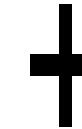


# <Smart Shoes Care>

TEAM MASEOK



201810876 문성준  
201910903 오세욱



-팀장 : 201810876 문성준

-노션을 활용한 일정 조정, 소스 코드 제어



Joon  
dessertgomjelly

Overview Repositories 5 Projects Packages Stars 11

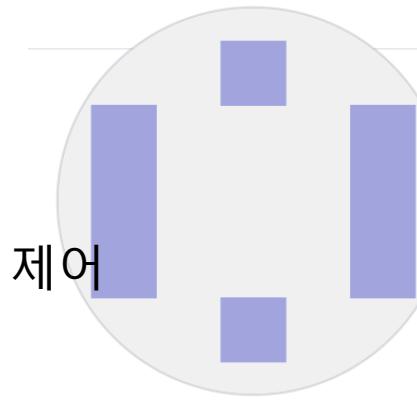
dessertgomjelly / README.md

@dessert\_gomjelly

Popular repositories

-팀원 : 201910903 오세욱

- 라즈베리파이, 회로 연결, 소스 코드 제어



OSEUK

Overview Repositories 6 Projects Packages Stars

OSEUK / README.md

Hi there 🙌

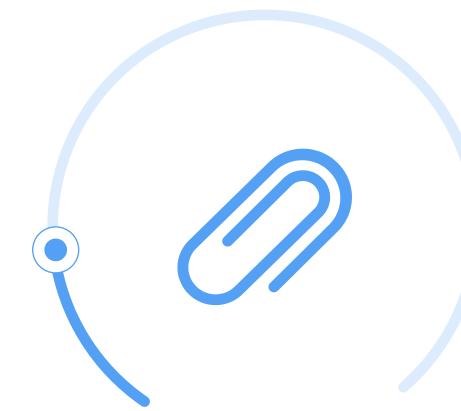
#### OSEUK's GitHub Stats

Total Stars Earned:	0
Total Commits (2023):	21
Total PRs:	0
Total Issues:	0
Contributed to (last year):	0

A+

#### Most Used Languages

Jupyter Notebook	79.24%
C++	3.65%
C	17.11%



협업 툴

## 1. SLACK을 활용한 소통

Maseok

- 채널
- # 랜덤
- # 일반
- # iot-sensor-control**
- 채널 추가
- 다이렉트 메시지
- 문성준 나
- 오세욱
- 직장 등록 추가
- 앱
- GitHub
- Notion
- 앱 추가

# iot-sensor-control

+ 책갈피 추가

문성준 오후 1:35 저희

프로젝트

진행할때

필요한 물품 정리해볼까요?

오세욱 오후 1:36 세우 네

언제 만날까요?

문성준 오후 1:36 세우 그거?

오세욱 오후 1:36 세우 그거?

**GitHub**  
Get updates from the world's leading development platform...

**Q, ChatGPT for Slack**  
Hi I'm Q, ChatGPT for Slack. Ask me anything by @Q

**Notion**  
The connected workspace for your docs, projects, and knowledge

## SLACK에 외부 API (CHAT GPT, GITHUB)연동

GitHub

메시지 정보

GitHub 님은 어떻게 작업하나요?

6월 2일 금요일

GitHub 오전 9:10 Welcome to the GitHub Slack app! To get started you have to sign into your GitHub account.

메시지 정보

문성준 6월 2일 금요일 자주 사용하는 채널( 6월 2일 금요일 )를 초대하세요.

문성준 9:22 hey

1개의 댓글 7일 전

오늘

문성준 오후 4:41 아두이노에 부저 센서를 추가하고 싶은데 어떤 부저 사용해야 할까

1개의 댓글 오늘 오후 4:41에

## 2. NOTION을 활용한 작업관리



### IOT센서를 활용한 SMART SHOE CASE

센서를 통해 신발이 항상 최적의 상태를 유지하도록 한다

iotsensor	OSULUK · 27일 전에 업데이트됨
-----------	-----------------------

ArduinoSensor	desertcompoly · 27일 전에 업데이트됨
---------------	------------------------------

캡린더 보기 | 표

#### IOT Project 진행 일정

Aa 이름	트래그	날짜
project 제안서 및 리플릿	완료	2023년 4월 21일
작업 환경 설정	진행 중	2023년 4월 28일
중간 보고서 제출	진행 예정	2023년 5월 3일
경진대회 시청	진행 예정	2023년 5월 5일

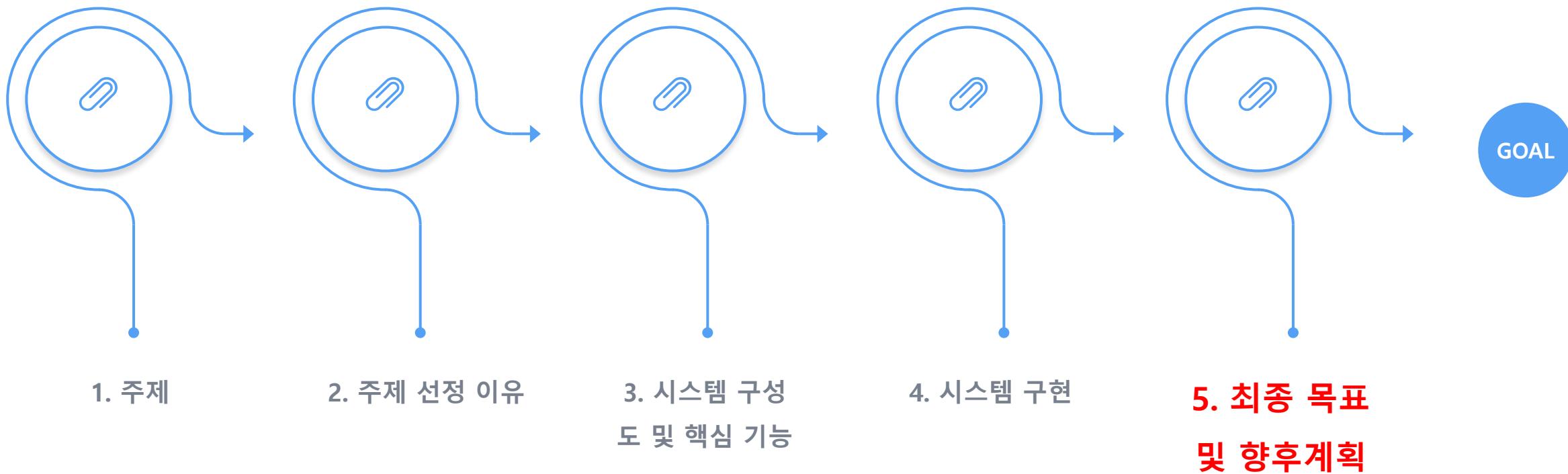
+ 새로 만들기

## 3. GITHUB 작업 내용 공유, 게시

iosensor

Public

C++





# 1. 주제



# 1. 주제 <Smart Shoe Care 2.0>



+



=



&lt;Smart Shoe Care&gt;

‘신발 관리’와 ‘리셀 시장’이라는 두가지 측면을 모두 만족 시킬 수 있는 <Smart Shoe Care> 를 주제로 선정.

## 1. 신발관리

온 습도를 실시간으로 측정하고, 모터를 이용해 온 습도를 조절하여 자동으로 신발을 관리 하게 한다.

LED를 활용하여 현재 상태를 파악하고 더 나아가 전시 용도로 사용할 수 있다.

## 2. 리셀

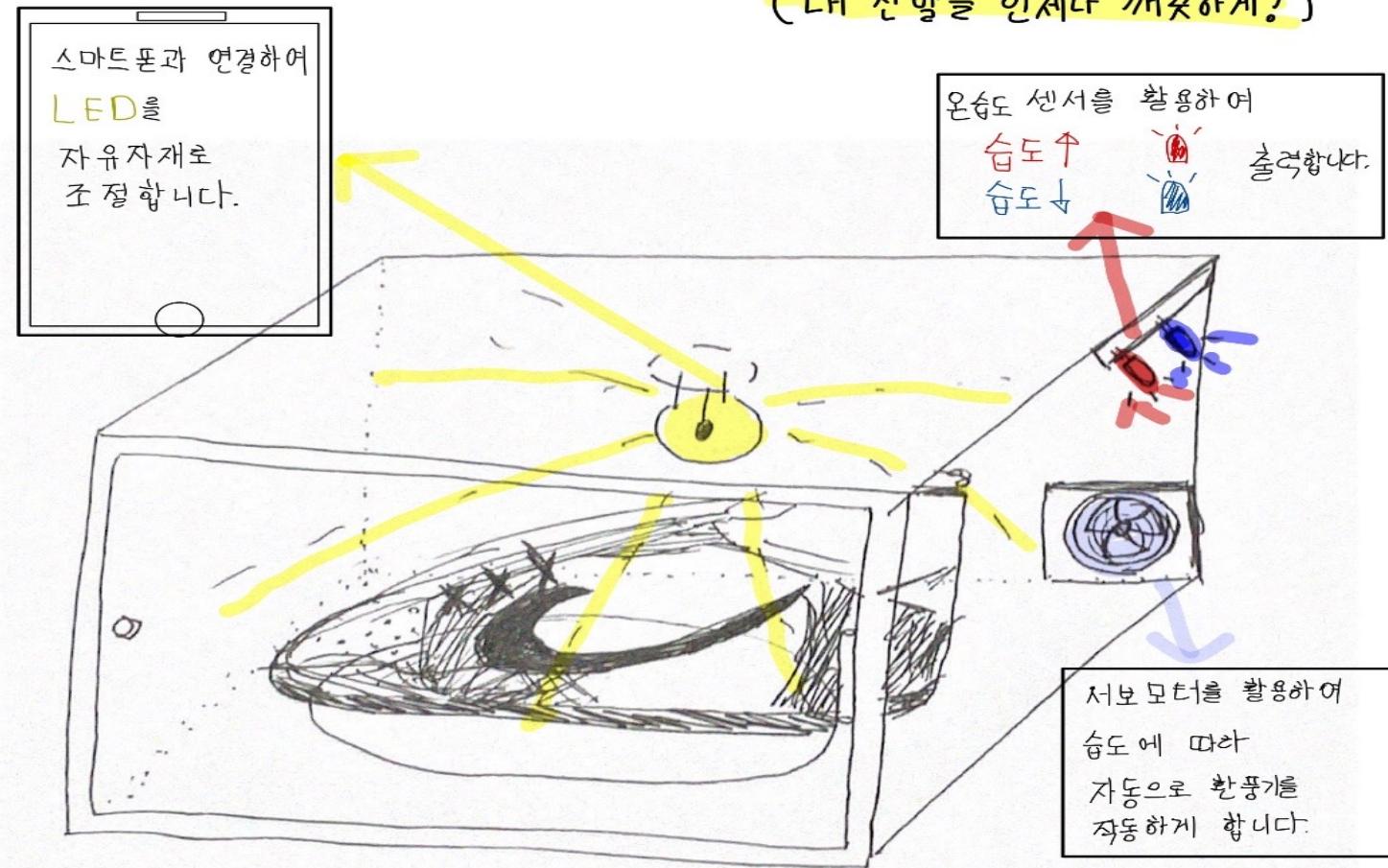
리셀 시 가장 중요한 착용시간에 대한 데이터를 수집하여 리셀(중고) 거래 시에 신뢰성 있는 자료로 활용할 수 있게 한다.

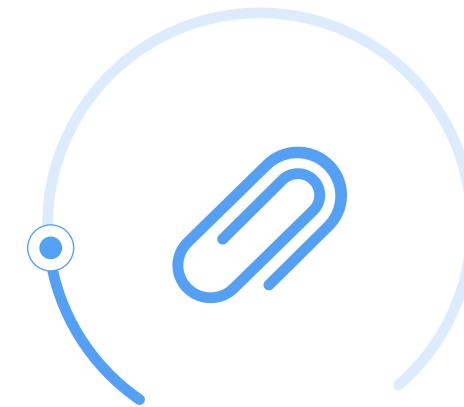


## 1. 리플릿 (Smart Shoe Care 1.0)

### <SMART SHOE CARE>

(내 신발을 언제나 깨끗하게!\*)





## 2. 주제 선정 이유

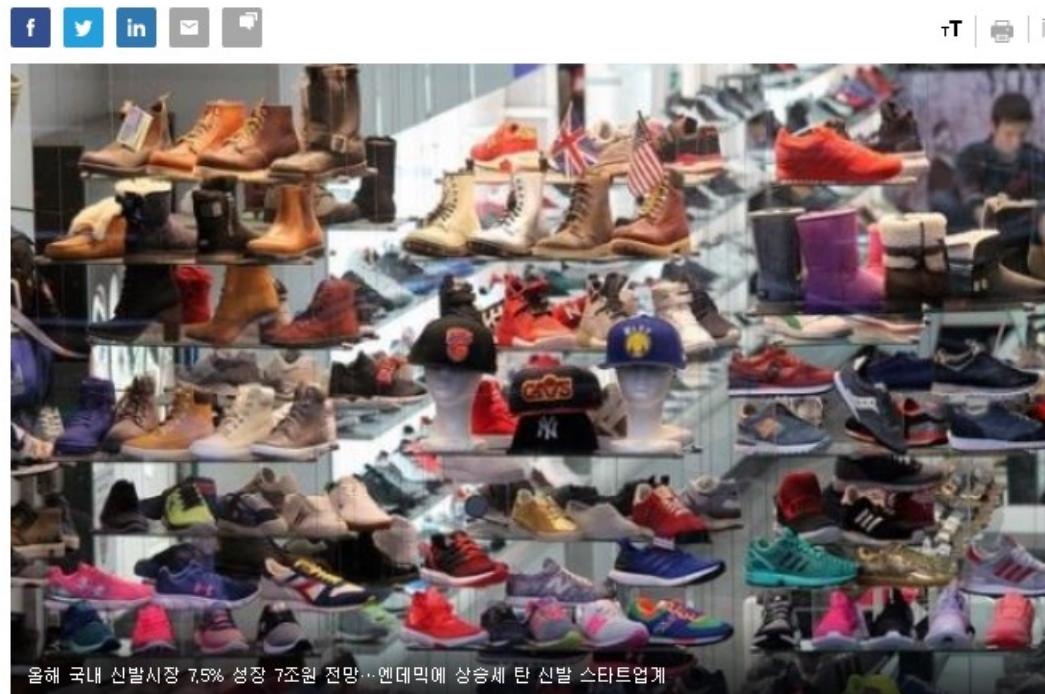
## 2. 주제 선정 이유 (신발 시장 규모)



올해 국내 신발시장 7.5% 성장 7조원 전망...엔데믹에 상승세 탄 신발 스타트업계

THE STOCK | IPO

입력: 2022년 06월 28일 03:31



올해 국내 신발시장 7.5% 성장 7조원 전망...엔데믹에 상승세 탄 신발 스타트업계

Coronavirus 19 Pandemic



Endemic

MZ세대가 신발을 중요한 패션아이템으로 여겨 지출

을 아끼지 않은 경향도 더욱 뚜렷해질 것으로 보인다.

출처: 퍽사베이

## 2. 주제 선정 이유 (리셀 시장 규모)



대세로 떠오른 중고 거래…리세일·리셀 트렌드

어디까지? ①

2022.11.22 중소벤처기업부



+

-



리셀 시장 규모 확대



명품의 한정판 등 물품을 사서 비싼 값에 되파는 ‘리셀(Resell)’ 또한 리세일의 한 유형으로서 최근 각광받는 트렌드다.

재테크 수단으로까지 성장한 전 세계 스니커즈(운동화) 리셀 시장 규모만 2019년 20억 달러에서 2025년에는 세배에 이르는 60억 달러(약 8조 5천억 원)로 증가하리란 전망이다.

**리셀(중고)** 시장의 확대에 있어서 가장 중요한 포인트는 신발의

관리와 착용 시간이다. 이에 따라, 스마트 슈케어를 통해 신발을

**관리**하고, 착용 **시간**에 대한 정보를 수집하여 중고 거래 시에

**신뢰성 있는 자료**로 활용할 수 있다.



### 3. 시스템 구성도 및 핵심기능

### 3. 시스템 구성도 및 핵심 기능



(Smart Shoe Care 2.0)

#### 핵심 기능



신발 보관

#### 조명을 이용한 상태 표현

LED를 이용하여 조명효과를 나타낼 수 있습니다. 현재 온습도에 따라 다른 조명을 나오도록 했습니다.



상태 관리

당근마켓 중고거래

#### 신발의 습도 조절

신발의 온도, 습도를 조절하여 신발 상태를 관리 할 수 있습니다.



착용 시간

당근마켓 중고거래

#### 신발의 착용 시간 저장

중고 거래 시 가장 중요한 POINT인 신발 착용 시간을 이를 이용하여 조금 더 공증된 자료로 활용 할 수 있습니다.

# 3. 시스템 구성도 및 핵심 기능



(Smart Shoe Care 2.0)

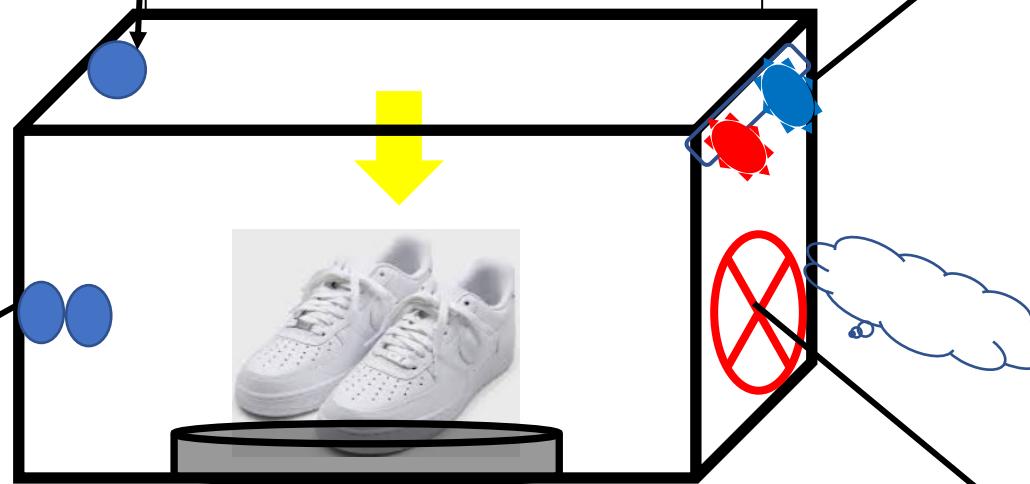
## 시스템 구성

### <앱 인벤터>

앱 인벤터를 활용하여 온 습도를 휴대폰으로 확인할 수 있도록 한다. 어떻게 조절할지 판단한다.

### <수동 부저>

수동 부저를 활용하여 신발 객체 탐지마다 도난 방지 시스템 알림 기능으로 활용한다.



### <초음파 센서>

초음파 센서를 이용해 거리를 측정한다.  
신발이 슈케이스 안에 있으면 시간이 흐르게 하고  
신발이 없으면 중지하는 방식으로 사용시간을 측정한다.  
APP inventor를 활용하여 사용시간과 누적시간에  
대한 데이터를 볼 수 있게 한다.

### <온 습도 센서 + Led>

온 습도 센서를 활용하여 습도가 높으면 빨간색 LED를 출력하고 습도가 낮으면 파란색 LED를 출력한다.

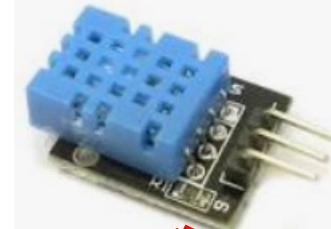
### <온 습도 센서 + 모터>

DHT 센서를 이용하여 온도와 습도 값을 측정한다.  
모터를 제어하여 적절한 온도, 습도에 따라 자동으로 작동시켜 환풍기로 활용한다.

### 3. 시스템 구성도 및 핵심 기능

#### 신발 관리 기능

##### <온 습도 센서 DHT 11>



온 습도 센서를 통해 슈케이스안의  
온습도를 측정

##### <블루투스 모듈>



블루투스와 연동해 휴대폰에서 현재 온습도를 알 수 있음

##### <서보 모터>



온도 > 25 || 습도 > 50

자동으로 신발의 적절한 보관  
상태를 유지

##### <LED>



온도 > 25 || 습도 > 50

온습도 상태에 따라 파란색  
빨간색으로 표현

#### 객체 탐지 기능

##### <초음파 센서>



초음파 센서를 통해 신발의  
유무를 정확히 파악

##### <시리얼 모니터>



##### <수동 부저>



신발의 도난방지를 위해  
신발 객체 탐지 시 부저로  
도난방지 음악이 나오도록함.

향후 로드셀과도 결합해 공중  
된 데이터셋을 만들 계획

# 3. 시스템 구성도 및 핵심 기능



(Smart Shoe Care 2.0)

## <코드 구성>

### 초기값 설정

```
#include <DHT.h>
#include <SoftwareSerial.h>
#include <Servo.h>

DHT mydht(A0, DHT11);           // A0 핀과 DHT11 센서를 사용하여 DHT 객체를
SoftwareSerial mySerial(2, 3);    // 소프트웨어 시리얼 통신을 위한 객체를 생성
Servo myServo;

int angle = 0;
int trigPin = 8;
int echoPin = 9;
bool shoeDetected = false;
unsigned long shoeDetectedTime = 0;
unsigned long totalElapsedTime = 0;
float detectionDistance = 10.0;

int redLedPin = 4;    // 빨간색 LED의 핀 번호
int blueLedPin = 5;   // 파란색 LED의 핀 번호
```

### 센서 및 핀모드 설정

```
void setup() {
    mySerial.begin(9600);
    Serial.begin(9600);
    mydht.begin();
    myServo.attach(12);
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
    pinMode(redLedPin, OUTPUT);
    pinMode(blueLedPin, OUTPUT);
}
```

### 3. 시스템 구성도 및 핵심 기능



(Smart Shoe Care 2.0)

#### <코드 구성>

#### 신발 관리 기능

온습도 센서 -> LED & 환풍기

```
void loop() {
    float temperature = mydht.readTemperature();      // 온도 값을 읽어옵니다.
    float humidity = mydht.readHumidity();           // 습도 값을 읽어옵니다.

    if (temperature > 25 || humidity > 50) {
        digitalWrite(redLedPin, HIGH);   // 빨간색 LED를 켭니다.
        digitalWrite(blueLedPin, LOW);   // 파란색 LED를 꺼냅니다.

        for (angle = 0; angle <= 180; angle += 20) {
            myServo.write(angle);        // 서보 모터를 각도에 따라 회전시킵니다.
            delay(15);
        }

        for (angle = 180; angle >= 0; angle -= 20) {
            myServo.write(angle);        // 서보 모터를 각도에 따라 회전시킵니다.
            delay(15);
        }
    } else {
        digitalWrite(redLedPin, LOW);   // 빨간색 LED를 꺼냅니다.
        digitalWrite(blueLedPin, HIGH); // 파란색 LED를 켭니다.
    }
}
```

# 3. 시스템 구성도 및 핵심 기능

(Smart Shoe Care 2.0)

## <코드 구성>

초음파 센서 -> 신발 감지, 사용시간 DATA

```
float duration, distance;
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
distance = ((float)(340 * duration) / 10000) / 2;

if (distance < detectionDistance) {
    if (!shoeDetected) {
        shoeDetected = true;
        shoeDetectedTime = millis();
        Serial.println("신발이 감지되었습니다.");
    }
} else {
    if (shoeDetected) {
        shoeDetected = false;
        unsigned long elapsedTime = millis() - shoeDetectedTime;
        totalElapsedTime += elapsedTime;
        Serial.println("신발이 없습니다.");
    }
}
```

## 객체 탐지 기능

부저 센서 -> 신발 도난 감지

```
if (distance < detectionDistance) {
    if (!shoeDetected) {
        shoeDetected = true;
        shoeDetectedTime = millis();
        Serial.println("신발이 감지되었습니다.");

        for (int i = 0; i < 9; i++) {
            tone(buzzerPin, myNote[i], myDu[i]);
            delay(myDu[i] * 1.3);
            noTone(buzzerPin);
        }
    } else {
        if (shoeDetected) {
            shoeDetected = false;
            unsigned long elapsedTime = millis() - shoeDetectedTime;
            totalElapsedTime += elapsedTime;
            Serial.println("신발이 없습니다.");

            for (int i = 0; i < 6; i++) {
                tone(buzzerPin, mouthNote[i], mouthDu[i]);
                delay(mouthDu[i] * 1.3);
                noTone(buzzerPin);
            }
        }
    }
}
```

## 앱 & 시리얼 모니터 시작화

```
unsigned long currentTime = millis();
unsigned long elapsedTime = 0;

if (shoeDetected) {
    elapsedTime = currentTime - shoeDetectedTime;
}

mySerial.println("t"+String(temperature));
delay(1000);
mySerial.println("m"+String(humidity));
delay(1000);

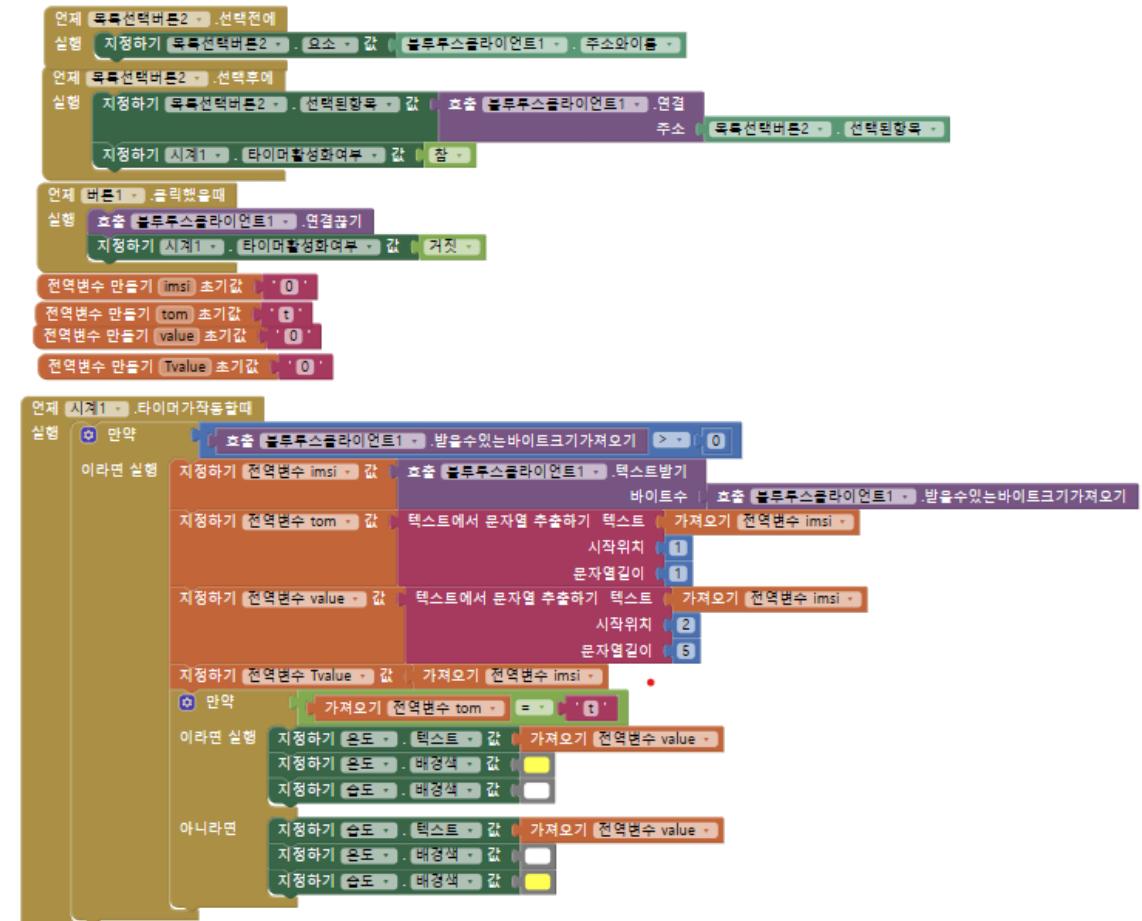
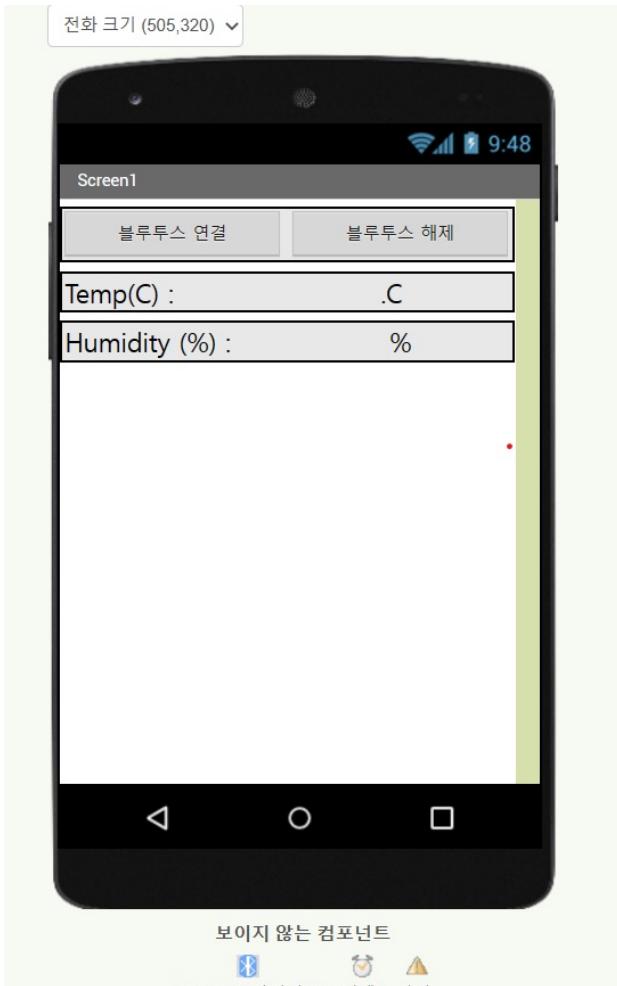
// 사용 시간 출력
Serial.print("사용시간 : ");
Serial.print(elapsedTime/1000);
Serial.println(" 초");

// 누적 사용 시간 출력
Serial.print("누적 사용 시간 : ");
Serial.print(totalElapsedTime/1000);
Serial.println(" 초");
}
```

### 3. 시스템 구성도 및 핵심 기능



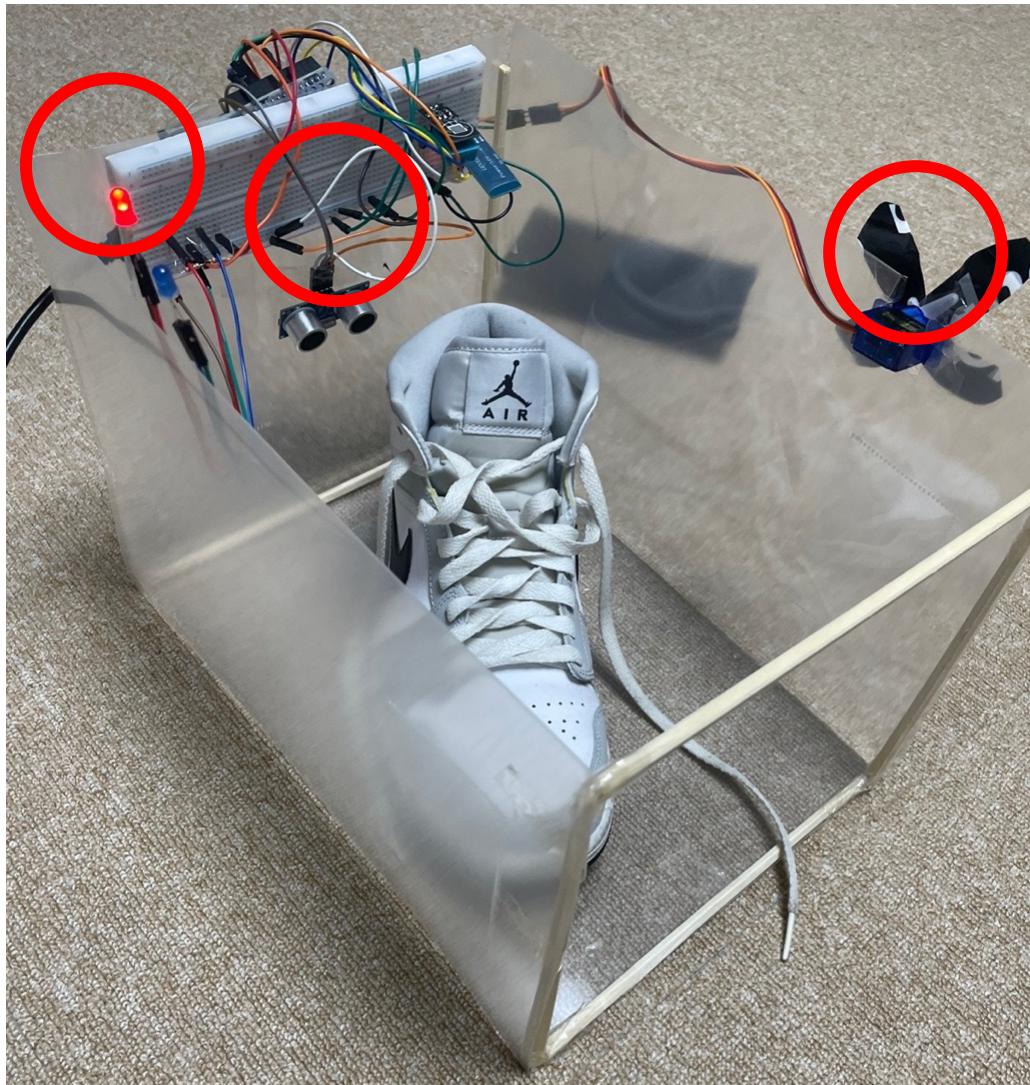
◀ 앞 인물 ▶





## 4. 시스템 구현

## Shoe Case Prototype



# 결과물



센서 별 구현 이미지

## 신발 관리 기능

<온 습도 센서 DHT 11>



온 습도 센서를 통해 슈케이스안의  
온습도를 측정

<블루투스 모듈>



블루투스와 연동해 휴대폰에  
서 현재 온습도를 알 수 있  
음

<서보 모터>



온도 > 25 || 습도 > 50

자동으로 신발의 적절한 보관  
상태를 유지

<LED>



온도 > 25 || 습도 > 50

온습도 상태에 따라 파란색  
빨간색으로 표현

## 객체 탐지 기능

<초음파 센서>



초음파 센서를 통해 신발의  
유무를 정확히 파악

<시리얼 모니터>

```
누적 사용 시간 : 0 초  
사용시간 : 0 초  
누적 사용 시간 : 14 초  
신발이 감지되었습니다.  
사용시간 : 0 초  
누적 사용 시간 : 14 초  
사용시간 : 2 초  
누적 사용 시간 : 14 초
```

<수동 부저>



신발 사용 정보를 시리얼 모  
니터를 통해 출력

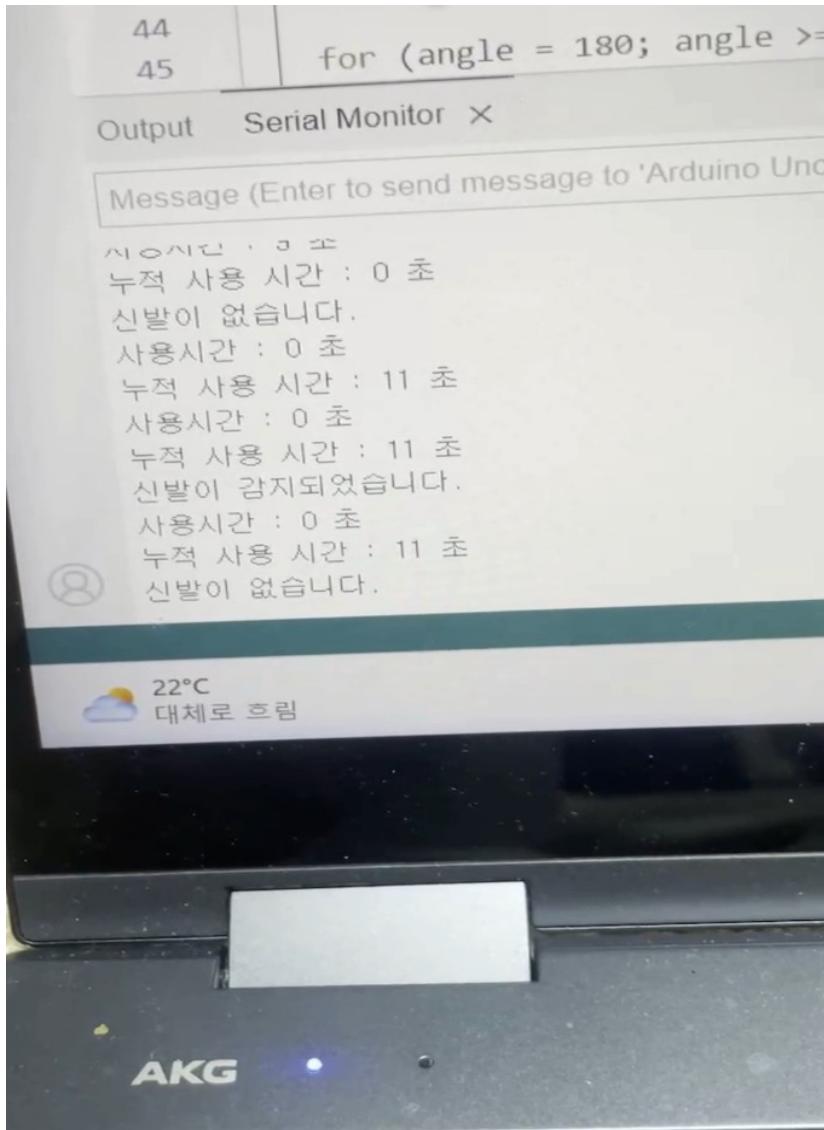
향후 로드셀과도 결합해 공증  
된 데이터셋을 만들 계획

신발의 도난방지를 위해  
신발 객체 탐지 시 부저로  
도난방지 음악이 나오도록 함.

## 신발 관리 기능



시리얼 모니터를 통해 신발 사용 시간 DATA SET 저장,  
신발 유무에 따라 시스템 작동



## 객체 탐지 기능

신발 객체 탐지를 통해 신발이 들어갈 때 나올 때 도난 방지 시  
스템(부저 알림) 작용





## 5. 최종 목표 및 향후계획

## UNIQUE SELLING POINT

## GOAL

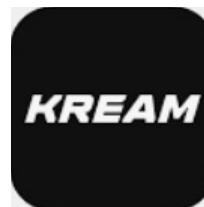
1. 최종 목표

이 프로젝트의 최종 목표는 사용자의 신발 관리를 스마트하게 도와주는 Smart Shoe Care 시스템을 구현하는 것이다.

사용자가 사용 시간, 온도 및 습도 등의 정보를 쉽게 추적하고 관리할 수 있는 시스템을 실제 완성 시킨다.

이를 위해 다양한 센서, 모듈을 활용하여 스마트폰 앱과 연동하며, 데이터베이스를 통해 정보를 저장 및 관리한다.

현재는 데이터베이스를 구현하기 어렵다고 판단하여, 간단히 작동 방식을 파악할 수 있도록 프로토타입을 만들었다.



B2B , B2C

2. 향후 계획

향후 계획으로는 더 많은 센서와 모듈을 활용하여 Smart Shoe Care 시스템의 기능을 확장하고, 사용자가 더욱 편리하게 사용할 수 있는 인터페이스를 개발하는 것이다.

<Smart Shoe Care>를 통해 얻은 사용시간의 DATABASE를 가지고 리셀 시장과 연동하여 리셀(중고) 거래 시에 신뢰성 있는 자료로 활용할 수 있게 한다.



수고 하셨습니다.