멘토링 이후 변경 사항 및 설문 분석

졸업 프로젝트 주제: SW 자율학습 강의 Docker+웹 서비스 개발

종합설계1_02분반
야간자율학습
고태완
김지혜
최수연

야간자율학습

멘토링 그 이후

- 현재 메인 시스템이 3개로 규모가 너무 큼 (웹기반 동영상강의플랫폼, 퀴즈-채점, 도커를 이용한 실습 환경)
- 도커를 이용한 실습 환경에 초점을 맞추기로 결정(선택과 집중)
- 이미 시장에서 잘 사용되고 있는 이론 강의에 대한 서비스 및 퀴즈는 최종 프로젝트에서 삭제
- 실습 환경을 이미지로 저장하여 수강생들 개개인에게 하나의 컨테이너로 제공

  => OS에 대한 접근이 필요한 실습에서 수강생은 root 권한으로 자유롭게 실습 진행 가능, 단 1회용 컨테이너, 실습이 끝난 후에 삭제됨

  => Shell과 Code Editor를 웹페이지에서 제공.
  - 단 Shell을 사용하지 않아도 되는 경우(OS에 접근하지 않아도 되는 경우) Shell을 제공하지 않음
  - Code Editor를 사용할 경우에 채점을 하지 않고 컴파일 된 결과만 보여줌.

  => 기본적으로 사용자가 업로드한 실습 강의에 대해서 저장된 실습의 난이도를 이용해 사용자에게 수준별 학습을 제공할 수 있도록 Curation한다.

  => 특정 강사가 특정 사용자들에 대해서만 실습을 가능하게 할 것을 원하는 경우, 이 실습은 private 상태가 되며 특정 강사가 저장한 환경으로 사용 권한을 가진 수강생 들에 한해서 해당 환경으로 바로 실습을 진행 할 수 있도록 한다.
- Docker와 Kubernetes를 이용한 Orchestration 방식의 유연한 서버 운영
- CI(Continuous Integration) 기반의 소스 코드 에디터 사용. 소스코드를 올리면 빌드 후 에러 포함한 결과 리포트를 사용자에게 보여줌.

위와 같은 이유로 웹기반 동영상강의플랫폼, 퀴즈-채점은 제거

도커를 이용한 실습 환경 중점적으로 구현 => SE문서에 좀 더 상세하게 기록

멘토링 그 이후

- 현재 메인 시스템이 3개로 규모가 너무 큼 (웹기반 동영상강의플랫폼, 퀴즈-채점, 도커를 이용한 실습 환경)
- 도커를 이용한 실습 환경에 초점을 맞추기로 결정(선택과 집중)
- 이미 시장에서 잘 사용되고 있는 이론 강의에 대한 서비스 및 퀴즈는 최종 프로젝트에서 삭제
- 실습 환경을 이미지로 저장하여 수강생들 개개인에게 하나의 컨테이너로 제공

  => OS에 대한 접근이 필요한 실습에서 수강생은 root 권한으로 자유롭게 실습 진행 가능, 단 1회용 컨테이너, 실습이 끝난 후에 삭제됨

  => Shell과 Code Editor를 웹페이지에서 제공.
  - 단 Shell을 사용하지 않아도 되는 경우(OS에 접근하지 않아도 되는 경우) Shell을 제공하지 않음
  - Code Editor를 사용할 경우에 채점을 하지 않고 컴파일 된 결과만 보여줌.

  => 기본적으로 사용자가 업로드한 실습 강의에 대해서 저장된 실습의 난이도를 이용해 사용자에게 수준별 학습을 제공할 수 있도록 Curation한다.

  => 특정 강사가 특정 사용자들에 대해서만 실습을 가능하게 할 것을 원하는 경우, 이 실습은 private 상태가 되며 특정 강사가 저장한 환경으로 사용 권한을 가진 수강생 들에 한해서 해당 환경으로 바로 실습을 진행 할 수 있도록 한다.
- Docker와 Kubernetes를 이용한 Orchestration 방식의 유연한 서버 운영
- CI(Continuous Integration) 기반의 소스 코드 에디터 사용. 소스코드를 올리면 빌드 후 에러 포함한 결과 리포트를 사용자에게 보여줌.

위와 같은 이유로 웹기반 동영상강의플랫폼, 퀴즈-채점은 제거

도커를 이용한 실습 환경 중점적으로 구현 => SE문서에 좀 더 상세하게 기록

→ 문서 재작성 및 프로토타입 재제작

문서

1. 문제정의서
2. 요구사항명세서
3. 유스케이스
4. 클래스다이어그램
5. 시퀀스다이어그램

문서

1. 문제정의서
2. 요구사항명세서
3. 유스케이스
4. 클래스다이어그램
5. 시퀀스다이어그램

그림 1. 유저 식별 서브시스템에 대한 유스케이스 다이어그램

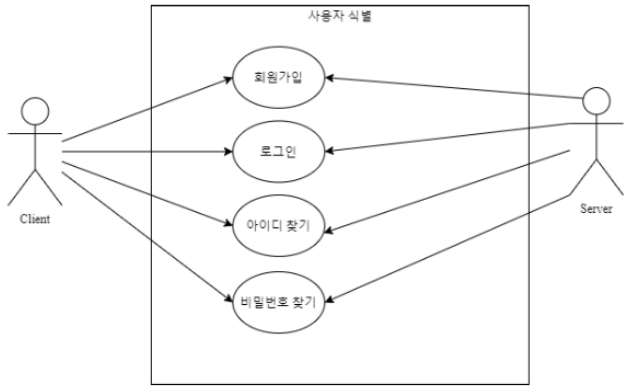


그림 2. 강의 수강 서브시스템에 대한 유스케이스 다이어그램

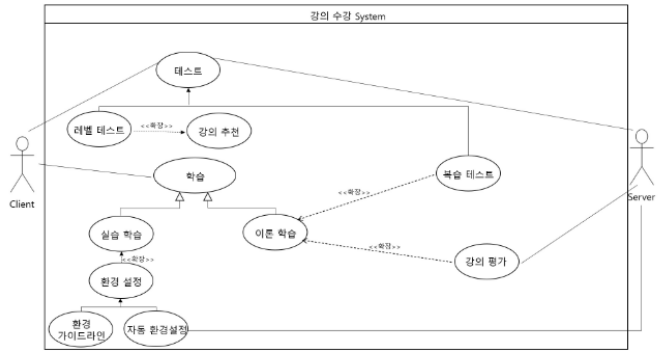


그림 3. 강좌 업로드 기능을 담당하는 서브시스템에 대한 유스케이스 다이어그램

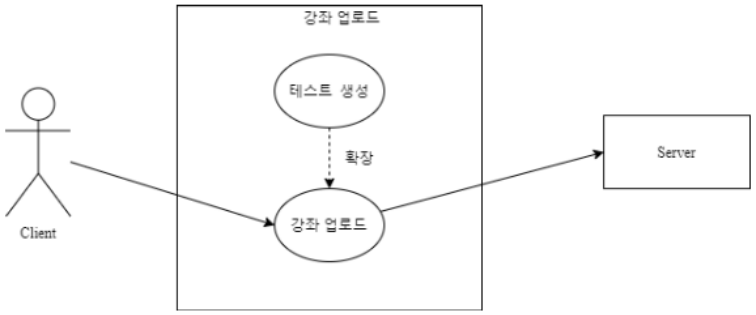
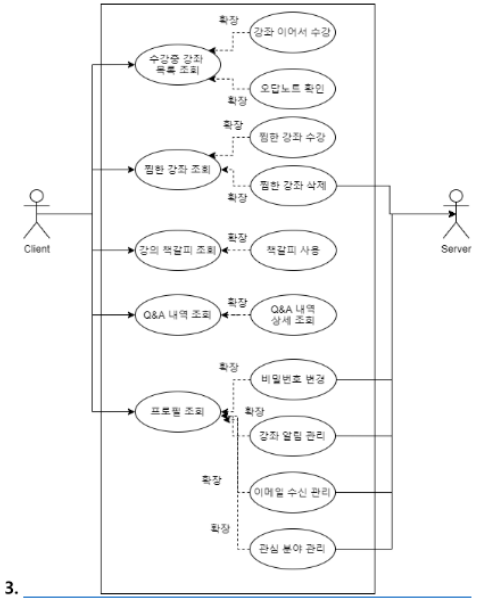


그림 4. 마이페이지 관리 기능을 담당하는 서브시스템에 대한 유스케이스 다이어그램



3.

문서

1. 문제정의서
2. 요구사항명세서
3. 유스케이스
4. 클래스다이어그램
5. 시퀀스다이어그램

그림 1. 유저 식별 서브시스템에 대한 유스케이스 다이어그램

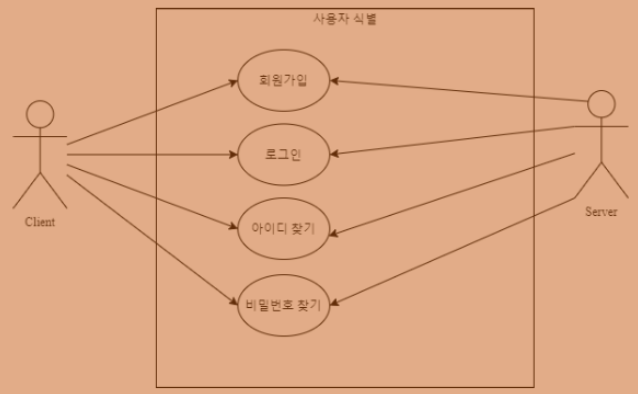
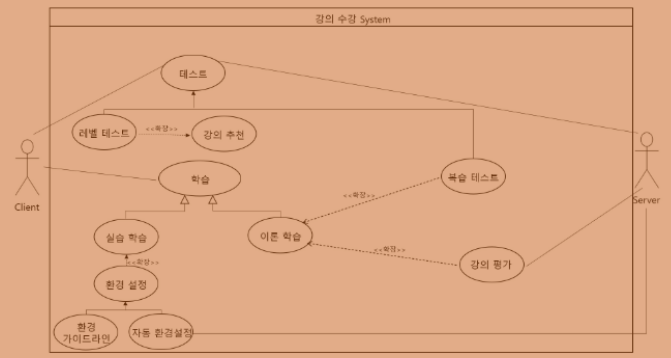


그림 2. 강의 수강 서브시스템에 대한 유스케이스 다이어그램



간소화

그림 3. 강좌 업로드 기능을 담당하는 서브시스템에 대한 유스케이스 다이어그램

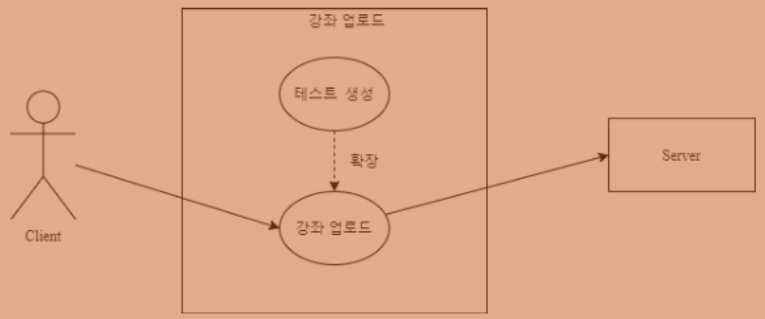
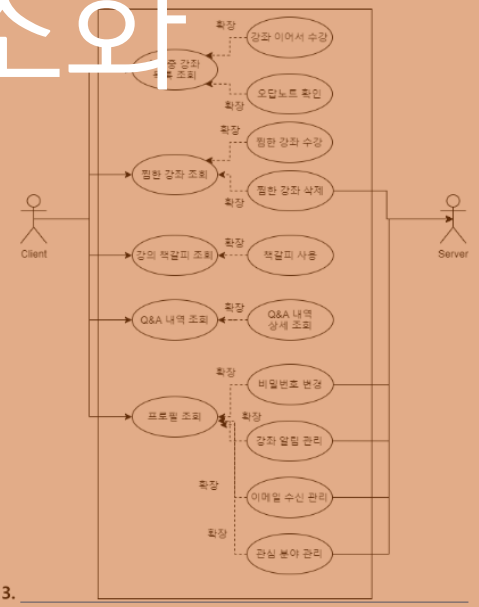


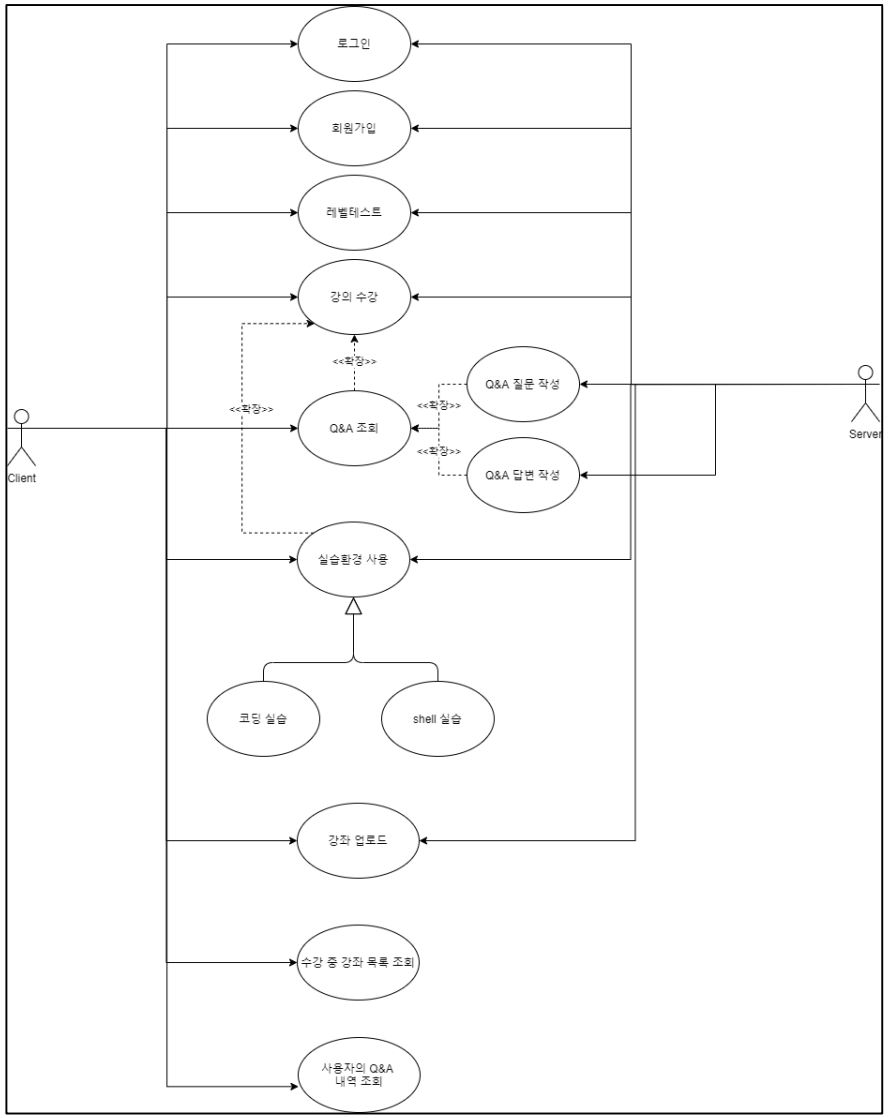
그림 4. 마이페이지 관리 기능을 담당하는 서브시스템에 대한 유스케이스 다이어그램



3.

문서

1. 문제정의서
2. 요구사항명세서
3. 유스케이스
4. 클래스다이어그램
5. 시퀀스다이어그램



하나의 시스템
Sub System X

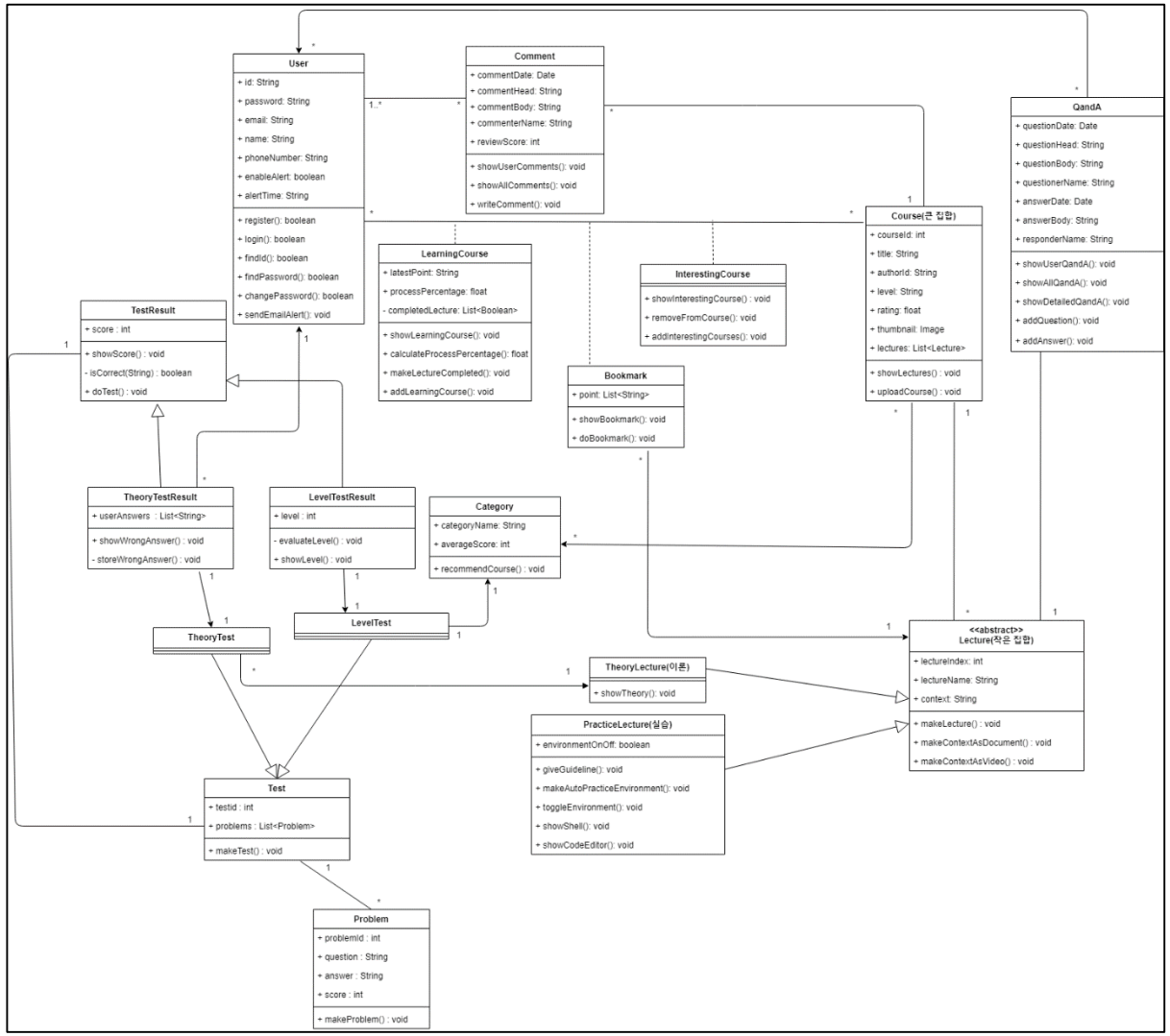
문서

1. 문제정의서
2. 요구사항명세서
3. 유스케이스
4. 클래스다이어그램
5. 시퀀스다이어그램

28개 -> 12개의 Specification, 핵심(실습 환경 구축)에 집중

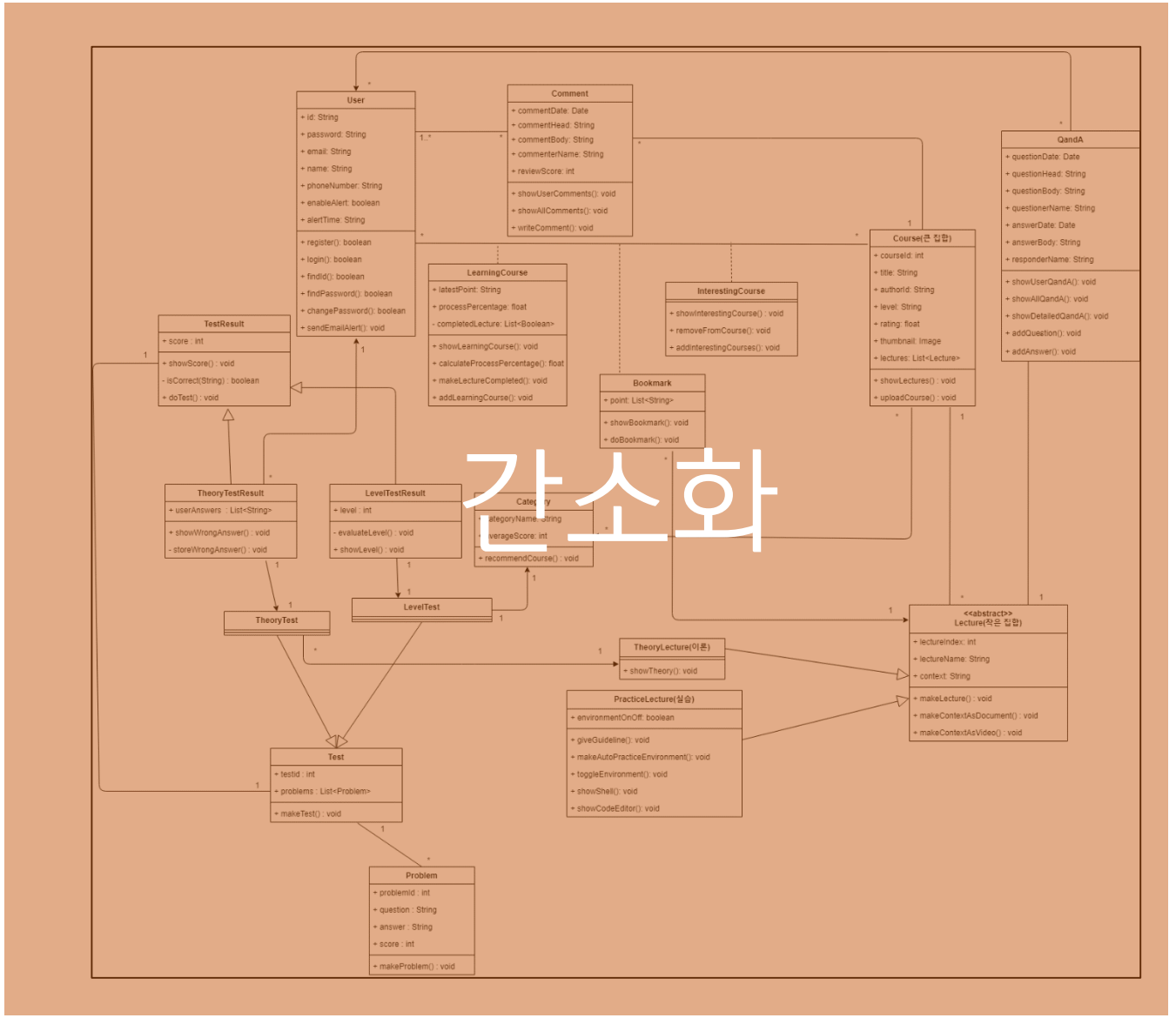
문서

1. 문제정의서
2. 요구사항명세서
3. 유스케이스
4. 클래스다이어그램
5. 시퀀스다이어그램



문서

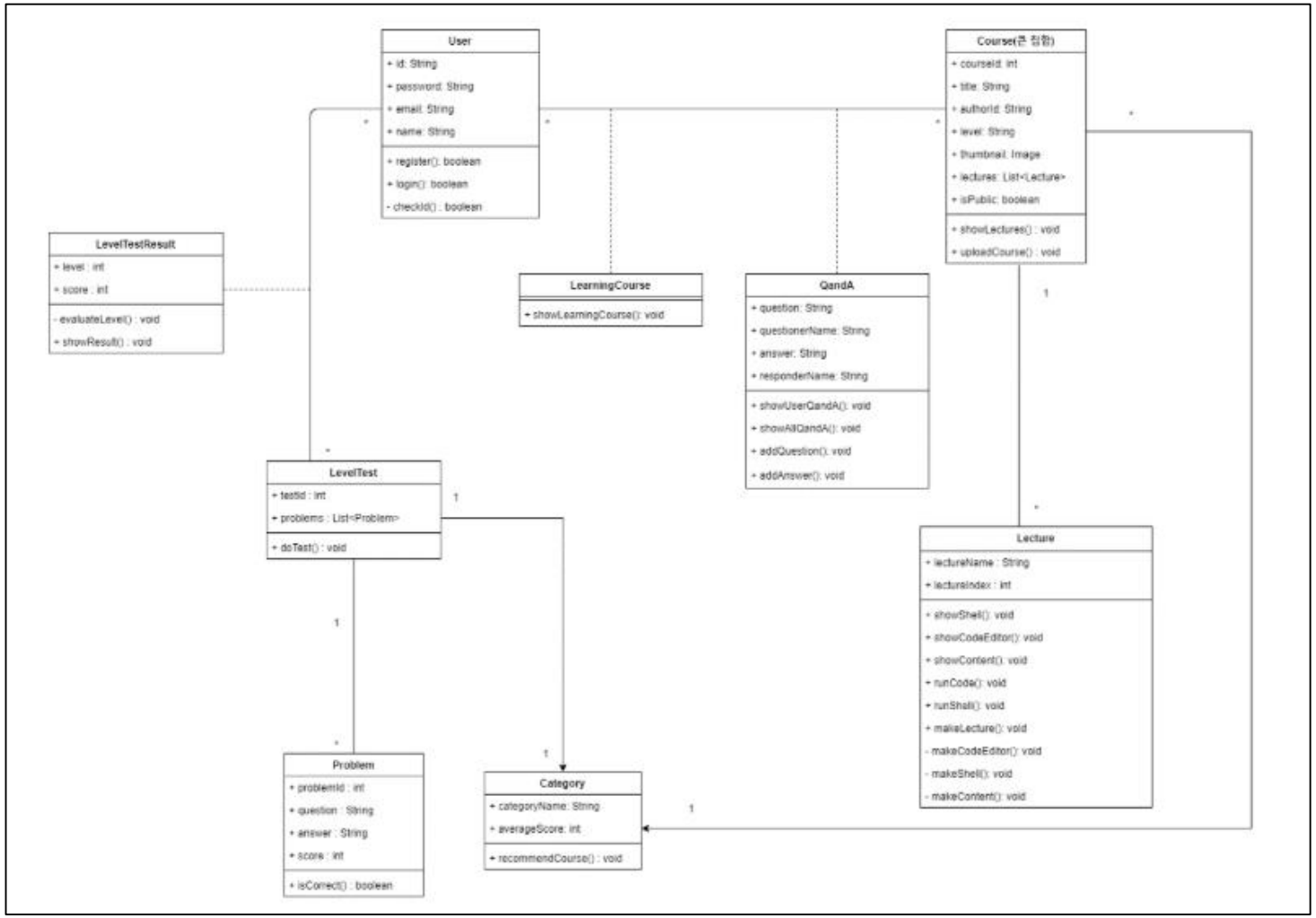
1. 문제정의서
2. 요구사항명세서
3. 유스케이스
4. 클래스다이어그램
5. 시퀀스다이어그램



문서

1. 문제정의서
2. 요구사항명세서
3. 유스케이스
4. 클래스다이어그램
5. 시퀀스다이어그램

18개 -> 9개의 Specification, 핵심(실습 환경 구축)에 집중



문서

1. 문제정의서
2. 요구사항명세서
3. 유스케이스
4. 클래스다이어그램
5. 시퀀스다이어그램

Table of Contents

목차

문서

1. 문제정의서
2. 요구사항명세서
3. 유스케이스
4. 클래스다이어그램
5. 시퀀스다이어그램

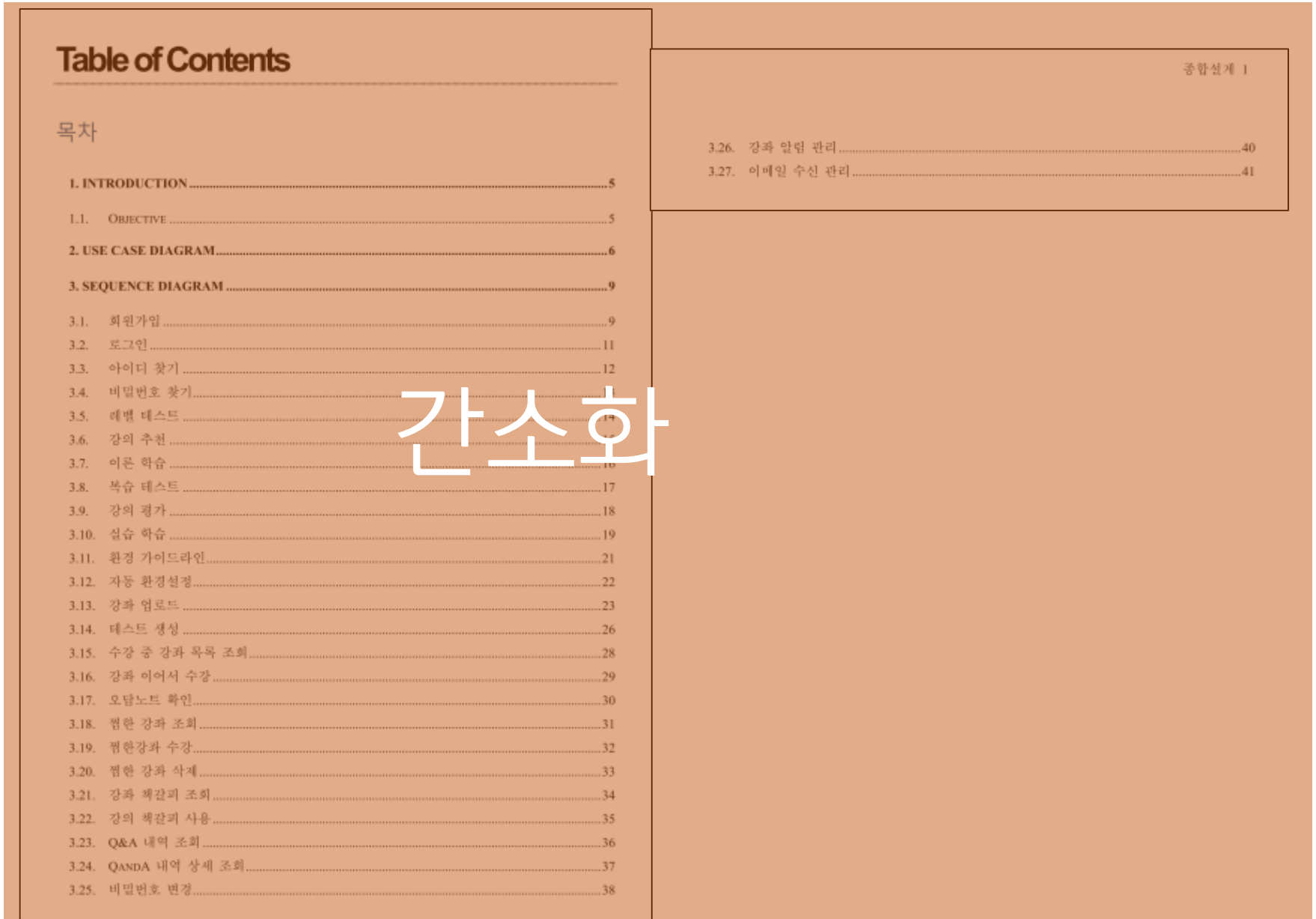


Table of Contents

목차

종합설계 1

간소화

문서

1. 문제정의서
2. 요구사항명세서
3. 유스케이스
4. 클래스다이어그램
5. 시퀀스다이어그램

프로토타입 V2

프로토타입 Version 2.

Level Test기능을 남긴 채 실습 환경에 관한 기능 위주로 구현
Code Editor, Shell, Result Windows 로 실습 환경 지원

강좌를 업로드할 때 실습 환경을 저장하여 수강생들에게 링크로 배포할 수 있음
(Public, Private 등 공개 여부 설정 가능)

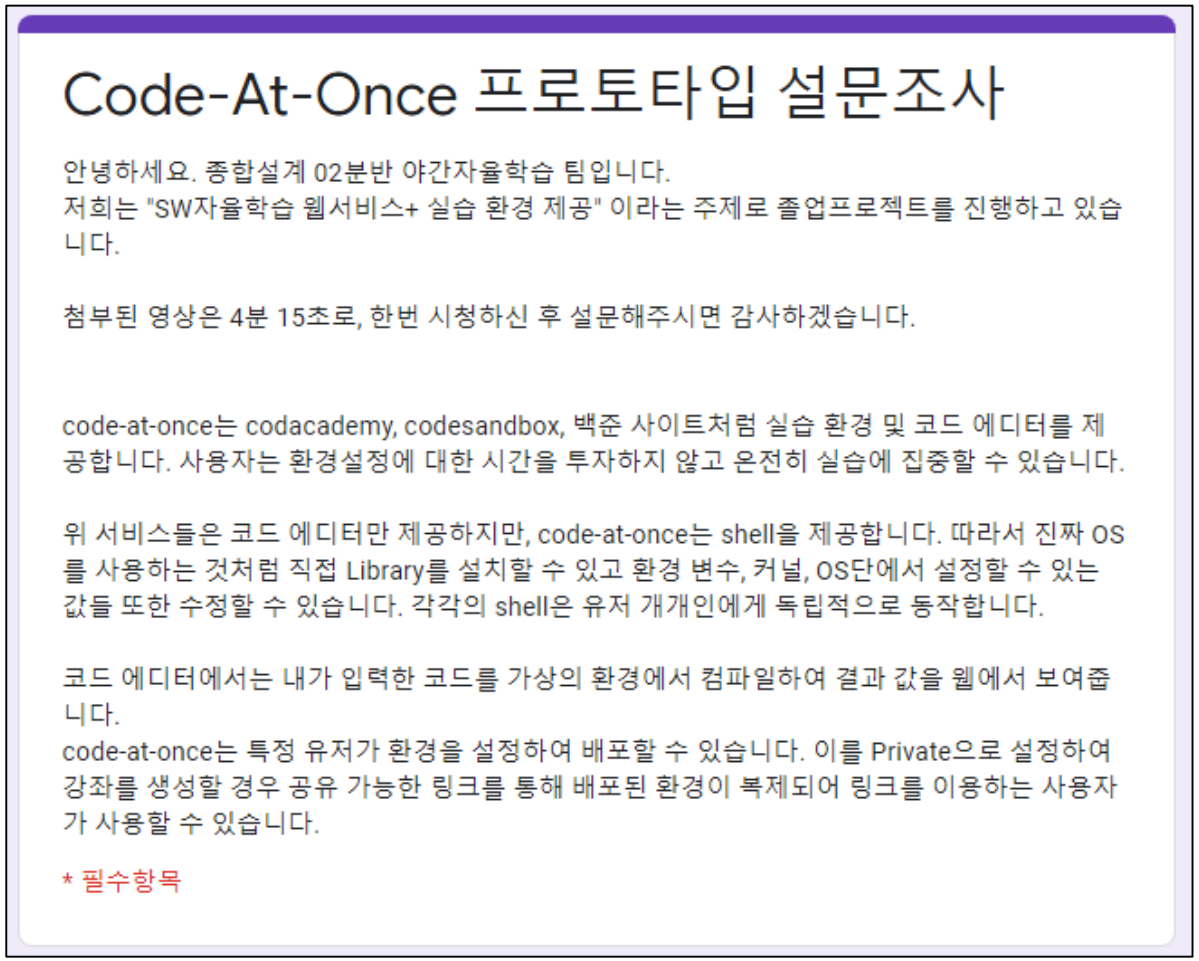
그 외 자잘한 편의 기능 삭제 => 추후 구현 가능

프로토타입 V2

Demo: https://youtu.be/rGqttuH5E8s

설문조사

SW와 관련이 있는 지인들에게 요청, 6명이 응답



Code-At-Once 프로토타입 설문조사

안녕하세요. 종합설계 02분반 야간자율학습 팀입니다.
저희는 "SW자율학습 웹서비스+ 실습 환경 제공" 이라는 주제로 졸업프로젝트를 진행하고 있습니다.

첨부된 영상은 4분 15초로, 한번 시청하신 후 설문해주시면 감사하겠습니다.

code-at-once는 codacademy, codesandbox, 백준 사이트처럼 실습 환경 및 코드 에디터를 제공합니다. 사용자는 환경설정에 대한 시간을 투자하지 않고 온전히 실습에 집중할 수 있습니다.

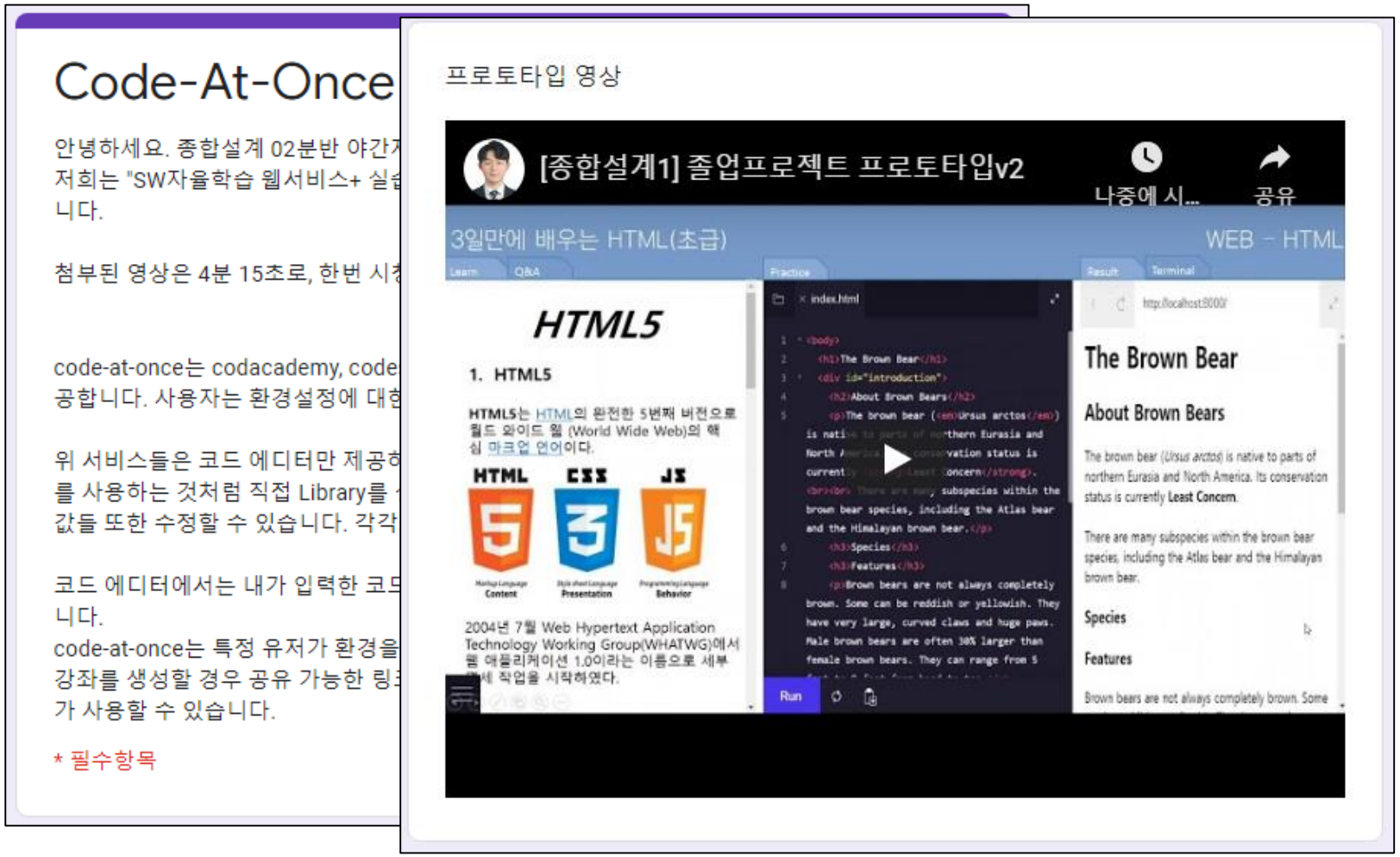
위 서비스들은 코드 에디터만 제공하지만, code-at-once는 shell을 제공합니다. 따라서 진짜 OS를 사용하는 것처럼 직접 Library를 설치할 수 있고 환경 변수, 커널, OS단에서 설정할 수 있는 값들 또한 수정할 수 있습니다. 각각의 shell은 유저 개개인에게 독립적으로 동작합니다.

코드 에디터에서는 내가 입력한 코드를 가상의 환경에서 컴파일하여 결과 값을 웹에서 보여줍니다.
code-at-once는 특정 유저가 환경을 설정하여 배포할 수 있습니다. 이를 Private으로 설정하여 강좌를 생성할 경우 공유 가능한 링크를 통해 배포된 환경이 복제되어 링크를 이용하는 사용자가 사용할 수 있습니다.

* 필수항목

설문조사

SW와 관련이 있는 지인들에게 요청, 6명이 응답



설문조사

1. 만약 환경설정의 번거로움을 해소하여 바로 코딩할 수 있는 서비스가 있다면 사용하시겠습니까?
2. codecademy, codesandbox.io, 백준과 같은 SW 학습 서비스를 사용해본 적이 있습니까?
   2-1. 위와 같은 SW 학습 서비스를 사용하면서 불편한 점이 있다면 적어주세요.
3. 영상에서 본 프로토타입 중 학습 과정에 대해서 추가되면 좋겠다, 혹은 없어도 되겠다라고 생각하는 기능이 있습니까?
4. 본 서비스가 출시된다면 사용할 의향이 있습니까?

설문조사

설문 결과

결론

사용자들은 기존의 다양한 기능을 제공할 때 보다 부족한 점을 느끼지 못하였고, 좀 더 전문화된 실습 환경 서비스 구축으로 프로젝트가 전환이 되었다.

야간자율학습