



소프트웨어공학 프로젝트

프로젝트명: 여행어때!

(멤버)

김승열 - 2015118566

서영빈 - 2016110123

고상민 - 2014118003

김현규 - 2018117524

이진 - 2018114468

이석희 - 2018117750

A person is sitting on a red wooden bench with white concrete legs, facing away from the camera and looking out at the ocean. The sky is blue with large white clouds. The person is wearing a dark t-shirt and shorts, and their arms are raised slightly.

Contents

01.

기획 배경

02.

개요

03.

Sequence Diagram &
Software Architecture

04.

계획대비 변경 사항

05.

Burn Down Chart

06.

Work Plan

07.

느낀 점,점,점



여행 언제!

사람들은 여행을 하기에 앞서 많은 고민을 하게 된다. 행선지를 결정, 여행지 정보, 여행지의 날씨, 동행자 등 고려하고 선택해야 할 것들이 많다.

이를 도와주고자 기획한 것이 '여행 언제'이다.

여행지 추천 및 이용자들간의 정보교류 그리고 함께할 동행자를 결정해서 즐거운 여행에 공헌 할 것이다.



1

카카오톡과 연동, 사용자 계정 관리

2

사용자가 선호하는 여행지를 선택지를 통해 분석한다.

2

여행지의 날씨를 정보를 제공

4

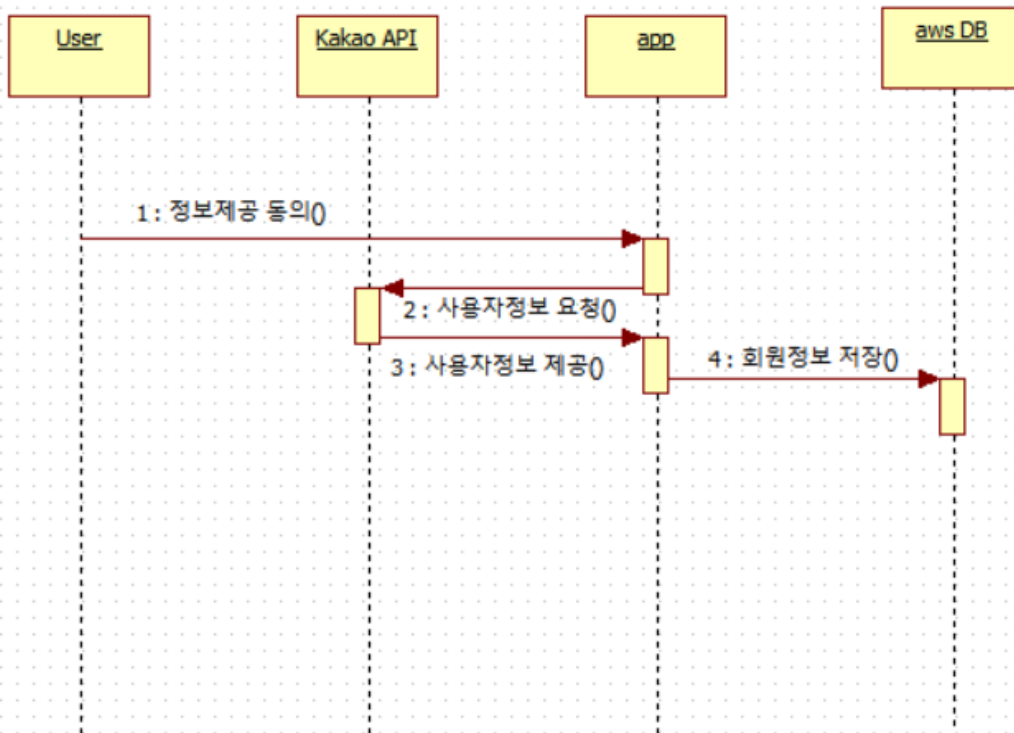
추천 여행지 커뮤니티 형성, 게시물 작성 + 수정 삭제

5

함께 여행할 동행자를 탐색

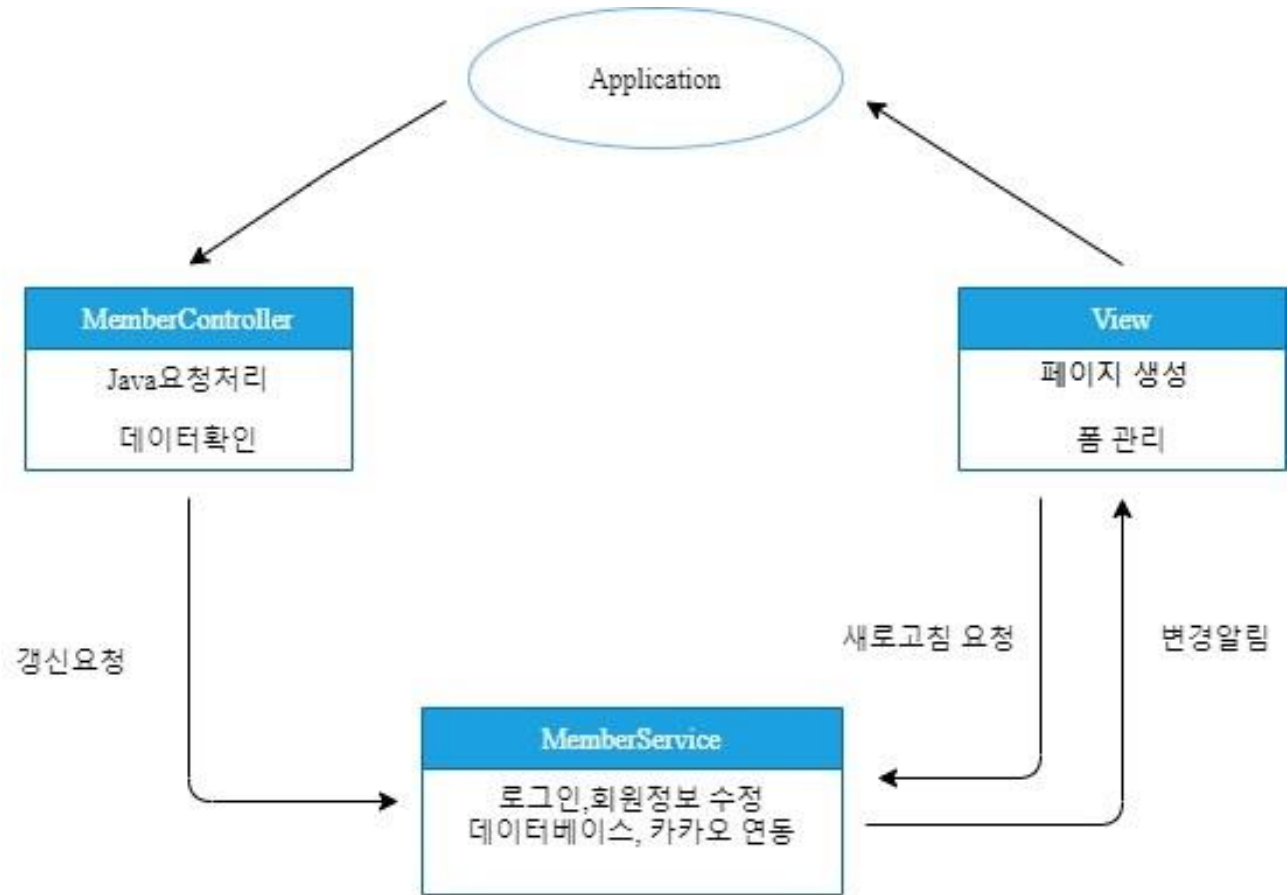


카카오톡 연동 및 로그인



(1)사용자가 app으로 정보제공 동의 → (2)app이 kakao api로 사용자 정보 요청
→ (3)kakao api 는 app으로 사용자 정보 제공 → (4)서버로 회원 정보 Data Base화

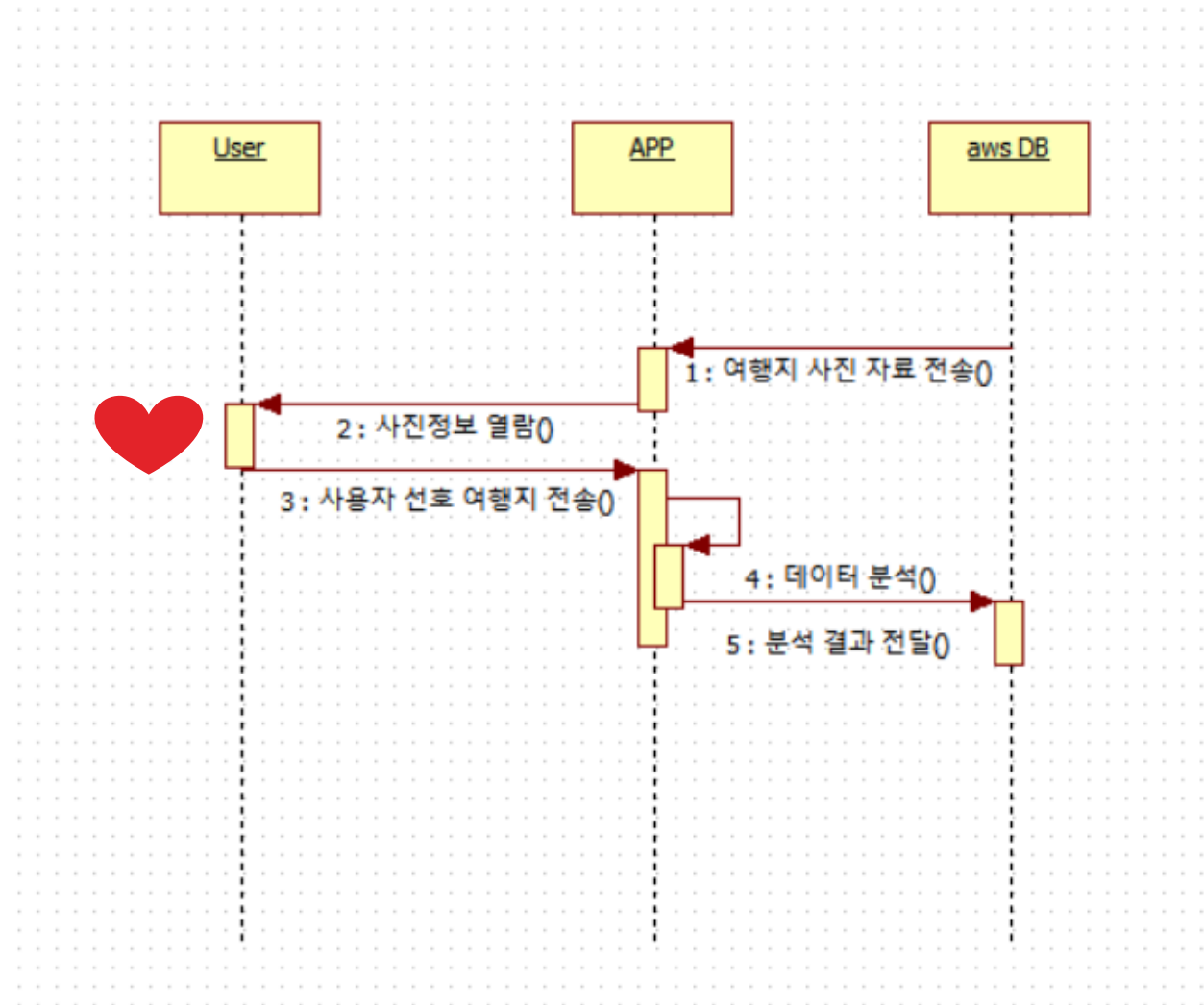
회원정보 SW 아키텍처



MemberController에서 데이터 확인 → MemberService에서 사용자정보 수정 및 DB, 카카오API 연동 갱신
↔ 완료된 페이지 생성, MemberService와 수정 및 확인 상호작용



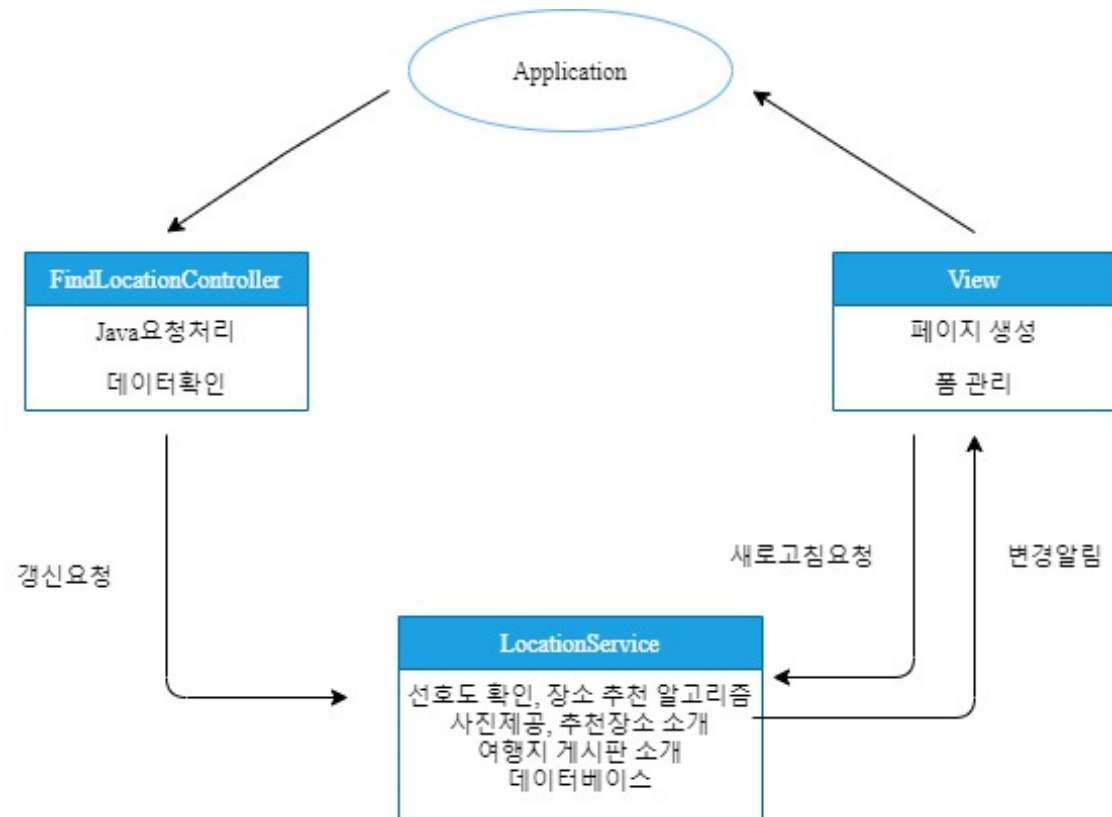
사용자 선호도 조사



(1)서버DB에서 App으로 여행지 사진자료 전송 → (2)App은 사용자에게 사진 열람권한 제공 → (3)사용자는 하트로 App에게 선호 정보 전송 → (4)데이터 분석 → (5)App은 분석결과를 서버로 전달



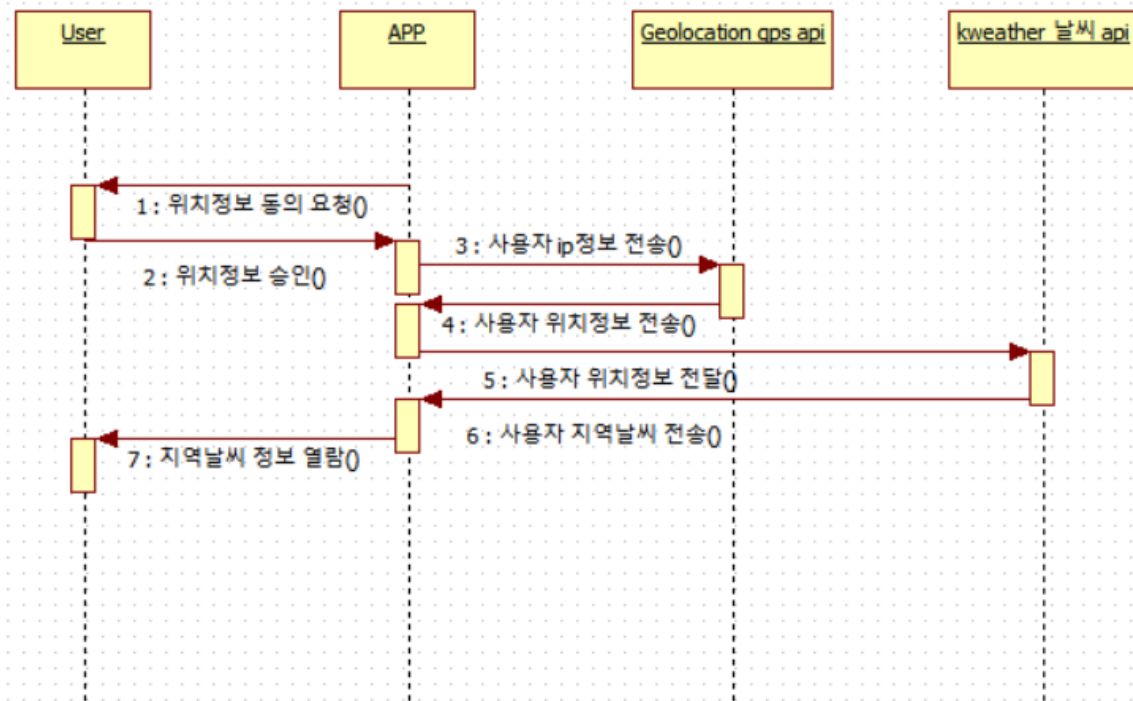
여행지 추천 SW 아키텍처



FindLocationController에서 데이터 확인 → LocationService에서 선호도 확인, 장소추천, 사진제공, 여행지 게시판 소개 DB 처리 ↔
완료된 페이지 생성, LocationService 와 수정 및 확인 상호작용

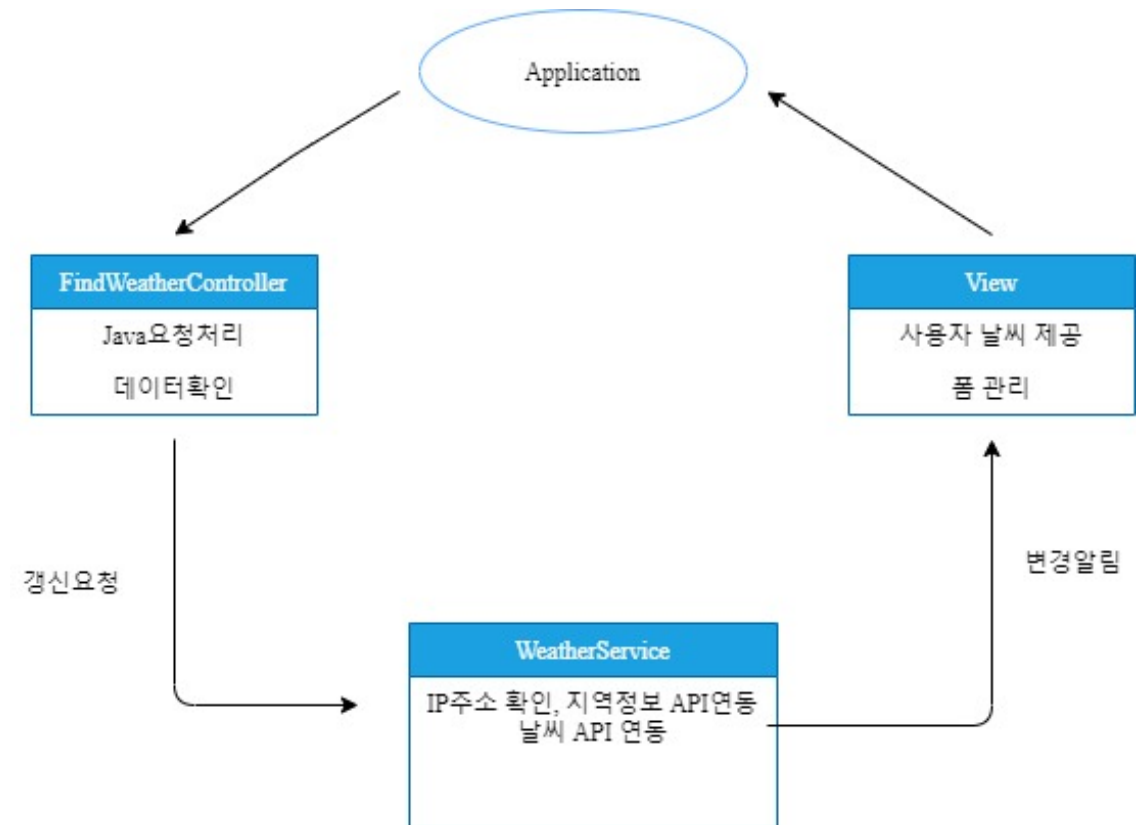


사용자 지역 날씨 정보제공



(1)App에서 사용자에게 위정정보 동의 요청 → (2)사용자는 App에게 승인 여부 전달 → (3)App은 위치정보 Api로 사용자 ip 전달 → (4)확인된 위치정보를 다시 App으로 전송 → (5)App은 케이웨더 날씨 api로 사용자 위치 정보 전달 → (6)케이웨더 날씨 api에서 App으로 날씨 정보를 받아옴 → (7)사용자는 App으로 부터 날씨 정보 열람

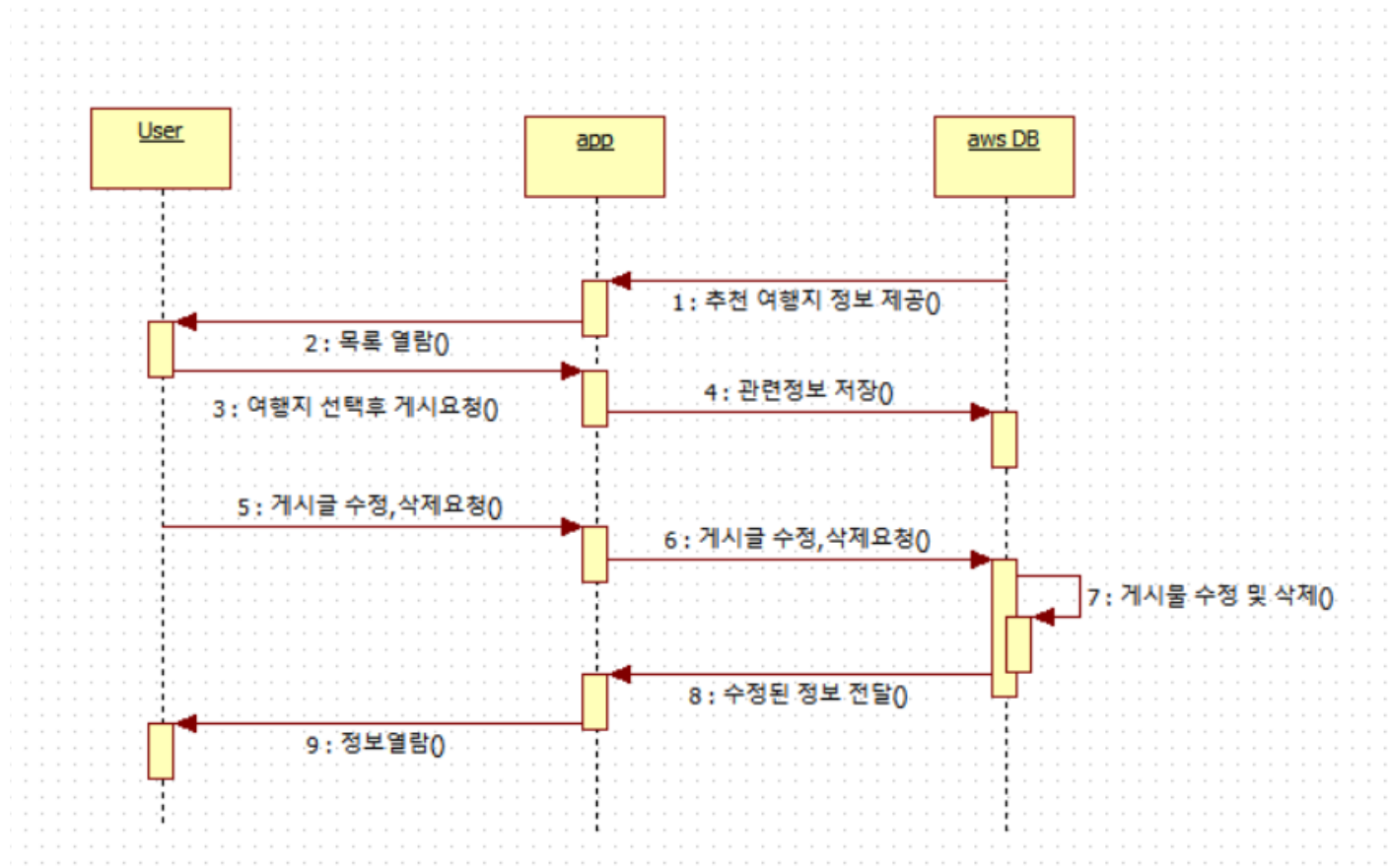
사용자 날씨 제공 SW아키텍처



FindWeatherController에서 데이터 확인 → WeatherService 에서 Ip 정보 확인 및 위치정보 Api와 케이웨더 날씨 Api 연동 ↔ 결과를 view로 전달



추천 여행지 제공 및 사용자 게시물 작성 + 수정 및 삭제

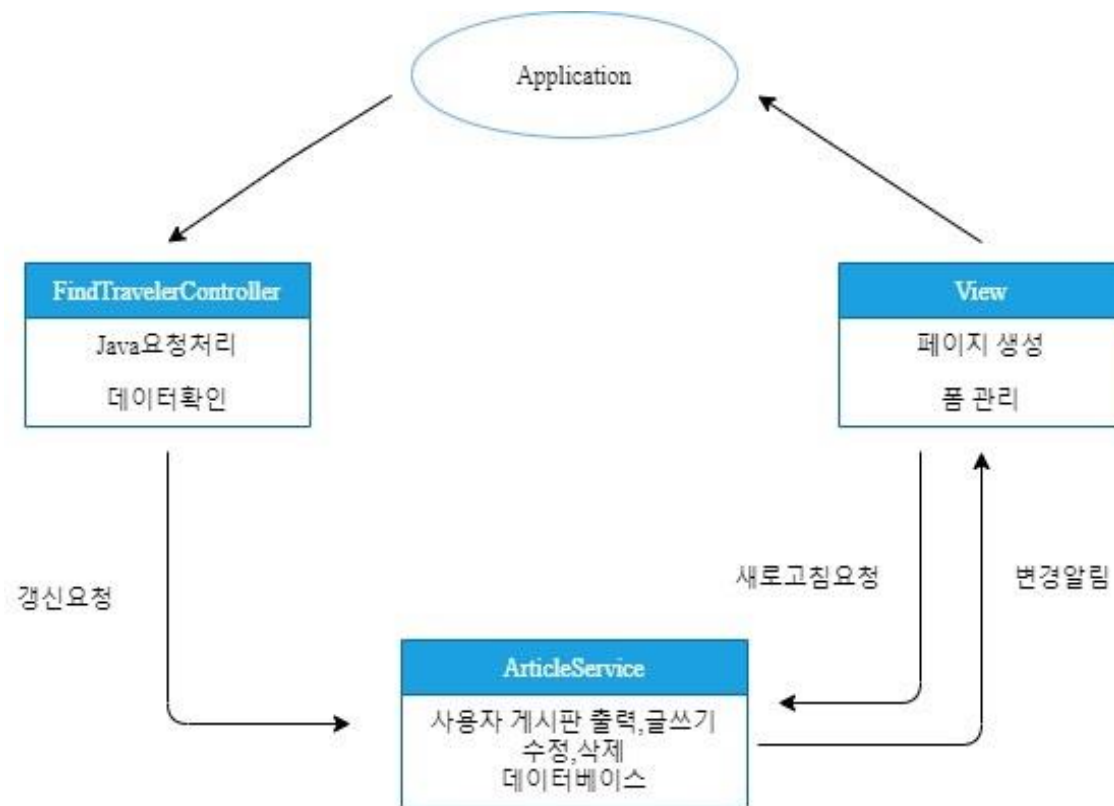


게시글 작성: (1) 서버는 APP으로 추천 여행지 정보 전달 → (2) App은 사용자에게 목록 제공 → (3) 사용자는 여행지 선택 후 App에게 게시요청 → (4) App은 데이터를 서버에 기록

수정/삭제: (5) 사용자가 App으로 수정 삭제 요청 전달 → (6) App은 사용자의 요청을 서버로 전달 → (7) 서버는 삭제/수정을 수행 → (8) 서버는 App으로 수정된 정보 전달 → (9) 사용자가 이를 열람



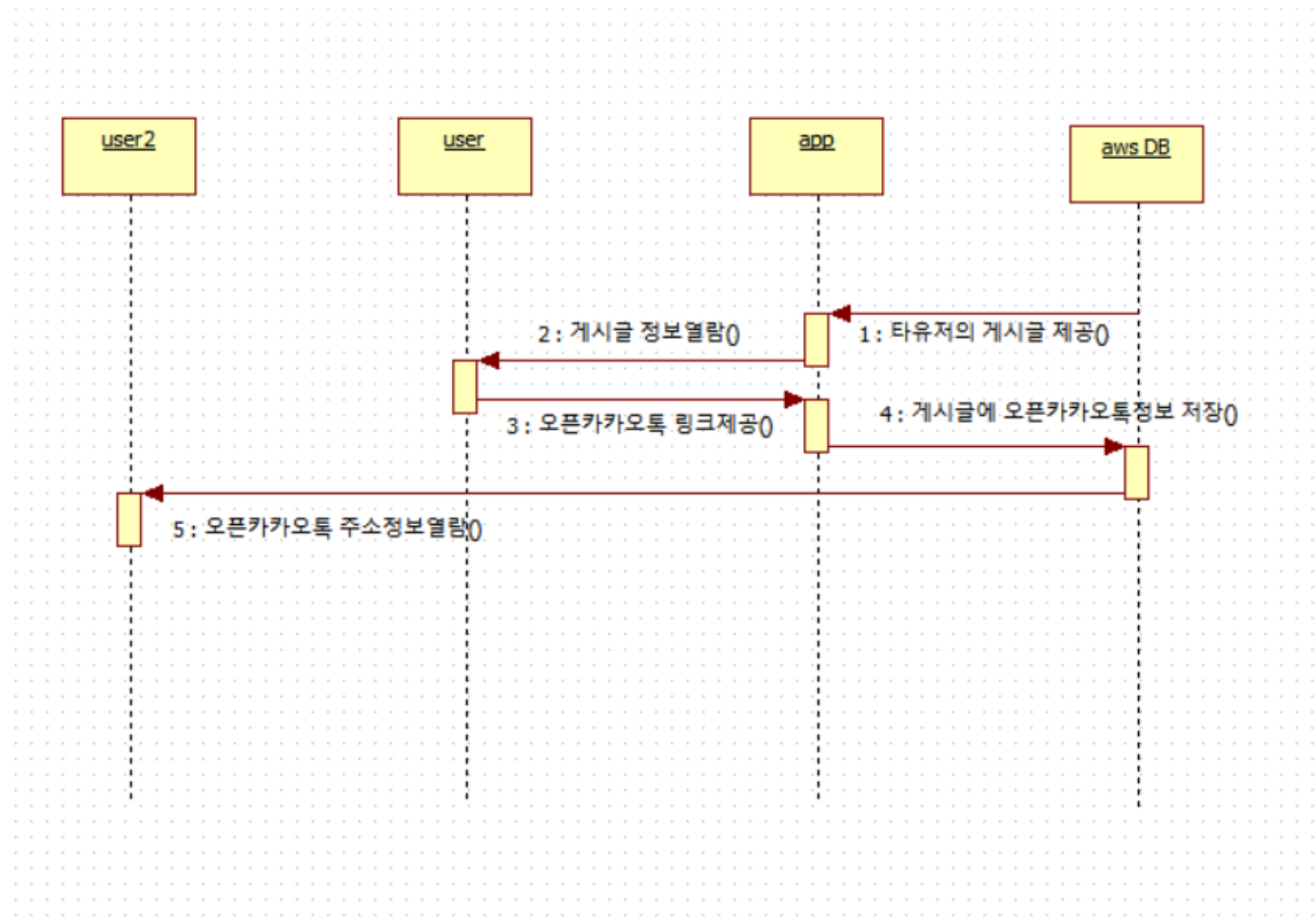
사용자 게시판 아키텍처



FindTravelerController에서 데이터 확인 → ArticleService에서 사용자 게시판 출력, 글 수정/삭제 DB화 ↔ 완료된 페이지 생성, ArticleService와 수정 및 확인 상호작용



여행 동행자 찾기



(1)서버는 다른 사용자의 게시글을 App에 제공한다. → (2)App은 사용자에게 다른 사용자의 게시글을 제공한다. → (3)사용자는 오픈 카카오톡 링크를 App으로 전달 → (4)서버는 App으로 부터 받은 오픈 카카오톡정보를 게시글에 저장 → (5)서버는 다른 유저에게 오픈 카카오톡 주소 제공



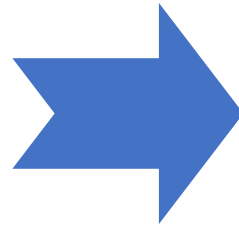
변경 전

공기 언제 ?

(환경 : 아두이노, 앱, 서버)

기존 제품들과의 차별성 부재,
실용성에 대한 의문

고정적인 역할분담



변경 후

여행 언제 !

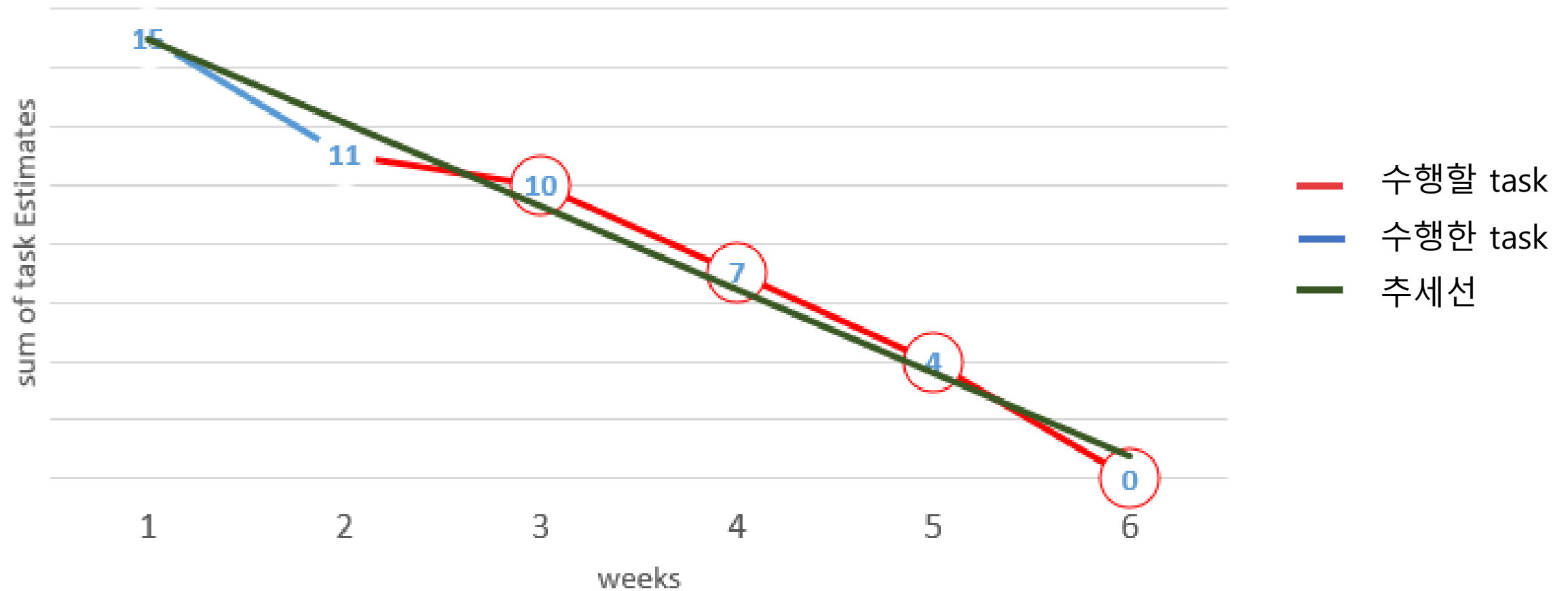
(앱, 서버)

실용성 개선, 수요 충족
















탄력적인 역할분담



번다운차트





TASK	8주차	9주차	10주차	11주차	12주차	13주차
[계획 수립]						
아이디어 구체화						
[웹페이지 생성]						
UX 디자인						
UI 디자인						
[구현기술 설계]						
기술설계						
개발 언어 선택						
DB생성						
[개발단계]						
카카오톡 연동 및 로그인기능						
사용자 선호도 조사 기능						
사용자 지역 날씨 정보 제공						
추천 여행지 제공						
사용자 게시글 작성						
게시글 수정 및 삭제						
여행 동행자 찾기 기능						
[오류수정]						
[완성도 향상]						



	개발 내적인 측면	개발 외적인 측면
어려웠던 점	프로젝트 구성 인원들의 안드로이드 스튜디오, 서버, db 등 경험 부재로 인해 초반에 개발 환경에 적응하는데 어려움을 느낌.	많지 않은 프로젝트 경험으로, 프로젝트 전체를 어떤 식으로 분배 할지에 대한 판단 이 어려웠음
	다른 IDE들은 코드에서 오류가 있는 부분들을 피드백을 해주기에 디버깅에 수월, 그러나 안드로이드 스튜디오는 컴파일에 에러가 없어도 어플 자체에서 오류가 나는 경우가 많았고 디버깅을 하는데에 어려움이 있었음	사용자의 니즈를 고려하면서 기능을 만들다 보니, 그에 대한 생각을 구체화하기가 힘들었음
아쉬웠던 점	개발 초기 UI 디자인에 많은 시간을 할애, 전체적인 기능 구현이 계획된 기간 내에 완료 되지 못함	초기 '공기어때' 기획 당시, 시장 경쟁력이 없다는 피드백 이 후 프로젝트 변경으로 계획의 차질이 있었음 -> 초반에 사용자들의 니즈 파악에 대한 이해가 부족했던 점이 아쉬웠음
		프로젝트에 관한 미팅이 비대면으로 진행되다 보니 프로젝트에 관한 계획을 구체화 하는데 어려움을 느낌
바라는 점	프로젝트의 순탄한 진행과 성공적인 작품 구현이 이루어지기를 희망함	



Thank you