

청각장애인을 위한 안경

BF Glasses

식스센스조

목차

1. 참여학생 역할 분담표

2. 작품 개요 및 필요성

3. 작품의 개발 방법 및 과정

4. 기대 효과 및 활용 방안

5. 작품 및 제작활동 사진

참여학생 역할 분담표

참여학생 역할 분담표

김세진	김종휘	박별이	백종수	한은진	박윤석
음성, 소리인식 AI 개발자	DB, 백엔드 개발자	시스템 음성, 소리 개발자	웹 개발자	앱 개발자	시스템 SOS 기능 개발자

작품 개요 및 필요성

작품 개요 및 필요성

사람이 소리를 듣는 능력



외부의 상황을 즉각적으로 인지

청력



작품 개요 및 필요성

구분	2017년			
	등록 장애인수	추정수	등록율	
계	2,511,051	2,668,411	94.1	
주된 장애 유형별 장애인수	지체장애	1,267,174	1,278,368	99.1
	뇌병변장애	250,456	306,855	81.6
	시각장애	252,794	266,823	94.7
	청각장애	271,843	288,251	94.3
	언어장애	19,409	21,087	92.0
	지적장애	195,283	223,228	87.5
	자폐성장애	22,853	23,961	95.4
	정신장애	100,069	116,079	86.2
	신장장애	78,750	83,906	93.9
	심장장애	5,507	6,217	88.6
	호흡기장애	11,831	14,111	83.8
	간장애	11,042	11,045	100.0
	안면장애	2,680	3,180	84.3
	장루·요루장애	14,404	15,707	91.7
	뇌전증장애	6,956	9,592	72.5

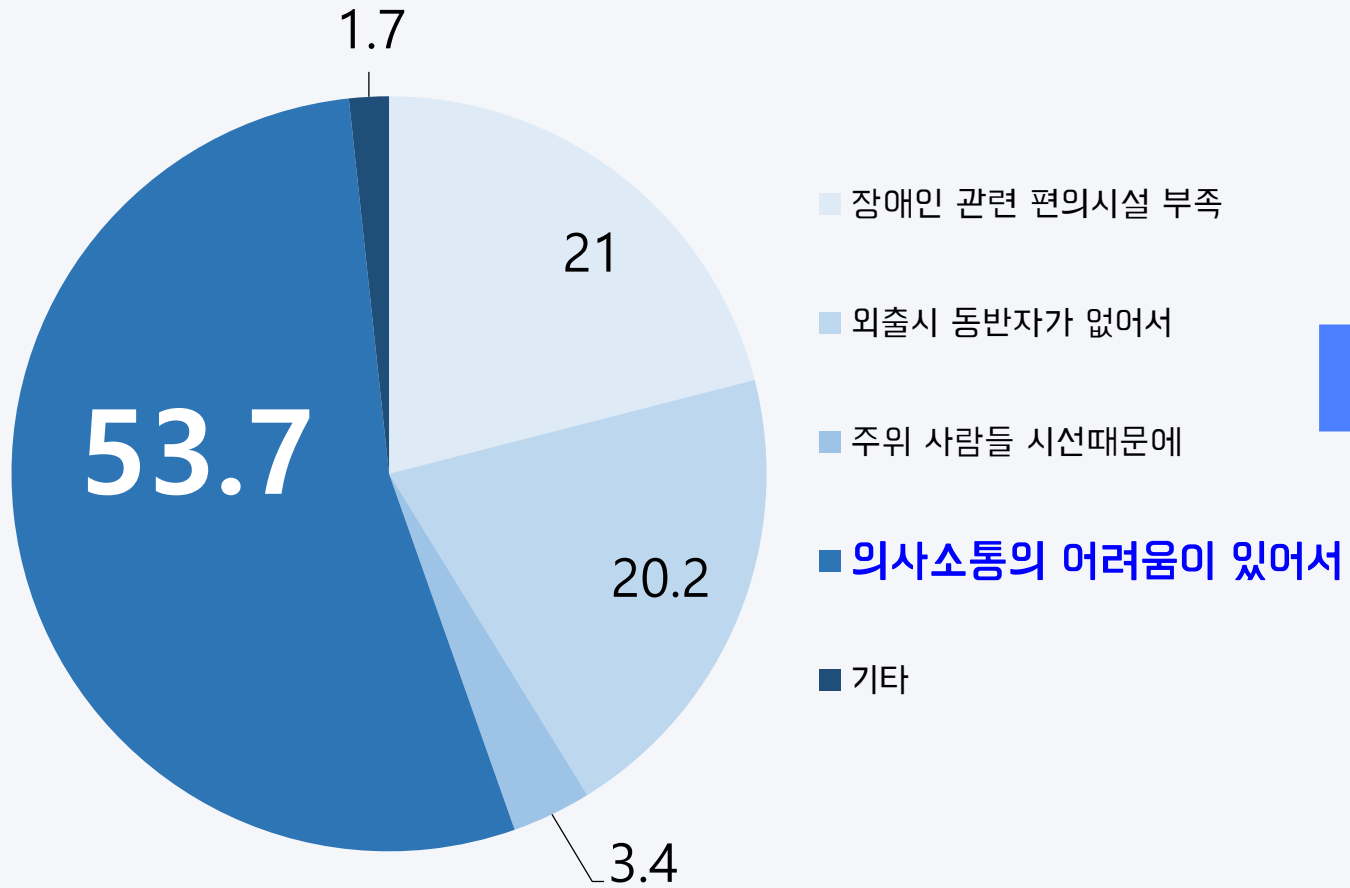


대한민국 인구수
190명 중 1명

[자료 1] 2017년 보건복지부 장애인 실태조사

작품 개요 및 필요성

청각장애인이 집 밖 활동 시 불편 이유



[자료 2] 2017년 보건복지부 장애인 실태조사

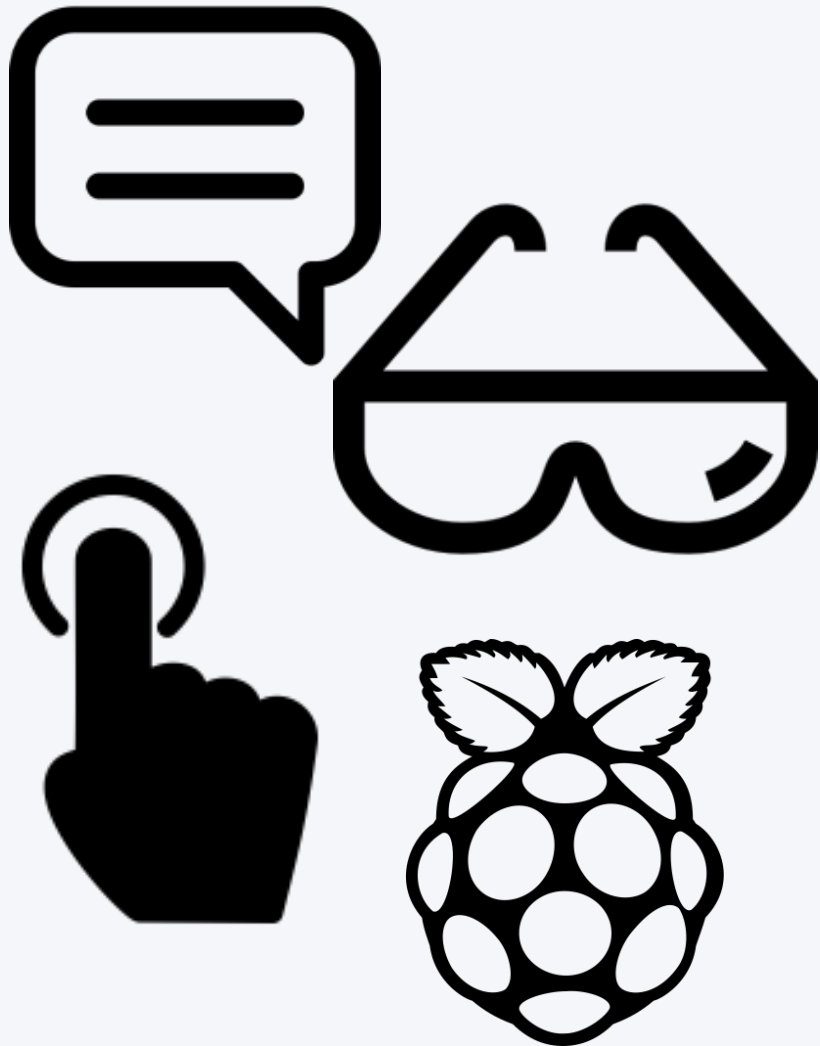


STT
(Speech-to-Text)



작품의 개발 방법 및 과정

작품의 개발 방법 및 과정 - 시스템



1. 음성, 소리를 텍스트로 출력

라즈베리파이의 **USB 마이크**를 통해 녹음한 음성, 소리를 시로 텍스트로 변환
-> **micro OLED**로 사용자의 안경에 나타낼 수 있도록 개발

1) STT 버튼 비 활성화

-> **위험 소리 인식 모드**로 사용이 되어 임의로 지정한 소리 및
사용자 지정 소리가 입력되는 경우 출력

2) STT 버튼 활성화

-> **상대방과 의사소통**을 위한 말을 텍스트 변환 기능이 적용

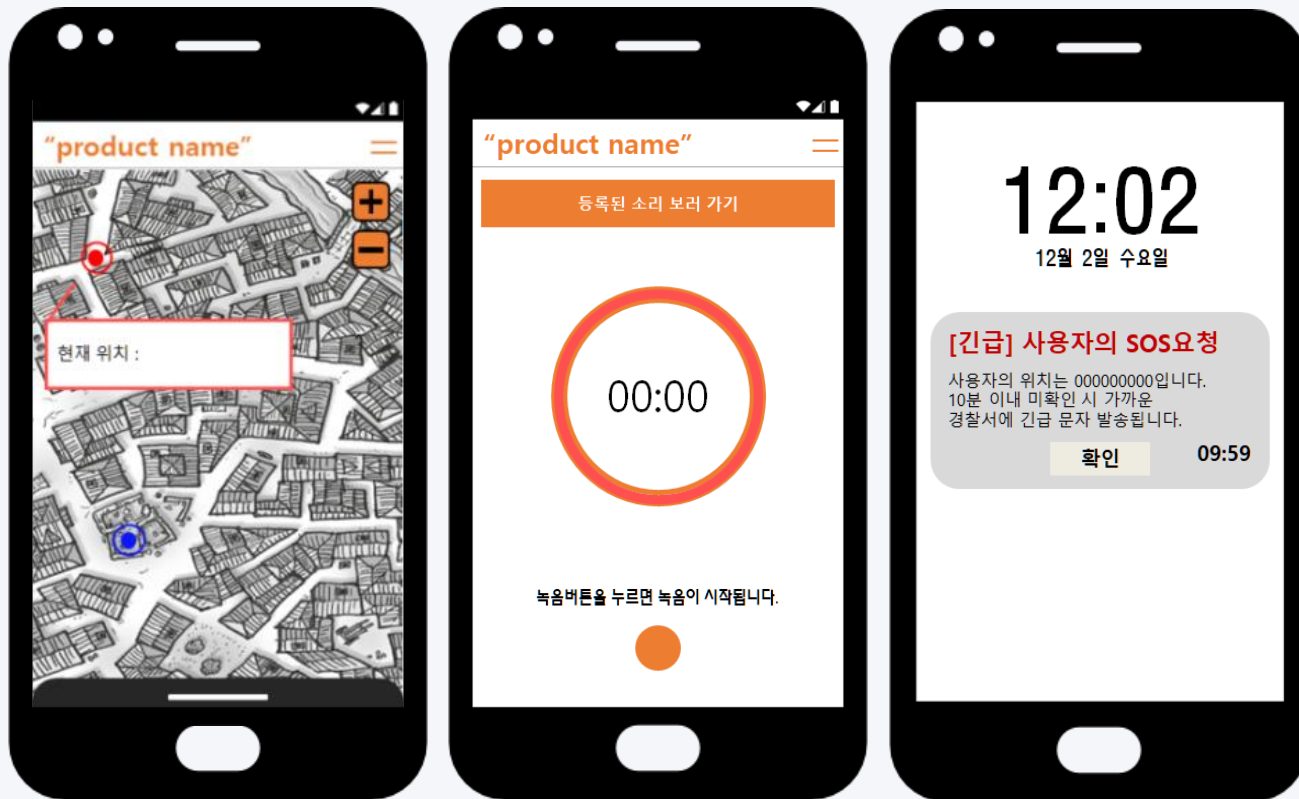
2. SOS 버튼 기능

사용자가 느끼기에 위험한 상황에 처했을 경우 **SOS버튼**으로 요청
라즈베리파이에 있는 마이크를 이용하여 얻은 소리 전송

-> 블루투스로 앱에 전달하여 보호자에게 소리 파일 전송 및 도움 요청

3. 전원 충전 및 배터리 잔여량 확인

작품의 개발 방법 및 과정 - 앱



1. 실시간 위치확인

스마트폰 GPS 기능을 이용해 사용자와 보호자의 위치를 실시간으로 공유 / 모니터링

2. 소리 등록 기능

지정된 소리 뿐 아니라 보호자가 직접 녹음해 다양한 소리를 등록하여 단어로 저장

-> 실제 상황에 해당 소리를 인식할 경우 단어 출력 가능

3. SOS 알림 기능

등록된 안경에서 받은 SOS 신호로 위험한 상황임을 보호자에게 앱 알림 메시지로 전달

-> 일정 시간이 지나면 인근 경찰서 알려 사용자 보호

작품의 개발 방법 및 과정 - 웹



1. 전체적인 페이지 구현

가시성과 가독성을 높인 페이지로 **개발자와 소비자간의 소통**을 위한 '제품 소식 페이지' 및 '고객 지원 페이지'를 DB와 서버 연결

2. 실시간 위치확인

안전을 위해 사용자, 보호자의 위치를 확인할 수 있도록 앱에서 얻은 **GPS 데이터**를 이용해 웹 지도 API에 출력

3. [관리자] 명령전송

관리자 로그인 시 명령 전송 텍스트 박스 입력으로 DB에 저장되어 관리자 계정에 등록된 **모든 안경 기기**로 **텍스트 형태의 명령 전송**

작품의 개발 방법 및 과정 - 음성, 소리인식 AI



Google Cloud
Speech API

1. 실시간 음성 인식

Python Open API (Speech Recognition, PyAudio)를 사용

- > PyAudio를 통해 음성 데이터를 입력 받아
Speech Recognition를 활용하여 음성을 인식
- > 사용 언어에 따른 설정값을 수정하여 **언어에 대한 제한X**
- > 주변 소음 수준에 대한 임계값을 보정O => **장소에 대한 제약X**

2. 머신러닝기반 소리(상황) 분석

위험소리를 분류하기 위해 Google에서 개발한
오픈소스 Library **Tensorflow**를 활용하여 **위험소리를 실시간으로 분석**



작품의 개발 방법 및 과정 - DB, 백엔드



회원 정보	MEMBER 테이블
위험 소리 정보	DANGERSOUND 테이블
사용자 소리 정보	MEMBERSOUND 테이블
질의응답 정보	Q&A 테이블
회사 소개 정보	INTRODUCE 테이블
SOS 메시지 정보	SOSMESSAE 테이블

1. 상황에 맞는 데이터 추출

백엔드는 **PHP**를 주로 사용하여 **웹과 앱**에서 얻은 정보들
-> 서버를 통해 데이터베이스로 전송 저장 및 전달

- 위험한 소리 (경적, 사이렌 등), 그에 맞는 텍스트 DB에 저장
- 사용자가 녹음한 소리나 위험 소리의 정보 DB에 저장
- 사용자, 보호자, 관리자에 대한 회원정보가 분류 저장 및 관리
- 게시판 정보 저장 및 관리

음성 데이터는 크기가 크기에 서버에 저장하고 링크를 통해 출력

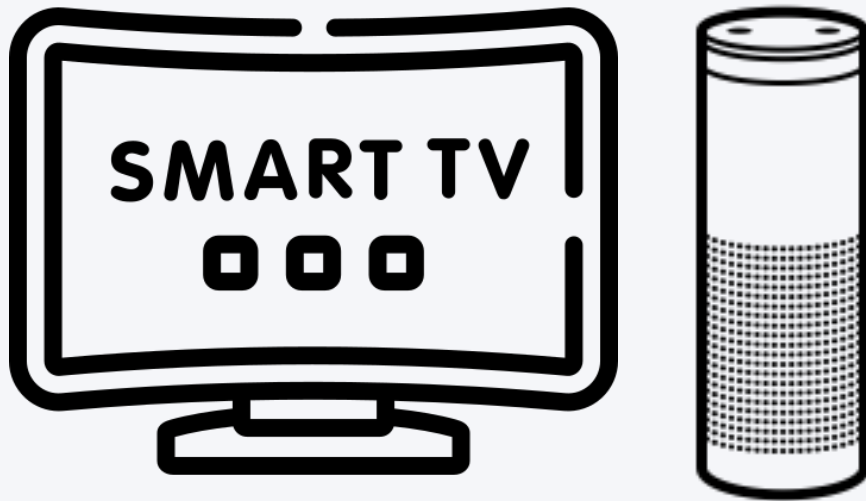
2. 보호자에게 정보 전송

시스템에서 얻은 **버튼 정보 및 음성 데이터**는 블루투스와 연결된 앱으로 서버를 통해 데이터베이스에 저장 및 서버를 호출하여 필요한 정보를 전달

기대 효과 및 활용 방안

기대 효과 및 활용 방안

실시간 음성인식과 화자 인식 기술



장애인을 위한 아이디어 상품



기대 효과 및 활용 방안



1. 의사소통의 어려움을 해결
2. 긴급한 상황에서도 빠르게 인지
->사용자의 안전 확보

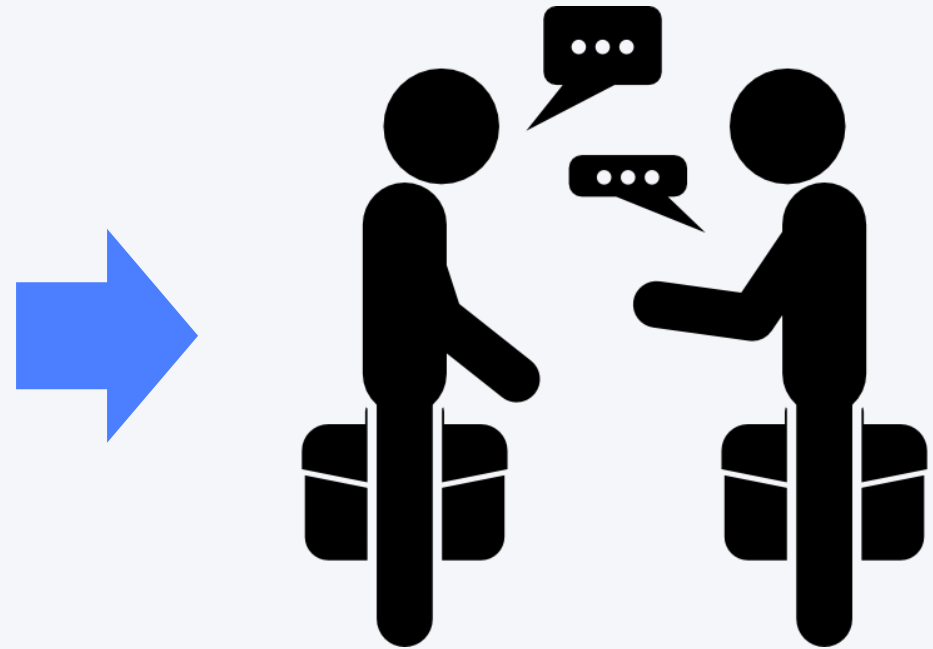
기대 효과 및 활용 방안

조용한 택시



차량 내외부 모든 소리 정보

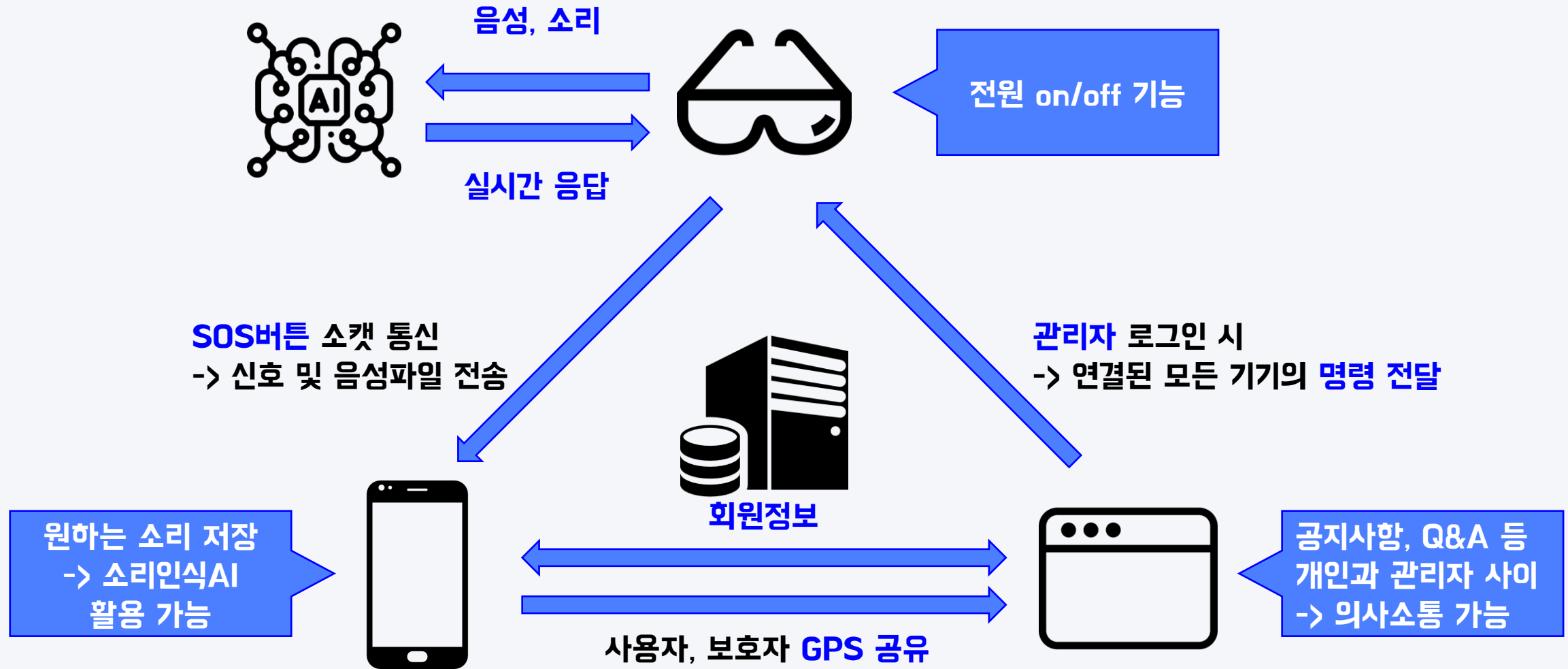
-> **시각, 촉각화**하여 청각장애인 운전자의 안전성 향상



청각 장애인의 **새로운 일자리 창출**

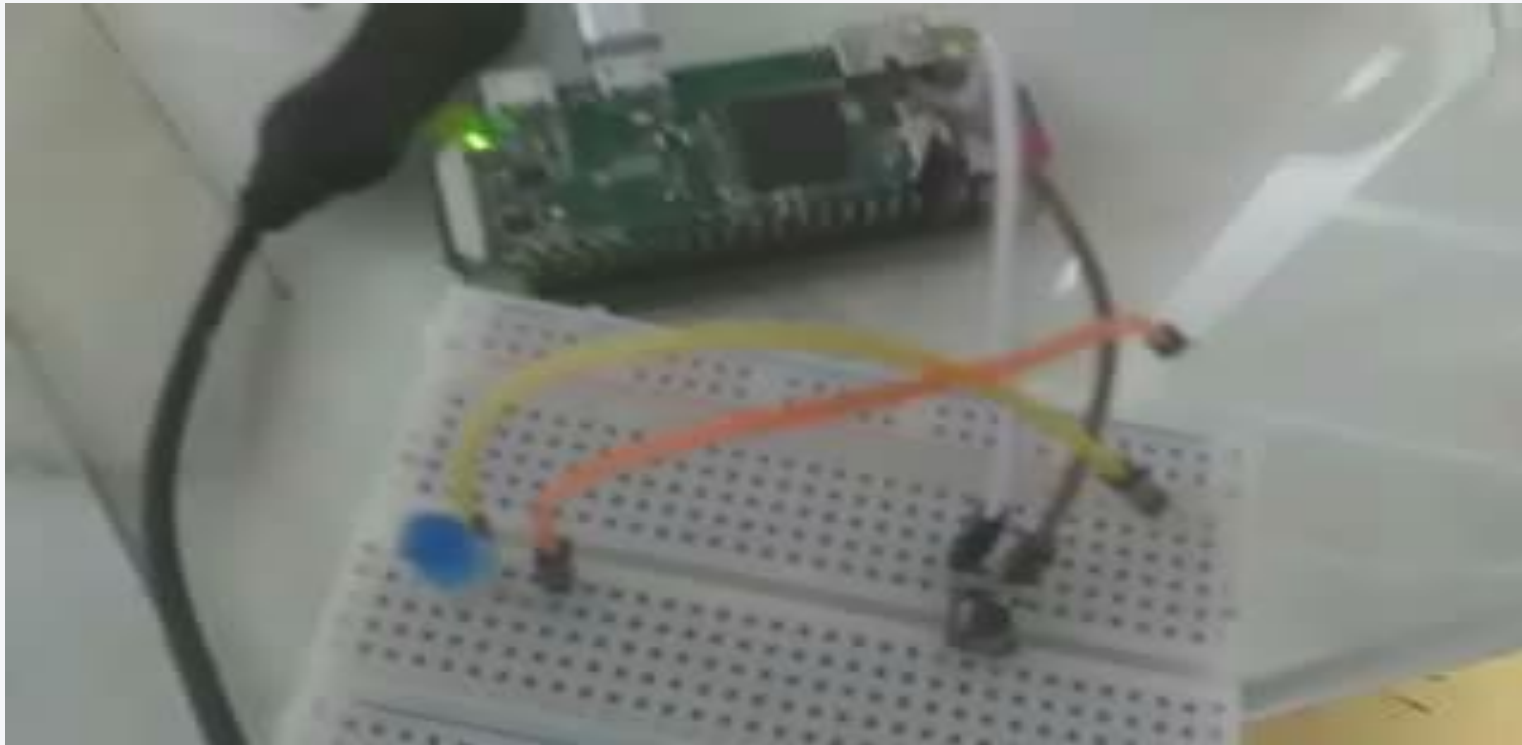
작품 및 제작활동 사진

작품 구조도



작품 및 제작활동 사진

on/off 기능 구현



작품 및 제작활동 사진

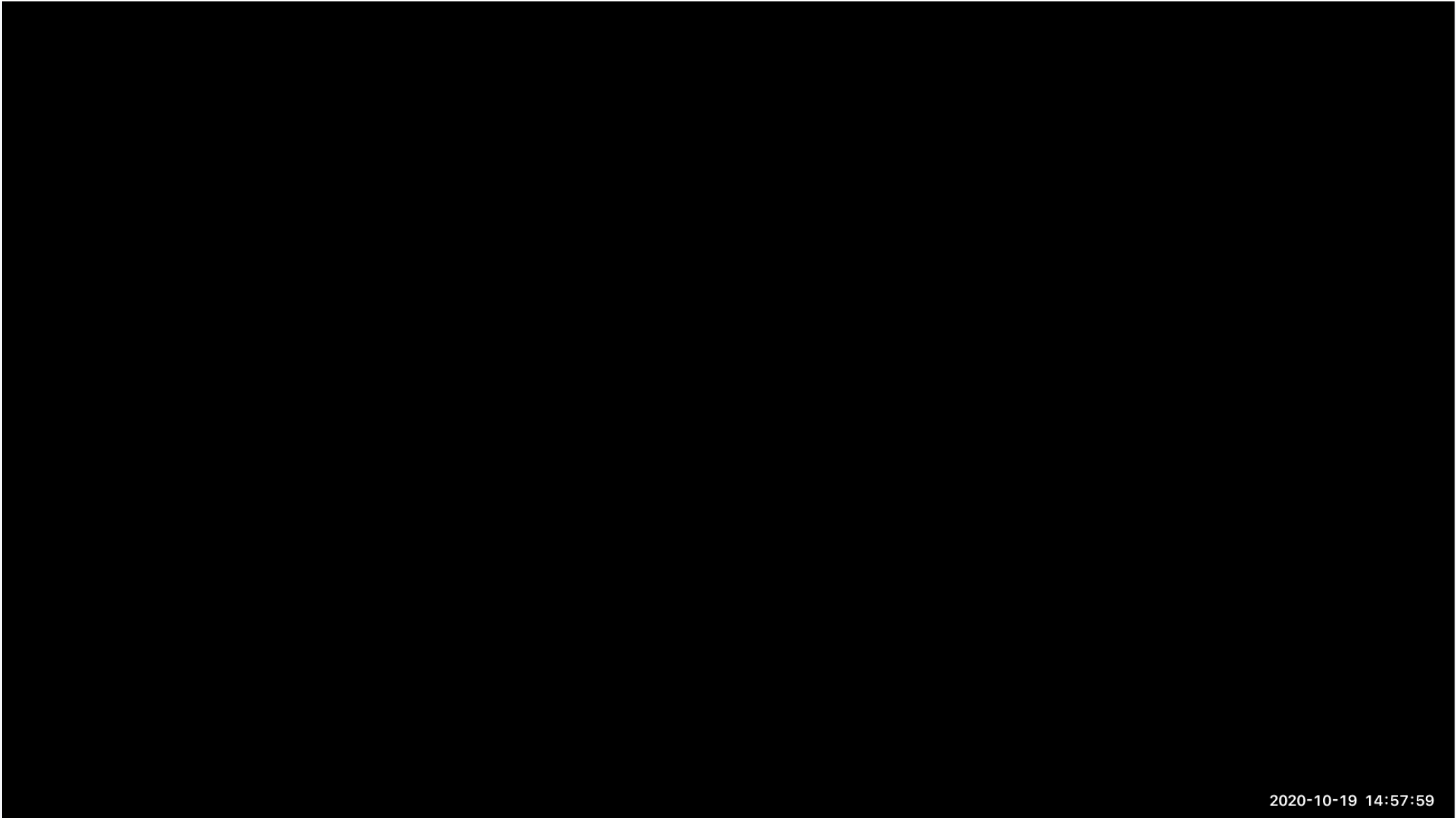
음성인식 AI 구현

```
raise self
grpc._channel._MultiThreadedRendezvous: <_MultiThreadedRendezvous of R
    status = StatusCode.OUT_OF_RANGE
    details = "Exceeded maximum allowed stream duration of 305 sec
    debug_error_string = "{\"created\":\"@1600701496.369702390\",\"desc
.163.234:443\",\"file\":\"src/core/lib/surface/call.cc\",\"file_line\":1062,\"
duration of 305 seconds.\",\"grpc_status\":11}"
>

The above exception was the direct cause of the following exception:

Traceback (most recent call last):
  File "transcribe_streaming_mic.py", line 185, in <module>
    main()
  File "transcribe_streaming_mic.py", line 181, in main
    listen_print_loop(responses)
  File "transcribe_streaming_mic.py", line 120, in listen_print_loop
    for response in responses:
  File "/home/pi/env/lib/python3.7/site-packages/google/api_core/grpc_
six.raise_from(exceptions.from_grpc_error(exc), exc)
  File "<string>", line 3, in raise_from
google.api_core.exceptions.OutOfRange: 400 Exceeded maximum allowed st
(env) pi@raspberrypi:~/python-docs-samples/speech/microphone $ ^C
(env) pi@raspberrypi:~/python-docs-samples/speech/microphone $ python
_streaming_mic.py
python: can't open file 'transcribe': [Errno 2] No such file or direct
(env) pi@raspberrypi:~/python-docs-samples/speech/microphone $ python
```





작품 및 제작활동 사진

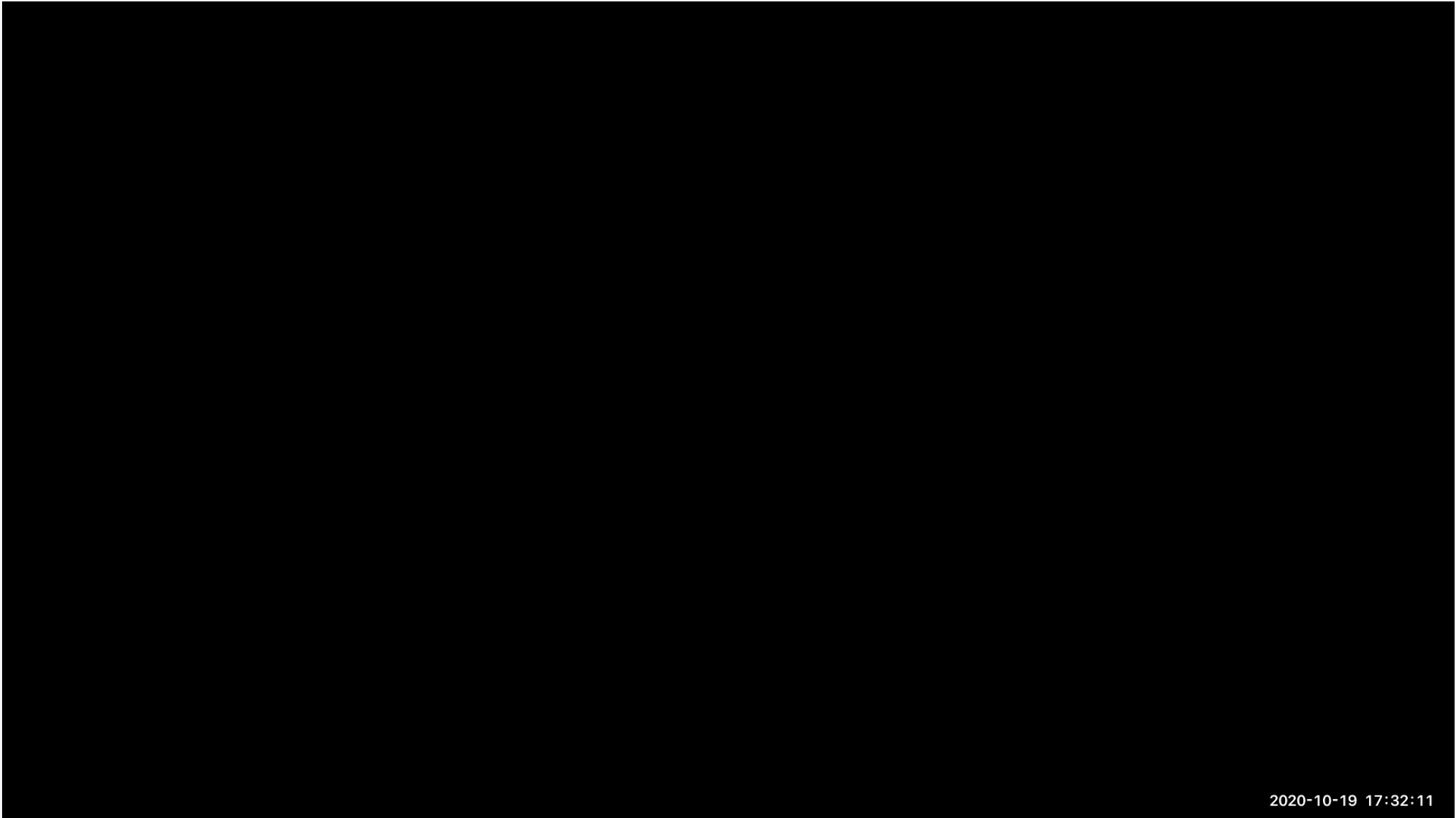
내 정보(member_info)를 효율적인 코드로 변경
미 개발 페이지 였던 개인 버전을 모두 개발

작품 및 제작활동 사진

삭제(delete)와 삭제 오류 방지(redirect) 개발
글 작성(write)에서 시간을 자동 삽입되게 변경

작품 및 제작활동 사진

파이어 베이스를 이용한 실시간 기능
-> 웹과 앱과 DB가 리얼 타임으로 연결

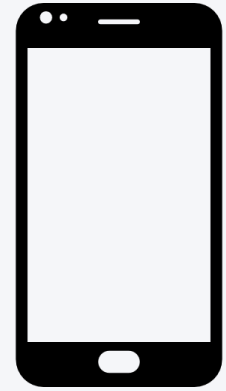


작품 및 제작활동 사진



서버

시스템 <-> 앱



클라이언트

SOS 버튼



1 버튼 ON



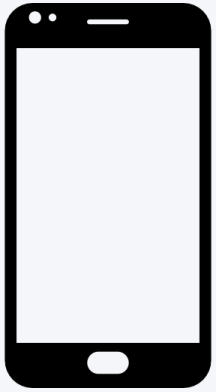
보호자에게 알림

주변 소리 10초 녹음

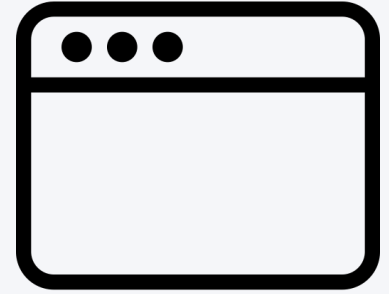


위험 상황 알림

작품 및 제작활동 사진



앱 <-> 웹



휴대폰 GPS 사용
-> 앱 확인 가능

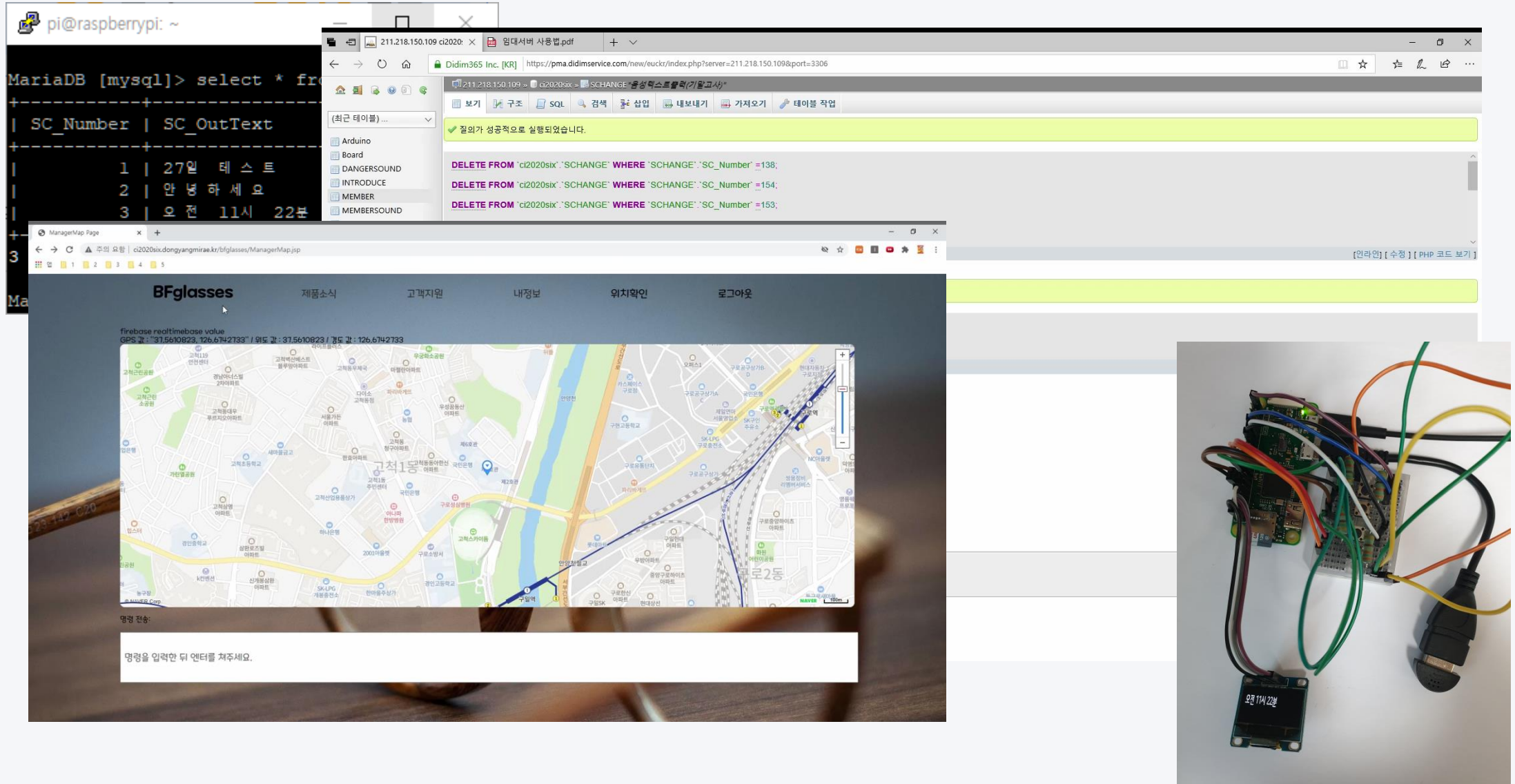


웹 확인 가능



Firebase

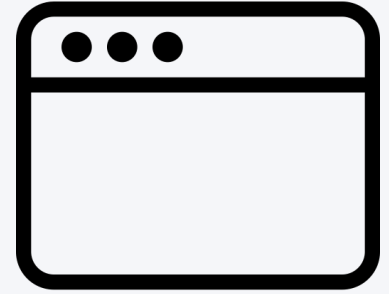
작품 및 제작활동 사진



작품 및 제작활동 사진



시스템 <-> 웹



연결된 모든 기기
명령 전달

관리자 로그인
명령 전송

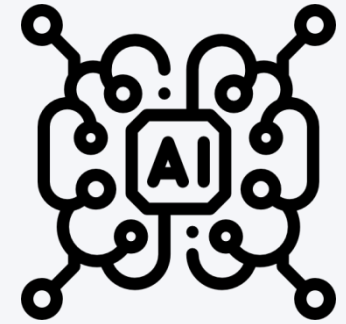


Apache Tomcat

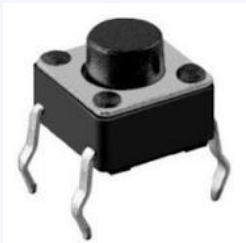
작품 및 제작활동 사진



시스템 <-> AI



STT 버튼

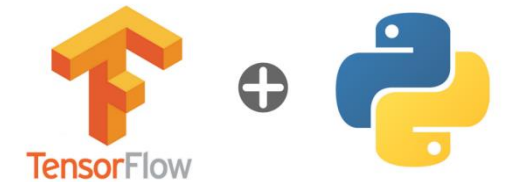


1 버튼 ON



소리 인식 AI

2 버튼 OFF



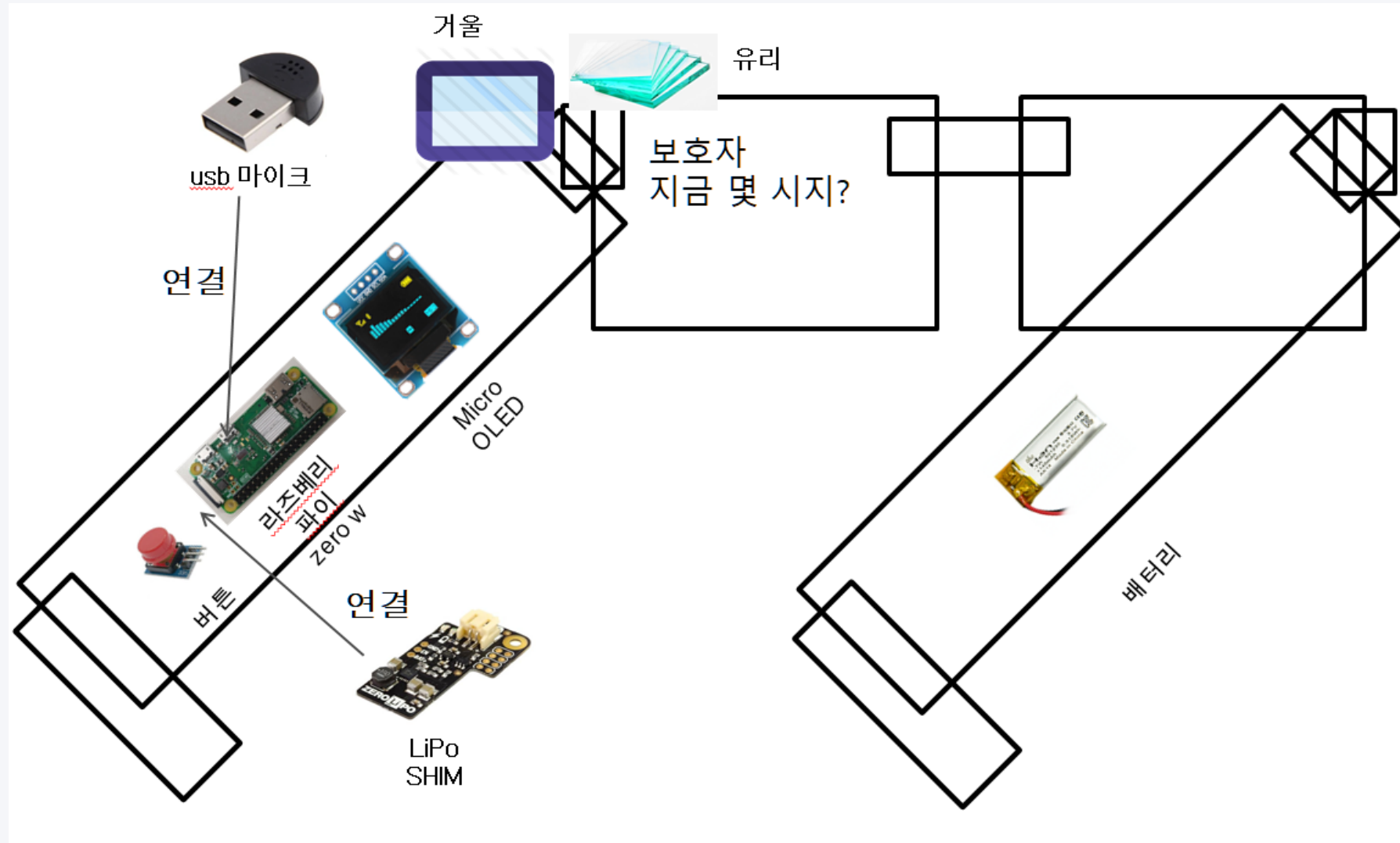
BF Glasses - 식스센스

감사합니다.

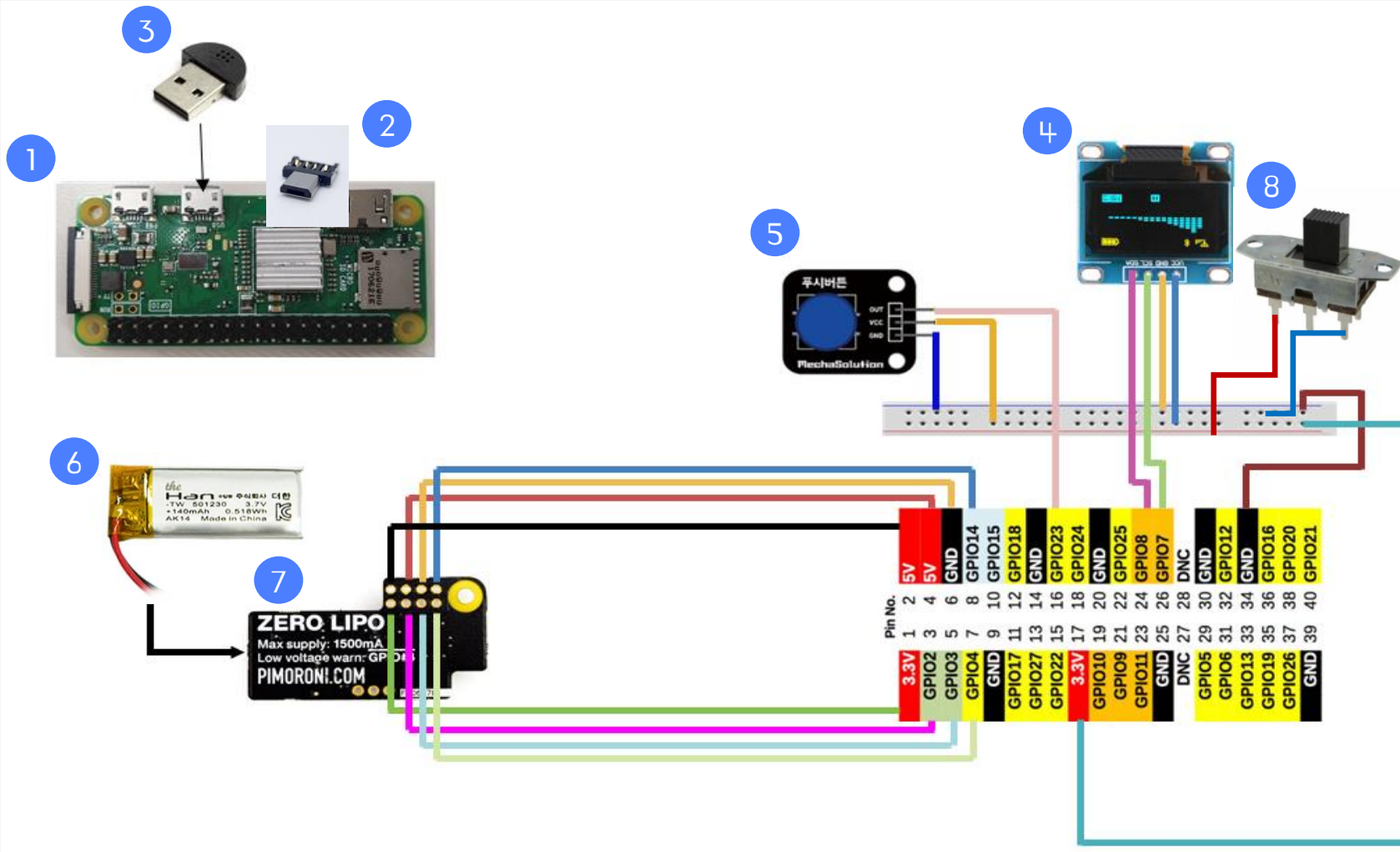
Chapter 5

작품 구조도

작품 구조도 - 시스템 배치도



작품 구조도 - 시스템 회로도

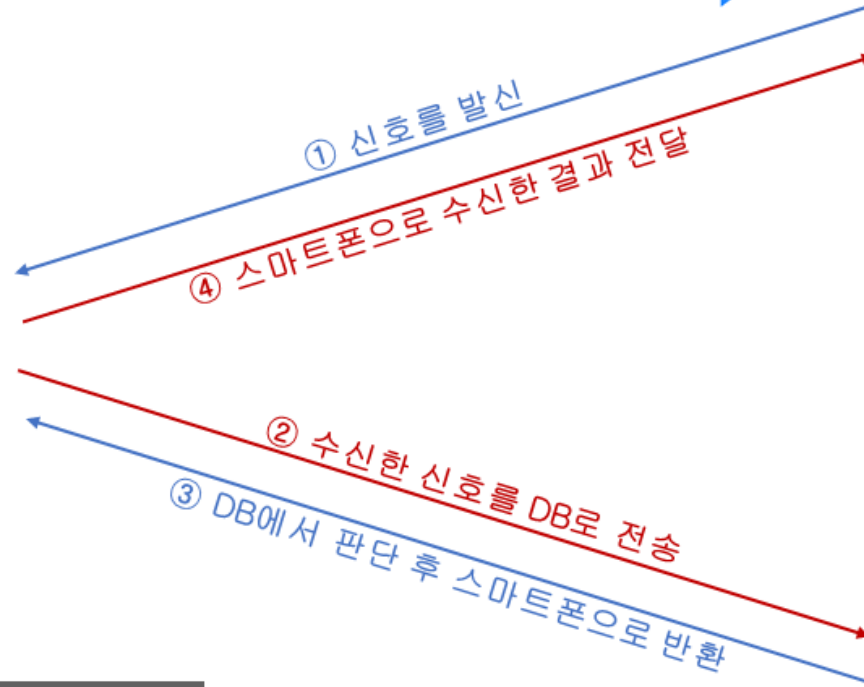


번호	기능명
1	라즈베리파이 ZERO W - 블루투스 모듈 내장(음성, 소리를 python으로 전송하고 텍스트를 받아와 출력 및 SOS 버튼 신호 전송)
2	USB A(M) to USB micro B(M) OTG 초경량 미니젠더 - micro USB단자에 일반 USB를 꽂을 수 있도록 바꾸는 선
3	초소형 USB 마이크 - USB에 연결해 음성, 소리 데이터 받아 블루투스로 전송
4	micro OLED(0.96인치) - 무슨 말을 하는지 텍스트로 출력 - 무슨 소리인지 텍스트로 출력
5	버튼(SOS 버튼) - 누르면 블루투스를 이용해 앱으로 신호를 보내 SOS 기능 실행
6	Lipo 배터리 - 소형 배터리로 충전하여 사용 가능
7	Lipo 배터리 연결 부품 - 라즈베리파이 ZERO W와 Lipo 배터리를 연결하는 부품으로 반드시 필요
8	슬라이드 스위치 - 전원 ON/OFF 기능

작품 구조도 - 앱 구성도



시스템과 블루투스를 통해 연동되는 앱



SmartWearable



Database

작품 구조도 - 웹 구성도

