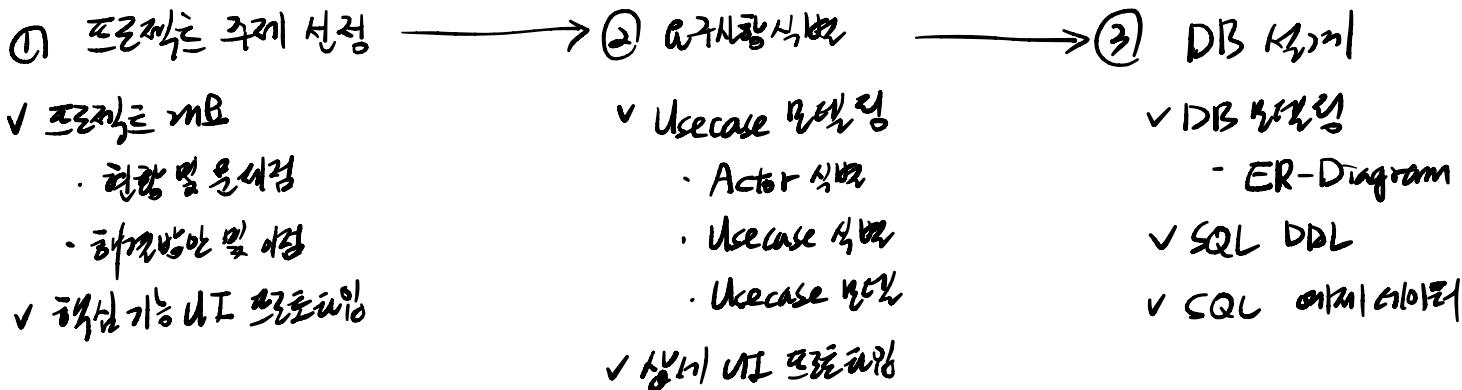


자|종 츠르르

* 설계 및 구현

100%

g+m



9/20

⑥ 배포 및 배포 ←

Naver Cloud

⑤ 통합 테스트 ←

④ 품질

30/11



* 주의 사항

① 기호 범주 간접을 통한

② 2D영 \rightarrow 1D영 기호



영 \rightarrow 1D영

:

* Use-case Modeling

① Actor 1) 흔적

~ 시스템을 사용하는 사용자는 주인공

→ 주인공은 다른 사람

Primary Actor

직원



↑
Actor의
역할

→ 주인공은 다른 사람

Secondary Actor

primary actor (주인공)

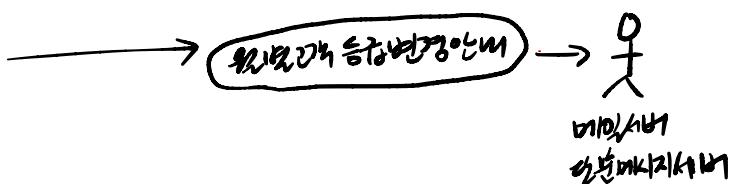
↓
운영시스템

직원



직원

직원



직원
직원
직원
직원

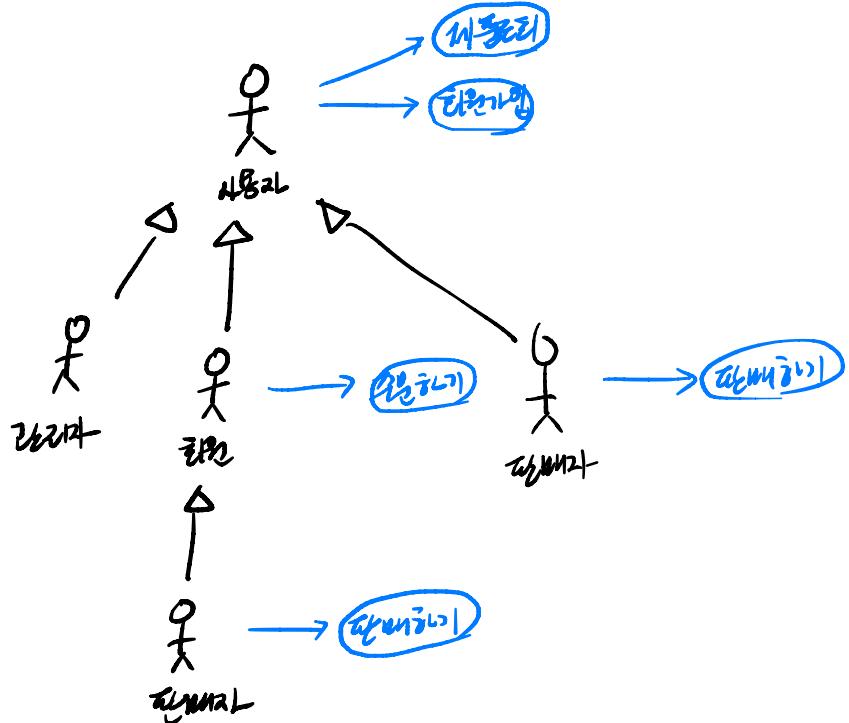
↑
primary actor
(주인공)

* Primary Actor 흔적 → Usecase 흔적 → Secondary Actor 흔적

② Actor 3단계 인식 (Identify)

→ 청중에게 전달하기

a) 성별



③ Use-case 허브

• 이용자가 시스템을 이용하여 일정과제를 완수 목표

- 수집한 기준
- ✓ 업무해야 한다 예) 주문하기, ~~도착하기~~
 - ✓ 사용자 품질 특성
↳ 가용성 가능
 - ✓ 해당 이용자가 한 시점에 한정되며 유통하는 경우



- 미시글 등록
- 미시글 조회
- 미시글 변경
- 미시글 삭제

Actor는 기관으로
Use-case 허브



③ disease 허가증 2

third Guide 2

✓ CRUD의 원칙과 Use case의 원칙을 서로 통합한다

- 개시를 등록
 - 개시를 조회
 - 개시를 수정
 - 개시를 삭제
- ⇒ "XX 관리"로 통합된다
- (1) 개시를 관리

✓ 관련 Usecase를 한데로 통합한다

- 등록
 - 조회
- } → 사용자 인증

✓ 여러 Usecase의 통합된 하나로 통합된다
관련 Usecase로 묶기



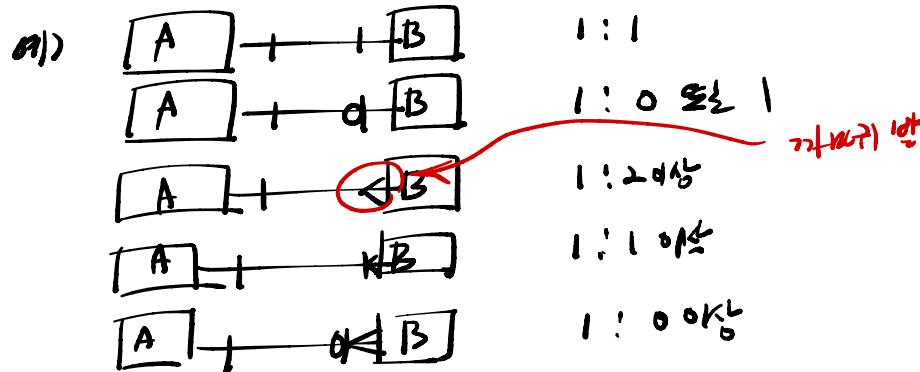
* DB 모델링

↳ 성과화 \Rightarrow DB(데이터) 속성 정의 및 관계 \Rightarrow SQL 구문
↓
설계

Entity-Relationship Diagram (ERD)

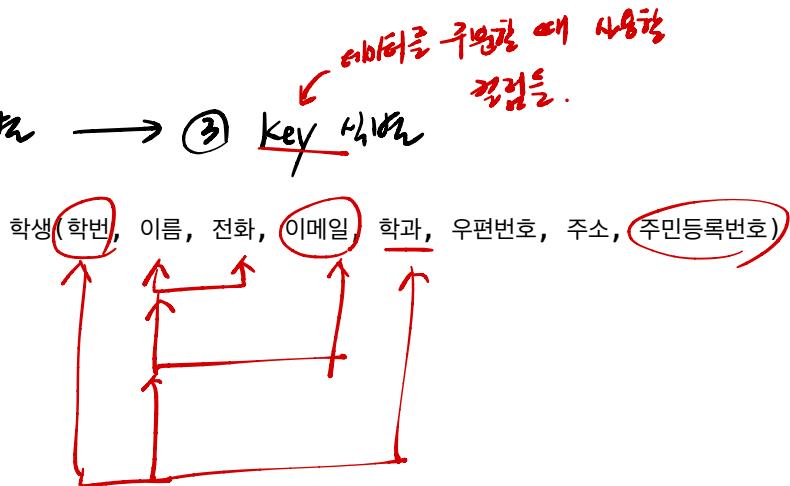
↓
설계

IE 설계 (Information Engineering)
(Crow's Foot 설계 모형을 확장)



* DB 모델링 체계

① Entity 선택 → ② Entity의 속성 선택 → ③ key 선택



- 학번 (0)
- 주민등록번호 (0)
- 이메일 (0)
- 전화 (X)
- 이름 (X)
- (이름, 전화) (0)
- (이메일, 이름) (0)
- (이름, 학과, 학번) (0)
- (이름, 학과, 전화) (0)

* DB 모델링 학자 II

④ 후보 (candidate key) 선정 \longrightarrow ⑤ 기본 키 (주키 ; primary key ; PK) 선정

“후보키” - 학생의 고유한
식별자 가능성이

- 학번
- 주민등록번호
- 이메일
- (이름, 전화) X
- (이메일, 이름) X
- (이름, 학과, 학번) X
- (이름, 학과, 전화) X

Alternate key
(선택키)

PK

- 학번
- 주민등록번호 ← 개인 정보 보호로 노출 불가!
- 이메일 ← 변경할 수 있음

PK처럼
중복되거나
인덱싱 가능



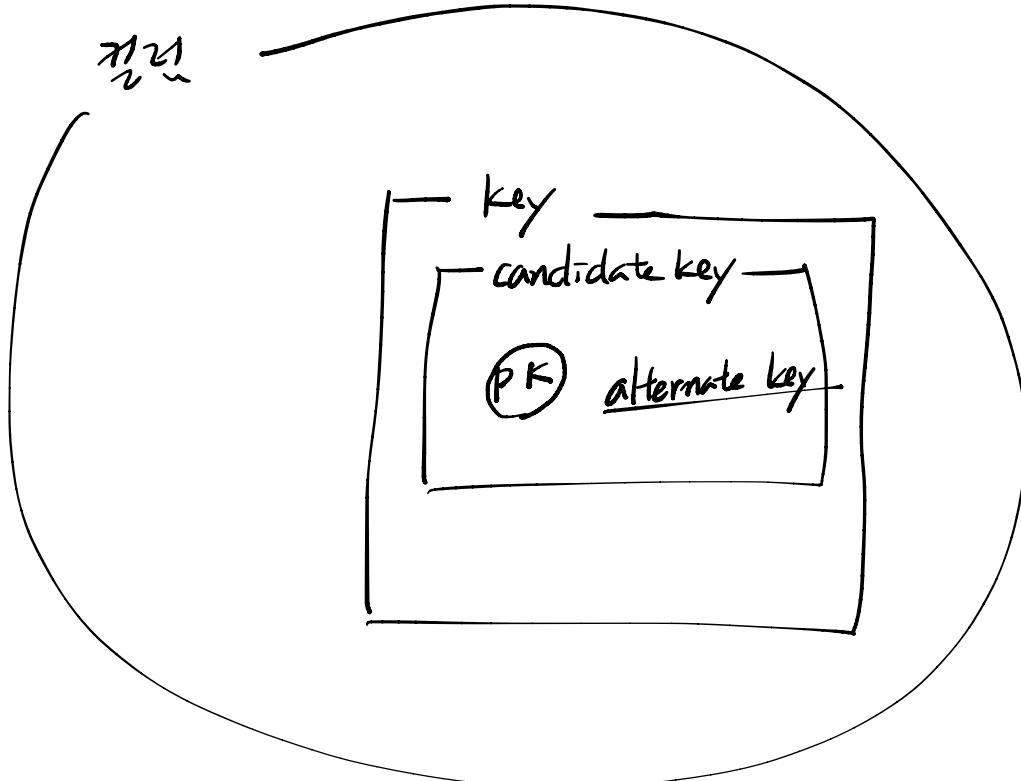
“Unique key”
지정한다.

* 기본 키 같은 다른 테이블의 데이터가 참조하는 경우
변경할 수 있다.



데이터 입력 후 변경하여 원래의
PK로 설정한다.

ключа



+ $ch \neq 1$
(surrogate key)
||
 $al \neq 1$
(artificial key)

* DB 설계 단계 III

대리기 (인공기) → PK로 설정할만한 고유한 값과 같은 경우

인의의 값을 만들기 PK로 사용
① 개시로 일련번호

제품
TV
냉장고
전자레인지
등록번호
주회수
개시로 일련번호

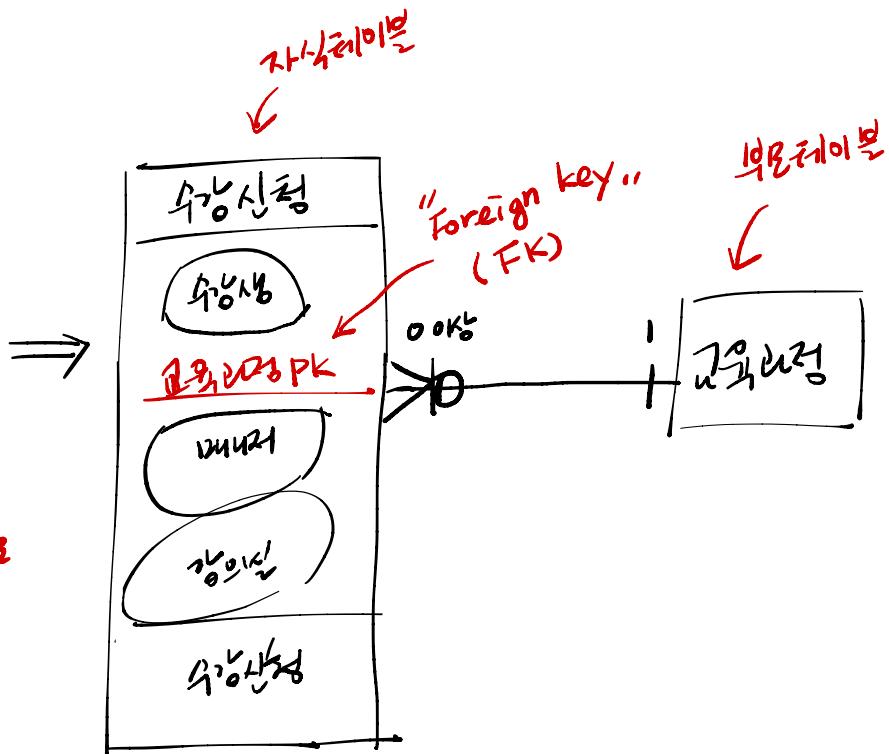
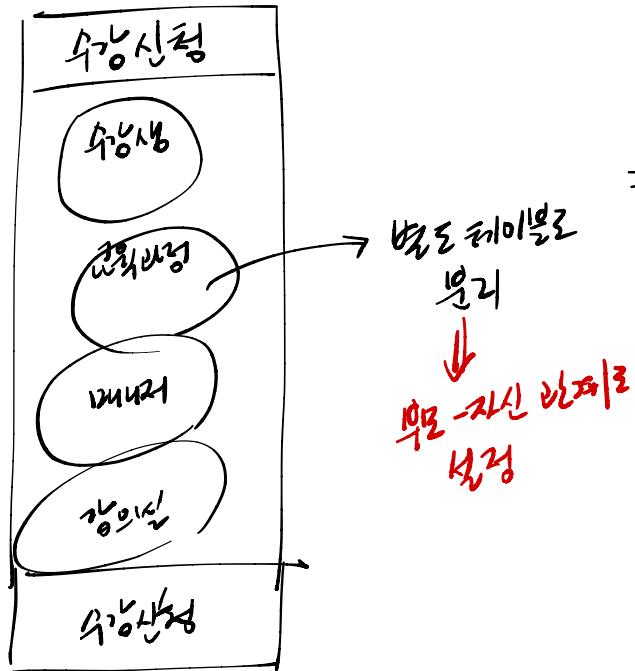
PK로 인증할 때는
필요?

인공기 = 대리기
(artificial key) (surrogate key)

* DB 모델링 대화 IV

⑥ 제1정기화

- ↳ 풍부한 접근하기
- ↳ 풍부한 데이터하기

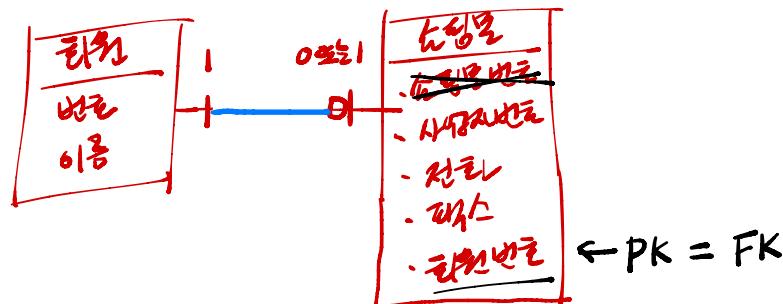


인식하지
않음

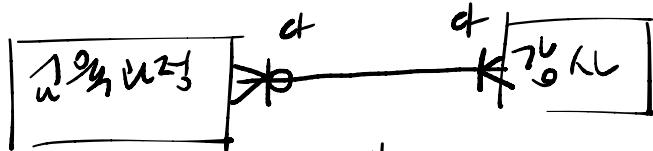
Non-identifying Relationship (비식별 관계) $FK \neq PK$



Identifying Relationship (식별 관계) $FK = PK$



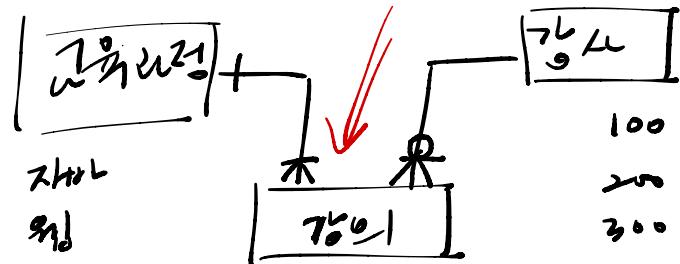
* 대각력 반경 $\frac{R}{\sqrt{2}}$



\leftarrow 대각력 반경 $\frac{R}{\sqrt{2}}$ 차질을 수 있다



"기울기 레이아웃"



100	총길동
200	임계교
300	유발로
400	안중근

- 1 지하철
- 2 철도
- 3 민물로이드

1 100

1 300

2 100

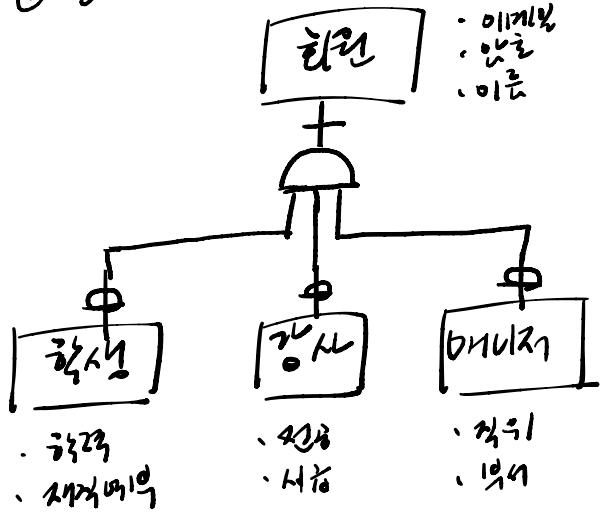
2 200

3 100

↓ 전환
1 층 CL

* 편향화 vs 비편향화

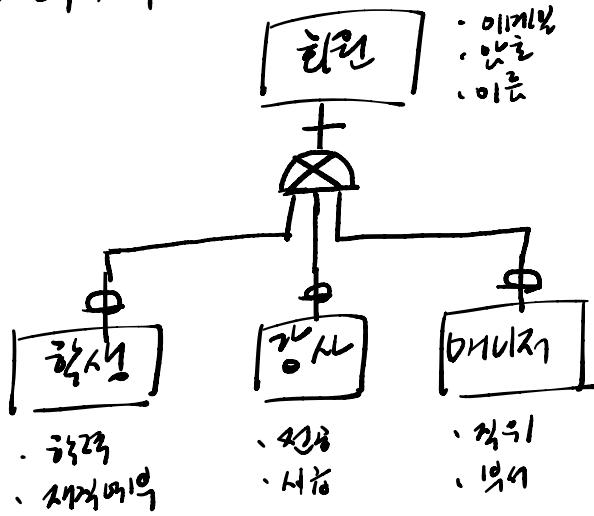
① 편향화 예제



!! 학원+IDM

국어, 영어, 수학 가
총 세에 있을 수 있는 때 !!

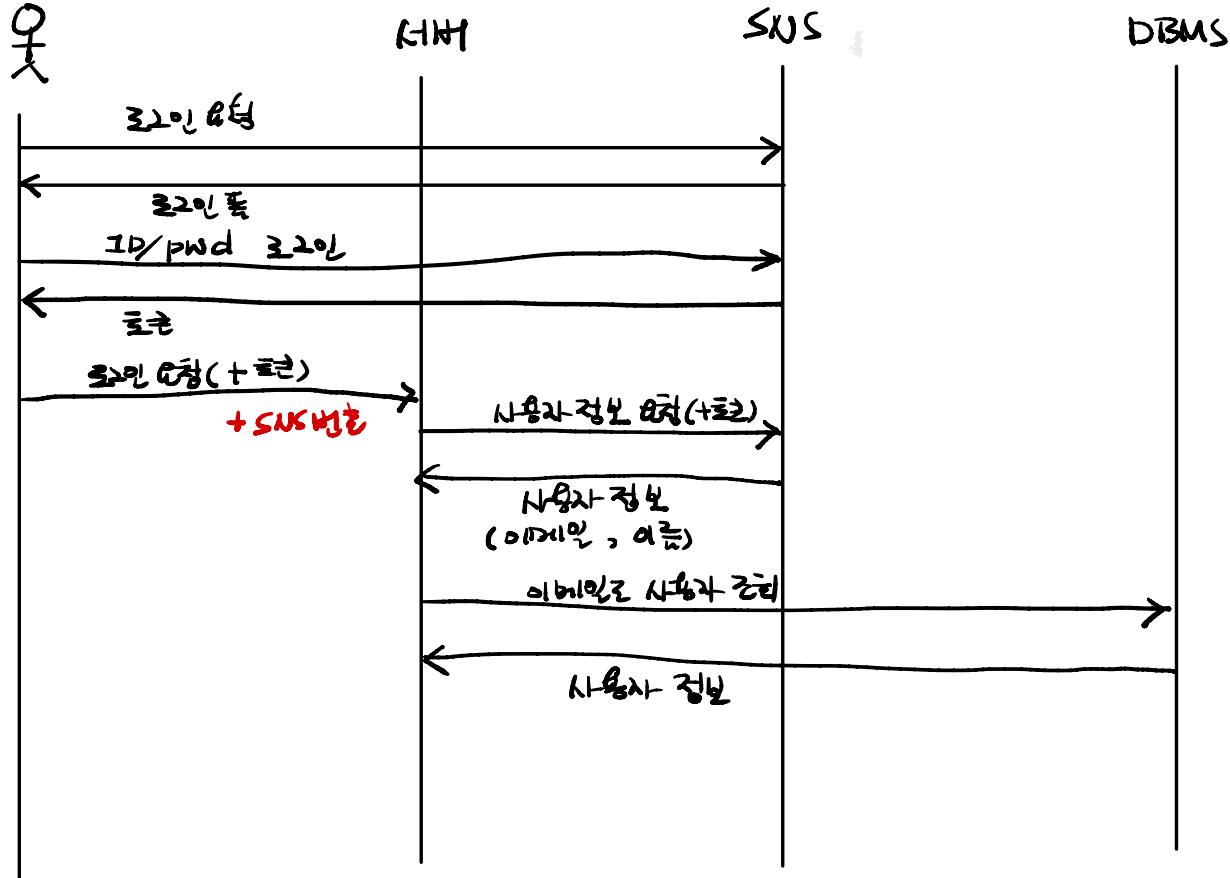
② 비편향화 예제



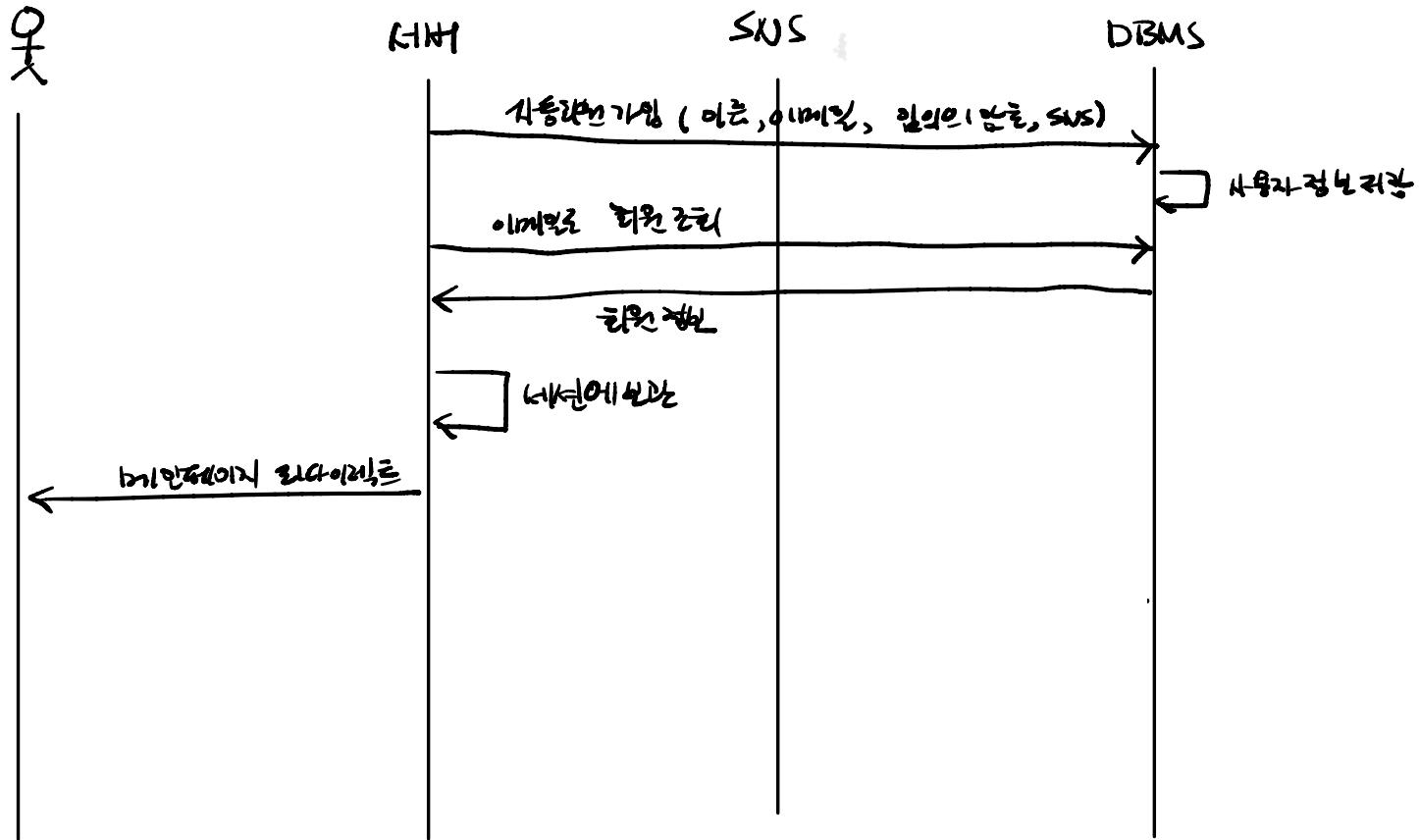
"학원이 국어, 영어, 수학을 있

을 예상할 수 있는 때 !!

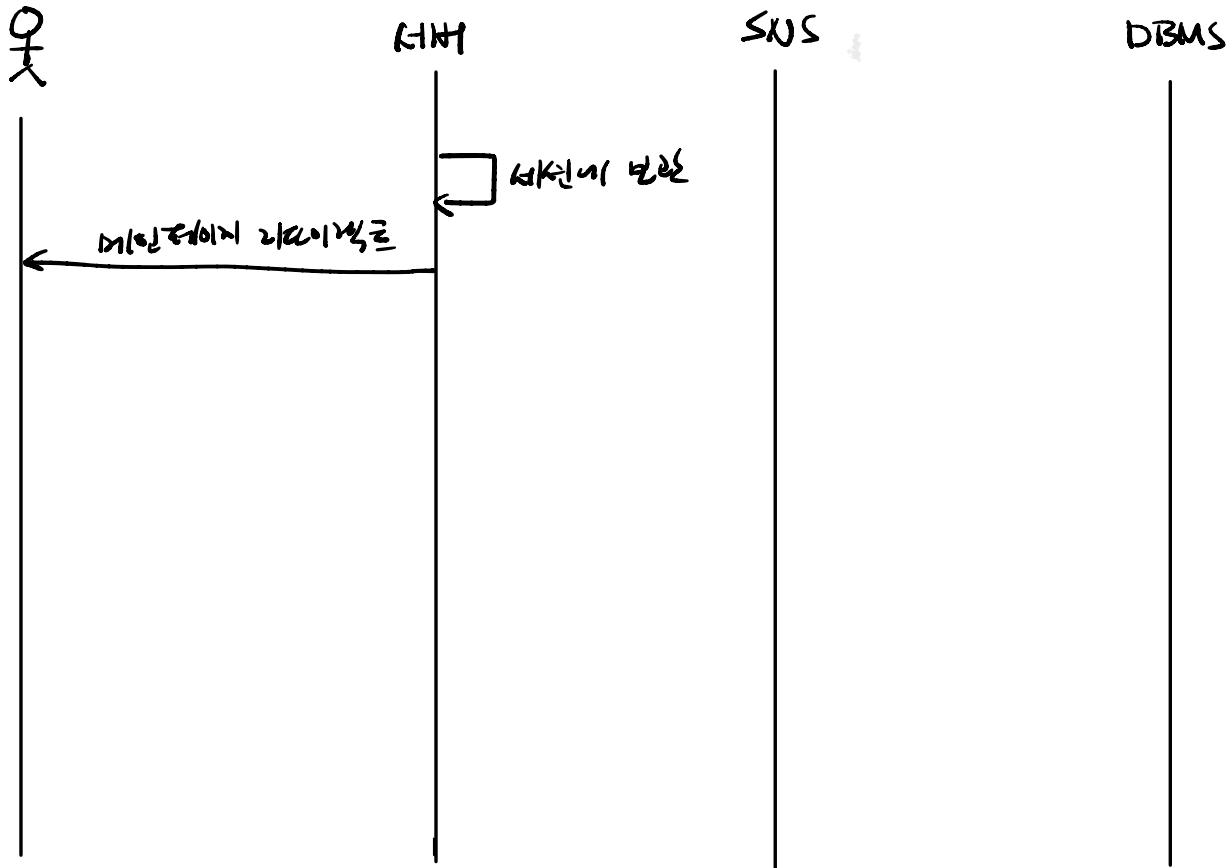
* SNS 로그인 프로세스 - 흐름도



* SNS로 20! 프로세스 - 그단계 : 존재하게 만들 때



* SNS 로그인 프로세스 - 단계 : 존재 확인



10 / 28 Review |

14 : 00

11/27(수) : 프로젝트 시연

11/28(목) : 발표자료 리뷰

11/29(금) :
(1) 발표자료 소개 및 리뷰
시연 동영상 리뷰

12/2(월) : 프로젝트 발표 (리액트(온라인)) → 발표 → 피드백

12/3(화) : 이제부터 리뷰 및 정리

* 프로젝트는 1주일 틈자로

- ① 소개 (가위드 중심으로 간단한 보고) 1 page
- ② 현황 및 문제점 (도표, 통계 등 자료를 활용한 프로젝트 주제는 1주간의 내용을 tm) 1~2 page
- ③ 해결방법 및 이점 (프로젝트 결과물을 얻을 수 있는 이점 보고) 1~3 page
- ④ 주제기능 소개 (화면 개요 + 간단한 설명) 3~5 page
- ⑤ 시연 (영상 + PPT)
- ⑥ S/W 아키텍처 / System 아키텍처
 - S/W 구조
 - 서버 구조 + 외부 서비스(정기, 파일업로드, Email, 앱문서제작 등)
 - 서버에 설치된 App 구조
 - DB 구조 (ERD)
 - Use-case 구조
 - UI 프로토타입

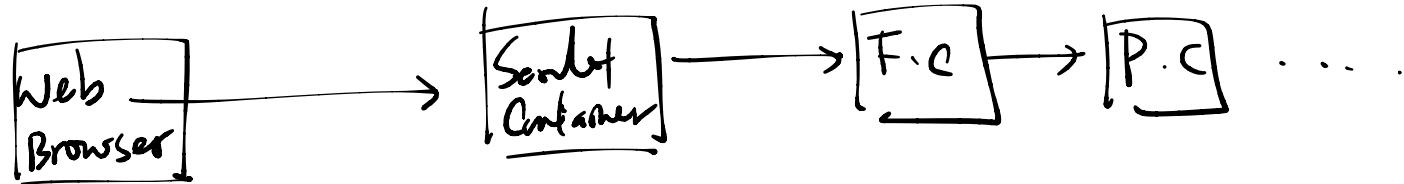
설명,
설명

▷ 그로 학습
+
문제해결(설명, 개선)

||
~~"설명"~~
+
"개선 주제"

- ⑦ 첨부 기록 및 예고
- ⑧ 선별

* S/W Architecture



* System Architecture (App + H/w)

