

2024년도 캡스톤 디자인

# 프로젝트 개발 보고서

1인 여행자 맞춤 테마 여행 안내  
(원데이 서울)

2024 년 11 월

신한대학교 소프트웨어융합학과





본 보고서를 “1인 여행자 맞춤 테마 여행 안내(원데이 서울)” 에 관한  
최종 개발보고서로 제출합니다.

프로젝트 제목	국 문: 1인 여행자 맞춤 테마 여행 안내 (원데이 서울)			
	영 문: A tour guide with a theme tailored to one person (OneDay Seoul)			
개 발 기 간	2024년 03월 02일 ~ 2024년 11월 10일까지 ( 9 개월)			
개발 참여자	구 분	소 속	학 번	성 명
	책임개발자	소프트웨어융합학과	20190854	장진원
	공동개발자	소프트웨어융합학과	20190814	김현호
		소프트웨어융합학과	20190834	오석민
		소프트웨어융합학과	20190831	신지섭

2024 년 11 월 10 일

신한대학교 소프트웨어융합학과

## 제정 및 개정 이력

버전	개정 내용	작성자	승인자	개정날짜
1.0	초안 작성	김현호	김현호	2024.10.28
1.1	수정	오석민	오석민	2024.10.29



I . 프로젝트 요약문 .....	P.5
II . 프로젝트 계획서 .....	P.23
III . 시스템 분석서 .....	P.39
IV . 요구사항 명세서 .....	P.73
V . 시스템 설계서 .....	P.90
VI . 시스템 테스트 결과서 .....	P.119
VII . 시스템 결과서 .....	P.125
VIII . 개발 후기 .....	P.130

# I. 프로젝트 요약문

## I. 프로젝트 배경 및 목적

---

### 1.1 시스템 개발의 배경 및 필요성

현재 사회는 1인 가구의 증가로 인해 많은 1인 가구 중심의 사업이 증가하고 발전해 가고 있습니다. 1인 가구는 더 많은 사람이 될 것이고 더욱더 혼자 활동하는 것이 익숙한 시대가 올 것입니다.

여행 관련 산업은 이미 많은 존재하지만 1인을 대상으로 여행을 추천해주는 사업은 많은 1인 여행객들에게 더욱 편리하고 고민을 덜어줄 선택지가 될 것으로 생각합니다.

1인 여행자 맞춤 테마 여행 안내 (원데이 서울)은 1인 여행객들에게 자신의 성향과 맞는 스타일을 지정해주고 그 성향의 맞는 여행 코스를 추려줍니다. 시스템을 개발하여 사용자가 원하는 스타일의 코스를 추천하여 여행지 선택에 참고하거나 선택할 수 있게 만드는 것을 목표로 하였습니다.

#### 현행 비즈니스 상황

1. 코로나 엔데믹 이후로 여행의 관심도는 꾸준히 증가하고 있다.
2. 혼자 여행 가는 것에 대한 관심도가 증가하고 있는데 모든 것을 혼자 계획하기에는 비용적인 문제와 여행지에서 무엇이 있는지 잘 모르기 때문에 여행 가기 전에 계획을 짜기에 많은 시간이 소요된다.

#### 현행 시스템의 문제점

##### 1. 획일화 되어있는 코스:

기존에 있는 여행 코스 추천은 이미 정해져 있는 코스를 사용자에게 제안할 뿐 개인의 선호도에 따른 여행 코스를 추천해 주지 않아 만족도가 떨어진다.

##### 2. UI가 불편성:

기존에 있는 여행 코스 추천은 여행을 갈 때마다 새로운 필터링 작업을 걸치는 불편이 있다.

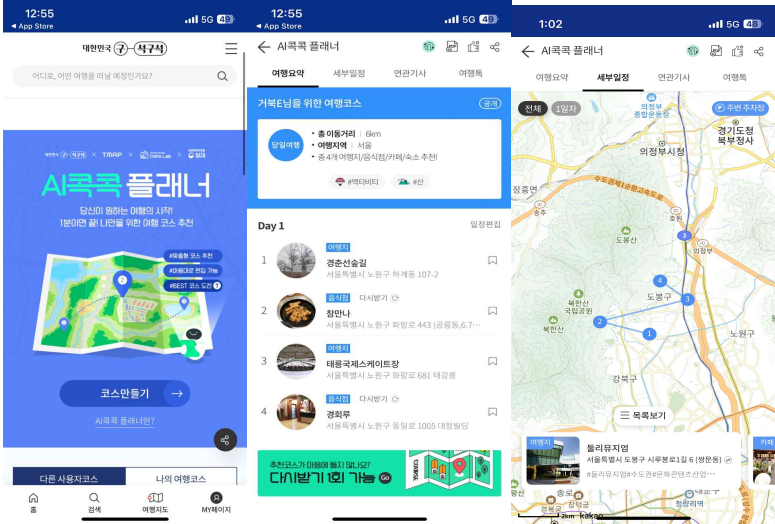
##### 3. 필터링의 부정확성:

사용자가 자신이 원하는 테마를 설정해도 여행지 추천이 테마에 맞지 않게 나오는 경우가 많아 문제가 있다.

##### 4. 교통수단 안내의 부족:

기존에 있는 여행 코스 추천은 여행지의 장소와 거리만 보여주기 때문에 어떻게 가야 하는지 알려주지 않음.

1.2 시스템 관련 동향 및 현황

이름	대한민국 구-석구석
사진	
특징	<ol style="list-style-type: none"><li>1. 운영 기간이 길어서 관광지에 대한 데이터가 많은 앱이다.</li><li>2. 관광지의 행사 및 축제에 대한 정보를 제공한다.</li><li>3. 미리 계획된 코스가 많다.</li></ol>
기능	<ol style="list-style-type: none"><li>1. 오늘의 시콕콕 추천으로 마음에 들 만한 관광지들을 소개 시켜준다.</li><li>2. 공연 및 행사의 정보들을 쉽게 찾을 수 있다.</li><li>3. 지역별 핫 플레이스를 소개 시켜준다.</li><li>4. 시콕콕 플래너로 여행 계획을 짜준다.</li><li>5. 테마별 추천이 있다.</li><li>6. 여행상품 홍보관이 있어 선택할 수 있는 계획이 많다.</li></ol>

이름	트리플
사진	 The screenshot shows the Triple app interface for planning a trip to Seoul. At the top, there's a header with the user's name '오석민님' and a search bar. Below the header, there are tabs for '여행, 어디로 떠나시나요?' and '국내도시'. The main content area is divided into three columns. The left column lists various cities and regions with circular icons and '선택' (Select) buttons. The middle column features a large promotional banner for '다낭 바나 힐 + 케이블카' with a '특가 보기' (View Special Price) button, and a section titled '트리플 매거진' with a '더보기' (More) button. The right column displays a list of travel-related items, each with a thumbnail image, title, and location, such as '서울 시립 북서울 미술관' and '서울 세계 불꽃 축제'. The bottom navigation bar includes icons for '국내여행', a central home icon, and '해외여행'.
특징	<div>1. UI 가독성이 좋다.</div> <div>2. 숙박과 항공편 정보 제공이 뛰어나다.</div> <div>3. 타인의 여행 기록을 통해 정보를 제공 받을 수 있다.</div>
기능	<div>1. 일정 관리가 쉽고, 날씨나 환율을 확인하기 편리하다.</div> <div>2. 일정 공유가 편리하다.</div> <div>3. 대표 관광지와 숙소에 대한 정보가 잘 정리되어 있다.</div> <div>4. 숙박과 항공편에 비교하기 수월하다.</div> <div>5. 빅데이터와 AI를 기반으로 사용자의 취향을 분석해, 여행 콘텐츠와 상품을 개인화해 제공할 수 있다.</div>



시스템 이름	대한민국 구-석구석	트리플
장점	<ol style="list-style-type: none"> <li>1. 운영 기간이 길어 데이터가 많다.</li> <li>2. 행사 일정이 알아보기 쉽다.</li> <li>3. 고를 수 있는 테마가 많다.</li> </ol>	<ol style="list-style-type: none"> <li>1. 위치에 기반해서 가까운 음식점 등을 추천해 주고, 갑자기 비가 와도 가볼 만한 곳을 알려준다.</li> <li>2. 여행 동행자들과 일정을 공유하여 계획을 세우기 편리하다.</li> </ol>
단점	<ol style="list-style-type: none"> <li>1. 인원수 설정이 없어 몇 명이 여행가는지 모른다.</li> <li>2. 웹 기반 앱이라 사용이 불편하다.</li> <li>3. 코스 추천이 메인이라 찾기 불편하다.</li> <li>4. 교통수단이 나와 있지 않다.</li> </ol>	<ol style="list-style-type: none"> <li>1. 사용자 본인이 일정을 스스로 등록해야 한다.</li> <li>2. 서울에 대한 필터링이 구체적이지 않다.</li> <li>3. 여행 스타일을 선택해도 사용자 본인이 원하는 스타일과 다르게 설정될 수 있다.</li> <li>4. 사용자의 취향을 분석해도 제대로 된 여행 경로를 추천해주지 않는다.</li> </ol>

### 1.3 시스템 개발 목적 및 목표

시스템의 목적 :

사용자가 회원가입시 처음에 사는 지역구 조사, 개인 맞춤형테마를 통해 여행 계획을 만들어 주고, 추후 새로운 설정 또는 필터링 작업을 거치지 않고 여행 계획을 만들어 준다. 그리고 여행 계획코스를 한 페이지 안에 지도와 설명, 이동수단 등을 간략하게 넣어 한눈에 알아보기 쉽게 디자인한다.

시스템의 목표 : 사용자에게 여행 계획을 만드는 피로도를 줄인다.

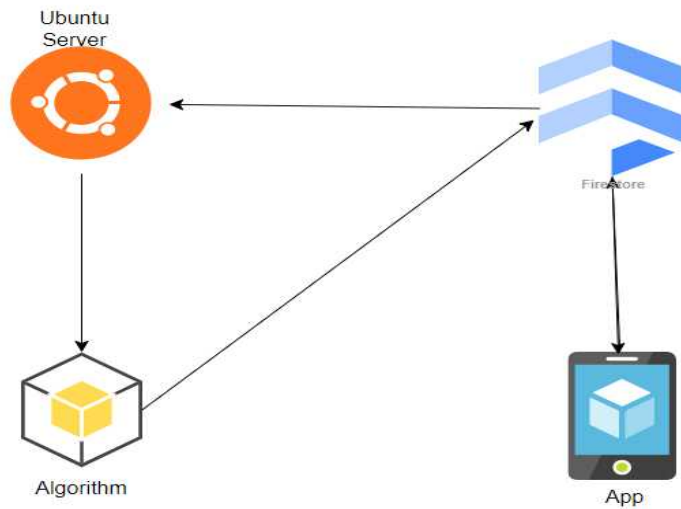
향후 시스템의 사용 대상 : 국내 20~30대 1인 여행자

### 1.4 시스템 기대효과 및 활용 방안

- 효율적인 여행 계획 : 자신의 스타일에 맞는 여행 코스를 추천해주기 때문에 보다 편리하게 여행 계획을 세울 수 있다.
- 새로운 장소 추천 : 자신이 몰랐던 새로운 여행지를 발견할 수 있다.

## II. 프로젝트 내용 및 범위

### 2.1 시스템 구성

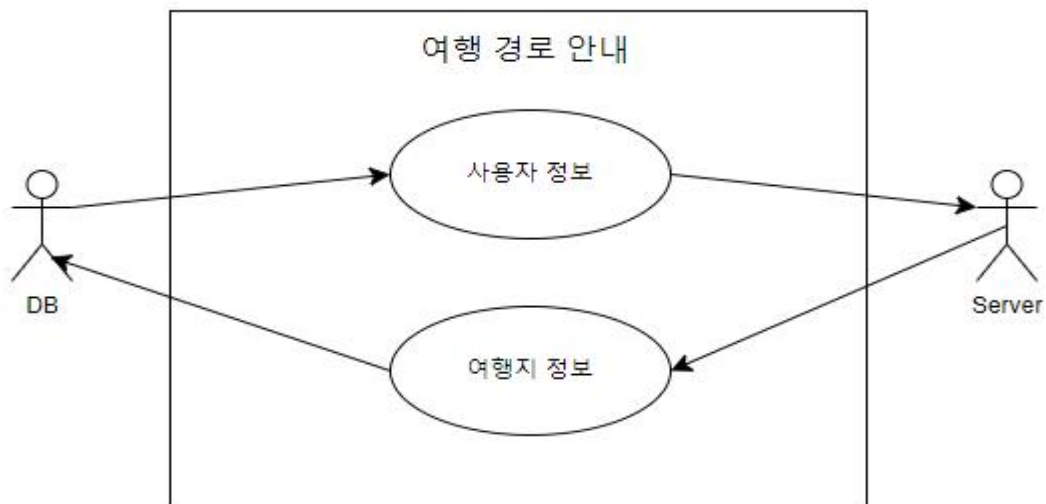
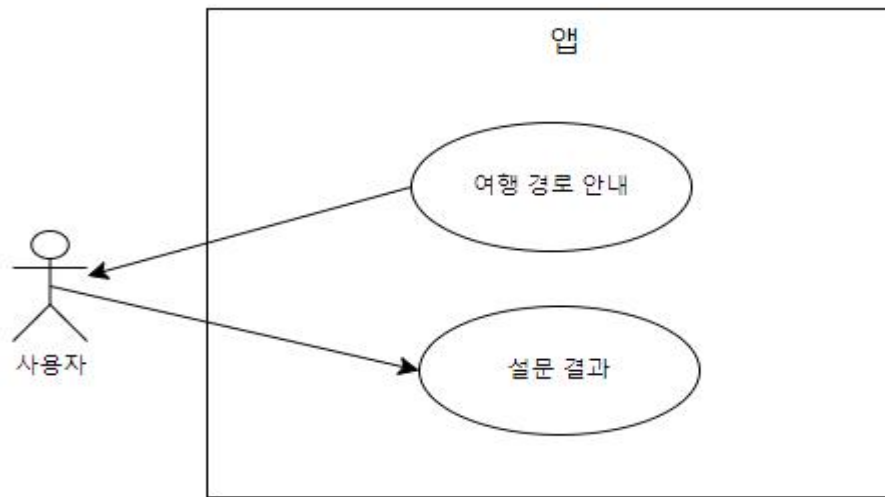


### 2.2 시스템 주요 기능

서브시스템	핵심 기능(Feature)	세부 기능(Function)
서버	실시간으로 데이터 전달	Firebase에 정보가 갱신될 때마다 알고리즘으로 값을 전달 시킨다.
사용자 데이터 관리시스템	사용자 데이터 관리	질문 저장
		사용자 데이터 저장
		여행 경로 저장
		여행지 피드백 정보 저장
경로 탐색 시스템	여행 코스를 만드는 알고리즘	사용자의 테마를 받아서 테마에 맞는 여행지를 추천하고 여행지의 최단 거리를 구한다.
앱	사용자 UI	로그인
		회원가입
		여행지의 코스를 보여준다
		질문을 보여준다

## 2.3 시스템 개발 범위

### 1) 시스템 개발 범위



### 2) 시스템 제외 범위

- 외부 API : Naver Login API, Google Login API, Google Map API, Google Directions API, Google Places API, Cloud Firebase API

### III. 시스템 개발 환경 및 기술

#### 3.1 시스템 개발 환경

언어	버전	사용처
Dart	Dart 3.6.0-277.0.dev	Android 개발
Python	Python 3.10.12	알고리즘 학습, 서버 개발

S/W	버전	사용처
Windows	Windows11	서버 개발, Android 개발, 학습
Firebase	5.4.0	DB 구축
Flutter	Flutter 3.26.0-1.0.pre.204	Android 개발
Ubuntu	Ubuntu 24.04.1 LTS	서버 개발

#### 3.2 시스템 개발 기술

- 1) TensorFlow 2.0 : 학습을 위한 딥러닝 프레임 워크
- 2) supernova : 디자인한 앱 화면을 dart 코드로변환.
- 3) 플러터 : android 개발이 가능한 프레임워크, dart 언어 사용
- 4) Google Colab : 구글에서 제공해주는 클라우드 기반 Jupyter Notebook 환경, 알고리즘 학습 시 사용
- 5) Firebase : 유저의 데이터를 저장하기 위해서 사용
- 6) AWS : 알고리즘, Firebase을 연동시켜 실시간으로 데이터 교환을 위해 사용

### III. 시스템 개발 환경 및 기술

#### 3.3 시스템 개발 인력 및 제약사항

시스템 개발 제약사항 : 애플리케이션 제작 시 최적화를 통해 기기의 성능제약에 영향을 최소화하게 제작한다.

사용자경험 요구사항 : 사용자들은 직관적이고 편리한 앱을 원한다. 사용자경험을 위해 최소한의 선택지로 필요한 서비스를 쉽게 이용할 수 있도록 해야 한다. 사용자에게 데이터를 제공받는 시간을 최소화해 불필요한 시간 사용을 줄인다.

성능 요구사항 : 사용자들이 원활하게 여행 코스에 대한 정보를 추천받을 수 있도록 성능 요구사항을 고려해야 한다. 대용량 데이터 처리, 캐싱 등의 기술을 활용하여 성능을 향상할 수 있다.

데이터 분류를 명확히 하여 학습을 시킨 후 학습된 데이터의 정확성과 불러오는 시간을 빠르게 해서 성능을 향상할 수 있다.

보안 요구사항 : 사용자의 데이터를 보관 관리하는 것이기 때문에 데이터를 저장하는 Firebase의 보안에 신경 써야 한다. 네이버 맵의 경우 요금제가 있는 api이기에 api키 암호화에 신경 쓴다.

법적, 표준, 조직의 제약사항 : 소셜 로그인 api를 사용하기 위해 어떤 목적으로, 어떻게 사용할 건지 동의를 받는다. 개인정보 수집 · 이용 동의서를 받는 절차를 포함한다.

## IV. 시스템 개발 결과

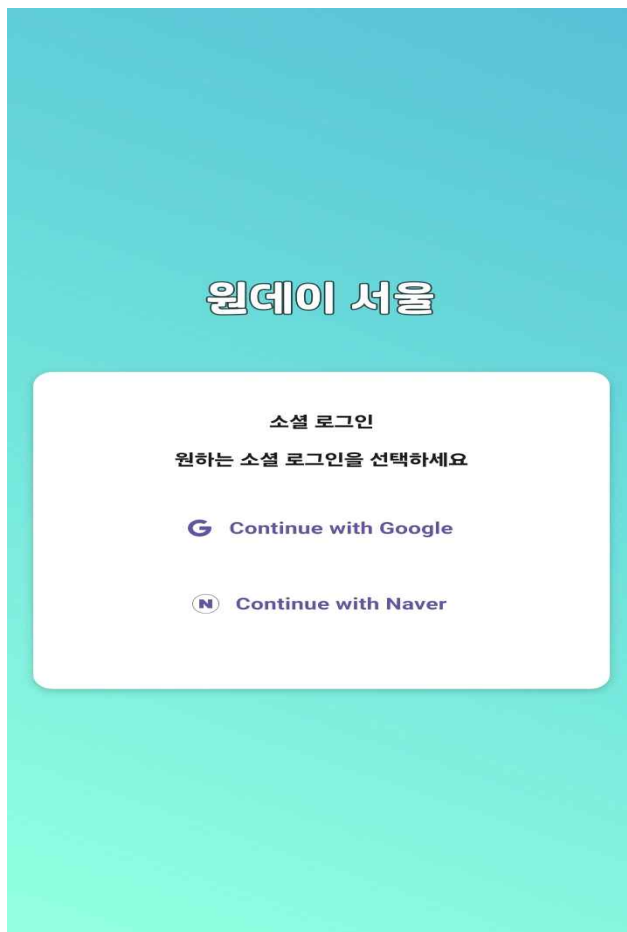
---

### 4.1 시스템 최종 개발 내용

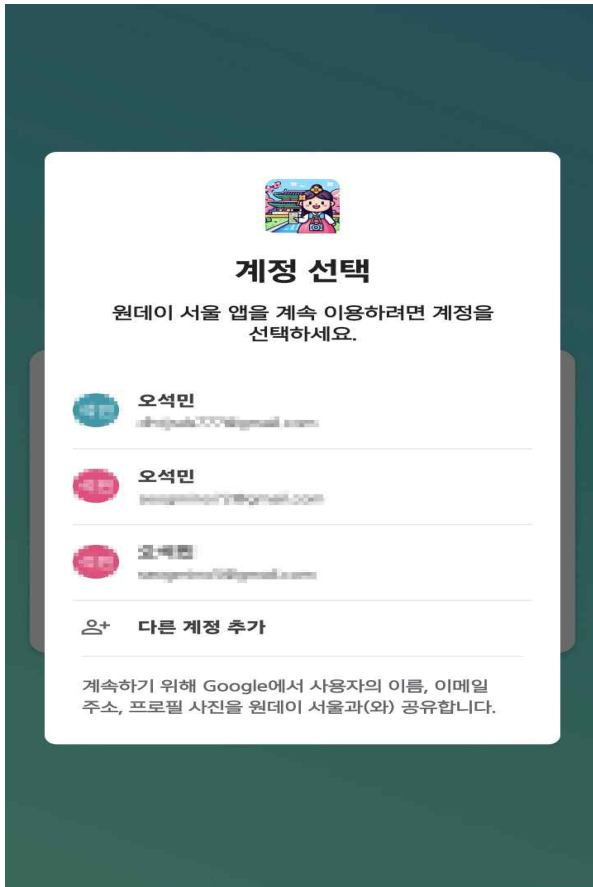
해당 서비스는 20대의 혼자 여행을 다니는 사람을 위해 만들어진 서비스로 현행 서비스와는 달리 기존에 있는 획일화된 코스가 아닌 개인의 선호도에 따른 여행 계획을 만들어 준다. 그리고 다음에 계획을 만들 때 새롭게 필터링, 설정하지 않고 여행 계획을 한눈에 보기 쉽게 만들었다.

### 4.2 시스템 개발 결과

#### 1. 로그인 화면



## 2. 계정 선택 화면



## 3. 설문 화면

### travel test

Question 1/45

관광지 조사..

1(매우 아니다), 2(아니다), 3(보통), 4(그렇다), 5(매우 그렇다)

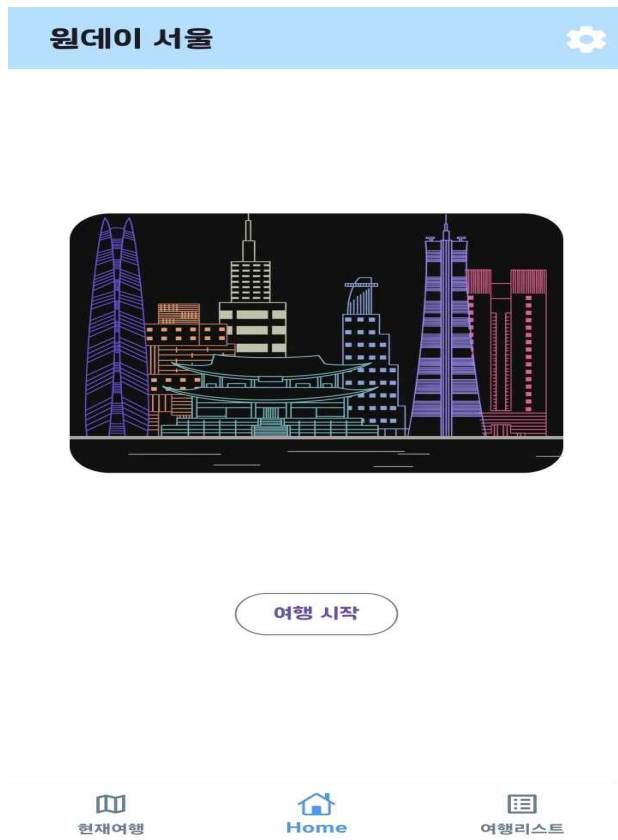
여행 중 현지 공연(연극, 콘서트 등)을 즐기고 싶습니까?

1 2 3 4 5

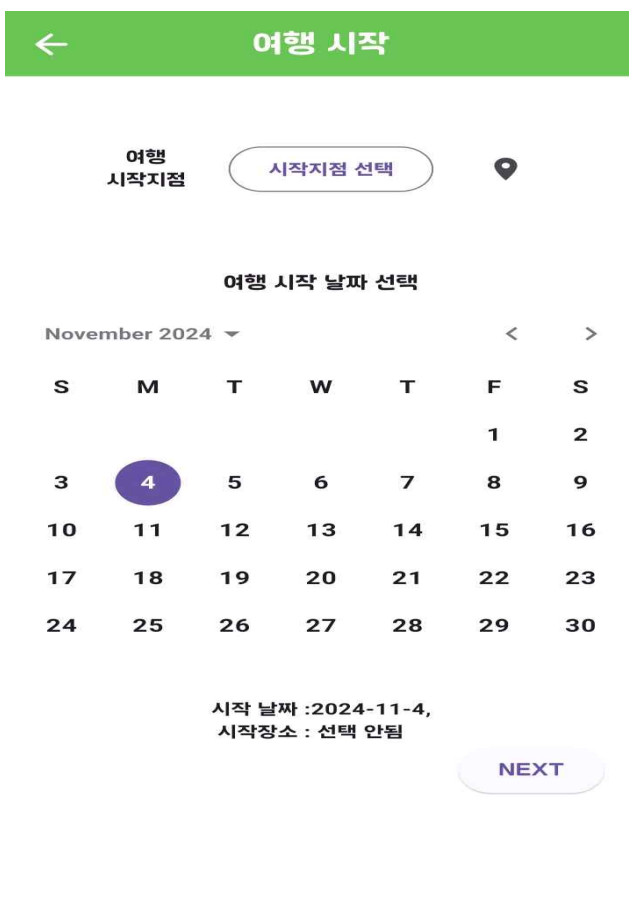
Next Question




#### 4. 메인 화면



#### 5. 여행 시작 화면(1)



## 6. 여행 시작 화면(2)

 여행 시작

가고싶은 지역구

없음

여행 이름 지정

여행 이름을 지어주세요

여행 시작

## 7. 여행 리스트 화면

SKT 5:54 79%  
원데이 서울



2024-11-27-여행  
Date: 2024년 11월 27일  
City: 충주





566  
Date: 2024년 11월 28일  
City: 동대문구

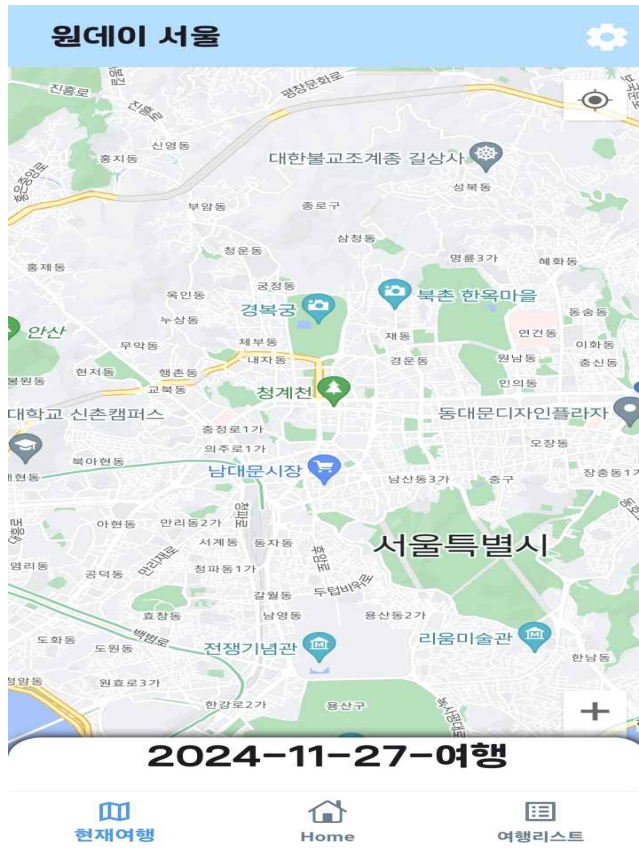


원재여행

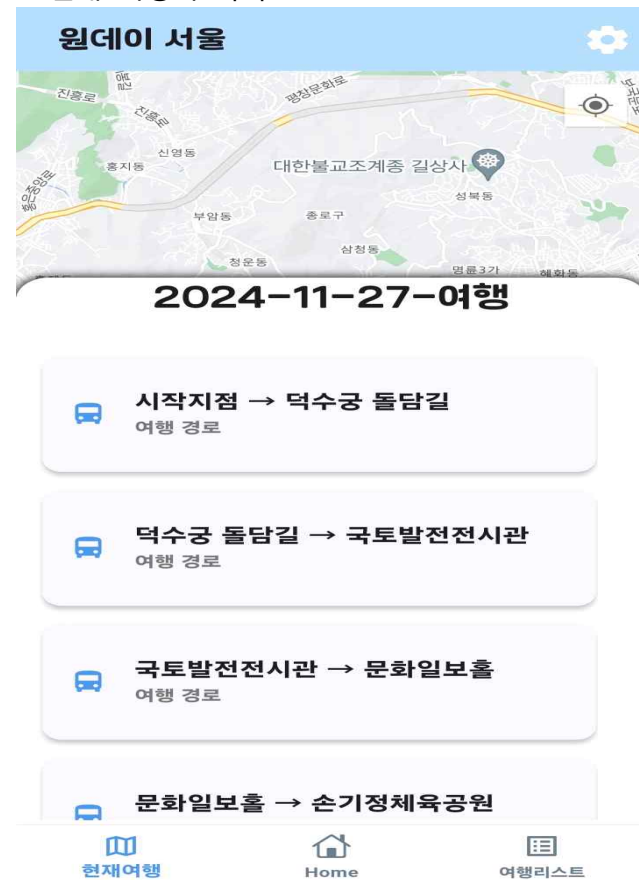
Home

여행리스트

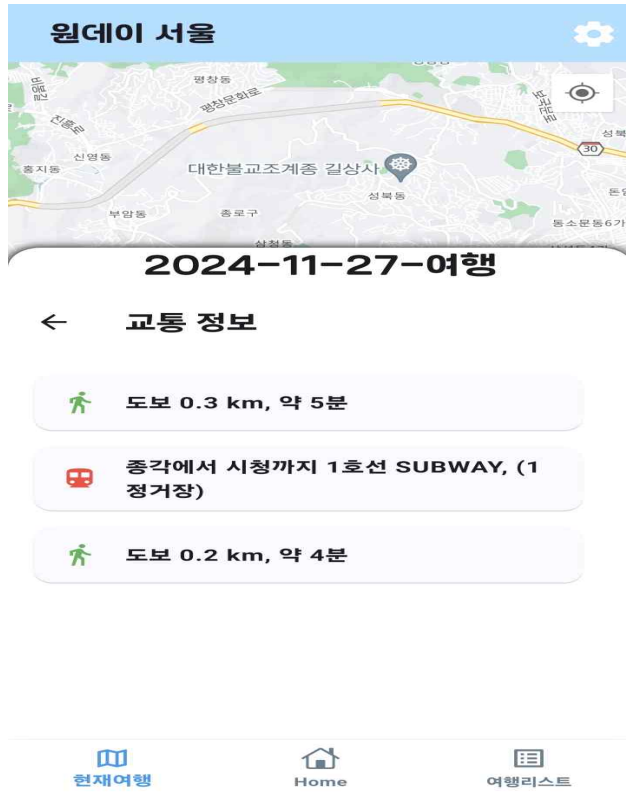
## 8. 현재 여행 화면



## 9. 현재 여행지 목록



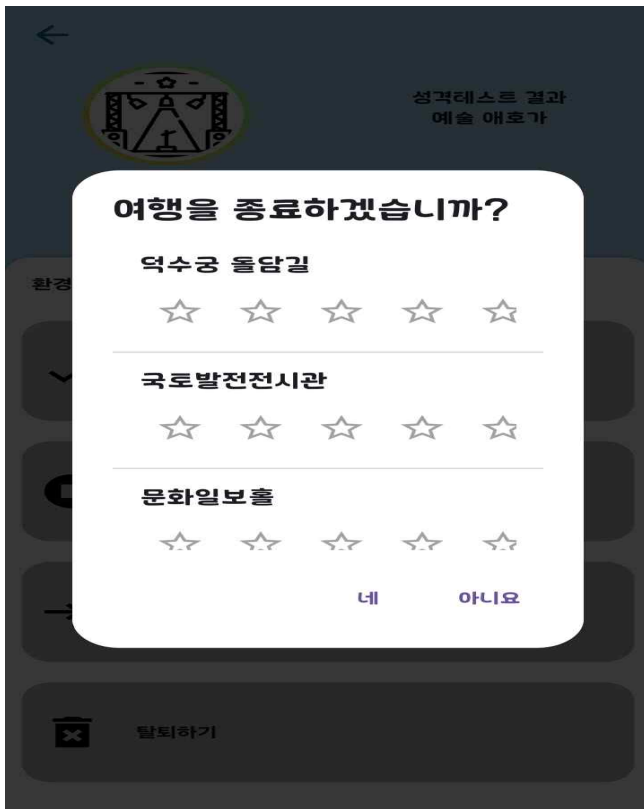
## 10. 여행 경로



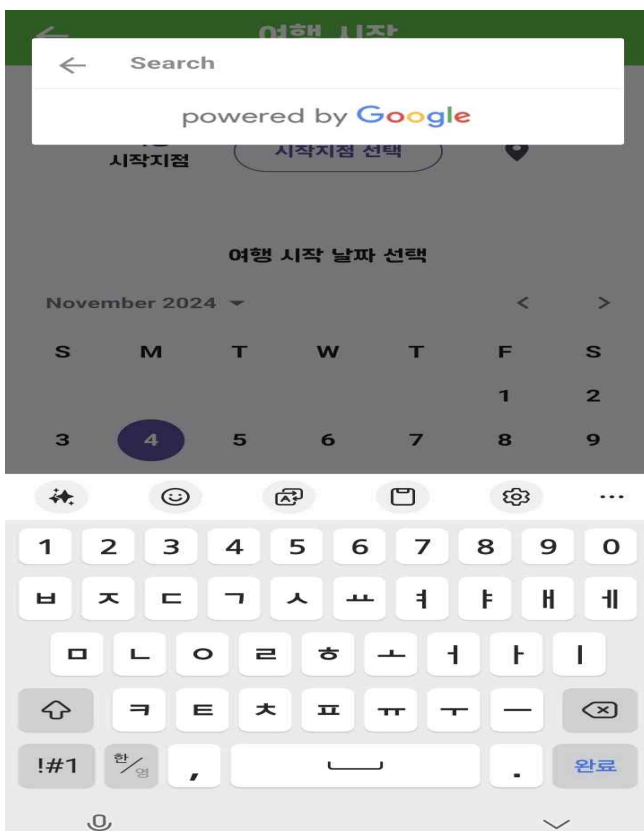
## 11. 설정 화면



## 12. 여행 종료 화면



## 1.13 시작 지점 선택



### 1.14 로딩 중 화면



## II. 프로젝트 계획서

## I. 프로젝트 개요

### 1.1 개발 배경 및 필요성

현재 사회는 1인 가구의 증가로 인해 많은 1인 가구 중심의 사업이 증가하고 발전해 가고 있습니다. 1인 가구는 더 많은 사람이 될 것이고 더욱더 혼자 활동하는 것이 익숙한 시대가 올 것입니다.

여행 관련 산업은 이미 많은 존재하지만 1인을 대상으로 여행을 추천해 주는 사업은 많은 1인 여행객들에게 더욱 편리하고 고민을 덜어줄 선택지가 될 것으로 생각합니다.

1인 여행자 맞춤 테마 여행 안내(원데이 서울)은 1인 여행객들에게 자신의 성향과 맞는 스타일을 지정해 주고 그 성향의 맞는 여행 코스를 추천합니다. 시스템을 개발하여 사용자가 원하는 스타일의 코스를 추천하여 여행지 선택에 참고하거나 선택할 수 있게 만드는 것을 목표로 하였습니다.

#### 현행 비즈니스 상황

1. 코로나 엔데믹 이후로 여행의 관심도는 꾸준히 증가하고 있다.
2. 혼자 여행 가는 것에 대한 관심도가 증가하고 있는데 모든 것을 혼자 계획하기에는 비용적인 문제와 여행지에서 무엇이 있는지 잘 모르기 때문에 여행 가기 전에 계획을 짜기에 많은 시간이 소요된다.

#### 현행 시스템의 문제점

1. 획일화 되어 있는 코스:

기존에 있는 여행 코스 추천은 이미 정해져 있는 코스를 사용자에게 제안할 뿐 개인의 선호도에 따른 여행 코스를 추천해 주지 않아 만족도가 떨어진다.

2. UI가 불편성:

기존에 있는 여행 코스 추천은 여행을 갈 때마다 새로운 필터링 작업을 걸치는 불편이 있다.

3. 필터링의 부정확성:

사용자가 자신이 원하는 선호도를 설정해도 여행지 추천이 선호도에 맞지 않게 나오는 경우가 많아 문제가 있다.

4. 교통수단 안내의 부족:

기존에 있는 여행 코스 추천은 여행지의 장소와 거리만 보여주기 때문에 어떻게 가야 하는지 알려주지 않음.



## 1.2 개발 목적 및 목표

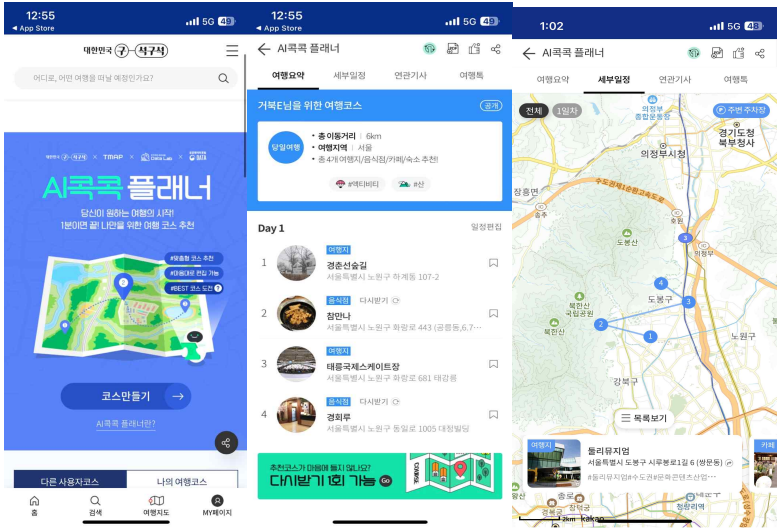
시스템의 목적 :

사용자가 회원가입 시 처음에 사는 지역구 조사, 개인 선호도를 통해 여행 계획을 만들어 주고, 추후 새로운 설정 또는 필터링 작업을 거치지 않고 여행 계획을 만들어 준다. 그리고 여행 계획코스를 한 페이지 안에 지도와 설명, 이동 수단 등을 간략하게 넣어 한눈에 알아보기 쉽게 디자인한다.

시스템의 목표 : 사용자에게 여행 계획을 만드는 피로도를 줄인다.

향후 시스템의 사용 대상 : 국내 20~30대 1인 여행자

1.3 관련 동향 및 현황

이름	대한민국 구-석구석
사진	
특징	<ol style="list-style-type: none"><li>1. 운영 기간이 길어서 관광지에 대한 데이터가 많은 앱이다.</li><li>2. 관광지의 행사 및 축제에 대한 정보를 제공한다.</li><li>3. 미리 계획된 코스가 많다.</li></ol>
기능	<ol style="list-style-type: none"><li>1. 오늘의 시콕콕 추천으로 마음에 들 만한 관광지들을 소개 시켜준다.</li><li>2. 공연 및 행사의 정보들을 쉽게 찾을 수 있다.</li><li>3. 지역별 핫 플레이스를 소개 시켜준다.</li><li>4. 시콕콕 플래너로 여행 계획을 짜준다.</li><li>5. 테마별 추천이 있다.</li><li>6. 여행상품 홍보관이 있어 선택할 수 있는 계획이 많다.</li></ol>

이름	트리플
사진	 The screenshot shows the Triple app interface for planning a trip to Seoul. At the top, there's a header with the user's name '오석민님' and a search bar. Below the header, there are tabs for '여행, 어디로 떠나시나요?' and '국내도시'. The main content area is divided into three columns. The left column lists various cities with circular icons and '선택' (Select) buttons. The middle column features a search bar, a '항공편' (Flight) button, a '숙소' (Accommodation) button, a '주요 티켓' (Main Ticket) button, and a '일정 만들기' (Create Itinerary) button. The right column displays a list of travel-related items, including '서울 시립 북서울 미술관', '서울 세계 불꽃 축제', '국제 갤러리 서울 점', '서울 빛 초롱 축제', '렛츠 런 파크 서울', '서울 공연 예술 고등학교', '서울 악령시 한의약 박물관', '신라 면세점 서울 점', '홍익 대학교 서울 캠퍼스', '서울' (Seoul), '대한 성공회 서울 주교좌 성당', and '서울 왕궁 수문장 교대 의식'. At the bottom, there are icons for '국내여행' (Domestic Travel), '해외여행' (Overseas Travel), and a globe icon.
특징	<ol style="list-style-type: none"><li>1. UI 가독성이 좋다.</li><li>2. 숙박과 항공편 정보 제공이 뛰어나다.</li><li>3. 타인의 여행 기록을 통해 정보를 제공 받을 수 있다.</li></ol>
기능	<ol style="list-style-type: none"><li>1. 일정 관리가 쉽고, 날씨나 환율을 확인하기 편리하다.</li><li>2. 일정 공유가 편리하다.</li><li>3. 대표 관광지와 숙소에 대한 정보가 잘 정리되어 있다.</li><li>4. 숙박과 항공편에 비교하기 수월하다.</li><li>5. 빅데이터와 AI를 기반으로 사용자의 취향을 분석해, 여행 콘텐츠와 상품을 개인화해 제공할 수 있다.</li></ol>

시스템 이름	대한민국 구-석구석	트리플
장점	<ol style="list-style-type: none"> <li>1. 운영 기간이 길어 데이터가 많다.</li> <li>2. 행사 일정이 알아보기 쉽다.</li> <li>3. 고를 수 있는 테마가 많다.</li> </ol>	<ol style="list-style-type: none"> <li>1. 위치에 기반해서 가까운 음식점 등을 추천해 주고, 갑자기 비가 와도 가볼 만한 곳을 알려준다.</li> <li>2. 여행 동행자들과 일정을 공유하여 계획을 세우기 편리하다.</li> </ol>
단점	<ol style="list-style-type: none"> <li>1. 인원수 설정이 없어 몇 명이 여행가는지 모른다.</li> <li>2. 웹 기반 앱이라 사용이 불편하다.</li> <li>3. 코스 추천이 메인이 아니라 찾기 불편하다.</li> <li>4. 교통수단이 나와 있지 않다.</li> </ol>	<ol style="list-style-type: none"> <li>1. 사용자 본인이 일정을 스스로 등록해야 한다.</li> <li>2. 서울에 대한 필터링이 구체적이지 않다.</li> <li>3. 여행 스타일을 선택해도 사용자 본인이 원하는 스타일과 다르게 설정될 수 있다.</li> <li>4. 사용자의 취향을 분석해도 제대로 된 여행 경로를 추천해주지 않는다.</li> </ol>

#### 1.4 기대효과 및 활용 방안

- 효율적인 여행 계획 : 자신의 선호도에 맞는 여행 코스를 추천해주기 때문에 보다 편리하게 여행 계획을 세울 수 있다.
- 새로운 장소 추천 : 자신이 몰랐던 새로운 여행지를 발견할 수 있다.

## II. 시스템 개요 및 특징

### 2.1 시스템 개요

#### (1) 시스템 이름 및 개요

##### (1-1) 시스템 이름

1인 여행자 맞춤 테마 여행 안내(원데이 서울)

##### (1-2) 시스템 개요

사용자에게는 현재 위치, 선호도, 가고 싶은 지역을 입력받아 최적의 동선으로 계획을 만들어 준다.

#### (2) 시스템 특성

혼자 여행을 다니는 국내 20~30대 1인 여행자를 위해 만든 앱으로, 기존에 있는 획일화된 코스가 아닌 개인의 선호도에 따른 여행 계획을 만들어 준다. 그리고 다음에 계획을 만들 때 새롭게 필터링, 설정하지 않고 여행 계획을 지도에 한눈에 보기 쉽게 만들어준다.

#### (3) 사용자 특성

국내 20~30대 1인 여행자

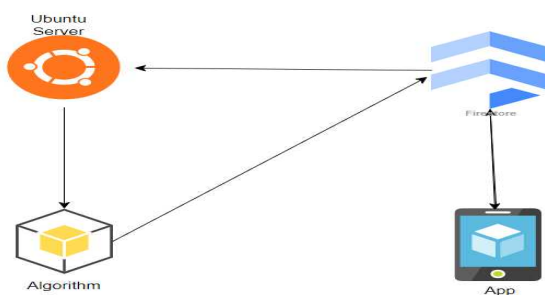
#### (4) 시스템 운영 환경

- 시스템 운영 개념과 환경을 기술

앱 : 모바일 기기에서 쉽게 사용할 수 있다.

DB : 질문에 대한 정보, 추천된 여행지, 사용자 정보를 저장하기 위해 사용한다.

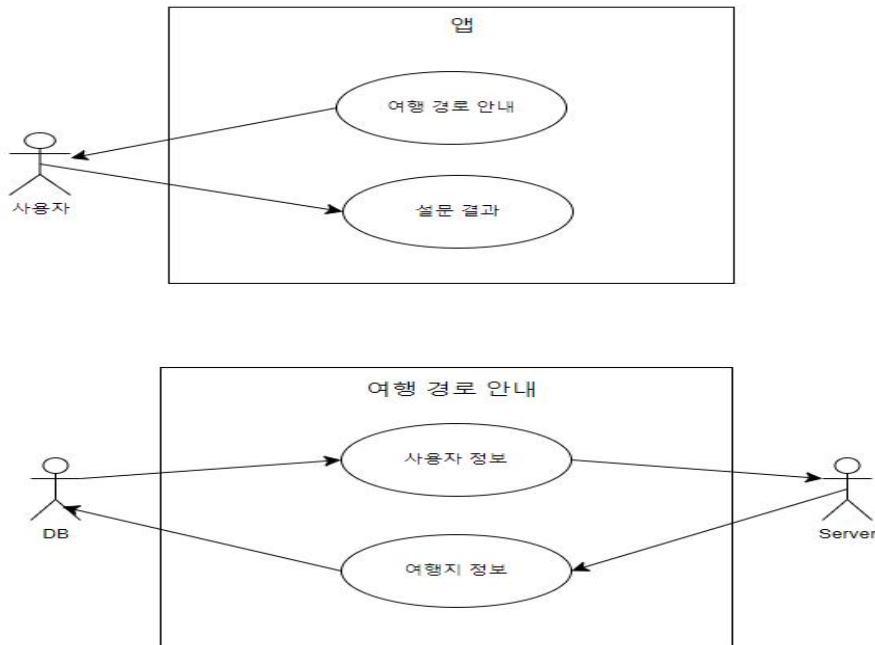
### 2.2 시스템 구성도



## II. 시스템 개요 및 특징

### 2.3 개발 범위

#### (1) 시스템 개발 범위



#### (2) 시스템 제외 범위

- 외부 API : Naver Login API, Google Login API, Google Map API, Google Directions API, Google Places API, Cloud Firebase API

## 2.4 주요 기능

서브시스템	핵심 기능(Feature)	세부 기능(Function)
서버	실시간으로 데이터 전달	Firebase에 정보가 갱신될 때마다 알고리즘으로 값을 전달 시킨다.
사용자 데이터 관리시스템	사용자 데이터 관리	질문 저장
		사용자 데이터 저장
		여행 경로 저장
		여행지 피드백 정보 저장
경로 탐색 시스템	여행 코스를 만드는 알고리즘	사용자의 테마를 받아서 테마에 맞는 여행지를 추천하고 여행지의 최단 거리를 구한다.
앱	사용자 UI	로그인
		회원가입
		여행지의 코스를 보여준다.
		질문을 보여준다.

## III. 시스템 개발 관리

---

### 3.1 추진 전략

#### (1) 기술적 추진 전략

각 관광지의 정보를 불러와 카테고리 분류, 지역 분류 등을 실시한다.

GitHub : 앱 개발, 학습 알고리즘에 필요한 Open Source를 사용한다.

관광지의 순번을 위도와 경도를 이용하여 최단 거리의 코스를 짜준다.

플러터를 활용한 개발로 android 애플리케이션 개발을 한다.

API나 Open Source SW 또는 다른 SW 등을 사용할 경우 해당 제품이나 기술에 관해 기술하고 적용 방법(적용, 개선 또는 참조) 등을 표현한다.

TensorFlow : 여행지의 추천을 위해 학습에 사용한다.

Ubuntu : DB에서 데이터가 갱신되면 알고리즘을 실행시키기 위해 사용한다.

Firebase : 사용자의 데이터와 설문 내용을 저장을 위해서 사용한다.



## (2) 관리적 추진 전략

개발 과정에서 발생 예상되는 문제점과 해결 방안 등을 기술

1. 관광지에 대한 정보를 분류하여 추천할 때 표본이 적어 내용이 정확하지 않을 수 있다.
2. 설문을 통해 개인 선호도를 분류해야 하는 데 필요한 설문 내용이 정확하지 않을 수 있다.

프로젝트를 추진하는 데 필요한 기술이나 정보를 획득하는 방법과 수행 방법을 기술

1. 관광 데이터 관리 : 관광지에 대한 정보를 분류하여 추천할 때 표본이 적어 내용이 정확하지 않을 수 있다.
2. DB 연결 : 학습시킨 관광지 정보와 앱 내의 DB 연동에 어려움을 해결하기 위해 DB 관련 지식을 학습하여 활용한다.
3. 반응형 앱 적용 : 상황별 전체 UI를 제작하는 데 있어서 UI의 일관성이 깨질 수 있다. 기종 별로 UI를 나누고 상세 코드를 부분별 구분하여 개발한다.
4. 보안적 요소 : NAVER map API가 요금이 드는 API인 만큼 API 키를 암호화 하는 등의 보안에 신경 쓴다.

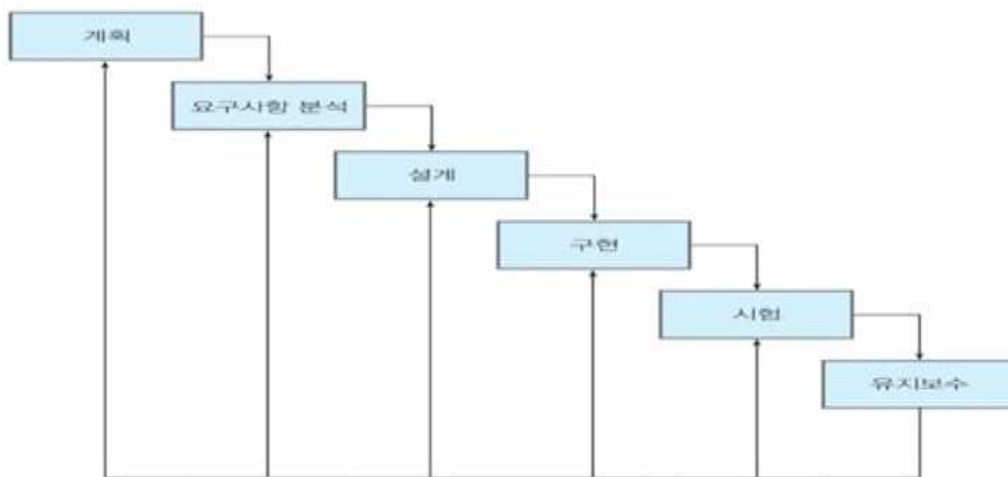
## 3.2 개발 기술

### (1) 적용 기술

- TensorFlow 2.0 : 학습을 위한 딥러닝 프레임 워크
- supernova : 디자인한 앱 화면을 dart 코드로 변환.
- 플러터 : IOS, android 동시개발이 가능한 프레임워크, dart 언어 사용
- Google Colab : 구글에서 제공하는 클라우드 기반 Jupyter Notebook 환경, 알고리즘 학습 시 사용
- Firebase : 사용자의 데이터와 설문 내용 데이터를 저장하기 위해서 사용
- AWS : 알고리즘, Firebase을 연동시켜 실시간으로 데이터 교환을 위해 사용

### (2) 개발 방법론

폭포수 모델 : 가장 기본적인 개발 프로세스 모형이고 초보자가 사용하기 쉽고 절차가 간결하고 이해하기 쉽고 요구사항을 확정시켜서 개발을 진행하기 때문에 폭포수 모델이 적합하다고 생각했기 때문에 폭포수 모델을 통해 개발을 진행하려 한다.



### III. 시스템 개발 관리

#### 3.3 개발 환경

언어	버전	사용처
Dart	Dart 3.6.0-277.0.dev	Android 개발
Python	Python 3.10.12	알고리즘 학습, 서버 개발

S/W	버전	사용처
Windows	Windows11	서버 개발, Android 개발, 학습
Firebase	5.4.0	DB 구축
Flutter	Flutter 3.26.0-1.0.pre.204	Android 개발
Ubuntu	Ubuntu 24.04.1 LTS	서버 개발

### 3.4 개발 제약사항

시스템 개발 제약사항 : 애플리케이션 제작 시 최적화를 통해 기기의 성능제약에 영향을 최소화하게 제작한다.

사용자경험 요구사항 : 사용자들은 직관적이고 편리한 앱을 원한다. 사용자경험을 위해 최소한의 선택지로 필요한 서비스를 쉽게 이용할 수 있도록 해야 한다. 사용자에게 데이터를 제공받는 시간을 최소화해 불필요한 시간 사용을 줄인다.

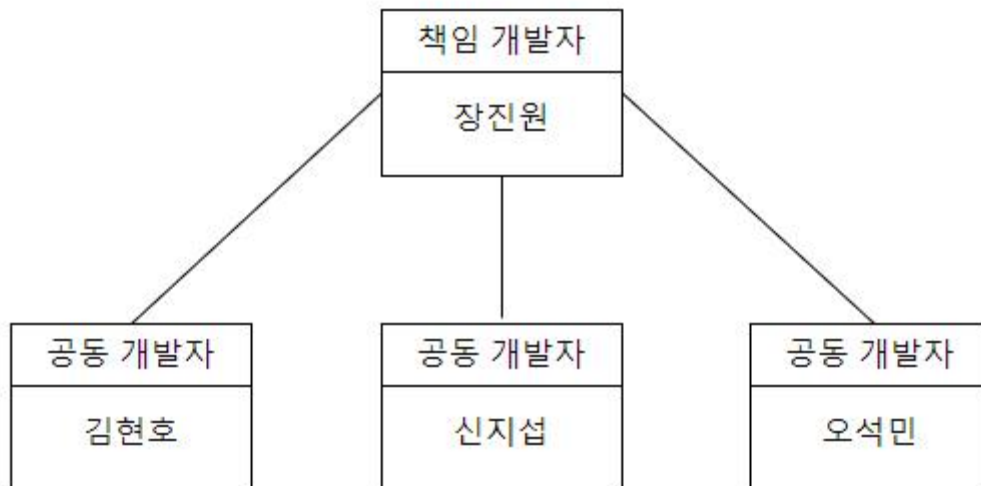
성능 요구사항 : 사용자들이 원활하게 여행 코스에 대한 정보를 추천받을 수 있도록 성능 요구사항을 고려해야 한다. 대용량 데이터 처리, 캐싱 등의 기술을 활용하여 성능을 향상할 수 있다.

데이터 분류를 명확히 하여 학습을 시킨 후 학습된 데이터의 정확성과 불러오는 시간을 빠르게 해서 성능을 향상할 수 있다.

보안 요구사항 : 사용자의 데이터를 보관 관리하는 것이기 때문에 데이터를 저장하는 Firebase의 보안에 신경 써야 한다. 네이버 맵의 경우 요금제가 있는 api이기에 api키 암호화에 신경 쓴다.

법적, 표준, 조직의 제약사항 : 소셜 로그인 api를 사용하기 위해 어떤 목적으로, 어떻게 사용할 건지 동의를 받는다. 개인정보 수집 · 이용 동의서를 받는 절차를 포함한다.

### 3.5 개발 조직



구분	이름	담당 업무
책임 개발자	장진원	알고리즘 작성 및 학습
공동 개발자	김현호	알고리즘 작성 및 학습, 서버 관리 및 연동
공동 개발자	신지섭	앱, DB 설계
공동 개발자	오석민	DB 설계, 서버 연동

## 3.6 개발 일정

분류	개발 내용	담당자	개 발 기 간 (2024.03.02 ~ 2024.10.31)							
			3(N)	4(N+1)	5(N+2)	6(N+3)	7(N+4)	8(N+5)	9(N+6)	10(N+7)
1. 시스템 기획		공동작업								
1.1	제안서작성									
1.2	계획서작성									
2. 요구분석		공동작업								
2.1	요구사항추출									
2.1.1	브레인스토밍									
2.1.2	관련 동향 및 현황 파악									
2.2	시스템분석									
2.3	요구사항명세서									
3. 시스템 설계										
3.1	앱 개발	신지섭								
3.1.1	앱 UI 설계									
3.1.2	현재 여행 설계									
3.1.3	여행 리스트 설계									
3.2	알고리즘 설계	김현호/장진원								
3.2.1	학습 알고리즘 설계									
3.2.2	추천 알고리즘 설계									
3.3	DB 설계	신지섭/오석민								
3.4	서버 구축 설계	김현호/오석민								
4. 개발										
4.1	앱 개발	신지섭								
4.1.1	앱 UI 구현									
4.1.2	내부 기능 구현									
4.2	DB 구현	신지섭/오석민								
4.3	알고리즘 구현	김현호/장진원								
4.3.1	학습 알고리즘 설계									
4.3.2	추천 알고리즘 설계									
4.4	서버 구축	김현호/오석민								
4.5	테스트 및 수정	공동작업								
4.6	보고서 작성									

## III. 시스템 분석서

## I. 개 요 (Introduction)

---

- 유스케이스 기술서는 구축할 시스템과 사용자와의 상호작용을 파악하여 개발할 기능을 정의한다. 즉, 고객의 요구사항을, 사용자의 관점에서, 시스템이 제공하는 서비스명으로 기술하는 것이 핵심이다.
- 시스템의 기능적, 비기능적 요구사항 중에서 기능적 요구사항을 대상으로 작성한다. 사용자 입장에서 시스템을 보았을 때, 시스템으로부터 어떤 서비스를 제공받기를 원하는지 정리한다.
- 이와 같은 방식을 통해 시스템이 제공해야 하는 기능을 분석한다. 즉, 시스템의 내부 기능의 상세한 작동 메커니즘을 기술하는 것이 아니고, 외부(=사용자)에 보여지는 모습(결과)을 중심으로 기술한다.
- 사용자와 기능 사이의 연관관계를 유스케이스 다이어그램과 유스케이스 정의서로 작성한다.

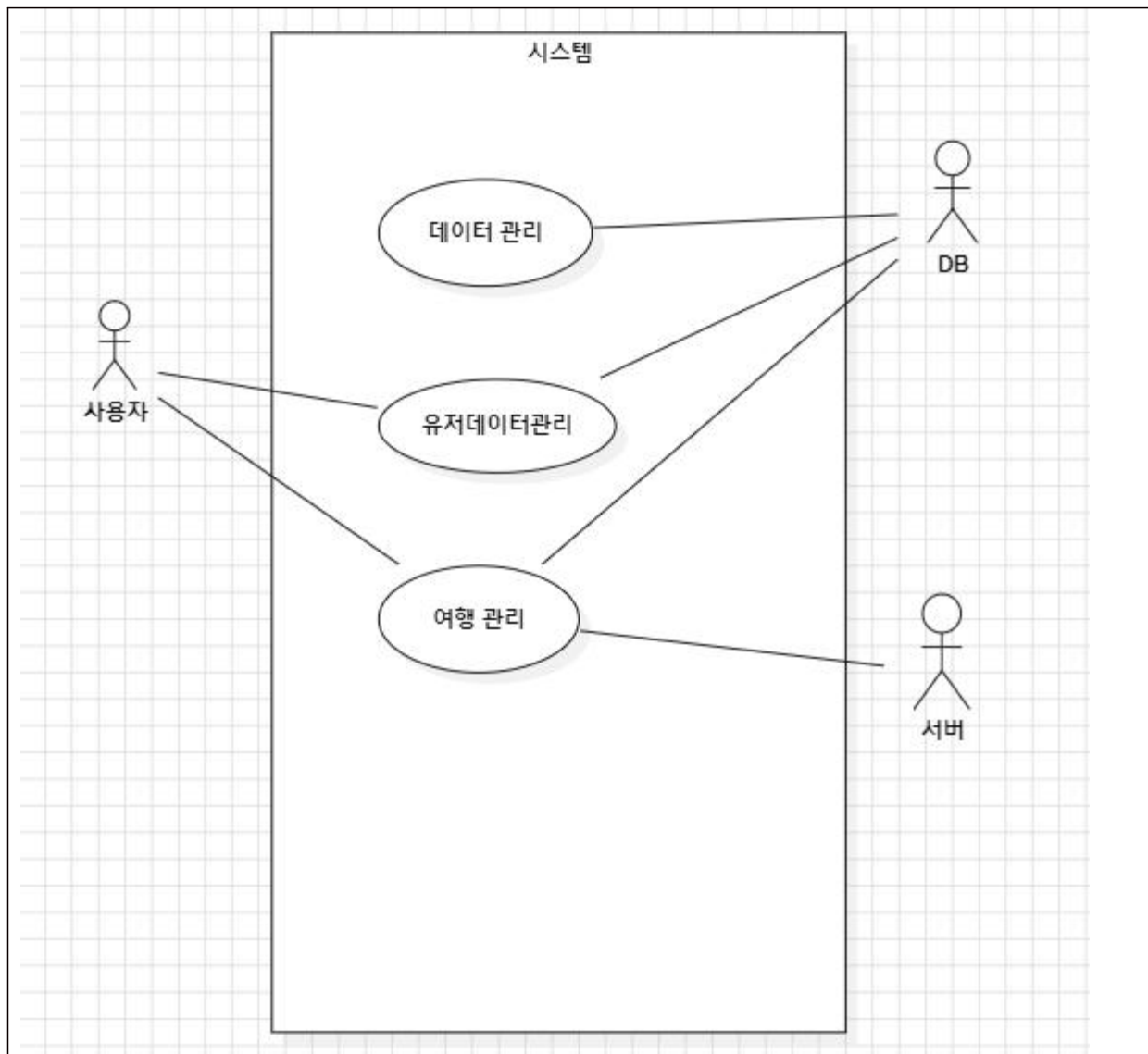


## II. 액터 목록 (Actor List)

번호	액터명	설명	빈도수
1	사용자	테마를 알아내기 위한 설문 하고, 추천받은 여행지를 조회할 수 있고, 여행지 간 최단 경로를 확인하고 대중교통을 사용하여 여행지로 갈 수 있는 경로를 조회할 수 있고 여행지의 별점, 여행지 삭제를 할 수 있다.	보통
2	DB	사용자의 개인정보와 설문 내용, 설문에 따른 점수를 보관하고 사용자의 테마를 사용한 추천 받은 여행지의 정보를 보관하고 있다.	높음
3	서버	사용자의 정보가 DB에 업데이트되면 그 정보를 사용해 사용자에게 추천하는 여행지, 여행지 간의 거리 정보를 DB에 저장시킨다.	보통

### III. 유스 케이스 목록 (Use Case List)

#### 3.1 Use Case Diagram



### III. 유스 케이스 목록 (Use Case List)

#### 3.2 전체 Use Case 목록

Use Case Package	Use Case ID	Use Case Name	Description	중요도	변경가능성	난이도	관련 요구사항
데이터 관리	Data-001	데이터 전송	추천된 데이터와 사용자데이터를 DB로 전송 및 설문조사를 보내준다.	상			
	Data-002	데이터 저장	전송된 데이터를 DB에서 저장한다.	상			
유저 데이터 관리	User-001	로그인 및 설문 검사	최초 실행 시에 로그인과 설문 검사로 설정을 완료한다.	중			
	User-002	로그아웃	사용자 데이터를 DB에 유지한 채 앱에서 로그아웃한다.	중			
	User-003	탈퇴하기	사용자 데이터를 DB에서 지우며 앱에서 나간다.	중			
	User-004	관광지 선호도 재설정	사용자의 선호도 검사를 재실행한다.	하			
여행 관리	Trav-001	여행 시작	사용자의 여행을 시작하는 설정을 진행한다.	상			
	Trav-002	여행 추천	사용자의 데이터를 기반으로 DB와 서버의 알고리즘을 이용해 여행 코스를 추천한다.	상			
	Trav-003	현재 여행	사용자에게 여행 중인 여행 코스를 보여준다.	상			
	Trav-004	여행 종료 및 피드백	현재 여행을 종료하고 피드백을 수행한다.	상			
	Trav-005	여행리스트 관리	여행 리스트를 관리한다.	상			

### III. 유스 케이스 목록 (Use Case List)

---

- Use Case ID: 각 요구사항별로 유일하게 구분할 수 있는 식별자를 기재한다.
- Use Case Name: 유스케이스 명칭을 기재한다.
- Description: 유스케이스가 제공하는 기능에 초점을 두어 간략하게 기술한다.
- Priority: 요구사항의 중요도에 따라 우선순위를 '상', '중', '하'로 기재한다. 필요시 1~5와 같이 상세화할 수 있다. 요구 속성의 상태, 안정성, 난이도를 종합적으로 고려한다. 예를 들어 안정성이 '하'이고 난이도가 '상'인 요구사항은 낮은 우선순위를 가지게 된다.
- Change: 향후 요구사항이 변경될 가능성이 없음을 '상', '중', '하'로 기재한다. 예를 들어 안정성이 '상'은 향후 요구사항이 변경될 가능성이 낮음을 의미한다.
- Difficulty: 해당 요구사항 구현에 어려운 정도를 '상', '중', '하'로 기재한다.
- Relative Requirement: 해당 요구사항으로의 추적이 가능하도록 관련된 요구사항 정의서에 있는 요구사항의 ID를 기재 한다.

## IV. 유스 케이스 명세서 (Use Case Specification)

### 4.1 데이터 전송 Use Case

시스템명	1인 여행자 맞춤 테마 여행 안내 (원데이 서울)		서브시스템명	데이터 전송	
단계명	완료	작성일자	2024-10-24	버전	V1.0
UC-ID	Data-001		이름	데이터 전송	
작성자	장진원		작성일	2024.10.24	
개요	앱에서 변경되거나 저장된 데이터를 DB로 전송한다.				
주요 액터	DB				
사용 빈도	높음				
우선순위	High				
상태	검토 중				
사전 조건	시스템이 구동 중이어야 한다.				
사후 조건	DB에 데이터가 전송되었다.				
기본 흐름	1.시스템은 DB와 통신하여 변경점이 있는지 확인한다. 2.시스템은 DB로 데이터를 전송한다.				
대안 흐름	A-1 분기점: 기본 흐름 1 분기 조건 : 변경점이 없는 경우 1. 시스템은 DB에 데이터를 전송하지 않는다.				
예외 흐름					
비즈니스 규칙	1.시스템과 DB는 이미 설정이 완료된 사용자에게 대해서 저장된다. Rel_0001 : 자료 무결성 1. 시스템은 서버로부터 자료 전송 시 정확성과 무결성을 제공해야 한다. 2. 처리 내역은 반드시 내역 정보에 기록되어야 한다.				
특별 요구사항	Rel_0002 : 시스템 가용성 1. 시스템은 00:00부터 24:00까지 365일 가동되어야 한다. 2. 시스템 오류는 1% 이내로 발생하여야 한다.				

4.2 데이터 저장 use Case

시스템명	1인 여행자 맞춤 테마 여행 안내 (원데이 서울)		서브시스템명	데이터 저장	
단계명	완료	작성일자	2024-10-24	버전	V1.0
UC-ID	Data-002		이름	데이터 저장	
작성자	장진원		작성일	2024.10.24	
개요	시스템에서 전송된 데이터를 DB에 저장한다.				
주요 액터	DB				
사용 빈도	높음				
우선순위	High				
상태	완료				
사전 조건	시스템에서 데이터가 전송되어야 한다.				
사후 조건	DB에 데이터가 저장되었다.				
기본 흐름	1.DB에 해당하는 데이터열이 있는지 확인한다. 2.해당하는 데이터 열에 데이터를 업데이트한다.				
대안 흐름	A-1 분기점: 기본흐름 1 분기 조건 : 해당하는곳이 없을경우 2. DB에 새로운 데이터열을 생성하여 저장한다.				
예외 흐름					
비즈니스 규칙					
특별 요구사항	Rel_0001 : 자료 무결성 1. 시스템은 서버로부터 자료 전송 시 정확성과 무결성을 제공해야 한다. 2. 처리 내역은 반드시 내역 정보에 기록되어야 한다.				
	Rel_0002 : 시스템 가용성 1. 시스템은 00:00부터 24:00까지 365일 가동되어야 한다. 2. 시스템 오류는 1% 이내로 발생하여야 한다.				

4.3 로그인 및 설문검사 use Case

시스템명	1인 여행자 맞춤 테마 여행 안내 (원데이 서울)		서브시스템명	로그인 및 설문검사	
단계명	완료	작성일자	2024-10-24	버전	V1.0
UC-ID	User-001		이름 작성일	로그인 및 설문검사	
작성자	장진원			2024.10.24	
개요	최초 실행 시에 로그인과 설문조사를 진행한다.				
주요 액터	사용자, DB, 서버				
사용 빈도	높음				
우선순위	High				
상태	완료				
사전 조건	시스템에서 데이터가 전송되어야 한다.				
사후 조건	DB에 데이터가 저장되었다.				
기본 흐름	1. 사용자가 최초 실행 시에 로그인을 클릭한다. 2. 시스템이 사용자에게 설문조사를 실시한다. 3. 설문조사가 완료되면 사용자에게 결과를 알려준다. 4. 사용자의 데이터가 DB에 저장된다. 5. 메인 페이지로 이동한다.				
대안 흐름	A-1 분기점: 기본 흐름 2 분기 조건 : 이미 DB에 사용자의 데이터가 있는 경우 기본 흐름 5로 이동한다.				
예외 흐름	E-1 분기점: 기본 흐름 1 분기 조건 : 사용자가 네이버와 구글 계정이 존재하지 않을 경우 1. 사용자가 구글 계정과 네이버 계정을 생성하게 한다.				
비즈니스 규칙	네이버와 구글 계정을 사용할 수 있어야 한다.				
특별 요구사항	Rel_0001 : 자료 무결성 1. 시스템은 서버로부터 자료 전송 시 정확성과 무결성을 제공해야 한다. 2. 처리 내역은 반드시 내역 정보에 기록되어야 한다. Rel_0002 : 시스템 가용성 1. 시스템은 00:00부터 24:00까지 365일 가동되어야 한다. 2. 시스템 오류는 1% 이내로 발생하여야 한다.				

4.4 로그아웃 use Case

시스템명	1인 여행자 맞춤 테마 여행 안내 (원데이 서울)		서브시스템명	로그아웃	
단계명	완료	작성일자	2024-10-24	버전	V1.0
UC-ID	User-002		이름	로그아웃	
작성자	장진원		작성일	2024.10.24	
개요	사용자가 시스템을 통해 로그아웃한다.				
주요 액터	사용자				
사용 빈도	높음				
우선순위	High				
상태	완료				
사전 조건	시스템에 사용자가 로그인되어 있어야 한다.				
사후 조건	시스템에서 로그아웃된다.				
기본 흐름	1. 사용자가 시스템에서 유저 데이터관리(톱니바퀴)를 선택한다. 2. 사용자가 시스템에서 로그아웃을 선택한다. 3. 사용자가 시스템에서 로그아웃된다.				
대안 흐름	A-1 분기 흐름: 로그아웃 선택에서 아니오를 선택한 경우 1. 시스템이 다시 유저 데이터관리로 돌아간다.				
예외 흐름					
비즈니스 규칙					
특별 요구사항	Rel_0001 : 자료 무결성 1. 시스템은 서버로부터 자료 전송 시 정확성과 무결성을 제공해야 한다. 2. 처리 내역은 반드시 내역 정보에 기록되어야 한다. Rel_0002 : 시스템 가용성 1. 시스템은 00:00부터 24:00까지 365일 가동되어야 한다. 2. 시스템 오류는 1% 이내로 발생하여야 한다.				



4.5 탈퇴하기 use Case

시스템명	1인 여행자 맞춤 테마 여행 안내 (원데이 서울)		서브시스템명	탈퇴하기	
단계명	완료	작성일자	2024-10-24	버전	V1.0
UC-ID	User-003		이름 작성일	탈퇴하기	
작성자	장진원			2024.10.24	
개요	사용자가 시스템에서 탈퇴한다.				
주요 액터	DB,사용자				
사용 빈도	높음				
우선순위	High				
상 태	완료				
사전 조건	시스템에 사용자가 로그인되어 있어야 한다.				
사후 조건	사용자가 시스템에서 탈퇴한다.				
기본 흐름	1. 사용자가 시스템에서 유저 데이터 관리(톱니바퀴)를 선택한다. 2. 사용자가 시스템에서 탈퇴하기를 선택한다. 3. 사용자의 데이터가 DB에서 삭제된다.				
대안 흐름	A-1 분기점: 기본흐름 2 분기 조건 : 탈퇴하기에서 아니오를 누른 경우 1. 탈퇴하기가 취소되고 유저 데이터 관리 창으로 돌아간다.				
예외 흐름					
비즈니스 규칙					
특별 요구사항	Rel_0001 : 자료 무결성 1. 시스템은 서버로부터 자료 전송 시 정확성과 무결성을 제공해야 한다. 2. 처리 내역은 반드시 내역 정보에 기록되어야 한다. Rel_0002 : 시스템 가용성 1. 시스템은 00:00부터 24:00까지 365일 가동되어야 한다. 2. 시스템 오류는 1% 이내로 발생하여야 한다.				

4.6 관광지 선호도 재설정 use Case

시스템명	1인 여행자 맞춤 테마 여행 안내 (원데이 서울)		서브시스템명	관광지 선호도 재설정	
단계명	완료	작성일자	2024-10-24	버전	V1.0
UC-ID	User-004		이름 작성일	관광지 선호도 재설정	
작성자	장진원			2024.10.24	
개요	사용자가 관광지 선호도를 재설정한다.				
주요 액터	DB,사용자				
사용 빈도	높음				
우선순위	High				
상 태	완료				
사전 조건	사용자가 로그인되어 있어야 한다.				
사후 조건	DB에 데이터가 저장되었다.				
기본 흐름	1. 사용자가 시스템에서 유저 데이터 관리(톱니바퀴)를 선택한다.				
	2. 사용자가 관광지 선호도 재설정을 선택한다.				
	3. 시스템이 사용자에게 설문조사를 실시한다.				
	4. 설문조사가 완료되면 사용자에게 결과를 알려준다.				
	5. 사용자의 데이터가 DB에 저장된다.				
	6. 유저 데이터 관리로 돌아간다.				
대안 흐름					
예외 흐름	E-1				
	분기점: 기본 흐름 3				
	분기 조건: 사용자가 설문조사 중에 뒤로가기를 누를 경우				
비즈니스 규칙	1. 유저 데이터 관리로 돌아간다.				
특별 요구사항	Rel_0001 : 자료 무결성				
	1. 시스템은 서버로부터 자료 전송 시 정확성과 무결성을 제공해야 한다.				
	2. 처리 내역은 반드시 내역 정보에 기록되어야 한다.				
	Rel_0002 : 시스템 가용성				
	1. 시스템은 00:00부터 24:00까지 365일 가동되어야 한다.				
	2. 시스템 오류는 1% 이내로 발생하여야 한다.				

4.7 여행 시작 use Case

시스템명	1인 여행자 맞춤 테마 여행 안내 (원데이 서울)		서브시스템명	여행 시작	
단계명	완료	작성일자	2024-10-24	버전	V1.0
UC-ID	Trav-001		이름 작성일	여행 시작	
작성자	장진원			2024.10.24	
개요	사용자의 여행을 시작한다.				
주요 액터	사용자,DB,서버				
사용 빈도	높음				
우선순위	High				
상태	완료				
사전 조건	사용자가 시스템에 로그인되어 있어야 한다.				
사후 조건	여행 추천으로 이동한다.				
기본 흐름	1. 사용자가 여행 시작을 선택한다.				
	2. 여행의 시작 위치와 시작 날짜를 선택한다.				
	3. 가고 싶은 지역구와 여행의 이름을 지정한다.				
	4. 해당 여행의 데이터를 DB에 저장한다.				
대안 흐름	A-1				
	분기점: 기본 흐름 2				
	분기 조건 : 시작 위치를 선택하지 않은 경우				
	1. 시작 위치를 사용자의 현재 위치로 설정한다.				
예외 흐름	A-2				
	분기점: 기본 흐름 3				
	분기 조건: 여행의 이름이 설정되지 않은 경우				
	1. 해당 여행의 이름을 여행을 시작하는 날짜의 이름으로 지정한다.				
비즈니스 규칙	E-1				
	분기점: 기본 흐름 1				
	분기 조건: 이미 여행이 진행 중인 경우				
	1. 현재 여행을 종료하고 피드백을 진행한다.				
특별 요구사항	2. 피드백 데이터가 DB로 전송된다.				
	사용자의 현재 위치 GPS를 사용할 수 있어야 한다.				
	Rel_0001 : 자료 무결성				
	1. 시스템은 서버로부터 자료 전송 시 정확성과 무결성을 제공해야 한다.				
특별 요구사항	2. 처리 내역은 반드시 내역 정보에 기록되어야 한다.				
	Rel_0002 : 시스템 가용성				
	1. 시스템은 00:00부터 24:00까지 365일 가동되어야 한다.				
	2. 시스템 오류는 1% 이내로 발생하여야 한다.				

4.8 여행 추천 use Case

시스템명	1인 여행자 맞춤 테마 여행 안내 (원데이 서울)		서브시스템명	여행 추천	
단계명	완료	작성일자	2024-10-24	버전	V1.0
UC-ID	Trav-002		이름 작성일	여행 추천	
작성자	장진원			2024.10.24	
개요	여행 코스를 추천한다.				
주요 액터	DB, 사용자, 서버				
사용 빈도	높음				
우선순위	High				
상태	완료				
사전 조건	여행시작 단계가 완료되어야 한다.				
사후 조건	여행 코스가 추천된다.				
기본 흐름	1. 시스템은 서버에 사용자 데이터를 보낸다.				
	2. 서버는 알고리즘 수행 후 데이터를 DB로 전송한다.				
	3. 시스템은 DB에 저장된 여행 코스를 읽어온다.				
	4. 시스템은 사용자에게 여행 코스를 보여준다.				
대안 흐름	A-1				
	분기점: 기본 흐름 2				
예외 흐름	분기 조건 : 가고 싶어 하는 여행지가 '없음'일 경우				
	1. 여행지 추천 알고리즘에서 카테고리별 추천에서 가장 첫 번째로 나온 여행지를 해당 여행의 목적지로 설정한다,				
비즈니스 규칙					
특별 요구사항	Rel_0001 : 자료 무결성				
	1. 시스템은 서버로부터 자료 전송 시 정확성과 무결성을 제공해야 한다.				
	2. 처리 내역은 반드시 내역 정보에 기록되어야 한다.				
	Rel_0002 : 시스템 가용성				
	1. 시스템은 00:00부터 24:00까지 365일 가동되어야 한다.				
	2. 시스템 오류는 1% 이내로 발생하여야 한다.				

4.9 현재 여행 use Case

시스템명	1인 여행자 맞춤 테마 여행 안내 (원데이 서울)		서브시스템명	현재 여행	
단계명	완료	작성일자	2024-10-24	버전	V1.0
UC-ID	Trav-003		이름 작성일	현재 여행	
작성자	장진원			2024.10.24	
개요	시스템은 사용자에게 현재 여행 코스를 보여준다.				
주요 액터	사용자, 서버, DB				
사용 빈도	높음				
우선순위	High				
상태	완료				
사전 조건	여행지가 추천되어 있어야 한다.				
사후 조건	세부 여행 코스가 보여진다.				
기본 흐름	1. 사용자가 현재 여행을 선택한다.				
	2. 추천된 여행 코스를 보여준다.				
대안 흐름	3. 각 코스별로 선택 시 세부 여행 코스가 보여진다.				
예외 흐름	E-1				
	분기점: 기본 흐름 2				
비즈니스 규칙	분기 조건: 현재 여행 중이 아닐 경우				
	1. 현재 여행에 아무것도 나오지 않는다.				
특별 요구사항	Rel_0001 : 자료 무결성				
	1. 시스템은 서버로부터 자료 전송 시 정확성과 무결성을 제공해야 한다.				
	2. 처리 내역은 반드시 내역 정보에 기록되어야 한다.				
	Rel_0002 : 시스템 가용성				
	1. 시스템은 00:00부터 24:00까지 365일 가동되어야 한다.				
	2. 시스템 오류는 1% 이내로 발생하여야 한다.				

4.10 여행종료 및 피드백 use Case

시스템명	1인 여행자 맞춤 테마 여행 안내 (원데이 서울)		서브시스템명	여행종료 및 피드백	
단계명	완료	작성일자	2024-10-24	버전	V1.0
UC-ID	Trav-004		이름 작성일	여행종료 및 피드백	
작성자	장진원			2024.10.24	
개요	시스템에서 전송된 데이터를 DB에 저장한다.				
주요 액터	DB, 사용자, 서버				
사용 빈도	높음				
우선순위	High				
상태	완료				
사전 조건	현재 여행이 진행 중이어야 한다.				
사후 조건	DB에 데이터가 저장되었다.				
기본 흐름	1. 사용자가 유저 데이터 관리(톱니바퀴)를 선택한다. 2. 사용자가 여행 종료를 선택한다. 3. 시스템이 사용자에게 피드백을 요청한다. 4. 사용자의 피드백 데이터를 서버에 전송한다. 5. 메인 페이지로 이동한다.				
대안 흐름	A-1 분기점: 기본 흐름 2 분기 조건 : 여행 종료 선택 시 아니오를 선택한다.				
	1. 유저 데이터 관리로 돌아간다. E-1 분기점: 기본 흐름 2 분기 조건: 현재 여행이 진행 중이지 않은 경우 1. 여행 종료 버튼이 동작하지 않는다.				
예외 흐름					
비즈니스 규칙					
특별 요구사항	Rel_0001 : 자료 무결성 1. 시스템은 서버로부터 자료 전송 시 정확성과 무결성을 제공해야 한다. 2. 처리 내역은 반드시 내역 정보에 기록되어야 한다.				
	Rel_0002 : 시스템 가용성 1. 시스템은 00:00부터 24:00까지 365일 가동되어야 한다. 2. 시스템 오류는 1% 이내로 발생하여야 한다.				

4.11 여행리스트 관리 use Case

시스템명	1인 여행자 맞춤 테마 여행 안내 (원데이 서울)		서브시스템명	여행리스트 관리	
단계명	완료	작성일자	2024-10-24	버전	V1.0
UC-ID	Trav-005		이름	여행리스트 관리	
작성자	장진원			2024.10.24	
개요	저장된 여행 리스트를 관리한다.				
주요 액터	사용자, 서버, DB				
사용 빈도	높음				
우선순위	High				
상 태	완료				
사전 조건	여행 리스트에 여행이 있어야 한다.				
사후 조건	여행 리스트 관리가 종료된다.				
기본 흐름	1. 사용자가 시스템의 여행리스트를 선택한다.				
	2. 시스템이 사용자에게 저장되어 있는 여행 목록을 보여준다.				
대안 흐름	3. 현재 여행이 선택한 여행으로 바뀐 후 사용자에게 여행이 바뀔을 알려준다.				
	A-1				
예외 흐름	분기점: 기본 흐름 3				
	분기 조건 : 이미 여행이 진행 중인 경우				
비즈니스 규칙	1. 시스템이 사용자에게 현재 진행 중인 여행을 종료하는지 물어본다.				
	2. 사용자가 종료 선택 시 여행이 종료된다.				
	3. 시스템이 사용자에게 피드백을 요청한다.				
	4. 사용자가 완료한 피드백을 DB로 전송한다.				
	5. DB에 전달된 정보를 서버가 읽어와 피드백 파일을 생성한다.				
	6. 시스템은 여행리스트 창으로 돌아간다.				
	E-1				
	분기점:모든 흐름				
	분기 조건: 사용자가 삭제 버튼을 누른 경우				
	1. 시스템이 사용자에게 여행을 삭제하는지 물어본다.				
	2. 사용자가 여행을 삭제한다.				
	3. 시스템이 DB에게 여행리스트의 변동을 전송한다.				
	4. DB는 변동된 여행리스트를 저장한다.				

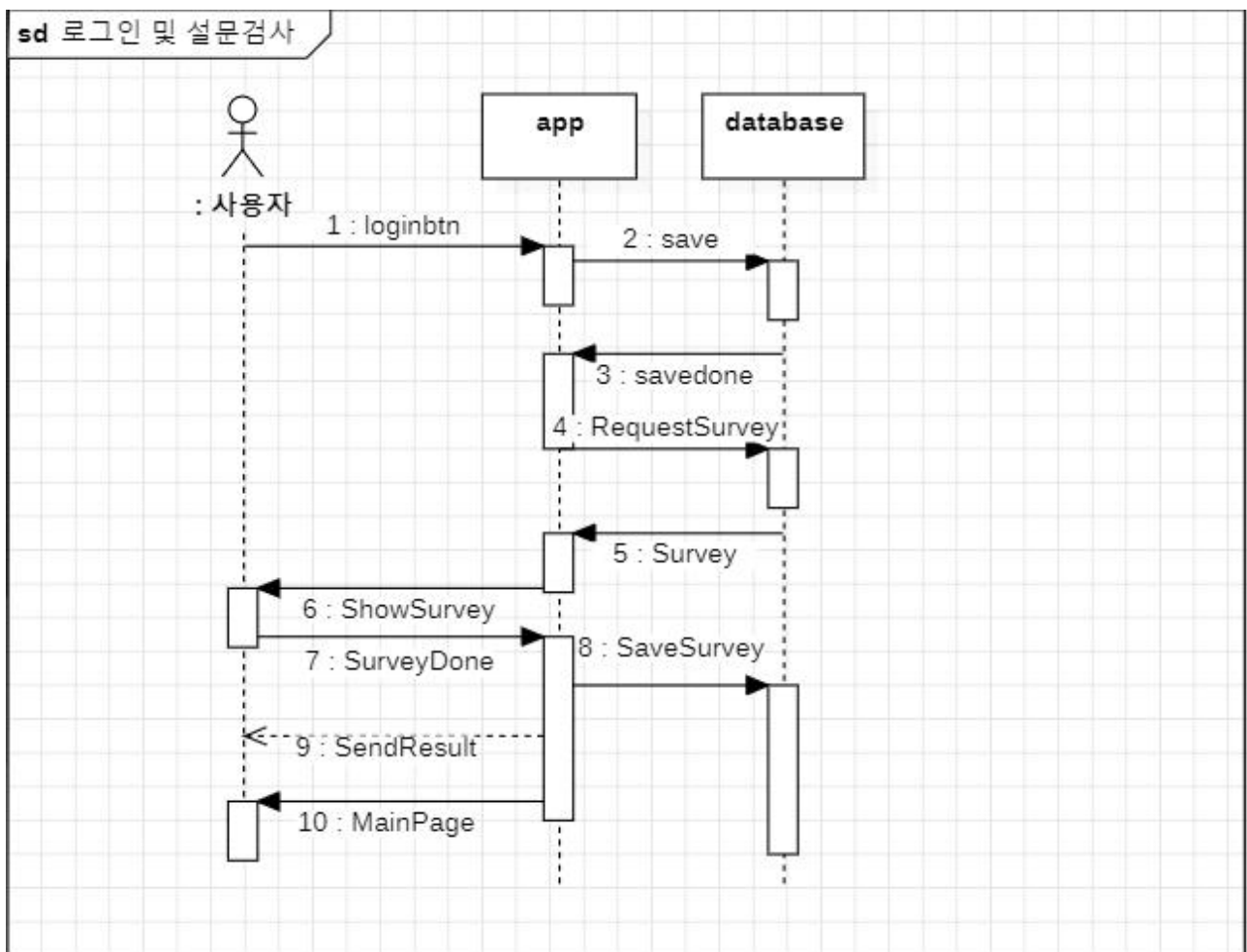
특별 요구사항	<p>Rel_0001 : 자료 무결성</p> <ol style="list-style-type: none"><li>1. 시스템은 서버로부터 자료 전송 시 정확성과 무결성을 제공해야 한다.</li><li>2. 처리 내역은 반드시 내역 정보에 기록되어야 한다.</li></ol> <p>Rel_0002 : 시스템 가용성</p> <ol style="list-style-type: none"><li>1. 시스템은 00:00부터 24:00까지 365일 가동되어야 한다.</li><li>2. 시스템 오류는 1% 이내로 발생하여야 한다.</li></ol>
------------	---



## V. 순서 다이어그램 (Sequence Diagram)

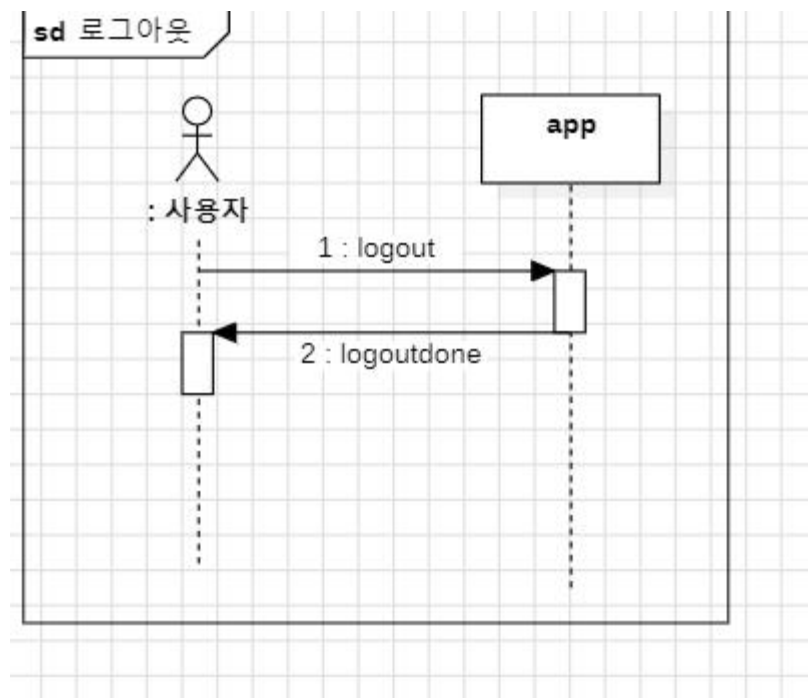
### 5.1 로그인 및 설문조사 Sequence Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	로그인 및 설문검사	
단계명	완료	작성일자	2024-10-28	버전	V1.0
Sequence-ID	User-001		이 름	로그인 및 설문검사	
작성자	장진원		작성일	2024.10.28	
개 요	최초 실행 시에 로그인과 설문 검사를 진행한다.				



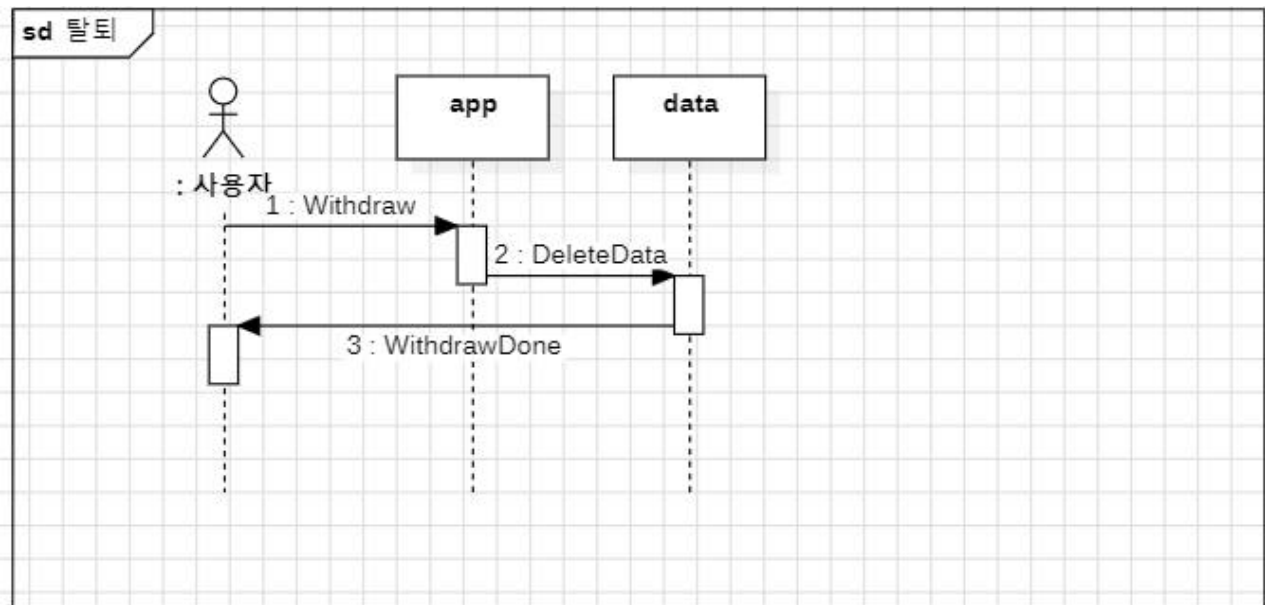
## 5.2 로그아웃 Sequence Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	로그아웃	
단계명	완료	작성일자	2024-10-28	버전	V1.0
Sequence-ID	User-002		이름	로그아웃	
작성자	장진원		작성일	2024.10.28	
개요	사용자가 시스템에서 로그아웃한다.				



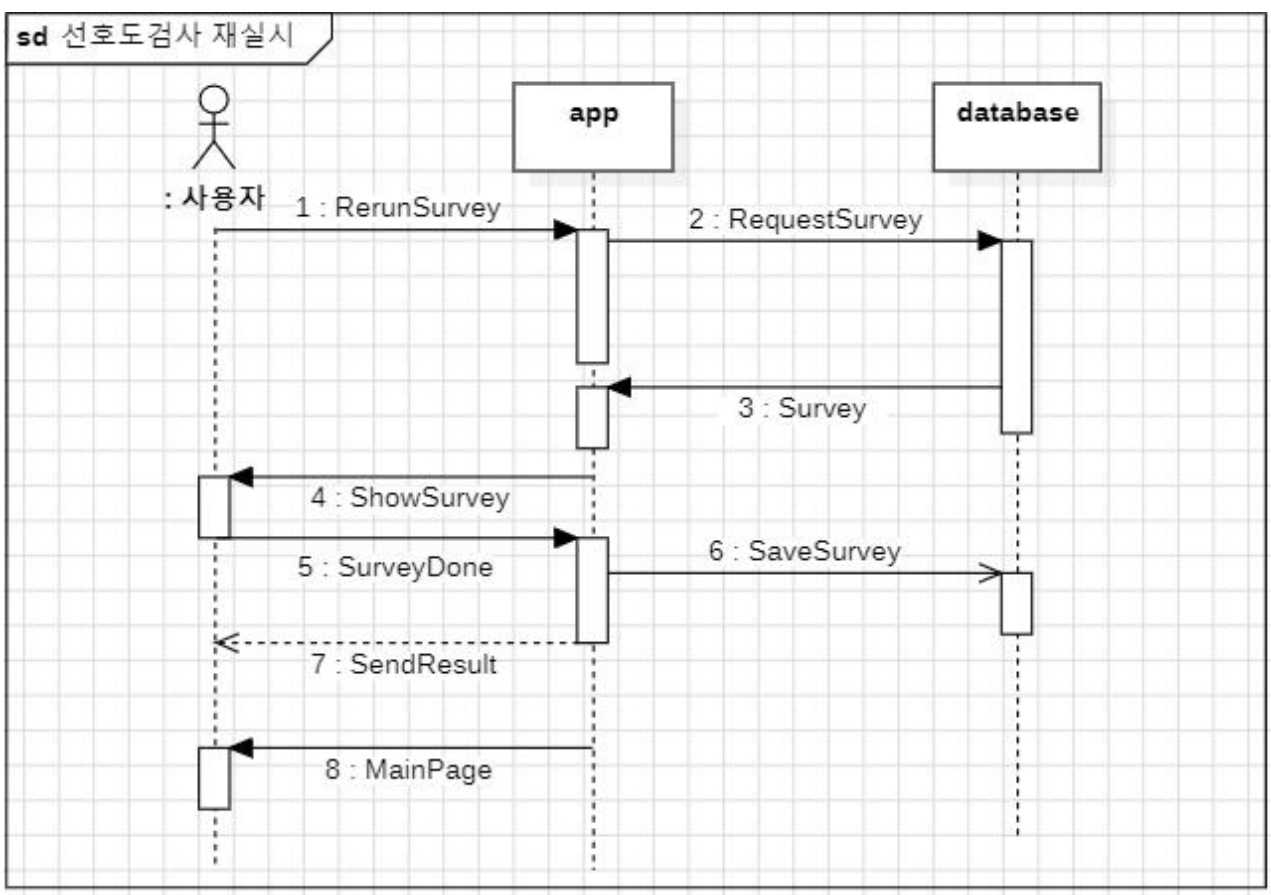
5.3 탈퇴하기 Sequence Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	탈퇴하기	
단계명	완료	작성일자	2024-10-28	버전	V1.0
Sequence-ID	User-003		이름	탈퇴하기	
작성자	장진원		작성일	2024.10.28	
개요	사용자가 시스템에서 탈퇴한다.				



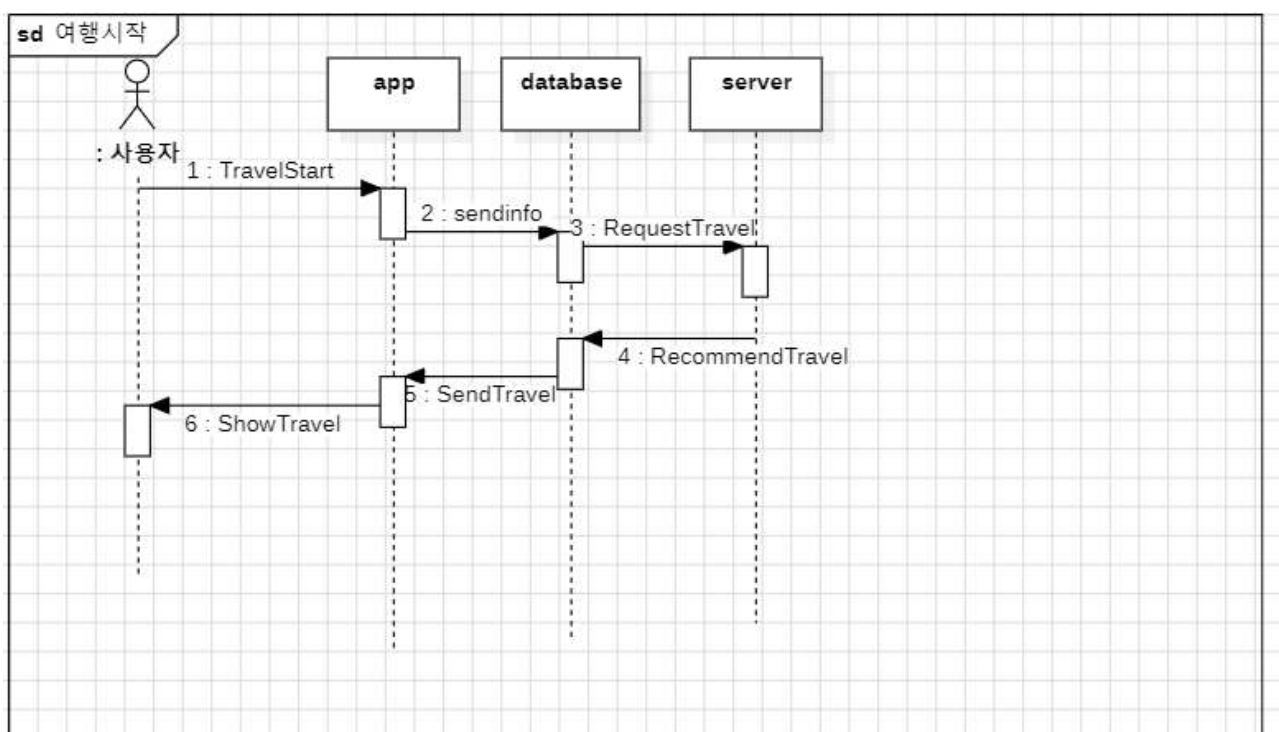
## 5.4 관광지 선호도 재설정 Sequence Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	관광지 선호도 재설정	
단계명	완료	작성일자	2024-10-28	버전	V1.0
Sequence-ID	User-004		이 름	관광지 선호도 재설정	
작성자	장진원		작성일	2024.10.28	
개 요	사용자의 관광지 선호도를 재설정한다.				



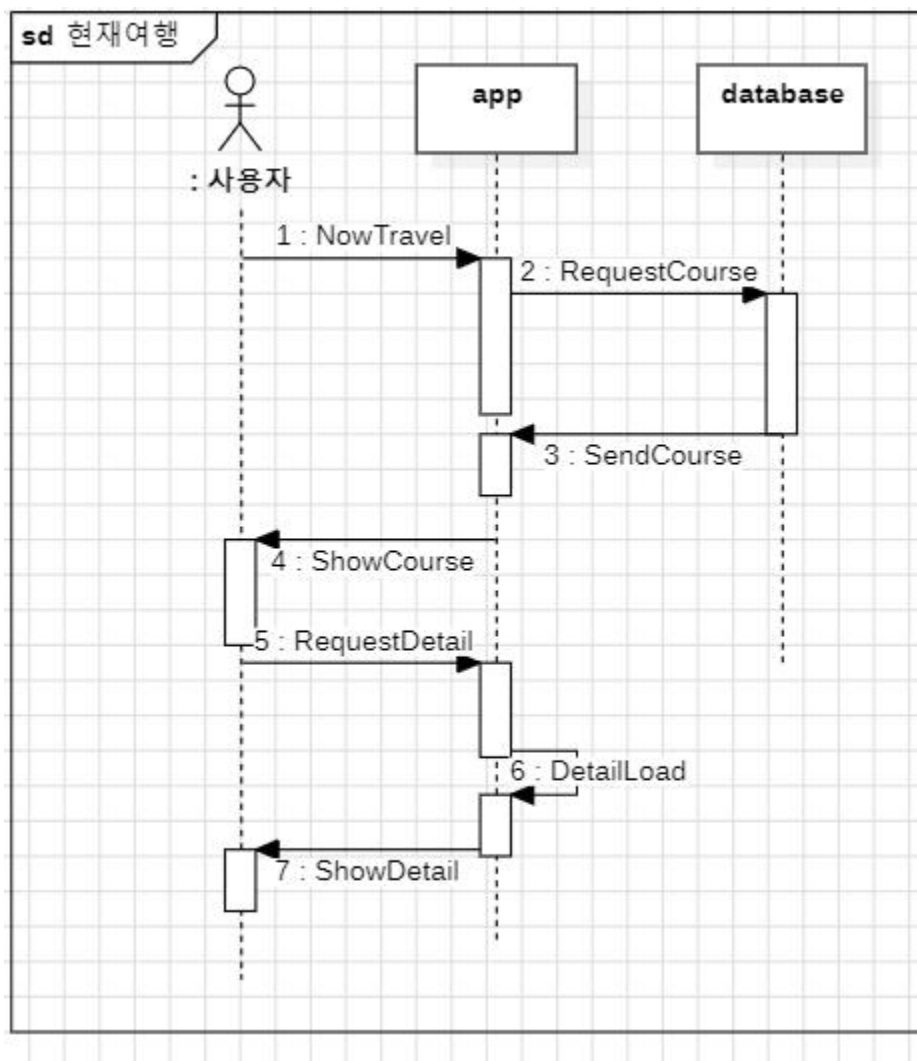
## 5.5 여행 시작 Sequence Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	여행시작	
단계명	완료	작성일자	2024-10-28	버전	V1.0
Sequence-ID	Trav-001		이름	여행시작	
작성자	장진원		작성일	2024.10.28	
개요	시스템이 여행 시작 시에 여행 추천을 해준다.				



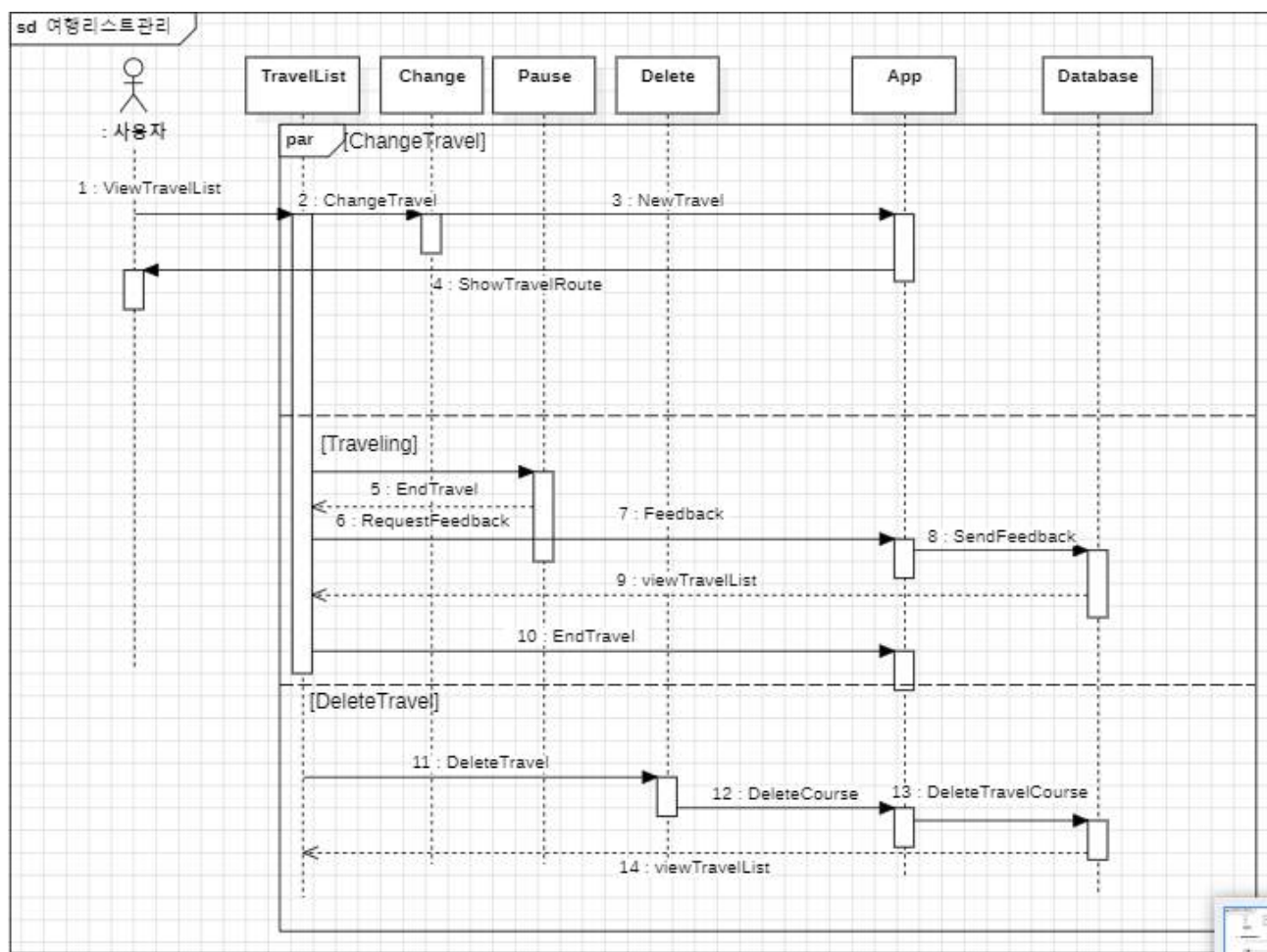
5.6 현재 여행 Sequence Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	현재 여행	
단계명	완료	작성일자	2024-10-28	버전	V1.0
Sequence-ID	Trav-002		이름	현재 여행	
작성자	장진원		작성일	2024.10.28	
개요	현재 여행 코스를 보여준다.				



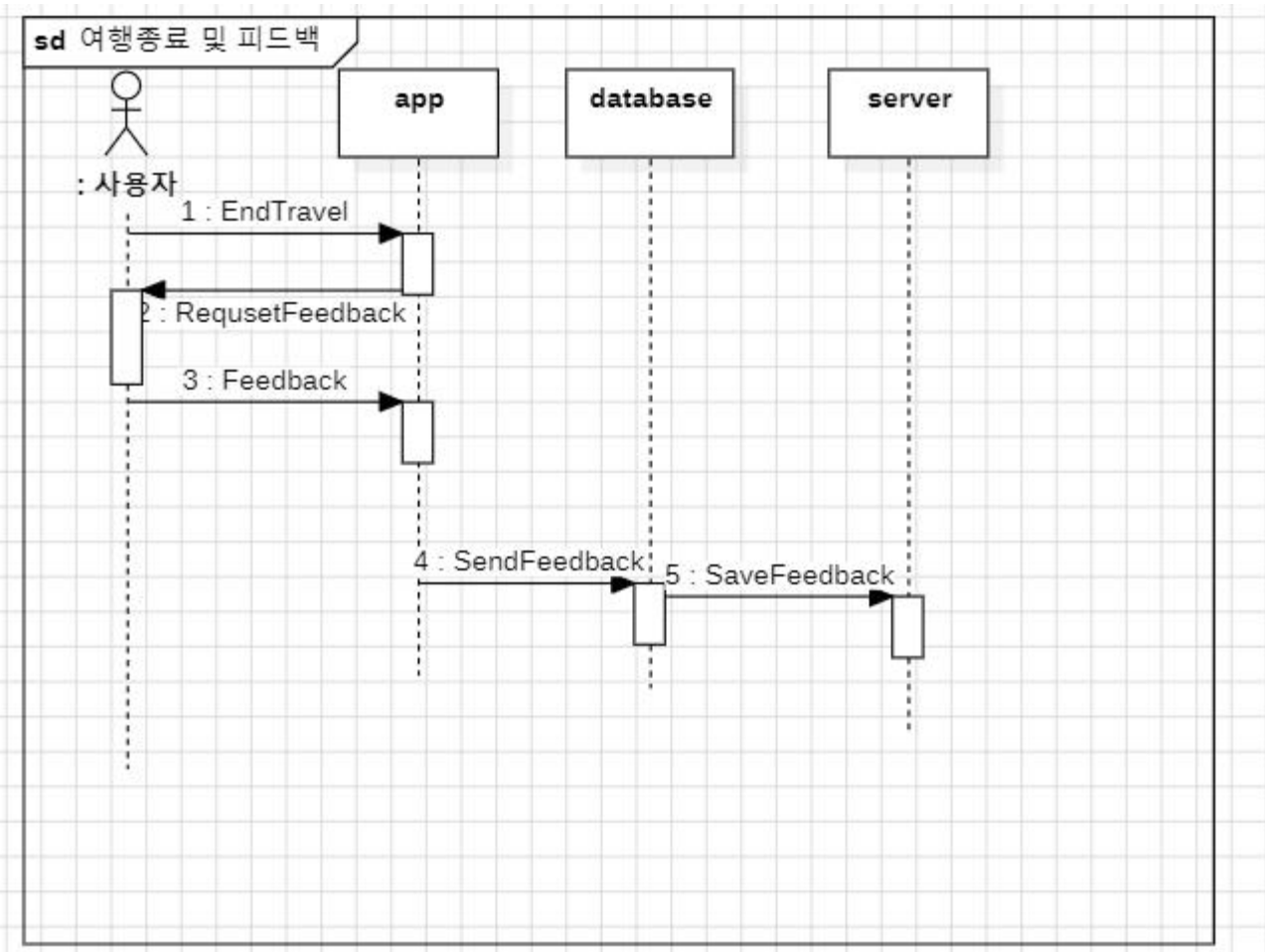
## 5.7 여행 리스트 관리 Sequence Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	여행 리스트 관리	
단계명	완료	작성일자	2024-10-28	버전	V1.0
Sequence-ID	Trav-003		이 름	여행 리스트 관리	
작성자	장진원		작성일	2024.10.28	
개 요	저장된 여행 코스를 관리한다.				



5.8 여행종료 및 피드백 Sequence Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	여행종료 및 피드백	
단계명	완료	작성일자	2024-10-28	버전	V1.0
Sequence-ID	Trav-004		이름	여행종료 및 피드백	
작성자	장진원		작성일	2024.10.28	
개요	여행이 종료되면 피드백을 진행한다.				

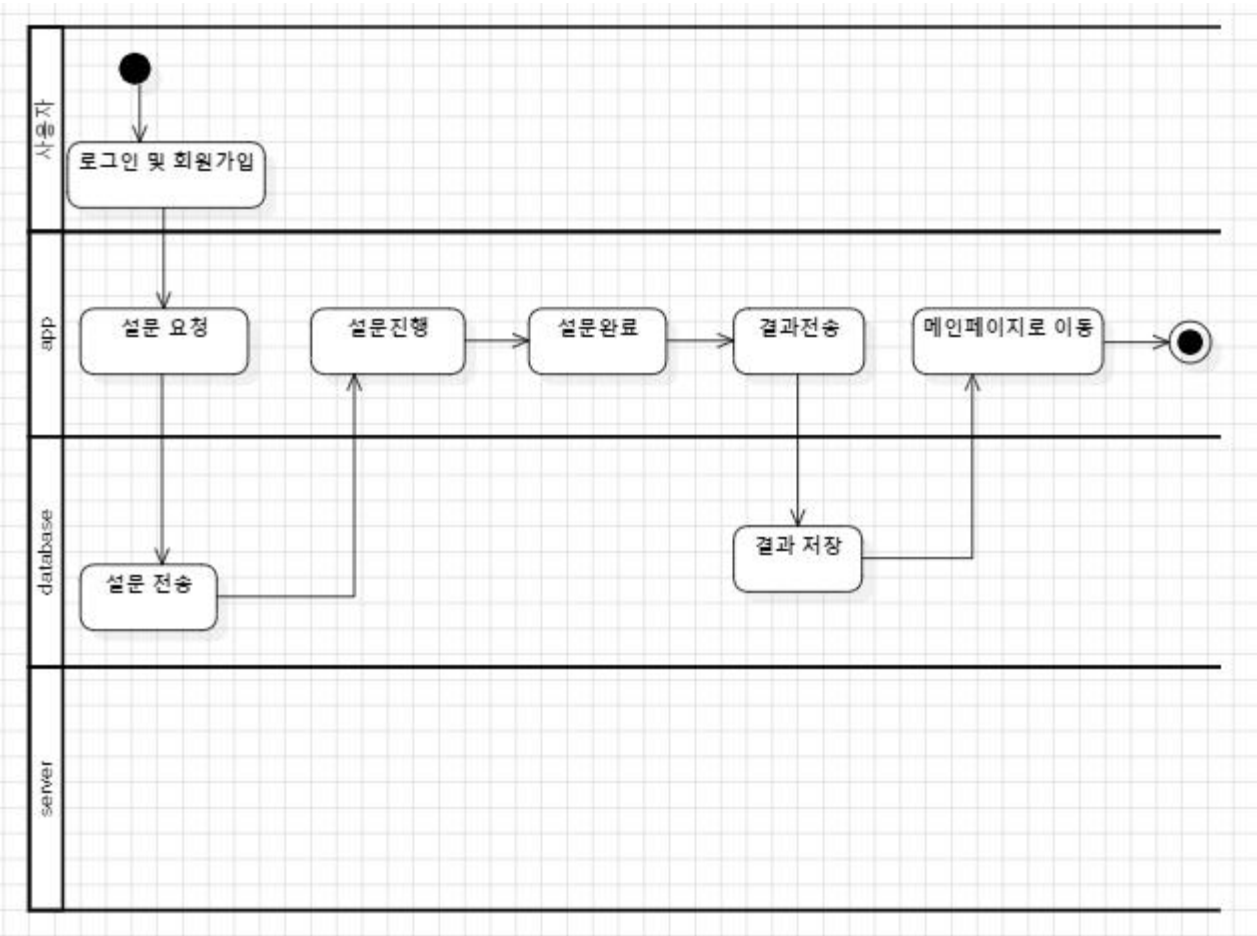




## VI. 활동 다이어그램 (Activity Diagram)

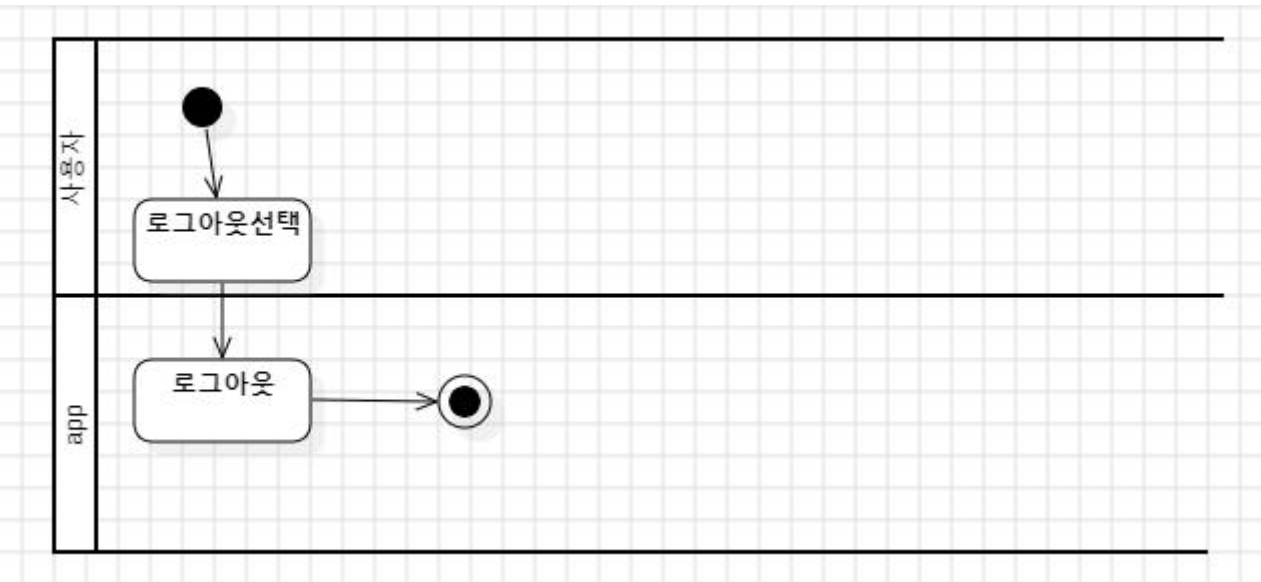
### 6.1 로그인 및 설문조사 Activity Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	로그인 및 설문조사	
단계명	완료	작성일자	2024.10.28	버전	V1.0
Sequence-ID	User-001		이 름	로그인 및 설문조사	
작성자	장진원		작성일	2024.10.28	
개 요	시스템에 로그인 및 설문 조사하는 과정을 기술한다.				



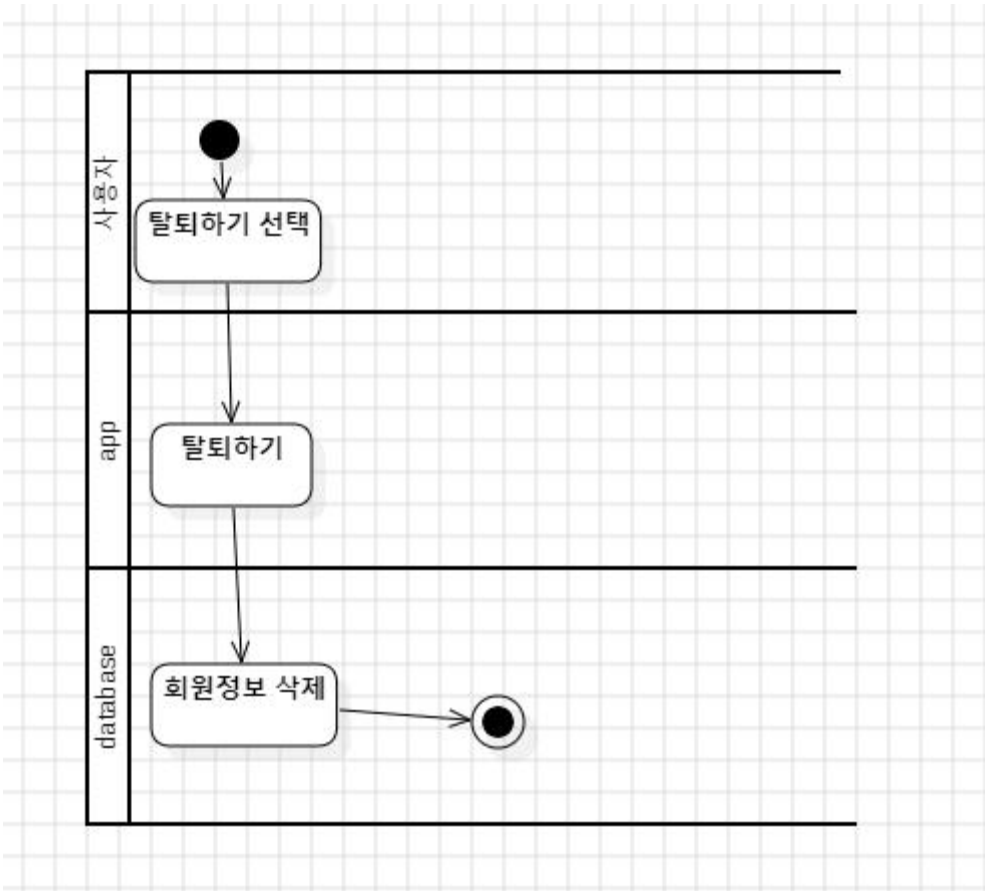
6.2 로그아웃 Activity Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	로그아웃	
단계명	완료	작성일자	2024.10.28	버전	V1.0
Sequence-ID	User-002		이 름	로그아웃	
작성자	장진원		작성일	2024.10.28	
개 요	시스템에서 로그아웃하는 과정을 기술한다.				



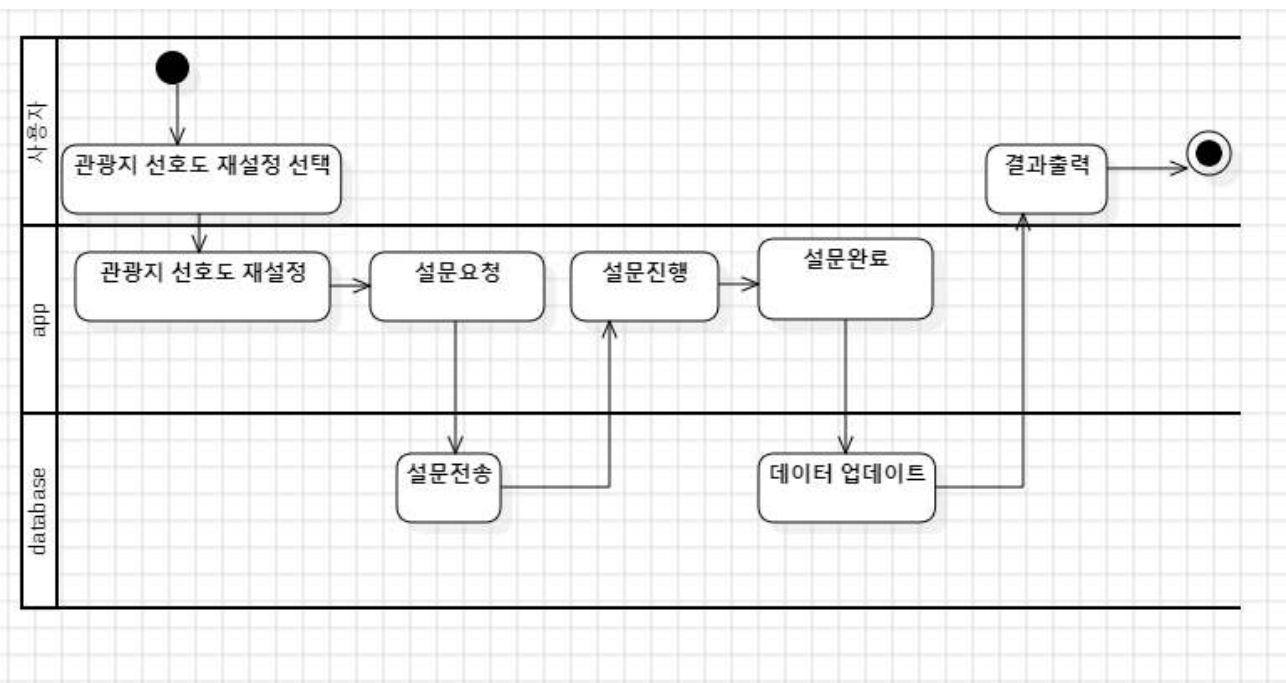
6.3 탈퇴하기 Activity Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	탈퇴하기	
단계명	완료	작성일자	2024.10.28	버전	V1.0
Sequence-ID	User-003		이 름	탈퇴하기	
작성자	장진원		작성일	2024.10.28	
개 요	시스템에서 탈퇴하는 과정을 기술한다.				



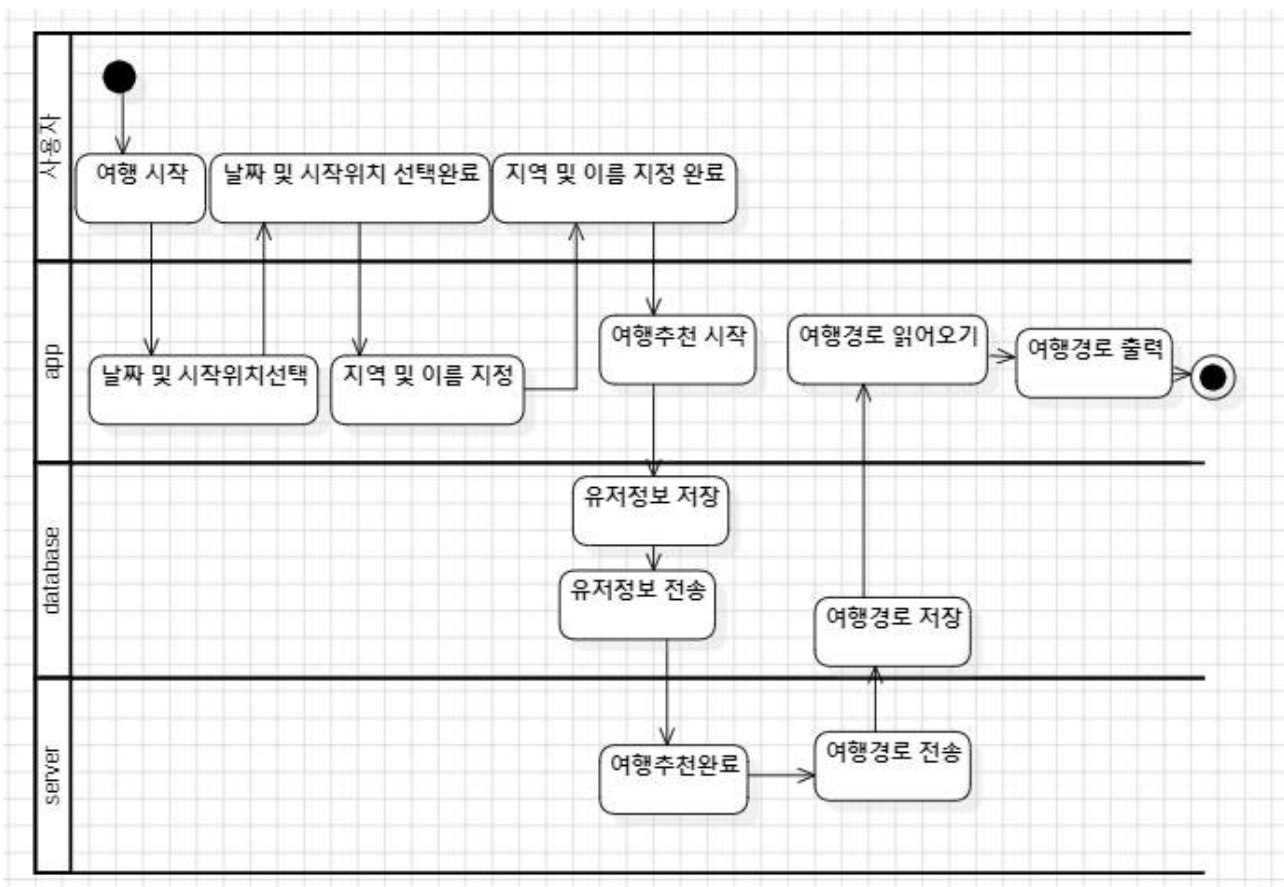
6.4 관광지 선호도 재설정 Activity Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	관광지 선호도 재설정	
단계명	완료	작성일자	2024.10.28	버전	V1.0
Sequence-ID	User-004		이 름	관광지 선호도 재설정	
작성자	장진원		작성일	2024.10.28	
개 요	사용자의 관광지 선호도 재설정하는 과정을 기술한다.				



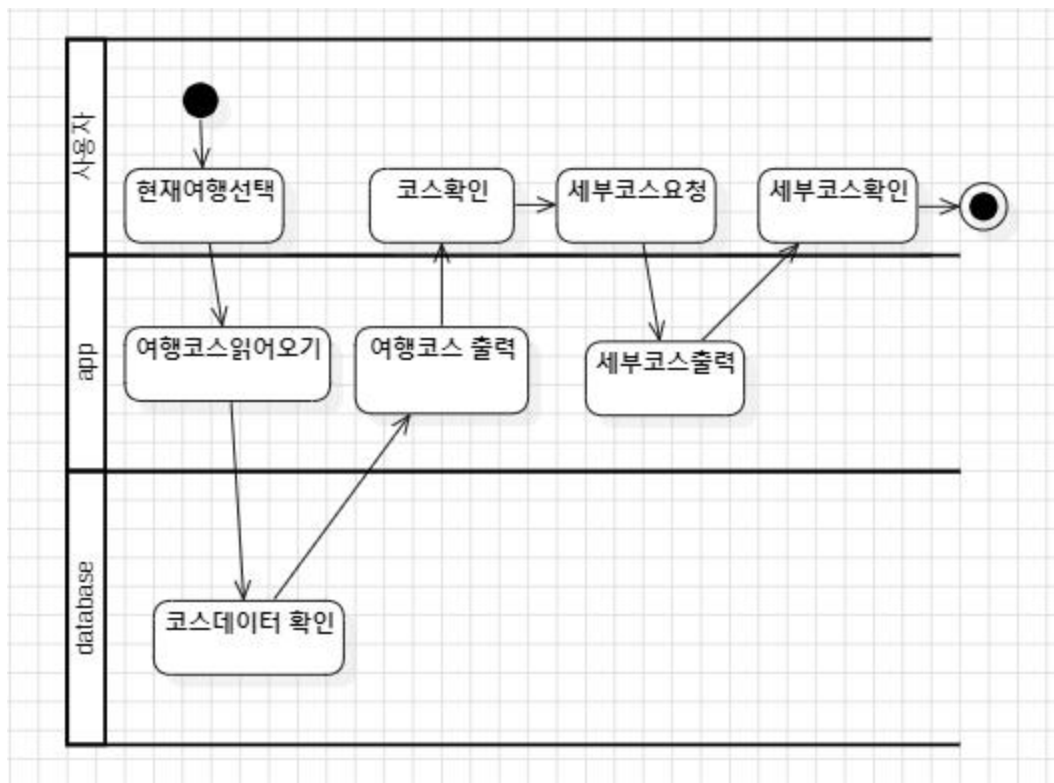
## 6.5 여행시작 Activity Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	여행 시작	
단계명	완료	작성일자	2024.10.28	버전	V1.0
Sequence-ID	Trav-001		이름	여행 시작	
작성자	장진원		작성일	2024.10.28	
개요	시스템이 여행 시작하여 여행 추천하는 과정을 기술한다.				



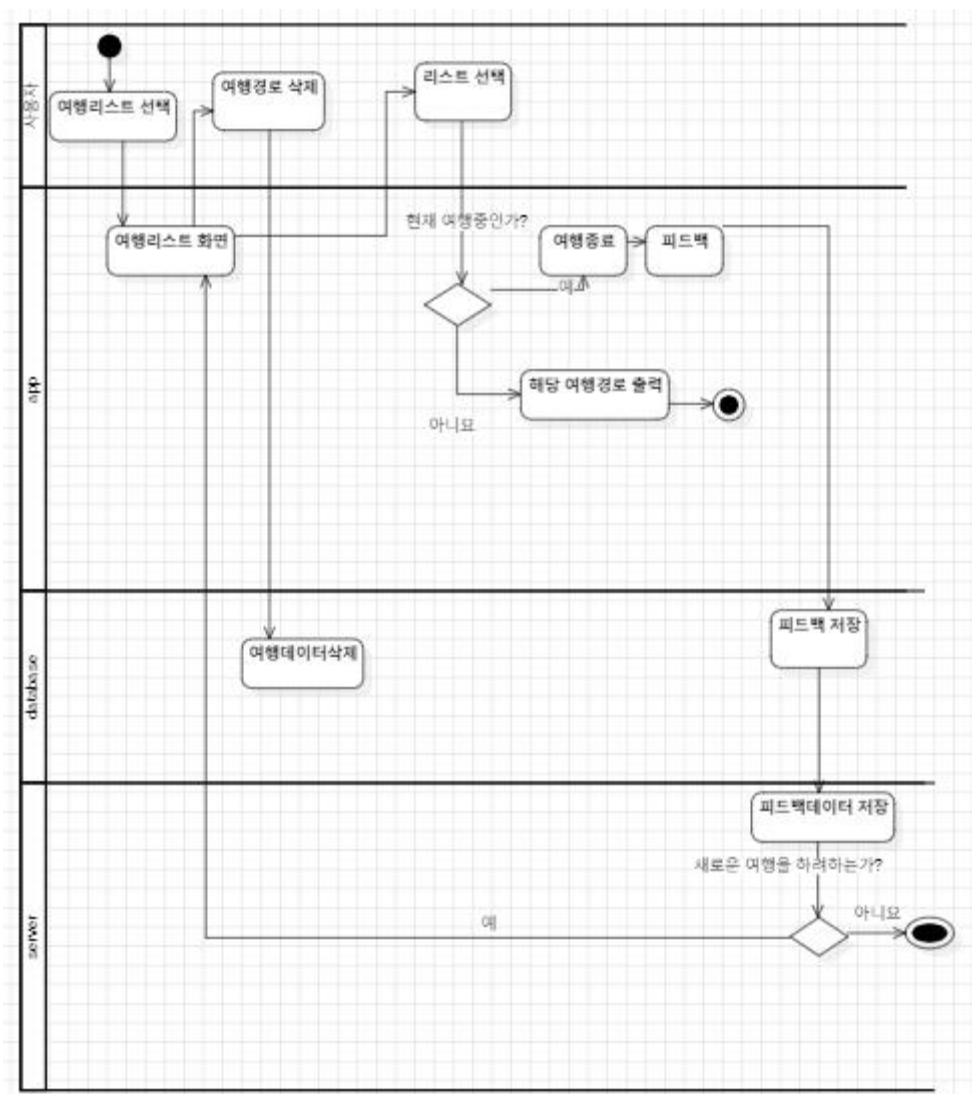
6.6 현재 여행 Activity Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	현재 여행	
단계명	완료	작성일자	2024.10.28	버전	V1.0
Sequence-ID	Trav-002		이 름	현재 여행	
작성자	장진원		작성일	2024.10.28	
개 요	시스템이 사용자에게 현재 여행 코스를 보여주는 과정을 기술한다.				



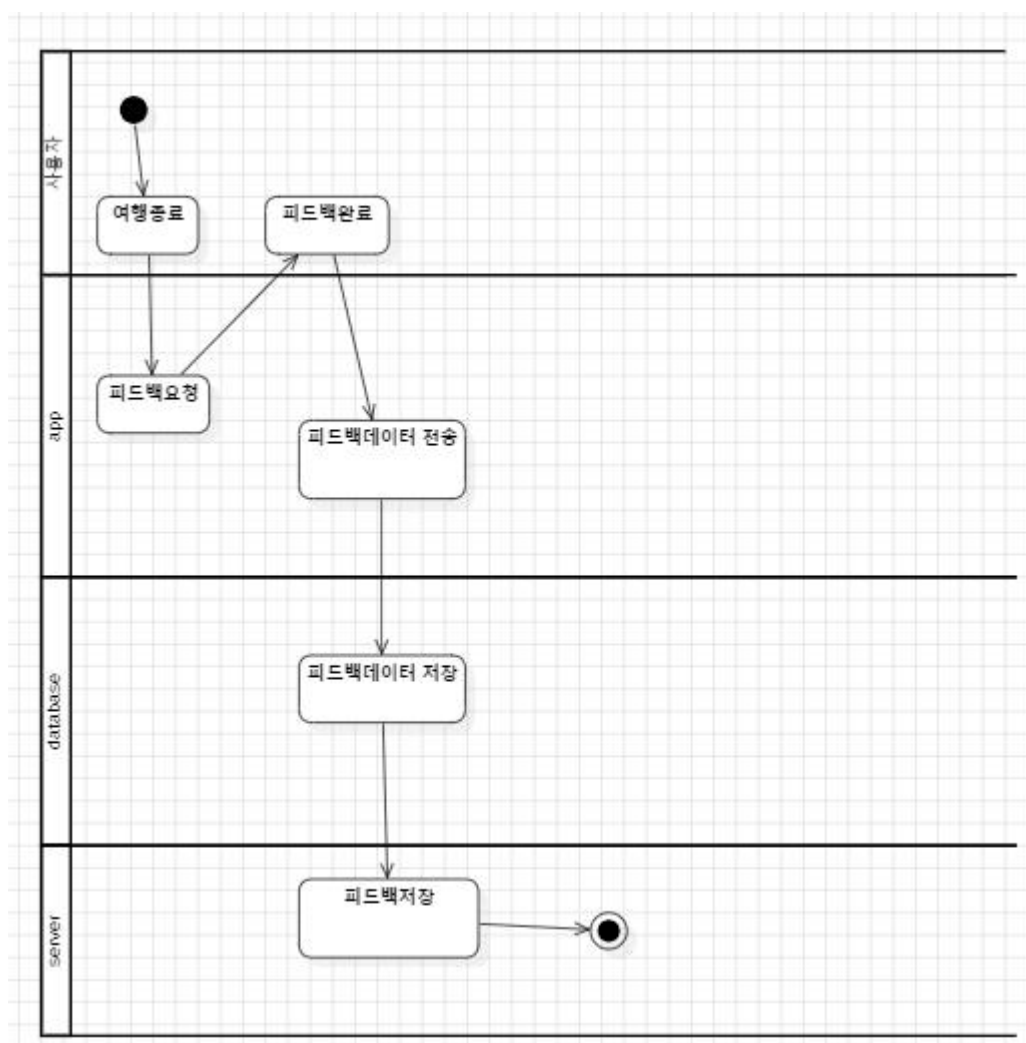
6.7 여행 리스트 관리 Activity Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	여행리스트 관리	
단계명	완료	작성일자	2024.10.28	버전	V1.0
Sequence-ID	Trav-003		이 름	여행리스트 관리	
작성자	장진원		작성일	2024.10.28	
개 요	사용자가 시스템을 통해 여행리스트를 관리하는 과정을 기술한다.				



## 6.8 여행 종료 및 피드백 Activity Diagram

시스템명	1인 여행자 맞춤 테마 여행 안내(원데이 서울)		서브시스템명	여행 종료 및 피드백	
단계명	완료	작성일자	2024.10.28	버전	V1.0
Sequence-ID	Trav-004		이 름	여행 종료 및 피드백	
작성자	장진원		작성일	2024.10.28	
개 요	시스템이 여행 종료 후 피드백하는 과정을 기술한다.				





## IV. 요구사항 명세서

## I. 개 요 (Introduction)

---

### 1.1 목 적 (Purpose)

- 본 소프트웨어 요구사항 명세서는 1인 여행자 맞춤 테마 여행 안내(원데이 서울)에 대한 SW 기능 및 비기능 요구사항을 기술한다.
- 이 문서의 목적은 프로젝트팀 구성원이 시스템의 완전한 기능을 구축하고 검증하기 위해 읽고 사용하기 위한 것이다.
- 이 문서에 기술된 모든 요구사항은 높은 우선순위를 가지고 릴리즈 1.0에 구축된다.

### 1.2 범 위 (Scope)

- 이 프로젝트의 범위는 혼자 여행을 다니는 20대를 위해 선호도를 검사하고 여행지와 경로를 추천해 주는 시스템을 구축하는 것이다.
- 사용자는 선호도 조사 후 여행지를 추천받을 수 있다.
- 1인 여행자 맞춤 테마 여행 안내(원데이 서울)은 Android에서 사용 가능하다.

### 1.3 용어 및 약어 (Term and Abbreviation)

약어	원어	설명
SRS	Software Requirement Specification	소프트웨어 요구사항 명세서
OE	Operating Environment	시스템 운영 환경
CO	General Constraints	일반 제약사항
AS	Assumption	고려사항
FUR	Functional Requirement	기능 요구사항
DAR	Data Requirement	데이터 요구사항
IFR	Interface Requirement	인터페이스 요구사항
PER	Performance Requirement	성능 요구사항
RER	Reliable Requirement	신뢰성 요구사항
USR	Usable Requirement	사용성 요구사항
MAR	Maintainable Requirement	유지보수성 요구사항
POR	Portability Requirement	이식성 요구사항
SER	Security Requirement	보안 요구사항
COR	Constraints Requirement	제약사항

## 2. 시스템 개요

---

### 2.1 시스템 특성 (System Perspective)

- 이 시스템은 혼자 서울 투어를 다니는 20대를 위해 선호도를 검사하고 여행지를 추천하고 경로를 안내하는 시스템이다.
- 이 시스템은 서울의 관광지 정보를 통해 사용자의 선호도에 맞는 관광지를 추천해 준다.

### 2.2 사용자 특성 (User Characteristics)

혼자 서울 투어 하는 20대

- 서울에 유명한 관광지는 많지만, 사용자의 선호도와 맞는 알려지지 않은 장소를 찾기에 어려움이 있다.
- 사용자는 선호도 검사를 통해 여행지를 추천받고 경로를 안내받을 수 있다.
- 사용자는 다시 투어를 갈 때 선호도 검사를 다시 하지 않아도 여행지를 추천받을 수 있다.

### 2.3 운영환경 (Operating Environment)

- OE-1: 시스템은 Android에서 작동해야 한다.
- OE-2: 시스템은 Ubuntu 서버에서 수행해야 한다.
- OE-3: 시스템은 사용자의 사용 환경에 따라 사용자의 화면이 조정되어야 한다.
- OE-4: 인공지능 모델 개발은 Python 언어를 사용한다.
- OE-6: 프레임워크는 Flutter, TensorFlow를 사용한다.

### 2.4 일반 제약사항 (General Constraints)

- CO-1: 개인정보 수집, 저장, 처리 및 공개에 대한 명확한 개인정보 처리 방침을 제공해야 한다.
- CO-2: 사용자가 쉽게 이해할 수 있는 UI를 제공해야 한다.
- CO-3: 회원 탈퇴 시 사용자의 정보를 파기한다.
- CO-4: 여행지가 콘서트나 극장이 나오는 경우 사용자가 직접 예약 해야 한다.
- CO-5: 사용자의 회원가입을 통해 사용자 고유의 정보를 저장할 수 있어야 한다.

### 2.5 고려사항 (Assumption and Dependencies)

- AS-1: 여행지를 추천받고 싶다면 로그인해야 한다.
- AS-2: 앱은 24시 상시 운영해야 한다.

### 3. 기능 요구사항 (Functional Requirement)

#### 3.1 전체 기능 리스트 (WBS)

대기능 (업무 기능)	중기능 (개략 기능)	소기능 (상세 기능)
여행지 관리	현재 여행	FUR-001 여행지 생성
	여행 리스트	FUR-002 여행 리스트 관리
정보 관리	사용자 데이터 관리	FUR-003 사용자 회원 정보 관리
		FUR-004 관광지 선호도 관리

### 3. 기능 요구사항 (Functional Requirement)

#### 3.2 개별 기능 요구사항

요구사항 번호	FUR-001	요구사항 유형	기능
요구사항 이름	여행지 생성		
요구사항 개요	사용자가 여행지를 생성한다.		
요구사항 내용	<div>1. 사용자가 여행 시작 버튼을 누른다.</div> <div>2. 시스템은 사용자에게 필요한 정보(출발 위치, 여행 시작 날짜, 가고 싶은 구, 여행 이름 등)을 입력 받는 폼을 제공한다.</div> <div>3. 사용자는 시스템이 요구하는 정보를 입력한다.</div> <div>4. 시스템은 사용자의 정보를 사용하여 여행지를 생성한다.</div>		
입력 데이터	출발 위치, 여행 시작 날짜, 가고 싶은 구, 여행 이름 등	출력 데이터	여행지
입출력 유형	외부 입력(EI)	데이터 파일 유형	내부파일
중요도	상	위험도	하
품질속성	기능성, 보안성, 효율성, 사용성		
평가 방법	<div>1. 여행지가 정상적으로 생성되는지 확인한다.</div> <div>2. 사용자에게 요구하는 정보가 DB에 성공적으로 저장이 되었는지 확인한다.</div>		
관련 요구사항	FUR-002		
요구사항 출처			

요구사항 번호	FUR-002	요구사항 유형	기능
요구사항 이름	여행 리스트 관리		
요구사항 개요	사용자가 생성한 여행 리스트를 관리한다.		
요구사항 내용	1. 사용자가 여행 리스트 버튼을 누른다. 2. 시스템은 사용자에게 사용자가 생성한 여행 리스트를 보여준다. 3. 사용자는 여행 리스트를 확인하고 여행 리스트를 삭제, 선택을 할 수 있다. 3.1 사용자가 여행 리스트 삭제를 요청하면 시스템은 해당 여행 리스트를 삭제한다. 3.2 사용자가 여행 리스트를 선택한다. 3.2.1 시스템은 사용자에게 현재 여행 창에서 여행지와 여행 경로를 표시한다. 3.2.2 사용자가 현재 여행 리스트와 다른 여행 리스트를 선택하면 시스템은 현재 여행 리스트 종료 창을 보여주고 여행 별점을 입력 받는다.		
입력 데이터	여행 리스트 선택, 여행 리스트 사용	출력 데이터	여행 리스트 삭제, 여행지, 여행 경로
입출력 유형	외부 입력(EI)	데이터 파일 유형	내부파일
중요도	상	위험도	중
품질 속성	기능성, 보안성, 효율성, 사용성		
평가 방법	1. 여행 리스트가 정상적으로 표시되는지 확인한다. 2. 여행 리스트 삭제 요청 시 해당 여행 리스트가 DB에서 정상적으로 제거되는지 확인한다. 3. 여행 리스트 선택 요청 시 현재 여행 창에 결과가 정상적으로 표시되는지 확인한다. 4. 현재 여행 리스트와 다른 여행 리스트를 선택했을 때 올바른 창이 표시되는지 확인한다.		
관련 요구사항	FUR-001		
요구사항 출처	예상 사용자 설문조사 및 회의록		

요구사항 번호	FUR-003	요구사항 유형	기능
요구사항 이름	사용자 회원 정보 관리		
요구사항 개요	사용자의 개인 정보를 관리한다.		
요구사항 내용	<p>1.1 시스템은 사용자의 선호도 결과를 사용자가 조회할 수 있도록 해야 한다.</p> <p>1.2 시스템은 사용자가 요청할 경우 등록된 모든 기본정보를 삭제할 수 있어야 한다.</p> <p>1.3 시스템은 사용자가 요청할 경우 로그아웃이 되어야 한다.</p>		
입력 데이터	사용자 데이터	출력 데이터	로그아웃, 회원 탈퇴, 선호도 결과 조회
입출력 유형	외부 입력(EI)	데이터 파일 유형	내부파일
중요도	상	위험도	중
품질 속성	기능성, 보안성, 효율성, 사용성		
평가 방법	<p>1-1. 사용자가 등록하고 입력한 정보를 내부 DB에서 확인한다.</p> <p>1-2. 사용자가 삭제하고 내부 DB를 확인한다.</p> <p>1-3. 사용자가 선호도 결과를 조회할 수 있는지 확인한다.</p>		
관련 요구사항	FUR-002		
요구사항 출처	예상 사용자 설문조사 및 회의록		

요구사항 번호	FUR-004	요구사항 유형	기능
요구사항 이름	관광지 선호도 관리		
요구사항 개요	사용자의 관광지 선호도를 관리한다.		
요구사항 내용	<p>1.1 시스템은 사용자가 선호도 검사 결과를 조회할 수 있도록 해야 한다.</p> <p>1.2 시스템은 사용자가 선호도 검사를 다시 받을 수 있도록 해야 한다.</p> <p>1.3 시스템은 사용자에게 방문한 여행지의 별점을 받을 수 있도록 해야 한다.</p>		
입력 데이터	관광지 선호도 데이터	출력 데이터	선호도 검사 결과, 여행지 별점
입출력 유형	외부 입력(EI)	데이터 파일 유형	내부파일
중요도	상	위험도	중
품질 속성	기능성, 보안성, 효율성, 사용성		
평가 방법	<p>1-1. 사용자가 등록한 선호도 검사 내용을 내부 DB에서 확인한다.</p> <p>1-2. 사용자가 수정한 선호도 검사 내용을 내부 DB에서 확인한다.</p> <p>1-3. 사용자가 준 별점을 내부 엑셀 파일에서 확인한다.</p>		
관련 요구사항	FUR-002		
요구사항 출처	예상 사용자 설문조사 및 회의록		



## 4. 데이터 요구사항 (Data Requirement)

요구사항 번호	DAR-001	요구사항 유형	데이터
요구사항 이름	데이터 보존		
요구사항 개요	데이터 보존 기간 및 자료 저장 형태를 정의한다.		
요구사항 내용	1. 시스템은 사용자가 여행 리스트 삭제를 요청할 때까지는 영구적으로 사용자의 여행 리스트를 보존한다. 2. 시스템은 사용자가 계정 삭제를 요청할 때까지는 영구적으로 사용자의 정보를 보존한다.		
중요도	상	위험도	중
품질속성	기능성, 보안성, 효율성, 신뢰성		
평가 방법	1. 여행 리스트 삭제 요청을 받은 데이터 정보가 DB에서 삭제되는지 확인한다. 2. 계정 삭제 요청을 받은 데이터 정보가 DB에서 삭제되는지 확인한다.		
관련 요구사항	FUR-003, FUR-004		
요구사항 출처			

## 5. 인터페이스 요구사항 (Interface Requirement)

요구사항 번호	IFR-001	요구사항 유형	인터페이스
요구사항 이름	앱 인터페이스		
요구사항 개요	개발 시스템의 인터페이스 내용을 기술한다.		
요구사항 내용	<ol style="list-style-type: none"> <li>1. 시스템은 사용자에게 보기 쉬운 인터페이스를 제공해야 한다.</li> <li>2. 사용자에게 친숙한 구글맵을 이용하여 경로를 보기 쉽게 해야 한다.</li> <li>3. 앱을 처음 이용하는 사용자들도 모든 기능을 잘 다룰 수 있도록 한다.</li> </ol>		
중요도	상	위험도	중
품질속성	상호운용성, 사용성		
평가 방법	<ol style="list-style-type: none"> <li>1. 사용자가 인터페이스와 상호 작용하는 데 문제가 없는지 확인한다.</li> <li>2. 앱을 처음 사용하는 사용자가 기능을 잘 다루는지 확인한다.</li> </ol>		
관련 요구사항	FUR-001,FUR-002		
요구사항 출처			

## 6. 성능 요구사항 (Performance Requirement)

요구사항 번호	PER-001	요구사항 유형	성능
요구사항 이름	응답시간		
요구사항 개요	사용자가 여행 생성을 누를 시 생성 시간을 정의한다.		
요구사항 내용	1. 사용자가 여행지 생성을 할 때 3초 이내에 여행지가 생성되어야 한다. 2. 사용자가 버튼을 앱에서 상호작용을 할 때 화면 이동의 시간이 짧아야 한다.		
중요도	상	위험도	상
품질속성	기능성, 효율성, 신뢰성, 사용성		
평가 방법	1. 여행지 생성 시간을 확인한다. 2. 화면 이동 시간을 확인한다.		
관련 요구사항	FUR-001, FUR-002, FUR-003, FUR-004		
요구사항 출처			

## 7. 품질 요구사항 (Quality Requirement)

### 7.1 신뢰성 요구사항 (Reliability Requirements)

요구사항 번호	RER-001	요구사항 유형	신뢰성
요구사항 이름	데이터의 정확성		
요구사항 개요	시스템이 제공하는 데이터의 정확성을 확인하기 위한 방법을 기술		
요구사항 내용	1. 시스템이 제공하는 데이터가 정확해야 한다. 2. 여행지 이름과 주소가 정확한지 확인한다. 3. 여행지 이동 간 길 찾기 경로가 정확해야 한다. 4. 여행지는 사용자의 선호도가 높은 테마인지 확인한다.		
중요도	상	위험도	상
품질속성	기능성, 효율성, 신뢰성		
평가 방법	1. 길 찾기 경로가 최적의 경로인지 확인한다. 2. 시스템이 제공하는 여행지와 실제 여행지 주소를 확인한다. 3. 생성된 여행지의 선호도 점수와 사용자의 선호도 점수를 비교한다.		
관련 요구사항	FUR-001, FUR-002		
요구사항 출처			

## 7. 품질 요구사항 (Quality Requirement)

### 7.2 사용성 요구사항 (Usability Requirements)

요구사항 번호	USR-001	요구사항 유형	사용성
요구사항 이름	기능 접근의 용이성		
요구사항 개요	사용자가 시스템의 기능을 쉽게 사용하고 제어하는 데 필요한 속성을 정의한다.		
요구사항 내용	1. 쉽고 간단한 인터페이스로 사용자가 여행 생성과 여행지 선택하기 쉽도록 만든다. 2. 시스템은 사용자의 환경에 맞춰서 화면 구성을 변경한다. 3. 시스템은 기능 메뉴의 위치와 논리적으로 추정 가능하게 배치하여 설계되어야 한다.		
중요도	상	위험도	상
품질속성	상호 운용성, 사용성		
평가 방법	1. 사용자의 단말 기종에 맞춰서 화면이 변경되는지 확인한다. 2. 사용자의 단말에서 시스템이 논리적(사용 빈도나 접근성)인 기준에서 화면이 구성되었는지 확인한다.		
관련 요구사항			
요구사항 출처			

## 7. 품질 요구사항 (Quality Requirement)

### 7.3 유지보수성 요구사항 (Maintainability Requirements)

요구사항 번호	MAR-001	요구사항 유형	유지보수성
요구사항 이름	시스템 업그레이드		
요구사항 개요	시스템의 변경이나 업그레이드 등이 발생할 때 변경처리 절차를 정의한다.		
요구사항 내용	1. 시스템은 새로운 기능을 구현하기 위해 필요한 경우 업그레이드를 해야 한다. 2. 시스템은 보안사고 예방, 성능 개선 등이 발생하는 경우 패치를 해야 한다.		
중요도	상	위험도	상
품질속성	상호 운용성, 사용성		
평가 방법	1. 기능 변경이 자동 패치되는지 확인한다.		
관련 요구사항			
요구사항 출처			

## 7. 품질 요구사항 (Quality Requirement)

### 7.4 이식성 요구사항 (Portability Requirements)

요구사항 번호	POR-001	요구사항 유형	이식성
요구사항 이름	시스템 이식성		
요구사항 개요	시스템의 변경이나 업그레이드 등이 발생할 때 변경처리 절차를 정의한다.		
요구사항 내용	1.1 시스템은 호환 가능한 Android 운영체제에서 수행될 수 있어야 한다. 1.2 시스템은 신규 업데이트가 등록되었을 때 업데이트 가능 여부를 안내해야 한다.		
중요도	상	위험도	상
품질속성	상호 운용성, 사용성		
평가 방법	1. 다양한 모바일 환경에서 기능이 동작하는지 확인한다.		
관련 요구사항			
요구사항 출처			

## 7. 품질 요구사항 (Quality Requirement)

### 7.5 보안성 요구사항 (Security Requirements)

요구사항 번호	SER-001	요구사항 유형	보안성
요구사항 이름	보안성 요구사항		
요구사항 개요	인가받지 못한 사용자나 시스템 접근으로부터 데이터나 서비스를 보호하기 위하여 사용자 인증과 접근 권한 제어 요소를 기술한다.		
요구사항 내용	1. 시스템의 접근은 등록된 아이디를 통해서만 가능하도록 한다. 2. 모든 개인정보의 저장, 전송 시에는 암호화, 복호화를 적용하여 논리적, 물리적으로 정보를 보호해야 한다. 3. Server 내부의 DB는 관리자 권한을 부여받은 아이디로만 접근이 가능해야 하며 인증되지 않은 사용자가 접근 시 접근할 수 없음을 알리고 접근을 불가하도록 한다.		
중요도	상	위험도	상
품질속성	상호운용성, 사용성		
평가 방법	1. 관리자만 보안 관련 시스템에 접근이 가능한지 확인한다. 2. 데이터의 보호 기능이 수행되는지 확인한다.		
관련 요구사항			
요구사항 출처			



## 8. 제약사항 (Constraints Requirement)

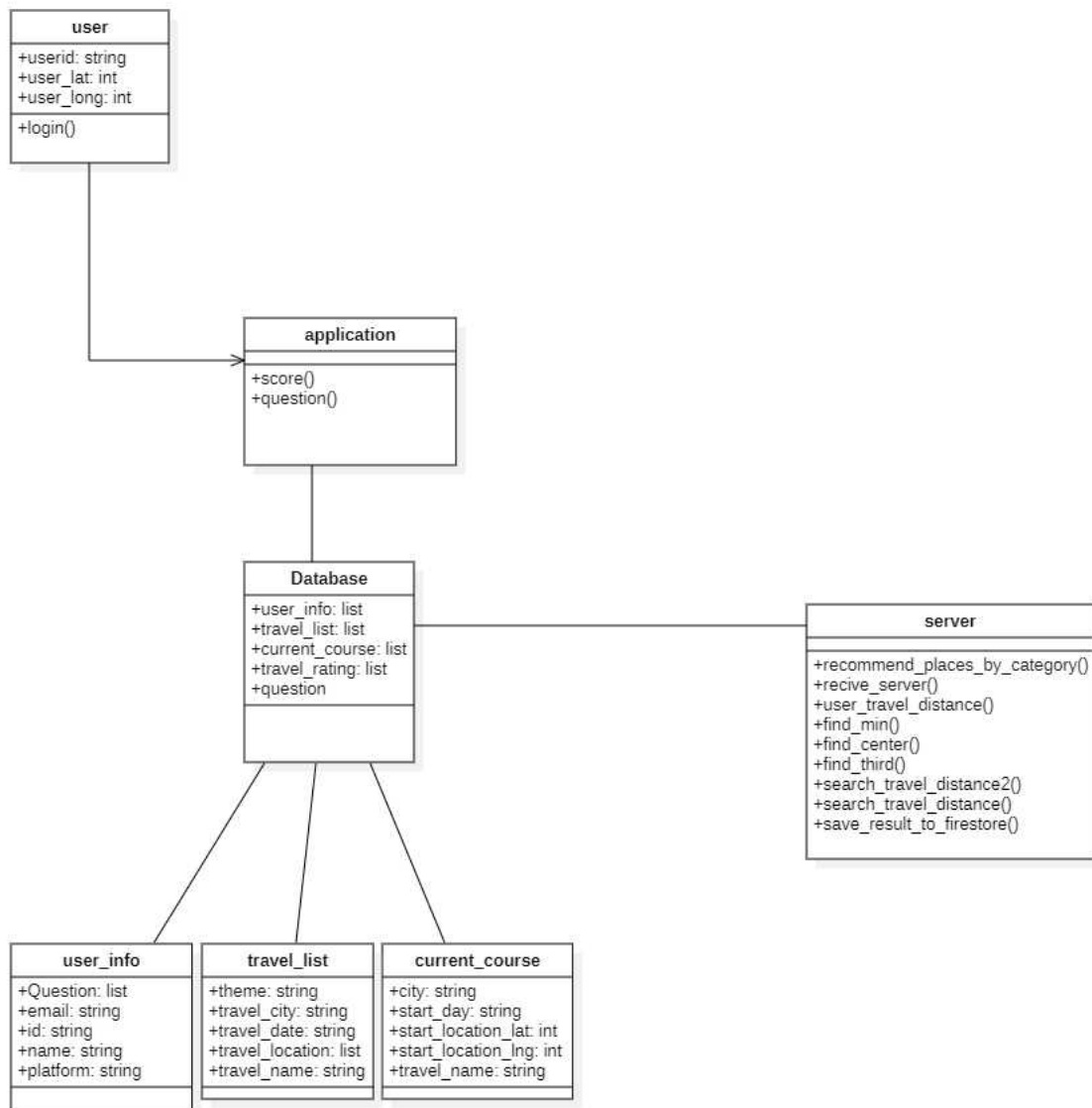
요구사항 번호	COR-001	요구사항 유형	제약사항
요구사항 이름	법적 적합성		
요구사항 개요	시스템이 서비스를 제공할 때 준수해야 할 법규 관련 제약사항을 기술한다.		
요구사항 내용	<p>1.1 시스템은 개인정보보호를 위하여 '개인정보 보호법, 시행일: 2011. 9.30)'을 준수해야 한다.</p> <ul style="list-style-type: none"> <li>· 시스템은 사용자의 개인정보 취급 시 사용자의 동의를 얻어야 한다.</li> <li>· 시스템은 운영에 필요한 최소한의 개인정보만을 취급하여야 한다.</li> <li>· 시스템은 취급한 개인정보에 대한 접근 권한을 업무수행에 필요한 최소한의 인원으로 제한하여야 한다.</li> <li>· 시스템은 개발 과정에서 저작권을 침해하지 말아야 한다.</li> </ul> <p>오픈소스 출처의 요소 또한 라이선스를 지켜야 한다.</p> <ul style="list-style-type: none"> <li>· 시스템은 사용자의 삭제 시 지체 없이 해당 개인정보를 파기하여야 한다.</li> </ul>		
중요도	상	위험도	상
품질속성	상호운용성, 사용성		
평가 방법	1. 시스템은 개인정보보호법에 따라 개인정보를 처리하는지 확인한다.		
관련 요구사항	FUR-003, FUR-004		
요구사항 출처			

## V. 시스템 설계서



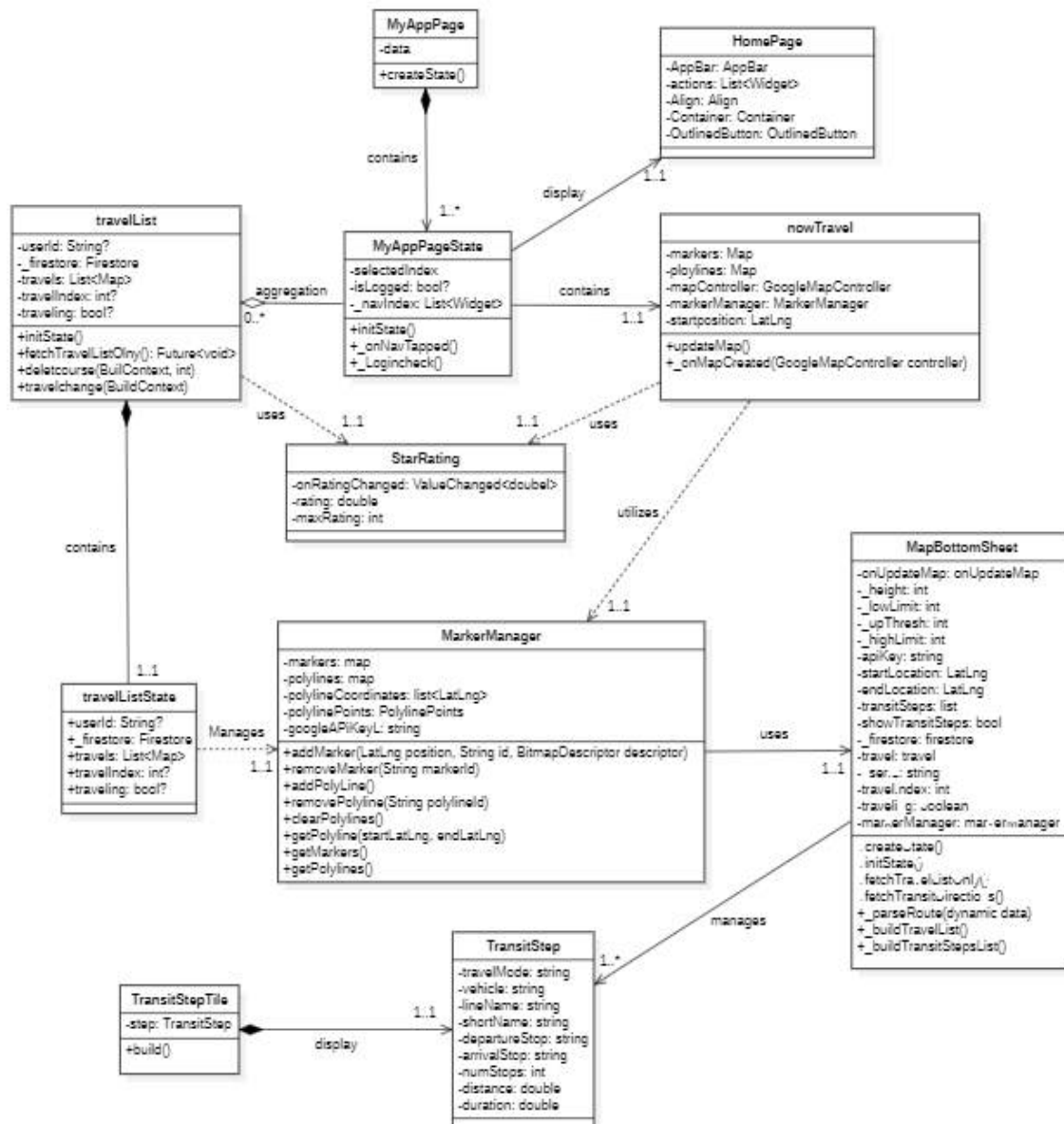
## 1. 소프트웨어 아키텍처 설계 (SW Architecture Design)

### 1.1 시스템 전체 Class Diagram

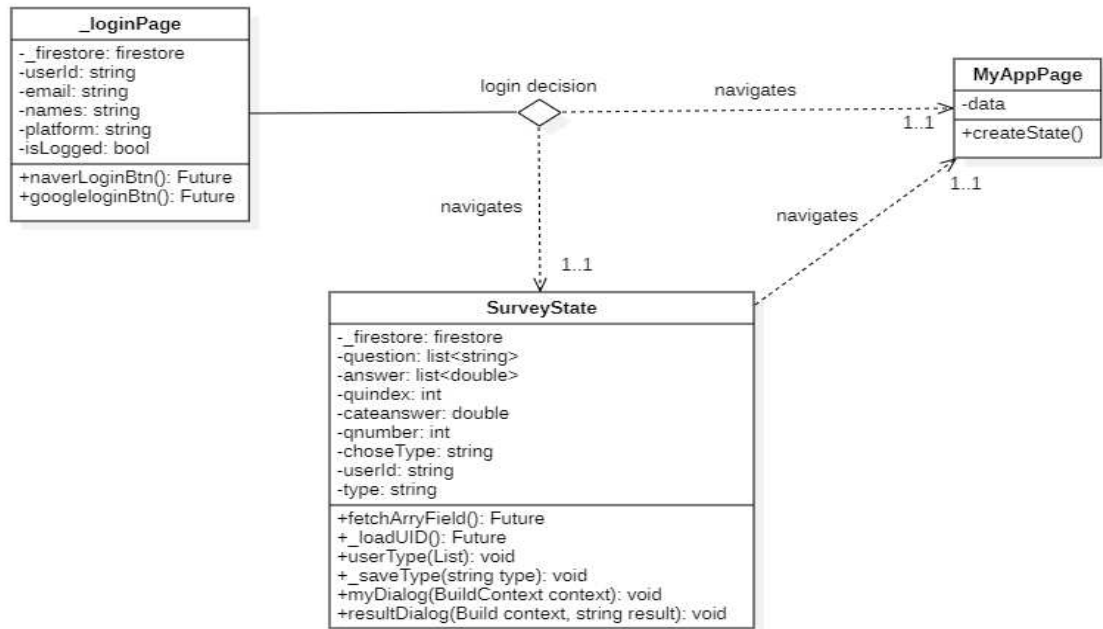


## 1. 소프트웨어 아키텍처 설계 (SW Architecture Design)

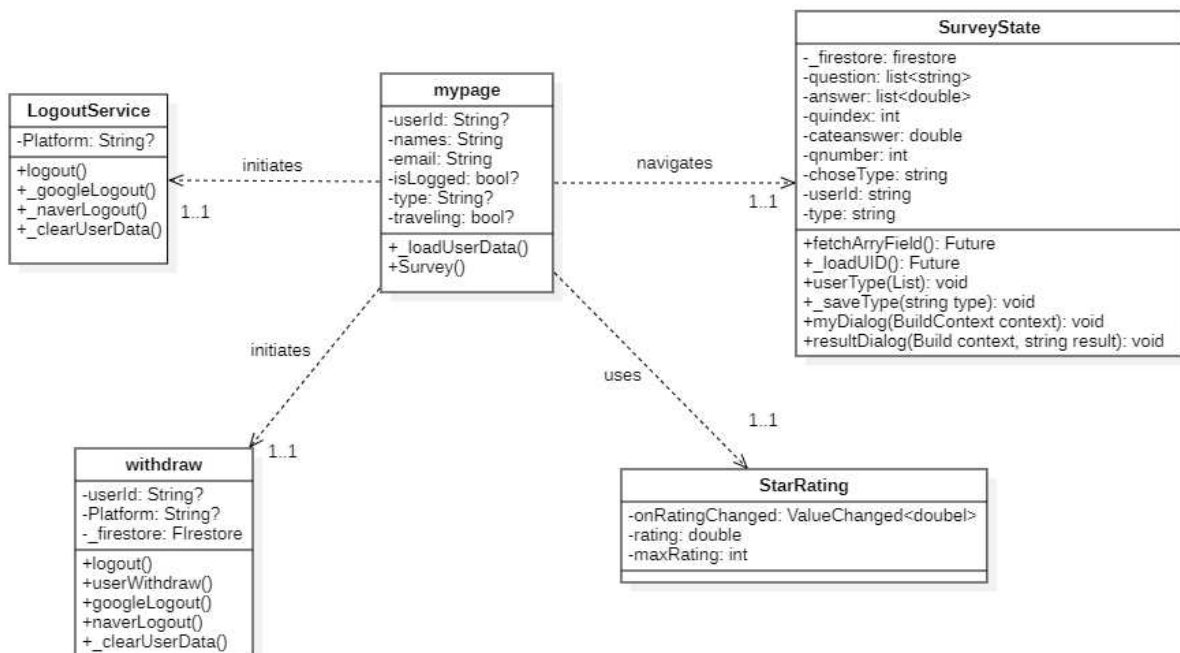
## 1.2 application main Class Diagram



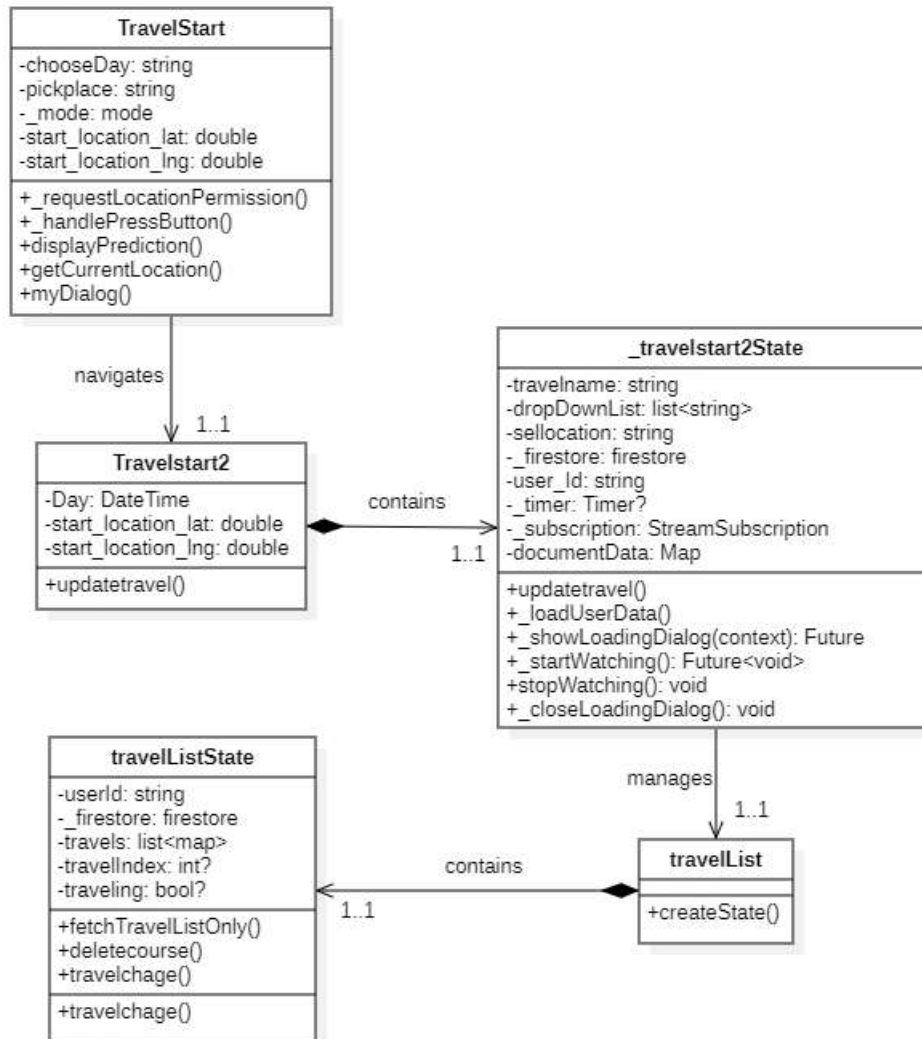
### 1.3 application login Class Diagram



### 1.4 application my page Class Diagram



## 1.5 application travel start Class Diagram

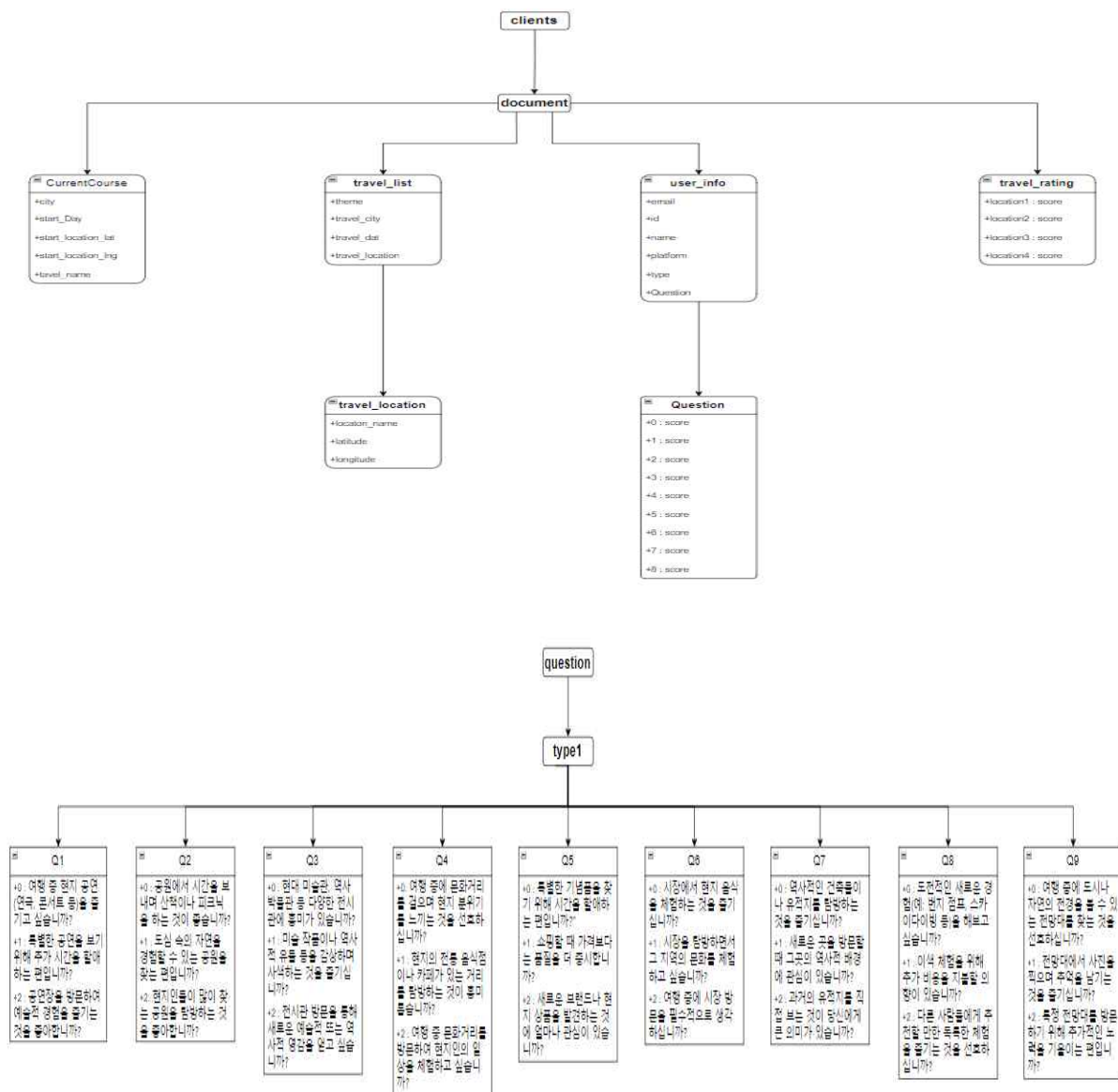


## 2. 데이터베이스 설계 (Database Design)

### 2.1. 객체 관계 다이어그램 (Entity-Relation Diagram)

#### 1. 엔티티 관계도

시스템이름	원데이 서울	서브시스템이름	원데이 서울 DB
ERD ID	Oneday Seoul_DB	ERD 이름	Oneday Seoul_DB_ERD



## 2. 데이터베이스 설계 (Database Design)

### 2.2 논리적 데이터 설계

테이블이름		사용자 정보						
Entity ID		user_info		Entity 이름		사용자 정보		
속성명	데이터타입	길이	NULL 허용	기본키	외래키	인덱스	기본값	제약조건
id	string	50	n	n				
document	string	50	n	n	y			
Question	array	50	y	n				
email	string	50	y	n				
name	string	50	y	n				
platform	string	50	y	n				
type	string	50	y	n				



테이블이름	현재 코스							
Entity ID	CurrentCourse			Entity 이름		현재 여행		
속성명	데이터타입	길이	NULL 허용	기본키	외래키	인덱스	기본값	제약조건
document	string	50	n	n	y			
city	string	50	y	n				
start_Day	timestamp	50	y	n				
start_location_lat	number	50	y	n				
start_location_lng	number	50	y	n				
travel_name	string	50	y	n				

테이블이름	여행 리스트							
Entity ID	travel_list			Entity 이름		여행 리스트		
속성명	데이터타입	길이	NULL 허용	기본키	외래키	인덱스	기본값	제약조건
travel_location	string	50	n	n				
document	string	50	n	n	y			
theme	string	50	y	n				
travel_city	string	50	y	n				
travel_date	string	50	y	n				

테이블이름		여행 점수						
Entity ID		travel_rating		Entity 이름		여행 점수		
속성명	데이터타입	길이	NULL 허용	기본키	외래키	인덱스	기본값	제약조건
document	string	50	n	n	y			
location1	number	50	n	n				
location2	number	50	n	n				
location3	number	50	n	n				
location4	number	50	n	n				

테이블이름		여행 위치 인덱스						
Entity ID		travel_location		Entity 이름		여행 위치 인덱스		
속성명	데이터타입	길이	NULL 허용	기본키	외래키	인덱스	기본값	제약조건
travel_location	map	50	n	n	y			
travel_name	string	50	y	n				
latitude	string	50	y	n				
longitude	string	50	y	n				

테이블이름		설문 결과						
Entity ID		Question		Entity 이름		여행 점수		
속성명	데이터타입	길이	NULL 허용	기본키	외래키	인덱스	기본값	제약조건
Question	array	50	n	n	y			
1	number	50	n	n				
2	number	50	n	n				
3	number	50	n	n				
4	number	50	n	n				
5	number	50	n	n				
6	number	50	n	n				
7	number	50	n	n				
8	number	50	n	n				
9	number	50	n	n				

테이블이름		설문1						
Entity ID		Q1		Entity 이름		설문1		
속성명	데이터 타입	길이	NULL 허용	기본키	외래키	인덱스	기본값	제약조건
type1	string	50	n	n	y			
0	string	50	n	n				
1	string	50	n	n				
2	string	50	n	n				

테이블이름		설문2						
Entity ID		Q2		Entity 이름		설문2		
속성명	데이터 타입	길이	NULL 허용	기본키	외래키	인덱스	기본값	제약조건
type1	string	50	n	n	y			
0	string	50	n	n				
1	string	50	n	n				
2	string	50	n	n				

테이블이름		설문3						
Entity ID		Q3		Entity 이름		설문3		
속성명	데이터 타입	길이	NULL 허용	기본키	외래키	인덱스	기본값	제약조건
type1	string	50	n	n	y			
0	string	50	n	n				
1	string	50	n	n				
2	string	50	n	n				

테이블이름		설문4						
Entity ID		Q4		Entity 이름		설문4		
속성명	데이터 타입	길이	NULL 허용	기본키	외래키	인덱스	기본값	제약조건
type1	string	50	n	n	y			
0	string	50	n	n				
1	string	50	n	n				
2	string	50	n	n				

테이블이름		설문5						
Entity ID		Q5		Entity 이름		설문1		
속성명	데이터 타입	길이	NULL 허용	기본키	외래키	인덱스	기본값	제약조건
type1	string	50	n	n	y			
0	string	50	n	n				
1	string	50	n	n				
2	string	50	n	n				

테이블이름		설문6						
Entity ID		Q6		Entity 이름		설문6		
속성명	데이터 타입	길이	NULL 허용	기본키	외래키	인덱스	기본값	제약조건
type1	string	50	n	n	y			
0	string	50	n	n				
1	string	50	n	n				
2	string	50	n	n				

테이블이름		설문7						
Entity ID		Q7		Entity 이름		설문7		
속성명	데이터 타입	길이	NULL 허용	기본키	외래키	인덱스	기본값	제약조건
type1	string	50	n	n	y			
0	string	50	n	n				
1	string	50	n	n				
2	string	50	n	n				

테이블이름		설문8						
Entity ID		Q8		Entity 이름		설문8		
속성명	데이터 타입	길이	NULL 허용	기본키	외래키	인덱스	기본값	제약조건
type1	string	50	n	n	y			
0	string	50	n	n				
1	string	50	n	n				
2	string	50	n	n				

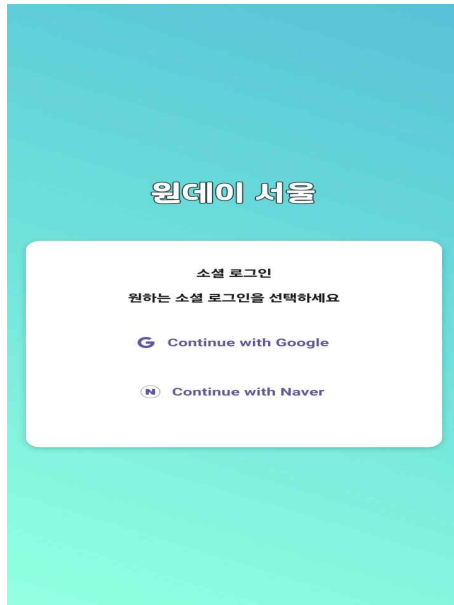
테이블이름		설문9						
Entity ID		Q9		Entity 이름		설문9		
속성명	데이터 타입	길이	NULL 허용	기본키	외래키	인덱스	기본값	제약조건
type1	string	50	n	n	y			
0	string	50	n	n				
1	string	50	n	n				
2	string	50	n	n				



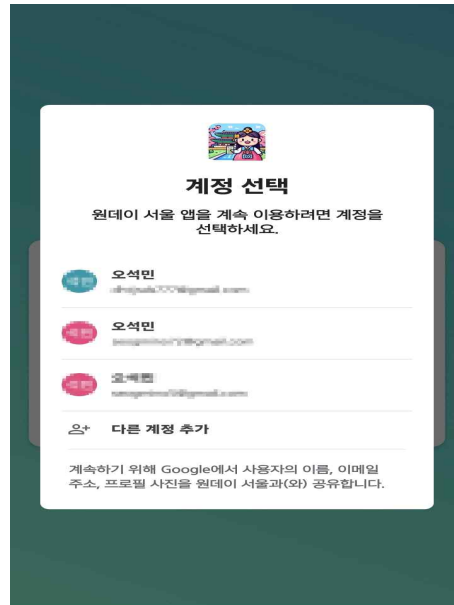
### 3. 사용자 인터페이스 설계 (User Interface Design)

#### 3.1 앱 화면 설계

##### 1.1 앱 로그인 화면



##### 1.2 계정 선택 화면



##### 1.3 설문 화면

#### travel test

Question 1/45

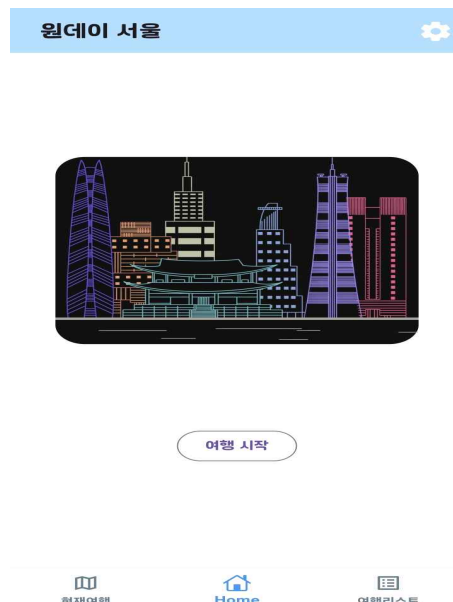
관람지 조사..

1(매우 아니다), 2(아니다), 3(보통), 4(그렇다), 5(매우 그렇다)  
여행 중 현지 공연(연극, 콘서트 등)을 즐기고 싶습니까?

1 2 3 4 5

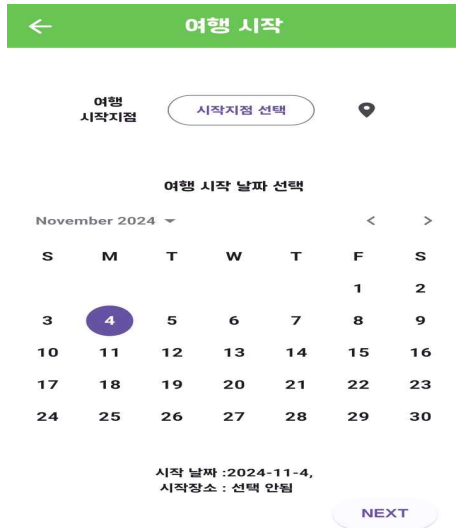
Next Question

##### 1.4 앱 회원가입

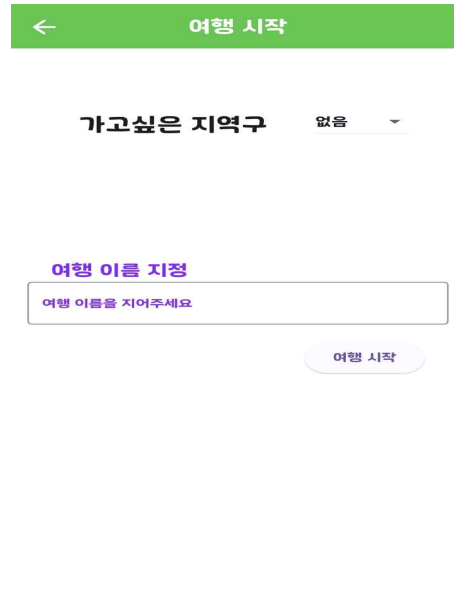


### 3. 사용자 인터페이스 설계 (User Interface Design)

#### 1.5 여행 시작 화면(1)



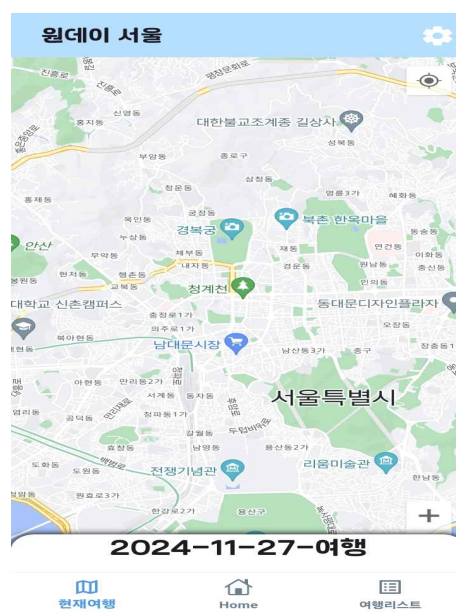
#### 1.6 여행 시작 화면(2)



#### 1.7 여행 리스트 화면

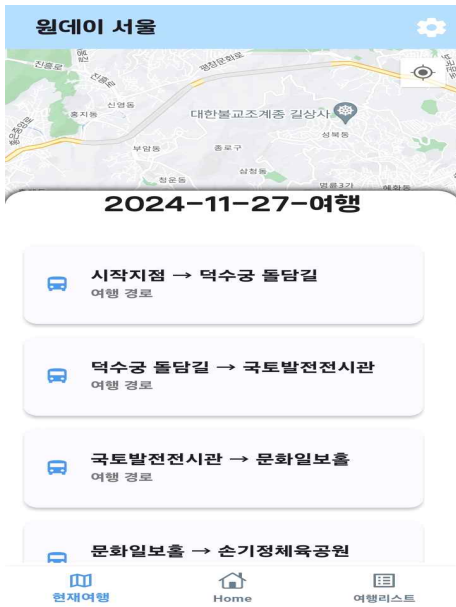


#### 1.8 현재 여행 화면

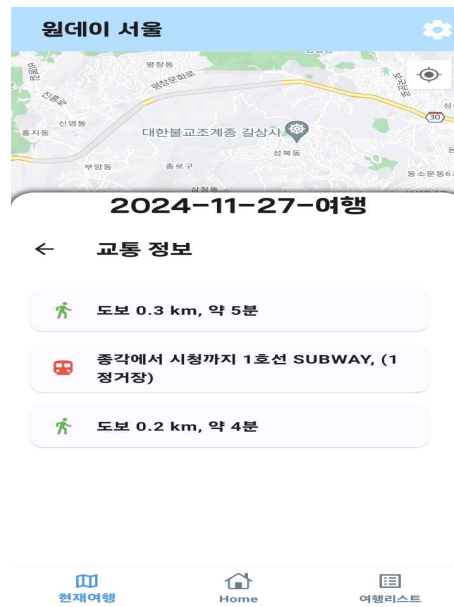


### 3. 사용자 인터페이스 설계 (User Interface Design)

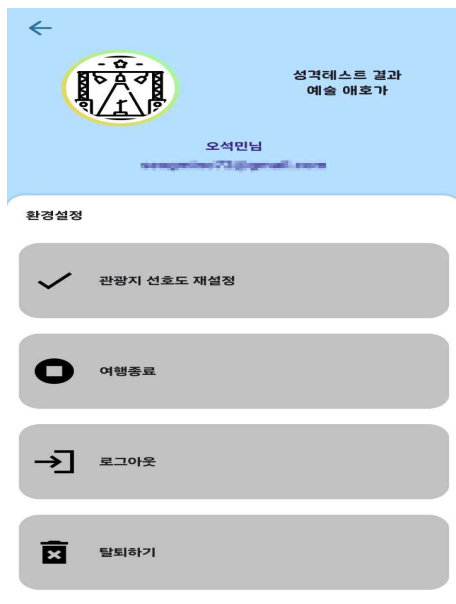
#### 1.9 현재 여행지 목록



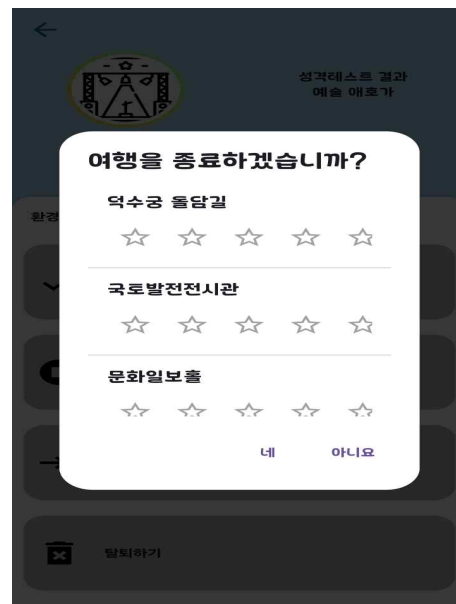
#### 1.10 여행 경로



#### 1.11 설정 화면

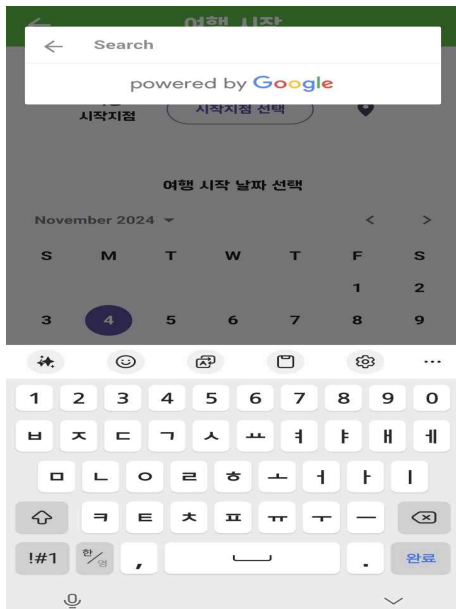


#### 1.12 여행 종료 화면

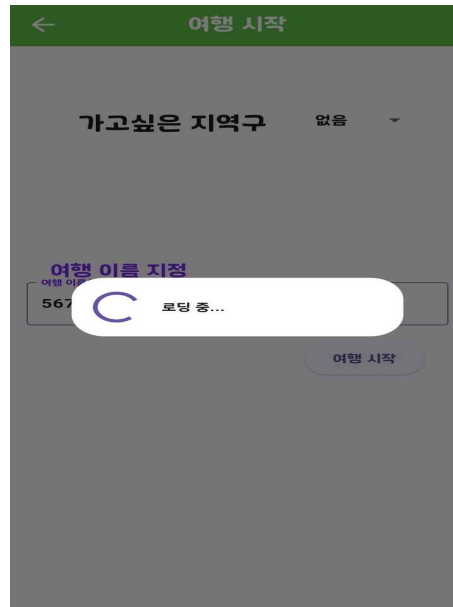


### 3. 사용자 인터페이스 설계 (User Interface Design)

1.13 시작 지점 선택



1.14 로딩 중 화면

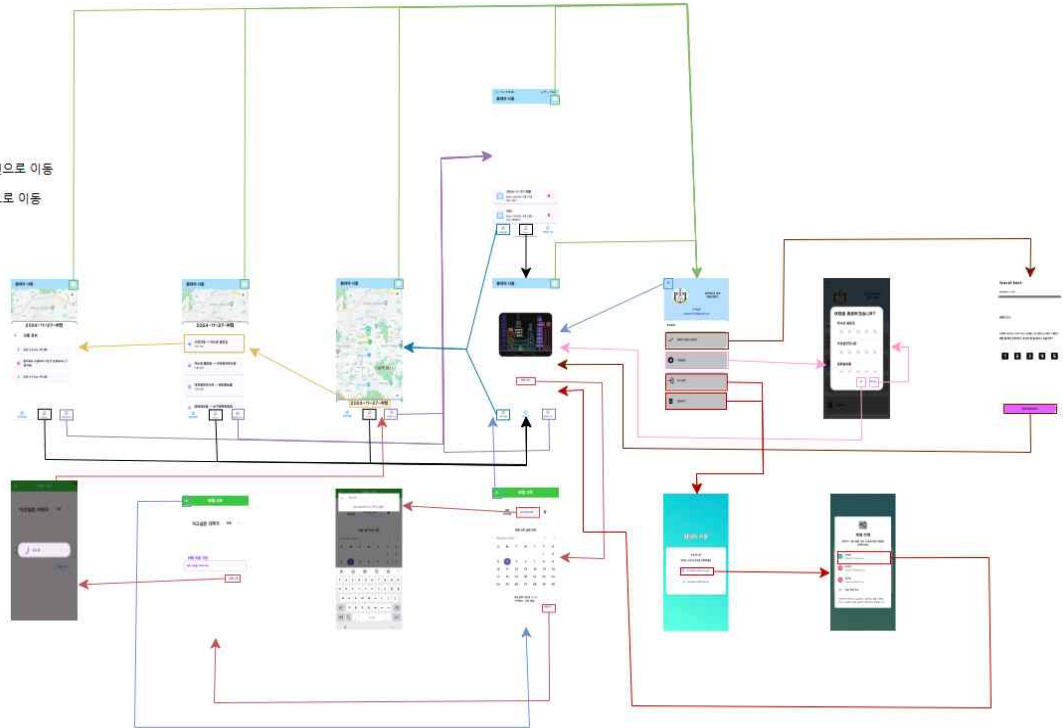


## 3. 사용자 인터페이스 설계 (User Interface Design)

### 3.3 화면 내비게이션

색깔 별 경로

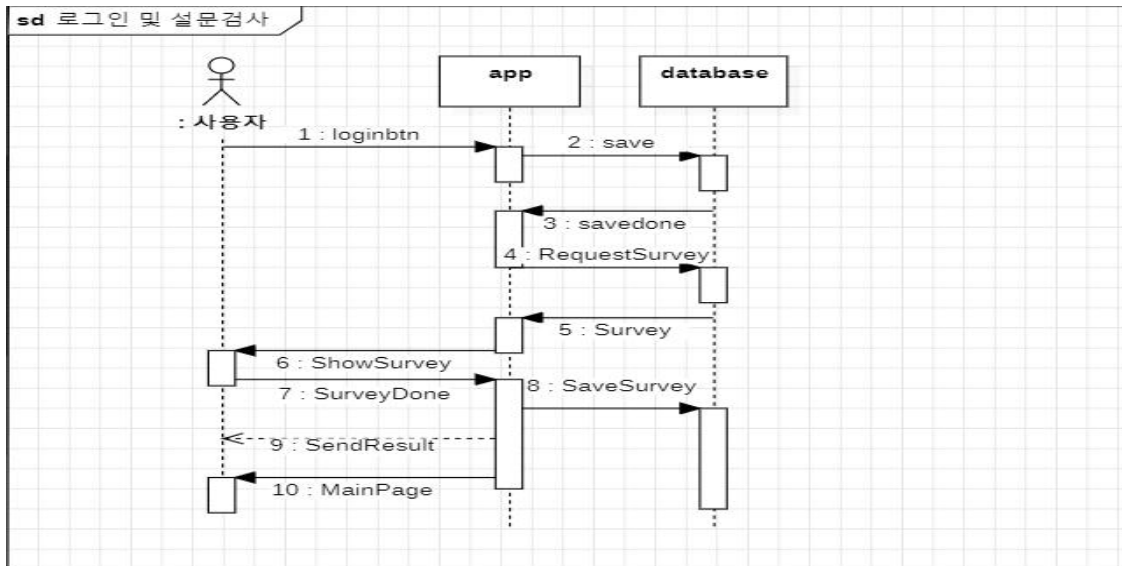
검정색 : 홈 화면으로 이동  
 보라색 : 여행 일정 리스트로 이동  
 노란색 : 여행 일정 상세 화면으로 이동  
 파란색 : 여행 일정 상세 화면으로 이동  
 자홍색 : 여행 일정 상세 화면으로 이동  
 하늘색 : 여행 일정 상세 화면으로 이동  
 연두색 : 로그인 화면으로 이동  
 분홍색 : 로그인 화면으로 이동  
 갈색 : 관리자 계정 설정 화면으로 이동



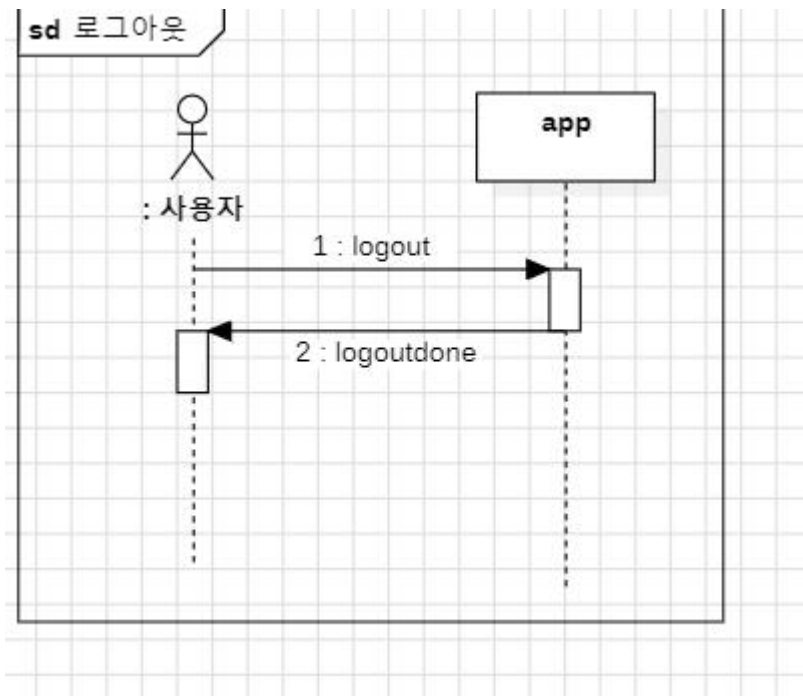
## IV. 소프트웨어 컴포넌트 설계 (SW Component Design)

### 4.1 Sequence Diagram

#### 1) 로그인 및 설문조사



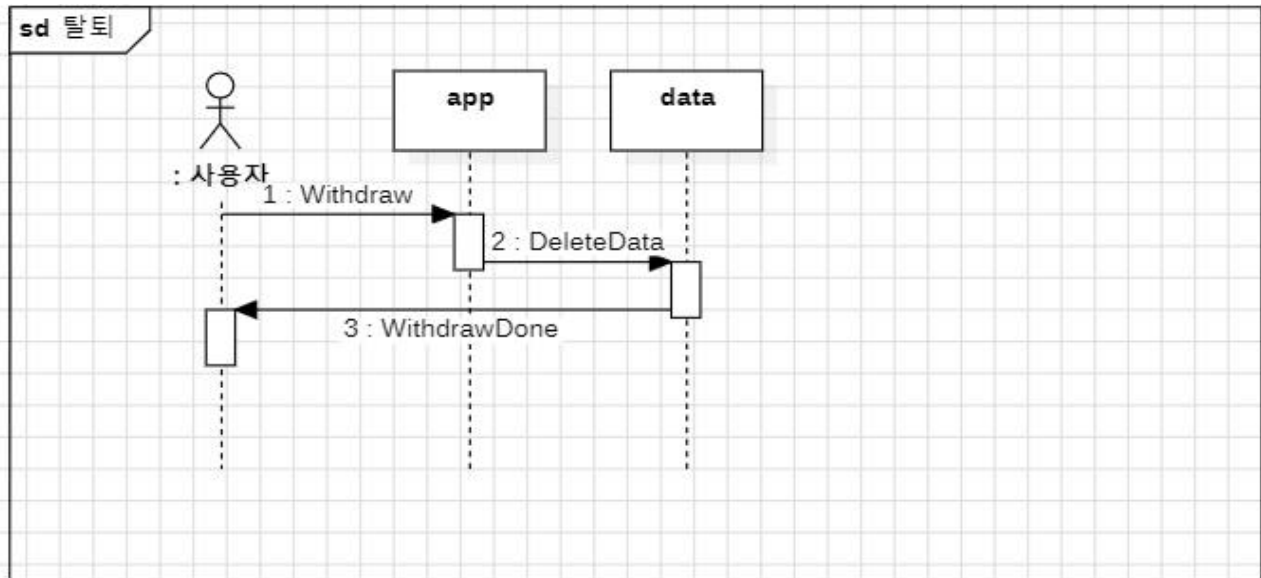
#### 2) 로그아웃



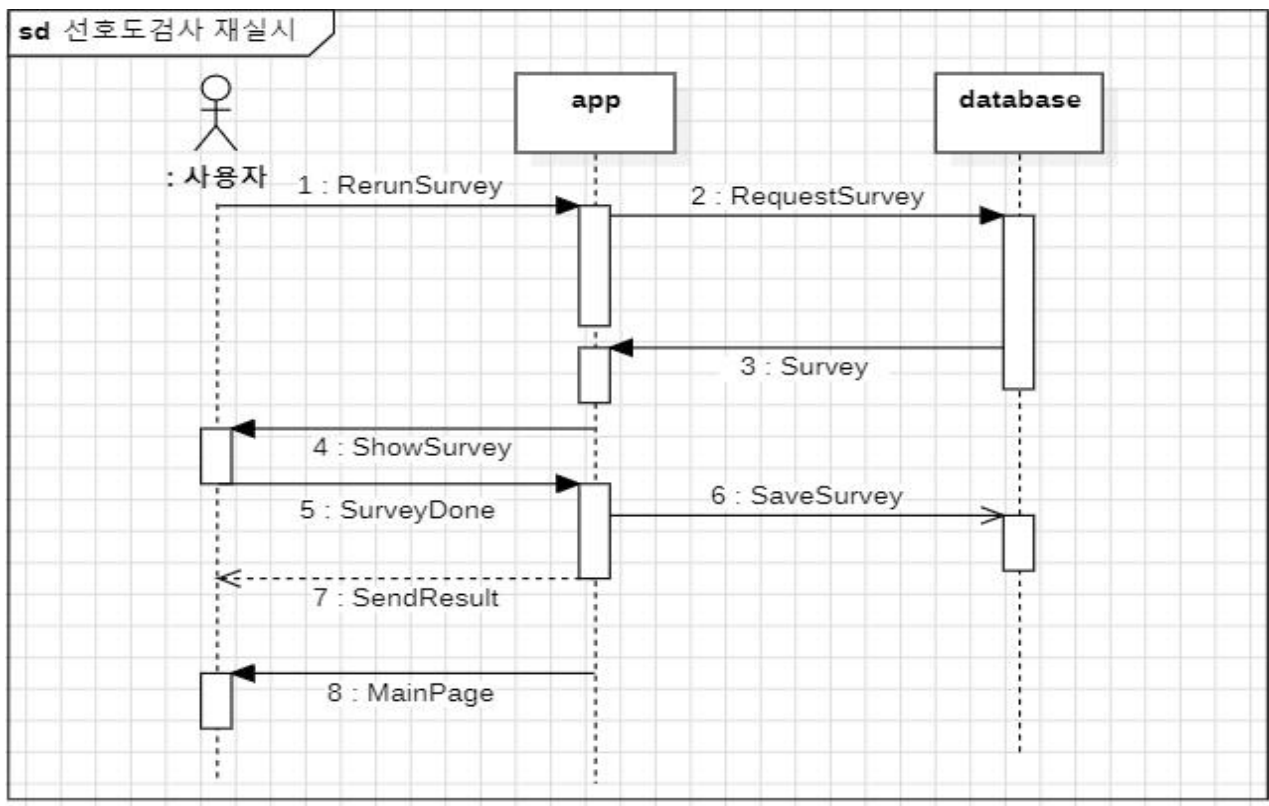
## IV. 소프트웨어 컴포넌트 설계 (SW Component Design)

### 4.1 Sequence Diagram

#### 3) 탈퇴 하기



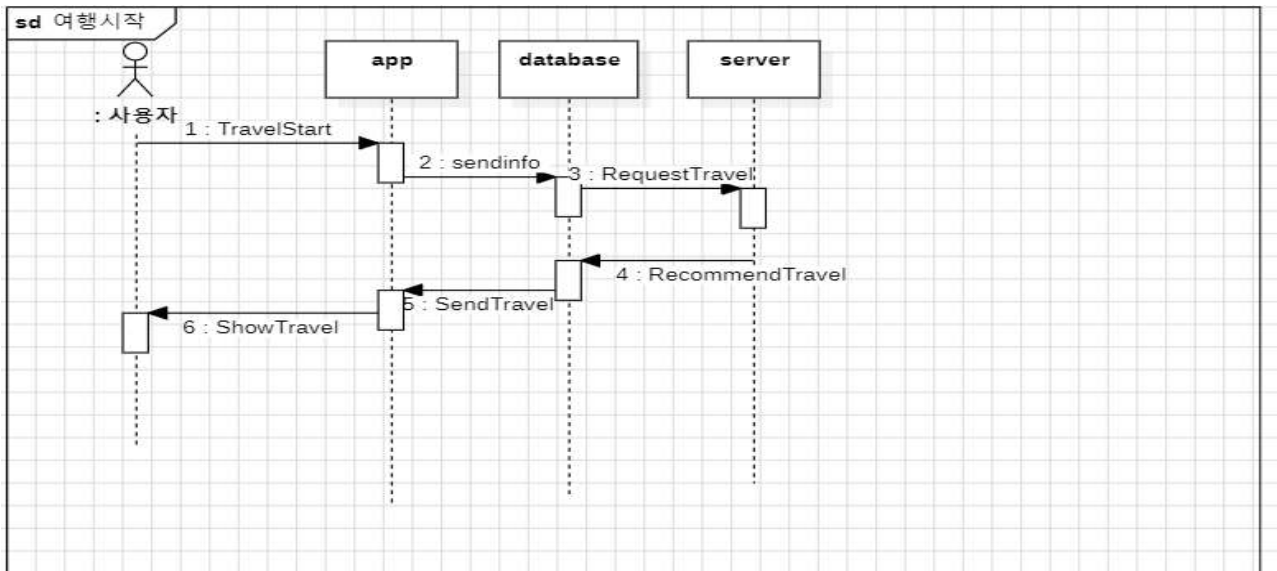
#### 4) 관광지 선호도 재설정



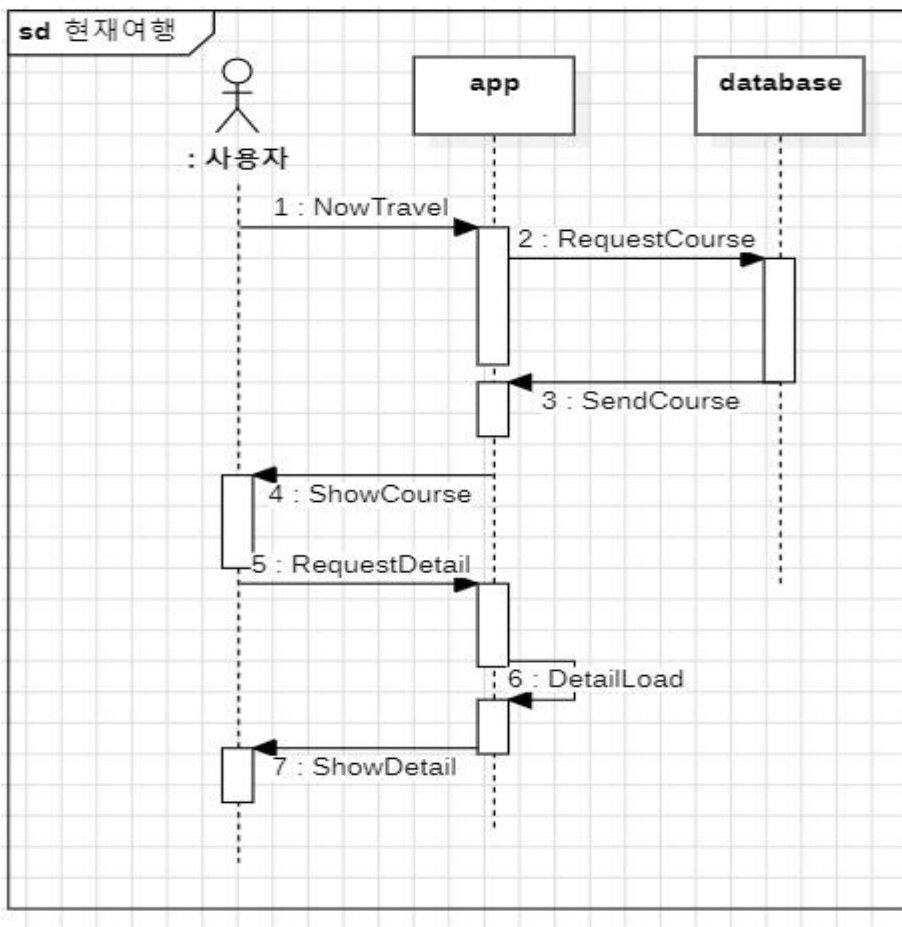
## IV. 소프트웨어 컴포넌트 설계 (SW Component Design)

### 4.1 Sequence Diagram

#### 5) 여행 시작



#### 6) 현재 여행

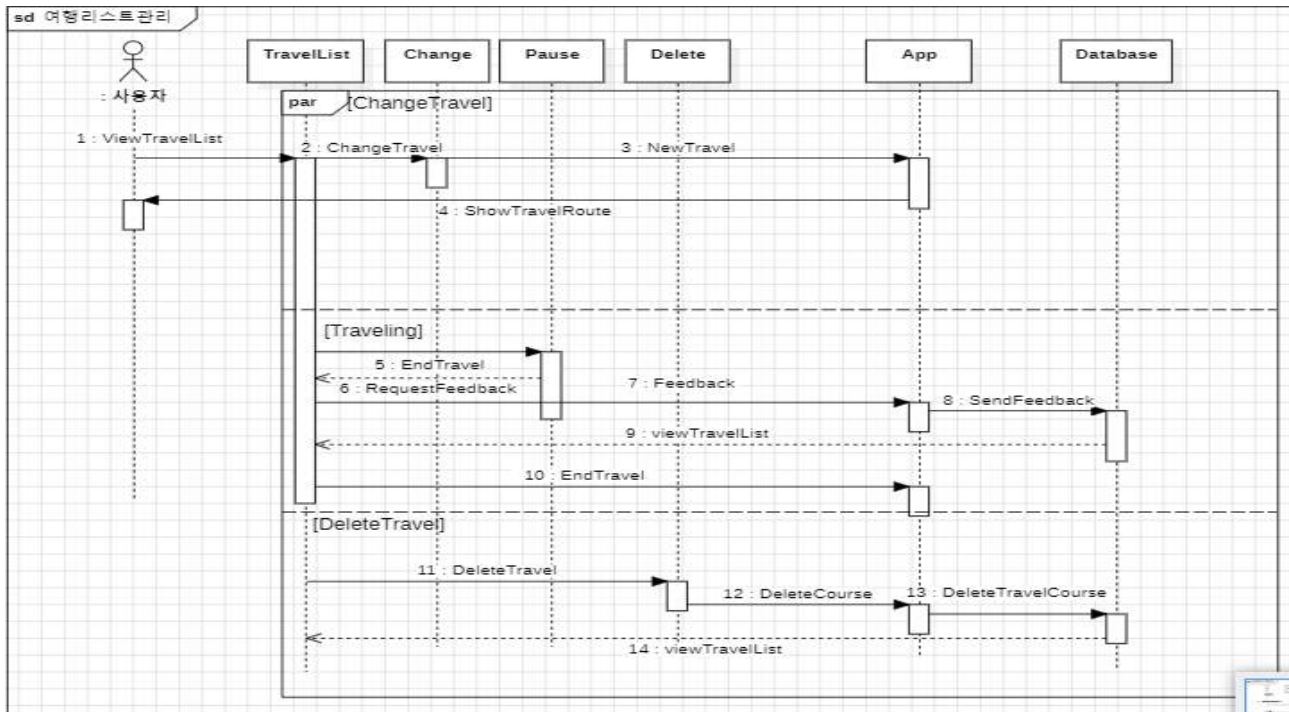




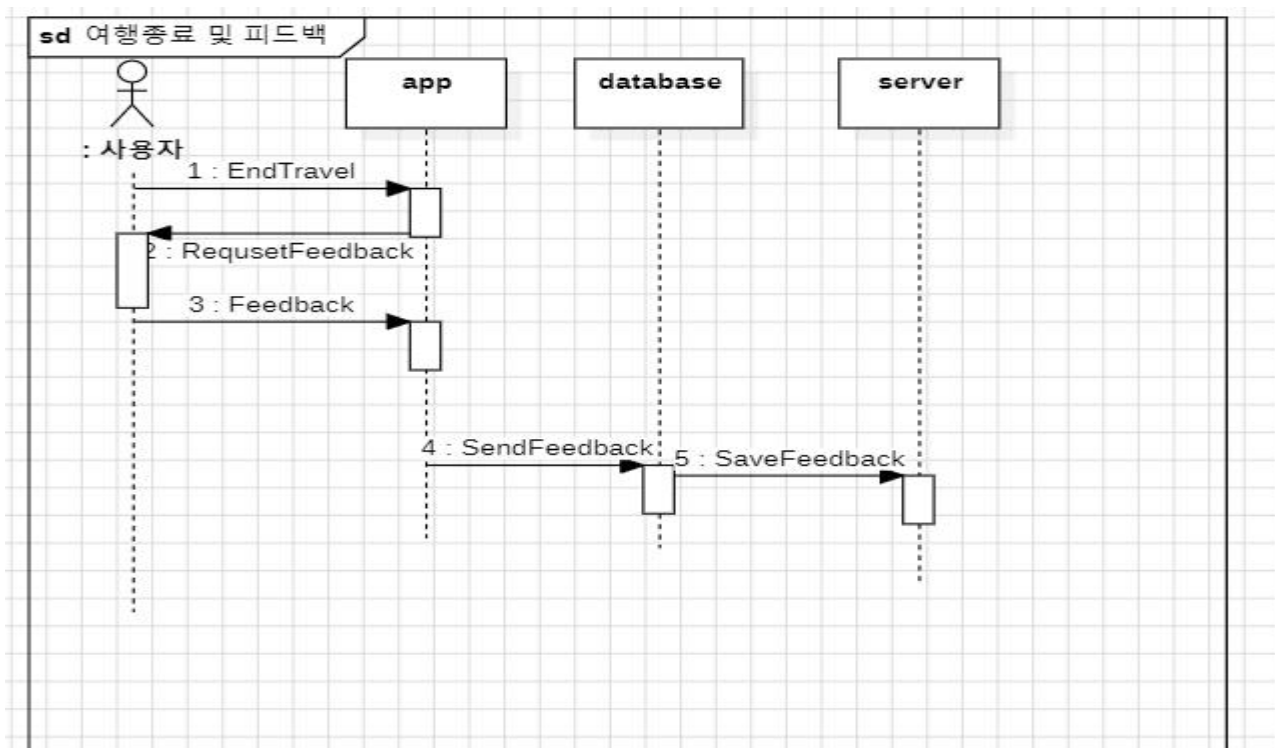
## IV. 소프트웨어 컴포넌트 설계 (SW Component Design)

### 4.1 Sequence Diagram

#### 7) 여행 리스트 관리



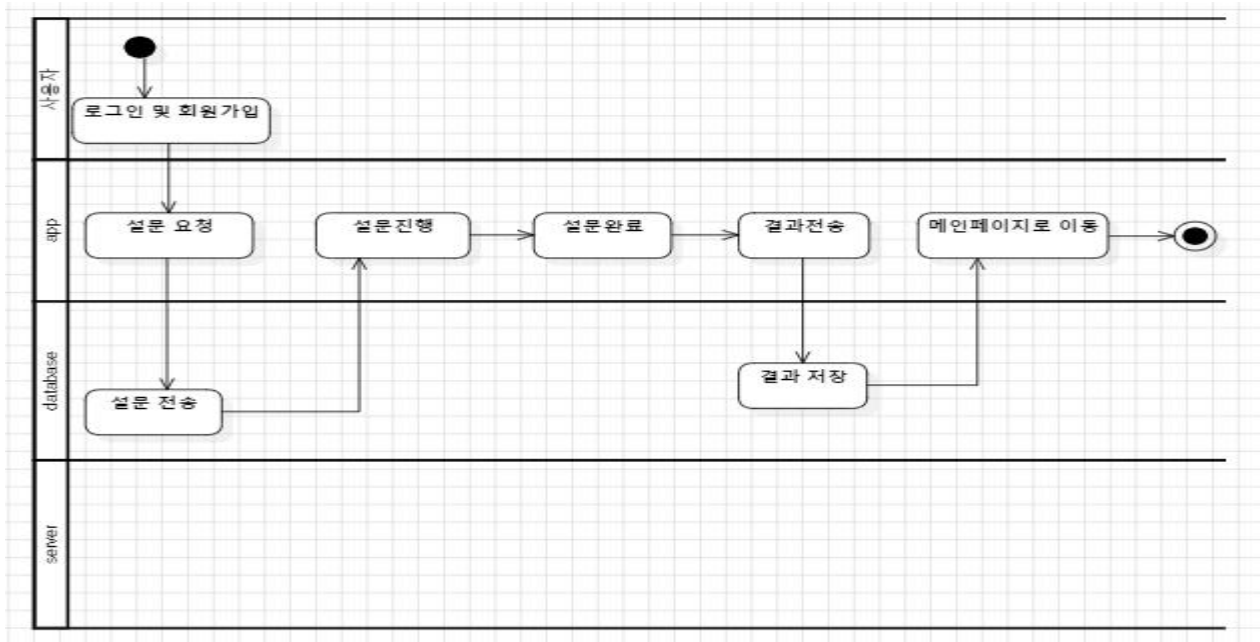
#### 8) 여행 종료 및 피드백



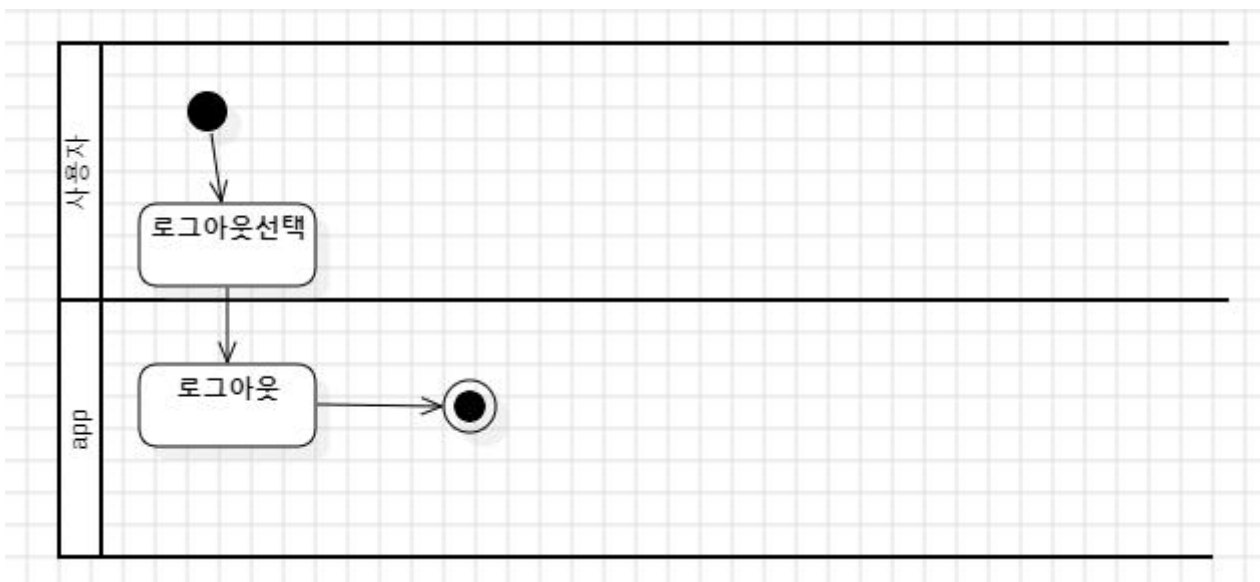
## IV. 소프트웨어 컴포넌트 설계 (SW Component Design)

### 4.2 Activity Diagram

#### 1) 로그인 및 설문조사



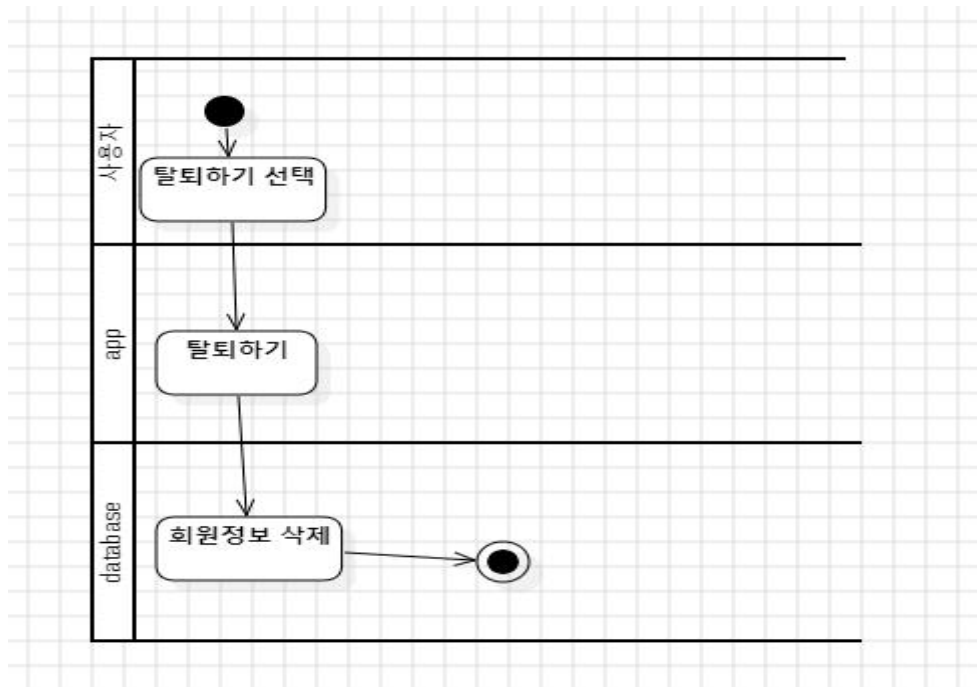
#### 2) 로그 아웃



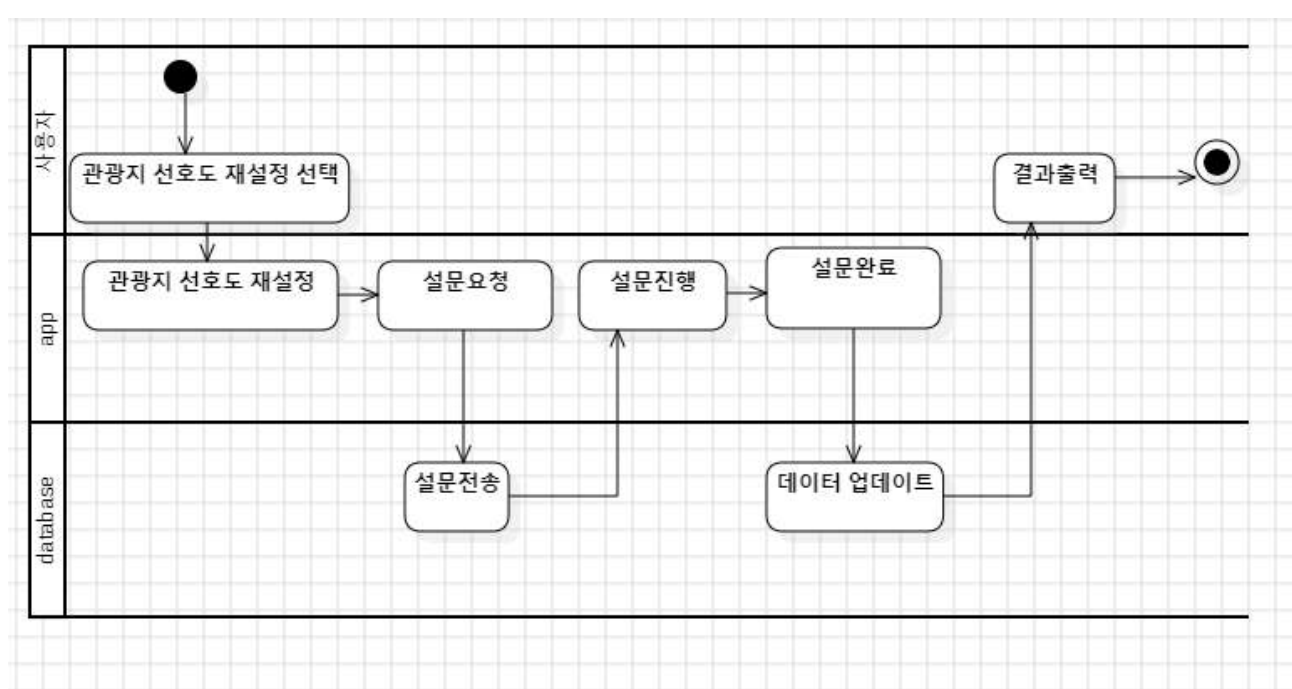
## IV. 소프트웨어 컴포넌트 설계 (SW Component Design)

### 4.2 Activity Diagram

#### 3) 탈퇴하기



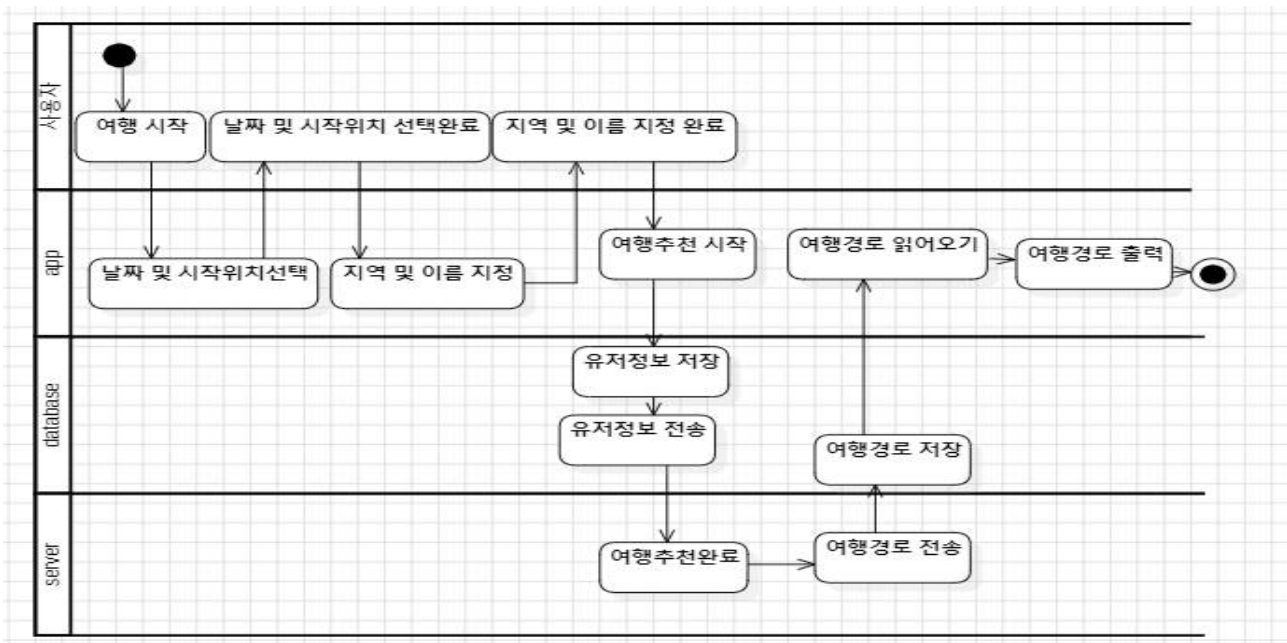
#### 4) 관광지 선호도 재설정



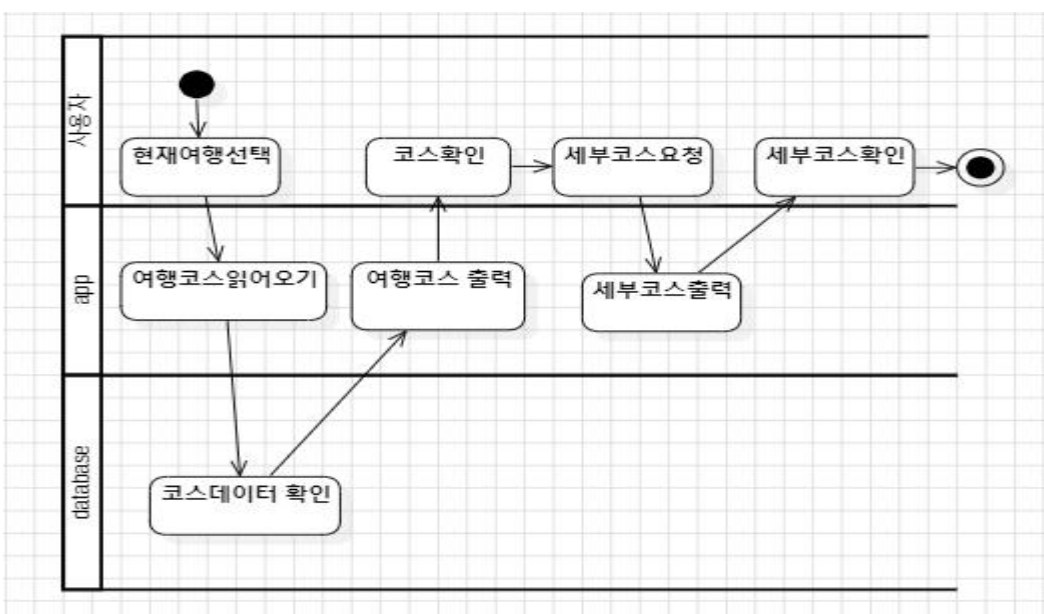
## IV. 소프트웨어 컴포넌트 설계 (SW Component Design)

### 4.2 Activity Diagram

#### 5) 여행 시작



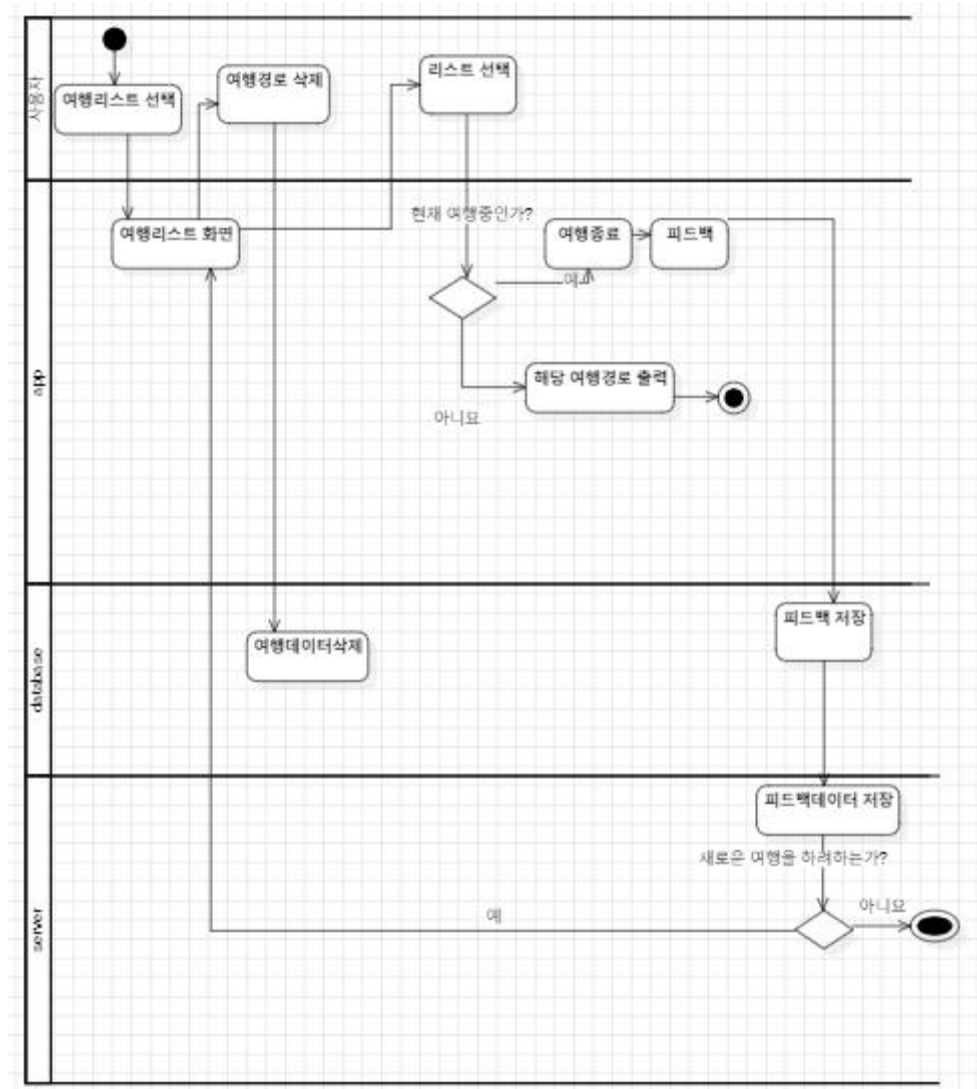
#### 6) 현재 여행



## IV. 소프트웨어 컴포넌트 설계 (SW Component Design)

### 4.2 Activity Diagram

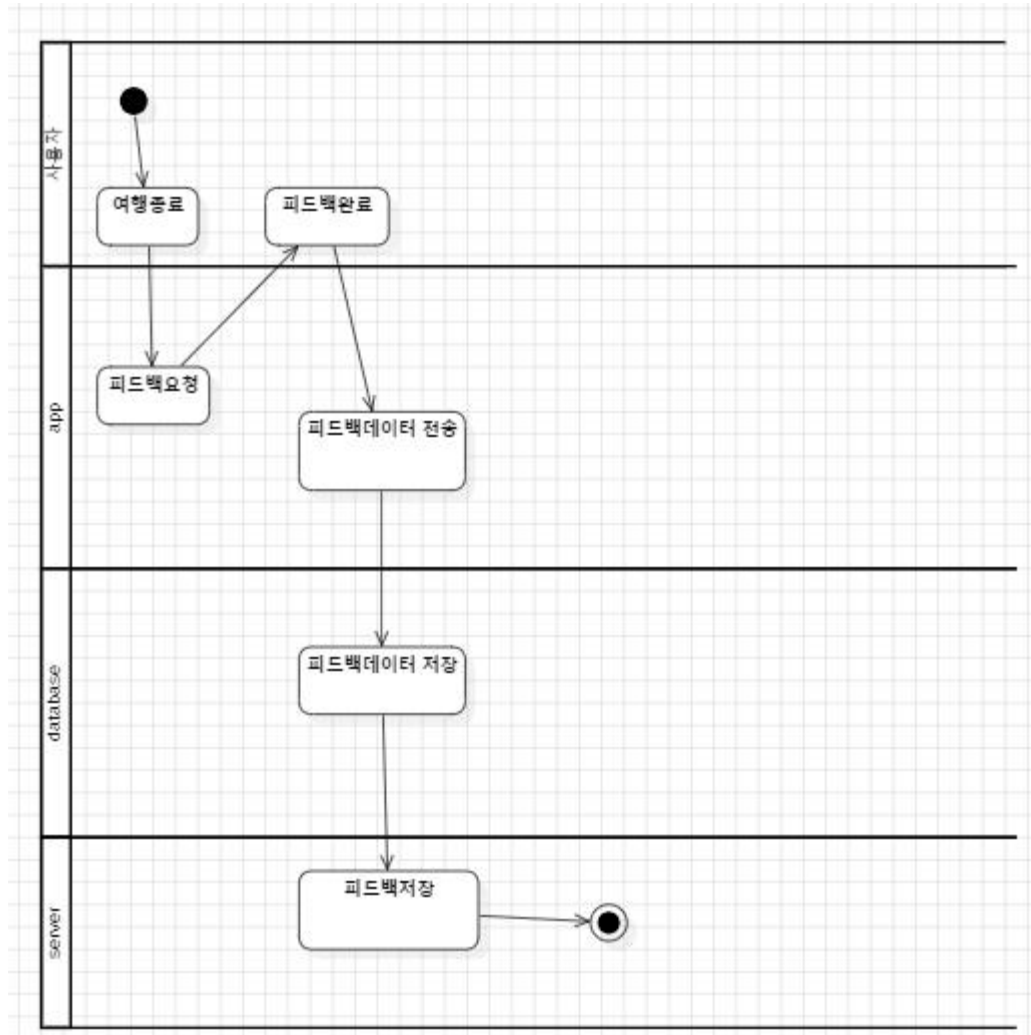
#### 7) 여행 리스트 관리



## IV. 소프트웨어 컴포넌트 설계 (SW Component Design)

### 4.2 Activity Diagram

#### 8) 여행 종료 및 피드백



## VI. 시스템 테스트 결과서

## I. 소프트웨어 테스트 종괄표 (SW Test Overall List)

NO	대분류	중분류	소분류	내용	기대 결과	실행 결과	비고
1	메인 화면	여행 시작	여행 시작 화면	메인 화면에서 여행 시작 버튼을 선택  현재 여행이 종료 가 안 된 상태면 별점을 먼저 입력	사용자 정보를 입력하는 화면을 이동  별점 입력받는 화면으로 이동	P	
2			여행 정보 입력 화면	여행 시작 날짜, 여행 시작 지점을 입력  여행 이름, 가고 싶은 지역구를 입력 후	여행 시작 날짜, 여행 시작 지점, 정보 저장 여행 이름, 가고 싶은 지역구 저장, 여행지 생성	P	
3			여행 생성 화면	여행지 생성 선택  여행 이름이 있는 바 위로 드래그	추천 여행지들 순서대로 출력 대중교통을 활용해서 경로를 출력	P	
4		현재 여행	여행리스트	여행지 선택	현재 여행에 선택된 여행지의 정보를 넘김,  여행을 종료시키고 별점을 받음	P	
5			여행지 삭제	여행지에 삭제 버튼을 선택	해당 여행지를 삭제시킴	P	
8		환경 설정	관광지 선호도	관광지 선호도 재설정	관광지 선호도 재설정 버튼 선택	P	
9				관광지의 선호도를 재설정 버튼 선택	관광지 선호도 재설정 화면을 보여준다.	P	
10			여행 종료	여행 종료	여행 종료 버튼 선택	P	
12				여행 종료 버튼 선택	별점을 받는 창을 보여준다.	P	
13			로그아웃	로그 아웃	로그아웃 버튼 선택	P	
14				로그아웃 버튼 선택	로그인 화면을 보여준다.	P	
15			탈퇴하 기	탈퇴하기	탈퇴하기 버튼 선택	P	
18				탈퇴하기 버튼 선택	로그인을 다시 하면 선호도 조사를 다시 보여준다.	P	



II. 소프트웨어 테스트 케이스 (SW Test Case)

Test Case ID	TC-001	Test Case 이름	로그인
Test Case 개요	사용자가 로그인한다.		
모듈 이름	로그인		
관련 요구사항			
사전 조건	사용자가 앱을 다운받은 상태		
테스트 환경	Galaxy s21 5G		
단계	테스트 절차	테스트 입력	예상 출력
1	1. Continue with Google 버튼을 선택 2. Continue with Naver 버튼을 선택	버튼 선택	1. 구글 계정을 선택하는 팝업이 나옴 그 후 설문 화면으로 이동 2. 네이버 계정에 로그인되고 설문 화면으로 이동
2	관광지 선호도 조사 시 점수를 입력하지 않고 Next Question 버튼 선택	버튼 선택	Error 질문 버튼을 눌러주세요. 팝업 표시
3	관광지 선호도 조사를 모두 하고 마지막 finish 버튼 선택	버튼 선택	선호도 결과를 보여주는 팝업을 표시
사후 조건	사용자 정보가 DB에 저장되어야 한다.		
작성자	김현호	작성일	2024.10.30
실행자	김현호	실행일	2024.10.30

Test Case ID	TC-002	Test Case 이름	여행 시작
Test Case 개요	사용자가 여행을 생성한다.		
모듈 이름	메인 화면		
관련 요구사항			
사전 조건	사용자가 로그인한 상태		
테스트 환경	Galaxy s21 5G		
단계	테스트 절차	테스트 입력	예상 출력
1	메인 화면에서 여행 시작 버튼을 선택	여행 시작 버튼 선택	여행 시작 지점과 여행 시작 날짜를 받는 화면 표시
2	여행 시작 지점과 여행 시작 날짜를 입력 후 next 버튼 선택	여행 시작 지점, 여행 시작 날짜, next	가고 싶은 지역구, 여행 이름을 입력하는 화면 표시
3	가고 싶은 지역구와 여행 이름을 입력 후 여행 시작 버튼 선택	가고 싶은 지역구, 여행 이름, 여행 시작	로딩 중 화면 출력
사후 조건	로딩이 종료되면 현재 여행에 설정한 여행 이름의 바 표시		
작성자	김현호	작성일	2024.10.30
실행자	김현호	실행일	2024.10.30

Test Case ID	TC-003	Test Case 이름	현재 여행
Test Case 개요	추천된 여행지와 경로를 보여준다.		
모듈 이름	메인 화면		
관련 요구사항			
사전 조건	사용자가 여행을 생성하고 로딩이 끝난 상태		
테스트 환경	Galaxy s21 5G		
단계	테스트 절차	테스트 입력	예상 출력
1	설정된 여행 이름의 바를 드래그 한다.	바 드래그	추천된 여행지와 여행 순서를 표시
2	여행 순서가 나와 있는 블록을 선택	블록 선택	여행지 간의 경로를 안내 표시
사후 조건			
작성자	김현호	작성일	2024.10.30
실행자	김현호	실행일	2024.10.30

Test Case ID	TC-004	Test Case 이름	여행리스트
Test Case 개요	여행리스트를 관리한다.		
모듈 이름	메인 화면		
관련 요구사항			
사전 조건	사용자가 여행을 선택한 상태		
테스트 환경	Galaxy s21 5G		
단계	테스트 절차	테스트 입력	예상 출력
1	메인 화면에서 여행리스트 아이콘 선택	아이콘 선택	여행리스트가 있는 화면 표시
2	여행리스트 삭제 버튼 선택	삭제 버튼 선택	여행리스트 삭제
3	여행리스트 중 하나를 선택	여행리스트 선택	별점 받는 팝업 표시
4	별점을 모두 입력	별점 입력	현재 여행으로 바뀌었다는 팝업 표시
사후 조건			
작성자	김현호	작성일	2024.10.30
실행자	김현호	실행일	2024.10.30

Test Case ID	TC-005	Test Case 이름	관광지 선호도
Test Case 개요	관광지 선호도를 재설정한다.		
모듈 이름	환경 설정		
관련 요구사항			
사전 조건	사용자가 관광지 선호도를 완료한 상태		
테스트 환경	Galaxy s21 5G		
단계	테스트 절차	테스트 입력	예상 출력
1	메인 화면에서 톱니바퀴 아이콘 선택	아이콘 선택	환경 설정 페이지 표시
2	관광지 선호도 재설정 블록 선택	블록 선택	관광지 선호도 조사를 다시 하는 페이지로 이동
3	관광지 선호도 조사 시 점수를 입력하지 않고 Next Question 버튼 선택	버튼 선택	Error 질문 버튼을 눌러주세요. 팝업 표시
4	관광지 선호도 조사를 모두 하고 마지막 finish 버튼 선택	버튼 선택	선호도 결과를 보여주는 팝업을 표시
사후 조건	선호도 조사 정보가 DB에 저장되어 있어야 함		
작성자	김현호	작성일	2024.10.30
실행자	김현호	실행일	2024.10.30

Test Case ID	TC-006	Test Case 이름	여행 종료
Test Case 개요	추천받은 여행지의 별점을 입력받는다.		
모듈 이름	환경 설정		
관련 요구사항			
사전 조건	사용자가 여행 시작을 완료한 상태		
테스트 환경	Galaxy s21 5G		
단계	테스트 절차	테스트 입력	예상 출력
1	메인 화면에서 톱니바퀴 아이콘 선택	아이콘 선택	환경 설정 페이지 표시
2	여행 종료 블록 선택	블록 선택	메인 화면으로 이동
사후 조건	현재 여행에 있는 바에 “여행이 없어요” 표시		
작성자	김현호	작성일	2024.10.30
실행자	김현호	실행일	2024.10.30

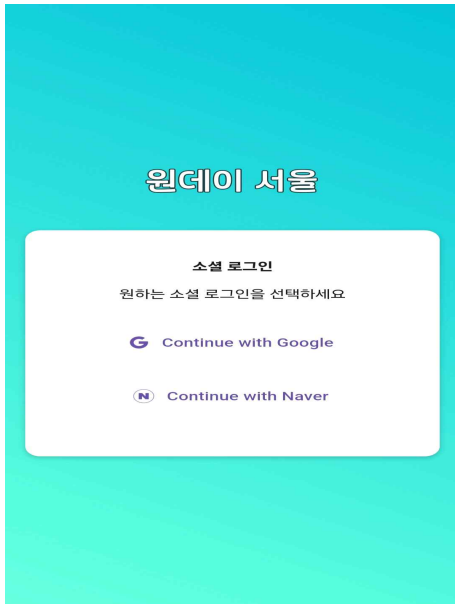
Test Case ID	TC-007	Test Case 이름	로그 아웃
Test Case 개요	사용자의 계정을 로그 아웃 시킨다.		
모듈 이름	환경 설정		
관련 요구사항			
사전 조건	사용자가 로그인한 상태		
테스트 환경	Galaxy s21 5G		
단계	테스트 절차	테스트 입력	예상 출력
1	메인 화면에서 톱니바퀴 아이콘 선택	아이콘 선택	환경 설정 페이지 표시
2	로그 아웃 블록 선택	블록 선택	로그인 화면으로 이동
사후 조건			
작성자	김현호	작성일	2024.10.30
실행자	김현호	실행일	2024.10.30

Test Case ID	TC-008	Test Case 이름	탈퇴하기
Test Case 개요	사용자의 계정을 삭제한다.		
모듈 이름	환경 설정		
관련 요구사항			
사전 조건	사용자가 로그인한 상태		
테스트 환경	Galaxy s21 5G		
단계	테스트 절차	테스트 입력	예상 출력
1	메인 화면에서 톱니바퀴 아이콘 선택	아이콘 선택	환경 설정 페이지 표시
2	탈퇴하기 블록 선택	블록 선택	정말로 탈퇴하시겠습니까? 팝업 표시
3	팝업에서 네 선택	네 선택	로그인 화면 표시
사후 조건			
다시 로그인하면 선호도 조사를 다시 받아야 한다.			
작성자	김현호	작성일	2024.10.30
실행자	김현호	실행일	2024.10.30

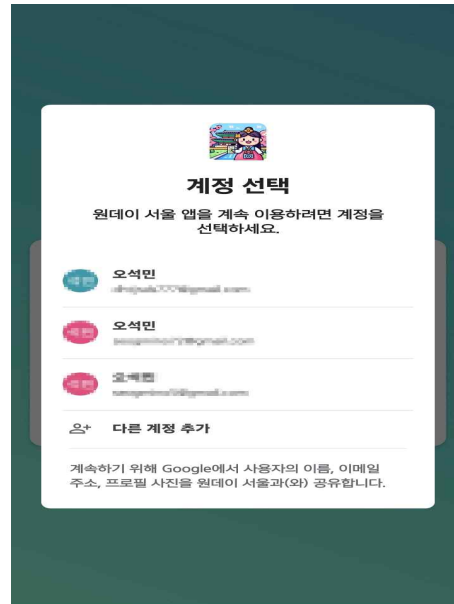
## VII. 시스템 결과서

## I. 소프트웨어 결과 화면 (SW Result)

### 1.1 앱 로그인 화면



### 1.2 계정 선택 화면



### 1.3 설문 화면

#### travel test

Question 1/45

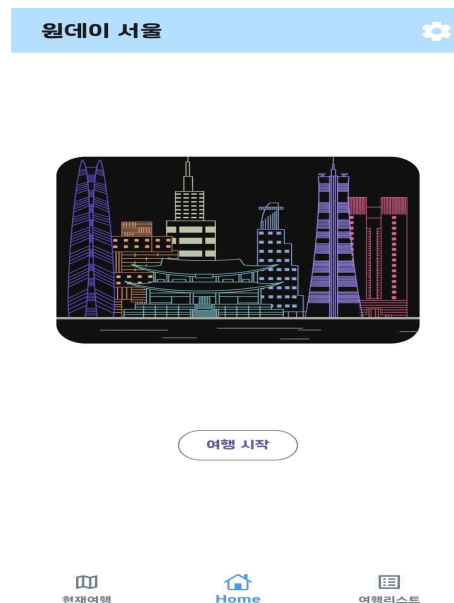
관광지 조사..

1(매우 아니다), 2(아니다), 3(보통), 4(그렇다), 5(매우 그렇다)  
여행 중 현지 공연(연극, 콘서트 등)을 즐기고 싶습니까?

1 2 3 4 5

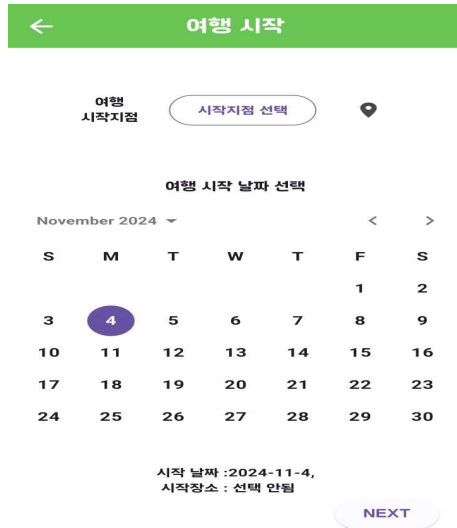
Next Question

### 1.4 앱 회원가입

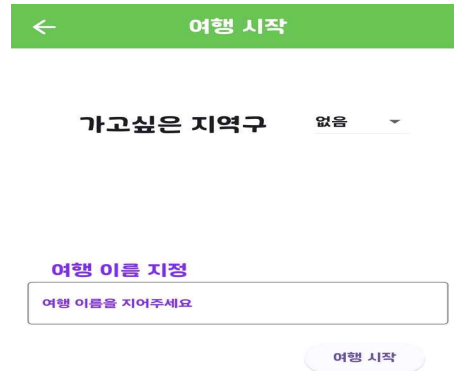


## I. 소프트웨어 결과 화면 (SW Result)

1.5 여행 시작 화면(1)



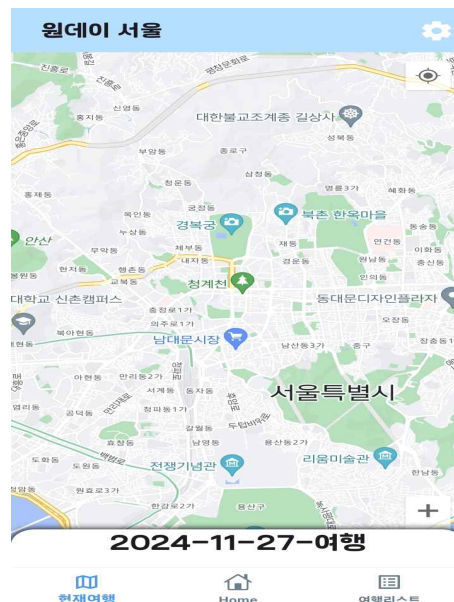
1.6 여행 시작 화면(2)



1.7 여행 리스트 화면



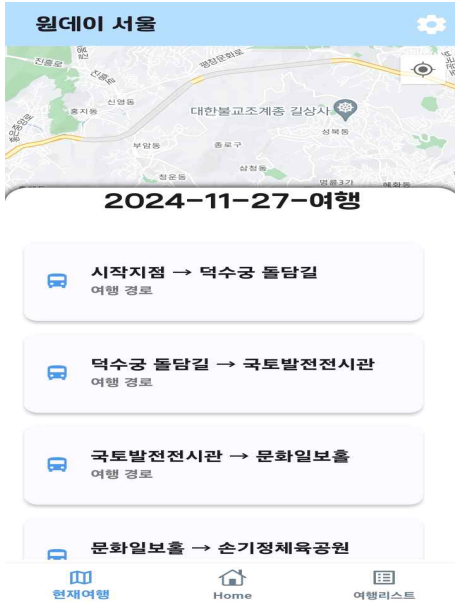
1.8 현재 여행 화면





## I. 소프트웨어 결과 화면 (SW Result)

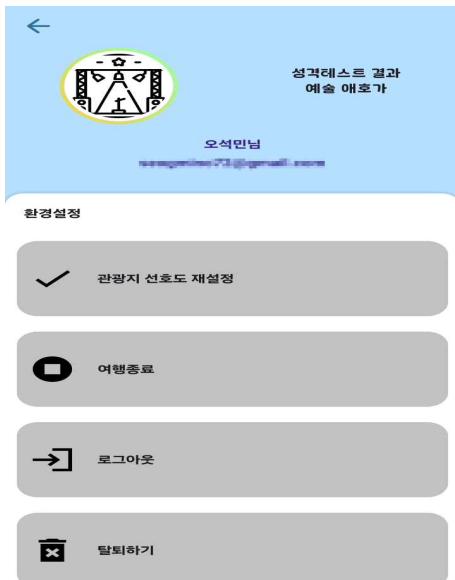
### 1.9 현재 여행지 목록



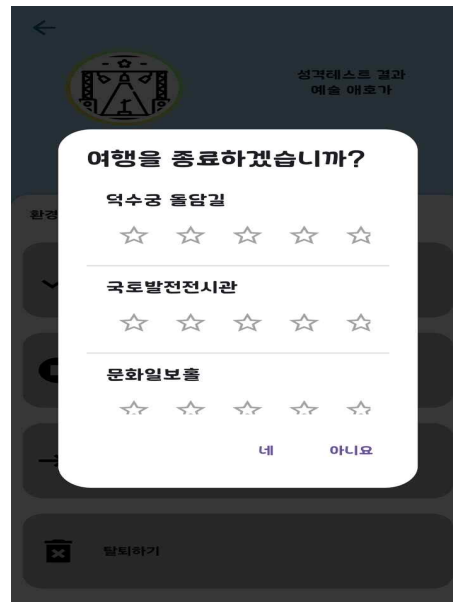
### 1.10 여행 경로



### 1.11 설정 화면

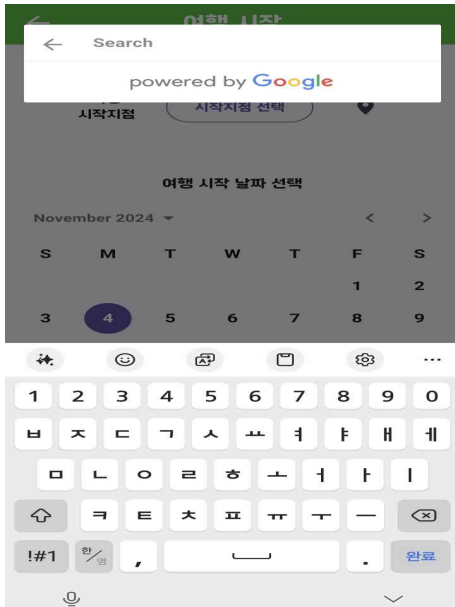


### 1.12 여행 종료 화면

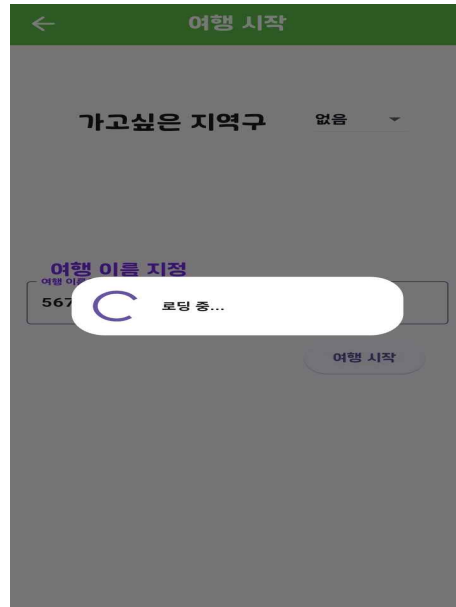


## I. 소프트웨어 결과 화면 (SW Result)

1.13 시작 지점 선택



1.14 로딩 중 화면





## VIII. 개 발 후 기

## 김현호

캡스톤 프로젝트를 1년 동안 진행하면서 처음에 아이디어를 생각했을 때는 쉬워 보였고 금방 완성되리라 생각했습니다. 하지만 보고서를 작성하면서 아이디어의 많은 문제점이 보여 점점 내용을 수정해 가며 개발을 시작했습니다. 이런 프로젝트를 하는 것이 처음이다 보니 보고서 대로 개발하는 과정에서 문제가 많이 발생해 보고서 수정과 개발을 동시에 진행하여 힘이 들었지만, 팀원이 모두 열심히 노력을 해주었기 때문에 함께 끝까지 완성 시킬 수 있었다고 생각합니다.

## 신지섭

이번 프로젝트 시작할 때 처음 원하던 게임이 아니라 여행 응용소프트웨어를 만들게 되어 초반에는 크게 의욕이 나지 않았습니다. 하지만 이참에, 평소에 관심이 있던 플러터를 해보자 생각을 하여 어플을 만들게 되었고 제 욕심으로 플러터를 선택하게 되어 어플은 대부분 제가 전담하게 되었습니다. 처음에는 UI를 어떻게 꾸밀지 기능은 무엇무엇이 필요한지 매우 막막했지만, 천천히 하나하나 구현해 나가면서 어플 개발에 대해서 많이 알게 되기도 하였고 많이 실패도 해봤으며 점점 어플 제작에 속도가 붙기 시작했습니다. 특히 이번 프로젝트에 대중교통 경로 안내 부분을 구현하면서 많은 어려움이 있었고 그것을 해결해 나가는 과정에서 어플 제작 실력이 많이 는 거 같습니다. 그렇게 점점 실력이 늘어나게 되니 나중에 가서는 제 코드에 아쉬운 부분이 많이 보이기 시작했습니다. 초반에는 별 생각없이 개발을 하다보니 코드의 객체화는 생각하지 않고 한곳에 다 때려 박는 식에 코드가 만들어진 것입니다. 그래도 후반으로 갈수록 최대한 각 기능에 대해 클래스나 함수로 모듈을 분리해 작동하도록 코드를 짜고 그것들을 연결 지으면서 재미도 있고 공부도 많이 된 것 같습니다. 마무리 단계에 접어들면서 최대한 리팩토링을 하며 객체화를 못 한 부분을 수정해 보려 했지만 그렇기엔 너무 스파게티 코드가 되어서 시간 부족으로 마무리 짓지는 못했으나 이번 프로젝트를 계기로 다음부터는 개발 시작 전 먼저 기능에 대한 정리를 시작하고 코드를 짤 때 최대한 객체화시켜 코드의 재사용성과 직관성을 좋게 만들어야겠다고 다짐하는 프로젝트였던 거 같습니다.

## 오석민

캡스톤 강의로 처음 팀 프로젝트를 진행하면서 잘할 수 있을까? 라는 의문이 먼저 들었지만, 팀 프로젝트를 진행하면서 단계별로 완성되면서 의구심이 점차 줄어들었습니다. 팀 프로젝트를 하면서 개발 과정에 관해 공부가 많이 되었습니다. 그래도 공부를 더 많이 해야겠다고 생각하게 되었습니다. 또한 저의 부족한 부분을 팀원들이 많이 도와주어서 개발에 힘든 점을 줄여줬습니다.

## 장진원

이번 캡스톤 프로젝트를 진행하면서 지난 4년간 배웠던 것들을 돌아 볼 수 있었고 실제로 작업을 해보니 계획과는 많이 틀어지게 되는 경험을 하며 보고서 수정과 개발을 동시에 하면서 최초 작업의 중요성을 알게 되었습니다. 또한 혼자하는 작업이 아니라 서로간의 소통의 중요성을 깨닫게 되었고 그와 동시에 제가 부족한 점을 팀원들이 채워주며 배우는점도 많았고 보다 공부를 열심히 해야겠다는 생각이 들었습니다.

