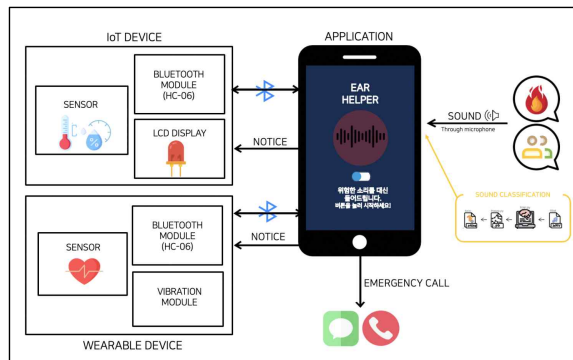


산학협력 프로젝트 결과보고서

과제명	청각장애인을 위한 소리 감지 IoT 웨어러블 디바이스		
협력기관명	(주)신라시스템	과제멘토	김민희 (부사장)
참여인원	(총 6명) 기업체 2명, 학부과정 4명(신현지, 엄다연, 이수경, 임지혁)		
수행기간	2022.03.07.~06.30.(4개월)	유형	중기
추진배경	<ul style="list-style-type: none"> ○ 신규 등록 장애인의 가장 많은 비중을 나타낸 장애 유형이 청각장애인으로 집계됨. ○ 청각장애인 재난 대응 욕구에 관한 연구에 따르면 재난이 발생했음을 알려주는 장치가 설치되어 있느냐는 질문에 66.3%가 부정 응답을 보임. ○ 따라서, 청각장애인들의 재난 대응에 도움을 줄 수 있는 소리 감지 IoT 웨어러블 디바이스 및 애플리케이션을 개발하고자 하여 전반적인 삶의 질 향상에 기여하고자 함. 		

목표 및 내용

- **위급 상황 시 알림 기능 개발**
 - 경적 소리, 화재 경보음, 사이렌 소리 등의 위험한 소리를 딥러닝을 통해 감지하는 기능 개발
 - 인식한 결과를 애플리케이션에서 알림을 주고 IoT 디바이스의 LCD에서 출력하는 기능을 개발
- **특정 단어 인식 기능 개발**
 - 일상생활 속에서 자주 접할 수 있는 단어들을 등록할 수 있는 기능을 개발.
 - 사용자가 등록한 단어들을 인식할 수 있도록 하는 기능을 개발
 - 인식한 결과를 애플리케이션에서 알림을 주고 IoT 디바이스의 LCD에서 출력하는 기능을 개발.
- **IoT 디바이스 제작 및 개발**
 - 위험 소리 감지, 특정 단어 인식 후 알림을 받기 위한 디바이스 제작
 - 온습도 센서, 심박수 센서 등을 활용하여 일상생활 속에서 도움이 되는 정보 기능 구현
- **애플리케이션과 디바이스 통신 구현**
 - 애플리케이션과 IoT 디바이스 간의 양방향 블루투스 통신을 위한 기능 구현



수행결과

- **애플리케이션 개발**
 - 단어 인식
WordList에 단어를 등록하고 Speech-to-text api를 사용해 단어를 인식할 수 있도록 함.
 - Watch-out
Tensorflow lite Yamnet 모델을 기반하여 Teachable Machine으로 음성 분류모델을 학습시키고 추출한 TFLite 모델을 어플리케이션에 적용시켜 작동하도록 함.
 - 블루투스
Foreground Service에서 Android Bluetooth API를 통해 IoT 디바이스와 어플리케이션 간의 블루투스 연결을 구현해 백그라운드에서 블루투스 연결이 가능하도록 구현함.
- **IoT 디바이스 개발**
 - 온습도 감지 기능 개발
 - 심박수&진동 모듈 기능 개발
- **논문투고 및 학술대회 참여**

1. 과제 수행 배경

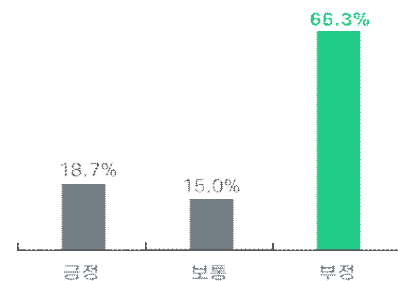
우리는 시각으로 한번에 약 120도 가량의 세상을 볼 수 있다. 하지만, 시야에 들어오지 않은 위험이나 정보를 수집하기 위해서는 청각을 이용한다. 예를 들면, 길을 건너려다가도 자동차 경적소리가 나면 주변을 둘러보게 되고, 화재경보가 울리면 눈앞에 불길이 없어도 대피할 수 있다. 이렇듯 청각은 위험한 상황을 인지하는데 필수적인 감각이다.

보건복지부에 따른 2021년도 신규 등록 장애인 현황 통계자료로 청각장애인(32.8%)이 두 번째로 큰 비중을 나타낸다. 청각장애인은 안내방송을 듣지 못하거나 사이렌 소리를 듣지 못해 재난 상황에 대한 대처가 어려운 상황이고, 자동차 경적 등 청각이 미치지 못하는 일상생활 속 위험 상황에 대한 많은 어려움에 노출되어 있지만 관련된 서비스가 굉장히 부족한 상황이다.



▶ 안내방송에 미처 대처하지 못한 사례

재난이 발생했음을 알려주는 장치가 설치되어 있습니까?
(시각경보기, 신호알람기, 소리전달기 등)



출처 - 청각장애인 재난대응 욕구에 관한 연구

▶ 청각장애인 재난대응 욕구에 관한 연구

<청각장애인들을 위한 소리 감지 프로젝트의 필요성>

이러한 문제를 해결하기 위해 애플리케이션에서 긴급 사이렌 소리 등을 인지하여 진동 및 알람을 주어 사용자가 주변의 위험하고 급박한 상황을 빠르게 인지하고 대피할 수 있도록 한다. 또한 다른 프로젝트와 차별성을 주기 위해 청각장애인들이 실제 필요로 하는 서비스가 무엇인지 고민하였고, 그러한 서비스로 단어 인식 서비스를 프로젝트에 추가하고자 하였다. 단어 인식 서비스는 사용자가 애플리케이션에서 저장한 특정 단어를 애플리케이션이 대신 들어주어 알람을 주는 역할을 하는 서비스로 청각장애인들의 사회적 고립을 해결하고 삶의 질 향상을 도모한다. 이는 청각장애인 뿐만 아니라 평소 이어폰 등을 자주 사용하는 사람들에게도 항상 위험에 노출되어 있다는 점에서 소리 감지 기능이 유용하게 쓰여 여러 안전사고를 예방할 수 있다는 점에서 큰 의미를 지닌다.

즉, 본 프로젝트에서는 청각장애인 개인 주변에서 발생한 이벤트에 대한 인지/감지 능력의 저하로 인해 안전을 위협하고 사회적 고립을 심화시키는 현상을 개선하고자 한다. 청각장애인의 실생활에서 일어나는 '제2의 장애 요소'를 개선함으로써, 보다 안전하고 편리한 생활환경을 제공하여 청각장애인의 삶의 질 향상에 기여한다. 더하여, 집에서도 사용하기 편리하도록 애플리케이션과 블루투스로 연결하여 사용할 수 있는 IoT를 함께 개발함으로써 편리성 및 차별성을 더한다. 또한, 여러가지 부가 기능을 추가하여 청각장애인들의 전반적인 삶의 질 향상에 기여하고자 한다.

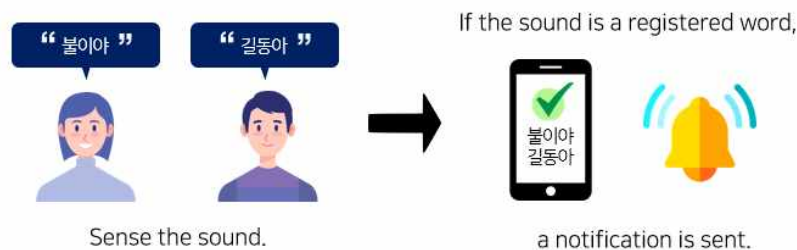
2. 과제 목표

위와 같은 과제의 필요성을 바탕으로, IoT 디바이스를 결합시킨 소리 감지 어플리케이션을 개발하고자 한다. 해당 어플리케이션은 일상생활에서 소리에 불편함을 느끼는 청각장애인에게 도움을 주는 것을 목적으로 한다. 핵심기능으로는 크게 특정 단어 인식 및 위험 소리를 인식하여 알람을 주고 위험 상황 발생 시 긴급 메시지 또는 전화를 연결할 수 있도록 한다. 또한, IoT 디바이스 기기를 통해서도 진동 및 화면 알림을 받음으로써 휴대폰을 확인하지 못할 상황까지 대처하여 차별성을 더한다. 뿐만 아니라 하드웨어를 통해 각종 온습도, 심박수 등을 측정하여 소리 이외에도 일상생활에 도움이 될 수 있는 여러 가지 기능을 포함한 통합적인 시스템을 개발하고자 한다.

1. Recognize specific words

사용자는 어플리케이션을 통해 원하는 특정 단어를 등록할 수 있다. 등록된 특정 단어를 말하는 소리가 인식되면 어플리케이션 및 IoT 디바이스 기기에 알람을 주도록 한다.

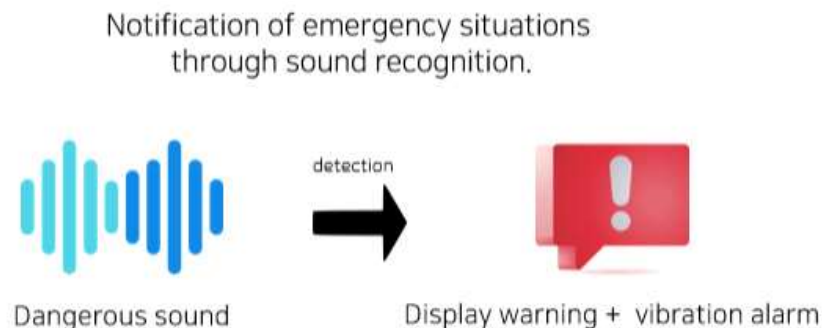
- 1) Speech recognizer와 구축된 데이터베이스를 통해 사용자가 등록한 특정 단어를 인식한다.
- 2) IoT 디바이스 기기 LCD 및 진동 모듈을 통해 진동과 화면을 통한 알림을 줄 수 있도록 한다.
- 3) 어플리케이션 푸시 알람을 통해 특정 단어가 인식되었다는 알림을 준다.
- 4) 블루투스 통신 IoT 디바이스 LCD에도 알림을 줌으로써 어플리케이션의 활용성을 높인다.



2. Sense the danger

경적 소리, 화재 경보음과 같은 위험 상황을 알리는 소리가 들리면, 이를 감지하여 알림을 주고 도움을 요청할 수 있도록 해준다.

- 1) 자동차 경적소리, 화재 경보음과 같은 위험 소리를 인지하는 딥러닝 학습을 진행한다.
- 2) 학습된 모델에 의해 소리가 인식될 경우 어플리케이션을 통해 사용자에게 알림을 주도록 한다.
- 3) 또한, 위험 소리 인지 시 알림과 함께 긴급 메시지 또는 전화로 연결될 수 있도록 해준다.
- 4) 블루투스 통신 IoT 디바이스 LCD에도 알림을 줌으로써 어플리케이션의 활용성을 높인다.

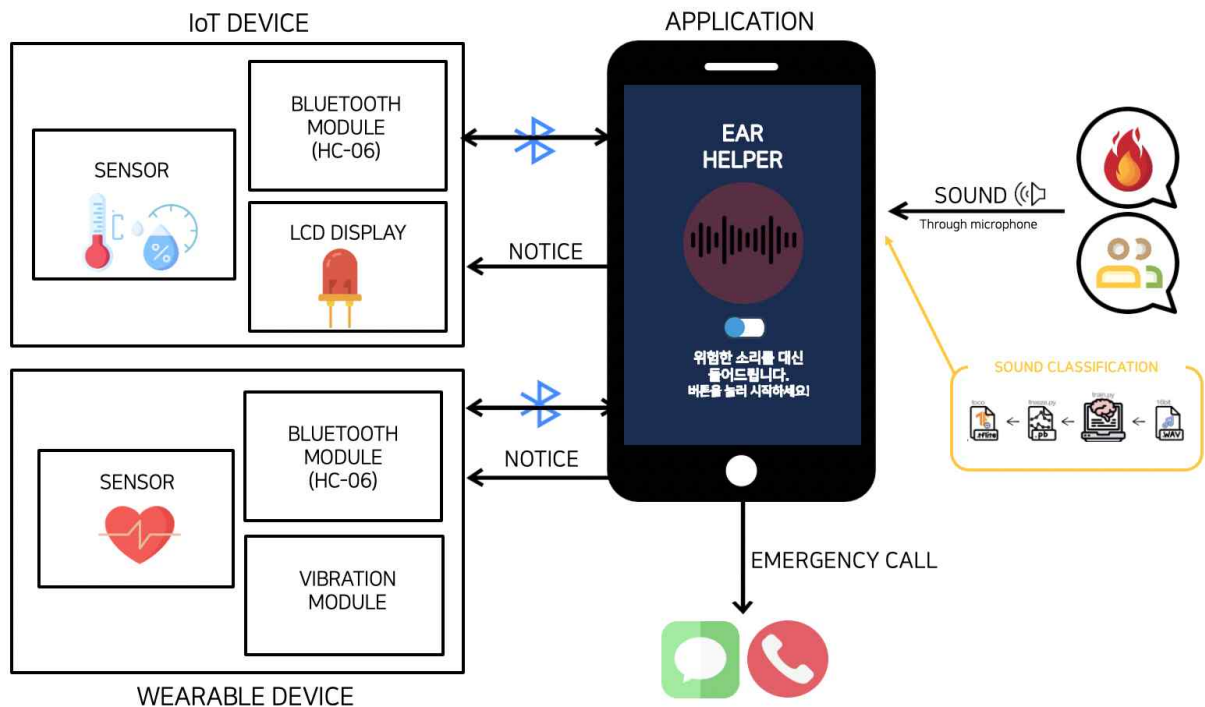


3. Add functions using various sensor

그 이외에 IoT 디바이스 기기에 다양한 센서를 이용하여 추가적인 기능을 줄 수 있도록 한다. 온습도를 통해 오늘의 날씨, 심박수 측정을 통해 건강 상태 확인 등을 할 수 있도록 개발한다.

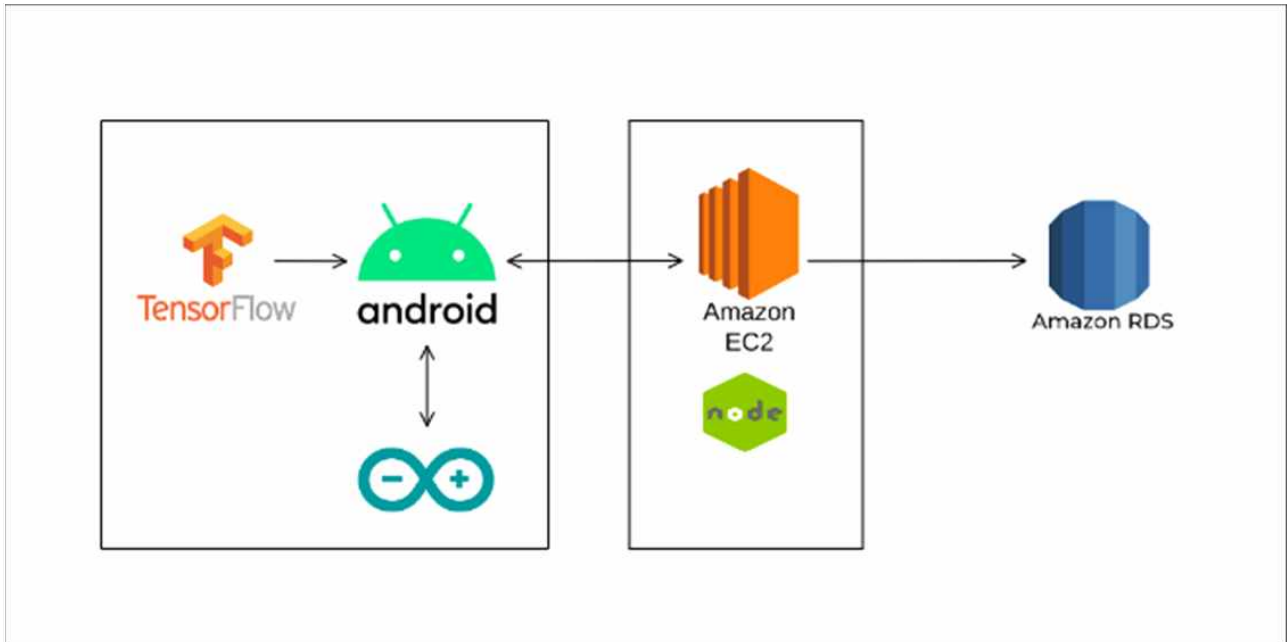


<System Scenario>



전체적인 시스템 시나리오로, 애플리케이션과 하드웨어 부분으로 나뉜다. 하드웨어는 심박수 측정 및 진동 알람을 위한 웨어러블 디바이스와 온습도 측정 및 LCD 알람을 위한 IoT 디바이스로 구성된다. 하드웨어와 애플리케이션은 블루투스 통신을 하며, 해당 디바이스는 여러가지 센서값을 인식하여 앱에 전달을 한다. 온습도, 심박수 등을 측정함으로써 소리 이외에도 많은 정보들을 앱에서 확인할 수 있도록 한다. 또한, 딥러닝을 통해 학습된 모델을 바탕으로 sound classification을 진행하여 인식한 위험한 소리가 감지되면 애플리케이션에 알림을 주고 긴급 전화를 연결할 수 있도록한다. 더하여, 애플리케이션을 확인하지 못할 경우에도 위급 상황에 대한 대처를 높이기 위하여 블루투스 통신을 통해 IoT 디바이스 기기에도 해당 정보를 전달하여 알림을 주도록 시스템 시나리오를 구성하였다.

<System architecture>



IoT 디바이스는 아두이노와 여러 센서로 제작되며 블루투스 모듈을 사용해 어플리케이션과 양방향 통신을 한다. Tensorflow를 이용해 sound classification에 필요한 학습을 진행하며 결과로 나온 인공지능 모델을 바탕으로 어플리케이션은 특정 소리를 인지하여 사용자에게 알람을 보낼 수 있다. 또한, 서버를 통해 단어 인식을 위한 사용자의 단어 리스트를 저장하고 사용자를 관리한다.

<Development Tool>

1. 소프트웨어 (SW Development)



① 구글의 티처블 머신

Teachable Machine은 누구나 머신러닝 모델을 쉽고 빠르게 만들 수 있도록 제작된 웹 기반 도구이다. 라벨링된 샘플들을 티처블 머신으로 학습시켜 사운드를 분류할 수 있는 모델을 만들고, 이 모델을 내보내 특정 소리를 인식하여 알람을 주는 기능 구현을 더하고자 한다.

② Speech-to-Text API

구글 텍스트 음성 변환은 안드로이드 운영 체제용으로 구글이 개발한 스크린 리더 어플리케이션이다. 사운드를 듣고 text로 변환하여 특정 단어가 인식되었을 때 알람을 줄 수 있도록 구현하고자 한다.

③ 텐서플로우 라이트

ML 모델을 개발하고 학습시키는 데 도움이 되는 핵심 오픈소스 라이브러리로 유연하게 ML이 접목된 어플리케이션을 빌드할 수 있다. 경적소리, 화재경보음과 같은 사운드를 학습시켜 사용자에게 알림을 줄 수 있도록 구현한다.

→ **텐서플로우 라이트와 구글 STT API을 결합하여 sound classification end-to-end 프로젝트를 개발하고자 한다.**

④ 프론트엔드 - Android Studio

안드로이드 전용 어플리케이션 제작을 위한 공식 통합 개발환경으로 사용자의 편의성이 보장되는 UI 디자인을 설계한다.

⑤ 백엔드 - AWS & nodejs

사용자는 단어를 등록해 특정 단어에 대한 알람을 받을 수 있는데 서버를 통해 사용자가 등록한 단어를 관리한다. 또한, 관리자 서버로서 사용자를 관리할 수 있도록 한다.

2. 하드웨어 (HW development)

1) Arduino

아두이노(Arduino)는 대중적으로 가장 많이 사용되고 있는 오픈 소스 하드웨어로 다양한 웨어러블 컴퓨터 프로토타입 개발 및 차세대 디지털 기기 발명에 활용되고 있다. 다수의 스위치나 센서로부터 값을 받아들이며, 외부 전자 장치들을 통제함으로써 환경과 상호작용이 가능하기에 위 프로젝트에서 IoT 디바이스 기기를 만들기 위한 핵심 장비로 사용할 예정이다.

2) Bluetooth module

아두이노는 유선통신인 시리얼 통신을 사용한다. 따라서, 무선통신을 하기 위해서는 무선통신을 시리얼 통신으로 바꿔주는 모듈이 존재해야 한다. 블루투스 모듈은 다른 wifi, xbee 모듈보다 저렴하고, 범용성이 높아 해당 모듈을 사용하고자 한다. 근거리 통신이라는 제약이 존재한다는 점이 있지만, 집에서 사용하는 IoT 기기 및 손목에 부착하는 웨어러블 기기라는 점에서 블루투스 통신을 선정하였다. 따라서 아두이노 보드에 해당 모듈을 적절한 핀에 연결함으로써 IoT 디바이스 기기에서 인식된 각종 센서값들과 어플리케이션의 각종 알림 등을 블루투스를 통한 양방향 통신을 하도록 한다.

3) Vibration module

웨어러블 디바이스에 연결하여 사용된다. 이는 주어진 신호값에 따라 진동을 하는 모터 부품으로, 특정 알림 발생 시 진동모듈이 작동할 수 있도록 한다. 진동모듈에 해당하는 V값에 따라 적절한 핀에 연결해야 한다는 점에 주의하여 웨어러블 디바이스를 만들고자 한다.

4) Heartbeat sensor

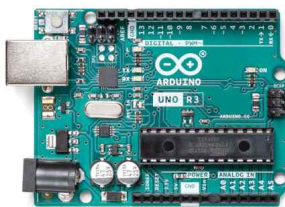
심박수 모듈 정 가운데에 LED가 있고, 그 바로 아래에 수광부가 존재한다. 혈류량이 높을 때 혈구의 밀도가 높아 빛이 많이 반사되고 혈류량이 낮을 때는 혈구의 밀도가 낮아서 빛이 많이 투과된다. 즉, 미세혈관에 적외선을 조사 후 빛의 반사율을 반복하여 측정함으로써 맥박을 감지하도록 한다. 해당 모듈을 웨어러블 디바이스 기기 아두이노에 적절한 핀으로 연결한다. 따라서 측정된 심박수를 블루투스 모듈을 통해 어플리케이션에 값을 전달할 수 있도록 한다.

5) Temperature and humidity sensor

정전식 습도센서(습도에 따라 저항값이 변하는 소자)와 서미스터(온도에 따라 저항값이 변하는 소자)를 사용하여 대기온도와 습도를 측정하고 디지털 신호로 출력한다. 두 전극 사이의 전기 저항 변화를 측정하여 수증기를 감지해 공기 중 습도 변화를 알아낸다. 온도 감지 모듈은 반도체 세라믹으로 이루어져 온도에 따라 물질의 저항 값이 변하는 소재의 특성을 이용하여 값의 변화를 감지해 온도를 출력한다. 따라서, 이를 IoT 디바이스 아두이노 및 브레드보드에 알맞은 저항값으로 알맞은 핀에 연결함으로써 어플리케이션에 전달해주어 소리 인식 뿐만 아니라 일상생활에 필요한 정보를 제공해준다.

6) LCD display

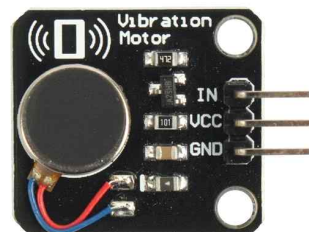
LCD는 액정 디스플레이로 액정의 성질을 가지는 전류를 사용하여 신호값을 시각적으로 확인할 수 있는 부품이다. 본 프로젝트에서는 위험한 소리가 인지되면 해당 소리의 정보와 주위를 둘러보라는 메시지를 출력하여 청각장애인들이 시각적으로도 확인하기 쉽게 하였다.



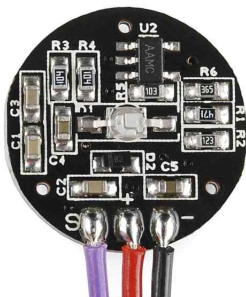
▶ Arduino



▶ Bluetooth module



▶ Vibration module



▶ Heartbeat sensor

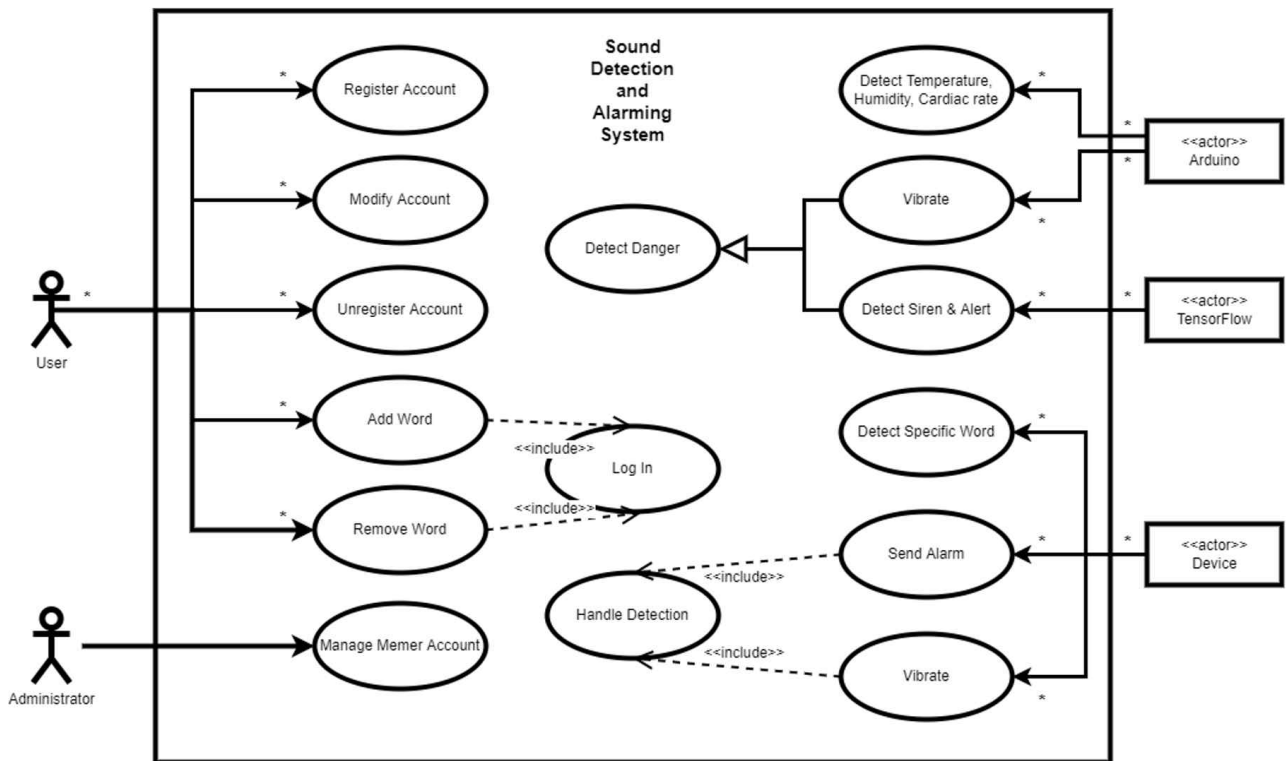


▶ Temperature & humidity sensor



▶ LCD Display

<Usecase Diagram>



이러한 시스템 시나리오, 아키텍처, 개발 툴, 다이어그램 등을 바탕으로 소프트웨어 및 하드웨어 개발을 진행하도록 한다. 따라서, 최종적으로 IoT 디바이스를 결합시킨 소리 감지 애플리케이션을 통해 청각 장애인 및 해당 기능을 필요로 하는 일반인들에게 유익한 기능을 제공하도록 한다.

3. 과제 수행 결과

[어플리케이션 부분]

1) 개발 필요성

청각장애인은 안내 방송을 듣지 못하거나 사이렌 소리, 자동차 경적 등 청각이 미치지 못하는 일상생활 속 위험 상황에 대한 대처가 어려운 상황이다. 그러한 어려움에 대처하기 위해 일상 속에서 가장 함께 지니고 다니는 스마트폰을 활용하여 어플리케이션을 개발하고자 한다. 따라서, 실행 화면 외에서도 계속해서 실시간으로 소리를 감지함으로써 위험한 소리 및 등록된 단어 소리가 감지되면 알람을 줄 수 있는 기능이 필요하다. 이에, 본 프로젝트에서는 실시간 소리 감지에 뛰어난 어플리케이션 Ear-Helper을 개발하고자 한다. 더하여 해당 어플리케이션이 IoT 디바이스와의 통신하도록 개발하여 편리성 및 차별성을 더하고자 한다.

2) 이슈사항 및 해결방안

가. 블루투스 연결 지속 문제

IoT 디바이스와 어플리케이션 간의 연결을 위해 블루투스를 구현하였다. 어플리케이션 프레임워크는 Android Bluetooth API를 통해 블루투스 기능에 대한 액세스 권한을 제공한다. 어플리케이션은 이러한 API를 통해 다른 블루투스 기기에 무선으로 연결하여 지점 간의 무선 기능을 활성화할 수 있다. 4가지 주요 작업(블루투스 설정, 로컬 영역에서 페어링이 되었거나 사용가능한 기기 찾기, 기기 연결, 기기 간 데이터 전송)을 거쳐 블루투스 통신을 구현하였고, IoT 디바이스로부터 센서 값들을 받아왔다.

i. 블루투스 설정

어플리케이션에서 블루투스 기능을 사용하기 위해 두 개의 권한을 선언하였다. 연결 요청, 연결 수락 및 데이터 전송과 같은 블루투스 통신을 수행하기 위한 BLUETOOTH 권한과 블루투스 스캔을 사용하여 사용자 위치에 대한 정보를 수집할 수 있는 ACCESS_FINE_LOCATION 권한이다. 그리고 앱에서 기기 검색을 시작하거나 블루투스 설정을 조작할 수 있도록 BLUETOOTH_ADMIN 권한도 선언해주었다.

블루투스를 사용하여 통신을 하기 위해 기기에서 블루투스가 지원이 되는지 확인하였고, 지원이 되는 경우 블루투스를 활성화시켜주었다. 정적 getDefaultAdapter() 메서드를 호출하여 BluetoothAdapter를 가져오면 어플리케이션이 해당 객체를 사용하여 상호작용할 수 있다. 어댑터가 null을 반환할 경우 해당 기기는 블루투스를 지원하지 않는 것이니 그에 대한 메시지를 띄워 사용자에게 알려주었다. isEnabled() 메서드를 호출해 현재 블루투스가 활성화되어있는지 확인하였고 false인 경우 startActivityForResult() 메서드로 인텐트를 전달해 블루투스 활성화를 요청하였다. 화면에 블루투스를 활성화하기 위해 사용자 권한을 요청하는 대화상자가 표시되며 사용자가 허락하면 시스템이 블루투스를 활성화하기 시작한다.

ii. 로컬 영역에서 페어링이 되었거나 사용가능한 기기 찾기

BluetoothAdapter를 통해 기기를 검색하거나 페어링된 기기 목록을 쿼리하여 블루투스 기기들을 찾을 수 있다. 기기 검색 전 페어링된 기기 집합을 쿼리하여 확인해주었고 getBondedDevices() 메서드를 호출하여 페어링된 기기를 나타내는 BluetoothDevice 객체 집합을 반환받았다. 다음으로 기기 검색을 하기 위해 startDiscovery() 메서드를 호출하였고 ACTION_FOUND 인텐트에 대한 BroadcastReceiver를 등록해 검색된 각 기기에 대한 정보를 수신할 수 있도록 했다. 블루투스 기기와 연결을 시작하려면 연결된 객체에서는 MAC주소만 있으면 되기 때문에 getAddress()를 호출하여 검색해주었다.

iii. 기기 연결

원격 기기와의 연결을 시작하려면 먼저 원격 기기를 나타내는 BluetoothDevice 객체를 얻어야 한다. 검색된 기기의 MAC주소를 통해 BluetoothDevice를 생성하기 위해 getRemoteDevice() 메서드를 호출하였

다. BluetoothDevice로 createRfcommSocketToServiceRecord(UUID)를 호출하여 BluetoothSocket을 가져왔고, connect()를 호출하여 연결을 해주었다.

iv. 기기 간 데이터 전송

연결된 BluetoothSocket을 사용해 본격적으로 기기 간에 정보를 공유해주었다. getInputStream() 및 getOutputStream()을 각각 사용하여 소켓을 통해 전송을 처리하는 InputStream 및 OutputStream을 가져와주었다. 그런 다음, 스트림에서 read()와 write()를 해주어 IoT 디바이스로부터 온습도 센서 값과 심박수 값을 받아왔고, watch-out에서 이벤트가 생겼을 때 그에 따른 상황 정보를 디바이스에게 전달해주고자 하였다.

위와 같은 과정으로 블루투스를 구현하였을 때 블루투스를 연결한 액티비티에서 통신을 하는 것은 가능했지만, 액티비티 전환이 이루어질 경우 블루투스 연결이 꺼지는 문제가 있었다. 해당 프로젝트에서는 다른 기능이 수행될 때도 블루투스 연결이 계속 되어야 하기 때문에 Foreground service를 이용하여 이와 같은 문제를 해결하고자 하였다.

=> 포그라운드 서비스에서의 블루투스 연결

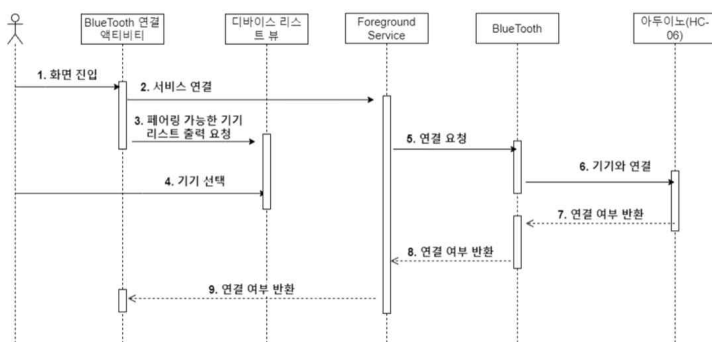


그림 1 블루투스 연결 시퀀스

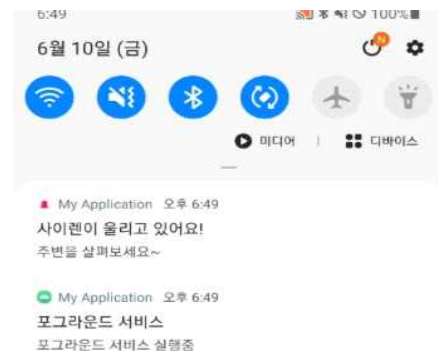


그림 2 포그라운드 서비스가 실행되고 있는 모습

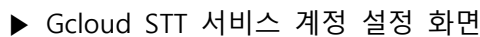
Service는 백그라운드에서 오래 실행되는 작업을 수행할 수 있는 애플리케이션의 구성 요소로, 그 중에서 포그라운드 서비스는 사용자에게 눈에 띄는 작업을 수행한다. 해당 프로젝트에서는 사용자가 블루투스의 연결을 끊기 전까지 블루투스가 계속 연결이 되고 있음을 사용자가 알 수 있도록 포그라운드 서비스를 선택해 구현하였다.

onBind() 메서드를 정의하여 클라이언트가 서비스와 통신을 주고받기 위해 사용할 인터페이스를 제공하고자 IBinder를 반환했다. 그렇게 되면, 다른 구성 요소가 해당 서비스에 바인딩되고자 하는 경우에 bindService()를 호출함으로써 서비스와 통신을 주고받는 것이 가능해진다. onCreate() 메서드는 서비스가 처음 생성되었을 때 일회성 설정 절차를 수행하는데 이 메서드에서 블루투스 연결과 관한 설정을 정의해주었다. onStartCommand() 메서드에는 블루투스로 연결된 기기로부터 데이터를 받아 처리하는 일을 정의해주어 다른 구성 요소에서 startService를 호출해 서비스가 시작되도록 요청할 경우 이 메서드가 실행되어 서비스가 시작되고 백그라운드에서 블루투스 통신을 실행할 수 있다. 사용자가 블루투스 연결을 끊고자 블루투스 off 버튼을 누르면 정의되어있던 onDestroy() 메서드가 호출되어 서비스가 소멸되도록 구현하였다.

ForegroundService에서 블루투스로 연결된 기기로부터 데이터를 받아 HomeFragment로 전달해줄 때 EventBus를 사용하였다. 이벤트 버스는 버스 네트워크의 동작 방식과 유사하게 하나의 이벤트를 동일 채널에 연결된 모든 노드(Subscriber)들에게 보내는 방식이다. 채널에 연결되어 있거나 하면 이벤트를 채널에 보내는 얇은 결합만으로도 간단하게 데이터를 보낼 수 있기 때문에 서비스와 프래그먼트 간에 데이

Watchout에서 화재 경보음과 같은 소리를 감지했을 때 이에 대한 정보를 IoT 디바이스에게 전달해야 하는데 액티비티와 서비스 간의 간단한 데이터 전송이므로 Intent 객체를 생성하여 putExtra 메소드를 사용하여 데이터를 전달하도록 하였다. ForegroundService는 이것을 getStringExtra() 메서드로 받아 블루투스 연결된 IoT 디바이스에게 전달한다. 또한, BroadCast Receiver를 통해 블루투스 연결과 끊김을 체크하였고, 연결이 원치않게 끊겼을 경우 블루투스 연결 화면으로 넘어 갈 수 있도록 해주었다.

단어 인식을 위해 마이크 입력을 무한히 처리해주고자 여러가지 API 및 라이브러리를 시도해보았다. Gcloud의 Speech to Text(STT) API를 무제한으로 사용하고자 하였으나, 무제한 인식 안드로이드 코드가 deprecated 되어 사용할 수 없다는 문제가 발생하였다. 네이버의 Clova Speech Recognition (CSR)의 경우 안드로이드의 Speech Recognizer와 기능상에서 차이가 없었으며 비용 문제가 발생하였다. 따라서 안드로이드의 Speech Recognizer를 활용하여 단어 인식 기능을 구현하였다.



▶ Gcloud 클라우드 setting 화면

3) 개발 내용

가. 위험 소리 인식 'Watch-Out'

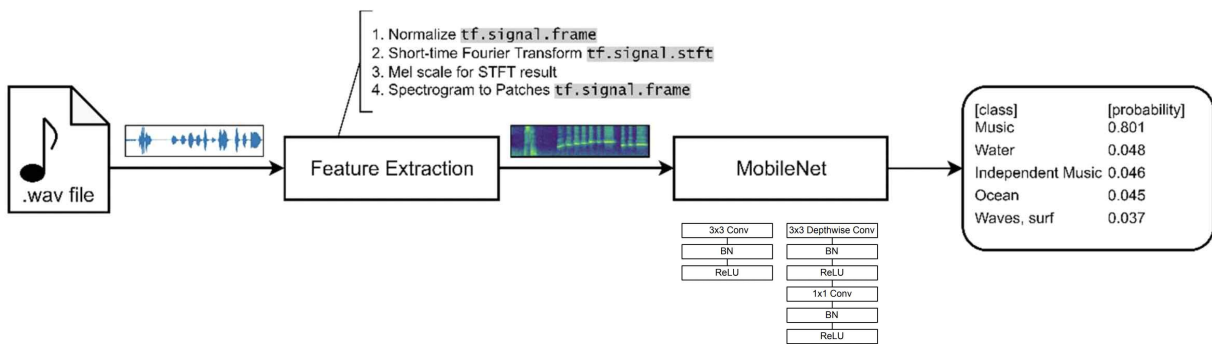
○ 기능설명

사용자는 애플리케이션에서 'Watch-Out' 화면을 볼 수 있다. 'Watch-Out' 페이지에서 사용자는 스위치 on/off를 통해 위험 소리 감지 기능을 키고 끌 수 있다. 만약, Watch-Out 기능을 on 했을 경우, 애플리케이션은 계속해서 소리 감지 및 소리 분류를 진행한다. 이는 사용자가 스위치 off 하기 전까지 계속해서 작동함으로써, 애플리케이션을 닫은 상황에서도 실시간으로 소리를 인식하여 알람을 주도록 하여 즉각적인 위험 상황 인지에 더욱 도움이 되도록 한다. 이러한 알람은 애플리케이션 푸시 알람 및 블루투스 통신을 통한 IoT 디바이스 LCD에도 실시간으로 전송해주어 사용자가 휴대폰을 사용하고 있지 않을 경우에도 해당 상황을 쉽게 감지할 수 있도록 통합적인 시스템을 제공한다.

Watch-out 기능이 켜지면 애플리케이션은 계속해서 위험 소리 (ex. 사이렌, 화재경보음)를 인지한다. 이러한 sound classification은 Tensorflow lite Yamnet 모델을 바탕으로 Teachable Machine으로 음성 분류모델을 학습시키고 추출한 TFLite 모델을 안드로이드 애플리케이션에 적용시킴으로써 작동한다. 애플리케이션 내부에서는 약 521가지의 소리를 classification하며 그중 위험한 소리라고 인식되는 특정 소리 일 경우에만 따로 인식하여 사용자에게 알람을 주도록 구현하였다.

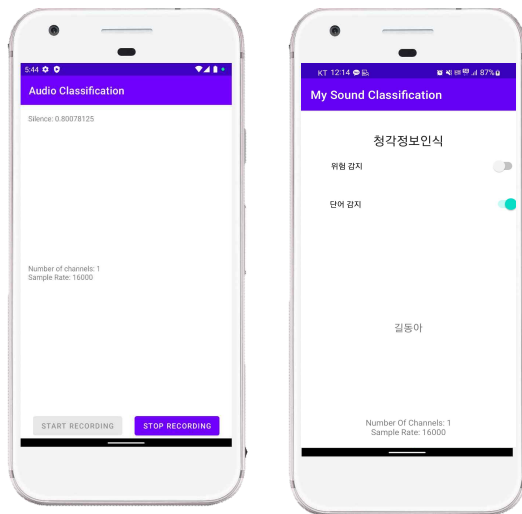
○ 구현방법

모바일 애플리케이션 마이크를 통해 실시간 오디오 분류를 하기 위해서는 gradle에 tensorflow audio 종속 항목을 추가해주어야 한다. 로드할 모델의 이름으로 변수를 생성하고, 모델에서 예측할 수 있는 최소 기준인 probability Threshold를 0.2로 설정하였다. 오디오 작업 라이브러리에 정의된 Audio classifier 는 모델을 로드할 준비 및 추론 실행, 오디오 녹음기 생성에 필요한 메소드를 제공한다. 추론을 위한 녹음을 저장하고 녹음기의 형식 사양을 빌드하는 텐서 변수를 만들고 Audio recorder를 만들어 녹음을 시작한다. 이후 단계에서는 추론을 진행하기 위해 0.5초 간격으로 실행되는 타이머 스레드를 생성하였다. 해당 스레드에서 먼저 녹음을 오디오 텐서에 로드하고 분류기에 전달한다. 이 때, probability Threshold 보다 큰 값들만 필터링하여 해당 label이 위험한 소리로 판단된 경우에 위험이 감지되었음을 알리도록 구현하였다. 따라서, 위험이 감지되었음을 인식하는 타이머 스레드가 8초 간격으로 동작하도록 하여 Notification Channel을 통해 어플리케이션 push 알람을 전송하고 Toast message를 띄우도록 한다. 화재 경보 소리가 감지된 경우에는, 119 긴급 전화를 할 수 있는 다이얼 화면으로 전환해줄 수 있도록 예외 처리 해주었다. 이때, 다이얼 화면으로 전환하기 위해 DIAL_PHONE에 대한 퍼미션 세팅을 해주었다. 다이얼 화면으로 전환하기 위해 android.intent.action.DIAL 소스를 사용하였으며, Intent의 데이터를 Uri.parse(tel:119)로 세팅하여 해당 객체를 startActivity 함수의 인자로 전달해줌으로써 119 긴급 전화 다이얼 화면으로 전환할 수 있도록 구현하였다. 아두이노와 계속해서 통신하기 위해서는 Foreground Service에서 동작하는 블루투스 서비스를 이용한다. 블루투스를 통해 아두이노에도 해당 위험 감지 알람을 전송해주기 위하여 위험이 감지되면 블루투스 서비스 Intent 객체를 생성하고 값을 세팅을 하여 startService 함수를 호출함으로써 데이터를 전달하도록한다.

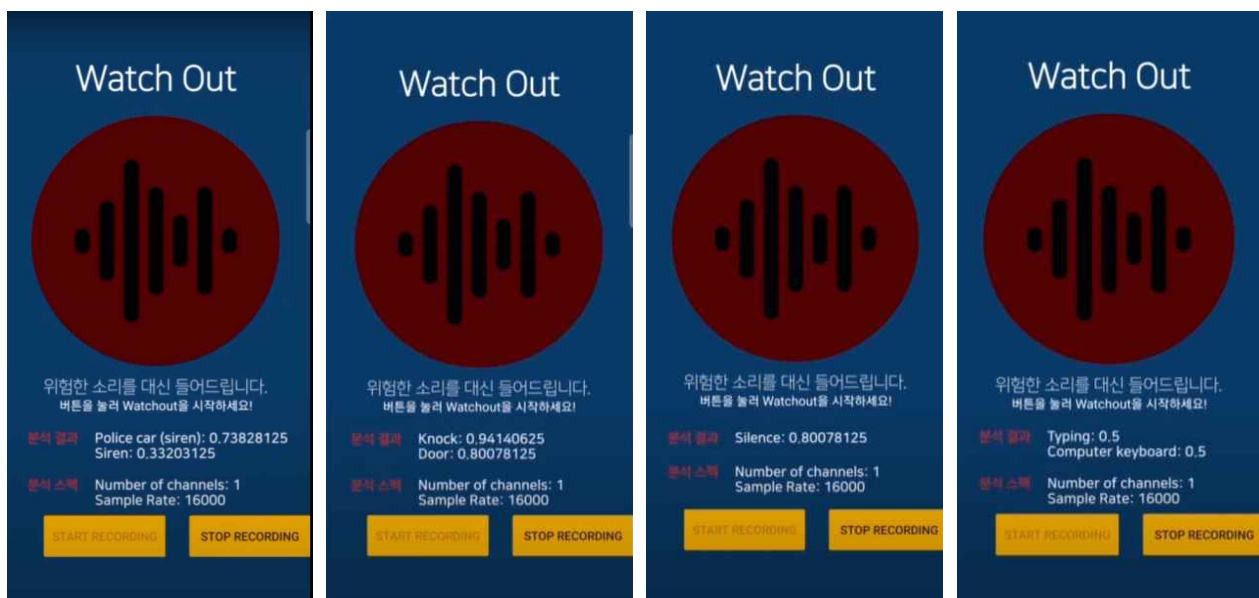


▶ 모델 아키텍처

Yamnet은 음원데이터로부터 특징을 추출한 후, 추출된 특징들을 CNN 기반 모델인 MobileNet에 학습시켜, 입력된 음원이 어떤 종류의 음원인지 classification한다. 이에 따라, 애플리케이션 내부에서는 실시간으로 모든 소리에 대한 sound classification을 진행한다.



해당 사진은 실시간으로 sound classification을 진행하는 애플리케이션 개발을 진행한 모습이다. 소리에 따라 Police car, siren, Knock, Door, Silence, Typing, Computer keyboard 등 실시간으로 소리를 분류해줌을 확인할 수 있다. 이러한 인식된 라벨을 바탕으로 위험 소리가 발생한 경우에만 알림을 줄 수 있도록 구성을 한 것이다.



나. WordList 단어 추가 및 삭제

○ 기능설명

사용자는 'Word List' 화면에서 본인이 듣고자 하는 단어를 등록하고 삭제할 수 있다. 화면의 + 버튼을 누르면 text를 입력할 수 있는 프레임과 저장 버튼과 취소 버튼이 출력된다. 공백인 상태에서는 저장할 수 없도록 경고 메시지를 출력해주고 공백이 아닐 경우 '저장했어요' 메시지를 출력하고 DB에 저장한다. 저장된 단어를 삭제할 경우, '삭제했어요' 메시지를 출력해주고 DB에서 삭제한다. 저장하기 전의 text 프레임의 취소 버튼을 누르면 별도의 메시지 출력 없이 단어 저장을 취소한다. 이렇게 등록된 단어 리스트를 바탕으로 애플리케이션은 해당 단어 소리를 인식하는 기능을 수행함으로써, 사용자가 원하는 단어를 손쉽게 듣고 감지할 수 있도록 한다.

○ 구현방법

사용자가 저장한 단어는 모든 애플리케이션 액티비티 및 서비스에서 활용 가능해야 한다. 따라서 이를 위해 안드로이드의 Application 클래스를 활용한다. Application 클래스는 전역 응용 프로그램 상태를 유지하기 위한 기본 클래스로 하위 클래스를 만들고 이 하위 클래스의 정규화 된 이름을 manifest 파일의 <application> 태그에서 "android:name" 속성으로 지정해 고유한 구현을 제공한다. Application 클래스를 상속한 SavedWords 클래스를 만들고 이를 매니페스트에 등록한 후 DB에 저장된 단어들을 ArrayList에 저장하여 애플리케이션 전체에서 참조할 수 있도록 함으로써 단어 인식이 원활하게 이루어지도록 한다.

다. WordList 단어 소리 감지

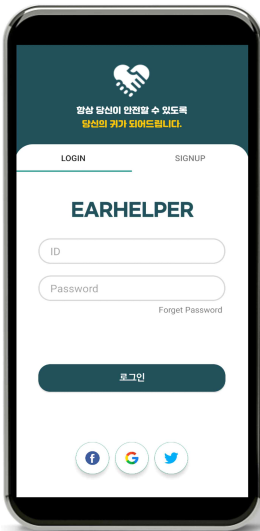
○ 기능설명

사용자는 'Word Recognition' 화면에서 단어를 등록하는 'Word List' 화면으로 넘어가거나 스위치를 on하여 저장된 단어를 감지할 수 있다. 스위치를 on 한 후, 등록된 Word List의 단어 소리가 인지되면 애플리케이션에서는 푸시 알람을 주고 IoT 디바이스 LCD에 정보를 출력해줌으로써 사용자에게 실시간 알람을 제공하여 사용자는 본인이 듣고자 하는 단어를 빠르게 인지할 수 있다.

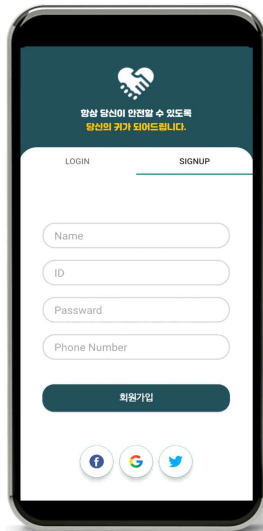
○ 구현방법

단어 감지를 위한 음성인식을 수행하기 위하여 안드로이드의 SpeechRecognizer 라이브러리를 활용하였다. 이때, 구글의 서버와 통신이 필요하기 때문에 android.permission.INTERNET에 대한 등록이 필요하다. 그 후, RecognizerIntent 객체를 생성하고 해당 객체의 언어를 한국어로 설정하였다. 다음으로 객체가 사용할 패키지 설정을 해준다. createSpeechRecognizer 함수를 이용하여 객체를 생성하고 recognizer에 리스너를 등록한다. 그리고 앞서 설정해준 RecognizerIntent 객체를 startListening 함수의 인자로 넘겨주어 음성 인식을 수행한다. 리스너는 RecognitionListener 클래스를 활용하여 해당 클래스의 메소드들을 오버라이드 하여 원하는 목적에 맞게 구현할 수 있다. 음성 인식이 시작되는 단계의 함수인 onReadyForSpeech 함수에서는 음성 인식이 시작됨을 알리는 토스트 메시지를 애플리케이션에 출력하도록 한다. 음성 인식이 완료된 후 결과를 출력하는 단계의 함수인 onResults 함수에서는 애플리케이션 전역에서 참조 가능한 단어가 있는지 없는지 확인한 후 일치하는 단어가 있을 경우 Notification Channel을 통해 알림을 주도록 한다. 또한, Intent를 통해 블루투스 서비스에 값을 전달함으로써 단어가 인식되었음을 애플리케이션에 푸시 알림을 전송하며, IoT 디바이스에도 어떠한 단어가 인식되었는지 LCD에 출력함으로써 알림을 주도록 구현하였다.

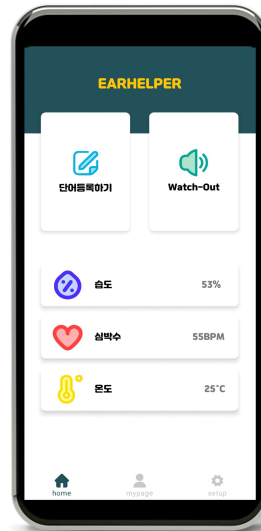
4) 개발 결과



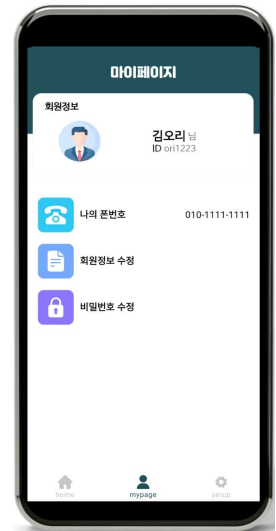
▶ 로그인 화면



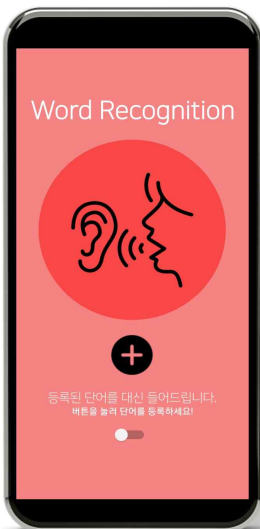
▶ 회원가입 화면



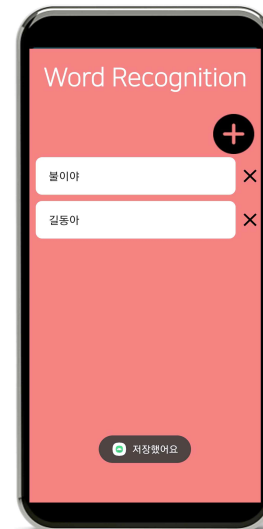
▶ 메인페이지 화면



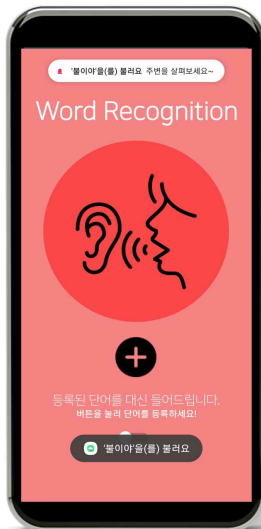
▶ 마이페이지 화면



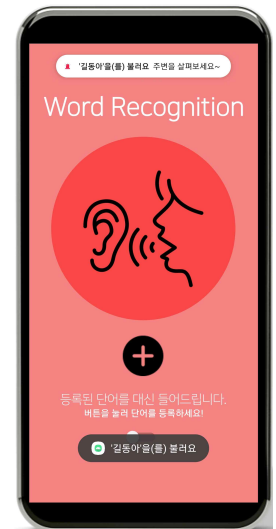
▶ 단어 인식 화면



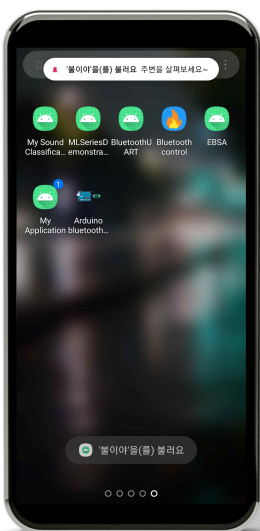
▶ 단어 등록 화면



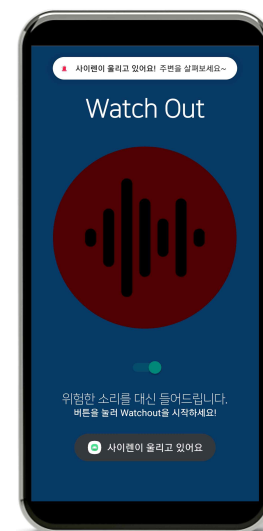
▶ '불이야' 단어인식



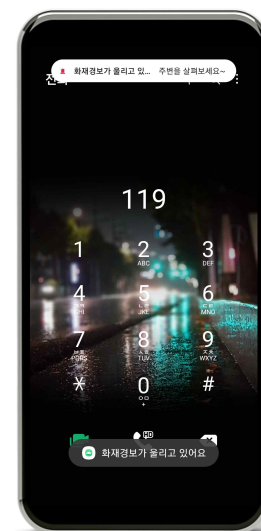
▶ '길동아' 단어인식



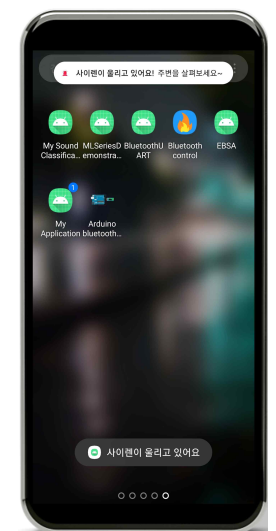
▶ 앱 외부에서도 단어 인식 알림 전송



▶ Watch-out 화면



▶ 119 연결 화면



▶ 앱 외부에서도 위험 감지 알림 전송

[하드웨어 부분]

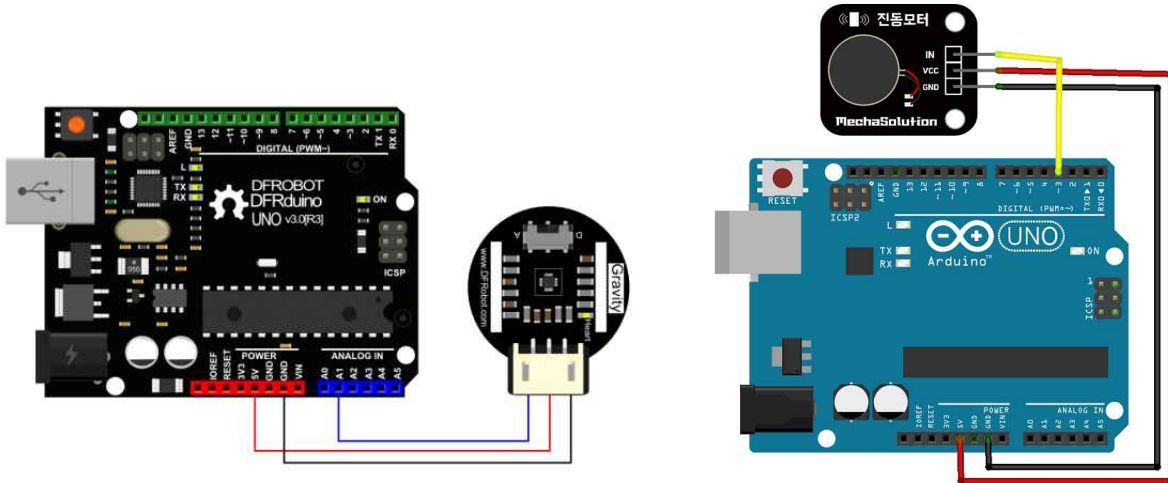
1) IoT의 필요성

집에서는 스마트폰을 항상 들고 다니지 않는다. 이때 청각장애인들의 소리 인식 알람을 어떻게 처리해줘야 할지 고민하다 IoT 하드웨어를 만들어 집에 두면 굳이 집에서 스마트폰을 들고 있지 않아도 중요한 소리 인식 및 알람을 줄 수 있다고 생각하였다. 스마트폰에서 위험한 소리를 인식하여 IoT의 LCD에서 인식된 소리의 정보를 밝게 띄어줌으로써 긴급 이벤트 발생을 빠르게 인식할 수 있도록 하고자 한다. 만약 화재경보음이 인식되면 "***Fire alarm*** Look around you"를 LCD에 출력해주어 무슨 소리가 인식되었는지 알려준다. 이처럼 IoT, 하드웨어를 도입하게 된다면 스마트폰을 가지고 있지 않은 순간에도 긴급한 알람을 인식할 수 있다는 점에서 스마트폰과 블루투스 연결을 하여 사용할 수 있는 기기를 제작할 필요성을 느끼게 되었다. 따라서, 현재 청각장애인들이 겪는 위험과 현실적인 문제들을 인식하여 이를 개선할 수 있는 기능을 가진 하드웨어와 앱 개발을 함께함으로써 편의와 안전을 제공하고자 한다.

2) 개발 내용

○심박수모듈&진동모듈

심박 센서는(SEN0203)을 사용하였고, 진동모듈은 진동모터 모듈(한글보드)를 사용하였다. 심박센서는 주로 모세혈관이 많은 손 끝과 손목에서 측정이 가능하다. 센서 중앙 부분에 위치한 LED는 밝은 초록색 빛이 방출된다. 모세혈관에 의해 반사된 빛은 LED아래에 위치한 Detector로 측정한다. 심장 박동시 증가하면 반사되어 돌아오는 빛의 양이 줄어들며, 혈류의 양이 적어지면 반사되어 돌아오는 빛의 양이 늘어나게 되는데 이를 Detector로 들어온 빛의 양을 가지고 심장 박동수를 측정하게 되는 원리로 동작한다.



▶set PIN1 as signal

▶set PIN3 as signal

Arduino UNO	Bluetooth Module (HC-06)	Heart rate Module	Vibration Module
5V	VCC	VCC	VCC
A1		DATA	
GND	GND	GND	GND
D8	TXD		
D7	RXD		
D3			OUT

▶ PIN Table

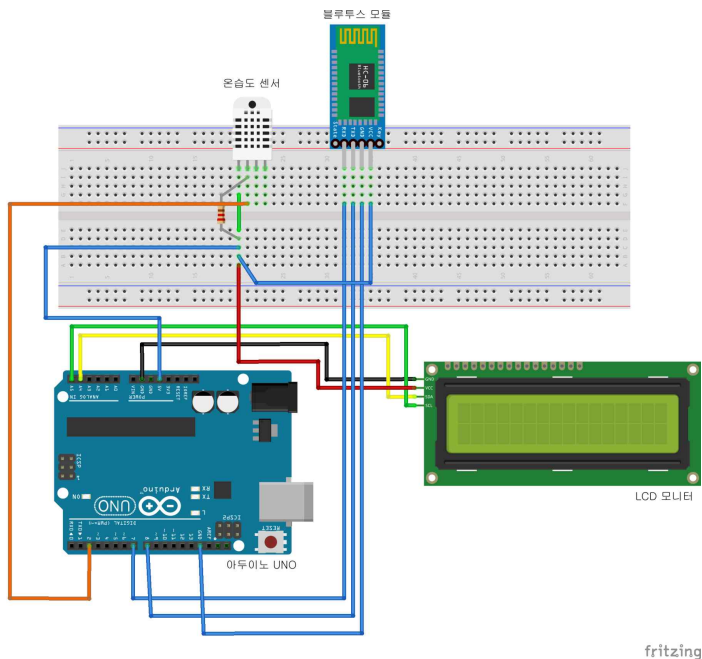
심박센서는 블루투스 통신으로 결과 값을 스마트폰에 띄운다. 블루투스 모듈은 수신을 위한 RX핀은 7번에, 송신을 위한 TX핀은 8번으로 설정한다. 심박센서의 Data핀은 아두이노 우노 보드에 A1으로 연결해주고, 진동 모터 모듈의 출력을 위한 핀은 3번으로 설정한다.

아두이노 코드로 블루투스를 구현할 때, TX, RX 연결 시 아두이노와 블루투스 모듈을 반대로 연결한다. 따라서 코드에서 RX를 8번으로, TX를 7번으로 설정한다. 블루투스 설정을 위하여 블루투스 시리얼 통신 라이브러리를 추가해주었고, SoftwareSerial bluetooth(BT_RXD, BT_TXD)를 통해 RX와 TX를 설정하였다. 심박센서의 경우에도 DFRobot_Heartrate 라이브러리를 추가해주었다. 심박센서의 경우 출력의 형식이 디지털 모드와 아날로그 모드가 있는데 디지털 형태로 출력하기 위해서 DFRobot_Heartrate heartrate (DIGITAL_MODE)로 설정해주었다.

setup 함수에서 시리얼과 블루투스는 9600 보레이트로 통신하도록 세팅해준다. loop 함수 내에서는 앞서 설정한 심박센서의 A1핀으로부터 getValue 함수를 통해 심박 수를 받아오고, 이를 getRate 함수를 통해 rateValue로 바꿔준다. 변환해준 rateValue 값이 유효하다면 블루투스로 송신하도록 한다. 또한, 블루투스로부터 받은 데이터가 있다면 출력핀을 3번으로 설정한 진동 모터 모듈을 analogWrite 함수와 delay 함수를 사용하여 3초간 100 / 255의 세기로 진동하고 꺼지도록 구현하였다.

○온습도 센서 모듈

온습도 센서는 DHT-22를 사용하였다. 습도 센서는 전극 2개와 가운데 수분 유지판(박막 폴리머)으로 구성되어 있다. 온습도 센서는 습도에 따라 저항 값이 변하게 되는데, 습도를 측정할 때는 상부 전극과 하부 전극 사이에 있는 저항 변화를 측정함으로써 공기 중 습도 변화를 감지할 수 있다. 온도센서는 고온에서 소결된 반도체 세라믹으로 온도에 따라서 물질의 저항값이 변하는 소재를 이용해서 저항값의 변화를 감지해서 온도를 측정하게 된다.



▶ 온습도 구성 회로도

Arduino UNO	Bluetooth Module (HC-06)	Humidity & temperature sensor (DHT-22)	Vibration Module	LCD Display
5V	VCC	VCC	VCC	VCC
A1		DATA		
GND	GND	GND	GND	GND
D8	TXD			
D7	RXD			
D3			OUT	
A4				SDA
A5				SCL

► PIN Table

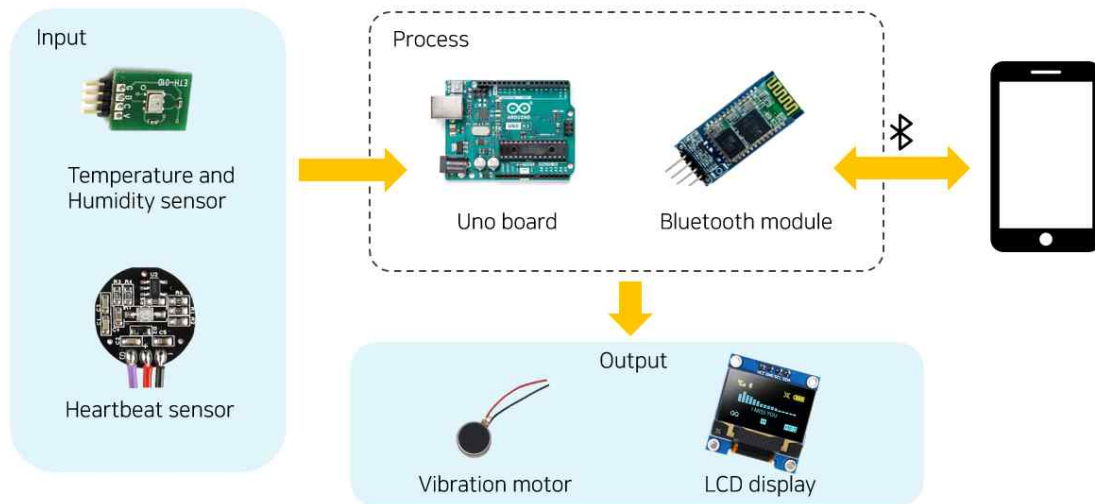
위 회로도 와 같이 디바이스는 온습도 센서, 블루투스 모듈, LCD로 구성된다. 블루투스 모듈의 수신을 위한 RX핀은 7번에, 송신을 위한 TX핀은 8번으로 설정한다. 온습도 센서는 Data핀을 A1으로 설정한다. LCD는 클럭 신호를 전송하는 SCL(Serial Clock)핀을 A5로, 데이터 전송을 위한 SDA(Serial Data)핀을 A4로 설정한다. 아두이노 코드로 블루투스를 구현할 때, TX와 RX 핀 연결 시 아두이노와 블루투스 모듈을 반대로 연결한다. 따라서 코드 상으로는 RX를 8번으로, TX를 7번으로 설정한다. setup 함수에서 시리얼과 블루투스는 9600 보레이트로 통신하도록 설정한다. loop 함수 내에서 온습도 측정값을 블루투스로 송신하고 블루투스로부터 받은 데이터가 있다면 이에 맞게 LCD에 출력하도록 한다.

온습도 센서 아두이노 코드에서 블루투스 시리얼 통신 라이브러리를 추가해 블루투스 통신을 가능하도록 하였다. 그리고 위에서 설명한 블루투스 송수신을 위한 코드를 작성해 수신을 위한 RX핀은 8번에, 송신을 위한 TX핀은 7번으로 설정한다. 블루투스 통신을 위한 코드는 setup함수에서 시리얼과 블루투스는 9600 보레이트로 통신하도록 설정한다. 또한 LCD에 출력하기 위해 LCD를 초기화해주고 LCD 불빛을 켜주는 작업을 위한 코드도 추가해주었다. 그리고 블루투스 통신이 가능하게 된다면 loop함수에서 블루투스에서 보낸 내용을 시리얼 모니터로 전송하게 해준다.

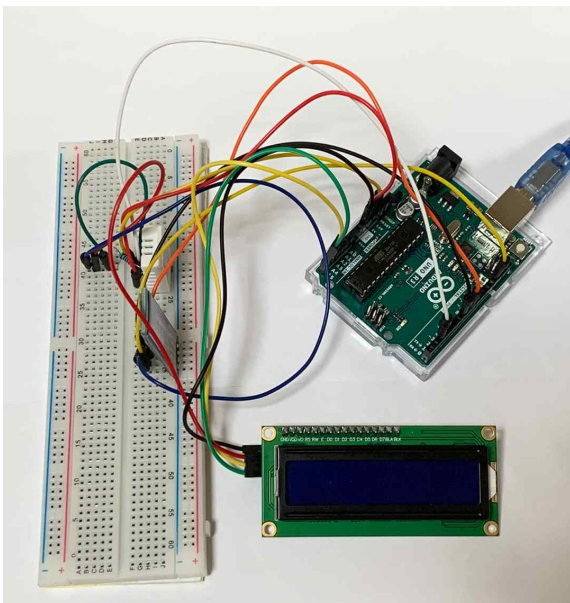
또한, 긴급 알람(화재경보음, 사이렌)이 인지되면 LCD모니터에 출력해주는 코드도 같이 적어주었다. 먼저 사이렌과 화재경보음을 긴급알람에서 1과 2로 분류하여 소리가 인식되었을 때 LCD에 출력하게 해주

었는데, 먼저 데이터로 "1"이 들어오게 되면 LCD를 초기화 시켜주며 "***Siren***"을 출력해주어 사이렌 소리가 인식되었음을 출력해주었다. 또한 "Look around you."의 문구도 추가해주어 청각장애인들의 빠른 대처를 유도하였다. 데이터로 "2"가 들어오게 되면 화재경보음으로 분류하여 LCD를 다시 초기화시켜 주어 "***Fire alarm***"을 출력해주며 화재경보음이 인식되었다는 것을 사용자에게 인지할 수 있도록 해당 문구를 출력하게 해주었다.

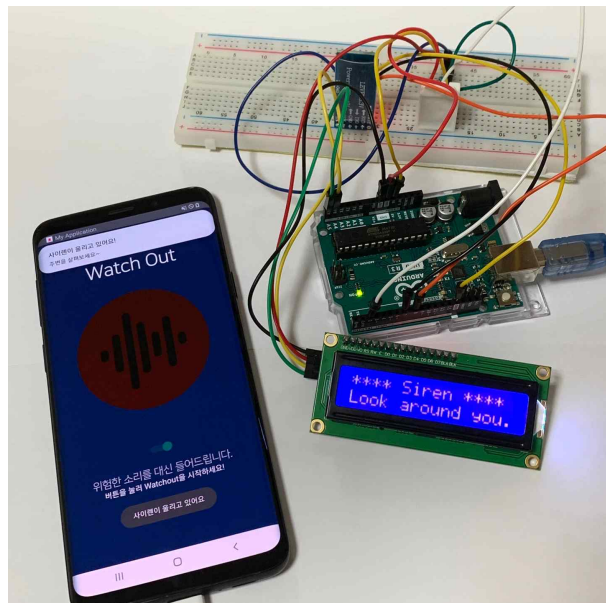
3) 개발 결과



▶하드웨어와 블루투스 통신 구성도



▶개발 하드웨어



▶어플리케이션과 하드웨어가 통신하는 모습

4. 기대 효과 및 활용 방안



○ 위험 상황에 대한 신속한 대처

- 청각 장애인이 해당 애플리케이션을 사용함으로써 사이렌이나 경보음이 울리는 것과 같은 위험한 상황이 발생하는 경우에 진동 및 알람을 통하여 신속히 대피할 수 있도록 한다. 이러한 알람은 애플리케이션 알람뿐만 아니라 IoT 디바이스의 디스플레이 화면에도 정보를 출력해줌으로써 휴대폰을 확인하지 못할 경우에도 신속히 대처할 수 있도록 하였다. 더하여 화재경보음 소리가 인지되었을 경우에는 119 다이얼 화면으로 전환해주어 긴급 전화를 연결할 수 있도록 한다. 이는 상황에 따라 보호자의 번호로 연결하도록 할 수도 있다.

- 또한, 본 프로젝트에서는 사용자가 듣기를 원하는 특정 단어를 등록하고 인식할 수 있도록 한다. 예를 들어 '불이야'와 같은 단어를 등록하면, 해당 단어가 들릴 시 인식하여 알람을 줌으로써 특정 단어에 의한 위험한 상황을 빠르게 인지할 수 있도록 한다. 이로써 전체적인 위험 재난 상황을 빠르게 인지하도록 함으로써 안전사고 예방에 기여할 수 있다.

○ 청각장애인의 사회적 고립 해결

- 단어 인식 기능을 통해 사용자의 이름을 등록하여 '길동아'와 같이 이름이 인식되면 사용자에게 알람 및 진동을 주어 사용자를 부르는 상황을 쉽게 인지할 수 있도록 한다. 더하여, 일상생활 속 자주 접할 수 있는 상황의 단어를 대신 들어주는 역할을 해줌으로써 청각장애인들의 사회적 고립감을 해결하고 일상 생활 속 편리함을 제공한다.

○ 애플리케이션의 높은 확장성

- 이러한 위험 소리 인지 기능 및 단어 인식 기능은 청각장애인이 아니더라도 누구에게나 필요한 기능으로 확장될 가능성이 높다. 대부분의 사람들이 이어폰을 착용하거나 휴대폰에 집중하며 걷어다니는 상황을 자주 만날 수 있는데, 이의 경우에는 주위 차량 경적 소리 및 위험 상황에 대해 인지하지 못하여 사고가 발생할 수 있다. 따라서, 휴대폰에 집중하고 이어폰을 낀으로써 소리를 인지하지 못하는 상황에서도 해당 애플리케이션이 재난 안전 사고 예방에 큰 도움을 줄 수 있다. 즉, 청각장애인만을 위한 애플리케이션에 그치는 것이 아니라 소리 인지로 인해 다양한 일상생활 속 사고를 예방할 수 있는 애플리케이션으로서의 큰 확장성을 지닌다.

○ IoT 디바이스의 차별성

- 블루투스 통신 IoT 디바이스를 개발함으로써, 주위에서 감지된 소리 정보를 시각적으로 빠르게 확인할 수 있도록 한다. 이는, 애플리케이션과 통신하며 휴대폰을 확인하지 못하는 상황에서도 현재 상황을 빠르게 인지할 수 있도록 한다는 점에서 다른 시스템과는 큰 차별성을 지닌다. 더하여, 해당 IoT 디바이스는 다양한 센서를 통해 온습도, 심박수 등을 측정하여 애플리케이션에 정보를 제공한다는 점에서 전반적인 애플리케이션 사용자의 편리성을 강화할 수 있다.

○ 안드로이드와 하드웨어 디바이스 연동을 통한 통신 이해도 상승

- 블루투스 통신 IoT 디바이스를 개발하여 안드로이드 애플리케이션과 통신할 수 있도록 시스템을 구성하는 과정에서 통신에 대한 이해도를 높인다. 블루투스 통신을 개발하고 구현해내는 과정에서 연결이 안정적으로 지속될 수 있는 방안에 대한 모색을 통해 애플리케이션과 하드웨어 간의 통신 개발에 대한 지식을 확장한다.

○ 소프트웨어, 하드웨어를 아우르는 컴퓨터공학적 문제 접근 및 능력 향상

- 애플리케이션 제작에 있어서 텐서플로우와 stt api를 사용해봄으로써 오디오 분류 모델 생성 과정과 활용 방법을 익히며 인공지능에 대한 지식을 확장한다. 이에 애플리케이션 개발에 그치지 않고, IoT 디바이스 하드웨어를 제작함으로써 온습도 감지, 심박수 감지 등 다양한 센서를 활용하여 프로젝트를 구성하는 능력을 향상시킨다. 이처럼 여러 센서와 인공지능 및 API를 활용하여 앱을 제작하는 과정에 있어서 여러 시행착오를 겪으며 정확도를 높이는 과정을 통해 엔지니어에게 필요한 컴퓨터 공학적 문제 접근 및 해결 능력을 키우며 팀원들과 협업하는 팀워크를 기른다.

○ 학부생 학술대회 논문 발표

- 제작한 프로젝트에 대해 팀원들과 함께 논문을 작성해봄으로써 국내 학술대회에 투고하는 경험을 쌓는다. 이에 완성된 프로젝트에 대한 학술대회 발표 경험까지 쌓으며 실무 역량과 함께 연구 역량을 강화한다.

○ 기업과의 연계

- 프로젝트에 관해 수차례 기업 멘토와의 회의를 통해 아이디어를 도출해내고 해당 아이디어에 대한 현실성을 함께 고려하여 현실가능한 프로젝트를 개발하고, 기업 멘토의 경험을 바탕으로 프로젝트 아이디어에 대해 구체화 하여 더욱 발전된 프로젝트를 설계할 수 있다. 해당 과정에서 기업과 연계하여 회의하고 피드백을 바탕으로 프로젝트를 발전해나가는 역량을 강화한다.

6. 참여인력(세부)

지도교수	소속	컴퓨터학부		성명	고석주
참여인력 (산업체)	기업명	성명	직위	전화	Email
	(주)신라시스템	김민희	부사장	010-6509-3399	mhkim@silasystem.com
과제 참여 학생	소속(학과)	학위과정 (성별)	학번	성명	담당업무
	컴퓨터학부	학사과정 (여)	2019113926	신현지	애플리케이션 개발 및 하드웨어 개발
	컴퓨터학부	학사과정 (여)	2019111594	엄다연	애플리케이션 개발 및 하드웨어 개발
	컴퓨터학부	학사과정 (여)	2019113667	이수경	애플리케이션 개발 및 하드웨어 개발
	컴퓨터학부	학사과정 (남)	2016111661	임지혁	애플리케이션 개발 및 하드웨어 개발

※ SW 종합설계(Capstone Design) 프로젝트 수행 시에 “설계 구성요소”와 “현실적 제한 요소”를 고려해야함. 아래 예시(이탈릭체)를 참고하여 프로젝트 수행시 고려했던 사항을 기술하시오.

종합설계 구성요소	
목표 설정	<p>→ 일상생활에서 다양한 상황에서 소리 인식에 불편을 겪는 청각장애인들을 해당 소프트웨어의 사용자로 설정한다.</p> <p>▶ 사용자 요구분석</p> <p>① 일상생활 속 자주 들을 수 있는 단어를 인식하고자 한다.</p> <p>② 사이렌, 경적, 화재경보음, 등의 위험한 상황에서 발생하는 소리를 인식하고자 한다.</p> <p>③ 인식한 결과에 있어서 시각적, 촉각적으로 빠르게 인지하고 싶다.</p> <p>④ 일상생활 속 다양한 정보를 부가적으로 알고 싶다.</p>
합성	<p>▶ 개발 기능</p> <p>① 위험 감지 : 오디오 분류 학습을 통해 위험 소리를 인지한다.</p> <ul style="list-style-type: none"> - 스위치를 사용하여 활성화 및 비활성화 할 수 있다. - 위험이 인식되면 애플리케이션 및 IoT 디바이스에 알림을 준다. - 화재경보음의 경우 119 긴급 전화를 할 수 있도록 한다. <p>② 단어 인식 : 사용자가 원하는 단어 소리를 대신 인식한다</p> <ul style="list-style-type: none"> - 인식할 단어를 등록하거나 삭제한다. - 스위치를 사용하여 활성화 및 비활성화 할 수 있다. - 단어가 인식되면 애플리케이션 및 IoT 디바이스에 알림을 준다. <p>③ IoT 디바이스 : 다양한 센서를 이용한 정보를 제공하고 알림을 제공한다.</p> <ul style="list-style-type: none"> - 블루투스를 통해 센서의 측정값을 전달하여 애플리케이션에서 출력한다. - 블루투스로부터 수신한 데이터 값을 기반으로 LCD 출력 및 진동한다. <p>▶ S/W 도구</p> <p>→ 일상생활 속 대부분을 함께하는 모바일 폰 어플리케이션을 활용해 ‘오디오 분류를 통한 위험 소리 인식’, ‘특정 단어 인식’을 제공하는 시스템을 개발하기로 결정하였다.</p> <p>① Front-end : 위 기능들을 사용자에게 시각화할 수 있는 편의성이 보장된 UI 디자인을 설계해야한다. ⇒ Android Studio</p> <p>② Back-end : 회원 가입, 로그인, 단어 등록 및 삭제, 등의 기능을 구현해야 한다. ⇒ node.js, AWS 서버</p>

<p>분석</p>	<p>① Android Studio : 구글이 안드로이드 앱 개발을 위해 만든 통합 개발 환경이다. 사용자의 편의성이 보장되는 UI 디자인을 설계한다.</p> <p>② AWS 서버 & node.js : 사용자는 단어를 등록해 특정 단어에 대한 알람을 받을 수 있는데 서버를 통해 사용자가 등록한 단어를 관리한다. 또한, 관리자 서버로서 사용자를 관리할 수 있도록 한다.</p> <p>③ TensorFlow Lite ML 모델을 개발하고 학습시키는 데 도움이 되는 핵심 오픈소스 라이브러리로 유연하게 ML이 접목된 어플리케이션을 빌드할 수 있다. 경적소리, 화재경보음과 같은 사운드를 학습시키고 추론하여 사용자에게 알림을 줄 수 있도록 구현한다.</p> <p>④ Speech-to-Text API 구글 텍스트 음성 변환은 안드로이드 운영 체제용으로 구글이 개발한 스크린 리더 어플리케이션이다. 사운드를 듣고 text로 변환하여 특정 단어가 인식되었을 때 알람을 줄 수 있도록 구현하고자 한다.</p> <p>⑤ 구글의 티처블 머신 Teachable Machine은 누구나 머신러닝 모델을 쉽고 빠르게 만들 수 있도록 제작된 웹 기반 도구이다. 라벨링된 샘플들을 티처블 머신으로 학습시켜 사운드를 분류할 수 있는 모델을 만들고, 이 모델을 내보내 특정 소리를 인식하여 알람을 주는 기능 구현을 더하고자 한다.</p>
<p>제작</p>	<p>① 위험 감지를 위한 오디오 분류 : TensorFlow Lite를 활용하여 구현 → 하나의 TimerTask는 추론과 위험 소리를 인지하여 플래그를 세우는 역할을, 또 다른 하나는 플래그를 확인하여 주기적으로 알람을 주고 블루투스를 통해 아두이노에 데이터를 보내어 위험이 감지되었음을 알리도록 구현하였다.</p> <p>② 특정 단어 인식 : Speech-to-Text API를 활용하여 구현 → 안드로이드에서는 구글의 STT를 내장 API로 지원한다. 이를 이용하여 사용자가 등록한 특정 단어를 인식하고 알람을 주고 블루투스를 통해 아두이노에 데이터를 보내어 특정 단어가 인식되었음을 알리도록 구현하였다.</p>
<p>시험</p>	<p>→ 사용자 요구사항 및 설계 사항이 제대로 반영되었는지 확인하기 위해 다양한 상황을 가정하고 시험을 실시하였다.</p> <p>① 단어 인식 : "불이야", "수지야"와 같은 단어를 등록하고 단어를 말했을 때 사용자에게 올바르게 알림이 오는지 확인하였다. 더하여, 단어를 삭제한 후에 단어를 인식하였을 경우 제대로 삭제가 반영되는지 테스트하였다.</p> <p>② Watch-out : 화재경보음, 사이렌 등의 소리가 들려올 때 어플리케이션에서 알림을 주는지와 IoT 디바이스의 LCD 모니터에 그에 따른 정보가 출력되는지 확인하였다.</p> <p>③ 블루투스 : IoT 디바이스와 블루투스 연결을 하고 홈 화면에 온습도, 심박수 센서 값이 올바르게 나오는지 확인하기 위해 온습도 및 심박수를 실시간으로 변경해보면서 시험을 실시하였다. 더하여, 액티비티를 전환하고 다른 기능을 수행할 때도 어플리케이션과 IoT 디바이스의 블루투스 연결이 지속되는지 확인하였다.</p>
<p>평가</p>	<p>다른 팀이 개발한 소프트웨어에 대해 최종 보고와 데모를 보고 창작성, 난이도, 완성도의 3가지 항목을 기준으로 평가를 실시하였다.</p>

종합설계 (현실적) 제한조건

경제성	<p>IoT 디바이스를 제작하기 위해서 아두이노 우노 및 각종 센서들을 구입하여 사용하였다. 제공할 주요한 부가 정보로는 온습도, 심박수가 있는데 이를 위해 온습도, 심박 센서가 필요하고 애플리케이션과 통신하기 위해서 블루투스 모듈이 필요하다. 또한 위험 소리를 감지하고 특정 단어를 인식했음을 인지시켜주기 위해서 LCD와 진동 모터 모듈을 사용하였다. 서버 구축을 위해 aws 서버를 사용하였는데, 이는 무료용으로 개설하였으므로 서버 사용 기간이 길어지면 추후비용이 발생 될 수 있음을 고려해야 한다.</p>
안전성	<p>애플리케이션의 기능상, IoT 디바이스와 통신하기 위해 블루투스를 사용하기 위해서 위치 정보에 액세스 할 수 있어야 하고, 위험 소리 감지, 특정 단어 인식을 위해서 마이크를 사용할 수 있어야 한다. 수집된 위치 정보나 마이크 입력은 민감한 정보일 수 있으므로 보호되어야 한다. 이를 위해서 안드로이드 스튜디오에서 제공하는 프로가드 설정을 해주었다.</p> <p>프로가드란 코드 난독화를 수행하여 디컴파일 시 애플리케이션의 코드가 노출되는 것을 방지하고 불필요한 메서드를 제거하여 멀티덱스를 피하도록 최적화하는 역할을 수행한다.</p>
신뢰성	<p>① 위험 소리 감지의 경우 소리에 따라 검출률과 정확도에 문제가 있을 수 있다. 따라서 추후에 오디오 분류 모델을 보완하는 것도 고려하면서 개발해야한다.</p> <p>② 단어 인식을 위해서는 인터넷 연결이 필요하다. 인터넷이 연결이 되지 않는 경우 단어 인식 기능을 활용할 수 없다는 점을 고려해야 한다.</p> <p>③ 단어 인식을 위해서는 DB에 저장된 단어들을 조회할 수 있어야 한다. DB를 조회할 수 없는 경우가 있을 수도 있다는 점을 고려해야 한다.</p>
외관성	<p>① 사용자의 편의성을 고려한 디자인 : 청각장애인들이 언제나 위험 상황을 감지하고 특정 단어를 인식할 수 있도록 사용하기 쉬운 깔끔한 디자인을 구상하고자 하였다. 한눈에 어플리케이션의 기능을 알 수 있어 사용자들은 편리하게 어플리케이션을 사용할 수 있다.</p> <p>② 캐치프레이즈 : "항상 당신이 안전할 수 있도록 당신의 귀가 되어드립니다"라는 문구를 앱의 시작화면에 두어 사용자의 주의를 끌고 어플리케이션의 인상을 결정지을 수 있도록 하였다.</p>

<p>윤리성</p>	<p>블루투스 기기 검색을 위해서 위치 정보 액세스 권한 요청을 위해 ACCESS_FINE_LOCATION과ACCESS_COARSE_LOCATION의 권한 설정이 필요하다. 블루투스 액티비티에서 해당 권한을 요청을 하게 되는데 사용자가 요청을 수락한 후 위치 정보에 액세스할 수 있게 된다. 이 때, 수집된 위치 정보를 악의적으로 사용하지 않도록 해야 한다.</p> <p>또한, 위험 소리 감지, 단어 인식에 있어서도 마이크 입력을 받기 위한 RECORD_AUDIO 권한이 필요한데 이때 입력된 마이크 입력이 악의적으로 사용되지 않도록 한다.</p>
<p>사회적 영향</p>	<p>개발된 청각 장애인을 위한 소리 감지 IoT 웨어러블 디바이스 및 애플리케이션을 통해, 청각장애인들이 위험한 소리에 대한 대응력을 높일 수 있다. 실제 청각장애인들의 재난 대응 욕구에 따른 연구에 따르면 ‘주변에 재난이 발생했음을 알려주는 장치가 설치되어 있는가?’ 라는 질문에 66.3%나 되는 청각장애인들이 그렇지 않다고 답변했다. 또한, 일상생활 속에서 자주 접할 수 있는 단어들을 등록하고 이를 인지할 수 있게 한다. 이로써 일상생활에서 자주 들을 수 있는 단어들에 대해 반응성을 높일 수 있고 이는 청각장애인들의 삶의 질 향상으로 이어질 수 있다.</p>

이번 프로젝트 수행을 통해 다음 항목에 대한 성취 여부를 자가 점검해 주세요. (4개 이상 해당되어야 함)

속성	컴퓨팅문제수준설명	해당 여부 (O/X 표기)
	심화된 컴퓨팅문제가 속성1(지식의 깊이)을 만족하고, 속성2 ~ 속성8 중 일부 또는 전부를 만족해야 한다.	
속성1 (지식의 깊이)	최문제의 신 정보와 관련 연구 결과를 활용하고 있다.	O
속성2 (상충되는 요건의 범위)	상충될 수 있는 기술적 또는 컴퓨팅적 이슈를 다루고 있다.	O
속성3 (분석의 깊이)	해답이 명확하지 않은 문제를 해결하기 위해 깊이 있는 사고와 분석과정을 다루고 있다.	O
속성4 (생소한 주제)	자주 접하지 않는 컴퓨팅 문제를 다루고 있다.	O
속성5 (문제의 범위)	전공분야의 일반적인 실무 영역을 벗어난 범위를 다루고 있다.	O
속성6 (이해당사자의 요구 수준 및 범위)	다양한 이해당사자들의 요구사항들을 고려하고 있다.	O
속성7 (상호의존성)	상호 의존적인 여러 세부문제들이 결합된 종합적인 문제로 구성되어 있다.	O
속성8 (다양한 영향 고려)	다양한 분야에 미치는 영향을 고려하고 있다.	O

(1) 멘토 정보







- o 멘토명: 김민희
- o 소속기관명: (주)신라시스템
- o 직위(혹은 직급): 부사장

(2) 멘토링 내역 (주요 사항 있는 날짜로 10회 정도 기입)

연 번	날짜	멘토링 방식	멘토링 내용 (세부적으로/결론 위주로 작성)
1	2022.03.08	온라인	<ul style="list-style-type: none">• 기업 멘토님과 함께 주제에 대한 멘토링 진행• 햅틱 디바이스가 어느 정도까지 구현 가능한지 논의• 햅틱 디바이스를 활용한 주제에 대한 논의 진행
2	2022.03.10	온라인	<ul style="list-style-type: none">• 햅틱 디바이스를 활용한 수화 장갑을 개발하기로 결정• 가속도 센서, 등 제작에 있어서 필요한 재료들을 위해 멘토링 진행.
3	2022.04.01	오프라인 (학교)	<ul style="list-style-type: none">• 햅틱 디바이스 및 하드웨어를 활용한 주제에 대한 전체적인 회의를 진행함.• 수화 장갑 제작을 직접 시도하였으나 구현 난이도가 너무 높고, 완성도 측면과 성능적인 측면에서 별로 좋지 못할 것이라 판단.• 수화를 잘 모르는 상황에서 수화 관련 프로젝트를 진행하는 것은 어려움이 있을 것이라 판단.• 현실성을 고려한 프로젝트 및 하드웨어를 사용한 프로젝트에 대한 여러 가지 조언을 얻음.
4	2022.04.13	온라인	<ul style="list-style-type: none">• 주제를 청각장애인을 위한 소리 감지 IoT 웨어러블 디바이스로 정하여 관련 회의를 진행.• 디바이스 제작에 있어서 필요한 재료 구매 및 개발 분야 분담을 위해 멘토링 진행.
5	2022.04.18	온라인	<ul style="list-style-type: none">• 위험 소리 인지를 위한 개발 방법에 대한 회의를 진행• 텐서플로우 및 구글 티처블 머신을 활용한 방법을 제안하여 개발에 도입하고자 함.
6	2022.04.21	온라인	<ul style="list-style-type: none">• 단어 인식 구현에 대한 멘토링을 진행.• 여러 Speech to Text 기술들에 대해 조사 및 학습 후 의견을 나눔.
7	2022.04.25	온라인	<ul style="list-style-type: none">• 실제 디바이스 제작에 관해 멘토링 진행.• 블루투스 이어폰 형태로 만들기 위해서는 케이스를 만들어야 하는데 아두이노를 활용한 형태로는 어렵다는 조언으로 하드웨어 제작 방식을 변경함.
8	2022.05.10	온라인	<ul style="list-style-type: none">• 논문 작성 계획에 대한 멘토링을 진행.• 논문 출원 학회를 한국디지털콘텐츠학회로 결정.• 단어 인식 구현에 대한 피드백- 무제한으로 마이크 입력을 Speech to Text 해주는 것은

			<p>무리가 있다고 판단함.</p> <ul style="list-style-type: none"> - Android 라이브러리인 Speech Recognizer를 활용하여 구현하기로 결정.
9	2022.05.16	온라인	<ul style="list-style-type: none"> • 디바이스와 애플리케이션 간의 블루투스 통신을 위한 회의를 진행함. • Application, Service를 활용한 방법을 조사하고 학습한 것을 바탕으로 의견을 나눔.
10	2022.05.18	온라인	<ul style="list-style-type: none"> • 디바이스를 하나로 할지 두 개로 할지에 대해 논의하고 최종 하드웨어 제작에 도입함.
11	2022.05.23	온라인	<ul style="list-style-type: none"> • 오프라인 면담을 위한 일정에 대해 논의. • 현재 최종 개발 상황 및 논문 진행 상황에 대해 보고.
12	2022.05.24	오프라인 (학교)	<ul style="list-style-type: none"> • 논문 최종본 제출을 위한 멘토링을 진행. - 공동저자: 신현지, 엄다연, 이수경, 임지혁 • 더하여, 거의 완성된 애플리케이션과 하드웨어 제작에 대해 더욱이 추가되었으면 하는 점들에 대한 의견을 나눔.

(3) 멘토링 증빙 자료 (관련 사진): 4~6개 정도

<p>2022.03.10</p> 	<p>2022.04.01</p> 
<p>2022.04.13</p> 	<p>2022.04.25</p> 
<p>2022.05.16</p> 	<p>2022.05.24</p> 

(4) 멘토링 소감(4~6줄)

한 학기 동안 신라시스템의 김민희 부사장님, 이진후 연구원님과 함께 멘토링을 진행하면서 팀워크 및 개방적 소통 능력을 기르고, 기업과 개발 협업으로 실무 역량을 강화할 수 있었습니다. 주제를 선정하는 과정에서 어려움이 많았는데 프로젝트 주제를 선정하는데 있어 소비대상, 기술, 현재 존재하는 서비스 등 다양한 부분을 고려하여 결정할 것을 조언해주셨고 많은 도움이 되었습니다. 멘토님과의 교류로 이전 보다 넓은 시야를 가지고 개발에 임할 수 있었으며 이러한 성장은 현업에 계신 멘토님의 도움이 없었으면 어려웠으리라 생각합니다.