

# 종합설계 최종보고서

( 2024학년도 1학기 )



설계제목	Smart Recycling System		
팀명	에코아리움		
연구기간	2024년 01월 10일 ~ 2024년 06월 11일		
연구참여자 (담당분야)	김찬영 (20190895): Project Manager, Model, Pi S/W		
	김태현 (20190898): Android APP, H/W		
	지성원 (20190948): AP Server, WEB Client		
	허진환 (20190954): DB Server, Network, Security		
지도교수	정복래 교수님	제출일	2024년 06월 11일

성결대학교 정보통신공학과

# 목 차

1. 설계 제목	3
2. 설계 목적	3
3. 과제의 필요성	3
4. 프로젝트 목표와 기준	12
5. 설계과정	14
6. 제작	35
7. 시험	41
8. 평가	49
9. 추진체계	53
10. 프로젝트 추진 일정	54
11. 결론	55

## 부록

- 1) 회의록: 부록1\_종합설계\_에코아리움\_회의록.pdf
- 2) 보고서: 부록2\_종합설계\_에코아리움\_DB서버보안점검리스트.pdf
- 3) 소스코드(GitHub Repository Link)  
링크: [SKU-Ecoarium https://github.com/mixgolem/SKU-Ecoarium.git](https://github.com/mixgolem/SKU-Ecoarium.git)

## 1. 설계 제목

Smart Recycling System : 라즈베리파이와 인공지능망을 활용한 플라스틱 컵 수거 장치 제작 및 Node.js 서버, AWS DB 서버, Kotlin APP 구축

## 2. 설계 목적

- 1) 그룹 프로젝트 진행 : Github를 활용한 그룹 프로젝트에 대한 이해
- 2) 책임의식 : 개인이 수행해야 할 부분에 대한 책임의식 향상
- 3) 소통 및 문제해결 : 문제 발생 시 상호간의 적극적인 소통을 통한 문제 해결
- 4) 새로운 BM : 단순한 프로젝트를 넘어서 시스템과 환경 구축을 통한 새로운 비즈니스 모델 형성
- 5) 전문성 : 각자가 맡은 분야에 대한 전문적인 개발을 통해 개개인의 역량 향상
- 6) 환경 보호 : 플라스틱 재활용을 통해 해양 생태계 및 지구 온난화에 긍정적 영향
- 7) H/W와 S/W의 연결 능력 향상 및 IoT 개발 능력 향상과 각종 센서 및 모듈 응용 능력 향상

## 3. 과제의 필요성

### 3.1. 과제 추진동기

일회용 플라스틱 사용과 그로 인한 환경 문제는 현재 전 세계적으로 큰 과제 중 하나로, 특히 환경 보호와 재활용은 우리가 살아가는 지구의 미래를 위해 핵심적인 가치이다. 독일의 판트(Pfand) 제도는 그 중에서도 높은 효과를 보여주고 있는 좋은 예시 중 하나이다. 이 제도는 음료수 병이나 캔을 반납하면 보증금을 돌려받을 수 있는 제도로, 소비자들에게 책임감을 부여하고 재활용을 촉진하고 있다. 이러한 시스템은 환경을 보호하면서 경제적 이득을 창출하는 좋은 방법 중 하나이다. 일회용 플라스틱 컵 수거 장치를 설계하려는 것은 이러한 증가하는 플라스틱 쓰레기 문제에 대한 간단하지만 효과적인 해결책을 찾고자 하는 의지의 표현이다.

이 장치를 통해 사용한 플라스틱 컵을 효과적으로 수거하고 재활용 프로세스에 참여 함으로써 우리는 지속 가능한 소비문화를 조성할 것을 기대한다. 환경 보호에 대한 우리의 책임감은 더 이상 미룰 수 없는 과제로, 일회용 플라스틱 컵 수거 장치의 개발은 우리가 지속 가능한 미래를 위해 나아가는 한 걸음이 될 것이다. 이 프로젝트를 통해 우리는 소비자들과 기업들 간의 협력을 촉진하고, 환경을 보호하는 데에 기여할 수 있는 기회를 만들어낼 것이다.

## 3.2. 관련 제품 시장조사 및 특허조사

### 3.2.1. 시장조사

#### 3.2.1.1. 수퍼빈(주) : NephronBasic\_PETtype(혁신기기)

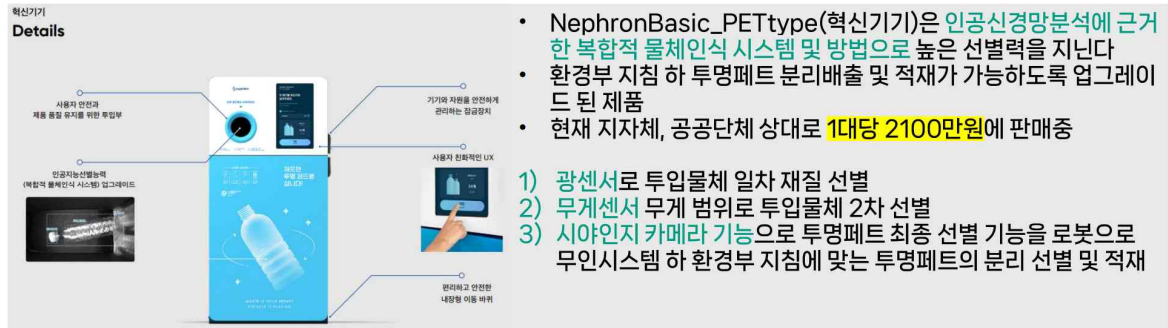


그림 - 네프론 조사 내용(수퍼빈superbin.co.kr 네프론 포트폴리오)

NephronBasic\_PETtype(이하 네프론)은 수퍼빈(주)사에서 개발하여 현재 국내에 상용화된 재활용 플라스틱 쓰레기 수거 장치이다. 네프론은 인공지능망분석에 근거한 복합적 물체인식 시스템 및 방법으로 높은 선별력을 지닌다는 장점이 있다. 환경부 지침 하 투명페트 분리배출 및 적재가 가능하도록 개선된 제품으로, 2023년 기준으로 지자체, 공공단체 상대로 1대당 약 2100만원에 판매중이다.

네프론의 사용 방법 및 동작은 아래와 같다.

- 1) 사용자가 스마트폰 마켓 앱(Play Store, App Store 등)에서 수퍼빈 App 설치 후 회원가입
- 2) 가까운 지역의 네프론 확인 및 App에서 네프론의 사용 가능 여부 확인
- 3) 네프론 기기에서 “시작하기” 버튼 누른 뒤, 라벨이 제거된 투명 플라스틱 쓰레기 투입
- 4) 플라스틱 쓰레기는 기기 내부로 들어가, 기기에서 광센서로 투입물체의 1차 재질을 선별함
- 5) 무게센서 무게 범위로 투입물체 2차 선별
- 6) 시야인지 카메라 기능을 통해 투명페트 최종 선별 기능을 로봇으로 무인시스템 하 환경부 지침에 맞는 투명페트의 분리 선별 및 적재가 이루어짐
- 7) 투입 종료 후 사용자 전화번호 입력하여 적립(일일 최대 적립 한도 존재함)

### 3.2.1.2. Tomra - RVM(Reverse Vending Machine)

**독일 판트 제도(Pfand)**

1. 언어 선택 후, 기계에 병을 넣는다.
2. 원료 버튼을 누른다.
3. 기계에서 나온 영수증을 보관한다.
4. 영수증을 제시하여 할인 받거나, 현금으로 교환한다.

\*해당 물건을 구입한 마트에서만 반환 및 영수증 사용 가능

## Pfand

보증금, 예치금을 뜻하는 독일어

- 빈 병을 돌려주고 병값을 돌려받는 독일의 '공병 보증금 제도'
- 일회용 제품을 줄이기 위해 2003년부터 시작된 **재활용 제도**
- 2022년 1월 개정 후 더욱 확대 적용됨
- 소비자가 빈 병을 판매처에 돌려주면 **구매가에 포함되어 지불했던 보증금을 돌려받는 방식이다.**

그림 - 판트 제도 소개(facebook@혼죽의제왕)



그림 - 판트 실제 사용 사진(YouTube@sinseolhee8946)

독일의 판트 제도 및 RVM(Reverse Vending Machine)은 일회용 음료용기의 재활용을 촉진하기 위한 혁신적인 시스템으로 알려져 있다.

독일의 판트 제도는 2003년에 도입된 것으로, 음료수 병이나 캔에 보증금을 부여하는 방식을 기반으로 한다. 소비자들은 음료용기를 소매점이나 자동 판매기에서 구매할 때 보증금을 지불하게 되고, 해당 용기를 반납하면 그에 상응하는 금액을 돌려받게 된다. 이로써 사용자들은 보증금을 환급 받기 위해 빈 용기를 재활용함으로써 환경 보호에 기여하게 되고, 동시에 재활용률이 높아지게 되어 지속 가능한 사회를 만들 수 있는 제도이다. 이 판트 제도에서 사용되는 기기 중 하나가 Tomra사의 RVM이다.

TOMRA는 세계적으로 알려진 재활용 기술 및 솔루션을 제공하는 노르웨이의 기업이다. 주로 자동화된 반납 시스템 및 정렬 시스템을 비롯한 재활용과 관련된 다양한 기술을 개발하고 생산하고 있다. TOMRA의 주요 제품 중 하나는 Reverse Vending

Machine(RVM)으로, 이는 일회용 음료용기를 반납하고 보증금이나 할인 혜택을 받을 수 있는 자동화된 장치이다. 아래는 RVM의 동작이다.

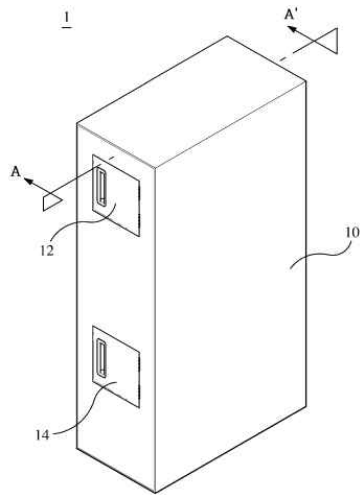
- 1) 용기 인식 기술: TOMRA의 RVM은 고정밀 센서와 카메라를 사용하여 음료수 병이나 캔을 신속하게 인식할 수 있다. 이 기술은 다양한 형태와 크기의 용기를 구별하고 정확한 인식을 가능하게 한다.
- 2) 자동 분류 및 분리: RVM은 사용자가 삽입한 용기를 자동으로 분류하고 분리할 수 있다. 이를 통해 재활용 가능한 재료와 그렇지 않은 부분을 효과적으로 분리하여 처리할 수 있다.
- 3) 보증금 처리: 사용자가 RVM에 용기를 삽입하면, 시스템은 해당 용기의 보증금을 계산하고 즉시 현금 또는 할인 혜택으로 제공한다. 이는 소비자들이 보상을 받으면서 재활용에 기여하도록 유도하는데 중요한 역할을 한다.
- 4) 데이터 관리와 분석: TOMRA의 시스템은 사용 데이터를 실시간으로 수집하고 분석함으로써 재활용 프로세스를 최적화하고 환경 성과를 평가할 수 있다.
- 5) 유지보수 및 원격 관리: TOMRA는 원격으로 RVM을 모니터링하고 유지보수할 수 있는 기능을 제공한다. 이는 효율적인 운영과 기기의 장애 대응을 가능하게 한다.

사용자들은 음료수 병이나 캔을 RVM에 삽입하면, 기계는 자동으로 해당 용기의 종류를 인식하고 보증금을 영수증 형태로 지불한다. 이를 통해 일회용 용기를 효과적으로 수거하고 재활용에 참여하도록 유도한다.

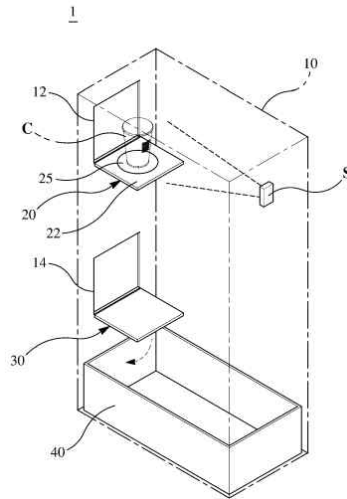
### 3.2.2. 특허조사

#### 3.2.2.1. 일회용 컵 수거 장치

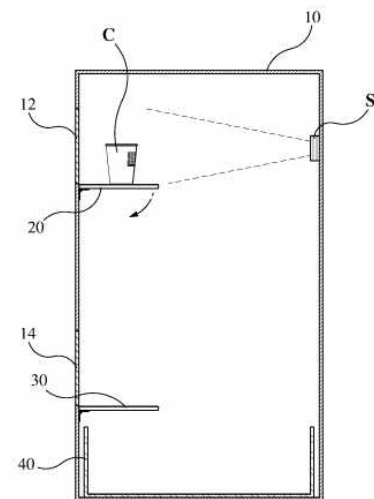
도면1



도면2



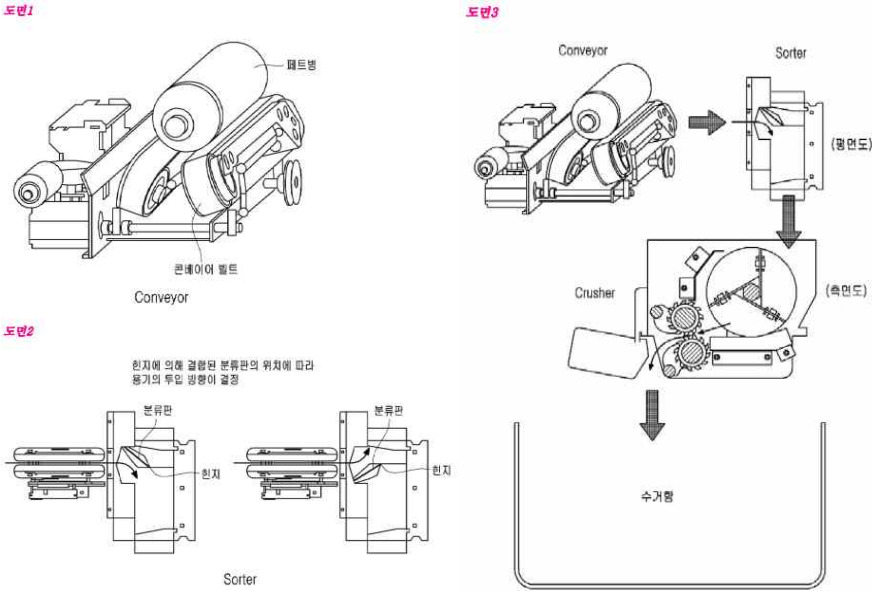
도면3a



명칭	일회용 컵 수거 장치
등록번호	10-2438220
등록일자	2022.08.25
법적상태	등록

- 사용자 정보가 담긴 QR코드가 구비된 컵
- 컵의 QR코드 인식과 컵의 모양에 대한 감지정보를 획득하는 감지모듈
- 컵 내부 내용물을 분리치 않고 버리는 경우에 사용자에게 컵과 잔유물을 분리할 수 있도록 유도
- 일회용 컵 수거 장치가 설치된 곳에서 보증금 환급이 가능
- 매장 직원은 일회용 컵을 일일이 수거하지 않기에 매장 직원의 수작업에 의한 분리 등 추가적인 업무 발생 없음

### 3.2.2.2. 재활용 용기 분류 및 압축 장치



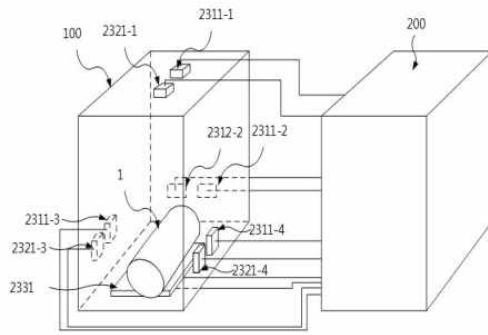
명칭	재활용 용기 분류 및 압축 장치
등록번호	10-2214151
등록일자	2021.02.03
법적상태	등록

- 재활용 용기를 영상 정보 분류 방식에 따라 분류
- 재활용 용기를 압축하는 압축기
- 재활용 용기를 저장하는 수거함
- 이송 롤러와 이송 플레이트는 재활용 용기의 영상 정보 분류 결과에 따라 사전 설정된 모드로 작동되어 재활용 용기가 분류 별로 사전 결정된 위치에 이르도록 함

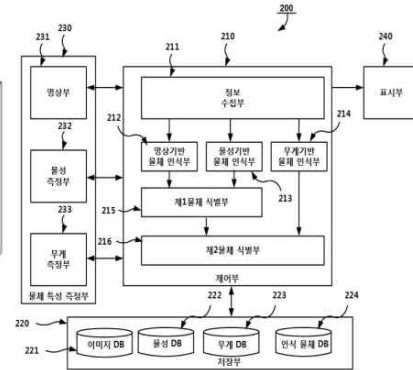


### 3.2.2.3. 인공지능망 분석에 근거한 복합적 물체 인식 시스템 및 방법

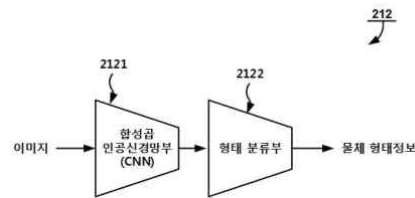
도면1



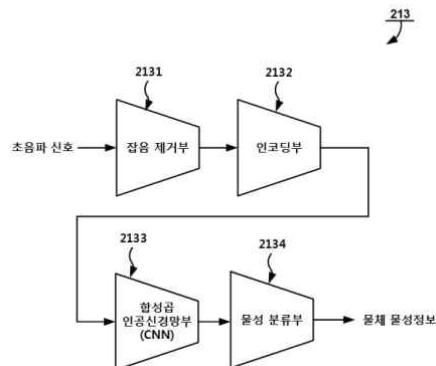
도면2



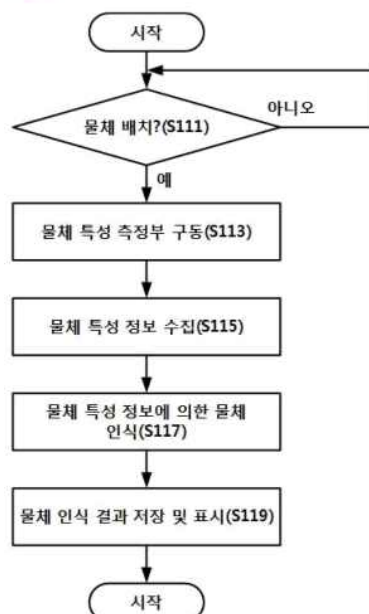
도면3



도면4



도면5



명칭	인공지능망 분석에 근거한 복합적 물체 인식 시스템 및 방법
등록번호	10-1779782
등록일자	2017.09.13
법적상태	등록

- 물체의 이미지, 물체의 주파수 특성 등과 같은 복합적 특성에 따라 단순하게 획득되는 로우레벨 정보를 인공지능망 분석에 적용
- 물체를 인식할 수 있는 하이레벨 정보로 변환된 형태정보, 물성정보 등을 획득
- 위의 하이레벨 정보 및 무게정보를 포함하는 물체의 복합적 특성들에 의해 물체를 인식

### 3.3. 과제추진의 필요성

#### 3.3.1. 증가하는 플라스틱 폐기물



표 - 플라스틱 폐기물 발생량 / 그린피스greenpeace.org 제공

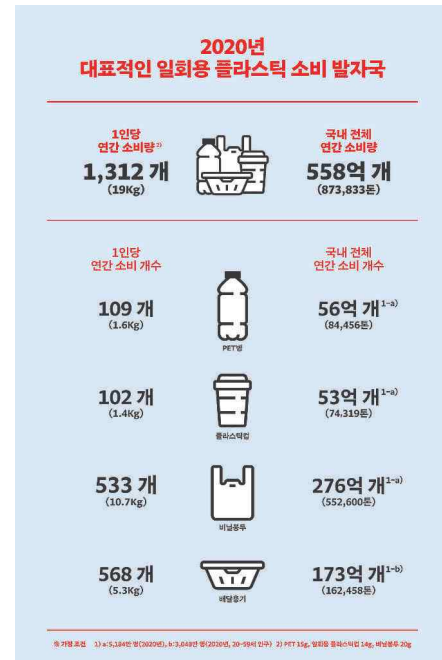


그림 - 일회용 플라스틱 사용량 / 그린피스 greenpeace.org 제공

- 코로나19 펜데믹 이후 국내 일회용 플라스틱 폐기물 발생량이 역대 최대 규모로 늘어났다는 조사 결과(그린피스 - ‘플라스틱 대한민국 2.0 보고서’)
- 지난 2021년에만 총 1193만 2천 톤의 플라스틱 폐기물 발생. 2017년에 비해 49.5%(395만1천 톤) 증가함
- 2021년 국내 플라스틱 물질 재활용률은 약 27%, 생활계 폐기물의 재활용률은 약 16.4%에 불과
- 증가량에 비하면 매우 낮은 수치라 판단
- 생활계 플라스틱 폐기물 70% 이상은 단순 소각하거나 에너지 회수 고형연료 형태로 처리하면서 온실가스를 대기 환경으로 대량 배출됨을 확인됨

기사 - [보도자료] 코로나 기간 ‘일회용 플라스틱 폐기물’, 역대 최대 규모로 증가 / 그린피스(greenpeace.org) / 2023년03월23일

### 3.3.2. 한국과 독일의 플라스틱 재활용 비교

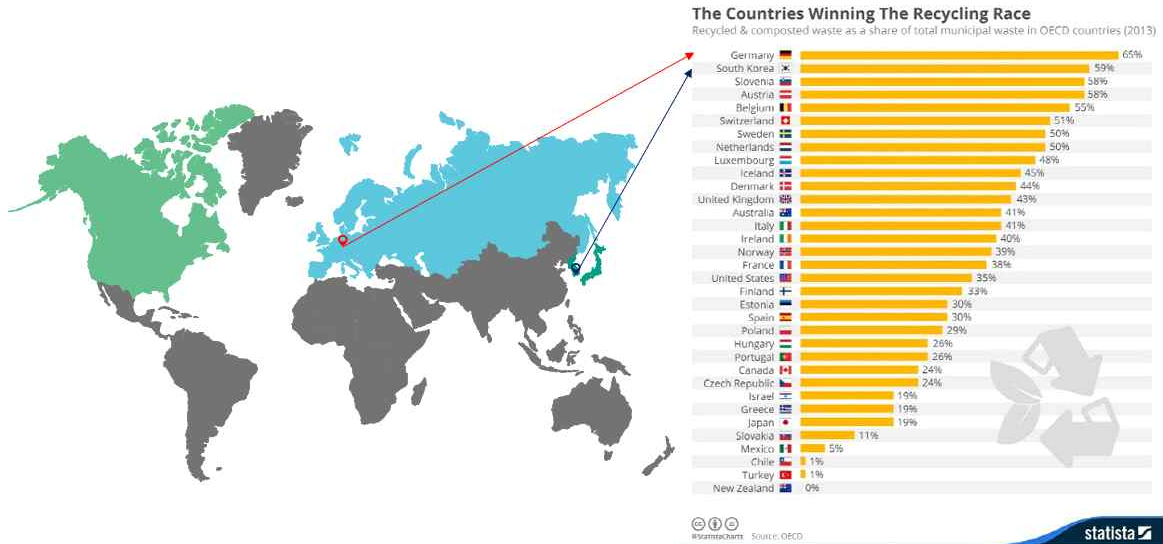
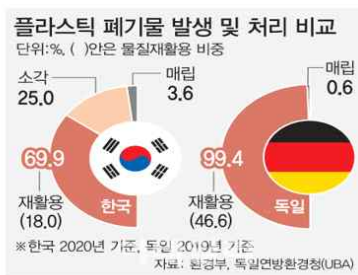


그림 - 세계적인 폐기물 재활용률(statista)

## 1. 한국-독일 차이점



한국 - 독일 간 플라스틱 폐기물 발생은 비슷하나, 재활용률에 있어 많은 차이를 보임

## 2. 플라스틱 폐기물 발생량



국내 플라스틱 폐기물 발생 증가



한국, 독일은 플라스틱 폐기물 발생량 상위권

그림 - 한국과 독일의 플라스틱 재활용 차이 및 플라스틱 폐기물 발생량 증가

독일 포장시장연구협회에 따르면 2019년 한 해 동안 재활용 가능 페트병으로 분류된 폐기물 중에서 판트로 수거된 페트병의 재활용률은 97.4%  
반면 그린피스 서울사무소 보고서에 따르면 같은 해 조사된 한국의 플라스틱 폐기물 재활용률은 EU 기준으로 집계했을 때 22.7%에 불과

신정아 / 독일, 한 해 판트(Pfand)로 수거된 페트병 재활용률 97.4% / 경제정보센터

플라스틱 폐기물의 분리수거 비율은 같은 해 기준 69.1%로 높은 편이었지만, 실제 재활용률은 14%에 그쳤고, 분리 배출한 플라스틱 폐기물 중 86%는 매립(62%)되거나 소각(24%)됐다. 이처럼 플라스틱 재활용률이 낮은 이유는 △플라스틱 재질 차이로 인한 재활용의 어려움, △이물질 오염으로 인한 재활용 불가, △잘못된 분리 배출 등이 원인으로 꼽혔다.

방사선 이야기 2 / 한국원자력환경공단

기사 - 판트 제도가 시행되는 독일과 대한민국의 플라스틱 재활용 현황

### 3.3.3. 본 프로젝트의 필요성

위 자료들을 토대로 본 프로젝트의 필요성을 설명할 수 있다. 우선, 국내 뿐만 아니라 전 세계적으로 증가하는 플라스틱 쓰레기들은 코로나 19 팬데믹 이후로 더욱 가속화되었을 뿐만 아니라, 앞으로도 계속 증가할 것이라는 여러 연구기관 및 단체의 예상이 정론으로, 이는 결국 현대 사회의 가장 큰 문제인 '환경 보호' 및 '지구 온난화 가속화'에 악영향을 끼칠 것이다.

국내와 독일의 플라스틱 쓰레기 발생률은 세계 최상위권이다. 그렇지만, 두 국가의 플라스틱 폐기물의 재활용률을 비교하면 발생률에 비하여 재활용률은 큰 차이를 확인할 수 있다. 이는 결국 국내에서 플라스틱 폐기물을 재활용하도록 유도하는 환경이 조성되어야 한다는 필요가 있다.

플라스틱 플라스틱은 제조, 공정 과정에서도, 그것들을 폐기하는 과정에서도 온실가스를 방출하는데, 사회 및 환경적으로 플라스틱의 재활용을 도모하는 장치가 존재한다면, 플라스틱의 재활용률을 늘려 결국 이는 플라스틱의 제조, 공정과 폐기되는 양을 줄일 것을 기대한다.

본 프로젝트에서는 현실적인 측면을 고려하여 플라스틱 재활용률을 증가하는 환경을 조성하는 것을 목적으로 장치를 설계한다. 사용자가 카페에서 사용하는 일회용 컵을 직접 재활용 가능한 형태로 분리배출하는 것을 도모하여 '환경 보호'라는 가치에 적합한 프로세스를 만들어 낼 수 있다.

## 4. 프로젝트 목표와 기준

본 프로젝트에서 달성하고자 하는 성능 목표는 다음과 같으며, 이외에도 본 과제에서 아래와 같은 현실적 제한 요소들을 달성한다.

- 1) JT(Linux Raspbian), AP(Windows)서버, DB서버(Linux Cent OS7)로 시스템 구성
  - $JT \leftrightarrow AP$  서버 간 WLAN 통신 (방화벽 특정 포트 개방)
  - $AP \leftrightarrow DB$  서버 간 NAT 통신 ( $Host\ OS(AP; Windows) \leftrightarrow Guest\ OS(DB; Cent\ OS7)$ )
- 2) AP서버에서 Node.js 기반 WEB서버 동작, DB서버에서 사용자 정보 MySQL 테이블링하여 저장
- 3) AP서버 접속을 위한 사용자 안드로이드 어플리케이션 개발
- 4) Raspberry Pi 기반의 JT로 센서 및 장치 제어하여 일회용 플라스틱 컵 수거
  - Raspberry Pi GPIO Port 사용하여 센서 및 장치 제어
  - 카메라 모듈 제어하여 투입된 물체 사진 캡처 후 AP 서버로 전송
- 5) 수거된 일회용 플라스틱 컵 청결도 검사를 위한 인공신경망 모델 구현
  - AP서버에서 인공신경망 모델을 이용하여 수신받은 물체 사진 식별 후 JT로 식별값 리턴(송신)
- 6) 수거 완료 시 해당 사용자에게 포인트 적립 (DB 상 저장됨)

현실적 제한 요소	내 용 (Content)	해결 방안 (Solution)
경제	<ul style="list-style-type: none"> <li>- 분리수거 비용과 사용자 보상금 책정 간 밸런스 유지</li> <li>- 하드웨어 리소스 최대한 적게 사용</li> </ul>	<ul style="list-style-type: none"> <li>- 사용자를 고려한 보증금 책정, 스탬프 방식</li> <li>- 소프트웨어에서 센서/장치 데이터 처리하여 하드웨어 비용 절감</li> </ul>
편리	<ul style="list-style-type: none"> <li>- 기기 주변 환경/접근성이 사용자가 편리하여야 함.</li> <li>- 분리수거함의 디자인(색과 소재, 모양)이 사용자의 이용률을 결정할 수 있음.</li> </ul>	<ul style="list-style-type: none"> <li>- 교내 / 주변 카페와 연계하는 방안 추진</li> <li>- 분리수거 함 외관 하드웨어 디자인 고려, 작은 사이즈</li> </ul>
윤리	<ul style="list-style-type: none"> <li>- 적립 방식을 이용한 악용(이미 분리수거 되어진 컵 계속하여 투입)</li> </ul>	<ul style="list-style-type: none"> <li>- 적립 횟수 제한 및 규격화된 컵만을 수거</li> </ul>
사회	<ul style="list-style-type: none"> <li>- 플라스틱 폐기물 발생에 따른 재활용률 상승의 필요성</li> </ul>	<ul style="list-style-type: none"> <li>- 분리수거 보상 제도를 통한 재활용률 향상</li> </ul>

표 - 본 설계과제의 현실적 제한요소 항목

#### 4.1. 원가 분석

장치	비용(원)
라즈베리 파이 4B	70,000
무게 센서	39,880
조도 센서	1,920
카메라 모듈	8,923
LCD 디스플레이	85,000
스테인리스 구조물	외부 지원

필요 총 비용 약 205,723원 예상

#### 4.2. 작품 외관

- 1) 장치의 외형은 스테인리스(STS304 1.2T POLISHING) 재질, W450\*D150\*H300mm 크기로 제작
- 2) 장치의 외부에는 7인치 터치스크린으로 사용자와 상호작용
- 3) 장치 내부에는 QR코드 인식 웹캠, 모터 2개, 카메라 모듈, 투입구, 로드셀, LED를 통해 각각의 기능을 수행

## 5. 설계과정

### 5.1. 설계 기초이론

#### 5.1.1. Raspberry Pi

라즈베리 파이(영어: Raspberry Pi)는 영국 잉글랜드의 라즈베리 파이 재단이 학교와 개발도상국에서 기초 컴퓨터 과학의 교육을 증진시키기 위해 개발한 신용카드 크기의 싱글 보드 컴퓨터이다. 라즈베리 파이는 모두 동일한 비디오코어 IV GPU와, 싱글코어 ARMv6에 호환되는 CPU 또는 신형의 ARMv7에 호환되는 쿼드코어(라즈베리 파이 2), 1 GB의 RAM(라즈베리 파이 2), 512 MB(라즈베리 파이 1 B와 B+), 또는 256 MB(모델 A와 A+, 구형 모델 B)의 메모리를 포함한다. 이들은 SD 카드 슬롯 (모델 A 와 B) 또는 부팅 가능한 매체와 지속적인 정보 저장을 위한 마이크로SDHC를 갖추고 있다. 라즈베리 파이는 리눅스 커널 기반 운영 체제를 사용한다. Raspbian이라는 라즈베리 파이에 최적화된 데비안 계열의 자유 운영 체제가 현재로서는 가장 권장되는 시스템이다.



그림 - Raspberry Pi

#### 5.1.2. WLAN

무선랜(無線 LAN, wireless LAN)은 무선 신호 전달 방식(일반적으로 확산 대역 또는 직교주파수분할다중화 방식)을 이용하여 두 대 이상의 장치를 연결하는 기술이다. 이를 이용해 사용자는 근거리 지역에서 이동하면서도 지속적으로 네트워크에 접근할 수 있다. 오늘날 대부분의 무선랜 기술은 IEEE 802.11 표준에 기반하고 있으며, 와이파이라는 마케팅 네임으로 잘 알려져 있다. 무선랜은 한때 미국 국방부에 의해 LAWN (Local Area Wireless Network)라고 불리기도 했다. 주로 편의, 비용 절감, 다른 네트워크와의 통합의 용이가 입증되면서 무선랜은 인기를 끌게 되었다. 오늘날 소비자에게 파는 대부분의 컴퓨터들은 필수적인 모든 무선랜 기술이 미리 장착되어 출시된다.

#### 5.1.3. NAT

네트워크 주소 변환(영어: network address translation, 줄여서 NAT)은 컴퓨터 네트워킹에서 쓰이는 용어로서, IP 패킷의 TCP/UDP 포트 숫자와 소스 및 목적지의 IP 주소 등을 재기록하면서 라우터를 통해 네트워크 트래픽을 주고 받는 기술을 말한다. NAT는 IPv4의 'IPv4 주소 고갈' 및 '라우팅 테이블 대형화 (Routing Scalability)'에 대한 해소책으로써 공인 IP 주소 사용시 ISP社를 바꿀 때 마다, 모든 컴퓨터 주소를 바꿔야하는 등의 단점을 해소하여 공인 IP 주소의 효율적 공유 및 절약이 가능하다. 또한 내부 사설망의 보안 및 부하 분산을 위해 주소 변환을 통한 기능이 사용된다. 이



는 주로 외부 네트워크와 내부 사설망 간의 연결 지점에 위치한 라우터에서 수행된다. 주소 변환은 내부 망의 정보를 외부에 노출시키지 않으며, 방화벽과 결합하여 프라이버시를 보호하고 정보를 은폐한다. 내/외 주소 변환 규칙은 외부에서 알 수 없어, 내부 망 주소 및 설정이 안전하게 숨겨져 있다. 이를 통해 내부 사설망은 안전하게 운영되며, 외부로의 노출을 최소화하여 보안성을 유지한다.

#### 5.1.4. CNN

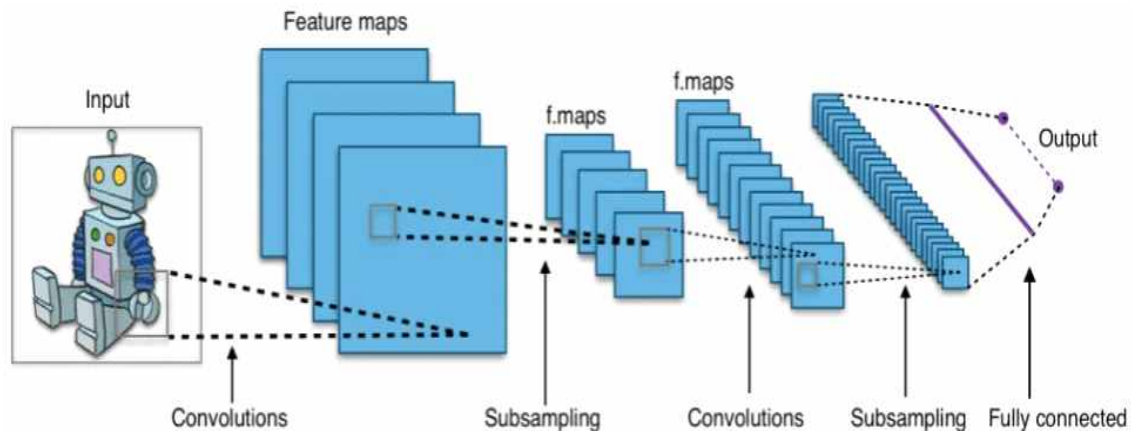


그림 - CNN 방식

합성곱 신경망(콘볼루션 신경망, Convolutional neural network, CNN)은 시각적 영상을 분석하는 데 사용되는 다층의 피드-포워드적인 인공신경망의 한 종류이다. 필터링 기법을 인공신경망에 적용하여 이미지를 효과적으로 처리할 수 있는 심층 신경망 기법으로 행렬로 표현된 필터의 각 요소가 데이터 처리에 적합하도록 자동으로 학습되는 과정을 통해 이미지를 분류하는 기법이다. 합성곱 신경망은 정규화 된 버전의 다층 퍼셉트론이다. 다층 퍼셉트론은 일반적으로 완전히 연결된 네트워크, 즉 한 계층의 각 뉴런이 다음 계층의 모든 뉴런에 연결되는 신경망 구조이다. 이와 같이 네트워크가 완전 연결된 경우 주어진 데이터에 과적합 되는 경향이 있다. 일반적인 정규화를 위해 최적화 함수에 특정 척도를 추가하는 방법이 흔히 쓰이지만, CNN은 정규화를 위한 다른 접근 방식을 취한다. 데이터에서 계층적 패턴을 활용하고 더 작고 간단한 패턴을 사용하여 더 복잡한 패턴을 표현함으로써 정규화와 같은 효과를 내는 것이다. 따라서 합성곱 신경망의 연결 구조의 복잡성은 유사한 기능의 다층 퍼셉트론에 비해 극단적으로 낮다. 합성곱 신경망은 크게 합성곱층(Convolution layer)과 풀링층(통합 계층, Pooling layer), 완전하게 연결된 층(fully connected layer)들로 구성된다. 그리고 분류연산이라는 행렬 곱 연산을 준비한다. 우선 기존 사진의 정보를 가진 신경망에 합성곱층과 풀링층을 완전하게 연결된 층 이전에 추가하여 원본 이미지에 필터링 기법을 적용한다. 이어 해당 이미지에 분류연산을 수행시킨다. 이를 통해 합성곱층이 이미지에 분류연산이 하고자 하는 필터를 씌울 수 있다. 여기서 다시 풀링 계층은 각 이미지의 국소적인 부분을 하나의 대표적인 스칼라 값으로 변환하여 행렬 곱 연산으로 인해 커진 이미지 크기를 줄이는 등의 기능을 수행한다.

### 5.1.5. QR Code

QR 코드(영어: QR code, Quick Response code, 순화어: 정보무늬)는 컴퓨터로 만든 흑백 격자무늬 코드로, 정보를 나타내는 매트릭스 형식의 이차원 코드이다. QR코드는 주로 한국, 일본, 중국, 영국, 미국 등에서 많이 사용되며 명칭은 텐소 웨이브의 등록 상표 'Quick Response'에서 유래하였다. 종래에 많이 쓰이던 바코드의 용량 제한을 극복하고 그 형식과 내용을 확장한 2차원의 패턴으로 중형의 정보를 가져서 숫자 외에 문자의 데이터를 저장할 수 있다. 보통 디지털카메라나 전용 스캐너로 읽어 들여 활용한다. 국립국어원에서는 QR 코드를 정보무늬로 다듬었다. 비슷한 용도로 먼저 사용된 이차원 코드로는 바코드가 있다. 바코드는 이름 그대로 단순한 막대기 모양의 바를 이차원으로 나열한 것이다

### 5.1.6. Servo Motor

서보모터(Servo Motor)는 범용 기계와 비교해 보면 핸들을 돌리는 손에 해당하는 부분으로 머리에 해당되는 정보처리회로(CPU)의 명령에 따라 동작기계 테이블 등을 움직이게 하는 모터이다. 물체의 위치·방위·자세·회전 속도 등을 제어량으로 하고 목표치의 변화에 뒤따르도록 구성된 자동제어계를 서보기구(servo機構)라하며, 보통 피드백(feed back) 회로를 가지고 있어, 출력의 검출부·목표치와의 오차 증폭부·조작부·제어 대상 등으로 구성된다. 이러한 서보계의 조작부에 사용되는 것이 서보모터인데, 전기식(직류·교류)·유압식·전기유압식으로 된 것들이 있다.



그림 - Servo Motor

### 5.1.7. Load Cell

로드셀(Load Cell)이란 하중계나 하중센서 또는 힘센서라고도 불린다. 힘 또는 하중을 측정하기 위한 변환기로서, 출력을 전기적으로 꺼낼 수 있는 것을 말한다. 측정 대상으로서 하중(힘)을 전기 신호로 변환하기 쉬운 양, 예컨대 변위나 변형으로 변환하는 1차 변환요소와, 1차 변환출력(변위/변형)을 전기신호로서 출력 가능한 양, 예컨대 인덕턴스 변화, 용량변화, 저항변화 등으로 변환하는 2차 변환요소로 구성된다. 표는 로드 셀의 변환방식의 조합과 그 특징을 나타낸 것이다. 변위형은 큰 변위를 취할 수 있기 때문에 일반적으로 낮은 하중(1kg 이하)을 정밀하게 검출할 수 있다. 변형형은 1차 변환요소와 2차 변환요소가 일체적으로 되어 소형화하기가 운 측면이 있다.



그림 - Load Cell



### 5.1.8. 클라우드 컴퓨팅 환경

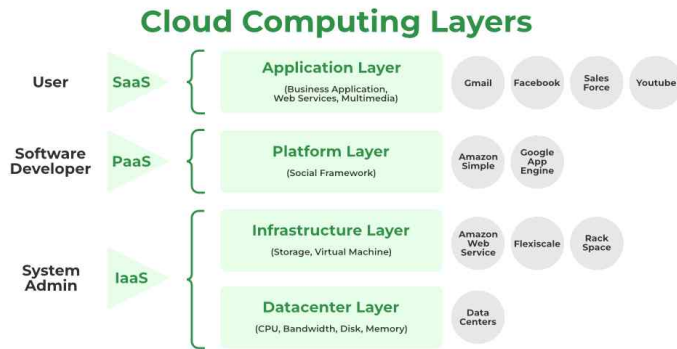


그림 - 클라우드 컴퓨팅 환경

클라우드 컴퓨팅 환경이란 기존 하드웨어적으로 구성된 컴퓨팅 리소스(ex. cpu, HDD, 더 나아가서는 Server 그 자체 등)를 인터넷을 통해 제공하는 기술로, 사용자는 인터넷을 통해 IT 리소스를 제공받고 사용한 만큼의 비용을 지불하는 것을 의미한다. 이러한 클라우드 컴퓨팅에는 사용자의 요구에 따라 IT 리소스를 유동적으로 할당하고 조정하므로, 제공하는 IT 리소스의 종류에 따라 IaaS(Infrastructure as a Service), PaaS(Platform as a Service), SaaS(Software as a Service) 등 다양한 서비스가 존재한다. 클라우드 서비스의 최상단 계층인 Application Layer에서는 사용자에게 직접적으로 서비스를 제공하며 SaaS가 해당 계층에 포함된다. Platform Layer에서는 공급자가 제공하는 플랫폼을 이용하여 애플리케이션과 서비스를 개발하는 영역으로 PaaS가 해당 계층에 포함된다. Infrastructure Layer에서는 사용자가 가상머신, 네트워크 등의 IT 리소스를 할당 받아 사용하는 계층으로 IaaS가 해당 계층에 포함된다. 클라우드 아키텍처의 최하단 계층으로는 Datacenter Layer로 실제 제공하는 클라우드 컴퓨팅의 하드웨어 및 리소스를 포함한다. 해당 계층은 클라우드 서비스의 인프라를 형성하는 기반이 된다.

### 5.1.9. Linux CentOS7

CentOS 7(Community ENTERprise Operating System 7)은 Red Hat Enterprise Linux(RHEL)의 소스 코드를 기반으로 하는 무료 오픈 소스 운영 체제이다. 주로 서버 환경에서 사용되며, 안정성과 보안성, 장기 지원(LTS)으로 서비스의 기반 운영 체제로 본 프로젝트에서는 다음과 같이 활용된다. CentOS 7은 YUM(Yellowdog Updater, Modified) 패키지 관리 시스템을 사용하여 사용자는 손쉽게 소프트웨어를 설치, 업데이트, 삭제할 수 있으며, 의존성 문제를 자동으로 해결할 수 있기에 서버 관리자의 편의성을 증대시킨다. 본 프로젝트에서 CentOS7은 DB서버의 OS로서 활용되어지는데 YUM 패키지 관리 시스템을 통해 MySQL을 설치하여 별다른 DBMS의 별도 구성없이 패키지를 다운하여 사용한다. 또한, CentOS7은 장기 지원 버전으로 장기간 보안 업데이트를 제공하며, SELinux(Security-Enhanced Linux)를 기본적으로 활성화하여 시스템 보안을 강화한다. 이는 다양한 최신의 보안 정책을 적용할 수 있게 하여, 시스템을 더 안전하게 보호하는 측면의 장점을 가져 DB서버를 운용함에 있어 데이터의 무결성과

기밀성을 지켜 더욱 안정성 있는 DB서버 구축/관리를 가능케 한다.

#### 5.1.10. 데이터베이스 - DBMS(MySQL)

데이터베이스는 구조화된 정보 또는 데이터의 조직화된 모음(통합데이터, 저장 데이터, 운영 데이터, 공유 데이터)으로서 일반적으로 컴퓨터 시스템에 전자적으로 저장되어 데이터베이스 관리 시스템(DBMS)에 의해 액세스, 관리, 수정, 업데이트, 제어 및 구성된다. 이와 같은 하나의 논리적 기능을 수행하는 작업의 단위를 트랜잭션이라고 하며, 트랜잭션은 다음과 같은 ACID 성질을 갖는다. MySQL은 관계형 데이터베이스 관리 시스템으로, 데이터를 연결된 목록으로 레코드를 저장하는 방식과 달리 열과 행이 있는 테이블 형태로 레코드를 저장, 관리, 검색한다. 본 프로젝트에서는 DBMS로 MySQL을 사용하며, MySQL의 스키마(Schema)는 논리적인 구조를 정의하는 데이터베이스의 단위로, CREATE SCHEMA 쿼리문을 이용하여 생성 시 이 안에 데이터베이스 테이블을 생성할 수 있다. 테이블(Table)은 데이터 특성을 기준으로 구조화하여 데이터를 저장하는 단위로, 테이블의 열은 데이터 속성을 나타내고, 행은 실제 데이터를 저장한다.

#### 5.1.11. AP - DB 서버 이원화

AP서버는 네트워크가 연결되어 있다면, 네트워크를 통해 서버와 EndPoint 간의 통신을 할 수 있는 Server로 Http, TCP, UDP 등 다양한 프로토콜을 전달받아 클라이언트에게 다양한 서비스를 제공한다. (DB서버는 5.1.9 참조) 본 프로젝트에서는 AP서버가 모든 노드의 중심이 되어 상호 간의 통신 매개체가 되며, CNN 모델을 통해 컵의 청결도를 검사하는 서비스와 웹서버로서의 역할을 수행한다. 이 때 AP - DB서버의 이원화를 구현하면 통신 매개체로서 AP서버는 클라이언트가 DB서버로의 직접 접근을 막도록(클라이언트 - AP서버(통신 매개체) - DB서버)하여 데이터의 기밀성을 유지할 수 있도록 도와주어 더 높은 보안성을 유지할 수 있게 하는 노드로서 적용되어진다. 한편, 웹서버 가동과 CNN모델에 대한 결과값 도출은 많은 리소스를 사용하는 작업으로, 이원화를 적용하면 AP서버와 DB서버가 각각의 자원을 독립적으로 사용하므로 성능 향상의 이점이 있다. 또한, 하나의 서버에 문제가 발생해도 다른 서버에 영향을 주지 않으므로 시스템의 전체적인 안정성이 높아지는 효과가 있다.

#### 5.1.12. 정보 보안 시스템

정보보호란 정보의 수집, 가공, 저장, 검색, 송신, 수신 중에 발생하는 정보의 훼손, 변조, 유출 등을 방지(기밀성 · 무결성 · 가용성 · 인증성 · 부인방지 보장)하기 위한 관리적, 물리적, 기술적 보호대책을 강구하는 것으로, 본 프로젝트에서는 데이터베이스에 저장된 데이터에 대한 기밀성 · 무결성을 보장하기 위해 Linux CentOS7 시스템에 대한 여러 가지 DB서버 보안 정책을 수립하고, 취약점 분석을 실시하여 발견된 취약점을 조치하여 보안성을 높였다. 본 프로젝트에서 여러 노드(JT기기, AP(Web)서버,

DB서버, 안드로이드 App)가 존재하여 상호 간 필요 서비스에 대한 통신이 수행되는데, 이 때 통신하는 서비스 간에 대해 특정 포트로만 데이터가 송·수신되도록하는 방화벽 정책을 설정하여 네트워크 보안성을 높였다.

## 5.2 개념설계

### 5.2.1. 전체 시스템 인프라 구성

#### 5.2.1.1. 전체 시스템 구성도

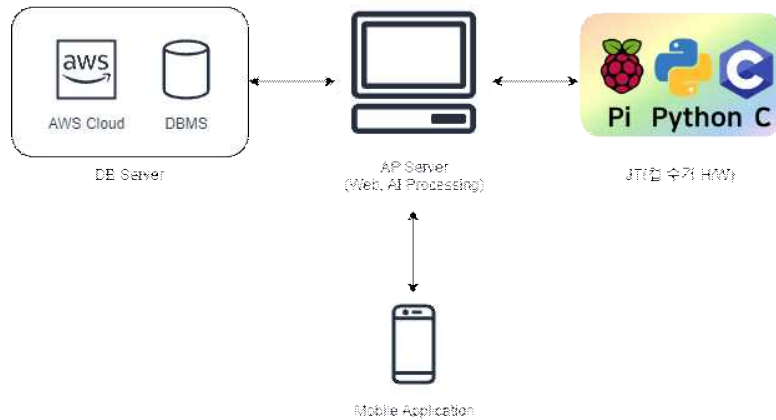


그림 - 전체 시스템 개념 구성도

안정성, 보안성, 성능 향상 등의 효율성 증대를 위해 각 기술 스택을 사용해 AP-DB 이원화 서버를 구축한다. 이 때, DB서버는 AWS 클라우드 컴퓨팅 서비스를 사용하고, 그 위에 서비스로 DBMS를 구축하여 사용한다. AP서버의 경우 웹서버와 AI Processing을 담당하기 위해 각각에 필요한 기술을 사용하여 구현한다. 사용자 편의를 위한 Mobile Application과 컴 수거를 위한 하드웨어인 JT는 각각 Client로서 AP Server와 통신하도록 설계한다.

#### 5.2.1.2. 기능적 요구사항 명세서

ID	요구사항	내용	설명	우선 순위
Sys01	AWS EC2 인스턴스 내 DBMS 세팅	DB서버를 구현하기 위해 AWS EC2 인스턴스 내 환경 구축	AWS EC2 인스턴스 내 Linux CentOS7 운영체제 구축 후 MySQL 서비스 실행 환경 구축	1
Sys02	AP-DB서버 이원화	AP 서버와 DB 서버를 분리하여 운영할 수 있도록 구축	AP-DB 이원화 서버를 구축해 3306포트로 Mysql 관련 데이터가 송·수신되어 동기화될 수 있도록 한다.	2
Sys03	정보보안 시스템 구축	시스템 보안 및 네트워크 보안 시스템 구축	각 노드별 시스템에 알맞는 보안정책을 적용하고, 노드 간 네트워크 보안정책 적용	3

## 5.2.2. Mobile Application

### 5.2.2.1. 기능블록도

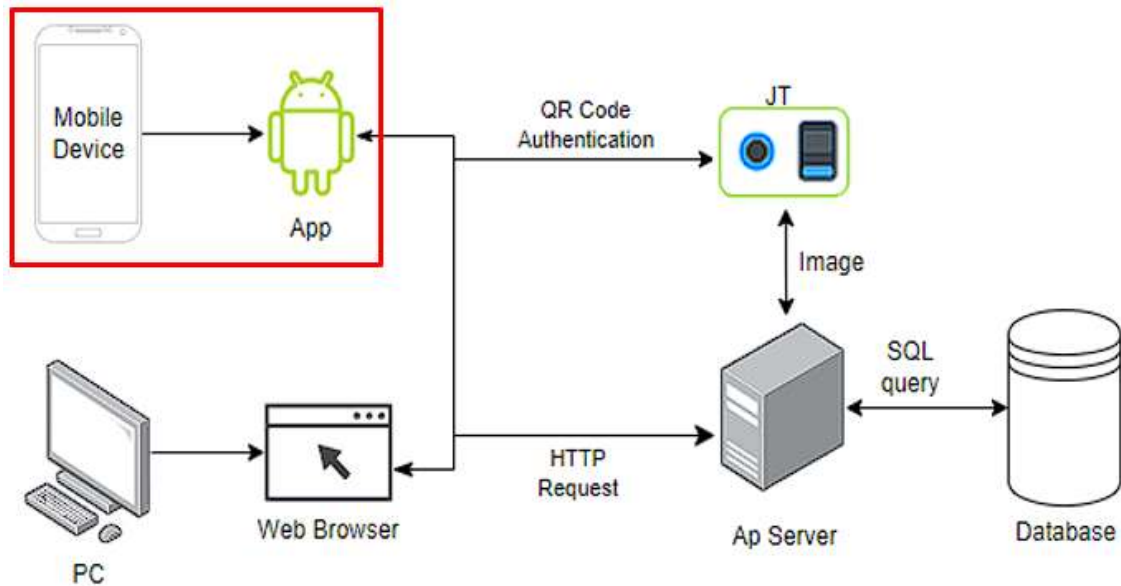


그림 - 전체 네트워크 구성도

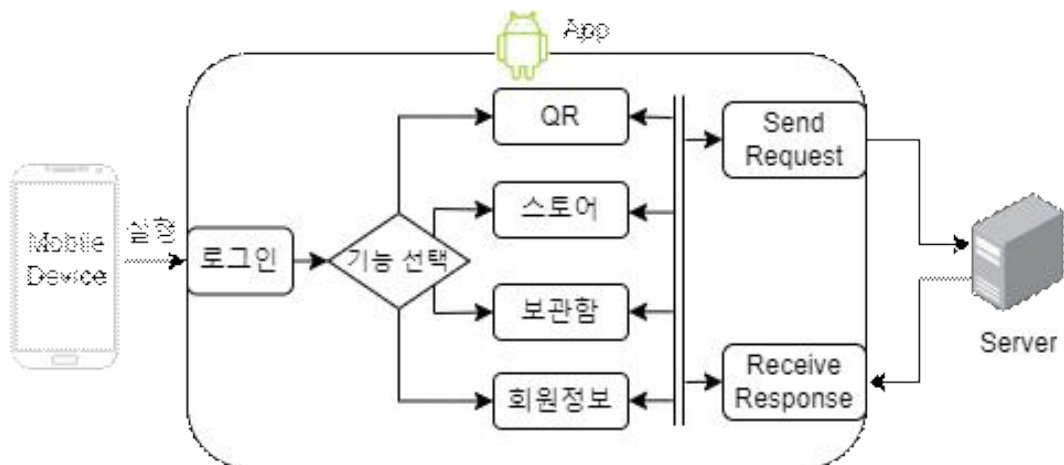


그림 - Mobile App 개념 기능 블록도

Kotlin언어 기반의 Mobile App을 개발한다. App은 Android기반의 핸드폰, 태블릿 등의 설치가 가능해야한다. 또한 Web Server과의 통신을 위해 네트워크에 연결되어 있어야한다. App은 HTTP 메서드를 사용하여 서버와 통신하고, 데이터 요청 및 응답을 처리할 수 있어야 한다. 사용자는 ID와 비밀번호를 이용해 회원가입을 할 수 있고 이를 통해 App에 로그인할 수 있어야한다. App과 서버가 연결이 되면 쿠키를 통해 세션을 유지하고, 유저가 로그인 상태를 유지하거나 로그아웃할 때 세션을 종료해야한다. 또한 사용자는 자신의 정보를 기반으로 생성된 QR 코드를 확인하고, 앱을 통해 QR 코드를 JT에 인식시킬 수 있어야 한다. 사용자는 플라스틱 폐기물을 반납한 후 스탬프/포인트를 획득해야 하고 자신의 스탬프나 포인트 잔액을 확인할 수 있어야 하며, 이를 사용하여 다양한 상품을 구매할 수 있어야 한다.

### 5.2.2.2. 기능적 요구사항 명세서

ID	요구사항	내용	설명	우선순위
App01	회원가입 및 로그인/로그아웃	사용자가 앱을 통해 회원가입을 하고 로그인 및 로그아웃할 수 있어야 한다.	사용자는 ID와 비밀번호를 이용해 회원가입을 할 수 있고 이를 통해 App에 로그인할 수 있어야 한다.	2
App02	세션관리	로그인 상태 유지를 위해 세션을 관리해야 한다.	쿠키를 통해 세션을 유지하고, 유저가 로그인 상태를 유지하거나 로그아웃할 때 세션을 종료해야 한다.	3
App03	QR 코드 생성 및 인식 기능	사용자의 정보에 따라 QR 코드를 생성하고, 이를 인식할 수 있어야 한다.	사용자는 자신의 정보를 기반으로 생성된 QR 코드를 확인하고, 앱을 통해 QR 코드를 JT에 인식시킬 수 있어야 한다.	4
App04	스탬프/포인트 관리 기능	사용자가 스탬프나 포인트를 확인하고 사용할 수 있어야 한다.	사용자는 자신의 스탬프나 포인트 잔액을 확인하고, 이를 사용하여 다양한 상품을 구매할 수 있어야 한다.	6
App05	회원 정보 확인 및 수정 기능	사용자가 자신의 회원 정보를 확인하고 수정할 수 있어야 한다.	사용자는 앱을 통해 자신의 프로필 정보, 스탬프/포인트 내역 등을 확인하고, 필요 시 정보를 수정할 수 있다.	5
App06	서버와의 데이터 통신	앱과 서버 간의 데이터를 주고받을 수 있어야 한다.	HTTP 메서드를 사용하여 서버와 통신하고, 데이터 요청 및 응답을 처리할 수 있어야 한다.	1
App07	App 사용 편의성	사용자가 앱을 사용할 때 불편함을 느끼지 않아야 한다.	자동 로그인 기능, 네비게이션 바, 사용자 친화적인 인터페이스 등을 추가한다.	7

### 5.2.3. AP Server

#### 5.2.3.1. 기능블록도

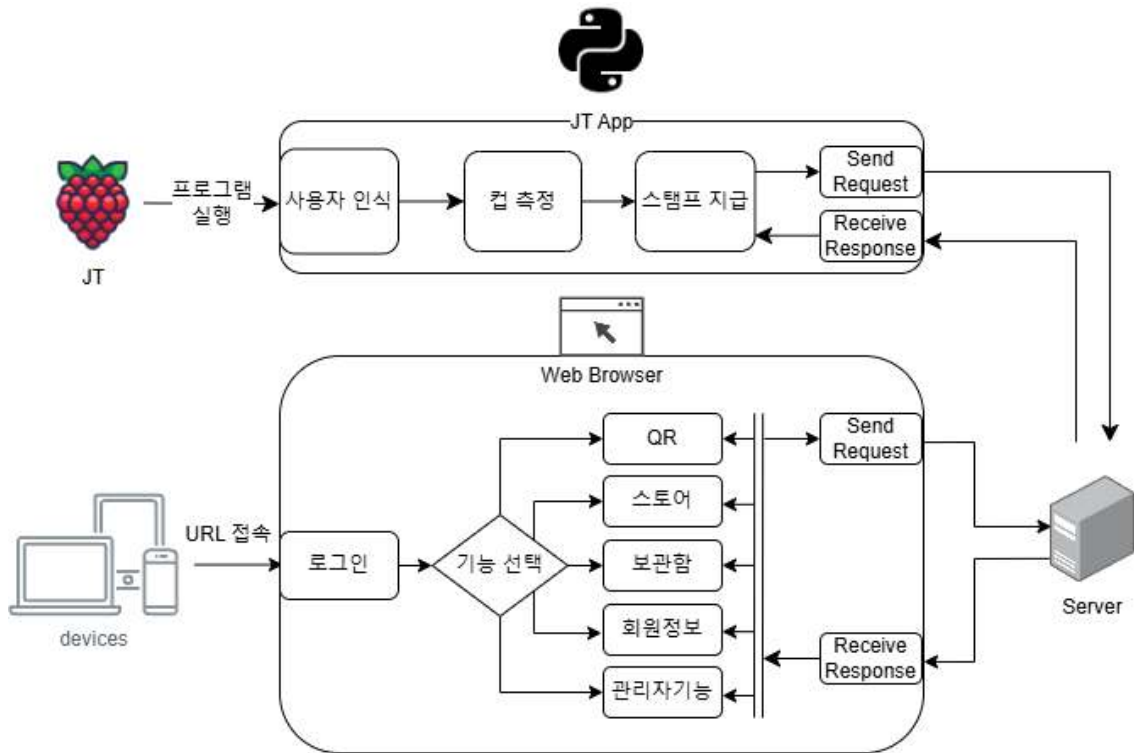


그림 - AP Server 개념 기능 블록도

Node.js 기반의 AP Server를 개발한다. AP 서버는 JT 기기 프로그램이 사용할 API와 사용자가 접속할 웹 서버를 제공해야 한다. JT 프로그램이 사용할 API는 사용자를 인식 API와 컵 사진의 통과여부를 머신러닝으로 측정하는 API를 필요로 한다. 웹 서버는 PC, 핸드폰, 태블릿 등 플랫폼 종류에 제약 없는 UI로 설계한다. 사용자는 웹 브라우저로 서버 URL에 접속한다. 회원가입하여 계정을 생성 후 로그인을 할 수 있어야 한다. 사용자는 자신의 정보를 기반으로 생성된 QR 코드를 확인하고, JT 기기에 인식시킬 수 있어야 한다. 사용자는 플라스틱 폐기물을 반납한 후 스탬프/포인트를 획득해야 하고 자신의 스탬프나 포인트 잔액을 확인할 수 있어야 하며, 이를 사용하여 다양한 상품을 구매할 수 있어야 한다.

### 5.2.3.2. 기능적 요구사항 명세서

ID	요구사항	내용	설명	우선 순위
AP01	QR 코드 유저 인증 API	JT 기기에서 사용자가 QR 코드를 요청받아 유저 정보를 응답해주는 API	JT 기기는 사용자가 인식시킨 QR코드 값을 담아 요청을 하고, 서버는 이 값으로 DB에서 유저 정보를 찾아 응답해야 한다.	1
AP02	플라스틱 컵 청결도 검사 API	JT 기기에서 촬영한 컵 사진을 요청받아 머신러닝 모델 판단 결과를 응답해 주는 API	JT 기기는 사용자가 투입한 일회용 플라스틱 컵의 사진을 촬영하여 서버에 요청하고, 서버는 머신러닝 모델로 컵의 청결도를 판단하여 결과값을 응답해야 한다.	2
AP03	웹 서버 구축	사용자가 접속 가능한 웹	사용자에게 서비스 제공을 위한 웹 서버를 구현해야 한다.	3
AP04	회원가입 및 로그인/로그아웃	사용자가 웹을 통해 회원가입을 하고 로그인 및 로그아웃할 수 있어야 한다.	사용자는 ID와 비밀번호를 이용해 회원가입과 로그인을 할 수 있어야 한다.	4
AP05	세션관리	로그인 상태 유지를 위해 세션을 관리해야 한다.	쿠키를 통해 세션을 유지하고, 유저가 로그인 상태를 유지하거나 로그아웃할 때 세션을 종료해야 한다.	5
AP06	QR 코드 생성 및 인식 기능	사용자의 정보에 따라 QR 코드를 생성하고, 이를 인식할 수 있어야 한다.	사용자는 자신의 정보를 기반으로 생성된 QR 코드를 확인하고, 웹을 통해 QR 코드를 JT에 인식시킬 수 있어야 한다.	6
AP07	스탬프/포인트 관리 기능	사용자가 스탬프나 포인트를 확인하고 사용할 수 있어야 한다.	사용자는 자신의 스탬프나 포인트 잔액을 확인하고, 이를 사용하여 다양한 상품을 구매할 수 있어야 한다.	7
AP08	상품 등록 기능	관리자가 스토어에 상품을 등록하는 기능	관리자 계정으로 스토어에 사진과 이름, 가격을 입력하여 상품을 등록할 수 있다.	8
AP09	회원 정보 확인 및 수정 기능	사용자가 자신의 회원 정보를 확인하고 수정할 수 있어야 한다.	사용자는 웹을 통해 자신의 프로필 정보, 스탬프/포인트 내역 등을 확인하고, 필요 시 정보를 수정할 수 있다.	9
AP10	비밀번호 찾기 기능	사용자가 인증코드를 통해 비밀번호를 재설정하는 기능	사용자는 아이디를 입력하여 계정에 저장된 이메일을 통해 인증코드를 받아 비밀번호를 재설정할 수 있다.	10



## 5.2.4. 플라스틱 컵 분류기

### 5.2.4.1. 기능블록도

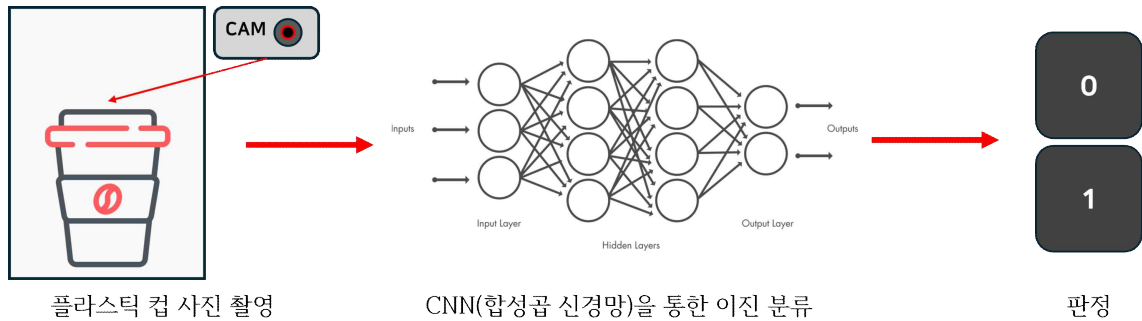


그림 - 플라스틱 컵 분류기 동작 과정

Python을 사용하여 플라스틱 컵 사진을 통해 더러운 컵을 분류하는 모델을 작성한다. 본 프로젝트에서 사진은 항상 정해진 위치(JT기기 내부)에서 정해진 조건으로 사진이 촬영되도록 제한되어 있기에 데이터셋도 촬영 환경에 맞춰 정량된 데이터로 모델을 학습시키며, 기기의 내부에 조명이 설치되는 것을 고려하여 데이터셋을 준비한다. 색이 분명한 음료 이외 투명한 음료 및 투명한 얼음이 담겨진 컵의 경우, 육안으로도 빈 컵인지를 분류하는 것이 어렵기 때문에 데이터셋 및 학습 과정에서의 하이퍼파라미터 튜닝을 통해 정확도를 높일 수 있도록 설계한다.

### 5.2.4.2. 기능적 요구사항 명세서

ID	요구사항	내용	설명	우선순위
ML01	촬영 환경 기준 설정	항상 일관된 조건에서 사진이 촬영될 수 있도록 한다.	모델의 정확도를 높이기 위해 사진 촬영의 각도, 배경, 조명, 크기, 해상도 등을 일관되게 촬영될 수 있도록 기준을 설정	1
ML02	적절한 데이터셋 확보	분류 목적을 달성할 수 있도록 적절한 데이터셋을 확보한다.	깨끗한 컵, 빈 컵에 대한 데이터셋과, 더러운 컵이나 이물질 및 음료, 얼음 등이 담겨있는 컵의 사진을 촬영, 선별, 가공	2
ML03	투명한 물질 분류	육안으로도 구별하기 힘든 물질들에 대해서 NG로 분류할 수 있도록 한다.	색이 분명한 음료 이외 투명한 음료 및 투명한 얼음이 담겨진 컵의 경우, 육안으로도 빈 컵인지를 분류하는 것이 어렵기 때문에 모델 학습 시 이를 분류하도록 튜닝	3

### 5.2.5. JT(플라스틱 컵 수거 장치) H/W, S/W

#### 5.2.5.1. 기능블록도

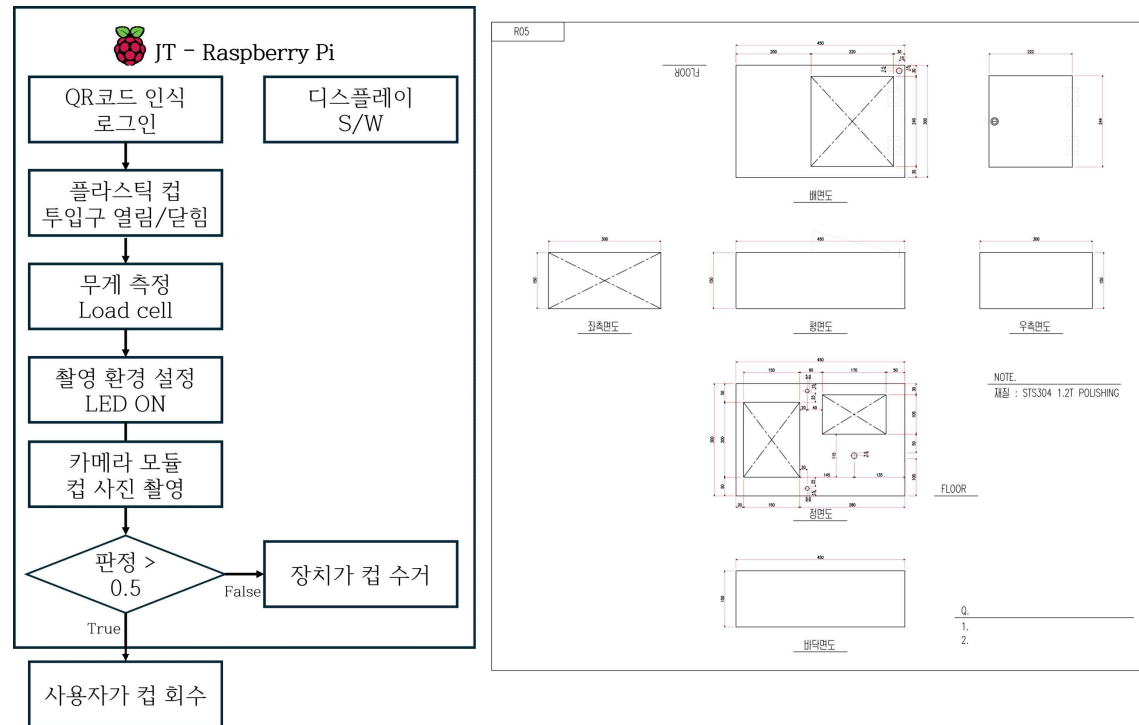


그림 - JT 기능블록도 및 외형 도면

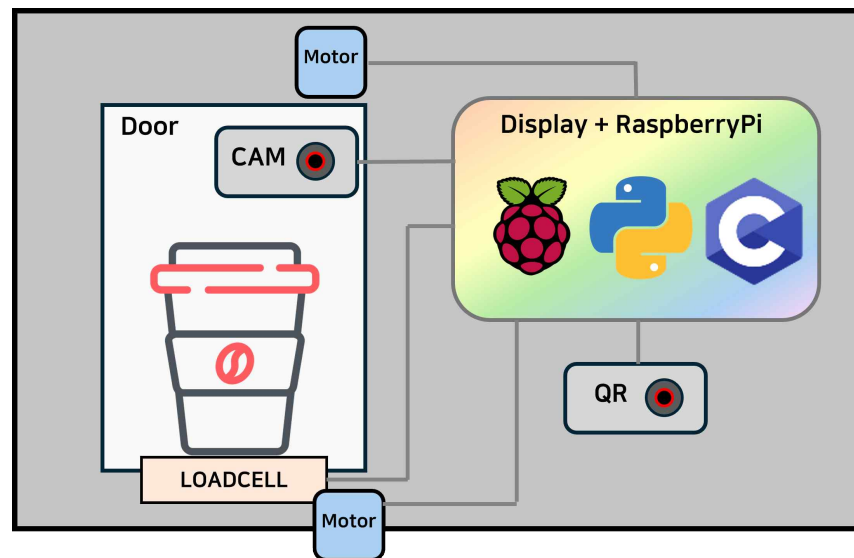


그림 - JT 모식도

Raspberry Pi를 기반으로 JT 장치를 설계한다. 장치의 외형은 스테인리스(STS304 1.2T POLISHING) 재질, W450\*D150\*H300mm 크기로 제작한다. 장치는 7인치 터치스크린, QR코드 인식 웹캠, 모터 2개, 카메라 모듈, 투입구, 로드셀, LED를 통해 각각의 기능을 수행한다. 결과적으로, 수거 과정 동안 JT와 Android APP, AP서버와 지속해서 통신하며 컵 수거에 관련된 기능들을 수행한다.

### 5.2.5.2. 기능적 요구사항 명세서

ID	요구사항	내용	설명	우선순위
JT01	JT S/W 구현	상황에 맞는 기능을 수행하도록 라즈베리파이를 제어, Android APP, AP서버와 통신 인터페이스 구축	사용자는 S/W 및 7인치 터치 디스플레이를 통해 장치와 상호작용 할 수 있으며 S/W는 라즈베리파이를 제어하여 각 센서들의 동작을 각각의 기능과 연결하며, APP과 AP서버와 실시간 통신한다.	1
JT02	QR코드 인식 기능	QR코드를 인식하면 올바른 QR인지 판독, 로그인을 수행한다.	OpenCV 라이브러리를 사용, Logitech C270 HD 웹캠으로 실시간 QR코드 인식하여 QR코드 데이터를 처리한다.	4
JT03	투입구 열림, 닫힘 기능	로그인 성공 시 문이 열리며 사용자로부터 컵을 전달받는다.	서보모터 2개를 사용해 문과 고정하여 문의 움직임을 제어한다. 로그인 성공 시 문이 열리며, 사용자가 컵을 넣고 닫기를 누르면 문이 닫힘.	6
JT04	컵 무게 측정 기능	컵의 무게를 측정하여 비어있는 컵인지 물리적으로 판별함.	투명한 물질(얼음, 물 등)은 육안으로도 플라스틱과 구별이 힘들기 때문에, 이를 구별하고자 물리적인 방법을 채택하였다.	7
JT05	촬영 환경 조성	LED를 틀어서 촬영 환경을 밝게 한다.	저조도 촬영 결과물은 판정 정확도가 낮기 때문에	2
JT06	컵 사진 촬영 기능	라즈베리파이의 카메라 모듈을 활용하여 컵 사진을 촬영한다.	최대한 일관된 환경에서 컵 사진이 촬영되도록 환경을 조성하여 서버로 사진을 전송해 판정값을 수신함.	3
JT07	판정 결과에 따른 컵 처리 기능	서버로부터 받은 판정 결과에 따라 컵을 처리한다.	판정 결과 0.5 초과일 경우 사용자가 컵을 직접 회수하도록 유도하며, 0.5 이하일 경우 기기에서 컵을 수거하도록 한다.	5

## 5.3 상세설계

### 5.3.1. 전체 시스템 인프라 구성

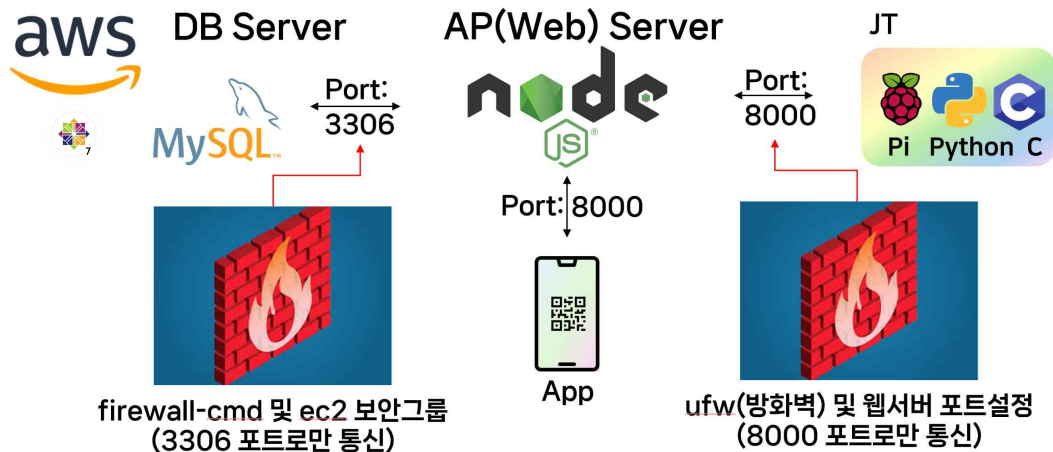


그림 - 전체 시스템 상세 구성도

안정성, 보안성, 성능 향상 등의 효율성 증대를 위해 각 기술 스택을 사용해 AP-DB 이원화 서버를 구축한다.

첫째, DB서버의 경우 보안성 및 안정성 등을 위해 클라우드 컴퓨팅 서비스인 AWS EC2 instance 위에 Linux CentOS7을 올려 AP서버와 이원화하고, 서비스로 MySQL DBMS를 사용한다.

둘째, AP서버의 경우 Client(Mobile App, JT, DB서버)와의 통신 및 웹서비스 제공을 담당하는 웹서버 기능을 Node.js(Back-End)와 HTML/CSS(Front-End)를 사용하여 구성하고, HTTP Protocol(tcp/8000)을 통해 JT로부터 이미지 파일을 전송받아 CNN 모델을 사용해 컵의 청결도를 JT에 반환한다.

셋째, AP-DB서버 간 데이터 동기화 및 통신을 위해 MySQL 전용 포트(tcp/3306)를 통해 MySQL 관련 데이터가 송·수신되게 한다.

넷째, 컵 수거를 위한 JT 하드웨어를 Raspberry Pi 기반으로 C언어와 Python을 사용해 설계하고, 사용자들의 편의를 위한 Mobile Application을 Kotlin을 사용하여 설계한다. 각각은 Client로서 AP서버(웹서버)와 통신한다.

또한, 보안성 향상을 위해 각각의 노드 간 통신에 사용되는 포트를 제외하고 방화벽 포트 정책을 Negative(All Deny)로 설정하며, 사용자 정보 등이 저장되는 DB서버와 DBMS(MySQL)에 대하여 추가적으로 시스템보안정책을 설정하고, 취약점 분석하여 해당 취약점에 대해 조치한다.

### 5.3.2. Mobile Application

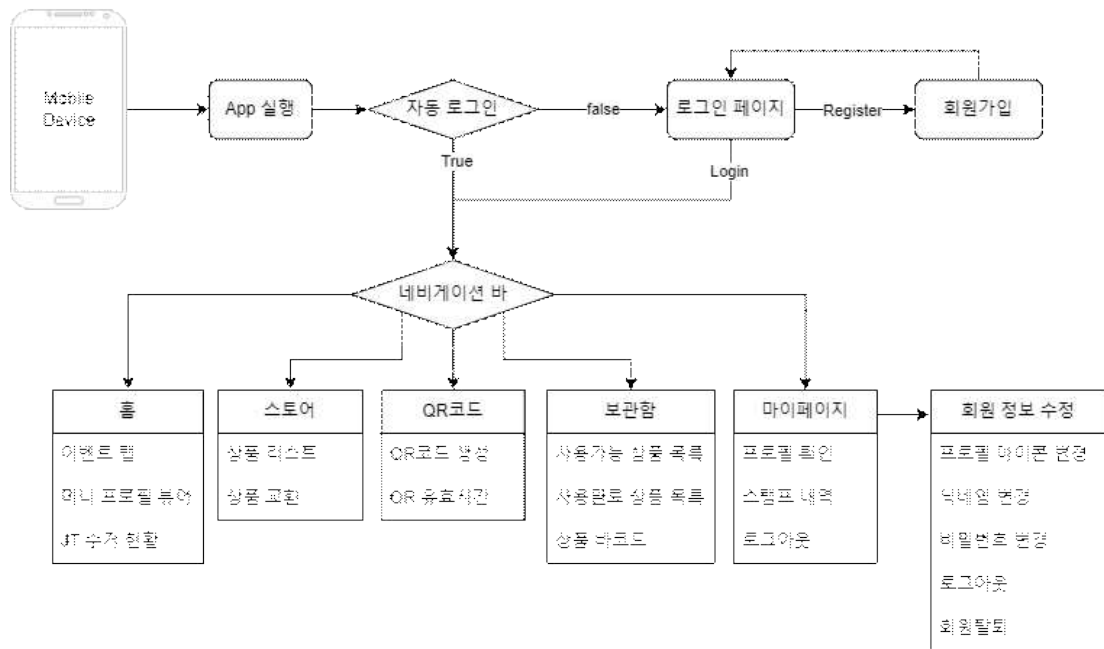


그림 - Mobile App 상세 기능 블록도

Kotlin을 이용하여 Android Mobile에 설치 가능한 App을 제작한다. App 실행 시 로그인 페이지로 이동한다. 로그인 페이지에서는 회원가입, 로그인 기능을 수행해야 한다. 회원가입 요청 시 ID, PW, 닉네임 등을 입력 받아야 한다. 사용자는 해당 ID와 PW를 통해 로그인/로그아웃이 가능해야 한다. 또한 자동로그인 기능 체크 시 다음 로그인 과정에서 이를 통과하여 바로 메인 페이지로 이동해야 한다. 메인 페이지 이동 시 네비게이션 바를 통해 홈, 스토어, QR코드, 보관함, 마이페이지로 이동할 수 있다. 홈 페이지에서는 전체적인 상황을 한눈에 볼 수 있도록 설계해야 하므로 현재 진행 중인 캠페인/이벤트를 확인할 수 있는 탭, 간단하게 프로필을 확인할 수 있는 뷰어, 현재 JT의 수거 현황 등을 체크할 수 있어야 한다. 스토어 페이지에서는 교환 가능한 상품 리스트를 출력하고 유저의 요청에 따라 포인트/스탬프를 사용하여 상품을 교환할 수 있어야 한다. QR코드 페이지에서는 유저의 고유한 ID를 활용해 일정시간동안 유효한 QR코드를 생성하여 JT에서 플라스틱 폐기물을 수거할 시 해당 유저를 인식시키기 위한 목적으로 사용한다. 보관함 페이지에서는 스토어에서 구매했던 상품을 확인할 수 있다. 사용 가능한 상품과 사용 완료한 상품을 볼 수 있고 사용 가능한 상품은 해당 상품의 바코드를 확인할 수 있다. 마이페이지에서는 회원 정보를 확인할 수 있고 사용한 스탬프 내역, 구매한 상품 내역 등을 확인할 수 있다. 또한 마이페이지에서 회원 정보 수정 페이지로 넘어가 프로필 아이콘, 닉네임, 비밀번호 등을 변경할 수 있어야 한다. 사용자가 로그아웃을 했을 경우 세션을 종료하고 쿠키를 삭제해야 한다. 사용자가 회원탈퇴 요청 시 로그아웃 과정과 더불어서 서버에 해당 유저에 대한 정보를 삭제할 것을 요청해야 한다.

### 5.3.3. AP Server

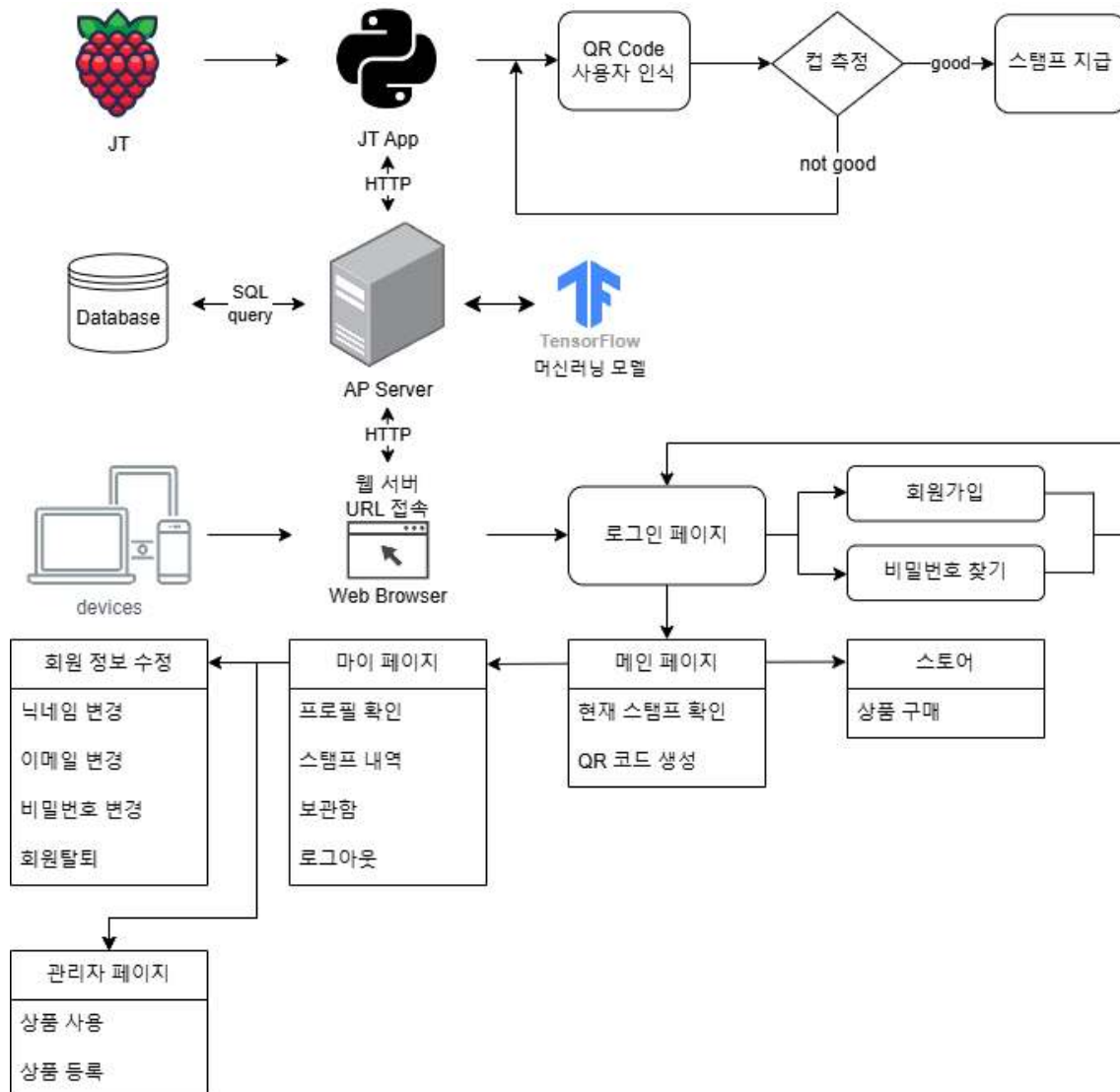
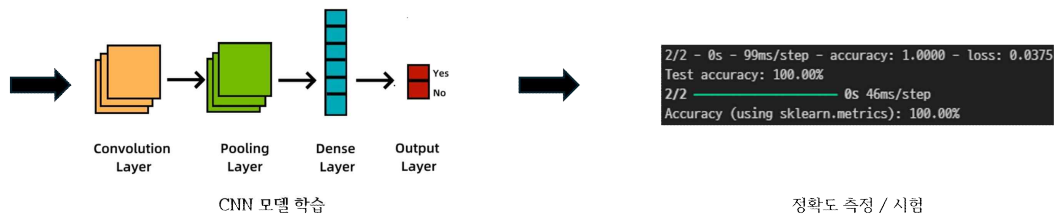
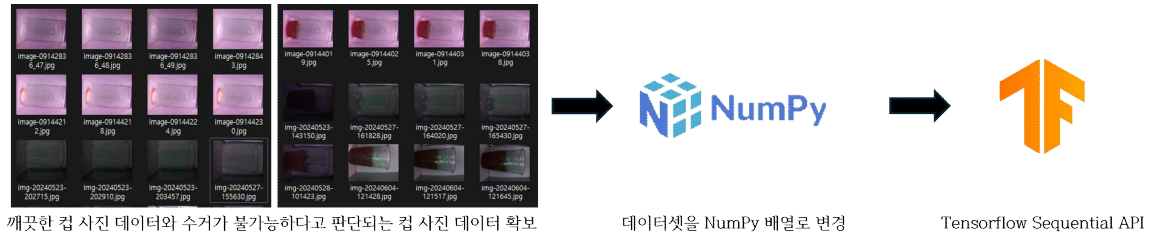


그림 - AP Server 상세 기능 블록도

Node.js 기반의 AP Server를 개발한다. AP 서버는 JT 기기 프로그램이 사용할 API와 사용자가 접속할 웹 서버를 제공해야 한다. JT 기기 프로그램이 사용하는 API는 QR 코드 유저 인증 API와 플라스틱 컵 사진을 요청받아 머신러닝 모델을 사용해 청결도를 측정하여 결과값을 응답하는 API를 필요로 한다. 웹 서버는 웹 브라우저로 접속하여 플랫폼 종류에 제약 없이 App과 동등한 기능을 이용할 수 있다. URL 접속 시 로그인 페이지로 이동한다. 로그인 페이지에서는 회원가입, 비밀번호 찾기, 로그인 기능을 수행해야 한다. 회원가입 요청 시 ID, PW, 닉네임 등을 입력 받아야 한다. 사용자는 해당 ID와 PW를 통해 로그인/로그아웃이 가능해야 한다. 메인 페이지에서는 현재 스탬프 개수를 확인할 수 있고, QR코드를 생성할 수 있다. QR코드는 유저의 고유한 ID를 활용해 일정시간동안 유효한 코드를 생성하여 JT에서 플라스틱 폐기물을 수거할 시 해당 유저를 인식시키기 위한 목적으로 사용한다. 또한 메인 페이지에서는 스토어 페이지와 마이 페이지로 이동할 수 있어야 한다. 스토어 페이지에서는 교환

가능한 상품 리스트를 출력하고 유저의 요청에 따라 포인트/스탬프를 사용하여 상품을 교환할 수 있어야한다. 마이 페이지에서는 회원 정보를 확인할 수 있고 스탬프 내역, 보관함을 확인할 수 있고, 로그아웃 기능과 회원 정보 수정 기능이 있어야 한다. 관리자 계정일 경우 관리자 기능을 사용할 수 있어야 한다.

### 5.3.4. 분류기 상세설계



분류기를 설계하기 위하여 우선 데이터셋을 확보한다. 데이터셋에는 두 가지 종류가 있는데, 우선 깨끗한 컵을 Label 0, 얼음이 담겼거나 음료가 담긴 재활용 수거가 불가능한 컵을 Label 1로 설정한다. 이후, 데이터셋을 NumPy 배열로 변환한 다음 Tensorflow Sequential API를 사용해 모델을 작성한다. train set과 test set으로 나뉜 데이터셋을 가지고 모델을 학습한 다음, 정확도 측정 및 시험을 진행한다. 이를 통해 깨끗한 컵은 측정 결과 0.5 이하, 더러운 컵은 측정 결과 0.5 초과인 모델을 얻을 수 있다.



### 5.3.5. JT 상세설계

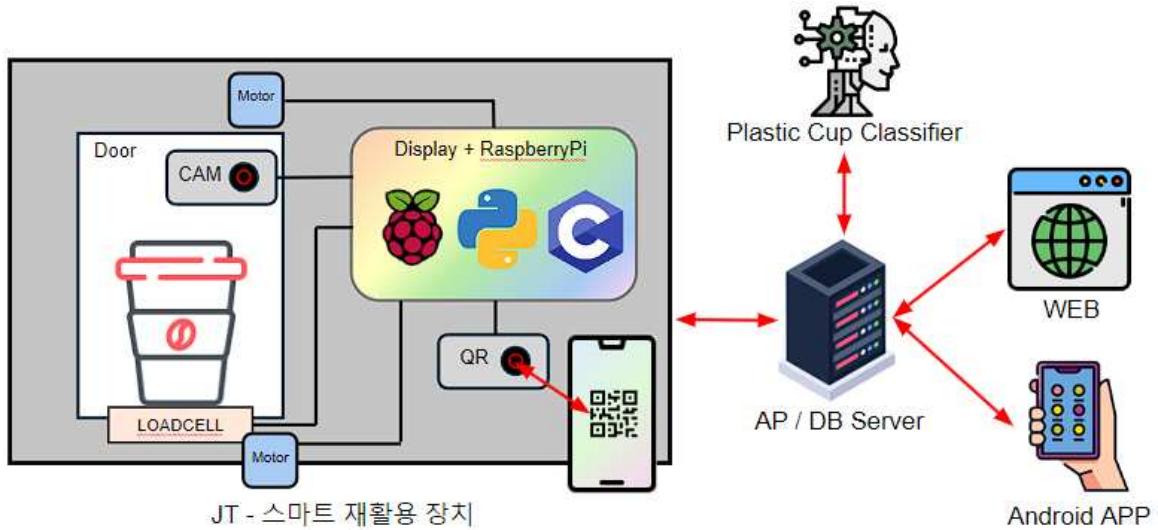


그림 - JT 동작 예상도

#### 5.3.5.1. JT S/W 구현

사용자가 기기를 조작하고, 라즈베리파이에 연결된 센서, 모듈들이 각자의 기능을 수행하도록 제어하는 S/W를 제작한다. 또한, 라즈베리파이는 모바일 APP 및 AP서버와 통신하는 기능을 구현한다. 구현 방법은 Python 라이브러리 중 GUI 인터페이스 구현이 가능한 tkinter를 사용하여 구현한다. 터치스크린으로 제어하는 방식을 감안하여 버튼의 크기는 7인치 터치 디스플레이 사이즈를 고려하여 설계하며, 다양한 사용자가 이용하는 것을 고려하여 장치의 사용이 복잡하게 느껴지지 않도록 디자인한다.

#### 5.3.5.2. QR코드 인식 기능

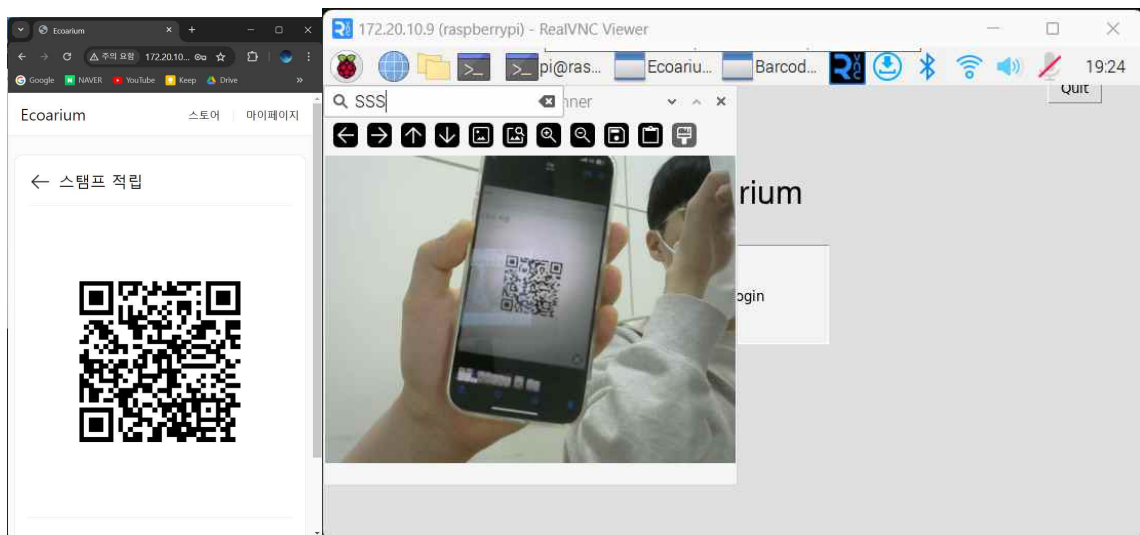


그림 - 로그인 QR

그림 - 실시간 QR코드 인식

JT의 전면부에 장착된 웹캠으로 실시간으로 QR코드를 인식하는 기능을 작성한다. 이를 위해 OpenCV와 imutils를 이용하여 바코드의 프레임을 인식해 실시간 촬영되는



영상에서 QR코드의 위치를 파악하여, 그 값을 읽어오는 기능을 구현한다. 또한, QR코드에서 읽어온 값을 AP서버와 통신하여 로그인 가능한지 아닌지를 판단하는 기능을 구현한다.

#### 5.3.5.3. 투입구 열림, 닫힘 기능

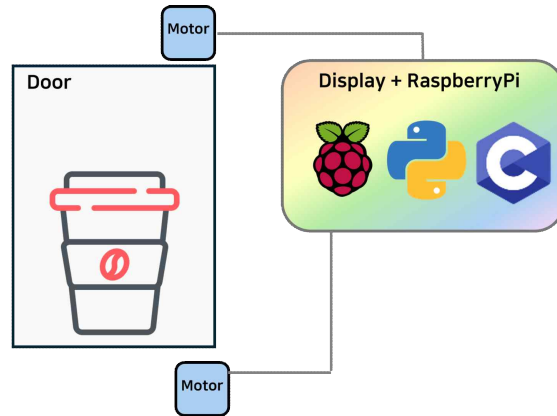


그림 - JT 전면부 문 설계

사용자 로그인이 성공할 경우, 컵의 사진 촬영 및 무게 측정을 위해 기기 내부로 집어넣을 수 있도록 투입구가 열리고 닫히는 기능을 구현한다. 문의 무게를 고려, 기기의 문 우측 위, 아래로 모터를 장착하여 문 제어를 안정적으로 할 수 있도록 한다.

#### 5.3.5.4. 컵 무게 측정 기능

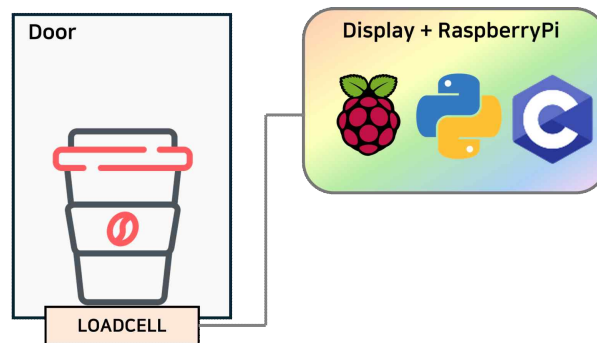


그림 - JT 로드셀 장착 설계

투명한 음료 및 이물질이 담겨있는 컵을 구분하기 위하여 물리적으로 구분이 가능한 로드셀을 장착한다. 투명한 음료 및 이물질은 사진으로 구분하기 어렵기 때문에 이와 같은 방식을 사용한다. 일정 무게 이상 측정되었을 경우, 빈 컵이 아니라고 판단하여 사진 촬영 과정을 거치지 않고 즉시 수거 불가능한 컵으로 판단하여 사용자가 다시 회수하도록 유도한다.

### 5.3.5.5. 촬영 환경 조성, 컵 사진 촬영 기능

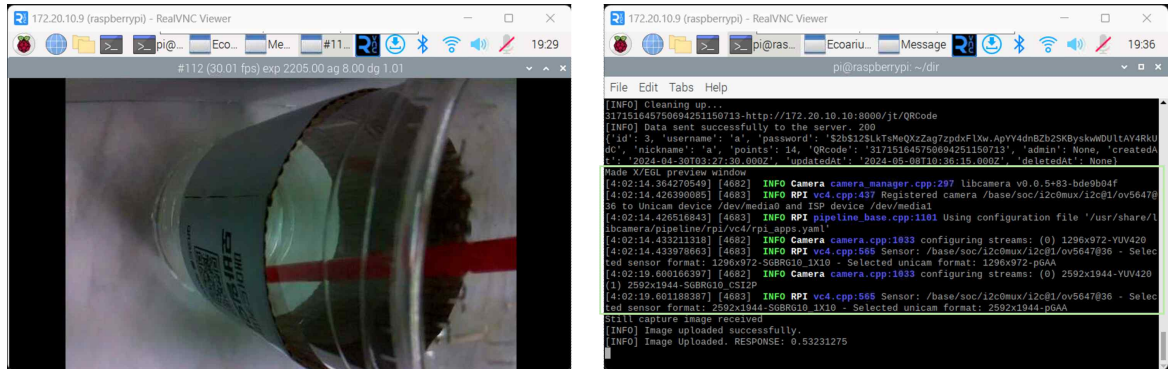


그림 - JT에서 컵의 사진을 촬영

컵 사진이 촬영될 때 항상 같은 환경에서 촬영되도록 환경을 설정하여 모델 판정의 정확도를 올리도록 한다. 또한, 기기 내부로 컵이 들어왔을 경우 카메라 모듈을 통해 사진을 촬영한다. 촬영은 Python 환경에서 라즈베리파이 터미널 명령어를 입력하는 방식으로 사용하며, 현재 라즈베리파이 버전에 맞는 libcamera 모듈을 사용하여 촬영한다.

### 5.3.5.6. 판정 결과에 따른 컵 처리 기능

촬영한 사진을 AP서버로 전송하여, 모델 예측값을 수신하는 기능을 구현한다. 예측값이 0.5 초과일 경우 더러운 컵으로 판단, 사용자가 직접 컵을 회수하도록 한다. 0.5 이하의 경우에는 적절한 컵으로 판단하여 최종적으로 JT에서 컵을 수거하고, 사용자에게 포인트를 지급하도록 한다.

## 6. 제작(Implementation)

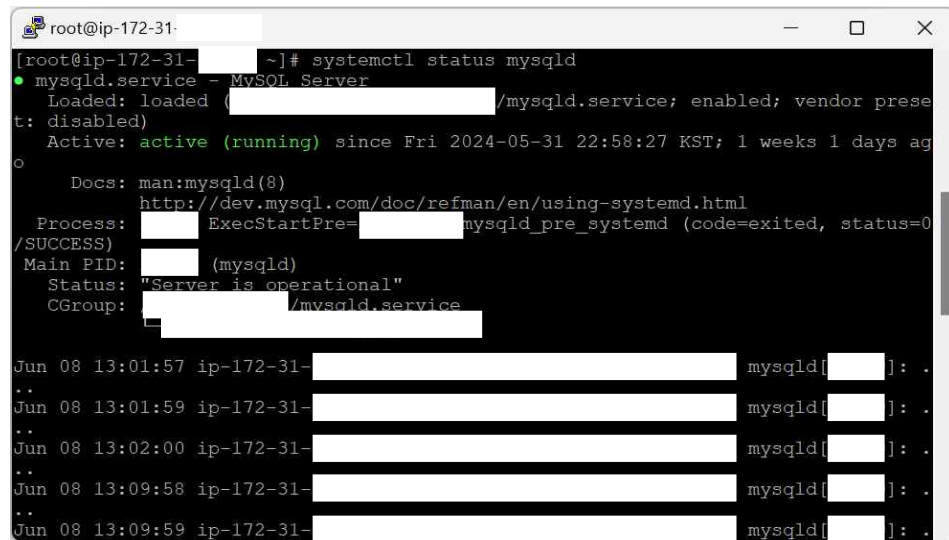
### 6.1 제작과정

#### 6.1.1. DB서버

##### 6.1.1.1 DB서버 구성



[사진1] - AWS EC2 instance 생성 화면



[사진2] - AWS EC2 instance 원격 접속 화면

DB서버 구축을 위해 [사진1]과 같이 Linux CentOS7 기반의 AWS EC2 instance를 생성하고, 해당 인스턴스에 키교환을 통한 원격접속(ssh)하여 root 사용자로 로그인한다. 그 후, 아래 명령어를 통해 mysql을 설치하고, 실행시킨 후 진입한다.

```
yum install -y https://dev.mysql.com/get/mysql80-community-release-el7-3.noarch.rpm
yum install -y mysql-community-server
systemctl start mysqld
systemctl enable mysqld
mysql -u root -p
```

[Linux CentOS7 명령어 1] - mysql 설치와 실행 명령어

### 6.1.1.2 AP(Web)-DB서버 연동

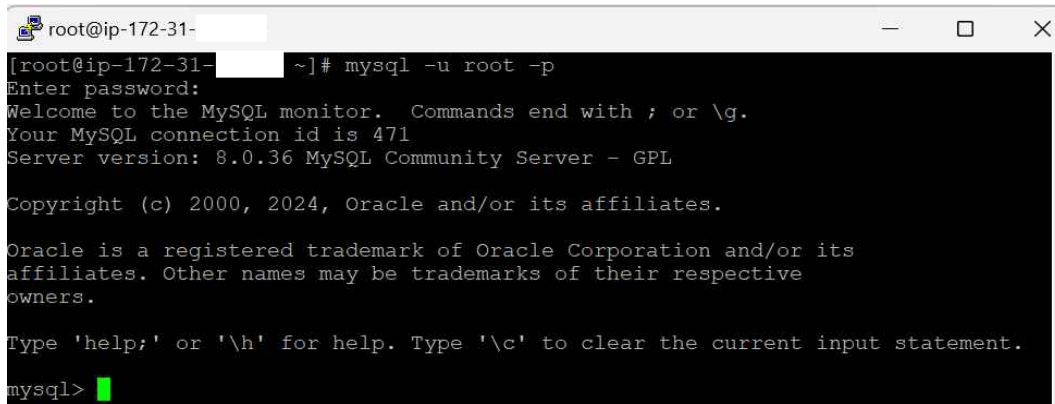


사진 - 명령어를 통한 MySQL 진입 화면

테이블링과 실제 DB서버에 적재되는 데이터 CRUD는 Node.js 웹서버 백엔드에서 Node.js와 함께 사용할 수 있는 ORM(Object-Relational Mapping) 라이브러리인 Sequelize에 의해 실행되므로, 스키마만 다음 쿼리문을 통해 생성하여 접속해준다.

```
CREATE DATABASE IF NOT EXISTS ecoarium;  
USE ecoarium;
```

[Linux CentOS7 명령어 1] - mysql 설치와 실행 명령어

DB서버와 AP(Web)서버의 시퀀라이즈 연동을 위해 node.js Web서버 config.json파일의 적색 부분을 다음과 같이 수정한다. 이때 root 계정으로의 원격접속을 제한하기 위해 보안정책을 설정하고, oracle12 계정을 생성하여 이를 사용하였다. 한편, 주의할 점으로 host의 ip(dns)주소는 DB서버로 통신이 성공된(tcp/3306) 상태에서 MySQL로 접속하므로 인스턴스의 내부ip주소인 public ip(dns)가 되어야 한다.

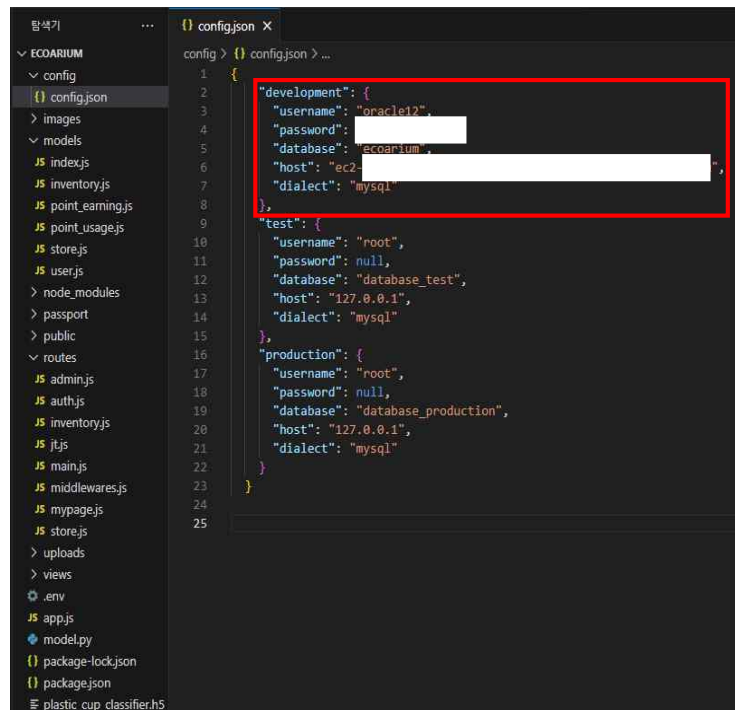


사진 - Node.js 웹서버 내 시퀀라이즈 설정파일(config.json) 수정 화면

#### 6.1.2. Mobile Application

[Ecoarium GitHub](#) KTH Branch 참조

#### 6.1.3. AP Server / WEB Server

[Ecoarium GitHub](#) jisw Branch 참조

#### 6.1.4. JT - 플라스틱 컵 수거 장치

[Ecoarium GitHub](#) kcy Branch 참조

#### 6.1.5. 플라스틱 컵 분류기

[Ecoarium GitHub](#) kcy Branch 참조

## 6.2. 제작시 문제점 및 개선사항

1) Web Server에서 A.I Model을 불러들이는 과정에서 기존에 있던 패키지와 A.I Model이 호환되지 않고 패키지 버전에 대한 종속성 문제 발생

- h5 형식에서 Node.js와 호환되는 json 형식으로 변환하여 해결 시도, 불가
- Python 모델 실행 파일(.exe)의 결과를 불러오는 방식으로 변경해서 해결

2) 조도에 따라서 A.I Model의 정확도가 낮아지는 문제가 발생

- Model 수정을 통해 일정부분 해결 됐지만 얼음이 담긴 컵은 정확히 구분 불가
- 무게, 조도 센서 등의 물리적인 센서를 통해 해결

3) DB서버 MySQL과 웹서버 시퀀라이즈가 통신하는 과정에서 ConnectionError [SequelizeConnectionError]: connect ETIMEDOUT 발생

- Linux Cent OS7 firewall-cmd 명령어를 통해 Mysql 서비스 포트인 3306을 Permanent 옵션을 주어 포트 개방 후 해결

4) Mobile Device에서 App Server에 대해 로그아웃 요청을 보낼 경우, 세션이 제대로 삭제되지 않아서 이후 다시 로그인에 불가능한 오류

- Web을 통해 접근하는 경우에는 브라우저에서 쿠키가 자동으로 생성되어 세션을 관리하지만 Mobile App은 쿠키를 사용하지 않음을 확인, 따라서 Mobile Device와 웹 브라우저와 동일하게 동작하도록 CookieJar 라이브러리를 활용하여 Mobile에서 보내는 요청에 쿠키를 포함시켜서 서버에서 세션을 관리하는 방식을 통일시켜 해결

5) 기존에 사용하던 라즈베리파이4B의 OS버전이 bookworm이었는데, 이는 터치스크린의 터치 기능을 지원하지 않음

- bullseye로 다운그레이드하여 새로 라즈베리파이 환경설정 완료

6) 현재 라즈베리파이에 연결된 카메라 모듈은 총 2개로, 플라스틱 컵 촬영용 카메라는 FFC케이블을 통한 연결이고, 다른 하나는 QR코드 실시간 인식을 위해 USB케이블을 통하여 웹캠(Logitech C270)이다. 기존에 작성한 카메라 센서를 사용하는 코드에서 자동으로 카메라 센서를 인식한 후, 활성화하여 사진을 찍도록 코드를 작성했지만 새로 웹캠을 장착한 후 자동으로 카메라 센서를 인식하지 못하는 현상을 확인

- 라즈베리파이의 .config 파일을 수정하여 직접 카메라 센서를 번호로 인식하도록 수정했고, 코드 또한 lsusb를 통해 카메라의 장치번호를 확인하여 두 카메라가 혼동되지 않도록 프로그램을 작성.

7) App의 Main Activity에서 뒤로가기 버튼 클릭시 isLoggedIn 변수를 사용하는 로그인 유지방식이 동작하지 않는 문제점이 발생, 세션이 제대로 종료되지 않음

- MainActivity에 onBackPressed함수를 override해서 뒤로가기 버튼을 클릭 시 앱이

꺼지지 않고 그대로 유지되도록 변경, 중첩되는 fragment 실행 시 (Ex)회원정보-회원 관리) 뒤로가기 버튼을 눌렀을 때 이전 fragment로 이동하도록 변경완료

8) DB서버(Linux CentOS7) 보안점검 중 사용자 암호 만료 기간 정책 설정 이후 ssh 원격접속이 차단됨

- 정책이 설정되자마자 계정 비밀번호가 만료되어 ssh 원격접속이 차단되었던 것으로 확인하여 비밀번호 변경 후 정상화

9) DBMS(Linux CentOS7 내 MySQL) 보안점검 중 사용자 암호 만료 기간 정책 설정 이후 MySQL로의 원격 접속(웹서버에서의 접근 포함) 차단

- 정책이 설정되자마자 계정 비밀번호가 만료되어 MySQL로의 원격 접속(웹서버에서의 접근 포함)이 차단되었던 것으로 확인하여 비밀번호 변경 후 정상화

10) DBMS(Linux CentOS7 내 MySQL) 보안점검 중 DBMS 계정에 대한 단순 패스워드 사용 금지 정책 설정 과정에서 SET GLOBAL validate\_password.policy = 'STRONG'; 설정 이후 MySQL로의 원격 접속(웹서버에서의 접근 포함) 차단

- SET GLOBAL validate\_password.policy = 'STRONG';으로 정책 설정 시 원격접속이 차단되어 SET GLOBAL validate\_password.policy = 'MEDIUM';으로 정책 수정 이후 /etc/my.cnf 파일 또한 수정하여 정책 설정

11) DBMS(Linux CentOS7 내 MySQL) 보안점검 중 DB 내 데이터 갱신 사항 로깅 정책 설정 과정에서 systemctl status mysqld 상태가 계속 OFF상태

- 일반 쿼리 로깅 활성화를 위한 general\_log = 1가 /etc/my.cnf 내에 두 번 정의되어 오류 발생되었던 것으로 확인하여 삭제 후 정상화

12) DB서버(Linux CentOS7) 취약점 진단 스크립트 실행 이후 ssh 원격접속 차단됨

- 스크립트 실행 과정에서 자동적으로 /home, /home/centos, /home/centos/.ssh, /home/centos/.ssh/authorized\_keys에 대한 접근 권한이 변경되어 ssh 원격접속 차단된 것으로 확인, aws 홈페이지 <https://repost.aws/knowledge-center/ec2-server-refused-our-key> 참조하여 원래대로 각 디렉토리 접근 권한 설정 후 정상화

13) 기존에 사용하던 라즈베리파이4B의 OS버전이 bookworm이었는데, 이는 터치스크린의 터치 기능을 지원하지 않음

- bullseye로 다운그레이드하여 새로 라즈베리파이 환경설정을 해 주었다.

14) 현지 라즈베리파이에 연결된 카메라 모듈은 총 2개로, 플라스틱 컵 촬영용 카메라라는 FFC케이블을 통한 연결이고, 다른 하나는 QR코드 실시간 인식을 위해 USB케이



블을 통하여 웹캠(Logitech C270)이다. 기존에 작성한 카메라 센서를 사용하는 코드에서 자동으로 카메라 센서를 인식한 후, 활성화하여 사진을 찍도록 코드를 작성했지만 새로 웹캠을 장착한 후 자동으로 카메라 센서를 인식하지 못하는 현상을 확인

- 라즈베리파이의 .config 파일을 수정하여 직접 카메라 센서를 번호로 인식하도록 수정했고, 코드 또한 lsusb를 통해 카메라의 장치번호를 확인하여 두 카메라가 혼동되지 않도록 프로그램을 작성하였다.

### 6.3. 지도교수 지도내용 및 지적사항 조치결과

일정	지도내용 및 지적사항	조치결과
2024/05/14	플라스틱 컵 분류기에서 0.5를 기준으로 분류를 하는데, 0.5라는 수치는 어떤 근거로 정해진 수치인가?	많은 이진 분류 모델 (예: 로지스틱 회귀, 신경망 등)은 예측 결과를 0과 1 사이의 확률 값으로 제공한다. 이 값은 특정 클래스 (보통 양성 클래스)에 속할 확률을 나타낸다. 예측 값이 0.5 이상인 경우 해당 클래스에 속할 확률이 50% 이상이므로 해당 클래스로 분류하고, 0.5 미만인 경우는 그 반대로 분류한다. 기본적으로 모델링에서는 잘못된 양성 (False Positive)과 잘못된 음성 (False Negative)의 비용이 대칭적이라고 가정한다. 즉, 두 가지 오류가 동일하게 중요하다고 가정할 때 0.5를 기준으로 삼는 것이 합리적이다. 마지막으로, 학습 과정에서 모델은 주로 로그 손실 함수 (log loss)를 최소화한다. 이 과정에서 0.5를 기준으로 하는 것이 수학적으로 자연스럽게 나타나며, 확률 예측을 하는 모델에서 최적화를 단순화한다.



## 7. 시험

### 7.1. 시험(측정) 환경

#### 7.1.1. DB Server

Cloud Service Platform : AWS EC2 instance

System information

OS : CentOS Linux release 7.9.2009 (Core)

CPU : 1 core

Kernel : 3.10.0-1160.114.2.el7.x86\_64

Memory : 537 MB / 989 MB (54.30%)

Disk : 2.8G / 16G (18%%)

DBMS : mysql Ver 8.0.36 for Linux on x86\_64 (MySQL Community Server - GPL)

#### 7.1.2. Mobile Application

IDE : IntelliJ IDEA 2023.2.2 (Community Edition)

Language : Kotlin 1.9.0

Android Virtual Device

Model : Pixel 4 XL (6.3inch flexible OLED, 19:9, 3040x1440, 537ppi)

OS : Android 11.0

Android Tablet

Model : Galaxy Tab S8+ (12.4inch Super AMOLED, 16:10, 2800x1752, 266ppi)

OS : Android 13.0

#### 7.1.3. AP Server

IDE : Visual Studio Code 1.90

Node.js : 18.16.0

Python : 3.12

#### 7.1.4. 플라스틱 컵 분류기

IDE : Visual Studio Code 1.90

Python: 3.11.9

Tensorflow: 2.16.1

#### 7.1.5. JT(플라스틱 컵 수거 장치) H/W, S/W

IDE : Visual Studio Code 1.90

Python: 3.11.9

gcc: 6.3.0

Raspberry Pi 4B Bullseye

## 7.2. 시험(측정) 결과

### 7.2.1. Mobile Application

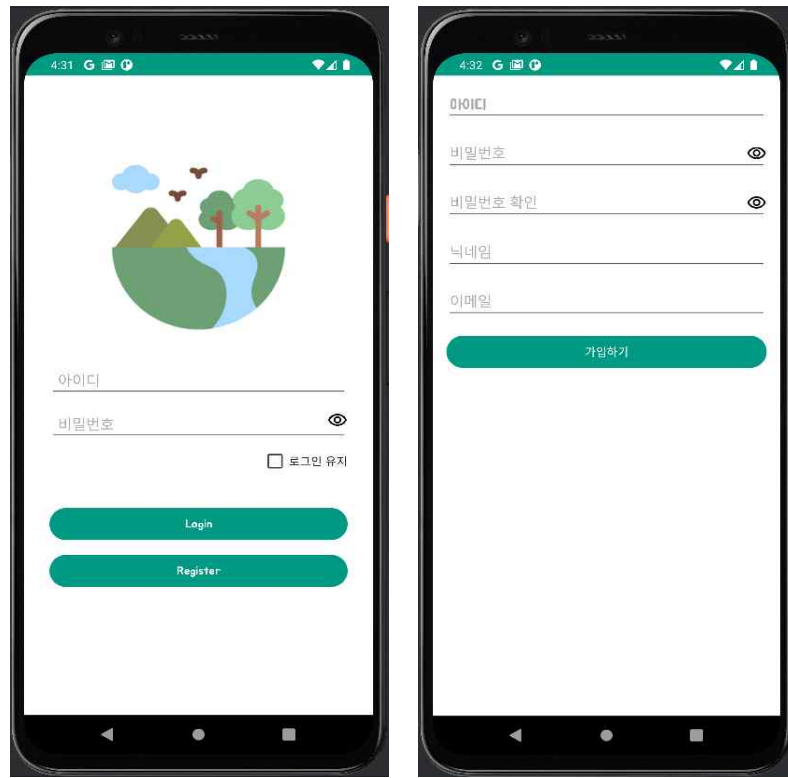


그림 - 로그인 페이지

그림 - 회원가입 페이지

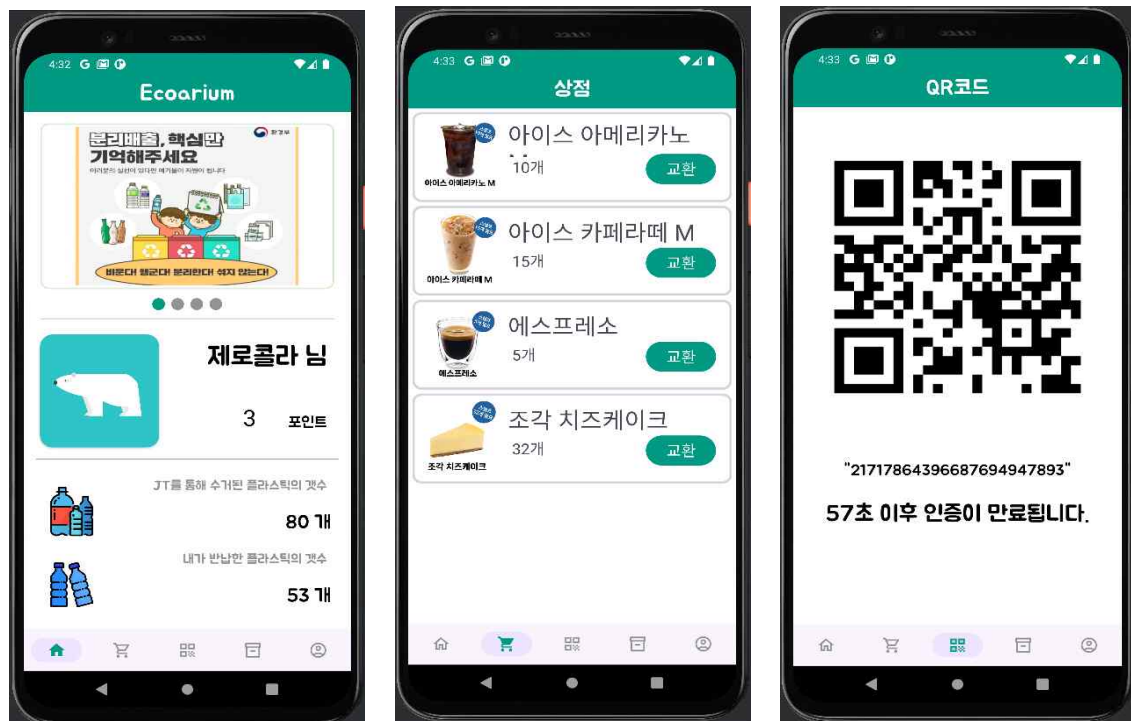


그림 - 홈 페이지

그림 - 스토어 페이지

그림 - QR코드 페이지

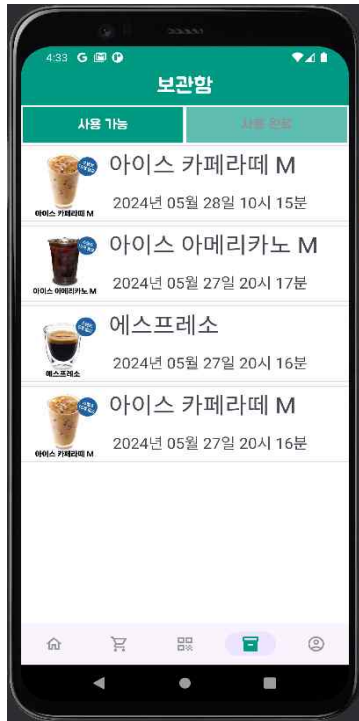


그림 - 보관함 페이지



그림 - 마이페이지

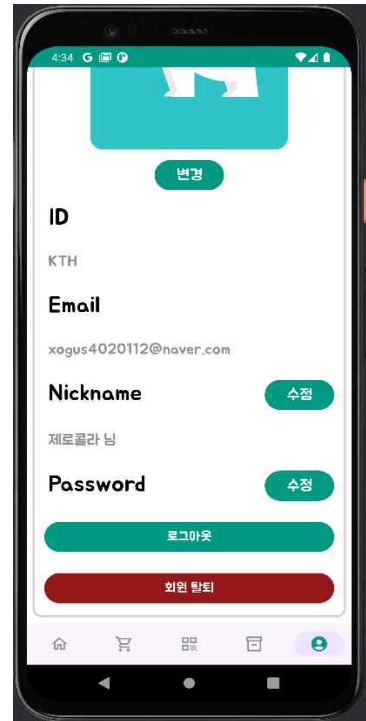


그림 - 회원 정보 수정 페이지

## 7.2.2. AP Server / Web Client

**Ecoarium**

아이디

---

비밀번호

---

로그인

비밀번호 찾기    회원가입

그림 - 로그인 페이지

**Ecoarium**

**회원가입**

아이디

---

비밀번호

---

비밀번호 확인

---

이메일

---

닉네임

---

가입하기

로그인

그림 - 회원가입 페이지

**Ecoarium**      스토어    마이페이지

내 스탬프    적립 QR      10개

스탬프  
전남

스탬프  
전남

스탬프  
전남

스탬프  
전남

스탬프  
전남

그림 - 메인 페이지

**Ecoarium**      스토어    마이페이지

← 스탬프 적립

유효시간: 58초

그림 - 적립 QR코드 생성

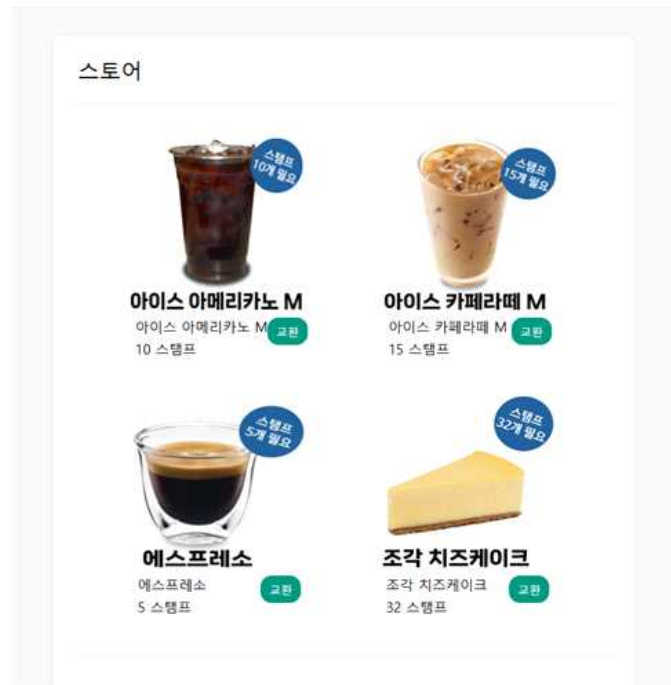


그림 - 스토어 페이지

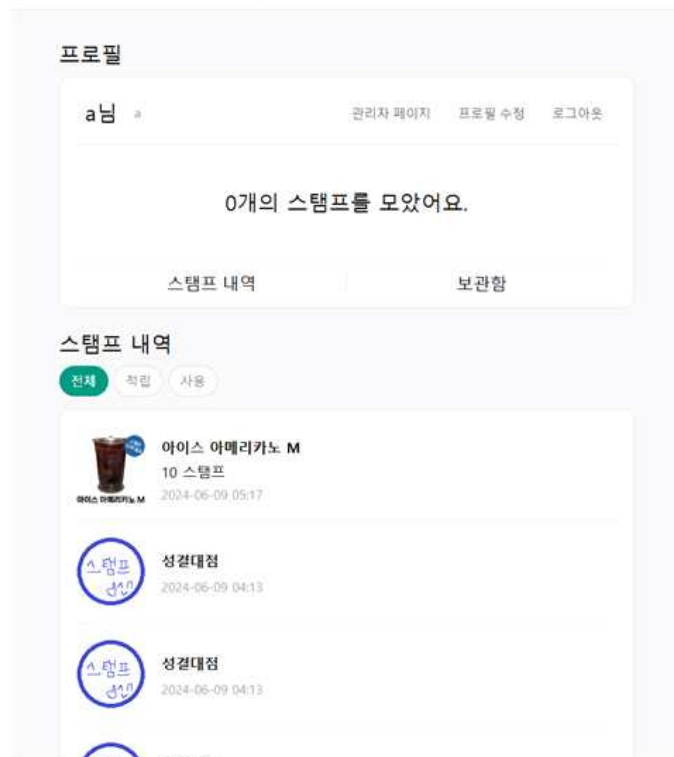


그림 - 마이페이지 - 스탬프 내역



그림 - 마이페이지 - 보관함

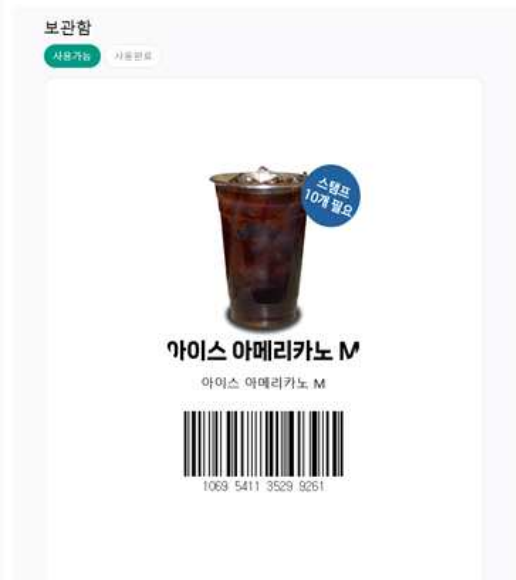


그림 - 상품 상세 정보



그림 - 프로필 수정 페이지



그림 - 비밀번호 변경 페이지



그림 - 회원탈퇴 페이지

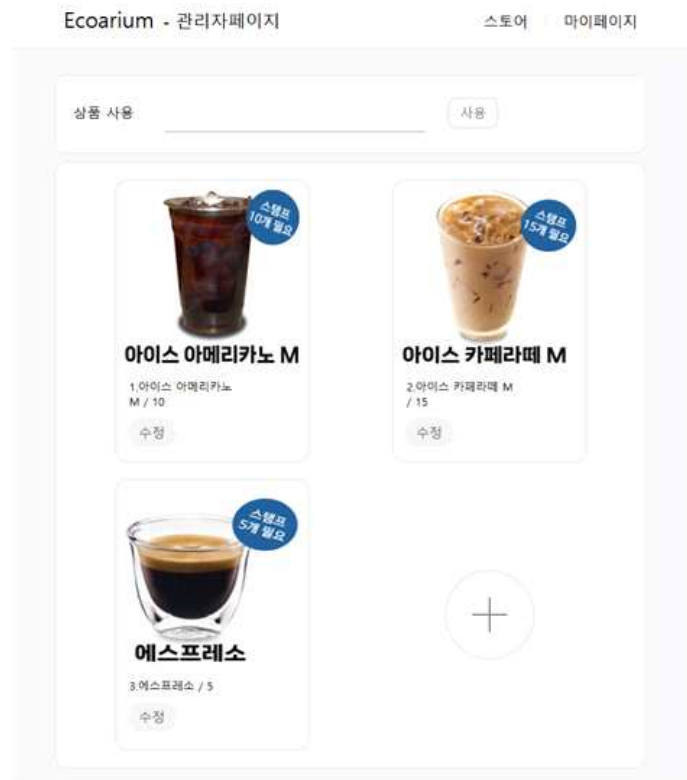


그림 - 관리자 페이지



그림 - 비밀번호 찾기

그림 - 비밀번호 재설정

그림 - 이메일 수신

### 7.2.3. JT - 플라스틱 컵 수거 장치

[Ecoarium GitHub](#) kcy README.md 참조

### 7.2.4. 플라스틱 컵 분류기

[Ecoarium GitHub](#) kcy README.md 참조



그림 - 플라스틱 컵 분류기 test용 프로그램 실행 결과



## 8. 평가

### 8.1. 정량적/정성적 목표달성도 평가

항목	달성률
JT(Linux Raspbian), AP(Windows)서버, DB서버(Linux Cent OS7)로 시스템 구성 - JT ↔ AP서버 간 WLAN 통신 (방화벽 특정 포트 개방) - AP ↔ DB 서버 간 NAT 통신 (Host OS(AP;Windows) ↔ Guest OS(DB;Cent OS7))	100%
AP서버에서 Node.js 기반 Web서버 동작, DB서버에서 사용자 정보 MySQL 테이블링하여 저장	100%
AP서버 접속을 위한 사용자 안드로이드 어플리케이션 개발	100%
Raspberry Pi 기반의 JT로 센서 및 장치 제어하여 일회용 플라스틱 컵 수거 - Raspberry Pi GPIO Port 사용하여 센서 및 장치 제어 - 카메라 모듈 제어하여 투입된 물체 사진 캡처 후 AP 서버로 전송	100%
수거된 일회용 플라스틱 컵 청결도 검사를 위한 인공신경망 모델 구현 - AP서버에서 인공신경망 모델을 이용하여 수신받은 물체 사진 식별 후 JT로 식별값 리턴(송신)	100%
수거 완료 시 해당 사용자에게 포인트 적립 (DB 상 저장됨)	100%

### 8.2 현실적 제한요소 달성도 평가

현실적 제한 요소	목표	달성결과
경제	- 분리수거 비용과 사용자 보상금 책정 간 밸런스 유지 - 하드웨어 리소스 최대한 적게 사용	- 사용자를 고려한 보증금 책정, 스탬프 방식 - 소프트웨어에서 센서/장치 데이터 처리하여 하드웨어 비용 절감
편리	- 기기 주변 환경/접근성이 사용자가 편리하여야 함. - 분리수거함의 디자인(색과 소재, 모양)이 사용자의 이용률을 결정할 수 있음.	- 교내 / 주변 카페와 연계하는 방안 추진 - 분리수거 함 외관 하드웨어 디자인 고려, 작은 사이즈
윤리	- 적립 방식을 이용한 악용(이미 분리수거 되어진 컵 계속하여 투입)	- 플라스틱 폐기물 회수를 통해 이를 방지 및 규격화된 컵만을 수거
사회	- 플라스틱 폐기물 발생에 따른 재활용률 상승의 필요성	- 분리수거 보상 제도를 통한 재활용률 향상

### 8.3 기능적 요구사항 달성도 평가

#### 8.3.1 전체 시스템 인프라 구성

ID	요구사항	내용	우선순위	달성률 (%)	비고
Sys01	AWS EC2 인스턴스 내 DBMS 세팅	DB서버를 구현하기 위해 AWS EC2 인스턴스 내 환경 구축	1	100%	.
Sys02	AP-DB서버 이원화	AP 서버와 DB 서버를 분리하여 운영할 수 있도록 구축	2	100%	.
Sys03	정보보안 시스템 구축	시스템 보안 및 네트워크 보안 시스템 구축	3	100%	.

#### 8.3.2 Mobile Application

ID	요구사항	내용	우선순위	달성률 (%)	비고
App01	회원 가입 및 로그인/로그아웃	사용자가 앱을 통해 회원 가입을 하고 로그인 및 로그아웃할 수 있어야 한다.	2	100%	.
App02	세션관리	로그인 상태 유지를 위해 세션을 관리해야 한다.	3	100%	.
App03	QR 코드 생성 및 인식 기능	사용자의 정보에 따라 QR 코드를 생성하고, 이를 인식할 수 있어야 한다.	4	100%	.
App04	스탬프/포인트 관리 기능	사용자가 스탬프나 포인트를 확인하고 사용할 수 있어야 한다.	6	90%	무한 스크롤 미구현
App05	회원 정보 확인 및 수정 기능	사용자가 자신의 회원 정보를 확인하고 수정할 수 있어야 한다.	5	100%	.
App06	서버와의 데이터 통신	앱과 서버 간의 데이터를 주고받을 수 있어야 한다.	1	100%	.
App07	App 사용 편의성	사용자가 앱을 사용할 때 불편함을 느끼지 않아야 한다.	7	95%	Display 크기에 따라 달라지는 UI 형식

### 8.3.3. AP Server / WEB Client

ID	요구사항	내용	우선 순위	달성률 (%)	비고
AP01	QR 코드 유저 인증 API	JT 기기에서 사용자 QR 코드를 요청받아 유저 정보를 응답해주는 API	1	100%	
AP02	플라스틱 컵 청결도 검사 API	JT 기기에서 촬영한 컵 사진을 요청받아 머신러닝 모델 판단 결과를 응답해 주는 API	2	100%	
AP03	웹 서버 구축	사용자가 접속 가능한 웹	3	100%	
AP04	회원가입 및 로그인/로그아웃	사용자가 웹을 통해 회원가입을 하고 로그인 및 로그아웃할 수 있어야 한다.	4	100%	
AP05	세션관리	로그인 상태 유지를 위해 세션을 관리해야 한다.	5	100%	
AP06	QR 코드 생성 및 인식 기능	사용자의 정보에 따라 QR 코드를 생성하고, 이를 인식할 수 있어야 한다.	6	100%	
AP07	스탬프/포인트 관리 기능	사용자가 스탬프나 포인트를 확인하고 사용할 수 있어야 한다.	7	100%	
AP08	상품 등록 기능	관리자가 스토어에 상품을 등록하는 기능	8	90%	상품 수정 기능 미구현
AP09	회원 정보 확인 및 수정 기능	사용자가 자신의 회원 정보를 확인하고 수정할 수 있어야 한다.	9	100%	
AP10	비밀번호 찾기 기능	사용자가 인증코드를 통해 비밀번호를 재설정하는 기능	10	100%	

#### 8.3.4. 플라스틱 컵 분류기

ID	요구사항	내용	우선 순위	달성률 (%)	비고
ML01	촬영 환경 기준 설정	항상 일관된 조건에서 사진이 촬영될 수 있도록 한다.	1	100%	.
ML02	적절한 데이터셋 확보	분류 목적을 달성할 수 있도록 적절한 데이터셋을 확보한다.	2	100%	.
ML03	투명한 물질 분류	육안으로도 구별하기 힘든 물질들에 대해서 NG로 분류할 수 있도록 한다.	3	90%	저조도 환경에서 투명 물질 분류 정확도가 완벽하지는 않았음

#### 8.3.5. JT(플라스틱 컵 수거 장치) H/W, S/W

ID	요구사항	내용	우선 순위	달성률 (%)	비고
JT01	JT S/W 구현	상황에 맞는 기능을 수행하도록 라즈베리파이를 제어, Android APP, AP서버와 통신 인터페이스 구축	1	100%	.
JT02	QR코드 인식 기능	QR코드를 인식하면 올바른 QR인지 판독, 로그인을 수행한다.	4	100%	.
JT03	투입구 열림, 닫힘 기능	로그인 성공 시 문이 열리며 사용자로부터 컵을 전달받는다.	6	100%	.
JT04	컵 무게 측정 기능	컵의 무게를 측정하여 비어있는 컵인지 물리적으로 판별함.	7	100%	.
JT05	촬영 환경 조성	LED를 틀어서 촬영 환경을 밝게 한다.	2	100%	.
JT06	컵 사진 촬영 기능	라즈베리파이의 카메라 모듈을 활용하여 컵 사진을 촬영한다.	3	100%	.
JT07	판정 결과에 따른 컵 처리 기능	서버로부터 받은 판정 결과에 따라 컵을 처리한다.	5	100%	.

#### 8.4. 문제점 및 해결방안

1) Application에서 Display의 크기, 비율에 따라 그림이 너무 작게 보이거나 글씨가 한 줄씩 밀리는 등의 UI 상의 의도치 않은 오류가 발생

Text View, Image View 등의 크기를 상위 layout과 종속적인 관계로 변경함을 통해 일정 크기, 길이가 넘어가는 이미지, 텍스트 등은 비율을 유지한 채로 크기, 길이를 조정하는 방안으로 해결 가능

2) 스템프 내역, 상품 사용 기록 등이 점차 쌓여서 Mobile App의 Recycler View에 불필요한 데이터들이 많이 들어가 메모리를 낭비하는 상황이 발생

서버의 Paging 방식과 데이터를 일정량씩 끊어서 요청하는 Recycler View의 무한 스크롤 동작을 같이 구현하여 불필요한 메모리 낭비를 줄이는 방안으로 해결 가능

### 9. 추진체계

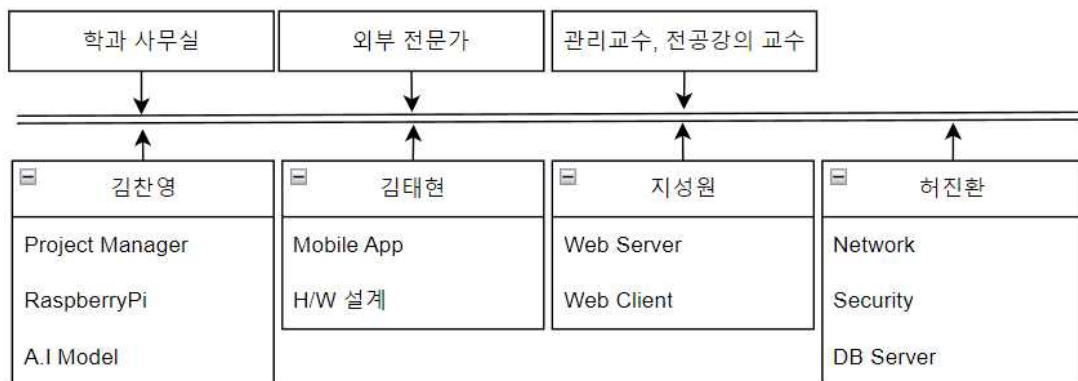


그림 - 본 프로젝트 추진을 위한 추진 체계

본 과제의 추진을 위한 팀은 4인으로 구성되며 다음과 같은 역할을 담당한다. 이론 연구 단계에서부터 과제 종료시점까지 협력과 보완을 통해 팀의 목표를 달성할 수 있도록 한다. 팀장은 전체 프로젝트 진행을 관리하고 목표를 설정하며, 팀원 간의 협업을 조율한다. 프로젝트의 주요 결정 사항을 총괄하며, 일정 관리와 프로젝트의 전반적인 성과를 책임진다. 프로젝트 초기 단계에서 관련 이론과 기술을 연구하고 학습한다. 최신 트렌드와 기술을 파악하여 프로젝트에 적용할 수 있는 방법을 연구한다. H/W 설계 과정에서 변수가 발생할 수 있으니 S/W 설계를 최우선 과제로 삼고 A.I Model 제작을 위한 데이터셋 준비, 서버와 클라이언트 프로토 타입 제작 후 미흡한 점을 단계적으로 고쳐나간다. 설계 과정 중 모듈화 프로그래밍에 대한 점을 중요하게 생각하고 정기적인 회의를 통해 각자의 진행 상황 및 문제점에 대해 활발히 토의한다. 또한 Github Repository를 생성하고 각자의 브랜치를 만들어서 개발 프로젝트에서 효율적으로 협업한다.

- 외부 자문 및 지원

관리교수, 전공강의 교수 정복래 : 프로젝트 진행을 위한 관리, 프로젝트 제작 과정에 대한 피드백

외부 전문가 김호진 : JT 구성을 위한 철제 스테인리스 프레임 지원

학과 사무실 : LCD Display, Sensor, LED, 저항 등 지원

- Project Member

조장 김찬영(20190895) : Python, 머신러닝, 수거 장치 SW 제작, 프로젝트 총괄

팀원 김태현(20190898) : Kotlin Android APP 제작, 수거 장치 HW 제작

팀원 지성원(20190948) : Node.js AP Server 제작, HTML/CSS 클라이언트 제작

팀원 허진환(20190954) : AWS MySQL DB Server 구축, 네트워크/보안 담당

## 10. 프로젝트 추진일정

수행 내용		일정 (2주 단위)							
		1,2	3,4	5,6	7,8	9,10	11,12	13,14	15,16
목표와 기준 설정	- 설계목표 설정								
	- 시장조사, 문헌조사								
	- 소프트웨어 요구사항 파악								
합성	- 기능 블록 구성								
	- 기능별 구현방법 결정								
	- 적용할 이론 및 기술								
	- UI 설계								
분석	- 세부 기능 블록도								
	- 목표달성가능성 확인								
	- 순서도 작성								
제작	- H/W 제작								
	- S/W 프로그래밍								
시험/ 평가	- 시험 및 검증								
	- Trouble shooting								
	- 재설계								
결과	- 결과보고서 작성								

## 11. 결론

### 11.1. 김찬영

처음 이 프로젝트를 구상했던 2023년 2학기에는, 계획대로만 진행된다면 충분히 모든 목표를 이룰 수 있을 것이라고 생각했었다. 하지만, 2024년 1학기에 접어들고 실제로 프로젝트를 구현하려고 하니 구상 및 기획 단계에서는 생각하지 못했던 많은 어려움들에 마주했다. 가능할 것이라고 생각했던 부분들은 현실의 벽에 막혀 불가능한 경우가 많았고, 이를 극복하기 위해 다른 방법들을 찾는 과정에서 굉장히 애를 먹었다. 또한, 우리 프로젝트의 경우에는 목표로 한 프로젝트 규모 자체가 굉장히 거대했기 때문에, 철저한 분업과 효율적인 협업이 아니었으면 이렇게 프로젝트를 완성할 수 없었다고 생각한다. 그래도 처음 기획 단계에서 최대한 효율적으로 프로젝트를 진행하고자 각자 파트를 나누었기에 다행이라고 생각한다. 팀원 모두가 본인의 파트에 불만 없이 성심성의껏 공부하고 연구하였고, 팀원 모두가 많은 발전을 이뤄낼 수 있었던 시간이었다고 생각한다.

이 프로젝트에서 나의 파트는 플라스틱 컵 분류기 제작과 JT 장치 제작이다. 모델 데이터셋 수집과 장치 S/W 제작은 집에서 할 수 있는 것이 아닌, 설계실에서 실제로 있는 장치로 프로젝트를 진행해야 했기 때문에, 수업이 없는 날에도 계속해서 설계실로 가서 프로젝트를 했었다. 이 과정에서 정말 포기하고 싶은 생각이 많이 들었지만, 목표로 한 기능들이 하나 둘씩 구현되는 모습을 볼 때마다 힘을 낼 수 있었다. BACK-END 부분은 익숙하기도 하고 기능 동작에만 집중해도 되기 때문에 수월하게 진행할 수 있었지만, FRONT-END는 실제로 사용자가 사용한다는 가정하에 제작해야 했기 때문에 여러 시행착오를 겪었다. 프로그램과 디스플레이의 해상도가 맞지 않는 부분, 라즈베리파이의 성능에 맞춰서 프로그램을 제작하는 부분, 센서와 모듈을 제어하고 서버와 통신하는 과정에서 많은 시간을 들였다. 그렇게 완성한 프로젝트이기 때문에, 완성하고 나니 뿌듯한 마음이 들었고 IoT 개발에 대해서 많은 실력 향상을 할 수 있었다. 마지막으로, GitHub를 사용하는 능력 또한 향상되었으며, 팀워크에 대한 부분에서도 많은 것을 배울 수 있었다.

### 11.2. 김태현

Ecoarium App 제작에 있어서 가장 큰 목표는 사용자에게 편리한 앱을 제공하는 것이었다는 점에서 대부분의 목표는 달성했다. 첫 번째 목표로는 사용자가 앱을 통해 회원 가입을 하고 로그인 및 로그아웃을 할 수 있는 기능을 100% 달성했다. 두 번째 세션 관리에 대한 점은 쿠키를 사용하여 서버와의 세션을 유지, 관리 하는 기능을 제작하여 달성했다. QR코드 생성 및 인식기능은 사용자 고유의 ID를 활용해 일정시간 동안 유효한 QR코드를 JT에 인식시키는 기능을 100% 달성했다. 스탬프/포인트 관리 기능은 JT를 통해 플라스틱 폐기물을 반납할 시 유저에게 스탬프를 적립하고 이를 스토어를 통해 사용하는 방식으로 구현했다. 회원 정보 확인 및 수정 기능은 사용자가 자신의 닉네임, 스탬프 등을 확인할 수 있고 필요에 따라 수정, 탈퇴 등의 기능을 제작함으로 달성했다. 서버와의 데이터 통신은 HTTP 메서드를 활용해 JSON객체에 정

보를 담아서 보내는 방식으로 로컬, 인터넷 등의 서버에서도 통신이 가능함을 확인했다. App 사용 편의성에 대한 점은 정렬 버튼, 자동 로그인 등의 기능들을 추가함으로써 사용자가 앱을 사용할 때 불편함을 느끼지 않도록 하는 목표를 대부분 달성할 수 있었다. 하지만 무한 스크롤을 통한 메모리 최적화에 어려움을 겪었고 Display 크기에 따라 달라지는 의도치 않은 UI 형식 등의 오류가 함께 있었다. 이번 설계 과정을 통해 배운 것은 기능 개발의 우선순위를 설정하는 일과 OCP 원칙에 따른 개발이었다. 여러 기능을 개발하면서 무엇을 먼저 개발해야 할지, 추후에 어떤 기능이 추가될 예정이니 코드의 format을 구성하는 과정이 얼마나 중요한지 알게되었다. 이를 통해 프로젝트 관리 능력을 향상시킬 수 있었다. 다음으로는 Front-Ent에 대한 부분이었다. 사용자가 앱을 사용할 때 겪을 불편함과 에러처리에 대한 부분을 최소화 하기 위해 백그라운드 로그인 유지 , 쿠키 등을 사용해서 개선방안을 찾을 수 있었다. 또한 Kotlin 언어에 대한 문제 해결 능력을 많이 기를 수 있었다. 본 프로젝트를 진행하기 전에 Kotlin 언어에 대한 지식이 전무해 간단한 강의를 들었지만 개발 과정에서 발생하는 문제들을 쉽게 해결할 수는 없었다. 이를 해결하기 위해 다양한 접근법을 시도해봤고 이 과정에서 개발 실력이 향상됨을 느낄 수 있었다. 본 프로젝트에서는 Node.js 기반의 서버를 구성했는데 이와 통신하기 위해 HTTP 메서드를 사용했고 이에 대한 설계 및 사용 방법, 사용 시점 등에 대해 이해하게 되었다. 또한 협업 툴로 GitHub를 사용함으로써 협업의 중요성과 경험의 차이에 대해서 다시 한 번 깨닫게 되었다. 원활한 커뮤니케이션과 협력이 프로젝트 성공의 중요한 요소임을 다시 한번 확인할 수 있었다.

### 11.3. 지성원

저번 방학부터 달려온 졸업작품 프로젝트가 마침내 완성되었다. 우리 팀이 기획한 주제를 잘 구현한 것 같다. 기획 당시에 주제의 넓은 파트의 폭으로 시스템이 유기적으로 연결되어 잘 동작할 수 있을까 걱정을 했었다. 하지만 팀원들과의 소통과 협업을 통해 프로젝트를 유연하게 진행할 수 있었다. 오히려 각 팀원들의 파트가 뚜렷하게 분담되어 팀원들이 각 파트를 전담하며 자신의 역할에 최선을 다할 수 있었고, 서로간의 협업도 잘 진행되었다. 프로젝트의 AP Server 파트를 맡으며 웹 서버 구현은 지난 경험들로 큰 문제가 되지 않았다. 다만 나머지 모든 파트와의 협업은 필수였고, 이를 위해 많은 노력이 필요했다. AP Server는 서비스의 프로세스가 유기적으로 동작하기 위한 각 파트들의 연결 다리 역할을 수행했기 때문이다. 모바일 어플리케이션과는 통신의 프로토콜과 웹서버와의 구조를 맞춰나갔고, JT 기기와도 통신 프로토콜을 맞춰나가고, 머신러닝 모델을 서버에서 실행할 수 있도록 연결하였다. DB Server와도 주소와 암호를 맞추며 성공적으로 연결되었다. 깃허브도 활용하며 협업과 폭 넓은 파일 구조를 관리함에 경험을 쌓을 수 있었다. 공부한 기술의 깊이보다 팀원들과 소통하고 협력하며 같이 문제를 해결해나간 경험들이 더 값진 가치라고 생각한다. 팀원 모두가 자신의 역할에서 최선을 다하였고, 한 명의 공백이라도 있었다면 이 프로젝트를 완성하지 못했을 거라 생각한다. 이 경험들은 앞으로의 공부와 진로에 큰 도움이



될 것이라 생각한다.

#### 11.4. 허진환

Smart Recycling System의 최종 목표는 단순 졸업작품만이 아닌 하나의 시스템으로서 개발되었다. 이에 따라 다양한 분야와 기술 스택이 필요하므로 설계 과정 중 각 팀원이 파트를 전문적으로 맡아 분업하였고, 큰 문제 없이 제작을 마칠 수 있었기에 프로젝트 진행에서 가장 중요한 팀워크의 부분에서 분업의 중요성을 재고할 수 있었다. 본 프로젝트의 System Infra Structure를 설계할 때 가장 중요하게 생각했던 부분은 확장성이다. 단순 졸업작품이 아닌 시스템을 설계하기에 확장성은 필수 요인이었으며, 이에 따라 Amazon Web Services에서 제공하는 확장 가능한 클라우드 컴퓨팅 서비스인 AWS EC2 Instance로 DB서버를 구축하기로 하였다. 이때 DB서버에 적재되는 데이터 정보는 기밀성과 무결성이 보장되어야 하는 중요한 데이터 객체로서 신뢰할 수 없는 불특정다수 Client들의 접속이 이루어지는 Web Server와 분리-이원화하여 구성하여야 한다. 한편, Web Server의 경우에도 확장성이 요구되어지는데, Node.js로 작성되어 npm 패키지를 사용하여 만들어졌으므로 이식성이 높은 특성을 가져 확장성이 필요할 때 보다 확장성을 갖춘 플랫폼에 맞추어 환경설정만 바꾸어 주면 되도록 하였다. 이에 더불어 AP-DB서버 이원화가 가지는 장점으로 성능향상, 안정성 등이 있어 더욱 시스템이 커졌을 때 결국 이점을 가질 수 있도록 설계하였다. 그러나 시스템이 확장된다는 것은 보다 높은 보안 위협 요소를 갖게됨을 의미한다. 그렇기에 기밀성과 무결성이 보장되어야 하는 중요한 데이터가 저장되는 DB서버와 DBMS(MySQL)에 보안 체크리스트(정책)를 작성하였고, 이 정책에 따라 접근통제 등의 기술적 보안을 실현하여 기존 취약점을 조치하였으며, DB서버(CentOS7)에 대한 보안 취약점 점검 스크립트를 실행하여 이에 대한 취약점을 조치하였다. 결론적으로, 본 프로젝트를 실시함에 있어 보다 확장성을 갖춘 시스템을 설계하는 목표에 있어 학부생으로서 본 졸업작품에서 어느정도 달성하였으며, 실무에서는 훨씬 더 큰 시스템을 운영하기에 앞으로도 본 프로젝트와 같은 목표를 잡고 자기계발에 나서겠다는 생각을 하였다.

#### 부록

- 1) 회의록: 부록1\_종합설계\_에코아리움\_회의록.pdf
- 2) 보고서: 부록2\_종합설계\_에코아리움\_DB서버보안점검리스트.pdf
- 3) 소스코드(GitHub Repository Link)

링크: [SKU-Ecoarium https://github.com/mixgolem/SKU-Ecoarium.git](https://github.com/mixgolem/SKU-Ecoarium.git)