

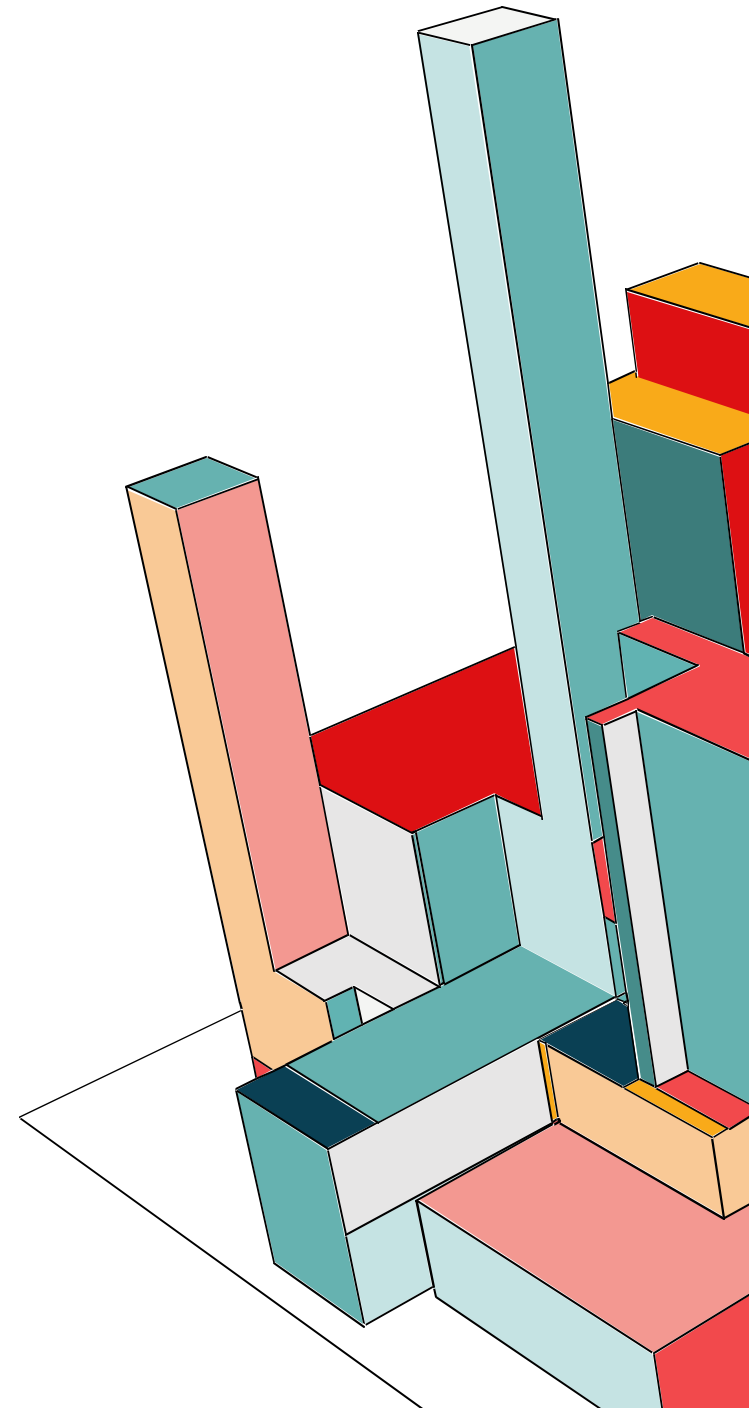
# 도돌이

< 해보조 >

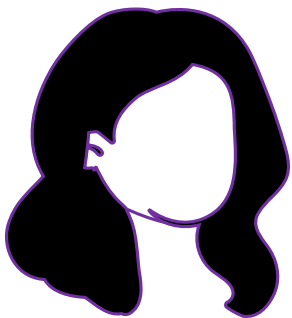
정우영, 이소민, 김태훈, 정재경

# 목차

- I. 팀원소개
- II. 도돌이 선정 이유
- III. 시장조사
- IV. 도돌이 소개, 핵심기술
- V. 일정 및 팀원 역할 분담



# 팀원 소개



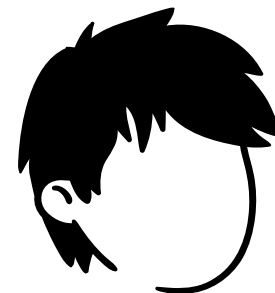
정우영



김태훈



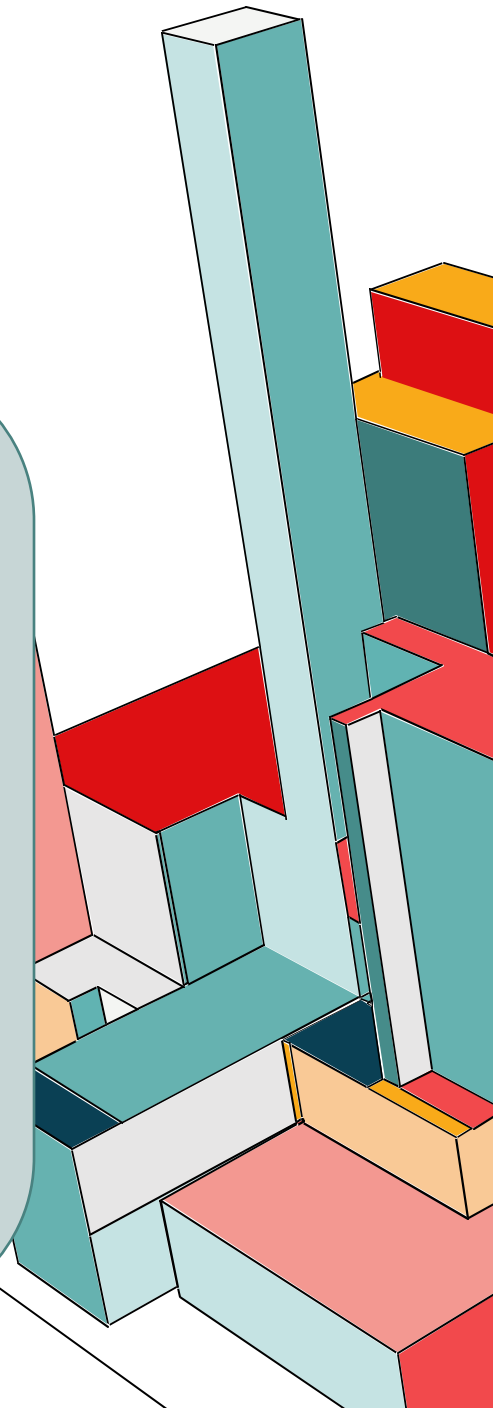
이소민



정재경

# 도돌이 선정 이유

- COVID-19 유행이 시작된 후 그에 따라 비대면 무인 상점들이 증가
- 현재시대의 무인으로 상점을 운영하는 것이 대세
- 가장 비율 높은 편의점 선정이 시장성이 가장 크다고 판단
- 강화시 어떠한 분야에서도 적용 가능



# 시장 조사

- 19 ~ 22년 3년 사이 급증
- 고용주들 무인점 선호 큰 이유  
인건비 부담
- 최저임금은 점점 증가하는 추세

5160원 증가

## 자영업자 무인점포 선호하는 이유



※ 복수응답 ※ 출처: 잡코리아, 알바몬

## 편의점 4사 무인 점포 수 현황

		2019년	2020년	2021년	2022년*
	CU	90개	200개	300개	400개
	GS25	16개	140개	565개	723개
	세븐일레븐	17개	46개	210개	330개
	이마트24	85개	113개	1050개	1330개
	총합	208개	499개	2125개	2783개

\*6월 말 기준

## 연도별 최저임금

단위: 원



# 시장 조사

- 코로나로 인해 2022년 1월 최고

고용률

감소

실업률

증가

- 코로나 재유행으로 더욱더 고용율은 감소, 실업률 증가 예상

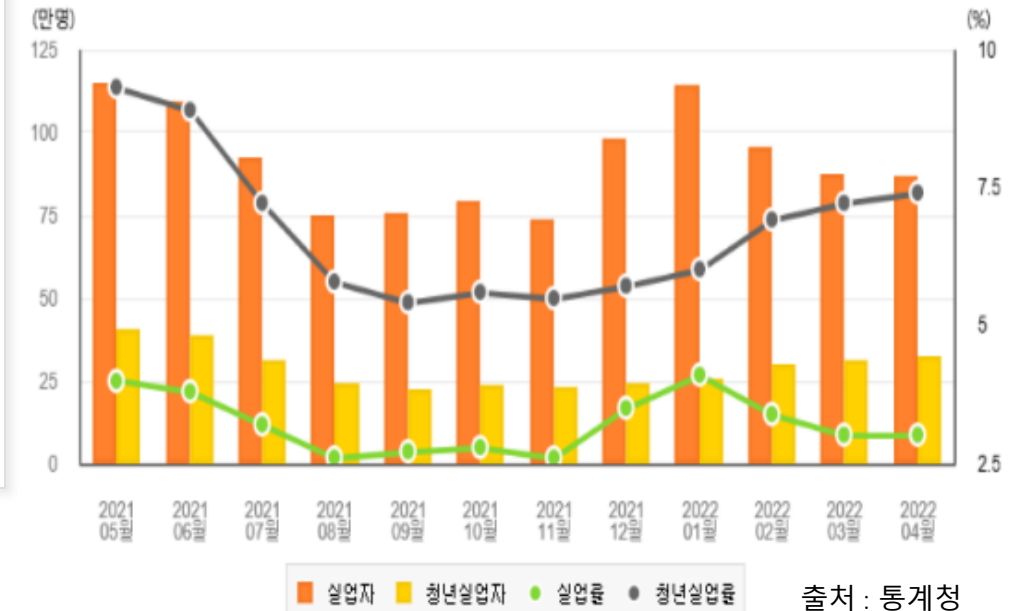
## 취업자 및 고용률 추이

22년 4월 15~64세 고용률 68.4%  
15세 이상 취업자 2807만8000명



## 실업자 및 실업률 추이

출처 : 통계청통계청 / 대한데일리

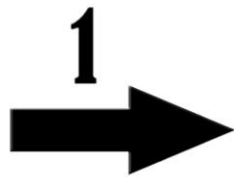
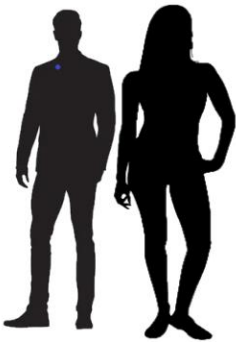
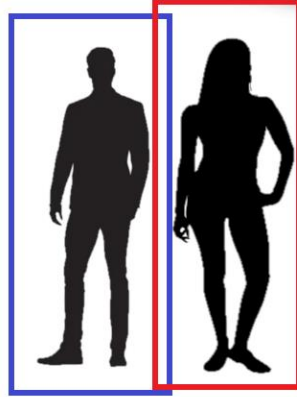


출처 : 통계청



로드셀 센서

객체추적  
→



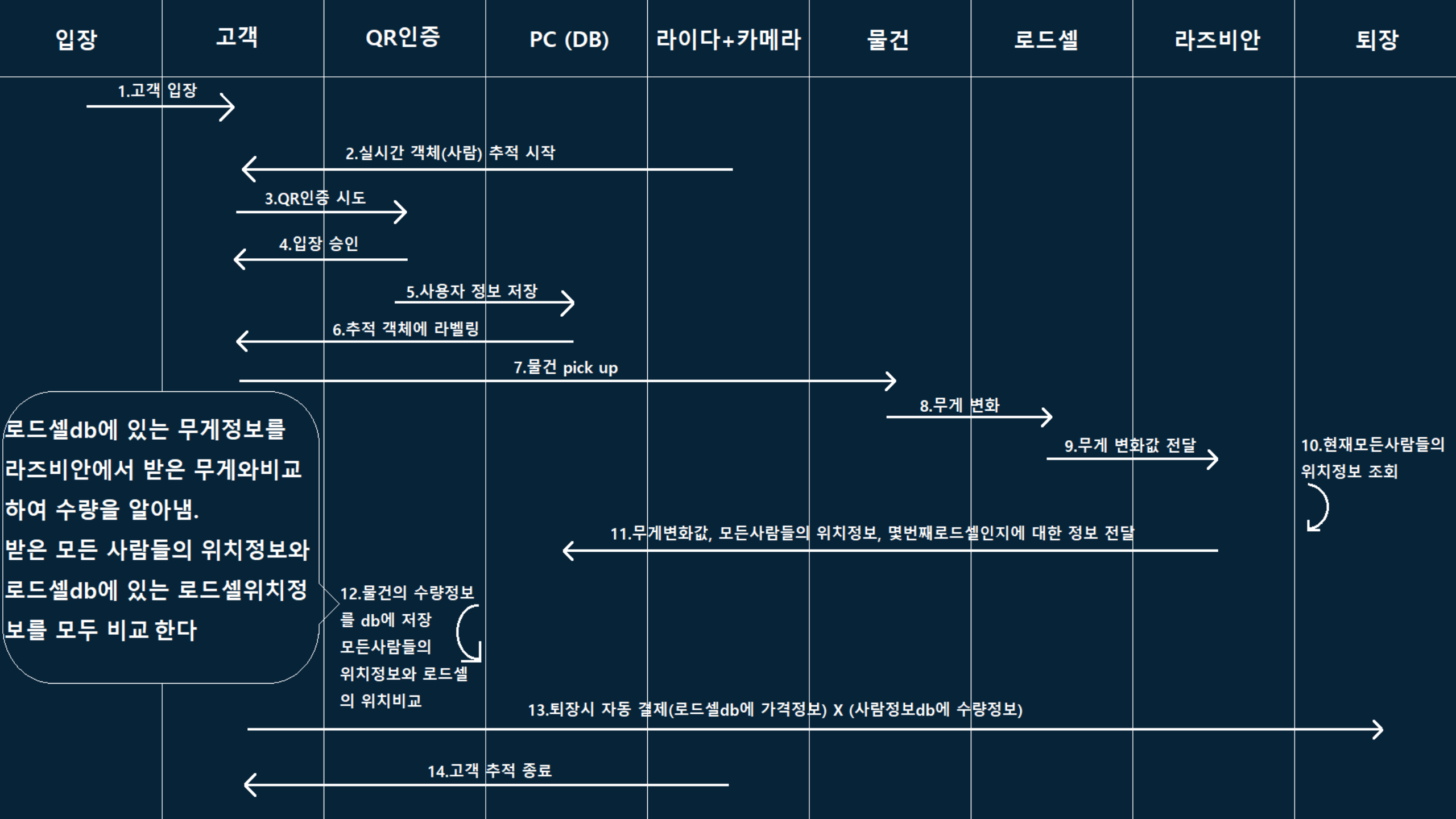
DB

콜라 3▷2  
콜라가격 1500₩  
지불할 금액  
 $1500 \times 1 = 1500₩$



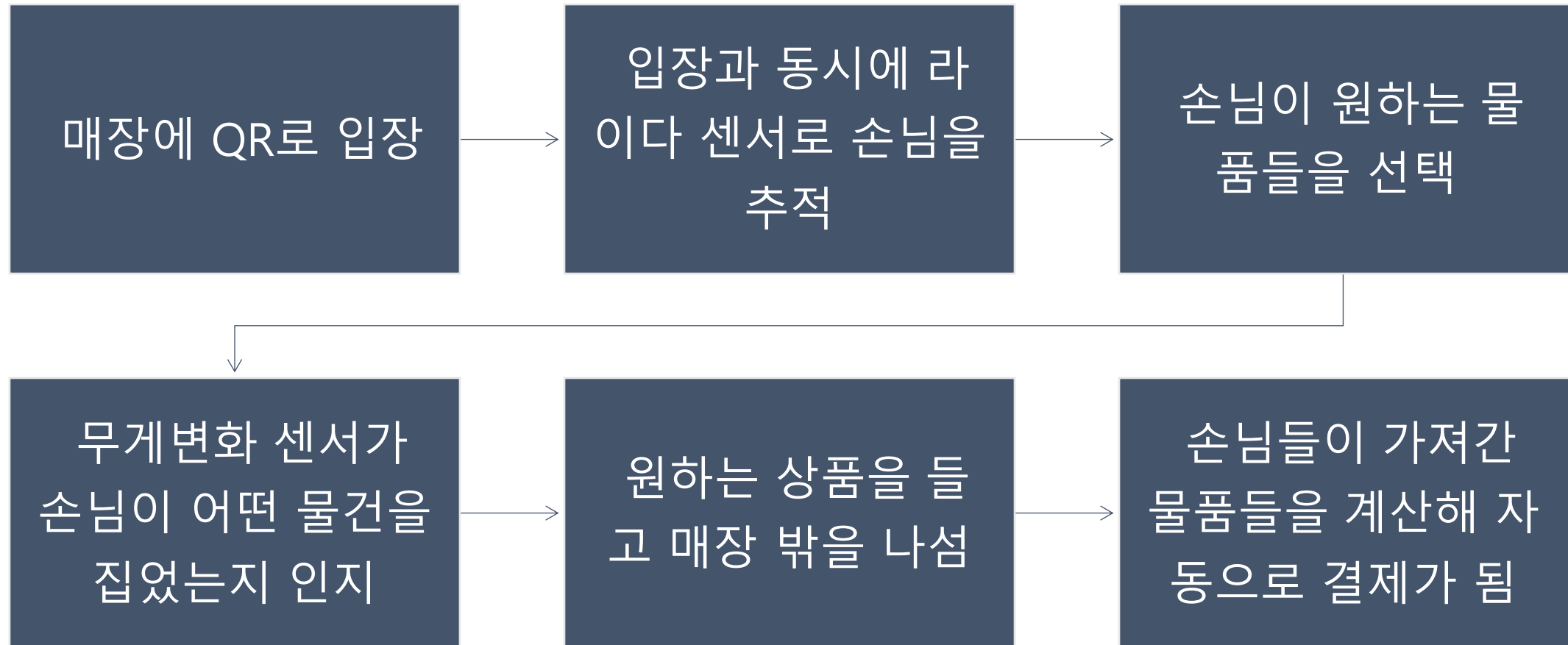
자동결제  
1500₩



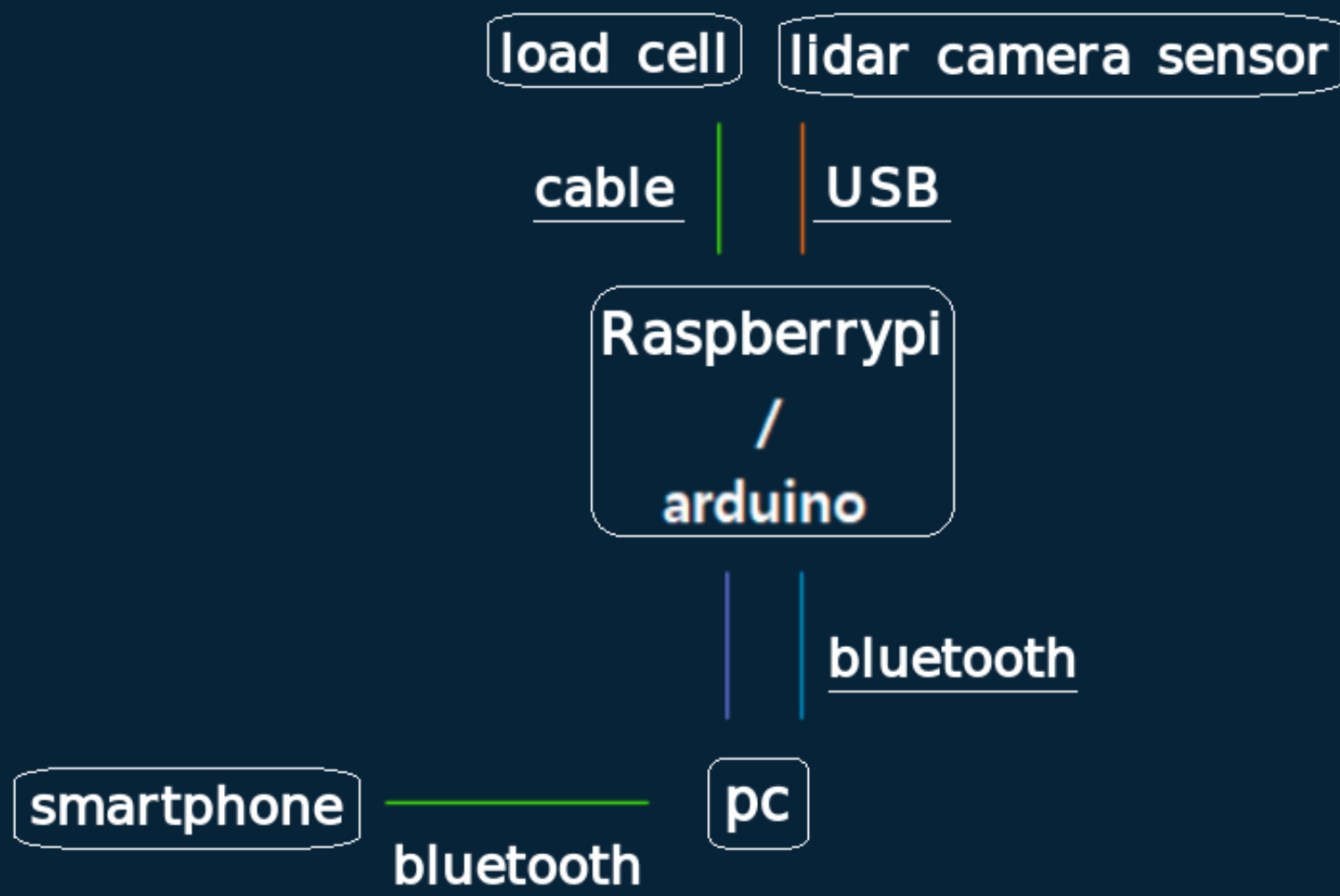




# 사용자 시나리오



# HARDWARE ARCHITECTURE

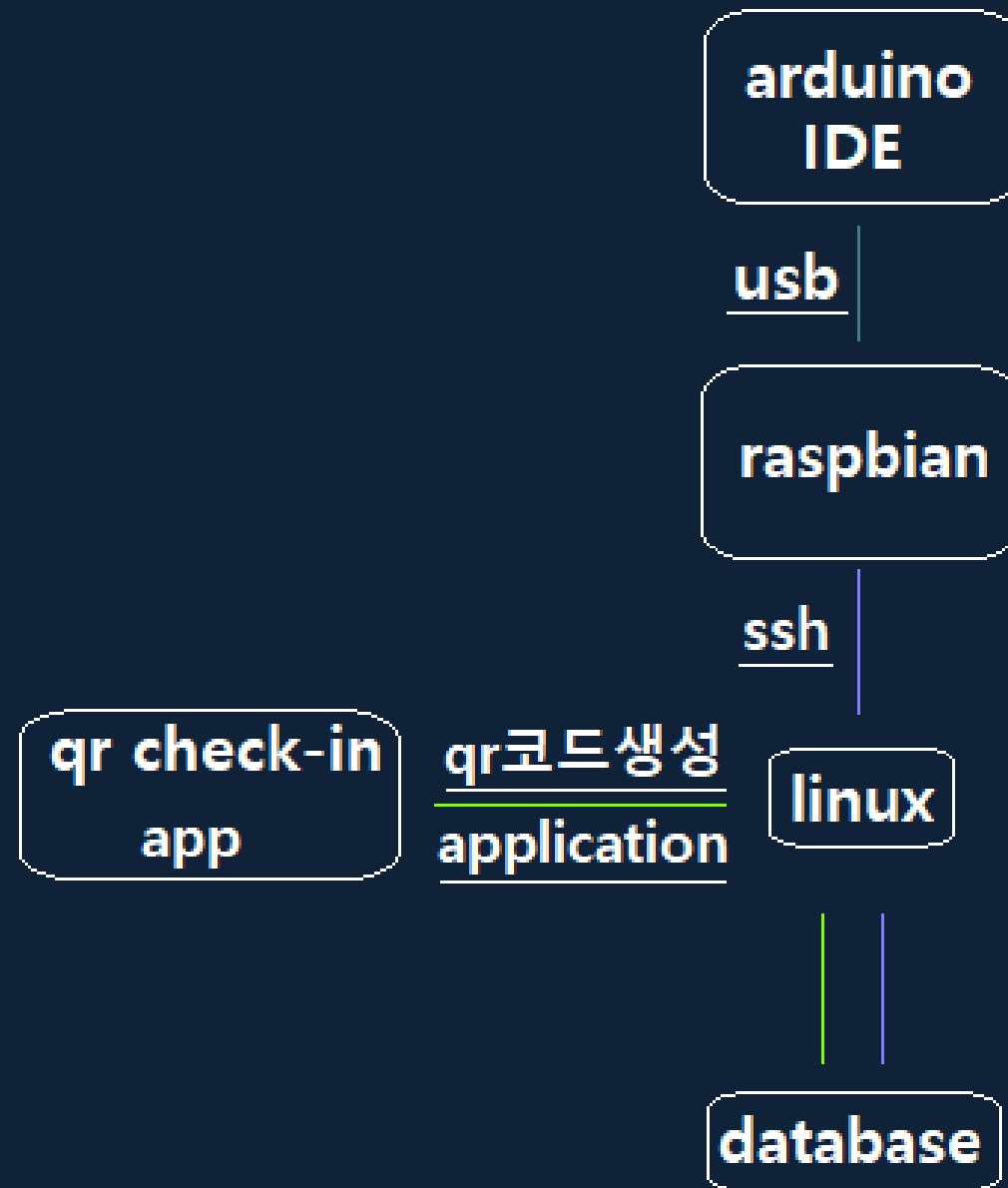


# Software

## 시나리오



# SOFTWARE ARCHITECTURE



**LOADCELL**



30000₩

**Raspberry Pi 4B – 4GB**



95000₩

**Raspberry Pi Cooler**



15000₩

**선반**



10140₩

**5V충전기**



3900₩

**실습자제 예산**



- 라즈베리파이 3B+ / 2개
- 2D 라이다 센서 / 1개
- 3D 라이다 센서 / 1개  
– (realsensor)
- 3D 라이다 센서 케이블 / 1개

실습자제 대여

# 핵심기술

---

## 카메라 센서

-----

YOLO 기술로  
객체를 추적한것을  
opencv로 실시간  
비교하는 기술

## 무게센서를 활용한 물건의 갯수 추정

-----

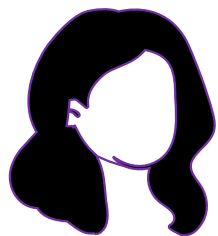
라즈비안에 연결된  
로드셀에 무게가 감소하면  
물건의 갯수가 얼마나  
남았는지 알수있는 기술

## DB에 실시간으로 정보저장

-----

매장입장=>insert  
매장퇴장=>delete  
물건픽업=>update  
퇴장시결제=>select

# 팀원



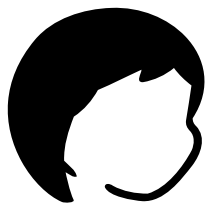
정우영

- OpenCV, 딥러닝
- PPT



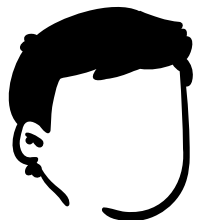
김태훈

- DB
- 아키텍처 부분 관리



이소민

- IOT
- 조별 스케줄 관리



정재경

- GUI, ROS
- 추가 조사 담당



## 팀 일정

[illegible]



# 7월

Sun	Mon	Tue	Wed	Thu	Fri	Sat
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
				팀별 주제 및 스케줄 발표 준비		
17	18	19	20	21	22	23
		1차 발표	요구분석 - 구현 될 시스템 기능이나 목표, 제약사항 등 정확히 파악,			
24	25	26	27	28	29	30
요구분석-목적 파악(기능, 성능, 사용 편의성, 이식 등 목표 시스템 품질)			설계 - 분석된 결과를 어떻게 프로그램으로 구성할 것인지.			

## 8월

### 팀 일정

Sun	Mon	Tue	Wed	Thu	Fri	Sat
31	1 휴가	2	3	4 설계 - 시스템 구조 설계, 프로그램 설계, 사용자 인터페이스 설계 → 설계서 완성.	5	6
7	8	9	10	11	12	13
학습 및 구현 - 미리 <u>정해진</u> 모듈 설계에 의하여 프로그래밍.						
14	15	16	17	18	19	20
학습 및 구현, <u>외부업체</u> <u>중간미팅</u>						
21	22	23	24	25	26	27
<u>테스팅</u> - 각 모듈들 <u>테스팅</u> 후 전체 시스템이 사용자 요구와 분석 내역에 적합하게 구현되었는지 시험.						
28	29	30	31	1	2	3
인수/설치 - 설치 후 인수를 받는 사용자나 발주자가 시험(미니어처로 대체)						



## 9월

Sun	Mon	Tue	Wed	Thu	Fri	Sat
28	29	30	31	1	2	3
인수/설치 - 설치 후 인수를 받는 사용어나 발주자가 시험(미니어처로 대체)						
4	5	6	7	8	9	10
최종 발표 1차 완성		<u>유지보수</u>		최종 발표		
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

# 개인일정



정우영

[illegible]





# 개발계획

ubuntu 18.04

- 전체 구현 OS

OpenCV – cv3  
Python

딥러닝 – tensorflow, – 학습시키기  
( darkflow 고려중 )



# 이슈 및 트러블 슈팅 - 1주차

```
habojo@zen:~$ gazebo
[Err] [REST.cc:205] Error in REST request

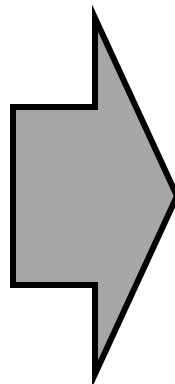
libcurl: (51) SSL: no alternative certificate subject name matches target host name 'api.ignitionfuel.org'
^Z
```



```
---
# The list of servers.
servers:
-
  name: osrf
  url: https://api.ignitionfuel.org

# -
# name: another_server
# url: https://myserver

# Where are the assets stored in disk.
# cache:
#   path: /tmp/ignition/fuel
```



```
---
# The list of servers.
servers:
-
  name: osrf
#   url: https://api.ignitionfuel.org
  url: https://api.ignitionrobotics.org

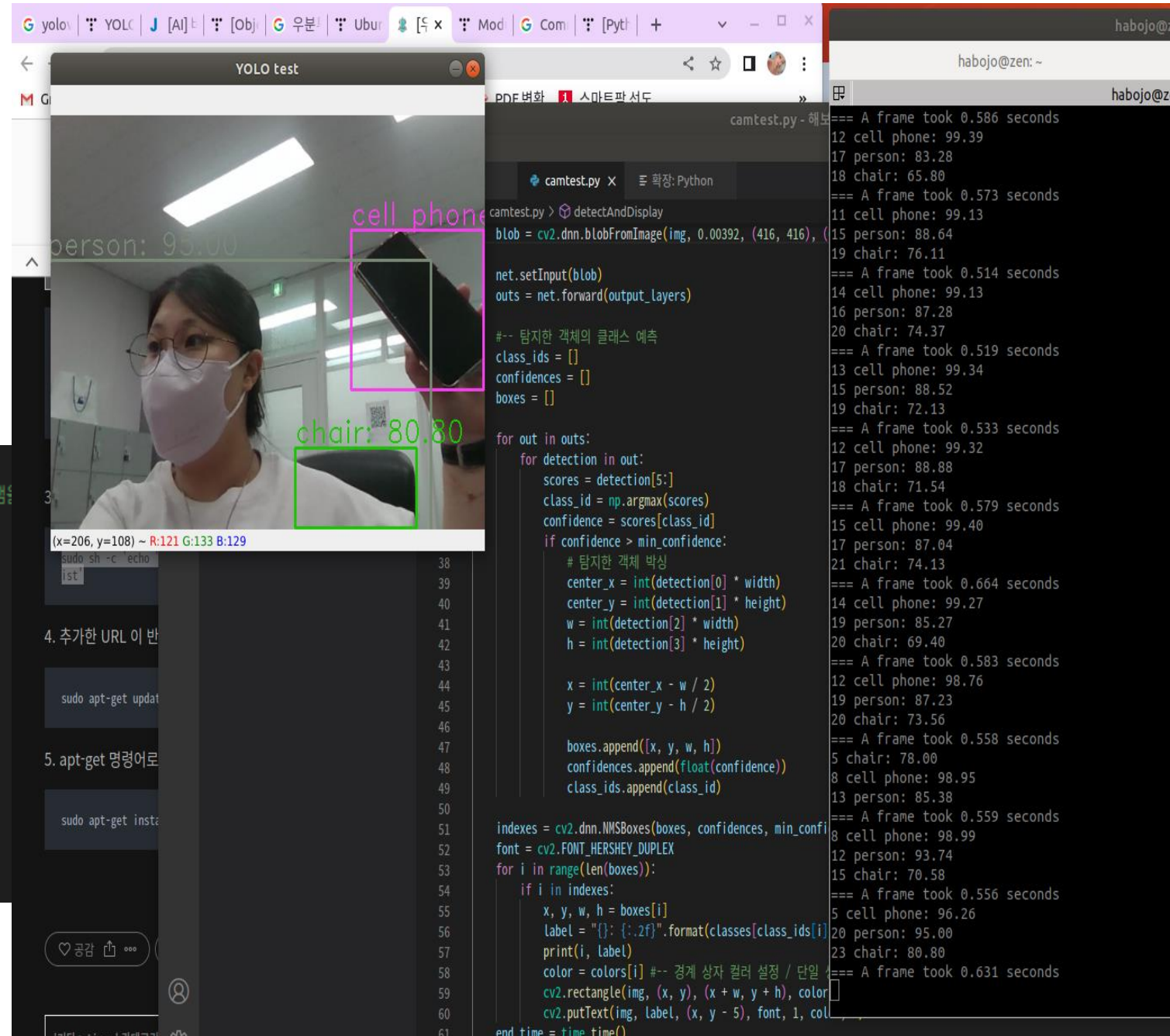
# -
# name: another_server
# url: https://myserver

# Where are the assets stored in disk.
# cache:
#   path: /tmp/ignition/fuel
```

# 이슈 - 2주차

- 웹캠을 사용시 Vedio\_path의 값은 4
- 내장캠을 사용시 Vedio\_path 값은 0

```
## 비디오 활성화
cap = cv2.VideoCapture(0) ## 웹캠 사용시 vedio_path를 0 으로 변경 / 웹캠을 쓸때는 왼쪽 4번 사용 , 내장캠을
if not cap.isOpened():
    print('--(!)Error opening video capture')
    exit(0)
while True:
    ret, frame = cap.read()
    if frame is None:
        print('--(!) No captured frame -- Break!')
        break
    detectAndDisplay(frame)
    ## q 입력시 종료
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
cv2.destroyAllWindows()
```



# 이슈 및 트러블 슈팅 - 2주차

- 오픈소스 다운받아 학원 PC에서 실행 Test 실패
  - 해당 파일들 같은 위치에 존재 하지 않았음
- > opencv 설치하였으나 cv2모듈이 없다고 해당 에러가 계속 됨
- > 개인 노트북으로 다시 Test 재진행, 성공
- 원래 계획은 Realsense로 객체추적
  - Rviz등 실행 Test 실패
- 매일 github에 업로드 할 계획이었지만 매일 까먹게 됨
  - > 알림 설정을 따로 해두어 매일 한 일들 업로드 및 정리할 필요성 존재
- Darkflow에 관해 좀더 관련 지식이 필요
- 오픈소스에서는 객체 구별을 80개만 가능
  - > 추가적으로 필요한 부분은 추가 학습 예정
- 웹캠에서 객체를 분류해서 인식
  - > 저장 후 DB에 어떻게 통신할 것인지 고민 필요

# 개인일정



이소민

[illegible]







# 이슈 및 트러블 슈팅 - 2주차

- 설계 부분이 끝난 뒤 로드셀이 필요한데 배송 하루 전 인 상황.

: 온습도 센서로 시리얼 통신 단위 test.

- 블루투스 모듈과 PC 간 연결이 구글링해도 잘 안 나옴.

: 조원의 도움으로 NodeMCU의 WIFI를 이용해 연결하는 방법 알게 돼서 변경.

- 로드셀 4개 중 2개가 불량?

: 추후 재 납땜 후 재구매/ 1칸에 로드셀 1개씩?

- 로드셀 무게 측정 판 주문 제작?

: SW 개발이 우선이라는 생각이 들어 일단 보류 중.

- 원래 잘 동작하던 코드가 아두이노 IDE 에서 업로드도 잘 되는데 이상하게 동작.

: 한참 헤매다 USB 포트 위치 바꿔서 꽂아보니 잘 됨.

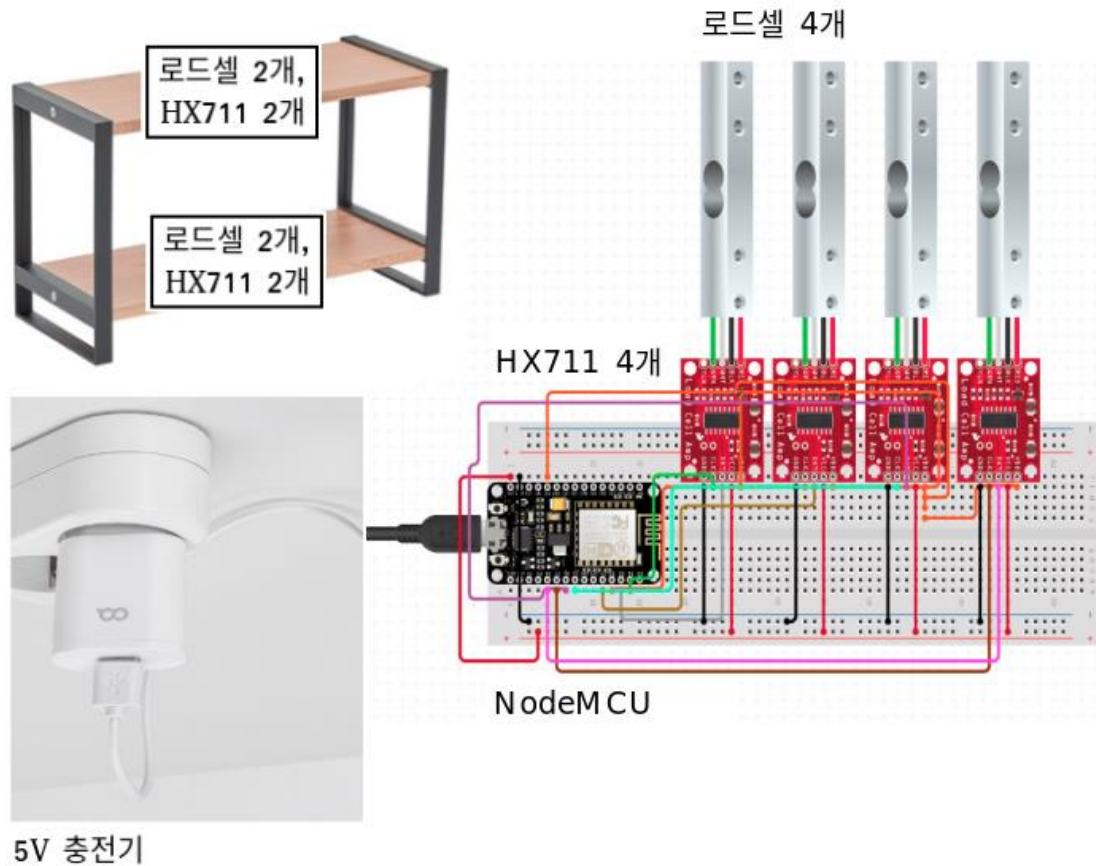
- 금방 하고 넘어갈 수 있을 것 같았던 부분에서 일단 넘어가고 다른 거 하기 실패.

: 막히면 붙잡고 있지 말고 다른 거 먼저 하기!

- 8/19까지 구현하는 게 목표인데 그때까지 하루하루 미리 계획하기 불가능?

: 매일 그 날 해야할 일을 조금 욕심내서 정해놓고 다 해보려 노력. 평균 절반 정도 진행되는 듯.

# 개발계획



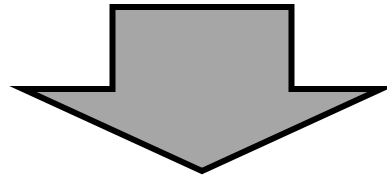
ubuntu 18.04  
~~CoolTerm (시리얼 통신 프로그램)~~  
➢ named pipe  
python  
DB(SQLite3)

=> 로드셀 4개, HX711 4개로 4가지 제품 구별 가능하게



[illegible]

# 1주차

[illegible]



# 개발계획 및

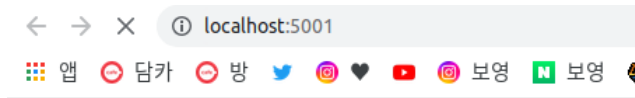
sqlite3  
python  
flask

트러블 슈팅 - 듀얼 부팅

이슈 - DB 설계전에 테이블 정의를  
위한 데이터의 흐름 파악

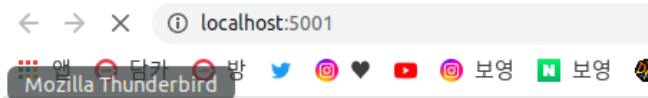
# 이슈 - 2주차

```
fhxlvnf@kth:~/pythonTest$ python3 test.py
* Serving Flask app 'test' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://localhost:5001/ (Press CTRL+C to quit)
```



Home

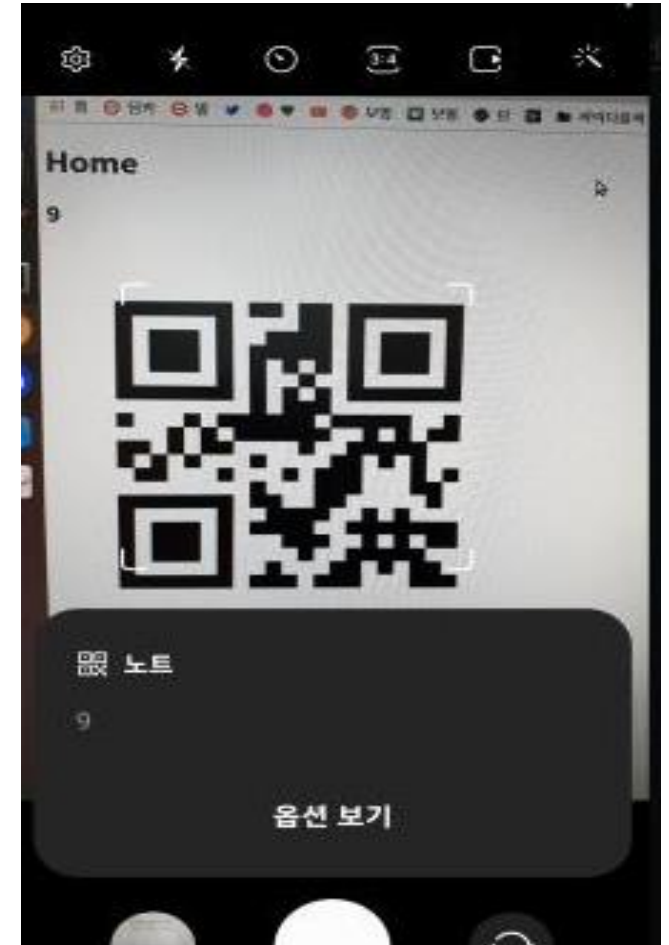
4



Home

7

터미널



# 개인일정



## 정재경

[illegible]

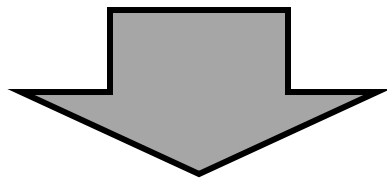
[illegible]

# 1주차

[illegible]

[illegible]

# 2주차

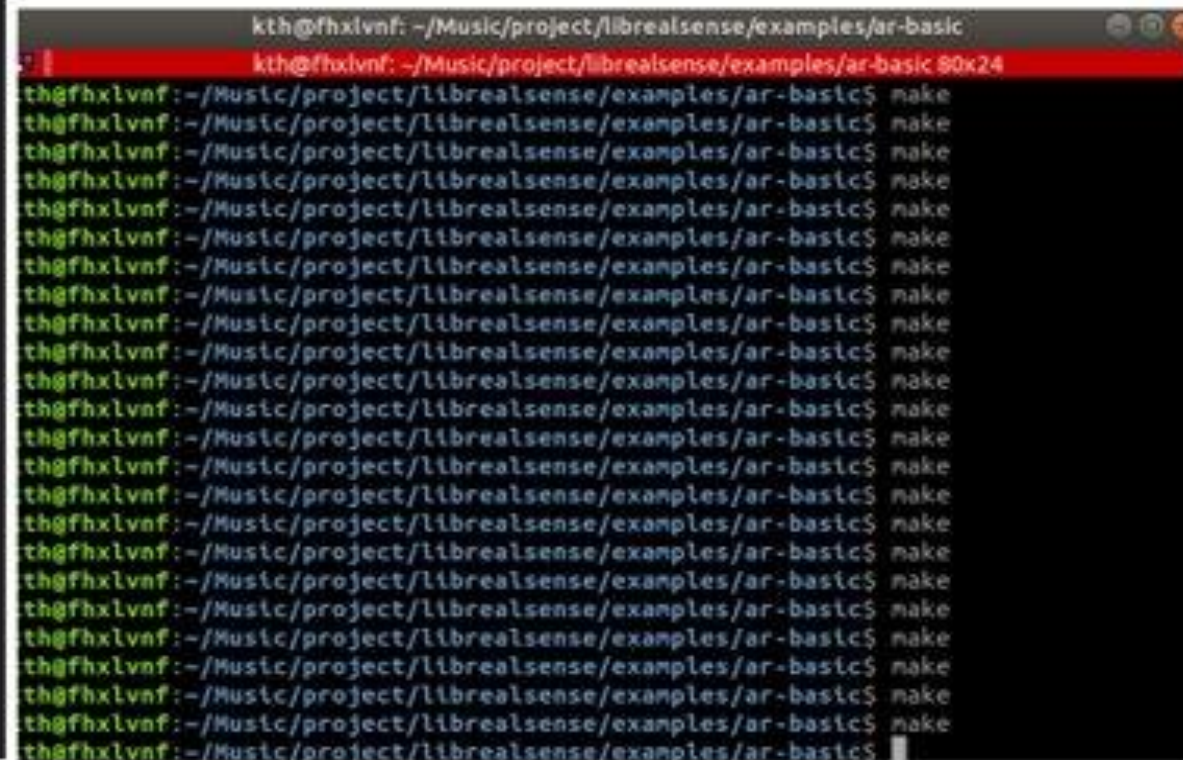
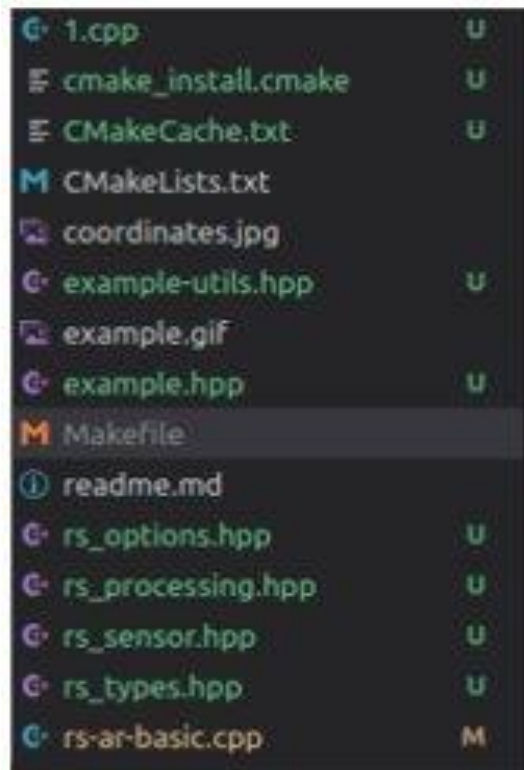
[illegible]

# 개발계획

qt designer  
qt tool  
Android  
rivz

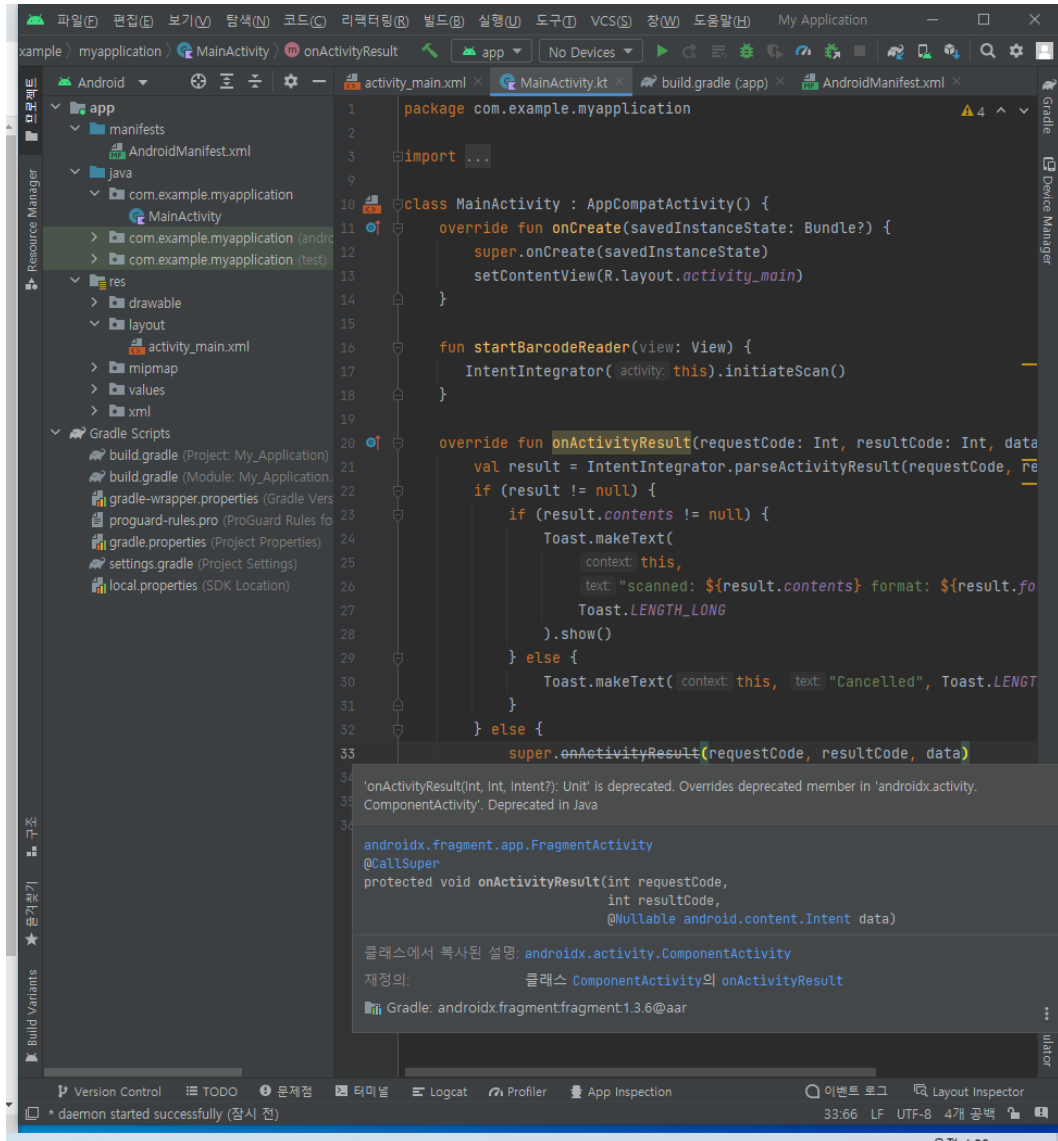


# 이슈 및 트러블 슈팅 - 1주차



```
kth@fhxlvnf:~/Music/project/cpp/build$ make
-- Configuring done
-- Generating done
-- Build files have been written to: /home/kth/Music/project/librealsense/examples/ar-basic
```

# 이슈 - 2주차



써야하는 코드중에 하나가  
이제 쓸 수 없는 코드가 됨.  
그래서 대체로 쓸 수 있는 코드를 찾아서 공부  
하다가 포기

onActivityResult  
->  
registerForActivityResult  
공부  
->  
포기

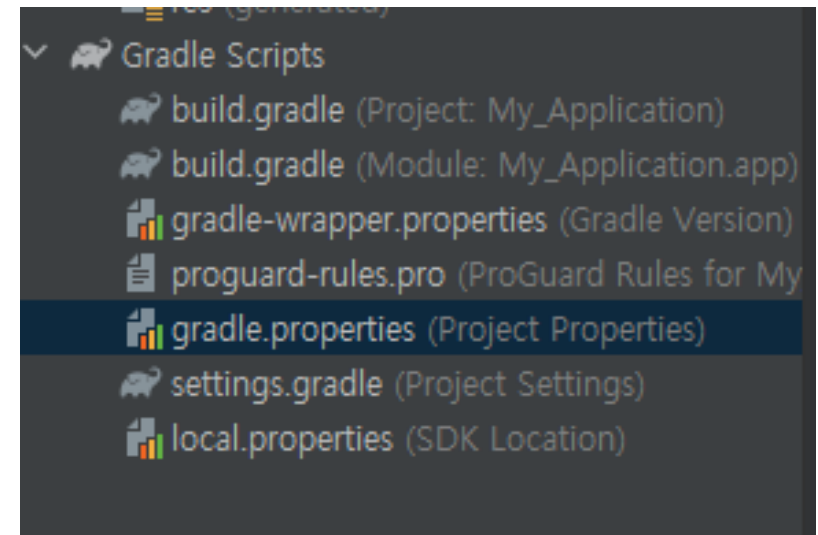
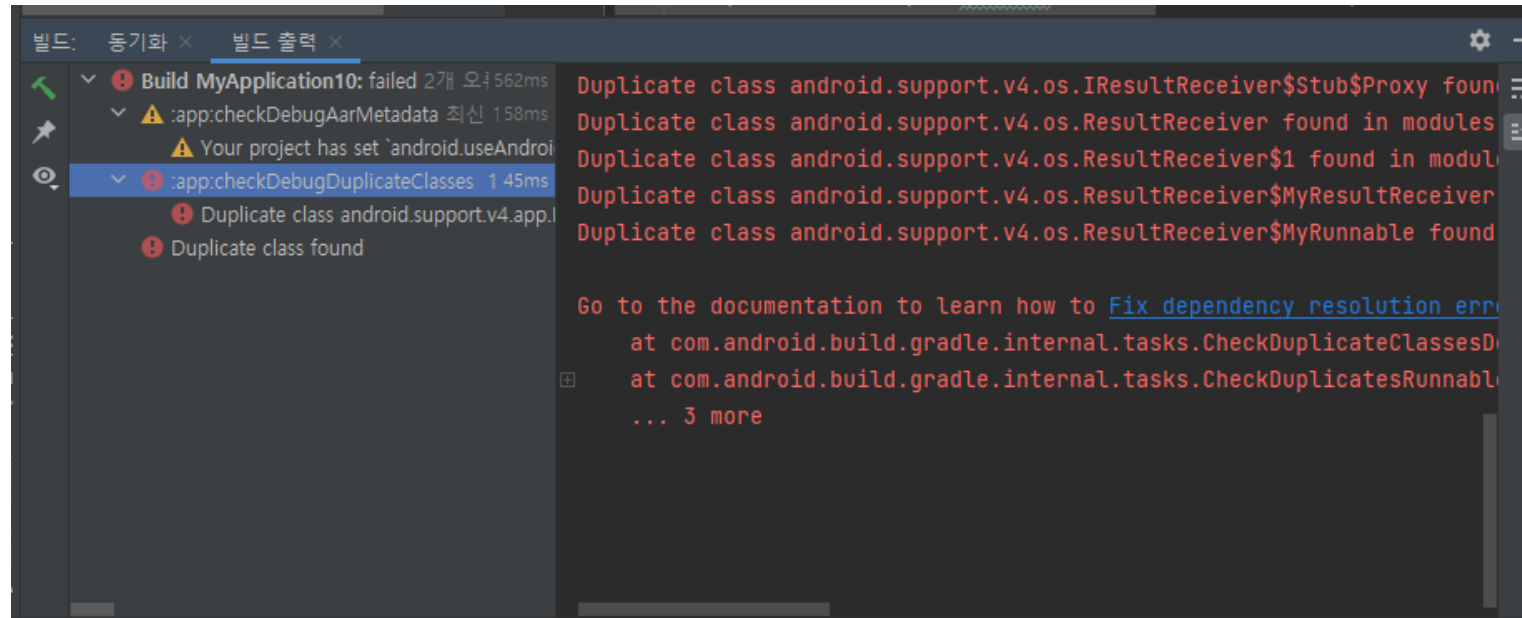
# 이슈 - 2주차

Duplicate class found 오류는  
클래스를 두 번  
중복해서 쓸시에 생기는  
오류라는데 쓴곳이 없었다  
그래서 일단 경고부터 고쳐보기로  
마음먹음

your project has set  
android.useandroidx=true but  
configuration

라는 오류여서  
android.useandroidx 라는 항목이  
어디에 존재하는지 찾아봄

android.useandroidx=true는  
위 사진에 보이는  
gradle.properties 항목에  
있었음

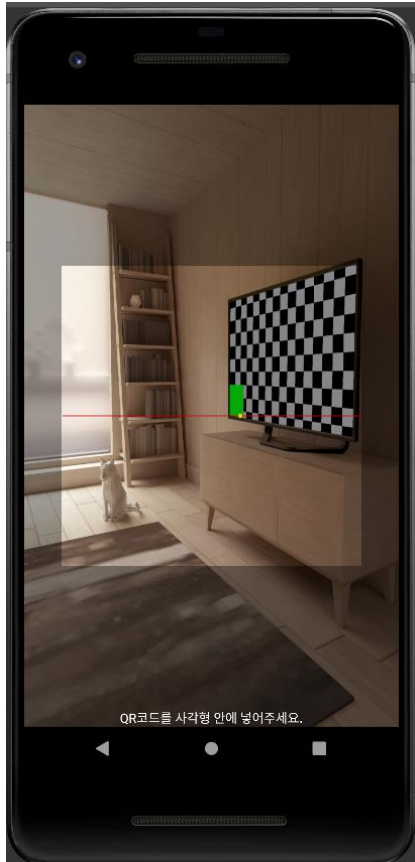


# 이슈 - 2주차

```
android.useAndroidX=true
# Enables namespacing of each library's R class so that its R class includes only the
# resources declared in the library itself and none from the library's dependencies,
# thereby reducing the size of the R class for that library
android.nonTransitiveRClass=true
android.enableJetifier=true
```

구글링 해보니 android.enableJetifier: true 를 추가 할시에  
고쳐진다고 하여서 추가 하여서 고침

오류의 이유는 android.enableJetifier: true 플러그인이 없을시에  
AndroidX 를 false 로 인식한다고 한다 .



Android studio 로 시뮬레이션은 성공  
C타입 연결 선 과져와서 앱으로 만들예정

감사합니다.

- 해보조

