



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Προηγμένα Συστήματα Πληροφορικής»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Αξιοποίηση της υπηρεσίας Google Cloud Messaging (GCM) στη σχεδίαση και την ανάπτυξη εφαρμογής άμεσης ανταλλαγής μηνυμάτων για κινητές συσκευές με λειτουργικό σύστημα Android Creating a fully featured Android instant messaging Android application through Google Cloud Messaging (GCM) service
Όνοματεπώνυμο Φοιτητή	Γεωργιόπουλος Κυριάκος
Πατρώνυμο	Νικόλαος
Αριθμός Μητρώου	ΜΠΣΠ/ 13021
Επιβλέπων	Αλέπης Ευθύμιος, Επίκουρος Καθηγητής

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

(υπογραφή)

(υπογραφή)

Όνομα Επώνυμο
Βαθμίδα

Όνομα Επώνυμο
Βαθμίδα

Όνομα Επώνυμο
Βαθμίδα

Περίληψη

Στη σύγχρονη κοινωνία της γνώσης και της αμφίδρομης διανθρώπινης επικοινωνίας, η πληροφορία αποκτάται, επεξεργάζεται, αποθηκεύεται, ανακτάται, μεταδίδεται και διαχέεται σε πραγματικό χρόνο και με ελάχιστο κόστος. Κατά την τελευταία δεκαετία σημειώνεται λοιπόν μία αλματώδης άνοδος της χρήσης των σύγχρονων εφαρμογών άμεσης ανταλλαγής μηνυμάτων. Συγκεκριμένα, η λειτουργία της ανταλλαγής μηνυμάτων σε πραγματικό χρόνο έχει πλέον υπερισχύσει της επικοινωνίας μέσω ηλεκτρονικού ταχυδρομείου σε προσωπικό επίπεδο, ενώ έχει αρχίσει να αποκτά έδαφος και σε επαγγελματικές συνομιλίες. Οι πιο δημοφιλείς εφαρμογές άμεσης ανταλλαγής μηνυμάτων για έξυπνες συσκευές είναι πλέον εμπλουτισμένες με επιπρόσθετες λειτουργίες, πέραν της αποστολής και λήψης κειμένου, οι οποίες επιτρέπουν στους χρήστες τους την ανταλλαγή φωτογραφιών, βίντεο, εικονιδίων με συγκεκριμένη έκφραση προσώπου για δήλωση διάθεσης και ορισμένων κοινών τύπων αρχείων, όπως είναι φέρ' ειπέιν τα έγγραφα Word και τα υπολογιστικά φύλλα Excel.

Στο πλαίσιο της παρούσας μεταπτυχιακής διατριβής ανεπτύχθη μια ολοκληρωμένη Android εφαρμογή άμεσης ανταλλαγής μηνυμάτων μέσω Google Cloud Messaging (GCM)· τουτέστιν ένα πλήρες σύστημα αμφίδρομης επικοινωνίας σε πραγματικό χρόνο που αξιοποιεί την ομώνυμη δωρεάν υπηρεσία της Google και βρίσκει εφαρμογή σ' έξυπνες συσκευές με λειτουργικό σύστημα Android. Η εν λόγω υπηρεσία, η οποία περιγράφεται λεπτομερειακά επί της παρούσας πραγματείας, προσφέρει έναν σταθερό διάλο-κάνάλι ανταλλαγής πληροφοριών, μέσω του οποίου μεταδίδονται άνευ χρηματικού αντιτίμου, αφενός μεν τα μηνύματα που αποστέλλει ένας Διακομιστής-Εξυπηρετητής σε μια ή πολλές συζυγείς και πιστοποιημένες GCM Εφαρμογές Χρήστη-Πελάτες – ήτοι τα κατάντη GCM μηνύματα, αφετέρου δε τα μηνύματα που αποστέλλει μια GCM Εφαρμογή Χρήστη-Πελάτη προς έναν δεδομένο και οικείο Διακομιστή-Εξυπηρετητή Εφαρμογής – δηλαδή τα ανάντη GCM μηνύματα.

Συμφώνως προς τις επιταγές των καιρών, η εκπονημένη Android Εφαρμογή GCM παρέχει στους χρήστες τη δυνατότητα άμεσης ανταλλαγής γραπτών μηνυμάτων και εικόνων σ' ένα ευπροσάρμοστο, μινιμαλιστικό και χρηστικό περιβάλλον διεπαφής, το οποίο έχει αναπτυχθεί επί τη βάση του συνόλου των σχεδιαστικών κανόνων του Material Design της Google.

Abstract

In the contemporary society of knowledge and two-way interhuman communication information is acquired, processed, stored, retrieved, transmitted and spread in real time and at minimal cost. During the last decade, an exponential increase in the use of up-to-date instant messaging applications has been noted. Specifically, the function of exchanging messages in real time has currently prevailed over communication via electronic mail on a personal level, while it has started gaining ground in professional talks as well. The most popular instant messaging applications for smart devices are enriched with additional functions, besides sending and receiving text, which allow the users to exchange photographs, videos, emoticons (icons with facial expressions to show emotion) and some common type files, such as, for example, Word documents and Excel spreadsheets.

In the wider context of the present postgraduate dissertation a fully featured Android instant messaging application through Cloud Messaging (GCM) has been developed; namely a complete system of two-way communication in real time which optimizes the homonymous free service of Google and applies it to smart devices with an Android operating system. The particular service, which is described thoroughly in this dissertation, offers a steady channel of exchanging information through which the following can be transmitted at no cost; on the one hand, the messages that are sent by an Application Server to one or more conjugate and certified GCM client applications – namely the downstream GCM messages, on the other hand, the messages that are sent by a GCM client application to a given and familiar Application Server– namely the upstream GCM messages.

In accordance with the demands of the times the devised GCM Android Application provides the users with the ability to instantly exchange text messages and images in an adaptable, minimalistic and easy to use Graphical User Interface GUI environment which has been developed on the basis of the totality of the designing rules of Google's Material Design.

Περιεχόμενα

	Σελ.
Εισαγωγή.....	6
Αναλυτική Περιγραφή της υπηρεσίας Google Cloud Messaging (GCM).....	7
Η διάρθρωση μιας GCM Εφαρμογής.....	7
Η κλιμακωτή οικοδόμηση μιας GCM Εφαρμογής Χρήστη για έξυπνες φορητές συσκευές και τηλεοράσεις με λειτουργικό σύστημα Android OS.....	24
Η ενδεδειγμένη διαμόρφωση του αντίστοιχου XML αρχείου ρυθμίσεως παραμέτρων AndroidManifest.xml για τον ορισμό και την περιγραφή της συμπεριφοράς μιας GCM Android Εφαρμογής Χρήστη.....	24
Η εγγραφή μιας GCM Android Εφαρμογής Χρήστη στην Υπηρεσία Νέφους Instance ID – Έκδοση, Ανανέωση των συζυγών Τεκμηρίων Γνησιότητας.....	29
Αποστολή Ανάντων Μηνυμάτων.....	30
Η Android Εφαρμογή Χρήστη για την άμεση ανταλλαγή μηνυμάτων μέσω GCM.....	31
Αρχική Οθόνη – Splash Screen.....	31
Εργασία με το AsyncTask.....	32
Οθόνη Εγγραφής – Registration Screen.....	34
Κεντρική Οθόνη – Καρτέλες: Χάρτης, Λίστα Έγγραμμων Επαφών-Φίλων Χρήστη, Κατάλογος Συνομιλιών Χρήστη.....	43
Οθόνη άμεσης αποστολής και λήψης μηνυμάτων κειμένου και φωτογραφιών – Instant Messaging Screen.....	50
Κατακλείδα και Ερευνητικές Προτάσεις επί τη βάσει της παρούσας πραγματείας.....	54
Βιβλιογραφικές Αναφορές.....	55

Εισαγωγή

Θυμάστε τότε που ένα τηλέφωνο ήταν απλά ένα τηλέφωνο; Όταν βασιζόμασταν σε σταθερές επίγειες γραμμές; Όταν τρέχαμε για το τηλέφωνο αντί απλά να το βγάλουμε από την τσέπη μας; Όταν χάναμε τους φίλους μας σ' ένα γεμάτο στάδιο και περιμέναμε ώρες στις εξόδους για να τους ξαναδούμε; Όταν ξεχνούσαμε τη λίστα με τα ψώνια και έπρεπε να βρούμε ένα καρτοτηλέφωνο ή να γυρίσουμε στο σπίτι;

Εκείνες οι μέρες αποτελούν μακρινό παρελθόν. Σήμερα, τέτοια προβλήματα λύνονται εύκολα με μία ταχεία κλήση που πραγματοποιούμε με το πάτημα ενός πλήκτρου ή μ' ένα απλό μήνυμα κειμένου όπως «Που είσαι;», «Θύρα 7;» και «Νιφάδες βρώμης, κακάο και;»

Οι έξυπνες συσκευές μας εγγυώνται την ασφάλεια και την επικοινωνία μας με όλους. Μπορούμε τώρα να κινούμαστε ελεύθερα και να βασιζόμαστε σ' αυτές όχι μόνο για να διατηρούμε την επαφή με φίλους, συγγενείς και συνεργάτες, αλλά επίσης για να μας πουν πού να πάμε, τι να κάνουμε και πώς να το κάνουμε. Ακόμα και πράγματα που γίνονται αποκλειστικά μέσα στο σπίτι φαίνεται να περιστρέφονται γύρω από ένα κινητό τηλέφωνο.

Η κοινότητα της ανάπτυξης εφαρμογών έξυπνων συσκευών βρίσκεται σε σημείο ανατροπής. Ζούμε σε μια συναρπαστική «έξυπνη» κι «ασύρματη» κοινωνία που εξελίσσεται ραγδαία και γεννά συνεχώς νέες ανάγκες στα μέλη της. Οι «έξυπνοι» χρήστες κινητών τηλεφώνων, ρολογιών, γυαλιών, ταμπλετών, τηλεοράσεων, απαιτούν περισσότερες επιλογές, περισσότερες δυνατότητες προσαρμογής των συσκευών τους και περισσότερες λειτουργίες. Οι πάροχοι υπηρεσιών κινητών τηλεφώνων και ίντερνετ θέλουν να προσφέρουν στους συνδρομητές τους υλικό προστιθέμενης αξίας με εύχρηστο και επικερδή τρόπο. Οι προγραμματιστές έξυπνων συσκευών επιζητούν την ευχέρεια να αναπτύσσουν τις ισχυρές εφαρμογές που απαιτούν οι χρήστες, με τα λιγότερα δυνατά εμπόδια στο δρόμο για την επιτυχία. Τέλος, οι κατασκευαστές έξυπνων συσκευών ζητούν μια σταθερή, ασφαλή και οικονομική πλατφόρμα που θα τροφοδοτεί τις συσκευές τους.

Σ' αυτό το ευμετάβλητο τοπίο η αποστολή και λήψη παντός είδους πληροφορίας σε πραγματικό χρόνο είναι μία παγιωμένη επιδίωξη των «έξυπνων» χρηστών. Δεν είναι τυχαίο ότι, κατά την τελευταία δεκαετία σημειώνεται μια εκρηκτική αύξηση της χρήσης των σύγχρονων εφαρμογών άμεσης ανταλλαγής μηνυμάτων (Global Industry Analyst, 2016). Εν συμφωνία προς τις επιταγές των καιρών για την εξ αποστάσεως ακαριαία αμφίδρομη διανθρώπινη επικοινωνία, στην παρούσα πραγματεία παρατάσσονται λεπτομερώς τα δομικά συστατικά και τα εργαλεία για την οικοδόμηση μιας Εφαρμογής άμεσης ανταλλαγής μηνυμάτων μέσω της δωρεάν υπηρεσίας Google Cloud Messaging για έξυπνες φορητές συσκευές και τηλεοράσεις με λειτουργικό σύστημα Android.

Στο πρώτο μέρος του παρόντος περιγράφονται τα εκ των ων ουκ άνευ στοιχεία της διάρθρωσης ενός ολοκληρωμένου συστήματος αμφίδρομης επικοινωνίας σε πραγματικό χρόνο μέσω της υπηρεσίας Google Cloud Messaging. Στο επόμενο μέρος περιέχονται τμήματα κώδικα σε Java για την κλιμακωτή οικοδόμηση μιας Android Εφαρμογής Χρήστη με GCM. Στο τρίτο μέρος παρουσιάζεται η Android Εφαρμογή Χρήστη για την άμεση ανταλλαγή μηνυμάτων μέσω GCM που εκπονήθηκε στο πλαίσιο της παρούσας μεταπτυχιακής διατριβής. Ακολουθεί κατακλείδα με ερευνητικές προτάσεις αξιοποίησης της περί ης ο λόγος Εφαρμογής.

Αναλυτική Περιγραφή της υπηρεσίας Google Cloud Messaging (GCM)

Η υπηρεσία Google Cloud Messaging (GCM) παρέχει τη δυνατότητα της δωρεάν διαβίβασης των μηνυμάτων που αποστέλλονται από έναν Διακομιστή-Εξυπηρετητή μιας δεδομένης Εφαρμογής (Application Server) προς έναν αντίστοιχο-οικείο GCM Πελάτη-Εφαρμογή Χρήστη (Client Application) και vice-versa. Ντε φάκτο, κάθε Εφαρμογή Χρήστη-Πελάτης πρέπει να τρέχει στον φυλλομετρητή Google Chrome ή σε έξυπνες φορητές συσκευές με λειτουργικό σύστημα Android OS και IOS ή στις αντίστοιχες πλατφόρμες για έξυπνες τηλεοράσεις (Smart TV platforms) Android TV και tvOS.

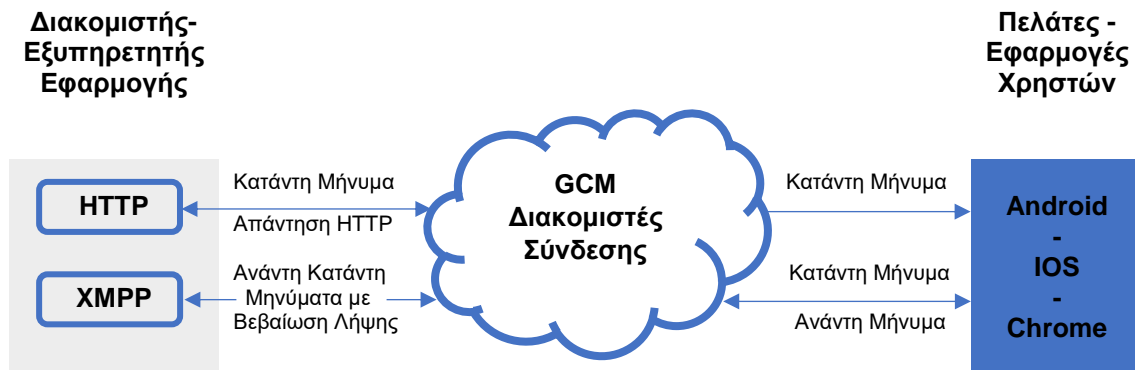
Η εν λόγω υπηρεσία επωμίζεται λοιπόν άνευ χρηματικού αντίτιμου τη μεταφορά τόσο των κατάντων (downstream) όσο και των ανάντων (upstream) GCM μηνυμάτων· προσφέρει δηλαδή έναν δίαυλο-κανάλι ανταλλαγής πληροφοριών, μέσω του οποίου μεταδίδονται δωρεάν, αφενός μεν τα μηνύματα που αποστέλλει ένας Διακομιστής-Εξυπηρετητής σε μια ή πολλές οικείες GCM Εφαρμογές-Πελάτες, αφετέρου δε τα μηνύματα που αποστέλλει ένας GCM Πελάτης-Εφαρμογή Χρήστη σ' έναν δεδομένο-οικείο Διακομιστή-Εξυπηρετητή. Κάθε τέτοιο μήνυμα δύναται να μεταφέρει μέχρι και 4 kb πληροφορίας. Κλασικά κατάντη GCM μηνύματα είναι οι ειδοποιήσεις (notifications). Αυτές καθιστούν ενήμερο έναν παραλήπτη-χρήστη για την ύπαρξη νέων δεδομένων, που θα πρέπει να τραβήξει η εφαρμογή του από τον αντίστοιχο Διακομιστή-Εξυπηρετητή, όπως είναι για παράδειγμα ένα νέο μήνυμα ηλεκτρονικού ταχυδρομείου ή ένα νέο αίτημα φιλίας σε κάποιο από τα μέσα κοινωνικής δικτύωσης. Είναι δε ντε φάκτο απαραίτητο, καθεμία GCM Εφαρμογή Χρήστη και εν δυνάμει Πελάτης να αποστείλει ένα άναπτες GCM μήνυμα ταυτοποίησης που να ενσωματώνει τα απαιτούμενα και μοναδικά τεκμήρια-πειστήρια γνησιότητας-αυθεντικοποίησης, τα οποία εκδίδονται από την Υπηρεσία Πιστοποίησης δια Νέφους Instance ID, προκειμένου να ολοκληρωθεί επιτυχώς η πρώτη χειραψία-σύσταση με έναν Διακομιστή Εφαρμογής και δυναμικό Εξυπηρετητή. Κλασικά ανάντη GCM μηνύματα είναι ακόμη τα μηνύματα που στέλνει μια Εφαρμογή-Πελάτης για να ενεργοποιήσει ή να απενεργοποιήσει την αποστολή ειδοποιήσεων από έναν οικείο Διακομιστή Εφαρμογής αλλά και τα εξερχόμενα μηνύματα με τελικούς παραλήπτες άλλες συζυγείς GCM Εφαρμογές Χρηστών. Η περί ης ο λόγος υπηρεσία της Google αναλαμβάνει λοιπόν τη διανομή τέτοιων μηνυμάτων και παράλληλα χειρίζεται τις αντίστοιχες ουρές αναμονής μηνυμάτων.

Η διάρθρωση μιας GCM Εφαρμογής

Ο Διακομιστής-Εξυπηρετητής Εφαρμογής, οι πιστοποιημένοι Πελάτες-Εφαρμογές Χρηστών και οι GCM Διακομιστές Σύνδεσης (GCM Connection Servers) αποτελούν τις εκ των ων ουκ άνευ συνιστώσες κάθε GCM Εφαρμογής. Οι GCM Διακομιστές Σύνδεσης δύνανται να μεταφέρουν τα ανάντη GCM μηνύματα βάσει του πρωτόκολλου XMPP (Extensible Messaging and Presence Protocol), τουτέστι το Επεκτάσιμο Πρωτόκολλο Μεταφοράς Στοιχείων XML (eXtensible Markup Language-Επεκτάσιμη Γλώσσα Σήμανσης) για την μετάδοση μηνυμάτων και πληροφοριών παρουσίας σε πραγματικό χρόνο (Saint-Andre (a) RFC: 6120, (b) RFC: 6121, 2011· RFC: 7622, 2015). Η διαβίβαση δε των κατάντων μηνυμάτων συντελείται ομοιοτρόπως (mutatis-mutandis) τηρώντας το πρωτόκολλο XMPP ή/και συμφώνως προς το Πρωτόκολλο Μεταφοράς Υπερκειμένου HTTP (Hyper Text Transfer Protocol) (Fielding & Reschke, RFC: 7230, 2014· Belshe, Peon & Thomson, RFC: 7540, 2015).

Το Σχήμα 1 αναπαριστά γραφικά την αλληλεπίδραση μεταξύ των τριών προαναφερόμενων στοιχειωδών οντοτήτων κάθε GCM Εφαρμογής, η οποία εξηγείται ακολούθως.

Σχήμα 1. Η διάρθρωση μιας GCM Εφαρμογής.



- Οι GCM Διακομιστές Σύνδεσης παρέχονται από την Google για να επωμίζονται τα κατάντη μηνύματα εκ του Διακομιστή-Εξυπηρετητή Εφαρμογής, τα οποία καταχωρούν σε ουρά αναμονής αποστολής (enqueue) και εν τέλει τα μεταβιβάζουν στην Εφαρμογή του Χρήστη-Πελάτη που ντε φάκτο εκτελείται στον διαδικτυακό περιηγητή Google Chrome ή σε έξυπνες φορητές συσκευές με λειτουργικό σύστημα Android OS και IOS ή στις αντίστοιχες πλατφόρμες για έξυπνες τηλεοράσεις Android TV και tvOS. Η Google διαθέτει Διακομιστές Σύνδεσης για τα πρωτόκολλα επικοινωνίας HTTP και XMPP. Σε μία υλοποίηση δια XMPP, εν αντιθέσει με την αντίστοιχη σε HTTP, δίνεται η δυνατότητα στον Διακομιστή της Εφαρμογής να λάβει ανάντη μηνύματα από την Εφαρμογή του Χρήστη. Ακόμη, δια του πρωτοκόλλου XMPP, η ανταλλαγή των μηνυμάτων από και προς το Διακομιστή της Εφαρμογής είναι ασύγχρονη στο μέγιστο ρυθμό αποστολής δεδομένων (at full line speed) επί αρραγών συνδέσεων XMPP. Στο δεδομένο σύστημα ασύγχρονης επικοινωνίας οι Διακομιστές Σύνδεσης απαντούν στο Διακομιστή Εφαρμογής συμφώνως προς το XMPP με αντίστοιχα κωδικοποιημένα μηνύματα αναγνώρισης λήψης ACK και διαβίβασης (acknowledgement) ή αποτυχίας NACK (negative-acknowledgement) διαβίβασης του αντίστοιχου κατάντους μηνύματος. Στον αντίποδα, ο Διακομιστής της Εφαρμογής σε μια υλοποίηση δια του πρωτοκόλλου HTTP αποστέλλει κατάντη μηνύματα ως αιτήματα POST HTTP και αναμένει απόκριση από τους GCM Διακομιστές Σύνδεσης, άρα ντε φάκτο το δεδομένο σύστημα ανταλλαγής μηνυμάτων είναι σύγχρονο, διότι αποκλείει την αποστολή νέων μηνυμάτων στον Διακομιστή της Εφαρμογής ωστόσο να λάβει απόκριση από τους GCM Διακομιστές Σύνδεσης. Αμφότερα τα δύο πρωτόκολλα επικοινωνίας δύνανται να διαβιβάσουν μηνύματα JSON (JavaScript Object Notation) (Bray, RFC: 7159, 2014· RFC: 7493, 2015)· μέσω του HTTP τα JSON μηνύματα αποστέλλονται ως αιτήματα HTTP POST, ενώ δια του XMPP τα μηνύματα τούτα ενσωματώνονται σε XMPP μηνύματα. Επίσης, σε μια υλοποίηση μέσω XMPP, δεν δύναται να αποσταλεί μήνυμα απλού-μη εμπλουτισμένου αναγνώσιμου κειμένου χαρακτήρων (plain text), η να εκπνευθεί ένα μήνυμα με πολλαπλούς παραλήπτες (multicast). Εν αντιθέσει, μέσω HTTP δύναται η αποστολή απλού αναγνώσιμου κειμένου χαρακτήρων ως κάταντες μήνυμα-αίτημα HTTP POST καθώς και η εκπομπή ενός μηνύματος με πολλαπλούς δέκτες με τη δομή μηνύματος JSON.
- Το πιστοποιητικό-κλειδί του Διακομιστή της Εφαρμογής (server key) παρέχει εγκεκριμένη πρόσβαση στον(ους) προγραμματιστή μιας συναφούς εφαρμογής στις χρειώδεις υπηρεσίες της Google, ενώ αποτελεί εκ των ων ουκ άνευ τεκμήριο για την οικοδόμηση του δίαυλου επικοινωνίας της οικείας αρχιτεκτονικής. Στον Διακομιστή της Εφαρμογής αποθηκεύεται εν κρυπτών και παραβύστω το εν λόγω πιστοποιητικό-κλειδί. Ο Διακομιστής της Εφαρμογής έχει δε την αρμοδιότητα της ορθής σύνταξης και αποστολής των αιτημάτων προς τους GCM Διακομιστές Σύνδεσης· ενώ βάσει του αλγόριθμου της Εκθετικής Υποχώρησης (Exponential Back-off) χειρίζεται την αύθις αποστολή εκείνων των αιτημάτων για τα οποία έλαβε ένα συναφές κωδικό κατάστασης διαβίβασης ως απάντηση από τον GCM Διακομιστή Σύνδεσης. Σύμφωνα με τους Tanenbaum και Wetherall (2011, σσ. 285-331), ο αλγόριθμος της Εκθετικής Υποχώρησης προλαμβάνει τη συμφόρηση ενός δικτύου δεδομένων αυξάνοντας εκθετικά το χρονικό διάστημα που

μεσολαβεί μεταξύ της λήψης ενός σχετικού κωδικού κατάστασης αιτήματος και της εκ νέου αποστολής των οικείων δεδομένων, έως μια δεδομένη χρονική οροφή.

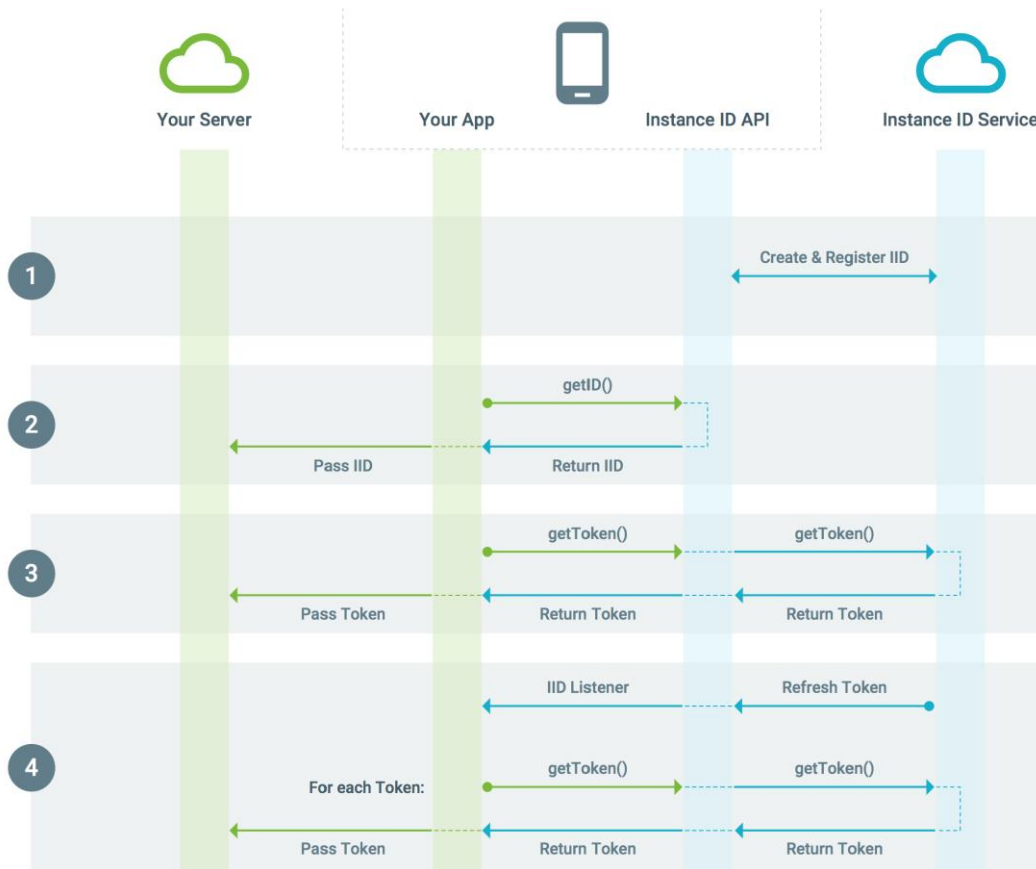
Για να θεμελιωθεί όμως ο δίαυλος επικοινωνίας μεταξύ του Διακομιστή Εφαρμογής και της GCM Εφαρμογής Χρήστη, καθίσταται επιβεβλημένη η επιτυχής περάτωση των εξής, πρώτα η Εφαρμογή Χρήστη θα πρέπει να εγγραφεί στην υπηρεσία νέφους (Cloud Service) Instance ID, κατόπιν να προβεί στην έκδοση του κατάλληλου πειστηρίου αυθεντικότητας (Instance ID tokens) και τέλος να μεταβιβάσει το δεδομένο τεκμήριο γνησιότητας στον Διακομιστή της Εφαρμογής, ώστε να ολοκληρωθεί η αντίστοιχη σύσταση. Το Γράφημα # απεικονίζει τη δεδομένη διαδικασία εγγραφής και την πρώτη χειραψία μεταξύ των δυο προαναφερόμενων συνιστωσών της εν λόγω αρχιτεκτονικής, η οποία επικυρώνει την επικοινωνία τους μέσω των παρεχόμενων GCM Διακομιστών Σύνδεσης. Επί της παρούσας πραγματείας περιέχεται δε στο κεφάλαιο «Η κλιμακωτή οικοδόμηση μιας GCM Εφαρμογής Χρήστη για έξυπνες φορητές συσκευές και τηλεοράσεις με λειτουργικό σύστημα Android OS» στην υπό-ενότητα που φέρει τον τίτλο «Η εγγραφή μιας GCM Android Εφαρμογής Χρήστη στην Υπηρεσία Νέφους Instance ID – Έκδοση, Ανανέωση και Διαγραφή των συζυγών Τεκμηρίων Γνησιότητας», ο κώδικας σε Java που αφορά στην περιγραφόμενη διαδικασία.

Αρχικά λοιπόν, ανεξάρτητα από το λειτουργικό σύστημα επί του οποίου εκτελείται η GCM Εφαρμογή, καθίσταται απαραίτητη η συμπερίληψη της συναφούς βιβλιοθήκης στον αντίστοιχο κώδικα. Κατόπιν, καθεμιά τέτοια εφαρμογή εγγράφεται διαδικτυακά στην υπηρεσία νέφους Instance ID και λαμβάνει τη συζυγή μοναδική ταυτότητα εφαρμογής (Instance ID). Το πιστοποιητικό αυτό, θωρακίζεται από ένα ζεύγος κλειδιών κρυπτογράφησης ενός δημόσιου και του αντίστοιχου ιδιωτικού. Το δεύτερο σώζεται τοπικά στη συσκευή του χρήστη ή εναλλακτικά εκχωρείται στον Διακομιστή της Εφαρμογής διαδικτυακά, ενώ το δημόσιο καταχωρείται στην οικεία υπηρεσία νέφους Instance ID. Η ταυτότητα αυτή επιτρέπει στην κάτοχο-εφαρμογή την απόκτηση των κατάλληλων τεκμηρίων γνησιότητας, τα οποία αφενός μεν διασφαλίζουν τη διαβίβαση των GCM μηνυμάτων μέσω των GCM Διακομιστών Σύνδεσης αφετέρου δε, κατοχυρώνουν την προσπέλαση της συζυγούς εφαρμογής στα δεδομένα που διαχειρίζεται ο συζυγής Διακομιστής Εφαρμογής.

Για την έκδοση όμως των πειστηρίων αυθεντικότητας μιας Εφαρμογής Χρήστη πέραν από την βεβαίωση εγγραφής της στην οικεία υπηρεσία νέφους Instance ID, καθίσταται επίσης απαραίτητη η κατάθεση του συζυγούς Κλειδάριθμου Λογισμικού Εφαρμογής (Project ID) που εκχωρείται στους διαχειριστές της από την Κονσόλα Προγραμματιστών της Google (Google Developers Console) ενώ, προαιρετική είναι η επισύναψη των αιτίων της έκδοσης των οικείων τεκμηρίων γνησιότητας ως συμβολοσειρά χαρακτήρων εύρους από κανέναν έως και χίλιους χαρακτήρες, οι οποίοι θα πρέπει να εμπίπτουν στην κατηγορία των «Ακίνδυνων Χαρακτήρων για Ενιαίους Εντοπιστές Πόρων (URL-Safe Characters)» βάσει των προδιαγραφών για Uniform Resource Locators (URL), RFC 1738 των Berners-Lee, Masinter και McCahill (1994).

Εν τέλει, η πρώτη αυτή χειραψία ολοκληρώνεται όταν η Εφαρμογή του Χρήστη μεταβιβάσει το αντίστοιχο τεκμήριο αυθεντικότητας στον Διακομιστή Εφαρμογής, ο οποίος αφού το αποθηκεύσει, ενημερώνει με κάταντες μήνυμα την αντίστοιχη Εφαρμογή ότι η οικεία διαδικασία ολοκληρώθηκε επιτυχώς. Σε περίπτωση αποτυχίας ολοκλήρωσης της περί ης ο λόγος συστάσεως, η μετέχουσα Εφαρμογή αποστέλλει αύθις το αντίστοιχο τεκμήριο γνησιότητας στον αντίστοιχο Διακομιστή Εφαρμογής βάσει του προαναφερόμενου αλγόριθμου της Εκθετικής Υποχώρησης.

Γράφημα 2. Η διαδικασία εγγραφής και η πρώτη χειραψία μεταξύ μιας Εφαρμογής Χρήστη και του Διακομιστή Εφαρμογής που επικυρώνει την επικοινωνία τους δια των παρεχόμενων GCM Διακομιστών Σύνδεσης



Ο Διακομιστής Εφαρμογής πρέπει όμως να σώζει το φρέσκο τεκμήριο γνησιότητας για καθεμιά GCM Εφαρμογή Χρήστη ώστε να αποφευχθεί η αποστολή απαντώντων πανομοιότυπων μηνυμάτων, στην περίπτωση που ανακύψει κάποιο κρούσμα εσφαλμένης αλληπάλληλης μεταβίβασης τεκμηρίων γνησιότητας από κάποια ελαττωματική εφαρμογή. Το πλέον πρόσφατο πειστήριο αυθεντικότητας επονομάζεται ως Κανονική Ταυτότητα (Canonical ID) μιας τέτοιας εφαρμογής. Ωστόσο, το δεδομένο πειστήριο γνησιότητας μιας γνώριμης εφαρμογής δεν αντιστοιχεί ντε φάκτο σε μια συγκεκριμένη συσκευή ενός χρήστη. Εάν μια τέτοια εφαρμογή διαγράψει το αντίστοιχο τεκμήριο γνησιότητας και προβεί αύθις στην έκδοση ενός φρέσκου, δεν είναι σίγουρο ότι θα λάβει ένα πανομοιότυπο τέτοιο πειστήριο. Εάν ληφθεί δε υπ' όψιν ότι, οι διεργασίες του σβησίματος, της επανέκδοσης και της μεταβίβασης ενός τεκμηρίου αυθεντικότητας από μια αντίστοιχη εφαρμογή ενδέχεται να διαρκέσουν αθροιστικά μέχρι και πέντε λεπτά, είναι πιθανό στο μεσοδιάστημα τα αντίστοιχα κατάνη μηνύματα να χαθούν ή να αποσταλούν σε λάθος παραλήπτη. Ως εκ τούτου, ο Διακομιστής Εφαρμογής πρέπει να φυλάσσει μια αμφιμονοσήμαντη σχέση μεταξύ των αναγνωρισμένων συσκευών και των συζυγών Κανονικών τους Ταυτοτήτων. Επίσης για κάθε εισερχόμενο μήνυμα, η αποδέκτης εφαρμογή πρέπει να ελέγχει την ορθή αντιστοιχία μεταξύ των αντίστοιχων τεκμηρίων γνησιότητας και των στοιχείων του παραλήπτη, πριν να προβεί σε οποιαδήποτε χρήση του.

Αν ένας χρήστης απεγκαταστήσει την περί ης ο λόγος εφαρμογή, τότε ντε φάκτο το συζυγές τεκμήριο γνησιότητας διαγράφεται από την οικεία υπηρεσία GCM. Ωστόσο, η κατάργηση του αντίστοιχου πειστηρίου αυθεντικότητας δεν περατώνεται μονοστιγμής. Αν λοιπόν στο μεσοδιάστημα ο Διακομιστής Εφαρμογής αποστείλει ένα κατάνη μήνυμα

στην απεγκαθισταμένη εφαρμογή, τότε ο GCM Διακομιστής Σύνδεσης, που φέρει την αρμοδιότητα της διαβίβασης τέτοιων μηνυμάτων, θα δύναται να αντιληφθεί ότι η συζυγής εφαρμογή έχει απεγκατασταθεί και αυτοστιγμεί θα προβεί στη διαγραφή του αντίστοιχου τεκμηρίου αυθεντικότητας από την εν λόγω υπηρεσία νέφους της Google. Μολοντούτο, ο οικείος Διακομιστής Εφαρμογής είναι πολύ πιθανό να λάβει μια θετική απόκριση στο συγκεκριμένο αίτημα διαβίβασης από τον GCM Διακομιστή Σύνδεσης, ασχέτως αν απέτυχε να παραδώσει το δεδομένο μήνυμα. Αφού περαιωθεί η διαγραφή του συζυγούς πειστηρίου γνησιότητας, στην περίπτωση που ο Διακομιστής της Εφαρμογής επιχειρήσει να διαβιβάσει ξανά ένα κάταντες μήνυμα προς την οικεία εφαρμογή, τότε ο GCM Διακομιστής Σύνδεσης θα απαντήσει με μήνυμα ανεπιτυχούς διαβίβασης ένεκα μη εγγεγραμμένου παραλήπτη (Not Registered error) στην αντίστοιχη υπηρεσία νέφους της Google. Μόλις ο Διακομιστής Εφαρμογής λάβει το συγκεκριμένο μήνυμα σφάλματος, θα πρέπει λοιπόν να διαγράψει το τεκμήριο γνησιότητας της απεγκαθιστάμενης εφαρμογής.

- Ανεξαρτήτως τηρουμένου πρωτόκολλου επικοινωνίας, τα βασικά δομικά μέρη κάθε κατόντος GCM μηνύματος είναι τα ακόλουθα, (α) το πεδίο "to" – ήτοι ο προορισμός του μηνύματος – που εμφανίζει ρητώς μια παραλήπτρια Εφαρμογή Χρήστη ή εμμέσως πολλές συναποδέκτριες συζυγείς GCM Εφαρμογές και στο οποίο καταχωρείται αντιστοίχως η Κανονική Ταυτότητα (Canonical ID) της μιας αποδέκτριας εφαρμογής, ο Κλειδάρηθος Ειδοποίησεως (Notification Key) που αντιστοιχεί σε ένα καθορισμένο υποσύνολο φρέσκων πειστηρίων αυθεντικότητας συζυγών Εφαρμογών Χρήστη (Device Group), ή το Θέμα (Topic) στο οποίο είναι εγγεγραμμένες συζυγείς Εφαρμογές Χρηστών-Συνδρομητών (Subscribers), (β) οι Επιλογές-Ρυθμίσεις Μηνύματος (Message Options), όπως είναι η υποσήμευση επιδεκτικότητας πτώσης και αντικατάστασης του μηνύματος τουτέστιν αν είναι ανατρέψιμο-αντικαταστάσιμο (Collapsible) ή τουναντίον ακατάρριπτο-αναντικατάστατο (Non-collapsible), η επιλογή απλής ή υψηλής προτεραιότητας (Normal ή High Priority) και ο καθορισμός της διάρκειας ύπαρξης (Lifespan) και τέλος το μήνυμα καθαυτό – ήτοι (γ) το Ωφέλιμο Φορτίο Μηνύματος (Message Payload) που δύναται να περιέχει μια Ειδοποίηση (Notification) ή/και Δεδομένα (Data).

Υπάρχουν λοιπόν δύο τύποι κατόντων GCM μηνυμάτων, οι Ειδοποιήσεις και τα Μηνύματα Δεδομένων (Data Messages). Ένα προκαθορισμένο αντικείμενο (Object) JSON – τουτέστιν ένα αδιάτακτο σύνολο από ζεύγη ονομάτων/τιμών, το οποίο φέρει το όνομα notification και έχει μορφοποιηθεί βάσει του προτύπου ανταλλαγής δεδομένων JSON – αποτελεί το στοιχειώδες συστατικό κάθε τέτοιας Ειδοποίησης. Η οικεία υπηρεσία GCM αναλαμβάνει λοιπόν να εμφανίσει την Ειδοποίηση στην οθόνη της συσκευής του παραλήπτη για λογαριασμό της συζυγούς Εφαρμογής Χρήστη-Πελάτη – επί τη υποθέσει της απουσίας της από το προσκήνιο – χρησιμοποιώντας τα ζευγάρια προκαθορισμένων ονομάτων/τιμών του JSON αντικειμένου notification. Για παράδειγμα, στο κάτωθι Σχήμα JSON (JSON Schema) μορφοποιείται μια τέτοια Ειδοποίηση για μια δεδομένη GCM Εφαρμογή Μετάδοσης Άμεσων Μηνυμάτων (Instant Messaging Application – IM App). Στην οθόνη του παραλήπτη αναμένεται να εμφανιστεί η ειδοποίηση με εικονίδιο «IMApp_icon», κείμενο τίτλου «2017 UEFA Champions League Final» και κείμενο σώματος «Real Madrid, the defending champions after defeating Atletico Madrid in the 2016 final, beat Juventus 4 – 1 to win a record-extending 12th title. With this victory, Real Madrid became the first team to successfully defend their title in the Champions League era, and the first to do so since Milan in 1990».

```
"to" : "bk3RNwTe3H0:CI2k_HHwgIpoDKCIZvvDMExUdFQ3P1...",
"notification" : {
  "body" : "Real Madrid, the defending champions after
defeating Atletico Madrid in the 2016 final, beat Juventus
4 - 1 to win a record-extending 12th title. With this
victory, Real Madrid became the first team to successfully
defend their title in the Champions League era, and the
first to do so since Milan in 1990",
  "title" : "2017 UEFA Champions League Final",
  "icon" : "IMApp_icon"
}
```

Τηρουμένων των αναλογιών, ένα ευάρμοστο αντικείμενο JSON – ήτοι ένα αδιάτακτο σύνολο από ζευγάρια προσαρμόσιμων ονομάτων/τιμών, το οποίο φέρει το όνομα data και συνιστά το Ωφέλιμο Φορτίο Μηνύματος – αποτελεί το ουσιώδες στοιχείο κάθε GCM Μηνύματος Δεδομένων. Για παράδειγμα, στο ακόλουθο Σχήμα JSON μορφοποιείται το συζυγές Μήνυμα Δεδομένων της άνωθεν Ειδοποίησης για τη δεδομένη GCM Εφαρμογή Ανταλλαγής Μηνυμάτων σε πραγματικό χρόνο, όπου η οικεία πληροφορία ενσωματώνεται πλέον στο JSON αντικείμενο data. Εξ ορισμού, η GCM Εφαρμογή Χρήστη ενωτίζεται τέτοια εισερχόμενα μηνύματα και κατόπιν εξορύσσει την ατόφια περιεχόμενη πληροφορία αναλύοντας το αντίστοιχο JSON αντικείμενο data συμφώνως προς ένα προκαθορισμένο Σχήμα JSON (JSON – Parsing). Κάθε τέτοιο Μήνυμα Δεδομένων δύναται να μεταφέρει μέχρι και 4 kb πληροφορίας.

```
"to" : "bk3RNwTe3H0:CI2k_HHwgIpoDKCIZvvDMExUdFQ3P1...",
  "data" : {
    "nick" : "Ronaldo_7"
    "body" : " Real Madrid, the defending champions after
defeating Atletico Madrid in the 2016 final, beat Juventus
4 - 1 to win a record-extending 12th title. With this
victory, Real Madrid became the first team to successfully
defend their title in the Champions League era, and the
first to do so since Milan in 1990",
    "subject" : " Juventus VS Real Madrid",
  }
```

Ωστόσο, το Ωφέλιμο Φορτίο μιας Ειδοποίησης μπορεί να περιέχει αμφότερα τα JSON αντικείμενα notification και data. Επί τη υποθέσει ότι, η παραλήπτρια GCM Εφαρμογή Χρήστη-Πελάτη δεν βρίσκεται στο προσκήνιο της οθόνης της συσκευής του αποδέκτη (When in the background), άμα τη αφίξει ενός τέτοιου εισερχόμενου μηνύματος με υβριδικό Ωφέλιμο Φορτίο πυροδοτείται μονοστιγμής η εμφάνιση της συντετημένης βερσιόν της αφιχθείσας ειδοποίησης στη Γραμμή Κατάστασης (Status Bar) της αντίστοιχης πλατφόρμας και της πλήρους ειδοποίησης, εάν ο χρήστης εκδιπλώσει την Γραμμή Κατάστασης, δια της χρήσης των ζευγών προκαθορισμένων ονομάτων/τιμών του γνωστού JSON αντικειμένου notification. Εφόσον ο παραλήπτης κλικάρει-επιλέξει την πλήρη ειδοποίηση, η GCM Εφαρμογή καλείται να εξορύξει και κατόπιν να χρησιμοποιήσει καταλλήλως την πληροφορία που κουβαλάει το οικείο JSON αντικείμενο data. Αν υποθεθεί ότι, η αποδέκτρια GCM Εφαρμογή Χρήστη-Πελάτη βρίσκεται στο προσκήνιο της οθόνης της συσκευής του παραλήπτη (When in the foreground), εν τω άμα τη ελεύσει ενός τέτοιου υβριδικού μηνύματος αμφότερα τα JSON αντικείμενα notification και data εκχωρούνται σ' αυτή μέσω ενός αντικειμένου της κλάσης Bundle. Με σκέτο ή μεικτό Ωφέλιμο Φορτίο κάθε GCM Ειδοποίηση μπορεί να κουβαλήσει μέχρι και 2 kb πληροφορίας. Ακολουθεί το Σχήμα JSON με υβριδικό Ωφέλιμο Φορτίο που μορφοποιεί τη συνάρθρωση των άνωθεν μηνυμάτων GCM για την οικεία GCM Εφαρμογή Ανταλλαγής Άμεσων Μηνυμάτων.

```
"to" : "bk3RNwTe3H0:CI2k_HHwgIpoDKCIZvvDMExUdFQ3P1...",
  "notification" : {
    "body" : "Real Madrid, the defending champions after
defeating Atletico Madrid in the 2016 final, beat Juventus
4 - 1 to win a record-extending 12th title. With this
victory, Real Madrid became the first team to successfully
defend their title in the Champions League era, and the
first to do so since Milan in 1990",
    "title" : "2017 UEFA Champions League Final",
    "icon" : "IMApp_icon"
  }
  "data" : {
    "nick" : "Ronaldo_7"
    "subject" : " Juventus VS Real Madrid"
  }
```

Κάθε GCM Ειδοποίηση είναι ντε φάκτο ανατρέψιμη και αντικαταστάσιμη (Collapsible), ενώ ένα συζυγές Μήνυμα Δεδομένων απριόρι δεν επιδέχεται ριζίματος κι αντικατάσταση (Non-collapsible), ωστόσο δύναται να είναι ανατρέψιμο και αντικαταστάσιμο εφόσον περιέχει μια υποσήμανση επιδεκτικότητας πτώσης και αντικατάστασης, τουτέστιν ένα JSON ζεύγος κλειδάριθμου πτώσης/δεδομένης αλφαριθμητικής τιμής (collapse key/string value). Η αποστολή ενός φρέσκου τέτοιου Μηνύματος Δεδομένων με συγκεκριμένο κλειδάριθμο κατάρρευσης και προορισμό επιφέρει την άμεση πτώση και αντικατάσταση του συζυγούς προγενέστερου αιωρούμενου μηνύματος – τουτέστιν του προτερόχρονου μηνύματος που είναι σε εκκρεμότητα παράδοσης. Οι επισημάνσεις επανεπικαιροποίησης δεδομένων (Send to Sync Messages) – ήτοι τα μηνύματα εν αναφορά προς το συγχρονισμό των φρέσκων δεδομένων του Εξυπηρετητή Εφαρμογής με εκείνα των συζυγών Πελατών-Εφαρμογών Χρήστη, είναι κλασικά ανατρέψιμα κι αντικαταστάσιμα GCM Μηνύματα Δεδομένων και χρησιμοποιούνται ως επί το πλείστον σ' εφαρμογές άμεσης μετάδοσης αποτελεσμάτων αθλητικών αγώνων. Ανά πάσα στιγμή, οι GCM Διακομιστές Συνδέσεως δύναται να σώζουν το πολύ τέσσερα ανατρέψιμα κι αντικαταστάσιμα Μηνύματα Δεδομένων προς κοινό προορισμό αλλά με διαφορετικούς κλειδάριθμους πτώσης. Εν τη περιπτώσει της υπερβάσεως του καθορισμένου πλήθους τέτοιων εκκρεμούντων μηνυμάτων καθίσταται αδύνατο να προβλεφθεί ποιο θα ριχθεί εντέλει. Ως εκ τούτου είναι επιβεβλημένο, ο Εξυπηρετητής μιας GCM Εφαρμογής να στέλνει ταυτοχρόνως το πολύ τέσσερα τέτοια μηνύματα με διαφορετικούς κλειδάριθμους κατάρρευσης σε κάθε πιστοποιημένη συζυγή Εφαρμογή Χρήστη. Κλασικά ακατάρριπτα κι αναντικατάστατα GCM Μηνύματα Δεδομένων είναι τα άμεσα μηνύματα (Instant Messages) που ανταλλάσσουν οι χρήστες μιας GCM Εφαρμογής Μετάδοσης Μηνυμάτων σε πραγματικό χρόνο. Ωστόσο οι GCM Διακομιστές Σύνδεσης μπορούν να βαστούν ανά πάσα στιγμή το πολύ εκατό τέτοια μηνύματα προς μια δεδομένη πιστοποιημένη συζυγή παραλήπτρια Εφαρμογή Χρήστη. Άμα τη περιπτώσει του ξεπεράσματος του καθορισμένου αριθμού τέτοιων αιωρούμενων μηνυμάτων οι GCM Διακομιστές Σύνδεσης τ' απορρίπτουν όλα κι αυτοστιγμεί στέλνουν ένα ειδικό ενημερωτικό μήνυμα στην αντίστοιχη Εφαρμογή Χρήστη εν αναφορά προς το δεδομένο περιστατικό, προκειμένου να στείλει μια ανάντη αίτηση επαναποστολής των ριγμένων μηνυμάτων στον συζυγή Εξυπηρετητή της Εφαρμογής GCM.

Κάθε GCM Μήνυμα Δεδομένων έχει απριόρι απλή προτεραιότητα παράδοσης (Delivery of Normal Priority) εν αντιθέσει με τις Ειδοποιήσεις που έχουν υψηλή προτεραιότητα (High Priority). Οι GCM Διακομιστές Συνδέσεως επιχειρούν να διαβιβάσουν εν ακαρεί τα κάτανα μηνύματα που κουβαλούν την υποσήμανση υψηλής προτεραιότητας παράδοσης – ήτοι το JSON ζεύγος "priority" : " high".

Ντε φάκτο η συμπεριφορά μιας GCM Εφαρμογής Χρήστη καθιστά επιβεβλημένη τη ρητή εκχώρηση του δικαιώματος πρόσβασης στο διαδίκτυο. Επί τη υποθέσει ότι η συμπεριφορά μιας δεδομένης τέτοιας εφαρμογής καθιστά δε απαραίτητη τη ρητή εκχώρηση του δικαιώματος αποτροπής της μετάβασης του επεξεργαστή της αντίστοιχης Android ή IOS συσκευής σε κατάσταση ύπνου, άμα τη αφίξει ενός GCM μηνύματος υψηλής προτεραιότητας, εν πάση καταστάσει της συσκευής (εν δράσει ή εν υπνώσει) δημιουργείται διαδικτυακή ζεύξη με τον αντίστοιχο Διακομιστή Εφαρμογής· εν αντιθέσει, άμα τη ελεύσει ενός GCM μηνύματος απλής προτεραιότητας, η παραλήπτρια Εφαρμογή Χρήστη δεν δημιουργεί ουδεμία ιντερνετική ζεύξη, εφόσον ο επεξεργαστής της αντίστοιχης συσκευής βρίσκεται εν υπνώσει. Ως εκ τούτων, τα μηνύματα με υψηλή προτεραιότητα παράδοσης είναι ντε φάκτο μπαταριοβόρα, μολοντούτο ενδείκνυνται σε εφαρμογές μετάδοσης άμεσων μηνυμάτων· εν αντιθέσει με τα ελάσσονα ενεργοβόρα μηνύματα απλής προτεραιότητας όπως είναι οι επισημάνσεις επανεπικαιροποίησης δεδομένων σε ειδησεογραφικές και αρθρογραφικές εφαρμογές. Στην αρχή της επόμενης σελίδας βρίσκεται το Σχήμα JSON που μορφοποιεί μια GCM Ειδοποίηση με υβριδικό Ωφέλιμο Φορτίο και απλή προτεραιότητα παράδοσης που αποσκοπεί να καταστήσει ενήμερο έναν συνδρομητή του ανεξάρτητου διεθνούς εβδομαδιαίου περιοδικού γενικής ιατρικής – The Lancet – όσον αφορά το νέο τεύχος του περιοδικού. Η δεδομένη επισημάνση επανεπικαιροποίησης δεδομένων περιέχει τον αντίστοιχο διαδικτυακό σύνδεσμό, ούτως ώστε ο παραλήπτης να κατεβάσει τη νέα έκδοση του περιοδικού.

```

"to" : "bk3RNwTe3H0:CI2k_HHwgIpoDKCIZvvDMExUdFQ3P1...",
"priority" : "normal",
"notification" : {
  "body" : "This week's edition is now available",
  "title" : "The Lancet",
  "icon" : "Notification_handbell_icon"
}
"data" : {
  "@type": "Periodical",
  "issn": "0140-6736",
  "hasPart": {
    "@id": "vol1389",
    "@type": "PublicationVolume",
    "volumeNumber": "389"
  },
  "hasPart": {
    "@id": "issue10087",
    "@type": "PublicationIssue",
    "datePublished": "17 June 2017",
    "pageEnd": "2349",
    "pageStart": "2442",
    "issueNumber": "10087"
  },
  "name": "The Lancet",
  "publisher": "Elsevier"
  "contents":
  "http://www.thelancet.com/journals/lancet/issue/vol1389no10087/PIIS0140-6736(17)X0026-3"
}

```

Παρά ταύτα, οι GCM Διακομιστές Συνδέσεως – εν τω άμα τη ελεύσει ενός καάντους GCM μηνύματος – επιδιώκουν την αυτοστιγμεί παράδοσή του στην μία ή τις πολλές παραλήπτριες Εφαρμογές Χρήστη ασχέτως προς τον τύπο (Ειδοποίηση ή Μήνυμα Δεδομένων) και προς τη δυνητική ύπαρξη των υποσημάνσεων της επιδεκτικότητας πτώσης κι αντικατάστασης (Collapsible ή Non-collapsible) ή της προτεραιότητας παραδόσεως (Normal ή High Priority). Μολοντούτο, η παράδοση ενός τέτοιου μηνύματος δεν καθίσταται δυνατόν να επιτευχθεί πάντοτε αστραπηδόν, επειδή μια παραλήπτρια Εφαρμογή Χρήστη ενδέχεται να μην είναι διαθέσιμη τη στιγμή της παράδοσης· όταν επί παραδείγματι, μια κάτοχος συσκευή Android ή IOS είναι απενεργοποιημένη ή βρίσκεται εκτός διαδικτυακής σύνδεσης. Εν τοιαύτη περιπτώσει, ο εμπλεκόμενος GCM Διακομιστής Συνδέσεως σώζει το απaráδοτο μήνυμα, προκειμένου να το διαβιβάσει εν ευθέτω χρόνω. Ωστόσο, ένα εκκρεμές μήνυμα μπορεί να έχει μια προθεσμία παράδοσης – τουτέστιν ένα καθορισμένο χρονικό όριο πέρα από το οποίο η παράδοση ενός GCM μηνύματος, που έχει περιέλθει σε εκκρεμότητα διαβιβάσεως, είναι ανωφελής ή άχρηστη για τον παραλήπτη – φέρ' ειπείν η παράδοση μιας Ειδοποίησης-πρόσκλησης σε μία κοινωνική, αθλητική ή καλλιτεχνική εκδήλωση κατόπιν εορτής. Τη διάρκεια της ύπαρξης ενός αιωρούμενου GCM μηνύματος (Lifespan of a Message) – δηλαδή το καθορισμένο χρονικό διάστημα δευτερολέπτων μέσα στο οποίο στέκει η διαβίβαση ενός GCM μηνύματος που είναι εν εκκρεμότητι παραδόσεως – εμφανίει ρητώς η τιμή του JSON πεδίου "time_to_live", εφόσον αυτό ενυπάρχει στο αντίστοιχο Σχήμα JSON. Το χρονικό περιθώριο παράδοσης ορίζεται στο σύνολο {0,1, 2, ..., 2.419.200} – ήτοι όχι παραπάνω από τέσσερεις εβδομάδες εν εκκρεμότητι· απριόρι δε – όπερ σημαίνει ένα αιωρούμενο GCM μήνυμα χωρίς το JSON ζεύγος "time_to_live"/καθορισμένη διάρκεια ύπαρξης σε δευτερόλεπτα – η διορία μεταβιβάσεως ορίζεται στο μάξιμουμ. Επί τη υποθέσει ότι, η προθεσμία παράδοσης ενός μηνύματος GCM έχει οριστεί στο μίνιμουμ, τότε ντε φάκτο αυτό διαβιβάζεται εν τω άμα τη εμφανίσει στον συζυγή GCM Διακομιστή Σύνδεσης, ειδάλλως απορρίπτεται. Στην επόμενη σελίδα βρίσκεται το Σχήμα JSON που μορφοποιεί μια GCM Ειδοποίηση με υβριδικό Ωφέλιμο Φορτίο και προθεσμία παράδοσης δύο

εβδομάδων – ήτοι "time_to_live":1.209.600 – που απσκοπεύει στην ενημέρωση του παραλήπτη αναφορικά με τα πακέτα προσφορών για τη Σαιξπηρική τραγωδία ο "Ιούλιος Καίσαρ" που αναμένεται ν' ανέβει στη σκηνή του ανακατασκευασμένο θέατρο Γκλομπ (Shakespeare's Globe) στο Λονδίνο την πρώτη Οκτωβρίου του 2017. Επί τη υποθέσει ότι, οι προσφορές για το συγκεκριμένο θεατρικό έργο θα παυτούν δύο εβδομάδες μετά την πρώτη παράσταση, η παράδοση της σχετικής Ειδοποίησης μετά τις 15 Οκτωβρίου δεν έχει ουδεμία ωφελιμότητα για τον παραλήπτη.

```
"to" : "bk3RNwTe3H0:CI2k_HHwgIpoDKCIZvvDMExUdFQ3P1...",
"time_to_live" : 1.209.600
"notification" : {
  "body" : "Book 10 seated tickets and get 1 additional ticket
  free. Tickets must be for the same performance. Please note
  there is limited availability for groups in all areas of the
  theatre. For groups including students aged under 18 a ratio
  of 1 adult per 10 students is required. Adults must remain
  with their group throughout the performance",
  "title" : "Julius Caesar at Shakespeare's Globe",
  "icon" : "Notification_handbell_icon"
}
"data" : {
  "location": {
    "@type": "PerformingArtsTheater",
    "name": "Shakespeare's Globe",
    "sameAs": "http://www.shakespearesglobe.com/",
    "address": "London, UK"
  },
  "offers": {
    "@type": "Offer",
    "url": "http://www.shakespearesglobe.com/your-
    visit/box-office/globe-theatre-seating-plan-ticket-
    prices"
  },
  "startDate": "2017-10-01T19:30",
  "endDate": "2017-10-20T19:30",
  "workPerformed": {
    "@type": "CreativeWork",
    "name": "Julius Caesar",
    "sameAs":
    "http://en.wikipedia.org/wiki/Julius_Caesar_(play)",
    "sameAs":
    "http://worldcat.org/entity/work/id/1807288036",
    "creator": {
      "@type": "Person",
      "name": "William Shakespeare",
      "sameAs":
      "http://en.wikipedia.org/wiki/William_Shakespeare"
    }
  }
}
```

Η άφιξη λοιπόν ενός GCM μηνύματος σ' έναν GCM Διακομιστή Συνδέσεως δεν συνεπάγεται εξάπαντος κι ανυπερθέτως την παράδοση του στην πιστοποιημένη GCM παραλήπτρια Εφαρμογή Χρήστη. Εις την καλύτερη των περιπτώσεων, εν τω άμα τη ελεύσει ενός τέτοιου εισερχόμενου μηνύματος, η αποδέκτρια GCM Εφαρμογή Χρήστη έχει πρόσβαση στο διαδίκτυο και συνάμα ο επεξεργαστής της αντίστοιχης συσκευής δεν συναντάται εν υπνώσει, άρα το μήνυμα παραδίδεται μεμιάς. Αν υποτεθεί ότι, η συσκευή – όπου είναι εγκατεστημένη η δυναμική αποδέκτρια GCM Εφαρμογή Χρήστη με πιστοποιητικό αυθεντικότητας – είναι συνδεδεμένη στο διαδίκτυο αλλά ο επεξεργαστής

της βρίσκεται εν υπνώσει, ο ερχομός ενός κατάντους GCM μηνύματος με απλή προτεραιότητα παραδόσεως για τη συγκεκριμένη δύναμη παραλήπτρια Εφαρμογή σ' έναν GCM Διακομιστή Σύνδεσης και η εν τω άμα αποτυχημένη παράδοσή του καθιστά εκκρεμή τη διαβίβασή του, μέχρις ότου ξυπνήσει ο επεξεργαστής. Εν τοιαύτη περιπτώσει, εάν στον GCM Διακομιστή Σύνδεσης είναι ήδη σωσμένο ένα εκκρεμές GCM μήνυμα με κοινό προορισμό και ίδια υποσήμανση επιδεκτικότητας πτώσης κι αντικατάστασης, όπερ σημαίνει κοινό JSON ζευγάρι κλειδάριθμου κατάρρευσης/δεδομένης αλφαριθμητικής τιμής (collapse key/string value), τότε το φρέσκο μήνυμα προξενεί την άμεση πτώση και την αντικατάσταση του προτερόχρονου αιωρούμενου μηνύματος: στην περίπτωση που το ήδη σωσμένο εκκρεμές μήνυμα περιλαμβάνει διαφορετικό κλειδάριθμο πτώσης απ' αυτόν του επιγενέστερου αιωρούμενου μηνύματος, τότε αμφότερα διατηρούνται στον GCM Διακομιστή Σύνδεσης έως ότου εκπνεύσει η προθεσμία παράδοσής τους (JSON ζεύγος "time_to_live"/καθορισμένη διάρκεια ύπαρξης σε δευτερόλεπτα) ή ν' αφυπνιστεί ο επεξεργαστής της συσκευής, όπου είναι εγκατεστημένη η δυναμική παραλήπτρια Εφαρμογή Χρήστη κι αμφότερα να διαβιβαστούν σε χρόνο ντε τε. Εν τη περιπτώσει που μια πιστοποιημένη GCM Εφαρμογή Χρήστη δεν έχει πρόσβαση στο διαδίκτυο, η απιόρι ανεπιτυχής διαβίβαση ενός κατάντους GCM μηνύματος προς αυτή τη μη διαδικτυωμένη αλλά εν δυνάμει παραλήπτρια Εφαρμογή πυροδοτεί ντε φάκτο την προαναφερόμενη διαδικασία σωσίματος του μηνύματος εν εκκρεμότητι στον αντίστοιχο GCM Διακομιστή Συνδέσεως. Όταν η συσκευή, όπου είναι εγκατεστημένη η δυναμική παραλήπτρια GCM Εφαρμογή, ξανασυνδεθεί στο ίντερνετ, επιχειρείται αύθις η διαβίβαση όλων των σωσμένων μηνυμάτων προς αυτή. Αν τυχόν, η δημιουργία ιντερνετικής ζεύξης μεταξύ του GCM Διακομιστή Σύνδεσης και μιας δεδομένης Εφαρμογής Χρήστη καταστεί διά παντός αδύνατη, ένεκα επαναφοράς της συσκευής – όπου μέχρι πρότινος ήταν εγκατεστημένη η τέως αποδέκτρια GCM Εφαρμογή Χρήστη – στις εργοστασιακές προεπιλεγμένες ρυθμίσεις, τότε κάθε αποθηκευμένο εκκρεμές μήνυμα με προορισμό αυτή την Εφαρμογή εν τέλει αποβάλλεται απ' τον GCM Διακομιστή Σύνδεσης, εφόσον εκπνεύσει η διορία παράδοσής του. Εν τη περιπτώσει της απεγκαταστάσεως μιας πρώην – ούτως εχόντων των πραγμάτων – αποδέκτριας GCM Εφαρμογής Χρήστη από την πλέον πρώην κάτοχο συσκευή, άμα τη αφίξει ενός κατάντους GCM μηνύματος με παραλήπτρια τη νυν απεγκαταστημένη Εφαρμογή Χρήστη σ' έναν GCM Διακομιστή Συνδέσεως, καθίσταται μονοστιγμής άκυρο το πιστοποιητικό εγγραφής της στην υπηρεσία νέφους Instance ID, ενώ η διαβίβαση του άρτι αφιχθέντος μηνύματος απορρίπτεται εν ακαρεί· από τούδε και στο εξής κάθε απόπειρα αποστολής ενός GCM μηνύματος προς την απιστοποιητή πλέον Εφαρμογή Χρήστη απ' τον Διακομιστή-Εξυπηρετητή της GCM Εφαρμογής επιφέρει αυθωρεί ένα απαντητικό μήνυμα αποτυχημένης διαβίβασης – λόγω μη εγγεγραμμένου παραλήπτη (Not Registered error) στην οικεία υπηρεσία νέφους της Google – απ' τον GCM Διακομιστή Σύνδεσης. Επί τη υποθέσει ότι μια πιστοποιημένη δυναμική αποδέκτρια GCM Εφαρμογή Χρήστη δεν έχει δημιουργήσει ουδεμία διαδικτυακή ζεύξη μ' έναν GCM Διακομιστή Συνδέσεως για πάνω από τέσσερις εβδομάδες, εν τω άμα τη αφίξει ενός νεοεισελθόντος GCM μηνύματος προς αυτή – την Εφαρμογή που είναι άφαντη επί μακρόν – σ' έναν GCM Διακομιστή Σύνδεσης απορρίπτεται μονοστιγμής. Στην περίπτωση που μια πιστοποιημένη κι εν δυνάμει αποδέκτρια GCM Εφαρμογή Χρήστη – έως πρότινος αποσυνδεδεμένη – αποκτήσει εκ νέου πρόσβαση σ' έναν GCM Διακομιστή Συνδέσεως, το μάξιμουμ τέσσερις εβδομάδες αφότου εστάλη το έσχατο κατάντες GCM μήνυμα με παραλήπτρια τη συγκεκριμένη Εφαρμογή Χρήστη, πυροδοτείται η μέθοδος onDeletedMessages() – για συσκευή που τρέχει στο λειτουργικό σύστημα Android OS – η οποία τίθεται σε λειτουργία, εφόσον σ' έναν εμπλεκόμενο GCM Διακομιστή Συνδέσεως ενεργοποιείται η απόρριψη των μηνυμάτων που έχουν περιέλθει σε εκκρεμότητα παραδόσεως, προκειμένου να σταλεί μια ανάντη αίτηση επαναποστολής των ριγμένων μηνυμάτων στον συζυγή Εξυπηρετητή της Εφαρμογής GCM (Full Sync Request).

Όπως προαναφέρθη, το JSON πεδίο "to" – όπερ δηλώνει τον προορισμό ενός GCM μηνύματος – εμφανίζει ξεκάθαρα τη μια δυναμική παραλήπτρια GCM Εφαρμογή Χρήστη ή εμμέσως τις πολλές εν δυνάμει συναποδέκτριες συζυγείς αντίστοιχες Εφαρμογές. Κατ' ακολουθίαν στο πεδίο τούτο καταχωρείται, το φρέσκο πειστήριο γνησιότητας μιας αποδέκτριας GCM Εφαρμογής Χρήστη, ο Κλειδάριθμος Ειδοποίησης (Notification Key) που αναφέρεται σε ένα καθορισμένο υποσύνολο Κανονικών Ταυτοτήτων (Canonical IDs)

συζυγών Εφαρμογών Χρήστη (Device Group), ή το Θέμα (Topic) στο οποίο έχουν εγγραφεί συζυγείς Εφαρμογές Χρηστών-Συνδρομητών (Subscribers).

Ο Κλειδάριθμος Ειδοποίησης αντιστοιχεί λοιπόν σε μια συγκεκριμένη ομάδα-οικογένεια GCM Εφαρμογών Χρήστη, όπερ αφορά σ' ένα σαφώς καθορισμένο υποσύνολο αντίστοιχων συζυγών φρέσκων τεκμηρίων γνησιότητας που μπορεί να περιέχει το μάξιμουμ είκοσι συζυγείς Κανονικές Ταυτότητες μιας συγκεκριμένης GCM Εφαρμογής, η οποία έχει εγκατασταθεί από έναν και μόνο έναν Χρήστη σ' αντίστοιχο αριθμό συσκευών ανεξαρτήτως πλατφόρμας. Η υπηρεσία GCM φέρει την αρμοδιότητα της δημιουργίας-έκδοσης και της αυτοστιγμεί απόδοσης, όπερ σημαίνει εν τω άμα τη αίτησει, ενός μοναδικού Κλειδάριθμου Ειδοποίησης κατόπιν αντίστοιχου αιτήματος POST στον διαδικτυακό σύνδεσμο <https://android.googleapis.com/gcm/notification>. Για παράδειγμα, στο ακόλουθο σχήμα JSON μορφοποιείται το αίτημα για την έκδοση ενός Κλειδάριθμου Ειδοποίησης ονόματι Ronaldo_7 που αφορά στο υποσύνολο ["4", "8", "15", "16", "23", "42"] των συζυγών Κανονικών Ταυτοτήτων του Διακομιστή GCM Εφαρμογής με Ταυτότητα Αποστολέα SENDER_ID – δηλαδή το μοναδικό ακέραιο αριθμό που πιστοποιεί τη δυνητικότητα σύζευξης ενός Διακομιστή με τη δεδομένη GCM Εφαρμογή – και Κλειδάριθμο Διακομιστή SERVER_KEY – τουτέστιν το άνωθεν αναφερόμενο Πιστοποιητικό-Κλειδί Πρόσβασης του Διακομιστή της Εφαρμογής στις απαιτούμενες υπηρεσίες της Google. Καθίσταται δε ντε φάκτο επιβεβλημένη η μοναδικότητα του επιλεγμένου ονόματος ενός Κλειδάριθμου Ειδοποίησης ανά Εφαρμογή GCM για την τη δημιουργία-έκδοση του Κλειδαρίθμου καθαυτού.

```
https://android.googleapis.com/gcm/notification
Content-Type:application/json
Authorization:key=SERVER_KEY
project_id:SENDER_ID

{
  "operation": "create",
  "notification_key_name": "Ronaldo_7",
  "registration_ids": ["4", "8", "15", "16", "23", "42"]
}
```

Το JSON πεδίο "operation" εμφανίζει το καθαυτό αίτημα. Σ' αυτό καταχωρείται μια συμβολοσειρά εκ των create, add και remove. Τηρουμένου του προβλεπόμενου ανωτάτου ορίου των είκοσι Κανονικών Ταυτοτήτων ανά ομάδα-οικογένεια συζυγών τεκμηρίων γνησιότητας μιας δεδομένης GCM Εφαρμογής – ένα αίτημα για την προσθήκη μιας ή πολλών Κανονικών Ταυτοτήτων σε ένα ήδη καθορισμένο τέτοιο υποσύνολο πιστοποιητικών αυθεντικότητας, για το οποίο έχει εκδοθεί και ο αντίστοιχος Κλειδάριθμος Ειδοποίησης, περιλαμβάνει το JSON ζευγάρι "operation":"add". Φερειπείν στο κάτωθι JSON σχήμα μορφοποιείται το αίτημα για την πρόσθεση της Κανονικής Ταυτότητας "51" στο ήδη καθορισμένο υποσύνολο αντίστοιχων συζυγών φρέσκων τεκμηρίων γνησιότητας με Κλειδάριθμο Ειδοποίησης "APA91bGHXQBB...9QgnYOEURwm0k2TXQ" και όνομα "Ronaldo_7". Συμφώνως προς το οικείο προβλεπόμενο JSON Σχήμα το ζευγάρι "operation": "remove" περιέχεται σε μια αίτηση για τη διαγραφή μιας ή πολλών Κανονικών Ταυτοτήτων από μια κείμενη-υπάρχουσα οικογένεια συζυγών τεκμηρίων αυθεντικότητας μιας δεδομένης GCM Εφαρμογής· αντιστοίχως στο άνωθεν παράδειγμα της αίτησης εκδόσεως Κλειδάριθμου Ειδοποίησης στο JSON πεδίο "operation" καταχωρείται η συμβολοσειρά create.

```
{
  "operation": "add",
  "notification_key_name": "Ronaldo_7",
  "notification_key": "APA91bGHXQBB...9QgnYOEURwm0k2TXQ",
  "registration_ids": ["51"]
}
```

Η επιτυχημένη υποβολή ενός τέτοιου ή των ομόλογων αιτήματων συνεπάγεται λοιπόν την έκδοση ενός φρέσκου Κλειδαριθμού Ειδοποιήσεως ή την ανανέωση ενός ξεπερασμένου υπάρχοντος εν ισχύη και την εν ακαρεί απόδοσή του δια του JSON ζευγαριού "notification_key"/συμβολοσειράς του αντίστοιχου Κλειδαριθμού Ειδοποιήσεως, όπως αποδίδεται φέρ' ειπείν ακολούθως.

```
{
  "notification_key": "APA91bGHXQBB...9QgnYOEURwm0k2TXQ"
}
```

Με σκέτο ή μεικτό Ωφέλιμο Φορτίο ένα ομαδικό GCM μήνυμα δύναται να μεταφέρει μέχρι 2 kb πληροφορίας σε συναποδέκτριες GCM Εφαρμογές Χρήστη που τρέχουν σε συσκευές με λειτουργικό σύστημα IOS και 4 kb δεδομένων σε συζυγείς Εφαρμογές που εκτελούνται στον φυλλομετρητή Google Chrome ή σε πλατφόρμα Android.

Επί παραδείγματι, ο Διακομιστής μιας Εφαρμογής GCM σε μια υλοποίηση δια του πρωτοκόλλου HTTP αποστέλλει το ακόλουθο αίτημα POST, το οποίο κομίζει ένα ομαδικό GCM Μήνυμα Δεδομένων που ενσωματώνει ένα Σχήμα JSON εν αναφορά προς ένα βιντεοπαιχνίδι ονόματι "The Elder Scrolls V: Skyrim" και το οποίο προορίζεται για μια καθορισμένη ομάδα-οικογένεια συζυγών GCM Εφαρμογών Χρήστη με Κλειδαριθμό Ειδοποιήσεως "APA91bGH...OEURwm0k2TXQ".

```
https://gcm-http.googleapis.com/gcm/send
Content-Type:application/json
Authorization:key=AIzaSyZ-1u...0GBYzPu7Udno5aA
```

```
{
  "to": "APA91bGH...OEURwm0k2TXQ"
  "data": {
    "@type": "VideoGame",
    "name": "The Elder Scrolls V: Skyrim",
    "url": "http://www.elderscrolls.com/skyrim",
    "playMode": "SinglePlayer",

    "characterAttribute": {
      "@type": "Thing",
      "name": "Deathbrand Instinct",
      "alternateName": "Ability",
      "description": "Increases armor rating by 100 points
        if wearing all Deathbrand Armor"
    }
  }
}
```

Επί τη υποθέσει της επιτυχημένης διανομής ενός αντίστοιχου ομαδικού GCM Μηνύματος με σκέτο ή υβριδικό Ωφέλιμο Φορτίο σ' όλες τις συναποδέκτριες συζυγείς GCM Εφαρμογές ενός Χρήστη, ο εμπλεκόμενος GCM Διακομιστής Συνδέσεως αποστέλλει εν ακαρεί ένα απαντητικό μήνυμα αναφορικώς προς τη μηδενική αποτυχούσα διαβίβαση στον αιτούντα Διακομιστή Εφαρμογής· όπως είναι φέρ' ειπείν το εξής απαντητικό μήνυμα εν αναφορά προς την επιτυχημένη παράδοση ενός τέτοιου μηνύματος και στις δυο παραλήπτριες GCM Εφαρμογές Χρήστη που συγκροτούν μια ομάδα-οικογένεια ομόλογων Εφαρμογών.

```
{
  "success": 2,
  "failure": 0
}
```

Εν τω άμα τη περιπτώσει της ανεπιτυχούς παραδόσεως ενός ομαδικού GCM Μηνύματος σε μια ή πολλές δυνητικές συναποδέκτριες συζυγείς GCM Εφαρμογές ενός Χρήστη, ο GCM Διακομιστής Συνδέσεως, που έχει αναλάβει τη διανομή του συγκεκριμένου

μηνύματος, αποστέλλει αυτοστιγμεί ένα απαντητικό μήνυμα στον Διακομιστή Εφαρμογής που υπέβαλε τ' αντίστοιχο αίτημα POST, εν αναφορά προς την εξ ολοκλήρου ή την εν μέρει αποτυχούσα διανομή, το οποίο περιέχει τον JSON πίνακα "failed_registration_ids" με τις αντίστοιχες Κανονικές Ταυτότητες των εν δυνάμει αποδεκτριών GCM Εφαρμογών που περιήλθαν εν εκκρεμότητα παραδόσεως του συγκεκριμένου ομαδικού μηνύματος· όπως είναι επί παραδείγματι, το ακόλουθο απαντητικό μήνυμα αναφορικά προς την αποτυχημένη διανομή ενός τέτοιου μηνύματος και στις τρεις δυνητικές παραλήπτριες GCM Εφαρμογές ενός Χρήστη, που σχηματίζουν μια ομάδα-οικογένεια ομόλογων Εφαρμογών.

```
{
  "success":1,
  "failure":2,
  "failed_registration_ids":[
    "regId1",
    "regId2"
  ]
}
```

Το κάτωθι ομαδικό GCM Μήνυμα Δεδομένων που είναι ενσωματωμένο σ' ένα XMPP μήνυμα, προορίζεται για μια καθορισμένη ομάδα-οικογένεια συζυγών GCM Εφαρμογών Χρήστη με Κλειδάριθμο Ειδοποίησης "APA91bGH...OEURwm0k2TXQ" και περιέχει ένα Σχήμα JSON αναφορικά με μια συνταγή μαγειρικής για την παρασκευή ψωμιού με μπανάνα.

```
<message id="m-1366082849205">
<gcm xmlns="google:mobile:data">
{
  "to": "APA91bGH...OEURwm0k2TXQ ",
  "message_id": "m-1366082849205" ,
  "data": {
    "@type": "Recipe",
    "author": "John Smith",
    "cookTime": "PT1H",
    "datePublished": "2017-14-07",
    "description": "This classic banana bread recipe comes
from my mom -- the walnuts add a nice texture and flavor
to the banana bread.",
    "image": "bananabread.jpg",
    "recipeIngredient": [
      "3 or 4 ripe bananas, smashed",
      "1 egg",
      "2 cups all-purpose flour",
      "3/4 cup of sugar"
    ],
    "name": "Mom's World Famous Banana Bread",
    "nutrition": {
      "@type": "NutritionInformation",
      "calories": "240 calories",
      "fatContent": "9 grams fat"
    },
    "prepTime": "PT15M",
    "recipeInstructions": "Preheat the oven to 350 degrees.
Mix in the ingredients in a bowl. Add the flour last.
Pour the mixture into a loaf pan and bake for one
hour."
  }
}
</gcm>
</message>
```

Επί τη υποθέσει της επιτυχημένης ή της εν μέρει αποτυχημένης διανομής ενός αντίστοιχου μηνύματος XMPP, που κομίζει ένα ομαδικό GCM Μήνυμα με μεικτό ή σκέτο Ωφέλιμο Φορτίο, ο εμπλεκόμενος GCM Διακομιστής Συνδέσεως συμφώνως προς το πρωτόκολλο XMPP αποστέλλει αυτοστιγμεί ένα κωδικοποιημένο απαντητικό μήνυμα αναγνώρισης λήψεως και διαβίβασης ACK (acknowledgement) στον αντίστοιχο-αποστολέα Διακομιστή της παραλήπτριας Εφαρμογής GCM, εν αναφορά προς τη μηδενικής αποτυχίας στη διαβίβαση του ομαδικού μηνύματος ή αντιστοίχως της μερικώς επιτυχημένης διανομής του, με το δεύτερο να περιέχει τον προαναφερμένο JSON πίνακα "failed_registration_ids" όπως είναι φέρ' ειπείν κατ' αντιστοιχία τα ακόλουθα δύο Σχήματα JSON.

```
{
  "from": "aUniqueKey",
  "message_type": "ack",
  "success": 3,
  "failure": 0,
  "message_id": "m-1366082849205"
}
```

```
{
  "from": "aUniqueKey",
  "message_type": "ack",
  "success":1,
  "failure":2,
  "failed_registration_ids":[
    "regId1",
    "regId2"
  ]
}
```

Εν τω άμα τη περιπτώσει της εν τω συνόλω αποτυχημένης διαβίβασης ενός τέτοιου XMPP μηνύματος στις δυνητικές συναποδέκτριες GCM Εφαρμογές Χρήστη, ο Διακομιστής Συνδέσεως, που φέρει την αρμοδιότητα της διαβίβασής του, αποστέλλει ένα απαντητικό μήνυμα αποτυχίας διανομής NACK (negative-acknowledgement) στον αντίστοιχο-αποστολέα Διακομιστή της Εφαρμογής GCM.

Ωσαύτως, ένα κάταντες θεματικό μήνυμα GCM με σκέτο ή υβριδικό Ωφέλιμο Φορτίο και μία ή πολλές αντίστοιχες εν δυνάμει συναποδέκτριες Εφαρμογές Χρηστών-Συνδρομητών (Subscribers), οι οποίες έχουν εγγραφεί σ' ένα δεδομένο Θέμα (Topic), ενσωματώνεται σ' ένα αίτημα POST συμφώνως προς το οικείο Πρωτόκολλο Μεταφοράς Υπερκειμένου HTTP ή σ' ένα μήνυμα XMPP βάσει του ομώνυμου Επεκτάσιμου Πρωτόκολλου Μεταφοράς Στοιχείων XML, ενώ περιέχει μια ακολουθία αλφαριθμητικών χαρακτήρων, που έχει διαμορφωθεί με μπουσούλα την κανονική έκφραση "/topics/[a-zA-Z0-9-_.~%]+" και η οποία είναι καταχωρημένη στο οικείο JSON πεδίο "to", που εμφανίζει τον προορισμό κάθε GCM μηνύματος σ' έναν εμπλεκόμενο GCM Διακομιστή Συνδέσεως. Το Θέμα αντιστοιχεί λοιπόν σε μια συγκεκριμένη ομάδα-οικογένεια GCM Εφαρμογών Χρηστών-Συνδρομητών, τουτέστιν ένα σαφώς καθορισμένο υποσύνολο αντίστοιχων συζυγών τεκμηρίων γνησιότητας (Instance ID tokens) στιγμιότυπων μιας δεδομένης Εφαρμογής GCM, που είναι εγγεγραμμένες σ' ένα συγκεκριμένο Θέμα. Δεν υφίσταται καθορισμένο ανώτατο όριο για το πλήθος των θεμάτων ανά Εφαρμογή GCM ούτε για τον αριθμό των εγγεγραμμένων Χρηστών-Συνδρομητών ανά Θέμα, ωστόσο ένα θεματικό GCM μήνυμα με μεικτό ή σκέτο Ωφέλιμο Φορτίο δύναται να κομίζει μέχρι 2 kb πληροφορίας. Επί παραδείγματι, στην επόμενη σελίδα βρίσκεται ένα αίτημα POST που κομίζει ένα GCM Μήνυμα Δεδομένων στις δυνάμει παραλήπτριες GCM Εφαρμογές Παικτών στοιχημάτων, που είναι εγγεγραμμένες στο Θέμα με το όνομα "/topics/InPlay_Odds_By_Fixture_Id", εν αναφορά προς την τρέχουσα απόδοση σημείου ενός εν εξελίξει ποδοσφαιρικού αγώνα όπως ορίζεται από τη στοιχηματική ιστοσελίδα bet365. Ο στοιχηματικός όρος της απόδοσης σημείου δηλώνει τον αριθμητικό συντελεστή που προσδιορίζεται από μια εταιρεία-πρακτορείο στοιχημάτων για κάθε πιθανή έκβαση μιας αθλητικής αναμέτρησης.

```

https://gcm-http.googleapis.com/gcm/send
Content-Type:application/json
Authorization:key=AIzaSyZ-1u...0GBYzPu7Udno5aA
{
  "to": "/topics/InPlay_Odds_By_Fixture_Id",
  "data": [
    {
      "name": "Fulltime Result",
      "bookmaker": {
        "data": [
          {
            "id": 2,
            "name": "bet365",
            "odds": {
              "data": [
                {
                  "label": "Home",
                  "value": "1.89",
                  "suspend": 0,
                  "description": "",
                  "handicap": "",

                  "last_update": {
                    "date": "2017-05-29 19:52:50.000000",
                    "timezone_type": 3,
                    "timezone": "UTC"
                  }
                },
                {
                  "label": "Away",
                  "value": "4.22",
                  "suspend": 0,
                  "description": "",
                  "handicap": "",

                  "last_update": {
                    "date": "2017-05-29 19:52:51.000000",
                    "timezone_type": 3,
                    "timezone": "UTC"
                  }
                }
              ]
            }
          }
        ]
      }
    }
  ]
}

```

Στο μήνυμα XMPP της επόμενης σελίδας είναι ενσωματωμένο ένα GCM Μήνυμα Δεδομένων που προορίζεται για μια καθορισμένη ομάδα-οικογένεια συζυγών GCM Εφαρμογών που είναι εγγεγραμμένες στο Θέμα "/topics/Tv_Stations_By_Fixture_Id" και ανήκουν σε Φιλάθλους του Ποδοσφαίρου ή/και σε υποστηρικτές-οπαδούς ενός εκ των δύο ποδοσφαιρικών συλλόγων που αναμετρώνται στον αγώνα με κωδικό 7611 και το οποίο περιέχει ένα Σχήμα JSON εν αναφορά προς τους ευρωπαϊκούς τηλεοπτικούς σταθμούς που θα καλύπτουν ζωντανά τη συγκεκριμένη ποδοσφαιρική αναμέτρηση.

```

<message id="m-126762856305">
<gcm xmlns="google:mobile:data">
{
  "to": "/topics/Tv_Stations_By_Fixture_Id",
  "message_id": "m-126762856305",
  "data": [
    {
      "fixture_id": 7611,
      "tvstation": "COSMOTE Sport"
    },
    {
      "fixture_id": 7611,
      "tvstation": "DAZN"
    },
    {
      "fixture_id": 7611,
      "tvstation": "Eurosport (Rom) "
    },
    {
      "fixture_id": 7611,
      "tvstation": "Futbol"
    },
    {
      "fixture_id": 7611,
      "tvstation": "SFR Sport 1"
    },
    {
      "fixture_id": 7611,
      "tvstation": "Sky Sport (Ita) "
    },
    ...
  ],
}
</gcm>
</message>

```

- Όπως αναφέρεται άνωθεν, σε μία υλοποίηση ενός Έργου GCM συμφώνως προς το οικείο πρωτόκολλο XMPP καθίσταται επιτεύξιμη η αποστολή ενός ανάντους GCM μηνύματος από μια πιστοποιημένη αποστολέα GCM Εφαρμογή ενός Χρήστη-Πελάτη προς τον συζυγή Διακομιστή-Εξυπηρετητή Εφαρμογής δια των παρεχομένων GCM Διακομιστών Σύνδεσης.

Τα εκ των ων ουκ άνευ συστατικά του αιτήματος διαβίβασης ενός ανάντους GCM μηνύματος είναι τα εξής: (α) το Αναγνωριστικό-Ταυτότητα Αποστολέα (SENDER_ID) του εν δυνάμει παραλήπτη Διακομιστή-Εξυπηρετητή της συζυγούς Εφαρμογής GCM – τουτέστιν ο μοναδικός ακέραιος αριθμός που επικυρώνει τη σύζευξη ενός Διακομιστή-Εξυπηρετητή με κάθε GCM Εφαρμογή Χρήστη-Πελάτη – μετασκευασμένη σε ακολουθία χαρακτήρων και επικολλημένη στη συμβολοσειρά "@gcm.googleapis.com", (β) το μοναδικό αναγνωριστικό-ταυτότητα του μηνύματος (message ID) και τέλος το καθαυτό μήνυμα – ήτοι (γ) το Ωφέλιμο Φορτίο του μηνύματος, το οποίο περιέχει ζευγάρια προκαθορισμένων ονομάτων/τιμών που μορφοποιούν το οικείο JSON αντικείμενο "data". Ένας GCM Διακομιστής Συνδέσεως λοιπόν, εν τω άμα τη ελεύσει ενός τέτοιου αιτήματος διαβίβασης παράγει και εν συνεχεία κομίζει εν ακαρεί στον δυνητικό αποδέκτη Διακομιστή-Εξυπηρετητή Εφαρμογής το αντίστοιχο XMPP μήνυμα, το οποίο εκτός απ' το οικείο JSON αντικείμενο "data" περιέχει ακόμη τα JSON πεδία "category" και "from", τα οποία εμφανίζουν αντιστοίχως το όνομα της Εφαρμογής GCM και την Κανονική Ταυτότητα του αιτούντος στιγμιότυπού της· όπως είναι φέρ' ειπείν το ακόλουθο XMPP μήνυμα εν

αναφορά προς την επανεπικαιροποίηση των γεωγραφικών δεδομένων του Διακομιστή-Εξυπηρετητή μιας GCM Εφαρμογής με την τρέχουσα γεωγραφική θέση – το στίγμα ενός ακίνητου ή κινούμενου χρήστη της.

```
<message id="">
  <gcm xmlns="google:mobile:data">
    {
      "category": "locationTracker",
      "data": {
        {
          "address": "Vokou 48, Chalkida 34100, Greece",
          "coordinates": {
            "latitude": 38.465498,
            "longitude": 23.599606
          }
        },
        "message_id": "m-1004875945437",
        "from": "bk3RNwTe3H0:CI2k_HHwgIpoDKCIZvvDMExUdFQ3P1..."
      }
    }
  </gcm>
</message>
```

Ο Διακομιστής-Εξυπηρετητής και παραλήπτης ενός τέτοιου XMPP μηνύματος, αποστέλλει μονοστιγμής ένα κωδικοποιημένο απαντητικό μήνυμα αναγνώρισης λήψεως ACK (acknowledgement) στον εμπλεκόμενο GCM Διακομιστή Συνδέσεως, εν αναφορά προς την επιτυχημένη διαβίβαση του ανάντους μηνύματος· όπως είναι φέρ' ειπείν το κάτωθι απαντητικό μήνυμα XMPP για το ανάντη μήνυμα με αναγνωριστικό-ταυτότητα "m-1004875945437". Επί τη υποθέσει της ανεπιτυχούς παράδοσης ενός αντίστοιχου XMPP μηνύματος ο εμπλεκόμενος Διακομιστής Συνδέσεως επιχειρεί αύθις τη διαβίβασή του.

```
<message id="">
  <gcm xmlns="google:mobile:data">
    {
      "to": "bk3RNwTe3H0:CI2k_HHwgIpoDKCIZvvDMExUdFQ3P1...",
      "message_id": "m-1004875945437"
      "message_type": "ack"
    }
  </gcm>
</message>
```

Η κλιμακωτή οικοδόμηση μιας GCM Εφαρμογής Χρήστη για έξυπνες φορητές συσκευές και τηλεοράσεις με λειτουργικό σύστημα Android OS

Τα χρειώδη δομικά εργαλεία για το χτίσιμο μιας GCM Εφαρμογής Χρήστη παρέχονται στη Διεπαφή Προγραμματισμού Εφαρμογών – GoogleCloudMessaging API μέσω της Συλλογής Εργαλείων Ανάπτυξης Λογισμικού (Software Development Kit) – Google Play Services SDK. Ντε φάκτο λοιπόν, η εξέλεξη της ορθής λειτουργείας μιας τέτοιας εφαρμογής απαιτεί, είτε τη χρήση μιας συμβατής έξυπνης συσκευής που να τρέχει σε Android 4.0.4 ή επιγενή συζυγή έκδοση με εγκατεστημένη την εφαρμογή Ψηφιακής Διανομής και Ενημέρωσης Λογισμικού για το Android OS – Google Play Store, είτε τη χρησιμοποίηση ενός εξομοιωτή μιας αντίστοιχης έξυπνης συσκευής (Android Emulator) που να τρέχει την Πλατφόρμα Διεπαφών Προγραμματισμού Εφαρμογών της Google (Google APIs Platform) πάνω σε Android 4.2.2 ή υστερογενή συζυγή έκδοση. Για να κατασταθεί όμως εφικτή η χρησιμοποίηση μιας φρέσκιας έκδοσης του GoogleCloudMessaging API εκ της συλλογής Διεπαφών Προγραμματισμού Εφαρμογών των υπηρεσιών Google Play, είναι επιβεβλημένη η συμπερίληψη της αντίστοιχης συμβολοσειράς στις εξαρτήσεις (dependencies) του αρχείου build.gradle της συζυγούς εφαρμογής, όπως είναι για παράδειγμα το εξής.

```
dependencies {  
    compile "com.google.android.gms:play-services-gcm:10.2.0"  
}
```

Η ενδειγμένη διαμόρφωση του αντίστοιχου XML αρχείου ρυθμίσεως παραμέτρων AndroidManifest.xml για τον ορισμό και την περιγραφή της συμπεριφοράς μιας GCM Android Εφαρμογής Χρήστη

Ντε φάκτο, η χαμηλότερη έκδοση της Συλλογής Εργαλείων Ανάπτυξης Λογισμικού Android που δύναται να υποστηρίξει μια GCM Android Εφαρμογή Χρήστη είναι η Android 4.0.4 SDK (Ice Cream Sandwich). Τούτο λοιπόν καθιστά επιβεβλημένο ότι, το χαρακτηριστικό minSdkVersion της ετικέτας <uses-sdk> θα λάβει κατ' ελάχιστο την τιμή 15. Η τιμή του συγκεκριμένου χαρακτηριστικού καθορίζει το χαμηλότερο επίπεδο των Διεπαφών Προγραμματισμού Εφαρμογών – APIs που μπορεί να υποστηρίξει η αντίστοιχη εφαρμογή. Το δε χαρακτηριστικό targetSdkVersion της ετικέτας <uses-sdk> ορίζει το βέλτιστο επίπεδο APIs που δύναται να στηρίξει η αντίστοιχη εφαρμογή Android. Η τιμή του δηλώνει την έκδοση του Android SDK, για την οποία κατασκευάστηκε και στην οποία δοκιμάστηκε η δεδομένη εφαρμογή. Με τον καθορισμό της έκδοσης του SDK που αποτελεί το στόχο της αντίστοιχης εφαρμογής, το οικείο λειτουργικό σύστημα επιχειρεί να αντιστοιχήσει τη δοκιμασμένη συμπεριφορά της εφαρμογής στην ακριβή έκδοση του Android SDK, ακόμη κι όταν εκτελείται σε μια διαφορετική (επιγενή) έκδοση της οικείας πλατφόρμας. Τούτο συνεπάγεται ότι, η δεδομένη εφαρμογή θα συνεχίσει να συμπεριφέρεται με τον οικείο-παλιό τρόπο, παρά τις νέες αλλαγές ή βελτιώσεις που έγιναν στο SDK και οι οποίες δύναται να προκαλέσουν μη σκόπιμες συνέπειες στην αντίστοιχη εφαρμογή (Conder & Darcey, 2010). Επειδή, το Google Play φιλτράρει τις εφαρμογές που διατίθενται σ' ένα συγκεκριμένο χρήστη βάσει ρυθμίσεων όπως η ετικέτα <uses-sdk> στο αρχείο manifest της αντίστοιχης κατόχου εφαρμογής, ο καθορισμός των χαρακτηριστικών αυτής της ετικέτας καθίσταται ντε φάκτο επιβεβλημένος για εκείνες τις εφαρμογές που τίθενται σε κυκλοφορία. Η δε παράλειψη της συγκεκριμένης ετικέτας καταλήγει στην αντίστοιχη προειδοποίηση από το προγραμματιστικό περιβάλλον ανάπτυξης (Conder & Darcey, 2010). Άρα, εντός της στοιχειώδους ομώνυμης ετικέτας του αρχείου manifest μιας GCM Εφαρμογής Χρήστη θα πρέπει να συμπεριλαμβάνεται μια ετικέτα που να προσομοιάζει με την κάτωθι .

```
<uses-sdk android:minSdkVersion="15" android:targetSdkVersion="21"/>
```


Οι εφαρμογές Android a priori δεν έχουν δικαιώματα πρόσβασης στα ευαίσθητα δεδομένα του χρήστη ή στο υλικό της συσκευής του. Τα χρειώδη δικαιώματα για την εκτέλεση μιας εφαρμογής καταχωρούνται ρητώς στο αντίστοιχο αρχείο manifest. Ως εκ τούτου, ο χρήστης δυνητικά ενημερώνεται για τα δικαιώματα που απαιτεί η κάθε εφαρμογή Android και τα εγκρίνει προτού περατωθεί η εγκατάστασή της στη συσκευή του. Επομένως, το αρχείο manifest θα πρέπει να περιλαμβάνει μόνο εκείνα τα δικαιώματα που είναι υποχρεωτικά για την προβλεπόμενη εκτέλεση της αντίστοιχης εφαρμογής. Κάθε τέτοιο δικαίωμα δηλώνεται διαρρήδη σε ξεχωριστές ετικέτες <uses-permission> εντός της ετικέτας <manifest>. Είναι επίσης επιβεβλημένο, όλες οι Δραστηριότητες (Activities), οι Υπηρεσίες (Services) και οι Δέκτες Εκπομπών (Receivers) που συμβάλουν στην προβλεπόμενη εκτέλεση μιας δεδομένης εφαρμογής Android, να καταχωρούνται στο αντίστοιχο αρχείο manifest εντός της ετικέτας <application>. Αυτά τα επιμέρους στοιχεία της κατόχου-εφαρμογής δύνανται να καταχωρίσουν ρητά τα δικαιώματα που παραχωρούν σε άλλες εφαρμογές χρησιμοποιώντας το χαρακτηριστικό android:permission.

Εξ ορισμού, η συμπεριφορά μιας GCM εφαρμογής Android καθιστά επιβεβλημένη τη ρητή εκχώρηση του δικαιώματος πρόσβασης στο διαδίκτυο. Η πρώτη χειραψία μιας δεδομένης GCM Εφαρμογής Χρήστη με τον αντίστοιχο Διακομιστή Εφαρμογής, η οποία επικυρώνει την μεταξύ τους επικοινωνία δια των παρεχόμενων GCM Διακομιστών Σύνδεσης, προϋποθέτει την εξής επαλληλία ενεργειών. Αρχικά, η εφαρμογή εγγράφεται στην υπηρεσία νέφους Instance ID, κατόπιν εκδίδει το κατάλληλο τεκμήριο γνησιότητας (Instance ID token) και τέλος το μεταβιβάζει στο συζυγή Διακομιστή της Εφαρμογής. Επιπλέον, σε μία υλοποίηση ενός Έργου GCM δια του πρωτοκόλλου XMPP, ο αντίστοιχος Διακομιστή Εφαρμογής δύναται να λάβει ανάντη μηνύματα από μια οικεία Εφαρμογή Χρήστη. Η επιτυχής λοιπόν περάτωση των δεδομένων ενέργειων χρήζει πρόσβασης στο διαδίκτυο, άρα το αρχείο manifest μιας GCM εφαρμογής Android είναι επιβεβλημένο να περιλαμβάνει την κατάλληλη ετικέτα καταχώρησης του αντίστοιχου δικαιώματος. Η συμπεριφορά ορισμένων GCM εφαρμογών Android καθιστά δε απαραίτητη την ρητή εκχώρηση του δικαιώματος αποτροπής της μετάβασης του επεξεργαστή της αντίστοιχης Android συσκευής σε κατάσταση ύπνου, οπότε η συζυγής εφαρμογή λαμβάνει ένα GCM μήνυμα. Τούτο εξασφαλίζεται αξιοποιώντας την κλάση PowerManager.WakeLock. Το αρχείο manifest μιας τέτοιας εφαρμογής είναι λοιπόν επιβεβλημένο να συμπεριλαμβάνει μια ετικέτα <uses-permission> όπου θα καταχωρείται η άδεια για WAKE_LOCK. Ως εκ τούτων, εντός της ετικέτας <manifest> του οικείου ομώνυμου αρχείου μιας GCM εφαρμογής Android τοποθετείται ντε φάκτο η πρώτη ετικέτα εκ των κάτωθι και δυνητικά η δεύτερη.

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.WAKE_LOCK" />
```

Όπως προαναφέρθη, μια GCM Εφαρμογή Χρήστη είναι ντε φάκτο επιβεβλημένο να εγγραφεί διαδικτυακά στην υπηρεσία νέφους Instance ID για να λάβει την αντίστοιχη μοναδική ταυτότητα εφαρμογής (Instance ID), που απαιτείται για την έκδοση φρέσκων πιστοποιητικών αυθεντικότητας εφαρμογής ή την ανανέωση των ήδη εκδοθέντων. Σε μια GCM Εφαρμογή Android καθίσταται λοιπόν απαραίτητη η υλοποίηση της μεθόδου onTokenRefresh(), σε αντίστοιχη κλάση η οποία να επεκτείνει την κλάση InstanceIDListenerService, ώστε να δύναται να εκδίδει και να αντικαθιστά τα οικεία τεκμήρια γνησιότητας. Αυτές οι δύο αρμοδιότητες ομού με, τη διαδικασία εκδόσεως της οικείας ταυτότητας εφαρμογής – Instance ID και τις ενέργειες για τη διαβίβαση όλων των πιστοποιητικών ταυτοποίησης στο συζυγή Διακομιστή Εφαρμογής, ενθυλακώνονται σε μια αντίστοιχη υπηρεσία Android, επειδή αποτελούν διεργασίες παρασκηνίου και έτσι ουδεμία απαιτεί είσοδο από τον χρήστη. Όμως, όπως αναφέρθη προηγουμένως, το Android σύστημα a priori διατελεί εν αγνοία των συμπεριλαμβανόμενων υπηρεσιών μιας εφαρμογής, εκτός αν αυτές καθορίζονται ρητώς στο αντίστοιχο αρχείο manifest δια της ετικέτας <service>. Η δε ετικέτα <intent-filter>, η οποία δυνητικά περιέχεται στις ετικέτες <service>, <activity> ή/και <receiver>, φέρει εις γνώσιν του συστήματος όσον αφορά την ονομασία, την κατηγορία και τον τύπο των χρησιμοποιούμενων δεδομένων, μιας παρασκηνιακής πρόθεσης (implicit intents) που η αντίστοιχη εφαρμογή δύναται να δεχτεί μέσω μιας συζυγούς καταχωρημένης υπηρεσίας, δραστηριότητας ή/και ενός αντίστοιχου καθορισμένου δέκτη εκπομπών. Άρα, το αρχείο manifest μιας GCM Εφαρμογής Χρήστη είναι επιβεβλημένο να συμπεριλαμβάνει τον εξής κώδικα XML

(eXtensible Markup Language), ώστε να καθιστά ενήμερο το σύστημα Android για την εν λόγω υπηρεσία.

```
<service android:name=".MyInstanceIDService"
android:exported="false">
  <intent-filter>
    <action android:name="com.google.android.gms.iid.InstanceID"/>
  </intent-filter>
</service>
```

Εξ ορισμού μια GCM Εφαρμογή Χρήστη θα πρέπει να ενωτίζεται τα εισερχόμενα GCM μηνύματα και κατόπιν να τα διοχετεύει καταλλήλως εντός ή/και εκτός της εφαρμογής σύμφωνα με το είδος τους. Ένεκεν τούτου, σε μια τέτοια εφαρμογή Android είναι επιβεβλημένη η υλοποίηση των κάτωθι τεσσάρων μεθόδων της κλάσης GcmListenerService σε μια επεκτεινόμενη κλάση. Η GcmListenerService ανήκει στην οικεία Διεπαφή Προγραμματισμού Εφαρμογών GoogleCloudMessaging.

(α) Της μεθόδου onMessageReceived (String from, Bundle data) που ενεργοποιείται οσάκις η αντίστοιχη εφαρμογή Android παραλαμβάνει ένα εισερχόμενο GCM μήνυμα. Τα στοιχεία του αποστολέα ενός τέτοιου μηνύματος καταχωρούνται στη συμβολοσειρά from, ενώ στην παράμετρο data κρατείται η πληροφορία που διαβιβάζει το συζυγές μήνυμα ως ένα ζεύγος δεδομένων, που σύγκεται από μια συμβολοσειρά-τίτλο-κλειδί και την αντίστοιχη διαβιβαζόμενη τιμή. Η χρήση κάθε τέτοιου μηνύματος ανάγεται στην προβλεπόμενη επιθυμητή συμπεριφορά της συζυγούς GCM Εφαρμογής Χρήστη και σύμφωνα με το είδος του αφιχθέντος μηνύματος. Για παράδειγμα οσάκις μια τέτοια εφαρμογή παραλαμβάνει ένα εισερχόμενο GCM μήνυμα να προκαλείται το στάλισμο μιας αντίστοιχης ειδοποίησης (notification) στον χρήστη της συσκευής.

(β) Της μεθόδου onDeletedMessages() που θέτεται σε λειτουργία, εφόσον σ' έναν εμπλεκόμενο GCM Διακομιστή Σύνδεσης ενεργοποιείται η εξάλειψη των μηνυμάτων που έχουν περιέλθει σε εκκρεμότητα διαβίβασης. Το συμβάν αυτό ανακλύπτει σε έναν GCM Διακομιστή Σύνδεσης ένεκα της υπέρβασης του προβλεπόμενου ορίου αποθήκευσης των αιωρούμενων μηνυμάτων με παραλήπτη μια δεδομένη συσκευή χρήστη. Μια τέτοια αιφνίδια ανωμαλία δύναται να ανακλύψει στην περίπτωση παρατεταμένης ανέφικτης προσπάθειας σε μια συγκεκριμένη συσκευή χρήστη από έναν GCM Διακομιστή Σύνδεσης.

Τέλος των μεθόδων (γ) onMessageSent(String msgId) και (δ) onSendError(String msgId, String error) που αντιστοίχως τίθενται σε λειτουργία, οσάκις ένα άναπτες – εξερχόμενο GCM μήνυμα αποστέλλεται επιτυχώς σε έναν GCM Διακομιστή Σύνδεσης ή αντιθέτως όταν ανακλύπτει ένα αστόχημα κατά τη διαβίβασή του στον συζυγή Διακομιστή Εφαρμογής. Στη συμβολοσειρά msgId, που είναι παράμετρος και στις δύο μεθόδους, κρατιέται η μοναδική ταυτότητα κάθε ανάντους GCM μηνύματος· ενώ στη συμβολοσειρά error, που είναι η δεύτερη παράμετρος στη μέθοδο onSendError, καταχωρείται η ονομασία του σφάλματος που ανέκυψε κατά διαβίβαση του GCM μηνύματος που κατέχει την αντίστοιχη ταυτότητα εξερχομένου μηνύματος.

Καθεμία εκ των παραπάνω μεθόδων αντιστοιχεί σε μια παρασκευαστική GCM διεργασία που δεν απαιτεί κάποια είσοδο από τον χρήστη. Ως εκ τούτου, οι οικείες διεργασίες ενθυλακώνονται σε μια αντίστοιχη υπηρεσία Android. Είναι λοιπόν επιβεβλημένο, το αρχείο manifest μιας GCM εφαρμογής Android να συμπεριλαμβάνει τον ακόλουθο κώδικα XML.

```
<service
  android:name=".MyGcmListenerService"
  android:exported="false" >
  <intent-filter>
    <action android:name="com.google.android.c2dm.intent.RECEIVE"/>
  </intent-filter>
</service>
```

Η παράδοση όμως ενός GCM μηνύματος στη συζυγή Android εφαρμογή καθίσταται εφικτή, υπό την προϋπόθεση ότι ενυπάρχει σ' αυτή και η αντίστοιχη γραμματοθυρίδα GCM. Εκ των ων ουκ άνευ για μια τέτοια εφαρμογή είναι λοιπόν ένας αντίστοιχος Android δέκτης, ο οποίος λειτουργεί ως παραλήπτης (GcmReceiver) των εκπεμπόμενων GCM μηνυμάτων. Ντε φάκτο, το οικείο αρχείο manifest κάθε GCM εφαρμογής Android είναι απαραίτητο να συμπεριλαμβάνει τον παρακάτω κώδικα XML, ώστε να καθιστά ενήμερο το σύστημα Android για τον εν λόγω δέκτη – γραμματοθυρίδα των μηνυμάτων GCM. Καθίσταται δε επιβεβλημένη η ρητή εκχώρηση του δικαιώματος της διαβίβασης τέτοιων μηνυμάτων στον οικείο δέκτη μέσω των παρεχόμενων GCM Διακομιστών Σύνδεσης. Οσάκις ο εν λόγω δέκτης παραλαμβάνει ένα τέτοιο μήνυμα το παραδίδει απευθείας στην προαναφερόμενη κλάση που αποτελεί επέκταση της GcmListenerService. Τούτο διεκπεραιώνεται με την αρμοστή υλοποίηση της μεθόδου onReceive (Context context, Intent intent) της κλάσης GcmReceiver σε μια επεκτεινόμενη κλάση. Η GcmReceiver περιλαμβάνεται επίσης στη Διεπαφή Προγραμματισμού Εφαρμογών GoogleCloudMessaging.

```
<receiver
    android:name="com.google.android.gms.gcm.GcmReceiver"
    android:exported="true"
    android:permission="com.google.android.c2dm.permission.SEND">
    <intent-filter>
        <action android:name="com.google.android.c2dm.intent.RECEIVE"/>
        <category android:name="PACKAGE_NAME" />
    </intent-filter>
</receiver>
```

Διακρίβωνοντας τη συμβατότητα μεταξύ της βερσιόν του Πακέτου Διανομής και Εγκατάστασης Λογισμικού Εφαρμογών Android – APK (Android Package Kit) των Υπηρεσιών Google Play που έχει εγκατασταθεί στη συσκευή του Χρήστη και της αντίστοιχης συλλογής Διεπαφών Προγραμματισμού Εφαρμογών – APIs (Application Programming Interface) των οικείων υπηρεσιών που χρησιμοποιεί η GCM Εφαρμογή Android.

Όπως προαναφέρθη, μια GCM Εφαρμογή Android χτίζεται με τα δομικά εργαλεία που παρέχει η Διεπαφή Προγραμματισμού Εφαρμογών – GoogleCloudMessaging API μέσω της Συλλογής Εργαλείων Ανάπτυξης Λογισμικού Android (Software Development Kit) – Google Play Services SDK. Τούτου δοθέντος και λαμβανομένου δε υπ' όψιν ότι, ντε φάκτο δεν είναι γνωστή η βερσιόν του Πακέτου Διανομής και Εγκατάστασης Λογισμικού Android – APK (Android Package Kit) των Υπηρεσιών Google Play που είναι ενδεχομένως εγκατεστημένο και ενεργοποιημένο σε καθεμιά Android OS συσκευή, όπου θα τρέχει δυνητικά μια τέτοια εφαρμογή, απαιτείται το τσεκάρισμα της συμβατότητας της εγκατεστημένης-ενεργοποιημένης βερσιόν με εκείνη που χρησιμοποιείται στη δεδομένη εφαρμογή, προτού εκτελεστεί οποιαδήποτε διεργασία που είναι συυφασμένη με κάποιο κομμάτι κώδικα από την οικεία Διεπαφή Προγραμματισμού Εφαρμογών – GoogleCloudMessaging API. Για την εξέταση της εν λόγω συμβατότητας είναι λοιπόν απαραίτητο, να αποκτηθεί προκαταρκτικά δια της μεθόδου GoogleApiAvailability.getInstance() το μοναδικό στιγμιότυπο (singleton instance) της αντίστοιχης κλάσης στην αρχική-πυροδοτούσα δραστηριότητα μιας GCM εφαρμογής και ακολούθως να χρησιμοποιηθεί η μέθοδος isGooglePlayServicesAvailable (Context context), που φανερώνει την αίτια της ανυπαρξίας ή την ύπαρξη της επιδιωκόμενης συμφωνίας. Με τη χρήση της μεθόδου αυτής αποκτάται ένας ακέραιος αριθμός που απεικονίζεται με αμφιμονοσήμαντη αντιστοίχιση σε ένα στοιχείο του εξής πεπερασμένου υποσυνόλου της συλλογής των τελικών στατικών ακέραιων τιμών της κλάσης ConnectionResult, {SUCCESS, SERVICE_MISSING, SERVICE_UPDATING, SERVICE_VERSION_UPDATE_REQUIRED, SERVICE_DISABLED, SERVICE_INVALID}. Εάν το αποτέλεσμα της οικείας μεθόδου είναι το 0, που απεικονίζεται στην σταθερά ConnectionResult.SUCCESS, τότε υπάρχει η επιζητούμενη συμβατότητα. Εν εναντία περιπτώσει το αποτέλεσμα της οικείας εξετάζουσας μεθόδου αντιστοιχεί σε μια εκ των υπολοίπων σταθερών του προαναφερθέντος υποσυνόλου που φανερώνει τον πρόξενο της υπάρχουσας αναντιστοιχίας. Προκειμένου να καταστεί λοιπόν ενήμερος ο χρήστης της περί ης ο λόγος εφαρμογής για τη δεδομένη ανωμαλία και να διακανονίσει δυνητικά την υπάρχουσα τροχοπέδη, χρησιμοποιείται η μέθοδος getErrorDialog (Activity activity, int errorCode, int requestCode) του

προαναφερθέντος μοναδικού στιγμιότυπου της κλάσης `GoogleApiAvailability`. Η μέθοδος τούτη δημιουργεί ένα αντικείμενο της κλάσης `Dialog`, η οποία διαθέτει τις θεμελιώδεις μεθόδους για τον ορισμό και τον έλεγχο του κύκλου ζωής ενός συνήθους–απλού παράθυρου διαλόγου Android. Το παραγόμενο διαδραστικό παράθυρο επισήμανσης πληροφορεί λοιπόν τον χρήστη για τη διαπιστωμένη τροχοπέδη και εκείνος κλικάροντας πάνω του κατευθύνεται είτε στις ρυθμίσεις του συστήματος Android στην περίπτωση που οι Υπηρεσίες Google Play είναι εγκαταστημένες στη συσκευή του αλλά έχουν απενεργοποιηθεί, είτε στο κατάστημα της Google (δηλαδή στην εφαρμογή για την Ψηφιακή Διανομή και Ενημέρωση του Λογισμικού Android OS – Google Play Store) στην περίπτωση που δεν έχει καν εγκατασταθεί στην αντίστοιχη συσκευή το APK των οικείων υπηρεσιών (Πακέτο Διανομής και Εγκατάστασης Λογισμικού Android – Application Package Kit) ή είναι εγκατεστημένη και ενεργοποιημένη μια βερσιόν του που είναι όμως ασύμβατη με την GCM Εφαρμογή. Η μητέρα–δραστηριότητα που κατέχει το οικείο ενημερωτικό παράθυρο διαλόγου ορίζεται δια της παραμέτρου `activity`. Κατ' ακολουθίαν, η γλώσσα του περι ου ο λόγος παραθύρου ορίζεται μέσω αυτής της παραμέτρου. Το προϊόν της γνωστής μεθόδου `isGooglePlayServicesAvailable` (`Context context`) περνιέται στην παράμετρο `errorCode` και στην παράμετρο `requestCode` εκχωρείται μια ακέραιη σταθερά τιμή που ορίζει τον κωδικό ανταπόκρισης–ανατροφοδότησης για την επανάκληση της μητέρας–δραστηριότητας του οικείου πληροφοριακού παράθυρου διανιδράσεως επί σκοπώ να κατασταθεί ενήμερη αναφορικά με το απότοκο της απόφασης του χρήστη επί του διαπιστωμένου κωλύματος. Στο κάτωθι τμήμα της αρχικής–πυροδοτούσας δραστηριότητα μιας GCM εφαρμογής Android κωδικοποιείται λοιπόν η διακρίβωση της περί ης ο λόγος επιδιωκόμενης συμβατότητας.

```
static final int REQUEST_CODE_GOOGLE_PLAY_SERVICES = 2017;

@Override
protected void onResume () {
    super.onResume ();
    if (GooglePlayServicesAvailability_isEnsured ()) {

        /* Καθίσταται δυνατή η εκτέλεση διεργασιών που σχετίζονται με
        Προγραμματισμού Εφαρμογών - APIs των Υπηρεσιών Google Play. */
    }
}

private boolean GooglePlayServicesAvailability_isEnsured () {

    GoogleApiAvailability
    googleApiAvailability = GoogleApiAvailability.getInstance ();

    int statusCode =
    googleApiAvailability.isGooglePlayServicesAvailable (this);

    if (statusCode != ConnectionResult.SUCCESS) {
        if (googleApiAvailability
        .isUserRecoverableError (statusCode)) {
            /* Το διαπιστωμένο πρόβλημα δύναται να διευθετηθεί με
            απόφαση του χρήστη. */

            googleApiAvailability.getErrorDialog (this, statusCode,
            REQUEST_CODE_GOOGLE_PLAY_SERVICES).show ();
        } else {
            Toast.makeText (this, "Η συσκευή δεν υποστηρίζεται",
            Toast.LENGTH_LONG).show ();
            finish ();
        }
        return false;
    }
    return true;
}
}
```

```

@Override
protected void onActivityResult
    (int requestCode, int resultCode, Intent data) {

    /* Μόλις ο χρήστης λάβει μια απόφαση επί του διακριβωμένου κωλύματος
    μέσω του οικείου ενημερωτικού διαλογικού παραθύρου και επανέλθει στην
    παρούσα δραστηριότητα, το λειτουργικό σύστημα πυροδοτεί τη μέθοδο
    onActivityResult για να της μεταφέρει το αποτέλεσμα της απόφασής του
    μέσω της παραμέτρου resultCode. */

    if (requestCode == REQUEST_CODE_GOOGLE_PLAY_SERVICES) {
        if (resultCode == RESULT_CANCELED) {
            Toast.makeText(this, "Απαιτείται η εγκατάσταση και η
            ενεργοποίηση της επικαιροποιημένης βερσιόν του APK των
            υπηρεσιών Google Play.", Toast.LENGTH_SHORT).show();
            finish();
        }
    }
    super.onActivityResult(requestCode, resultCode, data);
}

```

Η εγγραφή μιας GCM Android Εφαρμογής Χρήστη στην Υπηρεσία Νέφους Instance ID – Έκδοση, Ανανέωση των συζυγών Τεκμηρίων Γνησιότητας

Η παρακάτω γραμμή κώδικα επιστρέφει τη συμβολοσειρά (iid) της συζυγούς μοναδικής ταυτότητας ενός στιγμιότυπου της εκάστης Εφαρμογής Χρήστη (Instance ID) που πιστοποιεί τη εγγραφή της στην οικεία υπηρεσία νέφους Instance ID.

```
String iid = InstanceID.getInstance(context).getId();
```

Με τον κάτωθι κώδικα εκδίδεται το μοναδικό πιστοποιητικό γνησιότητας με την κλήση της μεθόδου getToken(authorizedEntity,scope).

```

String authorizedEntity = PROJECT_ID;
/* Η συμβολοσειρά authorizedEntity φέρει τον συζυγή κλειδάριθμο του
Λογισμικού μίας Εφαρμογής (Project ID) που εκχωρείται στους διαχειριστές
της από την Κονσόλα Προγραμματιστών της Google (Google Developers
Console. */

String scope = "GCM";
/* Η συμβολοσειρά χαρακτήρων scope αφορά στα αίτια της έκδοσης του
συζυγούς τεκμηρίου αυθεντικότητας. Η δεδομένη συμβολοσειρά δύναται να
ενθυλακώνει από κανέναν έως και χίλιους χαρακτήρες που θα πρέπει να
εμπίπτουν στους «Ακίνδυνους» χαρακτήρες για Ενιαίους Εντοπιστές Πόρων
(URL-Safe Characters)» βάσει των προδιαγραφών για Uniform Resource
Locators (URL), RFC 1738 των Berners-Lee, Masinter και McCahill.
(1994)*/

String token =
InstanceID.getInstance(context).getToken(authorizedEntity,scope);

```

Η εκτέλεση του ακόλουθου τμήμα κώδικα ανανεώνει ένα τεκμήριο γνησιότητας.

```
@Override
public void onTokenRefresh() {
    // Λήψη ενός νέου τεκμηρίου γνησιότητας και ενημέρωση του συζυγή
    // Διακομιστή Εφαρμογής
    Intent intent = new Intent(this, RegistrationIntentService.class);
    startService(intent);
}
```

Αποστολή Ανάντων Μηνυμάτων

```
public class MessageSender {
    AsyncTask<Void, Void, String> sendTask;
    AtomicInteger ccsMsgId = new AtomicInteger();
    public void sendMessage(final Bundle data, final
        GoogleCloudMessaging gcm) {

        sendTask = new AsyncTask<Void, Void, String>() {
            @Override
            protected String doInBackground(Void... params) {
                String id =
                    Integer.toString(ccsMsgId.incrementAndGet());
                try {
                    /* Τα απαραίτητα στοιχεία του αιτήματος μεταφοράς ενός
                    ανάντους GCM μηνύματος από έναν εμπλεκόμενο GCM Διακομιστή
                    Συνδέσεως είναι τα ακόλουθα: (α) το Αναγνωριστικό-Ταυτότητα
                    Αποστολέα (SENDER_ID) του δυνάμει αποδέκτη Διακομιστή-
                    Εξυπηρετητή της Εφαρμογής GCM - όπερ δηλώνει τον μοναδικό
                    ακέραιο αριθμό που επικυρώνει τη σύζευξη ενός Διακομιστή-
                    Εξυπηρετητή με μια GCM Εφαρμογή Χρήστη-Πελάτη - που είναι
                    μετασκευασμένη σε ακολουθία χαρακτήρων και επικολλημένη στη
                    στη συμβολοσειρά "@gcm.googleapis.com", (β) το μοναδικό
                    αναγνωριστικό-ταυτότητα του μηνύματος και (γ) το Ωφέλιμο
                    Φορτίο του μηνύματος, το οποίο περιέχει ζευγάρια
                    προκαθορισμένων ονομάτων/τιμών που μορφοποιούν το οικείο
                    JSON αντικείμενο "data". */
                    gcm.send(Config.GOOGLE_PROJECT_ID +
                        "@gcm.googleapis.com", id, data);
                } catch (IOException e) {
                    Log.d(TAG, "Exception: " + e);
                    e.printStackTrace();
                }
                return "Message ID: "+id+ " Sent.";
            }
        };
        @Override
        protected void onPostExecute(String result) {
            sendTask = null;
            Log.d(TAG, "onPostExecute: result: " + result);
        }
    };
    sendTask.execute(null, null, null);
}
```

Η Android Εφαρμογή Χρήστη για την άμεση ανταλλαγή μηνυμάτων μέσω GCM

Ο Διακομιστής της Εφαρμογής GCM ανεπτύχθη σε εικονική μηχανή – Virtual Machine (VM) Windows Server 2012 (2 CPUs, 4Gb Memory Size, 40Gb Disk Size, public IPv4 - 83.***.116.***) της υπηρεσίας Cyclades του υπολογιστικού νέφους ~Okeanos, που ανήκει στις υπολογιστικές και αποθηκευτικές υπηρεσίες του Εθνικού Δικτύου Έρευνας & Τεχνολογίας (ΕΔΕΤ ΑΕ - GRNET S.A.) και κτίσθηκε με τα εξής εργαλεία:

- Notepad++
- Mozilla Firefox
- Apache HTTP Server
- MySQL
- PHP
- phpmyadmin
- NetBeans IDE

Η Android Εφαρμογή άμεσης αμφίδρομης επικοινωνίας μέσω μηνυμάτων GCM εκπονήθηκε στο ολοκληρωμένο περιβάλλον ανάπτυξης λογισμικού (Integrated Development Environment - IDE) Android Studio 1.4 για SDK version 21 και δοκιμάστηκε στις Android συσκευές Samsung Galaxy Note 6, S4, S5, S6 καθώς και σε πλήθος διαφορετικών εικονικών συσκευών Android (Android Virtual Devices - AVDs) που κατασκευάστηκαν εργαλείο Genymotion, το οποίο επισυνάφθηκε στο Android Studio. Το περιβάλλον διεπαφής χρήστη φτιάχτηκε συμφώνως προς το σύνολο των σχεδιαστικών κανόνων του Material Design της Google και δομήθηκε από γραφικά που δημιουργήθηκαν με τις γεννήτριες γραφικών Notification Icon Generator, Generic Icon Generator και Android Icon Animator της δωρεάν συλλογής εργαλείων παραγωγής γραφικών για Android εφαρμογές Android Asset Studio.

Στην κεντρική οθόνη της περί ης ο λόγος Εφαρμογής ενυπάρχουν τρεις καρτέλες που αντιστοιχούν στις εξής λειτουργίες:

- Επισήμανση της εσχάτως καταγεγραμμένης γεωγραφικής θέσεως των φίλων του χρήστη σε ενσωματωμένο χάρτη.
- Εμφάνιση της λίστας των έγγραμμων (online) φίλων του χρήστη.
- Προβολή του καταλόγου των συνομιλιών του χρήστη.

Πέραν των άνωθεν καρτελών, οι τρεις κάτωθι αναφερόμενες οθόνες κείνται στην οικεία Εφαρμογή Χρήστη:

- Αρχική Οθόνη – Splash Screen, τουτέστιν η εναρκτήριος οθόνη της εφαρμογής
- Οθόνη Εγγραφής – Registration Screen, η οποία ενσωματώνει μια φόρμα εγγραφής για την καταχώρηση ενός νέου χρήστη στο οικείο σύστημα άμεσης ανταλλαγής μηνυμάτων.
- Οθόνη άμεσης αποστολής και λήψης μηνυμάτων κειμένου και φωτογραφιών – Instant Messaging Screen που ενσωματώνει τη διεπιφάνεια ανταλλαγής άμεσων μηνυμάτων γραπτής μορφής ή φωτογραφιών ανάμεσα σε δύο χρήστες, ενώ εμφανίζει στους συνδιαλεγόμενους το ιστορικό της συνομιλίας τους.

Αρχική Οθόνη – Splash Screen

Η εικόνα που βρίσκεται στην αρχή της επόμενης σελίδας είναι η εναρκτήρια οθόνη της εκπονημένης Εφαρμογής και η συμπερίληψή της αποσκοπεί αφενός μεν στο καλωσόρισμα του Χρήστη, αφετέρου δε στην απόσπαση της προσοχής του ενόσω η Εφαρμογή εκτελούσα τον κώδικα Java της μεθεπόμενης σελίδας ελέγχει εάν το τεκμήριο γνησιότητας της συσκευής του είναι καταχωρημένο στη βάση δεδομένων του συζυγή Διακομιστή Εφαρμογής.

Στιγμιότυπο Οθόνης 1: Η εναρκτήρια οθόνη της Android Εφαρμογής GCM**Εργασία με το AsyncTask**

Το Android Software Development Kit (SDK) παρέχει δύο εύκολους τρόπους διαχείρισης της ελάττωσης της επιβάρυνσης στην επεξεργασία απ' το κύριο νήμα διεπιφάνειας χρήστη (User interface - UI): την κλάση AsyncTask και την τυπική για τη Java κλάση Thread. Η κλάση AsyncTask είναι μια ειδική κλάση για ανάπτυξη με το Android η οποία παρέχει επεξεργασία στο παρασκήνιο και συμβάλλει στην διευκόλυνση της επικοινωνίας με το νήμα UI ενώ γίνεται διαχείριση του κύκλου ζωής της εργασίας του παρασκηνίου μέσα στο πλαίσιο του κύκλου ζωής της δραστηριότητας. Οι προγραμματιστές μπορούν επίσης να κατασκευάσουν τις δικές τους λύσεις αλληλουχίας ενεργειών χρησιμοποιώντας τυπικές μεθόδους και κλάσεις της Java – αλλά είναι τότε υπεύθυνες για τη διαχείριση και ολόκληρου του κύκλου ζωής του νήματος επίσης. Η

μεταφορά των δεδομένων της οικείας Εφαρμογής προς και από τον συζυγή Διακομιστή Εφαρμογής γίνεται ντε φάκτο στο παρασκήνιο μέσω της αφηρημένης βοηθητικής κλάσης AsyncTask τηρώντας το οικείο πρωτόκολλο HTTP.

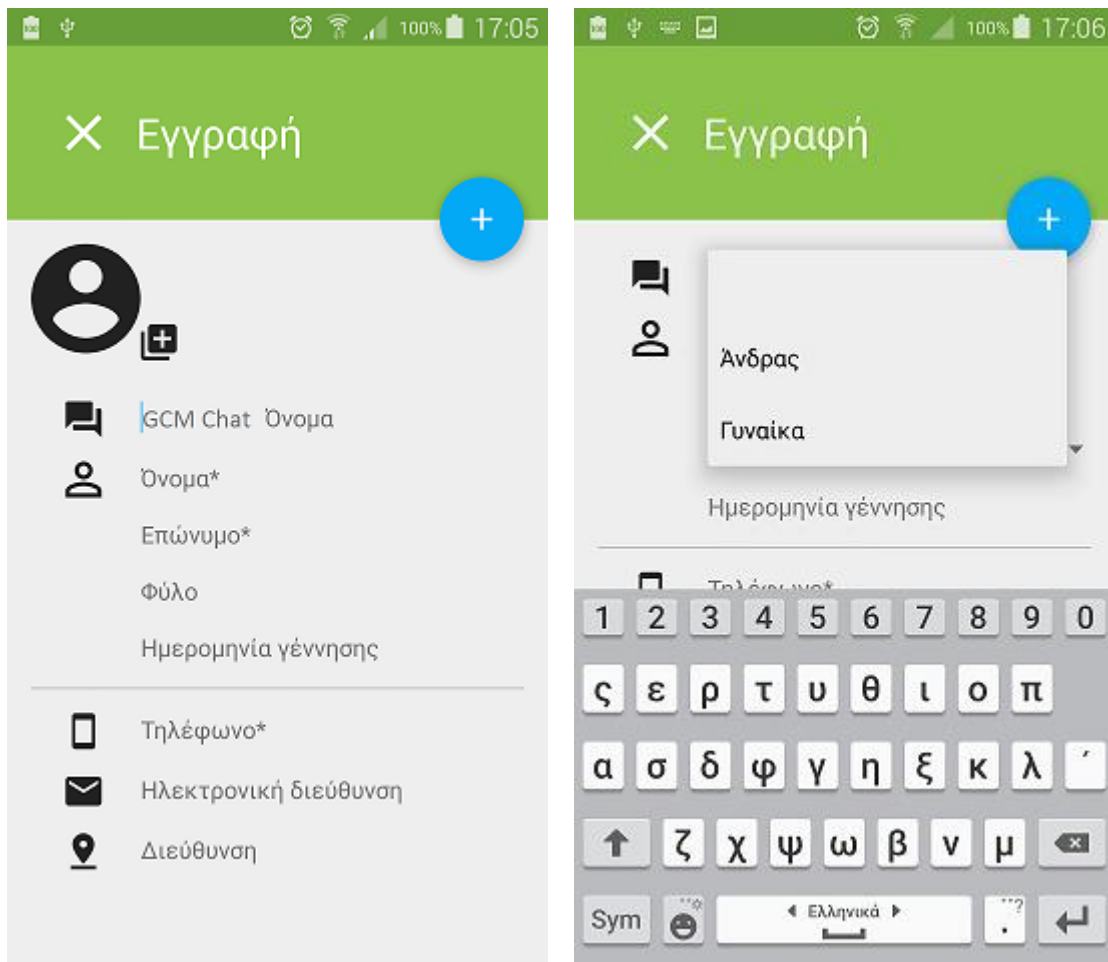
```
class CheckGoogleID extends AsyncTask<String, String, String>{
    int success;
    int update_google_id_flag = 0;
    @Override protected String doInBackground(String... arg0) {
        if(!sharedPreferences_google_id.equals("")){
            try{
                List<NameValuePair> params = new
                ArrayList<NameValuePair>();
                params.add(new
                BasicNameValuePair(TAG_MOBILE_ID, mobile_id));
                json =
                jsonParser.makeHttpRequest(GET_GOOGLE_ID_URL,
                "POST", params);
                success= json.getInt(TAG_SUCCESS);
                if (success == 1){
                    Server_google_id =
                    json.getString(TAG_GOOGLE_ID);
                if(!(Server_google_id.equals(sharedPreferences_google_id)))
                update_google_id_flag = 1;}
                else return json.getString(TAG_MESSAGE);
                }catch(JSONException e){
                    e.printStackTrace();}
                }//!sharedPreferences_google_id.equals("")
                return null;
            }//doInBackground
            protected void onPostExecute(String file_url) {
                if (file_url != null)
                Toast.makeText(SplashScreen.this, file_url,
                Toast.LENGTH_LONG).show();
                if(update_google_id_flag == 1) new
                UpdateGoogleID().execute();
            }//onPostExecute
        }//class CheckGoogleID extends AsyncTask

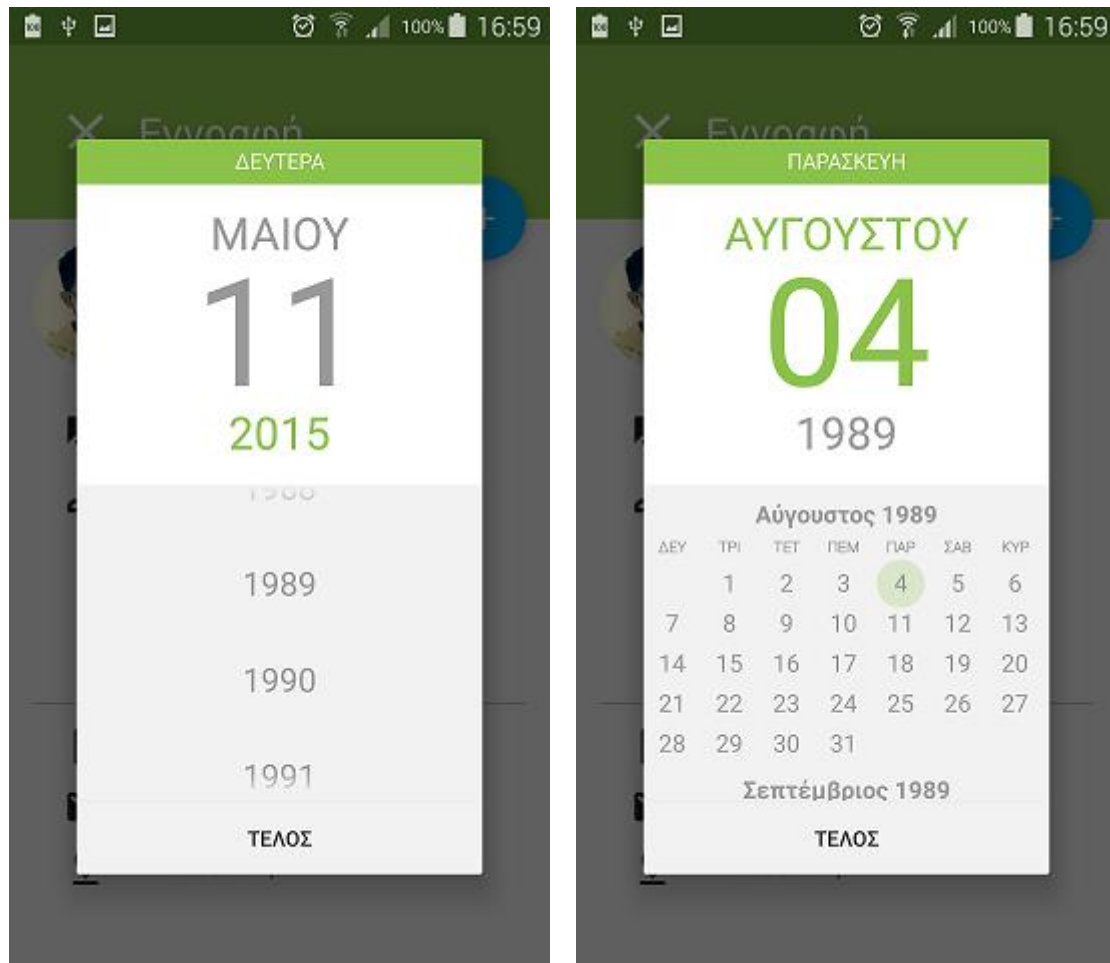
class UpdateGoogleID extends AsyncTask<String, String, String> {
    @Override protected String doInBackground(String... arg0) {
        try{
            List<NameValuePair> params = new
            ArrayList<NameValuePair>();
            params.add(new BasicNameValuePair(TAG_MOBILE_ID,
            mobile_id));
            params.add(new BasicNameValuePair(TAG_GOOGLE_ID,
            sharedPreferences_google_id));
            json =
            jsonParser.makeHttpRequest(UPDATE_GOOGLE_ID_URL,
            "POST", params);
            int success = json.getInt(TAG_SUCCESS);
            if (success != 1){
                startActivity(getIntent());
                finish();}
            }catch(JSONException e){
                e.printStackTrace();}
            return null;
        }
    }//class UpdateGoogleID extends AsyncTask
```

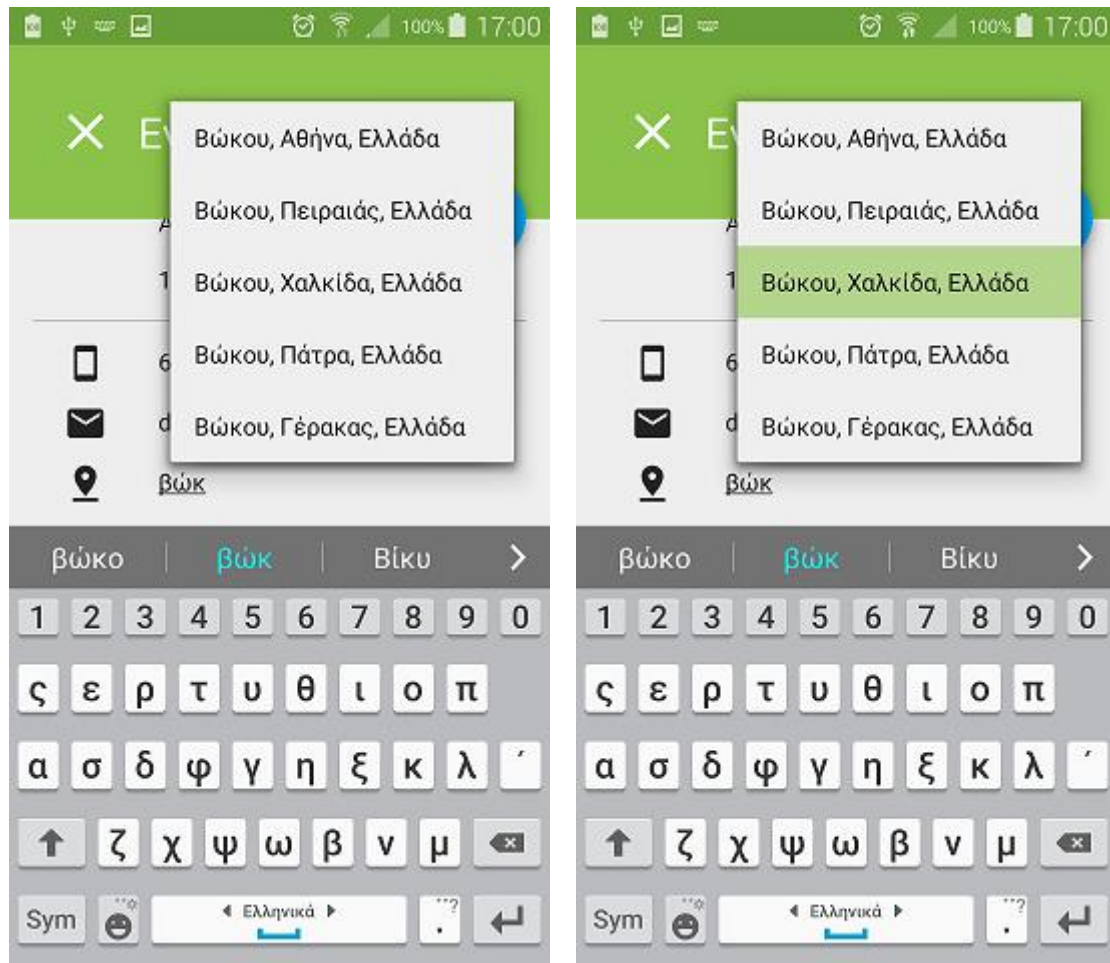
Οθόνη Εγγραφής – Registration Screen

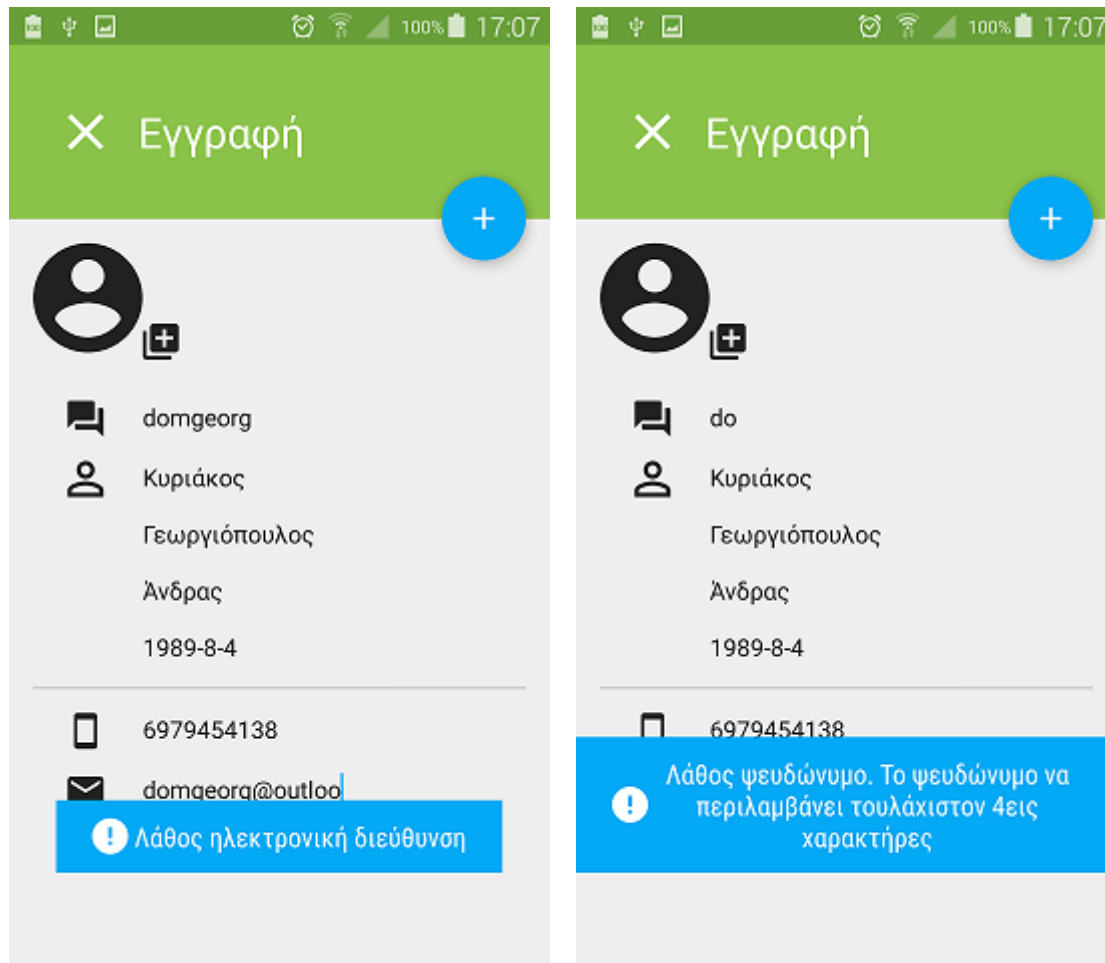
Η παρθενική πυροδότηση της περιγραφόμενης Εφαρμογής Χρήστη προκαλεί την εμφάνιση της οθόνης εγγραφής. Τα κάτωθι στιγμιότυπα οθόνης προβάλλουν τα συστατικά της αντίστοιχης διεπαφής Χρήστη. Ο μη εγγεγραμμένος χρήστης παρακινείται να συμπληρώσει την ενυπάρχουσα φόρμα εγγραφής δίδοντας τα ακόλουθα στοιχεία: φωτογραφία προφίλ χρήστη, όνομα χρήστη, όνομα, επώνυμο, φύλο, ημερομηνία γέννησης, τηλέφωνο, διεύθυνση ηλεκτρονικού ταχυδρομείου και διεύθυνση διαμονής. Εφόσον ο νέος χρήστης συμπληρώσει σωστά τη φόρμα εγγραφής, εκτελείται ο κώδικας που βρίσκεται στη σελίδα # προκειμένου να καταχωρηθούν τα στοιχεία του στη βάση δεδομένων του συζυγή Διακομιστή Εφαρμογής.

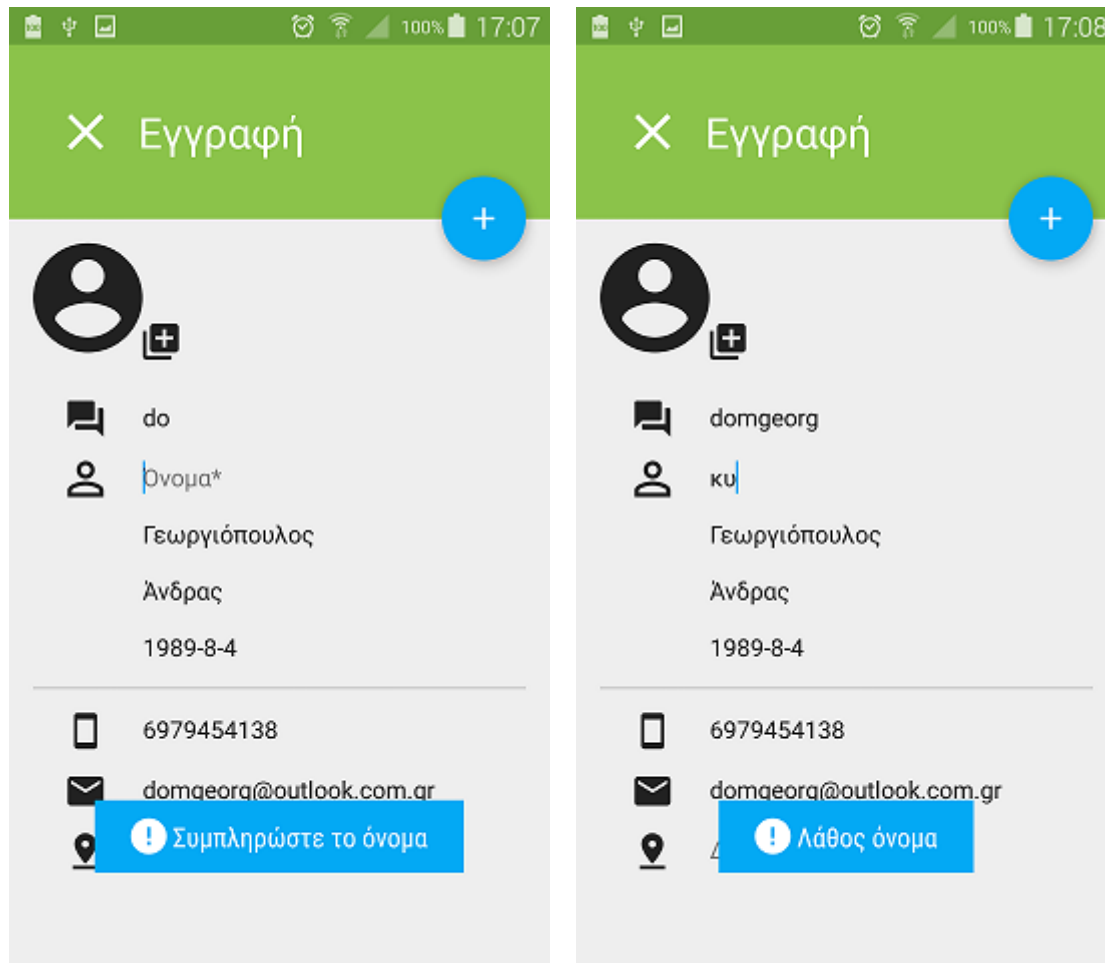
Στιγμιότυπα Οθόνης 2: Η οθόνη εγγραφής νέου χρήστη της Android Εφαρμογής GCM

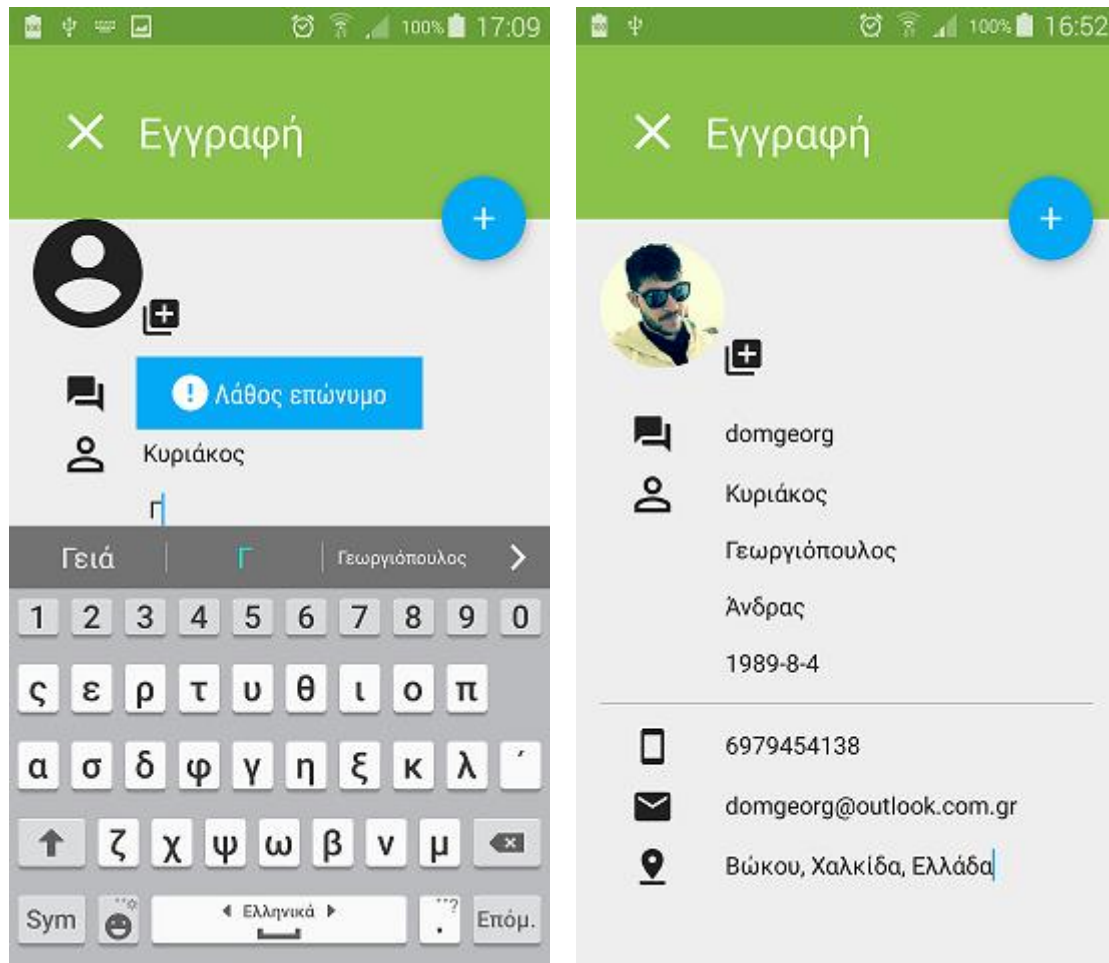


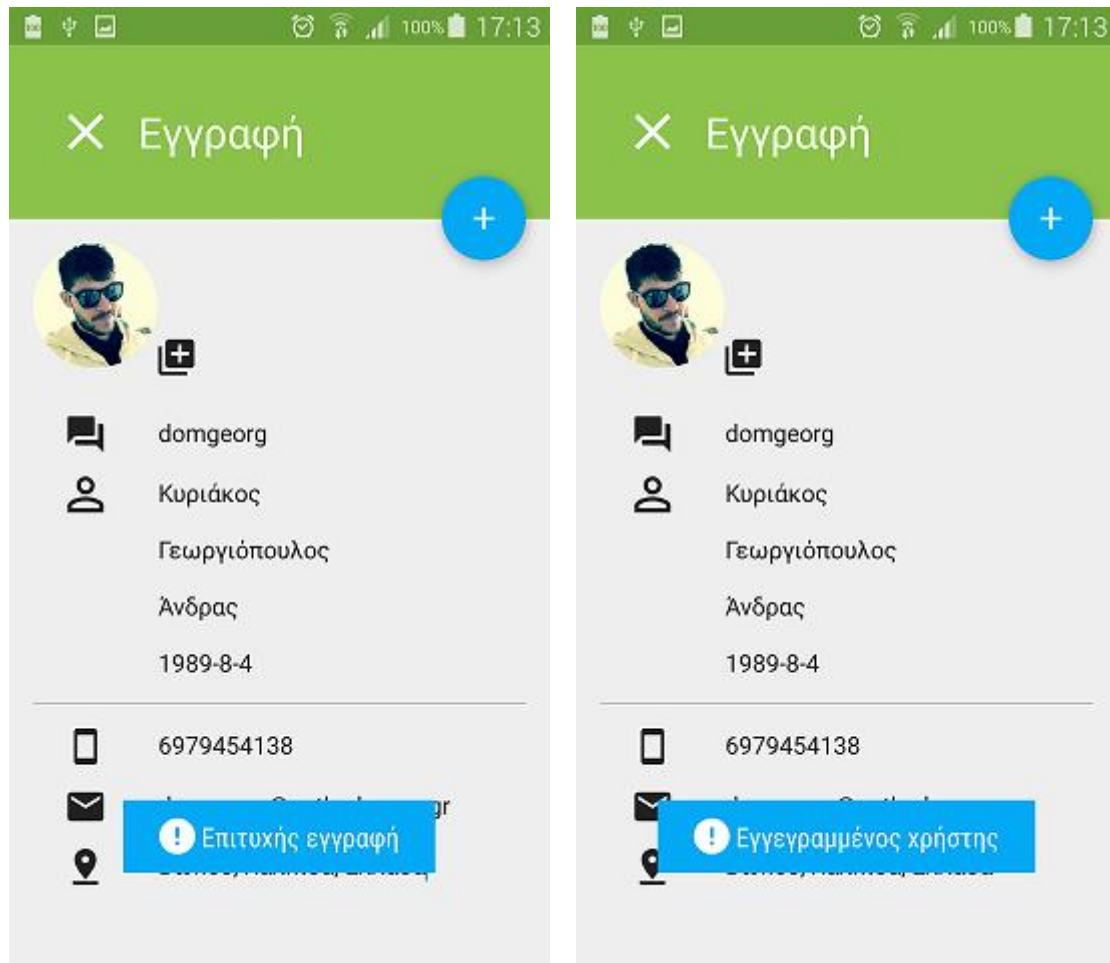












Number of rows: 25

+ Options												
	mobile_no	firstname	lastname	username	gender	birthday	e_mail	address	photo_path	status		
<input type="checkbox"/>	6979454138	Κυριάκος	Γεωργιόπουλος	domgeorg	Άνδρας	1989-8-4	domgeorg@outlook.com.gr	Βώκου, Χαλκίδα, Ελλάδα	photos/d-05-2015-1431353606.jpg	1		

Number of rows: 25

```

@Override protected String doInBackground(String... args){
    MultipartEntity params = new MultipartEntity();
    HttpClient client = new DefaultHttpClient();
    HttpPost post = new HttpPost(url);
    try {
        params.addPart(user.PHONE_NUMBER, new
        StringBody(phone_number, Charset.forName("UTF-8")));
        params.addPart(user.FIRSTNAME, new
        StringBody(firstname, Charset.forName("UTF-8")));
        params.addPart(user.LASTNAME, new
        StringBody(lastname, Charset.forName("UTF-8")));
        params.addPart(user.USERNAME, new
        StringBody(username, Charset.forName("UTF-8")));
        params.addPart(user.GENDER, new StringBody(gender,
        Charset.forName("UTF-8")));
        params.addPart(user.BIRTHDAY, new
        StringBody(birthday, Charset.forName("UTF-8")));
        params.addPart(user.EMAIL, new StringBody(e_mail,
        Charset.forName("UTF-8")));
        params.addPart(user.ADDRESS, new StringBody(address,
        Charset.forName("UTF-8")));
        params.addPart(user.STATUS, new StringBody("1",
        Charset.forName("UTF-8")));
        if (uploadedimage!=null) params.addPart("upload",
        uploadedimage);
        post.setEntity(params);
        HttpResponse response1 = client.execute(post);
        HttpEntity resEntity = response1.getEntity();
        String Response = EntityUtils.toString(resEntity);
        JSONArray jsonarray = new
        JSONArray("[ "+Response+" ]");
        jsonobject= jsonarray.getJSONObject(0);
        return jsonobject.getString("message");
    }catch (UnsupportedEncodingException e){
        e.printStackTrace();
    }catch (JSONException e){
        e.printStackTrace();
    }catch (ParseException e){
        e.printStackTrace();
    }catch (IOException e){
        e.printStackTrace();
    }
    return null;
}

@Override protected void onPostExecute(String arg0){
    superActivityToastProgress.dismiss();
    if (arg0!=null){
        superActivityToastServerMessage.setText(arg0);
        superActivityToastServerMessage.show();
        try {if (jsonobject.getInt("success")==1){
            EditSharedPreferences();
            context.startActivity(new Intent(context,
            SplashScreen.class));
            ((Activity) context).finish();
        }
        } catch (JSONException e) {
            e.printStackTrace();
        }
    }
}

```

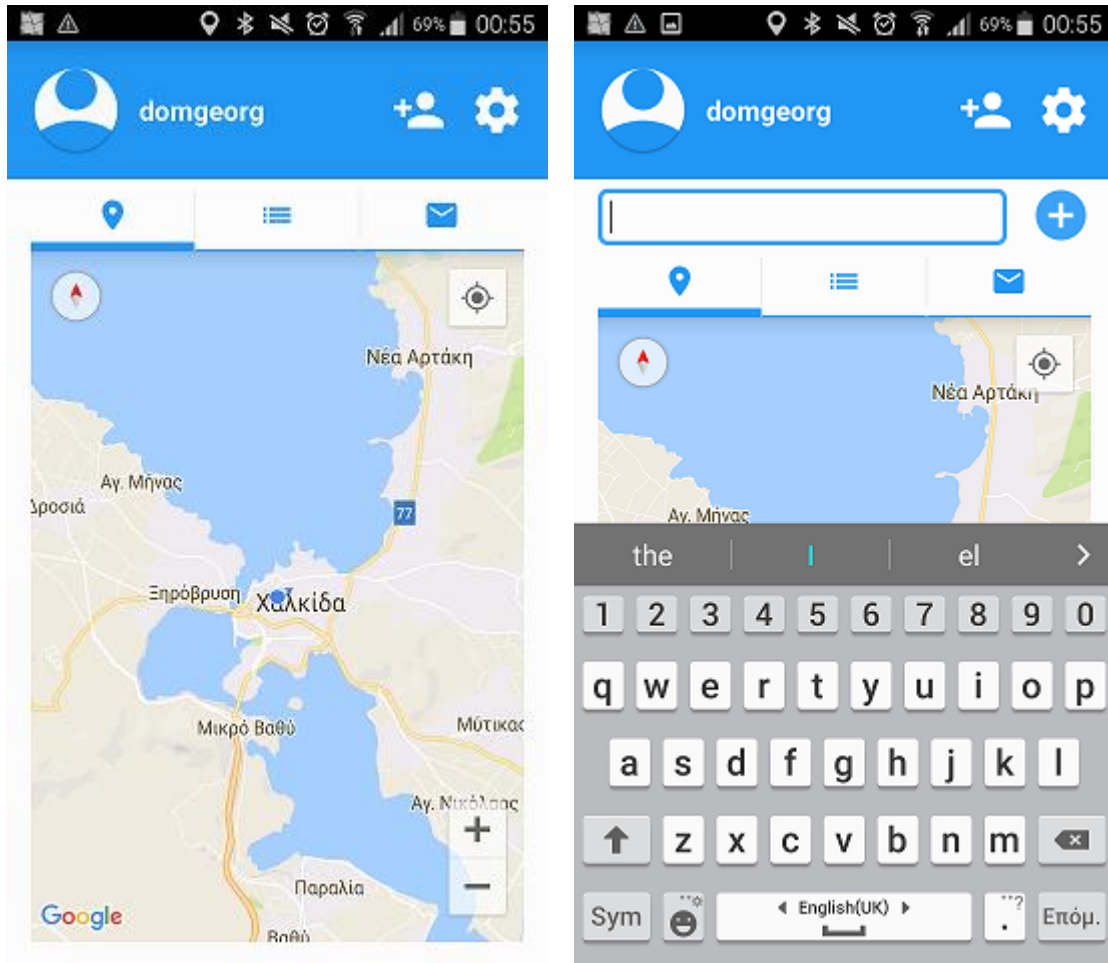
Κεντρική Οθόνη – Καρτέλες: Χάρτης, Λίστα Έγγραμμων Επαφών-Φίλων Χρήστη, Κατάλογος Συνομιλιών Χρήστη

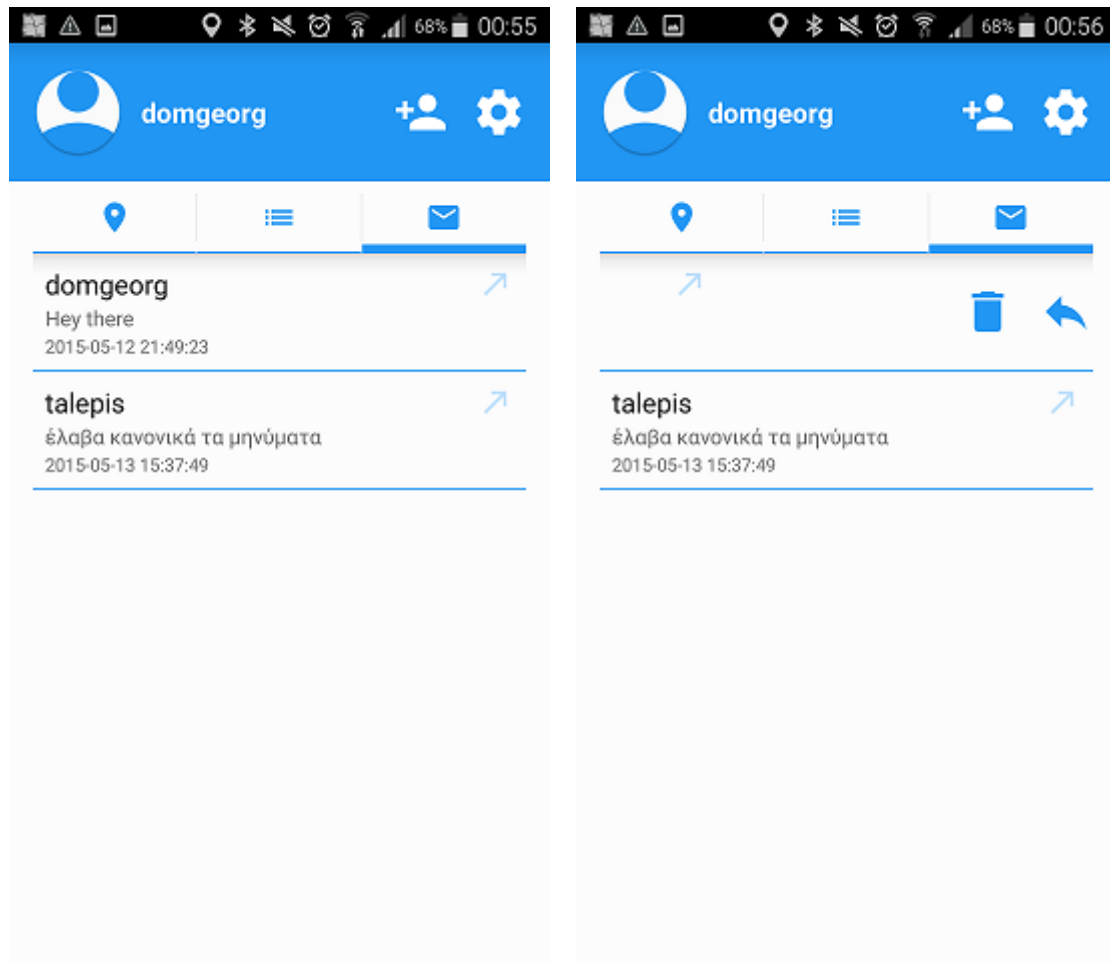
Η εμφάνιση της κεντρικής οθόνης στην οικεία Εφαρμογή Χρήστη πυροδοτείται εν ακαρεί είτε από την επιτυχημένη εγγραφή ενός νέου χρήστη, είτε από την αυθεντικοποίηση ενός ήδη εγγεγραμμένου χρήστη από τον Διακομιστή Εφαρμογής. Ο χρήστης μπορεί να κλικάρει το εικονίδιο προσθήκης νέου φίλου για να προσθέσει έναν άλλον εγγεγραμμένο χρήστη στη λίστα των φίλων του. Στη σελίδα 47 βρίσκεται το τμήμα κώδικα που εκτελείται προκειμένου να υποβληθεί ένα αίτημα σύναψης φιλίας.

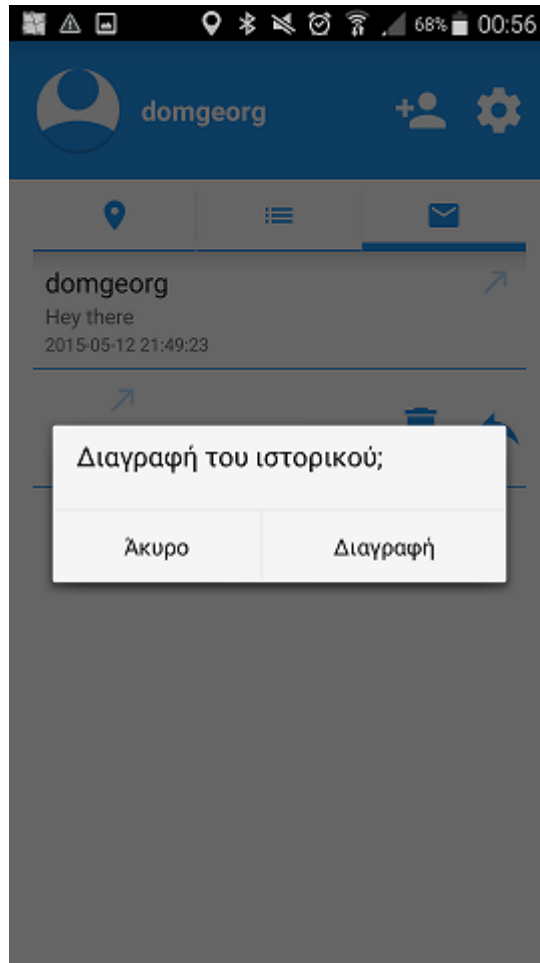
Η πρώτη καρτέλα της κεντρικής οθόνης ενσωματώνει τον χάρτη της τρέχουσας γεωγραφικής θέσης του χρήστη όπου επισημαίνεται δε, το τελευταίο καταγεγραμμένο γεωγραφικό στίγμα κάθε άλλου έγγραμμου χρήστη που ανήκει στη λίστα των φίλων του. Με την εκτέλεση του κώδικα των σελίδων 48 και 49 φορτώνεται η λίστα των φίλων ενός χρήστη απ' τον συζυγή Διακομιστή της Εφαρμογής και κατόπιν σημαδεύεται η αντίστοιχη γεωγραφική τους θέση στον χάρτη. Ο χρήστης δύναται να επιλέξει μια τέτοια επισήμανση-σημάδι προκειμένου να μεταβεί στην οθόνη άμεσης ανταλλαγής μηνυμάτων και να συνομιλήσει απευθείας με τον φίλο του που βρίσκεται σε κοντινή απόσταση απ' αυτόν. Για να μπορέσει όμως ένας χρήστης να εντοπίζει τους έγγραμμους φίλους του, η εν λόγω Εφαρμογή πρώτα τραβάει τις γεωγραφικές συντεταγμένες του, είτε από το δίκτυο κινητής τηλεφωνίας στο οποίο ανήκει η κινητή συσκευή του, είτε απ' το GPS (Global Positioning System - Παγκόσμιο Σύστημα Στιγματοθέτησης) της συσκευής του, είτε από το WLAN (Wireless Local Area Network - Ασύρματο Τοπικό Δίκτυο) στο οποίο είναι συνδεδεμένη η συσκευή του και έπειτα τις μεταφέρει στον συζυγή Διακομιστή Εφαρμογής προς καταχώρηση. Το γεωγραφικό στίγμα ενός χρήστη καταγράφεται με τη συναίνεσή του και το στάλσιμό του εκτελείται κάθε φορά που αυτός ανακατευθύνεται στην κεντρική οθόνη ή στην οθόνη ανταλλαγής μηνυμάτων.

Η δεύτερη καρτέλα της βασικής οθόνης, ενσωματώνει τη λίστα με τους έγγραμμους φίλους ενός χρήστη και φορτώνεται από τον Διακομιστή Εφαρμογής ομοιοτρόπως με αυτή του χάρτη. Η συμπερίληψη της καρτέλας αυτής, αποσκοπεί στην προβολή όλων των έγγραμμων φίλων ενός χρήστη – δηλαδή κι αυτών που βρίσκονται σε μακρινή απόσταση και επομένως δεν εμφανίζονται άμεσα στον χάρτη. Ο χρήστης μπορεί να συνομιλήσει με έναν φίλο του κλικάροντας το αντίστοιχο στοιχείο της λίστας.

Ο κατάλογος με τις συνομιλίες του χρήστη βρίσκεται στην τρίτη καρτέλα της κεντρικής οθόνης. Ο χρήστης επιλέγοντας ή σύροντας ένα στοιχείο του καταλόγου, μπορεί να κλικάρει τα γνωστά εικονίδια του κάδου ή του βέλους για να διαγράψει το ιστορικό της συνομιλίας από την τοπική βάση δεδομένων ή να μεταβεί στην οθόνη άμεσης ανταλλαγής μηνυμάτων και να συνεχίσει την κουβέντα με τον φίλο του.

Στιγμιότυπα Οθόνης 3: Η κεντρική οθόνη της Εφαρμογής Χρήστη – Καρτέλα Χάρτη, Προσθήκη νέου φίλου, Κατάλογος Συνομιλιών Χρήστη





```

class AddContact extends AsyncTask<String, String, String>{
    int success;
    @Override
    protected String doInBackground(String... arg0){
        try{
            List<NameValuePair> params = new
            ArrayList<NameValuePair>();
            params.add(new BasicNameValuePair(TAG_MOBILE_ID,
            mobile_id));
            params.add(new BasicNameValuePair(TAG_USERNAME,
            contact_username));
            json = jsonParser.makeHttpRequest(ADD_CONTACT_URL,
            "POST", params);
            success = json.getInt(TAG_SUCCESS);
            if(success==1) contact_google_id =
            json.getString(TAG_GOOGLE_ID);
            return json.getString(TAG_MESSAGE);
        }catch (JSONException e){
            e.printStackTrace();
        }
        return null;
    } //doInBackground
    protected void onPostExecute(String file_url){
        if(success==1){
            contact_field.setVisibility(View.GONE);
            add_contact_field_is_visible = false;
            editor.putBoolean("CONTACT_FIELD", false);
            notifyNewContact(contact_google_id);
            startActivity(getIntent());
            finish();
        }
        if (file_url!=null) Toast.makeText(Main.this, file_url,
        Toast.LENGTH_LONG).show();
    } //onPostExecute
} //class AddContact extends AsyncTask

```

```

class AddUsersOnMap extends AsyncTask<String, String, String>{
    int success,users_length;

    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressDialog.setMessage("Φόρτωση");
        progressDialog.setIndeterminate(false);
        progressDialog.setCancelable(true);
        progressDialog.show();
    }//onPreExecute

    protected String doInBackground(String... arg0){
        String
        mobile_id=null,username=null,current_location=null,photo_p
        ath=null,profile_photo_url=null,google_id=null;
        LatLng markerPosition=null;
        BlockedUsersDatabaseHelper blockedUsers = new
        BlockedUsersDatabaseHelper(context);
        SQLiteDatabase db = blockedUsers.getReadableDatabase();
        try{
            List<NameValuePair> params = new
            ArrayList<NameValuePair>();
            params.add(new BasicNameValuePair(TAG_MOBILE_ID,
            device_mobile_id));
            json =
            jsonParser.makeHttpRequest(ADD_USERS_ON_MAP_URL,
            "POST", params);
            success = json.getInt(TAG_SUCCESS);
            if (success == 1){
                users_length = json.getInt(TAG_LENGTH);
                jsonUsersListNode = json.optJSONObject(TAG_LIST);
                for(int i=1; i<=users_length; i++){
                    String position = String.valueOf(i);
                    try{
                        jsonChildNode =
                        jsonUsersListNode.getJSONObject(position);
                        mobile_id =
                        jsonChildNode.getString(TAG_MOBILE_ID);
                        username =
                        jsonChildNode.getString(TAG_USERNAME);
                        photo_path =
                        jsonChildNode.getString(TAG_PHOTO_PATH);
                        google_id =
                        jsonChildNode.getString(TAG_GOOGLE_ID);
                        markerPosition = new
                        LatLng((Double.parseDouble(jsonChildNode.getString(TAG_LATITUDE))), (Do
                        uble.parseDouble(jsonChildNode.getString(TAG_LONGITUDE))));
                        profile_photo_url =
                        PHOTO_PATH_URL+photo_path;
                    }
                }
            }
        }
    }
}

```



```

try{
Geocoder geocoder = new Geocoder(context);
Iterator<Address>addresses =
geocoder.getFromLocation(Double.parseDouble(jsonChildNode.getStr
ing(TAG_LATITUDE)),Double.parseDouble(jsonChildNode.getString(TA
G_LONGITUDE)), 1).iterator();
if (addresses != null){
    while (addresses.hasNext()){
        Address namedLoc = addresses.next();
        int addIdx = namedLoc.getMaxAddressLineIndex();
        while (addIdx >= 0){
            String addLine =
            namedLoc.getAddressLine(addIdx);
            if (addLine != null){
                current_location = String.format("%s",
                addLine);
            }
            addIdx--;}}
        }catch (IOException ioe){
            ioe.printStackTrace();}
        }catch(JSONException e){
            e.printStackTrace();}
        SQLiteQueryBuilder queryBuild = new SQLiteQueryBuilder();

        queryBuild.setTables(BlockedUsers.BLOCKED_USERS_TABLE_NAME);
        queryBuild.appendWhere(BlockedUsers.BLOCKED_USER_MOBILE_NO +
        "=" + mobile_id + "");
        Cursor cur = queryBuild.query(db, null, null, null, null,null,
        null);
        if(cur.getCount() == 0){
            mClusterManager.addItem(new User(mobile_id,username,
            current_location,profile_photo_url,markerPosition,go
            ogle_id));
            }//the user has not blocked
        }//for loop
    }//if (success == 1)
    else return json.getString(TAG_MESSAGE);
    }catch(JSONException e){
        e.printStackTrace();
    }
    return null;
}
}doInBackground

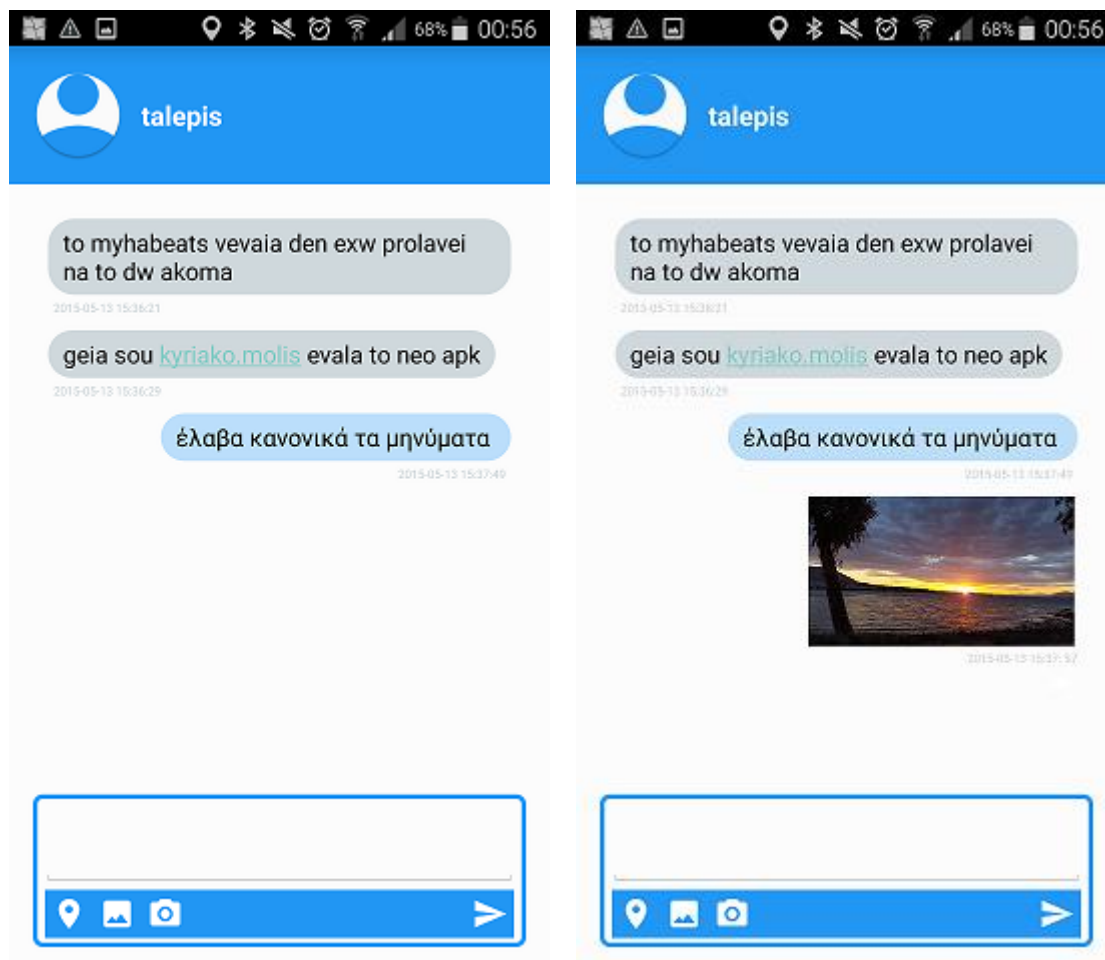
protected void onPostExecute(String file_url){
    progressDialog.dismiss();
    if (file_url != null)
        Toast.makeText(MapTabFragmentActivity.this,file_url,
        Toast.LENGTH_LONG).show();
    }//onPostExecute
}
}AddUsersOnMap

```

Οθόνη άμεσης αποστολής και λήψης μηνυμάτων κειμένου και φωτογραφιών – Instant Messaging Screen

Η διεπιφάνεια άμεσης ανταλλαγής μηνυμάτων εμφανίζει το ιστορικό μιας συνομιλίας ενώ δίνει τη δυνατότητα στους συνομιλούντες να στείλουν και να λάβουν μηνύματα κειμένου, φωτογραφίες και εικόνες. Η διαδικασία της ανταλλαγής των μηνυμάτων περιγράφεται αναλυτικά στα προηγούμενα κεφάλαια της παρούσας πραγματείας. Η αποστολή μιας φωτογραφίας ή μιας εικόνας προϋποθέτει το επιτυχημένο ανέβασμά της στον Διακομιστή Εφαρμογής και κατόπιν το στάλισμο του αντίστοιχου συνδέσμου στην συζυγή εφαρμογή του συνομιλητή ώστε να την κατεβάσει και μονοστιγμής να την προβάλει στη συνομιλία. Ο αντίστοιχος κώδικας για το ανέβασμα και το κατέβασμα βρίσκεται στις σελίδες 51, 52 και 53.

Στιγμιότυπα Οθόνης 4: Η Οθόνη άμεσης ανταλλαγής μηνυμάτων



```

class UploadImage extends AsyncTask<String, String, String>{
    int success;

    @Override
    protected void onPreExecute () {
        super.onPreExecute ();
        progressDialog.setMessage ("Αποστολή εικόνας");
        progressDialog.setIndeterminate (false);
        progressDialog.setCancelable (true);
        progressDialog.show ();
    } //onPreExecute

    @Override
    protected String doInBackground (String... arg0) {
        try {
            MultipartEntity params = new MultipartEntity ();
            HttpClient client = new DefaultHttpClient ();
            HttpPost post = new HttpPost (STORE_IMAGE_MESSAGE);
            try {
                params.addPart (TAG_SENDER_MOBILE_ID, new
StringBody (device_mobile_id, Charset.forName ("UTF-8")));
                params.addPart (TAG_RECEIVER_MOBILE_ID, new
StringBody (interlocutor_mobile_id, Charset.forName ("UTF-8")));
                params.addPart (TAG_SEND_TIMESTAMP, new
StringBody (getCurrentTimestamp (), Charset.forName ("UTF-8")));
            } catch (UnsupportedEncodingException e) {
                e.printStackTrace ();
            }
            params.addPart ("upload", uploadedimage);
            post.setEntity (params);
            HttpResponse response1 = client.execute (post);
            HttpEntity resEntity = response1.getEntity ();
            String Response = EntityUtils.toString (resEntity);
            JSONArray jsonarray = new JSONArray ("["+Response+"]");
            JSONObject jsonobject = jsonarray.getJSONObject (0);
            success = jsonobject.getInt (TAG_SUCCESS);
            if (success == 1) imageUrlToSend =
jsonobject.getString (TAG_PHOTO);
            else return jsonobject.getString (TAG_MESSAGE);
        } catch (JSONException e) {
            e.printStackTrace ();
        } catch (ClientProtocolException e) {
            e.printStackTrace ();
        } catch (IOException e) {
            e.printStackTrace ();
        }
        return null;
    } //doInBackground
}

```

```

protected void onPostExecute(String file_url){
    progressDialog.dismiss();
    if(success == 1){
        try{
            getImage(new
                URL(PHOTO_PATH_URL+ImageUrlToSend), false,
                true);
            sendMessage(ImageUrlToSend);

            message_status_TextView.setText("Pending");
        } catch (MalformedURLException e){
            e.printStackTrace();
        }
        if (file_url != null)
            Toast.makeText(ChatActivity.this, file_url,
                Toast.LENGTH_LONG).show();
    } //onPostExecute
} //class UploadImage extends AsyncTask

class GetPhoto extends AsyncTask<String, String, String>{
    @Override
    protected String doInBackground(String... arg0){
        try{
            List<NameValuePair> params = new
                ArrayList<NameValuePair>();
            params.add(new
                BasicNameValuePair(TAG_MOBILE_ID,
                interlocutor_mobile_id));
            json =
                jsonParser.makeHttpRequest(GET_PHOTO_PATH_
                H_URL, "POST", params);
            int success = json.getInt(TAG_SUCCESS);
            if (success == 1) photo_path =
                PHOTO_PATH_URL+json.getString(TAG_PHOTO_PATH);
            else return json.getString(TAG_MESSAGE);
        } catch (JSONException e){
            e.printStackTrace();
        }

        return null;
    } //doInBackground
}

```

```

protected void onPostExecute(String file_url){
    if(!photo_path.equals("")){
        ImageLoader.getInstance().displayImage(photo_p
            ath, profile_photo_ImageView, defaultOptions,
            new ImageLoadingListener(){
                @Override
                public void onLoadingStarted(String imageUrl, View
                    view){
                    profile_photo_ImageView.setImageResource(R.drawable.
                        ic_empty_user_photograph);
                }
                @Override
                public void onLoadingFailed(String imageUrl, View
                    view, FailReason failReason){
                    profile_photo_ImageView.setImageResource(R.drawable.
                        ic_empty_user_photograph);
                }
                @Override
                public void onLoadingComplete(String imageUrl, View
                    view, Bitmap loadedImage){
                    profile_photo_ImageView.setImageBitmap(getRoundedSha
                        pe(loadedImage));
                }
                @Override
                public void onLoadingCancelled(String imageUrl, View
                    view){
                    profile_photo_ImageView.setImageResource(R.drawable.
                        ic_empty_user_photograph);
                }
            }, new ImageLoadingProgressListener(){
                @Override
                public void onProgressUpdate(String imageUrl, View
                    view, int current, int total){
                    profile_photo_ImageView.setImageResource(R.drawable.
                        ic_empty_user_photograph);
                }
            });
    }
    if (file_url != null)
        Toast.makeText(ChatActivity.this,file_url,
            Toast.LENGTH_LONG).show();
} //onPostExecute
} //class GetPhoto extends AsyncTask

```

Κατακλείδα και Ερευνητικές Προτάσεις επί τη βάσει της παρούσας πραγματείας

Στο πλαίσιο της παρούσας μεταπτυχιακής διατριβής κτίσθηκε ένα ολοκληρωμένο σύστημα άμεσης αμφίδρομης διανθρώπινης επικοινωνίας μέσω της δωρεάν υπηρεσίας Google Cloud Messaging. Οι χρήστες της περι ης ο λόγος Android GCM Εφαρμογής, μέσω της εύχρηστης και μινιμαλιστικής διεπαφής της, δύνανται να ανταλλάσσουν άνευ χρηματικού αντιτίμου μηνύματα κειμένου και φωτογραφίες σε πραγματικό χρόνο. Το εκπονημένο έργο λογισμικού που τεκμηριώνεται στην παρούσα πραγματεία, πληροί λοιπόν την απαίτηση των μελών της σύγχρονης «έξυπνης» κι «ασύρματης» κοινωνίας για την εν ευθέτω χρόνω κι εν ακαρεί εξ αποστάσεως αποστολή και λήψη διαφόρων ειδών πληροφορίας και συνάμα αποτελεί θεμέλιο λίθο για την οικοδόμηση ανθεκτικών Εφαρμογών ευρείας κλίμακας αλλά και εφελτήριο για τη διερεύνηση των κάτωθι:

- Θέματα ασφάλειας που συνδέονται με τις εφαρμογές άμεσης ανταλλαγής μηνυμάτων. Το τοπίο των απειλών είναι αρκετά σύνθετο: κακόβουλο και ανεπιθύμητο λογισμικό μπορεί να θέσει σε κίνδυνο την εμπιστευτικότητα αλλά και την ακεραιότητα διάφορων ευαίσθητων προσωπικών δεδομένων ή κρίσιμων πληροφοριών που διατηρεί ο χρήστης στην έξυπνη συσκευή του (La Polla, Martinelli, & Sgandurra, 2013). Εισβολείς μπορούν να διεισδύσουν στη συσκευή του χρήστη και να υποκλέψουν τις συνομιλίες του. Μελέτες έχουν δείξει ότι μια κινητή συσκευή που μεταβαίνει σ' ένα ασύρματο τοπικό δίκτυο (wireless local area networking - Wi-Fi), είναι επιρρεπής σε επιθέσεις "Man In The Middle" (MITM). Η MITM είναι μια μέθοδος κατά την οποία οι εισβολείς αυτοσυστήνονται σε μία ροή επικοινωνίας. Στην ουσία ο εισβολέας, παίζει τον ρόλο του μεσολαβητή για μία συζήτηση, καταγράφοντας όλες τις πληροφορίες μεταξύ των δύο επικοινωνούντων μερών (Juniper, 2011). Ανάλογα με το πώς μια έξυπνη συσκευή χειρίζεται τη μετάβαση δεδομένων, οι επικοινωνίες μεταξύ δύο μερών μπορούν να μεταδοθούν μέσω απλού κειμένου και έτσι να είναι ορατές σε έναν εισβολέα που χρησιμοποιεί MITM.
- Θέματα που προκύπτουν από την ανάλυση των κοινωνικών δικτύων χρηστών αντίστοιχων εφαρμογών. Η ανάλυση κοινωνικών δικτύων χρησιμοποιεί τη θεωρία δικτύων για να αναλύσει κοινωνικές σχέσεις. Χρησιμοποιείται ευρύτατα στη σύγχρονη κοινωνιολογία, αλλά σχετίζεται και με την ανθρωπολογία, την κοινωνική ψυχολογία, τη βιολογία και άλλους επιστημονικούς κλάδους. Ένα κοινωνικό δίκτυο, ένα σύνολο δηλαδή ανθρώπων οι οποίοι έχουν σχέσεις μεταξύ τους, αναπαρίσταται ως ένας γράφος που αποτελείται από κόμβους και συνδέσεις. Ένας κόμβος αντιστοιχεί σε ένα άτομο, ενώ οι συνδέσεις δηλώνουν την ύπαρξη σχέσης μεταξύ των ατόμων. Οι σχέσεις μεταξύ των ατόμων προκύπτουν από αλληλεπιδράσεις μεταξύ τους (Bonchi, Castillo, Gionis & Jaimés, 2011). Πλέον είναι διαθέσιμος ένας τεράστιος όγκος δεδομένων, που αφορά υπαρκτούς ανθρώπους και τις σχέσεις τους. Τα δεδομένα αυτά, με τη μορφή γράφων, είναι διαφορετικού τύπου από τα δεδομένα που τηρούνται στις παραδοσιακές σχεσιακές βάσεις δεδομένων, και η επεξεργασία τους συνιστά μια πρόκληση, καθώς μπορούν να αποτελέσουν πηγή χρήσιμης και διαφορετικής πληροφορίας. Οι σχέσεις μεταξύ των κόμβων αποτελούν κανάλια διάδοσης πληροφοριών και αλληλεπίδρασης. Όσον αφορά τη διαφήμιση και τις πωλήσεις, οι χρήστες επηρεάζονται από τους φίλους τους και είναι αρκετά πιθανόν να αγοράσει κάποιος προϊόντα και υπηρεσίες που αγόρασαν φίλοι του. Οι φίλοι ανταλλάσσουν μεταξύ τους απόψεις και κάποιος μπορεί να συστήσει ένα προϊόν στους φίλους του. Στα δίκτυα τηλεφωνίας, λόγω προσφορών χαμηλότερων τιμών για κλήσεις εντός του δικτύου ενός παρόχου, είναι πολύ πιθανόν ένας καταναλωτής να επιλέξει το δίκτυο που χρησιμοποιούν άτομα με τα οποία σχετίζεται. Ορισμένοι κόμβοι, λόγω προσωπικής μόρφωσης, ενδιαφερόντων, έντονης δραστηριότητας στο δίκτυο κλπ. επηρεάζουν σε σημαντικότερο βαθμό άλλους κόμβους απ' ότι πιο αδρανείς κόμβοι. Ο εντοπισμός και η επιρροή των δυναμικών κόμβων επιτρέπει την ταχύτερη και αποτελεσματικότερη διάδοση ιδεών στο δίκτυο. Μια διαφημιστική εκστρατεία που στοχεύει στους δυναμικούς κόμβους, επιτυγχάνει την ταχεία διάδοση των μηνυμάτων της με μικρότερο κόστος.

Βιβλιογραφικές Αναφορές

- Belshe, M., Peon, R. & Thomson, M. (2015). Hypertext Transfer Protocol Version 2 (HTTP/2). *Request for Comments: 7540*. Internet Engineering Task Force Standards (IETF) Organization.
- Berners-Lee, T., Masinter, L. & McCahill, M. (1994). *Uniform Resource Locators (URL)*. *Request for Comments: 1738*. Internet Engineering Task Force Standards (IETF) Organization.
- Bray, T. (2014). The JavaScript Object Notation (JSON) Data Interchange Format. *Request for Comments: 7159*. Internet Engineering Task Force Standards (IETF) Organization.
- Bray, T. (2015). The I-JSON Message Format. *Request for Comments: 7493*. Internet Engineering Task Force Standards (IETF) Organization.
- Bonchi, F., Castillo, C., Gionis, A., & Jaimes, A. (2011). *Social Network Analysis and Mining for Business Applications*. ACM Transactions on Intelligent Systems and Technology, 2(3), 22-37. doi: 10.1145/1961189.1961194
- Conder, S. & Darcey, L. (2010). *Android Wireless Application Development, Developer's Library, 2nd Edition*. Addison-Wesley Professional.
- Global Industry Analyst, Inc. A Worldwide Business Strategy & Market Intelligence Source. (2016). *Mobile Messaging Apps - A Global Strategic Business Report*. (4 Jan 2016). Global Industry Analyst, Inc. 6150 Hellyer Ave., San Jose, CA 95138, USA.
- Fielding, R. & Reschke, J. (2014). *Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing*. *Request for Comments: 7230*. Internet Engineering Task Force Standards (IETF) Organization.
- Juniper Networks. (2011). *Mobile Device Security - Emerging Threats, Essential Strategies, Key Capabilities for Safeguarding Mobile Devices and Corporate Assets*. White Paper.
- La Polla, M., Martinelli, F. & Sgandurra, D. (2013). *A survey on security for mobile devices*. IEEE Communications Surveys & Tutorials, 15 (1), 446-471.
- Saint-Andre, P. (2011) (a). *Extensible Messaging and Presence Protocol (XMPP): Core*. *Request for Comments: 6120*. Internet Engineering Task Force Standards (IETF) Organization.
- Saint-Andre, P. (2011) (b). *Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence*. *Request for Comments: 6121*. Internet Engineering Task Force Standards (IETF) Organization.
- Saint-Andre, P. (2015). *Extensible Messaging and Presence Protocol (XMPP): Address Format*. *Request for Comments: 7622*. Internet Engineering Task Force Standards (IETF) Organization.
- Tanenbaum, A., S. & Wetherall, D., J. (2011). *Δίκτυα υπολογιστών: Ένατη αμερικανική έκδοση*. Σκουλαρίκης, Φ. & Ξυλωμένος, Γ. (μετάφραση). (σσ. 285-331). Αθήνα: Κλειδάριθμος, 2012.