



Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Μηχανικών Πληροφορικής

Πτυχιακή Εργασία

Τίτλος: Κινητή εφαρμογή για πληροφορίες σε πραγματικό χρόνο
σχετικές με τα δρομολόγια των αστικών λεωφορείων

Τριβιζάκης Ελευθέριος (ΑΜ: 2058)

Επιβλέπων Καθηγητής: Σπύρος Παναγιωτάκης

Επιτροπή Αξιολόγησης:

Ημερομηνία Παρουσίασης:

Πίνακας Περιεχομένων

Πρόλογος.....	3
1. Pervasing Computing ή Πανταχού παρόντες “υπολογιστές”	4
1.1 Θεμελίωση	4
1.2 Ανάπτυξη Pervasing Computing.....	5
1.3 Στόχευση.....	5
1.4 Σενάρια.....	6
1.5 Διαδίκτυο των Πραγμάτων.....	6
1.6 Κατηγορίες χρηστών IoT	6
1.7 Τομείς εφαρμογής.....	7
1.8 Άλλα είδη διαδικτύων	9
1.9 Η επανάσταση των ευφυών συστημάτων	10
2. Συστήματα γεωγραφικού εντοπισμού	12
2.1 GPS.....	12
2.2 Differential GPS	13
2.3 A-GPS	13
2.4 S-GPS	14
2.5 Galileo.....	14
2.6 Glonass	15
2.7 BeiDou	16
2.8 Συνδυασμός Συστημάτων Glonass/GPS.....	17
2.9 Τεχνικά Χαρακτηριστικά Συστημάτων	17
2.10 IP-geolocation.....	18
3. Location Based Services	19
3.1 Εφαρμογές Location Based Service.....	19
3.2 Σκεπτικισμός LBS	22
3.3 Απαιτήσεις LBS	23
4. Maps API	25
4.1 Google Maps APIs.....	25
4.2 Google Maps Android API	25
4.3 Yahoo! Maps.....	25
4.4 OpenLayers.....	25
4.5 Bing Maps	26
5. Ανάλυση Αγοράς	27

5.1 Μεγέθη Internet of Things	27
5.2 Συσκευές Κινητής Τηλεφωνίας	28
5.3 Λειτουργικά Συστήματα Κινητής.....	28
5.4 Αγορά Τηλεπικοινωνιών Κινητής	29
5.5 Location Based Service	30
6. Τα MMM Ηρακλείου σε πραγματικό χρόνο	32
6.1 Ανάλυση υλοποίησης.....	33
6.2 Χρησιμοποιηθέντα εργαλεία	33
6.3 Η πλευρά του διακομιστή	36
6.4 Google Maps Javascript API v3.....	38
6.5 Τεχνικές ανάκτησης δεδομένων από Βάση Δεδομένων με JQuery - Ajax.....	39
6.6 Ο ρόλος των Web Services	40
6.7 Αποστολή και λήψη δεδομένων από Βάση Δεδομένων για την εφαρμογή Android.....	42
6.8 Προγραμματισμένη εκτέλεση κώδικα σε καθορισμένα χρονικά διαστήματα	44
6.9 Android Maps API v2	45
6.10 Χρήση της βιβλιοθήκης Location	46
6.11 Google Directions API.....	48
6.12 setDirectionsPolyline(Location).....	51
6.13 resetDirectionPolyline(Location).....	53
6.14 popPointsFromPolyline(List<LatLng>, Location, int)	55
6.15 addPointsToPolyline(List<LatLng>, Location).....	55
6.16 getTheClosestMarker(Location, ArrayList<Marker>)	57
6.17 findClosestBus(LatLng, ArrayList<MarkerOptions>)	58
6.18 calculateETA(MarkerOptions, movingBus).....	60
6.19 isItComing(float).....	61
7. Συμπεράσματα και Μελλοντικές Επεκτάσεις.....	64
7.1 Συμπεράσματα	64
7.2 Μελλοντικές Επεκτάσεις	65
Βιβλιογραφία.....	66

Πρόλογος

Στην εν λόγω πτυχιακή εργασία επιχειρείται να αποτυπωθεί ο ορισμός και η ανάλυση των σύγχρονων εννοιών όπως «πανταχού παρούσα πληροφορική», διαδίκτυο των πραγμάτων καθώς και ο αντίκτυπός τους στον άνθρωπο, την κοινωνία και γενικότερα στις επιχειρήσεις.

Παρουσιάζονται τα τεχνικά χαρακτηριστικά συστημάτων θεσηγνωσίας, γίνεται ιστορική αναδρομή από την σύλληψη της ιδέας στα στάδια υλοποίησης τους, αναδεικνύεται ο ανταγωνισμός για την επικράτηση στην αγορά δορυφορικής πλοήγησης των ετερόκλητων υπερεθνικών οργανισμών διαστήματος καθώς και η διαφοροποίηση στον τρόπο λειτουργίας τους.

Επιχειρείται επίσης μια διερευνητική ματιά στο όραμα για σκεπτόμενα σπίτια και κατ' επέκταση στις σκεπτόμενες πόλεις όπου η ποιότητα ζωής, η ασφάλεια στις μετακινήσεις και η προστασία του περιβάλλοντος αποτελούν αντικειμενικό σκοπό.

Παράλληλα αναδεικνύονται οι επιπτώσεις από την χρήση έξυπνων συστημάτων στην καθημερινή ζωή, οι νέες μορφές αξιοποίησης αυτών των τεχνικών σε ένα ευρύ φάσμα δραστηριοτήτων και οι κίνδυνοι που ελλοχεύουν από τους πάροχους υπηρεσιών με επίγνωση θέσης.

Αναλύεται η αγορά σκεπτόμενων συσκευών παγκοσμίως, η μάχη για την υιοθέτηση του φορητού λειτουργικού συστήματος, η αγορά υπηρεσιών με επίγνωση θέσης και η ζήτηση των επιμέρους εφαρμογών τους. Επίσης παρουσιάζονται τα εμπόδια στην ανάπτυξη του κλάδου αλλά και τον σκεπτικισμό για την χρήση προσωπικών δεδομένων και της παραβίασης ιδιωτικότητας από μη εξουσιοδοτημένους οργανισμούς, εταιρίες ή κακόβουλα άτομα.

Τρόποι αξιοποίησης των εργαλείων ανάπτυξης εφαρμογών και υπηρεσιών ανατροφοδοτούμενες από την τρέχουσα θέση μέσω συμβάντων θέσης, ενσωμάτωση στην πράξη διεπαφών προγραμματισμού εφαρμογών χάρτη σε ιστοτόπους και σε εφαρμογές Android.

Τέλος παρουσιάζεται, ενδεικτικά υλοποιημένο, ένα δυναμικό σύστημα παροχής θέσης των λεωφορείων σε πραγματικό χρόνο για την πόλη του Ηρακλείου, στο smartphone του χρήστη ή σε περιηγητή ιστοσελίδων. Αναλύονται επιμέρους οι τεχνικές ανάκτησης δεδομένων από τον διακομιστή του συστήματος, η αποστολή δεδομένων θέσης και ο μετασχηματισμός τους σε αντικείμενα τα οποία αντιλαμβάνεται το επιλεγμένο Maps API καθώς και η διάδραση με τον χρήστη.

1. Pervasing Computing ή Πανταχού παρόντες “υπολογιστές”

Στη σύγχρονη εποχή η σχέση μεταξύ του ανθρώπου με τους υπολογιστές είναι απόλυτα διακριτή. Κάθε χρήστης έχει άμεση σχέση με τον ηλεκτρονικό υπολογιστή του, καθώς χειρίζεται το πληκτρολόγιο και το ποντίκι μπροστά σε μια οθόνη. Ωστόσο αυτή η διαδικασία απαιτεί βαθιά αλληλεπίδραση και αποκλειστική προσήλωση.

Καθώς λοιπόν ο ελεύθερος χρόνος του ανθρώπου μετακυλιέται σε χρόνο μπροστά από έναν ηλεκτρονικό υπολογιστή γεννιέται η ανάγκη ανάπτυξης νέου τύπου αντίληψης, λειτουργίας και χρήσης τους. Για την αντιμετώπιση των παραπάνω δυσκολιών, ο τρόπος σύνδεσης και αλληλεπίδρασης με το ανθρώπινο περιβάλλον πρέπει να αναθεωρηθεί με φυσικό τρόπο ως προς αυτό.

1.1 Θεμελίωση

“The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it.”

Η παραπάνω ρήση γράφτηκε το 1991 από τον Mark Weiser, ο οποίος υπήρξε θεμελιωτής του Pervasing Computing και ερευνητής του Xerox PARC. Ο Weiser περιγράφει ακριβώς την στόχευση και την φιλοσοφία συγκεκριμένου του κλάδου της πληροφορικής και της εξελικτικής διαδικασία που υπόκεινται οι υπολογιστές ως έννοια και ως φυσική παρουσία. Εν πολλοίς σκιαγραφεί έναν κόσμο με πολλούς διάσπαρτους και αόρατους υπολογιστές, όπου η επεξεργαστική ισχύ και το βάρος διεκπεραίωσης των εργασιών χάνεται στο παρασκήνιο.

Όσο πιο σημαντική είναι μια τεχνολογία, τόσο περισσότερο δεδομένη τείνουμε να την θεωρούμε. Ενδεχομένως αδυνατούμε να συνειδητοποιήσουμε την ύπαρξή της. Τις περισσότερες των περιπτώσεων ο χρήστης έχει ανάγκη, κατά την αλληλεπίδραση του με την τεχνολογία, μονάχα την απάντηση απλών ερωτημάτων. Με άλλα λόγια τη λύση ενός προβλήματος. Σημαντικό τμήμα της αλληλεπίδρασης καταλαμβάνει η διεκπεραίωση λειτουργιών, οι οποίες αφορούν τον ίδιο τον υπολογιστή, όπως για παράδειγμα η αναβάθμιση, η εγκατάσταση προγραμμάτων, η εκμάθηση, η εκσφαλμάτωση και η βελτιστοποίηση της λειτουργίας του.

Έκδηλα, λοιπόν, δημιουργείται η ανάγκη για αλλαγή της σχέσης ηλεκτρονικού υπολογιστή και ανθρώπου. Ο σύγχρονος κόσμος παρουσιάζει μια αυξανόμενη εξάρτηση από την τεχνολογία. Συνεπώς θα μπορούσε να συνυπάρξει ο ψηφιακός με τον αναλογικό κόσμο με μη ορατό τρόπο, έχοντας σα στόχο/σχέδιο την απελευθέρωση του χρόνου και το χαμένο ανθρωπισμό. Η πλήρης ενσωμάτωση του computing στην καθημερινότητα θα επιτευχθεί όταν ο άνθρωπος δεν θα είναι σε θέση να διακρίνει την ύπαρξή του.

1.2 Ανάπτυξη Pervasing Computing

Η ανάπτυξη του pervasing computing έχει ως απαίτηση οι κοινές, καθημερινές συσκευές που μας περιβάλλουν να αποκτήσουν υπολογιστική ικανότητα, δυνατότητα επικοινωνίας και δικτύωσης. Οι ηλεκτρονικοί υπολογιστές επιβάλλεται να αποκτήσουν αντίληψη του χώρου και των ατόμων που κινούνται ή βρίσκονται μέσα σε αυτόν. Η δικτυακή επικοινωνία πρέπει να γίνει πανταχού παρούσα όπως το δίκτυο ηλεκτρικού ρεύματος σήμερα. Τα υπολογιστικά υποσυστήματα δεν θα πρέπει να είναι αντιληπτά από τον χρήστη και δεν θα πρέπει να απαιτούν την ανάμιξη του για κάθε ενεργεία τους αλλά μόνο να κάνει χρήση πληροφοριών και υπηρεσιών. Συνεπώς η αυτορρύθμιση τους, ανάλογα με τις συνθήκες του περιβάλλοντος, είναι απαραίτητη. Είναι εμφανής η ανθρωποκεντρική φιλοσοφία που διέπει το pervasing computing.

Άμεση συνέπεια της διάδοσης αυτού του νέου τύπου computing είναι οι ακόμα πιο προσβάσιμες πληροφορίες και υπηρεσίες 24/7/360, με άλλα λόγια, 24 ώρες το 24ωρο, 7 ημέρες την εβδομάδα και από τις 360 μοίρες του πλανήτη. Παράλληλα ο φυσικός τρόπος επικοινωνίας με τον ηλεκτρονικό υπολογιστή θα γίνεται προφορικά ή με κινήσεις, έχοντας στόχο την αμεσότερη και γρηγορότερη λήψη πληροφοριών από απλά ερωτήματα και ανάθεση της διεκπεραίωσης εργασιών με αποδοτικότητα χρόνου χωρίς την απαίτηση από τον χρήστη της γνώσης περίπλοκων εντολών ή εξειδικευμένου λογισμικού.

Αναλυτικά, χαρακτηριστικά του pervasing computing είναι πολλοί υπολογιστές ανά χρήστη, things that think, wearables devices, σύνδεση των συσκευών σε δίκτυο, cloud computing, A.I και μόνη παρέμβαση του χρήστη είναι η σύνδεση των συσκευών στο ρεύμα. Ειδικότερα ένα τέτοιο οικοσύστημα απαρτίζεται από το κινητό computing, δηλαδή συσκευές που κουβαλάμε μαζί μας και ενσωματώνουν μικροεπεξεργαστή και λειτουργικό σύστημα, κατανομημένα συστήματα, δηλαδή πολλαπλές μηχανές στο περιβάλλον για την επίλυση ενός προβλήματος, συσκευές εισόδου για ανθρώπινη διεπαφή χρήστη οι οποίες κατανοούν το περιβάλλον όπως ο άνθρωπος, ασύρματα δίκτυα παντού διαθέσιμα, μικρές και ειδικού σκοπού συσκευές, αισθητήρες και συσκευές χειρισμού.

Οι επιμέρους διάσπαρτες συσκευές θα είναι σε θέση να αναγνωρίζουν το περιβάλλον που βρίσκονται και τον χρήστη που καλούνται να εξυπηρετήσουν, ώστε να μπορούν να παίρνουν αποφάσεις που να ανταποκρίνονται στις ανάγκες του και να εκτελούν διεργασίες που διευκολύνουν το χρήστη χωρίς απαραίτητα να έχουν την εντολή του. Επίσης η μόνη ενέργεια που θα πρέπει να κάνει ο χρήστης είναι να συνδέσει στο ρεύμα τις συσκευές και σε αυτές πέφτει το βάρος της αυτορρύθμισης, της εκμάθησης του χώρου και των συνηθειών του χρήστη, της συνεννόησης με τις γύρω μικροσυσκευές και αισθητήρες, της σύνδεσης στο δίκτυο και της λήψης αποφάσεων με κάποιου είδους τεχνητής νοημοσύνης.

1.3 Στόχευση

Η πορεία της επιστήμης των υπολογιστών ξεκινά με τα mainframes, όπου πολλοί χρήστες έχουν πρόσβαση μόνο σε ένα μηχάνημα, επεκτείνεται στους Personal Computers, όπου ο κάθε χρήστης έχει στη διάθεσή του ένα μηχάνημα, ενώ στο pervasing computing πολλά μηχανήματα εξυπηρετούν έναν χρήστη. Οι ηλεκτρονικοί υπολογιστές από εξειδικευμένη χρήση μετασχηματίζονται σε μικρά καθημερινά αντικείμενα για λύση απλών προβλημάτων.

Αναλυτικά η άμβλυνση της πολυπλοκότητας των σημερινών μηχανημάτων, η διευκόλυνση της ζωής, η μεγαλύτερη αποδοτικότητα της εργασίας, ο περισσότερος ελεύθερος χρόνος για τον άνθρωπο και ο μετασχηματισμός της σημερινής κατάστασης των φορητών συσκευών σε φορητές πληροφορίες. Βάζει τον άνθρωπο στο επίκεντρο και η πληροφορία μέσω της διάχυσης των τεχνολογιών στο περιβάλλον, βρίσκει αβίαστα μονοπάτι προς το επίκεντρο. Συγκέντρωση υπηρεσιών, ενοποίηση εφαρμογών, αλληλεπίδραση πληροφοριών. Απαιτείται όμως ενοποίηση του hardware, software, internet, services, TV και των τηλεπικοινωνιών.

Οι υπηρεσίες είναι το κλειδί της δόμησης αυτού του είδους computing αλλά ο σκεπτικισμός για την ασφάλεια των δεδομένων και των διαύλων διοχέτευσης τους στα υποσυστήματα, την διασφάλιση της ιδιωτικότητας με αποτροπή της έκθεσης συμπεριφορών που δεν επιθυμεί ο χρήστης, ο κοινωνικός

αντίκτυπος που θα υπάρξει από την εξάρτηση από τις μηχανές και τη ροή των πληροφοριών καθώς και η προσαρμοστικότητα της τεχνολογίας στους χρήστες.

1.4 Σενάρια

Παρουσίασης

1. εισέρχομαι στον χώρο της παρουσίασης με το laptop μου
2. με την τοποθέτησή του στον χώρο αυτό συνδέεται στο δίκτυο, ελέγχει τον προτζέκτορα για την προβολή του περιεχομένου και ρυθμίζει το ηχοσύστημα της αίθουσας
3. δίνω φωνητική εντολή για να αρχίσει η παρουσίαση
4. αυτό ανοίγει το ανάλογο πρόγραμμα, χαμηλώνει τα φώτα και προβάλλει την παρουσίαση με τον προτζέκτορα
5. μετά το τέλος της παρουσίασης μοιράζει ψηφιακά πληροφορίες επικοινωνίας με τις συσκευές των παρευρισκόμενων, επαναφέρει τις ρυθμίσεις περιβάλλοντος στις αρχικές

Ψώνια

1. επισκέπτομαι το αγαπημένο μου εμπορικό κέντρο
2. ψάχνω στο κινητό μου για τα διαθέσιμα καταστήματα με παπούτσια
3. η βιτρίνα του καταστήματος που είχα ψωνίσει την προηγούμενη φορά με καλωσορίζει και με ενημερώνει ότι διαθέτει ένα ζευγάρι στο νούμερο μου σε έκπτωση
4. ενώ το παραδίπλα κατάστημα με ενημερώνει μέσω κινητού για τις δικές του προσφορές σε πελάτες που βρίσκονται σε μικρή απόσταση και για περιορισμένο χρονικό διάστημα

1.5 Διαδίκτυο των Πραγμάτων

Το διαδίκτυο των πραγμάτων (Internet of Things - IoT) είναι το δίκτυο φυσικών αντικειμένων που περιέχουν ενσωματωμένη τεχνολογία για επικοινωνία και αλληλεπίδραση με τις εσωτερικές καταστάσεις ή το εξωτερικό περιβάλλον τους. Μια υποδομή IoT ξεκινά με ενσωματωμένες συσκευές, αισθητήρες, υποδομή επικοινωνίας, οι οποίες συγκεντρώνουν δεδομένα που αποθηκεύονται και αναλύονται ώστε να επιστρέψουν στον κατάλληλο προορισμό τον οποίο κρίνει το σύστημα, ως πληροφορία ή απόφαση, για να αλλάξει ή να βελτιστοποιήσει το περιβάλλον.

Βασικές έννοιες που συνθέτουν το IoT είναι η αίσθηση, η αποδοτικότητα, η δικτύωση, η ειδίκευση, ενώ βρίσκεται πανταχού παρόν. Η έννοια της αίσθησης, αναφέρεται στους αισθητήρες που φέρουν οι επιμέρους συσκευές οι οποίες απαρτίζουν το IoT και παράγουν τα δεδομένα ώστε να σχηματίσουν τις απαιτούμενες πληροφορίες. Η αποδοτικότητα έγκειται στην προστιθέμενη ευφυΐα που ελέγχει τις διεργασίες των συσκευών ώστε να επιτυγχάνεται ταχύτητα στην απόκριση των δεδομένων αλλά και μείωση των ενεργειακών απαιτήσεων. Η δικτύωση αναφέρεται στην συνδεσιμότητα όλων των αντικειμένων σε έναν ενιαίο δίαυλο αλληλεπίδρασης, η ειδίκευση αποτελεί την ποικιλία στην ενσωματωμένη τεχνολογία των διάφορων αντικειμένων που εξυπηρετούν διαφορετικές αναγκαιότητες, ενώ με το «πανταχού παρόν» τονίζει την ανάγκη ύπαρξης των εν λόγω αντικειμένων παντού στο παρασκήνιο της καθημερινότητας.

1.6 Κατηγορίες χρηστών IoT

Οι τρεις διακριτές κατηγορίες χρηστών είναι:

1. μεμονωμένοι πολίτες
2. κοινωνία των πολιτών
3. επιχειρήσεις

Αναλυτικά οι μεμονωμένοι πολίτες, με την χρήση IoT, επιδιώκουν αυξημένη ασφάλεια,

προγραμματισμό, υπενθύμιση και εκτέλεση διεργασιών, βελτιωμένο τρόπο ζωής, εξατομικευμένες συμβουλές σύμφωνα με τις εκάστοτε συνήθειες τους και μείωση κόστους ζωής.

Ως συνέπεια των παραπάνω, η κοινωνία των πολιτών προσδοκά δημόσια ασφάλεια και ενημέρωση για την αποφυγή των συνεπειών από καταστροφές όπως για παράδειγμα: τσουνάμι, σεισμούς, τρομοκρατικές ενέργειες, προστασία του περιβάλλοντος και της ποιότητας ζωής μέσω της ορθής διαχείρισης των απορριμμάτων και της εξοικονόμησης ενέργειας. Ακόμα μια προσδοκία είναι η δημιουργία θέσεων εργασίας προσανατολισμένες στην διασφάλιση υψηλού επιπέδου καθημερινότητας.

Τέλος οι επιχειρήσεις μπορούν να αυξήσουν την παραγωγικότητα τους, να διαφοροποιηθούν από τον ανταγωνισμό στην αγορά, να επιτύχουν αποτελεσματικότητα στο κόστος από την ορθή αξιοποίηση των διαθέσιμων πόρων και την καλύτερη πληροφόρηση για τις ανάγκες και τις συνήθειες των πελατών της.

1.7 Τομείς εφαρμογής

Στην υγεία, μεγάλο αντίκτυπο έχει η εξέλιξη των τηλεπικοινωνιών και δικτύων σε συνδυασμό με την ανάπτυξη της επιστήμης των βιοιασθητήρων παρέχοντας την δυνατότητα ατομικής παρακολούθησης των ζωτικών σημείων ενός ασθενή με στόχο την σύνθεση προσωποποιημένων υπηρεσιών υγείας προσαρμοσμένες στο ιστορικό και την αντίδραση του οργανισμού σε έναν συγκεκριμένο τύπο θεραπείας. Δυνατότητα ανίχνευσης πτώσεων ηλικιωμένων ή ατόμων που χρίζουν παρακολούθησης αλλά και γενικότερα απομακρυσμένη παρακολούθηση ασθενών ή και αθλητών.

Στις μεταφορές, δεσπόζουσα θέση στους συγχρόνους αυτοκινητόδρομους έχουν τα αυτόματα δίοδια όπου ο χρήστης δεν χρειάζεται να διαθέσει άμεσα χρόνο και χρήμα αλλά αυτά καταβάλλονται ασύγχρονα με προφανή αποτέλεσμα την μείωση των ουρών, άρα και καθυστερήσεων στους ταχείας κυκλοφορίας οδούς. Στα μέσα μαζικής μεταφοράς τα ηλεκτρονικά εισιτήρια εκμεταλλεύονται την εξέλιξη στην τηλεματική για την παροχή πιο σύνθετων υπηρεσιών μεταφοράς συνδυάζοντας διαφορετικά μέσα χωρίς τη μεσολάβηση εκδοτηρίων.

Στην κατανάλωση ενέργειας, οι έξυπνοι μετρητές μετασηματίζουν τα απλά δεδομένα κατανάλωσης σε πληροφορίες ενός οικοσυστήματος συσκευών που αλληλοεπιδρούν με στόχο στην ενεργειακή απόδοση. Χαρακτηριστικό τέτοιων συσκευών αποτελεί η απαίτηση σε υπερχαμηλή κατανάλωση και η ενσωμάτωση ανατροφοδοτούμενων τεχνικών εξοικονόμησης ενέργειας. Η υβριδική ηλεκτροκίνητη οδήγηση μετατοπίζει το βάρος της κατανάλωσης ορυκτών καυσίμων με τις γνωστές περιβαλλοντικές συνέπειες, στο δίκτυο ηλεκτροδότησης όπου η παραγωγή ενέργειας μπορεί να έχει ως προέλευση πιο οικολογικές τεχνικές, όπως οι ανανεώσιμες πηγές ενέργειας.

Στον τομέα παραγωγής ενέργειας, η αύξηση της απόδοσης της παραγωγής και ταυτόχρονα η προστασία του περιβάλλοντος μπορεί να έχει νόημα με τον έλεγχο και διαχείρισης της κατανάλωσης, την δυνατότητα ορθού συνδυασμού του μείγματος παραγωγής ανάλογα με τις συνθήκες του εξωγενούς περιβάλλοντος όπως η επίδραση των καιρικών φαινομένων ή των τιμών των ορυκτών καυσίμων σε μια δεδομένη χρονική στιγμή. Η παρακολούθηση των ακριβή επιπέδων πετρελαίου ή φυσικού αερίου στις δεξαμενές αποθήκευσης τους, η πίεση του νερού στο δίκτυο επιβάλλονται για την αδιάληπτη παροχή τους στο σύστημα. Για προφανείς λόγους ασφάλειας τα επίπεδα και η θερμοκρασία του νερού, καθώς και η παρακολούθηση της παραγόμενης ραδιενέργειας σε πυρηνικούς αντιδραστήρες μας δίνουν πληροφορίες για την κτηριακή υγεία τους και συνεπώς την ασφάλεια των γύρο περιοχών.

Η ασφάλεια αποτελεί κύριο προβληματισμό στις σύγχρονες κοινωνίες, ενώ η εξεύρεση τεχνικών αντιμετώπισης των εν λόγω προκλήσεων με συνδυασμό μέσων και πληροφοριών οδηγεί σε προσεγγίσεις προσωποποιημένης αντιμετώπισης τυχόν απειλών ή λειτουργεί ως παράγοντα αποτροπής. Οι έξυπνοι αισθητήρες θα είναι σε θέση να αντιληφθούν την ταυτότητα ενός ατόμου σε συγκεκριμένο χώρο με βάση την ανάλυση των βιομετρικών στοιχείων του ή και των συνηθειών του. Η αδιάληπτη λειτουργία, ως προϋπόθεση στα συστήματα ασφαλείας, μπορεί να εξασφαλιστεί μέσω της διαχείρισης τους από μια μηχανή, η οποία είναι προγραμματισμένη να διορθώνει προβλήματα μεταξύ των αισθητήρων ή συσκευών που απαρτίζουν το σύστημα είτε να ειδοποιεί έγκαιρα προς εξεύρεση εξωγενούς τρόπου αντιμετώπισης δυσλειτουργιών.

Η πρόοδος στον τομέα των επικοινωνιών αποτελεί ζωτικής σημασίας ζήτημα για την ανάπτυξη

των εφαρμογών πανταχού παρούσας πληροφορικής και του διαδικτύου των πραγμάτων με ιδιαίτερη συνεισφορά των ασύρματων τεχνολογιών επικοινωνίας. Όμως αυτή η πρόοδος επιταχύνεται με την υλοποίηση των τεχνικών αυτών είτε για εξοικονόμηση ενέργειας που οδηγεί σε πιο αποδοτικές εγκαταστάσεις είτε μέσω του ελέγχου της ποιότητας της παρεχόμενης υπηρεσίας βελτιώνοντας τα επίπεδα αυτής κατά το δοκούν και ανάλογα τις τοπικές επικρατούσες συνθήκες.

Σε ευρείας κλίμακας εφαρμογές, όπως οι έξυπνες πόλεις, επιφέρουν δραστική βελτίωση του επιπέδου διαβίωσης, της δημόσιας ασφάλειας, στους χρόνους μετακίνησης καθώς και σε περιβαλλοντικά ζητήματα όπως η ποιότητα του εισπνεόμενου αέρα ή τα επίπεδα θορύβου. Η βελτίωση της κυκλοφοριακής συμφόρησης των οδικών αρτηριών με παρακολούθηση της κίνησης των οχημάτων σε συνάρτηση με πληροφορίες όπως ο χρόνος, έκτακτα συμβάντα, προγραμματισμένες εκδηλώσεις, καιρικές συνθήκες, καθώς και τους συνδυασμούς έξυπνων χώρων στάθμευσης και έξυπνων συγκοινωνιών διοχετεύοντας τον όγκο της μετακίνησης σε άλλα μέσα ή άλλες εναλλακτικές οδούς. Η ροή όλων αυτών των πληροφοριών επιτυγχάνεται μέσω της έξυπνης σήμανσης που αλλάζει ανάλογα τις συνθήκες. Η σύνταξη χαρτών θορύβου και επιπέδων CO₂ ανα περιοχή, για την διαχρονική παρακολούθηση αυτών και για την αξιοποίησή τους ως εργαλείο για την λήψη αποφάσεων ανάπλασης περιοχών ή ανασυγκρότησης του οδικού δικτύου. Ο έξυπνος φωτισμός ανάλογα με τις περιβαλλοντικές συνθήκες ή την χρονική περίοδο. Η «Κτηριακή Υγεία» σε δημόσια κτήρια και μνημεία για την ανάληψη δράσεων αποκατάστασης, έγκαιρης αντιμετώπισης προβλημάτων και απολογισμού ζημιών μετά από φυσικές καταστροφές όπως σεισμοί και ακραία καιρικά φαινόμενα.

Η προστασία του περιβάλλοντος και των εύθραυστων οικοσυστημάτων, μέσω συστημάτων ανίχνευσης φωτιάς όσο βρίσκεται σε πρώιμο στάδιο αξιοποιώντας αισθητήρες ανίχνευσης καπνού ή λογισμικών αναγνώρισης καπνού και φωτιάς με χρήση καμερών, παρακολούθηση των επιπέδων των ρύπων στην ατμόσφαιρα, πρόβλεψη κατολισθήσεων και την διαχείριση απορριμμάτων.

Η διαχείριση υδάτινων πόρων αποτελεί πρόκληση για τον ολοένα αυξανόμενο πληθυσμό. Συνεπώς καίριας σημασίας ζήτημα αποτελεί ο έλεγχος των διαθέσιμων επιπέδων πόσιμου νερού, η ποιότητα αυτού με συνεχείς ελέγχους από ένα πλέγμα εγκατεστημένων αισθητήρων και η ενημέρωση για διαρροές για την άμεση αποκατάστασή τους ανεξαρτήτως της θέσης τους στο δίκτυο.

Στην βιομηχανία η παρακολούθηση μεγάλων εγκαταστάσεων ως προς την πρόσβαση ευαίσθητων περιοχών ή οι κλιματικές συνθήκες σε μέρη με ευαίσθητα ηλεκτρονικά συστήματα μπορεί να είναι προβληματική με την εφαρμογή των παραδοσιακών τεχνικών. Με machine to machine (M2M) εφαρμογές λύνονται τέτοιου είδους προβλήματα μέσω της αυτοδιάγνωσης της ποιότητας του εργασιακού περιβάλλοντος προλαμβάνοντας ατυχήματα και εξασφαλίζοντας την ασφάλεια του προσωπικού και των παραγόμενων αγαθών.

Το εμπόριο στις μέρες μας ξεφεύγει από το παραδοσιακό μοντέλο, όπου το εμπόρευμα βρίσκονταν είτε στα ράφια των καταστημάτων είτε στις αποθήκες όπου φυλασσόταν τα αποθέματα. Σε μια παγκοσμιοποιημένη αγορά το προϊόν ταξιδεύει μεγάλες αποστάσεις για να βρει τον αγοραστή του. Επομένως η ανάγκη ανίχνευσης της θέσης του και συνεπώς του εκτιμώμενου ή του υπολειπομένου χρόνου παράδοσης κρίνεται επιτακτική. Έξυπνα ράφια για προϊόντα διατροφής που ενημερώνουν για την ημερομηνία λήξης, τυχόν αλλεργιογόνες ουσίες ή προσφορές ανάλογα με τις συνθήκες του αγοραστή. Για τις μεγάλες επιχειρήσεις είναι απαραίτητη, για την διαμόρφωση της πολιτικής προμηθειών, η γνώση της κατάστασης, της διαθεσιμότητας και της θέσης των εμπορευμάτων μια δεδομένη χρονική στιγμή ή ακόμα και η σήμανσή τους με γνώμονα την ανθεκτικότητά τους σε συγκεκριμένες κλιματικές ή χρονικές συνθήκες και την επικινδυνότητά τους σε σχέση με τους δίαυλους ή τον τρόπο διανομής τους.

Στη γεωργία η παραγωγή κρασιού, για παράδειγμα, θα μπορούσε να βελτιστοποιηθεί ποσοτικά και ποιοτικά μέσω του ελέγχου εδάφους, τόσο με τον έλεγχο της σύνθεσης του σε θρεπτικά συστατικά όσο και στην αποτελεσματικότητα της χρήσης των υδάτινων πόρων. Ανάλογη βελτίωση, με έλεγχο του μικροκλίματος και των γενικότερων συνθηκών, μπορεί να υπάρξει με εφαρμογές των παραπάνω τεχνικών σε περιβάλλον θερμοκηπίου.

Στην κτηνοτροφία, με παρακολούθηση από ειδικούς αισθητήρες και πληροφοριακά συστήματα επιτυγχάνεται η αναχαίτιση ασθενειών λόγω των συνθηκών ανατροφής, η βελτίωση της ποιότητας του αέρα καθώς παρακολουθούνται τα παραγόμενα αέρια από τα ζώα.

Στους αυτοματισμούς σπιτιών, με απομακρυσμένο έλεγχο των οικιακών, παροχή

πληροφοριών για την κατανάλωση νερού και ηλεκτρικής ενέργειας, το περιβάλλον και η συντήρηση αγαθών τέχνης, επίσης σημαντικό κομμάτι είναι η ασφάλεια κατοικιών και η ανίχνευση εισβολέα σε χρονικές περιόδους απουσίας.

1.8 Άλλα είδη διαδικτύων

Δεν μπορεί να προβλεφθεί η εξέλιξη του διαδικτύου των πραγμάτων από την ευρεία χρήση του αλλά ούτε οι επιπτώσεις που θα επιφέρει στην βιομηχανία, στο εμπόριο, στην κοινωνία και κυρίως στον άνθρωπο. Μοιραία όμως συνέπεια αποτελούν το Internet of Energy (IoE) και Internet of Vehicles (IoV).

Το IoE αποτελεί ουσιαστικά την εφαρμογή του IoT στην παραγωγή και διανομή ενέργειας. Συνδυάζει διαφορετικούς τρόπους παραγωγής ενέργειας και τοπικής διαφοροποίησης της κατανάλωσης, εξυπηρέτηση της ζήτησης, καθώς και εισαγωγή ενέργειας στο τοπικό σύστημα από εξωτερικό σύστημα παραγωγής. Κύρια χαρακτηριστικά του αποτελούν η παρακολούθηση της κατανάλωσης για αποτελεσματικότερη διοχέτευση ηλεκτρικής ισχύος με τις μικρότερες απώλειες δικτύου, το σύστημα δικτύωσης και επικοινωνιών με αισθητήρες του οποίου η ύπαρξη είναι κρίσιμη για την λειτουργία του IoE. Η εξοικονόμηση ενέργειας σε επίπεδο χρήστη και οι ανανεώσιμες πηγές ενέργειας αποτελούν βασικό πυλώνα της μελλοντικής διαχείρισης ενός τέτοιου συστήματος. Μέσω των έξυπνων μετρητών και των αισθητήρων κατανάλωσης παίρνουμε πληροφορίες για την τρέχουσα ζήτηση, ενώ εύλογη είναι η εύρεση πηγών σπατάλης και η άμεση εξουδετέρωσή τους. Οι τιμές είναι κυμαινόμενες και καθορίζονται από την τρέχουσα ζήτηση. Έτσι με αυτόν τον τρόπο γίνεται η χειραγώγηση της κατανάλωσης. Η ηλεκτροκίνηση (electro mobility) είναι επίσης ένα σημαντικό στοιχείο ενός έξυπνου δικτύου, καθώς εν δυνάμει αποτελεί μέσω μεταφοράς ενέργειας από ετερόκλητα δίκτυα με διαφορετικές τιμές.

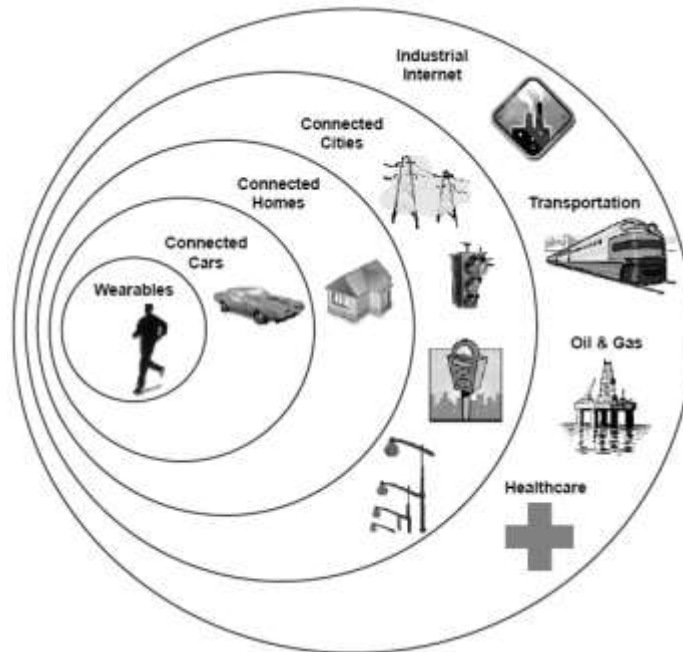
Οι προκλήσεις που πρέπει να αντιμετωπίσει άμεσα η ερευνά είναι η εξασφάλιση απόλυτα ασφαλών επικοινωνιών για την εμπέδωση της αξιοπιστίας ενός τέτοιου συστήματος, η δημιουργία συσκευών και συστημάτων για ισχυρή εξοικονόμηση ενέργειας, η κρυπτογράφηση των δεδομένων και των πληροφοριών ώστε να είναι εφικτή η ανώνυμη διακίνησή τους, καθώς και η διαχείριση των καθυστερήσεων ώστε οι πληροφορίες που προέχονται από την διαχείριση του συστήματος να είναι πάντα επίκαιρες, η αντιμετώπιση της κλιμάκωσης των συστημάτων με πρότυπα (standards) διαλειτουργικότητας, ο διαχωρισμός συστημάτων τοπικού και διαδικτυακού, η δυνατότητα μαζικής επεξεργασίας δεδομένων, το φιλτράρισμά της εξόρυξής τους για την αποφυγή της υπερχειλίσης των επικοινωνιών, τα μοντέλα πραγματικού χρόνου και σχεδιασμός μεθόδων περιγραφής διαλειτουργικότητας των ετερογενών συστημάτων, τεχνικών, περιβαλλόντων, κοινωνικών και οικονομικών, η αναγνώριση και η παρακολούθηση κρίσιμων στοιχείων του συστήματος και κρίσιμων καταστάσεων διαχρονικά, η δημιουργία πρότυπου συστήματος με δυνατότητα αυτοϊασης και περιορισμού της ζημιάς, οι κλιμακωτές διαδικασίες ασφάλειας, η γρήγορη και σωστή διακύμανση/διαχείριση της ισχύος ανάλογα με τις συνθήκες περιβάλλοντος που επηρεάζουν άμεσα τις ανανεώσιμες πηγές ενέργειας.

Το διαδίκτυο των οχημάτων (IoV), είναι η εφαρμογή του IoT στην αυτοκίνηση και τηλεματική, αποτελεί ένα κινητό οικοσύστημα βασισμένο στην εμπιστοσύνη, στην ασφάλεια, την βολικότητα των υπηρεσιών μεταφοράς. Αποτελεί μια καταναλωτικόκεντρική προσέγγιση στις μεταφορές, τα μέσα μαζικής μεταφοράς και τα αυτοκίνητα, και ουσιαστικά σαν όλον συνθέτει μια ενιαία υπηρεσία. Πρόκληση αποτελεί το γεγονός ότι δεν υπάρχει κατανόηση της ανθρώπινης συμπεριφοράς σε περιβάλλον μεταβλητού και προσαρμοστικού ελέγχου της κίνησης. Επιτακτική είναι η ανάγκη για καθορισμό προτύπων λειτουργίας του συστήματος, όπως για παράδειγμα το ποιος παίρνει τις αποφάσεις επαναφόρτισης ηλεκτρικών μπαταριών, αποφασίζει το όχημα ή ο σταθμός ανεφοδιασμού; Η ανάπτυξη εξαρτημάτων για αμφίδρομη λειτουργία με ευέλικτη χρέωση ενέργειας, σε περίπτωση που τα οχήματα θα χρησιμοποιούνται ως μέσα αποθήκευσης αυτής, το IoT αποτελεί εγγενή κομμάτι του ελέγχου του οχήματος και της διαχείρισης του συστήματος. Ήδη ορισμένα οχήματα μπορούν να ενημερώνουν online κέντρα για την προληπτική συντήρηση συστημάτων και την απομακρυσμένη διάγνωση προβλημάτων για την άμεση υποστήριξη και διάθεση ανταλλακτικών.

Στο γενικότερο πλαίσιο του συστήματος οι εφαρμογές και τα υποσυστήματα IoT ελέγχουν την

κυκλοφορία και τη διαχείριση της αυτοοργάνωσης οχημάτων προς αποφυγήν προβλημάτων όπως κυκλοφοριακή συμφόρηση, διασφάλιση της ενεργειακής απόδοσης, συνεργασία με υποδομές έξυπνης πόλης, κοινός διάυλος επικοινωνίας μεταξύ των οχημάτων για εμπέδωση της οδικής ασφάλειας και αποφυγής ατυχημάτων, υλοποίηση των πολυτρόπων μεταφορών, υπολογισμός της βέλτιστης μετακίνησης από την αφετηρία στον προορισμό με χρήση όλων των διαθέσιμων μέσων.

Εμπόδιο μπορεί να αποτελέσει η αλλαγή νοοτροπίας ως προς την αντιμετώπιση της ιδιοκτησίας. Εφόσον πλέον οι κατασκευαστές οχημάτων μετατρέπονται σε πάροχους υπηρεσιών κινητικότητας, τα οχήματα είναι κοινόχρηστα, όπως και ο σιδηρόδρομος ή η θέση στάθμευσης και όλα αυτά προσβάσιμα με εισιτήριο διαρκείας, διαθέσιμο ηλεκτρονικά. Τα επιμέρους ενσωματωμένα συστήματα του χρήστη πρέπει να αλληλοεπιδρούν με το εξωτερικό σύστημα ελέγχου της έξυπνης πόλης δίνοντας πιθανές λύσεις.



Εικόνα 1-1 Επίπεδα ΙοΤ

1.9 Η επανάσταση των ευφυών συστημάτων

Καθώς η επεξεργαστική ισχύ και δυνατότητα αυτής μετατοπίζεται από το επίκεντρο στο παρασκήνιο, αναδύονται οι πραγματικές δυνατότητες των ηλεκτρονικών οικοσυστημάτων που απαρτίζουν την σύγχρονη καθημερινότητα. Για να κινηθεί και να αξιοποιήσει τα οφέλη μιας έξυπνης κοινωνίας, ένας χρήστης πρέπει να είναι σε θέση να μιλήσει με όλα αυτά τα εξωτερικά συστήματα. Τα μέσα για να ενταχθεί και να αλληλοεπιδράσει στη νέα πραγματικότητα θα πρέπει να είναι «ορατός» στην έξυπνη κοινωνία χρησιμοποιώντας κάθε διαθέσιμο μέσο όπως wearables, smartphone, smart vehicle και προφανώς ενός είδους προγραμματισμού για το τι επιδιώκει από το γενικότερο σύστημα και προς τα που κατευθύνονται αυτές του οι ανάγκες.

Το έξυπνο σπίτι, ως κύτταρο μιας κοινωνίας με προσανατολισμό τη διασύνδεση, επικεντρώνεται στις ανάγκες του ατόμου με τρόπο συμβατό στην γενικότερη αποδοτικότητα αλλά με πρωταρχικό σκοπό τη βελτίωση του επιπέδου διαβίωσης του. Απαρτίζεται από δικτυωμένη οικοσκευή με δυνατότητα παραγωγής και επεξεργασίας απλών δεδομένων με κοινή συνισταμένη την ασύρματη υποδομή δικτύου, wearables, tablet, smartphones ως «τηλεχειριστήρια», μετρητές χρήσης, διαγνώστες, και πάροχους συμβουλών χρήσης σε φυσική προς τον άνθρωπο γλώσσα, αυτοματισμοί οικίας για ψυχαγωγία, παρακολούθησης υγείας, ενεργειακής κατανάλωσης, περιβαλλοντικής παρακολούθησης της ποιότητας ζωής, ασύρματοι αισθητήρες βελτίωσης της κτιριακής αποδοτικότητας για την επίβλεψη ενός υγιούς

περιβάλλοντος, με απόδοση κόστους και εύκολη συντήρηση.

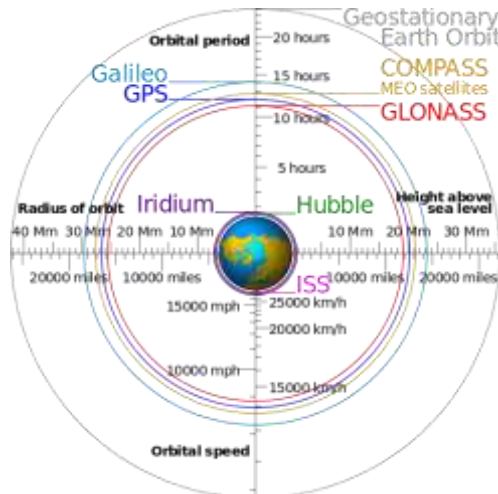
Η έξυπνη πόλη, αποτελεί τον εννοηστροπή όλων αυτών των υποσυστημάτων και συνιστά την ραχοκοκαλιά μιας IoT κοινωνίας και οικονομίας, αποτελείται από διάσπαρτες συστοιχίες από ανιχνευτές κίνησης και αισθητήρες παντός τύπου με σκοπό τον έλεγχο και την διαχείριση των διαθέσιμων θέσεων στάθμευσης, τον διαμοιρασμό των οχημάτων, τη φόρτιση των ηλεκτροκίνητων οχημάτων, σύστημα ελέγχου κλιματικών και περιβαλλοντικών συνθηκών καθώς και σύστημα ελέγχου, καταγραφής και επεξεργασίας πληροφοριών ώστε όλα τα υποσυστήματα να λειτουργούν με αποδοτικότητα. Για να είναι δυνατή η όποια άσκηση ενέργειας πρέπει να υπάρχει υποδομή αξιόπιστων καναλιών πληροφοριών, machine to machine (m2m) πρωτόκολλα επικοινωνίας για την τήρηση του χρονοδιαγράμματος με ασφάλεια και προστασία της ιδιωτικότητας της προσωπικής ζωής του χρήστη, δημιουργία και διαχείριση τεραστίου όγκου δεδομένων από εξελεγμένα συστήματα συλλογής και επεξεργασίας, ανάπτυξη διεθνώς τυποποιημένα πρωτόκολλα διαλειτουργικότητας ώστε να δημιουργηθεί ανταγωνισμός για την αποφυγή μονοπωλίων και για την ομαλή διάδραση προϊόντων διαφορετικών κατασκευαστών, ενώ για να κερδηθεί η εμπιστοσύνη των χρηστών απαραίτητη είναι η ασφάλεια, ανωνυμία, αποπροσωποποίηση των δεδομένων όπως θέση, σχεδιασμός, προορισμός.

Το διαδίκτυο των πραγμάτων βρίσκει εφαρμογή και στην παραγωγή προϊόντων αποφέροντας αποδοτικότητα στη χρήση των συντελεστών παραγωγής, πληροφόρηση για τα στάδια ενός κύκλου παραγωγής και των συντελεσμένων διεργασιών, ενημέρωση για τα αποθέματα πρώτων υλών και την διαθεσιμότητα αυτών από προμηθευτές, συντήρηση και αποκατάσταση εξοπλισμού με αυτοματοποιημένη ροή πληροφοριών. Απαιτείται εκσυγχρονισμός των μηχανημάτων στα πρότυπα m2m, χρήση μικρορομποτικής, πληροφοριακά συστήματα με τους κατάλληλους αισθητήρες, δίαλους επικοινωνίας, δυνατότητα παραμετροποίησης μηχανημάτων μέσω λογισμικού, ενσωμάτωση δυνατοτήτων δικτύωσης.

Η τωρινή βιομηχανία υγείας και παρακολούθησης ασθενών έχει επικεντρωθεί στην εφαρμογή λύσεων και εκ των πραγμάτων δεν είναι διαλειτουργική, σε σχέση με την φιλοσοφία της έξυπνης υγείας που επικεντρώνεται στο άτομα και παρέχει λύσεις σε σχέση και με συνεννόηση με το περιβάλλον που αυτό κινείται και ζει. Για την υλοποίηση της έξυπνης υγείας απαιτείται η δημιουργία εφαρμογών για την συλλογή πληροφοριών από αισθητήρες, πρέπει να υποστηρίζουν κάποιου είδους διεπαφή χρήστη, να είναι ενταγμένη ένα οικοσύστημα συσκευών με επίκεντρο τον ασθενή, διαδικτυακή συνδεσιμότητα για πρόσβαση σε υπηρεσίες υποδομής έξυπνης πόλης και όλα τα συστατικά της να χαρακτηρίζονται από χαμηλή κατανάλωση, ανθεκτικότητα στο χρόνο, ακρίβεια δεδομένων και αξιοπιστία κατά τη χρήση. Ο χώρος που κινείται το υποκείμενο της έξυπνης υγείας πρέπει να υλοποιεί ένα σύστημα περιβάλλουσας υποβοήθησης της ζωής (ambient assisted living), με υπηρεσίες υποβοήθησης καθημερινών δραστηριοτήτων, άμεση ιατρική υποστήριξη, σύστημα επείγουσας ειδοποίησης με στόχο την βελτίωση της ποιότητας ζωής ανθρώπων που χρήζουν μόνιμη παρακολούθηση και υποστήριξη για πρόληψη σοβαρών προβλημάτων.

2. Συστήματα γεωγραφικού εντοπισμού

Τα συστήματα εντοπισμού θέσης απαντούν στην ανάγκη του ανθρώπου, από την αρχαιότητα έως σήμερα, να ανακαλύπτει ταξιδεύοντας πέρα από τα στενά όρια της περιοχής που δραστηριοποιείται. Από την πλοήγηση μέσω αστερισμών στην αρχαία Ελλάδα στην πλοήγηση μέσω δορυφόρου στο σύγχρονη εποχή. Μπορεί το GPS να είναι στρατιωτικό προϊόν του ψυχρού πολέμου αλλά οι εφαρμογές στην βιομηχανία και στην καθημερινότητα είναι αναντικατάστατες.



Εικόνα 2-1: Γραφική αναπαράσταση της θέσης των διαφόρων συστημάτων θεσηγνωσίας

2.1 GPS

Στις αρχές τις δεκαετίας του 1960 το πρώτο αμερικάνικο σύστημα πλοήγησης, Transit, έδινε το στίγμα των πολεμικών πλοίων του ναυτικού και έβρισκε εφαρμογή στους βαλλιστικούς πυραύλους της εποχής. Βασισμένο σε μια «νέα έννοια» (novel concept) της δεκαετίας του 1950 και εμπνευσμένο από τον Σοβιετικό δορυφόρο Sputnik, που μετρούσε την μεταβολή Doppler από τα εκπεμπόμενα σήματα ενός δορυφόρου με γνωστή, καλώς ορισμένη τροχιά για την εκτίμηση θέσης του. Ο αστερισμός αποτελούνταν από 5 δορυφόρους σε χαμηλή τροχιά στα 1100 Km, με σήματα στενής ζώνης (narrowband) των 150 MHz και 400 MHz, με μόνον έναν δορυφόρο ορατό κάθε φορά. Η απαίτηση έως και 100 λεπτών για τον υπολογισμό της θέσης ενός χρήστη κατέστησε επιτακτική την ανάγκη για σχεδιασμό ενός καλύτερου, ακριβέστερου και ταχύτερου συστήματος θεσηγνωσίας

Το Παγκόσμιο Σύστημα θεσηγνωσίας, είναι ένας διαστημικός αστερισμός δορυφόρων με στόχο την πλοήγηση, παρέχει πληροφορίες θέσης, ταχύτητας, κατεύθυνσης και χρόνου ανεξάρτητα καιρικών συνθηκών οπουδήποτε και κοντά στη Γη αρκεί να υπάρχει ανεμπόδιστη οπτική επαφή σε τέσσερις ή περισσότερους δορυφόρους GPS.

Σχεδιάστηκε και υλοποιήθηκε από το αμερικάνικο υπουργείο άμυνας, το οποίο δεσμεύει το υψηλότερης ποιότητας σήμα για αποκλειστική χρήση και άφησαν ανοικτό σε όλους ένα επιτηδευμένο υποβαθμισμένο σήμα (Επιλεκτική Διαθεσιμότητα) με ακρίβεια 100 μέτρων. Ο πρόεδρος Bill Clinton όμως αποφάσισε να ακυρώσει την Επιλεκτική Διαθεσιμότητα πράγμα που υλοποιήθηκε τα μεσάνυχτα της 1ης Μαΐου 2000.

Παρόλα αυτά ο αμερικάνικος στρατός έχει αναπτύξει τεχνικές για άρνηση πρόσβασης στο δημόσιο σήμα του GPS ανά γεωγραφική περιοχή σε περίπτωση απειλής ή πολεμικών επιχειρήσεων για την προστασία των συμμαχικών και ιδίων στρατευμάτων.

Ο πλήρης αστερισμός απαιτεί 24 δορυφόρους, πράγμα που επιτεύχθηκε το 1991 μια προσπάθεια που είχε ξεκινήσει τη δεκαετία του 1960, επιπρόσθετα η προσθήκη άλλων 8 δορυφόρων βελτίωσε την ποιότητα και ακρίβεια του σήματος, στα 10 μέτρα. Επίσης κατ' ελάχιστο πρέπει 4 δορυφόροι να είναι σε οπτική επαφή με τον δέκτη για τον καθορισμό της θέσης και του χρόνου σε καρτεσιανό σύστημα συντεταγμένων.

Ένας δέκτης GPS εντοπίζει τη θέση του, χρησιμοποιώντας μια διαδικασία που ονομάζεται τριπλευρισμός, για τον προσδιορισμό της θέσης ενός σημείου στο χώρο αρκούν οι μετρήσεις των αποστάσεων από τρία σημεία γνωστών συντεταγμένων. Ο λόγος που απαιτούνται τουλάχιστον τέσσερις δορυφόροι είναι για να προσδιορίζεται η διαφορά ανάμεσα στην ένδειξη του χρονομέτρου του δέκτη και την ένδειξη του χρονομέτρου του δορυφόρου.

2.2 Differential GPS

Είναι η προσπάθεια βελτίωσης του δημόσιου σήματος GPS από ακρίβεια 10-15 μέτρα στα 10 cm. Χρησιμοποιεί ένα δίκτυο επίγειων σταθερών σταθμών αναφοράς που εκπέμπουν την διαφορά του δορυφορικού σήματος από την γνωστή σταθερή θέση.

Χρησιμοποιεί ένα δίκτυο σταθερών, σταθμών επίγειας αναφοράς για να μεταδώσει τη διαφορά μεταξύ των θέσεων που υποδεικνύεται από το GPS (δορυφορικά) και τις γνωστές σταθερές θέσεις. Αυτοί οι σταθμοί μεταδίδουν τη διαφορά μεταξύ των μετρούμενων δορυφορικών και των πραγματικών ψευδοαποστάσεων (υπολογίζονται εσωτερικά), για να μπορέσει ο δέκτης να διορθώσει τις ψευδοαποστάσεις του κατά το ίδιο ποσό. Το ψηφιακό σήμα διόρθωσης συνήθως μεταδίδεται σε τοπικό επίπεδο πάνω από το έδαφος με βάση πομπούς μικρότερης εμβέλειας.

2.3 A-GPS

Το Assisted-GPS, δηλαδή υποβοηθούμενο παγκόσμιο σύστημα θεσηγνωσίας, χρησιμοποιείται κατά κόρον σε συσκευές κυψελωτής τεχνολογίας διότι βελτιώνει την απόδοση εκκίνησης time to first fix (TTFF) στις πόλεις όπου η δόμηση εμποδίζει το σήμα.

Οι αυτόνομες μονάδες GPS εξαρτώνται αποκλειστικά από τα ραδιοσήματα των δορυφόρων. Το A-GPS λειτουργεί με τον ίδιο τρόπο χρησιμοποιώντας τα δεδομένα των σταθμών κινητής τηλεφωνίας για την ενίσχυση της ποιότητας και της ακρίβειας όταν προκύπτουν κακές συνθήκες δορυφορικού σήματος. Για παράδειγμα, στις αστικές περιοχές, τα δορυφορικά σήματα μπορούν να αντέξουν την παραμόρφωση πολλαπλής διαδρομής διάδοσης όπου τα σήματα αντανακλώνται από κτίρια, ή αποδυναμώνονται από τις επικρατούσες βαρομετρικές συνθήκες.

Μερικοί αυτόνομοι δέκτες πλοήγησης GPS, όταν χρησιμοποιούνται μέσα σε κακές συνθήκες, θα είναι σε θέση να καθορίσουν τη θέση από το κατακερματισμένο δορυφορικό σήμα μόνο εάν ο δέκτης κρατηθεί ψηλά για ακριβέστερο δορυφορικό σήμα. Μία μονάδα GPS μπορεί να απαιτήσει 12,5 λεπτά (ο χρόνος που απαιτείται για να κατεβάσετε το αλμανάκ GPS και τις θέσεις των δορυφόρων στον αστερισμό) για την επίλυση του προβλήματος και να είναι σε θέση να παρέχει μια σωστή τοποθεσία. Ένα A-GPS μπορεί να αντιμετωπίσει αυτά τα προβλήματα, χρησιμοποιώντας τα διαθέσιμα δεδομένα από ένα δίκτυο. Η χρήση αυτής της τεχνικής μπορεί να συνεπάγεται κάποιο κόστος για τον χρήστη. Για σκοπούς χρέωσης, οι πάροχοι δικτύου συχνά το μετράνε ως πρόσβαση σε δεδομένα, τα οποία κοστίζουν, ανάλογα με το συμβόλαιο του πελάτη. Για την ακρίβεια, τα A-GPS χαρακτηριστικά είναι ως επί το πλείστον εξαρτώμενα από το δίκτυο Internet ή / και τη σύνδεση με έναν ISP (ή CNP, στην περίπτωση του CP / συσκευή κινητού τηλεφώνου που συνδέεται με μια υπηρεσία δεδομένων Provider Cellular Network). Μια κινητή συσκευή (κινητό τηλέφωνο / SmartPhone) με ενσωματωμένο A-GPS (χωρίς επιπλέον S-GPS χαρακτηριστικό ή Standalone-GPS) για να επιλεγεί ως εναλλακτική λύση πλοήγησης, μπορεί να λειτουργήσει μόνο όταν υπάρχει μια σύνδεση στο Internet και είναι άχρηστο σε περιοχές όπου δεν υπάρχει κάλυψη δικτύου.

Υπάρχουν δύο υποκατηγορίες για την συγκεκριμένη τεχνολογία:

MSB – Mobile Station Based έχει ως στόχο το ταχύτερο κλείδωμα των δορυφόρων και την

παροχή ακριβή χρόνου μέσω της υποδομής δικτύου

MSA – Mobile Station Assisted συνδυάζει τα δεδομένα από τον δέκτη GPS με αυτά του server στον οποίο πέφτει και το βάρος υπολογισμού της θέσης. Αναλυτικά η συσκευή λαμβάνει ένα στιγμιότυπο GPS σήματος με κατά προσέγγιση χρόνο για υπολογισμό θέσης από τον server, ο οποίος λαμβάνει αξιόπιστο σήμα GPS από δική του συστοιχία δεκτών καθώς και μεγάλη υπολογιστική δύναμη για να μπορεί να συγκρίνει τα αποσπασματικά εισερχόμενα σήματα από το δίκτυο.

2.4 S-GPS

Το Simultaneous-GPS, έχει την δυνατότητα ταυτόχρονης χρήσης τηλεφωνικών κλήσεων ή δεδομένων και λήψη σήματος GPS με χρονική πολυπλεξία. Η λήψη των δύο διαφορετικών σημάτων με μια κεραία επιφέρει την σχεδιαστική πρόκληση και για αυτό μόνο οι υψηλού κόστους συσκευές την ενσωματώνουν αυτήν την περίοδο.

2.5 Galileo

Αναπτύχθηκε με σκοπό την απεξάρτηση από το GPS και την κατά βούληση διαθεσιμότητα του, από τον αμερικάνικο στρατό σε κρίσιμες στιγμές ή κατά την διάρκεια στρατιωτικών επιχειρήσεων. Επίσης έχει στόχο την μεγιστοποίηση της ακρίβειας του δημόσιου και εμπορικού σήματος χωρίς να δεσμεύει μέρος του φάσματος από στρατιωτικούς φορείς.

Παρόλα αυτά υπάρχει συμφωνία με τις Ηνωμένες Πολιτείες για συνύπαρξη των δύο συστημάτων αλλά και μελλοντικό συνδυασμό των δύο συστημάτων, όπως και κατανόηση των ανησυχιών που σχετίζονται με την προστασία των συμμάχων των ΗΠΑ και τις εθνικής ασφάλειας.

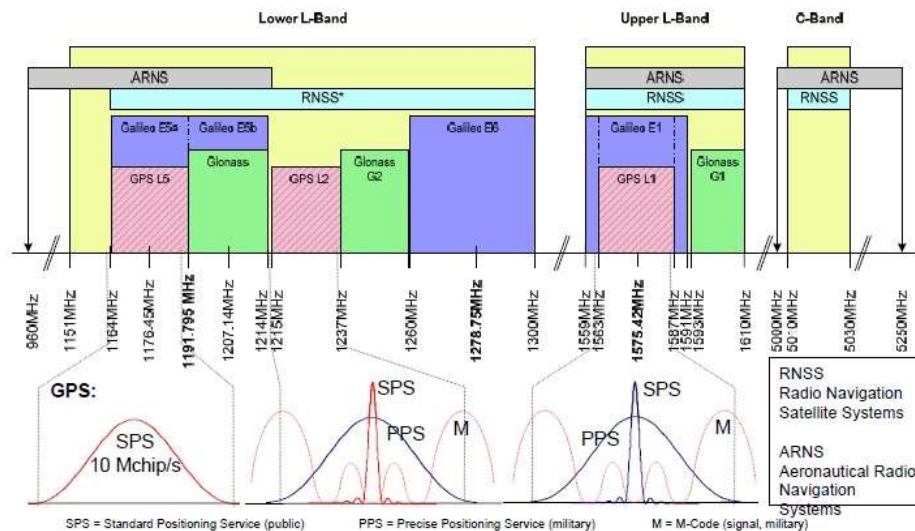
Το σύστημα έχει σε τροχιά 30 δορυφόρους, 24 απαιτούνται για να είναι λειτουργικό και 6 επιπλέον εφεδρικούς, σε 3 τροχιακά επίπεδα υψηλού υψομέτρου που δεν επιδρά το βαρυτικό πεδίο της γης, πράγμα που οδηγεί και σε υψηλή ακρίβεια, ακόμα και σε αστικό περιβάλλον με υψηλά κτίρια, της τάξεως του ενός μέτρου για τους χρήστες του δημόσιου σήματος και ακρίβεια χιλιοστού για την εμπορική χρήση, η οποία προσφέρεται επί πληρωμή, κρυπτογραφημένη, με έλεγχο ακεραιότητας και πιστοποιημένο σήμα για εφαρμογές που απαιτούν αυξημένη αξιοπιστία.

Επίσης προσφέρει υπηρεσίες safety of life navigation για εφαρμογές όπου η ακρίβεια είναι απαραίτητη για την δημόσια ασφάλεια με δυνατότητα μηνύματος ακεραιότητας που προειδοποιούν για τυχόν σφάλμα σήματος. Η δημόσια ρυθμιζόμενη πλοήγηση είναι κρυπτογραφημένο σήμα με συνεχή διαθεσιμότητα ακόμα και αν οι υπόλοιπες υπηρεσίες έχουν απενεργοποιηθεί, προσφέρεται αποκλειστικά σε κρατικές υπηρεσίες. Έρευνα και διάσωση, είναι η υπηρεσία εντοπισμού θέσης άμεσης προσοχής σε σχεδόν πραγματικό χρόνο με δυνατότητα αποστολής μηνύματος επιβεβαίωσης, όπως για παράδειγμα ενημέρωση για βοήθεια καθόδων.

Ο κάθε δορυφόρος του Galileo εκπέμπει 6 RHCP (Right-Handed Circularly Polarized) σήμα για την υποστήριξη των διαφορετικών υπηρεσιών που υλοποιεί, με πολύ μεγάλο εύρος σε κάποιες περιπτώσεις έως και 70 MHz. Αναλυτικά οι παρεχόμενες υπηρεσίες αποτελούνται από:

- L1F για ανοικτή, ελεύθερη, υπηρεσία και safety of life
- L1P για χρήση από κυβερνήσεις
- E6C για εμπορική χρήση, επι πληρωμή
- E6P για χρήση από κυβερνήσεις
- E5a για ανοικτή υπηρεσία
- E5b για ανοικτή υπηρεσία, εμπορική χρήση, safety of life

Τα σήματα αυτά καταλαμβάνουν τρεις ξεχωριστές συχνότητες από 1.1 GHz έως 1.5 GHz, ενώ επικαλύπτονται σε δύο συχνότητες με σήματα του GPS. Αυξημένη ακρίβεια λόγω της παραπάνω πολιτικής απολαμβάνουν οι χρήστες της ανοιχτής υπηρεσίας, καθώς τα πολλαπλά σήματα παρέχουν



Εικόνα 2-2 Αναλυτικά οι συχνότητες που καταλαμβάνουν τα συστήματα Galileo, Glonass, GPS

προστασία από αντανakλάσεις ειδικότερα σε αστικό περιβάλλον. Για την αξιοποίηση όλων των δυνατοτήτων του συστήματος απαιτούνται δέκτες με εξελιγμένους και περίπλοκους αλγόριθμους, όπως επίσης από την πλευρά του επίγειου συστήματος ελέγχου, ακριβής παρακολούθηση της τροχιάς των δορυφόρων με επιτρεπτό σφάλμα θέσης τα 65 εκατοστά, ακριβής συγχρονισμός ρολογιού με σφάλμα της τάξης του 1.5ns για κάθε 100 λεπτά και διασφάλιση υψηλής διαθεσιμότητας της υπηρεσίας σε επίπεδο 99.5%.

Όλες οι παραπάνω απαιτήσεις στην ποιότητα της παρεχόμενης υπηρεσίας, των επιμέρους συστημάτων διαστήματος, επίγειου ελέγχου και εξελιγμένων δεκτών χρήστη, η ποικιλία και η κρισιμότητα των εφαρμογών του το καθιστούν το πιο εξελιγμένο σύστημα θεσιγνωσίας που έχει σχεδιαστεί.

2.6 Glonass

Αναπτύχθηκε την δεκαετία του 1970 από επιστήμονες της Σοβιετικής Ένωσης για την αντικατάσταση του Tsiklon, ενός συστήματος θέσης 31 δορυφόρων και υψηλής ακρίβειας για αργά κινούμενο ή στατικό δέκτη, λόγω της μεγάλης απαίτησης χρόνου για τον υπολογισμό της θέσης σε κινούμενο δέκτη. Ολοκληρώθηκε στα τέλη της δεκαετίας του 1970 και αποτελείται από αστερισμό 24 δορυφόρων σε Μέση Γήινη Τροχιά, ακρίβεια 10 μέτρων και ήταν ικανό για τον εντοπισμό της θέσης, της κατεύθυνσης, της ταχύτητας του δέκτη λαμβάνοντας σήμα από 4 δορυφόρους.

Κατά την δεκαετία του 1990 και λόγω της οικονομικής και πολιτικής κρίσης της Σοβιετικής Ένωσης και μετέπειτα συρρίκνωσης της σε Ρωσική δημοκρατία η λειτουργικότητα του Glonass περιορίστηκε στη ρωσική επικράτεια με 13 δορυφόρους αφού για την διατήρηση του παγκόσμιου βεληνεκού του συστήματος απαιτούσε 3 εκτοξεύσεις Glonass δορυφόρων ανα έτος, κόστος που δεν μπορούσε να επωμιστεί η τότε ρωσική διαστημική υπηρεσία.

Έχοντας φτάσει τους 6 λειτουργικούς δορυφόρους το 2001, ιστορικό χαμηλό για το σύστημα, αποφασίζεται η πλήρης αποστρατικοποίηση του προγράμματος και μεταφορά του από το υπουργείο άμυνας στη διαστημική υπηρεσία όπου εκπονείται σχέδιο αποκατάστασης σε παγκόσμιας εμβέλειας σύστημα μέχρι το 2009. Πράγμα το οποίο επιτυγχάνεται στα τέλη του 2011 μετά από 15 χρόνια περιορισμένης λειτουργικότητας.

Σήμερα το σύστημα απαιτεί 18 δορυφόρους για την κάλυψη της Ρωσικής επικράτειας και 24 για

παγκόσμια κάλυψη και έχει ακρίβεια 4.46 – 7.38 μέτρα. Μπορεί να συνδυαστεί σε υβριδικές συσκευές που υποστηρίζουν το GLONASS και το GPS για καλύτερο συγχρονισμό και ακρίβεια σε αστικό περιβάλλον. Υπάρχει σε εξέλιξη σχεδιασμός επέκτασης του συστήματος με επίγειους σταθμούς διόρθωσης σήματος, στα πρότυπα του differential GPS, σε όλη την ρωσική επικράτεια και στο νότιο ημισφαίριο για αύξηση της ακρίβειας θέσης στα 0.6 μέτρα μέχρι το 2020. Ενώ το GPS χρησιμοποιεί μια συχνότητα εκπομπής στα 1575.42Mhz σε όλους τους δορυφόρους, το GLONASS χρησιμοποιεί για τους 24 δορυφόρους τους 12 συχνότητες με κοινή συχνότητα αυτή των αντιπόδων δορυφόρων. Ο χρόνος παρέχεται μέσα στο μήνυμα που στέλνει ο δορυφόρος και βασίζεται στην ατομική χρονική κλίμακα και ρυθμίζεται περιοδικά για την διόρθωση του εμβόλιμου δευτερολέπτου.

Το τμήμα ελέγχου, αποτελείται από το κέντρο ελέγχου συστήματος και ένα δίκτυο σταθμών παρακολούθησης σήματος στη Ρωσική επικράτεια. Παρακολουθεί την κατάσταση των δορυφόρων, καθορίζει τη θέση που πρέπει να καταλαμβάνουν οι δορυφόροι, τις μετατοπίσεις χρόνου και δυο φορές την ημέρα ανεβάζουν τα δεδομένα πλοήγησης. Το τμήμα διαστήματος, αποτελείται από 24 δορυφόρους σε 3 τροχιακά επίπεδα, 8 ανα επίπεδο και τέλος το τμήμα χρήστη, αποτελείται από τον δέκτη του σήματος, το οποίο πρέπει να είναι ικανό να ανιχνεύσει σήματα από τουλάχιστον 4 δορυφόρους για να λάβει ακριβή θέση, ταχύτητα, κατεύθυνση και μέτρηση χρονισμού. Επειδή το Glonass ακολουθεί την πολιτική του GPS στο θέμα της πρόσβαση σήματος η πολιτική χρήση υπόκειται στους ανάλογους περιορισμούς σε σχέση με το στρατιωτικό σήμα.

2.7 BeiDou

BeiDou σημαίνει Μεγάλη Άρκτος στην κινεζική γλώσσα και έχει σχεδιαστεί μετά από την αποχώρηση της Κίνας από το ευρωπαϊκής σύλληψης σύστημα θεσηγνωσίας Galileo. Το σύμβολο του συστήματος αντλεί την έμπνευσή του από το αρχαίο κινεζικό σύμβολο του Yin/Yang, σε σκούρο και ανοιχτό μπλε για τον ουρανό και την γη αντίστοιχα, ενώ δεσπόζουσα θέση έχει ο αστερισμός της Μεγάλης Άρκτου που συνδέεται επίσης με την αρχαία κινεζική παράδοση της πλοήγησης των πρώιμων κινεζικών πολιτισμών στον νυχτερινό ουρανό.

Αυτήν την περίοδο παρέχεται κάλυψη μόνο της ανατολικής Ασίας, δηλαδή την ευρύτερη περιοχή της Κίνας, της Ινδία έως και την Αυστραλία και υπολογίζεται ότι μέχρι το 2020 η κάλυψη θα είναι παγκόσμια.

Τις προδιαγραφές του BeiDou navigation satellite System, BDS εν συντομία, τις ορίζει η κινεζική υπηρεσία διαστήματος στο Interface Control Document (ICD) περιέχει αναλυτικά τη δομή των μηνυμάτων πλοήγησης, παραμέτρους για τα αλμανάκ του αστερισμού (satellite almanacs), τις θέσεις των δορυφόρων (ephemerides), τις σχέσεις μεταξύ της διεπαφής διαστημικού σήματος του BDS και του τερματικού δέκτη χρήστη, ενώ παρέχει τις βασικές τεχνικές προδιαγραφές για την ανάπτυξη και παραγωγή δεκτών και ολοκληρωμένων κυκλωμάτων.

Στην πλήρη ανάπτυξη του συστήματος ο αστερισμός θα απαρτιθεί 35 συνολικά δορυφόρους εκ των οποίων οι 5 γεωστατικοί υψηλής τροχιάς, 30 μέσης και κεκλιμένης γεωσύγχρονης τροχιάς, 27 και 3 δορυφόροι αντίστοιχα. Οι γεωστατικοί βρίσκονται σε υψόμετρο 35.786 Km με θέσεις 58.75°E, 80°E, 110.05°E, 140°E, 160°E, όπως και οι κεκλιμένοι με κλίση 55°, ενώ οι μέσης τροχιάς βρίσκονται σε υψόμετρο 21.528 Km με κλίση 55°.

Η ακρίβεια του συστήματος χωρίζεται στην δημόσια δωρεάν χρήση με σφάλμα ακτίνας 10 μέτρων και χρόνου 10 nsec, ενώ υπάρχει πρόβλεψη για κρυπτογραφημένη κυβερνητική ή στρατιωτική χρήση με ακρίβεια θέσης 10 cm. Τα φέροντα σήματα αυτών των δύο υπηρεσιών, είναι για την ανοικτή υπηρεσία B1I 1561.098 MHz, ενώ για την κρυπτογραφημένη B2I 1207.14 MHz. Το σήμα B1I περιλαμβάνει ένα εισαγωγικό του συστήματος, πρότυπο σήματος και το μήνυμα πλοήγησης που ορίζει το σχετικό περιεχόμενο του B1I στο τερματικό.

Το σήμα διαμορφώνεται με τεχνική Quadrature Phase Shift Key (QPSK), εκπέμπεται σύμφωνα με right-handed circularly polarized (RHCP), ενώ με πολυπλεξία Code Division Multiple Access (CDMA). Ο χρόνος συστήματος μετριέται σε BeiDou Time (BDT) με έναρξη χρονικής εποχής 00:00:00 Ιανουαρίου, 2006 UTC μέσω UTC(NTSC), με σφάλμα 100 nsec, ενώ όσον αφορά τα η διόρθωση εμβόλιμων δευτερόλεπτων αυτή εκπέμπεται μέσα στο μήνυμα πλοήγησης.

2.8 Συνδυασμός Συστημάτων Glonass/GPS

Η πληθώρα υπηρεσιών θεσιγνωσίας, ανοικτών και ελεύθερων, από διαφορετικούς πάροχους – χώρες μπορεί να αξιοποιηθεί προς όφελος των εφαρμογών που δημιουργούνται γύρω από αυτές ώστε να επιτευχθεί περαιτέρω αξιοπιστία, ακρίβεια σήματος καθώς και στην λύση γνωστών προβλημάτων σε γεωγραφικές τοποθεσίες όπου η απρόσκοπτη πρόσβαση στον ανοιχτό ουρανό, άρα στην πρόσβαση των δορυφόρων της εκάστοτε υπηρεσίας, δεν είναι δυνατή όπως τα πολύ μεγάλα αστικά κέντρα με τα υψηλά κτίρια, δάση ή φυσικά φαράγγια.

Η πρόκληση που αντιμετωπίζουν οι σχεδιαστές τέτοιων συστημάτων, σε επίπεδο χρήστη, είναι ο συνδυασμός των ετερόκλιτων δορυφορικών συστημάτων σε έναν δέκτη με τα παρακάτω οφέλη

- αυξημένη ακρίβεια παρατήρησης δορυφορικού σήματος
- αυξημένη χωρική διανομή της ορατότητας των δορυφόρων
- μειωμένη οριζόντια και κάθετη απόσβεση της ακρίβειας (Dilution of Precision - DOP)
- γρηγορότερη ανίχνευση της θέσης σε πραγματικό χρόνο

Γενικά για την σωστή ανίχνευση της θέσης με το GPS χρειαζόμαστε τουλάχιστον 4 δορυφόρους αντιπροσωπεύοντας τις 3 διαστάσεις και τον χρόνο, ενώ συνδυαστικά μας αρκούν 5 με τον επιπλέον να μας καθορίζει τη χρονική μετατόπιση των συστημάτων. Με τον συνδυασμό Glonass/GPS ο χρήστης επωφελείται συνολικά με πάνω από 48 δορυφόρους επιφέροντας ακρίβεια σε γεωγραφικά σημεία όπου ένα από τα δύο συστήματα δεν θα μπορούσε να ανταποκριθεί όπως για παράδειγμα σε πόλεις με πολύ ψηλά κτήρια ή περιοχές με μειωμένη ορατότητα. Ένας μεγαλύτερος αστερισμός δορυφόρων έχει ως αποτέλεσμα την δραστική βελτίωση της ανίχνευσης θέσης σε πραγματικό χρόνο και με αυξημένη ακρίβεια.

2.9 Τεχνικά Χαρακτηριστικά Συστημάτων

Παράμετρος	Λεπτομέρειες	GLONASS	GPS	GALILEO	
Δορυφόροι	Αριθμός δορυφόρων	21+3 εφεδρικοί	21+3 εφεδρικοί	24 + 6 εφεδρικοί	
	Αριθμός τροχιακών επιπέδων	3	6	3	
	Κλίση τροχιακών επιπέδων (μοίρες)	64.8	55	56	
	Ακτίνα τροχιάς (χιλιόμετρα)	25510	25560		
Σήματα	Θεμελιώδης συχνότητα ρολογιού(MHz)	5.0	10.23		
	Τεχνική διαχωρισμούς σήματος	FDMA	CDMA	CDMA	
	Συχνότητα φέρον (MHz) L1		1598.0625 - 1609.3125	1575.42	1176.45 - 1575.42
		L2	1242.9375 - 1251.6875	1227.6	
	Κωδικός ρυθμό του ρολογιού C/A		0.511	1.023	
		P	5.11	10.23	
	Μήκος κωδικού C/A		511	1023	
P		5.11×10^6	6.187104×10^{12}		
C/A-κωδικός Μηνύματος Πλοήγησης	Διάρκεια Superframe (λεπτά)	2.5	12.5	12	
	Χωρητικότητα Superframe (bits)	7500	37500		
	Εφεδρική χωρητικότητα Superframe(bits)	~620	~2750		
	Διάρκεια λέξης (δευτερόλεπτα)	2.0	0.6		

Χωρητικότητα λέξης (bits)	100	30	
Αριθμός λέξεων μέσα σε frame	15	50	
Τεχνική διαχωρισμού δορυφόρων θέση δορυφόρων	Γεωκεντρικό καρτεσιανό συντεταγμένες και τα παράγωγά τους	Κεπλεριανό τροχιακό Στοιχεία/παράγοντες διαταραχής	(Synchronised 30ns) UTC
Χρόνος αναφοράς	UTC (SU)	UTC (USNO)	
Θέση αναφοράς (γεωδατικά δεδομένα)	PZ-90	WGS84	

2.10 IP-geolocation

Η διαδικασία μετατροπής μιας IP διεύθυνσης σε πραγματική τοποθεσία. Υπάρχουν πολλές δωρεάν και εμπορικές υπηρεσίες που το υλοποιούν αλλά δεν υπάρχει ακρίβεια στα αποτελέσματα.

3. Location Based Services

Εφαρμογές ή Υπηρεσίες με γνώση τοποθεσίας τροποποιούν το περιεχόμενο ανάλογα με την φυσική τοποθεσία του χρήστη. Για την αξιοποίηση αυτών των εφαρμογών/υπηρεσιών απαιτείται μηχανισμός εύρεσης θέσης όπως το GPS, υποδομή κυψελωτής τηλεφωνίας, ασύρματα access points.

3.1 Εφαρμογές Location Based Service

Η υιοθέτηση διαφόρων ασύρματων συσκευών με ενσωματωμένους δέκτες θεσηγνωσίας, όπως wearables, tablets, smartphones έχουν αυξηθεί με τα χρόνια. Παρόλα αυτά η ασφάλεια των προσωπικών δεδομένων αποτελεί την κύρια ανησυχία για την ανάπτυξη της αγοράς υπηρεσιών με χρήση θέσης, καθώς οι κινητές συσκευές ανιχνεύουν, καταγράφουν και παρέχουν πληροφορίες θέσης.

Οι κύριες κατηγορίες LBS είναι η πλοήγηση και ιχνηλασία, το marketing και η διαφήμιση, η ψυχαγωγία, η ανάκτηση πληροφοριών με βάση τη θέση και οι υπηρεσίες ασφάλειας και προστασίας.

Εφαρμογές στις υπηρεσίες βασισμένες σε τοποθεσία περιλαμβάνουν την τοπική διαφήμιση, gaming, την επαυξημένη πραγματικότητα, αυτοματοποιημένη υποστήριξη έκτακτης ανάγκης, διαχείριση καταστροφών, πληροφορίες κίνησης αυτοκινητόδρομων, οπτικοποίηση αντικειμένων και κτηρίων στον πραγματικό κόσμο, οδηγίες, εντοπισμός κλεμμένου αυτοκινήτου, υπηρεσίες πληροφοριών αυτοκινητόδρομων, χρέωση κατά περιοχή, κοινωνική δικτύωση συγκοινωνιών, παρακολούθηση άθλησης και υγείας. Πλοήγηση, στις μεταφορές εμπορευμάτων, μετακινήσεις ανθρώπων, λήψη οδηγιών, εύρεση επιχειρήσεων, υπολογισμός χρόνου διαδρομής και απόστασης από τελικό προορισμό.

Τοπογραφία, οριοθέτηση της ιδιοκτησίας γης για αρχιτέκτονες και πολιτικούς μηχανικούς για τον σχεδιασμό και την υλοποίηση κατασκευής έργων υποδομών.

Επιχειρηματική διαδικασία για εύρεση καταστημάτων, επιχειρήσεων κατά προτεραιότητα κοντά στην περιοχή μας, δηλαδή κοντά στη φυσική τοποθεσία του χρήστη. Αγορά τέτοιων στοιχείων από τις επιχειρήσεις με κόστος την ιδιωτικότητα των χρηστών.

Αποθήκευση και δρομολόγηση (eshop/delivery service) αναφορά τόπου χρόνου αντικειμένου, εισερχόμενα και εξερχόμενα αντικείμενα αποθήκης και υπολογισμός κόστους δρομολογίου/διοδίου.

Για τους καταναλωτές, επιτρέπει νέες εφαρμογές “πανταχού παρόν computing”, αυτόματη επαναρύθμιση των συσκευών ανάλογα με την τοποθεσία τους, δημοσίευση της θέσης τους σε social media, οι χρήστες αποκτούν αυτοπεποίθηση με την επιβεβαίωση της τρέχουσας θέσης τους, κοινοποίηση ειδικών προσφορών από λιανεμπόριο σε δυνητικούς πελάτες που βρίσκονται κοντά για περιορισμένο χρονικό διάστημα.

Απαραίτητες υποδομές, ανάπτυξη υποδομών σε επίπεδο κτιρίου όπως δωρεάν Wi-Fi ή RTLS για εντοπισμό θέσης μέσα σε χώρους κτιρίων, σε επίπεδο επικράτειας GSM, LTE και σε παγκόσμιο επίπεδο GPS, Galileo, search and rescue, combat search and rescue.

Αναλυτικά οι λύσεις διαχείρισης στόλου, προσφέρουν ενσωματωμένη καταγραφή δεδομένων, δορυφορικό εντοπισμό θέσης και επικοινωνία δεδομένων για την υλοποίηση υπηρεσιών όπως συντήρηση οχημάτων, χρηματοδότηση τηλεματικής οχήματος για διαγνωστικά και ανίχνευση, διαχείριση οδηγών και ταχύτητας, διαχείριση καυσίμων.

Τα οφέλη χρήσης ενός τέτοιου συστήματος από τις εταιρίες είναι η βελτιστοποίηση λειτουργικών εξόδων στόλου και εκπομπών διοξειδίου του άνθρακα (CO₂), κάλυψη της ανάγκης για ασύρματη επίβλεψη και ασφάλεια, συμμόρφωση με εθνικούς κανονισμούς για ασφάλεια στις μεταφορές και στους εκπεμπόμενους ρύπους.

Πρόκληση για τις επιχειρήσεις αποτελεί η ενσωμάτωση της τεχνολογίας που απαιτείται στους στόλους των οχημάτων και την υποδομή, λόγω κόστους, έλλειψης συνεχούς και αδιάληπτου δίαυλου επικοινωνίας, νομικά κολλήματα περί ιδιωτικότητας και παρακολούθησης θέσης, η έλλειψη διεθνούς τυποποίησης και ζητήματα ενημέρωσης για τα οφέλη αυτών των συστημάτων.

Στην επιχειρησιακή πληροφόρηση και ανάλυση βοηθάει στην απόδοση του χωρικού και χρονικού γενικού πλαισίου για ικανότητα λήψης αποφάσεων ερμηνεύοντας την περιέρρευσα ατμόσφαιρα, στην

αυξημένη παραγωγικότητα, στην ανακάλυψη νέων προοπτικών ανάπτυξης, στην διαχείριση αλυσίδας τροφοδοσίας, στην σχέση με τον πελάτη, στην δυνατότητα αναγνώρισης προτύπων και την χρήση τους στην βελτιστοποίηση επιχειρηματικών δραστηριοτήτων όπως σχεδιασμός αλυσίδας καταστημάτων, μεγάλος διαθέσιμος όγκος πληροφοριών με γεωγραφικό αποτύπωμα μέσω της χρήσης GPS με ελάχιστο κόστος, καλύτερες προοπτικές ανάλυσης με την άνοδο των Big Data για αυξημένη αφοσίωση πελατών. Σκεπτικισμός για την χρήση προσωπικών δεδομένων από επιχειρήσεις και προβληματισμός από την πλευρά των πελατών.

Η βασισμένη στη θέση διαφήμιση, χρησιμοποιεί την τρέχουσα θέση, χρόνο, ιστορικό και άλλες συναφείς με τον χρήστη πληροφορίες για την προβολή στοχευμένων διαφημίσεων. Αφορά πελάτες μιας συγκεκριμένης περιοχής ή σε περιοχή που βρίσκεται κοντά στο κατάστημα, λειτουργεί με γεωπεριφρακτικές ειδοποιήσεις και υπενθυμίσεις σε ορισμένη ακτίνα από τον ενδιαφέρον χώρο, ανάλυση ιστορικού περιοχής που προτείνει νέες υπηρεσίες και προϊόντα βασισμένα σε προηγούμενες επισκέψεις. Οι λόγοι ανάπτυξης είναι η αμεσότητα προσέγγισης του πελάτη και ευκαιρία μεγιστοποίησης της αποτελεσματικότητας της διαφημιστικής δαπάνης μέσω της επικέντρωσης σε προκαθορισμένη γειτονιά, η διείσδυση των έξυπνων συσκευών, η συνεχής συνδεσιμότητα, η υψηλότερη διαφημιστική επίδραση και βελτιωμένη ικανοποίηση του πελάτη από την κατανόηση των αναγκών του καταναλωτή σε πραγματικό χρόνο και τόπο, δυνατότητα πρόγνωσης μελλοντικών προτύπων βασισμένα στο τοπικό ιστορικό, υψηλή τοπική αναγνωσιμότητα σήματος και οικειότητα πελάτη.

Η οπτικοποίηση βασισμένη σε χάρτη, είναι η προσθήκη ενός επίπεδου δεδομένων και πληροφοριών πάνω σε χάρτη όπως μέρη προτίμησης, σχεδιαζόμενες διαδρομές, χωρική ανάλυση, δελτία καιρού και heat maps.

Στις πωλήσεις και διανομή, ο εντοπισμός και η παρακολούθηση προϊόντων και ενεργειών ενδυναμώνουν την οργάνωση και διαχείριση των επιχειρήσεων παρέχοντας μια κοινή διεπαφή για την σύνδεση γεωγραφικής ανάλυσης. Επίσης καίρια για τον επιχειρησιακό προγραμματισμό είναι η πληροφόρηση για τις τρέχουσες πωλήσεις σε όλη την αλυσίδα εμπορίου, τα διαθέσιμα αποθηκών καθώς και την αξιοποίηση των περιουσιακών στοιχείων του χαρτοφυλακίου της επιχείρησης.

Πολύ σημαντική είναι η εφαρμογή των Location Based Services (LBS) στον κυβερνητικό τομέα περιλαμβάνοντας δράσεις όπως ο σχεδιασμός και το αστικό τοπίο που οδηγεί στην έγκαιρη ανάληψη πρωτοβουλιών, διαχείριση οδικής κυκλοφορίας και προειδοποιήσεις εντάσσοντας κρίσιμες πληροφορίες για την βελτιστοποίηση της κυκλοφορίας οχημάτων, της κυκλοφοριακής συμφόρησης και παρακολούθησή τους, διαχείριση και ανίχνευση δημόσιων μέσων μεταφοράς με σχεδιασμό αποδοτικών διαδρομών και ένταξής τους στα μέσα μεταφοράς, σχεδιασμός ασφάλειας, προστασίας και υπηρεσιών εκτάκτου ανάγκης μέσω της αναγνώρισης, προετοιμασίας, γρήγορης αντίδρασης σε απειλές κατά της δημόσιας ασφάλειας, φυσικών καταστροφών, τρομοκρατικών επιθέσεων και θαλάσσια παρακολούθηση βοηθά τις λιμενικές αρχές στην λήψη έγκαιρων αποφάσεων σε όλες τις θαλάσσιες δραστηριότητες.

Στον σχεδιασμό και το αστικό τοπίο βοηθά στις μεγάλου κλίμακας παρεμβάσεις για πρωτοβουλίες στην διαχείριση των αποβλήτων, τοποθέτηση ηλεκτρικών γραμμών, οπτικών ινών, συνδέσεις νερού όπως επίσης βοηθούν τους κυβερνητικούς παράγοντες να κρατάνε αρχείο χρήσης γης, συγκεντρώσεις ρύπων, ενημέρωση της διαθεσιμότητας των δημόσιων υπηρεσιών, καθορισμό περιπολιών της αστυνομίας, παροχή χωρικής απεικόνισης για τον αστικό σχεδιασμό γεωπληροφοριακών συστημάτων (GIS) με οπτικοποίηση, μοντελοποίηση, ανάλυση χωρικής πληροφορίας και διευκόλυνσης της συνεργασίας μεταξύ πολλαπλών παραγόντων για μεγάλης κλίμακας εφαρμογές όπως περιφερειακές, περιβαλλοντικές, οικολογικές, διαχείριση φυσικών πόρων, αποτίμηση ιδιοκτησίας και σχεδιασμό υποδομών.

Τέτοιες υλοποιήσεις πληροφοριακών συστημάτων, απαντούν στους περιορισμούς των παραδοσιακών μεθόδων σχεδιασμού στο ραγδαία εξελισσόμενο αστικό τοπίο, οι αυξημένες απαιτήσεις πόρων στις πόλεις απαιτούν έξυπνους τρόπους επίλυσης σε διάφορα σενάρια, δεδομένα υπολογισμών με εναλλακτικές δυνατότητες δυναμικής εξέτασης και ανάλυσης της προβαλλόμενης πληροφορίας για πιο κατανοητές αναπαραστάσεις.

Τα γεωπληροφοριακά συστήματα στην διαχείριση οδικής κίνησης και ειδοποίησης, βοηθούν στην ένταξη πληροφοριών για τον κυκλοφοριακό σχεδιασμό, την κυκλοφοριακή συμφόρηση, παρέχει πρόσβαση σε διαδραστικούς χάρτες πληροφοριών πραγματικού χρόνου με οπτικοποίηση, οι αρχές μπορούν να εξομοιώσουν κυκλοφοριακά σενάρια για μικρής εμβέλειας χρονικό διάστημα πράγμα που

τους επιτρέπει την ακαριαία τροποποίηση της κυκλοφοριακής πληροφορίας με σχεδιασμό της κυκλοφοριακής σήμανσης για την αποτροπή συμφόρησης και απόφραξης. Δορυφορική παρακολούθηση των δρόμων για συμφόρηση και κυκλοφοριακές καθυστερήσεις, αναδιοργάνωση της κίνησης γύρω από το πρόβλημα.

Παρέχει επίσης δυνατότητα ανάλυσης προτύπων και ανωμαλιών για την βελτίωση της συμφόρησης εν γένει και του χρόνου ταξιδιού, ιδιότητα χρήσης οδικής πληροφόρησης για επεκτάσεις δρόμων ή μονοδρομήσεις, ανάλυση πιθανότητας τροχαίων ατυχημάτων, άμεση αντίδραση με πολιτικές δρομολόγησης και επιβολή κανονισμών.

Τροχοπέδη αποτελεί η χαμηλή συμβατότητα με τα υπάρχοντα συστήματα διαχείρισης της κυκλοφορίας και το πιθανό υψηλό κόστος της επένδυσης σε υποδομές γεωπληροφοριακών συστημάτων.

Το σύστημα διαχείρισης δημοσίων μεταφορών, συνιστάται από συσκευές δεκτών εντοπισμού, GPS, που βρίσκονται εγκατεστημένες στα οχήματα της υπηρεσίας στέλνοντας την θέση τους, τον χρόνο και πληροφορίες για την ταχύτητα τους στο ενοποιημένο σύστημα γεωπληροφοριών. Βοηθά τις αρχές στην σχεδίαση αποδοτικών διαδρομών αξιοποιώντας τον μεγάλο όγκο πληροφοριών θέσης, στον εντοπισμό και την παρακολούθηση υπηρεσιών, στην ασφάλεια και προστασία του συστήματος, συνεισφέροντας στην παροχή αποδοτικότερων δημόσιων μεταφορών για τους πολίτες με δυναμικές, αξιόπιστες, εύρωστες και οικονομικές πληροφορίες για τις διαδρομές με τον περισσότερο κόσμο σε ώρες αιχμής, ποιες επηρεάζονται από την κυκλοφοριακή συμφόρηση και σε ποιες ώρες. Επίσης επιτρέπει στους επιβάτες την παρακολούθηση της θέσης των λεωφορείων υπολογίζοντας τον εκτιμώμενο χρόνο για την έλευση των οχημάτων απευθείας στο κινητό ή στον web browser μειώνοντας την αβεβαιότητα των μεταφορών.

Στον τομέα του σχεδιασμού προστασίας και υπηρεσιών άμεσης επέμβασης, επιτρέπει στις αρχές την οπτικοποίηση των προτύπων ενεργειών, πολυεπίπεδες χωροταξικές συνθήκες σε πραγματικών χρόνο για τον εντοπισμό, προετοιμασία και γρήγορη αντίδραση σε απειλές ασφαλείας, φυσικές καταστροφές, τρομοκρατικές ενέργειες, δολιοφθορές με μείωση του ρίσκου έκθεσης και σχεδίασης αποκατάστασης.

Παρέχουν στις χώρες ή πόλεις υπηρεσίας άμεσης δράσης, αυτόματου εντοπισμού θέσης και άμεσης επέμβασης όποτε οι πολίτες καλέσουν βοήθεια. Με την ανάλυση της χωρικής τάσης μιας περιοχής προβάλλουμε τις αδυναμίες και ζημιές για τον σχεδιασμό άμεσων δράσεων και ανακουφιστικών δράσεων προκαταβολικά.

Στην θαλάσσια παρακολούθηση, βοηθά τις λιμενικές αρχές στην λήψη έγκαιρων αποφάσεων για τις πιθανές θαλάσσιες δραστηριότητες συμπεριλαμβανομένων της ασφάλειας και της προστασίας, έλεγχο των συνόρων, επιβολή των νόμων, αμυντικές δράσεις, την προστασία του θαλάσσιου περιβάλλοντος και οικοσυστημάτων, τον έλεγχο της αλιείας, του εμπορίου και της οικονομικής δραστηριότητας. Βελτιώνει τις υπηρεσίες κάνοντας χρήση γραφικών απεικονίσεων για την περίληψη των δραστηριοτήτων συγκεκριμένων περιοχών και χρονικών περιόδων, παρέχει online σχεδιαγράμματα ναυσιπλοΐας και ηλεκτρονική υποστήριξη δεδομένων για επιχειρήσεις πελατών και προμηθευτές, αναπαράσταση διαδρομών εμπορευμάτων κατά παραγγελία και διαχείριση φόρτου λιμένων με εξομοίωση των διαδρομών των πλοίων.

Η υιοθέτηση των LBS για το καταναλωτικό κοινό, χρησιμοποιεί την πληροφορία θέσης για την διαμόρφωση του διαδικτυακού περιεχομένου και των κινητών πληροφοριών σε σχέση με την τοποθεσία που βρίσκεται ο χρήστης, συμβάλλοντας παράλληλα στην δημιουργία υπηρεσιών διαμορφούμενες από δεδομένα πλήθους χρηστών (crowdsourcing).

Παράγοντες ανάπτυξης τους σε αυτό το πλαίσιο αποτελεί η εξάπλωση των smart συσκευών με ενσωματωμένο δέκτη GPS σε χαμηλές τιμές, δωρεάν διάθεση υπηρεσιών θεσηγνωσίας, αυξημένης επίγνωσης των οφελών LBS με εφαρμογές χάρτη ή αναζήτησης, απρόσκοπτη ενσωμάτωση και εξάπλωση υπηρεσιών όπως η κοινωνική δικτύωση.

Στις υπηρεσίες χάρτη και φωνητικής καθοδήγησης, παρέχεται η δυνατότητα πλοήγησης, τοπική αναζήτηση, προειδοποίηση για κυκλοφοριακή συμφόρηση του οδικού δικτύου, πρόγνωση συμφόρησης από δεδομένα παλαιότερης χρονικής περιόδου, δρομολόγηση.

Οι περισσότεροι χρήστες, χρησιμοποιούν κάποιου είδους έξυπνης συσκευής για την πλοήγησή τους σε σχέση με το παρελθόν που έκαναν χρήση εξειδικευμένων συσκευών, με αποτέλεσμα το Google

Maps να είναι η δημοφιλέστερη υπηρεσία, ενώ ακόμα και οι σταθεροί ηλεκτρονική υπολογιστές χωρίς ενσωματωμένο δεκτή GPS μπορούν να επωφεληθούν από την τρέχουσα τοποθεσία τους, κατά προσέγγιση βέβαια, μέσω γεωδαιτικών APIs (Application Programming Interfaces) από την σύνδεση με WiFi.

Οι υπηρεσίες χάρτη παρέχουν, επιπλέον, τη δυνατότητα σε τρίτους κατασκευαστές την δημιουργία εφαρμογών σύμφωνα με τις ανάγκες τους.

Υπολογίζεται ότι υπήρχαν 400 εκατομμύρια ενεργοί χρήστες υπηρεσιών πλοήγησης το 2013, ενώ οι κύριοι πάροχοι Google και Nokia συλλέγουν πληροφορίες από χρήστες για τον εμπλουτισμό και την επικαιροποίηση των εφαρμογών τους, παράλληλα άλλες υπηρεσίες όπως η OpenStreetMaps και Waze, βασίζονται σχεδόν εξολοκλήρου στην ανάδραση των χρηστών και αναμένουν βελτίωση της θέσης τους στην αγορά.

Στην εξερεύνηση και τοπική αναζήτηση, η αυξημένη καινοτομία των εταιριών με χρήση τεχνικών κοινωνικής δικτύωσης αλλά και το ιστορικό του χρήστη με τα ενδιαφέροντα του, οι αξιολογήσεις χρηστών και το σύνολο των check-ins μιας τοποθεσίας, βοηθούν τους υπόλοιπους χρήστες στο σχηματισμό εικόνας μιας επιχείρησης ή ενός εμπορικού κέντρου.

Ήδη το Facebook, το Instagram και άλλα μέσα κοινωνικής δικτύωσης χρησιμοποιούν το υλικό που δημιουργούν οι χρήστες τους για την ανάδειξη δημοφιλών χώρων, ενώ αυτοί με την σειρά τους χτίζουν καλύτερη και πληρέστερη ταυτότητα για τις υπηρεσίες τους, κάνοντάς τες προσβάσιμες σε μεγαλύτερο κοινό.

Οι εφαρμογές επαυξημένης πραγματικότητας, είναι μια νέα τεχνολογία που προσφέρει τρόπους ανάμειξης ροής βίντεο με πληροφορίες σε πραγματικό χρόνο με την αξιοποίηση τεχνολογιών τοποθεσίας, camera και άλλους αισθητήρες για την αναγνώριση αντικειμένων και χώρων παρεμβάλλοντας σε αυτές ψηφιακές πληροφορίες από πάνω.

Επίσης έχει εφαρμογή στο gaming μέσω της παροχής νέων τρόπων ελέγχου των αποφάσεων του παίχτη, από φυσική κίνηση ή θέση. Ο δούρειος ίππος για την εξάπλωση της τεχνολογίας είναι οι εφαρμογές στα smartphones όπου αναπτύσσονται με πολύ χαμηλό κόστος και μεγάλη ευκολία, παράλληλα με την εξάπλωση των wearable συσκευών που φέρουν τους κατάλληλους αισθητήρες και το κατάλληλο λειτουργικό σύστημα.

3.2 Σκεπτικισμός LBS

Ιδιωτικότητα (privacy), είναι η δυνατότητα του ελέγχου των προσωπικών δεδομένων. Ο καθένας μπορεί να καθορίζει την πρόσβαση τρίτων για προσωπικά ζητήματα, κοινωνικής ζωής, υγείας και επαγγελματικών.

Η ιδιωτικότητα τοποθεσίας (location privacy), είναι η δυνατότητα του οποιουδήποτε να κινείται σε δημόσιους χώρους με την προσδοκία, σε κανονικές συνθήκες, η τοποθεσία τους να μην καταγράφεται συστηματικά και χωρίς την συναίνεση του για χρήση σε δεύτερο χρόνο.

Συλλογή πληροφοριών (concern for information privacy), ανησυχία μαζικής αποθήκευσης προσωπικών δεδομένων σε βάσεις δεδομένων.

Δευτερεύουσα χρήση χωρίς συγκατάθεση, συλλογή των πληροφοριών από τρίτους με στόχο την περαιτέρω επεξεργασία τους για άλλο σκοπό από την αρχική χρήση.

Σφάλματα, η ανησυχία ότι η προστασία έναντι των εκουσίων ή ακουσίων σφαλμάτων είναι ανεπαρκής.

Ακατάλληλη πρόσβαση, η ανησυχία της σκόπιμης, τυχαίας ή άμεσης διαθεσιμότητας σε άτομα χωρίς το δικαίωμα εμφάνισης ή επεξεργασίας των πληροφοριών αυτών.

Απόρρητη Κυβερνητική Χρήση, από κυβερνητικές υπηρεσίες για εξυπηρέτησης δικών τους σκοπών.

Κακόβουλη χρήση προσωπικών δεδομένων από ανταγωνιστές, παρακολούθηση, κακοπροαίρετα άτομα.

Πέντε βασικοί παίχτες αναμιγνύονται στην διαχείριση της ιδιωτικότητας των LBS

- εταιρίες κινητής τηλεφωνίας

- κατασκευαστές κινητής τηλεφωνίας
- location aggregators
- τρίτοι κατασκευαστές λογισμικού
- διαδικτυακές εταιρίες

Ρυθμίσεις της Ευρωπαϊκής Ένωσης με αναγκαστική οδηγία (95/46/EC, 97/66/EC και 2002/58/EC) σε τρεις βασικούς άξονες:

1. Γνωστοποίηση, οποιαδήποτε εταιρία συλλέγει προσωπικά δεδομένα θέσης πρέπει να γνωστοποιεί στον καταναλωτή τι είδους δεδομένα συλλέγει και για ποιον σκοπό και χρήση. Η διαφάνεια του συλλέκτη είναι η κεντρική αρχή.
2. Συναίνεση, ο συλλέκτης προσωπικών δεδομένων θέσης πρέπει να αποσπά την συγκατάθεση του χρήστη πριν τη συλλογή αυτών.
3. Ασφάλεια δεδομένων, τα δεδομένα που συλλέγονται πρέπει να προστατεύονται από επαρκή μέτρα ασφάλειας ενάντια στην τυχαία διαρροή ή κλέψιμο, γνωστοποίηση, καταστροφή, παράνομη επεξεργασία ή οτιδήποτε παρόμοιο. Η αρχειοθέτηση προσωπικών δεδομένων θέσης μπορεί να γίνει μόνο με την ρητή έγκριση του χρήστη.

Αποφυγή νομικών συνεπειών:

- δεν συλλέγουμε δεδομένα από την αρχή
- εάν πρέπει να συλλέξουμε δεδομένα δεν τα αποθηκεύουμε
- εάν υπάρχει πραγματική ανάγκη για αποθήκευση, τα διατηρούμε χωρίς ονόματα
- εάν οπωσδήποτε αποθηκεύουμε προσωπικά δεδομένα θέσης και αντέχουμε οικονομικά, τα κρυπτογραφούμε

3.3 Απαιτήσεις LBS

Κατηγορίες LBS	Απαιτήσεις
Πλοήγηση και ιχνηλασία	<ul style="list-style-type: none"> ○ αυξημένη διαθεσιμότητα ○ απόκριση σε πραγματικό χρόνο ή λίγων δευτερολέπτων ○ ακρίβεια μερικών μέτρων ή λιγότερο ○ υψηλή αξιοπιστία και συνέχεια ○ μέση με χαμηλή κατανάλωση ενέργειας ○ λογικές ή χαμηλές τιμές ○ διατήρηση ιδιωτικότητας
Marketing/Διαφήμιση	<ul style="list-style-type: none"> ○ μέση διαθεσιμότητα ○ απόκριση λίγων λεπτών ○ ακρίβεια μερικών εκατοντάδων μέτρων ○ μέση αξιοπιστία και συνέχεια ○ πολύ χαμηλή κατανάλωση ○ σχεδόν δωρεάν ή πολύ φθηνά ○ μέση διατήρηση ιδιωτικότητας
Διασκέδαση	<ul style="list-style-type: none"> ○ μέση με υψηλή διαθεσιμότητα ○ απόκριση σε πραγματικό χρόνο ή λίγων δευτερολέπτων ○ ακρίβεια μερικών δεκάδων μέτρων ○ χαμηλή κατανάλωση ○ λογικές ή χαμηλές τιμές ○ μέση διατήρηση ιδιωτικότητας
Ανάκτηση πληροφοριών βάση θέσης	<ul style="list-style-type: none"> ○ μέση διαθεσιμότητα ○ απόκριση σε πραγματικό χρόνο ή λίγων δευτερολέπτων ○ ακρίβεια μέτρων έως μερικών εκατοντάδων μέτρων ○ υψηλή αξιοπιστία και συνέχεια

	<ul style="list-style-type: none"> ○ χαμηλή κατανάλωση ○ λογικές ή χαμηλές τιμές ○ μέση διατήρηση ιδιωτικότητας
Ασφάλεια και Προστασία	<ul style="list-style-type: none"> ○ υψηλή διαθεσιμότητα ○ απόκριση σε πραγματικό χρόνο ή λίγων δευτερολέπτων ○ ακρίβεια μερικών δεκάδων μέτρων ○ πολύ υψηλή αξιοπιστία και συνέχεια ○ χαμηλή κατανάλωση ○ λογικές ή χαμηλές τιμές ○ μέση με χαμηλή διατήρηση ιδιωτικότητας

4. Maps API

Τα εργαλεία που υπάρχουν διαθέσιμα σήμερα για τους web/app developers δίνουν επαρκείς λύσεις και επεκτείνουν την λειτουργικότητα των σύγχρονων εφαρμογών με κόστος μικρό έως μηδενικό. Όλες οι μεγάλες εταιρίες του χώρου της πληροφορικής Microsoft, Yahoo!, Google αξιοποιούν την τάση για location-aware εφαρμογές με στόχο την επιβολή των δικό τους προτύπων και πρόσβαση σε πληροφορίες θέσης υψηλής αξίας.

4.1 Google Maps APIs

Google Maps API, γραμμένο σχεδόν εξολοκλήρου σε JavaScript και XML. Πριν την επίσημη έκδοση του API κάποιοι χρήστες βρήκαν τρόπο να ενσωματώσουν δικά τους client-side scripts και server-side hooks ώστε να δημιουργήσουν και να επεκτείνουν με νέα χαρακτηριστικά τις δυνατότητες ανάλογα με τις ανάγκες των υπηρεσιών τους χρησιμοποιώντας την βασική μηχανή και τις δορυφορικές εικόνες που διοχέτευαν οι server της Google μπόρεσαν να ενσωματωνόσουν προσαρμοσμένες δείκτες θέσης, συντεταγμένες και metadata περιοχών ακόμα και επεξεργασμένες εικόνες χαρτών μέσα στο Google Maps Interface.

Μετά την εξάπλωση αυτών των τεχνικών παρουσιάστηκε επίσημα το API τον Ιούνιο 2005 το οποίο επιτρέπει στους web developers να το ενσωματώσουν δωρεάν στις υλοποιήσεις τους, σε συνδυασμό με τα δικά τους δεδομένα από τα web services.

4.2 Google Maps Android API

Μια σειρά από βιβλιοθήκες για ανάπτυξη πλήρους λειτουργικής ενσωμάτωσης των google χαρτών σε οποιοδήποτε android app. Είναι γραμμένο σε JAVA και είναι αντίστοιχης φιλοσοφίας με το Maps API v3 σε JavaScript για web-based περιβάλλοντα.

4.3 Yahoo! Maps

Είναι μια δωρεάν υπηρεσία χάρτη από την Yahoo! με ενσωματωμένη λειτουργικότητα καιρού, εκτύπωσης χαρτών, ζωντανής μετάδοσης της κατάστασης της κυκλοφοριακής συμφόρησης, οδηγίες πλοήγησης μέσω κειμένου αλλά και σημείου προς σημείο ενώ για τους εγγεγραμμένους χρήστες υπάρχει η δυνατότητα του βιβλίου διευθύνσεων για τις πιο συχνά χρησιμοποιούμενες τοποθεσίες.

Μετά από χρόνια εγκατάλειψης της υπηρεσίας και την μη ανανέωση των δορυφορικών φωτογραφιών και των πληροφοριών χάρτη, η Yahoo! από το 2013 συνεργάζεται με την Nokia και την δικής υπηρεσία χαρτών Here Maps.

Η προγραμματιστική διεπαφή είναι διαθέσιμη σε 3 εκδοχές για υλοποίηση με την βοήθεια Adobe Flash το οποίο επιτρέπει χρήση JavaScript, ActionScript, Adobe Flex 1.5 και απαιτεί flash player plug-in, με την βοήθεια AJAX χωρίς να απαιτείται κάποιου είδους plug-in και οι εφαρμογές γράφονται σε JavaScript και το Simple API που βασίζεται σε XML data format αποτελώντας μια επέκταση του GeoRSS για την εμφάνιση των πληροφοριών στον χάρτη.

4.4 OpenLayers

Είναι μια ανοιχτού κώδικα βιβλιοθήκη γραμμένη σε JavaScript για να εμφανίζει δεδομένα χαρτών σε web browser. Χρησιμοποιεί τεχνολογίες όπως GeoRSS, KML το XML των δεδομένων χαρτών, GeoJSON και δεδομένα χαρτών από οποιαδήποτε πηγή χρησιμοποιεί το OGC-standards όπως Web Map Service ή Web Feature Service.

4.5 Bing Maps

Βασισμένο σε προγενέστερες υλοποιήσεις χαρτών, όπως το MapPoint και TerraServer, αρχικά είχε την δυνατότητα για ομαδοποίηση μόνο μιας συλλογής σημείων ενδιαφέροντος. Η Microsoft στα τέλη του 2006 πρόσθεσε την δυνατότητα προβολής των χαρτών σε 3D με την χρήση ενός .NET Direct3D API.

Τα βασικά χαρακτηριστικά της υπηρεσίας είναι ανταγωνιστικά ως προς την υλοποίηση της Google. Παρέχεται, μεταξύ άλλων, η λειτουργία Street Maps για προβολή τοπογραφικών πληροφοριών, ονόματα δρόμων, περιοχών καθώς και επισήμανση υποδομών κοινής ωφέλειας, το Road View είναι η όψη με στοιχεία πλοήγησης, το Aerial View αποτελεί ένα σύνολο δορυφορικών φωτογραφιών εμπλουτισμένες με πληροφορίες χάρτη, το Bird Eye κάνει χρήση αεροφωτογραφιών από αεροπλάνα σε χαμηλή πτήση και με κλίση 45 μοιρών για την απεικόνιση των πλαϊνών πλευρών των κτιρίων, το 3D Maps για ψηφιακή αναπαράσταση κτιρίων με τεχνικές φωτορεαλισμού. Επίσης υπάρχει η δυνατότητα λήψης οδηγιών ανάλογα με το μέσο μετακίνησης μας όπως το αυτοκίνητο, τα πόδια ή τα μέσα μαζική μεταφοράς.

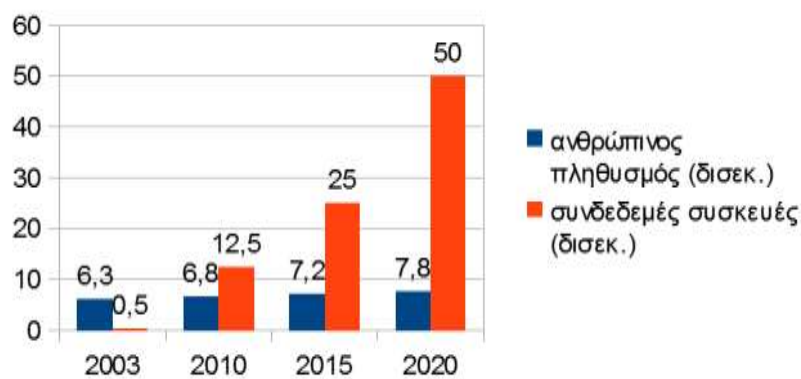
Τρία είναι τα κύρια APIs διαθέσιμα για την ανάπτυξη λογισμικού 3^{ov} κατασκευαστών. Η έκδοση Silverlight, απαιτεί το ομώνυμο plug-in, προσφέρει πλούσια και δυναμικά χαρακτηριστικά για μια ομαλή εμπειρία. Η έκδοση AJAX χωρίς την απαίτηση του Silverlight αλλά με τεχνικές αντικειμενοστραφούς προγραμματισμού JavaScript με διευρυμένη συμβατότητα και υποστήριξη περισσότερων λειτουργικών συστημάτων και περιηγητών διαδικτύου ενώ το 2010 συνδύασαν σε μια διεπαφή τα AJAX και Silverlight για μια υβριδική λύση με στόχο την επέκτασή των δυνατοτήτων τους με υποστήριξη Map Apps, Roadview, Bird Eye και Street Side View χωρίς όμως την εξάλειψη του Silverlight plug-in.

5. Ανάλυση Αγοράς

Για την διερεύνηση της βιωσιμότητας μιας επένδυσης χρόνου και χρήματος σε νέες τεχνολογίες απαιτείται μια στοιχειώδης μελέτη της δυναμικής μιας αγοράς, την εγκατεστημένη βάση πελατών ή την εν δυνάμει πλατφόρμα ανάπτυξης καθώς και πρόβλεψη τάσεων σε μια ρεαλιστική βάση με μελέτη της τρέχουσας ζήτησης.

5.1 Μεγέθη Internet of Things

Μέχρι το 2020, το κόστος εξαρτημάτων θα μειωθεί σε σημείο που η συνδεσιμότητα θα αποτελεί βασικό χαρακτηριστικό, ακόμα και οι μικροεπεξεργαστές θα κοστίζουν λιγότερο από 1\$. Αυτό ανοίγει νέους ορίζοντες στην πιθανότητα σύνδεσης των πάντων, από το πιο απλό έως το πιο περίπλοκο, για την παροχή απομακρυσμένου ελέγχου, παρακολούθησης και ανίχνευσης.



Γράφημα 1: ποσοτική αντιπαράθεση ανθρώπινου πληθυσμού και συνδεδεμένων συσκευών

Παράλληλα με την αύξηση στις συνδεδεμένες συσκευές, για την ολοκλήρωση των εφαρμογών IoT απαιτείται η ανάπτυξη του κλάδου ενσωματωμένου λογισμικού, υπηρεσιών τηλεπικοινωνίας και πληροφορικής σχετιζόμενων με τα «αντικείμενα».

Επιπρόσθετα, οι επιχειρήσεις θα κάνουν εκτενή χρήση των εφαρμογών IoT δημιουργώντας την ανάγκη ανάπτυξης ευρείας γκάμας προϊόντων για διάφορες αγορές, όπως εξειδικευμένες ιατρικές συσκευές, αισθητήρες αυτοματισμών εργοστασίων και εφαρμογές σε βιομηχανική ρομποτική, αναλώσιμους αισθητήρες στην αγροτική παραγωγή, αισθητήρες αυτοκινήτων, παρακολούθησης της ακεραιότητας των υποδομών για διάφορους τομείς, όπως οι οδικές και σιδηροδρομικές μεταφορές, στη διανομή νερού και ηλεκτρισμού.

Η οικονομική προστιθέμενη αξία που επιδιώκουν οι επιχειρήσεις από την χρήση IoT, διακλαδικά, αναμένεται να αγγίξει τα 1,9 τρισεκατομμύρια δολάρια έως το 2020.

Λόγω του χαμηλού κόστους ενσωμάτωσης λειτουργικότητας IoT σε καταναλωτικά αγαθά παράλληλα αναμένεται να δημιουργηθεί το φαινόμενο των συσκευών «φαντασμάτων» που ενώ έχουν την δυνατότητα για χρήση σε περιβάλλον IoT, εντούτοις δεν αξιοποιούνται ως τέτοιες ελλείπει κατάλληλο λογισμικό ή και ακόμα ενδιαφέροντος από τους χρήστες.

5.2 Συσκευές Κινητής Τηλεφωνίας



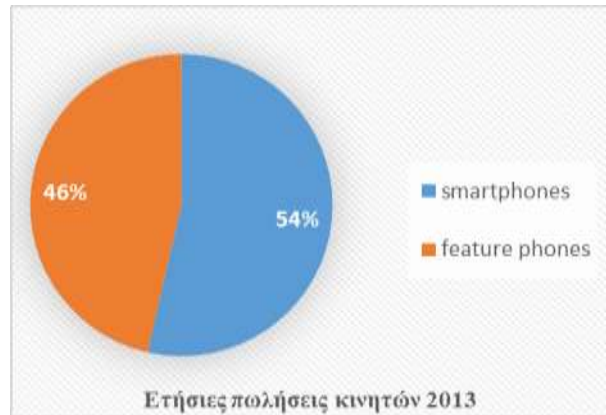
Πίνα 1: παγκόσμια αγορά κινητών 2010



Πίνα 2: αγορά smartphones 2010

Το 2013 ήταν η πρώτη χρονιά που τα smartphones ξεπέρασαν τα παραδοσιακά κινητά τηλέφωνα, με 53,6% συνολικές πωλήσεις ετησίως. Σε απόλυτους αριθμούς 968 εκατομμύρια συσκευές smartphones πωλήθηκαν παγκοσμίως αυξημένες σε σχέση με 2012 κατά 42,3%. Στην αύξηση αυτή συνέβαλαν οι χώρες της Λατινικής Αμερικής, Μέση Ανατολή, Αφρική, Ειρηνικός και στην Ανατολική Ευρώπη και γενικά στις αναδυόμενες αγορές, που συγκέντρωσαν περισσότερο από το 50% των συνολικών πωλήσεων. Εντυπωσιακή αύξηση των πωλήσεων σε σχέση με το προηγούμενο έτος παρουσία η Ινδία κατά 166,8%, λόγω των χαμηλού κόστους Android συσκευών, ακολουθεί η Λατινική Αμερική με 96,1%, ενώ αντίστοιχα η κινέζικη αγορά αναπτύχθηκε κατά 86,3%.

Χαρακτηριστικά μόνο οι συσκευές που ενσωματώνουν λειτουργικό Android ξεπέρασαν σε πωλήσεις το 1 δισεκατομμύριο συσκευές το 2014. Τα παραδοσιακά κινητά παρουσιάζουν δραματική μείωση 25%, λόγω της μειωμένης διαφοράς κόστους με τα οικονομικά Android-phones, ενώ σύμφωνα με προβλέψεις του Gartner έως το 2018 εννιά στα δέκα τηλέφωνα θα είναι smartphone.

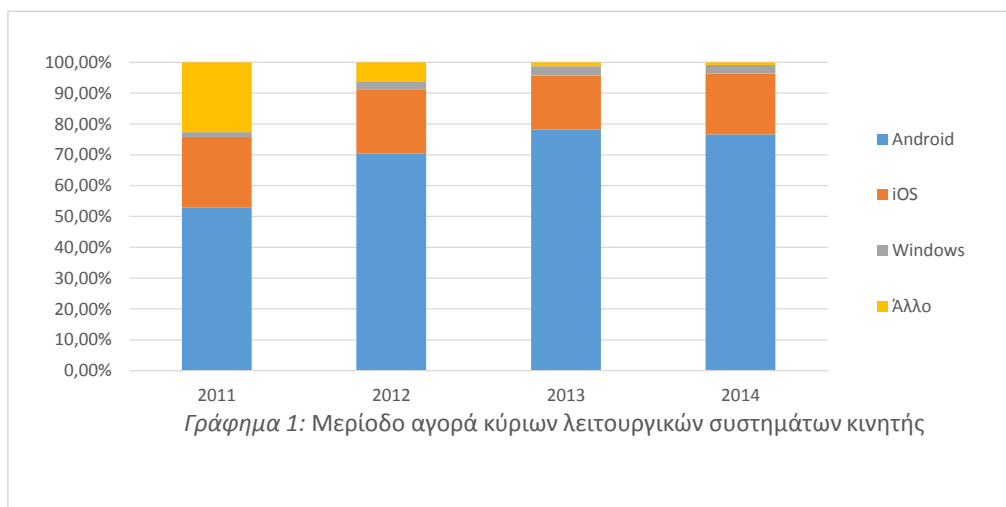


5.3 Λειτουργικά Συστήματα Κινητής

Ένα σύγχρονο λειτουργικό σύστημα για κινητά είναι ικανό να χειρίζεται τα υποσυστήματα των wearables, smartphones, tablets και TVs συνδυάζοντας χαρακτηριστικά προσωπικού ηλεκτρονικού υπολογιστή, κινητού τηλεφώνου, υποσυστημάτων συνδεσιμότητας όπως Bluetooth και Wi-Fi, πλοήγηση GPS, φωτογραφική μηχανή και video-camera, συναλλαγές Near Field Communication (NFC) μέσω ενός εύχρηστου περιβάλλοντος χρήστη σε φυσική γλώσσα όπως αφή, λεκτικά και χειρονομίες.

Αποτελεί τον ελάχιστο κοινό παρονομαστή για την ανάπτυξη εφαρμογών και υπηρεσιών με εύκολες διαδικασίες, ταχύτητα και χαμηλό κόστος καθώς αυτό ελέγχει τα επιμέρους υποσυστήματα χωρίς ο δημιουργός να χρειάζεται να επικεντρώνεται σε τεχνικά ζητήματα χειραγώγησης του υλικού

(hardware) της συσκευής παρά μόνο στον τρόπο υλοποίησης της ιδέας του.



Επίσης γεφυρώνει την υποστήριξη του υλικού που προέρχεται από διαφορετικούς κατασκευαστές, χωρίς να πρέπει μια εφαρμογή να έχει γνώση ή και εξάρτηση από συγκεκριμένων προδιαγραφών υλικό.

Στο εμπόριο δραστηριοποιούνται κατά κύριο λόγο τρεις εταιρίες η Google με το Android, η Apple με το iOS και η Microsoft με τα Windows με διαφορετικά επιχειρησιακά μοντέλα.

Η Apple επιλέγει να κατασκευάζει η ίδια τις συσκευές της, με δυο μοντέλα smartphone και δυο tablet με μόνη διαφοροποίηση το μέγεθος της οθόνης τους, επιπλέον το λειτουργικό της δεν το εξουσιοδοτεί σε καμία άλλη εταιρία για να κρατήσει την ποιότητα κατασκευής τους σε υψηλά επίπεδα. Επίσης ανανεώνει τα μοντέλα της ανα έτος.

Η Google, από την άλλη, παρέχει δωρεάν το Android σε όποιον κατασκευαστεί το επιθυμεί και αντλεί έσοδα από την μηχανή αναζήτησης που είναι προεγκατεστημένη, από τις υπηρεσίες της όπως email, maps, news, app store αλλά και από τις διαφημίσεις που εμφανίζονται εντός των εφαρμογών. Επίσης το λογισμικό της είναι ανοιχτού κώδικα. Αυτή η ελευθερία, έχει προκαλέσει και την ραγδαία άνοδο της, με πωλήσεις που είναι γύρω στις 1,5 δισεκατομμύρια συσκευές το χρόνο, αποσπώντας την μερίδα του λέοντος με το μερίδιο αγοράς να αγγίζει το 80% το 2015. Αξιοσημείωτη είναι η ποικιλία συσκευών που ενσωματώνουν Android, από υψηλούς κόστους και απόδοσης μέχρι συσκευές κάτω των 50\$ που απευθύνονται σε αναπτυσσόμενες χώρες, όπου η διείσδυσή του σε αυτές το οδηγούν στην κορυφή.

Η Microsoft, μπήκε καθυστερημένα και με αμφιλεγόμενη στρατηγική στην αγορά νέας γενιάς smartphones με τα Windows Phone 8, καθώς οι προηγούμενες εκδόσεις της ισοπεδώνονταν από τον ανταγωνισμό. Αρχικά διέθετε το λειτουργικό της σε χαμηλό κόστος προς όλους, πράγμα το οποίο απέτρεψε τους κατασκευαστές από την υιοθέτησή του, σε σχέση με το πιο ανεπτυγμένο και δωρεάν Android, και παράλληλα κατασκεύαζε τα δικά της μοντέλα μετά την εξαγορά του παραρτήματος mobile της Nokia. Το μερίδιο της είναι πολύ χαμηλό, γύρω στο 3%, αλλά με γκάμα συσκευών και θετικές προοπτικές για ανάπτυξη στο μέλλον, με την κυκλοφορία των Windows 10 όπου το βασικό χαρακτηριστικό τους θα είναι ότι οι εφαρμογές τους θα τρέχει ταυτόχρονα σε wearables, smartphones, tablets, xbox one και PC's διευρύνοντας το αγοραστικό κοινό τους χωρίς επιπλέον προσπάθεια από τις εταιρίες ανάπτυξης εφαρμογών.

5.4 Αγορά Τηλεπικοινωνιών Κινητής

Σύμφωνα με εκτιμήσεις της International Telecommunication Union, υπάρχουν περίπου 7

δισεκατομμύρια συνδέσεις κινητής τηλεφωνίας παγκοσμίως, διείσδυση στο 95,5% δηλαδή του παγκόσμιου πληθυσμού.

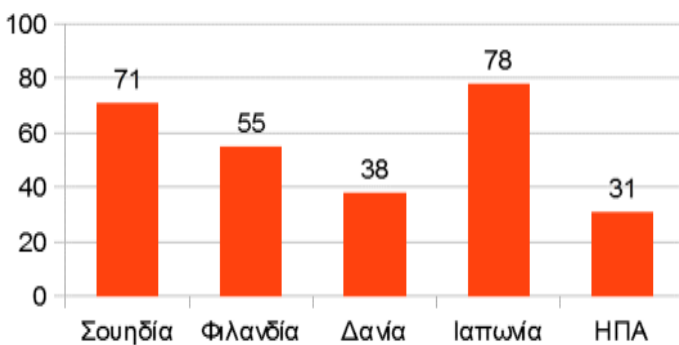
Την τελευταία τριετία αυξήθηκε ο αριθμός των συνδέσεων κατά ένα δισεκατομμύριο, αλλά ο ρυθμός ανάπτυξης τους επιβραδύνεται. Στον αναπτυσσόμενο κόσμο έχουν φτάσει σε σημείο κορεσμού, με 120,8% πληθυσμιακή διείσδυση, ενώ ήδη υπάρχουν για κάθε άτομο περισσότερες από μια συνδέσεις, αφήνοντας ελάχιστο χώρο περαιτέρω μεγέθυνσης.

Η ανάπτυξη της αγοράς, οδηγείται από την ζήτηση του αναπτυσσόμενου κόσμου και κυρίως από την ραγδαία υιοθέτηση smartphones στην Κίνα και την Ινδία, έθνη με τον μεγαλύτερο πληθυσμό. Το 78% των συνδέσεων παγκοσμίως βρίσκονται σε αναπτυσσόμενες περιοχές, ενώ η διείσδυση των συνδέσεων στον πληθυσμό αγγίζει το 90,2% με ανοιχτή την προοπτική περαιτέρω αύξησης ειδικότερα στην Αφρική όπου το αντίστοιχο ποσοστό βρίσκεται στο 69,3%.

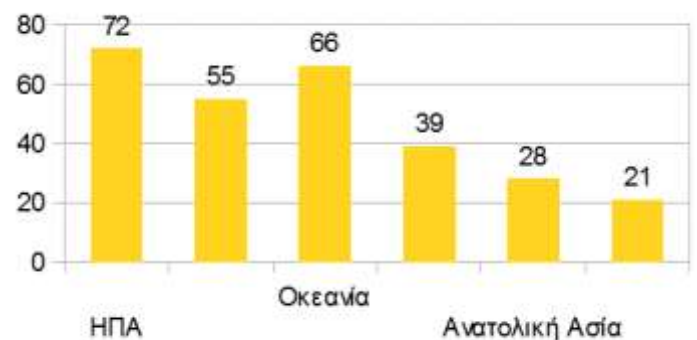
Σύμφωνα με τις προβλέψεις του Mobile Factbook 2013, στα τέλη του 2016 ο αριθμός των συνδέσεων κινητής θα ανέρχεται στα 8.5 δισεκατομμύρια.

Ο αριθμός όμως των ενεργών συνδέσεων δεν αποτυπώνει και τον αριθμό των μοναδικών χρηστών κινητής τηλεφωνίας που είναι κατά πολύ μικρότερος. Σύμφωνα με μελέτες της Ericsson, ο αριθμός των χρηστών ανέρχεται στα 4,5 δισεκατομμύρια, μιας και πολλοί συνδρομητές έχουν περισσότερες από μια συνδέσεις, είτε για οικιακή χρήση, για την εργασία ή σε κάποιο tablet ή PC. Επίσης σε αναπτυσσόμενες περιοχές υπάρχει το φαινόμενο διαμοιρασμού μιας σύνδεσης σε περισσότερα πρόσωπα.

Έτσι η Ericsson, προβλέπει ότι οι συνδρομές κινητής παγκοσμίως θα ανέλθουν στα 9,3 δισεκατομμύρια μέχρι το 2019, από τους οποίους τα 5,6 θα αφορούν χρήστες που διαθέτουν smartphone. Αντίστοιχα οι συνδέσεις κινητής ευρυζωνικής πρόσβασης ανήλθαν στα 2 δισεκατομμύρια το 2013, ενώ μέχρι το 2019 θα ξεπεράσουν τα 8 δις, από αυτές τα 2,6 δις θα αφορούν υπηρεσίες LTE (4G) και τα 4,8 δις θα είναι WCDMA/HSPA (3G).



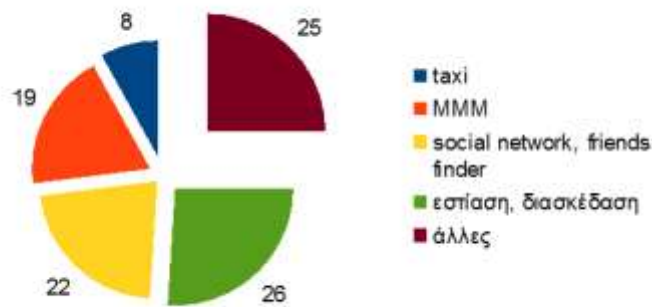
Γράφημα 2: αγορά 3G phones ανά χώρα



Γράφημα 3: mobile broadband ανά συνολικές συνδέσεις

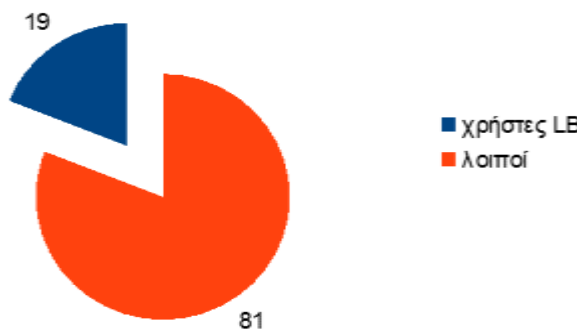
5.5 Location Based Service

Αν και η έννοια των Υπηρεσιών Βασιζόμενη στη χρήση Τοποθεσίας (LBS) προέκυψε πρόσφατα, ακολουθεί αξιοσημείωτη ανάπτυξη σε σχέση με τους υπόλοιπους κλάδους της πληροφορικής, λόγω της ζήτησης για διάφορες υπηρεσίες εξαρτώμενες από την τοποθεσία τους.

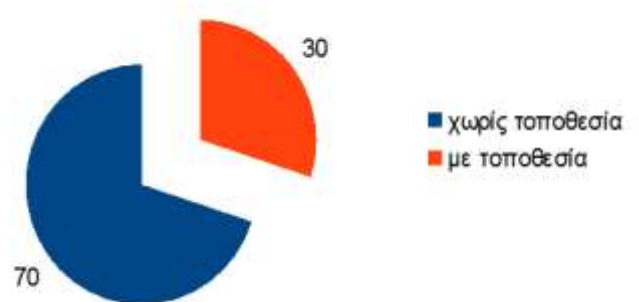


Πίνα 4: χρήσεις LBS

Αυτή η ανάπτυξη στην συγκεκριμένη αγορά, επηρεάζεται από έναν αριθμό στοιχείων που από μόνα τους παρουσιάζουν εκρηκτική ανάπτυξη και διείσδυση, όπως οι συσκευές που ενσωματώνουν σύνδεση στο διαδίκτυο, τερματικά θεσηγνωσίας, wearables, smartphones, tables, κοινωνικά δίκτυα. Επίσης παράγοντες που οδηγούν στην μεγέθυνση του κλάδου είναι το ενδιαφέρον των χρηστών για προσωποποιημένες υπηρεσίες πληροφοριών τοποθεσίας και ιστορίας, τα εξελισσόμενα επιχειρηματικά μοντέλα όπως οι διαφημίσεις κινητής, τα οφέλη κόστους από την εφαρμογή λύσεων εντοπισμού οχημάτων και η διαχείριση στόλου.



Πίνα 5: ενήλικες κάτοχοι smartphone(%)



Πίνα 6: posts ενηλίκων χρηστών σε social media

Η αγορά στα τέλη του 2014 είχε μέγεθος 12,2 δισεκατομμυρίων δολαρίων, ενώ υπολογίζεται ότι μέχρι τα τέλη του 2019 θα παράγει έσοδα 43 δισεκατομμυρίων. Σε αυτήν την αύξηση κύρια συμβολή θα έχουν οι εφαρμογές κοινωνικής δικτύωση με ενσωματωμένη διαφήμιση, όπως το Facebook, ενώ ακολουθούν οι εφαρμογές τοπικής αναζήτησης, όπως αυτή της Google.

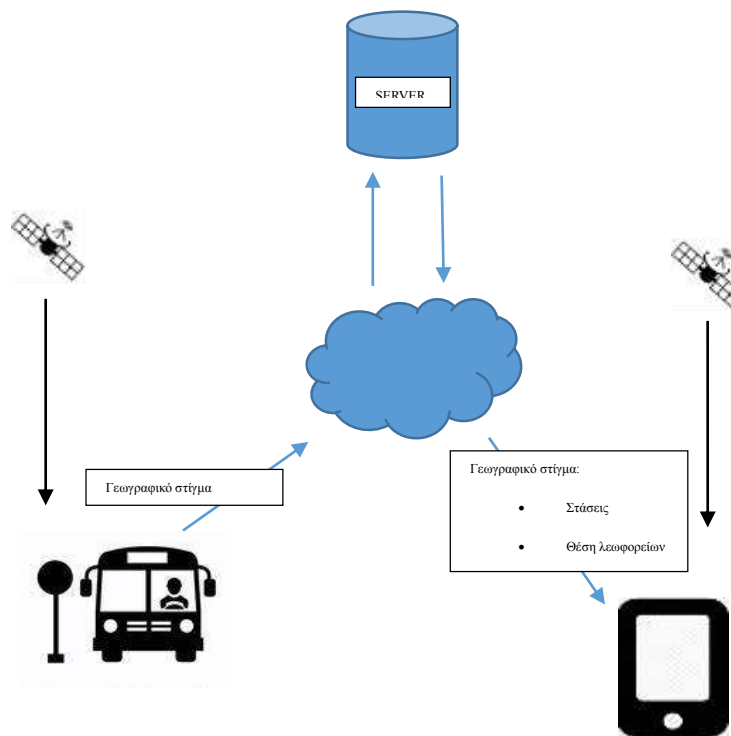
Άλλες κύριες υπηρεσίες όπως διασκέδασης και εστίασης, για την εύρεση χώρων ανάλογα με το που βρισκόμαστε μια δεδομένη χρονική στιγμή, η αναζήτηση φίλων σε κοντινή απόσταση, ο συνδυασμός θέσης και κατάλληλων μέσω μαζικής μεταφοράς για την καθοδήγηση στον προορισμό, η αυτόματη κλήση ταξί αλλά και υπηρεσίες πρόγνωσης του καιρού τοπικά, ενημέρωσης αρχών ασφαλείας για επείγοντα περιστατικά, εφαρμογές επαυξημένης πραγματικότητας.

6. Τα MMM Ηρακλείου σε πραγματικό χρόνο

Στον σύγχρονο κόσμο οι απαιτήσεις της καθημερινότητας και της εργασιακής κατάστασης του ατόμου δυσχεραίνουν την ποιότητα ζωής του. Βασική προϋπόθεση για την ανάκτηση της ποιότητας αυτής είναι η ελαχιστοποίηση του απαιτούμενου χρόνου για την εκτέλεση απλών διεργασιών. Μια από αυτές είναι οι μετακινήσεις, που στο αστικό περιβάλλον, λόγω αποστάσεων και κυκλοφοριακής συμφόρησης αποτελούν μεγάλη πρόκληση.

Τα μέσα μαζικής μεταφοράς μπορούν να δώσουν λύση στο πρόβλημα αυτό προσφέροντας ταχύτητα, ασφάλεια και οικονομία. Εμπόδιο, σε πολλές περιοχές, στην μαζική υιοθέτηση από το ευρύ κοινό αποτελεί η χρονική αβεβαιότητα έλευσης των οχημάτων στο σημείο επιβίβασης, η πληροφόρηση για τον γενικό προγραμματισμό των δρομολογίων καθώς και η ασαφής συχνότητα διέλευσης τους. Μονόδρομος για την προσέλκυση νέων επιβατών είναι η άρση τέτοιου είδους κολημάτων ικανοποιώντας τις ανάγκες τους αποτελεσματικά.

Η ύπαρξη μιας ροής πληροφοριών που θα ενημερώνει τους ενδιαφερόμενους σε πραγματικό χρόνο για τα διαθέσιμα οχήματα σε μια σειρά δρομολογίων, για τον εκτιμώμενο χρόνο και συχνότητα διέλευσης από τα σημεία επιβίβασης αλλά και προβλέψεις για συγκεκριμένες μέρες και χρονικές ζώνες μπορεί να συμβάλει στον προγραμματισμό της καθημερινότητάς τους.



Εικόνα 6-1 Ροή πληροφορίας στο σύστημα

Παράλληλα η δημιουργία όλων αυτών των δεδομένων οδηγεί στην αξιοποίηση, με τεχνικές δεδομένων μεγάλης κλίμακας για την εξαγωγή συμπερασμάτων όπως ποια είναι τα δρομολόγια με την μεγαλύτερη κίνηση πελατών, εάν υπάρχουν προβληματικές οδοί διέλευσης των οχημάτων λόγω κυκλοφοριακής συμφόρησης, η χρονική διάσταση των παραπάνω, εάν οι υπάρχουσες υποδομές μπορούν να εξυπηρετήσουν την ζήτηση ή εάν θα μπορούσε να αλλάξει ο προγραμματισμός των οχημάτων χρονικά και τοπικά. Επίσης σημαντική είναι η δυνατότητα συνδυασμού των συμπερασμάτων με διάφορα χρονικά στιγμότυπα ώστε να εξάγονται στατιστικά πληροφορίες και προβλέψεις για την μελλοντική κατάσταση του δικτύου των μέσων μεταφοράς σε ανάλογο χρόνο και συνθήκες. Για την

αρχή επίβλεψης και σχεδίασης του συστήματος των μέσων μαζικής μεταφοράς διευκολύνεται η λήψη αποφάσεων για μελλοντικές επενδύσεις σε προσωπικό ή οχήματα και υλικοτεχνικά στοιχεία.

Η γνώση των αναγκών του κοινού και της κατάστασης της κυκλοφοριακής συμφόρησης του οδικού δικτύου οδηγεί στον σχεδιασμό δρομολογίων πέρα από τον παραδοσιακό στατικό σχεδιασμό που απλώς διαιρεί χρονικά την διέλευση των οχημάτων και δεν λαμβάνει υπόψιν άλλους εξωγενείς και δυναμικά μεταβαλλόμενους παράγοντες.

6.1 Ανάλυση υλοποίησης

Τα κύρια μέρη ενός συστήματος που δημιουργεί, επεξεργάζεται και διανέμει πληροφορίες θέσης σε πραγματικό χρόνο, αποτελείται από ένα σύστημα δορυφορικής πλοήγησης, όπως είναι το GPS, το Galileo ή το Glonass, μια συσκευή με δυνατότητα πλοήγησης, σύνδεση δεδομένων με το διαδίκτυο και ένα εξελιγμένο λειτουργικό σύστημα με δυνατότητα εγκατάστασης και εκτέλεσης εφαρμογών για τον οδηγό, αντίστοιχη συσκευή από την μεριά του πελάτη καθώς και έναν διακομιστή.

Η δωρεάν διάθεση υπηρεσιών δορυφορικής πλοήγησης και θεσηγνωσίας είναι στρατηγικής σημασίας παράγοντας για την μαζική κυκλοφορία υπηρεσιών με επίγνωση της θέσης, εκμεταλλεζόμενες την ανάγκη για εφαρμογές με περιεχόμενο σχετικό με το που γίνεται η θέαση του. Παράλληλα ο ανταγωνισμός μεταξύ χωρών και υπηρεσιών αυξάνει την ποιότητα και την ακρίβεια του εκπεμπόμενου δορυφορικού σήματος είτε εγγενώς από τα ίδια τα συστήματα είτε από τον συνδυασμό αυτών. Με βάση τα παραπάνω είναι κατανοητό ότι εφαρμογές όπως της παρούσας πτυχιακής δεν θα είχαν νόημα χωρίς την δωρεάν και αδιάληπτη παρουσία του δορυφορικού σήματος για εντοπισμό θέσης.

Ο κορμός ενός τέτοιου συστήματος είναι ο διακομιστής, όπου συλλέγει τα παραγόμενα δεδομένα από τα οχήματα και διανέμει πληροφορίες. Αναλυτικά η βάση δεδομένων δέχεται ως εισερχόμενα δεδομένα την τρέχουσα θέση ενός οχήματος, τον μοναδικό κωδικό αυτού και τον κωδικό δρομολογίου, εν συνεχεία συσχετίζοντας τα δεδομένα αυτά με αποθηκευμένες πληροφορίες από άλλους πίνακες της βάσης δεδομένων και στέλνει στον πελάτη πληροφορίες για το όχημα, την θέση του και το όνομα του δρομολογίου. Επίσης διανέμει πληροφορίες όπως διαδρομές δρομολογίων στον χάρτη, σταθμούς επιβίβασης ή άλλες χρήσιμες πληροφορίες. Παράλληλα σε δεύτερο επίπεδο και μετά από επεξεργασία των παραγόμενων δεδομένων με βάση την διαδρομή, την χρονική στιγμή δημιουργίας και την ταχύτητα αλλαγής από θέση σε θέση, προκύπτουν πληροφορίες για την λειτουργία του όλου συστήματος.

Τα σύγχρονα smartphone ή tablet μπορούν να διαδραματίσουν ρόλο δημιουργού δεδομένων από το όχημα ή αποδέκτη πληροφοριών, δηλαδή ο πελάτης, εφόσον εκπληρώνουν τις 3 προαναφερθείσες απαιτήσεις του δέκτη δορυφορικής πλοήγησης, της συνδεσιμότητας δεδομένων και του λειτουργικού συστήματος. Επιπλέον πληροφορίες στέλνονται από τον διακομιστή στον πελάτη για το δίκτυο σταθμών επιβίβασης των μέσων μαζικής μεταφοράς.

Σημαντικό κομμάτι επίσης αποτελεί η συνεχής και αδιάληπτη σύνδεση στο διαδίκτυο ώστε να υπάρχει επικοινωνία με τον διακομιστή και από την πλευρά του οχήματα, με την συνεχής αποστολή της τρέχουσας του θέσης αλλά και από την πλευρά του πελάτη όπου ανανεώνονται στο χάρτη οι θέσεις των προβαλλόμενων οχημάτων. Επίσης οι περισσότερες συσκευές smartphone ενσωματώνουν δέκτη A-GPS όπου είναι απαραίτητη η υποβοήθηση του δέκτη δορυφορικού σήματος με δεδομένα από επίγειους σταθμούς του συστήματος μέσω διαδικτύου.

6.2 Χρησιμοποιηθέντα εργαλεία

Android Studio 0.4.4

Ολοκληρωμένο περιβάλλον ανάπτυξης εφαρμογών για λειτουργικό Android, με ενσωματωμένη λειτουργία IntelliJ IDEA για αυτόματη συμπλήρωση κώδικα χωρίς λάθη, ευκολίες ειδικά για κώδικα που προορίζεται για το εν λόγω λειτουργικό, Android Virtual Device (AVD) για εξομίωση συσκευής Android και δομική του κώδικα σε πραγματικές συνθήκες. Επίσης υποστηρίζει ανάπτυξη των εφαρμογών με γλώσσα Java αλλά και σε Android Native (NDK) που αποτελεί lower level γλώσσα με συνδυασμό Java και C/C++ για περιπτώσεις όπου η απόδοση είναι βασικό ζητούμενο για την εφαρμογή.

Ενσωματώνει επίσης το Android Debug Bridge (ADB) το οποίο είναι ένα ευέλικτο εργαλείο γραμμής εντολών που επιτρέπει την επικοινωνία με τον εξομοιωτή ή την συνδεδεμένη συσκευή Android. Πρόκειται για ένα πρόγραμμα πελάτη-διακομιστή που περιλαμβάνει τρεις συνιστώσες, έναν πελάτη ο οποίος τρέχει στο μηχάνημα ανάπτυξης λογισμικού και μπορεί να επικαλεστεί κάποιον πελάτη από ένα κέλυφος με την εκτέλεση μιας ADB εντολή, έναν διακομιστή ο οποίος λειτουργεί ως παρασκηνακή διαδικασία στο μηχάνημα ανάπτυξης λογισμικού και διαχειρίζεται την επικοινωνία μεταξύ του πελάτη και ενός ADB δαίμονα που τρέχει στον εξομοιωτή ή την συνδεδεμένη συσκευή Android και τέλος έναν δαίμονα που εκτελείτε ως παρασκηνακή διαδικασία για κάθε εξομοίωση ή συσκευή Android.

Το Gradle είναι ένα προηγμένο αυτοματοποιημένο εργαλείο κατασκευής (build) που διαχειρίζεται τις εξαρτήσεις της εκάστοτε εφαρμογής ενώ χρησιμοποιεί τεχνικές directed acyclic graph (DAG) για τον προσδιορισμό της σειράς εκτέλεσης των διεργασιών. Επίσης είναι ικανό να προσθέτει στο project βιβλιοθήκες τοπικά από το μηχάνημα ανάπτυξης λογισμικού, από απομακρυσμένο H/Y και να ελέγχει για νέες εκδόσεις αυτών με μόνο λίγες γραμμές απλού κώδικα.

Το Android SDK χωρίζει τα εργαλεία, τις πλατφόρμες, και άλλα συστατικά σε πακέτα που μπορείτε να κατεβάσετε χρησιμοποιώντας το SDK Manager. Για παράδειγμα, όταν τα εργαλεία SDK ενημερωθούν ή μια νέα έκδοση της πλατφόρμας Android κυκλοφορεί, χρησιμοποιούμε το εν λόγω εργαλείο για να τα κατεβάσετε γρήγορα στο περιβάλλον ανάπτυξης.

Το Android Virtual Device Manager (AVD Manager) παρέχει μια γραφική διεπαφή χρήστη με την οποία μπορείτε να δημιουργήσετε και να διαχειριστείτε τα Android Virtual Devices (AVDs), τα οποία απαιτούνται από το Android Emulator.

Το Logcat αποτελεί το σύστημα καταγραφής Android που παρέχει ένα μηχανισμό για τη συλλογή και την προβολή των σφαλμάτων της εξόδου συστήματος. Οι καταγραφές από διάφορες εφαρμογές και τα τμήματα του συστήματος, συλλέγονται σε μια σειρά κυκλικών ρυθμιστικών μηνυμάτων, τα οποία στη συνέχεια μπορούν να προβληθούν και να φιλτράρονται από μια εντολή logcat. Μπορείτε να χρησιμοποιήσετε logcat από ένα κέλυφος ADB ώστε να δείτε τα μηνύματα καταγραφής για διαγνωστικούς λόγους.

Android SDK

Το Android Software Development Kit συνιστάται από εξομοιωτές συσκευών, δείγμα κώδικα και εξειδικευμένες βιβλιοθήκες για την αξιοποίηση των υποσυστημάτων μιας συσκευής Android όπως GPS, Camera, Bluetooth, συνδεσιμότητα δικτύου αλλά και άλλων χαρακτηριστικών του εν λόγω λειτουργικού. Υποστηρίζει πληθώρα από προγραμματιστικά περιβάλλοντα όπως Eclipse με plugin ADT, Netbeans με plugin, IntelliJ IDEA αλλά και του επίσημου λογισμικού ανάπτυξης από την Google το Android Studio.

Google Play Services

Είναι μια ιδιοταγής υπηρεσία που εκτελείται στο παρασκήνιο αλλά και στο πακέτο API για Android συσκευές, η οποία παρέχει πρόσβαση για τις εφαρμογές στις βασικές υπηρεσίες της Google όπως Play Game, Location API, Google+, Drive, Ads, Wallet αλλά και στα Maps.

Για την ανάπτυξη εφαρμογών που κάνουν χρήση τέτοιων υπηρεσιών απαιτείται η προσθήκη της ανάλογης βιβλιοθήκης Google Play Services στο project μας, η οποία παρέχει έτοιμες κλάσεις για την παραγωγή αντικειμένων που αξιοποιούν και χειραγωγούν τα επιμέρους χαρακτηριστικά κάθε υπηρεσίας δημιουργώντας εξατομικευμένες εκδοχές τους για την εκάστοτε υλοποίηση. Επίσης απαιτείται το μοναδικό API key για κάθε υπηρεσία.

JDK 7.0

Java Development Kit αποτελούν το σύνολο των βασικών βιβλιοθηκών και Java Virtual Machine (JVM) για την ολοκληρωμένη ανάπτυξη εφαρμογών, επίσης αποτελεί την πιο ευρέως χρησιμοποιημένη εργαλειοθήκη λογισμικού. Όλες οι δυνατότητές του αξιοποιούνται από το Android Studio για την δημιουργία της ελάχιστης λειτουργικότητας εφαρμογών σε συνδυασμό με το Android SDK.

JRE 7.0

Java Runtime Environment περιέχει τις βασικές βιβλιοθήκες Java, ένα Java Virtual Machine, browser plugin, ένα εργαλείο για ρυθμίσεις και είναι απαραίτητο για την λειτουργία των εφαρμογών γραμμένων με Java ανεξάρτητα από το λειτουργικό σύστημα. Επίσης απαιτείται για την βασική λειτουργία του γραφικού περιβάλλοντος του Android Studio.

XAMPP

Είναι μια συλλογή από ανοιχτού κώδικα εργαλεία όπως Apache HTTP server, MySQL, FileZilla FTP server, Tomcat, Mercury Mail Server και με υποστήριξη σε γλώσσες PHP και Perl. Παρόλο που δημιουργήθηκε ως μια γρήγορη λύση ανάπτυξης website και webservices για προγραμματιστές πλέον χρησιμοποιείται από πολλούς και ως webserver.

Ο Apache HTTP server, χρησιμοποιείται για την φιλοξενία της ιστοσελίδας για την θέαση της ζωντανής ροής των οχημάτων αλλά και για την υλοποίηση των απλών webservices σε PHP για την διακίνηση δεδομένων μέσω AJAX αλλά και στην πλευρά του Android client με παρόμοιο τρόπο, δηλαδή πακέτα JSON.

Η MySQL ως βάση δεδομένων, για μόνιμη αποθήκευση στάσεων και δρομολογίων αλλά και ως ενδιάμεσος κόμβος από την εφαρμογή του οδηγού στο website και το Android client. Συμπληρωματικό ρόλο παίζει και το εργαλείο PhpMyAdmin για την οπτική διαχείριση των αποθηκευμένων πληροφοριών, της δομής της βάσης δεδομένων, την εξαγωγή εφεδρικών στιγμιότυπων της βάσης αλλά και ως βοηθητικό διαγνωστικό.

Notepad ++

Είναι το notepad των προγραμματιστών αφού προσφέρει ευανάγνωστη αναπαράσταση του κώδικα οποιασδήποτε γλώσσας με δυνατότητα επεξεργασίας και αποθήκευσης σε αρχείο ανάλογα με την γλώσσα χωρίς όμως την δυνατότητα εκτέλεσής τους, πολύ χρηστικό στην περίπτωση των Script (server-side) γλωσσών.

VirtualBox

Λογισμικό διαχείρισης και δημιουργίας guest virtual machines, υποστηρίζοντας μεγάλη ποικιλία λειτουργικών συστημάτων από Windows, Linux, Unix, OS X και Solaris.

Χρησιμοποιήθηκε για την δημιουργία ενός webserver σε λειτουργικό windows με την βοήθεια του XAMPP για την εγκατάσταση της βάσης δεδομένων και του http server ώστε να δοκιμάσουμε στην πράξη τους μηχανισμούς της ιστοσελίδας αλλά και την συμπεριφορά της βάσης δεδομένων στα διάφορα αιτήματα από τους android-client και android application του οδηγού.

Επίσης είναι απαραίτητο για την λειτουργία του λογισμικού εξομοίωσης Genymotion για λειτουργικό Android.

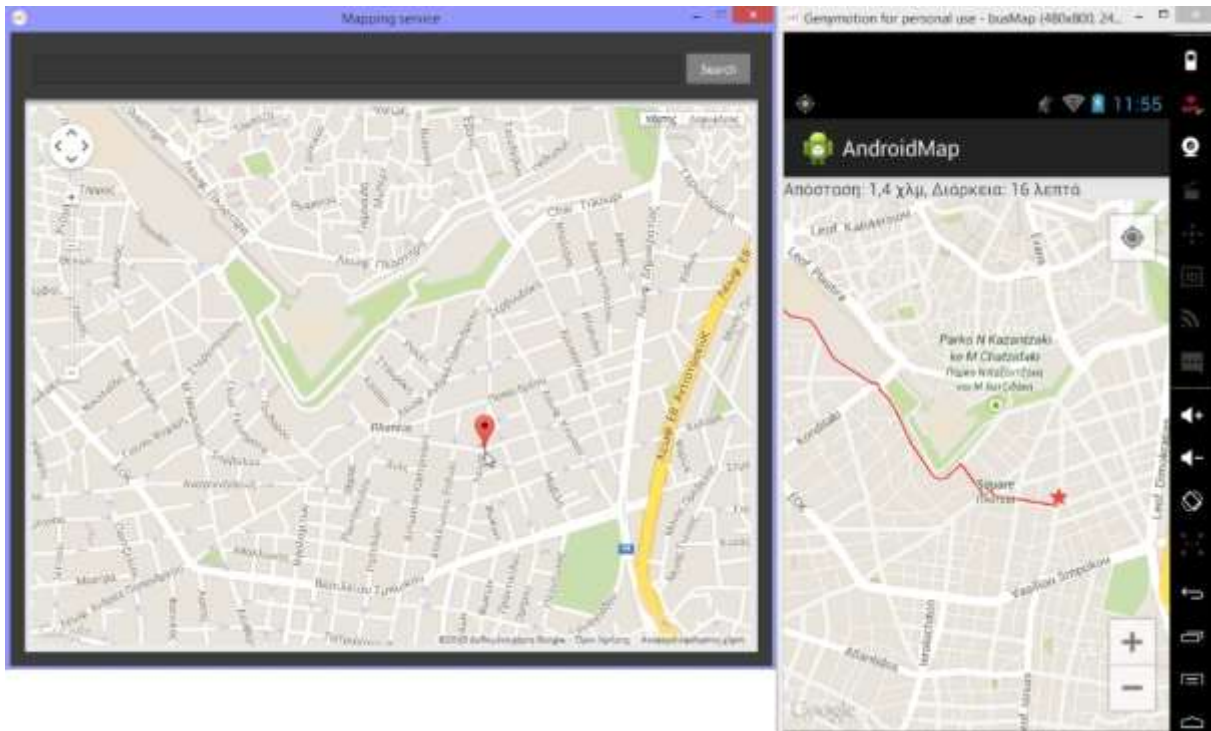
Genymotion

Εξειδικευμένο λογισμικό virtual machine για υποστήριξη λειτουργικού Android μέσω χρήσης VirtualBox αλλά με πολύ βελτιωμένες επιδόσεις και χρήσιμα εργαλεία όπως εξομοίωση πλήκτρων, κάμερας, wifi και GPS.

Συnergieζεται με plugin με το Android Studio αλλά δεν ενσωματώνει στις προρυθμισμένες συσκευές του τα Google Play Services, πράγμα το οποίο αποτελεί εμπόδιο για τον έλεγχο των εφαρμογών που απαιτούν αυτές τις βιβλιοθήκες όπως και στην παρούσα υλοποίηση όπου γίνεται χρήση των Google Maps. Μπορούμε όμως να εγκαταστήσουμε στις εικονικές συσκευές χειροκίνητα όποια βιβλιοθήκη θέλουμε μέσα από την γραμμή εντολών του Android Debug Bridge (adb) στο Android Studio.

Το μεγάλο πλεονέκτημα σε σχέση με τον ενσωματωμένο εξομοιωτή συσκευών Android είναι η ταχύτητα σε επίπεδο απόκρισης σχεδόν πραγματικής συσκευής που επιτρέπει την δοκιμή των αναπτυσσόμενων εφαρμογών σε πραγματικές συνθήκες αλλά και των εργαλείων που διαθέτει όπως αυτό της τρέχουσας θέσης όπου η μεταβολή αυτής γίνεται ευκολά και γρήγορα μέσω λογισμικού

ελέγχοντας την λειτουργία ή μη χαρακτηριστικών της υλοποίησης.



Εικόνα 6-2 αριστερά εργαλείο χαρτών εξομοιωτή, δεξιά παράθυρο της

Επίσης μπορεί να εκτελεστεί έλεγχος σεναρίων όπως κατάσταση σύνδεσης δεδομένων, εκτέλεση εφαρμογής LBS με κλειστό το GPS, εναλλαγή της διάταξης οθόνης από το εικονικό γυροσκόπιο και κατά πόσο η κατάσταση λειτουργίας αυτών επηρεάζουν την λειτουργία μιας εφαρμογής ή και τα παραγόμενα σφάλματα εξαίρεσης.

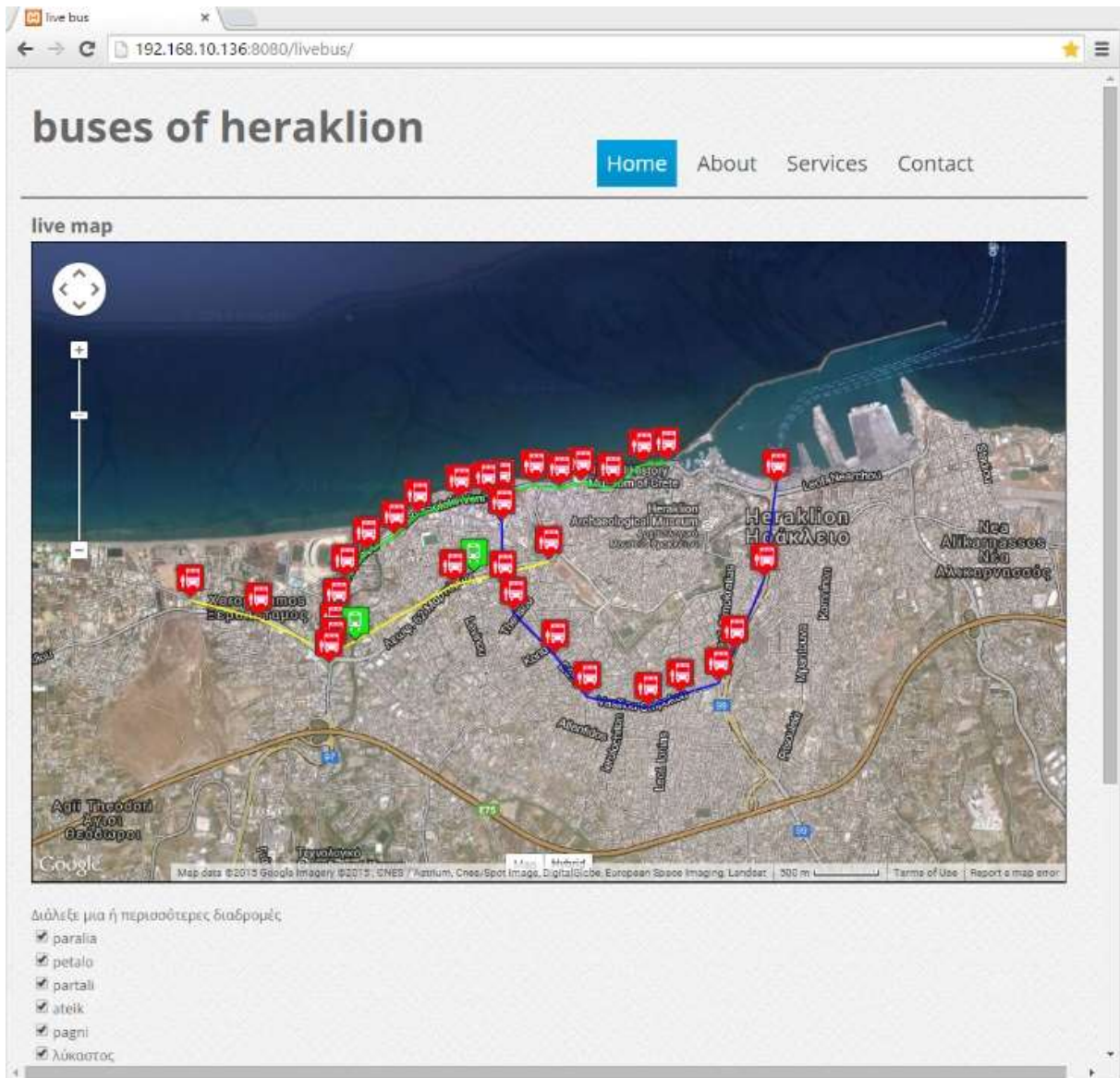
6.3 Η πλευρά του διακομιστή

Η ανάπτυξη ενός διακομιστή, που επωμίζεται το βάρος όλης της λειτουργίας ενός συστήματος δημοσίων μεταφορών με επίγνωση θέσης, απαιτεί τον συνδυασμό πολλών τεχνολογιών όπως εξειδικευμένο λογισμικό διακομιστή ιστοτόπων με υποστήριξη PHP, λογισμικό διακομιστή βάσης δεδομένων MySQL, υλικό που να μπορεί να διαχειρίζεται και να φέρνει εις πέρας αιτήματα από χιλιάδες χρήστες ταυτόχρονα αλλά και την επεξεργασία των εισερχόμενων δεδομένων με τρόπο ώστε να παράγονται πληροφορίες χρήσιμες για την καθημερινότητα του χρήστη.

Το κομμάτι του διακομιστή, αποτελεί τον βασικό κορμό του συστήματος αφού εκεί βρίσκεται η βάση δεδομένων ως βασικό αποθηκευτικό μέσο των δεδομένων που διακινούνται κατά μήκος του αλλά και η ιστοσελίδα για την ζωντανή πρόσβαση στα οχήματα που βρίσκονται την τρέχουσα χρονική στιγμή στο δρόμο με δυνατότητα επιλογής δρομολογίου, αναπαράσταση της θέσης των στάσεων και των δρομολογίων.

Συγκεκριμένα, πληροφορίες όπως το γεωγραφικό στίγμα από στάσεις λεωφορείων, ονόματα και κωδικοί δρομολογίων, ονόματα στάσεων, ονόματα οδηγών λεωφορείων αλλά και το τρέχον στίγμα των οχημάτων που είναι σε υπηρεσία.

Οι παραπάνω πληροφορίες είναι απαραίτητες για την ελάχιστη οπτικοποίηση ενός συστήματος δημόσιων μεταφορών με επίγνωση θέσης. Χρησιμοποιώντας το Google Maps Web API είναι δυνατή η υψηλή παραμετροποίηση χαρτών με βάση τις ανάγκες της εφαρμογής.



Εικόνα 6-3 browser based διεπαφή

Συγκεκριμένα τα δεδομένα που βρίσκονται στη βάση δεδομένων μετατρέπονται σε αντικείμενα, markers, ώστε να ενσωματωθούν στους χάρτες με πληροφορίες όπως όνομα στάσης, σε ποιο δρομολόγιο ανήκει, η θέση της στον χάρτη αλλά και με τι εικονίδιο θα εμφανίζεται. Για λόγους εξοικονόμησης χωρητικότητας και ταχύτητας απόκρισης από τον server, τα ίδια δεδομένα αξιοποιούνται ώστε να σχηματιστούν τα μονοπάτια, polylines, που ακολουθούν τα δρομολόγια.

Με παρόμοιο τρόπο δημιουργείται και ένας marker για κάθε όχημα με την διαφορά ότι όταν προκύψει νέα θέση στη βάση δεδομένων η εφαρμογή αντικαθιστά την προηγούμενη θέση με την τρέχουσα κινώντας ουσιαστικά στον χάρτη τον marker αυτόν.

Το πλεονέκτημα μιας τέτοια υλοποίησης είναι η δυναμική απεικόνιση του δικτύου των στάσεων και δρομολογίων μιας πόλης ή μιας συγκεκριμένης επιχείρησης παροχής υπηρεσιών μαζικής μεταφοράς. Έτσι η προσθήκη ή αφαίρεση στοιχείων πάνω στον χάρτη δεν απαιτεί προγραμματιστική

παρέμβαση από κάποιον ειδικό αλλά μόνο ένα απλό εργαλείο, ακόμα και web-based, για την επεξεργασία της βάσης δεδομένων το οποίο μπορεί να το χειριστεί ο οποιοσδήποτε έχει ελάχιστες γνώσεις ΗΥ.

Επίσης η επιλεκτική εμφάνιση δρομολογίων είναι δυνατή μέσω μιας λίστας με κουτιά επιλογής αποκλείοντας πληροφορίες άχρηστες προς τον χρήστη και επικεντρώνοντας στα σημαντικά για αυτόν, επιτρέποντας του τον προγραμματισμό των δραστηριοτήτων του από την γνώση των διαθέσιμων οχημάτων και την συχνότητα εμφάνισης τους σε συγκεκριμένα δρομολόγια, που εν δυνάμει θα τον εξυπηρετούσαν και εξαλείφοντας την αίσθηση της αβεβαιότητας.

Η εμφάνιση νέων οχημάτων στον χάρτη δεν απαιτεί ανανέωση ολόκληρης της ιστοσελίδας από τον πρόγραμμα περιήγησης καθώς η ροή δεδομένων είναι συνεχής από την βάση δεδομένων μέσω JQuery σε προκαθορισμένο χρονικό διάστημα.

Η συγκέντρωση όλων αυτών των δεδομένων σε ένα μέρος μπορεί να αξιοποιηθεί προσφέροντας επιπλέον πληροφορίες για την κατάσταση της κυκλοφοριακής συμφόρησης σε συγκεκριμένες ώρες και μέρες προβαίνοντας σε εκτιμήσεις χρονικής ολοκλήρωσης του δρομολογίου ή άφιξης των οχημάτων στην στάση, για την κίνηση του όγκου των επιβατών σε δρομολόγια για την αναδιοργάνωση της κατανομής των οχημάτων εξυπηρετώντας τις ανάγκες του επιβατικού κοινού, για τον έλεγχο και την διαχείριση των οδηγών που βρίσκονται σε υπηρεσία από την διοίκηση και για την εξασφάλιση και διεύρυνση της ασφάλειας των επιβατών.

6.4 Google Maps Javascript API v3

Τον Ιούνιο του 2005 ξεκίνησε την πρώτη δημόσια και δωρεάν παροχή εργαλείων και βιβλιοθηκών για την ανάπτυξη παραμετροποιήσιμων χαρτών μετά από διάφορες δημοφιλής και ανεπίσημες υλοποιήσεις αντίστροφης μηχανικής. Μέχρι σήμερα αποτελεί την πιο πολύ χρησιμοποιημένη υπηρεσία χαρτών με διαφορά και στην 3η έκδοση του API προστίθενται χαρακτηριστικά που κάνουν ακόμα ευκολότερη την παρεμβολή δεδομένων με μεγαλύτερη ποικιλία παρουσίασης τους και την βοήθεια του αντικεινοστραφή προγραμματισμού της JavaScript.

Το API key είναι ένας μοναδικός κλειδάριθμος που απαιτείται για την εισαγωγή της βιβλιοθήκης των Google Maps σε μια ιστοσελίδα, παρέχοντας παράλληλα μηχανισμούς παρακολούθησης της χρήσης. Σε περίπτωση υπέρβασης των ορίων χρήσης που έχει θέση η Google υπάρχει και ανάλογη χρέωση για κάτοχο του API. Παρέχεται δωρεάν με κάθε λογαριασμό για πρόσβαση στις υπηρεσίες της Google. Εφόσον έχουμε δηλώσει το Maps API, πρέπει να γίνει αρχικοποίηση ενός αντικειμένου χάρτη με JavaScript μέσα στο ανάλογο HTML Div καθώς και τις ρυθμίσεις κατάλληλες για την παραμετροποίηση που ταιριάζει στο είδος της εφαρμογής που προορίζεται.

```
<script type="text/javascript"
src="http://maps.google.com/maps/api/js?key=AIzaSyA8M_G5HQQiaPe9eOOvOnuwNEEq5KSYiU&sensor=false&language=eng"></script>
```

Αναλυτικά μπορούμε να ορίσουμε συντεταγμένες στις οποίες αρχικά κεντράρει ο χάρτης, το αρχικό επίπεδο μεγέθυνσης, τι είδους χάρτη επιθυμούμε αν θα είναι δηλαδή δορυφορική εικόνα ή ένας απλός οδικός χάρτης, αν μπορεί ο χρήστης να χρησιμοποιήσει την ροδέλα από το ποντίκι ή και άλλου είδους συντομεύσεις πληκτρολογίου, εάν θα εμφανίζονται τα εικονικά πλήκτρα πλοήγησης και μεγέθυνσης και σε ποια θέση.

Με την ολοκλήρωση της αρχικοποίησης το αντικείμενο του χάρτη (map) μπορεί να χρησιμοποιηθεί για την εισαγωγή αντικειμένων χάρτη όπως markers, polylines, overlays, events και άλλα καθώς και την εξαγωγή και επεξεργασία ήδη εισαχθέντων αντικειμένων.

```

window.onload = function() {
var mapDiv = document.getElementById('map');
var latlng = new google.maps.LatLng(35.336414, 25.124273);
var options = {
    center: latlng,
    zoom: 18,
    mapTypeId: google.maps.MapTypeId.HYBRID,

    scrollwheel: false,
    streetViewControl: false,
    draggable: true,
    disableDoubleClickZoom: true,
    keyboardShortcuts: false,
    scaleControl: true,

    navigationControl: true,
    navigationControlOptions: {
        position: google.maps.ControlPosition.TOP_RIGHT,
        style: google.maps.NavigationControlStyle.ZOOM_PAN
    },

    mapTypeControl: true,
    mapTypeControlOptions: {
        style: google.maps.MapTypeControlStyle.HORIZONTAL_BAR,
        position: google.maps.ControlPosition.BOTTOM,
        mapTypeIds: [
            google.maps.MapTypeId.ROADMAP,
            google.maps.MapTypeId.HYBRID
        ]
    }
};

map = new google.maps.Map(mapDiv, options);

```

6.5 Τεχνικές ανάκτησης δεδομένων από Βάση Δεδομένων με JQuery- Ajax

Το πιο σημαντικό κομμάτι της web-based εφαρμογής είναι η αδιάληπτη και συνεχής τροφοδοσία από τον server με πληροφορίες για την αλλαγή της θέσης των οχημάτων σε πραγματικό χρόνο.

Σε αυτό συμβάλει η τεχνική AJAX (Asynchronous JavaScript And XML) όπου δεν απαιτεί φορτώσεις ολόκληρης της ιστοσελίδας για την ανανέωση του περιεχόμενου αλλά αυτό γίνεται ασύγχρονα ακόμα και μετά την λήψη όλων των απαιτούμενων στοιχείων που συνθέτουν την ιστοσελίδα σε συνδυασμό με ένα εξειδικευμένο webservice.

Για την χρήση της βιβλιοθήκης του JQuery πρέπει να περιληφθεί στον κώδικα της ιστοσελίδας το script tag από την κεντρική ιστοσελίδα της JQuery στην έκδοση που επιθυμούμε.

```
<script src="http://code.jquery.com/jquery-1.10.2.min.js" type="text/javascript"></script>
```

Συγκεκριμένα για την εργασία χρησιμοποιώ έναν συνδυασμό AJAX με υποστήριξη επιστροφής δεδομένων μέσω JSON και με έναν απλό JavaScript Parser μετατρέποντας τα εισερχόμενα πακέτα σε αντικείμενα συμβατά με το Android Maps API όπως markers και polylines για την κατάδειξη στάσεων,

οχημάτων αλλά και δρομολογίων αντίστοιχα.

Στην περίπτωση των στάσεων και των δρομολογίων ο κώδικας δημιουργίας τους από την βάση δεδομένων τρέχει μια φορά αφού είναι λιγότερο πιθανό στην διάρκεια θέασης της ιστοσελίδας από τον χρήστη να αλλάξουν αυτές οι πληροφορίες από το σύστημα. Αντίθετα διαρκής είναι η λήψη πακέτων με πληροφορίες για την αλλαγή της θέσης των οχημάτων εφόσον κινούνται μέσα στην πόλη, κατ' επέκταση πρέπει να κινηθούν και στον χάρτη της web εφαρμογής.

```
$.ajax ( {
  url: "stops.php",
  dataType: "json",
  success: function( data, textStatus, jqXHR ) {
    var places = [];
    var titles = [];
    var tempString;
    var markerIcon = new google.maps.MarkerImage('images/bus/bus-tour.png');
    for(var i=0; i<data.length; i++)
    {
      tempString="<b>στάση:</b> "+data[i].address+"<br><b>δρομολόγιο:</b> "+data[i].route;
      places.push(new google.maps.LatLng(data[i].lat, data[i].lng));
      titles.push(tempString);
    }
    for(var j=0; j<data.length; j++)
    {
      marker = new google.maps.Marker({
        position: places[j],
        map: map,
        title: titles[j],
        icon: markerIcon
      });
      (function(j, marker){
        google.maps.event.addListener(marker, 'click', function(){
          var infowindow = new google.maps.InfoWindow({
            content: titles[j]
          });
          infowindow.open(map, marker);
        });
      })(j, marker);
      var obj ={};
      obj.route=data[j].route;
      obj.marker=marker;
      stops[j]=obj;
    } //end for
  } //end success func
} ); //end ajax
```

Συγκεκριμένα η συνάρτηση livebusUpdate() φέρνει συνεχώς δεδομένα (data[]) από την βάση δεδομένων κάθε ένα δευτερόλεπτο, μετασχηματίζοντας τα σε αντικείμενα JavaScript ώστε να είναι δυνατή η δημιουργία και τοποθέτηση στο χάρτη του marker σε περίπτωση επιτυχούς επικοινωνίας με τον server.

6.6 Ο ρόλος των Web Services

Το σύστημα όπως είναι δομημένο κλιμακώνεται σε πολλά επίπεδα αλλά η πληροφορία πρέπει να έχει την δυνατότητα να διασχίζει με ασφάλεια προς όλες τις κατευθύνσεις είτε αυτό σημαίνει λήψη πληροφοριών από την βάση δεδομένων είτε εισαγωγή δεδομένων σε αυτήν.

Σε αυτό συμβάλουν απλά Web Services γραμμένα σε PHP όπου με αιτήματα JSON μπορούν να επιστρέψουν τα κατάλληλα δεδομένων σε καθένα από τα επίπεδα που τα έχουν αιτηθεί.

```
function kml(){
$.ajax ( {
  url: "livebus.php",
  dataType: "json",
  success: function( data, textStatus, jqXHR ){

    howManyBuses = data.length;
    //get livebuses from cloud to js var
    for(var i=0; i<howManyBuses; i++)
    {
      busidXml.push(data[i].bus_id);
      routeXml.push(data[i].route_id);
      latXml.push(data[i].lat);
      lngXml.push(data[i].lng);
      balun.push(data[i].bus_name);
    }
  },      //end success
  complete: (function(){

    deleteMarkers();
    for(var i=0; i<howManyBuses; i++){
      localMark[i] = new google.maps.LatLng(latXml[i], lngXml[i]);
      addMarker(localMark[i], balun[i]);
    }
    localMark=[];
    latXml=[];
    lngXml=[];
    setAllMap(map);
  }) //end complete
}); //end ajax
} //kml editor end
setInterval(kml, 1000);
```

Συγκεκριμένα, η χειραγώγηση των αντικειμένων του χάρτη γίνεται με JavaScript πράγμα που σημαίνει ότι αφενός εκτελείται στο μηχάνημα του πελάτη αφετέρου ότι δεν υπάρχει άμεση πρόσβαση στη βάση δεδομένων και στις αποθηκευμένες πληροφορίες που αυτή διαθέτει. Το ρόλο του δίαυλου επικοινωνίας αναλαμβάνει ο συνδυασμός τεχνικών AJAX και Web Services.

Ουσιαστικά τα web services είναι προκατασκευασμένα ερωτήματα για την βάση δεδομένων τα

```

<?php
include "conDB.php";
//build query
$query = "SELECT * FROM stops INNER JOIN routes on stops.route_id=routes.route_id";
//Execute query
$result = mysql_query($query) or die(mysql_error());
//mysql query to php assoc array
while($row = mysql_fetch_assoc($result))
{
    $array[] = $row; // Inside while loop
}
//$json = json_encode($array, JSON_FORCE_OBJECT);
echo json_encode($array);
mysql_free_result($result);
?>

```

οποία εκτελούνται σε μη προβλέψιμο χρόνο μετά από μια κλήση AJAX στην περίπτωση της web-based εφαρμογής ή αιτήματος με πακέτο JSON στην περίπτωση του Android app και έχουν ως απάντηση ένα δομημένο πακέτο JSON, για εύκολη προσπέλαση των δεδομένων με έναν απλό JSON Parser και μετατροπή τους σε αντικείμενα κατανοητά από διαφορετικές γλώσσες προγραμματισμού.

Παράλληλα για την εισαγωγή πληροφοριών στη βάση δεδομένων απαιτείται ένα άλλου είδους web service που θα έχει την δυνατότητα να αναγνωρίζει πακέτα JSON, να ανακτά τα δεδομένα που αυτό περιέχει ή με όποια χρήσιμη πληροφορία τα συνοδεύει και ταυτόχρονα να καλεί ερωτήματα για την εισαγωγή ή ανανέωση αυτών εντός της βάσης δεδομένων με επιστροφή της έκβασης της διαδικασίας αυτής.

```

<?php
include "conDB.php";

$json = file_get_contents('http://input');
$obj = json_decode($json);
$lat = $obj->('lat');
$lng = $obj->('lng');
$bus_id=$obj->('bus_id');
$route_id=$obj->('route_id');

$query="UPDATE livebus SET lat=$lat, lng=$lng WHERE bus_id=$bus_id AND route_id=$route_id";
$result = mysql_query($query) or die(mysql_error());

mysql_close($con);

?>

```

6.7 Αποστολή και λήψη δεδομένων από Βάση Δεδομένων για την εφαρμογή Android

Όπως και στην περίπτωση του Ajax/JQuery, έτσι και η πρόσβαση σε δεδομένα μιας απομακρυσμένης βάσης δεδομένων από το Android γίνεται μέσω αιτημάτων HTTP post και απάντηση των web services με δεδομένα σε πακέτα JSON. Για την ανάκτηση των δεδομένων που βρίσκονται στα πακέτα αυτά απαιτείται και ο ανάλογος JSON parser αλλά και η ενσωματωμένη στο λειτουργικό android, βάση δεδομένων SQLite, για την προσωρινή αποθήκευση και ανάκτησή των δεδομένων σε οποιοδήποτε χρονική στιγμή.

Αναλυτικά με σε μια μέθοδο (makeHTTPPOSTRequest) δημιουργούμε ένα αντικείμενο HttpClient για την κλήση του αιτήματος αλλά και για την ανάκτηση του αποτελέσματός του, ένα αντικείμενο αιτήματος POST το HttpPost με όρισμα το URL του web service του οποίου τα δεδομένα αιτούμαστε, αντικείμενο για την απάντηση HttpResponse και τέλος το αντικείμενο JSONObject για να μπορέσει ο parser να διατρέξει και να ανακτήσει τα χρήσιμα δεδομένα από την απάντηση του HttpResponse σε JSON πακέτο και από εκεί να καταλήξει στην τοπική βάση δεδομένων SQLite της εφαρμογής. Από την στιγμή που ολοκληρώνεται η μέθοδος makeHTTPPOSTRequest() τα δεδομένα είναι διαθέσιμα για τις υπόλοιπες κλάσεις της υλοποίησης για τον μετασχηματισμό τους σε αντικείμενα συμβατά με το Google Maps API αλλά και για περεταίρω επεξεργασία αν προκύψουν διαφοροποιήσεις, όπως αλλαγή θέσης ενός οχήματος.

```

public void makeHTTPPOSTRequest(Context context) {
    try {
        HttpClient c = new DefaultHttpClient();
        HttpPost p = new HttpPost(this.apiUrl);
        p.setEntity(new StringEntity("{\"username\":\"" + this.apiusername + "\",\"password\":\"" + this.apipassword + "\"}"));
        HttpResponse r = c.execute(p);
        BufferedReader rd = new BufferedReader(new InputStreamReader(r.getEntity().getContent(), "UTF-8"));
        String line = rd.readLine();
        JSONObject token = new JSONObject(line);
        try {
            finalResult = new JSONArray(token);
        } catch (JSONException e) { e.printStackTrace(); }

        try {
            org.json.JSONObject jsonO;
            db = new DbAndroidMap(context);
            DatabaseManager lol = new DatabaseManager();
            lol.initializeInstance(db);
            if (theRoute == "stops") {
                for (int i=0; i<finalResult.length(); i++) {
                    jsonO = finalResult.getJSONObject(i);
                    String stopsRoute = jsonO.get("route").toString();
                    String stopsAddress = jsonO.get("address").toString();
                    String stopsLat = jsonO.get("lat").toString();
                    String stopsLng = jsonO.get("lng").toString();
                    db.insertStops(stopsRoute, stopsAddress, stopsLat, stopsLng);
                }
            } else if (theRoute == "routes") {
                for (int i=0; i<finalResult.length(); i++) {
                    jsonO = finalResult.getJSONObject(i);
                    String routeRoute = jsonO.get("route").toString();
                    db.insertRoutes(routeRoute);
                }
            }
        } catch (JSONException e) { e.printStackTrace(); }
    } catch (IOException e) { System.out.println(e); }
}

```

Το android περιορίζει την λειτουργία των εφαρμογών του σε ένα μόνο νήμα το UI Thread. Προκειμένου λοιπόν να μην παγώνει η διεπαφή χρήστη κάθε φορά που χρειαζόμαστε επικοινωνία με έναν διακομιστή ή την εκτέλεση κάποιας διεργασίας σε άλλο νήμα χρησιμοποιούμε μια περιληπτική (abstract) κλάση σχεδιασμένη για πρόσβαση σε αυτά τα νήματα.

Για την εκτέλεση λοιπόν μιας τέτοια διαδικασία είναι αναγκαία η κλήση της μέσω μιας AsyncTask, η οποία εκτελείται σε παρασκηνιακό νήμα με ασύγχρονο τρόπο, καθώς οτιδήποτε άλλο δεν είναι επιτρεπτό από το ίδιο το λειτουργικό λόγω του περιορισμένου υλικού (hardware) στο οποίο τρέχει.

```
private class AsyncSend extends AsyncTask<Void, Void, String>{

    private final String stopsURL = "http://"+serverIp+":8080/livebus/stops.php";
    private final String routesURL = "http://"+serverIp+":8080/livebus/routes.php";
    private static final String username = "lefteris";
    private static final String password = "thesis";

    @Override
    protected String doInBackground(Void... params) {
        new SimpleHTTPPOSTRequester(username, password, routesURL, "routes").makeHTTPPOSTRequest(context);
        new SimpleHTTPPOSTRequester(username, password, stopsURL, "stops").makeHTTPPOSTRequest(context);

        return "Completed!";
    }

    @Override
    protected void onPostExecute(String result) {
        //Log.i("POSA Async routes", result);
        //handler.post(placeMapStuff);
    }

    @Override
    protected void onProgressUpdate(Void... values) {
        //Log.d("ProgressUpdate", "You are in progress update ... ");
    }

    @Override
    protected void onPreExecute() {
        try{
            db = new DbAndroidMap(context);
            DatabaseManager lol =new DatabaseManager();
            lol.initializeInstance(db);
            SQLiteDatabase database = DatabaseManager.getInstance().openDatabase();
            database.setLocale(Locale.getDefault());
            DatabaseManager.getInstance().deletedb(table);
            DatabaseManager.getInstance().deletedb(tableRoute);
            DatabaseManager.getInstance().closeDatabase();
        }catch(Exception e){e.printStackTrace();}
    }
}
```

Η AsyncTask εκτελείται σε τέσσερά βήματα onPreExecute() πριν την εκτέλεση του παρασκηνιακού νήματος, doInBackground() για την εκτέλεση του κώδικα μας στο παρασκήνιο, onProgressUpdate() για την δημοσίευση στο UI Thread την πρόοδο της doInBackground() μέσω του αντικειμένου publishProgress και τέλος η μέθοδος onPostExecute() όπου περνιούνται τα αποτελέσματα από το παρασκήνιο στο UI Thread.

6.8 Προγραμματισμένη εκτέλεση κώδικα σε καθορισμένα χρονικά διαστήματα

Για να υλοποιηθεί μια εφαρμογή ζωντανής αναμετάδοσης πληροφοριών πρέπει να είναι σε θέση να επικοινωνεί με τον εξυπηρετητή ανα τακτικά χρονικά διαστήματα. Συγκεκριμένα, οι εφαρμογές Android είναι μονονηματικές, πράγμα το οποίο σημαίνει ότι η επικοινωνία με έναν εξυπηρετητή θα πάγωνε την διάδραση με τον χρήστη.

Γι' αυτόν τον λόγο στο Android API υπάρχει η περιληπτική κλάση AsyncTask η οποία παρέχει την δυνατότητα εκτέλεσης απαιτητικών διεργασιών σε ένα δεύτερο νήμα ασύγχρονα στο παρασκήνιο. Αντικείμενο με τέτοια λειτουργικότητα είναι το requestAsyncLivebus manipulateLivebus μέσω του οποίου εκτελούνται οι διεργασίες που έχουν σχέση με τα κινούμενα οχήματα, δηλαδή τον υπολογισμό της ταχύτητας, των αποστάσεων του από συγκεκριμένα σημεία ενδιαφέροντος, την επικοινωνία με τον εξυπηρετητή για την ανανέωση της θέσης τους με την τοποθέτηση ή την αλλαγή των Markers που αντιπροσωπεύουν τα οχήματα.

Την χρονική επανάληψη της εκτέλεσης των μεθόδων του αντικειμένου manipulate αναλαμβάνει ένα αντικείμενο Timer με την δικιά του μέθοδο schedule με καθορισμένο βήμα σε millisec. Η μεταβλητή

loopTime καθορίζει σε δευτερόλεπτα την περίοδο επανάληψης του κώδικα εντός του TimerTask. Παράλληλα όμως και μετά την εισαγωγή του στο αντικείμενο manipulateLivebus μέσω της μεθόδου setLoopPeriod(loopTime) βοηθά στον υπολογισμό της ταχύτητας του οχήματος, αρά και του εκτιμώμενου χρόνου έλευσης του στην στάση κοντά στον πελάτη.

```
(scheduled) {
    timer.cancel();
    timer.purge();
    scheduled = false;

    timer = new Timer();
    timer.schedule(new TimerTask() {
        @Override
        public void run() {
            scheduled = true;
            manipulateLivebus.setSelectedRoutes(choiceRoutes);
            if(gps.getTheClosestStop()!=null)
                manipulateLivebus.setClosestStop(gps.getTheClosestStop());
            else
                manipulateLivebus.setClosestStop(new LatLng(gps.latitude, gps.longitude));
            manipulateLivebus.setLoopPeriod(loopTime);
            manipulateLivebus.execute(ip);
        }
    });
}
```

6.9 Android Maps API v2

Είναι η Android έκδοση των βιβλιοθηκών για την ανάπτυξη εφαρμογών με ενσωματωμένες λειτουργίες χάρτη σε φορητές συσκευές με Google Play Services. Είναι αντίστοιχο του JavaScript API v3 στον τρόπο δημιουργίας των αντικειμένων, έχει αντικειμενοστραφή φιλοσοφία, απαιτεί ειδικό API key διαφορετικό της JavaScript έκδοσης για ανάλογες υπηρεσίες παρακολούθησης, χρήσης και χρέωσης και είναι γραμμένο σε Java. Επίσης κυκλοφορεί και για λειτουργικό iOS σε γλώσσα Objective C.

Η πληθώρα αισθητήρων της πλατφόρμας Android σε συνδυασμό με το Maps API δίνουν την δυνατότητα για εφαρμογές με αμεσότητα προς τον χρήστη, αυτοματοποιημένες διαδικασίες, συνδυασμό υπηρεσιών, ασφάλεια των προσωπικών δεδομένων και καλύτερη απόκριση από την αξιοποίηση των διαθέσιμων πόρων της συσκευής.

```
public class MapPane extends Fragment
{
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState)
    {
        view = inflater.inflate(R.layout.map_activity, container, false);

        map = ((MapFragment) getFragmentManager()
            .findFragmentById(R.id.map)).getMap();

        LatLng heraklion = new LatLng(35.336414, 25.124273);
        map.setMyLocationEnabled(true);
        map.moveCamera(CameraUpdateFactory.newLatLngZoom(heraklion, 13));
        map.setMapType(GoogleMap.MAP_TYPE_NORMAL);

        return view;
    }
}
```

Για την χρήση του αντικειμένου map απαιτείται η ενσωμάτωσή του σε ένα layout, για να είναι

δυνατή η εμφάνιση του χάρτη και των επιμέρους στοιχείων του όπως επίσης να είναι δυνατή η εναλλαγή μεταξύ των διαφορετικών layout, όψεις, της εφαρμογής.

Επίσης πρέπει να οριστεί το αρχικό κέντρο του χάρτη και το επίπεδο μεγέθυνσης, για εμφάνιση του κομματιού του χάρτη που μας ενδιαφέρει, ο τύπος του χάρτη, αν δηλαδή θα είναι οδικός ή αν θα είναι δορυφορικής εικόνας χάρτης, εάν θα κάνει χρήση των δυνατοτήτων των αισθητήρων θέσης αξιοποιώντας τις ανάλογες βιβλιοθήκες.

```
BitmapDescriptor busStopIcon = BitmapDescriptorFactory.fromResource(R.drawable.busstop);
for (MarkerOptions marker1 : markers) {marker1.icon(busStopIcon);}
for (MarkerOptions marker : markers) {map.addMarker(marker);}

ArrayList<PolylineOptions> polyLines = new returnPolyLines().getStopsPoly(markers, routesStrings);
for (PolylineOptions polyline : polyLines) {map.addPolyline(polyline);}
```

Μετά την αρχικοποίηση του αντικειμένου map, χρησιμοποιείται το στιγμιότυπό του για την εισαγωγή ή αφαίρεση των markers και polyLines ανάλογα το σενάριο εκτέλεσης της εφαρμογής. Η μέθοδος addMarker() του map τοποθετεί τα αντικείμενα Marker στον χάρτη έχει ως όρισμα το αντικείμενο MarkerOptions που περιέχει πληροφορίες όπως γεωγραφικό μήκος και πλάτος, εικονίδιο, ετικέτα και επιστρέφει ένα αντικείμενο Marker, το οποίο μπορούμε να επεξεργαστούμε ανάλογα για την αλλαγή των χαρακτηριστικών του. Αντίστοιχα για την τοποθέτηση των polyline στο χάρτη χρησιμοποιούμε την μέθοδο addPolyline().

Επιπλέον η Utility Library του εν λόγω API παρέχει μεγαλύτερη παραμετροποίηση όπως οι heatmaps, οι οποίοι χρησιμοποιούν χρώμα και σχήμα για να αναπαραστήσουν την συγκέντρωση και σχετικότητα των δεδομένων στο χάρτη, εξιδεικευμένα εικονίδια marker με πληροφορίες μέσω του IconGenerator, τα συμπλέγματα marker για την ομαδοποίηση πολλών markers σε έναν για συγκεκριμένα επίπεδα μεγέθυνσης χάρτη, η κωδικοποίηση και αποκωδικοποίηση polyline με το PolyUtil για συμβατότητα και διαλειτουργικότητα με το Directions API καθώς και διάφορες μεθόδους υπολογισμών αποστάσεων μεταξύ σημείων computeDistanceBetween(), κατεύθυνσης σε μοίρες μεταξύ δύο συντεταγμένων computeHeading(), επιφανειας σε τετραγωνικά μέτρα μιας κλειστής περιοχής εδάφους computeArea() και επιστροφή θέσης μεταξύ κλάσματος συντεταγμένων δύο σημείων με το interpolate().

6.10 Χρήση της βιβλιοθήκης Location

Τα σύγχρονα λειτουργικά συστήματα κινητών συσκευών, όπως smartphones και tablets, επιτρέπουν την χρήση και τον χειρισμό των υποσυστημάτων τους όπως wifi, GPS, Bluetooth ή NFC, προγραμματιστικά μέσω υψηλού επιπέδου βιβλιοθήκης σε σχέση με το παρελθόν εξαλείφοντας την ανάγκη γνώσης της λειτουργίας αυτών στο κατώτερο επίπεδο ή αποφεύγοντας τυχόν εξαρτήσεις από συγκεκριμένους κατασκευαστές για συγκεκριμένες συσκευές.

```
public Location getLocation() {
    try {
        locationManager = (LocationManager) mContext
            .getSystemService(LOCATION_SERVICE);

        // getting GPS status
        isGPSEnabled = locationManager
            .isProviderEnabled(LocationManager.GPS_PROVIDER);

        // getting network status
        isNetworkEnabled = locationManager
            .isProviderEnabled(LocationManager.NETWORK_PROVIDER);

        if (!isGPSEnabled && !isNetworkEnabled) {
            // no network provider is enabled
        } else {
            this.canGetLocation = true;
            if (isNetworkEnabled) {
                locationManager.requestLocationUpdates(
                    LocationManager.NETWORK_PROVIDER,
                    MIN_TIME_BW_UPDATES,
                    MIN_DISTANCE_CHANGE_FOR_UPDATES, this);
                Log.d("Network", "Network");
                if (locationManager != null) {
                    location = locationManager
                        .getLastKnownLocation(LocationManager.NETWORK_PROVIDER);
                    if (location != null) {
                        latitude = location.getLatitude();
                        longitude = location.getLongitude();
                    }
                }
            }
            // if GPS Enabled get lat/long using GPS Services
            if (isGPSEnabled) {
                if (location == null) {
                    locationManager.requestLocationUpdates(
                        LocationManager.GPS_PROVIDER,
                        MIN_TIME_BW_UPDATES,
                        MIN_DISTANCE_CHANGE_FOR_UPDATES, this);
                    Log.d("GPS Enabled", "GPS Enabled");
                    if (locationManager != null) {
                        location = locationManager
                            .getLastKnownLocation(LocationManager.GPS_PROVIDER);
                        if (location != null) {
                            latitude = location.getLatitude();
                            longitude = location.getLongitude();
                        }
                    }
                }
            }
        }

    } catch (Exception e) {
        e.printStackTrace();
    }
    return location;
}
```

Με αυτόν τον τρόπο επιταχύνεται η ανάπτυξη εφαρμογών που κάνουν χρήση των περίπλοκων υποσυστημάτων προσφέροντας νέες εμπειρίες διάδρασης στους κατόχους, επεκτείνουν την χρήση ήδη υπαρχόντων υπηρεσιών ή διευρύνουν τους τρόπους παρουσίασης των διαθέσιμων πληροφοριών.

Συγκεκριμένα διαθέσιμες προς χρήση βιβλιοθήκες για τον έλεγχο της κατάστασης που βρίσκεται ο δέκτης GPS, εάν δηλαδή είναι ενεργοποιημένος ή όχι, κάθε πότε επιστρέφει στην εφαρμογή δεδομένα σχετικά με την θέση του, εναλλακτική διαθεσιμότητα πληροφοριών θέσης σε περίπτωση αποτυχία του GPS μέσω του πάροχου κινητής αλλά και location listeners που ενεργοποιούν συγκεκριμένα αντικείμενα από τον κώδικα σε δεδομένες αλλαγές της θέσης του δέκτη.

```
@Override
public void onLocationChanged(Location location) {
    try{
        if(calledJustOnce){
            positionMarker.setPosition(new LatLng(location.getLatitude(), location.getLongitude()));
        }
        else{
            setMyLocationMarker(getLocation());
        }
    }catch (Exception e){e.printStackTrace();}
}
```

Παράλληλα και με την συμβολή των τεχνικών του αντικειμενοστραφή προγραμματισμού είναι εφικτή η επαναχρησιμοποίηση αντικείμενων για διαφορετικά σενάρια, όπως στην προκειμένη περίπτωση στην μεριά του πελάτη για την πλοήγηση του στην πιο κοντινή στάση και στην μεριά του οδηγού για να προωθήσει τη θέση του στον διακομιστή.

6.11 Google Directions API

Το Google Directions API είναι μια υπηρεσία που υπολογίζει τις οδηγίες ανάμεσα σε δύο τοποθεσίες χρησιμοποιώντας HTTP αιτήματα. Υπάρχει η δυνατότητα αναζήτησης οδηγιών με διάφορα μέσα μεταφοράς όπως με αυτοκίνητο, με τα πόδια ή με ποδήλατο, επίσης μπορεί να επιστρέφει την αφετηρία, τον προορισμό και την διαδρομή είτε λεκτικά είτε ως συντεταγμένες γεωγραφικού μήκους και πλάτους υπολογίζοντας ταυτόχρονα τον εκτιμώμενο χρόνο διάσχισης της διαδρομής ανάλογα με το επιλεγμένο μέσο.

```
public String getDirectionsUrl(LatLng origin,LatLng dest){

    // Origin of route
    String str_origin = "origin="+origin.latitude+", "+origin.longitude;

    // Destination of route
    String str_dest = "destination="+dest.latitude+", "+dest.longitude;

    // Sensor enabled
    String sensor = "sensor=false";

    // Building the parameters to the web service
    String parameters = str_origin+"&"+str_dest+"&"+sensor+"&mode=walking&language=EL";

    // Output format
    String output = "json";

    // Building the url to the web service
    String url = "https://maps.googleapis.com/maps/api/directions/" +output+"?"+parameters;

    return url;
}
```

Η μορφή του HTTP αιτήματος είναι συγκεκριμένη όπως φαίνεται παραπάνω, με διάφορες επιλογές και τρόπους επιστροφής της πληροφορίας. Μεταξύ άλλων παρέχεται η δυνατότητα επιστροφής πληροφοριών σε JSON ή XML ενώ υποχρεωτικοί είναι μόνο οι παράμετροι αφετηρίας και προορισμού.

```
public static String downloadUrl(String strUrl) throws IOException {
    String data = "";
    InputStream iStream = null;
    HttpURLConnection urlConnection = null;
    try{
        URL url = new URL(strUrl);

        // Creating an http connection to communicate with url
        urlConnection = (HttpURLConnection) url.openConnection();

        // Connecting to url
        urlConnection.connect();

        // Reading data from url
        iStream = urlConnection.getInputStream();

        BufferedReader br = new BufferedReader(new InputStreamReader(iStream));

        StringBuffer sb = new StringBuffer();

        String line = "";
        while( ( line = br.readLine()) != null){
            sb.append(line);
        }

        data = sb.toString();

        br.close();

    }catch(Exception e){
        Log.d("Exception while downloading url", e.toString());
    }finally{
        iStream.close();
        urlConnection.disconnect();
    }
    return data;
}
```

Στις προαιρετικές παραμέτρους συμπεριλαμβάνονται το μέσο μεταφοράς, αν θα επιστρέψει μονοπάτι σε πίνακα σημείων γεωγραφικού μήκους και πλάτους, αν θα δώσει εναλλακτικές διαδρομές, αν θα αποφύγει διόδια, αυτοκινητόδρομους ή πλοία μεταφοράς αυτοκινήτων, σε τι γλώσσα θα επιστρέψει τις πληροφορίες, σε τι μονάδα μέτρησης, δυνατότητα καθορισμού χρόνου αναχώρησης με επιστροφή εκτιμώμενου χρόνου άφιξης, δυνατότητα καθορισμού χρόνου άφιξης, καθορισμός επιθυμίας λιγότερου περπατήματος ή λιγότερων μεταφορικών μέσων.

Όπως και οι περισσότερες υπηρεσίες της Google έτσι και αυτή υπόκειται σε περιορισμούς με την δωρεάν πρόσβαση να επιτρέπει τις 2.500 αιτήσεις για οδηγίες ανα 24 ώρες και 2 αιτήματα το δευτερόλεπτο, ενώ η επι πληρωμή σε διάστημα 24 ωρών επιτρέπει 100.000 αιτήσεις με 10 ανα δευτερόλεπτο. Απαιτείται η ενσωμάτωση του API Key για την χρήση και παρακολούθηση του εν λόγω web service.

<http://maps.googleapis.com/maps/api/directions/output?parameters>

Η χρήση της υπηρεσίας σε συστήματα πλοήγησης αποτελεί πρόκληση αφού είναι σχεδιασμένη για στατικές διευθύνσεις και όχι κινούμενους χρήστες, επίσης οι απαντήσεις δεν δίνονται σε πραγματικό χρόνο ανάλογα με την θέση του χρήστη για παράδειγμα.

Παρόλα αυτά, υπάρχει η δυνατότητα επεξεργασίας μέσω αλγορίθμου στην πλευρά του χρήστη και με τον συνδυασμό συμβάντων θέσης του επιστρεφόμενου μονοπατιού (polyline) με υλοποίηση κοντά σε πραγματικό χρόνο.

Σε περιβάλλον Java, δεν υπάρχουν έτοιμες βιβλιοθήκες για τον χειρισμό των αιτημάτων του Directions API ούτε των απαντήσεων του από το Android SDK. Απαιτείται η δημιουργία αντικειμένου το οποίο κάνει κλήσεις σε HTTP, το HttpURLConnection, και λαμβάνει τα ανάλογα πακέτα πληροφοριών σε ένα InputStream από όπου μπορούν να ανακτηθούν.

Μετά από μια επιτυχή απάντηση του web service, τα εισερχόμενα δεδομένα πρέπει να περάσουν από έναν Parser, μέσω του οποίου μετασχηματίζονται σε χρήσιμες πληροφορίες και αντικείμενα που μπορούν να αξιοποιηθούν από την εφαρμογή.

```
private List<LatLng> decodePoly(String encoded) {  
  
    List<LatLng> poly = new ArrayList<LatLng>();  
    int index = 0, len = encoded.length();  
    int lat = 0, lng = 0;  
  
    while (index < len) {  
        int b, shift = 0, result = 0;  
        do {  
            b = encoded.charAt(index++) - 63;  
            result |= (b & 0x1f) << shift;  
            shift += 5;  
        } while (b >= 0x20);  
        int dlat = ((result & 1) != 0 ? ~(result >> 1) : (result >> 1));  
        lat += dlat;  
  
        shift = 0;  
        result = 0;  
        do {  
            b = encoded.charAt(index++) - 63;  
            result |= (b & 0x1f) << shift;  
            shift += 5;  
        } while (b >= 0x20);  
        int dlng = ((result & 1) != 0 ? ~(result >> 1) : (result >> 1));  
        lng += dlng;  
  
        LatLng p = new LatLng((((double) lat / 1E5)),  
                               (((double) lng / 1E5)));  
        poly.add(p);  
    }  
  
    return poly;  
}
```

Συγκεκριμένα, τα επιμέρους σημεία μιας διαδρομής, δηλαδή το γεωγραφικό μήκος και πλάτος της, έρχονται ως το κωδικοποιημένο πεδίο “points” στο πακέτο JSON. Η μέθοδος αποκωδικοποίησης δίνεται παραπάνω, παίρνει σαν όρισμα τη συμβολοσειρά της πληροφορίας του πεδίου “points”, με επανάληψη while() γίνεται η προσπέλαση των ζευγαριών των συντεταγμένων ενώ με τις εμφωλευμένες επανάληψης do-while() παίρνουμε διαδοχικά το γεωγραφικό μήκος και πλάτος ώστε εν τέλη να συσταθεί μια λίστα (List<LatLng>) από αντικείμενα ζευγών συντεταγμένων (LatLng) που απαιτείται για την δημιουργία μιας διαδρομής πάνω στο χάρτη, δηλαδή polyline.

6.12 setDirectionsPolyline(Location)

Η δυνατότητα παροχής υπηρεσιών πλοήγησης μέσα στην εφαρμογή ζωντανής αναμετάδοσης της θέσης είναι μια διαδικασία που καλείται να αξιοποιήσει την δωρεάν webservice Directions API. Όπως έχει προαναφερθεί, η εν λόγω υπηρεσία έχει κάποιους περιορισμούς ως προς τον όγκο των αιτημάτων παροχής απαντήσεων αλλά και στον χρόνο υπολογισμού και λήψης της απάντησης, καθιστώντας την ακατάλληλη για πλοήγηση σε πραγματικό χρόνο.

Γι' αυτό επιλέγεται μια συνδυαστική προσέγγιση όπου γίνεται πολύ περιορισμένη και υπό ειδικές συνθήκες χρήση της υπηρεσίας. Το βάρος της επεξεργασίας των οδηγιών ανάλογα με την θέση διευθετείται αλγοριθμικά και τοπικά από την εφαρμογή του πελάτη.

Ειδικότερα, με βάση τις συνθήκες η μέθοδος setDirectionalPolyline() αποφασίζει πότε θα επικοινωνήσει με την webservice για την παροχή της Polyline οδηγιών ή πότε απλώς θα την επεξεργαστεί τροφοδοτούμενη από την κίνηση του πελάτη.

```
private void setDirectionsPolyline(Location location) {
    if(myDirect==null)
        myDirect = new DirectionHandler(map, textVDD);
    if(!addedMarkers.isEmpty()) {
        indexOfClosestAddedMarker = getTheClosestMarker(location, addedMarkers);

        if(previousIndexOfClosestAddedMarker===-2)
            previousIndexOfClosestAddedMarker=indexOfClosestAddedMarker;
        else if(previousIndexOfClosestAddedMarker!=indexOfClosestAddedMarker){
            removeDirectionsPolyline();
            polylineToChange = myDirect.getFinalPolyline();
            previousIndexOfClosestAddedMarker=indexOfClosestAddedMarker;
            setRequestDirectionalPolyline();
        }

        if(indexOfClosestAddedMarker>=-1 && requestDirectionalPolyline) {

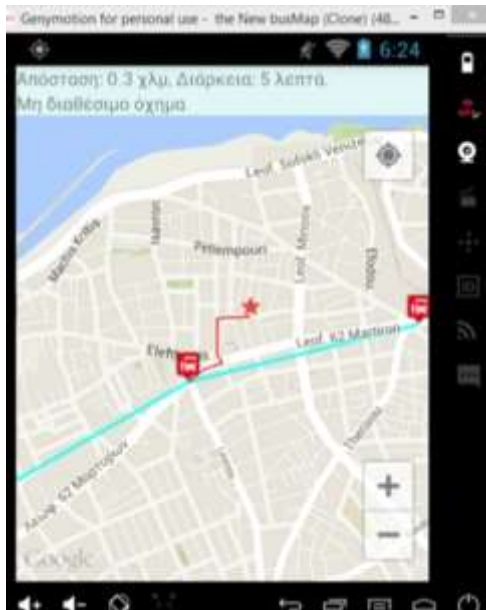
            downloadTask = new DirectionHandler.DownloadTask();
            theClosestStop=addedMarkers.get(indexOfClosestAddedMarker).getPosition();
            String urlForDirections = myDirect.getDirectionsUrl(new LatLng(location.getLatitude(), location.getLongitude()),
                addedMarkers.get(indexOfClosestAddedMarker).getPosition());

            downloadTask.execute(urlForDirections);
            requestDirectionalPolyline =false;
            removeDirectionsPolyline();
            resetDistanceCounter();
        }
        else if(indexOfClosestAddedMarker>=-1 && !requestDirectionalPolyline){
            if(polylineToChange==null && myDirect!=null) {
                polylineToChange = myDirect.getFinalPolyline();
            }
            resetDirectionPolyline(location);
        }
        else {
            removeDirectionsPolyline();
            setRequestDirectionalPolyline();
            resetDistanceCounter();
            textVDD.setText("#τάσσετε στην στάση!");
        }
    }
    else
        textVDD.setText("Προσωρινά μη διαθέσιμη διαδρομή!");
}
```

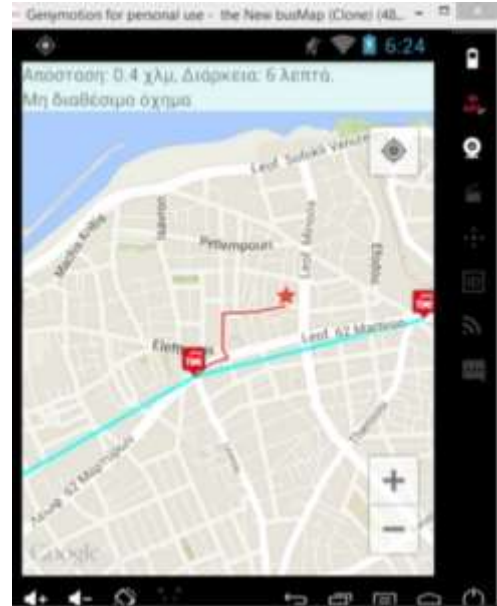
Αρχικά η μέθοδος getTheClosestMarker() με όρισμα την τρέχουσα τοποθεσία του πελάτη και τους Marker των στάσεων που έχουν τοποθετηθεί στον χάρτη ανάλογα με το ποια δρομολόγία έχει επιλέξει, υπολογίζει την κοντινότερη στάση και επιστρέφει τον ανάλογο δείκτη του ArrayList<Marker> addedMarkers.

Παράλληλα η σύγκριση previousIndexOfClosestAddedMarker με τον indexOfClosestAddedMarker ανιχνεύει κατά πόσο ο πελάτης ακολουθεί την Polyline οδηγιών ή κινείται τόσο λάθος ώστε να φτάνει σε διπλανή στάση. Αν αποδειχθούν διαφορετικοί αυτοί οι δείκτες τότε γίνεται επαναυπολογισμός της Polyline με βάση τα νέα δεδομένα δηλαδή την νέα κοντινότερη στάση.

Στην περίπτωση που ο δείκτης αυτός είναι μεγαλύτερος του -1 και η boolean μεταβλητή requestDirectionalPolyline αληθής, εκτελείται η AsyncTask για το αίτημα παροχής οδηγιών από το

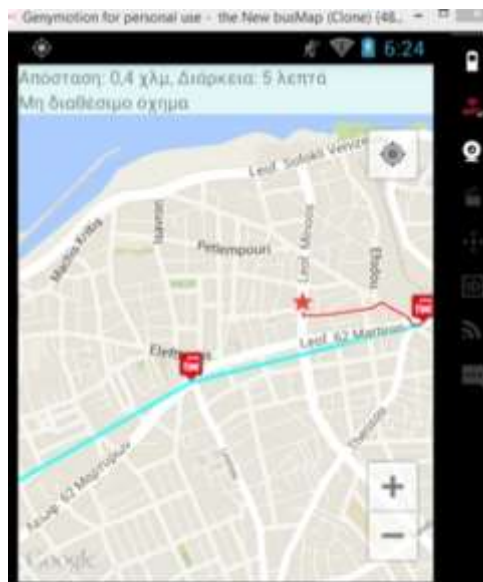


Εικόνα 6-12α: Αρχική θέση πελάτη.



Εικόνα 6-12β: Απομακρύνεται από την κοντινότερη στάση.

Directions API, αφαιρεί τυχόν προηγούμενες Polyline οδηγιών σε περίπτωση εκτέλεσης του κώδικα λόγω αλλαγής της κοντινότερης στάσης. Πρέπει να σημειωθεί ότι το αντικείμενο `downloadTask` ενσωματώνει `AsyncTask` πράγμα που σημαίνει ότι η Polyline οδηγιών θα προστεθεί στο αντικείμενο του χάρτη σε δεύτερο χρόνο και όχι με την εκτέλεση της εντολής μόλις ολοκληρωθεί η επικοινωνία με το Directions API.



Εικόνα 6-12γ: Πλέον η κοντινότερη στάση δεν είναι η αρχική.

Σε αντίθεση με τα παραπάνω όταν η μεταβλητή boolean είναι ψευδής τότε η ήδη τοποθετημένη Polyline πρέπει να αναθεωρηθεί σε σχέση με την θέση του πελάτη είτε αυξάνοντας τα σημεία που την συνθέτουν είτε μειώνοντας τα όπως ορίζει η μέθοδος `resetDirectionalPolyline(Location)` η οποία αναλύεται παρακάτω.

Τέλος στην περίπτωση που η μέθοδος `getTheClosestMarker()` επιστρέφει `-1` τότε ο πελάτης έχει φτάσει στην επιθυμητή στάση, ταυτόχρονα προβάλλεται το μήνυμα «Φτάσατε στην στάση» στο ανάλογο `TextView`, αρχικοποιείται η μεταβλητή `boolean requestDirectionalPolyline` (δηλαδή ορίζεται ως αληθής) καθώς επίσης ο `counter` για την ανίχνευση τυχόν απόκλισης της κατεύθυνσης του πελάτη από την `Polyline` οδηγιών αλλά και αφαιρείται αυτή καθαυτή η `Polyline`.

6.13 `resetDirectionPolyline(Location)`

Η μέθοδος `resetDirectionPolyline()` είναι σε θέση να παίρνει αποφάσεις για το είδος της επεξεργασίας που επίκειται στην `Polyline` οδηγιών ανάλογα με την θέση του πελάτη παραπέμποντας στις κατάλληλες μεθόδους. Έχει ως όρισμα την θέση του πελάτη αλλά έχει και πρόσβαση στην `polylineToChange` για την εξαγωγή των σημείων που απαρτίζουν την τρέχουσα `Polyline` οδηγιών.

```
private void resetDirectionPolyline(Location location){
    List<LatLng> remainingPoints = polylineToChange.getPoints();
    float[] kontinoteroSimio= new float[remainingPoints.size()];
    float[] distanceArray = new float[1];

    for(int i=0; i<remainingPoints.size(); i++){
        Location.distanceBetween(location.getLatitude(), location.getLongitude(),
                                remainingPoints.latitude, remainingPoints.longitude, distanceArray);
        kontinoteroSimio[i]=distanceArray[0];
    }

    int minIndex = 0;
    for (int i = 1; i < kontinoteroSimio.length; i++){
        float newnumber = kontinoteroSimio[i];
        if ((newnumber < kontinoteroSimio[minIndex])){
            minIndex = i;
        }
    }

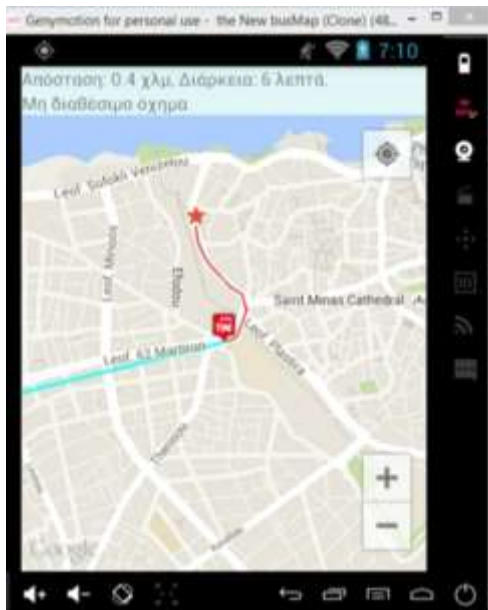
    if(minIndex==0 && getDistanceCounter()<10)
        addPointsToPolyline(remainingPoints, location);
    else if(minIndex==0 && getDistanceCounter()==10){
        requestDirectionalPolyline=true;
        resetDistanceCounter();
    }
    else{
        popPointsFromPolyline(remainingPoints, location, minIndex);
    }
}
```

Αρχικά υπολογίζει τις αποστάσεις από όλα τα σημεία της `Polyline` οδηγιών σε σχέση με την θέση του πελάτη, σημαντικό σε περιπτώσεις χαμηλής ποιότητας σήματος GPS, όπως σε αστικό περιβάλλον όπου η λήψη της ακριβείς θέσης του πελάτη παρέχεται διακοπτόμενα.

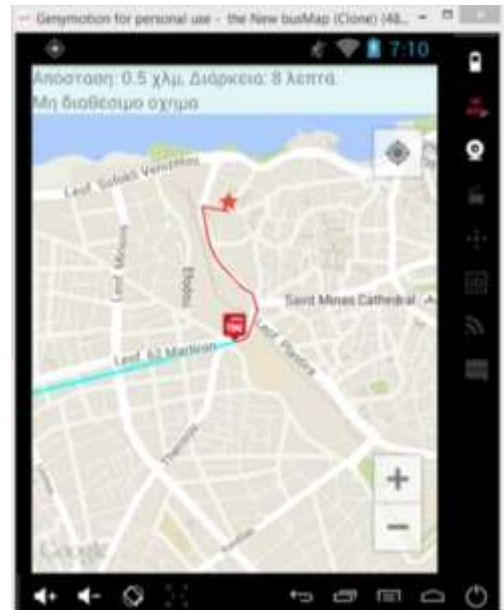
Με μια σειριακή αναζήτηση υπολογίζεται η μικρότερη απόσταση, άρα και το κοντινότερο σημείο στην `Polyline` οδηγιών, επιστρέφοντας τον δείκτη για την λίστα `remainingPoints`.

Σε περίπτωση που ο δείκτης αυτός είναι μηδέν σημαίνει ότι ο χρήστης κινείται σε αντίθετη κατεύθυνση από αυτήν που του καταδεικνύει στον χάρτη η `Polyline` οδηγιών, άρα πρέπει να προστεθεί σε αυτήν ως νέο σημείο καλώντας την μέθοδο `addPointsToPolyline()`.

Εάν ο δείκτης είναι μηδέν αλλά έχει ξεπεραστεί το δοσμένο όριο περί προσθήκης νέων σημείων σύμφωνα με την `getDistanceCounter()`, τότε θέτει την `requestDirectionalpolyline` αληθής, ώστε να ζητηθεί να επαναυπολογιστεί από το `Direction API` μια νέα διαδρομή προς την στάση, εφόσον ο πελάτης ακολουθεί λανθασμένη κατεύθυνση και ίσως υπάρχει νέα συντομότερη διαδρομή.

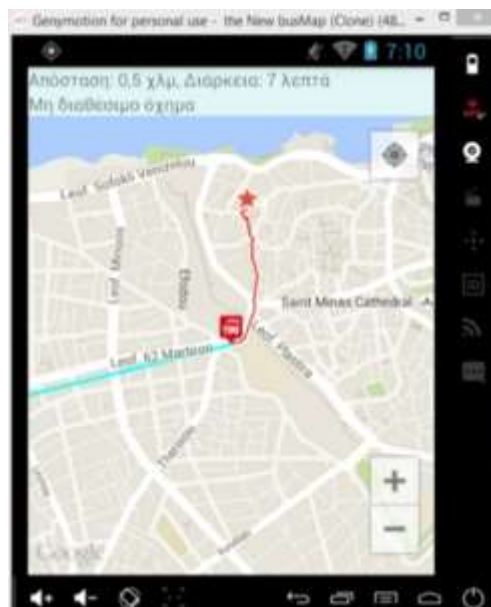


Εικόνα 6-13α: Αρχική θέση πελάτη.



Εικόνα 6-13β: Λάθος κατεύθυνση πελάτη.

Τέλος στην περίπτωση που ο δείκτης είναι μεγαλύτερος του μηδενός τότε καλείται η μέθοδος `polyPointsFromPolyline()` ώστε να αφαιρεθούν τα σημεία της Polyline οδηγίων μέχρι και τον δείκτη `minIndex` και να προστεθεί το σημείο που βρίσκεται ο πελάτης.



Εικόνα 6-13γ: Επαναυπολογισμός της Polyline μετά από αίτημα στο Directions API

6.14 popPointsFromPolyline(List<LatLng>, Location, int)

Η μέθοδος popPointsFromPolyline() έχει ως όρισμα μια λίστα με τις συντεταγμένες των σημείων της αρχικής Polyline οδηγίων, την θέση του πελάτη αλλά και τον δείκτη του σημείου αποκοπής για την λίστα.

```
private void popPointsFromPolyline(List<LatLng> initialPoints, Location location, int index){
    float totalDistance=0;
    float[] distanceArray = new float[1];
    LatLng lastpoint=new LatLng(location.getLatitude(), location.getLongitude());

    PolylineOptions polylineOptions =new PolylineOptions()
        .width(2)
        .color(Color.RED)
        .geodesic(true);

    polylineOptions.add(lastpoint);

    for(int i=index; i<initialPoints.size(); i++) {
        LatLng point = initialPoints.get(i);
        polylineOptions.add(point);
        Location.distanceBetween(lastpoint.latitude, lastpoint.longitude, point.latitude, point.longitude, distanceArray);
        totalDistance += distanceArray[0];
        lastpoint=point;
    }

    float time=((totalDistance/1000)/4)*60;
    totalDistance=(int)totalDistance/100;
    textVDD.setText("Απόσταση: "+totalDistance/10+" χλμ, Διάρκεια: "+(int)time+" λεπτά.");

    removeDirectionsPolyline();
    polylineToChange= map.addPolyline(polylineOptions);
}
```

Έχοντας αυτά τα δεδομένα ουσιαστικά δημιουργούμε μια νέα Polyline με στόχο την αντικατάσταση της παλιάς στον χάρτη. Επιλέγεται αυτή η διαδικασία διότι το Maps API v2 δεν επιτρέπει την επεξεργασία μιας Polyline μετά την δημιουργία και την τοποθέτηση της στον χάρτη σε αντίθεση με ότι ισχύει για τους Markers.

Παράλληλα με την δημιουργία σημείο προς σημείο της νέας Polyline, καλείται και η distanceBetween() της κλάσης Location για τον υπολογισμό του νέου μήκους της και την νέα εκτίμηση του χρόνου έλευσης του πελάτη, κινούμενος με τα πόδια, στην στάση εφόσον έχει αλλάξει και αυτή σε σχέση με την προηγούμενη κατάσταση.

Τέλος κατά την προσθήκη της νέας Polyline στον χάρτη ανανεώνεται και η μεταβλητή polylineToChange για να ανταποκρίνεται στην επεξεργασία της εν λόγω μεθόδου και για να είναι εφικτή η μελλοντική αλλαγή της.

6.15 addPointsToPolyline(List<LatLng>, Location)

Η μέθοδος addPointToPolyline() δέχεται ως όρισμα μια λίστα με τις συντεταγμένες των σημείων

της αρχικής Polyline οδηγιών αλλά και την θέση του πελάτη.

```
private void addPointsToPolyline(List<LatLng> initialPoints, Location location){
    float totalDistance = 0;
    float[] distanceArray = new float[1];
    LatLng lastpoint = new LatLng(location.getLatitude(), location.getLongitude());

    PolylineOptions polylineOptions = new PolylineOptions()
        .width(2)
        .color(Color.RED)
        .geodesic(true);

    polylineOptions.add(lastpoint);
    for (int i = 0; i < initialPoints.size(); i++) {
        LatLng point = initialPoints.get(i);
        polylineOptions.add(point);

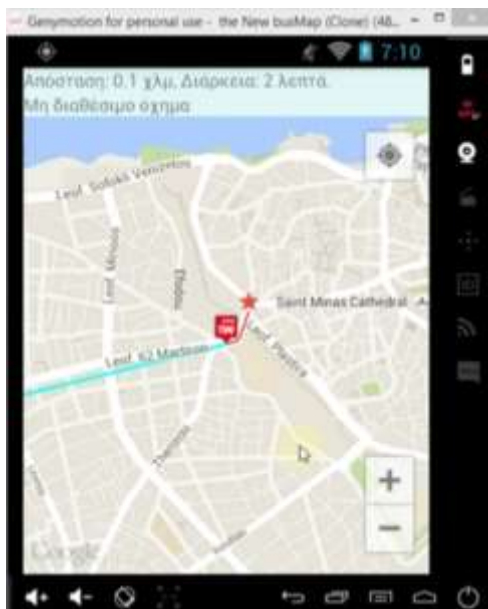
        Location.distanceBetween(lastpoint.latitude, lastpoint.longitude, point.latitude, point.longitude, distanceArray);
        totalDistance += distanceArray[0];
        lastpoint = point;
    }

    if(lastDistance<totalDistance)
        distanceCounterPlusOne();
    lastDistance=totalDistance;

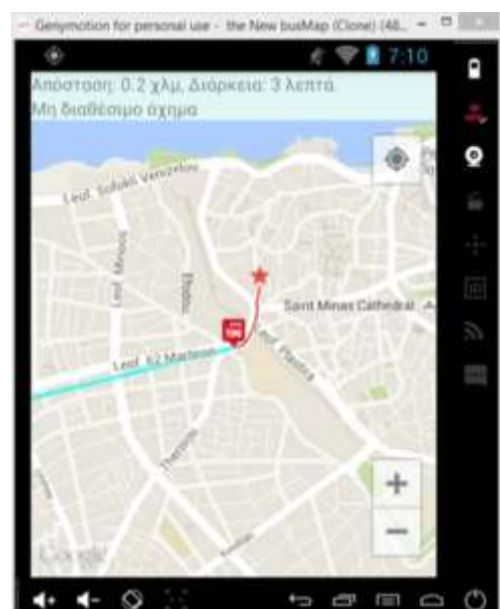
    float time = ((totalDistance / 1000) / 4) * 60;
    totalDistance = (int) totalDistance / 100;
    textVDD.setText("Απόσταση: " + totalDistance / 10 + " χλμ, Διάρκεια: " + (int) time + " λεπτά.");

    removeDirectionsPolyline();
    polylineToChange = map.addPolyline(polylineOptions);
}
```

Όπως στην περίπτωση της `polyPointsFromPolyline()` και λόγω των περιορισμών του Directions API πρέπει να αφαιρεθεί από τον χάρτη η προηγούμενη Polyline και να προστεθεί μια νέα ανανεωμένη.



Εικόνα 6-15α: Αρχική θέση πελάτη.



Εικόνα 6-15β: Λάθος κατεύθυνση του πελάτη. Προσθέτονται σημεία στην Polyline.

Παράλληλα με την εισαγωγή των ζευγών συντεταγμένων στο αντικείμενο PolylineOptions γίνεται υπολογισμός των αποστάσεων από σημείο σε σημείο ώστε να αθροιστούν στο συνολικό μήκος

της Polyline. Με αυτό το δεδομένο μπορεί να εκτιμηθεί εκ νέου ο χρόνος έλευσης του χρήστη στη στάση και να ανανεωθεί το σχετικό TextView.

Με την προσθήκη της τελικής Polyline στον χάρτη επαναπροσδιορίζεται και η μεταβλητή polylineToChange για την μελλοντική επεξεργασία της.

6.16 getTheClosestMarker(Location, ArrayList<Marker>)

Γενικά για την λήψη οδηγιών ανάμεσα σε δύο σημεία το Directions API πρέπει να έχει ως γνώση ακριβώς την θέση των σημείων αυτών. Η μέθοδος getTheClosestMarker() έχει ως όρισμα την τρέχουσα θέση του πελάτη καθώς και μια λίστα με τους Markers που αντιπροσωπεύουν τις στάσεις των λεωφορείων.

Στόχος της μεθόδου αυτής είναι η καθοδήγηση στην κοντινότερη στάση αυτόματα χωρίς την απαίτηση κάποιου είδους διάδρασης ή γνώση της πόλης από την μεριά του πελάτη. Με αυτά τα δεδομένα είναι σε θέση να επιστρέφει δείκτη για την λίστα των στάσεων ώστε να σχηματιστεί η Polyline οδηγιών.

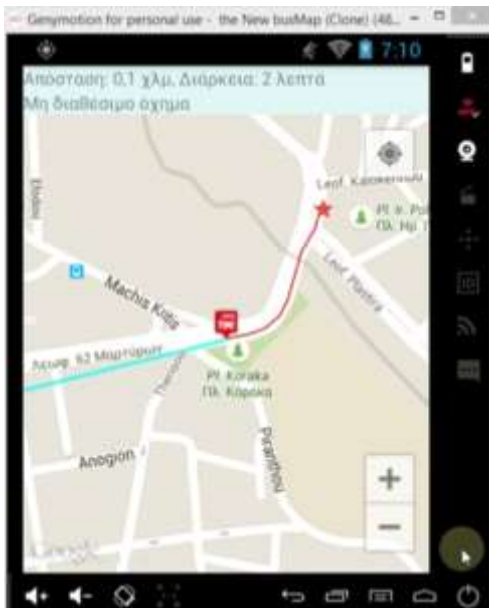
```
public int getTheClosestMarker(Location myLocation, ArrayList<Marker> addedMarkers) {
    float[] kontinoteriStasi= new float[addedMarkers.size()];
    float[] distanceArray = new float[1];
    for(int i=0; i<addedMarkers.size(); i++)
    {
        Location.distanceBetween(myLocation.getLatitude(), myLocation.getLongitude(),
            addedMarkers.get(i).getPosition().latitude, addedMarkers.get(i).getPosition().longitude, distanceArray);
        kontinoteriStasi[i]=distanceArray[0];
    }

    int minIndex = 0;
    for (int i = 1; i < kontinoteriStasi.length; i++){
        float newnumber = kontinoteriStasi[i];
        if ((newnumber < kontinoteriStasi[minIndex])){
            minIndex = i;
        }
    }

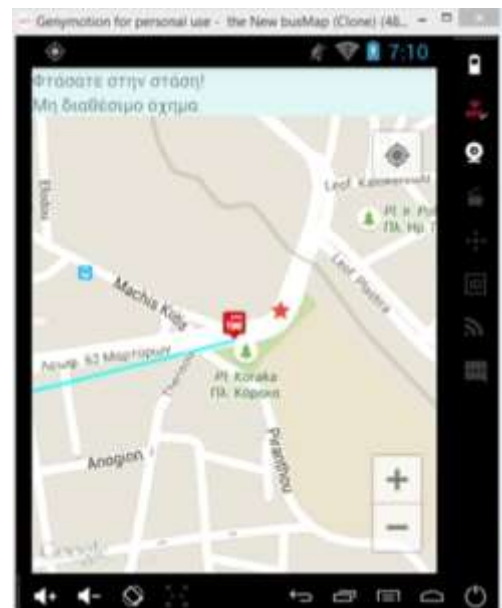
    if(kontinoteriStasi[minIndex]>50)
        return minIndex;
    else
        return -1;
}
```

Συγκεκριμένα γίνεται η χρήση της μεθόδου distanceBetween που ανήκει στην κλάση Location για τον υπολογισμό όλων των αποστάσεων από κάθε Marker της λίστας σε σχέση με την θέση του πελάτη.

Ο πίνακας float[] kontinoteriStasi περιέχει όλες τις αποστάσεις και με μια σειριακή αναζήτηση βρίσκουμε τον δείκτη της μικρότερης απόστασης, ο οποίος και επιστρέφεται από την μέθοδο εάν ο πελάτης βρίσκεται σε απόσταση μεγαλύτερη των 50 μέτρων.



Εικόνα 6-16α: Ο πελάτης βρίσκεται στα 100 μέτρα από την στάση.



Εικόνα 6-16β: Ο πελάτης έφτασε στη στάση.

Σε περίπτωση που η στάση είναι πολύ κοντά επιστρέφεται η τιμή -1 το οποίο ερμηνεύεται ως έλευση του πελάτη στη στάση, αφαιρώντας την Polyline από τον χάρτη αλλά και ανανεώνοντας το ανάλογο TextView με το μήνυμα «Φτάσατε στην στάση» όπως φαίνεται στις παραπάνω εικόνες.

6.17 findClosestBus(LatLng, ArrayList<MarkerOptions>)

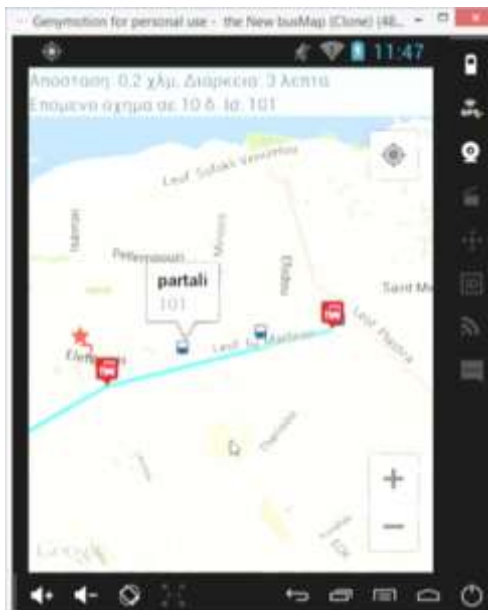
```
private movingBus findClosestBus(LatLng closestStop, ArrayList<MarkerOptions> liveBuses){
    ArrayList<movingBus> busQueue = new ArrayList<>();
    float[] distanceArray = new float[1];

    for (int i = 0; i < liveBuses.size(); i++) {
        movingBus toAddMovingBus = new movingBus();
        Location.distanceBetween(closestStop.latitude, closestStop.longitude, liveBuses.get(i).getPosition().latitude,
                                liveBuses.get(i).getPosition().longitude, distanceArray);
        toAddMovingBus.setIndex(i);
        toAddMovingBus.setDistance(distanceArray[0]);
        toAddMovingBus.setId(liveBuses.get(i).getSnippet());

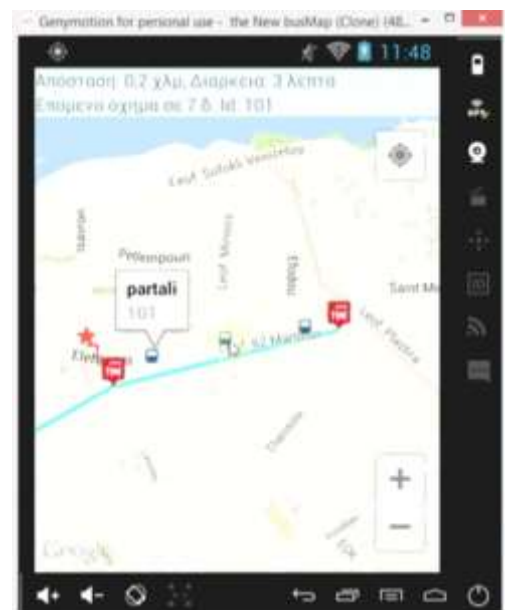
        if(toAddMovingBus.getDistance()>150 && !busesToIgnore.contains(toAddMovingBus.getId())) {
            busQueue.add(toAddMovingBus);
        }
        else if(toAddMovingBus.getDistance()<150 && !busesToIgnore.contains(toAddMovingBus.getId())){
            busesToIgnore.add(toAddMovingBus.getId());
            isItComing(-1);
        }
    }

    int minIndex = 0;
    for (int i = 1; i < busQueue.size(); i++){
        float newNumber = busQueue.get(i).getDistance();
        if ((newNumber < busQueue.get(minIndex).getDistance())){
            minIndex = i;
        }
    }
    if(busQueue.isEmpty())
        return (movingBus) null;
    else {
        return busQueue.get(minIndex);
    }
}
```

Για να είναι δυνατός ο υπολογισμός του εκτιμώμενου χρόνου έλευσης κάθε επόμενου οχήματος είναι απαραίτητο να καθορίσουμε, αρχικά, ποιο από τα διαθέσιμα οχήματα που βρίσκονται εν υπηρεσία είναι πιο κοντά στην στάση την οποία βρίσκεται ή πλησιάζει ο πελάτης.



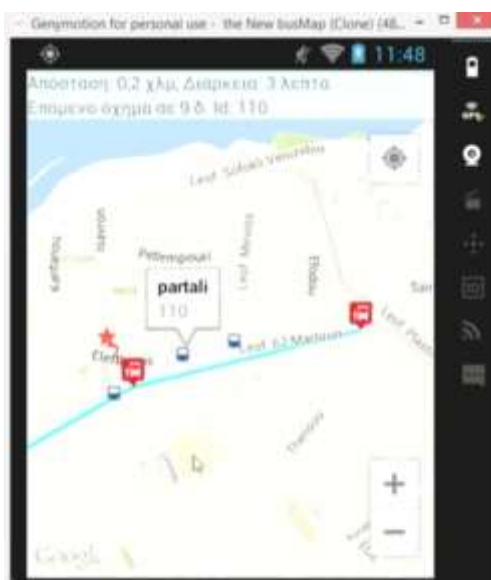
Εικόνα 6-17α: Στο επίκεντρο το όχημα 101.



Εικόνα 6-17β: το 101 πλησιάζει στη στάση.

Η μέθοδος `findClosestBus()` έχει ως όρισμα την θέση της κοντινότερης στάσης και ένα `ArrayList` με τα λεωφορεία που βρίσκονται στον δρόμο. Επίσης επιστρέφει ένα αντικείμενο `movingBus` με το αναγνωριστικό του κοντινότερου λεωφορείου και την απόσταση του από την κοντινότερη στάση.

Αρχικά εξετάζονται όλα τα στοιχεία που περιέχονται στο `ArrayList` ως προς την απόσταση από την μέθοδο `distanceBetween()` της κλάσης `Location`, παίρνει ως όρισμα 2 ζεύγη συντεταγμένων



Εικόνα 6-17γ: Αγνοείται το 101. Στο επίκεντρο πλέον το όχημα 110.

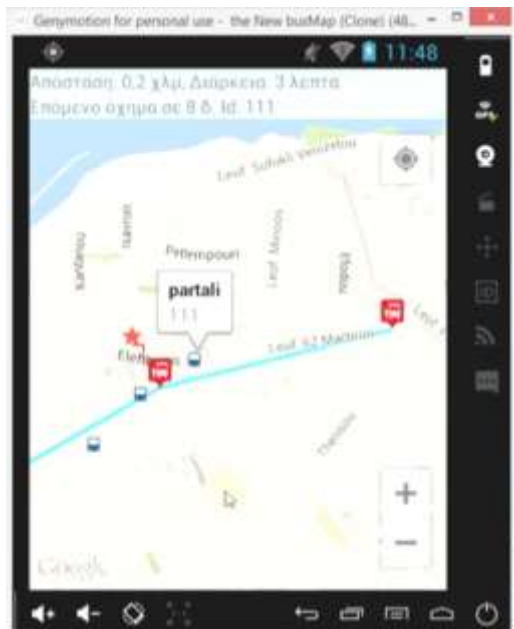


Εικόνα 6-17δ: Αισθητή βελτίωση στον ETA με χρήση δεδομένων από το προηγούμενο όχημα.

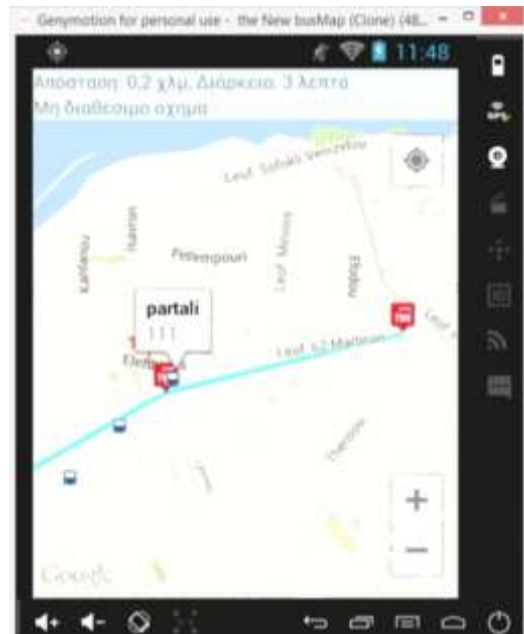
(Latitude, Longitude) και έναν float[] πίνακα όπου επιστρέφεται η απόσταση μεταξύ των δύο στοιχείων.

Αν το στοιχείο toAddMovingBus βρίσκεται κοντά στην κοντινότερη στάση τότε αυτό μπαίνει στην λίστα busToIgnore, αρχικοποιείται η μέθοδος isItComing(), ενώ εάν είναι αρκετά μακριά αλλά με κατεύθυνση την κοντινότερη στάση τότε μπαίνει στην λίστα busQueue.

Αρα μετά το πέρας της εξέτασης των στοιχείων του ArrayList<MarkerOptions> liveBuses με τις θέσεις των λεωφορείων, προκύπτουν δύο λίστες η busToIgnore, ώστε να αγνοούνται οχήματα που είναι



Εικόνα 6-17ε: Το όχημα 110 φτάνει στη στάση. Στο επίκεντρο το 111.



Εικόνα 6-17στ: Δεν υπάρχει επόμενο όχημα.

είτε μετά την κοντινότερη στάση είτε πέρασαν από την κοντινότερη στάση ενώ ήταν ήδη ανοικτή η εφαρμογή και η busQueue, για την εύρεση του κοντινότερου λεωφορείου.

Τέλος μετά από μια σειριακή αναζήτηση στην λίστα busQueue επιστρέφεται το όχημα (αντικείμενο movingBus) με την μικρότερη απόσταση από την κοντινότερη στάση, δηλαδή το κοντινότερο όχημα. Σε περίπτωση που δεν υπάρχει διαθέσιμο όχημα εκείνη την χρονική στιγμή με κατεύθυνση στην κοντινότερη στάση η μέθοδος επιστρέφει ένα κενό movingBus για την προβολή του μηνύματος στο TextView του ETA, «Μη διαθέσιμο όχημα».

6.18 calculateETA(MarkerOptions, movingBus)

Η μέθοδος calculateETA() έχει δύο ορίσματα, το κοντινότερο λεωφορείο με την μορφή MarkerOptions livebus ενσωματώνοντας πληροφορίες για την θέση του, το μοναδικό αναγνωριστικό του, τον τρόπο για την δημιουργία του Marker του όπως επίσης και το αντίστοιχο movingBus kontinoteroLeoforio το οποίο περιέχει την απόσταση του από την κοντινότερη στάση.

Σύμφωνα με την κλασική μηχανική, η ταχύτητα είναι η μετατόπιση της θέσης ως προς το χρόνο που χρειάστηκε να την διανύσει δηλαδή $v=d/t$.

Αρα μπορούμε να υπολογίσουμε την ταχύτητα σε ένα συγκεκριμένο χρονικό διάστημα χρησιμοποιώντας μια αρχική απόσταση από την στάση σε σχέση με μια δεύτερη.

Παράλληλα όμως πρέπει να διασφαλιστεί μια σχετική αξιοπιστία του παραγόμενου αποτελέσματος. Συγκεκριμένα, για αυτήν την περίπτωση το όχημα δεν κινείται συνεχώς αλλά κάνει στάσεις ή συναντά κυκλοφοριακή συμφόρηση. Επιβάλλεται λοιπόν ο προσδιορισμός μιας μέσης ταχύτητας.

```
private void calculateETA(MarkerOptions livebus, movingBus kontinoteroLeoforio){
    int eta,temp;
    int speedBus;
    float[] distanceArray = new float[1];
    try {
    if (counter == 1) {
        first =kontinoteroLeoforio.getDistance();
        isItComing(kontinoteroLeoforio.getDistance());
        counter++;
    } else if (counter == 3) {
        if(isItComing(kontinoteroLeoforio.getDistance())) {
            speedBus = ((int) (first - kontinoteroLeoforio.getDistance())) / (loopTime * counter);
            speeds.add(speedBus);

            temp = 0;
            for (int i = 0; i < speeds.size(); i++) {
                temp += speeds.get(i);
            }
            eta = (((int) kontinoteroLeoforio.getDistance())) / (temp / speeds.size());
            tvDD.setText("Επόμενο όχημα σε " + eta + " δ. Id: " + kontinoteroLeoforio.getId());
        }
        else{
            busesToIgnore.add(kontinoteroLeoforio.getId());
            tvDD.setText("Μη διαθέσιμο όχημα");
        }
        counter = 1;
    } else {
        counter++;
    }
    } catch (Exception e){Log.e("Exception eta ",e.getMessage());}
}
```

Στο `ArrayList<Integer> speeds` προστίθενται οι ταχύτητες που προκύπτουν για ένα δρομολόγιο σε κάθε κύκλο του `counter` ώστε αργότερα να μπορεί να υπολογιστεί ο χρόνος που απομένει από την έλευση ενός οχήματος σε σχέση με την εναπομένουσα απόσταση προς την μέση ταχύτητα.

Αναλυτικά, όταν ο `counter` έχει τιμή 1 καταχωρείται στην μεταβλητή `first` η αρχική απόσταση από την στάση του εξεταζόμενου οχήματος. Μετά από προκαθορισμένο χρόνο εκτέλεσης του κώδικα του αντικειμένου `requestAsyncLivebus manipulateLivebus` μπορεί να εξαχθεί ο χρόνος διάσχισης της εν λόγω απόστασης του οχήματος, άρα η ταχύτητα δίνεται από $(\text{first} - \text{kontinoteroLeoforio.getDistance()}) / (\text{loopTime} * \text{count})$.

Τέλος ο υπολογισμός του εκτιμώμενου χρόνου άφιξης είναι ο λόγος της απόστασης του οχήματος `kontinoteroLeoforio.getDistance()` προς την μέση ταχύτητα `temp/speeds.size()`.

6.19 isItComing(float)

Μια από τις προκλήσεις στον υπολογισμό της απόστασης μεταξύ κοντινότερης στάσης και των διαθέσιμων λεωφορείων είναι ο καθορισμός της κατεύθυνσης ενός οχήματος ανεξάρτητα με το πότε επιλέγει ο χρήστης να ανοίξει την εφαρμογή.

Όταν ένα όχημα απομακρύνεται από τον πελάτη τότε η μεταξύ τους απόσταση μεγαλώνει ενώ αντίθετα όταν κατευθύνεται σε αυτόν η εν λόγω απόσταση μικραίνει.

Σε αυτήν την παραδοχή βασίζεται και η μέθοδος `isItComing()` η οποία κατά την διάρκεια του προσδιορισμού της ταχύτητας ενός οχήματος ελέγχει κατά πόσο αυξάνεται ή μειώνεται η απόσταση του οχήματος με την θέση του πελάτη.

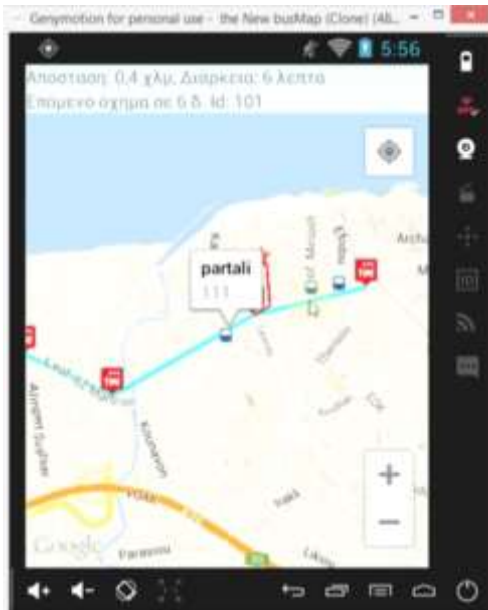
```
private boolean isItComing(float distance){
    if(initial)
        oldDistance=distance;

    if(oldDistance<distance) {
        oldDistance=0;
        initial=true;
        return false;
    }
    else if(distance==-1) {
        initial = true;
        return true;
    }
    else {
        initial=false;
        oldDistance=distance;
        return true;
    }
}
```

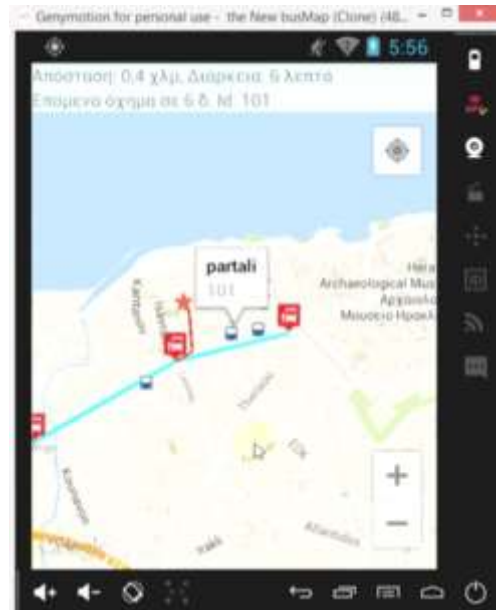
Αναλυτικά, κατά την πρώτη κλήση της `isItComing` στην `calculateETA` όταν ο `counter` είναι 1 δίνεται τιμή στην μεταβλητή `oldDistance` ενώ είναι αδιάφορο σε αυτήν τη φάση τι επιστρέφει. Σε δεύτερη φάση και όταν στην `calculateETA` ο `counter` είναι ίσος με 3 γίνεται ουσιαστικά η σύγκριση από την οποία εξαρτάται η απάντηση για το εάν απομακρύνεται ή έρχεται κάποιο όχημα στο μέρος του πελάτη. Για την αρχικοποίηση των στοιχείων της `isItComing` αρκεί να δώσουμε ως όρισμα το -1 και συνήθως όταν το όχημα αφαιρείται από την εφαρμογή λόγω έλευσης στην στάση κοντά στον χρήστη.

Με αυτήν την μέθοδο αποκλείονται όλα τα οχήματα που είναι κοντά στην στάση επιλογής του πελάτη αλλά σε τροχιά απομάκρυνσης από αυτήν. Ενδεικτικά παρουσιάζεται η περίπτωση όπου ο χρήστης ανοίγει την εφαρμογή του και από τα τρία διαθέσιμα οχήματα στο επιλεγμένο δρομολόγιο, το ένα είχε ήδη περάσει την στάση όπου βρίσκεται ενώ τα άλλα δύο κατευθύνονται προς το μέρος του.

Η εφαρμογή αγνοεί επιτυχώς το όχημα υπ' αριθμόν 101 και υπολογίζει άμεσα τον εκτιμώμενο χρόνο του 110, το οποίο είναι το όχημα που κατευθύνεται προς την στάση, παρόλο που το 101 είναι

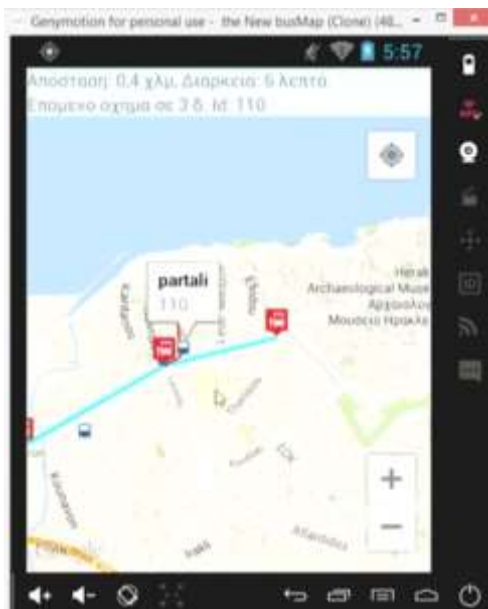


Εικόνα 6-19α: Ανοίγει ο πελάτης την εφαρμογή. Αγνοείται το όχημα 101.

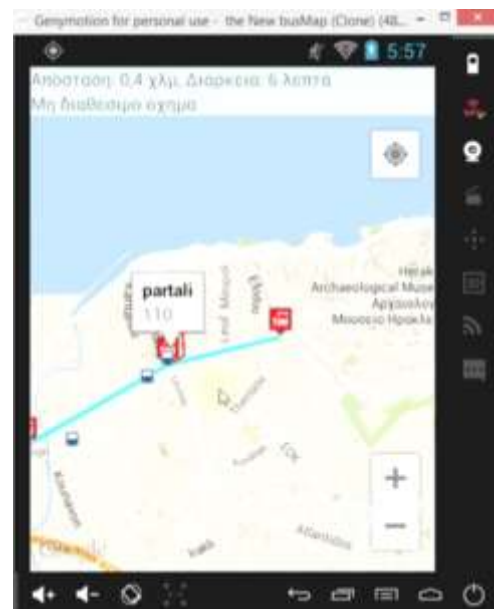


Εικόνα 6-19β: Εκτιμώμενος χρόνος του οχήματος 110.

πολύ πιο κοντά σε αυτή. Επίσης μόλις ένα όχημα πλησιάζει την στάση τότε μπαίνει στην λίστα busToIgnore εκτιμώντας τον χρόνο έλευσης του επόμενου. Ενώ όταν δεν υπάρχει άλλο διαθέσιμο όχημα εμφανίζεται το ανάλογο μήνυμα «Μη διαθέσιμο όχημα» στο TextView ETA.



Εικόνα 6-19γ: Όταν ένα όχημα φτάνει κοντά στην στάση μπαίνει στην busToIgnore.



Εικόνα 6-19δ: Δεν υπάρχει επόμενο όχημα επι του παρόντος.

7. Συμπεράσματα και Μελλοντικές Επεκτάσεις

7.1 Συμπεράσματα

Η ανάληψη της εν λόγω πτυχιακής εργασίας αποτέλεσε μια πρόκληση που δοκίμασε τις αντοχές και τις γνώσεις μου σε επίπεδο τόσο θεωρητικό όσο και προγραμματιστικό. Κατά το παρελθόν είχα ασχοληθεί εκτεταμένα με εφαρμογές server side σε PHP/MySQL/JavaScript/HTML και λιγότερο με desktop applications σε Java/C.

Η έρευνα και μελέτη που προηγήθηκε, σε θεωρητικό επίπεδο, για την ανάπτυξη, την δυναμική, την αγορά και τις τεχνολογίες γύρω από τις Location Based Services ήταν καταλυτικός παράγοντας στην εξέλιξη της υλοποίησης για την μετάδοση σε πραγματικό χρόνο των πληροφοριών θέσης των οχημάτων των μέσων μαζικής μεταφοράς.

Έχοντας έρθει σε επαφή με βιβλιογραφία σχετική με τα στάδια εξέλιξης της πληροφορικής από τα πρώτα βήματα της επιστήμης μέχρι και τις νέες τάσεις και προοπτικές, αποκόμισα μια νέα οπτική για το πώς μπορούν να αξιοποιηθούν συνδυαστικά ετερόκλητες τεχνολογίες με ένα κοινό παρονομαστή ώστε να παραχθούν καινοτόμες υπηρεσίες. Το σύνολο αυτών των υπηρεσιών θα γίνουν η βάση για τις κοινωνίες του μέλλοντος.

Συγκεκριμένα, στην υπηρεσία που κλήθηκα να δημιουργήσω, συνδύασα τεχνολογίες και βιβλιοθήκες από ένα ευρύ χάσμα. Αναλυτικά για το κομμάτι του server χρειάστηκα γλώσσες προγραμματισμού όπως PHP, HTML, JavaScript, MySQL και τεχνικές μεταφοράς δεδομένων μέσω JQuery με web services. Επίσης για την android εφαρμογή οι βιβλιοθήκες Android SDK, Google Maps API, Directions API και Apache Http Components.

Πέρα όμως από την αντικειμενική δυσκολία εκμάθησης και αξιοποίησης με τρόπο αποδοτικό όλων αυτών των τεχνολογιών και τεχνικών, αναπόφευκτη είναι η αντιμετώπιση προβλημάτων που δεν σχετίζονται απαραίτητα με συντακτικά ή λογικά σφάλματα.

Αναλυτικά, σε τέτοιου είδους υπηρεσίες πρέπει να λαμβάνεται υπόψιν και το γενικότερο πλαίσιο της ροής των δεδομένων σε σχέση με τις τεχνικές αίτησης αυτών από την βάση δεδομένων, τους διαθέσιμους διαύλους επικοινωνίας ή και τον χρόνο που απαιτείται για την ολοκλήρωση ενός τέτοιου κύκλου αιτήματος/λήψης απάντησης.

Για παράδειγμα, μια android εφαρμογή χρησιμοποιεί ένα νήμα για διάδραση με το γραφικό περιβάλλον, δίνεται όμως η δυνατότητα χρήσης ενός δευτέρου ασύγχρονου παρασκηνιακού νήματος για την επικοινωνία με τον εξυπηρετητή. Πρέπει λοιπόν να λαμβάνεται υπόψιν ο χρόνος εκτέλεσης και λήψης των αιτούμενων στοιχείων κατά την κατασκευή της εφαρμογής, διαφορετικά θα προκύπτουν προβλήματα και μη επιθυμητές παρενέργειες κατά την εκτέλεση της.

Επίσης για τόσο περίπλοκες υλοποιήσεις, θα πρέπει να προβλεφθεί ο τρόπος χρήσης τους από τον πελάτη δημιουργώντας συμπεριφορές ανάλογες των δράσεων αυτών, προλαμβάνοντας κακόβουλη ή καταχρηστική χρήση τους. Απαραίτητο λοιπόν κατά την ανάπτυξη είναι η δοκιμή όλων των πιθανών σεναρίων χρήσης και εισόδων.

Ανεξάρτητα όμως από τα παραπάνω, συχνά ήρθα αντιμέτωπος με ανυπέρβλητα εμπόδια όπως ανώριμα εργαλεία ανάπτυξης, περιορισμοί τρίτων υπηρεσιών, αδυναμίες βιβλιοθηκών ή και ακατάλληλους τρόπους επεξεργασίας δεδομένων. Σε τέτοιες περιπτώσεις είναι απαραίτητη η «out of the box» σκέψη για την εύρεση εναλλακτικών λύσεων.

Επιβεβλημένη, για παράδειγμα, είναι η επεξεργασία των Polyline οδηγιών, που παρέχεται από το Google Directions API, τοπικά στην εφαρμογή λόγω των περιορισμών χρήσης της υπηρεσίας. Αλλά και σε αυτήν την περίπτωση, το Android Maps API της 2ης έκδοσης, δε σου δίνει την δυνατότητα επεξεργασίας των στοιχείων μιας ήδη τοποθετημένης Polyline οπότε μονόδρομος είναι η δημιουργία νέας και η αφαίρεση της παλιάς από τον χάρτη.

Η αντιμετώπιση όμως όλων αυτών των δυσχερειών και προκλήσεων είχε ως αποτέλεσμα την αποκόμιση εμπειριών, ανάπτυξη δεξιοτήτων αλλά και την εμβάθυνση στις έννοιες και τεχνικές του αντικειμενοστραφούς προγραμματισμού. Πράγμα που θα με βοηθήσει στο εγγύς μέλλον κατά την

επαγγελματική μου σταδιοδρομία στον συγκεκριμένο κλάδο.

7.2 Μελλοντικές Επεκτάσεις

Η υλοποίηση όπως παρουσιάζεται, είναι σε θέση να καλύψει τις ανάγκες πόλεων μικρού έως μεσαίου μεγέθους, όπου τα κυρίαρχα μέσα μαζικής μεταφοράς είναι τα αστικά ή υπεραστικά λεωφορεία και τα ταξί. Για μεγαλύτερες πόλεις, όπου υπάρχουν διαθέσιμα μέσα σταθερής τροχιάς όπως τραμ, ηλεκτρικός ή υπόγειος σιδηρόδρομος, απαιτείται η δυνατότητα παροχής συνδυαστικών διαδρομών από όλα τα μέσα μεταφοράς, προσφέροντας την βέλτιστη δυνατή εξυπηρέτηση των πελατών.

Είναι σαφές ότι πρέπει να παρέχεται περιβάλλον χρήσης σε διάφορες γλώσσες, ώστε να επωφελούνται και οι επισκέπτες από τρίτες χώρες, σημαντικό για τις τουριστικές περιοχές όπως αυτή του Ηρακλείου. Παράλληλα βελτιώνεται το τουριστικό προϊόν μιας περιοχής αφού η πρόσβαση στα μέσα μαζικής μεταφοράς διευκολύνεται σημαντικά, επιτρέποντας τον προγραμματισμό των μετακινήσεων.

Στο ίδιο πνεύμα του προγραμματισμού, χρήσιμη θα ήταν η προσθήκη ενός δυναμικά ανανεώσιμου προγράμματος ωρών διέλευσης των οχημάτων. Ένα τέτοιο στοιχείο θα εμφάνιζε μόνο τις ώρες που έπονται και θα απέκρυπτε τις ώρες που έχουν παρέλθει για μέγιστη χρηστικότητα.

Στην παρούσα έκδοση της υλοποίησης, οι οδηγίες παρέχονται για πεζούς πελάτες, όποτε ο εκτιμώμενος χρόνος έλευσης του στην στάση είναι ανάλογος αυτού του μέσου μετακίνησης. Συγκεκριμένα, το Directions API προσφέρει διάφορες επιλογές όπως αυτοκίνητο ή ποδήλατο που θα μπορούσαν να ενσωματωθούν ως επιπλέον επιλογές μετακίνησης στην κοντινότερη στάση.

Επίσης υπάρχει η δυνατότητα εμφάνισης των οδηγιών με λεκτικό τρόπο, με μηνύματα όπως «Στρίψε δεξιά στην επόμενη οδό». Όμως λόγω του ότι και αυτά τα μηνύματα παρέχονται από το Directions API με τα λεγόμενα steps δε θα καλύπτει τις περιπτώσεις όπου ο πελάτης θα έχει πάρει λάθος δρόμο (σύμφωνα με την μέθοδο `resetDirectionPolyline`). Το παραπάνω πρόβλημα όμως θα μπορούσε να περιοριστεί μέσω του συνδρομητικού Directions API for Work, το οποίο επιτρέπει πολύ μεγαλύτερο όγκο αιτημάτων στην υπηρεσία και ανα δευτερόλεπτα αλλά και με το μέγιστο ημερήσιο επιτρεπτό όριο.

Η περιληπτική αρχιτεκτονική της εφαρμογής του πελάτη, την καθιστά ανεξάρτητη από τα δεδομένα που εμφανίζονται σε αυτήν. Έτσι χωρίς καμία αλλαγή στην υλοποίηση πάρα μόνο στην επιλεγμένη βάση δεδομένων, που αντιπροσωπεύει την κάθε πόλη, μπορεί να εξυπηρετεί διάφορες περιοχές ανάλογα την τρέχουσα θέση του χρήστη.

Βιβλιογραφία

Βιβλία:

Vermesen, Ov. & Friess, P. (2013). *Internet of Things – Converging Technologies for Smart Environment and Integrated Ecosystems*. Aalborg: Rivers Publishers

Mattern, Fr. & Floerkemeier, Chr. *From the Internet of Computers to the Internet of Things*. Zurich: Distributed Systems Group, Institute for Pervasive Computing

The Goldman Sachs Group, Inc. (2014). *The Internet of Things: Making sense of the next mega-trend*.

Evans, D. (2011). *The Internet of Things: How the Next Evolution of the Internet Is Changing Everything*. Cisco Internet Business Solutions Group (IBSG)

Hansmann, Uwe (2003). *Pervasive Computing: The Mobile World*. Springer.

Greenfield, Adam (2006). *Everyware: the dawning age of ubiquitous computing*. New Riders.

Weiser, Mark (1991). "The Computer for the 21st Century".

Weiser; Gold; Brown (1999). "Ubiquitous computing".

J. Höller, V. Tsiatsis, C. Mulligan, S. Karnouskos, S. Avesand, D. Boyle: *From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence*. Elsevier, 2014

O. Monnier (2013): *A smarter grid with the Internet of Things*. Texas Instruments.

Wigmore, I. (2014): "Internet of Things (IoT)". TechTarget

Violino, Bob (2014): "The 'Internet of things' will mean really, really big data". InfoWorld.

Hogan, Michael (2014): "The 'The Internet of Things Database' Data Management Requirements". ScaleDB

Μελέτες:

"Gartner Says the Internet of Things Installed Base Will Grow to 26 Billion Units By 2020". Gartner. 12 December 2013.

More Than 30 Billion Devices Will Wirelessly Connect to the Internet of Everything in 2020, ABI Research

Shrinath V, Yagna Teja & Jayanth Kolla (2013): *Overview of Various Location Based Services, Deployment & Adoption Trends Around the World*. Convergence Catalyst

Άρθρα:

"Assisted GPS: A Low-Infrastructure Approach". GPS World. March 1, 2002.

Watch out for data charges on a GPS phone. CNET, August 2007

Ιστοσελίδες:

http://forensicswiki.org/wiki/IP_geolocation, *Οκτώβριος 2014*

http://en.wikipedia.org/wiki/Differential_GPS, *Δεκέμβριος 2014*

http://en.wikipedia.org/wiki/Assisted_GPS, *Δεκέμβριος 2014*

<http://www.wpcentral.com/gps-vs-agps-quick-tutorial>, *Δεκέμβριος 2014*

<http://en.wikipedia.org/wiki/S-GPS>, *Δεκέμβριος 2014*

http://www.navipedia.net/index.php/Main_Page, *Ιανουάριος 2014*

http://en.wikipedia.org/wiki/Google_Maps, *Οκτώβριος 2014*

http://en.wikipedia.org/wiki/Yahoo!_Maps, *Οκτώβριος 2014*

<http://openlayers.org/>, *Οκτώβριος 2014*

<http://blogs.bing.com/maps/>, *Οκτώβριος 2014*

<http://www.gartner.com/newsroom/id/2636073>, *Νοέμβριος 2014*

<http://mobiforge.com/research-analysis/global-mobile-statistics-2014-part-a-mobile-subscribers-handset-market-share-mobile-operators>, *Νοέμβριος 2014*

<http://www.fiercewireless.com/tech/story/juniper-predicts-location-based-services-market-will-generate-433b-2019/2014-08-17>, *Ιανουάριος 2015*

<http://www.gartner.com/newsroom/id/2944819>, *Νοέμβριος 2014*

<http://www.gartner.com/newsroom/id/2665715>, *Νοέμβριος 2014*

<http://www.idc.com/prodserv/smartphone-os-market-share.jsp>, *Νοέμβριος 2014*

<http://www.gartner.com/newsroom/id/2665715>, *Νοέμβριος 2014*