

안드로이드 환경에서의 부산지역 통합 교통 서비스 시스템

이재호, 최정준, 설성문, 최종영, 박유현
동의대학교 컴퓨터소프트웨어공학과

An Integration Transportation Service System In Busan Using Android

Jae-Ho Lee, Jung-Joon Choi, Seung-Moon Seol, Jong-Young Choi, Yu-Hyun Park
Dept. of Software Engineering, Dong-Eui University
E-mail : happymax44@naver.com

요 약

현재 스마트폰 시장에는 다양한 모바일들이 등장하고 있다. 심비안과 윈도우 모바일이 주를 이루던 모바일 업계에 애플사가 iPhone을 들고 나타나 큰 반향을 일으켰다. 애플사의 iPhone은 앱 스토어라는 새로운 시장을 창출하여 모바일 폰의 거대한 변화를 일으키고 있다. 그리고 이제 구글이 지원하는 안드로이드가 나타나 자리를 차지해 나가고 있다. 휴대폰뿐만이 아닌 다양한 모바일 기기에 안드로이드 플랫폼이 올라가고 안드로이드가 우리 생활의 구석구석을 차지 할 것이다. 본 논문에서는 현재 부산버스노선, 지하철노선의 어플리케이션이 다른 지역에 비해서 개발이 미흡한 실정이다. 시민들이 많이 이용하는 버스, 지하철에 대한 정보를 스마트폰에서 구현함으로써 사용자가 실생활에 편리하게 이용할 수 있게 버스, 지하철에 대한 노선정보, 대기시간, 환승정보, GPS를 이용하여 현재위치에서 역검색을 할 수 있게 한다.

1. 서 론

지금까지 휴대전화를 사용하는 대부분의 사용자는 단순한 전화 통화나 문자 메시지 전송 등의 서비스에 만족하였으나, 최근 스마트폰의 선풍적인 인기로 인해 스마트폰 사용자들은 보다 다양한 서비스를 제공 받고 있다. 스마트폰은 2000년대 중반 미국 내 사무직 종사자들을 중심으로 소수의 사용자들만 사용하기 시작하다 최근 애플의 iPhone의 인기로 점차 대중화되고 그 뒤를 이어 안드로이드가 부상되고 있는 중이다. 초기의 스마트폰은 소수의 사무직 종사자를 위한 서비스가 주를 이루었기 때문에 언제 어디서든 업무를 할 수 있는 환경을 제공하는 데 그쳤지만, 최근 대중화된 스마트폰은 이러한 기능은 물론 일반 사용자들의 다양한 요구사항을 제공하고 있다. 구글의 안드로이드용 소프트웨어를

보급하고 있는 안드로이드 마켓에는 수많은 종류의 응용 프로그램이 등록되어 있어 일반 사용자들은 과거에 비해 다양한 종류의 소프트웨어를 저렴한 가격으로 구입해 사용할 수 있다. 그러나, 아직은 애플의 앱스토어보다는 어플리케이션의 종류도 많지 않고 규모도 작은 편이다. 본 논문에서는 안드로이드를 사용하는 부산지역 사용자들을 위한 공공 서비스의 하나로, 대중교통을 보다 편리하게 사용하기 위한 응용 소프트웨어를 개발하고 그 결과를 보이고자 한다. 부산지역에서는 이미 전화나 인터넷을 통해 버스는 BIMS(Busan Bus Information System)[1]가 구축되어 있으나 지하철은 시스템도 구축되어 있지 않은 상태이다. 또한, 구축되어 있는 BIMS는 스마트 폰 환경에는 적당하지 않은 실정이다. 따라서, 본 연구를 통해 BIMS와 지하철을 통합

한 시스템은 부산지역 스마트폰 사용자들이 대중교통 정보를 쉽고 다양하게 활용하는데 크게 기여할 것이다.

II. 관련 연구

현재 스마트폰용 부산교통 정보시스템 어플리케이션은 iPhone, Window Mobile용으로만 개발되어 있다. 개발자는 지금 개발된 Application 보다 기능이 더 향상된 Android용 Application을 구현한다.

<표1> 스마트폰용 교통Application 비교 [2]

기능	I-Bus Busan (iPhone)	Dynamic Busan (Window Mobile)	Android
즐거찾기	O	O	O
사용자 위치표시	O	O	O
정류소 번호검색	O	O	O
정류소 이름 검색	O	O	O
버스환승정보	X	X	O
지하철 정보	X	X	O

<표1>에서 현재 서비스되고 있는 iPhone, Window Mobile Application에서 구현되어 있지 않은 버스환승 정보, 지하철 정보를 추가하여 제공한다.

iPhone은 iPhone단말기에만 한정되어있고, iPhone을 개발하기 위해 Intel CPU기반의 Mac이 반드시 필요하다. Window Mobile은 실제 모든 이통사와 호환되지 않으며 국내 휴대폰만 한정되어 있다. 안드로이드은 다른 스마트폰의 단점을 보완하며 Google이라는 매리트, OpenGL ES지원 Google API지원한다.

III. Android 구조

개발자는 안드로이드 아키텍처에서 정의된 4가지 기본적인 구성요소 타입을 써서 안드로이드 어플리케이션을 만든다. Activity는 오피스 어플리케이션처럼 데스크탑에서 독립적으로 실행되는 어플리케이션과 비슷하다.

Activity란 사용자나 운영체제가 시작해 필요한 동안만 실행한 후 닫는 실행코드이다. 이 코드는 사용자와의 상호작용을 담당하거나 질의나 Intent를 통해 다른 Activity 또는 서비스에 데이터나 서비스를 요청한다. 대부분의 안드로이드 실행코드는 Activity에 속한다. 각 Activity는 사용자에게 하나의 스크린을 보여준다 Activity가 실행되고 있지 않다면 메모리 절약을 위해 안드로이드가 해당 Activity를 강제로 종료할 수 있다.

Service는 데스크탑이나 서버 운영체제에서의 서비스 프로세스나 데몬과 비슷하다. Service는 주로 백그라운드에서 돌아가는 실행코드로, 일단 실행되면 휴대폰이 꺼질 때까지 돌아간다. Service는 일반적으로 사용자 인터페이스가 없다. Service의 대표적인 예로는 음악 리스트를 끊임없이 재생해야 하는 MP3플레이어를 들 수 있다. 이런 서비스는 사용자가 다른 프로그램을 띄워 사용하더라도 계속 돌아가야 한다. 이처럼 사용자 인터페이스 없이 백그라운드에서 지속적으로 돌아가는 작업을 수행하려면 Service를 구현해야 한다.

Broadcast Receiver는 다른 어플리케이션으로부터 온 요청을 처리하는 구성요소들이다.

이런 이벤트는 배터리 수명이 낮아졌을 때처럼 안드로이드 자체에서 발생할 수도 있고 다른 프로그램에서도 발생할 수 있다. 반면 Intent Receiver는 어떤 Activity나 서비스가 그 기능을 다른 어플리케이션에 제공하기 위해 실행한다. Intent Receiver는 다른 프로그램의 데이터 요청이나 서비스 요청을 처리하는 작은 실행코드이다. 요청하는 프로그램이 Intent를 발행하면 안드로이드가 알아서 어떤 어플리케이션이 이를 받아 처리해야 할지 결정한다.

다른 Activity나 서비스로 정보를 공유하려면 Content provider를 만들어야 한다. Content provider는 다른 어플리케이션으로부터 데이터 요청을 처리하기 위해 URI형식의 표준 인터페이스를 사용한다. 요청하는 어플리케이션은 실제로 Content provider를 사용했는지 모를 수도 있다. 예를 들어 어떤 어플리케이션이 주소록 데이터에 대한 쿼리를 보낼 때 그 쿼리를 content://contacts/people로 보낸다. 이때 안드로이드는 이런 형식의 URI를 처리할 수 있는 Content provider라고 등록된 어플리케이션에 무엇이 있는지 찾아보고 적합한

어플리케이션에 그 요청을 보낸다. 만일 그 URI에 대해 여러 개의 Content provider가 등록돼 있다면 안드로이드는 사용자에게 그 중 어떤 Content provider를 사용하고 싶은지 묻는다.

IV. 프로그램 설계

BIMS는 대중교통 혁신 및 활성화프로그램의 일환으로 효과적인 버스운행관리 및 버스 경쟁력 확보를 통해 버스안전 및 이용자 편의를 증진시키고 효율적인 버스정책 지원 및 과학적 운행체계 확립을 실현하기 위한 대중교통 시스템이다. 시내버스의 차량단말기에서 위성을 통해 차량에 대한 GPS 위치 좌표를 수신하고, 수신된 GPS 좌표를 이용하여 차량단말기에서 버스의 위치, 운행속도, 도착정류장 등을 검출한 후, 해당자료를 무선통신망을 통해 센터(버스종합사령실)로 송신. 버스로부터 수신된 데이터를 이용하여 실시간 버스운행정보 가공 시간 버스운행정보를 시민 및 운전자에게 인터넷, Mobile, PDA, ARS, 운전자안내기, 정류장안내기 등으로 제공한다.

최단거리를 구하는 알고리즘에는 다이스트라, 벨만 포드, A*알고리즘 등이 있다. 다이스트라 알고리즘은 모든 간선 가중치가 음이 아닌 일반적인 경우 사용하고, 벨만-포드 알고리즘은 음의 가중치가 존재하는 경우 사용하지만, 속도가 느리다. 현재 길찾기 알고리즘으로 A*알고리즘을 사용하는 추세이다. 본 논문에서는 A*알고리즘을 사용하여 환승시스템을 구현한다. 이 알고리즘은 초기 노드에서 목표 노드까지의 경로를 찾는 그래프 탐색 알고리즘이다. 목표 노드까지의 가장 좋은 경로를 추정 (estimate of the best route) 하기 위해 각 노드에 랭킹을 부여하는 "heuristic estimate" 를 사용하고 그 순서대로 노드를 방문한다. A*알고리즘의 기본은 아직 조사하지 않은 state들 중 가장 유용할 듯한 state를 조사하는 과정을 반복하는 것이다.

A*알고리즘으로 최단거리를 구하여 버스/버스, 버스/지하철 환승시스템을 구현한다. 버스/버스 환승시스템을 사용하면 목적지에서 도착지까지의 같은 경로 버스들을 모두 보여준다. 버스/지하철 환승시스템을 사용하면 환승 가능한 지하철역을 Search하여 버스들을 보여준다. 사용자의 설정으로 개수를 제한 할 수 있다.

사용자는 BIMS를 통해 버스도착예정 시간을 알수 있지만 보다 편의를 위해 버스가 도착하기

전 설정을 통해 알림이 가능하다.

지하철 정보는 실시간으로 정보를 가져올 수 없기에 지하철 노선, 지하철 도착시간, 첫차/막차 시간, 배차 시간을 Database을 이용하여 Application에 저장해 놓는다.

그리고 본 시스템은 그림1, 2와 같이 GPS를 기반으로 하여 자신의 위치를 알려준 다음 근처의 버스, 지하철 위치를 표시하여 준다. Google API를 사용하여 GPS로 위도, 경도를 받아와서 MapView에 표시해 준다. 그리고 버스정류소와 지하철역 위치를 각각 아이콘으로 표시하며 사용자의 위치도 표시한다.



그림 1. 버스정류소



그림 2. 지하철역

IV. 결 론

안드로이드 환경에서의 부산지역 통합 교통서버스 시스템을 통하여 현실세계에서 이루어지고 있는 버스와 지하철의 환승시스템을 인공지능적으로 편리하게 보여주게 될 것이다. 기존의 어플리케이션은 버스, 지하철을 양분화 하여 하나의 운송수단에 관한 정보를 보여주고 있으나 본 어플리케이션을 통하여 현실세계에서 이루어지고 있는 시스템을 그대로 이용하게 될 것이다.

향후 연구로는 부산에 관광정보, 맛집정보를 추가할 것이다. 또한, 환승 시 교통혼잡, 운송수단의 과업 등 일어날 수 있는 경우의 최단시간을 연구할 예정이다.

참고문헌

- [1] <http://bus.busan.go.kr/>
- [2] http://cafe.naver.com/witunes.cafe?iframe_url=/ArticleRead.nhn%3Farticleid=1160
http://cafe.naver.com/bjphone.cafe?iframe_url=/ArticleRead.nhn%3Farticleid=648391
- [3] Mark Murphy "The Busy Coder's Guide to Android Development", Sep 28, 2009.
- [4] Nam Sang-Yup, Kang Min-Gu "Android Analysis and Practice", Sep 5, 2009.
- [5] Rick Rogers "As an example, to start Android development", Sep 15, 2009.
- [6] Ed Burnette "Hello, Android", Mar 18, 2009.
- [7] Shane Conder, Lauren Darcey "Start Now! Android Programming", Nov 19, 2009.
- [8] Lee Snag-Jin "열혈강의 자료구조" Jan 15, 2010