

ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ

ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΜΜΕ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**ΑΝΑΛΥΣΗ ΤΕΧΝΙΚΩΝ
ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΤΩΝ ΕΦΑΡΜΟΓΩΝ
ANDROID ΚΑΙ ΔΗΜΙΟΥΡΓΙΑ ΕΦΑΡΜΟΓΗΣ**

ΣΠΟΥΔΑΣΤΕΣ

ΓΙΟΒΑΣ ΘΕΟΔΩΡΟΣ

ΠΟΛΑΡΑΣ ΣΩΤΗΡΙΟΣ

ΕΠΟΠΤΕΥΩΝ ΚΑΘΗΓΗΤΗΣ

ΓΚΟΤΣΙΝΑΣ ΑΝΤΩΝΙΟΣ

ΠΥΡΓΟΣ 2016

ΥΠΕΥΘΥΝΗ ΔΗΛΩΣΗ ΠΕΡΙ ΜΗ ΛΟΓΟΚΛΟΠΗΣ

Βεβαιώνω/ουμε ότι είμαι/είμαστε ο/οι συγγραφείς/εις αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα/είχαμε για την προετοιμασία της, είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία.

Επίσης, έχω/έχουμε αναφέρει τις οποίες πηγές από τις οποίες έκανα /κάναμε χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες.

Ακόμη δηλώνω/ουμε ότι αυτή η γραπτή εργασία προετοιμάστηκε από εμένα/εμάς προσωπικά και αποκλειστικά και ειδικά για την συγκεκριμένη πτυχιακή εργασία ότι θα αναλάβω/ουμε πλήρως τις συνέπειες εάν η εργασία αυτή αποδειχτεί ότι δεν μου/μας ανήκει.

ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΣΠΟΥΔΑΣΤΗ 1	ΑΡΙΘ.ΜΗΤΡΩΟΥ	ΥΠΟΓΡΑΦΗ
<u>Σοφία Σ. Σολομού</u>	<u>1560</u>	
ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΣΠΟΥΔΑΣΤΗ 2	ΑΡΙΘ.ΜΗΤΡΩΟΥ	ΥΠΟΓΡΑΦΗ
<u>Πορφύρας Σουλίου</u>	<u>7905</u>	
ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΣΠΟΥΔΑΣΤΗ 3	ΑΡΙΘ.ΜΗΤΡΩΟΥ	ΥΠΟΓΡΑΦΗ
.....

Περίληψη

Σε αυτή την πτυχιακή εργασία αναλύονται η χρήση και οι λειτουργίες της πλατφόρμας android καθώς και τεχνικές ανάπτυξης εφαρμογών για κινητά τηλέφωνα και tablets.

Το πρώτο κεφάλαιο αναφέρεται στα χαρακτηριστικά του λειτουργικού συστήματος android. Περιγράφονται ορισμένες βασικές λειτουργίες του καθώς και η έκτη και πιο πρόσφατη έκδοσή του, το android 6.0 marshmallow.

Στο δεύτερο κεφάλαιο αναλύονται περισσότερο εξειδικευμένες λειτουργίες της συγκεκριμένης πλατφόρμας και γίνεται αναφορά σε διάφορα εργαλεία τα οποία χρησιμοποιούνται για την αξιολόγηση του υλικού(hardware) και τον τρόπο λειτουργίας του. Προτείνονται επιπλέον, διάφορες τεχνικές σωστής χρήσης μίας κινητής συσκευής για την ασφάλεια και τη διαχείριση των δεδομένων.

Στο τρίτο κεφάλαιο περιγράφεται αναλυτικά η δημιουργία μίας εφαρμογής σε android με τη χρήση των εργαλείων που προσφέρει το προγραμματιστικό περιβάλλον Android Studio. Χρησιμοποιώντας τη γλώσσα προγραμματισμού JAVA υλοποιούνται ορισμένες εφαρμογές οι οποίες έχουν σκοπό να παρουσιάσουν τη χρηστικότητα και την ευελιξία της συγκεκριμένης πλατφόρμας.

Περιεχόμενα

ΚΕΦΑΛΑΙΟ 1: Εισαγωγή στο Android	7
1.1 Τι είναι το android;	8
1.2 Χαρακτηριστικά και εφαρμογές	11
1.3 Android vs Ios	11
1.4 Android Marshmallow 6.0	12
ΚΕΦΑΛΑΙΟ 2: Λειτουργίες Android	13
2.1 Πως γράφουμε μηνύματα με την φωνή μας στο Android	14
2.2 Κλείδωμα και διαγραφή χαμένου κινητού	15
2.3 Τι είναι το root και πως λειτουργεί.	16
2.4 Τι είναι τα benchmark;	20
2.4.1 Ποιο benchmark να επιλέξω;	21
2.5 Τι είναι και πως κάνω επαναφορά εργοστασιακών δεδομένων.	25
2.6 Καρφίτσωμα εφαρμογής	29
2.7 Τι είναι η ασφαλής λειτουργία.	30
2.8 Δυσκολία μεταφοράς στην κάρτα SD.	34
2.8.1 Η ιδιαιτερότητα του Galaxy S4	36
2.9 Το Android χρησιμοποιείται ακόμα και σε tablet	37
2.10 Κινητά Smartphones με λογισμικό android	37
2.11 Βελτιώσεις στο File Manager	39
2.12 Καινούργια διαθέσιμη λειτουργία για το Android	40
2.13 Το Android στον υπολογιστή	41
2.14 Πως κάνουμε εκκαθάριση δεδομένων	43
2.15 Μεταφορά επαφών, εφαρμογών και αρχείων	44
2.16 Η Google παρουσιάζει τη νέα “Bottom Navigation Bar”	47
2.17 Εναλλαγή εφαρμογών στο Android N	50

2.18 Rom Windows 10 για Android συσκευές	50
2.19 Τι είναι ο Bootloader;	51
2.20 Τι είναι και πως κάνω S-OFF	52
2.21 Αλλαγή τα χρωμάτων για το Android 6.0	54
2.22 Χειριστείτε την μουσική σας μέσα από το Google Music	56
2.23 Έκδοση Android για υπολογιστές Remix OS	58
ΚΕΦΑΛΑΙΟ 3: Υλοποίηση σε Android	64
3.1 Πώς λειτουργεί μία εφαρμογή android;	65
3.2 Δημιουργία της εφαρμογής DatePicker	74
3.2.1 Ανάλυση απαιτήσεων και σχεδιασμός	74
3.2.2 Δημιουργία project με το Android Studio	75
3.2.3 Ολοκλήρωση του setup της εφαρμογής	77
3.2.4 Έναρξη συγγραφής κώδικα	78
3.3 Δημιουργία εφαρμογής clickable_button	80
3.3.1 Ανάλυση απαιτήσεων	80
3.3.2 Συγγραφή κώδικα	81
3.4 Δημιουργία εφαρμογής list_of_images	84
3.4.1 Ανάλυση απαιτήσεων εφαρμογής	84
3.4.2 Συγγραφή κώδικα	85
3.5 Εφαρμογή αποστολής SMS	89
3.5.1 Συγγραφή κώδικα	90
3.5.2 Ανάλυση εφαρμογής	90
3.6 Εφαρμογή Checkbox	91
3.6.1 Ανάλυση εφαρμογής	91
3.6.2 Συγγραφή κώδικα	91

ΚΕΦΑΛΑΙΟ 4 Δημιουργία εφαρμογής σε Android	94
4.1 Δημιουργία εφαρμογής Quiz_Game	95
4.2 Ανάλυση απαιτήσεων και σχεδιασμός	95
4.3 Συγγραφή κώδικα	96
4.4 Τελικός κώδικας εφαρμογής	101

ΚΕΦΑΛΑΙΟ 1

Εισαγωγή στο android

1.1 Τι είναι το android;

Το android είναι λειτουργικό σύστημα για συσκευές κινητής τηλεφωνίας το οποίο τρέχει τον πυρήνα του λειτουργικού Linux. Αρχικά αναπτύχθηκε από την Google και αργότερα από την Open Handset Alliance. Επιτρέπει στους κατασκευαστές λογισμικού να συνθέτουν κώδικα με την χρήση της γλώσσας προγραμματισμού Java ελέγχοντας την συσκευή μέσω βιβλιοθηκών λογισμικού ανεπτυγμένων από την Google. Το Android είναι κατά κύριο λόγο σχεδιασμένο για συσκευές με οθόνη αφής, όπως τα έξυπνα τηλέφωνα (smartphones) και τα τάμπλετ, με διαφορετικό περιβάλλον χρήσης για τηλεοράσεις (Android Tv), αυτοκίνητα (Android Auto) και ρολόγια (Android Wear).

Το Android είναι το πιο ευρέως διαδεδομένο λογισμικό στον κόσμο. Οι συσκευές με Android έχουν περισσότερες πωλήσεις από όλες τις συσκευές Windows, iOS και Mac OS X μαζί.

Η πρώτη παρουσίαση του Android έγινε στις 5 Νοεμβρίου 2007, παράλληλα με την ανακοίνωση της ίδρυσης του οργανισμού Open Handset Alliance, μιας κοινοπραξίας 48 τηλεπικοινωνιακών εταιριών, εταιριών λογισμικού καθώς και κατασκευής hardware, οι οποίες είναι αφιερωμένες στην ανάπτυξη και εξέλιξη ανοιχτών προτύπων στις συσκευές κινητής τηλεφωνίας. Η Google δημοσίευσε το μεγαλύτερο μέρος του κώδικα του Android υπό τους όρους της Apache License, μιας ελεύθερης άδειας λογισμικού. Το λογότυπο για το λειτουργικό σύστημα Android είναι ένα ρομπότ σε χρώμα πράσινου μήλου και σχεδιάστηκε από την γραφίστρια Ιρίνα Μπλόκ.

1. Open Handset Alliance (5 Νοεμβρίου 2007). Industry Leaders Announce Open Platform for Mobile Devices. Δελτίο τύπου. Ανακτήθηκε στις 5 Νοεμβρίου 2007.

Τα χαρακτηριστικά του Android :

Λειτουργίες Οθόνης : Η πλατφόρμα είναι προσαρμόσιμη σε πολλές αναλύσεις οθόνης (από VGA μέχρι 4K), δισδιάστατες ψηφιακές γραφικές βιβλιοθήκες, τρισδιάστατα γραφικά βασισμένα στην OpenGL ES 3.0+ έκδοση χαρακτηριστικών, καθώς και παραδοσιακές απεικονίσεις “έξυπνων” συσκευών τηλεφωνίας.

Αποθήκευση Δεδομένων : Χρήση βάσης δεδομένων SQLite για τις ανάγκες αποθήκευσης.

Συνδεσιμότητα : Το Android υποστηρίζει τεχνολογίες συνδεσιμότητας συμπεριλαμβανομένου GSM/EDGE, 3G, 4G, CDMA, EVDO, UMTS, Bluetooth, NFC και WiFi.

Αποστολή μηνυμάτων : SMS και MMS είναι οι διαθέσιμοι τρόποι ανταλλαγής μηνυμάτων.

Περιήγηση στον Ιστό : Για την περιήγηση στον ιστό το Android διαθέτει φυλλομετρητή βασισμένο στην ανοιχτή τεχνολογία WebKit. Και άλλοι φυλλομετρητές είναι διαθέσιμοι από το Google play.

Υποστήριξη Java : Λογισμικό γραμμένο στην Java είναι δυνατόν να μεταγλωττιστεί και να εκτελεστεί στην εικονική μηχανή Dalvik, η οποία αποτελεί εξειδικευμένη υλοποίηση εικονικής μηχανής, σχεδιασμένης για χρήση σε φορητές συσκευές. παρόλο που δεν είναι πρότυπη εικονική μηχανή Java.

Υποστήριξη Πολυμέσων : Το λειτουργικό Android υποστηρίζει τις παρακάτω μορφές ήχου, στατικής και κινούμενης εικόνας: H.263, H.264 (σε 3GP ή MP4 container), MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC, MP3, MIDI, OGG, Vorbis, WAV, JPEG, PNG, GIF, BMP. Επιπλέον υποστηρίζει υλικού. Το λειτουργικό Android

μπορεί να συνεργαστεί με κάμερες στατικής ή κινούμενης εικόνας, οθόνες αφής GPS, αισθητήρες επιτάχυνσης, μαγνητόμετρα, δισδιάστατους καθώς και τρισδιάστατους επιταχυντές γραφικών.

Περιβάλλον Ανάπτυξης Λογισμικού : Περιλαμβάνει ένα προσομοιωτή συσκευής, εργαλεία για διόρθωση σφαλμάτων, μνήμη και εργαλεία ανάλυσης της απόδοσης του εκτελέσιμου λογισμικού καθώς και ένα επιπρόσθετο για το Eclipse IDE.

Αγορά και Εγκατάσταση Εφαρμογών : Παρόμοια με το App Store του iPhone OS, το Google play είναι ένας κατάλογος εφαρμογών που μπορούν να μεταφορτωθούν και εγκατασταθούν στην συσκευή άμεσα μέσω ασύρματων καναλιών, χωρίς την χρήση υπολογιστή. Αρχικά μόνο δωρεάν εφαρμογές ήταν δυνατόν να εγκατασταθούν. Εφαρμογές επί πληρωμή ήταν μετέπειτα διαθέσιμες στο Google play στις ΗΠΑ ύστερα από τις 19 Φεβρουαρίου 2009.

Οθόνη Αφής Πολλαπλών Σημείων : Το λειτουργικό Android υποστηρίζει οθόνες αφής πολλαπλών σημείων αλλά η δυνατότητα αυτή είχε κλειδωθεί σε επίπεδο πυρήνα (πιθανόν για αποφυγή παραβιάσεων των πατεντών λογισμικού της Apple στις τεχνολογίες οθονών αφής). Κυκλοφορούσε μια ανεπίσημη τροποποίηση (mod) που έχει αναπτυχθεί για να υποστηρίζει πολλαπλή επαφή (multi-touch), αλλά απαιτούσε δικαιώμα πρόσβασης (superuser) στη συσκευή για να γραφτεί στην μνήμη flash ένας πυρήνας που να μην είναι υπογεγραμμένος (unsigned kernel). Παρόλο που το Android είναι ένα προϊόν ελεύθερου λογισμικού, ένα κομμάτι ανάπτυξης του λογισμικού είναι σε ιδιωτικό παρακλάδι. Για να έρθει αυτό το λογισμικό σε κοινή θέαση δημιουργήθηκε ένα παρακλάδι του μόνο ανάγνωσης, εν ονόματι "Cupcake". Το Cupcake συνήθως συγχέεται με τον τίτλο μιας ενημέρωσης, σε αντίθεση με όσα δηλώνει η ίδια η Google στην ιστοσελίδα

ανάπτυξης του Android: “το Cupcake αποτελεί ακόμη ένα έργο σε εξέλιξη,όχι μια επίσημη έκδοση.”Αξιοσημείωτες αλλαγές στο λειτουργικό Android θα παρουσιαστούν στο cupcake και περιλαμβάνουν αλλαγές στο σύστημα διαχείρισης μεταφορτώσεων (download manager),το framework,Bluetooth,το λογισμικό συστήματος, το ραδιόφωνο και το σύστημα τηλεφωνίας, εργαλεία προγραμματισμού, το κυρίως σύστημα και διάφορες εφαρμογές, καθώς και πληθώρα διορθώσεις σφαλμάτων.

Στις 30 Απριλίου 2009,κυκλοφόρησε η επίσημη ενημέρωση έκδοσης 1.5 για το Android.Αποτελείται από πολλά νέα χαρακτηριστικά και βελτιώσεις στο γραφικό περιβάλλον :

Δυνατότητα καταγραφής κινούμενης εικόνας με την χρήση της αντίστοιχης λειτουργίας του τηλεφώνου.

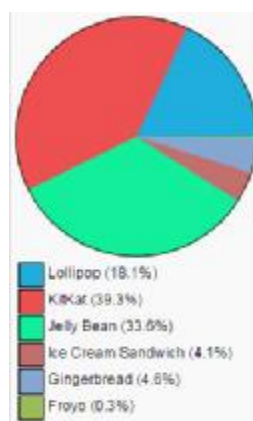
Μεταφόρτωση αρχείων βίντεο στο YouTube και εικόνων στο Picasa κατευθείαν από το τηλέφωνο.

Επανασχεδιασμένο λογισμικό πληκτρολογίου με λειτουργία αυτόματης συμπλήρωσης κειμένου.

Δυνατότητα αυτόματης σύνδεσης ασύρματης συσκευής ακουστικού Bluetooth εφόσον εντοπιστεί σε μια συγκεκριμένη απόσταση.

Νέα widgets και φάκελοι που μπορούν να τοποθετηθούν στην επιφάνεια εργασίας.

Εφέ αλλαγής οθονών και μενού.



1

¹ «Open Handset Alliance». Open Handset Alliance. Ανακτήθηκε στις 6 Νοεμβρίου 2007.
Jackson, Rob (10 Δεκεμβρίου 2008). «Sony Ericsson, HTC Androids Set For Summer 2009». Android Phone Fans. Ανακτήθηκε στις 3 Σεπτεμβρίου 2009.

1.2 χαρακτηριστικά και εφαρμογές

Οι οθόνες στα smartphones ποικίλλουν σε μεγάλο βαθμό τόσο σε μέγεθος όσο και σε ανάλυση. Τα πιο συνηθισμένα μεγέθη οθόνης κυμαίνονται από 3 ίντσες ως πάνω από 5 ίντσες(διαγώνια μέτρηση). Ορισμένες συσκευές με οθόνη 6-8 ιντσών τρέχουν σε λειτουργικά συστήματα κινητών τηλεφώνων και έχουν την ικανότητα να πραγματοποιούν τηλεφωνικές κλήσεις, όπως το Huawei Ascend Mate, το Asus Fonepad και το Samsung Galaxy Note 8.0. Έχουν ακουστεί ορισμένα επιχειρήματα σχετικά με την εργονομία, ότι η αύξηση των μεγεθών της οθόνης αρχίζει να επηρεάζει αρνητικά τη χρηστικότητα. Οι συχνότερες αναλύσεις για οθόνες smartphone ποικίλλουν από 300x240px(QVGA) έως 1920x1080(Full HD), με ναυαρχίδα τα τηλέφωνα Android, τα οποία έχουν συνήθως ανάλυση ίση ή μεγαλύτερη από 1280x720px (HD) και το iPhone 5 με 1136x640px.

Από τα τέλη του 2012 και τις αρχές του 2013, έχει προκύψει στο χώρο της βιομηχανίας μια τάση μετακίνησης προς τις οθόνες smartphone με πλήρη Full HD χαρακτηριστικά για τις high-end συσκευές.

Με το πέρασμα των ετών, η πυκνότητα των pixels στις οθόνες των smartphones γίνεται όλο και μεγαλύτερη. Η Apple έλαβα το αναγνωρισμένο σήμα για την οθόνη υψηλής πυκνότητας με την επωνυμία Retina. Η ανάλυση υψηλής πυκνότητας βολεύει για μικρούς, περίπλοκους χαρακτήρες (κυρίως για κείμενα ανατολικής Ασίας) και αιχμηρές άκρες (ιδίως γραμμικό σχέδιο και γραμματοσειρές Aliased).²

1.3 Android vs iOS

Το Android και το iOS είναι και τα δύο λειτουργικά συστήματα για κινητά τηλέφωνα και ταμπλέτες. Το Android τρέχει σε μια ποικιλία των τηλεφώνων και tablets, ενώ το iOS τρέχει αποκλειστικά για την Apple iPhones και iPads.

Το Android είναι ένα έργο ανοικτού κώδικα με επικεφαλής αρχικά από την Google, και τώρα από την Open Handset Alliance. Το iOS είναι ιδιόκτητο λογισμικό που αναπτύχθηκε από την Apple, αν και είναι χτισμένο στην κορυφή του μερικά συστατικά ανοικτού κώδικα όπως το WebKit. Το Android έχει ένα ανοικτό πλαίσιο που δεν θα επιτρέψει σε κανέναν να τροποποιήσει το ίδιο το λειτουργικό σύστημα και την ανάπτυξη εφαρμογών σε αυτό. Η Apple παρένει ένα SDK για iOS που επιτρέπει στους προγραμματιστές να έχουν πρόσβαση σε περισσότερα από τα χαρακτηριστικά iOS, αλλά είναι εξαιρετικά περιορισμένη. Τα προγράμματα που είναι γραμμένα για το Android συνήθως σε Java, αν και άλλες γλώσσες τρέξει επίσης από την JVM και

² «Smartphone» Phone Scoop. Ανακτήθηκε στις 2011-12-15.

«Feature Phone». Phone Scoop. Ανακτήθηκε στις 2011-12-15.

Andrew Nusca (20 August 2009). «Smartphone vs. feature phone arms race heats up; which did you buy?». ZDNet. Ανακτήθηκε στις 2011-12-15.

υποδοχής CPU. Όλες οι αιτήσεις που απελευθερώνονται μέσω των iPhone και iPad App Stores πρέπει να είναι γραμμένα σε C, C++, Objective-C ή Swift. Φημολογείται οι Dalvik εικονική μηχανή του Android μπορεί τελικά να επιστρέψει τη χρήση και με άλλες γλώσσες bytecode, όπως Python και Ruby.³

1.4 Android Marshmallow 6.0

Έφτασε λοιπόν και το Android 6.0, το οποίο μοιάζει περισσότερο με Android 5.2 εδώ που τα λέμε, το κατεβάσαμε, το εγκαταστήσαμε και τώρα που "είναι ζεστά τα κουλούρια".

Εκ πρώτης όψεως είναι ίσως η πιο σταθερή έκδοση μέχρι σήμερα, χωρίς τα κολλήματα και τις παθογένειες του Android 5.0, αλλά δεν είναι το ελαφρύτερο από την έκδοση 4.4. Αναμενόμενο αυτό καθώς τα πολλά animation βαραίνουν το σύστημα, αλλά πλέον οι περισσότερες συσκευές που κυκλοφορούν σε όλες τις κατηγορίες, έχουν τις δυνατότητες και τη RAM να το σηκώσουν.⁴

³ Why the Android Platform Will Beat the iPhone
Android versus iPhone 3.0: The Showdown

⁴ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Μαΐου, 2016, από <http://www.doctorandroid.gr/2015/10/android-60-marshmallow-overview.html>.

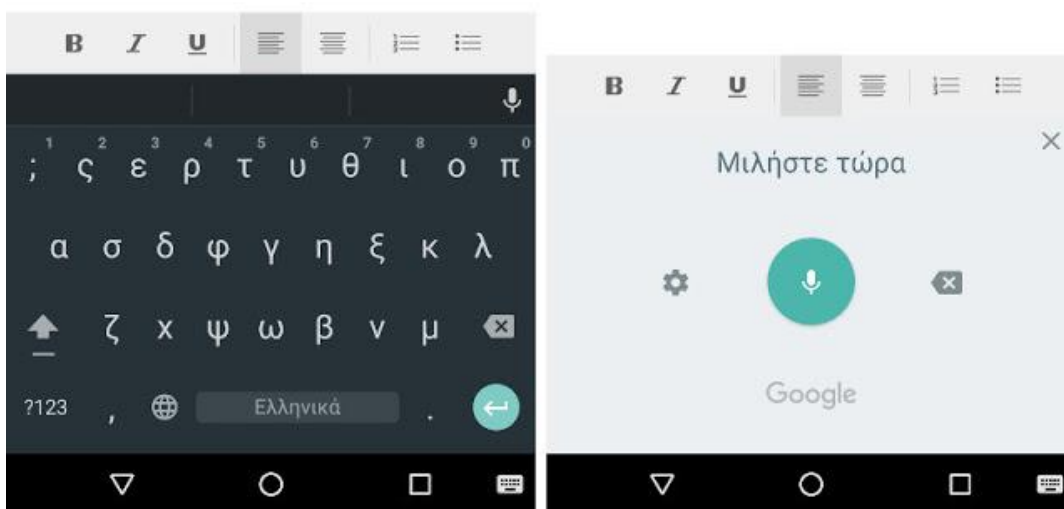
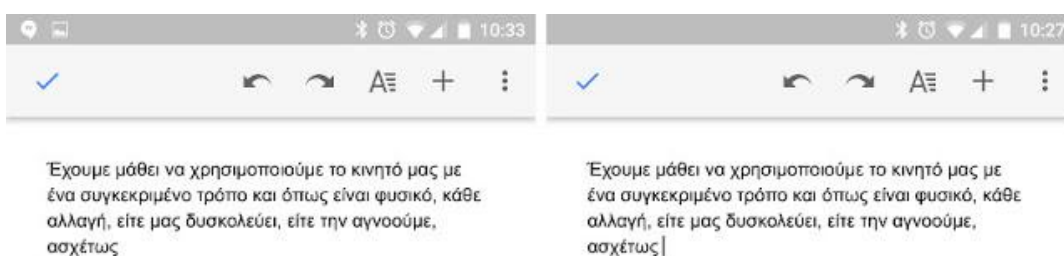
ΚΕΦΑΛΑΙΟ 2

Λειτουργίες Android

2.1 Πως γράφουμε μηνύματα με την φωνή μας στο Android

Έχουμε μάθει να χρησιμοποιούμε το κινητό μας με ένα συγκεκριμένο τρόπο και όπως είναι φυσικό, κάθε αλλαγή, είτε μας δυσκολεύει, είτε την αγνοούμε, ασχέτως αν στο τέλος μπορεί να μας εξυπηρετήσει καλύτερα. Ένα από αυτά είναι η πληκτρολόγηση μηνυμάτων, η οποία γίνεται φυσικά με τα χέρια, αλλά το Android εδώ και καιρό μας δίνει τη δυνατότητα να πληκτρολογήσουμε κείμενο μόνο με τη φωνή μας.

Με τη φωνητική εισαγωγή του πληκτρολογίου Google αλλά και άλλων όπως το Swype για παράδειγμα, μπορούμε να γράψουμε κείμενα με τη φωνή μας, σε οποιαδήποτε εφαρμογή και χωρίς περιορισμό στο μήκος. Απλά ανοίγουμε την εφαρμογή που θέλουμε και αντί να πληκτρολογήσουμε, πατάμε το μικρόφωνο πάνω δεξιά στο πληκτρολόγιο και εκφωνούμε το κείμενο που θέλουμε.



Η εκφώνηση θα πρέπει να είναι καθαρή και με άρθρωση, αν θέλουμε να γραφτούν όλα σωστά και επειδή το χρησιμοποιούμε αρκετά συχνά, προτείνουμε την εκφώνηση μίας μίας πρότασης, για να ελέγχουμε και να διορθώνουμε στη συνέχεια τυχόν λάθη. Κάποιες λέξεις που δεν περιλαμβάνονται στο λεξικό της Google, μπορούμε να τις προσθέσουμε στο προσωπικό μας λεξικό και μετά από λίγο διάστημα θα μπορούμε να τις εκφωνήσουμε και αυτές (προφανώς συγχρονίζονται με το server της Google και προστίθενται μετά από καιρό).

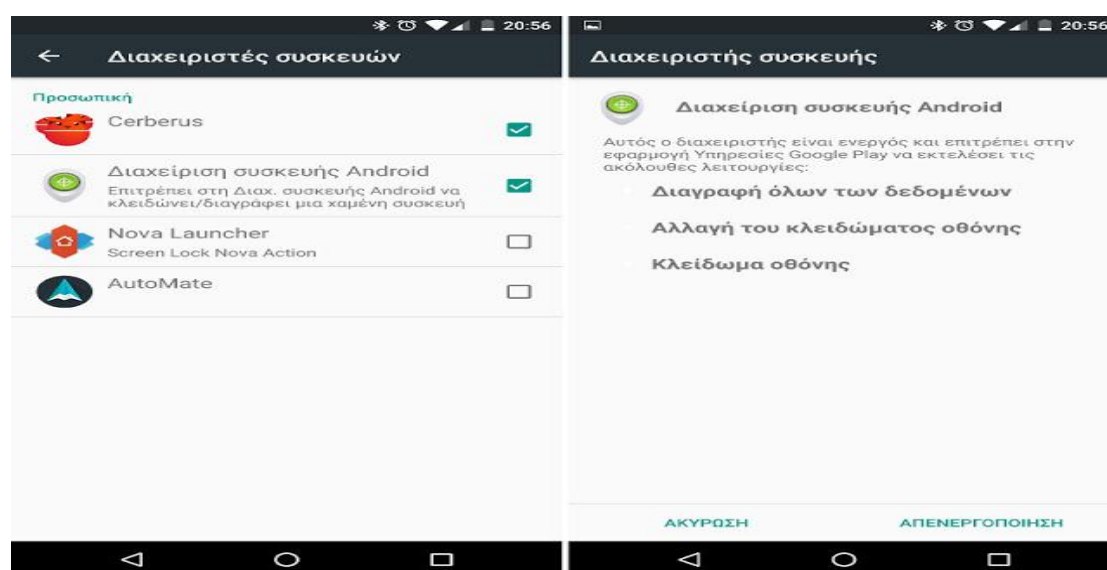
Τέλος μπορούμε να χρησιμοποιήσουμε τη φωνητική πληκτρολόγηση, όπου υπάρχει

το εικονίδιο του μικροφώνου, μεταξύ των οποίων, στην αναζήτηση Google, στους Χάρτες, σε εφαρμογές SMS και εφαρμογές Office. Πολύ εξυπηρετικό ειδικά αν θέλεις να αντιγράψεις ένα κείμενο σε δικό σου αρχείο, ενώ μπορεί να χρησιμοποιηθεί και σαν OCR, το κείμενο από ένα pdf σε ένα text αρχείο.⁵

2.2 Κλείδωμα και διαγραφή χαμένου κινητού

Το Android έχει ενσωματωμένες ένα σωρό υπηρεσίες, οι οποίες βρίσκονται εκεί για να κάνουν την ζωή μας ευκολότερη, αλλά και για να μας προστατεύσουν. Μία από αυτές είναι ο Διαχειριστής Android Συσκευής, ο οποίος ανήκει στη δεύτερη κατηγορία και μας επιτρέπει να προστατεύσουμε τα δεδομένα μας σε περίπτωση απώλειας.

Ο Διαχειριστής βρίσκεται ενσωματωμένος σε όλες τις Android συσκευές και μπορούμε να τον ενεργοποιήσουμε πηγαίνοντας στη διαδρομή “Ρυθμίσεις>Ασφάλεια>Διαχειριστές συσκευών”. Η δουλειά του είναι να εντοπίζει τη συσκευή μας όπου και αν βρίσκεται και μας επιτρέπει να την καλέσουμε, να την κλειδώσουμε ή να διαγράψουμε τα δεδομένα μας από αυτή. Είναι γενικά καλό να τον ενεργοποιούμε αμέσως μόλις αγοράζουμε τη συσκευή μας, ή αν έχουμε κάνει πρόσφατα επαναφορά εργοστασιακών δεδομένων, καθώς θα μας φανεί ιδιαίτερα χρήσιμος αν είμαστε απρόσεκτοι.



Ας πούμε λοιπόν ότι δεν ξέρουμε που είναι η συσκευή μας. Μπορούμε να τη βρούμε ως εξής:

Μέσω μιας άλλης συσκευής (υπολογιστή ή smartphone) μπαίνουμε στην επίσημη σελίδα του διαχειριστή είτε από τον browser, είτε κατεβάζοντας την ομώνυμη

⁵ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Μαΐου, 2016, από <http://www.doctorandroid.gr/2016/03/how-to-voice-typing.html>.

εφαρμογή. Συνδεόμαστε με τον Google λογαριασμό που βρίσκεται στο χαμένο τηλέφωνο και το επιλέγουμε για να δούμε την τοποθεσία του. Επιλέγουμε “Κλήση”

αν βρίσκεται για παράδειγμα κάπου μέσα στο σπίτι και θέλουμε να το βρούμε, “Κλείδωμα” αν το έχουμε χάσει και θέλουμε να το κλειδώσουμε ή “Διαγραφή” για να κάνουμε αποκρυμμένη επαναφορά εργοστασιακών δεδομένων, έτσι ώστε να μη δει ή χρησιμοποιήσει κάποιος ξένος τα προσωπικά μας στοιχεία.

Σημείωση: Ο διαχειριστής δεν μπορεί να συνδεθεί στην συσκευή μας αν δεν υπάρχει σύνδεση στο διαδίκτυο, για αυτό τον λόγο αφήνουμε πάντα τα δεδομένα κινητής τηλεφωνίας ανοιχτά. Επίσης δεν μπορεί να βρει την τοποθεσία μας αν έχουμε απενεργοποιήσει, οπότε χρησιμοποιούμε μόνιμα τη ρύθμιση “Εξοικονόμηση Μπαταρίας”.⁶

2.3 Τι είναι το root και πως λειτουργεί.

Το root είναι ένας όρος που υπάρχει πολύ πριν γεννηθεί το Android και στην πραγματικότητα έχει περισσότερες από μία έννοιες, οι οποίες μπερδεύουν ακόμη και σήμερα τους πολλούς από εμάς. Διάβαζα πρόσφατα το "επεξηγηματικό" άρθρο για το root που γράψαμε κάποτε και κατάλαβα ότι δεν εξηγεί απολύτως τίποτα στον αρχάριο Android χρήστη. Έτσι λοιπόν σήμερα πιάνουμε ξανά την κουβέντα για το root, σωστά και με παραδείγματα που θα καταλάβει και ο πιο άσχετος με το Android.

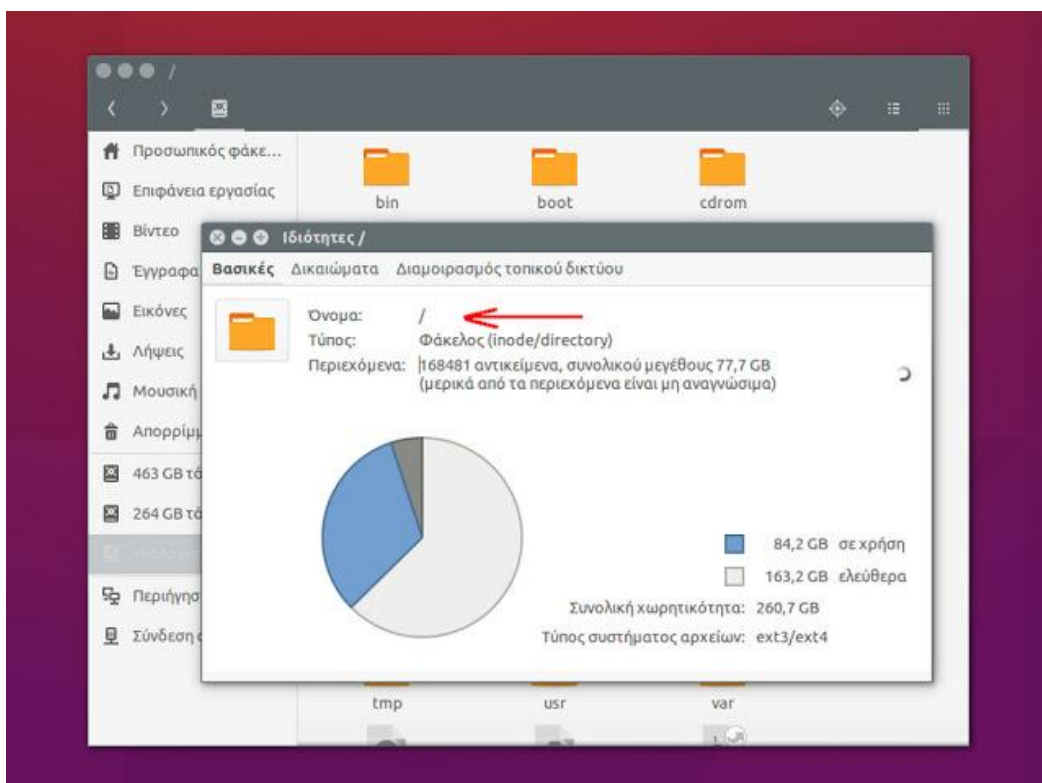
Το root ως γενικός όρος

Root σημαίνει ρίζα στα ελληνικά και ο όρος δεν είναι καθόλου τυχαίος, καθώς χρησιμοποιήθηκε για πρώτη φορά για να δοθεί μία ονομασία στον αρχικό (ριζικό) φάκελο ενός δίσκου αποθήκευσης. Ριζικός είναι ο πρώτος φάκελος που μπαίνουμε όταν ανοίγουμε ένα σκληρό δίσκο, ένα στικάκι, ένα CD / DVD και γενικά ένα μέσο αποθήκευσης και δε διαγράφεται καθώς δεν περιέχεται σε κάποιον άλλο φάκελο. Για να το καταλάβετε καλύτερα, βάλτε ένα στικάκι USB στον υπολογιστή σας και ανοίξτε το. Θα δείτε μέσα τους φακέλους και τα αρχεία που έχετε αποθηκεύσει, τα οποία βρίσκονται όλα χύμα μέσα στο στικάκι, σωστά; Αυτό το "χύμα" είναι ο φάκελος root. Ας πούμε επίσης ότι μόλις έχουμε αγοράσει ένα στικάκι και το βάζουμε για πρώτη φορά στον υπολογιστή. Αυτό που θα δούμε ανοίγοντάς το είναι ένας κενός φάκελος. Αυτό και πάλι είναι ο root φάκελος. Σε όποιον φάκελο και να μπούμε μέσα σε ένα δίσκο, μπορούμε να βγούμε από αυτόν πηγαίνοντας προς τα πίσω. Όταν φτάσουμε στο φάκελο root που είναι ο τελευταίος και περιέχει όλους τους υπόλοιπους, απλά βγαίνουμε από τον δίσκο.

Κάθε φάκελος πρέπει να έχει μία ονομασία για να μπορεί ο υπολογιστής μας να τον βρει και τον βρίσκει ακολουθώντας τη διαδρομή που έχει αποθηκευτεί, από έξω προς τα μέσα. Για παράδειγμα ας πούμε ότι έχω έναν φάκελο με το όνομα "boot", όπου μέσα του έχω δημιουργήσει το φάκελο "grub" και μέσα σε αυτόν το φάκελο "fonts".

⁶ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/02/android-device-manager.html>.

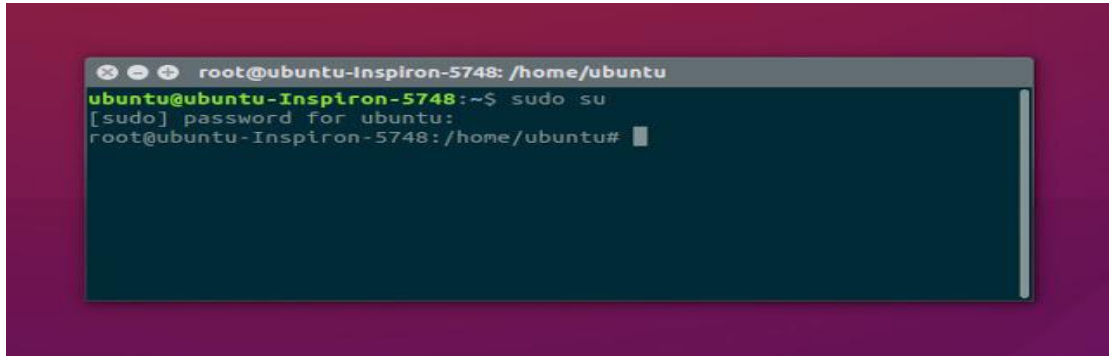
Για την υπολογιστή μου ο "fonts" βρίσκεται στη διαδρομή "/boot/grub/fonts". Ο πρώτος φάκελος που δημιούργησα ("boot") βρίσκεται λογικά στο αρχικό φάκελο root, ο οποίος όπως βλέπουμε στη διαδρομή δεν έχει καμία ονομασία, αλλά εμφανίζεται πάντα με το σύμβολο "/".



Το root στο Linux

Ο όρος root όπως τον γνωρίζουμε σήμερα ήρθε από το Linux, στο οποίο βασίζεται και το Android. Το Linux λοιπόν όπως και το Android, επιτρέπει στους χρήστες να

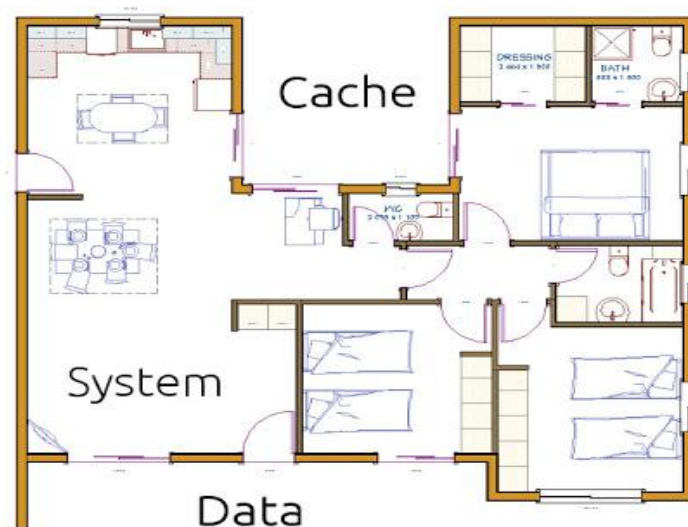
έχουν πρόσβαση μόνο σε συγκεκριμένους φακέλους και αυτό για να μην μολύνεται το σύστημα από ιούς που ενδεχομένως θα κατεβάσει ο χρήστης χρησιμοποιώντας τον υπολογιστή του. Για να αποκτήσουμε πρόσβαση στους υπόλοιπους φακέλους, πρέπει να έχεις πρώτα πρόσβαση στο root φάκελο, τον πρώτο φάκελο δηλαδή, που περιλαμβάνει όλους τους υπόλοιπους. Στο Linux το δικαίωμα πρόσβασης στο root φάκελο είναι προεγκατεστημένο και το αποκτάς καλώντας το εργαλείο "su", ανοίγοντας το τερματικό και δίνοντας την εντολή : sudo su.



Με το "sudo" λέμε στο σύστημα ότι θέλουμε να εκτελέσουμε μία εργασία ως root χρήστης και η εργασία που θέλουμε να κάνουμε είναι το "su", να γίνουμε δηλαδή ο root χρήστης. Το σύστημα ζητάει τον κωδικό μας, ελέγχει αν έχουμε τέτοιο δικαίωμα και τέλος μας δίνει την πρόσβαση. Προσέξτε στην εικόνα, όπου στην πρώτη εντολή είμαι ο χρήστης "ubuntu", ενώ στην τελευταία ο χρήστης "root".

Το root στο Android.

Στο Android λοιπόν όταν λέμε "θα κάνω root", εννοούμε ότι θα αποκτήσουμε πρόσβαση στο root φάκελο και αυτό γίνεται προσθέτοντας το su το οποίο λείπει από το Android και εγκαθιστώντας μία εφαρμογή σαν το SuperSU, η οποία λειτουργεί αντί για password. Κάνουμε root κυρίως για να επεμβούμε στον φάκελο system που περιέχει την ROM της συσκευής για να προσθέσουμε ή να αφαιρέσουμε τα πράγματα που θέλουμε σε αυτήν.



Σκεφτείτε τη συσκευή μας σαν μία μονοκατοικία στην οποία έχουμε κλειδωθεί απ' έξω. Μπορούμε να πάμε σε κάποιους χώρους όπως τη βεράντα (cache partition) και την αυλή (data partition), αλλά για να μπούμε στο σπίτι χρειαζόμαστε ένα κλειδί. Το κλειδί είναι η μέθοδος root που θα ακολουθήσουμε και είναι φτιαγμένο για μία συγκεκριμένη κλειδαριά. Η κλειδαριά είναι το κενό ασφαλείας (exploit) που εκμεταλλεύεται η συγκεκριμένη μέθοδος και μπορεί να μη λειτουργεί μετά από λίγο καιρό γιατί ο ιδιοκτήτης (Google) μας έχει πάρει χαμπάρι και έχει αλλάξει την πόρτα(OTA update).



Μόλις ξεκλειδώσουμε την πόρτα μπορούμε πλέον να πάμε σε όλα τα δωμάτια, αλλά μας ενδιαφέρει κυρίως το σαλόνι (system partition) όπου βρίσκονται τα πιο ενδιαφέροντα πράγματα. Μπορούμε να πετάξουμε ότι θέλουμε (bloatware), να βάψουμε τους τοίχους (theme), ακόμη και να το γκρεμίσουμε για να το φτιάξουμε όπως ακριβώς μας αρέσει (custom ROM), υπάρχει όμως ένα πρόβλημα.

Ο ιδιοκτήτης του σπιτιού (Google) έχει κάνει συμφωνία με τον εργολάβο (π.χ. Samsung) να έρχεται κάθε χρόνο και να βελτιώνει το σπίτι (update), αλλά αν έχουμε κάνει εμείς κάποιες αλλαγές, ο εργολάβος δεν το αναλαμβάνει (σφάλμα εγκατάστασης update). Αυτό συμβαίνει γιατί ο εργολάβος δε γνωρίζει πλέον τι αλλαγές έχουμε κάνει στη στατικότητα του κτιρίου και αν επέμβει μπορεί αυτό να γκρεμιστεί τελείως (bootloop), κάτι για το οποίο δεν έχει δώσει τη συγκατάθεσή του (απώλεια εγγύησης). Έτσι λοιπόν εμείς πρέπει είτε να επαναφέρουμε το σπίτι την αρχική του κατάσταση(εγκατάσταση stock ROM).

Είτε να το αναβαθμίζουμε και να το προσέχουμε μόνοι μας.*Εν κατακλείδι* Το root λοιπόν είναι μία ευρεία έννοια που έχει πολλές σημασίες, αλλά πάντα λίγο έως πολύ σημαίνει πρόσβαση στο σύνολο του αποθηκευτικού χώρου μια συσκευής. Κάνοντας root στο Android γινόμαστε κλειδοκράτορες της συσκευής μας και αυτό δεν είναι πάντα καλό γιατί αν ένας ιός μπει στο κινητό μας και αποκτήσει τέτοιου είδους δικαιώματα, μπορεί να αχρηστεύσει τελείως τη συσκευή μας και ακόμα χειρότερα να μας παρακολουθεί και να κλέβει τα δεδομένα μας χωρίς να το καταλαβαίνουμε.⁷

2.4 Τι είναι τα benchmark ;

⁷ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/01/android-basics-root.html>.



Όταν ψάχνουμε για να αγοράσουμε το επόμενο μας τηλέφωνο, το καλύτερο πράγμα που μπορούμε να κάνουμε είναι να πάμε σε ένα κατάστημα για να το δούμε από κοντά. Έτσι όμως ελέγχουμε μόνο την ποιότητα κατασκευής και όχι τα χαρακτηριστικά του, που είναι στο κάτω κάτω το πιο σημαντικό από όλα. Υπάρχει επίσης η περίπτωση μια συσκευή να πωλείται μόνο online (κινέζικα), πως λοιπόν μπορούμε να ελέγξουμε τις δυνατότητες της; Για αυτό τον σκοπό έχουν δημιουργηθεί τα benchmark τα οποία αν και δε λένε πάντα την αλήθεια, μπορούμε να βγάλουμε κάποια πολύ χρήσιμα συμπεράσματα για τις συσκευές μας.

Το benchmark είναι ένα εργαλείο που εκτελεί stress test σε συγκεκριμένα στοιχεία της συσκευής, δίνοντάς μας τη δυνατότητα να καταλάβουμε τα όριά της. Για παράδειγμα υπάρχουν benchmark που τσεκάρουν το μέγιστο φορτίο του επεξεργαστή, τις δυνατότητες του chip γραφικών, την ταχύτητα εγγραφής της εσωτερικής μνήμης και άλλα πολλά τα οποία συνθέτουν αυτό που λέμε "τεχνικά χαρακτηριστικά". Επειδή τα αποτελέσματά τους βγαίνουν σε διάφορες μονάδες μέτρησης που πολύς κόσμος δεν μπορεί να καταλάβει (GHz, Mflops, MB/s), στις περισσότερες περιπτώσεις έχει υιοθετηθεί το αριθμητικό μοντέλο. Βλέπουμε δηλαδή ένα νούμερο που μπορούμε να κατανοήσουμε και να συγκρίνουμε ευκολότερα με άλλες συσκευές, για να έχουμε μία καλύτερη τελική άποψη.



Οι προϋποθέσεις

Για να βγάλουμε ασφαλή συμπεράσματα από τα benchmark, πρέπει να τα τρέξουμε στις κατάλληλες συνθήκες, επειδή όμως σίγουρα θα θέλουμε να συγκρίνουμε τη

συσκευή μας με κάποια άλλη, οι συνθήκες πρέπει να είναι και ίδιες ανά συσκευή. Η κάθε συσκευή λοιπόν πρέπει να έχει περίπου τον ίδιο αριθμό εφαρμογών και διεργασιών που τρέχουν στο παρασκήνιο, να είναι συνδεδεμένη στο ίδιο WiFi, να τεστάρουμε τον ίδιο browser και λοιπά. Ακόμη και η έκδοση του Android και του kernel επηρεάζει ένα benchmark κι έτσι τα αποτελέσματα θα είναι διαφορετικά αν αλλάξουμε ROM, ενώ θα πρέπει να γνωρίζουμε επίσης ότι αν εκτελέσουμε πολλές φορές το ίδιο benchmark με τις ίδιες συνθήκες, αυτό δε θα μας δώσει ποτέ το ίδιο αποτέλεσμα στη συσκευή. Η απόκλιση συνήθως δεν είναι πολύ μεγάλη και έτσι βγάζουμε τα συμπεράσματά μας με σιγουριά. Δεν κατεβάζουμε επιπλέον εφαρμογές από τις προεγκατεστημένες, τις συνδέουμε στο ίδιο WiFi και δεν προσθέτουμε λογαριασμούς Google και κάρτες SIM. Τέλος τρέχουμε τα benchmark τρεις φορές και κρατάμε το καλύτερο αποτέλεσμα. Δεν είναι απαραίτητα ο σωστός τρόπος, αλλά είναι ο ίδιος για όλες τις συσκευές και αυτό είναι το σημαντικότερο αν θέλουμε στην συνέχεια να τις συγκρίνουμε.

2.4.1 Ποιο benchmark να επιλέξω;

Τα benchmark είναι πολλά και υπάρχουν για όλα τα λειτουργικά συστήματα. *Antutu*

Πρώτο από όλα είναι φυσικά το Antutu που είναι το πιο γνωστό και μας δίνει τη δυνατότητα να συγκρίνουμε τη συσκευή μας με οποιαδήποτε κυκλοφορεί στον κόσμο. Το Antutu στην έκτη έκδοσή του τεστάρει τέσσερις τομείς:

3D - τη δυνατότητα της συσκευής να τρέχει παιχνίδια με υψηλής ποιότητας γραφικά

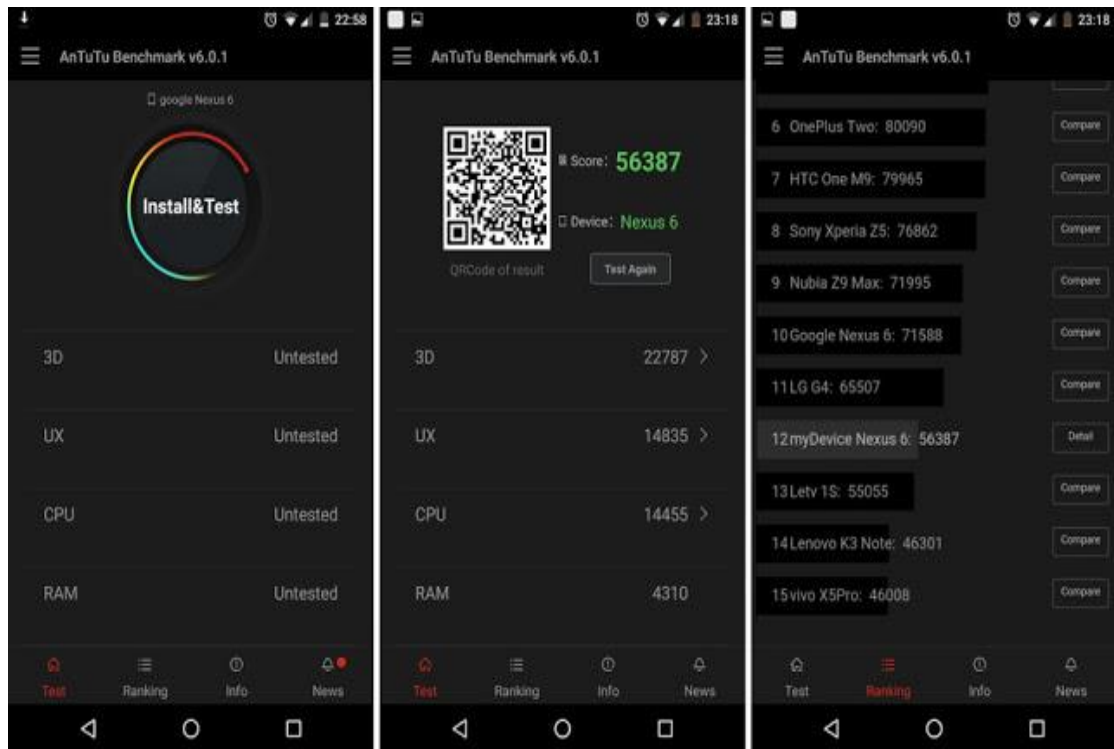
CPU - το μέγιστο φορτίο που μπορεί να αντέξει ο επεξεργαστής

RAM - το χώρο αποθήκευσης της RAM

UX - τις καθημερινές εργασίες που μπορεί να κάνουμε με τη συσκευή μας

Το Antutu παίρνει όλα τα επιμέρους σκορ και τα προσθέτει σε ένα, κάτι που είναι ιδιαίτερα βολικό για την άμεση σύγκριση της συσκευής με άλλες, αλλά δε μας δίνει τη δυνατότητα να καταλάβουμε πώς τα καταφέρνει σε κάθε μία κατηγορία ξεχωριστά. Μπορεί δηλαδή δύο συσκευές να έχουν 50.000 συνολικούς πόντους, αλλά στο 3D για παράδειγμα η μία να είναι χειρότερη από την άλλη, συμπληρώνοντας τους πόντους από κάποια άλλη κατηγορία. Έτσι λοιπόν η πρώτη δε θα τρέχει το ίδιο καλά ένα βαρύ παιχνίδι.

Τα επιμέρους σκορ είναι φυσικά στη διάθεσή μας, αλλά δεν υπάρχει κάποια κατάταξη για να τα συγκρίνουμε με άλλες συσκευές.



Vellamo

Το δεύτερο benchmark μας προσφέρει αυτό που λείπει από το Antutu. Το Vellamo τρέχει ένα σετ εργαλείων τα οποία μας δίνουν χρήσιμες πληροφορίες για τρεις κατηγορίες, που δείχνουν τη συμπεριφορά του σε καθημερινή χρήση:

Browser - στην ταχύτητά της συσκευής κατά την περιήγηση στο ίντερνετ

Metal - στις δυνατότητες του επεξεργαστή όταν τρέχει μία διεργασία, σε έναν μόνο πυρήνα

Multicore - στην ταχύτητα του επεξεργαστή γενικότερα

Πέρα από το Metal και το Multicore που επιβεβαιώνουν τα αποτελέσματα του Antutu, το πιο σημαντικό είναι το Browser με το οποίο μπορούμε όχι μόνο να δούμε αν η συσκευή μας είναι καλή στην περιήγηση στο ίντερνετ, αλλά να τσεκάρουμε ποιος browser τρέχει καλύτερα.



Geekbench 3

Μετά το Vellamo και αφού πήραμε μια ιδέα για τις δυνατότητες του επεξεργαστή, τρέχουμε το Geekbench, το οποίο είναι το καλύτερο σε αυτό τον τομέα και έρχεται να επιβεβαιώσει τα ως άνω αποτελέσματα. Το Geekbench λοιπόν τεστάρει κυρίως τρία πράγματα:

Single core - την ταχύτητα μόνο ενός πυρήνα του επεξεργαστή

Multi core - το φορτίο που μπορούν να αντέξουν όλοι οι πυρήνες

Battery - της αυτονομία της συσκευής

Το Geekbench είναι από τα πιο διαδεδομένα και όπως το Antutu, μας δίνει τη δυνατότητα να συγκρίνουμε τη συσκευή μας με πάρα πολλές από αυτές που ήδη κυκλοφορούν. Τα αποτελέσματα που αφορούν τον επεξεργαστή είναι ξεκάθαρα, αλλά ποιο σημαντικό είναι το test της αυτονομίας, το οποίο μπορεί να μας πει πόσο διαρκεί η μπαταρία σε μια πλήρη αποφόρτιση. Το μόνο κακό με αυτό είναι ότι το ενσωματωμένο συγκριτικό του δεν περιλαμβάνει μία τόσο μεγάλη γκάμα συσκευών και έτσι είναι κατά κάποιον τρόπο άχρηστο, αλλά μπορούμε να δούμε την πλήρη λίστα αποτελεσμάτων που υπάρχει στην επίσημη σελίδα του. Έτσι έχουμε τη δυνατότητα να αναζητήσουμε εκεί μία συσκευή πριν καν την αγοράσουμε.



3D Mark

Για το τέλος αφήσαμε αυτό που κάνει τη διαφορά σε μία συσκευή και δείχνει πραγματικά αν είναι δυνατή ή όχι. Πρόκειται για το 3D Mark το οποίο είναι το απόλυτο εργαλείο που σου δείχνει τι μπορεί να κάνει η συσκευή σου στο χειρότερο δυνατό σενάριο, τα βαριά παιχνίδια.

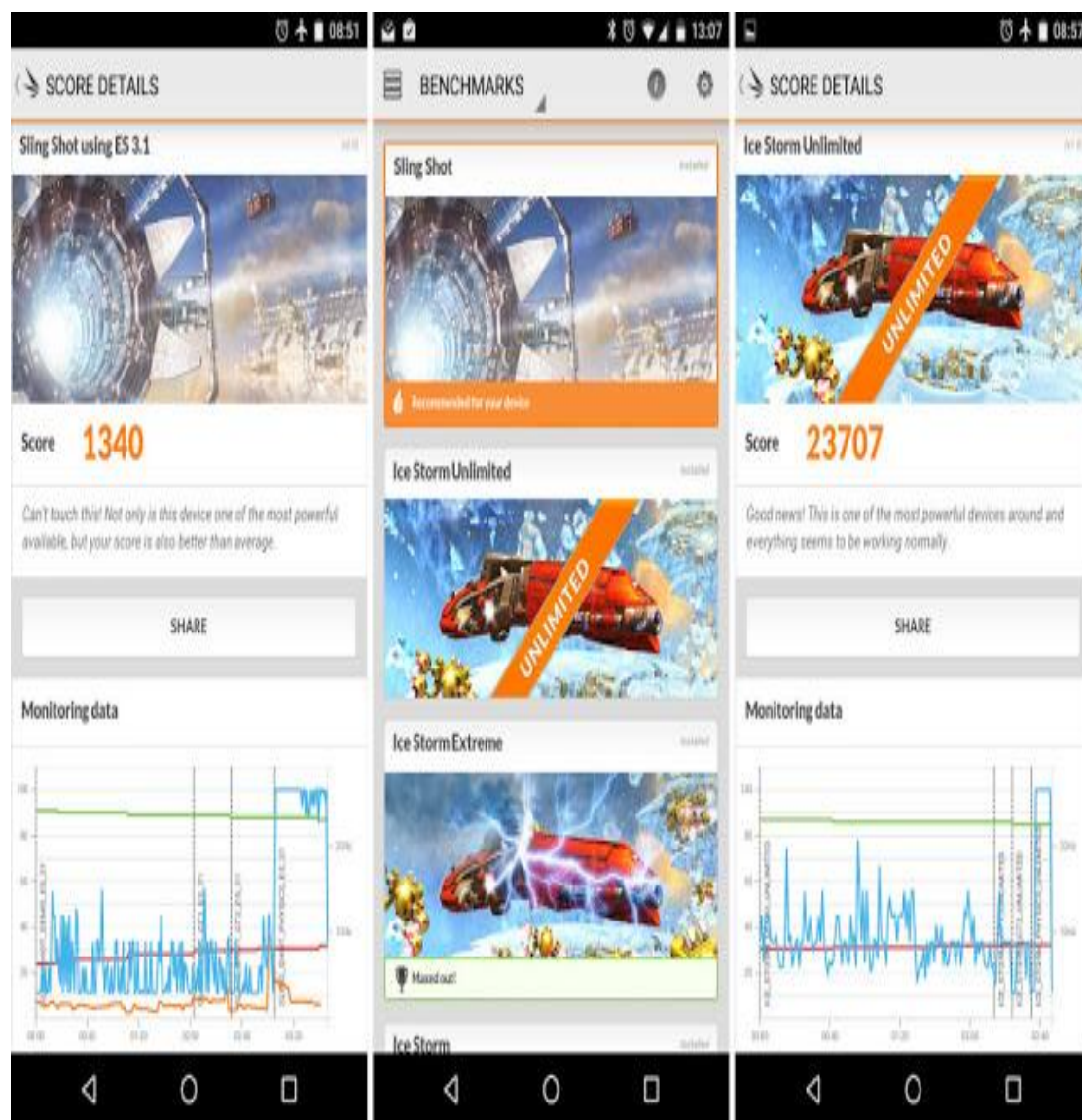
Το 3D Mark θα λέγαμε ότι ξεχωρίζει τους άντρες από τα γατάκια και το κάνει τεστάροντας πώς συμπεριφέρεται η συσκευή, όταν τρέχουμε παιχνίδια στην καλύτερη δυνατή έκδοση του OpenGL. Το συγκεκριμένο benchmark λαμβάνει υπ' όψιν και την ανάλυση της οθόνης της συσκευής μας, ενώ πριν καν ξεκινήσουμε, μας προτείνει ποιο είναι το καλύτερο benchmark και ποια είναι πολύ ελαφριά για τη συσκευή μας.

Το καλύτερο αυτή τη στιγμή είναι το Sling Shot, αλλά επειδή δεν υποστηρίζεται ακόμη από όλες τις συσκευές ή πολλές από αυτές απλά δεν μπορούν να το τρέξουν, στο Dr. Android τρέχουμε και το Ice Storm Unlimited, για να έχουμε συγκρίσιμα αποτελέσματα.

Σημείωση:

Εξίσου καλό εργαλείο, το οποίο δε χρησιμοποιούμε γιατί όλη τη μέρα θα έπρεπε να

κάνουμε benchmark, είναι το PC Mark το οποίο μας δίνει χρήσιμα συμπεράσματα όσον αφορά την μπαταρία και τις επιδόσεις σε καθημερινή χρήση.



8

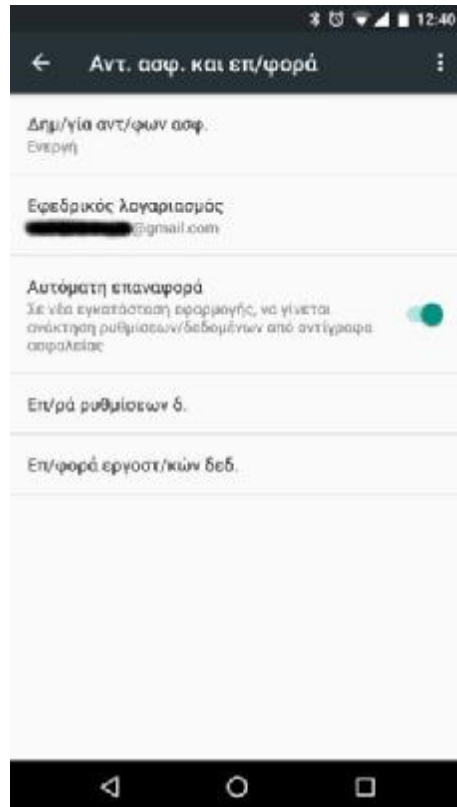
2.5 Τι είναι και πως κάνω επαναφορά εργοστασιακών δεδομένων.

Τα αντίγραφα ασφαλείας και η επαναφορά εργοστασιακών δεδομένων είναι δύο πολύ σημαντικές επιλογές που πιθανότατα δεν πρόκειται να χρησιμοποιήσει ποτέ ο μέσος χρήστης στο Android. Σε μία περίπτωση έκτακτης ανάγκης όμως, είναι τα καλύτερα εργαλεία για να σώσουμε ή να διαγράψουμε τα δεδομένα μας και μας γλιτώνουν από τεράστιο κόπο στη διαδικασία.

Βρίσκονται καταχωνιασμένα στο μενού της συσκευής μας και συγκεκριμένα στη διαδρομή "Ρυθμίσεις > Προσωπικό > Αντίγραφα ασφαλείας και επαναφορά" και δε

⁸ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/01/android-basics-benchmark.html>.

χρειάζεται να είσαι εξπέρ για να τα ρυθμίσεις, έτσι σήμερα θα δούμε τι ακριβώς κάνουν, σε ποιες περιπτώσεις μας εξυπηρετούν και πώς τα χρησιμοποιούμε.



Επαναφορά Εργοστασιακών Δεδομένων

Η επαναφορά εργοστασιακών δεδομένων (και όχι ρυθμίσεων πλέον) είναι ένα πολυεργαλείο που μπορεί να μας λύσει πολλά προβλήματα ταυτόχρονα. Τη χρησιμοποιούμε για να καθαρίσουμε τη συσκευή μας πριν την πουλήσουμε, για να απαλλαγούμε από κάποιο πρόβλημα που δεν εντοπίζεται ή απλώς για να τη φρεσκάρουμε, σε όλες όμως τις περιπτώσεις πρέπει να γνωρίζουμε πως ΟΛΑ ΤΑ ΔΕΔΟΜΕΝΑ ΜΑΣ ΘΑ ΧΑΘΟΥΝ!

Η επαναφορά εργοστασιακών ρυθμίσεων διαγράφει το “data”, το “cache” και το “sdcard” partition και μπορεί να γίνει είτε από τις ρυθμίσεις της συσκευής, είτε από το recovery mode όπου θα μπορούμε να το επιλέξουμε “Wipe data/factory reset”.



Το "sdcard" partition είναι στην πραγματικότητα ο εσωτερικός χώρος της συσκευής και όχι η πραγματική κάρτα SD, η οποία δε διαγράφεται. Στο μενού της επαναφοράς ορισμένων συσκευών, θα δούμε ότι υπάρχει η επιλογή να διαγραφεί και η SD κάρτα, αλλά θα πρέπει να την επιλέξουμε μόνοι μας. Στην έκδοση 6.0 του Android έχει εμφανιστεί και μία νέα επιλογή, η "Επαναφορά ρυθμίσεων δεδομένων" η οποία διαγράφει όλες τις ρυθμίσεις που αφορούν τα δίκτυα (WiFi, κινητής τηλεφωνίας, Bluetooth κλπ.), αλλά δε γνωρίζουμε ακόμη πολλά πράγματα για αυτήν. Κλείνοντας θα πρέπει να πούμε ότι σε περίπτωση που θέλουμε να πουλήσουμε τη συσκευή μας, πρέπει να κάνουμε οπωσδήποτε επαναφορά εργοστασιακών δεδομένων αλλά πρώτα θα πρέπει να κρυπτογραφήσουμε τα δεδομένα μας, καθώς μπορούν να ανακτηθούν.

Σημείωση:

Η λογική λέει ότι με την επαναφορά, επιστρέφουμε το λογισμικό της συσκευής μας στη μορφή που είχε όταν έφυγε από το εργοστάσιο, δηλαδή όπως ακριβώς την αγοράσαμε, σωστά; Όχι! Γιατί με το Android μάθαμε να επεμβαίνουμε μόνοι μας στο system partition, κάτι που δεν επηρεάζεται από την επαναφορά. Έτσι αν έχουμε αλλάξει ROM στη συσκευή μας ή την έχουμε χακάρει κάνοντας root ή εγκαθιστώντας για παράδειγμα το Xposed framework, μετά την επαναφορά οι αλλαγές αυτές θα παραμείνουν.

Υπάρχουν ένα σωρό εφαρμογές στο Play Store που μπορούν να μας εξυπηρετήσουν για να σώσουμε τα δεδομένα μας, αλλά έχουν τα δικά τους προβλήματα. Για παράδειγμα μας έχει τύχει στο παρελθόν να κάνουμε backup με μια από αυτές και αφού αναβαθμίσαμε σε ποιο καινούργια έκδοση να Android η εφαρμογή που είχαμε κάνει backup δεν ήταν συμβατή. Το Android λοιπόν διαθέτει τη δική του ενσωματωμένη λύση η οποία επίσης δεν καλύπτει όλες τις ανάγκες μας, αλλά βελτιώθηκε σημαντικά στις εκδόσεις 5.0 και 6.0.

Ενεργοποιώντας λοιπόν τη "Δημιουργία αντιγράφων ασφαλείας", αποθηκεύουμε στον προσωπικό μας λογαριασμό στη Google (Εφεδρικός λογαριασμός) κάποιες ρυθμίσεις όπως το ιστορικό κλήσεων και οι κωδικοί πρόσβασης στα δίκτυα WiFi που έχουμε συνδεθεί, ενώ στην έκδοση 6.0, το Android μπορεί να αποθηκεύσει και δεδομένα τρίτων εφαρμογών (έως 25MB), αν οι δημιουργοί τους έχουν προβλέψει κάτι τέτοιο φυσικά.

Σε αυτά τα αντίγραφα ασφαλείας δεν αποθηκεύονται:

οι επαφές μας, για τις οποίες υπάρχει το Google contacts,

τα save των παιχνιδιών μας, για τα οποία μπορούμε να χρησιμοποιήσουμε το Google Play Games,

τις φωτογραφίες και τα βίντεο που έχουμε τραβήξει, τα οποία συγχρονίζονται αυτόματα με το Google Photos,

τη μουσική μας, η οποία μπορεί να αποθηκευτεί στο Google Music,

τα υπόλοιπα αρχεία μας για τα οποία υπάρχει το Google Drive και τέλος

τα SMS, για τα οποία θα χρειαστούμε μία τρίτη εφαρμογή όπως το Super Backup.



Η Google κρατάει επίσης μια λίστα με τις εφαρμογές που έχουμε στην συσκευή μας, τις οποίες μπορούμε να κατεβάσουμε αυτόματα μετά από την επαναφορά εργοστασιακών δεδομένων, αν έχουμε από πριν ενεργοποιήσει την επιλογή "Αυτόματη

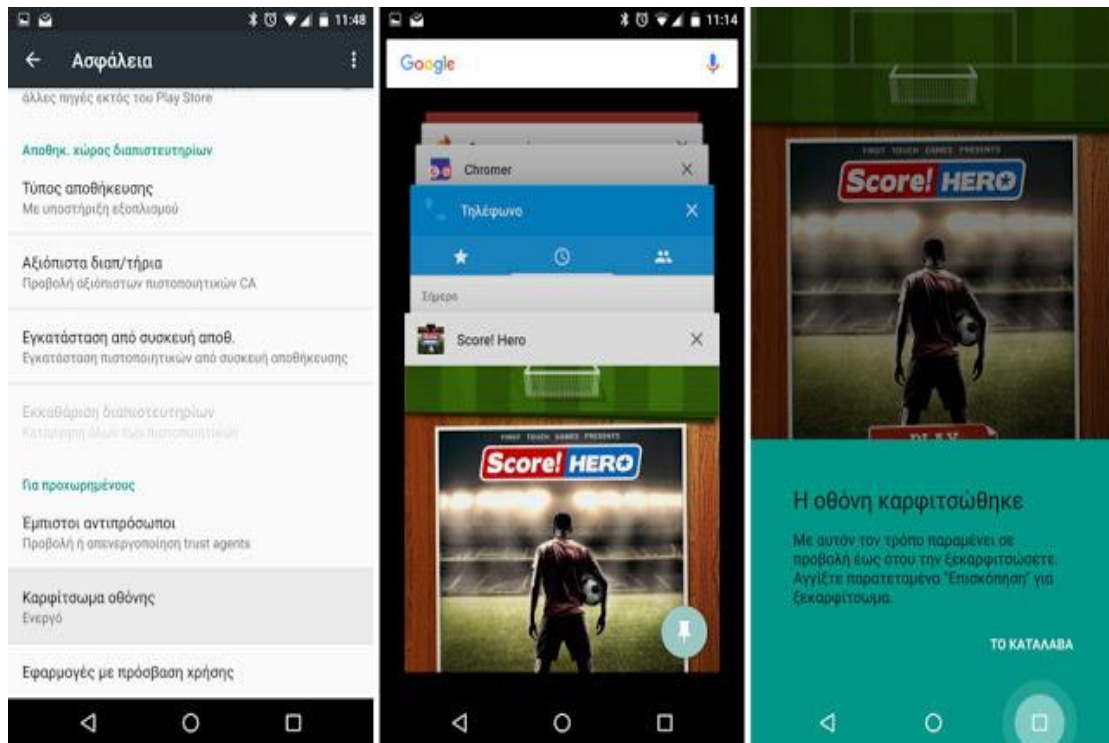
Επαναφορά”. Βασικό σε όλη αυτή την διαδικασία είναι κατά την πρώτη ενεργοποίηση της συσκευής μας, μετά από επαναφορά εργοστασιακών ρυθμίσεων, να ακολουθήσουμε τον οδηγό ενεργοποίησης συσκευής, χωρίς να προσπεράσουμε κάποιο βήμα, ειδικά θα πρέπει να ξανάκάνουμε επαναφορά για να επανέλθουν αυτόματα. Επειδή ως γνωστόν οι Android συσκευές διαφέρουν οπτικά μεταξύ τους, οι αναφορές στα μενού αφορούν κυρίως όσες διαθέτουν AOSP ROM, όπως τα Nexus. Σε ορισμένες περιπτώσεις μάλιστα οι ρυθμίσεις που αποθηκεύουμε στο server της Google δεν επανέρχονται ποτέ, αλλά αυτό είναι κάτι το οποίο δεν μπορούμε να διορθώσουμε.⁹

2.6 Καρφίτσωμα εφαρμογής

Δεν υπάρχει άνθρωπος που να μη δανείζει έστω και για λίγο το κινητό ή το tablet του σε κάποιο τρίτο πρόσωπο. Μπορεί να είναι ένας φίλος που θέλει να τσεκάρει μια σελίδα στο ίντερνετ ή ο μπόμπιρας του σπιτιού που θέλει να παίξει ένα παιχνίδι, αυτό όμως δε σημαίνει ότι του επιτρέπουμε να χρησιμοποιήσει όλες τις εφαρμογές της συσκευής.

Σε αυτές τις περιπτώσεις μπορούμε να χρησιμοποιήσουμε το "Καρφίτσωμα οθόνης", το οποίο απενεργοποιεί τα πλήκτρα πλοήγησης του Android και δε μας επιτρέπει αν βγούμε από την εφαρμογή που βρισκόμαστε. Για να το ενεργοποιήσουμε πηγαίνουμε στη διαδρομή "Ρυθμίσεις > Ασφάλεια" και επιλέγουμε το "Καρφίτσωμα οθόνης". Στη συνέχεια ανοίγουμε την εφαρμογή που θέλουμε, πατάμε το "τετράγωνο" πλήκτρο (επισκόπηση) και επιλέγουμε την πινέζα στην κάρτα της εφαρμογής.

⁹ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2015/09/backup-restore.html>.



Για να ξεκαρφίτσώσουμε ξανά την εφαρμογή και κατ' επέκταση να χρησιμοποιήσουμε κανονικά το τηλέφωνό μας, θα πρέπει να κρατήσουμε για λίγο πατημένα το "τρίγωνο" (πίσω) και το "τετράγωνο" (επισκόπηση) και να βάλουμε το PIN ή το Μοτίβο που έχουμε ορίσει.

Πολύ καλό χαρακτηριστικό του Android, το οποίο μπορεί να μας φανεί χρήσιμο και στις περιπτώσεις που παίζουμε ένα παιχνίδι και δε θέλουμε να το ελαχιστοποιήσουμε κατά λάθος.¹⁰

2.7 Τι είναι η ασφαλής λειτουργία.

Τα λειτουργικά συστήματα που τρέχουν στους υπολογιστές και τα κινητά μας, είναι ένας περίπλοκος οργανισμός μικρότερων τμημάτων software που λειτουργούν συνεργατικά. Πολλές φορές όμως η συνεργασία αυτή χαλάει, δημιουργώντας προβλήματα στο συστήμα μας, τα οποία είναι αρκετά δύσκολο να αντιμετωπιστούν χωρίς την ύπαρξη της ασφαλούς λειτουργίας (Safe mode).

¹⁰ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/03/android-pin-app-feature.html>.



Το Safe mode μπορεί να είναι στην ουσία μια λειτουργία η οποία αποκλείει όλα τα προγράμματα που έχουμε εγκαταστήσει, τρέχοντας το σύστημα μόνο με τα απαραίτητα (τα προ εγκατεστημένα). Στους υπολογιστές μας, η ασφαλης λειτουργία αποκλείει ακόμη και τμήματα του hardware αφού ούτε οι driver τους λειτουργούν αν φυσικά τους έχουμε εγκαταστήσει αργότερα. Έτσι έχουμε ένα δοκιμασμένο από τον κατασκευαστή περιβάλλον, το οποίο λειτουργεί 100% και μέσω αυτού μπορούμε να κάνουμε διάφορες εργασίες και εκκαθαρίσεις στο σύστημα.

Όταν είμαστε σε safe mode στο Android μπορούμε να κάνουμε κυρίως δύο πράγματα:

Να απαλλαγούμε από ένα κακόβουλο λογισμικό και να ξεχωρίσουμε αν η συσκευή μας έχει πρόβλημα με το hardware ή το software.

Όσον αφορά το πρώτο το αναλύσαμε αρκετά σε [παλαιότερο άρθρο](#) και μάθαμε πώς να αφαιρούμε μία τέτοια εφαρμογή, έτσι έχει πολύ μεγάλη αξία να αναλύσουμε το δεύτερο, το οποίο θα μας γλιτώσει ακόμη και από μία επαναφορά εργοστασιακών δεδομένων!

Crash, reboot, battery, drain

Τα τρία πιο γνωστά προβλήματα στο Android είναι τα κρashaρίσματα εφαρμογών, οι επανεκκινήσεις και η κατανάλωση ενέργειας. Επειδή συνήθως δεν μπορούμε να εντοπίσουμε ποια εφαρμογή προκαλεί το πρόβλημα, η μόνη λύση (εκτός από την εκκαθάριση δεδομένων εφαρμογής) είναι η επαναφορά εργοστασιακών δεδομένων, κάτι που προσπαθούμε πάντα να αποφύγουμε. Ιδιαίτερα στην περίπτωση της υψηλής κατανάλωσης ενέργειας υπάρχουν περιπτώσεις ανθρώπων που έσβησαν τα δεδομένα

του χωρίς λόγο, αφού τελικά έφταιγε η ίδια η μπαταρία (το hardware δηλαδή), την οποία έπρεπε να αλλάξουμε.

Αν λοιπόν παρατηρήσουμε μία τέτοιου είδους δυσλειτουργία όπως οι επανεκκινήσεις, η κατανάλωση ενέργειας αλλά και η χωρίς λόγο υπερθέρμανση της συσκευής μας, δε χρειάζεται να προχωρήσουμε άμεσα σε επαναφορά εργοστασιακών δεδομένων. Θα πρέπει να βάλουμε τη συσκευή μας σε Safe mode και να τη χρησιμοποιήσουμε με αυτό τον τρόπο για μία μέρα ή περισσότερο. Αν το πρόβλημα σταματήσει να υπάρχει, τότε είμαστε σίγουροι πως φταίει το λογισμικό, αν όμως συνεχίζεται τότε φταίει η ίδια η συσκευή και χρειάζεται αλλαγή.

Πως μπαίνουμε σε safe mode ;

1. Κρατάμε πατημένο το πλήκτρο "Power" μέχρι να εμφανιστεί το μενού της απενεργοποίησης.
2. Κρατάμε πατημένη την επιλογή "Απενεργοποίηση".
3. Στο παράθυρο που θα μας εμφανίσει επιλέγουμε να επανεκκινήσει η συσκευή μας σε Safe mode.



Αν όμως η ζημιά στο software είναι τέτοια που δεν μπορούμε να μπούμε στο Android για να κάνουμε αυτές τις επιλογές , τότε τι γίνεται;

Για αυτές τις περιπτώσεις υπάρχει ένας δεύτερος τρόπος μέσω των πλήκτρων της συσκευής.

Η συσκευή μας θα πρέπει να ενεργοποιηθεί κρατώντας το πλήκτρο "Power" και μόλις δούμε το logo του κατασκευαστή, το αφήνουμε και πιέζουμε ένα από τα παρακάτω:

για Samsung κρατάμε το "Volume Up" ή πιέζουμε πολλές φορές το "Menu",
για HTC κρατάμε το "Volume Down" και
για Motorola κρατάμε και τα δύο πλήκτρα "Volume".

Πότε το safe mode μας είναι άχρηστο ;

Τέλος υπάρχουν περιπτώσεις όπου το Safe mode είναι απλά άχρηστο, με μία από αυτές να οφείλεται σε root εφαρμογές. Αν κάποια από αυτές εγκατασταθεί στο system partition (όπως το SuperSU για παράδειγμα), θα τρέχει και στο Safe mode, αφού το Android θεωρεί πως ότι βρίσκεται στο system, είναι ελεγμένο και δεν αλλάζει. Εκεί δεν υπάρχει τρόπος λύσης μέσω safe mode και θα πρέπει να πάμε σε άλλες λύσεις, όπως για παράδειγμα απεγκατάσταση μέσω των εντολών ADB.

Όπως και να 'χει το Safe mode είναι ακόμη ένα εργαλείο που μπορεί να μας βγάλει από μια δύσκολη θέση, δίνοντάς μας την ευκαιρία να διορθώσουμε τη συσκευή μας, χωρίς να επισκεφτούμε το σέρβις και φυσικά χωρίς να πληρώσουμε κανέναν για αυτό.

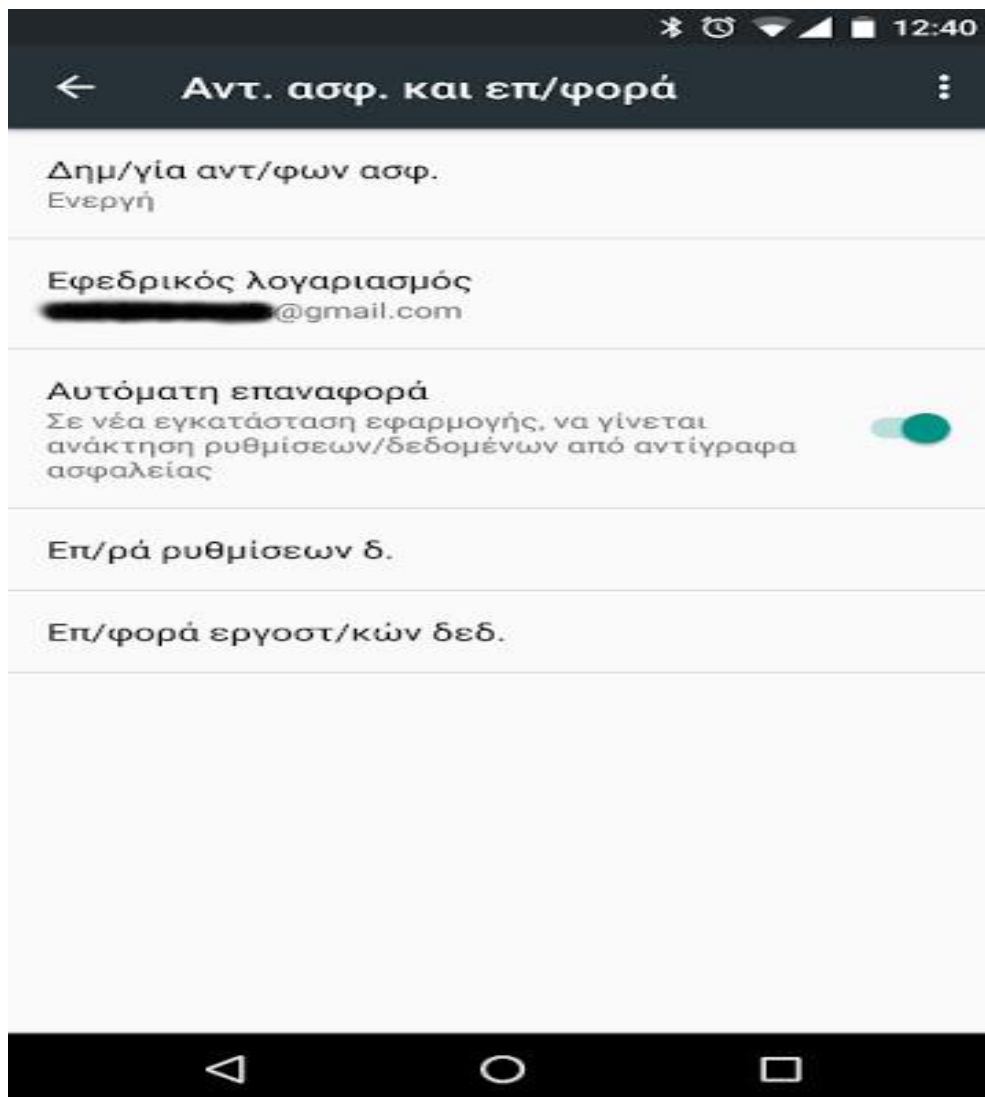
Τι είναι η επαναφορά εργοστασιακών δεδομένων και τα αντίγραφα ασφαλείας;

Τα αντίγραφα ασφαλείας και η επαναφορά εργοστασιακών δεδομένων είναι δύο πολύ σημαντικές επιλογές που πιθανότατα δεν πρόκειται να χρησιμοποιήσει ποτέ ο μέσος χρήστης στο Android. Σε μία περίπτωση έκτακτης ανάγκης όμως, είναι τα καλύτερα εργαλεία για να σώσουμε ή να διαγράψουμε τα δεδομένα μας και μας γλιτώνουν από τεράστιο κόπο στην διαδικασία.

Βρίσκονται στο μενού της συσκευής μας και συγκεκριμένα στην διαδρομή

''Ρυθμίσεις>Προσωπικό>Αντίγραφα ασφαλείας και επαναφορά''.¹¹

¹¹ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2015/09/backup-restore.html>.



2.8 Δυσκολία μεταφοράς στην κάρτα SD.

Το Android αλλάζει καθημερινά, όχι μόνο εξωτερικά αλλά και στο εσωτερικό του βελτιώνοντας σε κάθε έκδοση πολλά εκνευριστικά στοιχεία του παρελθόντος. Ένα από αυτά είναι και ο μειωμένος αποθηκευτικός χώρος των παλαιότερων συσκευών.

Προσαρτήσεις Partitions

Ο αποθηκευτικός χώρος κάθε Android συσκευής είναι ένας, στην πραγματικότητα όμως έχει χωριστεί σε διάφορα κομμάτια για να εξυπηρετεί διαφορετικούς σκοπούς. Μεταξύ άλλων στο Android έχουμε: το system partition στο οποίο βρίσκονται οι προεγκατεστημένες εφαρμογές και άλλα αρχεία του συστήματος, το data partition στο οποίο αποθηκεύονται οι εφαρμογές που κατεβάζουμε και το sdcardpartition, το οποίο εξυπηρετεί τα υπόλοιπα αρχεία μας και τα δεδομένα των εφαρμογών.



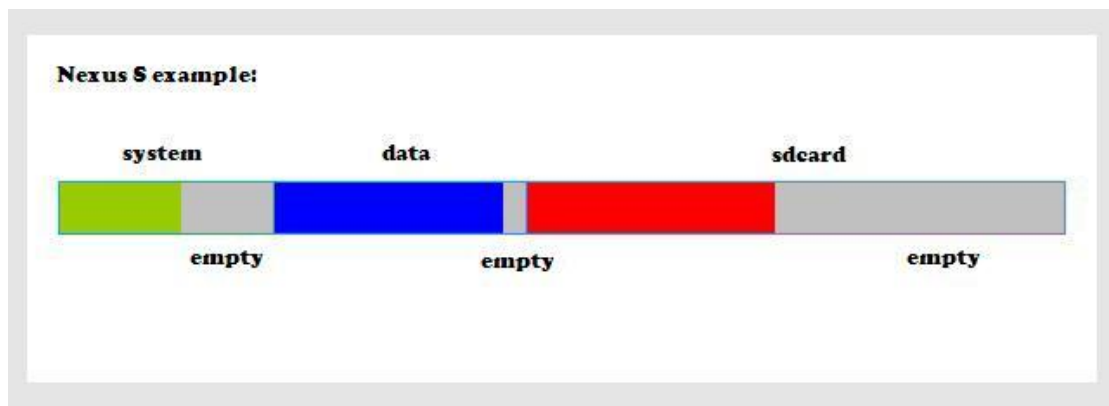
Εδώ πρέπει να διευκρινίσουμε ότι άσχετα αν ονομάζεται “sdcard” το τελευταίο, δεν είναι η SD κάρτα που βάζουμε εξωτερικά στη συσκευή μας. Αυτή ονομάζεται ως “external” ή “removable”.

Τα παλιά τα χρόνια, αφού ξεκαθαρίσαμε πώς είναι διαμορφωμένος ο δίσκος του Android, ας μιλήσουμε για το λάθος που γινόταν παλαιότερα. Σε μία συσκευή όπως για παράδειγμα το Nexus S το οποίο διαθέτει αποθηκευτικό χώρο 16GB, οι προσαρτήσεις είναι διαμορφωμένες ως εξής :

500MB – system

1GB – data

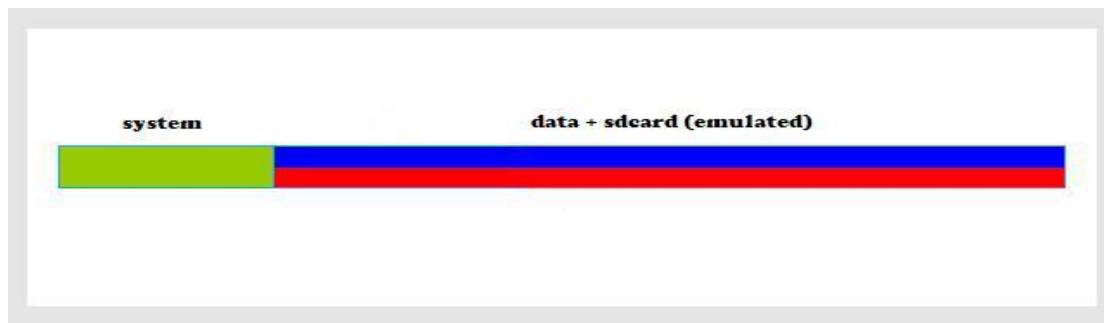
14GB – sdcard



Βλέπουμε λοιπόν ότι στο σύνολο των 16GB, μόνο το 1GB είναι διαθέσιμο για εφαρμογές, ενώ τα υπόλοιπα 14GB είναι διαθέσιμα για τα αρχεία μας, μιας και το Nexus S δε διαθέτει θύρα για φυσική SD κάρτα. Πρακτικά αυτό σημαίνει ότι οι εφαρμογές που μπορούμε να εγκαταστήσουμε μπορούν να φτάσουν έως το 1GB έχοντας την ευχέρεια να μεταφέρουμε στην sdcard (και εννοούμε το partition) ένα μέρος τους. Όπως βλέπουμε στην παραπάνω εικόνα όμως, πάντα θα υπάρχει ανεκμετάλλετος χώρος (empty) τον οποίο θα θέλαμε να χρησιμοποιήσουμε. Από το Android 4.2 και μετά. Πρόσφατα και με την έλευση του Android 4.2 το file system άλλαξε προς το καλύτερο και αυτό γιατί στη νέα έκδοση έχουμε τη δυνατότητα να

δημιουργούμε πολλαπλούς χρήστες σε μια συσκευή, των οποίων τα αρχεία δεν μπορούν να αποθηκεύονται στον ίδιο χώρο.

Έτσι το data και το sdcard partition ενώθηκαν και συνυπάρχουν μέσω ενός emulator.



Με αυτό τον τρόπο μπορούμε να διαχειριστούμε όπως θέλουμε το χώρο της συσκευής και για να δώσουμε ένα παράδειγμα, αν θέλουμε να εγκαταστήσουμε περισσότερες εφαρμογές σε μία συσκευή που δεν έχει πια χώρο, μπορούμε να διαγράψουμε αρχεία ήχου, βίντεο ή ότι άλλο θέλουμε για να το πετύχουμε, κάτι που δε γινόταν παλιότερα. Από το Android 4.2 και μετά λοιπόν, όταν λέμε ότι μία συσκευή διαθέτει εσωτερική μνήμη 8GB σημαίνει ότι είναι διαθέσιμα για αρχεία ή εφαρμογές.

2.8.1 Η ιδιαιτερότητα του Galaxy S4

Υπάρχει μια ιδιαιτερότητα με το Galaxy S4 της Samsung, το οποίο ως γνωστό διαθέτει 16 GB εσωτερικής μνήμης αλλά λόγω των πολλών υπηρεσιών που έχει εγκατεστημένες ο χώρος που είναι τελικά διαθέσιμος είναι μόνο 9GB. Η Samsung αποφάσισε με επερχόμενη αναβάθμιση να δώσει τη λειτουργία μεταφοράς εφαρμογών στην πραγματική SD κάρτα αυτή τη φορά, όπως ακριβώς κάνουμε έχοντας root με το App2SD,

Αποτέλεσμα;

Το αποτέλεσμα είναι περισσότερος διαθέσιμος χώρος για να εγκαταστήσουμε ότι θέλουμε και λιγότερες απώλειες ανά partition. Αν συσκευή που θέλουμε να αγοράσουμε είναι διαμορφωμένη με αυτόν τον τον τρόπο , αρκεί να ανοίξουμε τον ES File Explorer και να δούμε την διαδρομή που οδηγεί στο sdcard partition. Θα είναι :

/storage/emulated/0

ή

/storage/sdcard0¹²

¹² Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2013/06/android-basics-new-emulated-file-system.html>.

2.9 Το Android χρησιμοποιείται ακόμα και σε tablet

Ένας υπολογιστής tablet είναι ένας φορητός υπολογιστής ή προσωπικός ψηφιακός βοηθός και είναι σε μέγεθος μεγαλύτερο από ένα κινητό τηλέφωνο, ενσωματωμένο σε μια επίπεδη οθόνη αφής και κυρίως λειτουργεί αγγίζοντας την οθόνη αντι να χρησιμοποιεί φυσικό πληκτρολόγιο. Χρησιμοποιεί συχνά την οθόνη σαν εικονικό πληκτρολόγιο, μια παθητική γραφίδα ή μια ψηφιακή πένα. Ο όρος μπορεί επίσης να εφαρμοστεί σε μια ποικιλία από μορφές παραγόντων που διαφέρουν ως προς τη θέση της οθόνης σε σχέση με το φυσικό πληκτρολόγιο. Το τυποποιημένο έντυπο αυτό, που ονομάζεται πλάκα, η οποία δεν διαθέτει ενσωματωμένο πληκτρολόγιο, αλλά έχει την δυνατότητα να συνδεθεί με ένα ασύρματο σύνδεσμο ή με μια θύρα USB.

12. PC Magazine Editors. "Ορισμός του: το tablet computer" . περιοδικού PC Magazine . Ανακτήθηκε 17 του Απρίλη 2010 .

Πέραν του ότι έχει πλέον τις δυνατότητες ενός προσωπικού υπολογιστή, οι υπολογιστές ταμπλέτες που αγοράζονται κατά τα τελευταία έτη περιλαμβάνουν και : ασύρματη περιήγηση στο Διαδίκτυο Internet, έχει τις δυναμικές λειτουργίες του κινητού τηλεφώνου, με την πλοήγηση GPS, την φωτογραφική μηχανή και τις λειτουργίες βίντεο, το βάρος τους είναι μικρό και ζυγίζουν γύρω στα 500 με 600 γραμμάρια και τυπικά έχουν διάρκεια ζωής της μπαταρίας τους από 2,5 έως δέκα ώρες. Με πολλούς τρόπους ο φορητός υπολογιστής και τα smartphone πλησιάζουν στα χαρακτηριστικά τους, και αλληλοκαλύπτουν το ένα το άλλο.¹³

2.10 Κινητά Smartphones με λογισμικό android

Το smartphone, ή με τον ελληνικό όρο έξυπνο τηλέφωνο, είναι ένα κινητό τηλέφωνο βασισμένο σε ένα λειτουργικό σύστημα κινητής τηλεφωνίας με περισσότερο προηγμένη υπολογιστική ικανότητα και συνδεσιμότητα σε σχέση με ένα συμβατικό κινητό τηλέφωνο. Τα πρώτα smartphones συνδύαζαν τις λειτουργίες ενός προσωπικού ψηφιακού βοηθού (PDA) και ενός κινητού τηλεφώνου. Σε μεταγενέστερα μοντέλα προστέθηκαν οι λειτουργίες των φορητών media players, low-end compact ψηφιακές φωτογραφικές μηχανές, βιντεοκάμερες τσέπης, καθώς και μονάδες πλοήγησης GPS, με αποτέλεσμα την διαμόρφωση μιας πολυχρηστικής συσκευής. Πολλά σύγχρονα smartphones περιλαμβάνουν επίσης οθόνες αφής υψηλής

¹³ Συντάκτες Dictionary.com, «δισκίο υπολογιστή - ένα αποτέλεσμα λεξικό" Dictionary.com , ανακτήθηκε 17 του Απρίλη 2010

Gray, ο Ελισσαιέ (07.31.1888), Telautograph , Ηνωμένες Πολιτείες Διπλωμάτων Ευρεσιτεχνίας 386.815 (πλήρης εικόνα)

ανάλυσης και web browsers που εμφανίζουν τυποποιημένες ιστοσελίδες, καθώς και βελτιστοποιημένες ιστοσελίδες για κινητά. Η πρόσβαση σε δεδομένα υψηλής ταχύτητας παρέχεται μέσω Wi-Fi και μέσω κινητών ευρυζωνικών υπηρεσιών. Τα τελευταία χρόνια, η ταχεία ανάπτυξη στην αγορά των εφαρμογών για κινητά και στο εμπόριο κινητών τηλεφώνων έχει γίνει οδηγός για την ευρεία υιοθέτηση των smartphones.

Τα λειτουργικά συστήματα (OS) των κινητών τηλεφώνων που χρησιμοποιούνται από τα σύγχρονα smartphones περιλαμβάνουν, μεταξύ άλλων, το Android της Google, το iOS της Apple, το Symbian της Nokia, το BlackBerry OS της RIM, το Bada της Samsung, τα Windows Phone της Microsoft, το webOS της Hewlett-Packard, καθώς και ενσωματωμένες διανομές Linux όπως το Maemo και το MeeGo. Τετοιου είδους λειτουργικά συστήματα μπορούν να εγκατασταθούν σε πολλά διαφορετικά μοντέλα κινητών και συνήθως κάθε συσκευή μπορεί να λάβει πολλές ενημερωμένες εκδόσεις λογισμικού λειτουργικού συστήματος κατά την διάρκεια ζωής της. Μερικά άλλα επερχόμενα λειτουργικά συστήματα είναι το Firefox OS της Mozilla, το Ubuntu Phone της Canonical Ltd's και το Tizen. Η διάκριση μεταξύ των smartphones και των απλών κινητών τηλεφώνων μπορεί να είναι ασαφής και δεν υπάρχει επίσημος ορισμός για το ποιες είναι οι μεταξύ τους διαφορές. Μία από τις πιο σημαντικές διαφορές είναι ότι οι προηγμένες διεπαφές προγραμματισμού εφαρμογών (APIs) στα smartphones σχετικά με τη λειτουργία τρίτων εφαρμογών μπορούν να επιτρέψουν σε αυτές τις εφαρμογές να έχουν καλύτερη ενσωμάτωση στο λειτουργικό σύστημα και στο hardware του τηλεφώνου απ' ό,τι συμβαίνει συνήθως στα απλά κινητά τηλέφωνα. Συγκριτικά, τα συμβατικά κινητά τηλέφωνα τρέχουν συχνότερα σε ιδιόκτητο firmware, με υποστήριξη λογισμικού από τρίτους, μέσα από πλατφόρμες όπως το Java ME ή το BREW. Μια επιπλέον δυσκολία στη διάκριση μεταξύ smartphones και συμβατικών κινητών τηλεφώνων είναι ότι, με την πάροδο του χρόνου, οι δυνατότητες των νέων μοντέλων των απλών κινητών τηλεφώνων μπορούν να υπερβούν εκείνες των τηλεφώνων που είχαν προωθηθεί ως smartphones στο παρελθόν. Ορισμένοι κατασκευαστές και πάροχοι χρησιμοποιούν τον όρο *superphone* για τα υψηλής απόδοσης τηλέφωνα τους με τις ασυνήθιστα μεγάλες οθόνες και τα άλλα ακριβά χαρακτηριστικά τους. Άλλοι προτιμούν τον όρο <phablet>, αναγνωρίζοντας τη σύγκλιση τους με τους υπολογιστές tablet χαμηλής απόδοσης.¹⁴

¹⁴ «Smartphone». Phone Scoop. Ανακτήθηκε στις 2011-12-15.

«Feature Phone». Phone Scoop. Ανακτήθηκε στις 2011-12-15.

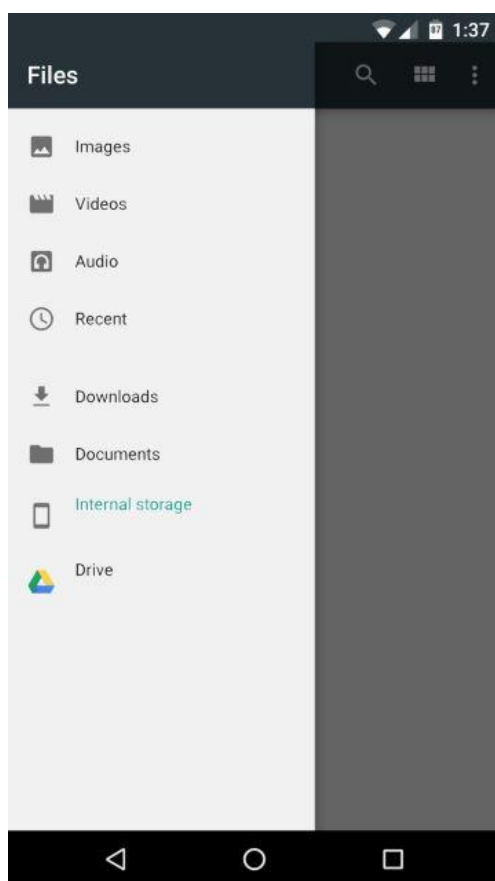
Andrew Nusca (20 August 2009). «Smartphone vs. feature phone arms race heats up; which did you buy?». ZDNet. Ανακτήθηκε στις 2011-12-15

2.11 Βελτιώσεις στο File Manager

Στο Android Marshmallow η Google ενσωμάτωσε έναν -ο οποίος όμως στερείτω βασικές λειτουργίες-. Στο Android N όμως, τον βελτίωσε ουσιαστικά.

Πλέον βρίσκουμε όλες τις βασικές λειτουργίες διαχείρισης αρχείων και φακέλων (Copy, Move, Rename) που θα περίμενε κάποιος από έναν File Manager (και όχι μόνο Copy όπως πριν). Επίσης υπάρχει η πλαϊνή μπάρα για εύκολότερη περιήγηση ανάμεσα σε Directories, αλλά και γρηγορότερη εμφάνιση αρχείων μίας κατηγορίας (π.χ. μόνο εικόνες ανεξαρτήτως φακέλου). Μπορούμε να έχουμε δύο (ή παραπάνω) instances του File Explorer ανοιχτά στο Multitasking Menu. Οι αλλαγές κλείνουν με κάποιες μικροβελτιώσεις όπως π.χ. εμφάνιση ενός "κενού συρταριού" όταν ανοίξουμε έναν φάκελο χωρίς περιεχόμενο.

Στα αρνητικά: ο συγκεκριμένος File Manager παραμένει κρυμμένος στην εφαρμογή των Ρυθμίσεων, μέσα στο Storage -> Explore. Τον θέλω ανεξάρτητα στον App Drawer. Δεν υπάρχει ακόμα κάποιο Properties/Details για αρχεία/φακέλους. Ακόμα όμως και με την "τωρινή" μορφή του στο Android N μπορεί να δώσει αρκετή χρηστικότητα στον μέσο χρήστη, αναιώντας (για πολλούς) την ανάγκη για χρήση εφαρμογής τρίτων.¹⁵



¹⁵ Greece, A. (15 Μαΐου, 15-5-2016). www.greeceandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.greeceandroid.gr/dev/1595-android-n-file-manager>.

2.13 Καινούργια διαθέσιμη λειτουργία για το Android

Το Android N μόλις κυκλοφόρησε για τους Developers, ή για όσους ζουν επικίνδυνα φλασάροντας Beta εκδόσεις λειτουργικών, και σιγά σιγά ανακαλύπτουμε όλα τα νέα χαρακτηριστικά και λειτουργίες του.

Μία ενδιαφέρουσα αλλαγή αφορά την δυνατότητα να κάνουμε calibration της οθόνης αναλόγως των "γούστων" μας, αλλά και να αλλάξουμε το Density, μεγαλώνοντας ή μικραίνοντας το σύνολο του περιεχομένου σε αυτήν.

Κάποιες από τις αλλαγές αυτές απαιτούσαν Root (ή/και adb) σε συσκευές με Vanilla Android. Προσωπικά βρίσκω ευχάριστο που ο μέσος χρήστης θα έχει πλέον αυτές τις δυνατότητες χωρίς να "μπλέκει" με το Root (και τους κινδύνους ασφαλείας που αυτό συνεπάγεται).

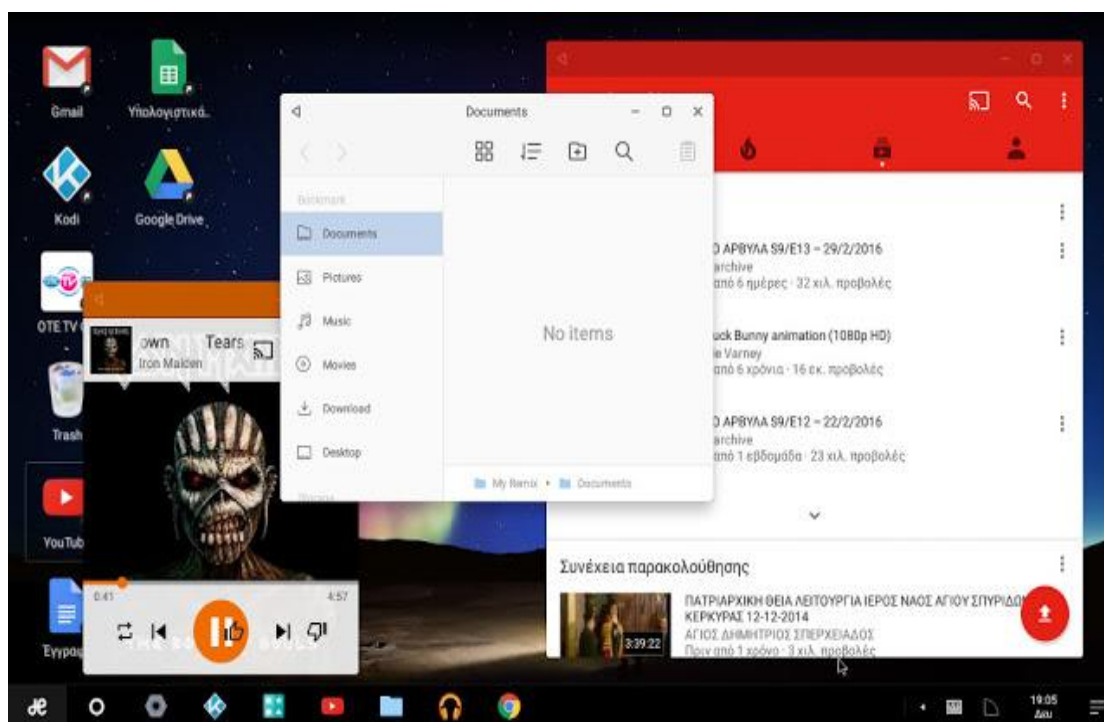
Όσον αφορά το Calibration των χρωμάτων πλέον μπορούμε μέσα από το System UI Tuner να αλλάξουμε τα R/G/B Channels για να πετύχουμε "ακριβώς" το οπτικό αποτέλεσμα που μας καλύπτει.



Αντίστοιχα μπορούμε να αλλάξουμε και το "density" (μέσα από επιλογή στο Accessibility) για να μεγαλώσουμε ή να μικρύνουμε όλο το περιεχόμενο της οθόνης. Η συγκεκριμένη αλλαγή στις νεότερες εκδόσεις του Android γινόταν και χωρίς Root από adb, αλλά και πάλι ο μέσος χρήστης δεν πρέπει να "χρειάζεται" να μπει στην διαδικασία να εγκαθιστά το Android SDK, να κατεβάζει τα Platform Tools από τον SDK Manager, να ορίζει μία νέα μεταβλητή στο Path, να εγκαθιστά τον adb driver της συσκευής του, να ενεργοποιεί το USB Debugging και μετά να γράφει adb shell wm density x. Πλέον η επιλογή στο Accessibility λύνει τα χέρια στον μέσο χρήστη, και διευκολύνει ταυτόχρονα και τον power user.¹⁶

¹⁶ Greece, A. (15 Μαΐου, 15-5-2016). www.greeceandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.greeceandroid.gr/dev/1594-android-n-calibrate-screen-feature>.

2.14 Το Android στον υπολογιστή



Η διαφορά του Android σε σχέση με τα υπόλοιπα mobile λειτουργικά συστήματα είναι ότι αυτό μπορεί να λειτουργήσει και σαν κανονικός υπολογιστής. Ειδικά τώρα που οι συσκευές που κυκλοφορούν έχουν τέτοια ανεκμετάλλευτη "ιπποδύναμη", λογικό είναι το επόμενο βήμα να μας πάει στη σύγκλιση.

Το Remix OS είναι μια ROM βασισμένη στο Android 5.1, η οποία έχει τρεις πολύ μεγάλες διαφορές σε σχέση με αυτές που έχουμε συνηθίσει μέχρι σήμερα:

Ανοίγει πολλές εφαρμογές σε παράθυρα ταυτόχρονα, δίνοντάς μας τη δυνατότητα να κάνουμε πολλά πράγματα μαζί, όπως στον υπολογιστή μας.

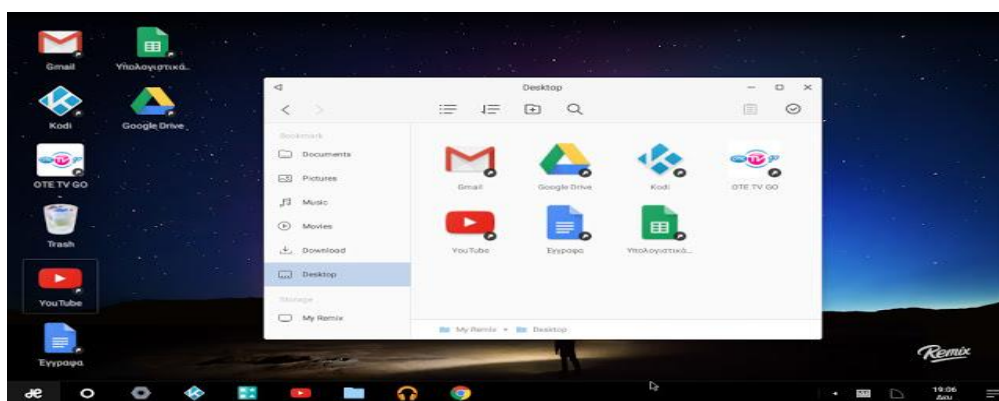
Διαθέτει έναν ειδικά διαμορφωμένο launcher, ο οποίος μας προσφέρει το οικείο περιβάλλον των Windows και είναι σχεδιασμένος για χρήση με πληκτρολόγιο και ποντίκι.

Μπορεί να εγκατασταθεί σε συστήματα με ARM αλλά και x86 αρχιτεκτονικής επεξεργαστή, δηλαδή σε κινητά, tablet, TV-box, αλλά και υπολογιστές.

Έχουμε λοιπόν ένα framework το οποίο μας επιτρέπει να ανοίξουμε εφαρμογές σε παράθυρα. Η λογική του είναι πάρα πολύ απλή και μάλιστα την έχουμε ξαναδεί στο παρελθόν με τα Xperia small apps για παράδειγμα. Κάθε εφαρμογή τσεκάρει την ανάλυση της οθόνης (dpi) πριν ανοίξει, για να προβληθεί σωστά στη συσκευή που έχουμε στα χέρια μας, αλλά στο Remix OS η διαφορά είναι ότι οι εφαρμογές αναγνωρίζουν μόνο ένα μέρος της οθόνης, δηλαδή το μέγεθος του παραθύρου που της δίνεται από το σύστημα.

Το πλεονέκτημα αυτής της μεθόδου είναι ότι μπορούμε να ανοίξουμε και να χρησιμοποιήσουμε πολλές εφαρμογές ταυτόχρονα, όπως για παράδειγμα να γράψουμε μια εργασία στο Google Docs, ενώ βλέπουμε ένα βιντεάκι στο YouTube, αλλά το αρνητικό σε αυτή την υπόθεση είναι ότι πολλές εφαρμογές δεν έχουν φτιαχτεί με αυτό σκεπτικό και έτσι όταν αυξομειώνουμε τα παράθυρά τους, σπάει το γραφικό τους περιβάλλον.

Αυτό το τελευταίο είναι πραγματικά ένα πρόβλημα στη χρήση του Remix OS, ιδιαίτερα στις οθόνες με χαμηλή ανάλυση, καθώς πολλές από τις εφαρμογές ανοίγουν μεν σε παράθυρα αλλά καλύπτουν σχεδόν όλη την επιφάνεια. Αν έχεις όμως οθόνη στα 1080p, μπορείς να έχεις μία εξαιρετική εμπειρία υπολογιστή, απλά ενδέχεται να μην το καταφέρεις εύκολα, καθώς το Remix Mini δεν μπορεί να δώσει περισσότερα από 720p.



Ο launcher είναι ότι ακριβώς θα περιμέναμε από ένα λειτουργικό σύστημα που έχει σχεδιαστεί για το desktop. Έχουμε τη γνωστή taskbar στο κάτω μέρος όπου βρίσκουμε τις εφαρμογές, τα δίκτυα και τις ειδοποιήσεις μας, η επιφάνεια εργασίας μπορεί να φιλοξενήσει συντομεύσεις, υπάρχει κάδος ανακύκλωσης και γενικά όλα έχουν σχεδιαστεί για να εξυπηρετούν τη χρήση με ποντίκι και πληκτρολόγιο. Από άποψη εφαρμογών διαθέτει τα απολύτως απαραίτητα, χωρίς ίχνος bloatware, μεταξύ των οποίων την αριθμομηχανή, τον file manager (ο οποίος προφανώς είναι εμπνευσμένος από το ναυτίλο), ένα media player, μερικά Google apps (όχι στην έκδοση για x86 επεξεργαστές) και την εφαρμογή User Feedback με την οποία μπορούμε να επικοινωνήσουμε με την εταιρία και να αναφέρουμε τυχόν προβλήματα.

Ο launcher κατά τη γνώμη μου είναι ότι καλύτερο έχει να δώσει το Remix OS, καθώς πολλές φορές σε κάνει να ξεχνάς ότι δουλεύεις στο Android και αυτή είναι η μεγάλη του επιτυχία. Δεν έχει απολύτως κανένα ψεγάδι και όλα λειτουργούν τέλεια είτε το χρησιμοποιείς για διασκέδαση, είτε για εργασία.

Μπορώ να αντικαταστήσω τον υπολογιστή μου με αυτό ;

Οι χρήστες που έφυγαν από τα Windows για να πάνε σε Linux και Mac ξέρουν την απάντηση, η οποία είναι "ναι αρκεί να μπορείς να αποχωριστείς κάποια προγράμματα και να τα αντικαταστήσεις με κάποια άλλα". Αν σκεφτούμε τι κάνουμε κάθε μέρα στον υπολογιστή μας, θα έλεγα ότι ο μέσος χρήστης μπαίνει στο ίντερνετ, παίζει παιχνίδια, χρησιμοποιεί εφαρμογές office και κάνει photo και video editing. Για όλα αυτά υπάρχουν ένα σωρό εφαρμογές για να κάνεις τη δουλειά σου, αν όμως θέλεις πιο εξεζητημένα πράγματα, δεν θα μας βολέψει.

Είναι επίσης εξαιρετική λύση για κάποιον που δεν είναι εξοικειωμένος με την τεχνολογία. Σκεφτείτε τους γονείς μας που τους έχουμε πάρει ένα smartphone με το ζόρι για να μπορούμε επιτέλους να μιλάμε δωρεάν μέσω Viber. Αν ποτέ χρειαστούν υπολογιστή αυτοί οι άνθρωποι, θα προτιμήσουν μία λύση σαν το Remix OS, η οποία θα τους φανεί πιο οικεία, λόγω του smartphone.

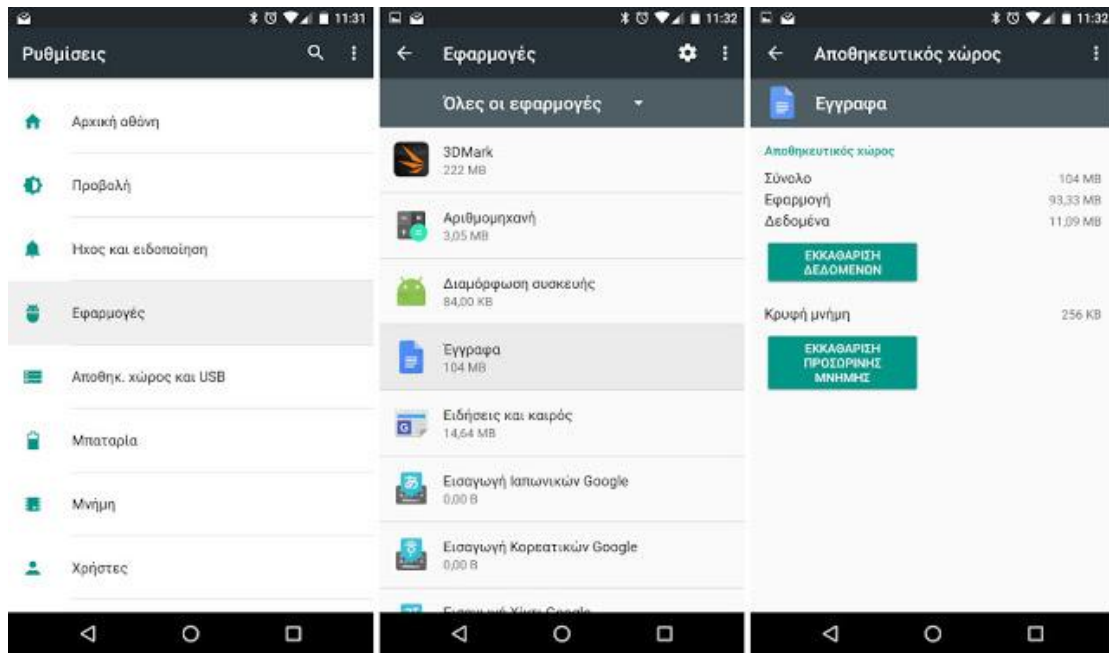
Το πόρισμα

Το Remix OS λοιπόν μου άφησε τις καλύτερες εντυπώσεις, καθώς δεν είχε κανένα πρόβλημα λειτουργίας, με την εξαίρεση ελάχιστων κολλημάτων που είναι λογικές όταν ανοίγεις τέσσερις εφαρμογές, σε μία πολύ μέτρια ARM συσκευή όπως το Remix Mini. Όσον αφορά τους υπολογιστές μας, οι οποίοι είναι πολύ δυνατότεροι, εκεί πιστεύω ότι το Remix OS θα ανθίσει και θα μας δώσει (μας δίνει ήδη δηλαδή) μία εναλλακτική που αρκετοί από εμάς χρειαζόμαστε.¹⁷

2.15 Πως κάνουμε εκκαθάριση δεδομένων

Το Android μας κρύβει πολλά κολπάκια στις ρυθμίσεις του, τα οποία μπορούν να μας λύσουν μικρά ή μεγάλα προβλήματα που μπορεί να έχουμε στην καθημερινή χρήση της συσκευής μας. Ένα από αυτά είναι η εκκαθάριση δεδομένων των εφαρμογών, η οποία μας βοηθάει να επαναφέρουμε κάθε εφαρμογή ξεχωριστά όταν αυτή δε λειτουργεί όπως θα έπρεπε. Κατά καιρούς όλοι έχουμε δει το τρομερό μήνυμα "Αναγκαστική διακοπή" όταν μία εφαρμογή κρασάρει. Δεν είναι κάτι που πρέπει να μας ανησυχεί αν συμβαίνει μια στο τόσο, αλλά σε πολλούς χρήστες αυτό το πρόβλημα είναι μόνιμο και δεν τους επιτρέπει να χρησιμοποιήσουν συγκεκριμένες εφαρμογές. Αυτό γίνεται γιατί όταν ξεκινάμε μία εφαρμογή, αυτή διαβάζει τα αρχεία των προσωπικών μας ρυθμίσεων πριν να ανοίξει, για να κάνει αυτόματα login στο λογαριασμό μας ή για να μας προβάλει το τελευταίο σημείο από όπου την αφήσαμε στην τελευταία χρήση. Αυτά τα αρχεία λοιπόν κάποιες φορές και ιδιαίτερα μετά από αναβάθμιση της εφαρμογής ή του συστήματος, καταστρέφονται. Έτσι η εφαρμογή δεν μπορεί να τα διαβάσει και κρασάρει.

¹⁷ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/03/remix-mini-os-review.html>.



18

2.16 Μεταφορά επαφών, εφαρμογών και αρχείων

Η αγορά ενός καινούργιου smartphone είναι η χαρά του γκατζετάκια, αναπόφευκτα όμως συνοδεύεται και από τη βαρετή διαδικασία της μεταφοράς των επαφών, εφαρμογών και των διαφόρων αρχείων μας από το παλιό στο νέο τηλέφωνο. Στην διάθεση μας όμως έχουμε αρκετά χρήσιμα εργαλεία που κάνουν αυτή τη διαδικασία ευκολότερη και γρηγορότερη είτε η νέα μας συσκευή είναι κάποιο άλλο android smartphone είτε ένα iPhone. Πάμε λοιπόν να δούμε μερικά χρήσιμα tips που θα κάνουν ευκολότερη τη ζωή μας.

1) Επαφές

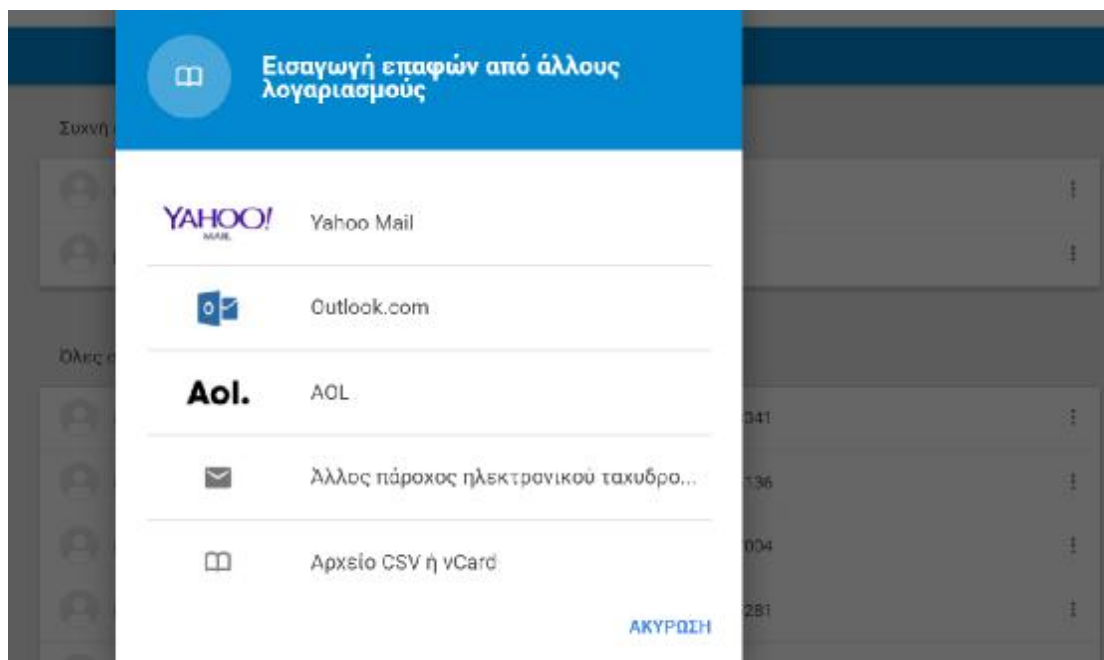
Η μεταφορά των επαφών μας είναι σίγουρα το πιο εύκολο κομμάτι στην όλη διαδικασία του backup. Εφόσον αποθηκεύσετε τις επαφές στον Gmail λογαριασμό σας όλα θα γίνουν αυτόματα. Με το που εισάγετε τον λογαριασμό σας στον νέο τηλέφωνο αυτό θα κατεβάσει αυτόματα τις επαφές σας. Όλα αυτά βέβαια εάν μεταφέρεστε από Android σε Android.

Η μεταφορά από iPhone σε Android ή αντίστροφα έχει μερικές παραπάνω δυσκολίες, αλλά μπορεί να γίνει με διάφορους τρόπους. Ένας από τους πιο απλούς είναι με την χρήση μιας κάρτας microSD. Από την εφαρμογή των επαφών μπορείτε να κάνετε 'εξαγωγή' των επαφών σας στην κάρτα και αντίστοιχα 'εισαγωγή' του αρχείου στο νέο σας τηλέφωνο.

Μία ακόμη επιλογή για μεταφορά από iPhone σε Android είναι η αποθήκευση των επαφών ως CSV αρχεία μέσω του Outlook. Στη συνέχεια μπορείτε να κάνετε

¹⁸ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/04/clear-data-on-apps.html>.

‘εισαγωγή’ το αρχείο από τον Gmail λογαριασμό σας ακολουθώντας τη διαδρομή Gmail -> Επαφές -> Περισσότερα -> Εισαγωγή.



2) Μηνύματα


Δε ξέρω γιατί κάποιος θα ήθελε να μεταφέρει τα παλιά του μηνύματα στο νέο τηλέφωνο, ο ευκολότερος ωστόσο τρόπος για να γίνει αυτό είναι με τη βοήθεια μιας τρίτης εφαρμογής. Το SMS Backup & Restore που θα βρείτε στο Google Play είναι ένα από τα καλύτερα για αυτή τη δουλειά και είναι εντελώς δωρεάν.


3) Φωτογραφίες – Μουσική – Videos

Εδώ τα πράγματα μπορούν να γίνουν πολύ χρονοβόρα ιδιαίτερα αν έχετε μια μεγάλη συλλογή από mp3 ή videos. Η μεταφορά μπορεί να γίνει είτε μέσω bluetooth είτε ανεβάζοντας τα αρχεία σας σε μια υπηρεσία cloud αποθήκευσης όπως το Dropbox και από εκεί στο νέο σας τηλέφωνο. Επιπλέον η Apple προσφέρει την εφαρμογή Copy my Data που θα σας βοηθήσει εφόσον μεταναστεύετε από Android σε Iphone. Ο καλύτερος τρόπος ωστόσο εδώ παραμένει η μεταφορά των αρχείων σας στο PC και από εκεί στο νέο σας τηλέφωνο.¹⁹

¹⁹ Android, H. (29 Μαΐου, 29-5-2016). www.androidhellas.gr. Ανακτήθηκε 11 Ιουνίου, 11-6-2016, από <http://androidhellas.com/2016/04/07/metafora-epafon-efarmogon-arxeion-kainourgio-tilefono/>.

< Media storage

 Pictures Not available

 Movies Not available

Space left Not available

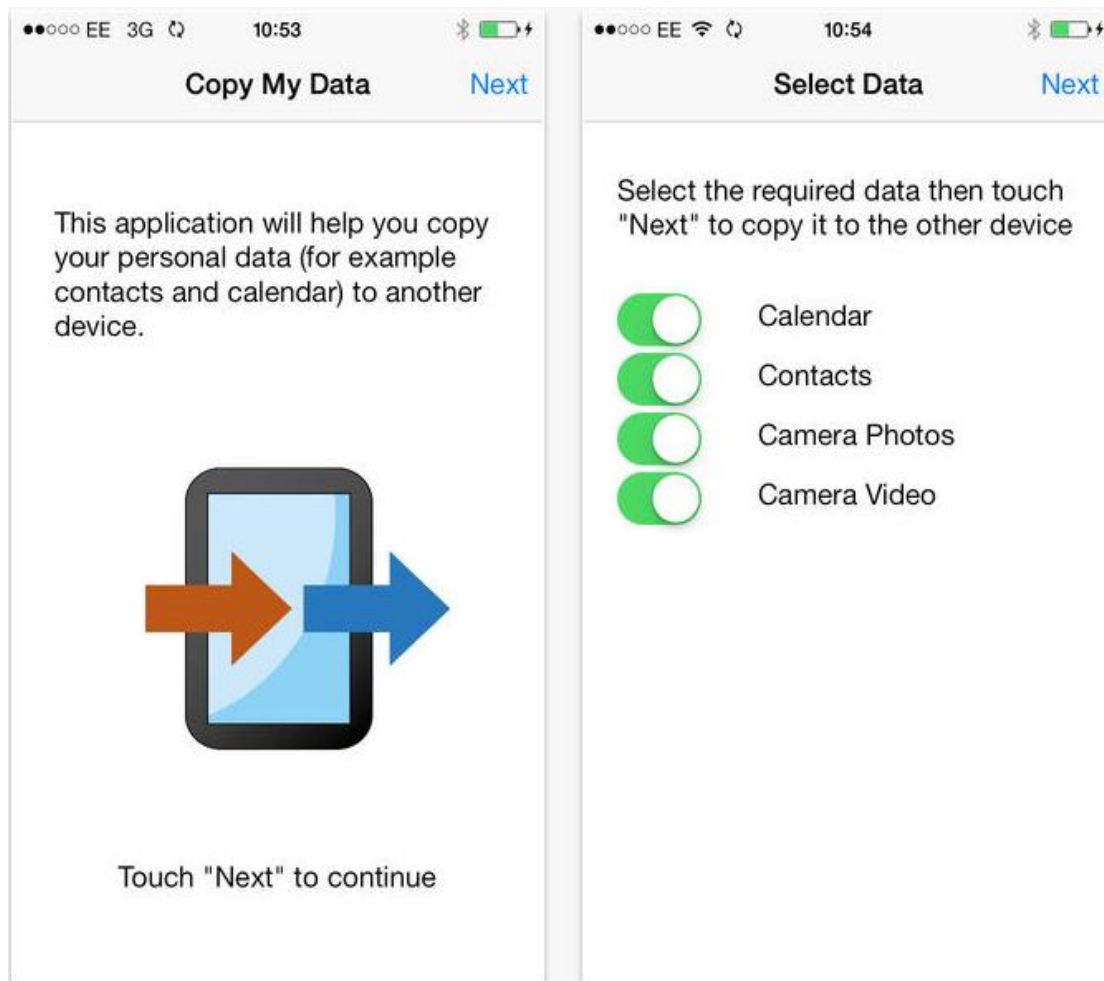
Share using USB 

Advanced

Default media location

Internal

Choose where photos, videos, music and downloads are stored by default.



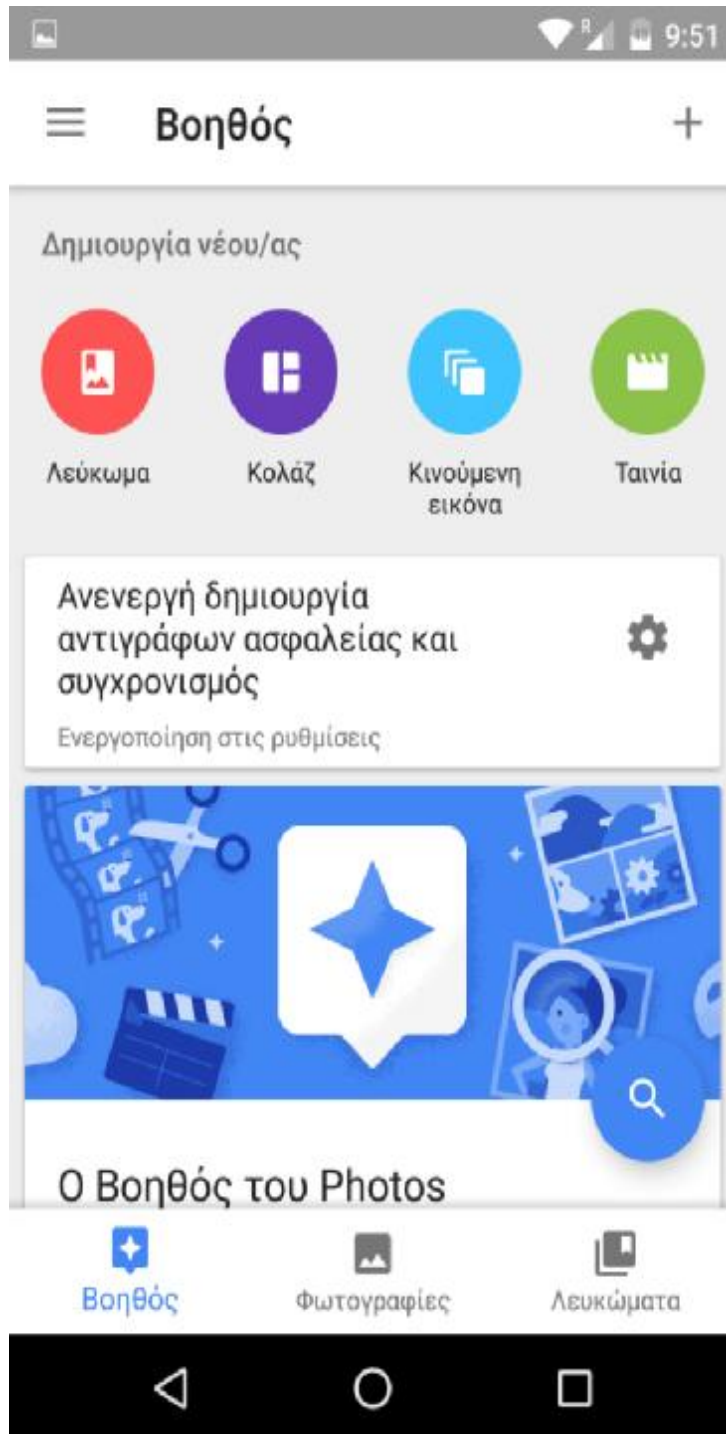
4) Εφαρμογές και τα δεδομένα τους

Αν και το Android προσφέρει πλέον επιλογή για backup και restore των εφαρμογών η καλύτερη λύση εδώ είναι να καταφύγετε σε κάποια τρίτη εφαρμογή. Στο Google Play θα βρείτε δεκάδες τέτοιες εφαρμογές με τα Helium και MyBackup να είναι τα δύο που τουλάχιστον εγώ ξεχωρίζω. Με την βοήθεια τους μπορείτε να αποθηκεύσετε και τα δεδομένα των εφαρμογών/παιχνιδιών σας ώστε να μην χάσετε ρυθμίσεις και saves. Αν πάλι μεταφέρεστε από Android σε iPhone ή αντίστροφα τα χέρια σας είναι δεμένα. Εδώ θα χρειαστεί να οπλιστείτε με αρκετή υπομονή εγκαθιστώντας από την αρχή και μία-μία τις εφαρμογές σας, κάτι που σίγουρα θα είναι αρκετά χρονοβόρο.

2,17 Η Google παρουσιάζει τη νέα “Bottom Navigation Bar” !!!

Η Google ανακοίνωσε εδώ και λίγες ώρες τις νέες “οδηγίες για developers” που αφορούν την σωστή χρήση του Material σχεδιασμού στις εφαρμογές για Android. Εκεί λοιπόν είχαμε μια πραγματική “βόμβα” από την Google παρουσιάζοντας μια καινούργια “Navigation Bar” στο κάτω μέρος των εφαρμογών, κάτι που έως σήμερα αποτελούσε σήμα κατατεθέν των εφαρμογών για iPhones!

Τα πρώτα δείγματα για το πως αυτή θα υλοποιείται υπάρχουν ήδη και μπορείτε να τα δείτε στις νέες εκδόσεις των εφαρμογών Google Photos και Google Plus.



Σε αντίθεση ωστόσο με το iOS, εδώ οι developers θα έχουν τη δυνατότητα να προσθέσουν όσα εικονίδια θέλουν, με την Google να προτείνει από 3 έως 5. Από την άλλη όλα αυτά δε σημαίνουν πως θα καταργηθεί το μενού τύπου 'Hamburger' που χρησιμοποιείται ευρέως στις πιο καινούργιες εφαρμογές. Αντίθετα η Google προτείνει την χρήση του στην περίπτωση που τα πέντε εικονίδια της μπάρας δεν είναι αρκετά για την προβολή όλου του περιεχομένου μιας εφαρμογής.



Σε κάθε περίπτωση η πλοήγηση με την βοήθεια της νέας “Navigation Bar” γίνεται ταχύτερη καθώς θα απαιτείται μόνο ένα κλικ αντί για δύο που θα θέλαμε έως τώρα. Δε μπορούμε όμως να παραβλέψουμε πως η νέα εμφάνιση των εφαρμογών για Android σε μερικούς μήνες θα θυμίζει αρκετά τις αντίστοιχες για iPhones, κάτι που ίσως ξενίσει αρκετούς. Εσείς συμφωνείτε με αυτή την ήδη αμφιλεγόμενη κίνηση της Google;²⁰

2.18 Εναλλαγή εφαρμογών στο Android N

Εδώ και δύο μέρες η Google έχει δημοσιεύσει την πρώτη Developer Preview έκδοση του Android N δίνοντας μας μια καλή γεύση για το τι έρχεται. Το Android N λοιπόν, με βάση τα όσα έχουμε δει ως σήμερα, θα φέρει μπόλικες αλλαγές με την σημαντικότερη από αυτές να είναι ίσως το multi-screen multitasking. Στο εξής θα μπορούμε να τρέχουμε παράλληλα δύο εφαρμογές στην οθόνη της συσκευής μας, κάτι που οι χρήστες των premium συσκευών της Samsung απολάμβαναν εδώ και χρόνια. Το αρνητικό στην όλη υπόθεση είναι πως οι developers θα πρέπει να προσθέσουν αυτή την δυνατότητα στις εφαρμογές, διαφορετικά δε θα το υποστηρίξουν. Έτσι θα χρειαστεί σίγουρα ένα σεβαστό χρονικό διάστημα για να δούμε το 100% των δυνατοτήτων του split-screen multitasking. Στο παρακάτω video μπορείτε να δείτε πως θα λειτουργεί όλο αυτό στην πράξη. Παράλληλα η Google έχει αλλάξει και την εμφάνιση των πρόσφατα χρησιμοποιημένων εφαρμογών. Οι κάρτες των εφαρμογών είναι τώρα μεγαλύτερες με το τελικό αποτέλεσμα να δείχνει αρκετά ομορφότερο. Όμως και εδώ έχουμε μια μικρή αλλά χρήσιμη προσθήκη. Το διπλό tap στο εικονίδιο των πρόσφατων εφαρμογών θα μας μεταφέρει απευθείας στην τελευταία εφαρμογή που χρησιμοποιήσαμε, κάτι που σε κάποιους ενδέχεται να φανεί αρκετά βολικό.²¹

2.19 Rom Windows 10 για Android συσκευές

Σύμφωνα με ανακοίνωση της ίδιας της εταιρίας, η Microsoft σε συνεργασία με την Xiaomi, ετοιμάζει μια ROM, η οποία θα επιτρέπει στους κατόχους συσκευών της κινέζικης εταιρίας (Αρχικά) να δοκιμάσουν τα Windows 10 στις Android συσκευές τους. Σύμφωνα με το Techcrunch, η rom αυτή θα είναι μια αρκετά διαφορετική προσέγγιση της MS από τη μέχρι τώρα διασπαρτη κυκλοφορία των Windows Apps στο Google Play, αφού θα μετατρέπει το εκάστοτε τηλέφωνο, σε μια πλήρως λειτουργική Windows Phone συσκευή. Η επιλογή της Xiaomi από τη Microsoft ως πρώτο πειραματόζωο, έγινε λόγω του μεγάλου όγκου feedback που δέχεται η εταιρία καθημερινά η Κινέζικη εταιρία από τους χρήστες των συσκευών της, πολύ σύντομα όμως αν όλα πάνε καλά, η ROM αυτή θα τροποποιηθεί ώστε να λειτουργεί σε περισσότερες συσκευές. Και αν δεν το κάνει η Microsoft, το XDA να'ναι καλά. Πριν

²⁰ Android, H. (29 Μαΐου, 29-5-2016). www.androidhellas.gr. Ανακτήθηκε 11 Ιουνίου, 11-6-2016, από <http://androidhellas.com/2016/03/15/android-material-nea-bottom-navigation-bar/>.

²¹ Android, H. (29 Μαΐου, 29-5-2016). www.androidhellas.gr. Ανακτήθηκε 11 Ιουνίου, 11-6-2016, από <http://androidhellas.com/2016/03/11/split-screen-multitasking-android-n/>.

ξεφορτωθείτε την παλιά συσκευή Android σας λοιπόν, κάντε λίγο υπομονή, αφού τα Windows 10 ίσως δώσουν νέα πνοή στο ταλαιπωρημένο σας τηλέφωνο. Ίσως πάλι και όχι.²²

2.20 Τι είναι ο Bootloader ;

Οι Android συσκευές αποτελούνται από πολλά partition. Ένα περιέχει την κονσόλα recovery, άλλο τα αρχεία συστήματος Android και άλλο τα δικά μας αρχεία. Ο κλειδωμένος Bootloader, είναι μια δικλείδα ασφαλείας η οποία δεν μας επιτρέπει να επέμβουμε πάνω σε αυτά τα partition, άρα δεν μπορούμε να εγκαταστήσουμε μια άλλη recovery. Σε τι εξυπηρετεί ο κλειδωμένος Bootloader; Εξυπηρετεί τις εταιρίες κινητής τηλεφωνίας και τα καταστήματα πώλησης των συσκευών, διότι δε θέλουν σε καμία περίπτωση να καλύπτουν τις συσκευές στις οποίες επεμβαίνουμε εμείς.

Η ιδιαιτερότητα του Android είναι ότι πρόκειται για λειτουργικό ανοιχτού κώδικα. Για αυτό το λόγο, πολύς κόσμος ασχολήθηκε με την ανάπτυξή του και έτσι είδαμε την πληθώρα custom ROM που κυκλοφορούν στο ίντερνετ. Οι συσκευές όμως συνέχιζαν και συνεχίζουν να έρχονται με κλειδωμένους και κρυπτογραφημένους Bootloader πράγμα που ώθησε την κοινότητα να διαμαρτυρηθεί απέναντι στις εταιρίες κατασκευής συσκευών και να απαιτήσει ευκολότερο ξεκλείδωμα. Η αρχή έγινε με τις συσκευές Nexus οι οποίες είναι κλειδωμένες αλλά τουλάχιστον ο Bootloader τους δεν είναι κρυπτογραφημένος πράγμα που σημαίνει ότι με την απλή εντολή “fastboot oem unlock” μέσα από τερματικό μπορούμε να τον ξεκλειδώσουμε. Το ίδιο συμβαίνει για τις περισσότερες συσκευές της Samsung αλλά και της LG. Οι υπόλοιποι κατασκευαστές μη θέλοντας να χάσουν το τρένο των εξελίξεων, έφτιαξαν επίσημα εργαλεία για το ξεκλείδωμα των δικών τους συσκευών με παραδείγματα:

[τον τρόπο της HTC](#)

[τον τρόπο της Sony](#)

[τον τρόπο της Motorola](#)

[τον τρόπο της Asus](#)

Αν λοιπόν δεν μπορείτε να βρείτε καμία μέθοδο για να ξεκλειδώσετε τον Bootloader της συσκευής σας και κατ' επέκτασιν να κάνετε root, μπορείτε να δοκιμάσετε την εντολή “fastboot oem unlock” όπως περιγράφεται στο βίντεο παρακάτω. Αν είναι μη κρυπτογραφημένος και δε σας βγάλει σφάλμα, μπορείτε να συνεχίσετε και στα υπόλοιπα βήματα του βίντεο για πλήρη δικαιώματα υπερχρήστη στη συσκευή σας.²³

²² Android, A. (26 Μαΐου, 26-5-2016). www.android.gr. Ανακτήθηκε 11 Ιουνίου, 11-6-2016, από <https://www.android.gr/android-news/logismiko/i-microsoft-etimazi-rom-pou-tha-feri-ta-windows-10-stis-android-siskeves>.

²³ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2012/02/android-basics-bootloader.html>.

2.21 Τι είναι και πως κάνω S-OFF .

Το S-ON (σημαίνει Security On) είναι μια δικλείδα ασφαλείας την οποία έχει προσθέσει η HTC στις Android συσκευές της. Όπως ξέρουμε το Android είναι στην ουσία μια Linux διανομή, στην οποία ο χρήστης δεν μπορεί να επέμβει σε μερικά κομμάτια του συστήματος. Ένα από αυτά είναι η μνήμη NAND, η οποία είναι το πιο πολύτιμο μέρος στη συσκευή μας. Αν διαγραφεί η μνήμη NAND η συσκευή αχρηστεύεται και δεν μπορούμε να κάνουμε τίποτα χωρίς τα περιεχόμενά της (Υπάρχουν φυσικά τρόποι για να την ξαναφλασάρουμε).

Τι κάνει το S-ON;

Προστατεύει τη μνήμη NAND μη επιτρέποντάς μας να “γράψουμε” πάνω της.

Ελέγχει αν τα αρχεία .img είναι υπογεγραμμένα πριν τα εγκαταστήσουμε.

Δεν επιτρέπει την αντικατάσταση του HBOOT.

Γιατί όμως μερικές εφαρμογές που λειτουργούν σε άλλες συσκευές μόνο με root, σε αυτές της HTC προϋποθέτουν και S-OFF; Η απάντηση είναι ότι το S-ON εκτός από τη NAND (που καλά κάνει και την προστατεύει), αρνείται την πρόσβαση και σε μερικά ακόμα partition, στα οποία θέλουν να μπουν αυτές οι εφαρμογές. Ο μόνος λόγος για να κάνετε S-OFF λοιπόν, είναι επειδή θέλετε να εγκαταστήσετε μια τέτοια εφαρμογή και τίποτε άλλο. Μπορείτε να αλλάξετε ROM και Recovery μόνο με το root ,οπότε το επιχειρούμε μόνο αν είναι απολύτως απαραίτητο!

Οι καλύτεροι τρόποι για S-OFF :

Το [Android N-ify](#) είναι Το νέο Xposed module που φέρνει τα χαρακτηριστικά του Android N σε παλιότερες συσκευές

Η Google μας έδειξε πρόσφατα τι σκέφτεται να κάνει με το Andorid N, παρουσιάζοντάς μας οκ ολίγα χαρακτηριστικά τα οποία ενδέχεται να συνοδεύσουν τη νέα έκδοση. Κάποιοι δαιμόνιοι προγραμματιστές όμως κατάφεραν να δημιουργήσουν ένα Xposed module, το οποίο φέρνει μερικά από αυτά στις συσκευές μας σήμερα.

Το [Xposed framework](#) αποτελεί το υπέρτατο hack των τελευταίων ετών στο Android και μας επιτρέπει να εγκαταστήσουμε επιπλέον στοιχεία στη ROM μας, χωρίς να την αλλάξουμε και έτσι το συγκεκριμένο module έχει την ίδια λογική. Αν το εγκαταστήσουμε σε συσκευές με Android 5.0+, μας δίνει τη δυνατότητα να έχουμε δύο μικρά αλλά πολύ χρήσιμα χαρακτηριστικά του Android N developer preview, όπως τους υπότιτλους στο μενού των ρυθμίσεων και το άνοιγμα της προηγούμενης εφαρμογής μας, κάθε φορά που πατάμε δύο φορές την "επισκόπηση" (το τετράγωνο).

Είναι σίγουρα κάτι πολύ μικρό σε σχέση με όλα όσα είδαμε στο preview, αλλά ο δημιουργός μας έχει αφήσει με υποσχέσεις και για άλλες προσθήκες, μεταξύ των οποίων:

το πλαϊνό μενού στις ρυθμίσεις,

τη χρήση του πλήκτρου "Επισκόπηση" για να βλέπουμε μία μία τις τελευταίες μας εφαρμογές,

την ενεργοποίηση του Doze ακόμη και όταν η συσκευή μας κινείται,

το σκούρο theme του Night mode,

το νέο μενού ειδοποιήσεων και

τα στοιχεία μας στη lockscreen όταν είμαστε σε κίνδυνο.

Θα το βρείτε όπως πάντα στο μενού του Xposed με τον τίτλο "Android N-ify", αλλά και στο νήμα του στο XDA.²⁴

Το "fastboot flashing" είναι η νέα εντολή για να ξεκλειδώσουμε τον bootloader μας, δείτε σε ποιες περιπτώσεις ισχύει.

Το Hacking στο Android είναι κατά κάποιο τρόπο μέρος της εμπειρίας χρήσης του, αλλά η έννοια του hacking δεν συμβαδίζει με αυτή της ασφάλειας. Η Google λοιπόν είναι υποχρεωμένη να κάνει το παν για να μας προστατεύσει από κάθε λογής κινδύνους στη δική μας ROM τουλάχιστον, αλλά επειδή δεν είναι συμβατική εταιρία, κάνει βήματα για να προστατέψει οσους χακάρουν τα μοντέλα τους.

Στην τελευταία έκδοση του του Fastboot λοιπόν, η γνωστή εντολή "fastboot oem unlock" με την οποία μπορούσαμε να ξεκλειδώσουμε έναν μη κρυπτογραφημένο bootloader, όπως αυτόν των Nexus, αλλάζει και αντικαθίσταται από τη "fastboot flashing unlock". Πιο συγκεκριμένα η Google θέλησε να «σπάσει» το ξεκλείδωμα του bootloader σε δύο μέρη και πρόσθεσε τις εξής εντολές στο ρεπερτόριο του :

fastboot flashing unlock

fastboot flashing lock

fastboot flashing unlock_critical

fastboot flashing lock_critical

fastboot flashing get_unlock_ability

Το "fastboot flashing unlock" το οποίο στην ουσία αντικαθιστά το "fastboot oem unlock" μας δίνει τη δυνατότητα να ξεκλειδώσουμε τον bootloader μέχρι ένα σημείο. Αν χρησιμοποιήσουμε αυτή την εντολή, θα μπορούμε να αναβαθμίσουμε τον επίσημο bootloader, να εγκαταστήσουμε μία custom recovery ή έναν custom kernel και να αλλάξουμε το "system.img" της συσκευής. Μπορούμε δηλαδή να κάνουμε το σύνολο των εργασιών που χρειάζεται ένας μέσος power user στο Android.

²⁴ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/03/android-n-ify-xposed.html>

Η διαφορά τώρα της εντολής "fastboot flashing unlock_critical" είναι ότι μας δίνει τη δυνατότητα να ξεκλειδώσουμε τον bootloader με τέτοιο τρόπο που μας επιτρέπει να τον αντικαταστήσουμε με έναν custom bootloader. Αυτό μπορεί να χρησιμοποιηθεί από δημιουργούς ROM αλλά και από κατασκευαστές συσκευών που θέλουν να ορίσουν νέες δικλίδες ασφαλείας για να κάνουν τις συσκευές τους ακόμη πιο ασφαλείς.

Τέλος έχουμε και το "fastboot flashing get_unlock_ability" για το οποίο δεν υπάρχουν προς το παρόν πληροφορίες, αλλά φαίνεται πως πρόκειται για μία εντολή που είτε μας βοηθάει να ελέγξουμε αν έχουμε κρυπτογραφημένο bootloader, είτε μας εξυπηρετεί στο να τον ξεκλειδώσουμε δίνοντας το ανάλογο κλειδί.

Τι αλλάζει στο μέλλον ;

Η πρώτη συσκευή που αντιδρά μόνο στην εντολή "fastboot flashing unlock" και όχι στην "fastboot oem unlock" είναι το Nexus 6P, ενώ το Nexus 5X "ακούει" και στα δύο. Εφόσον το Fastboot θα συνεχίσει να εξυπηρετεί και τις δύο, είναι στην ευχέρεια του εκάστοτε κατασκευαστή να επιλέξει ποια θέλει, εκτός φυσικά αν απαιτηθεί από τη Google. Το σίγουρο πάντως είναι ότι τα πράγματα αλλάζουν σε πολύ περισσότερους τομείς από ότι γνωρίζαμε και είναι προς το συμφέρον μας.²⁵

2.22 Αλλαγή χρωμάτων για το Android 6.0

Το Android 6.0 είναι φαινομενικά μία έκδοση που βελτιώθηκε κυρίως εσωτερικά, αφού η μεγάλη αλλαγή στην εμπειρία χρήσης ήρθε με την έκδοση Lollipop. Η Google όμως "έκρυψε" μέσα της κάποια πολύ ενδιαφέροντα χαρακτηριστικά όπως το dual window. Πρόσφατα ανακαλύψαμε ότι υπάρχει και ενσωματωμένο theme engine, με το οποίο μπορούμε να αλλάξουμε τα χρώματα της stock έκδοσης με ευκολία.

Όλα ξεκίνησαν από ένα framework (RRO) που δημιούργησε η Sony για τις συσκευές της, το οποίο έχει ενσωματωθεί πλέον στον κώδικα του Android και επιτρέπει την εύκολη εφαρμογή "θεμάτων" σε όλο το σύστημα. Αυτό εκμεταλλεύτηκε ο δημιουργός της εφαρμογής Layers Manager, με την οποία μπορούμε να "πειράξουμε" τα προεπιλεγμένα χρώματα και τα εικονίδια του Android, με το λεγόμενο "RRO theming".

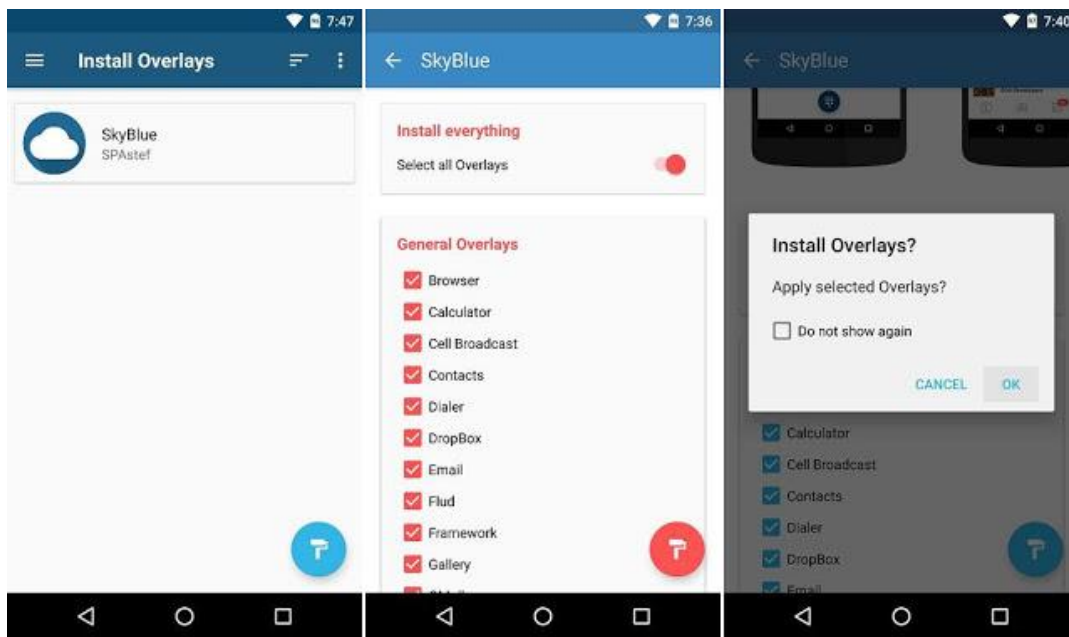
Για να το χρησιμοποιήσουμε θα πρέπει:

Να έχουμε κάνει root στη συσκευή μας

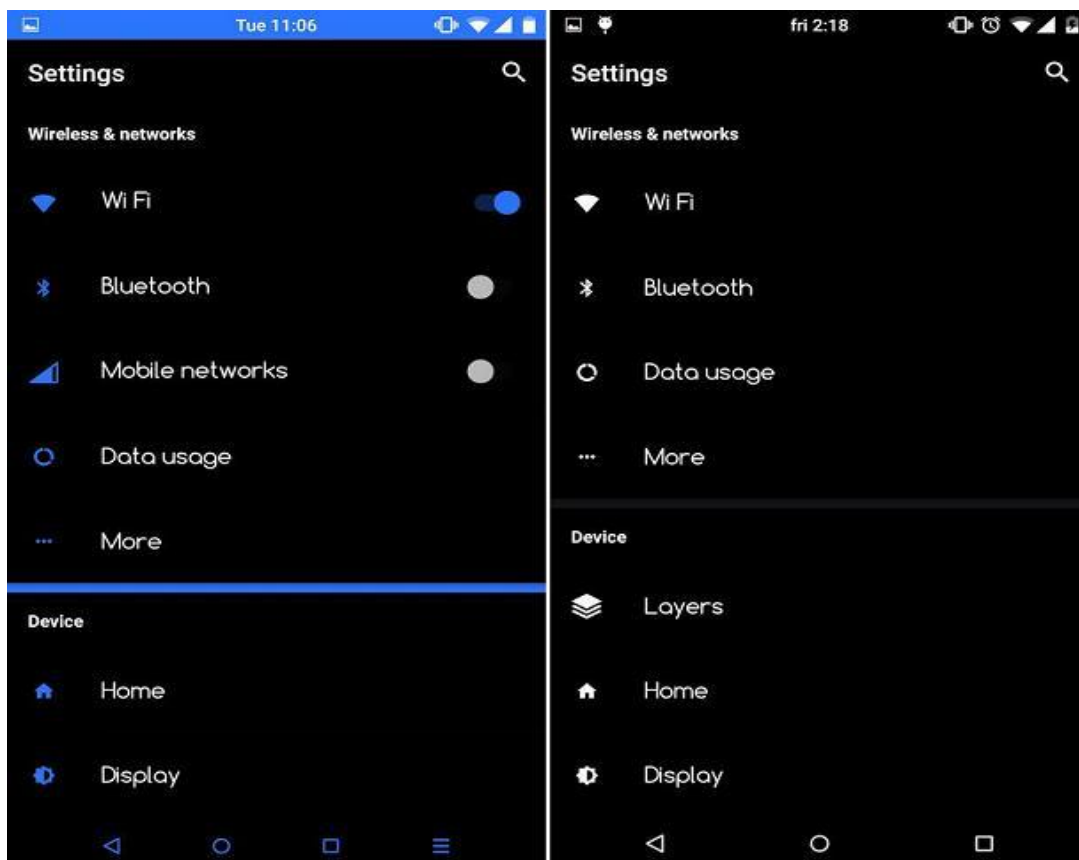
Να έχουμε εγκαταστήσει το Busybox και φυσικά

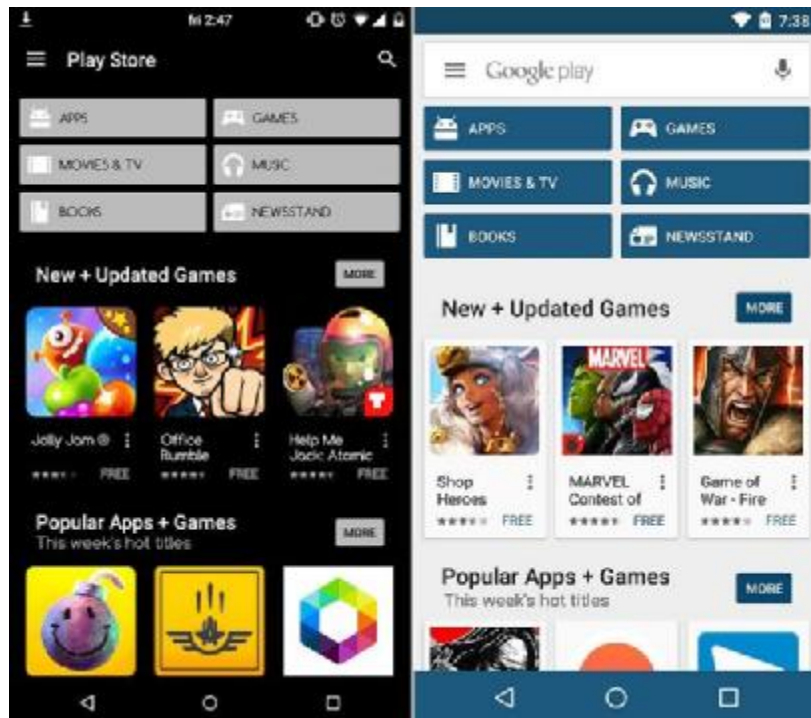
²⁵ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2015/11/fastboot-flashing-command.html>.

Να έχουμε κατεβάσει το Layer Manager από το Play Store.



Στη συνέχεια θα πρέπει να κατεβάσουμε έτοιμα theme μέσα από το Layer Manager, με τα οποία μπορούμε να "πειράξουμε" τα εικονίδια και τα χρώματα στη πανbar, στη μπάρα ειδοποιήσεων, στο launcher, στο μενού των ρυθμίσεων, αλλά και σε κάποιες υποστηριζόμενες εφαρμογές, κυρίως προεγκατεστημένες.





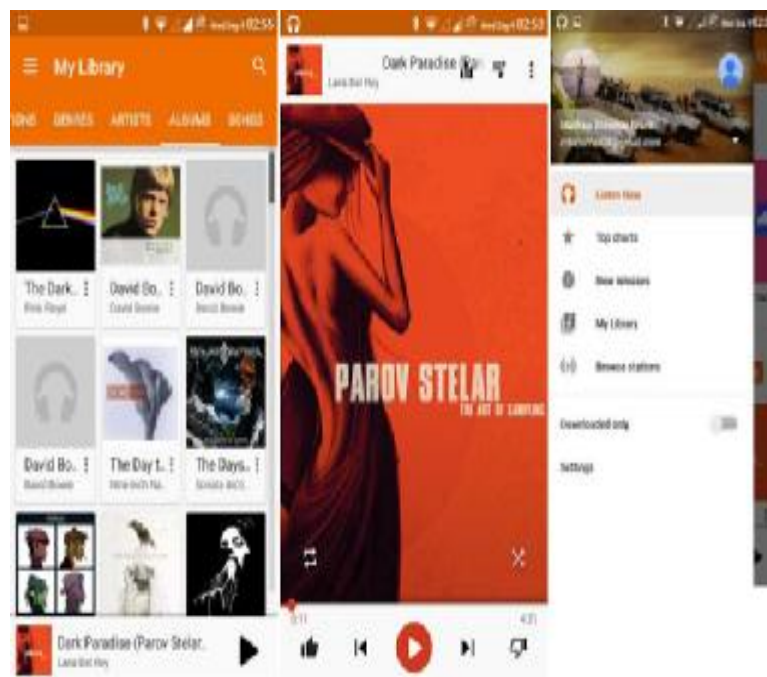
Υπάρχει ήδη πολύ μεγάλη ποικιλία θεμάτων στο Play Store που υποστηρίζουν το Layers Manager, με τα περισσότερα από αυτά να είναι και δωρεάν, έτσι θα βρούμε σίγουρα κάτι που θα μας αρέσει. Να κάτι ακόμη που έλειπε από τη stock έκδοση του Android και βλέπουμε ότι η Google μας το φέρνει σιγά σιγά ως προεγκατεστημένο, ελπίζουμε σύντομα και χωρίς root.²⁶

2.23 Χειριστείτε την μουσική σας μέσα από το Google Music

²⁶ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2015/11/android-60-theme-engine-rro.html>.

Μπορεί το Android να έχει κάνει μεγάλα βήματα προόδου αλλά οι ανεξάρτητοι προγραμματιστές δεν έχουν σταματήσει να δημιουργούν μικρά πολύτιμα κομμάτια κώδικα που προσθέτουν χαρακτηριστικά στη ROM μας. Ένα από αυτά είναι το XGPM που φέρνει νέα χαρακτηριστικά στην εφαρμογή του Google Play Music.

Εγκαθιστώντας λοιπόν το XGPM μέσα από το repository του Xposed framework κερδίζουμε πρώτα από όλα τη διαχείριση της playlist μέσα από τις ειδοποιήσεις. Στο κλασικό widget που εμφανίζεται εκεί όταν ακούμε μουσική, έχει προστεθεί ένα νέο πλήκτρο το οποίο μας εμφανίζει τα επόμενα έξι κομμάτια, σε περίπτωση που θέλουμε να προπεράσουμε κάποιο από αυτά. Πολύ βολικό γιατί μέχρι σήμερα έπρεπε να ξεκλειδώσουμε τη συσκευή μας και να ανοίξουμε την εφαρμογή για να το βρούμε.



Δε σταματάμε όμως εκεί καθώς το συγκεκριμένο Module:

ορίζει την καρτέλα " Η βιβλιοθήκη μου" ως κεντρική σελίδα και όχι το "Ακούστε τώρα",

εμφανίζει τρεις στήλες με άλμπουμ (αντί δύο) για να βλέπουμε ακόμη περισσότερες πληροφορίες,

διορθώνει τα album art τα οποία "πετσοκόβονται" στον player,

εμφανίζει το πλήκτρο του equalizer για να έχουμε άμεση πρόσβαση από τον player,

κρύβει κάποια λιγότερο χρήσιμα πλήκτρα από το πλαϊνό πάνελ και

αλλάζει τα χρώματα στα βασικά πλήκτρα (όπως το play) με μία απόχρωση που ταιριάζει στο album art του κομματιού που ακούμε. Δεν είναι και ο σημαντικότερος λόγος για να κάνεις root και να εγκαταστήσεις το Xposed framework, αλλά για

όσους το έχουν ήδη και χρησιμοποιούν το Google Music για να ακούν τη μουσική τους, είναι απαραίτητο.²⁷

2.24 Έκδοση Android για υπολογιστές Remix OS

Έχει μαλλιάσει η γλώσσα μας να λέμε ότι το Android πρέπει να κάνει τη γενναία μετάβαση στους υπολογιστές και ο μόνος τρόπος για να γίνει αυτό είναι να δημιουργηθεί ένα γραφικό περιβάλλον που θα ταιριάζει περισσότερο στη χρήση πληκτρολογίου και ποντικιού. Εφόσον λοιπόν η Google δεν ενώνει το Chrome OS με το Android να τελειώνουμε, δύο πρώην εργαζόμενοι της το ετοίμασαν και το προσφέρουν δωρεάν.

Μιλάμε φυσικά για το Remix OS που παρουσιάσαμε παλιότερα μαζί με το Remix Mini, τη demo συσκευή που έφτιαξε η εταιρία Jide για να μας αποδείξει ότι το λειτουργικό τους δουλεύει. Το Remix OS βασίζεται στην πρώτη μεγάλη προσπάθεια που έγινε από το Android-x86 project, που είχε ως σκοπό τη μεταφορά του λειτουργικού σε επεξεργαστής αρχιτεκτονικής x86. Χρησιμοποιεί ως βάση το Android 5.1 Lollipop και διαθέτει ένα ειδικό framework, το οποίο μας επιτρέπει να ανοίξουμε ταυτόχρονα πολλές εφαρμογές σε παράθυρα.

Επειδή έχουμε εδώ και μερικές εβδομάδες το Remix Mini, αξίζει να πούμε ότι σε αυτή τη συσκευή με Allwinner τετραπύρηνο επεξεργαστή που κάνει μόνο 20.000 πόντους στο Antutu, το Remix ανοίγει τρία με τέσσερα παράθυρα απλά με μία μικρή καθυστέρηση, ενώ παίζει και 4K βίντεο με άνεση. Φανταστείτε λοιπόν τι μπορεί να κάνει σε έναν υπολογιστή με Intel Core i7, όπου σύμφωνα με τη Jide το σκορ του στο Antutu είναι 214.000 πόντοι!



Η πρώτη έκδοση για προγραμματιστές είναι διαθέσιμη στην επίσημη σελίδα του και μπορεί να εγκατασταθεί παράλληλα σε υπολογιστές με Mac, Linux και Windows,

²⁷ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2015/09/xposed-google-music-xgpm.html>.

ενώ μπορούμε αν θέλουμε να το έχουμε και σε ένα USB στικάκι για να το παίρνουμε πάντα μαζί μας.

Η εγκατάσταση είναι απλή:

Κατεβάζουμε στον υπολογιστή μας το Remix OS και το αποσυμπιέζουμε.

Το εγκαθιστούμε σε ένα USB stick με το Remix OS USB Tool.

Κάνουμε επανεκκίνηση τον υπολογιστή μας.

Μπαίνουμε σε boot mode (Συνήθως πατάμε F9 ή F12 ή Del πριν μπούμε στο Windows).

Επιλέγουμε το "USB".

Μόλις boot-άρει, επιλέγουμε "Guest Mode" ή "Resident Mode" ανάλογα με το αν θέλουμε απλά να το δοκιμάσουμε ή να το χρησιμοποιούμε κανονικά από το στικάκι.²⁸

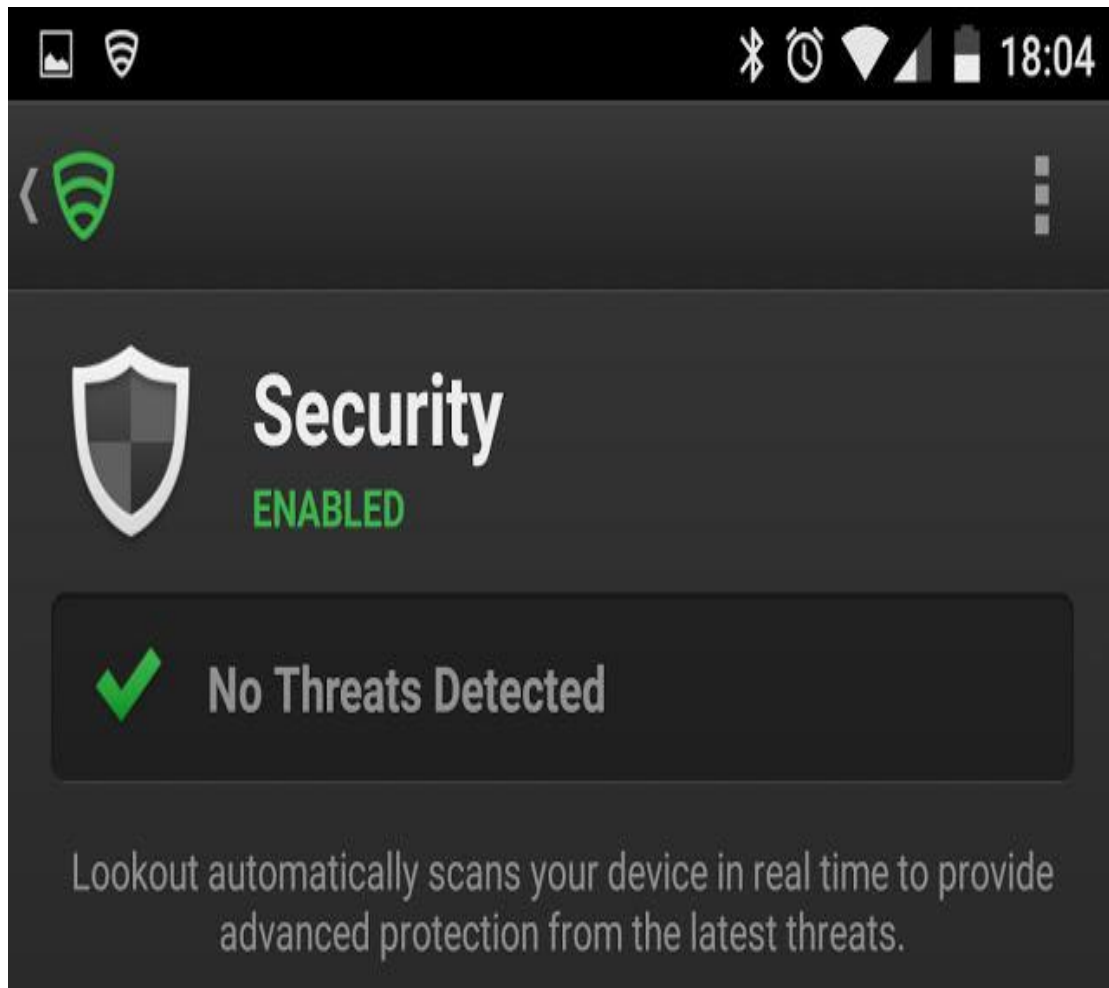
2.25 Αφαίρεση Ιού από το Smartphone ή tablet

Ιός, malware, spyware, είναι μερικά από τα προγράμματα που δε θέλουμε να έχουμε με τίποτα στον υπολογιστή αλλά και στο κινητό μας. Η ιδέα και μόνο ότι τα δεδομένα μας βρίσκονται στα χέρια κάποιου αγνώστου ή ότι κάποιος μας παρακολουθεί χωρίς να το γνωρίζουμε είναι το λιγότερο αγχωτική και έτσι καλό είναι να γνωρίζουμε μερικές απλές λύσεις, για να αντιμετωπίσουμε τέτοιες καταστάσεις αν ποτέ μας έρθουν.

Πώς "κολλάει" ο ιός στο Android;

Στο 99,9% των περιπτώσεων που θα βρεις ιό στο κινητό σου ευθύνεσαι εσύ, καθώς ο ιός έρχεται συνήθως στη μορφή apk που μόνο από εσένα μπορεί να εγκατασταθεί στη συσκευή. Στην περίπτωση που έχεις κάνει root η κακόβουλη εφαρμογή μπορεί να έχει ενσωματωμένο ένα script, το οποίο θα δώσει αυτόματες εντολές για να εγκατασταθεί στη συσκευή σου, αλλά για αυτό το λόγο υπάρχει το SuperSU, ο "κλειδοκράτορας" του root που θα σε ενημερώσει αν κάποια εφαρμογή διαθέτει ύποπτο κώδικα. Μία ακόμη δικλείδα ασφαλείας είναι επίσης τα antivirus, τα οποία σκανάρουν τον κώδικα κάθε νεοεγκατεστημένης εφαρμογής για να μας προστατεύσουν, δυστυχώς όμως έχει αποδειχθεί ότι αδυνατούν να αναγνωρίσουν όλων των ειδών τα λογισμικά. Το Android όμως διαθέτει και άλλα δύο "κρυφά" antivirus, τα οποία τρέχουν στους server της Google, το ένα για να σκανάρει μονίμως το Play Store και τις εφαρμογές του και το άλλο για να σκανάρει τις εφαρμογές της συσκευής μας, στέλνοντας από αυτή ανώνυμα δεδομένα στη Google. Είναι "σφιχτά" λοιπόν τα πράγματα και δε χρειάζεσαι antivirus αν εγκαθιστάς εφαρμογές μόνο από το Play Store ή τέλος πάντων από πηγές τις οποίες εμπιστεύεσαι.

²⁸ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/01/remix-os-android.html>.



Ναι αλλά ο ιός μπήκε τι κάνω;

Έστω λοιπόν ότι από απροσεξία έχουμε εγκαταστήσει μία τέτοιου είδους εφαρμογή, την οποία δεν έχει "πιάσει" καμία από τις προαναφερθείσες μεθόδους ασφαλείας. Τι κάνουμε σε αυτή την περίπτωση; Απλά αφαιρούμε το κακόβουλο λογισμικό, θα πρέπει όμως πρώτα να το αναγνωρίσουμε.

Αρχικά πηγαίνουμε στη διαδρομή "Ρυθμίσεις > Εφαρμογές" και τσεκάρουμε τη λίστα για να δούμε αν υπάρχει κάποια εφαρμογή την οποία δε γνωρίζουμε. Σίγουρα εκεί θα βρούμε ένα σωρό εφαρμογές συστήματος οι οποίες δεν ξέρουμε τι κάνουν, αλλά για αυτές μπορούμε να δούμε σε άλλη συσκευή αν υπάρχουν ή να ρωτήσουμε σε κάποια κοινότητα στο ίντερνετ. Μπορούμε επίσης να πάμε "Ρυθμίσεις > Μπαταρία" για να δούμε αν κάποια εφαρμογή καταναλώνει πόρους, αφού λογικά λειτουργεί μόνιμα στο παρασκήνιο, "Ρυθμίσεις > Χρήση δεδομένων" αφού θα πρέπει να στέλνει τις πληροφορίες που υποκλέπτει σε έναν εξωτερικό server και τέλος (στο Android 6.0)

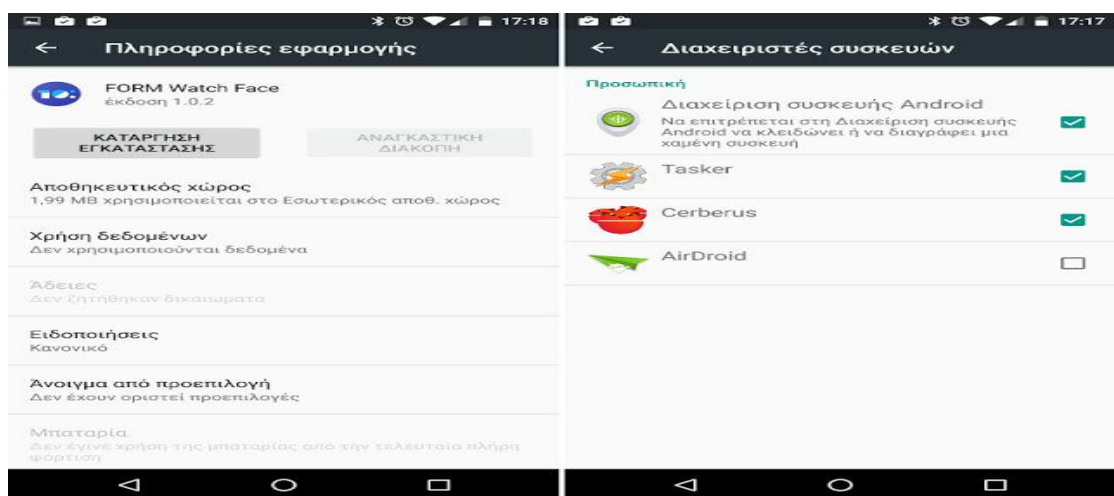
"Ρυθμίσεις > Μνήμη" γιατί σίγουρα θα χρησιμοποιεί και ένα μέρος της RAM.



Λύση πρώτη:

Απεγκατάσταση

Αφού λοιπόν αναγνωρίσαμε την εφαρμογή, τώρα θα πρέπει να την απεγκαταστήσουμε και αυτό γίνεται εύκολα στο "Ρυθμίσεις > Εφαρμογές", επιλέγοντάς την και πιέζοντας "Κατάργηση εγκατάστασης". Επειδή όμως μιλάμε για κακόβουλη εφαρμογή που δε θέλει να απεγκατασταθεί, σε πολλές περιπτώσεις το πλήκτρο της απεγκατάστασης είναι απενεργοποιημένο. Αυτό συμβαίνει διότι από μόνη της η εφαρμογή έχει λάβει δικαιώματα διαχειριστή, τα οποία μπορούμε να απενεργοποιήσουμε εύκολα πηγαίνοντας "Ρυθμίσεις > Ασφάλεια > Διαχειριστές συσκευών".

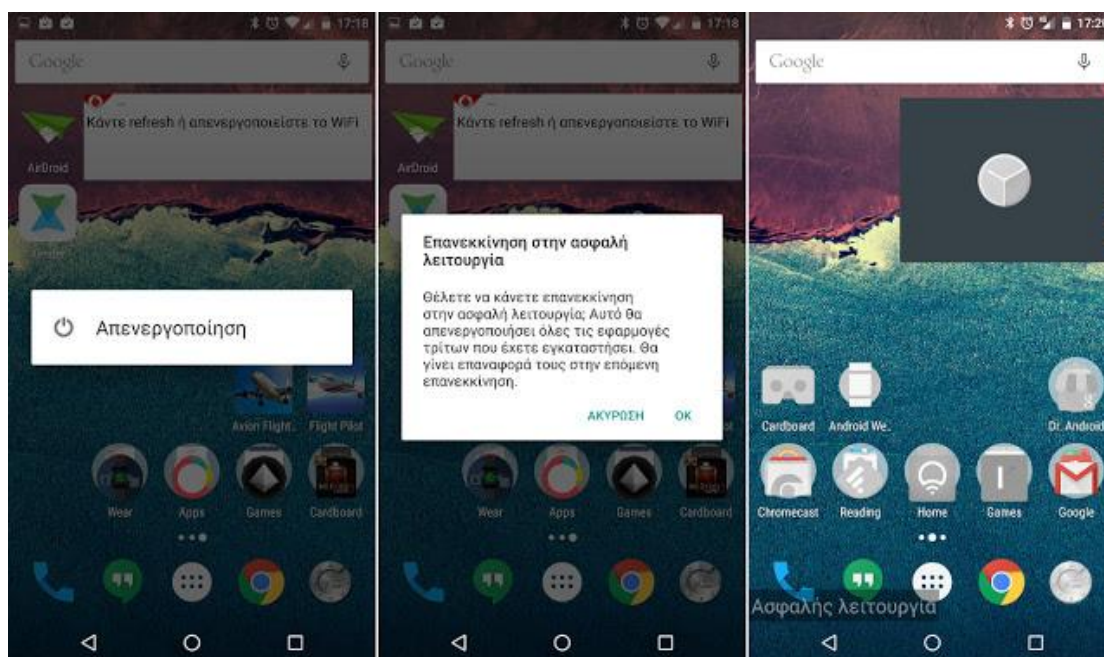


Λύση δεύτερη:

Safe mode

Υπάρχει επίσης το ενδεχόμενο η εφαρμογή να αποκρύπτει ή να απενεργοποιεί αυτά τα μενού και έτσι να μη μας δίνεται η δυνατότητα της απεγκατάστασης. Για αυτές τις περιπτώσεις υπάρχει το "Safe Mode", στο οποίο μπαίνουμε σε ένα ασφαλές περιβάλλον όπου καμία τρίτη εφαρμογή δε λειτουργεί και έτσι έχουμε τη δυνατότητα να τις απεγκαταστήσουμε.

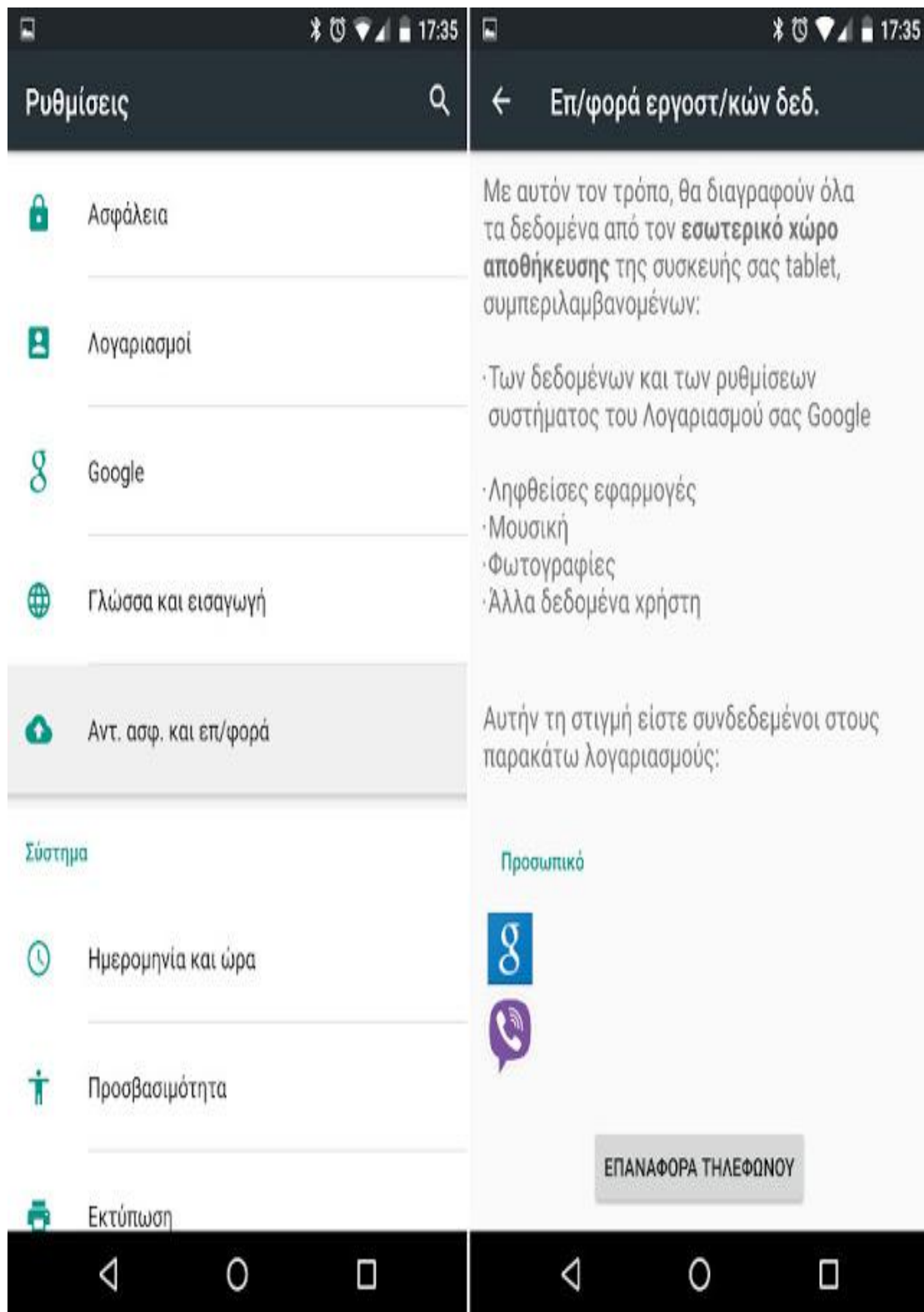
Για να μπούμε λοιπόν σε "Safe Mode" κρατάμε το πλήκτρο "Power" και στο μενού που θα μας εμφανίσει, κρατάμε την επιλογή "Απενεργοποίηση". Θα δούμε στη συνέχεια μία ειδοποίηση η οποία θα μας ενημερώνει αν θέλουμε να μπούμε σε "Safe Mode" και η συσκευή θα κάνει επανεκκίνηση. Σε περίπτωση που αυτό δε γίνεται, απενεργοποιούμε τη συσκευή (με το power ή βγάζοντας την μπαταρία ή κρατώντας το power για πάνω από 10 δευτερόλεπτα) και κατά την εκκίνηση (ή επανεκκίνηση) κρατάμε πατημένα τα δύο πλήκτρα "Volume". Σε αυτό το mode λοιπόν η συσκευή μας είναι "καθαρή" και μας επιτρέπει να απεγκαταστήσουμε όλες τις τρίτες εφαρμογές. Μόλις τελειώσουμε, επανερχόμαστε σε κανονικό mode κάνοντας απλά επανεκκίνηση και είμαστε έτοιμοι.



Υστατη λύση :

Επαναφορά

Αν υπάρχει δυσκολία να εκτελέσουμε όλα τα παραπάνω, υπάρχει πάντα η αξιόπιστη λύση της επαναφοράς εργοστασιακών δεδομένων. Η επαναφορά γίνεται είτε από τη διαδρομή "Ρυθμίσεις > Αντίγραφο ασφαλείας & επαναφορά", είτε μέσω recovery mode, επιλέγοντας "Wipe data / factory reset". Μπορεί να χάσουμε τα δεδομένα μας αλλά τουλάχιστον είμαστε σίγουροι ότι η συσκευή μας θα επανεκκινήσει χωρίς τέτοιου είδους προβλήματα.



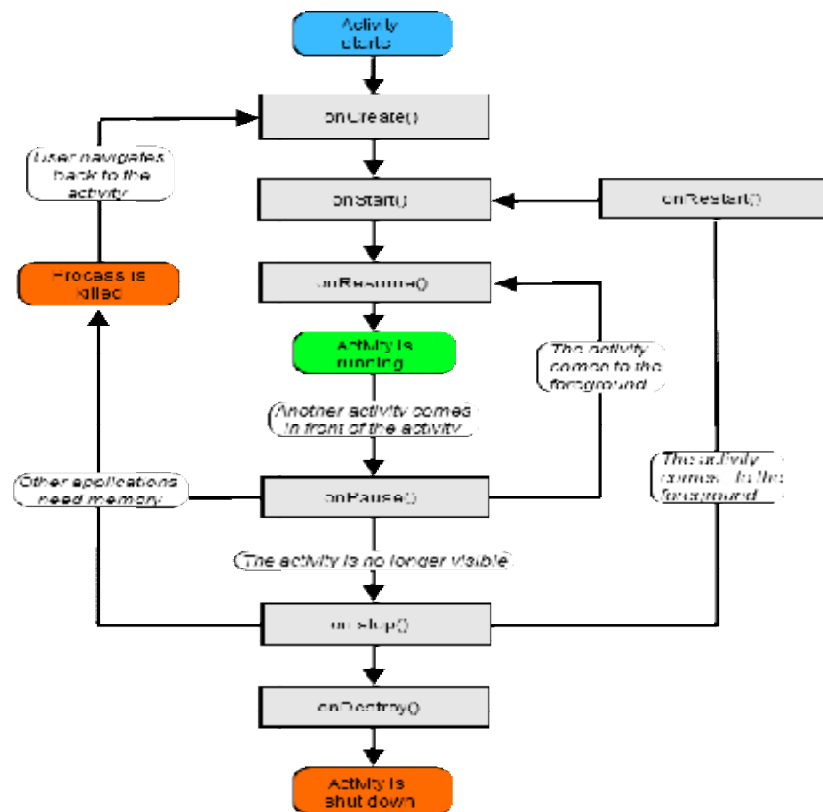
29

²⁹ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2015/09/remove-malware-android.html>.

ΚΕΦΑΛΑΙΟ 3

Υλοποίηση σε Android

3.1 Πώς λειτουργεί μία εφαρμογή android;



Κάθε εφαρμογή android αποτελείται από ένα σύνολο αρχείων και φακέλων δομημένα σε μορφή project. Αυτά όταν γίνουν compiled μέσω του Android SDK μας δίνουν το αρχείο .apk. Το αρχείο αυτό αποτελεί την εφαρμογή και μπορούμε να εγκαταστήσουμε στις συσκευές μας. Ξεκινώντας, η κάθε εφαρμογή αποτελείται όπως είπαμε από πολλά αρχεία δομημένα σε φακέλους. Όλες οι εφαρμογές πρέπει να έχουν ένα μοναδικό όνομα πακέτου (package name) το οποίο χρησιμοποιείτε από το λειτουργικό σύστημα για αναγνώριση της εφαρμογής. Μια εφαρμογή μπορεί να αποτελείται από πολλά υποπακέτα, εφόσον αυτό είναι απαραίτητο λόγω της πολυπλοκότητας της εφαρμογής, αλλά μόνο από ένα κύριο.

Φάκελοι src και res

Στον φάκελο src βρίσκονται τα αρχεία κλάσης της Java όλων των

Activities. Ο φάκελος περιέχει τα πακέτα της εφαρμογής στα οποία βρίσκονται τα αρχεία Java και εκεί είναι αποθηκευμένα τα αρχεία του κώδικα μας.

Ο φάκελος res περιέχει όλα τα αρχεία εικόνας, κειμένου, xml

layout, colors, strings, styles κτλ τα οποία χρησιμοποιούνται από τα Activities που βρίσκονται στον φάκελο src.

Τα αρχεία res είναι χωρισμένα σε υποφακέλους όπως είναι ο φάκελος drawable ο οποίος περιέχει τα αρχεία εικόνας (.png, .jpg, .gif) που χρησιμοποιεί η εφαρμογή μας, ο φάκελος layout ο οποίος περιέχει όλα τα αρχεία xml τα οποία ορίζουν τα διάφορα layouts που υπάρχουν στην εφαρμογή, και τέλος ο

φάκελος values στον οποίο αποθηκεύονται τα αρχεία colors.xml, strings.xml, styles.xml, dimens.xml.

AndroidManifest.xml

Κάθε εφαρμογή android περιέχει ένα αρχείο στο οποίο βρίσκονται καταχωρημένες οι σημαντικότερες πληροφορίες της εφαρμογής το οποίο ονομάζεται

AndroidManifest.xml. Αυτό είναι ένα αρχείο xml μέσα στο

οποίο καταχωρούνται οι σημαντικότερες πληροφορίες της εφαρμογής για

να χρησιμοποιηθούν από το λειτουργικό σύστημα. Μερικές πληροφορίες είναι:

Η έκδοση των APIs που χρησιμοποιούνται

Ο αριθμός έκδοσης της εφαρμογής

Οι άδειες χρήσης που ζητάει η εφαρμογή

Το όνομα του πακέτου της εφαρμογής

Το όνομα της εφαρμογής το οποίο είναι ορατό στον χρήστη

Ασφάλεια εφαρμογών android

Το λειτουργικό σύστημα **Android** συνδυάζει χαρακτηριστικά παρόμοια με εκείνα άλλων γνωστών λειτουργικών συστημάτων, όπως πολυεπεξεργασία (**multitasking**), δικαιώματα αρχείων (**file rights**), κ.τ.λ. Ένα πολύ σημαντικό χαρακτηριστικό είναι ότι χρησιμοποιεί το αναγνωριστικό χρήστη (user identifier) το οποίο αποτελεί έναν τρόπο ασφάλειας εν αντιθέσει για παράδειγμα με το λειτουργικό **Windows**, όπου οι εφαρμογές ανήκουν σε έναν χρήστη και λειτουργούν με το ίδιο **UID**. Κάθε εφαρμογή τρέχει σε μια εικονική μηχανή (VM). Η κάθε VM εκκινείται μόλις ζητηθεί από το σύστημα και κλείνει είτε επειδή δεν χρησιμοποιείται πλέον, είτε επειδή το

σύστημα θέλει να ελευθερώσει τους πόρους της μνήμης για χρήση από άλλη εφαρμογή. Η κάθε εφαρμογή έχει πρόσβαση μόνο σε όσους πόρους απαιτούνται από το σύστημα μέσω του androidmanifest.xml. Τα δικαιώματα τα οποία ελεγχονται από τον τελικό χρήστη συμβάλλουν άμεσα στην ασφάλεια των δεδομένων που αποθηκεύονται σε ένα κινητό και τα οποία θα πρέπει να ελέγχουμε κάθε φορά που κατεβάζουμε μια εφαρμογή στο κινητό μας, ώστε να προστατευτούμε από τυχόν κακόβουλο λογισμικό. Κάποια από τα δικαιώματα που απαιτούν οι εφαρμογές είναι τα ακόλουθα:³⁰

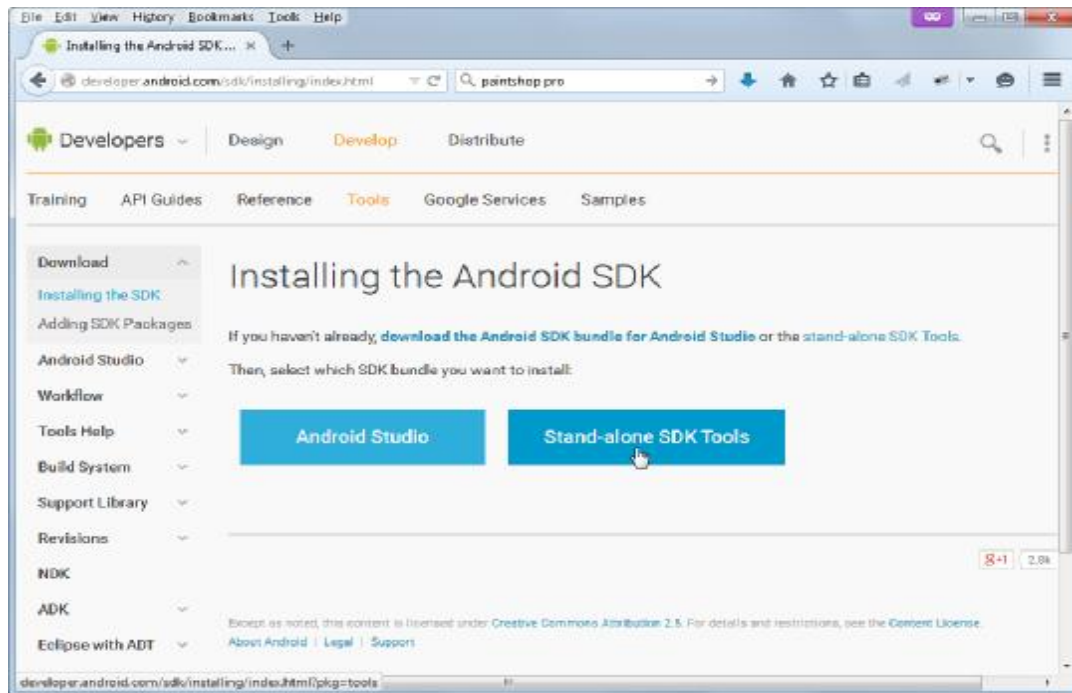
ADD_VOICEMAIL
READ_CALL_LOG
WRITE_CALENDAR
CAMERA
READ_CONTACTS
RING_VOLUME
NOTIFICATION_TOAST
KEEP_AWAKE
READ_SMS
RECEIVE_SMS
READ_EXTERNAL_STORAGE
WRITE_EXTERNAL_STORAGE

ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ ΣΕ ANDROID

Εγκατάσταση προγράμματος και sdk

Αρχικά ο προγραμματιστής προετοιμάζει το περιβάλλον στο οποίο θα γίνει η ανάπτυξη και ο σχεδιασμός της εφαρμογής. Για να το επιτύχει πρέπει να επιλέξει ένα περιβάλλον ανάπτυξης όπως είναι τα eclipse και android studio. Τα προγράμματα αυτά δίνουν τη δυνατότητα χρήσης των SDK, αφού πρώτα εγκατασταθούν, καθώς και άλλων εργαλείων. Αμέσως μετά πρέπει να δημιουργήσει μία εικονική συσκευή (AVD) ώστε να προσομοιώσει τη λειτουργία της εφαρμογής σε πραγματικό χρόνο. Μπορεί επίσης να δοκιμάσει την εφαρμογή σε μια πραγματική συσκευή με τη σύνδεση κινητού τηλεφώνου android και ηλεκτρονικού υπολογιστή μέσω καλωδίου usb.

³⁰ One Plus, G. (17 Μαΐου, 17-5-2016). www.oneplusgreece.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.oneplusgreece.com/android-and-security/>.



Σχεδίαση και ανάπτυξη πηγαίου κώδικα

Σε αυτό το σημείο ο προγραμματιστής λαμβάνει υπόψη τις προδιαγραφές και το περιεχόμενο της εφαρμογής και προχωρά στη σχεδίαση του layout και της συγγραφής του πηγαίου κώδικα σε γλώσσα java και xml.

Κατά τον σχεδιασμό θα πρέπει να αποφευχθεί η χρήση μεγάλης πληροφορίας η οποία θα κουράζει τον τελικό χρήστη. Αντιθέτως απαιτείται ομοιομορφία και λιτό περιεχόμενο. Η εφαρμογή θα πρέπει να ακολουθεί ορισμένους κανόνες σχεδίασης:

ΚΙΝΗΣΗ: Ο χρήστης θα πρέπει να μεταβαίνει ομαλά από το ένα στοιχείο στο άλλο με φυσική ροή και σε κάθε σελίδα να υπάρχει αρχή μέση και τέλος.

ΙΣΟΡΡΟΠΙΑ: Θα πρέπει να δοθεί έμφαση στο κείμενο καθώς και στην αναλογία του με τα υπόλοιπα στοιχεία.

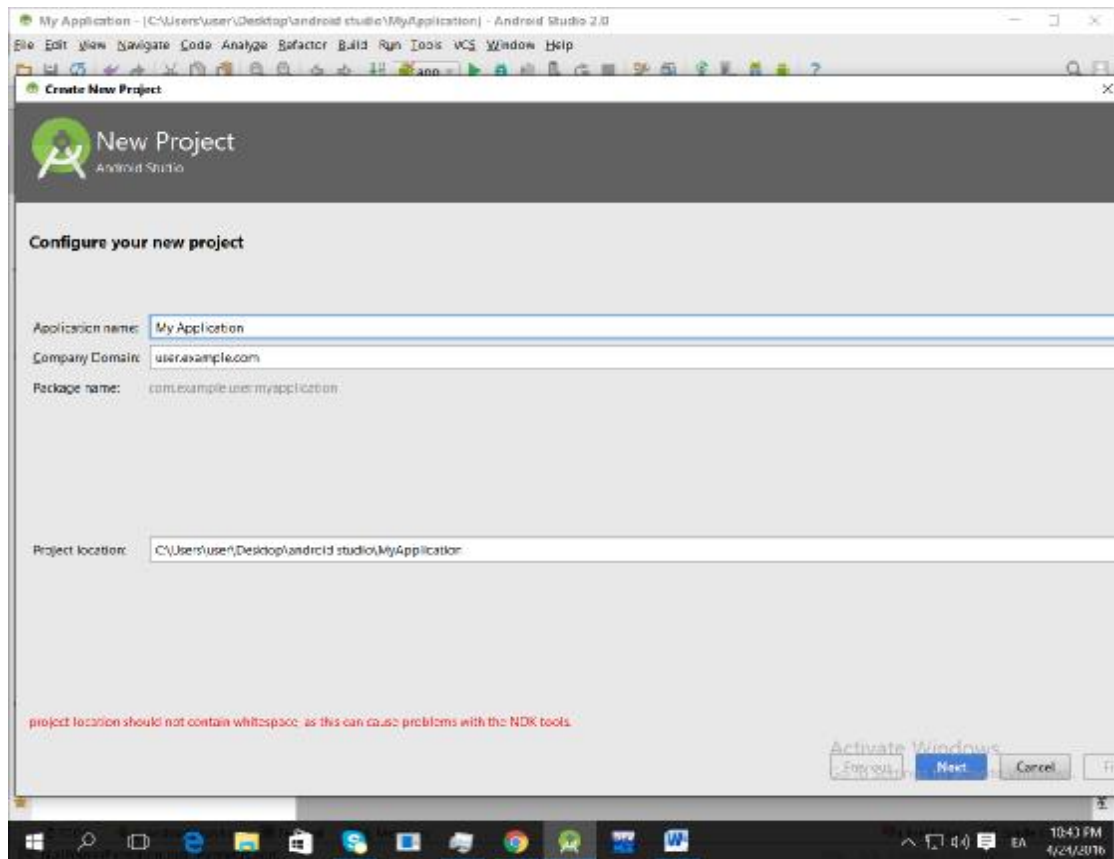
ΣΗΜΕΙΑ ΕΝΔΙΑΦΕΡΟΝΤΟΣ: Θα πρέπει ο προγραμματιστής να καταφέρει να επικεντρώσει το ενδιαφέρον του χρήστη στα σημεία που θέλει αυτός.

WHITESPACE: Η χρήση τους είναι απαραίτητη για να υπάρχει κενός χώρος ανάμεσα στα στοιχεία σχεδιασμού ώστε να μην κουράζεται ο χρήστης.

ΕΥΘΥΓΡΑΜΜΙΣΗ: Η σωστή ευθυγράμμιση απαιτείται για να δείχνει το κείμενο πιο συμμετρικό και προσεγμένο.

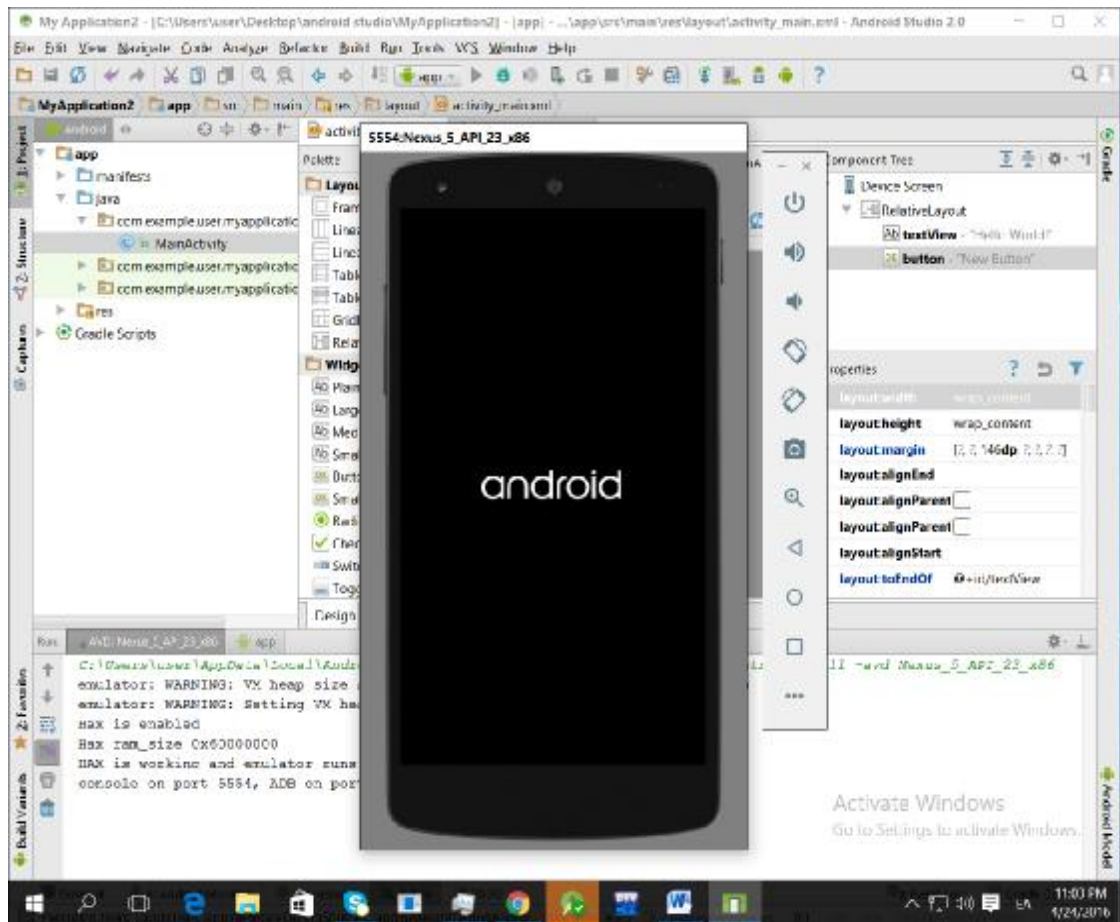
Η διαδικασία ανάπτυξης της εφαρμογής ξεκινά επιλέγοντας File->New->New project και ρυθμίζοντας τις παραμέτρους στα επόμενα βήματα. Το νέο project περιέχει τον πηγαίο κώδικα, το κείμενο, τις εικόνες και άλλα αρχεία που απαιτούνται. Ο

προγραμματιστής θα πρέπει να φροντίσει ώστε να μην υπάρχουν λάθη στον κώδικα. Στη συνέχεια ακολουθεί το debugging.



Αποσφαλμάτωση(Debugging)

Αρχικά γίνεται το χτίσιμο(build) της εφαρμογής η οποία λειτουργεί σε debug mode. Αμέσως μετά ακολουθεί η μεταγλώττιση(compile) του πηγαίου κώδικα σε γλώσσα μηχανής. Αν προκύψει σφάλμα εντος του πηγαίου κώδικα, τότε το πρόγραμμα δεν λειτουργεί και εμφανίζεται μήνυμα σφάλματος στην οθόνη. Αν δεν προκύψει κάποιο σφάλμα τότε προχωράμε στη δοκιμή της εφαρμογής είτε με χρήση εικονικής συσκευής που παρεχει το πρόγραμμα είτε με χρήση φυσικής συσκευής όπως για παράδειγμα ένα κινητό τηλέφωνο με λειτουργικό σύστημα android.



Έπειτα ο προγραμματιστής διορθώνει τυχόν λειτουργικά ή αισθητικά προβλήματα που έχουν προκύψει εντοπίζοντας τα στην εικονική ή τη φυσική συσκευή βελτιώνοντας τον κώδικα σε όποια σημεία απαιτείται. Σε αυτό βοηθά και το Logcat το οποίο επιστρέφει το stack trace του κώδικα στον οποίο υπάρχει σφάλμα.



DEBUGGING IN ANDROID

- Add **red** breakpoint in first line of `increment & decrement` methods

29
30
31
32

✓

<code>public void</code>
<code>int qua</code>
<code>display</code>
- Run in Debug Mode
- Step through each line of code.
 Click to resume execution of app.

Δημοσίευση της εφαρμογής

Στο τελευταίο στάδιο, ο προγραμματιστής, αφού ολοκληρωθεί η διαδικασία του debugging, ελέγχει ξανά αν λειτουργεί η εφαρμογή με τον τρόπο που πρέπει. Αν δεν προκύψουν σφάλματα τότε γίνονται οι τελικές ρυθμίσεις και το τελικό compile. Η εφαρμογή είναι έτοιμη προς δημοσίευση.



ΑΡΧΙΤΕΚΤΟΝΙΚΗ ANDROID

Το λειτουργικό σύστημα android αποτελείται από 4 συνιστώσες:

Πυρήνας Linux

Ο πυρήνας του Android βασίζεται στον πυρήνα του Linux τον οποίο μπορεί να βρει οποιοσδήποτε ελεύθερα στο ίντερνετ. Ο πυρήνας στο Linux είναι ένα από τα σημαντικότερα χαρακτηριστικά, διότι εκτός από το λειτουργικό σύστημα περιέχει και τους οδηγούς συσκευών και έναν μεγάλο αριθμό πρωτοκόλλων και συστημάτων αρχείων. Οι οδηγοί αυτοί είναι υπεύθυνοι για την επικοινωνία του software με το hardware της συσκευής. Ο πυρήνας αναλαμβάνει τη διαχείριση μνήμης, τη διαχείριση διεργασιών, τις λειτουργίες δικτύου και την ασφάλεια του λειτουργικού.

Βιβλιοθήκες(Libraries)

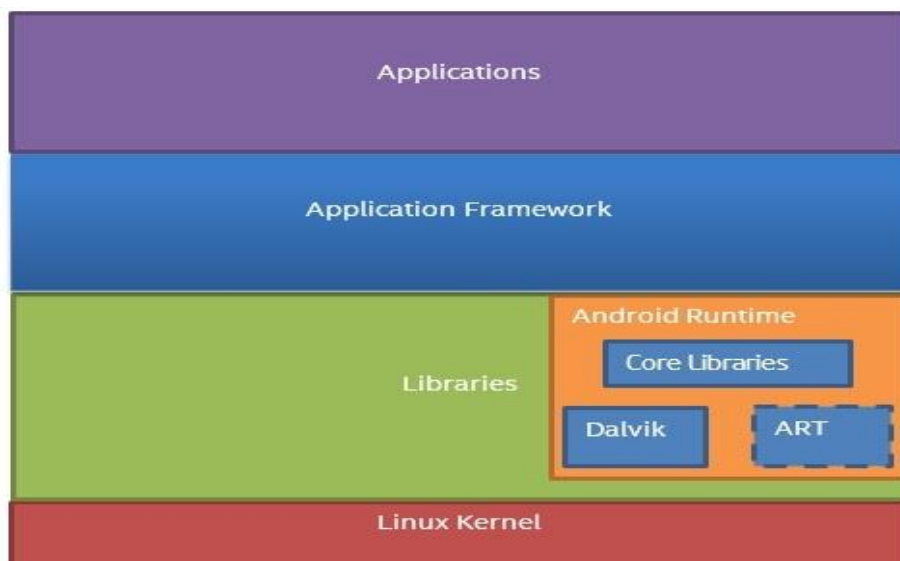
Στις βιβλιοθήκες συγκεντρώνεται όλος ο κώδικας του λειτουργικού συστήματος καθώς και τα χαρακτηριστικά που θα χρησιμοποιηθούν αργότερα στην εφαρμογή.

Εικονική Μηχανή (Android Runtime)

Εδώ περιέχονται οι βασικές βιβλιοθήκες που χρησιμοποιεί ο προγραμματιστής για να γράψει κώδικα σε γλώσσα java.

Πλαίσιο Εφαρμογής (Application Framework)

Σε αυτό το επίπεδο ο προγραμματιστής αποκτά πρόσβαση σε όλα τα APIs που χρησιμοποιούν οι εφαρμογές. Είναι δυνατό να γίνει χρήση των δυνατοτήτων μίας εφαρμογής από άλλες σύμφωνα με τις προδιαγραφές ασφαλείας του λειτουργικού συστήματος android.



Η ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ JAVA



Ένα από τα βασικά πλεονεκτήματα της Java έναντι των περισσότερων άλλων γλωσσών είναι η ανεξαρτησία του λειτουργικού συστήματος και πλατφόρμας. Τα προγράμματα που είναι γραμμένα σε Java τρέχουν ακριβώς το ίδιο σε πλατφόρμες όπως Windows, Linux, Unix και Macintosh χωρίς να χρειαστεί να ξαναγίνει μεταγλώττιση (compiling) ή να αλλάξει ο πηγαίος κώδικας για κάθε διαφορετικό λειτουργικό σύστημα. Για να επιτευχθεί όμως αυτό χρειαζόταν κάποιος τρόπος έτσι ώστε τα προγράμματα γραμμένα σε Java να μπορούν να είναι διαβαστούν από κάθε υπολογιστή ανεξάρτητα του είδους επεξεργαστή (Intel x86, IBM, Sun SPARC, Motorola) αλλά και λειτουργικού συστήματος (Windows, Unix, Linux, BSD, MacOS). Ο συμβολικός κώδικας (assembly) που μεταφράζεται και εκτελείται σε

Windows είναι διαφορετικός από αυτόν που μεταφράζεται και εκτελείται σε έναν υπολογιστή Macintosh. Η λύση δόθηκε με την ανάπτυξη της Εικονικής Μηχανής (Virtual Machine).

Η εικονική μηχανή της Java

Αφού γραφεί κάποιο πρόγραμμα σε Java, γίνεται μεταγλώττιση μέσω του μεταγλωττιστή javac, ο οποίος δημιουργεί έναν αριθμό από αρχεία .class. Όταν πρόκειται να εκτελεστεί η εφαρμογή σε ένα μηχάνημα, το Java Virtual Machine αναλαμβάνει να διαβάσει τα αρχεία .class. Στη συνέχεια αυτά μεταφράζονται σε γλώσσα μηχανής που να υποστηρίζεται από το λειτουργικό σύστημα και τον επεξεργαστή, με σκοπό να εκτελεστεί. Πιο σύγχρονες εφαρμογές της εικονικής Μηχανής μπορούν και μεταγλωττίζουν εκ των προτέρων τμήματα bytecode απευθείας σε κώδικα μηχανής με αποτέλεσμα να βελτιώνεται η ταχύτητα). Η Java Virtual Machine είναι λογισμικό που εξαρτάται από την πλατφόρμα, δηλαδή για κάθε είδος λειτουργικού συστήματος και αρχιτεκτονικής επεξεργαστή υπάρχει διαφορετική έκδοση του. Έτσι υπάρχουν διαφορετικές Java Virtual Machines για Windows, Linux, Unix, Macintosh, κινητά τηλέφωνα, παιχνιδιομηχανές κτλ.

Οτιδήποτε θέλει να κάνει ο προγραμματιστής γίνεται μέσω της εικονικής μηχανής. Αυτό βοηθάει στο να υπάρχει μεγαλύτερη ασφάλεια στο σύστημα γιατί η εικονική μηχανή είναι υπεύθυνη για την επικοινωνία χρήστη - υπολογιστή. Ο προγραμματιστής δεν μπορεί να γράψει κώδικα ο οποίος θα έχει καταστροφικά αποτελέσματα για τον υπολογιστή γιατί η εικονική μηχανή θα τον ανιχνεύσει και δε θα επιτρέψει να εκτελεστεί. Από την άλλη μεριά ούτε ο χρήστης μπορεί να κατεβάσει «κακό» κώδικα από το διαδίκτυο και να τον εκτελέσει.³¹

³¹ [Java Tutorial \(\(Linux\)\), \(\(Windows\)\)](#)

[Community \(\(Linux\)\), \(\(Windows\)\)](#)

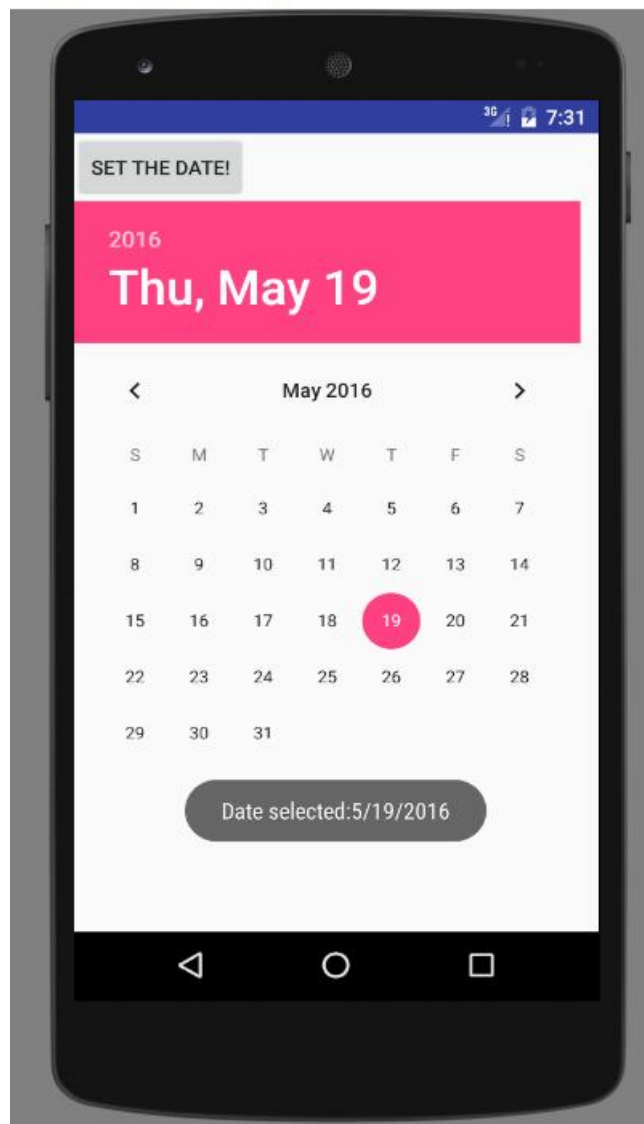
[Java στο Open Directory Project](#)

3.1 Δημιουργία της εφαρμογής DatePicker

3.1.1 Ανάλυση απαιτήσεων και σχεδιασμός

Η συγκεκριμένη εφαρμογή δίνει τη δυνατότητα στο χρήστη να επιλέξει μία συγκεκριμένη ημερομηνία απο το ημερολόγιο. Το ημερολόγιο εμφανίζεται αμέσως μετά την εκτέλεση της εφαρμογής. Η εφαρμογή θα πρέπει να σχεδιαστεί με κατάλληλο τρόπο ώστε ο χρήστης να μπορεί με το πάτημα ενός μόνο κουμπιού να επιλέγει οποιαδήποτε ημερομηνία επιθυμεί και ταυτόχρονα αυτή να εμφανίζεται στο κάτω μέρος την οθόνης σε δύο γραμμές. Στην πρώτη γραμμή αναγράφεται το επιλεγμένο έτος ενώ στη δεύτερη αναγράφονται κατά σειρά η ημέρα, ο μήνας και ο αριθμός.

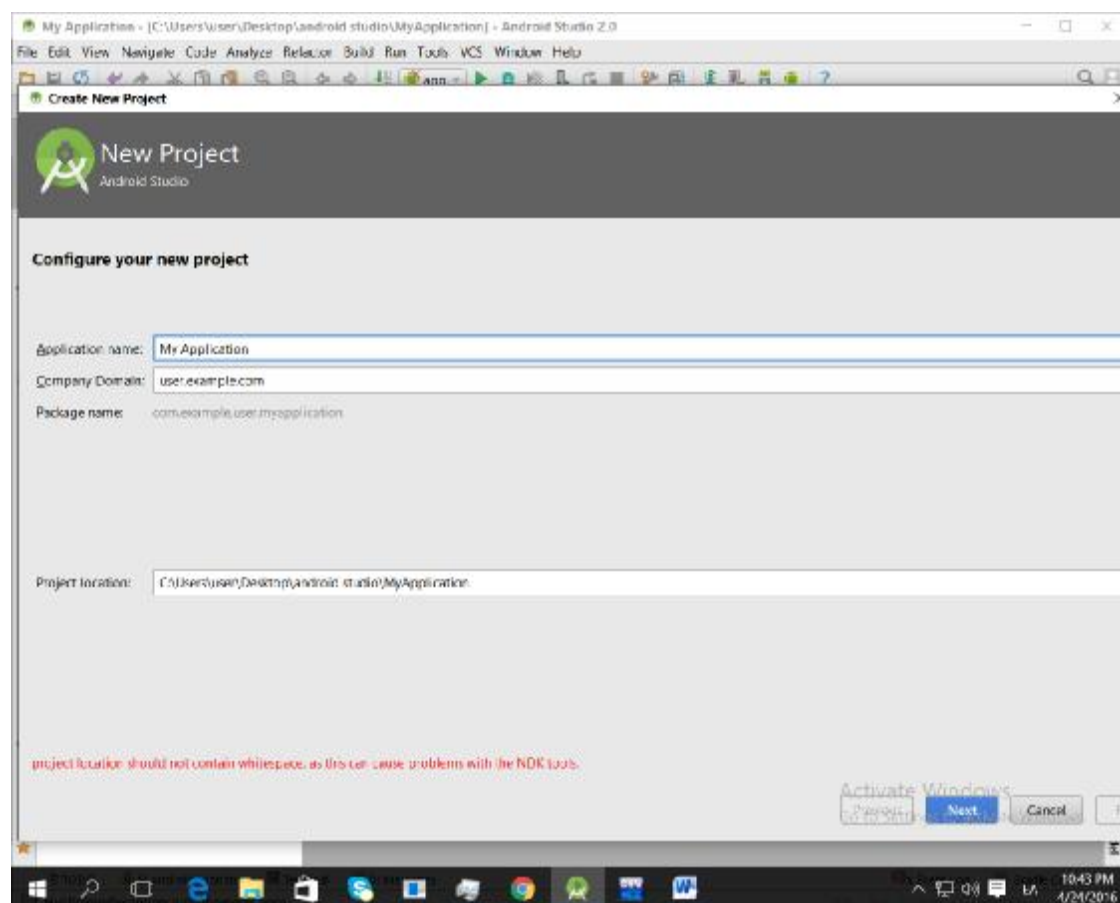
5554:Nexus_5_API_23_x86



Εικόνα 3.3.1 Υπολογισμός ημερομηνίας με την εφαρμογή DatePicker

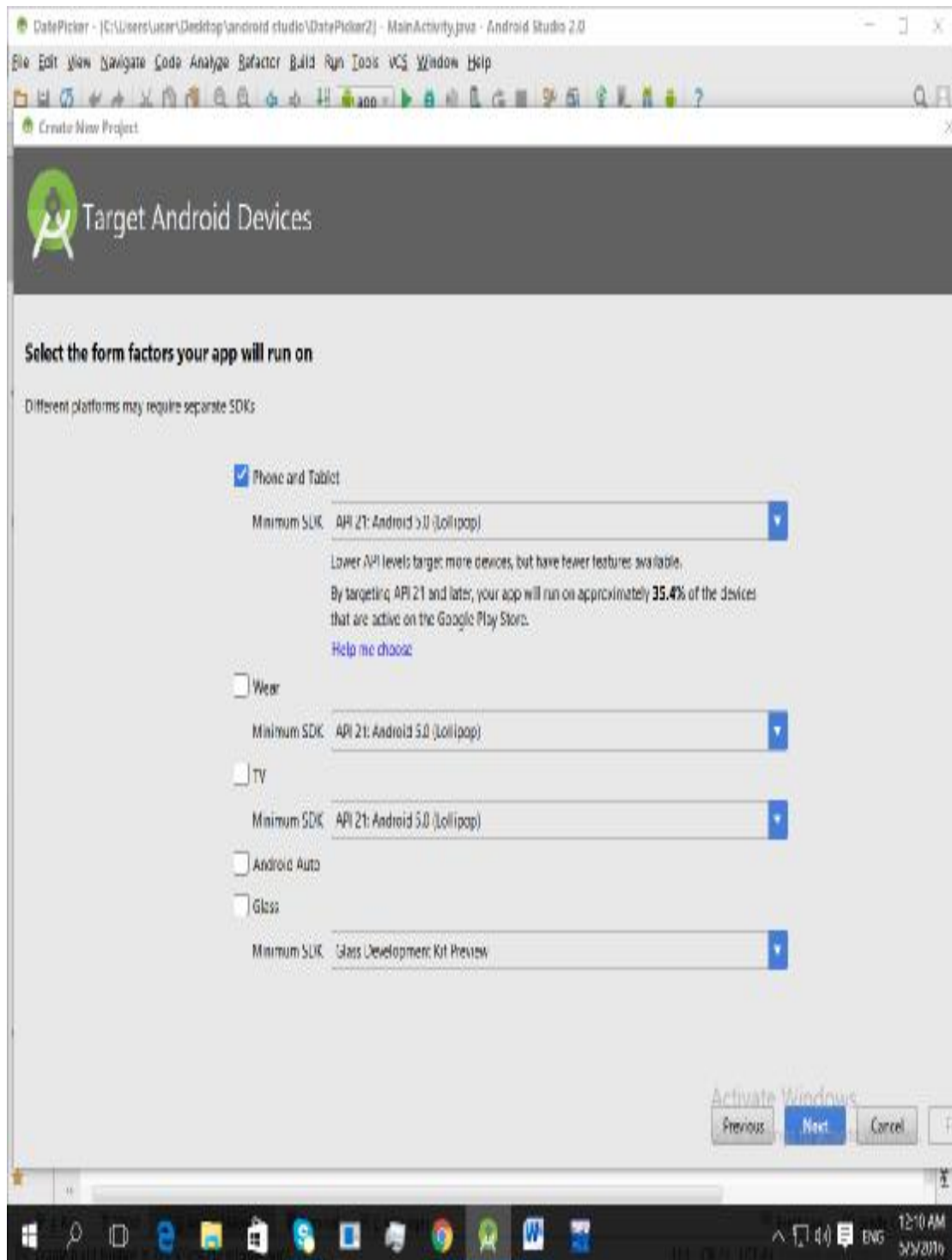
3.1.2 Δημιουργία project με το Android Studio

Αφού έχουν οριστεί οι απαιτήσεις της εφαρμογής θα προχωρήσουμε στην υλοποίηση της με το πρόγραμμα Android Studio. Αρχικά δημιουργούμε ένα νέο project επιλέγοντας File->Project->New Project.



Εικόνα 3.1.2 Δημιουργία νέου project με το Android Studio

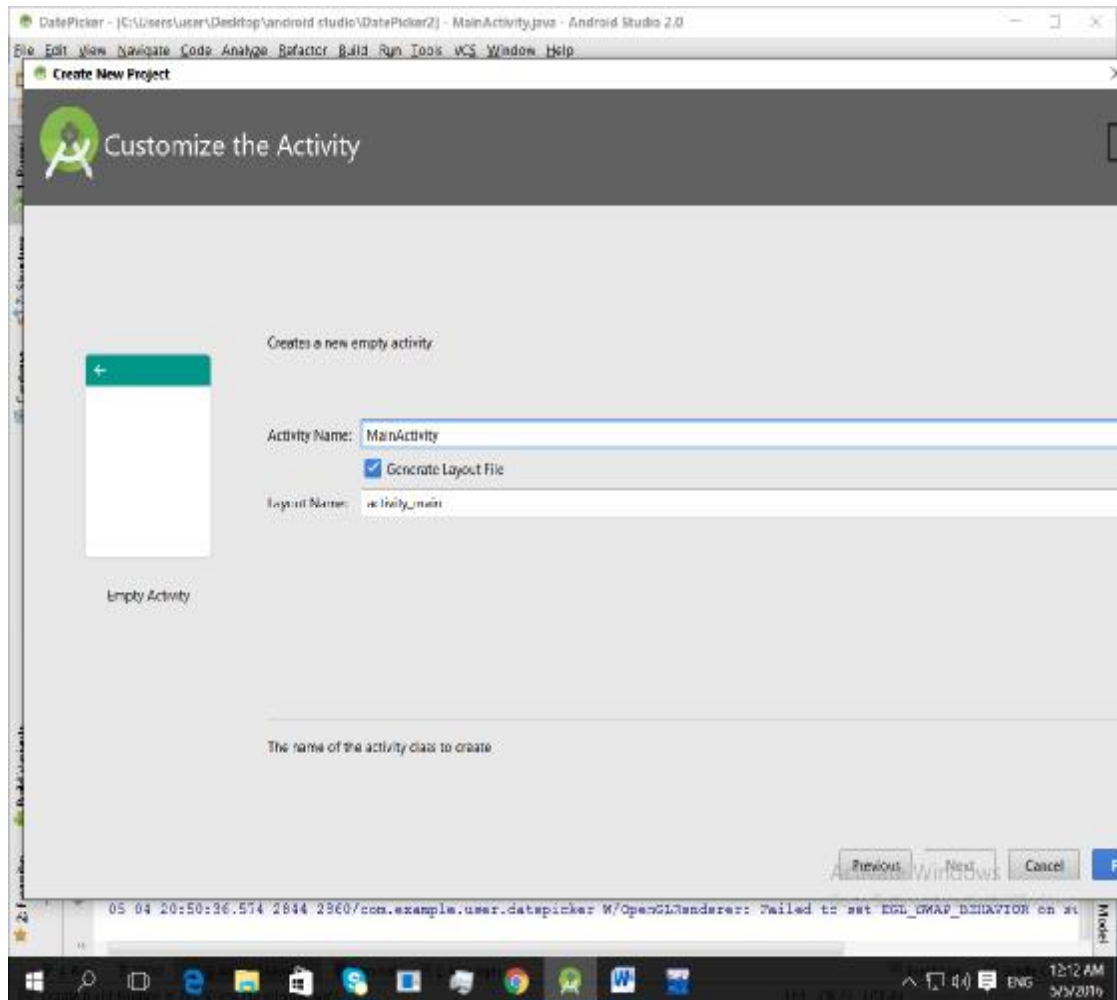
Στη συνέχεια δηλώνουμε το όνομα της εφαρμογής (Application name) και μετά τη διαδρομή της (Project location). Επιλέγουμε Next και στην αμέσως επόμενη σελίδα καλούμαστε να επιλέξουμε την ελάχιστη έκδοση του android καθώς και με ποιες συσκευές θα είναι συμβατή η εφαρμογή μας. Κάνουμε κλικ στην επιλογή Phone and Tablet και για minimum SDK επιλέγουμε το API 21:Android 5.0(Lollipop).Κάνουμε κλικ στην επιλογή Next.



Εικόνα 3.1.3 Επιλογή συσκευής android και ελάχιστου υποστηριζόμενου επιπέδου SDK

Επιλέγουμε Empty Activity, Next και στη συνέχεια δίνουμε όνομα στο Activity Name. στο τέλος κάνουμε κλικ στην επιλογή Finish.

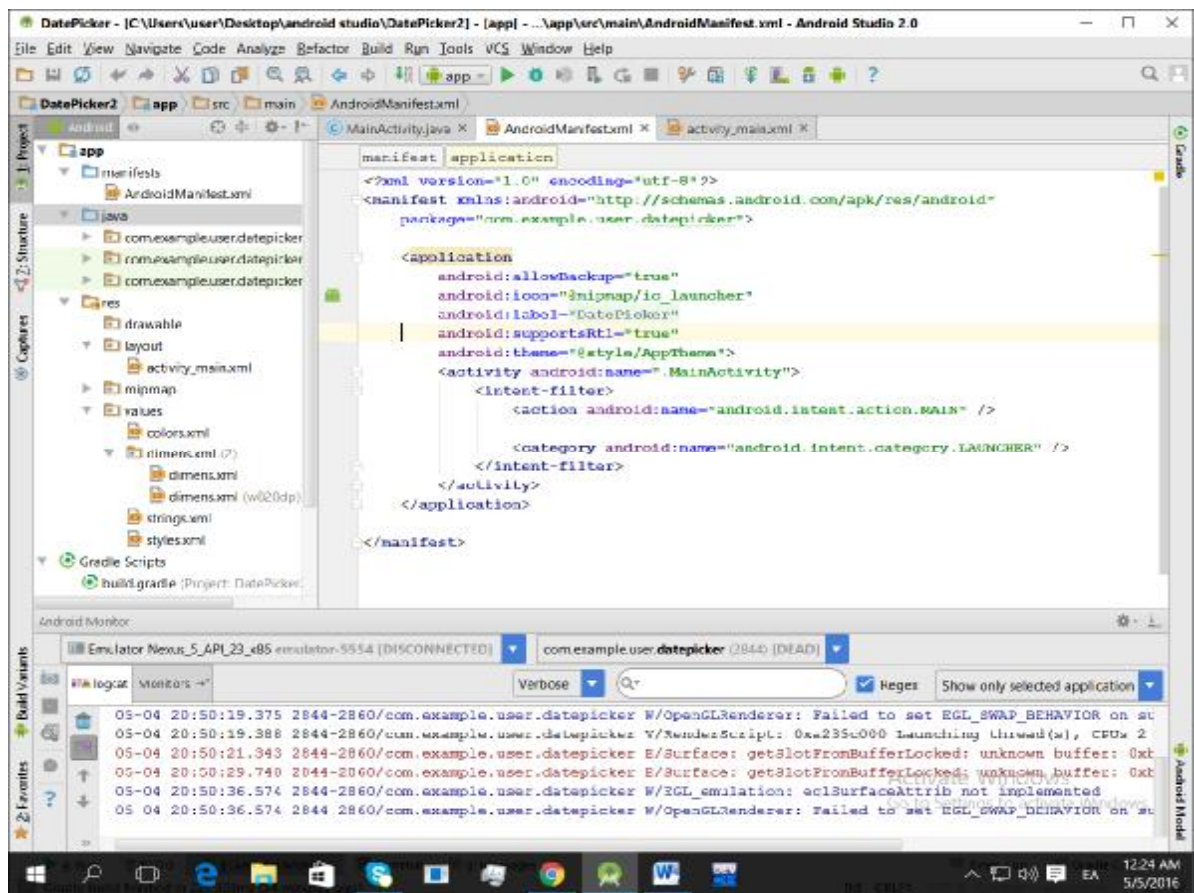
3.1.3 Ολοκλήρωση του setup της εφαρμογής



Εικόνα 3.1.4 Ολοκλήρωση setup και καθορισμός των activity name και layout name.

Το Android Studio δημιουργεί από προεπιλογή τα σημαντικά αρχεία που απαιτούνται για τη λειτουργία της εφαρμογής όπως είναι για παράδειγμα το `activity_main.xml`, το `AndroidManifest.xml`, το `MainActivity.java` και εμείς τροποποιούμε τον κώδικα με κατάλληλο τρόπο. Το αρχείο `AndroidManifest.xml` περιέχει όλες τις activities της εφαρμογής οι οποίες είναι δηλωμένες εκεί.

3.1.4 Έναρξη συγγραφής κώδικα



Εικόνα 3.1.5 Συγγραφή κώδικα εφαρμογής ξεκινώντας από το Android Manifest

Βλέπουμε ότι μεταξύ των tags «application» υπάρχει μία activity δηλωμένη με τα αντίστοιχα tags, τα οποία θα χρησιμοποιήσουμε για να προσθέσουμε τις υπόλοιπες activities. Συνέχεια έχει ο σχεδιασμός του layout γράφοντας κώδικα σε xml στο αρχείο activity_main.xml.

```
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent" >
    <Button
        android:id="@+id/btnSet"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:onClick="btnClick"
```

```

        android:text="Set the date!" />
<DatePicker
    android:id="@+id/datePicker"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/btnSet" />
</RelativeLayout>

```

Αρχικά δημιουργούμε ένα κουμπί (Button), θέτουμε ως id το `"@+id/btnSet"`, δίνουμε την εντολή `android:onClick="btnClick"` και με την εντολή `android:text="Set the date!"` δηλώνουμε το όνομα του κουμπιού. Στη συνέχεια δημιουργούμε ένα DatePicker θέτοντας ως id το `"@+id/datePicker"`.

Σειρά έχει η συγγραφή κώδικα σε java μέσα στο αρχείο MainActivity.java.

```

package com.example.user.datepicker;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.DatePicker;
import android.widget.Toast;

public class MainActivity extends Activity {
    DatePicker datePicker;
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        datePicker = (DatePicker) findViewById(R.id.datePicker);
    }
    public void btnClick(View view) {
        Toast.makeText(getApplicationContext(),
            "Date selected:" + (datePicker.getMonth() + 1) +
            "/" + datePicker.getDayOfMonth() +
            "/" + datePicker.getYear(),
            Toast.LENGTH_SHORT).show();
    }
}

```

Στο αρχείο Main_Activity.java δηλώνουμε ένα datePicker (`datePicker = (DatePicker) findViewById(R.id.datePicker);`); ώστε να εμφανιστεί στην οθόνη η

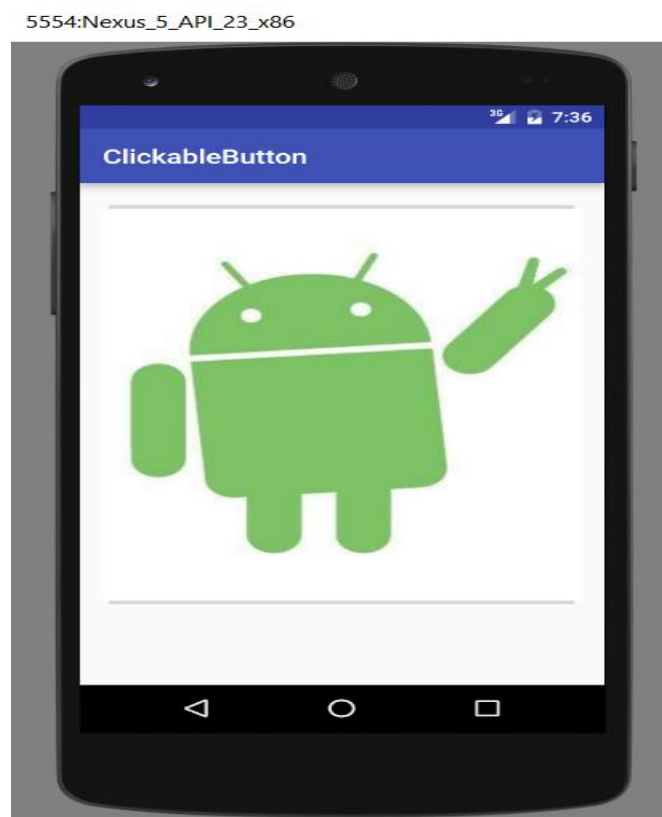
προεπιλεγμένη εφαρμογή του ημερολογίου. Η εντολή `btnClick(View view)` δίνει τη δυνατότητα στον χρήστη να κάνει κλικ πάνω σε οποιαδήποτε ημερομηνία επιθυμεί. Με το πάτημα του κουμπιού ενεργοποιείται η εντολή `Toast.makeText(getApplicationContext())` και εμφανίζεται αμέσως το μήνυμα **"Date selected:"** ακολουθούμενο από τον μήνα που έχουμε επιλέξει, την ημέρα και το έτος.

Τέλος καλούμε τις μεθόδους `getMonth()`, `getDayOfMonth()` και `getYear` στο `DatePicker` για να επιλέξουμε στην οθόνη τον επιλεγμένο μήνα, την ημέρα και το έτος αντίστοιχα.

3.2 Δημιουργία εφαρμογής `clickable_button`

3.2.1 Ανάλυση απαιτήσεων

Η εφαρμογή αυτή δίνει στον χρήστη τη δυνατότητα μετάβασης από μία φωτογραφία στην επόμενη με ένα μόνο κλικ. Θα χρησιμοποιήσουμε ένα `imagebutton` και τη μέθοδο **`android:onClick`** για τη διαδικασία αυτή. Στη διαδρομή `res->drawable` προσθέσαμε 2 φωτογραφίες (`image1`, `image2`) ώστε να μπορούν να αναγνωριστούν αργότερα από το πρόγραμμα.



Εικόνα 3.2.1 Εμφάνιση αποτελέσματος εφαρμογής `Clickable Button` στην οθόνη

3.2.2 Συγγραφή κώδικα

Αρχείο AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.user.clickablebutton">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

Αρχείο activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"

    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.user.clickablebutton.MainActivity">
```

```
<ImageButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/imageButton1"
    android:src="@drawable/image1"
    android:onClick="buttonClick"
/>

</RelativeLayout>
```

Προσθέσαμε ένα `imagebutton` και μέσα σε αυτό τη μέθοδο `android:onClick="buttonClick"` για να μπορεί ο χρήστης να κάνει κλικ πάνω στη φωτογραφία. Ορίσαμε το `android:id=` σε `"@+id/imageButton1"` και το `android:src=` σε `"@drawable/image1"` για να εμφανιστεί η φωτογραφία με όνομα `image1` κατά την εκτέλεση του κώδικα. Οι φωτογραφίες αποθηκεύονται στον φάκελο `drawable` του `android studio`.

Αρχείο `MainActivity.java`

```
package com.example.user.clickablebutton;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.ImageButton;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    public void buttonClick(View v) {
```

```
ImageButton aButton = (ImageButton) v;  
aButton.setImageResource(R.drawable.image2);  
}  
}
```

Αφού προσθέσαμε τις παρακάτω βιβλιοθήκες:

```
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.ImageButton;
```

δημιουργήσαμε μία νέα κλάση MainActivity και στη συνέχεια μία μέθοδο **public void** buttonClick όπου εκεί δηλώνουμε μία μεταβλητή aButton. Με το κλικ που θα κάνει ο χρήστης πάνω στη φωτογραφία εμφανίζεται μία άλλη το image2 με την ακόλουθη εντολή: `aButton.setImageResource(R.drawable.image2);`

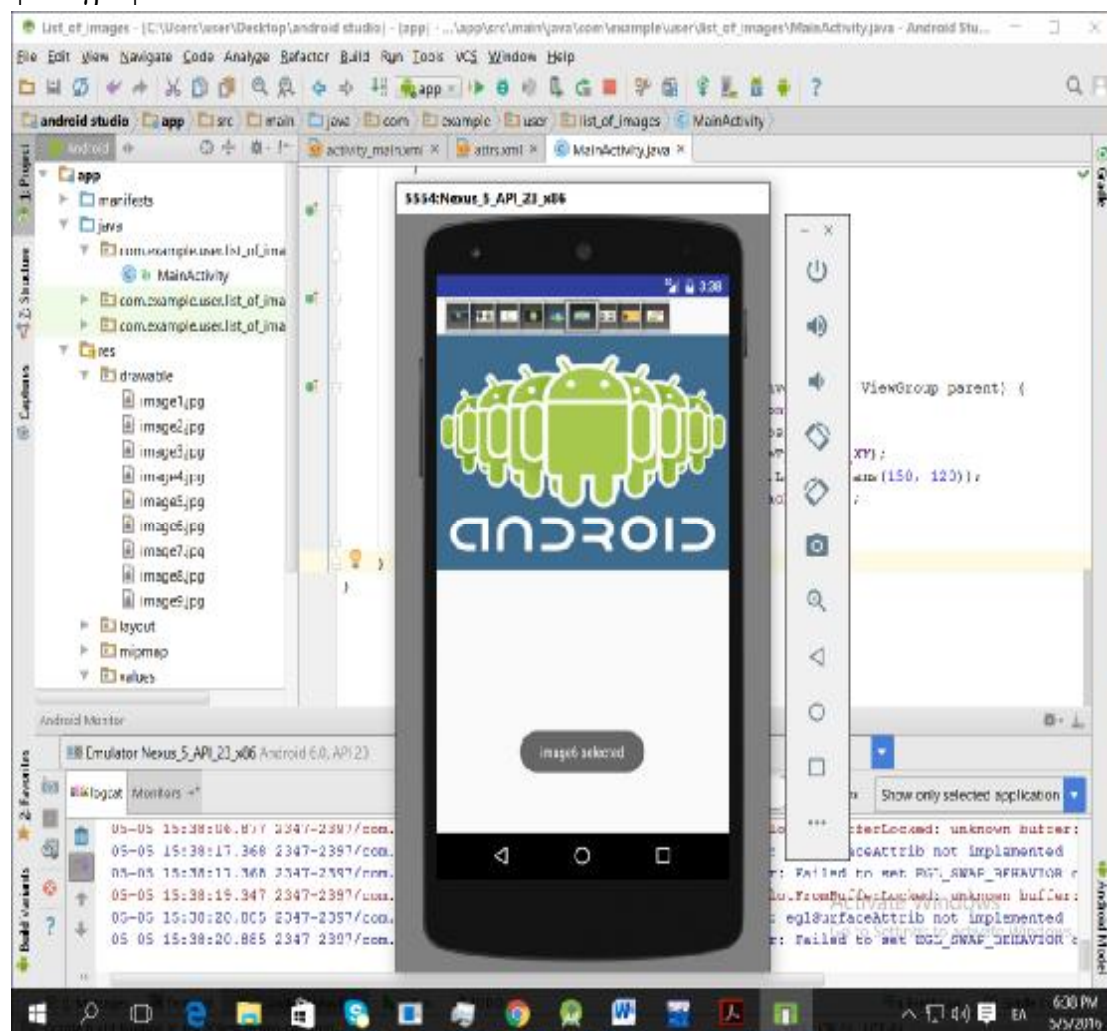
Σε αυτή την εφαρμογή υπάρχει αλλαγή μίας μόνο φωτογραφίας και εμφάνιση της επόμενης. Στην επόμενη εφαρμογή θα αναλύσουμε πώς μπορούμε να επιλέξουμε περισσότερες από δύο φωτογραφίες και να γίνει εναλλαγή τους.

3.3 Δημιουργία εφαρμογής list_of_images

3.3.1 Ανάλυση απαιτήσεων εφαρμογής

Αυτή η εφαρμογή έχει σκοπό να εμφανίσει στην οθόνη διάφορες φωτογραφίες σε μορφή λίστας.

Ο χρήστης έχει τη δυνατότητα να επιλέξει όποια επιθυμεί κάθε φορά. Οι φωτογραφίες βρίσκονται στο πάνω μέρος της οθόνης και κάθε φορά που επιλέγεται μία από αυτές, αυτή εμφανίζεται στο κέντρο της οθόνης σε μεγαλύτερο μέγεθος από τις υπόλοιπες. Προστίθεται δηλαδή animation στις φωτογραφίες έτσι ώστε αυτές να αλλάζουν γρήγορα και ομαλά χωρίς να χρειαστεί να κάνουμε κλικ πολλές φορές. Ένα μήνυμα εμφανίζεται στο κάτω μέρος της οθόνης το οποίο αναφέρει την επιλεγμένη φωτογραφία.



Εικόνα 3.3.1 Εμφάνιση της έκτης κατά σειρά φωτογραφίας της εφαρμογής List_Of_Images

3.3.2 Συγγραφή κώδικα

Στην αρχή αποθηκεύουμε τις φωτογραφίες που θα χρησιμοποιηθούν αργότερα στην εφαρμογή στον φακελο drawable του android studio για να τις χρησιμοποιήσουμε κατά την εκτέλεση του κώδικα. Δημιουργούμε ένα αρχείο με όνομα attrs.xml στο φάκελο values για να κάνουμε χρήση μίας gallery.

Αρχείο attrs.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <declare-styleable name="MyGallery">
    <attr name="android:galleryItemBackground" />
  </declare-styleable>
</resources>
```

Στη συνέχεια προσθέτουμε στο αρχείο activity_main.xml ένα Gallery και ένα ImageView βάζοντας ως id τα "@+id/gallery1" και "@+id/image1" αντίστοιχα. Ορίζουμε το **android:layout_width** σε **"fill_parent"** για το Gallery και το **android:layout_height** σε **"wrap content"** για το πρώτο και 250dp για το δεύτερο. Στο ImageView προσθέτουμε την εντολή **android:scaleType="fitXY"** για να προσαρμόσουμε το μέγεθος της φωτογραφίας μέσα στην οθόνη.

Αρχείο activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent"
  android:orientation="vertical">
  <Gallery
    android:id="@+id/gallery1"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content" />
  <ImageView
    android:id="@+id/image1"
```

```
    android:layout_width="fill_parent"  
    android:layout_height="250dp"  
    android:scaleType="fitXY" />  
</LinearLayout>
```

Στο αρχείο MainActivity.java δηλώνουμε τις ακόλουθες βιβλιοθήκες

```
import android.app.Activity;  
import android.content.Context;  
import android.content.res.TypedArray;  
import android.os.Bundle;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.AdapterView;  
import android.widget.AdapterView.OnItemClickListener;  
import android.widget.BaseAdapter;  
import android.widget.Gallery;  
import android.widget.ImageView;  
import android.widget.Toast
```

καθώς και μία μεταβλητή με όνομα `imageIDs` η οποία περιέχει τις εννέα φωτογραφίες που θα χρησιμοποιηθούν στην εφαρμογή. Δηλώνουμε τη μεταβλητή `Gallery` και τη μέθοδο `public void onItemClick(AdapterView)` για να μπορεί ο χρήστης να κάνει κλικ σε οποιαδήποτε φωτογραφία. Με τη μέθοδο `Toast.makeText` εμφανίζεται στην οθόνη ένα μήνυμα το οποίο μας ειδοποιεί σχετικά με το ποια φωτογραφία έχουμε επιλέξει. Με τη μέθοδο `ImageView imageView = (ImageView) findViewById` εμφανίζονται οι επιλεγμένες φωτογραφίες.

Αρχείο MainActivity.java

```
package com.example.user.list_of_images;  
  
import android.app.Activity;  
import android.content.Context;  
import android.content.res.TypedArray;  
import android.os.Bundle;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.AdapterView;  
import android.widget.AdapterView.OnItemClickListener;  
import android.widget.BaseAdapter;
```

```

import android.widget.Gallery;
import android.widget.ImageView;
import android.widget.Toast;

@SuppressWarnings("deprecation")
public class MainActivity extends Activity {
    // οι φωτογραφίες που θα εμφανιστούν
    Integer[] imageIDs = {
        R.drawable.image1,
        R.drawable.image2,
        R.drawable.image3,
        R.drawable.image4,
        R.drawable.image5,
        R.drawable.image6,
        R.drawable.image7,
        R.drawable.image8,
        R.drawable.image9
    };

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        Gallery gallery = (Gallery) findViewById(R.id.gallery1);
        gallery.setAdapter(new ImageAdapter(this));
        gallery.setOnItemClickListener(new OnItemClickListener() {
            public void onItemClick(AdapterView<?> parent, View v, int position,
                long id)
            {
                Toast.makeText(getBaseContext(),
                    "image" + (position + 1) + " selected",
                    Toast.LENGTH_SHORT).show();
            }
        });
    }
    //εμφάνιση των επιλεγμένων φωτογραφιών

    ImageView imageView = (ImageView) findViewById(R.id.image1);

    imageView.setImageResource(imageIDs[position]);
    }
}

```

```

public class ImageAdapter extends BaseAdapter {
    private Context context;
    private int itemBackground;
    public ImageAdapter(Context c)
    {
        context = c;
        //---εμφάνιση γκρι φόντου γύρω από τις φωτογραφίες
        TypedArray a =
            obtainStyledAttributes(R.styleable.MyGallery);
        itemBackground = a.getResourceId(
            R.styleable.MyGallery_android_galleryItemBackground, 0);
        a.recycle();
    }
    //επιστρέφει έναν αριθμό από φωτογραφίες
    public int getCount() {
        return imageIDs.length;
    }
    //---επιστρέφει το ID ενός αντικειμένου
    public Object getItem(int position) {
        return position;
    }
    //επιστρέφει το ID ενός αντικειμένου
    public long getItemId(int position) {
        return position;
    }
    //επιστρέφει ένα ImageView view
    public View getView(int position, View convertView, ViewGroup parent) {
        ImageView imageView = new ImageView(context);
        imageView.setImageResource(imageIDs[position]);
        imageView.setScaleType(ImageView.ScaleType.FIT_XY);
        imageView.setLayoutParams(new Gallery.LayoutParams(150, 120));
        imageView.setBackgroundResource(itemBackground);
        return imageView;
    }
}

```


3.4 Εφαρμογή αποστολής SMS

Στην εφαρμογή αυτή θα δούμε με ποιον τρόπο μπορούμε να στείλουμε ένα γραπτό μήνυμα SMS απευθείας από το κινητό τηλέφωνο android. Θα χρησιμοποιήσουμε την προεπιλεγμένη εφαρμογή της συσκευής. Το γραπτό μήνυμα καθώς και ο αριθμός του παραλήπτη είναι προετοιμασμένα από την εφαρμογή. Το μόνο που θα χρειαστεί είναι η συγγραφή σε κώδικα java. Δε θα χρειαστεί κανένα permission από το android manifest. Το μόνο που έχει να κάνει ο χρήστης είναι να πληκτρολογήσει το κουμπί της αποστολής μηνύματος μετά την εκτέλεση του κώδικα.



Εικόνα 3.4.1 Προβολή του μηνύματος «Hello Android!» μετά την εκτέλεση του κώδικα

3.4.1 Συγγραφή κώδικα

```
Main_Activity.java

package com.example.user.send_sms;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
public class MainActivity extends Activity {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Intent i = new Intent(android.content.Intent.ACTION_VIEW);
        i.putExtra("address", "6945891553");
        i.putExtra("sms_body", "Hello Android!");
        i.setType("vnd.android-dir/mms-sms");
        startActivity(i);
    }
}
```

3.4.2 Ανάλυση εφαρμογής

Αρχικά χρησιμοποιούμε ένα action (**android.content.Intent.ACTION_VIEW**). Δηλώνουμε τον τύπο του αντικειμένου intent σε ("**vnd.android-dir/mms-sms**"). Για να θέσουμε τον αριθμό τηλεφώνου του παραλήπτη χρησιμοποιούμε το «address». Με τη δήλωση του «sms_body» ορίζουμε το κείμενο το οποίο θέλουμε να γράψουμε στην οθόνη. Στην εφαρμογή έχουμε χρησιμοποιήσει έναν τυχαίο αριθμό τηλεφώνου ενώ ως μήνυμα το "**Hello Android!**". Θα μπορούσαμε να χρησιμοποιήσουμε περισσότερους τηλεφωνικούς αριθμούς ως παραλήπτες διατηρώντας όμως το ίδιο κείμενο για όλους.

3.5 Εφαρμογή Checkbox

3.5.1 Ανάλυση εφαρμογής

Στην εφαρμογή αυτή θα δούμε πώς χρησιμοποιούνται τα checkboxes. Τα checkboxes δίνουν τη δυνατότητα στους χρήστες να επιλέγουν από μία λίστα μία ή και περισσότερες επιλογές. Το checkbox διακρίνεται σε δύο καταστάσεις. Η πρώτη κατάσταση είναι αυτή κατά την οποία το checkbox είναι επιλεγμένο και η δεύτερη είναι αυτή κατά την οποία το checkbox δεν είναι επιλεγμένο. Τη δυνατότητα επιλογής του ή μη έχει αποκλειστικά ο χρήστης. Η εφαρμογή αυτή δεν απαιτεί κανένα permission απο το android manifest.



Εικόνα 3.5.1 Εκτέλεση κώδικα της εφαρμογής checkbox και εμφάνιση μηνύματος στην οθόνη της συσκευής

3.5.2 Συγγραφή κώδικα

Το στοιχείο checkbox ορίζει ένα αντικείμενο πάνω στην οθόνη το οποίο πρέπει να επιλεγεί ή όχι. Το android studio υποστηρίζει έναν τύπο checkbox που ονομάζεται star checkbox.

```
activity_main.xml
```

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <CheckBox
        android:id="@+id/chkAutosave"
```

```

    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Checkbox" />
<CheckBox
    android:id="@+id/star"
    style="?android:attr/starStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
</LinearLayout>

```

MainActivity.java

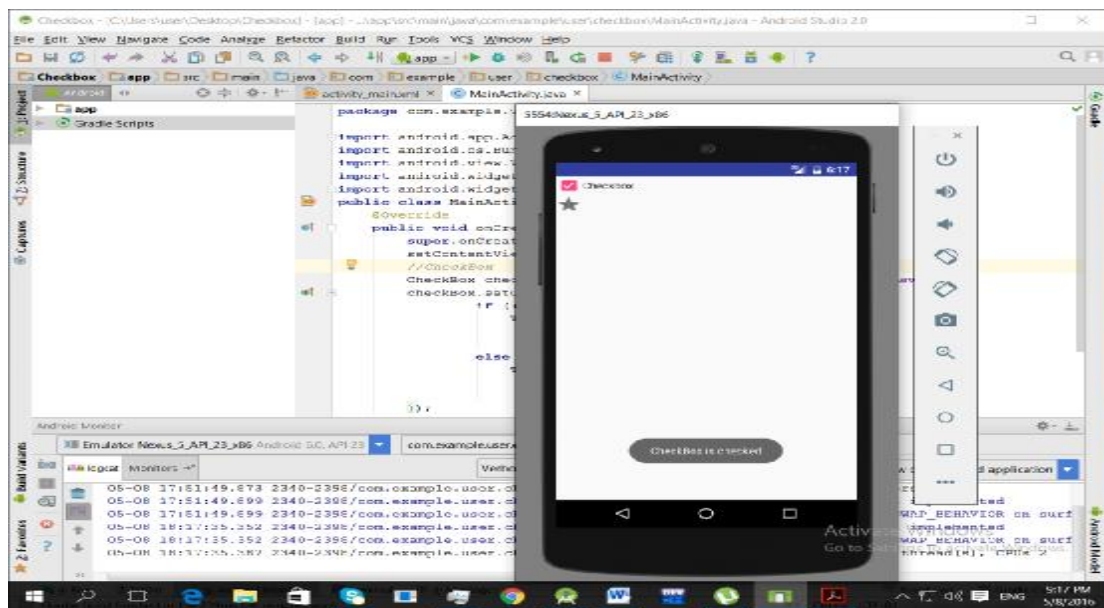
```

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.CheckBox;
import android.widget.Toast;
public class MainActivity extends Activity {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        //CheckBox
        CheckBox checkBox = (CheckBox) findViewById(R.id.chkAutosave);
        checkBox.setOnClickListener(new View.OnClickListener()
        {
            public void onClick(View v) {
                if (((CheckBox)v).isChecked())
                    Toast.makeText(getApplicationContext(),
                        "CheckBox is checked",
                        Toast.LENGTH_LONG).show();
                else
                    Toast.makeText(getApplicationContext(),
                        "CheckBox is unchecked",
                        Toast.LENGTH_LONG).show();
            }
        });
    }
}

```

Για να οριστεί η κατάσταση του checkbox πρέπει να ορίσουμε ένα onClick event. Αν το checkbox επιλεγθεί από τον χρήστη με τη μέθοδο `.isChecked()` επιστρέφει την τιμή `true`. Αν αποεπιλεγθεί επιστρέφει την τιμή `false`. Τέλος χρησιμοποιούμε την

κλαση toast. Με τη χρήση του toast εμφανίζεται στην οθόνη ένα μήνυμα το οποίο γράφει αν το checkbox έχει επιλεγθεί ή όχι ανάλογα με την επιλογή που θα δώσει ο χρήστης.



Εικόνα 3.5.2 Εμφάνιση του προσομοιωτή στο Android Studio και εκκίνηση εφαρμογής checkbox

³⁰ Deitel, P. (2012). ANDROID για προγραμματιστές. [χ.τ.]: Γκιουρδας. (19 Μαΐου, 17-5-2016). [www.stackoverflow.com](http://stackoverflow.com) Ανακτήθηκε 11 Ιουνίου, 2016, από <http://stackoverflow.com/questions/3299392/date-picker-in-android/>.

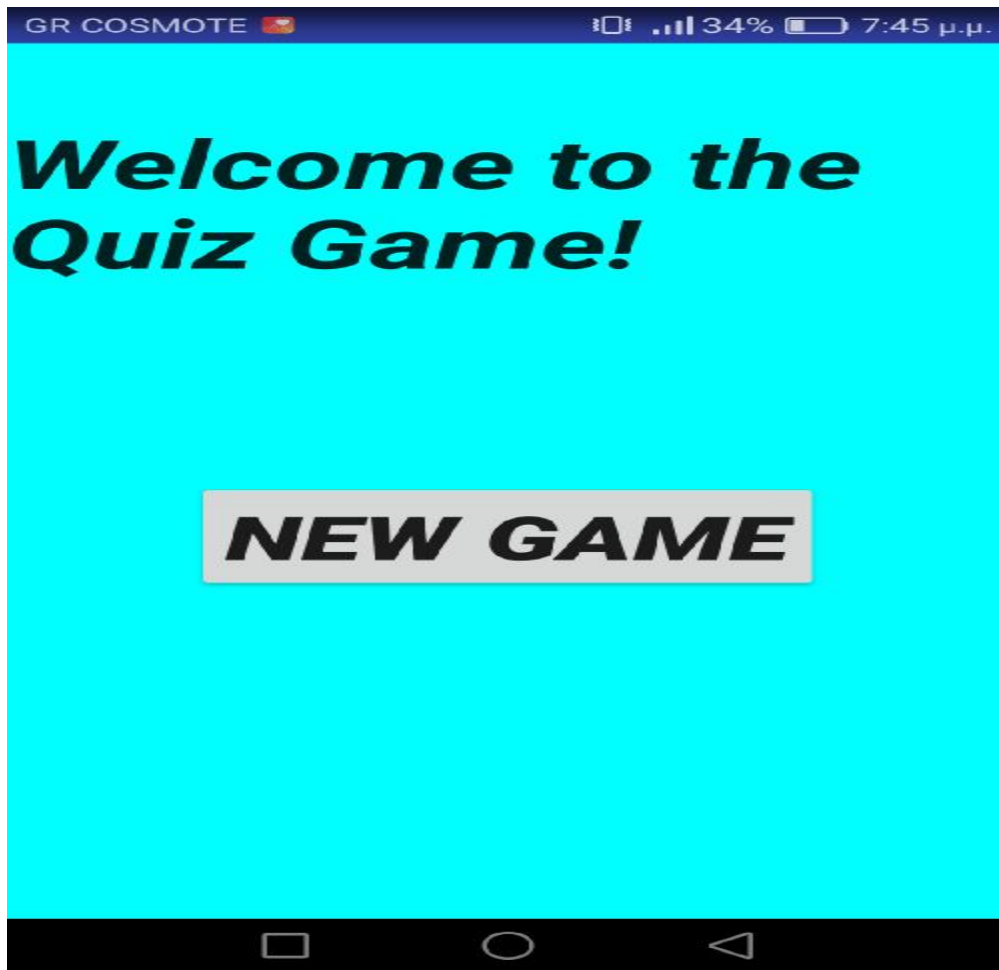
ΚΕΦΑΛΑΙΟ 4

Δημιουργία εφαρμογής σε Android

4.1 Δημιουργία εφαρμογής Quiz_Game

4.2 Ανάλυση απαιτήσεων και σχεδιασμός

Η εφαρμογή αυτή είναι ένα παιχνίδι κουίζ γνώσεων γενικού περιεχομένου. Η υλοποίησή της γίνεται με χρήση του προγράμματος Android Studio. Αρχικά εμφανίζεται μία οθόνη καλωσορίσματος όπου ο χρήστης πρέπει να πατήσει πάνω σε ένα κουμπί για να ξεκινήσει ένα νέο παιχνίδι. Στη συνέχεια ξεκινά το παιχνίδι όπου ο χρήστης διαβάζοντας την ερώτηση που δίνεται, απαντά και αμέσως εμφανίζεται ένα μήνυμα στην οθόνη το οποίο γράφει αν η απάντηση που επέλεξε είναι σωστή ή λανθασμένη. Πάνω δεξιά στην οθόνη υπάρχει ένα ακόμα κουμπί στο οποίο, αφού κάνει κλικ πάνω του ο χρήστης, περνά στην επόμενη ερώτηση. Σε κάθε ερώτηση υπάρχουν τέσσερις διαθέσιμες επιλογές με τη μορφή φωτογραφιών. Όταν ολοκληρωθεί το κουίζ εμφανίζεται στην οθόνη ένα μήνυμα το οποίο ενημερώνει ότι το παιχνίδι τελείωσε.



Εικόνα 4.1 Εμφάνιση οθόνης καλωσορίσματος κατα την έναρξη της εφαρμογής

4.3 Συγγραφή κώδικα

Αρχικά πρέπει να αποθηκεύσουμε όλες τις φωτογραφίες τις οποίες θα χρειαστούμε αργότερα στον φάκελο drawables του Android Studio ώστε να τις χρησιμοποιήσουμε μέσα στο πρόγραμμα. Σχεδιάζουμε ένα layout για να εμφανιστεί η οθόνη καλωσορίσματος και ένα κουμπί το οποίο να οδηγεί στις επόμενες σελίδες ή layouts. Αφού δημιουργήσουμε μία νέα κλάση `public class MainActivity extends Activity` {
με την εντολή

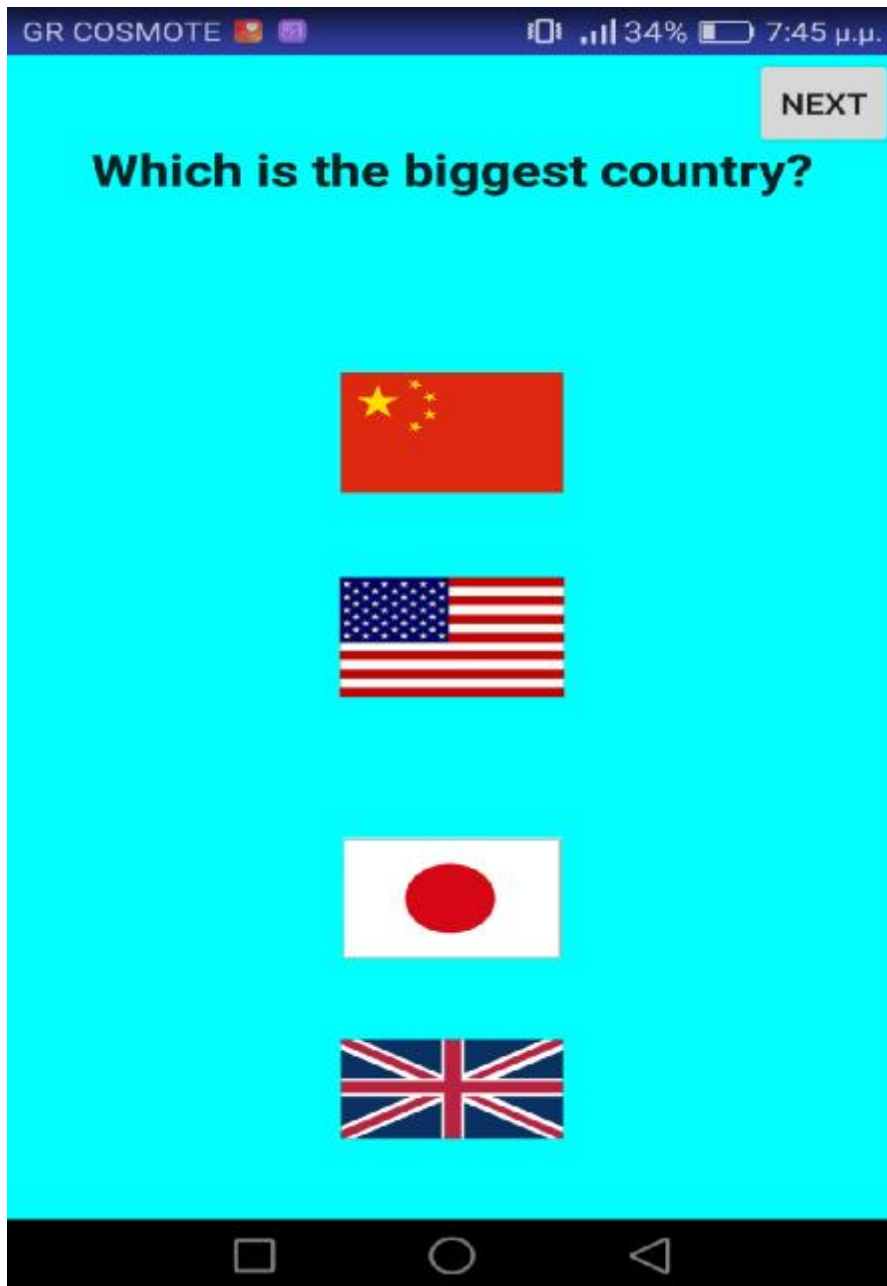
```
private final View.OnClickListener setLayout0Listener = new
View.OnClickListener(){
    @Override
    public void onClick(View view) {
        setContentView(R.layout.layout1);

        ((Button)findViewById(R.id.Btn1Id)).setOnClickListener(setLayout1Listen
er);
    }
};
```



```
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="New Game"
    android:id="@+id/Btn0Id"
    android:layout_below="@+id/textView3"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="129dp"
    android:enabled="true"
    android:focusable="false"
    android:textSize="40dp"
    android:textStyle="bold|italic" />
```

ορίζουμε ότι η εφαρμογή ξεκινά από το `layout0`(layout οθόνης καλωσορίσματος) και με το πάτημα του κουμπιού το οποίο έχει id `Btn0Id`, το πρόγραμμα περνά στο παιχνίδι ερωτήσεων δηλαδή στο επόμενο `layout(layout1)`. Με τον παραπάνω κώδικα στο αρχείο `layout0.xml` δημιουργούμε το αρχικό κουμπί με όνομα “New Game”:



Εικόνα 4.3.1 Εμφάνιση της πρώτης ερώτησης

Στη συνέχεια δημιουργούμε ένα ακόμη layout από όπου ξεκινά το παιχνίδι. Προσθέτουμε ένα κουμπί πάνω δεξιά το οποίο, με το πάτημα του χρήστη, περνά στην επόμενη ερώτηση και τέσσερα ImageButtons στα οποία τοποθετούμε τις εικόνες όπου ο χρήστης πρέπει να επιλέξει για να απαντήσει στην ερώτηση. Κάνουμε χρήση ενός Text View για να γράψουμε την ερώτηση. Με την παρακάτω εντολή στο αρχείο MainActivity.java

```

private final View.OnClickListener setLayout1Listener = new
View.OnClickListener() {
    @Override
    public void onClick(View view) {
        setContentView(R.layout.layout2);

        ((Button)findViewById(R.id.Btn2Id)).setOnClickListener(setLayout2Listener);
    }
};

```

ορίζουμε ως τρέχον layout το layout1 και αφού ο χρήστης απαντήσει στην ερώτηση, με το πάτημα του κουμπιού Next με id *Btn2Id* περνάμε στο επόμενο layout με την επόμενη ερώτηση. Στο φάκελο strings δημιουργούμε συμβολοσειρές για κάθε ImageButton όπως φαίνεται παρακάτω:

```

<resources>
    <string name="app_name">Quizgame7</string>
    <string name="chinaflag">Answer China</string>
    <string name="usaflag">Answer USA</string>
    <string name="japanflag">Answer Japan</string>
    <string name="ukflag">Answer UK</string>

</resources>

```

Γράφουμε ως όνομα για κάθε συμβολοσειρά το όνομα κάθε ImageButton όπως ακριβώς δηλώθηκε προηγουμένως στο φάκελο drawables κατά την αποθήκευση των φωτογραφιών και δίπλα την απάντηση. Μετατρέπουμε τα ImageButtons σε Zoom Buttons για να εμφανίζονται με σωστό τρόπο οι φωτογραφίες όταν εκτελεστεί ο κώδικας. Αφού ολοκληρωθεί αυτή η διαδικασία γράφουμε τον παρακάτω κώδικα στο αρχείο MainActivity.java:

```

}
public void onClickchinaflag(View view) {
    Toast.makeText(this, "Correct Answer!", Toast.LENGTH_SHORT).show();
}

public void onClickusaflag1(View view) {
    Toast.makeText(this, "Wrong! Try again",
    Toast.LENGTH_SHORT).show();
}

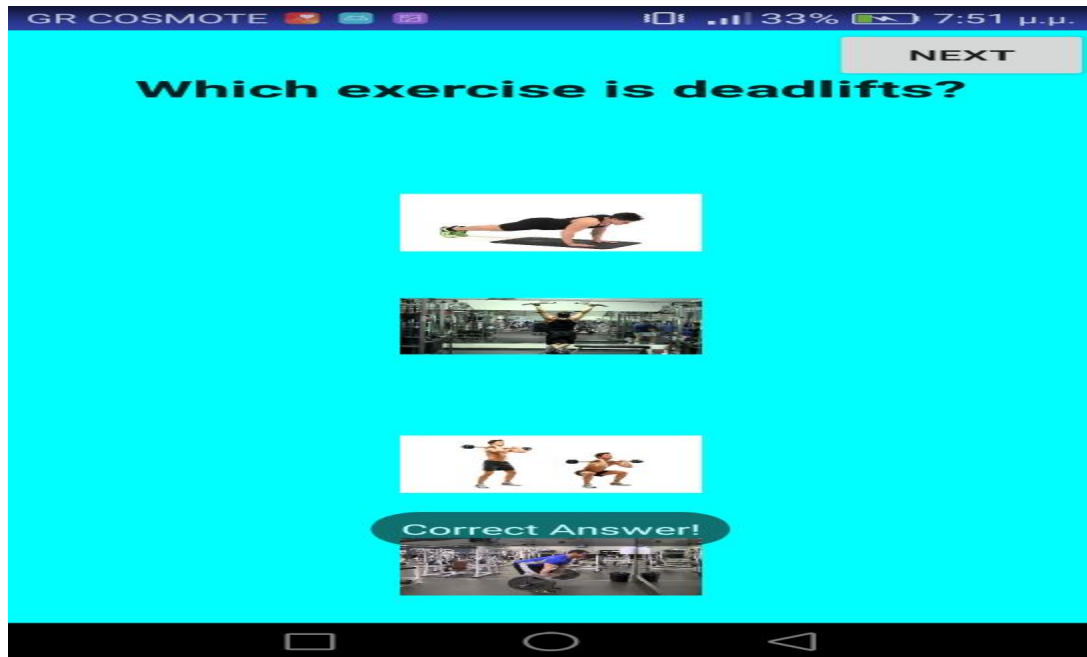
public void onClickjapanflag(View view) {
    Toast.makeText(this, "Wrong! Try again",
    Toast.LENGTH_SHORT).show();
}

public void onClickukflag1(View view) {
    Toast.makeText(this, "Wrong! Try again",
    Toast.LENGTH_SHORT).show();
}
}

```

Στην πρώτη γραμμή δηλώνεται μία public void όπου με το πάτημα του Zoom Button πλέον και με χρήση της εντολής `onClickchinaflag(View view)` εμφανίζεται στην οθόνη ένα

μήνυμα ("Correct Answer!") το οποίο ειδοποιεί τον χρήστη ότι επέλεξε τη σωστή απάντηση στην επόμενη γραμμή (`Toast.makeText(this, "Correct Answer!", Toast.LENGTH_SHORT).show()`). Αν ο χρήστης επιλέξει κάποιο από τα υπόλοιπα τρία Zoom Buttons εμφανίζεται στην οθόνη ένα μήνυμα λάθους με χρήση της ίδιας εντολής `Toast.makeText(this, "Wrong! Try again", Toast.LENGTH_SHORT).show()`;



Εικόνα 4.3.2 Επιλογή της σωστής απάντησης και εμφάνιση μηνύματος στην οθόνη

Έπειτα προσθέτουμε τις κατάλληλες βιβλιοθήκες για την αναγνώριση των εντολών από το πρόγραμμα:

```
import android.app.Activity;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;
```

Κατά τον ίδιο τρόπο δημιουργούμε άλλα εννέα layouts στα οποία χρησιμοποιούμε διαφορετικές φωτογραφίες ως Zoom Buttons αλλάζοντας κάθε φορά τα ids τους καθώς και τα ids του κουμπιού next. Μεταβάλλουμε επίσης το όνομα του τρέχοντος layout διατηρώντας ίδιο τον υπόλοιπο κώδικα για να γίνεται η μετάβαση από τη μία σελίδα στην άλλη. Τελος δημιουργούμε το τελευταίο layout το οποίο ειδοποιεί τον χρήστη ότι το παιχνίδι τελείωσε.

4.4 Τελικός κώδικας εφαρμογής

Αρχείο MainActivity.java

```
private final View.OnClickListener setLayout2Listener = new
View.OnClickListener() {
    @Override
    public void onClick(View view) {
        setContentView(R.layout.layout3);

        ((Button)findViewById(R.id.Btn3Id)).setOnClickListener(setLayout3Listen
er);
    }
};

private final View.OnClickListener setLayout3Listener = new
View.OnClickListener() {
    @Override
    public void onClick(View view) {
        setContentView(R.layout.layout4);

        ((Button)findViewById(R.id.Btn4Id)).setOnClickListener(setLayout4Listen
er);
    }
};

private final View.OnClickListener setLayout4Listener = new
View.OnClickListener() {
    @Override
    public void onClick(View view) {
        setContentView(R.layout.layout5);

        ((Button)findViewById(R.id.Btn5Id)).setOnClickListener(setLayout5Listen
er);
    }
};

private final View.OnClickListener setLayout5Listener = new
View.OnClickListener() {
    @Override
    public void onClick(View view) {
        setContentView(R.layout.layout6);

        ((Button)findViewById(R.id.Btn6Id)).setOnClickListener(setLayout6Listen
er);
    }
};

private final View.OnClickListener setLayout6Listener = new
View.OnClickListener() {
    @Override
    public void onClick(View view) {
        setContentView(R.layout.layout7);

        ((Button)findViewById(R.id.Btn7Id)).setOnClickListener(setLayout7Listen
er);
    }
};
```

```

private final View.OnClickListener setLayout7Listener = new
View.OnClickListener() {
    @Override
    public void onClick(View view) {

```

```

public void onClickchinaflag(View view) {
    Toast.makeText(this, "Correct Answer!",
Toast.LENGTH_SHORT).show();
}

public void onClickusaflag1(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}

public void onClickjapanflag(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}
public void onClickukflag1(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}

public void onClickelephant3(View view) {
    Toast.makeText(this, "Correct Answer!",
Toast.LENGTH_SHORT).show();
}

public void onClicktiger(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}

public void onClickgorilla(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}
public void onClickanaconda(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}

public void onClickarmwrestling(View view) {
    Toast.makeText(this, "Correct Answer!",
Toast.LENGTH_SHORT).show();
}

public void onClicktennis(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}

public void onClickarchery(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}
public void onClickwrestling(View view) {

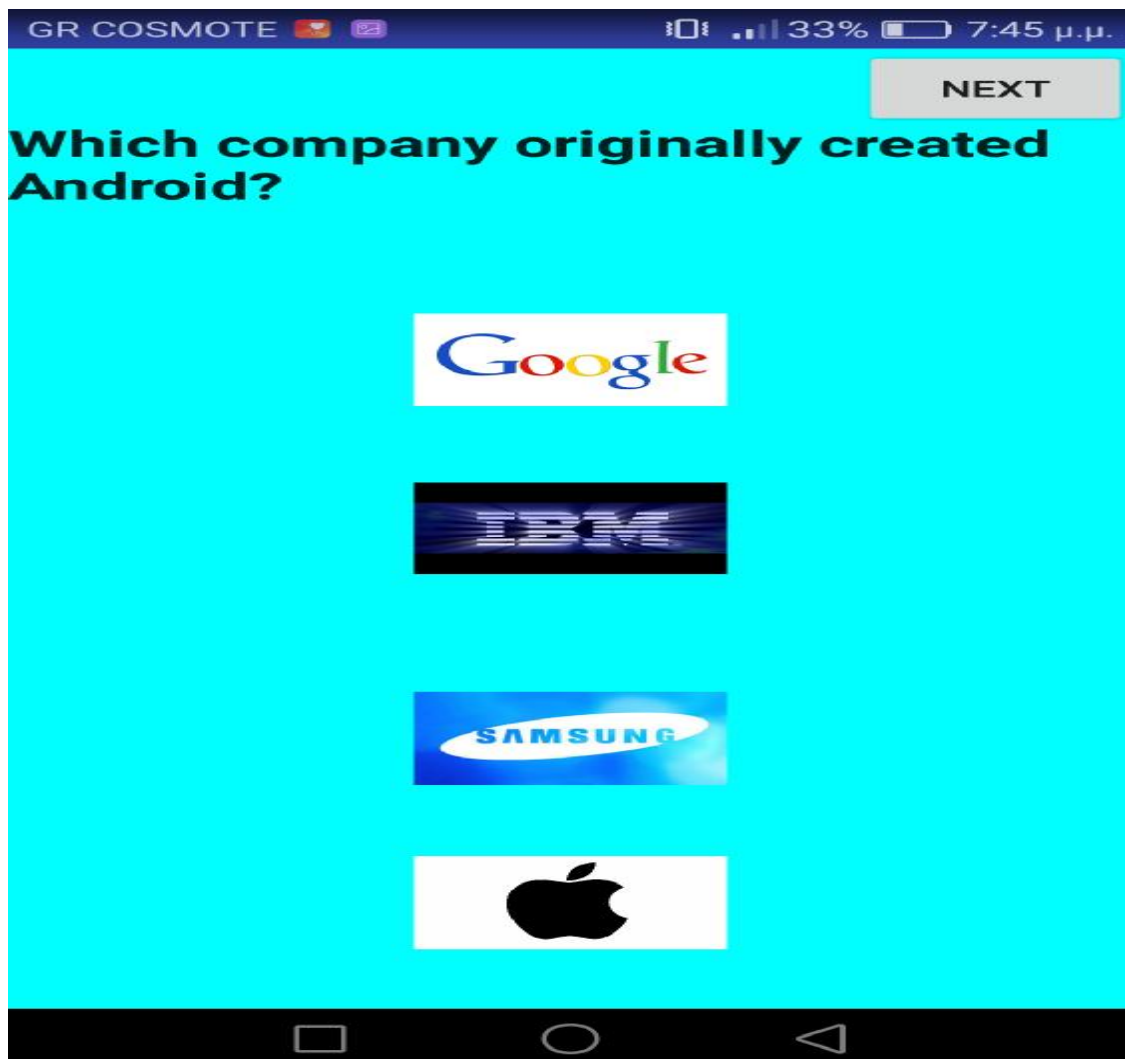
```

```
        Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
    }

    public void onClickmetaxas(View view) {
        Toast.makeText(this, "Correct Answer!",
Toast.LENGTH_SHORT).show();
    }

    public void onClickpapadopoulos(View view) {
        Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
    }

    public void onClicktrikoupis(View view) {
        Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
    }
}
```



Εικόνα 4.3.3 Ερώτηση κατά τη διάρκεια του παιχνιδιού

```

public void onClickvenizelos(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}

    public void onClickparthenonas(View view) {
        Toast.makeText(this, "Correct Answer!",
Toast.LENGTH_SHORT).show();
    }

    public void onClickagalmdios(View view) {
        Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
    }

    public void onClickkolossosrodou(View view) {
        Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
    }
    public void onClickpiramidagkizas(View view) {
        Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
    }

    public void onClickceltics(View view) {
        Toast.makeText(this, "Correct Answer!",
Toast.LENGTH_SHORT).show();
    }

    public void onClicklakers(View view) {
        Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
    }

    public void onClickspurs(View view) {
        Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
    }
    public void onClickbulls(View view) {
        Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
    }

    public void onClickgoogle(View view) {
        Toast.makeText(this, "Correct Answer!",
Toast.LENGTH_SHORT).show();
    }

    public void onClicksamsung(View view) {
        Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
    }

    public void onClickapple(View view) {
        Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
    }
    public void onClickibm(View view) {
        Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
    }
}

```



```

public void onClickindia(View view) {
    Toast.makeText(this, "Correct Answer!",
Toast.LENGTH_SHORT).show();
}

public void onClickpakistan(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}

public void onClickafghanistan(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}
public void onClickmalaysia(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}

public void onClickmaserati(View view) {
    Toast.makeText(this, "Correct Answer!",
Toast.LENGTH_SHORT).show();
}

public void onClickaudi(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}

public void onClickchevrolet(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}
public void onClickmitsubishi(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}

public void onClickdeadlifts(View view) {
    Toast.makeText(this, "Correct Answer!",
Toast.LENGTH_SHORT).show();
}

public void onClickpushups(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}

public void onClickpullups(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}
public void onClicksquats(View view) {
    Toast.makeText(this, "Wrong! Try again",
Toast.LENGTH_SHORT).show();
}}

```

Αρχείο layout1.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="#00ffff"

tools:context="com.example.user.quizgame7.MainActivity">

<Button
    style="?android:attr/buttonStyleSmall"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Next"
    android:id="@+id/Btn1Id"
    android:layout_alignParentTop="true"
    android:layout_alignParentEnd="true" />

<ZoomButton
    android:id="@+id/imageButton1"
    android:layout_width="120dp"
    android:layout_height="60dp"
    android:src="@drawable/chinaflag"

    android:layout_above="@+id/imageButton2"
    android:layout_centerHorizontal="true"
    android:layout_marginBottom="42dp"
    android:contentDescription="@string/chinaflag"
    android:onClick="onClickchinaflag"
    android:layout_gravity="center"
    android:adjustViewBounds="false"
    android:focusable="false" />

<ZoomButton
    android:id="@+id/imageButton2"
    android:layout_width="100dp"
    android:layout_height="60dp"
    android:src="@drawable/usaflag1"
    android:layout_centerVertical="true"
    android:layout_centerHorizontal="true"
    android:contentDescription="@string/usaflag"
    android:onClick="onClickusaflag1"
    android:adjustViewBounds="false"
    android:focusable="false" />

<ZoomButton
    android:id="@+id/imageButton3"
    android:layout_width="160dp"
    android:layout_height="60dp"
    android:src="@drawable/japanflag"
    android:contentDescription="@string/japanflag"
    android:onClick="onClickjapanflag"
    android:adjustViewBounds="false"
    android:layout_above="@+id/imageButton4"
    android:layout_centerHorizontal="true"
    android:focusable="false" />

<ZoomButton
    android:id="@+id/imageButton4"
    android:layout_width="90dp"
    android:layout_height="130dp"
    android:src="@drawable/ukflag1"
    android:contentDescription="@string/ukflag"
    android:onClick="onClickukflag1"
```

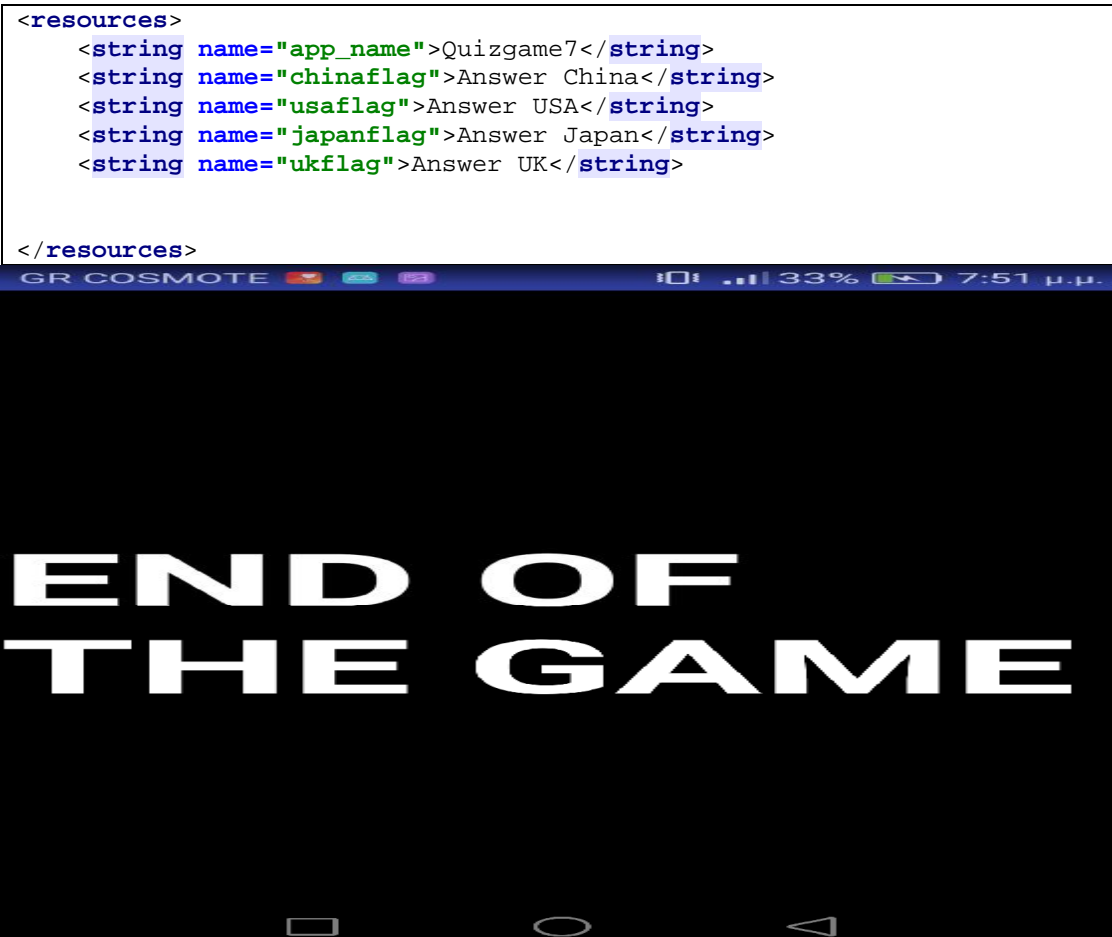
```

        android:adjustViewBounds="false"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:focusable="false" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:text="Which is the biggest country?"
    android:id="@+id/textView"
    android:textStyle="bold"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="42dp" />
</RelativeLayout>

```

Αρχείο strings.xml



Εικόνα 4.3.4 Εμφάνιση μηνύματος τέλους στην οθόνη

Πηγές

- ¹ «Open Handset Alliance». Open Handset Alliance. Ανακτήθηκε στις 6 Νοεμβρίου 2007.
Jackson, Rob (10 Δεκεμβρίου 2008). «Sony Ericsson, HTC Androids Set For Summer 2009». Android Phone Fans. Ανακτήθηκε στις 3 Σεπτεμβρίου 2009.
- ² «Smartphone» Phone Scoop. Ανακτήθηκε στις 2011-12-15.
«Feature Phone». Phone Scoop. Ανακτήθηκε στις 2011-12-15.
Andrew Nusca (20 August 2009). «Smartphone vs. feature phone arms race heats up; which did you buy?». ZDNet. Ανακτήθηκε στις 2011-12-15.
- ³ Why the Android Platform Will Beat the iPhone
Android versus iPhone 3.0: The Showdown
- ⁴ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Μαΐου, 2016, από <http://www.doctorandroid.gr/2015/10/android-60-marshmallow-overview.html>.
- ⁵ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Μαΐου, 2016, από <http://www.doctorandroid.gr/2016/03/how-to-voice-typing.html>.
- ⁶ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/02/android-device-manager.html>.
- ⁷ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/01/android-basics-root.html>.
- ⁸ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/01/android-basics-benchmark.html>.
- ⁹ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2015/09/backup-restore.html>.
- ¹⁰ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/03/android-pin-app-feature.html>.
- ¹¹ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2015/09/backup-restore.html>.
- ¹² Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2013/06/android-basics-new-emulated-file-system.html>.
- ¹³ Συντάκτες Dictionary.com, «δισκίο υπολογιστή - ένα αποτέλεσμα λεξικό" Dictionary.com , ανακτήθηκε 17 του Απρίλη 2010
- ¹⁴ «Smartphone». Phone Scoop. Ανακτήθηκε στις 2011-12-15.
«Feature Phone». Phone Scoop. Ανακτήθηκε στις 2011-12-15.

Andrew Nusca (20 August 2009). «Smartphone vs. feature phone arms race heats up; which did you buy?». ZDNet. Ανακτήθηκε στις 2011-12-15

¹⁵ Greece, A. (15 Μαΐου, 15-5-2016). www.greeceandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.greeceandroid.gr/dev/1595-android-n-file-manager>.

¹⁶ Greece, A. (15 Μαΐου, 15-5-2016). www.greeceandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.greeceandroid.gr/dev/1594-android-n-calibrate-screen-feature>.

¹⁷ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/03/remix-mini-os-review.html>.

¹⁸ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/04/clear-data-on-apps.html>.

¹⁹ Android, H. (29 Μαΐου, 29-5-2016). www.androidhellas.gr. Ανακτήθηκε 11 Ιουνίου, 11-6-2016, από <http://androidhellas.com/2016/04/07/metafora-epafon-efarmogon-arxeion-kainourgio-telefon/>.

²⁰ Android, H. (29 Μαΐου, 29-5-2016). www.androidhellas.gr. Ανακτήθηκε 11 Ιουνίου, 11-6-2016, από <http://androidhellas.com/2016/03/15/android-material-nea-bottom-navigation-bar/>.

²¹ Android, H. (29 Μαΐου, 29-5-2016). www.androidhellas.gr. Ανακτήθηκε 11 Ιουνίου, 11-6-2016, από <http://androidhellas.com/2016/03/11/split-screen-multitasking-android-n/>.

²² Android, A. (26 Μαΐου, 26-5-2016). www.android.gr. Ανακτήθηκε 11 Ιουνίου, 11-6-2016, από <https://www.android.gr/android-news/logismiko/i-microsoft-etimazi-rom-pou-tha-feri-ta-windows-10-stis-android-siskeves>.

²³ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2012/02/android-basics-bootloader.html>.

²⁴ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/03/android-n-ify-xposed.html>

²⁵ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2015/11/fastboot-flashing-command.html>.

²⁶ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2015/11/android-60-theme-engine-rro.html>.

²⁷ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2015/09/xposed-google-music-xgpm.html>.

²⁸ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2016/01/remix-os-android.html>.

²⁹ Android, D. (28 Μαΐου, 28-5-2016). www.doctorandroid.gr. Ανακτήθηκε 11 Ιουνίου, 2016, από <http://www.doctorandroid.gr/2015/09/remove-malware-android.html>.

³⁰ Deitel, P. (2012). ANDROID για προγραμματιστες. [χ.τ.]: Γκιουρδας. (19 Μαΐου, 17-5-2016). www.stackoverflow.com Ανακτήθηκε 11 Ιουνίου, 2016, από <http://stackoverflow.com/questions/3299392/date-picker-in-android/>.