

효율적 일상 관리를 위한 일정관리 어플리케이션의 설계와 구현[☆]

Design and Implementation of Scheduler Applications for Efficient Daily Management

박 은 주¹ 한 승 준² 윤 지 민¹ 임 한 규^{1*}
Eunju Park Seungjun Han Jimin Yoon Hankyu Lim

요 약

IT기술과 정보처리 기술의 발전에 따라 인터넷 사용량이 급증하고 다양한 스마트 기기들이 출현하고 있다. 이에 따라 현대인들은 장소와 시간에 구애받지 않고 원하는 정보 획득, 여가활동과 같은 일상생활에 스마트폰을 사용한다. 본 논문에서는 일정을 기록하고 약속을 공유하는 등 일상생활의 효과적 관리에 도움이 되는 일정관리 어플리케이션을 설계하고 이를 구현하였다. 본 논문의 일정관리 어플리케이션은 일정등록과 함께 일기작성 기능, 등록된 일정을 카카오톡을 이용하여 다른 사용자와 공유할 수 있는 기능, 삭제 시 특정 폴더로 삭제한 일정이나 일기를 저장하는 기능, 하루 일정에 대한 연속적인 알림 기능 등을 제공한다. 다른 일정관리 앱들과 차별점을 가지며 사용성을 높인 본 논문의 어플리케이션은 바쁜 일상생활의 효율적 관리에 도움이 될 것이라 기대한다.

☞ 주제어 : 모바일 어플리케이션, 일정관리 어플리케이션, 데이터 공유, 앱 사용성

ABSTRACT

According to the progress of IT and data processing technology, the usage of internet increased rapidly and various smart devices are appearing. As such, modern people use smartphone to acquire informations they wish and also on daily life including leisure activity free of place. This study has designed and implemented schedule managing application that can help effective managing of our daily life, such as taking note of schedule and sharing appointments. The schedule managing application in this study offers diary taking, sharing the registered schedule with other users on kakaoalk, saving the deleted schedule or diary to certain folder when users delete file, continuous alarm of daily schedule function together with schedule registration function. The application which is differentiated to other applications and raised usage is expected to effectively manage the busy everyday life.

☞ keyword : Mobile Application, Scheduler Application, Data Share, App Usability

1. 서 론

정보통신과학기술의 발전과 함께 인터넷 사용량이 급증하고 다양한 스마트 기기가 출현함에 따라 일상생활이 사이버 공간으로 확대되는 변화와 함께 삶의 질 또한 향상되고 있다[1]. 방송통신위원회 ‘2019 방송매체 이용행태조사’에 따르면 다양한 스마트 기기 가운데 매체이용의 개인화를 대변하는 국내 스마트폰 보유율이 91.1%로

지속적으로 증가하고 있고, 응답자 기준 63%가 스마트폰을 필수매체로 인식하고 있는 것으로 조사되어 스마트폰의 영향력이 점점 커지는 것을 알 수 있다[2]. 이에 스마트폰은 많은 사람의 일상과 커뮤니케이션에 커다란 변화를 가져오고 인간의 삶에 영향을 주는 강력한 도구로 산업과 개인 일상생활에 영향력을 미치고 있다. 또한, IT 기술의 발전과 정보처리 기술 발달로 인하여 다양한 분야에서 IT를 융합한 많은 정보 서비스들이 제공되고 있다 [3][4]. 이런 시대 흐름에 따라 현대인들은 언제, 어디서나 실시간으로 원하는 정보를 획득하고 여가활동, 교육 및 학습, 커뮤니케이션 활동 등 다양한 분야에 스마트폰을 활용하고 있다[5][6]. 특히, 모바일 네이티브(Mobile Native)라 불리는 젊은 세대들은 태어났던서부터 디지털 환경을 자연스럽게 받아들이고 모바일을 통해 얻은 정보를 기반으로 급변하는 사회 현상에 익숙하게 적응하고 생활한다[7][8].

1 Dept. of Multimedia Engineering, Andong National University, Andong, 36729, Korea.

2 Computer Science and Engineering, Kyungpook National University, Daegu, 41566, Korea.

* Corresponding author (hklim@anu.ac.kr)

[Received 21 October 2020, Reviewed 3 November 2020(R2 17 February 2021, R3 3 April 2021), Accepted 19 April 2021]

☆ 본 연구는 과학기술정보통신부 및 정보통신기획평가원에서 지원하는 SW중심대학사업(IITP-2019-0-01113)의 연구 결과로 수행되었음

디지털 환경에 익숙하고 모바일 정보를 활용하는 현대인들은 간단한 의사표시나 개인 간의 약속에 휴대폰 SMS나 모바일 메신저, 일정관리 어플리케이션 등을 사용하는 경우가 많다[9]. 특히 자신의 역할에 적합한 업무나 해야 할 일들을 정리하고 기억하기 위해 다양한 일정관리 서비스들을 사용하고 있는데 일정관리 서비스들은 편리성을 위해 PC Web 서비스보다는 모바일 어플리케이션을 이용하는 경우가 많다[10]. 이에 이지플래너, 원데이, 네이버 캘린더, 구글 캘린더, 애플 캘린더와 다양한 스케줄러 앱들이 간단한 일정관리와 시간 관리를 위해 도구로 사용되고 있다.

이에 본 논문에서는 기존 스케줄러 앱들이 가진 불편함을 보완하고 연령에 상관없이 쉽고 간편하게 이용할 수 있는 효율적인 일정관리 어플리케이션을 설계하고 이를 구현하였다. 본 논문에서 설계하고 구현한 일정관리 어플리케이션이 기존에 사용되는 스케줄러 어플리케이션들과 차별화된 기능은 다음과 같다. 본 논문의 일정관리 어플리케이션은 일정을 등록하는 기능과 함께 일기를 작성할 수 있는 기능을 함께 제공한다. 일정공유 기능을 사용하여 카카오톡을 통해 자신 또는 일정의 공유가 필요한 사용자와 일정을 공유할 수 있다. 실수로 삭제한 일정이나 일기의 복구가 가능하도록 삭제된 일정과 일기는 바로 삭제하지 않고 저장소로 이동한 후 사용자의 삭제 확인 요청으로 삭제한다. 이와 같이 차별화된 기능 이외에 알림설정 기능과 일정과 일기를 확인하고 평가할 수 있는 기능 등 일정관리 앱이 가지는 기능을 함께 설계하고 이를 구현하였다.

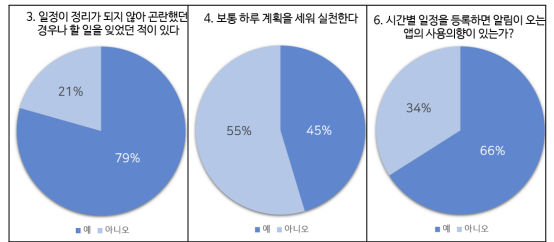
본 논문의 구성은 다음과 같다. 2장 관련 연구에서는 본 논문에서 설계한 어플리케이션의 벤치마킹과 구현 타당성을 위한 설문조사 등 관련 연구를 기술하였다. 3장에서는 구조적 설계와 기능 설계를 기술하였고 4장은 구현을 기술하였다. 5장은 결론에 관하여 기술하였다.

2. 관련 연구

본 논문에서 구현하고자 하는 일정관리 어플리케이션의 구현 타당성 조사를 위해 2019년 4월 2일부터 4월 12일까지 네이버 폼을 사용한 설문조사를 실시하였고 응답자는 97명이었다. 설문조사 결과 일정이 정리되지 않아서 해야 할 일을 잊어버린 경험이 있다고 대답한 응답자가 77명(79%)이었다. 또한, 하루의 계획을 세워서 실천한다고 응답한 응답자가 44명(45%)으로 응답자의 55%는 하

루의 계획을 세우지 않는 것으로 응답하였다. 일정관리와 하루를 점검할 수 있는 어플리케이션에 대한 사용 의사가 있다고 응답한 응답자는 64명(66%)이었다. 표 1은 설문조사 결과 가운데 설계에 도움이 되는 유의미한 결과를 나타내었다.




(표 1) 설문조사
(Table 1) Survey



이상의 설문조사 결과에서 하루의 계획을 세우지 않아 곤란한 일들을 겪는 경우가 많았고 이에 스마트폰을 사용하여 간편하게 일정을 등록하면 해야 할 일들을 알려주는 알림이 오는 어플리케이션에 대한 사용요구가 많은 것으로 조사되었다.

본 논문에서는 기존 스케줄러 앱들이 가진 단점들을 개선하고 유용한 기능 설계를 위한 벤치마킹을 실시하였다. 표 2는 벤치마킹한 어플리케이션을 나타낸 표이다.

(표 2) 벤치마킹
(Table 2) Benchmarking

구분	장점	단점
원데이 	-일기에 별점·스티커 -그룹기능 사용과 편리한 메모관리 -구글 드라이브 연동한 데이터 관리	-알림반복주기, 반복종료일자 설정 불가능 -일정을 시간으로만 등록 가능
이지플래너 	-상태바를 내리면 즉시 메모 가능 -일정 하루 전 알림 자동 설정 -다양한 위젯 제공(월간, 6일, 3일)	-다른 사용자와 공유 불가능 -이틀 이상 지속 일정 설정 번거로움 -메모 글자 수의 제한
Do 	-등록일정 위젯으로 뽐 -터블탭 사용한 간단한 완료 -드롭박스와 이메일로 데이터 백업·복원	-리스트 체크 시 내용 즉시 삭제 -지운 일정 복구 불가 -날짜별 정렬 가능, 구분 불가능

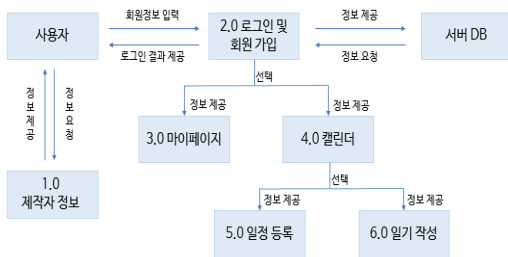
본 논문에서 벤치마킹한 어플리케이션은 메모와 일정 관리를 함께 할 수 있는 어플리케이션 가운데 사용자가 많은 어플리케이션을 위주로 벤치마킹하였다. 현재 많은 스케줄러 어플리케이션들이 개발되어 사용되고 있고 각 어플리케이션들은 기능들과 사용법들에서 조금씩의 차이점을 가진다. 본 논문에서는 사용자 편의성을 우선으로 고려하여 스케줄러 어플리케이션을 설계하여 벤치마킹한 어플리케이션들이 가진 장점들을 포함하고 단점들은 보완함으로써 편의성을 높인 어플리케이션으로 개발하고자 하였다.

이에 본 논문에서 설계하고 개발한 어플리케이션은 스케줄러 어플리케이션이 가지는 일정등록, 알림 기능을 제공하며 이와 함께 벤치마킹한 어플리케이션을 기반으로 사용자 편의성을 위한 등록된 일정의 다른 사용자와 공유기능, 삭제한 일정과 일기의 보관 기능, 알림 반복 기능, 일기작성 기능 등을 설계하고 이를 구현하였다.

3. 일정관리 어플리케이션 설계

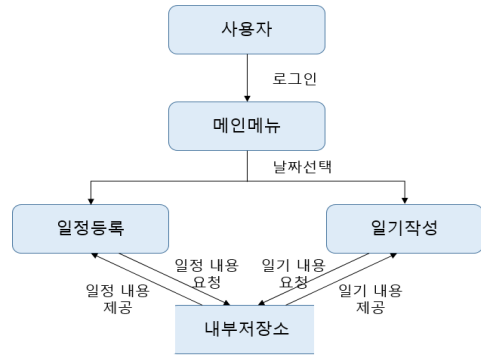
3.1 구조도 설계

본 논문의 일정관리 어플리케이션은 ‘마이페이지’, ‘캘린더’, ‘일정등록’, ‘일기작성’ 등의 기능을 제공한다. ‘마이페이지’ 기능은 자신의 정보를 저장하는 기능이다. 그림 1은 본 논문의 일정관리 어플리케이션의 전체적인 구조도를 나타낸 그림이다.



(그림 1) 일정관리 어플리케이션 구조도
(Figure 1) Structure Diagram of Scheduler Application

그림 2는 일정등록과 일기작성 기능에 대한 구조도이다. 사용자는 캘린더로 제공되는 메인 메뉴를 통해 날짜를 선택하고 해당 날짜의 ‘일정등록’이나 ‘일기작성’ 메뉴를 선택할 수 있다.

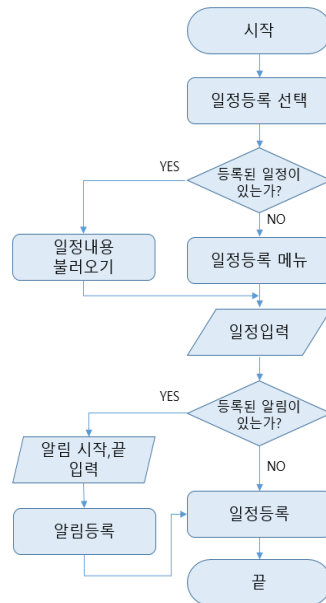


(그림 2) 일정등록, 일기작성 구조도
(Figure 2) Structure Diagram of Schedule Registration and Diary Writing

각 메뉴는 사용자가 선택한 날짜에 등록된 일정이나 일기가 있는지 확인하고 사용자에게 제공한다.

3.2 플로차트

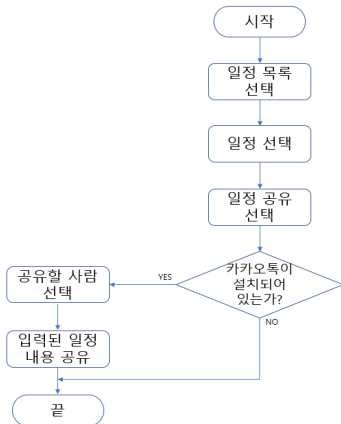
그림 3은 ‘일정등록’ 기능에 대한 플로차트를 나타낸 그림이다.



(그림 3) 일정등록 기능 플로차트
(Figure 3) Flowchart of Schedule Registration Function

일정을 등록하기 위해 ‘일정등록’ 메뉴를 선택하면 내부 저장소에 이미 등록된 일정이 있는지를 확인하고 등록된 일정이 있는 경우에는 등록된 내용을 일정등록 화면으로 불러와 내용을 수정할 수 있다. 등록된 일정이 없는 경우에는 새로운 일정을 작성할 수 있다. 알림설정을 위해 일정의 시작 시간, 끝나는 시간을 입력받아 시작 시간에 Notification을 이용하여 알림이 울리도록 설정한다.

그림 4는 일정공유 기능에 대한 플로차트이다.



(그림 4) 일정공유 기능 플로차트

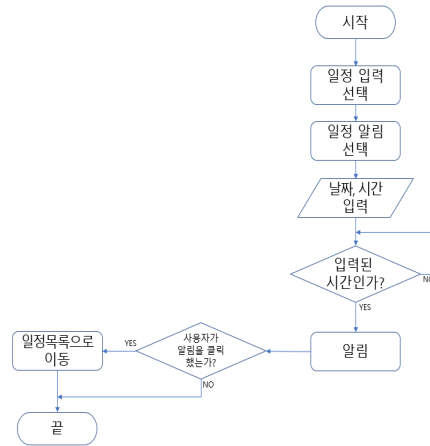
(Figure 4) Flowchart of Schedule Sharing Function

일정 목록들 가운데 하나의 일정을 선택하면 ‘일정공유’를 선택할 수 있다. 본 논문에서는 국내에서 많이 사용하는 카카오톡을 사용하여 일정을 공유하도록 설계하고 구현하였다. ‘일정공유’를 선택하고 카카오톡이 설치된 경우 공유할 사람을 선택하여 사용자가 입력한 일정 내용을 카카오톡을 사용하여 보냄으로 공유하게 된다.

그림 5는 알림반복 기능의 플로차트이다. 사용자가 일정을 입력하는 경우 일정에 대한 알림을 선택하고 알림을 받을 시간을 입력한다. 입력을 받은 날짜와 시간이 되면 Notification을 이용하여 사용자에게 알림을 보내고 사용자가 Notification을 클릭하지 않으면 일정한 시간마다 지속적으로 알림을 보내고 사용자가 Notification을 클릭하면 알림을 멈추고 일정목록 화면으로 이동한다.

4. 일정관리 어플리케이션 구현

본 논문에서 설계하고 구현한 일정관리 어플리케이션은 현재 사용되고 있는 스케줄러 어플리케이션들과 차별



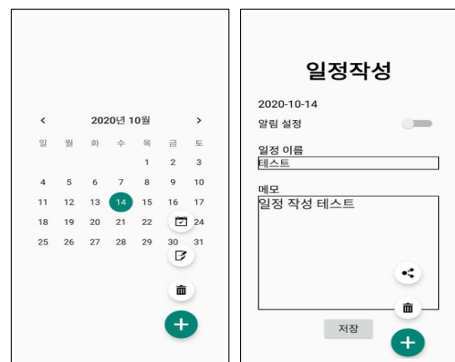
(그림 5) 알림반복 기능 플로차트

(Figure 5) Flowchart of Repeated Notification

화되는 기능들을 설계에 따라 구현함과 동시에 쉽고 편리한 사용이 가능하도록 복잡한 UI로 구성하지 않고 기능구현에 집중하여 구현하였다. 또한, 구현된 어플리케이션에 대하여 사용성 평가를 실시하였다. 추후 상용화되는 경우에는 디자인에 집중하여 UI를 구성할 것이다. 본 논문의 일정관리 어플리케이션은 언제 어디서나 쉽고 편리하게 사용할 수 있도록 안드로이드 스튜디오를 사용하여 모바일 어플리케이션 형태로 구현하였다.

4.1 일정관리 어플리케이션 구현 화면

그림 6은 일정관리 어플리케이션의 메인화면과 일정작성 기능을 구현한 화면이다.

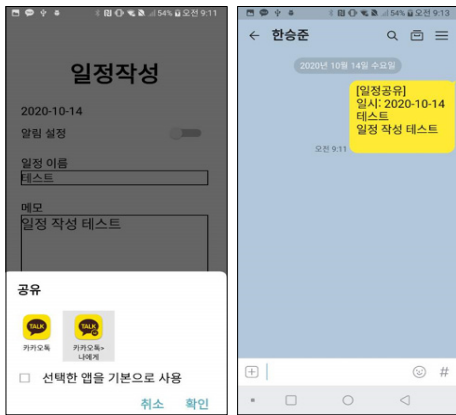


(그림 6) 메인화면과 일정작성 구현화면

(Figure 6) Implementation Screen of Main and Schedule Registration

‘일정작성’은 선택한 날짜와 알림의 설정이 가능한 SwitchButton과 일정의 이름과 메모의 작성이 가능한 EditText로 구성하였다. 일정등록 화면의 하단 저장 버튼을 선택하면 등록된 일정이 내부 저장소에 저장되고 FloatingActionButton을 클릭하면 일정공유 버튼과 일정삭제 버튼이 표시된다.

그림 7은 자신 또는 다른 사용자와 일정을 공유할 필요가 있는 경우 카카오톡을 사용하여 일정을 공유하는 기능을 구현한 화면이다.



(그림 7) 일정공유 구현 화면

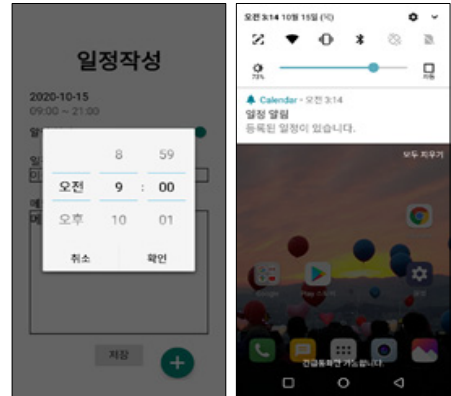
(Figure 7) Implementation Screen of Schedule Sharing Function

일상생활에 많이 사용되는 일정관리 앱들은 여러 명의 사용자가 일정을 공유할 필요성을 가진다. 본 논문의 일정관리 어플리케이션은 국내에서 가장 많은 사용자를 보유한 메신저 앱인 카카오톡을 사용하여 일정을 공유할 수 있도록 구현하였다.

그림 6의 ‘일정작성’ 화면에서 일정을 작성한 후 일정공유 FloatingActionButton을 클릭하여 카카오톡을 사용한 일정을 공유할 수 있다. 그림 7은 카카오톡에서 일정이 공유되었음을 확인할 수 있다.

그림 8은 ‘알림설정’ 기능을 구현한 화면이다. 본 논문의 어플리케이션은 계획을 세우지 않아 일정을 체크하지 못하는 사용자들의 편의성을 위한 어플리케이션으로 알림설정 시 사용자가 알림을 확인할 때까지 지속적으로 알림을 주도록 설계하고 구현하였다.

그림 8의 왼쪽 구현화면인 ‘일정작성’ 기능의 ‘알림설정’의 SwitchButton을 클릭하면 알림 시간을 나타내는 TextView가 나타난다. 여기서 알림시작시간, 알림마침시

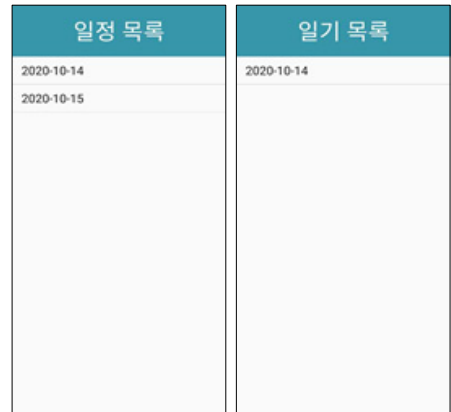


(그림 8) 알림설정 기능 구현화면

(Figure 8) Implementation Screen of Notification Setting Function

간을 설정할 수 있으며 각각의 TextView를 클릭하면 시간 설정이 가능한 Dialog가 표시된다. 알림을 설정하고 나면 그림 8의 오른쪽 구현화면과 같이 Notification을 통해 알림이 설정되었음을 알려주고 Notification을 선택하면 그림 9의 일정 목록으로 이동한다.

그림 9는 작성된 일정과 일기의 목록을 나타내는 기능을 구현한 화면이다.

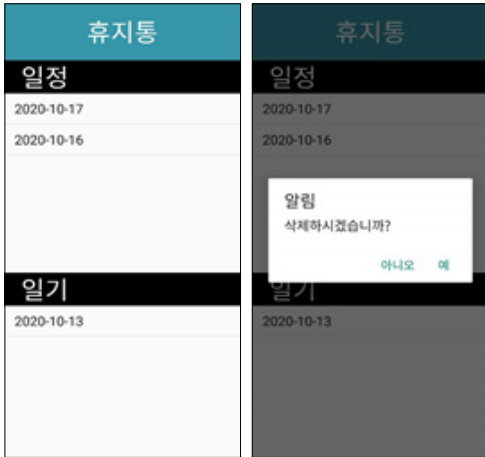


(그림 9) 일정과 일기목록 구현화면

(Figure 9) Implementation Screen of Schedule List and Diary List

작성된 일정이나 일기는 다양한 목적으로 확인을 할 필요가 있으며 이를 위해 목록으로 일정과 일기를 제공하는 기능을 구현하였다. 목록에서 일정이나 일기를 선택하면 선택된 일정 또는 일기의 내용 확인이 가능하다.

그림 10은 일정이나 일기 삭제 시 필요한 파일의 삭제를 줄이기 위해 저장을 하는 기능의 구현화면이다.



(그림 10) 일정삭제와 팝업 삭제 기능 구현화면
(Figure 10) Screen of Schedule Delete and Pop-up Delete Function

일정관리 어플리케이션 사용 시 등록된 일정이나 일기를 잘못 삭제하는 것을 줄이기 위한 장치로 사용자가 등록된 일정이나 일기를 선택하고 삭제를 하더라도 바로 삭제하지 않고 저장소에 저장한 뒤 사용자의 삭제 의사를 다시 확인한 후 삭제하는 기능이다.

FloatingActionButton 버튼을 사용하여 삭제한 일정이나 일기는 바로 삭제되지 않고 저장소로 이동된다. 본 논문의 구현에서는 휴지통을 사용하였다. 휴지통 목록에서 일정 또는 일기 파일을 선택하면 팝업창이 뜨고 이때 ‘예’ 버튼을 선택하면 파일은 완전히 삭제된다.

4.2 일정관리 어플리케이션 구현 코드

그림 11의 코드는 그림 6의 ‘일정작성’ 구현화면에서 저장 버튼을 누르는 경우 txt 파일의 형태로 내부 저장소에 일정을 저장하는 소스코드이다.

title 변수는 일정제목, main 변수는 메모내용 할당을 위한 변수이며 title 변수와 main 변수에 담긴 내용은 scheduleText에 함께 저장된다. date 변수는 메인화면에서 CalendarView를 사용하여 선택한 날짜를 String 형태로 저장하는 변수이다. getFilesDir()은 어플리케이션 내부 저장소의 주소이며 내부 저장소의 Schedule 폴더에 date 변수에 저장된 날짜의 txt 파일로 저장한다.

```
String title = scheduleName.getText().toString();
String main = scheduleContents.getText().toString();
scheduleText = title+"\n"+main;

try{
    File dir = new File(getFilesDir()+"/Schedule");
    File file = new File(dir, date+".txt");
    FileOutputStream fileOutputStream = new FileOutp
utStream(file);
    fileOutputStream.write(scheduleText.getBytes());
    fileOutputStream.close();
}catch (Exception e){
    e.printStackTrace();
}
```

(그림 11) 일정저장 코드
(Figure 11) Code of Schedule Save

그림 12는 일정공유 기능의 구현코드이다. 일정의 공유는 일정이 저장된 내부 저장소 Schedule 폴더에서 선택한 날짜로 저장된 txt 파일에 저장된 파일을 불러와 message 변수에 할당한 후에 그림 7과 같이 공유한다.

```
Intent kakaoTalkIntent = new
Intent(Intent.ACTION_SEND);
kakaoTalkIntent.setType("text/plain");
Intent intent = getIntent();

String date = intent.getStringExtra("date");
File dir = new File(getFilesDir()+"/Schedule");
File diaryFile = new File(dir, date+".txt");

String message = "(일정공유)+"\n"+일시:
"+date+"\n"+intent.getStringExtra("scheduleText");
kakaoTalkIntent.putExtra(Intent.EXTRA_TEXT,
message);
kakaoTalkIntent.setPackage("com.kakao.talk");
startActivity(kakaoTalkIntent);
```

(그림 12) 일정공유 코드
(Figure 12) Code of Schedule Sharing

다른 어플리케이션으로 메시지를 전달할 Intent를 생성하고 setType()을 통해 전송할 데이터의 유형을 text/plain으로 지정한다. Intent의 setPackage()에서 카카오톡으로 데이터가 전송되도록 카카오톡의 패키지명인 “com.kakao.talk”를 설정한다.

그림 13은 일정을 등록한 날짜의 알림설정 시간에 Notification을 통한 알림이 오도록 구현한 코드이다.

```
NotificationCompat.Builder builder = new
NotificationCompat.Builder(context.getApplicationConte
xt(), channelId);

Intent notificationIntent = new
Intent(context.getApplicationContext(),
ScheduleMenu.class);
notificationIntent.setFlags(Intent.FLAG_ACTIVITY_CLEA
R_TOP | Intent.FLAG_ACTIVITY_SINGLE_TOP);

PendingIntent pendingIntent =
PendingIntent.getActivity(context.getApplicationContext
(), 0, notificationIntent,
PendingIntent.FLAG_UPDATE_CURRENT);

builder.setTitle("일정 알림")
.setContentView("등록된 일정이 있습니다.")
.setAutoCancel(true)
.setPriority(NotificationCompat.PRIORITY_HIGH)
.setSmallIcon(R.drawable.alarm)
.setContentIntent(pendingIntent);
notificationManager.notify(0, builder.build());
```

(그림 13) 알림 코드
(Figure 13) Code of Notification

알림구현은 **BroadcastReceiver**를 상속받는 **AlarmReceiver** 클래스를 생성하여 해당 클래스에 **Notification**을 생성한다.

그림 14는 알림 실행 구현에 대한 소스코드이다.

```
Calendar calendar = Calendar.getInstance();

calendar.set(Calendar.YEAR, Integer.parseInt(year));
calendar.set(Calendar.MONTH,
Integer.parseInt(month)-1);
calendar.set(Calendar.DAY_OF_MONTH,
Integer.parseInt(day));
calendar.set(Calendar.HOUR_OF_DAY, alarmHour);
calendar.set(Calendar.MINUTE, alarmMinute);
calendar.set(Calendar.SECOND, 0);

int CODE = Integer.parseInt(date.replaceAll("-", ""));
Intent intent = new Intent(getApplicationContext(),
AlarmReceiver.class);

PendingIntent pendingIntent =
PendingIntent.getBroadcast(getApplicationContext(),
CODE, intent,
PendingIntent.FLAG_CANCEL_CURRENT);

AlarmManager alarmManager = (AlarmManager)
getSystemService(ALARM_SERVICE);
alarmManager.setAndAllowWhileIdle(AlarmManager.RT
C_WAKEUP, calendar.getTimeInMillis(),
pendingIntent);
```

(그림 14) 알림 실행 코드
(Figure 14) Code of Notification Practice

알림의 실행은 **Calendar**에 선택한 날짜와 시간을 설정 한 뒤 **AlarmManager**를 통해 정해진 날짜와 시간이 되면 **AlarmReceiver**를 실행시킨다.

그림 15는 일정이나 일기 파일을 삭제하는 경우 바로 삭제하지 않고 저장을 하는 기능에 대한 구현코드이다. 작성된 일정과 일기는 각각의 화면에 구현된 **FloatingActionButton**을 사용하여 삭제할 수 있다. 이때 필요한 일정이나 일기 파일이 삭제되는 것을 최대한 줄이기 위하여 삭제된 일정과 일기는 바로 삭제하지 않고 특정한 폴더로 이동시킨 후 사용자의 확인을 거쳐 삭제한다. 본 논문에서는 휴지통으로 이동한다.

```
Intent intent = getIntent();

File dir = new File(getFilesDir()+"/Schedule");
File scheduleFile = new File(dir,
intent.getStringExtra("date")+".txt");
File deleteFile = new
File(getFilesDir()+"/Delete/Schedule",
intent.getStringExtra("date")+".txt");

if(scheduleFile.exists()){
scheduleFile.renameTo(deleteFile);
}
```

(그림 15) 삭제된 일정과 일기 저장 구현
(Figure 15) Storage Implementation Code of Deleted Schedule and Diary

그림 16은 저장된 일정이나 일기 내용의 확인이 필요한 경우를 위해 일기목록과 일정 목록을 구현한 코드이다.

```
listView = findViewById(R.id.listView);

final File dir = new File(getFilesDir()+"/Schedule");
File file[] = dir.listFiles();

ArrayList<String> scheduleList = new ArrayList<>();

for(int i = 0; i< file.length; i++){
String scheduleName = file(i).getName();
if(scheduleName.contains(".txt")){
scheduleName = scheduleName.substring(0,
scheduleName.lastIndexOf('.'));
scheduleList.add(scheduleName);
}
}

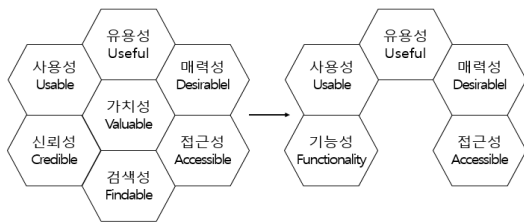
ArrayAdapter<String> adapter = new
ArrayAdapter<>(this,
android.R.layout.simple_list_item_1, scheduleList);
listView.setAdapter(adapter);
```

(그림 16) 일정목록과 일기목록 코드
(Figure 16) Code of Schedule List and Diary List

일정목록은 내부 저장소의 Schedule 폴더 내의 파일들의 이름을 scheduleList에 저장하고 해당 리스트를 ListView를 통해 화면에 나타나도록 구현하였다.

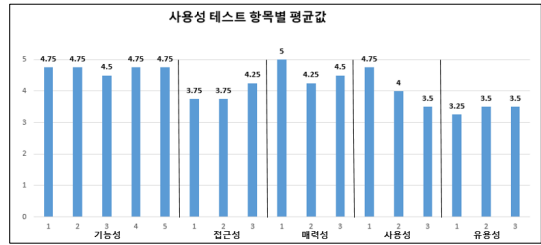
4.3 어플리케이션 사용성 평가

구현된 어플리케이션이 설계한 기능들을 제대로 구현하였는지와 유용한 어플리케이션인지를 확인하기 위해서는 구현된 모바일 어플리케이션의 사용성을 평가하는 것이 필요하다[11]. 디지털콘텐츠의 사용성을 평가하는 방법으로는 휴리스틱 평가, 인지적 시찰법, 기본 사용자 평가, 실험실 평가, 현장 평가 방법들이 주로 사용되고 있다[12-13]. 본 논문에서는 휴리스틱 평가 방법을 사용하여 사용성 평가를 진행하였다. 휴리스틱 평가는 2-5명의 소수의 UI/UX 전문가들이 가이드라인에 따라 사용성을 평가하는 것으로 비교적 적은 예산을 사용하여 짧은 기간에 평가가 가능한 방법이다[12]. 사용성 평가 기준으로는 피터 모빌의 허니콤 모형의 7가지 사용성 원칙을 기반으로 하였다. 허니콤 모형의 7가지 사용성 원칙은 유용성, 사용성, 매력성, 가치성, 신뢰성, 접근성, 검색성이다[14]. 본 논문에서는 구현된 스케줄 어플리케이션이 설계시 기능을 구현하였는지와 사용성을 위주로 평가하기 위하여 기존의 허니콤 모형을 재구성하였다. 허니콤 모형의 가치성, 신뢰성, 검색성은 제외하고 유용성, 사용성, 매력성, 접근성과 어플리케이션이 가진 기능성을 평가하기 위하여 기능성 항목을 추가하여 평가하였다. 기존 허니콤 모형과 본 논문의 사용성과 기능성 평가를 위하여 새로이 구성한 모형은 그림 17과 같다.



(그림 17) 재구성한 평가 항목
(Figure 17) Reconfigured Evaluation Item

평가는 IT 종사 경력 10년 이상인 4인의 전문가가 참여하였고 그림 18은 평가결과를 차트로 표현한 것이다.



(그림 18) 사용성 평가 결과
(Figure 18) Result of Usability Test

평가결과는 5는 매우 그렇다, 4는 그렇다, 3은 보통이다, 2는 그렇지 않다, 1은 매우 그렇지 않음을 나타낸다. 평가결과를 살펴보면 기능성 항목은 1개가 4.5이고 4개는 4.75로 구현된 스케줄 어플리케이션은 설계한 기능인 일정등록, 일기작성, 일정삭제, 일정등록 시 카카오톡 공유, 알림설정 기능을 모두 잘 구현한 것으로 평가하였다. 매력성 평가결과도 3개 항목에 대해 모두 4.25이상으로 필요한 기능만으로 화면을 심플하게 디자인함으로 평가결과가 높음을 알 수 있다. 평가결과가 낮은 부분은 접근성 부분과 유용성 부분으로 구현한 스케줄 어플리케이션이 도움말 등을 제공하지 않아 사용자가 오류를 범하는 경우가 많고 시중에서 사용되고 있는 다른 스케줄러 앱과 비슷한 편리성을 가지는 것으로 평가하였다.

표 3은 본 논문의 스케줄 어플리케이션의 사용성 평가를 위해 설계한 항목들을 나타낸 표이다.

(표 3) 사용성 테스트 항목
(Table 3) Usability Test Items

범주	Task	평균 값
기능성	1. 일정의 등록이 가능한가?	4.75
	2. 일기의 작성이 가능한가?	4.75
	3. 일정의 삭제가 가능한가?	4.5
	4. 일정등록 시 카카오톡과 공유가 가능한가?	4.75
	5. 알림의 설정이 가능한가?	4.75
접근성	1. 앱은 사용자가 쉽게 오류를 범하지 않도록 구현되었는가?	3.75
	2. 조작방법은 편리한가?	3.75
	3. 버튼의 크기와 버튼 사이의 간격은 적당한가?	4.25
매력성	1. 디자인은 심플하게 구성되었는가?	5
	2. 디자인은 조화롭게 구성되었는가?	4.25
	3. 아이콘들의 사이즈는 적당한가?	4.5
사용성	1. 앱은 필요한 기능만으로 구성되었는가?	4.75
	2. 기능은 사용하기에 쉽게 구성되었는가?	4
	3. 쉽게 원하는 기능을 사용할 수 있는가?	3.5
유용성	1. 서비스는 다른 스케줄러 앱보다 편리한가?	3.25
	2. 서비스는 다른 스케줄러 앱보다 쉽게 사용 가능한가?	3.5
	3. 서비스는 다른 스케줄러 앱보다 시간이 절약되는가?	3.5

5. 결 론

본 논문에서는 누구나 쉽게 사용할 수 있고 일상생활에 도움이 되는 일정관리 어플리케이션을 설계하고 이를 구현하였다. 본 논문에서 설계하고 구현한 스케줄러 앱은 일정작성 후 알림을 받을 수 있고 일기를 작성함으로 하루를 평가하는 기능, 메시지를 이용하여 다른 사용자와 일정을 공유하는 기능, 일정이나 일기 삭제 시 바로 삭제하지 않고 저장하는 기능 등을 설계하고 이를 안드로이드 스튜디오를 이용하여 어플리케이션을 구현하였다.

최근 일과가 복잡하고 건강증이 심한 현대인들이 본 논문에서 설계하고 구현한 스케줄러 어플리케이션을 이용하여 일정을 효율적으로 관리하여 시간적, 경제적 측면에서 도움이 되기를 기대한다.

참고문헌(Reference)

- [1] Sung-Joong Kim, JiHoon Yoo, HaengRok Oh, Dongil Shin, DongKyo Shin, “A Study on the Operation Concept of Cyber Warfare Execution Procedures”, *Journal of Internet Computing and Services*, Vol. 21, No. 2, pp. 73-80, Apr. 2020.
<https://doi.org/10.7472/jksii.2020.21.2.73>
- [2] Sanghyuk Han, “2019 Survey on usage behavior of broadcasting media”, Korea Communications Commission, pp. 7-9, 2019.12.
- [3] Yin-Hee Lee, “Influence of Smartphone Use Tendency and Addiction on Middle Aged People”, Graduate School of Daegu University, 2015.
- [4] Gyu-Sung Ham, Own-jeong Seo, Hoill Jung, Su-Chong Joo, “Implementation of Dynamic Situation Authentication System for Accessing Medical Information”, *Journal of Internet Computing and Services*, Vol. 19, No. 6, pp. 31-40, Dec. 2018
<https://doi.org/10.7472/jksii.2018.19.6.31>
- [5] Changsu Lee, Youngjoong Ko, “A Simultaneous Recognition Method of Named Entities and Objects for the Language Understanding Module of a Dialogue-based Private Secretary System”, *Journal of KISS : Software and Applications*, Vol. 41, No. 4, pp. 295-301, 2014.
<https://www.dbpia.co.kr/Journal/articleDetail?nodeId=NODE02407109>
- [6] Sang-A Park, Ji-Won Oh, In-Sik Hong, Yunyoung Nam, “Development of a Blocks Recognition Application for Children’s Education using a Smartphone Camera”, *Journal of Internet Computing and Services*, Vol. 20, No. 4, pp. 29-38, Aug. 2019
<https://doi.org/10.7472/jksii.2019.20.4.29>
- [7] Moonyoung Kang, Youngsu Ji, Hakrae Kang, “A Study on the Effects of Mobile Native Advertising based on Social Media on Using Intention”, *The Korean Journal of Advertising*, Vol. 27. No. 2, pp. 7-28, 2016.
- [8] Ye-Jong Kim, Su-An Bu, In-Soon Heo, Gyun Heo, “An Exploring Study on the Difference of Smart-phone Usage by Generation”, *Journal of Fisheries and Marine Sciences Education*, Vol. 29, No. 6, pp. 1686-1694, 2017.
<https://doi.org/10.13000/JFMSE.2017.29.6.1686>
- [9] Yunho Lee, “An SMS Notarization System Using Smartphones”, *Journal of Internet Computing and Services*, Vol. 19, No. 5, pp. 13-19, Oct. 2018.
<https://doi.org/10.7472/jksii.2018.19.5.13>
- [10] Rankey,
[https://www.rankey.com/blog/blog.php?type=column&sub_type=trend&no=442\(2014.11.12.\)](https://www.rankey.com/blog/blog.php?type=column&sub_type=trend&no=442(2014.11.12.))
- [11] Jung-Gil Cho, “Evaluating Mobile Apps with Usability Testing Approach”, *Journal of Multimedia Services Convergent with Art, Humanities, and Sociology*, Vol. 7, No. 11, pp. 857-865, 2017.
- [12] Hyebin Jeon, Haein Kim, Hyeyoung Ko, “A Study on Efficiency Usability Evaluation Method for Smart Learning App”, *The Journal of Image and Cultural Contents*, Vol. 19, pp. 443-464, 2020.
- [13] Hee-Sun Kim, “Usability Improvement Method of Shopping Mall based on Smartphone”, *The Journal of the Korea Contents Association*, Vol. 11, No. 10, pp. 77-84, 2011.
<http://dx.doi.org/10.5392/JKCA.2011.11.10.077>
- [14] So-Hee Yoon, Seung-In Kim, “Evaluation of Usability for Mobile Streaming Application - Mainly with Analysis on ‘Youtube’ and ‘Youku’ -”, *Journal of Digital Design*, Vol. 15, No. 3, pp. 705-713, 2015.

◎ 저 자 소개 ◎



박 은 주(Eunju Park)

1993년 안동대학교 전산통계학과 (이학사)
2001년 안동대학교 대학원 컴퓨터공학과 (공학석사)
2016년 안동대학교 대학원 정보통신공학과 멀티미디어 공학 전공 (공학박사)
2000년~2016년 안동대학교 강의
2016년~2017년 안동대학교 멀티미디어 공학과 강의전담교원
2019년~현재: 안동대학교 SW융합교육원 교수
관심분야 : 웹 접근성, 모바일 접근성, 유니버설 디자인, WAI-ARIA
E-mail : eunju@anu.ac.kr



한 승 준(Seungjun Han)

2021년 안동대학교 멀티미디어공학과(공학사)
2021~현재 경북대학교 대학원 컴퓨터학부 석사과정 재학 중
관심분야: 머신러닝, 딥러닝, 빅데이터
E-mail: dk07564@gmail.com



윤 지 민(Jimin Yoon)

2015년~현재 안동대학교 멀티미디어공학과 재학 중
관심분야 : 빅데이터, 데이터처리
E-mail : dodorofo@naver.com



임 한 규(Hankyu Lim)

1981년 경북대학교 전자공학과 (공학사)
1984년 연세대학교 컴퓨터공학과 (공학석사)
1997년 성균관대학교 컴퓨터공학과 (공학박사)
1981년~1982년 대한주택공사
1982년~1986년 한국전자통신연구소 위성통신연구실
1986년~1994년 한국IBM 소프트웨어연구소
1994년~1998년 한서대학교 전산정보학과 교수
1998년~현재 안동대학교 멀티미디어공학과 교수
관심분야 : 멀티미디어 콘텐츠, 웹 애플리케이션, NLP
E-mail : hklim@anu.ac.kr