



ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ανάπτυξη εφαρμογής για έξυπνο κινητό παρουσίασης σε πραγματικό χρόνο πληροφορίες μόλυνσης περιβάλλοντος



Του φοιτητή

Γρηγορίου Χρήστου

Αρ. Μητρώου: 05/2914

Επιβλέπων καθηγητής

Δρ. Ηλιούδης Χρήστος

Θεσσαλονίκη 2014

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή μου Δρ. Χρήστο Ηλιούδη, για την εμπιστοσύνη που μου έδειξε, την καθοδήγηση και την υποστήριξη που μου προσέφερε, καθώς και για την ευγενική του ανταπόκριση και πολύτιμη βοήθεια στην επίλυση των αποριών και διαφόρων θεμάτων που προέκυψαν κατά την διαμόρφωση της παρούσας πτυχιακής εργασίας.

Τέλος, θα ήθελα να απευθύνω τις ευχαριστίες μου στους Φίλους μου για την συμπαράσταση τους και ιδιαίτερα στην Οικογένεια μου, η οποία στήριξε την προσπάθεια μου οικονομικά και κυρίως ηθικά όχι μόνο κατά την διάρκεια της εκπόνησης αυτής της εργασίας αλλά και καθ' όλη την διάρκεια των σπουδών μου φροντίζοντας για την καλύτερη δυνατή μόρφωσή μου, δίνοντάς μου κουράγιο να προχωρώ και να υπερπηδώ κάθε εμπόδιο για να φτάσω στο στόχο μου.

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΥΧΑΡΙΣΤΙΕΣ	2
ΠΕΡΙΕΧΟΜΕΝΑ	3
Ευρετήριο σχημάτων	5
ΚΕΦΑΛΑΙΟ 1.....	6
ΕΙΣΑΓΩΓΗ.....	6
1.1 ΕΦΑΡΜΟΓΕΣ ΣΤΑ «ΕΞΥΠΝΑ» ΤΗΛΕΦΩΝΑ.....	6
1.2 ΣΗΜΑΝΤΙΚΟΤΗΤΑ ΕΦΑΡΜΟΓΩΝ	7
1.3 ΚΑΙΝΟΤΟΜΕΣ ΥΠΗΡΕΣΙΕΣ	8
1.4 ΣΤΟΧΟΣ ΕΡΓΑΣΙΑΣ.....	10
1.5 Η ΥΠΗΡΕΣΙΑ POLLUTION TRACKER	11
1.6 ΔΟΜΗ ΕΡΓΑΣΙΑΣ	12
ΚΕΦΑΛΑΙΟ 2.....	13
ΚΑΙΝΟΤΟΜΙΑ – ΕΞΥΠΝΑ ΚΙΝΗΤΑ	13
ΕΙΣΑΓΩΓΗ.....	13
2.1 ΤΙ ΕΙΝΑΙ ΚΑΙΝΟΤΟΜΙΑ.....	14
2.2 ΤΕΧΝΟΛΟΓΙΚΗ ΚΑΙΝΟΤΟΜΙΑ.....	15
2.3 ΚΑΙΝΟΤΟΜΙΑ ΠΡΟΙΟΝΤΟΣ	17
2.3.1 ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΕΝΟΣ ΚΑΙΝΟΤΟΜΟΥ ΠΡΟΙΟΝΤΟΣ.....	17
2.4 ΠΡΟΣΩΠΟΠΟΙΗΜΕΝΟ ΠΕΡΙΕΧΟΜΕΝΟ	17
2.4.1 ΟΦΕΛΗ ΠΡΟΣΩΠΟΠΟΙΗΣΗΣ ΠΕΡΙΕΧΟΜΕΝΟΥ	19
2.4.2 ΠΡΟΥΠΟΘΕΣΕΙΣ ΠΡΟΣΩΠΟΠΟΙΗΣΗΣ.....	19
ΕΠΙΛΟΓΟΣ.....	20
ΚΕΦΑΛΑΙΟ 3.....	21
ΥΦΙΣΤΑΜΕΝΕΣ ΛΥΣΕΙΣ	21
ΕΙΣΑΓΩΓΗ.....	21
3.1 ΜΕΛΕΤΗ ΠΕΡΙΠΤΩΣΗΣ AIR QUALITY	22
3.2 ΜΕΛΕΤΗ ΠΕΡΙΠΤΩΣΗΣ ΗΚ ΑQHI (HONG KONG AIR QUALITY INDEX).....	26
3.2.1 ΕΙΣΑΓΩΓΙΚΗ ΟΘΟΝΗ & ΟΘΟΝΗ ΜΕΤΡΗΣΕΩΝ	27
3.2.2 ΑΛΛΕΣ ΟΘΟΝΕΣ	28
3.3 ΣΥΓΚΡΙΤΙΚΗ ΑΞΙΟΛΟΓΗΣΗ ΥΦΙΣΤΑΜΕΩΝ ΛΥΣΕΩΝ.....	28
3.4 ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΥΠΗΡΕΣΙΑΣ POLLUTIONTRACKER	29
ΕΠΙΛΟΓΟΣ.....	31

ΚΕΦΑΛΑΙΟ 4.....	32
ΑΝΑΛΥΣΗ ΥΠΗΡΕΣΙΑΣ	32
ΕΙΣΑΓΩΓΗ.....	32
4.1 ΠΙΝΑΚΑΣ ΧΡΗΣΤΩΝ - ΣΤΟΧΩΝ.....	33
4.2 ΔΙΑΓΡΑΜΜΑ ΠΕΡΙΠΤΩΣΕΩΝ ΧΡΗΣΗΣ	33
4.3 ΛΕΚΤΙΚΗ ΠΕΡΙΓΡΑΦΗ ΠΕΡΙΠΤΩΣΗΣ ΧΡΗΣΗΣ	34
4.4 ΜΟΝΤΕΛΟ ΠΕΡΙΟΧΗΣ ΠΡΟΒΛΗΜΑΤΟΣ – ΕΝΝΟΙΟΛΟΓΙΚΟ ΜΟΝΤΕΛΟ	35
4.5 ΔΙΑΓΡΑΜΜΑ ΑΚΟΛΟΥΘΙΑΣ.....	36
4.6 ΔΙΑΓΡΑΜΜΑ ΚΛΑΣΕΩΝ.....	37
ΕΠΙΛΟΓΟΣ.....	40
ΚΕΦΑΛΑΙΟ 5.....	41
ΤΕΧΝΟΛΟΓΙΚΟ ΠΕΡΙΒΑΛΛΟΝ ΑΝΑΠΤΥΞΗΣ ΚΑΙ ΛΕΙΤΟΥΡΓΙΑΣ.....	41
ΕΙΣΑΓΩΓΗ.....	41
5.1 ΕΡΓΑΛΕΙΑ ΑΝΑΠΤΥΞΗΣ ANDROID	42
5.2 ΤΕΧΝΟΛΟΓΙΚΟ ΠΕΡΙΒΑΛΛΟΝ ΑΝΑΠΤΥΞΗΣ	43
5.3 ΛΕΙΤΟΥΡΓΙΑ ΥΠΗΡΕΣΙΑΣ POLLUTIONTRACKER	46
ΕΠΙΛΟΓΟΣ.....	46
ΚΕΦΑΛΑΙΟ 6.....	47
ΠΕΡΙΓΡΑΦΗ ΥΠΗΡΕΣΙΑΣ	47
ΕΙΣΑΓΩΓΗ.....	47
6.1 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΥΠΗΡΕΣΙΑΣ.....	47
6.2 ΤΡΟΠΟΣ ΛΕΙΤΟΥΡΓΙΑΣ ΥΠΗΡΕΣΙΑΣ	48
ΕΠΙΛΟΓΟΣ.....	53
ΚΕΦΑΛΑΙΟ 7.....	54
ΚΑΙΝΟΤΟΜΙΑ ΚΑΙ ΠΡΟΣΤΙΘΕΜΕΝΗ ΑΞΙΑ ΥΠΗΡΕΣΙΑΣ	54
ΕΙΣΑΓΩΓΗ.....	54
7.1 ΣΗΜΕΙΑ ΚΑΙΝΟΤΟΜΙΑΣ	54
7.2 ΠΡΟΣΤΙΘΕΜΕΝΗ ΑΞΙΑ ΕΝΑΤΙ ΑΛΛΩΝ ΥΠΗΡΕΣΙΩΝ.....	55
ΣΥΜΠΕΡΑΣΜΑΤΑ – ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ	56
ΠΑΡΑΡΤΗΜΑΤΑ.....	57
ΠΑΡΑΡΤΗΜΑ Α	57
ΠΑΡΑΡΤΗΜΑ Β	72
ΑΝΑΦΟΡΕΣ	82
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	84

Ευρετήριο σχημάτων

Σχήμα 1 Downloads εφαρμογών 2008 έως 2013 (eMarketer).....	8
Σχήμα 2 Ορισμός Καινοτομίας - Ολιστικό Μοντέλο [2].....	14
Σχήμα 3 Η χρήση πληροφοριών με βάση την τοποθεσία - γεω - κοινωνικές πληροφορίες [19]....	21
Σχήμα 4 Χρήστες γεω - κοινωνικών υπηρεσιών vs χρήστες υπηρεσιών (based location) [19].....	22
Σχήμα 5 Συνολική λειτουργία μελέτη περίπτωσης Air Quality [15]	23
Σχήμα 6 Καρτέλα Map [15]	24
Σχήμα 7 Καρτέλα Graph [15].....	25
Σχήμα 8 Καρτέλα Sensor Box [15].....	25
Σχήμα 9 Αρχική Οθόνη ΗΚ ΑQH1	27
Σχήμα 10 Οθόνη Μετρήσεων.....	27
Σχήμα 11 Οθόνη Πρόγνωσης (forecast).....	28
Σχήμα 12 Διαδραστικός Χάρτης ΚΕΠΕ (www.kepekozani.gr)	30
Σχήμα 13 Διαδικασία Ανάλυσης [17]	32
Σχήμα 14 Πίνακας Χρηστών	33
Σχήμα 15 Διάγραμμα Περίπτωσης Χρήσης.....	34
Σχήμα 16 Εννοιολογικό Μοντέλο.....	36
Σχήμα 17 Διάγραμμα Ακολουθίας	37
Σχήμα 18 Διάγραμμα Κλάσεων	39
Σχήμα 19 Επίπεδα Google Android.....	41
Σχήμα 20 Δοκιμαστικό Project.....	44
Σχήμα 21 Eclipse IDE	45
Σχήμα 22 Android Virtual Device Manager	46
Σχήμα 23 Αρχιτεκτονική Υπηρεσίας PollutionTracker	47
Σχήμα 24 Αρχική Οθόνη	48
Σχήμα 25 Μήνυμα μη ανανέωσης δεδομένων.....	49
Σχήμα 26 Εναλλακτικός τρόπος μεταφοράς στις Ρυθμίσεις	49
Σχήμα 27 Οθόνη Ρυθμίσεων	50
Σχήμα 28 Οθόνη Ρυθμίσεων - Μη εισαγωγή στοιχείων.....	51
Σχήμα 29 Αρχική Οθόνη - Επιλογή Τοποθεσίας.....	51
Σχήμα 30 Αρχική Οθόνη - Διαδικασία προβολής τοποθεσίας.....	52
Σχήμα 31 Αποτελεσμάτων - Τοποθεσία "Ακρινή"	52

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

1.1 ΕΦΑΡΜΟΓΕΣ ΣΤΑ «ΕΞΥΠΝΑ» ΤΗΛΕΦΩΝΑ

Τα κινητά τηλέφωνα βρίσκονται παντού στην ζωή μας και ο αριθμός τους συνεχίζεται να αυξάνεται σημαντικά. Πλέον, εκτός από τα απλά κινητά τηλέφωνα υπάρχουν και τα λεγόμενα «έξυπνα» τηλέφωνα (smart phones). Το «έξυπνο» τηλέφωνο είναι ένα πολυλειτουργικό κινητό τηλέφωνο που παρέχει επικοινωνία μέσω φωνής και μηνυμάτων κειμένου, διευκολύνει την επεξεργασία δεδομένων και έχει αυξημένη δυνατότητα ασύρματης σύνδεσης. Επιπλέον, δίνουν την δυνατότητα αποστολής μηνυμάτων πολυμέσων (MMS), έλεγχο ηλεκτρονικού ταχυδρομείου, αποθήκευσης αρχείων βίντεο κ.α. Μπορούμε να πούμε ότι ένα «έξυπνο» τηλέφωνο αποτελεί το συνδυασμό ενός ισχυρού κινητού τηλεφώνου και ενός προσωπικού ψηφιακού βοηθού (personal digital assistants, PDA) με δυνατότητα ασύρματης σύνδεσης.

Σημαντικό ρόλο σε ένα «έξυπνο» τηλέφωνο διαδραματίζουν οι εφαρμογές τις οποίες έχει. Εφαρμογή κινητού τηλεφώνου ονομάζουμε το λογισμικό το οποίο σχεδιάστηκε προκειμένου να λειτουργεί σε «έξυπνα» τηλέφωνα, τάμπλετ και άλλες φορητές συσκευές.

Οι εφαρμογές αυτές είναι συνήθως διαθέσιμες από πλατφόρμες διανομής εφαρμογών που άρχισαν να εμφανίζονται το 2008 και λειτουργούν συνήθως υπό την ιδιοκτησία εταιριών που έχουν αναπτύξει λειτουργικά συστήματα για κινητές συσκευές (Apple App Store, Google Play, Windows Phone Store κ.α.). Κάποιες εφαρμογές είναι ελεύθερες ενώ κάποιες άλλες είναι διαθέσιμες με την καταβολή κάποιου χρηματικού ποσού. Συνήθως, ο χρήστης τις κατεβάζει στη κινητή του συσκευή αλλά υπάρχει και η δυνατότητα να τις κατεβάσει από τον φορητό ή τον επιτραπέζιο υπολογιστή του.

Παγκοσμίως συνηθίζεται να χρησιμοποιείται για τις εφαρμογές στις κινητές συσκευές ο όρος «app» ο οποίος είναι μια συντομογραφία του όρου «application software» (λογισμικό εφαρμογής). Ο όρος αυτός έγινε πολύ δημοφιλής το 2010 και έγινε η λέξη της χρονιάς από το αρμόδιο οργανισμό παρακολούθησης την Αμερικανικής γλώσσας (American Dialect Society). Το 2009, ο αρθογράφος David Pogue πρότεινε τα νεότερα «έξυπνα» τηλέφωνα να ονομαστούν «τηλέφωνα εφαρμογών» για να γίνεται η διάκριση με τα πρώτα μη εξελιγμένα «έξυπνα» τηλέφωνα.

Αρχικά, οι εφαρμογές κινητών συσκευών προσέφεραν υπηρεσίες ηλεκτρονικού ταχυδρομείου, επαφών και πληροφορίες που είχαν να κάνουν με την τρέχουσα ημέρα, το χρηματιστήριο καθώς και πληροφορίες για τον καιρό. Ωστόσο, η συνεχή αύξηση ζήτησης εφαρμογών εκ μέρους των χρηστών καθώς και τα εργαλεία ανάπτυξης εφαρμογών που συνεχώς εμπλουτίζονται οδήγησε τις εφαρμογές να καλύψουν και άλλες κατηγορίες, όπως παιχνίδια, υπηρεσίες βασισμένες στην εύρεση τοποθεσίας μέσω GPS, υπηρεσίες τραπεζικών συναλλαγών, υπηρεσίες αγοράς εισιτηρίων καθώς και ιατρικές εφαρμογές. Αυτή η έκρηξη στην ανάπτυξη των εφαρμογών προκάλεσε την δημιουργία blog, περιοδικών και άλλων on line υπηρεσιών που αξιολογούν αυτές τις εφαρμογές μέσα από κριτικές.

1.2 ΣΗΜΑΝΤΙΚΟΤΗΤΑ ΕΦΑΡΜΟΓΩΝ

Οι χρήστες στην σημερινή εποχή βρίσκονται σε διαρκή κίνηση και χρησιμοποιούν τις εφαρμογές για να ψάξουν ότι επιθυμούν και να έχουν ότι χρειάζονται στα χέρια τους. Γι αυτό και οι εφαρμογές «έξυπνων» κινητών είναι τόσο σημαντικές.

Ο χρήστης μπορεί να κατεβάσει εφαρμογές είτε για να διασκεδάσει είτε για να ενημερωθεί. Το πλέον εντυπωσιακό είναι ότι σε πολύ μικρό χρονικό διάστημα έγιναν μέρος της ζωής μας μέσω των «έξυπνων» τηλεφώνων.

Σε μια εποχή που το Διαδίκτυο έχει εδραιωθεί και αποτελεί απαραίτητο αγαθό για την καθημερινή ζωή των ανθρώπων είναι σημαντικό ο χρήστης να έχει πρόσβαση άμεσα και γρήγορα στις αγαπημένες του ιστοσελίδες. Η εμπειρία αυτή είναι πολύ διασκεδαστική τόσο μέσω της πλατφόρμας iOS όσο και μέσω της πλατφόρμας Android. Επίσης με την εξέλιξη των ασύρματων τεχνολογιών αυτή η εμπειρία γίνεται ακόμα πιο γρήγορη και παρέχει μεγαλύτερα πλεονεκτήματα στους χρήστες.

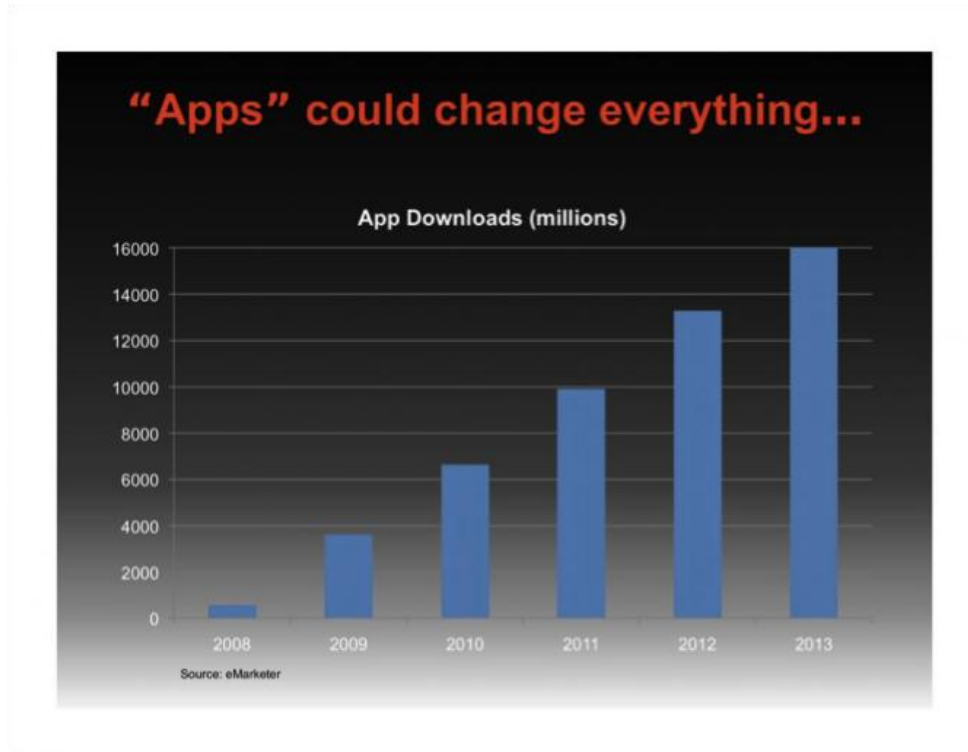
Η ποικιλία που υπάρχει σε εφαρμογές κάνει και τους πιο απαιτητικούς χρήστες να ικανοποιηθούν. Ένας χρήστης μπορεί να παίξει παιχνίδια, να αγοράσει ή να διαβάσει βιβλία στις διακοπές του, να ενημερωθεί για πληροφορίες που αναζητά. Οι επιχειρήσεις ολοένα και περισσότερο παρέχουν εφαρμογές προκειμένου να ικανοποιήσουν την κινητικότητα των χρηστών. Αυτό γίνεται προκειμένου οι επιχειρήσεις να:

- Κερδίσουν την εμπιστοσύνη των χρηστών
- Να αυξήσουν τους τρόπους προσβασιμότητας στην υπηρεσία που παρέχουν
- Να αυξήσουν τις διαδικτυακές πωλήσεις τους
- Να επικοινωνούν καλύτερα με τους καταναλωτές

Αυτά είναι τα κυριότερα στοιχεία που κάνουν μια εφαρμογή «έξυπνων» τηλεφώνων σημαντική για τις επιχειρήσεις. Από την άλλη μεριά μια εφαρμογή δίνει στους χρήστες την δυνατότητα να:

- Να έχουν ευκολότερη πρόσβαση σε μια υπηρεσία
- Να δέχονται ειδοποιήσεις για προβλήματα όπως η κυκλοφοριακή συμφόρηση, η υπερβολική ζέστη ή το κρύο, για εκδηλώσεις που θα πραγματοποιηθούν κ.α.
- Να έχει πρόσβαση σε στοιχεία επικοινωνίας για οποιαδήποτε επιχείρηση ή δημόσιο οργανισμό επιθυμεί
- Να εκμεταλλεύεται την τρέχουσα τοποθεσία του προκειμένου να μάθει πληροφορίες και γεγονότα που συμβαίνουν γύρω του.

Αναφέροντας κάποια από τα σημαντικότερα οφέλη για τους χρήστες αντιλαμβανόμαστε γιατί οι εφαρμογές είναι σημαντικές και η χρήση τους έχει αυξηθεί τα τελευταία χρόνια όπως βλέπουμε και στο παρακάτω σχήμα:



Σχήμα 1 Downloads εφαρμογών 2008 έως 2013 (eMarketer)

1.3 ΚΑΙΝΟΤΟΜΕΣ ΥΠΗΡΕΣΙΕΣ

Η συνεχής εξέλιξη των «έξυπνων» τηλεφώνων δημιούργησε την ανάγκη για την ανάπτυξη νέων εφαρμογών οι οποίες θα καλύπτουν ανάγκες των χρηστών οι οποίες μέχρι τότε εξυπηρετούνταν με εντελώς διαφορετικό τρόπο. Αυτές οι νέες ανάγκες έχουν να κάνουν με διάφορους τομείς. Χαρακτηριστικά παραδείγματα είναι ο τομέας της ηλεκτρονικής υγείας (e-health), του ηλεκτρονικού εμπορίου (e-commerce), της ηλεκτρονικής διακυβέρνησης (e-government) καθώς και τομείς που έχουν να κάνουν με πληροφορίες οι οποίες ζητούνται από τους χρήστες σε πραγματικό χρόνο (real-time) προκειμένου είτε να γνωστοποιήσουν στοιχεία της θέσης τους είτε να εκμεταλλευτούν διαθέσιμες πληροφορίες σχετικά με την θέση στην οποία βρίσκονται.

Στην συνέχεια θα αναφερθούμε σε τέτοιους τομείς, βλέποντας πως μπορούν οι εφαρμογές είτε να εξυπηρετήσουν ή εξυπηρετούν αυτούς τους τομείς.

Ηλεκτρονική Υγεία (e-health)

Η ηλεκτρονική υγεία ορίζεται ως το σύνολο τεχνολογιών πληροφορικής, έγκυρων – έγκαιρων πληροφοριών και διαδικασιών που επιτρέπουν την αποτελεσματική προσφορά υπηρεσιών υγείας και ενσωματώνουν τις τεχνολογίες Ηλεκτρονικού Ιατρικού Φακέλου και τηλεϊατρικής[8].

Αντιλαμβανόμαστε επομένως πως η εξάπλωση των εφαρμογών κάνει τις τεχνολογίες κινητών πιο ελκυστικές για την ηλεκτρονική υγεία και την ιατρική πληροφορική. Αυτή την στιγμή υπάρχουν διάφορες εφαρμογές που στοχεύουν σε διαφορετικά προβλήματα υγείας και σε διαφορετικές ομάδες ανθρώπων. Οι εφαρμογές αυτές χρησιμοποιούν τους περιορισμένους πόρους των κινητών συσκευών σε σύγκριση με τα επιτραπέζια συστήματα. Παρόλα αυτά μεγεθύνοντας τα πλεονεκτήματα των κινητών συσκευών όπως η φορητότητα, η πρόσβαση στο Διαδίκτυο και η δυνατότητα εύρεσης τοποθεσίας έχουν κάνει αυτές τις εφαρμογές πολύ δημοφιλείς και μέρος της καθημερινής μας ζωής.

Οι επαγγελματίες υγείας χρησιμοποιούν συνεχώς εφαρμογές που λειτουργούν στα «έξυπνα» τηλέφωνα έχοντας την δυνατότητα να παρέχουν απομακρυσμένη παρακολούθηση και διαχείριση υγειονομικής περίθαλψης. Από την άλλη μεριά οι χρήστες χρησιμοποιούν εφαρμογές υγείας με στόχο να βοηθήσουν την ζωή και την υγεία τους. Για παράδειγμα, υπάρχουν εφαρμογές οι οποίες καταγράφουν την αρτηριακή πίεση (συστολική, διαστολική και ρυθμό παλμών) σύμφωνα με χρονοδιαγράμματα που προτείνει ο γιατρός στον ασθενή ή εφαρμογές οι οποίες δίνουν την δυνατότητα στους χρήστες την δυνατότητα να καταγράφουν τις αλλεργικές τους αντιδράσεις, όπως το επίπεδο εμφάνισης των συμπτωμάτων καθώς και τον τύπο αυτών (άσθμα, δάκρυα, κ.λπ.). Φυσικά, παρά τα οφέλη από αυτές τις εφαρμογές τα οποία είναι θετικά και ελπιδοφόρα, υπάρχουν πολλά ακόμη που πρέπει να γίνουν στο συγκεκριμένο τομέα για να καταστήσουν τέτοιες εφαρμογές ακόμη πιο επιτυχημένες και διαδεδομένες.

Θα πρέπει οι εφαρμογές ηλεκτρονικής υγείας για «έξυπνα» τηλέφωνα, να λειτουργούν πιο αποδοτικά είτε λειτουργώντας ως απλές εφαρμογές, εκτός δικτύου (offline) είτε λειτουργούν μέσω Wi-Fi, LAN, ή μέσω Διαδικτύου. Το κυριότερο πρόβλημα που καλούνται να αντιμετωπίσουν είναι η αξιοπιστία τους καθώς συχνά καταγράφουν ή παρέχουν κρίσιμα ιατρικά δεδομένα επικοινωνώντας με άλλες εφαρμογές και βάσεις δεδομένων προκειμένου να παρέχουν λύσεις.

Ηλεκτρονικό Εμπόριο (e-commerce)

Με την έλευση του Διαδικτύου άλλαξαν πολλές πτυχές της κοινωνίας, όπως ο τρόπος δουλειάς και διασκέδασης, ο τρόπος επικοινωνίας καθώς και ο τρόπος οργάνωσης ταξιδιών και πραγματοποίησης αγορών. Η μορφή αυτή επικοινωνίας προσέφερε νέους τρόπους για την επιχειρηματική δραστηριότητα χρησιμοποιώντας την τεχνολογία. Το ηλεκτρονικό εμπόριο (e-commerce) επομένως επέτρεψε στις εταιρίες να προωθήσουν και να πουλήσουν τα προϊόντα τους σε νέες αγορές, ξεπερνώντας τα γεωγραφικά σύνορα. Οι καταναλωτές έχουν πρόσβαση σε μια ευρύτερη αγορά προϊόντων όταν χρησιμοποιούν ασύρματες τεχνολογίες και το Διαδίκτυο. Οι κινητές συσκευές όπως τα «έξυπνα» τηλέφωνα με ευρεία πρόσβαση στο Διαδίκτυο με τεχνολογίες Wi-Fi ή δικτύωσης μέσω των εταιριών κινητών τηλεφώνων επιτρέπουν στις επιχειρήσεις να προσεγγίσουν καταναλωτές με διαφορετικούς τρόπους εξασφαλίζοντας έτσι την βαθιά διεξόδυσή τους στην αγορά.

Τα ταχύτερα πρότυπα ασύρματης δικτύωσης επιτρέπουν στα «έξυπνα» τηλέφωνα να χρησιμοποιούν περισσότερες εφαρμογές ηλεκτρονικού εμπορίου και να παρέχουν πρόσβαση στο κινητό εμπόριο (m-commerce) που αποτελεί μια μορφή ηλεκτρονικού εμπορίου. Το κινητό εμπόριο (m-commerce) έχει οριστεί ως «ειδικό κλάδος του ηλεκτρονικού εμπορίου, στον οποίο οι κινητές συσκευές και τα μέσα διασύνδεσης τους χρησιμοποιούνται για να αγοραστούν, πουληθούν και προωθηθούν προϊόντα, υπηρεσίες και πληροφορίες»[16].

Οι εφαρμογές αποτελούν σημαντικό παράγοντα στην εξέλιξη του ηλεκτρονικού εμπορίου και την κατά επέκταση του κινητού εμπορίου στους καταναλωτές. Βέβαια, πέρα από τις εφαρμογές υπάρχουν και άλλες πτυχές όπως η δυνατότητα πρόσβασης μιας κινητής συσκευής στο Διαδίκτυο, η επεξεργαστική ισχύ της, το μέγεθος της οθόνης οι οποίοι είναι μερικοί μόνο παράγοντες που αποφασίζουν το επίπεδο χρήσης του κινητού εμπορίου και κατά επέκταση και το επίπεδο ανάπτυξης του.

Ηλεκτρονική Διακυβέρνηση (e-government)

Η χρήση των Τεχνολογιών Πληροφορίας και Επικοινωνιών (ΤΠΕ) στο δημόσιο τομέα με σκοπό τη βελτίωση των εσωτερικών διαδικασιών του δημόσιου τομέα, τη βελτίωση της επικοινωνίας και των παρεχόμενων υπηρεσιών προς τους πολίτες και τη βελτίωση των σχέσεων μεταξύ οργανισμών του δημοσίου με άλλους οργανισμούς ονομάζεται Ηλεκτρονική Διακυβέρνηση (e-government)[20].

Στο εξωτερικό πολύ πιο μπροστά και στην Ελλάδα τα τελευταία χρόνια προκειμένου να εξαλειφθούν τα φαινόμενα της γραφειοκρατίας έχουν δημιουργηθεί διαδικτυακές υπηρεσίες οι οποίες δίνουν την δυνατότητα στον πολίτη να εξυπηρετηθεί από υπουργεία και άλλες δημόσιες υπηρεσίες, την κυβέρνηση να συναλλάσσεται με επιχειρήσεις (ηλεκτρονικές προμήθειες, ηλεκτρονικές δημοπρασίες), με άλλες κυβερνήσεις καθώς και με υπαλλήλους.

Παρόλα αυτά η ηλεκτρονική διακυβέρνηση δεν έχει χρησιμοποιηθεί ευρέως τουλάχιστον στην Ελλάδα στα «έξυπνα» τηλέφωνα και ο κυριότερος λόγος είναι ότι χρειάζεται μεγάλη προσοχή με τα προσωπικά δεδομένα και την ασφάλεια μετάδοσης των πληροφοριών. Αντίθετα στις Ηνωμένες Πολιτείες έχουν γίνει αρκετά βήματα προς αυτή την κατεύθυνση καθώς και σε άλλα μέρη του πλανήτη.

1.4 ΣΤΟΧΟΣ ΕΡΓΑΣΙΑΣ

Αντικείμενο της εργασίας είναι η ανάπτυξη ενός συστήματος για «έξυπνα» τηλέφωνα που θα παρουσιάζει δημόσιες πληροφορίες μόλυνσης περιβάλλοντος για συγκεκριμένη γεωγραφική θέση χρησιμοποιώντας ένα μεθοδολογικό πλαίσιο ανάπτυξης που παρέχεται από την Google.

Μέσα όμως από την υλοποίηση του αντικειμένου της εργασίας στόχος είναι:

- Να δημιουργηθεί μια υπηρεσία η οποία θα πληροί συγκεκριμένα χαρακτηριστικά που θα την κάνουν καινοτόμα
- Να γίνει αντιληπτός ο ρόλος και οι δυνατότητες των «έξυπνων» τηλεφώνων
- Να αντληθεί τεχνογνωσία από υφιστάμενες λύσεις προκειμένου να αναπτυχθεί η υπηρεσία μας πιο ολοκληρωμένα
- Να χρησιμοποιηθεί η γλώσσα μοντελοποίησης εφαρμογών UML (Unified Model Language) προκειμένου να πραγματοποιηθεί το στάδιο ανάλυσης της υπηρεσίας μας
- Να χρησιμοποιηθεί ένα καινούριο σχετικά περιβάλλον ανάπτυξης εφαρμογών που παρέχει το Android μαθαίνοντας τον τρόπο λειτουργίας του και χρησιμοποίησης εργαλείων που παρέχει
- Να γίνει εξαγωγή συμπερασμάτων και παρουσίαση προτάσεων σχετικά με την υπηρεσία που δημιουργήσαμε καθώς και η έκθεση άλλων καινοτόμων προτάσεων με παραπλήσιο αντικείμενο

1.5 Η ΥΠΗΡΕΣΙΑ POLLUTION TRACKER

Η υπηρεσία PollutionTracker δημιουργήθηκε στα πλαίσια της πτυχιακής εργασίας με τίτλο "Ανάπτυξη εφαρμογής για έξυπνο κινητό παρουσίασης σε πραγματικό χρόνο πληροφορίες μόλυνσης περιβάλλοντος" για το Τμήμα Πληροφορικής του Αλεξάνδρειου Τεχνολογικού Ιδρύματος Θεσσαλονίκης.

Η υπηρεσία PollutionTracker δημιουργήθηκε με σκοπό να παρέχει στον χρήστη την δυνατότητα να ενημερώνεται σε πραγματικό χρόνο για την μόλυνση του περιβάλλοντος που επικρατεί στην γεωγραφική θέση την οποία βρίσκεται. Η υπηρεσία δοκιμάζεται στην περιφέρεια της δυτικής Μακεδονίας, μια περιοχή με τεράστιο πρόβλημα στην ποιότητα της ατμόσφαιρας καθώς υπάρχουν πάρα πολλά εργοστάσια τα οποία μολύνουν το περιβάλλον.

Η υπηρεσία χρησιμοποιεί πληροφορίες που προέρχονται από σταθμούς που βρίσκονται σε κεντρικές περιοχές της Δυτική Μακεδονίας και μετρούν την ποσότητα μόλυνσης της ατμόσφαιρας και τις παρέχει στον χρήστη. Ο χρήστης επιλέγοντας την τοποθεσία που επιθυμεί, προβάλλει τις πληροφορίες σχετικά με ωριαίες μετρήσεις, μέσες τιμές καθώς και άλλα είδη μετρήσεων της μόλυνσης που επικρατεί στο «έξυπνο» τηλέφωνο του.

Η καινοτομία η οποία φέρει ως υπηρεσία είναι ότι με την χρήση των έξυπνων τηλεφώνων τα οποία αναπτύσσονται ραγδαία πλέον ο χρήστης εκμεταλλευόμενος την φορητότητα και κινητικότητα της συσκευής του μπορεί να προστατεύεται από πιθανά αναπνευστικά προβλήματα και να μην χρειάζεται προκειμένου να ενημερωθεί να βρίσκεται μπροστά σε κάποιο υπολογιστή ή σε κάποιο άλλο μέσο ενημέρωσης.

1.6 ΔΟΜΗ ΕΡΓΑΣΙΑΣ

Στην ενότητα αυτή κάνουμε μια σύντομη επισκόπηση της εργασίας που ακολουθεί. Στο κεφάλαιο 2 αναφερόμαστε στην έννοια της καινοτομίας, πως αυτή ορίζεται, η καινοτομία στον χώρο της τεχνολογίας καθώς και στα χαρακτηριστικά ενός καινοτόμου προϊόντος. Επιπλέον παρουσιάζονται τα οφέλη και οι προϋποθέσεις για την δημιουργία προσωποποιημένου περιεχομένου. Στην συνέχεια (κεφάλαιο 3) στο πλαίσιο της εξέτασης υφιστάμενων λύσεων παρατίθενται δύο μελέτες περίπτωσης στα «έξυπνα» τηλέφωνα για την αντιμετώπιση προβλημάτων μόλυνσης καθώς και μια συγκριτική αξιολόγηση των υπηρεσιών αυτών. Επίσης γίνεται αναφορά στα χαρακτηριστικά της υπηρεσίας PollutionTracker.

Στο επόμενο κεφάλαιο (κεφάλαιο 4) προχωράμε σε ανάλυση της υπηρεσίας μας παρουσιάζοντας διαγράμματα περίπτωσης χρήσης, ακολουθίας, κλάσεων καθώς άλλων χρήσιμων εννοιών που αφορούν την μοντελοποίηση της υπηρεσίας μας με σκοπό την καλύτερη κατανόηση της. Στο κεφάλαιο 5, γίνεται παρουσίαση του τεχνολογικού περιβάλλοντος ανάπτυξης και πιο συγκεκριμένα των εργαλείων που μας παρέχει η πλατφόρμα Android, το περιβάλλον που επιλέξαμε για την ανάπτυξη της υπηρεσίας μας και σε ποιες συσκευές είναι λειτουργική. Η περιγραφή της υπηρεσίας PollutionTracker, δηλαδή η παρουσίαση της αρχιτεκτονικής της καθώς και οι λειτουργίες που περιλαμβάνει παρουσιάζονται στο κεφάλαιο 6.

Κλείνοντας την εργασία αναφερόμαστε την προστιθέμενη αξία της υπηρεσίας μας (κεφάλαιο 7), σε συμπεράσματα τα οποία προέκυψαν από την ενασχόληση μας με την ανάπτυξη της υπηρεσίας μας καθώς και στον κώδικα με την βοήθεια επεξηγηματικών σχολίων ο οποίος συντάχθηκε για την υλοποίηση της υπηρεσίας μας.

ΚΕΦΑΛΑΙΟ 2

ΚΑΙΝΟΤΟΜΙΑ – ΕΞΥΠΝΑ ΚΙΝΗΤΑ

ΕΙΣΑΓΩΓΗ

Το οικονομικό περιβάλλον έχει αλλάξει δραματικά τα τελευταία χρόνια. Οι επιχειρήσεις και οι οικονομίες των χωρών δεν μπορούν να βασίζονται στα παραδοσιακά τους προϊόντα και πρακτικές του παρελθόντος. Ο κύκλος ζωής των προϊόντων έχει μειωθεί και η αγορά επιζητά συνέχεια νέα προϊόντα και νέες υπηρεσίες. Αυτό οφείλεται τόσο στην παγκοσμιοποίηση όσο και στην χρήση των τεχνολογιών με αποτέλεσμα να διαμορφώνεται μια εντελώς νέα κατάσταση.

Οι οικονομίες των χωρών έχουν επηρεασθεί από την βελτίωση της παραγωγικότητας που αποδίδεται στην χρήση των Τεχνολογιών Πληροφορικής και Επικοινωνιών (ΤΠΕ). Οι χώρες που έχουν κάνει άνοιγμα προς αυτή την κατεύθυνση, δημιουργώντας αναγκαίες υποδομές δηλαδή άνοιγμα των τηλεπικοινωνιών τους αγορών με την συνακόλουθη μείωση του κόστους σε επενδύσεις και δίκτυα μεταφοράς δεδομένων υψηλών ταχυτήτων, έχουν παρουσιάσει βελτίωση της ανταγωνιστικότητας τους σε σύγκριση με χώρες, όπως για παράδειγμα η Ελλάδα, οι οποίες δεν έχουν επενδύσει προς την χρήση των ΤΠΕ και δεν υπάρχει ανάπτυξη στην ανταγωνιστικότητά τους.

Χαρακτηριστικό είναι ότι την δεκαετία του 1990 όπου η ανταγωνιστικότητα της αμερικανικής οικονομίας αναπτύχθηκε, σημαντικός παράγοντας στην ανάπτυξη αυτή έπαιξαν οι υψηλές επενδύσεις σε ΤΠΕ συγκριτικά με την ΕΕ και Ιαπωνία. Το 1999 οι επενδύσεις αυτές έφθασαν στο 7,3% του ΑΕΠ, όταν το ίδιο ποσοστό για την ΕΕ ήταν 5,8% [12].

Σύμφωνα με μελέτες [12] όμως οι επενδύσεις σε ΤΠΕ δεν αποτελούν την λύση για όλα. Οι αναπτυγμένες οικονομίες στηρίζονται και σε άλλες έννοιες όπως η επιχειρηματικότητα, το ανθρώπινο δυναμικό και η καινοτομία.

Στην σημερινή οικονομία της γνώσης είναι αδιαμφισβήτητη η σημασία της καινοτομίας. Η έννοια της καινοτομίας αποτελεί βασική παράμετρο για την ανάπτυξη μιας επιχείρησης ή μιας χώρας και γι αυτό το λόγο βρίσκεται σε θέση υψηλής προτεραιότητας για την πολιτική όλων σχεδόν των χωρών. Η διαφορά στην ανταγωνιστικότητα και στα εισοδήματα που έχει μια χώρα μπορεί σε ένα βαθμό να εξηγηθεί από το διαφορετικό επίπεδο καινοτομικής δραστηριότητας και ανάπτυξης από χώρα σε χώρα. Οι λόγοι για τους οποίους φτάνουμε σε ένα τέτοιο συμπέρασμα είναι ότι τα νέα προϊόντα βοηθούν στη διατήρηση των ήδη κεκτημένων μεριδίων αγοράς για μια επιχείρηση και επομένως στην κερδοφορία της. Είναι σημαντικό στην σημερινή εποχή που ο κύκλος των προϊόντων έχει μειωθεί αισθητά και οι ανάγκες των ανθρώπων είναι ευμετάβλητες μια επιχείρηση να μπορεί να δημιουργεί νέα καινοτόμα προϊόντα.

Έτσι εξηγείται και γιατί η καινοτομία έχει τεθεί ως σημαντική πολιτική για όλες τις χώρες. Η Ευρωπαϊκή Κοινότητα έχει θεσπίσει σχέδιο δράσης για την καινοτομία στην Ευρώπη προκειμένου η κάθε χώρα και συνολικά η Ευρώπη να αυξήσει τα επίπεδα καινοτομίας της.

2.1 ΤΙ ΕΙΝΑΙ ΚΑΙΝΟΤΟΜΙΑ

Ως καινοτομία ορίζουμε τη χρήση νέας γνώσης προκειμένου να προσφερθεί (δηλαδή να σχεδιαστεί και να εμπορευματοποιηθεί) ένα νέο προϊόν ή μια νέα υπηρεσία που θέλουν οι πελάτες. Η καινοτομία μπορεί να αφορά ένα νέο προϊόν ή μια νέα υπηρεσία που θέλουν οι πελάτες[3]. Η καινοτομία μπορεί να αφορά ένα νέο προϊόν ή μια νέα υπηρεσία, τον τρόπο ή την τεχνολογία παραγωγής ενός προϊόντος ή μιας υπηρεσίας, καθώς επίσης μπορεί να αναφέρεται σε διοικητική διαδικασία ή οργανωτική δομή. Μια καινοτομία μπορεί να είναι ριζική ή σταδιακή, καινοτομία προϊόντος ή διεργασίας και τεχνική ή διαχειριστική καινοτομία.



Σχήμα 2 Ορισμός Καινοτομίας - Ολοστικό Μοντέλο [2]

Καινοτομία Προϊόντος ή Καινοτομία Διεργασίας (Product vs. Process innovation). Αυτή η διάκριση αναφέρεται στις περιοχές και στις δραστηριότητες τις οποίες επηρεάζει μια καινοτομία. Ενώ οι καινοτομίες διεργασιών ορίζονται ως «εργαλεία», μηχανισμοί και γνώση σχετικά με την παραγωγική διαδικασία, οι καινοτομίες προϊόντος είναι προϊόντα ή υπηρεσίες που εισάγονται για το όφελος των πελατών[4].

Ριζική ή Σταδιακή Καινοτομία (Radical vs. Incremental innovation). Οι ριζικές καινοτομίες προκαλούν θεμελιώδεις αλλαγές στις δραστηριότητες ενός οργανισμού και αποτελούν ένα σαφές σημείο τομής σε σχέση με τις μέχρι τότε χρησιμοποιούμενες πρακτικές.

Τέτοιου είδους ριζικές καινοτομίες τείνουν να αλλάξουν πλήρως τις υπάρχουσες ικανότητες της επιχείρησης απαξιώνοντας τις, και μπορεί να έχουν ως αποτέλεσμα την αύξηση της αβεβαιότητας και τη ριζική αλλαγή της φυσιογνωμίας μιας επιχείρησης ή και ενός ολόκληρου κλάδου. Οι σταδιακές καινοτομίες, αντίθετα, έχουν περισσότερο το χαρακτήρα απλών βελτιώσεων και αντιπροσωπεύουν μικρές μόνο αλλαγές στις τρέχουσες πρακτικές της επιχείρησης[4].

Τεχνική ή Διαχειριστική Καινοτομία (Technical vs. Administrative Innovation). Αυτή η διάκριση είναι σημαντική διότι αντανακλά τη γενικότερη διάκριση μεταξύ της *εσωτερικής δομής* της επιχείρησης και της *τεχνολογίας* που χρησιμοποιεί. Οι τεχνικές καινοτομίες αναφέρονται σε προϊόντα, διεργασίες και τεχνολογίες που χρησιμοποιούνται προκειμένου να παραχθούν αυτά τα προϊόντα. Σχετίζονται άμεσα στις βασικές δραστηριότητες του οργανισμού. Οι διαχειριστικές καινοτομίες αναφέρονται στην οργανωτική δομή, τις διοικητικές διεργασίες και στους ανθρώπινους πόρους του οργανισμού. Σχετίζονται *έμμεσα* με το βασικό παραγωγικό κύκλο εργασιών του οργανισμού και σχετίζονται *άμεσα* με τον τρόπο διοίκησής του[4].

2.2 ΤΕΧΝΟΛΟΓΙΚΗ ΚΑΙΝΟΤΟΜΙΑ

Η τεχνολογική καινοτομία ορίζεται ως η μεταβολή σε ένα οργανισμό (επιχείρηση, οργανισμό παροχής υπηρεσιών):

- των προϊόντων – υπηρεσιών με νέες που έχουν καλύτερη ποιότητα
- των διαδικασιών παραγωγής με χρήση τεχνολογιών πληροφορικής, νέες μορφές ενέργειας και αυτοματισμούς
- της οργάνωσης του οργανισμού βελτιστοποιώντας έτσι την παραγωγική διαδικασία

Επιπρόσθετα, η τεχνολογική καινοτομία πρέπει να βασίζεται στα αποτελέσματα νέων τεχνολογικών εξελίξεων, νέων συνδυασμών υπάρχουσών τεχνολογιών ή στην χρησιμοποίηση άλλου είδους γνώσεων που αποκτήθηκαν από την επιχείρηση.

Στην συνέχεια παρουσιάζονται ενδεικτικά κάποια παραδείγματα που μπορεί να αφορά η τεχνολογική καινοτομία.

Εμπόριο

Καινοτομία προϊόντος ή διαδικασίας

- Εισαγωγή για πρώτη φορά υπηρεσιών διαδικτύου, όπως τραπεζικές συναλλαγές, συστήματα πληρωμής λογαριασμών, ηλεκτρονικές αγορές και έκδοση εισιτηρίων ταξιδιού και θεάτρου, ιστότοποι κοινωνικής δικτύωσης κ.ά.

- Απευθείας πώληση στον πελάτη πραγματοποιώντας ηλεκτρονικές ανταλλαγές προϊόντων
- Νέα είδη υπηρεσιών πιστοποίησης
- Εισαγωγή οικολογικών προϊόντων στη σειρά των αγαθών
- Δημιουργία ηλεκτρονικών καταλόγων σε ψηφιακά μέσα αποθήκευσης
- Εγκατάσταση θέρμανσης φυσικού αερίου σε υπαίθριο εστιατόριο και σε εξωτερικά μπαρ, σε βεράντες/ταράτσες ή εγκατάσταση οθόνων με βίντεο βάσει ζήτησης στο πίσω μέρος καθισμάτων αεροπλάνου, λεωφορείου ή τρένου.

Βιομηχανία / Παραγωγή

Καινοτομία προϊόντος ή διαδικασίας

- Φάρμακα βιολογικής βάσης
- Προϊόντα βιοτεχνολογίας
- Αξιοποίηση απορριμμάτων / αποβλήτων
- Συστήματα μέτρησης και ελέγχου αποθεματικών μιας αποθήκης με προϊόντα
- Αντικατάσταση υπαρχόντων υλικών με υλικά που διαθέτουν βελτιωμένα χαρακτηριστικά (υφάσματα που αναπνέουν, ελαφριά αλλά ισχυρά συνθετικά υλικά, πλαστικά φιλικά προς το περιβάλλον κ.ά.)
- Εισαγωγή νέων ή βελτιωμένων εξαρτημάτων στις υπάρχουσες γραμμές παραγωγής (όπως, κάμερες σε κινητά τηλέφωνα κ.ά.).

Μια τεχνολογική καινοτομία της εποχή μας αποτελούν τα επονομαζόμενα «έξυπνα» τηλέφωνα. Τα περισσότερα κινητά τηλέφωνα στις μέρες μας αντιμετωπίζονται ως «έξυπνα» τηλέφωνα δεδομένου ότι προσφέρουν προηγμένες δυνατότητες σε σύγκριση με ένα σύγχρονο τηλέφωνο. Ένα «έξυπνο» τηλέφωνο έχει δυνατότητες συγκρίσιμες με ενός υπολογιστή. Αυτό τον ισχυρισμό έρχεται να επιβεβαιώσει η σχετικά πρόσφατη απελευθέρωση των πολυπύρηνων επεξεργαστών στα «έξυπνα» τηλέφωνα.

Πέρα από τις βασικές δυνατότητες όπως η πραγματοποίηση φωνητικών κλήσεων, κλήσεων βίντεο καθώς και αποστολή SMS και MMS τα «έξυπνα» τηλέφωνα παρέχουν μια σειρά από λειτουργίες όπως, διαχείριση μέσω χρονοδιαγράμματος του προσωπικού χρόνου του χρήστη, πρόσβαση σε περιεχόμενο του Διαδικτύου, επεξεργασία εγγράφων, παροχή πληροφοριών με βάση την θέση του χρήστη και άλλες ενδιαφέρουσες λειτουργίες. Γι αυτό το λόγο και η είσοδός τους στην αγορά χαρακτηρίστηκε ως καινοτομία καθώς οι λειτουργίες τους ξεπερνούσαν τα συνηθισμένα τηλέφωνα και έδιναν την ευκαιρία να αναπτυχθούν νέοι κλάδοι γύρω από αυτή την καινοτομία όπως ο κλάδος των εφαρμογών για κινητά τηλέφωνα και κατά επέκταση καινοτόμες υπηρεσίες.

2.3 ΚΑΙΝΟΤΟΜΙΑ ΠΡΟΪΟΝΤΟΣ

Λέγοντας καινοτομία προϊόντος εννοούμε την εισαγωγή στην αγορά ενός νέου αγαθού ή υπηρεσίας, το οποίο θεωρείται νέο ή σημαντικά βελτιωμένο προϊόν ως προς τα βασικά του χαρακτηριστικά και τις προοριζόμενες χρήσεις του[3].

Περιλαμβάνονται σημαντικές βελτιώσεις στις τεχνικές προδιαγραφές, τα συστατικά και τα υλικά, το ενσωματωμένο λογισμικό, τη φιλικότητα προς τον χρήστη ή και άλλα λειτουργικά χαρακτηριστικά του προϊόντος.

Οι καινοτομίες προϊόντος μπορεί να βασίζονται σε νέα γνώση ή τεχνολογία, ή να βασίζονται σε νέες χρήσεις ή συνδυασμούς υπάρχουσας γνώσης ή τεχνολογίας. Ο όρος «προϊόν» αναφέρεται σε νέα αγαθά και υπηρεσίες. Οι καινοτομίες προϊόντος περιλαμβάνουν εξίσου την εισαγωγή νέων αγαθών και υπηρεσιών καθώς και τις σημαντικές βελτιώσεις στα λειτουργικά χαρακτηριστικά αγαθών και υπηρεσιών.

2.3.1 ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΕΝΟΣ ΚΑΙΝΟΤΟΜΟΥ ΠΡΟΪΟΝΤΟΣ

Ένα προϊόν ή υπηρεσία προκειμένου να χαρακτηριστεί καινοτόμο θα πρέπει να επιτυγχάνει τους παρακάτω στόχους – αποτελέσματα[12]:

- Να αντικαθιστά άλλα προϊόντα σταδιακά
- Αύξηση του εύρους των παρεχόμενων υπηρεσιών (νέα λειτουργικά και τεχνικά χαρακτηριστικά)
- Ανάπτυξη φιλικών προς το περιβάλλον προϊόντων
- Να αυξάνει ή να διατηρεί το μερίδιο της αγοράς της επιχείρησης που το δημιουργεί
- Να αποτελεί το «εισιτήριο» για την είσοδο σε νέες αγορές
- Βελτίωση της ποιότητας των υπηρεσιών
- Μείωση της κατανάλωσης υλικών και ενέργειας
- Κάλυψη υψηλών τεχνικών προδιαγραφών
- Μείωση των περιβαλλοντικών επιπτώσεων που προκύπτει από τη βελτίωση της υγείας και την παροχή ασφάλειας

Τα ακόλουθα χαρακτηριστικά εξαιρούνται από τις καινοτομίες προϊόντος[12]:

- Μικρές αλλαγές ή βελτιώσεις
- Αναβαθμίσεις ρουτίνας
- Αλλαγές προϊόντος καθαρά αισθητικού χαρακτήρα
- Εποχικές και περιοδικές αλλαγές (πχ η τάση της μόδας στον ρουχισμό)
- Η μεταπώληση των νέων αγαθών και υπηρεσιών που αγοράζονται από άλλες επιχειρήσεις

2.4 ΠΡΟΣΩΠΟΠΟΙΗΜΕΝΟ ΠΕΡΙΕΧΟΜΕΝΟ

Σημαντικό για ένα καινοτόμο προϊόν είναι να παρέχει περιεχόμενο το οποίο θα είναι προσωποποιημένο στις ανάγκες του πελάτη και θα προσφέρει μια μοναδική εμπειρία σε αυτούς. Επομένως το προϊόν πρέπει να χαρακτηρίζεται από προσωποποιημένο

περιεχόμενο. Σύμφωνα με τον Muvenna [9] το προσωποποιημένο περιεχόμενο έχει ως στόχο να παρέχει στους πελάτες αυτό που θέλουν χωρίς να χρειάζεται να ερωτηθούν για τις ανάγκες τους. Παρόλα αυτά, αυτό δεν είναι απολύτως σωστό καθώς υπάρχουν διαδικασίες τις οποίες ο δημιουργός ενός προϊόντος μπορεί να ακολουθήσει προκειμένου να δημιουργήσει προσωποποιημένο περιεχόμενο. Ο δημιουργός μπορεί να είναι μια επιχείρηση, ένας προγραμματιστής για λογαριασμό μιας επιχείρησης ή οποιοσδήποτε άλλος οργανισμός που παράγει ένα τύπο προϊόντος.

Σύμφωνα με τους Adomavicius και Tuzhilin [5] υπάρχουν τρία στάδια τα οποία υποστηρίζουν την διαμόρφωση προσωποποιημένου περιεχομένου:

- Κατανόηση πελάτη, όπου συλλέγονται πληροφορίες και γίνεται το χτίσιμο του προφίλ των πελατών
- Παροχή προσωποποίησης, όπου συνδυάζονται οι πληροφορίες και παρουσιάζονται στους πελάτες
- Μέτρηση αποτελεσματικότητας, όπου καταμετράται η αποτελεσματικότητα των πληροφοριών που παρέχονται και γίνονται προσαρμογές στην στρατηγική της επιχείρησης.

Η συλλογή των πληροφοριών που αναφέραμε στο πρώτο στάδιο μπορεί να γίνεται είτε με άμεσο είτε με έμμεσο τρόπο. Αφού γίνει περισυλλογή των στοιχείων θα πρέπει να γίνει επεξεργασία τους προκειμένου να χτιστεί το προφίλ των πελατών [5].

Άμεση συλλογή στοιχείων:

- Αξιολόγηση υπαρχόντων προϊόντων από τον χρήστη βάσει των εμπειριών του. Αυτό οδηγεί στην συλλογή πληροφοριών σχετικά με τις προτιμήσεις του.
- Εργαλεία τα οποία δίνουν την δυνατότητα στον πελάτη να ορίσει σχετικά ενδιαφέροντα.
- Συμπλήρωση ερωτηματολογίων εκ μέρους του πελάτη προκειμένου να καθοριστούν οι προτιμήσεις του [10].

Έμμεση συλλογή στοιχείων:

Στην έμμεση συλλογή στοιχείων κυρίαρχο ρόλο διαδραματίζει η τεχνολογία. Υπάρχουν μέθοδοι που συλλέγουν στοιχεία για τους πελάτες χωρίς αυτοί να ενοχλούνται. Τέτοιοι μέθοδοι είναι:

- Τα Web logs, τα οποία είναι αρχεία τα οποία καταγράφουν την IP του πελάτη και του χρόνου διάδρασης αυτού
- Τα App logs, τα οποία επιστρέφουν τις πληροφορίες που ζητήθηκαν από back end βάσεις δεδομένων
- Τα Cookies, τα οποία εγκαθίστανται στον σκληρό του χρήστη ενός υπολογιστή και σε κάθε επίσκεψη σε έναν ιστότοπο αντιλαμβάνεται ποιος χρήστης είναι

- Η ανίχνευση συναλλαγών η οποία γίνεται μέσω στοιχείων εγγραφής ή τα cookies και γίνεται γνωστό τα προϊόντα που αγοράστηκαν, οι τιμές τους καθώς και ο χρόνος που ξοδεύτηκε για την αγορά του καθενός από αυτού.

2.4.1 ΟΦΕΛΗ ΠΡΟΣΩΠΟΠΟΙΗΣΗΣ ΠΕΡΙΕΧΟΜΕΝΟΥ

Η προσωποποίηση του περιεχομένου μπορεί να αναφέραμε ότι γίνεται προκειμένου να ικανοποιήσει τις ανάγκες του πελάτη αλλά τα οφέλη αυτής της τάσης είναι και για τις επιχειρήσεις.

Οι επιχειρήσεις προσωποποιώντας το περιεχόμενο ενός προϊόντος δημιουργούν καλύτερη σχέση καθώς και συναισθηματικούς δεσμούς με τους πελάτες τους. Μπορούν να διατηρήσουν επαφή με μέρος αυτών προκειμένου να εντοπίζονται λάθη αλλά και καινούριες ανάγκες αυτών. Επίσης η προσωποποίηση επηρεάζει την ικανοποίηση των πελατών καθώς και την πιστότητα τους (e-loyalty). Τέλος βοηθάει στην προσέλκυση νέων πελατών, βελτιώνει την στρατηγική θέση της επιχείρησης και τα οικονομικά της αποτελέσματα [6].

Από την άλλη μεριά τα οφέλη των πελατών από την διαδικασία της προσωποποίησης έχουν να κάνουν αρχικά με το αυτονόητο, δηλαδή ότι υπάρχει μεγαλύτερο ταίριασμα στο περιεχόμενο με το κατάλληλο προϊόν προκειμένου να καλυφθούν οι ανάγκες του χρήστη [18]. Πέρα όμως από αυτό ο πελάτης εξοικονομεί χρόνο καθώς όταν βρίσκει ένα προϊόν εξατομικευμένο δεν χρειάζεται να αναζητήσει κάποιο άλλο. Επίσης ο πελάτης εξοικονομεί χρήματα και μετριάζεται το ρίσκο που παίρνει μιας και δεν θα ξοδέψει χρήματα για ένα προϊόν που δεν ανταποκρίνεται στις απαιτήσεις του [13]. Τέλος, βελτιώνεται η επικοινωνία με την επιχείρηση και ο πελάτης μπορεί να νιώθει πιο οικεία με αυτήν.

2.4.2 ΠΡΟΥΠΟΘΕΣΕΙΣ ΠΡΟΣΩΠΟΠΟΙΗΣΗΣ

Η εφαρμογή της προσωποποίησης πρέπει να πληροί κάποιες προϋποθέσεις καθώς δεν αποτελεί πάντα ενδεδειγμένη λύση. Πρέπει η επιχείρηση που θα δημιουργήσει το προϊόν να έχει γνώση του κλάδου που δραστηριοποιείται, το είδος των προϊόντων του κλάδου και να λαμβάνει υπόψη άλλους παράγοντες όπως για παράδειγμα οι διαφοροποιήσεις των πελατών (κατηγορίες). Επίσης, η επιχείρηση χρειάζεται να επενδύσει στην εκπαίδευση του ανθρώπινου προσωπικού της καθώς και σε τεχνολογία προκειμένου να υποστηριχθεί η εφαρμογή τέτοιων συστημάτων.

Επιπλέον θα πρέπει η επιχείρηση να καταγράφει και να παρακολουθεί συνεχώς τις προτιμήσεις των πελατών καθώς στην διάρκεια του χρόνου αυτές μεταβάλλονται με αποτέλεσμα διαφοροποιήσεις να μην γίνονται αντιληπτές από την επιχείρηση και να μην υφίσταται πλέον κάλυψη των αναγκών του πελάτη. Τέλος, ένα σημαντικό θέμα ειδικά για περιεχόμενο το οποίο αφορά το Διαδίκτυο είναι η διασφάλιση του απορρήτου των πελατών καθώς όταν η επιχείρηση συγκεντρώνει πληροφορίες με έναν από τους τρόπους τους οποίους αναφέραμε προκειμένου να δημιουργήσει προσωποποιημένο

περιεχόμενο οι πελάτες θέλουν να διατηρούν την ανωνυμία τους ακόμα και αν συμφωνούν με την όλη διαδικασία. Αυτό ονομάζεται παράδοξο της προσωποποίησης [11].

ΕΠΙΛΟΓΟΣ

Στο κεφάλαιο αυτό παρουσιάσαμε το πώς ορίζεται η καινοτομία καθώς και τις κατηγορίες στις οποίες χωρίζεται. Αναφερθήκαμε στην τεχνολογική καινοτομία, και παραθέσαμε ενδεικτικά παραδείγματα αυτής. Στην συνέχεια εξετάσαμε την έννοια καινοτομία προϊόντος καθώς και τα χαρακτηριστικά που πρέπει να πληροί ένα προϊόν για να χαρακτηριστεί καινοτόμο. Τέλος, μιλήσαμε για το προσωποποιημένο περιεχόμενο ενός προϊόντος.

Στο επόμενο κεφάλαιο θα μελετήσουμε που διαφοροποιείται η υπηρεσία την οποία έχουμε αναπτύξει σε σχέση με υφιστάμενες λύσεις που υπάρχουν. Θα παρουσιάσουμε κάποιες από αυτές και τέλος θα αξιολογήσουμε εάν η υπηρεσία - εφαρμογή την οποία αναπτύξαμε πληροί τα κριτήρια που χαρακτηρίζουν μια εφαρμογή καινοτόμα.

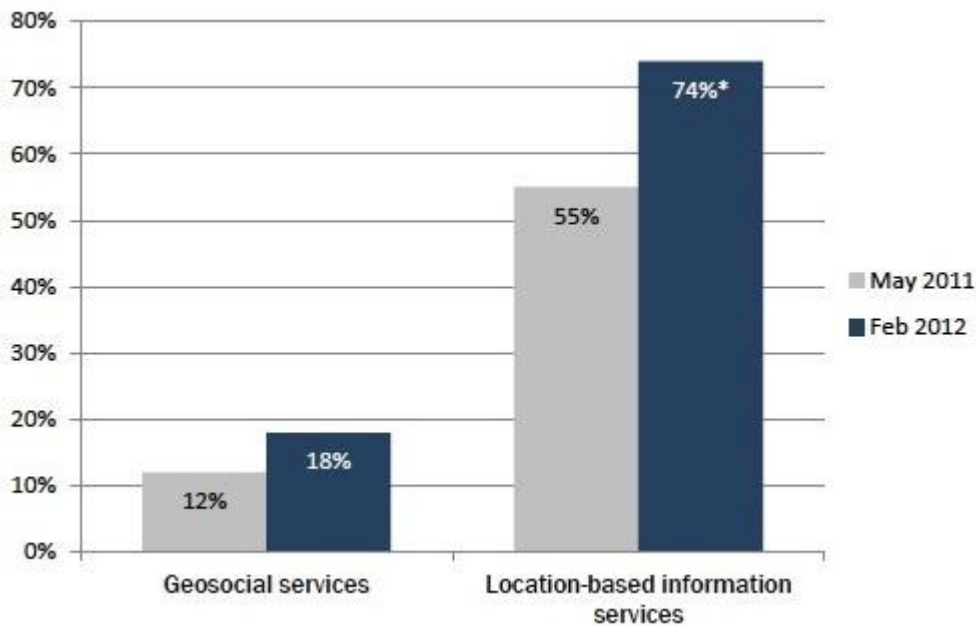
ΚΕΦΑΛΑΙΟ 3

ΥΦΙΣΤΑΜΕΝΕΣ ΛΥΣΕΙΣ

ΕΙΣΑΓΩΓΗ

Σχεδόν τα τρία τέταρτα των ιδιοκτητών «έξυπνων» τηλεφώνων χρησιμοποιούν την συσκευή τους για να πάρουν πληροφορίες με βάση την γεωγραφική τοποθεσία τους σε πραγματικό χρόνο. Πιο συγκεκριμένα, το 74% των χρηστών παίρνουν τέτοιες πληροφορίες όπως μετρήθηκε τον Φεβρουάριο του έτους 2012. Το ποσοστό αυτό είναι αυξημένο κατά 19% σε σύγκριση με τον Μάιο του 2011. Αυτή η αύξηση συμπίπτει με την αύξηση στον αριθμό των χρηστών «έξυπνων» τηλεφώνων (από 35% των ενήλικων χρηστών το 2011 σε 46% το 2012) κάτι το οποίο σημαίνει ότι το συνολικό ποσοστό των χρηστών που παίρνει πληροφορίες με βάση την τοποθεσία έχει διπλασιαστεί κατά την διάρκεια αυτής της χρονικής περιόδου (από 23% το Μάιο του 2011 σε 41% το Φεβρουάριο του 2012).

Εν τω μεταξύ, πολλοί χρήστες χρησιμοποιούν γεω-κοινωνικές υπηρεσίες κοινοποιώντας την θέση την οποία βρίσκονται. Στην εικόνα παρακάτω βλέπουμε ότι για τον Μάιο του 2011 των 18% των χρηστών χρησιμοποιούν γεω-κοινωνικές πληροφορίες για τα τηλέφωνα τους από το 12% που ήταν το 2011 [19].



Σχήμα 3 Η χρήση πληροφοριών με βάση την τοποθεσία - γεω - κοινωνικές πληροφορίες [19]

Στη παρακάτω εικόνα παρουσιάζεται το ποσοστό, όπως αυτό μετρήθηκε το Φεβρουάριο του 2012, των χρηστών «έξυπνων» τηλεφώνων που χρησιμοποιούν γεω-κοινωνικές υπηρεσίες σε σύγκριση με αυτούς που επιζητούν πληροφορίες / οδηγίες με βάση την τοποθεσία τους.

	All adults	All cell owners	Smartphone owners
Get location-based directions/information	41%	46%	74%
Use a geosocial or "check-in" service	10%	11%	18%

Σχήμα 4 Χρήστες γεω - κοινωνικών υπηρεσιών vs χρήστες υπηρεσιών (based location) [19]

Καταλαβαίνουμε ότι οι χρήστες επιζητούν τέτοιες πληροφορίες / οδηγίες. Το πρόβλημα της μόλυνσης του περιβάλλοντος και η ατμοσφαιρική ρύπανση που επικρατεί λόγω αυτής της μόλυνσης σε μία τοποθεσία μια δεδομένη χρονική στιγμή είναι πληροφορίες που μπορεί να χρειαστεί ο χρήστης ενός «έξυπνου» τηλεφώνου.

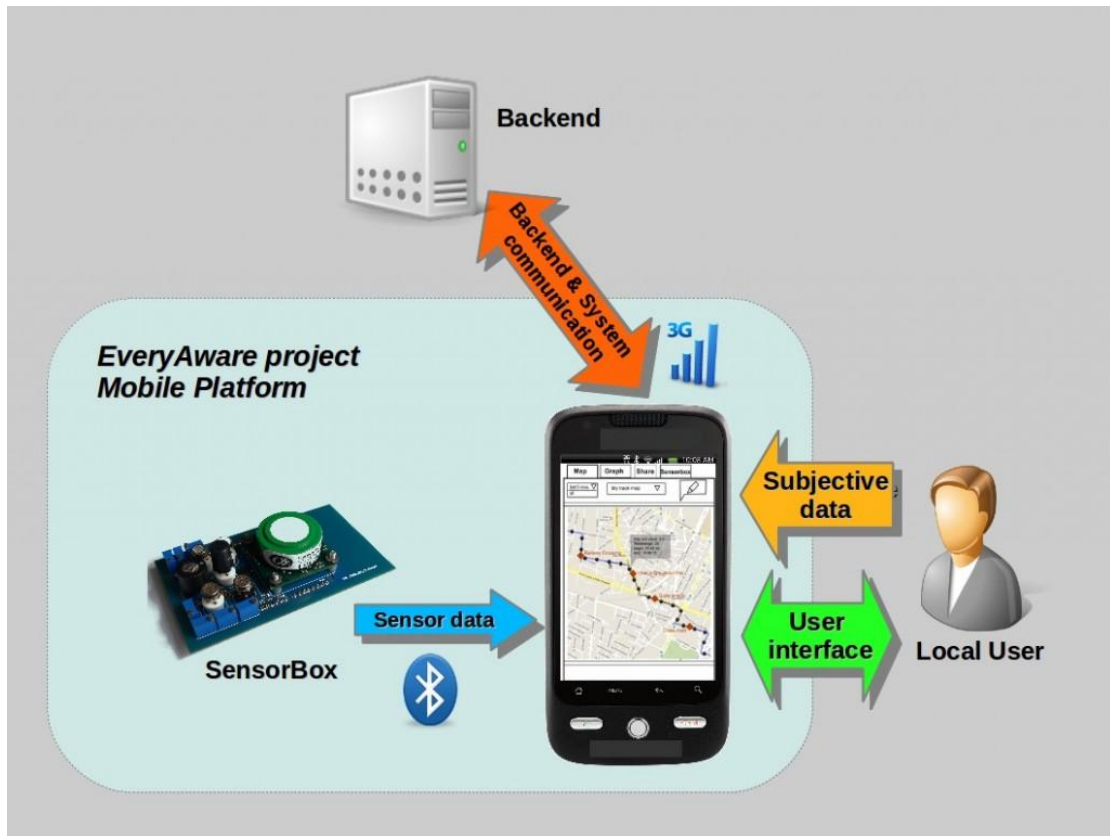
3.1 ΜΕΛΕΤΗ ΠΕΡΙΠΤΩΣΗΣ AIR QUALITY

Η μελέτη περίπτωσης που θα δούμε παρακάτω έχει πραγματοποιηθεί από το κονσόρτσιουμ EveryAware [16] που ασχολείται με την περιβαλλοντική ευαισθητοποίηση μέσω των κοινωνικών πληροφοριών χρησιμοποιώντας την τεχνολογία (Enhance Environmental Awareness through Social Information Technologies).

Η μελέτη περίπτωσης στηρίζεται σε τρία συστατικά:

- το κουτί το οποίο περιέχει έναν αισθητήρα (SensorBox)
- ένα «έξυπνο» τηλέφωνο με εγκατεστημένη την εφαρμογή AirProbe
- ένα εξυπηρετητή Παγκόσμιου Ιστού (Web Server) και εφαρμογές Παγκόσμιου Ιστού (Web Applications)

Το SensorBox είναι μια φορητή συσκευή η οποία μετρά τις συγκεντρώσεις των ρύπων στον αέρα και τις εντοπίζει μέσω του GPS. Αυτό γίνεται μία φορά ανά δευτερόλεπτο προκειμένου να υπάρχει ένας λεπτομερής χάρτης της ρύπανσης όταν ο χρήστης περπατάει. Χρησιμοποιώντας ένα «έξυπνο» τηλέφωνο που έχει εγκατεστημένη την εφαρμογή AirProbe είναι δυνατό ο χρήστης να ξέρει την μέτρηση σε πραγματικό χρόνο χρησιμοποιώντας το Bluetooth της συσκευής προκειμένου να επικοινωνήσει με το SensorBox. Επιπλέον η εφαρμογή χρησιμοποιώντας μια σύνδεση Internet μπορεί να στέλνει όλες τις μετρήσεις στον εξυπηρετητή της EveryAware ο οποίος συλλέγει σε πραγματικό χρόνο τα δεδομένα από όλους τους χρήστες. Μέσω διαδικτυακών εφαρμογών οι χρήστες μπορούν να έχουν πρόσβαση στα δεδομένα του εξυπηρετητή, τα οποία έχουν συλλεχθεί από το σύνολο των χρηστών, όπως εξατομικευμένες πληροφορίες σχετικά με τα προσωπικά επίπεδα έκθεσης στους ρύπους.



Σχήμα 5 Συνολική λειτουργία μελέτη περίπτωσης Air Quality [15]

SensorBox

Το SensorBox είναι ένα μικρό σύστημα μέτρησης που χρησιμοποιεί διαφορετικούς αισθητήρες αερίου χαμηλού κόστους για την αξιολόγηση της ποιότητας του αέρα. Είναι αρκετά μικρό για να χωρέσει σε ένα σακίδιο και τροφοδοτείται με μια εξωτερική μπαταρία.

Το κουτί έχει ένα μικρό ανεμιστήρα που ωθεί εξωτερικό αέρα σε ένα θάλαμο όπου υπάρχουν διαφορετικοί αισθητήρες που μετρούν ατμοσφαιρικούς ρύπους. Προκειμένου να υλοποιηθεί μια μικρή και φθηνή συσκευή έχουν χρησιμοποιηθεί μια σειρά από αισθητήρες αερίων χαμηλούς κόστους. Αυτός είναι και ο λόγος που σε ελεγχόμενο περιβάλλον έχουν καλές επιδόσεις ενώ σε πραγματικές συνθήκες η ακρίβεια των μετρήσεων επηρεάζονται από μια σειρά παραγόντων.

Η εφαρμογή για έξυπνα τηλέφωνα AirProbe

Η AirProbe επιτρέπει στους χρήστες να παρακολουθούν την ατμοσφαιρική ρύπανση και να εμπλουτίζουν αυτά που διαβάζουν με προσωπικές επισημειώσεις. Παρουσιάζει την διεπαφή μεταξύ του SensorBox και του κεντρικού εξυπηρετητή και είναι σχεδιασμένη για να:

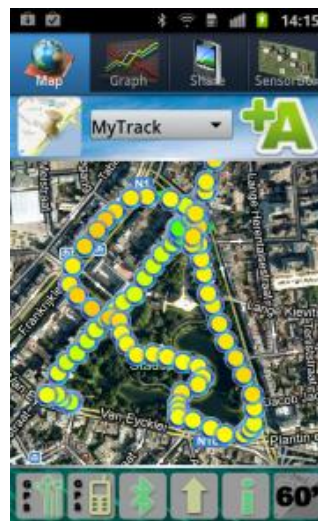
- παρουσιάζει πληροφορίες σχετικά με την τρέχουσα ατμοσφαιρική ρύπανση

- καταγράφει την διαδρομή του χρήστη και να ανακτά σε πραγματικό χρόνο πληροφορίες από το SensorBox προκειμένου να τις προβάλει στον χρήστη σε πραγματικό χρόνο
- επιτρέπει στον χρήστη να επισημαίνει την διαδρομή του. Κάθε ενέργεια του χρήστη γίνεται με απλό και γρήγορο τρόπο. Τέτοιες ενέργειες είναι το άνοιγμα της Bluetooth σύνδεσης, η καταγραφή της διαδρομής και η προβολή της στον χάρτη καθώς και προσωπικές επισημειώσεις οι οποίες πραγματοποιούνται εύκολα και σε μερικά βήματα.

Μια εγγραφή είναι η διαδρομή που ένας χρήστης κάνει και δημιουργείται όταν ο χρήστης ανοίγει το SensorBox και ολοκληρώνεται όταν το SensorBox κλείνει. Στην διάρκεια μεταξύ ανοίγματος και κλεισίματος η σύνδεση μεταξύ SensorBox και «έξυπνου» τηλεφώνου είναι ενεργή.

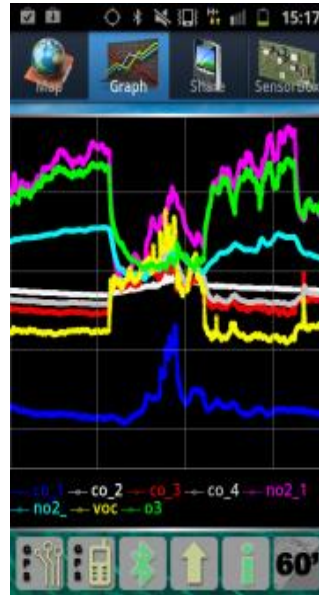
Η εφαρμογή αποτελείται από διαφορετικές προβολές που αναπαριστούν διαφορετικές λειτουργίες και οι οποίες συνδέονται μεταξύ τους με καρτέλες (tabs):

- Map: Υπάρχουν δύο είδη προβολής χαρτών. Ο πρώτος χάρτης παρουσιάζει την εγγραφή του χρήστη η οποία δημιουργείται σε πραγματικό χρόνο από το SensorBox και ο δεύτερος χάρτης παρουσιάζει τις επισημειώσεις οι οποίες γίνονται από διάφορους χρήστες γύρω από την τρέχουσα τοποθεσία του χρήστη.



Σχήμα 6 Καρτέλα Map [15]

- Graph: γραφικές παραστάσεις πραγματικού χρόνου που δείχνουν τις τρέχουσες τιμές που έχει λάβει η εφαρμογή από τους αισθητήρες του SensorBox.



Σχήμα 7 Καρτέλα Graph [15]

- Share: επιτρέπει στον χρήστη να συνδεθεί στο Facebook και στο Twitter και να κοινοποιήσει πληροφορίες μόλυνσης
- Sensor box: παρουσιάζει χρήσιμες πληροφορίες για την κατάσταση της Bluetooth σύνδεσης, τεχνικές πληροφορίες για τους αισθητήρες όπως επίσης και για το Gps.



Σχήμα 8 Καρτέλα Sensor Box [15]

Η εφαρμογή χρειάζεται σύνδεση στο Internet προκειμένου να στείλει δεδομένα στον εξυπηρετητή και να προβληθούν στους χάρτες της Google. Για την αποφυγή απώλειας δεδομένων, χρησιμοποιεί έναν μηχανισμό που επιτρέπει την αποθήκευση των

δεδομένων σε τοπικό επίπεδο προκειμένου όταν το «έξυπνο» τηλέφωνο συνδεθεί και πάλι στο Internet να τα στείλει στον διακομιστή.

Υιοθέτηση διαφορετικών προφίλ χρηστών

Οι λειτουργίες και ο τρόπος αλληλεπίδρασης του χρήστη με την εφαρμογή AirProbe γίνεται με απλό τρόπο προκειμένου οι διαφορετικοί χρήστες να χρησιμοποιούν την εφαρμογή με έναν απλό τρόπο. Για παράδειγμα, εάν ο χρήστης είναι ποδηλάτης η απλή διεπαφή της εφαρμογής του δίνει γρήγορα την δυνατότητα να ελέγχει την ποιότητα του αέρα, να εισάγει σχόλια, να καταγράφει διαδρομές και να στείλει πληροφορίες στον απομακρυσμένο εξυπηρετητή.

3.2 ΜΕΛΕΤΗ ΠΕΡΙΠΤΩΣΗΣ ΗΚ ΑQHI (HONG KONG AIR QUALITY INDEX)

Η κυβέρνηση του Χονγκ Κονγκ σε μια προσπάθεια να αντιμετωπίσει το τεράστιο πρόβλημα μόλυνσης που αντιμετωπίζει η χώρα και για την προστασία των πολιτών, τον Δεκέμβριο του 2013 ανακοίνωσε την ανάπτυξη εφαρμογής για «έξυπνα» τηλέφωνα με σκοπό την ενημέρωση των πολιτών σε πραγματικό χρόνο για την ποιότητα του αέρα που επικρατεί σε περιοχές της χώρας. Η εφαρμογή με την ονομασία ΗΚ ΑQHI κυκλοφόρησε τον Ιανουάριο του 2014 [7].

Η εφαρμογή προσφέρει στους χρήστες ειδοποιήσεις σε πραγματικό χρόνο για την ποιότητα του αέρα και για το εάν θα πρέπει να παραμείνουν σε εσωτερικούς χώρους σε περιόδους υψηλής ρύπανσης.

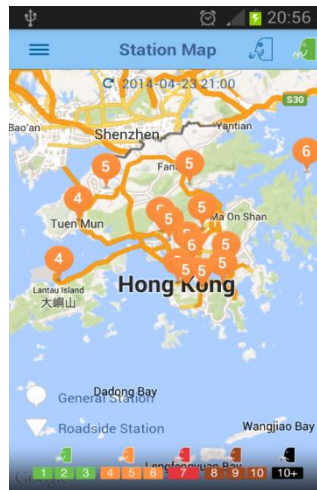
Πρόκειται για μια πρωτοβουλία του Τμήματος Προστασίας Περιβάλλοντος προκειμένου οι πολίτες να προστατευθούν από την μόλυνση. Για παράδειγμα με την βοήθεια της εφαρμογής μπορεί ένας δάσκαλος να καθορίσει εάν πρέπει τα παιδιά να βγούνε για διάλειμα έξω. Πρόκειται για ένα εργαλείο το οποίο είναι πολύ χρήσιμο για τα παιδιά, τους ηλικιωμένους καθώς και για ανθρώπους που οι δραστηριότητες τους είναι κυρίως στον εξωτερικό χώρο.

Η εφαρμογή είναι διαθέσιμη τόσο σε πλατφόρμα iOS όσο και σε Android. Παρουσιάζει τις συγκεντρώσεις ρύπων των PM_{10} και $PM_{2.5}$ που καταγράφονται σε 15 συνολικά σταθμούς (12 γενικοί σταθμοί και 3 οδικοί σταθμοί) τους οποίους έχει εγκαταστήσει το Τμήμα Προστασίας Περιβάλλοντος.

Η μέτρηση PM_{10} αναφέρεται στα αναπνεύσιμα αιωρούμενα σωματίδια τα οποία είναι μικροσκοπικά στίγματα των ρύπων που μπορούν να διεισδύσουν στους πνεύμονες και να προκαλέσουν καρκίνο (μέτρηση 10 ή λιγότερο). Αντίθετα, η μέτρηση $PM_{2.5}$ αφορά ένα μείγμα σωματιδίων το οποίο μπορεί εκτός από το να επηρεάσει αρνητικά την ανθρώπινη υγεία να προκαλέσει επιπτώσεις στο κλίμα και καθίζηση του εδάφους (μέτρηση 2.5 ή λιγότερο). Η εφαρμογή χρησιμοποιώντας την κλίμακα (AQHI – Air Quality Health Index) ενημερώνει για τη επικινδυνότητα των ρύπων που υπάρχουν στην ατμόσφαιρα.

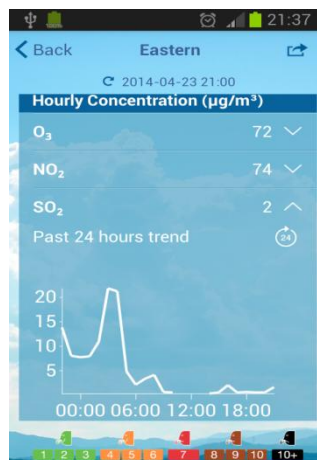
3.2.1 ΕΙΣΑΓΩΓΙΚΗ ΟΘΟΝΗ & ΟΘΟΝΗ ΜΕΤΡΗΣΕΩΝ

Στην εισαγωγική οθόνη της εφαρμογής βλέπουμε ένα χάρτη του Χονγκ Κονγκ στον οποίο απεικονίζονται όλοι οι σταθμοί οι οποίοι μετράνε την μόλυνση του αέρα. Κάθε σταθμός έχει συγκεκριμένο χρώμα προκειμένου ο χρήστης χρησιμοποιώντας την κλίμακα μέτρησης της μόλυνσης που βρίσκεται στο κάτω μέρος της οθόνης να γνωρίζει με μια ματιά την μόλυνση του αέρα σε κάθε περιοχή. Η κλίμακα αυτή έχει 5 επίπεδα επικινδυνότητας ξεκινώντας από το 1 έως 3 (ελάχιστος κίνδυνος) έως το 10+(αυξημένος κίνδυνος).



Σχήμα 9 Αρχική Οθόνη ΗΚ ΑQH1

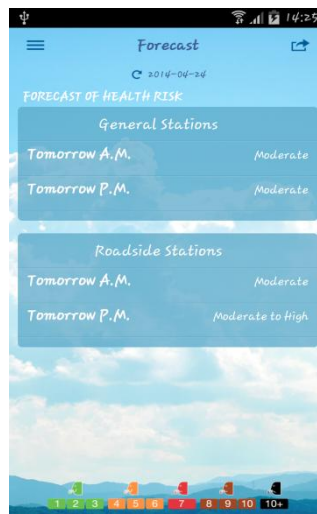
Ο χρήστης μπορεί επιλέγοντας μια περιοχή να λάβει πληροφορίες για τις μετρήσεις των τελευταίων εικοσιτεσσάρων ωρών σχετικά με τα επίπεδα Όζοντος (O_3), τα επίπεδα Οξειδίου (NO_2), τα επίπεδα του Διοξειδίου του Θείου (SO_2), τα επίπεδα PM_{10} καθώς και τα επίπεδα $PM_{2.5}$. Οι μετρήσεις για κάθε κατηγορία είναι διαθέσιμες και σε αριθμούς (περιεκτικότητα ανά κυβικό μέτρο - $\mu g/m^3$) αλλά και ως γραφική παράσταση.



Σχήμα 10 Οθόνη Μετρήσεων

3.2.2 ΑΛΛΕΣ ΟΘΟΝΕΣ

Πέρα από την αρχική οθόνη και την οθόνη μετρήσεων η εφαρμογή μέσω άλλων οθονών δίνει και άλλες δυνατότητες στον χρήστη. Ο χρήστης μπορεί να έχει πρόσβαση σε αυριανή πρόγνωση πιθανής μόλυνσης του αέρα και έτσι να κανονίζει τις μετακινήσεις του (forecast). Μπορεί επίσης να επιλέξει να προβάλει τις μετρήσεις σύμφωνα με την κλίμακα ποιότητας του αέρα τόσο ανά περιοχή όσο και ανά σταθμό (AQHI). Επίσης δίνονται συμβουλές υγείας για ηλικιωμένους, παιδιά, ανθρώπους με προβλήματα στην καρδιά και για ανθρώπους που είναι καλά στην υγεία τους (health advice). Τέλος, η εφαρμογή δίνει την δυνατότητα των ρυθμίσεων καθώς και οδηγίες για το πώς θα την χρησιμοποιήσει ο χρήστης.



Σχήμα 11 Οθόνη Πρόγνωσης (forecast)

3.3 ΣΥΓΚΡΙΤΙΚΗ ΑΞΙΟΛΟΓΗΣΗ ΥΦΙΣΤΑΜΕΩΝ ΛΥΣΕΩΝ

Οι υπηρεσίες που αναφέρθηκαν στην προηγούμενη ενότητα έχουν ως σκοπό να επεκτείνουν τον τομέα των υπηρεσιών με βάση την τοποθεσία (location – based). Πρόκειται για καινοτόμες υπηρεσίες με προσωποποιημένο περιεχόμενο οι οποίες χρησιμοποιούν την τεχνολογία προκειμένου με τα χαρακτηριστικά τους να αναδείξουν την χρησιμότητα των «έξυπνων» τηλεφώνων στην καθημερινότητα των χρηστών προστατεύοντας τους από την έκθεση σε περιβάλλοντα μόλυνσης.

Πιο συγκεκριμένα, η μελέτη περίπτωσης AirQuality είναι μια ολοκαίνουρια υπηρεσία η οποία δεν έρχεται να αντικαταστήσει ή να συμπληρώσει κάποια προηγούμενη σε αντίθεση με την υπηρεσία HK AQHI η οποία επεκτείνει και βελτιώνει την ενημέρωση των πολιτών σχετικά με την μόλυνση καθώς παρέχει πληροφορίες σε πραγματικό χρόνο σε σχέση με ένα ραδιόφωνο, μια τηλεόραση ή μια εφημερίδα για την χώρα του Χονγκ Κονγκ. Η εδραίωση της υπηρεσίας ίσως οδηγήσει και σε εξάλειψη της χρησιμοποίησης αυτών των μέσων για την εξάπλωση τέτοιου είδους πληροφοριών.

Κοινά χαρακτηριστικά και των δύο είναι ότι χρησιμοποιούν το «έξυπνο» τηλέφωνο ως μέσο εξάπλωσης της πληροφορίας και αισθητήρες μέτρησης αερίων για την καταγραφή μετρήσεων. Βασική διαφορά των δύο είναι ότι στην μεν πρώτη μπορεί το «έξυπνο» τηλέφωνο να λαμβάνει πληροφορίες απευθείας από τους αισθητήρες μετρήσεων ενώ η δεύτερη λαμβάνει τις πληροφορίες από εξυπηρετητή του Τμήματος Προστασίας Περιβάλλοντος.

Σχετικά με το προσωποποιημένο περιεχόμενο το οποίο παρέχουν, αξίζει να αναφέρουμε ότι και οι δύο λαμβάνουν υπόψη τους τις ανάγκες ενός χρήστη «έξυπνου» τηλεφώνου. Οι πληροφορίες οι οποίες δίνουν διευκολύνουν τον χρήστη στο να αντιληφθεί τις πληροφορίες είτε παρέχοντας γραφικές αναπαραστάσεις με κατάλληλους χρωματισμούς για το επίπεδο μόλυνσης (Air Quality) είτε παρέχοντας κλίμακα μέτρησης αερίων με κατάλληλη επεξήγηση (HK AQHI) προκειμένου ο χρήστης να κατανοήσει τις ενδείξεις. Η πρώτη μελέτη περίπτωσης σε σχέση με την μελέτη περίπτωσης HK AQHI παρουσιάζει μεγαλύτερο βαθμό προσωποποιημένου περιεχομένου καθώς καλύπτει την ανάγκη του χρήστη να χαράξει μια διαδρομή προκειμένου να μετακινηθεί από ένα σημείο σε ένα άλλο με την λιγότερη επίπτωση στην υγεία του ενώ στην μελέτη περίπτωσης HK AQHI παρέχεται μόνο η πληροφορία χωρίς να μπορεί ο χρήστης να την εξατομικεύσει στις ανάγκες του.

Κλείνοντας την ενότητα αυτή θα μπορούσαμε να αναφέρουμε ότι οι υφιστάμενες λύσεις που μελετήθηκαν βρίσκονται σε αρχικό στάδιο αλλά προς την σωστή κατεύθυνση για να αποτελέσουν ολοκληρωμένες λύσεις για την βελτίωση, την πρόληψη αναπνευστικών προβλημάτων καθώς και άλλων προβλημάτων υγείας από την μόλυνση του περιβάλλοντος.

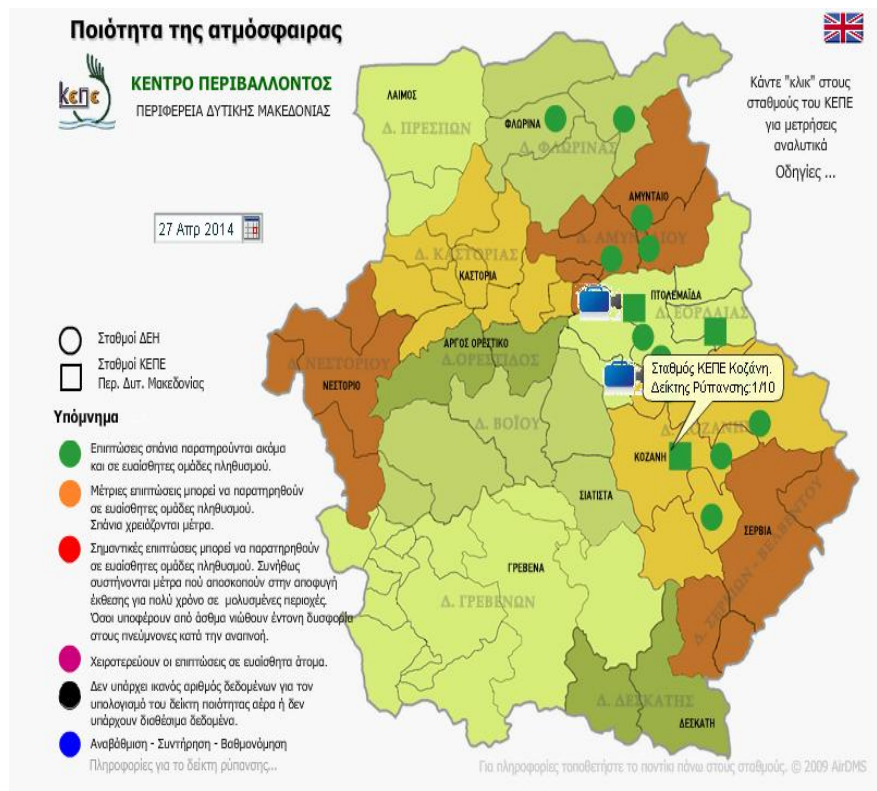
3.4 ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΥΠΗΡΕΣΙΑΣ POLLUTIONTRACKER

Πρωτίστως για την δημιουργία της υπηρεσίας μας θα έπρεπε να βρούμε που βρίσκονται διαθέσιμα τα δεδομένα τα οποία θα μας ήταν απαραίτητα. Αυτές οι πληροφορίες για την περιοχή της Δυτικής Μακεδονίας είναι διαθέσιμες στην ιστοσελίδα του κέντρου περιβάλλοντος της εν λόγω περιφέρειας μιας και το πρόβλημα της μόλυνσης της περιοχής έχει διαγνωσθεί εδώ και καιρό.

Το Κέντρο Περιβάλλοντος (ΚΕΠΕ) της Νομαρχιακής Αυτοδιοίκησης Κοζάνης έθεσε σε εφαρμογή σύστημα πρόσβασης σε δεδομένα ποιότητας της ατμόσφαιρας.

Οι μετρήσεις περιβαλλοντικών δεδομένων γίνονται από 11 σταθμούς που έχει εγκατεστημένους στη Δυτική Μακεδονία η ΔΕΗ. Όταν τα δεδομένα γίνουν «διαθέσιμα» από τη ΔΕΗ, κατεβαίνουν στο ΚΕΠΕ και στη συνέχεια γίνεται ο υπολογισμός του δείκτη ποιότητας του αέρα (της τελευταίας διαθέσιμης ημέρας) και μαζί με τις μέσες ημερήσιες και μέσες ωριαίες τιμές αποστέλλονται στην ιστοσελίδα του ΚΕΠΕ (www.kepekozani.gr).

Ο χρήστης μέσω της ιστοσελίδας του ΚΕΠΕ μπορεί να επιλέξει από την αρχική σελίδα το εικονίδιο «ΠΟΙΟΤΗΤΑ ΑΕΡΑ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ». Από εκεί οδηγείται σε ένα διαδραστικό χάρτη της Περιφέρειας Δυτικής Μακεδονίας, όπου αναπαριστούνται οι θέσεις των σταθμών του δικτύου μέτρησης με κατάλληλους χρωματικούς κύκλους, ανάλογα με την μετρούμενη ρύπανση του αέρα.



Σχήμα 12 Διαδραστικός Χάρτης ΚΕΠΕ (www.kepekozani.gr)

Έχοντας λοιπόν τα δεδομένα που χρειαζόταν η υπηρεσία μας τουλάχιστον για μια γεωγραφική περιοχή διαθέσιμα στο διαδίκτυο αυτό που χρειαζόταν ήταν να γίνουν διαθέσιμα και στην υπηρεσία την οποία θα αναπτύσσαμε για «έξυπνα» τηλέφωνα.

Ο αρχικός σχεδιασμός της υπηρεσίας περιελάμβανε ότι ο χρήστης θα μπορούσε να έχει προσωποποιημένο περιεχόμενο στην συσκευή του ανάλογα με την τοποθεσία την οποία βρίσκεται. Ο χρήστη εκμεταλλευόμενος την τεχνολογία του GPS στα «έξυπνα» τηλέφωνα θα μπορούσε με βάση την τοποθεσία του να πάρει πληροφορίες σχετικά με την ποιότητα του αέρα που υπάρχει στην συγκεκριμένη τοποθεσία. Επομένως ο χρήστης θα αρκούσε να προβάλλει τις πληροφορίες τις οποίες θα έπαιρνε αυτόματα σύμφωνα με την θέση του και όχι να παραμετροποιήσει την εφαρμογή.

Παρόλα αυτά και χωρίς την ύπαρξη κάποιας διεπαφής προγράμματος εφαρμογής (Application User Interface) αυτό αποδείχτηκε μια ιδιαίτερη δυσκολία για την περιοχή της Δυτικής Μακεδονίας. Η έλλειψη δυνατότητας λήψης των δεδομένων με τον παραδοσιακό τρόπο επικοινωνίας δύο υπηρεσιών – εφαρμογών μας ανάγκασε να

στραφούμε σε μια διαφορετική λύση. Η λύση αυτή υποχρέωνε τη εφαρμογή μας να δέχεται, είτε αρχικά είτε μέσω των ρυθμίσεων της, την διεύθυνση της ιστοσελίδας για κάθε σταθμό που επιθυμεί να προβάλει τα δεδομένα του ο χρήστης έτσι ώστε να διαβαστούν από την ιστοσελίδα και να γίνουν διαθέσιμα στην εφαρμογή και επομένως και στον χρήστη.

Σκοπός μας ήταν να δημιουργήσουμε όπως προαναφέραμε μια καινοτόμο εφαρμογή. Αυτό για να γίνει έπρεπε να πληρούνται κάποια συγκεκριμένα χαρακτηριστικά. Το κυριότερο χαρακτηριστικό της είναι ότι πρόκειται για μια υπηρεσία που δεν υπάρχει ανάλογη τουλάχιστον για «έξυπνα» τηλέφωνα στον χώρο της Ελλάδας. Η εφαρμογή επομένως έρχεται να επεκτείνει τον κλάδο τόσο της ηλεκτρονικής υγείας όσο και της ηλεκτρονικής διακυβέρνησης μιας και οι πληροφορίες είναι διαθέσιμες από εγκατεστημένους σταθμούς της περιφέρειας αλλά και γιατί χρησιμοποιώντας τεχνολογικά μέσα όπως τα «έξυπνα» τηλέφωνα ο χρήστης μπορεί να προστατευθεί από μελλοντικά ή υπαρκτά προβλήματα υγείας. Επιπρόσθετα στοχεύει να καταστήσει μη χρήσιμα διάφορα άλλα μέσα τα οποία μας ενημερώνουν για τέτοιου είδους πληροφορίες (τηλεόραση, ραδιόφωνο, εφημερίδες). Επομένως πληροί κάποια από τα χαρακτηριστικά τα οποία αναφέραμε σε προηγούμενο κεφάλαιο και καθιστούν μια υπηρεσία καινοτόμα.

ΕΠΙΛΟΓΟΣ

Συνοψίζοντας το κεφάλαιο που μόλις είδαμε, αναφερθήκαμε στην τάση των χρηστών «έξυπνων» τηλεφώνων να χρησιμοποιούν εφαρμογές – υπηρεσίες με βάση την τοποθεσία τους. Παρουσιάσαμε δύο μελέτες περίπτωσης για τέτοιες εφαρμογές που υπάρχουν στο εξωτερικό καθώς και την υπηρεσία PollutionTracker την οποία αναπτύξαμε, τονίζοντας τις δυσκολίες που αντιμετωπίσαμε, τους σκοπούς της και τα χαρακτηριστικά που την κάνουν καινοτόμα.

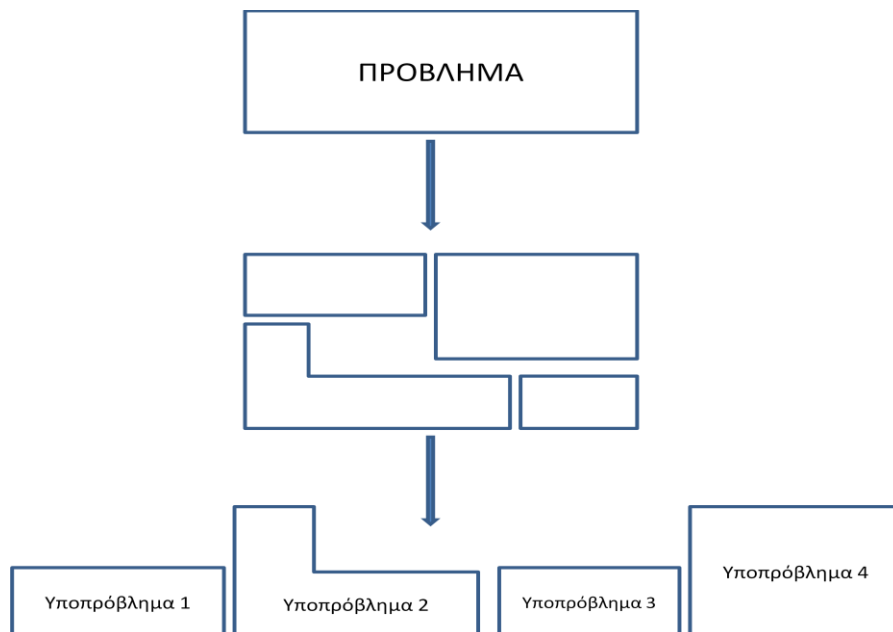
Στην συνέχεια θα αναλύσουμε την υπηρεσία PollutionTracker με βάση την μηχανική λογισμικού παραθέτοντας πίνακες, διαγράμματα και μοντέλα που θα την αναπαριστούν.

ΚΕΦΑΛΑΙΟ 4

ΑΝΑΛΥΣΗ ΥΠΗΡΕΣΙΑΣ

ΕΙΣΑΓΩΓΗ

Κάθε νέα εφαρμογή, η οποία είναι κάτι το εντελώς καινούριο ή έρχεται να αντικαταστήσει μια ήδη υπάρχουσα, που βασίζεται σε λογισμικό πρέπει να έχει ως στόχο την εκπλήρωση ενός συγκεκριμένου σκοπού. Επομένως και η υπηρεσία που δημιουργήθηκε και βασίζεται στην ανάπτυξη λογισμικού για «έξυπνα» κινητά πλατφόρμας Android έχει αυτό τον στόχο. Κατά συνέπεια είναι σημαντικό να έχουμε κατανοήσει το πρόβλημα το οποίο καλούμαστε να αντιμετωπίσουμε πριν ξεκινήσουμε την ανάπτυξη του λογισμικού. Γι αυτό το λόγο θα πρέπει να ξεκινάμε την διερεύνηση ενός προβλήματος με την ανάλυση του. Δηλαδή με τη διαίρεση του σε επιμέρους προβλήματα. Έτσι θα περιγράψουμε το αρχικό πρόβλημα ως μια σειρά από επιμέρους προβλήματα και στην συνέχεια θα είναι ευκολότερη η υλοποίηση του.



Σχήμα 13 Διαδικασία Ανάλυσης [17]

Για την εφαρμογή μας εφαρμόσαμε την Ενοποιημένη Γλώσσα Μοντελοποίησης (UML – Unified Model Language). Πρόκειται για μια πρότυπη γλώσσα μοντελοποίησης λογισμικού. Χρησιμοποιείται για την γραφική απεικόνιση, προσδιορισμό, κατασκευή και τεκμηρίωση των στοιχείων ενός συστήματος λογισμικού. Αποτελείται από ένα σύνολο προσυμφωνημένων όρων, συμβόλων και διαγραμμάτων που επιτρέπουν:

- την εμφάνιση των ορίων ενός συστήματος και των βασικών λειτουργιών του, χρησιμοποιώντας «περιπτώσεις χρήσης» (use-cases) και «actors»

- την επεξήγηση της πραγματοποίησης των περιπτώσεων χρήσης
- την αναπαράσταση μιας στατικής δομής ενός συστήματος χρησιμοποιώντας «διαγράμματα κλάσεων»
- την ροή της εφαρμογής με διαγράμματα ακολουθίας
- τη μοντελοποίηση της συμπεριφοράς των αντικειμένων (μοντέλο δραστηριοτήτων)
- την αποκάλυψη της υλοποίησης της αρχιτεκτονικής
- την επέκταση της λειτουργικότητας

Σύμφωνα με αυτή την γλώσσα παρακάτω παρουσιάζεται αναλυτικά η δομή και οι λειτουργίες τις οποίες αναπτύξαμε στην εφαρμογή μας μέσα από τον πίνακα χρηστών – στόχων, ενός διαγράμματος χρήστη καθώς και η λεκτική περιγραφή αυτής. Επιπλέον παρουσιάζεται το μοντέλο περιοχής προβλήματος, το μοντέλο δραστηριοτήτων, το διάγραμμα ακολουθίας καθώς και το διάγραμμα κλάσεων.

4.1 ΠΙΝΑΚΑΣ ΧΡΗΣΤΩΝ - ΣΤΟΧΩΝ

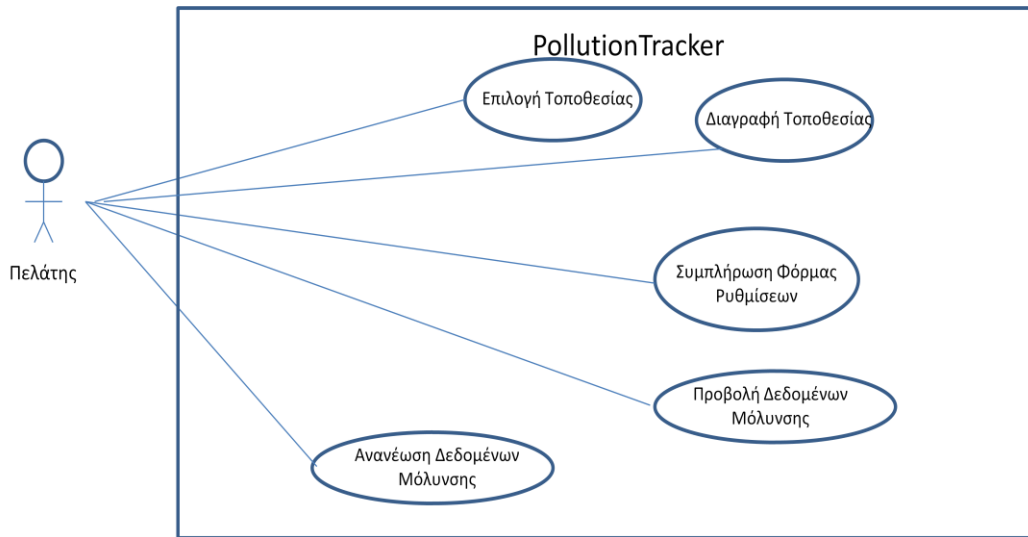
Στην εφαρμογή μας υπάρχει η οντότητα του χρήστη – πελάτη. Ο χρήστης καλείται να χρησιμοποιήσει τις λειτουργίες που επιτελεί η εφαρμογή προκειμένου να εκμεταλλευτεί την υπηρεσία που προσφέρεται από αυτήν. Δηλαδή την διαρκή ενημέρωση των περιβαλλοντολογικών συνθηκών που επικρατούν σύμφωνα με την γεωγραφική του θέση. Στον επόμενο πίνακα παρουσιάζεται ο πίνακας χρηστών στόχων της εφαρμογής μας.

Χρήστες	Στόχοι	Προτεραιότητα(1-5)
Πελάτης	Επιλογή Τοποθεσίας	1
Πελάτης	Συμπλήρωση Φόρμας Ρυθμίσεων	5
Πελάτης	Προβολή Δεδομένων Μόλυνσης	2
Πελάτης	Ανανέωση Δεδομένων Μόλυνσης	3

Σχήμα 14 Πίνακας Χρηστών

4.2 ΔΙΑΓΡΑΜΜΑ ΠΕΡΙΠΤΩΣΕΩΝ ΧΡΗΣΗΣ

Το διάγραμμα περιπτώσεων χρήσης της εφαρμογής μας θέτει τις βάσεις για την δημιουργία της καθώς δίνει έμφαση στις απαιτήσεις που έχει ο πελάτης. Το διάγραμμα αναπαριστά τις λειτουργίες τις οποίες ο χρήστης θα μπορεί να εκτελέσει στην εφαρμογή. Δηλαδή αναπαριστά το σύνολο των λειτουργιών της εφαρμογής. Στο παρακάτω διάγραμμα βλέπουμε το χρήστη να απεικονίζεται ως ένα ανθρωπάκι και οι λειτουργίες να αναπαρίστανται σε μια έλλειψη όπου μέσα αναγράφεται τι πραγματοποιεί η κάθε μια. Η συσχέτιση μεταξύ τους απεικονίζεται με μια γραμμή ενδιάμεσα.



Σχήμα 15 Διάγραμμα Περίπτωσης Χρήσης

4.3 ΛΕΚΤΙΚΗ ΠΕΡΙΓΡΑΦΗ ΠΕΡΙΠΤΩΣΗΣ ΧΡΗΣΗΣ

Πέρα όμως από την διαγραμματική αναπαράσταση των περιπτώσεων χρήσης υπάρχει και η λεκτική περιγραφή μιας περίπτωσης χρήσης. Παρακάτω θα δούμε ενδεικτικά μια που αφορά την υπηρεσία μας και αναπαριστά την συμπλήρωση της φόρμας ρυθμίσεων από τον χρήστη.

Περίπτωση Χρήσης ΠΧ1: Συμπλήρωση φόρμας ρυθμίσεων

Κύριος χρήστης: Πελάτης

Περιγραφή: Πραγματοποιεί προσθήκη μια νέας τοποθεσίας για να ελέγχει τα επίπεδα μόλυνσης

Εμπλεκόμενοι: Πελάτης, θέλει η διεπιφάνεια χρήστη να είναι κατανοητή και να προσθέτει αποτελεσματικά μια νέα τοποθεσία

1.1 Βασική ροή ή Κύριο Σενάριο

	Ενέργειες Χρήστη	Απόκριση Συστήματος
1	Ο πελάτης επιλέγει το κουμπί «Ρυθμίσεις»	
2		Το σύστημα τον μεταφέρει στην οθόνη των ρυθμίσεων
3	Ο πελάτης συμπληρώνει τα πεδία γεωγραφικό μήκος, πλάτος, όνομα τοποθεσίας, διεύθυνση του Παγκόσμιου Ιστού	
4	Ο πελάτης επιλέγει το κουμπί «Προσθήκη τοποθεσίας»	
5		Το σύστημα προσθέτει την νέα τοποθεσία
6	Ο χρήστης επιλέγει το κουμπί επιστροφής και οδηγείται στην αρχική οθόνη	

1.2 Εναλλακτική ροή (Κενό πεδίο Ονόματος)

	Ενέργειες Χρήστη	Απόκριση Συστήματος
1	Ο πελάτης επιλέγει το κουμπί «Ρυθμίσεις»	
2		Το σύστημα τον μεταφέρει στην οθόνη των ρυθμίσεων
3	Ο πελάτης συμπληρώνει το πεδίο διεύθυνσης του Παγκόσμιου Ιστού	
4	Ο πελάτης επιλέγει το κουμπί «Προσθήκη τοποθεσίας»	
5		Το σύστημα προβάλλει μήνυμα «Η τοποθεσία δεν προστέθηκε. Παρακαλώ εισάγετε διεύθυνση και όνομα»

1.3 Εναλλακτική ροή (Μη συμπλήρωση πεδίων)

	Ενέργειες Χρήστη	Απόκριση Συστήματος
1	Ο πελάτης επιλέγει το κουμπί «Ρυθμίσεις»	
2		Το σύστημα τον μεταφέρει στην οθόνη των ρυθμίσεων
3	Ο πελάτης δεν συμπληρώνει κανένα πεδίο	
4	Ο πελάτης επιλέγει το κουμπί «Προσθήκη τοποθεσίας»	
5		Το σύστημα προβάλλει μήνυμα «Η τοποθεσία δεν προστέθηκε. Παρακαλώ εισάγετε διεύθυνση και όνομα»

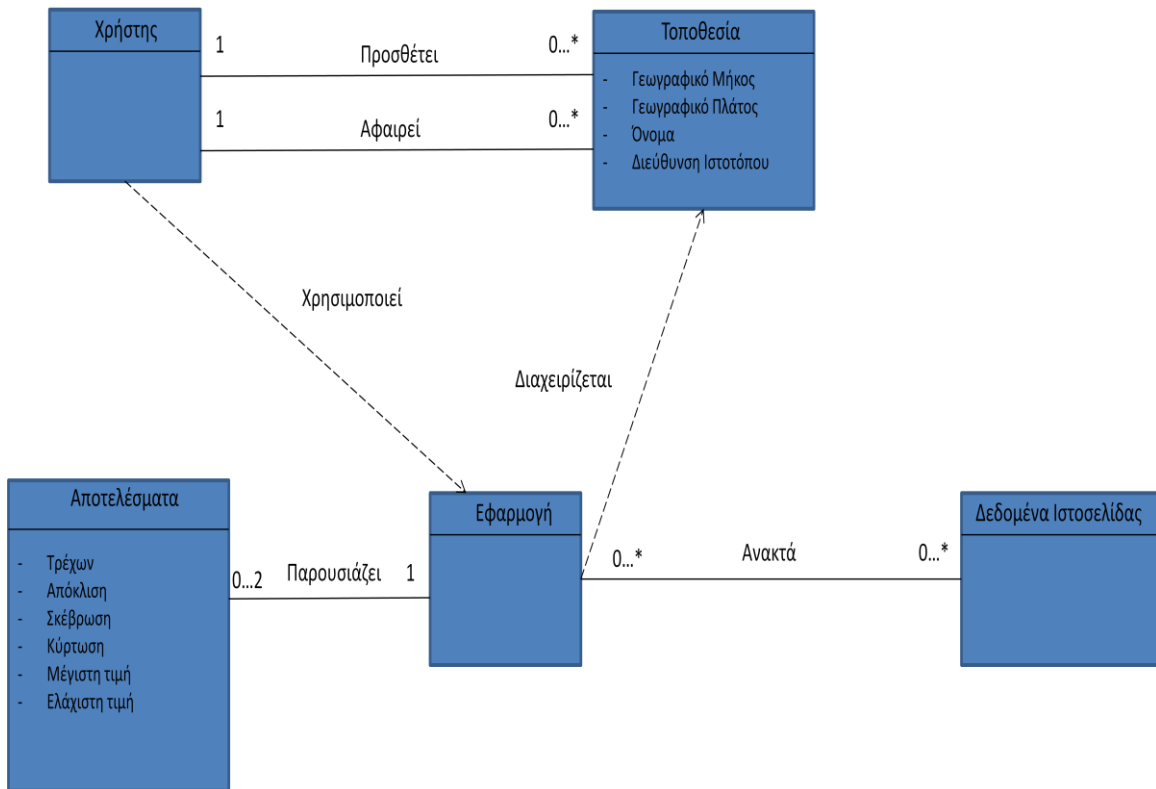
1.4 Εναλλακτική ροή (Διαγραφή Τοποθεσίας)

	Ενέργειες Χρήστη	Απόκριση Συστήματος
1	Ο πελάτης επιλέγει το κουμπί «Ρυθμίσεις»	
2		Το σύστημα τον μεταφέρει στην οθόνη των ρυθμίσεων
3	Ο πελάτης συμπληρώνει τα πεδία γεωγραφικό μήκος, πλάτος, όνομα τοποθεσίας, διεύθυνσης του Παγκόσμιου Ιστού	
4	Ο πελάτης επιλέγει το κουμπί «Προσθήκη τοποθεσίας»	
5		Το σύστημα προσθέτει την νέα τοποθεσία
6	Ο πελάτης επιλέγει την τοποθεσία και πατάει το κουμπί «Διαγραφή Τοποθεσίας»	
7		Το σύστημα διαγράφει την επιλεγμένη τοποθεσία

4.4 ΜΟΝΤΕΛΟ ΠΕΡΙΟΧΗΣ ΠΡΟΒΛΗΜΑΤΟΣ – ΕΝΝΟΙΟΛΟΓΙΚΟ ΜΟΝΤΕΛΟ

Το εννοιολογικό μοντέλο μας δείχνει την περιοχή του προβλήματος και στόχος είναι ο εντοπισμός των σχέσεων περιεκτικότητας μεταξύ των κλάσεων. Στην υπηρεσία PollutionTracker όπως βλέπουμε στον παρακάτω σχήμα ο χρήστης χρησιμοποιεί την εφαρμογή προκειμένου να προσθέσει ή να αφαιρέσει καμία, μία ή περισσότερες τοποθεσίες, δηλαδή υπάρχει μια σχέση μηδέν προς πολλά. Επίσης ο χρήστης χρησιμοποιώντας την εφαρμογή ανακτά από την ιστοσελίδα τα δεδομένα που αυτός επιθυμεί για καμία, μια ή περισσότερες τοποθεσίες και παρουσιάζει τα αποτελέσματα για καμία, μια ή δύο κατηγορίες (PM_{2,5} και PM₁₀).

Πτυχιακή εργασία του φοιτητή Γρηγορίου Χρήστου



Σχήμα 16 Εννοιολογικό Μοντέλο

4.5 ΔΙΑΓΡΑΜΜΑ ΑΚΟΛΟΥΘΙΑΣ

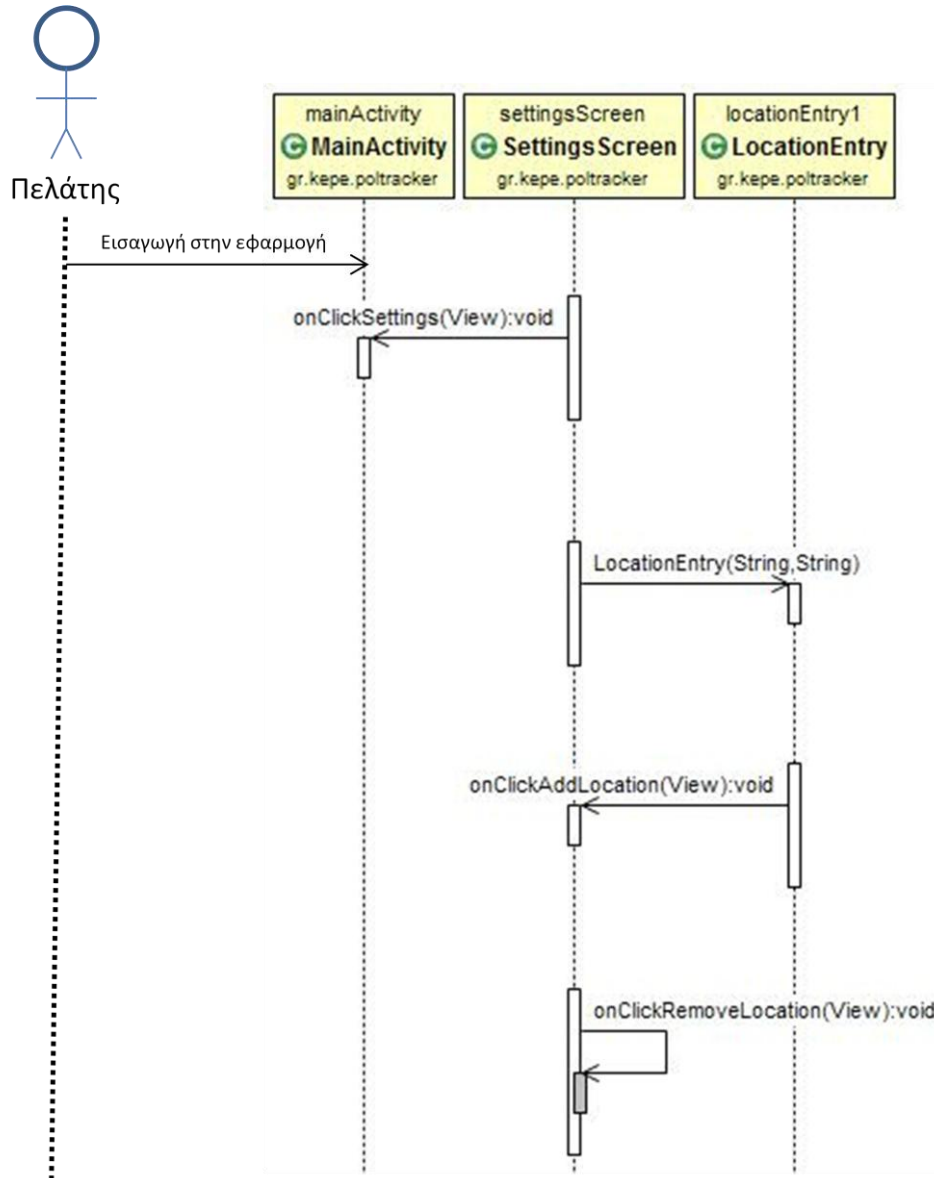
Το διάγραμμα ακολουθίας εκφράζει τα μηνύματα που ανταλλάσσουν τα αντικείμενα του συστήματος μεταξύ τους προκειμένου να ικανοποιήσουν μια μελέτης περίπτωσης. Το παρακάτω διάγραμμα ακολουθίας περιγράφει την περίπτωση χρήσης της συμπλήρωσης φόρμας. Ο χρήστης αφού ανοίξει την εφαρμογή χρησιμοποιεί την MainActivity μέσω της οποίας επιλέγοντας το κουμπί των ρυθμίσεων οδηγείται στην κλάση SettingsScreen. Έπειτα μέσω της κλάσης LocationEntry δημιουργείται η νέα τοποθεσία και με την επιλογή προσθήκης νέας τοποθεσίας προστίθεται. Στην περίπτωση της διαγραφής μίας τοποθεσίας η κλάση LocationEntry δεν αναμειγνύεται σε αυτή την διαδικασία παρά μόνο η SettingsScreen από την οποία αφαιρείται η επιλεγμένη τοποθεσία. Γι αυτό το λόγο και το βελάκι ενώ ξεκινάει από την κλάση SettingsScreen επιστρέφει σε αυτή. Παρακάτω απαριθμούμε τα βήματα τα οποία γίνονται:

Βασική Ροή:

1. Ο πελάτης ανοίγει την εφαρμογή
2. Επιλέγει την μέθοδο onClickSettings (View) και οδηγείται στην κλάση SettingsScreen
3. Εισάγει μια νέα τοποθεσία LocationEntry (String, String)
- 4.1. Προσθέτει επιλέγοντας την μέθοδο onClickAddLocation (View)

Εναλλακτική Ροή:

4.2 Διαγράφη μιας τοποθεσίας καλώντας την μέθοδο onClickRemoveLocation (View)



Σχήμα 17 Διάγραμμα Ακολουθίας

4.6 ΔΙΑΓΡΑΜΜΑ ΚΛΑΣΕΩΝ

Η εφαρμογή μας σχεδιάστηκε για να υλοποιηθούν τέσσερις κλάσεις. Η πρώτη κλάση ονομάζεται **MainActivity** η οποία χειρίζεται την προβολή activity_main και έχει τις μεθόδους:

- **getWebPage (String url)**, η οποία παίρνει μια ιστοσελίδα την οποία έχει εισάγει ο χρήστης και στην συνέχεια την ανακτά

- **onClickUpdate (View)**, η οποία πραγματοποιεί την ανανέωση των αποτελεσμάτων για μια τοποθεσία.
- **onClickSettings (View)**, η οποία με την κλήση της καλείται η οθόνη των ρυθμίσεων
- **onClickDisplayResults (View)**, που οδηγεί στην προβολή των αποτελεσμάτων
- **onCreateOptionsMenu (Menu)**, δημιουργεί το μενού που δίνεται μέσω κουμπιού της συσκευής Android
- **onCreate(Bundle)**, **onPause**, **onResume**, **onDestroy**, αυτές οι μέθοδοι αναλαμβάνουν τον κύκλο ζωής της εφαρμογής και φροντίζουν για την εύρυθμη λειτουργία της

Η επόμενη κλάση είναι η **ResultsScreen** η οποία χειρίζεται την προβολή **activity_results_screen** και έχει τις παρακάτω μεθόδους:

- **parseData()**, η οποία διαβάζει την ιστοσελίδα για να ανακτήσει τα δεδομένα μόνυσης της επιλεγμένης τοποθεσίας
- **getMax(int[])**, επιστρέφει την μέγιστη τιμή από το σύνολο των τιμών για συγκεκριμένο είδος ρύπανσης
- **getMean(double[])**, επιστρέφει την μέση τιμή της τιμής για συγκεκριμένο είδος ρύπανσης(όρισμα πίνακα δεκαδικών αριθμών)
- **getMean(int[])**, επιστρέφει την μέση τιμή της τιμής για συγκεκριμένο είδος ρύπανσης(όρισμα πίνακα ακεραίων αριθμών)
- **getMin(int[])**, επιστρέφει την ελάχιστη τιμή της τιμής για συγκεκριμένο είδος ρύπανσης
- **getSDev(int[], double)**, επιστρέφει την τιμή απόκλισης
- **getSkewness(int[], double, double)**, επιστρέφει την τιμή σκέβρωσης
- **getKurtosis(int[], double, double)**, επιστρέφει την τιμή κύρτωσης
- **onCreate()**, συμμετέχει στην διαχείριση του κύκλου ζωής της εφαρμογής

Η κλάση **SettingsScreen** η οποία χειρίζεται την προβολή **activity_settings_screen** που προβάλλει τις ρυθμίσεις τις οποίες μπορεί να καθορίσει ο χρήστης και έχει τις μεθόδους:

- **onClickAddLocation (View)**, καλείται προκειμένου να προστεθεί μια καινούρια τοποθεσία
- **onClickRemoveLocation (View)**, καλείται προκειμένου να αφαιρεθεί μια υπάρχουσα τοποθεσία
- **onCreate(Bundle)**, **onPause**, **onResume**, **onDestroy**, ίδιο σκοπό με την κλάση MainActivity

Τέλος, η **LocationEntry** η οποία αναπαριστά μια καινούρια τοποθεσία την οποία ο χρήστης προσθέτει στην εφαρμογή. Περιέχει πέντε ιδιότητες οι οποίες αναπαριστούν μία τοποθεσία με τις παρακάτω ιδιότητες

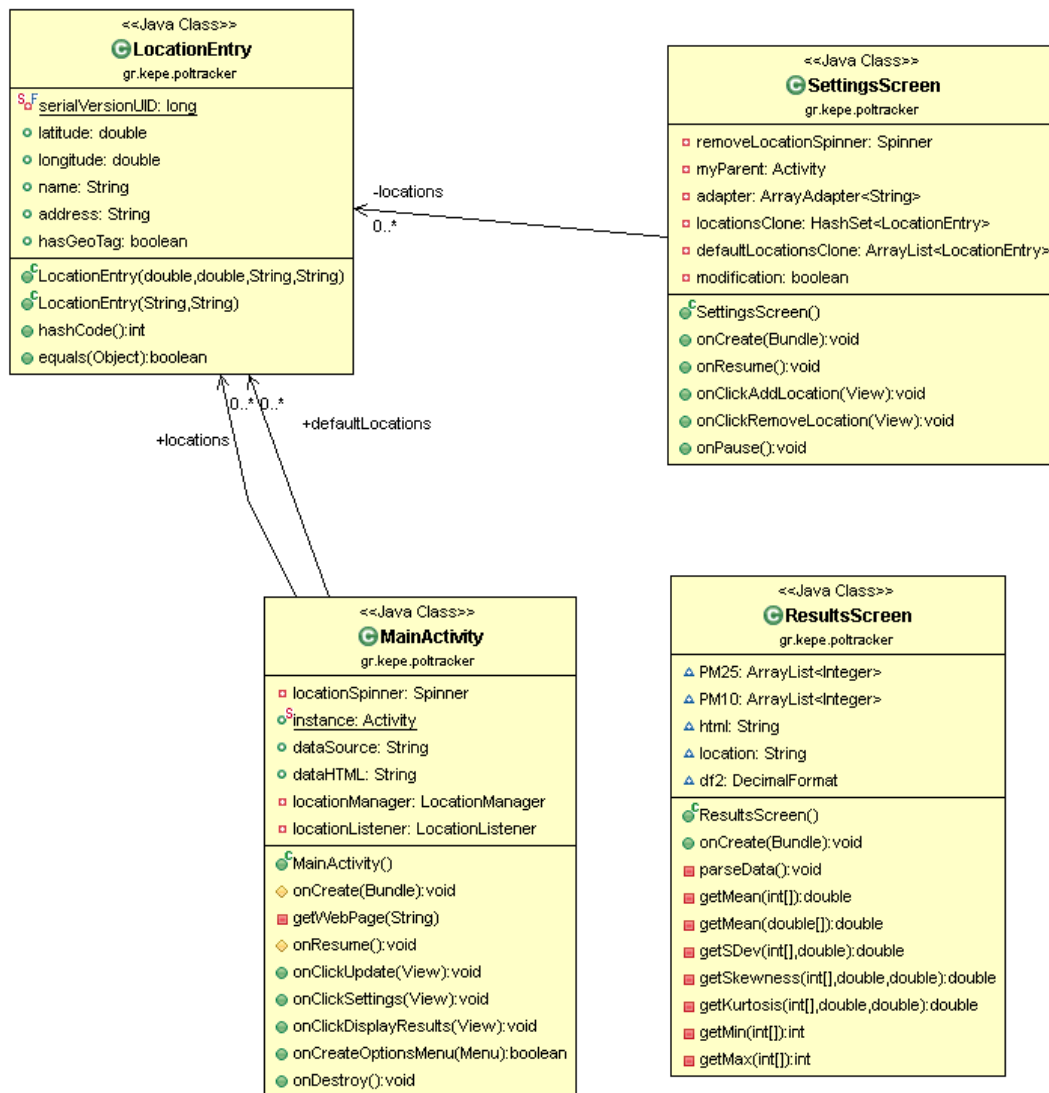
- **latitude** (γεωγραφικό μήκος)

- **longitude** (γεωγραφικό πλάτος)
- **name** (όνομα τοποθεσίας)
- **address** (διεύθυνση του Παγκόσμιου Ιστού)
- **hasGeoTag** (λογική μεταβλητή η οποία αναπαριστά εάν η τοποθεσία έχει γεωγραφικό μήκος και πλάτος)

και μέσω των δομητών (constructors) δημιουργείται η τοποθεσία με ορίσματα είτε το γεωγραφικό μήκος και πλάτος μαζί με τα πεδία name και address είτε με ορίσματα μόνο name και address.

- **LocationEntry (double, double, String, String)**
- **LocationEntry (String, String)**

Στην συνέχεια βλέπουμε το διάγραμμα κλάσεων της εφαρμογής μας και αναλυτικά τι περιέχει η κάθε μια.



Σχήμα 18 Διάγραμμα Κλάσεων

ΕΠΙΛΟΓΟΣ

Το κεφάλαιο που μόλις τελείωσε περιγράφει το πώς αναλύεται η υπηρεσία μας από την σκοπιά της Μηχανικής Λογισμικού. Αναφερθήκαμε σε περιπτώσεις χρήσεις της εφαρμογής μας παρουσιάζοντας βασικές και εναλλακτικές ροές για συγκεκριμένο σενάριο τόσο σε λεκτική μορφή όσο και σε διαγραμματική. Επιπλέον, παρουσιάστηκε το εννοιολογικό μοντέλο της υπηρεσίας, το διάγραμμα ακολουθίας καθώς και το διάγραμμα κλάσεων το οποίο υλοποιήσαμε. Στην συνέχεια θα μελετήσουμε το τεχνολογικό περιβάλλον το οποίο επιτρέπει την ανάπτυξη εφαρμογών για την υπολογιστική πλατφόρμα Android και πιο συγκεκριμένα το ολοκληρωμένο περιβάλλον ανάπτυξης το οποίο χρησιμοποιήσαμε (Integrated development environment - IDE), το πρόγραμμα διεπαφής εφαρμογής του Android που εργαστήκαμε καθώς και την έκδοση εξομοιωτή στον οποίο λειτουργεί η εφαρμογή μας.

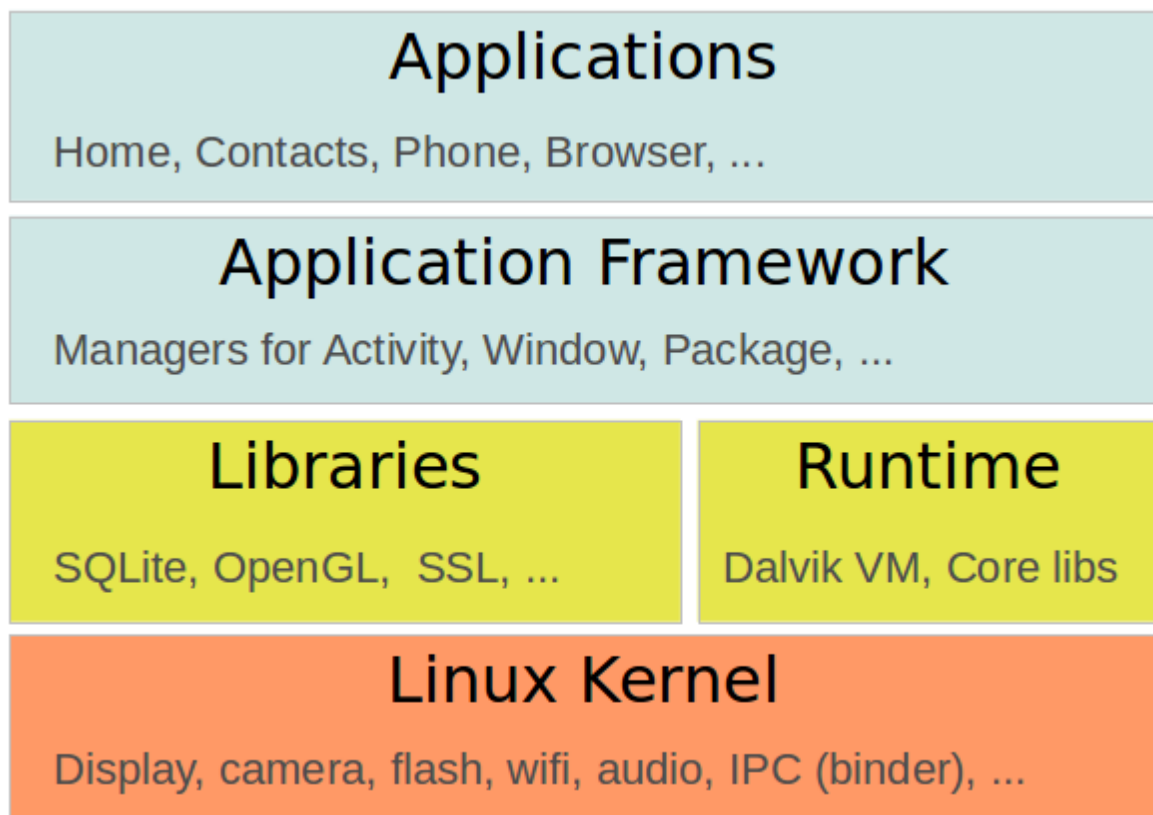
ΚΕΦΑΛΑΙΟ 5

ΤΕΧΝΟΛΟΓΙΚΟ ΠΕΡΙΒΑΛΛΟΝ ΑΝΑΠΤΥΞΗΣ ΚΑΙ ΛΕΙΤΟΥΡΓΙΑΣ

ΕΙΣΑΓΩΓΗ

Η υπηρεσία μας αναπτύχθηκε για «έξυπνα» τηλέφωνα πλατφόρμας Android. Πρόκειται για ένα λειτουργικό σύστημα για κινητές συσκευές το οποίο βασίζεται σε Linux. Το Android είναι ένα έργο το οποίο καλείται Android Open Source Project (AOSP) και καθοδηγείται από την Google.

Το Android είναι ένα σύστημα το οποίο αποτελείται από λογισμικό το οποίο διαχωρίζεται σε τέσσερις περιοχές όπως φαίνεται και στο παρακάτω σχήμα (19):



Σχήμα 19 Επίπεδα Google Android

- Εφαρμογές – Υπάρχουν προεγκατεστημένες εφαρμογές που αφορούν την πλοήγηση (Browser), την κάμερα, την μουσική, το τηλέφωνο κ.α.
- Πλαίσιο εφαρμογών – Ένα πρόγραμμα διεπαφής εφαρμογών το οποίο επιτρέπει την αλληλεπίδραση με το σύστημα μέσω των εφαρμογών
- Βιβλιοθήκες/Κώδικας Android – Περιέχονται βιβλιοθήκες για πολλές συνηθισμένες λειτουργίες (αποθήκευση δεδομένων, πλοήγηση στο διαδίκτυο κ.α.) στο πλαίσιο εφαρμογών και ο κώδικας Android που περιέχει τις βασικές βιβλιοθήκες και το λογισμικό Dalvik που εκτελούν τις εφαρμογές Android.

- Πυρήνας Linux – περιοχή για την επικοινωνία με το hardware κομμάτι μιας συσκευής

Οι περιοχές του πυρήνα Linux, των βιβλιοθηκών και του κώδικα Android είναι ενσωματωμένες στο πλαίσιο εφαρμογών. Οι προγραμματιστές Android χρησιμοποιούν κατά κύριο λόγο τις δύο πρώτες περιοχές για την ανάπτυξη Android εφαρμογών.

Η Google προκειμένου να αναπτυχθεί η πλατφόρμα του Android και να βοηθήσει τους προγραμματιστές να αναπτύξουν εφαρμογές για τις συσκευές Android δημιούργησε ένα πακέτο ανάπτυξης εφαρμογών που είναι ευρέως διαδεδομένο ως SDK (Software Development Kit). Αυτό το SDK στην πορεία της εξέλιξης του Android αναπτύχθηκε βελτιώνοντας κάθε φορά τα χαρακτηριστικά του. Δίνει στον προγραμματιστή τα κατάλληλα εργαλεία για την ανάπτυξη, την μεταγλώττιση και την δημιουργία Android εφαρμογών. Ο πρωτεύων τρόπος για την ανάπτυξη μιας εφαρμογής Android είναι με την χρήση της γλώσσα προγραμματισμού Java.

5.1 ΕΡΓΑΛΕΙΑ ΑΝΑΠΤΥΞΗΣ ANDROID

Το Android SDK που προαναφέραμε, περιέχει ένα εργαλείο το οποίο ονομάζεται Android debug bridge και το οποίο δίνει την δυνατότητα σε ένα προγραμματιστή να συνδεθεί με μια συσκευή πραγματική ή μη και να διαχειριστεί ή να διορθώσει λάθη σε μια εφαρμογή που αναπτύσσει.

Η Google επίσης παρέχει δύο ολοκληρωμένα περιβάλλοντα ανάπτυξης (IDE) για την ανάπτυξη εφαρμογών. Τα εργαλεία ανάπτυξης Android (Android Developer Tools - ADT) βασίζονται στο Eclipse IDE και είναι ένα σύνολο από συστατικά τα οποία επεκτείνουν την λειτουργία του Eclipse IDE προκειμένου να μπορεί να χρησιμοποιηθεί για την ανάπτυξη Android εφαρμογών.

Πρόσφατα δημιουργήθηκε ένα καινούριο IDE το οποίο υποστηρίζεται από την Google και ονομάζεται Android Studio. Το εν λόγω IDE βασίζεται στο IntelliJ IDE.

Και τα δύο IDE τα οποία υποστηρίζονται παρέχουν πλήρη υποστήριξη για την ανάπτυξη, την μεταγλώττιση και εκτέλεση Android εφαρμογών. Επίσης, επιτρέπουν στον προγραμματιστή να δημιουργήσει και να θέσει σε λειτουργία εικονικές συσκευές πλατφόρμας Android για την δοκιμή εφαρμογών. Επιπλέον παρέχουν ειδικούς συντάκτες (editors) για συγκεκριμένα είδη αρχείων. Τα περισσότερα αρχεία αρχικοποίησης στο Android είναι σε μορφή XML. Σε αυτή την περίπτωση και τα δύο IDE επιτρέπουν την εναλλαγή μεταξύ του αρχείου XML και της διεπαφής που επιτρέπει την εισαγωγή δεδομένων ή συστατικών από τον χρήστη.

5.2 ΤΕΧΝΟΛΟΓΙΚΟ ΠΕΡΙΒΑΛΛΟΝ ΑΝΑΠΤΥΞΗΣ

Οι απαιτήσεις που πρέπει να έχει ένα σύστημα προκειμένου να δημιουργήσει εφαρμογές Android είναι οι παρακάτω:

Λειτουργικό Σύστημα

- Windows XP (32bit), Vista (32 ή 64 bit) ή Windows 7 (32 ή 64 bit)
- Mac OS X 10.5.8 ή νεότερο
- Linux

Eclipse IDE

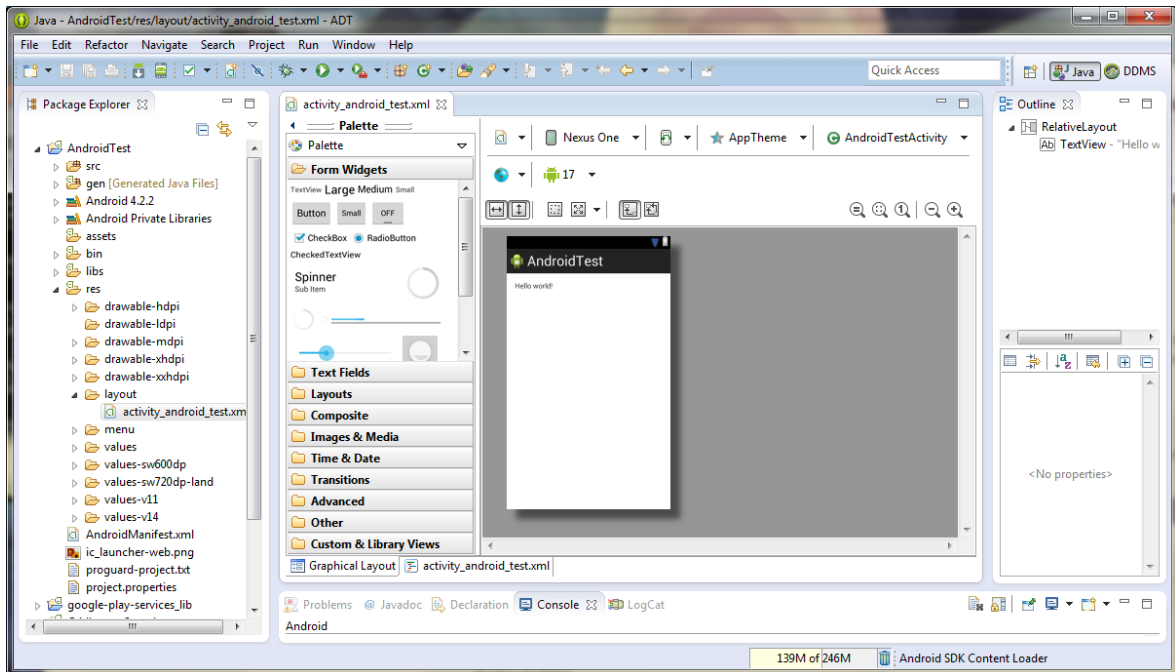
- Eclipse 3.7.2 (Indigo) ή μεγαλύτερο
- Eclipse JDT plug-in
- JDK 6
- Android Development Tools Plug-in (ADT)

Για την ανάπτυξη της υπηρεσίας μας χρησιμοποιήσαμε το SDK που παρέχεται από την ιστοσελίδα της Google. Το SDK περιέχει:

- Το Eclipse IDE(4.2) και το plug-in ADT για την ενσωμάτωση Android λειτουργιών στο Eclipse
- Τα εργαλεία του Android SDK
- Τα εργαλεία της πλατφόρμας
- Την τελευταία έκδοση του Android (4.4.2)
- Την τελευταία εικόνα για το σύστημα Android στον εξομοιωτή (emulator)
- JDK 6

Το παραπάνω SDK εγκαταστάθηκε σε υπολογιστή με λειτουργικό Windows 7 και επομένως υπερκάλυπτε τις απαιτήσεις που έχουν τεθεί από την Google.

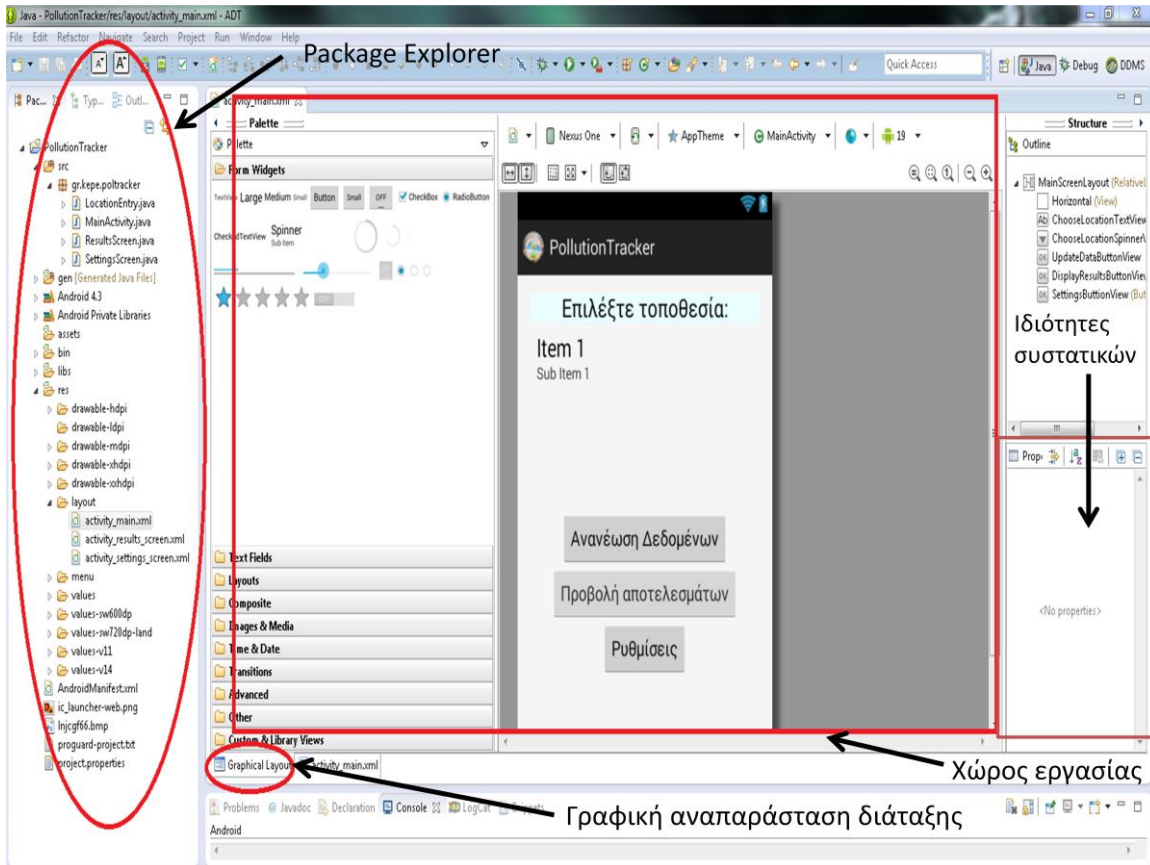
Αρχικά δημιουργήσαμε ένα δοκιμαστικό Project προκειμένου να εξοικειωθούμε με τον τρόπο λειτουργίας του IDE. Στα πλαίσια της δοκιμής φτιάξαμε μια δοκιμαστική οθόνη διάταξης (layout) η οποία απεικόνιζε μία ετικέτα. Η ετικέτα τοποθετήθηκε με την διαδικασία drag & drop από τα Form Widgets που μας παρέχει το Eclipse όπως φαίνεται παρακάτω (σχήμα 20).



Σχήμα 20 Δοκιμαστικό Project

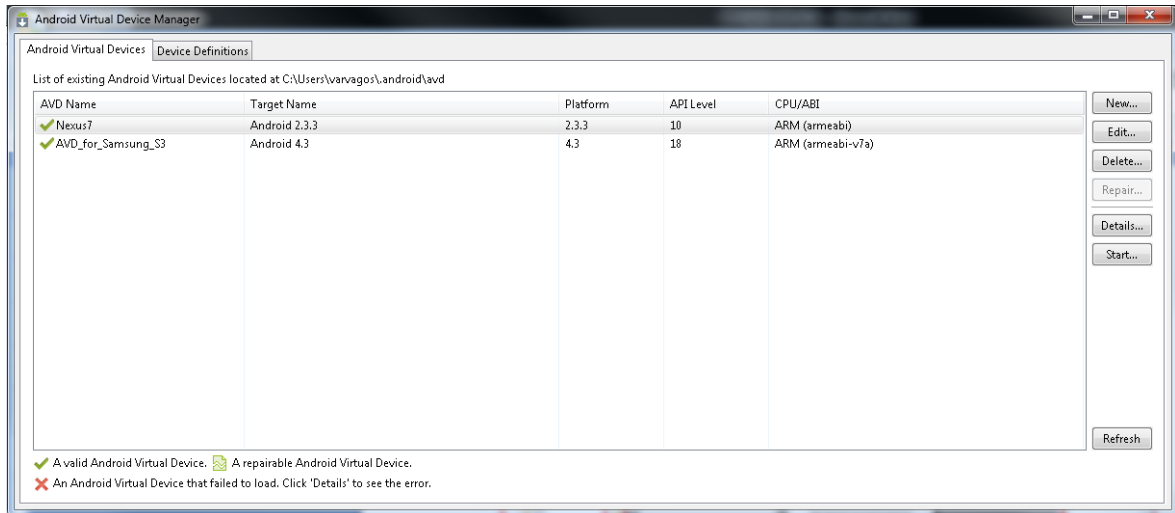
Στην συνέχεια ξεκινήσαμε να αναπτύσσουμε την υπηρεσία μας σύμφωνα με την ανάλυση την οποία κάναμε στο προηγούμενο κεφάλαιο. Το Eclipse IDE ως περιβάλλον ανάπτυξης μας δίνει την δυνατότητα να διαχειριζόμαστε τα αρχεία τα οποία δημιουργούμε στον Package Explorer. Πρόκειται για την περιοχή αριστερά από το χώρο εργασίας (workspace). Εκεί δημιουργήσαμε τα αρχεία των κλάσεων μας καθώς και των διατάξεων (layouts). Επιλέγοντας μια κλάση από τον package explorer ξεκινήσαμε την σύνταξη των κλάσεων μας.

Με τον ίδιο τρόπο έγινε και η σύνταξη των διατάξεων με την διαφορά όμως ότι χρησιμοποιήσαμε την τεχνική drag & drop. Το Eclipse IDE μας παρείχε δύο επιλογές είτε να συντάξουμε σε XML την οθόνη διάταξης ή με την τεχνική drag & drop να δημιουργήσουμε γρήγορα και αποτελεσματικά τις οθόνες μας τοποθετώντας συστατικά στην εικονική διεπαφή και να την παραμετροποιήσουμε απευθείας με ιδιότητες που παρέχονται από το Eclipse IDE.



Σχήμα 21 Eclipse IDE

Επίσης χρησιμοποιήσαμε την δυνατότητα της δημιουργίας εξομοιωτή προκειμένου να μπορέσουμε να δούμε στην πράξη ότι η υπηρεσία μας λειτουργεί και εμφανίζεται σωστά σε μια «εικονική» συσκευή Android. Για να επιτευχθεί αυτό επιλέξαμε την επιλογή Android Virtual Device Manager όπου εκεί υπήρχαν δύο επιλογές, είτε να δημιουργήσουμε εξολοκλήρου μια εικονική συσκευή εξ αρχής είτε μέσω προτύπων συσκευής που μας διαθέτει το Eclipse.



Σχήμα 22 Android Virtual Device Manager

Δημιουργήσαμε δύο εξομοιωτές, ο πρώτος με την ονομασία Nexus 7 ο οποίος χρησιμοποιήθηκε προκειμένου να δοκιμάσουμε πώς δημιουργείται ένας εξομοιωτής καθώς και για να εκτελέσουμε την δοκιμαστική εφαρμογή την οποία προαναφέραμε. Ο δεύτερος δημιουργήθηκε για την ανάγκη της εφαρμογής μας με την ονομασία AVD_for_Samsung_S3 αναπαριστώντας την ομώνυμη συσκευή.

5.3 ΛΕΙΤΟΥΡΓΙΑ ΥΠΗΡΕΣΙΑΣ POLLUTIONTRACKER

Η υπηρεσία την οποία δημιουργήσαμε λειτουργεί για συσκευές με έκδοση του Android από 4.0 (API 14) και νεώτερη (API 19). Ο εξομοιωτής ο οποίος χρησιμοποιήθηκε προκειμένου να δοκιμαστεί η υπηρεσία μας έχει τα παρακάτω χαρακτηριστικά:

- Ανάλυση Οθόνης: 720 X 1280 (υψηλή ανάλυση)
- Λογισμικό: Android 4.3 – API Level 18
- RAM: 512 MB
- Εσωτερική μνήμη: 200 MB

ΕΠΙΛΟΓΟΣ

Το κεφάλαιο που μόλις τελείωσε παρουσίασε το τεχνολογικό περιβάλλον ανάπτυξης της υπηρεσίας μας, κάνοντας λόγο για τα ολοκληρωμένα περιβάλλοντα ανάπτυξης που μας παρέχει η Google καθώς και ποιο από αυτά επιλέξαμε. Επίσης έγινε λόγος από ποια έκδοση και νεώτερη λειτουργεί η υπηρεσία μας.

Στην συνέχεια θα περιγράψουμε την υπηρεσία μας παρουσιάζοντας τις λειτουργίες της καθώς και τις αντίστοιχες οθόνες έτσι όπως αυτές είναι σχεδιασμένες ή διαμορφώνονται από την περιήγηση του χρήστη.

ΚΕΦΑΛΑΙΟ 6

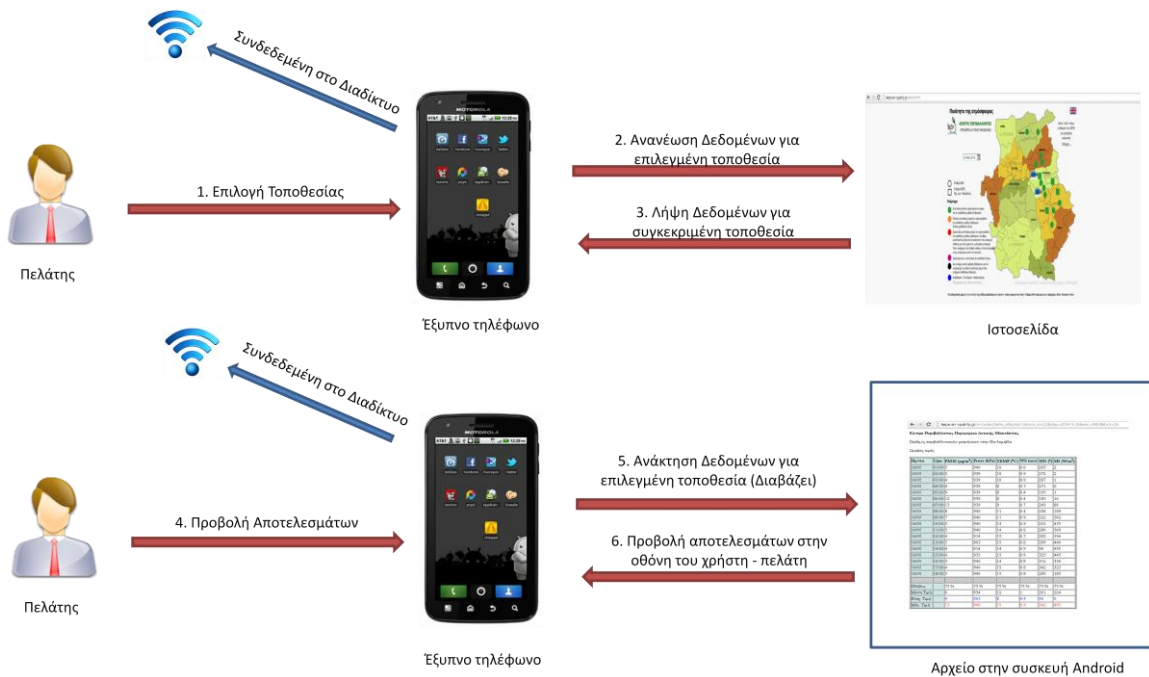
ΠΕΡΙΓΡΑΦΗ ΥΠΗΡΕΣΙΑΣ

ΕΙΣΑΓΩΓΗ

Έχοντας αναλύσει την υπηρεσία μας και το τεχνολογικό περιβάλλον στο οποίο αναπτύχθηκε, στο παρόν κεφάλαιο θα παρουσιάσουμε την αρχιτεκτονική της υπηρεσίας PollutionTracker καθώς και πως ακριβώς λειτουργεί.

6.1 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΥΠΗΡΕΣΙΑΣ

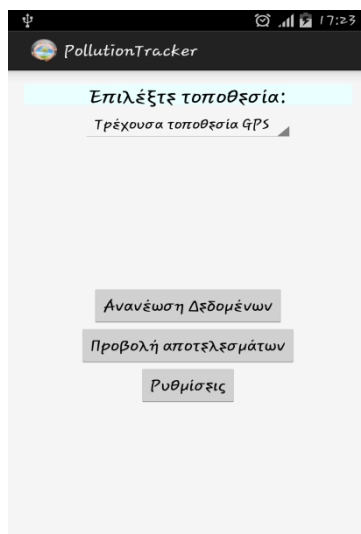
Προαναφέραμε την δυσκολία την οποία είχαμε για την λήψη δεδομένων μόλυνσης για να χρησιμοποιήσουμε στην υπηρεσία μας (κεφάλαιο 2). Για την επίλυση αυτής της δυσκολίας και για να γίνει δοκιμή της υπηρεσίας μας σε δεδομένα που αφορούν την Πτολεμαΐδα έπρεπε να διαβάσουμε τα δημόσια δεδομένα που μας δίνονται από την ιστοσελίδα του ΚΕΠΕ. Οι τέσσερις τοποθεσίες τις οποίες προβάλλει η υπηρεσία μας είναι οι τοποθεσίες της Ακρινής, της Κοζάνης, του Καρυχωρίου και της Πτολεμαΐδας. Το «έξυπνο» τηλέφωνο χρησιμοποιώντας την εφαρμογή και έχοντας πρόσβαση στο Διαδίκτυο (Internet) κατεβάζει την ιστοσελίδα για κάθε τοποθεσία και στην συνέχεια προκειμένου να προβάλλει αποτελέσματα διαβάζει (read) την ιστοσελίδα (parse) και γράφει (write) σε ένα αρχείο τα δεδομένα. Το αρχείο βρίσκεται στη μνήμη της συσκευής και από εκεί ανακτούνται οι πληροφορίες προκειμένου ο χρήστης να γνωρίζει για την τοποθεσία που επιλέγει τις τιμές μόλυνσης.



Σχήμα 23 Αρχιτεκτονική Υπηρεσίας PollutionTracker

6.2 ΤΡΟΠΟΣ ΛΕΙΤΟΥΡΓΙΑΣ ΥΠΗΡΕΣΙΑΣ

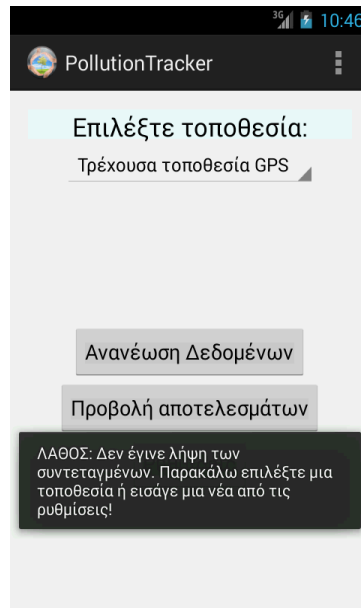
Ο χρήστης ανοίγοντας την εφαρμογή PollutionTracker στο «έξυπνο» τηλέφωνο πλατφόρμας Android βλέπει την παρακάτω οθόνη:



Σχήμα 24 Αρχική Οθόνη

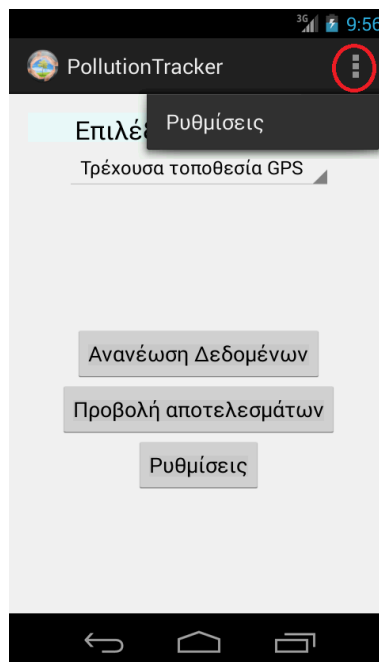
Αρχικά υπάρχει η ετικέτα «Επιλέξτε τοποθεσία:» και από κάτω ένα Spinner στο οποίο βρίσκονται διαθέσιμες όλες οι τοποθεσίες για τις οποίες ο χρήστης μπορεί να μάθει πληροφορίες. Επίσης είναι διαθέσιμα τρία κουμπιά, «Ανανέωση Δεδομένων», «Προβολή αποτελεσμάτων» και «Ρυθμίσεις».

Το κουμπί «Ανανέωση Δεδομένων» όταν πατηθεί από τον χρήστη η εφαρμογή ανανεώνει τα δεδομένα από την ιστοσελίδα του ΚΕΠΕ. Για να επιτευχτεί η ανανέωση των δεδομένων θα πρέπει το «έξυπνο» τηλέφωνο να έχει πρόσβαση στο Διαδίκτυο και ο χρήστης κατά την διαδικασία εισαγωγής των ρυθμίσεων που θα δούμε παρακάτω να έχει πληκτρολογήσει σωστά την διεύθυνση της τοποθεσίας όπου βρίσκονται τα δεδομένα μόλυνσης. Σε διαφορετική περίπτωση η εφαρμογή εμφανίζει κατάλληλο μήνυμα (Σχήμα 25)



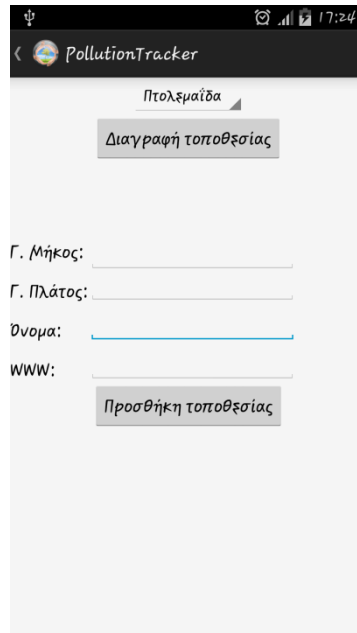
Σχήμα 25 Μήνυμα μη ανανέωσης δεδομένων

Πριν αναφερθούμε στην προβολή αποτελεσμάτων στην υπηρεσία μας θα παρουσιάσουμε τον τρόπο με τον οποίο ο χρήστης μπορεί να προσθέσει μια νέα τοποθεσία στην εφαρμογή μας. Ο χρήστης θα πρέπει να επιλέξει το κουμπί «Ρυθμίσεις». Εναλλακτικά μπορεί να επιλέξει το κουμπί που βρίσκεται πάνω δεξιά στην οθόνη (σχήμα 26) και να του εμφανιστεί ένα drop down μενού όπου μπορεί να επιλέξει ρυθμίσεις.



Σχήμα 26 Εναλλακτικός τρόπος μεταφοράς στις Ρυθμίσεις

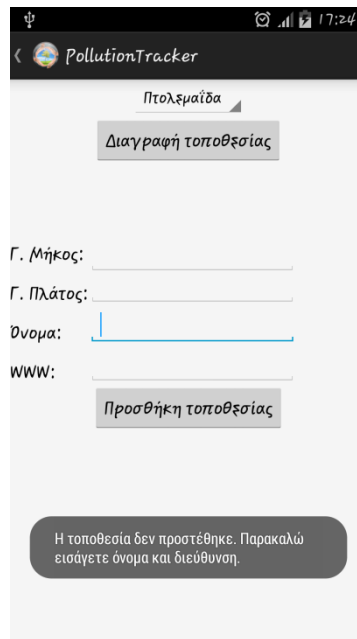
Όποιον από τους δύο τρόπους επιλέξει ο χρήστης μετά από αυτήν την επιλογή θα μεταφερθεί στην οθόνη του σχήματος 27.



Σχήμα 27 Οθόνη Ρυθμίσεων

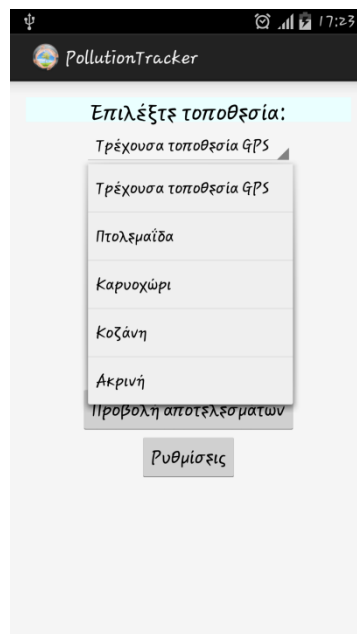
Στην συγκεκριμένη οθόνη ο χρήστης έχει δύο επιλογές, είτε να διαγράψει μια υπάρχουσα τοποθεσία που έχει εισάγει παλιότερα είτε να εισάγει τα στοιχεία μιας νέας. Για την νεοεισερχόμενη τοποθεσία ο χρήστης μπορεί να εισάγει το γεωγραφικό μήκος, το γεωγραφικό πλάτος, το όνομα και την διεύθυνση Παγκόσμιου Ιστού. Το όνομα και η διεύθυνση είναι υποχρεωτικά πεδία, σε περίπτωση που ο χρήστης δεν τα εισάγει η τοποθεσία δεν θα προστεθεί στο Spinner και θα εμφανιστεί το μήνυμα «Η τοποθεσία δεν προστέθηκε. Παρακαλώ εισάγετε όνομα και διεύθυνση» (Σχήμα 28).

Πτυχιακή εργασία του φοιτητή Γρηγορίου Χρήστου



Σχήμα 28 Οθόνη Ρυθμίσεων - Μη εισαγωγή στοιχείων

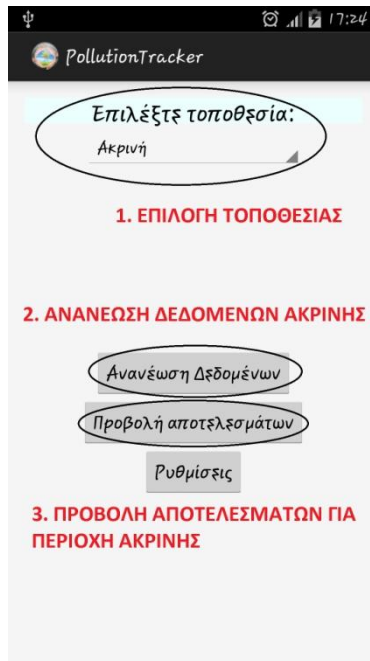
Μετά την επιτυχημένη εισαγωγή τοποθεσίας ο χρήστης επιστρέφοντας στην κύρια οθόνη της εφαρμογής στο Spinner μπορεί να επιλέξει μια από τις τοποθεσίες για τις οποίες θέλει να προβάλει τα αποτελέσματα που είναι διαθέσιμα μέχρι εκείνη την στιγμή (Σχήμα 29).



Σχήμα 29 Αρχική Οθόνη - Επιλογή Τοποθεσίας

Έπειτα προκειμένου ο χρήστης να προβάλει αποτελέσματα θα πρέπει (σχήμα 30):

- να επιλέξει μια τοποθεσία (πχ Ακρινή)
- να επιλέξει το κουμπί «Ανανέωση Δεδομένων»
- να επιλέξει το κουμπί «Προβολή αποτελεσμάτων»



Σχήμα 30 Αρχική Οθόνη - Διαδικασία προβολής τοποθεσίας

Μετά από αυτή την διαδικασία ο χρήστης θα έχει στην οθόνη του «έξυπνου» τηλεφώνου την οθόνη των αποτελεσμάτων της υπηρεσίας PollutionTracker όπως βλέπουμε στο παρακάτω σχήμα.

	PM10	PM2.5
Τρέχων:	4	2
Μέση Τιμή:	8.81	3.38
Απόκλιση:	5.01	1.11
Σκέβρωση:	0.54	0.03
Κύρτωση:	2.08	1.63
Μέγιστη Τιμή:	19	5
Ελάχιστη Τιμή:	3	2

Σχήμα 31 Αποτελεσμάτων - Τοποθεσία "Ακρινή"

Η οθόνη αποτελεσμάτων περιέχει αρχικά το όνομα της τοποθεσίας και στην συνέχεια έναν πίνακα με τρεις στήλες. Στην πρώτη στήλη υπάρχουν τα είδη των μετρήσεων τα οποία διαβάζει η εφαρμογή από το αρχείο, στην δεύτερη στήλη οι τιμές για τις συγκεντρώσεις ρύπων τύπου PM_{10} και στην Τρίτη στήλη οι τιμές για συγκεντρώσεις ρύπων τύπου $PM_{2.5}$. Τα είδη των μετρήσεων που καλύπτει η εφαρμογή μας είναι η

τρέχουσα τιμή, η μέση τιμή, η τιμή απόκλισης, η τιμή σκέβρωσης και κύρτωσης καθώς και η μέγιστη και ελάχιστη τιμή των ρύπων PM_{10} και $PM_{2.5}$ που ισχύουν την τρέχουσα χρονική στιγμή για την συγκεκριμένη τοποθεσία που έχει επιλεγεί.

ΕΠΙΛΟΓΟΣ

Έχοντας μιλήσει για την αρχιτεκτονική καθώς και για τον τρόπο λειτουργίας της υπηρεσίας μας, φτάνοντας στο τέλος της διπλωματικής μας εργασίας στο επόμενο κεφάλαιο θα αναφερθούμε σε συμπεράσματα στα οποία καταλήξαμε αναφορικά με την ενασχόληση μας με τα «έξυπνα» τηλέφωνα, με την πλατφόρμα Android καθώς και με την ανάπτυξη της εφαρμογής μας. Επιπλέον θα προτείνουμε μελλοντικές επεκτάσεις τόσο για την υπηρεσία PollutionTracker όσο και για νέες εφαρμογές γύρω από τις εφαρμογές με βάση την τοποθεσία.

ΚΕΦΑΛΑΙΟ 7

ΚΑΙΝΟΤΟΜΙΑ ΚΑΙ ΠΡΟΣΤΙΘΕΜΕΝΗ ΑΞΙΑ ΥΠΗΡΕΣΙΑΣ

ΕΙΣΑΓΩΓΗ

Ο τομέας των υπηρεσιών προσφέρει σημαντικές ευκαιρίες καινοτομίας, οι οποίες πρέπει να αξιοποιηθούν. Η καινοτομία στις υπηρεσίες μπορεί να αφορά νέες τεχνολογίες, διεργασίες, τρόπους αλληλεπίδρασης με τον χρήστη, αλλά και νέα επιχειρηματικά μοντέλα και πρακτικές.

Η ανάπτυξη των Τεχνολογιών Πληροφορικής και Επικοινωνιών και του διαδικτύου τα τελευταία χρόνια επιτρέπει την αυτοματοποίηση ορισμένων υπηρεσιών, και την ευρεία διάθεσή τους. Έτσι, δίνεται η δυνατότητα για την ανάπτυξη νέων καινοτόμων υπηρεσιών προστιθέμενης αξίας, οι οποίες αναδεικνύονται ως μια ευκαιρία-κλειδί για ανάπτυξη της επιχειρηματικότητας, της αγοράς, και της ανταγωνιστικότητας στην Ευρώπη. Στην Ελλάδα τα τελευταία χρόνια οι κλάδοι υψηλής τεχνολογίας είχαν να παρουσιάσουν μια ταχεία ανάπτυξη. Πάνω από 26.000 επιχειρήσεις δραστηριοποιούνται σε κλάδους υψηλής τεχνολογίας με την πλειονότητα αυτών να έχει ιδρυθεί την τελευταία 15ετία, ενώ και η σχετική απασχόληση παρουσίαζε σημαντική αύξηση (80.000 εργαζόμενοι την περίοδο 2001 – 2006). Οι κλάδοι αυτοί μπορούν να αποτελέσουν σημαντικές πηγές ανάπτυξης συνολικά για την ελληνική οικονομία. Αποτελούν τους πρωτοπόρους κλάδους σε όρους καινοτομίας, και επιπλέον συνεισφέρουν στην καινοτομική επίδοση των υπόλοιπων κλάδων της οικονομίας[1].

Παρακάτω θα μελετήσουμε τα σημεία καινοτομίας της υπηρεσίας PollutionTracker που αναπτύξαμε καθώς και τα νέα χαρακτηριστικά τα οποία εισάγει έναντι άλλων υπηρεσιών.

7.1 ΣΗΜΕΙΑ ΚΑΙΝΟΤΟΜΙΑΣ

Η υπηρεσία την οποία αναπτύξαμε έρχεται να χρησιμοποιήσει την τεχνολογία προκειμένου να εξυπηρετήσει καλύτερα τους χρήστες. Το κυριότερο σημείο που κάνει την υπηρεσία μας καινοτόμα είναι ότι εκμεταλλεύεται την ήδη εγκατεστημένη τεχνολογία μέτρησης της ποιότητας του αέρα για να παρέχει πληροφορίες στα «έξυπνα» τηλέφωνα. Για γεωγραφικές περιοχές της Ελλάδας δεν κατέστη δυνατό να βρεθεί ανάλογη υπηρεσία επομένως πρόκειται για μια καινοτομία τουλάχιστον για τα ελληνικά δεδομένα καθώς διαφοροποιείται ως προς το είδος της παρεχόμενης πληροφορίας που παρέχει.

Επίσης αξιοποιεί νέες τεχνολογίες για την άντληση αυτών των δεδομένων και τον διαμοιρασμό τους στα «έξυπνα» τηλέφωνα, μειώνει τις περιβαλλοντικές επιπτώσεις που μπορεί να έχει ένας χρήστης στην ζωή του από την μόλυνση της ατμόσφαιρας

παρέχοντας στην ουσία μια κοινωνική υπηρεσία. Αντιλαμβανόμαστε ότι πρόκειται για μια υπηρεσία - προϊόν που ακολουθεί τις νέες τεχνολογικές εξελίξεις .

7.2 ΠΡΟΣΤΙΘΕΜΕΝΗ ΑΞΙΑ ΕΝΑΝΤΙ ΑΛΛΩΝ ΥΠΗΡΕΣΙΩΝ

Οι ανάλογες υπηρεσίες που μελετήσαμε στο πλαίσιο της εργασίας μας αποτελούσαν λύσεις για περιοχές πέρα από την Ελλάδα. Δημιουργήσαμε την υπηρεσία αυτή και για την Ελλάδα δίνοντας δυνατότητες στον χρήστη που ξεπερνούν ανάλογες υπηρεσίες με βάση την γεωγραφική θέση του χρήστη.

Η υπηρεσία μας υπερτερεί στο ότι είναι διαθέσιμη στα «έξυπνα» τηλέφωνα και ο χρήστης μπορεί να την χρησιμοποιεί οπουδήποτε. Επίσης ο πολίτης μπορεί να ενημερώνεται ανά πάσα στιγμή για νέες μετρήσεις σε σχέση με μια άλλη υπηρεσία η οποία θα παρέχει τις ίδιες πληροφορίες μέσω ραδιοφώνου ή τηλεόρασης.

Από την άλλη μεριά η μεγαλύτερη δυσκολία της είναι η λήψη των δεδομένων καθώς η απαραίτητη ύπαρξη Wi-Fi δεν προσφέρει πανελλαδική κάλυψη σε σχέση με ένα ραδιόφωνο ή μιας τηλεόρασης.

ΣΥΜΠΕΡΑΣΜΑΤΑ – ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

Η εργασία αυτή μας βοήθησε να αντιληφθούμε ότι η ύπαρξη των «έξυπνων» τηλεφώνων αποτελεί ίσως μια από τις μεγαλύτερες τεχνολογικές καινοτομίες της εποχής μας. Η εδραίωση τους στην αγορά δημιουργεί νέες ανάγκες και απαιτήσεις.

Αρχικά, στο πλαίσιο της εργασίας παρατηρήσαμε τον ρόλο των εφαρμογών στα «έξυπνα» τηλέφωνα και τις καινοτόμες υπηρεσίες που προσφέρουν. Πραγματοποιήθηκε μελέτη του προσωποποιημένου περιεχομένου αναδεικνύοντας την σημαντικότητα του και τα οφέλη που παρέχουν τόσο στον χρήστη όσο και στην επιχείρηση.

Επίσης η ενασχόληση μας με μελέτες περίπτωσης για την αντιμετώπιση της ατμοσφαιρικής ρύπανσης μέσω εφαρμογών για «έξυπνα» τηλέφωνα μας έδειξε ότι οι υπηρεσίες αυτές βρίσκονται σε αρχικό στάδιο και πρέπει να γίνουν σημαντικά βήματα προς αυτή την κατεύθυνση. Για να γίνει αυτό είναι η ύπαρξη σωστών τεχνολογιών για την υποστήριξη τους όπως είναι τα Web Service.

Στην συνέχεια εξετάζοντας την γλώσσα μοντελοποίησης UML παρατηρήσαμε ότι αποτελεί έναν εύχρηστο τρόπο για την ανάλυση μιας υπό δημιουργία υπηρεσίας, δίνοντας στον δημιουργό της ευκαιρία με διαγραμματικές τεχνικές και μοντέλα να την υλοποιήσει. Η Google από την μεριά της παρέχει όπως είδαμε δύο τεχνολογικά περιβάλλοντα ανάπτυξης. Η λύση του Android Studio δεν είναι όμως σε θέση να μας προσφέρει όλα αυτά που χρειάζεται ένας προγραμματιστής σε περιβάλλον Android σε σχέση με το Eclipse IDE το οποίο προϋπήρχε και προσαρμόστηκε στις ανάγκες της ανάπτυξης εφαρμογών Android.

Σχετικά με την υπηρεσία που δημιουργήσαμε οι δυσκολίες που αντιμετωπίσαμε σχετίζονται με την έλλειψη εξόρυξης δεδομένων μόλυνσης κυρίως για την περιοχή της Δυτικής Μακεδονίας την οποία μελετήσαμε. Μελλοντικές επεκτάσεις τις εφαρμογής μας θα μπορούσε να ήταν η ορθή εύρεση δεδομένων μέσω υπηρεσιών Web Service και η δυνατότητα στον χρήστη να εντοπίζει το βαθμό μόλυνσης της ατμόσφαιρας του αέρα με την χρήση του GPS βασιζόμενης στην τρέχουσα θέση του. Επίσης θα μπορούσε η εν λόγω υπηρεσία να εξελιχθεί για να καλύψει και άλλους τομείς υπηρεσιών με βάση την τοποθεσία του χρήστη. Εφικτό θα ήταν με μικρές μετατροπές να παρέχονται πληροφορίες στον χρήστη αν μία περιοχή είναι σεισμογενής ή όχι καθώς και ακριβείς ενδείξεις σχετικά με την ποσότητα νερού βροχής που πρόκειται να πέσουν σε μία περιοχή. Επίσης θα μπορούσε μια τέτοιου είδους υπηρεσία να ενημερώνει τους χρήστες πόσοι γιατροί υπάρχουν γύρω του και τι ειδικότητα έχουν κάτι το οποίο θα ήταν πολύ χρήσιμο για άτομα με προβλήματα υγείας.

Κλείνοντας μπορούμε να πούμε ότι οι υπηρεσίες στα «έξυπνα» τηλέφωνα με βάση την τοποθεσία έχουν μεγάλες δυνατότητες ανάπτυξης και τα επόμενα χρόνια θα διαδραματίσουν κυρίαρχο ρόλο στην ζωή μας.

ΠΑΡΑΡΤΗΜΑΤΑ

ΠΑΡΑΡΤΗΜΑ Α

Στο παράρτημα Α που ακολουθεί παρουσιάζεται αναλυτικά ο κώδικας για κάθε κλάση την οποία χρησιμοποιήσαμε προκειμένου να επιτευχθεί η ανάπτυξη της υπηρεσίας μας. Υπάρχουν επεξηγήσεις σε σημαντικά σημεία τα οποία και επεξηγούν τις λειτουργίες της εφαρμογής τις οποίες αναφέραμε σε προηγούμενο κεφάλαιο. Επίσης πριν από την παράθεση του κώδικα της κάθε κλάσης υπάρχει μια σύντομη περιγραφή της λειτουργίας την οποία επιτελεί.

LocationEntry.java

```
/*
 * Κλάση η οποία περιγραφεί την οντότητα μιας τοποθεσίας (Location).
 * Περιέχει ιδιότητες που περιγράφουν μια τοποθεσία, τους κατασκευαστές
 * για την δημιουργία ενός αντικειμένου LocationEntry και μέθοδο για το
 * εάν δύο αντικείμενα αυτής της κλάσης είναι ίσα. Η κλάση χρησιμοποιείται
 * για την προσθήκη μιας τοποθεσίας στην υπηρεσία μας. Η κλάση υλοποιεί
 * το interface Serializable προκειμένου το στιγμιότυπο αυτής της κλάσης
 * όταν δημιουργείται να μπορεί να γραφεί και να αποθηκευτεί σε αρχείο.
 */

package gr.kepe.poltracker;

import java.io.Serializable;

public class LocationEntry implements Serializable
{
    // Οι ιδιότητες μιας Location (Τοποθεσίας)
    private static final long serialVersionUID = 1682418825693889002L;
    public double latitude; // γεωγραφικό πλάτος
    public double longitude; // γεωγραφικό μήκος
    public String name; // όνομα τοποθεσίας
    public String address; // διεύθυνση (www) τοποθεσίας
    public boolean hasGeoTag = false; // εάν δόθηκαν συντεταγμένες

    // Κατασκευαστής LocationEntry με ορίσματα (γεωγ. μήκος, γεωγ. πλάτος,
    όνομα, διεύθυνση)
    public LocationEntry(double getLatitude, double getLongitude, String
    getName, String getAddress)
    {
        name = getName;
        address = getAddress;
        latitude = getLatitude;
        longitude = getLongitude;
        hasGeoTag = true;
    }

    // Κατασκευαστής LocationEntry με ορίσματα (γεωγ. μήκος, γεωγ. πλάτος,
    όνομα, διεύθυνση)
    public LocationEntry(String getName, String getAddress)
    {
        name = getName;
    }
}
```

```
        address = getAddress();
        hasGeoTag = false;
    }

    // Μέθοδος που επιστρέφει το hash code για το όνομα της τοποθεσίας
    public int hashCode()
    {
        return name.hashCode();
    }

    // Μέθοδος που επιστρέφει εάν δύο αντικείμενα είναι ίσα
    public boolean equals(Object object)
    {
        if(object.hashCode() == this.hashCode())
            return true;
        else
            return false;
    }
}
```

MainActivity.java

```
/*
 * Κλάση η οποία καλείται στο ξεκίνημα της εφαρμογής μας. Φορτώνει την διάταξη
 * (Layout) activity_main.xml και τις διευθύνσεις των προκαθορισμένες τοποθεσιών.
 * Κατεβάζει την ιστοσελίδα η οποία περιέχει τα δεδομένα μόλυνσης, τα διαβάζει
 * και στην συνέχεια μπορεί να τα προβάλλει. Γενικά, χειρίζεται τα γεγονότα της
 * επιλογής τοποθεσίας, της ανανέωσης των δεδομένων, της προβολής τους, της
 * μεταφοράς στις ρυθμίσεις της εφαρμογής, ζητήματα διαχείρισης μνήμης για την
 * ίδια την κλάση καθώς και το μενού επιλογών.
 */
```

```
package gr.kepe.poltracker;

import java.io.BufferedReader;
import java.io.FileInputStream;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.ObjectInputStream;
import java.util.ArrayList;
import java.util.HashSet;
import java.util.Iterator;
import java.util.List;

import org.apache.http.HttpResponse;
import org.apache.http.client.HttpClient;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.impl.client.DefaultHttpClient;

import android.app.Activity;
import android.content.Context;
import android.content.Intent;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import android.os.StrictMode;
import android.view.Menu;
```

```

import android.view.View;
import android.widget.AdapterView;
import android.widget.Spinner;
import android.widget.Toast;

public class MainActivity extends Activity
{
    private Spinner locationSpinner; // Αναφορά στο Spinner View για την
    επιλογής τοποθεσίας
    public HashSet<LocationEntry> locations; // Το σύνολο αυτό κρατά τις
    αποθηκευμένες τοποθεσίες
    public static Activity instance; // Αναφορά στην οθόνη του activity
    (activity screen)
    public String dataSource; // Το τρέχων επιλεγμένο όνομα τοποθεσίας
    public String dataHTML; // Η HTML ιστοσελίδα που περιγράφει την μόλυνση
    στην τρέχουσα τοποθεσία

    private LocationManager locationManager;
    private LocationListener locationListener;
    public ArrayList<LocationEntry> defaultLocations = new
    ArrayList<LocationEntry>();

    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main); // Φόρτωση της διάταξης
        activity_main.xml
        instance = this;
        this.locationManager = (LocationManager)
        this.getSystemService(Context.LOCATION_SERVICE);
        this.locationListener = new LocationListener()
        {
            public void onLocationChanged(Location location){}

            public void onStatusChanged(String provider, int status,
            Bundle extras) {}

            public void onProviderEnabled(String provider) {}

            public void onProviderDisabled(String provider) {}
        };

        this.locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER,
        0, 0, locationListener);
        StrictMode.ThreadPolicy policy = new
        StrictMode.ThreadPolicy.Builder().permitAll().build();
        StrictMode.setThreadPolicy(policy); // επιτρέπει την πρόσβαση του
        δικτύου στο main thread της εφαρμογής μας

        //Εισαγωγή προκαθορισμένων τοποθεσιών
        defaultLocations.add(new LocationEntry(40.434078, 21.904922,
        "Ακρινή", "http://kepe.air-
        quality.gr/includes/table_allk.php?station_id=15&day=2014-5-
        9&sec=3472&bot=11"));
        defaultLocations.add(new LocationEntry(40.299167, 21.789951,
        "Κοζάνη", "http://kepe.air-
        quality.gr/includes/table_allk.php?station_id=14&day=2014-5-
        9&sec=3472&bot=12"));
        defaultLocations.add(new LocationEntry(40.503162, 21.776433,
        "Καρυχώρι", "http://kepe.air-

```

```

quality.gr/includes/table_allk.php?station_id=16&day=2014-5-
9&sec=3472&bot=14"));
        defaultLocations.add(new LocationEntry(40.513016, 21.679058,
"Πτολεμαΐδα", "http://kepe.air-
quality.gr/includes/table_allk.php?station_id=12&day=2014-5-
9&sec=3472&bot=16"));
    }

    /* Λήψη της ιστοσελίδας σύμφωνα με το URL που δίνεται όταν καλείται η
    * μέθοδος.
    * Επιστρέφει την ιστοσελίδα σε μορφή String.
    */
    private String getWebPage(String url)
    {
        String html = null;
        HttpClient client;
       HttpGet request;
        HttpResponse response;
        try
        {
            client = new DefaultHttpClient();
            request = new HttpGet(url);
            response = client.execute(request);
        }
        catch(Exception ex)
        {
            Toast.makeText(this, R.string.UnreachableWebSiteWarningText,
Toast.LENGTH_LONG).show();
            return null;
        }
        try
        {
            InputStream in = response.getEntity().getContent();
            BufferedReader reader = new BufferedReader(new
InputStreamReader(in));
            StringBuilder str = new StringBuilder();
            String line = null;
            while((line = reader.readLine()) != null)
            {
                str.append(line);
            }
            in.close();
            html = str.toString();
        }
        catch (Exception ex)
        {
            Toast.makeText(this, R.string.UnreadableWebSiteWarningText,
Toast.LENGTH_LONG).show();
            return null;
        }
        return html;
    }

    /* Μέθοδος η οποία καλείται όταν επιστρέφουμε στο activity από κάποια
    * άλλη activity ή μετά από παύση της εφαρμογής (πχ. SettingsScreen,
    * ResultsScreen)
    * */
    @SuppressWarnings("unchecked")
    protected void onResume()
    {

```

```

//Προσθήκη του προκαθορισμένου στοιχείου "Current GPS" στην λίστα
με τις τοποθεσίες
super.onResume();
locationSpinner = (Spinner)
findViewById(R.id.ChooseLocationSpinnerView);
List<String> choices = new ArrayList<String>();
choices.add((String)
getResources().getString(R.string.GPSSpinnerItem));

//Προσθήκη των αποθηκευμένων τοποθεσιών στην λίστα με τις
τοποθεσίες (διάβασμα από το αρχείο)
ObjectInputStream ois;
try
{
    ois = new ObjectInputStream(new
FileInputStream(this.getDir("RAW", 0).getPath().toString() +
(String)getResources().getString(R.string.LocationsFilename)));
    locations = (HashSet<LocationEntry>) ois.readObject();
    for(int i=0; i<defaultLocations.size(); i++)
    {
        locations.add(defaultLocations.get(i));
    }
    Iterator<LocationEntry> iter = locations.iterator();
    while(iter.hasNext())
    {
        choices.add(iter.next().name);
    }
}
catch (Exception e)
{
    Toast.makeText(this, R.string.NoDataWarningText,
Toast.LENGTH_LONG).show();
    locations = new HashSet<LocationEntry>();

    /* Αρχικοποίηση του Spinner View που επιτρέπει την επιλογή
    * τοποθεσιών με την λίστα των τοποθεσιών
    */
    ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
android.R.layout.simple_spinner_item, choices);

    adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_
item);
    locationSpinner.setAdapter(adapter);
}

/* Μέθοδος που καλείται όταν ο χρήστης επιλέξει το κουμπί "Ανανέωση
* Δεδομένων" και ανανεώνει τα δεδομένα για την συγκεκριμένη τοποθεσία σε
* περίπτωση που αυτό δεν γίνει εμφανίζονται κατάλληλα μηνύματα
* */
public void onClickUpdate(View button)
{
    String www = null;
    String selectedLocation =
(String)locationSpinner.getItemAtPosition(locationSpinner.getSelectedItemPositi
on()); // επιλεγμένη τοποθεσία
    if(selectedLocation.equals((String)
getResources().getString(R.string.GPSSpinnerItem)))
    {

```

Πτυχιακή εργασία του φοιτητή Γρηγορίου Χρήστου

```
Location lastKnownLocation =
locationManager.getLastKnownLocation(LocationManager.GPS_PROVIDER); //τελευταία
τοποθεσία σύμφωνα με το GPS
    if(lastKnownLocation == null)
    {
        Toast.makeText(this, R.string.NoGeoDataWarningText,
Toast.LENGTH_LONG).show();
        return;
    }
    float distance[] = new float[1];
    float minDist = Float.MAX_VALUE;
    Iterator<LocationEntry> iter = locations.iterator();
    while(iter.hasNext())
    {
        LocationEntry temp = iter.next();
        if(temp.hasGeoTag == false)
            continue;

        Location.distanceBetween(lastKnownLocation.getLatitude(),
lastKnownLocation.getLongitude(), temp.latitude, temp.longitude, distance);
        if(distance[0] < minDist)
        {
            minDist = distance[0];
            www = temp.address;
            selectedLocation = temp.name;
        }
    }
    if(www != null)
    {
        dataSource = selectedLocation;
        dataHTML = getWebPage(www);
    }
    else
    {
        Toast.makeText(this, R.string.NoGeoTagWarningText,
Toast.LENGTH_LONG).show();
    }
}
else
{
    this.locationManager.removeUpdates(this.locationListener);
    Iterator<LocationEntry> iter = locations.iterator();
    while(iter.hasNext())
    {
        LocationEntry temp = iter.next();
        if(temp.name.equals(selectedLocation))
        {
            www = temp.address;
            break;
        }
    }
    dataSource = selectedLocation;
    dataHTML = getWebPage(www);
}
}

/* Μέθοδος που καλείται όταν ο χρήστης επιλέξει το κουμπί "Ρυθμίσεις"
 * και μεταφέρει τον χρήστη στην οθόνη των ρυθμίσεων που την χειρίζεται
 * η κλάση SettingsScreen
 * */
public void onClickSettings(View button)
```

```

    {
        Intent startSettingsScreen = new Intent(this,
SettingsScreen.class);
        startActivity(startSettingsScreen);
    }

    /* Μέθοδος που καλείται όταν ο χρήστης επιλέξει το κουμπί "Προβολή
    * αποτελεσμάτων" και μεταφέρει τον χρήστη στην οθόνη των αποτελεσμάτων
    * που την χειρίζεται η κλάση ResultsScreen
    * */
    public void onClickDisplayResults(View button)
    {
        // εάν η ιστοσελίδα που περιγράφει την μόλυνση είναι κενή βγάζει
κατάλληλο μήνυμα και σταματάει
        if(dataHTML == null)
        {
            Toast.makeText(this, R.string.NoReceivedDataWarningText,
Toast.LENGTH_LONG).show();
            return;
        }
        Intent startResultsScreen = new Intent(this, ResultsScreen.class);
        startResultsScreen.putExtra("locName", dataSource); // περνάμε σαν
παραμέτρους στην κλάση ResultScreen το όνομα της τοποθεσίας
        startResultsScreen.putExtra("locHTML", dataHTML); // περνάμε σαν
παραμέτρους στην κλάση ResultScreen τη διεύθυνση (www) της τοποθεσίας
        startActivity(startResultsScreen);
    }

    /* Μέθοδος που καλείται όταν ο χρήστης επιλέξει το κουμπί του μενού που
    * έχουν όλες οι συσκευές έξυπνων τηλεφώνων πλατφόρμας Android ή στο
    * συστατικό Action Bar
    * */
    @Override
    public boolean onCreateOptionsMenu(Menu menu)
    {
        // Εμφανίζει το μενού. Προσθέτει αυτό το μενού στο συστατικό
action bar εάν αυτό υπάρχει
        getMenuInflater().inflate(R.menu.main, menu);
        return true;
    }

    /* Μέθοδος η οποία καλείται όταν το activity καταστρέφεται
    * */
    @Override
    public void onDestroy()
    {
        super.onDestroy();
        this.locationManager.removeUpdates(this.locationListener); //
απομακρύνει όλα τα updates για τις τοποθεσίες του συγκεκριμένου
LocationListener
        this.locationManager = null;
        this.locationListener = null;
    }
}

```

ResultScreen.java

```
/*
```

Πτυχιακή εργασία του φοιτητή Γρηγορίου Χρήστου

```
* Κλάση η οποία προβάλλει τα δεδομένα μόλυνσης για την επιλεγμένη τοποθεσία.  
* Διαβάζει τα δεδομένα τα οποία έχουν ληφθεί από την εφαρμογή και  
* τα εμφανίζει σύμφωνα με την διάταξη activity_resluts_screen.xml  
*/
```

```
package gr.kepe.poltracker;  
  
import java.text.DecimalFormat;  
import java.util.ArrayList;  
  
import android.app.Activity;  
import android.os.Bundle;  
import android.widget.TextView;  
  
public class ResultsScreen extends Activity  
{  
    ArrayList<Integer> PM25 = null; // αρχικοποίηση λίστας με τις μετρήσεις  
του PM2.5  
    ArrayList<Integer> PM10 = null; // αρχικοποίηση λίστας με τις μετρήσεις  
του PM10  
    String html;  
    String location;  
    DecimalFormat df2;  
  
    @Override  
    public void onCreate(Bundle savedInstanceState)  
    {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_results_screen);  
        df2 = new DecimalFormat("###.##");  
        PM25 = new ArrayList<Integer>();  
        PM10 = new ArrayList<Integer>();  
        Bundle extras = getIntent().getExtras(); // λαμβάνουμε τις  
παραμέτρους από την MainActivity  
        html = extras.getString("locHTML");  
        location = extras.getString("locName");  
        TextView locLabel = (TextView)  
findViewById(R.id.LocationLabelTextView);  
        locLabel.setText(location);  
        parseData(); // κλήση της μεθόδου parseData()  
        if(PM25.size() > 0)  
        {  
            int[] data = new int[PM25.size()];  
            Integer[] temp = new Integer[PM25.size()];  
            temp = PM25.toArray(temp);  
            for(int i=0; i<temp.length; i++)  
                data[i] = temp[i].intValue();  
            temp = null;  
            TextView currentVal = (TextView)  
findViewById(R.id.PM2CurValueTextView);  
            currentVal.setText(String.valueOf(data[data.length-1]));  
            currentVal = (TextView)  
findViewById(R.id.PM2MeanValueTextView);  
            double mean = getMean(data);  
            currentVal.setText(String.valueOf(mean));  
            currentVal = (TextView)  
findViewById(R.id.PM2DeviationValueTextView);  
            double sdev = getSDev(data, mean);  
            currentVal.setText(String.valueOf(sdev));  
            currentVal = (TextView)  
findViewById(R.id.PM2SkewValueTextView);
```



```

        currentVal.setText(String.valueOf(getSkewness(data, mean,
sdev)));
        currentVal = (TextView)
findViewById(R.id.PM2KurtosisValueTextView);
        currentVal.setText(String.valueOf(getKurtosis(data, mean,
sdev)));
        currentVal = (TextView)
findViewById(R.id.PM2MinValueTextView);
        currentVal.setText(String.valueOf(getMin(data)));
        currentVal = (TextView)
findViewById(R.id.PM2MaxValueTextView);
        currentVal.setText(String.valueOf(getMax(data)));
    }
    if(PM10.size() > 0)
    {
        int[] data = new int[PM10.size()];
        Integer[] temp = new Integer[PM10.size()];
        temp = PM10.toArray(temp);
        for(int i=0; i<temp.length; i++)
            data[i] = temp[i].intValue();
        temp = null;
        TextView currentVal = (TextView)
findViewById(R.id.PM10CurValueTextView);
        currentVal.setText(String.valueOf(data[data.length-1]));
        currentVal = (TextView)
findViewById(R.id.PM10MeanValueTextView);
        double mean = getMean(data);
        currentVal.setText(String.valueOf(mean));
        currentVal = (TextView)
findViewById(R.id.PM10DeviationValueTextView);
        double sdev = getSDev(data, mean);
        currentVal.setText(String.valueOf(sdev));
        currentVal = (TextView)
findViewById(R.id.PM10SkewValueTextView);
        currentVal.setText(String.valueOf(getSkewness(data, mean,
sdev)));
        currentVal = (TextView)
findViewById(R.id.PM10KurtosisValueTextView);
        currentVal.setText(String.valueOf(getKurtosis(data, mean,
sdev)));
        currentVal = (TextView)
findViewById(R.id.PM10MinValueTextView);
        currentVal.setText(String.valueOf(getMin(data)));
        currentVal = (TextView)
findViewById(R.id.PM10MaxValueTextView);
        currentVal.setText(String.valueOf(getMax(data)));
    }
}

/*
 * Μέθοδος η οποία διαβάζει τα δεδομένα που περιέχει η html σελίδα της
 * επιλεγμένης τοποθεσίας.
 * */
private void parseData()
{
    int idx;
    String sub1, sub2;
    if(html.contains("PM2.5"))
    {
        for(int i=1; i<10; i++)
        {

```

Πτυχιακή εργασία του φοιτητή Γρηγορίου Χρήστου

```
String hour = "0" + String.valueOf(i) + ":00";
idx = html.indexOf(hour);
if(idx > 0)
{
    sub1 = html.substring(idx+13, idx+19); //
    επιστρέφει μέρος του String (από, έως)
    sub2 = sub1.replaceAll("[^0-9]", "");
    if(!sub2.isEmpty())

        PM25.add(Integer.valueOf(Integer.parseInt(sub2))); // μετατρέπουμε την
τιμή String σε Integer
    }
}
for(int i=10; i<25; i++)
{
    String hour = String.valueOf(i) + ":00";
    idx = html.indexOf(hour);
    if(idx > 0)
    {
        sub1 = html.substring(idx+13, idx+19); //
        επιστρέφει μέρος του String (από, έως)
        sub2 = sub1.replaceAll("[^0-9]", "");
        if(!sub2.isEmpty())

            PM25.add(Integer.valueOf(Integer.parseInt(sub2))); // μετατρέπουμε την
τιμή String σε Integer
        }
    }
}
if(html.contains("PM10"))
{
    for(int i=1; i<10; i++)
    {
        String hour = "0" + String.valueOf(i) + ":00";
        idx = html.indexOf(hour);
        if(idx > 0)
        {
            sub1 = html.substring(idx+6, idx+12); //
            επιστρέφει μέρος του String (από, έως)
            sub2 = sub1.replaceAll("[^0-9]", "");
            if(!sub2.isEmpty())

                PM10.add(Integer.valueOf(Integer.parseInt(sub2))); // μετατρέπουμε την
τιμή String σε Integer
            }
        }
    }
    for(int i=10; i<25; i++)
    {
        String hour = String.valueOf(i) + ":00";
        idx = html.indexOf(hour);
        if(idx > 0)
        {
            sub1 = html.substring(idx+6, idx+12); //
            επιστρέφει μέρος του String (από, έως)
            sub2 = sub1.replaceAll("[^0-9]", "");
            if(!sub2.isEmpty())

                PM10.add(Integer.valueOf(Integer.parseInt(sub2))); // μετατρέπουμε την
τιμή String σε Integer
            }
        }
    }
}
```

```

    }
}

/*
 * Μέθοδος η οποία επιστρέφει τη μέση τιμή που περιέχει η html είτε για
 * PM10 είτε για PM2.5 σε δεκαδική τιμή. Δέχεται ως όρισμα ένα πίνακα από
 * ακεραίους.* */
private double getMean(int[] data)
{
    double mean = 0;
    for(int i=0; i<data.length; i++)
        mean += data[i];
    double val = (mean / (double)data.length);
    return Double.valueOf(df2.format(val));
}

/*
 * Μέθοδος η οποία υπολογίζει και επιστρέφει τη μέση τιμή είτε για PM10
 * είτε για PM2.5 την τρέχουσα στιγμή σε δεκαδική τιμή. Δέχεται ως όρισμα
 * πίνακα από δεκαδικούς αριθμούς.* */
private double getMean(double[] data)
{
    double mean = 0;
    for(int i=0; i<data.length; i++)
        mean += data[i];
    double val = (mean / (double)data.length);
    return Double.valueOf(df2.format(val));
}

/*
 * Μέθοδος η οποία υπολογίζει και επιστρέφει τη τιμή απόκλισης είτε για
 * PM10 είτε για PM2.5 την τρέχουσα στιγμή σε δεκαδική τιμή. Δέχεται ως
 * όρισμα ένα πίνακα από ακεραίους.
 * */
private double getSDev(int[] data, double mean)
{
    double[] deviations = new double[data.length];
    for(int i=0; i<data.length; i++)
        deviations[i] = Math.pow((data[i]-mean), (double)2);
    double val = Math.sqrt(getMean(deviations));
    return Double.valueOf(df2.format(val));
}

/*
 * Μέθοδος η οποία υπολογίζει και επιστρέφει τη τιμή σκέβρωσης είτε για
 * PM10 είτε για PM2.5 την τρέχουσα στιγμή σε δεκαδική τιμή. Δέχεται ως
 * όρισμα ένα πίνακα από ακεραίους, την μέση τιμή και την τιμή απόκλισης.
 * */
private double getSkewness(int[] data, double mean, double sdev)
{
    double[] deviations = new double[data.length];
    for(int i=0; i<data.length; i++)
        deviations[i] = Math.pow(((data[i]-mean)/sdev), (double)3);
    double val = getMean(deviations);
    return Double.valueOf(df2.format(val));
}

/*
 * Μέθοδος η οποία υπολογίζει και επιστρέφει τη τιμή κύρτωσης είτε για

```

```
* PM10 είτε για PM2.5 την τρέχουσα στιγμή σε δεκαδική τιμή. Δέχεται ως
* όρισμα ένα πίνακα από ακεραίους, την μέση τιμή και την τιμή
* απόκλισης.
* */
private double getKurtosis(int[] data, double mean, double sdev)
{
    double[] deviations = new double[data.length];
    for(int i=0; i<data.length; i++)
        deviations[i] = Math.pow((data[i]-mean), (double)4);
    double denom = Math.pow(sdev, (double)4);
    double val = (getMean(deviations)/denom);
    return Double.valueOf(df2.format(val));
}

/*
 * Μέθοδος η οποία υπολογίζει και επιστρέφει τη μικρότερη τιμή μέτρησης
 * είτε για PM10 είτε για PM2.5 την τρέχουσα στιγμή σε ακέραια τιμή.
 * Δέχεται ως όρισμα ένα πίνακα από ακεραίους.
 * */
private int getMin(int[] data)
{
    int min = Integer.MAX_VALUE;
    for(int i=0; i<data.length; i++)
    {
        if(data[i] < min)
        {
            min = data[i];
        }
    }
    return min;
}

/*
 * Μέθοδος η οποία υπολογίζει και επιστρέφει τη μέγιστη τιμή μέτρησης
 * είτε για PM10 είτε για PM2.5
 * την τρέχουσα στιγμή σε ακέραια τιμή. Δέχεται ως όρισμα ένα πίνακα από
 * ακεραίους.
 * */
private int getMax(int[] data)
{
    int max = Integer.MIN_VALUE;
    for(int i=0; i<data.length; i++)
    {
        if(data[i] > max)
        {
            max = data[i];
        }
    }
    return max;
}
}
```

SettingsScreen.java

```
/*
 * Κλάση η οποία χειρίζεται τη διάταξη activity_settings_screen.xml
 * και τις λειτουργίες των ρυθμίσεων, δηλαδή της προσθήκης και
 * της διαγραφής μίας τοποθεσίας
 */
```

```

* */

package gr.kepe.poltracker;

import java.io.FileOutputStream;
import java.io.ObjectOutputStream;
import java.util.ArrayList;
import java.util.HashSet;
import java.util.List;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;

public class SettingsScreen extends Activity
{
    private Spinner removeLocationSpinner;
    private Activity myParent;
    private ArrayAdapter<String> adapter;
    private LocationEntry[] locations = null;
    private HashSet<LocationEntry> locationsClone = null;
    private ArrayList<LocationEntry> defaultLocationsClone = null;
    private boolean modification = false;

    public SettingsScreen()
    {
        // τίποτα προς υλοποίηση
    }

    @Override
    public void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_settings_screen); // Φόρτωση της
        διάταξης activity_settings_screen.xml
    }

    /* Μέθοδος η οποία καλείται όταν επιστρέφουμε στο activity από κάποια
    * άλλη activity ή μετά από παύση της εφαρμογής
    * */
    public void onResume()
    {
        // Ανάκτηση των αποθηκευμένων τοποθεσιών
        super.onResume();
        modification = false;
        myParent = MainActivity.instance;
        locationsClone = ((MainActivity) myParent).locations;
        defaultLocationsClone = ((MainActivity)
myParent).defaultLocations;
        Object[] locs = locationsClone.toArray();
        locations = new LocationEntry[locs.length];
        for(int i=0; i<locs.length; i++)
            locations[i] = (LocationEntry)locs[i];

        // Αρχικοποίηση του Spinner View για την επιλογή της τοποθεσίας
        που διαγράφεται
    }
}

```

```

List<String> choices = new ArrayList<String>();
if(locations != null)
{
    for(int i=0; i<locations.length; i++)
        choices.add((String) locations[i].name);
}
adapter = new ArrayAdapter<String>(this,
android.R.layout.simple_spinner_item, choices);

adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_
item);
removeLocationSpinner = (Spinner)
findViewById(R.id.RemoveLocationSpinnerView);
removeLocationSpinner.setAdapter(adapter);
}

/* Μέθοδος η οποία καλείται όταν ο χρήστης επιλέγει το κουμπί
 * "Προσθήκη τοποθεσίας". Αναλαμβάνει την προσθήκη μιας νέας
 * τοποθεσίας (LocationEntry).
 * */
public void onClickAddLocation(View button)
{
    //Get user-provided data from text fields
    double longVal = 0;
    double latVal = 0;
    String nameVal = "", addressVal = "";
    boolean geoTagged = true;
    EditText latLoc = (EditText)
findViewById(R.id.LocationLatitudeTextFieldView);
    EditText longLoc = (EditText)
findViewById(R.id.LocationLongitudeTextFieldView);
    EditText nameLoc = (EditText)
findViewById(R.id.LocationNameTextFieldView);
    EditText addressLoc = (EditText)
findViewById(R.id.WWWTextFieldView);
    try
    {
        longVal = Double.parseDouble(longLoc.getText().toString());
        latVal = Double.parseDouble(latLoc.getText().toString());
    }
    catch (Exception e)
    {
        geoTagged = false;
    }
    try
    {
        nameVal = nameLoc.getText().toString();
        addressVal = addressLoc.getText().toString();
    }
    catch (Exception e)
    {
        Toast.makeText(this, R.string.InvalidDataWarningText,
Toast.LENGTH_LONG).show();
        return;
    }

    // Ανανέωση δεδομένων της main activity
    if((nameVal.equals("")) || (addressVal.equals("")))
    {
        Toast.makeText(this, R.string.InvalidDataWarningText,
Toast.LENGTH_LONG).show();
    }
}

```

```
        return;
    }
    if(geoTagged == false)
        locationsClone.add(new LocationEntry(nameVal, addressVal));
    else
        locationsClone.add(new LocationEntry(latVal, longVal,
nameVal, addressVal));

    // Μεταφόρτωση των τοποθεσιών στην τρέχουσα οθόνη
    adapter.add(nameVal);
    removeLocationSpinner.setAdapter(adapter);
    latLoc.setText("");
    longLoc.setText("");
    nameLoc.setText("");
    addressLoc.setText("");
    modification = true;
}

public void onClickRemoveLocation(View button)
{
    // Διαγραφή επιλεγμένης τοποθεσίας από την τρέχουσα οθόνη
    String moribund;
    try
    {
        moribund =
removeLocationSpinner.getSelectedItem().toString();
        adapter.remove(moribund);
        removeLocationSpinner.setAdapter(adapter);
    }
    catch (Exception e)
    {
        return;
    }

    //Διαγραφή της επιλεγμένης τοποθεσίας από τα δεδομένα της κύριας
οθόνης
    LocationEntry moribundLoc = new LocationEntry(moribund, "");
    locationsClone.remove(moribundLoc);
    defaultLocationsClone.remove(moribundLoc);
    modification = true;
}

public void onPause()
{
    super.onPause();
    // Ανανέωση ρυθμίσεων

    // Έλεγχος για οποιαδήποτε τροποποίηση στην λίστα τοποθεσιών
    if(modification == false)
    {
        myParent = null;
        adapter = null;
        return;
    }

    /* Εάν οι τοποθεσίες έχουν τροποποιηθεί, γράφει τις
    * ανανεωμένες τοποθεσίες στο αρχείο
    * */
    modification = false;
    ObjectOutputStream oos;
    try
```

```

    {
        oos = new ObjectOutputStream(new
FileOutputStream(this.getDir("RAW", 0).getPath().toString() +
(String)getResources().getString(R.string.LocationsFilename)));
        oos.writeObject(locationsClone);
        oos.flush();
        oos.close();
    }
    catch (Exception e)
    {
        Toast.makeText(this, R.string.CannotWriteFileWarningText,
Toast.LENGTH_LONG).show();
    }
    myParent = null;
    adapter = null;
    locationsClone = null;
}
}
}

```

ΠΑΡΑΡΤΗΜΑ Β

Στην συνέχεια παρουσιάζεται ο κώδικας σε xml ο οποίος αναπαριστά τον τρόπο διάταξης των αντικειμένων σε κάθε οθόνη της εφαρμογής μας. Όπως αναφέραμε στο Παράρτημα Α, οι κλάσεις MainActivity, ResultScreen, SettingsScreen φορτώνουν τις διατάξεις (layouts) activity_main, activity_results_screen και activity_settings_screen αντίστοιχα.

activity_main.java

```

<!-- RelativeLayout - Καθορίζει πως θα διαταχθούν τα components στην οθόνη-->
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/MainScreenLayout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/BackgroundColor"
    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=".MainActivity" >

    <!-- View με συγκεκριμένο χρώμα-->
    <View
        android:id="@+id/Horizontal"
        android:layout_width="fill_parent"
        android:layout_height="25dp"
        android:layout_alignParentTop="true"
        android:background="#EAF0FF" />

    <!-- Ετικέτα "Επιλέξτε τοποθεσία :"-->
    <TextView
        android:id="@+id/ChooseLocationTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_alignParentTop="true"
        android:text="@string/ChooseLocationTextViewText"
        android:textAppearance="?android:attr/textAppearanceLarge" />

```



```
<!-- Dropdown μενού (Spinner) με τις διαθέσιμες τοποθεσίες-->
<Spinner
    android:id="@+id/ChooseLocationSpinnerView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/ChooseLocationTextView"
    android:layout_centerHorizontal="true" />

<!-- Κουμπί "Ανανέωση Δεδομένων"-->
<Button
    android:id="@+id/UpdateDataButtonView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerVertical="true"
    android:layout_centerHorizontal="true"
    android:onClick="onClickUpdate"
    android:text="@string/GetDataButtonViewText" />

<!-- Κουμπί "Προβολή αποτελεσμάτων"-->
<Button
    android:id="@+id/DisplayResultsButtonView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/UpdateDataButtonView"
    android:layout_centerHorizontal="true"
    android:onClick="onClickDisplayResults"
    android:text="@string/DisplayResultsButtonViewText" />

<!-- Κουμπί "Ρυθμίσεις"-->
<Button
    android:id="@+id/SettingsButtionView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/DisplayResultsButtonView"
    android:layout_centerHorizontal="true"
    android:onClick="onClickSettings"
    android:text="@string/Settings" />

</RelativeLayout>
```

activity_results.java

```
<?xml version="1.0" encoding="utf-8"?>
<!-- RelativeLayout - Καθορίζει πως θα διαταχθούν τα components στην οθόνη -->
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/BackgroundColor" >

    <!-- View με συγκεκριμένο χρώμα -->
    <View
        android:id="@+id/HorizontalLineView1"
        android:layout_width="fill_parent"
        android:layout_height="5dp"
        android:layout_below="@+id/CurrentValueLabelTextView"
        android:background="#ffa500" />

    <!-- View με συγκεκριμένο χρώμα -->
    <View
```

```

        android:id="@+id/Horizontal"
        android:layout_width="fill_parent"
        android:layout_height="25dp"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        android:background="#EAffff" />

<!-- View με συγκεκριμένο χρώμα -->
<View
    android:id="@+id/View07"
    android:layout_width="fill_parent"
    android:layout_height="25dp"
    android:layout_alignBottom="@+id/PM2LabelTextView"
    android:layout_alignParentLeft="true"
    android:background="#EAffff" />

<!-- Ετικέτα "ΑΠΟΤΕΛΕΣΜΑΤΑ" -->
<TextView
    android:id="@+id/ResultsLabelTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:background="#0000"
    android:text="@string/ResultsLabelTextViewText"
    android:textAppearance="?android:attr/textAppearanceLarge" />

<!-- Ετικέτα όνομα τοποθεσίας -->
<TextView
    android:id="@+id/LocationLabelTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/ResultsLabelTextView"
    android:layout_centerHorizontal="true"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceMedium" />

<!-- Ετικέτα "PM2.5" -->
<TextView
    android:id="@+id/PM2LabelTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/PM10LabelTextView"
    android:layout_alignBottom="@+id/PM10LabelTextView"
    android:layout_alignParentRight="true"
    android:layout_marginRight="11dp"
    android:text="@string/PM2LabelTextViewText"
    android:textAppearance="?android:attr/textAppearanceMedium" />

<!-- Ετικέτα "Τρέχων:" -->
<TextView
    android:id="@+id/CurrentValueLabelTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/PM10LabelTextView"
    android:text="@string/CurrentValueLabelTextViewText"
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα "Μέση Τιμή:" -->
<TextView

```

```
        android:id="@+id/MeanValueLabelTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_below="@+id/CurrentValueLabelTextView"
        android:layout_marginTop="13dp"
        android:text="@string/MeanValueLabelTextViewText"
        android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα "Απόκλιση:" -->
<TextView
    android:id="@+id/DeviationValueLabelTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/MeanValueLabelTextView"
    android:layout_marginTop="13dp"
    android:text="@string/DeviationValueLabelTextViewText"
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα "Σκέβρωση:" -->
<TextView
    android:id="@+id/SkewValueLabelTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/DeviationValueLabelTextView"
    android:layout_marginTop="13dp"
    android:text="@string/SkewValueLabelTextViewText"
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα "Κύρτωση:" -->
<TextView
    android:id="@+id/KurtosisValueLabelTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/SkewValueLabelTextView"
    android:layout_marginTop="11dp"
    android:text="@string/KurtosisValueLabelTextViewText"
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα όνομα τοποθεσίας -->
<TextView
    android:id="@+id/CurrentLocationLabelTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/ResultsLabelTextView"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="14dp"
    android:textAppearance="?android:attr/textAppearanceMedium" />

<!-- View με συγκεκριμένο χρώμα -->
<View
    android:id="@+id/View01"
    android:layout_width="fill_parent"
    android:layout_height="5dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/MeanValueLabelTextView"
    android:background="#ffa500" />
```

```

<!-- View με συγκεκριμένο χρώμα -->
<View
    android:id="@+id/View02"
    android:layout_width="fill_parent"
    android:layout_height="5dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/DeviationValueLabelTextView"
    android:background="#ffa500" />

<!-- View με συγκεκριμένο χρώμα -->
<View
    android:id="@+id/View03"
    android:layout_width="fill_parent"
    android:layout_height="5dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/SkewValueLabelTextView"
    android:background="#ffa500" />

<!-- View με συγκεκριμένο χρώμα -->
<View
    android:id="@+id/View04"
    android:layout_width="fill_parent"
    android:layout_height="5dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/KurtosisValueLabelTextView"
    android:background="#ffa500" />

<!-- View με συγκεκριμένο χρώμα -->
<View
    android:id="@+id/View05"
    android:layout_width="fill_parent"
    android:layout_height="5dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/MaxValueLabelTextView"
    android:background="#ffa500" />

<!-- View με συγκεκριμένο χρώμα -->
<View
    android:id="@+id/View06"
    android:layout_width="fill_parent"
    android:layout_height="5dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/MinValueLabelTextView"
    android:background="#ffa500" />

<!-- Ετικέτα "PM10" -->
<TextView
    android:id="@+id/PM10LabelTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/CurrentLocationLabelTextView"
    android:layout_centerHorizontal="true"
    android:text="@string/PM10LabelTextViewText"
    android:textAppearance="?android:attr/textAppearanceMedium" />

<!-- Ετικέτα "Μέγιστη Τιμή:" -->
<TextView
    android:id="@+id/MaxValueLabelTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"

```

Πτυχιακή εργασία του φοιτητή Γρηγορίου Χρήστου

```
    android:layout_below="@+id/KurtosisValueLabelTextView"
    android:layout_marginTop="13dp"
    android:text="@string/MaxValueLabelTextViewText"
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα "Ελάχιστη Τιμή:" -->
<TextView
    android:id="@+id/MinValueLabelTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/View05"
    android:layout_marginTop="11dp"
    android:text="@string/MinValueLabelTextViewText"
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα με τρέχουσα τιμή PM10 στήλης-->
<TextView
    android:id="@+id/PM10CurValueTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/CurrentValueLabelTextView"
    android:layout_alignBottom="@+id/CurrentValueLabelTextView"
    android:layout_alignLeft="@+id/PM10LabelTextView"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα με μέση τιμή PM10 στήλης-->
<TextView
    android:id="@+id/PM10MeanValueTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/MeanValueLabelTextView"
    android:layout_alignBottom="@+id/MeanValueLabelTextView"
    android:layout_alignLeft="@+id/PM10LabelTextView"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα με τιμή απόκλισης PM10 στήλης-->
<TextView
    android:id="@+id/PM10DeviationValueTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/DeviationValueLabelTextView"
    android:layout_alignBottom="@+id/DeviationValueLabelTextView"
    android:layout_alignLeft="@+id/PM10LabelTextView"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα με τιμή σκέβρωσης PM10 στήλης-->
<TextView
    android:id="@+id/PM10SkewValueTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/SkewValueLabelTextView"
    android:layout_alignBottom="@+id/SkewValueLabelTextView"
    android:layout_alignLeft="@+id/PM10LabelTextView"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα με τιμή κύρτωσης PM10 στήλης-->
```

```

<TextView
    android:id="@+id/PM10KurtosisValueTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/KurtosisValueLabelTextView"
    android:layout_alignBottom="@+id/KurtosisValueLabelTextView"
    android:layout_alignLeft="@+id/PM10LabelTextView"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα με ελάχιστη τιμή PM10 στήλης-->
<TextView
    android:id="@+id/PM10MinValueTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/MinValueLabelTextView"
    android:layout_alignBottom="@+id/MinValueLabelTextView"
    android:layout_alignLeft="@+id/PM10LabelTextView"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα με μέγιστη τιμή PM10 στήλης-->
<TextView
    android:id="@+id/PM10MaxValueTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/MaxValueLabelTextView"
    android:layout_alignBottom="@+id/MaxValueLabelTextView"
    android:layout_alignLeft="@+id/PM10LabelTextView"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα με τρέχουσα τιμή PM2.5 στήλης-->
<TextView
    android:id="@+id/PM2CurValueTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/CurrentValueLabelTextView"
    android:layout_alignBottom="@+id/CurrentValueLabelTextView"
    android:layout_alignLeft="@+id/PM2LabelTextView"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα με μέση τιμή PM2.5 στήλης-->
<TextView
    android:id="@+id/PM2MeanValueTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/MeanValueLabelTextView"
    android:layout_alignBottom="@+id/MeanValueLabelTextView"
    android:layout_alignLeft="@+id/PM2LabelTextView"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα με τιμή απόκλισης PM2.5 στήλης-->
<TextView
    android:id="@+id/PM2DeviationValueTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/DeviationValueLabelTextView"
    android:layout_alignBottom="@+id/DeviationValueLabelTextView"

```

```
        android:layout_alignLeft="@+id/PM2LabelTextView"
        android:text=""
        android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα με τιμή σκέβρωσης PM2.5 στήλης -->
<TextView
    android:id="@+id/PM2SkewValueTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/SkewValueLabelTextView"
    android:layout_alignBottom="@+id/SkewValueLabelTextView"
    android:layout_alignLeft="@+id/PM2LabelTextView"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα με τιμή κύρτωσης PM2.5 στήλης-->
<TextView
    android:id="@+id/PM2KurtosisValueTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/KurtosisValueLabelTextView"
    android:layout_alignBottom="@+id/KurtosisValueLabelTextView"
    android:layout_alignLeft="@+id/PM2LabelTextView"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα με ελάχιστη τιμή PM2.5 στήλης -->
<TextView
    android:id="@+id/PM2MinValueTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/MinValueLabelTextView"
    android:layout_alignBottom="@+id/MinValueLabelTextView"
    android:layout_alignLeft="@+id/PM2LabelTextView"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceSmall" />

<!-- Ετικέτα με μέγιστη τιμή PM2.5 στήλης -->
<TextView
    android:id="@+id/PM2MaxValueTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/MaxValueLabelTextView"
    android:layout_alignBottom="@+id/MaxValueLabelTextView"
    android:layout_alignLeft="@+id/PM2LabelTextView"
    android:text=""
    android:textAppearance="?android:attr/textAppearanceSmall" />

</RelativeLayout>
```

activity_settings.java

```
<?xml version="1.0" encoding="utf-8"?>
<!-- RelativeLayout - Καθορίζει πως θα διαταχθούν τα components στην οθόνη -->
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/SettingsScreenLayout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/BackgroundColor"
    android:orientation="vertical" >
```

```

<!-- Ετικέτα "Γ. Μήκος:" -->
<TextView
    android:id="@+id/LatitudeTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBottom="@+id/LocationLatitudeTextFieldView"
    android:layout_alignLeft="@+id/LongitudeTextView"
    android:text="@string/LocationLatitudeTextViewText"
    android:textAppearance="?android:attr/textAppearanceMedium" />

<!-- Ετικέτα "Όνομα:" -->
<TextView
    android:id="@+id/LocationNameTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBottom="@+id/LocationNameTextFieldView"
    android:layout_alignLeft="@+id/LatitudeTextView"
    android:text="@string/LocationNameTextViewText"
    android:textAppearance="?android:attr/textAppearanceMedium" />

<!-- Κουμπί "Διαγραφή τοποθεσίας"-->
<Button
    android:id="@+id/RemoveLocationButtonView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/RemoveLocationSpinnerView"
    android:layout_centerHorizontal="true"
    android:onClick="onClickRemoveLocation"
    android:text="@string/RemoveLocationButtonViewText" />

<!-- Dropdown μενού (spinner) με τις διαθέσιμες τοποθεσίες-->
<Spinner
    android:id="@+id/RemoveLocationSpinnerView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_alignParentTop="true"/>

<!-- Ετικέτα "Γ. Πλάτος:" -->
<TextView
    android:id="@+id/LongitudeTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/RemoveLocationButtonView"
    android:layout_marginTop="78dp"
    android:text="@string/LocationLongitudeTextViewText"
    android:textAppearance="?android:attr/textAppearanceMedium" />

<!-- Πεδίο εισαγωγής χρήστη (Γ. Πλάτος) -->
<EditText
    android:id="@+id/LocationLongitudeTextFieldView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/LongitudeTextView"
    android:layout_alignBottom="@+id/LongitudeTextView"
    android:layout_toRightOf="@+id/LatitudeTextView"
    android:ems="10"
    android:inputType="number" />

```


Πτυχιακή εργασία του φοιτητή Γρηγορίου Χρήστου

```
<!-- Πεδίο εισαγωγής χρήστη (Όνομα) -->
<EditText
    android:id="@+id/LocationNameTextFieldView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/LocationLatitudeTextFieldView"
    android:layout_below="@+id/LocationLatitudeTextFieldView"
    android:ems="10"
    android:inputType="text" >
    <requestFocus />
</EditText>

<!-- Πεδίο εισαγωγής χρήστη (Γ. Πλάτος) -->
<EditText
    android:id="@+id/LocationLatitudeTextFieldView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/LocationLongitudeTextFieldView"
    android:layout_below="@+id/LongitudeTextView"
    android:ems="10"
    android:inputType="number" />

<!-- Πεδίο εισαγωγής χρήστη (Διεύθυνση) -->
<EditText
    android:id="@+id/WWWTextFieldView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/LocationNameTextFieldView"
    android:layout_below="@+id/LocationNameTextFieldView"
    android:ems="10"
    android:inputType="text" />

<!-- Ετικέτα ""WWW" -->
<TextView
    android:id="@+id/WWWTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBottom="@+id/WWWTextFieldView"
    android:layout_alignParentLeft="true"
    android:text="@string/WWWTextViewText"
    android:textAppearance="?android:attr/textAppearanceMedium" />

<!-- Κουμπί "Προσθήκη τοποθεσίας"-->
<Button
    android:id="@+id/AddLocationButtonView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/WWWTextFieldView"
    android:layout_centerHorizontal="true"
    android:onClick="onClickAddLocation"
    android:text="@string/AddLocationButtonViewText" />

</RelativeLayout>
```

ΑΝΑΦΟΡΕΣ

- [1]. Καινοτομία και κλάδοι υψηλής τεχνολογίας, Ίδρυμα Οικονομικών και Βιομηχανικών Ερευνών, 2008
- [2]. Κακούρης Α. (2010), Εννοιολογικές Προσεγγίσεις στην Επιχειρηματικότητα της Καινοτομίας, Εκδόσεις Δίαυλος
- [3]. Καραγιάννης Η.Γ. – Μπακούρος Ι.Λ. (2010), Καινοτομία & Επιχειρηματικότητα: Θεωρία – Πράξη, Θεσσαλονίκη: Σοφία
- [4]. Πραστάκος Γ, Σπανός Γ., Κωστόπουλος Κ. (2003), Καινοτομία: Προσδιοριστικοί Παράγοντες και προβληματισμοί για το Μέλλον της Ελληνικής Οικονομίας, Φεβρουάριος 2003, Αθήνα, Ελλάδα
- [5]. Adomavicius Gediminas and Tuzhilin Alexander, Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions, June 2005
- [6]. Asim Ansari and Carl F. Mela, E-customization, 2003
- [7]. Environmental Protection Department of the Government of Hong Kong Special Administrative Region,
<https://play.google.com/store/apps/details?id=hk.gov.epd.aqhi>, (accessed 8/5/2014)
- [8]. Meg Broderick and D. H. Smaltz, E-Health Defined, Proceedings of Student Research Day, CSIS, Pace University, May 9th, 2003
- [9]. Mulvenna, M. D., Anand, S. S., and Buchner, A. G. 2000. Personalization on the Net using Web Mining. *Communications of the ACM* 43, 8, 122-125.
- [10]. Murthi, B.P.S & Sarkar, S (2003). The Role of the Management Sciences in Research on personalization. *Management Science*, 49(10), 1344 – 1362
- [11]. Krishnan M. S. and Naveen Farag Awad, The personalization privacy paradox: An empirical evaluation of information transparency and the willingness to be profiled online for personalization, Detroit, USA, March 2006
- [12]. Organization for economic co-operation and development statistical office of the European communities (OECD), (2005), Guidelines for collecting and interpreting innovation Data
- [13]. Robert N. Stone, Kjell Grønhaug, (1993) "Perceived Risk: Further Considerations for the Marketing Discipline", *European Journal of Marketing*, Vol. 27 Iss: 3, pp.39 - 50
- [14]. Rutherford A. and Botha Reinhardt, Towards Personalized Content, SAICSIT 2003

- [15]. Sensing Air Pollution, Enhance Environmental Awareness through Social Information Technologies, <http://www.everyaware.eu/activities/case-studies/air-quality/> (accessed 9/5/2014)
- [16]. Shailendra C. Jain Palvia and Sushil S. Sharma, E-Government and E-Governance: Definitions/Domain Framework and Status around the World, USA, 2008
- [17]. Shari Lawrence Pflieger (2001), Software Engineering: Theory and Practice 2nd Edition, Pearson Education Inc, 2001
- [18]. Wind, Jerry and Arvind Rangaswamy (2001), “Customerization: The Next Revolution in Mass Customization,” Journal of Interactive Marketing, Vol. 15,. No. 1, p. 13-32.
- [19]. Zickuhr Kathryn, Three-quarters of Smartphone owners use location-based services, Pew Research Internet Project, <http://www.pewinternet.org/2012/05/11/three-quarters-of-smartphone-owners-use-location-based-services/> (accessed 8/5/2014)
- [20]. Zorayda Ruth Andam, e-Commerce and e-Business, e-ASEAN Task Force and the UNDP Asia Pacific Development Information Programme (UNDP-APDIP)May 2003

ΒΙΒΛΙΟΓΡΑΦΙΑ

Αυλωνίτης Γ. (2003), Καινοτομία: Προσδιοριστικοί Παράγοντες & Προβληματισμοί για το Μέλλον της Ελληνικής Οικονομίας, Οικονομικό Πανεπιστήμιο Αθηνών, Ελλάδα

Χατζηγεωργίου Α. (2008), Ανάπτυξη συστήματος λογισμικού βάσει της μεθοδολογίας ICONIX, Ελληνικό Ανοικτό Πανεπιστήμιο

Dan Pilone, Neil Pitman, "UML 2.0 in a Nutshell", O'Reilly Media, 2005

F. Lin and W. Ye, "Operating System Battle in the Ecosystem of Smartphone Industry", Proc. 2009 International Symposium on Information Engineering and Electronic Commerce, 2009, pp.617- 627, doi: 10.1109/IEEC.2009.136.

Gary Yakimov and Lindsey Woolsey with Contributions from MEP Staff, "Innovation and Product Development in the 21st Century", Hollings Manufacturing Extension Partnership Advisory Board February 2010

Guntram Graef and Martin Gaedke, "An Evolution-oriented Architecture for Web Applications", Telecooperation Office (TecO), University of Karlsruhe, Vincenz-Prießnitz Str. 1, 76131 Karlsruhe, Germany, september 2012

Jack Zhang, Shikhar Sagar, and Emad Shihab, "The Evolution of Mobile Apps: An Exploratory Study", Department of Software Engineering Rochester, New York, USA, 14623, August 2013

James Steele, Nelson To, "The Android developer's cookbook : building applications with the Android SDK", Boston, October 2010

Johan Hellström, "The Innovative Use of Mobile Applications in East Africa", Sida, Commissioned by Sida, Department for Human Development, Team for Knowledge, ICT and Education, May 2010

Judith S. MacCormick, Kristine Dery, Darl G.Kolb, Engaged or just connected? Smartphones and employee engagement, <http://staff.business.auckland.ac.nz/DesktopModules/StaffProfiles/Publications/9096.pdf> (accessed 3/5/2014)

Kathleen Webb, "Mobile Innovation: Ontario's Growing Mobile Content, Services, and Applications, Industry 2012", Mobile Experience Innovation Centre (MEIC) OCAD University Toronto, Ontario Canada, 2012

Laurie Williams, "An Introduction to the Unified Modeling Language", 2004

Mark Murphy, "Beginning Android 3", Apress, 2011

Matthew Baxter-Reynolds (2010), Multimobile Development: Building Application for the iPhone and Android Platforms, USA, 2010

Min Jung Kim, (2012), The Interdependence between Smartphones and Applications: The Role of Platforms, University of Minnesota, December 2012

Pei Zheng, Lionel M. Ni, Spotlight: The Rise of the Smart Phone, Microsoft - Hong Kong University of Science and Technology, March 2006

R. Ballagas, J. Borchers, M. Rohs, J.G. Sheridan, "The smart phone: ubiquitous input device", IEEE Pervasive Computing, vol. 5, issue 1, March 2006, pp. 70-77, doi: 10.1109/MPRV.2006.1

Richard E. Scott & Ayida Saeed, Global eHealth - Measuring Outcomes: Why, What, and How, A Report Commissioned by the World Health Organization's Global Observatory for eHealth, Bellagio, Italy, August 2008

Robert Cooper, Scott Edgett, "Ideation for Product Innovation: What are the best methods?", Stage-Gate International and Product Development Institute Inc., March 2008

Sushant A. Patinge, Pravin D. Soni, "A Survey on Instant Message and Location Sharing System for Android", Department of CSE, P R Patil College of Engg. & Tech., Amravati-444602, India

The Government of the Hong Kong Special Administrative Region Environmental Protection Department, http://www.epd.gov.hk/epd/english/environmentinhk/air/air_maincontent.html#top (accessed 10/5/2014)

Penelope Curtin, John Stanwick and Francesca Beddie, (2011), Fostering enterprise: the innovation and skills nexus – research readings, National Centre for Vocational Education Research, Australia

William Enck, Peter Gilbert, Byung-Gon Chun, Landon P. Cox, Jaeyeon Jung, Patrick McDaniel, Anmol N. Sheth (2011), TaintDroid: An Information-Flow Tracking System for Realtime Privacy Monitoring on Smartphones

Xuetao Wei, Lorenzo Gomez, Iulian Neamtiu, Michalis Faloutsos, "Permission Evolution in the Android Ecosystem", Department of Computer Science and Engineering University of California, Riverside, 2012