

# CLOUD MESSAGING FOR ANDROID IS A PUSH NOTIFICATION SERVICE

B.Dhivya<sup>1</sup> G.Lakshmi Prabha<sup>2</sup> P.Nivethitha<sup>3</sup>, K.Kala<sup>4</sup>

<sup>123</sup>UG Scholar, <sup>4</sup>Assistant Professor

<sup>1234</sup>Department of Computer Science and Engineering

<sup>1234</sup>P.S.R.Rengasamy College of Engineering for Women, Sivakasi

Email: jeyprabha.463@gmail.com, divyapriya994@gmail.com

, nivethithapurushothaman@gmail.com kalakannan.art@gmail.com

**Abstract** — Cloud Messaging for Android is a push notification service that allows sending data from server to user's Android mobile devices. Push messaging provides an important aspect of server to device communication. This service provides a simple, lightweight mechanism that servers can use to tell mobile applications to contact the server directly, to fetch updated application or user data. The Cloud Messaging service handles all aspects of queuing of messages and delivery to the target application running on the target device. Applications on an Android device do not need to be running to receive messages. This service will wake up the application as long as the application is set up with the proper broadcast receiver and

permissions. The application might post a notification, display a custom user interface, or silently sync data. It requires devices running Android 2.2 or higher. It uses an existing connection for Google services. This requires users to set up their Google account on their mobile devices. Each notification message size is limited to 1024 bytes, and Google limits the number of messages a sender sends in aggregate, and the number of messages a sender sends to a specific device. Thus in near future android mobiles will be in the hands of people in companies, schools, colleges, hospitals, banks, shops. Thus using this app, message from the superior, if posted to the cloud will be sent as push messages to the registered persons.

## INTRODUCTION:

Cloud computing is typically defined as a type of computing that relies on *sharing computing resources* rather than having local servers or personal [devices](#) to handle [applications](#). In cloud computing, the word [cloud](#) (also phrased as "the cloud") is used as a metaphor for "*the Internet*," so the phrase *cloud computing* means "a type of Internet-based computing," where different services — such as servers, storage and applications — are delivered to an organization's computers and devices through the Internet. In Cloud to device push messaging on android we are using GCM (Google Cloud Messaging) which allows 3rd-party application servers to send messages to their Android applications. The [GCM Cloud Connection Server](#) is used to send messages to the user's device. An Android application on an Android device doesn't need to be running to receive messages. The system will wake up the Android application via Intent broadcast when the message arrives, as long as the application is set up with the proper broadcast receiver and permissions. It does not provide any built-in user interface or other handling for message

data. GCM simply passes raw message data received straight to the Android application, which has full control of how to handle it. It provides a simple, lightweight mechanism that servers can use to tell mobile applications to contact the server directly, to fetch updated application or user data. It keeps the messages in the server store, and delivers them when the device comes back online. It handles all aspects of queuing of messages and delivery to the target application running on the target device.

## CLOUD COMPUTING:

When you store your photos online instead of on your home computer, or use webmail or a social networking site, you are using a "cloud computing" service. If you are an organization, and you want to use, for example, an online invoicing service instead of updating the in-house one you have been using for many years, that online invoicing service is a "cloud computing" service. Cloud computing refers to the delivery of computing resources over the Internet. Instead of keeping data on your own hard drive or updating applications for

your needs, you use a service over the Internet, at another location, to store your information or use its applications. Doing so may give rise to certain privacy implications. For that reason the Office of the Privacy Commissioner of Canada (OPC) has prepared some responses to Frequently Asked Questions (FAQs). We have also developed a Fact Sheet that provides detailed information on cloud computing and the privacy challenges it presents. Cloud computing is the delivery of computing services over the Internet. Cloud services allow individuals and businesses to use software and hardware that are managed by third parties at remote locations. Examples of cloud services include online file storage, social networking sites, webmail, and online business applications. The cloud computing model allows access to information and computer resources from anywhere that a network connection is available. Cloud computing provides a shared pool of resources, including data storage space, networks, computer processing power, and specialized corporate and user applications.

## **OBJECTIVE:**

To help developers send data from servers to their applications on Android devices. It provides a simple, lightweight mechanism that servers can use to tell mobile applications to contact the server directly, to fetch updated application or user data. It keeps the messages in the server store, and delivers them when the device comes back online. It handles all aspects of queuing of messages and delivery to the target application running on the target device. It uses the existing connections to the Google servers. This connection is highly optimized to minimize bandwidth and battery consumption.

## **EXISTING SYSTEM**

The existing system includes poll messaging where the client and server should be always connected. This results in more battery consumption and increased bandwidth. Also the system does not include cloud thus the client and server are directly connected. So once the message is sent is not received to the client due to some network problems the data will be lost. The existing system involves the student to come to college and view the marks. The marks

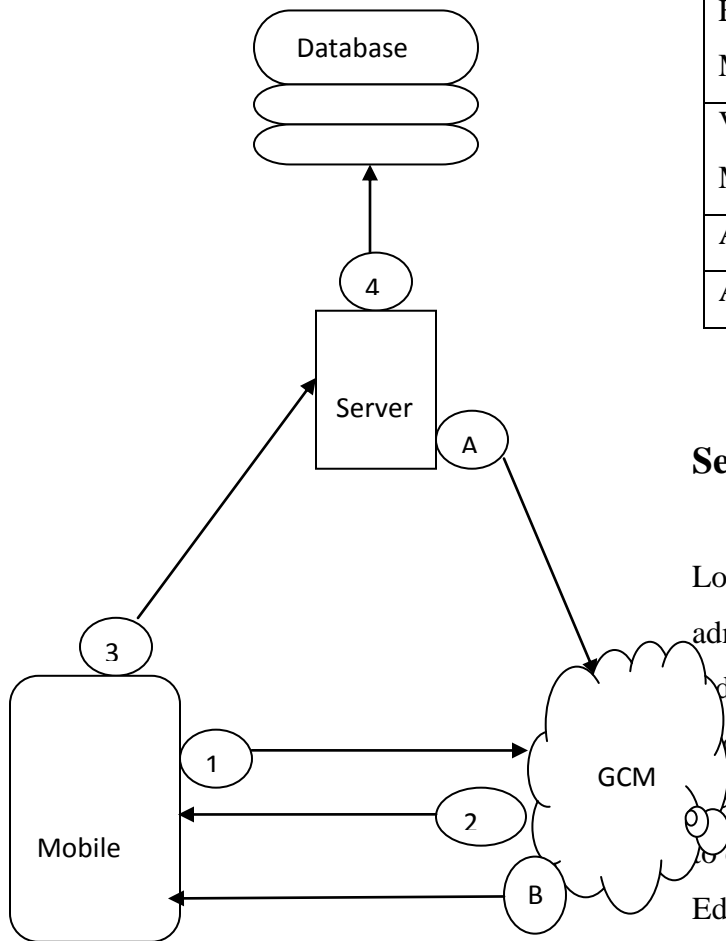
are given as mark sheet which is in paper format. The details of symposium are given as announcement or put in notice board and is not known to parents. The parents may not know about the students daily attendance It is time consuming. It is difficult to know about others marks. It is difficult to know about student activities by parents. The results have to be seen in college website or directly in the college and the parents will not be notified of the result date and time.

### **PROPOSEDSYSTEM**

The proposed system is very easy to operate. The main advantages of proposed system are minimizing the network bandwidth and battery consumption. It provides a simple, lightweight mechanism for the servers to send push messages to android devices. The proposed systems uses GCM server to send messages, a Google service thus provides tight security to data. C2DM allows third-party developers to use the same service the Google apps do. It uses existing connections for Google services. This requires the users to sign into their Google account on Android. An application doesn't need to be

running to receive data messages The GCM service handles all aspects of queuing of messages and delivery to the target. Android application running on the target device. Information can be send quickly and easily. The protection of data from either accidental or unauthorized, intentional modification, destruction is possible. It is adaptability. So it can be reprogrammed to do different tasks. The proposed system is android app which can be installed in all users android mobile devices. The user have to register with the college server to get marks and attendance details from college. The user can send their query or feedback to the college server. The queries will be answer and all the student activities will be known to parents. This app is time saving and user friendly. An application doesn't need to be running to receive data messages. Information can be send quickly and easily. The protection of data from either accidental or unauthorized, intentional modification, destruction is possible. It is adaptability. So it can be reprogrammed to do different tasks. The student activities are monitored by the parents.

**ARCHITECTURE:**



DELETE MARKS	STUDENT	ALL MARKS
EDIT MARKS	STUDENT	ATTENDANCE
VIEW MARKS	STUDENT	SYMPOSIUM
ADD SYMPOSIUM		QUERIES
ATTENDANCE		

**Server Side (PHP):**

Login – Allows only the authenticated admin to login and update student database.

Add student marks– This module allows to add a new student marks in database.

Delete student marks– This module allows to delete a new student marks from database.

Edit student marks– This module allows to edit a student marks from the database.

View student marks– This module allows to view all student marks and send to parents mobile devices.

Add symposium – This module allows to add a new symposium events organized in college.

**MODULES:**

SERVER SIDE	CLIENT SIDE
LOGIN	LOGIN
ADD STUDENT MARKS	VIEW MARKS

Attendance – This module allows to maintain students daily attendance and send to parent mobile.

### **Client side (Android) :**

Login –The user have to login the app to get registered with GCM and server.

View marks –The user can view the marks obtained by their child.

All marks– The user can view all students marks in the class.

Attendance –The user can view the attendance percentage of their child.

Symposium –The user can view the symposium details of the college.

Queries – The user can send feedback or queries to the college server.

### **CONCLUSION:**

To create a real time application oriented project which will be helpful in our own institution that will also be showing a better technological improvement as we are Professional Engineering students. To stop running back of postal letters for the report cards.

### **REFERENCE:**

M. F. Bulut, Y. S. Yilmaz, and M. Demirbas, “Crowdsourcing locationbased

queries,” in PerCom Workshops, 2011, pp. 513–518.

M. F. Bulut, Y. S. Yilmaz, M. Demirbas, N. Ferhatosmanoglu, and H. Ferhatosmanoglu, “Lineking Crowd sourced line wait-time estimation using smartphones,” in MobiCASE, 2012, pp. 205–224.

B. I. Aydin, Y. S. Yilmaz, M. F. Bulut, and M. Demirbas, “Crowdreply: A crowdsourced multiple choice question answering system,” ACM/IEEE IPSN '13 3rd International Workshop on Mobile Sensing: The Future, brought to you by Big Sensor Data, 2013.

B. Sanz, I. Santos, C. Laorden, X. Ugarte-Pedrero and P.G. Bringas, “On the automatic categorisation of android applications”, Proceedings of the Consumer Communications and Networking Conference, (2012).

C. Fernandes, Y. N. Kok and H. K. Boon, “Development of a convenient wireless control of an autonomous vehicle using apple iOS SDK”, Proceedings of the IEEE Region 10 Conference on TENCON, (2011).