

그룹 관리에 효율적인 어플리케이션의 구현

김현철*, 이상곤**
 (주)다날*, 전주대학교 컴퓨터공학과**
 e-mail : bless332@naver.com, samuel@jj.ac.kr

Implementation of a Group Manager for Social Network Service

Hyeoncheol Kim*, and Samuel Sangkon Lee**
 Danal Entertainment*
 Dept. of Computer Science and Engineering, Jeonju University**

요 약

본 논문에서는 그룹 전용으로 사용하기 위한 소셜 네트워크 서비스(SNS)를 제안한다. 본 논문의 시스템 개발을 위해 그룹 서비스, 채팅, 위치 추적, 전송 기능, 그림 그리기와 편집 기능, 사진첩 관리 기능, 파일 전송 기능 등을 구현하고 전체 모듈을 결합하여 안드로이드용 그룹 관리 어플리케이션 시스템을 설계하였다.

1. 서론

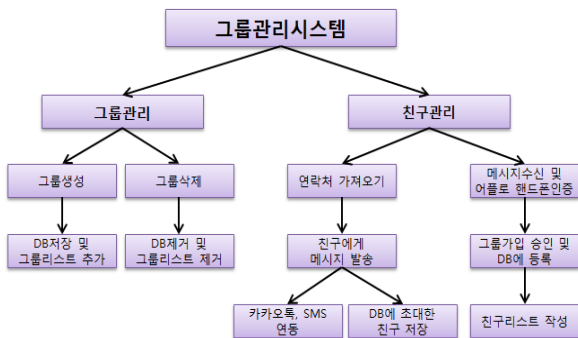
현재에는 페이스 북(facebook), 트위터(twitter), 카카오톡(Kakao Talk) 등 각 메신저 들의 소셜 네트워크(social network) 서비스가 그 위세를 떨치고 있다. 소셜 네트워크 시장의 많은 사용자들에게 새로운 기능이 탑재된 서비스를 제공하여야 할 시점이다. 본 논문의 시스템 개발에 여러 제약 사항을 열거해 보면 다음과 같다. 본 어플리케이션의 하드웨어 사양은 안드로이드 2.3 버전 진저 브레이드 이상에서 동작하며, 그 이하의 버전에서는 사용할 수 없다. 또한 제작 및 테스트를 위해 최소한 3대 이상 개통된 안드로이드 전용 스마트폰이 필요하다.

본 논문의 구성은 다음과 같다. 2장에서는 구현 기술로 JSP를 이용하여 데이터베이스의 통신과 XML Parsing 방법을 소개하고, SQLite의 데이터베이스 통신과 GPS 서비스의 구현 방법을 소개한다. 3장에서는 그룹 관리용 소프트웨어를 위해 구현한 기능을 중심으로 소개하고, 4장에서 결론 및 추후 개선 사항에 대해 토의한다.

2. 요구 분석

본 논문의 이론적 배경을 설명하기 위해 먼저 시나리오를 다음과 같이 설정하였다. 직장인 K씨는 고등학교 친구 5명과 함께 정기적으로 모임을 갖는다. 서로 약속을 정하거나 모임 이야기를 주고받는데 불편함을 느꼈다. 바쁜 L씨가 시간을 정해 팀원 모두에게 전체 문자를 보냈더니, P씨와 M씨가 시간이 안 되는 것이었다. 다시 모두에게 연락하여 가능한 시간을 조율하다 보니, 많은 시간이 오래 걸리고 여간 불편한 것이 아니었다. 이에 K씨는 소그룹을 관리하는 어플리케이션이 있다는 이야기를 듣고 (그림 1)과 같이 이를 설치한다. K씨는 어플리케이션을 설치하고 그룹을 생성한다. 생성 후, 연락처에 있는 5명의 친구에게 가입 초대 카카오톡(or 문자) 메시지를 발송한다.

메시지를 확인한 친구들은 수락을 하고, 안드로이드[1]마켓 설치 페이지로 이동된다. 설치가 완료되면 핸드폰 번호로 인증하고 그룹 가입이 완료된다. 가입된 그룹원은 [연락처] 메뉴에서 확인하고 전화, 문자, 카카오톡 메시지를 발송한다. (그림 2)와 같이 [채팅] 메뉴에서 서로 실시간 채팅을 이용한다. L씨가 찍은 셀카1) 사진을 사진첩에 올렸고, 실시간으로 다른 친구들에게 푸시(PUSH)로 알람이 간다. 사진을 확인한 친구들은 리플로 사진에 대한 생각을 서로 나눈다. [일정] 메뉴에서 몇 일/ 몇 시에 모임을 가질 것인지 등록하고, 각각의 그룹원들은 약속 X 시간 전에 푸시 알람을 받을 수 있는 기능으로 약속에 참석한다. 약속은 잡았는데 약속 장소를 정확히 설명하기 어려운 경우가 있는데, 이 때는 (그림 3)과 같이 '위치 전송' 기능을 통해 특정한 위치를 친구에게 전송하여 약속 장소에 찾아올 수 있도록 한다. 또한 친구들의 위치를 실시간

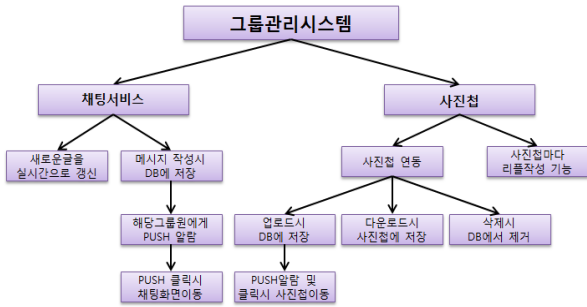


(그림 1) 그룹 관리와 그룹원 관리 모듈의 구조도

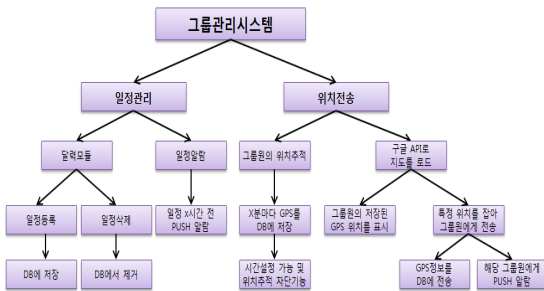
1) '셀프카메라(self camera)'를 줄여 이르는 말. 동아일보의 2004년 06월 18일 자 신문 기사에 다음의 문장에서 처음 출현함. 셀카가 유행처럼 번지는 것은 '잘 찍히는 것'에 대한 관심이 얼마나 많은지 엿볼 수 있는 사례다.

으로 확인할 수 있어 편리하게 이용할 수 있다.

본 논문의 어플리케이션이 실행되면 자신의 핸드폰 번호로 인증을 받는다. 인증 후, 그룹 목록이 나오고 자신만의 그룹을 개설한다. 개설 후, 자신의 연락처로 친구들을 초대할 수 있다. 초대하면 문자나 카카오톡으로 초대 메시지가 발송된다. 초대를 받은 사용자들은 메시지의 링크를 선택하여 마켓의 어플리케이션 다운로드 페이지로 이동한다. 설치 후, 핸드폰 번호로 인증하고 자동으로 그룹에 추가된다. 데이터베이스에 그룹명과 핸드폰 번호가 암호화되어 저장된다[2].



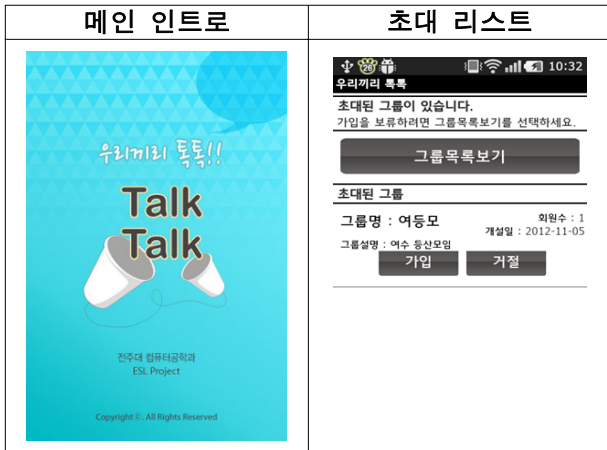
(그림 2) 채팅 서비스와 사진첩 모듈의 구조도



(그림 3) 일정 관리와 위치 전송 모듈의 구조

3. 구현 기능

본 논문에서 설계한 어플리케이션의 메인 화면을 아래의 (그림 4)와 같이 디자인하고, 그 옆은 친구 초대 리스트의 수행 결과를 나타낸다.



(그림 4) 초기 화면과 사용자 인터페이스



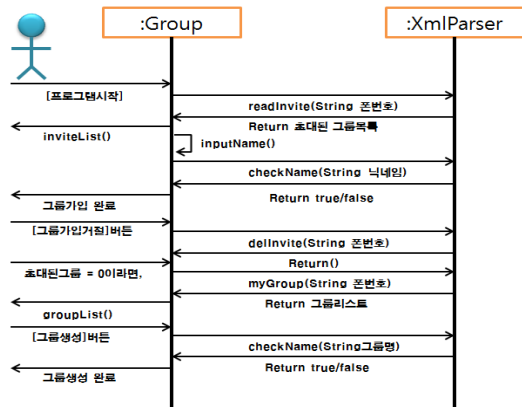
(그림 5) 그룹의 생성과 목록 관리 화면

3.1 그룹 관리를 위한 시퀀스 다이어그램

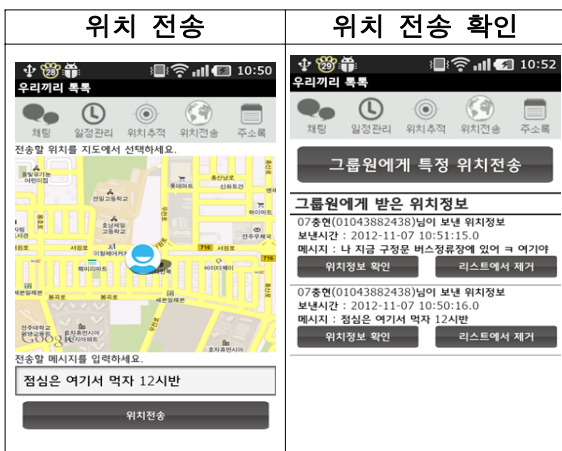


(그림 6) 그룹원의 채팅 화면과 위치 추적

그룹 관리의 시퀀스는 (그림 7)과 같은데, Group 클래스와 XmlParser 클래스가 참여한다. 프로그램이 시작되면, Group 클래스에서 XmlParser 클래스의 readInvite를 호출하여 초대된 그룹 목록을 받아와 inviteList를 이용하여 사용자에게 출력한다. 가입 시, 닉네임을 입력하면 XmlParser 클래스의 checkName을 호출하여 중복 확인을 하고 그룹 가입을 완료시킨다. 만약 가입을 거절하면, XmlParser 클래스의 delInvite가 동작하여 DB에서 초대된 데이터를 삭제하게 된다. 만약 초대된 그룹이 '0' 이면, XmlParser 클래스의 myGroup를 호출하여 자신이 가입된 그룹 목록을 가져와 사용자에게 출력한다. 만약 그룹 생성 버튼을 선택하면 XmlParser 클래스의 checkName을 호출하여 중복을 확인하고 그룹 생성을 완료한다.



(그림 7) 그룹 관리를 위한 시퀀스 다이어그램



(그림 8) 위치 전송의 구현 화면

3.2 친구 관리



(그림 9) 일정 관리

친구 관리의 시퀀스는 Address, XmlParser, AddressInvite 클래스가 참여한다. 이것은 주소록이 선택되었을 때 동작하며 Address 클래스에서 XmlParser 클래스의 readMyGroupList를 호출하여 자신의 그룹원 정보를 받아와 사용자에게 그룹원 목록을 출력한다. 초대하기 버

튼을 선택하면 AddressInvite 클래스의 readMyAddress가 호출되어 자신의 연락처의 데이터를 화면에 출력하게 된다. 다음으로 초대할 사용자를 터치할 경우, clickListener가 호출되면서 적절한 그룹원인지 확인하기 위해 Address 클래스의 checkGroupUser를 호출하여 중복 확인을 실시한다. 그룹원이 아니면 alertDialog로 다이얼로그를 호출하고, 초대하게 되면 sendSMS를 호출하여 선택된 대상에게 자동으로 SMS가 발송된다. 발송이 완료되면 XmlParser 클래스의 writeInvite가 호출되어 초대된 내역을 DB에 저장하게 된다.

3.3 채팅 서비스

친구 관리의 시퀀스는 Chat, XmlParser, SQLHelper 클래스가 참여한다. 채팅이 선택되면 동작하며 Chat 클래스에서 XmlParser 클래스의 readChatCount를 호출하여 DB의 채팅 개수를 가져온다. 같은 방법으로 SQLHelper 클래스의 readChatCount를 호출하여 어플리케이션의 채팅 개수를 가져온다. chatCountGap을 호출하여 업데이트해야 할 채팅 개수를 확인하고, XmlParser 클래스의 readChat를 호출하여 업데이트 된 채팅 개수 만큼을 불러온다. 업데이트된 채팅을 SQLHelper 클래스의 writeChat를 호출하여 자신의 어플리케이션에 저장하고, printListView를 호출하여 화면에 출력한다. 만약, 사용자가 채팅을 입력할 경우 XmlParser 클래스의 writeChat를 호출하여 DB에 저장하고, 위의 채팅 업데이트 로직을 다시 수행한다.

3.4 사진첩

사진첩의 시퀀스는 Photo, XmlParser, SQLHelper, PhotoDetail 클래스가 참여한다. 이 클래스는 사진첩이 선택되었을 때 동작하며 Photo 클래스에서 XmlParser 클래스의 readPhotoCount를 호출하여 DB의 사진첩 개수를 가져온다. 같은 방법으로 SQLHelper 클래스의 readPhotoCount를 호출하여 어플리케이션의 사진첩 개수를 가져온다. photoCountGap을 호출하여 업데이트 해야 할 사진첩 개수를 확인하고, XmlParser 클래스의 readPhoto를 호출하여 업데이트된 사진첩 개수 만큼 불러온다. 업데이트된 사진첩을 SQLHelper 클래스의 writePhoto를 호출하여 자신의 어플리케이션에 저장하고, photoGridView를 호출하여 화면에 출력한다.

만약 사용자가 사진을 클릭할 경우, PhotoDetail 클래스에서 SQLHelper 클래스의 readPhotoDetail를 호출하여 사진의 세부 정보를 화면에 출력한다. 다음으로 사진에 있는 리플도 출력해야 한다. XmlParser 클래스의 readRepleCount를 호출하여 DB의 해당 사진첩의 리플 개수를 가져온다. 같은 방법으로 SQLHelper 클래스의 readRepleCount를 호출하여 어플리케이션의 사진첩의 리플 개수를 가져온다. repleCountGap을 호출하여 업데이트해야 할 사진첩의 리플 개수를 확인하고, XmlParser 클래스의 readReple를 호출하여 업데이트 된 사진첩의 리플 개수 만큼을 불러온다 [3]. 업데이트 된 사진첩의 리플을 SQLHelper 클래스의



(그림 10) 그룹 관리 어플리케이션의 전체 모듈 구성도

writeReple를 호출하여 자신의 어플리케이션에 저장하고 화면에 출력한다. 만약 리플을 작성하면 XmlParser 클래스를 이용하여 DB에 저장하고, 리플 업데이트 로직을 다시 호출한다. 이상의 내용을 토대로 다음의 (그림 10)과 같이 그룹 관리 어플리케이션의 전체 구조도는 다음과 같다.

4. 결론

본 논문에서는 안드로이드용 그룹 관리 어플리케이션[4] 시스템을 설계하고, 그룹 서비스, 채팅, 위치 추적, 전송 기능, 그림 그리기와 편집 기능, 사진첩 관리 기능, 파일 전송 기능 등을 구현하고 전체 모듈을 결합하여 구글 플레이(구 마켓)에 ‘우리끼리 톡톡’이라는 이름으로 업로드 되어 있으며, 여러 사용자가 실제로 사용하고 있다. 메신저 어플리케이션의 특성상 성능면에서 더 빠른 속도를 보장하여야 하며, 데이터베이스의 철저 관리로 데이터의 소실이나 유출이 차단한다. 또한 사용자의 피드백을 통해 지속적으로 프로그램을 버전 업 하여야 한다. 현재 그룹 관리, 친구 관리, 채팅 서비스, 일정 관리, 위치 추적, 위치 전송 등의 사용 사례를 구현하였으며, 사진첩과 푸시 알림 기능은 추후 개선 사항으로 진행될 것이다.

본 시스템은 최근 사용량이 급격히 증가한 스마트폰 유저들을 공략할 것이다. 페이스북(facebook), 트위터(twitter), 카카오톡(KakaoTalk) 등 각 메신저 들이 소셜 네트워크라

는 서비스 아래 한창 위세를 떨치고 있는 중이다. 카카오톡의 약점인 그룹 기능을 잘 공략할 것으로 판단된다. 또한 어플리케이션을 통해 소셜 네트워크의 새로운 시장을 더 확대할 것으로 기대한다.

참고 문헌

[1] 안드로이드 정의 : <http://100.naver.com/100.nhn?docid=922356>
 [2] 최맹식, 김학수, "기계 학습에 기반한 한국어 미등록 형태소 인식 및 품사 태깅", 정보처리학회논문지(B), 제18권, 제1호, pp.45-50, 2011.
 [3] 김상형, "안드로이드 프로그래밍 정복", 한빛미디어, 2011
 [4] 안드로이드 개발 레퍼런스 : <http://developer.android.com/reference/android/package-summary.html>