

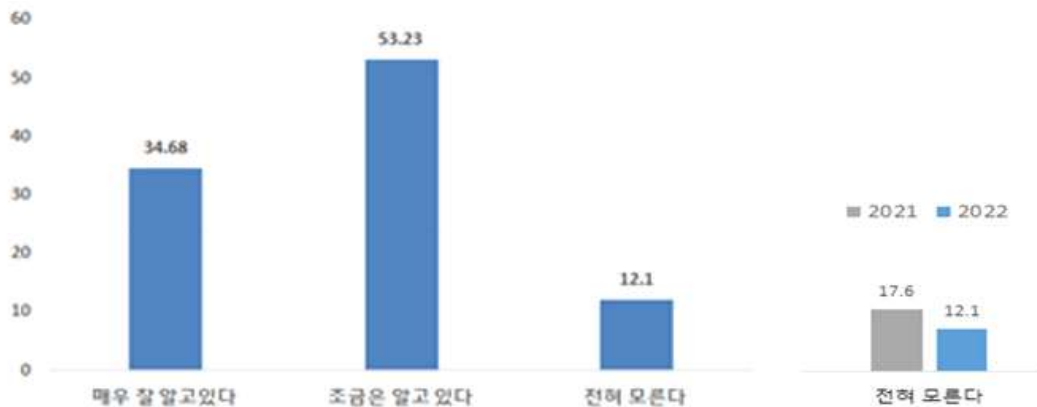
구르미 개발 보고서

한국공학대학교

강송구, 박정욱, 최지민

1. 작품 개요

반려식물이란 반려동물과 비슷한 의미로 정서적으로 가까이 두고 기르는 식물을 일컫는 말이다. 반려식물은 주인이 반려식물을 기르면서 교감하며 정서적으로 안정을 주는 역할을 할뿐더러 관상용, 인테리어, 치유 등의 역할을 하고 있다.



<그림1> 반려식물 인식 수준

농촌진흥청의 2022년도 자료 발표에 따르면 설문 응답자 중 반려식물의 개념에 대해 알고 있다는 응답이 87.9%로 매우 높게 나왔고, 반려식물을 아는 응답자 중 반려식물을 기르는 목적이 정서적인 교감인 사람이 55%가량이라고 했다. 이는 사람들이 반려식물에 대해 인지하고 있고, 반려식물을 기르면서 정서적인 교감을 원한다는 것을 알 수 있었다. 하지만 현재 시장에는 반려식물과의 교감을 보조해주거나, 체계적으로 식물을 관리해주는 제품이 부족하다고 생각하였다.

따라서 우리 팀은 이에 주목하여 더욱 정서적으로 주인에게 다가갈 수 있게 도와주며 주인이 외출 등의 이유로 반려식물의 관리가 힘들 시에 자동으로 관리해주고, 외출 시에도 식물의 상태를 확인할 수 있게 해주는 “구르미”를 만들어 사용자들이 더욱 간편하게 반려식물을 관리하고, 반려식물과 정서적으로 더욱 잘 교감할 수 있게 도움을 줄려고 한다.

2. 작품 설명

(1) 주요 동작 및 특징

우선 “구르미”의 주요 동작을 정리한 동영상 링크를 남긴다.

(링크 : https://youtu.be/4649_COoSPo)



<그림2> 구르미

“구르미”는 기존 화분 위에 설치할 수 있는 제품이며 전용 어플을 제공하여 사용자 편의성을 도모하였다. 또한 데이터베이스를 통해 반려식물에게 유용한 정보들을 기록, 열람할 수 있고 클라우드 컴퓨팅으로 사용자의 제스처를 분석하여 제스처에 따른 OLED디스플레이를 제어하여 캐릭터의 얼굴과 같은 화면을 제스처에 따라 바꾸어 사용자가 친근감을 더욱 잘 느낄 수 있도록 설계하였다. 아래에는 각 분야에 따른 주요 동작을 서술하겠다.

(I)ESP32-S3

“구르미”의 본체 제어에는 ESP32-S3 MCU를 사용하였고 이는 온습도센서(DHT-11), 수위 센서(HS-WATER-SENSOR), 조도 센서(HS-CDSM), 토양 습도 센서(HS-Soil Moisture Control Module-5V)의 센서를 제어하여 반려식물이 있는 환경의 온습도, 조도를 측정하고 수위 센서를 사용하여 반려식물에게 주는 물을 저장하는 물통의 수위를 측정하며 토양 습도 센서를 사용하여 반려식물이 심겨져 있는 토양의 습도를 측정한다.

액추에이터는 워터 펌프(HS-WATER PUMP II)와 서보 모터(SG90 MICRO SERVO)를 사용하였다. 워터 펌프로 물통에 있는 물을 식물에게 공급할 수 있게 하였고, 서보 모터는 카메라의 각도를 조절하여 사용자가 원하는 각도를 커스터마이징 할 수 있게 하고, 사용자가 어플을 통해 외출 모드를 설정 시 카메라를 식물 쪽으로 향하게 하여 어플로 식물의 모습을 확인할 수 있게 하였다.

“구르미”의 감정표현 및 정보를 나타내는데 사용된 디스플레이는 OLED디스플레이(0.96INCH-12864 4PIN)를 사용하였고, 제스처에 따른 상호작용 및 정보를 표시할 수 있게 하였다.

사용자의 제스처를 분석할 수 있게 영상을 서버에 스트리밍 해주는 카메라로는 OV-2460카메라 모듈을 사용하였고, ESP32-S3가 클라우드 컴퓨팅 서버에 WebSocket방식으로 영상을 스트리밍 해준 뒤 클라우드 컴퓨팅으로 사용자의 제스처를 분석한다.



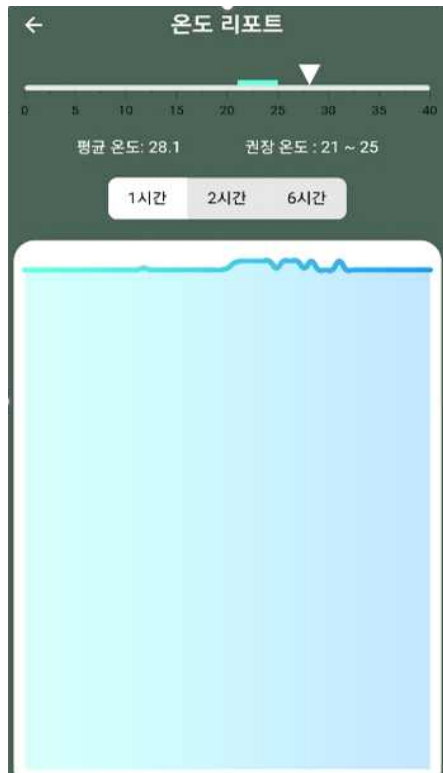
<그림3> 제스처에 따른 OLED디스플레이

(II)전용 어플

전용 어플은 농촌진흥청에서 제공하는 반려식물 데이터를 내부에 저장하며, 그를 통해 반려식물에게 적절한 온습도, 토양 습도, 조도가 무엇인지 정보 제공한다. 이를 통해 사용자는 주기별로 알맞은 양의 물을 줄 것인지, 적절한 토양 습도를 맞춰주게 물을 줄 것인지, 수동으로 물을 줄 것인지 설정할 수 있다.

또한 사용자는 식물이 어떤 온습도, 조도, 토양습도의 환경에 있었는지에

대한 시간별 데이터를 확인할 수 있고, 사용자의 이해도를 높이기 위해 와 같이 그래프를 제공한다. 이 정보는 ESP32-S3의 센서에서 측정된 값을 Firebase에 저장하여, 이 값을 열람하여 제공한다.



<그림4> 어플 그래프



<그림5> 어플 메인 화면



<그림6> 도전 과제 시스템

그리고 <그림5>와 같이 클라우드 컴퓨팅에서 사용자의 제스처를 분석한 값을 읽어와 어플의 메인 화면의 화분의 표정을 다르게 한다.

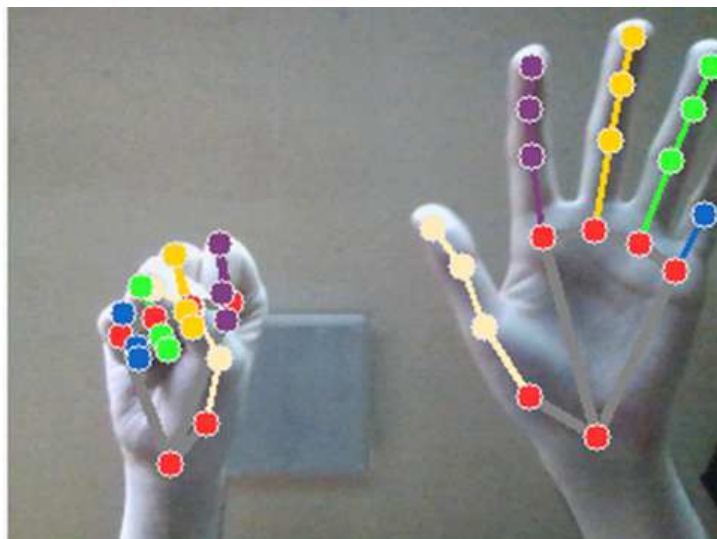
또한 우리 팀은 사용자가 제품을 단발성으로 사용하는 것이 아니라 오랜 시간동안의 관심을 유도할 방법을 고뇌하였고, 우리가 평소에 흔하게 즐기는 게임들의 시스템에 착안하여, 도전과제 시스템과 레벨 시스템을 어플에 도입하였다. 반려식물에게 “손하트”, “엄지척” 등의 제스처를 사용자가 선보이거나, 외출 모드를 활성화하고 카메라로 식물의 모습을 확인하는 등의 기능들을 사용자가 사용하면, 이에 따른 도전과제를 사용자는 성공할 수 있고, 식물의 레벨을 증가시켜 어플 화면에서도 식물의 모습이 점차 성장해 나가는 것을 관찰해 가면서 반려식물에 대한 관심을 장기적으로 유도할 수 있도록 하였다.

(III)클라우드 컴퓨팅

클라우드 컴퓨팅에서는 구글 클라우드 컴퓨팅(GCP)를 제스처 분석에 사용하였고, 구글 Firebase를 데이터를 저장하는 용도로 사용하였다.

클라우드 컴퓨팅으로 사용자가 카메라 앞에서 행하는 제스처를 실시간으로 인식, 예측이 필요하였고 제한된 컴퓨팅 자원으로 해결해야 되었기 때문에 scikit-learn 모델 중 빠른 처리 속도를 지닌 Random Forest를 사용하였고 2초에 10번 꼴로 제스처를 인식하여 실시간 처리를 구현하였다. 또한 CPU만으로 제스처를 처리하게 설계하여 서버 비용을 절감하였다.

제스처는 손의 1~5숫자와 “손하트”, “엄지척”의 제스처를 구분할 수 있고, 예기치 않은 상황에서 양손이 동시에 인식될 경우 각각의 두 손의 제스처를 읽은 후 두 제스처중 정확도가 높은 제스처만 사용하게 하여 장애 요인을 해결하였다.



<그림7> 제스처 처리 및 양손 인식

처리한 제스처 데이터는 실시간으로 Firebase에 전송되어 ESP32-S3와 어플에서 정보를 열람할 수 있게 하였다. 또한 Firebase는 각각의 정보를 저장, 열람하는 용도로 사용하였으며, 특히 반려식물의 데이터를 시간별로 저

장하는 역할을 주요하게 수행했다. 또한 각 상황별의 동기화도 Firebase를 통해 이루어졌다.

종합하면 “구르미”는 사용자가 수령 이후 사용 시 어플로 “구르미”설정을 할 수 있으며, 이는 농촌진흥청 데이터를 기반으로 사용자에게 추천을 해준다.

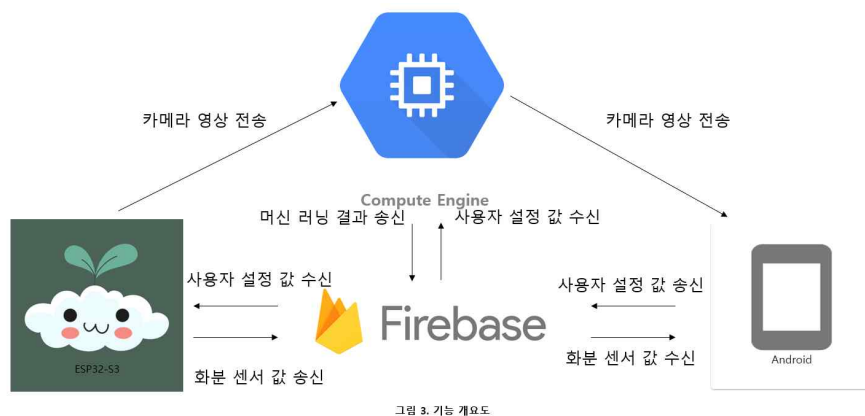
각종 센서와 액추에이터를 통해 다양한 데이터와 기능을 제공하고 반려식물과 더욱 친근한 교감을 유도하며 이는 도전과제 시스템, 레벨업 시스템, 화분에게 행하는 제스처와 이에 대한 피드백을 통한 상호작용을 통해 사용자에게 장기적인 반려식물에 대한 관심을 유도한다. 만약 상호 작용 시스템을 원하지 않는 사용자의 경우에도 식물에게 유용한 온습도, 조도, 토양 습도 값을 열람할 수 있고, oled디스플레이를 시간, 온습도, 조도, 토양 습도 디스플레이 모드로 바꾸어 사용할 수 있다.

또한 장기적인 관리가 어렵거나 긴 시간동안 사용자가 외출할 시에 대비해 주기별, 혹은 토양 습도에 따른 자동 물 공급 기능을 제공하며, 사용자가 이 기능을 원하지 않을 시 수동으로 물을 주는 모드를 제공하여 사용자가 식물의 데이터를 열람하여 수동으로 물을 공급할 수 있게 하였고, 외출 모드를 설정 시에는 카메라를 위의 링크의 영상에서처럼 사용자 방향이 아닌 식물 쪽으로 향하게 하여 사용자가 어플로 식물의 상태를 확인할 수 있게 하였다.

이러한 기능들로 사용자는 반려식물에게 더욱 친근감을 가질 수 있고 상호 작용 시스템을 통해 반려식물과 더욱 친밀감을 형성할 수 있으며, 농촌진흥청에서 제공하는 데이터를 사용하여 더욱 정확하게 반려식물을 기를 수 있고, 외출이 잦은 1인 가구의 경우에도 자동 관리 및 카메라로 상황을 확인하여 이러한 이유로 반려식물을 기르지 못하였던 사용자들에게도 반려식물을 기를 수 있게 하였다. 또한 화분을 새로 구매하는 것이 아닌 기존 화분을 유지하면서 사용할 수 있게 하여 사용자는 비용을 절감할 수도 있고, 기존에 기르던 반려식물에게도 사용할 수 있다는 장점이 있다.

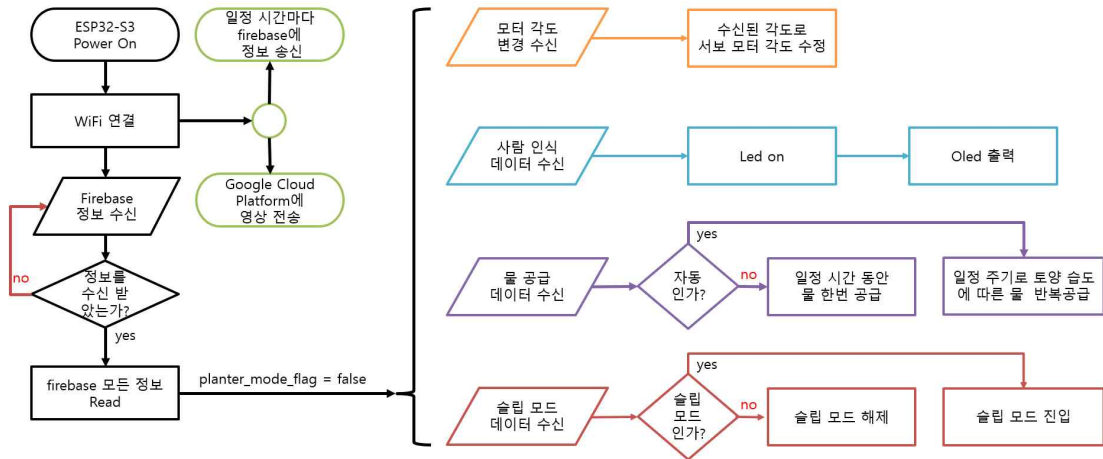
(2) 전체 시스템 구성

“구르미”는 아래의 <그림8>과 같은 구성을 지닌다.

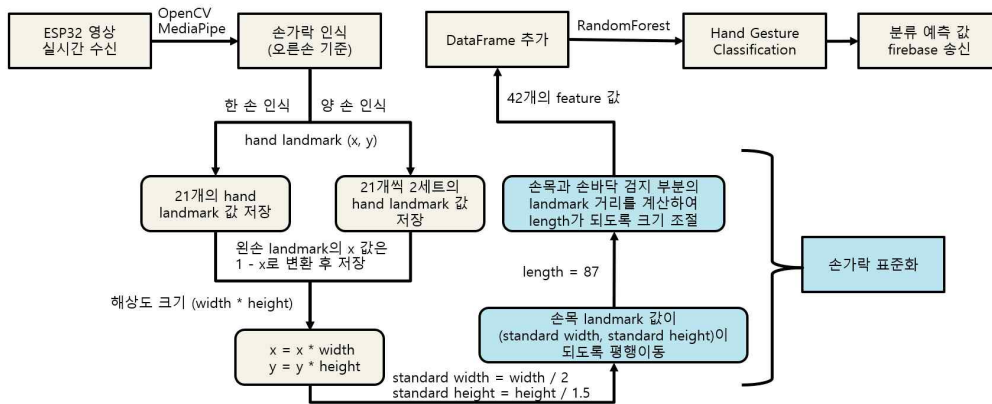


<그림8> 전체 시스템 구성도

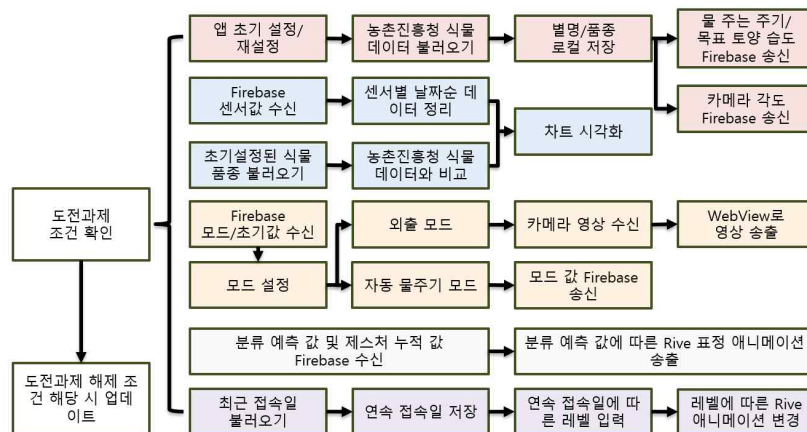
“구르미의 각 시스템 동작 흐름도는 아래 <그림9>, <그림10> <그림11>과 같다.



<그림9> ESP32-S3 동작 흐름도



<그림 10> 영상 처리 흐름도



<그림11> 어플 동작 흐름도

ESP32-S3는 “구르미”의 영상, 센서값을 Firebase에 전송한다. 영상의 경우에는 클라우드 컴퓨팅을 통해 제스처를 분석하거나, 결과 값을 Firebase 전송하고, 영상을 어플에 전송한다.

어플에서는 이러한 값을 바탕으로 사용자에게 애니메이션 및 인터페이스를 제공하고 제공된 값을 바탕으로 사용자가 여러 설정들을 변경하면 이를 다시 Firebase에 전송하고, ESP32-S3는 Firebase에서 이러한 값을 읽어 액추에이터를 제어한다.

종합하면, 사용자는 어플을 통해 “구르미”를 제어할 수 있고, 각각의 시스템은 Firebase를 통해 정보를 송/수신하며 영상 전송의 경우에는 ESP32-S3가 클라우드 컴퓨팅에 전송, 이를 다시 어플에 전송하는 식의 방식을 사용한다.

(3) 개발 환경

(1) 개발 언어, Tool, 사용 시스템

“구르미”의 개발은 <그림12>의 언어들과 Tool들을 사용하였다.



<그림12> 개발 언어 및 Tool

-ESP32-S3의 개발은 아두이노 Tool로 Arduino IDE를 사용해 개발을 하였다.

-클라우드 컴퓨팅 개발은 Google Cloud Platform에서 제공하는 컴퓨팅을 사용하여 ubuntu를 os로 설치하였고 python언어를 사용, OpenCV와 Flask, MediaPipe등의 Tool을 사용하였다.

-어플의 개발은 Flutter언어를 사용하여 개발하였다.

-“구르미”의 플라스틱 케이스는 Fusion360 Tool을 사용하여 3D 모델링을 만든 후 JLCPCB의 3D프린팅 서비스를 이용하여 출력하였다.

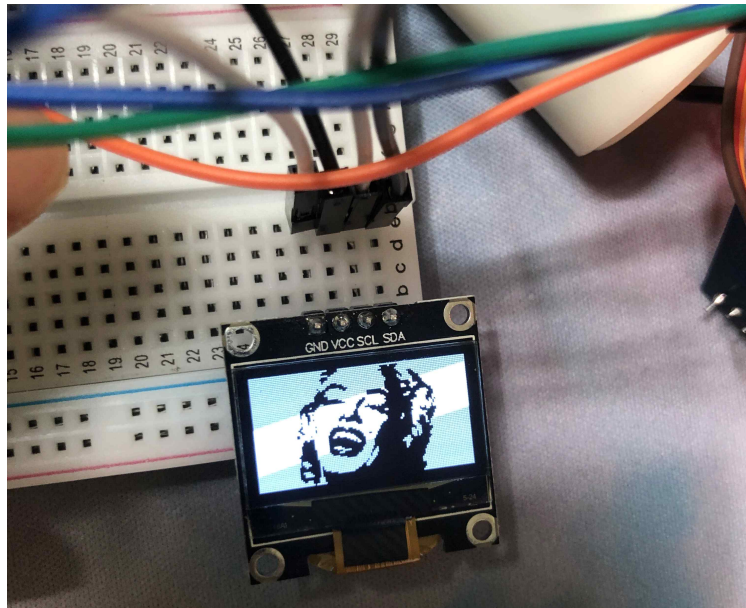
4. 단계별 제작 과정

(1) 아이디어 회의 및 부품 주문

반려식물을 관리하는 “구르미”의 제작 결정 이후, 반려식물의 데이터를 습득하기 위한 센서들을 찾아보았고, 온습도센서, 조도센서, 토양 습도 센서를 이용하기로 결정하였다. 그리고 물을 식물에게 자동으로 공급하기 위해서 워터 펌프의 사용을 결정하였고 이를 제어하기 위해 모터 드라이버를 사용하였다. 물통의 물의 양을 체크하여 물통에 물이 거의 남지 않았을 때 사용자에게 알려줘야 했기 때문에 수위 센서를 사용하기로 결정하였다.

사용자가 반려식물과 더욱 친근감을 느끼게 하기 위한 제스처 시스템은 카메라를 필요로 하였는데, 이는 OV2460카메라를 사용하기로 하였고, 이 모든 시스템을 제어하는 MCU로는 와이파이 기능을 내장하고 성능이 강화된 ESP32-S3를 사용하기로 하였다.

또한 최종 작품에는 카메라, 센서, 액추에이터를 조립할 수 있는 케이스가 필요하였고, 이를 해결하기 위해 직접 3D모델링을 하기로 하였다.



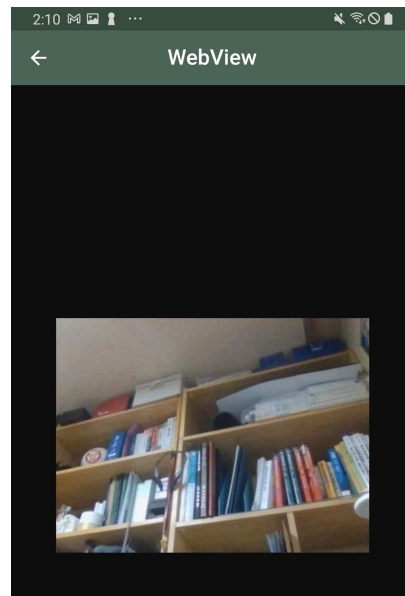
<그림13> 브레드보드 테스트 모습

(2) 브레드보드에 회로 구성

주문한 부품이 도착한 뒤, ESP32-S3와 각종 센서, 액추에이터, 카메라의 동작 확인을 각 테스트 코드를 아두이노 Tool을 이용해서 작성, 검증하였다. 아이디어 회의에서 나온 동작 방법을 목표로 하여 입력값에 따른 액추에이터의 동작과 센서 값을 oled의 디스플레이에 출력하는 방법을 사용하였다.

(3) ESP32-S3와 서버간 영상 전송 테스트

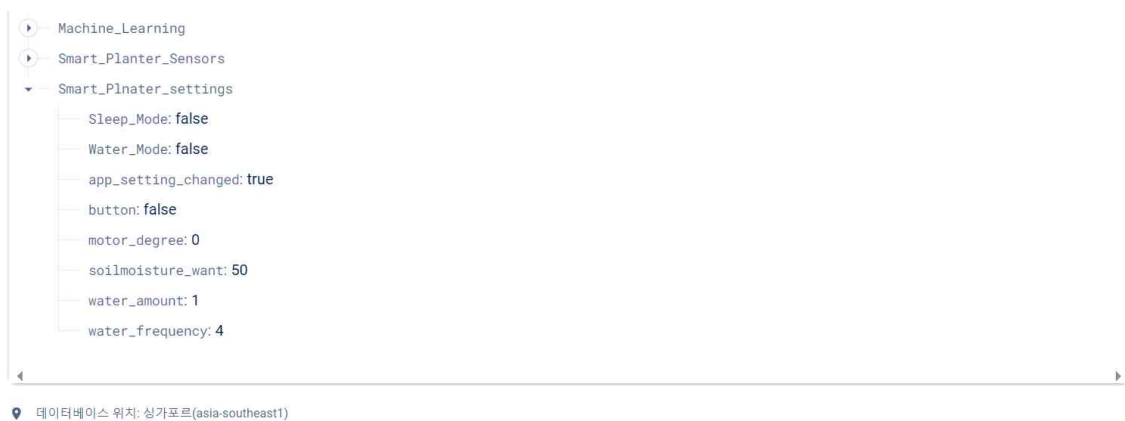
ESP32-S3의 OV2640카메라의 촬영되는 영상을 클라우드 컴퓨팅 서버로 보내고 영상을 처리하여 제스처를 구분해야 했기 때문에 ESP32-S3의 WiFi 기능으로 인터넷에 연결한 뒤 서버로 보내는 것을 검증하였고, 또한 어플의 화면에도 출력해주어야 했기 때문에 이를 검증하였다.



<그림14> 영상 전송 확인

(4) Firebase 정보 송수신 테스트

각 시스템간의 통신에는 Firebase를 사용하기로 하였기 때문에 각 시스템의 Firebase의 정보 송수신을 테스트 하여야 했다. 각 시스템간의 Firebase 실시간 데이터베이스에 접근하는 과정을 테스트 하고 혼동이 없도록 주고받는 데이터 형식을 정리하여 데이터베이스에 저장되는 데이터를 일치시켰다.



<그림15> 파이어베이스 ESP32 <-> 어플 간 세팅값



<그림16> 머신러닝 결과 데이터 실시간 저장 모습

(5) 각 시스템의 연동 테스트

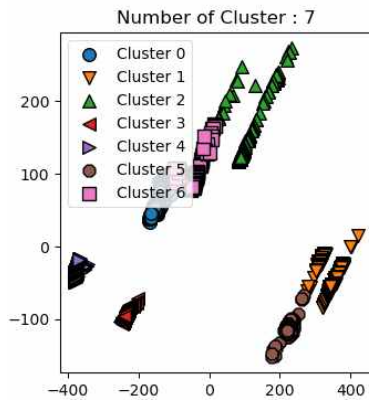
실시간으로 저장되는 머신러닝 데이터를 읽어오고 그에 따라 표정이 변하는 “구르미”의 모습을 구현하였다. 머신러닝 데이터는 ESP32 와 어플에서 실시간으로 읽어오며 어떤 제스처를 입력하느냐에 따라 “구르미” 본체와 어플에 있는 캐릭터 표정이 변화한다.

어플과 ESP32간에는 water_mode와 sleep_mode, 카메라 각도와 초기 세팅 값 등 어플에서 버튼을 통해 설정하는 모드 변경을 ESP32에 적용하는 것을 테스트 했고, 반대로 ESP32에서 Firebase에 보내는 온습도, 토양습도 센서, 조도센서 등의 값을 어플에서 읽어오는 것도 테스트 하였다.

(6) 제스처 분류 모델 학습, 어플 UI 정리

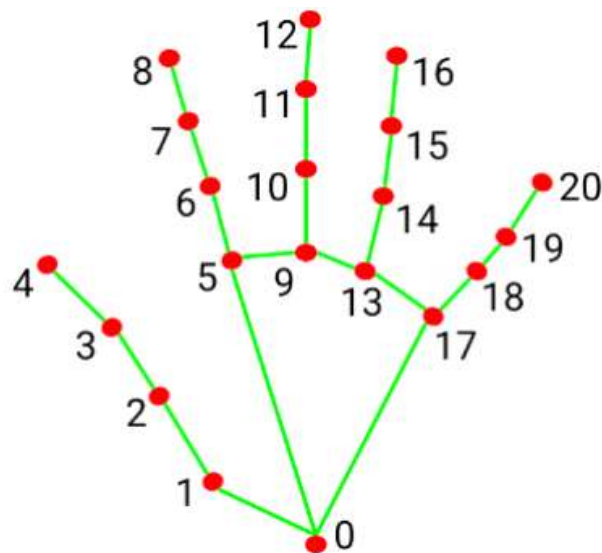
“손하트”, “엄지척” 과 같은 “구르미”와 상호작용할 수 있는 제스처의 인식을 강화하고, 센서 정보, 온습도, 날짜 및 시간 표시, 세팅 초기화의 4가지 상황을 손가락 숫자에 따라 명령을 내릴 수 있도록 7가지 hand gesture에 대한 학습을 진행했다.

해당 학습을 진행하기 위해 hand gesture 분류를 위해 Mediapipe에서 제공하는 hand detection 모델을 사용했다.



Cluster 0: 손가락 3개
 Cluster 1: 손하트
 Cluster 2: 엄지척
 Cluster 3: 인사(손가락 5개)
 Cluster 4: 손가락 2개
 Cluster 5: 손가락 4개
 Cluster 6: 손가락 1개

<그림17> hand gesture 데이터 분포도

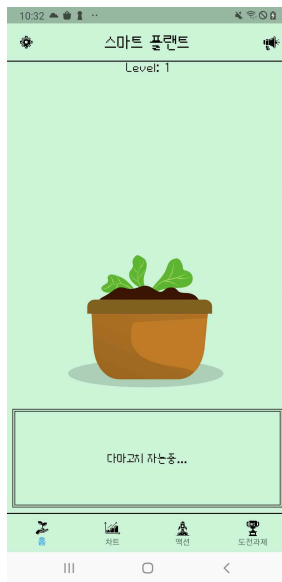


<그림18> hand landmark 위치

각 landmark의 x, y 정보를 사용해 총 42개의 특징을 가진 모델을 설계했고, 각 gesture의 분포도를 확인하기 위해 PCA 변환을 통해 시각화했다.

특징이 유사한 hand gesture(Cluster1&6)를 최대한 정확하게 분류하면서 고차원 데이터를 신속하게 처리하기 위해 앙상블 모델 중 RandomForestClassifier를 사용하여 hand gesture 분류 정확도를 높였다.

초기 어플 UI는 다마고치와 같은 컨셉으로 제작하려 했으나 기획했던 기능을 모두 넣으려고 하자 UI 정리가 잘 안되는 측면이 있었고, 스마트 IOT 어플 같은 디자인으로 UI를 다시 정리하였다.



<그림19>초기 UI

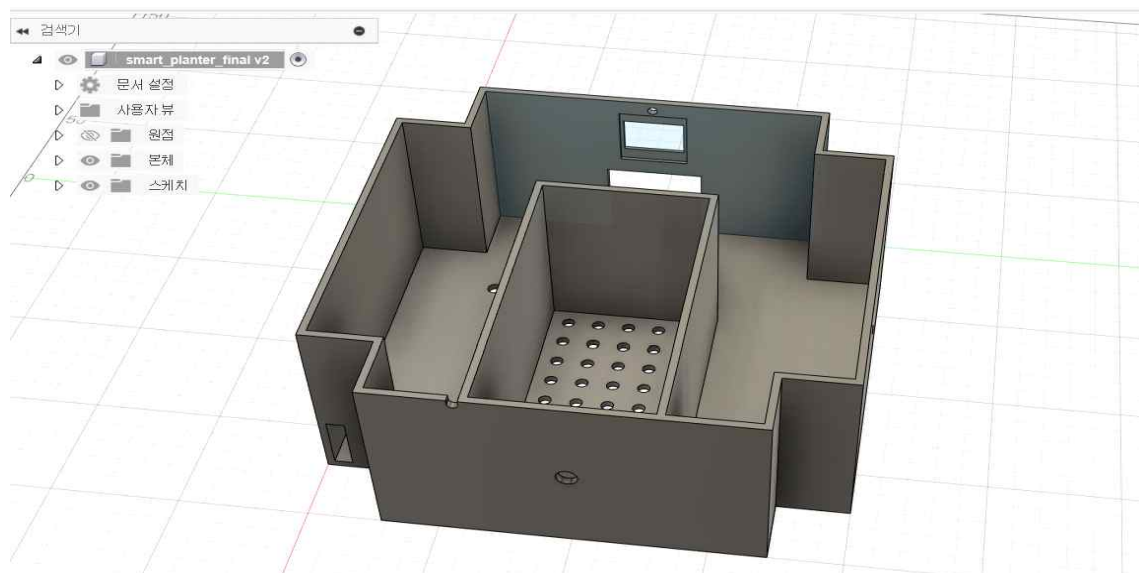


<그림20> 변경된 UI

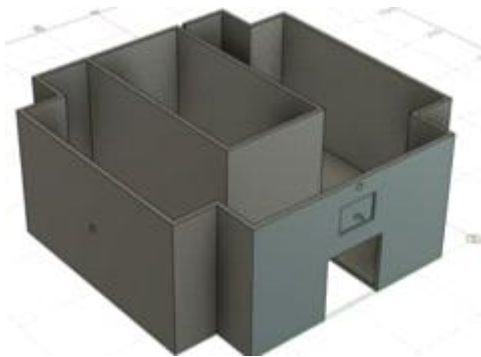


(7) “구르미” 케이스 제작

연동 테스트를 확인하고 구르미 케이스를 제작하였다. 물이 지나가는 부분은 방수가 확실하게 되어야하기 때문에 몸체를 3D 프린팅으로 제작하였고, 식물 위로 몸체를 지지할 목재와 물을 담을 방수 아크릴 상자를 합쳐 “구르미” 외관을 완성하였다.



<그림21> 3D 모델링 과정



<그림22> 최종 출력물 몸체



<그림23>카메라 박스 출력물



<그림24> 조립 사진

(8) 최종 점검

(시연 링크) : https://youtu.be/4649_COoSPo

최종적으로 “구르미” 본체, 서버, 어플이 동시에 동작하는 과정을 영상으로 촬영하였다.

5. 사용한 제품 리스트

[Espressif Systems] ESP32-S3-DEVKITC-1-N8R8

<https://www.devicemart.co.kr/goods/view?no=14236751>

[OEM] DHT11 온습도 센서모듈 [SZH-EK024]

<https://www.devicemart.co.kr/goods/view?no=1278926>

[SMG] 아두이노 수위 센서 모듈 [SZH-EK057]

<https://www.devicemart.co.kr/goods/view?no=1287088>

[SMG-A] 아두이노 토양 습도 센서 모듈 [SZH-EK106]

<https://www.devicemart.co.kr/goods/view?no=1376532>

[TowerPro] [정품] SG90 360도 디지털 서보모터

<https://www.devicemart.co.kr/goods/view?no=12503476>

[네오텍스] 0.96인치(Inch) OLED 디스플레이 (백색) 128 x 64
(PN-OLED-W096)

<https://www.devicemart.co.kr/goods/view?no=15348986>

[SMG-A] 아두이노 CdS 광/조도 센서 모듈 [SZH-SSBH-011]

<https://www.devicemart.co.kr/goods/view?no=1327438>

[ArduCAM] 아두이노 2Mp Megapixel OV2640 CMOS 1/4 inch Camera
Module with LS-4011 M12 Mount lens [U3359]

<https://www.devicemart.co.kr/goods/view?no=10824352>

[SMG] Dual H-Bridge L9110s 모터 드라이버 모듈 [SZH-MDBL-002]

<https://www.devicemart.co.kr/goods/view?no=1324245>

워터펌프 : 보유한 제품 사용

6. 기타

-소스코드-

(1). GooReumi('구루미'의 동작 코드를 모아놓은곳)

-주요 파일 구성

- ESP32_S3_camera : 카메라의 동작 코드
- ESP32_S3_firebase_read_writre: Firebase 송수신 코드
- ESP32_S3_led_motor : led점등과 워터펌프 동작 코드
- ESP32_S3_oled: oled동작 코드
- ESP32_S3_servo: 서보모터 동작 코드
- image_index.h: "구루미"에서 사용하는 이미지 코드
- GooReumi.ino: "구루미"의 동작 코드

- 주요 함수 별 기능

- read_firebase: Firebase의 정보를 수신하는 함수
- send_firebase: Firebase에 정보를 송신하는 함수
- display_ssd1306 : oled에 화면을 출력하는 함수
- set_camera: 카메라 동작 준비 및 websocket에 연결하여 영상 송신을 준비하는 함수
- send_camera_data: 카메라의 영상을 Google Cloud Platform에 송신하는 함수
- Task1code, Task2code : 듀얼코어 작동 함수
- servo_camera: 서보모터를 제어하여 카메라의 각도를 변경해주는 함수
- control_water_motor_auto: 화분에 물을 자동으로 주는 동작을 하는 함수
- control_water_motor_manual: 화분에 물을 수동으로 주는 동작을 하는 함수
- set_wifi: "구루미"의 와이파이 연결을 하는 함수

(2). 영상 처리 & 영상 송출

-주요 파일 구성

- ESP32_websocket : ov2640의 프레임 수신 코드
- hand_gesture_training: hand landmark 학습용 데이터 코드
- hand_gesture_learning: Random Forest 학습 코드
- hand_gesture_predict: Gesture 예측 후 Firebase 수신 코드
- hand_gesture_processing: hand landmark 데이터 가공 코드
- streaming_server: 영상 송출 API 코드

- 주요 함수 별 기능

- human_state: 매 프레임 얼굴, 손 인식 여부를 저장하는 함수
- human_change: 사람 인식 여부를 Firebase에 수신하는 함수
- hand_change: 예측된 Gesture 종류를 Firebase에 수신하는 함수
- hand_sum: 일정 시간 동안 예측된 Gesture 횟수를 저장하는 함수
- hand_data_process: hand_landmark 데이터를 모델 예측에 쓰이기 위해 가공하는 함수
- generate_frames: ov2640의 프레임을 읽어 송출하는 함수

(3). app(어플의 동작 코드를 모아놓은곳)

-주요 파일 구성

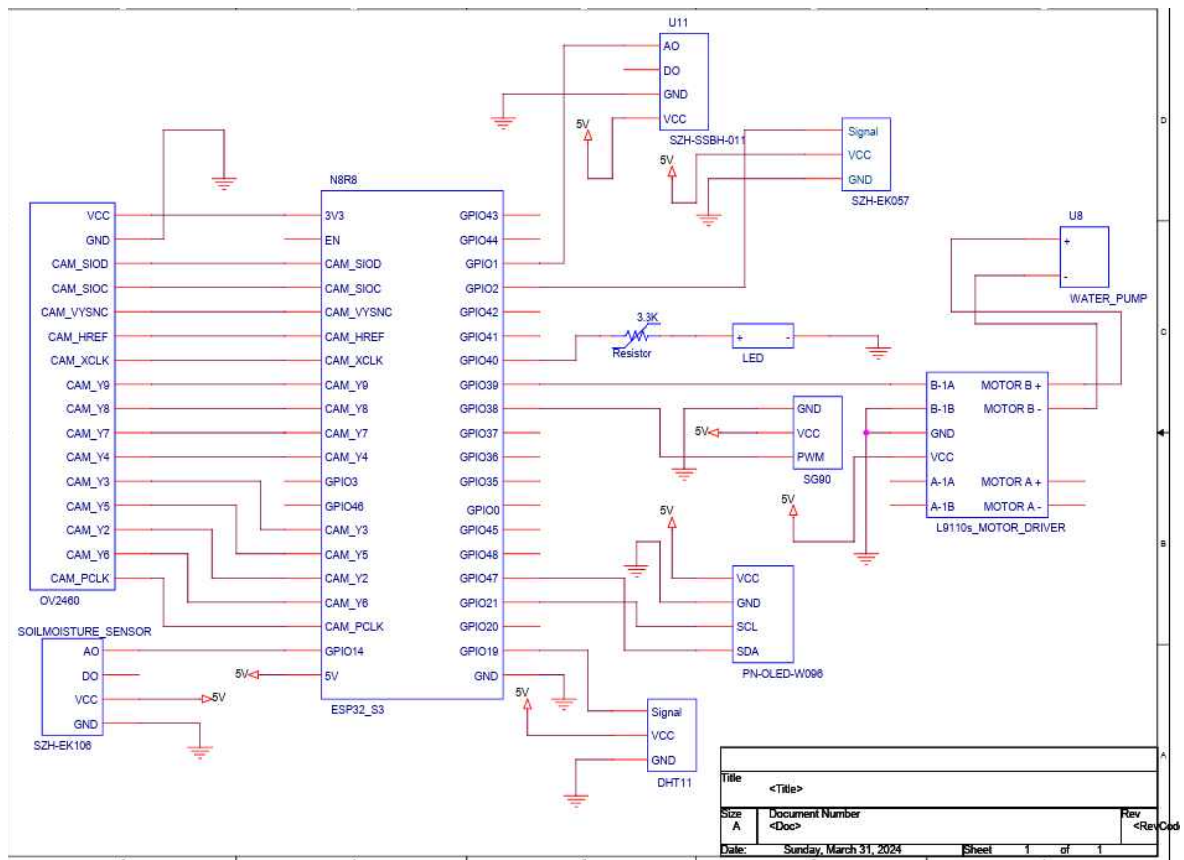
- main.dart: main파일, 어플 시작 시 첫 세팅 페이지 이동
- firebase_options.dart: Firebase 연결 key 포함
- introduction_page.dart: 초기 식물 세팅 페이지
- singleplant_page.dart: “구르미” 캐릭터와 차트, 설정 등 종합 기능 페이지
- webview_page.dart: 식물 영상 확인 페이지

- 주요 함수 별 기능

- checkFirstTime: 어플 실행이 처음인지 확인하는 함수

- loadUnlockStatus: 도전과제 달성 정보를 로딩하는 함수
- brightnessFetchDataUsingOnValue: 조도 정보를 가져오는 함수
- humidityFetchDataUsingOnValue: 습도 정보를 가져오는 함수
- moistureFetchDataUsingOnValue: 토양 습도 정보를 가져오는 함수
- tmperatureFetchDataUsingOnValue: 온도 정보를 가져오는 함수
- waterLevelFetchDataUsingOnValue: 수위센서 정보를 가져오는 함수
- getSleepMode: 외출 모드 정보를 가져오는 함수

-회로도-



<그림25> 회로도

-참고 문헌-

- 농촌진흥청 반려식물 데이터베이스
- 농촌진흥청 농촌진흥일보 기사

https://rda.go.kr/board/boardfarminfo.do?mode=view&prgId=day_farmprm_ninfoEntry&dataNo=100000784485&CONTENT1=