가로선

**캡스톤 디자인 회의록**

2020년 4월 25일 토요일

**─**

# **참석자**

유성근, 조혜령, 김동규, 송찬훈, 신현승

**회의 안건**

## 지난 회의 내용

1. DB 및 네트워크 구조 생각해보기

* 다중채널로 구현 or 단일채널+프라이빗데이터로 구현, ca, 오더러 등등

1. 각자 파트별로 javascript,node.js,ECMAscript 공부하기

## 새 내용

1. ca 정확히 어디에 어떻게 배치해야하는지 알아보기

* 펩카랑 커머셜 페이퍼 예제에서 어떻게 사용되는지 알아보기
* IBM evote 영상 참고

1. 기능설계준비

* 다른 투표시스템 기능조사(evote 등등) 체인코드 어떻게 나누어져있는지. 체인코드 기능은 어떤지 조사
* 요구사항명세서 유즈케이스 보면서 기능 상세기능 생각해보기

1. 각자 파트별로 javascript,node.js,ECMAscript 공부하기
2. 다음 회의일시 : 5월 2일 (토) 오후 2시

* 상세 기능 설계

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **구분** | **금주계획** | **완료예정** | **금주실적** | **진척율(%)** | **차주계획** | **완료예정** |
| 네트워크 구조 설계 | 1. 네트워크 설계 완성 | 4.21 | 1.채널 토의 | 90% | 1.ca 공부(펩카, 커머셜페이퍼, evote 영상 참조) | 5.2 |
| DB 설계 | 1. DB설계 완성 | 4.25 | 1.DB설계 완성 | 100% |  |  |
| 기능(체인코드) 설계 |  |  |  |  | 1. 기능설계준비  (다른 투표시스템 기능조사, 요구사항명세서보고 상세기능 생각) | 5.2 |
| 기타 | 1. 프로그래밍언어 공부  ( javascript,node.js,  ECMAscript 등 공부하기) | 미정 |  |  | 1. 프로그래밍언어 공부  ( javascript,node.js,  ECMAscript 등 공부하기) |  |
| 특이사항 |  | | | | | |

# **메모**

* **DB설계**
  + SSO DB(외부)
    - 학번
    - 학년
    - 비밀번호
    - 이름
    - 부서/학과
    - 재학상태
    - email
* 2차인증은 학교이메일로 하면안됨. 1차 로그인도 학번, 비밀번호로 들어가므로, 학교 이메일로 인증하는 것은 같은걸 두번 반복하는 꼴.
  + PD(블록체인)
    - 전체학생수(전체투표율이 노출되면 안됨, 그래서 PD)
    - 찬성수
    - 찬성률
    - 기권표수
    - 기권표율
    - 반대표수
    - 반대표율
    - 후보자의총투표수
    - 소속단과대

* + Ledger(블록체인)
    - 투표자ID (hash값:선거고유값+학번+secret\_key)
* 투표자ID생성 시에 투표 별로 고유값을 더해서 hash화해서 id값을 만든다면, 연도별로 고유값을 바꾸도록 해서, 1년 후에 또 해당 id를 사용할 필요가 없음. 즉 PD에 안넣어도됨
  + - 투표 여부
    - 투표한 후보자
    - 전체투표수
    - 단과대투표수
* 현행 학교 시스템은 전체 및 단과대 투표현황 공개 안하지만, 국회의원선거도 투표율 공개하는데 그냥 다 공개하는걸로 하자(투표독려기능도 있을듯)
  + - 투표시작기간
    - 투표종료기간
  + Server DB(내부)
* sso db를 실제로는 내부db로 구현한다고 하더라도, 외부db를 사용하는 것을 가정하고 만든것이므로, 서버db를 따로 만들 필요가 있음
  + - 후보자기호(몇번)
    - 후보자직책
    - 후보자성명
    - 정/부후보 구분
    - 후보자학과(전공)명
    - 후보자학년
    - IP주소(미정)
* 현재 전자투표시스템에서는 ip주소를 수집하는데, 아마 중복투표를 막기 위함인듯(2012년도에 투표인원 절반이 같은 ip주소를 사용했던 사고 사례가 있음)
* 우리는 1,2차 본인인증하고 원장에 투표여부 기록하는데 굳이 수집해야할까?
  + - 프로필 사진
    - 구호(캐치프레이즈) - ?
    - 선거공약관련 링크주소
    - 선거이름
    - 당선여부
    - 투표시작기간
    - 투표종료기간
* **네트워크 설계**
  + 단과대 안에 학과별로 피어를 만들지 않고, 선관위와 정보통신팀 피어를 모든 채널에 조인시키는 안
    - 학과별로 피어를 놓게되면 피어수가 너무 많아지고, 총학선거시에도 학과별로 피어를 해야해서 복잡하지 않을까?
    - 선관위/ 정보통신팀이 모든 채널에 조인한다면 선거당 최소 3개의 엔도싱피어가 생김
    - 선관위가 피어를 더 여러개 들고 있어도 될듯
    - 단과대는 학생측, 선관위는 중립, 정보통신팀은 학교측의 입장에서 검증가능
    - 선관위랑 정보통신팀이 모든 선거에 대한 인돌싱 및 커미팅해야되는데 정보통신팀은 그렇다 치더라도 선관위 컴퓨터가 버틸수있을까?
* 인돌싱 및 커미팅 등은 모두 네트워크 내부에서 이루어지는것. 선관위나 정보통신팀에 부하가 가는것은 아니다.
  + - 채널은 선거별로(총선, 단과대별 선거)- 채널 구조가 변경될 일은 없을 것이다(통폐합되지않는한)
  + 오더러
* 일단 구현은 싱글오더러로
  + ca
    - 조직별 ca, 오더러 ca 가 필요할듯
* 투표자 ca는 크립토젠을 활용하므로 불필요
  + - 사용자는 블록체인 네트워크에서 검증하지 않고, 1,2차 인증으로 충족한후, 미리 크립토젠을 통해 발급된 키 제공
    - 그 외 ca 정확히 어디에 어떻게 배치해야하는지 모름
* 펩카랑 커머셜 페이퍼 예제에서 어떻게 사용되는지 다시 알아보기
* IBM evote 영상 참고하기
* **기타**
  + 개발환경
* 아래와 같은 똑같은 걸로 할듯?



* + 다음회의 시간에 할것
    - 상세 기능 설계
    - 다음 시간 전까지
* 다른 투표시스템 기능조사(evote 등등)체인 기능은 어떤지 코드 어떻게 나누어져있는지. 체인코드조사
* 요구사항명세서 유즈케이스 보면서 상세기능 생각해보기