스마트버스

– 원활한 대중교통 경험을 위해 실시간 버스 추적, 기내 채팅 기능 및 탐지캠기반 AI 군중 밀도 모니터링을 통합한 웹 플랫폼 –

웹개발 1팀
허준우 : 김성민 : 권칠윤 : 이백승

웹 사이트 주소 : https://du-smart-bus.web.app/

CONTENTS

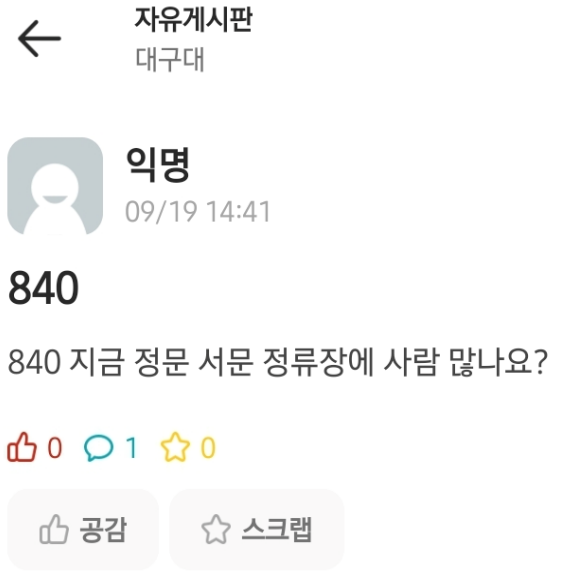
01

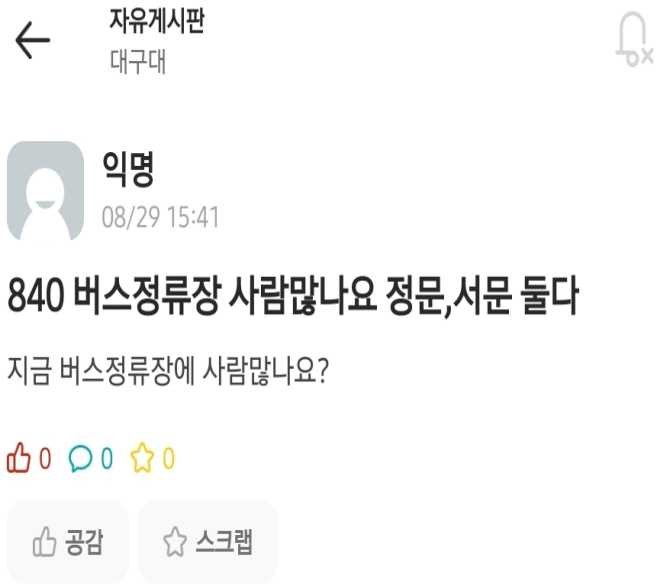
프로젝트 개요

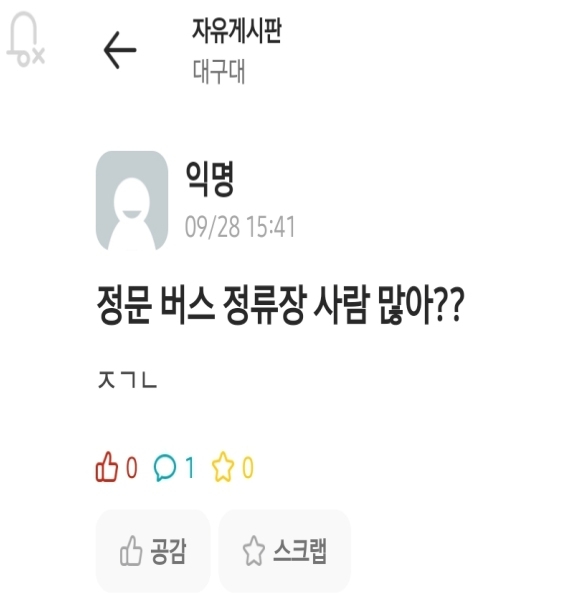
1-1 개발동기

1-2 기획의도

개발동기

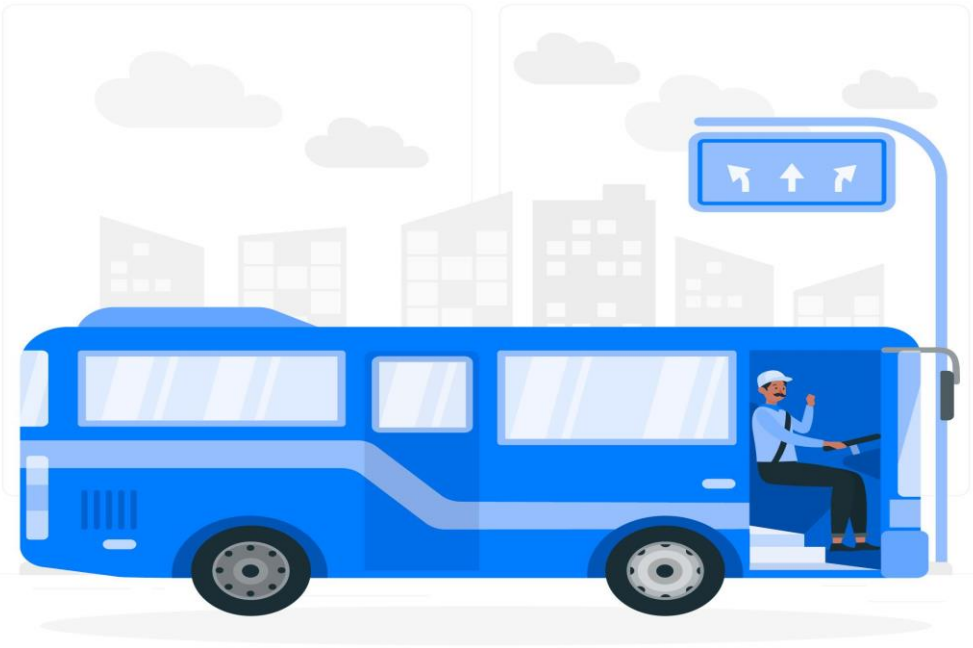






출처 에브리타임 대구대 학생들의 자유게시판 게시글

개발동기



사진출처 **Freepik**

기획의도

버스 내부의 혼잡도를 파악.

학생들의 대중 교통의 선택을 도움.

하교길을 쾌적하게 해주는 웹 서비스.

실시간 혼잡도 수집

데이터 수집 및 저장

혼잡도 수치 및 시각화

DATA 공공데이터포털 .GO.KR

실시간 버스위치 정보 통합

사진 출처 **Freepik**

02

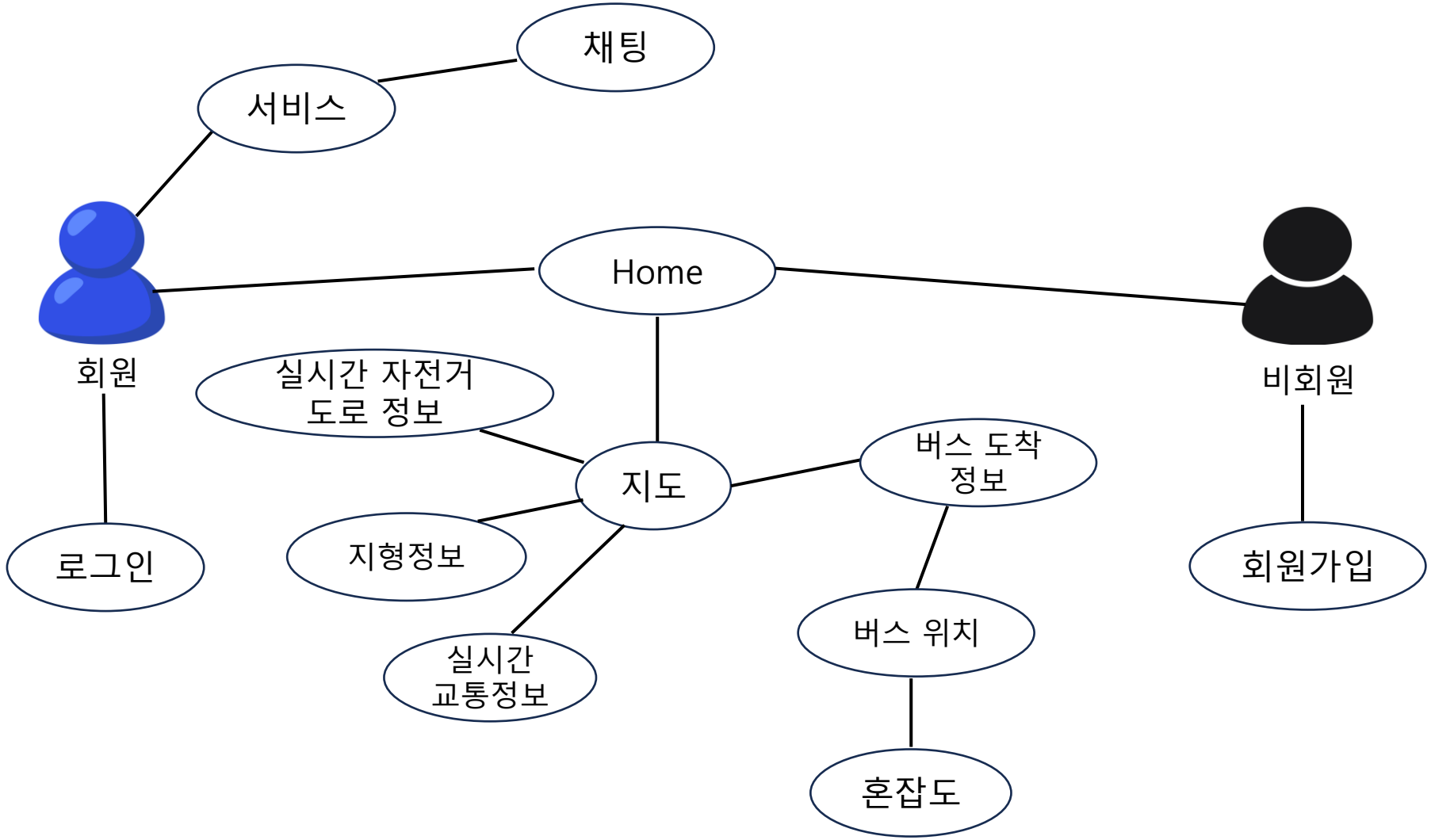
프로젝트 상세

2-1 구조도

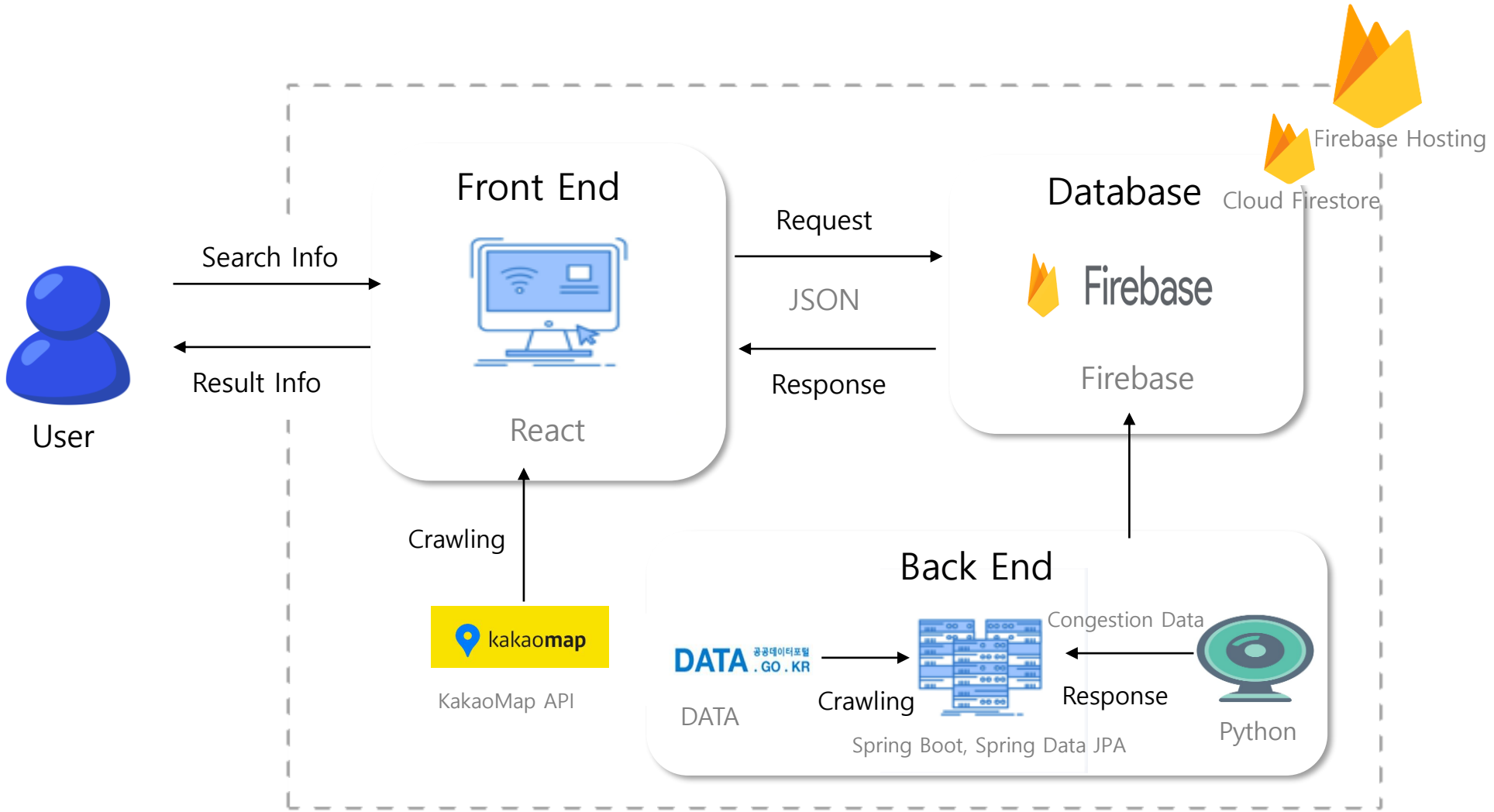
2-2 설계도

2-3 구현기능

구조도(기능구조)



구조도(서비스구조)



설계도(DB명세서)

City : 도시정보

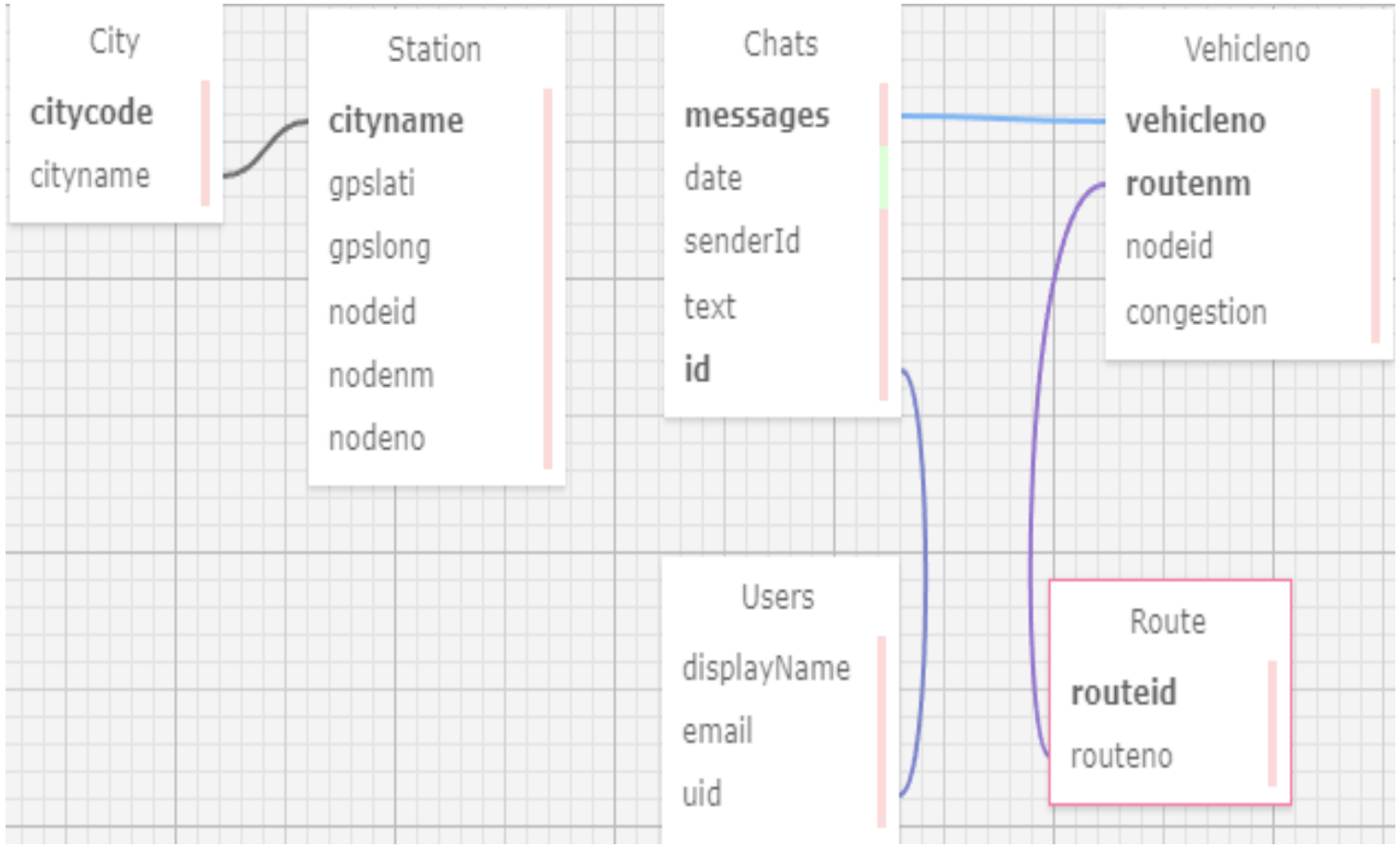
Station : 정류장 정보

Chats : 채팅 정보

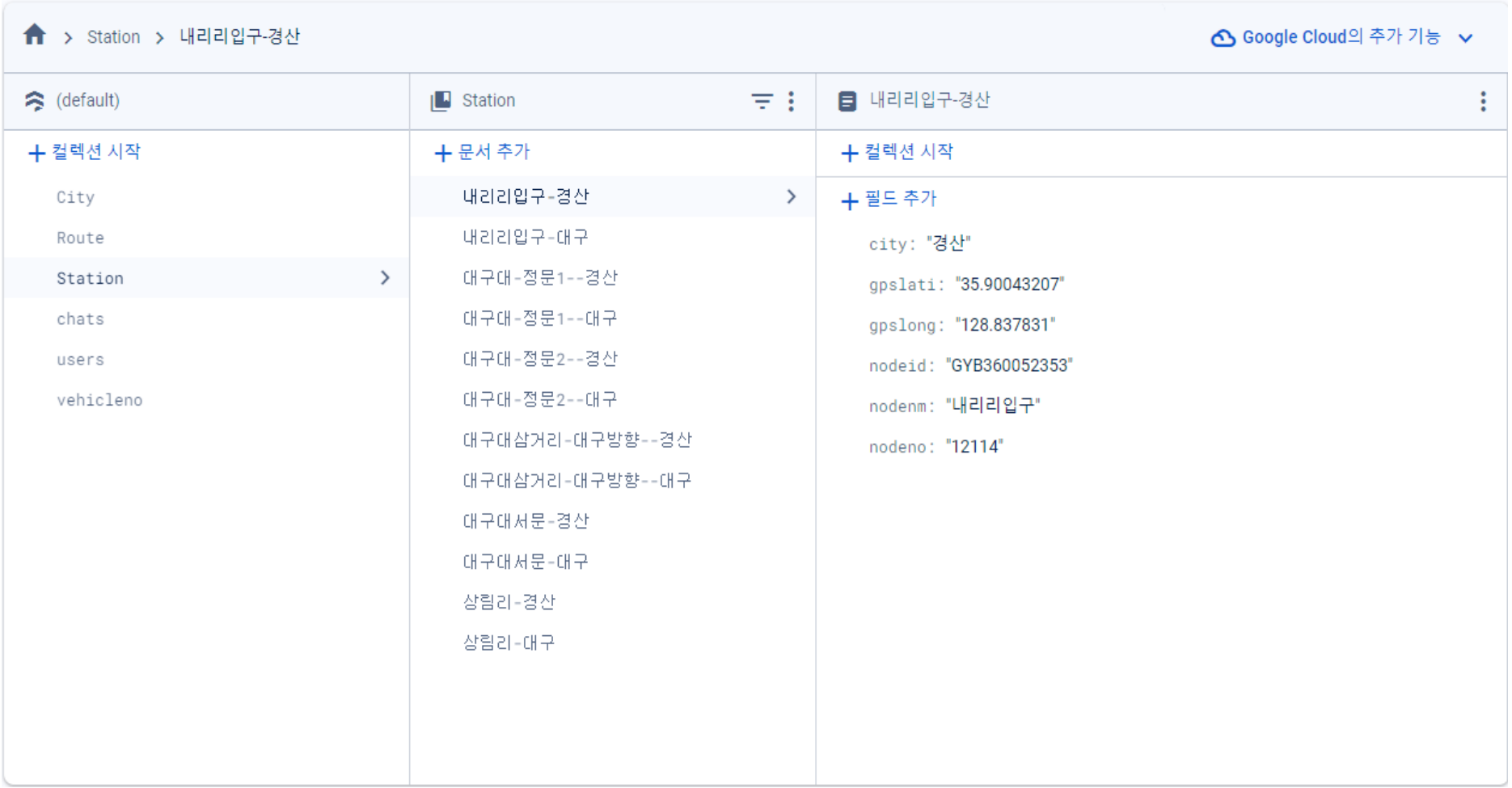
Vehicleno: 차량 정보

Users: 회원 정보

Route: 노선 정보



설계도(Firestore Database)



설계도(API명세서/City,Route)

CityController.java

Index	Method	URI	Description
1	POST	/create/city	도시 추가 API
2	GET	/get/city	도시 조회 API
3	PUT	/update/city	도시 수정 API
4	DELETE	/delete/city	도시 삭제 API

RouteController.java

Index	Method	URI	Description
1	POST	/create/route	노선 추가 API
2	GET	/get/route	특정 노선 조회 API
3	GET	/get/route/all	모든 노선 조회 API
4	PUT	/update/route	노선 수정 API
5	DELETE	/delete/route	노선 삭제 API

설계도(API명세서/Station,Vehicle)

StationController.java

Index	Method	URI	Description
1	POST	/create/station	정류장 추가 API
2	GET	/get/station	특정 정류장 조회 API
3	GET	/get/station/all	모든 정류장 조회 API
4	PUT	/update/station	정류장 수정 API
5	DELETE	/delete/station	정류장 삭제 API

VehicleController.java

Index	Method	URI	Description
1	POST	/create/station	정류장 추가 API
2	GET	/get/station	특정 정류장 조회 API
3	GET	/get/station/all	모든 정류장 조회 API
4	PUT	/update/station	정류장 수정 API
5	DELETE	/delete/station	정류장 삭제 API

설계도(API명세서/Chats)

ChatsController.java

Index	Method	URI	Description
1	POST	/create/chats	채팅방 추가 API
2	GET	/get/chats	특정 차량에서 보낸 채팅 조회 API
3	GET	/get/chats/all	모든 채팅 조회 API
4	PUT	/update/chats	채팅방 수정 API
5	DELETE	/delete/chats	채팅방 삭제 API
6	DELETE	/delete/text/{vehicleno}	특정 시간마다 특정 차량 채팅로그 삭제API
7	GET	/update/time/{time}	채팅로그 삭제 시간 설정API

구현기능(메인 화면)

■ 채팅방 검색

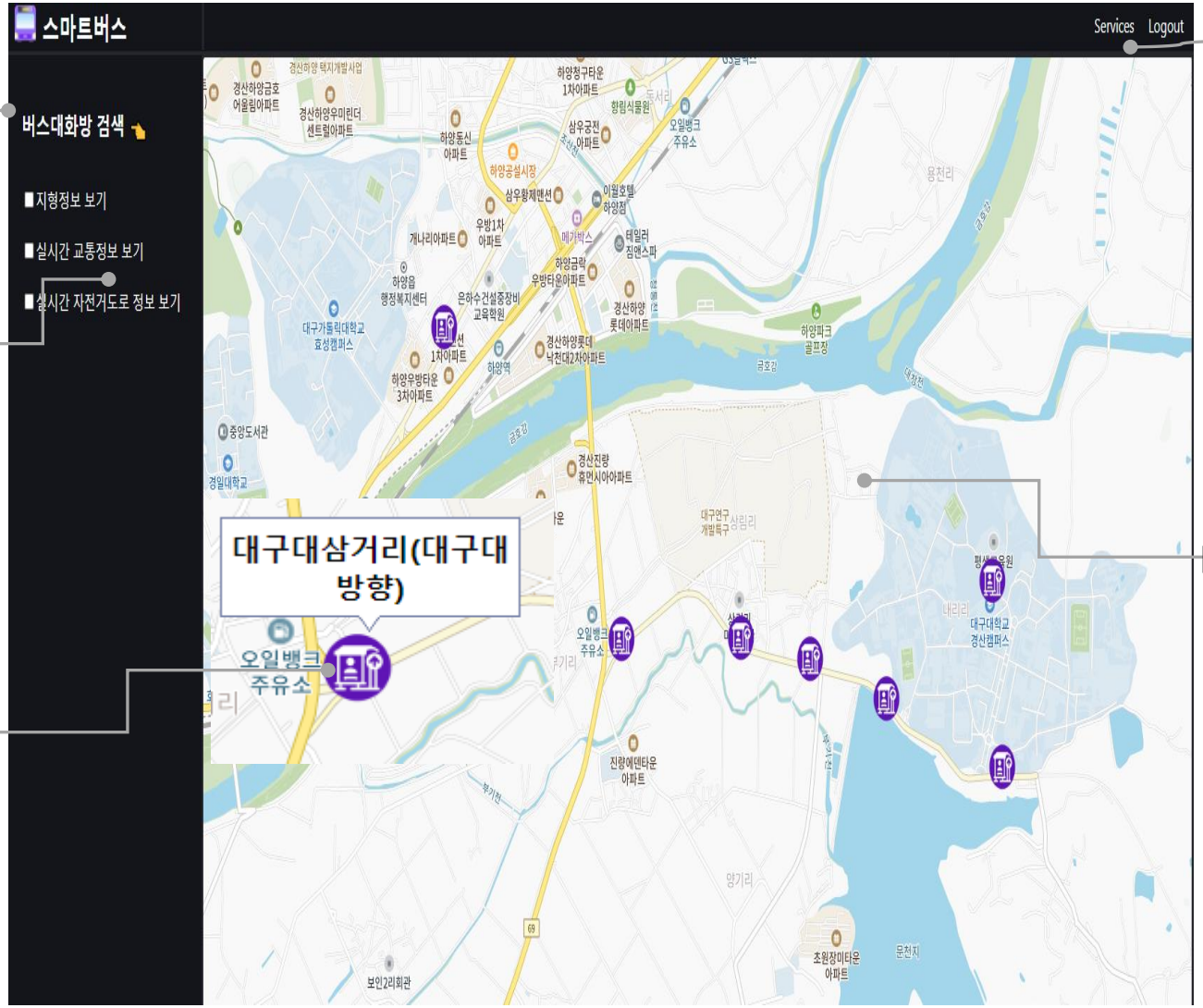
-차량번호 검색
-차량번호 별 채팅방 입장

■ 지도 정보 기능

-지형정보 보기
-실시간 교통정보 보기
-실시간 자전거도로 정보 보기
-대구대 주변으로 좌표 매핑
-카카오맵 API 사용

■ 정류장 정보

-mouseover시 버스 정류장 정보 display 활성화
-대구대 주변 정류장 아이콘 매핑



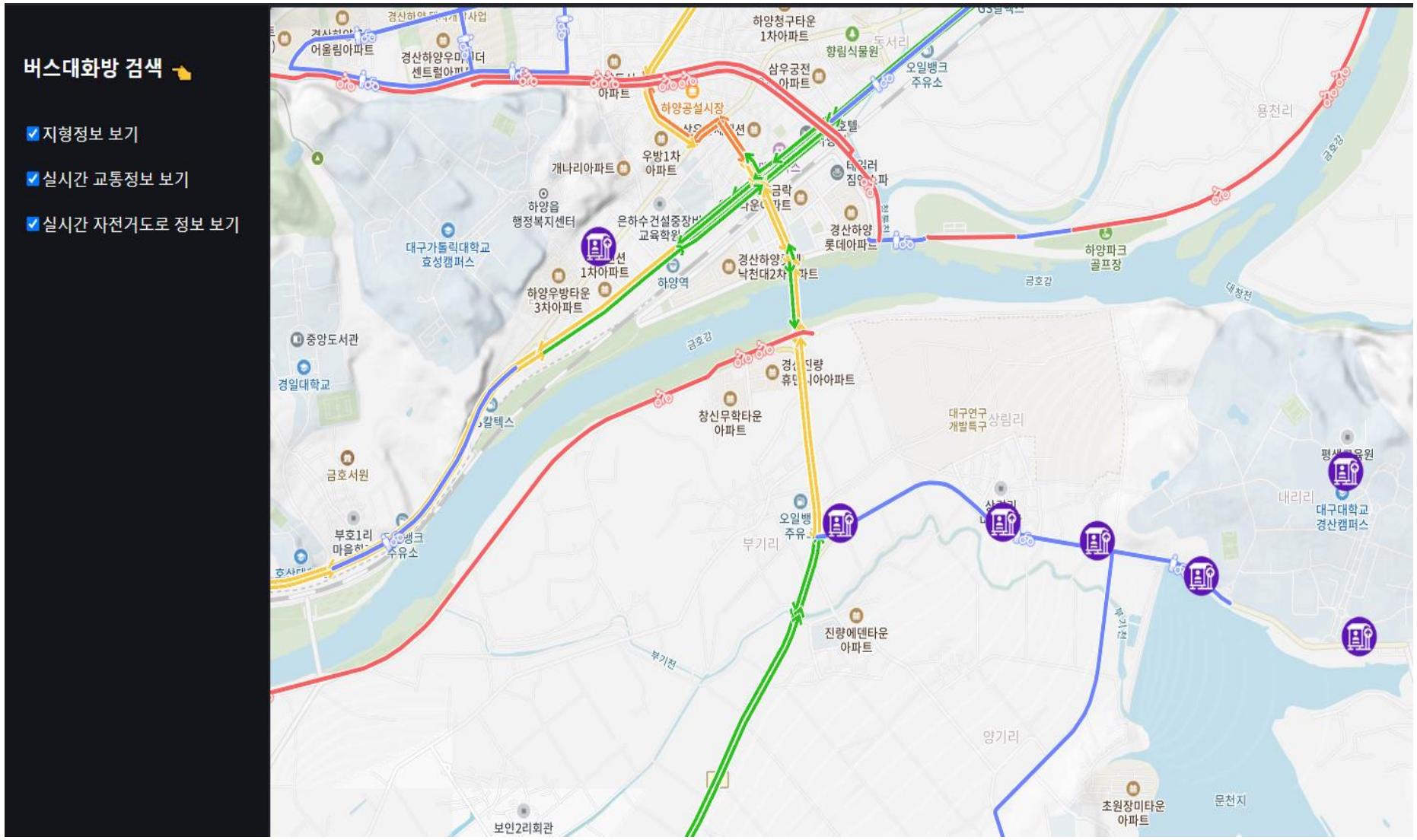
■ 회원 기능

Login/Logout/Register 기능 활성화

■ 버스 도착 정보 기능

-정류장 아이콘 클릭시 버스 도착 정보 display 활성화
-실시간 버스 정보 보기
-공공포털데이터 API 사용

구현기능(상세 지도 정보 화면)



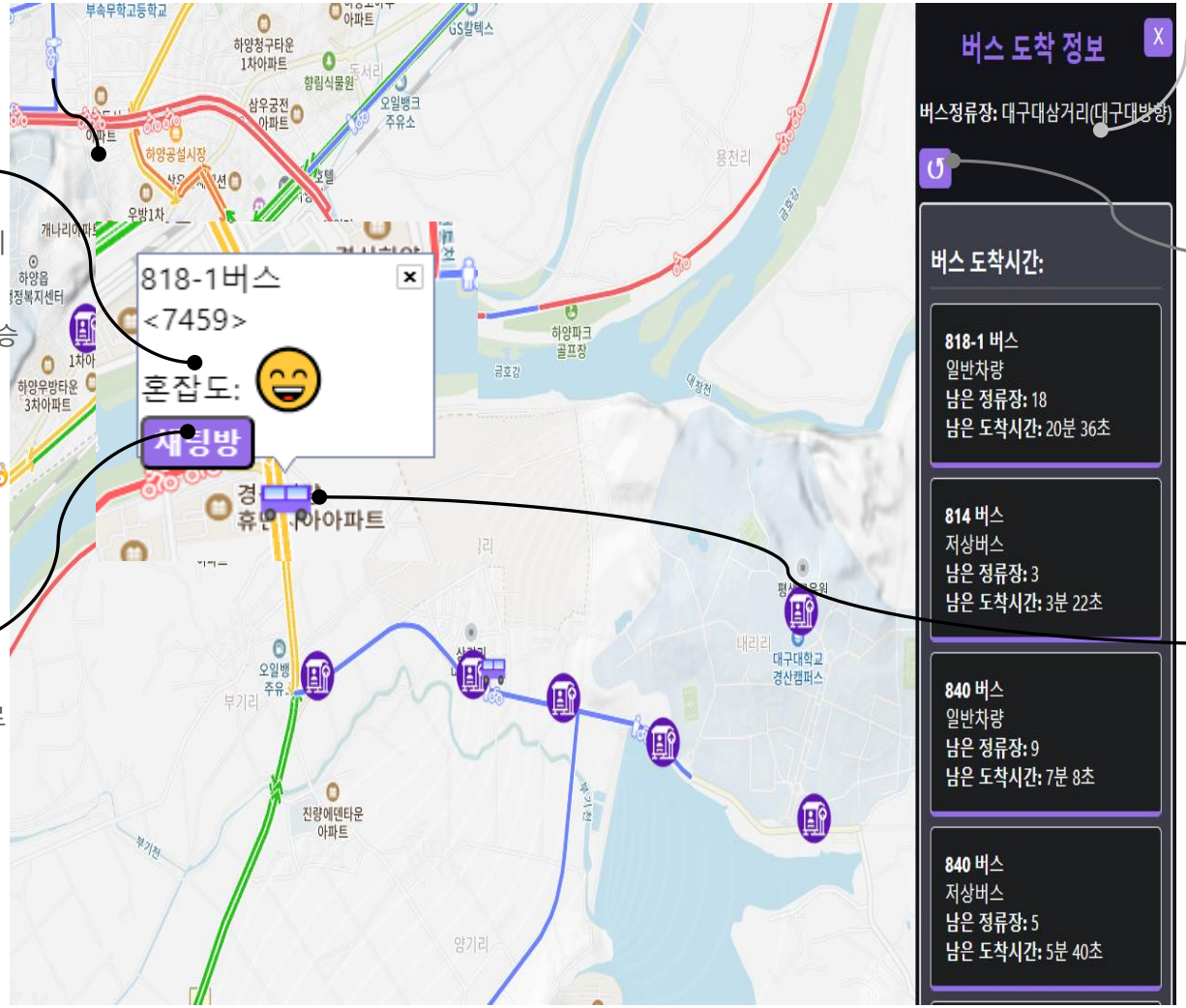
구현기능(버스 도착/혼잡도 정보 화면)

■ 혼잡도 정보

-선택한 차량의 혼잡도 표시
-혼잡도는 아이콘으로 매핑
-대구 시내 버스 기준 45인승
15명 이하 여유 😄
16~30명 보통 😐
31~45명 혼잡으로 표시 🤯

■ 채팅방

-해당 차량번호의 채팅방으로 이동



■ 버스 도착 정보

-선택한 버스 정류장의 이름 표시
-각 버스들 마다 남은 정류장과 시간 표시

■ 새로고침

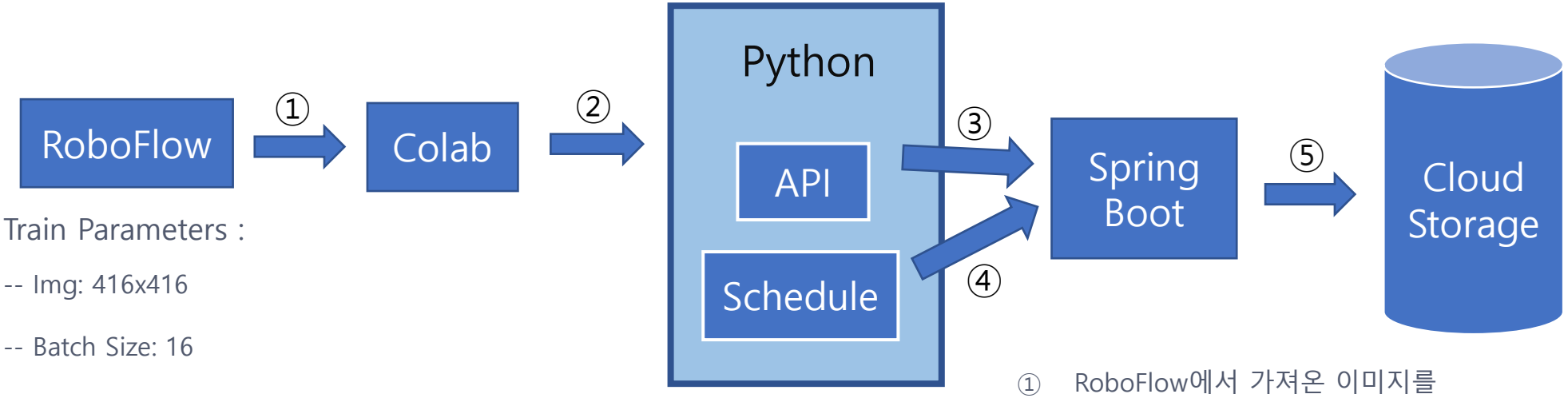
-버스 도착 정보를 실시간으로 가져옴

■ 버스 위치 정보

-선택한 버스의 실시간 위치 정보 표시
-버스의 위치는 버스 아이콘으로 매핑

구현기능(탐지캠을 이용한 AI 사람인식)

사람 인식(혼잡도 분석)



Train Parameters :

-- Img: 416x416

-- Batch Size: 16

-- Epochs: 50

Train Model : Yolov5s

API : http://[server url]/update/congestion/{vehicleno}/{congestion} (put)

Schedule : 매 5초 마다 실행

Storage : 서버 상의 /home/vehicleno/{vehicleno}/congestion 디렉토리

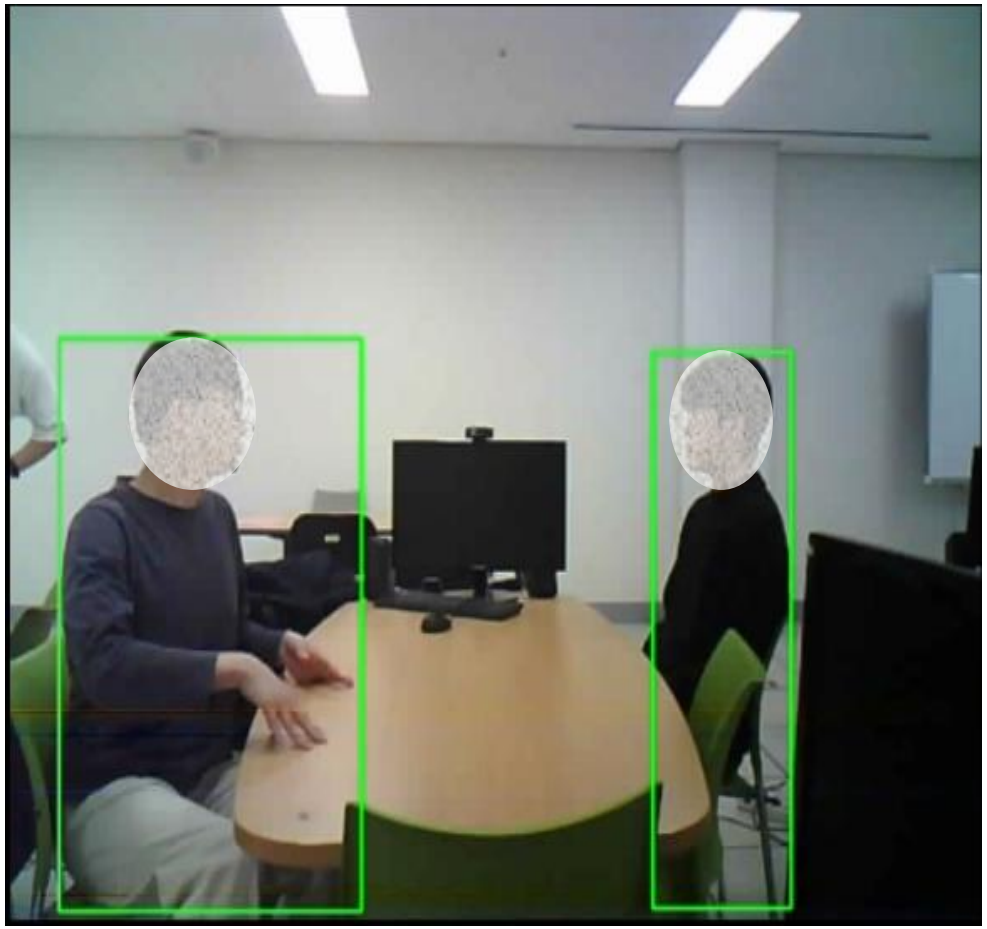
① RoboFlow에서 가져온 이미지를 Colab을 통해 학습

② 학습을 통해 나온 결과물' best.pt' 을 로컬에 저장, PyTorch를 사용해 로컬에 저장된 모델을 Python에 import

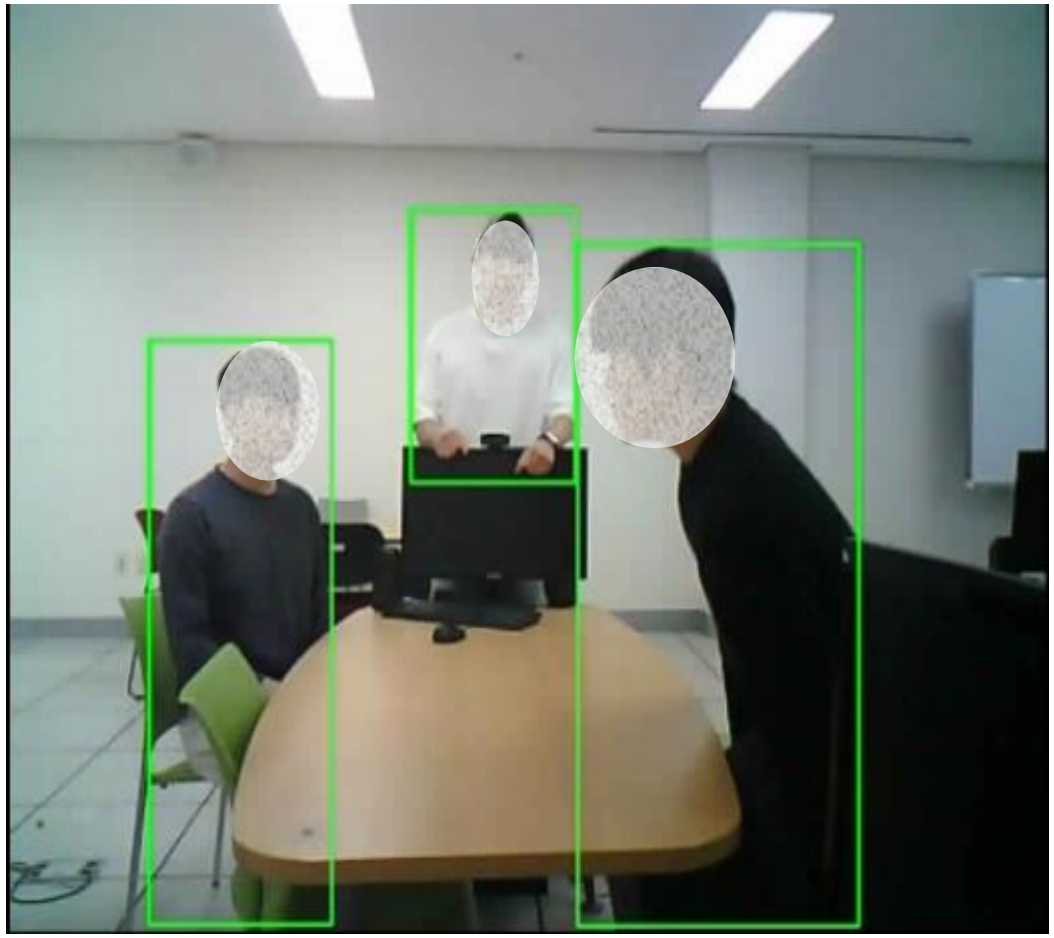
③ 스케줄에 지정된 시간 5초마다 혼잡도 갱신

④ Request받은 혼잡도를 Cloud Storage에 Request

구현기능(탐지캠 사람인식 화면)

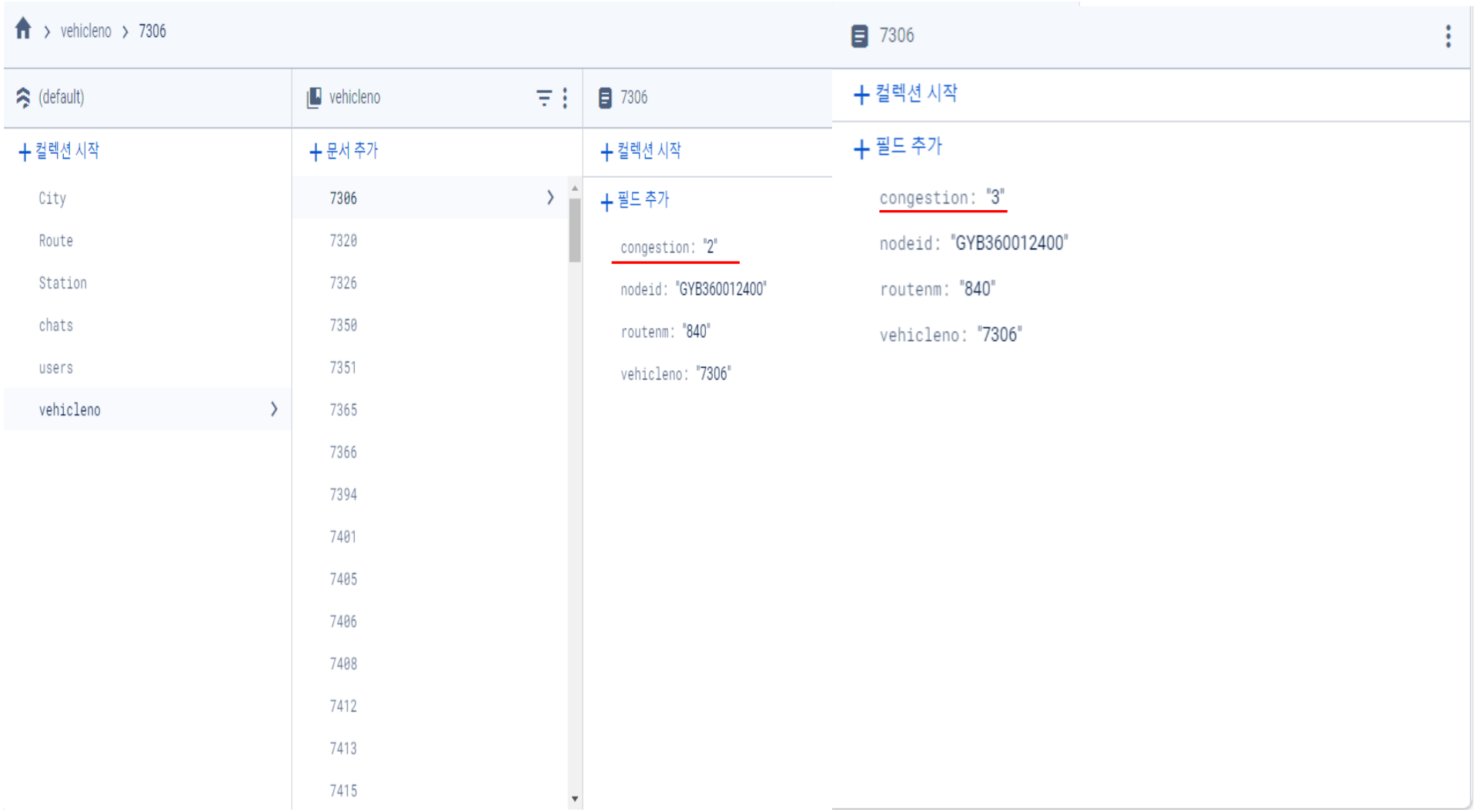


-- Congestion : 2

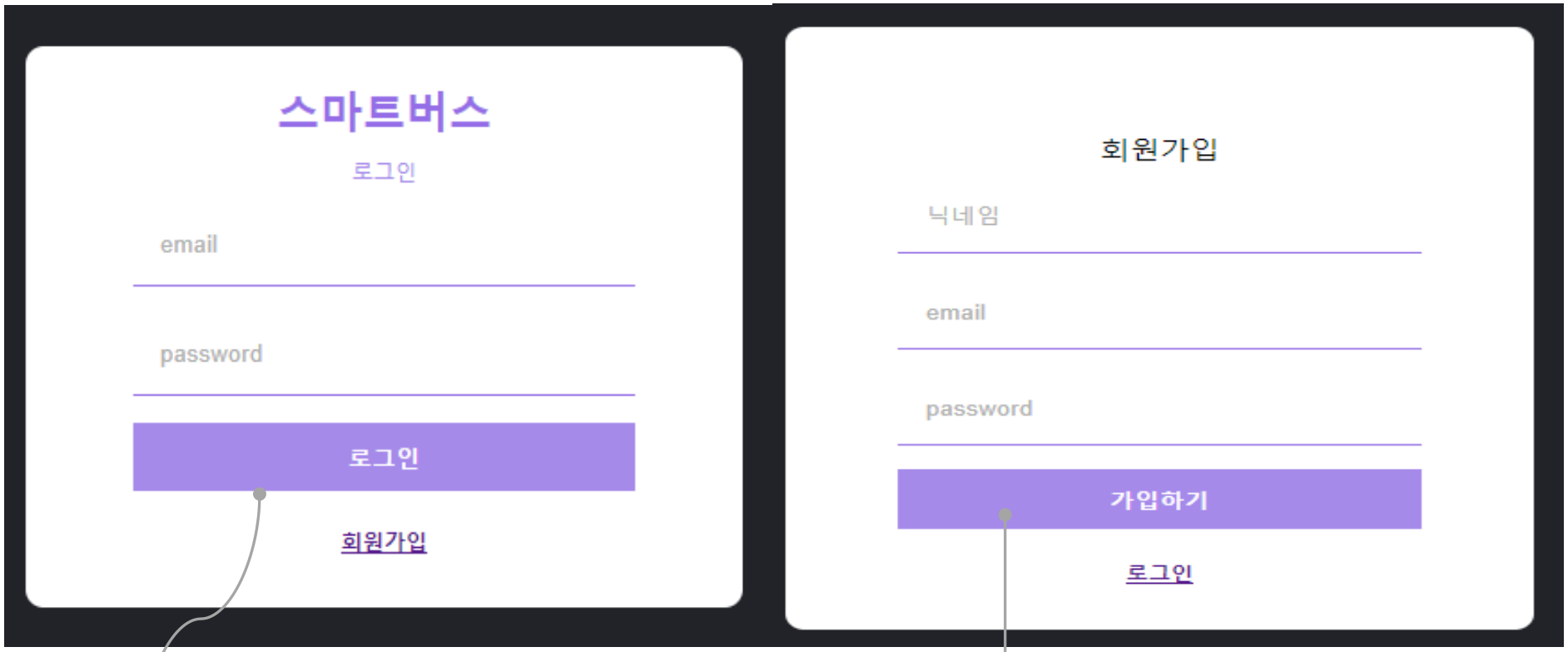


-- Congestion : 3

구현기능(혼잡도 DB 화면)



구현기능(로그인/회원가입 화면)



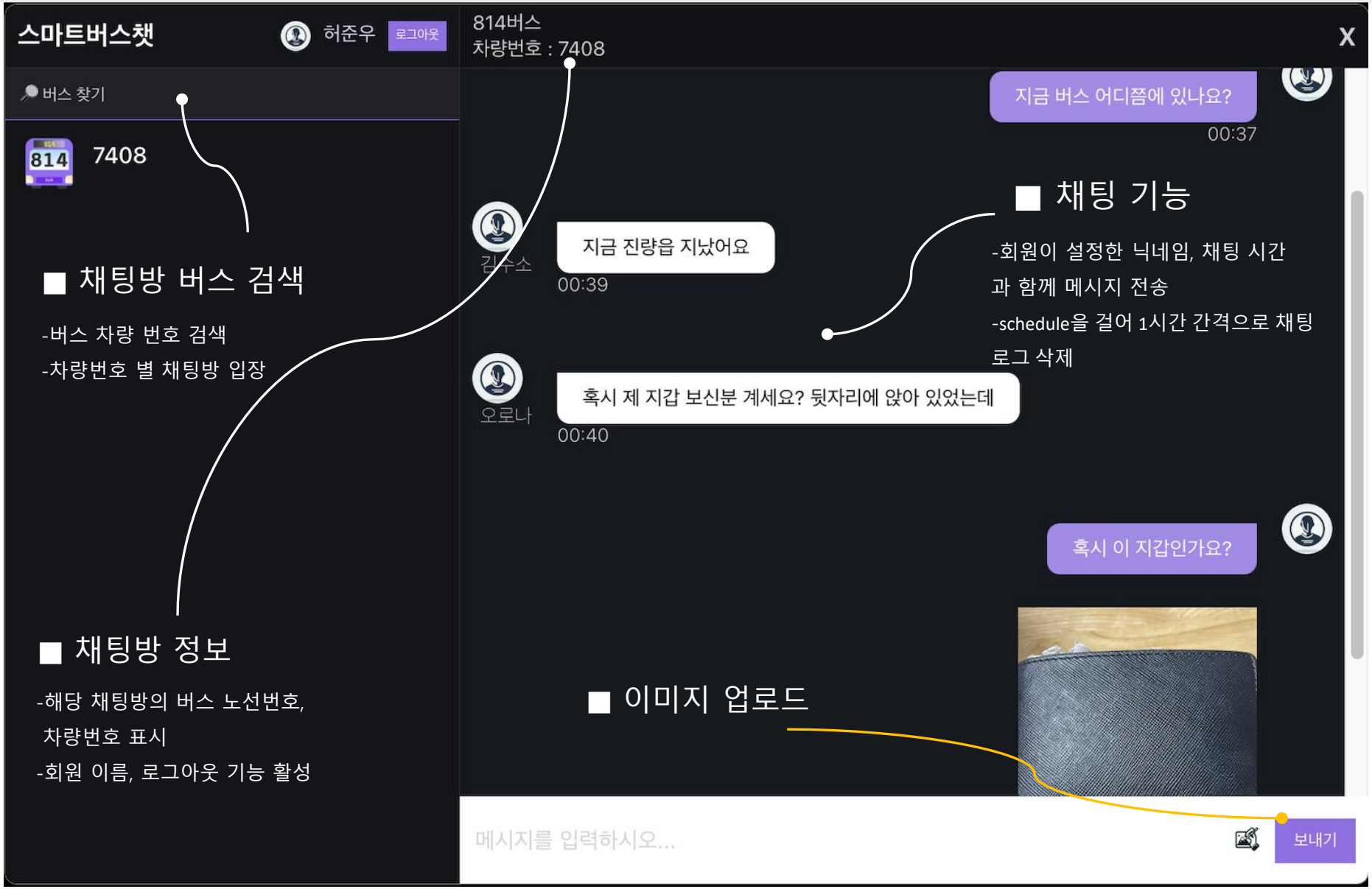
■ 로그인 기능

-스마트버스 홈페이지 회원가입 후 로그인을 진행할 수 있다.

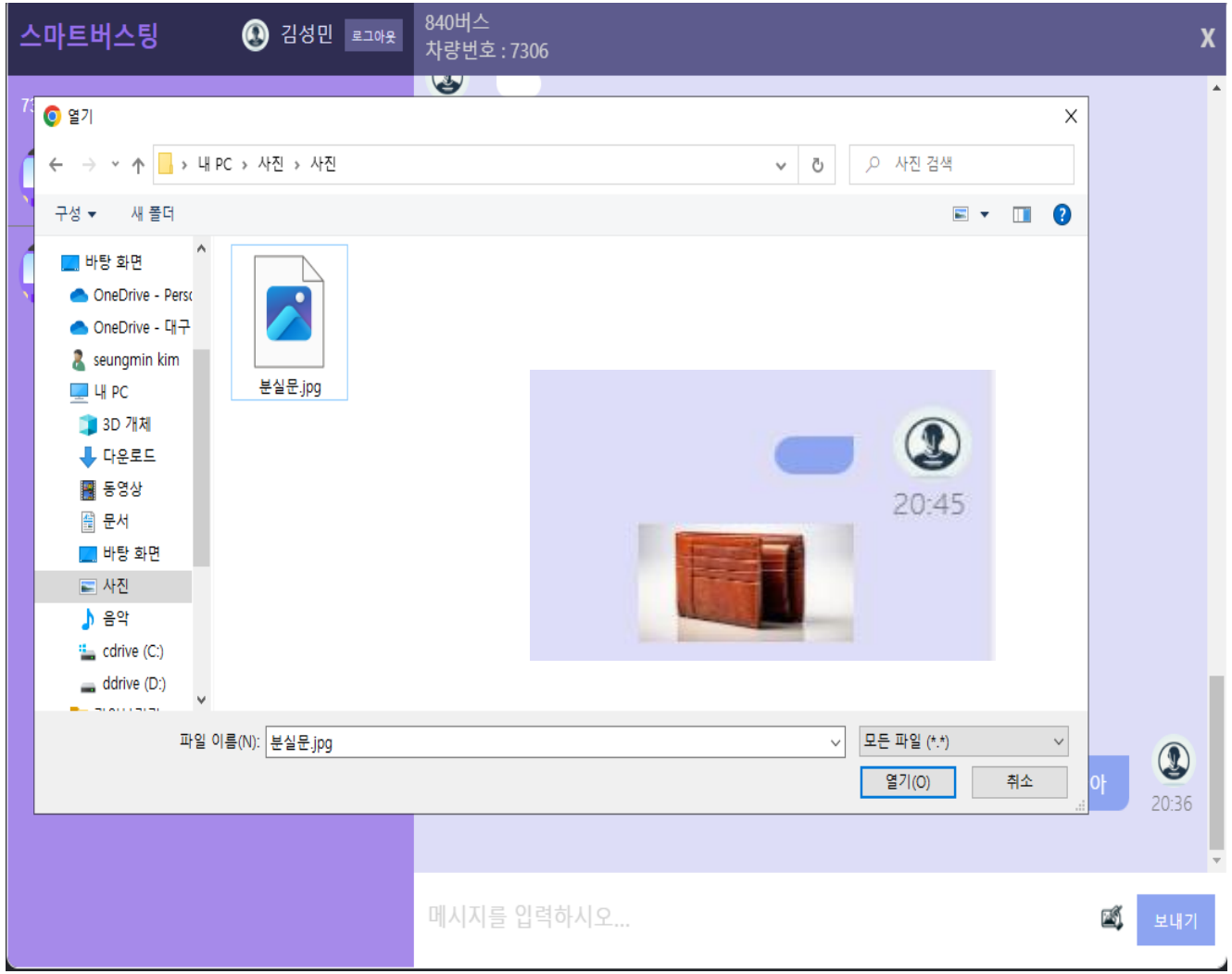
■ 회원가입 기능

-회원가입후 스마트버스 홈페이지의 채팅 서비스를 이용할 수 있다.

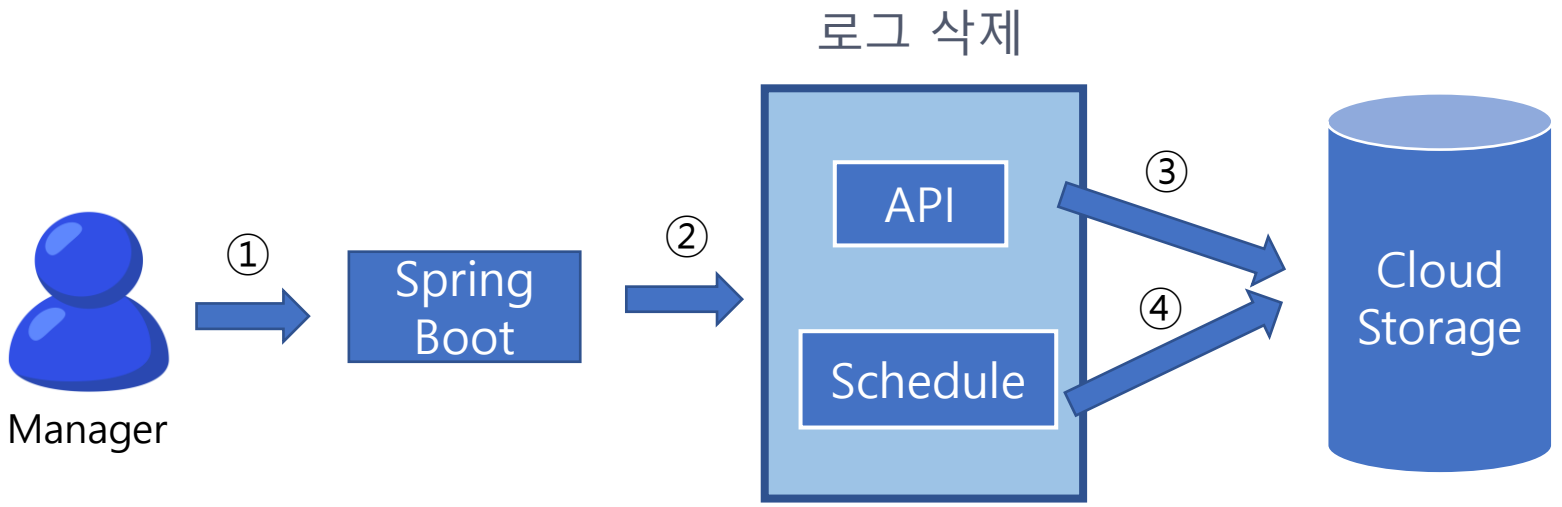
구현기능(채팅방 화면)


스마트버스챗
허준우
로그아웃
버스 찾기
7408
814
814버스
차량번호 : 7408
X
지금 버스 어디쯤에 있나요?
00:37
지금 진량읍 지났어요
00:39
혹시 제 지갑 보신분 계세요? 뒷자리에 앉아 있었는데
00:40
혹시 이 지갑인가요?
메시지를 입력하시오...
보내기
■ 채팅방 버스 검색
-버스 차량 번호 검색
-차량번호 별 채팅방 입장
■ 채팅 기능
-회원이 설정한 닉네임, 채팅 시간 과 함께 메시지 전송
-schedule을 걸어 1시간 간격으로 채팅 로그 삭제
■ 채팅방 정보
-해당 채팅방의 버스 노선번호, 차량번호 표시
-회원 이름, 로그아웃 기능 활성
■ 이미지 업로드

구현기능(채팅방 화면/분실문 사진 업로드)



구현기능(차량별 채팅기록 로그 삭제)



API : http://[server url]/delete/text/{vehicleno} (delete)

Schedule : 매 시간 마다 실행

Storage : 서버 상의 /home/chats/{vehicleno} 디렉토리

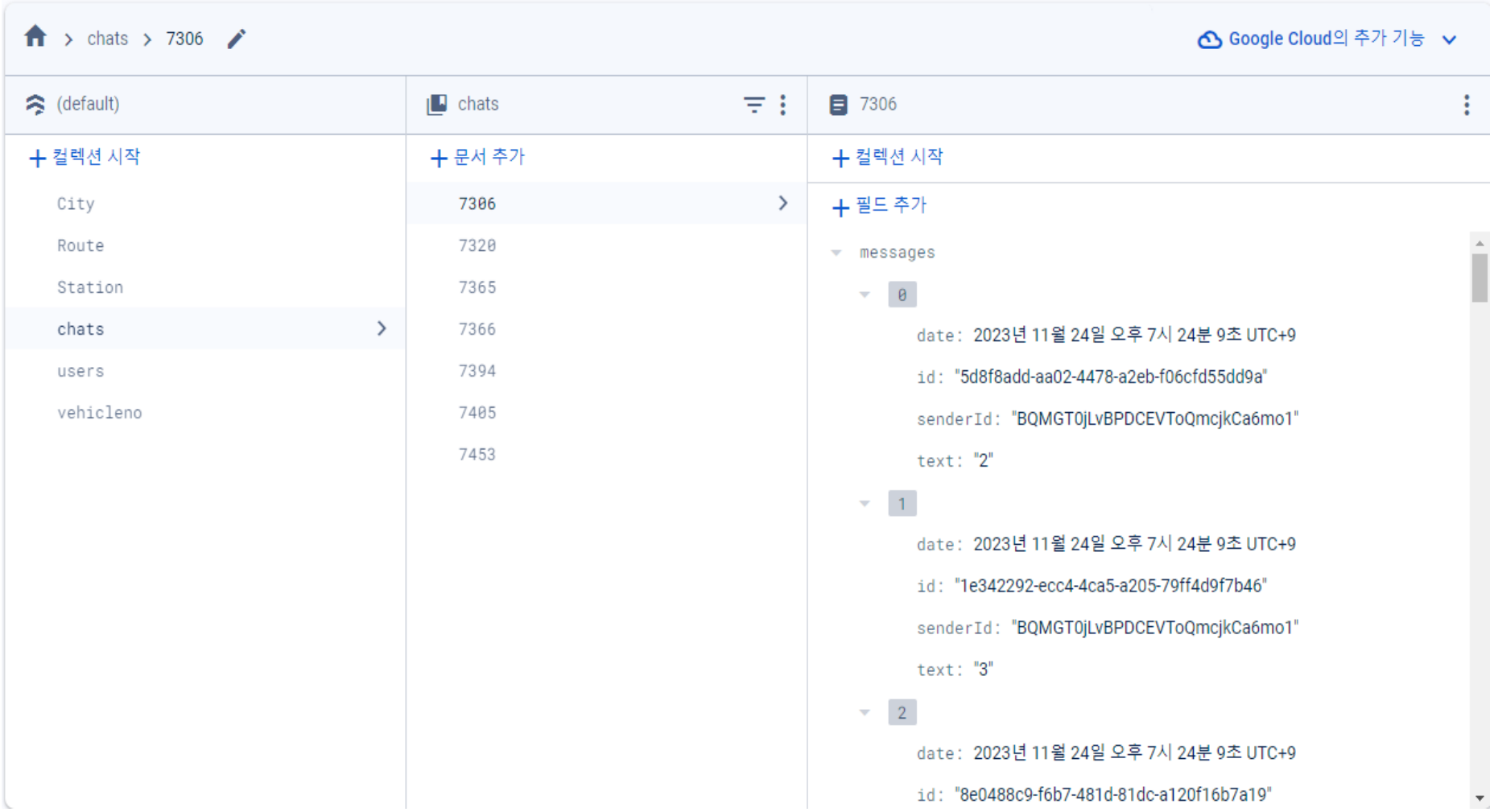
① 관리자가 로그 삭제를 위해 서비스에 접속

② 관리자가 설정한 setTime 값으로 Request

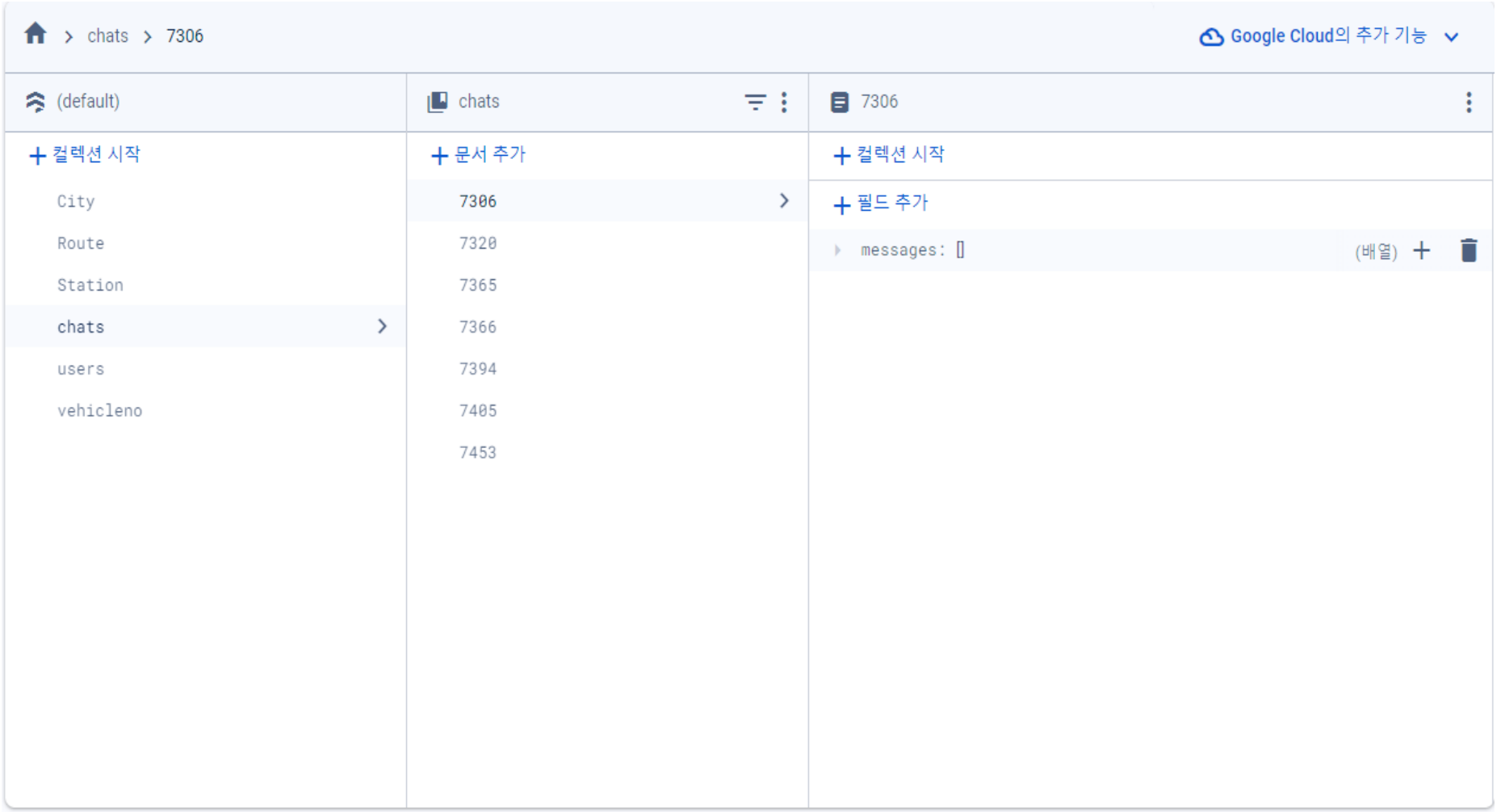
③ 삭제API를 Cloud Storage에 Request

④ 스케줄에 지정된 매 시간마다 해당 차량 번호 채팅방의 setTime 값 이후 해당 채팅 삭제

구현기능(차량별 채팅기록 로그 삭제 화면)



구현기능(차량별 채팅기록 로그 삭제 화면)



03

프로젝트 시연

3-1 시연영상

시연영상

3-1. 시연영상

링크 : https://youtu.be/5u2toMvL5LI

04

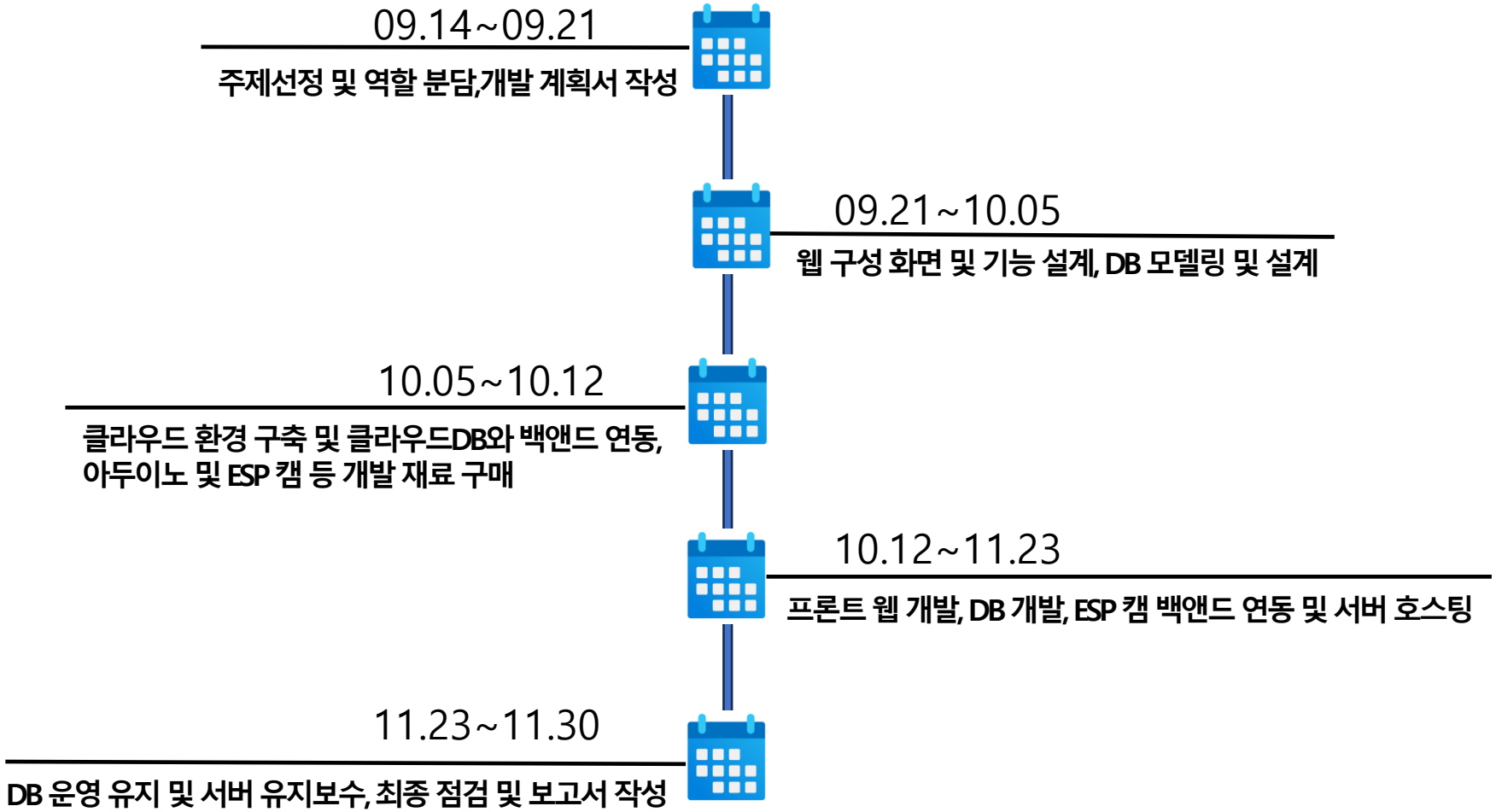
진행 프로세스

4-1 개발일정

4-2 개발환경

4-3 마치며

개발일정



개발환경

Back-End

spring boot
Maven
PyTorch
python
Java
OpenCV

Front-End

React
HTML
CSS
JS
kakaomap
JSX

Tools

GitHub
ARDUINO
IJ
colab
ESP32-CAM

Etc.

DATA 공공데이터포털 .GO.KR
Google
Firebase
roboflow

마치며

웹개발 1팀

권칠윤

-DB설계 및 프로그래밍

김성민

-DB설계 및 프로그래밍 보조

LEADER

허준우

-프론트 웹 개발

이백승

-아두이노 캠 탐지 기능 구현

Thank you