

# 시각장애인을 위한 버스 탑승 시스템 'We Tayo'

Development of Bus Boarding Service  
for the Visually Impaired 'We Tayo'

팀장 : 강석원  
팀원 : 박형근, 이지수, 홍의성

## 목 차

### 1 졸업 연구 개요

- 1.1 연구 개발 배경
- 1.2 연구 개발 목표

### 2 관련 연구 및 사례

- 2.1 관련 연구 및 사례1
- 2.2 관련 연구 및 사례2
- 2.3 비교 분석

### 3 시스템 수행 시나리오

- 3.1 시스템 수행 시나리오

### 4 시스템 구성도

- 4.1 하드웨어 시스템 구성도
- 4.2 소프트웨어 시스템 구성도

### 5 개발 환경 및 방법

- 5.1 하드웨어 시스템 개발 환경
- 5.2 소프트웨어 시스템 개발 환경
- 5.3 시스템 개발 방법

### 6 업무 분담 및 수행 일정

- 6.1 업무 분담
- 6.2 수행 일정

### 7 참고 문헌

- 7.1 참고 문헌

### 8 질의응답

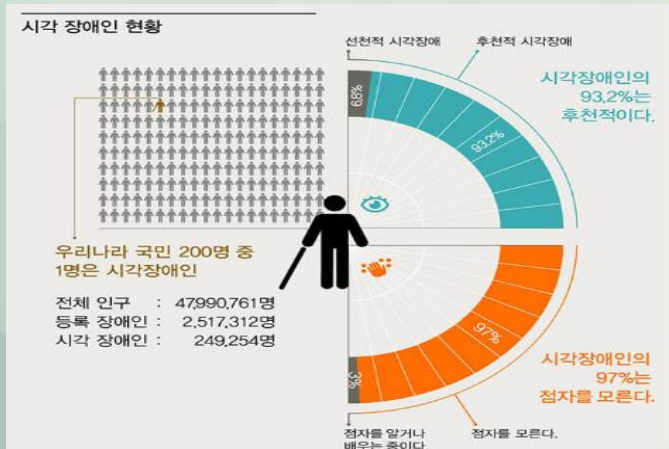
- 8.1 질의응답

# 1

## 졸업 연구 개요

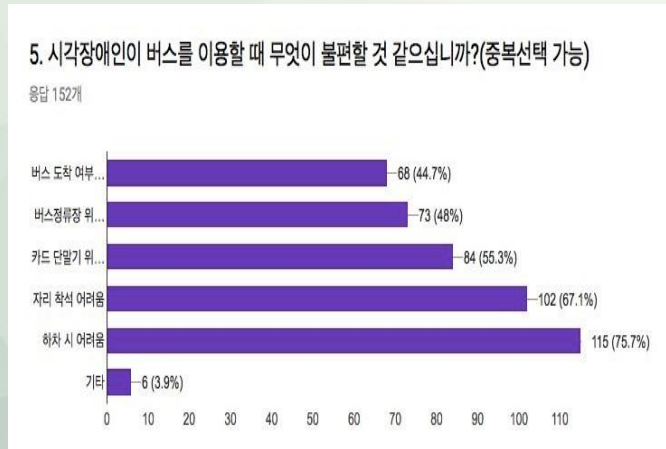
- 1.1 연구 개발 배경
- 1.2 연구 개발 목표

# 1.1 연구 개발 배경



## 국내 시각 장애인 현황

우리나라 국민 200명 중 1명은 시각장애인  
그 중 93.2% 후천적 시각장애를 갖고 있음



## 시각 장애인의 버스 이용시 불편사항

시각 장애인이 버스를 이용할 때 가장 불편함을 느끼는 1순위는 '하차 시 어려움'이다.  
<오마이 뉴스 2017년 조사자료>



## 시각장애인의 버스 탑승 국민 청원 내용

시각장애인의 이동권 보장을 요구하는 국민 청원 내용 3738명의 청원  
<2019.11.27 청원 시작>

## 1.1 연구 개발 배경



“일단 버스를 타고 이동하는 것은 고사하고 내가 원하는 버스를 기다리는 시간만 평균 40분 ~ 50분 소요된다.”

(MBC 소수의견 인터뷰 中)

**이동권**은 시각장애인의 사회참여를 위해서 반드시 보장되어야 하는 **기본권리**



“기존의 버스 알림 어플, 정류장 음성 시스템은 오류가 잦고 업데이트가 느려 실질적으로 도움이 되지 않는다.”

(한국 시각장애인 연합회 참고 의견 中)

그들도 우리와 똑같은 사회구성원으로서 인정받으면서 살아가야함

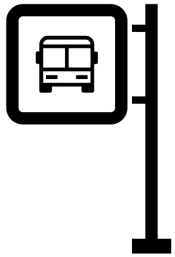


“우리(시각 장애인)들도 비장애인들과 다름없이 남 눈치 보지 않으면서 일상적인 삶을 사라고 싶다.”

(청와대 국민청원 발췌문 中)

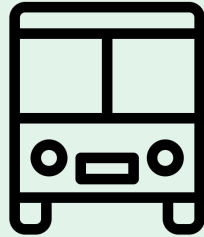
시각장애인의 주도적인 버스 이용을 위해 “WeTayo”를 개발 할 필요성을 느낌

## 1.2 연구 개발 목표



### 위치 정보 분석

사용자(시각장애인)의 위치와 버스 정류장의 위치를 정확하게 파악하여 가장 가까운 정류장에 쉽게 접근할 수 있도록 유도



### 탑승 여부 전송

사용자(시각장애인)의 APP에서 탑승을 희망하는 버스를 선택하면 가장 먼저 도착하는 버스 기사님의 APP의 탑승 여부를 전송



### APP으로 하차벨 동작

기존 버스에 장착된 하차벨과 비콘센서를 연동하여 사용자(시각장애인)의 APP에서 무선으로 동작



### UI/UX 적극 반영

사용자가 시각장애인인 점을 고려하여 사용자 경험을 적극 반영

# 2

## 관련 연구 및 사례

- 2.1 관련 연구 및 사례1
- 2.2 관련 연구 및 사례2
- 2.3 비교 분석

## 2.1 관련 연구 및 사례1



### 시민기자단 솔루션

- 음성 수신기를 통해 버스 도착과 승차 위치 알림
- 음성 수신기를 통해 버스 기사에게 하차 요청
- 음성 수신기를 통해 정류장의 정보 알림
- 실시간 버스 정보 확인 불가
- 사용자 위치에 기반한 근처 정류장 선별 안함

<https://www.youtube.com/watch?v=saoX-IR-iJ0>



## 2.2 관련 연구 및 사례2



### 삼성 투모로우 솔루션 (시각 장애인 지우 이야기)

- APP을 통해 버스 선택 후 도착 알림
- APP을 통해 버스 기사에게 하차 요구 알림
- 실시간 버스 정보 확인 가능
- 사용자 위치에 기반한 근처 정류장 선정 안함

<https://www.youtube.com/watch?v=gqGaf2EalBs>

## 2.3 비교 분석

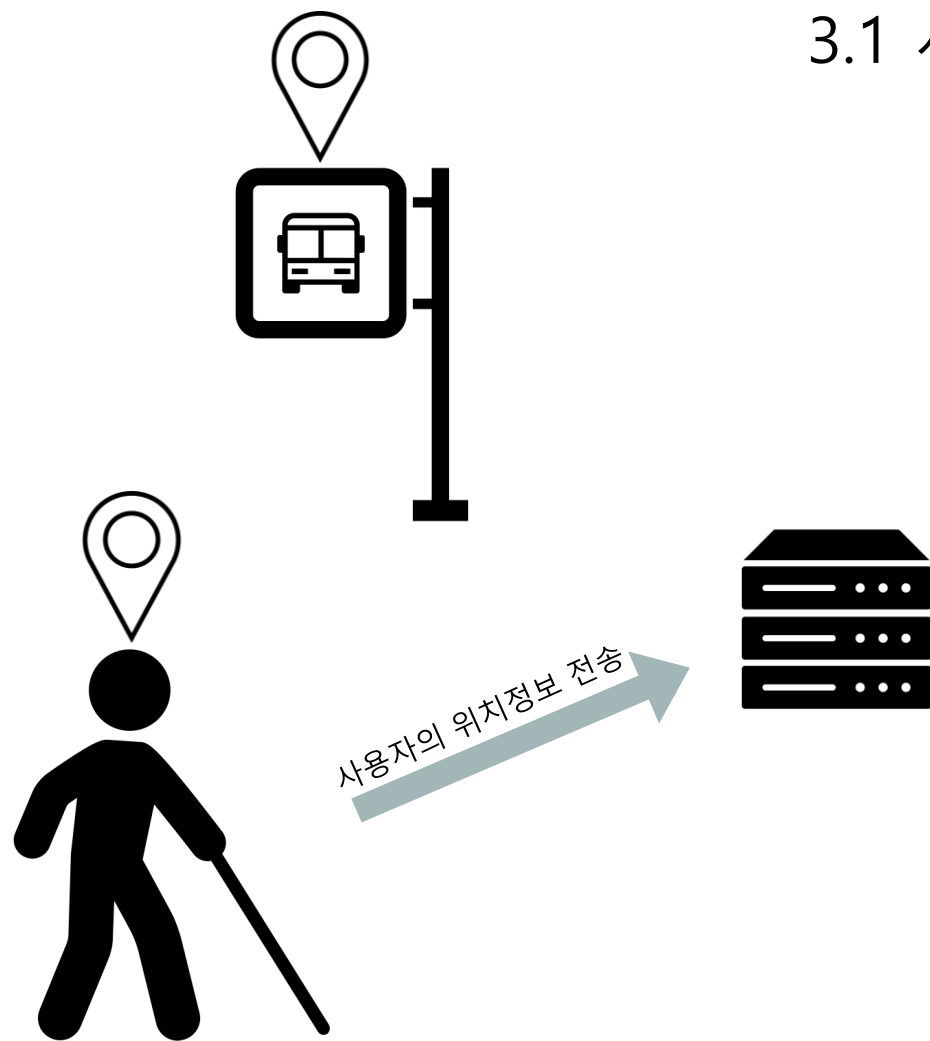
	버스정보 알림 스피커	무선 하차벨	실시간 버스 정보	APP을 통한 하차벨 동작	사용자 위치기반 정류장 선정
<b>함께 Tayo</b>	○	○	○	○	○
삼성 투모로우 솔루션	○	○	○	○	X
시민 기자단 솔루션	○	○	X	X	X

# 3

## 시스템 수행 시나리오

### 3.1 시스템 수행 시나리오

## 3.1 시스템 수행 시나리오



01 사용자가 버스 정류장에 접근하면 정류장 GPS와 사용자의 스마트폰 GPS 정보를 비교해 가장 가까운 정류장을 식별한다.

## 3.1 시스템 수행 시나리오



01 사용자가 버스 정류장에 접근하면 정류장 GPS와 사용자의 스마트폰 GPS 정보를 비교해 가장 가까운 정류장을 식별한다.

02 가장 가까운 정류장으로 식별된 정류장의 위치 정보를 사용자 앱에 전송

## 3.1 시스템 수행 시나리오

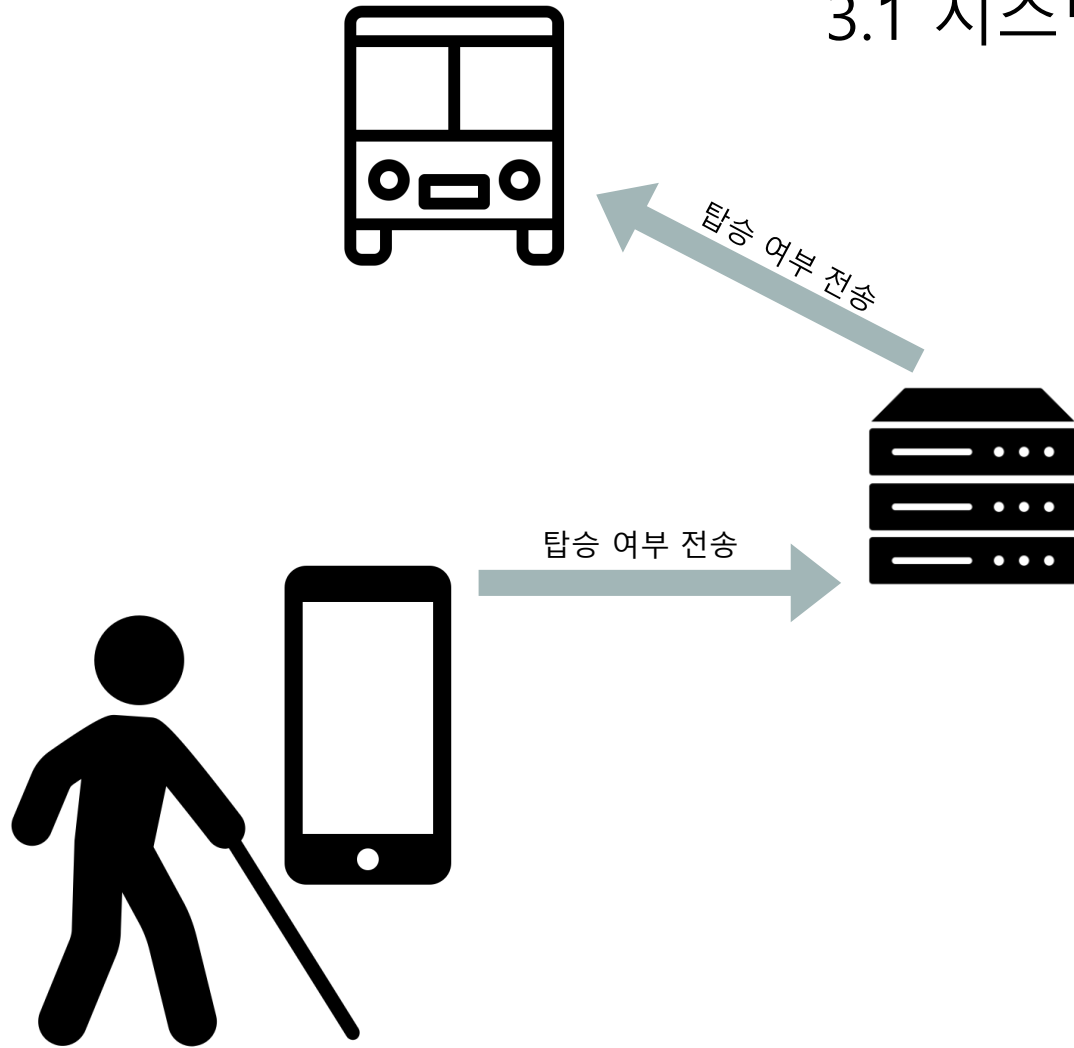


정류장의 도착 버스 전송

A grey arrow pointing from the cloud icon towards the smartphone, indicating the flow of data from the server to the user's device.

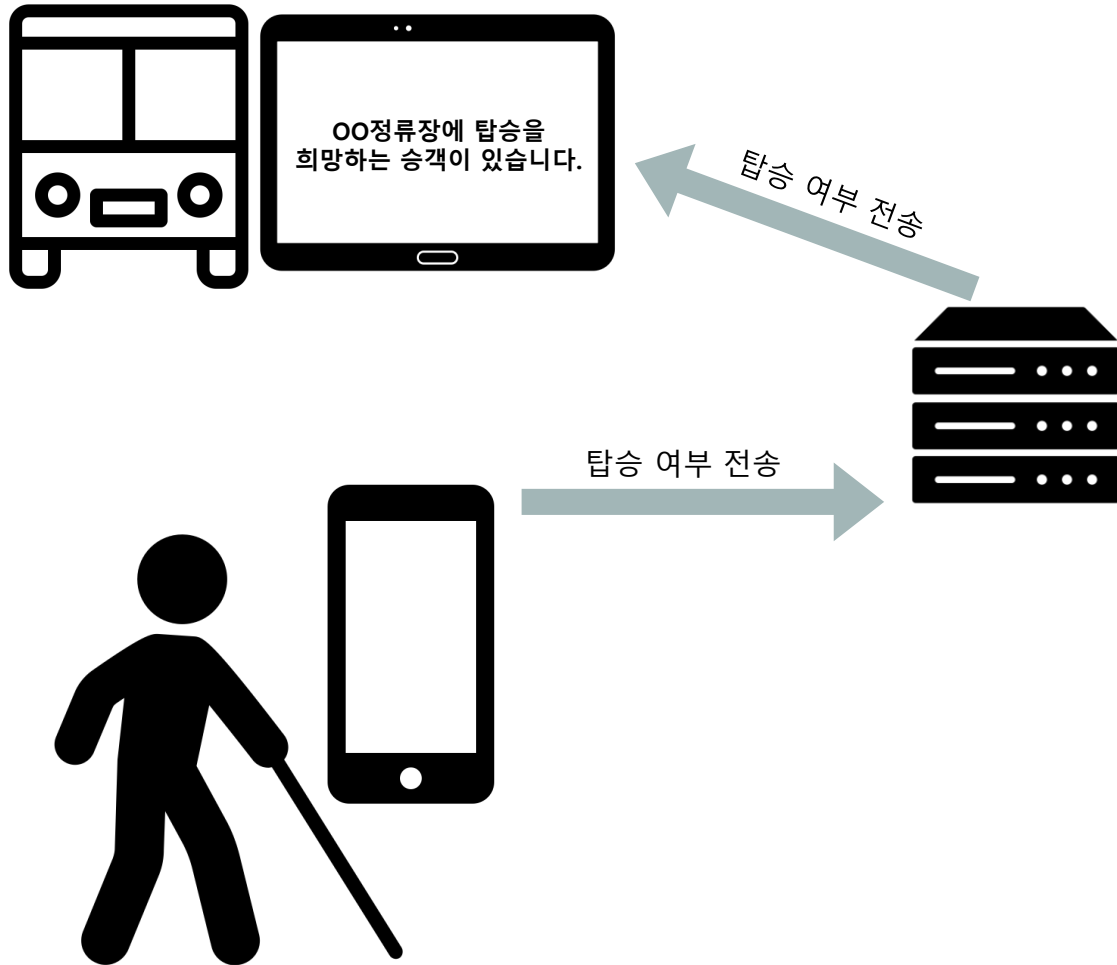
- 01 사용자가 버스 정류장에 접근하면 정류장 GPS와 사용자의 스마트폰 GPS 정보를 비교해 가장 가까운 정류장을 식별
- 02 가장 가까운 정류장으로 식별된 정류장의 위치 정보를 사용자 앱에 전송
- 03 사용자 앱에 서버로부터 받은 정류장 위치의 도착 버스를 좌우로 슬라이드하면 음성으로 안내하고 탑승할 버스를 선택

### 3.1 시스템 수행 시나리오



- 01 사용자가 버스 정류장에 접근하면 정류장 GPS와 사용자의 스마트폰 GPS 정보를 비교해 가장 가까운 정류장을 식별
- 02 가장 가까운 정류장으로 식별된 정류장의 위치 정보를 사용자 앱에 전송
- 03 사용자 앱에 서버로부터 받은 정류장 위치의 도착 버스를 좌우로 슬라이드하면 음성으로 안내하고 탑승할 버스를 선택
- 04 탑승할 버스를 선택하여 탑승여부를 버스기사님의 스마트폰 앱으로 전송한다.

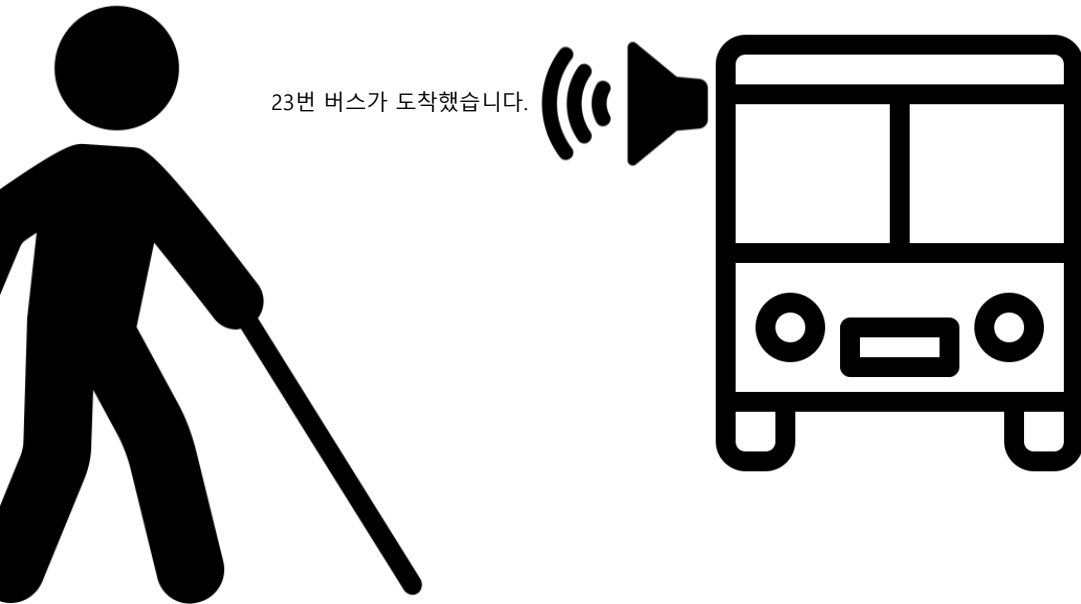
### 3.1 시스템 수행 시나리오



- 01 사용자가 버스 정류장에 접근하면 정류장 GPS와 사용자의 스마트폰 GPS 정보를 비교해 가장 가까운 정류장을 식별
- 02 가장 가까운 정류장으로 식별된 정류장의 위치 정보를 사용자 앱에 전송
- 03 사용자 앱에 서버로부터 받은 정류장 위치의 도착 버스를 좌우로 슬라이드하면 음성으로 안내하고 탑승할 버스를 선택
- 04 탑승할 버스를 선택하여 탑승여부를 버스기사님의 스마트폰 앱으로 전송한다.
- 05 버스기사님의 스마트폰 앱을 통해 해당 정류장에 탑승을 희망하는 시각장애인의 존재를 인식시킨다.



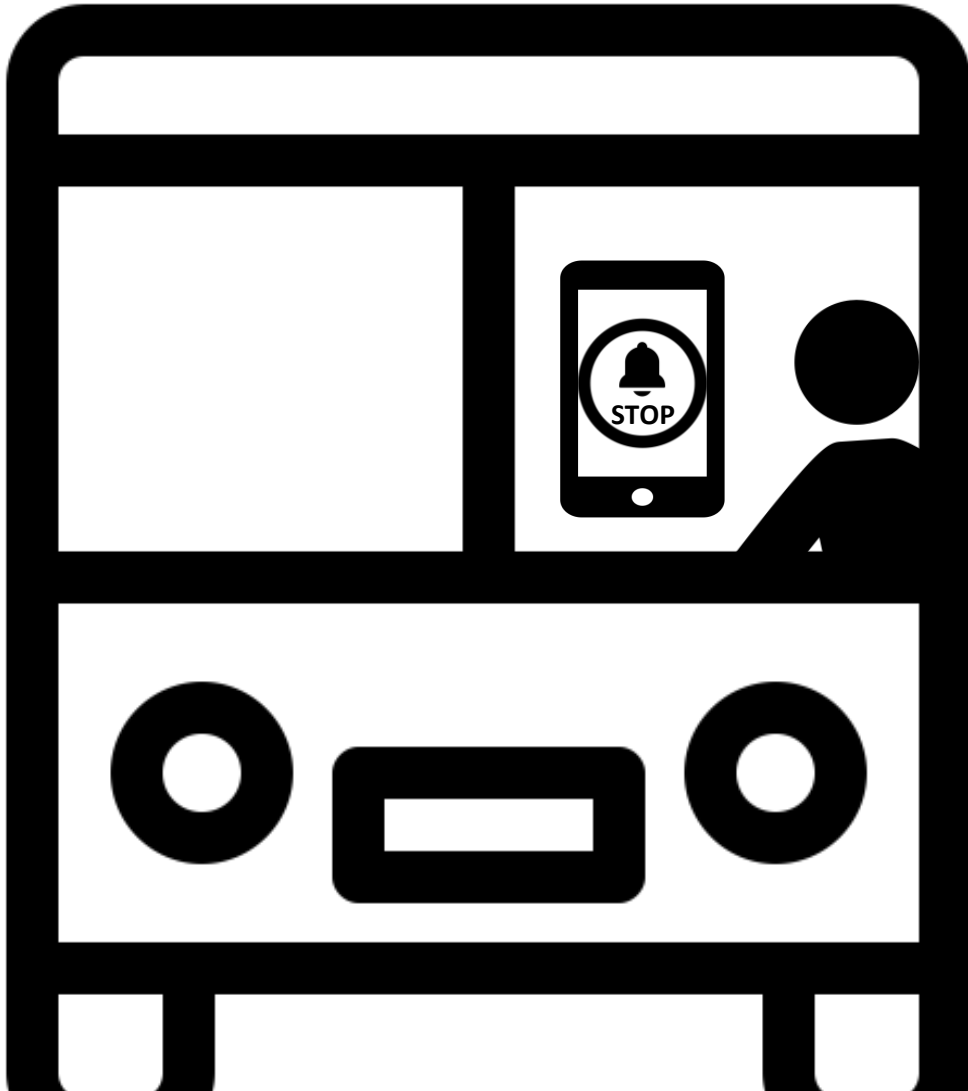
## 3.1 시스템 수행 시나리오



23번 버스가 도착했습니다.

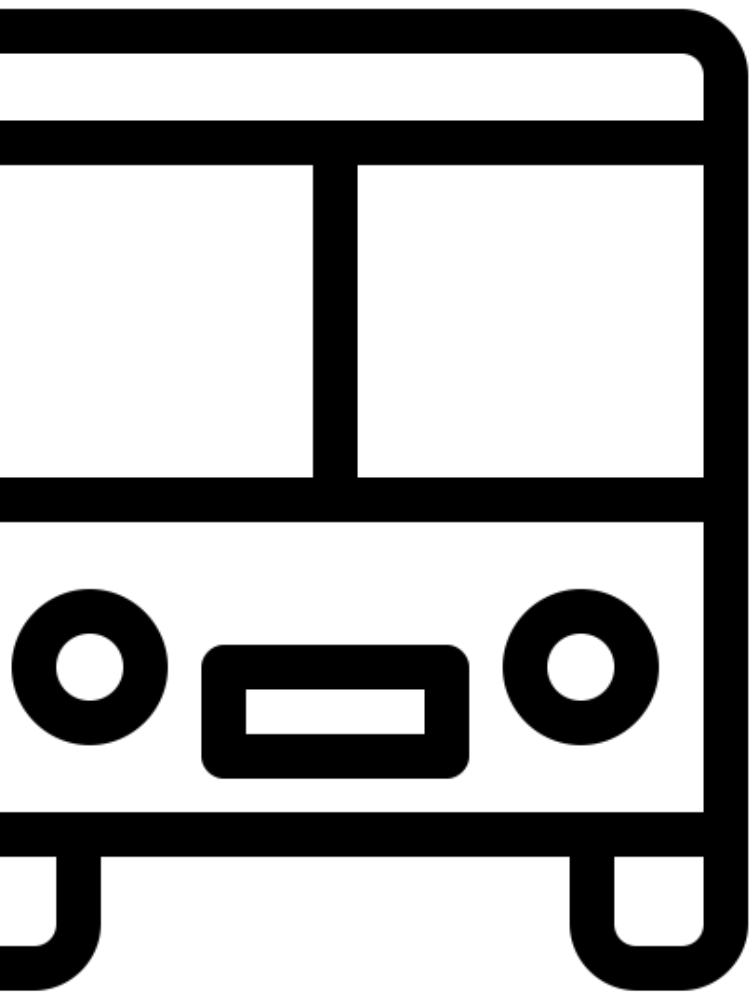
- 01 사용자가 버스 정류장에 접근하면 정류장 GPS와 사용자의 스마트폰 GPS 정보를 비교해 가장 가까운 정류장을 식별
- 02 가장 가까운 정류장으로 식별된 정류장의 위치 정보를 사용자 앱에 전송
- 03 사용자 앱에 서버로부터 받은 정류장 위치의 도착 버스를 좌우로 슬라이드하면 음성으로 안내하고 탑승할 버스를 선택
- 04 탑승할 버스를 선택하여 탑승여부를 버스기사님의 스마트폰 앱으로 전송한다.
- 05 버스기사님의 스마트폰 앱을 통해 해당 정류장에 탑승을 희망하는 시각장애인의 존재를 인식시킨다.
- 06 버스가 정류장에 도착하면 시각장애인이 탑승 희망한 버스 외부에 설치된 스피커를 통해 버스 도착을 알린다.

### 3.1 시스템 수행 시나리오



- 01 사용자가 버스 정류장에 접근하면 정류장 GPS와 사용자의 스마트폰 GPS 정보를 비교해 가장 가까운 정류장을 식별
- 02 가장 가까운 정류장으로 식별된 정류장의 위치 정보를 사용자 앱에 전송
- 03 사용자 앱에 서버로부터 받은 정류장 위치의 도착 버스를 좌우로 슬라이드하면 음성으로 안내하고 탑승할 버스를 선택
- 04 탑승할 버스를 선택하여 탑승여부를 버스기사님의 스마트폰 앱으로 전송한다.
- 05 버스기사님의 스마트폰 앱을 통해 해당 정류장에 탑승을 희망하는 시각장애인의 존재를 인식시킨다.
- 06 버스가 정류장에 도착하면 시각장애인이 탑승 희망한 버스 외부에 설치된 스피커를 통해 버스 도착을 알린다.
- 07 사용자가 버스에 탑승하고 하차할 때 기존 버스에 부착된 버스 하차벨과 연동된 비콘을 이용하여 사용자 APP으로 하차벨 동작

## 3.1 시스템 수행 시나리오



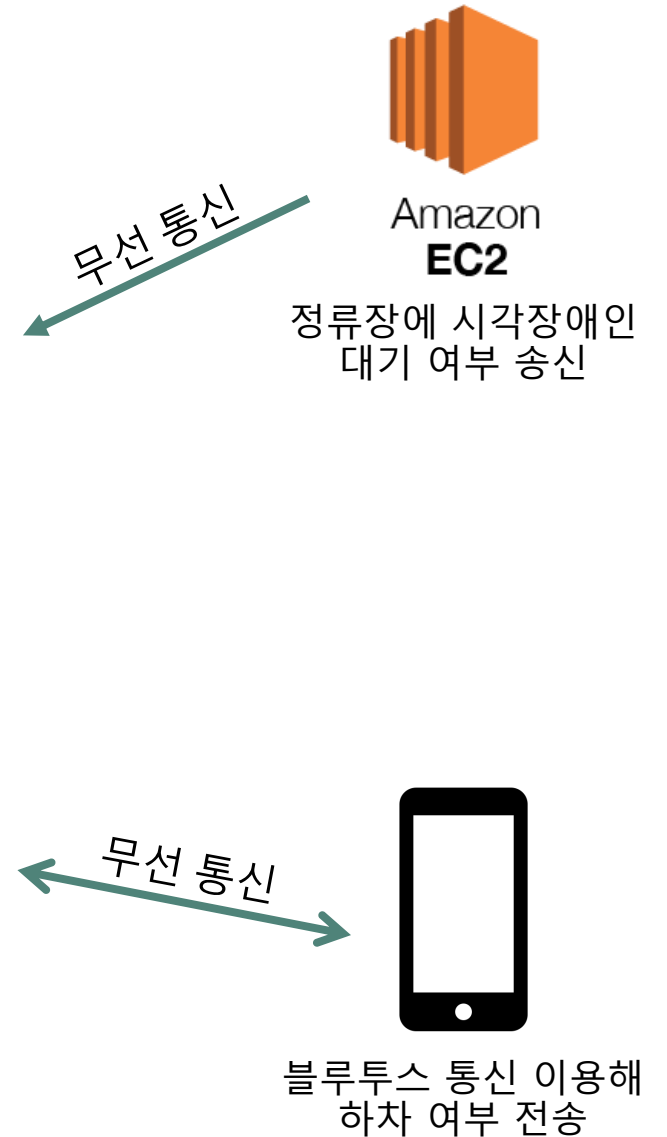
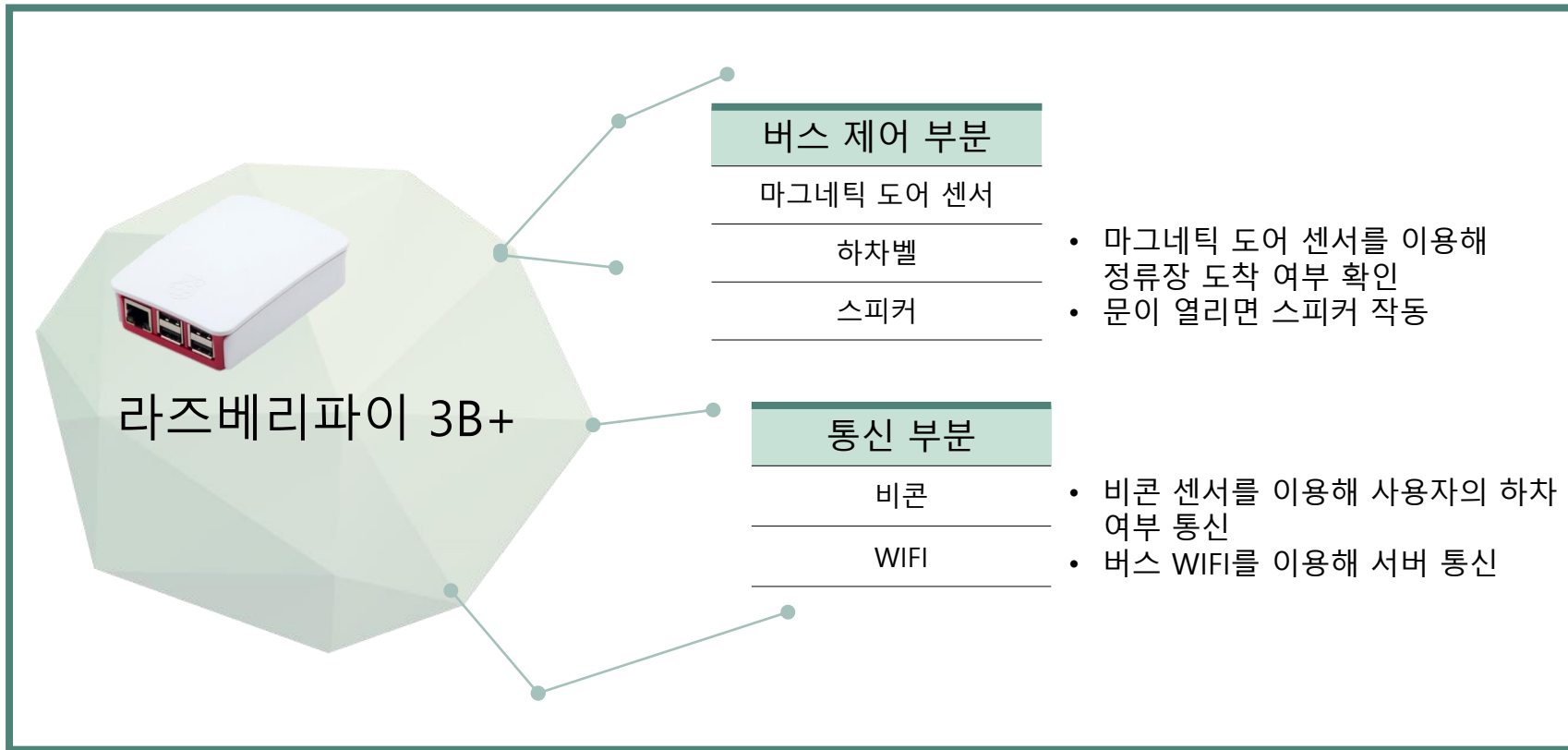
- 01 사용자가 버스 정류장에 접근하면 정류장 GPS와 사용자의 스마트폰 GPS 정보를 비교해 가장 가까운 정류장을 식별
- 02 가장 가까운 정류장으로 식별된 정류장의 위치 정보를 사용자 앱에 전송
- 03 사용자 앱에 서버로부터 받은 정류장 위치의 도착 버스를 좌우로 슬라이드하면 음성으로 안내하고 탑승할 버스를 선택
- 04 탑승할 버스를 선택하여 탑승여부를 버스기사님의 스마트폰 앱으로 전송한다.
- 05 버스기사님의 스마트폰 앱을 통해 해당 정류장에 탑승을 희망하는 시각장애인의 존재를 인식시킨다.
- 06 버스가 정류장에 도착하면 시각장애인이 탑승 희망한 버스 외부에 설치된 스피커를 통해 버스 도착을 알린다.
- 07 사용자가 버스에 탑승하고 하차할 때 기존 버스에 부착된 버스 하차벨과 연동된 비콘을 이용하여 사용자 APP으로 하차벨 동작
- 08 사용자가 버스에서 내린다.

# 4

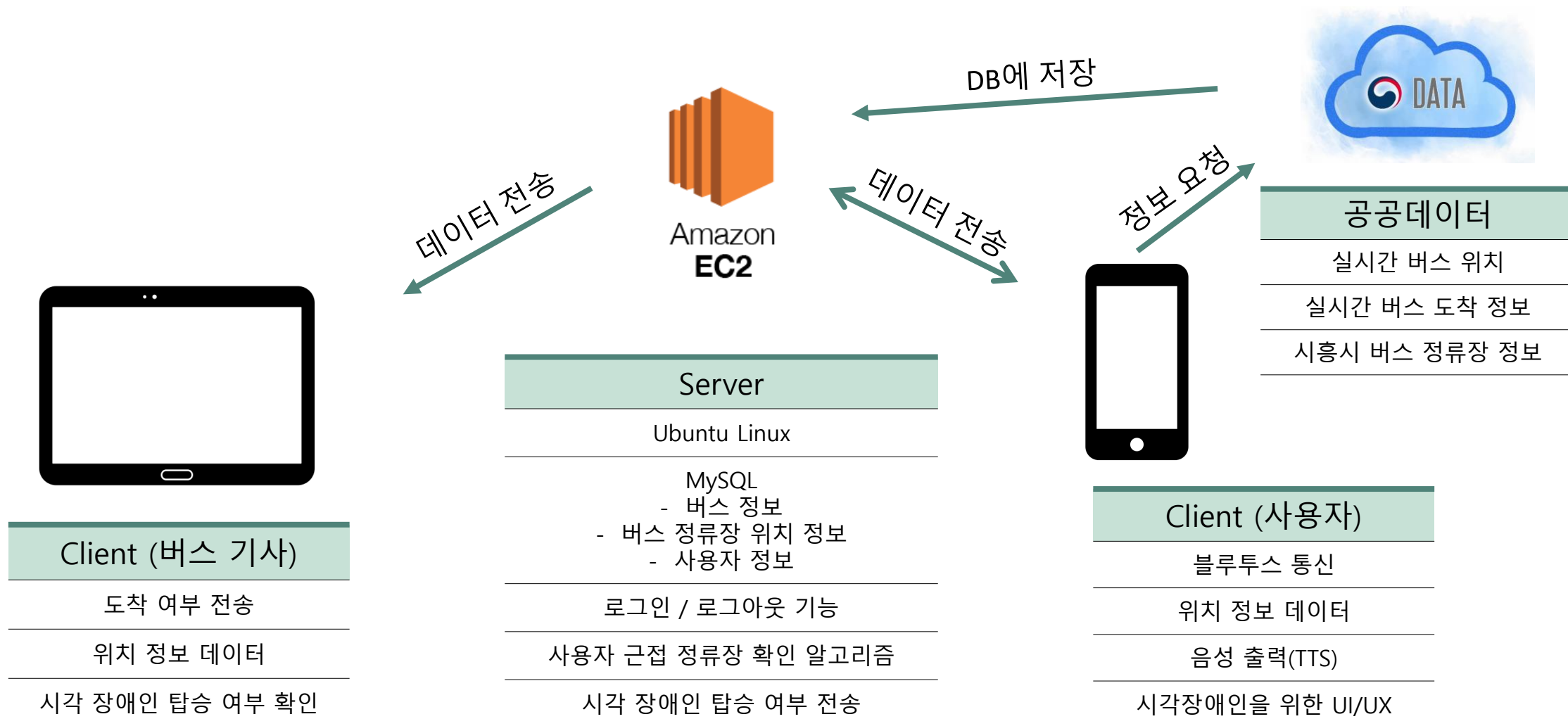
## 시스템 구성도

- 4.1 하드웨어 시스템 구성도
- 4.2 소프트웨어 시스템 구성도

## 4.1 하드웨어 시스템 구성도



## 4.2 소프트웨어 시스템 구성도



# 5

## 시스템 환경 및 개발 방법

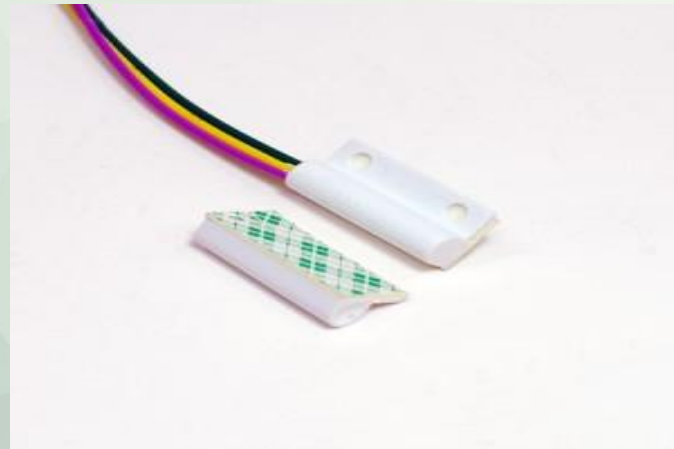
- 5.1 하드웨어 시스템 개발 환경
- 5.2 소프트웨어 시스템 개발 환경
- 5.3 시스템 개발 방법

## 5.1 하드웨어 시스템 개발 환경



라즈베리파이 3B+

- CPU : 1.4GHz Cortex-A53
- 1GB LPDDR2 SDRAM
- Bluetooth 4.2
- 무선 LAN
- 5V/2.5A DC 전원 입력



마그네틱 도어 스위치(MC-38)

- 입력 전압 : DC ~ 12V
- 작동거리 : 약 20mm
- 크기 : 27\*14\*10mm
- 소재 : 플라스틱

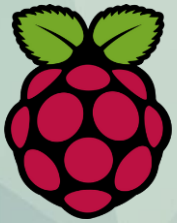


외부 차량용 스피커

- 출력 전원 : 3w
- 크기 : 80\*57\*45mm
- 임피던스 : 80hm



## 5.2 소프트웨어 시스템 개발 환경



### Rasbian

- 개발사 : 라즈베리파이 재단
- 사용 언어 : C/C++
- 버전 : 1.4



### VSCode

- 개발사 : Microsoft
- 사용 언어 : Flutter - Dart
- Flutter 버전 : 1.17



### IntelliJ

- 개발사 : Anaconda
- 사용 언어 : Java
- 버전 : 20.2



### AWS EC2

- 개발사 : Amazo
- T2.micro
- 최대 30GB
- 램 1GB
- Ubuntu Linux 20.04



### MySQL

- 개발사 : Oracle
- 사용 언어 : SQL
- 버전 : 8

## 5.2 소프트웨어 시스템 개발 환경



네이버 Clova TTS

- 개발사 : 네이버
- 입력 : 텍스트
- 요금 : 1000자당 4원



공공 API

- 개발사 : 경기도
- 출력 : XML, JSON



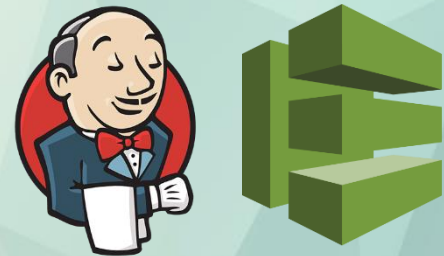
Github

- 개발사 : Microsoft
- 사용 언어 : Git
- 목적 : 버전관리, Issu 관리



Zenhub

- Kanban 이용
- 목적 : 프로세스 관리



Jenkins / Codedeploy

- Jenkins
- AWS CodeDeploy
- 목적 : 배포 자동화

## 5.3 시스템 개발 방법



GitHub 주소

<https://github.com/gowoonsori/weTaYo>



강석원

<https://github.com/ksssssw>

박형근

<https://github.com/ksssssw>

이지수

<https://github.com/leejisu9068>

홍의성

<https://github.com/gowoonsori>

README.md



함께 타요 ( We TaYo )

KPU 캡스톤 프로젝트로 시각 장애인을 위한 gps와 비콘을 이용 승하차 보조 시스템입니다.

팀원

- 강석원
- 박형근
- 이지수
- 홍의성

## 5.3 시스템 개발 방법

### 하드웨어 (라즈베리파이 3B+)

라즈베리파의 BLE기능을 활성화

스피커는 외부전력을 연결해 버스 외부에서 버스 정보를 출력

마그네틱 도어센서 사용해 문 열림을 감지, 감지에 따른 스피커 동작

TeamViewer를 사용해 원격으로 접속하여 개발할 수 있는 환경 구축

센서 동작 응용프로그램은 c를 사용해 작성

## 5.3 시스템 개발 방법

### 서버 (AWS ec2)

AWS ec2를 사용하여 서버 구축

AWS의 ec2의 개발 OS는 우분투/리눅스

Java Spring boot를 사용해 서버 개발

DB는 mysql8을 사용

공공데이터의 정적인 데이터(정류소 위치, 버스 노선 정보)는 DB에 저장

공공데이터의 동적인 데이터(실시간 버스 위치)는 실시간 공공 API를 이용

GPS 보정 알고리즘 (칼만 필터를 이용한 다중 보정 방법) 이용



## 5.3 시스템 개발 방법

### 애플리케이션 (Flutter)

Flutter을 이용하여 iOS와 Android 하이브리드 애플리케이션 개발

네이버 Clova TTS 음성 합성 기술 API를 사용한 음성 송출

사용자 앱은 시각장애인의 편의가 적용된 UI/UX를 반영

무선 하차벨을 동작시키는 과정에서 Beacon을 사용

Flutter은 Dart 언어를 사용하고 개발 환경은 Visual Studio Code를 사용

### DevOps

Zenhub를 이용하여 효율적으로 프로세스를 관리

애자일 방식을 사용해 Kanban 이용

Github로 Version과 Issue를 관리

# 60

## 업무 분담 및 수행 일정

- 6.1 업무 분담
- 6.2 수행 일정

## 6.1 업무 분담

	강석원	박형근	이지수	홍의성
자료수집	<ul style="list-style-type: none"> <li>- Flutter 조사</li> <li>- 네이버 TTS Clova 조사</li> <li>- GPS 오차 범위 조사</li> <li>- 유사 사례 조사</li> </ul>	<ul style="list-style-type: none"> <li>- 유사 사례 조사</li> <li>- 개발 방법 조사 및 목표 설정</li> <li>- 하드웨어 및 모듈 조사</li> <li>- Flutter 비콘 라이브러리 조사</li> </ul>	<ul style="list-style-type: none"> <li>- Flutter 조사</li> <li>- 네이버 TTS Clova 조사</li> <li>- Dart 언어 조사</li> <li>- 유사 사례 조사</li> </ul>	<ul style="list-style-type: none"> <li>- AWS 자료 조사</li> <li>- DB 조사</li> <li>- 프로세스 관리 툴 조사</li> <li>- 공공 데이터 조사</li> <li>- GPS 오차 범위 조사</li> </ul>
설계	IOS 앱과 서버 통신 설계	라즈베리파이 비콘 설정과 하차벨 회로 설계	Android 앱과 서버 통신 설계	AWS 서버 설계
구현	IOS 앱 구현	라즈베리파이 센서 연동 및 동작 구현	Android 앱 구현	AWS 서버 구축 및 통신 구현
테스트	Github로 수시로 Version 관리와 Issue관리를 함			



## 6.2 수행 일정

	11월	12월	1월	2월	3월	4월	5월	6~9월
현장 조사 및 요구사항 수집								
개발환경 조사 및 학습								
개발 환경 구축								
필요 알고리즘 개발								
데모 및 유지보수								
최종 거토 및 발표								

# 7

## 참고 문헌

### 7.1 참고 문헌

## 7.1 참고 문헌

Flutter 공식 문서 – <https://flutter.dev/docs>

CLOVA Speech Synthesis(CSS)

- [https://apidocs.ncloud.com/ko/ai-naver/clova\\_speech\\_synthesis/](https://apidocs.ncloud.com/ko/ai-naver/clova_speech_synthesis/)

칼만 필터를 이용한 GPS/INS 융합의 다중 보정 방법 – 권영민

- GPS/INS Fusion Using Multiple Compensation Method Based on Kalman Filter

Amazon Elastic Compute Cloud – Linux 인스턴스용 사용 설명서

- AWS

GB302 버스정보 Open API 서비스 명세서 – 버스 노선 조회 서비스(SOAP)

- 경기도 버스정보 시스템

GB307 버스정보 Open API 서비스 명세서 – 버스 위치 정보 조회 서비스(SOAP)

- 경기도 버스정보 시스템

GB308 버스정보 Open API 서비스 명세서 – 버스 도착 정보 조회(SOAP)

- 경기도 버스정보 시스템

QnA



THANKS