

졸업프로젝트 멘토링보고서

블록체인 떡상팀

201302397 문준영

201203395 이우연

201402315 곽현준

피드백 내용 :

피드백은 Notion을 통해 지도교수의 조교에게 받게 되었습니다.

저희가 작성한 모든 문서에 대해서 누락되거나 변경되어야 할 점,

또는 추가되어야 할 점 등에 대해서 피드백을 받았고, 이를 최대한 반영하여

문서 및 다이어그램들을 개선하였습니다.

1. 문제정의서

문제정의 부분:

문제 정의서

- 연구의 필요성 부분에서 신분증은 컴퓨터를 활용하는 공인인증서로 ⇒ 공인인증서와는 다름
 - 공인인증서는 굳이 대응시키면 도장 또는 사인과 관련된 프로그램
- 또한 공인인증서의 불편함을 언급하였는데, UseCase 혹은 다른 시퀀스 다이어그램에서의 공인인증서를 활용해서 본인인증에 대한 내용은 적절치 않음(공인인증서를 활용한 인증과정 제거)
- 연구 목표 및 내용에서 사생활 침해등의 문제 해결의 이유가 적절하지 않음
- 유스케이스 다이어그램에서 Host와 Verifier가 신원 인증 요청이라는 항목을 같이 접근하는데 Verifier와 Host의 작업이 다르므로 분리

이를 통해 공인인증서로 사용자의 신원을 검증하는 부분을 아예 없애버렸으며 Issuer에서 신원을 검증하고 신분증을 발급하는 부분의 피드백을 전달받아 이를 반영하였습니다.

2. 요구사항명세서

요구사항 명세서

- 하드웨어 인터페이스에 적힌 QR 코드식별할 하드웨어 요구 X
⇒ QR코드 식별은 하드웨어 단에 필요한게 없음

작성하였던 요구사항 명세서에서 하드웨어 인터페이스에 QR 코드 식별을 위한 하드웨어가 필요하다고 작성하였으나, QR 코드 식별은 하드웨어 단에서 필요하지 않다는 피드백을 받아 해당 내용을 삭제하였습니다.

2.2. 하드웨어 인터페이스 (Hardware Interface)⁴

안드로이드 시스템을 사용하는 스마트폰에서 해당앱을 설치 및 실행할 수 있다. DID 와 연동하기 위해 WIFI 혹은 모바일 데이터를 통해 통신할 수 있어야한다. ~~QR스캔 기능을 활용하기 위해서 하드웨어의 카메라가 제공되어야 하며, 생체기능 인식을 활용하고자 한다면 이를 지원하는 센서가 제공되는 스마트폰을 사용해야 한다. 또한, 인증요청자의 경우 사용자가 생성한 QR코드를 식별할 수 있는 하드웨어가 요구된다.~~⁴

3. 유스케이스명세서

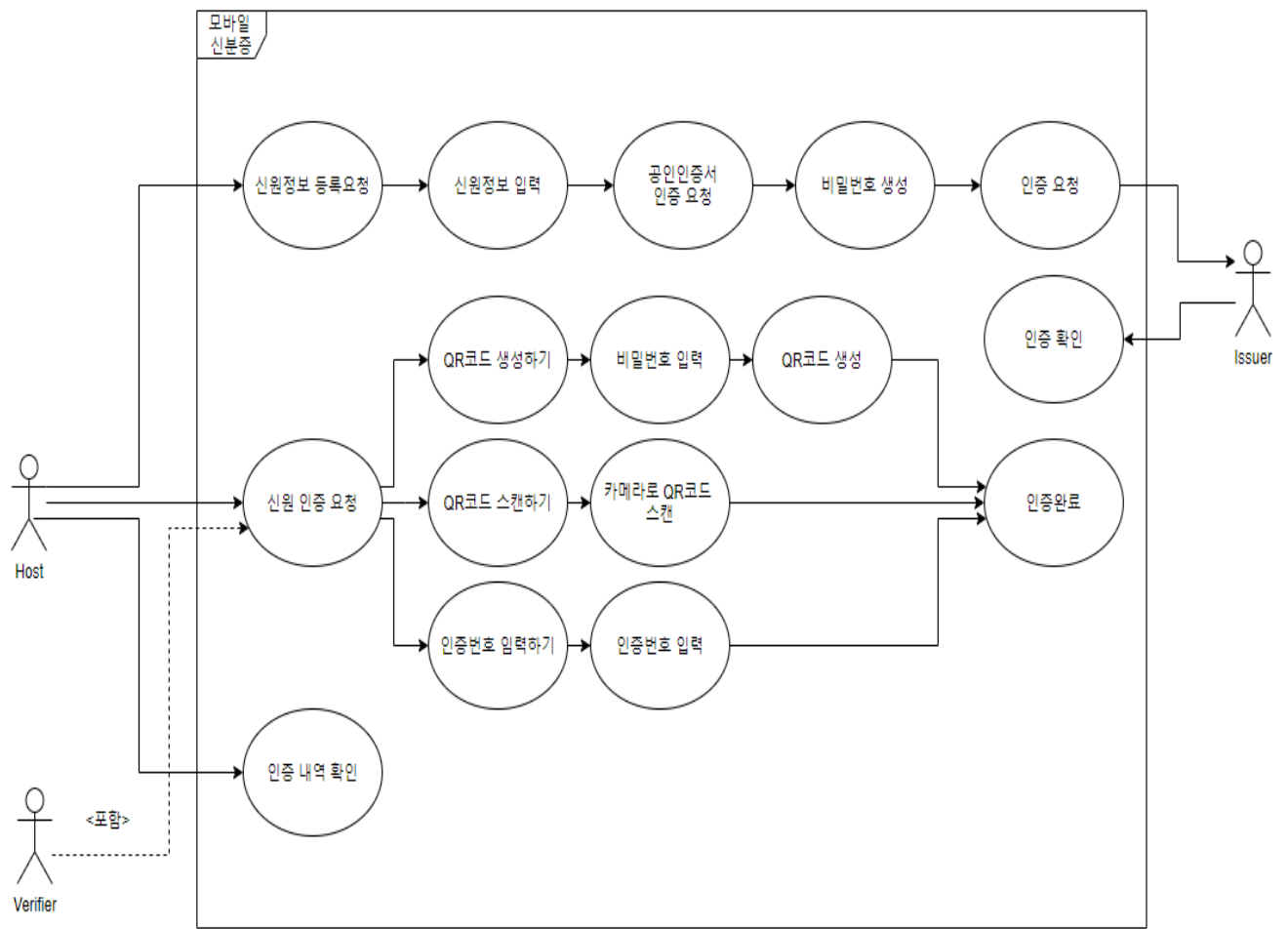
유스케이스 부분:

유스케이스 명세서

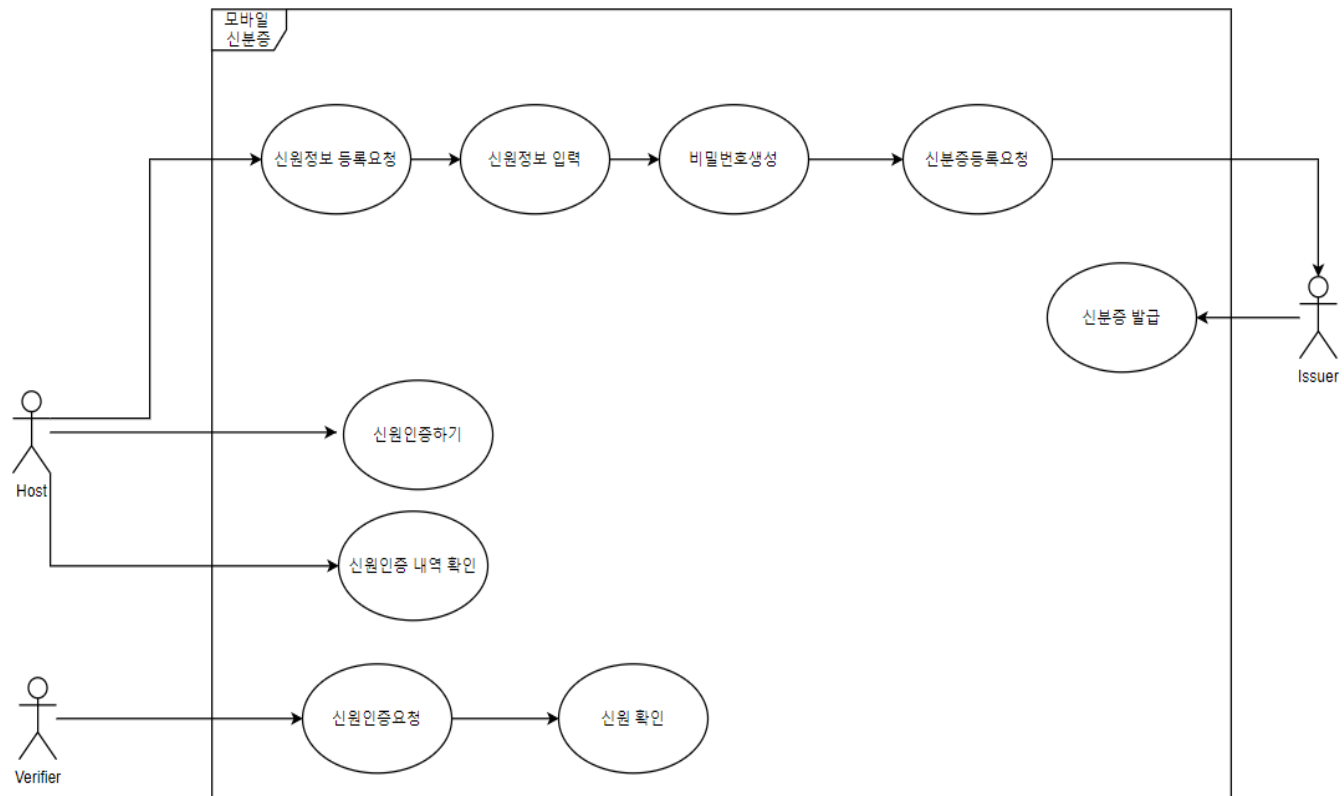
- (Verifier에서 QR코드 생성작업을 따로 분리)
- 지금은 너무 세부적으로 작성되어있는데 **어떠한 작업을 하는지**가 중요한 것이지 어떻게 세세하게 이루어지는지가 중요한 것이 아니므로 행위 단위로 항목들을 합치는 것이 좋을 것 같음

이것을 통해 유스케이스에서는 구체적인 행위 보다는 꼭 필요한 행위 만을 넣게 되어 이전보다 간소화를 시켰습니다. 그리고 이전과는 다르게 Verifier와 Host의 모바일 어플리케이션을 각자 분리시켰습니다. (프로토타입을 보시면 이해가 가실 겁니다)

유스케이스 명세서 (피드백 받기전) :



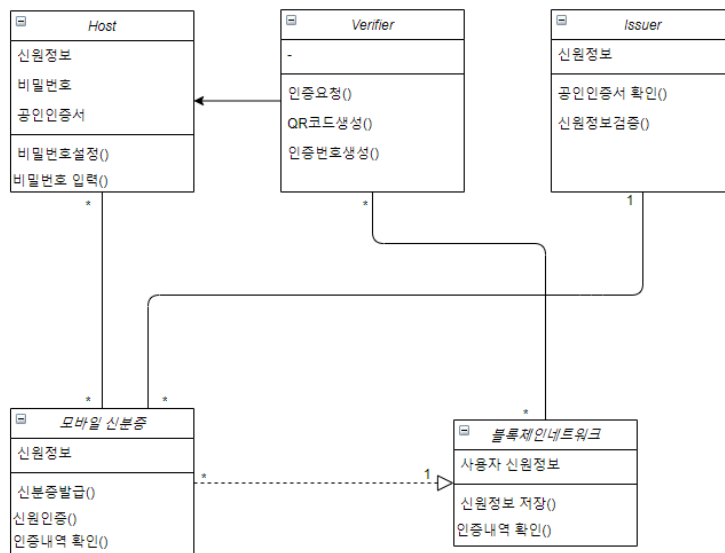
유스케이스 명세서 (피드백 받은뒤) :



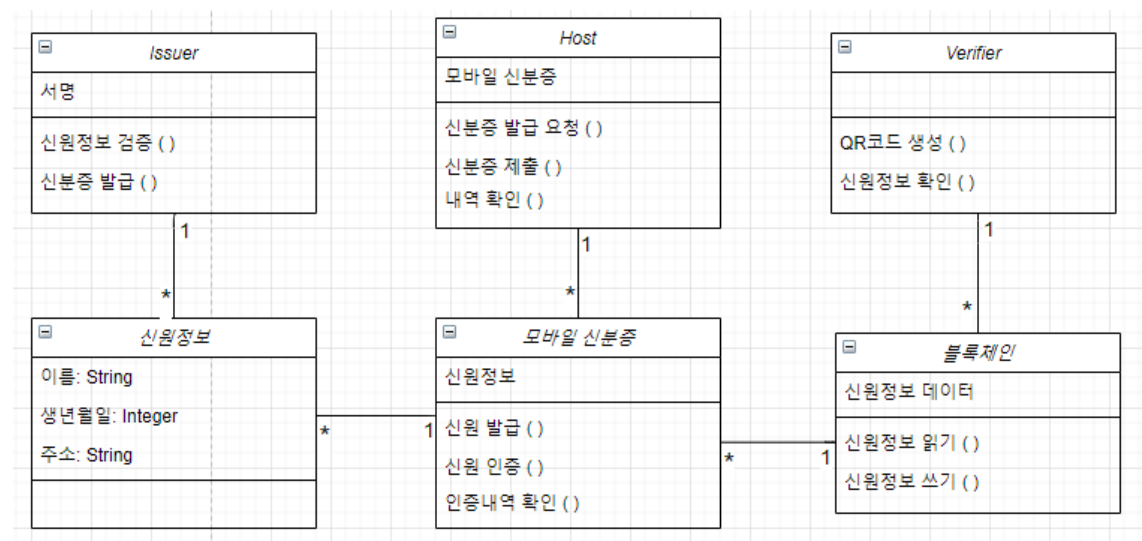
4. 클래스 다이어그램

Host 와 Verifier, Issuer, 모바일 신분증, 블록체인 5 개의 객체에서 신원정보를 추가하여 6 개의 객체로 변경되었습니다. 신원정보는 모바일 신분증을 통해 사용자가 입력한 정보이며, 신분증 발급시 Issuer 에게 전달되어야 할 객체입니다. Issuer 는 이 객체를 통해 신원정보를 검증합니다.

(피드백 전)



(피드백 후)



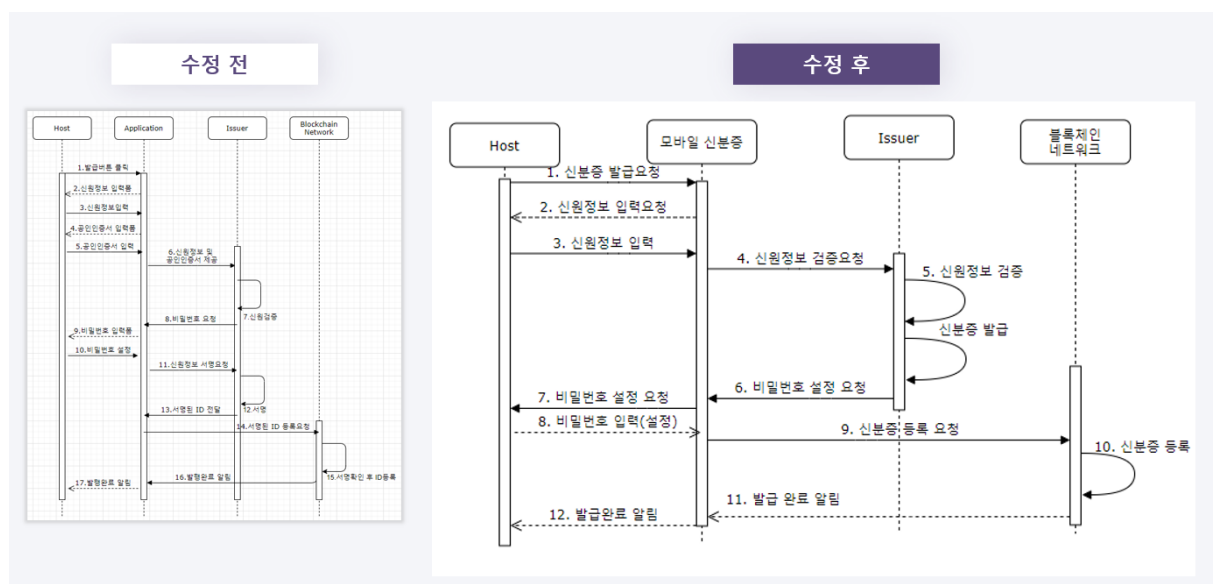
5. 시퀀스 다이어그램

시퀀스 다이어그램

- 3.1 신원정보 등록은 변경된 클래스 다이어그램에 맞게 작성
 - 비밀번호를 지정하는 것은 관용키 암호화와 관련된 사항으로 Issuer와 상관없이 application 단에서 따로 처리
 - 비밀번호는 어플리케이션에 저장하지 않음
- 3.2 시퀀스 다이어그램은 Verifier에서 QR코드를 제시했을 때 Host가 QR코드를 스캔하는 것으로 어플리케이션에서 저장된 신분증을 QR코드에서 얻은 정보를 통해 전송하는 것
 - Host에서 QR코드를 생성할 필요 없음
 - 인증번호는 아예 필요가 없는것 같은데

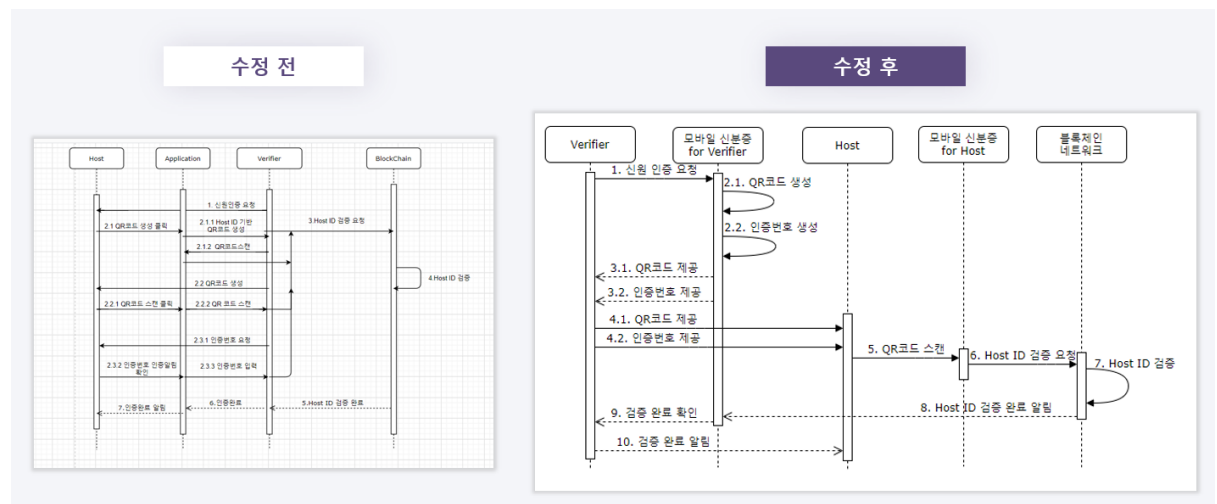
기존에 작성하였던 시퀀스 다이어그램은 3 가지로 각각 '신원정보 등록하기', '신원인증하기', '신원인증내역 확인하기'입니다. 이중 신원정보 등록하기와 신원정보 인증하기를 수정하였습니다.

신원정보 등록은 클래스다이어그램의 수정과 함께 신분증 발급과 공인인증서에 관련한 내용을 수정하고, 비밀번호 역시 모바일 신분증에서 관리할 수 있도록 하였습니다.



신원인증하기에서는 이전 시퀀스 다이어그램의 경우 Host 역시 QR 코드를 생성할 수 있도록 하였지만, 해당 기능이 불필요하다는 피드백을 받아 Host 는 QR 코드를 따로 생성하지 않고 QR 코드 스캔만 할 수 있도록 하였습니다.

반면 인증번호의 경우 기존에도 사용하고 있는 방법이며, 불가피하게 카메라를 사용할 수 없는 상황을 고려하여 인증번호를 통한 인증하기는 사용할 수 있도록 하였습니다.



6. 프로토타입 개선

개선전 프로토타입 :

<https://ovenapp.io/view/drulwkv1ZfsGNUwPyWFA1iqjLdaDkBmi/>

개선후 프로토타입 (Host,Verifier로 나눔) :

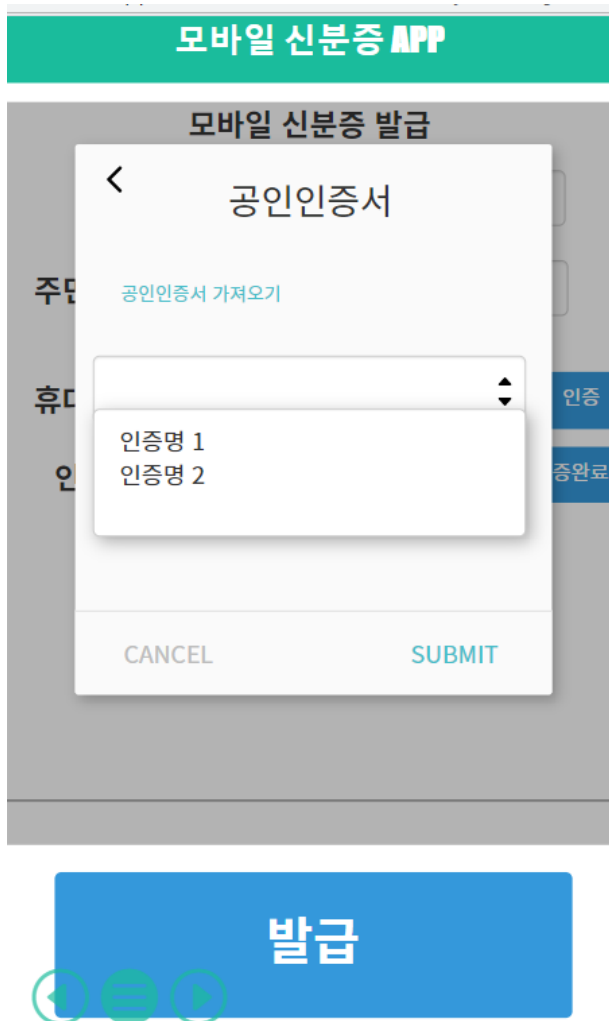
- Host (사용자 버전) :

<https://ovenapp.io/view/AFG12SETEOHECOJVD09iZtUdNIgbNvuw/>

- Verifier(인증자 버전):

<https://ovenapp.io/view/impqjDKl20dDfXbeRd21pNRK8gyKo1HR/>

개선전



개선전에는 발급시 공인인증서를 통해 서명을 해서 신원인증을 발급받았습니다.

개선후:

모바일 신분증 APP

모바일 신분증 발급

이름 :

주
휴

신분증 발급이 완료되었습니다.

OK

인증
완료

주소 :

발급

개선후에는 공인인증서 서명을 아예없애고 Issuer 측에서 사용자 신분을 인증하고 신분증 발급해주는 식으로 구성되었습니다.

인증방식 또한 간소화시켰습니다.

개선전에는 총 3가지의 인증방식이있었으나 QR코드생성은 Verifier관점에서 만 필요하다생각이 들어 없애주었고 UI도 간소화된 모습을 볼수있습니다.

개선전 :



개선후



마지막으로 Verifier 버전의 어플리케이션을 따로 만들어주었습니다.

모바일 신분증 APP



인증기관 : XXX

신원 인증 요청

신원 인증내역

모바일 신분증 APP



인증기관 : XXX

QR코드생성

인증번호생성

신원인증을 요청을 누를경우 Verifier측에서는 사용자(host)가 신원을 인증 받을 QR코드와 인증번호를 생성해줄수있습니다.

신원인증 내역은 사용자(host)가 해당 기관에 받은 인증내역들을 보여줍니다.

현재는 Verifier가 어플리케이션 형태로 되어있지만 만약 Verifier가 휴대폰 이 없을 상황이 생길수있으므로 Verifier만 접근할 수 있는 웹페이지를 만들어서 사용자를 인증할수 있는 방식도 넣어보려고합니다.

GitHub : <https://github.com/BlockChainupTeam/Comprehensivedesign>

Youtube: <https://youtu.be/gjiInnYKDE4>