A Proposed System for Push Messaging on Android

http://dx.doi.org/10.3991/ijim.v10i3.5567

Rasha el Stohy[1], Nashaat el Khamesy [2] and Haitham el Ghareeb[1]
[1] El Mansoura University, El Mansoura, Egypt
[2] Sadat Academy for Management Sciences, Cairo, Egypt

Abstract—**Rapid proliferation of mobile computing technology has massive potential for enhancing the interactions between different partners in educational institutions, particularly in schools. However, how communicating with parents via applications anytime anywhere is quite a challenging demand. On the other hand, the addressed success of the supporting Android features provides a good candidate to aid applications with real time information. Moreover the Android platforms supports efficiently customizable notifications, which allow applications based on Google Cloud Messaging (GCM) to specify the pushed events. In this paper, we look at different push messaging alternatives available for Android. Push messaging provides an important aspect of server to device communication, and we specifically focus on the integration of cloud computing with mobile devices through the use of push-based technologies. The paper presents a proposed notification system; ToParent, that actively provide alerts and messaging between parents and schools. Also, we investigate the performance of the two relevant push technologies for the Android devices, push notification service (PNS) and the short messaging service(SMS) .The proposed system was implemented using Eclipse. The system has been tested and validated. Evaluation have proven that the proposed ToParent enabled better efficiency, which satisfy both parties; parents and school administrations.**

Index Terms—**Mobile Computing, Android Systems, Push Notifications, Real-time Information.**

I. INTRODUCTION

As the society increasingly becomes dependent on technology, schools are investing more time and money in technological means of communication [1]. Although mobile technologies are vital in facilitating parental involvement in schools, the cost involved in the technologies have led to a lack of cell phone used in school administration. Mobile push notifications are an important feature of mobile application and have been widely applied due to the offered of mobile computing services [2]. The most commonly used mobile applications such as Facebook; hang out, a system like Tinder, instant messaging systems including WeChat were all implemented using mobile push notifications [3]. Other supporting services including weather, traffic and travel services were also set up using mobile push notifications to deliver real-time, sensitive and personalized content to users.

There are various technologies that can be used to develop a notification service, Push notifications may be delivered via various methods including standardized protocols (e.g. Server-Sent Events [SSE], the GSM Short Message Service [GSM-SMS], SIPMESSAGE [RFC3428], or OMA Push [OMA-PUSH]), or via browser-specific methods [4].A significant character of mobile push services is the real-time notifications to subscribers based on current contexts such as identity, location, status as well as emotions. However, implementation of mobile push notifications has proved challenging due to some reasons such as they require a third party service like Apple or Google for their implementation[4].

In this research, we focused on push notification service on Android (PNS) because it is an important aspect for application developers, Unlike SMS and Bluetooth; PNS are free and have no coverage problem. Push messaging in many situations is a vital aspect of the usability and functionality of an application. As stated by Gartner [5], the popularity of android Platform means that this area of research highlights challenges that affect a considerable amount of software developers. In addition to the good support of GCM and third party application server which promise high reliability and excellent performance, android is an open platform which provides a good candidate to reuse its components and easier in programming [4].

Accordingly, in this paper, our contribution exposes on proposing a new type of a mobile application that utilizes (PNS) and (SMS) in Android. The proposed system has been designed and implemented using Eclipse integrated with third party services. Our proposed Toparent system is for informing parents of their school updating messages, real-time news, and emergency messages.

This novel is good for parents, administration and service providers. Parents would subscribe to get the information as soon as it is available with minimum power consumption, and service providers would like to enable push service with minimum network bandwidth consumption, that administration need minimum communication cost.

A comparison between the two technologies-PNS and SMS- has been introduced at the end of this paper in regards of efficiency and cost.

This paper organized as follows: the rest of this section explores the related applications, Android push messaging and related techniques, section 3 introduce the proposed system architecture and functions section 4 clearly evaluate the proposed system experimentally and a comparison between the used techniques is conducted. Lastly, section 5 concludes the study by offering some key points focused on in other sections.

II. Related Works

There is a diverse collection of work related to the investigation of the influence of computing approaches in push notifications. Applications can use notifications to greatly improve user's experience. For instance, an email application might notify a user when new message arrive. A news reader application might notify a user when there are new articles to read, a game might notify a user when a friend has signed in or sent, an invitation to play or beaten a high score, and a weather application might notify a user of weather, Alerts. A stock market application might notify the user when certain stock price targets are met [4].

The push messaging is researched across a wide variety of domains; Joseph paper [6] contributes in m-government literature and provides insights on the user perceptions of m-government for emergency notifications. His study also has collaboration for all levels of government including state, local, and federal. He concluded that mobile technology which has grown in the public domain advances the goal of using information and communications technologies (ICTs) to benefit human well-being.

In Bhoi[7] android based child tracking system, he focused on tracking of child's movement to and from school. A particular area was defined for each child separately, so if child is moved outside of define area; an alert message will be given to parents. In addition if child is crying, an alert message will be sent to parents. The panic switch is provided at child module which can be easily used by child to alert his parents. Because of these powerful options, parents own tool to keep the observation on their child even when they can't physically see them.

In work more closed to ours using C2DM technology, Minstrel [8] has been developed as push-based messaging system.

His system utilizes the standard push messaging library for Android, which is called Cloud to Device Messaging (C2DM); subscribers must register at the content publisher to receive the messages.

In similar research, the Bakabs application has been created for Android and iOS, Paniagua et al. [9] used C2DM and the Apple Push Notification Service (APNS) to implement push messaging. His application targets to provide a management tool that allows retrieving information about the web application traffic and then launch or stop cloud instances based on the current load. Other solutions like M*yschoolway*[10], Edu system[11], Sync[12], SNAC[13] applied on android platforms to provide simple effective communication systems to serve the school community in advance for specific events or recurring activities.

Unlike other messaging app the features provided in Toparent mobile app are specially designed and tailored to meet the demand of a massive school community, we proposed push based messaging system combining two push technology, one is the customary SMS and the other is push notification service (PNS) for android platforms. We have focused our work on push notification from web services and third party GCM applications, this notification service can be used in multiple applications and fields at the same time by making little modifications, technologies. Next section will introduce the different push messaging

A. Push notification service(PNS)

Push service is a featured service that is gaining more and more popularity. It allows third-party application servers to actively send data to their installed applications even if the installed application is currently not running. To utilize the push service, the application vendor has to register at the service provider (e.g., Google and Apple). However, the push service provider has no knowledge about the contents of distributed message. Data distributed through the push service are messages that the vendor server wants to notify about. Depending on handling method defined in the installed application, the pushed data may or may not be displayed directly on the screen as notifications [14]. One popular push services provided include the Android Cloud to Device Messaging Framework (C2DM) for Android, the Black-Berry Push Essentials and Push Plus for BlackBerry OS, the Apple Push Notification Service for iOS, and the Microsoft Push Notification Service for Windows Phone [15]. In the first two services on Android and Blackberry, pushed data generally are used for data distribution. While, in the last two services on iOS and Windows Phone, pushed data are specifically for notification and can be displayed directly via the local notification service. Despite the differences in design and implementation, notification services on existing smartphone platforms can be classified into three types:

1. Pop-up Notification: notifies the user by popping up a notification dialog on screen, e.g., the toast Notification on Android and Dialog Notification on BlackBerry OS.
2. Status Bar Notification: adds a notification view as a status bar at the top of screen, e.g., message List on BlackBerry OS 7, Status Bar Notification on Android, and the Notification Center on iOS.
3. Icon Notification: shows notification by making changes on the icon of a sender application in the main menu, e.g., the Badge notification on iOS and the Icon Update function on BlackBerry OS [14].

We will focus on the push service on Android as an example to demonstrate the application of push service using local notification service and the latest service called google cloud messaging (GCM) for a status bar and pop up notification.

B. GCM Server

GCM server is used to develop Android applications where two major processes are involved in the development of client Android applications; these processes include [15]:

1. Creation of a manifest that contains GCM server.
2. Implementation of an appliance using GCM which requires the use of a code to start and stop the registration services. There are two ways to send push notifications using GCM server: channels and advanced targeting. Channels offer a simple and easy to use model for sending pushes, while advanced targeting offers a more powerful and flexible model [16].

For GCM credentials, IDs and tokens are used in different stages of GCM to ensure that all the parties are authenticated and that the message is directed to the correct destination. The sender ID is used in the registration process to identify an Android application that is permitted to send messages to a cell phone. An application ID registers the android application that should receive the messages. This is identified by a package name from the manifest file and guarantees that the messages are directed to the correct android application. A registration ID is also issued by GCM servers to the Android application to permit the reception of posts. Once the Android application has a registration ID; it sends it to the third party application server that employs it to identify each cell phone that is subscribed to the notifications. The registration tag is also related to a particular android application that is run on a definite device. To work with GCM server, the cell phone should at least contain a Google account, especially while running on a version lower than 4.0.4[16]. Authorization is given by the sender authenticating token that is an API key saved on the 3rd party application server for access to the Google services. The API key is included in the header of POST requests that send messages. Fig. 1 depicts this process.

Economical wise, The GCM is a free service and this leads to decreased costs in the messaging services. The GCM server allows up to 4 kb messages and can evidently send small messages including updates and real-time information on weather and sports. For efficient service, GCM requires devices that are running on Android version 2.2 or higher [4].

C. SMS-based Push Service

Short Message Service (SMS) is among the oldest and most widely used methods of mobile communication [17], SMS message interacts with a number of agents throughout its transmission process.

All SMS messages to and from a cell phone go through a Short Message Service Center (SMSC) which responsible of handling any retransmission if a message cannot be deliver is SMSC. If the target mobile phone is out of range or turned off, or if the network is not covered, the SMSC store the message and attempt a retransmission at a later time. SMSC has to find its location before forward a message to the target mobile phone, it must first.

The SMSC does this by querying the Home Location Register (HLC). The HLC is a database which can maintain the information of all the cellular network subscribers. It contains subscriber phone numbers, service plan information, and other subscriber information. When a cell phone is first turned on, or when it moves from the coverage area of one base station to another, the cell phone updates its location in the HLC. Once the SMSC determines the location of the target cell phone by investigate the HLC, it is able to route the SMS message to the correct Mobile Switching Center [18].Switching connections between the cellular network and the cell phones is The Mobile Switching Center (MSC) responsibility.

Each MSC has a Visitor Location Register (VLR) which contains the accurate location (cell) where the mobile phone is located. The MSC first queries the VLR and transfers the SMS message to the correct Base Station System (BSS). Then, the BSS uses transceivers to send SMS message to the target cell phone via a wireless control channel. Thus, a mobile phone can send and receive SMS messages during a voice call or while using data services [18]. Fig. 2 illustrates the entire SMS message transmission process.

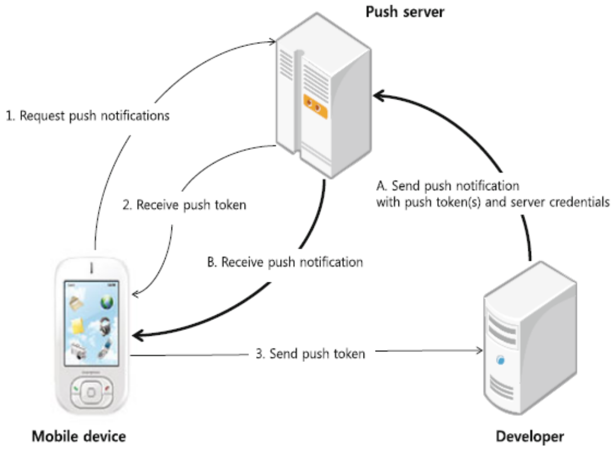

Push server
1. Request push notifications
2. Receive push token
A. Send push notification
with push token(s) and server credentials
B. Receive push notification
3. Send push token
Mobile device
Developer

Figure 1. Process of GCM messaging [16]

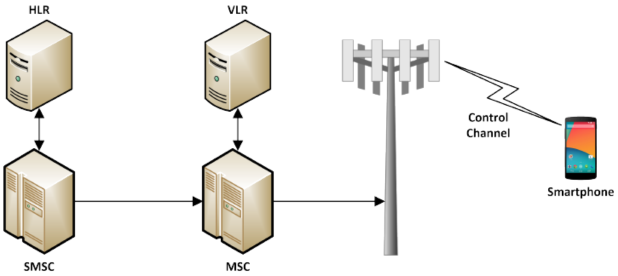


Figure 2. SMS architecture [17]

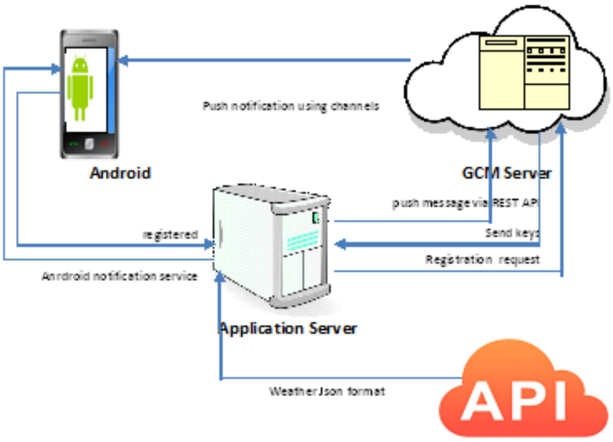


Figure 3. Toparent Architecture

III. OUR PROPOSED SYSTEM

Fig. 3 depict the basic architecture of the proposed notification system; ToParent. Our notification service model is based on two main mechanisms - pull mechanism & push mechanism. Pull mechanism used to register the device at the central storage system, this process achieved via the traditional request/response model where client raises a request and server upon receiving the request appropriately prepares a response and sends it back to the client.

Push mechanism utilized third party channels; Channels were the simplest way to push notifications to all registered devices. Advanced target push to segment of them classified by school name as well, while channel is identified by a string that starts with a letter and consists of alphanumeric characters, underscores, and dashes. Devices start by subscribing to one or more channels, and notifications can later be sent to these subscribers. An installation object represents an instance of our app being installed on a device. These objects are used to store subscription data for installations which have subscribed to one or more push notification channels. In order to push alarm contains current weather information or bad weather circumstances. The android app instantiates JSON, HTTP connections, AsyncTask to get this information. A free weather provider API has been utilized to achieve this task.

A. System Design and Implementation

The proposed system has been developed using software known as the Eclipse which is an integrated development environment (IDE); it contains a base workspace and extensible plug-in system for customizing the environment written mostly in Java [19-23]. The IDE can be integrated with database engine, android services and GCM server, the following sections present brief listings of some major processes.

1) Process of subscribing and verifying

In order to verify parent identity , android has to connect with SQL database instance, for achieving this task successfully,we utilized visual studio web app project. The project acts as a connector between the SQL server and Android server using entity .NET framework and a set of controller that permit a connection being established through HTTP post and get methods[22]. Our database model consisting of three tables: the school table, event table and parent table. android can get data from database tables which returned back with (school_name,phone_number,activation code) in Java Script Object Notation (JSON) format.

2) Process of Configuring Android manifest file

For android client application, it is required to register service and broadcast receivers to use the push notifications. Instantly, before the closing of the application, it is paramount to add XML permissions for other web servers and third parties to be invoked.

3) Process of Push Weather alert

In order to push the weather forecasting from the weather API instantly, the Android service class extend android service which run in the background to connect with the weather API within the school zone. OpenWeatherMap offers several API could be used to get weather data; a specific class that handles this task has been created. The proposed system can alarm parents in bad weather situations as well.

4) Push Events using GCM server

This process accomplished in three steps:

i. a proper permissions needs to ensure that only the targeted Android applications receive the messages [9].

ii. The device requests a PNS in the background by transmitting the package containing the name of the application and the developer's project ID to the Google push server. If the request is allowed, the server provides a registration ID to the device for identifying the application on the device[15].

iii. A developer must compose the push messages us ing JSON format or plain text. The message has a limited life span ranging from zero to four weeks [16]. This means that the Google push server can conditionally maintain the message depending on the expiration time in time the device cannot receive the messages temporarily. It is great concern that the GCM server does not restrict the number of push messages that a developer can transmit in same time it does not need to perform frequent polling to receive the messages and consume little network traffic, Consequently, mobile application requires a small amount of the battery power for its function [15], [23] that's why this technique promises of high efficiency and good performance for our proposed system.

Using our proposed system, developer can push notification to segment of parents, while every device own key ID registered in third party server, The system broadcasts the message using parents database via Intent broadcast, the Oncreate method is called to initialize third party keys as in the following algorithm:

```
public class CParentApplication extends com.orm.SugarApp{
        public      CParentApplication(){}
        @Override
        public void onCreate() {
                super.onCreate();

        Parse.initialize(this,"LBLtXC8j5cs2MZFma6LRrUjwZa7Bk0
Me4uqEx4af","oVmWRnL3uTtedcEPB6W1P2zIcStM6Kiv1JNwHrhI");
                PushService.setDefaultPushCallback(this, Verifi-
cationActivity.class);
```

5) Process of Push messaging using SMSC Server

In this process, The SMS server interacts with subscribed users that use their smart phones, the SMS server interfaces with 3G through an SMS service provider (GSM operator). We utilized SMS gateway that's fetch parent table. A connection established with carrier SMSC. Consequently, SMS has been sent to subscriber phone.

IV. EVALUATION

To assess the feasibility and level of efficiency posed by Toparent, we conducted two evaluations methods, experimental evaluation and comparison between the mentioned techniques.

A. Experimental Settings

In this part an application for Android push notification service has been developed and tested on real android devices as shown in table I.

TABLE I.
ANDROID PLATFORMS USED IN EVALUATION PHASE

Device Type	Platform	Used by
Samsung galaxy ace 4	Android OS, v4.4.2 (KitKat)	Administration ,user(parent)
Samsung Galaxy Note 4	Android OS, v4.4.4 (KitKat)	Administration ,user(parent)
Samsung Galaxy S3	Android OS, v4.3 (Jelly Bean)	Administration ,Developer

The testing set up is based on one PC with 1.4 GHz and 3 GB of RAM used to build the android client application and to create the data base model. Same PC used as a server. The main tasks of this machine were to access data storage and web application in the local server. Same machine used in pushing notification from third party in case of emergency or critical news, as well as pushing SMS via SMS gateway. Fig. 4 illustrate a warning alert holding real time information, Fig. 5 illustrate a reminder alert using SMS, testing phase take place with participants have previous knowledge of mobile applications and mobile communication, but have not previously used the type of application employed in this test, after installing the app on parents', administration staff android devices and testing it during 10 days, we found that experiment results exposed positive facts about Toparent performance.

B. *Comparison between SMS and Push Notification*

We also compared the comparative techniques in two different ways.

1) *Responsiveness*

Efficiency can be measured in a number of ways, one of them is responsiveness. In response to a push request, the gateway should deliver the consequent notification to the client with minimal latency. Because smartphone network connections usually have low bandwidth and are expensive to use, implementations should minimize their bandwidth consumption [24].

From this perspective, we measure the delay of delivering a notification using SMS against GCM. The results are shown in Fig 6. In average, it takes 9.8 second to deliver a notification using SMS. On the other hand, it takes on average 2.1 second to deliver notification using GCM. In conclusion, for reasons of responsiveness, both users and developers liked push notification efficiency particularly in delivering critical and real time alerts.

2) *Cost*

Finally, we compared the cost of push notification via GCM and web service to its counterpart SMS technology.

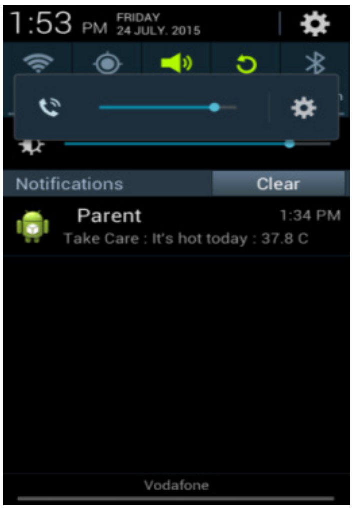
PNS messaging is performed via cellular network or Wi-Fi, the cost of PNS is almost zero. In contrast, the cost of sending SMS offered by our region service provider is about US$ 0.036 per message as depicted in Fig 7.In conclusion, for reasons of cost, the school administration felt satisfaction towards using PNS than SMS.

V. CONCLUSION

This paper presented school push notification system using different technologies, while also presented push messaging across a wide variety of domains.

The research proposes the use of notification based system on Android platform to improve communication in educational institution.

The system is implemented using Eclipse which has the ability of extensibility and allow the reuse of Android Components to push notifications from web services and third party GCM applications, in addition to sending the known SMS. The push notification service can be used in multiple applications simultaneously by making only small modifications. All the services utilized in the proposed solution were evaluated in real Android system devices with good performance results. This research opens up future work to evaluate the extension of the android based on push services to support other new features as well as establish more frameworks in different scenarios.

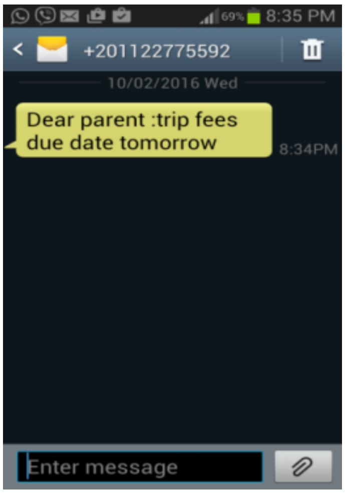

1:53 PM FRIDAY 24 JULY. 2015

Notifications Clear

Parent 1:34 PM
Take Care : It's hot today : 37.8 C

Vodafone

Figure 4. Screenshot of Toparent PNS based Messaging


69% 8:35 PM

+201122775592

10/02/2016 Wed

Dear parent :trip fees due date tomorrow 8:34PM

Enter message

Figure 5. Screenshot of Toparent based SMS Messaging

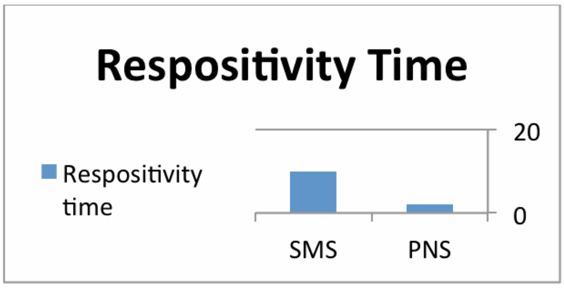


Figure 6. Responsivity time of GCM push notification versus SMS technology.

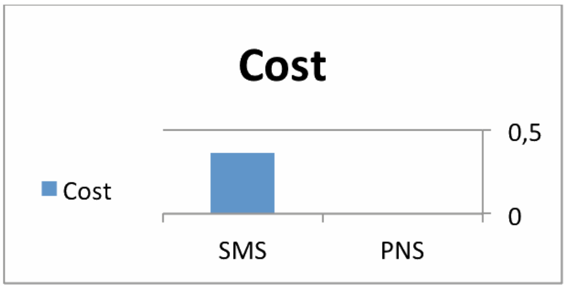


Figure 7. Cost of GCM push notification versus SMS technology

REFERENCES

[1] S.Husain, "Online communication between home and school: Case study: Improving the usability of the Unikum e-service in the primary schools of Tierp municipality", M.S. thesis, Uppsala university, 2012.

[2] R .Roger.;V. Wright, "Assessing technology's role in communication between parents and middle schools", *Electronic Journal for the Integration of Technology in Education*,2008,Vol 7,pp. 36-58.

[3] P.Holleis, M.Wagner, S.Böhm, & j.Koolwaaij, "Studying mobile context-aware social services in the wild", *Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries*,2010, pp. 207-216 http://dx.doi.org/10.1145/1868914.1868941

[4] M. Gusev and S. Ristory, "alert notification as service", Croatia, 2014. http://dx.doi.org/10.1109/mipro.2014.6859584

[5] Gartner, Inc. the world's leading information technology research and advisory company. http://www.gartner.com/technology/about.jsp.

[6] R. C.joseph and R. L.Lee, "M-Government foe Emergency Notification", *International Journal of E-Services and Mobile Applications*, ,vol. 7, no. 1, pp. 50-64,2015. http://dx.doi.org/10.4018/ijesma.2015010103

[7] N.Bhoi, Rohit, & S.B Somani, "Android Based Tracking System". *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 3,No.7 ,pp. 1-5,2015.

[8] M.Hauswirth, Pondnar, "Mobile Push: Delivering Content To Mobile Users", *Proceedings Of 22Nd International Conference On Distributed Computing Systems Workshops*, 2002, Vienna, pp. 563-568.

[9] C. Paniagua, S. N. Srirama & H. Flores, "Bakabs: Managing Load of Cloud-Based Web Applications from Mobiles", *Proceedings of the* 13*th International Conference on Information Integration and Web-based Applications and Services*, ACM, New York,2011, pp. 485-490. http://dx.doi.org/10.1145/2095536.2095636

[10] "Free Messaging App for School Administrators | SchoolWay", Myschoolway.com, 2016. [Online]. Available: https://www.myschoolway.com/administrators. [Accessed: 17- Apr- 2016].

[11] Edupage.org, 2016. [Online]. Available: https://edupage.org. [Accessed: 17- Apr- 2016].

[12] "Sync", Sync.com.my, 2016. [Online]. Available: http://www.sync.com.my. [Accessed: 17- Apr- 2016].

[13] "SNAC: Social Networks and Archival Context", Socialarchive.iath.virginia.edu, 2016. [Online]. Available: http://socialarchive.iath.virginia.edu. [Accessed: 17- Apr- 2016].

[14] X.Zhi, S.Zhu, "Abusing Notification Services on Smartphones for Phishing and Spamming", 2013, pp.1-3.

[15] I.Warren, A.Meads,S. Srirama., T.Weerasinghe & Paniagua, "Push Notification Mechanisms for Pervasive Smartphone Applications ",*Pervasive Computing, IEEE*, 2014,*13*(2), pp. 61-71. http://dx.doi.org/10.1109/MPRV.2014.34

[16] R.K. Lee &S.Lee, "Punobot: Mobile Botnet Using Push Notification Service in Android", doi: 10.1007/978-3-319-05149, 2014.

[17] E. Dondyk, "SPS: AN SMS-BASED PUSH SERVICE FOR ENERGY SAVING", M.S. thesis, Central Florida, 2014.

[18] "Ozeki messaging software", 2007. [Online]. Available: http://www.ozekisms.com/. [Accessed: 05- Feb- 2016].

[19] B.stefan, "Analysis of the Android Architecture ", Helmholtz-Gemeinschaft, 2010, pp.3-10.

[20] B. Lillian ,A.Oliveria, "CodeDroid: CodeDroid: A Framework to Develop Context-Aware Applications",*International proceeding of the fourth conference on Advances in Human-oriented and Personalized Mechanisms*, *Technologies, and Services*, 2011.

[21] A.G.Parada, & L.B. Brisolara, A model driven approach for Android applications development, "*International processding of IEEE on Computing System Engineering (SBESC)*", 2012, pp .192-197.

[22] L.Vogel, "Android Development Tutorial", 2013, Helmholtz.

[23] A.Joseph,D. Lauren& C.Shane, "Advanced Android Application Development" ,Addison-Wesley professional, p.200, 2011.

[24] J.Hansen, ,M. Gronli, ,G. Ghinea, "Towards cloud to device push messaging on Android: technologies, possibilities, and challenges", 2012. http://dx.doi.org/10.4236/ijcns.2012.512089

AUTHORS

Eng. Rasha el Stohy is a researcher at faculty of Computers and Information, Department of Information Systems, she works as lecturer assistance and interested in mobile computing, mobile learning, mobile services, cloud computing management information systems.ratsh@hotmail.com.

Dr. Nashaat el Khamesy is a professor at Sadat Academy for Management Sciences, Department of Information Systems, He did his Ph.D. in Software Engineering at Ohio, USA, and he is interested in computer distributed systems &Networks, Database management systems and Operating systems.wessasalsol@gmail.com.

Dr.Haitham el Ghareeb is a Lecturer at Faculty of computers and information , Department of Information Systems, He is interested in e-learning, enterprise architecture, information architecture, and software architecture, especially in Service-Oriented Architecture (SOA), Business Process Management (BPM), Business Process Management Systems (BPMS), Information Storage and Management, Virtualization, Cloud Computing, Big Data,helghareeb@gmail.com.

Submitted 12 February 2016. Published as resubmitted by the authots 24 April 2016.