

# 제4차 산업혁명 시대의 SW교육 혁신, 해법을 찾아서

2019년 1월 29일(화) 16:00~18:00

소프트웨어정책연구소 대회의실

## 학생 주도의 SW 교육 사례 및 시사점

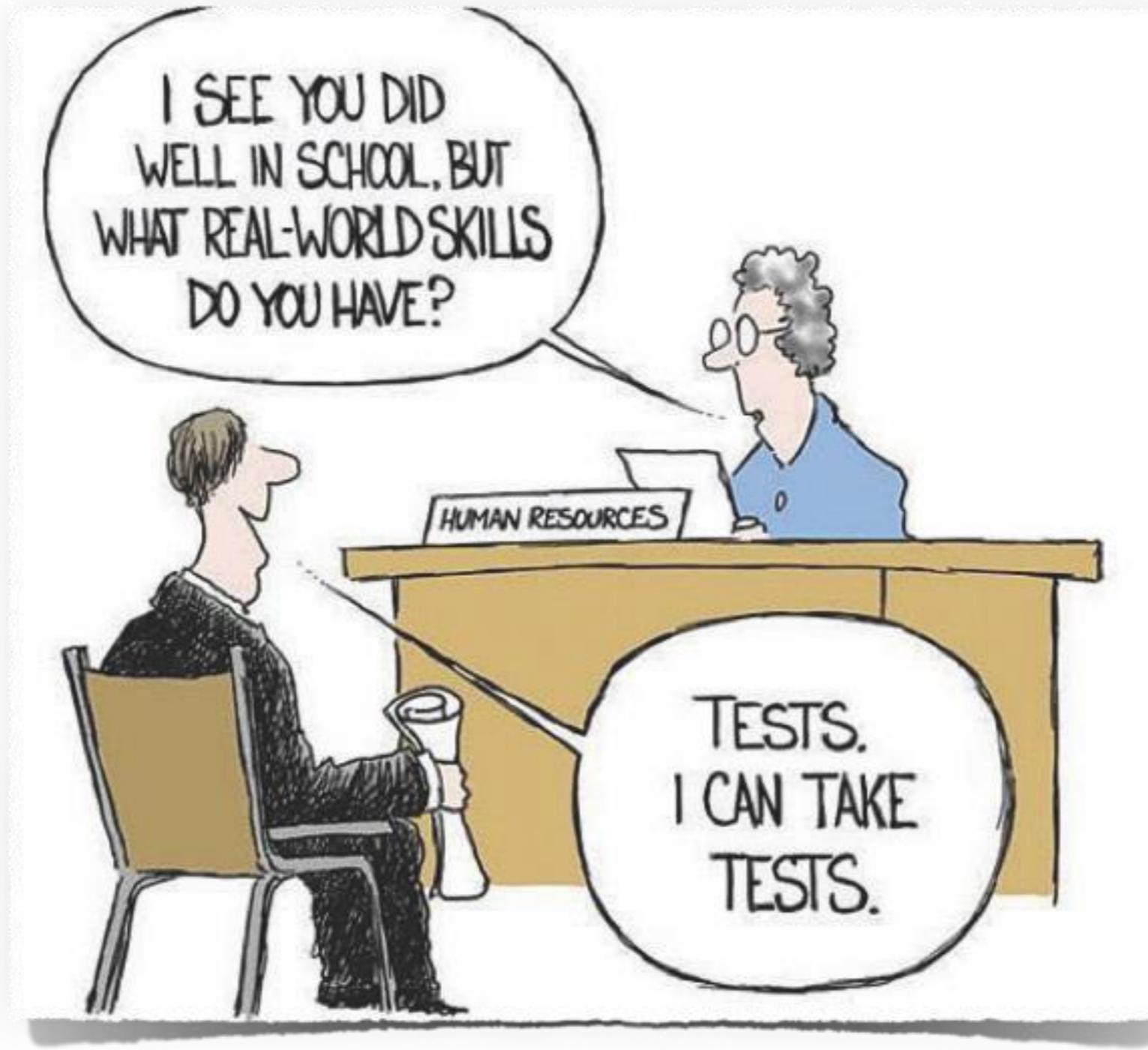
도경구 + Scott Uck-Jin Lee



한양대학교 ERICA 소프트웨어융합대학  
COLLEGE OF COMPUTING  
HANYANG UNIVERSITY ERICA



# 교육 목적 = 실전 역량 키우기



# 전통적인 대학 강의, 실습, 프로젝트



- 문제 풀이 중심
- 학습 활동에 학생의 자발적 참여 유도
- 의사 소통 능력 함양

If a picture is worth a  
thousand words, an  
experience is worth a  
thousand pictures

---

# SW전공교육 사례

교과목	#1: 프로그래밍기초	#2: 소프트웨어공학
교과 유형	기초필수	전공심화
수강 학기	1학년 1학기	4학년 1학기
학점	3학점 = 수업 2시간 + 실습 2시간	4학점 = 수업 3시간 + 실습 2시간
규모	122명	28명
수업 방식	<b>플립드러닝</b> (Flipped Learning) + <b>온라인 실습</b> (CodeOnWeb)	<b>IC-PBL</b> (Industry-Coupled Problem-Based Learning)
수업 내용	제어구조를 사용한 블록구조 중심 프로그램 설계 및 구현 기 법 (Python)	신뢰성 높은 SW를 효과적/효율 적으로 개발하기 위한 개발조직, 관리, 분석, 설계, 구현, 테스트 기법
담당 교수	도경구	Scott Lee

주차	내용
1	오리엔테이션
2	코딩 첫 걸음
3	함수 + 제어 구조
4	재귀와 반복 : 자연수 계산
5	재귀와 반복 : 정렬
6	재귀와 반복 : 검색
7	재귀와 반복 : 동적계획법 + 중첩 반복
8	코딩 시험 1

주차	내용
9	문제 해결 I : Sudoku
10	문제 해결 II : Blackjack
11	문제 해결 III : Blackjack 확장
12	팀 프로젝트 제안 발표
13	안전코딩
14	클래스와 오브젝트
15	팀프로젝트 발표 및 시연
16	코딩 시험 2

	~2017	2018
강의 2시간	재래식 일방향 강의	[플립드 러닝] 강의비디오 사전 시청 수업은 Q&A/토론
실습 2시간	지정된 장소에서 지정된 시간에	온라인 실습 시간/장소 제약 없음
특징	<ul style="list-style-type: none"> <li>강의 내용 이해도 점검 어려움</li> <li>코딩실습 문제 제한 시간 내 완성도 극히 저조</li> <li>코딩 경험이 있는 학생들과의 격차가 커서 초보 학생의 코딩에 대한 자신감 상실</li> </ul>	<ul style="list-style-type: none"> <li>수업 전 강의비디오 시청 필수</li> <li>간단한 퀴즈로 점검</li> <li>Q&amp;A로 강의 내용 이해여부 점검</li> <li>코딩실습 문제 풀이에 충분한 시간 부여</li> <li>전천후 수시 튜토링</li> </ul>

주간 스케줄	교수 + 조교	학생
1일차	강의비디오 + 강의자료 업로드	강의비디오 <b>온라인</b> 시청
2일차	-	
3일차	<b>오프라인</b> 수업 Q&A 2시간 (sli-do 활용) 수업 직후 코딩 실습 문제 공개	
+	실습 Q&A YouTube 방송	코딩실습 수행 <b>온라인</b> 제출 및 자동 채점
5일차	실습 마감 직후 코딩 숙제 문제 공개	코딩 숙제 수행 <b>온라인</b> 제출 및 자동 채점
+	숙제 Q&A YouTube 방송	
+		



CodeOnWeb  
Hanyang ERICA



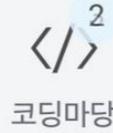
클래스



시험



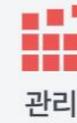
실습



코딩마당



코스



관리

관리



## CSE1017 프로그래밍기초 (2018)

2018년 1학기  
소프트웨어융합대학 소프트웨어학부

학기말 시험 - 개인 노트북 필히 지참  
일시: 6월 15일(금) 13:00~14:45  
장소: 제1학술관 202호

수업 : 금 13:00~15:00 (학연산클러스터지원센터 5층 Space Big) YouTube Live Streaming  
자료 : <https://trello.com/b/4AAqsk8v/cse10172018>

강사 : 도경구, doh@hanyang.ac.kr, 제4공학관 320호 (031-400-5667), 학장실 (031-400-1000)  
조교 : 박건우, pakgunwoo@gmail.com, 제4공학관 319호 (031-400-3781)

CSE1017  
프로그래밍기초

Easy

11  
144



구성원

관리자



도 경 구



평가 항목	비율
출석 (온라인)	6
출석 (오프라인)	6
숙제	12
팀 프로젝트	6
코딩시험 1	35
코딩시험 2	35
<b>합계</b>	<b>100 (%)</b>

### 개선한 사항

- 코딩실습 및 숙제 완성 제출 비율 대폭 증대 (~80%)
- YouTube 실시간 튜토링 시범 시행
- 학생 만족도 증가

### 개선할 사항

- 강의비디오 사전 시청여부 점검 방안 => 자동 체크 + 퀴즈
- Q&A 수업 참여 일부 학생에 제한 => 분반으로 참여 기회
- YouTube 실시간 실습 및 숙제 튜토링 본격 시행
- 실습 및 숙제 풀이 => YouTube 실시간 풀이 방송

주간 스케줄	교수 + 조교	학생
1일차	강의비디오 + 강의자료 업로드	강의비디오 <b>온라인</b> 시청
2일차	-	
3일차	<b>오프라인</b> 수업 Q&A 2시간 (sli-do 활용) 수업 직후 코딩 실습 문제 공개	
+	실습 Q&A YouTube 방송 실습 문제풀이 YouTube 방송	코딩실습 수행 <b>온라인</b> 제출 및 자동 채점
5일차	실습 마감 직후 코딩 숙제 문제 공개	코딩 숙제 수행 <b>온라인</b> 제출 및 자동 채점
+	숙제 Q&A YouTube 방송	
+	숙제 문제풀이 YouTube 방송	

**REAL-WORLD EXPERIENCE  
IS USUALLY  
THE BEST TEACHER**

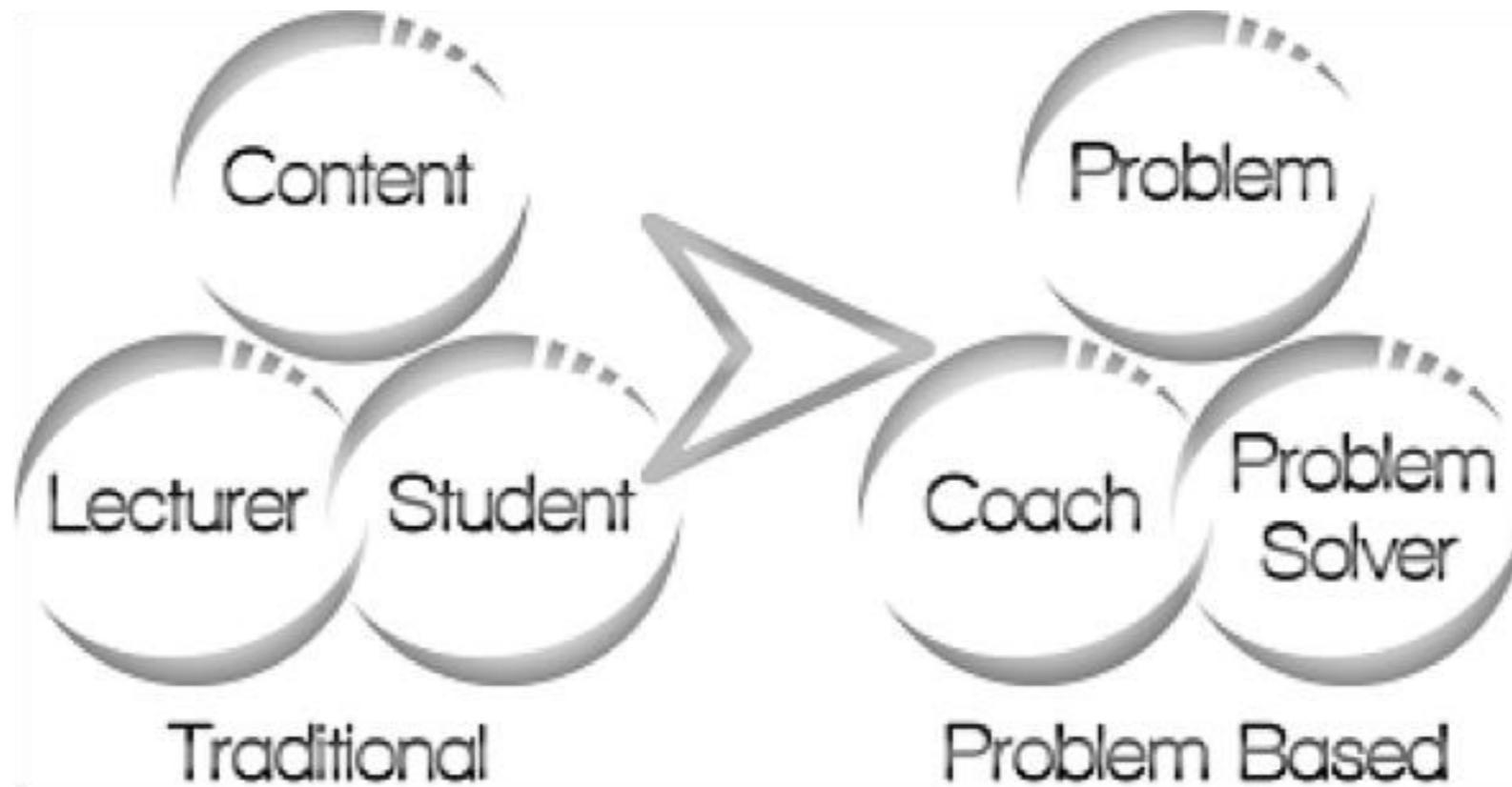


### Problem Based Learning

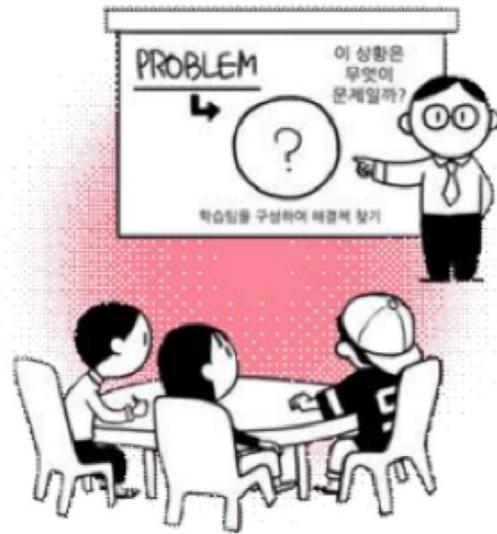
- 문제를 중심으로 수업 상황을 구조화
- 소그룹 학습을 통한 학생들의 능동적 참여
- 학생들이 협력적이고 자기주도적으로 문제를 해결

### Project Based Learning

- 프로젝트를 기반으로 한 목적 지향 학습 활동
- 학생들의 자기 주도적 참여
- 의사 결정권 행사의 기회 및 책임 부여



## 1. 문제 제시 및 팀 구성



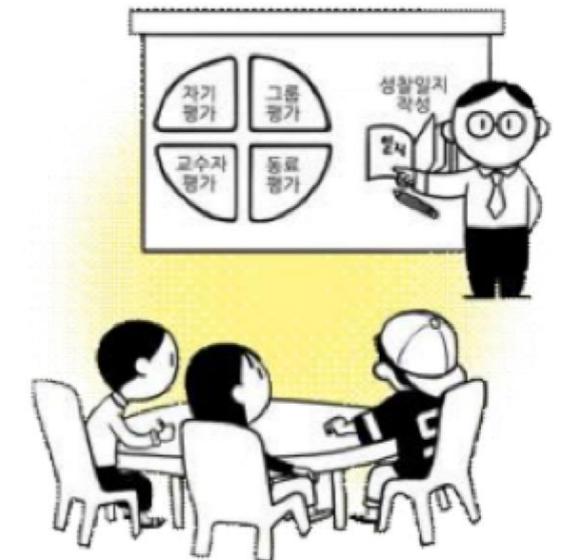
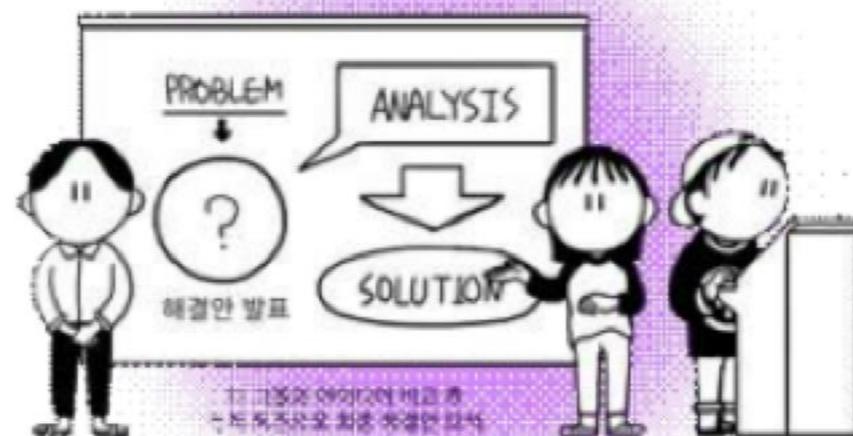
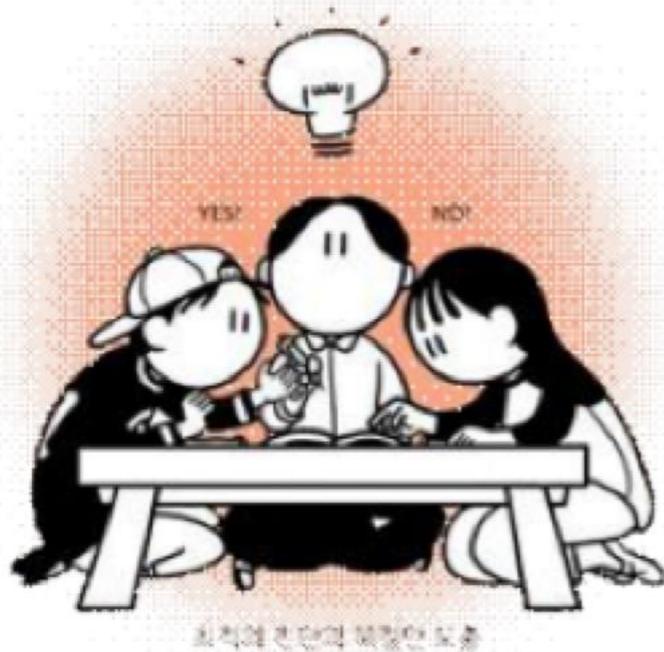
## 2. 문제 확인



## 3. 문제 해결을 위한 자료 수집



## 4. 문제 재확인 및 해결안 도출 문제 해결안 작성 및 발표 6. 학습결과 정리 및 평가



- 수업 진행 방식
  - **강의** : 수업의 전반 1.5 ~ 2시간
    - 문제 시나리오의 해결책 도출을 위하여 기본적으로 요구되는 소프트웨어공학의 기초적이며 필수적 이론 강의
  - **PBL 활동** : 수업의 후반 3 ~ 3.5 시간
    - 문제 파악, 문제 해결을 위한 협업 학습, 해결안 도출, 발표, 성찰, 평가 등 학생들에 의해 주도되는 전형적인 PBL 수업 방식으로 진행
- 참여자의 역할
  - 교수 : 중재자 및 멘토
  - 튜터 : 팀 활동을 지원
  - 학생 : 문제 파악, 분석, 해결 등 전적으로 담당

주차	수업 내용	강의	실습	강의	실습	강의	실습
1	강의 강의 계획 소개 / 자기소개 - Lecture Overview IC-PBL - Introduction to Software Engineering	6	PBL 문제해결을 위한 이론 강의 진행 및 자료수집 - Software Requirements Specification (SRS) 미니 PBL 2 문제 해결을 위한 협력학습 - 팀별 프로젝트에 적합한 요구공학 기법과 기법에 따라 명세하게될 SRS 내용 조사 및 선정	12	PBL 문제해결을 위한 강의 진행 및 자료수집 - Software Design and Implementation II 미니 PBL 3 문제 해결을 위한 협력학습 - 팀별 프로젝트에 적합한 최적 및 상세 설계 기법과 이에 기반하여 진행될 효과적인 구현 방법 조사 및 선정 (팀별 프로젝트에 중요한 / 필수적 설계 산출물 선정)	강의	실습
	실습 - 소프트웨어 개발 프로젝트 실패의 원인 (소프트웨어 공학의 필요성)						
2	강의 PBL 이해하기 - 프로젝트 문제 시나리오 제시 - 교육용 소프트웨어 개발: 팀별 프로젝트 시나리오 제시 (미니 PBL 세부 문제(시나리오) 포함에 대한 언급) 팀 구성 및 팀 소개	7	PBL 문제해결을 위한 강의 진행 및 자료수집 - Project Management : 선택한 개발 방법론에 따라 효과적인 프로젝트 관리 기법 조사, 분석, 지정	13	PBL 문제해결을 위한 강의 진행 및 자료수집 - Software Testing : 프로젝트에 효과적인 testing 기법 조사, 분석 및 적용	강의	실습
	실습 - 프로젝트 시나리오 파악, 이해, 조사, 분석에 거쳐 구성된 팀에 적합한 소프트웨어 개발 프로젝트 주제 선정 - 팀 환경 개발 도구인 Trello & Git 활용 능력 습득						
3	강의 PBL 문제에 대한 이론 강의 진행 - Software Development Process and its Model	8	PBL 프로젝트 문제 해결안 중간 점검 - 프로젝트 산출물 작성, 검토, 수정, 보완 - 중간 발표 자료 작성 / 검토 및 발표 / 토론	14	PBL 문제해결을 위한 강의 진행 및 자료수집 - Product Metrics & Quality Assurance : 대상 소프트웨어 개발에 가장 적합한 적절한 설계 도출\	강의	실습
	실습 미니 PBL 1 문제 (시나리오) 제시 및 문제 파악 - 팀별로 선정된 프로젝트에 가장 적합한 개발 프로세스 / 방법론 / 모델 조사 및 효과적인 적용에 대한 고찰						
4	강의 PBL 문제해결을 위한 이론 강의 진행 및 자료수집 - Agile Development 미니 PBL 1 문제 해결을 위한 협력학습 - 다양한 Agile Development 방법론 중 팀별 프로젝트에 가장 적합한 방법론과 효과적 적용 방법 조사 및 선정	9	PBL 문제해결을 위한 강의 진행 및 자료수집 - Analysis Modeling & Design Engineering	14	미니 PBL 4 문제 해결을 위한 협력학습 - 팀별 프로젝트에 적합한 Testing, Metrics, Refactoring, Quality Assurance 관련 기법 조사 및 선정	강의	실습
	실습 미니 PBL 1 문제 해결안 발표 및 평가 - 팀별 프로젝트에 적용될 개발 방법론 발표 및 평가 - 팀별 프로젝트에 적용될 개발 방법론 평가						
5	강의 PBL 문제에 대한 이론 강의 진행 - Requirement Engineering : Elicitation, Elaboration, Negotiation, Specification, Validation	10	PBL 문제해결을 위한 강의 진행 및 자료수집 - Architectural and Detailed Design Development	15	PBL 프로젝트 문제 최종 해결안 도출 - 프로젝트 산출물 작성, 검토, 수정, 보완 - 최종 발표자료 작성 / 검토 및 발표 / 토론	강의	실습
	실습 미니 PBL 1 문제 해결안 발표 및 평가 - 팀별 프로젝트에 적용될 개발 방법론 발표 및 평가 - 팀별 프로젝트에 적용될 개발 방법론 평가						
5	강의 PBL 문제에 대한 이론 강의 진행 - Requirement Engineering : Elicitation, Elaboration, Negotiation, Specification, Validation	11	PBL 문제해결을 위한 강의 진행 및 자료수집 - Software Design and Implementation I 미니 PBL 3 문제 (시나리오) 제시 및 문제 파악 - 팀별 프로젝트에서 소프트웨어를 효과적으로 디자인, 설계, 구현하기 위한 기법에 대한 고찰	16	PBL 최종 보고서 작성	강의	실습
	실습 미니 PBL 2 문제 (시나리오) 제시 및 문제 파악 - 팀별 프로젝트 요구사항을 최대한 효과적으로 하지만 적당하게 도출, 협상, 명세, 검증하는 방법에 대한 고찰						

1

2

3

4

- 팀 구성 방식
  - 5 ~ 6인으로 총 5 팀 구성
  - 소프트웨어 개발 능력, 현업 실무 경험, 관심 분야 등을 고려
  - 팀 전체의 역량이 균등할 수 있도록 교수가 일부 조정한 후 학생들의 의사를 반영하여 구성
  - IC-PBL 모듈 (총 4개의 Mini-PBL)을 동일 팀으로 한 학기 동안 진행
  - 소프트웨어 프로젝트 관리 역량을 사전 지식, 경험, 잠재력을 기준으로 판단하여 팀 별로 팀장 1인을 선출
  - 팀 구성원은 개별 능력 및 프로젝트 주제 및 진행 방식에 따라 소프트웨어 설계자, 개발자, 테스터 등으로 역할을 구분
  - 프로젝트 진행 시
    - 개인 : 문제 분석지, 성찰 일지, 팀 내부 평가서를 작성 및 제출
    - 팀 : 팀 활동 보고서, 문제 해결안 (산출물), 팀 간 평가서를 작성, 제출, 발표

- Week 1
  - 강의 및 강의 계획 소개
  - 수강생 소개
- Week 2
  - IC-PBL Problem (Scenario) 제시
  - Team building

집 만들기 : SW 개발 프로세스의 효과적인 적용에 대한 고찰



### Week 3 ~ 7 & 9 ~ 14

- Mini-PBL 1 ~ 4 진행

- 문제 제시
- 문제 파악

- 문제 해결을 위한

- 이론 강의
- 자료 수집
- 협력 학습

- 문제 해결안 발표
- 문제 해결안 평가



- Mini-PBL 1

- 문제 제시

주어진 프로젝트의 시나리오를 바탕으로 팀에서 선정한 주제 (소프트웨어 제품 및 서비스 개발)에 가장 적합하며 효과적인 *Agile* 개발 방법론 기반의 소프트웨어 개발 프로세스를 선정하고 이를 프로젝트 특성들에 맞게 적용하는 방법 및 방안 도출한다.

- 목표 :

- 팀 별 협업 학습을 통하여 다양한 Agile 개발 방법론에 대한 지식 습득
- 소프트웨어 개발 프로젝트 진행 시 개발 프로세스의 중요성 인지
- 현 소프트웨어 산업에서 적용되고 있는 Agile 개발 방법론의 실제 사례 분석을 통하여 간접적 경험 및 관련 지식 습득
- 소프트웨어 개발 프로젝트의 컨텍스트, 특성, 범위를 고려한 최적의 개발 프로세스 선정 및 적용 (수정 및 개선 포함) 능력 배양

## • Mini-PBL 1

### ◦ 문제 파악 (개별 + 팀별)

학번 : ( 2013043177 ) / 성명 : ( 김 주 호 )

② 생각(idea/hypothesis) = To-Be	① 사실(fact) = As-Is	③ 학습과제(learning issues)	④ 실천계획(action plans)
<p>시나리오에서 나타나는 가장 주된 문제는 프로젝트의 요구사항이 명확하지 않으며 또한 요구사항이 시시각각으로 변화함에 있다.</p> <p>첫 번째로 요구사항을 구체화하기 위해 전 주에 러프하게 구성한 프로젝트 제안서를 2개를 제출했으며 2개의 주제 (온라인 강의 플랫폼, 교육용 웹 콘텐츠 제작 도구)를 적절히 융합할 것을 요구 받았다.</p> <p>두 번째로 이 프로젝트는 개발기간이 1년이며 유지보수가 20년에 이르는 장기 프로젝트이다. 따라서 개발 중 여러 가지 요구사항이 나타나거나 변화하며 이는 유지보수 중에도 마찬가지이다. 따라서 고품질의 신뢰성 있는 소프트웨어의 개발을 위해 프로젝트에 맞는 적절한 개발 프로세스의 선정 및 운영이 필요하다.</p>	<p>제시된 시나리오에서 학습자는 현재 (2018년 3월) LES사의 B2B부서에서 소프트웨어 개발 프로젝트를 담당하고 있는 5년차 경력의 소프트웨어 엔지니어이며 B2B부서에서의 역할은 팀을 이루어 다른 회사에서 의뢰하는 소프트웨어 시스템을 개발하는 것이다. LES사 B2B 부서는 한양대학교로부터 교육용 소프트웨어 시스템 개발이라는 중대규모 프로젝트 의뢰를 받게 되었다. 한양대학교는 교육용 소프트웨어를 필요로 하며 필요 예산도 충분히 있으나 정확히 어떠한 시스템이 필요한지에 대해서 구체적인 요구사항을 명시하지 못하고 있어 반대로 LES사 B2B부서로부터 한양대학교에 시스템 제안을 받고자 함. 이를 위해 LES사가 자체적으로 의견을 제시하고 세부요구사항 도출을 위해 교수 1인과 교직원 1인을 배정하였다. 프로젝트 제안기간은 3개월이며 개발기간은 1년, 유지보수는 20년이다. 비용은 총 100억이며 진행여부는 한양대학교에서 검토하여 결정한다.</p>	<p>과거부터, 심지어 현재에 이르러서도 많은 소프트웨어 개발 프로젝트가 개발 프로세스로서 Waterfall Model을 사용하고 있다. 하지만 폭포수 모델을 사용하면 요구사항의 불확실성 수용에 어려움이 있으며 고객의 요구사항에 적합한지 확인해보기까지의 시간이 매우 오래 걸린다. 또한 선형 특성을 가지고 있어 막힘 상태에 도달하게 되어 비효율적으로 프로젝트가 진행하게 될 가능성도 있다. 따라서 폭포수 모델을 이 프로젝트의 개발 프로세스로서는 사용할 수 없다.</p> <p>따라서 이 시나리오의 프로젝트에 사용할 개발 프로세스는 가치 있는 소프트웨어를 빠르게 그리고 지속적으로 전달해서 고객을 만족시키며 연체라도 요구사항의 변경을 수용할 수 있는 Agile Software Development를 사용할 것이다.</p>	<p>앞으로 애자일 개발 프로세스에 대해서 학습하여야한다. 애자일은 변화에 민감하게 대처할 수 있는 개발 방법론으로 여러 방법론(스크럼, 익스트림 프로그래밍, 적응형 소프트웨어 개발 방법론, 린 소프트웨어 개발 방법론, 크리 스틸 패밀리, 기능 주도 개발 방법론, 동적 시스템 개발 방법론, 애자일 UP) 등이 있다.</p> <p>위의 개발 프로세스는 애자일이라는 기치를 같이할 뿐 세부 내용은 차이가 있기 때문에 위의 프로세스에 대해서 알아보고 시나리오의 프로젝트에 적합한 프로세스를 선택해야한다.</p>
문제의 원인, 결과, 가능한 해결안에 관한 학습자의 가설이나 추측 검토	문제에 제시된 사실과 학습자가 알고 있는 문제 해결과 관련된 사실 확인	문제를 해결하기 위해 학습자가 학습해야 할 필요가 있는 학습내용을 선정	문제를 해결하기 위해 학습자가 이후에 해야 할 일 또는 실천계획

[①→②→③→④] 순서대로 진행 ⇒ ①②를 근거로 ③의 학습과제를 도출하고 수행하다보면 또 다른 가설(②)이 세워질 수도 있고 가설이 사실(①)로 확인되기도 한다. ⇒ 그렇게 되면 다시 ①②③이 수정되거나 보완되면서 반복적으로 수정 보완을 통해 훨씬 더 정교화된 문제 분석결과가 나온다. ⇒ 보통 두 번 정도 이러한 사이클이 이루어진 후 ①②③이 확정되면 이를 바탕으로 실천계획(④)을 세운다.

## • Mini-PBL 1

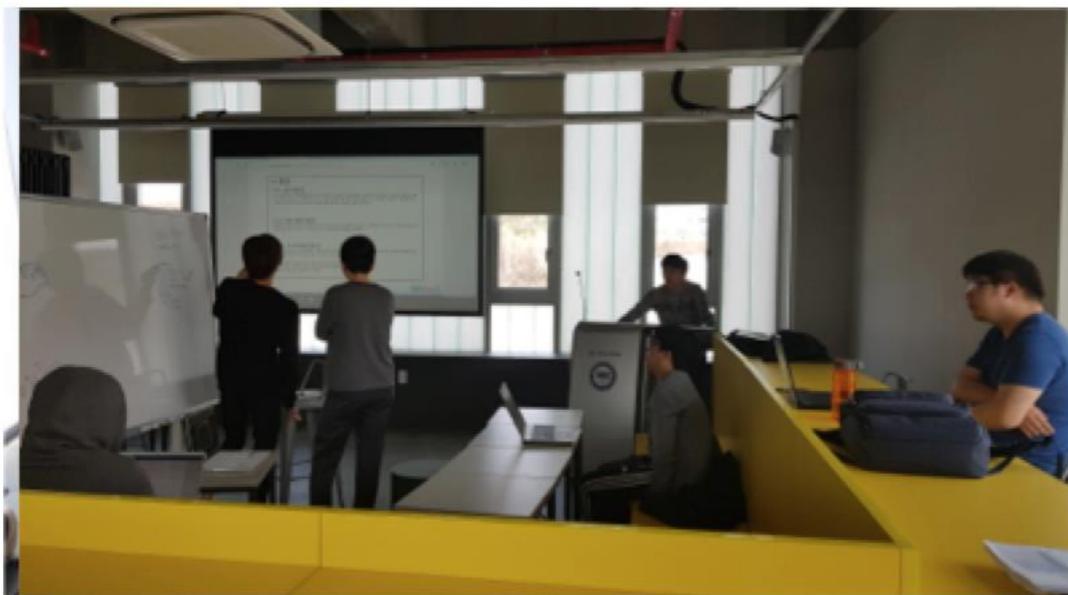
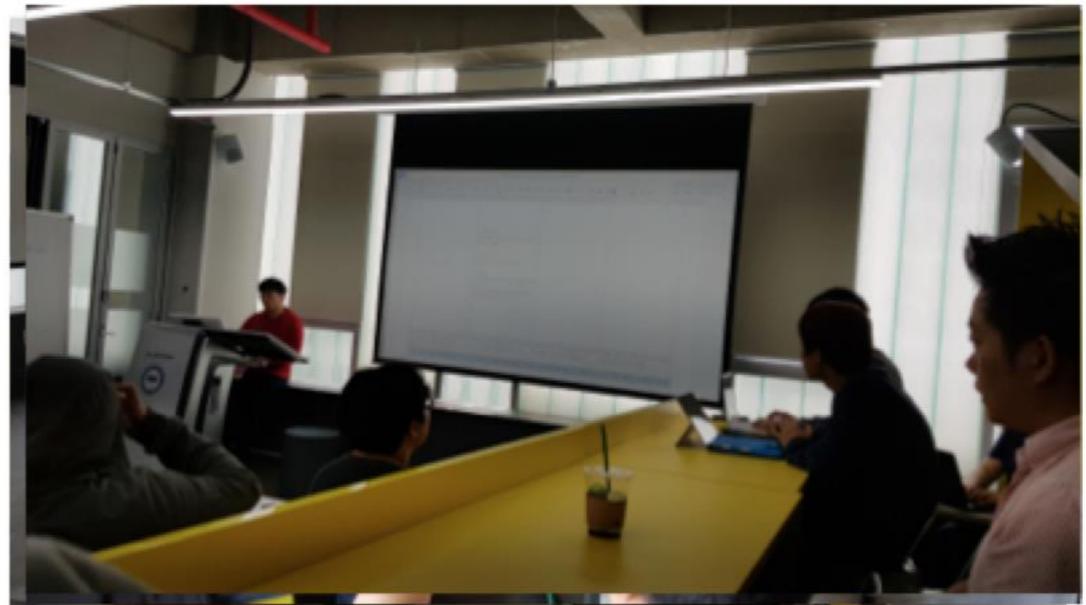
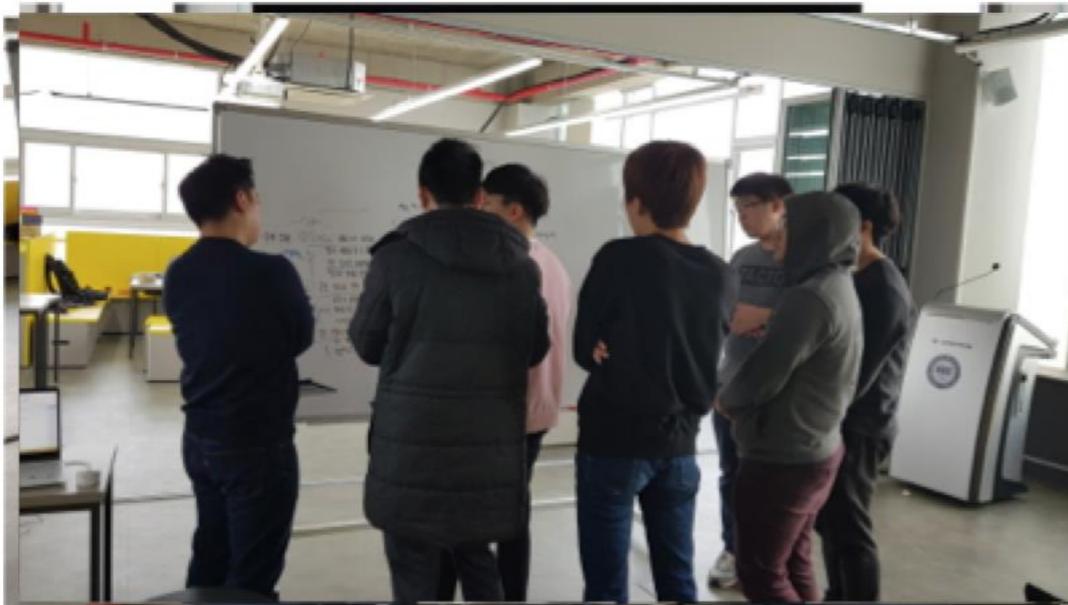
### ◦ 자료 수집 (개별)

<p><b>스크럼</b></p> <p>특징</p> <ul style="list-style-type: none"> <li>• 스프린트에 포함할 기능/개선점에 대한 우선 순위</li> <li>• 개발 : 개발 : 결과물</li> <li>• 개발 : 에 대해</li> <li>• 날마다</li> <li>• 항상 1</li> <li>• 원활한 는 열</li> </ul>	<p>진행</p> <p>스크럼에서는, 30일 (stakeholders)는 책임과 용기에 중점을 두는 경향이 크다.</p>	<p><b>크리스탈 페일리</b></p> <p>집중하는 것</p> <ul style="list-style-type: none"> <li>• People</li> <li>• Interaction</li> <li>• Community</li> <li>• Skills</li> <li>• Talents</li> <li>• Communication</li> </ul>	<p>를 받게 받을 수 있습니다.</p> <p><b>Personal safety</b></p> <p>이것은 한 무리의 사람들 사이에서 언론의 자유와 관련이 있다. 어떤 사람이 질문을 하고, 아이디어를 제안하고 할 때마다 조롱을 당한다면, 다음 번에 더 큰 소리로 말할 확률이 더 낮아질 것이다. 팀의 사람들은 서로를 믿을 수 있어야 하고, 문제나 어떤 일이 일어나더라도 거리낌 없이 의견을 말할 수 있어야 한다.</p>
<p><b>익스트림</b></p> <p>특징</p> <ul style="list-style-type: none"> <li>• 비즈니스 이 심</li> <li>• 10-12 (Pract 적은 ; 적용히</li> <li>• 개발 : 적인 ;</li> </ul>	<p><b>원 소프트웨어</b></p> <p>특징</p> <ul style="list-style-type: none"> <li>• 인간 조건</li> <li>• 지식 작업이 당연히</li> <li>• 더 나은 : 업합니다.</li> <li>• 효과적인</li> <li>• 광범위한 도출 및 /</li> <li>• 값을 기법 길이의 형</li> </ul>	<p>코드를 제거할 수 있습니다. 여기서 구성 관리와 자동화된 테스트가 시작됩니다. 이는 더 빠른 단위 시험과 궁극적인 문제 해결을 가능하게 한다.</p>	<p><b>기능 주도 개발 방법론</b></p> <p>정의</p> <p>기능 기반 개발은 반복적이고 점진적인 소프트웨어 개발 프로세스입니다. 이 방법은 소프트웨어를 개발하기 위한 경량의 신속한 방법입니다. FDD는 업계에서 인정되는 다양한 모범 사례를 응집력 있는 전체로 결합합니다. 이러한 사례는 클라이언트의 가치 있는 기능(기능)관점에서 이루어집니다. 소프트웨어의 주요 목적은 적시에 실질적이고 작동하는 소프트웨어를 반복적으로 전송하는 것입니다.</p>
		<p><b>기본 활동</b></p> <p><b>전체 모델 개발</b></p> <p>FDD 프로젝트는 시스템의 범위와 내용을 개괄적으로 설명하는 것으로 시작합니다. 다음으로, 각 모델링 영역에 대해 작업은 그룹별로 상세 도메인 모델이 생성되고, 안전 점검을 위해 제시된다. 제안된 모델 중 하나 이상이 각 도메인 영역의 모델로 선택되었습니다. 도메인 영역 모델이 점차 전체 모델로 병합됩니다.</p> <p><b>제조 기능 목록</b></p> <p>초기 모델링 동안 수집된 지식은 도메인을 주제 영역으로 기능적으로 분해하여 특정 목적을 식별하는 데 사용됩니다. 각 주제 영역에는 비즈니스 활동이 포함되어 있으며, 각 비즈니스 활동의 단계는 분류된 특정 목록에 대한 기초를 이룹니다. 이 점에서 특징은 고객 가치 함수의 작은 부분이며, "판매액 계산"은 총&lt;동작&gt;&lt;결과&gt;의 형태로 표현됩니다. 특징은 완성하는 데 2주 이상 걸리지 않아야 하고, 그렇지 않으면 작은 조각으로 나누어야 한다.</p> <p><b>특징별 계획</b></p> <p>형상 목록이 완성된 후에 다음 단계는 개발 계획을 작성하고 형상(또는 형상 집합)의 소유권을 프로그래머에게 클래스로 할당하는 것이다.</p> <p><b>특징별 설계</b></p> <p>각 형상에 대해 설계 패키지가 생성된다. 수석 프로그래머는 2주 안에 개발될 작은 특징 그룹을 선택한다. 수석 프로그래머는 해당 클래스 소유자와 함께 각 기능에 대한 상세 순서도를 작성하고 전체 모델을 재정의한다. 다음으로, 클래스와 방법에 대한 설명이 작성되고, 마지막으로 설계 검사가 실시됩니다.</p> <p><b>특징별 구축</b></p> <p>기능을 생성하기 위한 각 활동에 대한 성공적인 설계 검사가 계획된 후에, 클래스 소유자는 클래스에 대한 코드를 개발합니다. 유닛 테스트와 성공적인 코드 검사를 마치면 완료된 기능이 해당 빌드로 승격됩니다.</p>	

[다. 팀원들이 팀을 구성함으로써, 다른 팀원들이 무엇을 하고 있는지를 파악할 수 있으므로, 필요한 경우 팀 총회의 업무]

- Mini-PBL 1

- 협력 학습 (팀별)



- Mini-PBL 1

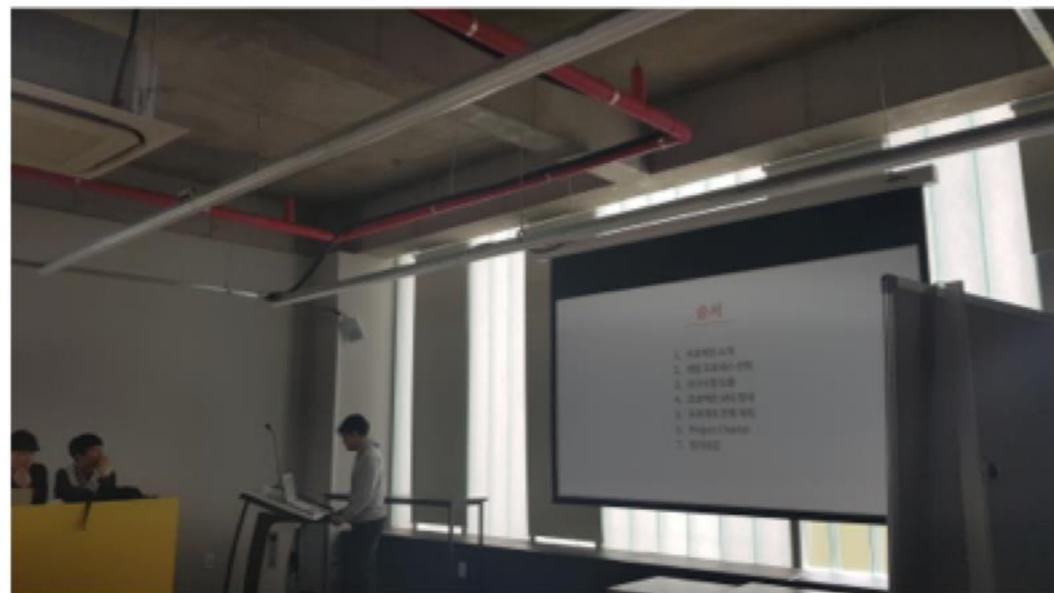
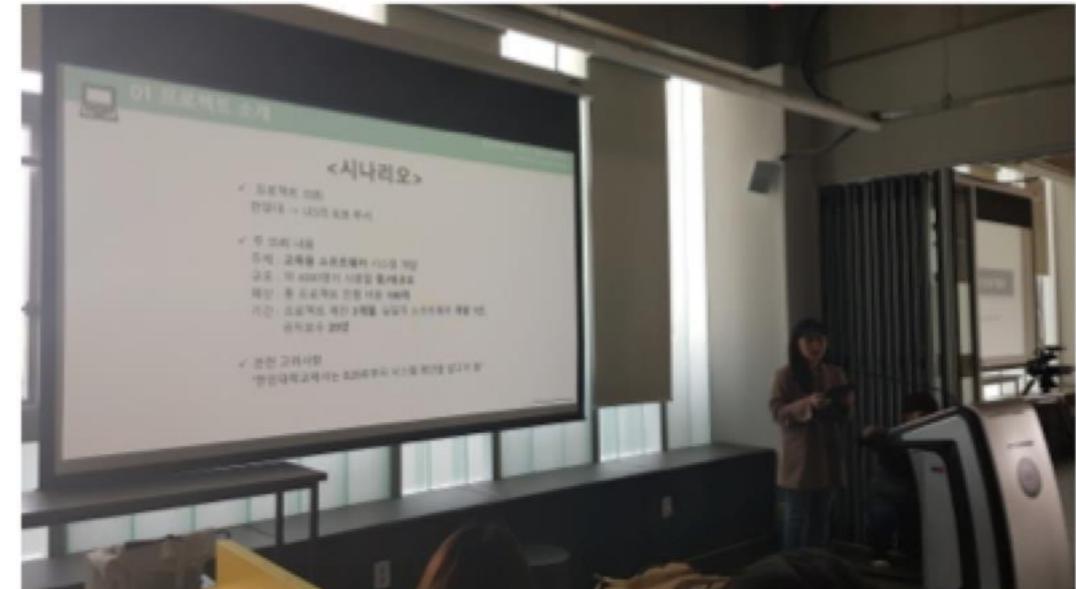
- 협력 학습 (팀별)

팀 활동 보고서

미팅 개요	일시	2018년 3월 26일 2018년 3월 28일	문제해결과정	문제 해결과정 : 사실-생각-학습과제-실천계획 해결방안 선택 각각 자신만의 파트를 정해서 정보를 취득 한 후 팀원과의 의사소통 후 의견종합(토의)	
	참석자	SDI(한수인, 김주호, 안민수, 정재훈, 황정준)			
	본 미팅의 주요활동	우리의 프로젝트에 맞는 효과적인 애자일 프로세스를 선정 프로젝트에 맞게 애자일 프로세스를 적용시키는 방법을 찾아 구현 우리의 문제 분석/파악/도출 과정을 UCC 영상으로 제작			
진행 사항	구분	활동 내용	조치사항		
	이번 미팅에서 한 일	2개의 애자일 프로세스 중 하나를 선택(스크럼 VS XP)	애자일 프로세스들 특징 비교 분석 가장 적합하다고 생각되는 스크럼 VS XP 중 선택 -> 스크럼을 선택 : 백로그를 통한 효율적인 일정관리가 가능하다.		
		스크럼을 프로젝트(온라인 강의 플랫폼) 상황에 맞춰 적용	1. 기존의 백로그 방식이 아닌 대분류 -> 중분류 -> 소분류로 과정으로 백로그 설정 및 스프린트 전개 2. 일일회의를 주 3회 회의로 변경 3. 역할분담		
		UCC 제작	추가로 하루 일정(18.3.28)을 잡아 다 같이 영상제작		
	논의사항	백로그 선정을 어떻게 하는 것이 효율적인가? 다른 프로세스에 가져 올만한 요소가 있는가?			
추후 계획	구분	활동 내용	역할분담	추진일정	
	다음 미팅에서 해야 할 일	스프린트 백로그 구체화 구체적인 스프린트 설정 일일 스크럼 회의 stakeholder와의 회의를 통한 요구사항 도출	PO: 한수인 SM: 정재훈 팀: 김주호, 안민수, 황정준	4. 2 : 백로그 구체화 4. 9 : 요구사항 도출 스프린트 설정의 경우 만날 때 마다 조금씩 구체화 시킨다.	
	기타				
피드백	교수님의 피드백 사항				

- Mini-PBL 1

- 문제 해결안 발표



- Mini-PBL 1
  - 문제 해결안 평가



팀간평가 + 팀원평가

- Mini-PBL 1

- 문제 해결안 평가

### PBL 팀원 평가지

Mini-PBL : 숙련적인 소프트웨어 개발 프로세스 설계 및 적용 방안 도출 / 요구공학

### PBL 팀간 평가지

Mini-PBL : 효과적인 요구사항 도출, 협상, 명세, 검출 방법 선정 및 적용 방안 도출

팀 이름	SD1				
팀명	BTS	Echo	SD1	취보	스까게티
프로젝트 주제 선정	5	5		5	5
프로젝트 문제 파악	4	3		4	3
문제에 대한 해결방안	3	3		4	3
소프트웨어 개발 프로세스 적합성	3	3		4	4
정보전달	4	4		3	4
주제 현실 적용 가능성	5	4		4	3
Q & A 대처	3	3		3	2
점수 합계	28	25		27	24

팀	안민수
한수인	<u>황정준</u>
5	5
5	4
4	5
5	5
5	4
5	5
5	5
4	4
38	37

## • Mini-PBL 1

### ◦ 성찰



### 성찰 일지

학번	2013043177	이름	김주호
1. 이번 학습을 통해 무엇을 배웠나요? 이번 학습을 통해 요구사항을 체계적으로 도출하는 요구사항 분석을 모두 진행함에 따라 결과물로 나올수 있는 명세서(Specification)에 대해서 학습하였습니다.	5. 이 수업에서 나의 부족한 부분은 무엇인가요? 이 수업을 배우면서 느낀 부족한 점은 제가 실제와 비견할만한 프로젝트를 경험해보지 못했다는 점입니다. 따라서 요구공학의 단계 단계에서 이해관계자와 개발팀의 미팅 빈도를 맞추는 과정 그리고 마지막 단계인 검증과정에서 요구사항을 검토하는 과정을 배우는 과정에서 실감이 나지 않았던 것 같습니다.		
2. 수업에서 어려웠던 활동은 무엇이었나요? 이번 프로젝트는 개발팀에서 먼저 아이디어 제안기간을 통해 제안하는 과정이 포함되어 있기 때문에 그저 요구사항만 나열하고 미리 아이디어를 적절히 구체화 해가는 과정이 포함되었습니다.	6. 이 수업의 학습과정을 통해 무엇을 느꼈나요? 이번 수업 학습과정을 통해서 이해관계자가 요구하는 요구사항들은 간단하게 얻어낼 수 없다는 사실에 대해서 깨달았습니다. 따라서 요구공학이 탄생하게 되었다는 사실에 대해서 알게 되었습니다. 다만 현실에 맞춰 애자일 프로세스를 사용함에 있어 요구공학이 설명하는 전부를 사용하는 것이 아니라 필요한 만큼 적절히 가져다 써야함을 느꼈습니다.		
3. 앞으로 내가 더 알고 싶은 내용은 무엇인가요? SRS는 IEEE 표준이기는 하지만, 이는 요구공학이 요구사항을 도출하는 방법과 다르다고 한다면 이것은 애자일 개발 프로세스와는 맞지 않게 사용중 필요에 의해 SRS를 작성한다면 어느 항목에 대해 더 알아보고 싶습니다.	7. 기타 느낀 점을 자유롭게 기술하세요. 아무리 폭포수 모델을 쓰던 애자일 개발 프로세스인 스크럼을 쓰던 가장 중요한 것은 고객이 요구하는 결과물을 인도하는 것입니다. 따라서 요구사항을 도출하는 과정은 정말 중요하다고 할 수 있습니다. 그야말로 요구사항을 도출하는 시작이 정말 프로젝트의 절반만큼이나 중요하다고 할 수 있을 것입니다. 이를 위해 요구공학이 탄생하였지만 시대에 따라 애자일 프로세스가 탄생하고 요구공학이 취사선택이 된다는 것을 보면 영원한 진리는 없는 것 같습니다.		
4. 학습한 내용을 적용할 수 있는 것은 무엇인가요? 이번에 학습한 내용을 토대로 실제 프로젝트에서 이해관계자와 소통하는 방법 사용하는 것뿐만 아니라 실생활의 이모저모에 이 무엇인지를 알아내는 핵심에 대해서 깨달았습니다.			

- 대상 : 연구실 대학원 졸업생



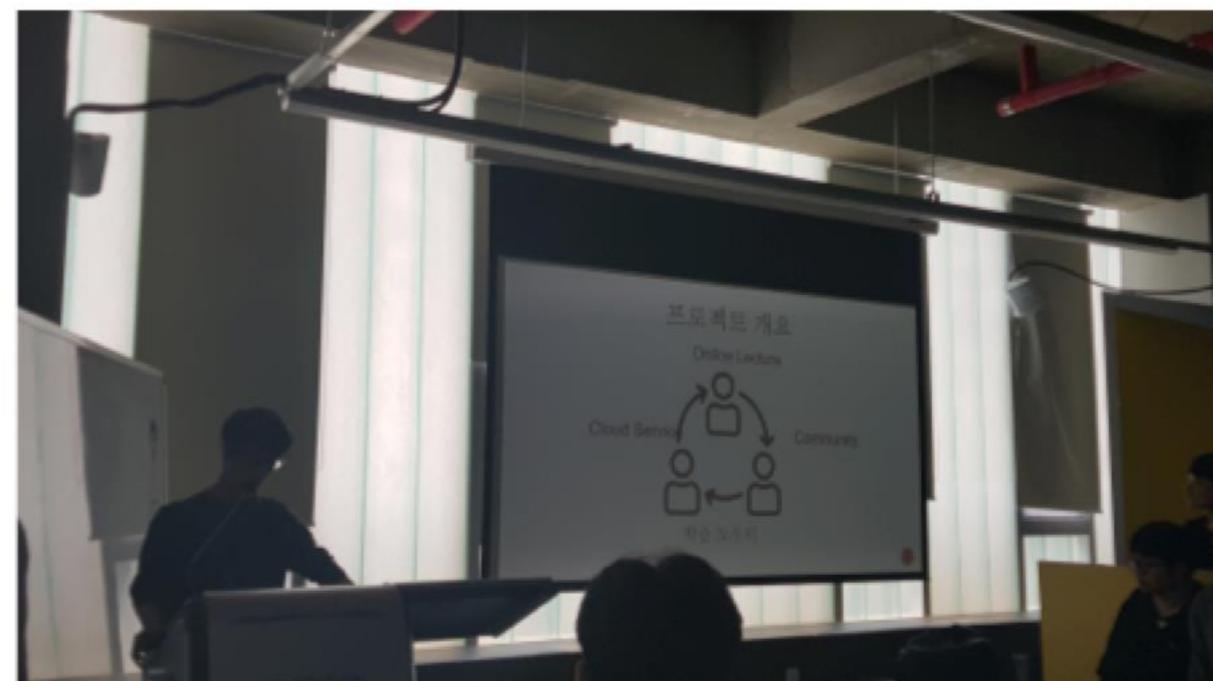
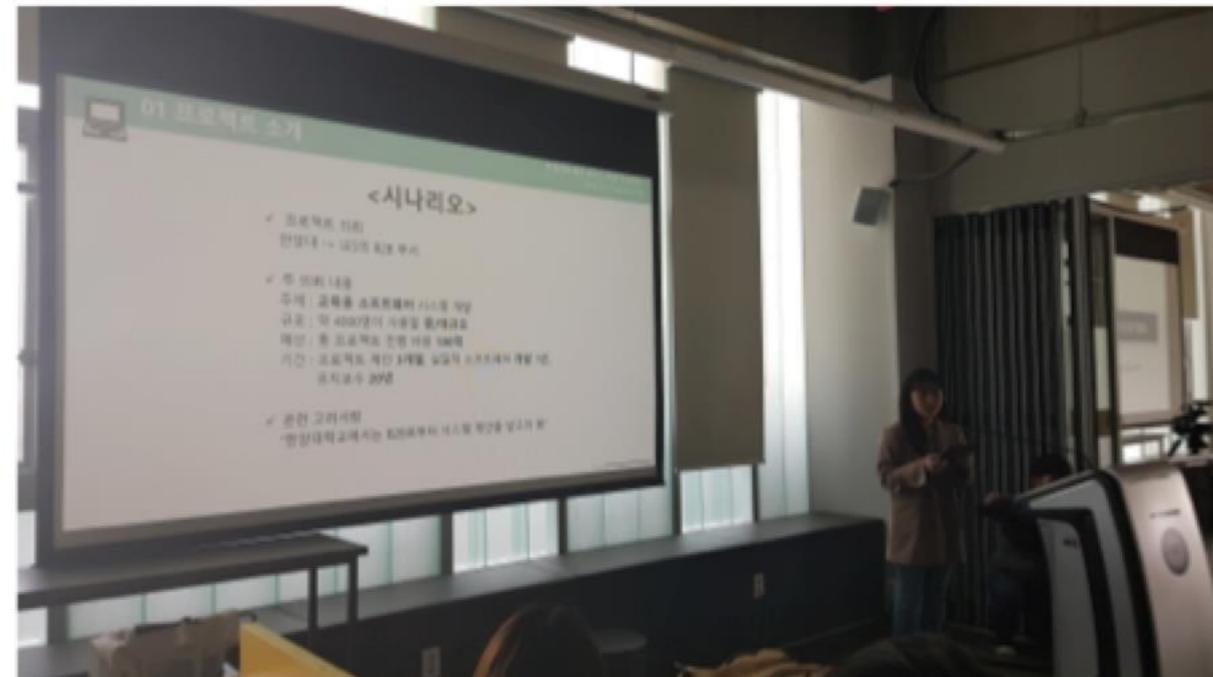
- 역할 :

- 문제 (시나리오) 개발 시 협업
- 특강
- 프로젝트 / Mini-PBL 결과물 검토, 평가, Feedback



- 학습참여도 (20%)
  - PBL 수업/활동 중 활발한 의견 표출, 지식 공유, 발표, 토론, Q&A, ...
- 과제 (30%)
  - PBL 활동 : 개별 산출물
  - 실습 과제
- 프로젝트 (50%) : 팀간 + 팀내 평가 포함
  - PBL 활동 : 팀별 산출물
  - 프로젝트 산출물 : SRS, 프로젝트 관리 문서, 설계 문서, QA 문서, ...
  - 프로젝트 중간 및 최종 발표
  - 프로젝트 관련 토론

## ◦ 중간 및 최종 발표



- 문제 해결안 평가



- 교수 + 산업체 멘토 평가 및 모니터링



수업을 진행하면서 IC-PBL이라는 수업방식 자체에 익숙해져 나가는 것도 좋았지만 무엇보다 문제에 대한 사고방식 자체가 변화하는 것 같이 느껴져서 좋았습니다. 일방적으로 과제가 주어지고 그 과제를 해결하는데 열중하다보면 가끔은 인터넷에서 그 적당한 해결책을 베끼는데 그치게 되는 경우가 생기기도 합니다. 하지만 PBL 수업에서는 문제를 해결하려면 인터넷을 검색하는 것보다 팀원들과 소통하고 치열하게 토론하는 과정이 더 중요합니다. 나와 팀원들의 의견들이 유기적으로 결합되고, 전혀 다른 결과물로 새롭게 변화되는 과정을 보는 것도 즐거웠습니다. 또한, 사실-가정-문제도출의 과정을 반복해 해결방안을 얻어내는 IC-PBL 고유의 적용과정을 사용하는 것이 처음에는 번거롭고 행정적으로 느껴졌지만 실제로 문제에 적용해보니 쉽고 논리적으로 해결방안을 얻어낼 수 있어서 좋았습니다.

IC-PBL 수업의 가장 큰 장점은 합리적인 주제의 프로젝트와 문제가 시나리오로서 제시된다는 점입니다. 이 시나리오는 주제에 몰입하는데 도움을 주거나 주제를 따라가는데 도움을 주는 가이드라인이 됩니다. 시나리오에 깊게 몰입해 팀원들과 더욱 더 많은 것들을 나누고 유연하게 토론하다보니 토론식 수업이 주는 학업적인 부담을 보상하고도 남을 만큼 많은 것을 얻을 수 있었습니다.



자기주도성



팀워크



의사소통 능력



발표 능력



문제 해결력



창의적, 비판적, 전략적 사고력 함양





HANYANG  
UNIVERSITY  
ERICA

**HYU** | HANYANG UNIVERSITY ERICA  
**COMPUTING**