CodeTourist (이동영, 이정명)

Enjoy Trip

CodeTourist (이동영, 이정명)

목차

기획 배경 및 목표
추진 계획 (일정)
시장분석
개발 결과
기대효과
개발후기
참조

기획 배경 및 목표

기획 배경

이번 방학기간을 활용해서
여행을 하기 위해 계획을 짜기 위해
다양한 사이트들을 돌아다녔는데,
서로 조금씩 부족한 부분을 공유하고 있다는
생각이 들어 이런 부분이
개선하고 싶어 개발하게 되었습니다.

목표

국내 여행 계획을 짜고
공유할 수 있는 웹 프로젝트 개발

다양한 정보를 우리 사이트를 중심으로
한 번에 확인할 수 있는 프로젝트 개발

기본기를 충실히 소화할 수 있고
데이터에 휘둘리지 않는 확장성 높은
국내 여행 중심의 웹 프로젝트

여행의 전 과정을
한번에 계획할 수 있는
기본기에 충실한 사이트

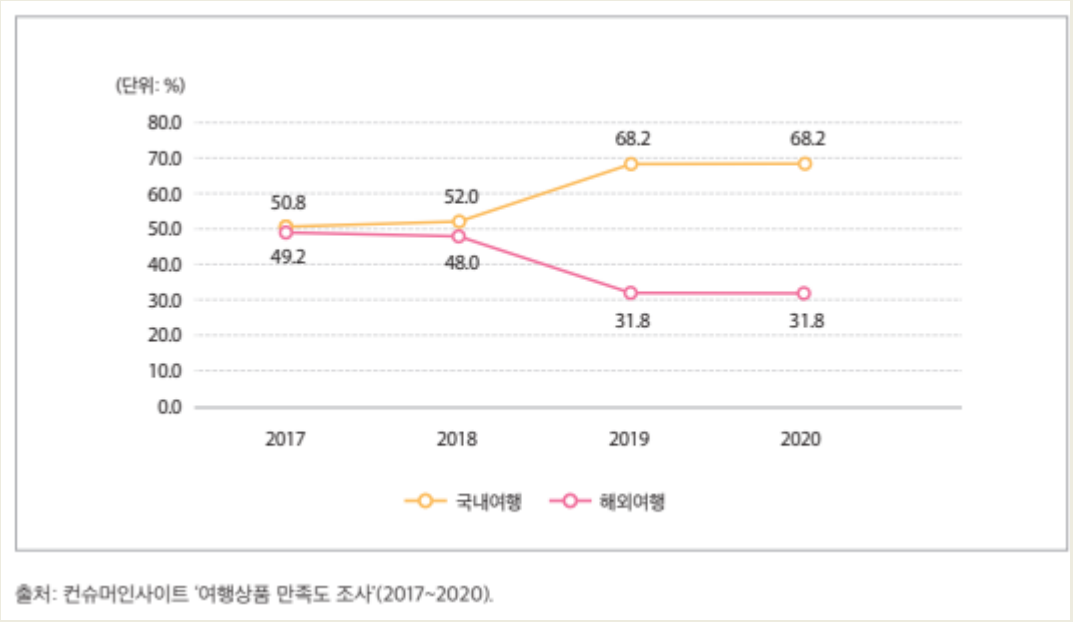
추진 계획 (일정)

추진일정 (5월)

프로젝트 기간 - 10일 (23.05.17 ~ 26)

시장분석

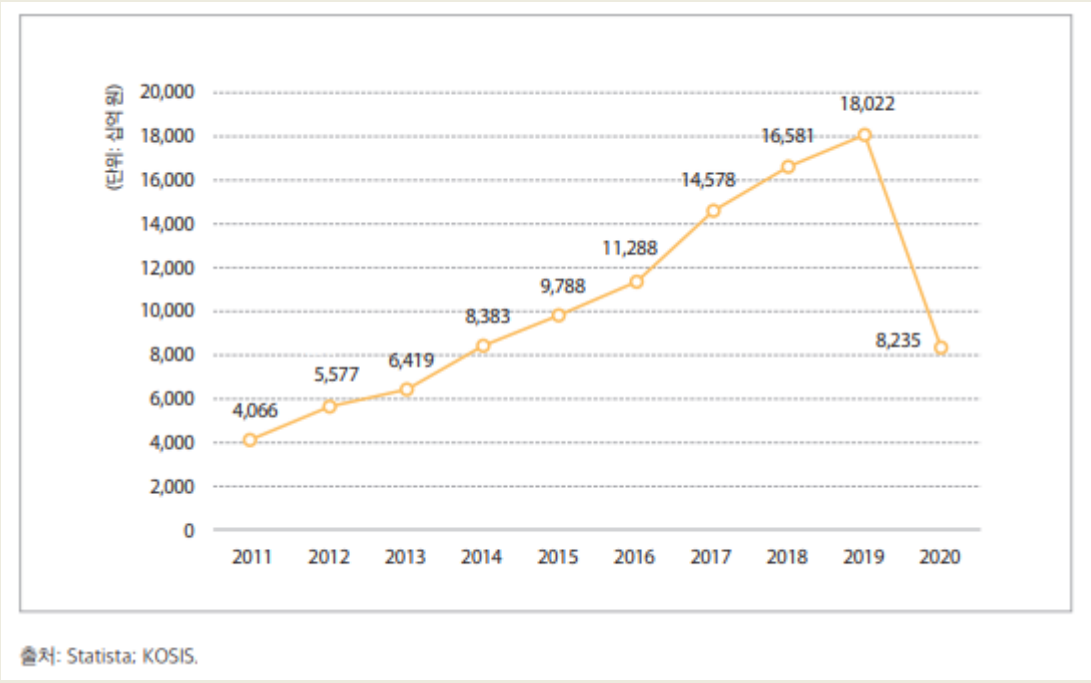
시장분석



출처: 컨슈머인사이트 '여행상품 만족도 조사'(2017~2020).

온라인 여행사를 통한 구입 상품의 여행지역

온라인 여행사이트 활성화로 인한 국내 여행 여행의 온라인 여행사를 통한 구매 비중 증가



출처: Statista; KOSIS.

국내 온라인 여행판매 금액(2011~2020)

코로나 19로 인한 매출액 감소

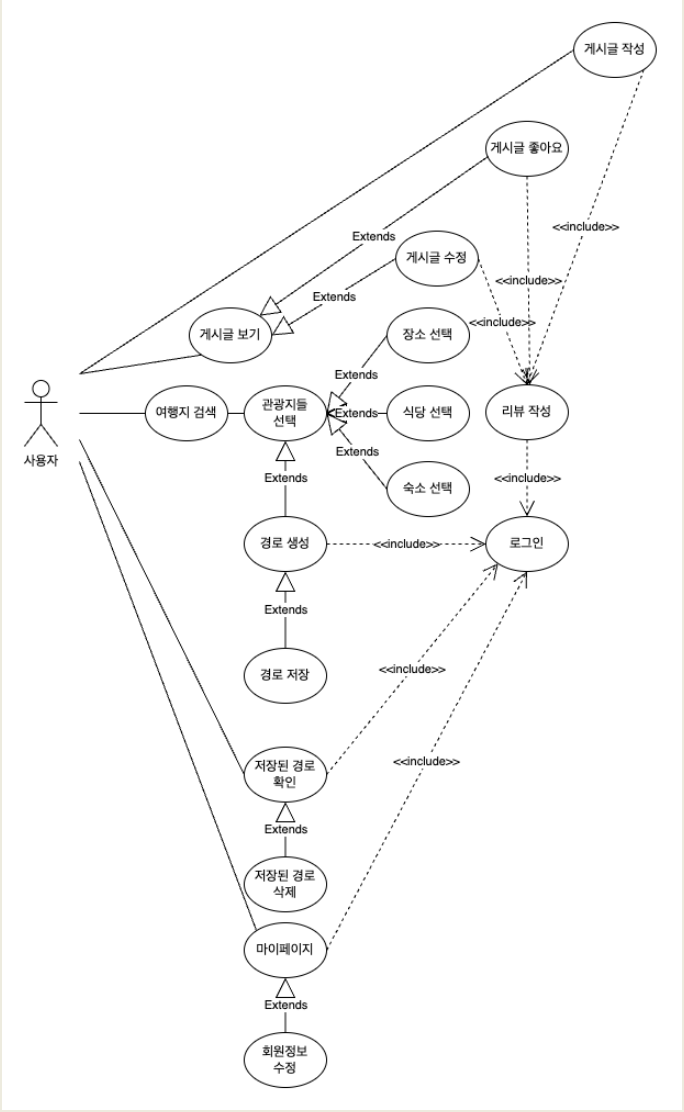
개발 결과

개발 환경 (프론트)

개발 환경 (백)

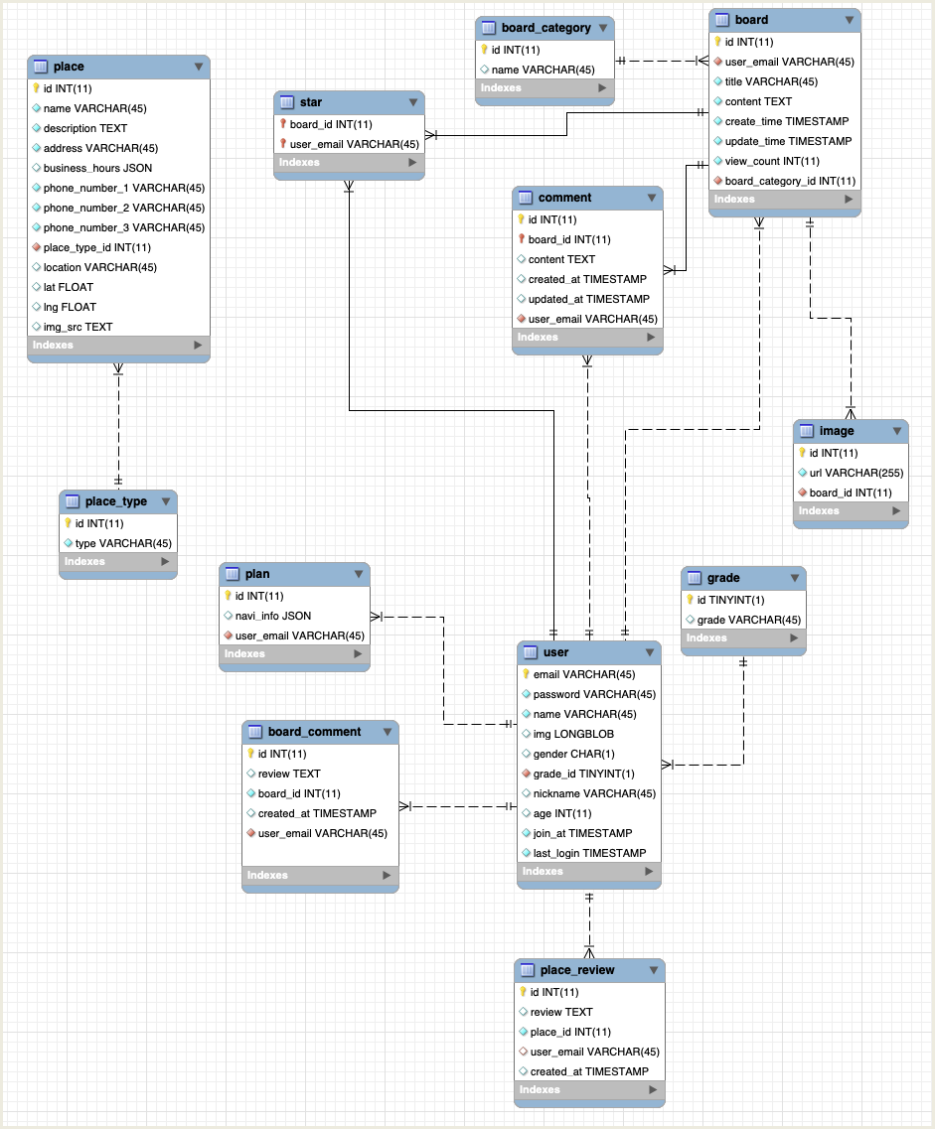
전체 시스템 구조도

UML
(Usecase Diagram)

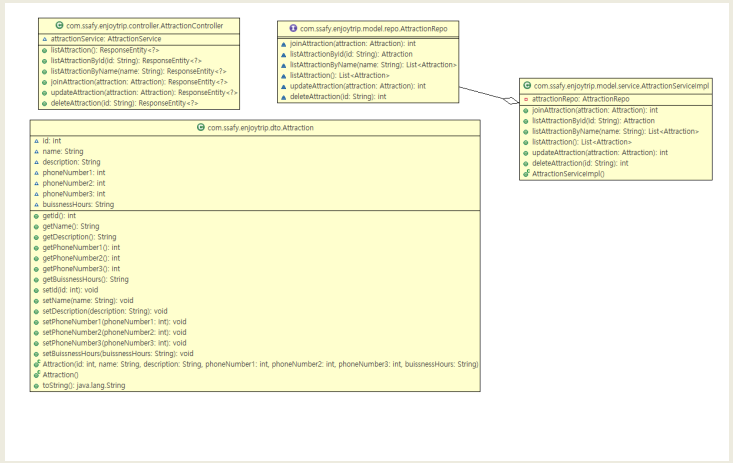


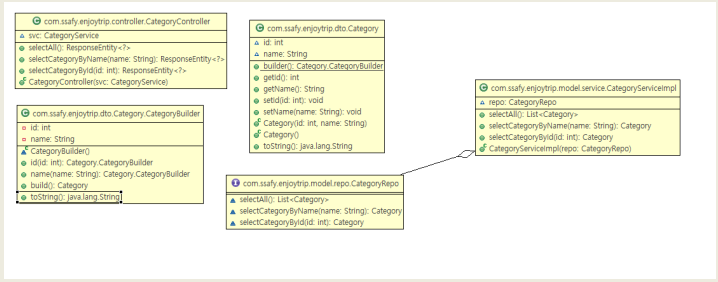
전체 시스템 구조도

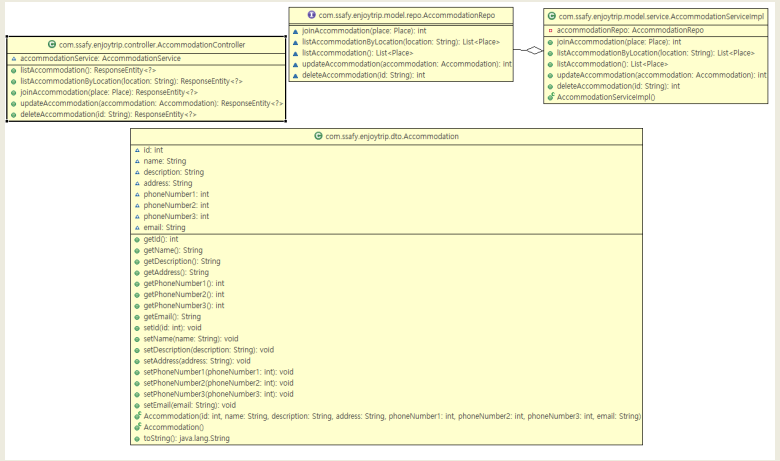
ERD
(Entity Relationship Diagram)

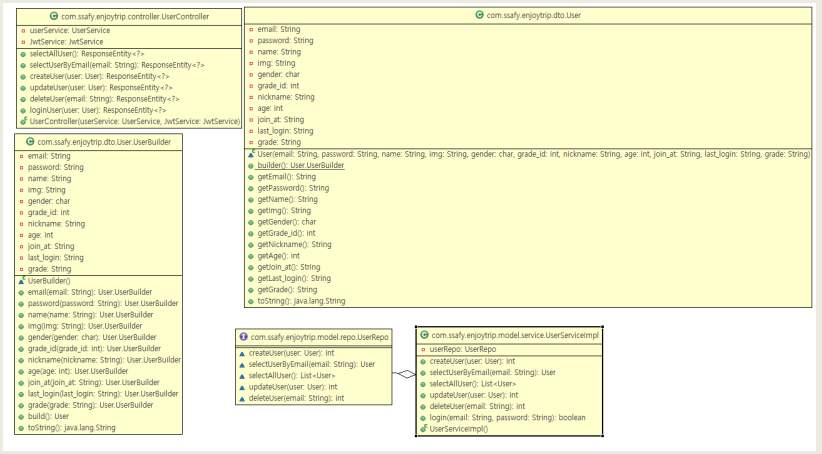


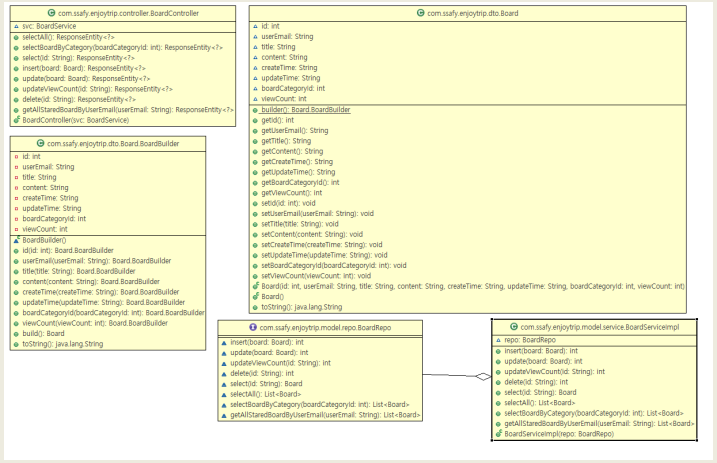
전체 시스템 구조도

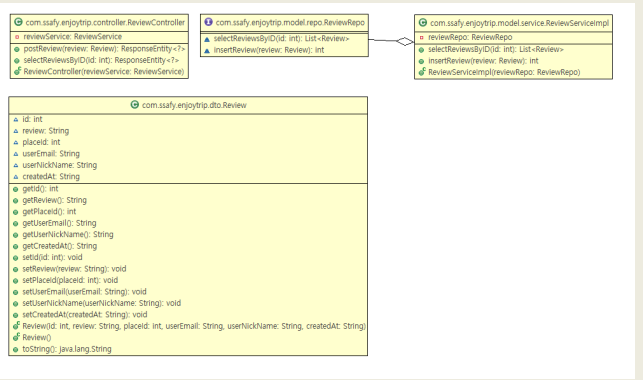

com.ssafy.enjoytrip.controller.AttractionController
attractionService: AttractionService
listAttraction(): ResponseEntity<?>
listAttractionById(id: String): ResponseEntity<?>
listAttractionByName(name: String): ResponseEntity<?>
joinAttraction(attraction: Attraction): ResponseEntity<?>
updateAttraction(attraction: Attraction): ResponseEntity<?>
deleteAttraction(id: String): ResponseEntity<?>
com.ssafy.enjoytrip.model.repo.AttractionRepo
joinAttraction(attraction: Attraction): int
listAttractionById(id: String): Attraction
listAttractionByName(name: String): List<Attraction>
listAttraction(): List<Attraction>
updateAttraction(attraction: Attraction): int
deleteAttraction(id: String): int
com.ssafy.enjoytrip.model.service.AttractionServiceImpl
attractionRepo: AttractionRepo
joinAttraction(attraction: Attraction): int
listAttractionById(id: String): Attraction
listAttractionByName(name: String): List<Attraction>
listAttraction(): List<Attraction>
updateAttraction(attraction: Attraction): int
deleteAttraction(id: String): int
AttractionServiceImpl()
com.ssafy.enjoytrip.dto.Attraction
id: int
name: String
description: String
phoneNumber1: int
phoneNumber2: int
phoneNumber3: int
buisnessHours: String
getId(): int
getName(): String
getDescription(): String
getPhoneNumber1(): int
getPhoneNumber2(): int
getPhoneNumber3(): int
getBuisnessHours(): String
setId(id: int): void
setName(name: String): void
setDescription(description: String): void
setPhoneNumber1(phoneNumber1: int): void
setPhoneNumber2(phoneNumber2: int): void
setPhoneNumber3(phoneNumber3: int): void
setBuisnessHours(buisnessHours: String): void
Attraction(id: int, name: String, description: String, phoneNumber1: int, phoneNumber2: int, phoneNumber3: int, buisnessHours: String)
Attraction()
toString(): java.lang.String

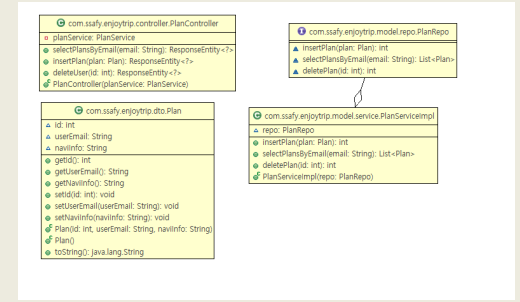

com.ssafy.enjoytrip.controller.CategoryController
svc: CategoryService
selectAll(): ResponseEntity<?>
selectCategoryByName(name: String): ResponseEntity<?>
selectCategoryById(id: int): ResponseEntity<?>
CategoryController(svc: CategoryService)
com.ssafy.enjoytrip.dto.Category.CategoryBuilder
id: int
name: String
CategoryBuilder()
id(id: int): Category.CategoryBuilder
name(name: String): Category.CategoryBuilder
build(): Category
toString(): java.lang.String
com.ssafy.enjoytrip.dto.Category
id: int
name: String
builder(): Category.CategoryBuilder
getId(): int
getName(): String
setId(id: int): void
setName(name: String): void
Category(id: int, name: String)
Category()
toString(): java.lang.String
com.ssafy.enjoytrip.model.repo.CategoryRepo
selectAll(): List<Category>
selectCategoryByName(name: String): Category
selectCategoryById(id: int): Category
com.ssafy.enjoytrip.model.service.CategoryServiceImpl
repo: CategoryRepo
selectAll(): List<Category>
selectCategoryByName(name: String): Category
selectCategoryById(id: int): Category
CategoryServiceImpl(repo: CategoryRepo)

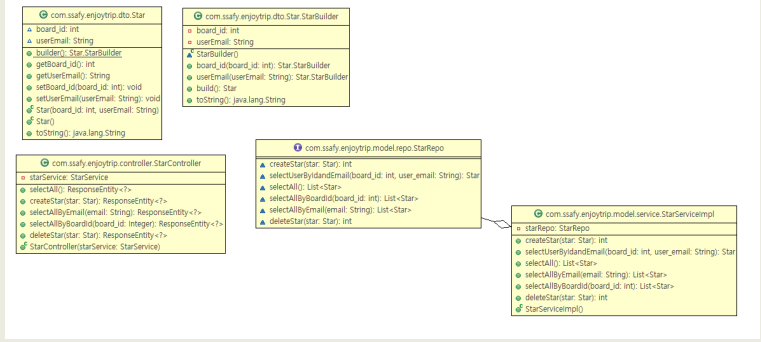

com.ssafy.enjoytrip.controller.AccommodationController
accommodationService: AccommodationService
listAccommodation(): ResponseEntity<?>
listAccommodationByLocation(location: String): ResponseEntity<?>
joinAccommodation(place: Place): ResponseEntity<?>
updateAccommodation(accommodation: Accommodation): ResponseEntity<?>
deleteAccommodation(id: String): ResponseEntity<?>
com.ssafy.enjoytrip.model.repo.AccommodationRepo
joinAccommodation(place: Place): int
listAccommodationByLocation(location: String): List<Place>
listAccommodation(): List<Place>
updateAccommodation(accommodation: Accommodation): int
deleteAccommodation(id: String): int
com.ssafy.enjoytrip.model.service.AccommodationServiceImpl
accommodationRepo: AccommodationRepo
joinAccommodation(place: Place): int
listAccommodationByLocation(location: String): List<Place>
listAccommodation(): List<Place>
updateAccommodation(accommodation: Accommodation): int
deleteAccommodation(id: String): int
AccommodationServiceImpl()
com.ssafy.enjoytrip.dto.Accommodation
id: int
name: String
description: String
address: String
phoneNumber1: int
phoneNumber2: int
phoneNumber3: int
email: String
getId(): int
getName(): String
getDescription(): String
getAddress(): String
getPhoneNumber1(): int
getPhoneNumber2(): int
getPhoneNumber3(): int
getEmail(): String
setId(id: int): void
setName(name: String): void
setDescription(description: String): void
setAddress(address: String): void
setPhoneNumber1(phoneNumber1: int): void
setPhoneNumber2(phoneNumber2: int): void
setPhoneNumber3(phoneNumber3: int): void
setEmail(email: String): void
Accommodation(id: int, name: String, description: String, address: String, phoneNumber1: int, phoneNumber2: int, phoneNumber3: int, email: String)
Accommodation()
toString(): java.lang.String


com.ssafy.enjoytrip.controller.UserController
userService: UserService
JwtService: JwtService
selectAllUser(): ResponseEntity<?>
selectUserByEmail(email: String): ResponseEntity<?>
createUser(user: User): ResponseEntity<?>
updateUser(user: User): ResponseEntity<?>
deleteUser(email: String): ResponseEntity<?>
loginUser(user: User): ResponseEntity<?>
UserController(userService: UserService, JwtService: JwtService)
com.ssafy.enjoytrip.dto.User.UserBuilder
email: String
password: String
name: String
img: String
gender: char
grade_id: int
nickname: String
age: int
join_at: String
last_login: String
grade: String
UserBuilder()
email(email: String): User.UserBuilder
password(password: String): User.UserBuilder
name(name: String): User.UserBuilder
img(img: String): User.UserBuilder
gender(gender: char): User.UserBuilder
grade_id(grade_id: int): User.UserBuilder
nickname(nickname: String): User.UserBuilder
age(age: int): User.UserBuilder
join_at(join_at: String): User.UserBuilder
last_login(last_login: String): User.UserBuilder
grade(grade: String): User.UserBuilder
build(): User
toString(): java.lang.String
com.ssafy.enjoytrip.dto.User
email: String
password: String
name: String
img: String
gender: char
grade_id: int
nickname: String
age: int
join_at: String
last_login: String
grade: String
User(email: String, password: String, name: String, img: String, gender: char, grade_id: int, nickname: String, age: int, join_at: String, last_login: String, grade: String)
builder(): User.UserBuilder
getEmail(): String
getPassword(): String
getName(): String
getImg(): String
getGender(): char
getGrade_id(): int
getNickname(): String
getAge(): int
getJoin_at(): String
getLast_login(): String
getGrade(): String
toString(): java.lang.String
com.ssafy.enjoytrip.model.repo.UserRepo
createUser(user: User): int
selectUserByEmail(email: String): User
selectAllUser(): List<User>
updateUser(user: User): int
deleteUser(email: String): int
com.ssafy.enjoytrip.model.service.UserServiceImpl
userRepo: UserRepo
createUser(user: User): int
selectUserByEmail(email: String): User
selectAllUser(): List<User>
updateUser(user: User): int
deleteUser(email: String): int
login(email: String, password: String): boolean
UserServiceImpl()

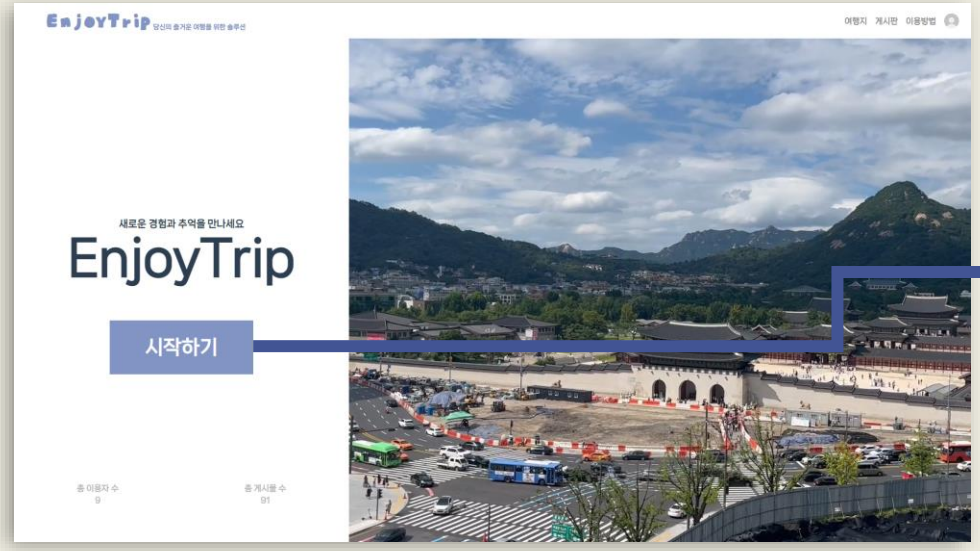

com.ssafy.enjoytrip.controller.BoardController
svc: BoardService
selectAll(): ResponseEntity<?>
selectBoardByCategory(boardCategoryId: int): ResponseEntity<?>
select(id: String): ResponseEntity<?>
insert(board: Board): ResponseEntity<?>
update(board: Board): ResponseEntity<?>
updateViewCount(id: String): ResponseEntity<?>
delete(id: String): ResponseEntity<?>
getAllStaredBoardByUserEmail(userEmail: String): ResponseEntity<?>
BoardController(svc: BoardService)
com.ssafy.enjoytrip.dto.Board.BoardBuilder
id: int
userEmail: String
title: String
content: String
createTime: String
updateTime: String
boardCategoryId: int
viewCount: int
BoardBuilder()
id(id: int): Board.BoardBuilder
userEmail(userEmail: String): Board.BoardBuilder
title(title: String): Board.BoardBuilder
content(content: String): Board.BoardBuilder
createTime(createTime: String): Board.BoardBuilder
updateTime(updateTime: String): Board.BoardBuilder
boardCategoryId(boardCategoryId: int): Board.BoardBuilder
viewCount(viewCount: int): Board.BoardBuilder
build(): Board
toString(): java.lang.String
com.ssafy.enjoytrip.dto.Board
id: int
userEmail: String
title: String
content: String
createTime: String
updateTime: String
boardCategoryId: int
viewCount: int
builder(): Board.BoardBuilder
getId(): int
getUserEmail(): String
getTitle(): String
getContent(): String
getCreateTime(): String
getUpdateTime(): String
getBoardCategoryId(): int
getViewCount(): int
setId(id: int): void
setUserEmail(userEmail: String): void
setTitle(title: String): void
setContent(content: String): void
setCreateTime(createTime: String): void
setUpdateTime(updateTime: String): void
setBoardCategoryId(boardCategoryId: int): void
setViewCount(viewCount: int): void
Board(id: int, userEmail: String, title: String, content: String, createTime: String, updateTime: String, boardCategoryId: int, viewCount: int)
Board()
toString(): java.lang.String
com.ssafy.enjoytrip.model.repo.BoardRepo
insert(board: Board): int
update(board: Board): int
updateViewCount(id: String): int
delete(id: String): int
select(id: String): Board
selectAll(): List<Board>
selectBoardByCategory(boardCategoryId: int): List<Board>
getAllStaredBoardByUserEmail(userEmail: String): List<Board>
com.ssafy.enjoytrip.model.service.BoardServiceImpl
repo: BoardRepo
insert(board: Board): int
update(board: Board): int
updateViewCount(id: String): int
delete(id: String): int
select(id: String): Board
selectAll(): List<Board>
selectBoardByCategory(boardCategoryId: int): List<Board>
getAllStaredBoardByUserEmail(userEmail: String): List<Board>
BoardServiceImpl(repo: BoardRepo)

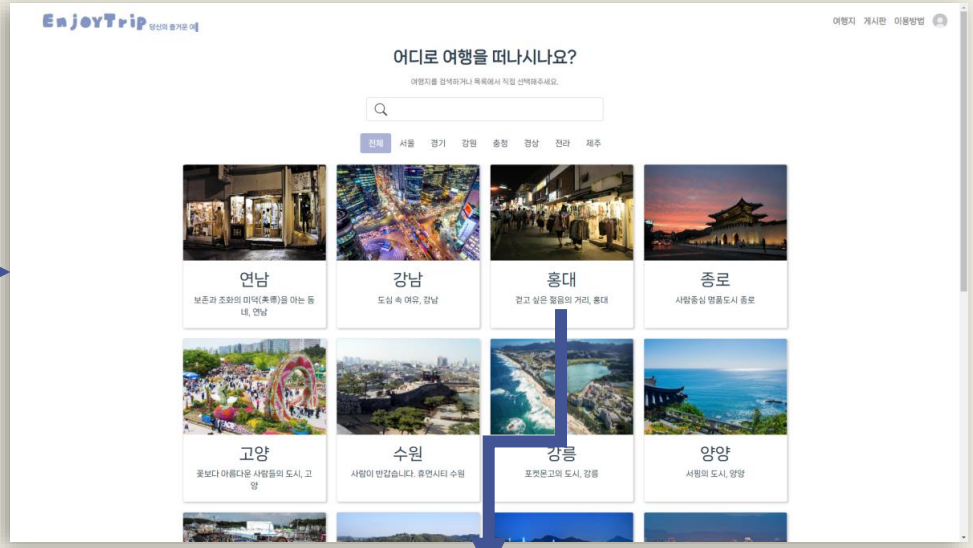

com.ssafy.enjoytrip.controller.ReviewController
reviewService: ReviewService
postReview(review: Review): ResponseEntity<?>
selectReviewsByID(id: int): ResponseEntity<?>
ReviewController(reviewService: ReviewService)
com.ssafy.enjoytrip.model.repo.ReviewRepo
selectReviewsByID(id: int): List<Review>
insertReview(review: Review): int
com.ssafy.enjoytrip.model.service.ReviewServiceImpl
reviewRepo: ReviewRepo
selectReviewsByID(id: int): List<Review>
insertReview(review: Review): int
ReviewServiceImpl(reviewRepo: ReviewRepo)
com.ssafy.enjoytrip.dto.Review
id: int
review: String
placeId: int
userEmail: String
userNickName: String
createdAt: String
getId(): int
getReview(): String
getPlaceId(): int
getUserEmail(): String
getUserNickName(): String
getCreatedAt(): String
setId(id: int): void
setReview(review: String): void
setPlaceId(placeId: int): void
setUserEmail(userEmail: String): void
setUserNickName(userNickName: String): void
setCreatedAt(createdAt: String): void
Review(id: int, review: String, placeId: int, userEmail: String, userNickName: String, createdAt: String)
Review()
toString(): java.lang.String

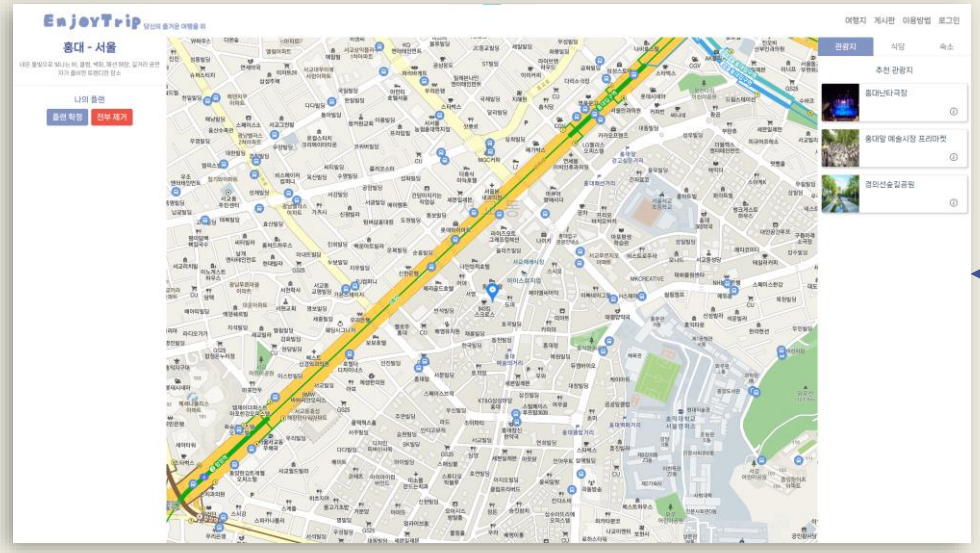

com.ssafy.enjoytrip.controller.PlanController
planService: PlanService
selectPlansByEmail(email: String): ResponseEntity<?>
insertPlan(plan: Plan): ResponseEntity<?>
deleteUser(id: int): ResponseEntity<?>
PlanController(planService: PlanService)
com.ssafy.enjoytrip.model.repo.PlanRepo
insertPlan(plan: Plan): int
selectPlansByEmail(email: String): List<Plan>
deletePlan(id: int): int
com.ssafy.enjoytrip.dto.Plan
id: int
userEmail: String
naviInfo: String
getId(): int
getUserEmail(): String
getNaviInfo(): String
setId(id: int): void
setUserEmail(userEmail: String): void
setNaviInfo(naviInfo: String): void
Plan(id: int, userEmail: String, naviInfo: String)
Plan()
toString(): java.lang.String
com.ssafy.enjoytrip.model.service.PlanServiceImpl
repo: PlanRepo
insertPlan(plan: Plan): int
selectPlansByEmail(email: String): List<Plan>
deletePlan(id: int): int
PlanServiceImpl(repo: PlanRepo)

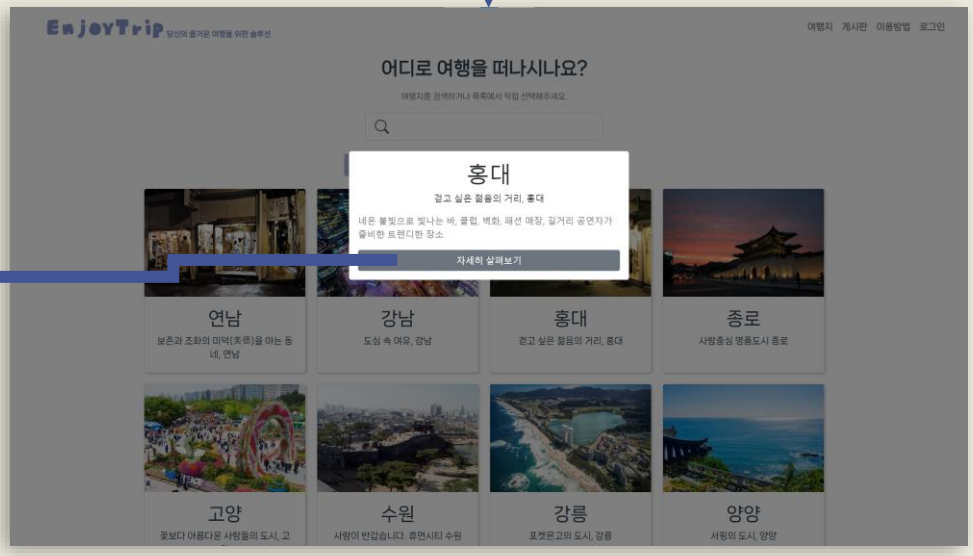

com.ssafy.enjoytrip.dto.Star
board_id: int
userEmail: String
builder(): Star.StarBuilder
getBoard_id(): int
getUserEmail(): String
setBoard_id(board_id: int): void
setUserEmail(userEmail: String): void
Star(board_id: int, userEmail: String)
Star()
toString(): java.lang.String
com.ssafy.enjoytrip.dto.Star.StarBuilder
board_id: int
userEmail: String
StarBuilder()
board_id(board_id: int): Star.StarBuilder
userEmail(userEmail: String): Star.StarBuilder
build(): Star
toString(): java.lang.String
com.ssafy.enjoytrip.controller.StarController
starService: StarService
selectAll(): ResponseEntity<?>
createStar(star: Star): ResponseEntity<?>
selectAllByEmail(email: String): ResponseEntity<?>
selectAllByBoardId(board_id: Integer): ResponseEntity<?>
deleteStar(star: Star): ResponseEntity<?>
StarController(starService: StarService)
com.ssafy.enjoytrip.model.repo.StarRepo
createStar(star: Star): int
selectUserByIdandEmail(board_id: int, user_email: String): Star
selectAll(): List<Star>
selectAllByEmail(email: String): List<Star>
selectAllByBoardId(board_id: int): List<Star>
deleteStar(star: Star): int
com.ssafy.enjoytrip.model.service.StarServiceImpl
starRepo: StarRepo
createStar(star: Star): int
selectUserByIdandEmail(board_id: int, user_email: String): Star
selectAll(): List<Star>
selectAllByEmail(email: String): List<Star>
selectAllByBoardId(board_id: int): List<Star>
deleteStar(star: Star): int
StarServiceImpl()

화면 흐름도 (메인 > 여행지 설정)

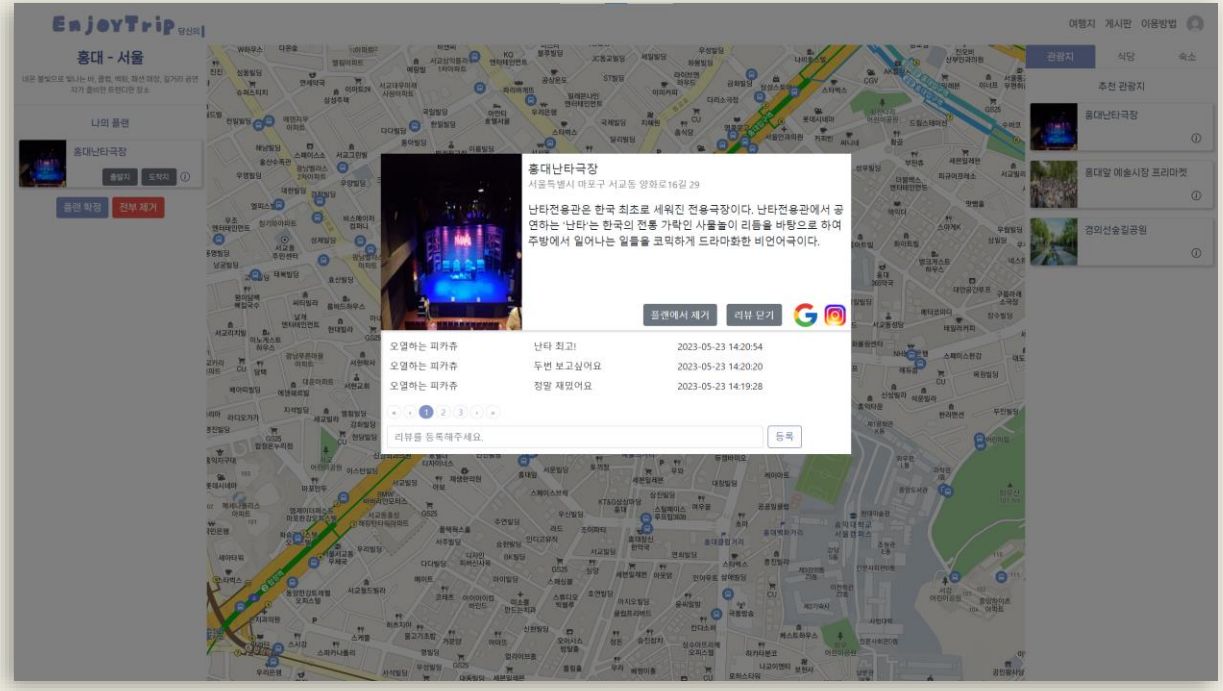


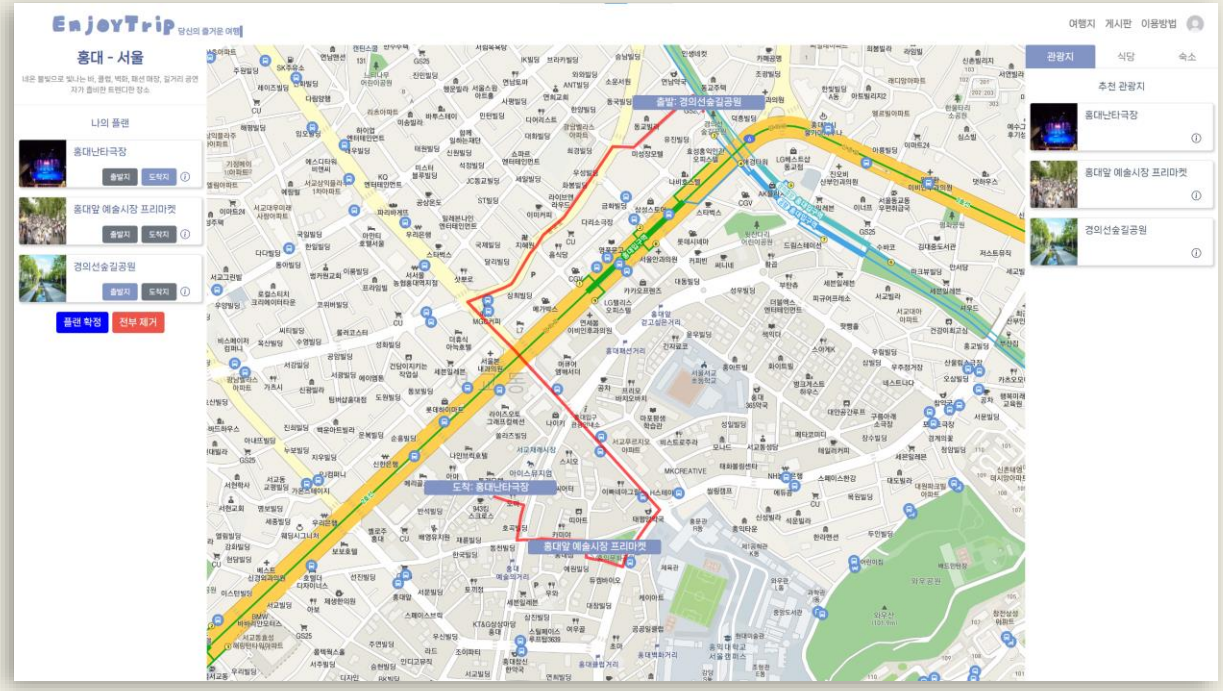






여행지 경로 설정

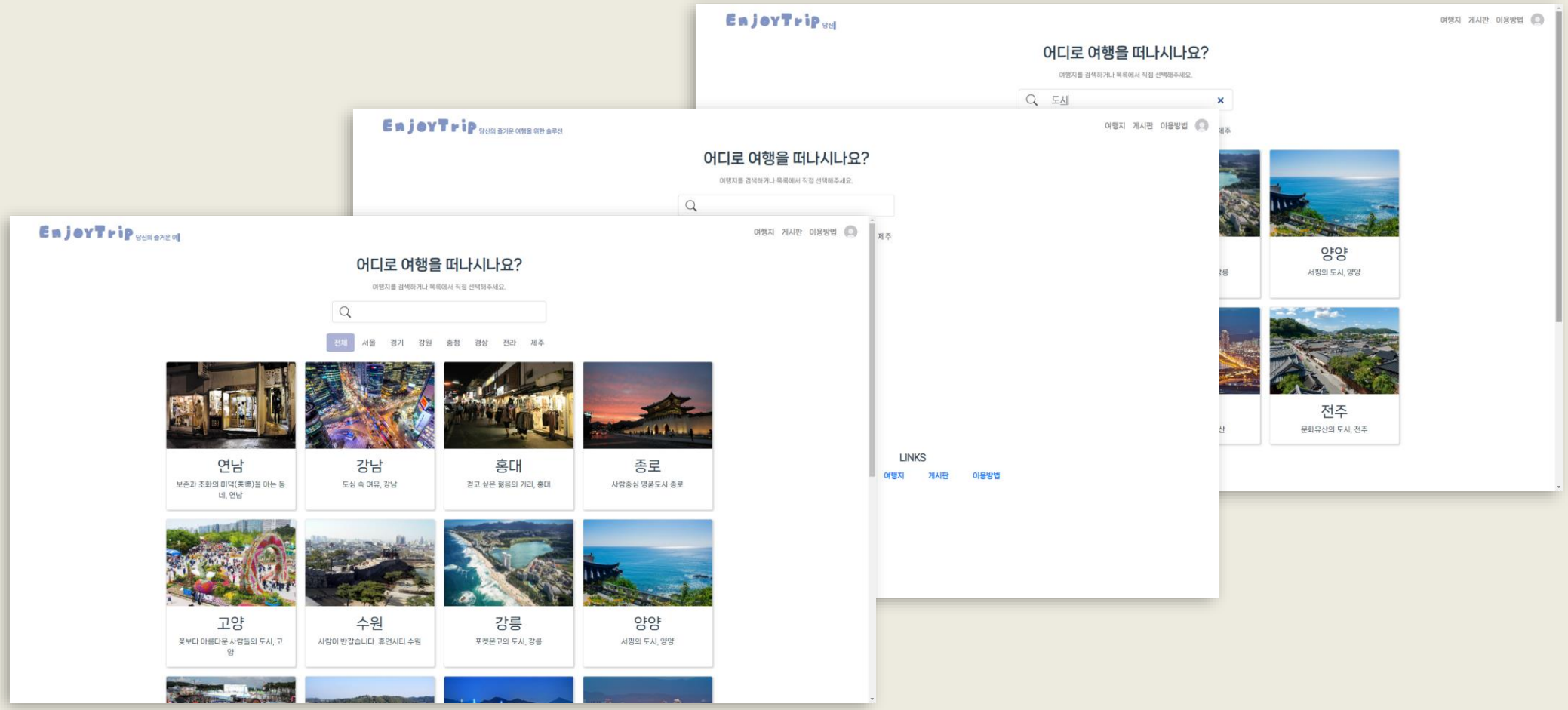




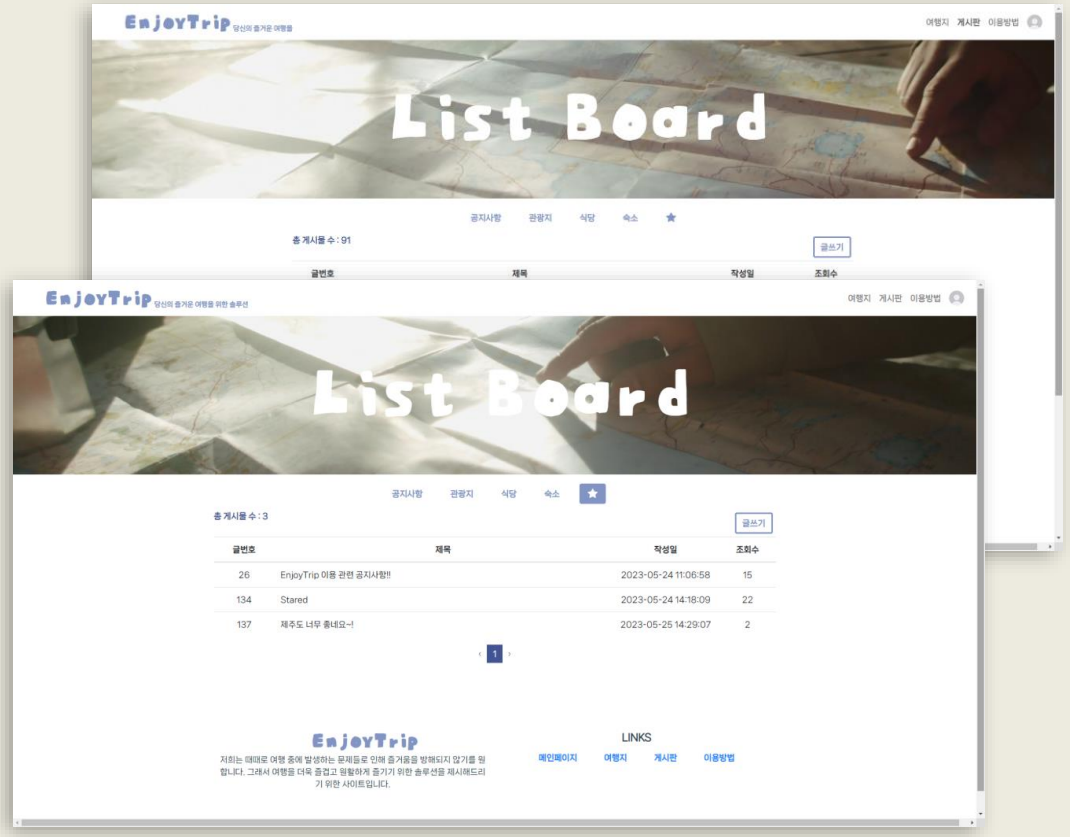
원하는 여행지 설정

설정된 여행지 순서로 길찾기

여행지 검색

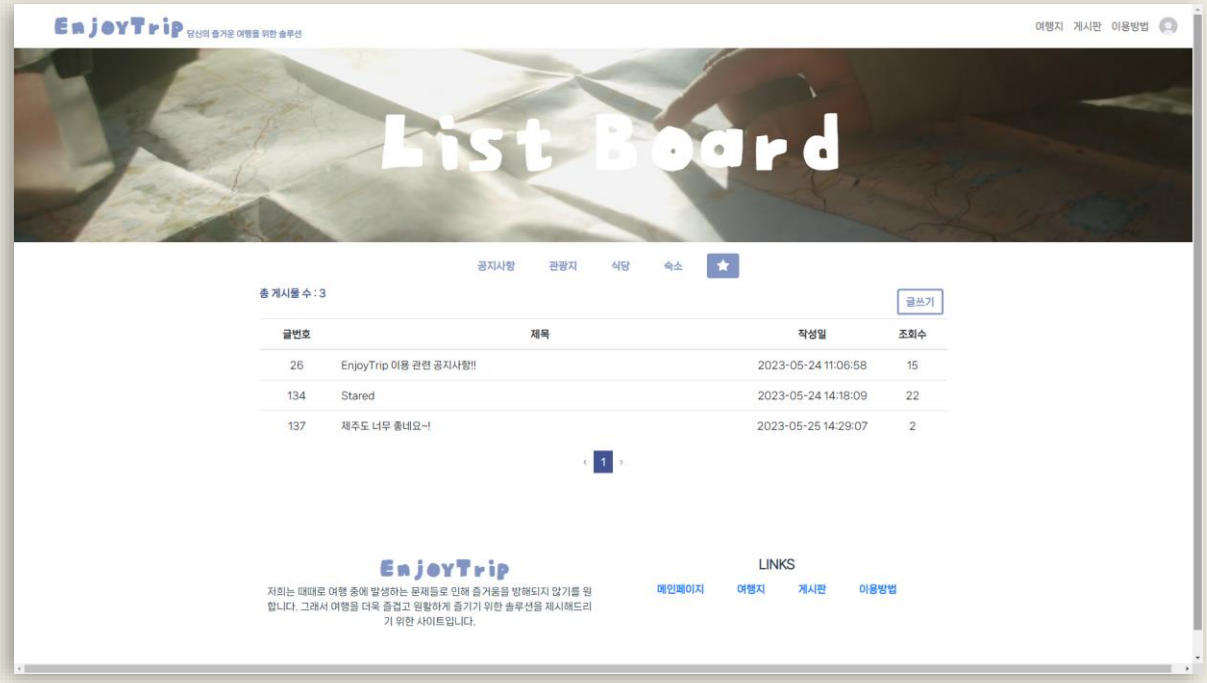

EnjoyTrip
여행지 게시판 이용방법
어디로 여행을 떠나시나요?
여행지를 검색하거나 목록에서 직접 선택해주세요.
도시
양양
서핑의 도시, 양양
전주
문화유산의 도시, 전주
LINKS
여행지 게시판 이용방법
어디로 여행을 떠나시나요?
여행지를 검색하거나 목록에서 직접 선택해주세요.
어디로 여행을 떠나시나요?
여행지를 검색하거나 목록에서 직접 선택해주세요.
전체 서울 경기 강원 충청 경상 전라 제주
연남
보존과 조화의 미덕(美德)을 아는 동네, 연남
강남
도심 속 여유, 강남
홍대
걷고 싶은 젊음의 거리, 홍대
종로
사람중심 명품도시 종로
고양
꽃보다 아름다운 사람들의 도시, 고양
수원
사람이 반갑습니다. 휴면시티 수원
강릉
포켓몬고의 도시, 강릉
양양
서핑의 도시, 양양

화면 흐름도 (게시판 화면)


EnjoyTrip
여행지 게시판 이용방법
List Board
공지사항 관광지 식당 숙소
총 게시물 수 : 91
글쓰기
글번호 제목 작성일 조회수
EnjoyTrip
여행지 게시판 이용방법
List Board
공지사항 관광지 식당 숙소
총 게시물 수 : 3
글쓰기
글번호 제목 작성일 조회수
26 EnjoyTrip 이용 관련 공지사항!! 2023-05-24 11:06:58 15
134 Stared 2023-05-24 14:18:09 22
137 제주도 너무 좋네요~! 2023-05-25 14:29:07 2
1
EnjoyTrip
LINKS
메인페이지 여행지 게시판 이용방법

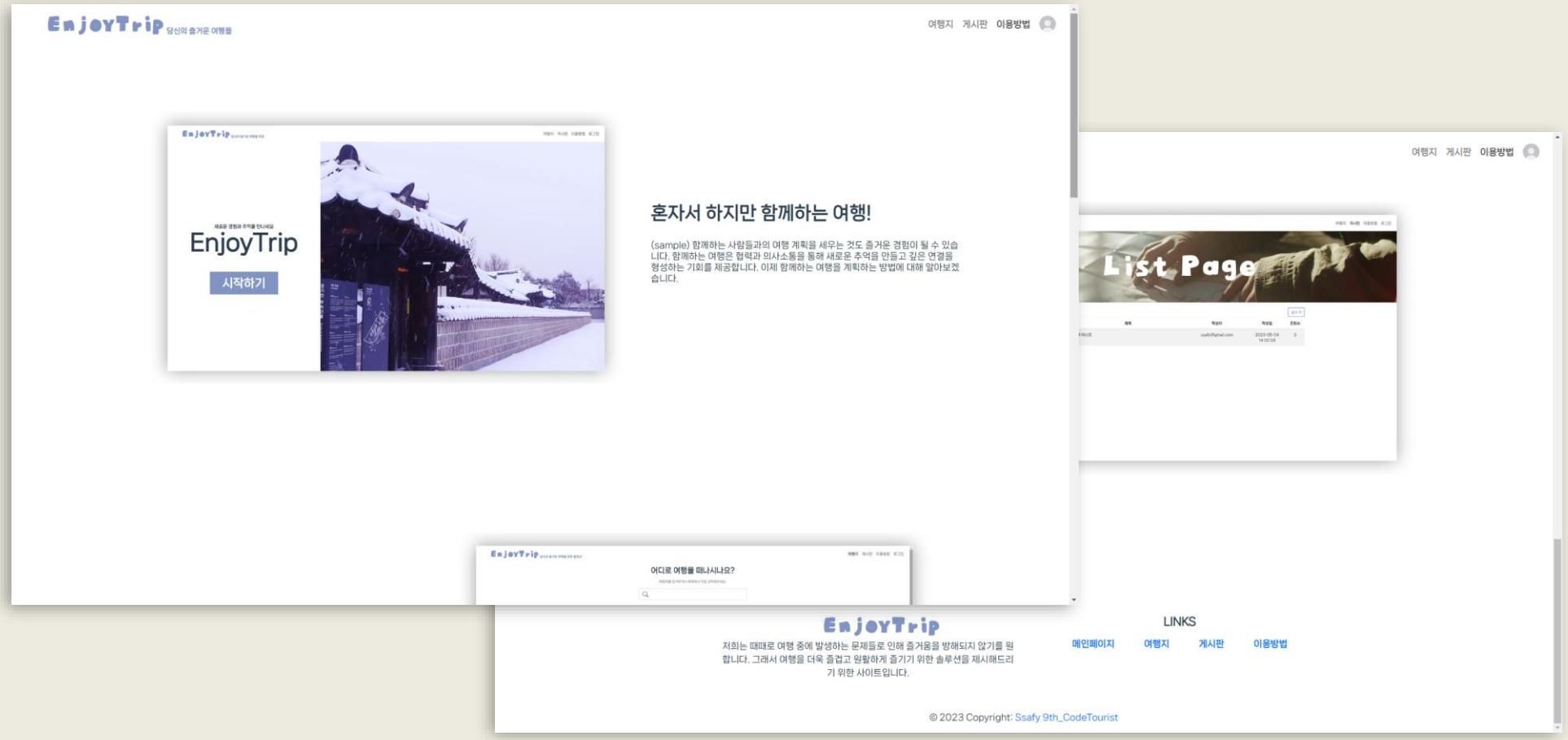
EnjoyTrip
여행지 게시판 이용방법
view
Stared
[어드민]
2023-05-24 14:18:09 조회수 22
Stared
EnjoyTrip
여행지 게시판 이용방법
view
제주도 너무 좋네요~!
[어드민]
2023-05-25 14:29:07 조회수 2
제주도
제주도 너무 좋아요
목록으로
글수정 글삭제

게시판 세부

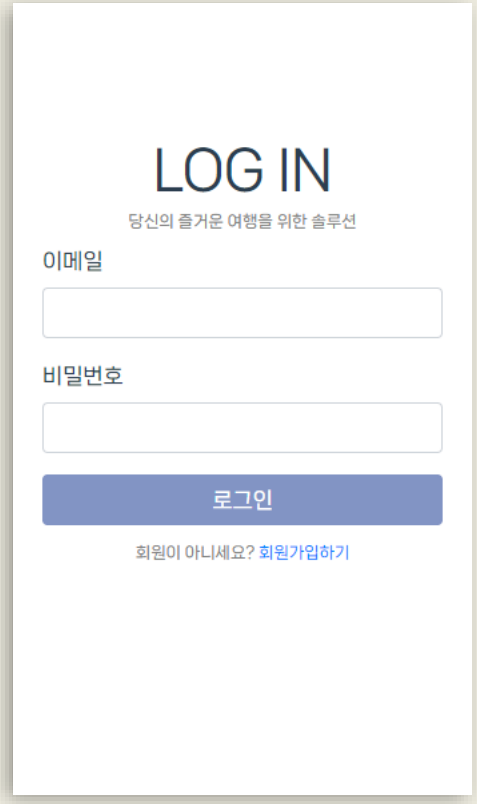

EnjoyTrip 당신의 즐거운 여행을 위한 솔루션
여행지 게시판 이용방법
List Board
공지사항 관광지 식당 숙소
총 게시물 수 : 3
글쓰기
글번호 제목 작성일 조회수
26 EnjoyTrip 이용 관련 공지사항!! 2023-05-24 11:06:58 15
134 Stared 2023-05-24 14:18:09 22
137 제주도 너무 좋네요~! 2023-05-25 14:29:07 2
1
EnjoyTrip
저희는 때때로 여행 중에 발생하는 문제들로 인해 즐거움을 방해되지 않기를 원합니다. 그래서 여행을 더욱 즐겁고 원활하게 즐기기 위한 솔루션을 제시해드리기 위한 사이트입니다.
LINKS
메인페이지 여행지 게시판 이용방법

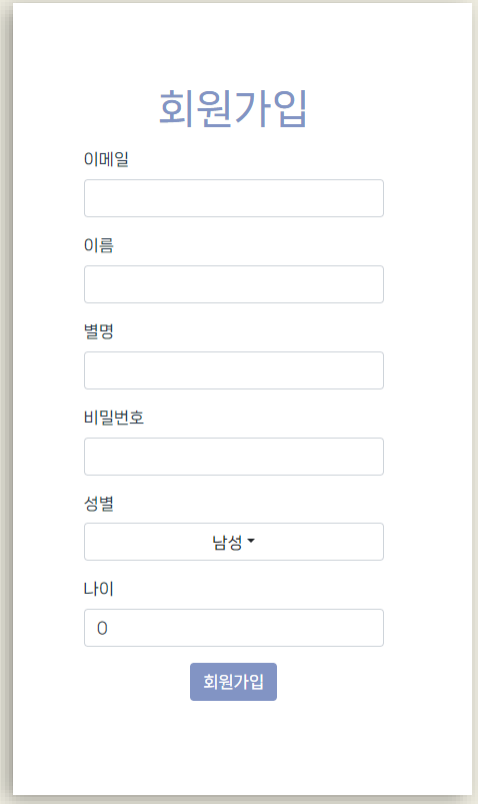
EnjoyTrip 당신의 즐거운 여행을 위한 솔루션
여행지 게시판 이용방법
제목 수정 테스트
[싸순이]
2023-05-04 14:06:58 조회수 15
내용 테스트
아직 댓글이 없습니다 첫 댓글을 작성해주세요
리뷰를 등록해주세요.
등록
목록으로
글수정 글삭제

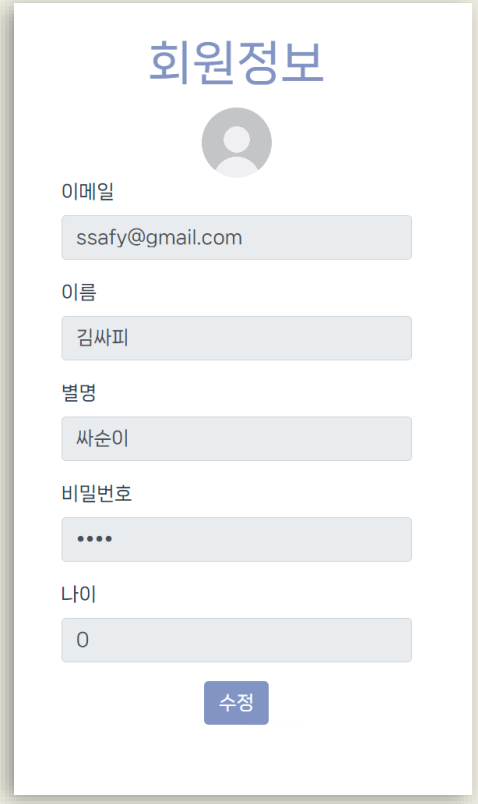
간단한 사이트 이용방법 설명



회원정보 관리


LOG IN
당신의 즐거운 여행을 위한 솔루션
이메일
비밀번호
로그인
회원이 아니세요? 회원가입하기


회원가입
이메일
이름
별명
비밀번호
성별
남성
나이
0
회원가입


회원정보
이메일
ssafy@gmail.com
이름
김싸피
별명
싸순이
비밀번호
••••
나이
0
수정

회원정보
이메일
ssafy@gmail.com
이름
김싸피
별명
싸순이
비밀번호
1234
나이
0
완료
취소

기대효과

기대효과

여행을 하기 위해서 다양한 사이트들을 돌아다녀야
했던 불편함을 개선하고, 한 곳에서 다양한 정보들을
모아서 관리할 수 있게 하여,
사람들의 귀찮음을 줄어줄 수 있을 것입니다.

개발후기

개발 후기 (이동영)

평소에 프론트엔드 작업을 많이 하지 않았는데
이번 프로젝트를 진행하면서
프론트엔드 작업에 재미를 붙이게 되었습니다.

짧은 기간동안 한 개발이라 아쉬움도 남지만
페어와 함께 프로그래밍 하는 즐거움을 느낄 수 있었습니다.

개발 후기 (이정명)

사실상 비전공자로서, 웹이라는 분야에 대해서
1학기라는 반년의 기간동안 배운 모든 것들을 쏟아부어
진행한 프로젝트였습니다.

그래서 그런지 부족한 점도 많았고, 벽도 많이 만났지만,
팀원의 도움으로 많이 성장하며 생각 이상의 프로젝트를
진행할 수 있었던 것 같습니다.

프로젝트를 진행하기 이전에는 과연 내 실력으로
정상적으로 끝마칠 수 있을까 하는 생각이 들었지만,
만족할 정도라고 감히 말하기는 힘들지만, 그래도
생각 이상으로 진행할 수 있었기에 만족하며 진행할 수
있었던 프로젝트였습니다.

참조

참고자료

DATA & TOURISM (여행업의 Next Level) Vol.03 보고서
한국관광공사

TTS

Typecast – 호빈이
https://app.typecast.ai/ko/editor?projectId=P31qqvL8QlKeKlKUcSJnF

Fonts (문서)

Swagger

Fonts (프로젝트)

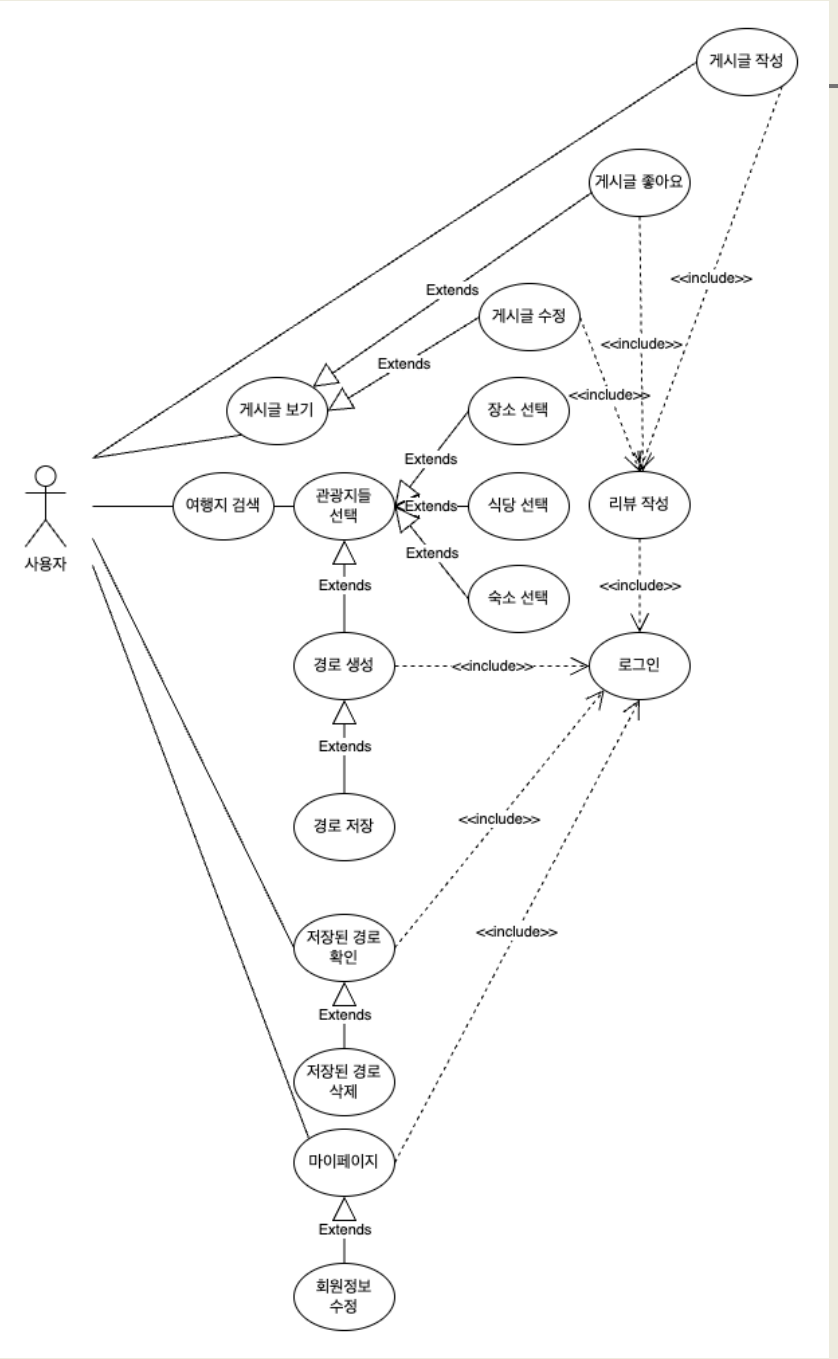
제주돌담체 나눔스퀘어 네오

Color

Document	Project
51, 38, 132	#8294C4
115,115,115	#ACB1D6
237,235,223	#DBDFEA
	#FFEAD2

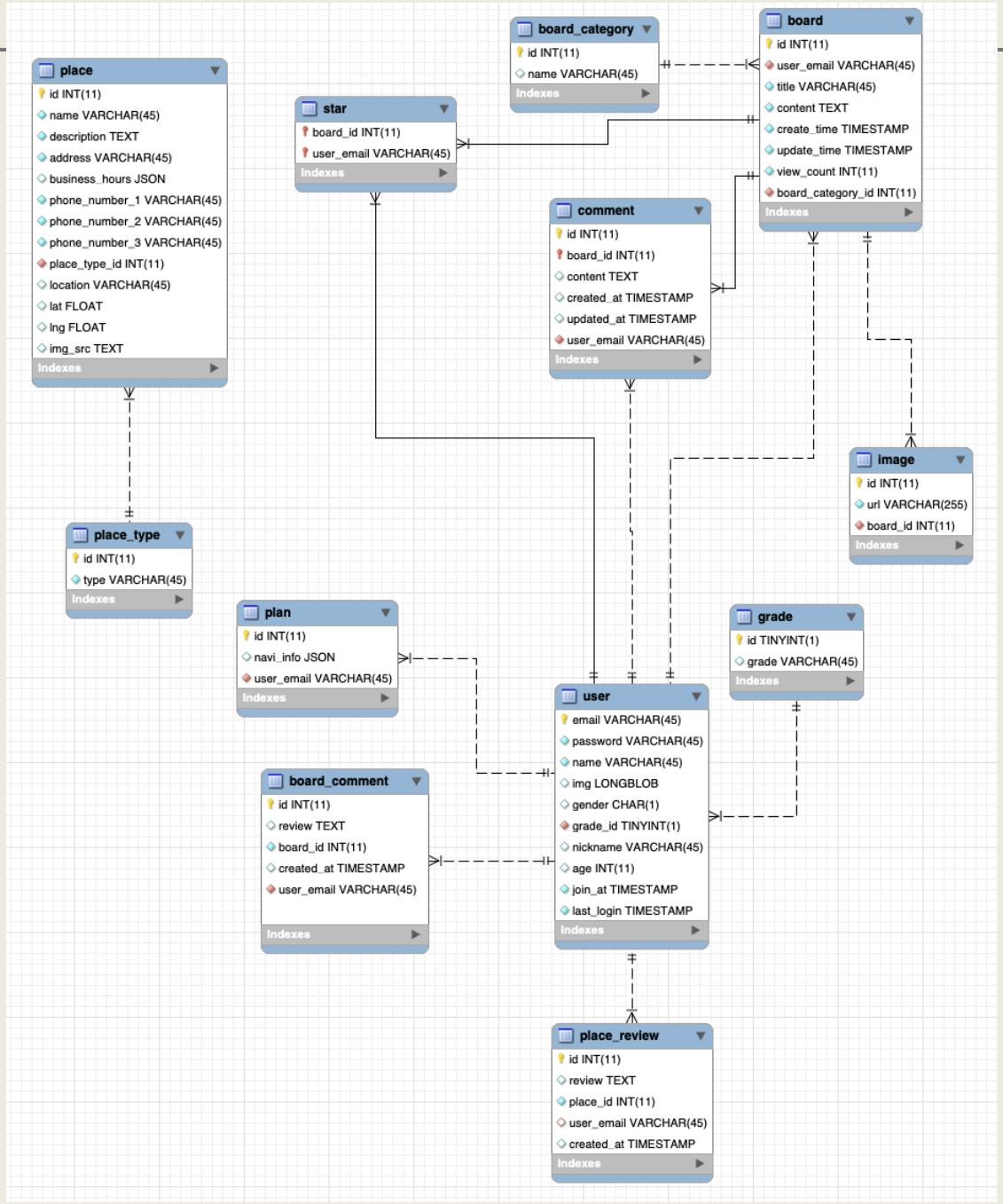
참조

UML
(Usecase Diagram)

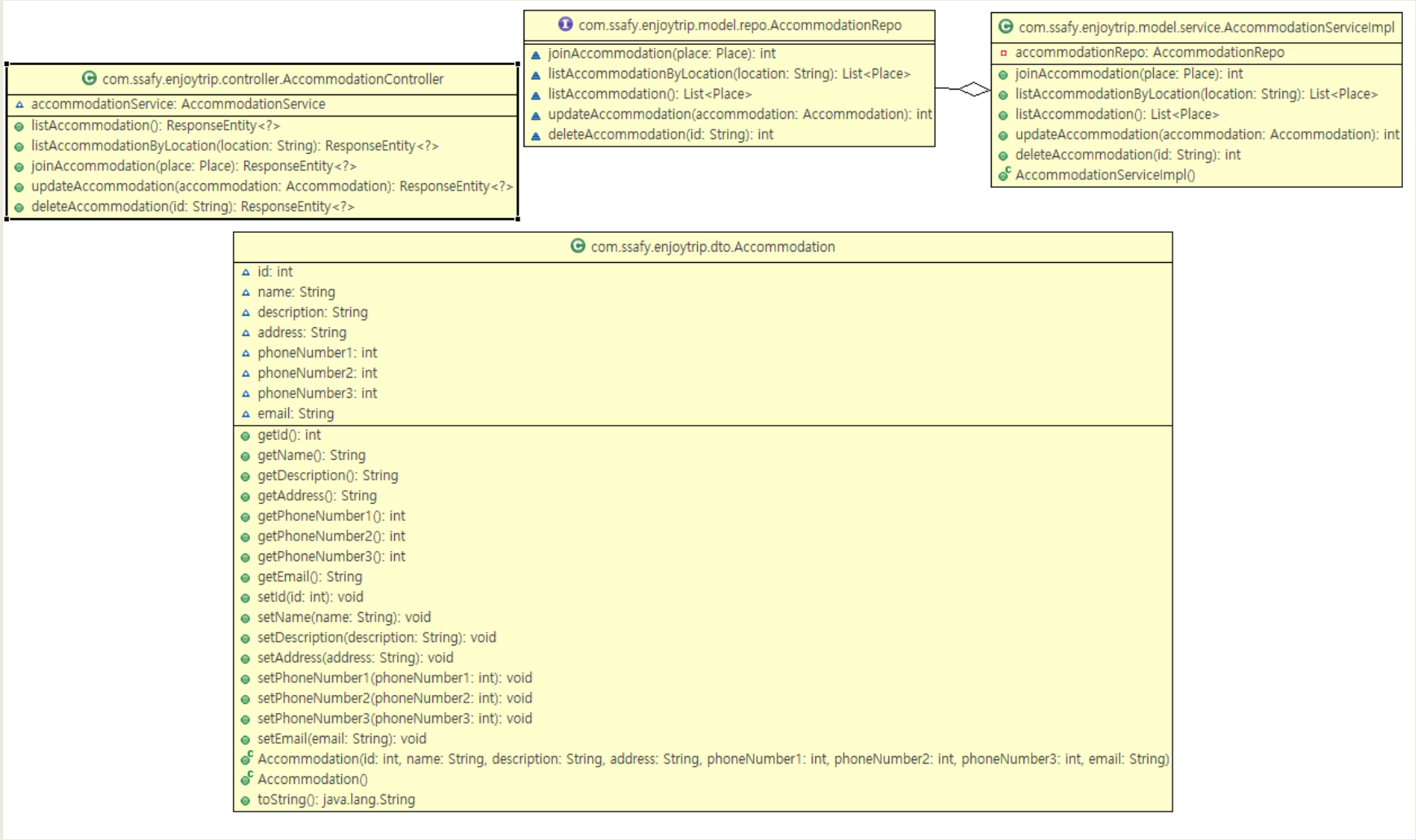


잡조

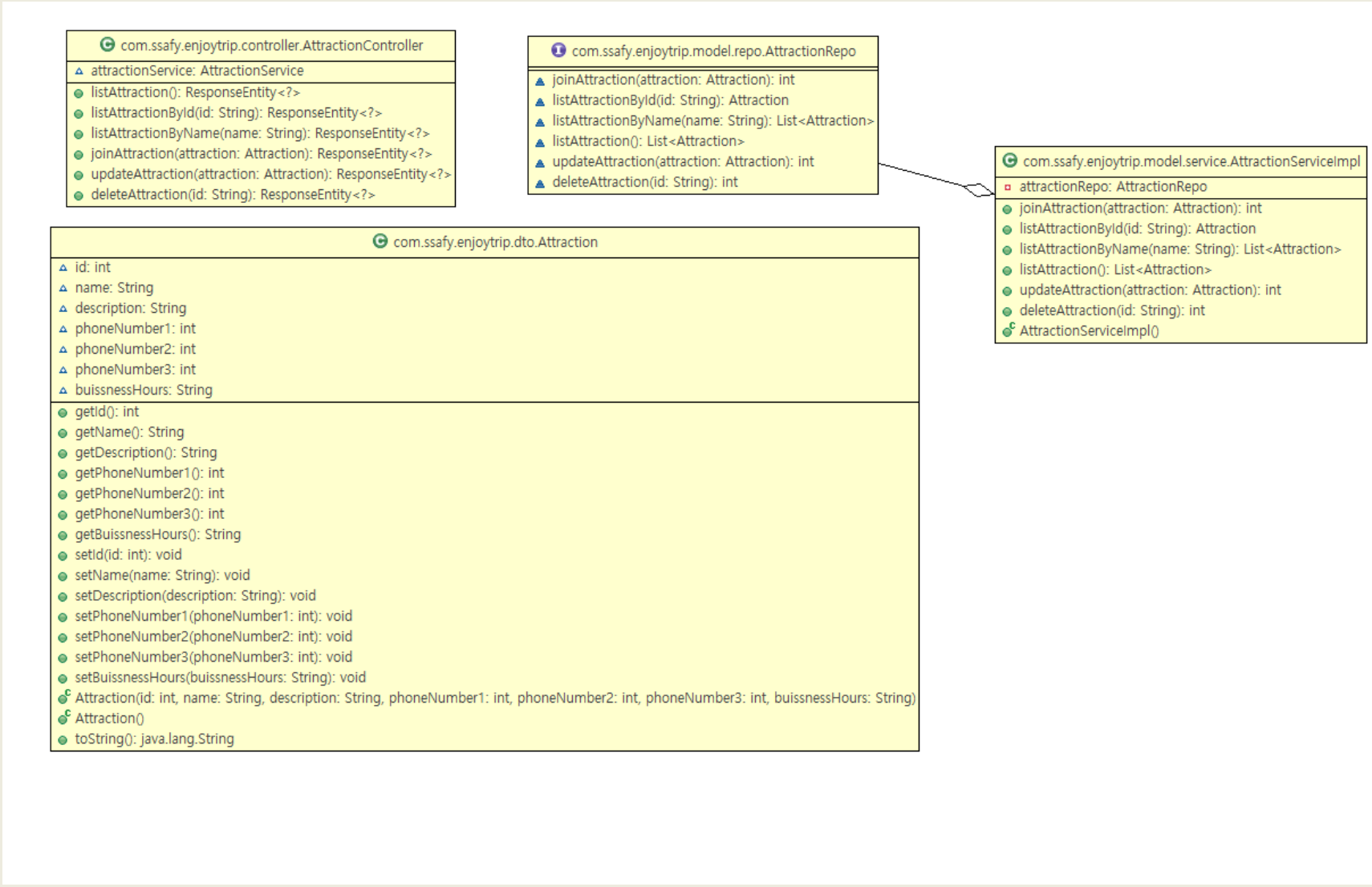
ERD
(Entity Relationship Diagram)



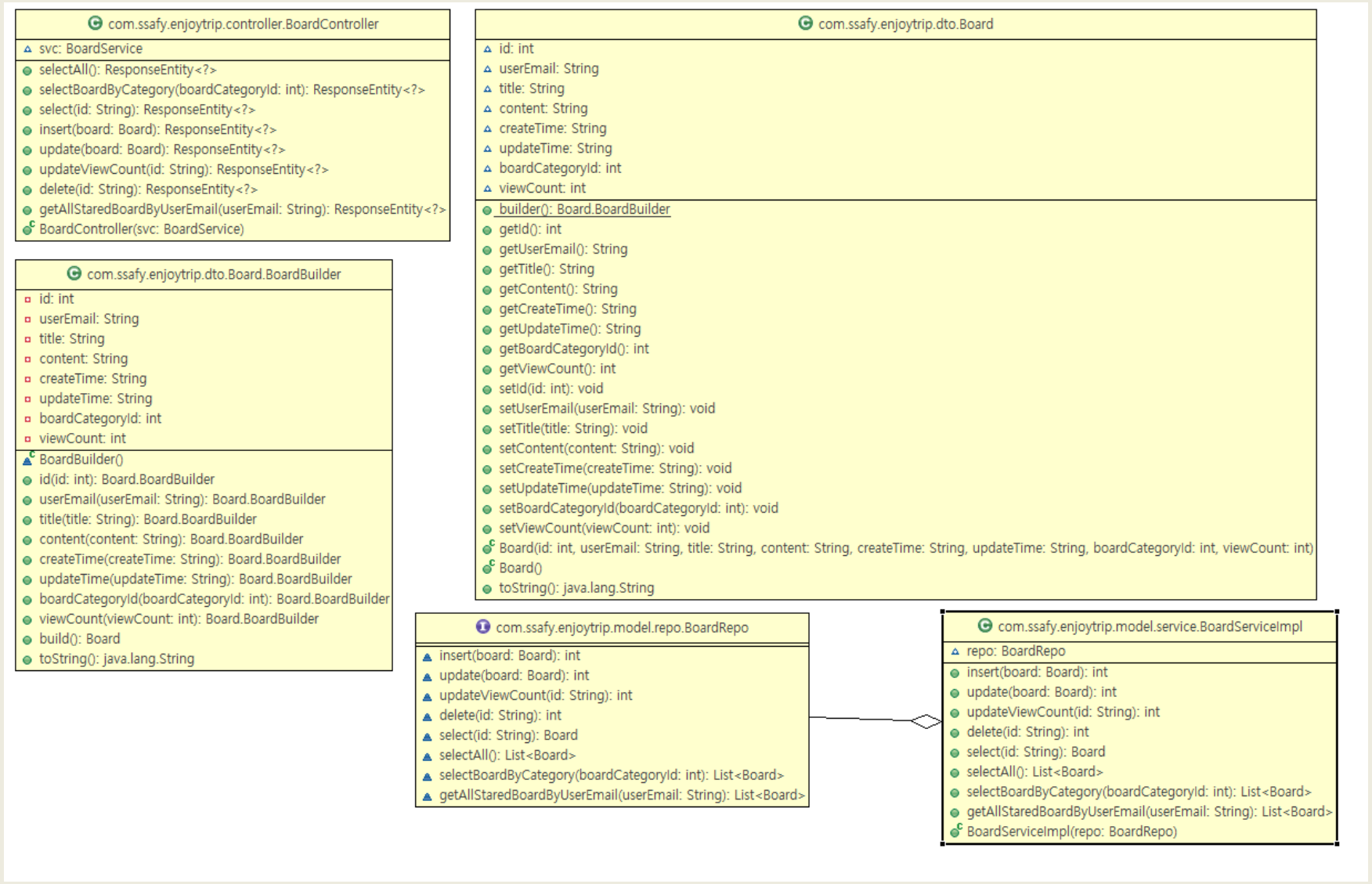
accomodation



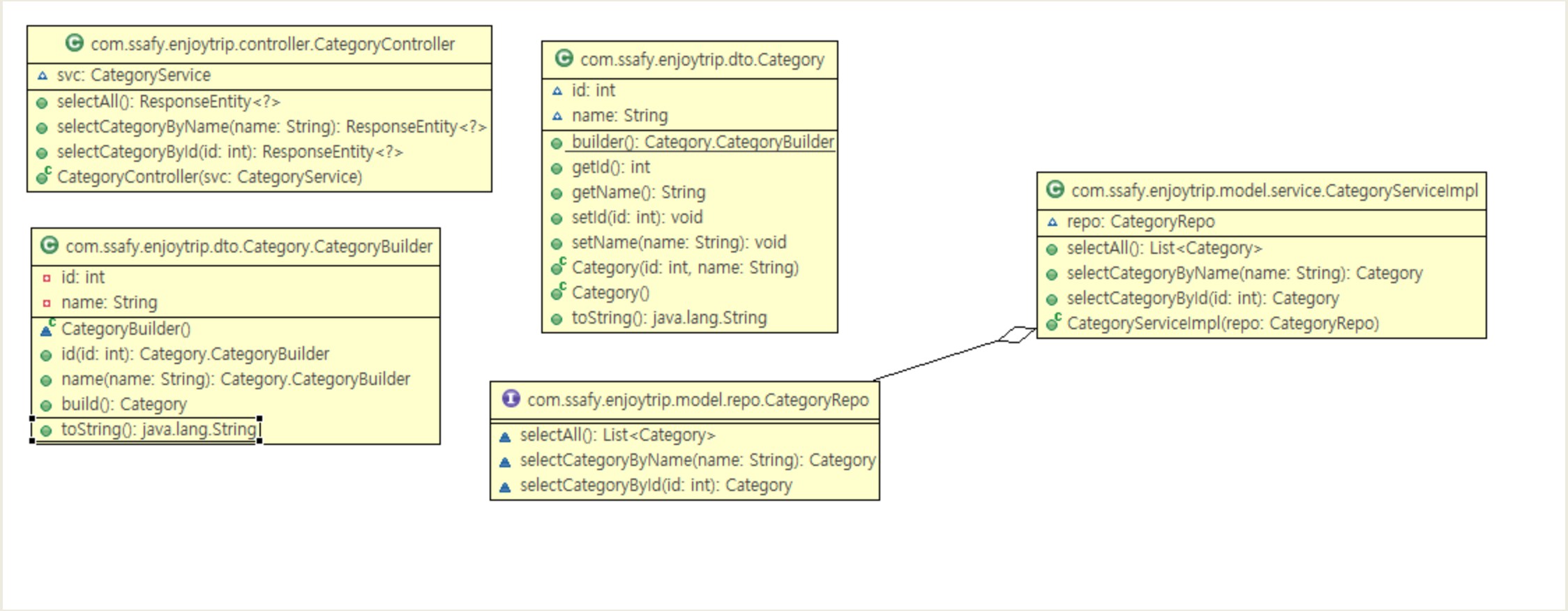
attraction


com.ssafy.enjoytrip.controller.AttractionController
attractionService: AttractionService
listAttraction(): ResponseEntity<?>
listAttractionById(id: String): ResponseEntity<?>
listAttractionByName(name: String): ResponseEntity<?>
joinAttraction(attraction: Attraction): ResponseEntity<?>
updateAttraction(attraction: Attraction): ResponseEntity<?>
deleteAttraction(id: String): ResponseEntity<?>
com.ssafy.enjoytrip.model.repo.AttractionRepo
joinAttraction(attraction: Attraction): int
listAttractionById(id: String): Attraction
listAttractionByName(name: String): List<Attraction>
listAttraction(): List<Attraction>
updateAttraction(attraction: Attraction): int
deleteAttraction(id: String): int
com.ssafy.enjoytrip.model.service.AttractionServiceImpl
attractionRepo: AttractionRepo
joinAttraction(attraction: Attraction): int
listAttractionById(id: String): Attraction
listAttractionByName(name: String): List<Attraction>
listAttraction(): List<Attraction>
updateAttraction(attraction: Attraction): int
deleteAttraction(id: String): int
AttractionServiceImpl()
com.ssafy.enjoytrip.dto.Attraction
id: int
name: String
description: String
phoneNumber1: int
phoneNumber2: int
phoneNumber3: int
buissnessHours: String
getId(): int
getName(): String
getDescription(): String
getPhoneNumber1(): int
getPhoneNumber2(): int
getPhoneNumber3(): int
getBuissnessHours(): String
setId(id: int): void
setName(name: String): void
setDescription(description: String): void
setPhoneNumber1(phoneNumber1: int): void
setPhoneNumber2(phoneNumber2: int): void
setPhoneNumber3(phoneNumber3: int): void
setBuissnessHours(buissnessHours: String): void
Attraction(id: int, name: String, description: String, phoneNumber1: int, phoneNumber2: int, phoneNumber3: int, buissnessHours: String)
Attraction()
toString(): java.lang.String

board

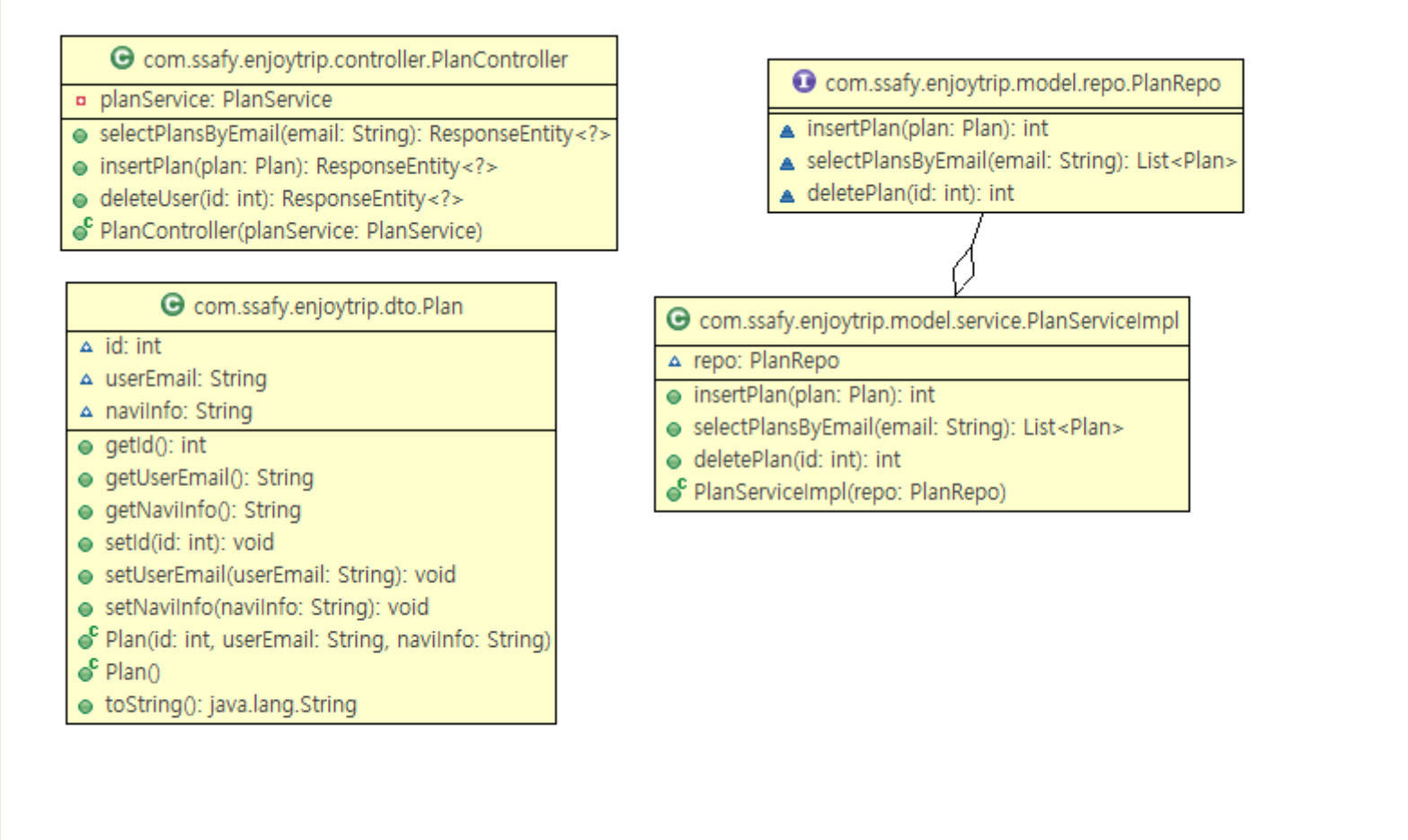

com.ssafy.enjoytrip.controller.BoardController
svc: BoardService
selectAll(): ResponseEntity<?>
selectBoardByCategory(boardCategoryId: int): ResponseEntity<?>
select(id: String): ResponseEntity<?>
insert(board: Board): ResponseEntity<?>
update(board: Board): ResponseEntity<?>
updateViewCount(id: String): ResponseEntity<?>
delete(id: String): ResponseEntity<?>
getAllStaredBoardByUserEmail(userEmail: String): ResponseEntity<?>
BoardController(svc: BoardService)
com.ssafy.enjoytrip.dto.Board
id: int
userEmail: String
title: String
content: String
createTime: String
updateTime: String
boardCategoryId: int
viewCount: int
builder(): Board.BoardBuilder
getId(): int
getUserEmail(): String
getTitle(): String
getContent(): String
getCreateTime(): String
getUpdateTime(): String
getBoardCategoryId(): int
getViewCount(): int
setId(id: int): void
setUserEmail(userEmail: String): void
setTitle(title: String): void
setContent(content: String): void
setCreateTime(createTime: String): void
setUpdateTime(updateTime: String): void
setBoardCategoryId(boardCategoryId: int): void
setViewCount(viewCount: int): void
Board(id: int, userEmail: String, title: String, content: String, createTime: String, updateTime: String, boardCategoryId: int, viewCount: int)
Board()
toString(): java.lang.String
com.ssafy.enjoytrip.dto.Board.BoardBuilder
id: int
userEmail: String
title: String
content: String
createTime: String
updateTime: String
boardCategoryId: int
viewCount: int
BoardBuilder()
id(id: int): Board.BoardBuilder
userEmail(userEmail: String): Board.BoardBuilder
title(title: String): Board.BoardBuilder
content(content: String): Board.BoardBuilder
createTime(createTime: String): Board.BoardBuilder
updateTime(updateTime: String): Board.BoardBuilder
boardCategoryId(boardCategoryId: int): Board.BoardBuilder
viewCount(viewCount: int): Board.BoardBuilder
build(): Board
toString(): java.lang.String
com.ssafy.enjoytrip.model.repo.BoardRepo
insert(board: Board): int
update(board: Board): int
updateViewCount(id: String): int
delete(id: String): int
select(id: String): Board
selectAll(): List<Board>
selectBoardByCategory(boardCategoryId: int): List<Board>
getAllStaredBoardByUserEmail(userEmail: String): List<Board>
com.ssafy.enjoytrip.model.service.BoardServiceImpl
repo: BoardRepo
insert(board: Board): int
update(board: Board): int
updateViewCount(id: String): int
delete(id: String): int
select(id: String): Board
selectAll(): List<Board>
selectBoardByCategory(boardCategoryId: int): List<Board>
getAllStaredBoardByUserEmail(userEmail: String): List<Board>
BoardServiceImpl(repo: BoardRepo)

category

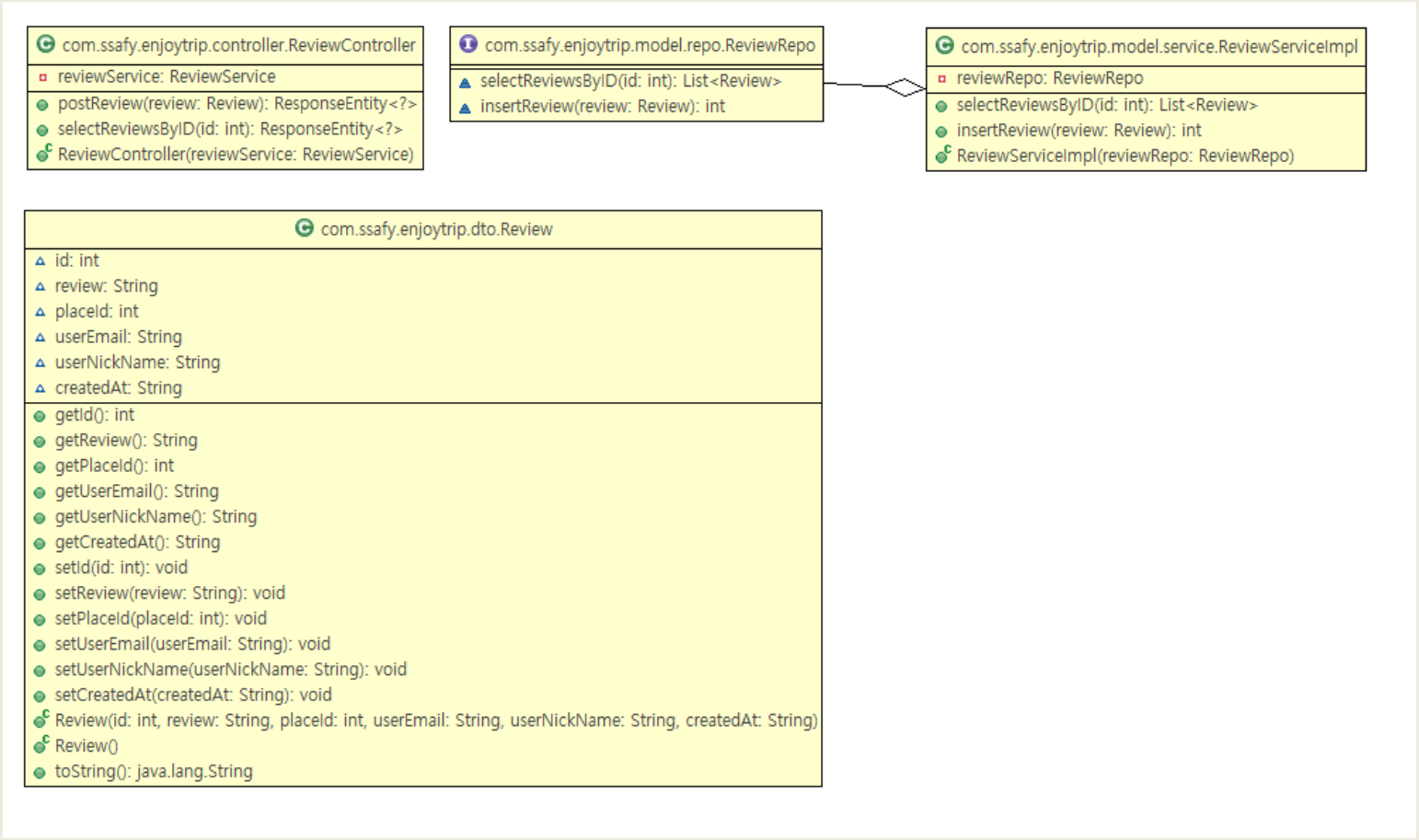

com.ssafy.enjoytrip.controller.CategoryController
svc: CategoryService
selectAll(): ResponseEntity<?>
selectCategoryByName(name: String): ResponseEntity<?>
selectCategoryById(id: int): ResponseEntity<?>
CategoryController(svc: CategoryService)
com.ssafy.enjoytrip.dto.Category
id: int
name: String
builder(): Category.CategoryBuilder
getId(): int
getName(): String
setId(id: int): void
setName(name: String): void
Category(id: int, name: String)
Category()
toString(): java.lang.String
com.ssafy.enjoytrip.model.service.CategoryServiceImpl
repo: CategoryRepo
selectAll(): List<Category>
selectCategoryByName(name: String): Category
selectCategoryById(id: int): Category
CategoryServiceImpl(repo: CategoryRepo)
com.ssafy.enjoytrip.dto.Category.CategoryBuilder
id: int
name: String
CategoryBuilder()
id(id: int): Category.CategoryBuilder
name(name: String): Category.CategoryBuilder
build(): Category
toString(): java.lang.String
com.ssafy.enjoytrip.model.repo.CategoryRepo
selectAll(): List<Category>
selectCategoryByName(name: String): Category
selectCategoryById(id: int): Category

참조

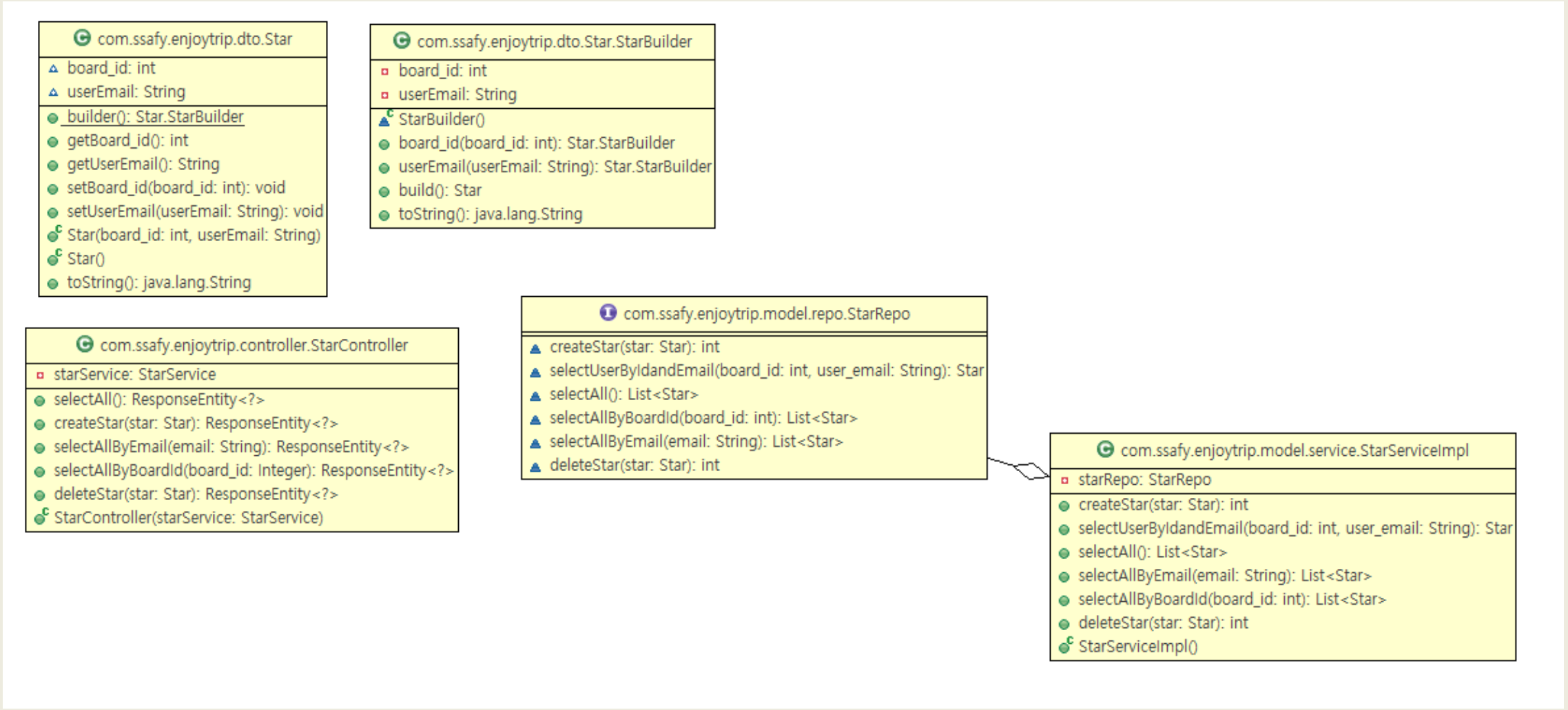
plan


com.ssafy.enjoytrip.controller.PlanController
planService: PlanService
selectPlansByEmail(email: String): ResponseEntity<?>
insertPlan(plan: Plan): ResponseEntity<?>
deleteUser(id: int): ResponseEntity<?>
PlanController(planService: PlanService)
com.ssafy.enjoytrip.model.repo.PlanRepo
insertPlan(plan: Plan): int
selectPlansByEmail(email: String): List<Plan>
deletePlan(id: int): int
com.ssafy.enjoytrip.dto.Plan
id: int
userEmail: String
naviInfo: String
getId(): int
getUserEmail(): String
getNaviInfo(): String
setId(id: int): void
setUserEmail(userEmail: String): void
setNaviInfo(naviInfo: String): void
Plan(id: int, userEmail: String, naviInfo: String)
Plan()
toString(): java.lang.String
com.ssafy.enjoytrip.model.service.PlanServiceImpl
repo: PlanRepo
insertPlan(plan: Plan): int
selectPlansByEmail(email: String): List<Plan>
deletePlan(id: int): int
PlanServiceImpl(repo: PlanRepo)

reviewRepo

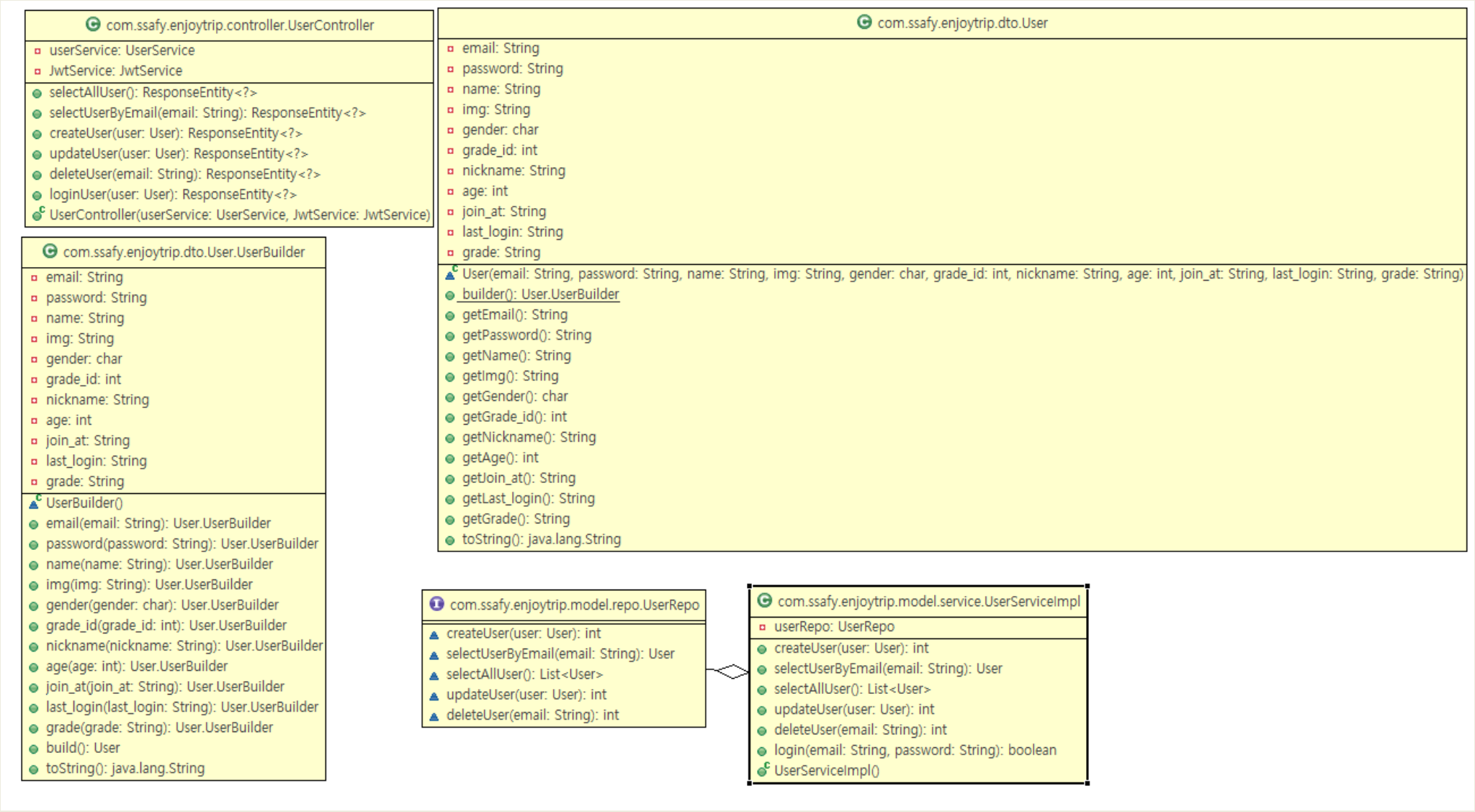

com.ssafy.enjoytrip.controller.ReviewController
reviewService: ReviewService
postReview(review: Review): ResponseEntity<?>
selectReviewsByID(id: int): ResponseEntity<?>
ReviewController(reviewService: ReviewService)
com.ssafy.enjoytrip.model.repo.ReviewRepo
selectReviewsByID(id: int): List<Review>
insertReview(review: Review): int
com.ssafy.enjoytrip.model.service.ReviewServiceImpl
reviewRepo: ReviewRepo
selectReviewsByID(id: int): List<Review>
insertReview(review: Review): int
ReviewServiceImpl(reviewRepo: ReviewRepo)
com.ssafy.enjoytrip.dto.Review
id: int
review: String
placeId: int
userEmail: String
userNickName: String
createdAt: String
getId(): int
getReview(): String
getPlaceId(): int
getUserEmail(): String
getUserNickName(): String
getCreatedAt(): String
setId(id: int): void
setReview(review: String): void
setPlaceId(placeId: int): void
setUserEmail(userEmail: String): void
setUserNickName(userNickName: String): void
setCreatedAt(createdAt: String): void
Review(id: int, review: String, placeId: int, userEmail: String, userNickName: String, createdAt: String)
Review()
toString(): java.lang.String

star


com.ssafy.enjoytrip.dto.Star
board_id: int
userEmail: String
builder(): Star.StarBuilder
getBoard_id(): int
getUserEmail(): String
setBoard_id(board_id: int): void
setUserEmail(userEmail: String): void
Star(board_id: int, userEmail: String)
Star()
toString(): java.lang.String
com.ssafy.enjoytrip.dto.Star.StarBuilder
board_id: int
userEmail: String
StarBuilder()
board_id(board_id: int): Star.StarBuilder
userEmail(userEmail: String): Star.StarBuilder
build(): Star
toString(): java.lang.String
com.ssafy.enjoytrip.controller.StarController
starService: StarService
selectAll(): ResponseEntity<?>
createStar(star: Star): ResponseEntity<?>
selectAllByEmail(email: String): ResponseEntity<?>
selectAllByBoardId(board_id: Integer): ResponseEntity<?>
deleteStar(star: Star): ResponseEntity<?>
StarController(starService: StarService)
com.ssafy.enjoytrip.model.repo.StarRepo
createStar(star: Star): int
selectUserByIdandEmail(board_id: int, user_email: String): Star
selectAll(): List<Star>
selectAllByBoardId(board_id: int): List<Star>
selectAllByEmail(email: String): List<Star>
deleteStar(star: Star): int
com.ssafy.enjoytrip.model.service.StarServiceImpl
starRepo: StarRepo
createStar(star: Star): int
selectUserByIdandEmail(board_id: int, user_email: String): Star
selectAll(): List<Star>
selectAllByEmail(email: String): List<Star>
selectAllByBoardId(board_id: int): List<Star>
deleteStar(star: Star): int
StarServiceImpl()

정호

userPNG


com.ssafy.enjoytrip.controller.UserController
userService: UserService
JwtService: JwtService
selectAllUser(): ResponseEntity<?>
selectUserByEmail(email: String): ResponseEntity<?>
createUser(user: User): ResponseEntity<?>
updateUser(user: User): ResponseEntity<?>
deleteUser(email: String): ResponseEntity<?>
loginUser(user: User): ResponseEntity<?>
UserController(userService: UserService, JwtService: JwtService)
com.ssafy.enjoytrip.dto.User
email: String
password: String
name: String
img: String
gender: char
grade_id: int
nickname: String
age: int
join_at: String
last_login: String
grade: String
User(email: String, password: String, name: String, img: String, gender: char, grade_id: int, nickname: String, age: int, join_at: String, last_login: String, grade: String)
builder(): User.UserBuilder
getEmail(): String
getPassword(): String
getName(): String
getImg(): String
getGender(): char
getGrade_id(): int
getNickname(): String
getAge(): int
getJoin_at(): String
getLast_login(): String
getGrade(): String
toString(): java.lang.String
com.ssafy.enjoytrip.dto.User.UserBuilder
email: String
password: String
name: String
img: String
gender: char
grade_id: int
nickname: String
age: int
join_at: String
last_login: String
grade: String
UserBuilder()
email(email: String): User.UserBuilder
password(password: String): User.UserBuilder
name(name: String): User.UserBuilder
img(img: String): User.UserBuilder
gender(gender: char): User.UserBuilder
grade_id(grade_id: int): User.UserBuilder
nickname(nickname: String): User.UserBuilder
age(age: int): User.UserBuilder
join_at(join_at: String): User.UserBuilder
last_login(last_login: String): User.UserBuilder
grade(grade: String): User.UserBuilder
build(): User
toString(): java.lang.String
com.ssafy.enjoytrip.model.repo.UserRepo
createUser(user: User): int
selectUserByEmail(email: String): User
selectAllUser(): List<User>
updateUser(user: User): int
deleteUser(email: String): int
com.ssafy.enjoytrip.model.service.UserServiceImpl
userRepo: UserRepo
createUser(user: User): int
selectUserByEmail(email: String): User
selectAllUser(): List<User>
updateUser(user: User): int
deleteUser(email: String): int
login(email: String, password: String): boolean
UserServiceImpl()