

Ατομική Διπλωματική Εργασία

**“ACQUISITION AND PROCESSING OF GPS INFORMATION
USING VARIOUS TELECOMMUNICATION METHODS”**

**“ΑΝΑΚΤΗΣΗ ΚΑΙ ΕΠΕΞΕΡΓΑΣΙΑ ΠΛΗΡΟΦΟΡΙΑΣ GPS ΧΡΗΣΙΜΟΠΟΙΩΝΤΑΣ
ΔΙΑΦΟΡΕΣ ΜΕΘΟΔΟΥΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΑΣ”**

Φρίξος Σωτηράκη

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2007

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

**“ACQUISITION AND PROCESSING OF GPS INFORMATION
USING VARIOUS TELECOMMUNICATION METHODS”**

**“ΑΝΑΚΤΗΣΗ ΚΑΙ ΕΠΕΞΕΡΓΑΣΙΑ ΠΛΗΡΟΦΟΡΙΑΣ GPS ΧΡΗΣΙΜΟΠΟΙΩΝΤΑΣ
ΔΙΑΦΟΡΕΣ ΜΕΘΟΔΟΥΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΑΣ”**

Φρίξος Σωτηράκη

Επιβλέπων Καθηγητής
Δρ. Μάριος Δικαιάκος

Η Ατομική αυτή Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση
των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής
του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου

Μάιος 2007

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέπων καθηγητή αυτής της διπλωματικής εργασίας, τον Αναπληρωτή Καθηγητή κ. Μάριο Δικαιάκο, ο οποίος μου έδωσε την ευκαιρία καθώς και την παρότρυνση να υλοποιήσω την εργασία αυτή και ειδικότερα στο συγκεκριμένο θέμα, το οποίο θεωρώ αρκετά ενδιαφέρον και ανοικτό προς μελέτη και ανέλιξη σε πολλούς τομείς.

Ευχαριστίες απευθύνω και στον κ. Λάμπρο Λαμπρινό, που με κατευθυντήριες γραμμές και ορθή καθοδήγηση συνέβαλε σε μεγάλο βαθμό και σε πολλά επίπεδα να έρθει σε πέρας η εργασία αυτή.

Επίσης, θα ήθελα να ευχαριστήσω τον κ. Christof Dallermassl από το Τεχνολογικό Ινστιτούτο του Graz στην Αυστρία, επιβλέπων και συμμετέχοντα στην υλοποίηση της εφαρμογής **GPSylon**, για τις χρήσιμες οδηγίες και επίλυση αρχικών προβλημάτων κατανόησης κώδικα και δομής του λογισμικού ανοικτού κώδικα που χρησιμοποιήθηκε.

Περίληψη

Στο έγγραφο αυτό περιγράφονται και αναλύονται τα κομμάτια που αποτελούν το σύστημα που αναπτύχθηκε στα πλαίσια αυτής της Ατομικής Διπλωματικής Εργασίας. Στόχος του συστήματος είναι η παρουσίαση της γεωγραφικής τοποθεσίας συσκευών GPS σε μια εφαρμογή, μέσα από την επεξεργασία πληροφορίας που ανακτάται στα πλαίσια της επικοινωνίας των συσκευών με ένα server που τις διαχειρίζεται.

Όπως θα αναλυθεί και στις ενότητες του εγγράφου, το σύστημα αποτελείται από τρία (3) κύρια μέρη που με ποικίλους τρόπους επικοινωνίας αλληλεπιδρούν μεταξύ τους .

Το πρώτο μέρος περιέχει το GUI (Graphical User Interface) της εφαρμογής. Μέσω αυτού ο χρήστης έχει την δυνατότητα εκτέλεσης λειτουργιών διαχείρισης των συσκευών του και των χαρακτηριστικών τους. Επιπρόσθετα, μέσω της διεπιφάνειας του λογισμικού παρουσιάζονται στον χάρτη οι τελευταίες γνωστές τοποθεσίες των συσκευών και παρέχονται επιλογές διαχείρισης των τελευταίων γνωστών σημείων κάθε συσκευής. Τέλος, παρέχονται όλες οι διαθέσιμες λειτουργίες επικοινωνίας με τις συσκευές GPS.

Το δεύτερο μέρος είναι το επικοινωνιακό κομμάτι του συστήματος. Αποτελείται από ένα UDP Receiver και από τις συσκευές GPS. Αναφορικά με τον UDP Receiver, είναι υπεύθυνος για την επικοινωνία και ανταλλαγή δεδομένων μεταξύ χρηστών του συστήματος και συσκευών GPS. Καθίσταται διαμεσολαβητής και υλοποιεί με τους χρήστες καθώς και με τις συσκευές GPS το μοντέλο client-server. Αποδέχεται αιτήματα και πακέτα πληροφοριών τα οποία διαχειρίζεται και απαντά ανάλογα στον αιτούμενο (συνήθως ο χρήστης μέσω της εφαρμογής). Όσο αφορά τις συσκευές GPS, δέχονται και απαντούν αιτήματα που αφορούν τοποθεσίες και διαδρομές μέσω των πρωτοκόλλων επικοινωνίας που υποστηρίζει η κάθε μια. Η συσκευή που χρησιμοποιείται ως πρότυπο για την υλοποίηση του συστήματος υποστηρίζει τα πρωτόκολλα SMS και GPRS μέσω των οποίων γίνεται και η ανάκτηση δεδομένων εκτελώντας τις ανάλογες εντολές.

Τελευταίο κομμάτι του συστήματος είναι η επεξεργασία δεδομένων μέσω μιας ευέλικτης βάσης. Όλες οι πληροφορίες που συλλέγονται είτε από την διαχείριση των συσκευών και τις λεπτομέρειες που τις αφορούν είτε από την ανάκτηση δεδομένων από

τις συσκευές GPS αποθηκεύονται σε μια βάση δεδομένων. Έχει γίνει ένας κατάλληλος σχεδιασμός αποθήκευσης και ταξινόμησης της δομής της βάσης δεδομένων έτσι ώστε οι λειτουργίες που την επηρεάζουν να εκτελούνται σε μικρό και αποδοτικό χρόνο.

Στην συνέχεια παρατίθενται αναλυτικά τα μέρη του συστήματος και επεξηγούνται ο τρόπος σκέψης, υλοποίησης και η λειτουργία τους.

Περιεχόμενα

Κεφάλαιο 1	Εισαγωγή	1
	1.1 Τεχνολογία GPS και Συστήματα Πλοήγησης	1
	1.2 Κίνητρα Εργασίας	3
	1.3 Αρχιτεκτονική Συστήματος	4
Κεφάλαιο 2	Προεργασία	6
	2.1 Εισαγωγή	6
	2.2 Επιλογή λογισμικού	7
	2.3 Επιλογή βάσης δεδομένων	9
Κεφάλαιο 3	Εφαρμογή - Graphical User Interface	10
	3.1 Εισαγωγή	10
	3.2 Βασικές λειτουργίες	12
	3.3 Επιπρόσθετες λειτουργίες	24
	3.4 Αλλαγές στις υφιστάμενες λειτουργίες του GPSylon	28
Κεφάλαιο 4	Επεξεργασία Δεδομένων	30
	4.1 Εισαγωγή	30
	4.2 Ανάλυση Δομής Βάσης Δεδομένων	31
	4.3 Προτάσεις Δομής Βάσης Δεδομένων με εφαρμογή εφεδρικής αποθήκευσης	34
	4.4 Πίνακας τοποθεσιών	46
Κεφάλαιο 5	Επικοινωνία	47
	5.1 Εισαγωγή	47
	5.2 Μέθοδοι επικοινωνίας	48
	5.3 UDP Receiver	58
	5.4 Συσκευές GPS	65
Κεφάλαιο 6	Συμπεράσματα	67
	6.1 Μελλοντική Εργασία	67
	6.2 Τελικά Συμπεράσματα	69
Βιβλιογραφία		71

Παράρτημα Α Α-1

Παράρτημα Β Β-1

Παράρτημα Γ Γ-1

Παράρτημα Δ Δ-1

Κεφάλαιο 1

Εισαγωγή

1.1 Τεχνολογία GPS και Συστήματα Πλοήγησης	1
1.2 Κίνητρα Εργασίας	3
1.3 Αρχιτεκτονική Συστήματος	4

1.1 Τεχνολογία GPS και Συστήματα Πλοήγησης

Το GPS είναι ένα παγκόσμιο σύστημα ακριβούς προσδιορισμού τοποθεσίας και βασίζεται σε μια διάταξη 24 δορυφόρων που καλύπτουν όλη την επιφάνεια του πλανήτη.

Από το 1978 που εκτοξεύτηκε ο πρώτος πειραματικός δορυφόρος Block-I GPS έχουν γίνει τεράστια βήματα προόδου και εξέλιξης στον τομέα του Global Positioning System. Αρχικά η χρήση του προοριζόταν για συγκεκριμένους στρατιωτικούς σκοπούς – όπως και συνεχίζει να είναι στην τελική ανάλυση – αλλά βρήκε εφαρμογές σε πολλούς τομείς τους οποίους βοήθησε στην βελτιστοποίηση των λειτουργιών τους.

Η βασική αρχή λειτουργίας του GPS είναι σχετικά απλή, ωστόσο η υλοποίηση της δεν είναι καθόλου προσιτή για τον οποιοδήποτε. Όπως αναφέρθηκε, το σύστημα GPS αποτελείται από ένα δίκτυο δορυφόρων που είναι τοποθετημένοι με τέτοιο τρόπο ώστε να καλύπτουν όλη τη Γη. Οι τροχιές των δορυφόρων ελέγχονται συνεχώς έτσι ώστε όταν παρατηρηθεί κάποιο πρόβλημα να διορθώνεται άμεσα για να μην προκαλεί λανθασμένοι υπολογισμοί. Η εύρεση των γεωγραφικών συντεταγμένων, καθώς και άλλων λεπτομερειών όπως ταχύτητα και υψόμετρο, βασίζεται στον λεγόμενο

τριπλευρισμό (*trilateration*). Είναι μια μέθοδος που χρησιμοποιεί αποστάσεις και όχι γωνίες. Για να καθοριστεί η ακριβής τοποθεσία ενός αντικειμένου χρειάζονται τουλάχιστον τρία γνωστά σημεία, και από αυτά τα σημεία (δορυφόροι), υπολογίζεται η απόσταση μέσω του χρόνου που χρειάζονται τα ραδιοκύματα που εκπέμπονται να φτάσουν στον δέκτη. Μια τέτοια μέτρηση δεν είναι και τόσο απλή αφού τα ραδιοκύματα ταξιδεύουν με την ταχύτητα του φωτός.

Στην σημερινή εποχή, τα συστήματα πλοήγησης έχουν μπει για τα καλά στην ζωή μας. Παρέχονται διάφορες εφαρμογές και σε ποικίλες μορφές.

Υπάρχουν τα συστήματα πλοήγησης αυτοκινήτων που χρησιμοποιούν οθόνες που δείχνουν την κατεύθυνση που πρέπει να ακολουθήσει το αυτοκίνητο για να φτάσει στον τελικό του προορισμό, ή, στην καλύτερη των περιπτώσεων, εκφωνούν στον οδηγό τις στροφές και τους δρόμους που θα πρέπει να ακολουθήσει. Επίσης, υπάρχουν συστήματα πλοήγησης προσαρμοσμένα σε υπολογιστές παλάμης ή ακόμα και σε τηλέφωνα εξελιγμένης μορφής που προσφέρουν αναζήτηση και προσδιορισμό διαδρομών σε χάρτες επιλογής του χρήστη.

Ευρεία χρήση των πιο πάνω γίνεται σε υπηρεσίες που αφορούν εναέρια και θαλάσσια-υποθαλάσσια κυκλοφορία, και σε εταιρίες που έχουν στο δυναμικό τους στόλο από οχήματα και χρειάζονται ανά πάσα στιγμή γνώση για το που βρίσκονται τα περιουσιακά τους στοιχεία.

1.2 Κίνητρα Εργασίας

Από την πρώτη στιγμή που δόθηκε η ευκαιρία εργασίας και μελέτης σε ένα τέτοιο θέμα, έγινε αντιληπτό το εύρος και το πλαίσιο της τεχνολογίας που περιβάλλει τον χώρο του GPS.

Τα κίνητρα που προσέδωσαν ενδιαφέρον σε μια τέτοια εργασία ήταν πρωταρχικά το νηπιακό στάδιο που βρίσκεται η συγκεκριμένη τεχνολογία και οι εφαρμογές της στην Κύπρο. Την χρονική περίοδο που άρχισε η ΑΔΕ, η πλήρης χαρτογράφηση της Κύπρου δεν είχε ολοκληρωθεί και δεν υπήρχαν προγράμματα εξ' ολοκλήρου υλοποιημένα για να καλύπτουν τις ανάγκες όλου του νησιού σε πόλεις και χωριά. Μέχρι την ολοκλήρωση της εργασίας, κάποια λογισμικά έκαναν την εμφάνιση τους αφού το Τμήμα Κτηματολογίου & Χωρομετρίας του Υπουργείου Εσωτερικών χαρτογράφησε μεγάλο μέρος του οδικού δικτύου, αλλά η εργασία σίγουρα δεν έχασε κανένα ενδιαφέρον.

Ουσιαστικά από τη αρχή είχαμε υπ' όψη ότι δεν υπάρχει δυνατότητα με τις υφιστάμενες πηγές να φτιαχτεί ένα προϊόν που θα ξεπερνά τα υφιστάμενα αλλά, με την σκέψη ότι όλες οι εφαρμογές που κυκλοφορούν έχουν εισαχθεί από το εξωτερικό και δεν υπήρχε οποιοδήποτε σύστημα ολοκληρωτικά ανεπτυγμένο στην Κύπρο, σίγουρα ήταν ένα επιπλέον κίνητρο.

Επιπρόσθετο κίνητρο σίγουρα αποτελούσε ότι πρώτη φορά είχαμε την ευκαιρία να ασχοληθούμε με τον τομέα του Global Positioning System (GPS), αφού μέχρι τώρα εκτός από το φιλολογικό ενδιαφέρον και τα άρθρα στο διαδίκτυο καθώς και τις γνώσεις από διάφορες δημοσιεύσεις, δεν υπήρχε καμιά ουσιαστική επαφή με το συγκεκριμένο ερευνητικό πεδίο σε επίπεδο γνώσεων αλλά και ανάπτυξης εφαρμογής.

1.3 Αρχιτεκτονική Συστήματος

Η εφαρμογή που αναπτύχθηκε παρέχει ανάλυση, επεξεργασία και παρουσίαση πληροφορίας που ανακτάται από GPS συσκευές που υποστηρίζουν πρωτόκολλα επικοινωνίας και βασίζεται στον ανεξάρτητο (modular) σχεδιασμό και υλοποίηση.

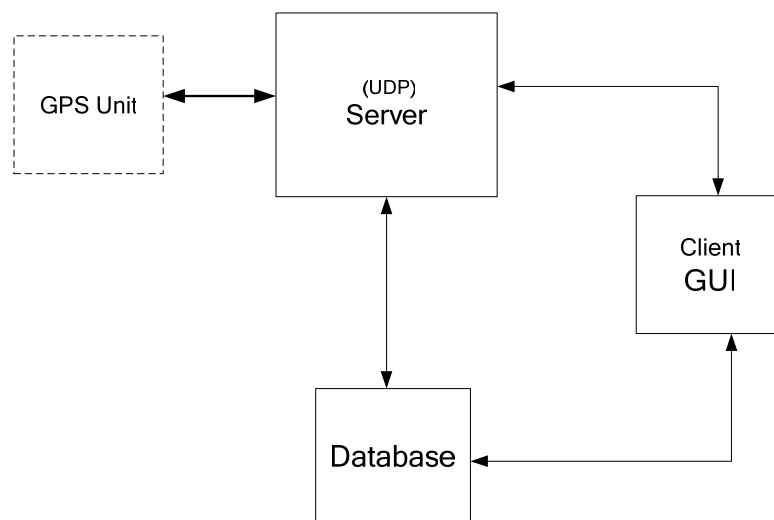
Η εφαρμογή χωρίζεται σε τρεις ανεξάρτητες ενότητες, όπως θα αναλυθούν στα επόμενα μέρη του εγγράφου:

- αλληλεπίδραση με τον χρήστη για παρουσίαση τοποθεσιών και διαχείριση συσκευών
- αποθήκευση και διαχείριση των δεδομένων και
- επικοινωνία μέσω διαφόρων μεθόδων/πρωτοκόλλων

Στόχος αρχικά του σχεδιασμού και ακολούθως της ανάπτυξης του συστήματος, είναι ο modular χειρισμός των τριών κομματιών που τον συγκροτούν. Προγραμματιστικά, όταν αυτό επιτευχθεί ο διαχειριστής του συστήματος θα μπορεί μεταβάλλοντας ένα μέρος του συστήματος, να μην επηρεαστεί η υπόλοιπη υλοποίηση και ουσιαστικά να μην χρειάζεται να γίνουν ριζικές αλλαγές για ικανοποίηση των εκάστοτε αναγκών.

Τα τρία πιο πάνω modules διαχωρίζονται με βάση την λειτουργία τους, την αλληλεπίδραση που έχουν με τον χρήστη και την μορφή επεξεργασίας της πληροφορίας που κατέχουν. Το Σχήμα 1.1 σκιαγραφεί γενικά την σύνδεση του κάθε module με το άλλο.

Σχήμα 1.1 – Σύνδεση κομματιών συστήματος



Ουσιαστικά το επικοινωνιακό κομμάτι αποτελείται από τον UDP Server (Receiver) και τις συσκευές GPS. Ο UDP Receiver είναι προγραμματισμένος να δέχεται και να “κατανοεί” πληροφορίες από τις συσκευές – που υλοποιούν ένα δικό τους πρωτόκολλο εντολών - και είναι υπεύθυνος να τις μεταφέρει για επεξεργασία παρουσίαση στον Client όταν αυτό ζητηθεί. Επίσης έχει άμεση αλληλεπίδραση με τον χρήστη της εφαρμογής (GUI/client) και σε αιτήματα και δεδομένα που αφορούν διαχείριση των συσκευών (προσθήκη, διαγραφή κ.τ.λ.).

Επεξεργασία, αποθήκευση και ανάκτηση δεδομένων γίνεται από την εφαρμογή (GUI/client) καθώς και από τον UDP Receiver. Διαχειρίζονται τους δικούς τους πίνακες στην επιλεγμένη βάση δεδομένων και χρησιμοποιούν την πληροφορία για την κάλυψη των δικών τους αναγκών ξεχωριστά.

Είναι κατανοητό ότι οι UDP Receiver και εφαρμογή (GUI/client) μπορούν να τρέξουν στους ίδιους ή και σε διαφορετικούς υπολογιστές, με την προϋπόθεση ότι δεν υπάρχουν τα γνωστά προβλήματα δικτύων (NAT, proxy servers κ.τ.λ.) που εμποδίζουν την ορθή επικοινωνία. Είναι επίσης εμφανές ότι η εφαρμογή δεν έχει άμεση επικοινωνία με τις GPS συσκευές και ο μόνος τρόπος αποστολής εντολών και ανάκτησης πληροφορίας είναι μέσω του UDP Receiver που επικοινωνεί διαμέσου πρωτοκόλλων που υποστηρίζουν οι συσκευές – GPRS και SMS.

Κεφάλαιο 2

Προεργασία

2.1 Εισαγωγή	6
2.2 Επιλογή λογισμικού	7
2.3 Επιλογή βάσης δεδομένων	9

2.1 Εισαγωγή

Σκοπός της εφαρμογής που έχει αναπτυχθεί είναι η ανάλυση, επεξεργασία και παρουσίαση πληροφορίας που ανακτάται από GPS συσκευές που υποστηρίζουν πρωτόκολλα επικοινωνίας. Πρωταρχικό μέλημα είναι η ορθή αναπαράσταση της γεωγραφικής θέσης συσκευών που διαχειρίζεται το σύστημα και η διαχείριση των χαρακτηριστικών που τις αφορούν και περιλαμβάνουν τις λεπτομέρειες κάθε συσκευής και των ανά χρονικά σημεία γεωγραφικών συντεταγμένων τους.

Για την υλοποίηση των πιο πάνω έχει γίνει μια πρωταρχική έρευνα και μελέτη που περιλαμβάνει υφιστάμενα αναπτυγμένα προγράμματα σ' αυτό το πεδίο ενδιαφέροντος, καθώς και τον τρόπο και μεθοδολογία που έπρεπε να ακολουθηθεί για την ολοκλήρωση μιας ορθής και λειτουργήσιμης εφαρμογής. Το θετικό στοιχείο ήταν ότι ερευνητικά ο τομέας που μελετείται έχει εξελιχθεί πάρα πολύ και με καλό ψάξιμο και κοσκίνισμα των προσφερόμενων επιλογών, υπήρξε μια αρκετά καλή βάση για ανάπτυξη του δικού μας συστήματος.

2.2 Επιλογή λογισμικού

Για την υλοποίηση των πιο πάνω έχει χρησιμοποιηθεί η αντικειμενοστραφής γλώσσα προγραμματισμού JAVA. Η απόφαση για την χρησιμοποίηση της JAVA πάρθηκε με γνώμονα την ευελιξία που προσφέρει στον προγραμματισμό εφαρμογών με GUI καθώς και εφαρμογών διαδικτυακής επικοινωνίας. Επιπρόσθετα, τα ήδη υπάρχοντα λογισμικά ανοικτού κώδικα τα οποία είναι διαθέσιμα βοηθούσαν άμεσα στην εκπόνηση των στόχων που είχαν τεθεί για την δική μας εφαρμογή.

Η δημιουργία από την αρχή ενός λογισμικού που θα υποστήριζε τις λειτουργίες και τα ζητούμενα ενός τέτοιου συστήματος για τα χρονικά πλαίσια μιας διπλωματικής εργασίας διάρκειας δύο ακαδημαϊκών εξαμήνων δεν θα ήταν εφικτή. Για τον λόγο αυτό, πάρθηκε απόφαση για χρήση ενός λογισμικού ανοικτού κώδικα (open source) ανεπτυγμένου σε JAVA.

Μετά από έρευνα στο διαδίκτυο και σε διάφορες πηγές όπου προσφέρεται ανοικτος κώδικας, με βάση τα χαρακτηριστικά κάθε λογισμικού που βρέθηκε και μελετήθηκε, επιλέχτηκε από μια μικρή λίστα πέντε (5) εφαρμογών το λογισμικό **GPSylon**. (<http://www.tegmento.org/gpsylon/>)

Οι εφαρμογές που απορρίφθηκαν είναι οι:

1. **ChartPlotter** - <http://www.eit.se/chartplotter/>
2. **JavaGPS** - <http://javagps.sourceforge.net/>
3. **TripTracker** - <http://triptracker.sourceforge.net/>
4. **OpenMap** - <http://openmap.bbn.com/>

Το **GPSylon** στηρίζεται πάρα πολύ στο λογισμικό **OpenMap** ως προς την μεθοδολογία παρουσίασης και επεξεργασίας των χαρτών, όμως η ειδοποιός διαφορά μεταξύ τους είναι ότι παρέχει μεγαλύτερη βάση για εξέλιξη του υφιστάμενου κώδικα για επικοινωνία με τις GPS συσκευές.

Επίσης, ο πηγαίος κώδικας του **GPSylon** είναι πολύ καλά τεκμηριωμένος και αυτοεπεξηγηματικός, παράγοντας που έπαιξε πρωτεύοντα ρόλο στην επιλογή. Επιπρόσθετα, είναι η μόνη εφαρμογή που υποστηρίζει συντεταγμένες σε όλες τις

μορφές και συγκεκριμένα στο δυαδικό σχήμα για παράδειγμα **39.392deg** το οποίο χρησιμοποιείται από τις GPS συσκευές που υποστηρίζουν πρωτόκολλα επικοινωνίας. Αντιθέτως, οι άλλες εφαρμογές υποστήριζαν συντεταγμένες μόνο στην μορφή **DD MM SS** για παράδειγμα **39 54 32 W** και θα υπήρχε ανάγκη να γίνουν τεράστιες αλλαγές για την προσαρμογή στα δικά μας δεδομένα.

Τέλος, το **GPSylon** βασισμένο στο NMEA standard, έχει ήδη ενσωματωμένο κώδικα που προσδίδει την δυνατότητα στην εφαρμογή να ανακτά δεδομένα από συσκευές συνδεδεμένες σειριακά με τον υπολογιστή, από αρχεία ή ακόμα και από GPSdaemons, φυσικά μετά από τις κατάλληλες ρυθμίσεις. Ουσιαστικά οι λειτουργίες που προσφέρονται για σειριακή σύνδεση με μια GPS συσκευή, καθώς και επεξεργασία και παρουσίαση δεδομένων δεν χρησιμοποιούνται για την κάλυψη των αναγκών του δικού μας συστήματος.

Ουσιαστικά, οι λειτουργίες που ήταν ήδη υλοποιημένες και αφορούσαν την παρουσίαση οντοτήτων (στην περίπτωση μας συσκευές) σε συγκεκριμένες γεωγραφικές συντεταγμένες στον χάρτη, καθώς και την αντιστοιχία των συντεταγμένων οθόνης σε γεωγραφικές συντεταγμένες και αντίστροφα έκανε το **GPSylon** το πιο προσιτό ως προς την ανάπτυξη των διεργασιών που σχεδιάστηκαν.

2.3 Επιλογή βάσης δεδομένων

Για τις ανάγκες αποθήκευσης, διαχείρισης και ελέγχου των δεδομένων της εφαρμογής θεωρήθηκε αναγκαία η χρήση μιας ευέλικτης βάσης δεδομένων που θα παρέχει ευκολία και ευκινησία για τις συγκεκριμένες λειτουργίες του συστήματος.

Η αρχική ιδέα ήταν η χρήση αρχείων κειμένου (text files) για την διαχείριση των δεδομένων. Τα αρχεία κειμένου χρησιμοποιούν ελάχιστο χώρο στον σκληρό δίσκο και η χρήση τους είναι απλή. Είναι συνηθισμένο στην JAVA να αντιμετωπίζουμε τα text files σαν streams. Ένα πρόγραμμα μπορεί να αρχίσει να διαβάζει ή να γράφει από/σε ένα text file σε οποιοδήποτε σημείο και μπορεί να τοποθετεί και να διαβάζει οποιοδήποτε αριθμό από bytes κάθε φορά.

Με δεδομένο την πολύπλοκη δομή και τον μεγάλο όγκο πληροφοριών που θα διαχειρίζεται και θα ανακτά-αποθηκεύει η εφαρμογή κατά την εκτέλεση της, καθώς και την συμβατότητα του open-source λογισμικού GPSylon, έγινε επιλογή χρήσης της βάσης δεδομένων **MySQL**.

Η συγκεκριμένη βάση δεδομένων επιλέχθηκε με γνώμονα την καλύτερη διασφάλιση της εγκυρότητας εκτέλεσης των λειτουργιών του συστήματος, της γρήγορης απόδοσης, της υψηλής αξιοπιστίας και της ευκολίας διαχείρισης της.

Οι βασικές λειτουργίες που θα εκτελούνται είναι η καταχώρηση, διαγραφή, αναζήτηση και ενημέρωση των πλειάδων – γεωμετρικών συντεταγμένων και άλλων λεπτομερειών που χαρακτηρίζουν την τοποθεσία των συσκευών σε κάθε χρονικό σημείο. Επίσης υπάρχει η ανάγκη υποστήριξης stored procedures καθώς σε καθημερινή βάση θα εκτελούνται μέσω του λογισμικού λειτουργίες backup και ουσιαστικά ομαδοποίησης των δεδομένων που θα έχει συλλέξει η βάση μέχρι την χρονική εκείνη στιγμή.

Ο κύριος λόγος που έγινε η επιλογή της MySQL είναι ότι το λογισμικό GPSylon που χρησιμοποιείται, περιέχει ήδη επιλογή χρήσης της συγκεκριμένης βάσης δεδομένων, με την προϋπόθεση ότι θα γίνουν κάποιες συγκεκριμένες αλλαγές σε κομμάτια του υφιστάμενου κώδικα. Το λογισμικό ανοικτού κώδικα περιέχει επίσης επιλογές χρήσης των Hypersonic DB (hsqldb) και PostgreSQL, αλλά μετά από μελέτη των χαρακτηριστικών των δύο αυτών επιπρόσθετων επιλογών, κρίθηκε ότι η MySQL υπερτερεί και πληρεί τα κριτήρια χρήσης της για την εφαρμογή μας.

Κεφάλαιο 3

Εφαρμογή - Graphical User Interface

3.1 Εισαγωγή	10
3.2 Βασικές λειτουργίες	12
3.3 Επιπρόσθετες λειτουργίες	24
3.4 Αλλαγές στις υφιστάμενες λειτουργίες του GPSylon	28

3.1 Εισαγωγή

Αφού έγινε η επιλογή λογισμικού, το επόμενο στάδιο ήταν οι αλλαγές που απαιτούνται ώστε να υλοποιηθούν οι απαραίτητες λειτουργίες. Πρωταρχικός στόχος ήταν η όσο πιο δυνατό καλύτερη αναπαράσταση των υφιστάμενων καταστάσεων των διαχειριζομένων συσκευών όσο αφορά τα χαρακτηριστικά τους (π.χ. αναγνωριστικό συσκευής, τύπος συσκευής κ.τ.λ.) καθώς και των ανά πάσα χρονική στιγμή τοποθεσιών τους.

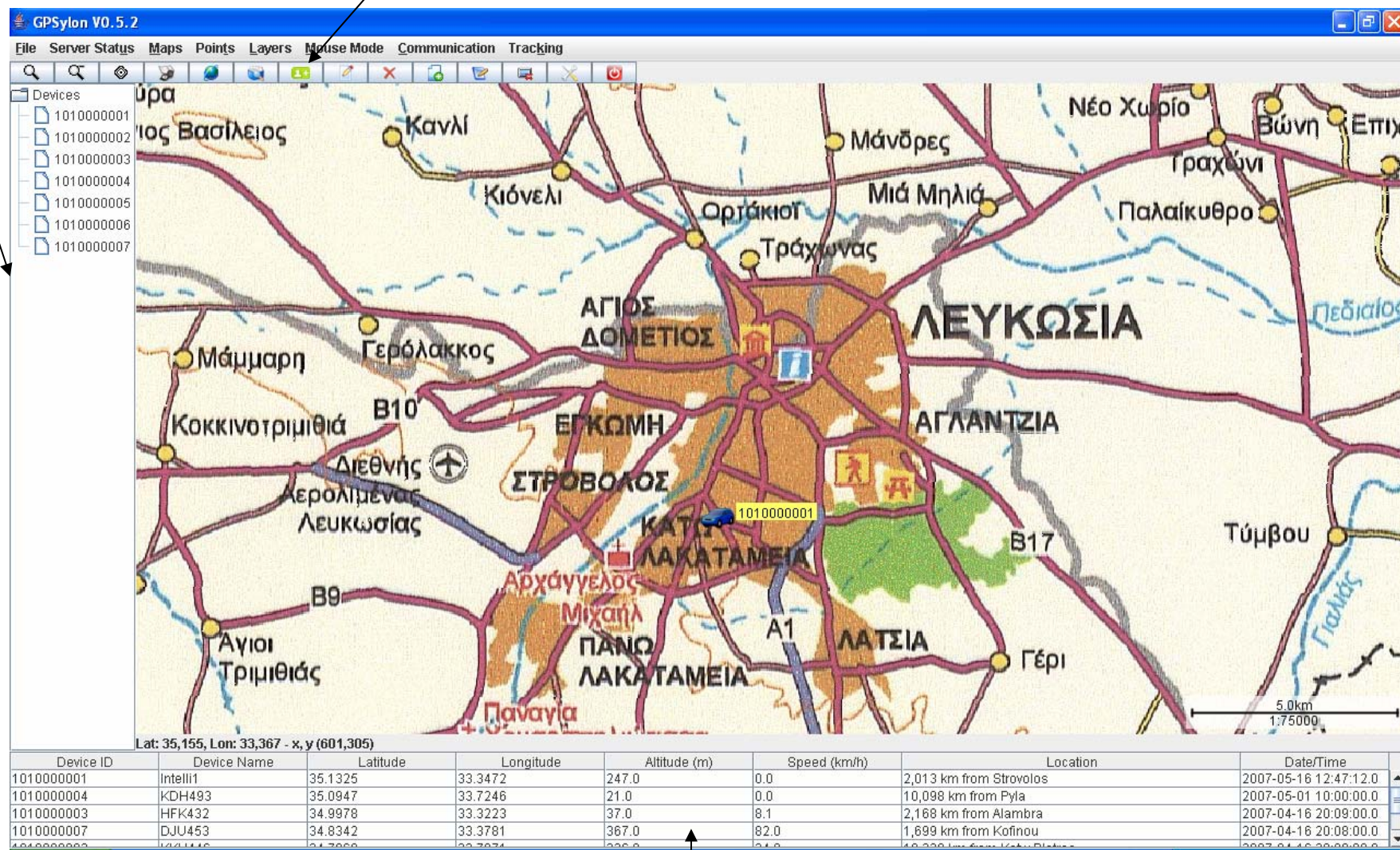
Σε γενικές γραμμές, έχουν αναπτυχθεί λειτουργίες που διευκολύνουν τον χρήστη στην διαχείριση των λεπτομερειών των συσκευών. Οι λεπτομέρειες αφορούν τις ίδιες τις διαχειριζόμενες συσκευές, οι οποίες παρατίθενται σε μια λίστα στο αριστερό μέρος της εφαρμογής, όπως και τα πεδία που παρέχουν την τελευταία γνωστή τοποθεσία κάθε συσκευής και παρουσιάζονται σε ένα πίνακα στο κάτω μέρος της εφαρμογής.

Παρακάτω παρουσιάζεται μια γενική άποψη της διεπιφάνειας της εφαρμογής (Εικόνα 3.1) και ακολούθως αναλύονται οι βασικές λειτουργίες που έχουν υλοποιηθεί καθώς και αλλαγές που έγιναν στο υφιστάμενο λογισμικό για να καλύψουν τις ανάγκες της δικής μας εφαρμογής. Οι λειτουργίες που υλοποιήθηκαν και αφορούν την εφαρμογή αναγράφονται κατά κλάση στο Παράρτημα Α.

Πλαίσιο πιο συχνά χρησιμοποιούμενων λειτουργιών

Διαχειριζόμενες Συσκευές

Εικόνα 3.1 – Διεπιφάνεια εφαρμογής



Πίνακας καταστάσεων τελευταίων γνωστών σημείων κάθε συσκευής

3.2 Βασικές λειτουργίες

Στην ενότητα αυτή αναλύονται οι βασικές λειτουργίες της εφαρμογής όσο αφορά την παρουσίαση και διαχείριση των συσκευών καθώς και άλλες επιπρόσθετες λειτουργίες.

3.2.1 Διαχείριση χαρτών

Μετά από την αρχική μελέτη του πηγαίου κώδικα για την επίτευξη απόκτησης λεπτομερούς γνώσης του λογισμικού, η πρώτη μετατροπή που έγινε ήταν η παρουσίαση του χάρτη της Κύπρου στο κέντρο της διεπιφάνειας. Ορίστηκε το όνομα, η κλίμακα, το μέγεθος (σε pixels) καθώς και οι συντεταγμένες του κέντρου της νέας εικόνας και αυτή καθορίστηκε να είναι η αρχική εικόνα του χάρτη. Το πιο μεγάλο εμπόδιο ήταν η εύρεση του κέντρου του νέου χάρτη που εισάχθηκε, έτσι ώστε στον χάρτη να υπάρχει σωστή αντιστοίχιση συντεταγμένων οθόνης με τις γεωγραφικές συντεταγμένες, αλλά με αρκετή προσπάθεια και πειραματισμό κατέληξε στο πιο βέλτιστο σημείο. Για την πιο ορθή τοποθέτηση του χάρτη στην οθόνη, έχουν γίνει δειγματοληπτικές μετρήσεις μέσω της συσκευής IntelliTrac X8 καθώς και της συσκευής MIO A701 που ήταν διαθέσιμες για αυτή την διπλωματική εργασία, σε διάφορα μέρη της Κύπρου (αυτοκινητόδρομος Λευκωσίας-Λάρνακας και Λευκωσίας-Λεμεσού, κυκλικοί κόμβοι Λεμεσού, κεντρικοί δρόμοι Λευκωσίας).

Στο σύστημα έχουν καταχωρηθεί πέντε (5) χάρτες σε δοκιμαστική βάση για τον έλεγχο ουσιαστικά της λειτουργίας εμφάνισης χαρτών σε διαφορετικές κλίμακες, ανάλογα με το επίπεδο εστίασης. Συγκεκριμένα, έχουν προστεθεί οι οδικοί χάρτες των πόλεων Λευκωσία, Λάρνακα, Λεμεσού, Πάφου και Αγίας Νάπας. Η εφαρμογή διαβάζει τα στοιχεία κάθε εικόνας από ένα αρχείο κειμένου (maps.txt) που αναγράφει εκτός από το όνομα της πόλης, το κέντρο, την κλίμακα και το μέγεθος του χάρτη και ακολούθως τους παρουσιάζει στην διεπιφάνεια όταν το επίπεδο εστίασης και η τοποθεσία πλοήγησης στον χάρτη είναι στα πλαίσια μεγέθους της εκάστοτε εικόνας.

Είναι προφανές ότι η εφαρμογή δεν πρέπει να περιορίζεται στους χάρτες που έχουμε εμείς συλλέξει και εισάγει. Με αυτό το σκεπτικό, έχει δημιουργηθεί η λειτουργία προσθήκης νέου χάρτη.

Ο νέος χάρτης πρέπει να έχει την μορφή *jpg* ή *gif* και υπάρχει δυνατότητα επιλογής του (ως εικόνα) από οποιοδήποτε αρχείο περιέχεται στον υπολογιστή του χρήστη. Μετά την επιλογή του νέου χάρτη, ζητούνται από τον χρήστη τα ακόλουθα:

- όνομα αρχείου εικόνας για μετατροπή σε χάρτη
- γεωγραφικό μήκος κέντρου του χάρτη
- γεωγραφικό πλάτος κέντρου του χάρτη
- κλίμακα
- πλάτος εικόνας
- ύψος εικόνας

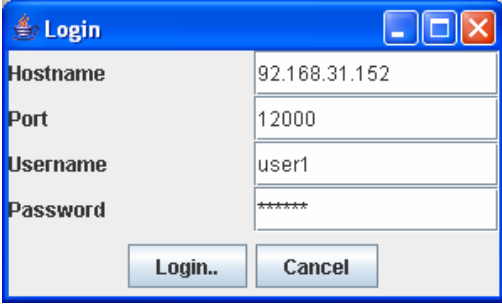
Καταχωρώντας τις πιο πάνω πληροφορίες έτσι ώστε ο χάρτης να παρουσιάζεται στο κατάλληλο επίπεδο εστίασης και στην ορθή τοποθεσία, η εικόνα αντιγράφεται στο home directory της εφαρμογής και παράλληλα δημιουργείται μια νέα γραμμή στο αρχείο κειμένου “maps.txt” με αναγραφμένα τα στοιχεία αυτά. Από την στιγμή που εκτελεστούν τα δύο προαναφερθέντα βήματα, έχουν αντιστοιχιστεί οι συντεταγμένες χάρτη με τις συντεταγμένες οθόνης και ο χάρτης εμφανίζεται σε κάθε μεταφορά της αναζήτησης σε κλίμακα και συντεταγμένες που βρίσκονται ανάλογα στα δικά του πλαίσια.

3.2.2 Σύνδεση με τον UDP Receiver

Για να έχει ο χρήστης την δυνατότητα να διαχειριστεί και να παρακολουθήσει τις συσκευές που είναι στην δικαιοδοσία του, πρέπει αναγκαστικά να έχει πρόσβαση στον UDP Receiver (μοντέλο client-server).

Με την εκκίνηση της εφαρμογής, ο χρήστης υποχρεούται να συνδεθεί με τον UDP Receiver (Εικόνα 3.2), εισάγοντας τις σχετικές λεπτομέρειες που αφορούν την διεύθυνση IP και την θύρα (port) σύνδεσης, καθώς και τα συνθηματικά πρόσβασης (username και password), ο χρήστης έχει πλέον πρόσβαση στις συσκευές που διαχειρίζεται.

Εικόνα 3.2 – Σύνδεση με UDP Receiver



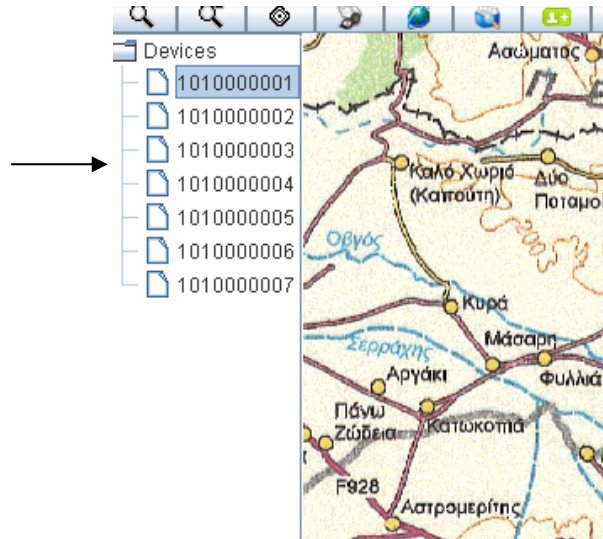
Ο λόγος που θεωρήθηκε σωστό να γίνεται πάντα η διαδικασία αυτή είναι για λόγους ασφαλείας. Ο χρήστης μπορεί να διαχειριστεί ΜΟΝΟ τις συσκευές που έχει πρόσβαση και όχι όλες τις συσκευές που περιέχονται στις αποθηκευμένες πληροφορίες του UDP Receiver. Επιπρόσθετα, κατά την διάρκεια της λειτουργίας της εφαρμογής, αν για κάποιο λόγο θεμιτό ή αθέμιτο η εφαρμογή αποσυνδεθεί από τον UDP Receiver, σταματούν να είναι επιτρεπτές οποιεσδήποτε λειτουργίες πλην της επανασύνδεσης και της πλοήγησης στον χάρτη.

Να συμπληρωθεί στο σημείο αυτό ότι η κάθε εφαρμογή είναι αναπτυγμένη για χρήση μόνο από ένα χρήστη. Αν για παράδειγμα συνδεθεί με τον UDP Receiver κάποιος χρήστης κατά την εκκίνηση της εφαρμογής και σε κάποια φάση αποσυνδεθεί, η εφαρμογή σε περίπτωση επανασύνδεσης δεν παρέχει πρόσβαση σε άλλο χρήστη παρά μόνο στον χρήστη που είχε αρχικά πρόσβαση. Στην περίπτωση που άλλος χρήστης θέλει να χρησιμοποιήσει την εφαρμογή για διαχείριση των δικών του συσκευών επιβάλλεται να την επανεκκινήσει.

3.2.3 Διαχείριση Συσκευών

Με την εκκίνηση της εφαρμογής δημιουργείται μια λίστα με τις συσκευές τις οποίες διαχειρίζεται ο χρήστης ο οποίος έχει πρόσβαση στο σύστημα και έχει συνδεθεί με τον UDP Receiver (Εικόνα 3.3).

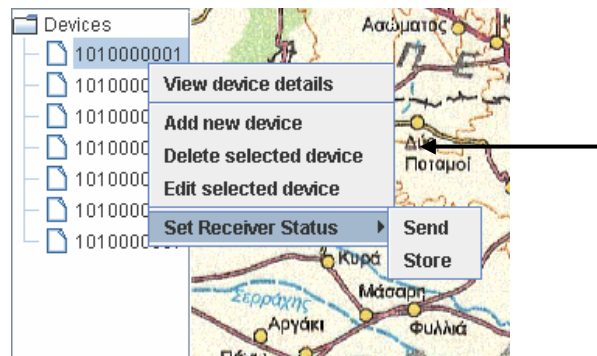
Εικόνα 3.3 – Λίστα διαχειριζομένων συσκευών



Η λίστα τοποθετείται στο αριστερό μέρος της και είναι διαθέσιμες οι πιο κάτω επιλογές (μέσω του δεξιού κουμπιού του ποντικιού):

- Καταχώρηση νέων συσκευών
- Διαγραφή Συσκευής
- Ενημέρωση Στοιχείων
- Παρουσίαση Στοιχείων Συσκευής
- Καθορισμός Κατάστασης Λήψης Δεδομένων

Εικόνα 3.4 – Διαθέσιμες επιλογές διαχείρισης συσκευών



Παρατίθεται παρακάτω οι κύριες λειτουργίες διαχείρισης συσκευών.

3.2.3.1 Καταχώρηση νέων συσκευών

Με την επιλογή καταχώρησης νέας συσκευής (Add new device) παρουσιάζεται το πλαίσιο εισαγωγής χαρακτηριστικών της συσκευής. Τα χαρακτηριστικά είναι:

- *Device ID*: Μοναδικό αναγνωριστικό συσκευής
- *Device Name*: Όνομα συσκευής (π.χ. πινακίδα αυτοκινήτου)
- *Driver*: Οδηγός
- *Type*: Τύπος συσκευής (π.χ. αυτοκινήτου, βάρκας κ.τ.λ.)
- *Owner ID*: Διαχειριστής συσκευής
- *Password*: Συνθηματικό διαχειριστή συσκευής

Εικόνα 3.5– Καταχώρηση νέας συσκευής



Field	Value
Device ID	1010000009
Device Name	Intel1
Driver	John K.
Type	Car
Owner ID	Frixos F.
Password	*****

Όταν ο χρήστης επιπλέον από τα χαρακτηριστικά της συσκευής για λόγους ασφάλειας εισάγει και το ορθό συνθηματικό σύνδεσης με τον UDP Receiver η συσκευή καταχωρείται στην βάση δεδομένων, προστίθεται στην λίστα συσκευών και καταχωρείται επίσης στον πίνακα κατάστασης χρηστών/συσκευών του UDP Receiver.

3.2.3.2 Διαγραφή Συσκευής

Με την επιλογή διαγραφής επιλεγμένης συσκευής (Delete selected device) ο χρήστης διαγράφει από το σύστημα, την βάση δεδομένων και τον πίνακα κατάστασης χρηστών/συσκευών του UDP Receiver την επιλεγμένη συσκευή.

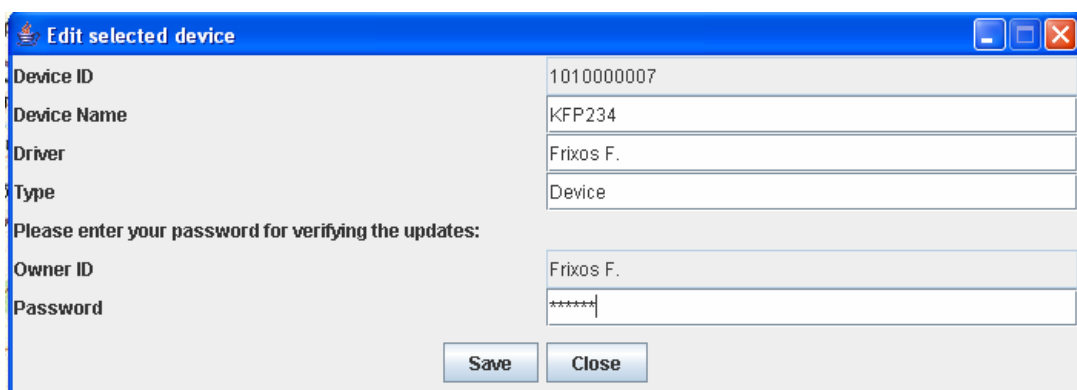
Ο χρήστης επίσης ερωτάται αν θέλει να διαγραφούν όλες οι καταχωρήσεις που αφορούν την συγκεκριμένη συσκευή. Αν ο χρήστης απαντήσει καταφατικά ο εφεδρικός πίνακας που υπάρχει στην βάση δεδομένων και έχει αποθηκευμένες όλες τις παλιές καταχωρήσεις θα διαγραφεί και ο χρήστης δεν θα έχει πια πρόσβαση για ανάκτηση ιστορικού δεδομένων για την συσκευή.

Με την ολοκλήρωση της διαδικασίας το τελευταίο γνωστό σημείο στον Πίνακα Λεπτομερειών Τελευταίου Γνωστού Σημείου καθώς και η παρουσίαση της συσκευής στον χάρτη διαγράφονται.

3.2.3.3 Ενημέρωση Στοιχείων

Με την διεργασία ενημέρωσης στοιχείων επιλεγμένης συσκευής (Edit selected device) ο χρήστης ενημερώνει τον UDP Receiver και την βάση δεδομένων με νέα (επιτρεπόμενα) χαρακτηριστικά για την επιλεγμένη συσκευή.

Εικόνα 3.6 – Ενημέρωση Συσκευής



The screenshot shows a window titled "Edit selected device" with a blue header bar. Inside the window, there are several input fields and buttons. The fields are labeled "Device ID", "Device Name", "Driver", "Type", "Please enter your password for verifying the updates:", "Owner ID", and "Password". The "Device ID" field contains the value "1010000007". The "Device Name" field contains "KFP234". The "Driver" field contains "Frixos F.". The "Type" field contains "Device". The "Owner ID" field contains "Frixos F.". The "Password" field contains "*****". At the bottom of the window, there are two buttons: "Save" and "Close".

Device ID	1010000007
Device Name	KFP234
Driver	Frixos F.
Type	Device
Please enter your password for verifying the updates:	
Owner ID	Frixos F.
Password	*****

Save Close

Ο χρήστης έχει την δυνατότητα αλλαγής των:

- *Device Name*: Όνομα συσκευής (π.χ. πινακίδα αυτοκινήτου)
- *Driver*: Οδηγός
- *Type*: Τύπος συσκευής (π.χ. αυτοκινήτου κ.τ.λ.)

Τα *Device ID* (Μοναδικό αναγνωριστικό συσκευής) και *Owner ID* (Διαχειριστής συσκευής) δεν επιδέχονται αλλαγές αφού αποτελούν μοναδικά στοιχεία που χαρακτηρίζουν την συσκευή.

Για να καταχωρηθούν οι αλλαγές και για λόγους ασφάλειας ο χρήστης υποχρεούται να εισάγει το ορθό συνθηματικό σύνδεσης με τον UDP Receiver.

3.2.3.4 Παρουσίαση Στοιχείων Συσκευής

Με την διεργασία παρουσίασης στοιχείων επιλεγμένης συσκευής (View device details) ο χρήστης ανακτά από τον UDP Receiver (αν και εφόσον έχει πρόσβαση στην συγκεκριμένη συσκευή) τα χαρακτηριστικά της επιλεγμένης συσκευής.

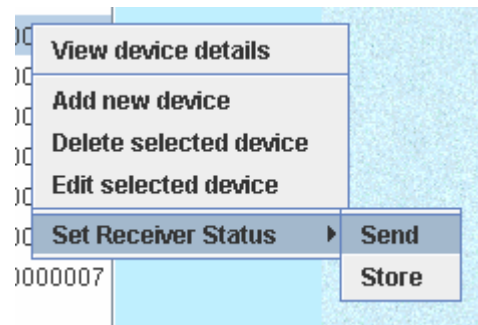
Εικόνα 3.7 – Παρουσίαση λεπτομερειών συσκευής



3.2.3.5 Καθορισμός Κατάστασης Λήψης Δεδομένων

Με την λειτουργία καθορισμού κατάστασης λήψης δεδομένων για μια συσκευή (Set Receiver Status), αποστέλλεται η πληροφορία στον UDP Receiver για το πώς θα χειρίζεται τα δεδομένα που εισέρχονται για την επιλεγμένη.

Εικόνα 3.8 – Καθορισμός κατάστασης συσκευής



Συγκεκριμένα υπάρχουν δύο επιλογές:

- **Send**

Ο UDP Receiver μόλις λάβει νέα δεδομένα-πληροφορίες που αφορούν την συσκευή τα αποστέλλει αμέσως στην αντίστοιχη εφαρμογή-χρήστη για ανάλυση, επεξεργασία και παρουσίαση στον χάρτη

- **Store**

Ο UDP Receiver μόλις λάβει νέα GPS δεδομένα που αφορούν την συσκευή τα αποθηκεύει στην βάση δεδομένων και η εφαρμογή τα ανακτά μόνο όταν ο χρήστης το καθορίσει.

Η κατάσταση της συσκευής αποθηκεύεται στον πίνακα διαχείρισης χρηστών/συσκευών που διατηρεί ο UDP Receiver (Περιγραφή λειτουργίας UDP Receiver στο Κεφάλαιο 5).

3.2.4 Διαχείριση Γεωγραφικών τοποθεσιών συσκευών

Όπως προαναφέρθηκε, με την εκκίνηση της εφαρμογής δημιουργείται ένας πίνακας με τα τελευταία γνωστά σημεία για τις συσκευές τις οποίες διαχειρίζεται ο χρήστης ο οποίος έχει πρόσβαση στο σύστημα (Εικόνα 3.8). Ο πίνακας τοποθετείται στο κάτω μέρος της εφαρμογής .

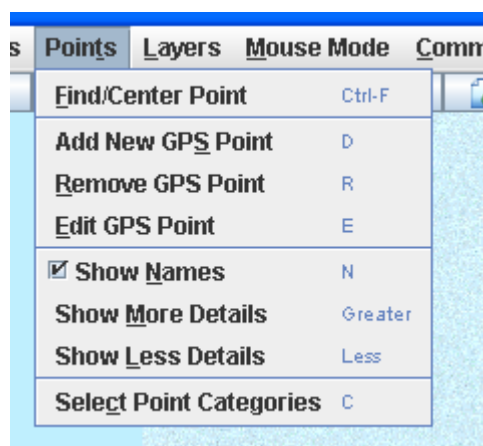
Εικόνα 3.8 – Πίνακας τελευταίων γνωστών τοποθεσιών

Lat: 34,597, Lon: 34,274 - x, y (983,578)							
Device ID	Device Name	Latitude	Longitude	Altitude (m)	Speed (km/h)	Location	Date/Time
1010000001	Intelit	35.1325	33.3472	247.0	0.0	2,013 km from Strovilos	2007-05-16 12:...
1010000004	KDH493	35.0947	33.7246	21.0	0.0	10,099 km from Pyla	2007-05-01 10:...
1010000003	HFK432	34.9978	33.3223	37.0	8.1	2,168 km from Alambra	2007-04-16 20:...
1010000007	DJU453	34.8342	33.3781	367.0	82.0	1,699 km from Kofinou	2007-04-16 20:...
1010000000	Intelit	34.3000	33.3000	300.0	0.0	10,000 km from Pyla	2007-04-16 20:...

Οι λειτουργίες που έχουν υλοποιηθεί και αφορά την διαχείριση γεωγραφικών τοποθεσιών των συσκευών είναι:

- Καταχώρηση νέων σημείων
- Διαγραφή σημείου
- Ενημέρωση σημείου
- Εύρεση σημείου

Εικόνα 3.9 – Μενού διαχείρισης πίνακα τελευταίων γνωστών σημείων



3.2.4.1 Καταχώρηση νέων σημείων

Η λειτουργία αυτή (Add New GPS Point) δίνει την δυνατότητα στον χρήστη να καταχωρήσει manually νέα τοποθεσία μιας συσκευής στον χάρτη (Εικόνα 3.10).

Τα χαρακτηριστικά του νέου σημείου της συσκευής που θα καταχωρηθεί είναι:

- *Device ID*: Μοναδικό αναγνωριστικό συσκευής
- *Device Name*: Όνομα συσκευής (π.χ. πινακίδα αυτοκινήτου)
- *Category*: Κατηγορία συσκευής (π.χ. αυτοκίνητο κ.τ.λ.)
- *Latitude*: Γεωγραφικό μήκος σημείου
- *Longitude*: Γεωγραφικό πλάτος σημείου
- *Altitude*: Υψόμετρο
- *Location*: Auto/Unknown – Αν η επιλογή είναι “Auto” τότε στην καταχώρηση της τοποθεσίας σημείου θα γίνει αλγοριθμικός υπολογισμός ανάλογα με γεωγραφικές συντεταγμένες και στο πεδίο αυτό θα αναγραφεί η απόσταση από την πιο κοντινή γνωστή περιοχή. Αν η επιλογή είναι “Unknown” η τιμή του πεδίου τοποθεσίας θα παραμείνει κενή
- *Speed*: Ταχύτητα συσκευής
- *Date*: Ημερομηνία λήψης δεδομένων

Εικόνα 3.10 – Καταχώρηση νέου σημείου

Device ID	1010000001
Name	Intelli1
Category	Car GPS Device
Latitude	35,12920°N
Longitude	33,42450°E
Altitude(m)	234
Location	Auto
Speed(km/h)	12.7
Date(YYYY-MM-DD hh:mm:ss)	2007-01-01 00:00:00

OK Close

Click into the map to change the position.

Ο χρήστης έχει την δυνατότητα να επιλέξει την συσκευή που θα καταχωρήσει νέο σημείο από μια λίστα που του παρουσιάζεται και αυτόματα μετά την επιλογή παρουσιάζεται το όνομα της. Τονίζεται ότι οι συσκευές που παρουσιάζονται και για τις οποίες μπορούν να ανανεωθούν – καταχωρηθούν νέα σημεία αφορούν μόνο συσκευές που είναι στην δικαιοδοσία του χρήστη.

Η επιλογή του γεωγραφικού μήκους και πλάτους γίνεται με το ποντίκι επιλέγοντας την επιθυμητή τοποθεσία στον χάρτη ή αναγράφοντας στην σωστή μορφή στο κατάλληλο πεδίο τις συντεταγμένες. Αν τα εισαγμένα δεδομένα είναι ορθά, ανανεώνεται η βάση δεδομένων, ο χάρτης και ο πίνακας κατάστασης τελευταίων γνωστών σημείων με το εισαχθέν σημείο ως το τελευταίο γνωστό σημείο της συγκεκριμένης συσκευής.

Επιπρόσθετα η ίδια λειτουργία καλείται όταν ανακτηθούν πληροφορίες τοποθεσίας από μια συσκευή, για παρουσίαση στον χάρτη και αποθήκευση στην βάση δεδομένων. Άρα ουσιαστικός σκοπός της διεργασίας αυτής είναι η καταχώρηση νέων δεδομένων μιας διαχειριζόμενης συσκευής όταν αυτό ζητηθεί με τις ανάλογες εντολές και όχι η πειραματική προσθήκη στοιχείων μιας τοποθεσίας μιας συσκευής.

3.2.4.2 Διαγραφή σημείου

Η λειτουργία αυτή (Remove GPS Point) δίνει την δυνατότητα στον χρήστη να διαγράψει το τελευταίο γνωστό σημείο μιας συσκευής από την βάση δεδομένων, τον χάρτη και τον πίνακα σημείων.

Η χρήση της λειτουργίας διαγραφής έγκειται στην περίπτωση που ο χρήστης δεν χρειάζεται μια καταχώρηση που αφορά μια συσκευή. Αυτό μπορεί να γίνει όταν πειραματικά καταχωρούνται από τον χρήστη στον χάρτη και την βάση δεδομένων πληροφορίες για την τοποθεσία μιας συσκευής και ουσιαστικά είναι αχρείαστα σε σύγκριση με τις πραγματικές πληροφορίες που ανακτώνται από τις διαχειριζόμενες συσκευές κατά την λειτουργία του όλου συστήματος.

3.2.4.3 Ενημέρωση σημείου

Η λειτουργία αυτή (Remove GPS Point) δίνει την δυνατότητα στον χρήστη να ενημερώσει την βάση δεδομένων και το τελευταίο γνωστό σημείο μιας συσκευής στον πίνακα λεπτομερειών των τελευταίων γνωστών σημείων των διαχειριζόμενων συσκευών, με νέες πληροφορίες.

Η εφαρμογή δίνει την δυνατότητα στον χρήστη επιλέγοντας από τον πίνακα την επιθυμητή τοποθεσία να αλλάξει κάποια από τα χαρακτηριστικά της. Όπως και στις άλλες διεργασίες διαχείρισης σημείων, οι αλλαγές θα ισχύουν μόνο για την εφαρμογή και την βάση δεδομένων του χρήστη, δεν θα γίνει καμιά αλλαγή στα σημεία της ίδιας συσκευής που διαχειρίζονται άλλοι χρήστες από άλλες εφαρμογές. Το γεγονός αυτό έγκειται στην όλη λειτουργία του συστήματος αφού, τα σημεία που ανακτώνται από κάθε συσκευή αποθηκεύονται στην βάση δεδομένων εκάστης εφαρμογής. Δεν υπάρχει καμία αλληλεπίδραση των χρηστών με άλλους χρήστες. Όπως αναφέρθηκε η μόνη περίπτωση ανταλλαγής τέτοιων δεδομένων, αφορά τα στοιχεία-χαρακτηριστικά των συσκευών τα οποία διαβιβάζονται μέσω του UDP Receiver και είναι καθολικά.

3.2.4.4 Εύρεση Σημείου

Η λειτουργία αυτή προσφέρεται από την εφαρμογή και παρέχει εύρεση και εστίαση μιας συσκευής στον χάρτη. Είναι αρκετά χρήσιμη όταν η εφαρμογή διαχειρίζεται μεγάλο αριθμό συσκευών και μεγάλους σε κλίμακα χάρτες. Είναι αδύνατο σε οποιοδήποτε επίπεδο εστίασης στον χάρτη να παρουσιάζονται όλες οι τοποθεσίες συσκευών, άρα η εύρεση σημείου στον χάρτη είναι αναγκαία και υιοθετήθηκε.

3.3 Επιπρόσθετες λειτουργίες

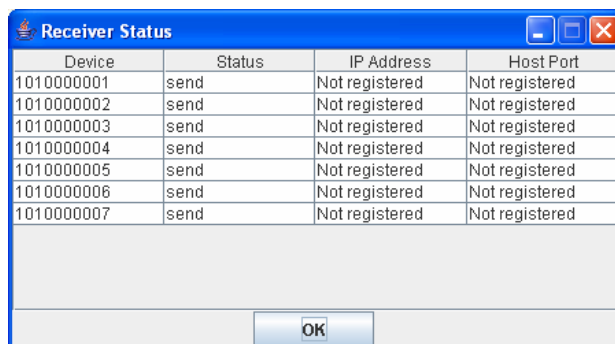
Εκτός από τις κύριες λειτουργίες που αναλύθηκαν πιο πάνω, έχουν αναπτυχθεί και κάποιες δευτερεύουσες λειτουργίες που υποβοηθούν στην καλύτερη οργάνωση και προσφέρουν πληρέστερη ενημέρωση και παρουσίαση στον χρήστη για όλα όσα αφορούν τις συσκευές που διαχειρίζεται και γενικότερα την εφαρμογή.

3.3.1 Ανάκτηση κατάστασης συσκευών από UDP Receiver

Όπως περιγράφεται στην Ενότητα 5 που αναλύει την λειτουργία του UDP Receiver, με την εκκίνηση του, δημιουργεί ένα πίνακα με χαρακτηριστικά των συσκευών καθώς και δυναμικά πεδία που ενημερώνονται κατά την διάρκεια εκτέλεσης του.

Κάθε χρήστης έχει την δυνατότητα εκτός από την ανάκτηση των στατικών χαρακτηριστικών της συσκευής να ζητήσει και να παρακολουθήσει την κατάσταση των δυναμικών πεδίων που αφορούν την κατάσταση λήψης δεδομένων (send/store), την διεύθυνση IP και την θύρα (port) επικοινωνίας κάθε συσκευής. Η λειτουργία αυτή υποβοηθά στην ολοκληρωμένη γνώση κάθε χρήστη για την τρέχουσα κατάσταση διαχείρισης εισερχόμενων δεδομένων κάθε συσκευής ως προς το επικοινωνιακό μέρος της εφαρμογής. Ουσιαστικά ξέρει ανά πάσα χρονική στιγμή κάποια συσκευή έχει εγκαθιδρύσει σύνδεση με τον UDP Receiver άρα μπορεί να ανακτήσει πληροφορίες. Η διεργασία αυτή υλοποιείται μέσω ενός UDP αιτήματος από την εφαρμογή, ο UDP Receiver συλλέγει τις ανάλογες πληροφορίες για τις συσκευές που διαχειρίζεται ο αιτούμενος χρήστης και τις επιστρέφει σε ένα UDP πακέτο το οποίο εν συνεχεία αναλύεται από την εφαρμογή και εμφανίζεται σε μορφή πίνακα (Εικόνα 3.11).

Εικόνα 3.11 – Πίνακας καταστάσεων συσκευών



Device	Status	IP Address	Host Port
1010000001	send	Not registered	Not registered
1010000002	send	Not registered	Not registered
1010000003	send	Not registered	Not registered
1010000004	send	Not registered	Not registered
1010000005	send	Not registered	Not registered
1010000006	send	Not registered	Not registered
1010000007	send	Not registered	Not registered

3.3.2 Προσομοίωση διαδρομής συσκευής

Ένας από τους πρωταρχικούς στόχους που τέθηκαν για την ανάπτυξης της εφαρμογής ήταν η παρακολούθηση διαδοχικών τοποθεσιών μιας συσκευής που καταγράφηκαν και αποθηκεύτηκαν στην βάση δεδομένων.

Ο στόχος αυτός επιτεύχθηκε με την υλοποίηση της λειτουργίας προσομοίωσης διαδρομής. Ο χρήστης έχει την δυνατότητα να επιλέξει το χρονικό διάστημα για το οποίο επιθυμεί να παρακολουθήσει την διαδρομή και αν στην βάση δεδομένων περιέχονται καταχωρήσεις που εμπίπτουν στο διάστημα αυτό, παρουσιάζονται με την χρονική σειρά αποθήκευσης τους στον χάρτη με διαφορά δύο (2) δευτερολέπτων έτσι ώστε να δημιουργείται η εντύπωση κίνησης της συσκευής.

Παράλληλα με την εμφάνιση της συσκευής, σε κάθε χρονική στιγμή αναγράφεται στον πίνακα τελευταίας γνωστής τοποθεσίας κάθε συσκευής οι ανά χρονική στιγμή λεπτομέρειες που χαρακτηρίζουν της διαδρομή της συσκευής.

3.3.3 Εστίαση στην επιλεγμένη συσκευή

Η λειτουργία αυτή εξυπηρετεί τον χρήστη να εστιάζει πάνω σε μια συσκευή όταν ο χάρτης είναι σε κλίμακα η οποία δεν του επιτρέπει να βλέπει την συσκευή αυτή.

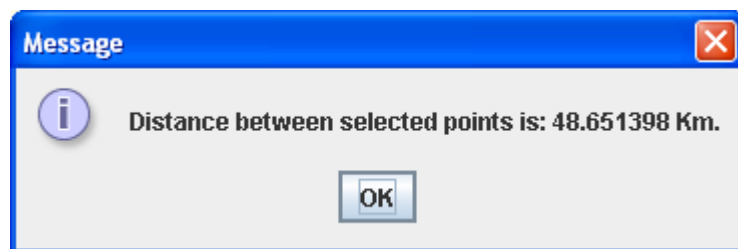
Είναι προφανές ότι ο χρήστης δεν μπορεί να παρακολουθεί όλες τις συσκευές που διαχειρίζεται σε όλες τις κλίμακες όταν η εφαρμογή είναι σε κατάσταση πλοήγησης. Συνεπώς η λειτουργία αυτή μέσω της λίστας συσκευών που βρίσκεται στο αριστερό μέρος προσφέρει μετά από επιλογή της επιθυμητής συσκευής το κεντράρισμα της επιλεγμένης συσκευής στο μέσο του χάρτη (στην υφιστάμενη κλίμακα).

3.3.4 Υπολογισμός απόστασης μεταξύ δύο επιλεγμένων σημείων

Η λειτουργία αυτή αποσκοπεί στην ευκολία προσανατολισμού και κατατόπισης του χρήστη στον χάρτη της εφαρμογής. Η πραγματική απόσταση μεταξύ δύο επιλεγμένων σημείων υπολογίζεται σαν η ευθεία γραμμή που ενώνει τα δύο αυτά σημεία. Δεν λαμβάνονται υπόψη δρόμοι και παρακάμψεις αφού ο συγκεκριμένος τομέας δεν είναι στο πλαίσιο της διπλωματικής εργασίας. Η απόσταση υπολογίζεται σε χιλιόμετρα (Εικόνα 3.12).

Η υλοποίηση στηρίζεται σε πράξεις που γίνονται μεταξύ των συντεταγμένων οθόνης των δύο σημείων – μετά την μετατροπή τους σε γεωγραφικές συντεταγμένες – και την αναπαράσταση του κάθε pixel της οθόνης σε χιλιόμετρα ανάλογα με την κλίμακα που βρίσκεται ο χάρτης την στιγμή που διεκπαιρώνονται οι υπολογισμοί.

Εικόνα 3.12 – Υπολογισμός απόστασης μεταξύ δύο επιλεγμένων σημείων



3.3.5 Υπολογισμός γεωγραφικής τοποθεσίας-περιοχής συσκευής

Ένα από τα πεδία που αποθηκεύονται στην βάση δεδομένων και παράλληλα παρουσιάζονται στον πίνακα τελευταίων γνωστών σημείων στην εφαρμογή είναι η τοποθεσία (location) της συσκευής. Έχει αναπτυχθεί μια λειτουργία με την οποία ανάλογα με τις γεωγραφικές συντεταγμένες σημείου της συσκευής, με κριτήριο κάποιες γνωστές πόλεις, χωριά και τοποθεσίες στο χάρτη της Κύπρου, εισάγεται στο πεδίο αυτό η απόσταση από το πιο κοντινό γνωστό τοπωνύμιο.

Έχει δημιουργηθεί ένας πίνακας στην βάση δεδομένων ο οποίος έχει αποθηκευμένες περίπου σαράντα (40) τοποθεσίες με τις συντεταγμένες τους. Για παράδειγμα: 'Nicosia Center', 35.175, 33.363 που αντιπροσωπεύουν το γεωγραφικό

μήκος (35.175) και πλάτος (33.363) του κέντρου της Λευκωσίας. Να προσθέσω ότι οι συντεταγμένες καταγράφηκαν με βάση τους υφιστάμενους χάρτες της εφαρμογής.

Η υλοποίηση στηρίζεται στην εύρεση της απόλυτης τιμής της απόστασης της ζητούμενης τοποθεσίας από όλα τα αποθηκευμένα σημεία της βάσης δεδομένων (Κεφάλαιο 3.4.6), την ταξινόμηση τους ανά αύξουσα σειρά με βάση την απόσταση αυτή και την ανάκτηση της πρώτης καταχώρησης, συνεπώς και της τοποθεσίας με την μικρότερη απόσταση.

3.3.6 Συντεταγμένες θέσης ποντικιού/ πλαίσιο συχνά χρησιμοποιούμενων λειτουργιών

Η λειτουργία αυτή υλοποιήθηκε με γνώμονα τον εύκολο προσανατολισμό του χρήστη στον χάρτη όσο αφορά τις γεωγραφικές συντεταγμένες τόσο και τις συντεταγμένες οθόνης. Με την εκκίνηση της εφαρμογής δημιουργείται ένα panel το οποίο τοποθετείται στο κάτω μέρος του χάρτη και περιέχει τις τρέχουσες γεωγραφικές συντεταγμένες και τις συντεταγμένες του χάρτη στο σημείο που βρίσκεται το ποντίκι. (Εικόνα 3.13).

Εικόνα 3.13 – Γεωγραφική θέση ποντικιού στον χάρτη



Επιπρόσθετα έχει δημιουργηθεί μια λίστα με κουμπιά που αντιστοιχούν στις πιο συχνά χρησιμοποιούμενες λειτουργίες του συστήματος για διευκόλυνση του χρήστη και έχει τοποθετηθεί στο πάνω μέρος της εφαρμογής (Εικόνα 3.14).

Εικόνα 3.14 – Πλαίσιο συχνά χρησιμοποιούμενων λειτουργιών



3.4 Αλλαγές στις υφιστάμενες λειτουργίες του GPSylon

Χρησιμοποιώντας το λογισμικό ανοικτού κώδικα GPSylon ως βάση για ανάπτυξη της εφαρμογής θεωρήθηκε αναγκαίο να γίνουν ορισμένες προσθαφαιρέσεις σε λειτουργίες και διεργασίες για την κάλυψη των αναγκών και ολοκληρωμένης μετατροπής με βάση τις προδιαγραφές που ορίστηκαν.

3.4.1 Διαγραφή λειτουργιών

Αρχίζοντας από τις λειτουργίες που έχουν αφαιρεθεί εντελώς, ότι αφορούσε σειριακή σύνδεση με GPS συσκευή δεν είναι πλέον σε λειτουργήσιμη κατάσταση. Έχουν διαγραφεί οι λειτουργίες ανάκτησης, αποστολής και παρακολούθησης δεδομένων, παρουσίασης δείγματος εικόνας από την συσκευή και ανάκτησης πληροφορίας για τον κατασκευαστή της συσκευής (πάντα αναφερόμενοι σε σειριακή σύνδεση με GPS συσκευή).

Το GPSylon υποστηρίζει αυτόματη ή επιλεγμένη ανάκτηση και παρουσίαση χαρτών από τον expedia server ανάλογα με την τοποθεσία, κλίμακα και γεωγραφικές συντεταγμένες. Η λειτουργία αυτή έχει αφαιρεθεί για το λόγο ότι οι συγκεκριμένοι χάρτες δεν περιέχουν μεγάλο βαθμό πληροφορίας (αναγράφονται σποραδικά κάποια ονόματα πόλεων και χωριών) και επίσης για το λόγο ότι χρησιμοποιούνται οι δικοί μας χάρτες. Με την λειτουργία αυτή έχουν αφαιρεθεί επίσης οι επιλογές ρύθμισης χρήσης proxy server για κατέβασμα των χαρτών καθώς και η λειτουργία εισαγωγής και εξαγωγής χάρτη από/σε την GPS συσκευή.

Τέλος, έχουν αφαιρεθεί οι λειτουργίες παρουσίασης διαδρομής συσκευής που ανακτάται σειριακά. Στην συγκεκριμένη λειτουργία, οι ανά χρονική στιγμή τοποθεσίες της συσκευής ενώνονται με ευθεία γραμμή και σχηματίζουν την διαδρομή. Επίσης, δεν είναι σε λειτουργία πλέον η διεργασία διαχείρισης της βάσης δεδομένων αφού χρησιμοποιείται MySQL ενώ ήταν προσαρμοσμένη για την Apache Derby .

3.4.2 Αλλαγές σε υφιστάμενες λειτουργίες

Η βασική αλλαγή που έγινε αφορά την διεπιφάνεια της εφαρμογής. Έχει αφαιρεθεί ένα πλαίσιο στο κάτω μέρος της διεπιφάνειας που κατέγραφε και παρουσίαζε τα υφιστάμενα χαρακτηριστικά της σειριακά συνδεδεμένης (γεωγραφικό μήκος και πλάτος, ταχύτητα, υψόμετρο κ.τ.λ.) και αντικαταστάθηκε όπως αναφέρεται και στις προηγούμενες ενότητες του κεφαλαίου αυτού με τον πίνακα χαρακτηριστικών τελευταίων γνωστών σημείων, την λίστα συσκευών καθώς και το πλαίσιο παρουσίασης συντεταγμένων ποντικιού. Επίσης, ο αρχικός χάρτης (παγκόσμιος) έχει αντικατασταθεί από τον χάρτη της Κύπρου που απεικονίζεται σε αρχική κλίμακα 1:60000.

Επιπρόσθετα, έχουν γίνει αλλαγές σε όλες τις κλάσεις που αφορούν παρουσίαση τοποθεσιών συσκευών μέσω ανάκτησης πληροφορίας από την βάση δεδομένων για να συμβαδίζουν με την MySQL βάση δεδομένων που χρησιμοποιείται και τους ανάλογους πίνακες και τα πεδία τους. Οι αλλαγές έγιναν επίσης σε κλάσεις που υλοποιούν καταχώρηση, διαγραφή και ενημέρωση ήδη υπάρχοντων τοποθεσιών στην εφαρμογή.

Κεφάλαιο 4

Επεξεργασία Δεδομένων

4.1 Εισαγωγή	30
4.2 Ανάλυση Δομής Βάσης Δεδομένων	31
4.3 Προτάσεις Δομής Βάσης Δεδομένων με εφαρμογή εφεδρικής αποθήκευσης	34
4.4 Πίνακας τοποθεσιών	46

4.1 Εισαγωγή

Για τις ανάγκες αποθήκευσης, διαχείρισης και ελέγχου των δεδομένων της εφαρμογής όπως προαναφέρθηκε, χρησιμοποιείται η βάση δεδομένων **MySQL**. Οι λόγοι επιλογής και χρήσης της βάσης MySQL έχουν προσδιοριστεί στο Κεφάλαιο 2.3. Στις παρακάτω ενότητες αναλύεται ο καθορισμός της δομής της βάσης δεδομένων που αφορά την δημιουργία πινάκων και την διαχείριση των πεδίων τους. Επίσης, αναλύεται η λειτουργία που αφορά την εφεδρική αποθήκευση (backup) προηγούμενων καταχωρήσεων για τις τοποθεσίες των συσκευών και οι προτάσεις για την δομή της που απορρίφθηκαν και που επιλέχθηκαν. Επιπρόσθετα επεξηγείται το σκεπτικό δημιουργίας και η μορφή του πίνακα που έχει αποθηκευμένες γνωστές τοποθεσίες στην Κύπρο από τις οποίες υπολογίζονται οι αποστάσεις τοποθεσιών των συσκευών όταν αυτό ζητηθεί.

4.2 Ανάλυση Δομής Βάσης Δεδομένων

Για την καταχώρηση, αποθήκευση και ανάκτηση δεδομένων από την βάση χρησιμοποιούνται δύο (2) βασικοί πίνακες.

Ο πρώτος πίνακας αφορά την διαχείριση των συσκευών και των χαρακτηριστικών τους και άμεση αλληλεπίδραση μαζί του έχει ο UDP Receiver. Για οποιοσδήποτε αλλαγές στις πληροφορίες κάθε συσκευής, υπεύθυνος είναι ο UDP Receiver και διενεργούνται πάντα μετά από αίτημα ενός χρήστη μέσω της εφαρμογής. Ο κάθε χρήστης έχει δικαίωμα καταχώρησης, ενημέρωσης, διαγραφής και ανάκτησης πληροφοριών που αφορούν μόνο συσκευές στις οποίες έχει πρόσβαση. Ο πίνακας ονομάζεται **DEVICES** και αποτελείται από τέσσερα (6) πεδία:

- **ID:** Είναι το πρωτεύων κλειδί (PRIMARY KEY) και χαρακτηρίζει μοναδικά την κάθε συσκευή - **VARCHAR (10)**.
- **NAME:** όνομα συσκευής - **VARCHAR (15)**.
- **OWNER_ID:** αναγνωριστικό ιδιοκτήτη της συσκευής - **VARCHAR (50)**.
- **OWNER_PASSWORD:** συνθηματικό χρήστη - **VARCHAR (10)**.
- **DRIVER:** αναγνωριστικό οδηγού του οχήματος που είναι ενσωματωμένη η συσκευή - **VARCHAR (50)**.
- **TYPE:** τύπος της συσκευής (αυτοκίνητο κ.τ.λ.) - **VARCHAR (45)**.

Ο δεύτερος πίνακας είναι υπεύθυνος για την αποθήκευση γεωγραφικών συντεταγμένων και άλλων λεπτομερειών που αφορούν τα χαρακτηριστικά της θέσης των συσκευών σε διάφορα χρονικά σημεία. Άμεση αλληλεπίδραση με τον πίνακα αυτό και τις καταχωρήσεις του, έχουν μόνο οι χρήστες μέσω της εφαρμογής (GPSylon) και των λειτουργιών που σχετίζονται με την διαχείριση των γεωγραφικών σημείων. Ο πίνακας έχει ονομαστεί **POINTS** και αποτελείται από έντεκα (11) πεδία:

- **NUM:** Αποτελεί σε συνδυασμό με το device_id το πρωτεύων κλειδί (PRIMARY KEY) του πίνακα. Αυξάνεται αυτόματα με κάθε καταχώρηση και είναι αναγκαίο για την αποθήκευση περισσότερων από μιας καταχώρησης για την ίδια συσκευή (ίδιο device_id) – **INTEGER**

- **DEVICE_ID:** Χαρακτηρίζει μοναδικά την κάθε συσκευή **VARCHAR(10)**
- **DEVICE_NAME:** Όνομα συσκευής - **VARCHAR(15)**
- **LATITUDE:** Τιμή γεωγραφικού μήκους συσκευής σε συγκεκριμένο χρονικό σημείο - **FLOAT**
- **LONGITUDE:** Τιμή γεωγραφικού πλάτους συσκευής σε συγκεκριμένο χρονικό σημείο - **FLOAT**
- **ALTITUDE:** Τιμή ύψους συσκευής σε συγκεκριμένο χρονικό σημείο - **FLOAT**
- **LOCATION:** Τοποθεσία συσκευής σε συγκεκριμένο χρονικό σημείο - **VARCHAR(100)**
- **SPEED:** Ταχύτητα συσκευής σε συγκεκριμένο χρονικό σημείο-**FLOAT**
- **DATE_TIME:** Χρονικό σημείο επικοινωνίας και λήψης δεδομένων από τη συσκευή - **DATETIME**
- **CATEGORY_ID:** Κατηγορία συσκευής- **VARCHAR(45)**
- **LAST_KNOWN:** Δηλώνει αν η καταχώρηση που είναι/δεν είναι η τελευταία τοποθεσία της συσκευής για σκοπούς απεικόνισης στην εφαρμογή - **BOOLEAN**

Μια επιπλέον λειτουργία που θα εκτελεί το σύστημα είναι η εφεδρική αποθήκευση (backup) των ημερολογιακά παλιών καταχωρήσεων τοποθεσιών των διαχειριζόμενων συσκευών. Οι προβληματισμοί που προέκυψαν για την υλοποίηση της λειτουργίας αυτής αφορούσαν αφ' ενός πότε μια καταχώρηση θεωρείται “παλιά” και αφ' ετέρου ποια μεθοδολογία θα έπρεπε να ακολουθηθεί για την φύλαξη των καταχωρήσεων αυτών. Με γνώμονα την εύκολη ανάκτηση δεδομένων για λειτουργίες που χρειάζονται πληροφορίες από παλαιότερες καταχωρήσεις, όπως για παράδειγμα παρουσίαση διαδρομών μιας συσκευής σε ένα χρονικό διάστημα, μελετήθηκαν διάφορες λύσεις για την πιο βέλτιστη υλοποίηση. Να ξεκαθαριστεί στο σημείο αυτό ότι η εφεδρική αποθήκευση αφορά κάθε χρήστη (και τις καταχωρήσεις των σημείων των συσκευών του) ξεχωριστά και δεν γίνεται καμία αλληλεπίδραση με τον UDP Receiver.

Ακολουθεί το αρχικό σχεσιακό διάγραμμα και οι προτάσεις/σκέψεις που αφορούν την διαμόρφωση της βάσης δεδομένων με την επιπρόσθετη λειτουργία της εφεδρικής αποθήκευσης (backup) από την εφαρμογή.

Σχήμα 4.1 – Αρχικό Relational Schema Diagram

UDP Receiver:

DEVICES

<u>ID</u>	NAME	OWNER	DRIVER	TYPE
-----------	------	-------	--------	------

Client:

POINTS

<u>NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

4.3 Προτάσεις Δομής Βάσης Δεδομένων με εφαρμογή εφεδρικής αποθήκευσης

Για τη εφεδρική αποθήκευση των ‘παλιών’ σημείων που βρίσκονταν μια συσκευή δημιουργείται ένα άλλο schema το **gps_backup**. Να σημειώσω ότι παλιά δεδομένα είναι ΟΛΑ τα δεδομένα πλην των δεδομένων που έχουν αποθηκευτεί την τρέχουσα ημέρα. Η λογική της νέας δομής που αναπτύχθηκε βασίζεται στο να κρατηθεί ο αριθμός και το μέγεθος των πινάκων όσο πιο μειωμένος και μικρός αντίστοιχα γίνεται και παράλληλα να γίνεται πιο εύκολη η πρόσβαση σε αυτούς για ανάκτηση δεδομένων.

4.3.1 Πρόταση 1

Στο σχήμα αυτό, θα περιέχονται 31 πίνακες που αντιστοιχούν στις μέρες κάθε μήνα, 12 πίνακες που αντιστοιχούν στους μήνες του χρόνου, και ένας (1) πίνακας ανά χρόνο που χρησιμοποιείται το λογισμικό. Άρα σύνολο 44 πίνακες (τουλάχιστον).

Καθημερινά, με την έναρξη της εφαρμογής, το πρόγραμμα διαβάζει από τον πίνακα POINTS τα τελευταία γνωστά σημεία για την κάθε συσκευή και τα παρουσιάζει στην διεπιφάνεια. Ακολούθως, όλα τα δεδομένα που υπάρχουν στον POINTS αντιγράφονται στον αντίστοιχο εφεδρικό πίνακα της προηγούμενης μέρας, στον πίνακα του τρέχοντα μήνα καθώς και στον πίνακα του τρέχοντα χρόνου και στην συνέχεια διαγράφονται από τον POINTS.

Άρα ο πίνακας POINTS δεν έχει πλέον δεδομένα και είναι έτοιμος να δεχτεί-αποθηκεύσει πληροφορίες για την τρέχουσα μέρα. Επιπρόσθετα, οι πίνακες που αντιστοιχούν στις μέρες, μήνες και χρόνο που έχουν παρέλθει μέχρι την τρέχουσα ημερομηνία είναι ενημερωμένοι με όλα τα μέχρι την τρέχουσα ημερομηνία δεδομένα που έχουν συλλεχτεί. Επίσης, με την αλλαγή κάθε μήνα, διαγράφονται όλα τα δεδομένα από τους 31 πίνακες που αντιστοιχούν στις μέρες και με την αλλαγή του χρόνου διαγράφονται όλα τα δεδομένα από τους 12 πίνακες που αντιστοιχούν στους μήνες.

Για ανάκτηση πληροφοριών που έχουν διαγραφεί από τις μέρες η εφαρμογή θα αναζητεί στις καταχωρήσεις του μήνα (ανά ημερομηνία) και αντίστοιχα για ανάκτηση πληροφοριών που έχουν διαγραφεί από τους μήνες η εφαρμογή θα αναζητεί στις καταχωρήσεις του χρόνου (ανά μήνα).

Στο τέλος κάθε χρόνου δημιουργείται ο πίνακας του νέου χρόνου και τα δεδομένα των προαναφερθέντων πινάκων καταχωρούνται πλέον σε αυτόν.

Πλεονεκτήματα

- Η αποθήκευση των δεδομένων με την μορφή αυτή διευκολύνει την ανάκτηση του ιστορικού των συσκευών ανά μέρα, μήνα και χρόνο.
- Τα δεδομένα είναι ομαδοποιημένα

Μειονεκτήματα

- Ο αριθμός των πινάκων είναι πολύ μεγάλος και επαναλαμβάνονται ίδιες καταχωρήσεις σε διαφορετικούς πίνακες

Σχήμα 4.2 – Πρόταση 1

GPS_SCHEMA

UDP Receiver:

DEVICES

<u>ID</u>	NAME	OWNER	DRIVER	TYPE
-----------	------	-------	--------	------

Client:

POINTS

<u>NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

GPS_BACKUP_SCHEMA

DAY_1

<u>NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

DAY_2

<u>NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

.

.

DAY_31

<u>NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

MONTH_1

<u>M_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

MONTH_2

<u>M_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

.

.

.

MONTH_12

<u>M_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

YEAR_XXXX

<u>Y_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

4.3.2 Πρόταση 2

Στο σχήμα αυτό, θα περιέχονται ένας (1) πίνακας που αντιστοιχεί στην προηγούμενη μέρα, ένας (1) πίνακας που αντιστοιχεί στον τρέχοντα μήνα, και ένας (1) πίνακας ανά χρόνο που χρησιμοποιείται το λογισμικό. Άρα σύνολο 3 πίνακες (τουλάχιστον).

Καθημερινά, με την έναρξη της εφαρμογής, το πρόγραμμα διαβάζει από τον πίνακα POINTS τα τελευταία γνωστά σημεία για την κάθε συσκευή και τα παρουσιάζει στην διεπιφάνεια. Ακολουθώς, όλα τα δεδομένα που υπάρχουν στον POINTS αντιγράφονται στον πίνακα της προηγούμενης μέρας, στον πίνακα του τρέχοντα μήνα καθώς και στον πίνακα του τρέχοντα χρόνου και στην συνέχεια διαγράφονται από τον POINTS.

Άρα ο πίνακας POINTS δεν έχει πλέον δεδομένα και είναι έτοιμος να δεχτεί-αποθηκεύσει πληροφορίες για την τρέχουσα μέρα. Επίσης, με την αλλαγή κάθε μέρας, διαγράφονται όλα τα δεδομένα από τον πίνακα της προηγούμενης μέρας, και με την αλλαγή κάθε μήνα, διαγράφονται όλα τα δεδομένα από τον πίνακα του τρέχοντα μήνα. Στο τέλος κάθε χρόνου δημιουργείται ο πίνακας του νέου χρόνου και τα δεδομένα των δύο προαναφερθέντων πινάκων καταχωρούνται πλέον σε αυτόν.

Πλεονεκτήματα

- Ο αριθμός των πινάκων είναι μικρός
- Μπορεί να γίνει ανάκτηση προηγούμενων καταχωρήσεων σε αναφορά ανά ημερομηνία στον πίνακα του μήνα ή χρόνου

Μειονεκτήματα

- Η αποθήκευση των δεδομένων με την μορφή αυτή δεν διευκολύνει ιδιαίτερα την ανάκτηση του ιστορικού των συσκευών ανά μέρα και μήνα.
- Τα δεδομένα δεν είναι πλήρως ομαδοποιημένα

Σχήμα 4.3 – Πρόταση 2

GPS_SCHEMA

UDP Receiver:

DEVICES

<u>ID</u>	NAME	OWNER	DRIVER	TYPE
-----------	------	-------	--------	------

Client:

POINTS

<u>NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

GPS_BACKUP_SCHEMA

PREVIOUS_DAY

<u>D_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

CURRENT_MONTH

<u>M_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

YEAR_XXXX

<u>Y_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

4.3.3 Πρόταση 3

Στο σχήμα αυτό, θα περιέχονται N πίνακες που αντιστοιχούν στις συσκευές (N = αριθμός συσκευών), 12 πίνακες που αντιστοιχούν στους μήνες του χρόνου, και ένας (1) πίνακας ανά χρόνο που χρησιμοποιείται το λογισμικό. Άρα σύνολο 13 + N πίνακες (τουλάχιστον).

Καθημερινά, με την έναρξη της εφαρμογής, το πρόγραμμα διαβάζει από τον πίνακα POINTS τα τελευταία γνωστά σημεία για την κάθε συσκευή και τα παρουσιάζει στην διεπιφάνεια. Ακολουθώντας, ανά συσκευή τα δεδομένα που υπάρχουν στον POINTS αντιγράφονται στον πίνακα που αντιστοιχεί στην κάθε συσκευή, και όλες οι καταχωρήσεις αντιγράφονται στον πίνακα του τρέχοντα μήνα καθώς και στον πίνακα του τρέχοντα χρόνου και στην συνέχεια διαγράφονται από τον POINTS.

Άρα ο πίνακας POINTS δεν έχει πλέον δεδομένα και είναι έτοιμος να δεχτεί-αποθηκεύσει πληροφορίες για την τρέχουσα μέρα. Επιπρόσθετα, οι πίνακες που αντιστοιχούν στις συσκευές, καθώς και μήνες και χρόνο που έχουν, είναι ενημερωμένοι με όλα τα μέχρι την τρέχουσα ημερομηνία δεδομένα που έχουν συλλεχτεί. Επίσης, με την αλλαγή του χρόνου διαγράφονται όλα τα δεδομένα από τους 12 πίνακες που αντιστοιχούν στους μήνες. Στο τέλος κάθε χρόνου δημιουργείται ο πίνακας του νέου χρόνου και τα δεδομένα των δύο προαναφερθέντων πινάκων καταχωρούνται πλέον σε αυτόν.

Για ανάκτηση πληροφοριών που έχουν διαγραφεί από τους μήνες η εφαρμογή θα αναζητεί στις καταχωρήσεις του χρόνου (ανά μήνα).

Πλεονεκτήματα

- Η αποθήκευση των δεδομένων με την μορφή αυτή διευκολύνει την ανάκτηση του ιστορικού ανά συσκευή, μήνα και χρόνο.
- Τα δεδομένα είναι ομαδοποιημένα
- Ο αριθμός των πινάκων είναι ικανοποιητικός

Μειονεκτήματα

- Δεν διευκολύνεται η ανάκτηση του ιστορικού των συσκευών ανά μέρα
- Ο αριθμός των συσκευών δεν είναι σταθερός (δυναμικοί πίνακες) – δυσκολία στην διαχείριση

Σχήμα 4.4 – Πρόταση 3

GPS_SCHEMA

UDP Receiver:

DEVICES

<u>ID</u>	NAME	OWNER	DRIVER	TYPE
-----------	------	-------	--------	------

Client:

POINTS

<u>NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

GPS_BACKUP_SCHEMA

DEVICE_1

<u>D_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

DEVICE_2

<u>D_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

.

.

DEVICE_N

<u>D_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

MONTH_1

<u>M_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

MONTH_2

<u>M_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

•
•

MONTH_12

<u>M_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

YEAR_XXXX

<u>Y_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

4.3.4 Πρόταση 4

Στο σχήμα αυτό, θα περιέχονται N πίνακες που αντιστοιχούν στις συσκευές (N = αριθμός συσκευών). Άρα σύνολο N πίνακες .

Καθημερινά, με την έναρξη της εφαρμογής, το πρόγραμμα διαβάζει από τον πίνακα POINTS τα τελευταία γνωστά σημεία για την κάθε συσκευή και τα παρουσιάζει στην διεπιφάνεια. Ακολούθως, ανά συσκευή τα δεδομένα που υπάρχουν στον POINTS και δεν είναι τελευταία γνωστά σημεία αντιγράφονται στον πίνακα που αντιστοιχεί στην κάθε συσκευή ανάλογα και στην συνέχεια διαγράφονται από τον POINTS.

Άρα ο πίνακας POINTS δεν έχει πλέον παλαιότερα δεδομένα, περιέχει μόνο τα τελευταία γνωστά σημεία κάθε συσκευής και είναι έτοιμος να δεχτεί-αποθηκεύσει πληροφορίες για την τρέχουσα μέρα. Επιπρόσθετα, οι πίνακες που αντιστοιχούν στις συσκευές είναι ενημερωμένοι με όλα τα μέχρι την τρέχουσα ημερομηνία δεδομένα που έχουν συλλεχτεί σε φθίνουσα σειρά με βάση την ημερομηνία καταχώρησης.

Πλεονεκτήματα

- Η αποθήκευση των δεδομένων με την μορφή αυτή διευκολύνει την ανάκτηση του ιστορικού ανά συσκευή, μέρα, μήνα και χρόνο.
- Τα δεδομένα είναι ομαδοποιημένα
- Ο αριθμός των πινάκων είναι ικανοποιητικός

Μειονεκτήματα

- Ο αριθμός των συσκευών δεν είναι σταθερός (δυναμικοί πίνακες) – δυσκολία στην διαχείριση

Σχήμα 4.5 – Πρόταση 4

GPS_SCHEMA

UDP Receiver:

DEVICES

<u>ID</u>	NAME	OWNER	DRIVER	TYPE
-----------	------	-------	--------	------

Client:

POINTS

<u>NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

GPS_BACKUP_SCHEMA

DEVICE_1

<u>D_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

DEVICE_2

<u>D_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

...

DEVICE_N

<u>D_NUM</u>	DEVICE_ID	DEVICE_NAME	LATITUDE	LONGITUDE	ALTITUDE	SPEED	LOCATION	DATE_TIME	CATEGORY_ID	LAST_KNOWN
--------------	-----------	-------------	----------	-----------	----------	-------	----------	-----------	-------------	------------

4.3.5 Συμπέρασμα - Επιλογή πρότασης

Η πρόταση που επιλέγηκε είναι η **Πρόταση 4**.

Όπως αναλύθηκε και στα πλεονεκτήματα της Πρότασης 4, η αποθήκευση των δεδομένων με την μορφή αυτή διευκολύνει την ανάκτηση του ιστορικού ανά συσκευή, μέρα, μήνα και χρόνο. Επίσης τα καταχωρημένα δεδομένα είναι ομαδοποιημένα.

Με κύριο γνώμονα ότι μια από τις λειτουργίες θα είναι η αναζήτηση μιας σειράς από θέσεις μιας συσκευής σε καθορισμένα συνεχόμενα χρονικά σημεία, συγκριτικά δίνει πλεονέκτημα στην χρήση αυτής της προτεινόμενης λύσης η οποία παρέχει την ομαδοποίηση και δομή που υποβοηθά την λειτουργία αυτή.

Ο λόγος που απορρίφθηκαν οι άλλες προτάσεις είναι ότι κατά κύριο λόγο είναι πλεονασμός να χωρίζονται οι καταχωρήσεις ανά μέρα ή μήνα. Ουσιαστικά, με τα κριτήρια που θέσαμε από την αρχή, η κύρια λειτουργία ανάκτησης ιστορικού, θα γίνεται κατά την προσομοίωση της διαδρομής μιας συσκευής σε καθορισμένο χρονικό διάστημα. Άρα, με βάση τον αναμενόμενο τρόπο χρήσης και διαχείρισης των δεδομένων ανά χρονικό σημείο, είναι σπάνιο να αναζητούνται καταχωρήσεις μιας μέρας για όλες τις συσκευές αλλά είναι πιο λογικό να ζητά ο χρήστης το ιστορικό των μετακινήσεων μιας συγκεκριμένης συσκευής σε μια χρονική περίοδο όπως αναφέρεται και πιο πάνω.

4.4 Πίνακας τοποθεσιών

Για τον υπολογισμό κατά προσέγγιση της τοποθεσίας – περιοχής που βρίσκεται η συσκευή πριν από την αποθήκευση της, έχει δημιουργηθεί ένας πίνακας τοποθεσιών. Ο πίνακας αυτός αποτελείται από τρία (3) πεδία:

- **LOCATION_NAME:** όνομα τοποθεσίας **VARCHAR(45)**
- **LATITUDE:** γεωγραφικό μήκος τοποθεσίας - **FLOAT**
- **LONGITUDE:** γεωγραφικό πλάτους τοποθεσίας - **FLOAT**

Σε κάθε λειτουργία που σχετίζεται με καταχώρηση νέου σημείου συσκευής στον ανάλογο πίνακα, χρησιμοποιούνται τα δεδομένα που είναι αποθηκευμένα στον πίνακα LOCATIONS και γίνονται διάφοροι υπολογισμοί, οι οποίοι έχουν ως αποτέλεσμα την καταχώρηση στο πεδίο “location” του πίνακα POINTS, της περιοχής που βρίσκεται η συσκευή σύμφωνα με την τελευταία πληροφορία που ανακτήθηκε και την απόσταση από το κέντρο της περιοχής αυτής.

Να σημειωθεί ότι οι γεωγραφικές συντεταγμένες κάθε τοποθεσίας έχουν καταχωρηθεί με βάση τις συντεταγμένες των τοποθεσιών στον δικό μας χάρτη. Η διαδικασία υπολογισμού του πιο κοντινού γνωστού σημείου περιγράφεται στην Ενότητα 3.3.4

Σχήμα 4.6 – Πίνακας τοποθεσιών

LOCATIONS

<u>LOCATION_NAME</u>	LATITUDE	LONGITUDE
----------------------	----------	-----------

Ένα παράδειγμα καταχώρησης στον πίνακα αυτό είναι:

'Larnaca Airport',	35.875	33.622
--------------------	--------	--------

Κεφάλαιο 5

Επικοινωνία

5.1 Εισαγωγή	47
5.2 Μέθοδοι επικοινωνίας	48
5.3 UDP Receiver	58
5.4 Συσκευές GPS	65

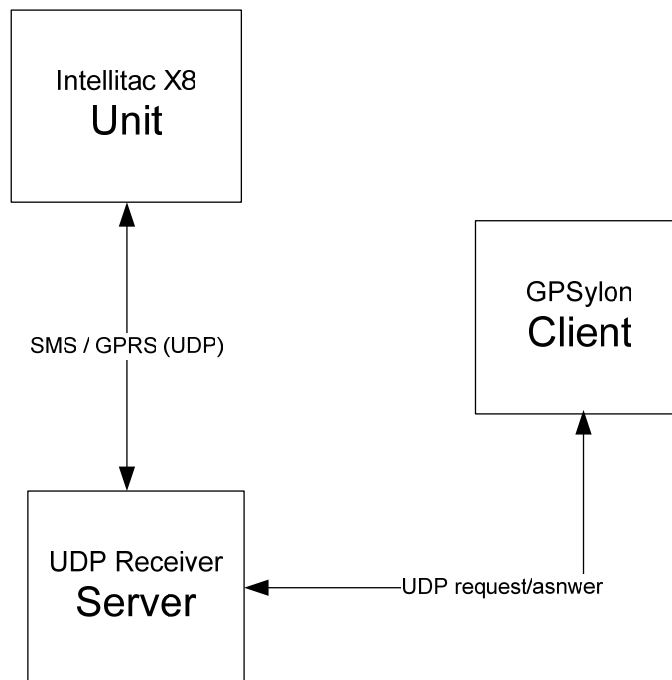
5.1 Εισαγωγή

Όπως θα αναλυθεί στην ενότητα 5.3 και θα επεξηγηθούν οι λόγοι επιλογής συγκεκριμένων πρωτοκόλλων, η επικοινωνία μεταξύ των κομματιών που αποτελούν την εφαρμογή υλοποιείται χρησιμοποιώντας SMS και GPRS με ανταλλαγή πακέτων UDP/IP.

Η γενική ιδέα βασίζεται στην τεχνολογία που υλοποιούν οι περισσότερες συσκευές GPS που υποστηρίζουν πρωτόκολλα επικοινωνίας. Η εφαρμογή παίζει τον ρόλο του client απ' όπου γίνονται διάφορα αιτήματα για ανάκτηση δεδομένων. Επίσης, ένας Server (στην περίπτωση μας UDP Receiver) δημιουργεί ένα “κανάλι” επικοινωνίας με την άλλη πλευρά που είναι ασφαλώς η συσκευή, και παράλληλα δέχεται αιτήματα από τον client. Η επικοινωνία αυτή φαίνεται στο Σχήμα 5.1.

Ουσιαστικά, το μέσο επικοινωνίας και ανάκτησης δεδομένων από μια συσκευή είναι ο UDP Receiver. Ο UDP Receiver έχει άμεση επαφή με την συσκευή, καθώς και με την εφαρμογή και υλοποιεί ένα νοητό συνδετικό κρίκο ανάμεσα τους.

Σχήμα 5.1 – Διάγραμμα επικοινωνίας



5.2 Μέθοδοι επικοινωνίας

Σκοπός του συστήματος είναι η σύνδεση και επικοινωνία με συσκευές που υποστηρίζουν πρωτόκολλα επικοινωνίας και η ανάκτηση δεδομένων τοποθεσίας από αυτές. Με βάση την συσκευή IntelliTrac και τις τεχνολογίες που υλοποιεί, έχει γίνει προσπάθεια ανάπτυξης των πιο βασικών λειτουργιών ανάκτησης τοποθεσίας και διαδρομών με δύο μεθόδους, μέσω SMS και μέσω GPRS(UCP/IP).

5.2.1 Επικοινωνία μέσω SMS

5.2.1.1 Γενικά

Η μια από τις δύο μεθόδους επικοινωνίας και ανάκτησης δεδομένων από τις GPS συσκευές είναι μέσω του πρωτοκόλλου SMS (Short Message Service). Βασική προϋπόθεση είναι ο UDP Receiver να έχει συνδεδεμένη και ενεργοποιημένη συσκευή τύπου GSM modem που του επιτρέπει να αποστέλλει και να λαμβάνει πληροφορίες και δεδομένα.

Μέσα από την εφαρμογή, κάθε χρήστης έχει την δυνατότητα να ανακτήσει πληροφορίες από τις συσκευές που διαχειρίζεται, ζητώντας τις από τον UDP Receiver μέσω ενός UDP-request πακέτου που περιέχει την εκάστοτε εντολή. Να τονιστεί στο σημείο αυτό ότι την υλοποίηση της αποστολής των SMS εντολών προς τις συσκευές αναλαμβάνει πλήρως ο UDP Receiver. Από την πλευρά του χρήστη το μόνο που χρειάζεται είναι να σταλούν οι ορθές για κάθε εντολή παράμετροι σε ένα UDP πακέτο έτσι ώστε ο UDP Receiver να ακολουθήσει την σωστή διαδικασία αποστολής της εντολής στις συσκευές.

Όταν ο UDP Receiver αποστέλλει την εντολή και εν συνεχεία παραλάβει την απάντηση από την συσκευή, συμπιέζει όλα τα δεδομένα (σε καθορισμένο format ανάλογα με το πρωτόκολλο εντολών που υλοποιεί η κάθε συσκευή) σε ένα πίνακα χαρακτήρων και αποστέλλει την απάντηση στον χρήστη. Από το σημείο αυτό και έπειτα αναλαμβάνει η εφαρμογή. Τα εισερχόμενα δεδομένα αναλύονται, διαχωρίζεται το κάθε πεδίο στην απάντηση που παραλαμβάνεται και δημιουργείται μια νέα καταχώρηση που ανάλογα με την εντολή παρουσιάζεται, τοποθετείται στους πίνακες τελευταίων γνωστών τοποθεσιών κάθε συσκευής ή απλά αποθηκεύονται στην βάση δεδομένων για λόγους αρχείου.

Αναγκαία προϋπόθεση επίσης είναι οι διαχειριζόμενες συσκευές να είναι σε κατάσταση SMS έτσι ώστε να μην αγνοούν τις εντολές που τους αποστέλλονται.

Στην παρακάτω ενότητα παρουσιάζονται μια οι λειτουργίες που έχουν υλοποιηθεί και αφορούν επικοινωνία μέσω SMS.

5.2.1.2 Λειτουργίες

- *Ενεργοποίηση GPRS επικοινωνίας μέσω SMS*

Εκτελείται όταν ο χρήστης θέλει να αλλάξει την κατάσταση της συσκευής για να δέχεται και να απαντά GPRS αιτήματα ενώ η τρέχουσα της κατάσταση είναι SMS. Καθορίζονται όλοι οι παράμετροι που αφορούν μια σύνδεση GPRS όπως για παράδειγμα το Access Point Name (APN), η διεύθυνση IP και η θύρα που θα δέχεται ο σταθμός βάσης (base station) τα μηνύματα κ.τ.λ.

- *Ανάκτηση τρέχουσας τοποθεσίας συσκευής μέσω SMS*

Εκτελείται όταν ο χρήστης θέλει να ανακτήσει δεδομένα που αφορούν την τρέχουσα τοποθεσία της συσκευής όταν η τρέχουσα της κατάσταση είναι SMS.

Η απάντηση που θα στείλει η συσκευή μετά από μερικά δευτερόλεπτα έχει το εξής format:

1010000002,20030217132813,121.646060,25.061725,20,157,133,7

Όπου:

Device ID = 1010000002

Year = 2003

Month = 02

Day = 17

Hour = 13

Minute = 28

Second = 13

Longitude = 121.646060

Latitude = 25.061725

Speed = 20 km/h

Heading = 157 degrees

Altitude = 133 meters

Satellites = 7

Η νέα θέση της συσκευής αποθηκεύεται σαν μια νέα καταχώρηση στον αντίστοιχο πίνακα στην βάση δεδομένων.

- *Καταγραφή διαδρομής συσκευής (tracking) μέσω SMS*

Εκτελείται όταν ο χρήστης θέλει να παρακολουθήσει σε πραγματικό χρόνο την διαδρομή της συσκευής όταν η τρέχουσα της κατάσταση είναι SMS. Η διαδικασία εκτελείται τόσες φορές όσες έχει ορίσει ο χρήστης ανάλογα με τις παραμέτρους της εντολής (χρόνο ή απόσταση).

- *Ακύρωση καταγραφής διαδρομής συσκευής (stop tracking) μέσω SMS*

Εκτελείται όταν ο χρήστης θέλει να σταματήσει την παρακολούθηση σε πραγματικό χρόνο της διαδρομής της συσκευής όταν η τρέχουσα της κατάσταση είναι SMS.

- *Αποθήκευση διαδρομής συσκευής (logging) στην μνήμη της συσκευής μέσω SMS*

Εκτελείται όταν ο χρήστης θέλει η συσκευή να αποθηκεύσει στην μνήμη της μια διαδρομή όταν η τρέχουσα της κατάσταση είναι SMS.

- *Ακύρωση αποθήκευσης διαδρομής συσκευής (stop logging) στην μνήμη της συσκευής μέσω SMS*

Εκτελείται όταν ο χρήστης θέλει να διακόψει την αποθήκευση στην μνήμη της συσκευής μιας διαδρομής όταν η τρέχουσα κατάσταση της συσκευής είναι SMS.

- *Διαγραφή αποθηκευμένων διαδρομών (clear logging) στην μνήμη της συσκευής μέσω SMS*

Εκτελείται όταν ο χρήστης θέλει να διαγράψει όλες τις αποθηκευμένες στην μνήμη της συσκευής διαδρομές όταν η τρέχουσα κατάσταση της συσκευής είναι SMS.

- *Καθορισμός παραμέτρων τηλεφώνου σταθμού βάσης (ο αριθμός που θα στέλνει η συσκευή τα δεδομένα όταν είναι σε SMS κατάσταση) μέσω SMS*

Εκτελείται όταν ο χρήστης θέλει να καθορίσει τις παραμέτρους του τηλεφώνου σταθμού βάσης της συσκευής όταν η τρέχουσα της κατάσταση είναι SMS. Καθορίζει τον τρόπο αποστολής δεδομένων στον σταθμό βάσης (base station), αν η συσκευή θα δέχεται εντολές μόνο από τον αριθμό τηλεφώνου του σταθμού βάσης ή από οποιοδήποτε αριθμό, τον αριθμό τηλεφώνου παροχέα-κέντρου μηνυμάτων (Κύπρος: +35799700000) και τον αριθμό τηλεφώνου σταθμού βάσης (base station)

- *Καθορισμός PIN της κάρτας που βρίσκεται στην συσκευή μέσω SMS*

Εκτελείται όταν ο χρήστης θέλει να καθορίσει το PIN της SIM card που είναι εγκατεστημένη στην συσκευή όταν η τρέχουσα της κατάσταση είναι SMS.

- *Καθορισμός συνθηματικού (password) συσκευής μέσω SMS*

Εκτελείται όταν ο χρήστης θέλει να καθορίσει νέο συνθηματικό την συσκευής όταν η τρέχουσα της κατάσταση είναι SMS.

5.2.2 Επικοινωνία μέσω GPRS

5.2.2.1 Γενικά

Η δεύτερη μέθοδος επικοινωνίας και ανάκτησης δεδομένων από τις GPS συσκευές εκτελείται χρησιμοποιώντας το πρωτόκολλο GPRS (General Packet Radio Service). Μέσα από την εφαρμογή, ο κάθε χρήστης έχει την δυνατότητα ανάκτησης πληροφορίας από τις συσκευές που διαχειρίζεται, ζητώντας τις από τον UDP Receiver μέσω ενός UDP-request πακέτου που περιέχει την εκάστοτε εντολή. Από την πλευρά του χρήστη, το μόνο που χρειάζεται είναι να σταλούν οι ορθές για κάθε εντολή παράμετροι σε ένα UDP πακέτο έτσι ώστε ο UDP Receiver να ακολουθήσει την σωστή διαδικασία.

Όταν ο UDP Receiver αποστέλλει την εντολή και εν συνεχεία παραλάβει την απάντηση από την συσκευή, συμπιέζει όλα τα δεδομένα (σε καθορισμένο format ανάλογα με το πρωτόκολλο εντολών που υλοποιεί η κάθε συσκευή) σε ένα πίνακα χαρακτήρων και αποστέλλει την απάντηση στον χρήστη. Από το σημείο αυτό και έπειτα αναλαμβάνει η εφαρμογή. Τα εισερχόμενα δεδομένα αναλύονται, διαχωρίζεται το κάθε πεδίο στην απάντηση που παραλαμβάνεται και δημιουργείται μια νέα καταχώρηση που ανάλογα με την εντολή παρουσιάζεται, τοποθετείται στους πίνακες τελευταίων γνωστών τοποθεσιών κάθε συσκευής ή απλά αποθηκεύονται στην βάση δεδομένων για λόγους αρχείου.

Αναγκαία προϋπόθεση επίσης είναι οι διαχειριζόμενες συσκευές να είναι σε κατάσταση GPRS έτσι ώστε να μην αγνοούν τις εντολές που τους αποστέλλονται. Έχει επιλεγεί η χρήση του πρωτοκόλλου UDP αντί του TCP για λόγους που θα επεξηγηθούν στην συνέχεια (Ενότητα 5.3). Όταν η συσκευή τεθεί σε λειτουργία, σε διαδοχικά διαστήματα χρόνου (time interval) που καθορίζονται από τον χρήστη στέλνει μηνύματα συγχρονισμού στον UDP Receiver. Με τον τρόπο αυτό, ο UDP Receiver έχει αποθηκευμένο την ανά πάσα στιγμή IP διεύθυνση της συσκευής καθώς και την θύρα επικοινωνίας (port) έτσι ώστε να υπάρχει δυνατότητα ορθής αποστολής μηνυμάτων και αναγνώρισης της προέλευσης οποιουδήποτε εισερχόμενου μηνύματος.

Στην παρακάτω ενότητα παρουσιάζονται μια προς μια οι λειτουργίες που έχουν υλοποιηθεί και αφορούν επικοινωνία μέσω GPRS.

5.2.2.2 Λειτουργίες

- *Απενεργοποίηση/καθορισμός νέων παραμέτρων GPRS επικοινωνίας μέσω GPRS*

Εκτελείται όταν ο χρήστης θέλει να αλλάξει την κατάσταση της συσκευής για να δέχεται και να απαντά SMS αιτήματα ενώ η τρέχουσα της κατάσταση είναι GPRS. Καθορίζονται όλοι οι παράμετροι που αφορούν μια σύνδεση GPRS όπως για παράδειγμα το Access Point Name (APN), η διεύθυνση IP και η θύρα που θα δέχεται ο σταθμός βάσης (base station) τα μηνύματα κ.τ.λ.

- *Ανάκτηση τρέχουσας τοποθεσίας συσκευής μέσω GPRS*

Εκτελείται όταν ο χρήστης θέλει να ανακτήσει δεδομένα που αφορούν την τρέχουσα τοποθεσία της συσκευής όταν η τρέχουσα της κατάσταση είναι GPRS.

Η νέα θέση της συσκευής αποθηκεύεται σαν μια νέα καταχώρηση στον αντίστοιχο πίνακα στην βάση δεδομένων.

- *Καταγραφή διαδρομής συσκευής (tracking) μέσω GPRS*

Εκτελείται όταν ο χρήστης θέλει να παρακολουθήσει σε πραγματικό χρόνο την διαδρομή της συσκευής όταν η τρέχουσα της κατάσταση είναι GPRS.

Η απάντηση που θα στείλει η συσκευή μετά από μερικά δευτερόλεπτα έχει το εξής format:

\$OK:TRACKING

1010000002,20030217144230,121.646102,25.061398,0,0,139,0

1010000002,20030217144245,121.646102,25.061398,0,0,139,0

1010000002,20030217144300,121.646102,25.061398,0,0,139,0

1010000002,20030217144315,121.646102,25.061398,0,0,139,0

1010000002,20030217144330,121.646102,25.061398,0,0,139,0

Όπου (π.χ. για την πρώτη καταχώρηση):

Device ID = 1010000001

Year = 2003

Month = 02

Day = 17
Hour = 14
Minute = 42
Second = 30
Longitude = 121.646102
Latitude = 25.061398
Speed = 0 km/h
Heading = 0 degrees
Altitude = 139 meters
Satellites = 0

Η διαδικασία εκτελείται τόσες φορές όσες έχει ορίσει ο χρήστης ανάλογα με τις παραμέτρους της εντολής (χρόνο ή απόσταση).

- *Ακύρωση καταγραφής διαδρομής συσκευής (stop tracking) μέσω GPRS*

Εκτελείται όταν ο χρήστης θέλει να σταματήσει την παρακολούθηση σε πραγματικό χρόνο της διαδρομής της συσκευής όταν η τρέχουσα της κατάσταση είναι GPRS.

- *Αποθήκευση διαδρομής συσκευής (logging) στην μνήμη της συσκευής μέσω GPRS*

Εκτελείται όταν ο χρήστης θέλει η συσκευή να αποθηκεύσει στην μνήμη της μια διαδρομή όταν η τρέχουσα της κατάσταση είναι GPRS.

- *Ακύρωση αποθήκευσης διαδρομής συσκευής (stop logging) στην μνήμη της συσκευής μέσω GPRS*

Εκτελείται όταν ο χρήστης θέλει να διακόψει την αποθήκευση στην μνήμη της συσκευής μιας διαδρομής όταν η τρέχουσα κατάσταση της συσκευής είναι GPRS.

- *Διαγραφή αποθηκευμένων διαδρομών (clear logging) στην μνήμη της συσκευής μέσω GPRS*

Εκτελείται όταν ο χρήστης θέλει να διαγράψει όλες τις αποθηκευμένες στην μνήμη της συσκευής διαδρομές όταν η τρέχουσα κατάσταση της συσκευής είναι GPRS.

- *Ανάκτηση αποθηκευμένων διαδρομών (get log data) από την μνήμη της συσκευής στον υπολογιστή μέσω GPRS*

Εκτελείται όταν ο χρήστης θέλει να ανακτήσει και να αποθηκεύσει στην βάση δεδομένων όλες τις διαδρομές που έχει αποθηκευμένες στην μνήμη της η συσκευή μετά από την ανάλογη εντολή, όταν η τρέχουσα της κατάσταση είναι GPRS.

Η απάντηση που θα στείλει η συσκευή μετά από μερικά δευτερόλεπτα έχει το εξής format:

\$OK:GETLOG

1010000001, 20030105092129, 121.651598, 25.052325, 0, 0, 33, 0

1010000001, 20030105092130, 121.651598, 25.052325, 0, 0, 33, 0

1010000001, 20030105092131, 121.651598, 25.052325, 0, 0, 33, 0

1010000001, 20030105092132, 121.651598, 25.052325, 0, 0, 33, 0

1010000001, 20030105092133, 121.651598, 25.052325, 0, 0, 33, 0

1010000001, 20030105092134, 121.651598, 25.052325, 0, 0, 33, 0

\$MSG:Download Completed

Όπου (π.χ. για την πρώτη καταχώρηση):

Device ID = 1010000001

Year = 2003

Month = 01

Day = 05

Hour = 09

Minute = 21

Second = 29

Longitude = 121.651598

Latitude = 25.052325

Speed = 0 km/h

Heading = 0 degrees

Altitude = 33 meters

Satellites = 0

Τα στοιχεία αυτά αναλύονται και παράλληλα, οι νέες θέσεις της συσκευής μέχρι το μήνυμα “\$MSG:Download Completed” αποθηκεύονται σαν νέες καταχωρήσεις στον αντίστοιχο πίνακα στην βάση δεδομένων.

- *Ακύρωση ανάκτησης αποθηκευμένων διαδρομών από την μνήμη της συσκευής στον υπολογιστή μέσω GPRS*

Εκτελείται όταν ο χρήστης θέλει να διακόψει την ανάκτηση των αποθηκευμένων διαδρομών από την μνήμη της συσκευής, όταν η τρέχουσα κατάσταση της συσκευής είναι GPRS

- *Ανάκτηση επιλεγμένων (κατά ημερομηνία) αποθηκευμένων διαδρομών (get selected log data) από την μνήμη της συσκευής στον υπολογιστή μέσω GPRS*

Εκτελείται όταν ο χρήστης θέλει να ανακτήσει και να αποθηκεύσει στην βάση δεδομένων όλες τις διαδρομές που έχει αποθηκευμένες στην μνήμη της η συσκευή σε καθορισμένο χρονικό διάστημα που επιλέγει ο χρήστης, μετά από την ανάλογη εντολή, όταν η τρέχουσα της κατάσταση είναι GPRS.

5.3 UDP Receiver

5.3.1 Γενικά

Για την διαχείριση των χρηστών και των συσκευών τους, την αποστολή και παραλαβή μηνυμάτων και δεδομένων από τις συσκευές καθώς και αιτημάτων από την εφαρμογή (χρήστη) αναπτύχθηκε ένας παραλήπτης/αποστολέας UDP πακέτων.

Ο λόγος που προτιμήθηκε η χρήση του πρωτοκόλλου UDP αντί του TCP είναι βασικά η αποφυγή του 3-way handshake που εγκαθιδρύει την σύνδεση μεταξύ client-server. Επίσης, αν οι ανταλλαγές μηνυμάτων μεταξύ client και server γίνονται σε αραιά χρονικά διαστήματα (π.χ. κάθε 4-5 λεπτά) είναι απαραίστο να ανταλλάσσονται τα περιττά αυτά μηνύματα εγκαθίδρυσης επικοινωνίας ή να παραμένει η σύνδεση ανοικτή και ανενεργή χωρίς λόγο για μεγάλο χρονικό διάστημα,.

Για την υλοποίηση των λειτουργιών της εφαρμογής (ουσιαστικά την διαχείριση των συσκευών) γίνεται αρχικά μια σύνδεση με τον UDP Receiver από τον χρήστη χρησιμοποιώντας τα συνθηματικά εισόδου. Αν η σύνδεση είναι έγκυρη ο χρήστης μπορεί να εκτελέσει οποιαδήποτε λειτουργία αρχικά ζητώντας και ακολούθως ενημερώνοντας με νέα δεδομένα τον UDP Receiver για τυχόν αλλαγές που έγιναν. Οι λειτουργίες αυτές είναι απλές και το μέγεθος των δεδομένων που ανταλλάσσονται είναι τέτοια ώστε το packet ordering να μην είναι τόσο σημαντικό και το UDP να προσφέρει την απαραίτητη ταχύτητα στις ανταλλαγές μηνυμάτων.

Όσο αφορά την εγκαθίδρυση σύνδεσης με τις συσκευές αποφεύχθηκε το 3-way handshake για λόγους κόστους και multicasting. Συγκεκριμένα, στο πρωτόκολλο επικοινωνίας που υλοποιεί η συσκευή IntelliTrac X8 (στην οποία βασίζεται η υλοποίηση της εφαρμογής) μετά από τα γνωστά μηνύματα SYN, SYN-ACK και ACK, για την επικύρωση της σύνδεσης αποστέλλεται από την συσκευή μια σειρά χαρακτήρων την οποία ο UDP Receiver πρέπει να απαντήσει/στείλει πίσω όπως ακριβώς την παραλαμβάνει (ίσως για λόγους περαιτέρω ασφάλειας και επικύρωσης σύνδεσης). Το σημείο αυτό μας είχε δυσκολέψει ιδιαίτερα μέχρι να βρούμε την λύση (απλά δηλαδή, την αποστολή του μηνύματος όπως ακριβώς έχει σταλεί), αφού παρατηρώντας τα εισερχόμενα πακέτα εγκαθίδρυσης σύνδεσης, ο UDP Receiver

απαντούσε κανονικά στα μηνύματα από την συσκευή IntelliTrac X8 αλλά η σύνδεση σε κάποια φάση χανόταν χωρίς λόγο.

Επιστρέφοντας στην επιλογή του UDP, αυτό έχει ως αποτέλεσμα την δημιουργία και αποστολή αρκετών πακέτων σε κάθε εγκαθίδρυση σύνδεσης από την συσκευή με συνέπεια την αύξηση της ογκοχρέωσης για την GPRS επικοινωνία. Επιπρόσθετα, με την χρήση του πρωτοκόλλου UDP, η επικοινωνία με πάνω από μια συσκευή γίνεται πιο αποτελεσματική αφού υπάρχει δυνατότητα αποστολής μηνυμάτων σε πολλαπλούς προορισμούς ταυτόχρονα. Αντίθετα, χρησιμοποιώντας TCP, για κάθε αποστολή πακέτων σε μια συσκευή θα έπρεπε να δημιουργηθούν ξεχωριστές συνδέσεις για να αποσταλούν ξεχωριστά μηνύματα.

Επίσης, ο UDP Receiver όπως έχει υλοποιηθεί υποστηρίζει αποστολή και παραλαβή μηνυμάτων SMS. Με την εκκίνηση του και μετά από κάποιες ενέργειες που αναλύονται πιο κάτω, ξεκινά μια διεργασία που τρέχει σε όλο το χρόνο εκτέλεσης του και αφορά το διάβασμα εισερχομένων SMS μέσω μιας συσκευής τύπου GSM που είναι συνδεδεμένο με τον υπολογιστή που τρέχει ο UDP Receiver. Ανάλογα με το αίτημα που κάνει ο χρήστης από την εφαρμογή (μέσω ενός UDP πακέτου), πάντα αναφερόμενοι στην κατάσταση που η συσκευή είναι σε κατάσταση αποδοχής/αποστολής δεδομένων μέσω SMS, ο UDP Receiver “διαβάζει” εισερχόμενα μηνύματα SMS και τα διαβιβάζει όπως τα άλλα πακέτα στον χρήστη.

5.3.2 Λειτουργία

Η ουσιαστική λειτουργία του UDP Receiver είναι η παραλαβή πακέτων που περιέχουν εντολές και δεδομένα και αφορούν αιτήματα μέσω της εφαρμογής καθώς και επικοινωνία μέσω SMS και GPRS με τις συσκευές. Πιο κάτω αναλύονται σε θεματικές ενότητες οι διάφορες λειτουργίες.

5.3.2.1 Πίνακας χαρακτηριστικών συσκευών

Με την εκκίνηση του, ο UDP Receiver αρχικά δημιουργεί ένα πίνακα με όλες τις διαχειριζόμενες συσκευές ανακτώντας τις λεπτομέρειες τους από την βάση δεδομένων. Συγκεκριμένα στον πίνακα αποθηκεύονται τα εξής στοιχεία για κάθε συσκευή:

- i. Ιδιοκτήτης συσκευής (owner)
- ii. Συνθηματικό πρόσβασης ιδιοκτήτη (password)
- iii. Αναγνωριστικό συσκευής (device id)

Επίσης δημιουργούνται ακόμα τέσσερα (4) πεδία που δεν ανακτώνται από την βάση δεδομένων αλλά κατακρατούν άλλες runtime πληροφορίες που αφορούν την συσκευή:

- iv. Logged in/Logged out: Το πεδίο παίρνει την ανάλογη τιμή για κάθε συσκευή αν ο χρήστης που την διαχειρίζεται είναι συνδεδεμένος ή όχι με τον UDP Receiver.
- v. Status: Το πεδίο αυτό παίρνει τιμές *send* ή *store*.

Αν η τιμή του πεδίου είναι *store* τότε κάθε δεδομένο που έρχεται από την συγκεκριμένη συσκευή στον UDP Receiver αποθηκεύεται και δεν αποστέλλεται στον χρήστη για παρουσίαση στον χάρτη.

Αν η τιμή του πεδίου είναι *send* τότε κάθε δεδομένο που έρχεται από την συγκεκριμένη συσκευή στον UDP Receiver αποστέλλεται στον client για παρουσίαση στον χάρτη.

- vi. IP Address: Το πεδίο αυτό αποθηκεύει την διεύθυνση της συσκευής όταν αυτή εγκαθιδρύσει σύνδεση με τον UDP Receiver. Αν η συσκευή δεν είναι συνδεδεμένη το πεδίο έχει την τιμή “Not registered”.
- vii. Port: Το πεδίο αυτό αποθηκεύει το port της συσκευής όταν αυτή εγκαθιδρύσει σύνδεση με τον UDP Receiver. Αν η συσκευή δεν είναι συνδεδεμένη το πεδίο έχει την τιμή “Not registered”.

Τα στοιχεία αυτά που αποθηκεύονται χρησιμεύουν στον έλεγχο εγκυρότητας και ανταλλαγή πληροφοριών μεταξύ χρηστών και UDP Receiver.

5.3.2.2 Επικοινωνία μέσω Sockets

Αφού δημιουργηθεί ο πίνακας, ο UDP Receiver ανοίγει ένα socket σε ένα συγκεκριμένο port και “ακούει” για παραλαβή πακέτων από χρήστες ή συσκευές.

Όταν παραληφθεί ένα πακέτο εξετάζεται, φιλτράρεται και επεξεργάζεται ανάλογα με τον αποστολέα και το περιεχόμενο του. Τα πακέτα αφορούν:

- i. Διαχείριση συσκευών (καταχώρηση, διαγραφή, ενημέρωση)
- ii. Σύνδεση/Αποσύνδεση χρήστη με τον UDP Receiver
- iii. Καθορισμός κατάστασης συσκευών (status, IP address, port)
- iv. Μήνυμα χρήστη για επικοινωνία/ανάκτηση δεδομένων από συσκευή που του ανήκει.
- v. Πακέτο πληροφορίας τοποθεσίας ή άλλων δεδομένων από συσκευή GPS.

Απαριθμούνται παρακάτω τα πιθανά μηνύματα από την εφαρμογή (χρήστη):

1. Σύνδεση/αποσύνδεση εφαρμογής με UDP Receiver
2. Καθορισμός κατάστασης αποστολής/αποθήκευσης (store/send) συσκευής
3. Καταχώρηση νέας συσκευής
4. Διαγραφή υφιστάμενης συσκευής
5. Ενημέρωση στοιχείων συσκευής
6. Ανάκτηση χαρακτηριστικών συσκευών (πίνακα) που διαχειρίζεται ο αιτούμενος χρήστης
7. Ενεργοποίηση GPRS επικοινωνίας μέσω SMS
8. Απενεργοποίηση/καθορισμός νέων παραμέτρων GPRS επικοινωνίας μέσω GPRS
9. Ανάκτηση τρέχουσας τοποθεσίας συσκευής μέσω SMS
10. Ανάκτηση τρέχουσας τοποθεσίας συσκευής μέσω GPRS
11. Καταγραφή διαδρομής συσκευής (tracking) μέσω SMS
12. Καταγραφή διαδρομής συσκευής (tracking) μέσω GPRS
13. Ακύρωση καταγραφής διαδρομής συσκευής (stop tracking) μέσω SMS
14. Ακύρωση καταγραφής διαδρομής συσκευής (stop tracking) μέσω GPRS

15. Αποθήκευση διαδρομής συσκευής (logging) στην μνήμη της συσκευής μέσω SMS
16. Αποθήκευση διαδρομής συσκευής (logging) στην μνήμη της συσκευής μέσω GPRS
17. Ακύρωση αποθήκευσης διαδρομής συσκευής (stop logging) στην μνήμη της συσκευής μέσω SMS
18. Ακύρωση αποθήκευσης διαδρομής συσκευής (stop logging) στην μνήμη της συσκευής μέσω GPRS
19. Διαγραφή αποθηκευμένων διαδρομών (clear logging) στην μνήμη της συσκευής μέσω SMS
20. Διαγραφή αποθηκευμένων διαδρομών (clear logging) στην μνήμη της συσκευής μέσω GPRS
21. Ανάκτηση αποθηκευμένων διαδρομών (get log data) από την μνήμη της συσκευής στον υπολογιστή μέσω GPRS
22. Ακύρωση κατεβάσματος αποθηκευμένων διαδρομών από την μνήμη της συσκευής στον υπολογιστή μέσω GPRS
23. Ανάκτηση επιλεγμένων (κατά ημερομηνία) αποθηκευμένων διαδρομών (get selected log data) από την μνήμη της συσκευής στον υπολογιστή μέσω GPRS
24. Καθορισμός παραμέτρων τηλεφώνου σταθμού βάσης (ο αριθμός που θα στέλνει η συσκευή τα δεδομένα όταν είναι σε SMS κατάσταση) μέσω SMS
25. Καθορισμός PIN της κάρτας που βρίσκεται στην συσκευή μέσω SMS
26. Καθορισμός συνθηματικού (password) συσκευής μέσω SMS

- Τα μηνύματα 1 - 5 επιστρέφουν στον χρήστη επιβεβαίωση ότι το αίτημα του έχει ικανοποιηθεί ή όχι από τον UDP Receiver
- Το μήνυμα 6 επιστρέφει στον χρήστη τα δεδομένα που αφορούν μόνο τις συσκευές που διαχειρίζεται.
- Τα μηνύματα 7 - 26 καλούν τον UDP Receiver να αποστείλει συγκεκριμένο αίτημα σε μια συσκευή ανάλογα μέσω SMS ή GPRS και παίρνοντας την απάντηση μετά από κάποια δευτερόλεπτα, ο UDP Receiver την αποστέλλει πίσω στον χρήστη. Η απάντηση μπορεί να περιέχει μια απλή επιβεβαίωση λειτουργίας ή δεδομένα τοποθεσίας που θα διαχειριστούν ανάλογα από τον χρήστη.

5.3.2.3 Διαχείριση μηνυμάτων SMS

Ο UDP Receiver κατά την διάρκεια της εκτέλεσης του, ελέγχει την συσκευή τύπου GSM που είναι συνδεδεμένη με τον υπολογιστή για νέα μηνύματα που σχετίζονται με τοποθεσία συσκευής ή άλλα συναφή μηνύματα.

Για την υλοποίηση της λειτουργίας αυτής έχουν γίνει κάποιες ρυθμίσεις παραμέτρων που είναι αναγκαίες για να ικανοποιούν τις λειτουργίες του UDP Receiver σε ότι αφορά SMS επικοινωνία, καθώς και του τηλεφωνικού δικτύου της Κύπρου. Ουσιαστικά οι αλλαγές αφορούν τα εξής:

1. Καθορισμός serial port που είναι συνδεδεμένη η συσκευή τύπου GSM – αναγκαία η χρησιμοποίηση οδηγών (drivers) για την εγκατάσταση και λειτουργία στον υπολογιστή που τρέχει ο UDP Receiver αν το λειτουργικό σύστημα είναι Windows.
2. Καθορισμός αριθμού PIN της κάρτας SIM της συσκευής τύπου GSM
3. Καθορισμός αριθμού SMSC (Short Message Service Center)
4. Καθορισμός διαστήματος χρόνου μεταξύ διαδοχικών ελέγχων για νέο εισερχόμενο μήνυμα (Poll Interval)

Η υλοποίηση του συνεχούς ελέγχου νέων μηνυμάτων έγινε με χρήση Threads. Κάθε πέντε (5) δευτερόλεπτα γίνεται έλεγχος στην συσκευή τύπου GSM και αν υπάρχει νέο (και παράλληλα έγκυρο-που αφορά το σύστημα) μήνυμα αποθηκεύεται ή αποστέλλεται στον χρήστη που ανήκει η αναφερόμενη συσκευή. Επιπρόσθετα, σε περίπτωση που υπάρχει αίτημα για αποστολή μηνύματος SMS, το serial port που είναι συνδεδεμένο με την συσκευή τύπου GSM για διάβασμα, απελευθερώνεται από τον συνεχή έλεγχο για νέα μηνύματα, έτσι ώστε να χρησιμοποιηθεί από την νέα διεργασία. Όταν το μήνυμα αποσταλεί επιτυχώς επανεκκινείται η λειτουργία συνεχούς ελέγχου για την σωστή συνέχεια της λειτουργίας του UDP Receiver.

5.3.3 Υλοποίηση

Για να ικανοποιείται και να ισχύει ο modular σχεδιασμός του συστήματος, έχει υλοποιηθεί μια abstract κλάση που περιέχει τις βασικές μεθόδους του UDP Receiver και μια κλάση που τις υλοποιεί με βάση τον τύπο συσκευής που χρησιμοποιείται στο σύστημα (IntelliTrac X8).

Οι κύριες λειτουργίες του Receiver είναι:

1. Εκκίνηση λειτουργίας του UDP Receiver δημιουργώντας τον πίνακα χαρακτηριστικών συσκευών και ανοίγοντας την θύρα (port) για παραλαβή αιτημάτων από χρήστες και συσκευές.
2. Τερματισμός λειτουργίας UDP Receiver.
3. Παραλαβή και αποκωδικοποίηση δεδομένων πληροφορίας – μετατροπή bytes σε πίνακα χαρακτήρων για επεξεργασία.
4. Επεξεργασία αιτήματος από χρήστη που αφορά διαχείριση συσκευών (καταχώρηση, διαγραφή, ενημέρωση) ή αναφορά κατάστασης συσκευών (διεύθυνση IP , θύρα παραλαβής πακέτων)
5. Επεξεργασία πακέτων που σχετίζονται με την αμφίδρομη επικοινωνία με τις συσκευές.
6. Επεξεργασία αιτήματος από χρήστη που αφορά επικοινωνία (SMS/GPRS) και αποστολή εντολής - ανάκτηση πληροφορίας από τις συσκευές.

Οι εντολές που υλοποιήθηκαν και αφορούν την επικοινωνία και ανάκτηση δεδομένων από την συσκευή IntelliTrac X8 αναγράφονται στα Παραρτήματα Β και Γ.

5.4 Συσκευές GPS

Σύμφωνα με τις προδιαγραφές ανάπτυξης του συγκεκριμένου συστήματος, θα πρέπει να χρησιμοποιηθούν συσκευές GPS που υλοποιούν ένα πρωτόκολλο επικοινωνίας, από τις οποίες θα ανακτώνται δεδομένα σε κάποια γνωστή μορφή έτσι ώστε να τυγχάνουν ανάλυσης και επεξεργασίας.

Για το δικό μας σύστημα, αρχικά χρησιμοποιήθηκε η συσκευή **Mitac MIO A701**. Η συγκεκριμένη συσκευή είναι τύπου Pocket PC και λειτουργεί κάτω από την πλατφόρμα Windows Mobile 5.0. Όσο αφορά μεθόδους επικοινωνίας, υποστηρίζει Bluetooth, USB και GPRS. Επίσης, σαν συσκευή GPS υποστηρίζεται από το σύστημα SiRFstar III.

Το μειονέκτημα της συσκευής MIO A701 έγκειται στο γεγονός ότι ουσιαστικά είναι μια τηλεφωνική συσκευή που έχει την επιπρόσθετη λειτουργία πλοήγησης μέσω ανάκτησης γεωγραφικών συντεταγμένων από την GPS τεχνολογία που διαθέτει. Αν και λειτουργεί κάτω από Windows Mobile, η συσκευή δεν είχε εγκατεστημένη Java Virtual Machine (JVM) για παροχή διεργασιών που είναι γραμμένες σε JAVA. Για τον λόγο αυτό, έγινε εγκατάσταση δωρεάν JVMs όπως για παράδειγμα της **Mysaifu** που πρόσφερε την επιλογή στον χρήστη να τρέξει .jar αρχεία - ήδη μεταγλωττισμένα Java αρχεία από τον υπολογιστή.

Ουσιαστικά έγινε προσπάθεια προγραμματισμού της συγκεκριμένης συσκευής, όμως προέκυπταν συνεχώς προβλήματα που αφορούσαν αφ' ενός την ανάκτηση των GPS δεδομένων από την προκαθορισμένη θύρα εξόδου (για την συγκεκριμένη συσκευή το COM4) και αφ' ετέρου την επικοινωνία της συσκευής μέσω SMS ή GPRS με τον σταθμό βάσης για αποστολή δεδομένων ή παραλαβή διαφόρων αιτημάτων.

Η συσκευή αν και προσφέρει την αποστολή γεωγραφικών συντεταγμένων μέσω SMS σε περίπτωση ενεργοποίησης έκτακτου συναγερμού (SOS), αποστέλλει μόνο το γεωγραφικό μήκος και πλάτος σε ένα επιλεγμένο τηλεφωνικό αριθμό (η εφαρμογή μας απαιτεί επιπρόσθετα στοιχεία όπως ταχύτητα, υψόμετρο, ώρα αποστολής δεδομένων κ.τ.λ.). Η διεργασία αυτή γίνεται χειροκίνητα (manually) και δεν βρέθηκε τρόπος να ζητείται αυτόματα η αποστολή - έστω αυτών - των δεδομένων, λόγω της δόμησης και της απόκρυψης κάποιων φακέλων από το λειτουργικό σύστημα Windows Mobile 5.0.

Επίσης, παρουσιάζονταν διάφορα προβλήματα σύνδεσης στο GPRS αφού σε δοκιμές που έγιναν για αποστολή/παραλαβή TCP/UDP πακέτων, τα δεδομένα χάνονταν.

Όσο αφορά την άλλη πτυχή, της ανάκτησης τοποθεσιών GPS, αν και η συσκευή υποστηρίζει επιλογή της θύρας εξόδου από την οποία μια εφαρμογή μπορεί συνδεθεί και να διαχειριστεί δεδομένα δεν έγινε κατορθωτή η ανάγνωσή τους. Ο λόγος είναι ότι η συσκευή επιτρέπει να χρησιμοποιούν την θύρα εξόδου δεδομένων, μόνο εφαρμογές - πλοηγοί που είναι εγκατεστημένοι στο σύστημα και όχι ανεξάρτητα προγράμματα. Το συμπέρασμα αυτό εξάχθηκε μετά την χρήση του λογισμικού Java Wireless Toolkit το οποίο προσομοιώνει στον προσωπικό υπολογιστή προγράμματα που είναι γραμμένα για J2ME εφαρμογές. Από την στιγμή που τα προγράμματα ανάκτησης δεδομένων από σειριακές θύρες λειτουργούσαν μέσω του εργαλείου αυτού, δεν υπήρχε λόγος να μην δουλεύουν στην συσκευή η οποία περιλάμβανε τις ίδιες βιβλιοθήκες.

Μετά από συνεννόηση με τον επιβλέποντα καθηγητή, αφού η συσκευή αυτή δεν πρόσφερε τις λειτουργίες με τις οποίες η εφαρμογή μας θα ανέπτυξε επικοινωνία, έγινε αλλαγή πλάνων. Προτάθηκε και χρησιμοποιήθηκε η συσκευή **IntelliTrac X8**.

Η συσκευή αυτή είναι κατασκευασμένη για παρακολούθηση οχημάτων στα οποία είναι εγκατεστημένη και υλοποιεί ένα δικό της πρωτόκολλο εντολών και δίνει την δυνατότητα στον χρήστη, χρησιμοποιώντας τις εντολές αυτές να επικοινωνήσει και ανακτήσει δεδομένα τοποθεσίας. Στην συσκευή εγκαθίσταται μια κάρτα SIM μέσω της οποίας παρέχονται επιλογές επικοινωνίας SMS και GPRS(TCP/UDP). Με την εκτέλεση των εντολών – ανάλογα σε πια κατάσταση είναι η συσκευή (SMS/GPRS) – υπάρχει δυνατότητα ανάκτησης της τρέχουσας τοποθεσίας, καταγραφής διαδρομής, αποθήκευσης στην μνήμη της συσκευής μιας συγκεκριμένης χρονικής διάρκειας διαδρομής και πολλές άλλες λειτουργίες.

Η επιλογή της συσκευής έγινε με γνώμονα την ολοκληρωμένη τεκμηρίωση των εντολών που υλοποιούν το πρωτόκολλο, την εύκολη αλλαγή κατάστασης από SMS σε GPRS και αντίστροφα, καθώς και την αυτόματη και άμεση ανταπόκριση της συσκευής στα αιτήματα. Μετά από μελέτη του πρωτοκόλλου, κατανοήθηκε ότι για την παραλαβή και αποστολή δεδομένων χρειάζεται μια συσκευή τύπου GSM, GSM modem ή αλλιώς ένα κινητό τηλέφωνο που λειτουργεί σαν modem για επικοινωνία μέσω SMS και ένας Server που θα “ακούει” σε μια θύρα για επικοινωνία μέσω GPRS.

Κεφάλαιο 6

Συμπεράσματα

6.1 Μελλοντική Εργασία	67
6.2 Τελικά Συμπεράσματα	69

6.1 Μελλοντική Εργασία

Μετά την εκπόνηση των πρωταρχικών στόχων που τέθηκαν, έγιναν κάποιες σκέψεις για επέκταση των υφιστάμενων λειτουργιών που υλοποιήθηκαν, καθώς και κάποιες προτάσεις για βελτιστοποίηση τόσο στην λειτουργικότητα αλλά και στην παρουσίαση της εφαρμογής και κατ' επέκταση του όλου συστήματος.

Μια αρκετά φιλόδοξη προσπάθεια θα ήταν να μετατραπεί η εφαρμογή σε ένα σύστημα GIS. Κάθε χάρτης – εικόνα να ήταν προσαρμοσμένος με χωρικά δεδομένα και ο χρήστης της εφαρμογής να μπορεί να ενσωματώσει, επεξεργαστεί και αναλύσει γεωγραφικά συσχετισμένες πληροφορίες (geo-referenced). Η υλοποίηση ενός τέτοιου συστήματος θα ήταν από μόνο του ένα μεγάλο εγχείρημα και δεν υπήρξε τάση προς τον τομέα αυτό για το λόγο ότι, δεν θα μπορούσαν να επιτευχθούν οι στόχοι που τέθηκαν από την αρχή λόγω περιορισμένου χρόνου.

Επιπρόσθετα, για να γίνει μια πιο ολοκληρωμένη παρουσίαση και αντιστοίχιση των γεωγραφικών συντεταγμένων σε συντεταγμένες οθόνης σε όσο πιο μικρή κλίμακα χάρτη γίνεται (για παράδειγμα μέχρι κλίμακα οδικών χαρτών), θα ήταν καλό αφού η χαρτογράφηση των δρόμων της Κύπρου είναι σε αρκετά προχωρημένο στάδιο, να

ενσωματωθούν χάρτες καλύτεροι και πληρέστεροι από τους υφιστάμενους οι οποίοι είναι από το διαδίκτυο.

Επίσης, ανάλογα με την GPS συσκευή που χρησιμοποιείται, θα είναι χρήσιμο να υλοποιηθούν περισσότερες λειτουργίες. Συγκεκριμένα, στην εφαρμογή που αναπτύχθηκε, λόγω χρόνου έχουν υλοποιηθεί οι βασικές λειτουργίες ανάκτησης θέσης και λειτουργίες που αφορούν διαδρομές και γενικές ρυθμίσεις.

Υπάρχουν GPS συσκευές που προσφέρουν λειτουργίες όπως, αποστολή μηνύματος συναγερμού σε έκτακτες περιπτώσεις ή ακόμα και μεταφοράς δεδομένων μέσω σειριακής σύνδεσης ή άλλου πρωτοκόλλου επικοινωνίας, για παράδειγμα CS Data Call. Η εφαρμογή όπως έχει αναπτυχθεί και με βάση το λογισμικό ανοικτού κώδικα που χρησιμοποιήθηκε είναι εξ' ολοκλήρου επεκτάσιμη και λόγω του γεγονότος ότι τα κομμάτια που την αποτελούν είναι ανεξάρτητα, η προσθήκη επιπρόσθετων λειτουργιών είναι πλήρως εφικτή.

6.2 Τελικά Συμπεράσματα

Μετά από μήνες εντατικής μελέτης και εργασίας, έχει αναπτυχθεί ένα σύστημα που υλοποιεί τις βασικές λειτουργίες διαχείρισης συσκευών GPS. Γενικότερα, το σύστημα αυτό αναλύει, επεξεργάζεται, παρουσιάζει και διαχειρίζεται πληροφορίες τοποθεσίας που έχουν ανακτηθεί από συσκευές καθώς και παλαιότερες καταχωρήσεις από μια βάση δεδομένων. Επιπρόσθετα μέσω των πρωτοκόλλων SMS και GPRS, έχει γίνει κατορθωτή η επικοινωνία και εγκαθίδρυση σύνδεσης με μια συσκευή GPS (που υποστηρίζει προφανώς τα πρωτόκολλα αυτά). Ουσιαστικά έχει επιτευχθεί η δημιουργία ενός ολοκληρωμένου λογισμικού που παρέχει δυνατότητες διαχείρισης στόλου οχημάτων.

Μέσα από την ανάπτυξη του συστήματος έχουν εξαχθεί κάποια πολύ σημαντικά συμπεράσματα.

Το κυριότερο συμπέρασμα είναι η γνώση της ευρύτερης λειτουργικότητας του GPS και η εμβάθυνση στην λεπτομέρεια που παρέχει η τεχνολογία αυτή. Μέσω της πληροφορίας που δημιουργείται από την κάθε συσκευή, παρέχεται ακριβέστατη και καθορισμένη πληροφορία η οποία μπορεί να επεξεργαστεί και χρησιμοποιηθεί με πάμπολους τρόπους και από διαφορετικού ερευνητικού και καταναλωτικού τομέα αποδέκτες.

Επίσης, μέσα από την εργασία αυτή, δόθηκε η ευκαιρία μελέτης των διαφόρων λύσεων που υπάρχουν στην αγορά και αφορούν συσκευές GPS ως πλοηγοί και ως συσκευές επικοινωνίας. Ο χρόνος που είχαμε να υλοποιήσουμε το επικοινωνιακό κομμάτι για την συσκευή IntelliTrac X8, δεν έδινε περιθώρια για περισσότερη εμβάθυνση σε άλλες λειτουργίες και χρήσεις της συσκευής που σίγουρα προσφέρει. Αντιθέτως, έγινε αρκετή μελέτη και ψάξιμο όσο αφορά την λειτουργία της συσκευής MIO A701 στα πλαίσια της προσπάθειας ανάκτησης δεδομένων αλλά και επικοινωνίας με άλλη συσκευή (ηλεκτρονικό υπολογιστή). Σίγουρα έγινε εμβάθυνση στην λειτουργία των Pocket PC σε λειτουργικό σύστημα Windows Mobile 5.0 , καθώς και μελέτη των υφιστάμενων λογισμικών πλοηγών (navigators) που είναι διαθέσιμοι για τις συγκεκριμένες συσκευές.

Επιπρόσθετα, αποδείχτηκε ότι η γλώσσα προγραμματισμού Java με τις βιβλιοθήκες που την περιβάλλουν, είναι εξαιρετικά ευέλικτη και παρέχει όλα τα πλαίσια εργασίας

που μπορεί μια εφαρμογή να αναπτύξει. Προσφέρει πλαίσια υλοποίησης που έχουν σχέση με αισθητική και λειτουργική παρουσίαση, όπως για παράδειγμα λίστες και πίνακες καθώς και μια αρκετά πλούσια βάση για επικοινωνία και ανταλλαγή πληροφοριών μέσω διαφόρων πρωτοκόλλων, καθώς και αποθήκευση, ανάκτηση και επεξεργασία δεδομένων από/σε μια βάση δεδομένων.

Σίγουρα, όσο αφορά τους χάρτες που περιέχονται στην εφαρμογή – ειδικά τον μεγάλο αρχικό χάρτη της Κύπρου και τον χάρτη της Λευκωσίας – έγινε μεγάλη προσπάθεια για ανεύρεση του κέντρου των συγκεκριμένων πόλεων με δοκιμές και πειραματισμούς. Σε μεγάλο βαθμό αυτό έχει επιτευχθεί αλλά για τον λόγο ότι οι χάρτες έχουν βρεθεί από το διαδίκτυο και υπάρχει πιθανότητα να επήλθαν κάποιες αλλαγές (ειδικά στους οδικούς χάρτες) από την μέρα εισαγωγής τους στην εφαρμογή, δεν μπορούμε να είμαστε σίγουροι 100% για την γεωγραφική ακρίβεια τους.

Σε γενικές γραμμές, μέσα από την Ατομική Διπλωματική Εργασία, έχει γίνει εμβάθυνση σε πολλούς και ανεξάρτητους τομείς προγραμματισμού όπως η ανάπτυξη μιας εφαρμογής GUI, η επικοινωνία και ανταλλαγή πακέτων πληροφοριών με μια συσκευή GPS καθώς και η επεξεργασία και αποθήκευση πληροφοριών σε μια βάση δεδομένων. Οι πιο πάνω βασικοί τομείς ανάπτυξης και υλοποίησης, την χρονική αυτή στιγμή βρίσκονται σε περίοπτη θέση όσο αφορά την χρήση και λειτουργικότητα τους από τους προγραμματιστές λογισμικών. Άρα, η εμπλοκή και ο συνδυασμός όλων αυτών σε ένα σύστημα, σίγουρα απέφερε ένα προϊόν που προσφέρει σίγουρα περισσότερα από τα όσα προσφέρουν οι τομείς αυτοί ανεπτυγμένοι ανεξάρτητα.

Βιβλιογραφία

[1] K. Walrath and M. Campione, “*The JFC Swing Tutorial, A Guide To Constructing GUIs*”, June 1999

[2] Steven J. Gutz, “*Up to Speed With Swing: User Interfaces With Java Foundation Classes*”, 1998

[3] James Elliott, Robert Eckstein, Marc Loy, and David Wood, “*Java Swing, Second Edition*”

[4] Elliotte Rusty Harold , “*Java network programming – The Java Series*”, 1997

[5] S. Ch. Voskarides, C.S. Pattichis, R. Istepanian , C. Michaelides , C.N. Schizas, “*Practical Evaluation of GPRS user in a Telemedicine System in Cyprus*”

[6] IntelliTrac® X Series Hardware Installation Guide

[7] IntelliTrac® X Series Protocol Document

[8] IntelliTrac® X Series Program Manual for Install Wizard

[9] IntelliTrac® X Series Program Manual for Tracer

[10] Support Guide for The Nokia phones and AT-commands

[11] “*ESRI Shapefile Technical Description, An ESRI White Paper*”, July 1998

[12] <http://openmap.bbn.com/> - OpenMap™, Open Systems Mapping Technology

[13] <http://www.tegmento.org/gpsylon/> - GPSylon

[14] <http://javagps.sourceforge.net/> - JavaGPS

[15] <http://triptracker.sourceforge.net/> - TripTracker

[16] <http://www.eit.se/chartplotter/> , Rikard & Henrik Bjorkman's ChartPlotter 0.46

[17] <http://java.sun.com/docs/books/tutorial/> - Java Tutorials

[18] <http://java.sun.com/j2se/1.5.0/docs/api/> - Java™ 2 Platform Standard Edition 5.0 API Specification

[19] <http://mathforum.org/library/drmath/view/68373.html> - Finding the Center of Three Latitude/Longitude Points, Doctor Rick, The Math Forum

[20] “*Network Security for the Windows Mobile Software Platform*”, White Paper, December 2004

[21] “*Network Security for the Windows Mobile Software Platform*”, White Paper, March 2006

[22] Applications for Mobile Information Devices, *Helpful Hints for Application Developers and User Interface Designers using the Mobile Information Device Profile*, Sun

[23] J2ME Building Blocks for Mobile Devices, *White Paper on KVM and the Connected, Limited Device Configuration (CLDC)*, Sun

[24] Dr. Robert Lilley, Gary Church, Michael Harrison, “*GPS Backup For Position, Navigation and Timing, Transition Strategy for Navigation and Surveillance*”, August 22

[25] “*The U.S. Air Force Academy GPS Flight Experiment Using the NAVSYS TIDGET®*”, September 1997

[26] Alison Brown, Ben Mathews and Dien Nguyen, “*Integrated GPS/INS/Star Tracker Space Navigation System Using a Software Defined Radio*”, November. 2006

Παράρτημα Α

Στο παράρτημα αυτό περιγράφονται – αναλύονται όλα τα java αρχεία που περιέχονται στην εφαρμογή ανά package και παρουσιάζονται δείγματα κώδικα όπου είναι αναγκαίο. Οι κλάσεις που είναι σημειωμένες με ‘ * ‘ έχουν δημιουργηθεί εξ’ ολοκλήρου από την αρχή, για να ικανοποιήσουν τις ανάγκες και τον δικό μας σκοπό.

- ***gpstool***

***Gpsylon.java** – Κύριο αρχείο εφαρμογής – Εφαρμογή που εμφανίζει χάρτες, διαδρομές, τοποθεσίες-σημεία συσκευών GPS. Οι πληροφορίες που περιέχονται στην εφαρμογή οργανώνονται και παρουσιάζονται σε layers.

* Στο αρχείο έχουν προστεθεί λειτουργίες που εξυπηρετούν την δική μας εφαρμογή και είναι:

- Δημιουργία του δέντρου-λίστας παρουσίασης των συσκευών που διαχειρίζεται η εφαρμογή
- Δημιουργία και παρουσίαση των τελευταίων γνωστών συντεταγμένων και άλλων πληροφοριών για τις υφιστάμενες συσκευές
- Παρουσίαση των συντεταγμένων κίνησης του ποντικιού στην οθόνη-χάρτη
- Δημιουργία συντομεύσεων για τις πιο συχνά χρησιμοποιούμενες λειτουργίες
- Δημιουργία λειτουργίας εστίασης πάνω σε μια συσκευή – κεντράρισμα χάρτη με βάση την επιλεγμένη συσκευή από την λίστα συσκευών

***GpsylonKeyConstants.java** – Αρχείο που κρατά τα keywords για τις ιδιότητες/ρυθμίσεις της εφαρμογής

- *gpstool.assets*

***AssetsBackup.java** – Εκτέλεση λειτουργίας εφεδρικής αποθήκευσης – μεταφορά όλων των δεδομένων των προηγούμενων μερών σε αποθηκευτικό χώρο ανά συσκευή. Η λειτουργία εξυπηρετεί στην εύκολη ανάκτηση παλιών σημείων – διαδρομών μίας συσκευής.

Δείγμα κώδικα:

```
Class.forName("com.mysql.jdbc.Driver");
connection_ =
DriverManager.getConnection("jdbc:mysql://localhost:3306/gps?user=root&password=root");

    stmt = connection_.createStatement();

    //get NOT last known records from db
    if (stmt.execute("SELECT device_id, latitude, longitude, altitude,
location, speed, date_time, category_id, last_known,device_name FROM POINTS
WHERE last_known=0;")) {
        rs = stmt.getResultSet();
    } else {
        System.out.println("No valid query statement");
    }

    //insert new updated records into respective table
    while (rs.next()) {
        stmt2 = connection_.createStatement();

stmt2.execute("INSERT INTO gps_backup.bck_"+rs.getString(1)+"
(device_id,latitude,longitude,altitude,location,speed,date_time,category_id,
last_known, device_name) VALUES
('"+rs.getString(1)+"','"+rs.getString(2)+"','"+rs.getString(3)+"','"+rs.getStrin
g(4)+"','"+rs.getString(5)+"','"+rs.getString(6)+"','"+rs.getString(7)+"','"+rs
.getString(8)+"','"+rs.getString(9)+"','"+rs.getString(10)+"')");

    }

...
```

***AssetsDetails.java** – Δημιουργία και διαχείριση πίνακα με τα τελευταία γνωστά στοιχεία (αν υπάρχουν) για την κάθε συσκευή που διαχειρίζεται η εφαρμογή. Προσφέρονται οι λειτουργίες προσθήκη, αφαίρεση, αναζήτηση και ενημέρωση των καταχωρήσεων. Με την εκτέλεση των προαναφερθέντων λειτουργιών γίνεται και η ανάλογη ενημέρωση του GUI καθώς της βάσης δεδομένων.

Χρήσιμες μεθόδους:

- *editRecord*: ενημέρωση σημείου πίνακα
- *removeRecord*: διαγραφή σημείου από πίνακα
- *addRecord*: καταχώρηση νέου σημείου στον πίνακα

Δείγμα κώδικα (editRecord): Ενημέρωση επιλεγμένου στοιχείου πίνακα

```
...
/** saves input to the database proper record */
JButton save_button = new JButton("Save");
save_button.setActionCommand(COMMAND_EDIT_SAVE);
save_button.addActionListener(new ActionListener(){

    public void actionPerformed(ActionEvent e) {
        try {
            PreparedStatement statement = connection_
                .prepareStatement("UPDATE POINTS " + "SET device_name= '"
                + name_text_.getText() + "', latitude="
                + latitude_text_.getText() + ", longitude="
                + longitude_text_.getText() + ", altitude="
                + altitude_text_.getText() + ", location='"
                + location_text_.getText() + "'" + ", speed="
                + speed_text_.getText() + ", date_time='"
                + date_time_text_.getText() + "'"
                + " WHERE device_id='" + id_text_.getText()+"' AND
last_known=1");

            statement.execute();

        /** updates table model after editing database record */
        model.setValueAt(name_text_.getText(), selectedRow, 1);
        model.setValueAt(latitude_text_.getText(), selectedRow, 2);
        model.setValueAt(longitude_text_.getText(), selectedRow, 3);
        model.setValueAt(altitude_text_.getText(), selectedRow, 4);
```

```

model.setValueAt(speed_text_.getText(), selectedRow, 5);
model.setValueAt(location_text_.getText(), selectedRow, 6);
model.setValueAt(date_time_text_.getText(), selectedRow, 7);
JOptionPane.showMessageDialog(details_frame_, "Last known point for
device '"+id_text_.getText()+"' succesfully updated.");
...

```

***AssetsList.java** – Δημιουργία και διαχείριση λίστας (δέντρου) που περιέχει τις διαχειριζόμενες από το σύστημα συσκευές. Προσφέρονται οι λειτουργίες προσθήκη, αφαίρεση, αναζήτηση, ενημέρωση και καθορισμός κατάστασης (send/receiver) των συσκευών (popup menu). Με την εκτέλεση των προαναφερθέντων λειτουργιών γίνεται και η ανάλογη ενημέρωση του GUI καθώς της βάσης δεδομένων.

Χρήσιμες μέθοδοι:

- *editSelectedDevice*: ενημέρωση επιλεγμένης συσκευής
- *removeSelectedDevice*: διαγραφή επιλεγμένης συσκευής
- *addNewDevice*: καταχώρηση νέας συσκευής
- *viewDeviceDetails*: παρουσίαση χαρακτηριστικών επιλεγμένης συσκευής

Δείγμα κώδικα (removeSelectedDevice): Διαγραφή επιλεγμένης συσκευής

```

/**
 * Removes the selected device from device list
 * node_ is the selected node from list
 */
public void removeSelectedNode(DefaultMutableTreeNode node_)
{
    if (node_ != null)
    {
        //get the parent of the selected node
        MutableTreeNode parent =
(DefaultMutableTreeNode)(node_.getParent());
        // if the parent is not null
        if (parent != null)
        { //remove the node from the parent
            m_model.removeNodeFromParent(node_);
            tree.updateUI();
        }
    } ...
}

```

***FocusOnDevice.java** – Εστίαση στις γεωγραφικές συντεταγμένες της επιλεγμένης συσκευής

***FindNearestPoint.java** – Εύρεση κοντινότερου γνωστού σημείου σύμφωνα με τις καταχωρημένες τοποθεσίες στην βάση δεδομένων, κατά την προσθήκη νέας τοποθεσίας για μια συσκευή

Δείγμα κώδικα:

```
...
/* find closest point searching through known locations table */
String query = ("SELECT abs(latitude-"+String.valueOf(latitude_)+") +
abs(longitude-"+String.valueOf(longitude_)+") as nearest,
location_name as var, latitude, longitude from gps.locations GROUP by
nearest LIMIT 1;");

    if (stmt.execute(query)){
        rs=stmt.getResultSet();
    }
    else{
        System.out.println("No valid query statement");
    }
    if(rs.next()){
        nearest_=rs.getString(2);

        pos1 = new LatLonPoint(rs.getFloat(3),rs.getFloat(4));
        pos2 = new LatLonPoint(latitude_,longitude_);

        //calculate distance between two points
        distance = Gpsylon.calculateDistance(pos1,pos2);
    }

...
```

- *gpstool.gui*

***CoordsStatusBar.java** – Παρουσίαση συντεταγμένων κίνησης ποντικιού στον χάρτη

Δείγμα Κώδικα:

```
public void mouseMoved(MouseEvent e) {

    int x = e.getX();
    int y = e.getY();
    LatLonPoint llp = null;

    //gets mouse position and translates in decimal
coordinates
    if (e.getSource() instanceof MapBean) {
        llp = ((MapBean)
e.getSource()).getProjection().inverse(x, y);
    }

    //label representing map coords
infoLine = createCoordinateInformationLine(x, y, llp);

coordsLabel.setText(infoLine);
}

...
```

ImageFrame.java – Παρουσίαση και αποθήκευση εικόνας

LatLonInputDialog.java – Παράθυρο που επιτρέπει στον χρήστη να εισάγει γεωγραφικό μήκος και πλάτος

MapKeyHandler.java – Υπεύθυνο για γεγονότα που αφορούν επιλογή κουμπιών

MiniProgressBar.java – Ένα μικρό progress bar που αλλάζει το χρώμα ενός pixel σε περίπτωση αλλαγής μιας κατάστασης

MouseMode.java – Είναι ένας listener ποντικιού που μπορεί να ενεργοποιηθεί/απενεργοποιηθεί από τον χρήστη. Για παράδειγμα το ZoomMouseMode εστιάζει με αρνητικό ή θετικό συντελεστή στο σημείο που κάνει κλικ ο χρήστης. Επίσης, το CalculatorMouseMode υπολογίζει αποστάσεις μεταξύ δύο διαδοχικών κλικ του ποντικιού σε επιλεγμένα μέρη του χάρτη

MouseModeManager.java – Διαχειρίζεται τα mouse modes

NmeaDataTextFrame.java – Χρησιμοποιείται για επίβλεψη των μη κατεργασμένων εισερχόμενων gps δεδομένων

SatelliteActivity.java – Παρουσιάζει τις δραστηριότητες των δορυφόρων, σε μορφή δύναμης εκπομπής σήματος παρουσιάζοντας σε στήλες το ποσοστό εκπομπής κάθε δορυφόρο

***ShortCutBar.java** – Υλοποιεί ένα μενού με τις πιο συχνά χρησιμοποιούμενες λειτουργίες στο πάνω μέρος της εφαρμογής (διαχείριση συσκευών, σημείων GPS κτλ.)

- **gpstool.gui.layer**

***CommunicationLayer.java** – Επίπεδο που δίνει τη δυνατότητα στον χρήστη να διαχειριστεί διάφορες SMS ή GPRS συνδέσεις με μια συσκευή

Χρήσιμες Υποκλάσεις:

smsStartTrackingGPSdeviceAction: Καταγραφή διαδρομής συσκευής (tracking) μέσω SMS

smsStopTrackingGPSdeviceAction: Ακύρωση καταγραφής διαδρομής συσκευής (stop tracking) μέσω SMS

gprsStartTrackingGPSdeviceAction: Καταγραφή διαδρομής συσκευής (tracking) μέσω GPRS

gprsStopTrackingGPSdeviceAction: Ακύρωση καταγραφής διαδρομής συσκευής (stop tracking) μέσω GPRS

smsGetPosGPSdeviceAction: Ανάκτηση τρέχουσας τοποθεσίας συσκευής μέσω SMS

smsEnableGPRSAction: Ενεργοποίηση GPRS επικοινωνίας μέσω SMS

gprsDisableGPRSAction: Απενεργοποίηση/καθορισμός νέων παραμέτρων GPRS επικοινωνίας μέσω GPRS

gprsGetPosGPSdeviceAction: Ανάκτηση τρέχουσας τοποθεσίας συσκευής μέσω GPRS

smsStartLogAction: Αποθήκευση διαδρομής συσκευής (logging) στην μνήμη της συσκευής μέσω SMS

smsStopLogAction: Ακύρωση αποθήκευσης διαδρομής συσκευής (stop logging) στην μνήμη της συσκευής μέσω GPRS

smsClearLogAction: Διαγραφή αποθηκευμένων διαδρομών (clear logging) στην μνήμη της συσκευής μέσω SMS

gprsStartLogAction: Αποθήκευση διαδρομής συσκευής (logging) στην μνήμη της συσκευής μέσω GPRS

gprsStopLogAction: Ακύρωση αποθήκευσης διαδρομής συσκευής (stop logging) στην μνήμη της συσκευής μέσω SMS

gprsGetLogAction: Ανάκτηση αποθηκευμένων διαδρομών (get log data) από την μνήμη της συσκευής στον υπολογιστή μέσω GPRS

gprsGetSelectedLogAction: Ανάκτηση επιλεγμένων (κατά ημερομηνία) αποθηκευμένων διαδρομών (get selected log data) από την μνήμη της συσκευής στον υπολογιστή μέσω GPRS

gprsCancelLogAction: Ακύρωση ανάκτησης αποθηκευμένων διαδρομών από την μνήμη της συσκευής στον υπολογιστή μέσω GPRS

gprsClearLogAction: Διαγραφή αποθηκευμένων διαδρομών (clear logging) στην μνήμη της συσκευής μέσω GPRS

setBaseSTsettingsAction: Καθορισμός παραμέτρων τηλεφώνου σταθμού βάσης (ο αριθμός που θα στέλνει η συσκευή τα δεδομένα όταν είναι σε SMS κατάσταση) μέσω SMS

setDeviceNewPassAction: Καθορισμός συνθηματικού (password) συσκευής μέσω SMS

setDevicePINAction: Καθορισμός PIN της κάρτας που βρίσκεται στην συσκευή μέσω SMS

* Οι πιο πάνω λειτουργίες, συνοδεύονται πάντα με ένα frame/dialog μέσω του οποίου ο χρήστης έχει την δυνατότητα εισαγωγής των επιθυμητών παραμέτρων για κάθε εντολή. Με την ενεργοποίηση της εντολής, η εφαρμογή συνδέεται με τον UDP Receiver και αποστέλλει τις πληροφορίες για να σταλούν στην συνέχεια στην ανάλογη συσκευή GPS.

****DestinationLayer.java*** – Επίπεδο που έχει την δυνατότητα υπολογισμού απόστασης μεταξύ δύο επιλεγμένων σημείων. Το αρχείο έχει μεταβληθεί ανάλογα για να εξυπηρετεί τις ανάγκες της δικής μας υλοποίησης. Με την ενεργοποίηση του επιπέδου αυτού με ένα πάτημα του αριστερού κουμπιού του ποντικιού επιλέγεται το αρχικό σημείο και με δεύτερο πάτημα επιλέγεται το δεύτερο σημείο (προορισμός) για υπολογισμό απόστασης σε χιλιόμετρα μεταξύ τους.

GraticuleLayer.java – Επίπεδο που παρουσιάζει σταυρόνημα στον χάρτη με τα γεωγραφικό μήκος και πλάτος.

****LocationLayer.java*** – Επίπεδο που παρουσιάζει – διαχειρίζεται τοποθεσίες των συσκευών GPS που είναι διαθέσιμες στο σύστημα

Έχουν γίνει επιπρόσθετες αλλαγές και διαρρυθμίσεις των υφιστάμενων υποκλάσεων καθώς και καταχώρηση νέων λειτουργιών που είναι:

LoadMapAction: Καταχώρηση νέου χάρτη στην εφαρμογή επιλέγοντας τις πληροφορίες που τον χαρακτηρίζουν (γεωγραφικό μήκος και πλάτος του κέντρου του χάρτη, κλίμακα, διαστάσεις)

SetDeviceNewGPSPositionAction - Καταχώρηση νέου σημείου για μια συσκευή

RemovePointAction - Διαγραφή τελευταίου γνωστού σημείου συσκευής

EditPointAction - Ενημέρωση τελευταίου γνωστού σημείου συσκευής

MultimapLayer.java – Επίπεδο που παρουσιάζει χάρτες σε διάφορες κλίμακες ανάλογα με το zoom level

PositionLayer.java – Επίπεδο που παρουσιάζει στον χάρτη την θέση μιας συνδεδεμένης συσκευής με την εφαρμογή

ScaleLayer.java – Επίπεδο που παρουσιάζει την υφιστάμενη κλίμακα και ρίγα παραλληλισμού αποστάσεων

ShapeLayer.java – Επίπεδο που παρουσιάζει σχήματα σημείων

- *gpstool.gui.layer.location*

AbstractLocationMarkerSource.java – Αποθηκεύει τα τελευταία γνωστά σημεία στην μνήμη

CategoryCellRenderer.java – Ιδιότητες κατηγορίας σημείου (παρουσίαση εικόνας κτλ.)

***JDBCLocationPointSource.java** – Υπεύθυνο για ανάκτηση καταχωρήσεων από την MySQL βάση δεδομένων

Έχουν γίνει επιπρόσθετες αλλαγές και διαρρυθμίσεις των υφιστάμενων υποκλάσεων καθώς και καταχώρηση νέων λειτουργιών που είναι:

- Ανάκτηση τελευταίων γνωστών σημείων για κάθε GPS συσκευή
- Αλλαγή στα queries της βάσης (μετατροπή σε MySQL από Apache Derby)
- Τοποθέτηση στον χάρτη των ανακτηθέντων σημείων

Δείγμα κώδικα(ανάκτηση τελευταίου γνωστού σημείου για κάθε συσκευή από B.A):

```
public GeoScreenList getLocationMarkers(float north, float south,
                                       float west, float east,
                                       GeoScreenList location_list)
{
    ...
    ...

    try
    {
        PreparedStatement prst = connection_.prepareStatement("SELECT
device_id,latitude,longitude,category_id FROM points WHERE latitude <
? AND latitude > ? AND longitude > ? AND longitude < ? AND
last_known=1 ORDER BY device_id");
        prst.setFloat(1,north);
        prst.setFloat(2,south);
        prst.setFloat(3,west);
        prst.setFloat(4,east);

        ResultSet rs = prst.executeQuery();
        if(rs != null)
        {
            while(rs.next())
            {
                //insert device to map
                location_list.add(createLocationMarker(rs));
            }
        }
        ...
        ...
    }
```

JDBCPreparedStatementPart.java – Κατακρατεί την σειρά χαρακτήρων καθώς και τα αντικείμενα – σημεία που θα εισαχθούν. Χρησιμοποιείται από την πιο πάνω κλάση.

****JDBCUtil.java*** - Χρησιμοποιείται ως ένα αφαιρετικό επίπεδο για διαφορετικές SQL βάσεις δεδομένων

Δείγμα κώδικα:

```
String jdbc_driver = "com.mysql.jdbc.Driver";  
String jdbc_url = "jdbc:mysql://localhost:3306/";  
String jdbc_username = "root";  
String jdbc_password = "root";  
...
```

LocationPoint.java – Αναπαράσταση σημείου με συντεταγμένες και γεωγραφική τοποθεσία.

LocationPointCategory.java – Αναπαράσταση κατηγορίας σημείου

LocationPointCellRenderer.java – Ιδιότητες κατηγορίας σημείου (παρουσίαση εικόνας κτλ.)

LocationPointFilter.java – Χρησιμοποιείται ως φίλτρο για τα gps σημεία

****LocationPointFrame.java*** – Πλαίσιο επιλογής μοναδικού αναγνωριστικού, ονόματος, συντεταγμένων, κατηγορίας και άλλων πληροφοριών που αφορούν μια τοποθεσία.

Έχουν γίνει επιπρόσθετες αλλαγές και διαρρυθμίσεις για την κάλυψη των αναγκών μας όπως η καταχώρηση περισσότερων χαρακτηριστικών ενός σημείου π.χ. υψόμετρο, τοποθεσία, ώρα καταχώρησης δεδομένων κτλ.

LocationPointSearchPanel.java - Αναζήτηση συσκευής – τελευταίου γνωστού σημείου στον χάρτη και εστίαση στην συσκευή μετά από την επιλογή της από τον χρήστη.

LocationPointSource.java – Υλοποιήσεις αυτού του interface είναι πηγές για LocationPoints (βάση δεδομένων).

LocationPointSourceException.java – exception

SearchLocationPointFrame.java – Παράθυρο αναζήτησης

SelectCategoryFrame.java – Επιλογή κατηγορίας συσκευής

- ***gpstool.gui.util***

AngleCellRenderer.java – έλεγχος σωστής μορφής γεωγραφικού μήκους και πλάτους

AngleJTextField.java – JTextField αντικείμενο που χρησιμοποιεί τις επιτρεπόμενες μορφές συντεταγμένων για εισαγωγή και έλεγχο στο σύστημα

BasicLayer.java – Επιφάνεια στην εφαρμογή που είναι υπεύθυνη να ενημερώνει για αλλαγές στην παρουσίαση των γεωγραφικών δεδομένων σε περίπτωση οπτικών αλλαγών (π.χ. zoom in, zoom out κτλ.)

ImageInfo.java – Κρατά πληροφορίες για τις εικόνες και το ορατό τους μέρος

VisibleImage.java – Βρίσκει τα κομμάτια μιας εικόνας – χάρτη που είναι ορατά

- ***gpstool.hook***

HookManager.java – Παρέχει όλα τα διαθέσιμα hooks και συνθετικά της εφαρμογής τα οποία χρειάζονται όλες οι λειτουργίες για να δουλέψουν ορθά

MapManagerHook.java – Αφορά hooks που διαχειρίζονται χάρτες

MapNavigationHook.java – Αφορά hooks που διαχειρίζονται την πλοήγηση στους

- ***gpstool.map***

MapInfo.java – Κρατεί πληροφορίες για τους χάρτες που χρησιμοποιούνται από την εφαρμογή (συντεταγμένες κέντρου, μέγεθος, κλίμακα κτλ.)

MapManager.java – Υπεύθυνο για την παρουσίαση χαρτών, διαδρομών, τοποθεσιών. Έχει γίνει μια αλλαγή στις ρυθμίσεις έτσι ώστε να χρησιμοποιείται σαν κέντρο του χάρτη το (υπολογιζόμενο) κέντρο της Κύπρου και η συγκεκριμένη εικόνα της Κύπρου που έχει επιλεγεί.

Δείγμα κώδικα:

```
URL cyprusmap_url =  
getClass().getClassLoader().getResource("cyprus.gif");  
float scale = 480000.0f;  
MapInfo map_info = new MapInfo(cyprusmap_url, 35.1292, 33.4245,  
scale, 1280, 800);  
map_infos.add(map_info);
```

- ***gpstool.projection***

FlatProjection.java – Υλοποιεί την παρουσίαση των χαρτών χρησιμοποιώντας αλγορίθμους από την εφαρμογή Fritz Ganter's gpsdrive. Περιέχει δύο πολύ σημαντικές και συχνά χρησιμοποιούμενες μεθόδους, την **inverse** που μετατρέπει τις συντεταγμένες οθόνης σε γεωγραφικό μήκος και πλάτος και η **forward** που υλοποιεί το αντίστροφο.

- ***gpstool.simulation***

****SimulateTracks.java*** – Αυτή η κλάση υλοποιεί ένα thread που προσομοιώνει την διαδρομή μιας συσκευής πάνω στον χάρτη σύμφωνα με συνεχόμενες αποθηκευμένες τοποθεσίες

Δείγμα κώδικα:

```
...
//gets records from current days table
rs_temp= stmt.executeQuery("SELECT latitude, longitude, altitude,
speed, location,
date_time,num,device_id,category_id,last_known,device_name FROM
gps.points s WHERE
device_id='"+ids_box_.getSelectedItem().toString()+"' AND date_time
BETWEEN '"+from_date_text_.getText()+"' AND
 '"+to_date_text_.getText()+"'");

//gets records from backup table
rs = stmt2.executeQuery("SELECT latitude, longitude, altitude, speed,
location, date_time,num,device_id,category_id,last_known,device_name
FROM gps_backup.bck_"+ids_box_.getSelectedItem().toString()+" s WHERE
date_time BETWEEN '"+from_date_text_.getText()+"' AND
 '"+to_date_text_.getText()+"'");
...
//execute thread
while(rs_.next())
    try
    {
        Thread.sleep (2000);
    }
    catch (InterruptedException e) {
        System.out.println("PlaybackLayer-InterruptedException:
"+e.getMessage());
    }

    try{
        point_ = new
        LocationPoint(id_,getDeviceName(id_),rs_.getFloat(1),rs_.getFlo
at(2),rs_.getFloat(3),rs_.getFloat(4),"Auto",rs_.getString(6),c
ategory_);

        LatLonPoint centerPoint = new LatLonPoint(35.1292, 33.4245);
        //center of the map
        LatLonPoint pointOnMap = new
        LatLonPoint(rs_.getFloat(1),rs_.getFloat(2)); //position of
device on map

        LLXY llxy = new LLXY(centerPoint, 480000.0f, 1280,800);
        Point point = llxy.forward(pointOnMap);

        point_.setX(point.x);
        point_.setY(point.y);
    }
```

```

        details_.addRecord(point_);
        try {
            Thread.sleep(1000);
        } catch (InterruptedException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        details_.removeRecord(id_);
        location_layer_.addNewLocationPoint(point_);
    ...
    ...

```

- ***gpstool.util***

ExtentionFileFilter.java – Επέκταση FileFilter.java

FileUtil.java – Παρέχει βοηθητικές λειτουργίες για διαχείριση αρχείων

GeoExtend.java – Είναι μια γεωγραφική επέκταση που αντιπροσωπεύεται σε Βορρά, Νότο, Ανατολή και Δύση

GeoMath.java – Παρέχει κάποιους υπολογισμούς για τα γεωγραφικό μήκος και πλάτος

GeonetDataConverter.java – Διαβάζει geonet αρχεία και τα τροποποιεί ανάλογα με τις απαιτήσεις σε διάφορες μορφές

GeonetStreamTokenizer.java – Επιτρέπει να διαβάζονται tab-seperated streams, αρχεία κτλ.

****JDBCUtil.java*** – Αφαιρετικό επίπεδο για διαφορετικές SQL βάσεις δεδομένων. Στην κλάση αυτή έχουν γίνει διαφοροποιήσεις έτσι ώστε όταν ο χρήστης ανοίγει για πρώτη φορά την εφαρμογή να δημιουργείται η νέα MySQL βάση δεδομένων.

MapInfoScaleComparator.java – Συγκρίνει δύο MapInfo με πρώτο κριτήριο την κλίμακα τους

Positionable.java – Interface που χρησιμοποιείται στην ανάκτηση της τρέχουσας gps θέσης

UnitHelper.java - Βοηθητική κλάση που επιτρέπει σε όλες τις διεργασίες να χρησιμοποιούν το ίδιο format για την ταχύτητα, γεωγραφικό μήκος και πλάτος κτλ.

- ***gpstool.util.angle***

Angle.java – Μια γωνιά σε μοίρες που αντιπροσωπεύει την περιστροφή μιας γραμμής έτσι ώστε να συμπίπτει με μια άλλη

AngleFormat.java – Το format των γωνιών που χρησιμοποιούνται σύμφωνα με ένα pattern

Latitude.java – Μια γωνιά γεωγραφικού μήκους

Longitude.java – Μια γωνιά γεωγραφικού πλάτους

XMath.java – Απλές μαθηματικές συναρτήσεις

- ***gpstool.util.geoscreen***

-

GeoScreen.java – Interface που αντιπροσωπεύει ένα γεωγραφικό σημείο και την παρουσίαση του σε συντεταγμένες οθόνης

GeoScreenList.java – Αντιπροσωπεύει μια λίστα από GeoScreen αντικείμενα

GeoScreenPoint.java - Αντιπροσωπεύει ένα γεωγραφικό σημείο και την παρουσίαση του σε συντεταγμένες οθόνης

- ***gpstool.util***

Debug.java – Βοηθητική κλάση για εκτύπωση μηνυμάτων αποσφαλμάτωσης

PrintUtilities.java – Απλή κλάση για εκτύπωση χρήσιμων αφαιρετικών μηνυμάτων

ProgressListener.java – Listener που ενημερώνεται για την πρόοδο λήψης/καταγραφής δεδομένων από gps συσκευές ή αρχεία

Resources.java – Interface που έχει πρόσβαση σε όλες τις πηγές – ρυθμίσεις της εφαρμογής. Οι πηγές μπορεί να είναι αρχεία, βάση δεδομένων, κύρια μνήμη κτλ

ResourceManager.java – Διαχείριση της πιο πάνω κλάσης – Φορτώνεται στην εφαρμογή όταν χρειάζεται ένα συγκεκριμένο αποθηκευμένο αντικείμενο

- ***gpstool.util.debug***

- Πακέτο που είναι υπεύθυνο (μέσα από τις περιεχόμενες κλάσεις) για την εκτύπωση μηνυμάτων λάθους και αποσφαλμάτωσης

- ***gpstool.util.gui***

ActionGenerator.java – Δημιουργία action ανάλογα με την αίτηση του gui της εφαρμογής

ActionStore.java – Αποθηκεύει όλες τις ενέργειες που δημιουργούνται κατά την εκκίνηση της εφαρμογής και αντιστοιχούν σε λειτουργίες μενού, παραθύρων, κουτιών διαλόγου κτλ.

LoginDialog.java – Εισαγωγή και έλεγχος συνθηματικών εισόδου (δεν έχει υλοποιηθεί πλήρως)

MenuFactory.java – Δημιουργεί το menu bar της εφαρμογής ανακτώντας όλες τις ανάλογες λεπτομέρειες από τα resources

ResourceEditorFrame.java – Παράθυρο διαχείρισης ρυθμίσεων εφαρμογής

SelectedButtonActionSynchronizer.java – Συγχρονισμός επιλογών κουμπιών (radio buttons, check marks κτλ.) και ανάλογων ενεργειών

SplashScreen.java – Οθόνη εκκίνησης εφαρμογής που παρουσιάζει χρονικά την φόρτωση των λειτουργιών

- **gpstool.util.io**

FileChangeDetection.java – Κλάση που ελέγχει ένα δοσμένο αρχείο και ελέγχει οποιαδήποτε αλλαγή που προκύπτει

FileChangeListener.java – Interface που ενημερώνεται για αλλαγές σε αρχεία

Tokenizer.java – Προσφέρει χαμηλού και ψηλού επιπέδου interface στα tokenizer

- **gpstool.util.resource**

AbstractRecourses.java – Αφαιρετική κλάση που αντιπροσωπεύει και διαχειρίζεται τα δεδομένα που βρίσκονται στα resources της εφαρμογής

FileResources.java – Παρέχει την λειτουργικότητα στην εφαρμογή να διαβάζει resources από το σύστημα ή ένα αρχείο πηγών του χρήστη

ResourceGroup.java – Αναγνώριση group πηγών

- *gpstool.util.servicediscovery*

RepositoryClassLoader.java – Επιτρέπει την φόρτωση εξωτερικών jar αρχείων – βιβλιοθηκών

ServiceDiscovery.java – Υλοποίηση του JDK 1.3 ‘Service Provider’

ServiceInfo.java – Υπηρεσία που βρίσκεται στην κλάση ServiceDiscovery

- *udp*

UDPClient.java – Υλοποιεί τον client στο μοντέλο client-server στο πρωτόκολλο UDP. Σε αρκετές περιπτώσεις έχει την ευθύνη επεξεργασίας και ανάλυσης των εισερχόμενων δεδομένων καθώς και την παρουσίαση στον χάρτη σε περιπτώσεις που οι πληροφορίες περιέχουν δεδομένα τοποθεσίας από συσκευές GPS.

Δείγμα κώδικα (σύνδεση μέσω SMS ή GPRS για ανάκτηση τοποθεσίας):

```
public void sms_gprs_Connection(){
    try{
        DatagramSocket aSocket = new DatagramSocket();

        byte []m = content_.getBytes();
        InetAddress aHost = InetAddress.getByName(inet_address_);

        int serverPort=Integer.valueOf(port_);
        DatagramPacket request = new DatagramPacket(m,
content_.length(), aHost, serverPort);
        aSocket.send(request);

        byte [] buffer = new byte[1000];
        DatagramPacket reply = new DatagramPacket(buffer,
buffer.length);

        aSocket.receive(reply);

        String[] values=new String[20];

        //analyse incoming data
        String data_= new String(reply.getData(),"ASCII").trim();
```

...ακολουθεί κώδικας που επεξεργάζεται και παρουσιάζει στον χάρτη τα εισερχόμενα δεδομένα τοποθεσίας

UDPConnection.java – Υπεύθυνη για σύνδεση και αποσύνδεση στον UDP Receiver καθώς και για την ανάκτηση του πίνακα καταστάσεων για τις διαχειριζόμενες συσκευές.

Βοηθητικές κλάσεις:

- *ViewServerStatusAction*
- *LoginServerAction*
- *LogoutServerAction*

UDPLoginFrame.java – πλαίσιο εισαγωγής πληροφοριών πρόσβασης στον UDP Server

Επίσης χρησιμοποιούνται τα αρχεία:

Gpsylon.properties – περιέχει όλα τα δεδομένα ρυθμίσεων για την λειτουργία της εφαρμογής

create_location_mysql.sql – περιέχει τις SQL εντολές δημιουργίας πινάκων στην βάση δεδομένων αν αυτοί δεν υπάρχουν

Επιπρόσθετα, εσωκλείεται ένας φάκελος που περιέχει τις εικόνες που χρησιμοποιούνται στην εφαρμογή.

Οι βιβλιοθήκες που χρησιμοποιούνται είναι οι εξής:

- comm.jar
- RXTXcomm.jar
- gnu-regexp-1.1.3.jar
- log4j-1.2.8.jar
- mysql-connector-java-5.0.3-bin.jar
- openmap.jar velocity-dep-1.3.1.jar
- batik-gui-util.jar

Παράρτημα Β

Στο παράρτημα αυτό περιγράφονται – αναλύονται όλα τα java αρχεία που περιέχονται στον UDP Receiver και παρουσιάζονται δείγματα κώδικα όπου είναι αναγκαίο. Να σημειωθεί ότι τα σχετικά αρχεία υλοποιήθηκαν εξ' ολοκλήρου από την αρχή.

- *receiver*

UDP_Receiver.java – Αφαιρετική κλάση υλοποίησης ενός UDP Server ο οποίος δέχεται και απαντά αιτήματα που αφορούν διαχείριση χαρακτηριστικών συσκευών. Επίσης έχει την ευθύνη ελέγχου και αποστολής μηνυμάτων SMS.

Δείγμα Κώδικα:

```
.....
//create table with each manipulated device fields
createReceiverTable();

try{
    aSocket = new DatagramSocket(port);

    while(true){
        System.out.println("Receiver listening...");

        buffer = new byte[200];
        owner_data = new String[30];

        request = new DatagramPacket(buffer,
buffer.length);

        //read received data
        req_data=receive(request);

//requests concerning connection initialization and unit commands)
        if(.....){
            processPacket(owner_data);
            System.out.println("ClientUnit: "+ request.getAddress());
        }
        //requests concerning device manipulation
        else{
            processUserRequest(owner_data);
            System.out.println("ClientUser: "+ request.getAddress());
        }
    }
}
.....
```

```

protected String receive(DatagramPacket request){
    try{
        aSocket.receive(request);

        reply = new DatagramPacket(new
byte[200], request.getLength(), request.getAddress(), request.getPort())
;

        address_=String.valueOf(request.getAddress());
        port_=String.valueOf(request.getPort());

        //decoding request data
        req_data = new String(request.getData(), "ASCII");

        return req_data;
    }
}

```

Χρήσιμες μέθοδοι

start: Εκκίνηση λειτουργίας του UDP Receiver δημιουργώντας τον πίνακα χαρακτηριστικών συσκευών και ανοίγοντας το port για παραλαβή αιτημάτων από χρήστες και συσκευές.

stop: Τερματισμός λειτουργίας UDP Receiver.

receive: Παραλαβή και αποκωδικοποίηση δεδομένων πληροφορίας – μετατροπή bytes σε πίνακα χαρακτήρων για επεξεργασία.

processUserRequest: Επεξεργασία αιτήματος από χρήστη που αφορά διαχείριση συσκευών (καταχώρηση, διαγραφή, ενημέρωση) ή αναφορά κατάστασης συσκευών (IP address, port)

processPacket: Κλήση μεθόδου analyse (αφορά πακέτα που σχετίζονται με την αμφίδρομη επικοινωνία με συσκευές)

analyse: Επεξεργασία αιτήματος από χρήστη που αφορά επικοινωνία (SMS/GPRS) και αποστολή εντολής - ανάκτηση πληροφορίας από τις συσκευές.

X8Receiver.java – Κλάση υλοποίησης ενός UDP Server ο οποίος έχει την ευθύνη επικοινωνίας και ανταλλαγής μηνυμάτων με την συσκευή Intellitrac X8. Υλοποιεί τις μεθόδους της πιο πάνω κλάσης. Επίσης παρέχει διαχείριση των χρηστών του συστήματος (καταχώρηση και διαγραφή χρηστών).

Δείγμα Κώδικα (αίτημα ανάκτησης τρέχουσας θέσης μέσω GPRS):

```
if( (owner_data[0].compareTo(GPRS_GET_POSITION_COMMAND)==0) &&
(owner_data[3].compareTo(GPRS)==0)){

//define command format
String getPosition=new
String(GPRS_GET_POSITION_COMMAND+COMMAND_EQUALS+owner_data[1]);

    byte[] b=new byte[128];

    b=getPosition.getBytes();
//send request to device
    DatagramPacket req_unit = new
DatagramPacket(b,b.length,getDeviceIP(owner_data[2]),getDevicePort(ow
ner_data[2]));

    aSocket.send(req_unit);

    //wait for answer
    Thread.sleep(5000);

    byte[] buffer=new byte[128];

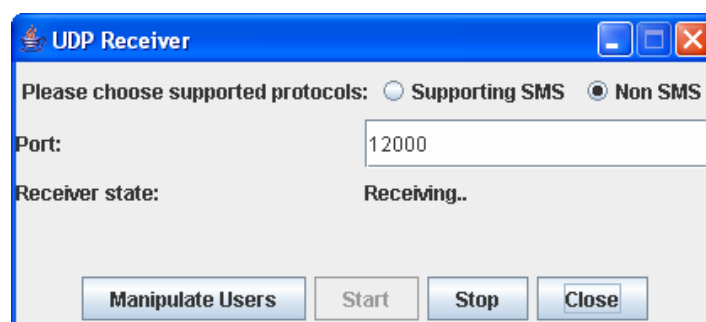
    //receiver answer from device
    DatagramPacket request_data = new DatagramPacket(buffer,
buffer.length);
    aSocket.receive(request_data);

    //set incoming data stream for sendingback to user
    reply.setData(request_data.getData());

    resumeSMSThread();

    aSocket.send(reply);
}
```

Εικόνα Β.1



- **sms**

ReadMessagesAsync.java για παραλαβή και διάβασμα περιεχομένου ενός SMS

SendMessage.java για αποστολή ενός SMS

Δείγμα κώδικα:

```
...
//initialize SMS service with connected mobile device
CService srv = new CService("COM7", 460800, "Nokia 6230i", "");

System.out.println("SendMessage(): Send a message.");
System.out.println(" Using " + CService._name + " " +
CService._version);

    try
    {

//set mobile device parameters so connection can be established
        srv.setSimPin("3313");

        srv.setSmscNumber("+35799700000");

        srv.connect();
    }
...

```

- **Smslib & smslib.handler**

Περιέχουν τις βιβλιοθήκες που υλοποιούν τις δύο πιο πάνω κλάσεις.

Οι βιβλιοθήκες που χρησιμοποιούνται είναι οι εξής:

- comm.jar
- RXTXcomm.jar
- gnu-regexp-1.1.3.jar
- log4j-1.2.8.jar
- mysql-connector-java-5.0.3-bin.jar

Παράρτημα Γ

Στο παράρτημα αυτό δίνονται οι εντολές που πρέπει να αποσταλούν στην συσκευή IntelliTrac X8 για υλοποίηση των λειτουργιών που έχουν αναπτυχθεί.

- Ενεργοποίηση GPRS επικοινωνίας μέσω SMS = "\$ST+GPRS"
- Απενεργοποίηση/καθορισμός νέων παραμέτρων GPRS επικοινωνίας μέσω GPRS = "\$ST+GPRS"
- Ανάκτηση τρέχουσας τοποθεσίας συσκευής μέσω SMS = "\$ST+GPOS"
- Ανάκτηση τρέχουσας τοποθεσίας συσκευής μέσω GPRS = "\$ST+GETPOSITION"
- Καταγραφή διαδρομής συσκευής (tracking) μέσω SMS = "\$ST+TRAC"
- Καταγραφή διαδρομής συσκευής (tracking) μέσω GPRS = "\$ST+TRACKING"
- Ακύρωση καταγραφής διαδρομής συσκευής (stop tracking) μέσω SMS = "\$ST+STOP"
- Ακύρωση καταγραφής διαδρομής συσκευής (stop tracking) μέσω GPRS = "\$ST+STOPTRACKING"
- Αποθήκευση διαδρομής συσκευής (logging) στην μνήμη της συσκευής μέσω SMS = "\$ST+STARTLOG"
- Αποθήκευση διαδρομής συσκευής (logging) στην μνήμη της συσκευής μέσω GPRS = "\$ST+STARTLOG"
- Ακύρωση αποθήκευσης διαδρομής συσκευής (stop logging) στην μνήμη της συσκευής μέσω SMS = "\$ST+STOPLOG"
- Ακύρωση αποθήκευσης διαδρομής συσκευής (stop logging) στην μνήμη της συσκευής μέσω GPRS = "\$ST+STOPLOG"
- Διαγραφή αποθηκευμένων διαδρομών (clear logging) στην μνήμη της συσκευής μέσω SMS = "\$ST+CLEARLOG"

- Διαγραφή αποθηκευμένων διαδρομών (clear logging) στην μνήμη της συσκευής μέσω GPRS = "\$ST+CLEARLOG"
- Κατέβασμα αποθηκευμένων διαδρομών (get log data) από την μνήμη της συσκευής στον υπολογιστή μέσω GPRS = "\$ST+GETLOG"
- Ακύρωση κατεβάσματος αποθηκευμένων διαδρομών από την μνήμη της συσκευής στον υπολογιστή μέσω GPRS = "\$ST+CANCELLOG"
- Κατέβασμα επιλεγμένων (κατά ημερομηνία) αποθηκευμένων διαδρομών (get selected log data) από την μνήμη της συσκευής στον υπολογιστή μέσω GPRS = "\$ST+GETLOGSEL"
- Καθορισμός παραμέτρων τηλεφώνου σταθμού βάσης (ο αριθμός που θα στέλνει η συσκευή τα δεδομένα όταν είναι σε SMS κατάσταση) μέσω SMS= "\$ST+SBAS"
- Καθορισμός PIN της κάρτας που βρίσκεται στην συσκευή μέσω SMS = "\$ST+CPAS"
- Καθορισμός συνθηματικού (password) συσκευής μέσω SMS = "\$ST+SPIN"

Παράρτημα Δ

Στο παράρτημα αυτό περιγράφονται οι εκτελέσιμες εντολές για την λειτουργία του συστήματος καθώς και οι προδιαγραφές που πρέπει να πληρεί μια μηχανή για να υποστηρίζεται η λειτουργία αυτή.

Για εκτέλεση της εφαρμογής:

```
java -jar gpsylon.jar
```

Για εκτέλεση του UDP Receiver:

```
java -jar receiver.jar
```

Αναγκαίες προϋποθέσεις για λειτουργία του συστήματος:

- Έκδοση Java 1.4 (ή πιο πρόσφατη) είναι αναγκαία. Μπορεί να ανακτηθεί από: [Sun's Website](#). Η έκδοση Java 1.3 δεν είναι λειτουργίσιμη!
- Βάση δεδομένων MySQL, ανοικτή για διαχείριση στην θύρα 3306 – καλό να δημιουργηθούν τα schemata gps και gps_backup. Αν τα σχήματα δεν υπάρχουν με την πρώτη λειτουργία του συστήματος δημιουργούνται αυτόματα.