

Bike Friønds

프로젝트 주제 : 자전거 커뮤니티 사이트

팀명: 육캔두잇!

팀 소개: 자전거 이용률이 증가함에 따라 자전거와 관련된 커뮤니티 구성과 사람들 간의 소통을 활성화하려는 서비스 제공을 위해 모인 팀

자전거를 이용하는 누구나 쉽게 사용할 수 있고 오프라인으로 같은 지역의 사람들과 모임을 형성 할 수 있는 자전거 커뮤니티 사이트

6조

팀장 :신범식

조원 :조창현, 채홍균, 이연수, 서예주, 이돈우

주제 선정의 이유

- 지역 기반 플랫폼 증가로 인해 지역 커뮤니티가 활성화되고 있음
- 공공기관에서 탄소 저감으로써 성장 가능성이 높은 사업으로 인식하고 있음
- 전국적으로 자전거 보유율이 상승하고 있고 자전거 대여율도 높아지고 있다.

서울시 공공자전거 '따릉이', 탄소감축 캠페인..참여 시민에 백화점 상품권

김두열 선임기자 | 입력 2023-05-11 14:03

기사출처: [\[기사\]](#) [\[인쇄\]](#) [\[가기\]](#)



서울시청 [사진=서울시]

실시간 인기

[총급](#) [경제](#) [정치](#) [사회](#) [문화](#)

지역 기반 플랫폼 설치자 수 추이

ETDI
ECONOMIC & TECHNOLOGY DEVELOPMENT INSTITUTE



분석기간: 21.01 ~ 05/ 분석방법: 2021년 1월 대비 출처: TDI
Copyright © TDI. All rights reserved. 무단전재 및 재배포 금지

서울시 따릉이 이용 현황

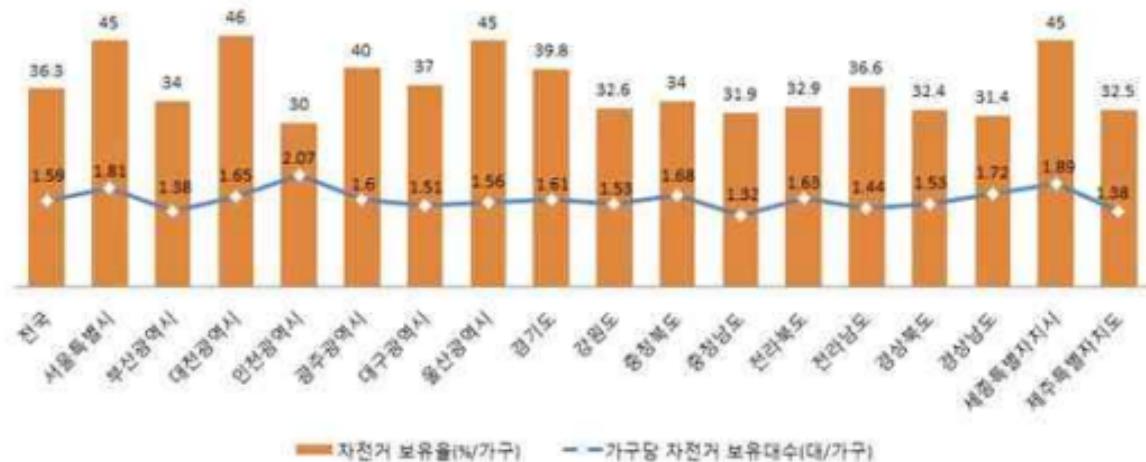
자료: 서울시



대전시, 공공자전거 '타슈2' 연말까지 5000대로 늘린다

등록 2021.05.09 16:09:39

인쇄 PDF PDF



기존 사례 분석

카페 154건



파시메니아 (파시자전거 커뮤니티)

파시메니아 (파시자전거, 콘스탄틴, 트레바운, 언노운, 펜...
주제 스모즈/여자 > 자전거 | 조회수 109,795
행길 써웃4단계 | 댓글/문제글 8/983,615
카페포토갤 >

FLATS

자전거 입문 커뮤니티 플래츠(Flats)

자전거 입문 및 조언리를 위한 커뮤니티로 라이팅 및 교...
주제 스모즈/여자 > 자전거 | 조회수 102
행길 써웃1단계 | 댓글/문제글 0/129
지역 서울특별시 | 카페포토갤 >

한국 자전거 커뮤니티

자전거를 좋아하는 사람들의 각자 가지고 있는 정보를 ...
주제 스모즈/여자 > 자전거 | 조회수 9
행길 써웃1단계 | 댓글/문제글 0/5
카페포토갤 >



장쩌도운✓ | 렇길은 | 걸어 수 | 사용 수 | 현재 수

자출사 '자전거로 출퇴근하는 사람들' |
회원수 79만 국내 최대 대표 자전거 커뮤니티! 자전거의 ...
주제 생활 > 생활정보 | 조회수 792,582
행길 출 | 댓글/문제글 448/2,152,243
카페포토갤 >



라이더즈랩 - 온라인 자전거 커뮤니티

안녕하세요. 라이더즈랩의 캐런나입니다.
주제 스모즈/여자 > 자전거 | 조회수 1
행길 써웃1단계 | 댓글/문제글 0/1
카페포토갤 >



방수드 자전거 스타일 커뮤니티 [패션, 코스, 및
길]
스란힐라리한 로드 사이클링 패션 및 라이팅 패스, 그리고...
주제 스모즈/여자 > 자전거 | 조회수 36
행길 써웃2단계 | 댓글/문제글 0/6
지역 서울특별시 > 마포구 | 카페포토갤 >

- 기존에 자전거 게시판이 여러개 있지만 여행 목적으로 개설된 게시판이 많음.
- 서울시 내에서 각 지역별로 자전거 모임을 가질 수 있는 게시판이 대중화가 되어있지 않음.
- 각 게시판에서 따로 자전거를 대여할 수 있는 기능이 없음.
- 따라서 자신의 주변 지역 사람들과 함께 할 수 있는 자전거 커뮤니티를 개설하려고 함.

두 플랫폼의 차이점

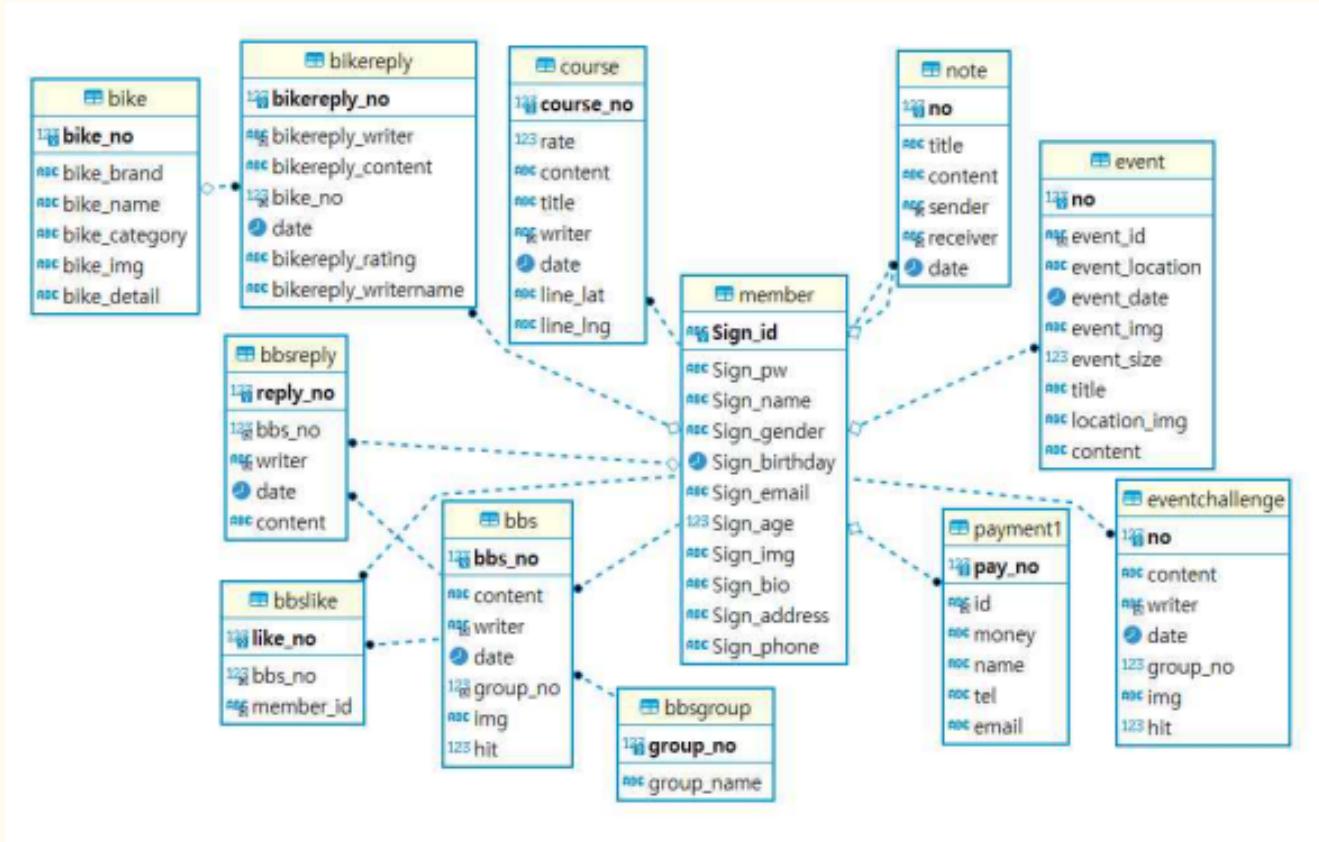
따릉이	더바이크(자전거 전문 웹진)
<ol style="list-style-type: none">1. 공유 자전거를 결제하고 이용할 수 있는 시스템2. 자전거 위치를 확인하고 이용하는 개인적 용도로 많이 쓰임3. 해당 대여소에 따릉이가 몇대 있는지 실시간으로 확인 가능4. 회사에서 제공하는 한가지 종류의 자전거만 이용가능(제한적)	<ol style="list-style-type: none">1. 자전거 행사, 쇼핑몰, 갤러리를 포함한 사이트2. 어떤 자전거 관련 정보가 있는지, 모임이 있는지에 대한 공유하려는 용도로 쓰임3. 자전거관련 정보를 업로드를 통해 사이트를 이용하도록 실시간으로 내가 원하는 정보를 찾기는 어려움4. 내가 어떤 자전거를 타야 하는지, 장비를 착용해야 하는지 관련글을 읽어본 뒤 선택 가능

두 플랫폼의 장단점을 고려하여 공유자전거 기능과 자전거 커뮤니티를 동시에 제공하는 프로젝트 완성이 설계방향

주요 기능 및 역할

서예주	이돈우	신범식	이연수	조창현	채홍균
마이페이지	쪽지	메인페이지+자전 거정보	그룹별 게시판	이용권 구매	이벤트
회원가입/로그인	쪽지함 조회	메인페이지	게시글 등록/수정/ 삭제	결제 시스템 이용 하여 이용권 구매	파동이 대여소 나 타내기
내 정보 수정	쪽지전송/답장	자전거 정보	댓글 등록/수정/삭 제	구매내역	자전거 모임 개설
비밀번호 변경	쪽지 일괄 삭제	자전거 후기 등록	인기 게시글	Import(아임포트)	인증샷 철린지
비밀번호 찾기			코스 공유하기		지도 API
네이버 로그인 API			지도 API		파동이 API

ERD



협업 툴(Notion)

◎ 프로젝트(BF)

● 진행 중	● 타임라인	● 보드	■ 정체	+
+ 날짜		상태 계획 총진행 중완료		
+ 달력		담당자		
Aa 프로젝트 이름	● 상태	■ 날짜	■ 달당자	■ 정체
● 주제 및 메뉴 확장	● 완료	2023년 5월 2일		
● 요구사항, db테이블 정의	● 완료	2023년 5월 3일 ~ 2023년 5월 4일		
● 순서도, 유스케이스	● 완료	2023년 5월 3일		
● 프로젝트 청의서	● 완료	2023년 5월 4일		
● WBS	● 완료	2023년 5월 4일		
● 지역별 개시판	● 완료	2023년 5월 5일 ~ 2023년 5월 16일	● 이연수	
● 쪽지 가능 구현	● 진행 중	2023년 5월 11일 ~ 2023년 6월 2일	● 이든우	
● 미번트 페이지	● 진행 중	2023년 5월 11일 ~ 2023년 6월 2일	● 홍근재	
● 자전거 페이지	● 진행 중	2023년 5월 11일 ~ 2023년 6월 2일	● 염식신	
● 로그인, 회원가입 페이지	● 진행 중	2023년 5월 11일 ~ 2023년 6월 2일	● 서예주	
● 이용권 구매 페이지	● 진행 중	2023년 5월 12일 ~ 2023년 5월 19일	● CHJO	
● 자전거 코스 공유 페이지	● 진행 중	2023년 5월 15일 ~ 2023년 5월 29일	● 이연수	
● 프로젝트 합치기	● 진행 중	2023년 5월 23일 ~ 2023년 6월 9일		

회의

■ 회의 ■ 캘린더 ■ 유형별 리스트 ■ 대회의 ■ 정체 +

● 참석자 ● 유형 ● 생성자 ● 회의 날짜 + 필터 추가

- 미니 프로젝트 회의 1
- 미니 프로젝트 회의 2
- 미니 프로젝트 회의 3
- 미니 프로젝트 회의 4
- 미니 프로젝트 회의 5
- 미니 프로젝트 회의 6
- 미니프로젝트 리뷰
- 최종 프로젝트 회의
- 최종 프로젝트 회의 구현(예시)
- 최종프로젝트 순서도와 유스케이스
- 최종프로젝트 요구사항 +db테이블정의서
- 멘토링 1
- 멘토링 2
- 멘토링 3

필터 정렬 Q ← 새로 만들기 +

- 미니 프로젝트 회의 1 4월 18일 오후 1:12 4월 14일
- 미니 프로젝트 회의 2 4월 20일 오후 11:47 4월 17일
- 미니 프로젝트 회의 3 4월 19일 오후 1:24 4월 18일
- 미니 프로젝트 회의 4 4월 21일 오후 5:54 4월 19일
- 미니 프로젝트 회의 5 4월 21일 오후 5:03 4월 20일
- 미니 프로젝트 회의 6 4월 21일 오후 4:23 4월 21일
- 브레인风暴 4월 26일 오후 11:47 4월 27일
- 미니 프로젝트 회의 7 5월 2일 오후 2:42 5월 2일
- 회의 구현 5월 3일 오후 1:58 5월 2일
- 미니 프로젝트 회의 8 5월 3일 오후 5:17 5월 3일
- 미니 프로젝트 회의 9 5월 3일 오후 5:27 5월 3일
- 멘토링 1 5월 31일 오후 2:17 4월 29일
- 멘토링 2 6월 2일 오후 11:04 5월 13일
- 멘토링 3 6월 2일 오후 11:06 5월 27일

협업 툴(Slack)

The screenshot shows the Slack desktop application interface. At the top, the title bar reads "[멀티잇]백엔드 개발자 채용캠프[Java] 우회자 경매". The left sidebar lists various channels and direct messages, including:

- [멀티잇]백엔드 개...
- # 나중에
- △ 다이렉트 메시지
- ⑤ 맨션 및 반응
- ▷ 초안 및 전송됨
- Slack Connect
- 더 보기
- * 즐겨찾기
- # 0-공지
- # 1-질문-및-채팅
- # 6조-신범식-홍균-장현-예주-연수-돈우...
- △ 멀티캠퍼스_총무정 매니저
- 멀티캠퍼스_임정원
- 멀티캠퍼스_임정원, 박상진_...
- 예상임_보조강사
- * 채널
- # 0-영상공유
- # 0-자유게시판
- # 0-정보공유
- △ 알고리즘3_윤호-서범식-신범...
- + 채널 추가
- * 다이렉트 메시지
- 신범식_6조 나
- 예연수_6조
- 6조-신범식-홍균-장현-예주...

The main window displays the "# 6조-신범식-홍균-장현-예주-연수-돈우" channel. A message from "체홍균_6조" at 9:19 AM says "마음이 api 지도구한 마무리하기". Below it, "존창현_6조" at 9:20 AM says "정기권 일정권의 {d-day}기간만료 부분을 추가". Another message from "체홍균_6조" at 9:20 AM follows. A message from "예연수_6조" at 9:20 AM says "자전거 고스 공유 상세페이지 및 db연동". A message from "6조 이돈우" at 10:08 AM says "쪽지 crud 기능 구현 및 db연동". A timestamp at the bottom indicates "5월 19일 금요일".

The message list has a search bar at the top right with placeholder text "필: 이 채널을 검색하기 위해 Ctrl F 시도 x". Below the messages is a rich text editor with various formatting options.

깃 관리(소스트리)

파일(F) 편집(E) 보기(V) 저장(S) 예전(A) 도구 도움말(H)

finalbike exfinalbike

커밋 Pull Push 대치 브랜치 병합 스테시 메시지 태그

.workspace 파일 상태 History Search

브랜치 dev master 태그 origin 흐름

모든 브랜치 원적 브랜치 날짜순으로 정렬

설명

날짜 작성자 커밋

6/2 6 번수 이름 수정 Merge branch 'dev' of https://github.com/shinbeomslk/finalbike into dev
최종 버전입니다.
검색하기 위해서 likeController과 Bkemapper 수정 검색결과로 인해 검색결과 페이지를 대체합니다 리뷰 수정할때 링크를 데이타베이스에 받아서 바로 적용수도록 수정

6/2 2 6 2023 9:33 Lee Yeon Su <ylee> 2dd164f
2 6 2023 9:32 DESKTOP-6IRPENI 942bd4b
shinbeomslk <sbs> c763767
tsley <yeju04346> 1565e6b
origin/donwoo ui 수정입니다.
토론 확인용
검색 제대로 하기위해 searchIO를 따로 만들
ui 수정본입니다.

6/2 3 6 2023 9:27 <origin/donwoo> <olparkt> a8f0101
shinbeomslk <sbs> b1ed2bc
shinbeomslk <sbs> c763767
olparkt <olparkt> 326d1ef
shinbeomslk <sbs> 0c93f7
olparkt <olparkt> 7683e18
shinbeomslk <sbs> 365f2d8
tsley <yeju04346> 516fb44
shinbeomslk <sbs> 4621244
shinbeomslk <sbs> 2a92b28
shinbeomslk <sbs> 05b0632
tsley <yeju04346> 56f6b63
Lee Yeon Su <ylee> 335cc1b
DESKTOP-6IRPENI 21d5db3
olparkt <olparkt> 118df17
shinbeomslk <sbs> 7ac5817
Lee Yeon Su <ylee> e970dbc
shinbeomslk <sbs> b6b2590
shinbeomslk <sbs> f69a92e
DESKTOP-6IRPENI e452eb3
shinbeomslk <sbs> 6c5da49
olparkt <olparkt> 49cb2a0
shinbeomslk <sbs> ee8321
shinbeomslk <sbs> f72fdad
olparkt <olparkt> 3b1b198

6/1 4 6 2023 17:21 6/1 5 6 2023 22:54
6/1 6 6 2023 17:33 6/1 7 6 2023 17:53
6/1 8 6 2023 17:52 6/1 9 6 2023 17:51
6/1 10 6 2023 17:51 6/1 11 6 2023 17:49
6/1 12 6 2023 17:21 6/1 13 6 2023 17:20
6/1 14 6 2023 16:33 6/1 14 6 2023 14:37
6/1 15 6 2023 14:07 6/1 15 6 2023 14:07
6/1 16 6 2023 12:19 6/1 16 6 2023 12:19
6/1 17 6 2023 11:31 6/1 17 6 2023 11:31
6/1 18 6 2023 10:49 6/1 18 6 2023 10:49
6/1 19 6 2023 22:42 6/1 19 6 2023 22:42
6/1 20 6 2023 17:18 6/1 20 6 2023 17:18
6/1 21 6 2023 17:15 6/1 21 6 2023 17:15
6/1 22 6 2023 15:51 6/1 22 6 2023 15:51

파일 상태 순 정렬 블록

커밋 1935fb69730efb831ec811d598a6c689a47fb6c98f [1935fb69]
상위 브랜치: main/master, origin/main

src/main/resources/mapper/noteMapper.xml

부분 : 5-11 중

src/main/resources/mapper/noteMapper.xml

<Insert id="create" parameterType="noteVO">

시스템구성도

개발



협업



오픈 API



DBMS



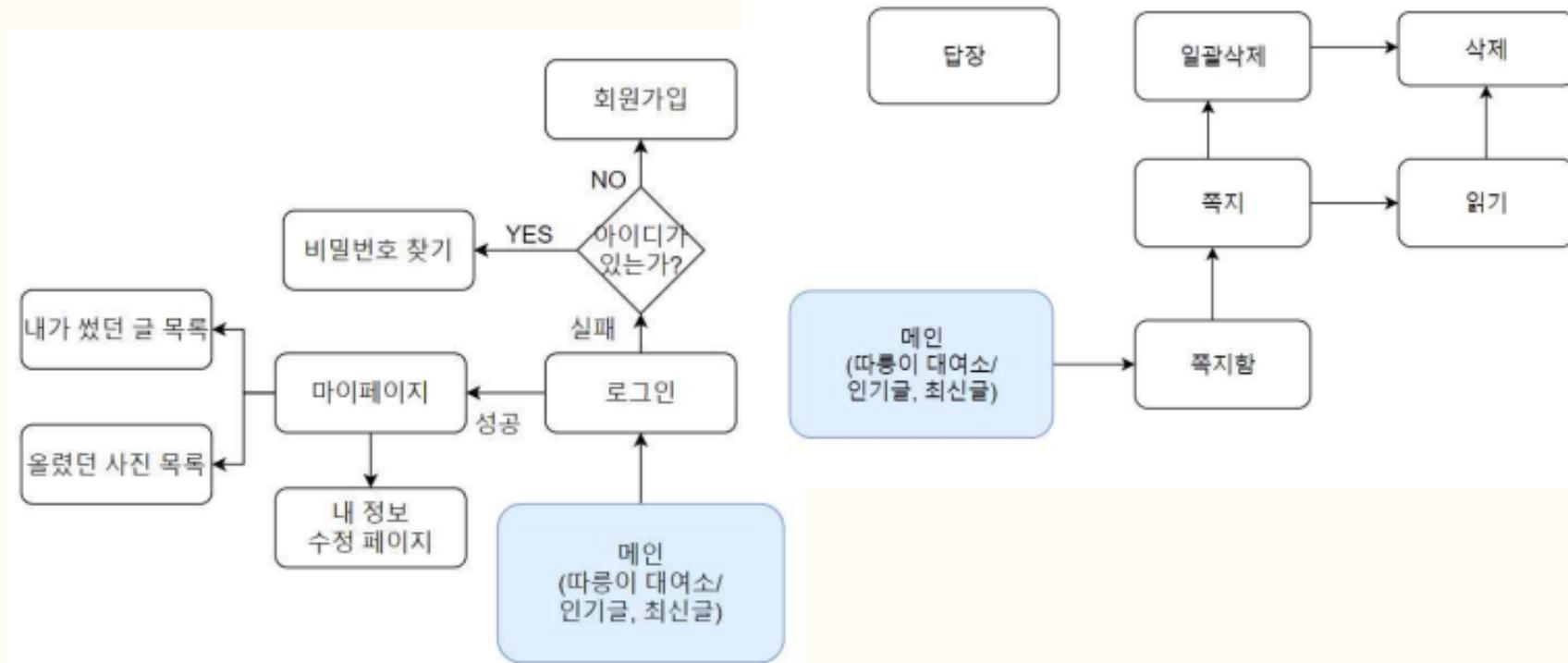
배포



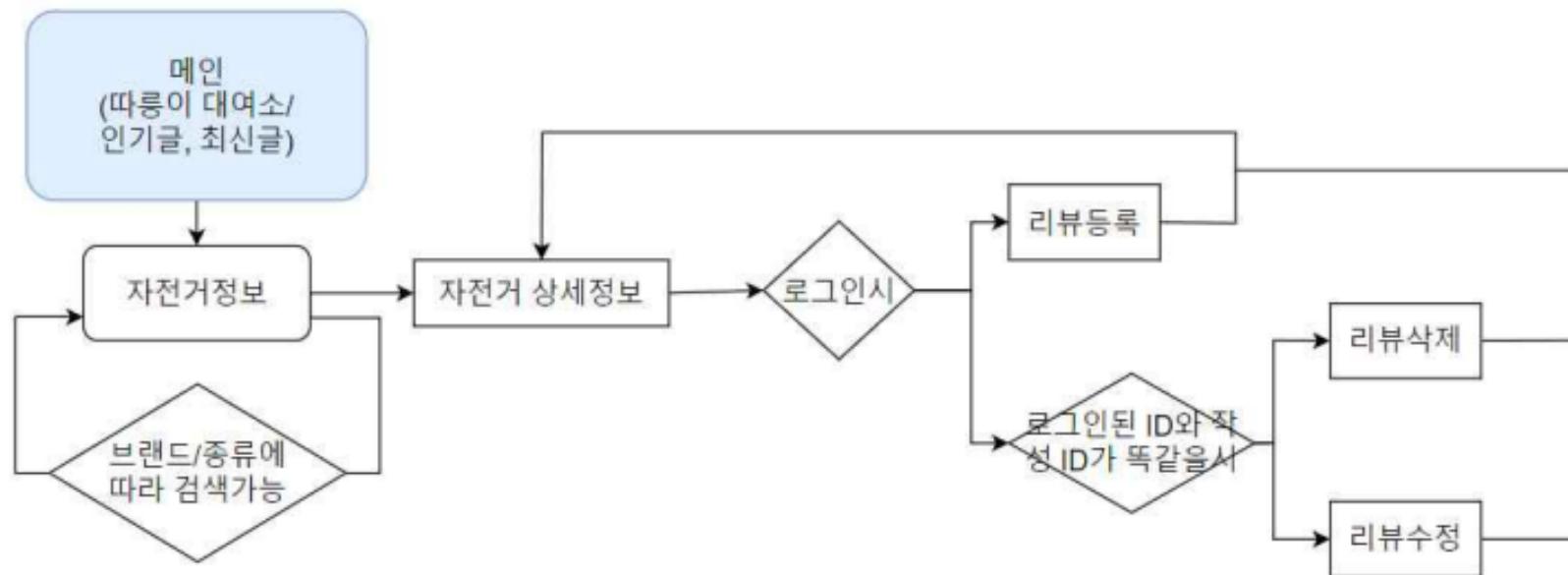
import



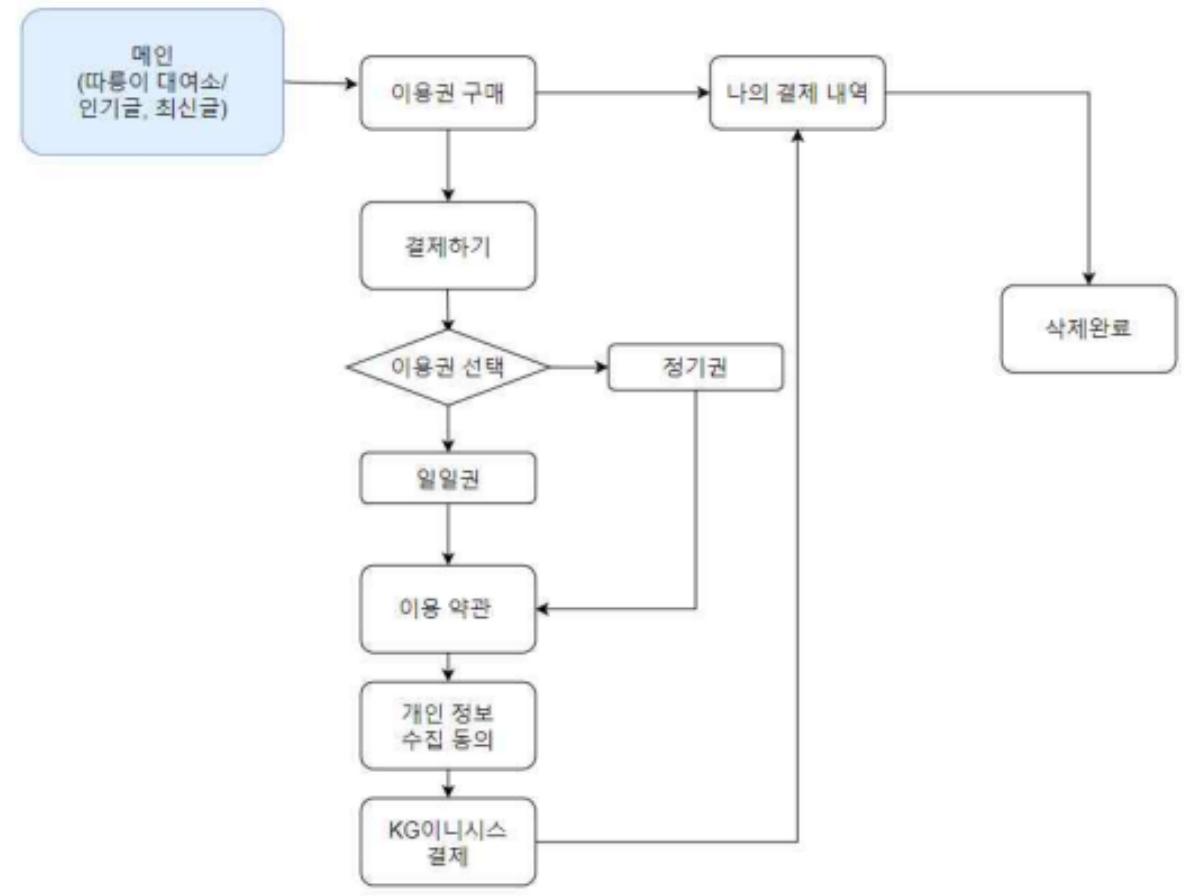
전체 순서도



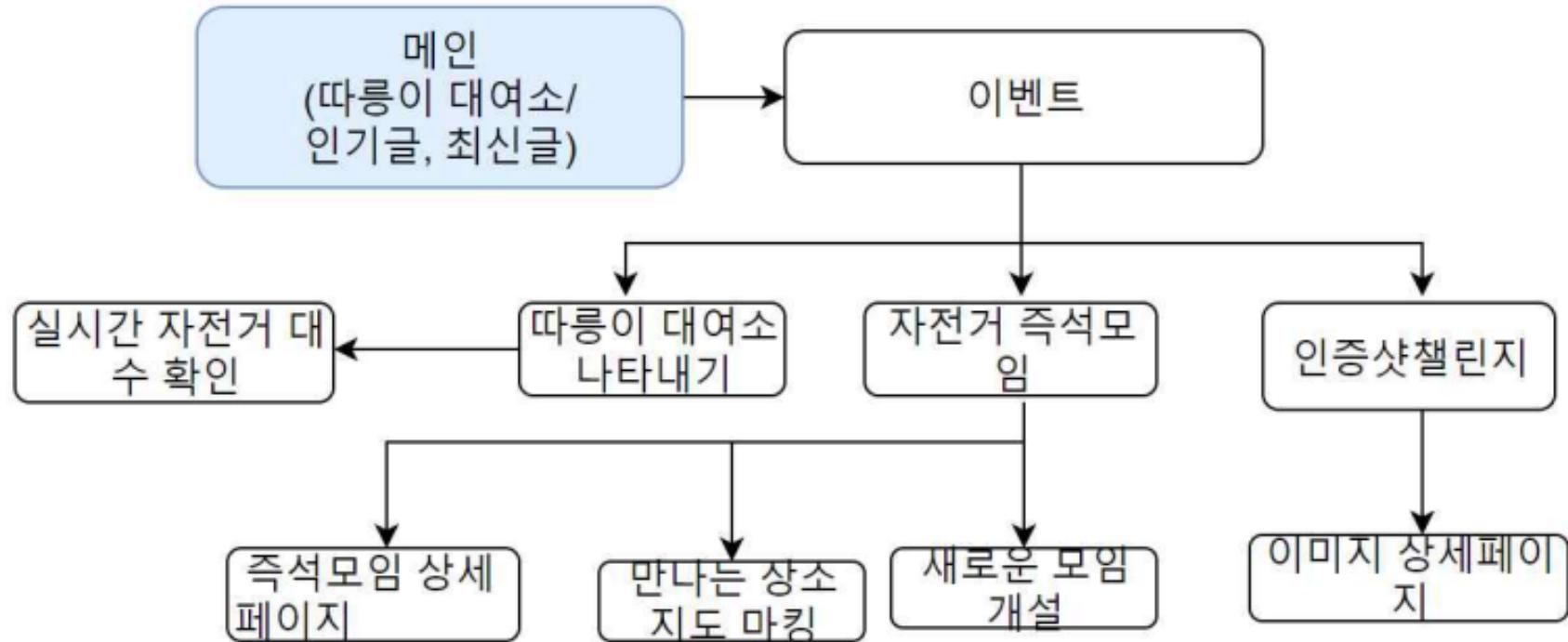
전체 순서도



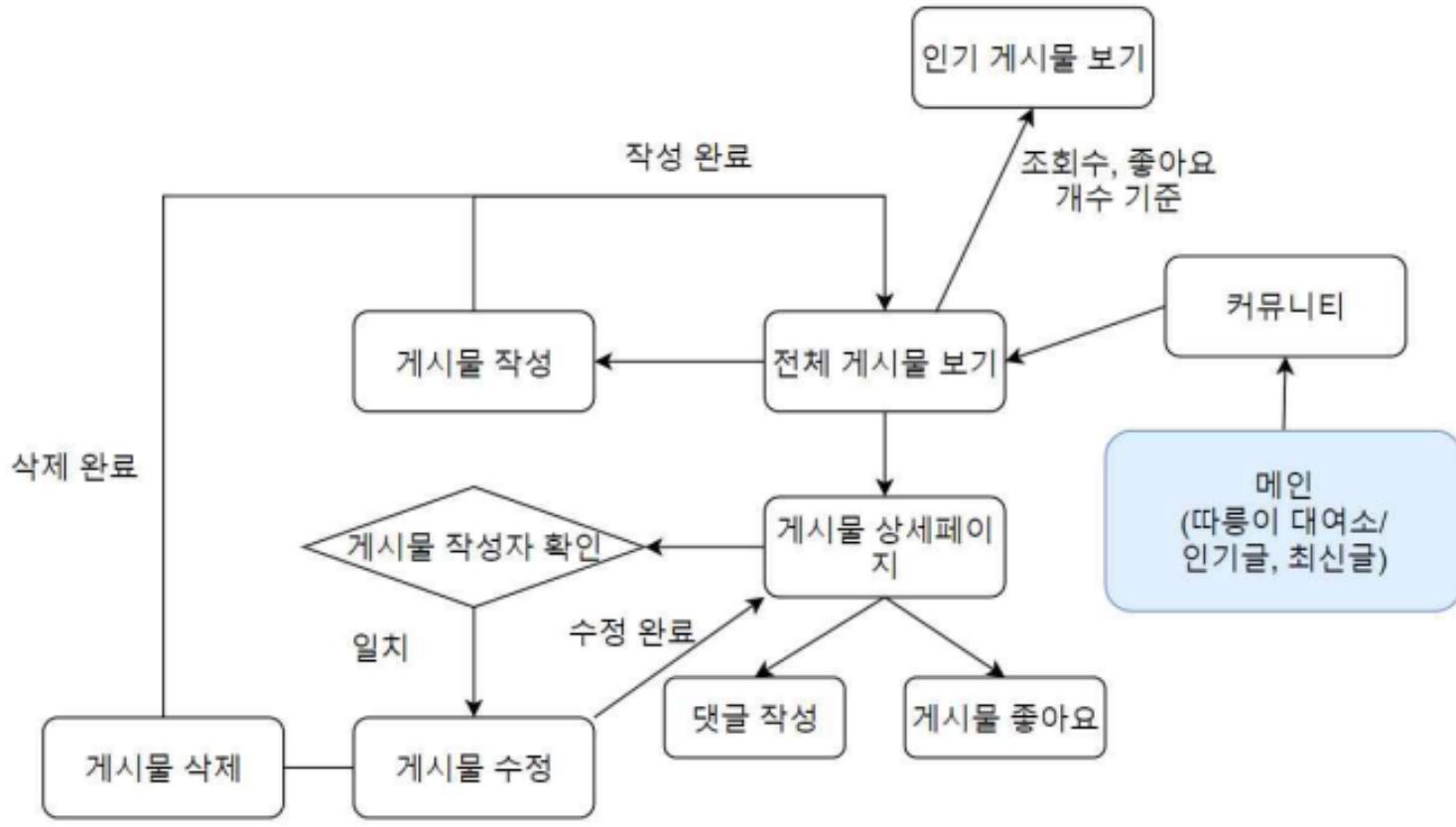
전체 순서도



전체 순서도



전체 순서도



요구사항 정의서 (로그인, 게시판, 자전거 정보)

자전거 커뮤니티			
작성자	서예주	작성일	23.05.03
버전	1.0.0	최종수정일	23.06.07

구분	요구ID	내용	비고
로그인 폴더지	SYJ_001	아이디 입력창	Input
	SYJ_002	비밀번호 입력칸	Input 필드
	SYJ_003	로그인 버튼	속성에 아이디 패스워드를 ID와 비교하여 승인
	SYJ_004	소셜 로그인(네이버)	네이버 로그인 페이지로 이동
	SYJ_005	비밀번호 찾기	비밀번호 찾기 페이지로 이동
	SYJ_006	회원가입 버튼	회원가입 페이지로 이동
	SYJ_007	로그인	로그인
회원가입 폴더지	SYJ_008	Pw입력	
	SYJ_009	pw확인	
	SYJ_010	이름	
	SYJ_011	설명	
	SYJ_012	업로드	
	SYJ_013	주소	
비밀번호 찾기 폴더지	SYJ_014	이메일	
	SYJ_015	전화번호	
	SYJ_016	회원가입	아이디 중복 확인 후 가입 승인
	SYJ_017	아이디 입력	

프로젝트명	자전거 커뮤니티 Bike Friends		
작성자	이영수	작성일	2023.05.03
버전	1.0.0	최종수정일	2023.06.02

구분	요구ID	내용	비고
User Requirement -Service	LYS-001	지역별 게시판 목록 기준으로 그룹 나눔	
	LYS-002	전체/인기게시판 조회	날짜(조회수, 좋아요수 기준 정렬)
	LYS-003	게시글 상세페이지	게시글 작성자, 글 내용, 날짜 조회
	LYS-004	게시글 작성, 수정, 삭제, 좋아요	로그인시에만 가능
	LYS-005	댓글 작성, 수정, 삭제	로그인시에만 가능
	LYS-006	게시글 검색	게시글 내용 검색
Functional Requirement -Technology	LYS-007	게시판별 인기글 선택	게시글 조회수, 좋아요수 제크
	LYS-008	게시글 사진 첨부	
	LYS-009	지도 위에 선 그리기	카카오맵 api
Non Functional Requirement -Protection &Change	LYS-010	본인 게시물만 수정, 삭제 가능	로그인 유저 체크
	LYS-011	본인 댓글만 수정, 삭제 가능	로그인 유저 체크

프로젝트명	자전거커뮤니티		
작성자	신봉식	작성일	2023.05.03
버전	1.0.0	최종수정일	2023.05.03

구분	요구 ID	내용	비고
User Requirement -Service	SBS-001	자전거 정보를 확인하는 페이지로 이동	자전거 정보를 확인하는 페이지로 이동
	SBS-002	자전거 살피질보	자전거 상세정보와 리뷰쓰는게 가능
	SBS-003	리뷰작성,수정,삭제	로그인시에만 가능
Functional Requirement -Technology	SBS-004	자전거 검색	브랜드와 종류로 검색가능
Non Functional Requirement -Protection &Change	SBS-005	로그인페이지 리뷰등록가능	로그인유저 체크
	SBS-006	본인 리뷰만 수정, 삭제가능	로그인유저 체크

비밀번호 찾기 폴더지	SYJ_017	아이디 입력	
-------------	---------	--------	--

요구사항 정의서 (이벤트, 쪽지, 이용권)

개인별 요구사항 정의서

자전거 커뮤니티			
프로젝트명	작성자	작성일	최종수정일
자전거 커뮤니티	채홍근	2023.05.03	2023.06.03
버전	1.0.0	최종수정일	2023.06.07
구분	요구ID	내용	비고
User Requirement -Service	CHK-001	파티에 대여소 나타내기	파티에 대여소를 알려주는 페미지로 이동
	CHK-002	대여소의 자전거 대수 실시간 조회	
	CHK-003	자전거 모임 클릭	자전거 모임 장소, 날짜를 알려주는 페미지로 이동
	CHK-004	모임 장소 지도 표시	
	CHK-005	새로운 이벤트 개설	새로운 이벤트 개설 페미지로 이동
	CHK-006	미벤트 상세페이지 클릭	미벤트 상세페이지로 이동
	CHK-007	인증샷 갤러리 헬메로 표시	
	CHK-008	인증샷 상세페이지 클릭	인증샷 상세페이지로 이동
Non Functional Requirement -Protection &Change	CHK-009	지도에 로케이션 표시	카카오톡 API

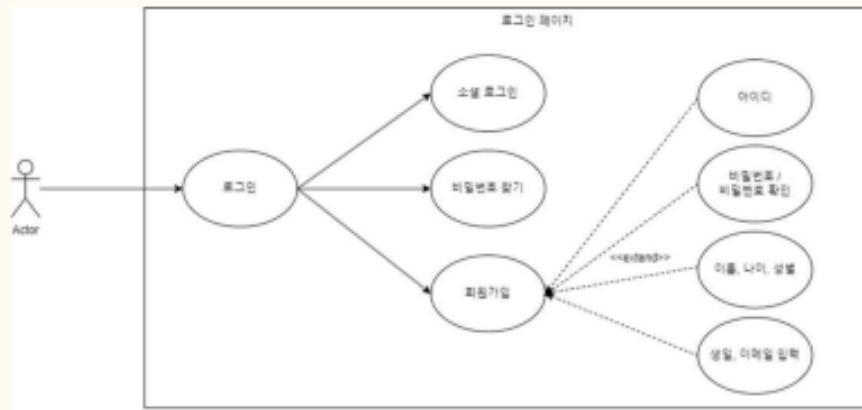
개인별 요구사항 정의서

개인별 요구사항 정의서			
프로젝트명	자전거 커뮤니티		
작성자	이종호	작성일	2023.05.02
버전	1.0.0	최종수정일	2023.06.03
구분	요구ID	내용	비고
User Requirement -Service	UDW-001	쪽지를 보내는 메시지를 보내거나 받아들이고 보내거나 받아들이고 보내는 것을	
	UDW-002	쪽지 쪽지 내용 확인	
Functional Requirement -Technology	LDW-003	쪽지를 로그인 해야만 접속 가능	
	LDW-004	쪽지 등록하기	
	LDW-005	쪽지 삭제하기 일을 삭제 가능	
Non Functional Requirement -Protection &Change	LOW-006	본인 품목으로 쪽지 및 메시지 보내는 접속 가능	

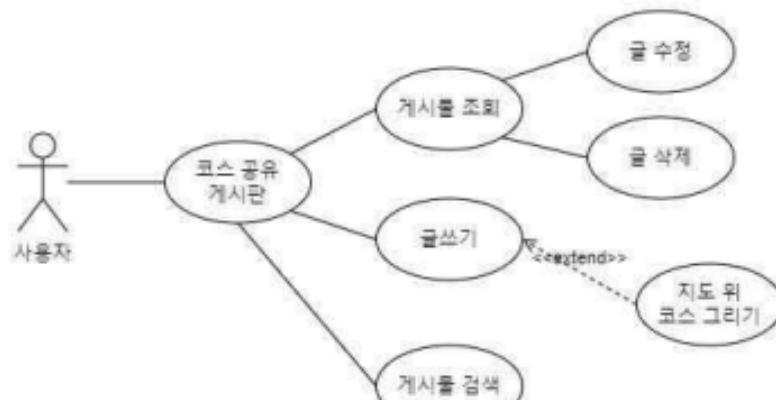
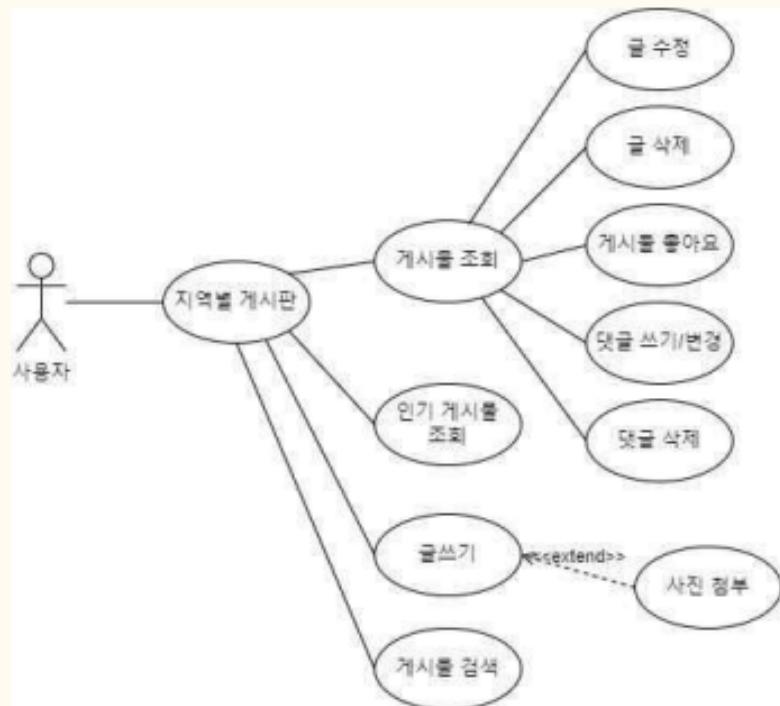
개인별 요구사항 정의서

개인별 요구사항 정의서			
프로젝트명	자전거 커뮤니티		
작성자	조창현	작성일	2023.05.03
버전	1.0.0	최종수정일	2023.06.02
구분	요구ID	내용	비고
User Requirement -Service	JCH-001	하루만 이용할 이용권 구매로 이용	
	JCH-002	점기적으로 이용할 이용권 구매로 이용	
	JCH-003	본인의 구매내역 확인	
Functional Requirement -Technology	JCH-003	이용약관	결제 이용약관 확인
	JCH-004	결제하기	결제창으로 넘어가기
Non Functional Requirement -Protection &Change	JCH-001	결제방법	다양한 결제방법 본인 선택

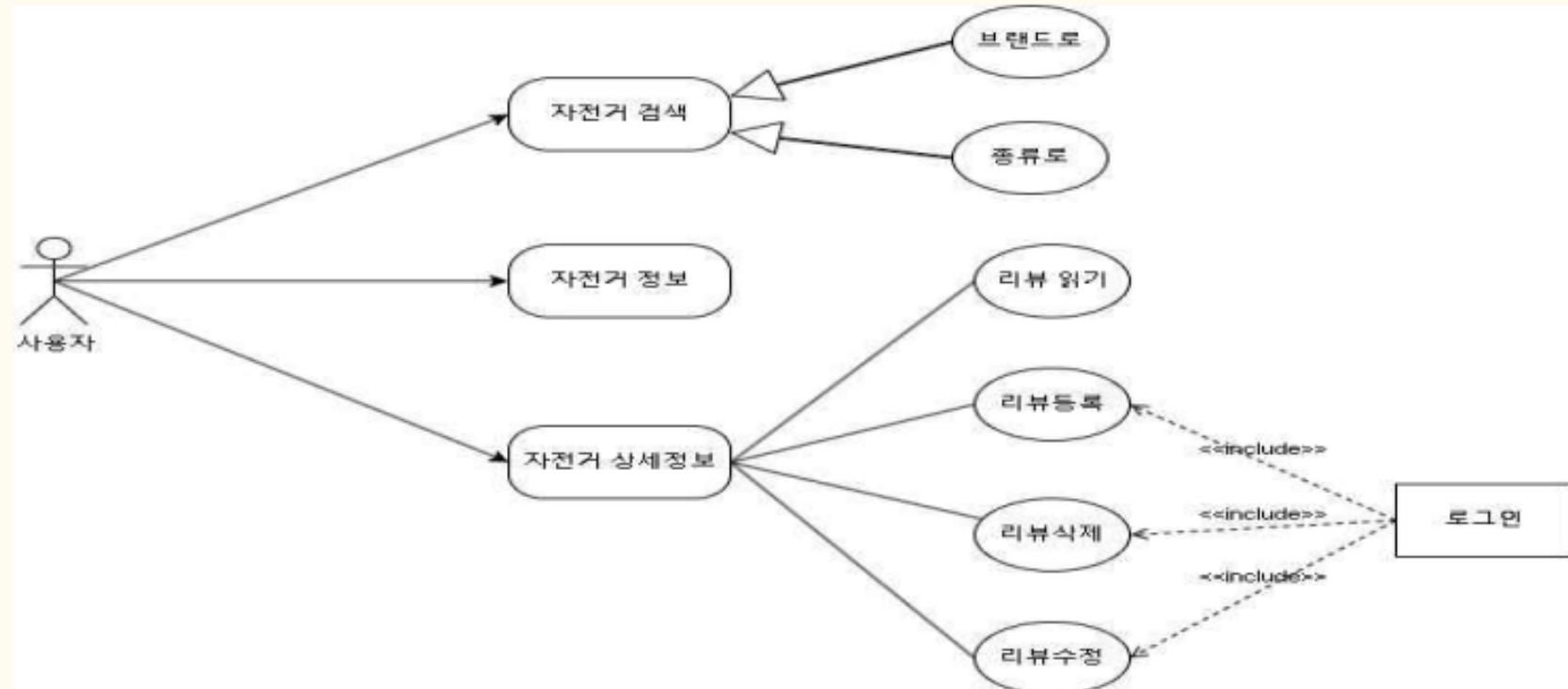
USE CASE (로그인)



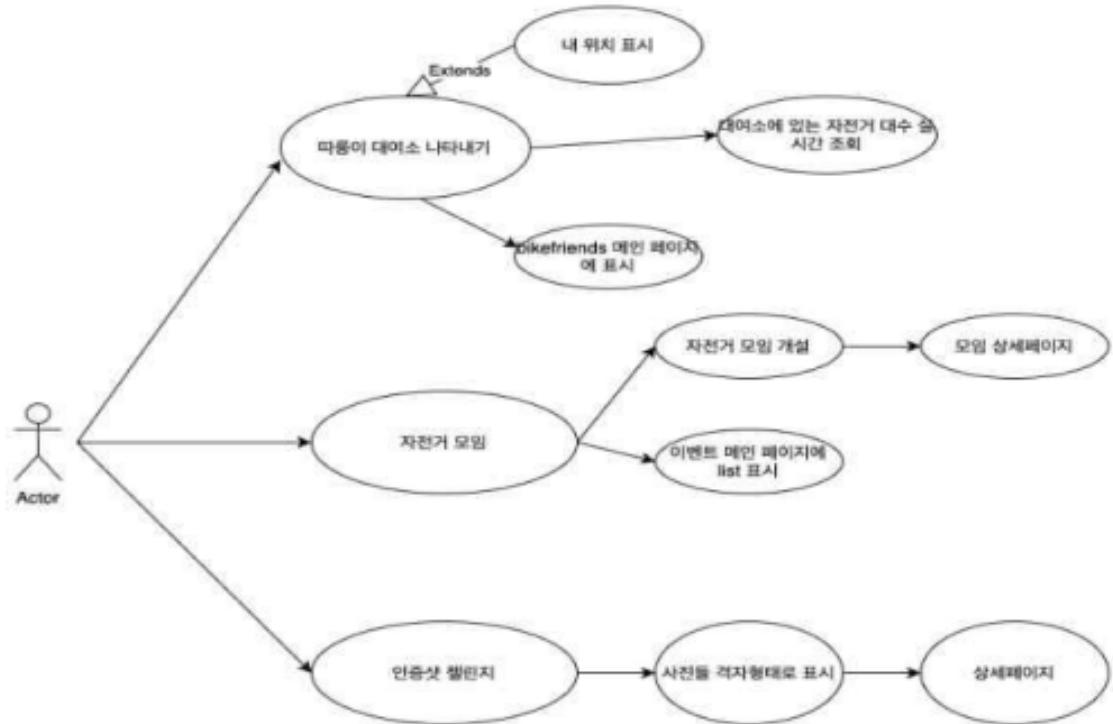
USE CASE (게시판)



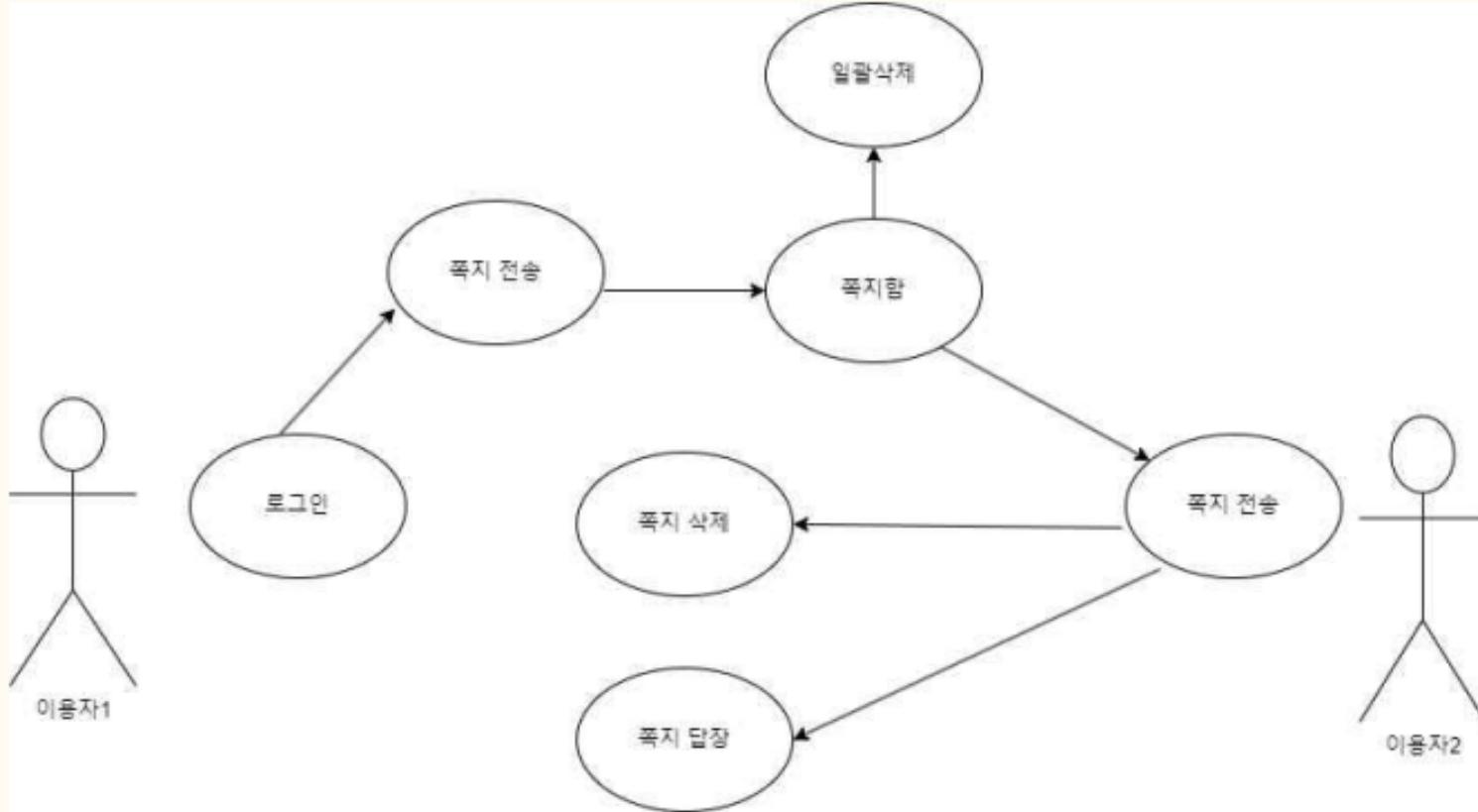
USE CASE (자전거정보)



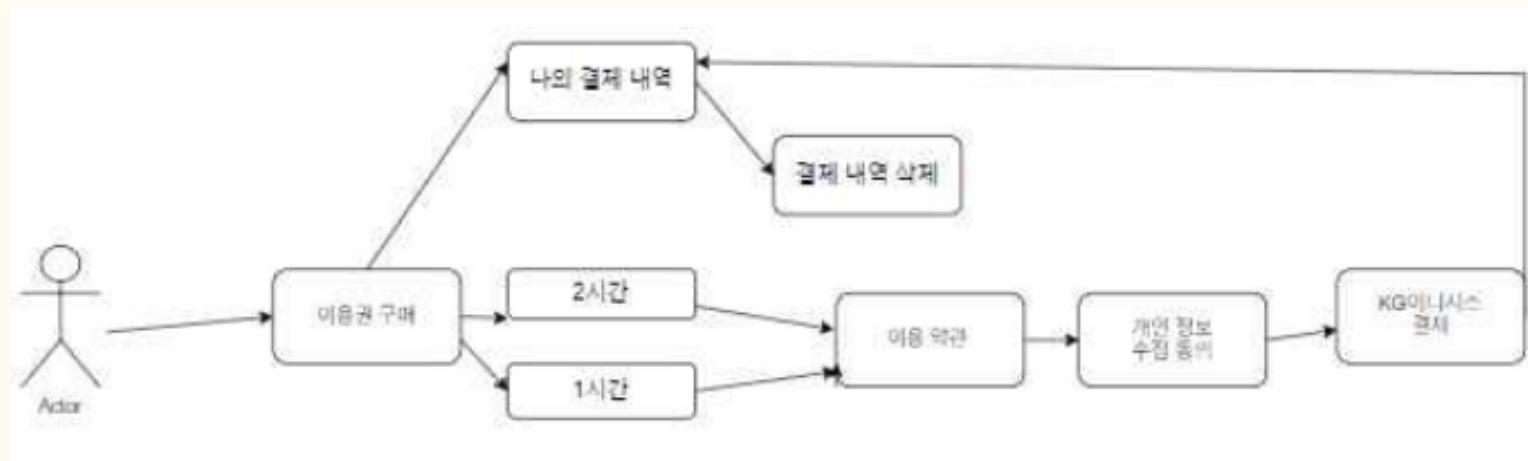
USE CASE (이벤트)



USE CASE (쪽지)



USE CASE (이용권)



테이블 정의서 (로그인)

Table_01

테이블ID		member		작성자	서예주	
테이블설명		회원정보, 회원가입, 로그인 관리 테이블				
번호	칼럼ID	형식	NULL	KEY	내용	비고
1	member_id	VARCHAR	Not Null	PK	회원 아이디	
2	member_pw	VARCHAR	Not Null		회원 패스워드	
3	member_name	VARCHAR	Not Null		회원 이름	
4	member_gender	VARCHAR			성별	
5	member_birthday	DATE	Not Null		생일	

6	member_email	VARCHAR			이메일	
7	member_age	VARCHAR			나이	생일 데이터에 기반해서 나이 입력
8	member_img	VARCHAR			프로필 사진	
9	member_bio	VARCHAR			소개말	
10	member_address	VARCHAR			주소	
11	member_phone	INT			전화번호	

테이블 정의서 (게시판)

Table_LYS_01

테이블ID		bbs		작성자	이연수	
테이블설명		게시물 정보				
번호	칼럼ID	형식	NULL	KEY	내용	비고
1	bbs_no	int	Not Null	PK	게시물 번호 auto increment	
2	content	varchar	Not Null		내용	
3	writer	varchar	Not Null	FK	작성자 id	
4	date	datetime	Not Null		작성 날짜 default now()	
5	group_no	int	Not Null	FK	그룹 번호	
6	img	varchar			첨부파일 이미지	
7	hit	int	Not Null		조회수	

Table_LYS_02

테이블ID		bbsexpaly		작성자	이연수	
테이블설명		게시물 댓글				
번호	칼럼ID	형식	NULL	KEY	내용	비고
1	reply_no	int	Not Null	PK	댓글 번호 auto increment	
2	bbs_no	int	Not Null	FK	게시물 번호	
3	content	varchar	Not Null		댓글 내용	
4	writer	varchar	Not Null	FK	작성자 id	
5	date	datetime	Not Null		작성 날짜 default now()	

Table_LYS_03

테이블ID		bbsslike		작성자	이연수	
테이블설명		게시물 좋아요				
번호	칼럼ID	형식	NULL	KEY	내용	비고
1	like_no	int	Not Null	PK	번호 auto increment	
2	bbs_no	int	Not Null	FK	게시물 번호	
3	member_id	varchar	Not Null	FK	좋아요 누른 사용자	

Table_LYS_04

테이블ID		bbsgroup		작성자	이연수	
테이블설명		지역별 그룹				
번호	칼럼ID	형식	NULL	KEY	내용	비고
1	group_no	int	Not Null	PK	그룹 번호	
2	group_name	varchar	Not Null		그룹 이름 (지역)	

Table_LYS_05

테이블ID		course		작성자	이연수	
테이블설명		포스터 큐레이션				
번호	칼럼ID	형식	NULL	KEY	내용	비고
1	course_no	int	Not Null	PK	코스 번호 auto increment	
2	rate	int	Not Null		별점	
3	content	varchar			게시물 내용	
4	title	varchar	Not Null		제목	
5	writer	varchar	Not Null	FK	작성자	
6	date	datetime	Not Null		작성 날짜 default now()	
7	line_bt	varchar	Not Null		주소	
8	line_img	varchar	Not Null		경로	

테이블 정의서 (자전거정보)

Table_SBS_01						
테이블 ID		bike		작성자	신법식	
테이블설명		자전거 테이블				
번호	칼럼 ID	형식	NULL	KEY	내용	비고
1	bike_no	INT	Not Null	PK	자전거 번호	auto increment
2	bike_brand	VARCHAR			자전거 브랜드명	
3	bike_name	VARCHAR			자전거 제품명	
4	bike_categ ory	VARCHAR			자전거 종류	
5	bike_img	VARCHAR			자전거 이미지	
6	bike_detail	VARCHAR			자전거 상세정보	

Table_SBS_02						
테이블 ID		bikereply		작성자	신법식	
테이블설명		자전거 리뷰				
번호	칼럼 ID	형식	NULL	KEY	내용	비고
1	bikereply_no	INT	Not Null	PK	리뷰번호	auto *** increment
2	bikereply_writer	VARCHAR	Not Null	FK	리뷰작성자ID	
3	bikereply_content	VARCHAR			리뷰내용	
4	bike_no	VARCHAR	Not Null	FK	자전거번호	
5	date	VARCHAR			입력시간	default: now()
6	bikereply_r ating	VARCHAR			리뷰별점	
7	bikereply_writername	VARCHAR			작성자이름	

테이블 정의서 (이벤트)

Table_CHK_01						
테이블ID		event		작성자	체홍균	
테이블설명		이벤트 정보				
번호	칼럼ID	형식	NULL	KEY	내용	비고
1	no	INT	Not Null	PK	게시글번호	
2	event_id	VARCHAR			회원 아이디	
3	event_location	VARCHAR			이벤트 장소	
4	event_date	DATE			이벤트날짜	
5	event_img	VARCHAR			이미지	
6	event_size	INT			이벤트 수용인원	

Table_CHK_02						
테이블ID		eventchallenge		작성자	체홍균	
테이블설명		인증샷 사진들을 남기는 게시판				
번호	칼럼ID	형식	NULL	KEY	내용	비고
1	no	INT	Not Null		게시글번호	
2	content	VARCHAR	Not Null		게시글 간단 설명	
3	writer	VARVHAR	Not Null		글쓴이	
4	date	DATETIME	Not Null		글쓴날짜	
5	group_no	INT	Not Null			
6	img	VARCHAR			이미지	
7	hit	INT	Not Null		좋아요표시	

테이블 정의서 (쪽지)

Table_LDW_01

테이블ID	note	작성자	이돈우			
테이블설명	쪽지 내용과 관련된 테이블					
번호	칼럼ID	형식	NULL	KEY	내용	비고
1	no	int	Not Null	PK	쪽지 번호	auto increment
2	title	VARCHAR	Not Null		제목	
3	content	VARCHAR	Not Null		내용	
4	receiver	VARCHAR	Not Null		받은사람	
5	sender	VARCHAR	Not Null		보낸사람	
6	date	datetime	Not Null		날짜	

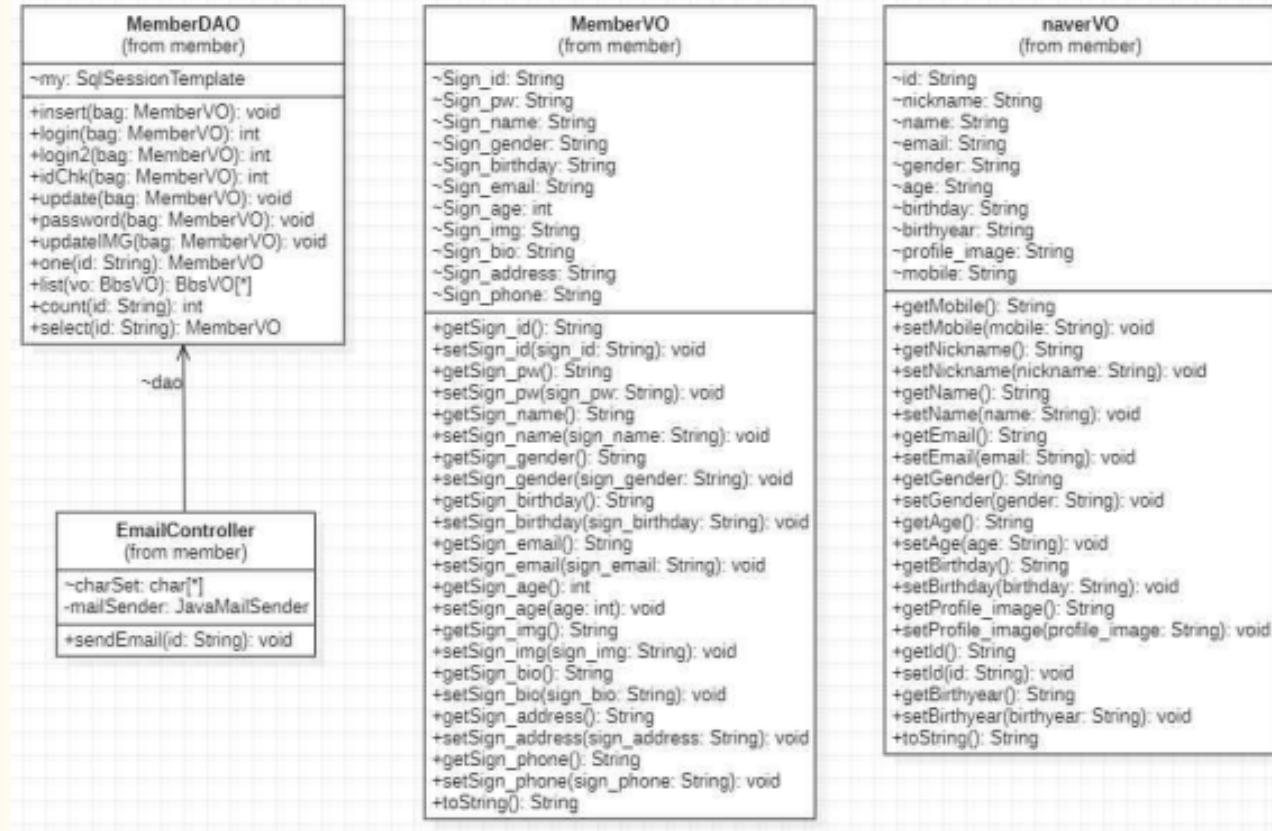
테이블 정의서 (이용권)

Table_01						
테이블ID		payment1		작성자	조항현	
테이블설명		이용권 구매 후 구매 금액과 사용자 개인별 데이터 저장				
번호	칼럼ID	형식	NULL	KEY	내용	비고
1	pay_no	int	Not Null	PK	구매자의 데이터 고유번호	autoincrement
2	id	VARCHAR		FK (member 테이블의 id 참조)	구매자 id	
3	money	VARCHAR			구매금액	
4	name	VARCHAR			구매자 이름	
5	tel	VARCHAR			구매자 전화번호	
6	email	VARCHAR			구매자 이메일	

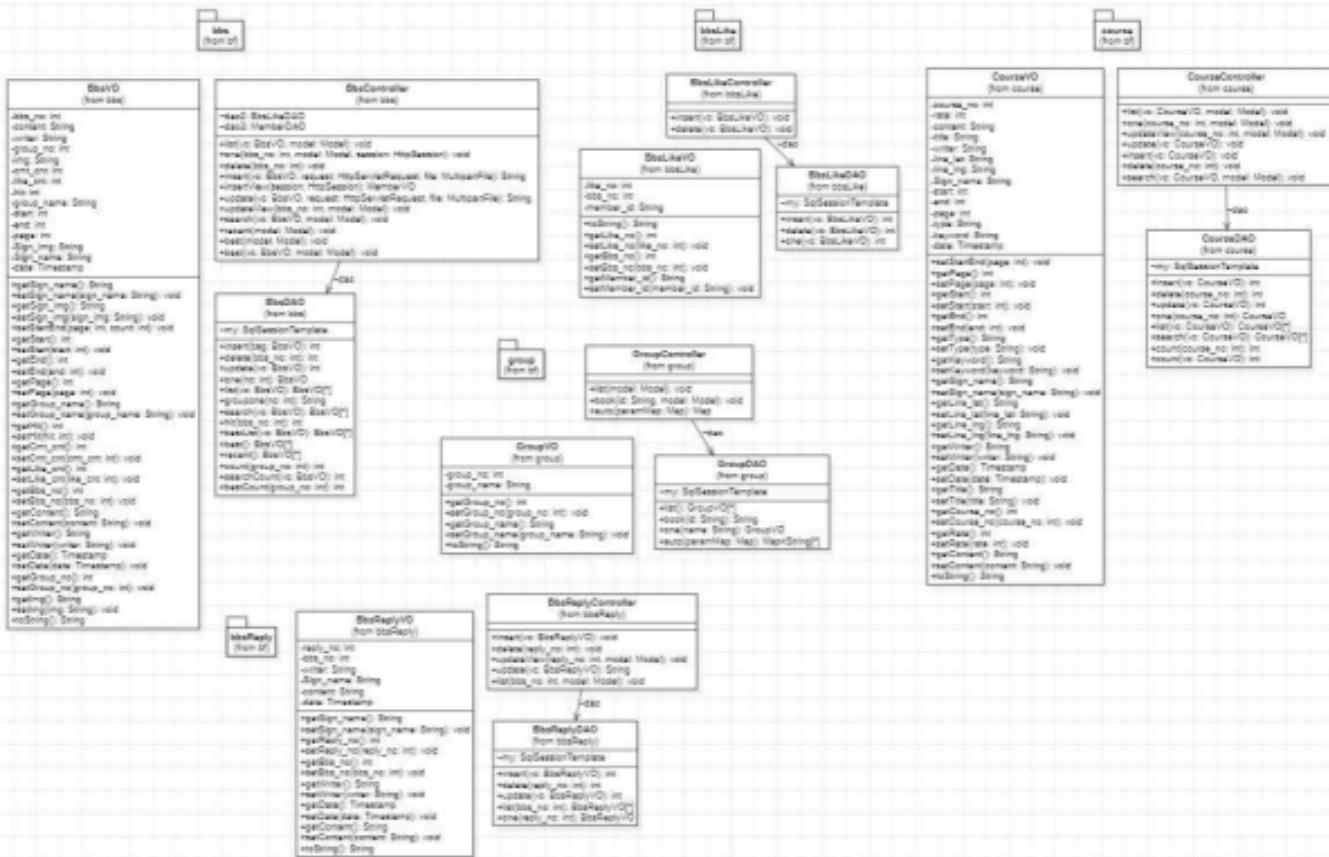
Project Tree



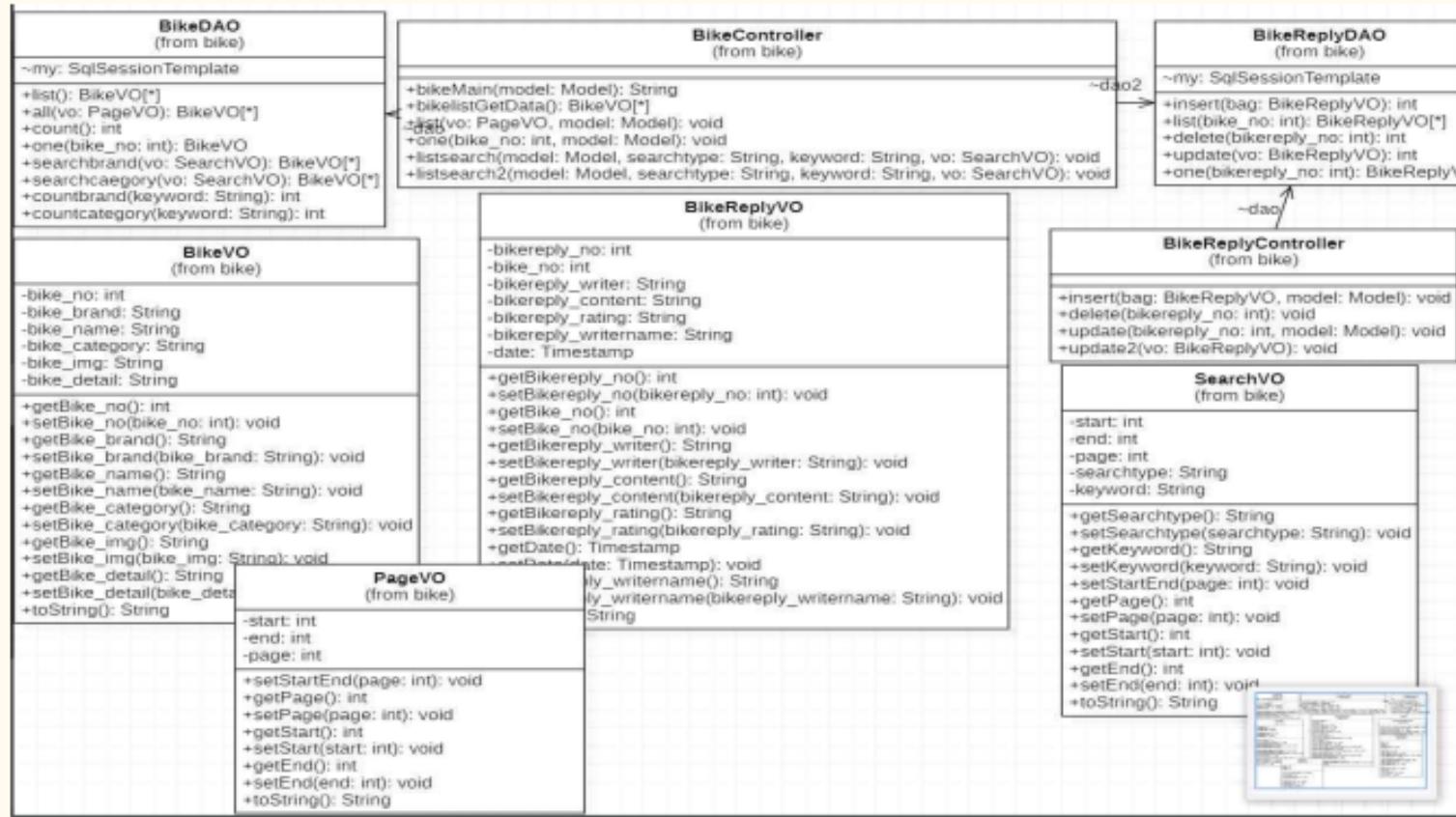
UML Class Diagram 로그인



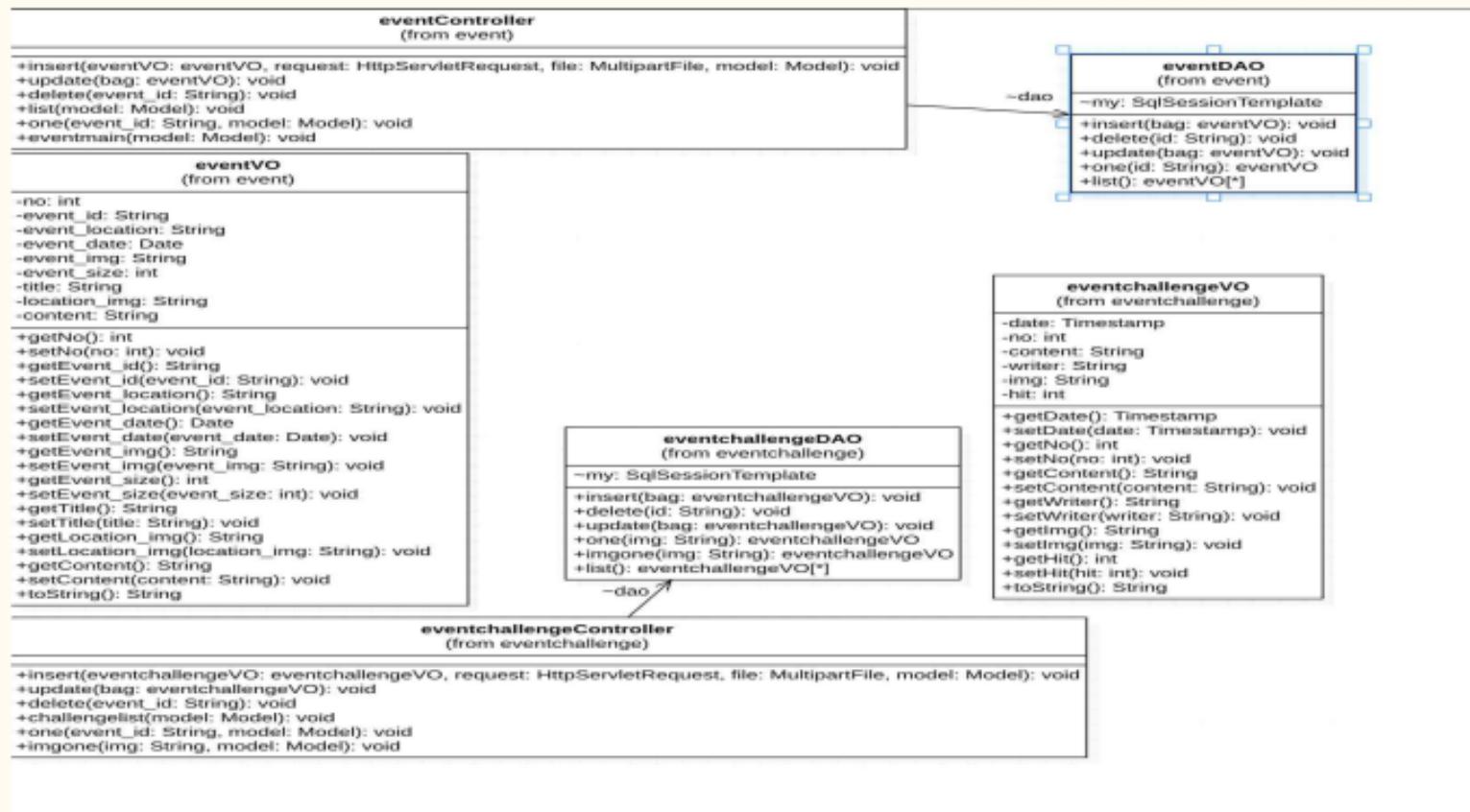
UML Class Diagram 게시판



UML Class Diagram 자전거정보



UML Class Diagram 이벤트



UML Class Diagram 쪽지

NoteController
(from note)

```
+insert(bag: NoteVO, session: HttpSession): String  
+delete(no: int): void  
+one(no: int, model: Model): void  
+one2(no: int, model: Model): void  
+send(vo: NoteVO, model: Model): void  
+receive(vo: NoteVO, model: Model): void
```



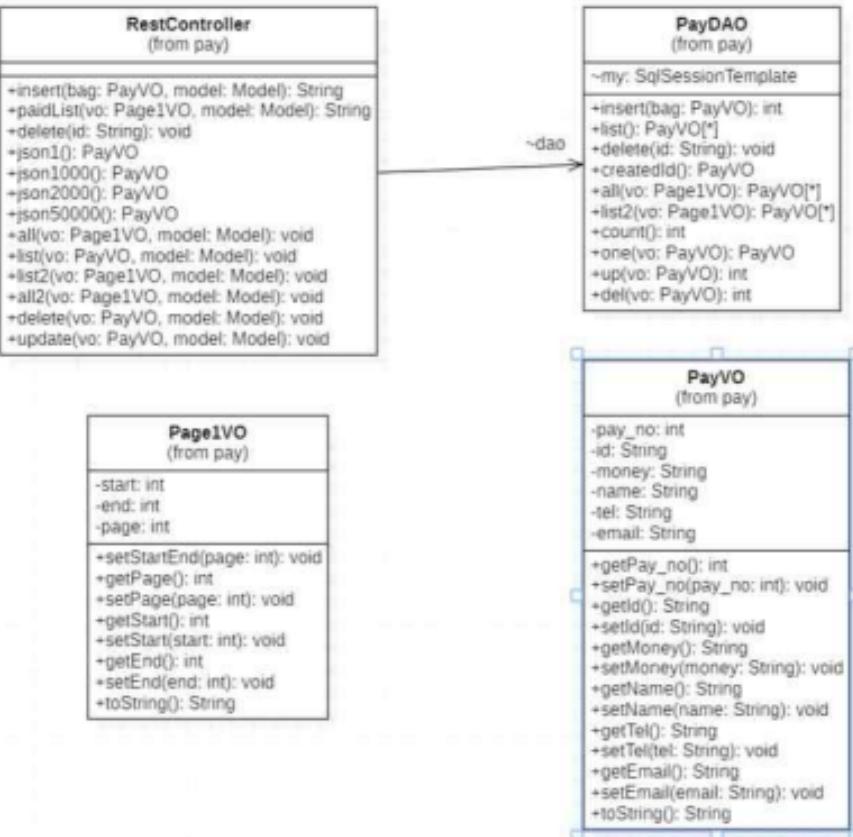
NoteDAO
(from note)

```
~my: SqlSessionTemplate  
  
+insert(bag: NoteVO): void  
+delete(no: int): void  
+send(bag: NoteVO): NoteVO[]  
+receive(bag: NoteVO): NoteVO[]  
+one(no: int): NoteVO
```

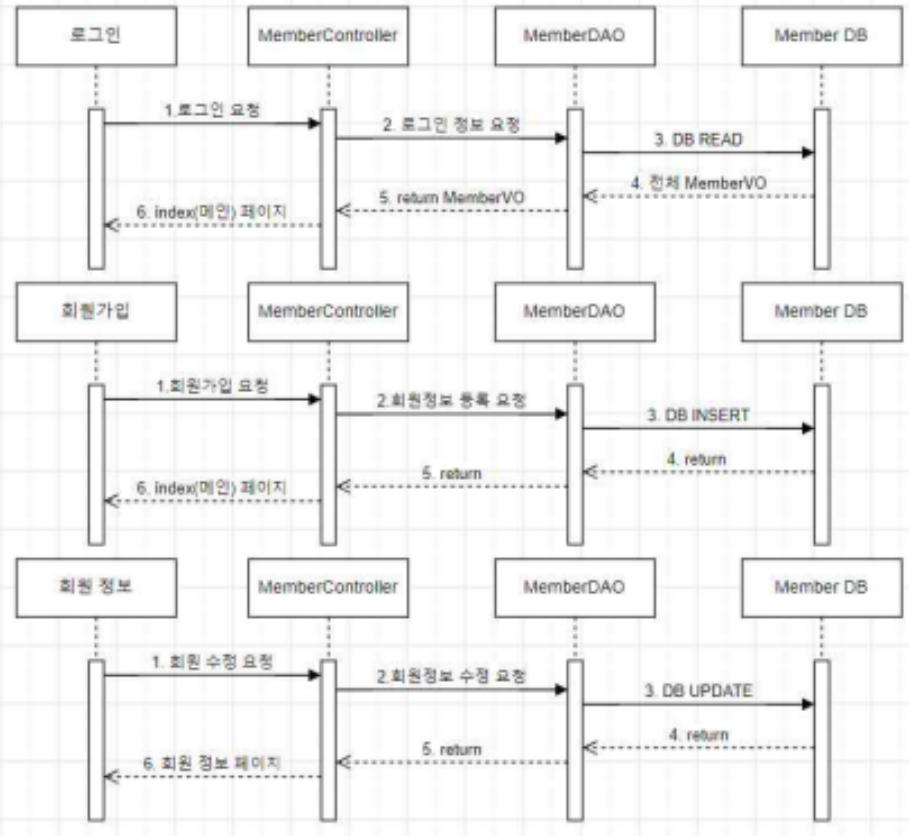
NoteVO
(from note)

```
-no: int  
-title: String  
-content: String  
-sender: String  
-receiver: String  
-date: String  
  
+getNo(): int  
+setNo(no: int): void  
+getTitle(): String  
+setTitle(title: String): void  
+getContent(): String  
+setContent(content: String): void  
+getSender(): String  
+setSender(sender: String): void  
+getReceiver(): String  
+setReceiver(receiver: String): void  
+getDate(): String  
+setDate(date: String): void
```

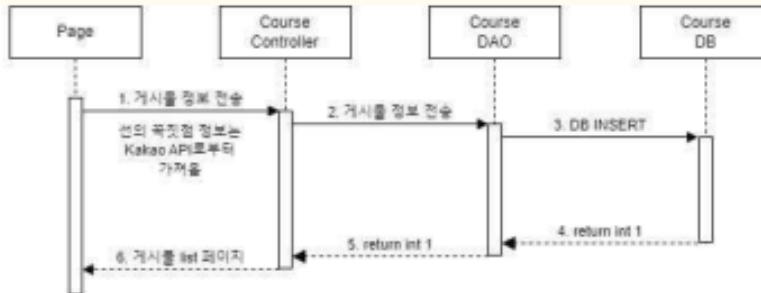
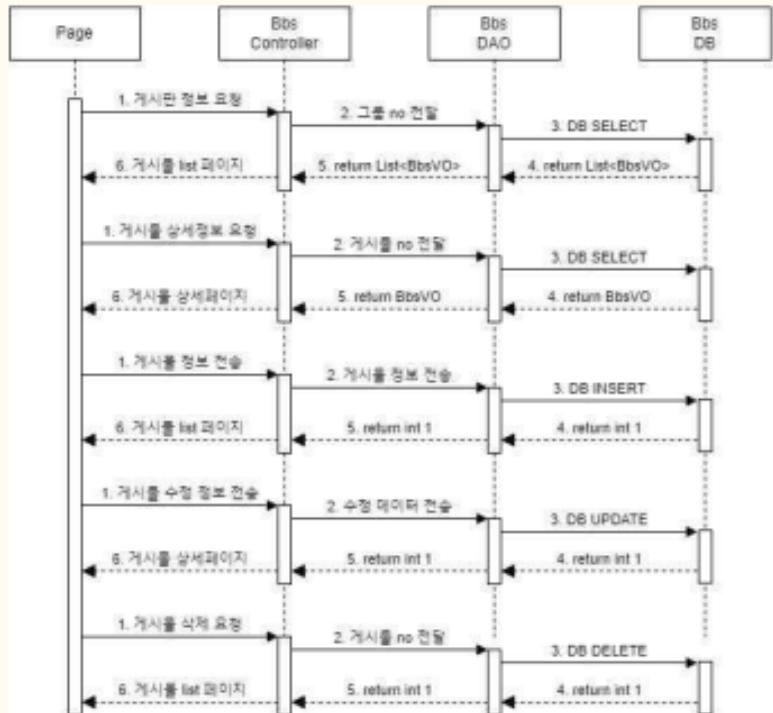
UML Class Diagram 이용권



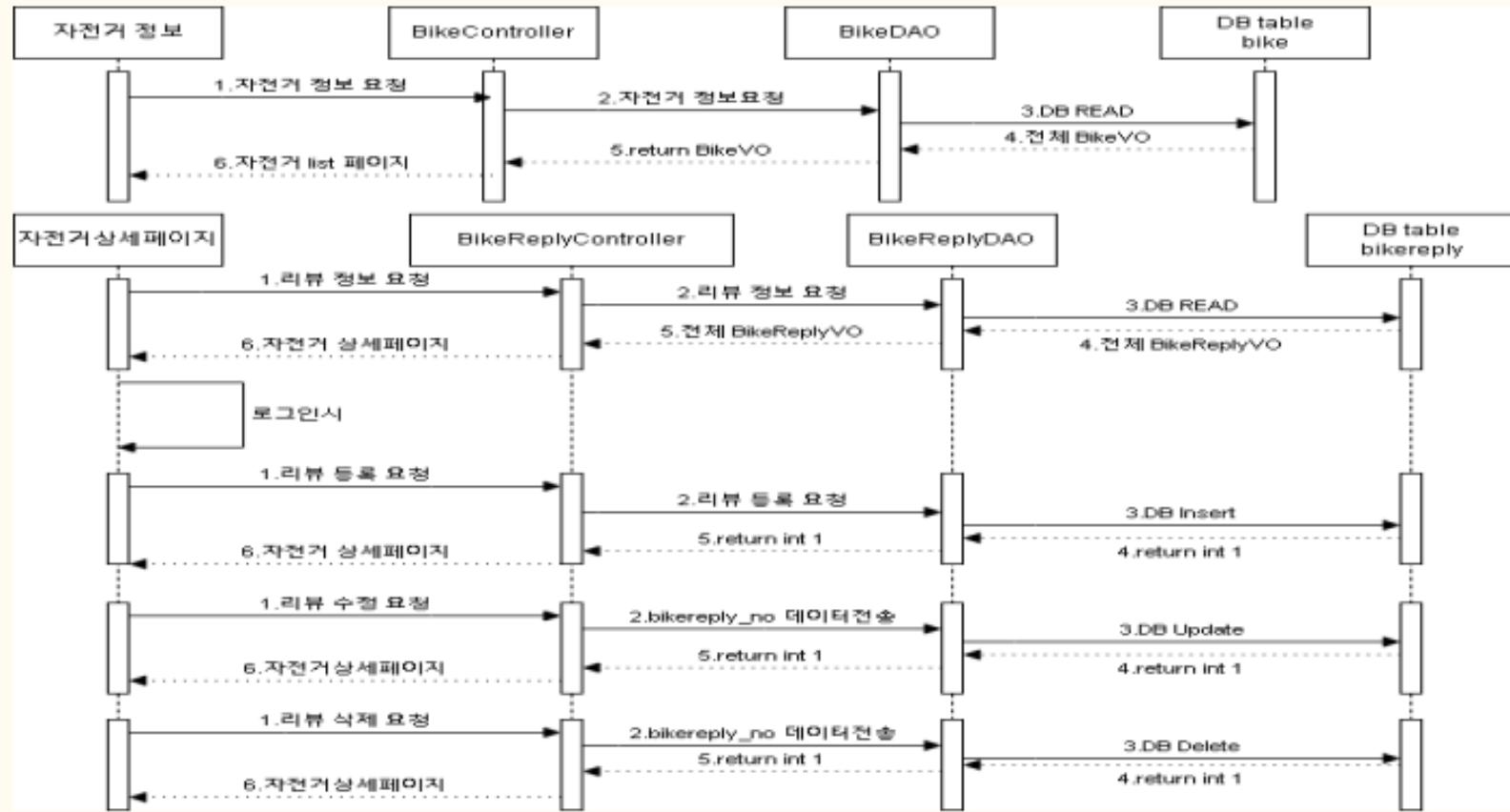
Sequence Diagram 로그인



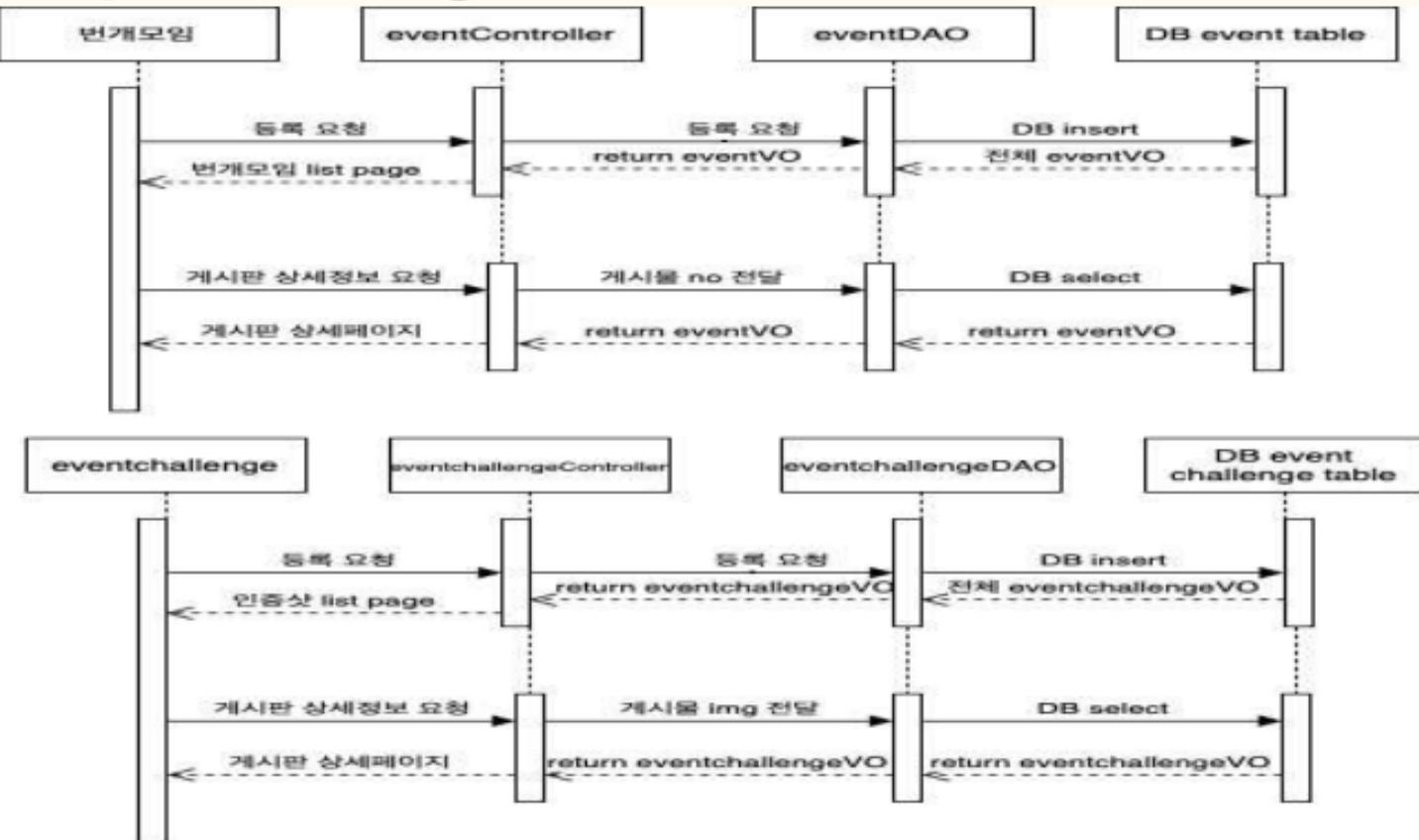
Sequence Diagram 게시판



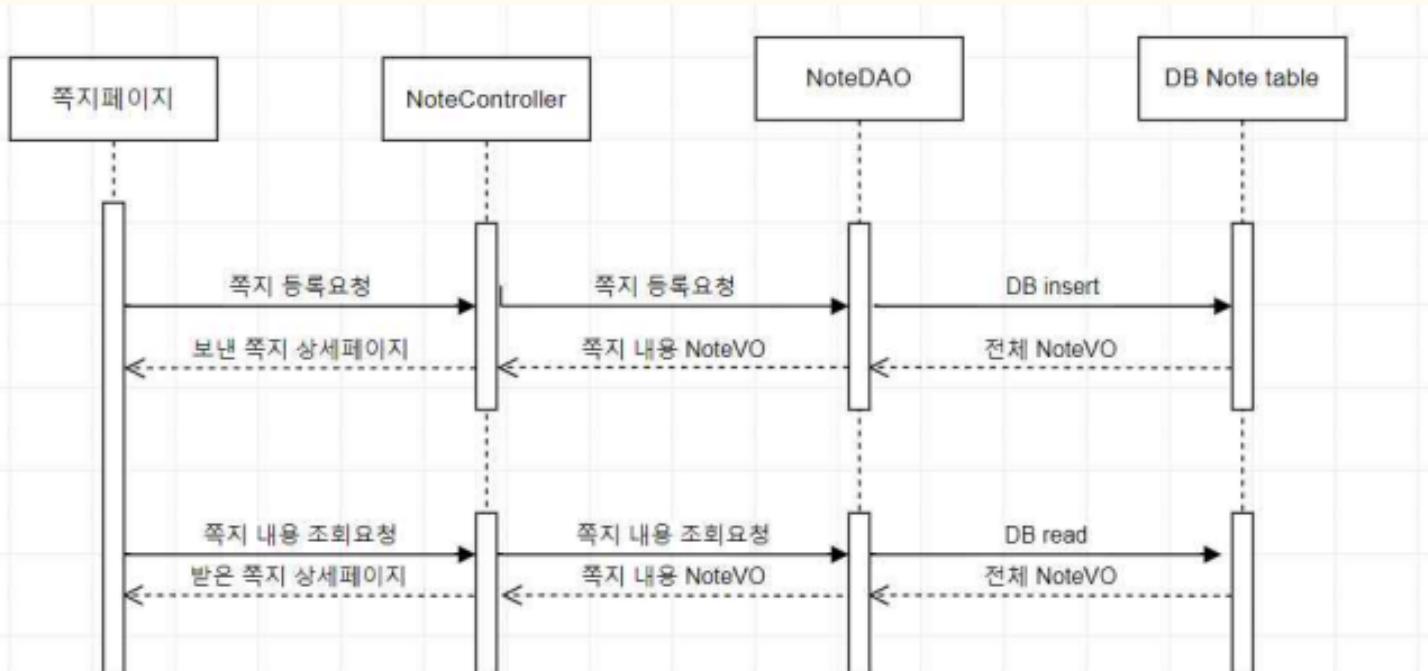
Sequence Diagram 자전거정보



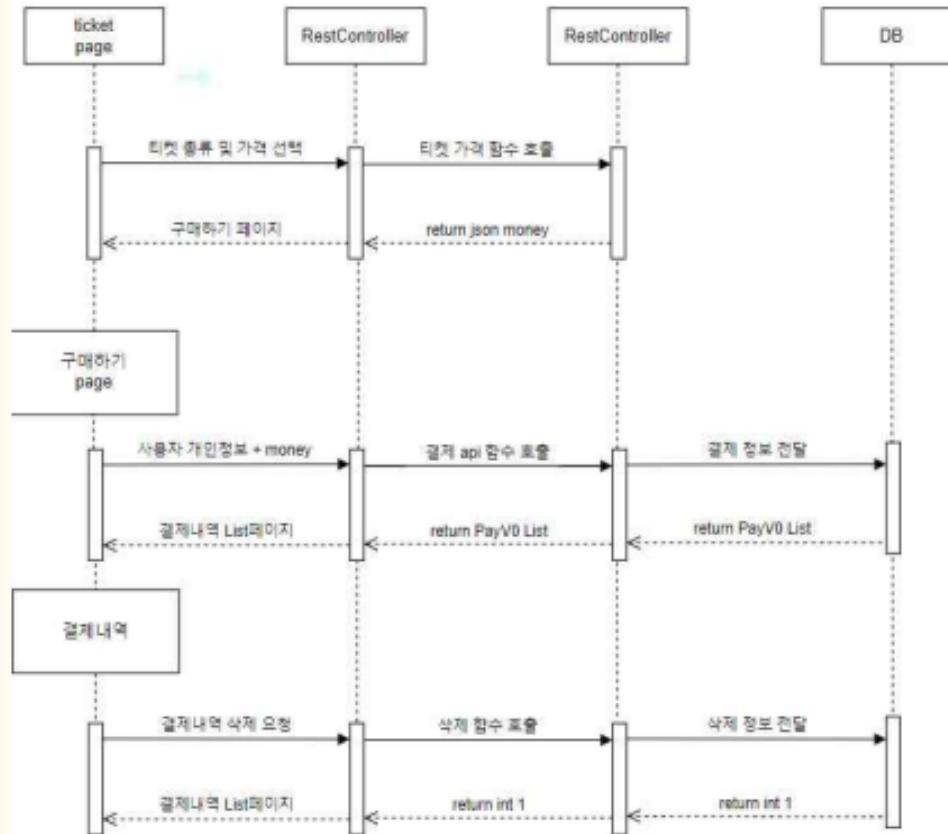
Sequence Diagram 이벤트



Sequence Diagram 쪽지



Sequence Diagram 이용권



로그인

로그인

아이디

비밀번호

로그인



로그인

비밀번호 찾기

| 회원가입



비밀번호를 찾고자 하는 ID를 입력해주세요

아이디

찾기

```
@Autowired  
private JavaMailSender mailSender;  
  
MimeMessage message = mailSender.createMimeMessage();  
message.setFrom("bikefriends2023@gmail.com");  
message.setRecipient(Message.RecipientType.TO, new InternetAddress(bag.getSign_email()));  
message.setSubject("Bike Friends 비밀번호 변경 메일입니다.");  
message.setText("비밀번호가 [" + pass + "]로 변경되었습니다. 회원 정보에서 비밀번호를 반드시 수정해주세요.");  
  
bag.setSign_pw(pass);  
dao.password(bag);  
  
mailSender.send(message);  
System.out.println("메일 전송 완료");
```

JavaMailSender 패키지를 통해
MIME 메세지(전자 우편 전송 포맷)를
쉽게 생성 후 지정된 메일로 전송한다.

```
<bean id="mailSender" class="org.springframework.mail.javamail.JavaMailSenderImpl">  
    <property name="host" value="smtp.gmail.com" />  
    <property name="port" value="587" />  
    <property name="username" value="bikefriends2023@gmail.com" />  
    <property name="password" value="[비밀번호]" />  
    <property name="javaMailProperties">  
        <props>  
            <prop key="mail.smtp.starttls.enable">true</prop>  
            <prop key="mail.smtp.auth">true</prop>  
            <prop key="mail.transport.protocol">smtp</prop>  
            <prop key="mail.debug">true</prop>  
            <prop key="mail.smtp.ssl.trust">smtp.gmail.com</prop>  
            <prop key="mail.smtp.ssl.protocols">TLSv1.2</prop>  
        </props>  
    </property>  
</bean>
```

회원가입과 마이페이지

회원가입

아이디	<input type="text"/>
비밀번호	<input type="password"/>
비밀번호 확인	<input type="password"/>
이름	<input type="text"/>
성별	<input type="text"/>
생일	연도-월-일
주소	<input type="text"/>
이메일	<input type="text"/>
전화번호	<input type="text"/>

회원가입

아이디 체크

```
<insert id="create" parameterType="memberVO">
    insert into member values
        (#Sign_id, #Sign_pw, #Sign_name, #Sign_gender, #Sign_birthday,
        #Sign_email, #Sign_age, #Sign_img, #Sign_bio, #Sign_address, #Sign_phone)
</insert>
```

```
$(function() {
    $('#b1').click(function() {
        $.ajax({
            url : "idChk",
            type : "POST",
            dataType : "text",
            data : {"Sign_id" : $('#Sign_id').val()},
            success : function(x) {
                if(x == 1){
                    alert("중복된 아이디입니다.");
                } else{
                    $('#submit').removeAttr("disabled");
                    alert("사용가능한 아이디입니다.");
                }
            },
            error:function(e) {
                alert("error: " + e);
            }
        }) //ajax
    })//b1
})/$
```

```
function isSame() {
    var password = document.getElementById('pw').value;
    var confirmPassword = document.getElementById('pwCheck').value;

    if(password != '' && password.length >= 6) {
        if(password == confirmPassword) {
            document.getElementById('same').innerHTML='비밀번호가 일치합니다.';
            document.getElementById('same').style.color='blue';
        }
        else {
            document.getElementById('same').innerHTML='비밀번호가 일치하지 않습니다.';
            document.getElementById('same').style.color='red';
        }
    } else{
        document.getElementById('same').innerHTML='비밀번호는 6자리 이상이야 합니다.';
        document.getElementById('same').style.color='red';
    }
}
```

수정하기

이름	이름	<input type="button" value="변경"/>
나이	7	
성별	M	
소개	aa	
이메일	yeju0434@naver.com	
주소	강남구	
전화번호	a	

마이 페이지



이름	이름
성별	M
나이	7
소개	aa
이메일	yeju0434@naver.com
주소	강남구
전화번호	a

```
<update id="up" parameterType="memberVO">
    update `member`
    set Sign_name = #{Sign_name}, Sign_age = #{Sign_age}, Sign_gender = #{Sign_gender},
    Sign_bio = #{Sign_bio}, Sign_email = #{Sign_email}, Sign_address = #{Sign_address},
    Sign_phone = #{Sign_phone}, Sign_birthday = #{Sign_birthday}
    where Sign_id = #{Sign_id}
```

비밀번호 변경

비밀번호	<input type="text"/>
비밀번호 확인	<input type="text"/>

```
<update id="password" parameterType="memberVO">
    update `member`
    set Sign_pw = #{Sign_pw}
    where Sign_id = #{Sign_id}
```

회원가입과 마이페이지

파일 선택 : 선택된 파일 없음

마이 페이지



이름	이름
성별	M
나이	7
소갯말	aa
이메일	yeju0434@naver.com

```
public String upload(String imageUrl, MultipartFile file, MultipartHttpServletRequest request) throws Exception {
    MemberVO bag = new MemberVO();

    if (!file.isEmpty()) {
        String savedName = file.getOriginalFilename();
        String uploadPath = request.getSession().getServletContext().getRealPath("resources/upload");
        System.out.println(uploadPath);
        File target = new File(uploadPath + "/" + savedName);
        file.transferTo(target);

        bag.setSign_img("/bf/resources/upload/" + savedName);
    } else {
        bag.setSign_img(imageUrl);
    }
}
```

소셜 로그인

```
<script type="text/javascript">
    var naver_id_login = new naver_id_login([REDACTED] API Client ID
        "http://localhost:8898/bf/member/allback.jsp");
    //alert(naver_id_login.oauthParams.access_token);
    //naver_id_login.setPopup();
    naver_id_login.get_naver_userprofile("naverSignInCallback()");
    // 네이버 사용자 프로필 조회 이후 프로필 정보를 처리할 callback function
    function naverSignInCallback() {
        id = naver_id_login.getProfileData('id') // 고유 식별번호
        nickname = naver_id_login.getProfileData('nickname') // 별명
        name = naver_id_login.getProfileData('name') // 이름
        email = naver_id_login.getProfileData('email') // 메일 주소
        gender = naver_id_login.getProfileData('gender') // 성별 F여성 M남성 U학번불가
        age = naver_id_login.getProfileData('age') // 연령대
        birthday = naver_id_login.getProfileData('birthday') // 생일 MM-DD
        birthyear = naver_id_login.getProfileData('birthyear') // 출생연도
        profile_image = naver_id_login.getProfileData('profile_image') // 프로필 사진 URL
        mobile = naver_id_login.getProfileData('mobile') // 휴대전화번호

        location.href = "naverLogin?nickname=" + nickname
            + "&name=" + name + "&email=" + email + "&gender=" + gender
            + "&age=" + age + "&birthday=" + birthday
            + "&profile_image=" + profile_image
            + "&id=" + id
            + "&birthyear=" + birthyear
            + "&mobile=" + mobile
    }
<script type="text/javascript">
    var naver_id_login = new naver_id_login([REDACTED] API Client ID
    var state = naver_id_login.getUniqState();
    naver_id_login.setButton("white", 2,40);
    naver_id_login.setDomain("http://localhost:8898/bf");
    naver_id_login.setState(state);
    //naver_id_login.setPopup();
    naver_id_login.init_naver_id_login();
</script>
```

The screenshot shows the NAVER login interface. At the top, there are three tabs: 'ID 로그인' (selected), '일회용 번호', and 'QR코드'. Below the tabs, a message says '네이버에 로그인하여 test 서비스를 이용하실 수 있습니다.' (You can log in to NAVER to use the test service). A note below states '공용 PC에서 사용하시는 경우 보안을 위해 서비스 네이버에서도 반드시 로그아웃해 주세요' (When using a public computer, log out from both services). On the right, there is a '네이버 로그인' button. The main form has fields for '아이디' (ID) and '비밀번호' (Password), with a '로그인 상태 유지' (Remember me) checkbox. A large green '로그인' (Login) button is centered. To the right of the form, there is a sidebar with user information: 'test' (아이디), '비밀번호' (Password), and '전체 동의하기' (Agree to all). Below this, it says 'test에서 [REDACTED] 회원님의 개인정보에 접근합니다.' (test is accessing your personal information). It also lists '이름' (Name), '생년' (Birth year), '성별' (Gender), and '생일' (Birth date). At the bottom, there are buttons for '취소' (Cancel) and '동의하기' (Agree).

지역별 게시판

회원정보 중 주소와 일치하는 게시판 나
타내기.ajax 이용)

GroupController

```
// 내 지역 게시판
@RequestMapping("bbs/book")
public void book(String id, Model model) {
    //회원정보에서 주소 가져오기
    String name = dao.book(id);
    // 그룹 정보 가져오기
    GroupVO vo = dao.one(name);
    model.addAttribute("vo", vo);
}
```

GroupDAO

```
public String book(String id) {
    String address = my.selectOne("group.book", id);
    return address;
}

//그룹이름, 주소 비교
public GroupVO one(String name) {
    GroupVO vo = my.selectOne("group.one", name);
    return vo;
}
```

The screenshot shows a web application interface for a local bike board. At the top, there's a green header bar with the site name 'Bike Friends' and navigation links for '커뮤니티', '자전거정보', '이벤트', and '이용권'. On the right side of the header, there are links for 'apple' and '로그아웃'. Below the header, the main content area has a green background. In the center, there's a map with coordinates 26.02° and 27.66° / 23.89°. To the right of the map is a section titled '게시판 목록' (Board List). This section includes a '전체 게시판 목록' (Full Board List) link and a '내 지역 게시판' (My Local Board) link, which is highlighted with a red box. Below these are several buttons for different districts: 강남구, 강동구, 강북구, 강서구, 관악구, 광진구, 구로구, 금천구, 노원구, 도봉구, 등대문구, and 동작구. At the bottom left, there's a section titled 'GroupMapper' with two database query snippets:

```
<select id="one" parameterType="String" resultType="groupVO">
    select * from `bbsgroup` where group_name = #{group_name}
</select>

<select id="book" parameterType="String" resultType="String">
    select Sign_address from member
    where Sign_id = #{Sign_id}
</select>
```

지역별 게시판

23.66°
23.66° / 20.69°

강남구 BEST

전체 게시판 목록

강남구

코스 공유하기



admin

자전거

2023-05-30

0 4 2



연수

oooo

2023-05-30

0 1 1



aaa

내일 자전거 모임 있음

2023-05-18

0 4 4



ccc

자전거

2023-05-11

0 0 3

각 그룹에 해당하는 게시물 가져오기(좋아요수, 댓글수 포함)

```
<select id="list" parameterType="bbsVO" resultType="bbsVO">
    select * from (
        select row_number() over(order by date desc) as no, b.* , m.* ,
        (select count(*) from bbslike where bbs_no = b.bbs_no) as like_cnt,
        (select count(*) from bbsreply where bbs_no = b.bbs_no) as cmt_cnt
        from bbs b, member m
        where b.writer = m.Sign_id
        and group_no = #{group_no}
    ) as b2
    where b2.no between #{start} and #{end}
</select>
```

forEach 이용하여 list 생성

```
<c:forEach var="vo" items="#{list}">
    <table>
        <tr class="post">
            <td class="profile" rowspan="3"></td>
            <td class="bbswriter"><b>${vo.sign_name}</b></td>
            <td class="bbsdate"><fmt:formatDate value="${vo.date}" pattern="yyyy-MM-dd" /></td>
        </tr>
        <tr>
            <td class="bbscon" colspan="2">
                <a href="one?bbs_no=${vo.bbs_no}">${vo.content}</a>
                <c:if test="${!empty vo.img}">
                    <i class="fa-regular fa-image"></i>
                </c:if>
            </td>
        </tr>
        <tr>
            <td class="bbsinfo" colspan="2">
                <i class="fa-regular fa-comment"></i> ${vo.cnt_cmt}
                <i class="fa-regular fa-heart"></i> ${vo.like_cnt}
            </td>
        </tr>
    </table>
    <br>
</c:forEach>
```

게시물 좋아요

강남구

수정

삭제

bb

2023-05-09 10:34:13

5/9 자전거 모임

② 7


댓글 내용 입력

작성

댓글 0개 새로고침

☰ 목록

bbs/one.jsp

```
/* 좋아요 */
like = ${like};
like_cnt = ${bbsVO.like_cnt};
if(like >= 1) {
    $('#like').html('<i class="fa-solid fa-heart" style="color: red;"></i>');
}
```

```
// 좋아요 눌렀는지 확인
String id = (String) session.getAttribute("id");
BbsLikeVO vo2 = new BbsLikeVO();
vo2.setBbs_no(bbs_no);
vo2.setMember_id(id);
int count = dao2.one(vo2);
model.addAttribute("like", count);      BbsController
```

기존에 좋아요 클릭했으면 model을 이용하여 like로 1을 넘겨줌

BbsLikeController

```
public int one(BbsLikeVO vo) {
    int count = my.selectOne("bbslike.findLike", vo);
    return count;
```

bbsLikeMapper

```
<select id="findLike" parameterType="bbslikeVO" resultType="int">
    select count(*) from bbslike
    where bbs_no = #{bbs_no} and member_id = #{member_id}
</select>
```

게시물 좋아요

좋아요 취소(delete)
좋아요(insert) 실행

```
id = '${id}';  
if (id != '') {  
    $('#like').click(function() {  
        if (like >= 1) {  
            $.ajax({  
                url: "likeDown",  
                data: {  
                    bbs_no: ${bbsVO.bbs_no},  
                    member_id: '${id}'  
                },  
                success: function() {  
                    like = 0;  
                    $('#like').html('<i class="fa-regular fa-heart"></i>');  
                    like_cnt = like_cnt - 1;  
                    $('#like_cnt').html(like_cnt);  
                }  
            })  
        }  
    })  
  
} else {  
    $.ajax({  
        url: "likeUp",  
        data: {  
            bbs_no: ${bbsVO.bbs_no},  
            member_id: '${id}'  
        },  
        success: function() {  
            like = 1;  
            $('#like').html('<i class="fa-solid fa-heart" style="color: red;"></i>');  
            like_cnt = like_cnt + 1;  
            $('#like_cnt').html(like_cnt);  
        }  
    })  
}
```

BbsLikeController

```
@RequestMapping("bbs/likeUp")  
public void insert(BbsLikeVO vo) {  
    dao.insert(vo);  
}  
  
@RequestMapping("bbs/likeDown")  
public void delete(BbsLikeVO vo) {  
    dao.delete(vo);  
}
```

```
<insert id="likeUp" parameterType="bbslikeVO">  
    insert into bbslike (bbs_no, member_id)  
    values(${bbs_no}, ${member_id})  
</insert>  
  
<delete id="likeDown" parameterType="bbslikeVO">  
    delete from bbslike  
    where bbs_no = ${bbs_no} and member_id = ${member_id}  
</delete>
```

bbsLikeMapper

지역별 인기게시판

강남구 BEST

글쓰기



aaa

2023-05-18

내일 자전거 모임 있음

© 116 ⚪ 4 ⚪ 4



ccc

2023-05-11

자전거 🚲

© 79 ⚪ 0 ⚪ 3



ccc

2023-05-10

화요일 자전거 타기

© 86 ⚪ 5 ⚪ 4



aaa

2023-05-09

자전거 🚲

© 120 ⚪ 3 ⚪ 4

```
@RequestMapping("bbs/bestlist")
public void best(BbsVO vo, Model model) {
    vo.setStartEnd(vo.getPage(), 10);
    List<BbsVO> list = dao.bestList(vo);
    model.addAttribute("list", list);

    int count = dao.bestCount(vo.getGroup_no());
    int pages = 0;
    if (count % 10 == 0) {
        pages = count / 10;
    } else {
        pages = count / 10 + 1;
    }
    model.addAttribute("pages", pages);

    // 그룹이름
    vo.setGroup_name(dao.groupone(vo.getGroup_no()));
    model.addAttribute("bbsVO", vo);
}
```

BbsController

각 조건에 맞는 게시물만
가져오기

- 조회수 30, 좋아요수 3 이상

```
<!-- 각 게시판 인기 게시물 -->
<select id="bestList" parameterType="bbsVO" resultType="bbsVO">
    select * from (
        select *, row_number() over(order by date desc) as no,
        (select count(*) from bbslike where bbs_no = b.bbs_no) as like_cnt
        (select count(*) from bbsreply where bbs_no = b.bbs_no) as cmt_cnt
        from bbs b, member m
        where b.writer = m.Sign_id
        and group_no = #{group_no}
    ) as b2
    where hit >= 30 and like_cnt >= 3
    and b2.no between #{start} and #{end}
</select>
```

코스 공유 - 글쓰기

그리는 선의 정보(위도, 경도)를 변수에 저장

```
var lat = '';
var lng = '';

// Drawing Manager에서 가져온 데이터 중
// 선의 꼭지점 정보를 kakao.maps.LatLng 객체로 생성하고 배열로 반환하는 함수입니다
function pointsToPath(points) {
    var len = points.length,
        path = [],
        i = 0;

    lat = '';
    lng = '';
    for (; i < len; i++) {
        var latlng = new kakao.maps.LatLng(points[i].y, points[i].x);
        path.push(latlng);
        lat += points[i].y.toFixed(5) + '/'
        lng += points[i].x.toFixed(5) + '/'
    }
}
```

코스 공유 상세페이지

반포GS-남산 청계천

수정 삭제

작성자: bb 날짜: 2023-05-23 17:18:41



▣ 자전거도로 정보 보기

출발 서울 서초구 반포동 115-5

내용

서울에서 가장 대표적인 암월 코스인 남산을 배경으로 남산 통과 후 비교적으로 금강과 지형을 통과하지 않고 안전하게 내려오고 싶다면 추천할 수 있는 코스입니다.

별점 ★★★★☆

☰ 목록

CourseController

DB에 저장된 값을 기준으로 분리시켜 jsp로 보냄

```
@RequestMapping("course/one")
public void one(int course_no, Model model) {
    CourseVO vo = dao.one(course_no);

    String[] lat = vo.getLine_lat().split("/");
    String[] lng = vo.getLine_lng().split("/");

    model.addAttribute("vo", vo);
    model.addAttribute("lat", lat);
    model.addAttribute("lng", lng);
    model.addAttribute("length", lng.length);
}
```

CourseDAO

```
public CourseVO one(int course_no) {
    CourseVO vo = my.selectOne("course.one", course_no);
    return vo;
}
```

courseMapper

```
<select id="one" parameterType="int" resultType="courseVO">
    select * from course c, member m
    where c.writer = m.Sign_id
    and course_no = #{course_no}
</select>
```

코스 공유 상세페이지

반포GS-남산 청계천

수정 삭제

작성자: bb 날짜: 2023-05-23 17:18:41



자전거도로 정보 보기

출발 서울 서초구 반포동 115-5

내용

서울에서 가장 대표적인 암월 코스인 남산을 배경으로 남산 통과 후 비교적으로 금강과 지형을 통과하지 않고 안전하게 내려오고 싶다면 추천할 수 있는 코스입니다.

별점 ★★★★☆

목록

course/one.jsp

forEach 이용하여 위도, 경도 값 가져오기

```
<c:forEach var="lat" items="${lat}" varStatus="status">
    <input value="${lat}" class="lat${status.index}" type="hidden">
</c:forEach>
<c:forEach var="lng" items="${lng}" varStatus="status">
    <input value="${lng}" class="lng${status.index}" type="hidden">
</c:forEach>
```

카카오맵 API 이용하여 지도에 polyline 나타냄

```
var len = ${length},
path = [];

for (var i = 0; i < len; i++) {
    var lat = $('.lat' + i).val();
    var lng = $('.lng' + i).val();
    var latlng = new kakao.maps.LatLng(lat, lng);
    path.push(latlng);
}

// 지도에 표시할 선을 생성합니다
let polyline = new kakao.maps.Polyline({
    map: map, // 지도에 선을 표시합니다
    path: path, // 선을 구성하는 좌표 배열입니다
    strokeWeight: 4, // 선의 두께입니다
    strokeColor: 'red', // 선의 색깔입니다
    strokeOpacity: 1, // 선의 불투명도입니다 1에서 0 사이의 값이며 0에 가까울수록 투명합니다
    strokeStyle: 'solid' // 선의 스타일입니다
});
```

자전거정보

Bike Friends 카뮤니티 - 자전거정보 이벤트 이용권

apple - 로그아웃

```
//검색조건으로 검색
@RequestMapping("bike/bikelistsearch")
public void listsearch(Model model, String searchtype, String keyword, SearchVO vo) {
    vo.setStartEnd(vo.getPage());
    vo.setSearchtype(searchtype);
    vo.setKeyword(keyword);

    List<BikeVO> list = null;
    int count = 0;
    //카테고리와 검색종류가 다르게 판별하기
    if(searchtype.equals("bike_brand")){
        list = dao.searchbrand(vo);
        count = dao.countbrand(keyword);
    }else if(searchtype.equals("bike_category")){
        list = dao.searchcategory(vo);
        count = dao.countcategory(keyword);
    }
    int pages = count / 6 + 1; // 전체 페이지 개수
    model.addAttribute("list", list);
    model.addAttribute("count", count);
    model.addAttribute("pages", pages);
    model.addAttribute("keyword", keyword);
    model.addAttribute("searchtype", searchtype);
}
```

제품종류와 브랜드 2가지로 검색가능



클릭시 자전
거 상세페이지로 이동

```
//자전거 전체 정보 불러오기
@RequestMapping("bike/bikelist")
public void list(PageVO vo, Model model) {
    vo.setStartEnd(vo.getPage());
    List<BikeVO> list = dao.all(vo);
    int count = dao.count();
    int pages = count / 6 + 1; // 전체 페이지 개수
    model.addAttribute("list", list);
    model.addAttribute("count", count);
    model.addAttribute("pages", pages);
}
```

페이지 처리 6개씩 불러옴

자전거상세정보

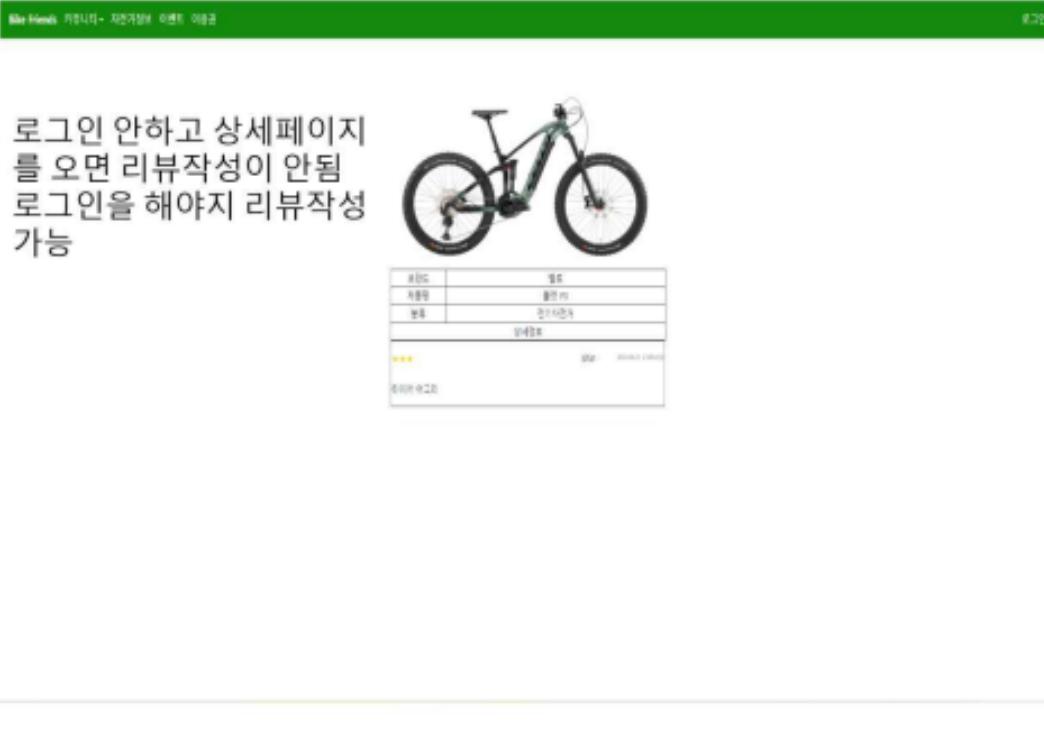
```
① 로그인한 사용자일 경우 →  
if(session.getAttribute("id") != null){ }  


별점 ★★★★★★★★★★★★★★★

작성


```

로그인 안하고 상세페이지를 보면 리뷰작성이 안됨
로그인을 해야지 리뷰작성 가능



The screenshot shows a bicycle detail page with a review section. The review section is grayed out, indicating it is disabled. The page includes a title, price information, a summary, and a detailed description.

제품	SSD
제조사	Western Digital
본점	한국 디＃
판매점	

상세설명보기	

상세설명보기

자전거정보 리뷰등록

```
$function() {
    $('#bikereplybtn').click(function() {
        bikereply_writer = "<%-session.getAttribute("id")%>"  

        bikereply_writername = "<%-session.getAttribute("name")%>"  

        bikereply_content = $('#content').val()  

        bikereply_rating = $('input[name=rating]:checked').val();  

        if(bikereply_content == ''){  

            alert("리뷰를 작성해주세요")  

        }else{
            $.ajax({
                url : "bikereplyinsert",
                data : {
                    bike_no : ${bag.bike_no},
                    bikereply_writer : bikereply_writer,
                    bikereply_content : bikereply_content,
                    bikereply_rating : bikereply_rating,
                    bikereply_writername : bikereply_writername
                },
                success : function(x) {
                    $('#insertreply').html(x)
                    location.reload()
                }
            })
        }
    })
}
```

localhost:8887 내용:

리뷰를 작성해주세요

확인



브랜드	셀트
제품명	풀셋 FX
분류	전기자전거
상세정보	

★ ★ ★ ← 주천점수

1000자 이내로 작성해주세요

작성

★★★ shin 2023-06-01 21:08:45.0

타이어 애그리

리뷰 작성시
내용이 없으면
등록 불가능

마우스를
위에 올리
면 노란색
활성화

자전거정보 리뷰삭제

Bike Friends 커뮤니티 - 자전거정보 이벤트 이용권

shin • 로그아웃

리뷰 작성할시 등록한 ID와 로그인
한 ID가 동일할시 삭제나 수정 가능

```
<!--로그인 ID와 등록한 ID가 동일할시-->
<c:choose>
<c:when test="${id == vo.bikereply_writer}">
<tr>
    <td colspan="3">
        <button id="replyup" style="float: right;" data-no="${vo.bikereply_no}" class="btn btn-secondary">수정</button>
        <button id="replydel" style="float: right; margin-right: 5px;" data-no="${vo.bikereply_no}" class="btn btn-secondary">삭제</button>
    </td>
</tr>
</c:when>
</c:choose>
```



브랜드	질료
제품명	풀셋 FX
분류	한기자전거
상세정보	

★ 주제: 자전거 리뷰

1000자 이내로 작성해주세요

작성 shin 2024-06-01 23:08:45

★★★ 판이며 어그리

삭제 수정

바로 삭제
가 아니라
삭제할것인
지 한번더
물어봄

```
$ (document).on('click', '#replydel', function() {
    if (confirm('리뷰 삭제하시겠습니까?')) {
        $.ajax({
            url: "bikereplydelete",
            data: {
                bikereply_no: $(this).data('no')
            },
            success: function() {
                alert('리뷰 삭제 완료')
                location.reload()
            }
        })
    }
})
```

자전거정보 리뷰수정

```
$ document ).ready( function() {
    var ratingcheck = ${vo.bikereply_rating};
    $('input:radio[name=rating][value=' + ratingcheck + ']').prop('checked', true);

    $(function() {
        $("#b1").click(function() {
            bikereply_content = $('#content').val();
            bikereply_rating = $('input[name=rating]:checked').val();
            $.ajax({
                url : 'bikereplyupdate2',
                data : {
                    bikereply_no : ${vo.bikereply_no},
                    bikereply_content : bikereply_content,
                    bikereply_rating : bikereply_rating
                },
                success : function(result){
                    opener.location.href = "bikonestbike_no=" + ${vo.bike_no}
                    window.close();
                }//success
            });//ajax
        })
    })
})
```

The screenshot shows a browser window with a green header bar containing the text 'Bike Friends 커뮤니티 - 자전거정보 이벤트'. Below the header, there is a message in Korean: '적었던 리뷰와 별점을 가지고 올' (Bring the review and rating you marked). The main content area contains the following JavaScript code:

```
// 예상본
$(document).on("click", "#b1", function(){
    e.preventDefault();
    bikereply_no = ${vo.bikereply_no}
    let popOpt = "bikereplyupdate?bikereply_no=" + bikereply_no;
    let popOption = "width=600px, height=400px, top=300px, left=300px, scrollbars=1";
    window.open(popOpt, "55_44", popOption);
})
```

Below the code, there is a diagram illustrating the interaction. A large arrow points from the '수정버튼 클릭시 팝업 화면 띄움' (Pop-up window appears when the update button is clicked) text to the 'window.open' line in the code. Another arrow points from the '수정화면' (Update window) part of the diagram to the modal dialog shown in the screenshot.

The modal dialog has a title '수정화면' and a URL 'localhost:8888/mad/bike/bikereplyupdate?bikereply_no=1'. It contains a rating section with three yellow stars and a '별점수정' (Rating adjustment) button. The main form fields include 'bikereply_content' (content), 'bikereply_rating' (rating), and 'bikereply_no' (review ID). At the bottom, there are '삭제' (Delete) and '수정' (Update) buttons.

The bottom part of the screenshot shows the updated review list. One review by 'shin' is visible, showing a rating of 3 stars, the content '타이어 애그리', and the date '2021-08-01 22:00:00'. There are '삭제' (Delete) and '수정' (Update) buttons next to the review.

이벤트

<따릉이 정보>

```
var ajax1=$.ajax({
  url : "http://openapi.seoul.go.kr:8888/49747167436369633836706f6a5367/json/bikeList/1/1000",
  success : function(response) {
    for (let i = 0; i < 1000; i++) {
      var jsonObject = {
        "data" : response['rentBikeStatus'][['row']][i].parkingBikeTotCnt,
        "lat":response['rentBikeStatus'][['row']][i].stationLatitude,
        "lon": response['rentBikeStatus'][['row']][i].stationLongitude
      }
      jsonArray1[i] = jsonObject;
    }
  }
});
```

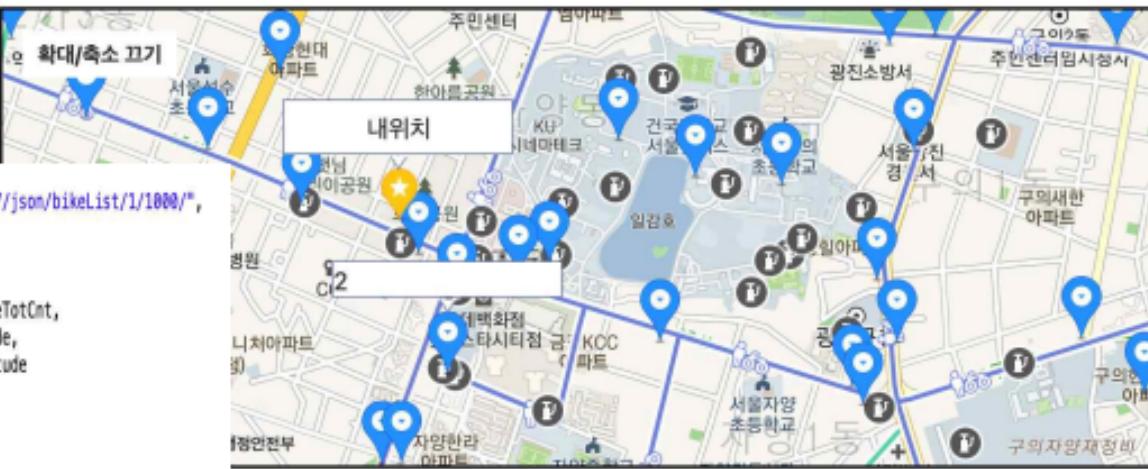
따릉이 API에서 실시간 정보를
jsonArray에 저장

```
// 마커를 표시합니다
var marker = new kakao.maps.Marker({
  map : map, // 마커를 표시할 지도
  position : new kakao.maps.LatLng(response1[i][0],response1[i][1]) // 마커의 위치
});

// 마커에 표시할 인포윈도우를 생성합니다
var infowindow = new kakao.maps.InfoWindow({
  content :jsonArray1[i].data
});

// 인포윈도우에 표시할 내용
});
```

따릉이 정보 내위치로 이동



내 위치 주변 따릉이 정보 실시간
으로 확인
따릉이 정보 확인 후 이용권
페이지로 넘어가 결제과정을
거침
jsonArray의 data를 kakao.maps
로 전달

이벤트

<즉석모임 개설>

총 20.66명 20.66% / 18.69%
같이 달려요
인증샷챌린지

번개 모임					
	제목	글쓴이	장소	모임 날짜	참여 인원
2	bikell	banana	서울시 송파구 석촌호수로 262	2023-06-04	3
3	race	aaa	서울 강남구 도산대로 57길 24	2023-06-04	10
11	gogogo	bbb	서울 강남구 암구정로 39길 20	2023-06-07	20
12	no.1	ccc	서울 성동구 성수동1가 685-715	2023-06-04	14
14	번개 모여요	apple	서울시 광진구 자양동 427-6	2023-05-24	6
			서울 성동구 성수동1가 685-715	2023-06-04	6

```

<c:forEach var="eventVO" items="${list}">
    <tr align="center">
        <td class="down">${eventVO.no}</td>
        <td class="down"><a href="#one?event_id=${eventVO.event_id}">${eventVO.title}</a></td>
        <td class="down">${eventVO.event_id}</td>
        <td class="down" style="text-align: center;">${eventVO.event_location}</td>
        <td class="down" style="text-align: center;">${eventVO.event_date}</td>
        <td class="down" style="text-align: center;">${eventVO.event_size}</td>
    </tr>
</c:forEach>

```

번개 모여요



모임장소: 서울시 강남구 저동 427-6
모임날짜: 2023-06-04

글쓴이: apple
같이모이는 첫 사람 구현~!!~



상세페이지로 이동

eventcontroller로 저장된
eventVO로 상세페이지 구성

이벤트

<인증샷 챌린지>

유저들이 자전거 활동 셀피를 올려
자유롭게 공유하는 페이지



```
<c:foreach var="vo" items="${list}">
<td class="down" style="text-align:center"><a href="../eventchallenge/imgone?img=${vo.img}">
</a></td>
<br>
if(k++%3==0){
<br><tr></tr>
}
</c:foreach>
```



작성자:aaa

Hi

작성날짜:2023-04-06 08:03:19.0

상세페이지로 이동

```
<div style="text-align : center;"></div>
<div id="main">
작성자:${vo.writer}<br>
</div>
<div id="main">
${vo.content}<br>
</div>
<div id="main">
작성날짜:${vo.date}<br>
</div>
```

eventchallengeVO의 img로
갤러리 형태의 list 구성

이용권

Bike Friends 커뮤니티 * 자전거정보 * 이벤트 * 이용권

조장현 * 로그아웃

파울이 철로



지역별 행사판

회원별 인기행사판

회원별 인기행사판	최신행사판
강남구 자전거	05-09
강남구 화료일 자전거 대여	05-10
강남구 5.9 자전거 보험 일정	05-09
강남구 자전거	05-11
강남구 새길 자전거 모임 일정	05-18
강남구 화료일 자전거 대여	05-18
용산구 내일 자전거 사진 공모	05-18
용산구 자전거	05-15
용산구 자전거 사진 공모	05-15
마포구 자전거 사진 공모	05-15

메인 메뉴의 이용권 버튼을 누르면
이용권 구매화면으로 넘어감

이용권 선택 화면



```
$(function() {
    $('#b2').click(function() {
        $.ajax({
            url: "json1000",
            dataType : "json",
            success: function(json) {
                money = json.money
                money_string = '결제금액 ' + money
                $('#result_1').html(money)
            } //success
        }) //ajax
    }) //b3
})
```

```
@RequestMapping("payment/json1000")
@ResponseBody //프정을 받으면 return되는 데이터를 json으로 만들어서 요청하는 곳으로 다시 보냄.
public PayVO json1000() {
    PayVO bag = new PayVO();
    bag.setMoney("1000");
    return bag; //{"id" : "root", "pw" : "1234", ;
}
```

일일권, 정기권 선택이 가능하며
AJAX를 활용해 시간권 선택이 가능
함

이용권 선택 코드 및 설명

```
@RequestMapping("payment/json1000")
@ResponseBody //요청을 받으면 return되는 곳으로 다시 보냄.
public PayVO json1000() {
```

```
    PayVO bag = new PayVO();
    bag.setMoney("1000");
    return bag; //{"id": "root", "m": "1000", ...}
}
```

```
@RequestMapping("payment/json2000")
@ResponseBody //요청을 받으면 return되는 곳으로 다시 보냄.
public PayVO json2000() {
```

```
    PayVO bag = new PayVO();
    bag.setMoney("2000");
    return bag; //{"id": "root", "m": "2000", ...}
}
```

```
@RequestMapping("payment/json5000")
@ResponseBody //요청을 받으면 return되는 곳으로 다시 보냄.
public PayVO json5000() {
```

```
    PayVO bag = new PayVO();
    bag.setMoney("5000");
    return bag; //{"id": "root", "m": "5000", ...}
}
```

```
var money = 0;
$(function() {
    $('#b2').click(function() {
        $.ajax({
            url: "json1000",
            dataType: "json",
            success: function(json) {
                money = json.money;
                money_string = '결제금액 ' + money;
                $('#result_1').html(money)
            } //success
        }) //ajax
    })//b3

    $('#b5').click(function() {
        $.ajax({
            url: "json2000",
            dataType: "json",
            success: function(json) {
                money = json.money;
                money_string = '결제금액 ' + money;
                $('#result_1').html(money)
            } //success
        }) //ajax
    })//b3

    $(".button5").click(function() {
        //value = $(".button4").val();
        alert(money)
    })
})
```

```
<header id="header">
    <div>이용권 선택</div>
    <div style="padding: 1px 2px; text-align: right; line-height: 1;">
        <a href="paidlist?page=1">결제내역</a>
    </div>
</header>

<main id="contents">
    <a href="dailyticket.jsp">
        <button class="button buttons">일일별</button>
    </a>
    <a href="regularticket.jsp">
        <button class="button buttons">정기별</button>
    </a>
    <br color="gray"><br><br>
    <div class="c">가장 먼저 시간</div>
    <br> <br> <br>
    <br> <br>
    <button id="b2" class="button button4" value="1시간">1시간<br>1000</button>
    <br> <br> <br>
    <button id="b3" class="button button4" value="2시간">2시간<br>2000</button>
    <br> <br>
    <br color="red">
    <br>
</main>

<footer id="footer">
    <br>
    <span>결제금액</span>
    <span id="result_1" style="font-size:large"></span>
    <br>
    <div id="result" style="padding: 1px 2px; text-align: right; line-height: 1;">
        <a href="#" class="button buttons">결제</button>
    </div>
</footer>
```

Ajax를 이용해 각 버튼을 클릭하면 해당 금액을 받아옴

Controller에서 요청을 받으면 return되는 데이터를 json으로 만들어 요청하는 곳으로 보냄

구매하기 화면으로 넘어갈 때 결제금액의 데이터를 함께 보냄

이용권 구매 페이지

구매하기

결제수단

pay

구매자 정보

아이디: jocharlie
이메일: jocharlie@naver.com
구매자이름: 조찰현
구매자 전화번호: 010-5555-5555

아이디: \${id}

이메일: \${email}

구매자이름: \${name}

구매자 전화번호: \${phone}

이용약관

제 1 장 총칙 제1조(목적) 이 약관은 회원의 권리보호 및 거래관계의 명확화를 위하여 신용카드사(이하 '카드사'라 함)와 신용카드(이하 '카드'라 함)를 이용하고자 하는 회원 간의 필요에 상호 협약을 설정으로 합니다 제2조(회원)

결제수단

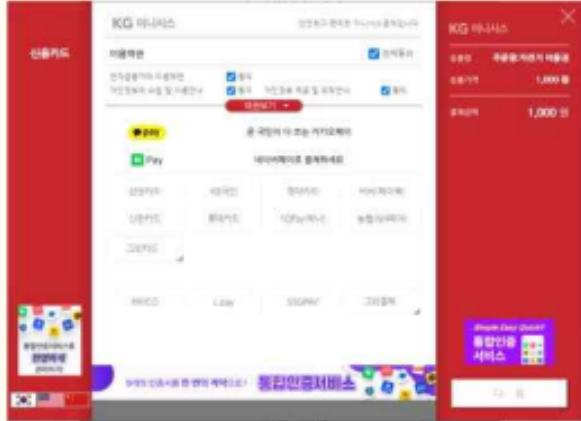
구매금액

결제하기

```
$(function() {  
    $("#order2_chk").click(function() {  
        var order2Chk = $("#order2_chk").prop("checked");  
        // prop 를 사용하여 checkbox의 속성을 접근  
  
        if (order2Chk) {  
            $(".order2_chks").prop("checked", true);  
        } else {  
            $(".order2_chks").prop("checked", false);  
        }  
  
    });  
  
    // 체크박스가 체크되었을 때는 활성화 되게 만들기  
  
    $('.checkT input[type="checkbox"]').click(function() {  
        var tmp = $(this).prop('checked');  
        // this는 사용하여 클릭한 checkbox 가 체크되었을 경우  
        if (tmp) {  
            $(".btnArea button").css({  
                "backgroundColor" : "#d2201d",  
                "cursor" : "pointer",  
                "color" : "#fff"  
            }).prop("disabled", false);  
        } else {  
            $(".btnArea button").css({  
                "backgroundColor" : "#cbcbcb",  
                "cursor" : "auto",  
                "color" : "#303033"  
            }).prop("disabled", true);  
        }  
    });  
});
```

- 구매자 본인의 개인정보 확인
- 이용약관을 동의하면 결제하기 버튼이 활성화 되고 결제하면 구현

이용권 구매 코드 및 설명



- 구매자의 id와 email 등 개인정보를 불러 올 수 있게 표시
- 구매자의 개인정보를 입력받고 iampay 결제 api를 활용해 결제 시스템을 진행

```
const name = $('#name').text(); //value값을 의미한다...
const email = $('#email').text();
const tel = $('#tel').text();
const id = $('#id').text();

const money = ${param.money};

//alert(email);
DP.init('iampay'); // 'iampay' 대신 결제방법은 "가상계좌_직불코드"를 사용
DP.request_pay({
    pg : 'inicis', // version 1.1.0부터 지원
    pay_method : 'card',
    merchant_uid : 'merchant_' + new Date().getTime(),
    name : '주유령_차전거 이용권',
    amount : money,
    // amount : ${money}.val(),
    buyer_email : email,
    buyer_name : name,
    buyer_tel : tel,
    buyer_addr : '서울특별시 강남구 삼성동',
    buyer_postcode : '123-456',
    _redirect_url : 'http://www.yourdomain.com/payments/complete'
}), function(rsp) {
    if (rsp.success) {
        //여기서부턴 들어주는 정보
        var msg = '결제가 완료되었습니다.';
        console.log('rsp : ',rsp);
        msg += '고유ID : ' + rsp.imp_uid;
        msg += '상점 거래ID : ' + rsp.merchant_uid;
        msg += '결제 금액 : ' + rsp.paid_amount;
        msg += '카드 유인번호 : ' + rsp.apply_num;

        console.log(msg);
        location.href="insertpay?email=" + email + "&id=" + id + "&tel=" + tel + "&money=" + money + "&name=" + name
    }
}
```

구매내역 페이지

구매내역					
아이디	구매금액	구매자	전화번호	email	
jocharlie	1000	조창현	010-5555-5555	jocharlie@maven.com	<input type="button" value="상세"/> <input type="button" value="삭제"/>
jocharlie	1000	조창현	010-5555-5555	jocharlie@maven.com	<input type="button" value="상세"/> <input type="button" value="삭제"/>

삭제

```
$function() {
  $(".delete").click(function() {
    //alert($(this).attr("value"))
    $.ajax ({
      url : "delete",
      data : {
        id : $(this).attr("value")
      },
      success : function(x) {
        location.href="list_result.jsp"
        $('#result').append(x+"  
")
      }
    })
  })
}
```

```
<cc:forEach items="${list}" var="bag">
  <tr>
    <td class="down">${bag.id}</td>
    <td class="down">${bag.money}</td>
    <td class="down">${bag.name}</td>
    <td class="down">${bag.tel}</td>
    <td class="down">${bag.email}</td>
    <td><button class="delete" value="${bag.pay_no}">삭제</button></td>
  </tr>
```

결제가 끝나면 구매내역 화면으로 넘어감

페이징 기법과 연동해 각 구매내역이 10개 이상이 되면 추가내용은 2페이지에 만들어짐

초반 이용권 구매 메뉴에서도 본인의 구매 내역 확인 가능

구매내역 코드 및 설명

```
<cc:forEach items="${list}" var="bag">
  <tr>
    <td class="down">${bag.id}</td>
    <td class="down">${bag.money}</td>
    <td class="down">${bag.name}</td>
    <td class="down">${bag.tel}</td>
    <td class="down">${bag.email}</td>
  </tr>
  <td><button class="delete" value="${bag.pay_no}">삭제</button></td>
</cc:forEach>
```

```
public classRestController {
  @Autowired
  PayDAO dao;
  // @PostMapping("/insertpay")
  @RequestMapping("payment/insertpay")
  public String insert(PayVO bag, Model model) {
    System.out.println("insertpay : "+bag);
    PageIVO vo = new PageIVO();
    vo.setPage(1);
    vo.setStartInd(1);
    dao.insert(bag);
    List list = dao.all(vo); // //리스트를 받고자 한다면
    int count = dao.count();
    int pages = count / 10 + 1;
    model.addAttribute("list", list);
    model.addAttribute("count", count);
    model.addAttribute("pages", pages);
    System.out.println("list : "+list);
    return "payment/list_result";
  }
}
```

```
10<div style="display: flex; justify-content: center; align-items: center;">
11  <div style="margin: 0 auto; width: fit-content; text-align: center;">
12    <div style="border: 1px solid gray; padding: 2px; margin-right: 10px; border-radius: 5px; background-color: #e0e0e0;">
13      <span style="font-size: 1em;">${list}
14    </div>
15    <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 10px;">
16      <div style="text-align: center;">
17        <span style="background-color: #ccc; border: 1px solid #ccc; padding: 2px 10px; border-radius: 5px; color: #333; font-weight: bold; font-size: 1em; margin-bottom: 5px;">${list}
18        <span style="color: #333; font-size: 0.8em;">${list}개입니다.
19      </div>
20      <div style="text-align: center;">
21        <span style="background-color: #ccc; border: 1px solid #ccc; padding: 2px 10px; border-radius: 5px; color: #333; font-weight: bold; font-size: 1em; margin-bottom: 5px;">${list}
22        <span style="color: #333; font-size: 0.8em;">${list}개입니다.
23      </div>
24    </div>
25  </div>
26</div>
```

- 이용자의 구매내역을 가져와 나타냄
- 페이징된 내용을 Controller에서 리스트를 받아 보내줌
- 이용자의 구매내역 리스트를 페이징을 이용해 나타냄

쪽지

The screenshot shows a messaging application interface. At the top, there's a green header bar with the word "쪽지" (Message) in white. Below the header is a map of Seoul with several blue location markers. To the right of the map is a sidebar with the text "마이페이지 쪽지". The main content area contains a table titled "쪽지 목록" (Message List) with the following data:

보낸 사람	제목	날짜
abcd	aaaaa	2023-05-27 13:28:00
abcd	안녕하세요	2023-05-27 13:34:00
abcd	□ □ □	
abcd	□ □ □ □	
abcd	ffff	
abcd	aaaa	
abcd	bbb	2023-05-30 14:26:43

Below the message list, there are two buttons: "쪽지 정보" and "쪽지 목록". At the bottom left, there's a small copyright notice: "© Bluefriends All rights reserved."

받은 쪽지함

보낸 사람	제목	날짜
abcd	aaaaa	2023-05-27 13:28:00
abcd	안녕하세요	2023-05-27 13:34:00
abcd	□ □ □	
abcd	□ □ □ □	
abcd	ffff	
abcd	aaaa	
abcd	bbb	2023-05-30 14:26:43

로그인하면 쪽지 페이지에 접속 가능
쪽지 페이지 접속 시 받은 쪽지 함으로 이동

쪽지 조회 및 삭제

받은 쪽지함

보낸 사람	제목	날짜
<input checked="" type="checkbox"/> banana	9	2023-05-30 14:37:00
<input checked="" type="checkbox"/> banana	10	2023-05-30 14:37:08
<input type="checkbox"/> banana	11	2023-05-30 14:37:12
<input type="checkbox"/> banana	12	2023-05-30 14:37:16



보낸 사람:

banana

날짜:

2023-05-30 14:37:05

제목:

9

내용:

9

[답장 보내기](#)

```
$(function() {
    $('#list').on('click', function() {
        var checkboxes = document.querySelectorAll('input[name="moveCheckboxes"] :checked');
        if (checkboxes.length == 1) {
            alert("선택한 내용을 보내거나 편집합니다.");
        } else {
            alert("선택한 쪽지를 선택하여 편집 혹은 보낼 때");
        }
        checkboxes = $(this).find('input');
        var index = 1;
        checkboxes.each(function(index, checkbox) {
            index++;
            checkbox.push(checkboxes[index].checked);
        });
    });

    $('#list').on('click', '#del', function() {
        var index = 1;
        checkboxes = $(this).find('input');
        checkboxes.each(function(index, checkbox) {
            index++;
            if (checkbox.checked) {
                $.ajax({
                    url: "delete",
                    type: "post",
                    data: {
                        index: index
                    },
                    success: function(response) {
                        location.href = response.location;
                        var session = getCookie("id");
                        console.log("로그인한 사용자:" + session);
                    },
                    error: function(error) {
                        console.log("로그인한 사용자:" + error);
                    }
                });
            }
        });
    });
});
```

쪽지를 선택하여, 선택한 쪽지만큼 삭제

삭제 기능을 ajax로 구현

쪽지 제목 부분을 누르면 쪽지 상세 페이지로 이동

DB에서 로그인한 이용자와 관련된 쪽지만 조회

쪽지 보내기

```
@RequestMapping("note	insert")
public String insert(NoteVO bag, HttpSession session) {
    System.out.println(bag);
    dao.insert(bag);

    String receiver = (String) session.getAttribute("id");
    return "redirect:receive_notelist?receiver=" + receiver;
}
```

받는 사람:

보내는 사람:

제목:

내용:

받은 쪽지함

보낸 사람	제목	날짜
<input checked="" type="checkbox"/> banana	9	2023-05-30 14:37:05
<input checked="" type="checkbox"/> banana	10	2023-05-30 14:37:08
<input type="checkbox"/> banana	11	2023-05-30 14:37:12
<input type="checkbox"/> banana	12	2023-05-30 14:37:16

로그인 된 아이디가 자동으로 보내는 사람으로
출력
쪽지 발송시 서버 시간이 db에 자동으로 입력
쪽지 발송 후, 받은 쪽지함으로 이동

감사합니다