colla</div>	ID	Data	fundraising	<pre>{   "\$class": "org.sample.Fundraising",   "fundraisingId": "fundraising",   "description": "",   "dpoint": 9600 }</pre>
ID	Data				
fundraising	<pre>{   "\$class": "org.sample.Fundraising",   "fundraisingId": "fundraising",   "description": "",   "dpoint": 9600 }</pre>				

# 구현한 웹 동작 원리

로그인

ID :

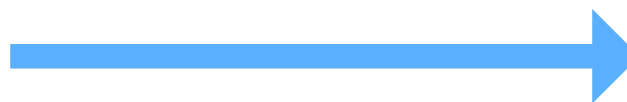
PW :



amdjd0704@gmail.com

상품 이름

상품 가격



상품명 볶음밥 가격 5000

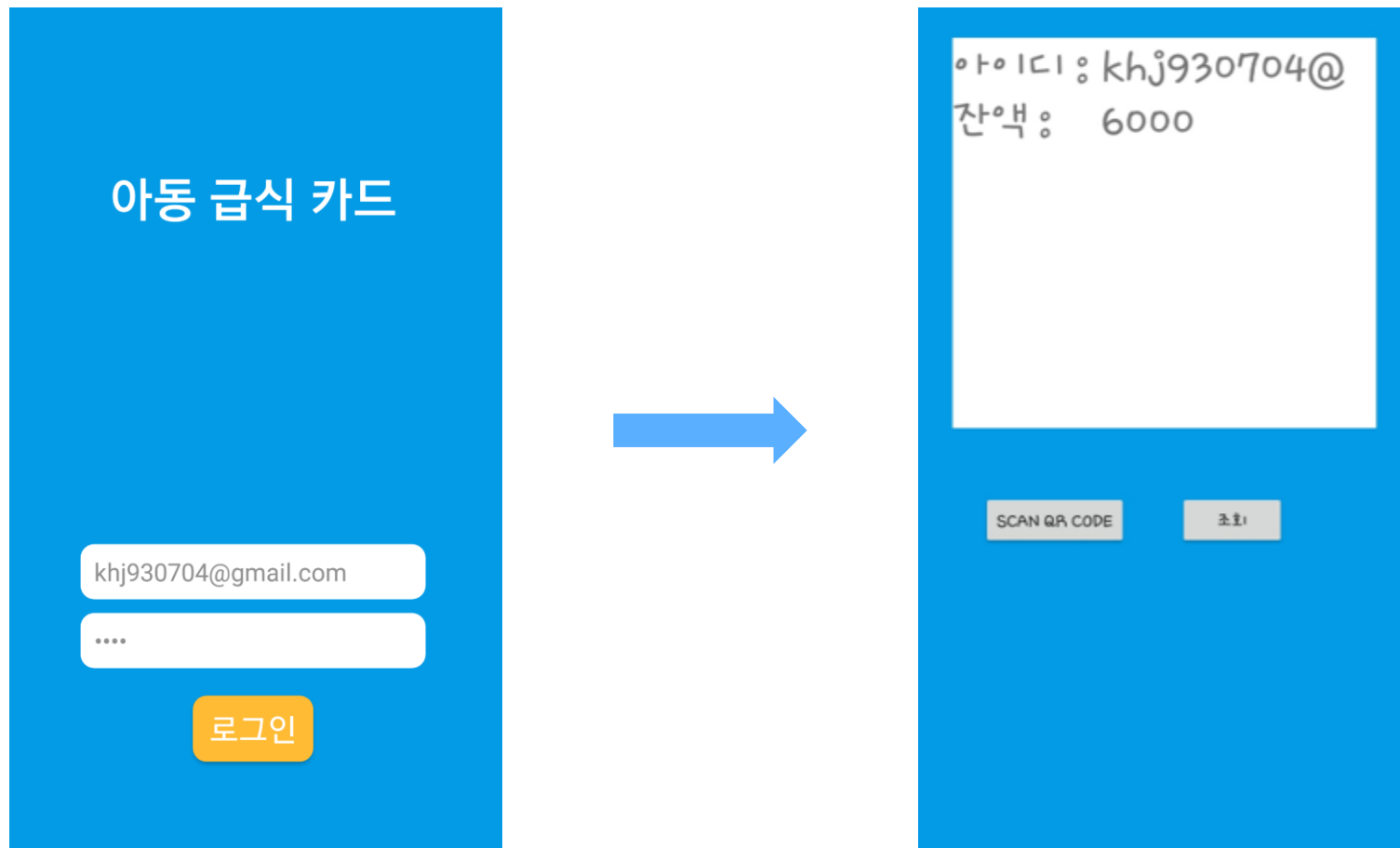


상품명 자장면 가격 5000

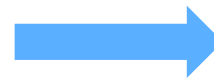
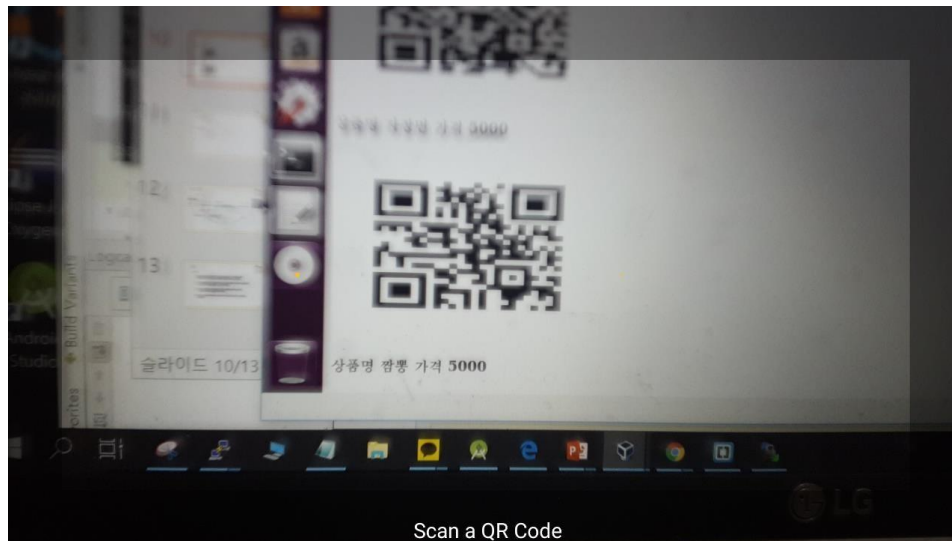


상품명 짬뽕 가격 5000

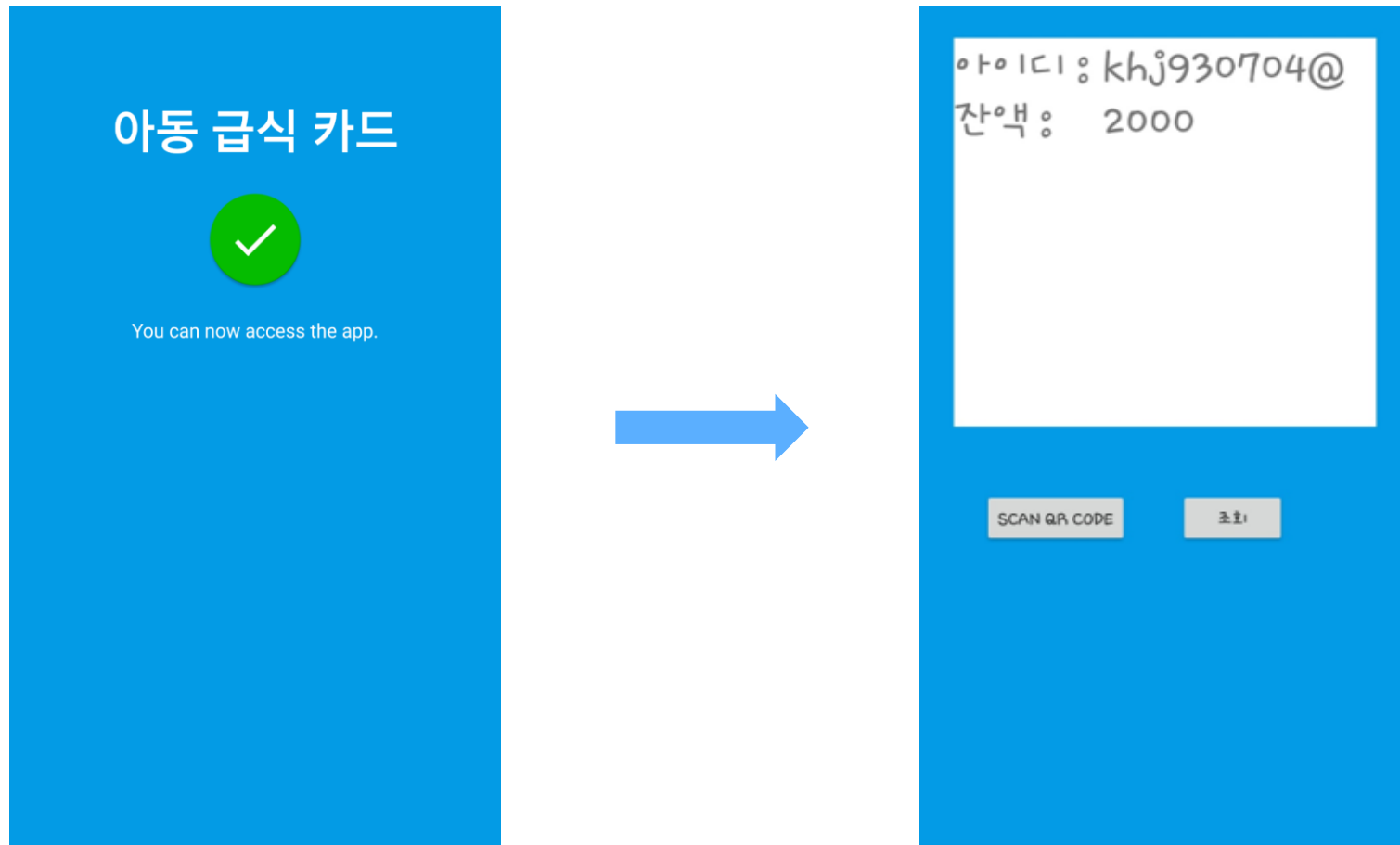
# 구현한 앱 동작 원리



# 구현한 앱 동작 원리



# 구현한 앱 동작 원리





# 기존 방식과의 비교

급식 카드		급식 앱
타인이 사용 가능	사용자	본인만 사용 가능
있음	노출 위험	없음(어플 숨기기)
고정적	지원 금액	기부 시스템으로 인해 유동적
가맹점의 별도 설치 필요	리더기 설치	별도 설치 필요 없음
폐쇄된 구조로 악용 가능성이 많음	악용 위험성	블록체인의 무결성을 이용해 악용 가능성이 적음

# 결론

아동급식카드의 큰 사건 이슈로 여러 문제점이 부각되었고  
**스마트 컨트랙트**와 **스마트 폰 앱을 사용**하여 해결 방안을 제시함.

제안하는 방법에서 블록체인의 투명성과 무결성을 활용해서  
기부 시스템을 추가하고 **충분한 금액 지원 및 가맹점 확대**와  
**기존 제도의 신뢰성 회복**에 도움이 될 것이라고 생각함.

# 결론

앱을 통해 구동되므로 업데이트가 간단하고  
아동에게 최신의 정보를 전달하는 것이 편리함.

제안하는 방법을 통해서 아동들이  
만족할 수 있는 급식의 제공이 이뤄졌으면 함.

감사합니다