

TRIP HUNTER: HAPLA

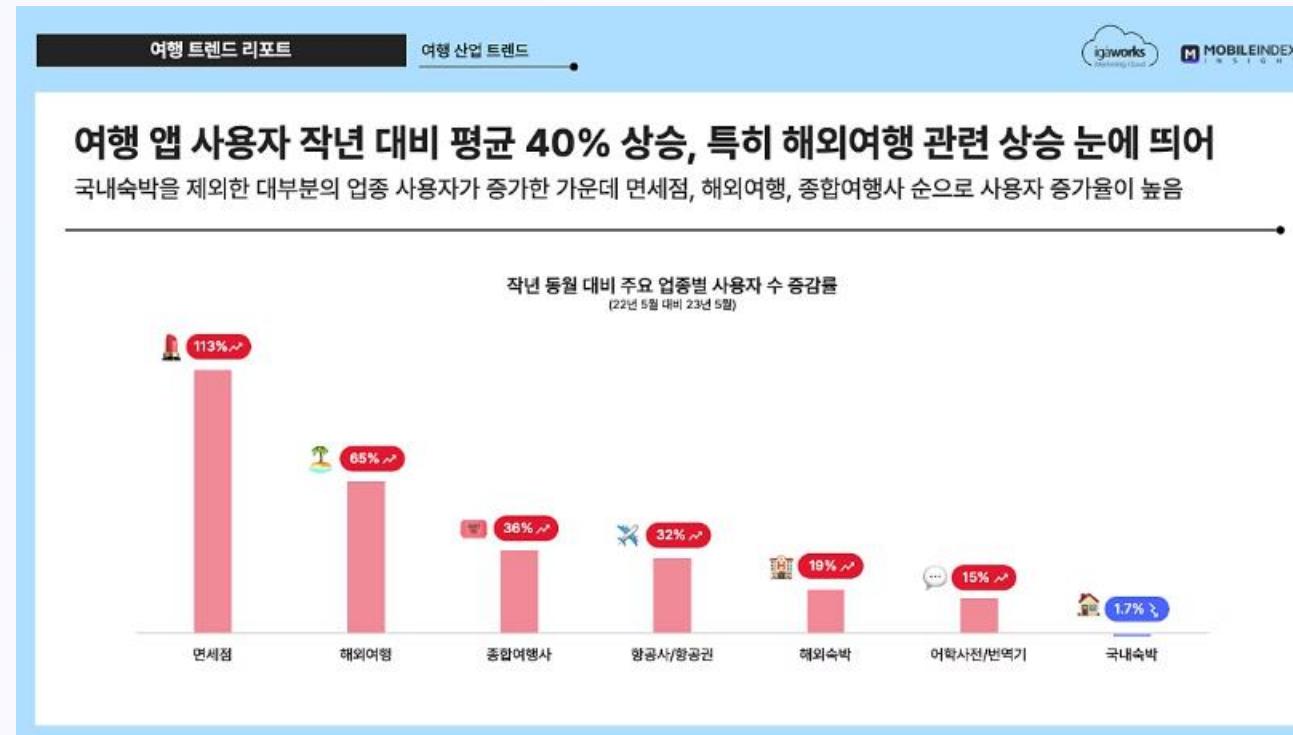
HAPLA는 HAPPY PLACE라는 의미

팀원 : 강현준, 박성훈, 이창, 조민주, 황성현

목차

1. 프로젝트 개요
2. 시장 분석
3. 벤치마킹 사이트
4. 기획 회의
5. UI 회의
6. DB 회의
7. 코드리뷰
8. 작품시연
9. Q&A

프로젝트 개요



개발 배경

온라인 여행 시장 급성장과 사용자 중심 정보 수요 증가에 대응

차별화 전략

소셜 & 커뮤니티 기능 같은 일정에 여행하는 사람들과
소통할 수 있는 여행 메이트 매칭 여행 후기 공유

시장 분석



여행 산업 트렌드

개인화된 여행 경험 선호, 지속 가능한 여행 관심 증가, 디지털 노마드 증가

온라인 플랫폼 현황

모바일 예약 비중 확대, 메타서치 엔진 성장

벤치마킹 사이트

TRIPADVISOR & TRIPLE

주요 기능

- 호텔 및 관광지 리뷰
- 가격 비교
- 예약 서비스
- 맞춤형 계획 도구

사용자 경험

- 직관적 검색
- 사용자 생성 콘텐츠
- 개인특화
- 반응형 디자인



요구사항 정의서

팀명	TRIP HUNTER		
프로젝트 명	HAPLA		
구분	요구사항 ID	요구사항 명	요구사항 상세 설명
공통	common-01	탑바 좌측 메인버너	클릭 시 메인페이지로 이동
	common-02	Recommend 버튼	클릭 시 여행지 추천 품 이동
	common-03	Review 버튼	클릭 시 리뷰 작성 품으로 이동
	common-04	Trip 버튼	클릭 시 여행 품으로 이동
	common-05	Login 버튼	클릭 시 모달로 로그인 화면 출력
	common-06	로그인 모달창-1	해당 사이트에서 사용할 이름 입력
	common-07	로그인 모달창-2	소셜로그인(google,kakao)
	common-08	로그인 됐을시 프로필버튼	프로필버튼 클릭 시 드롭다운 방식으로 여행, 리뷰작성, 프로필, 문의하기, 로그아웃
	common-09	드롭다운 프로필버튼	클릭 시 내 정보 조회 페이지로 이동 하단에 수정 버튼
	common-10	내 정보 수정	소셜 로그인 할 때 웹사이트에서 사용할 닉네임 수정 및 프로필 사진 변경
메인	main-01	카테고리별 검색바	전체 검색, 호텔, 관광명소, 음식점, 항공권 검색
	main-02	카테고리별 검색바 배경	해당 카테고리에 어울리는 사진이 배경에 들어가 있고 그 위에 검색바가 있음
	main-03	검색바 하단에 인기 여행지	대륙별 BEST 여행지 5개씩 or 해쉬태그를 통한 추천 여행지
	main-04	전체 검색 기능	지역명 및 관광명소, 숙박시설 등 검색
	main-05	전체 검색 결과 - 지역명	검색한 지역(나라)의 사진, 설명 및 해당 지역 관광명소, 맛집, 호텔 추천
	main-06	전체 검색 결과 - 관광명소	검색한 관광명소 설명 및 위치 해당 명소 주변 맛집, 숙박시설 등 정보 추천
	main-07	전체 검색 결과 - 호텔	검색한 호텔의 위치 및 시설 정보 및 예약 가능 정보(결제???, 인근 다른 평점이 높은 숙박시설 추천)
	main-08	전체 검색 결과 - 음식점	검색한 키워드가 들어가는 음식점 리스트 출력(디폴트-> 정확도 순)
	main-09	카테고리 별 검색 결과	전체 검색의 결과에서 각 해당하는 카테고리의 결과 출력
	main-10	항공권 검색 기능	왕복, 편도, 다구간 + 출발, 도착 장소 및 날짜를 선택하여 항공권 검색
	main-11	항공권 검색 결과	기본적으로는 가성비 순으로 정렬이 되고, 가성비, 비행 시간, 가격을 기준으로 선택정렬 가능
	main-12	항공권 검색 결과 메인	검색 결과 항공권들을 띄워줌
	main-13	항공권 검색 결과 좌측	직항, 경유, 항공사, 가격으로 필터링 가능
	main-14	추천 여행지 및 콘텐츠	전세계에서 1~12월까지 각 날짜(월기준)의 TOP5 여행지 추천
	main-15	여행지 추천 detail	각 여행지의 사진 + 해당 여행지의 필요한 정보들 텍스트

관리자	admin-01	사용자 목록 조회	이름, 이메일, 가입일 등으로 정렬
	admin-02	사용자 활동 내역	리뷰 작성 내역, 예약 내역 등
	admin-03	결제, 거래 내역-	결제 내역 조회 및 거래 내역 관리
	admin-04	환불-	환불 요청 처리 및 승인
	admin-05	예약 현황	호텔, 투어, 액티비티 등 예약 현황 조회
	admin-06	예약 상태	예약 상태 변경 및 취소 처리
	admin-07	방문자 수 통계	일별, 주별, 월별 방문자 수 통계
	admin-08	예약 통계 분석	상품별, 지역별, 테마별 통계 분석
	admin-09	매출 분석	매출 현황 및 예상 매출 분석
	admin-10	인기	인기 여행지, 호텔, 투어 분석
	admin-11	고객 문의	고객 문의 및 피드백 관리
	admin-12	리뷰, 평점	리뷰 및 평점 관리(부적절한 리뷰 필터링, 삭제)
	admin-13	신고 처리	부적절한 컨텐츠 신고 처리(사용자 간 커뮤니티)
일정	SCH-01	일정 등록	1. 일정 제목, 날짜, 여행장소 입력 필드 2. 사진 및 메모 추가 가능 3. 일정 저장 후 목록에서 확인 가능
	SCH-02	일정 등록 - 일정 제목	일정 제목 입력
	SCH-03	일정 등록 - 일정 날짜	일정 날짜 캘린더 띄우고 입력
	SCH-04	일정 등록 - 여행 장소	여행 장소 입력
	SCH-05	일정 조회	1. 일정 목록을 최신순으로 정렬 (10개의 게시글 행 노출) 2. 일정 제목 클릭시 상세보기 페이지로 이동 3. 검색 기능 제공
	SCH-06	일정 조회 - 일정 제목	일정 제목 클릭시 상세보기 페이지로 이동
	SCH-07	일정 조회 - 검색 기능	검색한 키워드가 들어가는 리스트 출력
	SCH-08	일정 상세보기	1. 일정 제목, 날짜, 여행 장소, 사진 및 메모 확인 2. 일정 수정 및 삭제 기능 제공 3. 공유하기 기능 추가 (카카오톡, 이메일 등)
	SCH-09	일정 상세보기 - 수정	일정 수정하는 폼으로 이동
	SCH-10	일정 상세보기 - 삭제	일정 삭제
	SCH-11	일정 수정	1. 일정 제목, 날짜, 여행 장소 변경 가능 2. 수정 후 일정 목록에서 자동 반영
	SCH-12	일정 삭제	1. 삭제 시 확인 메시지 출력 후 삭제 2. 삭제된 일정은 복구 불가
	SCH-13	일정 공유	특정 사용자에게 공유 가능(링크...?)
	SCH-14	지도 연동	1. 지도 API 연동 (Google Maps) 2. 여행 일정에 등록한 장소 자동 표시
	SCH-15	알림 기능	1. 일정 하루 전, 당일 알림 기능 (푸시 알림, 이메일) 2. 사용자가 알림 설정 가능 (시간, 방식)

날씨	weather-01	실시간 날씨 조회	OpenWeatherMap API를 이용해 검색창에 사용자가 입력한 도시의 현재 날씨 정보를 제공
	weather-02	기온/습도/풍속 제공	검색한 도시의 현재 기온, 습도, 풍속 및 풍향 정보를 표시
	weather-03	단기 날씨 예보	검색한 도시의 3일간의 예상 날씨 데이터 제공
환율	ex-01	환율 조회	Fixer.io API를 이용한 환율 조회
	ex-02	환율에 따른 금액 변환	특정 금액 입력 시 변환된 통화 금액 표시
커뮤니티	community-01	새 글 작성	글 작성 시 제목, 본문, 카테고리, 사진 업로드 가능
	community-02	글 수정	글 수정하는 폼으로 이동(글 수정 폼은 글 작성 폼과 동일)
	community-03	글 삭제	글 삭제(삭제된 게시글 복구 불가)
	community-04	게시글 상세보기	1. 게시글마다 좋아요 기능 추가 2. 댓글 및 대댓글 기능 제공 3. 특정 댓글에 좋아요 기능 제공
	community-05	댓글 수정	댓글 수정버튼 클릭시 해당 댓글 수정 가능
	community-06	댓글 삭제	댓글 삭제(삭제된 댓글 복구 불가)
	community-07	게시판	1. 검색 입력칸 오른쪽에 위치 2. 페이지마다 10개의 게시글 행 노출 3. 관리자가 작성한 게시물은 공지게시물로서 상단 고정 4. 일반 작성자 게시물은 10개의 게시글 중 관리자 게시글을 제외한 나머지 아래에 위치 5. 게시물의 고유 번호, 작성자, 작성일, 조회수, 좋아요 수 노출
	community-08	게시글 조회 및 검색	검색 시 최신순, 인기순으로 정렬 기능 제공
리뷰	review-01	새 리뷰 작성	글 작성 시 제목, 본문, 카테고리, 사진 업로드 가능
	review-02	리뷰 카테고리	세계지도 or 나라 별로 카테고리 선택
	review-03	카테고리 속 페이지	페이지마다 10개의 게시글 행 노출
	review-04	리뷰 상세보기	1. 게시글마다 좋아요 기능 추가 2. 별점 및 리뷰 내용 출력
	review-05	리뷰 수정	수정 버튼 클릭 시 수정 폼으로 이동(수정 폼은 작성 폼과 동일)
	review-06	리뷰 삭제	리뷰 삭제(삭제된 리뷰 복구 불가)

스토리보드

인덱스

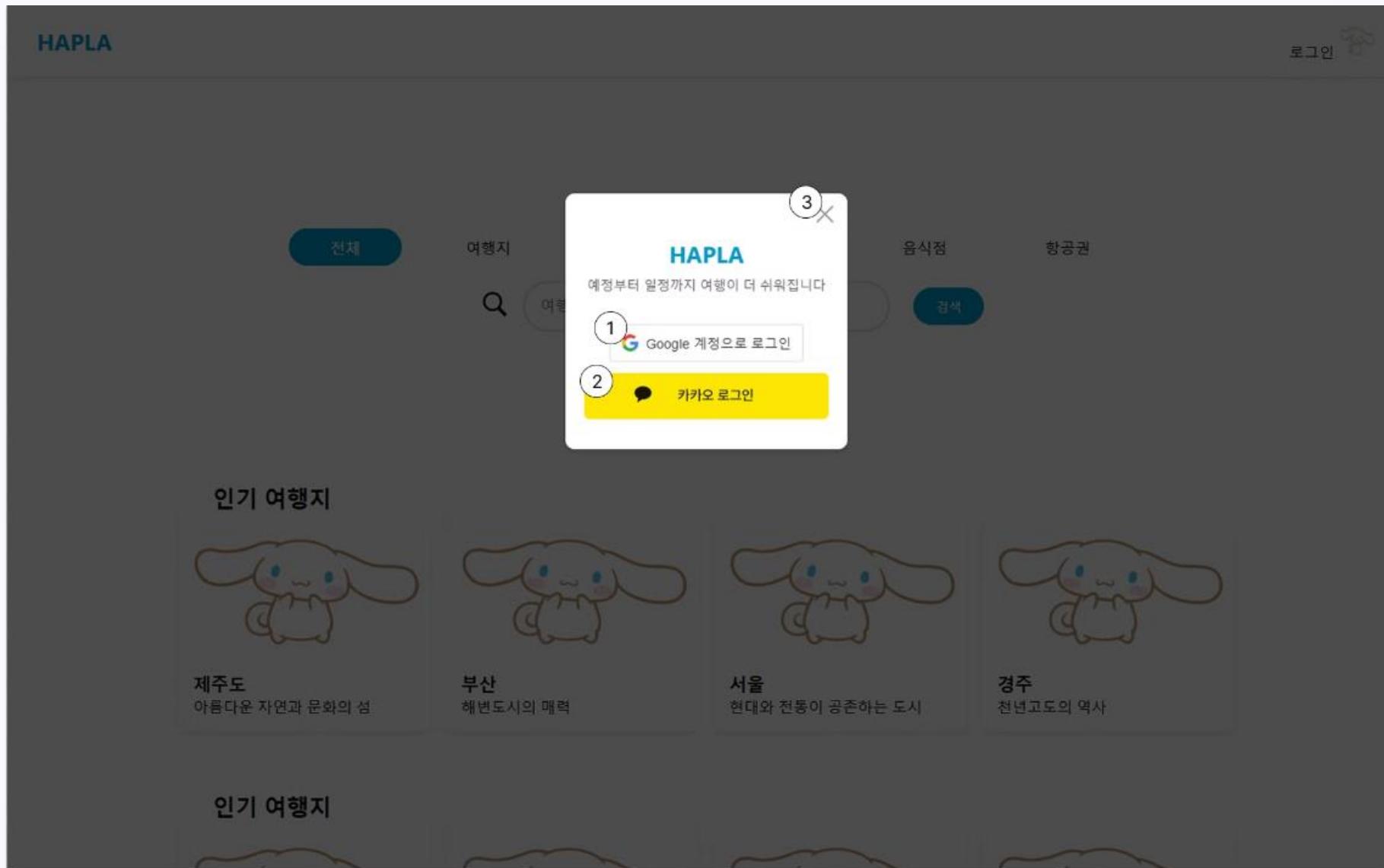
HAPLA

계획부터 예약까지, 여행이 쉬워지는

나를 아는 여행 앱
① HAPLA와 떠나기

메인 페이지

로그인 모달



1. 카카오 로그인 창을 띄운다.
2. 구글 로그인 창을 띄운다.
3. 모달창을 종료한다.

회원가입 모달창



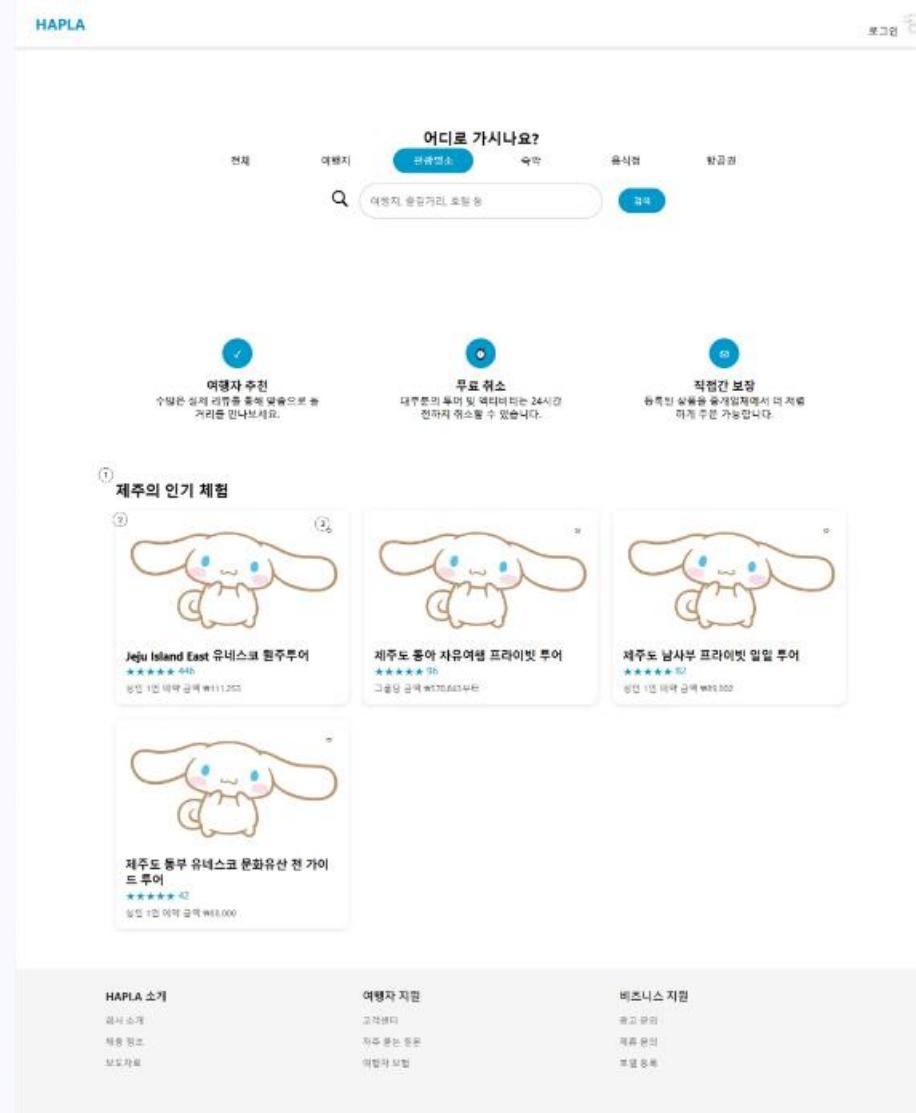
1. 모달창을 종료한다.

마이페이지



1. 모달창을 종료한다.
2. 로그인한 회원의 사진을 가져온다.
3. 프로필 사진을 변경한다.
4. 로그인한 회원의 닉네임을 가져온다.
5. 회원 수정
6. 회원 탈퇴

관광명소 검색



1. 검색한 관광명소의 인기 체험 목록을 불러온다.
2. 카드를 클릭하면 관광명소 상세정보 화면으로 이동한다.
3. 로그인한 회원이면 좋아요를 누를수 있다.

관광명소 상세 보기

① 우도

◀ 을끌거리 모두 보기 ⌂ ⌂ ②

★★★★★ 1.082건의 리뷰 · 개수의 을끌거리 189개 중의 1위 · 신

리뷰 쓰기

포정하지 않음

사람들이 이야기하는 내용

· 세우면 부도가 있다 ·
· 해변에 물이 차운다 ·
제주관광지 중 디저트 봄으로 풀 해다른 듯한 우도를
방문해보는게 아닐까? 해수 쓰러우면서 일출 및 일몰
기록 즐기는 그곳 우도이다

· 쓰레기가 너무 많아요 ·
· 2022년 8월
바위 절벽과 바위는 물결이나, 산 경치가 거울인 드
내기산 노른 바닷가 물결이 문화 "세계에 있는 남해
진을, 보기만 해도 아름다운 거울인 물결 바다" 등

✓ 수정을 제안하여 표시 사용 개선하기
이 리스팅 정보 개선하기

1,701



현재 여행 중이신가요?

회원님이 이용할 수 있는 체험을 찾도록 도와주세요.

예 아니요

지역 설명

직접 연락하기
전화 번호

네스트 주변
음식점
모두 보기
5km 내 30개

습기탱크 ★★★★★ 5 0.4 km · 6 · 미처포 · 아프리카 유키 · 앤서니 유키

미지방탕 ★★★★★ 7 1.3 km · 미처포 · 아프리카 유키 · 앤서니 유키

풀교포 ★★★★★ 1 1.5 km · 미처포 · 경단감 유키 ·

경제 보기



③ 다른 여행자들에게 인기가 좋습니다

모든 보기

성산 일출봉 ★★★★★ 2,088 산 · 자연

한라산 국립공원 ★★★★★ 1,198 산 · 국립공원

대포해안주상절리대 ★★★★★ 685 지질 · 경관이 좋은 산책로

협재 해수욕장 ★★★★★ 479 해변



음식점 검색

The screenshot shows the HAPLA website homepage. At the top, there is a navigation bar with links for '전체' (All), '여행지' (Travel Destinations), '관광명소' (Tourist Attractions), '숙박' (Accommodation), '음식점' (Restaurant) (which is highlighted in blue), and '항공권' (Airline Tickets). There is also a '로그인' (Login) button with a user icon. Below the navigation, a search bar contains the placeholder text '여행지, 풀길거리, 호텔 등' (Travel destination, walking street, hotel, etc.) with a magnifying glass icon. A '검색' (Search) button is located to the right of the search bar.

In the center, there are three call-to-action boxes:

- 최고의 음식점을 찾아보세요**: '검색을 통해서 나의 일상에 맞는 음식점을 찾아보세요.'
- 최신 리뷰 보기**: '전 세계 여행자들이 남긴 수많은 리뷰를 확인해 보세요.'
- 테이블 예약하기**: '원하는 음식점을 온라인으로 예약하세요.'

Below these boxes, there is a section titled '① 제주의 최고 인기 음식점' (Top 1: Popular Restaurants in Jeju). It displays four cards, each featuring a cartoon character and a restaurant name with its rating and review count. The cards are:

- 제주도 톡톡구이집 (★★★★★ 244개 리뷰)
- 고등어조림 (★★★★☆ 188개 리뷰)
- 바다뷰카페 (★★★★★ 156개 리뷰)
- 제주전통밥집 (★★★★★ 122개 리뷰)

At the bottom of the page, there are three columns of links:

HAPLA 소개	여행자 지원	비즈니스 지원
회사 소개	고객센터	광고 문의
채용 정보	자주 묻는 질문	제휴 문의
보도자료	여행자 보험	포털 등록

1. 검색한 음식점의 목록을 불러온다.
2. 카드를 클릭하면 음식점 상세정보 화면으로 이동한다.
3. 로그인한 회원이면 즐겨찾기(좋아요) 등록이 가능하다.

숙박 검색 결과

The screenshot shows the HAPLA search interface for Seoul hotels. The search parameters are set to '여행자' (Guest), '2월 10일 (화)' (February 10th, Tue), and '2월 25일 (화)' (February 25th, Tue). The results are listed under '서울의 시설 3,434곳' (3,434 facilities in Seoul). The results are categorized by price and rating:

- 1. 호텔 피체이 명동**:
 - Agoda: ₩274,800
 - Hotels.com: ₩189,602
 - Booking.com: ₩189,602
- 2. 메이필드 호텔 앤 리조트**:
 - Hotels.com: ₩194,857
 - Agoda.com: ₩164,462
 - Booking.com: ₩164,391
- 3. 신라스테이 구도**:
 - Hotels.com: ₩130,350
 - Agoda.com: ₩117,884
 - Booking.com: ₩117,884
- 4. 그랜드 하얏트 서울**:
 - Hotels.com: ₩384,218
 - Agoda.com: ₩384,210
 - Booking.com: ₩384,214

Each result card includes a small image of the hotel, its name, price, and a '더보기' (View more) button.

- 검색한 숙박의 목록을 불러온다.
- 각 카테고리에 맞는 숙박의 전체 목록을 불러온다.

숙박 검색 상세보기



여행 날짜의 가격 보기

체크인 2월 18일 (화)	체크아웃 2월 25일 (화)	인원수 객실 1개, 성인 2명, 아동 0명
agoda ●●●●●	✓ 2월 15일 (토) 일전 취소 시 전액 환불	₩274,898 ₩189,602 특가보기
Via.com	✓ 2월 15일 (토) 일전 취소 시 전액 환불	₩274,898 특가보기

가격은 트립어드바이저 파트너가 안내하는 것으로 파트너가 파악하고 있는 모든 세금과 수수료를 포함한 1박 객실 요금을 반영합니다. 자세한 정보는 파트너사에 문의하세요.

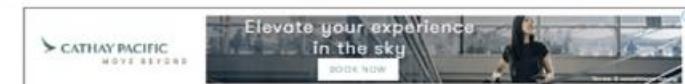
이 호텔을 이미 예약하셨나요?

계획하는데 도움을 드릴 수 있도록 확인해 주세요.

예 아니요

다음 30일간 가격 동향

현재 가격이 평균적인 편입니다.



소개

4.5

리뷰

항공권 검색

The screenshot shows the HAPIA flight search interface. At the top, there are two sets of input fields for departure and arrival cities, and a date selector. Below these are two dropdown menus: one for '여행 노선' (Travel Route) and another for '항공권 추가하기' (Add Flight Ticket). The main content area displays a list of flight search results. Each result includes the departure and arrival times, the airline, price, and a '선택하기' (Select) button. On the left side of the results, there are several filter sliders: '정원' (Capacity), '가는 날' (Departure Date), '가는 날' (Arrival Date), '출발 시간 (PDT)' (Departure Time PDT), '도착 시간 (PDT)' (Arrival Time PDT), '항공사' (Airline), and '가격' (Price). At the bottom of the results list, there are navigation arrows for page 1, 2, and 3. The footer contains links for 'HAPIA 소개' (About HAPIA), '여행자 지원' (Traveler Support), and 'ierzis' (ierzis) services.

1. 왕복, 편도, 다구간 버튼을 통해 검색 필터링 변경 가능
2. 좌측사이드바의 필터링들을 통해 사용자가 원하는 항공권 검색 가능
3. 선택하기 버튼 클릭을 통하여 대행사 별 항공권 가격 확인 가능
4. 다구간 버튼을 클릭시에는 편도, 왕복과 다르게 항공권 추가하기를 클릭하여 구간을 더 추가해서 검색 가능
- 5.

지역 검색 결과

HAPLA

로그인 User Profile

어디로 가시나요?

여행지 관광명소 숙박 음식점 항공권

검색어: 런던, SOHO

LONDON

런던

런던은 예술, 역사, 문화 등 분야를 가리지 않고 모든 것의 중심에 있는 거대한 도시입니다. 하지만 다른 주요 중심지와 차별화되는 점은 과학마다 고유한 문화가 있다는 것입니다. 런던에서 가족과 함께 여행을 보내면서 박물관, 미술관, 그 외에도 각종 이벤트를 즐기거나 편안한 휴식처가 필요합니다. 또는 세련된 소디카페에서 맛진 상점과 가격의 베티나는 활동에 인상적인 구경을 하거나 다른 소포트에서 책과 친 친구나 출판에서 새해까지 판권을 찾을 수 있습니다. 음식도 배울 수 있습니다. 파이프 풍선과 같은 아트와 문화, 그리고 차의 맛은 방문하면서 저마다 박물에 작품의 다른 캐릭터처럼 각각의 수준은 품질과 향에서 선택의 여지가 있고, 소비로운 막바지에 흥미로운 세트가 운영하는 음식점이 점점 더 늘고 있습니다. 항상 새로운 일들이 일어나고 있으며 아래에 더 많은 추천 정보가 있습니다.

항공 정보: 1 GBP + 1,500 KRW

현재 날씨: 서울: 맑음, 5°C / 경기: 맑음, 12°C

즐길거리

빅笨
런던 앤더
베이징 공연
타워 브리지

숙박시설

런던 배스 투어
2층 버스로 런던 관광
월드컵 콤팩트
런던 워터 투어

- 지역 이름을 검색 할 시 해당 지역의 간단한 정보들과 사진을 확인할 수 있다.
- 아래의 해당 지역의 BEST 즐길거리, 음식점, 숙박정보를 확인할 수 있다.
- 해당 지역에 관한 리뷰를 확인할 수 있다.
-

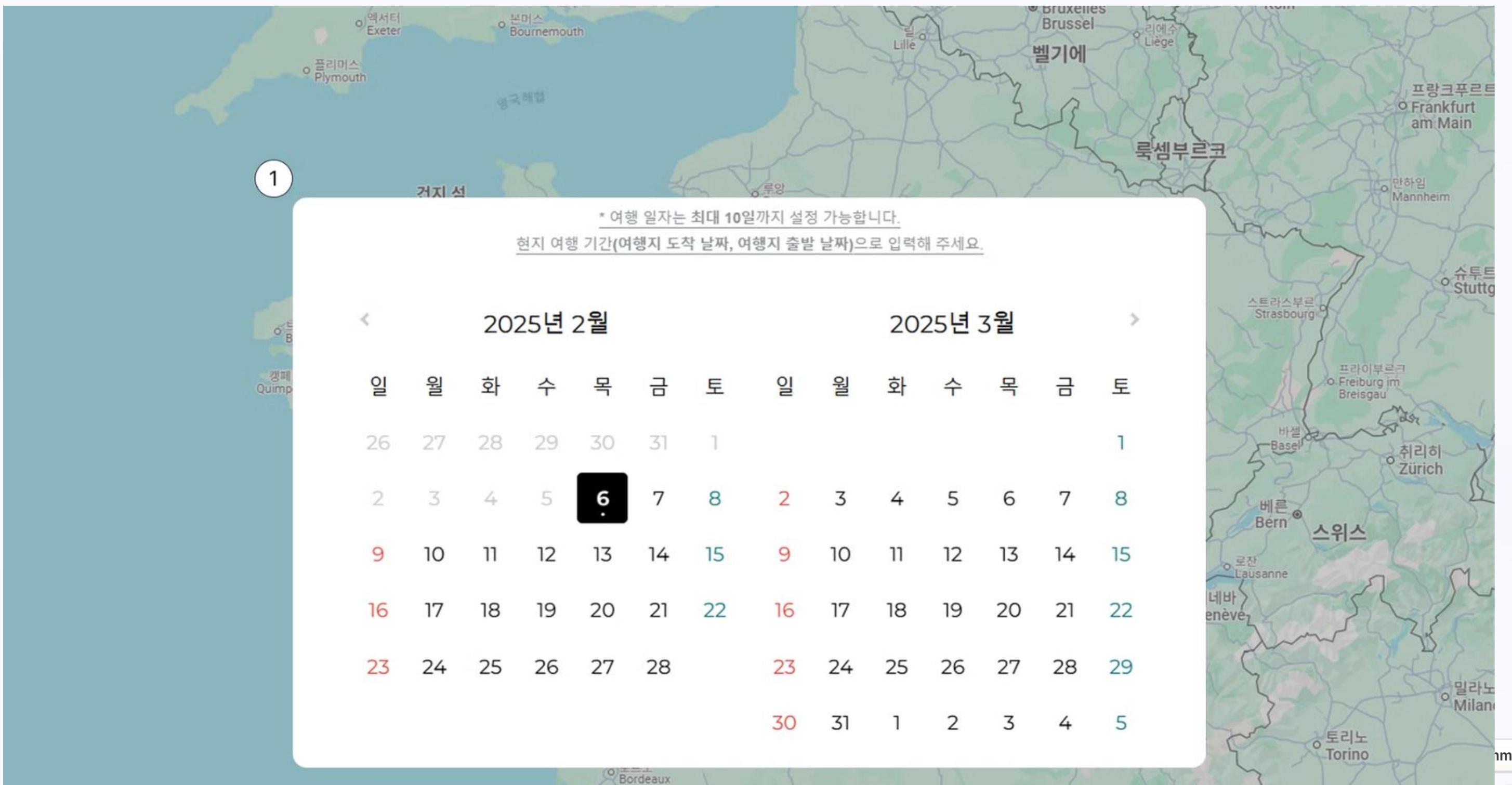
여행 일정



여행일정 모달

1. 여행일정 : 버튼 클릭시 모달화면 출력
2. 여행일정 - 여행 장소 검색 : 여행 장소 입력 검색칸
3. 여행일정 : 검색하면 장소 출력되고 장소 클릭시 여행 일정 페이지로 이동

여행일정 : 날짜 지정



여행 일정 만들기

1 여행 제목

2

3

2월 6일

+추가

2월 7일

4

5 6 7 8 ×

2월 8일

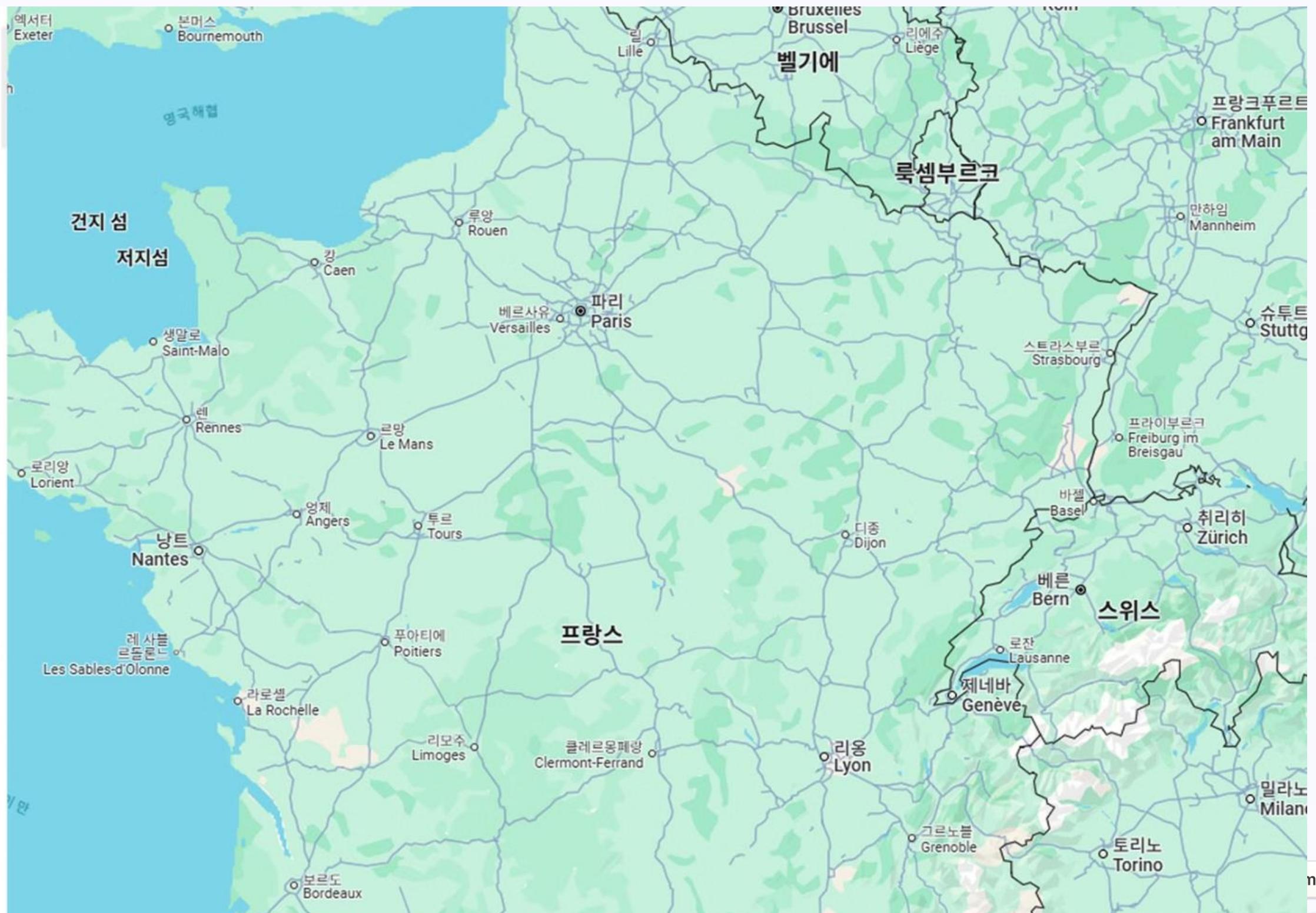
+추가

2월 9일

+추가

2월 10일

+추가



여행일정 목록

나의 일정

2



여행 제목

2025.02.07 - 2025.02.15
파리, 프랑스



여행일정 만들기: 장소 추가

☰ 여행 제목

2월 6일

+추가

2월 7일

1

루브르 박물관

명소 75001 Paris, France
12514 ★ 4.7

+

2월 8일

+추가

2월 9일

+추가

2월 10일

+추가

2

장소명을 입력하세요.

3

에펠탑

명소 Champ de Mars, 5 Av. Anatole Fra...
13809 ★ 4.6

+

루브르 박물관

명소 75001 Paris, France
12514 ★ 4.7

+

샹젤리제 거리

명소 Av. des Champs-Élysées, 75008 Pa...
11020 ★

+

에투알 개선문

명소 Pl. Charles de Gaulle, 75008 Paris, ...
10590 ★ 4.7

+

베르사유 궁전

명소 Place d'Armes, 78000 Versailles, Fr...
9889 ★ 4.6

+



여행일정 : 메모 추가

☰

여행 제목

2월 6일

+추가

1

메모 내용

2



2월 7일



메모 내용

2월 8일

+추가

메모 내용

2월 9일

+추가

메모 내용

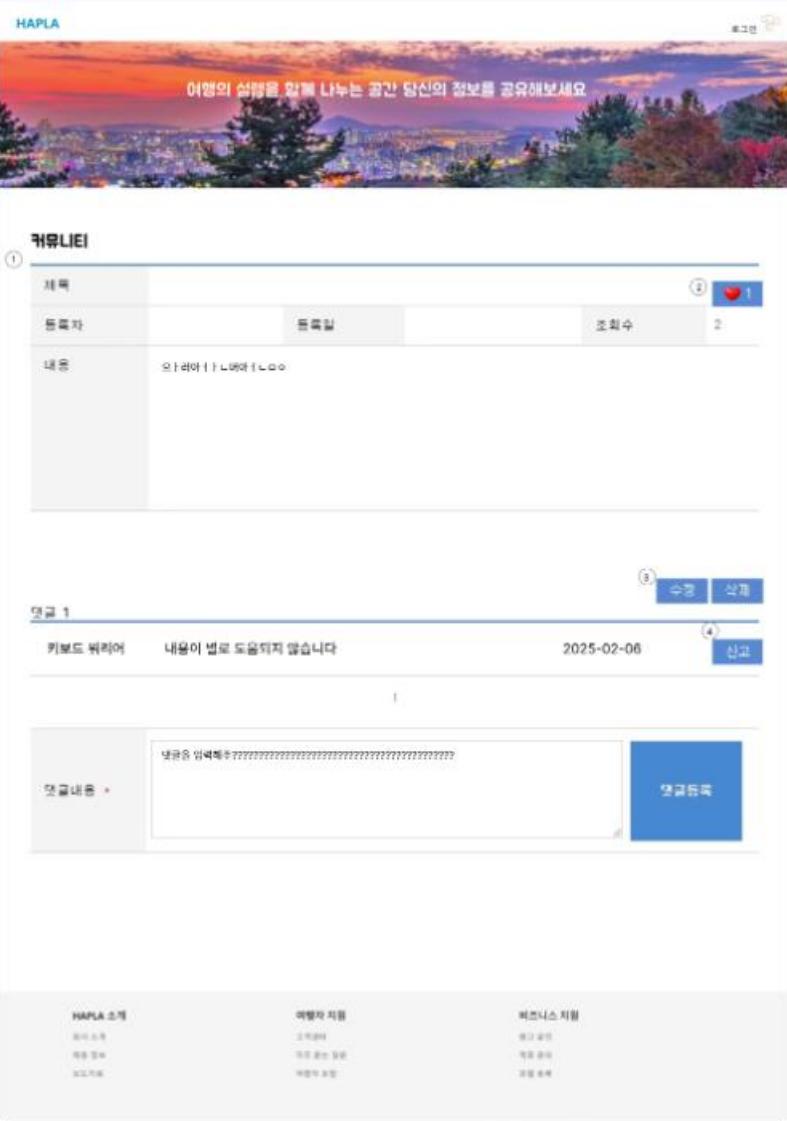
2월 10일



The screenshot shows a search results page from the HAPLA website. At the top, there is a banner with the text "여행의 선례를 함께 나누는 공간 당신의 정보를 공유해보세요." Below the banner, a search bar contains the text "청소 또는 액티비티로 검색". The main content area is titled "커뮤니티" and displays a table of 10 search results. The table has columns for 번호 (번호), 좋아요 (좋아요), 작성자 (작성자), and 등록일 (등록일). The results are numbered 1 through 10. Each result row includes a small thumbnail image, the number, the number of likes, the author's name, and the registration date. Below the table is a navigation bar with numbers 1 through 10, where the first number is highlighted in blue. At the bottom of the page, there are three sections: "HAPLA 소개" (HAPLA Introduction) with links to "회사 소개", "제품 설명", and "부드러운"; "미행자 지원" (Support for Victims) with links to "고객센터", "자주 묻는 질문", and "이용자 부정"; and "비즈니스 지원" (Business Support) with links to "광고 문의", "제휴 문의", and "사업 등록".

번호	좋아요	작성자	등록일
1	5	테스트1	2016-09-20
2	5	테스트1	2016-09-20
3	10	테스트1	2016-09-12
4	7	테스트1	2016-08-01
5	0	테스트1	2016-07-27
6	11	테스트1	2016-07-27
7	2	테스트1	2016-07-07
8	3	테스트1	2016-07-07
9	4	테스트1	2016-07-07
10	0	테스트1	2016-07-07

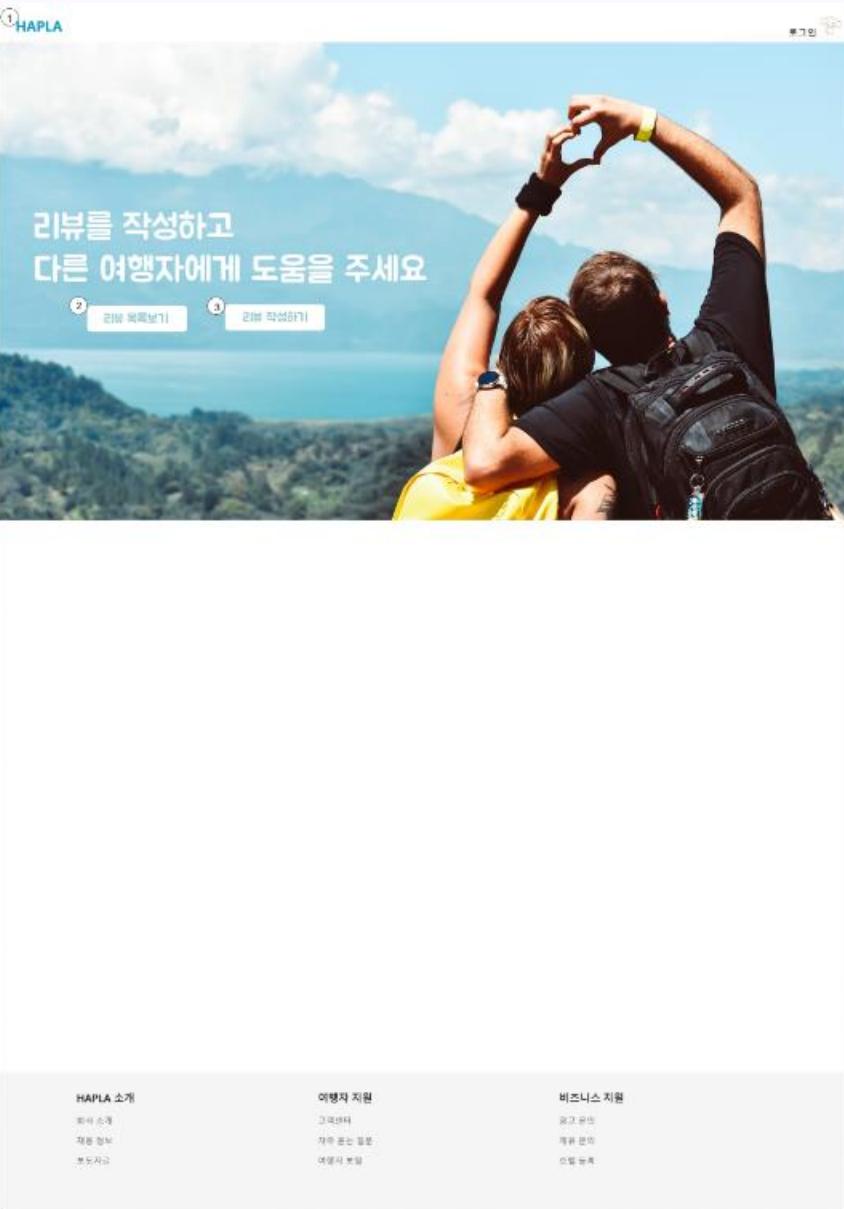
1. 화면 어디든 클릭 시 메인 페이지로 이동
2. 검색어 입력 후 엔터 누를 시 해당 키워드 출력
3. 커뮤니티 게시판으로서 번호, 내용, 좋아요, 작성자, 등록일 표시
4. 등록 버튼으로서 누를 시 게시판 작성 페이지로 이동
5. 페이지 번호로서 누를 시 해당 페이지로 이동



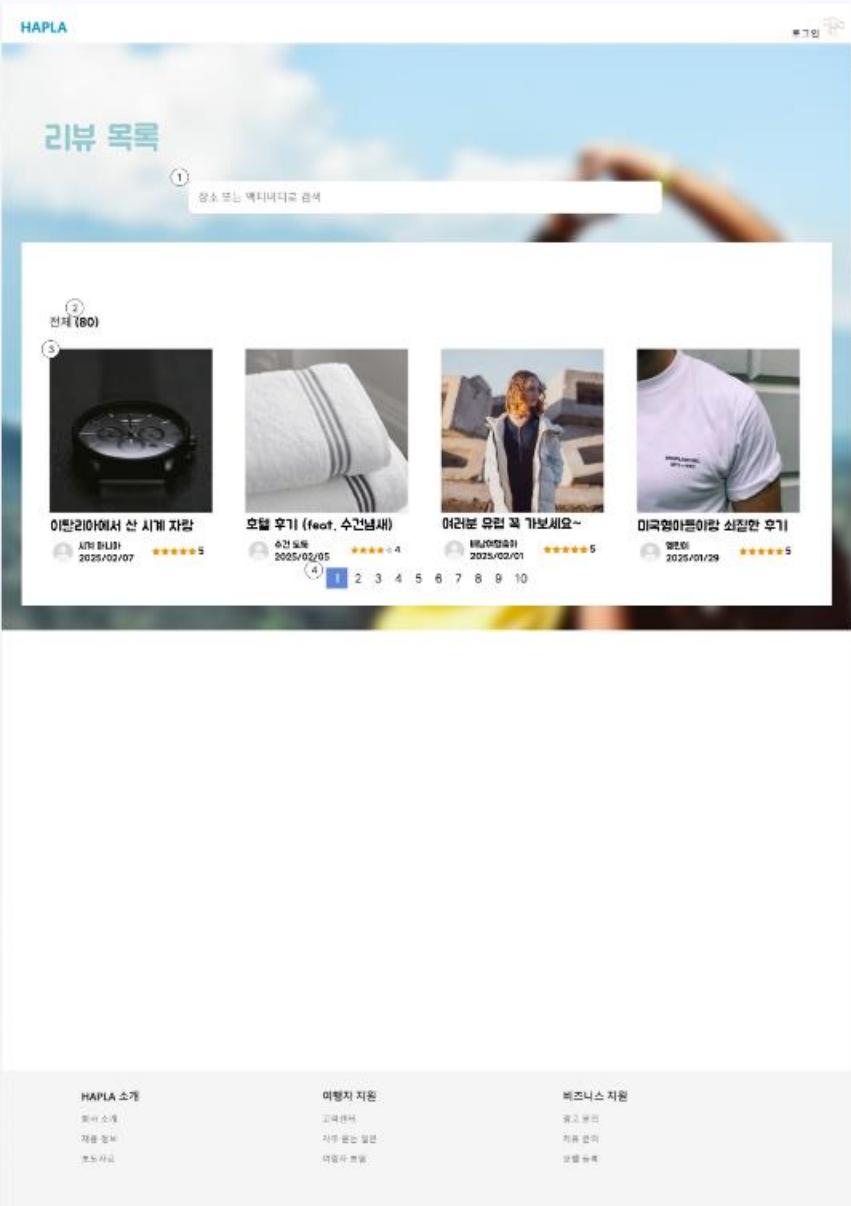
1. 커뮤니티 게시글의 제목, 작성자, 작성일, 조회수, 내용 확인 가능
2. 해당 게시물의 좋아요 기능으로서 버튼 클릭 시 숫자 상승
3. 게시글, 수정 및 삭제 버튼으로서 작성자에게만 보이는 기능
4. 댓글 신고버튼으로서 누르면 '해당 댓글을 삭제하시겠습니까?' alert 창 생성
5. 댓글 입력 후 댓글등록 버튼 누를 시 댓글 추가



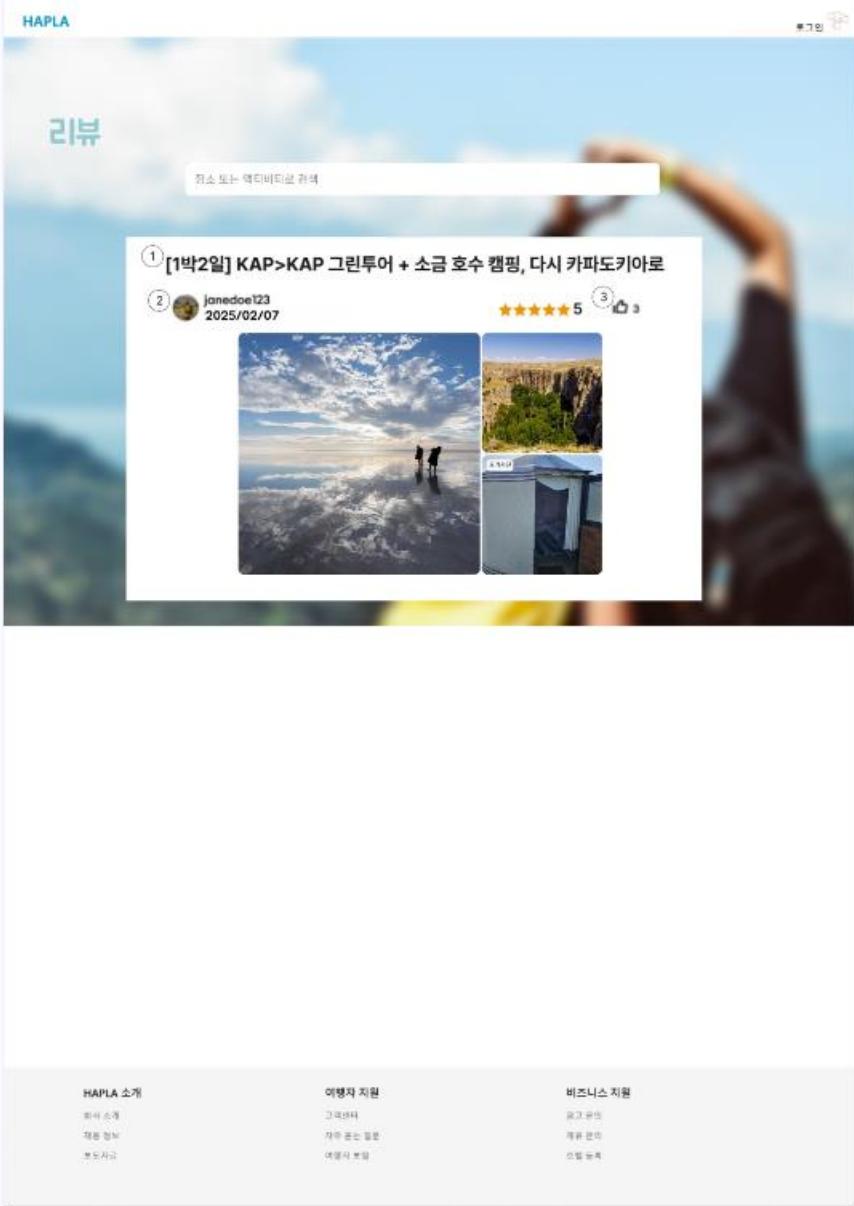
1. 제목 및 내용 작성, 글꼴 및 정렬 선택
2. 취소 버튼을 누를 경우 저장되지 않은 상태로 이전 페이지,
3. 저장 버튼을 누를 경우 게시글 등록
3. 첨부파일 추가 가능



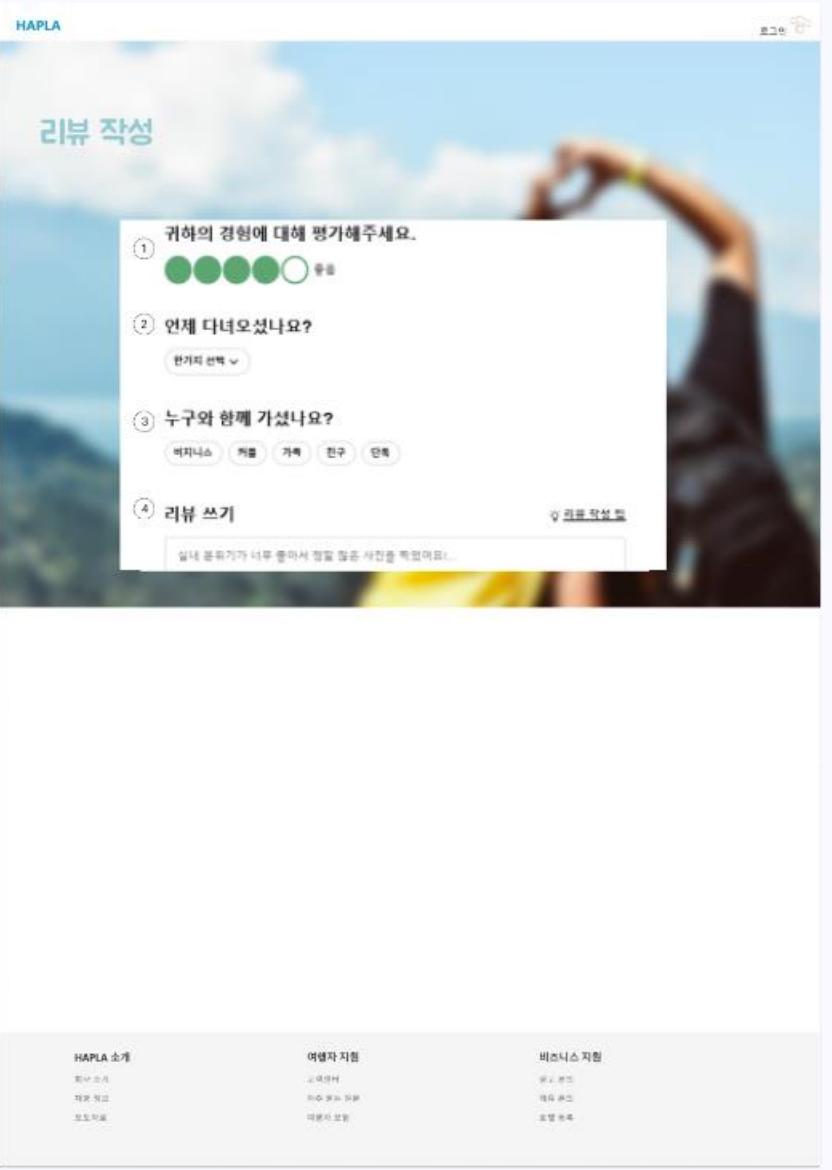
1. 메인페이지로 이동하는 버튼
2. 리뷰 목록 페이지로 이동하는 버튼
3. 리뷰 작성 페이지로 이동하는 버튼



1. 특정 키워드로 검색할 수 있는 기능
2. 전체 게시글의 수 나타내는 기능
3. 리뷰 게시글의 썸네일, 제목, 작성자, 작성일, 별점 나타남
4. 전체 쪽 수와 현재 쪽 수 나타내는 기능



1. 리뷰 게시글의 제목 상단에 위치
2. 게시글 작성자의 이름, 작성일 그리고 별점이 나타남
3. 게시물의 좋아요 기능으로 누르면 빨간색으로 변하며 숫자가 1씩 증가



1. 별점 선택 기능
2. 여행 날짜 선택 기능
3. 동행인 선택 기능
4. 리뷰 내용 작성 기능

관리자 페이지

The screenshot shows the '여행 웹사이트 관리자 페이지' (Travel Website Admin Page). The top navigation bar includes links for MAIN, 사용자 조회, 통계 조회, 예약 현황, 매출 분석, 글자사항, and 신고 관리.

사용자 조회

닉네임	이메일	가입일	최근 활동
여행자1	hong@example.com	2023-01-15	2023-06-20
세계탐험가	kim@example.com	2023-02-20	2023-06-18
맛집탐방	lee@example.com	2023-03-10	2023-06-21
호텔리뷰어	park@example.com	2023-04-05	2023-06-19
프로그램퍼	jung@example.com	2023-05-01	2023-06-22

사용자 활동 조회

총 사용자	총 게시글	총 리뷰	총 댓글 수
5명	25기	37기	50개

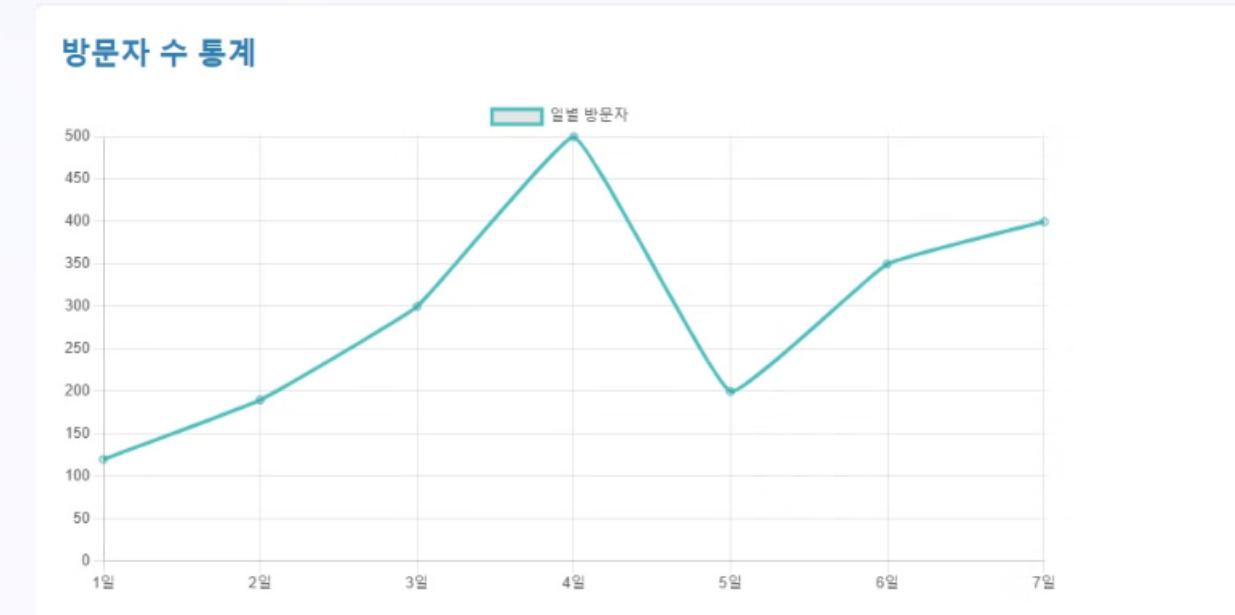
닉네임	게시글	시작 리뷰
여행자1	<ul style="list-style-type: none">서울 여행 끌립제주도 맛집 추천	<ul style="list-style-type: none">서울 인조 리뷰 (★★★★★)제주도 해변 리뷰 (★★★★★)
세계탐험가	<ul style="list-style-type: none">유럽 비ز니스 여행 준비동남아 여행 경비 힙	<ul style="list-style-type: none">파리 어却被 주변 리뷰 (★★★★☆)방콕 카오산로드 리뷰 (★★★★★)
맛집탐방	<ul style="list-style-type: none">부산 해운대 맛집 리스트강원도 속초 맥방 여행	<ul style="list-style-type: none">부산 강안리 해변 리뷰 (★★★★★)속초 중앙시장 리뷰 (★★★★☆)

사용자 조회 페이지

: 사이트를 이용하는 사용자 목록 조회, 사용자가 쓴 댓글, 게시글, 리뷰 조회 가능

: 닉네임, 이메일로 사용자 검색 조회 가능

관리자 페이지



사용자 예약 통계 분석 페이지

- : 카테고리별(놀거리, 음식, 항공권, 숙박, 지역) 상위 5개의 예약 통계 분석
- : 일주일 단위 일별로 사이트 방문자 수 통계 분석

관리자 페이지

여행 웹사이트 관리자 페이지

MAIN 사용자 조회 통계 조회 예약 현황 매출 분석 공지사항 신고 관리

예약 현황

사용자 닉네임	이메일	카테고리	예약 내용	예약 일자
맛집탐험가	foodie@example.com	음식점	서울 미쉐린 스타 레스토랑	2023-06-25
놀이공원매니아	themepark@example.com	놀거리	에버랜드 종일권	2023-07-01
국내여행러	koreatravel@example.com	지역	제주도 3박4일 패키지	2023-07-15
해외여행족	globetrotter@example.com	항공권	인천-파리 왕복	2023-08-01
호텔리뷰어	hotelreview@example.com	숙소	부산 해운대 5성급 호텔	2023-07-10

검색

관리자 페이지

₩ 매출 분석



매출 분석 페이지

: 사이트 총 매출 일별, 월별, 년별 매출 분석

Made with Gamma

관리자 페이지

여행 웹사이트 관리자 페이지

MAIN 사용자 조회 통계 조회 예약 현황 매출 분석 공지사항 신고 관리

공지사항 관리

번호	제목	작성일자
5	고객 만족도 조사 참여 안내	2023-05-25
4	코로나19 관련 여행 협약 변경	2023-05-30
3	앱 업데이트 안내	2023-06-05
2	새로운 파트너십 안내	2023-06-10
1	여행 즐기기 시즌 특별 할인	2023-06-15

+ 새 공지사항

여행 웹사이트 관리자 페이지

MAIN 사용자 조회 통계 조회 마케팅 분석 매출 분석 공지사항 신고 관리

공지사항 작성

제목
내용
작성하기

공지사항 페이지

: 공지사항 목록, 작성 페이지

관리자 페이지

여행 웹사이트 관리자 페이지

 MAIN  사용자 조회  통계 조회  예약 현황  매출 분석  공지사항  신고 관리

공지사항 수정

제목

내용

수정하기

삭제하기

관리자 페이지

여행 웹사이트 관리자 페이지

MAIN 사용자 조회 통계 조회 예약 현황 매출 분석 공지사항 신고 관리

신고 관리

신고 ID	신고자	피신고자	신고 유형	신고 사유	신고 날짜	컨텐츠 내용	상태
1	여행자1	세계탐험가	게시글	부적절한 내용	2023-06-20	이 지역은 정말 최악이에요. 절대 가지 마세요...	검토 중
2	맛집탐방	호텔리뷰어	댓글	욕설	2023-06-21	당신의 리뷰는 완전 쓰레기네요. ****...	검토 완료
3	포토그래퍼	여행자1	리뷰	허위 정보	2023-06-22	이 지역은 위생 상태가 엉망입니다. 바퀴벌레도 봤어요...	검토 중
4	여행자1	세계탐험가	게시글	부적절한 내용	2023-06-20	이 지역은 정말 최악이에요. 절대 가지 마세요...	검토 중
5	여행자1	세계탐험가	게시글	부적절한 내용	2023-06-20	이 지역은 정말 최악이에요. 절대 가지 마세요...	검토 완료

검색

문의하기 페이지

HAPLA

로그인  User Profile

문의하기

문의하실 사항은 aaaaa@naver.com 으로 문의해 주세요

상담 가능 시간

평일: 오전 9시 - 오후 6시

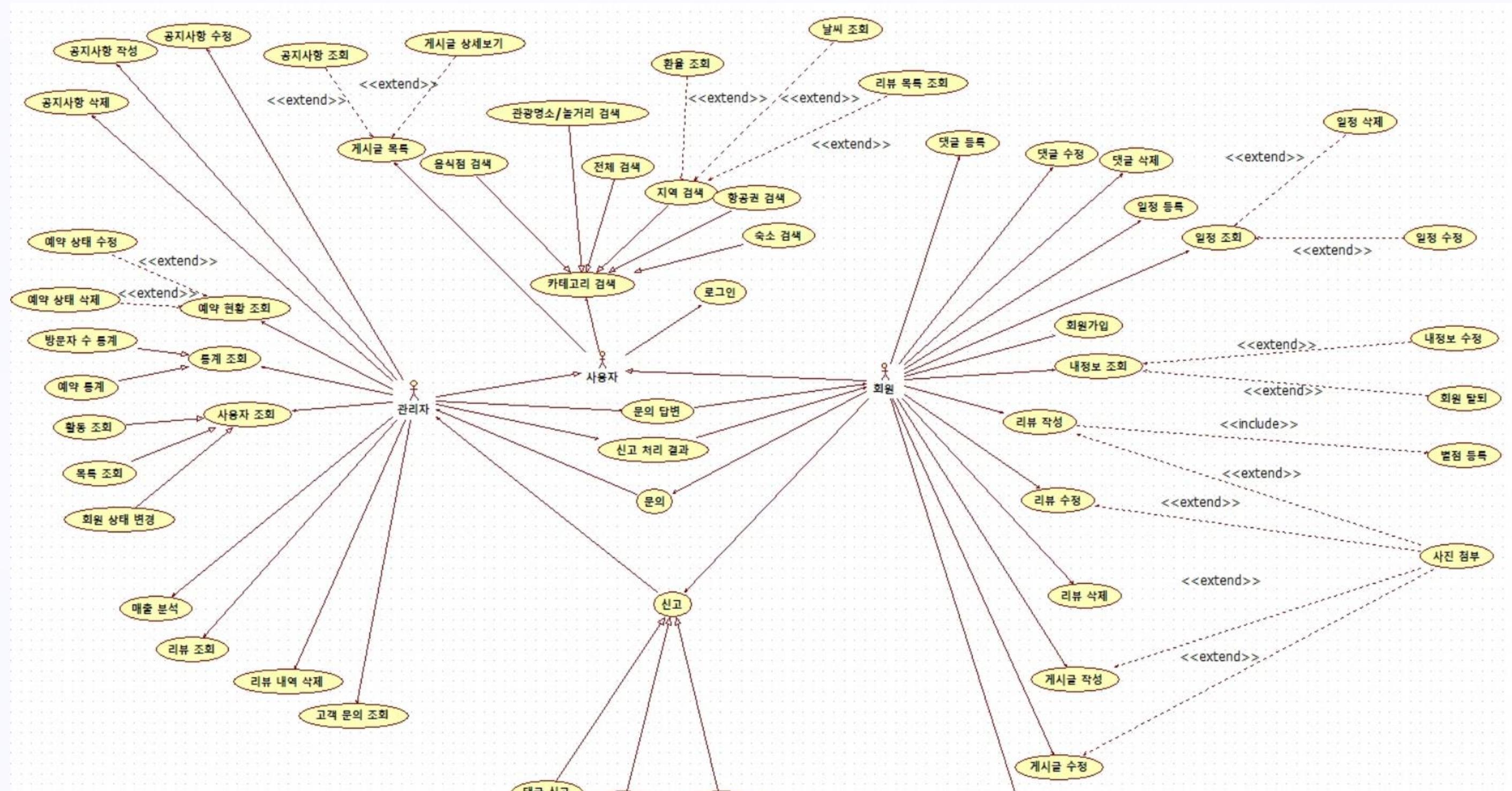
점심시간: 오후 12시 - 오후 1시

주말 및 공휴일: 휴무

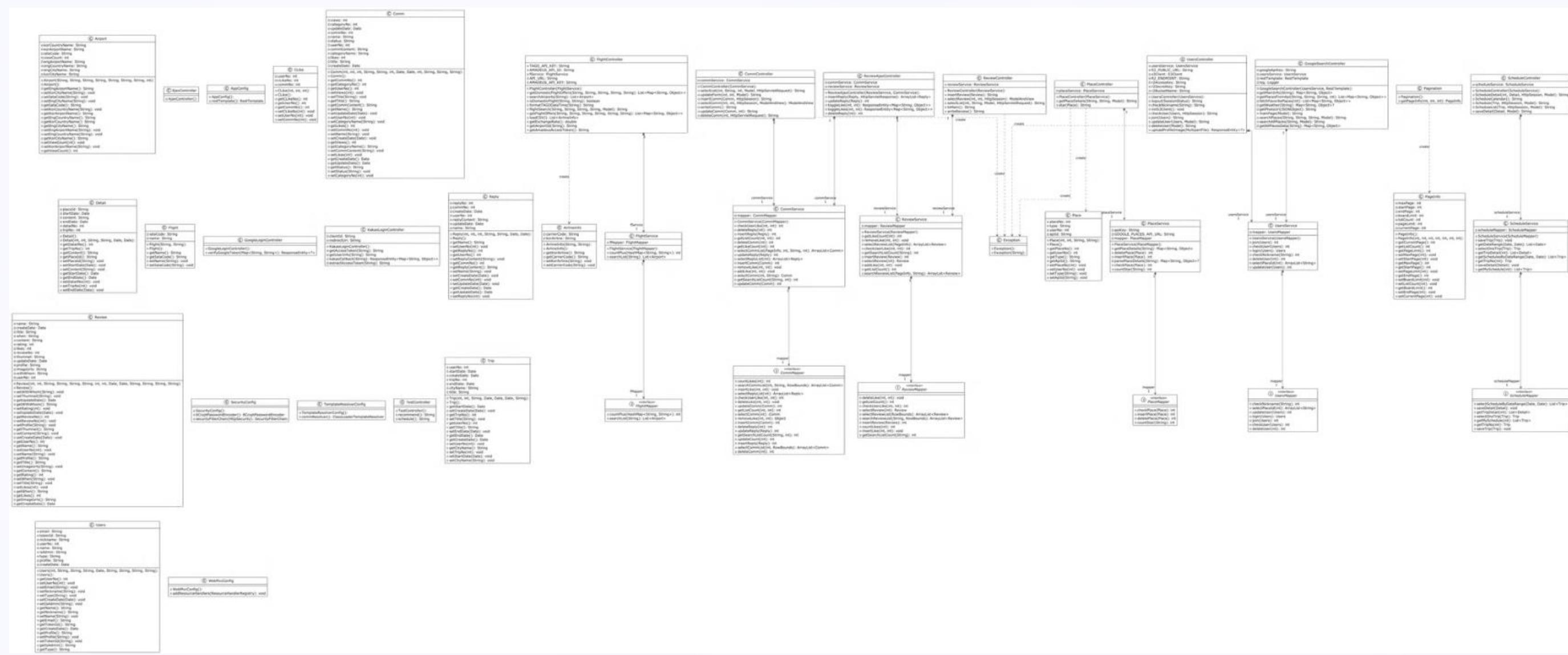
자주 묻는 질문

- 예약 취소 문의
- 여행 일정 변경 문의
- 패키지 여행 상품 문의

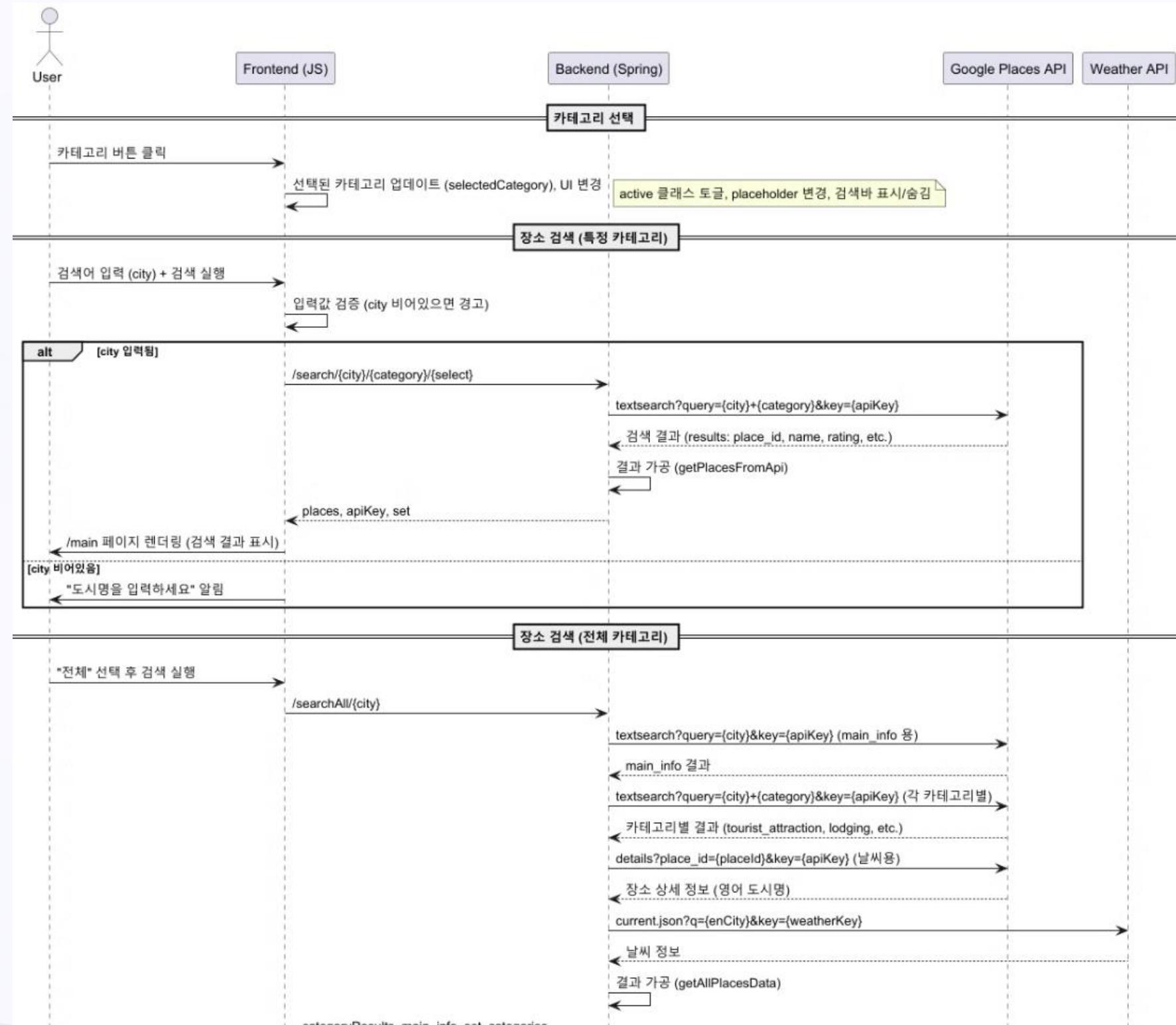
Use Case Diagram

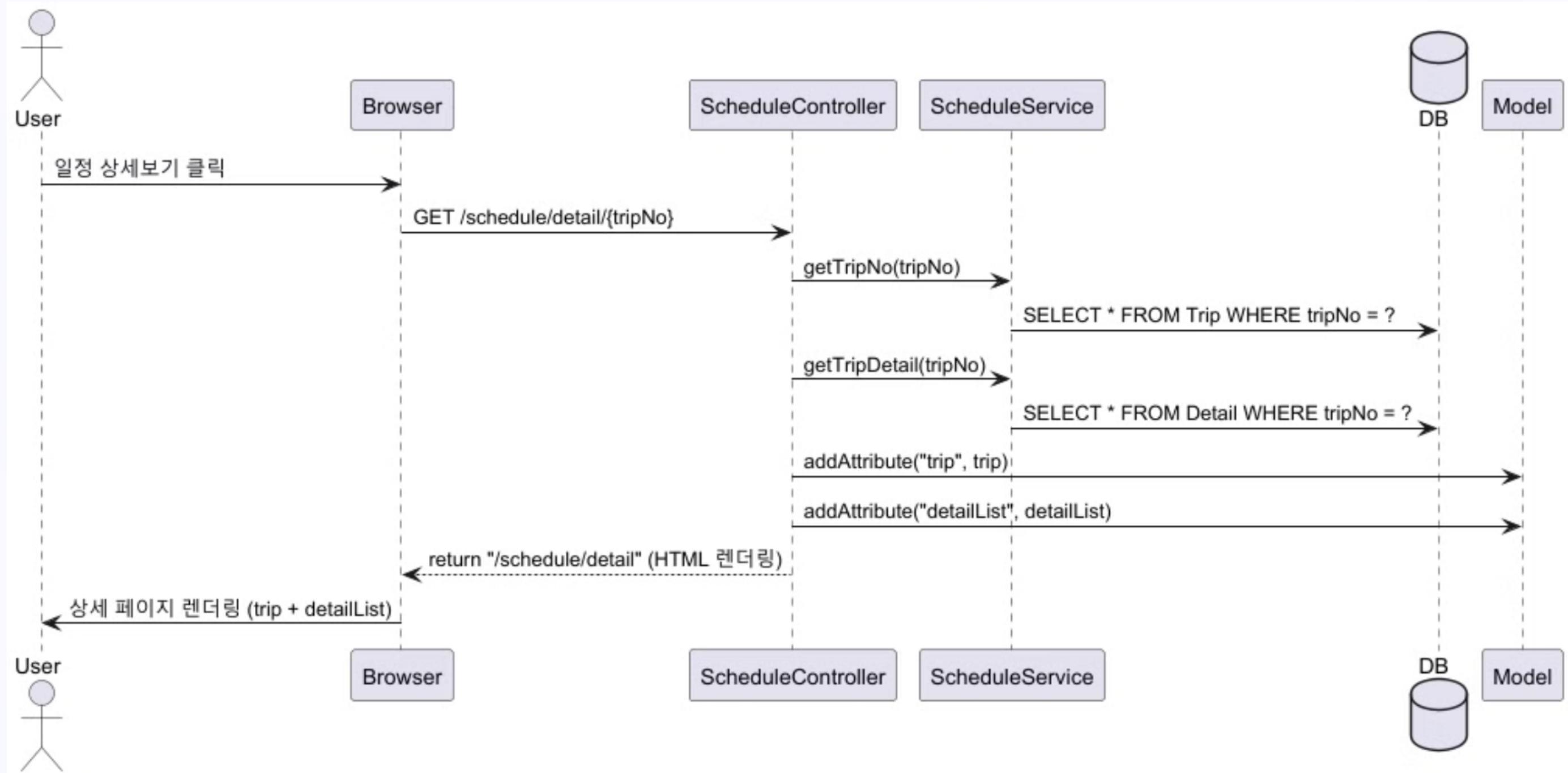


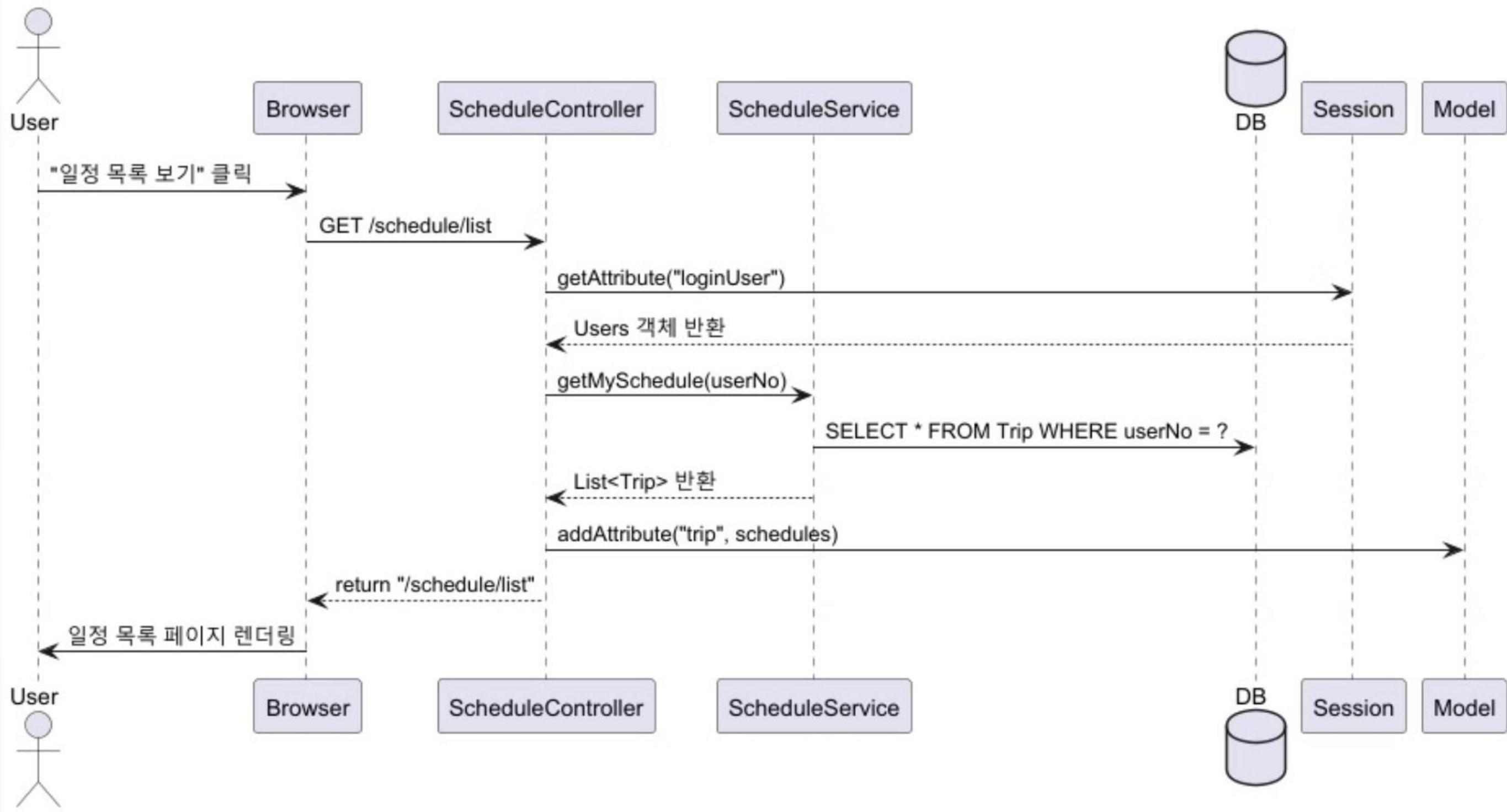
Class Diagram

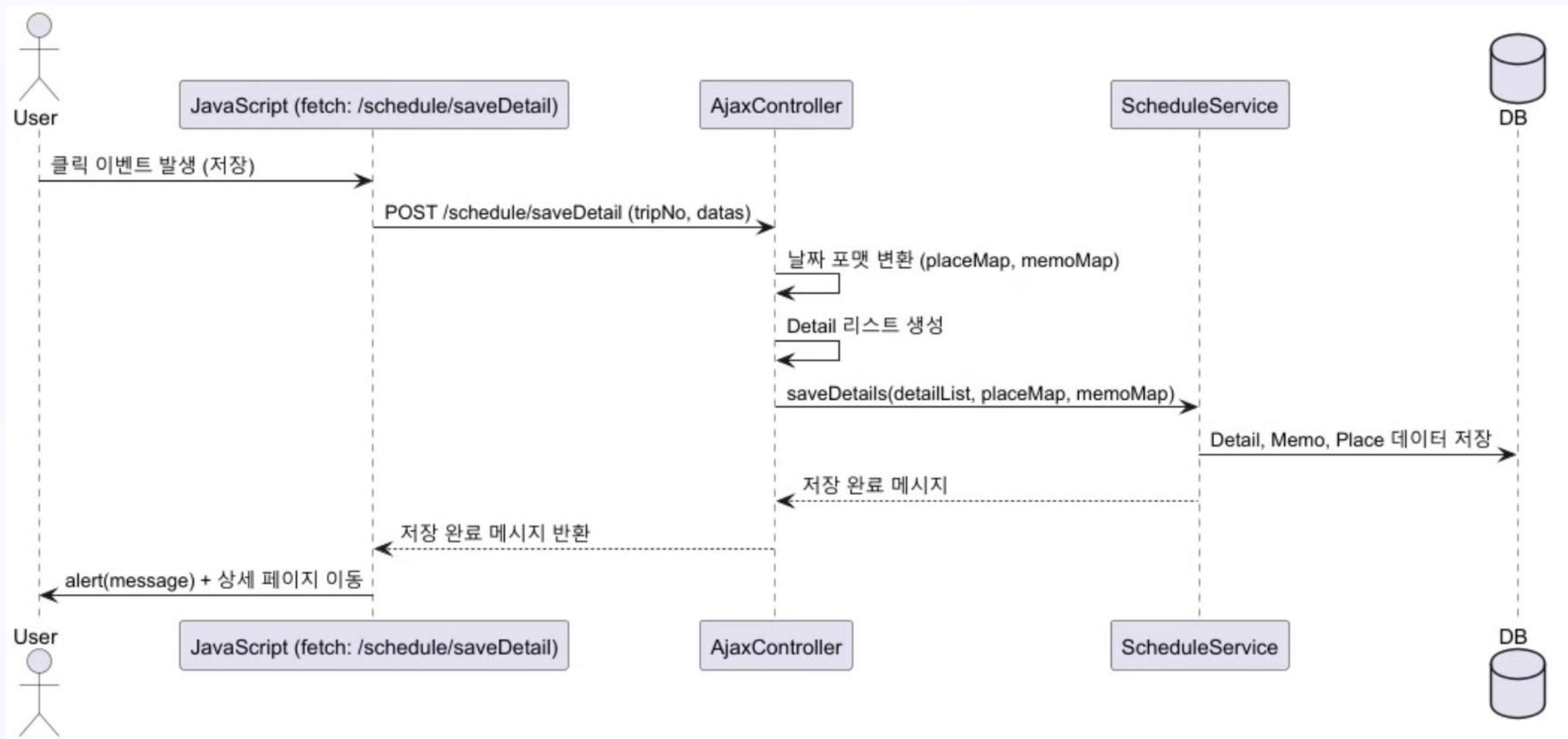


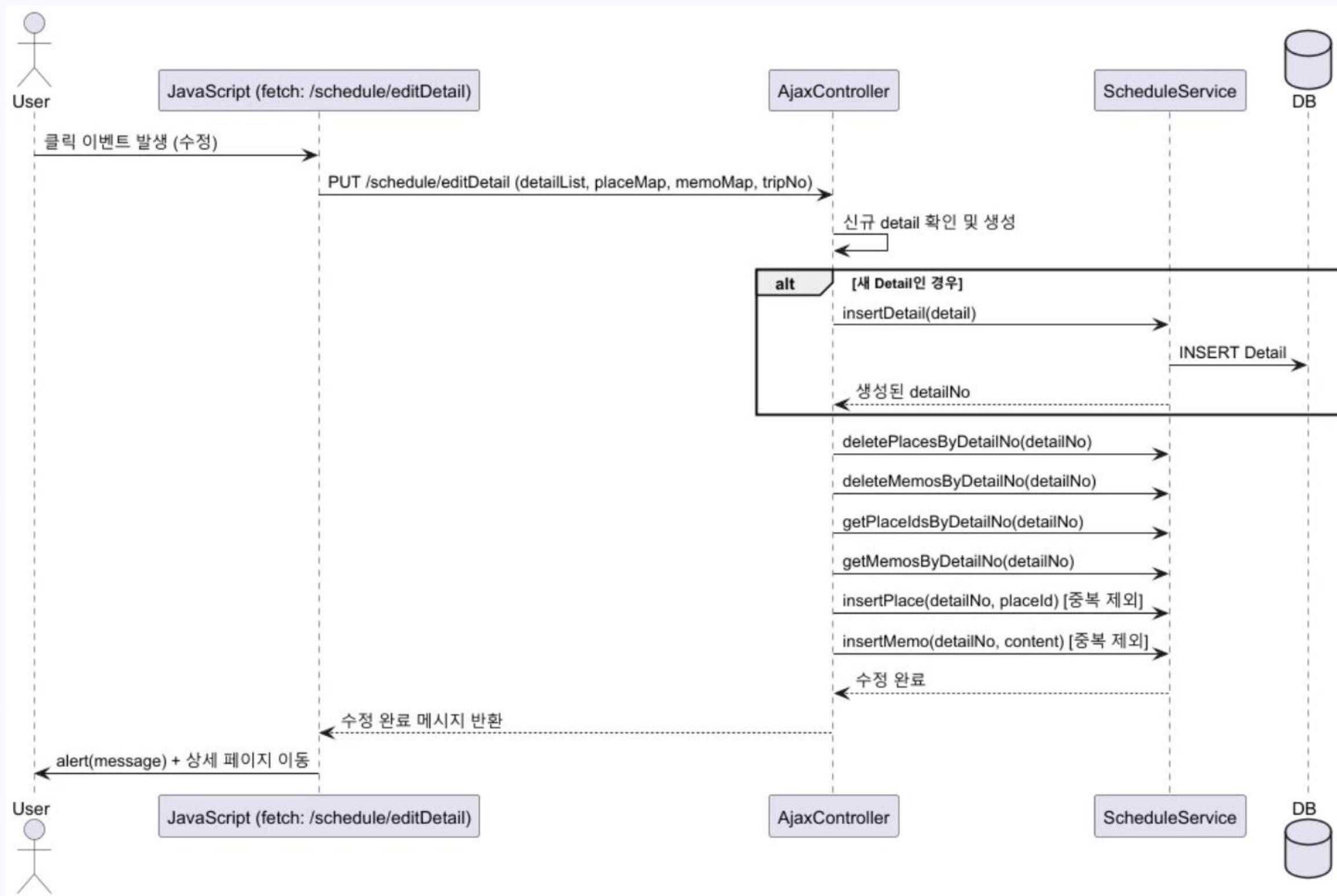
Sequence Diagram

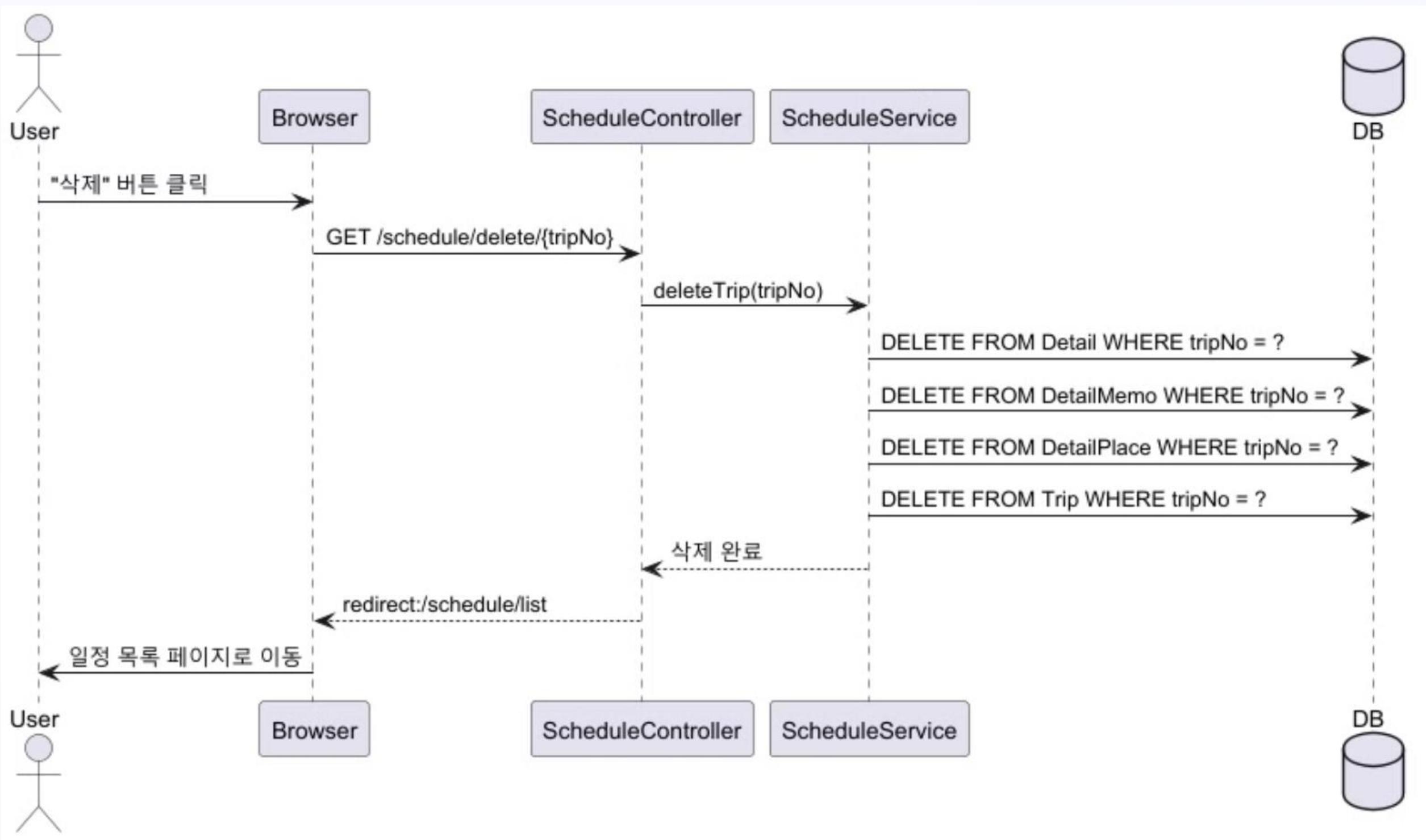


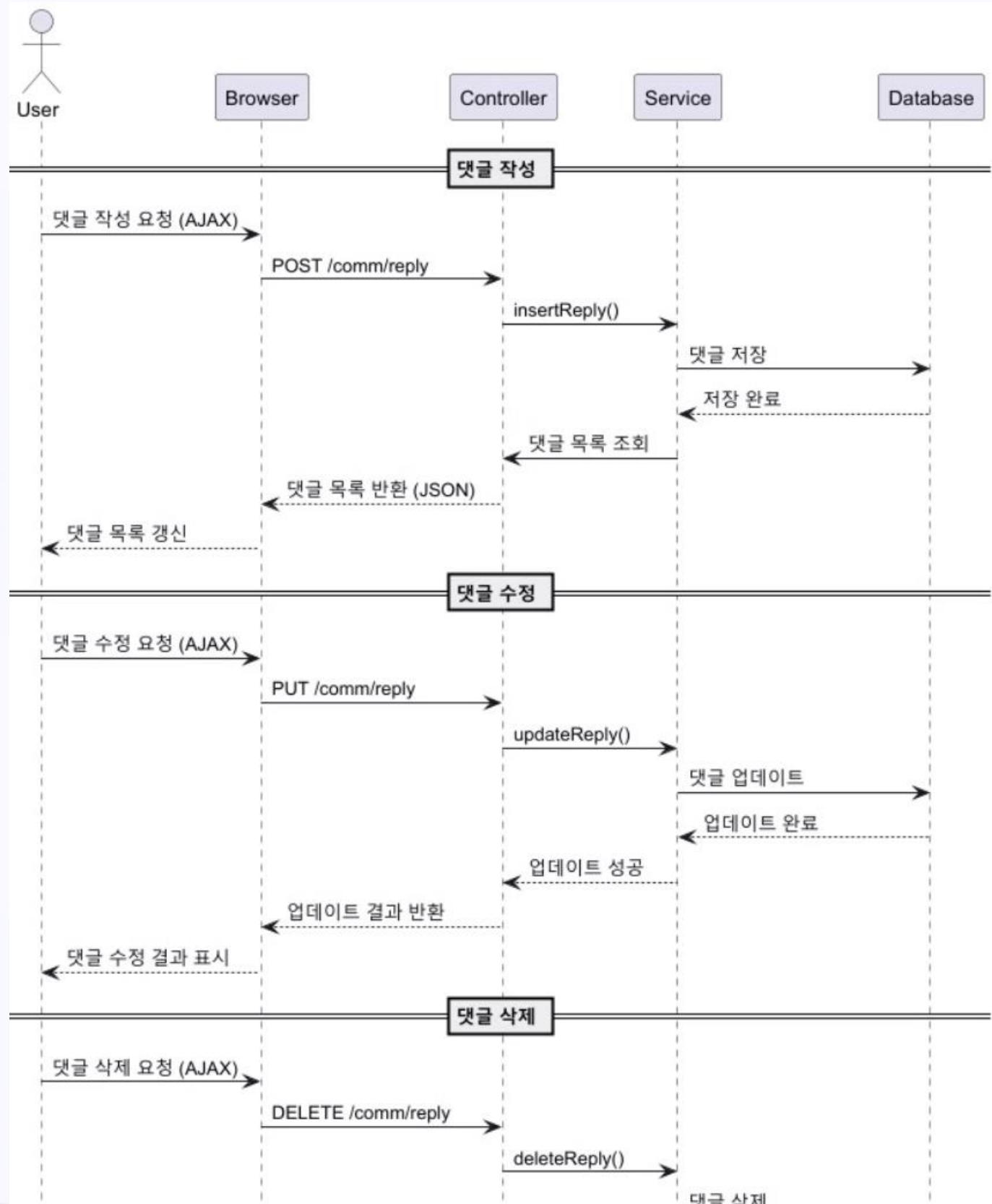


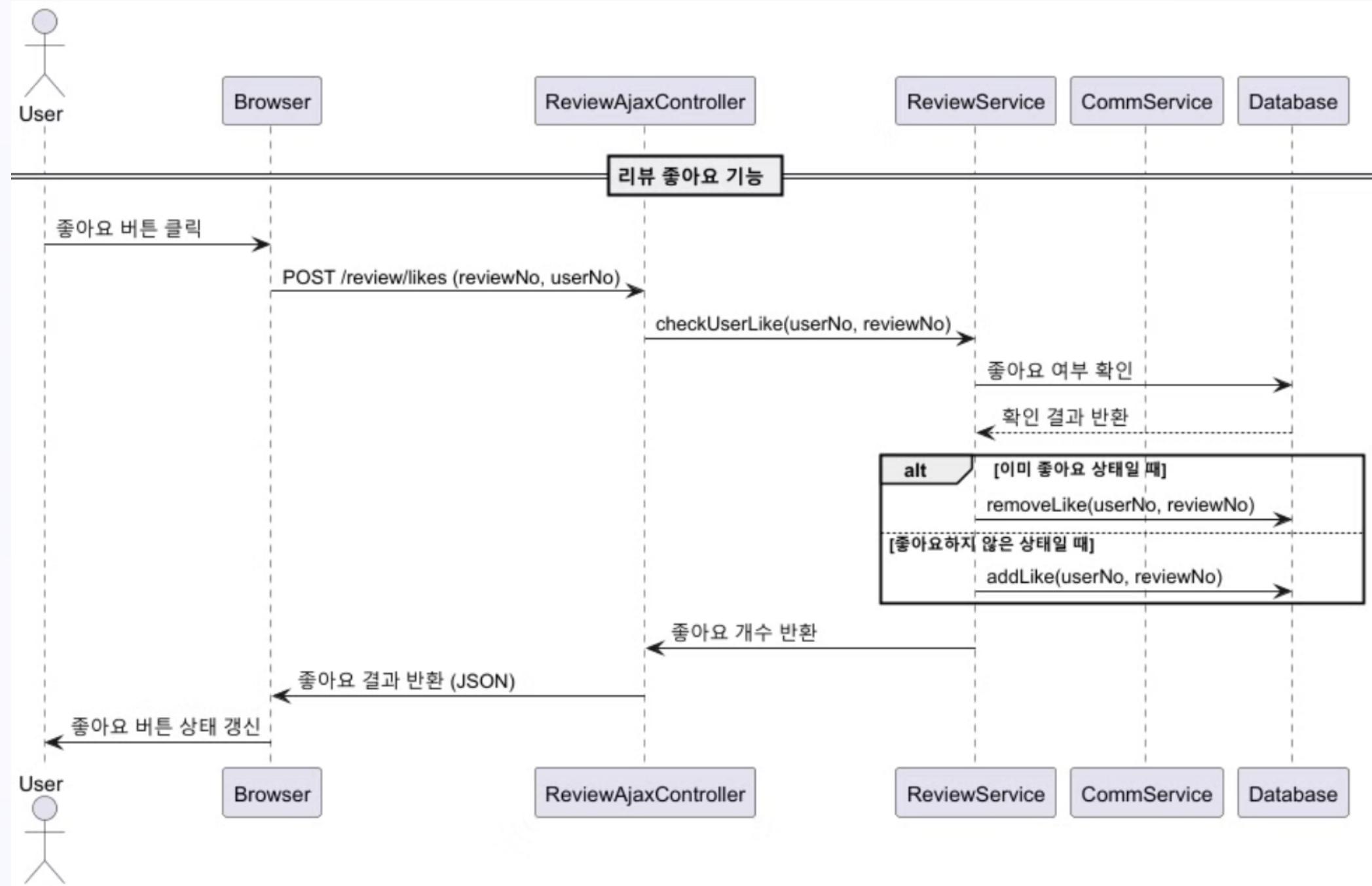




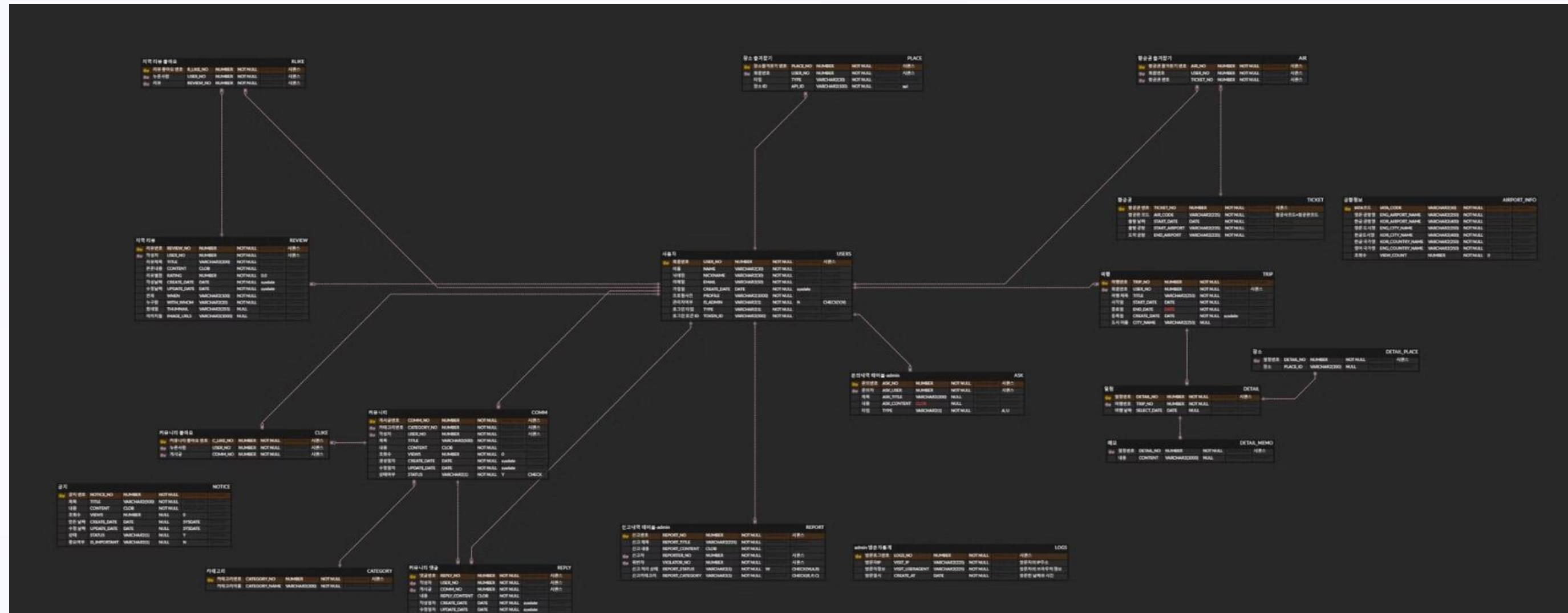








Entity-Relationship Diagram



테이블 정의서

HAPLA Table Layout

USERS

Attribute Name	Column Name	Data type	Null Option	Is PK	Is FK	Parent table	Default	Comment
회원번호	USER_NO	NUMBER	NOT NULL	Yes	No			시퀀스
이름	NAME	VARCHAR2(30)	NOT NULL	No	No			
닉네임	NICKNAME	VARCHAR2(30)	NOT NULL	No	No			
이메일	EMAIL	VARCHAR2(50)	NOT NULL	No	No			
가입일	CREATE_DATE	DATE	NOT NULL	No	No			SYSDATE
프로필사진	PROFILE	VARCHAR2(1000)	NOT NULL	No	No			
관리자여부	IS_ADMIN	VARCHAR2(1)	NOT NULL	No	No		N	Y, N
로그인 타입	TYPE	VARCHAR2(1)	NOT NULL	No	No			
로그인 토큰 ID	TOKEN_ID	VARCHAR2(500)	NOT NULL	No	No			

COMM

Attribute Name	Column Name	Data type	Null Option	Is PK	Is FK	Parent table	Default	Comment
게시글번호	COMM_NO	NUMBER	NOT NULL	Yes	No			시퀀스
카테고리번호	CATEGORY_NO	NUMBER	NOT NULL	No	Yes	USERS		
작성자	USER_NO	NUMBER	NOT NULL	No	Yes	CATEGORY		
제목	TITLE	VARCHAR2(500)	NOT NULL	No	No			
내용	CONTENT	CLOB	NOT NULL	No	No			
조회수	VIEWS	NUMBER	NOT NULL	No	No		0	
생성일자	CREATE_DATE	DATE	NOT NULL	No	No			nma

CATEGORY

Attribute Name	Column Name	Data type	Null Option	Is PK	Is FK	Parent table	Default	Comment
카테고리번호	CATEGORY_NO	NUMBER	NOT NULL	Yes	No			시퀀스
카테고리이름	CATEGORY_NAME	VARCHAR2(200)	NOT NULL	No	No			

REPLY

Attribute Name	Column Name	Data type	Null Option	Is PK	Is FK	Parent table	Default	Comment
댓글번호	REPLY_NO	NUMBER	NOT NULL	Yes	No			시퀀스
작성자	USER_NO	NUMBER	NOT NULL	No	Yes	USERS		
게시글	COMM_NO	NUMBER	NOT NULL	No	Yes	COMM		
내용	REPLY_CONTENT	CLOB	NOT NULL	No	No			
작성일자	CREATE_DATE	DATE	NOT NULL	No	No		SYSDATE	
수정일자	UPDATE_DATE	DATE	NOT NULL	No	No		SYSDATE	

CLIKE

Attribute Name	Column Name	Data type	Null Option	Is PK	Is FK	Parent table	Default	Comment
커뮤니티 좋아요 번호	C_LIKE_NO	NUMBER	NOT NULL	Yes	No			시퀀스
누른사람	USER_NO	NUMBER	NOT NULL	No	Yes	USERS		
게시글	COMM_NO	NUMBER	NOT NULL	No	Yes	COMM		

REVIEW

Attribute Name	Column Name	Data type	Null Option	Is PK	Is FK	Parent table	Default	Comment
리뷰번호	REVIEW_NO	NUMBER	NOT NULL	Yes	No			시퀀스
작성자	USER_NO	NUMBER	NOT NULL	No	Yes	USERS		
리뷰제목	TITLE	VARCHAR2(200)	NOT NULL	No	No			
본문내용	CONTENT	CLOB	NOT NULL	No	No			
리뷰별점	RATING	NUMBER	NOT NULL	No	No		0	소수
작성날짜	CREATE_DATE	DATE	NOT NULL	No	No			SYSDATE
수정날짜	UPDATE_DATE	DATE	NOT NULL	No	No			SYSDATE
언제	WHEN	VARCHAR2(100)	NOT NULL	No	No			
누구랑	WITH_WHOM	VARCHAR2(20)	NOT NULL	No	No			
썸네일	THUMBNAIL	VARCHAR2(255)	NULL	No	No			
이미지들	IMAGE_URLS	VARCHAR2(3000)	NULL	No	No			

RLIKE

Attribute Name	Column Name	Data type	Null Option	Is PK	Is FK	Parent table	Default	Comment
리뷰 좋아요 번호	R_LIKE_NO	NUMBER	NOT NULL	Yes	No			시퀀스
누른사람	USER_NO	NUMBER	NOT NULL	No	Yes	USERS		
리뷰	REVIEW_NO	NUMBER	NOT NULL	No	Yes	REVIEW		

PLACE

Attribute Name	Column Name	Data type	Null Option	Is PK	Is FK	Parent table	Default	Comment
장소 즐겨찾기 번호	PLACE_NO	NUMBER	NOT NULL	Yes	No			시퀀스
회원번호	USER_NO	NUMBER	NOT NULL	No	Yes	USERS		
타입	TYPE	VARCHAR2(30)	NOT NULL	No	No			
장소 ID	API_ID	VARCHAR2(100)	NOT NULL	No	No			API

TRIP

Attribute Name	Column Name	Data type	Null Option	Is PK	Is FK	Parent table	Default	Comment
여행번호	TRIP_NO	NUMBER	NOT NULL	Yes	No			시퀀스
회원번호	USER_NO	NUMBER	NOT NULL	Yes	Yes	USERS		
여행 제목	TITLE	VARCHAR2(255)	NOT NULL	No	No			
시작일	START_DATE	DATE	NOT NULL	No	No			
종료일	END_DATE	DATE	NOT NULL	No	No			
등록일	CREATE_DATE	DATE	NOT NULL	No	No		SYSDATE	
도시 이름	CITY_NAME	VARCHAR2(255)	NULL	No	No			

DETAIL

Attribute Name	Column Name	Data type	Null Option	Is PK	Is FK	Parent table	Default	Comment
일정번호	DETAIL_NO	NUMBER	NOT NULL	Yes	No			시퀀스
여행번호	TRIP_NO	NUMBER	NOT NULL	No	Yes	TRIP		
일정 내용	CONTENT	CLOB	NULL	No	No			
장소 번호	PLACE_ID	VARCHAR2(200)	NULL	No	No			API
여행 날짜	SELECT_DATE	DATE	NULL	No	No		SYSDATE	

ASK

Attribute Name	Column Name	Data type	Null Option	Is PK	Is FK	Parent table	Default	Comment
문의번호	ASK_NO	NUMBER	NOT NULL	Yes	No			시퀀스
문의자	ASK_USER	NUMBER	NOT NULL	No	Yes	USERS		
문의제목	ASK_TITLE	VARCHAR2(200)	NOT NULL	No	No			
문의내용	ASK_CONTENT	CLOB	NOT NULL	No	No			

ANSWER

Attribute Name	Column Name	Data type	Null Option	Is PK	Is FK	Parent table	Default	Comment
문의번호	ANSWER_NO	NUMBER	NOT NULL	Yes	No			시퀀스
문의자	ASK_NO	NUMBER	NOT NULL	No	Yes	ASK		
제목	TITLE	CLOB	NULL	No	No			
내용	CONTENT	VARCHAR2(100)	NULL	No	No			
타입	TYPE	VARCHAR2(1)	NOTNULL	No	No		U	A,U

REPORT

Attribute Name	Column Name	Data type	Null Option	Is PK	Is FK	Parent table	Default	Comment
신고번호	REPORT_NO	NUMBER	NOT NULL	Yes	No			시퀀스
신고 제목	REPORT_TITLE	VARCHAR2(225)	NOT NULL	No	No			
신고 내용	REPORT_CONTENT	CLOB	NOT NULL	No	No			
신고자	REPORTER_NO	NUMBER	NOT NULL	No	Yes	USERS		
위반자	VIOLATOR_NO	NUMBER	NOT NULL	No	Yes	USERS		
신고 처리 상태	REPORT_STATUS	VARCHAR2(1)	NOT NULL	No	No		W	W, A, R
신고 카테고리	REPORT_CATEGORY	VARCHAR2(1)	NOT NULL	No	No			R, P, C

LOGS

Attribute Name	Column Name	Data type	Null Option	Is PK	Is FK	Parent table	Default	Comment
방문로그번호	LOGS_NO	NUMBER	NOT NULL	Yes	No			시퀀스
방문자IP	VISIT_IP	VARCHAR2(225)	NOT NULL	No	No			방문자의 IP주소
방문자정보	VISIT_USERAGENT	VARCHAR2(225)	NOT NULL	No	No			방문자의 브라우저 정보
방문일시	CREATE_AT	DATE	NOT NULL	No	No			방문한 날짜와 시간

코드 리뷰

카카오 로그인

자바스크립트

- **SDK 초기화:** `Kakao.init()`을 통해 카카오 SDK를 초기화합니다.
- **로그인 요청:** 사용자가 버튼을 클릭하면 `Kakao.Auth.login()`이 실행되며, 프로필 정보를 가져올 권한을 요청합니다.
- **사용자 정보 조회:** 로그인 성공 후, `fetchUserInfo()`를 통해 사용자 정보를 가져오고, `checkUser()`로 추가 처리합니다.

```
// ✅ 카카오 SDK 초기화
Kakao.init("04cff7eb62f268a2bb506c81bf1de17b"); // 카카오 JavaScript 키를 사용해 SDK 초기화

// ✅ 카카오 로그인 버튼 클릭 시 실행
document.getElementById("kakao-login-btn").addEventListener("click", listener: function () : void {
    Kakao.Auth.login({
        scope: "profile_nickname, profile_image", // 요청할 사용자 정보 범위 (닉네임, 프로필 이미지)
        success: function (authObj) : void {
            console.log("✅ 로그인 성공! 액세스 토큰:", authObj.access_token); // 로그인 성공 시 액세스 토큰 출력

            // ✅ 액세스 토큰을 이용하여 사용자 정보 가져오기
            fetchUserInfo(authObj.access_token); // 사용자 정보를 가져오는 함수 호출
        },
        fail: function (err) : void {
            console.error("✗ 로그인 실패", err); // 로그인 실패 시 에러 출력
        },
    });
});
```

```
// ✅ 사용자 정보 가져와서 모달 업데이트
function fetchUserInfo(token) : void { Show usages
    Kakao.API.request({
        url: "/v2/user/me", // 카카오 API에서 사용자 정보를 가져오는 엔드포인트
        success: function (response) : void {
            console.log("✅ 사용자 정보:", response); // 가져온 사용자 정보 출력
            console.log(response.id); // 사용자 ID 출력

            loginImage = response.properties.profile_image; // 프로필 이미지 URL 저장
            loginName = response.properties.nickname; // 닉네임 저장
        }
    });
}
```

항공편 필터링

```
const applyFilter = (flightData) => {
  let hasInbound = null;

  const flightContainers = document.querySelectorAll('.flight-container');

  // 경유 필터링 체크박스
  const checkedLayovers = document.querySelectorAll('input[name="layover"]:checked');
  const selectedLayoverOptions = Array.from(checkedLayovers).map(input => input.value);

  // 가는 날 출발 시간 필터링 체크박스
  const outChecked = document.querySelectorAll('input[name="out-departure-time"]:checked');
  const outSelectedOptions = Array.from(outChecked).map(input => input.value);

  // 오는 날 출발 시간 필터링 체크박스
  const inChecked = document.querySelectorAll('input[name="in-departure-time"]:checked');
  const inSelectedOptions = Array.from(inChecked).map(input => input.value);

  const airline = document.querySelectorAll('input[name="airline-filter"]:checked');
  const airlineSelectedOptions = Array.from(airline).map(input => input.value);

  flightContainers.forEach((flightContainer, index) => {
    const flight = flightData[index] || {};
    const outboundStops = parseInt(flightContainer.dataset.outboundStops || 0);
    const inboundStops = parseInt(flightContainer.dataset.inboundStops || 0);

    // 시간 데이터에서 "HH:mm:ss" 형식으로 변환
    const outboundTime = (flightContainer.dataset.outboundTime?.split('T')[1] || '').replace(/:/g, '');
  });
}
```

```
// 1. 경유 필터링
if (selectedLayoverOptions.length > 0) {
    const isDirect = selectedLayoverOptions.includes('direct');
    const isOneStop = selectedLayoverOptions.includes('oneStop');
    const isMultiStop = selectedLayoverOptions.includes('multiStop');

    if (
        // 직항만 선택: 출발 또는 귀국이 직항이 아니면 숨김
        (isDirect && !isOneStop && !isMultiStop && (outboundStops != 0 || (hasInbound && inboundStops != 0))) ||
        // 1회 경유만 선택: 출발과 귀국(있을 경우)이 1회가 아니면 숨김
        (isOneStop && !isDirect && !isMultiStop &&
            (outboundStops != 1 || (hasInbound && inboundStops != 1))) ||
        // 2회 이상 경유만 선택: 출발과 귀국(있을 경우)이 2회 미만이면 숨김
        (isMultiStop && !isDirect && !isOneStop &&
            (outboundStops < 2 && (!hasInbound || inboundStops < 2))) ||
        // 직항 + 1회 경유: 2회 이상 경유 숨김
        (isDirect && isOneStop && !isMultiStop &&
            (outboundStops >= 2 || (hasInbound && inboundStops >= 2))) ||
        // 1회 경유 + 2회 이상 경유: 직항 숨김
        (isOneStop && isMultiStop && !isDirect &&
            (outboundStops == 0 || (hasInbound && inboundStops == 0))) ||
        // 직항 + 2회 이상 경유: 1회 경유 숨김
        (isDirect && isMultiStop && !isOneStop &&
            (outboundStops == 1 || (hasInbound && inboundStops == 1)))
    ) {
        shouldHide = true; // 조건에 해당하면 숨김 플래그 설정
    }
}

// 2. 가는 날 출발 시간 필터링
```

```
// 3. 오는 날 출발 시간 필터링 (왕복일 경우에만 적용)
if (hasInbound && inSelectedOptions.length > 0 && !shouldHide) {
    const inboundTimeSlot = getTimeSlot(inboundTime);
    const timeMatch = inSelectedOptions.includes(inboundTimeSlot);
    if (!timeMatch) {
        shouldHide = true; // 시간대가 선택된 옵션과 맞지 않으면 숨김
    }
}

if (airlineSelectedOptions.length > 0 && !shouldHide) {
    if (!airlineSelectedOptions.some(option => flight.airline.includes(option))) {
        // .some -> 일치하는게 하나라도 있으면 true 반환(배열을 돌다가 한번이라도 true면 true 반환 후 탈출)
        shouldHide = true;
    }
}

// 최종적으로 숨김 여부 결정
if (shouldHide) {
    flightContainer.classList.add('hidden');
}

});
```

CSV로드

```
// 조회해온 항공편 데이터에서 항공사 명을 한글로 띄워주기 위한 CSV 파일 로드
private final FlightService fService;

public List<AirlineInfo> loadCSV() {
    List<AirlineInfo> airlineList = new ArrayList<>();
    try (CSVReader csvReader = new CSVReader(new FileReader("src/main/resources/static/csv/airlineNames.csv"))) {
        String[] values;
        while ((values = csvReader.readNext()) != null) {
            AirlineInfo airline = new AirlineInfo();
            airline.setKorAirline(values[1]);
            airline.setCarrierCode(values[2]);

            airlineList.add(airline);
        }
    } catch (FileNotFoundException e) {
        e.printStackTrace();
    } catch (CsvValidationException e) {
        e.printStackTrace();
    } catch (IOException e) {
        e.printStackTrace();
    }
    return airlineList;
}
```



Travel Story

전체

도시 이름을 입력하세요...



항공권

서울, 대한민국

부산, 대한민국

제주, 대한민국

오사카, 일본

도쿄, 일본

방콕, 태국

즐겨찾기

HAPLA 소개

[회사 소개](#)[제휴 정보](#)[보도자료](#)

여행자 지원

[고객센터](#)[자주 묻는 질문](#)[여행자 보험](#)

미스너스 지원

[광고 문의](#)[제휴 문의](#)[호텔 등록](#)

1. 도시 선택 이벤트 등록

```
autocomplete.addListener("place_changed", function () {
  const place = autocomplete.getPlace();
  if (!place.geometry || !place.types) {
    console.error("도시 정보를 찾을 수 없습니다.");
    return;
}
```

- ‘autocomplete’은 Google Maps API에서 제공하는 자동완성 객체
- 사용자가 도시를 선택하면 “place_changed”이벤트 발생
- getPlace()로 선택된 장소 정보를 가져옴

2. 주소 파싱: 도시명 & 국가명

```
let addressParts = place.formatted_address.split(", ");
let cityName = addressParts[0];
const countryName = addressParts[addressParts.length - 1];
```

- `formatted_address`: 구글이 반환하는 완전한 주소 문자열
- `split(",")`을 통해 쉼표로 나누어 배열로 변환
 - 첫번째 요소는 도시명
 - 마지막 요소는 국가명

3. 선택된 도시 정보를 localStorage에 저장

```
selectCity({  
    name: cityName,  
    country: countryName,  
    lat: place.geometry.location.lat(),  
    lng: place.geometry.location.lng(),  
});
```

```
function selectCity(city) {  
    localStorage.setItem("selectedCity", JSON.stringify(city));  
    window.location.href = "/schedule/calendar";  
}
```

- 도시 정보를 객체로 만들어 localStorage에 JSON 문자열로 저장
- 저장 후 “/schedule/calendar” 페이지로 이동함

4. 일정 페이지에서 정보 활용

```
function initMap() {  
  const storedCity = localStorage.getItem("selectedCity");
```

- 일정 페이지에 진입하면 localStorage에서 도시정보를 꺼내 활용

1. 내가 누른 좋아요 빨간 하트로 변경

```
<button id="LikeBtn" class="btn btn-outline-danger btn-sm" style="margin-left: 10px;">
    <!-- 좋아요가 눌리지 않은 상태(빈 하트) -->
    <svg th:attr="style=${isLiked == false} ? 'display: inline;' : 'display: none;'" f:
        <g>
            <path d="M433.601,67.001c-24.7-24.7-57.4-38.2-92.3-38.2s-67.7,13.6-92.4,38
        </g>
    </svg>

    <!-- 좋아요가 눌린 상태(빨간색 하트) -->
    <svg th:attr="style=${isLiked == true} ? 'display: inline;' : 'display: none;'" he:
        <g>
            <path fill="red" d="M433.601,67.001c-24.7-24.7-57.4-38.2-92.3-38.2s-67.7,13.6-92.4,38
        </g>
    </svg>
    <span id="LikeCount" th:text="${r.likes != null ? r.likes : 0}"></span>
</button>
```

```
$.ajax({
    url: "/review/likes",
    type: "POST",
    contentType: "application/x-www-form-urlencoded",
    data: {
        reviewNo: /*[$r.reviewNo]]*/ 1,
        userNo: /*[$session.loginUser.userNo]]*/1
    },
    success: function(response) {
        // 좋아요 개수 업데이트
        $("#likeCount").text(response.likes);

        // 좋아요 상태 확인 및 하트 색상 변경
        if (response.isLiked) {
            // 좋아요가 눌린 상태 - 빨간 하트 보여주기
            $("#heart-outline").hide();
            $("#heart-fill").show();
        } else {
            // 좋아요가 눌리지 않은 상태 - 빈 하트 보여주기
            $("#heart-fill").hide();
            $("#heart-outline").show();
        }
    }
})
```

```
@PostMapping("/likes")
public ResponseEntity <Map<String, Object>> toggleLikes(@RequestParam("reviewNo") int reviewNo,
                                                               @RequestParam("userNo") int userNo) {
    Map<String, Object> response = new HashMap<>();
    try {
        boolean isLiked = reviewService.checkUserLike(userNo, reviewNo) > 0;
        if (isLiked) {
            reviewService.removeLike(userNo, reviewNo);
        } else {
            reviewService.addLike(userNo, reviewNo);
        }
        int updatedLikeCount = reviewService.getLikeCount(reviewNo);
        response.put("status", "success");
        response.put("likes", updatedLikeCount);
        response.put("isLiked", !isLiked);
        return ResponseEntity.ok(response);
    } catch (Exception e) {
        e.printStackTrace();
        response.put("status", "error");
        response.put("message", "서버 오류 발생: " + e.getMessage());
        return ResponseEntity.status(HttpStatus.INTERNAL_SERVER_ERROR).body(response);
    }
}
```



이창
2025/03/23

제주도 다녀왔어요~



언제 다녀오셨나요?

1년 이상 이전

누구와 함께 가셨나요?

친구



삭제

목록



이창
2025/03/23

제주도 다녀왔어요~



언제 다녀오셨나요?

1년 이상 이전

누구와 함께 가셨나요?

친구



삭제

목록



2. 리뷰 작성 시 사진 미리보기 기능

```
<h3>미리보기</h3>
<div id="imageSection">
  <!-- 미리보기 이미지 -->
</div>
<input type="hidden" id="image-urls" name="imageUrls" value=""> <!-- 업로드된 이미지 URL 저장 -->

fetch('/upload', {
  method: 'POST',
  body: formData
})
.then(response => {
  if (!response.ok) throw new Error('업로드 실패: ' + response.status);
  return response.json();
})
.then(data => {
  console.log("받은 데이터:", data);
  if (data.imagePath) {
    document.getElementById('imageSection').innerHTML +=
      ``;
    console.log(data.imagePath)
    const url = document.getElementById('image-urls');
    if (url.value.trim() != ''){
      url.value += ',' + data.imagePath
    }else{
      url.value = data.imagePath;
    }
    console.log(url)
  }
})
.catch(error => {
  console.error('이미지 업로드 에러:', error);
  alert('이미지 업로드에 실패했습니다. 다시 시도해주세요.');
});
```

6. 사진 첨부

파일 선택

미리보기



리뷰 작성 완료

게시글 승인, 반려 처리

```
reportButton.forEach((btn) => {
  btn.addEventListener("click", function () {
    const tr = this.closest("tr");
    const status = tr.querySelector(".status");
    const reportNo = this.getAttribute("data-reportno");
    const newStatus = this.classList.contains('approve') ? 'A' : 'R';
    $.ajax({
      url : 'status',
      type : 'post',
      data : {
        reportStatus : newStatus,
        reportNo : reportNo
      },
      success : data => {
        if(data == 'success') {
          if (this.classList.contains("approve")) {
            status.innerText = "승인"
            status.className = "status approved"
            status.style.background = "#1cc88a"
          } else if (this.classList.contains("reject")) {
            status.innerText = "반려"
            status.className = "status rejected"
            status.style.background = "#e74a3b";
          }
        }
        // 처리 완료된 신고는 버튼 비활성화
        tr.querySelectorAll(".action-btn.approve, .action-btn.reject").forEach((btn) => {
          btn.disabled = true;
          btn.remove();
        });
      }
    },
    error : data => {
      alert("신고 처리 중 오류가 발생했습니다.");
    }
  })
});
```

```
@PostMapping("admin/status")
@ResponseBody
public String updateStatus(@ModelAttribute Report report, @RequestParam("reportNo") int reportNo,
                           @RequestParam("reportStatus") String reportStatus) {
    report.setReportNo(reportNo);
    report.setReportStatus(reportStatus);

    // 신고 상태 업데이트
    int result = aService.updateStatus(report);

    // 승인된 경우 해당 게시글/댓글 삭제
    if (reportStatus.equals("A")) {
        // 신고 대상 카테고리 확인 후 삭제 처리
        Report reportDetail = aService.getReportDetail(reportNo); // 신고 목록 조회(3회 이상 신고된 항목만 조회)
        if (reportDetail != null) {
            if ("C".equals(reportDetail.getReportCategory())) {
                // 게시글인 경우
                int commNo = aService.getCommNo(reportNo);
                if (commNo > 0) {
                    aService.deleteCommReport(commNo);
                }
            } else if ("R".equals(reportDetail.getReportCategory())) {
                // 댓글인 경우
                int replyNo = aService.getReplyNo(reportNo);
                if (replyNo > 0) {
                    aService.deleteReplyReport(replyNo);
                }
            }
        }
    }
    return result > 0 ? "success" : "fail";
}
```

모달창으로 신고 정보

```
// 전역 변수로 현재 보고 있는 콘텐츠 정보 저장
let currentContentInfo = {
  category: '',
  contentNo: 0,
  commNo: 0,
  reportNo: 0
};

// 신고된 게시글/댓글 보기 함수
function viewReportedContent(category, contentNo, reportNo) {
  const contentDetails = document.getElementById("contentDetails");
  contentDetails.innerHTML = "<p>로딩 중...</p>";

  const modal = document.getElementById("viewContentModal");
  modal.style.display = "block";
  // 현재 콘텐츠 정보 저장
  currentContentInfo.category = category;
  currentContentInfo.contentNo = contentNo;
  currentContentInfo.reportNo = reportNo;
  // contentNo가 유효한지 확인
  if (!contentNo || contentNo <= 0) {
    contentDetails.innerHTML = `<div class="error-message">유효한 콘텐츠 번호가 필요합니다. (전달된 값: ${contentNo})</div>`;
    document.getElementById("goToContentBtn").style.display = "none";
    return;
  }
  // 게시글/댓글 정보 가져오기
  $.ajax({
    url: '/report-viewer/content',
    data: {
      category: category,
      contentNo: contentNo
    },
    dataType: 'json',
    success: function(response) {
      console.log("콘텐츠 조회 응답:", response);

      // 에러 메시지가 있는 경우
      if (response.error) {
        contentDetails.innerHTML = `<div class="error-message">${response.error}</div>`;
        document.getElementById("goToContentBtn").style.display = "none";
        return;
      }
    }
  });
}
```

```
let html = '';
document.getElementById("goToContentBtn").style.display = "inline-block";

if (category === 'C') {
    // 게시글인 경우
    currentContentInfo.commNo = response.commNo;

    html = `
        <h3>게시글 정보</h3>
        <p><strong>제목:</strong> ${response.title || '제목 없음'}</p>
        <p><strong>작성자:</strong> ${response.name || '작성자 정보 없음'}</p>
        <p><strong>작성일:</strong> ${response.createDate || '날짜 정보 없음'}</p>
        <p><strong>내용:</strong></p>
        <div style="padding: 10px; border: 1px solid #ddd; border-radius: 4px; margin-bottom: 20px;">
            ${response.commContent || '내용 없음'}
        </div>
    `;
} else if (category === 'R') {
    // 댓글인 경우
    currentContentInfo.commNo = response.commNo;

    html = `
        <h3>댓글 정보</h3>
        <p><strong>작성자:</strong> ${response.name || '작성자 정보 없음'}</p>
        <p><strong>작성일:</strong> ${response.createDate || '날짜 정보 없음'}</p>
        <p><strong>내용:</strong></p>
        <div style="padding: 10px; border: 1px solid #ddd; border-radius: 4px; margin-bottom: 20px;">
            ${response.replyContent || '내용 없음'}
        </div>
        <p><strong>원본 게시글:</strong> ${response.commTitle || '게시글 정보 없음'}</p>
    `;
}

contentDetails.innerHTML = html;
```

작품 시연

Q&A