<QR코드현장결제시스템>

Software Requirements Specification

For < 서브 백엔드 개발자 >

Version <1.0>

2024.05.17

팀 명 - 19Plus

조 장 - 이 정 민

조 원 - 이 석 재

오 자 현

임 성 준

지도교수 - 이 귀 봉 교수님

바쁜 현대사회에서 살아가는 우리에게는 시간이 중요합니다. 그래서 소요시간을 줄여주는 것은 정말 우리에게 필요한 부분 중 하나 입니다. 현재 계산대에서 결제하는 방식은 줄을 서서 기다려야 해서 어느 정도의 소요시간이 발생할 수 있습니다. 이러한 문제점을 해결하고자 QR코드 현장 결제 시스템을 개발하기로 하였습니다.

QR코드 현장 결제 시스템은 큰 기능으로 상품의 QR코드를 스캔 하여 장바구니에 담는 기능, 간편 결제 기능, 회원에게 상품을 추천하는 기능, 상품 리뷰 평점 기능 등이 있습니다. 추가로 상품정보를 스캔 하지 않고도 애플리케이션에서 확인할 수 있는 페이지도 있습니다.

사용방법은 구매를 희망하는 상품을 애플리케이션의 카메라 기능을 통해 QR스캔을 진행하면 장바구니에 담기고 최종 결제를 진행할 수 있습니다.

이처럼 계산대에 줄을 서지 않고도 간편하게 결제를 할 수 있기 때문에 QR코드 현장 결제 시스템은 바쁜 현대사회인들에게 안성맞춤 애플리케이션입니다.

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| **<2024/03/26>** | **<1.0>** | **QR현장결제시스템 문서 초안** | **임성준** |
| **<2024/03/26>** | **<1.0>** | **자료조사 및 참고문헌 작성** | **오자현** |
| **<2024/03/27>** | **<1.0>** | **문서 오탈자 수정** | **이석재** |
| **<2024/03/27>** | **<1.0>** | **문서 검토** | **이정민** |
| **<2024/04/04>** | **<1.0>** | **문서 1차 수정** | **임성준** |
| **<2024/04/05>** | **<1.0>** | **문서 오탈자 수정** | **이석재** |
| **<2024/04/05>** | **<1.0>** | **문서 검토** | **이정민** |
| **<2024/04/16>** | **<1.0>** | **문서 2차 수정** | **임성준** |
| **<2024/05/16>** | **<1.0>** | **문서3 차 수정** | **임성준** |
| **<2024/05/17>** | **<1.0>** | **개인문서작성** | **오자현** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. 서론 4

1.1 애플리케이션 개발 개요 및 목적 4

1.2 정의 4

1.3 사용범위 4

1.4 비슷한 사용 사례 4

2. 프로젝트 4

2.1 애플리케이션 설명 4

2.2 개발환경 및 기술 스택 5

2.3 담당업무 5

2.4 애플리케이션 흐름도 7

2.5 UI 구조 7

2.5.1 사용자 UI 7

2.5.2 관리자 UI 7

2.6 DB 구조 10

2.6.1 상품테이블 10

2.6.2 카운터테이블 10

2.7 기능 설명 10

2.8 개발일정 10

2.9 프로젝트 진행상황 11

2.9.1 이정민 11

2.9.2 임성준 11

2.9.3 이석재 11

2.9.4 오자현 11

3. 결론 11

4. 참고문헌 11

4.1 참고문헌 11

# 서론

## 애플리케이션 개발 개요 및 목적

* 바쁜 현대사회에서 살아가는 우리에게는 시간이 중요합니다. 그래서 소요시간을 줄여주는 것은 정말 우리에게 필요한 부분입니다. 현재 계산대에서 결제하는 방식은 줄을 서서 기다려야 해서 어느 정도의 소요시간이 발생할 수 있습니다. 이러한 문제점을 해결하고자 저희의 목적은 고객이 직접 현장에서 바로 결제할 수 있는 QR코드 현장 결제 시스템 애플리케이션 개발입니다.

## 정의

* 등록된 상품의 정보를 볼 수 있고 매장에서 QR코드를 활용하여 상품 결제가 가능한 QR코드 현장 결제 시스템 애플리케이션 입니다.

## 사용범위

* 제품에 저희가 생성한 QR코드를 스티커에 담고 붙여서 판매 가능한 상품이면서 가격대가 낮은 매장에서 사용가능 합니다. 예를 들면 다이소, 편의점 등이 있습니다.

## 비슷한 사용 사례

* 현재 다이소, 편의점 등 셀프 계산대를 사용하는 매장들이 있고 대표적으로 PAYBOOC이 있습니다.

# 프로젝트

## 애플리케이션 설명

* QR코드 현장 결제 시스템의 범위는 고객의 입장에서는 애플리케이션에서 고객의 위치로부터 가장 가까운 매장의 위치정보를 알 수 있으며 매장에 있는 상품의 정보와 재고현황을 알 수 있습니다.
* 또한 애플리케이션의 결제 메뉴를 통해 상품을 스캔 하면 상품이 장바구니에 담기고 간단한 상품정보를 볼 수 있습니다. 쉽게 말해, 화면의 절반은 QR을 스캔 할 수 있는 카메라화면이고 나머지 절반은 현재 사용중인 결제시스템인 포스기 입니다. 결제된 후에는 My Page 에서 최종 결제내역까지 볼 수 있는 범위까지 개발할 예정입니다.
* 사용자는 회원과 비회원으로 구분하고 비회원일 경우 임의의 ID를 발급하여 애플리케이션을 이용할 수 있습니다. 비회원과 회원의 차이점은 회원가입을 하고 결제를 하면 포인트 적립 기능을 통해 자동으로 포인트가 적립되어 후에 결제를 할 때 포인트를 사용할 수 있습니다. 하지만 비회원일 경우 포인트 적립을 할 수 없습니다. 추가로 비회원일 경우 평점을 남길 수 없습니다.
* 관리자의 입장에서는 회원의 탈퇴 처리가 가능하고 최근 애플리케이션에서 결제된 내역 전체를 볼 수 있고 결제 취소 요청이 들어오면 회원ID를 검색하여 해당 결제를 취소할 수 있습니다. 또한 상품데이터를 등록, 수정, 삭제가 가능하여 상품을 관리할 수 있고 추가 관리자를 등록할 수 있게 하여 규모가 커졌을 경우에도 관리가 어렵지 않게 추가 관리자 등록 기능을 구현하는 범위까지 개발할 예정입니다.

## 개발환경 및 기술 스택

* 개발 환경 : Android Studio
* 기술 스택 : Java, Figma, Photoshop, Firebase, SQLite
* SDK 최소 버전 : 33

## 담당업무

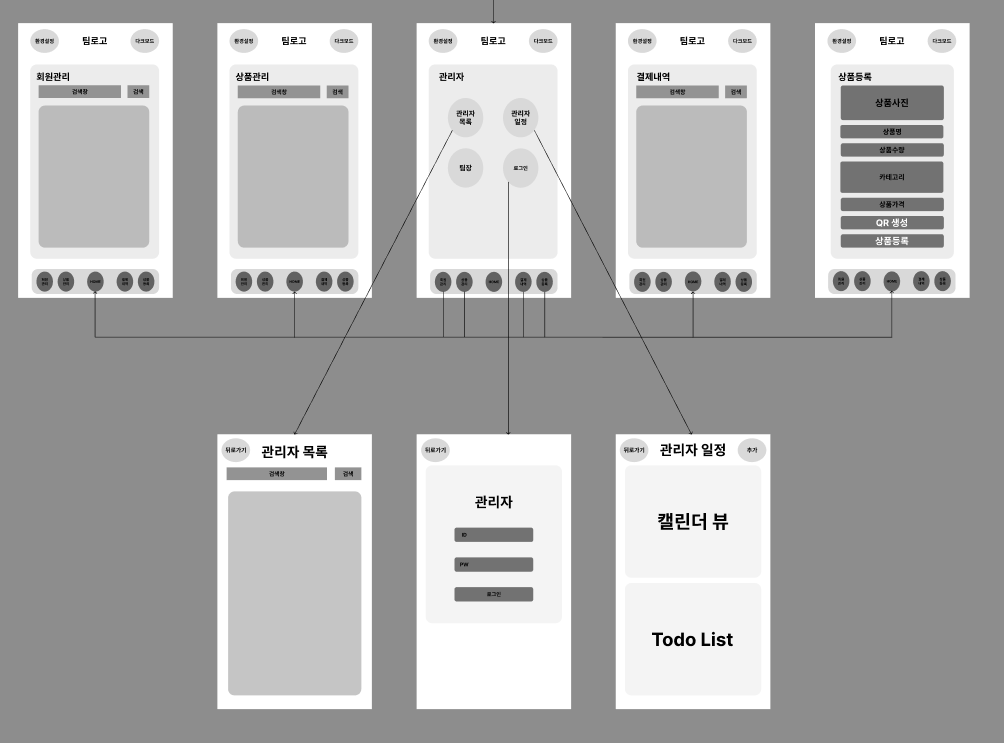
* 이정민 (팀장 및 프론트엔드 개발자)
  + Android Application UI 설계 및 디자인, 구현  
    - 홈 페이지  
    - 회원정보 페이지  
    - 회원가입 페이지  
    - 상품 페이지   
    - 결제 성공/실패 및 에러 페이지s
  + 발표 자료 준비
* 임성준 (시나리오 설계 및 프론트엔드 개발자)
  + 시나리오 수립
  + Android Application UI 설계 및 디자인, 구현   
    - 관리자 메인 페이지  
    - 상품관리 페이지  
    - 상품등록 폼 페이지  
    - 회원관리 페이지   
    - 관리자 로그인 페이지   
    - 관리자 환경설정 페이지   
    - 관리자 Todo 페이지  
    - 추가 관리자 관리 페이지  
    - 추가관리자 등록 페이지  
    - 결제 내역 페이지   
    - 매장 위치 검색 페이지   
    - 결제 페이지(장바구니)  
    - 상품 리뷰 작성 및 평점 페이지
  + 상품 리뷰 평점 기능 구현
  + 매장 위치 검색 기능 구현
  + 관리자 다크모드 기능 구현
  + 관리자 모달창 기능 구현
  + 관리자 Todo 추가, 삭제, 조회 기능 구현
  + 관리자 일정 데이터베이스 구축
  + 코드 리팩터링 및 주석 정리
  + 발표 자료 준비
  + 전체 문서 작업
* 이석재 (메인 백엔드 개발자)
  + 회원가입 및 로그인 기능 구현
  + 사용자 데이터베이스 구축
  + 상품 결제 기능 구현
  + 포인트 적립/차감 기능 구현
  + 결제 내역 데이터베이스 구축
  + 회원관리 페이지 기능 구현
  + 회원 탈퇴 기능 구현
  + 회원관리 검색 기능 구현
  + 결제 내역 및 결제 취소 기능 구현
  + 최초 관리자 및 추가 관리자 등록, 삭제 기능 구현
  + 발표 자료 준비
* 오자현 (서브 백엔드 개발자)
  + QR코드 생성 및 스캔 기능 구현
  + 장바구니 기능 구현(장바구니 추가 및 삭제)
  + 상품 데이터베이스 구축
  + 상품 관리 기능 구현(상품 등록, 수정, 삭제)
  + 상품관리 검색기능 구현
  + 상품페이지 기능 구현
  + 발표 자료 준비

## 애플리케이션 흐름도

## UI 구조

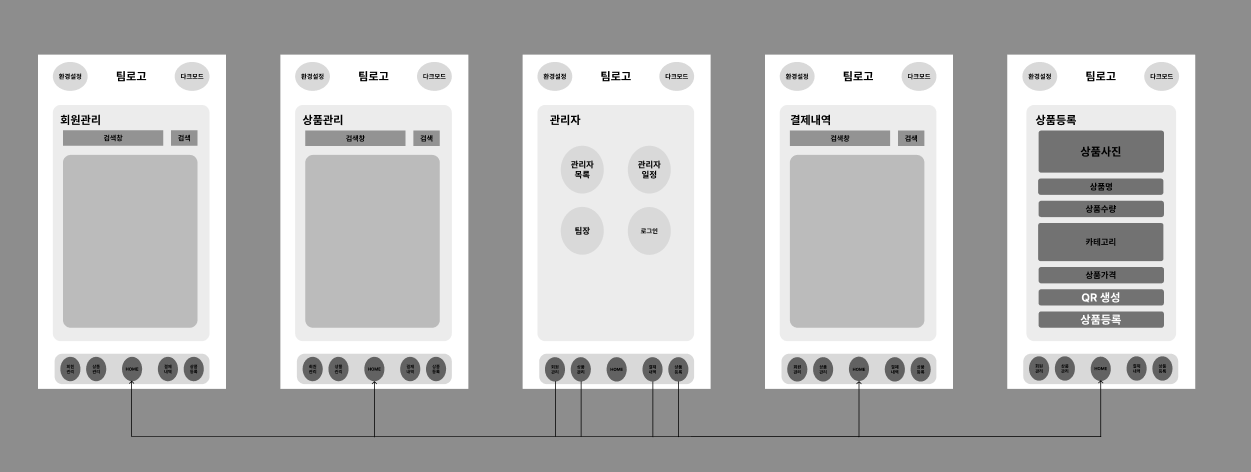
### 사용자 UI

### 관리자 UI

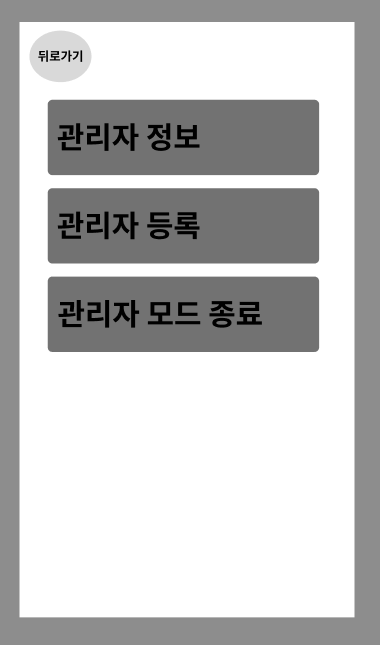


전체적인 관리자 UI 설계도 입니다. ViewFlipper를 사용해서 상단 좌측 페이지부터 인덱스가 1, 2, 0, 3, 4 입니다. 가운데 페이지가 관리자 메인 페이지이고 인덱스를 0으로 지정해서 좌 -> 우 방향이 아닌 중앙에서 양측으로 이동하게 구현하였습니다. 아래 3개의 페이지는 중앙페이지인 관리자 메인 페이지의 3개의 버튼을 클릭해서 넘어가도록 구현했습니다.

#### 관리자 메인 구조



* 관리자 메인
  + 상단
    - 상단은 환경설정 버튼과 팀 로고, 다크 모드 버튼이 있습니다.
      * 다 크모드 : 기본은 라이트 모드입니다. 다크모드 버튼을 클릭하면 페이지 UI의 전체 색상이 어둡게 적용되도록 설정하였습니다.
      * 환경설정
        + 환경설정 버튼은 클릭 시 환경설정 페이지로 넘어가게 되어있습니다. 환경설정 페이지에서는 현재 관리자 정보, 추가 관리자 등록 페이지, 관리자 모드 종료 메뉴 총 3가지의 메뉴가 있습니다. 관리자 정보는 간단한 알림 창으로 볼 수 있게 구현했고 관리자 등록은 등록 버튼 클릭 시 등록 페이지로 넘어가게 구현하였습니다. 관리자가 추가되었을 시 여기서 관리자를 추가하면 됩니다. 관리자 모드 종료는 버튼 클릭 시 종료되게 구현하였습니다. 아래는 관리자 환경설정 페이지와 관리자 등록페이지 입니다.

* + 중앙
    - 회원관리
      * 회원관리 페이지는 회원ID를 검색할 수 있는 기능이 있습니다. 검색을 통해 해당 회원을 찾을 수 있습니다. 해당 회원을 비활성화할 수 있는 기능도 있습니다. 회원은 ListView를 통해 목록 형식으로 볼 수 있습니다.
    - 상품관리
      * 상품관리 페이지는 상품을 검색할 수 있는 기능이 있습니다. 검색을 통해 해당 상품을 찾을 수 있습니다. 여기서는 해당 상품을 수정할 수 있고 삭제할 수 있습니다. 수정은 수정 버튼을 통해 수정할 수 있는 페이지로 넘어갑니다. 상품목록에서는 간단한 정보만 보여주고 자세한 정보를 보려면 해당 상품을 클릭하면 알림 창으로 볼 수 있게 구현하였습니다. 삭제도 한번 더 확인할 수 있게 알림 창이 뜨도록 구현하였습니다.
    - 결제내역
      * 결제내역 페이지는 최근 결제된 내역부터 뜨도록 구현하였습니다. 여기도 검색기능을 통해 회원ID를 검색해서 해당 회원의 결제내역을 알 수 있게 했습니다. 추후 회원이 매장을 찾아와 반품요청을 할 경우 제품 상태 확인 후 결제 취소 버튼을 통해 반품처리 되도록 구현하였습니다.
    - 상품등록
      * 상품등록 페이지는 관리자가 새로운 상품을 등록할 때 사용할 수 있습니다. 상품정보를 입력 후 QR생성 버튼을 통해 상품QR을 생성한 후 등록을 해서 상품이 등록되도록 구현하였습니다.

#### 관리자 서브 구조



관리자 메인 페이지에서 넘어가는 페이지들 입니다. 각각 좌측부터 관리자 목록을 볼 수 있는 관리자 메인 페이지, 관리자 로그인 페이지, 관리자 일정 페이지 입니다.

* 관리자 목록 페이지
  + 관리자 목록 페이지는 검색과 비활성화만 가능합니다. 검색을 통해서 해당 관리자를 찾을 수 있고 비활성화를 통해 해당 관리자를 비활성화 시킬 수 있습니다.
* 관리자 로그인 페이지
  + 관리자 로그인 페이지에서는 관리자로 로그인할 수 있습니다. 관리자 모드를 접속할 때 정해진 코드를 입력해서 들어올 수 있는데 추가로 관리자 메인 페이지의 관리자 로그인 버튼을 클릭 후 관리자 로그인을 해야 관리자 기능들을 사용할 수 있게 해서 보안을 강화했습니다.
* 관리자 일정 페이지
  + 관리자 일정 페이지는 CalendarView를 통해 날짜를 확인할 수 있고 상단 우측 추가 버튼을 클릭하면 일정을 추가할 수 있는 창이 뜨도록 구현했습니다. 하단 TodoList는 ListView를 사용해서 관리자가 등록한 일정을 볼 수 있게 했습니다. 그런데 등록된 일정을 전부 볼 수 있는 것이 아니라 비어있는 목록에서 해당 날짜를 선택하면 해당 날짜에 등록된 일정만 볼 수 있게 구현했습니다. 일정 삭제도 가능합니다. 관리자 일정은 간단한 데이터들이고 관리자만 이용할 수 있으므로 Database는 SQLite를 사용하였습니다.

## DB 구조

### Products Table

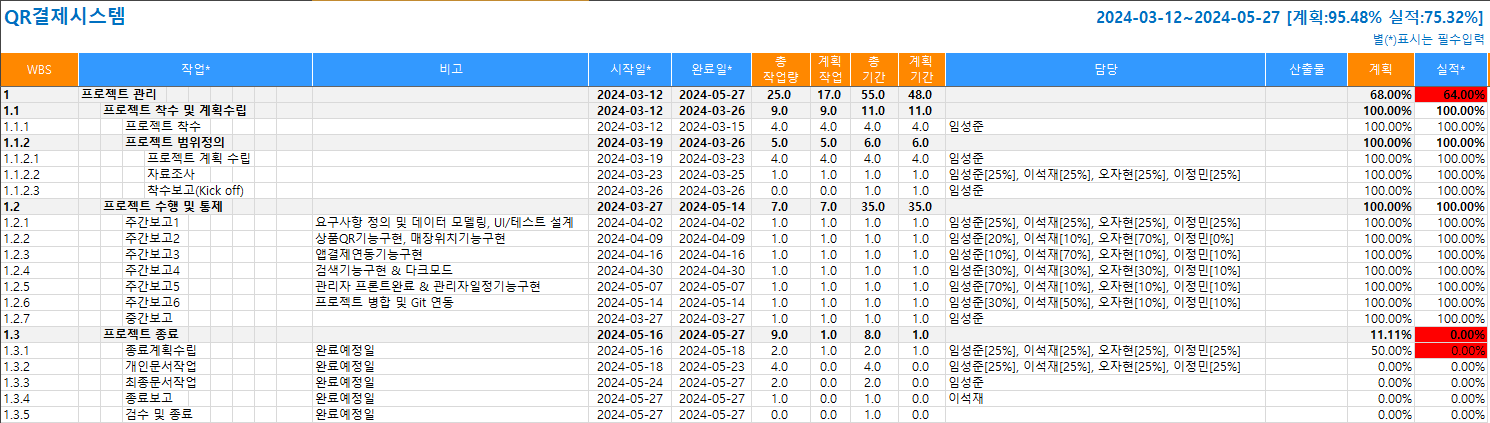
 상품정보를 관리하는 테이블입니다.

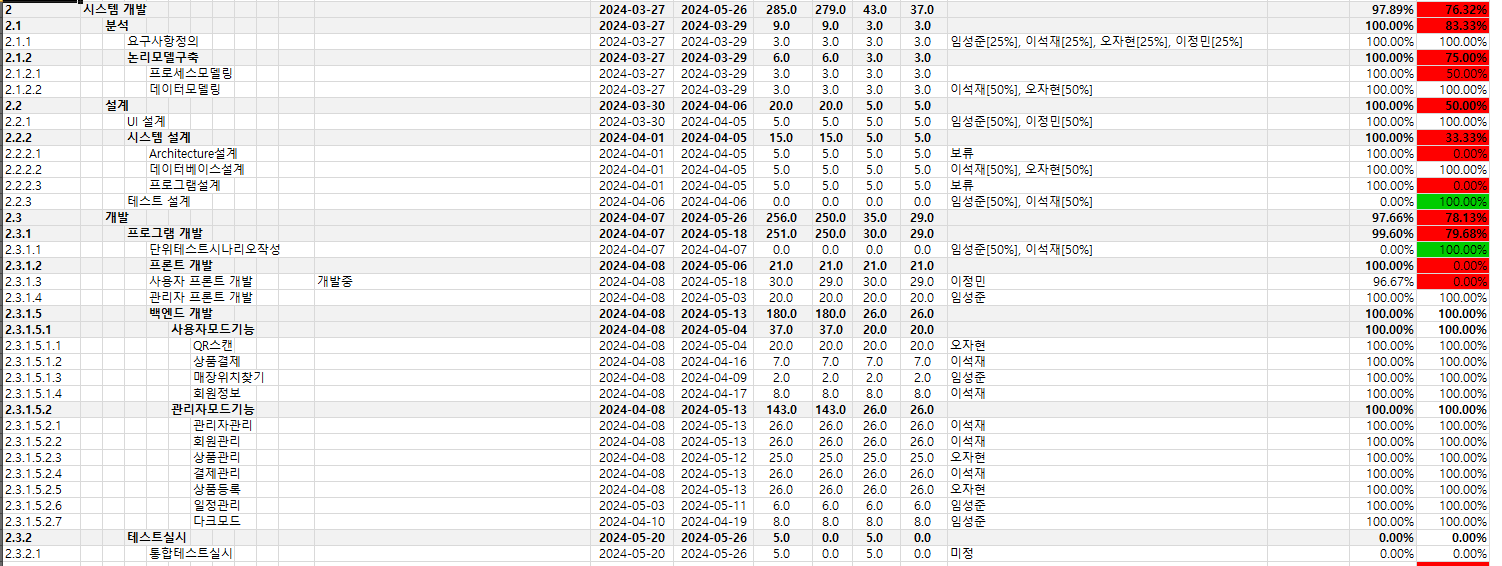
### Counters Table

 상품코드를 관리하는 테이블입니다.

## 기능 설명

## 개발일정





## 프로젝트 진행상황

### 이정민

### 임성준

### 이석재

### 오자현 java, xml파일 첨부할 것

QR스캔기능 구현

설명 - 기능은 안드로이드 애플리케이션에서 QR 코드 스캔 기능입니다. onCreate 메서드에서 DecoratedBarcodeView를 초기화하고, CaptureManager를 통해 인텐트로부터 설정을 초기화하여 QR 코드 스캔을 시작합니다. barcodeScannerView.decodeContinuous를 사용하여 QR 코드를 지속적으로 스캔하고, 스캔된 결과를 로그에 출력한 후 processScannedData 메서드를 호출합니다. 스캔 작업은 capture.onPause로 일시 중지할 수 있습니다. 또한, payButton을 클릭하면 processPayment 메서드를 호출하여 결제 처리를 진행합니다.

QR이미지생성기능 구현

설명 - 이 기능은 Firestore에서 lastProductCode 값을 읽어와 QR 코드를 생성하고 저장하는 기능입니다. generateAndDisplayQRCode 메서드는 MultiFormatWriter를 사용해 텍스트를 QR 코드로 변환하고, BarcodeEncoder를 통해 Bitmap 이미지로 변환하여 ImageView에 표시합니다. saveQRCodeToGallery 메서드는 QR 코드를 갤러리에 저장하는 기능을 수행하며, ContentValues를 사용해 이미지의 메타데이터를 설정하고 MediaStore에 이미지를 저장합니다. 저장이 완료되면 사용자에게 QR 코드가 갤러리에 저장되었다는 토스트 메시지를 표시합니다.

~~SQLITE를 이용한 데이터베이스에 상품을 저장하는 기능을 구현~~ 추후 firebase로 변경됨

firebase를 이용한 데이터베이스 구현 및 상품저장기능 구현

설명 - 이 기능은 사용자가 입력한 상품 정보를 Firebase Storage와 Firestore에 저장하는 과정을 구현한 것입니다. 먼저, uploadFileAndSaveProductInfo 메서드는 사용자로부터 상품 이름, 가격, 재고 수량, 카테고리 및 파일 URI를 입력받아 유효성을 검사합니다. 모든 입력 필드가 채워진 경우, Firebase Storage에 파일을 업로드하고 다운로드 URL을 가져옵니다. 이후, Firestore 트랜잭션을 사용하여 productCounter 문서의 lastProductCode 값을 갱신하고, 새로운 productCode를 생성합니다. (데이터가 저장된 이후 lastProductCode의 값은 1이 추가됩니다.) 생성된 상품 정보는 Firestore의 products 컬렉션에 저장되며, 성공 시 사용자에게 알림 메시지를 표시하고, 작업이 완료됩니다. 만약 업로드나 저장 중 오류가 발생하면 적절한 오류 메시지를 표시하여 사용자에게 알립니다.

상품리스트뷰에 Adapter연결로 상품정보표시 구현

설명 - listView.setAdapter로 아답터를 연결합니다. 이 어댑터는 ListView에서 각 상품의 정보를 표시하는 역할을 합니다. getView 메서드에서 ViewHolder 패턴을 사용하여 뷰의 성능을 최적화하고, 재사용 가능한 뷰를 설정합니다. 각 상품의 이름, 가격, 재고 수량 및 카테고리 정보는 TextView에 설정되며, Glide 라이브러리를 통해 상품 이미지를 ImageView에 로드합니다. 이렇게 하면 사용자에게 깔끔하게 상품 정보를 제공할 수 있습니다.

상품리스트뷰에 Adapter연결로 상품별 버튼이벤트 구현

설명 - listView.setAdapter로 아답터를 연결합니다. 각 상품 항목에는 재고를 증가시키거나 감소시키는 두 개의 버튼이 있습니다. '추가' 버튼을 클릭하면 해당 상품의 재고가 증가하고, notifyDataSetChanged 메서드를 호출하여 어댑터에 데이터 변경을 알립니다. '감소' 버튼을 클릭하면 재고가 감소하며, 재고가 1일 경우 사용자에게 아이템 삭제 확인 대화상자를 표시합니다. 사용자가 '네'를 선택하면 해당 상품이 리스트에서 삭제되고, '아니요'를 선택하면 대화상자가 닫힙니다. 이러한 이벤트 처리는 사용자 상호작용을 통해 상품 관리 기능을 구현합니다.

데이터 입력 시 이미지를 함께 입력하고, URL을 이미지로 변환하는 기능을 Glide 라이브러리를 사용하여 구현

설명 - 이 기능은 사용자가 입력한 상품 정보와 함께 상품 이미지를 Firebase Storage에 업로드하고, Firestore에 저장된 이미지 URL을 Glide 라이브러리를 사용하여 로드합니다. 먼저, 사용자로부터 입력받은 상품 이름, 가격, 재고 수량, 카테고리 및 이미지 파일 URI를 검증합니다. 이후, Firebase Storage에 이미지를 업로드하고, 업로드된 이미지의 다운로드 URL을 가져옵니다. Firestore 트랜잭션을 사용하여 상품 코드를 업데이트하고, 새로운 상품 정보를 Firestore에 저장합니다. 마지막으로, 저장된 이미지 URL을 Glide 라이브러리를 통해 ImageView에 로드하여 사용자에게 상품 이미지를 표시합니다. 이 과정은 사용자에게 통합된 데이터 입력 및 이미지 관리를 제공하며, 사용자가 쉽게 상품 정보를 관리할 수 있도록 돕습니다.

장바구니의 총 금액계산 후 결제 시 테이블 데이터 수정 구현

QR스캔으로 상품정보 읽기 구현

Products Table에 카테고리 추가

# 결론

# 참고문헌

## 참고문헌

* [BootPay](https://docs.bootpay.co.kr/?front=web&backend=curl#introduce)
* [Zxing 라이브러리](https://github.com/zxing/zxing)
* [Firebase](https://firebase.google.com/docs?hl=ko&_gl=1*w086pj*_up*MQ..*_ga*NDMxNDA0OTgxLjE3MTIyMzg5MDk.*_ga_CW55HF8NVT*MTcxMjIzODkwOC4xLjAuMTcxMjIzODkwOC4wLjAuMA..)
* [글래스모피즘](https://www.pinterest.co.kr/choi1698096/ê¸ëì¤ëª¨í¼ì¦/)
* [Flexbox-layout](https://github.com/google/flexbox-layout)
* [뉴모피즘](https://github.com/fornewid/Neumorphism?source=post_page-----3a298e0337b0--------------------------------)
* [glide](https://github.com/bumptech/glide)