

2020-2학기

캡스톤디자인종합프로젝트1

프로젝트 최종보고서

팀번호 (팀명)	20_가_5조 (Study With Me)
프로젝트명	Study With Me
프로젝트 수행기간	2020. 9. 2 ~ 2020. 12. 16
팀장 및 팀원	<u>김찬민</u> , 김병준, 이주환, 장유수
프로젝트 지도교수	이수원 교수
작성일	2020. 12. 15.

깃허브주소:

https://github.com/racksals96/study_with_me_frontend

〈목 차〉

1. 프로젝트 개요 -----	3
1.1. 프로젝트 배경 -----	3
1.2. 프로젝트 최종 목표 -----	3
1.3. 중심어 -----	3
2. 관련 기술 현황 -----	4
2.1. 국내 시장 현황 -----	4
2.2. 국내 경쟁 제품 현황 -----	5
2.3. 본 프로젝트의 차별성 -----	6
2.4. 관련 기술 -----	7
3. 프로젝트 설계 -----	8
3.1. 주요 기능 -----	8
3.2. 시스템 구조도 -----	9 ~ 15
3.3. 개발 모듈과 컴포넌트 목록 -----	16
3.4. 개발 모듈과 컴포넌트 상세 내용 -----	17 ~ 22
4. 프로젝트 수행 결과 -----	23
4.1. 개발 환경 -----	23
4.2. 활용 Framework, API, Library -----	24
4.3. 결과 화면 예시 -----	25 ~ 36
4.4. 평가 -----	37, 38
5. 기대 효과 및 활용 방안 -----	38
5.1. 기대 효과 -----	38, 39
5.2. 활용 방안 -----	39
6. 프로젝트 수행 체계 -----	39
6.1. 팀원 인적 사항 -----	39
6.2. 프로젝트 역할 분담 -----	40
6.3. 프로젝트 수행 일정 -----	40
6.4. 프로젝트 수행 방법 -----	41
6.5. 프로젝트 자체 평가 -----	41
별첨 A. 회의록	
별첨 B. Source code	

1. 프로젝트 개요

1.1. 프로젝트 배경

최근 코로나 사태 이후 사회적 거리 두기 강화로 학교 열람실, 도서관, 카페 등에서 공부하기가 어려워지면서 캠퍼스터디를 활용한 '캠공족(웹 캠 켜놓고 공부하는 사람)'이 더 늘고 있다. 캠퍼스터디 유형으로는 모의고사 캠퍼스터디, 공휴일 동안 진행하는 캠퍼스터디 등이 있으며 '학원형' 캠퍼스터디도 등장하고 있다. 캠퍼스터디는 캠퍼스터디 관리자가 직접 캠으로 학생들을 모니터링하여 줄거나 자리를 비우는 학생에게는 직접 연락해 경고하는 등 공부에 집중할 수 있는 조건을 제공하는 플랫폼이다.

사람들 간 직접적인 모임이 어려워지고, 공부를 할 수 있는 공간이 점점 줄어들고 있는 상황에서 캠 스터디에 대한 수요는 계속 증가하고 있다. 이와 더불어 서비스를 통해 공부의 몰입도 상승효과를 기대하는 사람들의 수요도 증가하고 있다. 기존의 캠퍼스터디 서비스를 이용해본 결과 아쉬운 점들이 몇몇 발견되었고, 이를 보완한 형태의 플랫폼을 개발하고자 한다.

캠스터디를 하다보면 몰입도가 줄어들어 집중력이 떨어질 수 있다. 따라서 기존의 캠퍼스터디 기능과 더불어 공부에 집중할 수 있는 도구를 제공함으로써 공부에 몰입할 수 있도록 지원한다. 또한 같이 공부하는 사람들끼리 정보나 자료를 공유함으로써 시너지 효과를 줄 수 있는 게시판을 지원하는 플랫폼을 개발하고자 한다.

1.2. 프로젝트 최종 목표

스스로 공부 습관을 관리할 수 있는 캠퍼스터디 플랫폼을 이용하고자 하는 사용자를 위해 ASMR, 포모도로 시계와 같은 도구를 제공하여 스터디의 효과를 증진시키는 플랫폼을 개발한다. 또한 장소, 디바이스에 제한을 받지 않도록 데스크탑 뿐만 아니라 모바일에서도 작동하도록 플랫폼을 개발한다.

1.3. 중심어

한글	비대면	캠스터디	웹 어플리케이션	플랫폼	스트리밍
영문	Untact	Cam Study	Web Application	Platform	Streaming

2. 관련 기술 현황

2.1. 국내 제품 현황

코로나 사태가 장기화 됨에 따라 2020년 2월을 기점으로 구루미 캠퍼스터디 플랫폼 가입자 수가 약 3배 가량 증가하였으며, 현재도 수요자는 계속 증가 추세이다. (<그림1>¹⁾ 참고) 구루미 캠퍼스터디 플랫폼은 캠퍼스터디 플랫폼중 자사에서 직접 개발한 플랫폼을 활용하여 서비스를 제공하는 플랫폼이며, 이용자수가 많은 대표적인 캠퍼스터디 플랫폼중 하나이다.



<그림1> 구루미 캠퍼스터디 신규 가입자 및 신규 스터디룸 개설 추이

1) "'취준생-공시족 온라인서 모이자' 캠퍼스터디 폭증", 동아닷컴, 입력 2020-03-03 03:00:00 수정 2020-03-03 03:00:00

<http://www.donga.com/ISSUE/Vote2016/News?m=view&date=20200302&gid=99972444>

2.2. 국내 경쟁 제품 현황

<그림2>²⁾과 같이 캠퍼스터디를 위한 자체 플랫폼을 개발하여 서비스를 제공하는 업체도 있고, <그림3>³⁾과 같이 기존의 화상회의 플랫폼을 활용하여 캠퍼스터디 서비스를 제공하는 업체도 있다. 특히 <그림4>⁴⁾는 잇올 온라인 캠퍼스터디케어는 기존의 플랫폼과 차별점을 두기 위해 관리자가 스터디룸에 참가하여 집중 여부를 확인하는 서비스를 제공하고 있다.



<그림2> 캠퍼스터디 플랫폼 구루미 스터디 화면



<그림3> 화상회의 플랫폼 줌을 이용한 캠퍼스터디 모인

2) "‘카공족’ 가고 ‘캠공족’ 온다", 조선일보, 2020.09.19 작성, 2020.09.26 접속.

<https://www.chosun.com/national/weekend/2020/09/19/YA6BGBHNNRDMPNUEBHAOGAEOTY/>

3) "'온라인 독서실' 구루미캠퍼스터디, 밀레니얼 세대 새 공부문화로 부상", 동아닷컴, 입력 2019-06-26 11:29 수정 2019-06-26 16:21, 2020.09.26 접속

<https://www.donga.com/news/List/Politicshttp/article/all/20190626/96190773/2>

4) "잇올 그룹, 학습 관리형 ‘캠퍼스터디’ 출범… 교육 시장에 ‘묘수’ 던지다", 동아닷컴, 입력 2020-09-21 17:39수정 2020-09-21 22:56, 2020.09.26 접속

<https://www.donga.com/news/article/all/20200921/103038682/2>

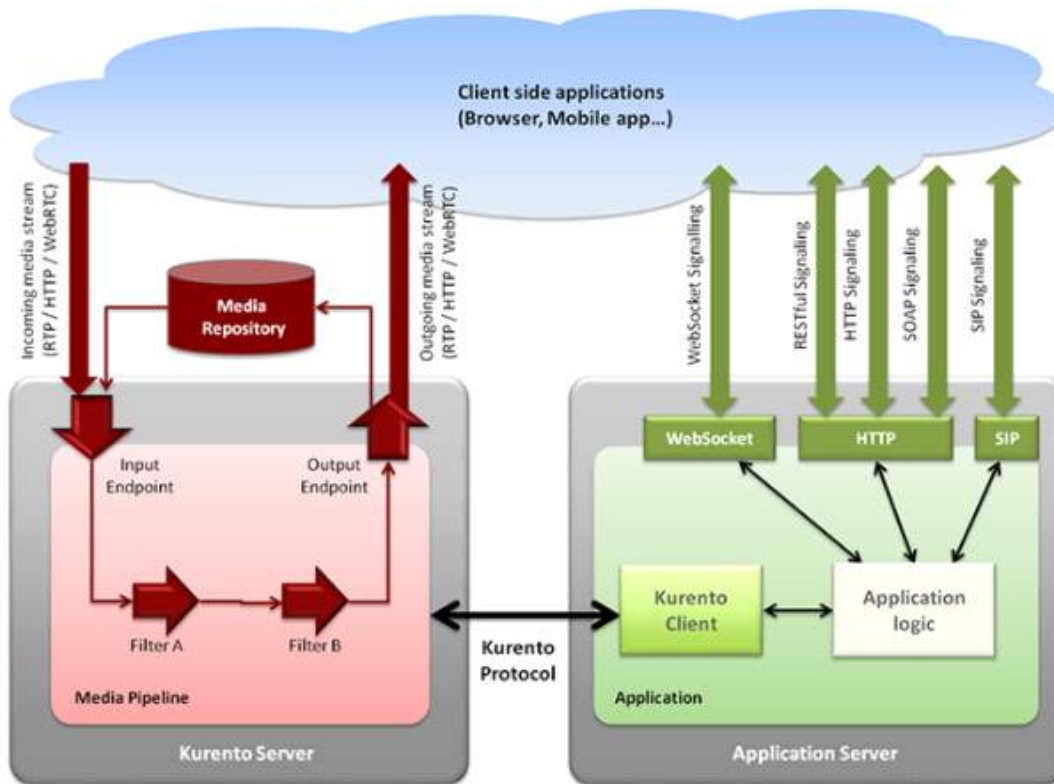


〈그림4〉 집중 여부 관리를 위한 캠퍼터디 관리자

2.3. 본 프로젝트의 차별성

기존 캠퍼터디 플랫폼에서는 단순히 캠퍼터디를 할 수 있는 방을 개설하여 사람들이 온라인상에서 모여 공부할 수 있는 기능만을 제공한다. 하지만 본 프로젝트의 경우 코인을 이용하여 방 개설, ASMR재생, 클래식 음악 제공, to do list 등을 구매하여 기존의 캠퍼터디 플랫폼의 부족한 점을 보완하여 공부에 좀 더 몰입할 수 있는 보조도구를 제공한다.

2.4. 관련 기술



<그림5> Kurento Architecture

WebRTC는 P2P 연결을 통해 브라우저와 모바일 애플리케이션에 RTC (실시간 통신) 기능을 제공하는 프로토콜, 메커니즘 및 API 세트이다.(<그림5>⁵⁾ 참고)

WebRTC 미디어 서버는 소스에서 대상으로 이동할 때 미디어 트래픽이 통과하는 멀티미디어 미들웨어이다.

- 그룹 통신 : 한 피어가 생성하는 미디어 스트림을 여러 수신자에게 배포한다. 즉, 다중회의장치 ("MCU") 역할을 한다.
- 믹싱 : 여러 수신 스트림을 하나의 단일 복합 스트림으로 변환한다.
- 트랜스 코딩 : 호환되지 않는 클라이언트 간 코덱 및 형식을 즉석에서 조정한다.
- 녹음 : 미디어가 피어간에 교환되는 지속적인 방식으로 저장한다.
- Kurento Media Server는 미디어 전송, 처리, 기록 및 재생을 담당한다.
- HTTP , RTP 및 WebRTC를 포함한 네트워크 스트리밍 프로토콜 .
- 미디어 믹싱 및 미디어 라우팅 / 디스 패칭을 모두 지원하는 그룹 통신

5) "Kurento Architecture. Kurento architecture follows the traditional separation between signaling and media planes."

http://builds.kurento.org/dev/master/latest/docs/mastering/kurento_architecture.html

(MCU 및 SFU 기능).

- Computer Vision 및 증강 현실 알고리즘을 구현하는 필터에 대한 일반적인 지원.
- WebM 및 MP4에 대한 쓰기 작업을 지원하고 GStreamer에서 지원하는 모든 형식으로 재생 하는 미디어 저장소이다 .
- VP8, H.264, H.263, AMR, OPUS, Speex, G.711 등을 포함하여 GStreamer에서 지원하는 코덱 간의 자동 미디어 트랜스 코딩.

미디어 파이프 라인을 통해 media element를 연결하는 방식은 멀티미디어 처리 방식의 복잡성을 줄이기 위한 방식이다.

3. 프로젝트 설계

3.1 주요기능

3.1.1. 회원가입 기능

Account Module을 사용하여 기능이 동작한다. 이메일과 비밀번호 입력만으로 회원가입이 가능하며, 각 회원별로 접근권한을 암호화한 토큰을 발급하여, 회원의 개인정보를 암호화하고, 회원을 관리한다.

3.1.2. 다양한 형태의 방 개설 및 참여 기능

Room Module을 사용하여 기능이 동작한다. 스터디 종류에 따라서 오픈 스터디의 경우 누구나 스터디에 참여할 수 있다. 다만 공부에 도움이 되는 도구들을 사용할 수 없다. 초대방식의 스터디의 경우 방장이 방을 개설하면서 비밀번호를 입력하게 함으로써 방장으로부터 비밀번호를 안내 받은 이용자들만 해당 방을 이용할 수 있다. 학습을 공유하는 장소인 게시판의 경우, 회원이라면 누구든지 게시물을 업로드하고, 읽을 수 있다.

3.1.3. 화면 공유 기능

스터디 방에 입장한 모든 사람들이 해당 기능을 이용할 수 있으며, 자신이 공유하고자 하는 창을 선택하면 모든 사람들이 같은 화면을 공유할 수 있다.

3.1.4. 채팅 기능

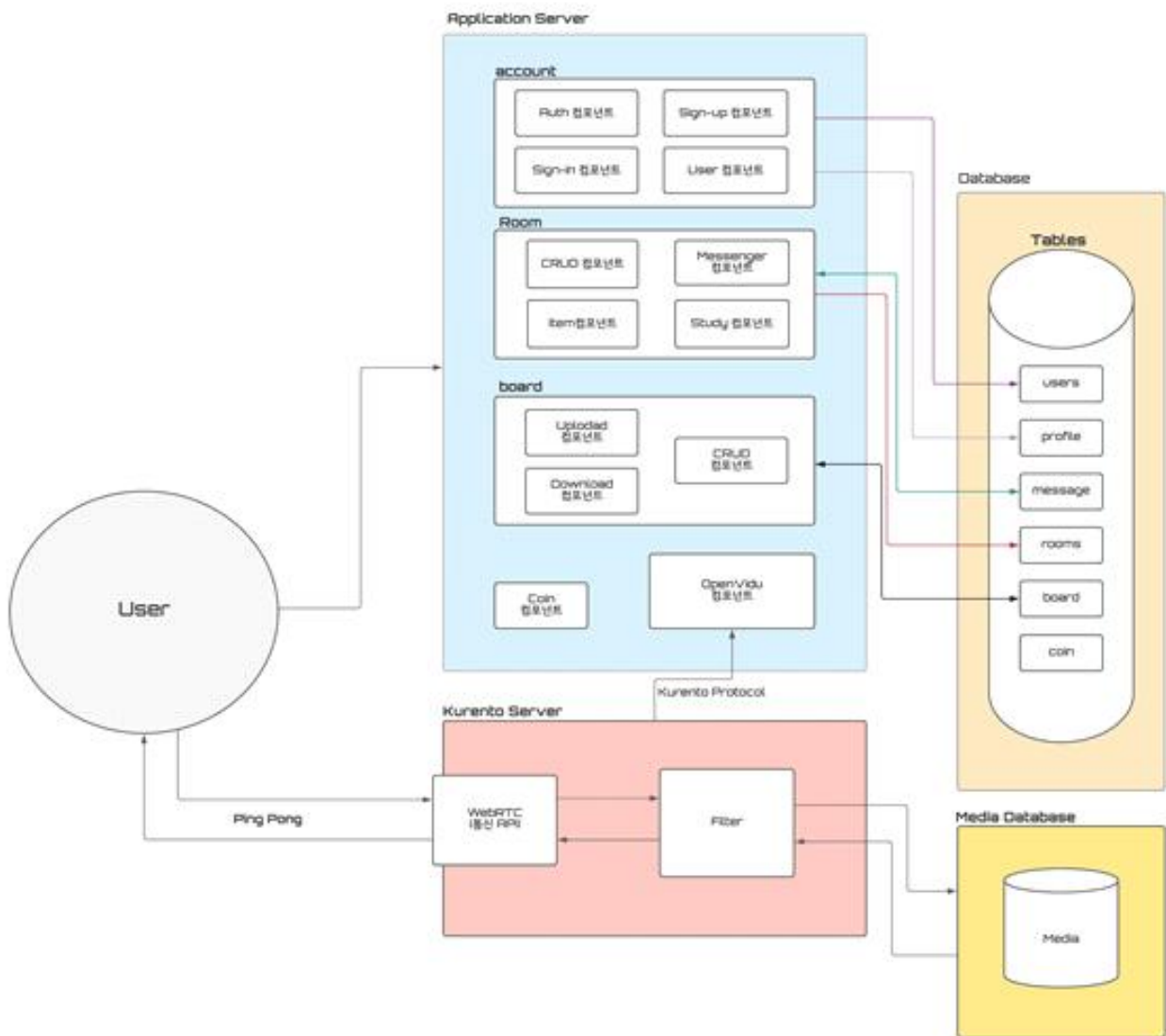
Room Module을 사용하여 기능이 동작한다. 간단한 채팅기능을 제공하여 오픈스터디나 초대방식의 스터디에서 자유롭게 채팅을 활용할 수 있다

3.1.5. 코인 시스템 제공

Coin 컴포넌트를 사용하여 기능이 동작한다. 최초 회원가입시 일정량의 코인이 제공된다. 초대방 개설시 코인을 사용하여 개설할 수 있고, 초대방에는 포모도로 시계, ASMR 등 공부에 도움이 되는 도구를 제공한다. 초대방에 있는 이용자들은 모두 해당 기능의 구매하여 도구를 이용할 수 있다.

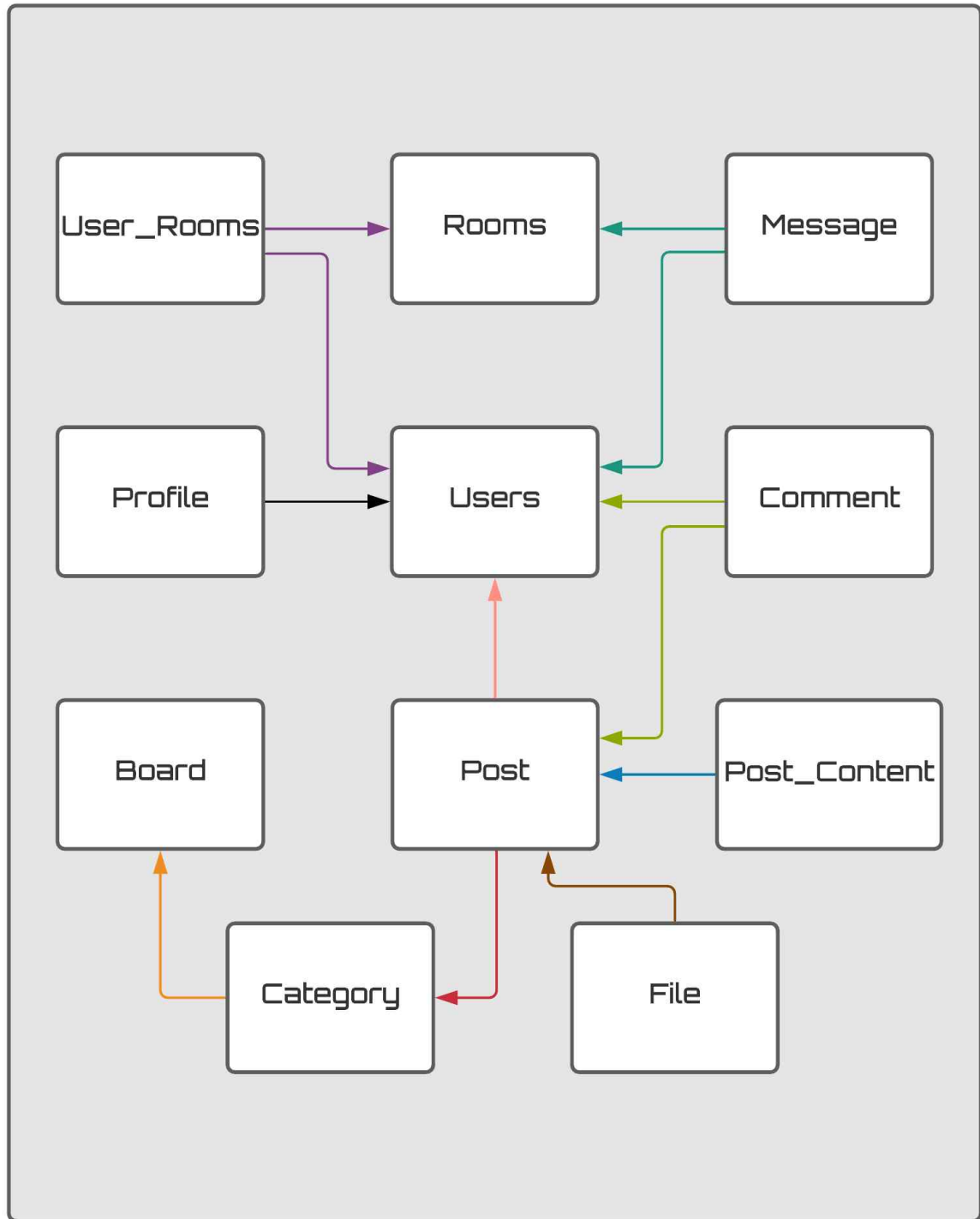
3.2 시스템 구조도

3.2.1 시스템 구조도



<그림6> 시스템 구조도

3.2.2 Database



<그림7> ER-Diagram

users							
Table logical name							
PK	AI	FK	Null	Logical Name	Name	Type	
✓	✓	+			id	INT	-
		+	✓		email	VARCHAR(45)	-
		+	✓		password	VARCHAR(45)	-
		+	✓		created_at	DATETIME	-
		+	✓		updated_at	DATETIME	-
SQL / Menu							

<그림8> users 테이블

profile							
Table logical name							
PK	AI	FK	Null	Logical Name	Name	Type	
✓	✓	+			id	INT	-
		+	✓		name	VARCHAR(45)	-
		+	✓		thumbnail	VARCHAR(45)	-
		+	✓		phone	VARCHAR(45)	-
		✓	✓		user_id	INT	-
		+	✓		created_at	DATETIME	-
		+	✓		updated_at	DATETIME	-
SQL / Menu							

<그림9> profile 테이블

users_rooms							
Table logical name							
PK	AI	FK	Null	Logical Name	Name	Type	
✓	✓	+			id	INT	-
		✓	✓		room_id	INT	-
		✓	✓		user_id	INT	-
		+	✓		status	ENUM	-
SQL / Menu							

<그림10> users_rooms 테이블

rooms							
Table logical name							
PK	AI	FK	Null	Logical Name	Name	Type	
✓	✓	+			id	INT	-
		+	✓		passcode	VARCHAR(45)	-
		+	✓		invite_link	TEXT	-
		+	✓		member_number	INT	-
		+	✓		is_private	TINYINT	-
		+	✓		created_at	DATETIME	-
		+	✓		updated_at	DATETIME	-

<그림11> rooms 테이블

message							
Table logical name							
PK	AI	FK	Null	Logical Name	Name	Type	
✓	✓	+			id	INT	-
		✓	✓		room_id	INT	-
		+	✓		content	BLOB	-
		✓	✓		user_id	INT	-
		+	✓		created_at	DATETIME	-
		+	✓		updated_at	DATETIME	-

<그림12> message 테이블

board							
Table logical name							
PK	AI	FK	Null	Logical Name	Name	Type	
✓	✓	+			board_id	INT	-
		+	✓		name	VARCHAR(45)	-

<그림13> board 테이블

post_content							
Table logical name							
PK	AI	FK	Null	Logical Name	Name	Type	
✓	✓	+			post_content_id	INT	-
		✓	✓		post_id	INT	-
		+	✓		content	BLOB	-

<그림14> post_content 테이블

post							
Table logical name							
PK	AI	FK	Null	Logical Name	Name	Type	
✓	✓	+			post_id	INT	-
		+	✓		name	VARCHAR(45)	-
		✓	✓		user_id	INT	-
		+	✓		created_at	DATETIME	-
		+	✓		updated_at	DATETIME	-
		✓	✓		category_id	INT	-

<그림15> post 테이블

category							
Table logical name							
PK	AI	FK	Null	Logical Name	Name	Type	
✓	✓	+			category_id	INT	-
		+	✓		name	VARCHAR(45)	-
		✓	✓		board_id	INT	-

<그림16> category 테이블

comment							
Table logical name							
PK	AI	FK	Null	Logical Name	Name	Type	
✓	✓	+			comment_id	INT	-
		✓	✓		post_id	INT	-
		+	✓		content	BLOB	-
		✓	✓		user_id	INT	-
		+	✓		created_at		-
		+	✓		updated_at		-

<그림17> comment 테이블

file							
Table logical name							
PK	AI	FK	Null	Logical Name	Name	Type	
✓	✓	+			file_id	INT	-
		+	✓		name	VARCHAR(45)	-
		+	✓		url	VARCHAR(45)	-
		+	✓		created_at	DATETIME	-
		+	✓		updated_at	DATETIME	-
		✓	✓		post_id	INT	-

<그림18> file 테이블

3.2.3. 시스템구조도와 데이터베이스 설명

<그림6>은 시스템 구조도이다. 구조도 상에서는 Application 서버와 Kurento 서버가 있는데 사용자들은 Application 서버 내의 3가지 모듈 기능을 사용할 수 있다. 사용자가 처음으로 웹에 들어가면 account 모듈을 이용하게 된다. Account 모듈은 사용자와 관련된 정보를 담당하며 sign-up 시에 UID, 이메일, 비밀번호를 받아 DB의 User Table에 저장하기 전에 비밀번호를 암호화한다. sign-in 할 때는 입력된 이메일과 비밀번호를 Auth 컴포넌트를 통해 User table에 있는 값과 비교, 검증하여 입력된 정보가 동일하다면 Sign-in을 허용한다. User 컴포넌트에서는 사용자의 개인 정보를 Profile table에 저장하고, 웹사이트 내에서 사용자의 개인 정보를 요청하면 Profile table에서 이름과 전화번호 등과 같은 정보를 가져온다. Room 모듈에서는 사용자가 방 생성과 탐색, 업데이트, 삭제를 할 수 있으며 담당하는 CRUD 컴포넌트에서 Rooms table에 있는 데이터를 사용한다. 사용자 간의 메시지를 담당하는 Messenger 컴포넌트에서는 Messages table을 이용하여 사용자 간의 메시지를 제공한다. 그리고 ASMR과 같이 공부에 도움이 되는 도구를 담당하는 Item 컴포넌트를 통해 사용자의 공부에 도움이 되는 아이템을 제공하며, Study 컴포넌트에서는 스터디 방에 관련된 정보를 다룬다. Board 모듈은 게시판을 담당한다. 사용자는 CRUD 컴포넌트를 사용하여 Category table과 Post table을 통해 게시판의 카테고리 및 게시글의 생성과 탐색, 업데이트, 삭제를 할 수 있다. 사용자가 게시글 내의 파일을 다운로드하거나 게시글을 올릴 때 파일을 업로드할 때 Upload와 Download 컴포넌트를 사용한다. Coin 컴포넌트는 Users table 내의 Coin column을 사용하여 초대방을 생성하거나 초대방에 입장할 때와 초대방에서 사용 가능한 도구를 사용한 횟수에 따라 코인을 차감할 때 사용된다. 각 사용자는 Kurento 서버의 WebRTC를 통해 사용자의 영상 데이터를 Media DB에 저장한다. 사용자가 방에 입장함과 동시에 OpenVidu 컴포넌트를 사용하게 되는데 Kurento 서버에서 다른 사용자의 영상 데이터를 Media DB로부터 가져와 다수가 화상 채팅을 할 수 있도록 한다.

3.3 개발 모듈과 컴포넌트 목록

Module	컴포넌트 Name	기능	Input	Output
Account	Auth 컴포넌트	토큰 인증	token	status code
	Sign-in 컴포넌트	로그인	email, password	token
	User 컴포넌트	계정 정보	email, 계정정보	status code
	Sign-up 컴포넌트	회원가입	email, password	token
Room	room crud 컴포넌트	채팅방 생성, 삭제, 변경, 탐색	email, room id	status code
	messenger 컴포넌트	채팅하기	email, 채팅메시지	status code
	item 컴포넌트	공부도구 관리	email, time	status code
	study 컴포넌트	스터디 오픈시간, 스터디 시간, 스터디 종류	study 관련 데이터	status code
Board	upload 컴포넌트	파일 저장	업로드 할 파일	status code
	download 컴포넌트	파일 다운로드	파일 id	파일
	게시물 crud 컴포넌트	게시물 CRUD	게시물 id, 내용	status code
	coin 컴포넌트	코인 CRUD	id, 코인 사용량, 코인 획득량	status code
	openvidu 컴포넌트	영상 송수신	영상 연결 정보	연결 성공 여부

3.4 개발 모듈과 컴포넌트 상세내용

Server는 Kurento Server, Application Server로 나뉜다. Kurento Server는 네트워크 스트리밍을 담당하는 서버로 영상을 스트리밍 하는 것과 더불어 Application Server를 통해 개발자가 기능을 추가하도록 함으로써 다양한 형태의 스트리밍 서비스를 지원 한다. Application Server는 부가적인 기능을 추가하기 위한 서버로 본 서비스를 위해 Account, Room, Board, Coin, 총 4가지 모듈로 구성되어 있다.

3.4.1. Account Module

Account모듈은 계정관리 모듈이다. 사용자가 회원가입, 로그인, 소셜 로그인을 할 때, 로그인 시 회원 인증의 용도와 계정관리를 할 때 사용되며 Sign Up 컴포넌트, Sign In 컴포넌트, User 컴포넌트, Social Login 컴포넌트, Auth 컴포넌트로 구성되어 있다.

3.4.1.1. Auth 컴포넌트

Auth 컴포넌트는 로그인시 회원 인증을 담당하고 있다. Auth는 인증(Authentication)과 인가(Authorization)을 의미하는 것으로 클라이언트가 누구인지 식별하고, 식별된 클라이언트가 해당 리소스에 접근할 수 있는 권한이 있는지 검사한다. 클라이언트 측에서 HTML5 Web storage(Local Storage, Session Storage)나 Cookies에 가지고 있던 token을 input값으로 보내면 해당 시크릿 키와 알고리즘으로 디코딩(복호화)을 한다. 디코딩을 통해 나온 값을 이용해 클라이언트의 권한이 어디까지인지 확인하며, 데이터베이스 user 테이블에 해당되는 row를 꺼내온 뒤 해당 output값을 JSON 형태로 sign in 컴포넌트에 보내준다.

3.4.1.2. Sign In 컴포넌트

Sign In 컴포넌트는 로그인 기능을 담당한다. HTTP 메소드의 post를 사용하며, input 값으로 데이터베이스 user테이블의 row가 들어오면 user 테이블에 해당 테이블이 있는지 확인한 후 `{{message:'success'} , status code=200}` 을 보낸다. input값으로 id 또는 email과 password가 들어오면 해당 값이 유효한지 validation 하고 해당 email 또는 id가 데이터베이스 내에 존재하는지 user테이블에서 찾아본다. 데이터베이스 내에 값이 존재하지 않으면 `{{"message" : "INVALID_EMAIL_OR_PASSWORD"}, status = 400}`을 반환한다. 값이 존재할 경우, 비밀번호를 Bcrypt로 암호화한 후 check한다. check한 값이 틀리면 `{{'message':'WRONG_EMAIL_PASSWORD'},status=400}`를 반환하고, 맞으면 `{{'token':token},status=200}`을 output값으로 보낸다.

3.4.1.3. User 컴포넌트

User 컴포넌트는 계정 정보를 통한 계정 관리와 관련된 컴포넌트이다. HTTP 메소드의 get을 사용하며, 클라이언트가 가지고 있는 token을 input값으로 보내면 해당 시크릿 키와 알고리즘으로 디코딩(복호화)한 후 여기서 나온 값을 이용해 detoken을 한다. User 테이블에서 해당 클라이언트의 id를 가져오고, 가져온 id를 사용하여 Profile 테이블의 id와 동일한 row를 찾는다. 찾아온 row에서 name, thumbnail, phone, created_at 등과 같은 계정 정보를 가져와 JSON형식으로 변환해서 output으로 보내준다.

3.4.1.4. Sign Up 컴포넌트

Sign Up 컴포넌트는 회원가입을 담당한다. HTTP 메소드의 post를 사용하며, 클라이언트 측으로부터 회원 id, email, password를 input으로 받으면, 데이터베이스 User 테이블에서 받아온 컬럼의 타입과 비교하여 유효(Authorization)하고 타당(Validation)한지 확인하는 과정이 이루어지게 되고, 이 때 이메일 형식인지, 동일한 id 또는 email을 가지고 있는 사람이 이미 존재하는지 등을 확인한다. 만약 데이터베이스 내에 input값과 중복되는 값이 있다면 output값으로 `{{'message':'EXISTS_EMAIL'}, status = 400}`을 보내고, 이메일 형식이 아니라면 `{{'message':'INVALID_EMAIL'}, status = 400}`을 보낸다. 중복값이 없다면 input값으로 받은 회원 id, email, password를 데이터베이스 user 테이블에 전달하고 회원가입한 시점을 created_at와 updated_at에 전달하며 성공 시 `{{'token':token}, status=200}`을 output값으로 보낸다.

3.4.2. Room Module

Room모듈은 캠퍼스터디 방을 관리하는 모듈이다. 스터디방은 CRUD(Create Retrieve Update Delete)한 형태로 관리가 이뤄진다. 또한 스터디방에서 메시지와 공부시간을 관리하게 되는데 이때 사용되는 컴포넌트는 Room Crud 컴포넌트, Messenger 컴포넌트, Timer 컴포넌트, Study 컴포넌트이다.

3.4.2.1. CRUD 컴포넌트

Room Crud 컴포넌트는 스터디방에서 행해지는 생성, 탐색, 업데이트, 삭제 기능을 담당한다. 각각 HTTP 메소드 post(Create), get(Retrieve), put(Update), delete(Delete)로 이루어진다.

Get메소드는 스터디방을 조회하거나 검색할 때 사용한다. room_id를 input으로 보내면 데이터베이스의 rooms 테이블에서 현재 생성되어 있는 방중에서 input으로 들어온 room_id와 일치하는 방을 찾고, 존재하면 해당 방 정보와 상태값을 JSON형태의 output을 보낸다.

Post메소드는 스터디방을 생성할때 사용한다. room_id와 해당 방의 정보를 input으로 보내면 데이터베이스의 rooms테이블에 방을 생성하고, OpenVidu 컴포넌트로 클라이언트들끼리 WebRTC로 연결시킨 후, output으로 생성한 방의 정보와 초대id, 그리고 상태값을 JSON형식으로 output을 보낸다.

Put메소드는 스터디방 정보를 업데이트 할 때 사용한다. room_id와 바꾸고 싶은 방의 정보를 input으로 보내면, input값으로 보낸 방의 정보로 업데이트 하고, output으로 업데이트 된 방 정보를 JSON형식으로 보낸다.

Delete메소드는 스터디방을 삭제할 때 사용한다. 삭제하고 싶은 방의 room_id를 input으로 보내면 해당 방을 삭제하고, OpenVidu 컴포넌트를 통해 클라이언트 간 WebRtc를 끊는다.

3.4.2.2. Messenger 컴포넌트

Messger 컴포넌트는 스터디방에서 간단하게 문자로 채팅하게 해주는 기능을 담당한다. 컴포넌트로 웹소켓을 이용한 채팅 메시지 수신, 발신을 한다. HTTP프로토콜을 통해 연결이 이루어지고, handshake과정이 성공적으로 끝나면 HTTP를 웹소켓 프로토콜로 바꾸는 protocol switching과정이 진행된다. 그러면 웹소켓을 위한 새로운 소켓이 만들어지고 이 소켓을 이용하여 통신을 한다. input값으로 WebSocketSession이 들어오면 리스트에 넣어서 클라이언트를 추가하고 세션객체에서 getAcceptedProtocol()을 호출하여 메세지를 주고 받는다.

3.4.2.3. Study 컴포넌트

Study 컴포넌트는 스터디 오픈시간, 스터디 종류를 생성, 탐색, 업데이트, 삭제하는 기능을 담당한다. 즉, 스터디방을 카테고리별로 정리해주는 기능이다. 각각 HTTP 메소드 post(Create), get(Retrieve), put(Update), delete>Delete)로 이루어져 있다.

Post메소드는 새로운 스터디 종류를 생성할 때에 사용한다. category_id와 해당 방의 정보를 input으로 보내면 데이터베이스의 rooms테이블에 새로운 방의 종류를 생성하고, output으로 새로 생성한 스터디방의 종류, 상태값을 JSON형식으로 output을 보낸다.

Get메소드는 스터디 오픈시간 및 스터디 종류를 조회하거나 검색할 때 사용한다.

category_id를 input으로 보내면 데이터베이스의 rooms 테이블에서 현재 생성되어 있는 스터디 종류중에서 input으로 들어온 category_id와 일치하는 방을 찾고, 존재하면 해당 스터디 종류와 상태값을 JSON형태의 output을 보낸다.

Put메소드는 스터디 오픈시간, 스터디 종류를 업데이트 할 때 사용한다. category_id와 바꾸고 싶은 방의 정보를 input으로 보내면, input값으로 보낸 정보로 업데이트 하고, output으로 업데이트 된 스터디의 정보를 JSON형식으로 보낸다.

Delete메소드는 스터디 종류를 삭제할 때 사용한다. 삭제하고 싶은 스터디 종류의 category_id를 input으로 보내면 컴포넌트는 delete메소드를 통해 category_id를 보내면 해당 방을 삭제하고 {status:200}을 output값으로 리턴한다.

3.4.2.4. Item 컴포넌트

Item 컴포넌트는 공부 보조 도구를 관리하는 컴포넌트이다. 사용자로부터 사용하고자 하는 도구를 입력받으면 사용자의 이메일과 코인 금액이 Coin 컴포넌트로 넘어가게 되고, 이후 Coin 컴포넌트는 email과 코인 금액을 JSON형식으로 서버에 POST한다. 그리고 서버는 Coin 테이블에서 사용자의 이메일과 일치하는 레코드를 찾은 후 입력된 코인 금액만큼 차감한다. 그리고 차감이 완료되면 status:200과 함께 업데이트된 코인 값을 리턴한다.

3.4.3. Board Module

Board모듈은 게시판 관리 모듈이다. 이용자가 게시판에 글과 첨부파일을 업로드하거나 다운로드 받을 때, 그리고 게시글에 대한 CRUD(Create Retrieve Update Delete)를 할 때 사용하게 되는 모듈로 Upload 컴포넌트, Download 컴포넌트, CRUD 컴포넌트로 구성되어 있다.

3.4.3.1. Upload 컴포넌트

Upload 컴포넌트는 게시판에 올라오는 파일을 업로드하는 기능을 담당한다. 클라이언트로부터 HTML의 <form>을 통해 받은 데이터 값을 input으로 받으면 클라이언트 토큰과 함께 데이터가 컴포넌트에 전달되고, 컴포넌트는 BOTO3 라이브러리를 통해 localhost 객체에 input값을 전달한다. localhost 객체는 업로드가 완료되면 URL을 리턴해주고, 해당 URL을 Database의 File Table에 저장하게 된다. 저장이 완료되면 output으로 {"message": "success", status=201}이 리턴 된다.

3.4.3.2. Download 컴포넌트

Download 컴포넌트는 게시판에 등록된 파일을 다운로드하는 기능을 담당한다. 클라이언트가 플로피 디스켓 모양의 이미지를 클릭하면, input값으로 해당 이미지

태그에 저장된 파일의 ID값을 컴포넌트에 전달하게 된다. 컴포넌트는 Database의 File 테이블에 접근하여 ID값에 대응하는 value인 URL을 가져온 뒤 localhost에서 다운로드를 받게 된다. 그리고 다운로드 된 파일을 이용자에게 output으로 리턴 해준다.

3.4.3.3. CRUD 컴포넌트

CRUD 컴포넌트는 게시판에서 행해지는 생성, 탐색, 업데이트, 삭제 기능을 담당한다. CRUD 컴포넌트는 HTTP method post(Create), get(Retrieve), put(Update), delete(Delete)를 활용한다.

게시물 생성의 경우, 클라이언트가 내용을 채운 뒤 저장 버튼을 누르면 해당 게시물에 대한 id가 할당되어, 클라이언트 토큰, 내용과 함께 컴포넌트에 전달된다. 그러면 컴포넌트는 post method를 통해 게시물id, 클라이언트 토큰, 내용을 DB의 board 테이블에 전달하고 output으로 {status=200}을 유저에게 리턴 한다.

게시물 검색의 경우, 유저가 게시물을 누르면 해당 게시물에 등록된 id가 컴포넌트에 전달된다. 컴포넌트는 get method를 통해 데이터베이스의 board테이블에 전달되고, board 테이블에서 id에 대응하는 value 값을 리턴 해준다. 그러면 컴포넌트는 해당 value를 페이지에 가져와 내용을 채운다.

게시물 업데이트의 경우, 클라이언트가 내용을 업데이트한 후 수정 버튼을 누르면, 클라이언트 토큰, 해당 게시물의 id, 그리고 업데이트 된 내용을 컴포넌트에 전달하게 되고 컴포넌트는 put method를 통해 데이터베이스의 board 테이블에서 게시물 id,유저 토큰에 대응하는 value를 업데이트 된 내용으로 바꿔서 저장한다. 그리고 데이터베이스는 성공 시 output으로 status=200을 리턴 한다.

게시물 삭제의 경우, 클라이언트가 삭제 버튼을 누르면, 클라이언트 토큰, 해당 게시물의 id가 컴포넌트에 전달되고 컴포넌트는 delete method를 통해 데이터베이스의 board table에 전달하게 되고 테이블에서 클라이언트 토큰, 게시물 id에 대응하는 value를 삭제한다. 그리고 성공 시 output값으로 {status=200}을 리턴 한다.

3.4.4. Other Module

3.4.4.1. Coin 컴포넌트

COIN 컴포넌트는 프로그램 내에서만 통용되는 가상화폐인 코인을 담당한다. COIN 컴포넌트도 CRUD 방식으로 coin을 관리하며 HTTP method인 post(Create), get(Retrieve), put(Update)를 활용한다.

코인 생성은, 클라이언트가 회원가입시 생성되는 클라이언트 토큰을 컴포넌트에 전달한다. 컴포넌트는 해당 토큰을 post method를 이용하여 db의 coin 테이블에

전달하고 coin 테이블에서 value로 10000을 초기화한다. 그리고 리턴값으로 status=200을 보낸다.

코인 검색의 경우, 클라이언트가 코인 조회 시, 클라이언트 토큰이 컴포넌트에 전달되고, 컴포넌트는 get method를 이용하여 토큰을 DB coin 테이블에 전달한다. 테이블에서 클라이언트 토큰에 대응하는 value를 찾은 뒤 성공 시 status=200과 함께 value를 리턴 해준다.

코인 업데이트의 경우, 업데이트 조건을 충족하는 경우, 클라이언트의 토큰이 컴포넌트에 전달되고, 컴포넌트는 get method를 이용하여 토큰을 DB coin 테이블에 전달한다. 테이블에서 클라이언트 토큰에 대응하는 value를 찾은 뒤 value를 업데이트 한다. 그리고 성공 시 리턴 값으로 status=200과 함께 value를 리턴 한다.

CURD의 Delete의 경우, 코인의 삭제를 의미하나 가입된 회원에 대한 탈퇴 기능은 고려하지 않았으므로 사용되지 않는다.

3.4.4.2. Openvidu 컴포넌트

영상 송수신을 담당하는 컴포넌트로 클라이언트가 방에 입장할 때 전달받은 invite_link 값과 Kurento Server에서 가지고 있는 값을 Kurento Client를 통해 비교하여 연결이 성공적으로 이루어졌다면 클라이언트들로부터 받은 영상 스트리밍 정보를 Kurento Server로부터 받아오며 이에 따른 output은 영상 연결 성공 여부이다. HTTP method인 post, get을 활용한다.

4. 프로젝트 수행 결과

4.1. 개발 환경

back-end:

IDE: IntelliJ

Java version: 1.8

Server: localhost

DB: MySql

front-end:

IDE: Vscode, WebStorm

JavaScript version: ES6

Package Manager: npm, yarn

이미지 편집도구: photoshop cc 2020

browser: chrome

build tool:

Gradle, Maven

4.2. 활용 Framework, API, Library

분류	이름	목적
Framework	Spring Boot	상대적으로 간편하고 효율적인 서버 구축을 위해 사용
	Spring Security	사용자 인증과 접근 권한을 설정하기 위해 사용
	React	웹 페이지에서 사용될 객체들을 만들기 위해 사용
	Redux	React컴포넌트를 효율적으로 관리하고 상태를관리 하기 위해 사용
API	Spring Data JPA	jpa 기반의 레포지토리를 쉽게 실행할 수 있도록 도와주는 인터페이스로 사용
	OpenVidu	실시간 영상 스트리밍 기능을 지원하고, 부가적인 기능을 추가하기 위해 사용
	WebRTC	WebRTC 는 P2P 연결을 통해 브라우저와 모바일 애플리케이션에 RTC (실시간 통신) 기능을 제공하는 프로토콜, 메커니즘 및 API 세트
	JDBC	자바에서 DB에 접속하여 query를 이용하여 데이터를 저장, 불러오기, 업데이트를 할 때 사용
	Toast UI Chart	주어진 데이터를 도표로 시각화 하기 위해 사용
	JWT	주어진 데이터를 JSON형태로 송수신하기 위해 사용
Library	Kurento Media Server	미디어 전송, 처리, 기록 및 재생을 담당 하는 서버로 사용

4.3. 결과 화면 예시

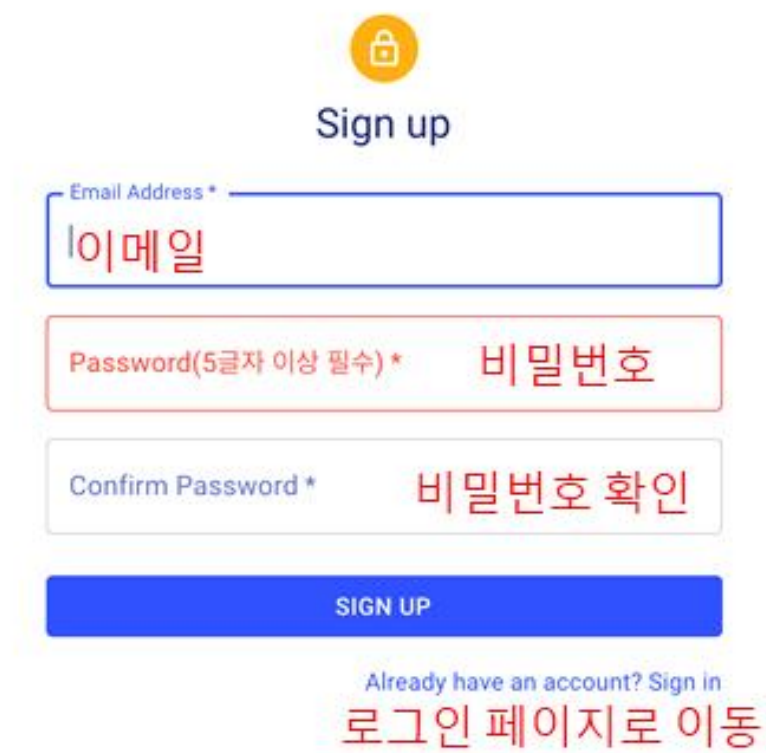
4.3.1. 시작 화면



<그림 19> 시작화면

study with me 웹 서비스에 접속하면 나타나는 첫 화면이다. Study with Me에 대한 간단한 소개와 함께 로그인과 회원가입을 할 수 있는 버튼이 등장한다. 스크롤 다운을 하면 Study with Me에 대한 자세한 설명을 볼 수 있다.

4.3.2 회원가입 페이지

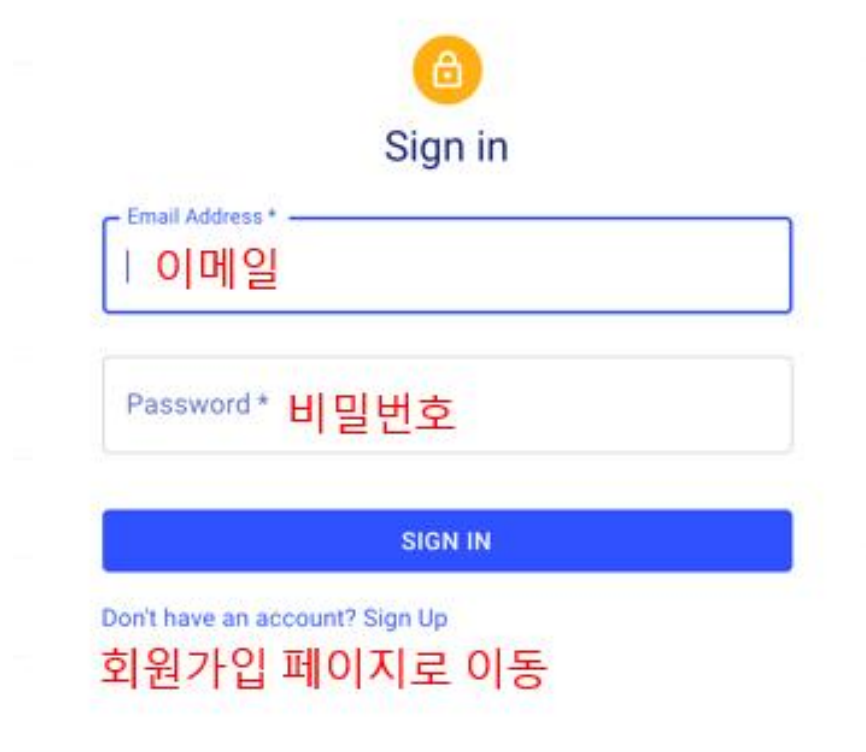


The image shows a 'Sign up' form. At the top is a yellow lock icon and the text 'Sign up'. Below are three input fields: 'Email Address *' with the placeholder '이메일', 'Password(5글자 이상 필수) *' with the placeholder '비밀번호', and 'Confirm Password *' with the placeholder '비밀번호 확인'. A blue 'SIGN UP' button is at the bottom. Below the button is a link: 'Already have an account? Sign in' and '로그인 페이지로 이동'.

<그림 20> 로그인 페이지

시작화면에서 sign up버튼을 누르면 해당 페이지로 이동한다. 이메일을 입력하고 5자리 이상의 비밀번호, 비밀번호 확인을 위해 동일한 비밀번호 값을 입력한다. 비밀번호 입력값과 비밀번호 확인 입력값이 일치하지 않는 경우, 확인란이 빨간색으로 바뀌면서 값이 일치한다는 메시지가 출력된다. 값이 일치하는 경우, 확인란이 파란색이 되면서 불일치 안내 메시지가 사라진다. sign up 버튼을 누르면, 입력된 값이 서버로 전송되며, 정상 가입이 된 경우 '회원가입 완료' 알림이 뜨면서 시작화면으로 돌아온다. 이미 계정이 있는 경우 sign up 버튼 하단에 있는 글씨를 클릭하면 로그인 페이지로 이동한다.

4.3.3 로그인 페이지



The image shows a login page with a yellow lock icon at the top. Below it is the text "Sign in". There are two input fields: "Email Address *" with a red placeholder "이메일" and "Password *" with a red placeholder "비밀번호". A blue "SIGN IN" button is below the fields. At the bottom, there is a link "Don't have an account? Sign Up" and a red text "회원가입 페이지로 이동".

<그림 21> 회원가입 페이지

로그인 화면이다. 회원가입 시 입력했던 이메일과 비밀번호를 입력 후 sign in 버튼을 클릭하면 서버로 입력 값이 전송되고, 서버에서 계정이 있는 것으로 확인 되면 서버로부터 토큰을 수령하게 되고, 로그인 완료 안내창이 뜨게 되며, 확인버튼 클릭시 대기방으로 이동하게 된다. 계정이 존재하지 않거나, 정보를 잘못 입력한 경우, 값을 다시 입력하라는 안내창이 뜨고, 해당 페이지에 머물게 된다. 회원가입이 되어 있지 않은 경우, sign in 버튼 하단의 글을 클릭하면 회원가입 창으로 이동하게 된다.

4.3.4. 캠퍼디 방선택 페이지

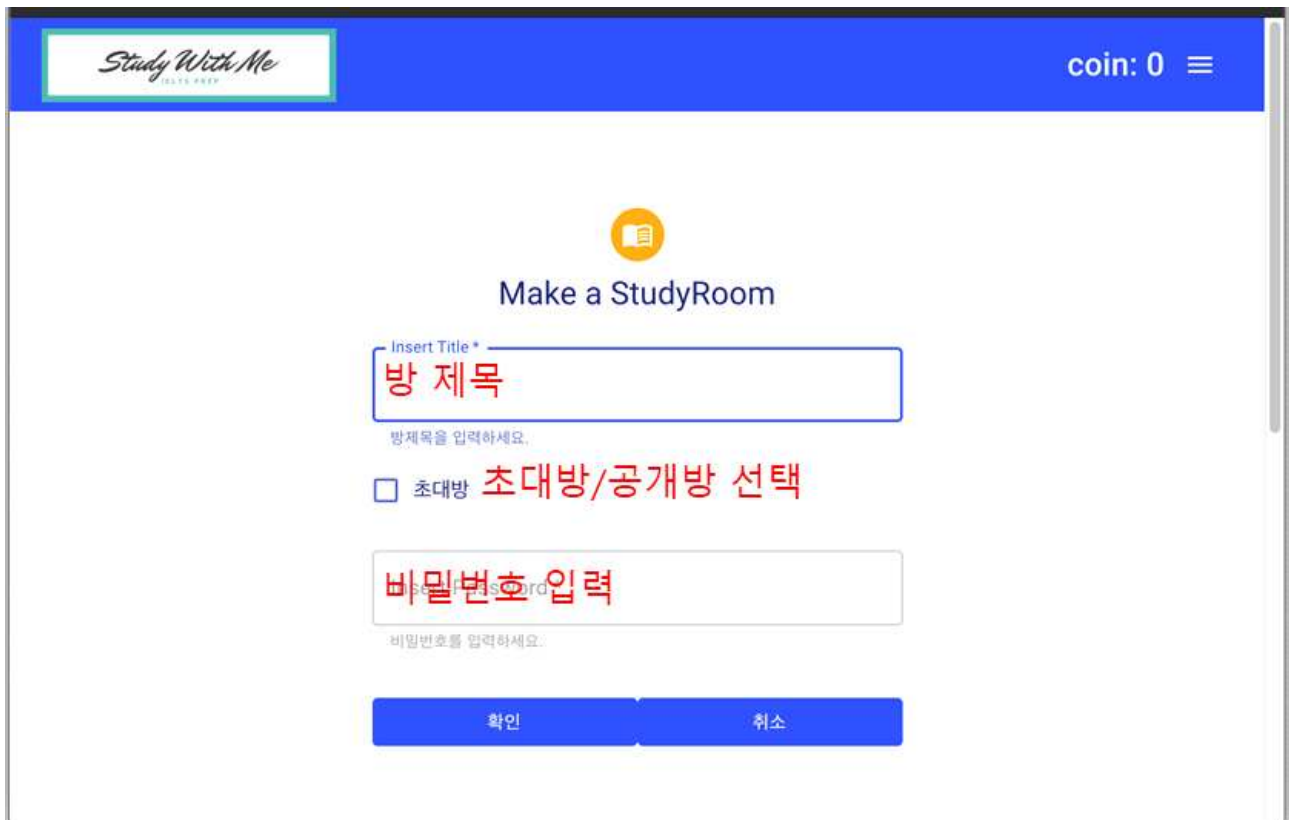


<그림 22> 방선택 페이지

정상적인 로그인 후에 접속하게 되는 방 선택 페이지이다. 상단의 바에는 홈 버튼, 자신이 가지고 있는 코인의 수, 로그아웃 버튼으로 구성된다. 홈 버튼을 누르게 되면 대기방 페이지로 이동하게 된다. 코인 옆 숫자는 자신이 가지고 있는 코인의 수를 나타내며, 해당 코인을 이용하여 초대방을 개설하거나, 초대방에서 공부 관련 도구들을 구매할 수 있다. 로그아웃 버튼을 클릭하면, 서버로부터 수령했던 토큰을 삭제하고, 로그아웃하게 된다. 로그아웃이 되면, 시작페이지로 이동하게 된다.

상단바 밑에 있는 검색란을 통해 방을 검색할 수 있다. 검색하고자 하는 방의 이름을 입력하고 검색 버튼을 누르면, 해당 방이 검색 결과로 나오게 되며, 일치하는 방이 없는 경우 아무것도 나오지 않는다. 이후 초기화 버튼을 누르면 검색 결과가 초기화 되면서 서버에 등록된 모든 방들의 목록이 나오게 된다. 방 만들기 버튼을 누르면 방을 만들 수 있는 창이 나온다.

4.3.5. 캠퍼스터디 방 만들기 페이지



Study With Me

coin: 0

Make a StudyRoom

Insert Title *

방 제목

방제목을 입력하세요.

☐ 초대방 초대방/공개방 선택

비밀번호 입력

비밀번호를 입력하세요.

확인 취소

<그림 23> 방 만들기 페이지

방 만들기 버튼 클릭 시 나오는 방 만들기 페이지이다. 방 제목을 입력 후 초대방, 공개방을 선택한다. 초대방의 경우 비밀번호를 입력해야 입장할 수 있는 방으로, 방장이 비밀번호를 알려준 경우에만 접속할 수 있다. 해당 방의 경우 일정 금액의 코인을 지불해야 생성이 가능하다. 초대방을 체크하면 비밀번호를 입력후 확인을 누르면 방이 개설된다. 반면 공개방의 경우 누구나 입장 가능한 방으로 코인 지불 없이 방을 개설할 수 있다. 초대방 체크를 하지 않고, 바로 확인 버튼을 누르면 공개방이 개설된다.

4.3.6. 스터디방 페이지



<그림 24> 스터디 방 페이지

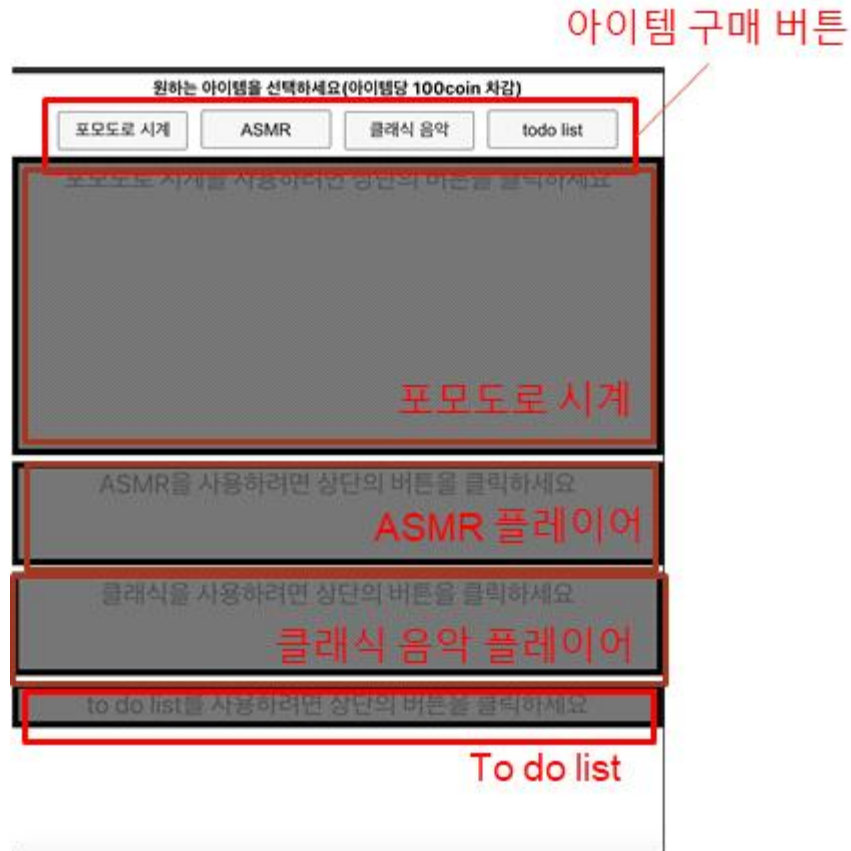
스터디 방에 입장하게 되면 나타나는 페이지이다. 상단바에는 방제목이 있으며, 아이콘은 순서대로 마이크 on/off, 카메라 on/off, 화면공유 on/off, 전체크기, 모든 음성 제거, 종료를 의미한다. 화면 공유의 경우 화면을 화면을 선택할 수 있는 탭이 나타나며 본인이 원하는 탭을 클릭하여 공유하고자 하는 화면을 선택할 수 있다. 상단바 맨 끝에 있는 버튼은 채팅활성화 버튼이다. 해당 버튼을 누르면 오른쪽에 채팅창이 활성화되며 하단의 입력창을 통해 채팅을 할 수 있다. 화면 메인에는 스터디 참여자들의 화면이 카메라를 통해 실시간으로 스트리밍 된다.



<그림 25> 스터디 방 동시접속 화면

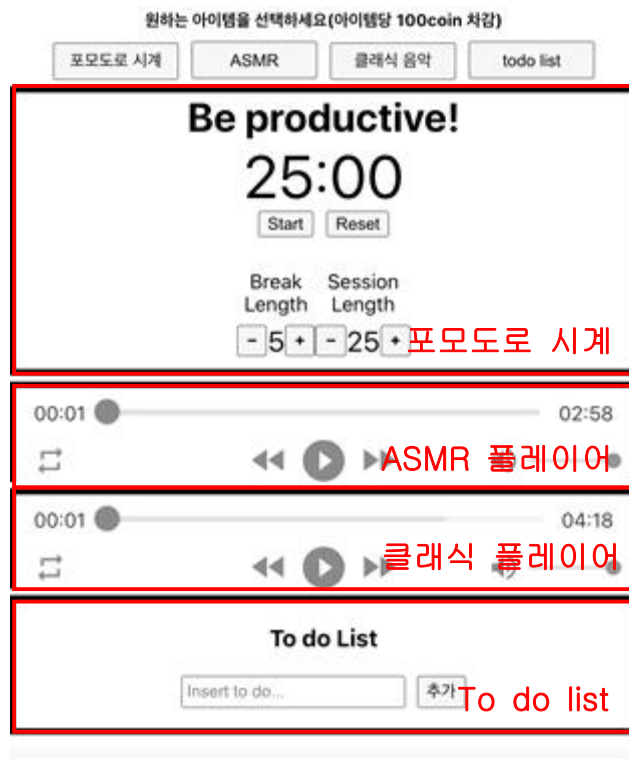
여러 명의 사용자들이 동시에 접속한 화면이다. 접속자의 수에 따라 화면의 크기가 유동적으로 변하며, 각 화면에서 마이크의 활성화 여부를 확인할 수 있다. 데스크탑을 비롯하여 스마트폰, 태블릿PC에서도 접속할 수 있다.

4.3.7. 공부 도구 페이지



<그림 26> 공부 도구 페이지(도구 비활성화)

초대방에 입장하면 나타나는 공부 도구 페이지이다. 코인을 이용하여 원하는 공부 도구를 구매할 수 있으며, 원하는 도구의 버튼을 클릭하면 해당 아이템을 구매하였다는 안내 메시지가 나타나며, 100코인이 차감된다. 이후 해당 아이템이 활성화된다.



<그림 27> 공부 도구 페이지(도구 활성화)

구매한 아이템들이 활성화된 화면이다. 순서대로 포모도로 시계, ASMR 플레이어, 클래식 음악 플레이어, To do List이다.

포모도로 시계의 경우, 공부할 시간과 중간 휴식시간을 사용자가 직접 선택할 수 있다. Session Length를 통해 공부시간을 설정하고, Break Length를 통해 휴식시간을 설정할 수 있다. start버튼을 누르면 공부 시간이 활성화 되면서 카운트 다운이 시작된다, 이후 시간이 종료되면 비프음이 나오면서 설정한 휴식시간만큼 카운트 다운이 시작된다. 설정한 값을 초기화 하고 싶으면 Reset 버튼을 클릭하면 된다.

ASMR과 클래식 음악의 경우, 아이템을 구매한 동시에 활성화 되어 자동으로 내장된 음원이 재생된다. 각 플레이어 별로 시작/일시정지, 음량 조절을 할 수 있다.

To do list의 경우, 자신이 해야 할 일을 입력후 추가 버튼을 클릭하거나 엔터 키를 누르면 자동으로 리스트에 추가된다. 입력된 리스트의 좌측에는 체크버튼이 생기며, 우측에는 수정, 삭제 버튼이 있다. 수정 버튼을 누르면, 입력값을 수정할 수 있으며, 수정 버튼을 누르면 수정된 입력값이 저장된다. 삭제 버튼을 누르면 해당 리스트는 삭제된다.

4.3.8. 게시판 리스트 페이지

글쓰기

번호	제목	작성자	작성일
1	함께 프로그래밍 공부해요(자료 링크 첨부)	rla96@gmail.com	2020-12-15T17:30:23.593021
2	study with me	rla96@gmail.com	2020-12-15T17:30:37.677292
3	공무원 9급 후기	rla96@gmail.com	2020-12-15T17:31:03.019487

1

게시물 보기

게시물 리스트

Pagenation(10개 단위로 나눔)

<그림 28> 게시판 리스트 페이지

메인화면에서 게시판 버튼을 클릭하면 나타나는 게시판 리스트 페이지이다. 각 게시물마다 고유번호가 있으며, 리스트에서는 게시물번호, 제목, 작성자, 작성일시 순으로 나열된다. 한 페이지에 게시물의 개수가 10개가 넘어가는 경우 자동으로 pagination이 되어 페이지 번호가 자동으로 생성된다. 게시물 내용을 확인하려면 해당 게시물의 제목을 클릭하면 된다.

4.3.9. 게시판 작성 페이지

Study With Me coin: LOGOUT

제목
제목을 입력하세요. **제목 작성**

내용
내용을 입력하세요. **내용 작성**

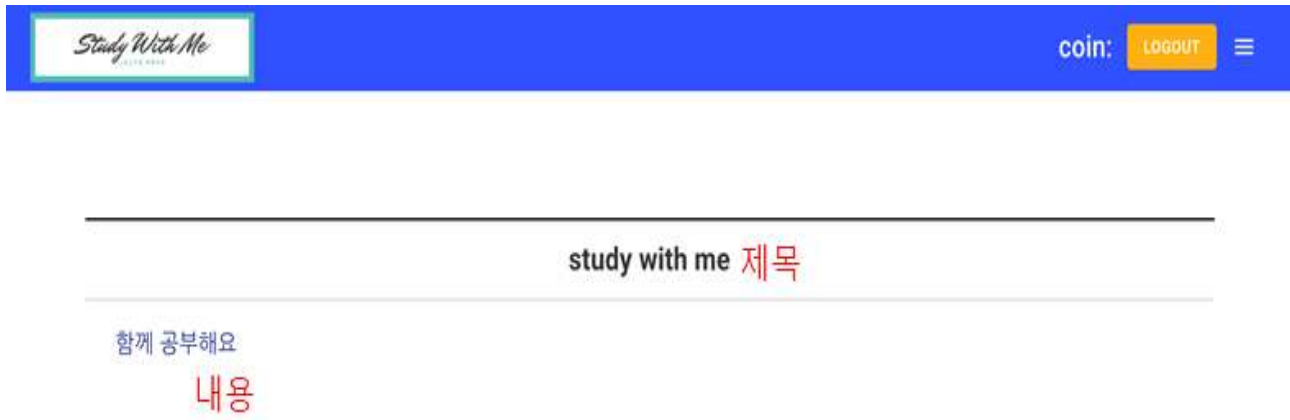
첨부파일
파일 선택 선택된 파일 없음

저장 **저장**

<그림 29> 게시판 작성 페이지

게시판 리스트 화면에서 글쓰기 버튼을 클릭하면 나타나는 게시판 작성 페이지이다. 게시물 제목을 입력하고 내용을 입력한 후 저장 버튼을 누르면 자동으로 게시물에 고유번호가 할당되고, 작성자 이메일, 작성일시와 함께 서버로 전송된다. 이후 해당 내용은 게시물 리스트에서 확인할 수 있다.

4.3.10. 게시판 읽기 페이지



<그림 30> 게시판 읽기 페이지

게시물 리스트에서 읽고자 하는 페이지의 제목을 클릭하면 나타나는 게시판 읽기 페이지이다. 게시물의 제목과 내용을 확인할 수 있다.

4.4. 평가

4.4.1 기능 평가 항목

기능	평가 목적
로그인	로그인이 정상적으로 작동하도록 한다.
통합 회원가입	통합 회원가입이 정상적으로 작동하도록 한다.
스터디방 개설	사용자가 원하는 스터디방을 개설하여 참여하도록 한다.
스터디방 참여	사용자가 원하는 스터디방을 탐색하여 참여하도록 한다.
게시판	게시판을 통하여 사용자가 원활하게 이용하도록 한다.
채팅	채팅을 실시간으로 이용하도록 한다.
코인 시스템	코인 시스템을 정상적으로 작동하도록 한다.
실시간 영상 송수신	OpenVidu API를 이용하여 실시간 영상 송수신이 정상적으로 작동하도록 한다.

4.4.2 기능 평가 결과

기능	시나리오	평가 결과
로그인	이메일 미입력	아이디 미입력 시 서버에서 오류반환
	비밀번호 미입력	비밀번호 미입력 시 서버에서 오류반환
	계정정보 없음	같은 계정일 때 서버에서 오류반환
통합 회원가입	이메일 중복	이메일 중복일 때 서버에서 오류반환
	비밀번호 유효성 검사	비밀번호 불일치 시 오류 메시지 반환
	이메일, 비밀번호 미입력	이메일, 비밀번호 미입력 시 서버에서 오류반환
스터디방 개설	방이름 미입력	방이름 미입력 시 서버에서 오류반환
	초대방 비밀번호 미입력	초대방 비밀번호 미입력 시 서버에서 오류반환
	방이름 중복	방이름 중복 시 서버에서 오류반환
스터디방 참여	초대방 입장시 비밀번호 불일치	방생성시 입력된 비밀번호와 불일치시 오류반환
	초대방 입장시 비밀번호 일치	스터디방에 입장되면서 공부 도구 페이지도 동시에 열림
게시판	게시판 작성후 서버에 등록	서버로 데이터 전송되어 DB에서 입력된 내용 확인 가능
	게시판 리스트에서 일정 게시물 개수 초과시 자동 페이지 네이션	10개 이상 초과시 10개 단위로 페이지네이션
	게시물 내용 확인	게시물 리스트에서 제목 클릭시 해당 게시물 내용 확인가능

채팅	채팅글 입력후 구성원 전부에게 전달	구성원 전부 채팅 입력 내용 확인 가능
코인 시스템	초대방/공개방 구분하여 초대방 생성시 코인 차감	초대방에 한하여 코인 차감 확인 가능
	공부 도구 구매시 코인 차감	도구별로 일정 금액 차감 확인 가능
실시간 영상 송수신	다수 인원 동시 접속	인터넷 환경에 따라 딜레이가 있긴 하나 동시에 접속 확인 가능
	다양한 디바이스에서 스터디룸 접속	데스크탑(MacOS, windows), 스마트폰(iOS,Android), 태블릿PC(IPad)에서 접속 가능
	화면 공유 기능	공유하고자 하는 페이지 선택시 화면 공유 가능

5. 기대 효과 및 활용 방안

5.1. 기대 효과

□ 사회.문화적 측면

5.1.1. 비대면 시대에 집에서 혼자 공부할 때 경쟁자의 모습을 보며 공부 의욕향상

우리가 평상시에 도서관에 가서 공부하면서 다른 공부하는 사람들을 바라보면서 공부 의욕이 향상되고 공부에 더욱 집중 할 수 있듯이, 코로나 바이러스로 인해 도서관에 직접 가서 공부할 수 없는 비대면 시대를 맞이하여 집에서 도서관에서 공부하는 효과를 줄 수 있는 캠 스터디를 통해 서로 공부하는 모습을 공유하여 공부 의욕을 향상 할 수 있다.

5.1.2. 같은 목표로 공부하는 사람들 간 정보 교환 및 질의응답을 통한 수행능력 향상

같은 시험을 목표로 공부하는 사람들 간에 게시판을 통해 시험 자료를 공유하고 질문을 함으로써 서로 간에 정보 교환 및 질의응답을 하면서 수행능력을 향상시킨다.

5.1.3. 혼자 공부할 때 집중이 힘든 사람들을 위한 인터넷 스터디 제공으로 집중력 향상

자신의 공부하는 모습을 남에게 보여준다는 생각에 다른 짓을 하지 않고, 상대방의 공부하는 모습을 보면서 자극받고 공부에 집중할 수 있도록 도와준다.

□ 기술적 측면

웹 어플리케이션 구현을 통한 웹에 대한 이해도 향상 및 구축 능력 향상
 웹 어플리케이션 구현을 직접 해봄으로써 Front-End와 Back-End 시스템을 이해할 수 있고, 웹 플랫폼을 직접 프로그래밍 해봄으로써 구축 능력이 향상된다.
 또한 Front-End와 웹서버 간에 통신을 함으로써 서버통신 시스템을 이해하고 프로젝트 시 팀원들 간에 의사소통을 함으로써 소통능력과 팀 프로젝트 수행 관리 능력을 향상시킨다.

5.2. 활용 방안

- 그룹 스터디 플랫폼으로 같은 시험을 목표로 하는 사람들 간 공부의욕을 향상시킬 수 있다.
- 오프라인 스터디에서 이뤄지는 자료 공유가 게시판을 통해 용이하게 이뤄질 수 있다.

6. 프로젝트 수행 체계

6.1. 팀원 인적사항

성명	소속	경력
김찬민 (팀장)	소프트웨어학부 (3학년)	- “졸음 방지 시스템 개발” 금상 (창의적공학설계, 2017.9~12) - "보드 게임, war game 개발" (네트워크프로그래밍, 2020.6~7)
김병준	소프트웨어학부 (3학년)	- 쇼핑몰 사이트 클론 개발, 여행 예약 사이트 클론 개발
이주환	소프트웨어학부 (3학년)	- 미아사고 방지용 rfid 모니터링시스템에 관한 연구 (창의적 공학설계 2017.9~12) - 블록체인을 활용한 온라인 투표 시스템 (소프트웨어 프로젝트, 2020.3~6)
장유수	소프트웨어학부 (3학년)	- “보행자 감지 신호체계” (창의적공학설계, 2017.9~12) - "성취 동기 증진을 위한 앱 어플리케이션 개발" (소프트웨어 프로젝트, 2020.3~6)

6.2. 프로젝트 역할 분담

성명	소속	역할	기여 도
김찬민	소프트웨어학부 (3학년)	팀장 Front-End 개발 : 로그인 페이지, 게시판 페이지 제작	25%
김병준	소프트웨어학부 (3학년)	Back-End 개발 : 스트리밍, rekognition API 제작	25%
이주환	소프트웨어학부 (3학년)	Back-End 개발 : 회원관리, 로그인, 채팅 API 제작	25%
장유수	소프트웨어학부 (3학년)	Front-End 개발 : 방 리스트 페이지 제작, 스터디 서비스 페 이지 제작	25%

6.3. 프로젝트 수행 일정

구분	추진내용	수행기간													
		9-1	9-2	9-3	9-4	10-1	10-2	10-3	10-4	11-1	11-2	11-3	11-4	12-1	12-2
계획	프로젝트 구체화, 구현 회의														
분석	기존 웹 프로젝트 톨 분석 요구사항 정의 개념모델 구성														
설계	웹 설계														
	DB 및 서버기능 설계														
개발	Front-End(웹 기능 개발)														
	Back-End(서버 및 DB 기능 개발)														
	Fromt-End – Back-End간 통신														
테스 트	최종 프로젝트 마무리														
종료	진행된 프로젝트에 대한 피드백 및 후속작업														

6.4. 프로젝트 수행 방법

- SW개발 방법론: 애자일 소프트웨어 개발 방법론
- 회의 방법 (On/Off-line 여부) 및 주기

Front End: 매주 목요일 On-line에서 1시간 가량 진행

Back End: 매주 수요일 On-line에서 1시간 가량 진행

전체: 격주 일요일 On-line에서 2시간 가량 진행

- 기타 SW 공유, 문서 공유 방법

On-line회의: zoom

자료수집: Slack

프로젝트 관련 문서: Slack

회의 및 프로젝트 진행 상황 관련 자료 공유: Slack

6.5. 프로젝트 자체 평가

평가 항목	평가 항목별 포함 여부			자체평가결과 (10점 만점)
	수행계획서	중간보고서	최종보고서	
개발의 필요성	○	△	△	7
목표의 명확성 및 타당성	○	△	△	6
제안 방법의 독창성	○	△	△	7
설계의 구체성	○	○	○	8
프로젝트 기술성	○	○	○	9
프로젝트 완성도	○	△	○	8
성능 평가의 우수성	○	△	△	7