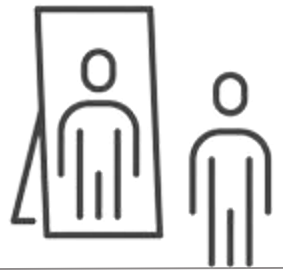
2nd Semester Graduation Project

Smart Mirror for Virtual Fitting



201735894 Choi Hyeongin

201835486 Lee Minseo

201935098 Lee Sojeong

Contents

How can we save time on deciding what to wear everyday?

Let's recommend styling with **Smart Mirror**

- It provides **virtual fitting**
- It can reduce the time to choose or purchase clothes

According to the 'Mobile Shopping Trend Report 2020' released on August 3 by Mobile Research Open Survey,

오프라인 구매 이유

		Gap (vs. 19년)	남성 (382)	여성 (411)	20대 (271)	30대 (257)	40대 (265)
상품을 직접 착용/확인할 수 있어서	76.0	-0.2	73.8	78.1	77.5	74.7	75.8
배송/운반이 빨라서/편리해서	28.4	-3.8	27.7	29.0	31.0	31.1	23.0
편리해서	24.5	+0.6	24.3	24.6	24.7	23.7	24.9
쇼핑경험이 좋아서	18.3	-0.4	19.9	16.8	14.8	17.9	22.3
안전성에 대해 신뢰할 수 있어서	9.1	+2.0	7.6	10.5	12.9	5.8	8.3
고객서비스가 좋아서	8.4	+2.6	9.7	7.3	8.5	9.7	7.2
상품구성이 좋아서	7.6	+1.8	7.1	8.0	6.3	7.4	9.1
가격/가치가 좋아서	6.7	-0.9	5.5	7.8	4.8	7.0	8.3
결제가 편리해서	4.8	-1.2	6.3	3.4	4.1	5.1	5.3
멤버쉽 혜택이 좋아서	4.0	-0.7	4.2	3.9	4.1	2.7	5.3

[Base : 3개월 내 오프라인 채널 구매자, N=793, 순위형 응답(1+2순위), %]

Description

Clothing Recommendation Service

- Musinsa store styling + User’s clothes styling
- Clothing recommendation considering the temperature

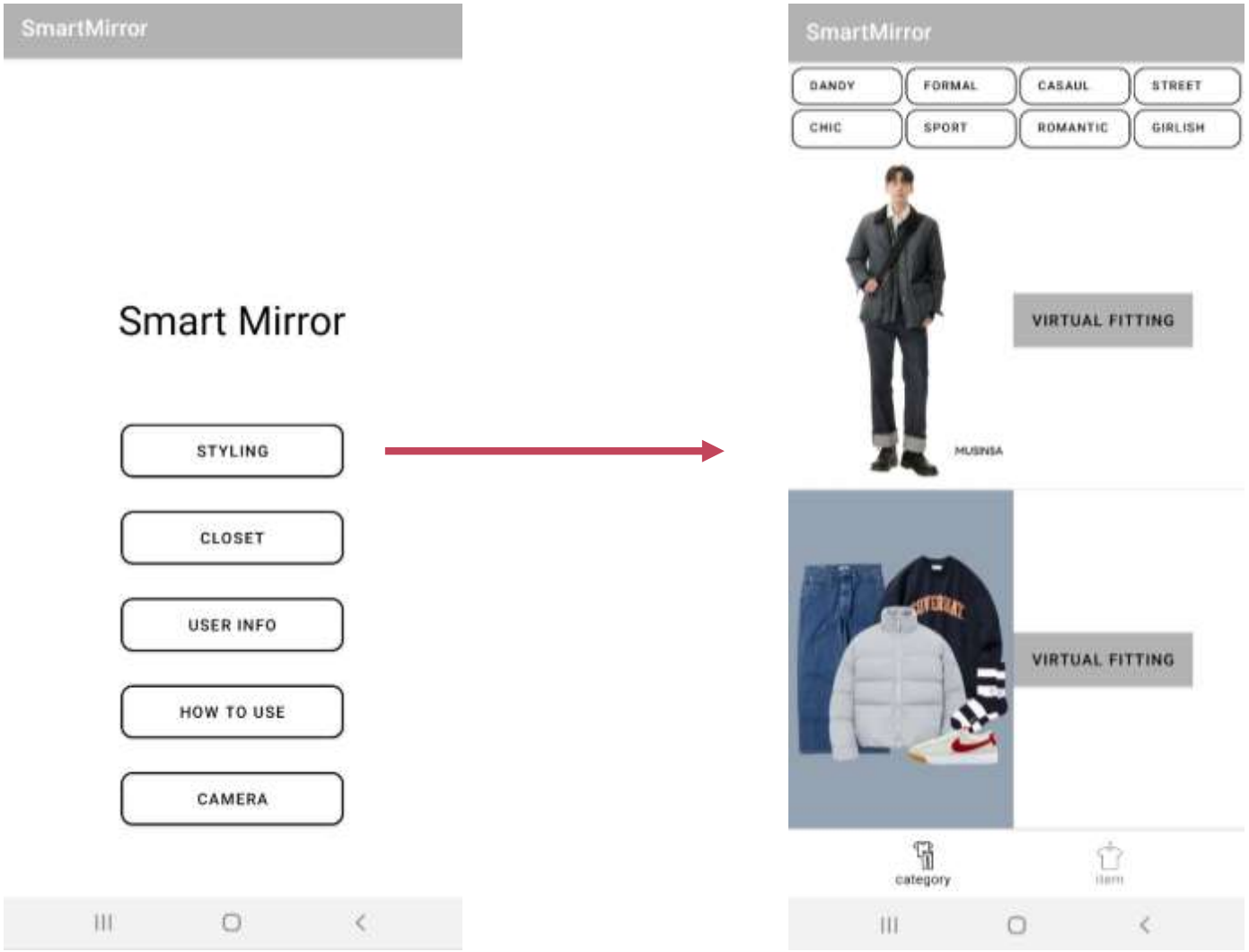
Amount of clothes data: 384 images

Smart Mirror?

- Combines display and mirror
- Provides virtual fitting

Key Features

1. STYLING Button



Key Features

1. STYLING Button

http://52.79.59.24/getData.php?TEMP=7&STYLING=casual

{"Look":[{"ID":"16","Image":"https:#/#/smartmirror-bucket.s3.ap-northeast-2.amazonaws.com#/casual#/styling_11.jpg","Category":"styling","Temp":"7","outerLink":"https:#/#/store.musinsa.com#/app#/goods#/2070932#/0","topLink":"https:#/#/store.musinsa.com#/app#/goods#/685878#/0","bottomLink":"https:#/#/store.musinsa.com#
{"ID":"17","Image":"https:#/#/smartmirror-bucket.s3.ap-northeast-2.amazonaws.com#/casual#/styling_12.jpg","Category":"styling","Temp":"7","outerLink":"https:#/#/store.musinsa.com#/app#/goods#/2211156#/0","topLink":"https:#/#/store.musinsa.com#/app#/goods#/2081723#/0","bottomLink":"https:#/#/store.musinsa.com#
{"ID":"18","Image":"https:#/#/smartmirror-bucket.s3.ap-northeast-2.amazonaws.com#/casual#/styling_13.jpg","Category":"styling","Temp":"7","outerLink":"https:#/#/store.musinsa.com#/app#/goods#/2196442#/0","topLink":"https:#/#/store.musinsa.com#/app#/goods#/2216371#/0","bottomLink":"https:#/#/store.musinsa.com#

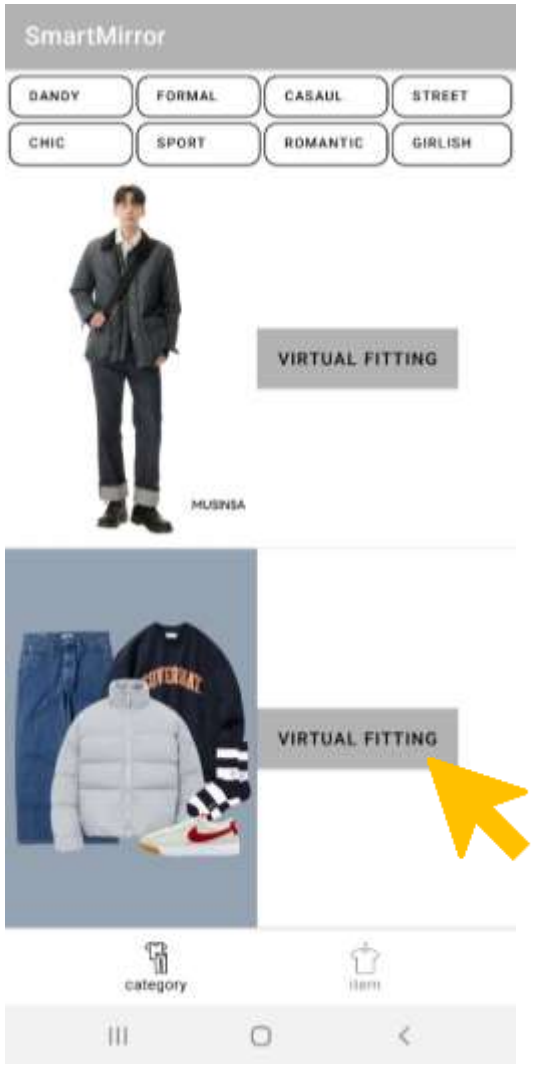
```
E/id:: 16
E/image_url:: https://smartmirror-bucket.s3.ap-northeast-2.amazonaws.com/casual/styling_11.jpg
E/category:: styling
E/temperature_section:: 7
E/outer_link:: https://store.musinsa.com/app/goods/2070932/0
E/top_link:: https://store.musinsa.com/app/goods/685878/0
E/bottom_link:: https://store.musinsa.com/app/goods/1569030/0
```

Detail – Key Features

1. STYLING Button

Key Features

1. STYLING Button




```
Python 3.7.3 (/usr/bin/python3)
>>> %Run socket_android.py

  wait..
  Connected by?! ('192.9.116.235', 65355)
  http://52.79.59.24/virtualFitting.php?STYLING=dandy&ID=1
```

Key Features

1. STYLING Button

VIRTUAL FITTING

Uses haarcascade to distinguish the user's upper and lower body coordinates.

```
cap = cv2.VideoCapture(0) #return 0 or -1

upper_path = 'Downloads/haarcascade_upperbody.xml'
lower_path = 'Downloads/haarcascade_lowerbody.xml'
upperCascade = cv2.CascadeClassifier(upper_path)
lowerCascade = cv2.CascadeClassifier(lower_path)
```

```
lowerRect = lowerCascade.detectMultiScale(imageGray, scaleFactor=1.3, minNeighbors=1, minSize=(1,1))
temp = []
temp2 = []
for x,y,w,h in lowerRect:
    if w>100:
        print("lower: %d, %d, %d, %d"%(x,y,w,h))
        temp = [x,y,w,h]
        roi_color = img[y:y + h, x:x + w]
        upperRect = upperCascade.detectMultiScale(imageGray, scaleFactor=1.3, minNeighbors=1, minSize=(1, 1))
        for lx,ly,lw,lh in upperRect:
            if lw>100 and lx<350:
                print("upper: %d, %d, %d, %d"%(lx,ly,lw,lh))
                ly = 0
                roi_color_u = img[ly:ly + lh, lx:lx + lw]
                temp2=[lx,ly,lw,lh]
```

Key Features

2. CAMERA Button

Take a picture of the user's outfit

↓

Remove background

↓

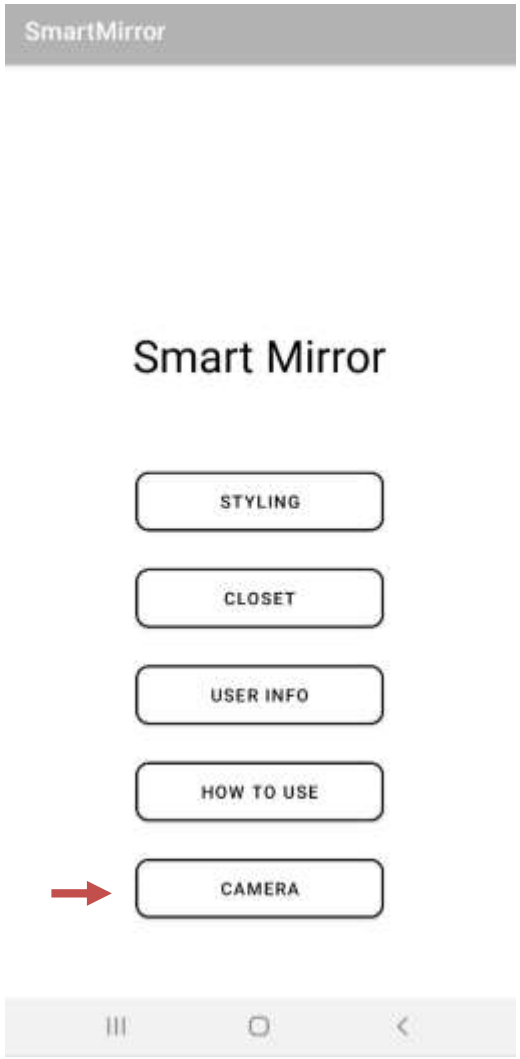
Separate upper and lower body

↓

Use CNN models to upload database

CNN model

bottom_categories.h5
bottom_fit.h5
bottom_length.h5
outer_categories.h5
print.h5
sleeve_length.h5
top_categories.h5



Key Features

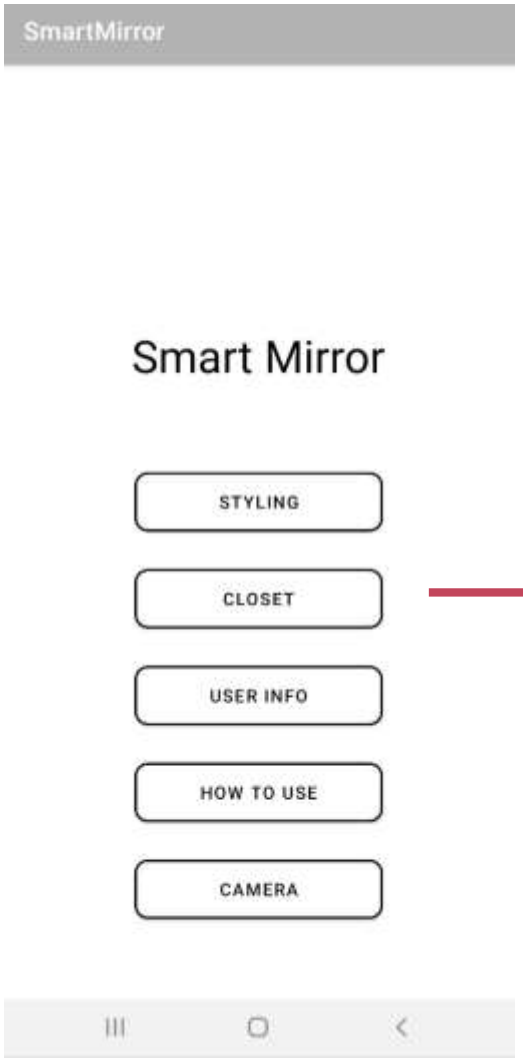
2. CAMERA Button

Remove background

- Edge detection using a structured forest ML model to do edge detection
- Get an approximate contour of the object
- Use OpenCV’s GrabCut algorithm to get a more accurate contour

Key Features

3. CLOSET Button



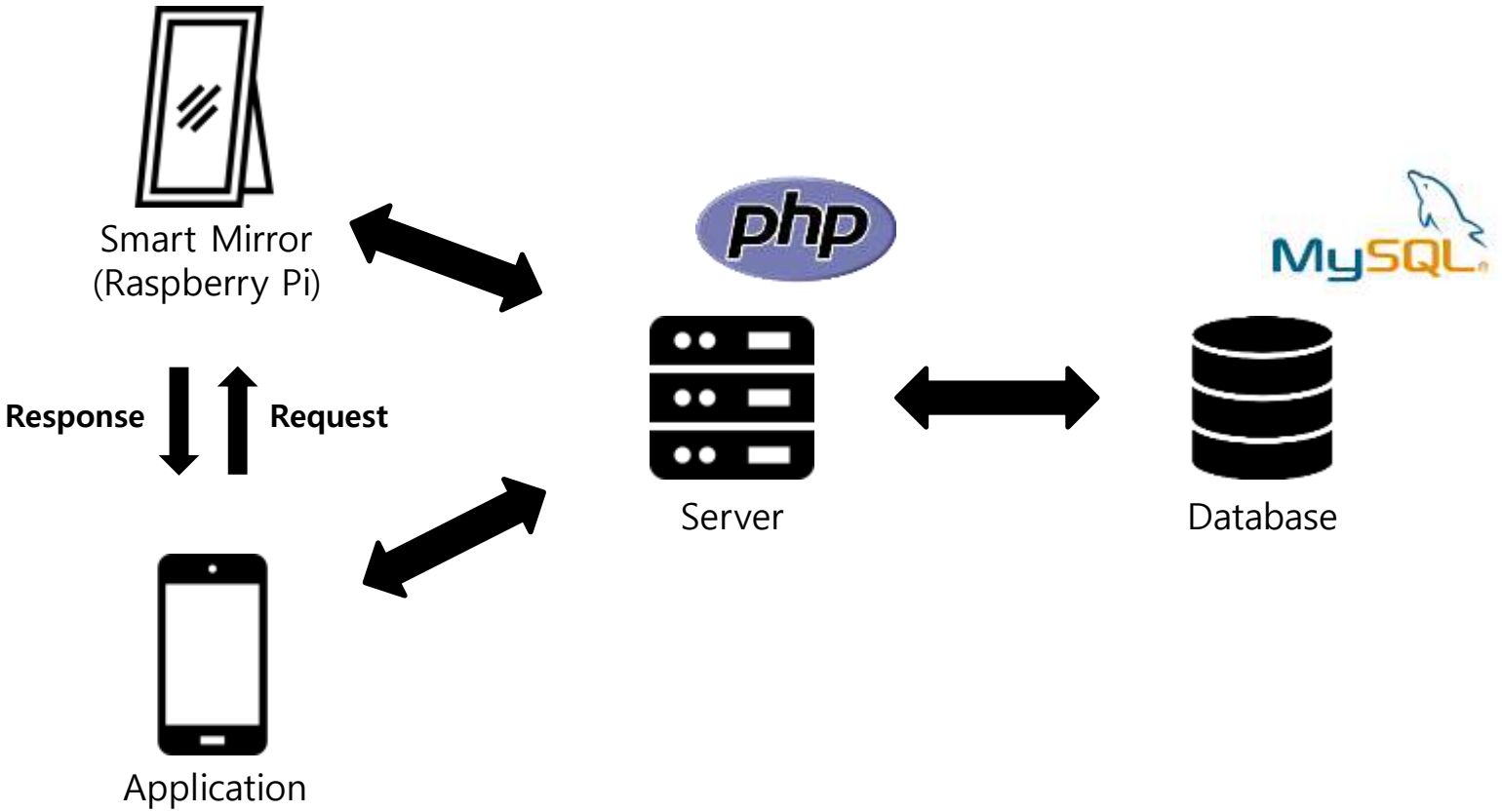
3. CLOSET Button

Recommendation clothes

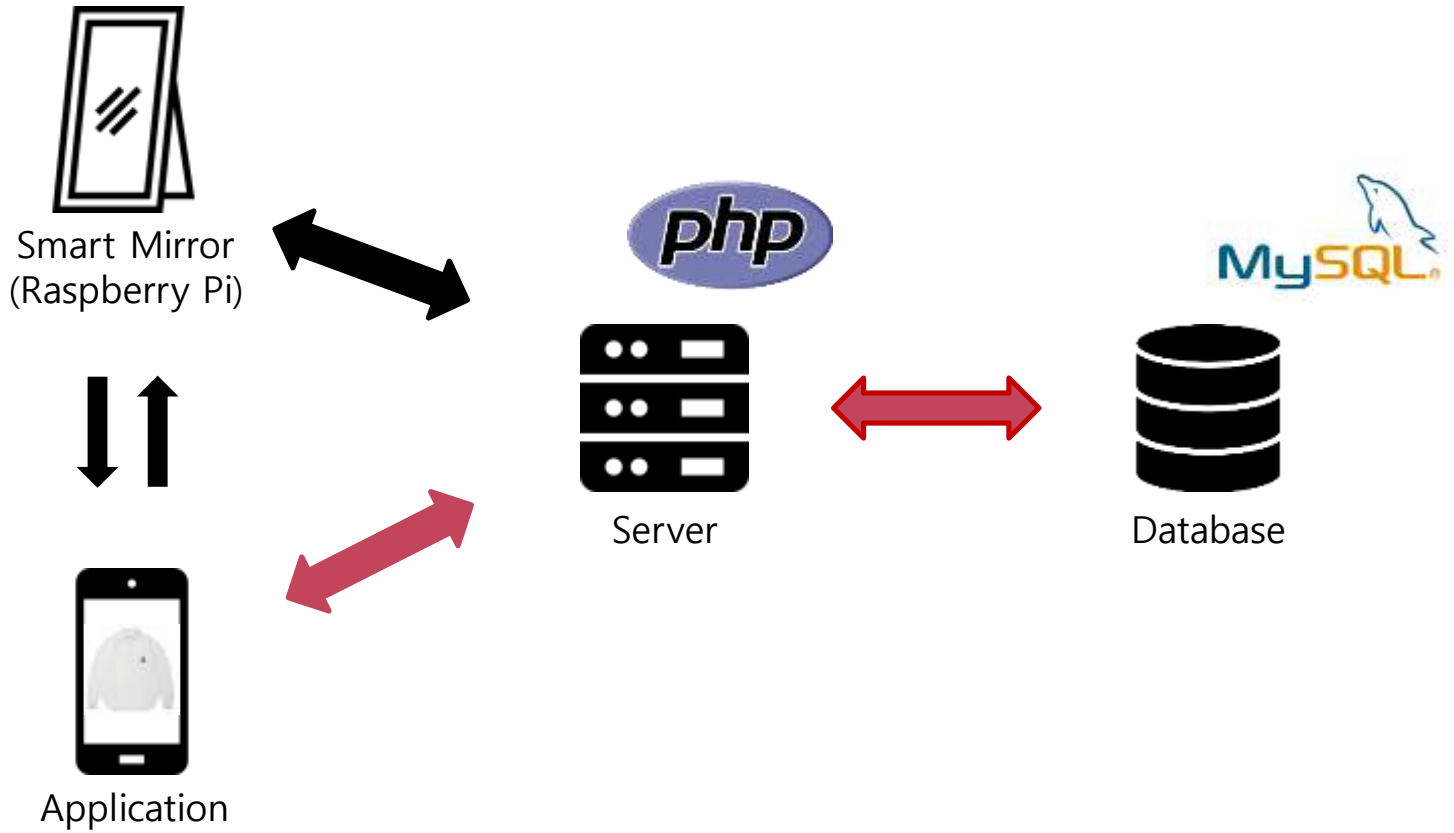
```
# make a combination to recommend styling with user's clothes
if temperature_section < 4:
    recommedation.top_bottom(top_temperature_section, bottom_temperature_section)
else:
    recommedation.outer_top_bottom(temperature_section, top_temperature_section, bottom_temperature_section)
```

```
# for top and bottom
def top_bottom(top_temperature_section,bottom_temperature_section):
    sql="SELECT  t.ID, b.ID, t.category,b.category, t.color, b.color, b.bottom_fit FROM U_top_info as t JOIN U_bottom_info as b WHERE t.temperature_
    section="+str(top_temperature_section)+" AND b.temperature_section="+str(bottom_temperature_section)+";"
    curs.execute(sql)
    result=pd.DataFrame(curs.fetchall())
    print(result)

    result_size=len(result) # 모든 경우의 수 갯수
    scores=[]
    for index,row in result.iterrows():
        topID=row[0]
        bottomID=row[1]
        topCate=row[2]
        bottomCate=row[3]
        topColor=row[4]
        bottomColor=row[5]
        bottomFit=row[6]
        score=0
        combi=[topID,bottomID]

        score=topCate_bottomCate.loc[bottomCate][topCate]+topColor_bottomColor.loc[bottomColor][topColor]+topCate_bottomFit.loc[bottomFit][topCate]
        scores.append([combi,score])
```

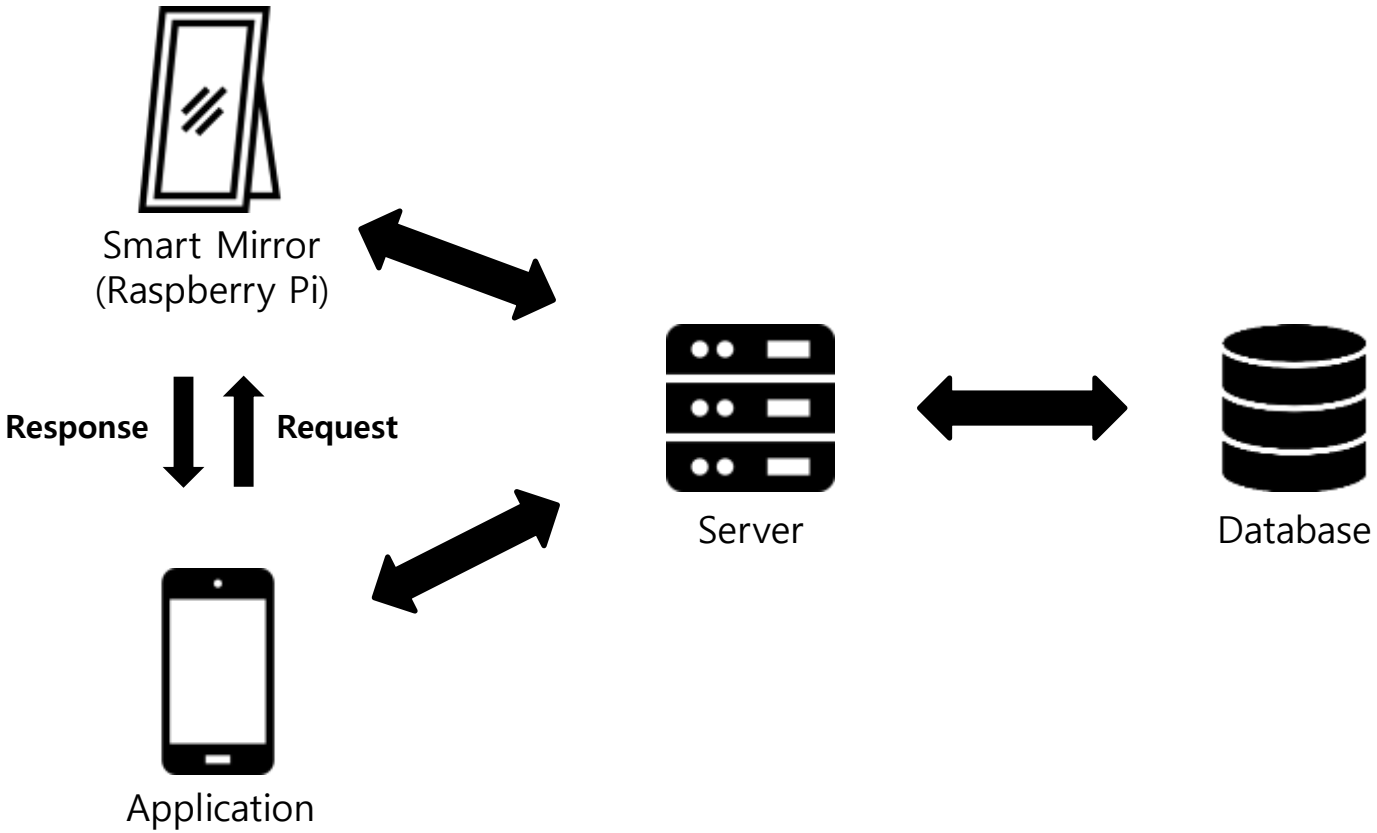
1. STYLING Button



1. STYLING Button



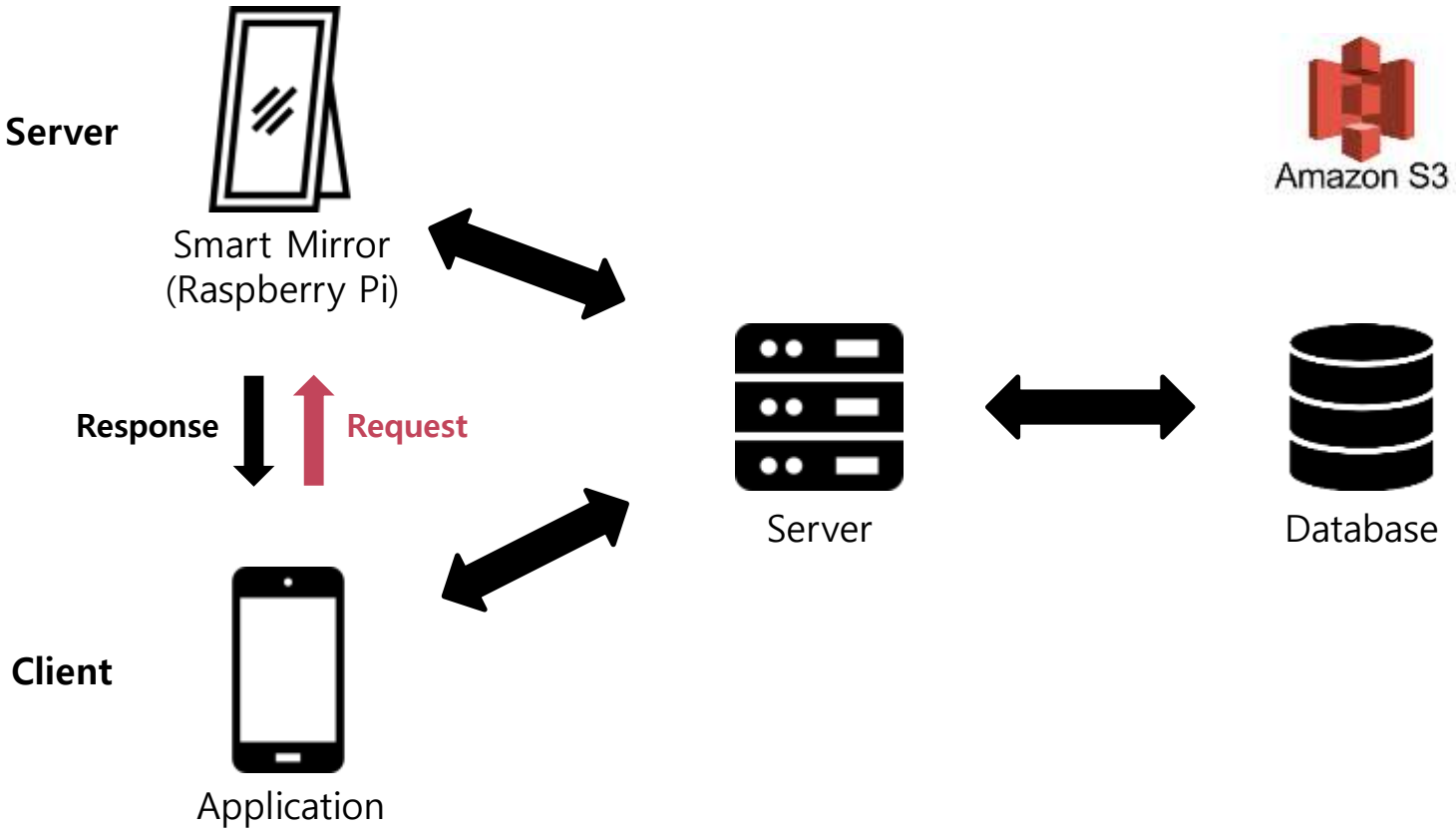
2. CAMERA Button



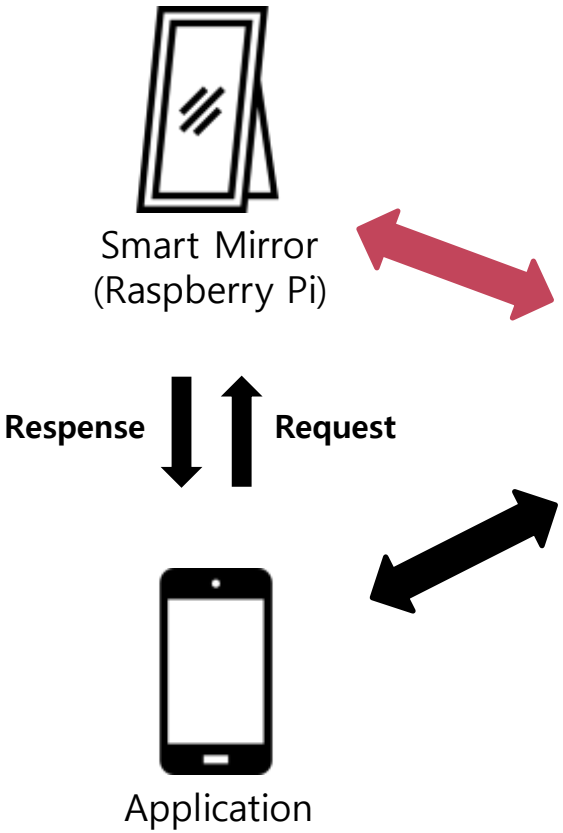
Implementation

2. CAMERA Button

이름	유형	마지막 수정
top_1.jpg	jpg	2021. 12. 13. am 11:07:44 AM KST
top_2.jpg	jpg	2021. 12. 13. am 11:07:42 AM KST
top_3.jpg	jpg	2021. 12. 13. am 11:07:43 AM KST
top_4.jpg	jpg	2021. 12. 13. am 11:07:34 AM KST
top_5.jpg	jpg	2021. 12. 13. am 11:07:44 AM KST



2. CAMERA Button



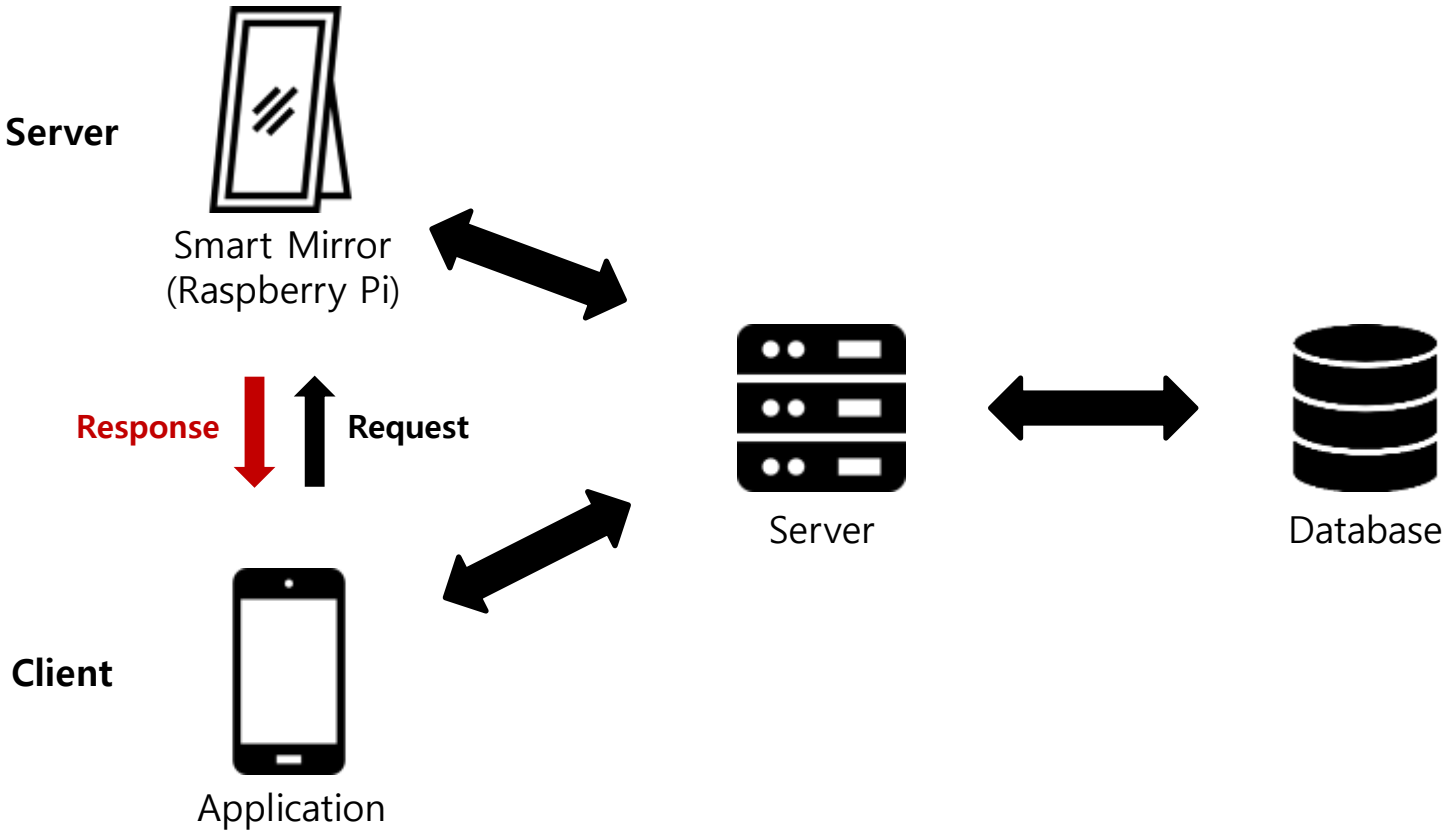
Information table

ID	category	sleevelength	color	print	temperature_section
1	맨투맨	긴팔	silver	스트라이프	3
2	맨투맨	긴팔	silver	스트라이프	3
3	민소매	민소매	white	스트라이프	1
4	민소매	민소매	silver	스트라이프	1
5	맨투맨	반팔	brown	스트라이프	3

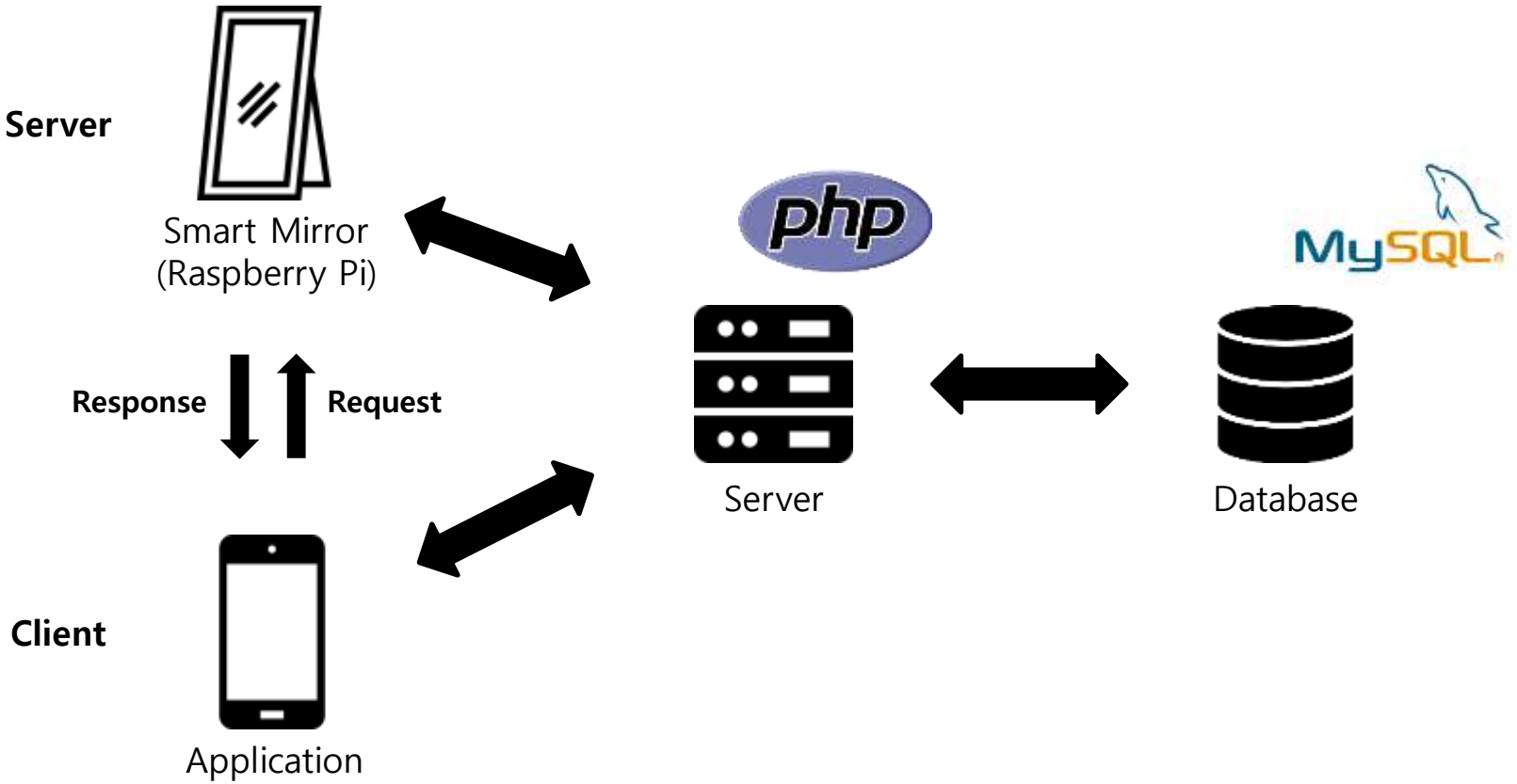
Image table

ID	image
1	https://smartmirror-bucket.s3.ap-northeast-2.a...
2	https://smartmirror-bucket.s3.ap-northeast-2.a...
3	https://smartmirror-bucket.s3.ap-northeast-2.a...
4	https://smartmirror-bucket.s3.ap-northeast-2.a...
5	https://smartmirror-bucket.s3.ap-northeast-2.a...

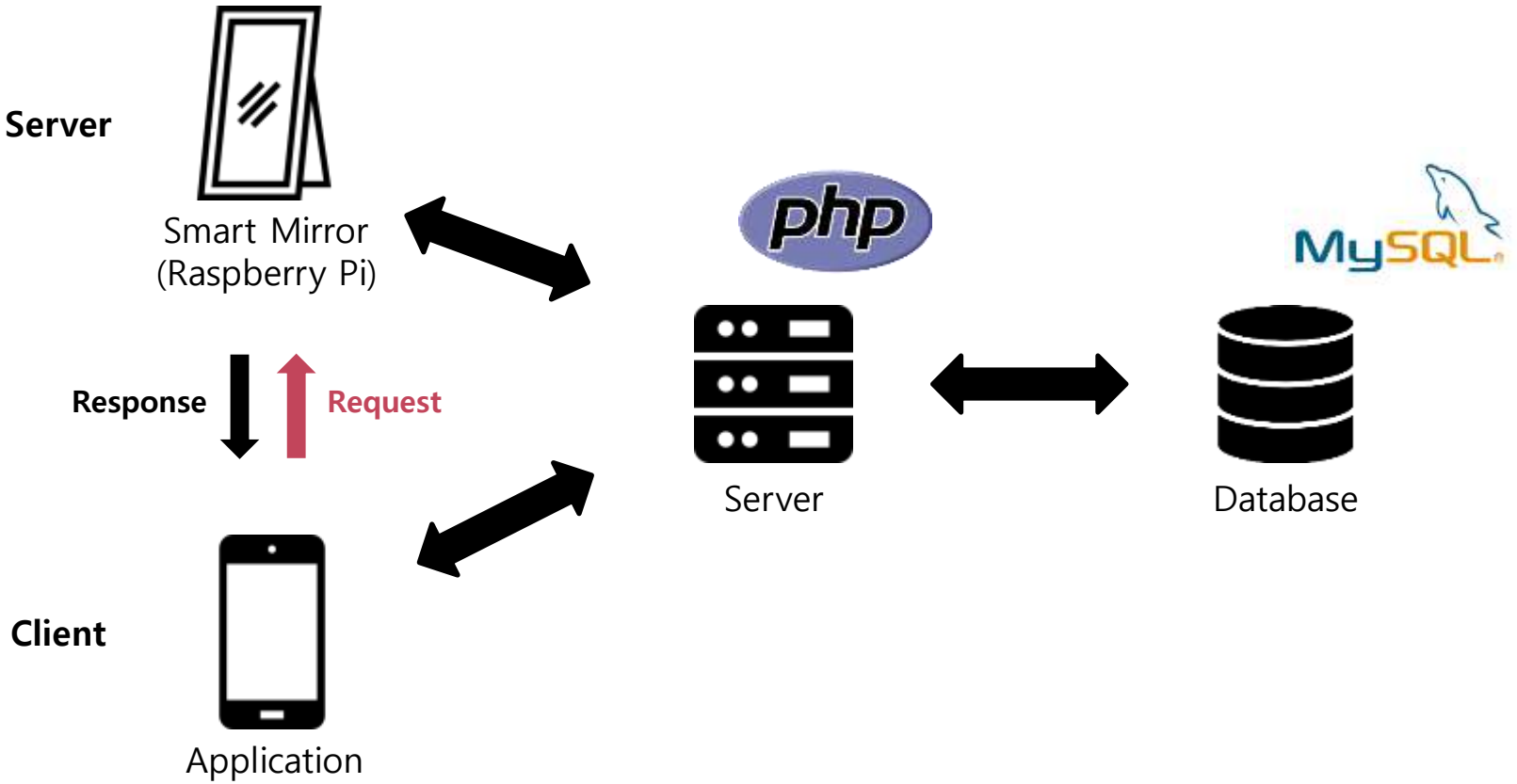
2. CAMERA Button



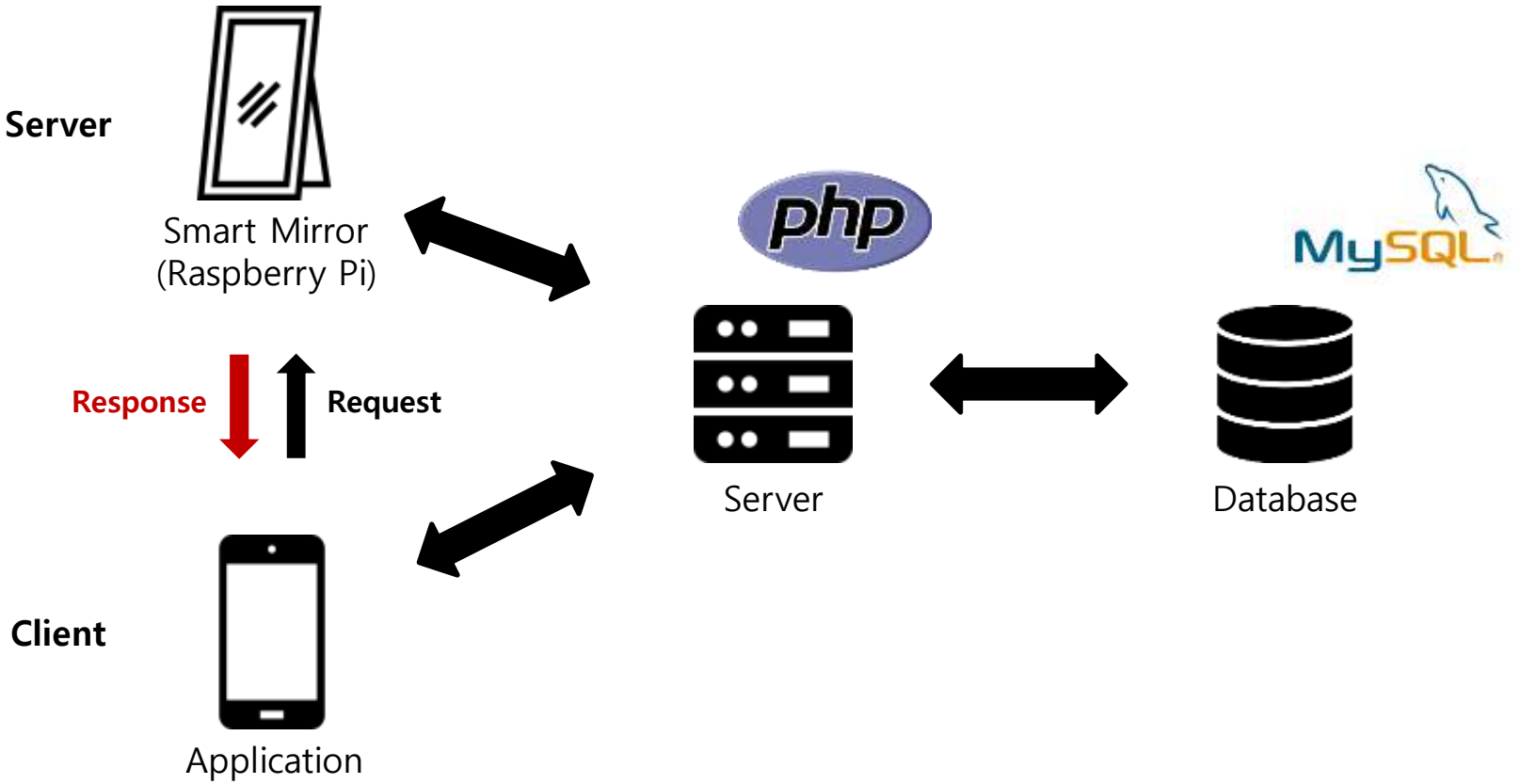
3. CLOSET Button



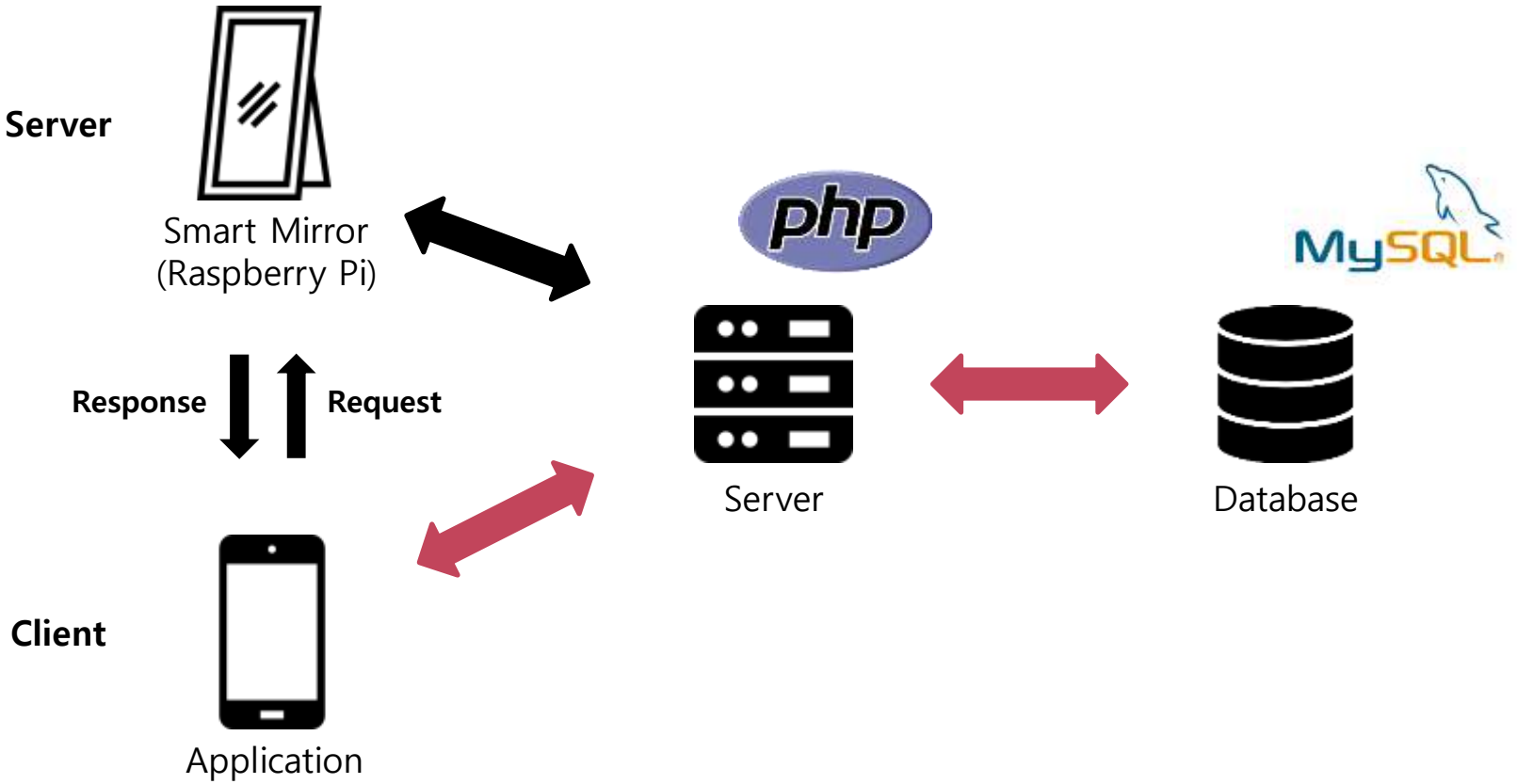
3. CLOSET Button



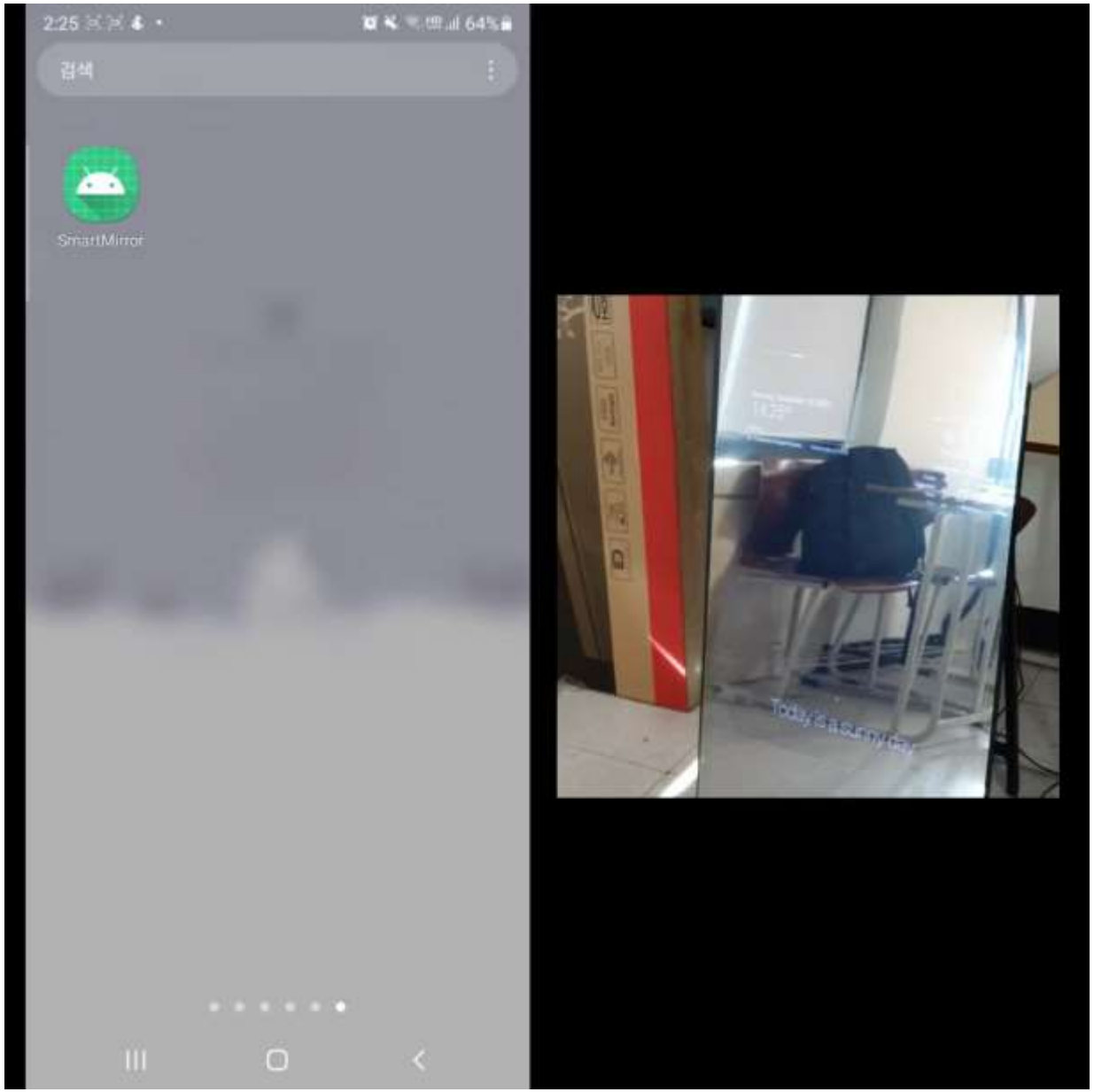
3. CLOSET Button



3. CLOSET Button



Project Demo



Future Plan

DECEMBER

- Modified application and add voice recognition

JANUARY

- Testing

FEBRUARY

- Wiki page & Documentation

Role

Choi Hyeongin

- OpenCV
- RasberryPi

Lee Minseo

Database
Appication

Lee Sojeong

UI
Crawling

Thank you

Github:

https://github.com/minseo300/GraduationProject