

ΑΤΟΜΙΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

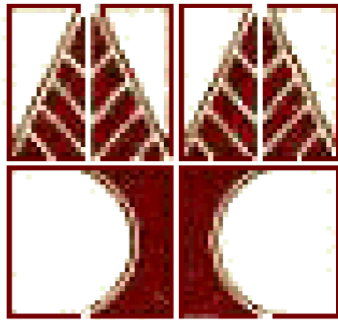
ΕΠΑΝΑΧΡΗΣΙΜΟΠΟΙΗΣΗ ΚΑΙ ΕΠΕΚΤΑΣΙΜΟΤΗΤΑ ΕΝΔΙΑΜΕΣΟΥ

ΛΟΓΙΣΜΙΚΟΥ ΜΕΣΩ RSCM

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΙΟΥΝΙΟΣ 2014



ΛΟΥΚΙΑ ΘΕΜΙΣΤΟΚΛΕΟΥΣ

986715

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΓΕΩΡΓΙΟΣ ΠΑΠΑΔΟΠΟΥΛΟΣ

**ΜΕΛΕΤΗ ΤΗΣ ΑΡΧΙΤΕΚΤΟΝΙΚΗΣ ΕΝΔΙΑΜΕΣΟΥ ΛΟΓΙΣΜΙΚΟΥ ΚΑΙ ΧΡΗΣΗ
ΤΟΥ RSCM ΕΝΔΙΑΜΕΣΟΥ ΛΟΓΙΣΜΙΚΟΥ ΓΙΑ ΕΠΑΝΑΧΡΗΣΙΜΟΠΟΙΗΣΗ ΚΑΙ
ΕΠΕΚΤΑΣΙΜΟΤΗΤΑ ΣΕ ΕΦΑΡΜΟΓΕΣ ANDROID**

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής στο Πανεπιστήμιο Κύπρου.

Ευχαριστίες

Θα ήθελα αρχικά να ευχαριστήσω τον επιβλέποντα καθηγητή μου κύριο Γιώργο Παπαδόπουλο ο οποίος μου ανέθεσε ένα θέμα το οποίο ήταν ενδιαφέρον και μου έδινε την δυνατότητα να αποκτήσω νέες γνώσεις σε θέματα τα οποία δεν είχα την ευκαιρία μέχρι σήμερα να μελετήσω.

Επίσης, θα ήθελα να ευχαριστήσω τον βοηθό μου Αχιλλέα Αχιλλέως για την καθοδήγηση και την άμεση βοήθεια που μου προσέφερε καθόλη την διάρκεια της προσπάθειας μου για την ολοκλήρωση της Διπλωματικής μου εργασίας.

ΠΕΡΙΛΗΨΗ

Η αυξανόμενη ανάπτυξη στην αγορά των κινητών συσκευών εισήγαγε την ανάπτυξη κινητών εφαρμογών που θα ασπάζονται τις ανάγκες του κάθε ατόμου προσφέροντας του την όσο το δυνατόν καλύτερη διαχείριση, σε προσωπικό όσο και σε επαγγελματικό επίπεδο. Αν κάποιος σήμερα συγκρίνει την ζήτηση που υπήρχε στην αγορά στο παρελθόν για αυτού του είδους συσκευές και τον τρόπο χρήσης τους, τις τεχνολογίες, τους χρήστες αλλά και αυτές καθαυτές τις συσκευές ίσως κάποιος χαρακτηρίσει την περίοδο που διανύουμε ως την χρυσή εποχή της κινητής τεχνολογίας. Αυτό ταυτόχρονα δεν σημαίνει ότι η τεχνολογία δεν μπορεί να εκπλήξει ακόμα περισσότερο. Αντίθετα, μπορεί να πει κανείς ότι αυξάνει τον πήχη για την τεχνολογική ανάπτυξη.

Η ανάγκη για περεταίρω έρευνα για βελτιστοποίηση των πηγών, εφαρμογών και τεχνικών ανάπτυξης είναι δεδομένη όπως η αέναη ανάγκη του ανθρώπου να εξερευνά το περιβάλλον γύρω του και να αλληλεπιδρά μαζί του. Η έρευνα αυτή βασίζεται κυρίως σε άξονες όπως λύση προβλημάτων, χρησιμοποίηση πόρων, ελαχιστοποίηση του κόστους και μείωση του χρόνου ανταπόκρισης των μηχανών και των λογισμικών τους.

Μια καινοτομία για την δημιουργία και βελτιστοποίηση λογισμικού ή εφαρμογών βασίζεται σε ένα ή περισσότερους από τους πιο πάνω άξονες. Οι εφαρμογές για κινητές συσκευές εσωκλείουν ακόμα ένα άξονα έρευνας, την αλληλεπίδραση συσκευής, περιβάλλοντος και χρήστη. Ο άξονας αυτός απαιτεί συσχέτιση των τριών πιο πάνω μεταβλητών με εναλλασσόμενες σχέσεις μεταξύ τους. Η τεχνολογία της εφαρμογής πλαισίου επιτρέπει την συσχέτιση αυτή, μειώνοντας παράλληλα χρόνο, πόρους και κόστος μέσω επαναχρησιμοποίησης λογισμικού.

Η επαναχρησιμοποίηση λογισμικού είναι γνωστή στον κόσμο της Πληροφορικής από την εισαγωγή βιβλιοθηκών σε οποιαδήποτε εφαρμογή για να εξυπηρετεί διάφορες λειτουργίες, οι οποίες διαφορετικά θα ήταν χρονοβόρες να σχεδιαστούν και να υλοποιηθούν από την αρχή. Στις εφαρμογές επίγνωσης πλαισίου η επαναχρησιμοποίηση συστατικών για χρησιμοποίηση τους από ένα μεγάλο εύρος εφαρμογών είναι βασική, όπως η χρήση sensors και reasoners plug-ins, παρέχοντας έτσι μια γενικευμένη λύση για ένα συγκεκριμένο πρόβλημα (Generic Solution). Οι άξονες οι οποίοι επηρεάζονται από τις συγκεκριμένες εφαρμογές είναι η μείωση στο χρόνο και κόστος και βελτιστοποίηση της αλληλεπίδρασης. Αυτό αυτόματα μπορεί να επιφέρει μείωση χρόνου και κόστους για τις εταιρίες που αναπτύσσουν τέτοιου είδους λογισμικά και εφαρμογές, το οποίο πολλές φορές είναι το νούμερο ένα ζητούμενο για κάποια εταιρία. Αναζητείται πολλές φορές το βέλτιστο αποτέλεσμα στον ελάχιστο χρόνο και με το ελάχιστο κόστος.

Το RSCM είναι ένα ενδιάμεσο λογισμικό επίγνωσης πλαισίου, το οποίο είναι υλοποιημένο για την πλατφόρμα Android και παρέχει την δυνατότητα ανάκτησης, διαχείρισης και επεξεργασίας δεδομένων σε πραγματικό χρόνο για την εξαγωγή των ανάλογων αποτελεσμάτων και παρουσίαση των αναγκαίων πληροφοριών προς το χρήστη. Αυτό επιτυγχάνεται με την χρήση των υφιστάμενων sensor και reasoner plugins που μας παρέχει το RSCM αλλά και την υλοποίηση νέων sensor και reasoner

plugins τα οποία ενσωματώνονται και υποστηρίζονται άμεσα μέσω του RSCM. Αυτό παρέχει την δυνατότητα επιτάχυνσης της υλοποίησης εφαρμογών επίγνωσης πλαισίου με την επαναχρησιμοποίηση λογισμικού (π.χ., RSCM plug-ins), το οποίο δίνει την δυνατότητα εκτέλεσης πολύπλοκων λειτουργιών.

Με τις δυνατότητες που δίνονται για την ανάπτυξη τέτοιων εφαρμογών, μέσω της χρήσης του ενδιάμεσου λογισμικού RSCM, η διπλωματική αυτή αποσκοπεί στην μελέτη και εκμάθηση του λογισμικού RSCM και την δημιουργία μιας εφαρμογής επίγνωσης πλαισίου με απώτερο σκοπό την τεκμηρίωση, μέσω των συμπερασμάτων της διπλωματικής, παρουσιάζοντας την δυνατότητα βελτιστοποίησης του χρόνου και συνεπώς του κόστους υλοποίησης εφαρμογών επίγνωσης πλαισίου. Ασφαλώς κάποιος χρόνος είναι αναγκαίος για την μελέτη και εκμάθηση χρήσης του RSCM αλλά μέσω της επαναχρησιμοποίησης επιτυγχάνεται μακροπρόθεσμα ο σκοπός της μείωσης του χρόνου και του κόστους υλοποίησης εφαρμογών επίγνωσης πλαισίου. Η εφαρμογή η οποία έχει υλοποιηθεί στα πλαίσια της διπλωματικής χρησιμοποιεί το RSCM ενδιάμεσο λογισμικό, υλοποιημένα RSCM συστατικά (plug-ins) και υλοποιεί επιπρόσθετο reasoner plug-in για την ανάκτηση της θέσης του χρήστη μέσω της συσκευής του, την επεξεργασία και προσδιορισμό του τμήματος του Πανεπιστημίου Κύπρου στο οποίο βρίσκεται ο χρήστης την δεδομένη στιγμή, ούτως ώστε να παρουσιάσει στο χρήστη πληροφορίες για εκδηλώσεις (π.χ. σεμινάρια) που έχουν προγραμματιστεί στο συγκεκριμένο τμήμα αυτή την εβδομάδα.

Contents

1.	ΕΙΣΑΓΩΓΗ.....	7
1.1.	Κινητές Τεχνολογίες και Εφαρμογές Επίγνωσης Πλαισίου	7
1.2.	Context Awareness.....	11
2.	ΤΕΧΝΟΛΟΓΙΑ ANDROID.....	15
2.1.	Εισαγωγή στην Τεχνολογία Android	15
2.1.1.	Πλεονεκτήματα	16
2.1.2.	Μειονεκτήματα	17
2.2.	Η Αρχιτεκτονική της Πλατφόρμας Android	18
2.3.	Επιμέρους συστατικά της στοίβας του Android.....	19
2.3.1.	Linux Kernel	19
2.3.2.	Βιβλιοθήκες.....	20
2.3.3.	Χρόνος Εκτέλεσης-Android Runtime.....	20
2.3.4.	Πλαίσιο Εφαρμογής-Application Framework.....	21
2.3.5.	Εφαρμογές	21
3.	ΕΝΔΙΑΜΕΣΑ ΛΟΓΙΣΜΙΚΑ.....	23
3.1.	Εισαγωγή στα Ενδιάμεσα Λογισμικά Επίγνωσης Πλαισίου	23
3.1.1.	Πλεονεκτήματα	25
3.1.2.	Μειονεκτήματα	26
3.2.	Σύνοψη των Τεχνολογιών Ενδιάμεσου Λογισμικού Επίγνωσης Πλαισίου	27
3.2.1.	IST MUSIC	27
3.2.2.	WASP-Web Architectures for Services Platforms	28
3.2.3.	RSCM-Really Simple Context Middleware.....	28
4.	ΥΛΟΠΟΙΗΣΗ ΕΦΑΡΜΟΓΗΣ	31
4.1.	Γενική Περιγραφή της RSCM-based Εφαρμογής CampusQuest	31
4.2.	Υλοποίηση της Εφαρμογής Campus Quest	32
5.	ΣΥΜΠΕΡΑΣΜΑΤΑ	41
	Αναφορές.....	44

Λίστα Εικόνων

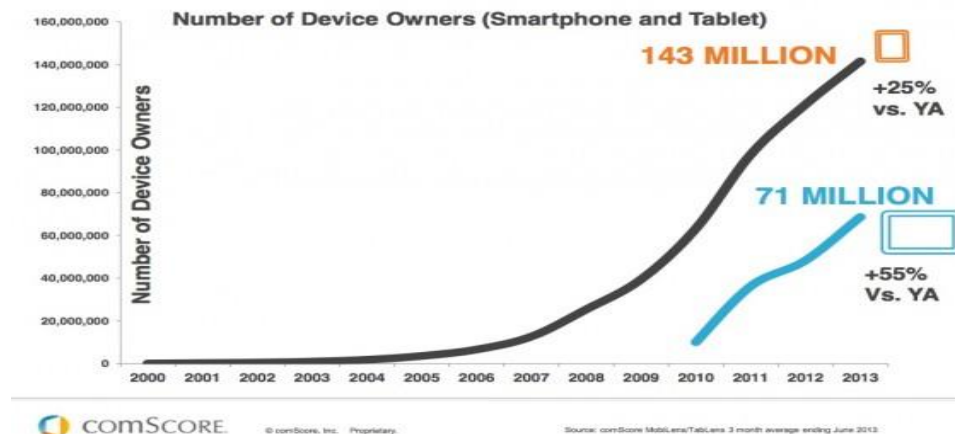
Εικόνα 1: Ρυθμός αύξησης των χρηστών των κινητών συσκευών [1].....	7
Εικόνα 2: Στάδια που ακολουθήθηκαν για την ανάπτυξη της μεθοδολογίας επίγνωσης πλαισίου [3]	9
Εικόνα 3: Μεθοδολογία ανάπτυξης εφαρμογών επίγνωσης πλαισίου [10]	12
Εικόνα 4: Μερίδιο αγοράς λειτουργικών συστημάτων κινητών συσκευών για το τρίτο τεταρτημόριο του 2013 (Ιούλιος-Αύγουστος-Σεπτέμβριος) [17]	16
Εικόνα 5: Αρχιτεκτονική Android [22].....	19
Εικόνα 6: Μήνυμα για ενεργοποίηση GPS της συσκευής.....	38
Εικόνα 7: Παραδείγματα Εκτέλεσης και Παρουσίασης Εκδηλώσεων.	39

1. ΕΙΣΑΓΩΓΗ

1.1. Κινητές Τεχνολογίες και Εφαρμογές Επίγνωσης Πλαισίου

Η διείσδυση των κινητών συσκευών και των κινητών εφαρμογών στην ζωή των ανθρώπων έχει γίνει με ένα απίστευτα γρήγορο ρυθμό. Το μερίδιο της παγκόσμιας αγοράς που σχετίζεται τόσο με τις κινητές συσκευές, όσο και με την παροχή υπηρεσιών είναι πολύ υψηλό. Οι δυνατότητες που παρέχονται από τέτοιες συσκευές είναι τεράστιες καθώς έχουν φτάσει σε τέτοιο σημείο ανάπτυξης που τείνουν να αντικαταστήσουν τους υπολογιστές. Πλέον δεν υπάρχει κάτι που να μπορεί να ξεχωρίσει τί ακριβώς είναι υπολογιστής και τί έξυπνη κινητή συσκευή και ποια ακριβώς είναι η διαφορά τους. Οι συσκευές αυτές έχουν γίνει ιδιαίτερα δημοφιλείς αφού μέσα σε μικρό χρονικό διάστημα έχουν αναπτυχθεί εφαρμογές που μπορούν να αναπληρώσουν άλλα τεχνολογικά μέσα, όπως φωτογραφικές μηχανές, mp3 players και φυσικά παρέχουν πρόσβαση στο διαδίκτυο.

Σύμφωνα με έρευνα της εταιρίας Deloitte οι καταναλωτές ανά την υφήλιο χρησιμοποιούν περισσότερες από ποτέ κινητές συσκευές. Οι πωλήσεις των κινητών συσκευών αυξήθηκαν κατά το 2013 με ποσοστό 38.4% από την προηγούμενη χρονιά. Όπως έχει δείξει η έρευνα η αγορά αναπτύσσεται τόσο πολύ που διαφαίνονται κάποια ζητήματα όπως η αύξηση του όγκου των δεδομένων, η σημασία της ταχύτητας, ακόμα και η ύπαρξη 'ανεκμετάλλετου' καταναλωτικού κοινού.



Εικόνα 1: Ρυθμός αύξησης των χρηστών των κινητών συσκευών [1]

Με την αύξηση της αγοράς των κινητών συσκευών έχει καταγραφεί και αύξηση των τεχνολογιών που υποστηρίζονται από αυτές καθώς και η συνεχής βελτίωση και ανάπτυξη στις κινητές συσκευές, προσφέροντας πάντα καινοτομία. Οι χρήστες των κινητών συσκευών αυξάνονται καθημερινά με ραγδαίους ρυθμούς. Δεν μπορούν να κατηγοριοποιηθούν οι χρήστες καθώς οι συσκευές αυτές έχουν εισβάλει πλέον στη ζωή κάθε κατηγορίας ανθρώπων ανεξαρτήτως ηλικίας, επαγγέλματος, εθνικότητας. Υπάρχουν φυσικά κάποιες συσκευές οι οποίες απευθύνονται σε συγκεκριμένους χρήστες και οι οποίες υποστηρίζουν εξειδικευμένες εφαρμογές. Η ίδια έρευνα έχει δείξει ότι οι καταναλωτές των ανεπτυγμένων αγορών και οι επαγγελματίες σε πόλεις

των αναπτυσσόμενων αγορών έχουν μέχρι και εννιά συσκευές στην κατοχή τους, ποσοστά τεράστια τα οποία αποδεικνύουν την χρηστικότητα των συσκευών. Μέσα από μια έρευνα που έχει γίνει παρουσιάζονται στην Εικόνα 1 τα ποσοστά χρηστών των κινητών συσκευών.

Όπως διακρίνεται από την γραφική παράσταση ο ρυθμός με τον οποίο αυξάνονται οι χρήστες των κινητών συσκευών από το 2005 και ύστερα είναι εκθετικός. Σήμερα τα ποσοστά χρηστών μπορούν να χαρακτηριστούν υπέρογκα και ο ρυθμός αύξησης των χρηστών απίστευτα γρήγορος.

Η ραγδαία εξάπλωση των κινητών συσκευών έχει φέρει την ανάγκη για δημιουργία και ακολούθως βελτιστοποίηση των τεχνικών ανάπτυξης λογισμικού. Ερευνούνται συνεχώς καλύτερες μεθοδολογίες και εργαλεία για την ανάπτυξη λογισμικού. Η ανάγκη για ποιοτικό λογισμικό συνεχώς αυξάνεται. Καθημερινά όλο και περισσότερες ανθρώπινες δραστηριότητες αυτοματοποιούνται με τη χρήση των υπολογιστών. Παράλληλα όμως αυξάνεται σε μεγάλο βαθμό το κόστος για την ανάπτυξη λογισμικού. Η παραγωγή λογισμικού είναι σήμερα σε κατάσταση κρίσης και απαιτούνται νέα εργαλεία και μέθοδοι προγραμματισμού. Αφού γίνεται αναφορά σε κινητές συσκευές τότε επιτάσσεται η ανάγκη για ανάπτυξη μεθοδολογιών ούτως ώστε να επιδιώκεται η μέγιστη αυτονομία της συσκευής. Οι χρήστες των κινητών συσκευών έχουν αρχίσει να καταλαβαίνουν την σημασία των κινητών εφαρμογών και την σπουδαιότητα της λειτουργίας τους. Η τεχνολογική επανάσταση των κινητών συσκευών έχει εκτινάξει την διάδοση των κινητών εφαρμογών και έτσι αναζητούνται συνεχώς τεχνικές οι οποίες θα μειώσουν τον χρόνο υλοποίησης τους και ταυτόχρονα αυτό θα μειώσει το κόστος της εφαρμογής. Ακολουθώντας μια σωστή στρατηγική δημιουργίας κινητών εφαρμογών, όπως για παράδειγμα η μεθοδολογία γραφής κώδικα να μπορεί να είναι κατανοητή από διάφορους προγραμματιστές χρησιμοποιώντας μια αρκετά διαδεδομένη γλώσσα προγραμματισμού, μειώνονται τα κόστη και ο χρόνος δημιουργίας της εφαρμογής σε μεγάλο βαθμό, εξασφαλίζοντας το επιθυμητό αποτέλεσμα και μεγαλύτερο κέρδος για τις εταιρίες που θα πλασάρουν στην αγορά την εφαρμογή.

Η τεχνολογία των κινητών εφαρμογών έχει διευρύνει τους ορίζοντες τις και έχουν δημιουργηθεί εφαρμογές οι οποίες είναι σχεδιασμένες με συγκεκριμένες στρατηγικές. Μια πολύ διαδεδομένη μορφή είναι οι εφαρμογές επίγνωσης πλαισίου, οι οποίες μετατρέπουν χαμηλού επιπέδου πληροφορίες που λαμβάνονται από το υλικό της κινητής συσκευής, από το περιβάλλον της συσκευής και σχετικά με τον χρήστη της συσκευής σε υψηλού επιπέδου πληροφορίες, οι οποίες μπορούν να παρουσιαστούν σε μορφή κατανοητή προς το χρήστη. Για τέτοιου είδους εφαρμογές είναι ακόμα πιο αναγκαίο να βελτιστοποιηθούν οι τεχνικές δημιουργίας λόγω της πολυπλοκότητας της λειτουργίας τους, η οποία πηγάζει κυρίως από τις πολλές πηγές πληροφοριών που χρειάζεται να λάβουν υπόψη και να επεξεργαστούν ώστε να λάβουν έξυπνες και δυναμικές αποφάσεις ως προς το όφελος του χρήστη [2]. Άρα, βελτιώνοντας την τεχνική δημιουργίας μιας τέτοιας εφαρμογής αυτόματα θα ελαχιστοποιηθεί ο χρόνος υλοποίησης τους και θα μειωθεί σε μεγάλο βαθμό το κόστος. Υπάρχει συνεπώς η ανάγκη για δημιουργία εργαλείου ή τεχνικής που να

υποστηρίζει εξ'ολοκλήρου την ανάπτυξη εφαρμογών επίγνωσης πλαισίου, αφού η χρήση τους αυξάνεται ολοένα και πιο πολύ. Οι εφαρμογές επίγνωσης πλαισίου πολλές φορές για να μειώσουν το χρόνο ολοκλήρωσης τους χρησιμοποιούν την μέθοδο επαναχρησιμοποίησης λογισμικού. Στο πιο κάτω γράφημα παρουσιάζεται η πορεία των context-aware content:



Εικόνα 2: Στάδια που ακολουθήθηκαν για την ανάπτυξη της μεθοδολογίας επίγνωσης πλαισίου [3]

Το 2011 αναπτύσσοντας τέτοιου είδους μεθόδους το σύστημα μπορούσε να πάρει μια πληροφορία και να την κατανοήσει χωρίς όμως να μπορέσει να την επεξεργαστεί και να την παρουσιάσει ως έξοδο σε κατανοητή μορφή. Προοδεύοντας το 2012 μπορέσαμε να δώσουμε χαρακτηριστικά στην πληροφορία που λαμβάναμε για να πάρουμε καλύτερα αποτελέσματα, τα οποία θα μας ήταν χρήσιμα. Το 2013 μέχρι σήμερα αναζητείται η χρήση της πληροφορίας σε πραγματικό χρόνο και η έξοδος του αποτελέσματος σε ορθή μορφή χωρίς ελλείψεις. Στο μέλλον σκοπός των επιστημόνων είναι να χρησιμοποιούν πληροφορίες από διαφορετικές εφαρμογές επίγνωσης πλαισίου, να τις επεξεργάζονται και να εξαγάγουν ένα πολύπλοκο αποτέλεσμα συνδυάζοντας πολλές διαφορετικές πληροφορίες πραγματικού χρόνου.

Η επαναχρησιμοποίηση λογισμικού είναι μια διαδικασία ανάπτυξης λογισμικού που δίνει την δυνατότητα δημιουργίας επαναχρησιμοποιήσιμων συστατικών (components, plug-ins), τα οποία με τη βοήθεια ενός ενδιάμεσου λογισμικού επίγνωσης πλαισίου παρέχουν την δυνατότητα χρήσης τους σε διάφορες εφαρμογές. Υπάρχουν πολλές λειτουργίες οι οποίες είναι πολύ δύσκολο να υλοποιηθούν και η ανάπτυξη τους πολλές φορές είναι ζημιογόνα και εμπεριέχει μεγάλο ρίσκο για την ορθότητα τους. Έτσι, με την επαναχρησιμοποίηση λογισμικού υπάρχει η δυνατότητα επαναχρησιμοποίησης ήδη υλοποιημένων συστατικών τα οποία επιτελούν πολύπλοκες λειτουργίες. Τέτοιου είδους λειτουργίες αποτελούν η αναγνώριση του επιπέδου μπαταρίας μιας κινητής συσκευής, η αναγνώριση της θέσης της κινητής συσκευής ή αν η συσκευή είναι συνδεδεμένη στο διαδίκτυο ή όχι. Ακολουθώντας

αυτή την μέθοδο ανάπτυξης λογισμικού ο προγραμματιστής δεν είναι αναγκαίο να υλοποιήσει μια τέτοια πολύπλοκη εφαρμογή από την αρχή αλλά παίρνει τα ήδη υλοποιησιμα κομμάτια και τα προσαρμόζει τροποποιώντας τα ανάλογα με τις ανάγκες της δικής του εφαρμογής. Έτσι, ελαχιστοποιείται το κόστος ανάπτυξης της εφαρμογής αφού μειώνεται ο χρόνος υλοποίησης της και ταυτόχρονα αποφεύγεται το ρίσκο της όλης διαδικασίας. Η διαδικασία αυτή πολλές φορές μπορεί να επιφέρει άριστα αποτελέσματα εξασφαλίζοντας ταυτόχρονα αύξηση της αξιοπιστίας της εφαρμογής, καθώς η χρησιμοποίηση συστατικών στοιχείων τα οποία είναι ήδη σε εφαρμογή και έχουν ελεγχθεί για την ορθότητα τους είναι πολύ δύσκολο να επιφέρει λάθη και ατέλειες κατά την χρήση τους.

Όπως έχει προαναφερθεί για να εφαρμοστεί η διαδικασία επαναχρησιμοποίησης λογισμικού χρειάζεται ένα ενδιάμεσο λογισμικό. Το ενδιάμεσο λογισμικό μπορεί να χαρακτηριστεί ως ο συνδετικός κρίκος μεταξύ συστήματος και εφαρμογής, αφού διαχειρίζεται και επεξεργάζεται τις πληροφορίες που μπαίνουν ως είσοδος και βγαίνουν ως έξοδος από το σύστημα. Ουσιαστικά είναι κάποια αντικείμενα και διαδικασίες που επικοινωνούν μεταξύ τους ούτως ώστε να συνδυάσουν τις πληροφορίες που λαμβάνουν και να τις μετατρέψουν στην επιθυμητή έξοδο σε ένα ή πολλούς πόρους εξόδου[4].

Το ενδιάμεσο λογισμικό το οποίο μελετήθηκε και εφαρμόστηκε στην διπλωματική εργασία είναι το RSCM. Είναι ένα ενδιάμεσο λογισμικό που σχεδιάστηκε για την πλατφόρμα Android, το οποίο επιτρέπει το σχεδιασμό context-aware εφαρμογών (εφαρμογών επίγνωσης πλαισίου) μέσα από υπάρχοντα ή καινούρια plug-ins. Η χρησιμότητα του είναι πολύ σημαντική καθώς δίνεται η ευκαιρία στο χρήστη να λάβει τις πληροφορίες που χρειάζεται στο σωστό χρόνο. Χρησιμοποιείται συχνά σε εφαρμογές που χρειάζονται πληροφορίες πραγματικού χρόνου, όπου αλλάζουν συνεχώς οι τιμές κάποιων μεταβλητών και μέσα από την επεξεργασία τους εμφανίζονται τα αποτελέσματα στον χρήστη με αξιοπιστία και ορθότητα [5].

Μια τέτοια εφαρμογή επίγνωσης πλαισίου (context-aware application) έχει υλοποιηθεί στα πλαίσια αυτής της διπλωματικής. Η εφαρμογή είναι χτισμένη για Android και βασίζεται στην διαδικασία επαναχρησιμοποίησης λογισμικού. Η ανάπτυξη της βασίστηκε στην μελέτη του ενδιάμεσου λογισμικού RSCM και την χρήση υπάρχοντων συστατικών αισθητήρων (plug-ins), π.χ. να λαμβάνει μέσω ενός αισθητήρα GPS την θέση μιας κινητής συσκευής μέσα στο Πανεπιστήμιο Κύπρου. Η ανάκτηση του δεδομένου εισόδου (δηλ. θέση μέσω GPS) επιταχύνεται με ευκολία με την χρήση του ήδη υλοποιημένου αισθητήρα συστατικού (plug-in) του ενδιάμεσου λογισμικού και με την υλοποίηση ενός επιπλέον συστατικού (plug-in) το οποίο παρέχει την δυνατότητα επεξεργασίας της θέσης GPS με στόχο να παρουσιάζει ως έξοδο το κτίριο στο οποίο βρίσκεται την συγκεκριμένη στιγμή ο χρήστης με την κινητή συσκευή του. Βάση της λογικής επεξεργασίας της θέσης GPS και την απόφαση όσο αφορά το κτίριο που επισκέπτεται ο χρήστης του παρουσιάζονται σαν τελικό αποτέλεσμα οι εκδηλώσεις που θα πραγματοποιηθούν στο συγκεκριμένο Πανεπιστημιακό τμήμα μέσα στις επόμενες μέρες.

Η χρήση του RSCM ενδιάμεσου λογισμικού κρίνεται απολύτως αναγκαία καθώς οι πληροφορίες που λαμβάνονται αλλάζουν συνεχώς σε πραγματικό χρόνο και συνεπώς βάση της διαχείρισης πληροφοριών και της επεξεργασίας τους τα ανάλογα αποτελέσματα παρουσιάζονται στον χρήστη [6].

1.2. Context Awareness

Με τον όρο context awareness εννοούμε στα ελληνικά την επίγνωση πλαισίου, μια από τις πιο σημαντικές πτυχές στον κόσμο του υπολογισμού. Μπορούμε να πούμε ότι είναι η ικανότητα των κινητών υπολογιστικών συστημάτων να αντιλαμβάνονται και να ανακτούν πληροφορίες, οι οποίες σχετίζονται με το περιβάλλον του χρήστη και να προσαρμόζονται σε αυτό, έχοντας ως σκοπό την λήψη αποφάσεων που αφορούν τον χρήστη [7].

Οι εφαρμογές οι οποίες χρησιμοποιούν τέτοια τεχνολογία αποσκοπούν στην αναγνώριση, ταξινόμηση και πρόβλεψη της πληροφορίας που παίρνουν με σκοπό να δρουν αποδοτικά εκ των προτέρων και με βάση τις προτιμήσεις του χρήστη. Παρέχει δυνατότητες σε ένα σύστημα να προσαρμόζεται στις ανάγκες των χρηστών λαμβάνοντας αυτόνομα αποφάσεις και ελαχιστοποιώντας την παρέμβαση τους. Δηλαδή, πρέπει να έχει επίγνωση της κατάστασης του χρήστη ώστε να τροποποιεί κατάλληλα την συμπεριφορά του, να λαμβάνει αποφάσεις, να προβαίνει σε προκαθορισμένες ενέργειες και να προσαρμόζεται σε διάφορες καταστάσεις [8].

Η πληροφορία η οποία παίρνει το σύστημα μπορεί να έχει διάφορες μορφές. Μπορεί να είναι πληροφορίες κίνησης, ήχου, φωτός, τοποθεσίας του χρήστη, χρήσης της συσκευής, εφαρμογών που εκτελούνται στη συσκευή, προσώπων που βρίσκονται στον ίδιο χώρο με το χρήστη καθώς και άλλες μορφές. Οι πληροφορίες λαμβάνονται συνήθως μέσω αισθητήρων, π.χ. Wi-Fi, όπως επίσης και από την ίδια την συσκευή. Για να μπορέσουμε να κατανοήσουμε την κατάσταση του χρήστη, παίρνουμε συνήθως ως πληροφορίες πλαισίου χρήστη την ταυτότητα του, την θέση του ακόμα και την ώρα. Από αυτά μπορούμε να συμπεράνουμε δευτερεύοντες πληροφορίες όπως είναι το τηλέφωνο του χρήστη, η πόλη στην οποία βρίσκεται την συγκεκριμένη στιγμή καθώς και ο χρόνος που χρειάζεται για να μεταβεί από κάποιο κτίριο που βρίσκεται σε κάποιο άλλο [9].

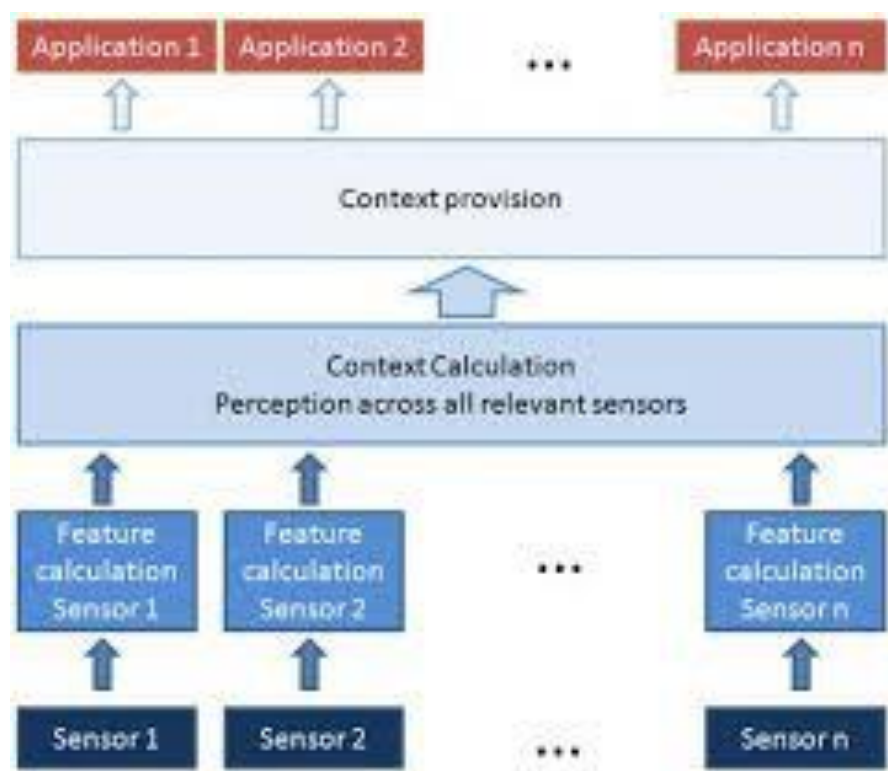
Η χρήση των εφαρμογών επίγνωσης πλαισίου αποτελεί ένα από τα καλύτερα μέσα αναπαράστασης της γνώσης. Είναι πολύ σημαντικό το ότι μέσα από ένα μεγάλο όγκο πληροφοριών που λαμβάνει το σύστημα μπορούμε να απομονώσουμε και να διαμοιράσουμε τη γνώση, κρατώντας μόνο τις πληροφορίες τις οποίες χρειαζόμαστε αφού παρέχεται στο σύστημα η ευκολία να βγάζει συμπεράσματα για την πληροφορία πλαισίου την οποία λαμβάνει. Η χρήση αυτών των εφαρμογών βοηθά το σύστημα να παίρνει πιο γρήγορα αποφάσεις και τους χρήστες να έχουν γρηγορότερες λύσεις.

Χρησιμοποιώντας εφαρμογές επίγνωσης πλαισίου είναι δυνατόν για το χρήστη να καταφέρει να αποκτήσει μια άλλη σχέση και γνώση με το περιβάλλον στο οποίο βρίσκεται, μέσα και έξω από την συσκευή. Οι αισθητήρες καταφέρνουν να δώσουν

αυτή την σύνδεση χρήστη περιβάλλοντος μετατρέποντας τις πληροφορίες που παίρνει σε αξιοποιήσιμα δεδομένα για τον χρήστη. Μπορεί να συνδυάσει πληθώρα πληροφοριών.

Οι ερευνητές υποστηρίζουν ότι στα επόμενα χρόνια η ανάπτυξη εφαρμογών με επίγνωση πλαισίου θα είναι βασικό δομικό υλικό για την υποστήριξη των συστημάτων των μεγαλύτερων εταιριών του κόσμου. Παρόλα αυτά αποτελεί μείζον θέμα αντιπαράθεσης κατά πόσον αυτή η μεθοδολογία ανάπτυξης συστημάτων διασφαλίζει την ιδιωτικότητα των χρηστών λόγω της χρήσης των προσωπικών δεδομένων τους.

Το παρακάτω σχήμα (Εικόνα 3) μας δίνει την μεθοδολογία με την οποία δουλεύει ένα σύστημα που χρησιμοποιεί επίγνωση πλαισίου και συνοψίζει τα προαναφερθέντα:



Εικόνα 3: Μεθοδολογία ανάπτυξης εφαρμογών επίγνωσης πλαισίου [10]

Κατά την εκπόνηση αυτής της διπλωματικής μελετήθηκαν κάποια άρθρα σχετικά με την μεθοδολογία ανάπτυξης εφαρμογών επίγνωσης πλαισίου. Μέσα αυτά τα άρθρα παρουσιάζεται η μορφή με την οποία αντιλαμβάνεται ο κόσμος της τεχνολογίας τους όρους context και context-aware [7] [10]. Από το 1992 και ύστερα εμφανίστηκαν πολλές απόψεις σύμφωνα με τον ορισμό του τί είναι context-aware (επίγνωση πλαισίου). Παλαιότερα όταν αναφερόταν κάποιος στον όρο context μπορούσε να υπονοηθεί μια τοποθεσία, ένα περιβάλλον ή μια κατάσταση. Οι διάφορες προσπάθειες που εφαρμόστηκαν για να κατανοηθεί ο όρος αυτός άρχισαν να εμφανίζουν αρκετές αντικρούσεις καθώς πολλές φορές δεν μπορούσε να αναφερθεί το context στις πιο πάνω εξηγήσεις που δόθηκαν για τον ορισμό.

Αργότερα, υπήρχε η ανάγκη για γενικοποίηση του ορισμού καθώς το context δεν μπορεί να είναι κάτι συγκεκριμένο. Ο ακριβής ορισμός που δόθηκε από τους αρθρογράφους, ο οποίος δίνει μια πιο κατανοητή και παράλληλα όχι τόσο συγκεκριμένη διάσταση στο context, αναφέρει ότι είναι οποιαδήποτε πληροφορία μπορεί να χρησιμοποιηθεί για να χαρακτηριστεί η κατάσταση μιας οντότητας. Ως οντότητα εννοούμε ένα άτομο, ένα χώρο ή ένα αντικείμενο που συσχετίζεται με την αλληλεπίδραση μεταξύ του χρήστη και της εφαρμογής.

Έγιναν αρκετές προσπάθειες και έρευνες για να εντοπιστούν τα σημαντικά χαρακτηριστικά των context-aware εφαρμογών. Οι έρευνες αυτές κατηγοριοποίησαν αυτά τα χαρακτηριστικά που μια context-aware εφαρμογή μπορεί να υποστηρίξει ως ακολούθως:

- Παρουσίαση πληροφορίας και υπηρεσιών σε ένα χρήστη
- Αυτόματη εκτέλεση μιας υπηρεσίας για ένα χρήστη
- Κατοχύρωση ετικέτων στην πληροφορία για μεταγενέστερη χρήση

Έτσι, με την πάροδο των χρόνων και την τριβή των ανθρώπων με το context-aware άρχισαν πλέον οι ειδικοί να αναζητούν τα χαρακτηριστικά της δικής τους εφαρμογής και να την αναπτύσσουν με τον σωστό τρόπο έτσι όπως έχει ειπωθεί.

Ένα παράδειγμα που έχει απασχολήσει τους ειδικούς είναι το σύστημα προώθησης κλήσης ενός γραφείου. Όταν ένας χρήστης λύπει από το γραφείο του και χτυπήσει το τηλέφωνο, το σύστημα πρέπει να είναι υπεύθυνο να εντοπίζει την τοποθεσία του και να προωθεί την κλήση στο κοντινότερο του τηλέφωνο ή στο κινητό του τηλέφωνο. Εάν το σύστημα ανιχνεύσει ότι ο χρήστης είναι σε συνεδρίαση τότε το τηλεφώνημα προωθείται στο φωνοκιβώτιο. Στη συγκεκριμένη εφαρμογή χρησιμοποιούνται δυο είδη context: το location context (τοποθεσία) και το activity context (δραστηριότητα χρήστη). Έχουμε δυο κατηγορίες διότι έχουμε δυο είδη πληροφορίας. [11]

Ακόμη ένα παράδειγμα είναι το σύστημα βοηθού για τα ψώνια. Ο χρήστης μπαίνει στο κατάστημα και ανοίγει το τηλέφωνο του. Στο τηλέφωνο παρουσιάζεται η πληροφορία για ένα αντικείμενο του καταστήματος. Το σύστημα αναλύει το προφίλ του χρήστη και του παρουσιάζει προτεινόμενα αντικείμενα που πιθανώς να τον ενδιαφέρουν. Και εδώ χρησιμοποιείται το location context (τοποθεσία) με την διαφορά ότι τώρα χρησιμοποιείται αντί του activity context το attribute context (χαρακτηριστικά χρήστη). [12]

Ένα τρίτο παράδειγμα είναι μια τουριστική εφαρμογή επίγνωσης πλαισίου, η οποία δίνει την δυνατότητα εντοπισμού της τοποθεσίας του χρήστη καθώς και τις προτιμήσεις του, ώστε να του προτείνει αξιοθέατα, εστιατόρια και πιθανές εκδηλώσεις τα οποία θα είναι ενδιαφέροντα για τον χρήστη. Εδώ η τουριστική εφαρμογή επίγνωσης πλαισίου επεξεργάζεται και προτείνει ή λαμβάνει αποφάσεις με βάση το προφίλ και την τοποθεσία του χρήστη [13].

Μέσα από τα πιο πάνω παραδείγματα μπορεί να κατανοήσει κανείς την σημαντικότητα των context-aware εφαρμογών καθώς και τον τρόπο με τον οποίο εισέβαλαν στη ζωή μας. Οι ανάγκες των ανθρώπων να αναλύουν και να

επεξεργάζονται το περιβάλλον στο οποίο βρίσκονται χωρίς να χρειάζεται μεγάλος κόπος έχουν προωθήσει τη σχεδίαση και υλοποίηση εφαρμογών context-aware. Μέσα από πολλά άλλα παραδείγματα μπορεί να δει κανείς ότι η πληροφορία που τυγχάνει ανάλυσης και επεξεργασίας δίνει αποτελέσματα και συμπεράσματα τα οποία δεν μπορούσε κανείς να φανταστεί ότι θα λάμβανε με τόση ευκολία.

Στη περίπτωση της εφαρμογής που αναπτύσσεται στην παρούσα διπλωματική, τα πιο πάνω στοιχεία που περιγράφηκαν μπορούν να υλοποιηθούν και να εφαρμοστούν ανάλογα με τις ανάγκες των χρηστών. Όπως αναφέρθηκε το context της εφαρμογής είναι το σημαντικότερο ούτως ώστε να συμπεριλιφθούν όλα τα στοιχεία που είναι σημαντικά τόσο για το χρήστη όσο και για την συσκευή αποζητώντας ένα αποτέλεσμα από κοινού. Ο χρήστης είναι ο παραλήπτης της τελικής πληροφορίας χωρίς οποιαδήποτε διαμεσολάβηση του, αφού το context στο οποίο βασιστήκε δεν προβλέπει οποιαδήποτε είσοδο πληροφορίας από τον χρήστη.

Το context της εφαρμογής επιτάσσει την εύρεση της τοποθεσίας του χρήστη διαμέσου του αισθητήρα GPS την παρουσίαση όλων των κοντινών σημείων αναφοράς και η παρουσίαση των διάφορων πληροφοριών για το συγκεκριμένο σημείο. Άρα, μπορούμε να πούμε ότι αναλύει το location context του χρήστη και εξάγει πληροφορίες σύμφωνα με αυτό.

2. ΤΕΧΝΟΛΟΓΙΑ ANDROID

2.1. Εισαγωγή στην Τεχνολογία Android

Το Android είναι μια στοίβα λογισμικού για κινητές συσκευές που έχει γίνει παγκόσμια διαδεδομένη και μπορούμε να πούμε χωρίς αμφιβολία ότι έχει ανέβει στη λίστα προτιμήσεων των χρηστών των κινητών συσκευών. Αποτελεί μια καινοτόμα πλατφόρμα ανάπτυξης λειτουργικού συστήματος με ένα ενδιάμεσο λογισμικό και κάποιες βασικές εφαρμογές, η οποία μέρα με την μέρα κερδίζει μεγαλύτερο μερίδιο στην αγορά των έξυπνων τηλεφώνων και tablets. Η δημοτικότητα που απόκτησε τον τελευταίο καιρό η πλατφόρμα, οφείλεται στην δυνατότητα που έδωσε στο χρήστη να μπορεί εύκολα και αποδοτικά να κάνει πράγματα με την κινητή του συσκευή τα οποία παλαιότερα θα θεωρούσε αδύνατον να συμβούν. Η ανάγκη του ανθρώπου να μπορεί να δουλεύει σε κινητούς χώρους εργασίας σε συνδυασμό με την υιοθέτηση του διαδικτύου στη ζωή μας, ανάγκασαν τους ειδικούς να σχεδιάσουν συσκευές οι οποίες να ανταποκρίνονται σε αυτές τις ανάγκες.

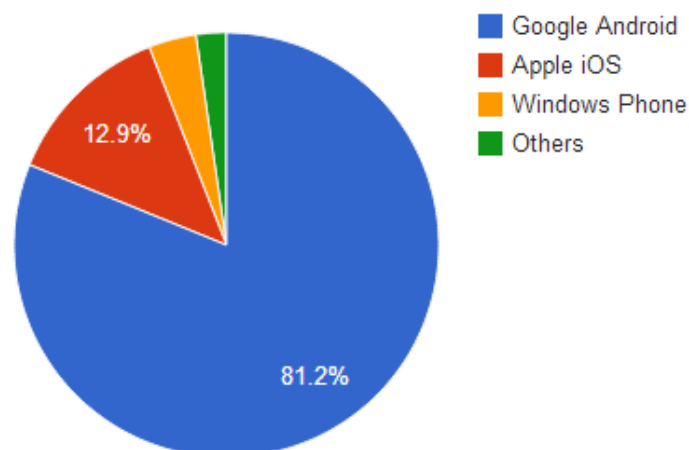
Η τεχνολογία ανάπτυξης εφαρμογών κινητών τηλεφώνων βρίσκεται στο τελικό στάδιο της μεγάλης αλλαγής που υπέστη καθώς μέσα σε λίγα χρόνια οι απαιτήσεις των καταναλωτών έχουν γίνει τεράστιες. Έτσι, με την ανάπτυξη της πλατφόρμας Android γίνονται πραγματικότητα πολλές εφαρμογές που ευκολύνουν την ζωή μας.

Η Android ακολουθώντας τον παλμό των καταναλωτών δημιούργησε μια πλατφόρμα «ανοιχτού κώδικα», μέσα από την οποία κάθε προγραμματιστής εφαρμογών έχει τη δυνατότητα να δημιουργήσει τη δική του εφαρμογή που να υποστηρίζεται από λειτουργικό σύστημα Android και συνεπώς ο ανταγωνισμός και η ποιότητα των υπηρεσιών που μπορεί να προσφέρει μια κινητή συσκευή πλέον δεν έχει καμία σχέση με τις δυνατότητες που παρέχονταν παλαιότερα στους χρήστες.

Το 2003 σπουδαίοι επιστήμονες στον χώρο της τεχνολογίας και πιο συγκεκριμένα οι Andy Rubin, Rich Miner, Nick Sears και Chris White αφουγκράστηκαν τις ανάγκες του τότε τεχνολογικού κόσμου και δημιούργησαν την εταιρία Android, που είχε σκοπό την ανάπτυξη ενός λειτουργικού συστήματος και μιας πλατφόρμας λειτουργικού συστήματος για κινητές συσκευές. Η Google βλέποντας τις μεγάλες προοπτικές που μπορούσε να χαρίσει στον κόσμο της τεχνολογίας το συγκεκριμένο λειτουργικό σύστημα, αγόρασε το 2005 την Android για 50 εκατομμύρια [14] [15] [16].

Σήμερα, το Android είναι η πιο διαδεδομένη πλατφόρμα για κινητές συσκευές. Έχει κατακλήσει τις αγορές αφού οι πλήστες εταιρίες κινητών συσκευών χρησιμοποιούν το λειτουργικό σύστημα Android. Πάνω από 480 εκατομμύρια άνθρωποι έχουν κινητή συσκευή Android και κάθε μέρα ενεργοποιούνται περίπου 1 εκατομμύριο καινούριες κινητές συσκευές που υποστηρίζουν Android λειτουργικό σύστημα. Οι ανταγωνιστές των Android έχουν χάσει μεγάλο έδαφος και αυτό φαίνεται και από την πιο κάτω γραφική παράσταση:

Mobile OS Market Share, Q3 2013



Εικόνα 4: Μερίδιο αγοράς λειτουργικών συστημάτων κινητών συσκευών για το τρίτο τεταρτημόριο του 2013 (Ιούλιος-Αύγουστος-Σεπτέμβριος) [17]

2.1.1. Πλεονεκτήματα

Όπως προαναφέρθηκε σήμερα παρακολουθείται ο όγκος των συσκευών που υποστηρίζουν αυτή την πλατφόρμα και των χρηστών τους να αυξάνονται με ραγδαίους και ασταμάτητους ρυθμούς. Η ανταπόκριση του κόσμου στις συσκευές που υποστηρίζουν αυτή την πλατφόρμα είναι δικαιολογημένη καθώς έχει τόσα πολλά πλεονεκτήματα που αναγκάζει τους ανθρώπους να γίνει μέρος της ζωής μας.

Είναι πολύ σημαντικό ότι με την χρήση της τεχνολογίας Android παρέχεται η δυνατότητα στον χρήστη να μπορεί να κάνει multitasking, δηλαδή να εκτελεί ταυτόχρονα πολλές ενέργειες. Αυτό γίνεται δυνατό με την μνήμη που μας προσφέρουν οι συσκευές και οι πολλαπλοί επεξεργαστές. Για παράδειγμα ένας χρήστης μπορεί ταυτόχρονα να ακούει μουσική, να διαβάζει το email του, να έχει ανοικτό κάποιο παιχνίδι καθώς και άλλες τόσες δυνατότητες που παρέχει η σύγχρονη τεχνολογία κινητών συσκευών.

Το συγκεκριμένο λειτουργικό σύστημα είναι φτιαγμένο για να υποστηρίζεται από πληθώρα συσκευών από διαφορετικές εταιρείες που έχουν πολλές φορές διαφορετικά μηχανικά χαρακτηριστικά και οι ίδιες οι εταιρίες αυτές διαφοροποιούν το λειτουργικό σύστημα για να συμβαδίζουν με τις δικές τους προδιαγραφές.

Με συσκευές χτισμένες στην πλατφόρμα Android οι χρήστες έχουν την δυνατότητα να προσαρμόζουν πλήρως τη συσκευή τους ανάλογα με τις ανάγκες και τις απαιτήσεις τους. Η Google δίνει την δυνατότητα στους χρήστες της να μπορούν να τροποποιήσουν την κινητή τους συσκευή και έχοντας το Google λογαριασμό τους ως δείκτη αποθήκευσης και εγκατάστασης εφαρμογών να μπορούν να μεταφέρουν τις επιλογές τους σε άλλη συσκευή μέσω αυτού.

Είναι πολύ σημαντικό ότι ο χρήστης μπορεί να βρει όλες τις εφαρμογές που είναι σχεδιασμένες για να λειτουργήσουν πάνω σε πλατφόρμα Android μέσα στο Google

Play και να κατεβάσει όσες εφαρμογές θέλει όπου καλύπτονται από την χωρητικότητα της συσκευής. Υπάρχουν πάρα πολλές εφαρμογές οι οποίες προσφέρονται δωρεάν και αυτό δίνει την δυνατότητα στο χρήστη να κερδίσει όσα πιο πολλά μπορεί από την κινητή του συσκευή. Αυτό θα μπορούσε να είναι μια εφαρμογή πλοήγησης, ένα παιχνίδι, ένα λεξικό και άλλες τόσες χρήσιμες εφαρμογές που ευκολύνουν και εμπλουτίζουν την ζωή του χρήστη.

Η διαφορά του Android λογισμικού με τους ανταγωνιστές του μέσα από την μοναδικότητα που το χαρακτηρίζει, δίνει την δυνατότητα σε κάθε ενδιαφερόμενο μέσω ενός SDK (SOFTWARE DEVELOPMENT KIT) να δημιουργήσει την δική του εφαρμογή και το κόστος σε τέτοιου είδους ανάπτυξη εφαρμογών είναι αρκετά χαμηλό. Στο διαδίκτυο παρέχονται αμέτρητες πληροφορίες για την ανάπτυξη εφαρμογών Android και αυτό δίνει την ευκαιρία στον καθένα να αναπτύξει μια εφαρμογή ακριβώς όπως αυτός την έχει φανταστεί. Αυτό επεκτείνει μέρα με την μέρα των μικρόκοσμο των Android εφαρμογών και τις επιλογές των χρηστών και δείχνει την μελλοντική συνεχή πρόοδο.

Για να μπορέσει κάποιο λογισμικό να χαρακτηριστεί αξιόπιστο θα πρέπει να έχει δοκιμαστεί πολλές φορές στο παρελθόν και να έχει ένα μεγάλο ποσοστό επιτυχίας στη χρήση του. Το Android αν και είναι σχετικά καινούριο στις αγορές αποτελεί μια αξιόπιστη λύση καθώς ο πυρήνας του είναι βασισμένος στον πολύ γνωστό και αρκετά διαδεδομένο πυρήνα Linux και έχει κληρονομήσει τις αρετές και την εμπειρία του.

Συνεπώς, μπορούμε να αναφερόμαστε στο Android ως ένα αξιόπιστο λειτουργικό σύστημα, αρκετά διαδεδομένο και με τεράστιες προοπτικές επέκτασης και περαιτέρω ανέλιξης [18] [19] [20].

2.1.2. Μειονεκτήματα

Η διάδοση της τεχνολογίας Android ευκολύνει φυσικά την ζωή των ανθρώπων αλλά καμιά τεχνολογία δεν μπορεί να είναι η ιδανική.

Το σημαντικότερο μειονέκτημα έγκειται στο ότι οι εφαρμογές που δημιουργούνται δεν ελέγχονται αρχικώς από την Google ούτως ώστε να εκδοθεί κάποια άδεια πριν να διατεθεί μέσω του Google Play Store. Υπάρχει φυσικά έλεγχος, μια και το Google Play Store παρακολουθείται και να ελέγχεται για δυνητικά επικίνδυνες εφαρμογές ή δόλια προγράμματα. Μερικές φορές κακόβουλες εφαρμογές έχουν γλιστρήσει από τον έλεγχο και έχουν παραμείνει προς διάθεση. Ένα ψεύτικο BBM app εμφανίστηκε πρόσφατα στο κατάστημα Google Play και κατάφερε να εξασφαλίσει πάνω από 100.000 downloads πριν να αφαιρεθεί. Η ίδια η εφαρμογή δεν ήταν τίποτα περισσότερο από μια υπηρεσία spamming. Άρα όταν κατεβάζουμε εφαρμογές είναι επιτακτική ανάγκη να το κάνουμε μόνο από ένα νόμιμο κατάστημα, π.χ. Google Play. Αυτό σημαίνει ότι υπάρχει ο καλύτερος δυνατός έλεγχος από εταιρείες όπως η Google, η Amazon, η Samsung, ή άλλος μεγάλος κατασκευαστής ή φορέας.

Σε σχέση με το πιο πάνω η Android, ως μια τεχνολογία «ανοικτού κώδικα», αφήνει ανοικτό το ενδεχόμενο οποιοσδήποτε προγραμματιστής να ενεργήσει κακόβουλα και

να προσθέσει στο Google Play κάποια εφαρμογή που να εκμεταλεύεται τα προσωπικά δεδομένα του χρήστη. Έτσι, μπορούμε να συμπεράνουμε ότι δεν είναι δυνατόν να υπάρξει πλήρης έλεγχος από την Google, ειδικά για θέματα ασφάλειας δεδομένων, ώστε να προστατεύει τα προσωπικά στοιχεία του χρήστη καθώς και την ίδια την συσκευή.

Επίσης, για να λειτουργήσει η συγκεκριμένη τεχνολογία σε τόσο χαμηλά κόστη προϋποθέτει ότι πρέπει να βρει πόρους από αλλού. Έτσι, ένας προγραμματιστής που έχει υλοποιήσει και παρέχει την εφαρμογή του δωρεάν πιθανόν να χρησιμοποιήσει κάποιες διαφημίσεις για οικονομικό όφελος ή να παρέχει μια έκδοση της εφαρμογής με περιορισμένες ιδιότητες δωρεάν και μια με περισσότερες δυνατότητες με χρέωση. Για το πρώτο παράδειγμα έχουμε πολλές εφαρμογές όπου ένας χρήστης κάποιας συσκευής που υποστηρίζεται από Android λειτουργικό σύστημα θα έχει ενοχληθεί από τις διαφημίσεις της εφαρμογής. Είναι φυσικό επακόλουθο ότι οι πόροι πρέπει να προέλθουν από τις διαφημίσεις και ίσως να είναι κάτι το αναπόφευκτο, ωστόσο αυτό μπορεί να αναφερθεί ως ένα μειονέκτημα της χρήσης της τεχνολογίας Android.

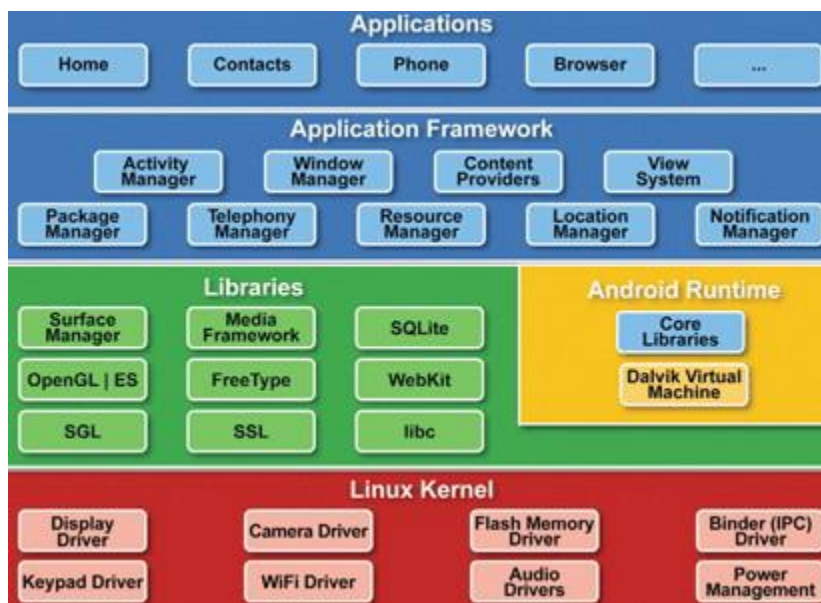
Ένα λειτουργικό σύστημα το οποίο έχει την δυνατότητα να τρέχει σε πληθώρα διαφορετικών συσκευών προφανώς του δίνει ένα μεγάλο πλεονέκτημα αλλά αν κανείς αναζητήσει να δει και την μεριά του προγραμματιστή αυτό προσθέτει επιπλέον δυσκολία στην ανάπτυξη των εφαρμογών καθώς πρέπει να ληφθούν υπόψιν κατά τη σχεδίαση της διεπαφής χρήστη την μεγάλη ποικιλία μεγεθών και αναλύσεων που μπορεί η εφαρμογή να συναντήσει. Αυτό το μειονέκτημα φυσικά ισχύει για τις πλείστες κινητές τεχνολογίες προγραμματισμού.

Με την πάροδο των χρόνων η τεχνολογία ανοίγει τους ορίζοντες της με γοργούς ρυθμούς σε σημείο που να εντυπωσιάζει και τον πιο αισιόδοξο. Μπορούμε να πούμε με ασφάλεια ότι τα βήματα που γίνονται καθημερινά είναι προοδευτικά. Άρα στις πλείστες των περιπτώσεων τα αρνητικά της τεχνολογίας Android διορθώνονται και βελτιστοποιούνται οι όποιες ελλείψεις υπάρχουν [18] [20].

2.2. Η Αρχιτεκτονική της Πλατφόρμας Android

Το Android είναι μια ανοιχτή πλατφόρμα που υποστηρίζεται από κινητές συσκευές και περιλαμβάνει ένα λειτουργικό σύστημα, ένα ενδιάμεσο λογισμικό, βιβλιοθήκες και βασικές εφαρμογές. Το Android έχει 5 βασικά συστατικά που το δομούν.

Αποτελείται από τον πυρήνα Linux Kernel, τον Χρόνο Εκτέλεσης (Android Runtime), τις Βιβλιοθήκες, το πλαίσιο Εφαρμογής και στην κορυφή της στοίβας βρίσκονται οι εφαρμογές που αλληλεπιδρούν άμεσα με το χρήστη. Εφαρμογές για αυτή την πλατφόρμα μπορούν να αναπτυχθούν χρησιμοποιώντας το Android SDK και την γλώσσα προγραμματισμού JAVA (για την ακρίβεια είναι μια έκδοση για κινητές συσκευές με βάση την Java). Με βάση αυτά τα χαρακτηριστικά, το Android αποτελείται από συνιστώσες λογισμικού οι οποίες συνθέτουν ένα ενιαίο και ολοκληρωμένο σύστημα [21]. Το σχεδιάγραμμα παρακάτω παρουσιάζει επακριβώς την αρχιτεκτονική των Android με όλα τα συστατικά του:



Εικόνα 5: Αρχιτεκτονική Android [22]

2.3. Επιμέρους συστατικά της στοίβας του Android

2.3.1. Linux Kernel

Το λειτουργικό σύστημα τρέχει στον πυρήνα του Linux. Η Google χρησιμοποιεί μια τροποποιημένη μορφή του πυρήνα του Linux για τα Android. Παρόλο που δεν είναι ο αυθεντικός πυρήνας ωστόσο έχει κληρονομήσει όλα τα βασικά χαρακτηριστικά του πυρήνα Linux μπορεί να θεωρηθεί και αυτός και το λειτουργικό σύστημα αξιόπιστο. Ο πυρήνας αυτός λειτουργεί ως ένα ενδιάμεσο επίπεδο μεταξύ της στοίβας λογισμικού και του υλικού, δηλαδή μπορεί να τον χαρακτηρίσει κάποιος ως τον συνδετικό κρίκο αυτών των δύο. Ο πυρήνας Linux Kernel δίνει τη βασική λειτουργικότητα στο σύστημα όπως διαχείριση επεξεργασιών, διαχείριση μνήμης και διαχείριση των περιφερειακών μερών της συσκευής καθώς είναι αυτός που προσδίδει και ασφάλεια στο σύστημα.

Για τις πιο πάνω λειτουργίες έχουν τροποποιηθεί αρκετά τα πρότυπα για τα Android καθώς ο σχεδιασμός για τα Android αποσκοπεί σε κινητές συσκευές. Μια τόσο μεγάλη αλλαγή εμπεριέχει αρκετούς περιορισμούς και κατά κύριο λόγο μπορούμε να αναφερθούμε στους περιορισμένους πόρους που μπορεί να έχει μια κινητή συσκευή

σε αντίθεση με ένα υπολογιστή. Ο πυρήνας είναι υπεύθυνος στο να ελέγχει την σωστή λειτουργία των διάφορων drivers της συσκευής. Τα drivers αυτά μπορεί να είναι το driver οθόνης, της κάμερας, του Wi-Fi, του ήχου, του πληκτρολογίου, drivers διαχείρισης ενέργειας.

Έτσι, όταν μια εταιρία αποφασίσει να κυκλοφορήσει μια καινούρια συσκευή που να υποστηρίζεται από πλατφόρμα Android το μόνο που έχει να κάνει είναι να προσαρμόσει τα ενσωματωμένα drivers του Android ώστε να ταιριάζουν με τα περιφερειακά μέρη της συσκευής του. Μια σημαντική ιδιότητα του πυρήνα των Android είναι ότι το Linux είναι ένας μονολιθικό Kernel, δηλαδή όλο το λειτουργικό σύστημα δουλεύει μόνο στον χώρο του πυρήνα και είναι μόνος του σε θέση επιτηρητή.

2.3.2. Βιβλιοθήκες

Το επόμενο επίπεδο στην αρχιτεκτονική είναι οι βιβλιοθήκες που περιλαμβάνει η πλατφόρμα των Android και είναι γραμμένες σε γλώσσα προγραμματισμού C/C++. Οι βιβλιοθήκες αυτές είναι ενσωματωμένες και παρέχουν κάποια σταθερά συστατικά που μπορούν να δώσουν χαρακτήρα και συμπεριφορά σε μια εφαρμογή και είναι ευέλικτες στην επαναχρησιμοποίηση κώδικα. Όλες οι βιβλιοθήκες των Android τρέχουν στον πυρήνα του Linux. Οι βιβλιοθήκες αυτές δεν είναι εφαρμογές που μπορούν να σταθούν από μόνες τους. Υπάρχουν για να μπορούν να κληθούν από προγράμματα υψηλότερου επιπέδου.

Οι εφαρμογές που τρέχουν σε μια κινητή συσκευή μπορούν να έχουν πρόσβαση στις βιβλιοθήκες αυτές μέσω του Dalvik JVM. Παραδείγματος χάριν, η SQLite βιβλιοθήκη παρέχει υποστήριξη έτσι ώστε μια εφαρμογή να χρησιμοποιήσει την αποθήκευση δεδομένων και η βιβλιοθήκη Webkit παρέχει λειτουργίες για την διαδικτυακή πλοήγηση. Βασικές βιβλιοθήκες όπως η libc και η lirm αναπτύχθηκαν ειδικά για χαμηλή κατανάλωση ενέργειας του Android. Η βιβλιοθήκη Surface Manager βοηθά στη δημιουργία παραθύρων καθώς και δυσδιάστατων και τρισδιάστατων γραφικών. Η βιβλιοθήκη Media Framework περιέχει αποκωδικοποιητές για αναπαραγωγή αρχείων πολυμέσων, όπως εικόνες, βίντεο(MPEG, MP3).

Μέσα από τις βιβλιοθήκες παρέχονται όλα τα απαραίτητα συστατικά στοιχεία για να υπάρξει η δυνατότητα υλοποίησης του Android Runtime που είναι ο απαραίτητος μηχανισμός για να μπορούν να αναπτυχθούν και να εκτελεστούν οι εφαρμογές των Android.

2.3.3. Χρόνος Εκτέλεσης-Android Runtime

Στο ίδιο επίπεδο με τις βιβλιοθήκες βρίσκεται και ο χρόνος εκτέλεσης Android. Το Android Runtime αποτελείται από δύο βασικές συνιστώσες. Η πρώτη συνιστώσα είναι οι βασικές βιβλιοθήκες για την αλληλεπίδραση των εφαρμογών JAVA με το περιβάλλον της συσκευής στην οποία εκτελούνται και η δεύτερη συνιστώσα είναι η Dalvik Virtual Machine η οποία είναι υπεύθυνη για την δημιουργία των εκτελέσιμων

αρχείων των εφαρμογών προκειμένου να μπορεί να εκκινήσει το λειτουργικό σύστημα.

Η Dalvik είναι η υλοποίηση της JAVA από την Google, βελτιστοποιημένη για φορητές συσκευές με χαμηλές απαιτήσεις μνήμης. Έτσι, αντί να τρέχουν τα προγράμματα αυτά σε Java Virtual Machine τρέχουν σε δική τους μηχανή (Dalvik Virtual Machine), υπεύθυνη για φορητές συσκευές Android. Οι βιβλιοθήκες πυρήνα και η Dalvik Virtual Machine αποτελούν την κινητήρια δύναμη των εφαρμογών και μαζί με τις βιβλιοθήκες του Android αποτελούν τη βάση για το πλαίσιο εφαρμογών.

2.3.4. Πλαίσιο Εφαρμογής-Application Framework

Στο παραπάνω επίπεδο της στοίβας της αρχιτεκτονικής των Android βρίσκεται το επίπεδο εφαρμογής το οποίο είναι μια ανοικτή πλατφόρμα ανάπτυξης εφαρμογών και είναι υπεύθυνο να παρέχει τις απαραίτητες δομικές μονάδες οι οποίες μπορούν να χρησιμοποιηθούν για την κατασκευή των εφαρμογών. Αυτό το επίπεδο παρέχει πολλές δυνατότητες στο σύστημα αφού μέσω αυτού δίνεται προσβασιμότητα στα περιφερειακά μέρη της συσκευής καθώς και στις λειτουργικότητες του ίδιου του λειτουργικού συστήματος. Οι εφαρμογές έχουν πρόσβαση στις βασικές βιβλιοθήκες του λειτουργικού συστήματος μέσω κατάλληλων διεπαφών και έτσι με το πλαίσιο εφαρμογής μπορούν να προσφέρουν επιπρόσθετες υπηρεσίες προς άλλες εφαρμογές και επίσης δίνεται η δυνατότητα στο χρήστη να μπορεί να αλλάξει τα συστατικά κάθε εφαρμογής.

Αυτό το πλαίσιο είναι προ-εγκατεστημένο στο Android ωστόσο μπορεί να τροποποιηθεί από κάποιο κατασκευαστή και να το προσαρμόσει ανάλογα με τις ανάγκες και τις προδιαγραφές του. Από τα πιο σημαντικά δομικά στοιχεία του πλαισίου εφαρμογής είναι ο Διαχειριστής Δραστηριοτήτων (Activity Manager) ο οποίος είναι υπεύθυνος για τον έλεγχο της διάρκειας ζωής των εφαρμογών. Ο Παροχέας Περιεχομένου (Content Provider) έχει κάποια αντικείμενα τα οποία μπορούν να διαμοιραστούν μεταξύ των εφαρμογών που τρέχουν στην συσκευή. Ο Διαχειριστής Πόρων (Resource Manager) αποτελεί ακόμη ένα επιμέρους συστατικό του πλαισίου εφαρμογής και ελέγχει τους πόρους της συσκευής, δηλαδή οτιδήποτε υπάρχει σε ένα πρόγραμμα και δεν είναι κώδικας, για παράδειγμα μπορεί να είναι κωδικοί χρωμάτων, αλφαριθμητικοί χαρακτήρες ή ακόμα και έτοιμα σχεδιαγράμματα οθονών φτιαγμένα σε XML.

2.3.5. Εφαρμογές

Στο πιο ψηλό επίπεδο της στοίβας βρίσκονται οι ίδιες οι εφαρμογές οι οποίες αποτελούν την πραγματική απεικόνιση του όλου σχεδιασμού και είναι αυτό που παρουσιάζεται και κατανοεί ο χρήστης στον οποίο απευθύνεται η συγκεκριμένη εφαρμογή. Όλες οι διεργασίες που εφαρμόζονται στα προηγούμενα επίπεδα είναι στην διάθεση των εφαρμογών. Ο τελικός χρήστης έχει στα χέρια του την εφαρμογή χωρίς να γνωρίζει την όλη διαδικασία σχεδίασης της. Ως εφαρμογές θα μπορούσαμε

να χαρακτηρίσουμε τις εφαρμογές που υλοποιούν οι κατασκευαστές λογισμικού καθώς και αυτές που είναι ήδη εγκατεστημένες στην συσκευή. Ουσιαστικά εφαρμογή είναι το μέσο με το οποίο ένας χρήστης καταφέρνει να αλληλεπιδράσει με το όλο σύστημα και του παρέχονται σαν αποτέλεσμα πληροφορίες σε μορφή που μπορεί να κατανοήσει.

3. ΕΝΔΙΑΜΕΣΑ ΛΟΓΙΣΜΙΚΑ

3.1. Εισαγωγή στα Ενδιάμεσα Λογισμικά Επίγνωσης Πλαισίου

Η δημιουργία εφαρμογών, κινητών και μη, οι οποίες μπορούν να συνδεθούν με οποιοδήποτε δίκτυο, ελλοχεύουν τον κίνδυνο πολλών λαθών και σφαλμάτων. Για αυτό το λόγο δημιουργήθηκε η ανάγκη για την δημιουργία λογισμικού που θα υποστηρίζει το υπάρχον σύστημα και θα απομακρύνει οποιαδήποτε πιθανότητα λάθους, διευκολύνοντας τους προγραμματιστές εφαρμογών να σχεδιάζουν τα προγράμματα τους με μεγάλη επιτυχία.

Το λογισμικό αυτό ονομάζεται ενδιάμεσο λογισμικό γιατί βρίσκεται μεταξύ των διάφορων επιπέδων της εφαρμογής και έχει τη δυνατότητα να διαχειρίζεται τα διάφορα μέρη της. Το ενδιάμεσο λογισμικό χρησιμοποιείται για να παρέχει υπηρεσίες σε εφαρμογές διαφορετικές από αυτές που του παρέχει το σύστημα. Μπορεί να χαρακτηριστεί και ως ο συνδετικός κρίκος μεταξύ συστήματος και εφαρμογής για να κάνει ευκολότερη την είσοδο και την έξοδο πληροφορίας από και προς αυτήν, αφού αποτελεί ένα επίπεδο επικοινωνίας το οποίο επιτρέπει στις εφαρμογές να αλληλεπιδρούν μεταξύ τους διαμέσου του εξοπλισμού και του περιβάλλοντος του. Ουσιαστικά είναι ένα στρώμα λογισμικού πάνω από το λειτουργικό σύστημα το οποίο μπορεί να λάβει πληροφορίες από τα συστατικά του συστήματος τα οποία μπορεί να βρίσκονται σκορπισμένα σε διαφορετικούς υπολογιστές οι οποίοι είναι συνδεδεμένοι μέσω κάποιου δικτύου και να διαχειρίζεται τις πληροφορίες που παρέχουν.

Το ενδιάμεσο λογισμικό αποτελείται από αντικείμενα και διαδικασίες που επικοινωνούν μεταξύ τους ώστε να υποστηρίζεται η επικοινωνία και ο διαμοιρασμός πόρων μεταξύ κατανεμημένων εφαρμογών κρύβοντας τις λεπτομέρειες του λειτουργικού συστήματος. Ενώνει τους παροχείς υπηρεσιών και τους χρήστες των υπηρεσιών. Το ενδιάμεσο λογισμικό ικανοποιεί την ανάγκη για διασυνδεσιμότητα δικτύων μέσω Internet και για την ανταλλαγή δεδομένων σε πραγματικό χρόνο.

Στα συστήματα τα οποία υποστηρίζουν την λειτουργία τους μέσα από κινητές συσκευές είναι απαραίτητο το ενδιάμεσο λογισμικό να γνωρίζει τη θέση του συστήματος αλλά και να βρίσκει διόδους μέσα από τις οποίες να γίνεται εφικτή η επικοινωνία μεταξύ των στρωμάτων του συστήματος. Έτσι, το ενδιάμεσο λογισμικό οφείλει να επεξεργάζεται τις πληροφορίες που παίρνει από τα κατώτερα επίπεδα του συστήματος και να μην τις αφήνει ανεκμετάλλευτες. Έχει, επίσης, ρόλο συντονιστή αφού καθορίζει τον χρονοπρογραμματισμό των λειτουργιών του συστήματος, βάζοντας προτεραιότητα στις λειτουργίες όπου θεωρεί ότι είναι άμεση ανάγκη η εκτέλεση τους.

Για να επιτρέπεται στις εφαρμογές να προσαρμοστούν στην ετερογένεια των συσκευών που τις υποστηρίζουν και στο περιβάλλον του χρήστη, τα συστήματα πρέπει να δίνουν την δυνατότητα στις κινητές εφαρμογές να γνωρίζουν το πλαίσιο στο οποίο χρησιμοποιούνται. Συνεπώς, η λειτουργία κάποιου ενδιάμεσου

λογισμικού επίγνωσης πλαισίου έχει κάποια ιδιαίτερα χαρακτηριστικά που το κάνουν να ξεχωρίζει.

Η αρχή της αντανάκλασης, η οποία χρησιμοποιείται για να επιτρέψει τη δυναμική αναδιαμόρφωση του ενδιάμεσου λογισμικού και έχει αποδειχθεί χρήσιμη στο να παρέχει επίγνωση πλαισίου. Η αρχή αυτή επιτρέπει σε ένα πρόγραμμα να έχει πρόσβαση, λόγο και δυνατότητα αλλαγής της σύστασης του. Οι λεπτομέρειες υλοποίησης κάποιας εφαρμογής επίγνωσης πλαισίου είναι κρυμμένες από τους χρήστες αλλά και από τους σχεδιαστές εφαρμογών και είναι εγκλεισμένες μέσα στο ενδιάμεσο λογισμικό. Αυτό μπορεί να επιφέρει κινδύνους κατά την υλοποίηση εφόσον όλη η πολυπλοκότητα χειρίζεται εσωτερικά από το ενδιάμεσο λογισμικό. Σε μια τέτοια περίπτωση το ενδιάμεσο λογισμικό λαμβάνει αποφάσεις για την εφαρμογή χωρίς να μπορεί η ίδια η εφαρμογή να αλλάξει την όποια επιλογή πάρθηκε.

Ένα ανακλαστικό σύστημα μπορεί να κάνει τροποποιήσεις στο ίδιο μέσω της αρχής της επιθεώρησης και/ή της προσαρμογής. Ένα σύστημα μπορεί μέσω της επιθεώρησης να εκθέσει την εσωτερική του συμπεριφορά έτσι ώστε να καθίσταται εύκολο να προσθέσει επιπλέον συμπεριφορά για να παρακολουθείται η υλοποίηση του ενδιάμεσου λογισμικού. Μέσω της προσαρμογής, η εσωτερική συμπεριφορά του συστήματος μπορεί να αλλάξει δυναμικά, τροποποιώντας τα υφιστάμενα χαρακτηριστικά του ή προσθέτοντας καινούρια. Αυτό σημαίνει ότι μπορεί να εγκατασταθεί σε κάποια κινητή συσκευή ένα ελάχιστο σύνολο λειτουργικών δυνατοτήτων και τότε η εφαρμογή είναι υπεύθυνη να παρακολουθεί και να προσαρμόζει τις δυνατότητες τις σύμφωνα με τις ανάγκες της.

Οι δυνατότητες που ανοίγονται από αυτή την προσέγγιση είναι πολλές. Μπορεί να δημιουργηθεί ένα όχι τόσο βαρυφορτομένο ενδιάμεσο λογισμικό. Οι πληροφορίες πλαισίου μπορούν να διατηρηθούν από το ενδιάμεσο λογισμικό στις εσωτερικές δομές δεδομένων του και οι εφαρμογές μπορούν να αποκτήσουν πληροφορίες σχετικά με το πλαίσιο εκτέλεσης τους και να συντονίσουν τη συμπεριφορά του ενδιάμεσου λογισμικού αναλόγως.

Τα συστήματα τα οποία είναι όμοια με αυτό που εξετάζεται στην παρούσα διπλωματική, δηλαδή συστήματα που σχετίζονται με την τοποθεσία του χρήστη χρησιμοποιώντας μια κινητή συσκευή (Location-aware), έχουν ως βασική ιδέα να προσφέρουν την δυνατότητα να αλλάζουν τη συμπεριφορά του ενδιάμεσου λογισμικού και των εφαρμογών βασιζόμενοι στη γνώση του πλαισίου το οποίο αλλάζει συνεχώς.

Εν κατακλείδι, μπορούμε να πούμε ότι οι εφαρμογές επίγνωσης πλαισίου είναι μια καινοτόμα μέθοδος ανάπτυξης λογισμικού, μια και οι εφαρμογές στηρίζονται κυρίως στο ενδιάμεσο λογισμικό που τις υποστηρίζει και μελλοντικά μπορούν να επιφέρουν τεράστιες αλλαγές στον κόσμο της τεχνολογίας, καθώς μπορούν να προσαρμόζονται στα διάφορα περιβάλλοντα σε πραγματικό χρόνο [23] [24].

3.1.1. Πλεονεκτήματα

Η υλοποίηση ενός συστήματος που χρησιμοποιεί ενδιάμεσο λογισμικό προϋποθέτει ότι το σύστημα είναι κατανοημένο και οι πόροι του διασκορπισμένοι. Αυτό πολλές φορές λειτουργεί υπέρ του σχεδιαστή καθώς είναι πιο εύκολο και αυξάνονται οι δυνατότητες υλοποίησης.

Υπάρχουν πολλά πλεονεκτήματα που μπορεί να μας προσφέρει αυτή η τεχνολογία και πολλοί λόγοι που θα θέλαμε να χρησιμοποιήσουμε κάποιο ενδιάμεσο λογισμικό κατά την σχεδίαση κάποιας εφαρμογής ή κάποιου συστήματος. Τα κυρίως πλεονεκτήματα είναι η επαναχρησιμοποίηση, η επεκτασιμότητα και η διαλειτουργικότητα.

Είναι πολύ σημαντικό το ότι παρέχεται πρόσβαση σε πληροφορίες μεταξύ συστημάτων σε πραγματικό χρόνο. Αυτό επιτρέπει στο χρήστη να λαμβάνει τα απαραίτητα δεδομένα που χρειάζεται την στιγμή που είναι χρήσιμα για την εφαρμογή. Δεν εξαρτάται από τον χρήστη ποια είναι η πληροφορία η οποία θα λάβει αλλά το ενδιάμεσο λογισμικό είναι αυτό που την καθορίζει.

Επίσης, είναι πάρα πολύ σημαντικό για την οργάνωση εταιριών αφού η ύπαρξη συστημάτων όπου οι πόροι είναι διασκορπισμένοι σε διαφορετικές τοποθεσίες επιβάλλει την χρήση ενός ενδιάμεσου λογισμικού για συλλογή και ομοιογένεια των πληροφοριών που λαμβάνει από τους πόρους αυτούς. Προστίθεται σταθερότητα μεταξύ των συστημάτων αυτών και μειώνεται το βάρος του δικτύου. Μπορεί να χρησιμοποιηθεί ανάμεσα σε πληθώρα λειτουργικών συστημάτων κάνοντας τη συνδεσιμότητα ακόμα καλύτερη.

Η χρήση του ενδιάμεσου λογισμικού πάνω από όλα δίνει ένα πολύ σημαντικό πλεονέκτημα κάνοντας δυνατή την επαναχρησιμοποίηση λογισμικού. Στις μέρες μας έχει γίνει μεγάλη στροφή και η παραγωγή λογισμικού δεν γίνεται πλέον με την ανάπτυξη νέων στοιχείων αλλά με βάση την επαχρησιμοποίηση στοιχείων έχοντας σαν αποτέλεσμα πολλά οφέλη.

Αρχικά, μπορεί να αναφερθεί το γεγονός ότι η επαναχρησιμοποίηση κάποιου ενδιάμεσου λογισμικού προσφέρει φερεγγυότητα καθώς έχει δοκιμαστεί από ήδη υπάρχοντα συστήματα που βρίσκονται σε λειτουργία και λογικά θα έχουν εντοπιστεί και διορθωθεί τα περισσότερα ή τουλάχιστον τα πιο σημαντικά λάθη που έχουν προκύψει. Έτσι, η επαναχρησιμοποίηση λογισμικού απαλείφει τους σχεδιαστές και τους προγραμματιστές από την διαδικασία αυτή. Αντίθετα, η δημιουργία καινούριων στοιχείων μπορεί να τους βάλει σε μπελάδες αφού θα έπρεπε να γίνει αναζήτηση για σφάλματα και λάθη και να επιλυθούν.

Ακόμη, το κόστος του υπάρχον λογισμικού είναι γνωστό εκ των προτέρων ποιο είναι ενώ η δημιουργία ενός καινούριου λογισμικού δεν εγγυάται από την αρχή μέχρι που θα φτάσει το κόστος του και έτσι εμπεριέχει ένα ποσοστό ρίσκου. Όταν γίνεται επαναχρησιμοποίηση ολόκληρων υποσυστημάτων τότε μειώνεται εξαιρετικά το περιθώριο λάθους στην εκτίμηση του κόστους. Οι σχεδιαστές αποφεύγουν την συνεχή επανάληψη καταγραφής της γνώσης τους καθώς με την

επαναχρησιμοποίηση παραθέτουν μια φορά την σχεδίαση τους και στη συνέχεια μπορούν απλά να πάρουν τα όσα έχουν σχεδιάσει ακριβώς όπως είναι με τις πιθανές διορθώσεις και βελτιώσεις που έχουν γίνει [2].

Πολλές φορές είναι χρήσιμο να σχεδιάζονται πρότυπα, όπως για παράδειγμα τα πρότυπα διασύνδεσης χρήστη, τα οποία να είναι υλοποιημένα ως ένα σύνολο επαναχρησιμοποιήσιμων συστατικών στοιχείων. Αυτό μειώνει και την πιθανότητα λάθους του χρήστη καθώς με την χρήση προτύπων ο χρήστης θα έχει συμμορφωθεί με αυτή τη σχεδίαση. Σε γενικές γραμμές η επαναχρησιμοποίηση λογισμικού συνεισφέρει στην γρηγορότερη κυκλοφορία ενός συστήματος στην αγορά.

Υπάρχει η ανάγκη μερικές φορές για άμεση υλοποίηση κάποιου συστήματος και αυτό μπορεί να αποτελεί την πρώτη προτεραιότητα για την εταιρία ανάπτυξης. Ωστόσο, σε πολύ μεγάλα συστήματα μπορεί να προκύψουν λάθη τα οποία κανένας δεν τα είχε προβλέψει από πριν. Έτσι, αποτελεί σημαντική ανάγκη το σύστημα να μπορεί να είναι επεκτάσιμο ούτως ώστε να μην καταλήξουν οι σχεδιαστές σε πολλά προβλήματα μετά από μακροχρόνια χρήση κάποιας εφαρμογής ή και συστήματος.

Το ενδιάμεσο λογισμικό μπορεί να προσφέρει την ευκολία αυτή καθώς μπορεί να συνδέσει πολλούς διαφορετικούς πόρους και δεν είναι εξαρτημένο από οποιοδήποτε επιμέρους συστατικό του συστήματος. Έτσι, όταν προκύψει ανάγκη για επέκταση του συστήματος το ενδιάμεσο λογισμικό είναι αυτό που επιτρέπει να λυθούν τέτοια προβλήματα.

Επίσης, το ενδιάμεσο λογισμικό παρέχει τη δυνατότητα να ελέγχεται η τιμή ή οποιαδήποτε πληροφορία δίνεται στο χρήστη, γνωρίζοντας ότι αυτή η πληροφορία θα διοχετευθεί στην εφαρμογή με τον καλύτερο δυνατό τρόπο, δηλαδή με την μέγιστη ασφάλεια, με σταθερότητα και γνησιότητα [25] [26].

3.1.2. Μειονεκτήματα

Παρόλα τα θετικά που μας προσφέρει η χρήση ενδιάμεσου λογισμικού κατά την σχεδίαση και υλοποίηση κάποιου συστήματος υπάρχουν και αρνητικά που πρέπει να αναφερθούν.

Το ενδιάμεσο λογισμικό τρέχει πάνω από το λειτουργικό σύστημα και αυτό απειλεί την απόδοση του συστήματος σε πραγματικό χρόνο. Υπάρχει, δηλαδή, μεγάλη πιθανότητα αν έχει γίνει ένα προγραμματιστικό λάθος στο ενδιάμεσο λογισμικό το οποίο μπορεί να σταματήσει τη λειτουργία ολόκληρου του συστήματος.

Τα εργαλεία τα οποία χρειάζονται για την δημιουργία ενός ενδιάμεσου λογισμικού δεν είναι τόσο ανεπτυγμένα ακόμη, καθώς επίσης είναι πολύ λίγοι αυτοί που έχουν την απαραίτητη εμπειρία και τεχνική κατάρτιση για να δημιουργήσουν και να υποστηρίξουν την τεχνολογία αυτή. Συνεπώς, υπάρχει έλλειψη γνώσης, πόρων και ανθρώπινου δυναμικού.

Το κόστος δημιουργίας ενός ενδιάμεσου λογισμικού είναι πολύ υψηλό σε αντίθεση με το χαμηλό κόστος που παρέχει στους σχεδιαστές που θα το

επαναχρησιμοποιήσουν. Η επαναχρησιμοποίηση λογισμικού που παρέχεται με την χρήση ενδιάμεσου λογισμικού, μπορεί να αυξήσει τα κόστη όταν γίνονται μεγάλες αλλαγές στο σύστημα μας, μια και ο πηγαίος κώδικας δεν είναι προσβάσιμος. Υπάρχουν πολλά εργαλεία που δεν υποστηρίζουν επαναχρησιμοποίηση λογισμικού και αυτό μπορεί να επιφέρει πολλά προβλήματα κατά την ανάπτυξη της εφαρμογής [25] [26].

3.2. Σύνοψη των Τεχνολογιών Ενδιάμεσου Λογισμικού Επίγνωσης Πλαισίου

Οι τεχνολογίες ενδιάμεσου λογισμικού έχουν αναπτυχθεί τα τελευταία χρόνια με σκοπό την χρησιμοποίηση τους από μελλοντικούς σχεδιαστές εφαρμογών, κινητών και μη συσκευών. Οι τεχνολογίες αυτές σχεδιάστηκαν αποσκοπώντας στο να μπορούν οι εφαρμογές που τις χρησιμοποιούν να εκτελούνται σε διάφορα πλαίσια και λειτουργικά περιβάλλοντα ανάλογα με τις ανάγκες των χρηστών. Μια ειδική κατηγορία είναι τα ενδιάμεσα λογισμικά επίγνωσης πλαισίου τα οποία έχουν μελετηθεί στην διπλωματική αυτή και παραθέτονται πιο κάτω. Στην διπλωματική έχει γίνει η χρήση ενός από αυτά τα λογισμικά για την δημιουργία μιας εφαρμογής επίγνωσης πλαισίου, καθώς και της παρουσίασης στο τελευταίο κεφάλαιο των συμπερασμάτων από την χρήση του λογισμικού.

3.2.1. IST MUSIC

Το ενδιάμεσο λογισμικό που δημιουργήθηκε στο πρόγραμμα IST MUSIC EU FP6 έλαβε το όνομα του Self Adapting Applications for **M**obile **U**Sers In Ubiquitous **C**omputing **E**nvironments. Είναι μια πλατφόρμα ανοικτού κώδικα για σχεδίαση και εφαρμογή προσαρμοσμένη ανάλογα για την χρήση κινητών εφαρμογών, οι οποίες είναι φτιαγμένες αποκλειστικά για το χρήστη και το πλαίσιο των πληροφοριών στο οποίο εκτελούνται. Το ενδιάμεσο λογισμικό που προσφέρεται από την πλατφόρμα διευκολύνει την ανάπτυξη context-aware εφαρμογών, οι οποίες προσαρμόζονται δυναμικά στις οποιεσδήποτε αλλαγές κατάστασης του χρήστη.

Η δημιουργία εφαρμογών με τη συγκεκριμένη τεχνολογία έχει διαδοθεί αρκετά με την ανάπτυξη των κινητών συσκευών. Το ενδιάμεσο λογισμικό τρέχει πάνω από το πλαίσιο εργασίας του OSGI, το οποίο είναι ένα πλαίσιο ανεπτυγμένο για την γλώσσα προγραμματισμού JAVA, που προνοεί την ανάπτυξη συστημάτων χωρίζοντας τα σε μικρά κομμάτια, τα λεγόμενα context plug-ins (i.e., sensors, reasoners).

Η αρχιτεκτονική του IST MUSIC περιέχει δυο μέρη. Το πρώτο είναι το MUSIC STUDIO το οποίο παρέχει την μεθοδολογία και τα εργαλεία για να υποστηρίξει την ανάπτυξη context-aware εφαρμογών. Με αυτό ο σχεδιαστής μπορεί να σχεδιάσει ένα γενικό μοντέλο για την εφαρμογή και να καθορίσει πώς η εφαρμογή θα χρησιμοποιήσει το ενδιάμεσο λογισμικό. Το δεύτερο είναι το ίδιο το ενδιάμεσο λογισμικό (MUSIC MIDDLEWARE) το οποίο παρέχει ένα περιβάλλον εκτέλεσης. Το MUSIC MIDDLEWARE χωρίζεται σε 4 επίπεδα:

- **Πυρήνας.** Χαρακτηρίζει το μοντέλο πληροφορίας του MUSIC και είναι υπεύθυνο να εγγράφει, να ξεκινάει και να λαμβάνει όλα τα drivers από το MUSIC.
- **Υπηρεσία του συστήματος.** Παρουσιάζει μια πληθώρα υπηρεσιών με στόχο να χτίζει ένα περιβάλλον και να διαχειρίζεται τους πόρους της συσκευής.
- **Πλαίσιο και εφαρμογή ενδιάμεσου λογισμικού.** Υποδεικνύει πώς οι πληροφορίες πλαισίου μπορούν να διαχειριστούν για να δημιουργηθεί μια context-aware εφαρμογή και πώς είναι καλύτερη η διαμόρφωση των πληροφοριών για κάθε εφαρμογή.
- **Περιβάλλον εφαρμογής.** Είναι το μοντέλο της διεπαφής του χρήστη το οποίο χρησιμοποιούμε για το ενδιάμεσο λογισμικό.

Το MUSIC παρέχει μια μεθοδολογία υποστήριξης ανάλογη με αυτή της σχεδίασης προγραμμάτων. Για να κατανοηθεί τι χρειάζεται η εφαρμογή και πώς θα αντιδρά στο περιβάλλον και στις υπηρεσίες της, πρέπει αρχικά να γίνει μια ανάλυση για να εντοπιστούν τα παραπάνω. Αμέσως μετά την ανάλυση ακολουθεί η μοντελοποίηση των συστατικών που θα αποτελούν την εφαρμογή. Για να μπορεί η εφαρμογή να χρησιμοποιηθεί σε πολλές πλατφόρμες, γίνεται μετασχηματισμός του μοντέλου για παραγωγή κώδικα ο οποίος προσαρμόζεται στις ανάγκες της εφαρμογής. Μετά την παραγωγή του κώδικα και το τέλος της επιπλέον ανάπτυξης της εφαρμογής, ξεκινά μια περίοδος δοκιμών και επικύρωσης για την ίδια την εφαρμογή [27].

3.2.2. WASP-Web Architectures for Services Platforms

Το WASP είναι μια πλατφόρμα για την δημιουργία context-aware ενδιάμεσου λογισμικού, που βασίζεται σε υπηρεσίες διαδικτύου. Λειτουργεί πάνω από δίκτυα 3G χρησιμοποιώντας το Parlay X σαν την διεπαφή για τις λειτουργίες του δικτύου 3G. Οι χρήστες μπορούν να χρησιμοποιήσουν αυτές τις υπηρεσίες και τις εφαρμογές που προσφέρει αυτή η πλατφόρμα μέσω της κινητής συσκευής. Στην αρχιτεκτονική του WASP υπάρχει ένα κεντρικό υποσύστημα το οποίο είναι υπεύθυνο για την συλλογή και παροχή της πληροφορίας. Η τεχνολογία αυτή χρησιμοποιείται κυρίως για την εύρεση τοποθεσιών που ενδιαφέρουν τον χρήστη, όπως μουσεία, αίθουσες σε κάποιο πανεπιστήμιο [28].

3.2.3. RSCM-Really Simple Context Middleware

Το RSCM είναι ένα ενδιάμεσο λογισμικό το οποίο επιτρέπει το σχεδιασμό context-aware εφαρμογών για το λειτουργικό σύστημα Android, μέσα από υπάρχοντα ή καινούρια plugins. Η αρχιτεκτονική του RSCM είναι βασισμένη στο IST MUSIC. Σε αντίθεση όμως με το IST MUSIC το οποίο χρησιμοποιεί τις OSGi υπηρεσίες, το RSCM δημιουργήθηκε για να χρησιμοποιεί τις υπηρεσίες Android (Android Services). Υπάρχει διαχωρισμός μεταξύ των plugins και των εφαρμογών.

Τα plugins υπάρχουν ως APK's και μπορούν να χρησιμοποιηθούν αυτόματα μετά την εγκατάσταση τους στο σύστημα. Τα plugins διαχωρίζονται σε sensors και reasoners. Τα sensor plugins είναι υπεύθυνα για να δημιουργήσουν τις πληροφορίες πλαισίου

από τα δεδομένα που παίρνουν από τους αισθητήρες (πχ. GPS sensor). Οι πληροφορίες που παίρνονται από τους αισθητήρες χαρακτηρίζονται ως πληροφορίες χαμηλού επιπέδου. Οι πληροφορίες αυτές είναι οτιδήποτε μπορεί να δώσει ο αισθητήρας και μπορεί να τα διαβάσει το σύστημα, αλλά ο χρήστης της εφαρμογής δεν μπορεί να τα ερμηνεύσει. Το reasoner plugin παράγει από τις χαμηλού επιπέδου πληροφορίες, πληροφορίες υψηλού επιπέδου όπως ο προγραμματιστής θέλει να παρουσιαστούν στον χρήστη. Δηλαδή, είναι ο ερμηνευτής και μετατροπέας των δεδομένων που παίρνει από τους αισθητήρες σε κάτι που ο χρήστης μπορεί να καταλάβει και θα του είναι χρήσιμο. Οι πληροφορίες που λαμβάνει ο χρήστης δεν τις παίρνει από το reasoner αλλά το reasoner αποστέλλει αυτές τις πληροφορίες στην εφαρμογή εκεί όπου δηλώνεται ο τρόπος και η μορφή που ο χρήστης θα τις δει.

Η μεθοδολογία πάνω στην οποία βασίστηκε το RSCM είναι η ύπαρξη μιας κεντρικής υπηρεσίας που τρέχει στο παρασκήνιο του συστήματος Android, η οποία διαμέσων κάποιων κλειδίων (i.e., Scores) μπορεί να χρησιμοποιήσει το ανάλογο Service που επιζητεί ο προγραμματιστής. Με αυτό τον τρόπο δίνεται η ευκαιρία στα διάφορα plugins που προυπάρχουν να διασφαλίσουν ποιες πληροφορίες είναι προς επεξεργασία. Τα service αυτά προυπάρχουν στο λειτουργικό των Android με αποτέλεσμα οι πληροφορίες να λαμβάνονται σε πραγματικό χρόνο ανάλογα με το περιβάλλον του χρήστη [29].

Η χρήση των κλειδίων αυτών είναι εμφανής στην λειτουργία του RSCM αφού με τον ίδιο τρόπο τα πιο πάνω plugins χρησιμοποιούνται για την επικοινωνία μεταξύ αυτών και των reasoners. Με αυτό τον τρόπο εξυπηρετείται ο διαχωρισμός μεταξύ των πληροφοριών για τις οποίες τα reasoner plugins είναι υπεύθυνα να επεξεργαστούν και να παρουσιάσουν στον χρήστη μέσω της τελικής εφαρμογής. Η σειρά με την οποία γίνονται τα πιο πάνω είναι κάπως πιο περίπλοκη αφού το plugin πρέπει να δεσμεύσει την υπηρεσία μόνο όταν αυτή χρειάζεται και το αντίθετο. Έτσι, λαμβάνοντας υπόψιν τα προαναφερθέντα μπορεί να κατανοηθεί ότι το RSCM μπορεί να λειτουργήσει δυναμικά και σε πραγματικό χρόνο δεσμεύοντας τις αναγκαίες πληροφορίες που χρειάζεται η εφαρμογή.

Το σύστημα των Android είναι υπεύθυνο για την συσκευή στην οποία είναι εγκαταστημένο. Το ίδιο το λειτουργικό σύστημα χειρίζεται τα drivers που δουλεύουν, τις εξόδους που παράγει η συσκευή, τους πόρους που είναι έτοιμοι για επεξεργασία. Αν αυτοί οι πόροι είναι διαθέσιμοι, όπως για παράδειγμα ένα GPS sensor, τότε το RSCM μπορεί να δουλέψει σε οποιαδήποτε Android συσκευή.

Το RSCM είναι σχεδιασμένο διαχωρίζοντας την εφαρμογή, από τα plugins και τα reasoners και αυτό δίνει στο σύστημα την δυνατότητα να μην είναι βαρυφορτωμένο. Αυτό συμβαίνει καθώς δεν τρέχουν όλα τα προγράμματα μαζί και το σύστημα είναι ευέλικτο με τους πόρους του, έτσι χρησιμοποιεί μόνο ό,τι χρειάζεται τη συγκεκριμένη στιγμή, χωρίς να σπαταλά άσκοπα πόρους και να μειώνει συνεπώς την ποιότητα του συστήματος.

Τα κυριότερα μέρη για να χτίσουμε μια εφαρμογή με RSCM ενδιάμεσο λογισμικό είναι η RSCM βιβλιοθήκη και το RSCM runtime. Η RSCM βιβλιοθήκη περιέχει τον

κώδικα που χρειάζεται για τον σχεδιασμό μιας εφαρμογής RSCM και πρέπει να ενσωματωθεί σε όλα τα μέρη του ενδιαμέσου λογισμικού. Κατά την υλοποίηση μιας context-aware εφαρμογής είναι πάντα αναγκαίο να προστίθεται η RSCM βιβλιοθήκη. Το RSCM runtime αποτελεί μια κεντρική υπηρεσία η οποία τρέχει στο παρασκήνιο του συστήματος Android και είναι υπεύθυνη για την σύνδεση μεταξύ των plugins και της εφαρμογής. Αυτή η υπηρεσία είναι συνδεδεμένη με τα διάφορα συστατικά του ενδιαμέσου λογισμικού και εκτελείται όποτε ένα από αυτά τα μέρη την χρειάζεται. Χωρίς αυτή την υπηρεσία δεν είναι δυνατή η λειτουργία του ενδιαμέσου λογισμικού.

Τα κυριότερα plugins τα οποία προσφέρονται από το RSCM και έχουν ήδη δημιουργηθεί, και άρα μπορούν να τύχουν επαναχρησιμοποίησης είναι:

- **Battery sensor plugin:** εμφανίζει την υπολοιπούμενη μπαταρία της συσκευής.
- **Power sensor plugin:** εμφανίζει αν η συσκευή είναι ενωμένη ή όχι σε υποδοχή ηλεκτρικού ρεύματος.
- **Location (coarse/fine) plugin:** εμφανίζει την θέση της συσκευής.
- **User activity plugin:** είναι το reasoner plugin το οποίο είναι υπεύθυνο για την αλλαγή της μορφής των πληροφοριών που έχουν να κάνουν με το user activity.

Η κύρια δραστηριότητα και χρήση του RSCM δίνει τη δυνατότητα σε πραγματικό χρόνο να λαμβάνει συνεχώς την αλλαγή της τιμής όπως μας την παρέχει το service και στον ίδιο χρόνο η πληροφορία αυτή μετατρέπεται, εφόσον είναι αναγκαίο μέσω του reasoner και αποστέλνεται στην εφαρμογή. Με αυτό τον τρόπο δίνεται η ευκαιρία στο χρήστη να λάβει τη σωστή πληροφορία στο σωστό χρόνο [30] [31].

4. ΥΛΟΠΟΙΗΣΗ ΕΦΑΡΜΟΓΗΣ

4.1. Γενική Περιγραφή της RSCM-based Εφαρμογής CampusQuest

Η ανάπτυξη και η υλοποίηση αυτής της διπλωματικής βασίζεται στη δημιουργία μιας Android εφαρμογής επίγνωσης πλαισίου, η οποία χρησιμοποιεί ως ενδιάμεσο λογισμικό το RSCM, το οποίο παρουσιάστηκε και εξηγήθηκε ο τρόπος λειτουργίας του στο τέλος του προηγούμενου κεφαλαίου. Η λειτουργία της εφαρμογής αφορά την λήψη ως δεδομένο εισόδου της θέσης του χρήστη μέσα στο χώρο του Πανεπιστημίου Κύπρου, μέσω των συντεταγμένων που λαμβάνει από τον αισθητήρα GPS της κινητής συσκευής. Ακολουθώντας, μέσα από την βάση δεδομένων που διατηρεί η εφαρμογή, εντοπίζει και παρουσιάζει το κτίριο στο οποίο βρίσκεται ο χρήστης με την κινητή συσκευή του καθώς και άλλες χρήσιμες πληροφορίες για το κτίριο αυτό, όπως εκδηλώσεις, γραφεία και εργαστήρια.

Η υλοποίηση της εφαρμογής απαιτεί από το ενδιάμεσο λογισμικό (RSCM), το λειτουργικό σύστημα της κινητής συσκευής (στην δεδομένη περίπτωση Android) και την εφαρμογή να αλληλεπιδρούν μεταξύ τους ώστε να παρέχουν εύκολα και αποτελεσματικά χρήσιμες πληροφορίες. Πιο συγκεκριμένα, η εφαρμογή βασίζεται στην λήψη πληροφοριών από τον αισθητήρα τοποθεσίας (GPS), παίρνοντας τις γεωγραφικές συντεταγμένες και την ανάλυση τους από το ενδιάμεσο λογισμικό ώστε να μετατραπούν σε υψηλού επιπέδου πληροφορίες για την παρουσίαση στην τελική εφαρμογή και ως τελικό προϊόν στον χρήστη. Τα δυο μέρη του RSCM που είναι υπεύθυνα για αυτή την μετατροπή είναι το λογισμικό το οποίο θα λαμβάνει συντεταγμένες και θα τις αποστέλνει στο λογισμικό το οποίο είναι υπεύθυνο για την μετατροπή τους σε λογικές , πληροφορίες για τον χρήστη όπως η τοποθεσία που βρίσκεται στο πανεπιστήμιο, τις εκδηλώσεις που θα υπάρχουν στο συγκεκριμένο κτίριο και λεπτομέρειες για τις διαφορετικές εκδηλώσεις.

Πιο συγκεκριμένα με την έναρξη του αισθητήρα (RSCM-based sensor plug-in) αλλά και της εφαρμογής το πρώτο λογισμικό ξεκινά να λαμβάνει συντεταγμένες σε πραγματικό χρόνο διαχωρίζοντας τις σε γεωγραφικό μήκος και πλάτος , το οποίο τα αποστέλνει στο δεύτερο μέρος το Reasoner (RSCM-based reasoned plug-in). Το Reasoner λαμβάνει το αποτέλεσμα και αποσκοπεί στην μετατροπή του σε κατανοητή πληροφορία για τον χρήστη. Μέσα από μία βάση δεδομένων οι πληροφορίες αυτές φιλτράρονται αντιστοιχώντας την θέση του χρήστη με πραγματικό χώρο του πανεπιστημίου και επιλέγοντας τις κατάλληλες πληροφορίες που πρέπει να αποσταλούν στην εφαρμογή που χρησιμοποιεί ο χρήστης. Δημιουργείται έτσι μια λογική σειρά ανταλλαγής και μετατροπής δεδομένων με στόχο το αποτέλεσμα που λαμβάνει ο χρήστης.

4.2. Υλοποίηση της Εφαρμογής Campus Quest

Η εφαρμογή μας Campus Quest δεν περιλαμβάνει μια εφαρμογή με τον όρο της λέξης, αλλά ένα σύνολο εφαρμογών, plug-ins και υπηρεσιών που λειτουργούν ταυτόχρονα με σκοπό την παρουσίαση στον χρήστη των δεδομένων που λαμβάνουν σαν κατανοητή πληροφορία.

Το σημαντικότερο συστατικό στην λειτουργία της εφαρμογής είναι η εγκατάσταση στην κινητή συσκευή του RSCM runtime, το οποίο είναι το ενδιάμεσο λογισμικό που είναι υπεύθυνο για την μεσολάβηση μεταξύ του plug-in και της εφαρμογής μας. Χωρίς το πιο πάνω η παροχή πληροφοριών μεταξύ των συστατικών της εφαρμογής δεν είναι δυνατή αφού αυτό είναι που καταλαβαίνει τότε ο αισθητήρας μας είναι σε λειτουργία και εν συνεχεία ενημερώνει το plug-in να λάβει τα δεδομένα που χρειάζεται, καθιστώντας έτσι και μείωση στην σπατάλη πόρων της συσκευής μας. Έτσι, για την εγκατάσταση του λογισμικού αυτού πάρθηκε βοήθεια από τον ιστότοπο του RSCM και ακολούθως έγινε λήψη του [rscm-runtime 0.2.1.apk](#). Με την εγκατάσταση του λογισμικού αυτού συνεχίζεται η υλοποίηση των επιμέρους κομματιών της εφαρμογής.

Ως πρώτο κομμάτι εστίασης και υλοποίησης ορίστηκε η δημιουργία του plug-in το οποίο θα δεσμεύει την υπηρεσία τοποθεσίας (GPS) και θα λαμβάνει την θέση του χρήστη σύμφωνα με το γεωγραφικό του μήκος και πλάτος. Για να γίνει δυνατή η χρήση των μεθόδων που παρέχει το RSCM τοποθετήθηκαν στις βιβλιοθήκες της εφαρμογής η βιβλιοθήκη του RSCM επονομαζόμενης και ως [rscm-library.jar](#). Πρώτο μέλημα για το συγκεκριμένο plug-in ήταν η δέσμευση και αποδέσμευση της υπηρεσίας GPS που παρέχει το λειτουργικό σύστημα των Android για να δοθεί στο plug-in η δυνατότητα να κατανοεί τότε πρέπει να αρχίσει να λαμβάνει δεδομένα από τον αισθητήρα. Με τις μεθόδους OnBind και OnUnbind πιο συγκεκριμένα δηλώνεται τότε είναι χρήσιμη η δέσμευση της υπηρεσίας.

```
@Override
    public IBinder onBind(Intent intent) {

        locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER,
                                                MIN_TIME, MIN_DISTANCE, this);
        return super.onBind(intent);
    }
@Override
    public boolean onUnbind(Intent intent) {
        locationManager.removeUpdates(this);
        return super.onUnbind(intent);
    }
```

Η υπηρεσία η οποία ήταν αναγκαίο να υλοποιηθεί είναι αυτή της παροχής δεδομένων για την τοποθεσία του χρήστη και για τον λόγο αυτό χρησιμοποιήθηκαν συγκεκριμένες μέθοδοι τις οποίες παρέχει το ίδιο το λειτουργικό σύστημα των Android επιλέγοντας το στοιχεία που χρειάζονται για την όσο καλύτερη και ακριβέστερη τοποθέτηση του χρήστη σε γεωγραφική τοποθεσία σε πραγματικό

χρόνο. Τα δύο στοιχεία που θα παρέχει η υπηρεσία αυτή είναι το γεωγραφικό μήκος και πλάτος της συσκευής με απόκλιση ενός μέτρου, πράγμα που εισάγεται στην υπηρεσία αυτή θυσιάζοντας όμως χρόνο στον εντοπισμό αφού το GPS θα χρειαστεί περισσότερους δορυφόρους για να επιβεβαιώσει την τοποθεσία, μειώνοντας όμως παράλληλα και τα λάθη που μπορούν να παρουσιαστούν, αφού τα κτήρια του πανεπιστημίου βρίσκονται κοντά το ένα στο άλλο και υπάρχει η περίπτωση παρουσίας λάθος στοιχείων όταν ο χρήστης βρίσκεται σε διαφορετικό κτίριο. Επίσης, δίνεται στο σύστημα χρόνος μέχρι και δέκα δευτερολέπτων για επιβεβαίωση και αλλαγή της τοποθεσίας αφού υπολογίζεται ο μέσος όρος που χρειάζεται ένας φοιτητής να μετακινηθεί από το ένα κτίριο στο άλλο βρισκόμενος στην μέση του άλλου κτηρίου.

```
public static final long MIN_TIME = 10000;  
public static final float MIN_DISTANCE = 1;
```

Τέλος το plug-in πρέπει να είναι υπεύθυνο ώστε να αποστέλλει τα δεδομένα που παίρνει την στιγμή που αυτό τα λαμβάνει από τον αισθητήρα στην επόμενη εφαρμογή για να μεταποιηθούν. Τα στοιχεία που παρέχει το plug-in θεωρούνται χαμηλού επιπέδου πληροφορίες αφού το γεωγραφικό μήκος και πλάτος σε μορφή πραγματικού αριθμού δεν θα έχουν καμία σημασία για τον χρήστη. Το rscm-library παρέχει την παράμετρο, ContextValue , η οποία με οποιαδήποτε μεταβολή των τιμών που παρέχει ο αισθητήρας θα τις τροφοδοτήσει προς την επόμενη εφαρμογή. Η μέθοδος αυτή προϋποθέτει όμως ένα περιορισμό ο οποίος επιβάλλει την μετάδοση πληροφοριών να γίνεται μόνο με την χρήση αρχέγονων τύπων ή με την μορφή String. Σε αυτή την περίπτωση όμως υπάρχει η παραγωγή δύο double τιμών, οι οποίες δεν μπορούν να αποσταλούν ξεχωριστά, έτσι πρέπει να βρεθεί λύση ώστε να γίνει η αποστολή των δυο αυτών τιμών.

Η λύση η οποία θεωρείται η καλύτερη για την αποστολή ήταν η χρήση του JSON String το οποίο υποστηρίζει και το ίδιο το λειτουργικό Android. Το JSON String είναι μια μορφή String το οποίο χρησιμοποιεί χαρακτηριστική σύνταξη ως προς την δημιουργία του αλλά και ανάγνωση του, επιτρέποντας την επισύναψη πολλών αρχέγονων τύπων σε αυτό. Η σύνταξη του JSON στηρίζεται στην δημιουργία ζευγών κλειδιού-τιμής μέσα σε αγκύλες , δίνοντας πρώτα το κλειδί για την τιμή το οποίο θα αναγνωριστεί κατά την ανάγνωση, και στην συνέχεια την τιμή. Σε αυτή την περίπτωση θα χρησιμοποιηθεί η δημιουργία ενός String το οποίο όταν συνταχθεί, θα τοποθετήσει τις τιμές και την ονομασία των κλειδιών τους σε αυτό το οποίο θα αποσταλεί μέσω της μεθόδους notifyListener() ως ένα String. Με τις πιο πάνω μεθόδους και διαδικασίες το plug-in είναι σε θέση να αποστείλει τα δεδομένα στην επόμενη εφαρμογή. Το RSCM για την αναγνώριση των διαφόρων μερών της εφαρμογής χρησιμοποιεί τα επονομαζόμενα Scores, τα οποία δηλώνουν από πού θα λάβει τις πληροφορίες του το κάθε μέρος. Πιο συγκεκριμένα, το plug-in αποστέλλει το Score "location.fine" το οποίο πρέπει να παραληφθεί από την επόμενη εφαρμογή έτσι ώστε να συνδεθούν οι δύο μεταξύ τους.

```

@Override
public void onLocationChanged(Location location) {
    double latitude = location.getLatitude();
    double longitude = location.getLongitude();

    double sixDlatitude = roundSixDecimals(latitude);
    double sixDlongitude = roundSixDecimals(longitude);

    String coordinatesAsJSONString = "{ lat: " +
sixDlatitude + ", lng: "
        + sixDlongitude + " }";

    sendContextValue(coordinatesAsJSONString);
}

private double roundSixDecimals(double d) {

    DecimalFormat sixDecimalForm = new
DecimalFormat("#.#####");

    double sixDdouble =
Double.valueOf(sixDecimalForm.format(d));
    return sixDdouble;
}

private void sendContextValue(String value) {

    ContextValue lastContextValue =
ContextValue.createContextValue(
        SCOPE_LOCATION_FINE, value);

    notifyListener(lastContextValue);
}

```

Ο χρήστης πρέπει να ενημερώνετε όμως και για την κατάσταση που βρίσκεται ο αισθητήρας, λόγω του ότι αν ο αισθητήρας είναι κλειστός η απόκτηση θέσης μέσω αυτού δεν θα είναι δυνατή. Θεωρήθηκε ότι το πιο σωστό είναι η παροχή String με στατικές ονομασίες, οι οποίες θα χρησιμοποιηθούν σε όλα τα μέρη της εφαρμογής. Δημιουργήθηκε ένα Listener, ο οποίος είναι υπεύθυνος για να ελέγχει την κατάσταση του αισθητήρα. Αυτός ο Listener κατανοεί πότε ο αισθητήρας προσπαθεί να βρει θέση και ενημερώνει τον χρήστη για αυτό, όταν όμως ο αισθητήρας κλειδώσει μια θέση αφαιρείται αυτό το Listener και επιτρέπεται έτσι η αποστολή των δεδομένων από την μέθοδο OnLocationChanged.

```

public Listener myGPSListener = new GpsStatus.Listener() {
    public void onGpsStatusChanged(int event) {

        switch (event) {
            case GpsStatus.GPS_EVENT_STARTED:

                sendContextValue(GPS_SEARCHING);

                break;
            case GpsStatus.GPS_EVENT_FIRST_FIX:

                Location gpslocation = locationManager
                    .getLastKnownLocation(LocationManager.GPS_PROVIDER);

                if (gpslocation != null) {

                    locationManager.removeGpsStatusListener(myGPSListener);
                }
                break;
        }
    }
};

```

Το επόμενο βήμα στην σύνδεση των στοιχείων είναι η δημιουργία του Reasoner plugin της εφαρμογής η οποία θα είναι υπεύθυνη για την μετατροπή των χαμηλού επιπέδου πληροφοριών σε υψηλού, δηλαδή σε πληροφορίες λογικές και κατανοητές προς τον χρήστη. Το πρώτο μέλημα για την συγκεκριμένη εφαρμογή είναι και πάλι η δέσμευση και αποδέσμευση της υπηρεσίας με τις μεθόδους OnBind() και OnUnbind() και στην συνέχεια η παραλαβή των δεδομένων τα οποία αποστάληκαν από τον αισθητήρα τοποθεσίας (sensor plug-in).

Η παραλαβή των δεδομένων γίνεται με την μέθοδο OnContextValueChanged(), η οποία λαμβάνει το ContextValue που δημιουργήθηκε στον αισθητήρα τοποθεσίας (sensor plug-in) και περιλαμβάνει το Scope και το String που αποστάληκε. Εξετάζεται αρχικά αν το Scope είναι το "location.fine" και εν συνεχεία λαμβάνεται το String χρησιμοποιώντας τη συνάρτηση getValueAsString. Το String αυτό, όπως προαναφέραμε, είναι στην μορφή JSON και για την ανάγνωση και μετατροπή του σε δύο ξεχωριστά String δημιουργείται ένα JSON Object και εν συνεχεία περνά σε αυτό την μεταβλητή του String. Με τον τρόπο αυτό μπορούν να ληφθούν οι μεταβλητές που αποστάληκαν χρησιμοποιώντας τα κλειδιά για την κάθε μια τα οποία ανατέθηκαν στο plug-in. Έτσι, στο Reasoner υπάρχουν οι χαμηλού επιπέδου πληροφορίες και είναι στην κρίση του προγραμματιστή πως αυτές θα πρέπει να φιλτραριστούν, να μετατραπούν και να χρησιμοποιηθούν για να παρουσιαστούν τα δεδομένα στον χρήστη. Δίνεται η δυνατότητα στο Reasoner να διαβάσει το String και αν η τιμή του ισούται με τις στατικές τιμές για την προσπάθεια εξεύρεσης θέσης ή αν ο αισθητήρας είναι κλειστός ούτως ώστε να διοχετευθούν αυτές οι πληροφορίες στην τελική εφαρμογή.

Η εφαρμογή χρειαζόταν να γνωρίζει τις ακριβείς συντεταγμένες για το κάθε κτίριο με σκοπό να ανατεθούν στο καθένα ο συγκεκριμένος δικός του πραγματικός γεωγραφικός χώρος. Η σκέψη ήταν η χρήση δεδομένων από το Google Maps, το οποίο αποτελεί ένα αξιόπιστο παγκόσμιο δορυφορικό χάρτη που παρέχει το γεωγραφικό μήκος και πλάτος. Ο αισθητήρας GPS όμως δίνει δεδομένα με περισσότερα από έξι δεκαδικά ψηφία σε αντίθεση με το Google Maps το οποίο δίνει έξι ψηφία ακριβώς. Στόχος όπως προαναφέραμε είναι και η ακρίβεια στην θέση του χρήστη αφού οποιαδήποτε απόκλιση μπορεί να παρουσιάσει ψευδή στοιχεία. Για την επίλυση του συγκεκριμένου προβλήματος αποφασίστηκε ότι πρέπει να δημιουργηθεί μια μέθοδος η οποία θα παρουσιάζει το γεωγραφικό μήκος και πλάτος σε έξι δεκαδικά ψηφία στρογγυλοποιώντας τα κατάλληλα. Με αυτή την μέθοδο εξάγεται ο μέσος όρος των τεσσάρων σημείων στη γωνία κάθε κτηρίου. Χρησιμοποιήθηκαν μόνο οι γωνιές κάθε κτηρίου αφού τα πλείστα κτήρια στην Πανεπιστημιούπολη του Πανεπιστημίου Κύπρου έχουν σχήμα ορθογώνιο και βρίσκοντας το μέγιστο και ελάχιστο για το γεωγραφικό μήκος και πλάτος τοποθετείται ο χρήστης στο κτίριο αυτό αν η γεωγραφική του θέση το προνοεί.

Με τα στοιχεία των κτηρίων δημιουργήθηκε μία βάση δεδομένων η οποία διαθέτει πέντε στήλες με τα στοιχεία του μέγιστου και ελάχιστου γεωγραφικού μήκους και πλάτους και του ονόματος του κάθε κτηρίου. Με την συγκεκριμένη βάση δίνεται η δυνατότητα στο Reasoner να κατανοήσει που βρίσκεται ο χρήστης στον χρόνο που λαμβάνονται τα δεδομένα. Η βάση αυτή θεωρήθηκε σωστό να δημιουργηθεί μέσα στον ίδιο τον Reasoner αφού τα κτήρια που υπάρχουν αυτή την στιγμή στην Πανεπιστημιούπολη είναι σταθερά και οποιαδήποτε αλλαγή τους θα χρειαστεί κάποιο χρονικό διάστημα να πραγματοποιηθεί, δίνοντας την ευκαιρία να αλλαχτεί η ίδια η εφαρμογή και να παρουσιαστεί ως ενημέρωση ώστε να μην επιβαρυνθεί ο χρήστης με την χρήση του διαδικτύου αλλά και των πόρων της συσκευής του.

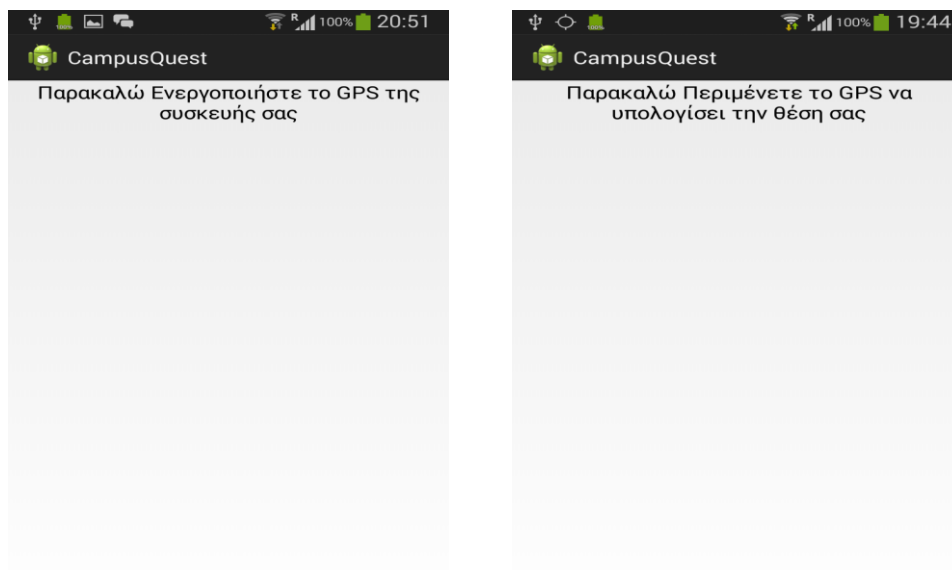
Η βάση δεδομένων όμως για τις εκδηλώσεις δεν θα ήταν σωστό να βρίσκεται στατική μέσα στο Reasoner διότι οι εκδηλώσεις εναλλάσσονται μέρα με την ημέρα, οπότε για μια δυναμική διαδικασία πρέπει να δοθεί στον χρήστη η ικανότητα να την ανανεώνει και το συγκεκριμένο μπορεί να γίνει με μια βάση δεδομένων από το διαδίκτυο και ανανέωση της από αυτό, χωρίς την ανάγκη συνεχών ενημερώσεων της εφαρμογής.

Πιο συγκεκριμένα, δημιουργήθηκε μια βάση δεδομένων SQLite για τα κτήρια όπως επιτρέπει το λογισμικό Android και τοποθετήθηκαν σε ένα πίνακα με όνομα Places και ο οποίος δημιουργείται μαζί με την εφαρμογή. Όσον αφορά τα Events δημιουργήθηκε ένας πίνακας ο οποίος περιέχει την ονομασία του κτηρίου και ένα JSON Array το οποίο περιλαμβάνει όλα τα Events σε μορφή JSON String με τις διάφορες μεταβλητές που το αποτελούν να έχουν τοποθετηθεί σε ένα αρχείο .txt. Για την αποφυγή δημιουργίας μιας βάσης δεδομένων σε MySQL και σύνδεσής της με PHP αποφασίστηκε ότι ο πιο εύκολος τρόπος για παραλαβή δεδομένων μέσω του διαδικτύου είναι η δημιουργία ενός .txt αρχείου στο οποίο μπορούν να έχουν πρόσβαση οι προγραμματιστές. Η δημιουργία του αρχείου αυτού είναι πάρα πολύ εύκολη να γίνει αφού το μόνο πρόγραμμα που χρειάζεται είναι το Notepad. Στο αρχείο αυτό δημιουργήθηκε σε JSON Syntax ένα JSON Array το οποίο περιλαμβάνει

τον αριθμό των JSON Object που αντιστοιχούν στα κτήρια που καταγράψαμε. Μέσα σε κάθε ένα από αυτά τα JSON Object τοποθετήθηκαν στην συνέχεια μια μεταβλητή με όνομα building και η οποία είναι η ονομασία του κτηρίου και ένα JSON Array με όλα τα Events για το συγκεκριμένο κτίριο. Για να μην υπάρξει η ανάγκη για εγγραφή σε Online Host Server, θεωρήθηκε εύκολο και πρακτικό να τοποθετηθούν το αρχείο στο Public φάκελο του Dropbox και να χρησιμοποιείται το Public URL για ανάγνωση του .txt αρχείου. Με τον τρόπο αυτό γίνεται εύκολα η ανανέωση των πληροφοριών για τα Events αλλά και τα ίδια τα Events μόνο με την χρήση του προγράμματος Notepad.

Με τα πιο πάνω δεδομένα βγαίνει το συμπέρασμα ότι για την σωστή λειτουργία της εφαρμογής είναι σημαντική και η χρήση του διαδικτύου έτσι ώστε ο Reasoner να μπορεί να συνδεθεί και να προσθέσει στην βάση δεδομένων τα στοιχεία που την αποτελούν. Για τον λόγο αυτό προστέθηκε στα Permission του Reasoner που βρίσκονται στο AndroidManifest.xml, το Permission για να επιτραπεί πρόσβασης στο INTERNET. Τροχοδρομείται έτσι η δημιουργία μιας μεθόδου, η οποία θα κατανοεί αν υπάρχει σύνδεση στο διαδίκτυο και θα ενημερώνει τον χρήστη για αυτό, διαφορετικά τα δεδομένα που λαμβάνει ο χρήστης θα είναι λανθασμένα και εκτός χρόνου.

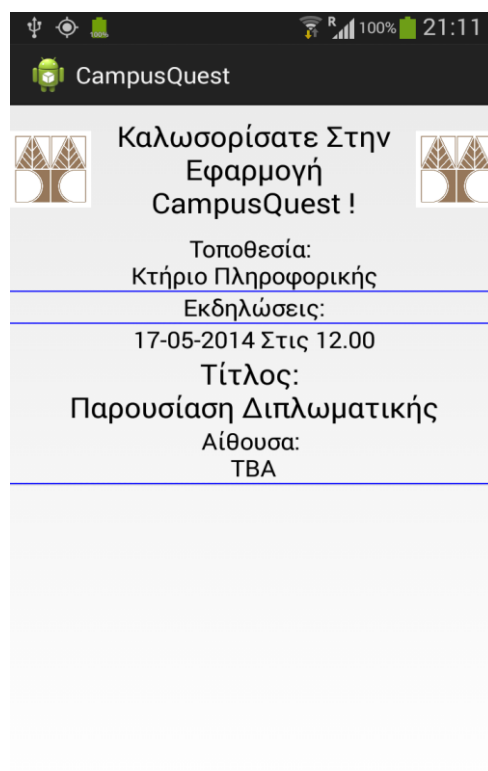
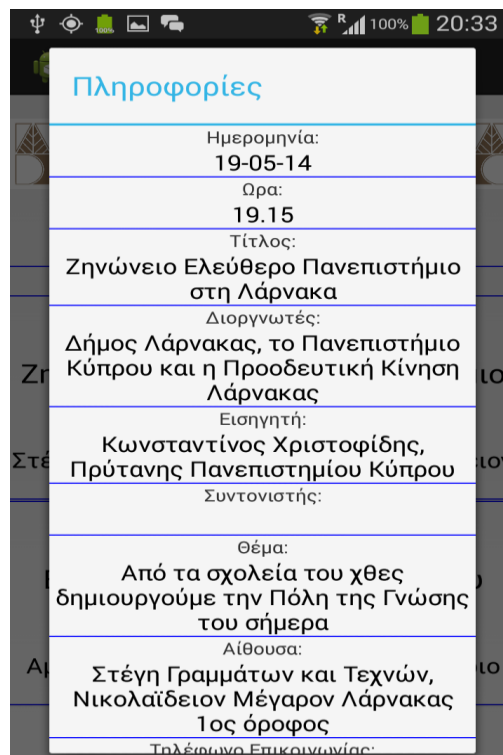
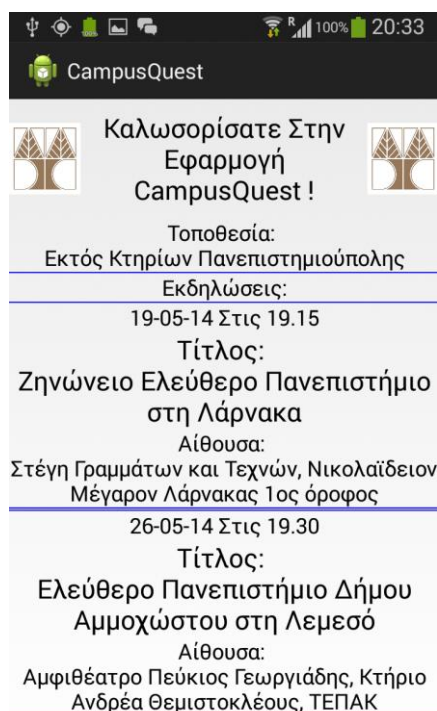
Στη συνέχεια εστιάζεται η προσοχή στην αποστολή των δεδομένων από το Reasoner στην τελική εφαρμογή για την παρουσίαση των αποτελεσμάτων στον χρήστη. Όπως αναφέρθηκε πιο πάνω το Reasoner λαμβάνει πληροφορίες για την κατάσταση της τοποθεσίας του χρήστη, κάτι το οποίο είναι πολύ σημαντικό για τον χρήστη. Για τον λόγο αυτό χρησιμοποιείται η ίδια τακτική για αποστολή των δεδομένων που καταδεικνύουν αν το GPS ψάχνει για θέση ή αν είναι κλειστό και τα δεδομένα αυτά αποστέλλονται με τον ίδιο τρόπο στην τελική εφαρμογή. Για την αποστολή των δεδομένων για την θέση του χρήστη δημιουργήθηκε μια μέθοδος στην βάση δεδομένων για αναγνώριση της τοποθεσίας του. Έτσι, δίνεται στον χρήστη μια πραγματική τοποθεσία η οποία ενεργοποιά την αναγνώριση των Events που την αποτελούν. Με την ύπαρξη του ονόματος της τοποθεσίας, όπως αυτά τοποθετήθηκαν, μπορούν να χρησιμοποιηθούν στην βάση δεδομένων των Events και να ανασύρθουν για αποστολή στην τελική εφαρμογή. Τα δεδομένα που πρέπει να παρουσιαστούν στον χρήστη είναι πάρα πολλά και πρέπει να σταλούν στην τελική εφαρμογή πάλι ως αρχέγονοι τύποι ή String. Επομένως τα δεδομένα αποστέλλονται ως JSON, αποστέλλοντας το κτίριο και όλα τα Events για αναγνώριση τους από την τελική εφαρμογή.



Εικόνα 6: Μήνυμα για ενεργοποίηση GPS της συσκευής.

Η δημιουργία της τελικής εφαρμογής CampusQuest έχει ως σκοπό την παρουσίαση των δεδομένων που δημιουργήθηκαν ως κατανοητές πληροφορίες. Έτσι, με τα όσα προαναφέρθηκαν πιο πάνω η εφαρμογή μπορεί να λάβει String αφού πρώτα αναγνωρίσει το Scope “user.activity” που αποστέλνει ο Reasoner και να κατανοήσει την σημασία τους. Οι πρώτες περιπτώσεις που θα παρουσιαστούν στον χρήστη είναι αυτές της μη λειτουργίας του GPS και της αναμονής για κλείδωμα θέσης (δείτε Εικόνα 6). Με τα πιο πάνω παρουσιάζεται στον χρήστη μήνυμα το οποίο θα τον παροτρύνει να ενεργοποιήσει τον αισθητήρα ή να περιμένει για την ανάκτηση θέσης. Όταν όμως παρουσιάζεται θέση, παρουσιάζονται στον χρήστη και τα δεδομένα που αφορούν την θέση αυτή. Για τον λόγο αυτό στην βάση δεδομένων παρουσιάζεται και η ένδειξη ότι ο χρήστης βρίσκεται εκτός κτηρίων. Για την περίπτωση που η θέση καταδεικνύει κτίριο της Πανεπιστημιούπολης παρουσιάζονται στον χρήστη όλα τα Events τα οποία υπάρχουν για το συγκεκριμένο κτίριο στην βάση.

Η παρουσίαση γίνεται μέσω μίας Listview που παρέχεται από τα Android και ενός Adapter, ο οποίος δημιουργήθηκε και προστέθηκε στο Listview για να παρουσιαστεί απο αυτήν. Στον adapter δόθηκε το JSON Array που περιέχει όλα τα Events και ανακτούνται όλες οι λεπτομέρειες ξεχωριστά για να παρουσιαστούν από την λίστα. Για τον λόγο όμως ότι οι πληροφορίες που πάρθηκαν από την ιστοσελίδα του Πανεπιστήμιο Κύπρου για τα Events είναι αρκετές, παρουσιάζονται αρχικά μόνο οι πιο σημαντικές εκδηλώσεις, οι οποίες περιέχουν την ημερομηνία και την ώρα του Event, ο τίτλος του και η αίθουσα στην οποία θα λάβει χώρα. Για τις άλλες λεπτομέρειες δημιουργήθηκε ένα Dialog που παρέχεται από τα Android και πάλι και τοποθετήθηκε όλο το φάσμα των πληροφοριών σε αυτό, ο οποίος θα παρουσιάζεται όταν ο χρήστης πατήσει στο συγκεκριμένο Event. Η Εικόνα 7 παρουσιάζει παραδείγματα αποτελεσμάτων της εφαρμογής:



Εικόνα 7: Παραδείγματα Εκτέλεσης και Παρουσίασης Εκδηλώσεων.

Η εφαρμογή που υλοποιείται σε αυτή τη διπλωματική έχει χρησιμοποιηθεί και δοκιμαστεί πιο εκτεταμένα σε διαφορετικά σενάρια εκτέλεσης για να αναγνωριστούν τυχών λάθη ή προβλήματα. Οι δοκιμές έγιναν σε όλα τα κτήρια που απαρτίζουν την Πανεπιστημιούπολη. Μέσα από συστηματικό έλεγχο και πολλές δοκιμές τα λάθη έχουν διορθωθεί και η εφαρμογή λειτουργεί στην εντέλεια. Η παρούσα διπλωματική ολοκληρώνεται με το επόμενο κεφάλαιο όπου παρουσιάζονται τα συμπεράσματα που εξάχθηκαν από την υλοποίηση της εφαρμογής.

5. ΣΥΜΠΕΡΑΣΜΑΤΑ

Μέσα από τις μεγάλες εξελίξεις που έχει παρουσιάσει η τεχνολογία των κινητών συσκευών δίνεται η ευκαιρία στους προγραμματιστές να αναπτύσσουν και να βελτιστοποιούν συνεχώς καλύτερες μεθοδολογίες και εργαλεία για την ανάπτυξη λογισμικού. Η τεράστια αυτή εξέλιξη δίνει την δυνατότητα να δημιουργούνται εφαρμογές που απευθύνονται σε όλες τις κοινωνικές ομάδες του κόσμου και ευκολύνουν την ζωή των ανθρώπων στο μέγιστο δυνατό.

Οι προγραμματιστές με τα μέσα και την ελευθερία που τους δίνεται αναπτύσσουν καινούριες μεθόδους. Οι μέθοδοι αυτοί υλοποιούν καινοτόμες λειτουργίες που μέχρι σήμερα δεν υπήρχαν και μπορούν να χρησιμοποιηθούν αυτούσιες και να συμπεριληφθούν σε πολλές διαφορετικές εφαρμογές. Έτσι, αναπτύχθηκε η έννοια της επαναχρησιμοποίησης και της επεκτασιμότητας που επιτρέπει την χρήση υφιστάμενων μεθόδων και γενικότερα υφιστάμενου λογισμικού σε μια πληθώρα από συστήματα. Αυτό εξοικονομεί χρόνο και χρήμα όταν αναφερόμαστε σε χρονοβόρα και πολύπλοκα συστήματα.

Η επαναχρησιμοποίηση λογισμικού έχει δώσει πολλές λύσεις και έχει φανεί χρήσιμη σε πολλές διαδικασίες ανάπτυξης εφαρμογών που ήταν χρονοβόρες και είχαν μεγαλύτερη δυσκολία. Μια από αυτές αποτελεί η σχεδίαση και υλοποίηση εφαρμογών επίγνωσης πλαισίου, οι οποίες έχουν την ικανότητα να αντιλαμβάνονται και να ανακτούν πληροφορίες οι οποίες σχετίζονται με το περιβάλλον του χρήστη. Τέτοιες εφαρμογές έχουν επίγνωση της κατάστασης του χρήστη και δρουν αποδοτικά σύμφωνα με τις πληροφορίες που λαμβάνουν, οι οποίες τυγχάνουν επεξεργασίας και κατανόησης. Είναι συχνά εφαρμογές που χρειάζονται πληροφορίες πραγματικού χρόνου και μέσω της επεξεργασίας των πληροφοριών ο χρήστης της εφαρμογής λαμβάνει ένα αποτέλεσμα αξιόπιστο.

Η μελέτη που έχει γίνει στην παρούσα διπλωματική, βασίζεται στην υλοποίηση μιας εφαρμογής επίγνωσης πλαισίου στη πλατφόρμα Android με τη βοήθεια του RSCM ενδιάμεσου λογισμικού το οποίο παρέχει επαναχρησιμοποίηση και επεκτασιμότητα λογισμικού. Η εφαρμογή επίγνωσης πλαισίου υπολογίζει τη θέση του χρήστη μέσα στην Πανεπιστημιούπολη και του παρουσιάζει τις εκδηλώσεις που γίνονται τις ερχόμενες μέρες στο συγκεκριμένο κτίριο, καθώς και άλλες πληροφορίες για το κτίριο.

Το πρώτο βήμα της μελέτης απαιτούσε να υπολογιστούν και να εντοπιστούν τα όρια κάθε κτιρίου μέσα από τα οποία το GPS θα αναγνωρίζει και θα καθορίζει τη θέση του χρήστη μέσα στο χώρο. Λόγω του ότι τα κτήρια της Πανεπιστημιούπολης έχουν ορθογώνιο σχήμα υπήρχε η δυνατότητα να οριστεί ένα μέγιστο και ένα ελάχιστο όριο γεωγραφικού μήκους και ένα μέγιστο και ένα ελάχιστο όριο γεωγραφικού πλάτους. Τα όρια για κάθε κτίριο και το όνομα του κάθε κτηρίου αποθηκεύονται σε μια βάση δεδομένων που υποστηρίζεται από τα Android, την SQLite, και παρόλο που αυτές οι τιμές είναι δύσκολο να αλλάξουν δεν αποθηκεύονται σε σταθερές μεταβλητές ή πίνακες αλλά στην βάση λόγω του ότι μας δίνεται η δυνατότητα μέσα από την κλήση ενός ερωτήματος (query) να παρουσιάζονται τα αποτελέσματα πιο γρήγορα. Σε μια επέκταση της εφαρμογής, η οποία θα περιέχει μεγάλο όγκο κτηρίων, όπως για

παράδειγμα τα κτήρια που ανήκουν στην Κυπριακή Δημοκρατία, η παρουσίαση των αποτελεσμάτων σε πραγματικό χρόνο θα ήταν δυνατή με την χρήση βάσης και όχι μεταβλητών.

Το επόμενο στάδιο απαιτεί τη δημιουργία μιας δυναμικής μεθόδου, η οποία να επιτρέπει την αναβάθμιση των εκδηλώσεων σε βάθος χρόνου. Οι εκδηλώσεις οι οποίες παρουσιάζονται στον χρήστη αλλάζουν συνεχώς καθώς η παρουσίαση για κάποιες εκδηλώσεις έχει λήξει από τη στιγμή που έχει περάσει η καθορισμένη ημερομηνία και ώρα της. Αυτό όμως πρέπει δεν μπορεί να γίνει αυτοματοποιημένα καθώς ο σχεδιαστής της εφαρμογής είναι υπεύθυνος να ενημερώνεται συνεχώς για καινούριες εκδηλώσεις και όταν μια καινούρια εκδήλωση ανακοινωθεί, πρέπει να εισάγεται στην εφαρμογή και να ενημερώνεται άμεσα ο χρήστης για την ύπαρξη της. Δημιουργείται ένας καινούριος πίνακας στη βάση δεδομένων, ο οποίος αποθηκεύει τις πληροφορίες για τις εκδηλώσεις και στην αρχή είναι άδειος. Από τη στιγμή που ο χρήστης ενώνεται στο διαδίκτυο γίνεται σύνδεση με τον διακομιστή για να παραληφθούν τα στοιχεία από το JSON αρχείο. Το JSON αρχείο βρίσκεται στο δημόσιο φάκελο του Dropbox, ο οποίος χρησιμοποιείται για να μπορεί να υπάρχει πρόσβαση στο αρχείο. Έτσι, τοποθετείται κάθε εκδήλωση σύμφωνα με ένα 'κλειδί', το οποίο είναι το όνομα του κτηρίου και το συγκρίνει με την στήλη που συνδέει τα κτήρια με τις εκδηλώσεις. Κάθε εκδήλωση τοποθετείται στο κτίριο στο οποίο αντιστοιχεί το όνομα του κτιρίου της εκδήλωσης.

Μέσα από τον κώδικα που δόθηκε για επαναχρησιμοποίηση γίνεται εφικτή η χρήση του plugin και του reasoner για αποστολή οποιασδήποτε μορφής πληροφοριών που έχουν να κάνουν με τις εκδηλώσεις. Το plugin από μόνο του εντόπιζε τη θέση της κινητής συσκευής. Επεκτείνοντας τις δυνατότητες του plugin δίνεται η δυνατότητα για επαναχρησιμοποίηση του, ούτως ώστε να κατανοείται πότε ο αισθητήρας του GPS είναι κλειστός, προσπαθεί να βρει θέση ή έχει εντοπίσει θέση και την παρουσιάζει. Συνεπώς, οι χαμηλού επιπέδου πληροφορίες που λαμβάνει η εφαρμογή μετατρέπονται σε υψηλού επιπέδου πληροφορίες, για να γίνουν κατανοητές από τον χρήστη. Τα δεδομένα αυτά δίδονται με σκοπό να τα χρησιμοποιήσει οποιαδήποτε εφαρμογή ή οποιοσδήποτε άλλος Reasoner. Το Reasoner σχεδιάστηκε έτσι ώστε να μετατρέπει την πληροφορία της θέσης σε πληροφορίες κτηρίου και εκδηλώσεων. Ουσιαστικά, το RSCM λειτουργεί με μια σειρά από διαδικασίες που οποιοδήποτε μέρος του μπορεί να τροποποιηθεί για να δημιουργηθεί ένα καινούριο αποτέλεσμα σύμφωνα με τις προδιαγραφές και τις απαιτήσεις του σχεδιαστή.

Η χρήση του RSCM επιτρέπει στην εφαρμογή να επομίζεται μόνο το βάρος της εμφάνισης των αποτελεσμάτων χωρίς να γίνεται μέσα στην ίδια την εφαρμογή η λήψη και η μετατροπή των πληροφοριών στο μεγαλύτερο της μέρος. Η εφαρμογή γίνεται πιο αξιόπιστη με το RSCM αφού θα παρουσιάσει αποτελέσματα μόνο όταν αυτά υπάρχουν. Άρα, χρειάζεται να τρέξει το Reasoner και να υπολογίσει ένα σωστό και αξιόπιστο αποτέλεσμα, να το στείλει στην εφαρμογή και αυτή με τη σειρά της να το παρουσιάσει στο χρήστη.

Η μελέτη που έγινε για να δημιουργηθεί και να υλοποιηθεί η εφαρμογή που απαιτεί η παρούσα διπλωματική στηρίζεται στις προδιαγραφές οι οποίες δόθηκαν από τους

υπεύθυνους καθηγητές. Για περαιτέρω επέκταση της εφαρμογής μπορεί να δημιουργηθεί ένα plugin για τον έλεγχο ύπαρξης σύνδεσης στο διαδίκτυο για να φανεί αν η εφαρμογή μπορεί να ανανεώσει τον πίνακα των εκδηλώσεων. Σε αυτή την διπλωματική ο συγκεκριμένος έλεγχος γίνεται με κώδικα που παρέχεται από τα Android. Επίσης, μπορεί να υλοποιηθεί ένα plugin το οποίο να ελέγχει το επίπεδο της μπαταρίας και να ενημερώνει το χρήστη για τυχόν χαμηλό επίπεδο της μπαταρίας και να του συστήνει να κλείσει τον αισθητήρα GPS. Αυτές είναι κάποιες από τις μελλοντικές επεκτάσεις της εφαρμογής που μπορούν να την κάνουν πιο χρηστική και αξιόπιστη.

Εν κατακλείδι, η υλοποίηση και η δημιουργία μιας εφαρμογής επίγνωσης πλαισίου που να χρησιμοποιεί ως ενδιάμεσο λογισμικό το RSCM προσφέρει την δυνατότητα της επαναχρησιμοποίησης λογισμικού και της επεκτασιμότητας κώδικα, η οποία βελτιστοποιεί τον χρόνο και συνεπώς το κόστος υλοποίησης. Καθώς σχεδιάζεται ένα τέτοιο σύστημα γίνεται επίσπευση της διαδικασίας ανάπτυξης της εφαρμογής, αφού τα επαναχρησιμοποιήσιμα συστατικά πολλές φορές είναι δύσκολο να υλοποιηθούν και η υλοποίηση τους είναι χρονοβόρα. Συνεπώς, επιταγχύνεται η υλοποίηση της εφαρμογής αφού η επαναχρησιμοποίηση λογισμικού προσφέρει στους σχεδιαστές φερεγγυότητα για το υπάρχον σύστημα και τους απαλλάσσει από την διαδικασία εύρεσης λαθών. Υπάρχει φυσικά η ανάγκη μελέτης του ενδιάμεσου λογισμικού και η απόκτηση της γνώσης για το πώς να γίνει σωστή χρήση των διαθέσιμων plug-ins. Αυτό είναι αναγκαίο να γίνει μία φορά και μετέπειτα μπορεί να εφαρμοστεί και στις επόμενες εφαρμογές επίγνωσης πλαισίου που θα κληθεί να υλοποιήσει ένας προγραμματιστής.

Ένα αξιόπιστο λογισμικό που είναι διαθέσιμο για επαναχρησιμοποίηση μπορεί να μειώσει σε μεγάλο βαθμό το κόστος ή τουλάχιστον το περιθώριο λάθους στην εκτίμηση του κόστους. Οι δυο βασικοί παράγοντες που ενδιαφέρουν τις εταιρίες είναι ο χρόνος και το κόστος και με την ανάπτυξη εφαρμογών με την μεθοδολογία που αναπτύσσεται σε αυτή την διπλωματική εξασφαλίζεται η μείωση του χρόνου ολοκλήρωσης και ελαχιστοποιείται το κόστος. Μεγάλο μερίδιο της επιτυχίας που διανύει η τεχνολογική ανάπτυξη οφείλεται στη σχεδίαση και υλοποίηση συστημάτων που χρησιμοποιούν τη μεθοδολογία της επαναχρησιμοποίησης λογισμικού και της επεκτασιμότητας.

Αναφορές

- [1] Heidi Cohen, “67 Mobile Facts To Develop Your 2014 Budget”, actionable marketing guide, October 21, 2013.
- [2] Patrick Fahy, Siobhan Clarke. “CASS - Middleware for Mobile Context-Aware Applications”, Distributed Systems Group, Computer Science Department, Trinity College Dublin, Ireland.
- [3] Juntae DeLane, “ Top 10 Marketing Tips for 2013”, *Judae Delane*, April 29, 2013.
- [4] “What is Middleware?”, Prismtech, <http://www.prismtech.com/opensplice/technologies/what-middleware>
- [5] Reconfigurable Context-Sensitive Middleware for Pervasive Computing, Pervasive Computing, 2002, Arizona State University.
- [6] Chung-Seong Hong, Joonmyun Cho, Kang-Woo Lee, Young-Ho Suh, Hyun Kim, Hyun-Chan Lee. “Developing a Context-Aware System for Providing Intelligent robot Services”, Electronics and Telecommunication Research Institute Intelligent Robot Research Division, 161 Gajeongdong, Yuseonggu Daejeon, Republic of Korea.
- [7] Guanling Chen and David Kotz, 14/11/2005, A Survey of Context-Aware Mobile Computing Research.
- [8] Narseo Vallina-Rodriguez and Jon Crowcroft, “The case for context-aware resources management in mobile operating systems”.
- [9] Marc Novakowski, “Group-Context-Aware Mobile Applications”, Handheld Devices , Tactical Systems, 21 May 2012.
- [10] Gregory D. Abowd, Anind K. Dey, Peter J. Brown, Nigel Davies, Mark Smith, and Pete Steggles. 1999. “Towards a Better Understanding of Context and Context-Awareness”. In Proceedings of the 1st international symposium on Handheld and Ubiquitous Computing (HUC '99), Hans-Werner Gellersen (Ed.). Springer-Verlag, London, UK, UK, 304-307.
- [11] Harry Chen, Dept. of CSEE, UMBC, PhD Dissertation Proposal Defense, “An Intelligent Broker Architecture for Context-Aware Systems”, January 2003.
- [12] Darren Josef Black, Nils Jakob Clemmensen, “Exploring a Context-Aware Shopping Trolley”, When More is Less: Designing for Attention in Mobile Context-Aware Computing, Aalborg University Spring 2006.
- [13] W. Schwinger, Ch. Grün, B. Pröll, W. Retschitzegger, A. Schauerhuber, “Context-awareness in Mobile Tourism Guides – A Comprehensive Survey”, 1 Johannes Kepler Universität Linz, Altenbergerstrasse 69, A-4040 Linz.
- [14] Marziah Karch, “What Is Google Android?”, About.com Google
- [15] Android Developers, <http://developer.android.com/index.html>
- [16] Lars Vogel, “Android Development - Tutorial”, <http://www.vogella.com/tutorials/Android/article.html>
- [17] “Worldwide Mobile Sales: Samsung grows”, November 16, 2012.
- [18] “Advantages and Disadvantages Android mobile phone”, <http://mobilecon.info/advantages-and-disadvantages-android-mobile-phone.html>
- [19] “The Android Advantage – Why Android?”, Android Advantage, <http://androidadvantage.blogspot.com/2012/01/android-advantage-why-android.html>

- [20] Manishankar, "ADVANTAGES AND DISADVANTAGES OF ANDROID", MARCH 14, 2013, ALLPOST, TECHNOLOGY.
- [21] "Android operating system", March 5, 2013, <http://pcmonster.gr/2013/03/android-operating-system/>
- [22] "Android Architecture", http://www.tutorialspoint.com/android/android_architecture.htm
- [23] Ενδιάμεσο Λογισμικό http://www.it.uom.gr/teaching/ince_2e_gr/Text/C3/Middleware_3.htm
- [24] Chad Brooks, "What is Middleware?", February 25, 2014, <http://www.businessnewsdaily.com/5977-middleware.html>
- [25] Job Seekers, "Advantage and Disadvantage of Middleware", August 3, 2009
- [26] "Advantage and Disadvantage of Middleware", <http://www.docstoc.com/docs/82331539/Advantages-and-Disadvantages-of-Middleware>
- [27] "Context-aware self-adaptive platform for mobile applications", EU FP6 Project, <http://ist-music.berlios.de/site/>.
- [28] Martijn Zuidweg, José Gonçalves Pereira Filho, Marten van Sinderen, "Using P3P in a web services-based context-aware application platform", Centre for Telematics and Information Technology, University of Twente.
- [29] Nearchos Paspallis, "Really Simple Context Middleware - Offers context-awareness for ANDROID", Google Code, <https://code.google.com/p/rscm/>
- [30] Nearchos Paspallis, Middleware-based development of context-aware applications with reusable components, Ph.D. Thesis, University of Cyprus, September 2009, p. 286.
- [31] Chung-Seong Hong, Joonmyun Cho, Kang-Woo Lee, Young-Ho Suh, Hyun Kim, Hyun-Chan Lee. "Developing a Context-Aware System for Providing Intelligent robot Services", Electronics and Telecommunication Research Institute Intelligent Robot Research Division, 161 Gajeongdong, Yuseonggu Daejeon, Republic of Korea.

