

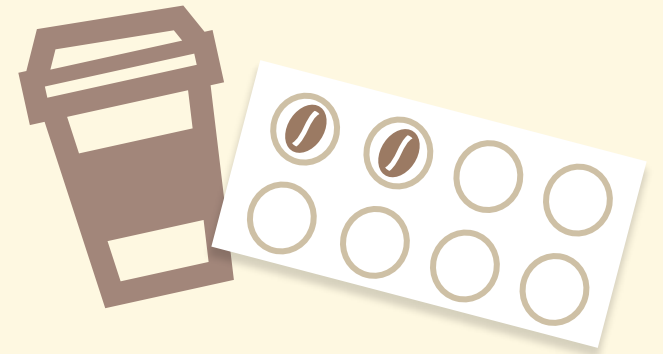
2021학년도 영남대학교 IoT 경진대회 최우수상작

제4회 KB국민은행 소프트웨어 경진대회 (출품)

과제에 묻힌 컴공 3학년 : 박지연, 현명은

ECOFFEE

: 환경을 바꾸기 위한 작은 실천을 돕다



목차 A table of contents.

- 1 작품배경 및 소개
- 2 구현기술
- 3 구현기능
- 4 시연영상
- 5 가치 및 기대효과
- 6 활용방안

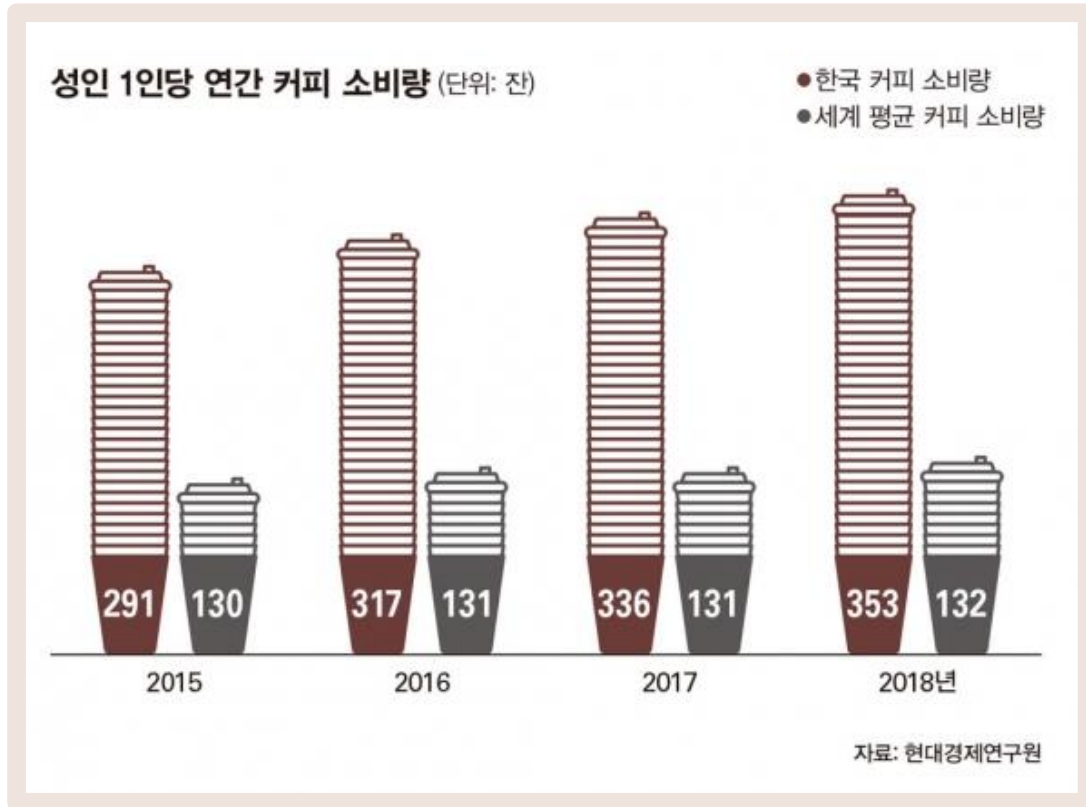
작품 배경 및 소개



2021년 전세계적으로 기업경영의 기준으로 ESG를 요구
정부에서도 사회적 활동을 담은 **‘지속 가능 경영’**을 중요시
기업의 상품이나 사업에도 **환경의 중요성**이 증가됨

작품 배경 및 소개

한국의 커피 수요와 증가에 따른 쓰레기배출



하루 평균 커피 섭취량



남성 2.0잔

여성 1.5잔

자료: 2019년 국민건강영양조사

우리나라의 커피 수요

성인 1인당 커피 소비량은 연간 353잔으로 세계 평균 소비량 132잔의 약 2.7배 수준으로 하루 평균 1.8잔을 마시는 것으로 조사됨

커피를 소비한다는 것은?

음료를 마시기 위한 도구의 소비도 포함됨

머그컵, 유리, 플라스틱 컵

<다회용품>

종이, 플라스틱 컵, 빨대, 컵홀더

<일회용품>

커피로 인해 발생하는 일회용품 폐기 물량은 가늠할 수 없음. 빨대만으로도 연간 약 657톤으로 추정

사용되고 버려진 일회용품들은 환경에 어떤 영향을 미칠까?

작품 배경 및 소개

일회용품 쓰레기가 환경에 미치는 영향

일회용 종이컵



종이컵을 만들기 위해 필요한 나무 수

일회용 플라스틱컵



바다로 흘러가버린 플라스틱으로 인한 바다 쓰레기

텀블러



일회용 컵 처리 방법

소각 대기오염 인간에게 치명적인 다이옥신이 검출되기 때문에 보통 땅에 매립을 하는 경우가 많음

매립 토양오염 토양 생태계에 큰 피해를 줌. 일회용품은 잘 썩지 않기 때문에 (최소 20년) 토양 미생물이 생기는데 방해가 됨

재활용 재활용의 어려움

종이컵은 종이 위에 액체가 새는 것을 방지하기 위해 플라스틱의 일종인 PE를 내부에 코팅하는 과정을 거침

플라스틱 컵은 각기 다양한 플라스틱 성분으로 만들어져 컵을 일일이 성분별로 분리해야함

제로 웨이스트 실천

간단한 세척만으로 재사용 가능
zero waste? 모든 제품이 재사용될 수 있도록 장려하며 폐기물을 방지하는데 초점을 맞춘 원칙

일회용품 사용량 감소

컵 홀더, 종이/플라스틱컵, 빨대 모두 필요 없음

쉬운 재활용

텀블러의 철소재는 친환경적이면서 높은 재활용율을 보임

환경을 지킬 수 있는 선택

작품 배경 및 소개

카페 유형별 사용되는 일회용품의 현실

프랜차이즈 카페

줄이기 위한 노력

텀블러 사용 시 할인혜택

프랜차이즈 카페들은 텀블러 사용을 권장하기 위해 텀블러 이용 시 쿠폰 제공 및 가격 할인 등 혜택을 제공하고 있음

[현재 브랜드별 할인 정보]

풀바셋 : 500원

엔젤리너스 : 400원

스타벅스, 투썸플레이스 외 14개: 300원

이디야, 맥도날드 외 6개: 200원

백다방 : 100원

But, 현재 상황

미미한 텀블러 사용자

텀블러 사용 시 주어지는 할인 혜택에 대해 홍보를 하고 있지만, 소비자의 텀블러 사용이 잘 이루어지지 않음

[프랜차이즈 카페 일회용품 사용실태 조사]

1,500명의 소비자를 대상으로 텀블러 지참 비율 조사 결과 **93.2%**의 소비자가 텀블러를 지참하지 않은 것으로 나타났다.

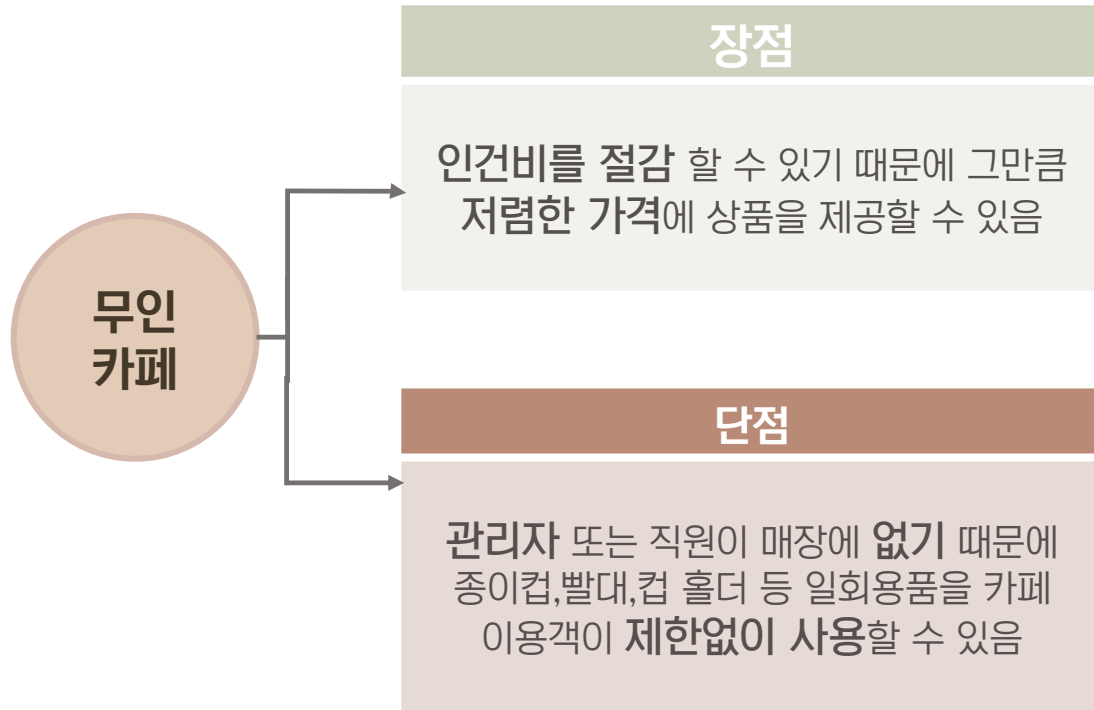
텀블러 미사용자
일회용종이, 플라스틱컵/매장컵
(1,398명)

텀블러 사용자
(102명)

➡ 어떻게 하면 텀블러 사용자를 더 늘릴 수 있을까?

작품 배경 및 소개

무인 카페의 성장



➡ 무인 카페의 특성으로 인해 환경에 좋지 않은 영향을 미침

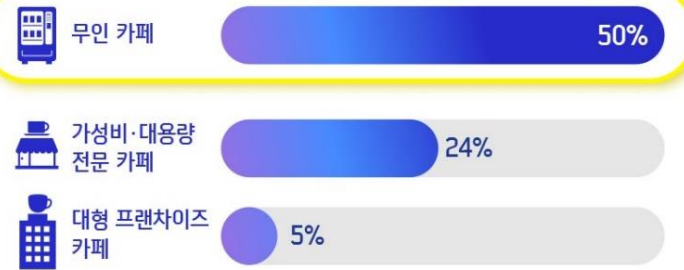
카페·문구점·펫그로서리...무인점포 전성시대

출처: 김규희 기자 | 승인 2021.10.01 08:54 | 수정 2021-10-01 08:54 | 댓글 0

출처: 신한카드 빅데이터 연구소 "점포에 직원이 없어도 익숙한 시대"

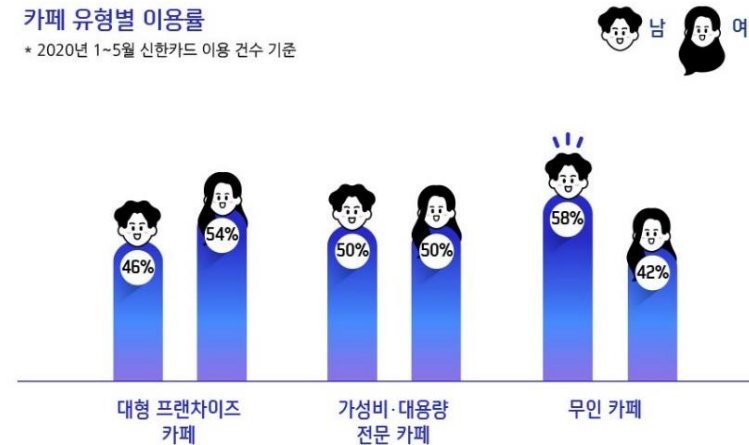
카페 유형별 전년도 대비 성장률

* 2019년 8월~2020년 1월 신한카드 이용 금액 기준



카페 유형별 이용률

* 2020년 1~5월 신한카드 이용 건수 기준



환경을 위한 커피 자판기

사용자 어플 01

멤버십 애플리케이션과 서비스의
연동 주요 고객층 확보 가능



02 텀블러 권장

스탬프는 누구에게나 주문 당 1회 씩 적립해주
지만, 8회 적립시 지급해주는 쿠폰을 이용해
텀블러 사용을 유도

단계적인 주문 03

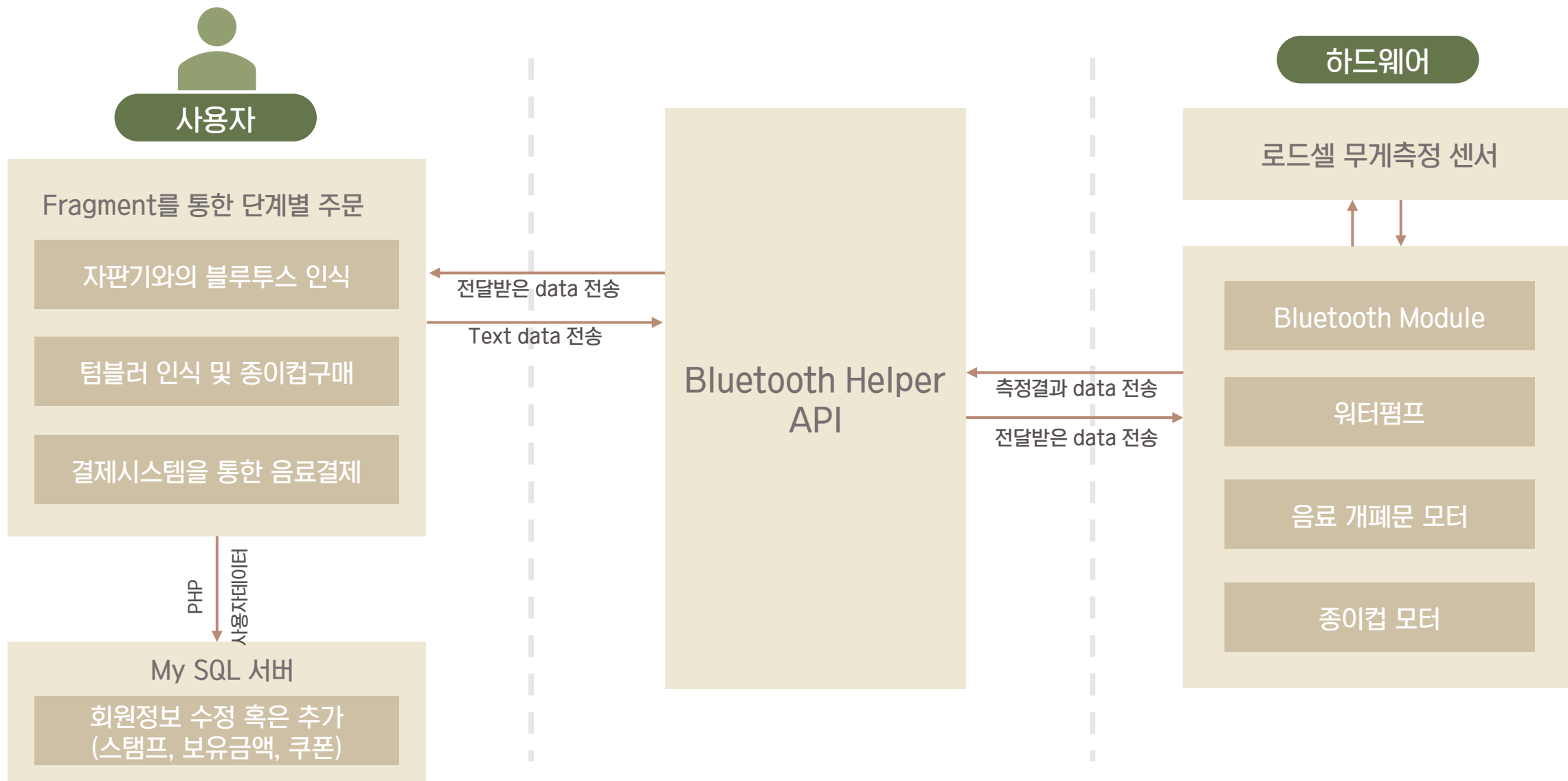
어플이가 사용자의 명확한 행동을
요구해 직관적인 사용가능. 자판기와
실시간 소통으로 음료 주문 가능

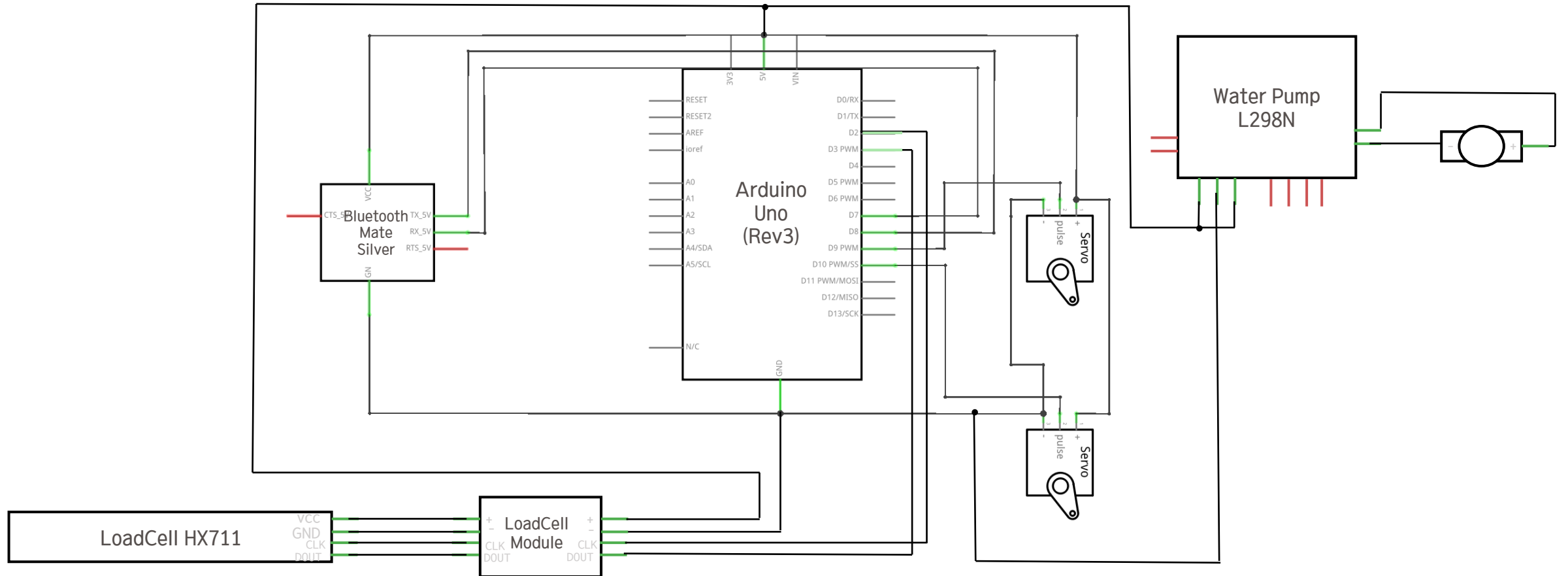
04 무인 서비스

무인 카페 서비스의 장점을 취하고 단점을 보완
해 기존의 카페와 달리 일회용품의 무분별한
사용을 막음

구현 기술

작품 구성도





워터펌프, 로드셀 무게센서, 서보모터, 블루투스 모듈을 사용하여 자판기 하드웨어를 구성함

구현 기술

서버 통신
Server



MySQL과 php



FileZilla

사용자 데이터 GET



사용자 데이터 POST



소프트웨어

Application



안드로이드 스튜디오



자바

사용자의
사용 신호 전송



자판기에서 인식한
데이터 전송



하드웨어

ECOFEE Machine



아두이노



블루투스

사용자에게 User Interface 제공
서버, 하드웨어와의 연결로 양방향 통신가능

구현 기술

주문하기

사용자는 주문하기 STEP에 따라 순차적으로 주문

주문하기

Step1 자판기와 연결해주세요

연결되었습니다

Step2 텀블러나 종이컵을 자판기에 올려주세요

대기시간: 25초

텀블러가 없으신가요? [종이컵 보기]

Step3 텀블러를 확인합니다

종이컵 or 텀블러

Step4 음료 선택



아메리카노

카페라떼

녹차라떼

Step5 결제하기

쿠폰 사용

(보유: 2장)

결제하기

실시간 통신

블루투스 모듈을 통해 사용자 앱과 자판기의 통신
사용자의 행동에 따른 실시간 응답



Step1 자판기와 연결해주세요

연결되었습니다

Step2 텀블러나 종이컵을 자판기에 올려주세요

무게가 측정되었습니다

Step3 텀블러를 확인합니다

텀블러 입니다

그 외 기능

로그인, 회원가입, 충전하기,
쿠폰, 스탬프 적립 등 부가 기능

hyun 님, 환영합니다



주문하기

충전하기

스탬프 3/8



쿠폰 x2

아메리카노 교환권

본 쿠폰은 텀블러만 사용이 가능합니다.

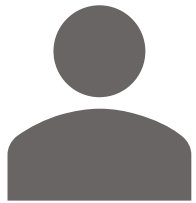
FREE

구현 기능

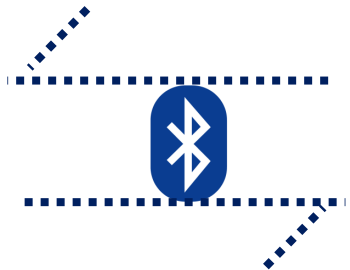
화면구성도



음료주문 시스템 흐름도



음료 주문하기



음료 자판기와
블루투스 통신 시작



로드셀 무게센서를 통해
텀블러or종이컵 판단



결과를 사용자에게 전송

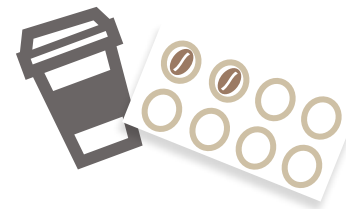


아메리카노

카페라떼

녹차라떼

음료 선택 및 결제



음료 제조 및
스탬프 획득

STEP1 자판기와 사용자 어플이 통신할 수 있도록 연결한다.

```
View view = inflater.inflate(R.layout.order_step1, container, attachToRoot: false);
step1_resultment = view.findViewById(R.id.step1_resultment);
highlight= view.findViewById(R.id.highlight);
highlight.setVisibility(View.VISIBLE);

mBluetoothAdapter = BluetoothAdapter.getDefaultAdapter();
mBluetoothAdapter.enable();

if (mBluetoothAdapter == null) {
    // Device does not support Bluetooth
    onStop();
}

@Override
public void onBluetoothHelperConnectionStateChanged(BluetoothHelper bluetoothhelper, boolean isConnected) {
    Log.d( tag: "tag", msg: "블루투스 연결 상태 체크 함수 속");
    if (isConnected) {
        step1_resultment.setTextColor(0xFF428681); //초록
        step1_resultment.setText("연결되었습니다");
        step1_result=Infomation.ConnectSuccess;
        onResume();
    } else {
        Log.d( tag: "tag", msg: "연결끊김");
        step1_resultment.setTextColor(0xAef484a); //빨강
        step1_resultment.setText("연결을 실패하였습니다 다시 시도해주세요");
        step1_result=Infomation.ConnectFail;
        onResume();
    }
}

return view;
```

- 사용자가 주문하기 화면에 들어가면 블루투스를 자동으로 켜 Bluetooth Helper API를 통해 사용자 어플과 자판기 간에 블루투스 연결을 시도한다.
- 연결을 시도하고 연결을 성공한 경우 연결 성공 메시지를 띄우며 다음 STEP2로 넘어가도록 한다. 자판기는 문을 연다.
- 연결을 실패할 경우 연결 사용자가 다시 연결을 시도하도록 실패 메시지를 띄운다.

주문하기

Step1 자판기와 연결해주세요

연결 준비중...

Step2 텀블러나 종이컵을 자판기에 올려주세요

대기시간:

Step3 텀블러를 확인합니다

종이컵 or 텀블러

Step1 자판기와 연결해주세요

연결되었습니다

or

Step1 자판기와 연결해주세요

연결을 실패하였습니다. 다시 시도해주세요

STEP2 대기시간 동안 사용자가 텀블러나 종이컵을 자판기에 올려 놓는다.

```
if(getArguments()!=null) {  
  
    if (getArguments().getInt( key: "step2_finish" ) == 0) {  
  
        Log.d( tag: "tag", msg: "step2_finish"+ getArguments().getInt( key: "step2_finish" ) );  
        highlight.setVisibility(View.VISIBLE);  
        step1_result = getArguments().getInt( key: "step1_result" ); // step1에서 받아온 값 넣기  
  
        if (step1_result == Infomation.ConnectSuccess) {  
  
            Log.d( tag: "tag", msg: "ConnectSuccess in step2");  
  
            step2_waiting.setText("대기시간: 30초");  
            timer = new CountDownTimer( millisInFuture: 30000, countDownInterval: 1000 ) {  
  
                public void onTick(long millisUntilFinished) {  
                    step2_waiting.setText("대기시간: " + millisUntilFinished / 1000 + "초");  
                }  
  
                public void onFinish() { step2_waiting.setText("대기 시간 종료"); }  
            }.start();  
  
            mBluetooth.SendMessage( msg: "Check" ); //무게 체크 메시지 보냄  
            Log.d( tag: "tag", msg: "Check신호 보냄");  
        }  
    }  
}
```

- STEP2가 시작되면 대기시간(30초) 타이머를 시작하고 자판기에게 Check라는 신호를 보내 자판기 내부에서도 타이머를 30초 작동시켜 그 시간 동안 무게센서를 활성화 시킨다.
- 타이머가 1초 씩 줄어들고 25초가 되었을 시점, 5초가 지나도 자판기의 응답이 없어 사용자는 텀블러가 없을 수도 있다고 판단해 종이컵 뽑기 버튼이 나타난다.
- 30초가 다 지나 타이머가 종료된다면 대기시간 종료 메시지를 출력하고 자판기는 문을 닫는다. 자판기와 블루투스 연결을 끊고 다시 시도하도록 한다.

Step1 자판기와 연결해주세요

연결되었습니다

Step2 텀블러나 종이컵을 자판기에 올려주세요

대기시간: 24초

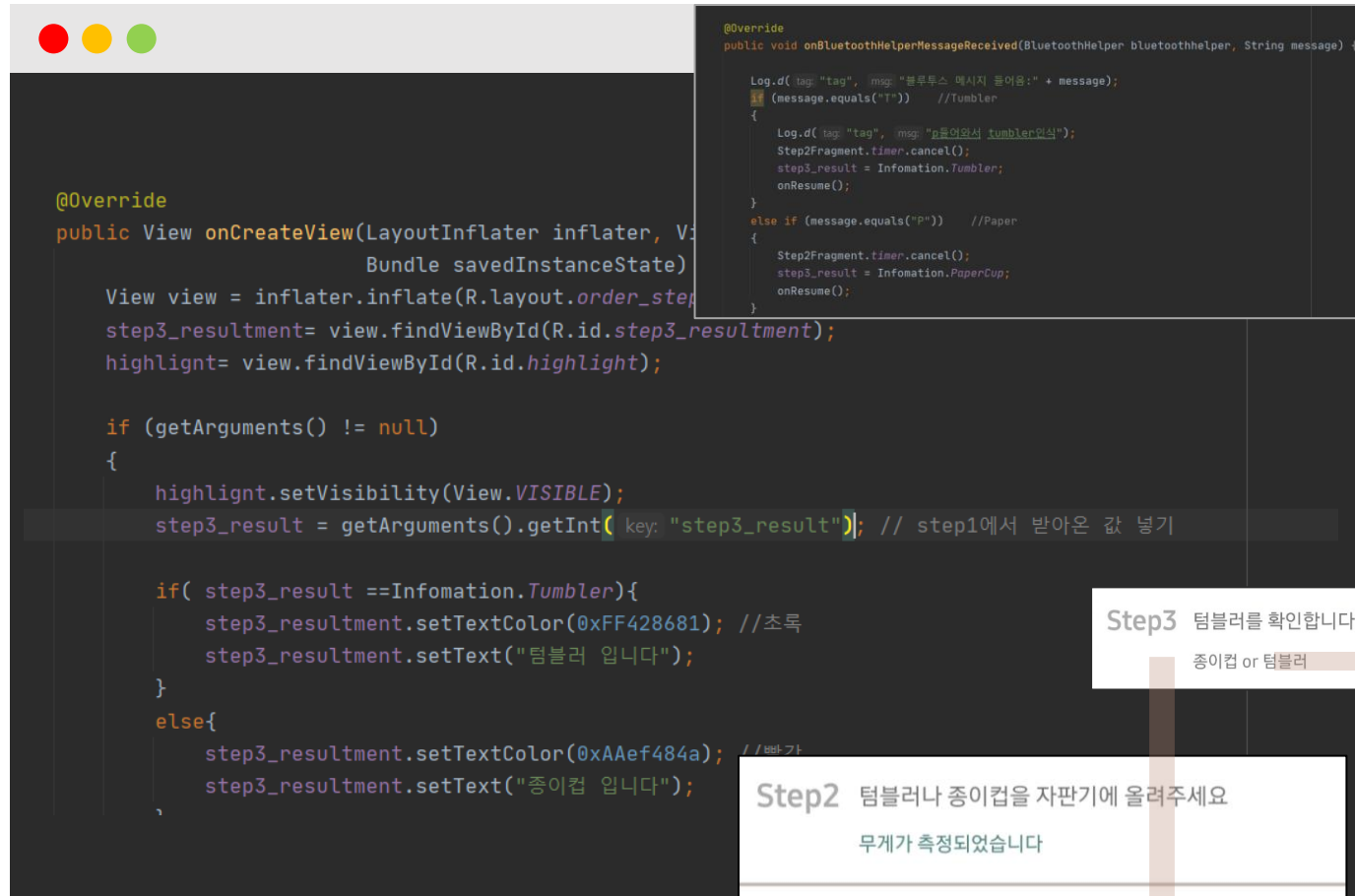
텀블러가 없으신가요? [종이컵 뽑기]

30초

Step2 텀블러나 종이컵을 자판기에 올려주세요

대기시간 종료

STEP3 자판기가 인지한 음료컵 종류 결과를 사용자에게 알려준다



- STEP2의 대기시간 동안 사용자가 사용할 음료컵을 자판기 올려 놓았다면, 자판기가 무게를 이용하여 텀블러와 종이컵을 구별한다. 이때 판단한 음료컵의 종류를 Check신호의 응답으로 사용자 어플에 전송한다.
- 어플이 자판기의 응답을 받은 후 대기시간 타이머를 종료한다. 응답이 'T' 라면 텀블러로 인지하고 'P' 라면 종이컵으로 인지한 후 STEP3의 결과를 알려준다.
- STEP3결과를 알려준 후 약 2초 후 STEP4로 넘긴다. 나머지 단계에서 STEP3의 결과값인 음료컵 종류를 활용할 것이다.

Step3 텀블러를 확인합니다

종이컵 or 텀블러

Step2 텀블러나 종이컵을 자판기에 올려주세요

무게가 측정되었습니다

Step3 텀블러를 확인합니다

종이컵 입니다

Step2 텀블러나 종이컵을 자판기에 올려주세요

무게가 측정되었습니다

Step3 텀블러를 확인합니다

텀블러 입니다

STEP4 사용자의 음료 종류를 선택한다

```
if (getArguments() != null) {  
    step3_result = getArguments().getInt( key: "step3_result"); // step1에서 받아온 값 넣기  
    highlight.setVisibility(View.VISIBLE);  
  
    americano_img.setOnClickListener(new View.OnClickListener() {  
        @Override  
        public void onClick(View v) {  
            americano.setTextColors(0xFF9B7A63); //brown  
            caffelatte.setTextColors(0xFF7F7F7F); //black  
            greenlatte.setTextColors(0xFF7F7F7F); //black  
            choice_info.setText("아메리카노 2000원");  
            choice_info.setVisibility(View.VISIBLE);  
            choice_finish.setVisibility(View.VISIBLE);  
            step4_result=Information.Americano;  
        }  
    });  
  
    caffelatte_img.setOnClickListener(new View.OnClickListener() {  
        @Override  
        public void onClick(View v) {  
            americano.setTextColors(0xFF7F7F7F); //black  
            caffelatte.setTextColors(0xFFE4C066); //Yellow  
            greenlatte.setTextColors(0xFF7F7F7F); //black  
            choice_info.setText("카페라떼 2500원");  
            choice_info.setVisibility(View.VISIBLE);  
            choice_finish.setVisibility(View.VISIBLE);  
            step4_result=Information.CaffeLatte;  
        }  
    });  
}
```

- 사용자가 음료 사진을 클릭할 때 클릭한 음료에 따라 음료의 이름과 가격을 상단에 띄워 사용자에게 음료정보를 제공한다. 하단의 음료이름 색깔을 음료색깔에 맞춰 변경하여 시각적으로 클릭된 음료를 표시한다.

- 사용자가 음료 종류를 결정하여 [선택 완료시 터치]버튼을 클릭할 때 STEP5결제하기로 넘어갈 수 있다.



STEP5 사용자가 결제하여 음료를 제공받는다.

```
@Override
public void onResume() {
    super.onResume();

    if(step5_result==Infomation.PaySuccess){

        mBluetooth.SendMessage( msg: "Pay"); //결제 메시지
        user.addStamp();
        user.subBalance(payment);

        if(step5_result==Infomation.PayUseCoupon)
            user.useCoupon();

        Intent intent = new Intent(getActivity(),ReadyAc
        intent.putExtra( name: "user",user);
        createCnt--;
        getActivity().startActivity(intent);
        getActivity().finish();

    }else if (step5_result==Infomation.PayFail){

        Intent intent = new Intent(getActivity(), MainActivity.class);
        intent.putExtra( name: "user",user);
        getActivity().startActivity(intent);

        onDetach();
    }
}
```

```
if (createCnt == 0) {

    Log.d( tag: "tag", msg: createCnt+"");
    Log.d( tag: "tag", msg: "초기화시 들어온 변동임");
    this.user = (User) getArguments().getSerializable( key: "user");

    coupon = user.getCoupon();
    step5_coupon_ment.setText( "[보유: " + coupon + "장]"); //보유 쿠폰개수만 보여줌

    createCnt++;
}

else {

    Log.d( tag: "tag", msg: createCnt+"");
    highlight.setVisibility(View.VISIBLE);
    step3_result = getArguments().getInt( key: "step3_result"); // step3에서 받아온 값 텀블러 on 종이컵
    step4_result = getArguments().getInt( key: "step4_result"); // step4에서 받아온 값 선택 음료 가격

    productmoney_won.setText(step4_result + "원");
    productmoney_won.setVisibility(View.VISIBLE);
    payment = step4_result;
    payment_won.setText(payment + "원");
    payment_won.setVisibility(View.VISIBLE);

    //텀블러 할인, 쿠폰사용 유무
    if (step3_result == Infomation.Tumbler) {
        Log.d( tag: "tag", msg: "텀블러");
    }
}
```

- STEP4에서 사용자가 선택한 음료 정보와 음료컵 종류를 판단해 상품 금액과 결제 금액, 텀블러 할인(-1000원 또는 0원)을 먼저 뺀다.
텀블러 사용자의 경우 텀블러 할인 -1000원을 적용한다. 반면 종이컵 사용자의 경우 텀블러 할인을 적용하지 않는다.
- 쿠폰사용 버튼 클릭 시 텀블러 사용자이고 쿠폰을 보유하고 있다면 “쿠폰이 적용되었습니다” 메시지를 출력하고 쿠폰 적용에 -1000원을 뺀다. 결제 금액도 1000원이 할인된 금액으로 변경한다.
- 쿠폰사용 버튼 클릭 시 종이컵 사용자일 경우 쿠폰을 보유하고 있어도 쿠폰 적용이 불가능하다. “종이컵 사용자는 쿠폰 적용이 불가합니다” 메시지를 출력하고 쿠폰 적용에 0원을 뺀다. 결제금액도 변경 되지 않는다.
- 결제하기 버튼 클릭 시 사용자의 ECOFFEE머니가 결제금액보다 많은 경우 자판기 Pay라는 신호를 전달하여 자판기가 워터펌프를 작동해 음료추출을 시작한다.
- 사용자의 ECOFFEE머니가 결제금액 보다 적은 경우 사용자의 머니와 결제금액을 출력하며 금액이 부족하다 충전할 것을 요구하는 메시지를 띄운다.

텀블러 사용자



사용자의 음료컵의 종류를 아직 모르기 때문에 보유쿠폰 개수만 알려준다.

음료선택이 되지 않았기 때문에 결제 상세 정보를 띄우지 않는다.

텀블러 사용자는 쿠폰사용이 가능하기 때문에 사용가능 쿠폰 개수를 보유 쿠폰 개수와 동일하게 표시한다.

결제 상세는 선택음료 금액을 상품금액에, 텀블러 할인을 -1000원, 쿠폰 적용을 0원, 결제금액은 녹차라떼 가격에서 텀블러 할인을 적용한 금액이다.

쿠폰을 사용하여 쿠폰적용 메시지를 띄운다.

이전과 동일한 결제 상세는 쿠폰 적용에 -1000원을 결제 금액에 녹차라떼에서 텀블러 할인과 쿠폰을 적용한을 뺀 금액이다.

종이컵 사용자



사용자의 음료컵의 종류를 아직 모르기 때문에 보유쿠폰 개수만 알려준다.

음료선택이 되지 않았기 때문에 결제 상세 정보를 띄우지 않는다.

종이컵 사용자는 쿠폰사용이 불가능하기 때문에 사용가능 쿠폰 개수는 0개로 보유 쿠폰 개수는 이전과 동일하게 표시한다.

결제 상세는 선택음료 금액을 상품 금액에, 텀블러 할인을 0원, 쿠폰 적용을 0원, 결제금액은 아메리카노 금액 그대로이다.

쿠폰사용이 불가능하여 쿠폰적용 불가능 메시지를 띄운다.

결제 상세 정보는 쿠폰 적용 전과 동일하다.



Arduino - 블루투스 모듈을 사용해 어플과 블루투스 연결

```
bluetooth.begin(9600);  
void loop() {  
    unsigned long now = millis()/1000; // 현재 시간을 저장  
    past = now;  
    while(bluetooth.available())  
    {  
        unsigned long now = millis()/1000; // 현재 시간을 저장  
        past = now;  
  
        char rcvChar = (char)bluetooth.read();  
        if(rcvChar=='\0')  
            continue;  
  
        rcvString+=rcvChar;  
        delay(5);  
    }  
  
    if(!rcvString.equals("")){  
  
        if(rcvString[0]=='C'){ //check를 받아왔다면  
  
            for(int i=0;i<90;i++){ //자판기 개폐문 열림  
                ang_door = ang_door+5;  
                if(ang_door>=180)  
                    ang_door = 180;
```

- 블루투스 모듈과의 연결 뒤, Bluetooth.begin()을 통해 블루투스 통신을 시작한다.
- Loop문 안에서 만약 블루투스 통신이 사용자 어플과 연결되어 사용할 수 있는 상황이 되었을 때 새로운 while문을 들어가 사용자 한명과의 통신을 시작한다.
- 그리고 문자열을 통해 사용자로부터 특정신호가 왔다면 신호의 문자열을 분석하여 어떤 신호인지 판단해 서보모터를 작동시키거나 워터펌프를 작동하는 등의 모션을 취한다.



Arduino - 로드셀 무게센서를 사용한 음료 컵 구분코드

```
now_weight = scale.get_units()*10;

if(now_weight <0.0 ){
    now_weight=0.0;
}
if(before_weight <0.0 ){
    before_weight=0.0;
}

if(now_weight-before_weight>=0.7){ //뭔가 무게가 느껴지는걸 인식
    if(now_weight-before_weight >=3) //텀블러로 인식
    {
        bluetooth.println("T");
        delay(2000);
        for(int i=0;i<90;i++){ //자판기 개폐문 닫힘
            ang_door = ang_door+5;
            if(ang_door>=90)
                ang_door = 90;
        }
        before_weight = now_weight;
    }
}
```

- 무게를 직접적으로 재기 전, 최초로 사용자와 통신이 되었을 경우, Scale.set_scale(calibration_factor)내장 함수와 tare()을 통해 로드셀 무게센서의 현재 값을 0으로 초기화한다.
- Now_weight = scale.get_units()*10을 통해 현재 무게센서에 감지된 값을 10을 곱해 받아온다. 무게의 오차율을 줄이기 위함이다.
- 무게를 측정하는 조건은 (현재 측정한 무게 값 - 이전의 무게 값 >=0.7)로 무게가 있다라는 것을 이전 값과 비교해서 감지한다.
- 그래서 30초의 대기시간 while문 가장 마지막에 before_weight = now_weight로 해주어 이전 값을 저장한다.



Arduino - 로드셀 무게센서를 사용한 음료 컵 구분코드

```
        servo_door.write(ang_door);  
        delay(10);  
    }  
  
    break;  
}  
  
else{ //종이컵으로 인식  
  
    bluetooth.println("P");  
    delay(2000);  
  
    for(int i=0;i<90;i++){ //종이컵 개폐문 닫힘  
        ang_paper = ang_paper+5;  
        if(ang_paper>=90)  
            ang_paper = 90;  
  
        servo_paper.write(ang_paper);  
        delay(10);  
    }  
}
```

- 텀블러로 인식하는 것은 현재무게-이전무게 값이 3이상으로 많이 차이가 날 경우 사용자가 종이컵 보다 무거운 텀블러를 올린 것으로 판단한다.
- 텀블러나 종이컵을 인식하였을 때 사용자에게 각각 'T', 'P' 문자를 블루투스 통신으로 보내고 음료컵을 인지하기위해 열어 둔 자판기 개폐문(서보 모터)을 닫는다.
- 종이컵의 경우에는 종이컵 개폐문이 열렸을 가능성을 고려해 종이컵 개폐문까지 닫아준다.
- 무게는 대기시간 30초동안인 while문 안에서 항상 감지하도록 하고 이전 값과 현재 값을 통해 텀블러와 종이컵을 구분한다.

구현 기능

실시간 통신

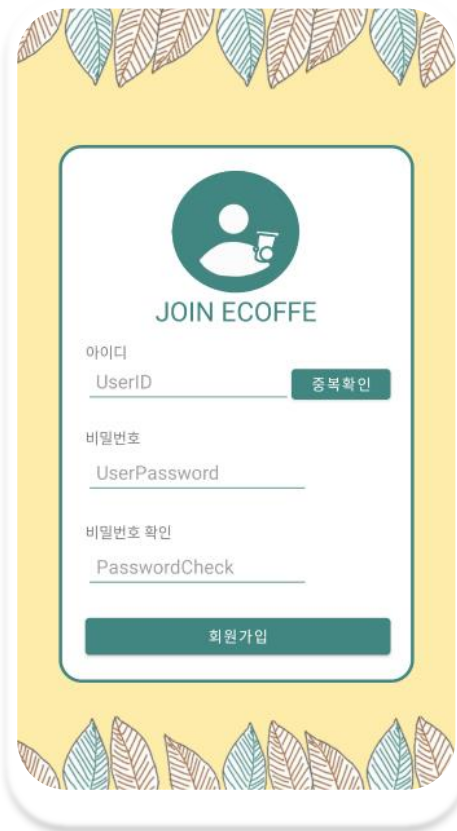
Arduino _ 로드셀 무게센서를 사용한 음료 컵 유무 체크코드

```
while(abs(drink_weight)>=abs(exist_weight)){ //현재 무게가 음료무게보다 작을때까지 켜다.  
    exist_weight = scale.get_units()*10;  
  
    drink_weight = scale.get_units()*10;//음료 무게를 켜다.  
    exist_weight =drink_weight;  
  
}  
//while문을 빠져나왔을 경우  
delay(9000);  
    for(int i=0;i<90;i++){ //자판기 개폐문 닫힘  
        ang_door = ang_door+5;  
        if(ang_door>=90)  
            ang_door = 90;  
  
        servo_door.write(ang_door);  
        delay(10);  
    }  
    for(int i=0;i<90;i++){ //종이컵 개폐문 닫힘  
        ang_paper = ang_paper+5;  
        if(ang_paper>=90)  
            ang_paper = 90;  
  
        servo_paper.write(ang_paper);  
        delay(10);  
    }
```

- 음료가 다 완성되고 난 후, 문이 열리며 무게센서는 추출이 다 된 음료의 무게를 측정한다
- 측정된 무게(drink_weight)와 현재 무게(exist_weight)의 절대값을 비교하여 현재 값이 더 작을 경우 무게측정을 그만한다
- 무게 측정을 그만한다면 사용자가 음료를 무사히 가져간 것으로 보아 개폐문을 닫는다.

구현 기능

그 외 기능



JOIN ECOFFE

아이디
UserID 중복확인

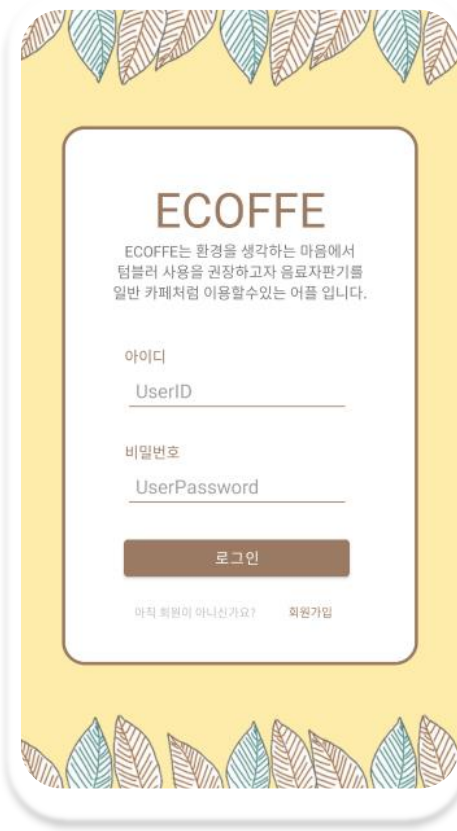
비밀번호
UserPassword

비밀번호 확인
PasswordCheck

회원가입

회원가입

사용자가 회원정보를 입력한 뒤, DB 확인 후 ECOFFEE회원가입이 완료



ECOFFE

ECOFFE는 환경을 생각하는 마음에서 텀블러 사용을 권장하고자 음료자판기를 일반 카페처럼 이용하실수있는 어플 입니다.

아이디
UserID

비밀번호
UserPassword

로그인

마지막 회원이 아니신가요? 회원가입

로그인

회원가입이 되어있는 ECOFFEE회원의 경우 로그인 을 통해 ECOFFEE서비스를 이용



hyun 님, 환영합니다

ECOFFE

5000원

주문하기 충전하기

스탬프 3/8

쿠폰 x2

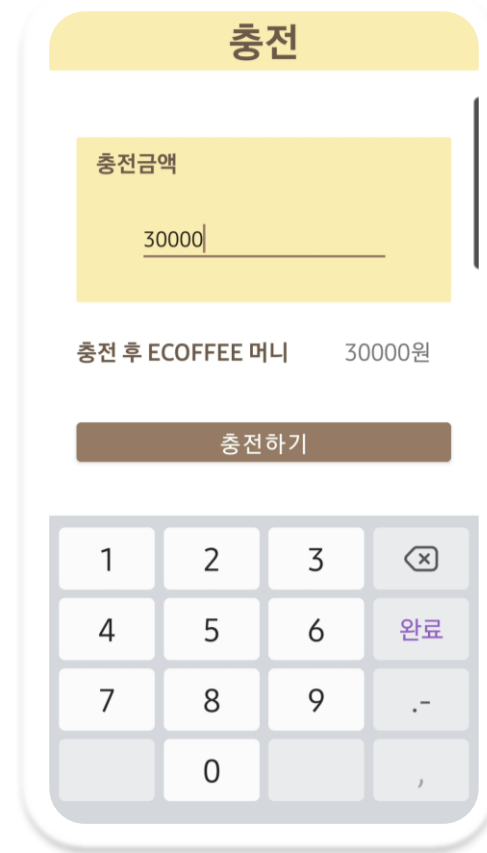
아메리카노 교환권

본 쿠폰은 텀블러만 사용이 가능합니다.

FREE

멤버십 기능

ECOFFEE어플의 서비스를 이용
사용자의 충전된 ECOFFEE머니와
스탬프, 쿠폰을 확인



충전

충전금액
30000

충전 후 ECOFFEE 머니 30000원

충전하기

| | | | |
|---|---|---|----|
| 1 | 2 | 3 | ✕ |
| 4 | 5 | 6 | 완료 |
| 7 | 8 | 9 | . |
| | 0 | | , |

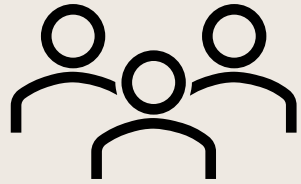
충전하기

사용자가 음료를 주문하기
위해서 필요한 화폐인
ECOFFEE머니를 충전

시연영상

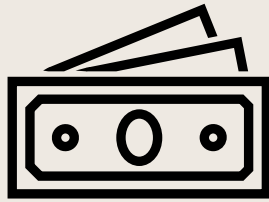


가치 및 기대효과



멤버십 서비스

텀블러 사용시의 혜택과 같은 소비자를 위한 서비스를 제공하여 사용자 편의가 증가함



무인 카페의 저렴한

인건비 부담이 적어 커피의 가격을 낮출 수 있던 무인 카페의 장점을 그대로 가져와 가격은 저렴함



종이컵 사용률 감소

종이컵의 경우에는 '돈을 더 많이 소비 한다'라는 인식을 주어 사용자의 종이컵 사용률을 감소

빨대나 홀더를 아예 제공하지 않음으로써 일회용품 사용을 최소화



텀블러 사용 권장

텀블러를 이용할 시 더 큰 혜택을 많이 주는 것으로 사용자에게 텀블러 사용을 유도

가치 및 기대효과

- ✓ 무인 카페의 편리성과 합리적인 가격, 멤버십혜택은 **사용자의 이용률을 더욱 증가시킴**
- ✓ 기계의 인지능력과 작동원리로 인해 무분별한 카페 일회용품 사용을 제한 및 텀블러 사용 증가시킴
- ✓ 커피산업에 따라 함께 **증가하는 일회용품들의 폐기처리비용과 처리과정에서 발생하는 제2환경오염 감소시킴**
- ✓ 기존 자판기에 적용하여 사용중인 자판기 뿐만 아니라 버려진 자판기를 **재활용**할 수 있어 폐기시 유출될 **환경오염물질들을 감소시킴**

➔ **사람들의 커피 수요와 환경보호를 모두 충족시킴**

활용방안

기존 무인 카페 적용 가능



현재 성장하고 있는 기존 무인 카페에
ECOFFEE 서비스를 적용한다면
더 친환경적이고 스마트한 무인 카페가
될 수 있어 일반 카페보다
더 많은 성장률을 기대 가능

이벤트성 환경보호 캠페인 추진



매년 특정 기간 동안
텀블러 권장 캠페인을 진행하여
ECOFFEE의 이용객을 늘려
수익성과 환경 보호에 대한
공공의 이익을 추구

다른 자판기의 대체 가능



기업에서 편의 시설로 사용되는
기존 커피 자판기를 ECOFFEE로 대체해
환경을 위한 실천을 행하고 있는
기업의 이미지에 도움됨

Thank you

GtiHub : <https://github.com/EcoffeeProject>

