

XML 파싱기법을 이용한 안드로이드 기반의 종합 날씨 정보 앱 설계 및 구현

이진욱*, 윤형수**, 하수철***

요약

본 논문은 XML(eXtensible Markup Language) 파싱(parsing)기법을 이용한 안드로이드 기반의 날씨 정보 어플리케이션의 설계 및 구현에 관한 사항이다.

GPS 위치 프로바이더와 XML 파싱기법 이용하여 현재 위치에 따라 날씨 정보가 변하게 된다. 또한 알람 시스템을 이용하여 어플리케이션의 접속과 종료에 상관없이 알람 이벤트가 발생 될 때의 시간을 기준으로 날씨 정보를 음성(English)으로 출력한다. 모든 어플리케이션에서 사용자 인터페이스는 앱의 첫 인상이 중요하기 때문에 복잡하지 않고, 단순한 방법으로 사용자 인터페이스 설계를 하였다.

Design and Implementation of Android-based Total Weather Information Application using XML Parsing Techniques

Jin-Wook Lee*, Hyoung-Soo Yyeon**, Soo-Cheol Ha***

Abstract

This paper presents a design and implementation of Android-base Total Weather Information using XML(eXtensible Markup Language) techniques.

Weather Information are changed using GPS location provider and XML parsing techniques according to the current location, This application alarms the weather information as the English voices when some alarm events appear regardless of the application access or finish. We design the user interface simple because the first impression is important in all applications.

Keywords : Weather Information, GPS Location Provider, XML Parsing techniques, Alarm System

1. 서론

스마트폰의 급속한 기술 발전에 의해 이전의

휴대 전화와 달리 이동 중 인터넷 통신과 팩스 전송 등 어플리케이션의 활용이 가능한 지능형 단말기(스마트폰)가 출시되면서 스마트폰은 거의 모든 휴대용 디지털 기기를 한 곳에 모은 휴대 전화로 인식되고 있다. 이처럼 지능형 단말기가 대중화됨에 따라 스마트폰을 활용하는 어플리케이션이 대두되고 있는 가운데 사용자에게 눈길을 사로잡을 수 있는 창의적이면서 독창적인 어플리케이션이 필요하게 되었다.

본 논문은 XML(eXtensible Markup Language) 파싱(parsing)기법을 이용한 안드로이드 기반의 날씨 정보 어플리케이션의 설계 및 구현에 관한

※ 제일저자(First Author) 이진욱
접수일자 : 2011년 월 일, 심사완료 2011년 월 일
* 대전대학교 컴퓨터공학과
kdhgo1104@naver.com
** 대전대학교 컴퓨터공학과
ni_kke@naver.com
*** 대전대학교 컴퓨터공학과 교수(교신저자)

사항이다.

본 논문에서는 편리하고 사용하기 쉬운 어플리케이션 개발을 목적으로 한다. 따라서 사용자가 자신이 위치한 지역의 날씨를 직접 찾아서 사용하는 것보다 GPS 위치 프로바이더를 이용하여 사용자 자신이 위치한 지역의 위치 정보를 받아와서 그 위치에 맞는 날씨 정보를 추출 및 출력할 수 있도록 편리한 기능을 제공하고 있다. 또한 알람 시스템을 이용하여 알람 인텐트가 동작할 때의 날짜와 시간대의 날씨 정보를 음성(English)로 출력하는 기능이 포함된다. 이러한 편의성을 가진 기능들은 앱의 중점 요소로 작용하기 때문에 사용자 인터페이스 또한 중요한 시각적 요소로 작용한다. 모든 어플리케이션에서 사용자 인터페이스는 앱의 첫 인상을 강조하기 때문에 복잡하지 않고, 단순한 방법으로 사용자 인터페이스 설계를 진행하였다.

2. 관련연구

2.1 안드로이드(Android) OS

안드로이드 OS는 가장 최근의 모바일 운영체제로서 비독점 개방형 플랫폼으로 OS의 모든 소스 코드와 관련 개발 도구가 무료로 배포되고 개발자들이 안드로이드 기반의 프로젝트 개발을 하기 위해 유용하게 사용되고 있다. 안드로이드는 공개 운영체제인 Linux 기반의 커널을 사용하며, 운영 체제의 핵심이라고 할 수 있는 것으로 하드웨어와 프로세스의 보안 관리, 하드웨어 레이어 추상화 시스템 자원 관리 등의 역할을 담당하는 부분이다. 또한 고수준의 언어인 Java를 공식 언어로 사용하므로 생산성이 높고, 하드웨어를 직접 제어하지 않아도 하드웨어 추상 레이어를 통해 간접적이고도 직관적으로 제어할 수 있는 장점이 있다[6,10].

2.2 안드로이드 어플리케이션

안드로이드 어플리케이션은 각각의 컴포넌트들이 느슨하게 결합된 형태로 구성되며, 서로 상호 작용하는 방식을 프로젝트 매니페스트(Project Manifest)를 통해 전달하고 결합되는 구조를 가지고 있다. 안드로이드 어플리케이션 컴포넌트는 서로 의존하지 않고 독립적으로 구현되기 때

문에 콘텐츠 프로바이더(Content Providers)나 서비스와 같은 개별적인 요소를 다른 어플리케이션과 공유하거나 주고받을 수 있는 장점이 있다.

2.3 기존 유사 앱(어제 오늘)

아이비소프트(IB.SOFT)에서 서비스 중인 ‘어제 오늘’이라는 날씨 정보 앱으로 기본 온도와 체감온도, 최저 최고 온도 등으로 메뉴를 나타내고 있으며 날씨 이야기로 이 앱을 사용하는 유저들과의 이야기를 나누는 공간이 있다. 또한 이 앱은 지역 설정을 하지 않을 경우 서울의 날씨로 출력되기 때문에 지역 설정을 통해 기본적으로 자신이 보고자 하는 지역 설정을 (그림 1)과 같이 해주어야 한다.

이러한 요소들을 고려하여 본 논문의 날씨 정보 앱은 GPS시스템을 탑재하여 지역 설정 없이도 자신이 위치한 지역에 날씨를 동 단위 및 시간단위로 출력할 수 있도록 설계하였다.



(그림 1) ‘어제 오늘’ 앱의 지역설정



(그림 2) ‘어제 오늘’ 앱의 메인화면

2.4 기존 유사 앱(Olleh 날씨)

스마트폰 사용자들이 주로 찾는 날씨 앱으로 아래 (그림 3)과 같은 화면을 나타낸다. 3개의 지역으로 구분하여 데이터를 저장할 수 있는 기능이 있다. 일반적으로 다른 날씨 앱과 기능들은 비슷하나, 사용자 인터페이스 면이 좋다.

(그림 3)과 같이 출퇴근 예보, 시간 별 예보, 주간 예보, 전국 날씨 등으로 날씨 정보의 기능을 4가지로 나눌 수 있다.

이러한 기존 앱을 분석하여 기존의 유사 앱과 다른 기능을 중심으로 연구하여 알람 시스템을 탑재하여 알람 이벤트가 작동할 때 마다 음성으로 날씨 정보를 동 단위 및 시간 단위로 제공하도록 설계하였다.



(그림 3) 'Olleh 날씨' 앱의 메인화면

2.5 매니페스트(Manifest)

모든 안드로이드 프로젝트는 AndroidManifest.xml의 매니페스트(Manifest)파일을 포함하고 있는데, 이 파일은 어플리케이션의 속성과 버전, 컴포넌트의 구조 및 메타데이터를 정의하고 있다. 매니페스트는 어플리케이션을 구성하는 각각의 액티비티, 콘텐츠 프로바이더, 브로드캐스트 리시버, 서비스 컴포넌트에 대한 노드를 포함하고 있으며, 인텐트 필터(Intent Filter)태그와 권한(Permissions)태그를 사용하여 한 컴포넌트가 다른 컴포넌트나 어플리케이션과의 상호 작용하는 방식을 결정한다.

2.6 콘텐츠 프로바이더(Content Providers)

콘텐츠 프로바이더(Content Providers)는 어플리케이션 간에 데이터를 공유할 수 있는 공간이며 여러 어플리케이션이 사용하는 데이터베이스를 관리하고 공유하기 위해 사용된다. 각각의 어플리케이션은 독자적인 데이터베이스를 구축하지만 이 데이터베이스에 다른 어플리케이션이 접근할 수 있도록 콘텐츠 프로바이더를 구성할 수 있다[5,6].

2.7 위치 기반 서비스(LBS: LocationBased Services)

최근에 출시되고 있는 스마트폰은 차량 내비게이션 장치를 비롯해 휴대전화 그리고 디지털 카메라에 이르기까지 위치 정보를 사용하는 것이 보편화 되어 있고, (그림 4)와 같이 2007년부터 2012년까지의 LBS시장 규모를 통해 각 년도마다 발전된 GPS 탑재 비율을 확인 할 수 있다[3].



(그림 4) LBS 시장 규모

위치 기반 서비스(LBS)는 GPS와 구글의 셀 기반 위치 찾기(Cell-Based Locating)기술과 같이 여러 개의 기술을 엮어서 제공한다. 일반적으로 흔히 지역을 나타낼 때 사용하는 주소도 이용할 수 있도록 하기 위해 순방향 지오코딩(Forward Geocoding)과 역방향 지오코딩(Backward Geocoding)을 지원한다.

2.8 순방향 지오코딩(Forward Geocoding)

순방향 지오코딩(Forward Geocoding)은 주어진 주소 정보를 기반으로 지도상의 경위도 좌표를 결정한다.

일반적으로 경위도 값을 얻어오는 지오코딩 조

회는 Geocoder.getFromLocationName 메소드를 통해 수행할 수 있다. 아래 그림을 통하여 순방향 지오코딩 조회를 확인한다.

```
try {
    List<Address> arrAddress =
    geocoder.getFromLocation(dblLatitude, dblLongitude, 1);
    StringBuilder strBuilder = new StringBuilder();
```

(그림 5) 순방향 지오코딩

2.9 역방향 지오코딩(Backward Geocoding)

역방향 지오코딩(Backward Geocoding)은 경위도 좌표로 구성된 위치 정보를 기반으로 일반적으로 인지 가능한 도로 기반의 주소로 변환하는 것을 뜻한다. 역방향 지오코딩에 의한 도로 기반의 주소는 Geocoder.getFromLocation 메소드에 경위도 좌표를 전달하는 것으로 아래 그림과 같은 방법을 통하여 조회할 수 있다.

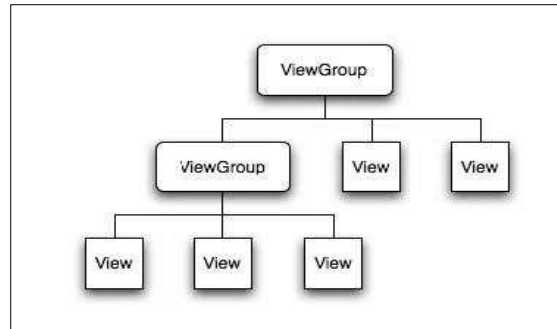
```
Location location1 =
locationManager.getLastKnownLocation(LocationManager.GPS_PROVIDER);
Geocoder geocoder = new Geocoder(this, Locale.getDefault());
List<Address> arrAddress = null;
try{
    arrAddress =
        geocoder.getFromLocation(location1.getLatitude(),
        location1.getLongitude(), 10);
}
catch (IOException e){
}
```

(그림 6) 역방향 지오코딩

3. 사용자 인터페이스

항상 곁에 두고 사용하는 스마트폰의 특성상, 사용자 인터페이스 디자인, 기기와 사용자간의 상호 작용, 그리고 사용성은 매우 중요한 부분이 된다. 사용자 인터페이스(User Interface)의 가장 기본 단위인 뷰(View)와 뷰그룹(ViewGroup)

을 객체로 사용하여 구성되며, 액티비티(Activity)안에 배치되어 액티비티 호출시 화면에 출력된다. 그리고 UI 이벤트가 발생하면 처리하기 위해 이벤트리스너를 호출한다. 이러한 안드로이드의 사용자 인터페이스는 크게 뷰와 뷰 그룹 그리고 액티비티 세 가지로 (그림 7)을 통하여 뷰 클래스의 관계를 확인 할 수 있다[4,7].



(그림 7) 관계도

3.1 GUI 뷰(View)

안드로이드의 모든 비주얼 컴포넌트는 뷰 클래스로부터 시작되며, 이 클래스 컴포넌트를 모두 묶어 '뷰'라고 부른다. 뷰는 컨트롤(Control) 또는 위젯(Widget)으로 대표되는 비주얼 인터페이스 요소를 위한 가장 기본적인 사용자 인터페이스 클래스이며, 사용자 인터페이스와 레이아웃(Layout)클래스는 모두 뷰에서 파생된다.

3.2 뷰 그룹(View Group)

뷰 그룹은 여러 개의 뷰를 적재할 수 있는 뷰 클래스의 확장 버전으로 뷰 그룹 클래스를 확장하면 서로 연결된 여러 개의 뷰로 이루어져 있는 복잡한 컨트롤을 생성할 수 있다. 뷰 그룹 클래스는 사용자 인터페이스를 지원하는 Linear Layout과 같은 레이아웃을 관리할 때 사용된다.

3.3 액티비티(Activity)

액티비티는 어플리케이션 컴포넌트 중 사용자와 직접 상호 작용하는 컴포넌트로서 프레젠테이션 계층에 속하면서 다른 플랫폼의 폼(Form)에 해당하는 안드로이드의 구성 요소로 사용자 인터페이스를 표시하기 위해 뷰나 레이아웃을 액티비티에 지정한다.

4. 설계

4.1 시스템 설계

모든 어플리케이션의 설계는 기존 유사 어플리케이션의 분석 과정으로부터 진행되며, 기존의 앱과 차이점을 분석하여 진행한다.

Weather Info 앱의 XML 파싱기법과 GPS 위치 프로바이더를 이용하여 날씨 데이터를 정확하게 로딩하고, GPS 위치 프로바이더를 이용하여 현재 위치정보를 가져와서 데이터를 로딩 받는 방법으로 설계한다. 또한, 알람(Alarms) 시스템을 적용하여 앱의 접속과 종료에 상관없이 데이터를 음성(English)으로 로딩 받는 방법으로 설계한다.

4.2 XML 파싱기법

XML은 인터넷 웹페이지를 만드는 HTML에서 개선하여 발전된 언어라고 볼 수 있는데, 안드로이드 어플리케이션에서 사용되는 XML 파싱 기법은 주로 인터넷에서 특정 정보를 얻어올 때 사용한다.

XML 파싱 기법은 크게 DOM과 SAX 두 종류의 방식이 존재하는데, DOM 방식은 XML 데이터를 트리 형으로 메모리에 저장하는 방식으로 메모리 사용량이 많으며, 속도가 느린 단점이 있어 SAX방식을 사용하였다. SAX방식은 XML 데이터를 파싱하면서 메모리에 저장하므로 메모리 사용량이 적고, 빠른 속도처리가 장점인 방식이다. 또한, 안드로이드에서 앱이 실행되면 기본적으로 스레드가 동작하게 되는데 XML 파싱을 이용하여 날씨 정보를 로딩 하는 과정 동안 모든 작업을 멈추고 파싱이 완료되기만을 기다리기 때문에 로딩시간이 길어지는 단점이 있다. 이러한 단점을 해결하기 위해 자바의 특징 중 하나인 멀티스레드를 이용하여 독립적인 작업을 동시에 실행시켜 데이터 로딩 시간을 단축시키도록 하였다.

4.3 GPS위치 프로바이더

GPS위치 프로바이더는 현재 위치에 대한 주소 정보를 받아오는 방법으로, 위치정보를 통해 해당 지역의 날씨를 파싱하기 위해 GPS를 사용한 위치기반서비스(LBS)를 이용하였다. LocationManager는 LocationProvider를 통해 위치정보

(경도,위도)를 감지해 geocoder를 통해 역방향 지오코딩을 하여 주소정보를 가져오게 된다.

지오코딩은 관련 연구에서 말했듯이 두 가지 방법으로 사용되는데, 어느 한 주소의 위도와 경도를 찾을 경우 사용되는 순방향 지오코딩, 위도와 경도에 대한 주소를 찾는 역방향 지오코딩이다. 이러한 방법으로 가져온 주소정보는 안드로이드에서 제공하는 SQLite를 통한 Database에 저장된 기상청의 각 지역 동별 XML주소를 Select query를 통해 얻어와 해당 XML페이지를 파싱하게 된다.

```

LocationManager locationManager = (LocationManager) getSystemService(LOCATION_SERVICE);
LocationListener listener;
GeoCoder geoCoder = new GeoCoder(this, Locale.KOREAN);
listener = new LocationListener() {
    public void onStatusChanged(String provider, int status, Bundle extras) {
        switch(status) {
            case LocationProvider.OUT_OF_SERVICE:
                break;
            case LocationProvider.AVAILABLE:
                break;
            case LocationProvider.TEMPORARILY_UNAVAILABLE:
                break;
        }
    }
    public void onProviderEnabled(String provider) {
    }
    public void onProviderDisabled(String provider) {
    }
    public void onLocationChanged(Location location) {
        double lat = location.getLatitude();
        double logi = location.getLongitude();
        try {
            List<Address> addresses = geoCoder.getFromLocation(lat, logi, 1);
            Address address = addresses.get(0);
            if(address.getAdminArea() != null)
                msg1 = address.getAdminArea(); //주명
            msg1 = address.getLocality(); //도시명
            msg2 = address.getThoroughfare(); //동명
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
};

```

(그림 8) 위치 및 주소 간략 소스

4.4 알람(Alarms) 시스템

알람은 예정된 시간에 인텐트를 발생시키는 동작으로 앱의 실행 여부와 무관한 기능으로, 앱의 접속과 종료에 상관없이 이벤트 또는 액션을 구현할 수 있다.

이러한 알람 기능의 사용에는 android.app 패키지 안에 있는 NotificationManager를 통해 위임형 인텐트로 직접 특정 액티비티를 호출하는 것이 아닌 다른 컴포넌트에 특정 액티비티를 실행시켜달라고 요청하여 알람을 실행하므로 어플리케이션의 시작과 종료와 상관없이 이벤트 또는 액

션을 구현할 수 있다. 알람 서비스를 이용하여 특정시간에 날씨정보를 음성으로 출력하여 어플리케이션을 실행시킬 필요가 없어 사용자에게 편리함을 제공한다.

알람을 통해 처리하는 음성 기능은 구글에서 제공하는 TTS(Text To Speech)를 사용하는데 한국어는 아직 제공하지 않아 영어를 통해 짧게 날씨 정보를 아래 그림과 같은 방법으로 제공한다.

```

OnInitListener MyTTSListener = new OnInitListener(){
    public void onInit(int status){
        if(status == TextToSpeech.SUCCESS)
        {
            int result = tts.setLanguage(Locale.US); //미국식 발음
            if(result == TextToSpeech.LANG_MISSING_DATA ||
                result == TextToSpeech.LANG_NOT_SUPPORTED)
            {
                Log.e("TAG", "Language is not available.");
                tts.speak(strText, TextToSpeech.QUEUE_FLUSH, null);
            }
        }
    }
};
    
```

(그림 9) 음성 정보 코드

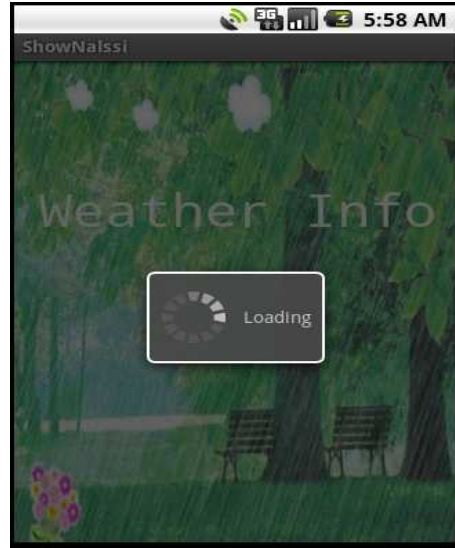
5.구현

5.1 GPS정보를 이용한 날씨 출력

(그림 10)은 어플리케이션의 메인화면으로 GPS위치 프로바이더를 이용하여 순방향 지오코딩을 통하여 경도와 위도 값을 가져와 위치 정보에 맞는 날씨 정보를 출력한다. 이러한 위치 정보를 이용해서 기상청의 날씨 정보를 XML 파싱기법을 이용하여 (그림 11)과 (그림 12)와 같이 동 단위와 시간 단위로 출력한다.



(그림 10) GPS 위치



(그림 11) 정보 로딩



(그림 12) 현재위치 날씨 정보

5.2 XML 파싱기법을 이용한 전국 날씨

어플리케이션의 대표적인 4가지 메뉴 중 하나의 메뉴로 XML 파싱기법을 SAX방식으로 이용하여 XML데이터를 파싱하면서 메모리에 저장하므로 메모리 사용량이 적고, 빠른 속도로 처리하여 전국 날씨를 출력한다. 또한 동 단위 및 시간 단위로 전국 날씨 정보를 (그림 13)과 같이 출력한다.



(그림 13) 전국 날씨 정보



(그림 15) 주간 날씨

5.3 알람 시스템의 음성(English)기능

기존 유사 날씨 어플리케이션의 차별화 된 기능 중 하나로, 알람 기능을 탑재하여 알람 이벤트가 진행 될 때 마다 그에 맞는 시간대의 날씨 정보를 음성(English)으로 아래 그림과 같이 대화상자와 함께 출력한다. 이러한 알람 시스템을 이용하면 어플리케이션의 접속과 종료의 상관없이 그 날의 날씨 정보를 시간 단위로 확인하므로 사용자에게 편의성을 제공한다.



(그림 14) 음성(English) 날씨 정보

5.4 지역 설정을 이용한 주간 날씨 출력

GPS 현재 위치와 상관없이 특정 지역의 주간 날씨 정보만 보고 싶을 때에 사용하는 기능으로, 지역 설정 후 그 특정 지역만 날씨 정보를 주간으로 불러온다. 최저 최고온도와 날씨의 상태를 이미지로 표현하며, 간단한 구조의 사용자 인터페이스를 설계하여 리스트 방식과 다르게 이미지를 불러오는 형식으로 (그림 15)와 같이 출력 한다.

5.5 테스트

안드로이드 기반의 어플리케이션 테스트 방법은 안드로이드 SDK를 이용한 에뮬레이터를 실행하는 방법과 스마트폰 이용하는 방법으로 두 가지가 존재한다. 일반적으로 개발 도중에는 에뮬레이터에서 테스트를 진행하고, 단계별로 휴대 단말에서 확인하는 형태로 테스트를 진행하였다.

본 논문의 날씨 정보 어플리케이션은 이미 수십 번의 테스트를 진행하고, 테스트 중 발견되는 잘못된 경로 인식 오류와 Failed to Location을 나타내는 오류를 검출하였다. 이러한 오류들을 방지하고, 관련 연구와 설계 과정을 통해 GPS 위치 프로바이더를 이용한 날씨 추출과 기상청의 날씨 정보를 이용한 전국 날씨 추출, 위치에 상관없이 해당되는 날씨 정보만 보고 싶을 때 지역 설정을 통하여 그에 맞는 지역의 날씨 추출, 그리고 알람 기능을 이용한 날씨(음성) 추출, 마지막으로 실시간으로 날씨 정보 업데이트 등 Weather App의 테스트를 완료하였다.

또한, 이러한 테스트 과정을 통해 다른 앱과 비교한 내용은 <표 1>과 같다.

<표 1> 유사 앱과 기능 비교

	본 논문	어제 오늘	Olleh 날씨
전국 날씨	○	○	○
주간 날씨	○	○	○
GPS 날씨	○		
음성 날씨	○		
출퇴근예보			○
날씨 이야기		○	

6. 결론

본 논문에서 제시하는 GPS 위치 프로바이더를 이용한 안드로이드 기반의 날씨 정보 어플리케이션은 사용자에게 보다 편리함을 줄 수 있는 안드로이드 기반의 날씨 정보 어플리케이션이 되리라 본다. 각종 정보가 많은 어플리케이션보다 단순해보여서 좋은 어플리케이션으로 보지 않을 수도 있지만, 날씨 어플리케이션은 날씨만 제공하는 것으로 사용자 인터페이스의 요소와 정보들이 너무 포괄적인 경우 복잡한 사용자 인터페이스가 될 수 있기 때문에 날씨를 어떻게 나타내느냐에 따라 어플리케이션의 신뢰성과 편의성, 시장성을 보장 받을 수 있다고 생각된다.

본 논문에서 제시하는 방법은 XML 파싱 기법과 GPS 위치 프로바이더를 이용하는 것에 중점을 두고 있으며, 정확한 데이터 로딩과 Fail to Location 오류를 해결할 수 있고, 알람 시스템을 이용하여 어플리케이션의 접속과 종료에 상관없이 음성(English)으로 날씨 정보를 출력하며 사용자에게 편의성을 제공한다.

참고 문헌

[1]안드로이드웹, <http://www.androidpub.com>
 [2]안드로이드사이드,<http://www.androidside.com>
 [3]테크산업연구소,<http://blog.naver.com/idacoi>
 [4] 김상형, 안드로이드 프로그래밍 정복, 한빛 미디어, 2010.
 [5] 이두진, 안드로이드 앱 개발 완벽 가이드, 피씨북, 2011.
 [6] 표성배 신성윤 서정민 장영현 하안, 톡톡 튀는 어플 개발을 위한 안드로이드 프로그래밍, Global, 2011,
 [7] 고현철 유형목, 안드로이드의 모든 것 분석과 포팅, 한빛 미디어, 2011
 [8] 세인 콘더 로런 다시, 시작하세요! 안드로이드 프로그래밍, 위키북스, 2009.7
 [9] 김정훈, 구글의 안드로이드 프로그래밍, 성안당, 2009.
 [10] 진성주 최종열 백정현 신중훈, 안드로이드 프로그래밍, 북스홀릭, 2011.
 [11] Reto Meier, Professional Android Application Development, J.F, 2009 pp.20~110
 [12] 윤형수, 이진욱, 하수철, "XML Parsing 기법을 이용한 Android 기반의 종합 날씨 정보 어플리케이션", 한국디지털콘텐츠학회 학술대회 논문집, Vol.12, No.1, pp.123~127, 2011.

이진욱



2012년 대전대학교 컴퓨터공학과 졸업예정 (공학사)

관심분야 : 멀티미디어, 안드로이드 등

윤형수



2012년 대전대학교 컴퓨터공학과 졸업예정 (공학사)

2009년 ~ 현재 : 한영시스텍 근무

관심분야 : 네트워크, 데이터베이스, 안드로이드 등

하수철



1990년: 홍익 대학교 컴퓨터공학과 (공학박사)

1991년 ~ 1992년 플로리다 주립대/텍사스 주립대 교환교수

2000년 위성통신 주립대 방문연구학자

2008년 ~ 2010년: (사) 한국계입학회 부회장

2000년 ~ 현재 : 디지털콘텐츠학회/멀티미디어학회 이사

1999년 ~ 현재 : 멀티미디어 콘텐츠기술센터소장

1987년 ~ 현재 : 대전대학교 컴퓨터공학과 교수

관심분야 : 소프트웨어공학, 게임공학, 융합공학, 콘텐츠공학, 멀티미디어, 프로그래밍언어, 디지털/아날로그 아트 등