

Mobile Programming Final Presentation

- 대중교통



5 팀

18101966 김동주

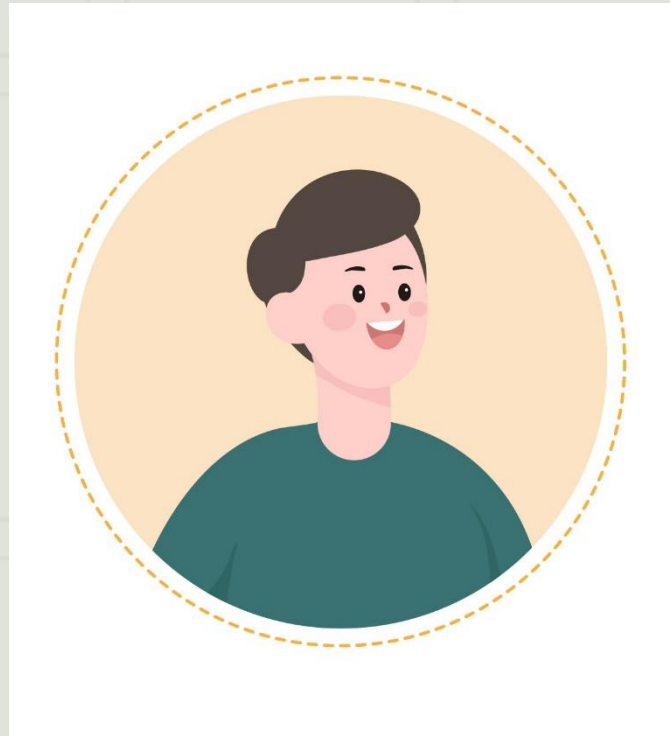
18101998 이승준

1) Application name : 대중교통



지도 어플을 이용한 지하철 커뮤니티 어플
학교 주변에 있는 지하철 역 대한 도착 정보를 불러오고 오픈채팅방을 열어 익명의 사람들과
지하철 출발 및 도착에 대한 정보를 공유하는 어플

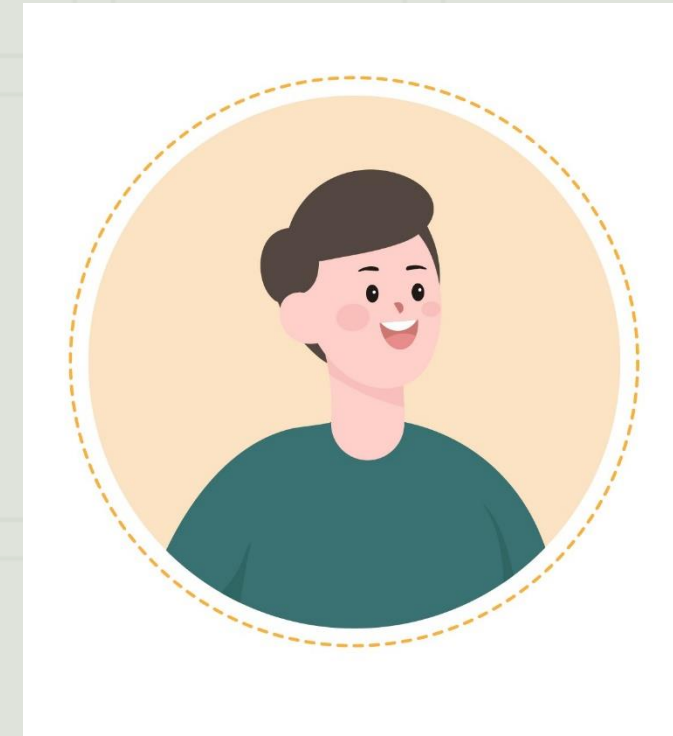
2) Role & Responsibilities



김동주

지도

지도 서비스
장소 검색 기능
지하철역 도착 정보
알람 기능



이승준

채팅

사용자 익명 처리
채팅 기능
사진 촬영 및 이미지 전송기능

3) Features of App

001

지도 서비스

002

장소 검색 기능

003

실시간 지하철 역 도착 정보

004

지하철 도착 알람

005

실시간 채팅

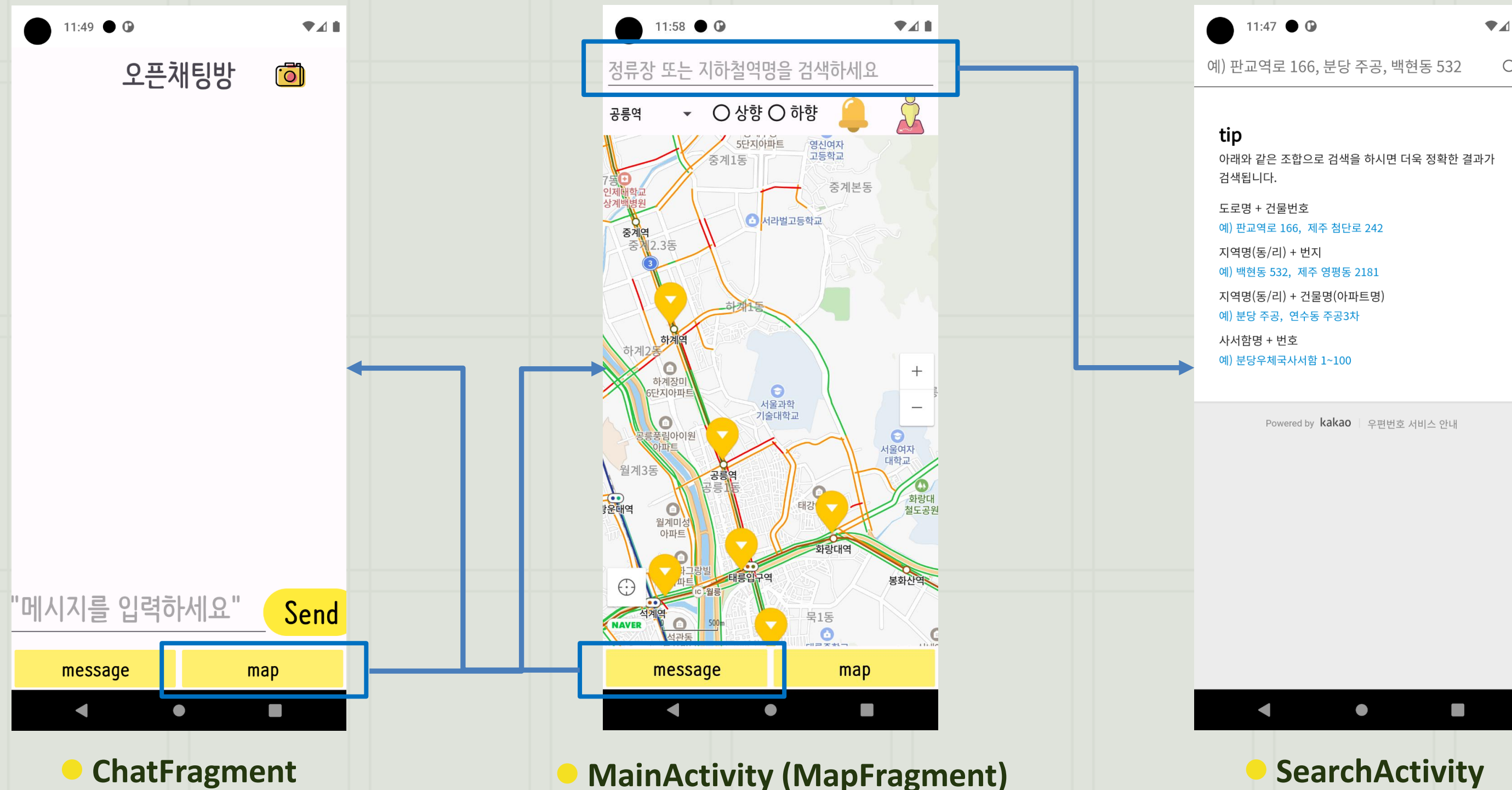
006

사용자 익명 인증

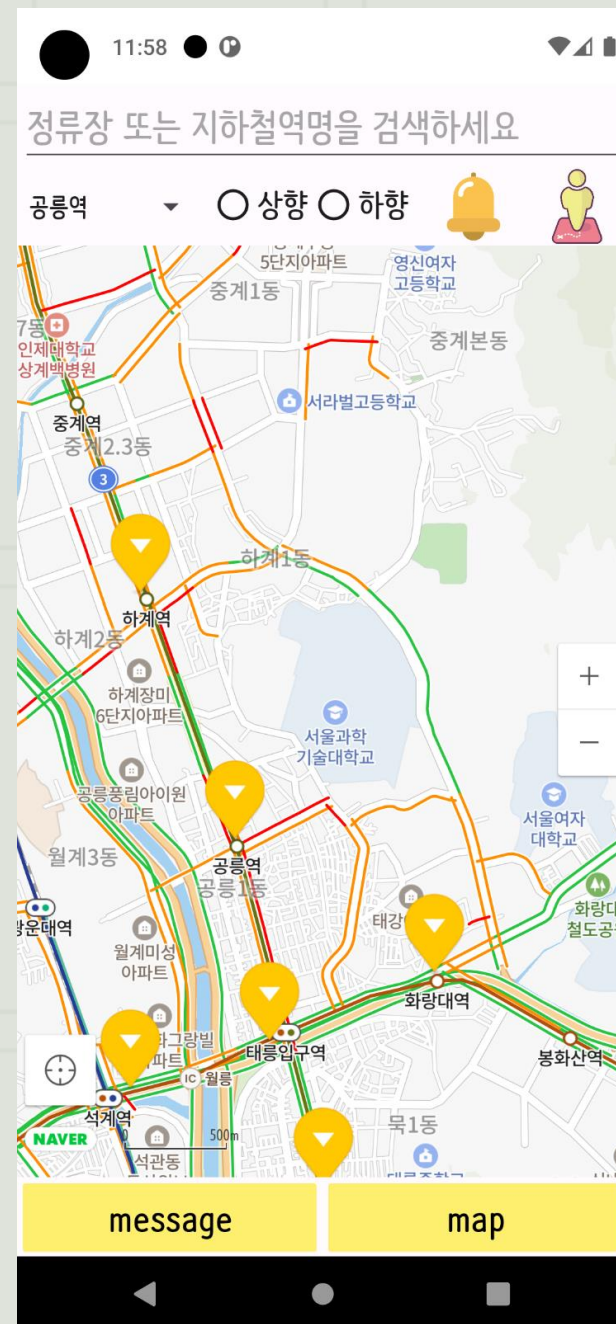
007

카메라 촬영 및 사진 공유

4) Functions & Technical Methodology of App

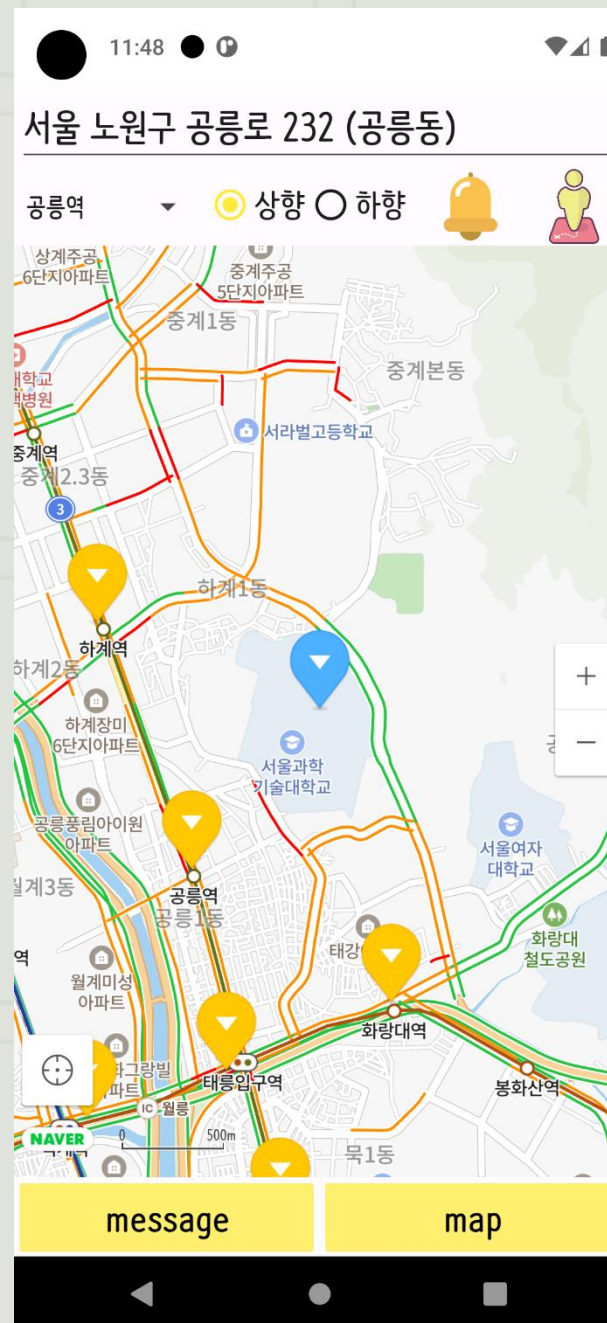


4) Functions & Technical Methodology of App - 지도 서비스



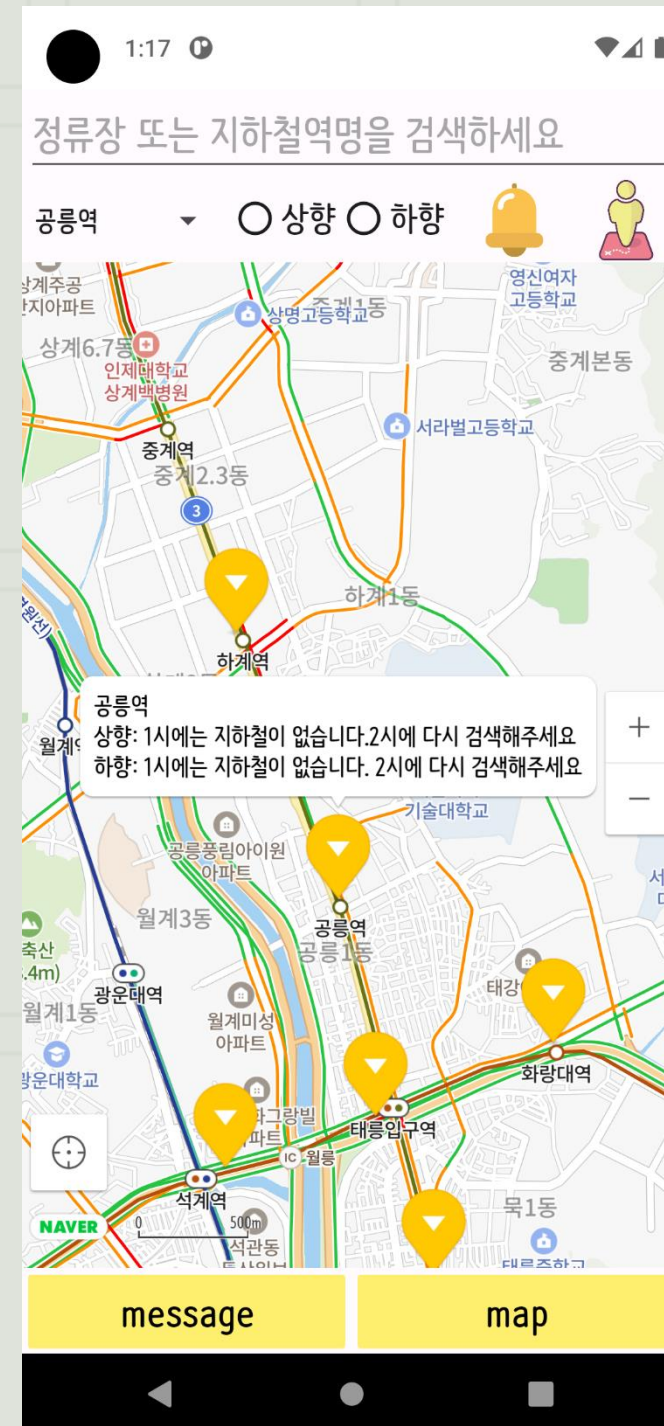
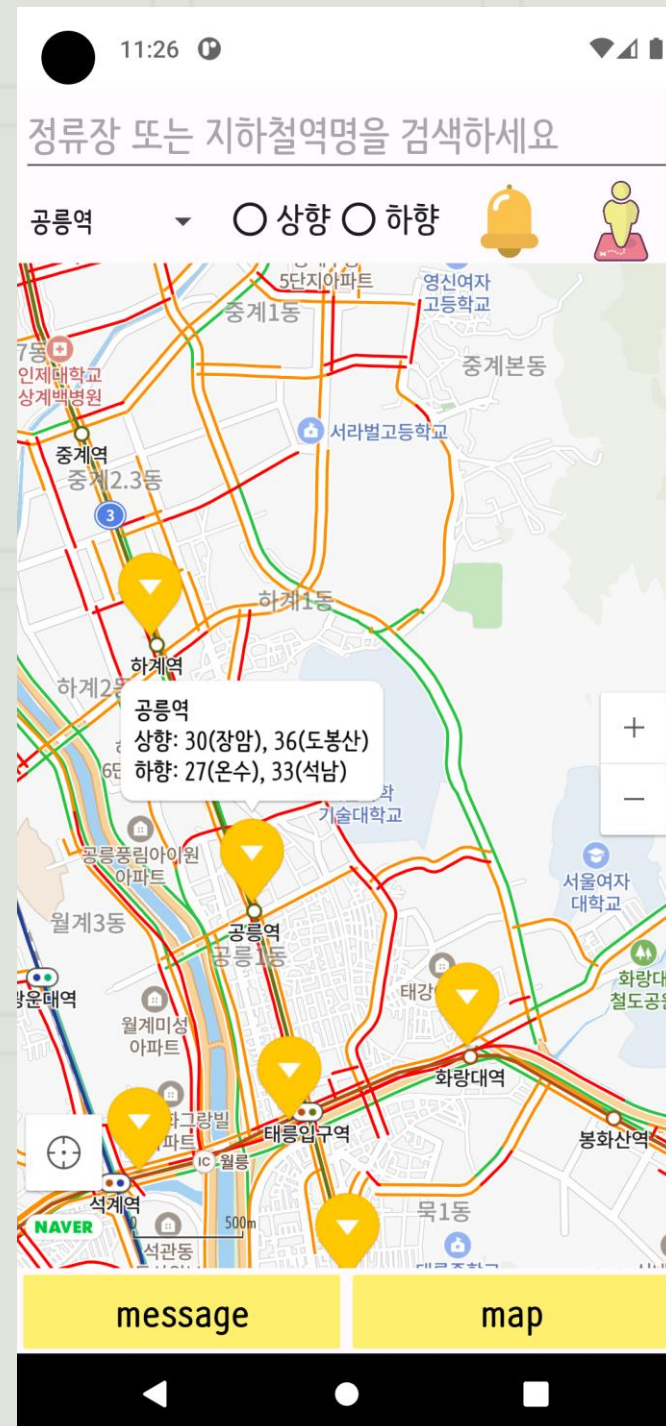
- 위젯/컴포넌트/레이아웃
 - 1) mapView : naverMap api를 활용
 - 2) 노란색 마커 : 지하철 역
 - 3) Edit_search_text : searchActivity로 이동
- 서드파티 라이브러리
 - naverMap API

4) Functions & Technical Methodology of App - 장소 검색 기능



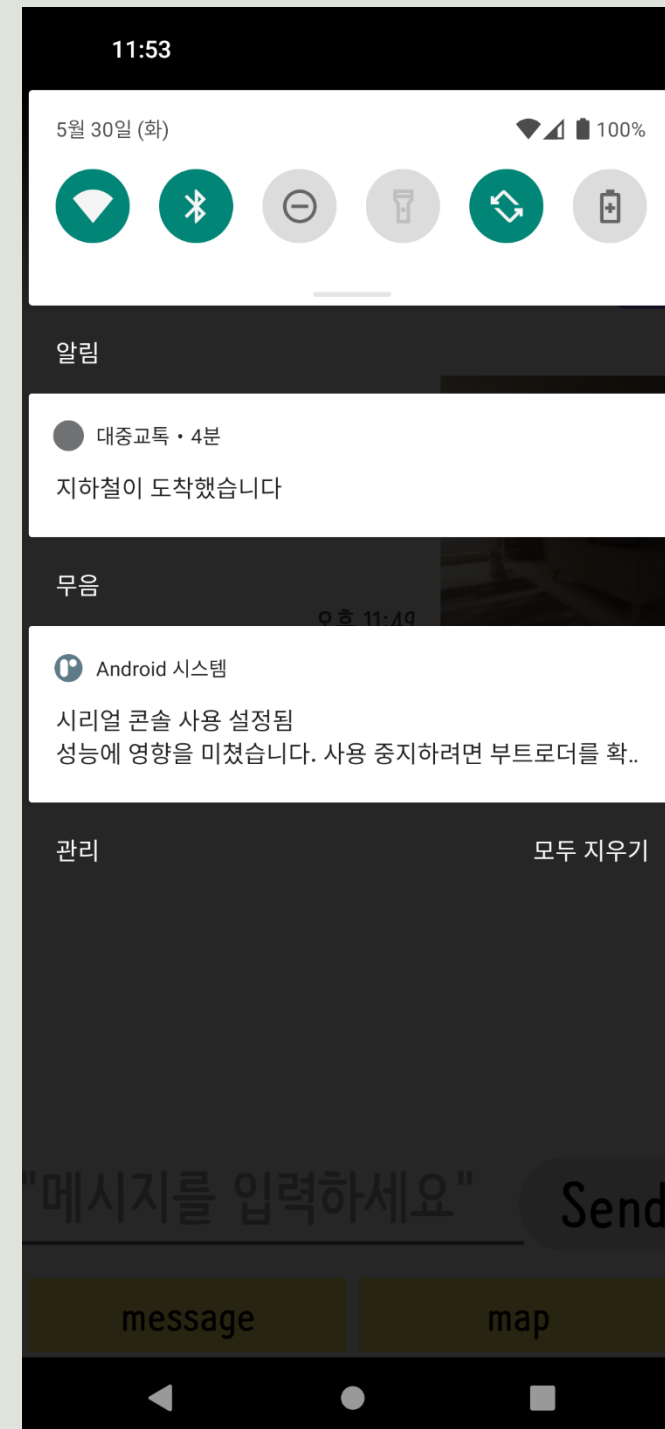
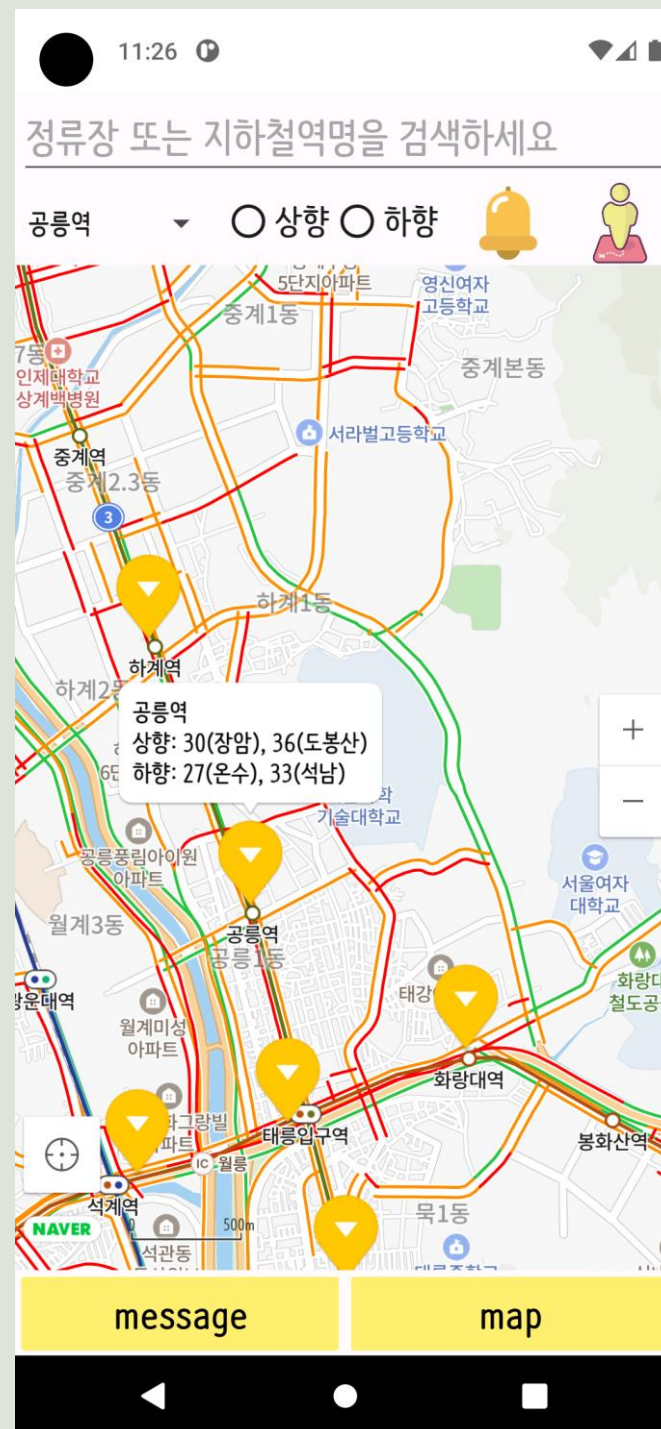
- 위젯/컴포넌트/레이아웃
 - 1) WebView : Firebase에 Daum 도로명 주소 API를 적용시킨 html을 hosting
 - 2) Edit_search_text : registerActivityResult를 이용해 검색한 주소 받아오기
 - 3) 사람 이미지 버튼 : 입력된 주소를 지도에 매핑
 - 4) 파란색 마커 : cameraposition을 가운데로 이동하여 입력된 주소 위치 표시
- 장소 검색 기능
 - 1) Geocoder : 가져온 주소에 해당하는 위도, 경도로 변경
- 서드파티 라이브러리
 - Daum 도로명 주소 API
 - Firebase hosting

4) Functions & Technical Methodology of App - 실시간 지하철 역 도착 정보



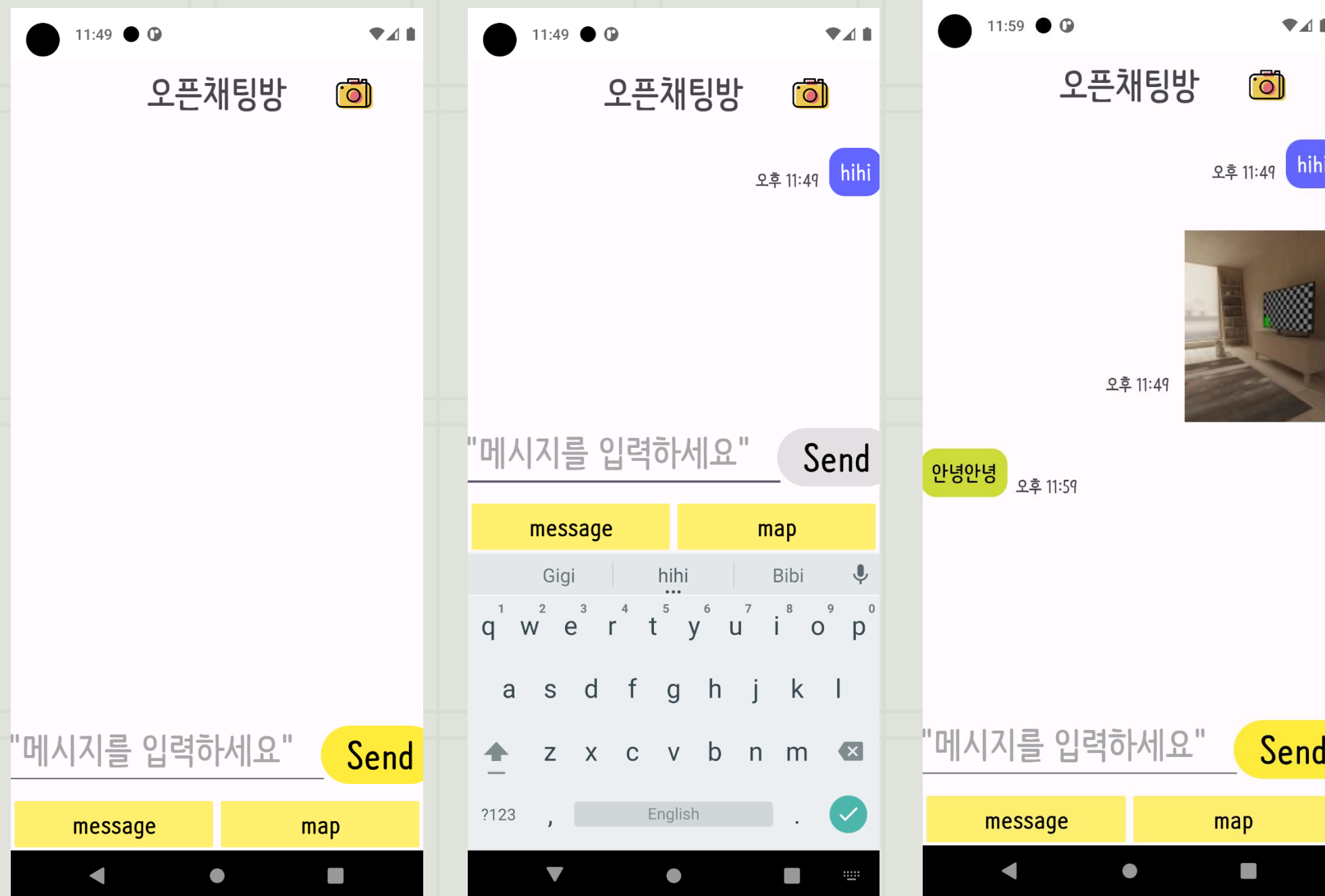
- 위젯/컴포넌트/레이아웃
 - 1) 마커 : 지하철 역 마커, 터치하면 infoWindow가 나타남
 - 2) infoWindow : Odsay API를 활용해 현재 시간 기준 가장 빨리 도착하는 지하철 시간표 2개 표시, 터치 시 chatFragment로 이동
- 서드파티 라이브러리
 - Odsay API
 - 1) json 데이터를 받아와 retrofitConnection 이용

4) Functions & Technical Methodology of App – 지하철 도착 알람



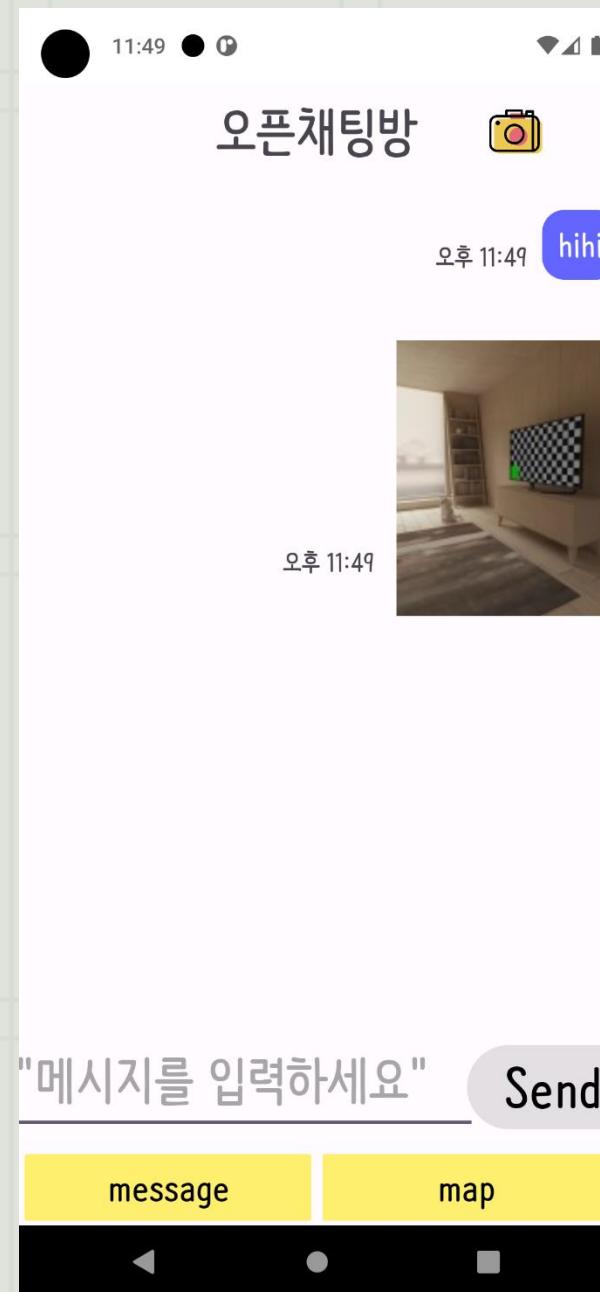
- 위젯/컴포넌트/레이아웃
 - 1) Spinner : 지하철 역 리스트를 만들어 ArrayAdapter로 연결
 - 2) RadioGroup : 지하철 방향 (상향 & 하향)
 - 3) 알람 이미지 버튼 : Spinner와 RadioGroup버튼에 해당하는 내용을 갖고 실시간 지하철 시간표를 가져와 현재 시간 기준 가장 최근 지하철 도착시간에 알람 설정
 - 4) Toast : 알람 생성 내용
- 알람 기능
 - 1) AlarmReceiver : BroadcastReceiver 클래스 정의
 - 2) BroadcastReceiver : 알람을 수신, 알림을 생성하여 표시
 - 3) setAlarm 함수 : 원하는 시간을 AlarmReceiver에 전달
 - 4) AlarmManager : 알람이 울리면 AlarmReceiver가 알람을 받고 알림을 보냄
- 서드파티 라이브러리
 - NotificationCompat
 - 1) NotificationManager : 객체로 알림 관리
 - 2) NotificationCompat.Builder : 알림의 속성 설정
 - 3) NotificationManager.notify() : 설정된 알림을 표시

4) Functions & Technical Methodology of App – 실시간 채팅, 사용자 익명 인증



- 위젯/컴포넌트/레이아웃
 - 1) 메시지 입력 : 사용자가 텍스트를 입력
 - 2) Send 버튼 : 입력한 메시지를 전송, 입력 창 공백이면 비활성화
 - 3) textView : 입력한 메시지와 시간대 표시
 - 4) RecyclerView : MessageAdapter의 getItemViewType에 따른 ViewHolder를 설정하여 메시지 목록으로 나타냄
 - 5) 바인딩 클래스 : 레이아웃 내의 뷰 요소와 데이터를 바인딩
- 메시지 기능
 - 1) Firebase 참조 경로에 입력한 메시지 push
 - 2) Firebase 참조 경로에 event가 발생했을 때 변경된 데이터를 ChatData 객체로 가져와 MessageAdapter를 통해 메시지 표시
 - 3) ChatData 객체 send_uid로 나와 상대방을 구분
 - 4) 일반 메시지 : ChatData 객체 msg에 내용이 있고 imageUrl 이 null
- 서드파티 라이브러리
 - Firebase
 - 1) Firebase Authentication : signAnonymously()를 활용하여 사용자를 익명으로 관리, User 객체 uid에 저장
 - 2) Firebase Realtime Database : 채팅 메시지를 실시간으로 저장하고 불러오기

4) Functions & Technical Methodology of App – 카메라 촬영 및 사진 공유



- 위젯/컴포넌트/레이아웃
 - 1) 카메라 Button 위젯 : Intent를 사용, 기본 카메라앱 실행
 - 2) RecyclerView : MessageAdapter의 getItemViewType에 따른 ViewHolder를 설정하여 메시지 목록으로 나타냄
 - 3) imageView : 촬영한 이미지를 표시
- 카메라 촬영 및 사진 공유 기능
 - 1) 촬영한 이미지 Bitmap 형태로 가져와 Firebase Storage에 업로드
 - 2) 업로드한 이미지 URL을 문자열로 변환해 ChatData 객체 imageUrl에 저장하여 Firebase 참조 경로에 push
 - 3) Firebase 참조 경로에 event가 발생했을 때 변경된 데이터를 ChatData 객체로 가져와 MessageAdapter를 통해 메시지 표시
 - 4) 이미지 메시지 : ChatData 객체 msg는 null, imageUrl은 값 존재
- 서드파티 라이브러리
 - Firebase
 - 1) Firebase Storage : 촬영한 이미지를 업로드
 - 2) Firebase Realtime Database : 업로드한 이미지 URL을 ChatData database에 저장하고 불러오기
 - Glide
 - 1) 이미지를 로드하고 크기를 조정하여 imageView에 표시

5) Demo



[채팅방 화면 영상](#)



[Map 화면 영상](#)



[GitHub](#)

Thank You