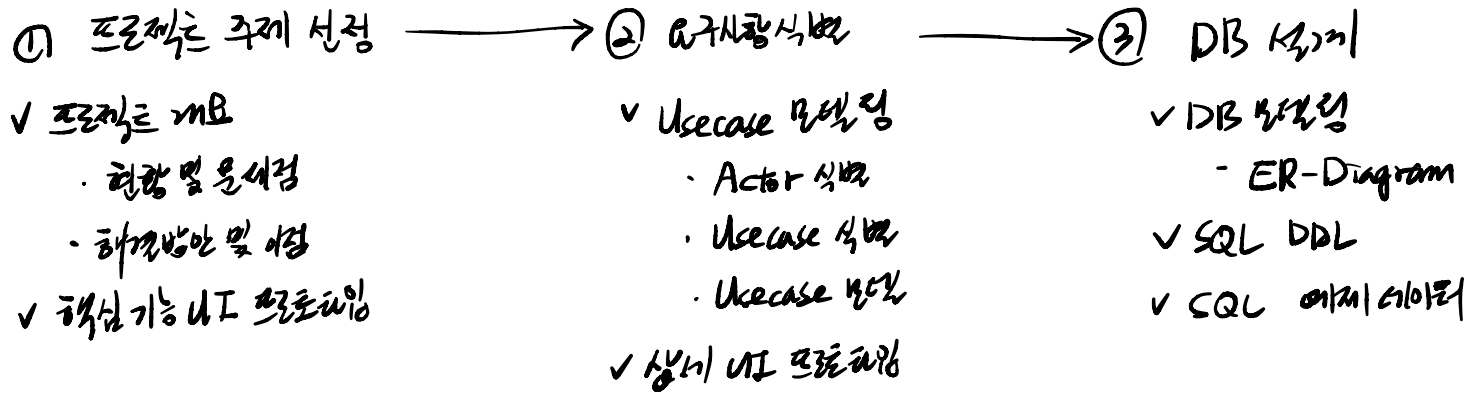
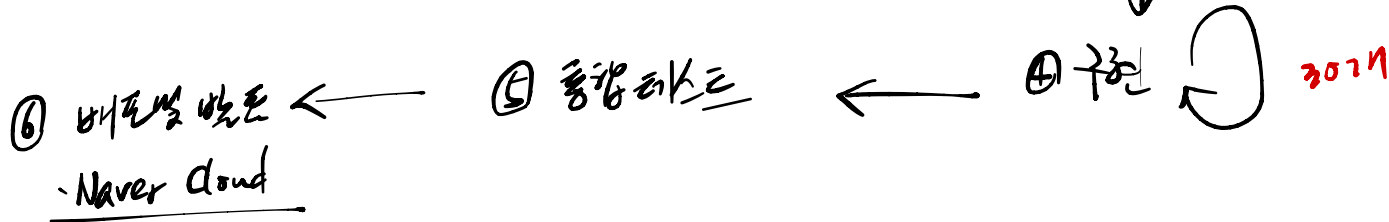


1/2 음 프르펙트

* 절차 및 산출물



9/20



* 주력 방법론

① 기능 별로 작업을 분할

② 2명 → 1개 기능
1개 기능



1명 → 1개 기능

⋮

* Use-case Modeling

① Actor 식별

↗ 시스템을 사용하는 사람 또는 프론트엔드

← 프론트엔드, 외부시스템, 라이브러

← 시스템의 서비스 이용

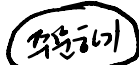
Primary Actor

시스템

Secondary Actor

← 시스템에게 서비스 제공

사람에)



Use Case

↑
Actor의
업무목표

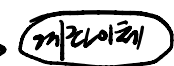


계시시스템

primary actor (프론트엔드)
외부시스템

↓

은행시스템



고객

↑
primary actor
(KDG)

라이버리에)



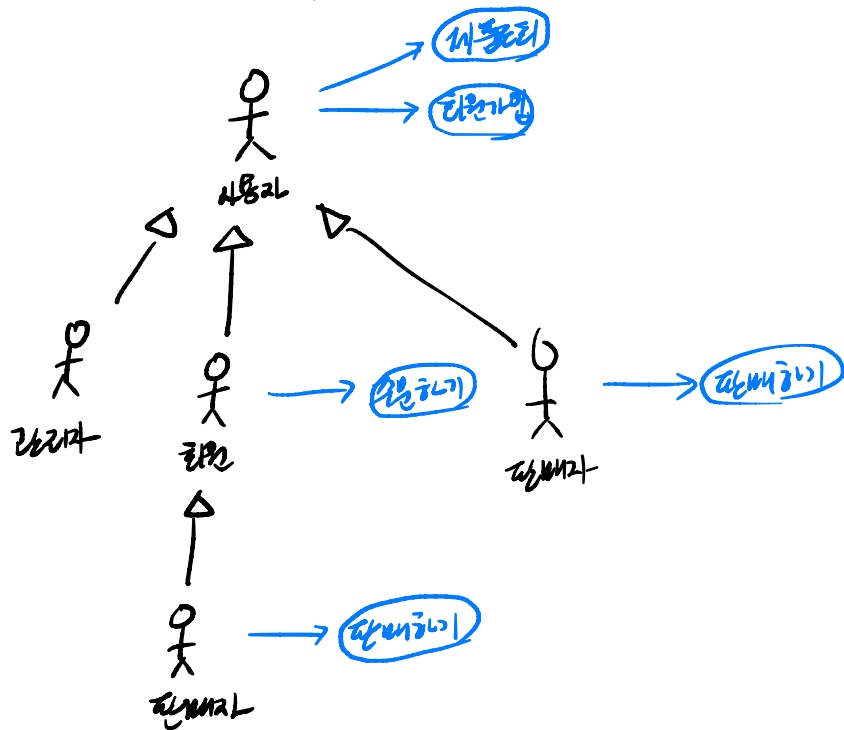
데이터베이스
및 정보관리시스템

* Primary Actor 식별 → Use Case 식별 → Secondary Actor 식별

⊖ Actor 간에 식별 (identify)

↳ 클래스에서 정의하기

예) 시스템



③ Use-case 하이라

↑ 액터가 시스템을 이용하여 달성하려는 업무 목표

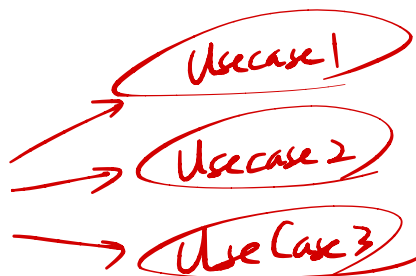
작성 Guide

- ✓ 명목어야 한다 예) 주문하기, ~~로그인하기~~
- ✓ 시작과 끝이 분명
↳ 종료 가능
- ✓ 한 액터가 한 번에 한 번만 수행하는 업무



- 메시지 등록
- 메시지 조회
- 메시지 변경
- 메시지 삭제

Actor를 기준으로
Usecase 하이라



③ Usecase 식별 2

How Guide 2

✓ CRUD 이 해당하는 Usecase 인정 하나로 통합

- 게시글 등록
- 게시글 조회
- 게시글 변경
- 게시글 삭제

⇒ "Xxx 관리" 로 통합한다

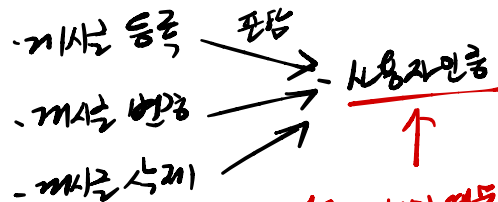
예) 게시글 관리

✓ 여러 Usecase 이 통합되는 시나리오 인정
1명의 Usecase로 분리

✓ 관련 Usecase를 하나로 통합한다.

- 로그인
 - 로그아웃
- } → 사용자 인증하기

- 게시글 등록
↳ 사용자 인증
- 게시글 변경
↳ 사용자 인증
- 게시글 삭제
↳ 사용자 인증



업무가 아니어서
Usecase가 될 수 없게만
변경되는 내근
분리

