

휴먼 디자인 프로젝트(웹/모바일) 프로세스

- 애자일 프로세스(스크럼)

—



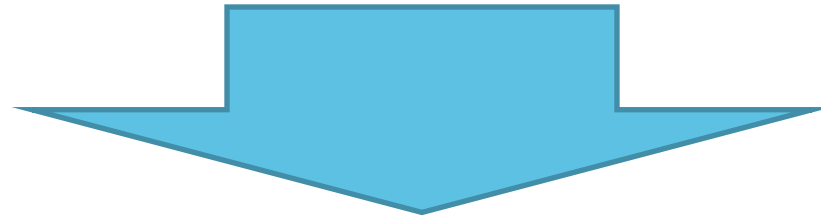
애자일이란?

사람 중심

의사소통

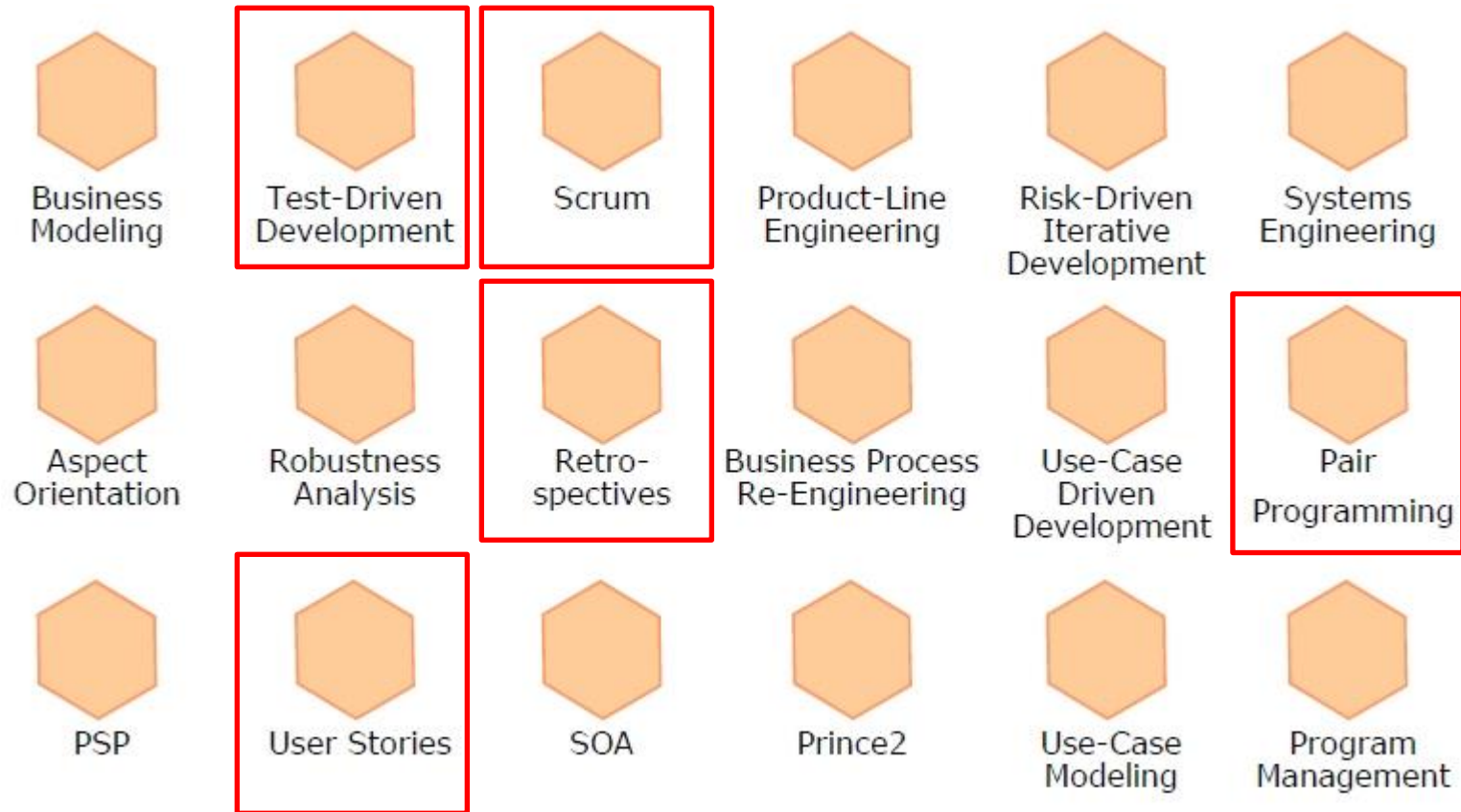
협업

신뢰



품질 좋은 제품

Context에 따라 다른 실천 방법 적용

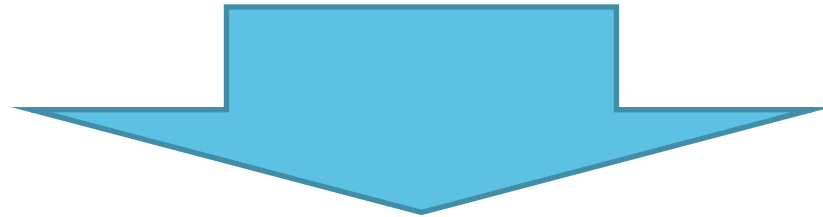


사람 중심

의사소통

협업

신뢰



프로젝트 시작 : 어떻게 형성할 것인가?

애자일 원칙

- 결정을 최대한 뒤로 미뤄라
- 고객에게 최대 가치를 줄 수 있는 것을 먼저 개발한다.
- 기능, 개발 일정 등 모든 상황은 언제든지 변할 수 있다.
- 변화에 빠르게 대응할 수 있어야 한다.

스크럼(Scrum)

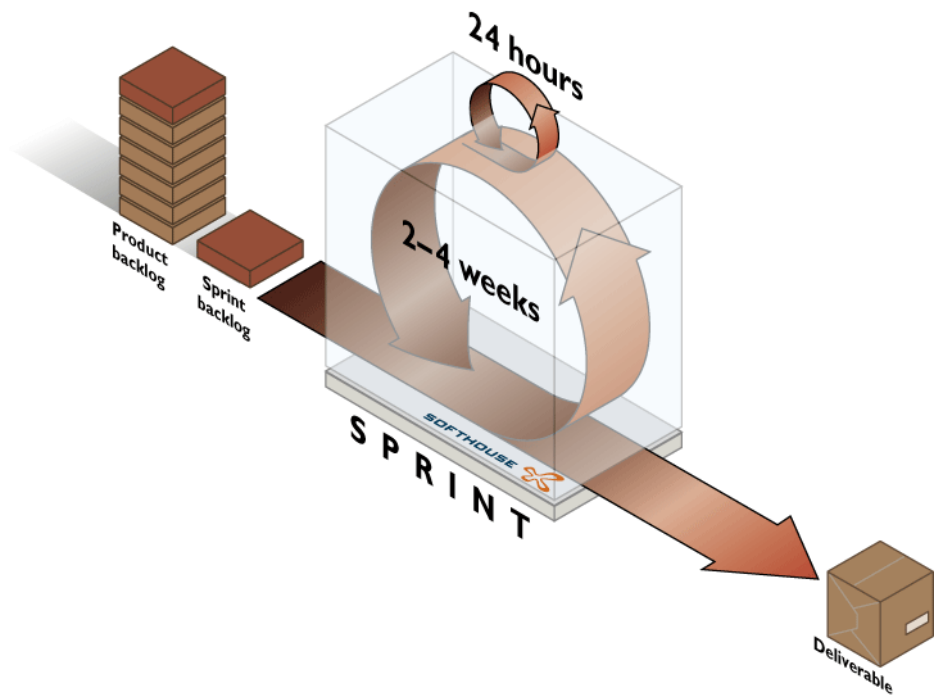
스크럼 용어

스프린트(sprint) : 반복주기를 의미함. Milestone, iteration 용어도 종종 사용

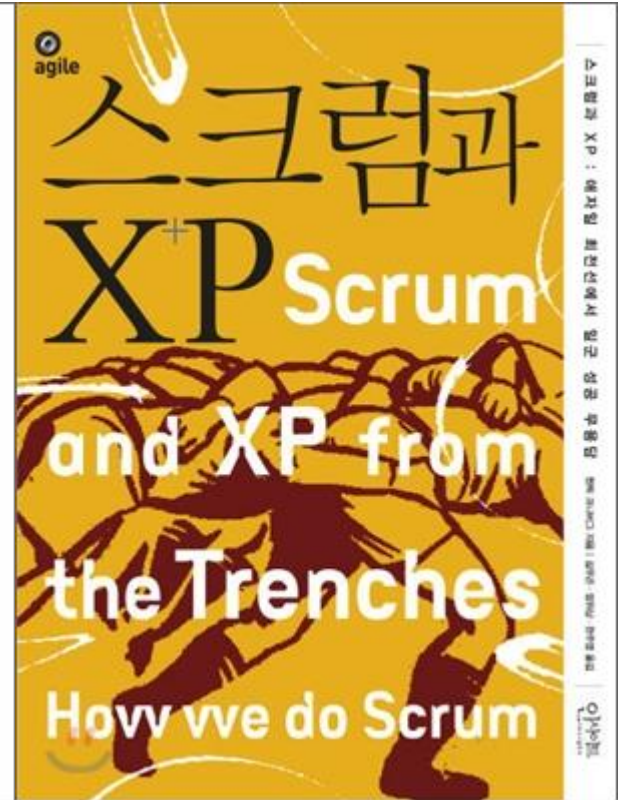
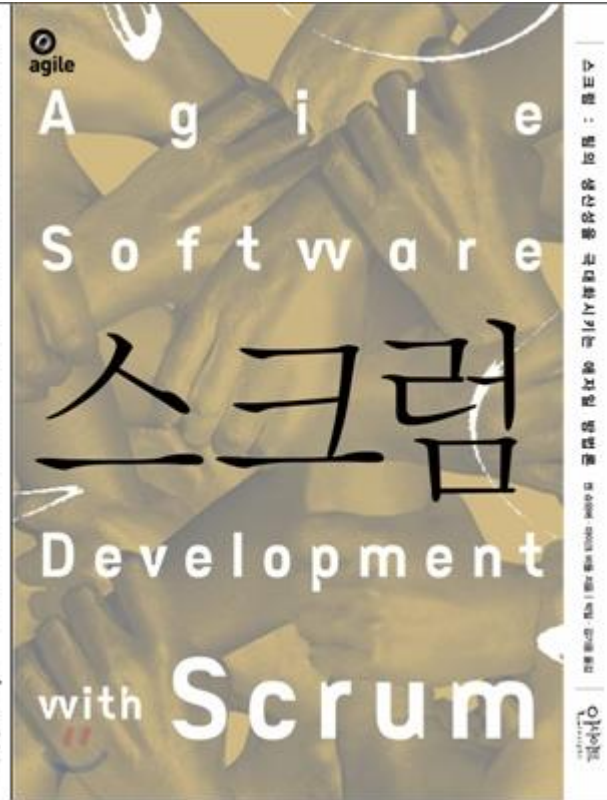
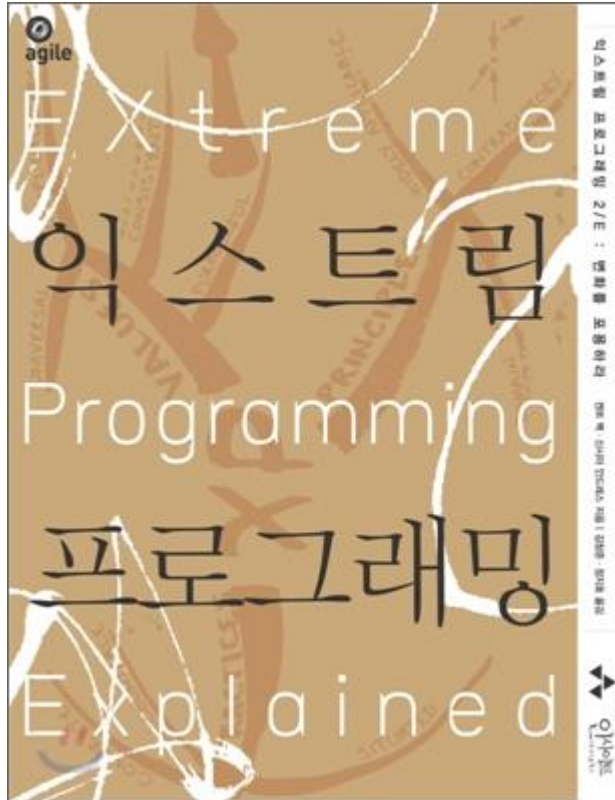
백로그(backlog) : 구현할 기능 목록을 의미함. Feature, User Story 용어도 종종 사용

중요도 : 백로그의 중요 정도. 10 또는 150과 같이 점수로 나타냄. 숫자가 클수록 더 중요하다는 의미

추정치 : 다른 백로그와 비교해 이 백로그를 구현하는데 상대적인 개념의 시간. 추정치를 Story Point라고도 부른다.



참고 서적





Sprint 0

(휴디 1주 ~ 4주)

Sprint 0(zero)란?

프로젝트를 진행해 보면 실질적으로 애플리케이션을 개발하기 전에 **요구사항을 분석하고, 기능 추출, 우선 순위, 일정 산정, 개발 환경을 세팅**하는 작업들이 반드시 따르기 마련이다.

실질적으로 가치있는 결과물을 만들어 내는 전 단계이기 때문에 이 단계를 Sprint 0로 관리해 프로젝트를 진행할 수 있다.

Sprint 0(1-4주) 할 일

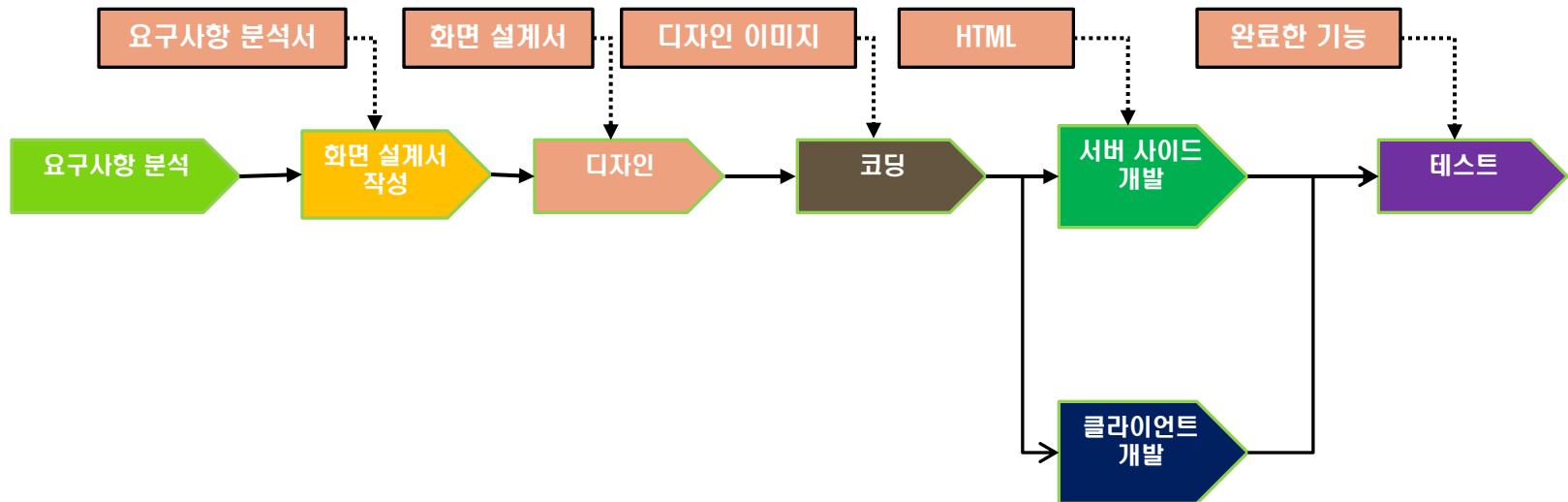
- 팀원 구성
- 지도 교수 배정
- 지도 교수 통해 github repository 추가, 팀원 추가
- 팀 원칙 만들기(소통, 협업, 신뢰를 어떻게 형성할 것인가?)
- 팀의 역할 및 책임 정의
- 수강 신청할 과목 결정해 지도 교수에게 공유
- 스터디 및 학습 계획서 작성
- 프로젝트 요구사항 분석
- 기능 목록 추출
- 우선 순위 및 일정 산정
- 로컬 개발 환경 세팅
- 개발 서버 환경 세팅
- 지속적 통합 도구 세팅
- 지속적 배포 환경 세팅

작업 관리

(이슈 관리 시스템)

모든 작업 목록을 이슈관리시스템으로 통합

이슈관리시스템(Github)





Sprint 0 작업 목록 정리



- 지도 교수 통해 github repository 추가, 팀원 추가
- 팀 원칙 만들기(소통, 협업, 신뢰를 어떻게 형성할 것인가?)
- 팀의 역할 및 책임 정의
- 수강 신청할 과목 결정해 지도 교수에게 공유
- 스터디 및 학습 계획서 작성
- 프로젝트 요구사항 분석
- 기능 목록 추출
- 우선 순위 및 일정 산정
- 로컬 개발 환경 세팅
- 개발 서버 환경 세팅
- 지속적 통합 도구 세팅
- 지속적 배포 환경 세팅





Sprint 0 작업 목록을 독립된 단위로 구체화

- 수강 신청할 과목 결정해 지도 교수에게 공유
 - 이원호 수강 신청 과목 결정한다.
 - 김주희 수강 신청 과목 결정한다.
 - 최재은 수강 신청 과목 결정한다.
 - 3명 수강 신청 과목 취합해 지도 교수에게 공유한다.
- 스터디 및 학습 계획서 작성
 - 팀만의 스터디 계획서 샘플을 만든다.
 - HTML & Javascript Basic 과목 스터디 계획서 작성한다.
 - Java Web 프로그래밍 과목 스터디 계획서 작성한다.
 - 안드로이드 Basic 과목 스터디 계획서 작성한다.
 - 스터디 계획서 취합한다.
- 로컬 개발 환경 세팅(자바 기반 웹 개발 환경의 경우)
 - 메이븐을 학습한다.
 - 메이븐 학습한 내용을 공유한다.
 - 메이븐 기반 프로젝트를 생성한다.
 - Github에 프로젝트 공유한다.
 - 로컬 개발 환경을 팀원에게 공유한다.



Sprint 0 작업 목록의 우선순위를 선정한다



- 이원호 수강 신청 과목 결정한다. - 1
- 김주희 수강 신청 과목 결정한다. - 1
- 최재은 수강 신청 과목 결정한다. - 1
- 3명 수강 신청 과목 취합해 지도 교수에게 공유한다. - 2
- 팀만의 스터디 계획서 샘플을 만든다. - 3
- HTML & Javascript Basic 과목 스터디 계획서 작성한다. - 11
- Java Web 프로그래밍 과목 스터디 계획서 작성한다. - 4
- 안드로이드 Basic 과목 스터디 계획서 작성한다. - 5
- 스터디 계획서 취합한다. - 12
- 메이븐을 학습한다. - 6
- 메이븐 학습한 내용을 공유한다. - 9
- 메이븐 기반 프로젝트를 생성한다. - 7
- Github에 프로젝트 공유한다. - 8
- 로컬 개발 환경을 팀원에게 공유한다. - 10



모든 작업을 github에 등록한다.

Browse Issues Milestones New Issue

Open Milestones 5
Closed Milestones 0

Create a new milestone

2 with issues 3 without issues Due Date Completeness

sprint 0 1 closed — 4 open Edit Close Delete Browse issues →
Due in 2 days 20%
휴디 수업 시작 전 준비

sprint 1 0 closed — 2 open Edit Close Delete Browse issues →
Due in 9 days 0%
2주차 수업 진행













Github이 우선순위 지원 기능이 부족하다. 우선순위를 관리할 방법을 찾아야 한다.
Github과 다른 도구를 연동해 해결해도 된다.
(이 자체도 하나의 작업 목록이 된다.)





업무 배정은?

(원호가 리더 역할로 프로젝트 전체 조율 및 업무 배정한다.)

- 이원호 수강 신청 과목 결정한다. - 1 
- 김주희 수강 신청 과목 결정한다. - 1 
- 최재은 수강 신청 과목 결정한다. - 1 
- 3명 수강 신청 과목 취합해 지도 교수에게 공유한다. - 2 
- 팀만의 스터디 계획서 샘플을 만든다. - 3 
- HTML & Javascript Basic 과목 스터디 계획서 작성한다. - 11
- Java Web 프로그래밍 과목 스터디 계획서 작성한다. - 4 
- 안드로이드 Basic 과목 스터디 계획서 작성한다. - 5 
- 스터디 계획서 취합한다. - 12
- 메이븐을 학습한다. - 6 
- 메이븐 학습한 내용을 공유한다. - 9
- 메이븐 기반 프로젝트를 생성한다. - 7 
- Github에 프로젝트 공유한다. - 8 
- 로컬 개발 환경을 팀원에게 공유한다. - 10



원효가 리더 역할로 프로젝트 전체 조율 및 업무 배정한다.



push 방식.

현재 대부분의 수직 구조 회사의 업무 배정 방식



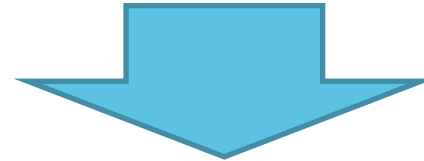


원효가 리더 역할로 프로젝트 전체 조율 및 업무 배정한다.



push 방식.

현재 대부분의 수직 구조 회사의 업무 배정 방식






- 원효는 이 일이 재미있을까?
- 업무를 할당 받는 주희와 재은이는 즐거울까?



작업자가 명확한 경우를 제외하고 업무배정 하지 않는다.



- 이원호 수강 신청 과목 결정한다. - 1 
- 김주희 수강 신청 과목 결정한다. - 1 
- 최재은 수강 신청 과목 결정한다. - 1 
- 3명 수강 신청 과목 취합해 지도 교수에게 공유한다. - 2
- 팀만의 스터디 계획서 샘플을 만든다. - 3
- HTML & Javascript Basic 과목 스터디 계획서 작성한다. - 11
- Java Web 프로그래밍 과목 스터디 계획서 작성한다. - 4
- 안드로이드 Basic 과목 스터디 계획서 작성한다. - 5
- 스터디 계획서 취합한다. - 12
- 메이븐을 학습한다. - 6
- 메이븐 학습한 내용을 공유한다. - 9
- 메이븐 기반 프로젝트를 생성한다. - 7
- Github에 프로젝트 공유한다. - 8
- 로컬 개발 환경을 팀원에게 공유한다. - 10



작업자가 명확한 경우를 제외하고 업무배정 하지 않는다.



pull 방식.

우선 순위에 따라 본인이 본인의 업무를 할당한다.

- 수평 구조에 적합.
- 수직 구조의 경우 이 방식으로 업무 진행할 경우
 - 리더에게는 더 많은 여유를 준다.
 - 팀원은 자신이 주도적으로 업무를 선택한다.(사람의 본성은 남이 시키는 일을 하기 싫어한다.)



산출물 및 상태

산출물

- 지도 교수
- 업무 관리/소스 코드 관리를 위한 github 저장소
- 팀 원칙(소통, 협업, 신뢰를 어떻게 형성할 것인가?)
- 팀의 역할 및 책임
- 개인별 또는 팀별 수강 신청 과목에 따른 스터디 및 학습 계획서
- 프로젝트 요구사항 분석서/기획서
- 기능 목록
- 기능 목록 별로 중요도(우선순위)
- 기능 목록 별로 일정 산정 결과(추정)
- 1차 개인별 개발 저널(3주차에 제출)
- 개발 서버 확보(지도 교수에게 요청)

4주 목요일(3월 27일) 회고 시간의 상태

- 팀원 간의 신뢰 기반 소통, 협업이 가능한 상태
- 개인별 추가 과목에 대한 수강 신청이 완료된 상태
- Github 이슈 목록에 모든 기능이 등록되어 있는 상태
 - 기능 우선순위 별로 개발이 가능한 상태
- 로컬 개발 환경이 준비되어 개발 가능한 상태
- 지속적 통합 도구를 활용해 자동 배포가 가능한 상태
- 개발 서버 환경이 준비되어 배포 및 데모 가능한 상태

마치며...

각 과목별 교수가 지식을 전달하지 않습니다. 여러분이 최적의 학습과 배움에 필요한 환경을 만들기 위해 노력할 겁니다. 학습과 배움에 저해가 되는 부분이 있으면 적극적으로 요청하세요.

지도 교수가 무엇을 하라고 강요하지 않습니다. 여러분이 직접 문제점을 찾고, 해결책을 찾으세요. 그래도 안되면 지도 교수에게 요청하세요.

템플릿이 제공되지 않을 경우 여러분이 직접 만드세요. 그래도 힘들면 지도교수에게 요청하세요.