

제1회 경남소프트웨어경진대회

[공공데이터를 활용한 의약품 정보 기반 복약 관리 서비스] 완료 보고서

소속 : [BCD]

버전번호 : [Version 1.0]

제출일자 : [2021-10-15]

제. 개정내역

버전/릴리스	작성일자	개요	
1.0	2021년 10월 15일	신규 제정	박수아

대체 인도물

본 문서는 다음의 인도물을 대체한다.

문서번호	버전/릴리스	내용

*해당사항 없음

목 차

1. 개발 개요.....	5
1.1 개발 명.....	5
1.2 개발 기간.....	5
1.3 개발 목적.....	5
1.4 개발 범위.....	5
1.5 개발 목표.....	6
1.6 개발 기대효과	6
2. 배경 지식	6
2.1 DUR	6
3. 연구 동향	6
3.1 개발연구 동향	6
3.2 유사 서비스와 비교	7
3.2.1 만약	7
3.2.2 헬피	8
4. 개발 수행 조직.....	9
4.1 조직도.....	9
4.2 책임과 역할	9
5. 개발환경	10
5.1 개발환경.....	10
6. 개발 일정	10
6.1 개발일정 및 투입인원	10
7. 형상관리 계획.....	10
7.1 형상관리 보관 장소	10
7.2 백업관리.....	11
8. 의사소통관리 계획	11
9. PA활동	11
10. 테스트 계획.....	12
11. 설계 및 구현.....	12
11.1 시스템 설계.....	12

11.2 구성요소별 구현 기능	13
11.2.1 데이터베이스(SQLite)생성 및 연결하기	13
11.2.2 Open API 데이터 가져오기	14
11.2.3 Cloud Storage for Firebase에서 이미지 가져오기	16
11.2.4 Firebase Realtime Database에서 데이터 가져오기	17
11.2.5 일반의약품 상세 정보 검색	18
11.2.6 복약 기록	19
11.2.7 복용 의약품에 따른 맞춤형 주의사항 제공	20
11.2.8 복약 시각 알림	21
11.2.9 의약품 즐겨찾기	22
12. 구현결과	23
12.1 로딩 화면	23
12.2 초기 정보 설정 화면	23
12.3 복약 관리 화면	24
12.4 복약 기록 리스트에 따른 맞춤형 주의사항 제공 관련 화면	24
12.5 회원 정보 화면	25
12.6 복약 시각 알림 화면	25
12.7 일반의약품 검색 화면	26
12.8 검색 결과 리스트 화면	26
12.9 일반의약품 상세 정보 화면	27
12.10 즐겨찾기 화면	27
13. 결론	28
13.1 결론 및 기대 효과	28
13.2 프로젝트를 마치며	28
14. 최종 산출물	31

1. 개발 개요

1.1 개발 명

제1회 경남소프트웨어경진대회 [공공데이터를 활용한 의약품 정보 기반 복약 관리 서비스]
(이하 “프로그램 개발”이라 칭한다.)

1.2 개발 기간

2021년 09월 16일 ~ 2021년 10월 15일

1.3 개발 목적

- 1) 공공데이터포털에서 제공되는 의약품 정보와 DUR 정보를 가공하여 종합적인 의약품 정보의 접근성 강화
- 2) 복용량 및 복용 시각 알림 등 사용자 맞춤형 복약 관리 서비스 제공
- 3) 복용 시 따르는 부작용 속지 및 방지

1.4 개발 범위

- 1) 개발 명: [공공데이터를 활용한 의약품 정보 기반 복약 관리 서비스]
- 2) 의약품 상세 정보 검색
 - 의약품의 제품명 및 성분명, 약모양으로 의약품 정보 검색
 - 제품명, 성분정보, 효능효과, 용법용량, 사용상 주의사항, 식별정보, DUR 등 제공
- 3) 알약 이미지 검색
 - 제형, 모양, 색상, 분할선의 버튼 선택으로 조건을 걸어 검색
- 4) 복약 기록
 - 달력을 이용한 복약 기록 기능
 - 제품명, 복약 시작 날짜, 종료 날짜, 복용 시각 설정 시 복약 기록에 추가됨
- 5) 복용 의약품에 따른 맞춤형 주의사항 정보 제공
 - 복용 의약품에 대한 병용 금기, 부작용 등에 대한 주의를 줌
 - 사용자의 특이사항에 관련한 의약품에 대해 주의를 줌
- 6) 복약 시각 알림
 - 복약 기록 리스트에 추가된 항목들에 대해 알림을 줌
- 7) 의약품 즐겨찾기
 - 자주 찾는 의약품 정보에 대한 즐겨찾기를 할 수 있는 기능

1.5 개발 목표

구분	항목	목표치	비고(추가설명, 사유 등)
일정	납기 준수	21.10.15	납기 지연율 0% 유지
비용	투입 공수	[30] D	D = Day
품질	ST 제품결함율	90 % ~ 120 %	제품결함율=BUG적출건수/BUG예상건수
	시스템 성능		

1.6 개발 기대효과

- 1) 사용자 맞춤형 주의사항을 제공하여 쉽고 간단하게 복약 관리 가능
- 2) 사용자의 건강상태에 따른 주의사항 정보와 DUR 정보를 제공하여 복약 시 주의사항 숙지 가능
- 3) 일반의약품에 복약 시 따르는 위험성 감소
- 4) 사용하기 쉬운 직관적인 인터페이스로 모든 연령층이 이용할 수 있음

2. 배경 지식

2.1 DUR

DUR(Drug Utilization Review)은 병용 시 또는 어린이, 노인, 임부에게 투여 시 주의해야 하는 의약품 정보 등을 알리고, 정해진 기준에 따라 약물 사용이 적절하게 이뤄지는지 점검하고 평가하는 제도이다. DUR의 주요 목적은 예방할 수 있는 부적절한 약물 사용을 미연에 방지함으로써 부작용을 예방하고 환자에게 제공하는 의료서비스의 질을 향상시키며 의약품을 안전하게 사용할 수 있는 환경을 조성하는 데 있다. 우리나라의 DUR 정보 제고 현황을 2020년 12월 28일을 기준으로 병용 금기, 특정연령대금기, 임부 금기, 용량 주의, 투여 기간 주의 노인 주의 등이 있다.

3. 연구 동향

3.1 개발연구 동향

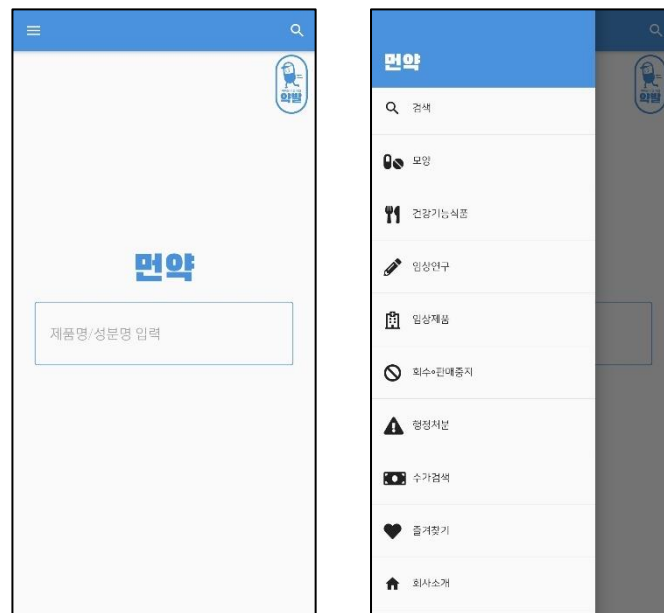
의약품을 검색할 수 있는 서비스로는 ‘약학정보원 공식 사이트’, ‘먼약’, ‘헬피’, ‘파프리카케어’ 애플리케이션 등이 있다. 이 서비스들을 이용하면 의약품의 이름 및 성분으로 의약품 정보를 얻을 수 있다.

의약품 복용을 관리할 수 있는 서비스로는 ‘헬피’, ‘파프리카케어’ 애플리케이션 등이 있다. 이들은 처방전 기반으로 복약 설정을 하거나 복용 중인 의약품을 직접 작성하여 복약 설정을 할 수 있다.

현재까지 나온 애플리케이션이나 웹 사이트는 의약품 검색 기능만을 제공하거나 검색을 기반으로 의약품을 관리할 수 있는 기능을 제공하고 있다. 서비스들의 공통점으로 의약품 복용 시 주의해야 할 사항을 알려주는 DUR 정보를 사용자가 쉽게 파악할 수 있게 보여주지 못한다는 점이 있다.

3.2 유사 서비스와 비교

3.2.1 먼약

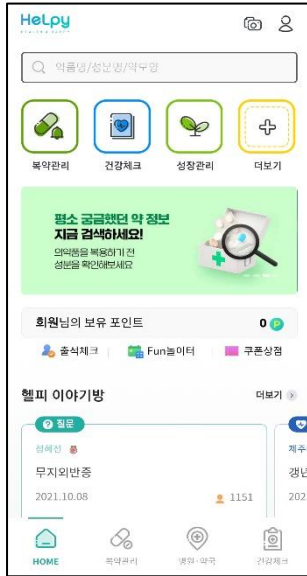


[그림 1] '먼약' 메인 화면

[그림 2] '먼약' 메뉴

'먼약' 애플리케이션은 제품명 및 성분명을 입력하면 의약품 정보를 검색할 수 있다. 또한, 알약 모양으로도 검색할 수 있다. 이후 업데이트를 통해 의약품에 대한 임상 연구를 검색하거나 회수 및 판매 중지된 의약품 정보를 검색할 수 있는 기능이 추가되었다. 사용자의 의약품 복용을 관리할 수 있는 기능은 제공하지 않는다.

3.2.2 헬피



[그림 3] ‘헬피’ 메인 화면



[그림 4] ‘헬피’ 복약 관리 화면

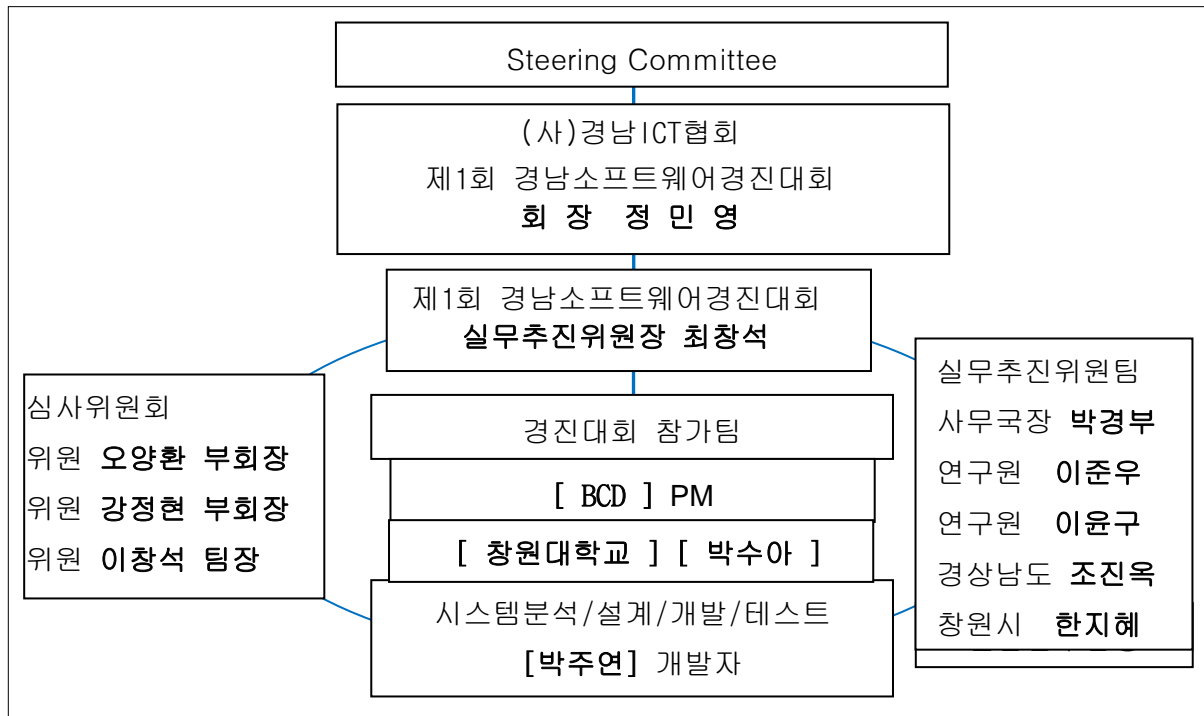


[그림 5] ‘헬피’ 병원 찾기 화면

‘헬피’ 애플리케이션은 제품명 및 성분명을 입력하면 의약품 정보를 검색할 수 있으며, 의약품의 모양으로도 검색이 가능하다. 또한, 처방전 기반의 복약 관리 기능과 약국 검색을 통한 약국 정보를 제공하고 있다. ‘헬피’는 DUR 정보를 제공하고 있지만, 복약 관리 기능에서 DUR 정보를 확인할 때 자세한 정보를 한눈에 파악하기 어렵다. 그리고 의약품 이외의 부수적인 기능들이 많고 수많은 기능이 하나의 화면에 전부 보이기 때문에 애플리케이션을 처음 사용하는 경우 사용하기 복잡하다고 느껴지는 경향이 있다.

4. 개발 수행 조직

4.1 조직도



4.2 책임과 역할

역할	책임	담당자
Steering Committee	<ul style="list-style-type: none"> - 경진대회 추진방향 및 수행 결과 확인 - 운영인력 Resource 지원 	(사)경남ICT협회
실무추진위원회	<ul style="list-style-type: none"> - 일정 관리 - 관리 및 통제 - 심사위원회 운영 - 시상 및 시상식 운영 	실무추진위원회 7개 대학교
개발팀 PM	<ul style="list-style-type: none"> - 프로젝트 진행에 대한 고객 최종 의사 결정권자 - 일정 관리 - 관리 및 통제 - 코딩 및 테스트 	박수아
팀원	<ul style="list-style-type: none"> - 사용자 면담 및 요구사항 분석 - 시스템 처리 설계 - 코딩 및 테스트 - 프로젝트 산출물 작성 및 관리 - 프로젝트 진행 보고 - 요구사항 제출 - 관련자료 정의 	박주연 개발자
실무추진위원	<ul style="list-style-type: none"> - 관련 업무 협조 및 지원 - 이슈 사항 협의 및 지원 	실무추진위원회 경상남도, 창원시






5. 개발환경

5.1 개발환경

구분		Spec.
시스템 환경	OS	Microsoft Windows 10
	CPU	Qualcomm Kryo 280 MP4 1.9 GHz
	Memory	4GB
	Disk	내부: 64GB
	개발방법	JAVA, XML, PHP
	Data Base	SQLite, Firebase
디자인 환경		CLIP STUDIO

6. 개발 일정

6.1 개발일정 및 투입인원

	1W	2W	3W	4W	공수(MM)	비고
착수					2	착수: 2021.09.16.
분석/설계	 분석 Review 설계 Review				2	분석 Review: 2021.09.20. 설계 Review: 2021.09.25.
개발					2	
단위/통합 테스트					2	
Go Live 및 안정화			 Go Live		2	Go Live: 2021.10.13. 완료: 2021.10.15.
참여 인원(명)	2	2	2	2		

[그림 6] 개발일정 및 투입인원

7. 형상관리 계획

7.1 형상관리 보관 장소

- 소스보관: https://github.com/juyeeon/Aigoyak_project.git

7.2 백업관리

백업 대상	주기	시기	담당자	방법(매체)	기준
전자문서	1일	오후6시	박주연	HDD	일일 작업 전체
프로그램 및 디자인 소스	1일	오후6시	박주연	HDD	일일 작업 전체

8. 의사소통관리 계획

수시 소통	보고가 필요하다고 판단되는 안건	팀원	수시	메일	개발팀PM
의사 결정 요청	중대한 의사 결정에 대한 요청	PM	필요 시	메일	개발팀PM
완료	개발 완료 결과 보고	PM	완료 시	완료보고서	개발팀PM

9. PA활동

공정	활동 내용	일정	비고
계획	일정관리표 점검	2021.09.16	
	프로젝트계획서 점검	2021.09.16	
분석	분석서 점검	2021.09.16	
설계	외부설계서 점검	2021.09.20	요구사항정의서 포함
테스트	PT테스트산출물 점검	2021.10.10	버그관리대장 포함
	ST테스트산출물 점검	2021.10.10	버그관리대장 포함

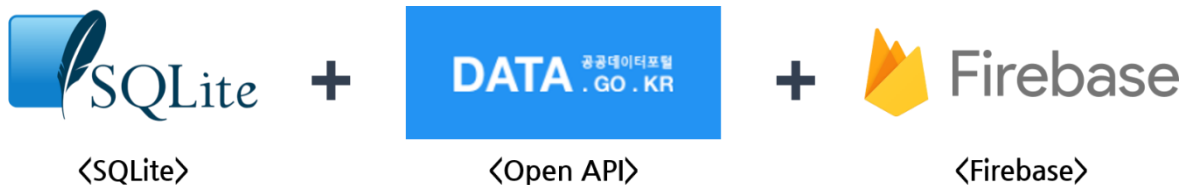
공정	활동 내용	일정	비고
이행	Go-Live점검	2021.10.13	실 운영 서버
완료	운영이관 상황 점검 프로젝트완료보고서 점검	2021.10.15	완료보고서 외 제출서류

10. 테스트 계획

시스템 테스트	실시자		실시시점	테스트기준
	개발자	인수자		
단위테스트	◎		각 모듈에 대한 코딩이 끝난 시점	<ul style="list-style-type: none"> ◆ 각 모듈이 코딩 표준 문서에 따라 작성 여부 ◆ 개발 과정 중 변경된 설계와 수정 보완 여부
통합테스트	◎		단위프로그램 테스트 완료 후 실시	<ul style="list-style-type: none"> ◆ 설계명세서의 모든 요구사항 만족 여부 ◆ 인터페이스 요구사항 확인
인수테스트	◎	◎	시스템 개발 완료 후 실시	<ul style="list-style-type: none"> ◆ 최종적으로 구현된 시스템의 인수 기준 만족 여부

11. 설계 및 구현

11.1 시스템 설계

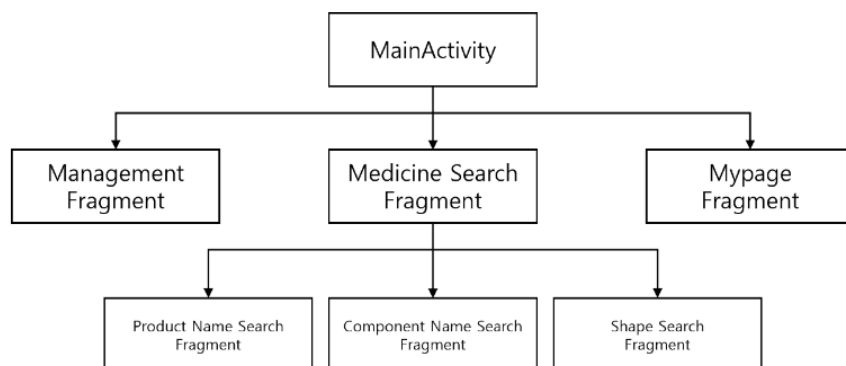


[그림 7] 애플리케이션에 사용된 데이터베이스 및 API

‘아이고약!’은 Android Studio와 SQLite, Firebase를 연결하여 데이터를 받아오고, 공공데이터포털에서 제공하는 Open API를 사용하였다. SQLite 데이터베이스에는 DUR 데이터가 저장되어 있으며 애플리케이션 사용자의 정보를 저장한다. 또한, Firebase의 Realtime Database와 Cloud Storage를 사용한다. Realtime Database에는 약 모양 검색을 위한 데이터와 일반의약품의 성분과 관련된 데이터가 저장되어 있으며, Cloud Storage에는 일반의약품의 이미지 파일이 저장되어 있다.

데이터베이스를 SQLite와 Firebase로 나누어서 사용하는 이유는 SQLite와 Firebase가 다른 특징을 가지고 있기 때문이다. SQLite 데이터베이스는 애플리케이션이 설치되면 같이 다운로드되는 데이터베이스로 데이터에 대한 접근이 빠르지만, 용량이 커지면 애플리케이션의 용량도 같이 커진다는 단점이 있으며 데이터의 수정이 필요한 경우 애플리케이션을 다시 설치해야 한다. 따라서 따로 관리할 필요가 없는 사용자의 정보와 수정되지 않는 DUR 데이터는 SQLite 데이터베이스를 사용하였다. Firebase는 네트워크를 이용하여 데이터를 전송하기 때문에 상대적으로 속도가 느리지만, 데이터의 수정이 필요한 경우 Firebase 콘솔에서 바로 수정 및 적용을 할 수 있다. 또한, 용량이 큰 이미지 파일은 Cloud Storage를 이용하여 이미지 파일 형식 그대로 저장하며 데이터를 받아올 수 있다. Open API로는 일반의약품의 상세정보에 필요한 데이터들을 받아온다. 데이터베이스와 데이터에 관한 자세한 설명은 ‘11.2 구성요소별 구현 기능’에서 다룬다.

11.2 구성요소별 구현 기능



[그림 8] 애플리케이션 구성도

애플리케이션의 기능은 크게 복약 관리 기능, 검색 기능, 사용자 정보 관리 기능으로 나뉜다. MainActivity에 Management, Medicine Search, Mypage가 Fragment로 들어있으며, 각각이 세부적인 기능이 추가되어있다. Management에는 달력을 활용한 복약 기록, 맞춤형 DUR 정보 제공 기능이 있다. Medicine Search에는 제품명 검색(Product Name Search), 성분명 검색(Component Name Search), 약 모양 검색(Shape Search)이 있으며, 검색을 통한 의약품 상세정보 조회가 가능하다. Mypage에는 사용자 정보 조회 및 수정, 즐겨찾기 기능이 있다.

11.2.1 데이터베이스(SQLite)생성 및 연결하기

Android Studio와 SQLite를 연결하여, 사용자의 정보를 저장하는 데이터베이스와 DUR 데이터를 저장하는 데이터베이스를 만들었다.

사용자의 정보를 저장하는 데이터베이스(note.db)는 복약 관리에서 사용자가 추가한 항목을 저장하는 테이블(TABLE_NOTE), 사용자가 즐겨찾기에 추가한 항목을 저장하는 테이블

(TABLE_BOOKMARK), 사용자의 특이사항을 저장하는 테이블(TABLE_USER)로 구성되어 있다. note.db에 테이블을 만들 때는 SQL문을 사용하여 생성하였다.

```
String CREATE_SQL_NOTE = "create table " + TABLE_NOTE + "("
    + " _id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT, "
    + " CODE TEXT DEFAULT '', "
    + " NAME TEXT DEFAULT '', "
    + " CORP TEXT DEFAULT '', "
    + " CLOCK TEXT DEFAULT '', "
    + " DATE INTEGER , "
    + " ALARM INTEGER , "
    + " DATE2 INTEGER);";

String CREATE_SQL_BOOKMARK = "create table " + TABLE_BOOKMARK + "("
    + " _id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT, "
    + " CODE TEXT DEFAULT '', "
    + " NAME TEXT DEFAULT '', "
    + " CORP TEXT DEFAULT '');";

String CREATE_SQL_USER = "create table " + TABLE_USER + "("
    + " _id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT, "
    + " FEATURE TEXT DEFAULT '');";
```

[그림 9] 사용자 정보 데이터베이스(note.db) 구조

DUR 데이터를 저장하는 데이터베이스(dur.db)는 DUR 데이터의 형식이 CSV이므로 'DB Browser(SQLite)' 프로그램을 이용하여 CSV 파일을 import 하는 방법으로 SQLite 데이터베이스 파일을 생성하고, 이 파일을 애플리케이션과 연결하는 방식으로 구현하였다.

DUR 데이터는 공공데이터포털에서 다운로드받았으며, 다운로드받은 데이터를 애플리케이션에서 사용하기 쉽도록 가공하는 작업을 거쳐 사용하였다. DUR 데이터는 DUR의 종류에 따라 테이블이 분리된 형식으로, 특정연령금기(TABLE_AGE), 용량주의(TABLE_AMOUNT), 노인주의(TABLE_OLD_, 투여기간 주의(TABLE_PERIOD), 임부 금기(TABLE_PREG), 병용 금기(TABLE_WITH_ 데이터를 각각 저장하는 6개의 테이블로 구성되어 있다.

이름	타입	스키마
테이블 (6)		
> TABLE_AGE		CREATE TABLE "TABLE_AGE" ("code" TEXT, "limit_nm" TEXT, "unit" TEXT, "standard" TEXT, "content" TEXT, PRIMARY KEY("code"))
> TABLE_AMOUNT		CREATE TABLE "TABLE_AMOUNT" ("code" TEXT, "content" TEXT, PRIMARY KEY("code"))
> TABLE_OLD		CREATE TABLE "TABLE_OLD" ("code" TEXT, "content" TEXT, PRIMARY KEY("code"))
> TABLE_PERIOD		CREATE TABLE "TABLE_PERIOD" ("code" TEXT, "content" TEXT, PRIMARY KEY("code"))
> TABLE_PREG		CREATE TABLE "TABLE_PREG" ("code" TEXT, "content" TEXT, PRIMARY KEY("code"))
> TABLE_WITH		CREATE TABLE "TABLE_WITH" ("code_a" TEXT, "code_b" TEXT, "name_b" TEXT, "content" TEXT)
인덱스 (0)		
뷰 (0)		
트리거 (0)		

[그림 10] DB Browser(SQLite)로 생성된 데이터베이스

11.2.2 Open API 데이터 가져오기

일반의약품이 정보를 가져와서 화면에 출력하기 위해 두 개의 Open API를 사용하였다. 두 개의 Open API의 모두 공공데이터포털에서 활용신청을 하여 사용한 API이다. 제공되는 API의 데이터 포맷은 JSON과 XML 형식이다. 서비스 유형은 REST(GET)로 제공되고 활용한 요청 메시지와 응답 메시지는 다음과 같다.

'식품의약품안전처_의약품개요정보(e약은요)' API 요청 메시지

항목명(영문)	항목명(국문)	항목크기	항목구분	항목설명
serviceKey	인증키	100	1	공공데이터포털에서 발급받은 인증키
pageNo	페이지 번호	5	0	검색할 생성일 범위의 시작
numOfRows	한 페이지 결과수	3	0	검색할 생성일 범위의 종료
itemName	제품명	4000	0	제품명
itemSeq	품목기준코드	4000	0	품목기준코드

'식품의약품안전처_의약품개요정보(e약은요)' API 응답 메시지

항목명(영문)	항목명(국문)	항목크기	항목구분	항목설명
entpName	업체명	4000	0	업체명
itemName	제품명	4000	0	제품명
itemSeq	품목기준코드	4000	0	품목기준코드
efcyQesitm	문항1(효능)	4GB	0	이 약의 효능은 무엇입니까?
useMethodQesitm	문항2(사용법)	4GB	0	이 약은 어떻게 사용합니까?
atpnWarnQesitm	문항3(주의사항 경고)	4GB	0	이 약을 사용하기 전에 반드시 알아야 할 내용은 무엇입니까?
atpnQesitm	문항4(주의사항)	4GB	0	이 약의 사용상 주의사항은 무엇입니까?
intrcQesitm	문항5(상호작용)	4GB	0	이 약을 사용하는 동안 주의해야 할 약 또는 음식은 무엇입니까?
seQesitm	문항6(부작용)	4GB	0	이 약은 어떤 이상반응이 나타날 수 있습니까?
depositMethodQesitm	문항7(보관법)	4GB	0	이 약은 어떻게 보관해야 합니까?
itemImage	날알이미지	이미지	3000	

[그림 11] API 요청 및 응답 메시지1

'의약품 낱알식별정보(DB) 서비스' API 요청 메시지

항목명(영문)	항목명(국문)	항목크기	항목구분	항목설명
ServiceKey	인증키	100	1	공공데이터포털에서 발급받은 인증키
itemSeq	품목기준코드	4000	0	품목기준코드

'의약품 낱알식별정보(DB) 서비스' API 응답 메시지

항목명(영문)	항목명(국문)	항목크기	항목구분	항목설명
CHART	성상	3000	0	성상
DRUG_SHAPE	의약품모양	3000	0	의약품모양
COLOR_CLASS1	색깔(앞)	3000	0	색깔(앞)
COLOR_CLASS2	색깔(뒤)	3000	0	색깔(뒤)
LINE_FRONT	분할선(앞)	3000	0	분할선(앞)
LINE_BACK	분할선(뒤)	3000	0	분할선(뒤)
LENG_LONG	크기(장축)	3000	0	크기(장축)
LENG_SHORT	크기(단축)	3000	0	크기(단축)
THICK	크기(두께)	3000	0	크기(두께)
CLASS_NAME	분류명	3000	0	분류명
ETC_OTC_NAME	전문/일반	3000	0	전문/일반
FORM_CODE_NAME	제형코드이름	3000	0	제형코드이름

[그림 12] API 요청 및 응답 메시지2

이 애플리케이션에서는 API를 XML 문서 형식으로 가져와 사용하는 방식을 사용하였다. 사용한 요청 URL의 형식은 다음과 같다.

```
String address = "http://apis.data.go.kr/1471000/DrbEasyDrugInfoService/getDrbEasyDrugList"
    + "?serviceKey=" + apiKey
    + "&itemSeq=" + sampleCode;

String address2 = "http://apis.data.go.kr/1470000/MdcinGrnIdntfcInfoService/getMdcinGrnIdntfcInfoList"
    + "?ServiceKey=" + apiKey
    + "&item_seq=" + sampleCode;
```

[그림 13] API 요청 URL

Open API의 요청 URL을 API마다 각각 설정하여 저장한다. 위 코드의 요청 URL은 공공데이터포털에서 데이터 활용신청을 하여 발급받은 서비스 키와 의약품의 품목 기준코드를 결합한 것으로, 서비스 키와 품목 기준코드에 모두 일치하는 데이터만 결과로 나타나게 된다. [그림 13]에서 API 2는 ‘식품의약품안전처_의약품개요정보(e약은요)’ API이며, API 3은 ‘의약품 낱알식별정보(DB) 서비스’ API를 말한다.

요청 URL을 사용하여 데이터를 요청하게 되면 API에서 응답 메시지를 반환하게 된다. 이를 사용하기 위해 두 API의 응답 메시지에서 필요한 정보에 맞는 태그의 내용만 저장하여 사용하였다.

11.2.3 Cloud Storage for Firebase에서 이미지 가져오기

의약품의 이미지를 사용하기 위해 Firebase에서 제공하는 Cloud Storage를 사용하였다. 의약품의 품목 기준코드를 파일 이름으로 하여 Cloud Storage의 ‘medicine’ 디렉토리에 저장되어 있으며, 이미지의 형식은 png 형식이다.

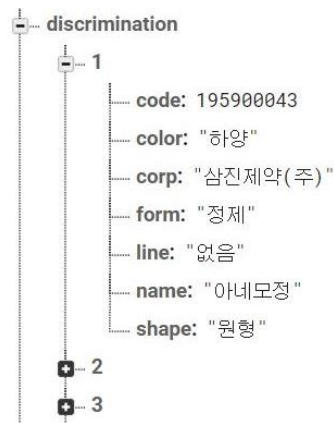
의약품의 품목 기준코드에 맞는 이미지를 가져오기 위해 소스 코드에서 변수 ‘image-name’에 이미지 파일의 이름을 포함한 이미지의 경로를 저장하였고, 품목 기준코드와 일치하는 이미지가 존재하면 해당 이미지를 불러온다.

의약품 이미지는 [그림 14]과 같이 저장되어 있다.

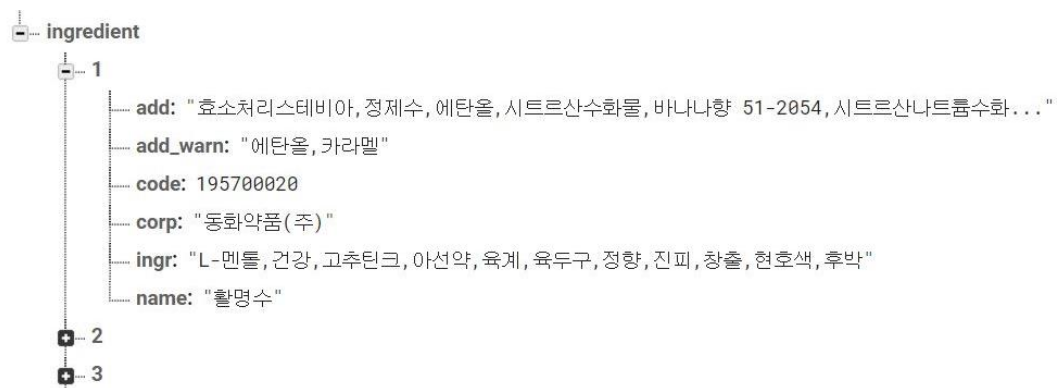
<input type="checkbox"/> 이름	크기	유형
<input type="checkbox"/> 195900043.png	57.2 KB	image/png
<input type="checkbox"/> 196000011.png	53.8 KB	image/png
<input type="checkbox"/> 196200034.png	179.86 KB	image/png
<input type="checkbox"/> 196200043.png	59.11 KB	image/png
<input type="checkbox"/> 196400046.png	140.22 KB	image/png

[그림 14] Cloud Storage에 저장된 이미지

11.2.4 Firebase Realtime Database에서 데이터 가져오기



[그림 15] Firebase Realtime Database에 저장된 데이터(약 모양)



[그림 16] Firebase Realtime Database에 저장된 데이터(성분)

Firebase Realtime Database에 약 모양에 대한 데이터가 [그림 15]와 같이 ‘discrimination’ 디렉토리에 저장되어 있으며, 의약품 성분에 대한 데이터가 [그림 16]과 같이 ‘ingredient’ 디렉토리에 저장되어 있다.

약 모양에 대한 데이터는 의약품의 품목 기준코드(code), 제품명(name), 제조 회사명(corp), 색상(color), 제형(form), 분할선(line), 모양(shape)의 정보가 저장되어 있다. dataSnapshot 객체를 통해 데이터베이스로부터 데이터를 가져와 약 모양 검색 기능에서 사용자가 선택한 색상, 종류, 분할선, 모양에 대한 조건에 모두 부합하는 의약품을 찾고, ArrayList에 품목 기준코드, 제품명, 제조 회사명을 저장한다. ArrayList에 저장된 정보를 List 형태로 띄워준다.

의약품 성분 데이터는 품목 기준코드(code), 제품명(name), 제조 회사명(corp), 성분(ingr), 첨가제(add), 첨가제 주의 관련 성분(add_warn)이 저장되어 있다. 이 또한 dataSnapshot 객체를 통해 데이터를 가져오고, 성분 검색 기능과 성분 상세정보에서 사용된다. 성분 검색 기능에서는 사용자가 검색한 단어가 성분 데이터에 포함되어 있으면 해당 의약품의 품목 기준코드, 제품명, 회사명을 가져오는 방식으로 설계되어 있으며, 성분 상세정보를 띄워줄 때는 해당 의약품의 품목 기준코드와 데이터베이스의 code가 일치하는 데이터를 가져온다.

11.2.5 일반의약품 상세 정보 검색

아이고약! 제품명, 성분명, 약모양 탭이 있습니다. 제품명 탭이 선택되어 있으며, '제품명 입력' 텍스트 필드와 검색 아이콘이 있습니다. 하단에는 '복약관리', '의약품 검색', '마이페이지' 아이콘이 있습니다.

[그림 17] 제품명 검색

아이고약! 제품명, 성분명, 약모양 탭이 있습니다. 성분명 탭이 선택되어 있으며, '성분명 입력' 텍스트 필드와 검색 아이콘이 있습니다. 하단에는 '복약관리', '의약품 검색', '마이페이지' 아이콘이 있습니다.

[그림 18] 성분명 검색

아이고약! 제품명, 성분명, 약모양 탭이 있습니다. 약모양 탭이 선택되어 있으며, '제형', '모양', '색상', '분할선' 필터가 있습니다. 하단에는 '확인' 버튼과 '복약관리', '의약품 검색', '마이페이지' 아이콘이 있습니다.

[그림 19] 약모양 검색

일반의약품의 제품명, 성분명, 약 모양으로 의약품 정보를 검색할 수 있다. 제품명 검색에서는 사용자가 제품명을 입력하여 검색하면 제품명에 검색어가 포함되는 의약품을 리스트로 나타낸다. 성분명 검색에서는 사용자가 입력한 성분명을 포함하는 의약품을 리스트로 나타낸다. 제품명 검색에서 검색 결과 리스트를 나타낼 때 사용하는 데이터베이스는 Open API이며, 성분명 검색에서 검색 결과 리스트를 나타낼 때 사용하는 데이터베이스는 Firebase Realtime Database의 'ingredient' 이다. 약 모양 검색에서는 사용자가 제형, 모양, 색상, 분할선의 조건을 선택하고 확인 버튼을 누르면 해당하는 의약품을 리스트에 나타낸다. 약 모양 검색에서 검색 결과를 리스트를 나타낼 때 사용하는 데이터베이스는 Firebase Realtime Database의 'discrimination' 이다.

아이고약! 검색결과 526건. 아네모정 (삼진제약(주)), 아로나민골드정 (일동제약(주)), 태극아즈렌에스연고(구아아졸렌) (태극제약(주)), 에이티피장응정(아데노신트리포스페이트이나트륨삼수화물) (제일약품(주)), 리렌스연고(구아아졸렌) (주)씨엠지제약. 각 항목에는 하트 아이콘이 있습니다.

[그림 20] 검색 결과 리스트

아이고약! 상세정보, 성분정보, DUR 정보 탭이 있습니다. 상세정보 탭이 선택되어 있으며, 회사명(경남제약(주)), 제품명(경남나프록센연질캡슐), 효과 및 효능, 사용법 정보가 표시되어 있습니다.

[그림 21] 일반의약품 상세 정보

검색 결과 리스트는 [그림 20]과 같이 나타나게 되며, 검색 결과 리스트에서 하나의 의약품을 선택하면 해당 의약품의 상세정보가 [그림 21]과 같이 나타나게 된다. 상세정보 페이지는 의약품의 이미지와 상세정보 탭, 성분 정보 탭, DUR 정보 탭으로 이루어져 있다.

상세정보 탭에서는 의약품의 기본 정보인 회사명, 제품명, 효과 및 효능, 사용법, 사용 전 반드시 알아야 할 사항, 사용상 주의사항, 사용하는 동안 주의해야 할 사항, 부작용, 보관법을 확인할 수 있으며, 이 데이터는 ‘식품의약품안전처_의약품개요정보(e약은요) API에서 가져온다. 또한, 상세정보 탭에서 의약품 구분, 의약품 분류, 색상 및 형태, 제형, 모양, 색깔, 크기, 분할선 정보도 확인할 수 있으며, 이 데이터는 ’ 의약품 낱알식별정보(DB) 서비스 ‘API에서 가져온다.

성분 정보 탭에서는 의약품의 제품명, 주성분, 첨가제, 첨가제 주의 관련 성분을 확인할 수 있으며, 이 데이터는 Firebase Realtime Database의 ‘ingredient’ 에서 가져온다.

DUR 정보 탭에서는 의약품의 DUR 정보인 병용 금기, 연령 금기, 임부 금기, 노인 주의, 용량 주의, 투여 기간 주의를 확인할 수 있다. 이 데이터는 SQLite 데이터베이스인 ‘dur.db’ 에서 가져온다.

의약품 상세정보에서 데이터를 가져오는 기준은 품목 기준 코드이며, 검색 결과 리스트에서 선택한 의약품의 품목 기준코드와 각각의 데이터베이스의 데이터 중 품목 기준코드가 일치하는 데이터를 상세정보 페이지에서 보여주도록 구현하였다.

11.2.6 복약 기록



[그림 22] 복약 기록 추가 페이지

[그림 23] 복약 관리 페이지

달력을 활용하여 복약 기록을 할 수 있는 기능이다. 복약 관리 페이지에서 ‘+’ 버튼으로 복약 추가 페이지로 이동할 수 있다. 복약 추가 페이지에서 제품명, 복약 시작 및 종료 날짜, 복용

시각을 설정하면 복약 기록 리스트에 추가되며 복약 관리 페이지에서 확인할 수 있다. 추가된 의약품에 대한 정보는 사용자 정보 데이터베이스인 ‘note.db’의 ‘TABLE_NOTE’에 저장된다. 달력에서 날짜를 선택하게 되면 해당 날짜에 복용해야 하는 의약품 항목들을 볼 수 있다.

복약 기록 리스트의 각각의 항목은 ‘미복용 및 복용’ 버튼을 사용하여 복용 여부를 저장할 수 있으며, 휴지통 버튼을 누르게 되면 복약 기록 리스트에서 삭제할 수 있다. 복약 기록 리스트에서 삭제된 항목은 ‘note.db’의 ‘TABLE_NOTE’에서도 삭제된다.

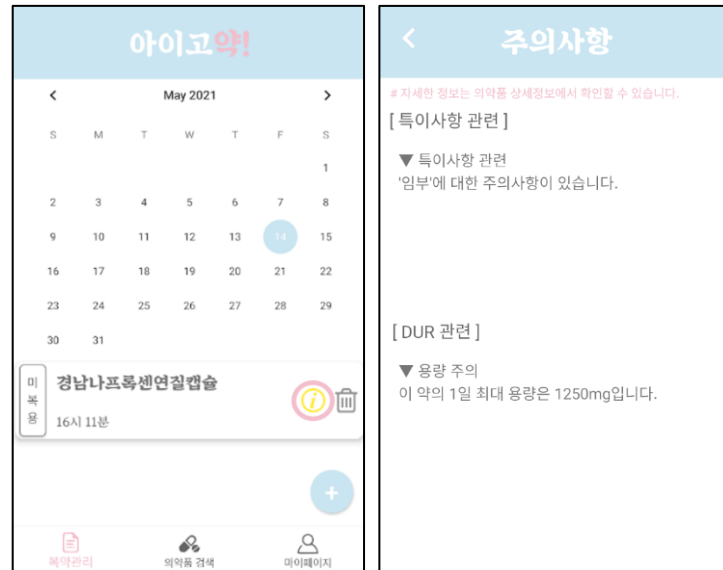
11.2.7 복용 의약품에 따른 맞춤형 주의사항 제공

복용하는 의약품을 복약 기록 리스트에 추가하면 사전에 설정한 사용자의 건강 상태(특이사항)에 대한 주의사항과 해당 의약품에 대한 병용 금기, 부작용 등에 대한 주의를 주는 기능이다.

[그림 24] 나이 설정

[그림 25] 특이사항 설정

애플리케이션을 설치하고 처음 실행하게 되면 [그림 24], [그림 25]와 같이 사용자의 나이와 특이사항을 설정할 수 있다. 이를 기반으로 사용자 맞춤형 복용 시 주의사항을 알려준다. 사용자의 나이와 특이사항은 ‘note.db’의 ‘TABLE_USER’에 저장된다. 설정한 사용자의 정보는 ‘회원 정보’ 메뉴에서 확인 및 수정할 수 있다.



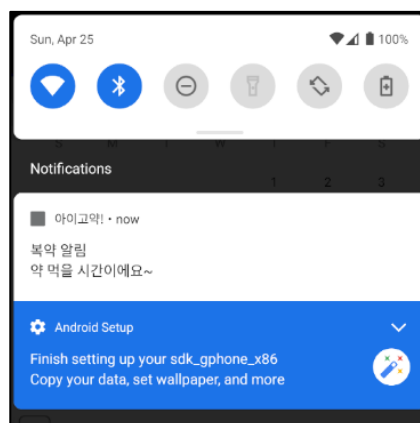
[그림 26] 복약 기록 항목의 ① 버튼 [그림 27] 주의사항 페이지

복약 기록 리스트에 추가된 항목의 노란색 ①버튼을 누르게 되면 사용자 맞춤형 주의사항을 볼 수 있다.

‘특이사항 관련’ 항목에는 사용자가 설정한 특이사항과 관련이 있다. 의약품의 주의사항에 특이사항이 포함되어 있는 경우 해당 특이사항과 관련하여 주의사항이 있음을 알려준다. 또한, 사용자의 나이가 65세 이상으로 설정된 경우 의약품의 주의사항에 ‘고령자’ 및 ‘노인’이 포함되면 특이사항 관련 항목에서 알려준다. 의약품의 주의사항은 ‘식품의약품안전처_의약품개요 정보(e약은요)’ API의 ‘atpnQesitm’ 태그의 내용을 사용한다.

‘DUR 관련’ 항목에는 의약품에 대해 DUR 정보가 존재하면 내용을 띄워주며, 연령 금기의 경우 사용자의 나이가 의약품의 연령 금기에 해당하는 경우에만 나타낸다.

11.2.8 복약 시각 알림

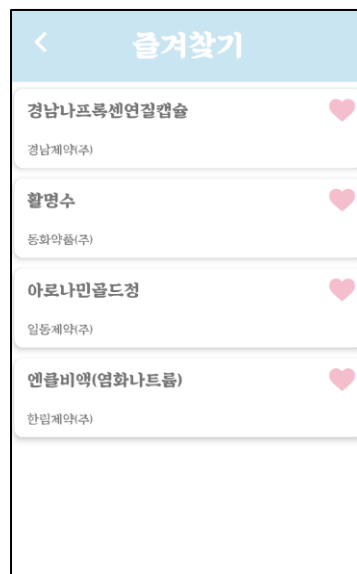


[그림 28] 복약 알림

복약 기록 리스트에 저장된 항목들의 복약 시각에 대한 알림 기능이다. 복약 기록 리스트에 추가하면 복용 시각에 알람이 자동 생성되며, 복약 시각임을 알리는 알람이 울리게 된다. 복약 추가 페이지에서 복약 시각을 설정하면 'note.db'의 'TABLE_NOTE'의 'ALARM' 컬럼에 알람의 ID가 복약 시각을 기반으로 자동으로 설정되며 알람의 ID에 따라 알람이 울린다. 알람은 복약 기록 리스트에 추가된 항목마다 알람이 생성되므로 항목이 여러 개인 경우 알람이 여러 개 설정된다. 복약 기록 리스트에 항목을 추가할 때 복약 시작 날짜와 복약 종료 날짜를 다르게 설정할 경우 해당 기간 동안 설정된 복약 시간에 알람이 울리게 된다.

복약 기록에서 항목을 삭제하게 되면 'note.db'의 'TABLE_NOTE'에서 항목에 대한 데이터가 삭제되기 때문에 알람도 같이 삭제된다.

11.2.9 의약품 즐겨찾기



[그림 29] 즐겨찾기 페이지

자주 찾는 의약품 정보에 대해 즐겨찾기 추가할 수 있는 기능이다. 즐겨찾기 추가는 의약품 검색리스트나 의약품 상세정보의 하트 아이콘으로 할 수 있다. 추가한 즐겨찾기 항목은 즐겨찾기 메뉴에서 확인할 수 있다. 즐겨찾기 리스트에서 항목을 선택하게 되면 상세정보를 볼 수 있다. 즐겨찾기 리스트에서도 즐겨찾기 설정, 해제가 가능하다. 사용자의 즐겨찾기 정보는 'note.db'의 'TABLE_BOOKMARK'에 저장되며, 즐겨찾기 해제 시 삭제된다.

12. 구현결과

12.1 로딩화면



[그림 30] 로딩화면

12.2 초기 정보 설정 화면

아이고약!

나이를 입력하세요

세

다음

아이고약!

특이사항을 선택하세요

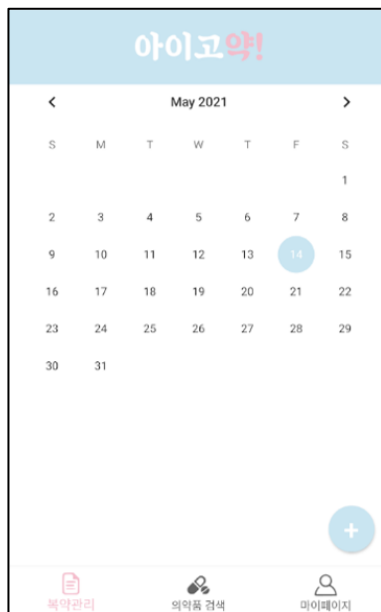
MAO	과민증	피부염	녹내장
당뇨	두드러기	배뇨	부정맥
소화성궤양	신장	심장	알레르기
유당불내효소 결핍증	위장	임부	천식
혈액응고	피부감염증	혈압	

완료

[그림 31] 나이 설정

[그림 32] 특이사항 설정

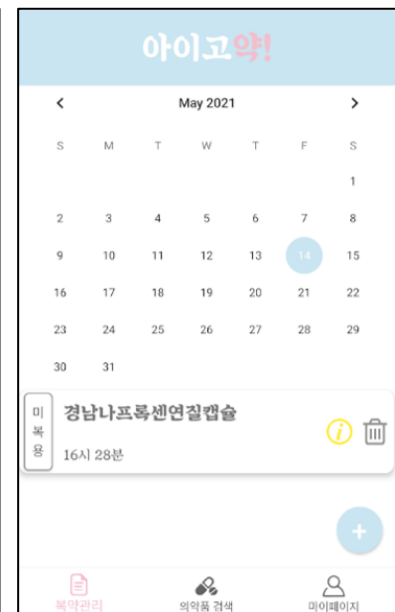
12.3 복약 관리 화면



[그림 33] 복약 관리 메인 화면



[그림 34] 복약 기록 추가 화면



[그림 35] 복약 기록 항목이 추가된 화면

12.4 복약 기록 리스트에 따른 맞춤형 주의사항 제공 관련 화면



[그림 36] 노란색 ① 버튼



[그림 37] 주의사항 화면

12.5 회원 정보 화면

< 회원 정보

나이 20 세

특이사항 위장
임부

< 회원 정보 수정

나이 10 세

특이사항

MAO	과민증	피부염	녹내장
당뇨	두드러기	폐렴	부정맥
소화성궤양	신장	심장	알레르기
유당불내증	위장	임부	천식
혈액응고	피부감염증	혈압	추가등록

#새로운 키워드를 등록하세요!

추가하기

삭제하기

완료

[그림 38] 회원 정보 화면

[그림 39] 회원 정보 수정 화면

12.6 복약 시각 알림 화면

[그림 40] 복약 알림

GNICT- 20211015-001

[BCD]

Page: 25/31

12.7 일반의약품 검색 화면

[그림 41] 제품명 검색

[그림 42] 성분명 검색

[그림 43] 약모양 검색

12.8 검색 결과 리스트 화면

[그림 44] 검색 결과 리스트

12.9 일반의약품 상세 정보 화면



[그림 45] 일반의약품 상세 정보

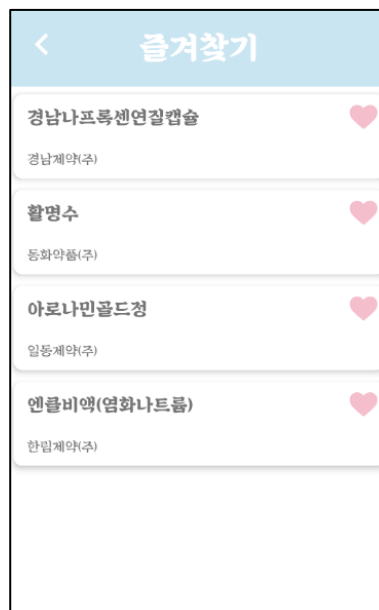


[그림 46] 일반의약품 성분 정보



[그림 47] 일반의약품 DUR 정보

12.10 즐겨찾기 화면



[그림 48] 즐겨찾기 화면

13. 결론

13.1 결론 및 기대효과

‘아이고약!’은 사용자 맞춤형 주의사항을 제공하여 복약 관리를 쉽고 간단하게 할 수 있도록 제작하였다. 복약 관리 기능에서 사용자의 건강 상태에 따른 주의사항 정보와 DUR 정보를 제공하여 사용자가 복약 시 주의사항을 숙지할 수 있도록 도와준다는 최종적인 목표를 담았다. 이 애플리케이션을 사용하면 일반의약품의 사용상 주의사항을 일일이 검색하여 읽어보지 않아도 사용자 맞춤형 복약 관리 기능을 통해 한눈에 확인할 수 있으며, 이를 통해 일반의약품 복약 시 따르는 위험성을 줄일 수 있을 것이다. 또한, 복약 관리 시스템을 가진 기존의 애플리케이션들을 사용법이 복잡한 단점이 있다. 이 단점을 보완하기 위해 쉽게 사용할 수 있게 직관적으로 만들었고, 애플리케이션의 단순한 사용법으로 모든 연령층이 이용할 수 있음을 기대할 수 있을 것이다.

13.2 프로젝트를 마치며

이 프로젝트는 처음부터 끝까지 새로움과 배움의 연속이었다. 주제를 선정하여 프로그램을 기획하는 것부터 시작해서 새롭게 접한 안드로이드 애플리케이션 개발을 거쳐 보고서 작성까지, 하나하나 새로운 과정을 밟아가며 프로젝트를 진행하였다. 처음이었기에 모든 것이 어려웠고, 그만큼 많은 일이 있었다. 그로 인해 배운 점들도 많았다.

우리는 간단한 코딩만 할 줄 아는 채로 시작하게 되었다. 모바일 애플리케이션을 개발하는 것은 처음이라 안드로이드 애플리케이션을 개발하기 위해 사용하는 안드로이드 스튜디오를 사용하는 방법조차 몰라서 막막하였다. 처음 기획할 때 애플리케이션의 기능을 구상하고 사용할 데이터를 찾는 과정까지는 순조롭게 진행되는 듯하였다.

하지만 본격적으로 코딩을 하면서 구현하는 과정에서부터 어려움을 겪게 되었다. 화면 구현부터 진행하였는데, UI 설계서대로 구현하는 과정에서 화면 비율을 고려하여 개체들을 만드는 부분과 Fragment를 설계하는 부분에서 다양한 문제가 있었고 해결하는 과정에서 오랜 시간이 걸렸다. 기능을 구현하는 과정에서는 특히 여러 개의 알람을 생성하는 방법을 검색해도 나오지 않아 힘들었다. 또한, 공공데이터를 사용하기 때문에 데이터베이스를 구상하는 것이 중요했는데, 데이터베이스를 설계하고 구축할 때 애플리케이션의 성능과 데이터베이스 관리의 용이성을 생각해야 했다. 구축한 데이터베이스를 애플리케이션에 연결할 때 여러 부분에서 오류가 발생한 것을 포함하여 전반적으로 고충이 있었다.

이처럼 개발 과정은 문제의 연속이었는데, 문제를 해결하기 위해 난해하고 두꺼운 책들도 찾아서 연구해보고 풀리지 않는 부분에서는 팀원들과 머리를 맞대어 고민하고 의논하며 해결해갔다.

이 프로젝트를 진행하면서 새롭게 배운 점들 또한 많았다. 프로젝트의 모든 과정을 팀원들과 힘을 합쳐 검색하고 공부하여 하나씩 알아가며 개발하였다. 학교에서 개인 과제를 할 때와는 다르게 팀원과 서로 의논하고 배워가며 진행하는 점이 참 좋았다.

개발하면서 주로 Java를 사용하였는데, 학교에서 수강한 Java 강의에서 배운 것들을 실제로 개발할 때 어떤 식으로 사용해야 하는지를 알게 되었다. 좋은 코드를 작성하기 위해서는 Java에 대해서 깊이 있게 이해하고 있는 상태에서 코딩해야 한다는 것을 다시금 깨닫게 되었다.

어떤 프로그램이든 가장 중요한 부분 중 하나가 데이터와 데이터베이스라고 생각한다. 데이터를 효과적으로 활용하는 것도 중요하지만 유의미한 데이터를 수집하는 것도 많은 시간과 돈이 든다. 개발 과정에서 공공데이터를 사용하게 되었는데, 공공데이터가 있었기에 데이터를 수집 하는데 사용되는 돈과 시간을 아낄 수 있었다. 공공데이터를 활용하면서 국가에서 무료로 다양하고 유용한 데이터를 제공하기 때문에 우리나라의 IT 분야가 급속도로 발전하고 인정받고 있다고 느꼈다.

팀원과의 협업을 위해 Git을 사용하게 되었다. Git이 있다는 것을 알고 있었지만 본격적으로 써보게 된 건 처음이었는데, 개발자의 대부분이 왜 Git과 GitHub를 사용하고 있는지 알게 되었다. Git은 간단한 사용법으로 협업과 버전 관리를 한 번에 할 수 있다는 점이 유용하였고, 이번 계기로 다음 협업에서는 Git의 기능을 좀 더 다채롭게 활용할 수 있을 것 같다.

진행 과정에서 프로그램 개발뿐만 아니라 개발에 관련된 다양한 문서도 작성하게 되었다. 학교에서 소프트웨어공학을 배우며 작성해 본 문서들도 있었고, 이름조차 처음 듣는 문서도 있었다. 이전에 작성해 본 문서는 경험을 살려서 부족했던 점을 보완하여 완성도 있게 작성할 수 있었다. 처음 접한 문서는 검색을 통해 어떤 문서인지 알게 되었고, 다양한 자료를 참고하며 작성하였다. 하나의 프로그램을 개발하기 위해서는 다양한 분야의 사람들이 투입되고, 사람들끼리의 의견 교환을 위해 문서를 사용한다는 것을 알게 되었다. 여러 분야의 사람들이 함께 진행하기 때문에 문서의 종류도 다양하고, 커뮤니케이션을 위해 사용되는 만큼 최대한 자세하고 정확하게 작성해야 한다는 것을 깨닫게 되었다.

이번 프로젝트는 기술 외적으로도 느낀 점이 많았다. 가장 크게 다가온 점은 팀원들과 커뮤니케이션의 중요성이다. 물론 팀원이 두 명뿐인 작은 팀이었지만, 사소한 부분에서도 서로의 생각을 물어보고 조율해가며 진행하였다. 구성 단계부터 수시로 회의했음에도 불구하고 같은 부분을 다르게 이해할 수 있다는 것을 깨달았다. 그로 인해 팀원 간의 커뮤니케이션이 매우 중요하다는 것을 느끼게 되었고, 팀원들과 함께 개발할 때는 나의 생각을 상대방에게 정확하게 전달하기 위해서는 어떻게 하면 좋은가를 배웠다.

그리고 개발에 대해 자신감을 가지게 되었다. 이전에는 개발할 때 막히는 부분이 생기면 자신

감이 떨어졌고, 이번 프로젝트를 진행하면서도 많은 고난을 겪었다. 하지만 팀원과 함께 고민하고 여러 방법을 찾으면서 해결할 때마다 어떤 것이든 진심으로 노력하면 언젠간 될 것이라는 자신감을 얻었다.

이렇게 간단한 프로그램, 알고리즘을 짜는 방법만 알던 과거와 달리 안드로이드, 데이터베이스 구축 등 여러 가지 기술을 사용하여 애플리케이션의 기능을 구현하고 실행시킬 수 있는 프로그램을 만들 수 있게 되었다. 처음으로 하나의 완성된 애플리케이션을 만들었다는 점이 뿌듯하였고, 애플리케이션이 만들어지는 과정을 이해할 수 있게 되었다.

이 애플리케이션에 적용하고 싶은 또 다른 기능들이 있었지만 담지 못하였다. 시간이 부족했던 것도 있었지만 지금은 그 기능들을 구현하기에 실력이 부족했다. 프로젝트가 끝나도 관련 공부를 계속하여 생각만 했던 기능들도 전부 구현해보고 싶다. 또한, 모바일 애플리케이션의 개발 경험이 부족했기 때문에 체계적으로 계획을 세워 진행하지 못한 점이 아쉬웠다.

이 프로젝트를 진행하면서 어느 부분에서 부족함이 있는지 깨닫게 되었고, 앞으로 보완해 나가야 할 방향을 알게 되었다. 아직은 클린 코드를 작성할 만큼 실력이 되지 않지만, 이후의 학습을 통해서 체계적으로 구조를 설계하고 코드를 작성하여 또 하나의 작품을 만들어가고 싶다.

14. 최종 산출물

본 프로젝트 수행 시에 각 단계별 산출물에 대한 제출 문서는 아래와 같음

NO	산출물명	수량	매체	제출시기	비고
1	개발계획서	1	문서	시작 후 2주 이내	
2	분석/설계명세서 USECASE 다이어그램(요구사항분석서) ERD 기능차트 용어사전 클래스다이어그램, 시퀀스다이어그램(DFD), 테이블목록, 테이블정의서 프로그램목록 사용자 인터페이스 설계서 메뉴 구성도		문서	* ERD는 2021.10.15일 필수제출이며 기타자료도 제출 가능함.	
3	사용자 메뉴얼	1	문서	2021.10.15	
4	프로젝트 완료보고서, 구동경로, 시연 동영상 등 인수인계 확인서(제출로 대체함)	1	문서	2021.10.15	