

실무중심산학협력프로젝트 최종 결과보고서

옷찾사 (옷을 찾아주는 사람들)

People who find clothes

2019년 12월 4일

32151671 박민혁

32153180 이상민

32170655 김소담

32171825 박지현

I . 과제 개요 ————— 3 page

- (1) 개발 과제 목표 및 변경 내용
- (2) 개발 과제 필요성
- (3) 개발 제한 조건
- (4) 개발 과제의 기대 효과

II . 과제 수행 내용 ————— 7 page

- (1) 개발 환경 및 요구조건
- (2) 시스템 블록 다이어그램
- (3) 알고리즘
- (4) 구현 방법
- (5) 실험 결과 및 성능 분석

III . 과제 수행 결과 내역 ————— 17 page

- (1) 추진계획 및 실적
- (2) 참여 인원별 역할 및 수행 소감
- (3) 과제 결과물
- (4) 활용 방안

IV . 후기 ————— 23 page

1. 과제 개요

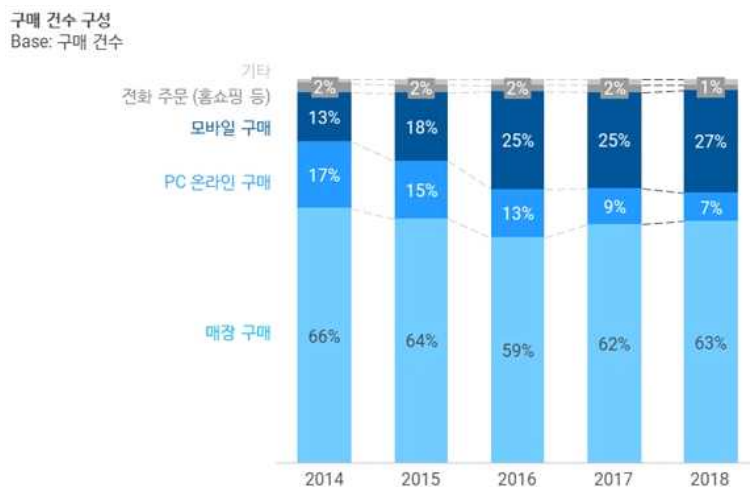
(1) 개발 과제 목표 및 변경 내용

- 변경 내용 없음

(2) 개발 과제 필요성

- Requirement Analysis

온라인 시장이 계속해서 성장하고 있는 반면 소비자들은 여전히 오프라인 매장을 적극 이용한다. 다음 통계자료를 보면 2014년부터 모바일 쇼핑이 꾸준히 성장하면서 매장, PC 및 온라인의 비율은 지속 하락했지만 결과적으로 오프라인의 비율은 다시 상승세를 보이고 있다.



출처 : <http://cosmornig.com/news/article.html?no=29966>

온라인 시장이 성장하고 있는 가장 큰 이유 중 하나는 편리성이다. 따라서 우리는 온라인 쇼핑의 장점인 편리성을 오프라인 쇼핑에 적용하고자 오프라인 쇼핑 이용 고객들을 위한 애플리케이션 개발이 필요하다고 생각하였다.

- 설문조사 결과

101명 응답

[응답 별 결과보기 >](#)

당신은 옷을 구매하러 갔는데 원하는 제품의 재고가 없는 상황에 직면했다. 이때 당신이 가장 취하고 싶은 행동은 ?



해당 제품이 있는 주변 매장을 찾아간다.	36	35.6%
해당 제품과 비슷한 제품을 추천 받는 다.	28	27.7%
해당 제품의 재고가 들어오면 연락 받는 다.	15	14.9%
해당 제품을 온라인으로 주문한 뒤 기다린다.	12	11.9%
해당 제품 구매를 포기한다.	6	5.9%
기타	4	4.0%

(3) 개발 제한조건

가. 공학적 및 경제적

해당 애플리케이션은 패션 상품 브랜드와의 협약이 필요하다. 하지만 기업에서는 애플리케이션의 완성도, 디자인 등을 고려해 계약을 체결해주기 때문에 마케팅 전략이 중요할 것이다. 현재 패션 상품 브랜드가 등록된 것이 ‘ ’개이다. 이 많은 양을 데이터베이스로 구축한다는 것은 상당한 시간이 소요된다. 또한 다른 브랜드가 입점할 때마다 계속적으로 상품들이 추가되기 때문에 그때마다 자동으로 입력해주는 기술도 필요하다고 생각한다.

나. 환경적

우리는 제한된 기기에서만 테스트를 진행할 수 있다. 애플리케이션을 새로 개발하거나, 기존에 사용 중인 애플리케이션을 배치할 때 테스트는 아주 중요한 부분이기 때문에 여러 기기에서 테스트하지 않을 경우 문제가 생길 수 있다. 하지만 현재 우리의 개발환경은 한정적이므로 여러 가지 문제점을 파악하기에는 다소 제한된다.

다. 사회적

오프라인 쇼핑의 매출액이 상승함과 동시에 온라인 쇼핑의 매출액과 이용자 수 또한 계속적으로 증가한다. 기존에 출시된 애플리케이션들은 대부분 온라인 쇼핑을 위한 것이기 때문에 사람들의 기대감을 충족시키는 것에 어려움이 있을 수 있다.

(4) 개발 과제의 기대효과

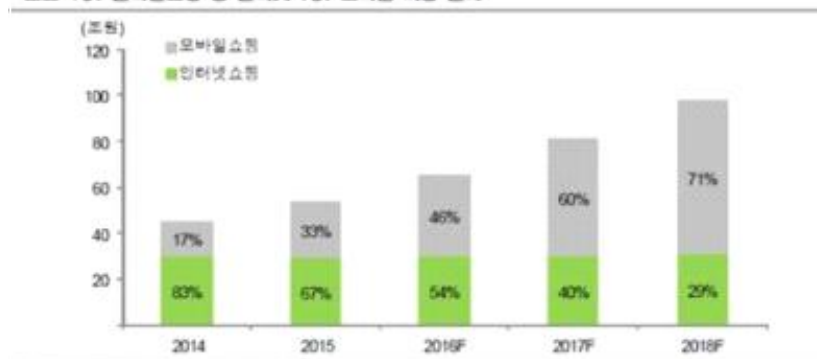
가. 기존 기술의 현황, 문제점 및 개선 방안

현재 국내 온라인 쇼핑 시장은 급속도로 성장하고 있다. 온라인 쇼핑이 이렇게 급격한 성장을 이룬 가장 큰 이유는 “편리해서”가 압도적인 1위를 차지했다. 하지만 온라인 쇼핑에 대한 소비자들의 불만은 여전히 존재한다. 대표적인 불만 사항은 다음과 같다.

- 제품 종류에 오차 발생
- 직접 착용해보지 못해 발생하는 불편함
- 총동구매를 하게 되는 쇼핑환경

위와 같은 불편함이 온라인 쇼핑 문화에 공존하기 때문에 오프라인 매장은 여전히 높은 매출액을 기록하고 있다. 소비자들이 선호하는 온라인 쇼핑 분야는 모바일이 압도적으로 높은 비중을 차지하고 있으며 현재에는 이러한 소비자들의 특성을 고려하여 모바일과 오프라인을 연계한 서비스 및 할인을 제공하여 영업 효과를 극대화하는 마케팅을 하는 추세이다.

도표 19. 온라인쇼핑 중 인터넷 vs. 모바일 비중 변화



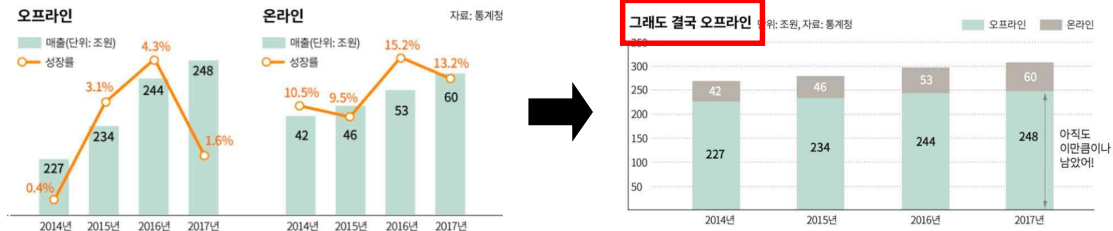
자료 : 통계청, 신영증권 리서치센터

출처 : <https://www.asiae.co.kr/article/2016112507034231641>

이러한 소비자들의 특성을 고려하여 ‘오프라인 매장 추천 모바일 앱’ 개발을 통해 온라인 쇼핑의 문제점을 보완하는 동시에 오프라인 시장을 활성화한다.

나. 과제 개발에 따른 기대효과

현재 소비자 시장에서의 오프라인 매출 성장률은 하락세처럼 보이지만, 사실상 아직까지는 온라인 시장보다 오프라인 시장이 훨씬 큰 매출을 기록하고 있다.



또한 2018년 7월, Monetate에서 23세 이하 자녀를 둔 소비자 조사 결과에 의하면, 신학기를 대비하기 위한 쇼핑으로 오프라인 매장 쇼핑이 절대적으로 선호되었음을 볼 수 있다.

- 56% 소비자가 오프라인 매장에서 쇼핑
- 20% 소비자는 온라인 쇼핑
- 7% 소비자만 소매점 앱으로 쇼핑

오프라인 매장이 여전히 선호되는 이유는 다음과 같다.

- 상품을 직접 만지고 느껴보고 싶기 때문, 70%
- 상품을 즉각 구입할 수 있기 때문, 66%
- 보다 많은 쿠폰과 할인을 받을 수 있기 때문, 65%
- 둘러보거나 쇼핑하는 즐거운 경험을 주기 때문, 41%
- 매장 직원과 대화하는 것이 즐거워서, 17%

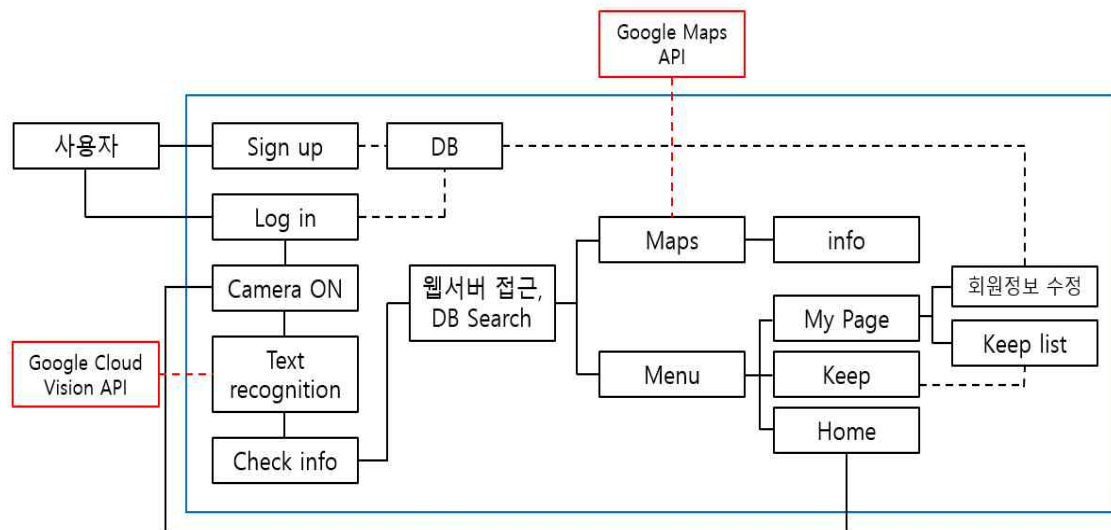
이러한 사실들을 통해 오프라인 시장의 꾸준한 발전을 예측할 수 있으며, 오프라인 매장 쇼핑을 즐기는 소비자들의 특성을 고려한 “옷찾사” 앱의 개발을 통해 오프라인 매장의 영업 효과를 배가시킬 수 있을 것으로 예측한다.

2. 과제 수행 내용

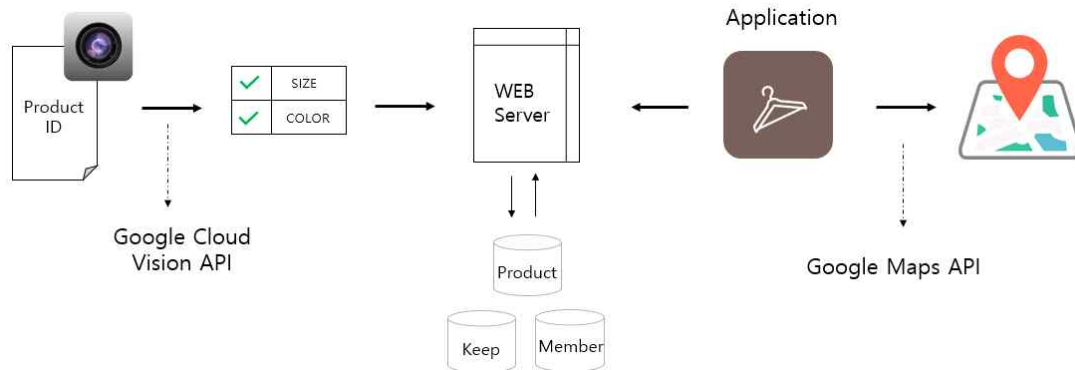
(1) 개발 환경 및 요구조건

- Android studio : Java language
- Linux Centos : 서버 및 워크 스테이션
- mariaDB : 데이터베이스 관리
- Apache : 웹서버로서 웹페이지 접근 요청을 관리
- PHP : 데이터베이스와 Apache 웹 서버를 연동

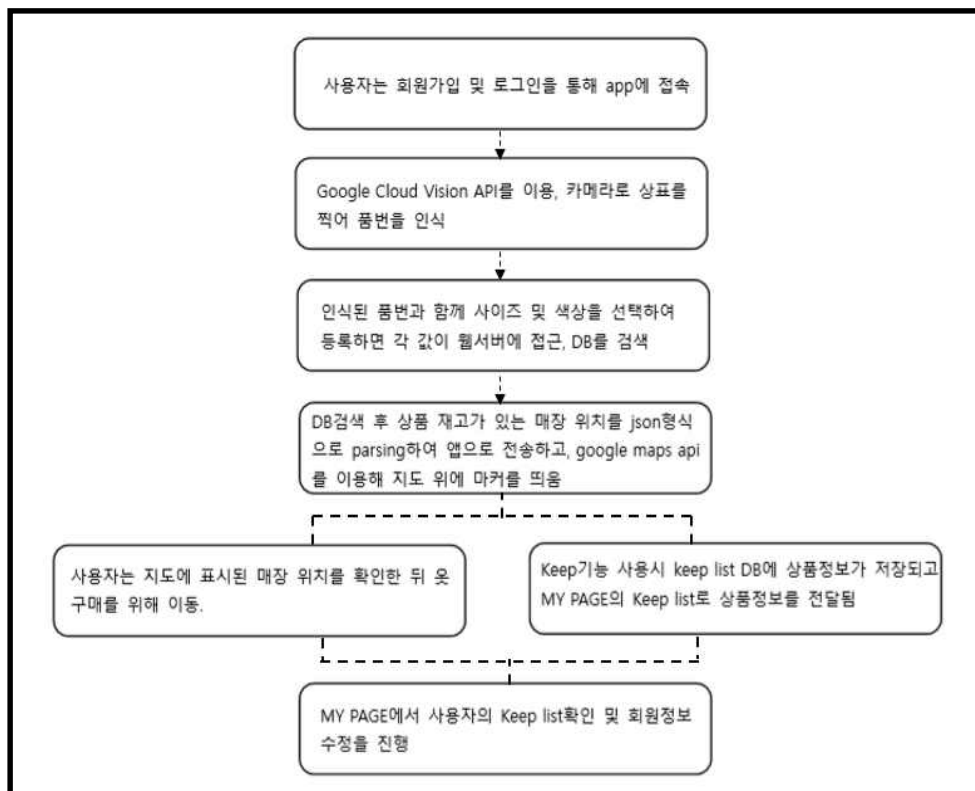
(2) 시스템 블록 다이어그램



(3) 알고리즘

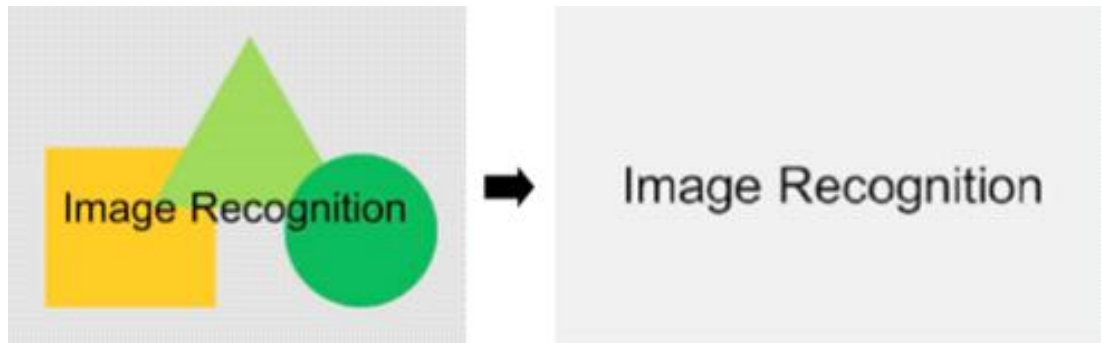


- ① 카메라로 찍은 품번은 [Google Cloud Vision API](#)를 통해 텍스트로 인식한다.
- ② Size와 Color 값을 받아 웹서버에 접근한 뒤 database를 검색한다.
- ③ 해당 상품의 재고가 있는 주변 매장의 위치데이터를 받고, json형식의 문자열을 parsing하여 애플리케이션으로 전송한다.
- ④ 위치 정보를 나타내기 위해 [Google maps API](#)를 사용하였고, database에 각 매장의 위치를 위도, 경도로 저장한다.
- ⑤ 앱 상에서 사용자가 입력한 데이터를 통해 주변 매장의 위치를 맵 위에 띄운다.



- Google Cloud Vision API

텍스트 인식을 위해 사용한 Google Cloud Vision API는 OCR 방식을 사용한다. OCR은 Optical Character Recognition의 약자로 optical, 즉 빛이나 광학을 이용해 문자를 인식하는 것이다. 사진에서 문자 부분을 인식하기 위해서는 우선적으로 아래 사진과 같이 배경과 문자를 구분해야 한다.



이렇게 나뉜 문자 부분을 파싱하여 다음과 같은 사각형 형태로 재구성한다.

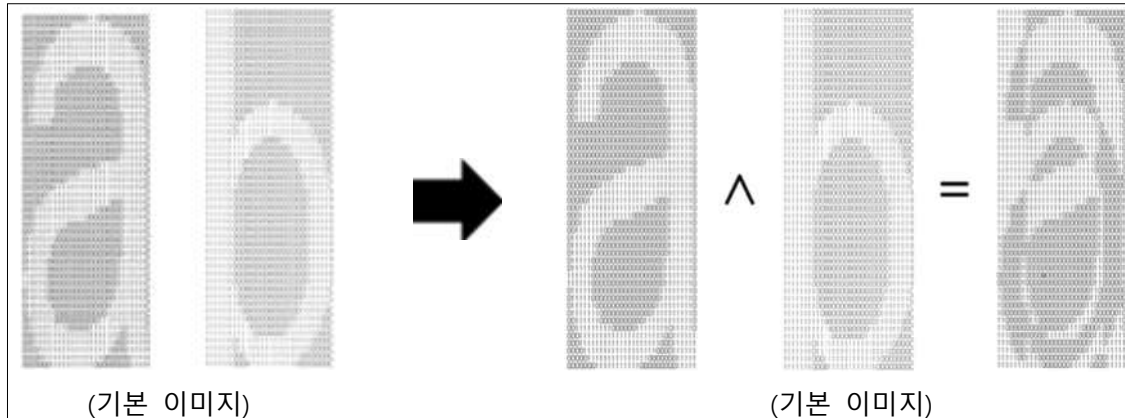


각 문자의 바탕은 0 bit, 텍스트는 1 bit로 표현된다. 우측에 있는 사진에서는 가로 32개, 세로 48개의 bit로 구성되어 있다. 이것은 정수형 데이터 타입으로 생각하고 32bit 크기에 맞춰기 때문이다. 만약 문자형 데이터 타입 크기에 맞추고 싶다면 8 by 12 크기도 가능하다.

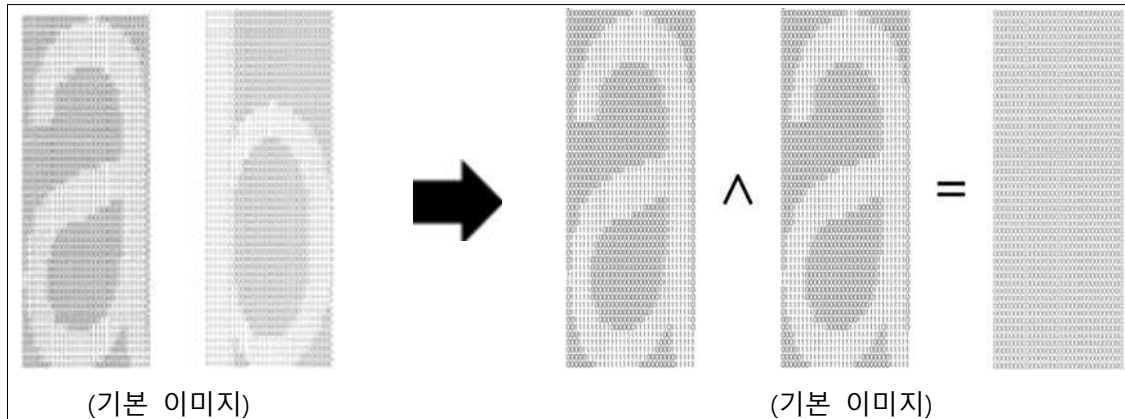


이렇게 bit 표현으로 바꿔준 후 각 문자가 어떤 문자인지를 인식하는 과정은 확률적으로 추측하게 된다. 문자를 인식하기 위해 비교할 대상인 기본 이미지가 있고, 이렇게 미리 학습된 여러 개의 기본 이미지들과 인식할 이미지를 차례로 비교하여 가장 근접한 알파벳을 결과로 도출하게 된다. 이 때 연산은 exclusive or 방식을 사용하여 기본 이미지와 일치할 때, 즉 0 bit가 상대적으로 0 bit가 많이 나오는 알파벳이 결과값이 된다.

Case 1



Case 2



예를 들어, ‘a’라는 문자를 판단하기 위해 기본 이미지들과 순차적으로 비교를 하게 되는데, case1에서는 ‘b’를 나타내는 기본 이미지와 xor 연산을 진행하여 1bit가 상대적으로 많이 표시된다. 하지만 Case2를 살펴보면 ‘a’를 나타내는 기본 이미지와 xor 연산을 진행하여 모든 bit가 0bit로 표시된다. 실제로 위와 같이 모든 bit가 0bit가 되는 경우는 드물기 때문에 상대적으로 0bit를 많이 표시한 기본 이미지를 결과로 도출하게 된다.

(4) 구현 방법

① 텍스트 인식

- Google Cloud Vision API를 사용해 텍스트 인식을 진행하였다. Vision API는 이미지에서 텍스트를 감지하고 추출할 수 있다. 우리는 모바일 카메라로 사진을 찍고 저장되는 형식이 bitmap 형식이기 때문에 이러한 파일을 텍스트로 인식하는 알고리즘을 작성하였다.

인식에 사용할 이미지 선택

```
if (requestCode ==
CropImage.CROP_IMAGE_ACTIVITY_REQUEST_CODE) {
    CropImage.ActivityResult result = CropImage.getActivityResult
(data);
    if (resultCode == RESULT_OK) {
        Uri resultUri = result.getUri(); // get image uri
        // set image to image view
        mPreviewlv.setImageURI(resultUri);
    }
}
```

Bitmap 형식의 이미지 텍스트로 변환

```
BitmapDrawable bitmapDrawable = (BitmapDrawable)
mPreviewlv.getDrawable();
Bitmap bitmap = bitmapDrawable.getBitmap();

TextRecognizer recognizer = new TextRecognizer.Builder
(getApplicationContext()).build();

if (!recognizer.isOperational()) {
    Toast.makeText(this, "Error", Toast.LENGTH_SHORT).show();
}
else {
    Frame frame = new Frame.Builder().setBitmap
(bitmap).build();
    SparseArray<TextBlock> items = recognizer.detect(frame);
    StringBuilder sb = new StringBuilder();
    // get text from sb until there is no text
    for (int i = 0; i < items.size(); i++) {
        TextBlock myItem = items.valueAt(i);
        sb.append(myItem.getValue());
        //sb.append("\n");
    }
}
```

인식한 텍스트 결과 출력 / 에러 처리

```

        mResultEt.setText(sb.toString());
        input_product_id = sb.toString();
        product = mResultEt.getText().toString();
    }
}

else if (resultCode ==
CropImage.CROP_IMAGE_ACTIVITY_RESULT_ERROR_CODE) {
    // if there is any error show it
    Exception error = result.getError();
    Toast.makeText(this, ""+error, Toast.LENGTH_SHORT).show();
}

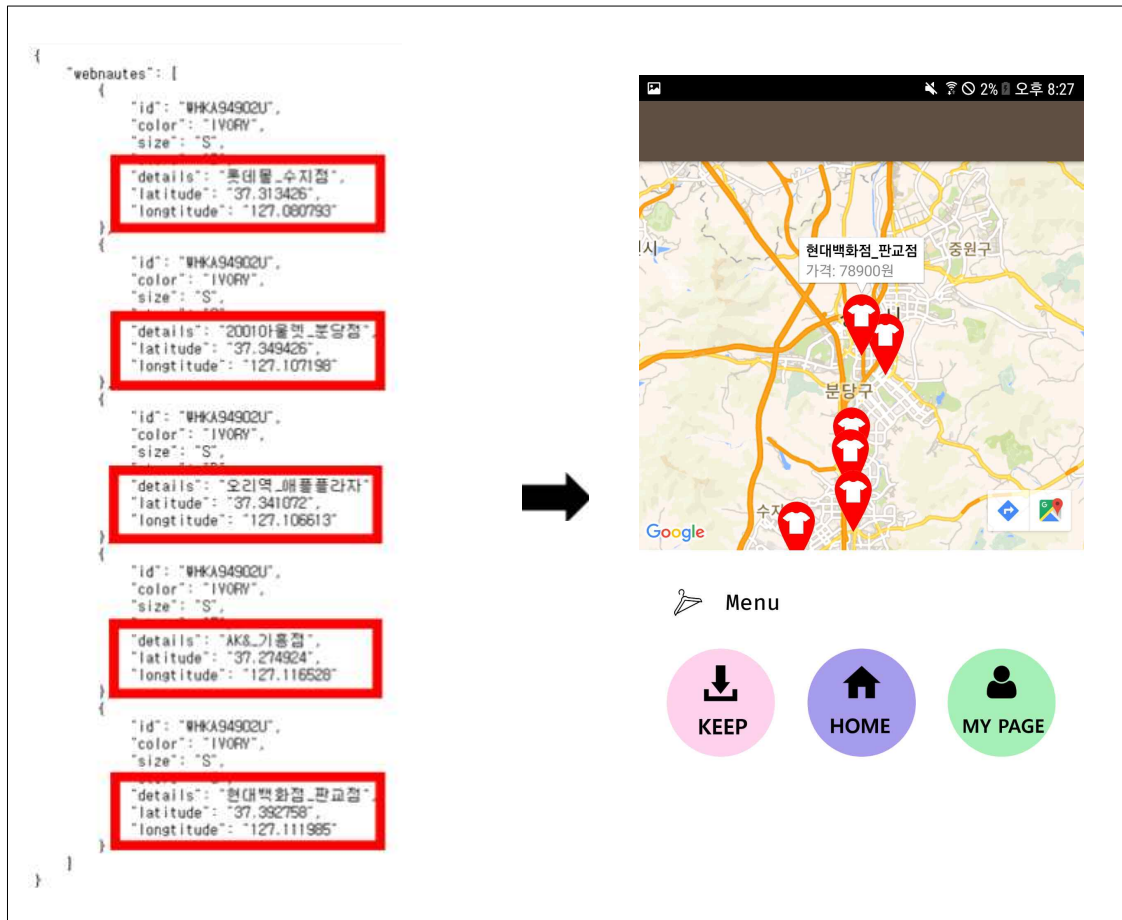
```

② Database

- 데이터베이스 개발 환경으로는 PHP, Apache, MariaDB를 이용해 Linux로 웹서버를 구축하였다. 웹서버에 저장 되어있는 데이터베이스는 JSON형식의 문자열을 parsing하여 어플리케이션으로 전송한다. 또한 Database에 각 매장의 위치를 위도와 경도로 저장하고 Database에 있는 정보를 Map에 표현하기 위해 Google Maps API를 사용하였다.

Database changed
MariaDB [product]> select * from information;

id	brand	color	size	store	details	price	latitude	longitude
CM1409	ADIDAS	BLACK	S	A	신세계백화점 경기점	79000	37.325375	127.108075
CM1407	ADIDAS	BLACK	S	A	신세계백화점 경기점	89000	37.325375	127.108075
CM1407	ADIDAS	BLACK	M	A	신세계백화점 경기점	89000	37.325375	127.108075
CM1407	ADIDAS	BLACK	L	A	신세계백화점 경기점	89000	37.325375	127.108075
CM1407	ADIDAS	BEIGE	S	A	신세계백화점 경기점	89000	37.325375	127.108075
CM1407	ADIDAS	BEIGE	M	A	신세계백화점 경기점	89000	37.325375	127.108075
CM1407	ADIDAS	BEIGE	L	A	신세계백화점 경기점	89000	37.325375	127.108075
WHTM94T11F	WHOAU	BLACK	S	A	신세계백화점 경기점	29900	37.325375	127.108075
WHTM94T11F	WHOAU	BLACK	M	A	신세계백화점 경기점	29900	37.325375	127.108075
WHTM94T11F	WHOAU	BLACK	L	A	신세계백화점 경기점	29900	37.325375	127.108075
228951YE15	8SECONDS	BLACK	S	A	신세계백화점 경기점	98900	37.325375	127.108075
228951YE15	8SECONDS	BLACK	M	A	신세계백화점 경기점	98900	37.325375	127.108075
228951YE15	8SECONDS	BLACK	L	A	신세계백화점 경기점	98900	37.325375	127.108075
228951YE15	8SECONDS	IVORY	S	A	신세계백화점 경기점	98900	37.325375	127.108075
228951YE15	8SECONDS	IVORY	L	A	신세계백화점 경기점	98900	37.325375	127.108075
228951YE15	8SECONDS	GREY	S	A	신세계백화점 경기점	98900	37.325375	127.108075
228951YE15	8SECONDS	GREY	M	A	신세계백화점 경기점	98900	37.325375	127.108075
228951YE15	8SECONDS	GREY	L	A	신세계백화점 경기점	98900	37.325375	127.108075
228951YE15	8SECONDS	NAVY	M	A	신세계백화점 경기점	98900	37.325375	127.108075
228951YE15	8SECONDS	NAVY	L	A	신세계백화점 경기점	98900	37.325375	127.108075
228951YE15	8SECONDS	BEIGE	S	A	신세계백화점 경기점	98900	37.325375	127.108075
228951YE15	8SECONDS	BEIGE	M	A	신세계백화점 경기점	98900	37.325375	127.108075
228951YE15	8SECONDS	BEIGE	L	A	신세계백화점 경기점	98900	37.325375	127.108075
C1804SL01WH	COVERNAT	WHITE	S	A	신세계백화점 경기점	39000	37.325375	127.108075
C1804SL01WH	COVERNAT	WHITE	M	A	신세계백화점 경기점	39000	37.325375	127.108075
C1804SL01WH	COVERNAT	YELLOW	S	A	신세계백화점 경기점	39000	37.325375	127.108075
C1804SL01WH	COVERNAT	YELLOW	M	A	신세계백화점 경기점	39000	37.325375	127.108075
C1804SL01WH	COVERNAT	NAVY	S	A	신세계백화점 경기점	39000	37.325375	127.108075
C1804SL01WH	COVERNAT	NAVY	M	A	신세계백화점 경기점	39000	37.325375	127.108075
C1804SL01WH	COVERNAT	NAVY	L	A	신세계백화점 경기점	39000	37.325375	127.108075
C1804SL01WH	COVERNAT	BLACK	S	A	신세계백화점 경기점	39000	37.325375	127.108075
C1804SL01WH	COVERNAT	BLACK	L	A	신세계백화점 경기점	39000	37.325375	127.108075



(5) 실험 결과 및 성능 분석

평가항목	평가방법	결과 및 분석
상품 인식의 정확성	텍스트를 정확하게 인식하는가	텍스트 - 높은 인식을 손글씨 - 낮은 인식을
위치 표시의 정확성	근처 매장을 제대로 찾아내는가	단국대 근처 매장 DB로의 접근, 지도 위에 정확한 매장 위치 표시 성공.
데이터 정보의 정확성	상품 정보가 제대로 저장되어있는가	mypage에 keep 상품 정보가 정확히 저장됨을 확인.

① 상품 인식의 정확성

카메라로 찍은 이미지에서 문자를 얼마나 정확하게 인식하는가를 알아보기 위해 실제 상표사진 10개, 직접 작성한 문자를 찍은 이미지 10개로 실험을 진행하였다. 결과적으로 직접 작성한 문자를 찍은 사진에서의 문자인식은 20%의 정확도 밖에 나오지 않았고, 실제 상표사진 10개에서의 문자인식은 100% 정확도로 10장의 이미지에서 완벽히 문자인식을 해낸 것을 확인할 수 있었다.

- 직접 작성한 문자에 대한 문자인식 (정확도 : 20%)

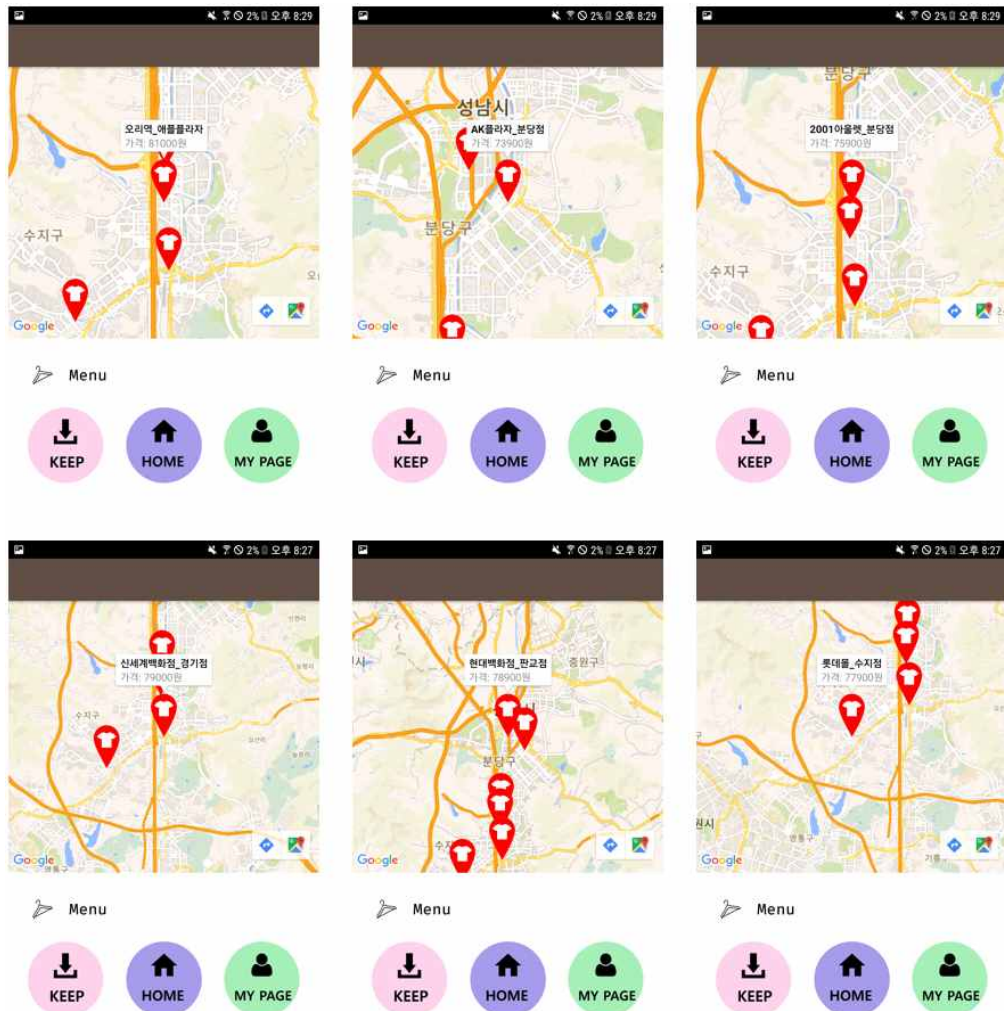
<div>Product Name</div> <div>Nike177</div> <div>Image Recognition</div> <div>Nike177</div>	<div>Product Name</div> <div>LAP785</div> <div>Image Recognition</div> <div>LAP7895</div>	<div>Product Name</div> <div>Addidas 123</div> <div>Image Recognition</div> <div>Addidas123</div>
<div>Product Name</div> <div>Micro Sot AL 156</div> <div>Image Recognition</div> <div>Microsoft567</div>	<div>Product Name</div> <div>FIGMENT 918</div> <div>Image Recognition</div> <div>FIGMENT1998</div>	<div>Product Name</div> <div>CM140</div> <div>Image Recognition</div> <div>CM1409</div>
<div>Product Name</div> <div>H capstone</div> <div>Image Recognition</div> <div>Hi capstone</div>	<div>Product Name</div> <div>ENGLISH 665</div> <div>Image Recognition</div> <div>ENGLISH665</div>	<div>Product Name</div> <div>performance check</div> <div>Image Recognition</div> <div>performance check</div>
<div>Product Name</div> <div>WHFA 949 020</div> <div>Image Recognition</div> <div>WHFA949020</div>		

- 실제 상표 이미지에서의 문자인식 (정확도 : 100%)



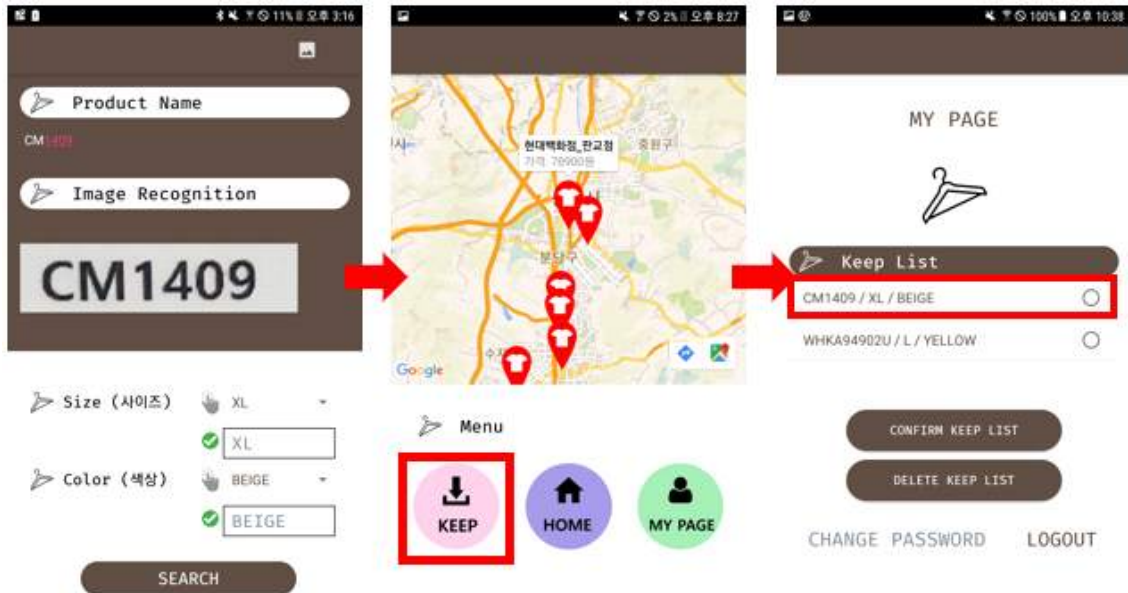
② 위치 표시의 정확성

단국대학교 위치에서 상품을 검색했을 때, 미리 데이터베이스에 입력해놓은 단국대 근처 매장 7개의 위도, 경도 정보를 정확하게 접근하여 7개의 매장 중 상품 재고를 확보한 매장에 대해 지도 위에 마커를 띄우는 것을 확인하였다.



③ 데이터 정보의 정확성

상품을 검색한 뒤 keep 기능을 이용하면 my page에서 해당 상품과 일치하는 상품 정보가 저장되는 것을 확인하였다.



3. 과제 수행 결과 내역

(1) 추진계획 및 실적

- 초기 마일스톤

업무내용	담당자	9				10					11				12	
		1	2	3	4	1	2	3	4	5	1	2	3	4	1	2
아이디어 구상 팀 역할 분배	공통															
자료 수집	공통															
시스템 설계	공통															
기능별 코딩	공통															
테스트	공통															
중간보고서	공통															
수정 및 보완	공통															
최종제작	공통															
최종보고서	공통															

- 세부 계획 및 실적

세부 목표	세부 계획	실 적	달성도(%)	참여자
상품의 태그를 카메라로 찍어 해당 사진에서 품명을 인식한다.	Google Cloud Vision API를 이용하여 이미지에서 문자를 인식한다.	상품 태그를 찍은 이미지에서 품명을 정확하게 인식하는 것을 확인 하였다.	100%	이상민 박지현
인식한 상품의 재고가 있는 주변 매장을 검색하여 지도 위에 마커로 표시한다.	Google Maps API와 DB검색을 통하여 주변 매장의 위치를 지도 위에 표시한다.	현재 위치를 인식하고 재고가 있는 근처 매장의 위치를 정확하게 지도 위에 띄우는 것을 확인 하였다.	100%	박민혁 김소담

(2) 참여 인원별 역할 및 수행 소감

- 역할

역할 구분	성 명	수행 역할
데이터베이스	박민혁	데이터베이스 구축 , DB & 어플 연동
애플리케이션	박지현	텍스트 인식 , 애플리케이션 UI 개발
애플리케이션	이상민	텍스트 인식 , 애플리케이션 UI 개발
데이터베이스	김소담	데이터베이스 구축 , DB & 어플 연동

- 수행 소감

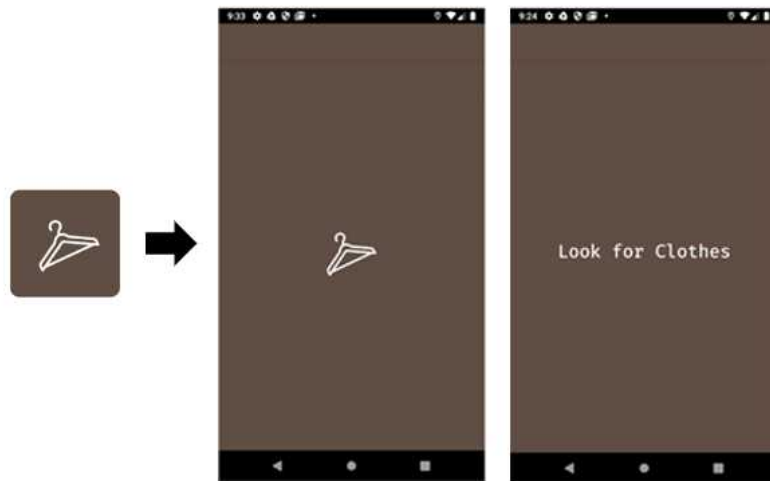
박지현 : 안드로이드 스튜디오 환경에서 텍스트 인식 및 애플리케이션 UI를 개발하는 과정에서 수도 없이 많은 오류들과 직면해야했고, 이 오류들을 해결하기 위해 여러 번의 시행착오를 겪었습니다. 초반에는 계획대로 쉽사리 진행이 되지 않아 작업이 순조롭지만은 않았지만, 팀원들과 합심하여 서로의 부족한 부분을 보완해주면서 작업한 덕분에 좋은 결과를 낼 수 있었습니다.

이상민 : 팀원들 모두 앱 개발은 처음이기 때문에 초반에는 진행 속도가 다소 부진했습니다. 하지만 각자의 분야에서 스스로 공부하고 서로 도와주며 프로젝트를 잘 끝마칠 수 있었습니다. 단순히 작품의 완성만을 추구하는 것이 아니라 교수님께서 말씀하신 것처럼 어떠한 알고리즘이 사용되었고, 어떻게 동작하는지 공부해가며 팀원 모두의 실력이 성장하는 좋은 경험이었습니다.

박민혁 : 안드로이드 스튜디오는 처음 접하고 데이터베이스는 배웠지만 잘 모르는 분야였기에 초반에 역할 정할 때 부담감이 많았습니다. 두 가지 모두 자신이 없었기 때문입니다. 하지만 팀원들이 열심히 하는 모습을 보고 덩달아 열심히 공부를 하게 되었던 것 같습니다. 이번 프로젝트를 하면서 우리 과에서 혼자서 할 수 있는 것이 많이 없다고 느꼈습니다. 팀이 얼마나 중요한지 자신이 맡은 역할에 대한 책임감이 얼마나 중요한지 알게 되는 경험이었습니다.

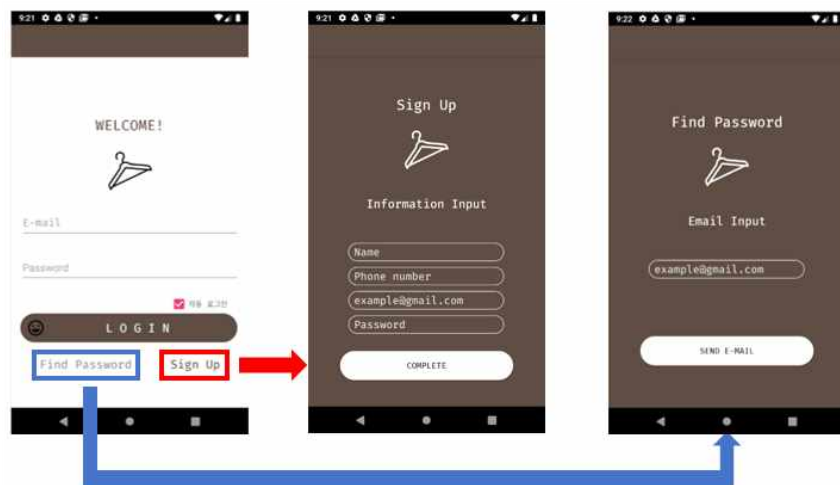
김소담 : 팀원들과 함께 프로젝트를 수행하면서 각자 맡은 역할을 해내기 위해 모두 노력하였습니다. 이번 기회에 다양한 개발 환경을 새롭게 접하고 공부할 수 있었고, 오픈소스를 분석, 활용해보면서 많이 배울 수 있었습니다. 또한 google API를 직접 어플리케이션에 적용해보면서 API 알고리즘을 보다 정확하게 이해하고자 했던 의미 있는 프로젝트였습니다.

(3) 과제 결과물



<Intro>

앱 실행 시 첫 화면으로 실행되는 인트로 페이지이다.

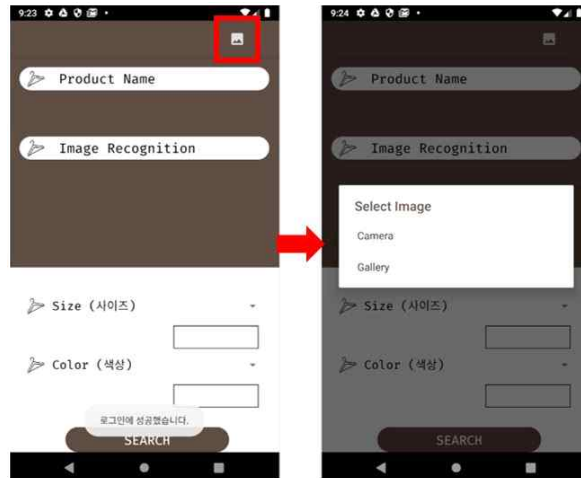


<Login>

“Sign up” 페이지에서 4가지 정보를 입력받아 회원가입을 진행한다.

“Find password” 페이지에서는 잊어버린 비밀번호를 찾아주며, 이메일을 입력하면 사용자의 비밀번호가 해당 이메일로 전송된다.

회원가입을 완료한 뒤에는 가입한 e-mail, password로 로그인 가능하며, 자동 로그인을 체크할 경우 이후에 다시 앱에 접속할 때 자동으로 로그인 된다.



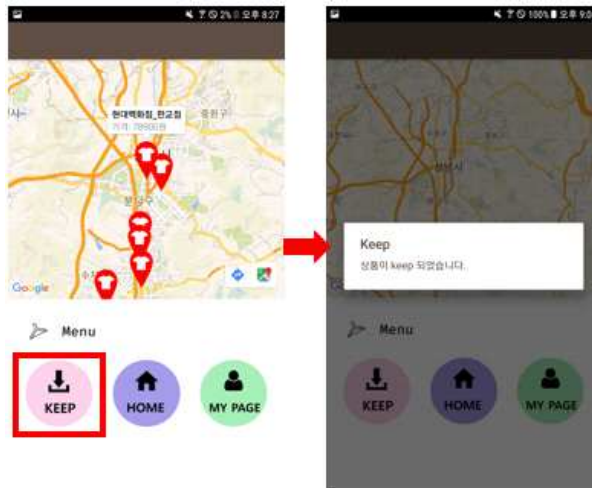
<Camera>

로그인에 성공하면 상품을 인식할 페이지가 나타나며 우측 상단에 위치한 카메라 버튼을 클릭하여 카메라를 구동하도록 한다.



<Text recognition>

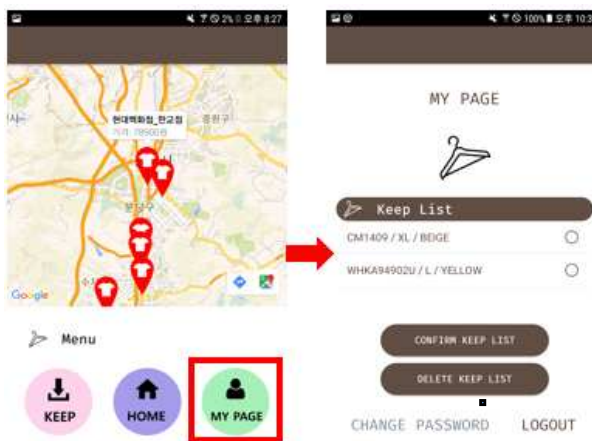
카메라로 상표를 찍어 상품명에 있는 영역만 선택하여 인식 시킨다.
앱이 상품명에 인식하여 텍스트로 보인다.
하단의 사이즈, 색상을 사용자가 직접 선택한 뒤 search한다.



<Map>

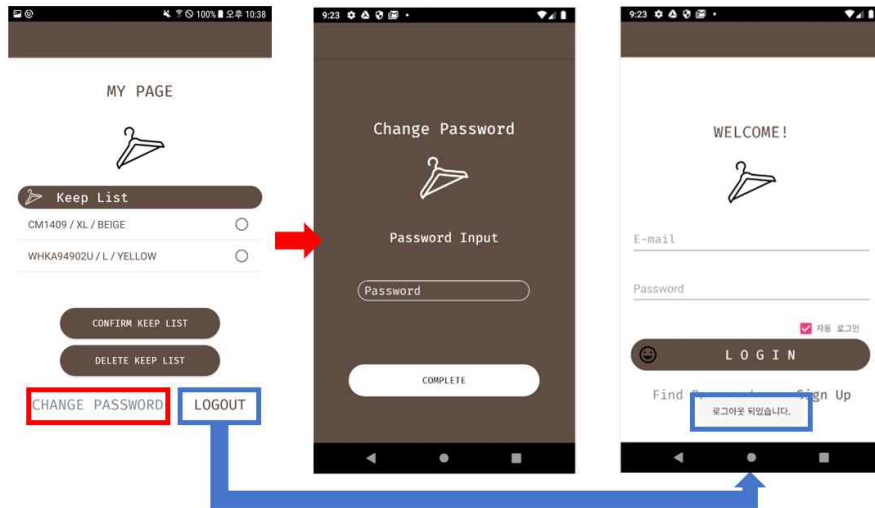
search가 진행되면 앱은 인식한 상품명으로 검색하여 재고가 있는 주변 매장을 지도에 띄워 준다.

이렇게 추천된 매장들이 사용자의 예상보다 멀 경우에는 keep 버튼을 눌러 상품을 keep list에 저장한다.



<Keep>

My page에 들어가면 사용자의 keep list에 저장된 상품정보들을 확인할 수 있다.



<My page>

“change password” 페이지에서 사용자 비밀번호를 변경할 수 있도록 한다.

“logout”을 누르면 로그아웃 되도록 한다.

(4) 활용 방안

‘옷찾사’ 애플리케이션은 오프라인 매장 쇼핑을 이용하는 소비자를 대상으로 한 오프라인 쇼핑 앱으로, 온라인 쇼핑의 단점(제품 종류에 오차 발생, 직접 착용해보지 못하는 불편함,冲动구매를 하게 되는 쇼핑환경 등)을 개선하며 온라인 쇼핑의 편리성까지 제공하도록 구상되었다.

이 앱은 오프라인 쇼핑을 즐기는 고객들을 대상으로 서비스가 제공되며, 앱의 사용자들은 오프라인 쇼핑과 온라인 쇼핑의 두 가지 장점 모두를 챙긴 쇼핑을 할 수 있다. 따라서 오프라인 매장의 영업 효과를 배가시킬 수 있을 것으로 예측된다.

4. 후기

박지현 : 실무중심산학협력프로젝트 팀 프로젝트를 통해 처음 접해보는 안드로이드 앱 개발을 진행하면서, 안드로이드 스튜디오로 작업하는 과정이 익숙하지 않아 큰 어려움을 겪었습니다. 이 과정에서 다양한 방법으로 공부를 하며 실력을 키우고자 시도하였는데 가장 속도가 빠르고 기억에 남는 공부 방법은, 모르더라도 일단 직접 만들어보는 것 이었습니다. 이는 책을 보고 차근차근 개념을 익히는 공부 방법만이 정확한 공부라고 생각했던 저의 생각을 변화시킨 계기가 되었습니다. 또한 이러한 장기적인 팀 프로젝트를 진행하면서 팀원 각자의 강점을 살린 역할 분담의 중요성을 깨달을 수 있었습니다. 각자 맡은바가 명확히 분할되어 있었기 때문에 모두들 자신의 역할에 책임감을 갖고 프로젝트에 임할 수 있었습니다.

이상민 : 처음 주제를 정했을 때 팀원들 모두가 안드로이드 앱 개발을 처음 수행하는 것에 대한 걱정이 많았습니다. 그래서 책이나 유튜브를 참고해가며 앱의 한 페이지씩 만들어갔고, 그 과정에서 수많은 오류를 접했습니다. 그 중 제일 오래 걸렸던 부분이 바로 텍스트 인식이었습니다. 우리 팀의 프로젝트 주제가 상품명을 인식시켜 주변 매장을 찾아주는 것이기 때문에 텍스트 인식을 성공하지 못한다면 개발 진행을 하지 못하는 상황이었습니다. 하지만 팀원들끼리 합심하여 성공해냈고, 그에 대한 성취감이 상당히 높았습니다.

그리고 안드로이드 스튜디오는 레이아웃의 변화를 직접 눈으로 보며 작업하기 때문에 처음 접하는 프로그래밍에도 불구하고 흥미를 갖고 할 수 있었습니다. 여러 레이아웃의 UI를 작업함으로써 제가 서버와 같은 back-end보다는 front-end에 더 적합하다고 생각하는 좋은 계기가 되었습니다.

박민혁 : 처음 아이디어를 제시하고 기능들을 생각 했을 당시만 해도 자신감이 많았습니다. 하지만 초반부 데이터베이스와 안드로이드 스튜디오 공부를 할 때 많았던 자신감은 다 사라지고 걱정이 되기 시작했습니다. 완성을 할 수 있을까? 이런 걱정들이 팀원들 모두가 하고 있어서 팀원 전체가 자기가 맡은 분야에 대해 각자 공부를 열심히 했던 것 같습니다. 책이나 유튜브 구글링을 통해 개발을 하기 시작했습니다. 초반에는 투자한 시간에 비해 결과물이 조금씩 나와서 팀원들 모두 힘들어 했습니다. 하지만 조금씩 팀원들 모두 안드로이드 스튜디오를 잘 다룰 수 있게 되었고 데이터베이스구축 역할을 맡은 인원들도 조금씩 안드로이드 스튜디오와 연동이 되다 보니 흥미를 갖고 맡은 역할에 충실히 임할 수 있었습니다.

실제 어플리케이션 개발을 과에서 접할 기회가 없어서 이번 기회에 어플리케이션 개발에 대해 공부 할 수 있게 되어서 좋았고, 3년전 데이터베이스 강의를 들었지만 기억이 잘 나지 않았는데 이번 기회에 다시 한번 공부 할 수 있게 되어서 좋은 경험이 되었던 것 같습니다.

김소담 : 이번 프로젝트를 수행하면서 지금까지 다양한 학부 수업을 통해 학습한 지식들을 실무적으로 직접 활용해볼 수 있었습니다. 여러 개발 환경을 새롭게 다뤄보면서 스스로도 많은 자료를 찾아 공부해 나갔습니다. 안드로이드 스튜디오에서 java 언어로 안드로이드 앱을 개발하면서 다양한 오픈소스를 찾아 분석해보았고, 직접 웹서버를 구축하고 데이터베이스를 관리해보면서 이론적으로 이해했던 개념들이 실제로 어떻게 적용되는지 확인해볼 수 있었습니다. 물론 프로젝트를 수행하는 과정에서 많은 오류들을 해결해야 했지만, 팀원들과 머리를 맞대고 하나씩 해결해나가기 위해 노력하였습니다. 이 프로젝트는 팀원들이 다 함께 협력하며 하나의 결과물을 완성해가며 보람을 느낄 수 있었던 뜻 깊은 기회였습니다.