

# 접촉 시점을 기반으로 하는 Android 플랫폼의 소셜 네트워킹 애플리케이션

정영교\*, 윤희용°

\*°성균관대학교 정보통신대학

e-mail: {jyoung0491, youn7147}@skku.edu\*°

## Social Networking Application on Android Platform based on Contact Timing

Young Gyo Jung\*, Hee Yong Youn°

\*°School of Information and Communication Engineering, Sungkyunkwan University

### ● 요약 ●

최근 직장인들이 직장생활을 효과적으로 하기 위한 인맥의 필요성이 높아지는 가운데 소셜 네트워크의 중요성이 큰 이슈로 떠오르고 있다. 이는 현재 직장인들이 온라인뿐만 아니라 오프라인에서도 소셜 네트워크 활동이 활발히 이루어지고 있기 때문이다. 소셜 네트워크를 통하여 지속적인 연락을 취함으로써, 두터워진 좋은 인맥은 자기에게 결정적으로 필요할 때 도움을 주기 때문에 직장인들 사이에서 소셜 네트워크 활동이 중요해 지고 있다. 소셜 네트워크는 고객을 대상으로 하는 영업사원들에게는 실적으로 바로 연계된다고 할 수 있다. 따라서 본 논문에서는 영업사원이 애플리케이션을 이용해 고객을 효율적이고 능률적인 고객을 관리할 수 있도록 한다. 추가적으로 원하는 고객을 따로 관리할 수 있는 기능과 고객 정보 동기화 기능이 있는 애플리케이션을 설계 및 구현하였다.

키워드: 안드로이드 플랫폼(Android Platform), 소셜 네트워크(Social network)

## I. 서론

최근 직장인들이 직장생활을 효과적으로 하기 위한 인맥의 필요성이 높아지는 가운데 소셜 네트워크의 중요성이 큰 이슈로 떠오르고 있다. 이는 현재 직장인들이 온라인뿐만 아니라 오프라인에서도 소셜 네트워크 활동이 활발히 이루어지고 있다. 소셜 네트워크를 통하여 지속적인 연락을 취함으로써, 두터워진 좋은 인맥은 자기에게 결정적으로 필요할 때 도움을 주기 때문에 직장인들 사이에서 소셜 네트워크 활동이 중요해 지고 있다. 소셜 네트워크는 고객을 대상으로 하는 영업사원들에게는 실적으로 바로 연계된다고 할 수 있다.

본 논문에서는 직장인 중 100명이상을 관리하는데 어려움을 느끼는 영업사원을 대상으로 고객관리에 편의성을 주는 애플리케이션을 개발하였다. 안드로이드에서 기본적으로 제공하고 있는 주소록 정보는 ContactsContract를 이용하고, 통화기록은 CallLog를 이용하여 그룹별 고객 정보를 불러들여, 전화와 문자 기록을 시간 단위로 표현하여, 시간이 높은 순으로 우선순위를 만들고, 어떤 고객에게 연락이 더 시급한지를 한눈에 알 수 있게 보여준다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 소셜 네트워킹의 이론적 고찰, 안드로이드 플랫폼 설명하며, 3장에서는 제안된 애플리케이션의 설계 및 구현을 서술/증명, 4장에서는 본 논문의 결론을 맺는다.

## II. 관련 연구

### 2. 관련연구

#### 2.1 소셜 네트워킹의 이론적 고찰

인간은 누구나 친밀하고 의미 있는 인간관계를 추구하는데 서로에게 의미 있는 존재가 됨으로써 삶이 가치 있고 풍요로워지기를 원한다. 이러한 인간관계는 관계의 대상 선택, 발전과 심화, 해체의 종결의 과정을 통해서 발전하고 심화된다. 흔히 인간관계의 발전 정도는 친밀감에 의해 평가되는데, Huston과 Levinger(1978)는 그러한 친밀성은 두 사람 사이의 상호의존성 정도에 따라 결정 된다고 주장한다. 인간관계의 발전 단계는 첫째, 쌍방 간의 직접적 접촉이 없는 면식의 단계, 접촉이 일어나는 단계, 관계가 심화된 상호의존의 단계로 나눌 수 있다. 각각의 단계에 따라 쌍방 간의 관계는 상호의존성의 깊이가 깊어지고 영역이 넓혀진다. 이렇게 친밀한 인간관계는 만남의 횟수가 많아지고 교제의 기간이 길다고 해서 깊어지는 것은 아니다. 서로간의 관계가 깊어질 수 있도록 촉진하기 위해서는 정서적 지지와 공감, 즐거운 체험의 공유, 현실적 도움의 교환, 자기공개와 같은 요인들이 필요하다.[3]

## 2.2 안드로이드 플랫폼

안드로이드 구조는 그림 1과 같은 구성 요소로 구성되며 이 구성요소에는 응용 프로그램, 응용 프로그램 프레임워크, 라이브러리, 안드로이드 런타임, 리눅스 커널의 총 5개의 계층으로 분류되어 있다.[4] 각 색의 의미는 빨간색은 C언어로만 사용되었고, 녹색은 C와 C++의 혼용, 파란부분은 JAVA로 구성되었음을 의미한다.

제일 하단에 리눅스 커널(Linux Kernel)이 있고 그 위에는 시스템 라이브러리(Libraries)가 존재한다. 시스템 라이브러리(Libraries)는 응용 프로그램들이 공통적으로 사용하는 라이브러리(Libraries)이다. 이 시스템 라이브러리(Libraries)는 장비의 전반적인 속도를 결정하는 부분으로 C로 작성되어 있다.

안드로이드 런타임(Android Runtime)은 달빅 가상머신(Dalvik Virtual Machine)과 코어 라이브러리(Core library)로 구성되어 있다. 안드로이드는 자바 가상머신을 직접 사용하지 않는다. 대신 모바일 환경에 최적화된 달빅 가상머신(Dalvik Virtual Machine)을 사용한다. 달빅 가상머신(Dalvik Virtual Machine)은 각 프로세스별로 별도의 가상머신이 할당되며, 안정성이 높고 메모리 사용량을 줄여 여러 개의 가상머신도 효율적으로 동작되도록 설계되어 있다. 달빅 가상머신(Dalvik Virtual Machine)은 안드로이드 전용 가상 머신으로 자바 클래스를 바로 실행할 수는 없으며, 자바 클래스 파일을 dex 포맷으로 변환해야만 실행 가능하다.

시스템 라이브러리와 런타임 위에는 안드로이드 AP가 있다. 응용 프로그램들은 시스템 라이브러리, 커널을 직접 호출할 수 없으며 안드로이드 API를 통해 요청해야 한다. 안드로이드 프로그램을 한다는 것은 안드로이드 API를 통해 액티비티 관리자, 뷰, 리소스 관리자 등을 활용한다는 것을 의미한다. 따라서 주로 프레임워크 API를 이용하는 방법을 익힘으로써 안드로이드 기반 앱을 제작할 수 있다. 최상위에는 응용프로그램이 있다. 응용프로그램은 안드로이드 프로그래밍을 통해 만든다.

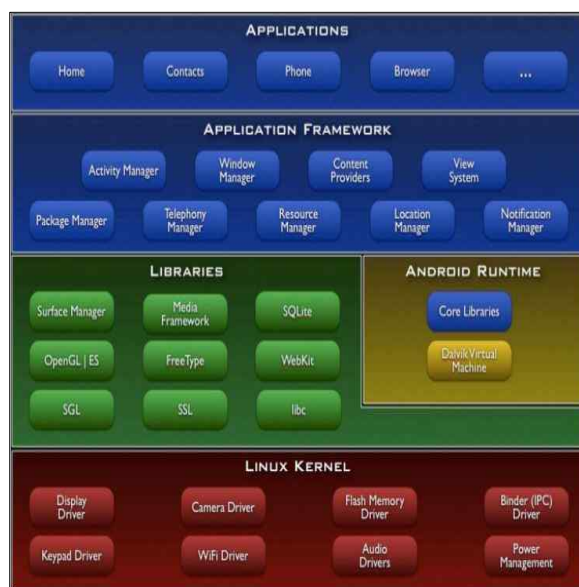


그림 1. 안드로이드 플랫폼  
Fig. 1. Android Platform

## III. 애플리케이션의 설계 및 구현

### 3. 애플리케이션의 설계 및 구현

애플리케이션 메인화면은 그림 2와 같이 랭킹, 즐겨찾기, 동기화 세 가지 기능으로 되어있다.



그림 2. 애플리케이션 메인화면  
Fig. 2. Main-screen of Application

#### 3.1 랭킹(Ranking)

그림 3 랭킹에서는 사용자의 핸드폰에 있는 그룹들을 추가 할 수 있으며, 각 메뉴는 친구, 가족, 회사등 사용자의 핸드폰에 있는 그룹명으로 구성 된다. 각 그룹을 보면 연락을 자주 안하는 고객 순을 중심으로 통화기록을 시간단위로 표현해 연락하지 않은 시간이 높은 랭킹 순으로 보여준다. 메뉴에 오름차순, 내림차순 기능이 있어 연락을 안 한 순으로 또는 자주한 순으로 볼 수 있다.

고객이름을 누르면 고객의 상세 정보창이 뜨며, 통화기록 및 메모를 할 수 있는 기능을 제공하여, 바로 통화와 문자도 보낼 수 있다. 또한 메뉴 제일 뒤에는 그룹추가도 가능하게 만들 수 있어 원하는 그룹을 추가 삭제가 가능하게 구현하였다.

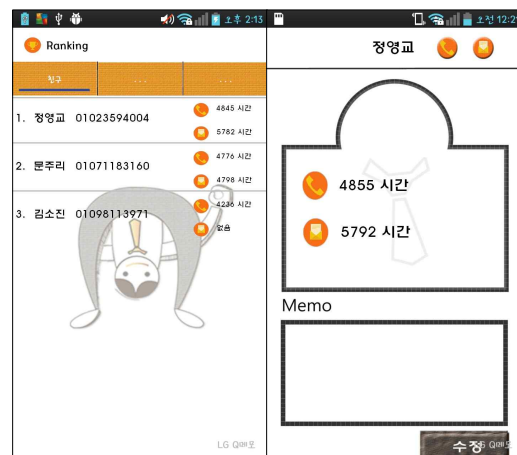


그림 3. 랭킹 기능 및 상세 정보창  
Fig. 3. Ranking function and detailed information

### 3.2 즐겨찾기(BookMark)

영업사원이 고객을 특별관리 하기 위해서 그림 4 즐겨찾기 기능을 추가하여 보다 손쉽게 볼 수 있게 구현하였으며, 랭킹과 마찬가지로 메뉴에 오름차순, 내림차순 기능을 제공하여 연락 하지 않는 기간을 시간으로 보여주는 한편 삭제기능도 추가 하였다.

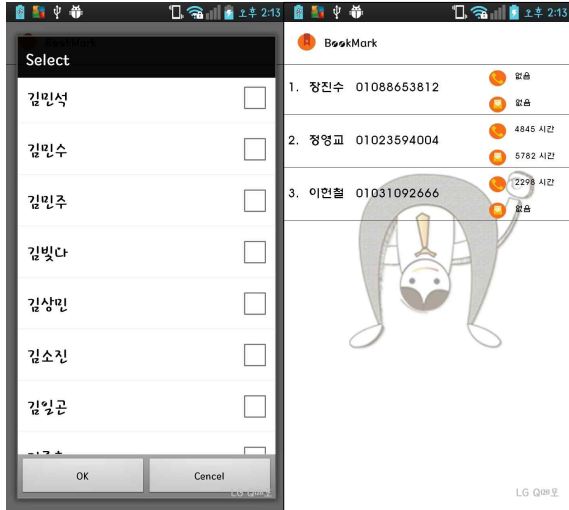


그림 4. 즐겨찾기 기능  
Fig. 4. BookMark Function

### 3.3 동기화(Synchronization)

그림 5 동기화기능은 핸드폰에 탑재된 전화번호부를 서버에 저장하여 사용자가 원할 때 전화번호를 업데이트 할 수 있고 수정된 고객정보 또한 실시간으로 서버에 저장된다. 만약 고객을 실수로 삭제하거나 핸드폰을 분실 했을 경우에도 서버에 저장된 정보를 통해 전화번호부에 고객정보를 다시 복구 할 수 있도록 구현하였다.



그림 5. 동기화 기능  
Fig. 5. Synchronization Function

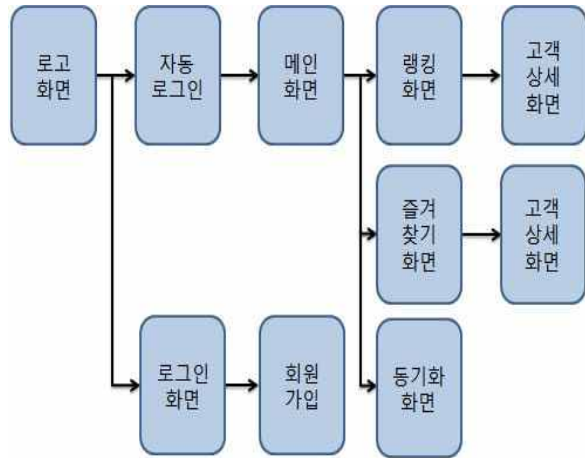


그림 6. 애플리케이션 흐름도  
Fig. 6. Flow Chart of Application

## IV. 결 론

본 논문에서는 영업사원이 고객의 전화와 문자 기록을 시간 단위로 표현하여, 연락 하지 않은 시간이 높은 순으로 연락이 시급한 고객을 먼저 관리하고, 즐겨찾기 기능으로 원하는 고객만을 따로 특별관리 하게 해주며, 핸드폰을 분실했을 경우 동기화 기능과 복구 기능을 통해 연락처의 손실을 막는다.

하지만 실시간으로 알려주는 알림 서비스가 없어 사용자가 애플리케이션을 매일 들어가서 확인해야하는 단점이 있다.

향후 연구계획으로 관리가 필요한 고객들의 리스트를 자동으로 알려주는 알림서비스 기능을 추가할 예정이다.

## ACKNOWLEDGEMENT

본 연구는 BK21+사업, 한국연구재단 기초연구사업 (2012R1A1A2040257), (2013R1A1A2060398), 삼성전자(S-2014-0700-000), 미래창조과학부 및 정보통신기술연구진흥센터의 정보통신방송 연구개발사업 (1391105003)의 일환으로 수행하였음.

## 참고문헌

[1] <http://www.jobkorea.co.kr>  
 [2] <http://www.upkorea.net>  
 [3] T. Y. Kim, J. H. Jung(2006)A study on the mobile Interface Design for the human Network in the social network environment. Korean Society of Design Science, May.2006. 76-77(2 pages)  
 [4] <http://developer.android.com/index.html>