

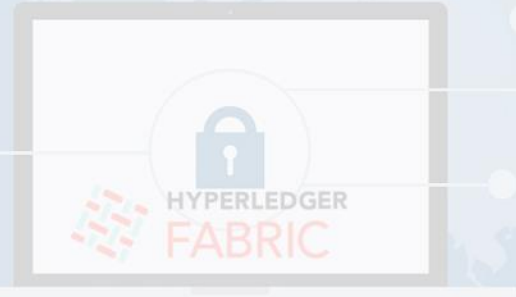
# 졸업프로젝트 결과

보.uckss



# 목 차

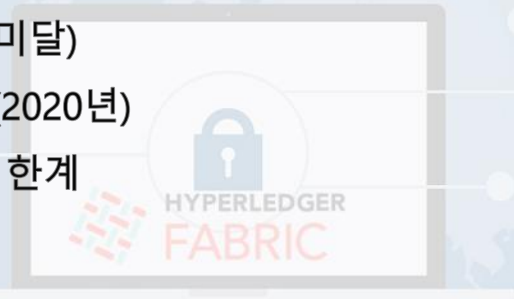
1. 프로젝트 소개
  - a. 프로젝트의 필요성 및 목적
  - b. 프로젝트 진행 계획
  - c. 설계 진행 과정
  - d. 개발 진행 과정
2. 프로젝트 전체 구조도
3. 최종 결과물 시연(영상)
4. 최종 결과물 추가설명
  - a. 주요기능
  - b. 기존 전자투표와 차별점
  - c. 퍼블릭 블록체인 기반 전자투표와 차별점
  - d. 주요 보안과제 달성사항
5. 향후 발전 방향
6. 프로젝트를 통해 느낀점
7. QnA



# 프로젝트 소개

## a. 프로젝트의 필요성 및 목적

- 2017~2019년 연속 총학생회 선거 부결 (투표율 50% 미달)
- 대부분 단과대 또한 비상대체위원회가 학생회 대체중(2020년)
- 코로나19 상황으로 오프라인 선거만으로는 투표 독려 한계
- 기존 온라인 전자투표 시스템의 문제점  
: 보안성 (무결성 검증 문제), 신뢰성(부정선거 가능성)
- **개인의 비밀정보를 보호하고, 기록의 조작을 불가능하게 만드는 블록체인 기술을 전자투표시스템에 접목하여 이러한 문제를 해결**



# 프로젝트 소개

## a. 프로젝트의 필요성 및 목적

### - 컨소시움 블록체인



검증자 노드들을 허가를  
통하여 네트워크에  
참여시키고 사용자는  
자유롭게 참여하는  
방식으로써 한정된 탈  
중앙집중식을 구현하게 된다.

#### <장점>

- \*신원관리 및 개인정보 보호 용이(책임 스푸핑 방지, 트랜잭션 발행자 익명화)
- \*재생 공격 대처 가능
- \*액세스 제어 용이(스마트계약 실행하는 사용자 제한)
- \*공유 원장 다중 관리(멀티 채널)

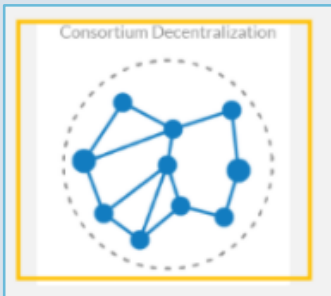


# 프로젝트 소개

## a. 프로젝트의 필요성 및 목적

- 채택한 블록체인 모델 : hyperledger-fabric

컨소시엄 블록체  
인



기존 블록체인  
과 차별화된 구  
조



Pluggable  
Consensus



모듈성



# 프로젝트 소개

## a. 프로젝트의 필요성 및 목적

- 채택한 블록체인 모델 : hyperledger-fabric

### 컨소시움 블록체인

네트워크에 접속하는  
사용자를 식별하여 중  
복투표를 방지한다.

### 기존 블록체인과 차별화된 구조

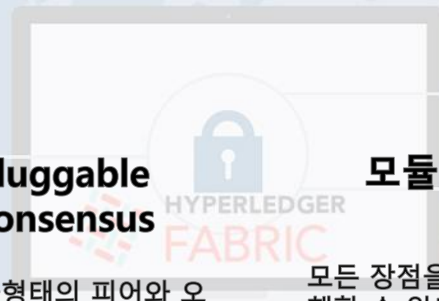
execute-order-validate  
구조를 도입함  
트랜잭션 흐름을  
세 단계로 분리하여  
탄력성, 유연성, 확장성,  
성능 및 기밀성 문제를  
해결

### Pluggable Consensus

모듈형태의 피어와 오더러를 통해서 RAFT, kafka 합의 프로토콜을 수행

### 모듈성

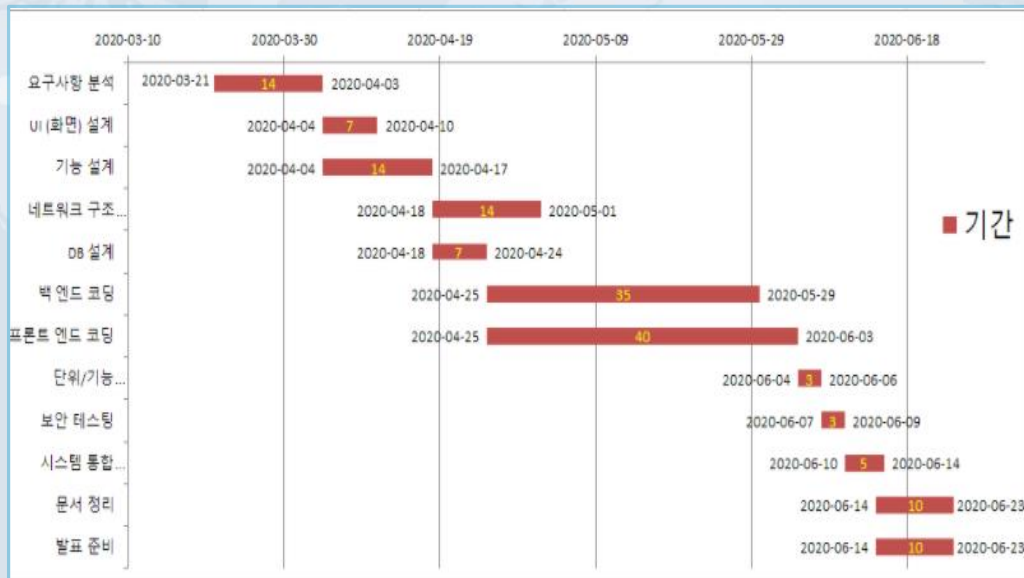
모든 장점을 동시에 수행할 수 있도록 각각의 기능을 도커 컨테이너 모듈로 구성할 수 있음



# 프로젝트 소개

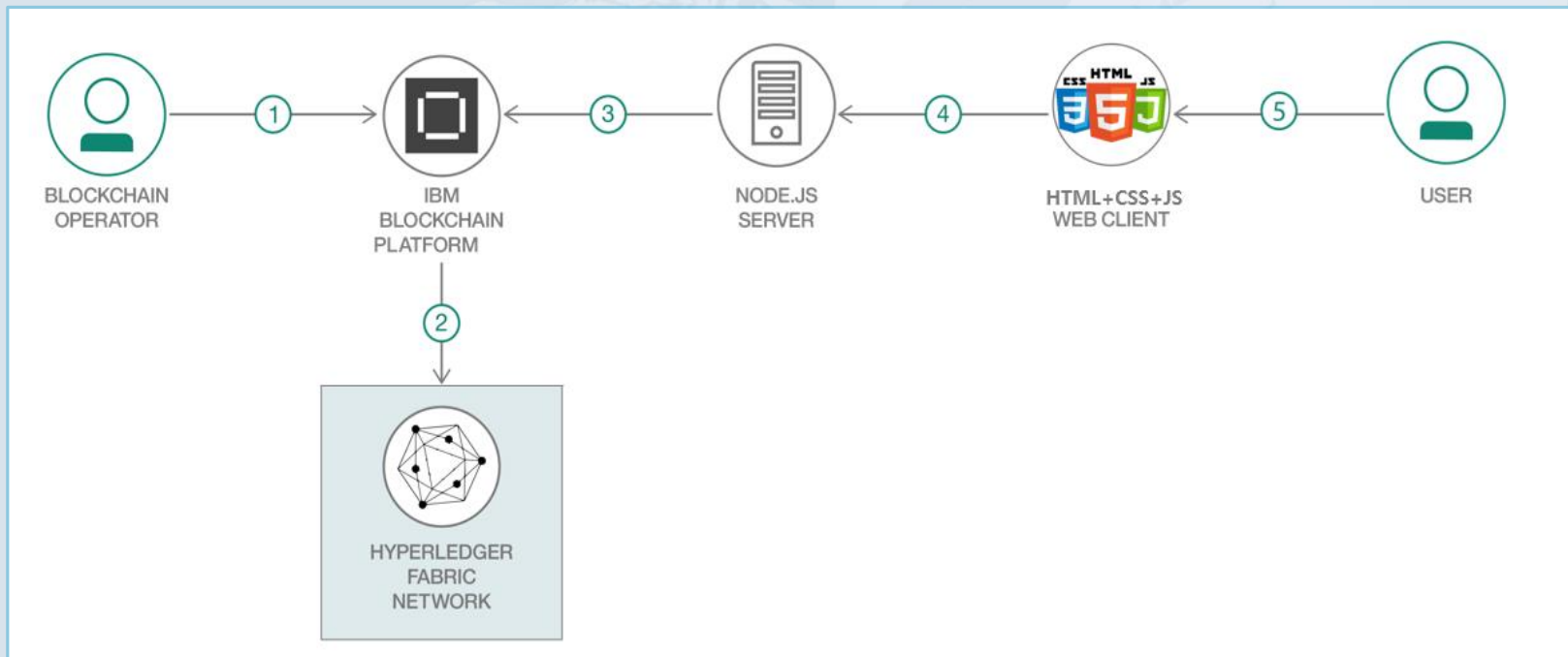
## b. 프로젝트 진행 계획

작업 이름	시작 일자	기간	종료 일자
요구사항 분석	2020년 3월 21일	14	2020. 4. 4
UI (화면) 설계	2020년 4월 5일	6	2020. 4. 11
기능(체인코드) 설계	2020년 4월 12일	13	2020. 4. 25
네트워크 구조 설계	2020년 4월 12일	13	2020. 4. 25
DB 설계	2020년 4월 12일	6	2020. 4. 18
백 엔드 코딩	2020년 4월 26일	34	2020. 5. 30
프론트 엔드 코딩	2020년 4월 26일	37	2020. 6. 2
단위/기능 테스트	2020년 6월 3일	3	2020. 6. 6
보안 테스트	2020년 6월 7일	2	2020. 6. 9
시스템 통합 테스트	2020년 6월 10일	6	2020. 6. 16
문서 정리	2020년 6월 17일	8	2020. 6. 25
발표 준비	2020년 6월 17일	8	2020. 6. 25



# 프로젝트 소개

## b. 프로젝트 진행 계획





# 프로젝트 소개

## c. 설계 진행과정

### (1) 요구사항명세서(기능설계)

프로젝트 명	[BL,UCKSS] 블록체인 투표 시스템	프로젝트 기간	2019.12 -
문서 명	[BL,UCKSS]요구사항 명세서	버전	2.1.0

### 3. 요구사항 분석

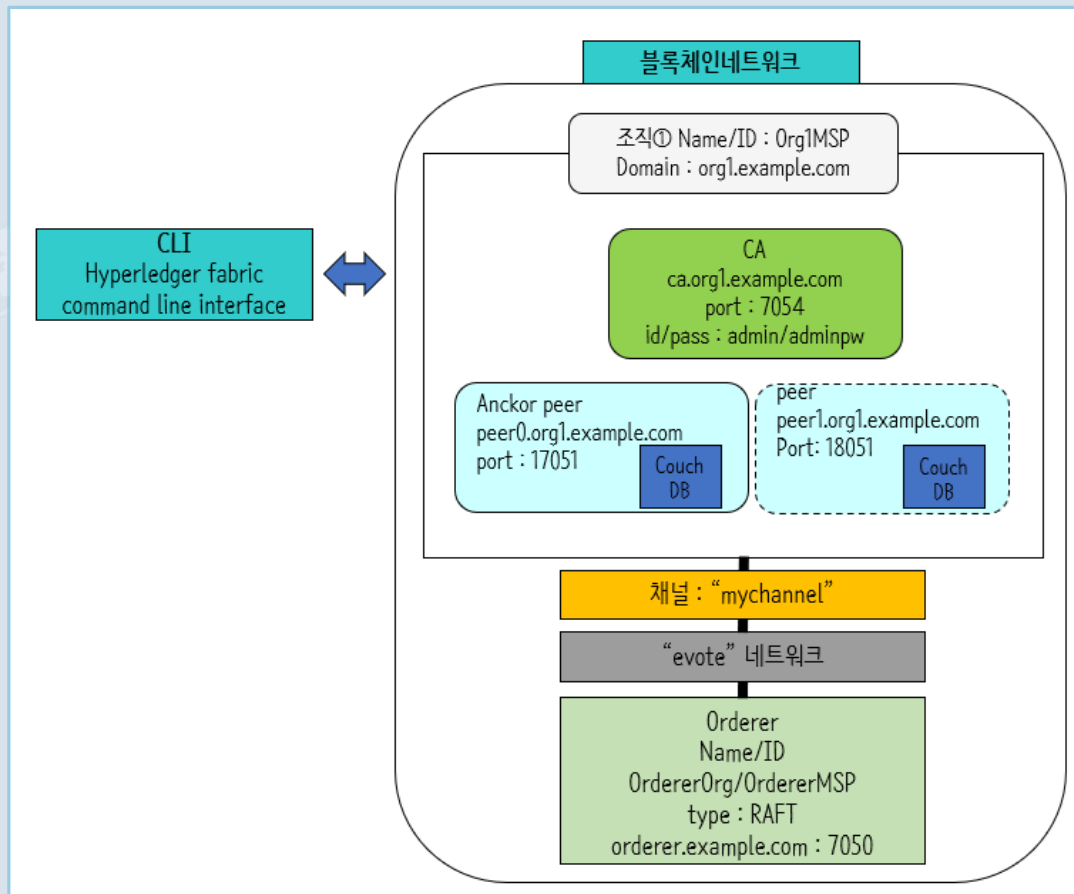
#### 3.1 고객 기능 요구사항

구분	상세	비고
로그인	<ul style="list-style-type: none"> <li>- 투표를 위해 로그인을 할 경우 자신의 학번과 비밀번호를 학교 포털과 동일하게 입력하여 로그인할 수 있도록 한다. 학번과 비밀번호가 일치하지 않는 경우 적절한 경고창(오류 메시지)을 출력한다.</li> <li>- 투표자는 학교 시스템(SSO)과 연동되었다면, 기존의 학교 서버에서 사전에 해당 인원이 투표가 가능한 학생인지 확인해주어야 한다. 예) 휴학생, 퇴학생, 졸업생, 재학생 등의 여부 확인 필요</li> </ul>	
투표	<ul style="list-style-type: none"> <li>- 투표하기 전에 항상 유의사항을(대리 투표 금지, 개인정보 활용 등) 숙지하도록 알림창을 띄우고 유권자의 동의를 구한다.</li> <li>- 자신이 투표할 수 있는 선거(총학생회/단과대 학생회)에 해당하는 입후보의 목록과 선거 공약을 확인한 후 투표를 진행한다. 모든 선거의 선택함에 기권이 존재한다.</li> <li>- 투표기간이 끝났을 경우 더 이상 투표할 수 없게 제한된다.</li> <li>- 투표를 실시간으로 집계하여 마무리 후 결과를 발표한다. (당선여부, 투표율 미달로 인한 공지 등)</li> <li>- 집계 결과가 변조되지 않은 결과임을 검증할 수 있어야 하므로, 자신의 투표값을 확인할 수 있다.</li> </ul>	
선거 결과 확인	<ul style="list-style-type: none"> <li>- 종료된 선거의 결과를 확인한다. 투표율이 40% 이상이라면 각 후보별 투표율을 출력하고, 40%미만일 경우 투표율 미달로 인해 선거가 무산되었음을 알리고, 총 투표율을 출력한다.</li> <li>- 관리자는 진행중인 선거의 실시간 투표율을 확인할 수 있다.</li> </ul>	
내 투표 확인	<ul style="list-style-type: none"> <li>- 선거 별로 자신이 어떤 후보에게 투표하였는지 확인할 수 있다.</li> </ul>	
거의하기	<ul style="list-style-type: none"> <li>- 서비스에서 상시에 언제라도 선거관리위원회</li> </ul>	

# 프로젝트 소개

## c. 설계 진행과정

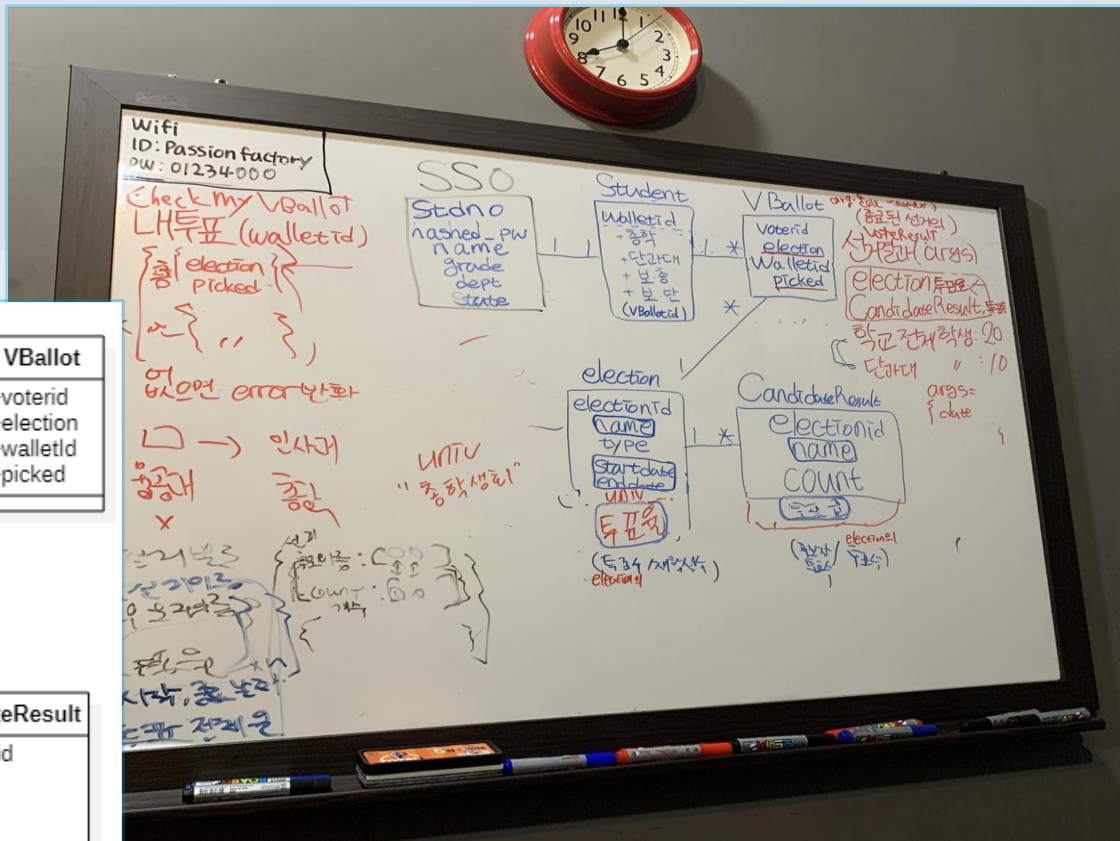
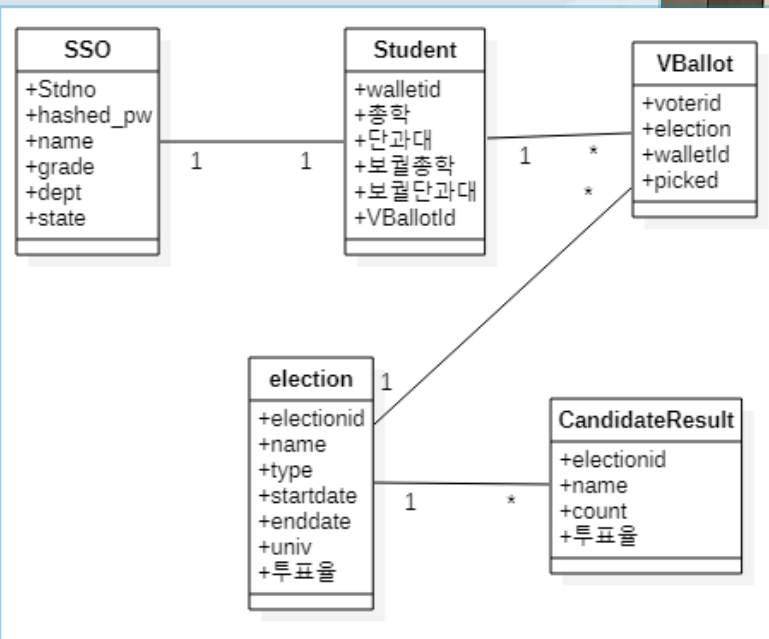
### (2) 네트워크구조설계



# 프로젝트 소개

## c. 설계 진행과정

### (3) 자료구조 및 DB설계



# 프로젝트 소개

## d. 개발 진행과정

### (1) 네트워크 개발

- 블록체인 네트워크
- 설정 구성네트워크 구성원 암호화 키/인증서 생성
- 도커 컨테이너 구성



```
# ID to load the MSP definition as
ID: OrdererMSP

# MSPDir is the filesystem path which contains the MSP configuration
MSPDir: crypto-config/ordererOrganizations/example.com/msp

# Policies defines the set of policies at this level of the configuration
# For organization policies, their canonical path is usually
# /Channel/<Application|Orderer>/<OrgName>/<PolicyName>
Policies:
  Readers:
    Type: Signature
    Rule: "OR('OrdererMSP.member')"
  Writers:
    Type: Signature
    Rule: "OR('OrdererMSP.member')"
  Admins:
    Type: Signature
    Rule: "OR('OrdererMSP.admin')"
```

# 프로젝트 소개

## d. 개발 진행과정

### (1) 네트워크 개발

```
23 peer0.org1.example.com:
24   container_name: peer0.org1.example.com
25   extends:
26     file: peer-base.yaml
27     service: peer-base
28   environment:
29     - CORE_PEER_ID=peer0.org1.example.com
30     - CORE_PEER_ADDRESS=peer0.org1.example.com:7051
31     - CORE_PEER_LISTENADDRESS=0.0.0.0:7051
32     - CORE_PEER_CHAINCODEADDRESS=peer0.org1.example.com:7052
33     - CORE_PEER_CHAINCODELISTENADDRESS=0.0.0.0:7052
34     - CORE_PEER_GOSSIP_BOOTSTRAP=peer1.org1.example.com:8051
35     - CORE_PEER_GOSSIP_EXTERNALENDPOINT=peer0.org1.example.com:7051
36     - CORE_PEER_LOCALMSPID=Org1MSP
```

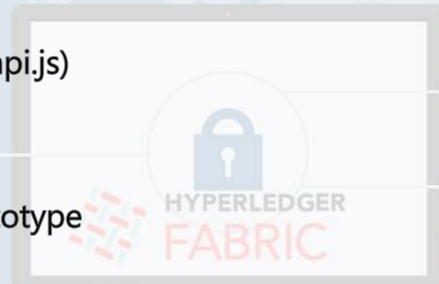
```
-----BEGIN CERTIFICATE-----
MIICVjCCAf2gAwIBAgIQd7i9sACQs7gQboY43mxkUTAKBggqhkJOPQQDAjB2MQsw
CQYDVQQGEWJlbnVzETMBEGA1UECBMKQ2FsaWZvcml5TEWMBQGA1UEBxMNU2FulEZy
YW5jaXNjbzEZMBcGA1UEChMQb3JnMS5leGFtcGxlLmNvbTFFMjB0GA1UEAxMwZGxz
Y2Eub3JnMS5leGFtcGxlLmNvbTAeFw0yMDA4MzExMTIyMDBaFw0zMDA4MjIxMjIy
MDBaMHYxCzAJBgNVBAYTAiVTRMRWwEQYDVQQIEwplYXN0Zm90ZDA4MjIxMjIy
Ew1TYW4gRnJhbmNpc2NvMRkwFwYDVQQKEwBvcml5TEWMBQGA1UEBxMNU2FulEZy
VQDEExZ0bHNjYS5vcml5LmV4YW1wbGUuY29tMFkwEwYHKoZIzj0CAQYIKoZIzj0D
AQcDQgAEITABefQnbUBMEztfiPsc9tiHW0c1nod4goSHL7ry+crwVzl0wDTW0WoO
revjrCJA3BZIEL0yAGDkjZqT5gZR1KNtMGswDgYDVR0PAQH/BAQDAgGmMB0GA1Ud
JQQWMBQGCCsGAQUFBwMCBgggrBgEFBQcDATAPBgNVHRMBAf8EBTADAQH/MCKGA1Ud
DgQIBCBX3lqbp0L170xwqtB7fro8vLPCHRsrdx6l8NzJb/5L4DAKBggqhkJOPQQD
AgNHADBEAiAdrQ8akjVpb92EwxM11etiN6M1Q7OUWhxOw75stNgqVglgGR2b3rY7
B1SmPLaU9Vg5jaPcYjFyz8JhAqFn2a0fc0Q=
-----END CERTIFICATE-----
```

# 프로젝트 소개

## d. 개발 진행과정

### (2) 체인코드 개발

- 학생회 투표 서비스 체인코드 구현(HLF SDK - fabric-contract-api.js)
- 정적 데이터(하드코딩)에 대한 함수 구현 - 1차 prototype
- 동적 데이터에 대한 반응을 가진 비동기 함수 구현 - 2,3차 prototype



# 프로젝트 소개

## d. 개발 진행과정

### (2) 체인코드 개발



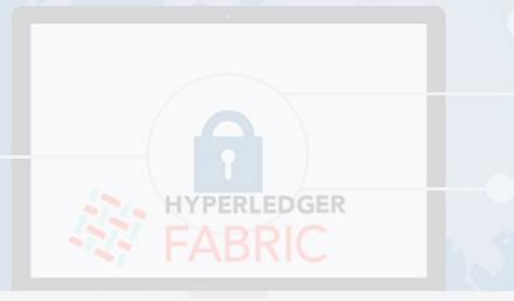
```
51  async createElection(ctx, args) {
52
53      args = JSON.parse(args);
54
55      if(await this.validate(args.name) && await this.validate(args.univ) && await this.validate
56      && await this.validate(args.enddate) ){
57          //Nov 3 is election day
58          let electionStartDate = await new Date(args.startdate);
59          let electionEndDate = await new Date(args.enddate);
60
61          //create the election
62          let electionId = await this.generateElection(ctx, args.name, args.univ, electionStartDa
63
64          // 후보자 등록하는 부분
65
66          await this.generateCandidateResult(ctx , "기권" , electionId);
67
68          for(var i in args.candidates){
69              await this.generateCandidateResult(ctx ,args.candidates[i] , electionId);
70          }
71
72
73          //선거 아이디 리턴
74          let response = {};
75          response.success = electionId;
76          return response;
77
78      } else {
```

# 프로젝트 소개

## d. 개발 진행과정

### (3) 웹 클라이언트(front-end)/ 서버(back-end) 개발

- 클라이언트 : \*학생회 투표 서비스 모바일 UI 디자인 및 구현
- 서버 : \*학생회 투표 서비스 백엔드 기능 구현  
\*체인코드 연동 구현(HLF SDK - fabric-network.js)





# 프로젝트 소개

## d. 개발 진행과정

(3) 웹 클라이언트(front-end)/ 서버(back-end) 개발



# 프로젝트 소개

## d. 개발 진행과정

### (3) 웹 클라이언트(front-end)/ 서버(back-end) 개발

```
628 //투표 기간이 되었는지 확인하기
629 let curDate = new Date();
630 let t1 = new Date(election.startDate);
631 let t2 = new Date(election.endDate);
632 t2.setHours(t2.getHours() - 9);
633 console.log(curDate);
634 console.log(t1);
635 console.log(t2);
636 if(curDate >= t1 && curDate <= t2){
637     console.log(election.univ + ' 투표기간입니다.');
```

```
638 }else{
639     console.log(election.univ + ' 투표 기간이 아닙니다.');
```

```
640     res.send('<head><meta charset=\'utf-8\'></head><script>alert(\''+election.univ+' 투표 기간이
641     return;
642 }
643
644 let context = {
645     session:req.session,
646     univ:election.univ
647 };
648 htmlrender(req, res, 'sign', context);
```

Hyperledger Fabric SDK for node.js

Modules ▾

Classes ▾

Interfaces ▾

Tutorials ▾

Global ▾

Search



## Module: fabric-network

### Overview

This module provides a higher level API for interacting with smart contracts, and is the recommended API for client applications to interact with smart contracts deployed to a Hyperledger Fabric blockchain network.

Note that administrative capabilities, such as installing and starting smart contracts, are not currently provided by this API. For these tasks or other specific advanced usage, the lower level *fabric-client* API should be used. Access to related *fabric-client* objects is provided through the *fabric-network* API objects.

TypeScript definitions are included in this module.



**HYPERLEDGER**  
**FABRIC**

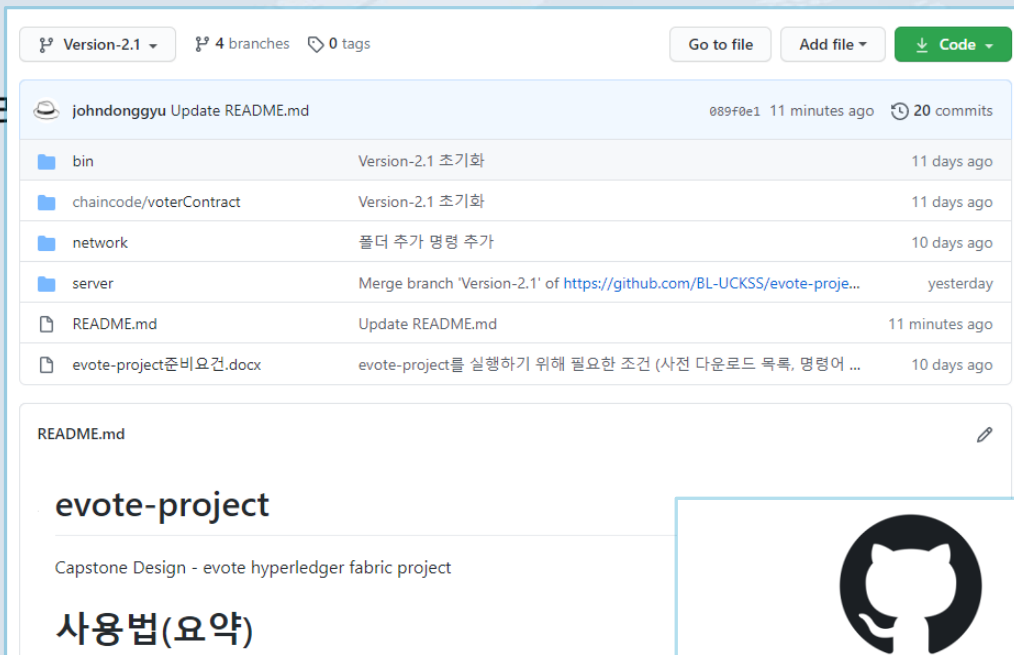


# 프로젝트 소개

## d. 개발 진행과정

### (4) 개발 과정 산출물 버전관리

- github 이용
- Prototype 1 : Ver1.0
- Prototype 2 : Ver2.0
- Prototype 3 : Ver2.1



The screenshot shows a GitHub repository interface for 'evote-project'. At the top, it indicates 'Version-2.1' with 4 branches and 0 tags. Below this is a commit history table:

Commit Message	Commit Hash	Time	Commits
johndonggyu Update README.md	089f0e1	11 minutes ago	20 commits
bin	Version-2.1 초기화	11 days ago	
chaincode/voterContract	Version-2.1 초기화	11 days ago	
network	플더 추가 명령 추가	10 days ago	
server	Merge branch 'Version-2.1' of <a href="https://github.com/BL-UCKSS/evote-proje...">https://github.com/BL-UCKSS/evote-proje...</a>	yesterday	
README.md	Update README.md	11 minutes ago	
evote-project준비요건.docx	evote-project를 실행하기 위해 필요한 조건 (사전 다운로드 목록, 명령어 ...)	10 days ago	

Below the commit history, the README.md file is displayed. The title is 'evote-project' and the subtitle is 'Capstone Design - evote hyperledger fabric project'. The main heading is '사용법(요약)'.



# 프로젝트 전체 구조도



# Hyperledger Fabric

CA  
참가자 식별

Endorser  
Endorser  
ledger  
chaincode A B  
Org1  
My Channel

RAFT Ordering-service

Web Server

Register User/Voter

내부DB or FileSystem  
세션  
이미지 파일

login service

Restful API

외부 DB

회원

- 회원정보
- 아이디
- 비밀번호
- 학과
- 학년
- 재학여부
- ...

유권자 등록  
인증서 발급

등록 필요시

유권자일 경우

return

Transaction

권한에 따라 정보제공

어플리케이션 사용자

학생1 학생2 ...

모든 연결에 TLS  
보안 통신 적용  
(grpc, https)



# 최종 결과물 시연 (영상)





# 최종 결과물 추가 설명

## a. 주요기능

### - 로그인

사전 db에 있는 정보와 사용자 비교하여 인증

### - 투표하기

설정된 투표 기간 내에서 자신의 신분으로 가능한 모든 투표 행사

### - 내 투표 확인

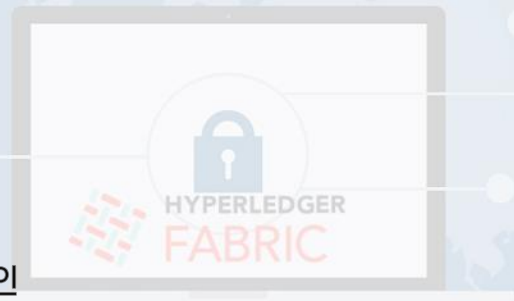
투표를 완료한 후 내가 행사한 표에 대해 무결성 및 부인 방지 확인

### - 선거 결과 확인

투표율 및 당선결과 확인

### - 선거 등록/관리

선거 정보, 후보자 정보 등을 등록 및 선거 진행 관리





# 최종 결과물 추가 설명

## b. 기존 전자투표와 차별점

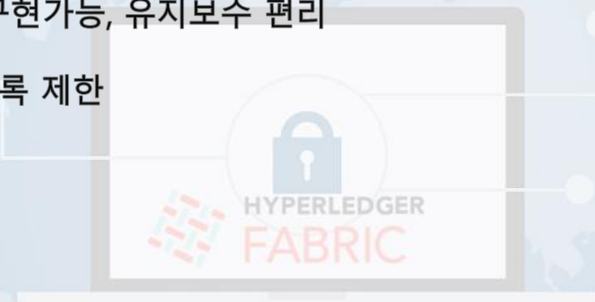
- 분산 시스템 사용하여, 중앙관리서버에 문제가 생겨도 시스템 운용가능
- 해시함수와 머클트리를 이용하여 선거관련 정보나 후보자 관련 정보에 대한 무결성 보장
- 사용자 정보 등 공개되지 않아야 할 데이터를 시스템 상에서 감출 수 있음
- 한 번 데이터가 원장에 들어가게 되면 수정 및 삭제가 불가능하므로 이중 투표를 자동으로 방지



# 최종 결과물 추가 설명

## c. 퍼블릭 블록체인 기반 전자투표와 차별점

- 범용적인 프로그래밍 언어를 사용하여 투표시스템을 완전히 구현가능, 유지보수 편리
- 인증서를 발급받은 사용자(재학중인 학생)만이 이용할 수 있도록 제한
- 실시간 투표현황을 반영할 수 있을 정도로 속도향상



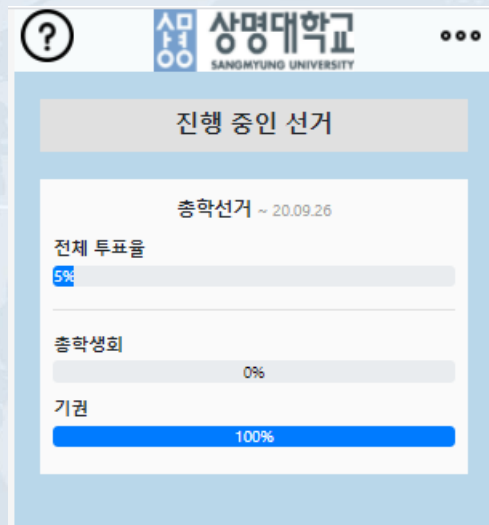
# 최종 결과물 추가 설명

## d. 주요 보안과제 달성 사항

### (1) 무결성

```
Save Changes Cancel
```

```
1 {
2   "_id": "ZT4Hu6B1SxaI2B42kWUzqOyHaxShYkCbPlA42rIzc6c=",
3   "_rev": "7-2e7166883a3cd874ce7602c4f41941be",
4   "election": {
5     "electionId": "41m1nc6gunskurxgmbv31v",
6     "endDate": "2020-09-26T00:00:00.000Z",
7     "name": "총학선거",
8     "startDate": "2020-09-20T00:00:00.000Z",
9     "type": "election",
10    "univ": "총학생회"
11  },
12  "picked": "총학생회",
13  "type": "vballot",
14  "voterId": "ZT4Hu6B1SxaI2B42kWUzqOyHaxShYkCbPlA42rIzc6c=",
15  "walletId": "8b0246226ddfa7eca487c0854a0c6a61fe21bd4c95b419928c6a74cbf7f5e76c",
16  "~version": "\u0000CgMBBQA="
17 }
```



```
[ { electionName: '총학선거',
  startDate: '2020.9.20',
  endDate: '2020.9.26',
  candidateName: [ 'voteNum', '총학생회', '기권' ],
  count: [ 10, 50, 50 ] } ]
/myvote 호출됨
```

# 최종 결과물 추가 설명

## d. 주요 보안과제 달성 사항

### (2) 인증 및 기밀성

```
/process/login 라우팅 함수 호출됨 .
요청 파라미터 : 201710001, jZae727K08Ka0mKSg0aGzww/XVqGr/PKEgIMkjrcbJI=
authUser 호출됨 : 201710001, jZae727K08Ka0mKSg0aGzww/XVqGr/PKEgIMkjrcbJI=
아이디 201710001로 검색됨 .
비밀번호 일치함 .
registerUser 호출됨
voterId
7e08e72e6deab47127565c429374674f1767059414f4b54fb871ac171979fd66
Successfully registered voter 7e08e72e6deab47127565c429374674f1767059414f4b54fb871ac171979fd66. Use vot
erId 7e08e72e6deab47127565c429374674f1767059414f4b54fb871ac171979fd66 to login above.
성공적으로 wallet에 등록되었습니다 .
```

```
-----BEGIN CERTIFICATE-----
MIICVjCCAf2gAwIBAgIQd7i9sACQs7gQboY43mxkUTAKBggqhkJOPQDajB2MQsw
CQYDQQGEGeWJUzETMBEGA1UECBMkQ2FsaWZvcml5PTEwMTU2FulEZY
YW5jaXNjbzEZMBCGA1UEChMQb3JnMS5leGFtcGxlLmNvbTFFMjB0GA1UEAxMwZGxz
Y2Eub3JnMS5leGFtcGxlLmNvbTFFMjB0GA1UEAxMwZGxzY2Eub3JnMS5leGFtcGxl
MDBaMHYxCzAJBgNVBAYTAiVTRMRmEQYDQVQlEwPDIWxpZm9ybmlhMRYwFAYDQVQH
Ew1TYW4gRnJhbmNpc2NvMRkwFwYDQVQKExBvcml5LmV4YW1wbGUuY29tMR8wHQYD
VQQDEExZ0bHNjYS5vcml5LmV4YW1wbGUuY29tMFkwEwYHKoZIzj0CAQYIKoZIzj0D
AQcDQgAEITAfBefQnbUBMEztfiPsc9tiHW0c1nod4goSHL7ry+crwVzl0wDTW0WoO
revjrCJA3BZIEL0yAGDkjZqT5gZR1KNtMGswDgYDVR0PAQH/BAQDAgGmMB0GA1Ud
JQQWMBQGCsGAQUFBwMCBggrBgEFBQcDATAPBgNVHRMBAf8EBTADAQH/MCKGA1Ud
DgQiBCB31qbp0LI70xwqtB7fro8vLPCHrSrdx6I8NzJb/5L4DAKBggqhkJOPQD
AgNHADBEAiAdRQ8akjVpb92EWxM11etiN6M1Q7OUWhxOw75stNGqVglgGR2b3rY7
B1SmPLaU9Vg5jaPcYjFyz8JhAqFn2a0fc0Q=
-----END CERTIFICATE-----
```

```
-----BEGIN PUBLIC KEY-----
MFkwEwYHKoZIzj0CAQYIKoZIzj0DAQcDQgAEISmvs7w6tg5jrs24ML112WGHq7RW
Gli3m16rB1QmRIIa6bpwYY3qctb8e1hxJCKVT4WYVWkqY0y+XuVwvCFkA==
-----END PUBLIC KEY-----
```

# 최종 결과물 추가 설명

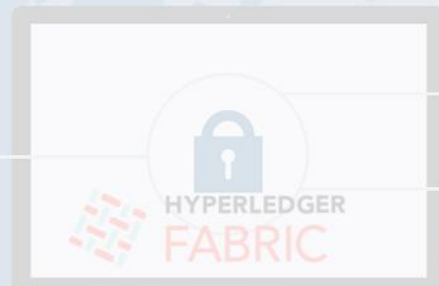
## d. 주요 보안과제 달성 사항

### (3) 부인방지 ( + 이중투표 방지 )

localhost:8081 내용:

이미 모든 투표를 완료했습니다.

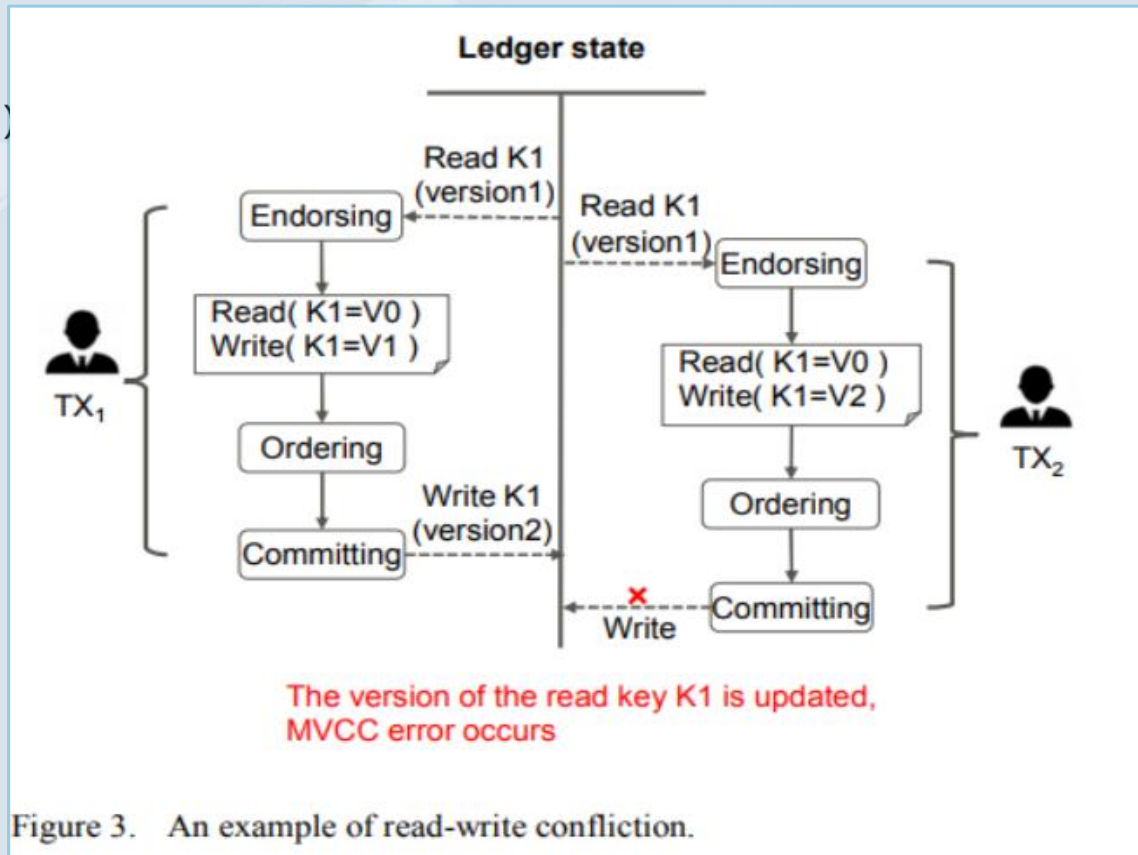
확인



# 최종 결과물 추가 설명

## d. 주요 보안과제 달성 사항

(3) 부인방지 (+ 이중투표 방지)



# 최종 결과물 추가 설명

## d. 주요 보안과제 달성 사항

### (3) 부인방지 ( + 이중투표 방지 )

```
2020-09-20T22:48:56.964Z - warn: [TransactionEventHandler]: _strategyFail: strategy fail for transaction "b70c2b665c44ac760f46928e6d273a17e4419163cd9b87b0d4993a3d59076a10": TransactionError: Peer localhost:7051 has rejected transaction "b70c2b665c44ac760f46928e6d273a17e4419163cd9b87b0d4993a3d59076a10" with code "MVCC_READ_CONFLICT"  
Failed to submit transaction: TransactionError: Peer localhost:7051 has rejected transaction "b70c2b665c44ac760f46928e6d273a17e4419163cd9b87b0d4993a3d59076a10" with code "MVCC_READ_CONFLICT"
```

```
2020-09-20T22:50:55.578Z - warn: [DiscoveryEndorsementHandler]: _build_endorse_group_member >> G0:0 - endorsement failed - Error: transaction returned with failure: ReferenceError: election is not defined  
2020-09-20T22:50:55.579Z - error: [DiscoveryEndorsementHandler]: _endorse - endorsement failed::Error: Endorsement has failed
```

# 향후 발전 방향

## - 탐색 알고리즘 개선

현재 학생의 [투표하기] 기능의 속도가 느림

## - 추가적인 기능

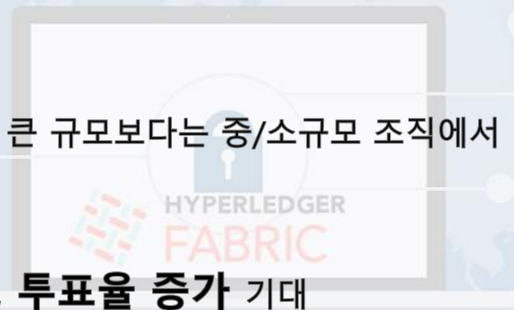
- 1) 현재 건의할 사항은 관리자의 이메일을 통해 연락하는 것으로 되어있으나, **게시판** 형태로 업그레이드 하는 것도 좋은 방법
- 2) 보안성 강화를 위해 생체인증 등의 **2차인증** 추가
- 3) 내 투표 확인 기능은 현재 자신의 투표 값만을 게재하고 있으나, **블록 인덱스** 기능을 추가하여 투표값이 전체 블록체인 어디에 들어가 있는지 가시적으로 제공
- 4) **샘물 어플에 연동**하여 선거 시작일에 핸드폰 알림 기능을 이용하여 쉽게 투표시스템에 접근





# 느낀점

- 기존 **서버 시스템**을 사용했을때 당연한 신뢰를 바탕으로 시스템을 이용했으나, **블록체인 시스템**을 통해 투표 조작이 불가능함을 확인하고, 모든 트랜잭션을 직접 검증해보니 **보안**에 있어서 좀 더 경각심을 갖게 됨
- 가까운 미래에 블록체인 투표 시스템이 활용될 곳은 국가적인 큰 규모보다는 **중/소규모 조직**에서 활용될 가능성이 높을 것으로 보임
- **블록체인** 활용 전자투표 적용을 통해 **투표 절차 간소화, 투표율 증가** 기대
- 블록체인의 가장 특징인 **투명성, 불변성, 탈중앙성, 확장성**을 무조건적으로 낙관적 전망은 아직 이를 것으로 보임 (우리의 시스템도 아직은 소규모에서 테스트한 것이기 때문)



# Q&A



Thank you

