**파스-타 기반 서비스 개발 및 아이디어 공모전 작품 접수 신청서**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 공모 부문 | | **◼**서비스 개발 □ 아이디어 | | | | | | |
| 중고등학생  참가팀 여부 | | □ |  | | | | | |
| 신  청  인 | 팀명 | 봉골레머골레 | | | | | | |
| 성명 | 박예진 | | | | | | |
| 소속 | 상명대학교 | | | | | | |
| 이메일 | yeyeah5131@naver.com | | | | 연락처 | 010-3284-5131 | |
| 팀  원 정  보 | 성명 | 소속 | | | 이메일 | | | 연락처 |
| 김윤진 | 상명대학교 | | | yunjin3052@naver.com | | | 010-3052-3181 |
| 박예진 | 상명대학교 | | | yeyeah5131@naver.com | | | 010-3284-5131 |
| 황명연 | 상명대학교 | | | auddus16@naver.com | | | 010-7659-2180 |
| 김성은 | 상명대학교 | | | k0qf\_1e@naver.com | | | 010-17159-6599 |
|  |  | | |  | | |  |
| 개인정보 수집·이용  안내 및 동의 | | ① 개인정보의 수집 및 이용목적과 수집항목 및 수집방법 | | | | | | |
| 파스-타 기반 서비스 개발 및 아이디어 공모전 작품 접수 신청을 위하여 다음과 같은 개인 정보를 수집합니다.  - 필수항목 : 성명, 이메일, 연락처 (팀원 정보 포함)   * 선택항목 : 소속   ※ 필수항목 동의 거부 시 공모전 신청이 불가하며, 선택항목 동의 거부에 대한 불이익은 없음 | | | | | | |
| ② 개인정보 보유 및 이용기간 (파기절차 및 방법) | | | | | | |
| 수집된 개인정보는 2021년 12월 31일까지 파스-타 기반 서비스 개발 및 아이디어 공모전 관리를 위해 사용되며, 이후 전체 수집 데이터 삭제를 통해 파기처리를 합니다. | | | | | | |
| ③ 동의 여부 | | **◼** 동의 □ 비동의 | | | | |
| 작품명 | | 런업툴 | | | | | | |
| 작품 접속 URL | | https://runuptoolcloud22.paas-ta.org/swagger-ui/index.html | | | | | | |
| 작품 요약 | | 런업툴은 코로나 19 사태 이후 비대면 수업 진행으로 인해 약화된 교사와 학생 사이의 상호작용을 개선하고자 개발한 교육 플랫폼입니다. 교사와 학생이 다양한 화상회의 플랫폼을 통해 비대면 수업을 진행하는 동안 “런업툴” 서비스를 사용하여 효과적으로 수업을 진행할 수 있습니다.  대표적으로 실시간 퀴즈, 출석체크 기능을 제공합니다. 교사는 수업시간 이전에 퀴즈를 생성할 수 있습니다. 수업시간에 교사가 퀴즈를 내면 학생은 실시간으로 제한시간 내에 해당 퀴즈에 답을 제출해야 합니다. 추후 학생은 정답과 풀이를, 교사는 학생의 퀴즈 제출여부를 확인할 수 있습니다. 출석체크 기능은 실시간으로 학생의 수업 출석여부를 구분할 수 있습니다. 교사가 출석체크를 시작하면 학생은 10분 이내에 확인 버튼을 눌러야 출석이 인정되고 10분이 넘어갈 시 자동으로 지각처리가 됩니다.  이와 같은 기능을 통해 효율적으로 학생을 관리하고, 원활한 실시간 쌍방향 교육이 이루어지는 데에 도움을 주고자 개발하였습니다. | | | | | | |
| **◇ 작품 접수 신청 시 유의사항 ◇**   * 공모한 SW, 아이디어는 타 공모전에 출품하여 수상한 경력이 없어야 하며, 타 공모전 수상 경력 확인 시 시상을 취소할 수 있다. (미수상작은 출품 가능) * 공모한 SW의 저작권은 대회 참가자에게 있다. * 공모한 SW가 타인의 저작권·지적재산권을 침해했을 경우, 추후 일어날 수 있는 모든 법적 책임은 대회 참가자에게 있다.   상기 본인은 작품 접수 신청서 제출에 있어 유의사항을 충분히 숙지하였으며, 대회 진행에 필요한 요구사항에 성실히 응할 것에 동의합니다.  2021년 11월 21 일  신청인 박예진 (서명)  (대표자) | | | | | | | | |

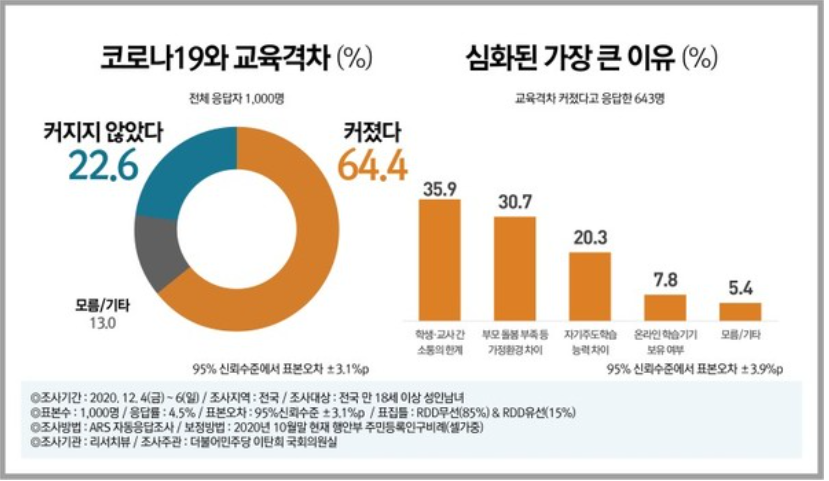
|  |  |  |
| --- | --- | --- |
| 별첨2 |  | 서비스 개발 상세 |

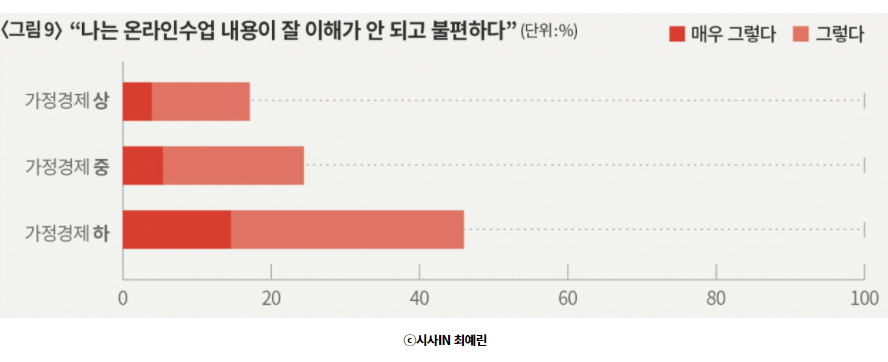
|  |  |
| --- | --- |
| **작품명** | 런업툴 |

□ 목 적 (또는 추진배경)

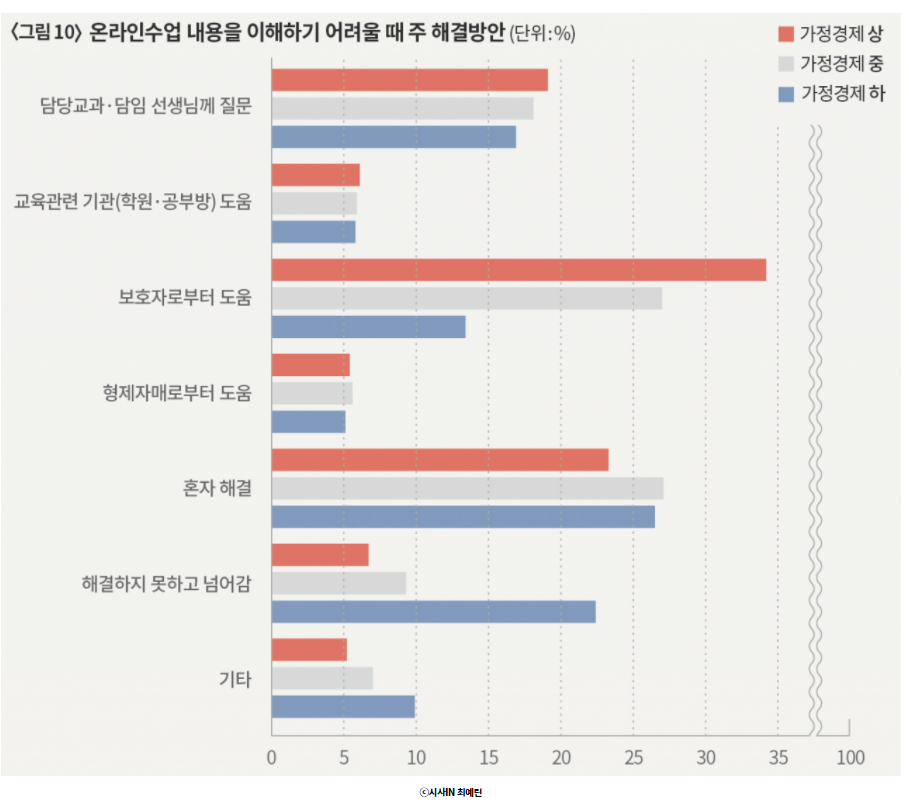
코로나 19 사태 이후 온라인 수업이 많이 시행되면서 많은 문제점이 발생하고 있습니다. “런업툴”이라는 교육 플랫폼을 구현하여 이를 개선하고자 합니다.

현재 비대면 온라인 수업 때문에 교사와 학생 모두 불편함을 겪고 있습니다. 실시간 온라인 수업에서 교사는 학생의 모습을 직접 확인하기 어려워 수업에 대한 학생별 집중력과 이해도를 파악하기 어려우며, 학생은 수업 참여에 한계가 있습니다.

****

위의 학습격차 심화의 설문조사 결과에서도 알 수 있듯이 온라인 수업 때문인 ‘학생·교사 간 소통의 한계’가 현재 온라인 교육 현실의 큰 문제점임을 알 수 있습니다.****

또한, 위의 통계자료를 통해 학생은 온라인 수업 내용을 완벽하게 이해하는 데에 어려움을 겪고 있다는 점을 알 수 있습니다. 특히, 가정형편이 어려운 학생은 온라인 수업의 불편함이 가장 큰 것으로 보입니다. 따라서 교육의 양극화가 더 악화할 우려의 목소리가 커지고 있습니다.

****

다음으로 학생은 교실이 아닌 혼자만의 장소에서 온라인 수업을 듣게 되어 오로지 수업에 집중하기 어려운 환경에 처하는 문제가 발생할 수 있습니다.

기존 화상회의 플랫폼이 제공하지 않는 교사와 학생을 위한 실시간 기능을 추가하여 위와 같은 한계를 극복하고, 효과적인 학생 관리 시스템을 제공하여 초, 중, 고등학생의 원활한 실시간 쌍방향 교육을 위한 플랫폼을 구축하는 것이 목표입니다.

**[출처]** [[소통광장-학습격차]③ 교사·학부모·학생, 코로나시대 교육을 말하다 < 소통광장 학습격차 < 사회 < 기사본문 - 뉴스포스트 (newspost.kr)](http://www.newspost.kr/news/articleView.html?idxno=92659), [적나라하게 드러난 팬데믹 시대 교육 불평등 - 시사IN (sisain.co.kr)](https://www.sisain.co.kr/news/articleView.html?idxno=42803)

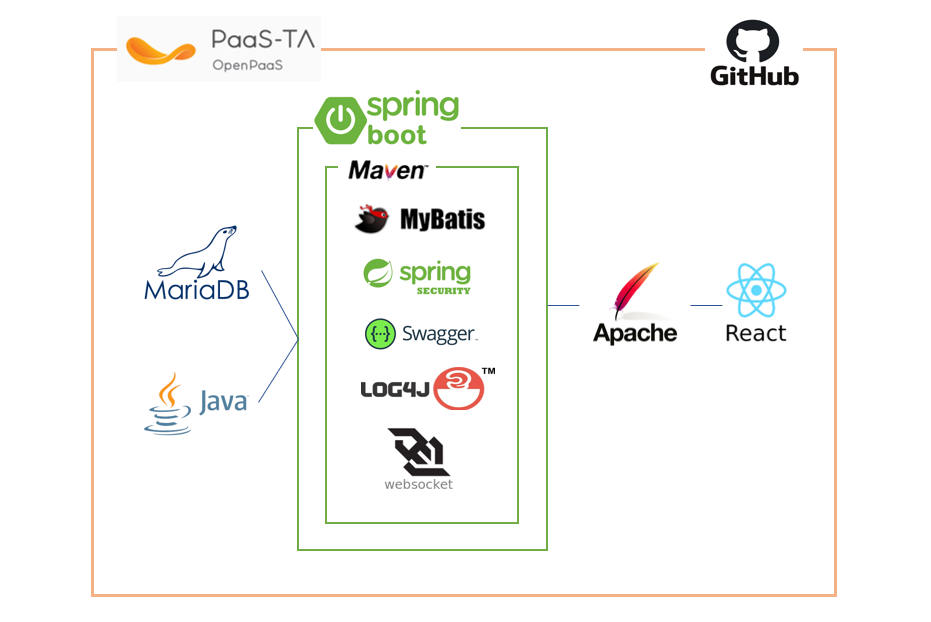
□ 개발 환경 구성

**사용언어: Java, Javascript, JSON**

**프레임워크: Spring boot, React**

**서버&DB: Apache tomcat, MariaDB**

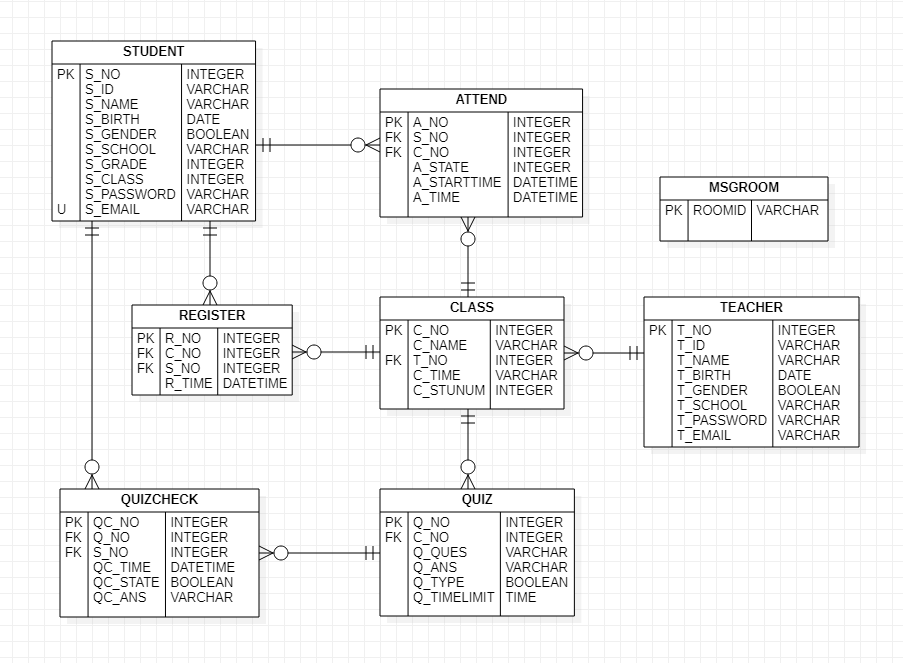
**실행환경: Paas-TA 5.5.1**

****

|  |
| --- |
| ※ 개발 환경 구성도 |

□ 내 용

* **DB 설계**



**<STUDENT> : 학생 사용자의 정보를 저장한다.**

S\_NO(PK-int) 학생번호

S\_ID(varchar) 학생아이디

S\_NAME(varchar) 학생이름

S\_BIRTH(date) 학생생년월일

S\_GENDER(boolean(0:남자 1:여자)) 학생성별

S\_SCHOOL(varchar) 학생학교명

S\_GRADE(int) 학생학년

S\_CLASS(int) 학생반

S\_PASSWORD(varchar) 학생비밀번호

S\_EMAIL(varchar) 학생이메일

**<TEACHER> 교사 사용자의 정보를 저장한다.**

T\_NO(PK-int) 선생님번호

T\_ID(varchar) 선생님아이디

T\_NAME(varchar) 선생님이름

T\_BIRTH(date) 선생님생년월일

T\_GENDER(boolean(0:남자 1:여자)) 선생님성별

T\_SCHOOL(varchar) 선생님학교명

T\_PASSWORD(varchar) 선생님비밀번호

T\_EMAIL(varchar) 선생님이메일

**<CLASS> : 수업의 정보를 저장한다.**

C\_NO(PK-int) 수업번호

C\_NAME(varchar) 수업이름

T\_NO(FK-int) 선생님번호

C\_TIME(varchar) 수업시간

C\_STUNUM(int) 수업참여학생수

**<REGISTER> : 수업의 등록 정보를 저장한다.**

R\_NO(PK-int) 등록번호

C\_NO(FK-int) 수업번호

S\_NO(FK-int) 학생번호

R\_TIME(datetime) 등록시간

**<ATTEND> : 수업의 출석 정보를 저장한다.**

A\_NO(PK-int) 출석번호

S\_NO(FK-int) 학생번호

C\_NO(FK-int) 수업번호

A\_STATE(int(0: 출석 1: 지각 2(default): 결석)) 학생출석현황

A\_STARTTIME(datetime) 출석시작시간

A\_TIME(datetime) 출석시간

**<QUIZ> : 퀴즈 정보를 저장한다.**

Q\_NO(PK-int) 퀴즈번호

C\_NO(FK-int) 수업번호

Q\_QUES(varchar) 퀴즈문제

Q\_ANS(varchar DEFAULT : ‘답 없음.’) 퀴즈정답

Q\_TYPE(boolean(0:단답형 1:서술형)) 퀴즈타입

Q\_TIMELIMIT(time) 퀴즈제한시간

**<QUIZCHECK> : 제출한 퀴즈의 정보를 저장한다.**

QC\_NO(PK-int) 퀴즈답안번호

Q\_NO(FK-int) 퀴즈번호

S\_NO(FK-int) 학생번호

QC\_STATE(boolean(0:false(default) 1:true)) 퀴즈제출여부

QC\_TIME(datetime) 퀴즈제출시간

QC\_ANS(varchar DEFAULT : ‘답 없음.’) 학생제출답안

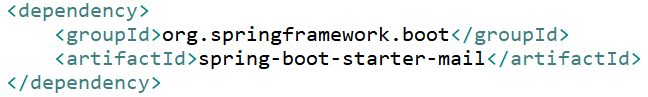
**<MSGROOM> : 실시간 수업 정보를 저장한다.**

ROOMID(varchar) 실시간수업아이디

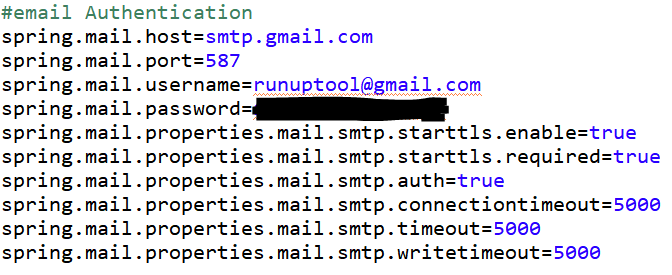
1. **이메일 발송 기능**

교사가 수업에 학생을 초대(등록)하기 위해 초대 링크를 포함한 이메일을 발송한다. 초대 이메일을 수신한 학생 사용자는 초대 링크로 접속하여 로그인을 완료하면, 수업 등록이 완료된다.

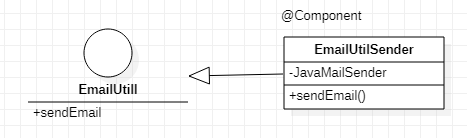
* **Spring-boot-starter-mail 의존성 추가**

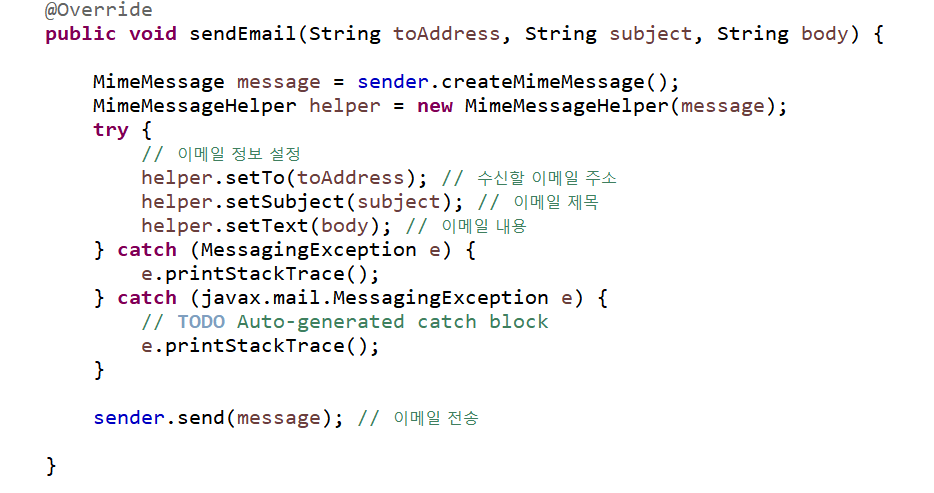
****

* **Gmail SMTP Server 사용 설정 : application.properties에 설정 추가**

****

* **EmailSender Service 구현**

****

****

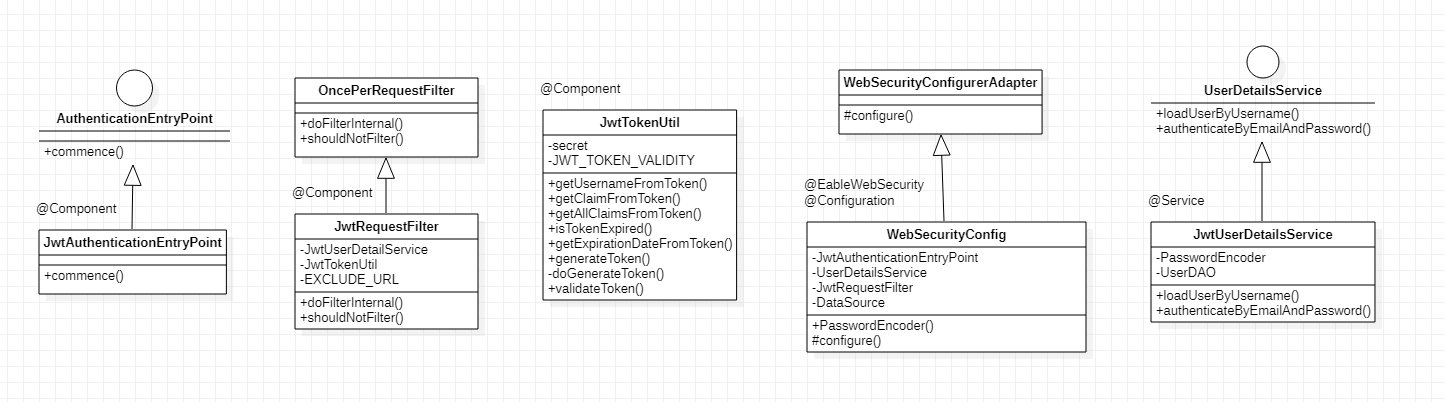
* **EmailSender의 사용**

****

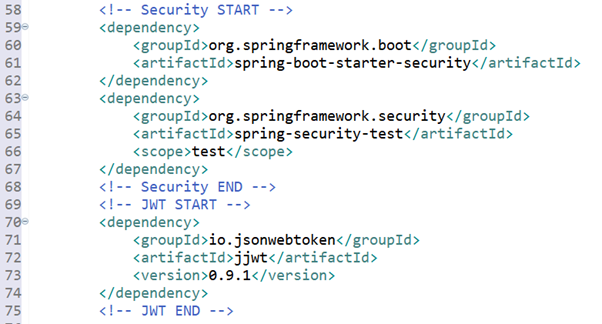
ClassController(@RestController)에서 EmailUtilSender의 sendEmail 메소드를 사용하여 이메일 발송한다**.**

1. **Spring Security+JWT**

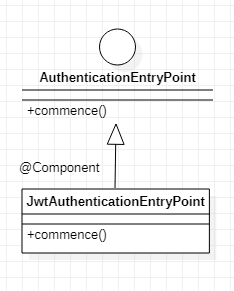
Filter를 통해 인증과 권한 등의 보안 관련 로직을 처리하기 위해서 사용한다.

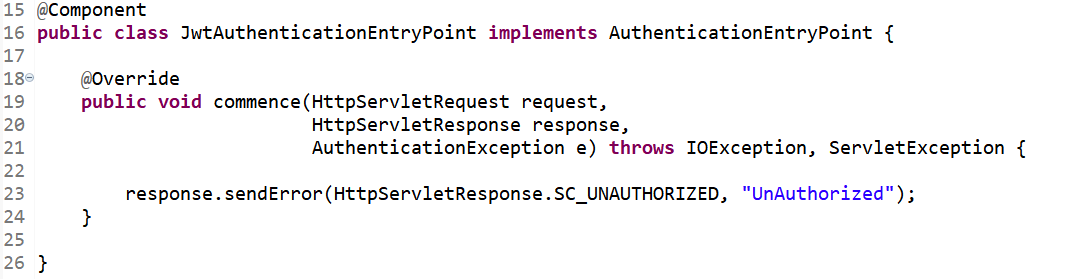
****

* **Spring Security와 JWT 의존성 추가**

****

* **JwtAuthenticationEntryPoint**

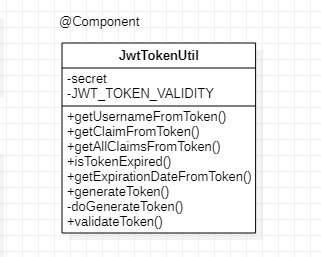
****

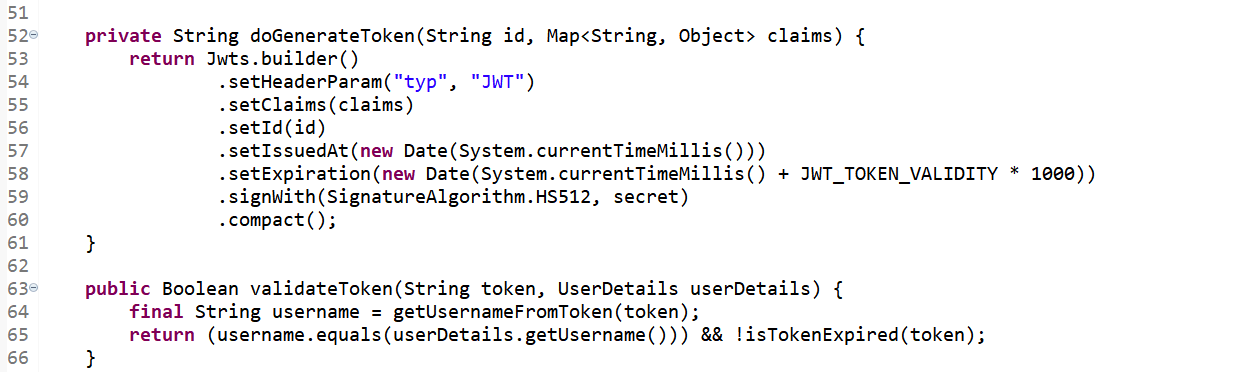
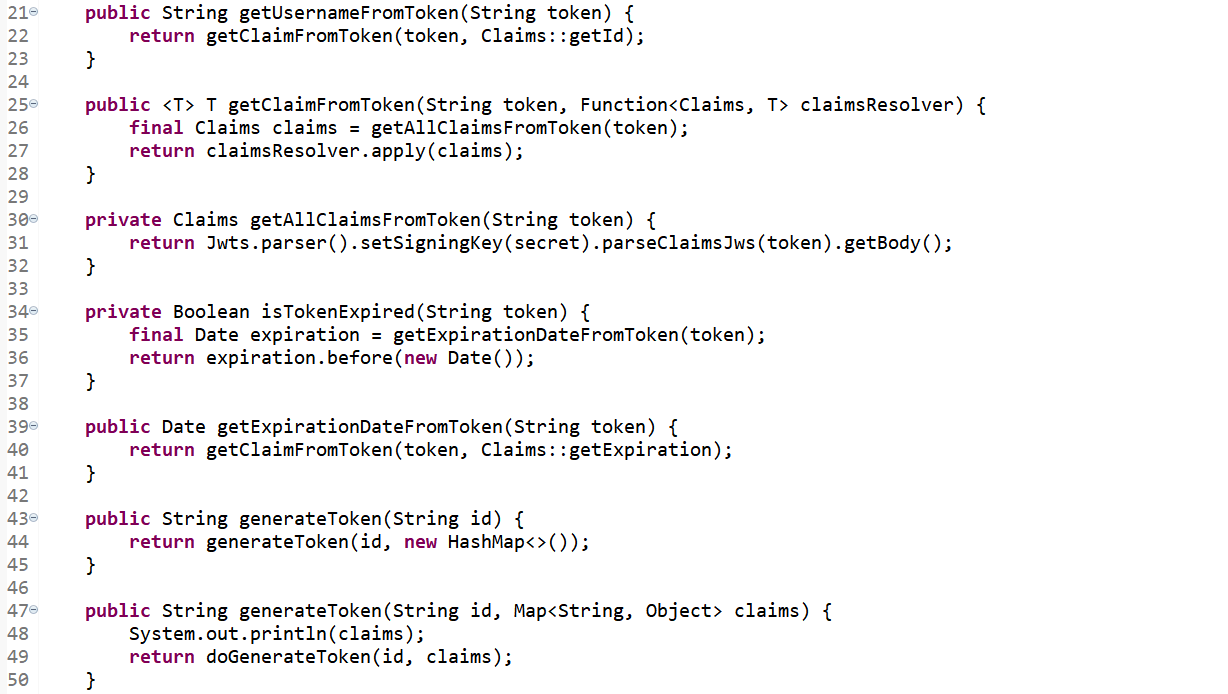
****

AuthenticationEntryPoint를 구현하여 인증에 실패한

사용자의 response에 HttpServletResponse.SC\_UNAUTHORIZED를 담아주도록 구현한다.

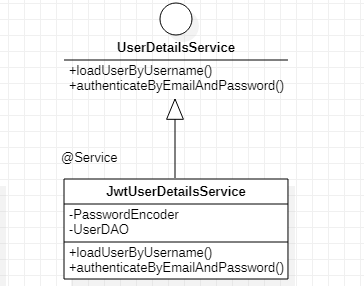
* **JwtTokenUtil**

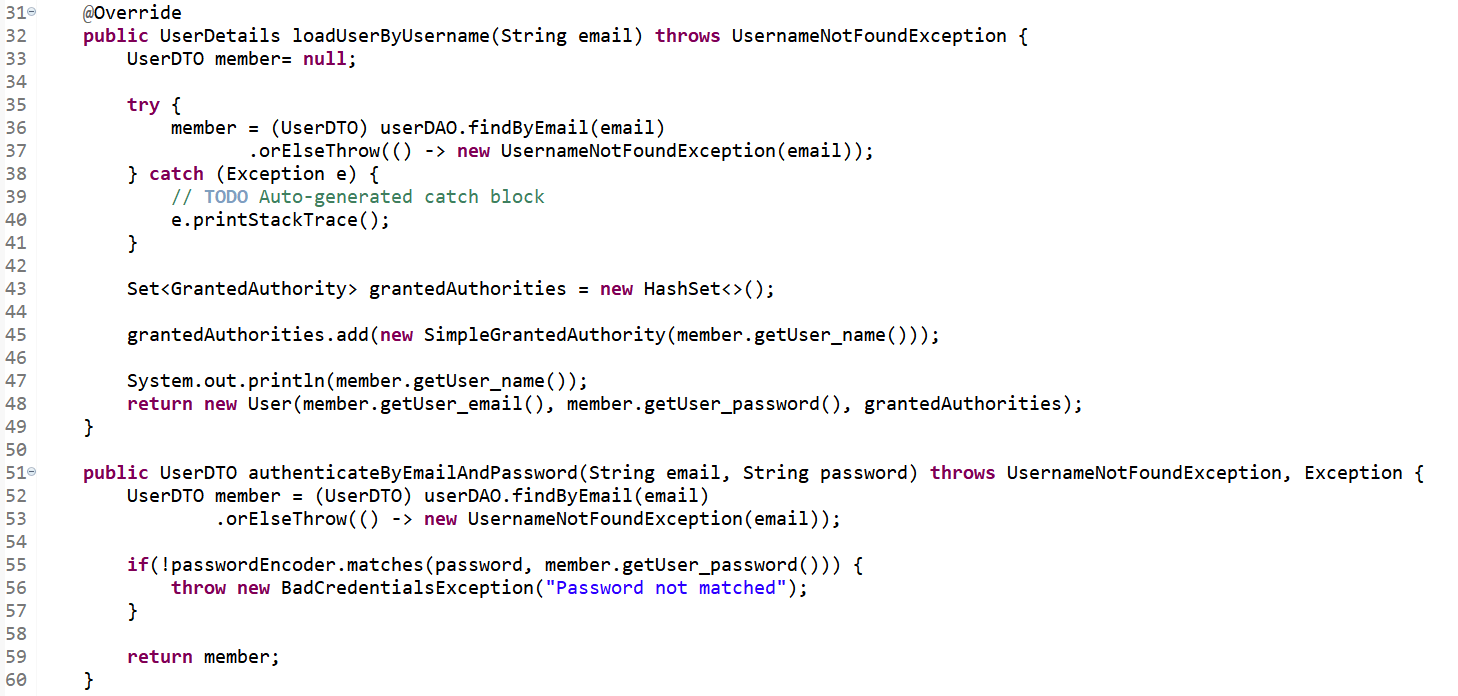
****

****

token을 발급하고 token에서 username을 추출하고 token의 유효성 검사 처리를 한다.

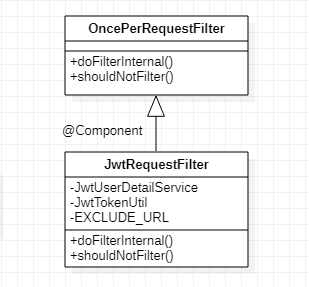
* **JwtUserDetailsService**

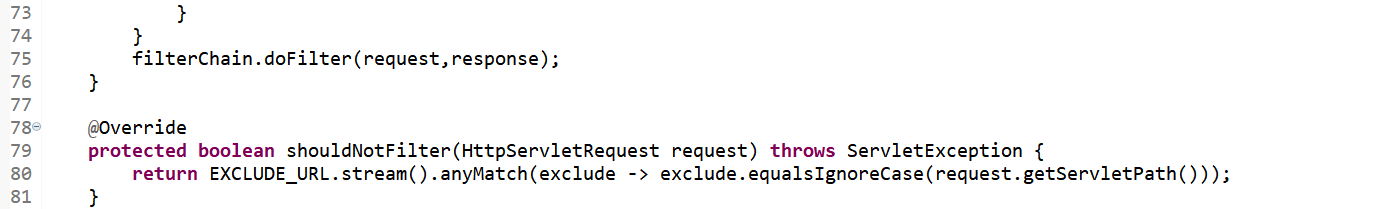
****



JwtUserDetailsService는 들어온 email로 Member를 찾아서 결과적으로 User 객체를 반환한다. 컨트롤러에서 넘어온 email과 password 값이 DB에 저장된 비밀번호와 일치하는지 검사한다.

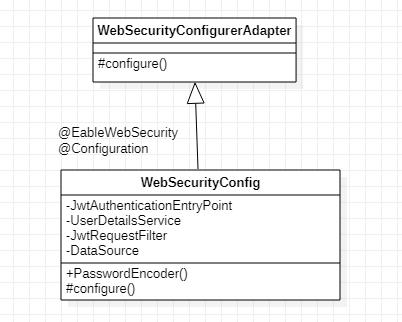
* **JwtRequestFilter**

****

****

doFilterInternal() 메서드를 통해, 요청할 때마다 filter가 한 번씩 수행되도록 구현한다. 헤더에서 Authorization 값을 꺼내서 토큰을 검사하고 해당 사용자가 실제 DB에 있는지 검사하는 등의 전반적인 인증처리를 진행한다.

* **WebSecurityConfig**

****

텍스트이(가) 표시된 사진

자동 생성된 설명

@Override protected void configure(HttpSecurity http) throws Exception {} : 보안 처리

@Override public void configure(WebSecurity web) throws Exception {} : 보안 예외 처리(정적 리소스, HTML 파일 허용 처리)

인증하지 않을 요청을 작성하고, 이 외에 다른 모든 요청은 인증을 수행한다. 또한, Spring Security에서 Session을 생성하거나 사용하지 않도록 설정하고, 모든 요청에 토큰을 검증하는 필터를 추가한다.

1. **Swagger**

개발과 동시에 API문서를 명세화하기 위해서 사용한다.

**- Swagger 의존성 추가**

텍스트이(가) 표시된 사진

자동 생성된 설명

**- application.properties 에 Swagger 설정 추가**



**- SwaggerConfig**

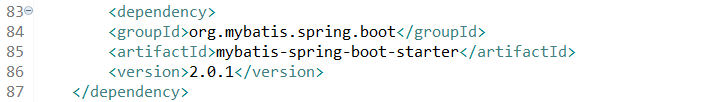
텍스트이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷이(가) 표시된 사진

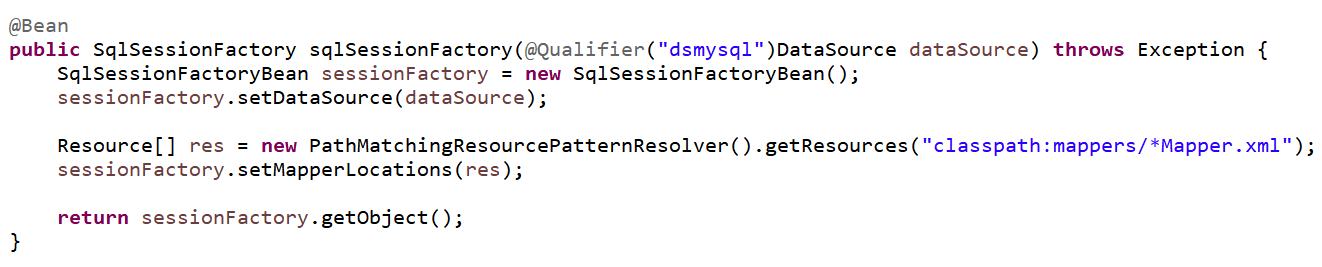
자동 생성된 설명

1. **Mybatis**

**- Mybatis 의존성 추가**

****

**- SqlSessionFactory**

****

SqlSessionFactory 객체를 통해 Mysql 서버와 MyBatis를 연결

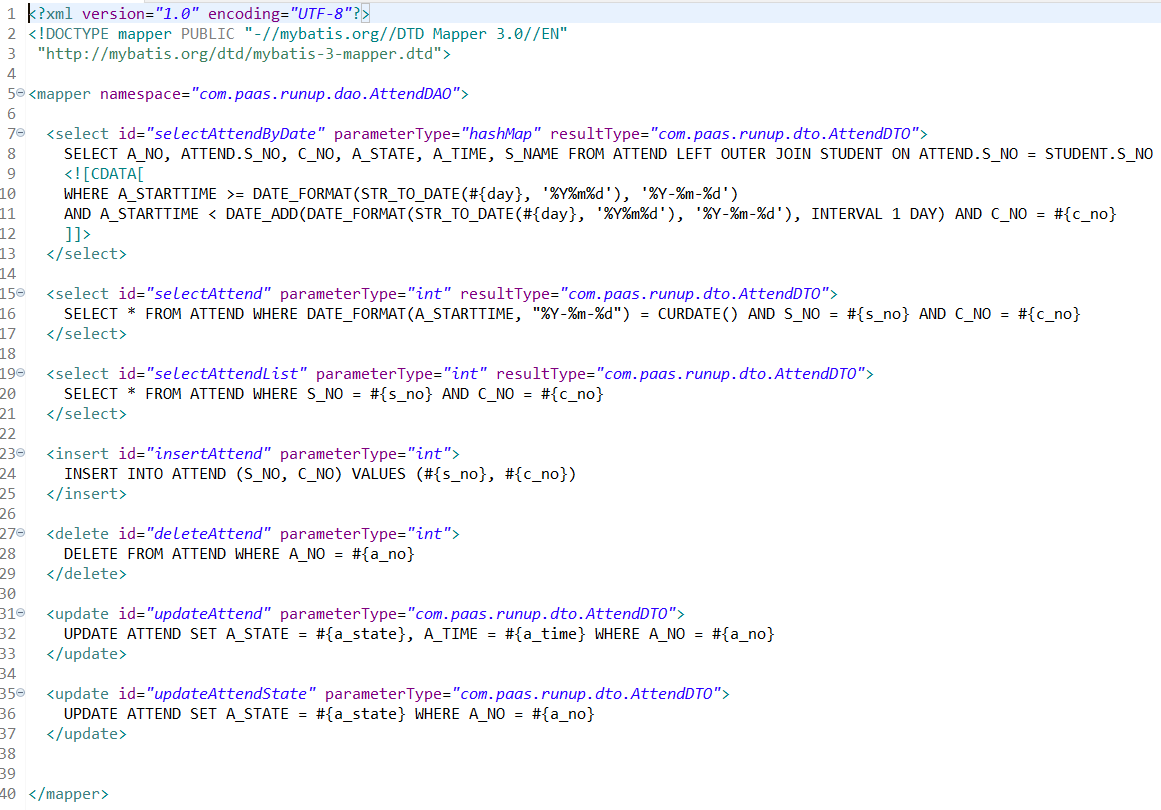
@Bean public SqlSessionFactory sqlSessionFactory(@Qualifier("dsmysql")DataSource dataSource) {} :SqlSessionFactory 객체 생성 함수

**- mybatis-config.xml**

****

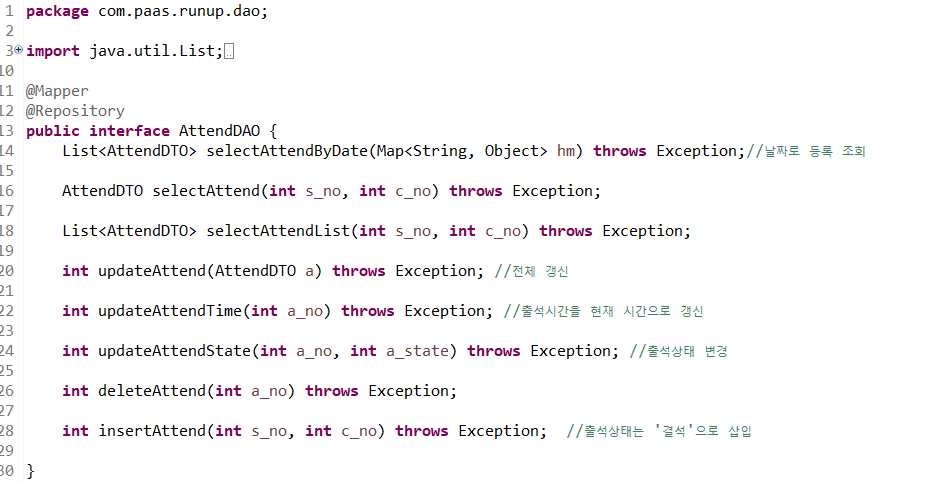
모든 Mybatis의 설정 기능을 활용할 수 있도록 src/main/resources에 mybatis-config.xml 파일을 추가

AttendMapper, ClassMapper, MsgRoomMapper, QuizCheckMapper, QuizMapper, RegisterMapper, StudentMapper, TeacherMapper, UserMapper

****

학생, 선생님, 출석부, 수업, 퀴즈 관련 xml 파일 생성

AttendDAO, ClassDAO, MsyRoomDAO, QuizCheckDAO, QUIZDAO, RegisterDAO, StudentDAO, TeacherDAO, USerDAO

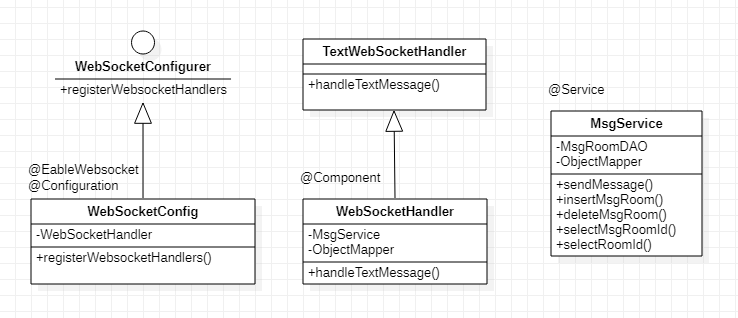
****

각각의 Mapper 파일을 DAO 파일로 연결

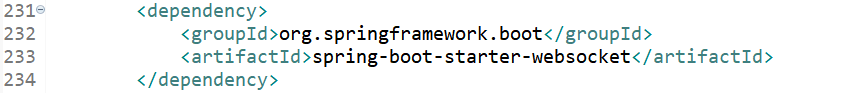
학생, 선생님, 출석부, 수업, 퀴즈 데이터를 검색, 추가, 갱신 및 삭제를 하고, 그 결과를 매핑한다.

1. **WebSocket**

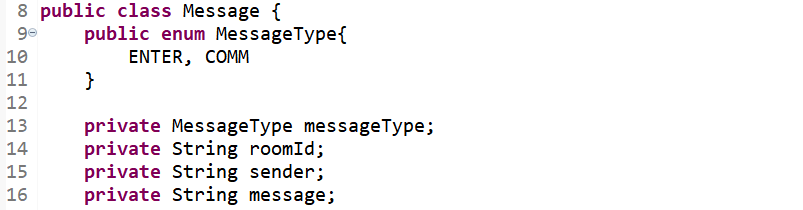
**온라인 수업 중 선생님과 학생이 실시간 통신을 할 수 있도록 WebSocket을 사용.**

****

* **WebSocket 의존성 추가**

****

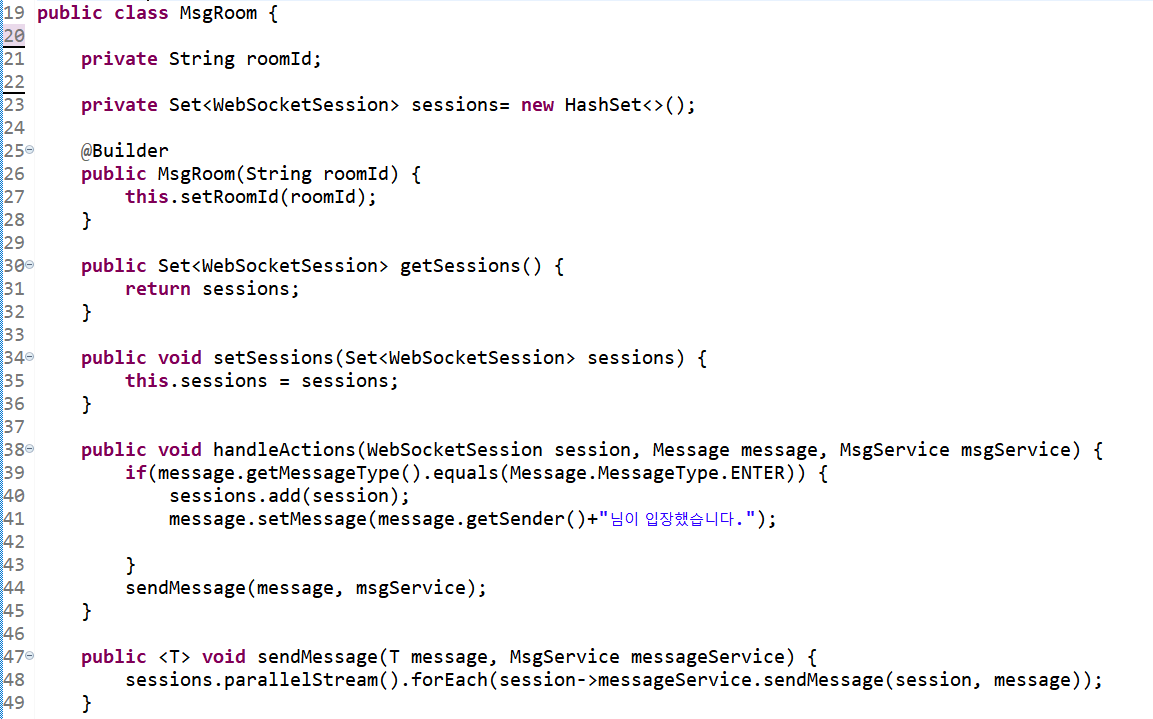
* **Message**

****

room에서 메시지를 주고받기 위한 DTO

room 입장, room에 메시지 보내기 두 가지 상황에 맞춰서 enum을 구현

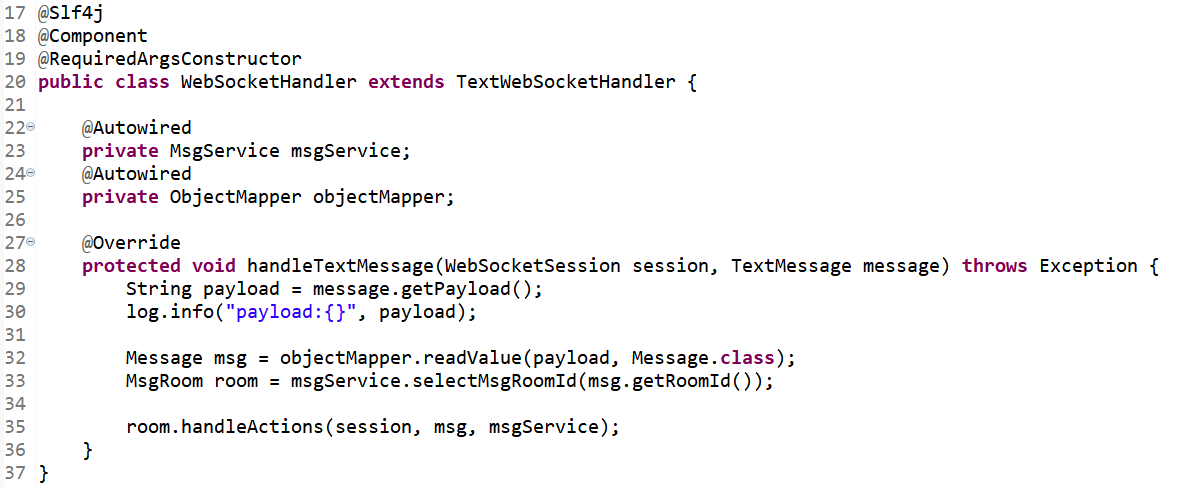
* **MsgRoom**

****room은 현재 방에 입장한 클라이언트의 세션 정보를 가져야 한다.

handleAction을 통해 입장, 통신 기능을 분기 처리한다.

입장 시에는 room의 세션 리스트에 클라이언트의 세션을 추가하고, 통신할 경우 room의 모든 세션에 메시지를 발송한다.

* **WebSocketHandler**

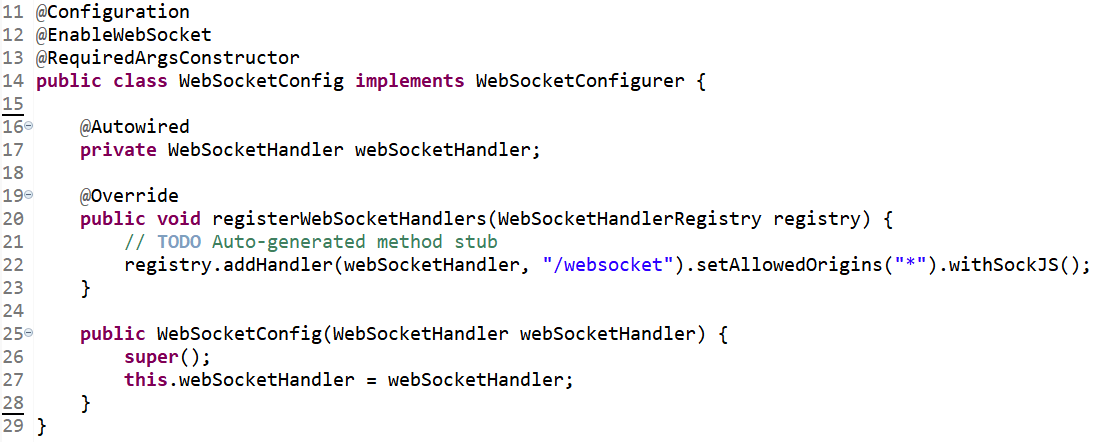
****

웹 소켓 클라이언트로부터 메시지를 전달받아 Message 객체로 변환한다.

전달받은 Message 객체에 담긴 roomId로 발송 대상 room 정보를 조회한다.

해당 room에 입장해 있는 모든 클라이언트에게 실시간으로 전송

* **WebSocketConfig**

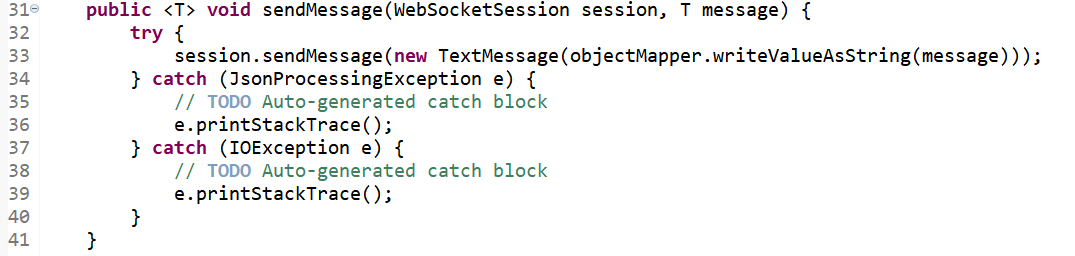
****

@EnableWebSocket을 선언하여 WebSocket을 활성화한다.

WebSocket에 접속하기 위한 endpoint는 “/websocket”으로 설정한다.

도메인이 다른 서버에서도 접속할 수 있도록 setAllowedOrigins(“\*”)를 추가한다.

* **MsgService**

****

지정한 WebSocket 세션에 메시지를 발송한다.

1. **클라이언트 화면 구성 및 설명 작성**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

App.jsx에서 불러온 메인 페이지들은 ‘/teacher, /login’ 등의 세부 페이지들과 각각의 component를 연결한다.

**<교사, 학생 공통 플랫폼>**

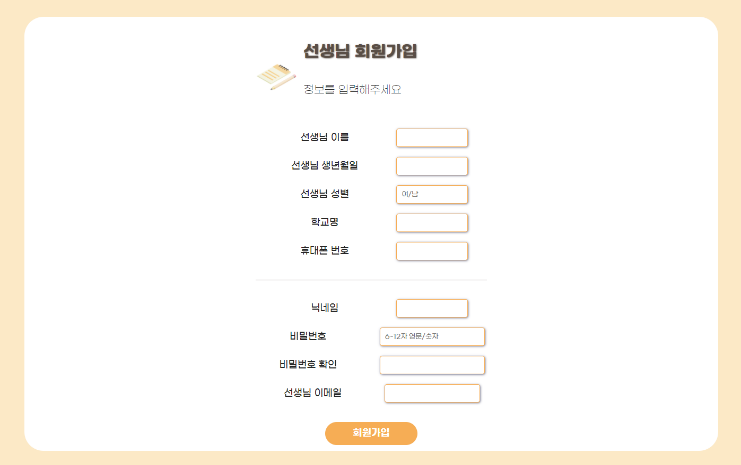
**- 로그인 화면**

****

로그인 화면은 AuthInput Component를 따로 만들어 학생, 선생님 로그인에 사용하였다. 학생과 선생님 따로 서비스 구현이 되어있기 때문에 두 부분으로 나누어 구성하였다.

**- 회원가입**

****

****

‘회원가입’을 들어가면 학생으로 회원가입 할 건지 선생님으로 회원가입을 할 건지 선택할 수 있도록 한다. 예시로 선생님 회원가입 화면을 보면 이름과 번호, 초등학교 이메일 비밀번호 등을 입력할 수 있게 한다.

**- 마이페이지**

****

사용자의 정보를 확인하고 수정할 수 있다.

텍스트를 클릭하면 입력 칸이 생기면서 수정할 수 있다. 수정하고 난 뒤에는 그 데이터를 받는 API와 연결되어 전달될 수 있도록 했다.

**<교사 관리 플랫폼>**

**- 교사 수업 리스트 조회 화면**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

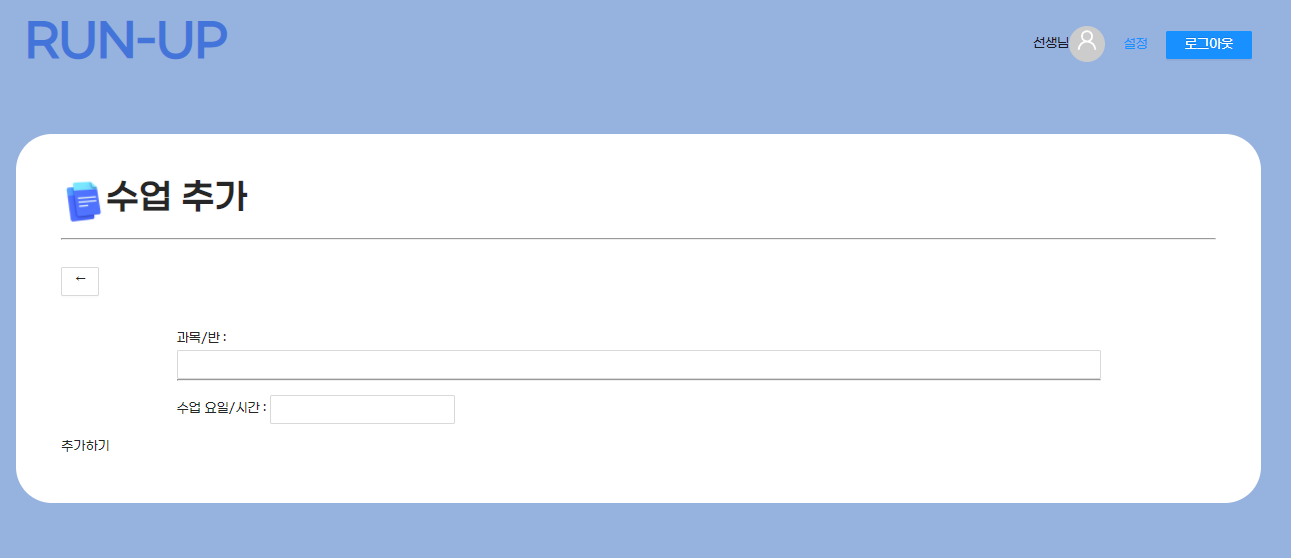
교사는 전체 수업 목록을 조회할 수 있다. 수업 이름, 반 이름, 수업 요일 및 시간, 학생 수를 확인할 수 있다. 수업 추가 화면으로 이동할 수 있다.

**텍스트이(가) 표시된 사진

자동 생성된 설명**

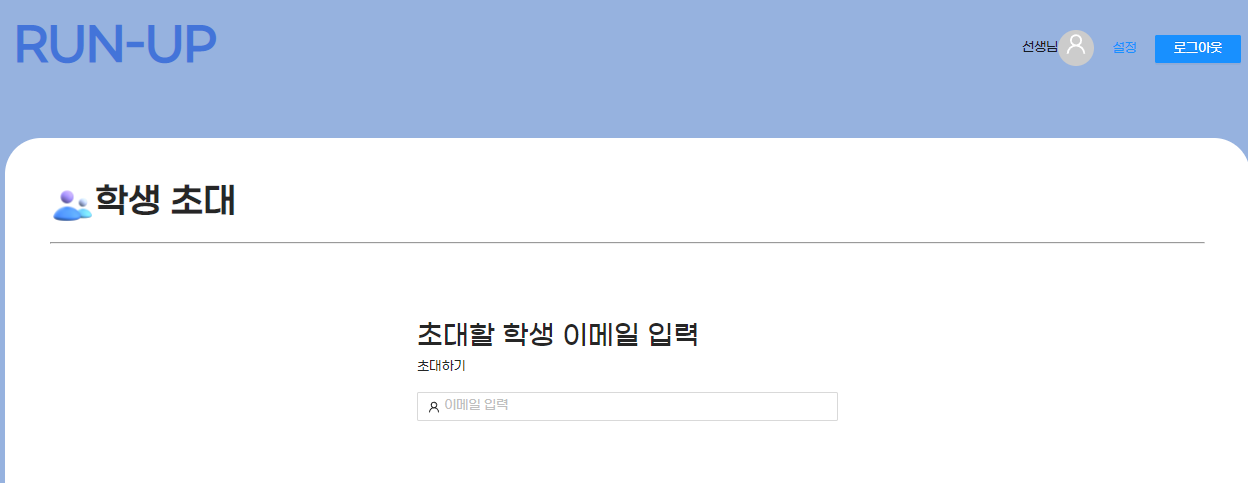
antd 플러그인의 Table과 그 안의 Column data를 어떻게 표현할지 구분해놓기 위해서 반, 수업명, 시간, 요일을 switch 문을 통해서 수정할 수 있도록 하였다.

**- 교사 수업 추가 화면**

****

수업 추가 화면에서는 수업의 과목, 반을 함께 입력할 수 있는 Input 칸과 수업 요일/시간을 입력할 수 있는 입력 칸이 있다.

**- 교사 학생 초대 화면**

****

교사는 학생 이메일을 작성한 후 해당 학생을 수업에 초대할 수 있다.  클라이언트 측에서는 수업번호와 메일, 그리고 학생이 들어와야 하는 초대링크가 존재한다. 초대 메일을 입력하게 되면 api에서 메일을 받아 서버내에서 문구를 추가해 입력한 메일로 초대메일이 전송된다.

**- 교사 출결 조회 및 수정 화면**

**테이블이(가) 표시된 사진

자동 생성된 설명**

교사는 수업 별 학생의 출결 상태를 확인할 수 있다.

출결 상태는 출석, 지각, 결석으로 구분한다. 학생의 출결 상태를 수정할 수 있다.

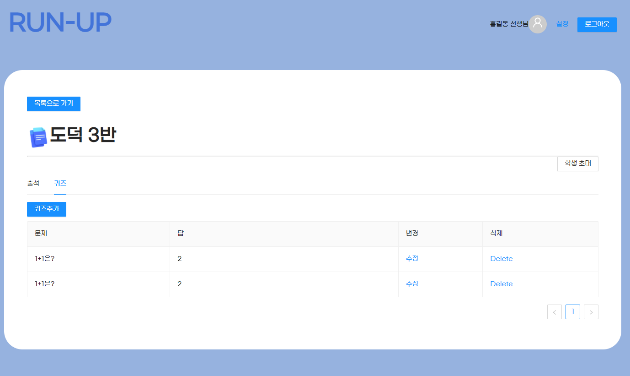
**텍스트이(가) 표시된 사진

자동 생성된 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

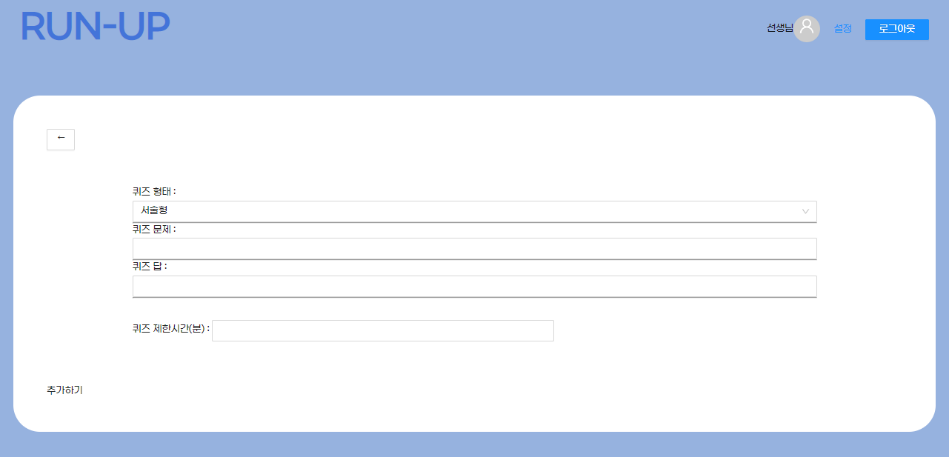
**- 교사 퀴즈목록 및 상세 화면**

****

교사는 수업 별 퀴즈 목록을 조회할 수 있다.

하나의 퀴즈를 선택하면 학생별 퀴즈 제출 여부를 확인할 수 있다.

**- 교사 퀴즈추가 화면**

****

교사는 단답형 및 서술형 구분 체크박스를 선택하고, 문제와 답, 제한시간을 설정하여 퀴즈를 추가할 수 있다.

**<학생 관리 플랫폼>**

**- 학생 수업목록 조회 화면**

****

학생은 수강한 수업 목록을 조회하여 수업 정보를 확인할 수 있고, 해당 수업의 출결과 퀴즈 확인 화면으로 이동할 수 있다.

**텍스트이(가) 표시된 사진

자동 생성된 설명**

Table 안에 Column 태그를 이용해서 수업 이름과 시간을 띄우게 되고 그 수업의 내용, 퀴즈 내용, 출석할 수 있는 버튼을 배치했다.

**- 학생 출결 조회 화면**

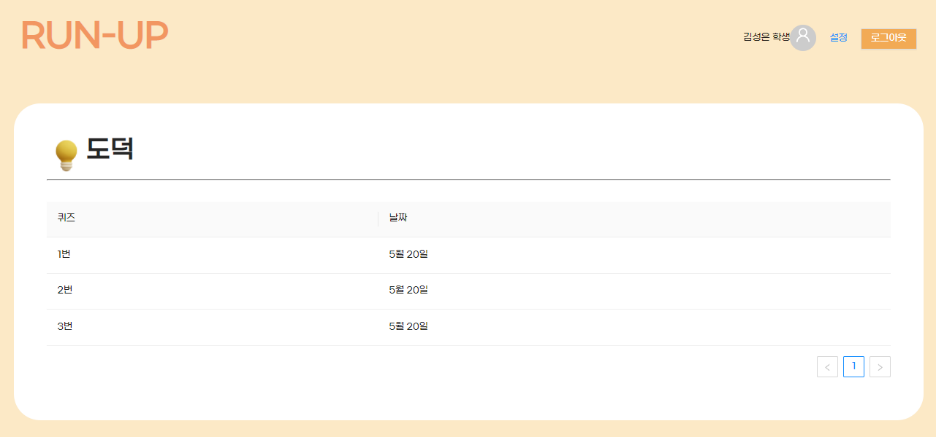
**텍스트이(가) 표시된 사진

자동 생성된 설명**

학생은 해당 수업의 본인 출결 상태를 확인할 수 있다.

수업 리스트에서 수업 내용 확인을 눌러 들어가게 되면 확인할 수 있다. 이 부분도 API를 호출하여 학생이 들어가 있는 수업에 출석했는지 데이터로 받아 확인할 수 있게 하였다.

**- 학생 퀴즈 목록 화면**

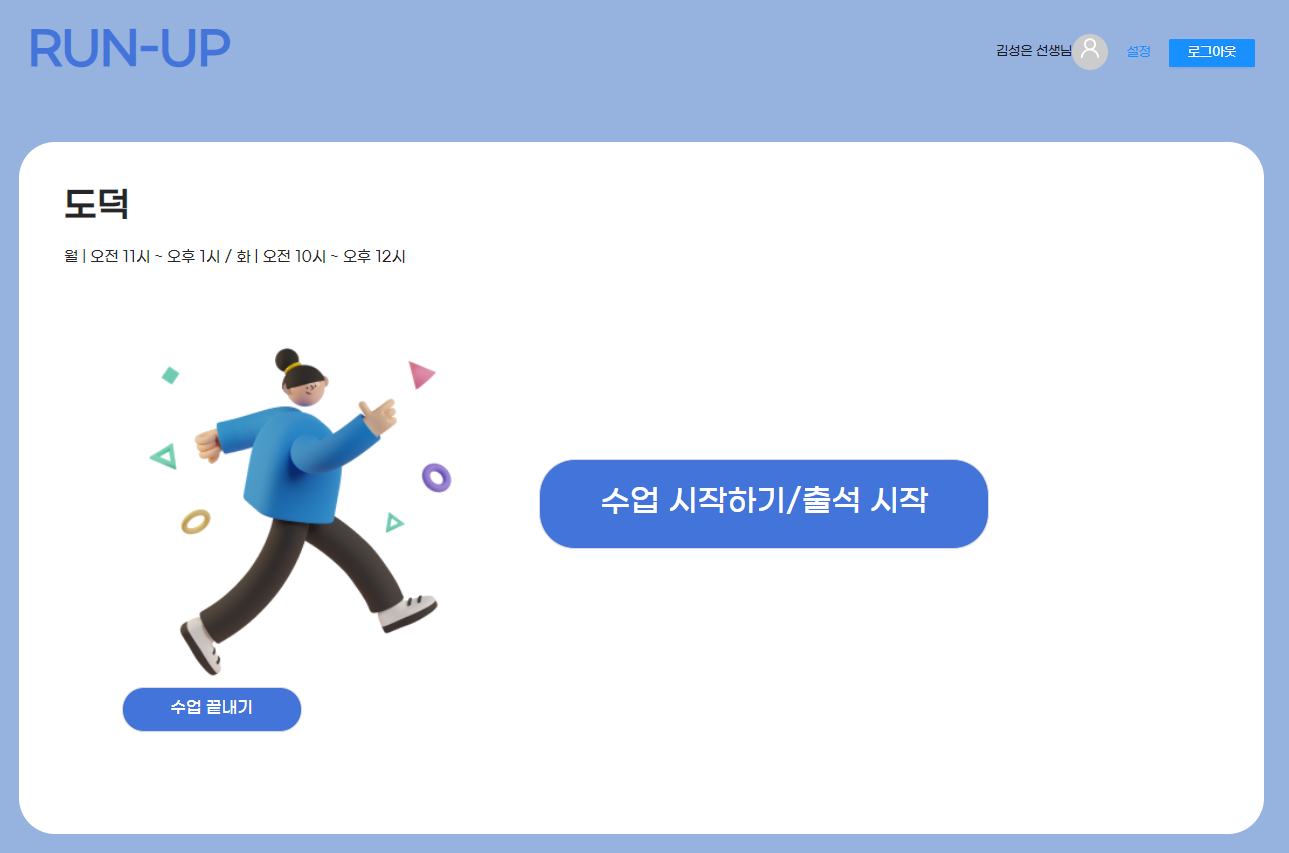
****

학생은 해당 수업에서 본인이 제출한 퀴즈 목록을 확인할 수 있다.

수업 리스트에서 퀴즈 내용 확인을 눌러 들어가게 되면 확인할 수 있다. 이 부분도 API를 호출하여 학생이 퀴즈를 제출했는지 데이터로 받아 확인할 수 있게 하였다.

<교사 실시간 플랫폼>

- 교사 실시간 도구창 화면

****

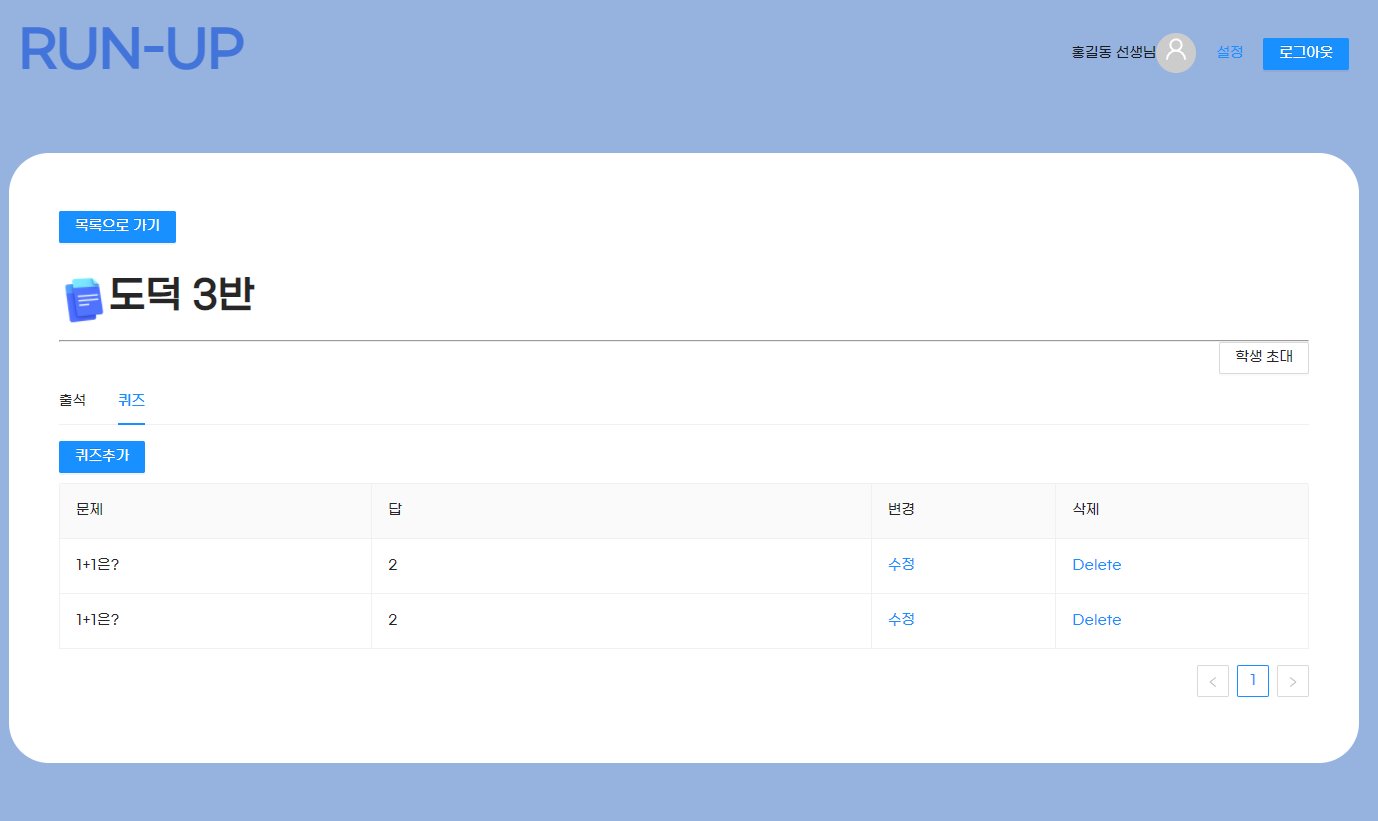
교사는 실시간 도구창 화면을 통해 출석 시작과 퀴즈 시작을 할 수 있다. 출석 시작 버튼을 누르면, 접속된 학생들에게 출석 시작 알림이 보내지고, 퀴즈 시작 버튼을 누르면, 퀴즈 목록 조회 화면으로 이동한다.

- 교사 실시간 출결 조회 화면

****

교사는 해당 수업의 출결 현황을 조회할 수 있다.

- 교사 실시간 퀴즈 리스트 조회 화면

****

교사가 이전에 만들어 놓은 퀴즈 리스트에서 퀴즈를 선택하여 학생들에게 실시간으로 퀴즈를 낼 수 있다.

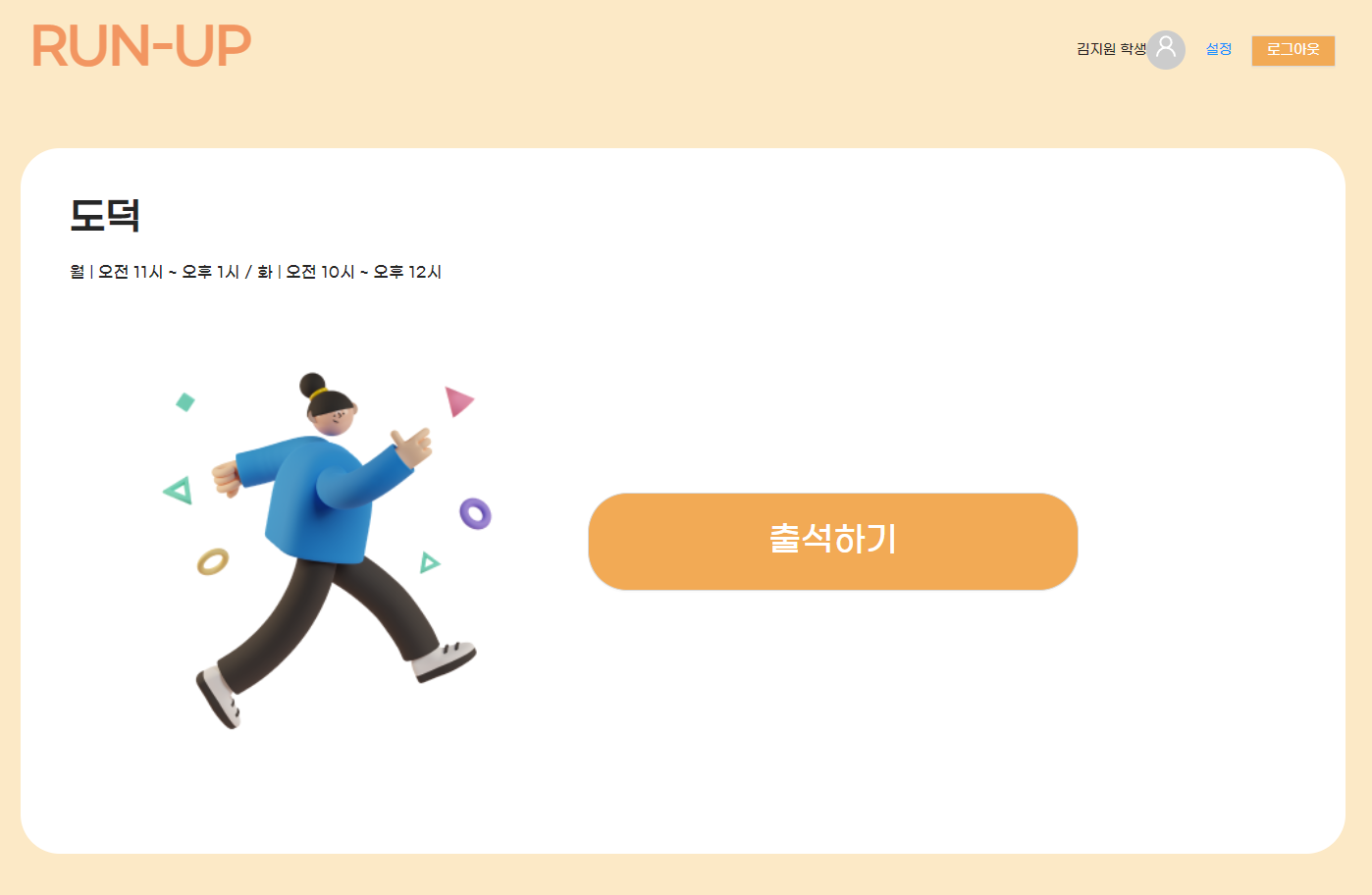
- 교사 실시간 퀴즈 제출 여부 화면

****

학생들의 퀴즈 제출 여부를 실시간으로 확인할 수 있다.

**<학생 실시간 플랫폼>**

**- 학생 실시간 도구창 화면**

****

학생은 실시간 도구창 화면을 통해 출석과 퀴즈 풀이를 할 수 있다.

**- 학생 실시간 출석 화면**

**테이블이(가) 표시된 사진

자동 생성된 설명**

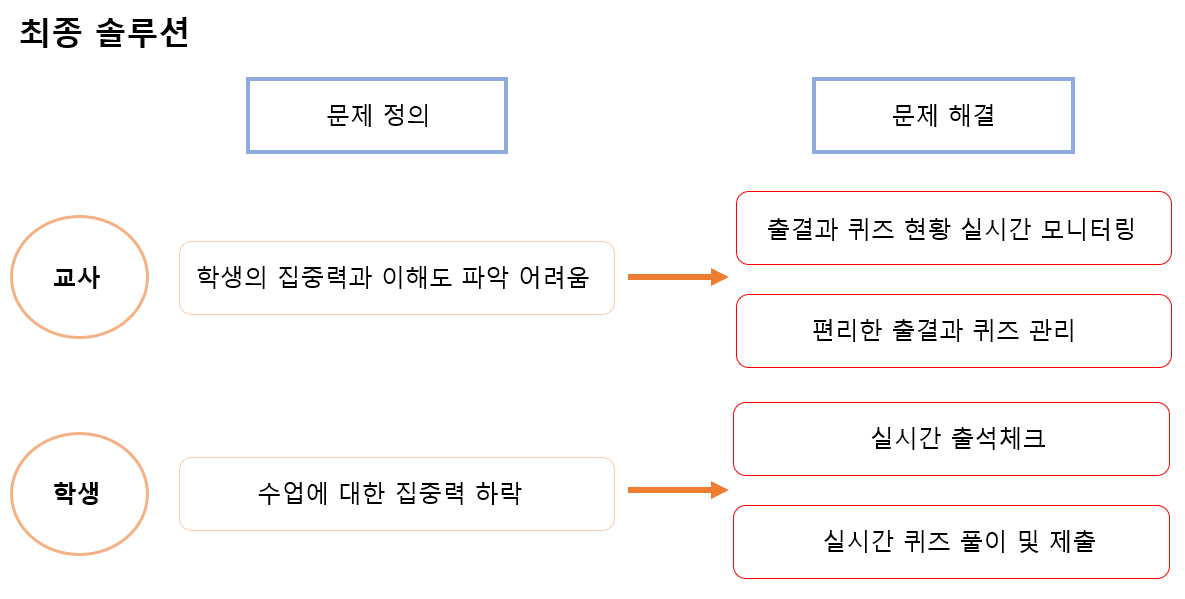
교사가 출석을 시작하면 학생의 출석하기 버튼이 활성화된다. 이 부분에서 WebSocket이 쓰였는데 학생이 출석하기 버튼을 눌렀을 때 API 호출을 하여 제시간에 맞게 들어왔는지 데이터를 전송하고 서버에서 구분하여 선생님 쪽에 출석인지 지각인지를 보내준다.

**- 학생 실시간 퀴즈 풀이 화면**

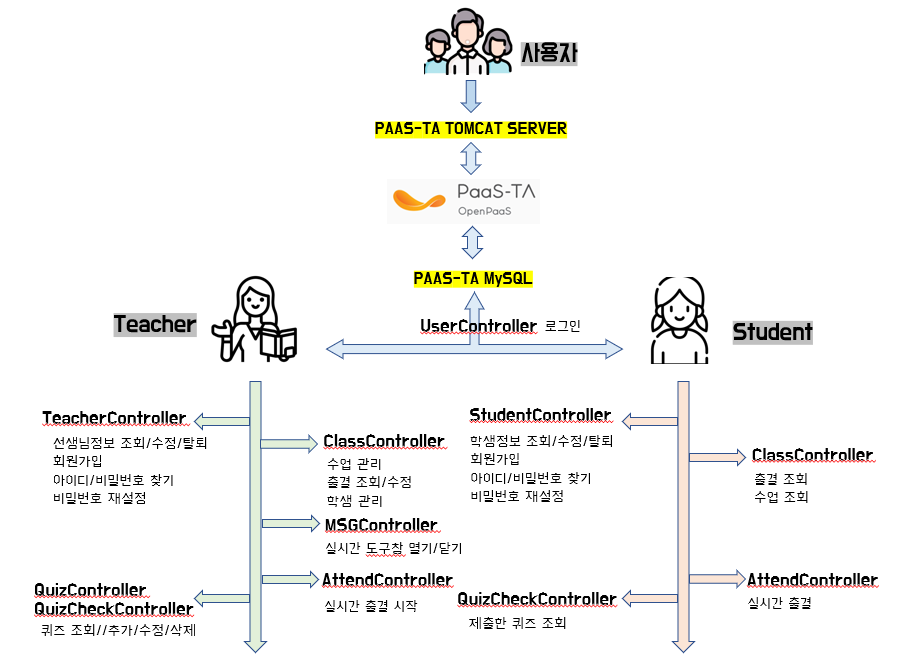
****

학생은 교사가 낸 퀴즈를 제한시간 내에 실시간으로 풀어서 제출한다.

**- 최종 솔루션**

****

**-서비스 구성도**

****

|  |
| --- |
| ※ 최종 솔루션 및 서비스 구성도 |

□ 기대효과

**- 교사의 효율적인 학생 관리**

출석 체크가 실시간 자동으로 수행되고, 학생들의 출석 여부와 학생들이 수업 중에 제출한 퀴즈 목록이 기록됨으로써 교사가 학생들의 출석과 수업 이해도를 확인하며 학생들을 효율적으로 관리할 수 있습니다.

**- 학생의 수업 참여도 증가**

수업 중에 실시간으로 퀴즈를 풀고 제출함으로써 비대면 수업의 가장 큰 문제인 학생 수업 참여의 부재를 개선할 수 있습니다. 학생들의 수업 참여도 증가는 적극적인 학습 분위기를 형성하고 결과적으로 학습 능력 향상에 도움이 됩니다.

**- 원활한 실시간 쌍방향 교육**

교사가 출석과 퀴즈 현황을 실시간 모니터링 함으로써 학생의 수업 참여도 평가와 원활한 수업 운영이 가능합니다. 교사와 학생의 원활한 실시간 쌍방향 교육이 이루어져 온라인 교육의 질이 향상됩니다.

**- 화상회의 플랫폼에 제약받지 않는 편리한 사용**

화상회의 플랫폼의 종류와 상관없이 독립적으로 사용 가능 합니다. 교사의 선호 및 수업에 따라 서로 다른 화상회의 플랫폼을 사용하여도, 하나의 플랫폼으로 학생 및 수업 관리가 가능하여 편리합니다.

**- 수업 내용 외 시간 절약**

출석 체크할 때 교사가 학생을 일일이 호명할 필요 없이 출석체크가 이뤄지고, 교사는 수업 주제와 맞는 퀴즈를 미리 준비할 수 있어 수업 내용과 상관없는 시간을 절약합니다.