

푸시 서비스를 이용한 모바일 커스터마이제이션 기법

정창훈¹, 예준호¹, 김철진^{1*}
¹인하공업전문대학 컴퓨터시스템과

A Mobile Customization Technique using Push Service

Chang-Hun Jung¹, Jun-Ho Ye¹ and Chul-Jin Kim^{1*}

¹Dept. of Computer Systems and Engineering, Inha Technical College

요약 스마트 워크로 인해 모바일 환경에서 다양한 업무가 수행되고 있다. 이러한 흐름에 따라 다양한 업무 서비스의 제공 및 업무 서비스의 즉각적인 변경을 요구하고 있다. 이에 본 논문에서는 모바일 서비스를 실시간으로 변경하기 위한 모바일 커스터마이제이션 기법을 제안한다. 본 논문의 모바일 커스터마이제이션 기법은 구글의 GCM 푸시 서비스와 안드로이드의 인텐트 기술을 기반으로 커스터마이제이션 아키텍처 및 프로세스를 제안한다.

Abstract Due to smart work, the variety of tasks are performed in the mobile environment. According to the main flow, it is required to provide the various business services and change the business services instantly. Thus, in this paper, we propose a mobile customization technique for changing mobile services in real time. The mobile customization technique of this paper propose the customization architecture and process that is based on GCM's push service of Google and Intent technology of Android.

Key Words : Push Service, Mobile Customization, GCM Service

1. 서론

최근 스마트폰의 사용이 증가함에 따라 사람들 생활에 있어서 모바일 어플리케이션의 사용은 중요한 부분으로 자리 잡고 있다. 이에 따라 모바일 어플리케이션은 실생활은 물론, 기업의 업무처리, 엔터테인먼트 등 다양한 방면으로 편의성을 목적으로 개발되고 있다. 특히, 최근 기업에서는 업무에서 스마트워크를 주제로 모바일 어플리케이션을 이용한 업무처리에 관심이 증가하고 있으며, 이를 통해 업무처리와 모바일 어플리케이션의 결합도 또한 증가하고 있다. 그 중 기업 내 커스터마이제이션을 이용한 업무처리에 모바일 어플리케이션이 쓰이는 빈도가 증가하고 있는 것에 반해, 커스터마이제이션 기법을 모바일 플랫폼에 적합하게 적용되고, 활용되도록 하는 연구는 미흡한 상황이다.

본 논문에서는 모바일 어플리케이션을 기반으로 구글에서 제공하는 GCM(Google Cloud Messaging) 푸시 서

비스(Push Service)를 이용하여 기업 내 특정 업무 분야에서 관리자와 사용자간의 데이터 전달을 통한 커스터마이제이션 기법을 제안한다. 모바일 어플리케이션을 통한 커스터마이제이션의 가장 큰 장점은 관리자가 시간과 공간의 영향을 받지 않고서도, 언제 어디서나 사용자 서비스를 커스터마이제이션 할 수 있다는 것이다. 이에 기업 내 업무처리에 대한 시간과 공간으로 인한 낭비와 불편을 최소화 할 수 있으며, 관리자가 업무 진행 상황에 따라 업무 처리의 방향을 유연하면서도 적재적소로 활용하고 제어할 수 있으므로 업무생산성과 효율성이 증대 될 것으로 기대된다.

본 논문은 다음과 같이 구성한다. 2장에서는 관련 연구를 주제로 GCM 서비스와 GCM서비스를 통한 커스터마이제이션 관련연구를 분석하고, 3장에서는 GCM서비스를 기반으로 하는 모바일 커스터마이제이션 기법을 제안한다. 4장에서는 본 논문에서 제안한 기법을 기업 내 IT고장신고센터 사례에 적용하여 타당성을 검증하고, 5

*Corresponding Author : Chul-Jin Kim(Inha Technical College)

Tel: +82-2-10-6398-7471 email: cjkim@inhac.ac.kr

Received June 17, 2013

Revised July 8, 2013

Accepted September 6, 2013

장에서 결론을 맺는다.

2. 관련 연구

2.1 GCM 서비스

GCM이란 Google에서 무료로 제공하는 메시지 발송 서비스로서, 기존에는 C2DM(Android Cloud To Device Messaging Framework)이라는 이름이었으나 현재는 GCM이라는 이름으로 변경되어 서비스 되고 있다. 이 서비스는 Google 서버를 이용하여 사용자 개인의 디바이스 내에 어플리케이션 단위로 메시지를 전송할 수 있는 서비스 시스템이다. 요즘 카카오톡, 마이피플 및 여러 어플리케이션에서 사용되는 푸쉬(알림)서비스의 한 종류이며, 일반적으로 어플리케이션의 공지사항 또는 홍보등의 목적으로 쓰이고 있다[1].

GCM 서비스를 사용하기 위해서는 우선 단말기의 안드로이드 운영체제가 최소 Android 2.2 Froyo(API Level 8) 이상이 되어야 한다. 그리고 사용자는 Google의 GCM 홈페이지(<http://code.google.com/apis/console/>)에서 API key와 Project Number 2개의 값을 발급받아야 한다[2]. Table 1과 같이 API Key는 개발자가 GCM 서버에 데이터를 보낼 때 사용하는 주소값으로 쓰이고, Project Number는 어플리케이션의 주소값을 GCM 서버에 등록할 때, 그리고 GCM 서버에서 해당 어플리케이션으로 데이터를 받아올 때의 주소값으로 쓰인다. 그리고, GCM 서비스를 적용할 어플리케이션에 라이브러리, 퍼미션, 리시버 그리고 가장 중요한 GCMIntentService등의 GCM 모듈이 적용되어 있어야 한다. 다음으로, GCMIntentService에 의해서 자동으로 생성되어 해당 어플리케이션의 고유 주소값으로 쓰이는 RegistrationId와 개발자와 GCM 서버간 중간단계 역할을 하는 Message Sender모듈이 준비되어 있어야 한다.

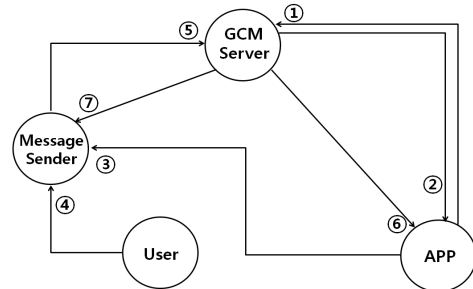
Fig. 1에서와 같이 GCM 서비스를 사용하기 위한 첫 번째 과정은 어플리케이션의 주소값인 RegistrationId를 Message Sender로 보내서 저장시켜야 한다. 두 번째로는 사용자가 Message Sender를 통해서 RegistrationId와 Api Key값을 이용하여 GCM 서버에 데이터를 보낸다. 세 번째로는 GCM 서버에서 Message Sender를 통해 받은 값들을 이용하여 해당 어플리케이션으로 데이터를 전송한다. 이 때 Table 2와 같이 어플리케이션은 GCMIntentService라는 모듈의 가장 핵심적인 onMessage()라는 함수를 통하여 데이터 전달받는다[3]. 최종적으로 GCM서버가 Message Sender에 전송 완료 또는 에러 메시지를 반환하면 데이터 전송 과정은 완료된다.

[Table 1] Example Api key, Project Number and RegistrationId of GCM Push Service

Api Key	AlzaSyDAZpQQHOP9WE68GBKFMhL1jIwNpNgupc
Project Number	123511053724
RegistrationId	APA91bFDfd_HglwBtk0MufJxEfc_Y921794wugP4vEAdUUEJN-Wj6T4w1COUU7y7YJ0S3-WRFSM3MoU11EnbjPjNMye8aTTTD9tWAc hLUMjkgfrx7KienYR69R9NI6yRgeczKZRsoqhTe5vT-be71jtLNBYGfQzxA

[Table 2] Functions of GCMIntentService

onMessage()	Is called when the message is received from GCM server.
onRegistered()	Is called after the application is registered on the GCM server.
onUnregistered()	Is called after the application is deleted on the GCM server.
onError()	Is called when the error is happened on the GCM server.



[Fig. 1] Service Flow of GCM Push Service

- ① RegistrationID 요청 (Project Number 필요)
- ② RegistrationID 발송
- ③ RegistrationID 전달
- ④ 어플리케이션에 전송할 데이터 전달
- ⑤ 데이터와 RegistrationID 전달 (Project Number, API Key 필요)
- ⑥ GCMIntentService를 통하여 데이터 전송
- ⑦ 데이터 전송 완료 및 에러 메시지 전송

사용자는 이러한 GCM 서비스를 통해서 원하는 어플리케이션에 최대 4Kbyte 크기에 데이터를 전송할 수 있고, 이를 통해서 Vibrator, Noti 또는 Message 기능 등 자

신이 원하는 액션을 어플리케이션 사용자에게 제공할 수 있다.

2.2 푸시서비스 기반 커스터마이제이션연구

2.2.1 푸시 서버를 이용한 교육 시스템[4]

연구 [4]에서는 푸시 서비스를 이용하여 학습을 돕는 시스템을 제시한다. 요즘 초등학교 수업시간에 컴퓨터를 활용하는 것은 학생 전체에 대한 일괄적인 수업방식에 불과하므로 이 시스템에서는 학교 웹서버를 활용해 학습자의 학습 성취 여부에 따라 학습을 진행할 수 있는 서비스를 제안한다. 이 서비스는 푸시기반 단계별 원격 시스템으로 채널기법을 이용하여 학습자에게는 자신의 학습 진행 상태에 따라 다음 학습이 주어지게 되며 푸시서버는 데이터베이스에서 학습자의 상태 및 수준을 고려한 학습을 학습자에게 제공해준다. 푸시 기술은 클라이언트와 서버 사이에서 사용자가 인식하지 못하는 시간 단위로 서로 통신을 하고, 서버는 학습자에게 필요한 다음 학습을 사용자의 하드웨어로 전송한다. 그리고 클라이언트는 학습자의 현재 진행 상태를 서버에게 알려 다음 학습에 필요한 정보를 요청한다. 따라서 푸시 기술에 의해서 학습자가 다음 학습단계를 직접 요청하지 않거나 찾아가지 않아도 자동적으로 다음 학습을 수행할 수 있다.

연구 [4]는 푸시 기술을 이용해서 사용자에게 다음 단계의 액션을 취할 수 있게 한다는 것이 본 논문과 유사하지만, 관리자가 직접 커스터마이제이션 할 수 있다는 기술이 포함되어 있지 않다. 즉, 이 시스템은 서버 데이터베이스의 정보만으로 사용자에게 자동으로 제공되는 서비스이다.

2.2.2 푸시 메시지를 이용한 채점시스템[5]

연구 [5]에서는 푸시 메시지를 통하여 자동채점이 되는 시스템을 제시한다. 학습자가 주제별 주관식문제를 응시완료하면 해당 답안정보는 출제자 무선단말기에 푸시 메시지 형태로 즉시 보내지게 된다. 그리고 출제자는 학습자의 답안정보를 받아 무선단말기를 이용하여 학습자가 작성한 답안의 배점을 결정하고 채점결과를 서버에 보낸 후 서버는 이 배점으로 채점결과를 학습자에게 신속하게 알려주는 시스템이다. 객관식문제 같은 경우는 컴퓨터로 채점하여 응시자가 즉각적인 채점결과를 받을 수 있는 것에 비해 주관식문제의 채점결과는 상대적으로 시간이 많이 소요된다. 이러한 단점에 대해 본 논문에서 제시한 커스터마이제이션 기법을 적용시키면 채점시간을 단축시킬 수 있을 것이다. 또한 학습자는 신속한 채점결과를 확인하여 학습능력을 향상시킬 수 있을 것이다. 연

구 [5]는 푸시 기술을 이용하지만 결과적으로 보았을 때 관리자가 푸시 기술을 이용하여 커스터마이제이션을 하는 것이 아니라 학습자의 응시 결과를 확인하기 위해 푸시 기술을 이용하는 것이다.

2.2.3 공간 데이터를 활용한 푸시 서비스[6]

특히 [6]은 고객 또는 사업자의 공간데이터를 활용한 맞춤형 푸시 서비스 제공 방법 및 시스템에 관한 것이다. 이 발명은 특정 단말기가 주기적으로 위치정보제공서버에 현재 위치를 전송하였을 때 서버는 서버 데이터베이스를 바탕으로 특정 단말기에 근처 가맹점이나 유적지 등 특정 공간에 부합하는 정보를 푸시 서비스로 전송하는 시스템이다. 이 시스템은 사용자 위치정보를 토대로 차별화된 고객을 추출하여 그에 맞는 푸시 서비스를 제공하며 더 나아가 쿠폰, SNS서비스, 친구찾기, 아이지킴이 등의 다양한 부가서비스로 확대 응용할 수 있다. 또한 가맹점별 즉시성 쿠폰이나 다양한 메시지 정보의 전송도 가능하므로 사용자와 가맹점 모두에게 편리성을 높일 수 있는 시스템이다. 만약 관리자가 고객의 위치서비스를 이용하여 커스터마이제이션으로 특수한 정보를 제공하는 시스템이라면 본 논문에서의 중점적 연구와 비슷할 것이다. 그러나 푸시 서비스를 이용하여 단순히 고객의 정보를 바탕으로 서버 데이터베이스로부터 정해져있는 정보를 되돌려 받는 서비스이므로, 관리자가 직접 커스터마이제이션을 하는 본 논문의 연구와 차이를 보인다.

2.2.4 비디오 푸시/팝 서비스기반 감시시스템[7]

연구 [7]은 크게 비디오 푸시 서비스(Video Push Service)와 비디오 팝 서비스(Video Pop Service)로 나뉜다. 비디오 푸시 서비스는 특정대상에게 영상을 제공하는 서비스이고, 이와 반대로 비디오 팝 서비스는 푸시된 영상을 가져오는 것을 의미한다. 푸시 서비스는 범죄, 응급, 재난 등의 상황 발생 시 그 사건을 영상으로 촬영 후 특정한 대상에게 전송하는 서비스이다. 이때 GPS의 기능을 이용하여 자신의 위치도 같이 전송하는데, 오차의 범위가 클 수 있으므로 GPS의 오차값을 제거하여 근집화하고 Google Elevation API를 이용하여 GPS의 고도를 보정하며, 주변 감시카메라는 신고자의 위치를 향해 조정이 된다. 또한 영상을 전송할 때에는 영상의 화질 손실문제와 좋지 못할 수도 있는 무선인터넷의 환경 때문에 H.264 코덱(Codec)을 이용하여 영상압축과 복원을 하며, 별도의 인코딩을 거치지 않는다. 이렇게 얻어진 자신의 위치정보와 신고영상은 푸시 서비스를 통하여 특정한 대상에게 보내지게 되며, 이를 받는 대상은 팝 서비스를 이용하여 이 영상을 전달 받는다. 이때의 특정대상은 신고된 영상

을 토대로 사건을 조치할 수 있는 경찰, 자경단, 방범대, 소방서등이 되며 조치가 필요하다고 판단되면 받은 데이터를 가지고 조치를 취하게 된다. 연구 [7]에서는 푸시 서비스를 이용할 때 상대방을 지정하는 방법에 대해서 자동선택 방법과 수동선택 방법을 제시한다. 자동선택은 미리 지정한 사용자에게 보내는 경우와 자동으로 위치를 파악 후 가장 근거리에 있는 대상자에게 보내는 방식이고, 수동선택은 대상자를 직접 선택하여 보내는 방식이다. 그러나 이러한 방식들로 인해서 여러 가지 문제점들이 생길수도 있는데 자동으로 대상자를 판단하는 경우는 필요한 대상자가 아닌 다른 대상자가 서비스를 받게 될 수 있고, 수동으로 대상자를 선택하는 방법은 신고자가 필요한 상황을 정확히 인지하고 대상자를 찾아낸다는 것에 대하여 판단이 어려울 수 있다. 이 점은 위급상황에 대하여 결정적인 결함으로 작용할 수 있는데, 최선의 경우는 신고 후 바로 조치가 되는 것이지만, 최악의 경우 대상자가 서로의 관할을 미루다 조치가 늦어지거나, 서로가 같이 출동하여 인력이 낭비되는 등의 문제가 발생할 수 있다. 이러한 낭비와 손실을 막기 위해서는 신고자와 조치자간에 원활히 커뮤니케이션을 할 수 있는 시스템이나, 신고 및 조치를 관리할 수 있는 시스템이 필요하다 여겨진다.

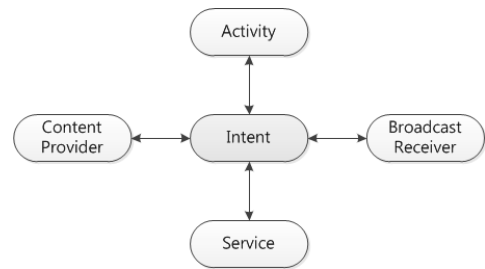
2.2.5 음성통신 서비스를 위한 푸시 서비스앱[8]

연구 [8]은 사용자가 문자 메시지에 익숙하지 않다면 더욱 유용하게 사용할 수 있는데, 특히 노인이나 유아들에게 효과적이며, 스마트폰 환경에 적응하기 힘들어하는 사용자 또한 그 대상이 된다. 이러한 대상들 외에도 가장 효율적으로 사용할 수 있는 곳은 병원으로 볼 수 있는데 담당 의사가 환자에 대한 정보를 음성으로 저장하면 다른 그룹에게 즉각적으로 반응을 보내고, 음성 데이터를 확인한 그룹에서는 다시 질문을 할 수 있는 시스템이다. 또한 당직의사들이 비상상황에서 이 앱을 사용하여 다른 의사들에게 음성을 전송하면 상황에 따라 도움을 줄 수도, 받을 수도 있기 때문에 실시간 알람 기능으로도 사용될 수 있다. 위의 사례들은 다음과 같은 메카니즘으로 작동하게 되는데, 첫째로 음성을 녹음하는 것으로 녹음된 음성은 별도로 Send를 통하지 않더라도 즉각적으로 웹서버로 전송된다. 이때 웹서버는 톰캣과 JSP를 이용하며, 웹서버로 전송된 데이터는 데이터베이스에 저장된 후에, 관련 대상에게 알람 메시지를 보내게 된다. 보내는 과정에서는 GCM의 Sender를 지정하고 Message를 통해 데이터를 보내게 된다. 이렇게 보내는 알람 메시지는 대상자가 받게 되고, URL 접근을 통해 서버에 데이터를 요구하여 저장된 음성을 듣게 된다. 연구 [8]은 푸시 서비스를

이용하여 응용 분야에 이용되는 사례이며, 본 논문에서 제시하는 관리나 개발 측면에서 커스터마이제이션하기 위한 기술로는 차이를 보인다.

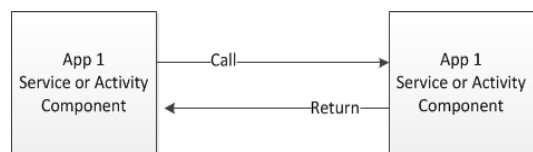
2.3 안드로이드의 액티비티/인텐트

안드로이드의 액티비티(Activity)는 어플리케이션의 흐름을 쉽게 이해하기 위한 하나의 화면 구성이다. 인텐트(Intent)를 이용하여 데이터를 주고받으며 동작하게 되고, 이를 활용하기 위해서는 액티비티 클래스를 상속받게 되며, 일정한 생명주기를 가지고 작동한다. 안드로이드 앱은 여러 액티비티를 가질 수 있는데, 이 사이에서 메시지를 주고받을 수 있게 해주는 것을 인텐트라 한다. 또한 자신의 어플리케이션이 아닌 다른 어플리케이션의 액티비티와 서비스(Service), 브로드캐스트 리시버(Broadcast Receiver), 콘텐츠 프로바이더(Content Provider)를 활성화시키는 역할을 하기도 한다[Fig. 2]. 즉, 그들 사이에서 데이터를 주고받기 위한 용도이다.



[Fig. 2] Intent Structure

액티비티를 통해서 인텐트 메시지를 주고받을 때에는 단순히 전달이 목적인 startActivity()와 실행된 액티비티가 끝난 뒤 특정 값을 반환 받기 위한 startActivityForResult()를 이용하게 된다. 인텐트를 내부적으로 활용하는 데에는 주로 액티비티간에 데이터 전달을 목적으로 한다. 사용 방법은 나와 상대의 액티비티 클래스 정보를 담은 인텐트를 생성하고 startActivityForResult()를 이용하거나 startActivity()를 이용하여 전달 및 실행을 한다. 이러한 경우를 명시적 인텐트(Explicit Intent)라 한다[Fig. 3]. 즉, 상대방을 명확히 하여 데이터를 보내는 것이다.



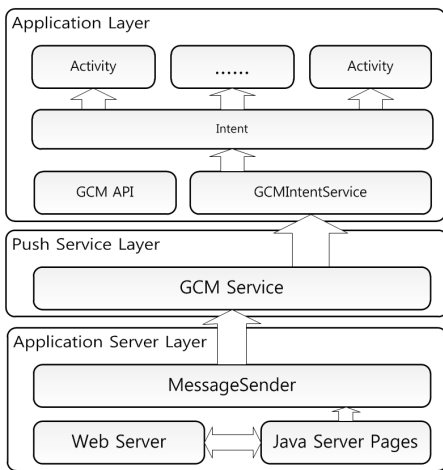
[Fig. 3] Conception of Explicit Intent

3. 푸시 서비스를 이용한 모바일 커스터마이제이션 기법

본 논문의 푸시 서비스를 이용한 커스터마이제이션 기법은 안드로이드 플랫폼 기반으로 구글 GCM 서비스와 안드로이드 인텐트 기술을 이용하여 제안한다.

3.1 푸시 서비스 기반 커스터마이제이션 구조

본 논문에서 제안하는 커스터마이제이션 기법의 계층도는 Fig. 4와 같다. 웹 서버 계층과 푸시 서비스 계층(GCM), 그리고 어플리케이션 계층으로 구분된다.



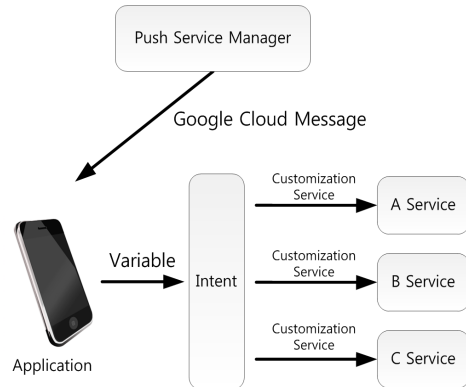
[Fig. 4] Architecture of Customization Technique using Push Service

웹 서버 계층에서는 관리자가 웹서버를 기반으로 GCM 서버에 데이터를 전달하는데, 이 데이터는 목적지 어플리케이션의 주소값과 해당 어플리케이션을 어떻게 커스터마이제이션(제어)할 것인지의 대한 기준이 된다. 푸시 서비스 계층인 GCM 서버에서는 관리자가 웹 서버 계층에서 보낸 데이터를 참조하여, 어떠한 어플리케이션으로 데이터를 전송할지 결정한다. 어플리케이션 계층에서는 GCM 서버로부터 받은 데이터로 인해 인텐트를 담당하는 변수의 값이 제어가 된다. 이 결과로 인텐트되는 액티비티가 변경이 되며, 관리자가 커스터마이제이션을 할 수 있다.

3.2 푸시 서비스 기반 커스터마이제이션 과정

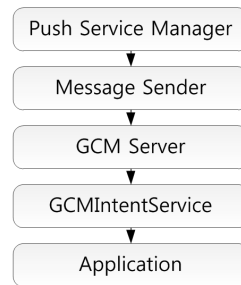
푸시 서비스를 이용한 커스터마이제이션은 관리자가 GCM 서비스를 이용하여 요청하는 어플리케이션으로 데이터를 전달하며, 해당 어플리케이션은 전달받은 데이터

를 기반으로 어플리케이션 내 변수 값이 변경된다. 이와 같이 관리자는 GCM 서비스를 이용하여 해당 어플리케이션의 변수 값을 제어할 수 있고, 이에 따라 안드로이드 인텐트를 제어할 수 있다[Fig. 5]. 또한 어플리케이션은 변경된 변수 값에 의해서 인텐트 되는 액티비티의 흐름이 변경되며, 관리자가 특정한 어플리케이션을 커스터마이제이션 할 수 있다.



[Fig. 5] Customization using Push Service

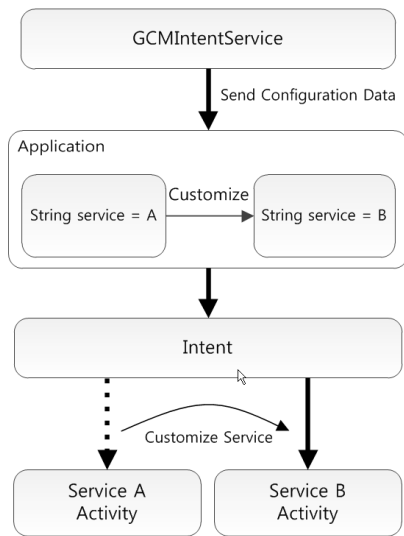
푸시 서버와 어플리케이션 간의 데이터 흐름을 Fig. 6에서와 같이 보면, 관리자는 GCM의 Message Sender를 이용하여 GCM Server로 데이터를 전달하고, 이 데이터를 받은 GCM Server는 GCMIntentService를 이용하여 요청된 어플리케이션으로 데이터를 전송한다.



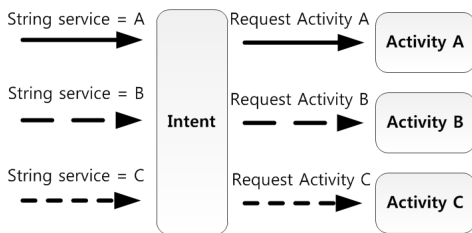
[Fig. 6] Service Flow between Push Server and Application

이렇게 어플리케이션이 GCM 서비스를 통하여 데이터를 받으면, 해당 어플리케이션은 받은 데이터에 의하여 인텐트값 역할을 하는 해당 변수가 변경이 된다.

이 과정을 거치면 Fig. 7, Fig. 8과 같이 변경된 변수값에 따라 해당 어플리케이션의 인텐트 되는 액티비티가 변경이 되며, 이 과정을 통해 관리자가 커스터마이제이션을 할 수 있다.



[Fig. 7] Customization by Configuration Data of Push Service(1)



[Fig. 8] Customization by Configuration Data of Push Service(2)

3.3 푸시 서비스 커스터마이제이션 제어 코드

관리자가 GCM서비스를 이용하여 사용자 어플리케이션을 제어하려면 첫 번째로 JSP, PHP 또는 관리자 어플리케이션을 이용하여 Message Sender에 데이터를 전달해야 한다. 이때 데이터 전달은 Http Get방식을 이용해야 하는데 Fig. 9와 같이 ‘day_Time’, ‘expert’ 라는 데이터를 각각 ‘time’, ‘group’ 이라는 변수에 넣는 형식을 이용하여 Message Sender에 전달해 주면 된다.

```
http://192.168.0.5:8080/jsp/GCMMessageSender.jsp?time=day_Time&group=expert
```

[Fig. 9] Code of Sending Data to Message Sender by Get Method

다음으로는 Message Sender에서 JSP로부터 제공되는 ‘request.getParameter()’ 함수를 이용하여 관리자가 보낸 데이터를 각각 ‘_time’, ‘_group’ 이라는 변수에 받는다. 그 후 GCM 라이브러리로부터 제공되는 ‘Message.Builder.addData()’ 함수를 이용해서 각각 ‘Time’, ‘Group’ 이라는 Key에 담아 GCM 서버로 전송할 수 있다. 이 때 Message Sender는 GCM 서버의 주소값 역할을 하는 Api Key, RegistrationId를 참조하여 데이터를 보낸다[Fig. 10].

```
String _time = request.getParameter("time");
String _group = request.getParameter("group");

Sender sender = new Sender("AlzaSyBzZ8_
y3t7xwifktocxUosUjDwmlSE6PHs");

Message msg;
Message.Builder builder = new Message.Builder();
builder.addData("Time", _time);
builder.addData("Group", _group);
msg = builder.build();

Result result = sender.send(msg, "APA91bF
Dfd_HglwBtk0MufJxEfc_Y921794wugP4vE
AdUUeJN-Wj6T4w1C0UU7y7YJ0S3-WRFS
M3MoU11EnbjPjNMye8aTTTD9tWACHLUMj
kgrfx7KienYR69R9NI6yRgeczKZRsoqHte5v
T-be71jtLNBYGfQzxA", 5);
```

[Fig. 10] Message Sender Code in JSP

이렇게 GCM 서버로 전송이 완료된 데이터는 Message Sender에서 보낸 Api Key와 매칭이 되면서, 어플리케이션 주소값 역할을 하는 Project Number를 참조하여 해당 어플리케이션의 GCMIntentService로 전송이 된다. 해당 어플리케이션이 데이터를 받아들 때에는 GCM 서버로부터 데이터를 받기 위해 백그라운드에서 활성화 되어있는 상태여야만 한다. 그런 다음 GCMIntentService에서는 iterator 인터페이스를 이용하여 GCM서버에서 보낸 Key 값과 그에 매칭되는 데이터를 받아들 수 있다. 이때 if 함수를 이용하여 ‘Time’ 과 ‘Group’ 이라는 이름을 가진 Key의 데이터만 필터링하여 받아 올 수 있고, 그 데이터를 받아 어플리케이션 내에 인텐트값을 변경하기 위하여 사용되는 변수인 ‘GCM_TIME_value’와 ‘GCM_GROUP_value’의 값을 변경할 수 있다[Fig. 11].

```

Bundle b = intent.getExtras();
Iterator<String> iterator = b.keySet().iterator();

while (iterator.hasNext()) {
    String key = iterator.next();
    String value = b.get(key).toString();

    if (key.equals("Time")) {
        GCM_TIME_value = value;
    }
    if (key.equals("Group")) {
        GCM_GROUP_value = value;
    }

    MainActivity.time = GCM_TIME_value;
    MainActivity.group = GCM_GROUP_value;
}
    
```

[Fig. 11] Application Code including Configuration Information

```

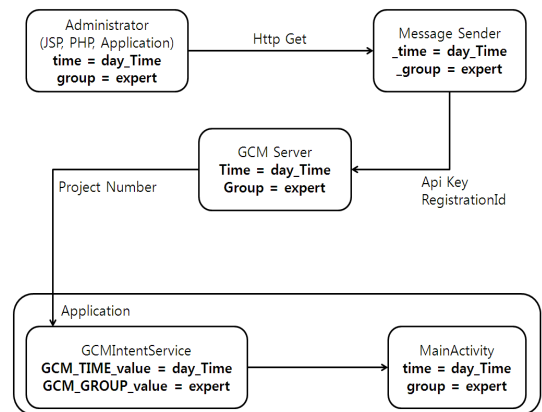
if (time.equals("day_Time")) {
    Intent go = null;
    go = new Intent
        (this, Day_Time_Activity.class);
    startActivity(go);
    finish();
} else if (time.equals("night_Time")) {
    Intent go = null;
    go = new Intent
        (this, Night_Time_Activity.class);
    startActivity(go);
    finish();
} else if (time.equals("dawn_Time")) {
    Intent go = null;
    go = new Intent
        (this, Dawn_Time_Activity.class);
    startActivity(go);
    finish();
}
    
```

[Fig. 12] Customized Application Code by Intent including Configuration Information

그 다음으로 해당 어플리케이션 내에 인텐트값 변경을 하기위하여 사용되는 변수인 'GCM_TIME_value'와 'GCM_GROUP_value'의 값이 관리자 어플리케이션에서 보낸 데이터대로 변경 완료가 되면, 어플리케이션 내에 MainActivity에서 실질적으로 인텐트하기 위하여 사용되는 변수인 'time'과 'group' 변수를 변경하여야 한다. 그

러므로 MainActivity에서 쓰이는 'time', 'group' 변수에 각각 'GCM_TIME_value', 'GCM_GROUP_value' 값을 대입하여 넣어주면 된다. 마지막으로 MainActivity에서 실질적으로 인텐트 역할을 하는 변수인 'time'과 'group' 변수의 값이 관리자가 제어하는대로 변경 완료되면, if 함수를 이용하여 변경된 값에 따라 인텐트를 변경하는 로직을 구현하면 된다. 그 결과로 인텐트되는 액티비티는 자동으로 변경될 것이다[Fig. 12].

이러한 데이터 흐름(Fig. 13)을 통해 관리자는 어플리케이션 사용자 모르게 해당어플리케이션의 인텐트를 변경할 수 있고, 결과적으로 커스터마이제이션을 통해 어플리케이션 제어가 가능하다.

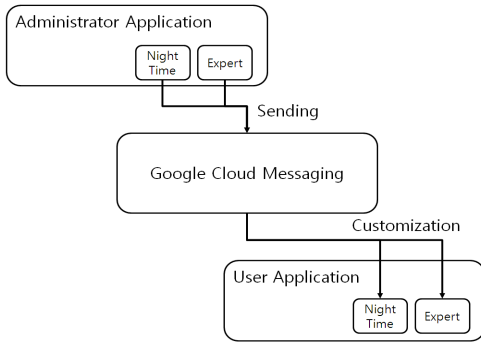


[Fig. 13] Data Flow

4. 실험 및 평가

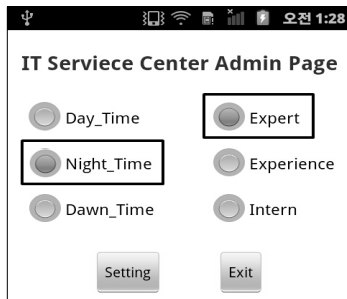
본 논문에서 제안한 푸시 서비스를 이용한 커스터마이제이션 기법은 GCM 서비스를 기반으로 회사 내 IT고장 신고센터 사례에 적용하여 적합성을 검증한다.

회사 내 직원들이 업무처리 도중 컴퓨터, 프린터, 네트워크 등의 문제가 생기면 IT센터로 실시간 신고가 가능한 어플리케이션을 기반으로 한다. IT센터는 주간조, 야간조, 새벽조로 나뉘어서 24시간 3교대 체제로 운영이 되며, 각 조의 하위부서는 인턴직, 경력직, 전문가로 나뉘어져 있다. 그리고 관리자는 관리 어플리케이션을 이용하여 회사 내 직원들이 신고한 건에 대해서 접수 및 처리 부서를 커스터마이제이션하여 서비스를 제공할 수 있다. 또한 이 모든 과정은 GCM 푸시서비스를 이용한 관리자와 사용자 어플리케이션 간의 데이터 전달을 기반으로 한다 [Fig. 14].



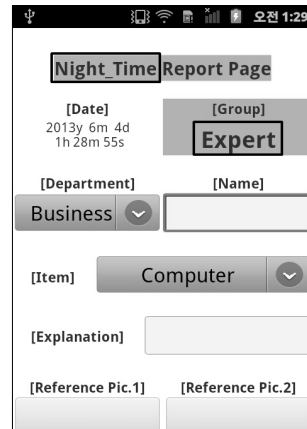
[Fig. 14] Customization Flow for IT Reporting System

관리자는 첫 번째로, 관리 어플리케이션을 이용하여 신고가 들어오는 건에 대한 접수부서를 시간기준으로 하여 설정할 수 있다. 그 다음으로는 하위부서를 선택해야 하는데, 그 기준은 각 부서의 현재 업무량에 따라 결정될 수 있다. 즉, 관리자가 현재 각 부서의 업무량을 파악하여 업무량이 별로 없는 부서를 지정할 수 있다. 예를 들어 저녁 9시경에 인턴직과 경력직의 업무량이 많다는 가정을 하였을 때 관리자는 Fig. 15와 같이 ‘야간’, ‘전문가’를 선택하여 커스터마이제이션을 할 수 있다. 이러한 커스터마이제이션 후에, 회사 내 직원이 신고자 어플리케이션으로 신고를 한다면 야간조 전문가에게 신고내용이 전달될 것이다.



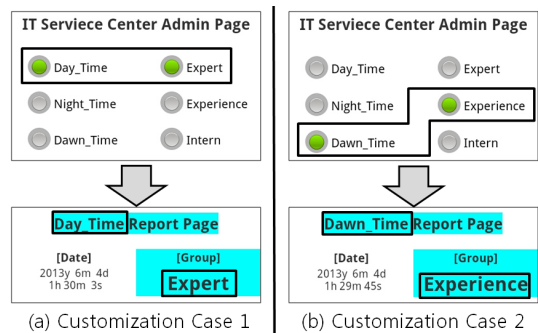
[Fig. 15] Administrator Application UI

관리자가 ‘야간’, ‘전문가’ 부서로 설정하면 해당 데이터들은 GCM 서버로 전달이 되고, GCM 서버에서 Api Key, Project Number, ProjectId 등의 데이터를 참조하여 사용자 어플리케이션으로 데이터가 전송된다. 그 후 사용자 어플리케이션은 전송받은 데이터에 의해서 신고내용이 ‘야간’, ‘전문가’ 부서에 전달될 수 있게 설정이 된다. 이렇게 GCM 서비스기반의 데이터 전달을 통하여 관리자는 본 사례에서 제시한 시간기준과 업무량기준 외에도 다른 기준을 세워서 유연하게 사용자 어플리케이션을 제어할 수 있다.



[Fig. 16] Reporter Application UI

신고자는 신고자 어플리케이션을 통해 IT고장신고가 가능하며, 위에 제시된 Fig. 16은 관리자가 ‘야간’, ‘전문가’로 커스터마이제이션을 했을때의 상태이다. 여기서 한 가지 확인할 점이 있는데, 본 논문에서 제시하는 연구는 신고자 모르게 관리자가 신고자의 어플리케이션을 커스터마이제이션 할 수 있는 기법을 제시하였다. 그러나 Fig. 16과 같이 사용자 어플리케이션의 신고화면을 보면 ‘야간’, ‘전문가’라고 신고 건에 대한 접수부서가 나타나 있어서, 신고자가 자신의 신고 건이 어떤 부서에게 전달되는지 알 수 있다. 이것은 본 사례연구의 이해를 쉽게하기 위해서 임시적으로 구현한 인터페이스일 뿐이다. 실제로는 신고자가 신고하는 내용이 어떤 부서에게 전달되는지 알 수 없다.



[Fig. 17] Customization Case

위의 사례를 통해 안드로이드 플랫폼에서 GCM 서비스를 기반으로 하는 푸시서비스를 통한 커스터마이제이션 기법이 실현 가능함을 검증하였으며, 제안된 연구를 통하여 관리자 및 개발자가 사용자 어플리케이션을 유연하게 커스터마이제이션 할 수 있다는 것을 알 수 있다.

5. 결론 및 향후 연구과제

본 논문은 푸시 서비스를 기반으로 모바일 서비스를 커스터마이제이션하기 위한 기법을 제안하였다. 구글의 GCM 서비스를 기반으로 푸시 메시지에 커스터마이제이션 하기 위한 설정값을 포함하여 전달하면 모바일 서비스에서는 새로운 서비스로 변경될 수 있다. 안드로이드의 인텐트에 해당 설정값을 적용하면 실시간으로 변경될 수 있다. 본 논문에서 제안한 기법을 모바일 IT고장수리 신고 서비스에 적용하여 실시간으로 서비스를 커스터마이제이션할 수 있음을 검증하였다. 본 연구는 운영 중인 서비스를 변경하기 위한 어플리케이션 개발에 활용될 수 있을 것이다. 향후에는 본 기법을 확장하여 개발자에게 제공될 수 있는 플랫폼 기반의 PaaS(Platform as a Service) 서비스에 적용 연구한다.

References

- [1] Leminity's tistory, "GCM(Google Cloud Messaging) for Android" <http://leminity.tistory.com/26>, 2013
- [2] Google apis, <http://code.google.com/apis/console/>
- [3] Android Developers, <http://developer.android.com/reference/com/google/android/gcm/GCMBaseIntentService.html>
- [4] Nam S. H., Oh Y. J., Han D. W., Youn S. D., "A design of Phased Education System for Computer Study on Elemently School using Push-Server Technology", Korea Information Processing Society, p:1085-1088, 2002
- [5] Bang H., Won D. H., Lee J. Y., "A Grading System for Subjective Tests using Push Message in WAP Environments", Korea Information Processing Society, p:1463-1466, 2002
- [6] Lee G. B., Seo B. K., "Method and system customized push service using place data of a customer or store", Korea Intellectual Property Office, 2013
- [7] Kim K. T., Kim S. M., Park E. S., Park C. J., Sung D. S., Lee G. B., "A Smart Surveillance System using Push/Pop Service", Institute of Industrial Technology Journal, p:51-58, 2012
- [8] Baek K. H., "Implementation of Android Apps for Voice Communication Using JSP Server", Enture Journal of Information Technology. p:207-216. 2012

정 창 훈(Chang-Hun Jung)

[준회원]



- 2008년 3월 ~ 현재 : 인하공전 컴퓨터시스템과 재학

<관심분야>

커스터마이제이션 기법, 소프트웨어 공학, 모바일 서비스, 3D 게임

예 준 호(Jun-Ho Ye)

[준회원]



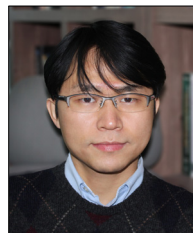
- 2009년 3월 ~ 현재 : 인하공전 컴퓨터시스템과 재학

<관심분야>

모바일 푸시 서비스, 클라우드 컴퓨팅

김 철 진(Chul-Jin Kim)

[종신회원]



- 2004년 2월 : 송실대학교 대학원 컴퓨터학과 (공학박사)
- 2004년 3월 ~ 2009년 2월 : 삼성전자 책임연구원
- 2009년 3월 ~ 현재 : 인하공전 컴퓨터시스템과 조교수

<관심분야>

컴포넌트 기반 개발 방법론, 컴포넌트 커스터마이제이션, 모바일 서비스, 클라우드 컴퓨팅