

반복된 일상을 도와주는 스마트 비서

변지훈, 이예슬, 박누리, 이지원
세종대학교 컴퓨터공학과
e-mail:bjh933@naver.com
김미정
한국산업기술대학교 융합디자인학과

Smart secretary to help your repeated everyday life

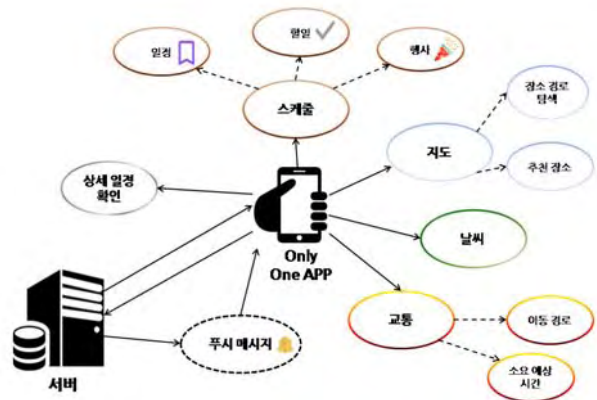
Ji-Hoon Byun, Ye-Seul Lee, Nu-ri Park, Ji-Won Lee
Sejong Univ. Computer Engineering
e-mail : bjh933@naver.com
Mi-Jung Kim
Korea Polytechnic Univ. Fusion Design

요 약

스마트 비서 어플리케이션을 통해 하나의 앱에서 일정 관리와 날씨, 지도 및 교통 정보를 제공한다. 이를 통해 매일 반복되는 일상을 소화하는 데에 필요한 여러 종류의 정보를 얻기 위해 다양한 앱을 실행시켜야 하는 현대인들의 불편함을 없애고 편리함과 일의 효율성을 높인다.

1. 서론

대다수의 현대인들은 매일 같은 일상을 반복한다. 매일 같은 시간에 일어나, 동일한 교통수단을 이용하고 자신에게 주어진 책임을 다하다가 자신의 보금자리로 돌아가는 일상을 반복한다. 우리는 이미 우리의 반복적인 일상에 도움이 되는 모든 정보를 스마트폰으로 확인하고 있다. 그러나 각각의 앱 (알람 앱, 교통 앱, 기상정보 앱, 달력 앱)을 따로 실행하고 검색하여 이용한다. '매일 반복되는 일상에 대한 정보를 하나의 앱에서 좀 더 편리하게 보여줄 수 있을까?'라는 호기심에서 이번 프로젝트는 시작되었다.



<그림1. 어플리케이션 컨셉>

2. 프로젝트 내용

단순히 일정을 관리하는 것이 아닌, 일정을 소화하는 데에 필요한 모든 정보(그 날의 날씨, 교통, 장소)를 사용자에게 제공하여 편리성과 효율성을 제공한다. 일정을 등록할 때 시간과 장소를 등록하면, 그 시간의 기상 정보, 그 장소까지의 경로 및 총 소요 시간, 목적지 주변의 추천 장소 기능을 제공한다.

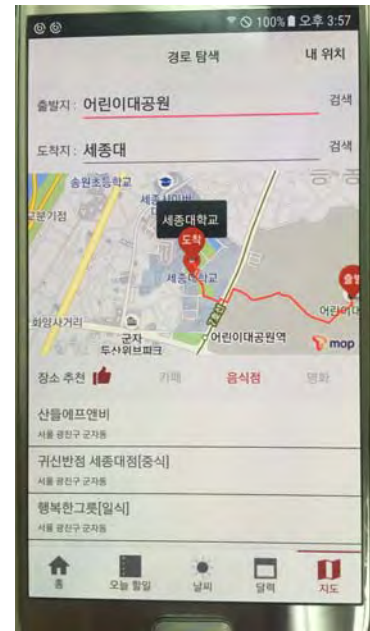
3. 작품의 특징

스케줄 관리의 편리성

- 사용자의 스케줄을 일정, 할일, 행사 3개의 항목으로 분할하여 목적에 맞는 스케줄을 생성하여 편리한 관리가 가능하다. 일정 항목에는 사용자의 구체적인 일정을 기록하고, 할일 항목에는 사용자가 해야 할 일을 저장하고 마감일까지 남은 기간을 알려준다. 행사 항목에는 지인의 생일 같은 경조사 정보를 저장한다.



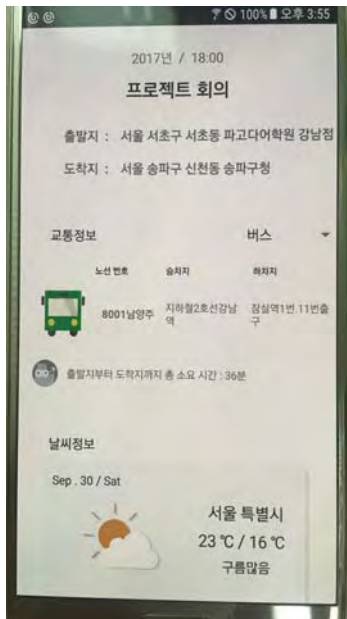
<그림2. App 홈 화면>



<그림4. App 내 장소 검색 화면>

일정과 함께 날씨, 교통 정보 동시에 확인 가능

- 스케줄 확인 화면에 등록된 일정의 일정 카드를 클릭하면, 일정뿐만 아니라 날씨와 교통, 목적지 까지의 총 소요 시간을 동시에 확인이 가능하다.



<그림3. 상세 일정 화면>

푸시 메시지

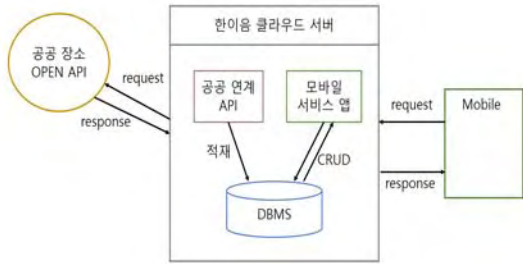
- 등록한 일정과 관련된 정보나 자기 전에 내일의 스케줄을 확인하는 푸시 메시지를 보여줌으로써, 스케줄 관리를 세심하게 도와준다.

지도 & GPS

- 지도와 GPS 기능을 통한 현재 위치 확인, 장소 및 경로 탐색이 가능하다.

4. 유사 제품과의 기능적/기술적 차별성

- 하나의 앱으로 여러 앱에서 각각 제공하는 정보를 한번에 얻을 수 있다.
- 일정 확인 및 등록 푸시 메시지가 자기 전(사용자 설정 시간)에 뜨도록 설정할 수 있다.
- 일정에 주소를 등록하면 등록된 주소 주변에 카테고리마다 장소를 추천해줌으로써 장소 탐색 시간 단축 및 효율성 증진 효과가 있다.
- 일정을 확인할 때, 날씨와 교통정보도 같이 제공하여 편리성을 제공한다.
- 출발지에서 목적지까지의 총 소요 시간을 제공하며, 약속에 늦지 않게 소요 시간을 기준으로 한 시간 전에 푸시 메시지를 통해 사용자에게 알린다. (여기서 총 소요 시간이란 출발지에서 목적지까지 걸리는 총 소요 시간으로, 예를 들어 집에서 정류장까지 걸어서 걸리는 시간 + 정류장에서 교통수단을 통해 하차지까지 가는 데 걸리는 시간 + 하차지에서 목적지까지 걸어서 걸리는 시간을 말한다.)



<그림5. 시스템 구성도>

JAVA 기반의 안드로이드 스튜디오를 사용하여 모바일 어플리케이션을 개발하며 이 프로그램을 통해 사용자 인터페이스를 제공한다. RESTful API의 GET, PUT, POST, DELETE 방식을 통해 모바일 어플리케이션과 서버간의 데이터 송수신이 이루어진다. 이 통신을 통해 일정 관리와 공공데이터, 푸시 메시지를 받아온다.

서버와 데이터베이스와의 연결 및 데이터 교환을 위해서 JDBC를 사용한다. 이 연결을 통해 모바일 앱에서 보내온 일정 생성/수정/삭제 데이터를 서버가 데이터베이스에 저장하고, 앱에서 요청 시에 다시 이 정보를 꺼내 모바일로 보낸다.

또한 모바일에서 푸시 메시지를 띄우기 위해 서버가 외부에서 공공 데이터를 가져오거나, 데이터베이스에서 일정 정보를 가져와 모바일로 데이터를 보낸다.

index	function	method	path	request param	ex
1	일정 목록 조회	GET	/master/schedule/list	userid,date	-?userid=abcd&date=2017-08-09
2	일정 목록 조회	GET	/master/schedule/list	userid, date	-?userid=abcd&date=2017-09-09
3	행사 목록 조회	GET	/master/event/list	userid,date	-?userid=abcd&date=2017-08-09
4	일정 저장	PUT	/master/savechedule	userid, title, ampm, startHour, startMinute, s	-?userid=abcd&title=운동하기&m=a&start
5	일정 저장	PUT	/master/saveToDo	userid, title, startDate, endDate, fix, destinat	-?userid=abcd&title=운동하기&startDate=2017-0
6	행사 저장	PUT	/master/saveevent	userid, icon, title, startTime, startDate, endD	-?userid=abcd&icon=bird&title=전구생들&se
7	오늘 날씨정보	GET	/master/weather/today	date, time	-?date=2017-08-22&time=13:00:00
8	생일 시수	GET	/master/birthday/index	date, season, location	-?date=2017-08-09&season=Summer&location
9	일정에 방문할 곳 찾기	GET	/master/calcchedule	userid, startDate, endDate	-?userid=abcd&startDate=2017-08-09&endDate
10	장소 추천기능	GET	/master/location/recommend	gender, destination	-?gender=female&destination=Gangnam

<그림6. REST 설계>

```

Response VO
private String userid = "";
private Timestamp date = new Timestamp(System.currentTimeMillis());

private List<Schedule> schedules = new ArrayList<Schedule>();
3:27&endDate=2017-08-27&fix=true&destination=Gangnam&memo=한시간 하기
public List<Todo> getSchedules() {
    return todos;
}
2017-08-09
public void setSchedules(List<Schedule> todos) {
    this.todos = todos;
}

public String getUserid() {
    return userid;
}

public void setUserid(String userid) {
    this.userid = userid;
}

public Timestamp getDate() {
    return date;
}

public void setDate(Timestamp date) {
    this.date = date;
}
    
```

<그림7. Response VO 설계>

각 기능들의 데이터를 모바일 앱과 서버가 주고받기 위해 REST 방식으로 설계하여 그에 따른 Response Value Object 코드를 짰다. 이를 통해 데이터의 효율적인 요청과 응답이 가능하다.



<그림8. 기능 흐름도>

1) Java database connectivity의 약자로, 자바 프로그램 안에서 SQL을 실행하기 위해 데이터베이스를 연결해주는 응용프로그램 인터페이스를 말한다.

5. 주요 적용 기술

- 날씨 정보 공공데이터와 SK플래닛의 날씨 정보 제공 API를 통해 최고/최저 기온, 강수 확률, 미세먼지, 불쾌지수 정보를 가져온다.



<그림9. 날씨 정보 화면>

- 서울시의 교통 정보를 가져오기 위해 정류소 정보 조회, 대중교통 환승 정보 조회 API를 사용하고 SK플래닛의 Tmap API를 사용하여 버스 정류장 혹은 역까지의 거리와 소요시간을 가져와 이를 조합하여 출발지에서 목적지까지의 이동 경로와 총 예상 소요 시간을 제공한다.

- REST 서비스를 통해 앱과 서버의 통신을 통해 일정을 저장하고, 수정, 삭제(CRUD)가 가능하다.

※ **REST** : URI와 HTTP 메소드를 이용해 객체화된 서비스에 접근하는 것으로, 직관적인 URI를 이용해서 오브젝트의 멤버를 추적하는 것처럼 리소스에 접근하여 HTTP 메소드를 이용해 그것을 조작하는 것이다. HTTP에서 제공하는 GET, PUT, POST, DELETE 4개의 메소드를 이용해서 서비스를 제공한다.

METHOD	역할
POST	POST를 통해 해당 URI를 요청하면 리소스를 생성합니다.
GET	GET를 통해 해당 리소스를 조회합니다. 리소스를 조회하고 해당 도큐먼트에 대한 자세한 정보를 가져옵니다.
PUT	PUT를 통해 해당 리소스를 수정합니다.
DELETE	DELETE를 통해 리소스를 삭제합니다.

<그림10. HTTP 기본 메소드 4가지>

- 모바일에서 REST 서비스를 보다 편리하게 제공하도록 도와주는 Retrofit 기술을 적용하였다.

※ **Retrofit** : Square라는 회사에서 만든 REST API통신을 위한 Android/JAVA 라이브러리로, 클라이언트와 서버간 통신을 AsyncTask, HttpURLConnection을 사용해서 직접 네트워크 통신을 구현할 때 고려해야 했던 많은 작업들을 보다 편리하게 만들어 줌.

- 달력UI 구현에 오픈소스를 활용하였다.



<그림11. 달력 화면>

6. 작품의 기대효과

• 자기 전(사용자 설정 가능) 일정 팝업 창을 띄워줌으로써, 꼼꼼한 스케줄 관리와 규칙적인 습관을 가능케 한다.

• 여러 개의 정보를 하나로 모아 보여주는 편리성을 제공한다.

-> 일정을 확인할 때, 날씨와 교통정보 동시에 확인 가능, 장소 검색 가능

• 주변 장소 추천 기능으로 더 편리하고 적합한 장소 선택 가능하다.

• 출발지부터 목적지까지 총 예상 소요시간 제공으로, 현대인들의 시간 절약, 편리성과 효율성 제공한다.