

요구사항 정의서

팀명	1팀
팀원	김진원, 윤문현, 정우진, 김초롱

구분	NO	권한	요구사항ID	요구사항명	담당	기능명	요구사항	요구사항 상세	구현방안
	1	공통	LOGIN	로그인	정우진	로그인	* 올바른 ID와 PW를 입력하여 로그인할 수 있다. * 카카오, 네이버 계정을 연동하여 로그인을 할 수 있다.	* ID와 PW를 입력하여 로그인을 할 수 있다. 로그인 성공 시 메인 화면으로 이동되게 된다. * 카카오 및 네이버 계정을 연동하여 로그인을 할 수 있다. 로그인 성공 시 메인 화면으로 이동되게 된다.	* input 태그를 사용하여 사용자가 입력한 값을 처리한다. * 로그인 요청 시 JWT 토큰을 생성, 검증 및 접근 허용한다. * 입력한 정보가 DB에 저장되어 있는 ID, PW와 동일하다면 로그인한다. * ID와 PW가 일치하지 않을 경우, 오류창 노출 및 로그인 화면이 유지된다. * checkbox를 사용하여 ID저장 및 자동 로그인 기능을 구현한다. * Oauth2를 이용하여 Access token을 발급, DB에 저장되어 있는 기존 정보와 매칭하여 로그인 처리를 진행한다.
	2	공통	LOGOUT	로그아웃	정우진	로그아웃	* 로그아웃을 할 수 있다.	*사용자가 로그아웃 버튼을 누를 경우, 로그아웃 된다.	* 로그아웃 요청 시 저장된 JWT를 삭제하여 로그아웃 처리한다.
	3	공통	FIND ID/PW	ID/PW 찾기	정우진	ID/PW 찾기	* 아이디와 비밀번호를 찾을 수 있다	* 가입한 회원의 정보를 입력하여 아이디와 비밀번호를 찾을 수 있다.	* 가입한 정보의 이메일을 입력하여 자바스크립트를 사용, 아이디의 일부분을 *로 가려 노출 * starter-mail 라이브러리를 사용, 입력한 이메일로 임시 비밀번호 발송 * nurigo 라이브러리를 사용, 입력한 휴대전화로 임시 비밀번호 발송
	4	공통	JOIN	회원가입	정우진	회원가입	* 주어진 양식에 알맞는 정보를 입력하여 회원가입을 할 수 있다.	* 주어진 양식에 각 정보를 입력하여 회원가입을 할 수 있다. * 회원가입 성공 시 로그인 화면으로 이동되게 된다.	* input 태그를 사용하여 사용자가 입력한 값을 처리한다. * 올바른 정보를 입력 시 DB에 입력한 정보를 저장한다. * 주어진 양식에 부적합할 경우, 부적합한 정보의 input 태그로 focus된다. * Oauth2를 이용하여 사용자의 정보를 저장하고, 추가적인 정보를 input태그를 사용하여 사용자가 입력한 값을 DB에 저장한다.
	5	공통	MY PAGE	마이페이지	정우진	마이페이지	* 로그인한 사용자의 정보를 볼 수 있다.	* 로그인한 유저의 정보, 속해있는 채팅방, 구독권, 예약되어 있는 일정 등의 정보를 볼 수 있다. * 유저의 정보를 수정할 수 있다. * 회원 탈퇴를 할 수 있다.	* 토큰의 정보와 DB의 정보를 매칭하여 일치하는 사용자의 정보를 불러온다. * 사용자의 정보와 채팅방의 정보를 매칭하여 DB에서 불러온다. * 구독권의 정보를 가져오며, TOSS 결제 API를 이용하여 구독권 결제를 처리한다. * 예약되어 있는 일정을 볼 수 있다.(예약 파트와 연동) * input 태그를 사용하여 사용자의 정보를 수정할 수 있다.

회원	6	일반회원	SUBSCRIBE	구독	정우진	구독	* 일반 회원은 구독권을 구매하여 여러가지 기능들을 사용할 수 있다.	* 로그인 한 유저는 구독 페이지에서 구독권을 구매할 수 있다. * 구독 연장/취소를 할 수 있다. (환불은 불가능) * STOP 상태의 회원은 모든 기능을 사용할 수 없다.	* Toss 결제 API를 사용하여 실존하는 결제 정보를 입력해야 결제가 진행되도록 구현한다. * Toss API에서 요구하는 빌링키를 발급받아 스케줄러를 사용하여 매월 / 매년 정기결제가 되도록 구현한다. * 일반 회원은 가입 시 STOP 상태로 가입되며 구독권 구매 전까지 모든 기능을 사용 불가하다. * 정기결제 취소시 CANCEL 상태로 전환되며, 구독권 만료 시 STOP 상태로 전환된다. * 취소를 하지 않는 이상 자동으로 정기결제가 이루어지며, CANCEL 상태에서도 정기결제 전환이 가능하다.
	7	공통	FRIEND	친구	정우진	친구	* 회원 간 친구 추가를 하여 채팅을 시작할 수 있다.	* 회원 간 친구 추가를 통하여 친구를 등록할 수 있고, 채팅을 시작할 수 있다.	* 현재 가입된 회원의 리스트를 추천 친구 리스트에 노출시킨다. * 친구 요청을 보냄에 따라 친구 요청 테이블의 상태값에 변화를 준다. * 요청을 수신/발신한 기록은 각 리스트에 노출되며, 요청을 삭제, 수락할 수 있다. * 요청을 수락하면 친구 리스트에 노출되며, 채팅 시작 탭에서 친구와 채팅을 시작할 수 있다.
	8	관리자, 직원	MEMBER MANAGEMENT	회원 관리 페이지	정우진	회원 관리 페이지	* 회원들의 정보를 관리할 수 있다.	* 가입한 회원들의 정보를 수정 및 삭제할 수 있다. * 회원의 등급을 변경할 수 있다.	* 회원을 승인, 승인 대기, 결제 취소, 구독 만료 네가지로 분류하여 상태를 나타낸다. * 일반회원들은 모두 구독 만료 상태로 가입된다. * 트레이너 등급으로 가입하는 회원은 관리자의 승인 전까지 승인 대기 상태이며 일반회원 등급이다.
	9	공통	MESSAGE	채팅	정우진	채팅	* 회원 간 채팅을 할 수 있다.	* 회원 1:1, 1:N 간 채팅방을 생성하여 대화할 수 있다.	* WebSocket과 Redis Pub/Sub을 이용하여 채팅 기능을 구현한다. * 윈도우 함수를 사용하여 채팅방 생성 및 채팅창은 새로운 창으로 실행된다. * 1:N 채팅방의 경우 채팅방을 처음 개설한 회원이 방장이 된다. * 방장은 속해있는 채팅방의 회원을 강퇴할 수 있다. * 회원은 친구인 회원을 채팅방에 초대할 수 있다. * 접속 중인 회원과 그렇지 않은 회원은 다르게 노출된다. * 스케줄러를 사용하여 일정 시간마다 채팅방 생성 후 채팅이 한번도 이루어지지 않은 방은 자동 삭제된다.
		공통	CALENDAR	달력UI	김진원	달력UI 생성	달력 형태로 트레이너 스케줄을 보여 줄 수 있다.	* fullCalendar 라이브러리를 사용하여 달력을 생성한다	* fullCalendar 라이브러리를 사용하여 달력형태로 예약 가능 날짜를 보여준다.
		직원		일정관리(트레이너)	김진원	트레이너 일정	사용자(트레이너)는 본인의 스케줄을 확인 할 수 있다.	* 본인 계정으로만 본인 스케줄을 확인 및 관리 할 수 있다.	* 개인 전용 화면에서 FullCalendar로 본인 SSCHEDULE을 표시하고, 이벤트 클릭 시 상세 모달을 띄워준다. * 세션의 사용자명을 받아 해당 트레이너의 SSCHEDULE 레코드를 조회해 반환한다.
		관리자		일정관리(관리자)	김진원	관리자 일정 등록/수정/삭제 및 예약 통계 확인	관리자는 전체 일정과 통계를 확인 할 수 있어야 한다	* 관리자는 전체 일정에 대한 CRUD 및 통계 조회 기능을 사용할 수 있다.	* 관리자 전용 UI 생성 * 캘린더/리스트 뷰 전환 지원 * 엑셀(CSV) 내보내기·가져오기 기능 포함 * 관리자는 스케줄을 직접 등록 수정 삭제 할 수 있다

일정/예약		공통		일정 관리	김진원	예약된 일정 보기	트레이너가 예약(시설,시간)된 일정을 확인 할 수 있어야 한다.	* 예약된 일정을 달력 형태로 볼 수 있다.	* DB연동, FullCalendar 출력
		관리자		수정/취소 승인	김진원	수정/취소 승인	수정/취소 요청은 관리자 승인 후 확정 되어야 한다.	* 요청 -> 승인 -> 확정 흐름 확정되면 트레이너에게 메일 발송	* 서버에서는 요청 상태를 “요청 → 승인 → 확정”으로 업데이트한다. * 자바 메일 센터를 이용하여 이메일 전송
		관리자		스케줄 현황	김진원	스케줄 현황	관리자는 모든 스케줄 현황을 볼 수 있어야 한다	* 관리자는 회원들의 모든 스케줄 현황을 확인 할 수 있다.	* 리스트 또는 캘린더로 확인 가능하게 한다. (fullCalendar 라이브러리 사용해서) * 모든 SSCHEDULE 데이터를 조회해 반환한다. * 스케줄 통계 대시보드(시간대·일별)
		직원		비품 관리	김진원	비품 고장/신고	트레이너는 비품에 문제가 생겼을 시 신고 할 수 있다	* 트레이너는 비품에 문제가 생겼을 시 관리자에게 신고 할 수 있다.	* 비품이 고장 난것을 관리자에게 요청을 보낸다. *카테고리(헬스·복싱·수영) 필터 UI
		관리자		비품 관리	김진원	비품 고장/접수	관리자는 비품에 문제가 생긴것을 확인하고 처리할 수 있다.	* 관리자는 트레이너에게 요청 받은 것을 처리 할 수 있다.	* 비품이 고장 난것을 트레이너에게 요청 온 것을 확인하고 처리 한다. * 비품 컬럼인 FAULT_STATUS, ASSIGNED_TO, RESOLVED_AT 컬럼을 업데이트하고 결과를 반환한다.
		관리자		근태 관리	김진원	근태 관리	트레이너의 출근/퇴근 및 휴가, 지각, 결근 등의 정보를 관리하고 확인할 수 있어야 한다.	* 트레이너가 매일 출근 시 "출근" 버튼을 클릭하고, 퇴근 시 "퇴근" 버튼을 클릭하여 근무시간을 기록한다. * 휴가 신청, 지각/결근 처리는 관리자가 확인/수정할 수 있다.	* 출퇴근 시 서버시간으로 테이블에 기록, 스케줄러로 지각·결근 자동 처리, 휴가 신청은 테이블에 저장 후 승인 시 근태 반영, 관리자 페이지에서 전체 근태현황 열람 및 수정 가능 *DB 트리거 사용 BEFORE UPDATE checkout_time을 쓰려 할 때 checkin_time이 비어 있으면 예외 발생 시켜 업데이트 막기  *퇴근 시간이 있을 땐 반드시 출근 시간이 있어야한다. CHECK 제약 사용 CHECK (checkout_time IS NULL OR checkin_time IS NOT NULL);  3.프론트 단에서 출근 기록이 없으면 퇴근 버튼 비활성화
		직원		수업 예약	김진원	수업 예약	일반 회원은 로그인 후 트레이너의 등록된 수업 일정 중 원하는 시간대를 선택하여 수업을 예약하거나, 이미 예약한 수업을 취소할 수 있다.	* 일반 회원들은 트레이너의 일정에 맞추어 수업을 예약/취소 할 수 있다.	*트레이너 일정 목록에서 예약 가능한 항목만 표시 *버튼 클릭 시 Ajax로 예약 요청 전송 *로그인 체크 → 중복 예약 체크 → DB 저장
	1	관리자		양식 생성	윤문현	양식 생성	관리자는 전자결재 양식을 생성 및 저장 할 수 있다.	* 관리자는 에디터를 통해 양식을 생성할 수 있다. * HTML 또는 JSON 형태로 저장된다.	* CKEditor 라이브러리를 사용하여 에디터를 통해 양식을 만들. * 만든 양식을 HTML 또는 JSON 형태로 만들어서 DB에 저장
	2	관리자		양식 등록	윤문현	양식 등록	관리자는 생성한 양식을 등록할 수 있다.	* 양식을 생성하면 각 카테고리 별로 해당 양식을 등록할 수 있다. * 관리자 설정에서 등록된 양식들을 볼 수 있다. * 카테고리는 보고서, 출장 신청서, 연차/휴가 신청서, 교육/훈련 신청서, 근무시간 변경, 채용 승인 요청(강사포함), 시간 외 근무 신청, 사직서, 장비 물품 구매, 행사계획서, 신규 시설 개설	* 카테고리 테이블에 데이터를 저장 * 양식 테이블에 카테고리id 컬럼을 참조하여 카테고리 테이블과 연결시켜 양식을 저장한다.
						양식 사용	사용자는 등록되어있는 양식을 가져와 사용할 수 있다.	* 사용자는 결재 신청 시 카테고리를 선택한다. * 카테고리 별로 해당 양식을 가져와 쓸 수 있다.	* 카테고리를 선택 * 해당 카테고리에 저장된 양식을 DB에서 HTML을 조회 후 JSP에 렌더링하여 사용자에게 보여준다
							사용자는 결재 리요을 지정할 수	* 사용자는 총 회원 목록에서 결재 요청하고 싶은 사람을 결	* jsTree 라이브러리(CDN방식)를 사용하여 결재라인을 트리 형식으로 만든다. * 회원의 조 결재지를 선택하면 트리형식이 결재리이에 해당

결재 관리	3	직원		결재 신청	윤문현	결재 라인	사용자는 결재 라인을 시정할 수 있다.	새사 순번을 통해 시정할 수 있다 * 결재자는 최소 1명이상이어야 한다. * 드래그를 통해 순번을 바꿀 수 있다.	* 회원들 중 결재사를 선택하면 트리영식의 결재라인에 해당 결재자가 보여진다. * 드래그를 활용한 이벤트처리로 결재자 순번을 바꿀 수 있다.
						서명 및 도장	사용자는 서명 및 도장을 등록할 수 있다.	* 사용자는 결재 신청란에 서명을 할 수 있다. * 사용자는 도장 이미지를 업로드할 수 있다.	* 서명/도장 둘 중 하나 선택. * 서명은 Signature Pad 라이브러리를 사용하여 서명 패드 구현 * 도장은 JSP의 파일 업로드를 사용하여 도장 이미지 업로드 * 서명일 경우 base64형식으로 변환하여 서버로 보낸 후 서버에서 이미지 파일로 변환하여 HDD와 DB에 저장 * 도장일 경우 바로 서버에서 이미지 파일로 HDD와 DB에 저장(사용자별로 저장) * 서명테이블에 서명/도장 여부 저장 * 전자결재문서를 렌더링할 때 사용자 ID 기반으로 DB에 저장된 이미지 파일을 불러와 문서의 서명 위치에 삽입
	4	직원		결재 변경	윤문현	결재 수정	결재내용을 변경할 수 있다.	* 결재요청자는 결재내용(카테고리, 양식, 내용 등)을 변경하여 다시 제출할 수 있다.	* CRUD의 U방식이며 메소드형식은 PATCH를 사용하여 구현. * 한번이라도 승인 OR 거부가 되면 변경 불가.
	5	직원, 관리자		결재 취소	윤문현	결재 삭제	결재를 취소할 수 있다.	* 결재요청자나 관리자는 결재요청을 취소할 수 있다.	* CRUD의 D방식이며 메소드 형식은 DELETE를 사용하여 구현. * 한번이라도 승인 OR 거부가 되면 취소 불가.
	6	직원, 관리자		승인	윤문현	승인	각 결재자는 결재 요청을 승인, 거부 할 수 있다.	* 결재자는 승인요청이 오면 알람이 오고 승인, 거부를 할 수 있다. * 승인이 되면 다음 결재자에게 알림을 보낸다. * 최종 승인이 되면 완료 상태로 변경 후 PDF저장	* 결재 요청이 오면 전자결재 문서테이블에 데이터를 INSERT 상태는 대기 * 결재상황 테이블에 트리구조로 결재라인의 결재자들의 정보와 상황을 저장(document_id로 결재문서 테이블 참조, approver_id는 결재자 아이디로 user 테이블 참조, parent_id는 다음 결재자의 결재상황 테이블의 id) * 결재문서 테이블의 결재문서ID를 참조하여 최초 결재자의 parent_id = null로 하여 루트결재자로 하고 상태는 대기 * 결재자가 승인 하면 상태는 승인 후 다음 결재자의 상태를 대기로 변경 * WebSocket 라이브러리를 활용하여 첫 결재자에게 알림을 전송. * 거부 시 상위 결재자들의 상태는 미도달, 해당 결재자는 거부로 변경 후 결재 요청자에게 알림전송 * 승인 시 다음 결재자에게 알림 전송 * 최종 승인 시 결재문서 테이블의 상태를 완료로 변경 후 결재 요청자에게 알림전송
		공통		글 목록 조회	김초롱	글 목록 조회	* 카테고리별 필터,제목/내용 검색,페이징 지원할 수 있다	* 화면 상단에 카테고리 드롭다운을 배치 * 제목·내용 검색창과 검색 버튼(또는 엔터로 검색) 제공 * 페이지당 10건씩, 이전·다음·숫자 페이지 컴포넌트 노출	* GET /posts/{id} → SELECT + view_count UPDATE
		공통		글 상세 조회	김초롱	글 상세 조회	* 제목·작성자·작성일·조회수·본문·첨부파일 표시	* 조회 시 view_count +1 * 첨부파일 리스트에서 다운로드 지원	* GET /posts/{postId} 호출 시 조회수 +1 처리 후 글 본문과 첨부파일을 조회하여 DTO로 변환·반환 (없으면 404)

전자게시판		회원		글 작성/글 수정	김초롱	글 작성/수정	* 서식 있는 글을 작성할 수 있어야 한다. * 파일 첨부는 글당 최대 5개까지 지원해야 한다. * 작성자는 자신이 쓴 글만 수정할 수 있어야 한다	* CKEditor 연동된 본문 입력 영역 제공 * ‘파일 첨부’ 버튼으로 이미지(.jpg/.png)·문서(.pdf) 선택 * 업로드 전 확장자 및 크기 검사 (≤10 MB) * 수정 시 기존 제목·본문을 에디터에 로드하여 편집 * 저장 시 updated_at 컬럼 갱신 * 작성자가 아닌 경우 수정 API 접근 차단(403 반환)	* 글 작성 REST POST /posts @RequestParam로 MultipartFile 처리 → 파일 저장 → board_post, attachment 테이블 INSERT * 글 수정 REST PUT /posts/{id} JWT 토큰 검증 → author_id 비교 → UPDATE board_post SET title=?, content=?, updated_at=NOW() 수정 불가 시 HTTP 403 반환
		관리자		글 삭제	김초롱	글 삭제	* 관리자만 글 삭제(소프트 삭제) 가능	* 상세보기 ‘삭제’ 클릭 시 확인 모달 표시 * DB에는 삭제 대신 status = 'DELETED' 처리 * UI(목록·상세)에서는 해당 글 숨김	* REST DELETE /posts/{id} role = ADMIN 체크 후 status UPDATE
		공통		검색 기능	김초롱	검색	* 제목·내용 full-text 검색, 결과 하이라이트	* MySQL Full-Text Index 생성 * 검색어가 포함된 부분을 <mark> 태그로 감싸 강조	* MyBatis MATCH(title,content) AGAINST("#{kw} IN BOOLEAN MODE) 서비스 레이어에서 하이라이트 처리 후 DTO 반환
		공통		페이징	김초롱	페이징	* 서버 사이드 페이징 지원	* 쿼리파라미터 page, size 사용 * UI는 Bootstrap Pagination 컴포넌트 연동	* SQL LIMIT #{size} OFFSET #{(page-1)*size} 컨트롤러에서 DTO에 전체 페이지 수 계산 후 전달
		회원		좋아요 기능	김초롱	좋아요/좋아요취소	* 로그인한 회원만 가능, 중복 방지	* 목록·상세에 하트 아이콘 표시 * 하트 클릭 시 AJAX 호출로 토글 처리	* POST /posts/{id}/like → post_like 테이블 INSERT/DELETE → board_post.like_count UPDATE
		공통		댓글/대댓글	김초롱	댓글/대댓글	* 로그인한 회원은 모든 게시판에서 댓글을 달 수 있어야 한다. * 댓글 작성자만 자신의 댓글을 수정·삭제할 수 있어야 한다. * 대댓글(답글)을 최대 5단계까지 작성할 수 있어야 한다.	* 게시글 상세보기 하단에 댓글 리스트 노출 * “댓글 작성” 입력창과 “답글” 버튼 제공 * ‘답글’ 클릭 시 해당 댓글 아래 인라인 입력폼 생성 * parent_id 컬럼으로 댓글·대댓글 계층 구조 관리 (최대 5단계) * 수정·삭제 버튼은 댓글 작성자에게만 노출	* comment 테이블에 INSERT (parentId 없으면 최상위) → JWT로 작성자 검증 → UPDATE comment SET content=?, updated_at=NOW() → 작성자 검증 후 소프트 삭제(status='DELETED') 또는 물리 삭제
		공통		파일 업로드/다운로드	김초롱	파일 업로드/다운로드	* 글당 최대 5개 첨부, 다운로드 지원	* 첨부파일 리스트 노출(파일명, 용량) * 다운로드 시 Content-Disposition 헤더로 파일명 지정	* POST /posts/{id}/files → MultipartFile 저장 → attachment INSERT  * GET /files/{fileId} → 파일 스트림 반환
	1	공통		알림 시스템	윤문현	알림 시스템	알림 기능을 사용할 수 있다.	* 결재 요청/완료 알림 * 예약 승인/취소 결과 알림 * 휴가 승인 결과 알림 * 채팅 초대/투표 결과 알림 등등	* WebSocket을 사용해서 내부 알림 기능으로 구현.
	2	관리자		로그/감사 기록	윤문현	로그/감사 기록	로그/감사 기록을 볼 수 있다.	* 관리자는 중요한 변경(예약 수정, 결재 변경, 근태 수정) 등에 대한 기록을 열람할 수 있다.	* 관리자 설정에서 카테고리 별로 해당 기록을 볼 수 있도록 구현.

편의/기타	3	관리자		통계/차트 시각화	윤문현	통계/차트 시각화	각종 통계와 차트를 시각화 하여 볼 수 있다.	* 관리자는 각종 예약률, 출석률, 휴가 현황, 전자결제 처리율 등 통계와 차트를 시각화하여 볼 수 있다.	* 관리자 설정에서 볼 수 있도록 구현. * Chart.js, Recharts, ApexCharts 등 라이브러리를 활용하여 통계를 그래프 등으로 시각화하여 구현.
	4	공통		다국어 지원	정우진	다국어 지원	다국어를 지원한다.	* 영문 UI 지원.	* application_dev.properties에서 각종 메시지 설정. * messages_en.properties 파일을 활용. * 언어 카테고리를 활용하여 설정 * JSP에서 다국어 메시지 출력
	5	관리자		점검/휴관 등록	김진원	점검/휴관 등록	시스템 점검과 휴관일 등록할 수 있다.	* 관리자는 시스템 점검과 휴관일을 등록할 수 있다. * 등록된 해당 날짜는 모든 예약이 취소된다. * 해당 날짜는 예약이 불가능하다.	* 관리자 설정에서 점검/휴관 등록 카테고리로 해당날짜를 등록한다. * 등록된 해당 날짜의 모든 예약은 DELETE시킨다. * 더이상 해당 날짜에 예약은 불가능하다.


비고(전제조건 등)
<div>* DB에 로그인하려는 사용자의 정보가 저장되어 있어야 한다.</div> <div>* 구독권이 만료된 사용자일 경우, 로그인 시 대부분의 기능을 사용할 수 없으며, 구독권 결제 알람이 페이지 Header부분에 노출되게 된다.</div>
<div>* 로그인한 사용자만이 로그아웃할 수 있다.</div>
<div>* 가입한 회원의 정보를 입력해야 아이디와 임시 비밀번호를 얻을 수 있음</div>
<div>* 회원가입하려는 사용자의 ID는 DB에 저장되어 있는 ID와 중복되지 않아야 한다.</div> <div>* 회원가입 후 첫 로그인 시, 구독권 결제 페이지로 이동된다.</div>
<div>* 로그인한 사용자만이 마이페이지 탭을 이용 가능하다.</div> <div>* 채팅방 신규 생성은 불가하며 이미 속해 있는 채팅방만 노출된다.</div> <div>* 일정 또한 생성은 일정 탭에서만 가능하다.</div>

- \* 로그인한 '일반유저'만이 구독페이지를 사용할 수 있다.
- \* 하나의 카드 당 한개의 빌링키가 생성된다.
- \* 구독 취소를 하지 않는 이상 정기 결제가 이루어 진다.
- \* 취소를 한 뒤 구독권 만료일까지 지나게 되면 모든 기능을 사용할 수 없다.

- \* 로그인한 유저만이 구독 페이지를 사용할 수 있다.
- \* 구독권이 만료된 일반회원이나 관리자에 의해 승인받지 못한 트레이너는 해당 기능을 사용할 수 없다.
- \* 필수적으로 친구가 되어야 채팅을 시작할 수 있으나, 특정 인원예 의해 친구가 아님에도 그를 채팅에 추가될 경우 채팅방 내에서 친구 추가가 가능하다.

\* 관리자 등급 ID로 로그인 해야만 이 탭으로 이동 가능하다.

- \* 방장 권한의 회원만이 채팅방의 회원들을 강퇴할 수 있다.
- \* 방장은 채팅방 퇴장 시 방장 권한을 위임 후 퇴장할 수 있다.
  - \* 친구 사이가 아닌 회원은 초대할 수 없다.
- \* 같은 채팅방에 친구가 아닌 회원에게 친구 요청을 보낼 수 있다.
- \* 트레이너는 가입 시 관리자에 의해 부서에 추가될 경우 부서 특방에 자동 추가된다.

\* 로그인한 사용자만 확인 할 수 있다.

\*로그인한 사용자만 확인 가능하다.  
\*본인 계정으로만 본인 스케줄 확인 가능하다.

\* 관리자 전용  
\* 트레이너 스케줄 등록



* 동일 시설/시간대 예약 방지 로직 필요 (서비스 단에서 체크)
* 수정/취소 확정 시 트레이너에게 알림 전송 (메일 또는 알림센터 형태)
*관리자 전용
*관리자 전용
<div>* 관리자는 전체 트레이너 근태현황 열람 및 수정 가능. 트레이너는 본인의 근태정보만 열람 가능.<div>* 9시까지 정상 출근</div><div>* 10분 늦을시 지각</div><div>* 1시간 늦을시 결석 (패널티 적용해야 함)</div></div>
<div>*일반 회원만 수업 예약 가능 (트레이너는 예약 불가)</div> <div>*로그인한 사용자만 예약 기능 접근 가능</div> <div>*하나의 일정에 중복 예약 방지 (같은 시간대 중복 제한 필요)</div>
<div>* 로그인이 완료된 상태여야 한다.</div> <div>* 등급이 관리자여야 한다.</div>
<div>* 로그인이 완료된 상태여야 한다.</div> <div>* 양식 테이블에 저장된 양식이 있어야한다.</div>
<div>* 로그인이 완료된 상태여야 한다.</div>

- \* 로그인이 완료된 상태여야 한다.
- \* 결재 순번은 높은 직급일수록 더 낮은순번이어야 한다.

- \* 로그인 완료된 상태여야 한다.
- \* 도장일 경우 base64형식을 안걸침

- \* 로그인 완료된 상태여야 한다.
- \* 해당 결재 요청건이 있어야 한다.
- \* 승인이나 거부가 안된 상태여야 한다.

- \* 로그인이 완료된 상태여야 한다.
- \* 해당 결재 요청건이 있어야 한다.
- \* 승인이나 거부가 안된상태여야 한다.

- \* 로그인이 완료된 상태여야 한다.
- \* 결재문서 테이블은 한 결재에 대한 여러 결재자들의 그룹 테이블
- \* 결재상황 테이블(트리구조)은 결재문서 테이블의 한 데이터에 해당하는 결재자들의 결재 진행 상황을 따로 분리한것

\* 누구나(비로그인 포함)접근 가능하다

\* 로그인이 완료된 상태여야 한다.

\* 수정 이력은 필요 시 별도 테이블에 최대 5건만 보관 예정

<div>* 로그인 이 완료된 상태여야 한다.</div> <div>* 수정 이력은 필요 시 별도 테이블에 최대 5건만 보관 예정</div>
<div>* 관리자 전용 , 삭제된 글은 관리자 페이지에서만 조회 가능</div>
<div>* 누구나(비로그인 포함)접근 가능 공통 검색 로직 모듈화 추천</div>
<div>* SQL LIMIT #{size} OFFSET #{(page-1)*size} 컨트롤러에서 DTO에 전체 페이지 수 계산 후 전달</div>
<div>* 로그인 이 완료된 상태여야 한다.</div>
<div>* 로그인 이 완료된 상태여야 한다.</div> <div>댓글·대댓글 계층 구조는 parentId와 depth 컬럼으로 관리</div>
<div>* 로그인된 사용자만 첨부 가능 대용량 파일은 별도 스토리지 사용 권장</div>
<div>* 요구사항 상세뿐 아니라 모든 알림이 필요한 기능에 적용 시킬 예정</div>
<div>* 보안 및 운영 이슈 대비</div> <div>* 관리자 등급 ID로 로그인 해야만 노출 가능하다.</div>

<div>* 각종 통계에 대한 데이터가 있어야한다. * 관리자로 로그인 해야한다.</div>
<div>* 외국인 회원들을 위함.</div>
<div>* 갑작스런 휴일 OR 시스템점검 대비 * 관리자로 로그인 해야한다.</div>