

2020학년도 졸업작품 보고서

인공지능시스템프로젝트

2020년 12월

개발 팀 : COLORFUL

(지도교수 : 조강홍)

팀장 : 이찬준(20173853)

박민성(20164107)

김태헌(20164114)

류상배(20191796)

임현아(20191783)

조나영(20190717)

동양미래대학교 컴퓨터정보공학과

〈차 례〉

I. 사용자 설명서.....	3
1. COLOR FULL이란?.....	3[이찬준]
2. 앱 설치/실행 방법.....	3[조나영]
3. 메뉴 설명.....	3[조나영]
(1) 주 메뉴.....	4[조나영]
(2) 커뮤니티 메뉴.....	4[이찬준]
(3) 기타 메뉴.....	4[조나영]
4. 각 부분/버튼 사용법.....	6[박민성, 조나영]
5. 오류 발생 시 대처 방법.....	7[박민성, 조나영]
II. 개발완료보고서.....	9
1. 개발 작품명.....	9[이찬준]
2. 개발 팀 소개.....	9[이찬준]
3. 개발 작품 개요.....	9[이찬준]
4. 개발 작품 목적.....	9[이찬준]
5. 개발작품 분야 : 개발작품 시스템 환경, 개발 언어 등.....	10
[이찬준, 김태현, 류상배, 임현아, 조나영]	
6. 개발 작품 기능 : 시스템 구성도, 메뉴 구성도.....	11[김태현, 조나영]
(1) 검색.....	11[조나영]
(2) 옷감리스트.....	11[조나영]
(3) 커뮤니티.....	11[이찬준]
(4) 설정.....	12[조나영]

7. 유사 제품 비교.....	13[김태현]
8. 개발 방법 및 범위.....	15[이찬준, 김태현, 류상배, 임현아, 조나영]
9. 개발 일정.....	16[이찬준, 김태현, 류상배, 임현아, 조나영]
10. 개발 결과 및 효과.....	18[이찬준]
11. 개발 소감.....	18[이찬준, 박민성, 김태현, 류상배, 임현아, 조나영]
12. 소스 코드.....	21[이찬준, 박민성, 류상배, 임현아, 조나영]

〈그림 차례〉

[그림 I -1] 개발 작품 - [시스템] 버니어캘리퍼스 사용 예시.....	6[박민성]
[그림 I -2] 개발 작품 - [시스템] 블루투스모듈 사용 예시.....	6[박민성]
[그림 I -3] 개발 작품 - [APP] 상인.....	7[조나영]
[그림 I -4] 개발 작품 - [APP] 고객.....	7[조나영]
[그림 II -1] 개발 작품 기능 : 시스템 구성도 (시스템).....	11[김태현]
[그림 II -2] 개발 작품 기능 : 시스템 구성도 (AI-APP-DB).....	11[조나영]
[그림 II -3] 유사제품비교 - KIWI.....	14[김태현]
[그림 II -4] 유사제품비교 - 미니노트.....	14[김태현]
[그림 III -1] 개발 작품 소스코드 - [DB] COLOR FULL ERD.....	21[임현아]

I. 사용자 설명서

1. COLOR FULL이란?

저희 COLORFUL조의 작품 COLOR FULL이란 원단을 측정하여 그 값을 통해 사용자가 원하는 정보를 제공해주는 프로젝트입니다. 버니어 캘리퍼스와 AI를 이용하여 원단의 두께, 비침, 색상 그리고 비침의 정도 등을 측정하고 어플리케이션을 통해 측정값과 가장 유사한 원단을 찾아줍니다. COLOR FULL은 원단 시장에서 가장 효율적으로 활용될 수 있습니다. 원단 시장에서 원단에 대한 배경지식이 없는 일반 구매자들의 경우 본인이 찾고 있는 원단을 찾는데 어려움을 겪는 경우가 많다. 이러한 불편을 해소하고자 누구나 쉽게 원단을 찾을 수 있는 프로젝트를 개발하였다.

2. 앱 설치/실행 방법

조나영의 깃허브(<https://github.com/chonayoung>)에서 해당 앱을 만든 프로그램을 다운받아서 안드로이드 스튜디오 폴더에 압축을 푼 다음에 안드로이드 스튜디오를 실행을 합니다. 그 후 노트북이나 컴퓨터에 핸드폰을 연결하시고 안드로이드 스튜디오에서 실행하시면 핸드폰에 앱이 깔립니다.

3. 메뉴 설명

처음에 들어가게 되면 스플래시 화면이 보입니다. 그 화면은 앱에 대한 전체적인 사용에 대해 기록되어 있습니다.

처음으로 사용하시는 사용자라면 회원가입을 하셔야 합니다.

회원가입을 하시는 이유는 사용자가 고객인가, 상인인가에 대해 서비스가 달라지기 때문입니다.

(1) 주메뉴

옷감 검색 기능 (센싱 -> 옷감추출 -> 선택지 결정 -> 다음 -> 결과)

1. 로그인해서 들어가시면 첫 화면에 ‘옷감 찾기’ 버튼이 있습니다. 그 버튼을 누릅니다.

2. 새로운 화면에서 처음으로 연결을 누릅니다. 모두 허용을 누릅니다.

3. 기기 스캔을 누릅니다. HC-06를 선택합니다.

4. 연결완료가 뜰 때까지 기다립니다.

5. 센싱기를 이용해 측정을 합니다. 시간이 지나면 값이 센싱기에서 옵니다.

6. 연결해제를 누릅니다.

7. 색상분석을 누릅니다. 그 후 나오는 메시지에 허용을 누릅니다.

8. 이미지 가져오기를 눌러서 찾고 있는 옷감사진을 선택합니다.

9. 색상추출을 누릅니다.

10. 옷감을 만져서 해당되는 옷감 촉감, 광택감, 신축성을 선택하시고 다음을 누르면 해당되는 옷감들이 검색됩니다.

11. 해당 옷감이 맞다고 여겨지면, 그 옷감카드를 누르면 해당 옷감을 보유하고 있는 가게리스트가 뜨게 됩니다.

(2) 커뮤니티 메뉴

어플에 로그인한 상인과 고객 등 모든 이용자들이 옷감의 정보 및 구매 시 필요한 정보를 공유하고 구매 후기 등을 남길 수 있습니다. 글쓰기, 댓글달기 등의 기능을 제공하며 해당 글의 조회 수 및 게시 날짜가 표시됩니다.

(3) 기타 메뉴

옷감리스트 - 상인이 보유하고 있는 옷감들을 볼 수 있는 리스트입니다.

1. 추가 - 상인이 옷감리스트에 해당 옷감을 추가 할 수 있습니다.

2. 수정 - 옷감리스트에서 선택 후 수정 버튼을 누르면 해당 옷감정보를 수정할 수 있습니다.

3. 삭제 - 옷감리스트에서 선택 후 삭제 버튼을 누르면 해당 옷감을 삭제 할 수 있습니다.

설정 - 자신의 정보수정은 못합니다. 처음 회원가입하실 때 정보 그대로 갑니다

.

해당 메뉴를 누르시면 해당되는 웹페이지로 이동합니다.

4. 각 부분/버튼 사용법

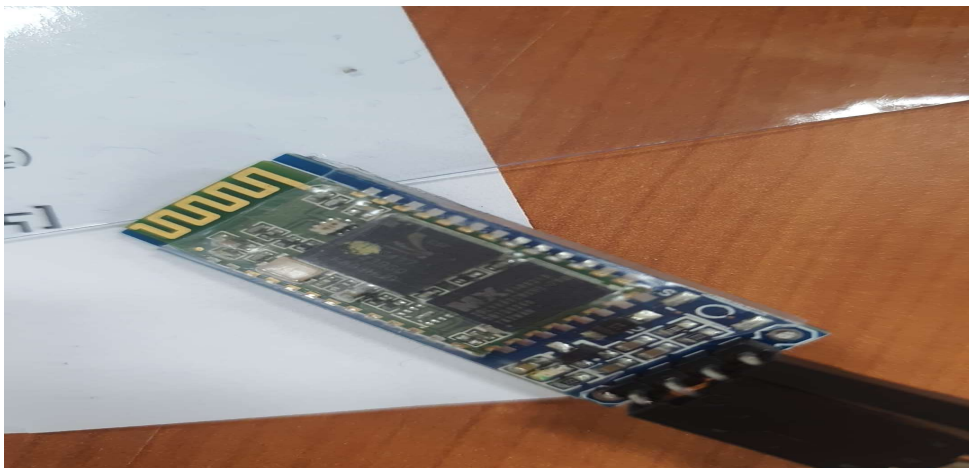
[시스템]

1. 스마트폰의 블루투스를 이용하여 센싱기를 연결한다.
2. 스마트폰 어플을 실행.
3. 원단을 사이에 넣고 서로 맞닿게 한다.



[그림 I-1]

4. 맞닿게 되면 블루투스 모듈을 이용하여 자동으로 스마트폰으로 전송을 한다.



[그림 I-2]

[APP] <상인>



[그림 I-3]

차례대로 옷감리스트, 검색, 커뮤니티, 설정부분입니다.

옷감리스트는 상인이 보유하고 있는 옷감들을 볼 수 있는 리스트들입니다.

검색은 검색 기능을 사용할 수 있는 화면입니다.

커뮤니티는 상인과 고객들이 사용할 수 있는 커뮤니티 부분입니다.

설정은 자신의 상태, 궁금한 점을 물어볼 수 있는 부분입니다.

[APP] <고객>



[그림 I-4]

차례대로 검색, 커뮤니티, 설정입니다.

검색은 검색기능을 사용할 수 있는 화면입니다.

커뮤니티와, 설정부분은 위에 상인과 동일합니다.

5. 오류 발생 시 대처 방법

[시스템]

스마트폰으로 측정값 전송 불가 오류 시

- 스마트폰의 블루투스가 센싱기와 연결이 되어있는지 확인한다.
- 스마트폰의 어플이 실행되어 있는지 확인한다.

[APP]

오류 발생시 해당 앱을 다시 깔아봅니다.

다시 깔아도 오류가 나는 부분은 오류가 날 수 밖에 없는 부분이라 생각하시고 그 부분은 선택하지 않으시면 됩니다.

만약 오류를 수정하시고 싶으시다면, 해당 파일을 확인하신 후 수정해서 사용해 주세요.

II. 개발완료보고서

1. 개발 작품명

작품명은 COLOR FULL입니다.

2. 개발 팀 소개

개발 팀명은 COLORFUL이며 박민성(시스템), 김태현(시스템), 류상배(AI), 임현아(DB), 조나영(APP), 이찬준(팀장, WEB) 등 총 6명으로 구성되었습니다.

3. 개발 작품 개요

버니어캘리퍼스를 통하여 원단의 두께를, AI를 통하여 색상 및 촉감 그리고 비침의 정도를 측정하여 작품 이용자가 찾고자 하는 원단과 가장 유사한 값을 가지고 있는 원단을 찾아주는 장치이다.

4. 개발 작품 목적

주문 제작 혹은 대량으로 옷을 제작해야 하는 경우 원단 시장에 방문하여 원하는 옷감을 찾는다. 하지만 원단에 대한 배경지식이 없는 일반 구매자들의 경우 본인이 찾고 있는 원단을 찾는데 어려움을 겪는 경우가 많다. 이러한 불편을 해소하고자 누구나 쉽게 원단을 찾을 수 있는 장치를 개발하였다.

5. 개발작품 분야 : 개발작품 시스템 환경, 개발 언어 등

박민성, 김태현(시스템): 시스템 하드웨어로는 디지털 버니어 캘리퍼스, 조도센서 모듈, 압력센서 모듈, 블루투스 모듈을 사용하였고 소프트웨어로는 아두이노를 사용하여 개발하였습니다.

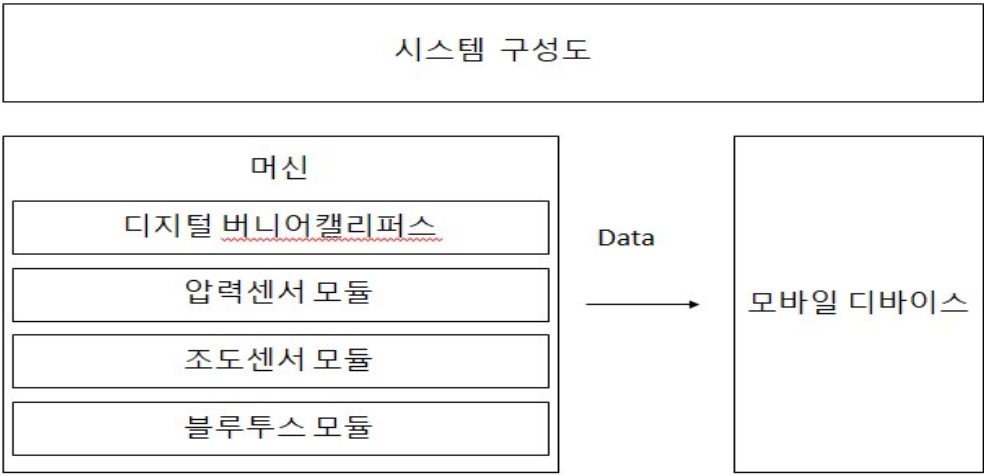
류상배(AI): eclipse와 Android Studio로 개발환경을 사용했고, 언어는 JAVA와 PHP를 사용했습니다. PHP를 사용해 어플에서 DB에 사진을 보내고, JAVA를 사용해 색상분석을 하는 코드를 작성하여 개발했습니다.

임현아(DB): phpMyAdmin 툴로 MySQL 을 사용하여 데이터베이스를 구축하였습니다. PHP를 사용하여 어플에서 DB 정보를 얻어올 수 있도록 개발하였습니다.

조나영(APP) : Android Studio를 개발환경으로 사용하였습니다. 언어는 JAVA를 사용하였습니다. 앱과 연동되는 PHP 파일들은 Brackets을 이용하여 확인하였으며, 수정을 했습니다.

이찬준(웹): 기본적인 HTML 웹페이지와 데이터베이스와의 연동에 필요한 PHP 파일은 브라우저의 개발자 도구를 활용하였습니다. 또한 WebStorm 프로그램을 통해 추가적인 개발을 하였습니다.

6. 개발 작품 기능 : 시스템 구성도, 메뉴 구성도



[그림 II-1]



[그림 II-2]

센싱기 : 버니어캘리퍼스와 조도센서를 통해 원단의 두께와 비침의 정도를 측정하여 값을 전달한다.

AI : APP와 연동하여 원단 검색 시 색상 구별을 가능하게 해준다.

APP : AI 및 센싱기와 연동하여 받은 값으로 고객이 원하는 원단을 찾게 한다.

(1) 검색

해당 기능은 옷감표본은 가지고 있으나, 옷감이름을 모르고 옷감을 찾는 사람에게 유용한 기능이다. 이 기능에서는 센서와, AI와, APP, SQL을 이용하여 옷감을 찾는다. 처음에는 센싱기를 이용하여 옷감의 두께와 비침을 측정하여 값을 APP으로 넘겨준다. 다음으로는 AI에서 옷감의 RGB를 추출하여 RGB를 APP으로 넘겨준다. 그 후 촉감, 광택감, 신축성을 고객이 직접 옷감을 만져서 선택을 한다. 센싱기에서 얻은 값과, AI에서 얻은 값과 APP에서 얻은 값을 가지고 SQL을 이용하여 DB에 접근하여 해당되는 옷감의 이름과 사진을 가지고 와서 APP에 보여준다. 그 후 해당 옷감이 있으면 옷감사진을 선택하여 옷감보유가게를 선택한다.

(2) 옷감리스트

이 기능은 상인들이 보유하고 있는 옷감을 추가, 수정, 삭제를 할 수 있다. 이 기능을 이용하여 상인들이 보유하고 있는 옷감에 대해 DB에 저장하므로, 검색기능에서 고객이 찾고 있는 옷감을 보유하고 있는 가게를 보여줄 수 있다. 해당 기능은 리스트 뷰를 이용하여 구현하였다. 추가를 누르면 옷감을 추가할 수 있는 페이지로 넘어가게 되고 해당 값들이 DB에 저장된다. 옷감리스트에서 하나의 옷감을 선택한 후 수정버튼을 누르면 해당 옷감을 수정하여 DB에 upload를 할 수 있다. 똑같이 하나의 옷감을 선택한 후 삭제버튼을 누르면 DB에서 해당 데이터를 삭제할 수 있다.

(3) 커뮤니티

이 기능은 상인과 고객들이 자유롭게 의견을 주고받으며 옷감의 정보 및 구매 시 필요한 정보를 공유하고 구매 후기 등을 남길 수 있습니다.

(4) 설정

이 기능은 로그인한 사용자의 정보와 문의정보에 대한 기능이다. 이 기능을 넣을 때 고객 정보를 수정할 수 있게 기획을 했으나, 시간이 부족하여 수정하는 부분은 빠지게 되었다. 현재 사용자의 이름과 사용자가 고객인지, 상인인지 보여주고, 상인이면 상인의 매장이름을 보여준다. 문의정보는 공지사항, Colorful이란?, 문의사항이 있다. 해당 정보들은 웹뷰를 이용하여 웹과 연동하였다. 웹을 이용하여 해당 정보들을 볼 수 있다.

7. 유사 제품 비교

● Dplaza 원단찾기

동대문 원단시장 웹페이지에 존재하는 원단 찾기 의뢰 페이지다. 원단을 찾기 원하는 손님이 원단의 사진을 찍어서 원단 전문가들이 직접 찾아서 연락을 준다.

장점: 전문가가 직접 찾아서 올려주는 것이기 때문에 의뢰인이 원하는 원단을 찾아 줄 거라는 신뢰가 있다.

단점 및 예시안과 비교시: 의뢰 신청을 위해선 각 원단마다 건당 4만원의 선불을 받고 찾는 시간은 보통 3~4일이 걸린다. 또한 원단을 찾지 못하는 경우에도 수수료 5,000원 정도만 환불 받을 수 있으며 직접 찍은 사진이 아니거나 원단의 실제 이름을 모른다면 찾을 수 없다는 단점도 있다. 머신이 구현이 된다면 의뢰를 자주 하는 사람이라면 장기적으로 볼 때 금전적인 실 이 줄어들 것 같다. 시간은 확실히 더 빨라질 수 있을 것이라고 예상한다.



[그림 II-3]

원단 및 부자재를 취급 매장과 이를 찾고 있는 고객을 스마트하게 연결하는 앱이다. 매장의 사장들이 직접 사진을 올리고 고객은 그 사진들을 직접 보는 시스템이다.

장점: 각 매장의 사장들이 직접 사진을 올리기 때문에 이미지 컷이 아닌 원단별 실제 사진을 직접 볼 수 있다. 또한 검색결과로 매장의 위치를 지도로 표시해준다.

단점 및 예시안과 비교 시: 매장의 위치를 지도로 표시는 해주나. 그 지도가 시장상가가 익숙한 사람이 아니라면 찾기 힘들 정도로 간략하기 때문에 테마가 직접 같이 찾아주는 것이 더 좋을 것 같다. 또한 원단의 경우 비슷해 보이는 것들 중에서도 조금씩 다른 것들이 있기 때문에 직접 측정값으로 비교하는 부분 보다는 눈으로만 비교하는 것은 차이가 클 것이라고 생각한다.



[그림 II-4]

스마트 검색을 이용해서 고객이 찾는 원단을 검색해 구매할 수 있는 앱이다. 일정 관리 기능을 제공하며 방문할 매장들을 설정하여 임의의 코스를 만들 수도 있다.

장점: 여러 가지 검색 기능을 통하여 찾고자 하는 원단을 빠르게 찾을 수 있고 필터 기능을 사용하여 원하는 무늬의 원단만을 찾을 수 있다고 한다. 또한 방문 코스 같은 기능은 실제로 시장상가에 자주 가는 사람들에게 아주 필요한 기능이라고 생각한다.

단점 및 예시안과 비교 시: 이 어플리케이션 또한 눈으로만 보는 것이기 때문에 위에

키위랑 같은 부분이 있을 것이라 생각한다. 또한 방문코스 같은 부분은 능숙한 사람에게서는 정말 좋은 기능이지만 그렇지 않은 사람에게서는 사실 있으나 마나한 부분이기 때문에 어플리케이션이 전체적으로 능숙한 사람에게 맞춰져 있는 것 같다.

8. 개발 방법 및 범위

박민성, 김태현(시스템): 시스템 개발 사용언어로 아두이노를 사용하였습니다. 작품 제작 및 개발 도구로 인두, 납, 아두이노 키트를 사용하였습니다. 또한 실제 작품의 전체적인 구성 설계 및 조립을 전담하였으며, 작품의 기능 코딩과 서버 연결 부분 PHP를 작성하였습니다.

류상배(AI): 사진의 색상분석을 하는 부분을 맡았기 때문에 그에 맞게 표본 이미지를 쉽게 많이 축적할 수 있도록 APP에서 서버에 사진을 전송하는 PHP 코드를 작성하였고, 직접적으로 사진을 JAVA를 사용해 색상을 R(빨강) G(초록) B(블루)로 분류해 0~255 사이의 숫자 값으로 표현합니다.

임현아(DB): DB(데이터베이스), 기본적인 데이터를 구축하기 위해 phpMyAdmin 툴로 MySQL 을 사용하였습니다. 어플 내에서 데이터베이스에 있는 정보를 사용하기 위해 PHP를 개발하였습니다.

조나영(APP) : 전체적인 앱을 구성하였습니다. 대표적으로 로그인화면, 회원가입 화면, 상인에게 보여줄 화면, 고객에게 보여줄 화면, 검색과정, 설정부분, 웹뷰를 이용하여 커뮤니티를 개발하였습니다. 해당 화면들은 모두 Android Studio를 이용하여 만들었습니다.

이찬준(WEB): 크게 데이터베이스와의 연동을 위한 php 파일 작업과 내용을 출력하는 웹페이지 구성 두 가지를 중점적으로 개발을 하였습니다. php 파일 및 웹페이지 기본 틀을 잡을 때는 브라우저 개발자 도구를 이용하였고, 추가적으로 WebStorm 프로그램을 이용하였습니다.

9. 개발 일정

시스템: 5월~7월에는 부품 지원 대기 및 시스템 구상을 하였고, 8월엔 미리 설계해 놓았던 구상도를 토대로 아두이노 임시 코드를 작성하였습니다. 9월엔 센싱 장치 중간 완성 및 코드 업데이트를 진행하였고, 10월에는 장치 DB 연결 및 테스트를 하였습니다. 최종적으로 11월에는 코드를 완성하고 작품의 완성도를 위하여 전체적인 테스트를 진행하였습니다.

AI: 초기 개발 단계인 4~5월 에는 원단의 이미지 자체를 비교하는 데에 중점을 두어 Python을 사용해 이미지의 모든 특징을 분류하는 OpenCV를 사용한 이미지 유사 분석을 사용하였는데 DB에 있는 여러 이미지들과 분석을 하는데 제한이 있어 다른 방법을 찾았습니다. 5~7월까지 Mahotas라는 모듈을 사용하여 유사도 자체를 측정하는 코드를 작성하였고 원하는 결과와 값을 얻었습니다. 하지만 로컬에선 동작을 잘하다가 서버에 넣어 실행하면 어김없이 오류가 발생해 오류를 잡고 있었습니다. 그러던 중 2학기가 되고 교수님과 면담 후 다른 파트와 협업하는데 중점을 두어 색상만을 분석하는 코드를 JAVA로 구현하였습니다.

DB: 4, 5월 달엔 기본적인 데이터베이스의 ERD 설계하고 틀을 구축하였습니다. 5월 막바지에 PHP와 어플이 연동할 수 있도록 찾아보기도 하였습니다. 6월~11월은 필요에 따른 데이터베이스 수정과 어플과 DB를 연동해주기 위해 PHP를 설계하였습니다. 6월엔 회원가입과 로그인, 원단 찾기 PHP를 설계하였습니다. 8월 달에 설정 기능에서 회원에 대한 정보를 보여줄 수 있도록 PHP를 설계하였으며 9월 달에 상인의 옷감을 보여주고 삭제, 업데이트할 수 있도록 PHP를 설계하였습니다. 이후에 PHP 오류가 나오면 수정하고 필요에 따라 데이터베이스 틀을 수정하였습니다.

APP: 1학기(4~6월)엔 스플리시화면, 로그인화면, 회원가입화면, 옷감기록화면을 개발하였습니다. 이 때 php와 안드로이드 앱과 연동하기 위해 2주간의 시간을 보냈습니다. 여름방학 기간(8~9월)에는 상인들이 확인하는 옷감리스트부분을 중점적으로 개발하였습니다. 팀원과 소통을 하면서 옷감기록은 빠기로 결정하였고, 옷감리스트와 검색기능을 위주로하기로 결정했습니다. 2학기(10~11월)에는 설정부분, 웹뷰로 커뮤니티, 검색기능을 위주로 개발하였습니다. 옷감 검색기능에서 센서와 연동되는 부분을 거의 3주간 붙잡고 구현하였습니다.

WEB: 1학기 개발 기간엔 개발자 도구를 이용한 기본적인 HTML 웹페이지 제작 및 데이터베이스와의 연동을 위한 php 개발을 준비하였고, 여름방학 기간 및 2학기에 해당 내용을 완성 및 WebStorm 프로그램을 이용하여 추가적인 웹페이지 구성 및 제작을 하였습니다.

10. 개발 결과 및 효과

최초 프로젝트 구상 시 중요시 했던 두 가지 포인트를 잘 살린 작품을 완성하였습니다. 첫째는 버니어 캘리퍼스를 이용하여 구매자가 찾고자 하는 옷감의 두께의 값을 측정해내는 것이고, 둘째는 AI를 이용하여 해당 옷감의 색상을 찾아내고, 사용자가 원하는 촉감 등을 선택하여 찾아내는 것이었습니다. 최종적으로 이러한 두 가지 경로를 통해 얻어낸 값에 모두 적합한 원단을 찾아내어 원단의 판매처 위치에 대한 정보까지 구매자에게 제공해주는 기능까지 구현하였습니다. 이 작품이 실제로 활용된다면 구매자에게 더 합리적인 선택을 할 수 있는 기회를 제공할 수 있을 것입니다.

11. 개발 소감

박민성 (시스템)

팀 프로젝트로 COLOR FULL이라는 팀을 이루어 프로젝트를 진행하며 팀원들과 어울려 개발을 진행할 때의 결과물은 나 혼자 열심히 하여 얻어지는 결과물도 좋을 수가 있지만 혼자 개발을 할 때보다 더 좋은 결과물을 낳는다고 생각이 들게 되는 계기가 되었고 이런 생각과 경험들은 사회생활을 할 때에 많은 도움이 될 것이라고 생각합니다. 또한 COLOR FULL에서 시스템을 맡아 개발을 진행하면서 느낀 점으로 초반에는 시스템에 대한 지식이 부족하여 진행에 차질이 많았지만 시스템에 대해 공부를 진행하며 지식을 쌓았고 이 지식들을 활용하여 개발에 성공하였습니다. 개발에 성공을 하고나니 저는 부족하다고 생각만 하지 않고 노력을 하는 삶을 살아가야겠다는 마음가짐이 생기게 되었습니다.

김태헌 (시스템)

1학기 팀 프로젝트 팀장을 맡으면서 좋은 경험을 했습니다. 각자에게 적합한 계

획과 할일을 지정해주고 그 계획을 할 수 있게 계속 확인해주고 응원하는 것과 혹시 못했더라도 다시 할 수 있게 다독여주는 것들 말입니다. 이러한 경험을 통해 앞으로 관리자 위치에 올랐을 때 좋은 초석이 될 수 있을 거라 느꼈습니다. 2학기에는 취업에 의해 팀원으로 역할이 바뀌었지만 같이 정한 계획에 동참해서 맡은 일을 정해진 기간까지 해내는 일이 있었습니다. 지식으로나 경험으로나 부족한 부분이 많아서 진행하면서 어려운 일들이 많았지만 후에 도움이 많이 될 수 있는 경험들 이었습니다.

류상배(AI)

처음 종합설계 수업을 들으면서 선배 분들이 만든 졸업 작품들을 보고 사실 나름 졸업 작품인데 너무 쉬운 걸 하신 게 아닌가? 생각을 했었습니다. 그래서 뭔가 의미가 있는 작품을 선정하고 싶어서 팀 내 작품을 투표로 결정했을 때 관심이 있는 사람이라면 찾을 법하고 수익성이 생길만한 아이템에 한 표를 던졌습니다. 그런데 막상 개발을 하면서 설계도 해보고 시스템 파트랑 이야기 하면서 시스템 적인 부분 도면을 구상해보고 앱과 웹의 구조를 그 것들이 어떤 기능을 하고 어떤 정보들을 주고받을지 구상하면서 정말 쉽지 않다는 생각을 많이 했습니다. 결과적으로 큰 틀은 잡았지만 개인적으로 퀄리티가 많이 떨어지는 작품을 냈다고 생각합니다. 원하는 방향으로 진행되지 않아 작성한 코드들을 지우고 또 지우고 스트레스도 많이 받았고, 도중에 취업한 팀원들도 많아서 힘든 시기도 있었지만 인원들 모두 단합하여 안정적으로 끝마친 것 같아 끝나고 난 뒤 기분이 굉장히 상쾌했습니다. 졸업 작품이라 신경을 많이 썼음에도 실력 부족으로 완성도가 낮은게 남은 과업이라면 과업인 것 같습니다.

임현아(DB)

프로젝트의 중점은 처음 설계를 유의하라는 것이었습니다. 컬러풀은 제 생애 첫 프로젝트였습니다. ERD를 설계하는 데에 ‘어떻게 설계해야 데이터 접근성이 높아

질까?’ 고민하였습니다. 몇 주를 투자하여 만든 ERD는 개발하는데 문제가 있어 여러 번 수정되었습니다. 후엔 매우 복잡해보이던 ERD가 간단하게 수정되었습니다. 처음 설계를 유의하였다면 개발에서도 문제가 많이 일어나지 않았을 것이라 아쉬웠습니다. 이외에도 실제 프로젝트에서는 모르는 것이 있어도 혼자서 헤쳐 나가는 방법을 배웠으며 점차 더 쉽게 개발하기 위해 무엇이 필요한지 알게 되는 좋은 경험을 하였습니다.

조나영(APP)

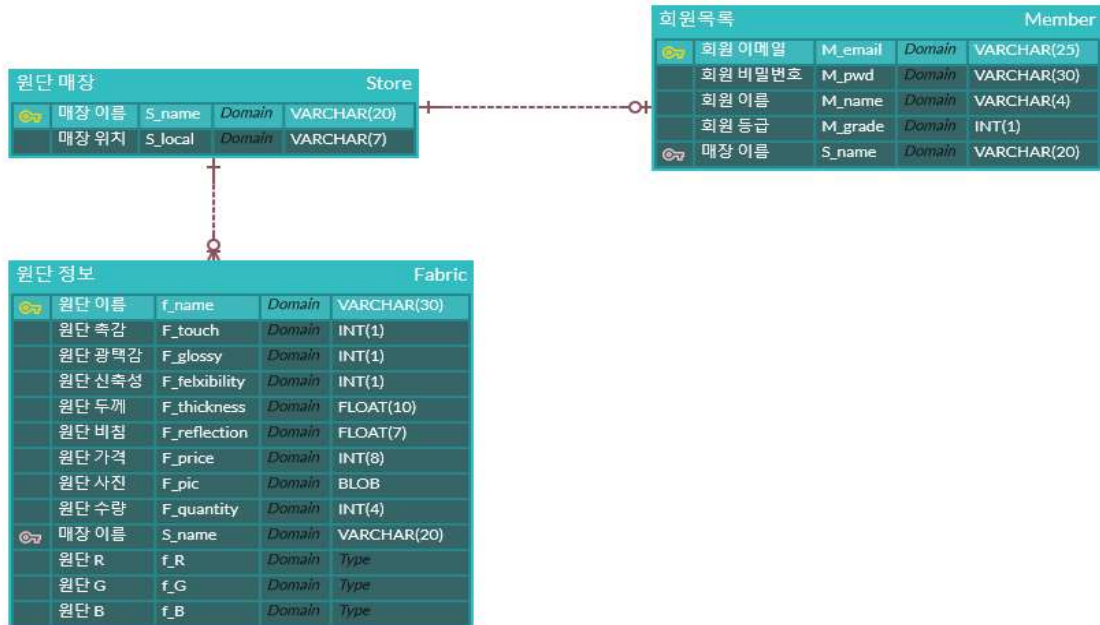
이번 프로젝트를 통해서 앱을 만드는 지식이 많이 부족하다는 사실을 깨달았습니다. 처음에는 잘 할 수 있다는 자신감으로 앱 위주의 프로젝트를 기획했으나, 시간이 가면 갈수록 저의 자신감은 떨어지게 되고 제가 만든 앱이 너무 퀄리티가 떨어지고, 고객이라면 이런 앱을 사용할까라는 불안감도 생겼습니다. 하지만, 끝까지 붙잡고 하다 보니 결국에는 컬러풀이라는 앱을 만들었습니다. 이 때 저의 감정은 성취감과 부끄러움, 공부를 더 해야겠다는 생각밖에 나지 않았습니다. 그래도 처음으로 기획하고 만든 앱이고 그래서 그런지 많은 애착이 가는 프로젝트였습니다. 이번 프로젝트로 앱에 대해 여러 가지를 알게 되었고, 그 경험을 가지고 다른 앱들을 만들어야겠다는 생각을 했습니다.

이찬준(WEB)

졸업 작품 프로젝트를 통해 비록 배경지식이 부족한 과업일지라도 포기하지 않고 노력하면 해낼 수 있다는 걸 느꼈습니다. 웹 개발에 대한 배경지식이 전무한 상황에서 해당 파트를 담당하게 되었는데 프로젝트 초기엔 엄청난 부담감과 스트레스를 받으며 프로젝트에 참여하고 있다는 자체만으로도 심적 부담을 갖게 되었는데 조금씩 공부를 하고, 팀원들의 도움을 받으면서 성장하면서 프로젝트 작품 완성보다 더 큰 경험을 얻은 것 같다는 생각이 들었습니다.

12. 소스 코드

ERD_colorful (컬러풀 ERD)



[그림 Ⅲ-1]

andsing_join2.php (안드로이드 회원가입 PHP)

```
<?php
// 안드로이드 회원가입 연동 부분
header('content-type: text/html; charset=utf-8');
// 데이터베이스 접속 문자열. (db위치, 유저 이름, 비밀번호)

$conn = mysqli_connect("211.218.150.109", "ci2020colorful", "2020colorful", "ci2020colorful"); //
MySQL 접속 및 설정 저장

// 세션 시작
session_start();

// 1. 회원 정보 넣기
// nm_? : registDB 에서 정의, sm_id : insert 에서 사용
// sm_grade: 0 상인 1 고객
// 이름, 이메일, 비번, 비번 확인(확인)

$sm_email = $_POST[u_email]; // 이메일(아이디) $_POST 내의 u_email 은 안드로이드 또는 html 에
서 받아서 사용하게 될 변수
$sm_pwd = $_POST[u_pw]; // 비번
$sm_name = $_POST[u_name]; // 이름
$sm_grade = $_POST[u_job]; // 등급

// 1. 고객일 때 정보 넣기
if ($sm_grade == 1) {
```

andsing_join2.php (안드로이드 회원가입 PHP)

```
// member 테이블의 레코드 삽입하기
$sing = "INSERT INTO `members` (m_email, m_pwd, m_name, m_grade) VALUES ('" . $sm_email
. "','" . $sm_pwd . "','" . $sm_name . "','" . $sm_grade . "')";
// 질의(위 내용)를 삽입하라.
$sing_result = mysqli_query($conn, $sing);

// sql_result 중 어디가 잘못 되었는가 확인
if (! $sing_result) {
    echo ("쿼리오류 발생: " . mysqli_error($conn));
}
}

// 2. 상인일 때 정보 넣기
if ($sm_grade == 0) {
    $own_sname = $_POST[u_storeName]; // 매장 이름
    $own_local = $_POST[u_local]; // 매장 위치

    // member 테이블의 레코드 삽입하기
    $sing1 = "INSERT INTO `members` (m_email, m_pwd, m_name, m_grade, s_name) VALUES ('" .
$sm_email . "','" . $sm_pwd . "','" . $sm_name . "','" . $sm_grade . "', '" . $own_sname . "')";
    // 질의(위 내용)를 삽입하라.
    $sing1_result = mysqli_query($conn, $sing1);

    if (! $sing1_result) {
        echo ("쿼리오류 발생: " . mysqli_error($conn));
    }

    // store 테이블의 레코드 삽입하기
    $own = "INSERT INTO store (s_name, s_local) VALUES ('" . $own_sname . "','" . $own_local .
    "')";
    // 질의(위 내용)를 삽입하라.
    $own_result = mysqli_query($conn, $own);

    // sql_result 중 어디가 잘못 되었는가 확인
    if (! $own_result) {
        echo ("쿼리오류 발생: " . mysqli_error($conn));
    }
}
}

?>
```

And_finfo.php (안드로이드 원단 정보 PHP)

```
<?php

// 안드로이드 회원가입 연동 부분
header('content-type: text/html; charset=utf-8');
// 데이터베이스 접속 문자열. (db위치, 유저 이름, 비밀번호)
$conn = mysqli_connect("211.218.150.109", "ci2020colorful", "2020colorful", "ci2020colorful"); //
MySQL 접속 및 설정 저장

// 세션 시작
session_start();

$ps_email = $_POST[u_email]; // 이메일(아이디)

//1. 원단 정보 가져오기
//1.1. 회원 테이블에서 s_name 가져오기
$sn_slct = "select s_name from members where m_email='".$ps_email."'";

//1.2. s_name으로 원단 정보 가져오기
```


And_info.php (안드로이드 원단 정보 PHP)

```

$f_info="select f_no, f_name, f_quantity, f_price, f_touch, f_glossy, f_flexibility from fabric where
s_name = (".$sn_slct.");";
$f_info_slct = mysqli_query($conn, $f_info);

$f_info_slct_result = array();
// 쿼리문의 결과(res)를 배열형식으로 변환(result)
while ($row = mysqli_fetch_array($f_info_slct)) {
    array_push($f_info_slct_result, array(
        'f_no' => $row[0],
        'f_name' => $row[1],
        'f_quantity' => $row[2],
        'f_price' => $row[3],
        'f_touch' => $row[4],
        'f_glossy' => $row[5],
        'f_flexibility' => $row[6]
    ));
}

// 2. json 형식으로 원단 이름, 매장 이름 출력
$sfslct_list = array();
// 쿼리문의 결과(res)를 배열형식으로 변환(result)
$sfslct_list = json_encode(array(
    // 배열 하나로 출력
    'finfo'=>$f_info_slct_result
));

if ($sfslct_list != null)    echo $sfslct_list;
else                        echo "Fail<br>";

?>

```

and_sinfo.php (안드로이드 가게 정보 PHP)

```

<?php

// 안드로이드 회원가입 연동 부분
header('content-type: text/html; charset=utf-8');

// 데이터베이스 접속 문자열. (db위치, 유저 이름, 비밀번호)
$conn = mysqli_connect("211.218.150.109", "ci2020colorful", "2020colorful", "ci2020colorful"); //
MySQL 접속 및 설정 저장

// 세션 시작
session_start();

//1. 옷감 이름
$pf_name = $_POST[sf_name];

//1. 옷감 이름을 통해 가게 이름, 가게 위치 가져오기
$sns_slct="select s_name from fabric where f_name = '". $pf_name ."'";

$sinfo="select s_name, s_local from store where s_name = (". $sns_slct .")";
$sinfo_slct = mysqli_query($conn, $sinfo);

$s_info_slct_result = array();
// 쿼리문의 결과(res)를 배열형식으로 변환(result)
while ($row = mysqli_fetch_array($sinfo_slct)) {
    array_push($s_info_slct_result, array(
        's_name' => $row[0],
        's_local' => $row[1]
    ));
}

```

and_sinfo.php (안드로이드 가게 정보 PHP)

```

    ));
}

// 2. json 형식으로 원단 이름, 매장 이름 출력
$s_s_list = array();
// 쿼리문의 결과(res)를 배열형식으로 변환(result)
$s_s_list = json_encode(array(
    // 배열 하나로 출력
    'sinfo'=>$s_sinfo_slct_result
));

if ($s_s_list != null)    echo $s_s_list;
else                    echo "Fail<br>";

?>

```

and_survey.php (안드로이드 원단 검색)

```

<?php
// 안드로이드 회원가입 연동 부분
header('content-type: text/html; charset=utf-8');
// 데이터베이스 접속 문자열. (db위치, 유저 이름, 비밀번호)

$conn = mysqli_connect("211.218.150.109", "ci2020colorful", "2020colorful", "ci2020colorful"); //
MySQL 접속 및 설정 저장

session_start(); // 세션

$pf_touch = $_POST[af_touch]; // 촉감(3)
$pf_glossy = $_POST[af_glossy]; // 광택감(1)
$pf_flex = $_POST[af_flex]; // 신축성(0)

$pf_thick = $_POST[bf_thick];
$pf_refle = $_POST[bf_refle];

$pf_red = $_POST[bf_red];
$pf_green = $_POST[bf_green];
$pf_blue = $_POST[bf_blue];

//1. 원단 이름, 아두이노 원단 이미지 slct
$survey = "select f_name, f_pic from fabric where f_touch='".$pf_touch.'" AND
f_glossy='".$pf_glossy.'" AND f_flexibility='".$pf_flex.'"
AND f_R='".$pf_red.'" AND f_G='".$pf_green.'" AND f_B='".$pf_blue.'";
$survey_result = mysqli_query($conn, $survey);

$sv_result = array();
// 쿼리문의 결과(res)를 배열형식으로 변환(result)
while ($row = mysqli_fetch_array($survey_result)) {
    array_push($sv_result, array(
        'f_name' => $row[0],
        'f_pic' => $row[1]
    ));
}

// 2. json 형식으로 원단 이름, 매장 이름 출력
$show = array();
// 쿼리문의 결과(res)를 배열형식으로 변환(result)
$show = json_encode(array(
    // 배열 하나로 출력
    'survey'=>$sv_result
));

```

and_survery.php (안드로이드 원단 검색)

```
if ($show != null)    echo $show;

?>
```

login.php (웹 페이지 로그인 PHP)

```
<?php
// 안드로이드 회원가입 연동 부분
header('content-type: text/html; charset=utf-8');
// 데이터베이스 접속 문자열. (db위치, 유저 이름, 비밀번호)

$conn = mysqli_connect("211.218.150.109", "ci2020colorful", "2020colorful", "ci2020colorful"); //
MySQL 접속 및 설정 저장

// 세션 시작
session_start();

$sm_email = $_POST[u_email];

//1. 회원 이름, 매장 이름, 위치 slf
$setting_info = "SELECT members.m_name FROM members WHERE m_email = ' ".$sm_email."";
$setting_info_result = mysqli_query($conn, $setting_info);

$setting_result = array();
// 쿼리문의 결과(res)를 배열형식으로 변환(result)
while ($row = mysqli_fetch_array($setting_info_result)) {
    array_push($setting_result, array(
        'm_name' => $row[0]
    ));
}

// 2. json 형식으로 원단 이름, 매장 이름 출력
$setting_list = array();
// 쿼리문의 결과(res)를 배열형식으로 변환(result)
$setting_list = json_encode(array(
    // 배열 하나로 출력
    'setting_list'=>$setting_result
));

if ($setting_list != null)    echo $setting_list;
else                        echo "Fail<br>";

?>
```

register.php (웹 페이지 회원가입 PHP)

```
<!DOCTYPE html>
<!-- HTML 파일 시작 -->

<html>
    <head>
        <title>Register</title>
        <meta charset="utf-8" />
        <meta name="viewport" content="width=device-width, initial-scale=1, user-scal
able=no" />
        <link rel="stylesheet" href="assets/css/main.css" />
```

register.php (웹 페이지 회원가입 PHP)

```

</head>
<body>
<style>
* {
    margin: 0;
    padding: 0;
    font-family: sans-serif;
}
.wrap {
    height: 100%;
    width: 100%;
    position: relative;
    padding: 4em 0 2em 0;
    background-image: url("images/header.svg");
    background-image: url("images/overlay.png"), url("images/header.svg"), -moz-linear-gradient(75deg, #4032b9 15%, #c485fa 55%);
    background-image: url("images/overlay.png"), url("images/header.svg"), -webkit-linear-gradient(75deg, #4032b9 15%, #c485fa 55%);
    background-image: url("images/overlay.png"), url("images/header.svg"), -ms-linear-gradient(75deg, #4032b9 15%, #c485fa 55%);

    background-position: center;
    background-size: cover;
    position: absolute;
}
.form-wrap {
    width: 380px;
    height: 78%;
    position: relative;
    margin: 6% auto;
    background: #fff;
    padding: 1px;
    overflow: hidden;
}
.button-wrap {
    width: 230px;
    margin: 35px auto;
    position: relative;
    box-shadow: 0 0 600px 9px #ffffff;
    border-radius: 30px;
}
.togglebtn {
    padding: 10px 30px;
    cursor: pointer;
    background: transparent;
    border: 0;
    outline: none;
    position: relative;
}
#btn {
    top: 0;
    left: 0;
    position: absolute;
    width: 110px;
    height: 100%;
    background: linear-gradient(to right, #8b85bc 15%, #c485fa 55%);
    border-radius: 30px;
    transition: .5s;
}
.social-icons {
    margin: 30px auto;
    text-align: center;
}
.social-icons img {

```

register.php (웹 페이지 회원가입 PHP)	
	<pre> width: 30px ; cursor: pointer; } .input-group { top: 80px; position: absolute; width: 280px; transition: .5s; } .input-field { width: 100%; padding: 10px 0; margin: 5px 0; border: none; border-bottom: 1px solid #999; outline: none; background: transparent; } .submit { width: 85%; padding: 10px 30px; cursor: pointer; display: block; margin: auto; background: linear-gradient(to right, #4032b9 15%, #c485fa 55%); border: 0; outline: none; border-radius: 30px; } .checkbox { margin: 30px 10px 30px 0; } span { color: #777; font-size: 12px; bottom: 68px; position: absolute; } #login { left: 50px; } #register { left: 450px; } </style> <div class="wrap"> <div class="form-wrap"> <form id = "login" class="input-group" method="post" action="sing_join.php"> <div style="color: #800080;"> <center><h1> 회원가입</h1></center>
 </div> <input type="text" class="input-field" name="m_email" placeholder="아이디(이 메일 형식)" required> <input type="text" class="input-field" name="m_pwd" placeholder="비밀번호" required> <input type="text" class="input-field" name="m_name" placeholder="이름" req uired> <label><input type="radio" name="subject" value="mer"> 상인</label> <label><input type="radio" name="subject" value="cus"> 고객</label> <input type="text" class="input-field" name="s_name" placeholder="가게이름 (상인만 입력)"> </pre>

register.php (웹 페이지 회원가입 PHP)

```

    <input type="text" class="input-field" name="m_no" placeholder="가게위치(상
인만 입력)">
        <br>
        <button class="submit"> 가입하기 </button>
    </form>
</div>
</div>
<script>
    var x = document.getElementById("login");
    var y = document.getElementById("register");
    var z = document.getElementById("btn");

    function login(){
        x.style.left = "50px";
        y.style.left = "450px";
        z.style.left = "0";
    }

    function register(){
        x.style.left = "-400px";
        y.style.left = "50px";
        z.style.left = "110px";
    }
</script>
</body>
</html>

```

psignActivity.java (앱에서 서버에 이미지 업로드하는 것을 도와주는 안드로이드 JAVA파일)

```

package com.example.colorful;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import androidx.loader.content.CursorLoader;

import android.Manifest;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.database.Cursor;
import android.net.Uri;
import android.os.Build;
import android.os.Bundle;

```

psignActivity.java (앱에서 서버에 이미지 업로드하는 것을 도와주는 안드로이드 JAVA파일)

```

import android.provider.MediaStore;
import android.view.View;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Toast;

import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.error.VolleyError;
import com.android.volley.request.SimpleMultiPartRequest;
import com.android.volley.toolbox.Volley;

public class psignActivity extends AppCompatActivity {

    EditText etName,etMsg;
    ImageView iv;

    //업로드할 이미지의 절대경로(실제 경로)
    String imgPath;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_psign);
        etName=findViewById(R.id.et_name);
        etMsg=findViewById(R.id.et_msg);
        iv=findViewById(R.id.iv);

        //업로드 하려면 외부저장소 권한 필요
        //동적 퍼미션 코드 필요..

        //동적퍼미션 작업
        if(Build.VERSION.SDK_INT >= Build.VERSION_CODES.M){
            int permissionResult= checkSelfPermission(Manifest.permission.WRITE_EXTERNAL_STORAGE);
            if(permissionResult== PackageManager.PERMISSION_DENIED){
                String[] permissions= new String[]{ Manifest.permission.WRITE_EXTERNAL_STORAGE};
                requestPermissions(permissions,10);
            }
            else{
                //cv.setVisibility(View.VISIBLE);
            }
        }

    }

    //onCreate() ..

    @Override
    public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults);
        switch (requestCode){
            case 10 :
                if(grantResults[0]==PackageManager.PERMISSION_GRANTED) //사용자가 허가 했다면
                {
                    Toast.makeText(this, "외부 메모리 읽기/쓰기 사용 가능", Toast.LENGTH_SHORT).show();
                }
                else{//거부했다면
                    Toast.makeText(this, "외부 메모리 읽기/쓰기 제한", Toast.LENGTH_SHORT).show();
                }
            default:
                //Do nothing
        }
    }
}

```

psignActivty.java (앱에서 서버에 이미지 업로드하는 것을 도와주는 안드로이드 JAVA파일)

```

        }
        break;
    }
}

public void clickBtn(View view) {
    //갤러리 or 사진 앱 실행하여 사진을 선택하도록..
    Intent intent= new Intent(Intent.ACTION_PICK);
    intent.setType("image/*");
    startActivityForResult(intent,10);
}

@Override
protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    switch (requestCode){
        case 10:
            if(resultCode==RESULT_OK){
                //선택한 사진의 경로(Uri)객체 얻어오기
                Uri uri= data.getData();
                if(uri!=null){
                    iv.setImageURI(uri);
                    //갤러리앱에서 관리하는 DB정보가 있는데, 그것이 나온다 [실제 파일 경로
가 아님!!]
                    //얻어온 Uri는 Gallery앱의 DB번호임. (content://-----/2854)
                    //업로드를 하려면 이미지의 절대경로(실제 경로: file:// -----/aaa.png 이런
식)가 필요함
                    //Uri --> 절대경로(String)로 변환
                    imgPath= getRealPathFromUri(uri);    //임의로 만든 메소드 (절대경로를 가
져오는 메소드)

                    //이미지 경로 uri 확인해보기
                    new AlertDialog.Builder(this).setMessage(uri.toString()+"\n"+imgPath).create
().show();
                }
            }else
            {
                Toast.makeText(this, "이미지 선택을 하지 않았습니다.", Toast.LENGTH_SHORT).s
how();
            }
            break;
        }
    }
}

//onActivityResult() ..
//Uri -- > 절대경로로 바꿔서 리턴시켜주는 메소드
String getRealPathFromUri(Uri uri){
    String[] proj= {MediaStore.Images.Media.DATA};
    CursorLoader loader= new CursorLoader(this, uri, proj, null, null, null);
    Cursor cursor= loader.loadInBackground();
    int column_index= cursor.getColumnIndexOrThrow(MediaStore.Images.Media.DATA);
    cursor.moveToFirst();
    String result= cursor.getString(column_index);
    cursor.close();
    return result;
}

public void clickUpload(View view) {
    //서버로 보낼 데이터
    String name= etName.getText().toString();

```


psignActivty.java (앱에서 서버에 이미지 업로드하는 것을 도와주는 안드로이드 JAVA파일)

```

String msg= etMsg.getText().toString();

//안드로이드에서 보낼 데이터를 받을 php 서버 주소
String serverUrl="http://umul.dothome.co.kr/Android/insertDB.php";

//Volley plus Library를 이용해서
//파일 전송하도록..
//Volley+ 는 AndroidStudio에서 검색이 안됨 [google 검색 이용]

//파일 전송 요청 객체 생성[결과를 String으로 받음]
SimpleMultiPartRequest smpr= new SimpleMultiPartRequest(Request.Method.POST, serverUrl,
new Response.Listener<String>() {
    @Override
    public void onResponse(String response) {
        new AlertDialog.Builder(psignActivity.this).setMessage(" 응답:"+response).create().show
0;
    }
}, new Response.ErrorListener() {
    @Override
    public void onErrorResponse(VolleyError error) {
        Toast.makeText(psignActivity.this, "ERROR", Toast.LENGTH_SHORT).show();
    }
});

//요청 객체에 보낼 데이터를 추가
smpr.addStringParam("name", name);
smpr.addStringParam("msg", msg);
//이미지 파일 추가
smpr.addFile("img", imgPath);

//요청객체를 서버로 보낼 우체통 같은 객체 생성
RequestQueue requestQueue= Volley.newRequestQueue(this);
requestQueue.add(smpr);

}
}

```

surveyActivity.java (앱에서 저장된 정보를 DB와 연동해 검색하고 색상분석을 하는 JAVA코드)

```

package com.example.colorful;

import androidx.appcompat.app.AppCompatActivity;

import android.app.Activity;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.bluetooth.BluetoothSocket;
import android.content.Intent;
import android.graphics.Color;
import android.hardware.Sensor;
import android.hardware.SensorManager;
import android.os.AsyncTask;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.ListView;
import android.widget.RadioGroup;
import android.widget.Switch;
import android.widget.TextView;
import android.widget.Toast;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.OutputStream;
import java.io.OutputStreamWriter;
import java.io.PrintWriter;
import java.util.ArrayList;
import java.util.UUID;

import app.akexorcist.bluetotohspp.library.BluetoothSPP;
import app.akexorcist.bluetotohspp.library.BluetoothState;
import app.akexorcist.bluetotohspp.library.DeviceList;

import static android.graphics.Color.BLUE;
import static android.graphics.Color.GREEN;

public class SurveyActivity extends AppCompatActivity {
    RadioGroup touch, gloss, elastic;
    Button sense_btn, next_btn;
    TextView Sresult,tv,Red,Green,Blue;
    int sre = 0;
    int t_result,g_result,e_result;
    String u_email, bt_tink, bt_light;
    String CRED,CGREEN,CBLUE;

    public static final int color = 1001;

    //블루투스 초기 세팅
    static final int REQUEST_ENABLE_BT = 1;
    static BluetoothAdapter mbluetoothAdapter = null;
    public static final int REQUEST_DEVICE_BT = 1002;
    public static final int REQUEST_COLOR_BT = 1003;

    //블루투스 연결
    private static final String NAME = "BluetoothChat";
    //Bluetooth Serial Service 중 SerialPortServiceClass UUID 값
    private UUID MY_UUID = UUID.fromString("00001101-0000-1000-8000-00805F9B34FB");
    BluetoothAdapter mAdapter;

```

surveyActivity.java (앱에서 저장된 정보를 DB와 연동해 검색하고 색상분석을 하는 JAVA코드)

```

BluetoothDevice device;

String address;
String chatName;

BluetoothSocket mmSocket;
InputStream in;
OutputStream out;

//private PrintWriter out;
//private BufferedReader in;

BackgroundTask bt;
boolean isThread = true;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_survey);

    touch=findViewById(R.id.touch_rd);
    gloss=findViewById(R.id.gloss_rd);
    elastic=findViewById(R.id.elastic_rd);
    next_btn=findViewById(R.id.next_btn);

    Red=findViewById(R.id.Red);
    Green=findViewById(R.id.Green);
    Blue=findViewById(R.id.Blue);
    tv=findViewById(R.id.tv);

    Intent i = getIntent();
    u_email=i.getStringExtra("u_email");

    Intent intent = new Intent(this.getIntent());

    Button btnColor = findViewById(R.id.btnColor);
    btnColor.setOnClickListener(new View.OnClickListener() {
        public void onClick(View v) {
            Intent intent = new Intent(getApplicationContext(), ColorActivity.class);
            startActivityForResult(intent, REQUEST_COLOR_BT);
        }
    });

    Button btnConnect = findViewById(R.id.btnConnect); //연결시도
    // 블루투스 지원여부 확인. 블루투스 어댑터 객체
    mbluetoothAdapter = BluetoothAdapter.getDefaultAdapter();
    if (mbluetoothAdapter == null) {
        Toast.makeText(getApplicationContext(), "블루투스를 지원하지 않습니다.",
            Toast.LENGTH_LONG).show();
    }

    btnConnect.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            if(!mbluetoothAdapter.isEnabled()) {
                Intent enableIntent = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
                //새로 띄운 액티비티로부터 응답을 받아야 할 경우
                startActivityForResult(enableIntent, REQUEST_ENABLE_BT);
                // Otherwise, setup the chat session
            } else {
                Toast.makeText(getApplicationContext(), "블루투스가 켜져 있습니다.",

```

surveyActivty.java (앱에서 저장된 정보를 DB와 연동해 검색하고 색상분석을 하는 JAVA코드)

```

        Toast.LENGTH_LONG).show();
        //기기 선택 ACTIVITY로 넘어감
        Intent i = new Intent(getApplicationContext(), deviceList.class);
        startActivityForResult(i, REQUEST_DEVICE_BT);
    }
}
});
//연결 끊기
Button btnDisconnect=findViewById(R.id.btnDisconnect);
btnDisconnect.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

        try {
            if(mmSocket.isConnected()) {
                if(in != null) in.close();
                if(out != null) out.close();
                mmSocket.close();
            }
            if(isThread==true) isThread = false;
        } catch(Exception e) { }
    }
});

touch.setOnCheckedChangeListener(new RadioGroup.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(RadioGroup group, int checkedId) {
        if(checkedId == R.id.touch_1) t_result = 0;
        else if(checkedId == R.id.touch_2) t_result = 1;
        else if(checkedId == R.id.touch_3) t_result = 2;
        else if(checkedId == R.id.touch_4) t_result = 3;
        else t_result = 4;
    }
});

gloss.setOnCheckedChangeListener(new RadioGroup.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(RadioGroup group, int checkedId) {
        if(checkedId == R.id.gloss_1) g_result = 0;
        else if(checkedId == R.id.gloss_2) g_result = 1;
        else if(checkedId == R.id.gloss_3) g_result = 2;
        else if(checkedId == R.id.gloss_4) g_result = 3;
        else g_result = 4;
    }
});

elastic.setOnCheckedChangeListener(new RadioGroup.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(RadioGroup group, int checkedId) {
        if(checkedId == R.id.elastic_1) e_result = 0;
        else if(checkedId == R.id.elastic_2) e_result = 1;
        else if(checkedId == R.id.elastic_3) e_result = 2;
        else if(checkedId == R.id.elastic_4) e_result = 3;
        else e_result = 4;
    }
});

//라디오 버튼 값 넘어가기
next_btn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        bt_tink="1.12";
        bt_light="869";
        //카드리스트로 이동
    }
});

```

surveyActivty.java (앱에서 저장된 정보를 DB와 연동해 검색하고 색상분석을 하는 JAVA코드)

```

        Intent it = new Intent(getApplicationContext(),ResultActivity.class);
        it.putExtra("touch",t_result);
        it.putExtra("gloss",g_result);
        it.putExtra("elastic",e_result);
        it.putExtra("light",bt_light); //비침
        it.putExtra("thik",bt_tink); //두께
        it.putExtra("u_email",u_email);
        startActivity(it);
    }
}

public void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    switch (requestCode) {

        case REQUEST_ENABLE_BT:
            if (resultCode == Activity.RESULT_OK) {
                Toast.makeText(getApplicationContext(), "블루투스를 ON 시킵니다.",
                    Toast.LENGTH_SHORT).show();
            } else {
                Toast.makeText(getApplicationContext(), "블루투스를 ON 없습니다.",
                    Toast.LENGTH_SHORT).show();
            }
            break;
        case REQUEST_DEVICE_BT :
            if (resultCode == Activity.RESULT_OK){
                //기기를 가지고 옴
                address = data.getExtras().getString("DeviceAddress");
                //블루투스
                mAdapter = BluetoothAdapter.getDefaultAdapter();
                device = mAdapter.getRemoteDevice(address);

                bt = new BackgroundTask();
                bt.execute("클라이언트 시작");

            }else{
                Toast.makeText(getApplicationContext()," 블루투스 기기가 없습니다.",Toast.LENGTH
H_LONG).show();
            }
        case REQUEST_COLOR_BT :
            if (resultCode == Activity.RESULT_OK){

                CRED = data.getExtras().getString("Red"); /*String형*/
                Red.setText(CRED);
                CGREEN = data.getExtras().getString("Green"); /*String형*/
                Green.setText(CGREEN);
                CBLUE = data.getExtras().getString("Blue"); /*String형*/
                Blue.setText(CBLUE);

            }else{
                Toast.makeText(getApplicationContext()," 취소",Toast.LENGTH_LONG).show();
            }
            break;

    }

}

class BackgroundTask extends AsyncTask<String, String, Integer> {

    protected void onPreExecute() {
        tv.setTextColor(Color.RED);
        tv.setText("연결 전");
    }
}

```

surveyActivty.java (앱에서 저장된 정보를 DB와 연동해 검색하고 색상분석을 하는 JAVA코드)

```

}

protected Integer doInBackground(String... value) {
    publishProgress(" 연결중...");
    try {
        if(mmSocket == null) {
            mmSocket = device.createRfcommSocketToServiceRecord(MY_UUID);
            mmSocket.connect();
            publishProgress(" 연결되었습니다.");
        }

        //바이트 단위의 입출력을 위한 스트림
        in = mmSocket.getInputStream();
        out = mmSocket.getOutputStream();

        //스트링 단위로 한줄씩 입출력을 위한 스트림
        //in = new BufferedReader(new InputStreamReader(mmSocket.getInputStream()));
        //out = new PrintWriter(new OutputStreamWriter(mmSocket.getOutputStream()));

    } catch (IOException e) {
        Log.i("connect", "Connection Error:"+e.getMessage());
        try {
            mmSocket.close();
        } catch (IOException e2) { }
        return 0;
    }
}

//아두이노로부터 보내는 데이터를 읽어들이는 부분
//현재는 아두이노에서 받는 데이터가없음

while (isThread) {
    try {
        // InputStream 을 통해 바이트단위로 읽어드림
        byte[] buffer = new byte[80]; //String 10글자 제한
        in.read(buffer);
        String arr = new String(buffer);
        publishProgress(arr);
        String[] array = arr.split(",");
        bt_tink=array[0]; //두께
        bt_light=array[1]; //비침

        // BufferedReader를 통한 한줄단위로 읽어드림
        //String msg = in.readLine();
        //publishProgress(msg);
        //
        //tv.setText(bt_light+", "+bt_tink);
    } catch (IOException e) { }
}
return 0;
}

protected void onProgressUpdate(String... msg) {
    tv.setTextColor(BLUE);
    tv.setText(msg[0]);
}

protected void onPostExecute(Integer result) {
    tv.setTextColor(Color.RED);
    tv.setText(" 연결을 종료합니다...");
}

```

surveyActivity.java (앱에서 저장된 정보를 DB와 연동해 검색하고 색상분석을 하는 JAVA코드)

```

}

}

```

ImageDB.php (핸드폰 카메라로 사진을 찍거나 모아서 DB에 연동되어 있어 서버로 들어가는 PHP)

```

<?php

    header("Content-Type:text/html; charset=UTF-8");

    $name= $_POST['name'];
    $msg= $_POST['msg'];
    $file= $_FILES['img'];

    //이미지 파일을 영구보관하기 위해
    //이미지 파일의 세부정보 얻어오기
    $srcName= $file['name'];
    $tmpName= $file['tmp_name']; //php 파일을 받으면 임시저장소에 넣는다. 그곳이 tmp

    //임시 저장소 이미지를 원하는 폴더로 이동
    $dstName= "uploads/" .date("Ymd_his").$srcName;
    $result=move_uploaded_file($tmpName, $dstName);
    if($result){
        echo "upload success\n";
    }else{
        echo "upload fail\n";
    }

    echo "$name\n";
    echo "$msg\n";
    echo "$dstName\n";
?>

```

시스템 - 두께측정

```

//*****버니어(두께)*****//
int i;
int sign;
long value;
float result; //두께저장
int clockpin = 4; //clockpin
int datapin = 5; //datapin
unsigned long tempmicros;

void setup() {
    Serial.begin(9600);
    pinMode(clockpin, INPUT_PULLUP); //버니어 clockpin
    pinMode(datapin, INPUT_PULLUP); //버니어 datapin
}
void loop() {

    while (digitalRead(clockpin)==HIGH) {}
    tempmicros = micros();
    while (digitalRead(clockpin)==LOW) {}
    if ((micros() - tempmicros) > 500) {
        decode();
    }

}

void decode() { // 버니어 두께 측정함수
    sign=1;

```

시스템 - 두께측정

```

value=0;
for (i=0;i<23;i++) {
  while (digitalRead(clockpin)==HIGH) {}
  while (digitalRead(clockpin)==LOW) {}
  if (digitalRead(datapin)==LOW) {
    if (i<20) {
      value|= 1<<i;
    }
    if (i==20) {
      sign=-1;
    }
  }
}
result=(value*sign)/100.00;
Serial.println(result,2);
delay(1000);
}

```

시스템 - 압력, 비침, 블루투스 전송

```

/**압력센서값 저장변수 sensorval**/
int sensorval = 0;

//*****조도(비침)*****//
int intensity;

#include <SoftwareSerial.h>

SoftwareSerial bluetooth(3, 4); // 3번 TX, 4번 RX

void setup() {
  Serial.begin(9600);
  bluetooth.begin(9600);
}

void loop() {
  //압력센서에 값이 들어오면 while을 빠져나옴
  while (sensorval == 0){
    // 압력센서값 sensorval에 저장 (0~1023)
    sensorval = analogRead(A0);
  }

  intensity = analogRead(A0);

  bluetooth.print("(" + intensity + "," + result + ")");
}

```