



Τ.Ε.Ι. Δυτικής Ελλάδας  
Τμήμα Μηχανικών Πληροφορικής Τ.Ε

# Ανάπτυξη εφαρμογής τηλεμετρίας σε συσκευές κινητής τηλεφωνίας

---

**Επιβλέπων καθηγητής:** Ασημακόπουλος Γιώργος

**Φοιτητές:** Λυμέρης Βησσαρίων (Α.Μ. 0762)

Ιωαννίδης Δημήτριος (Α.Μ. 0224)

*Αντίρριο 9/7/2014*

---

## Περιεχόμενα

Περίληψη	2
Εισαγωγή	3
Σκοπός εργασίας	3
Τηλεμετρία	
Ορισμός	5
Υλικό και λογισμικό του Συστήματος	
Απομακρυσμένες συσκευές	6
Υλικό της Συσκευής	7
Αρχιτεκτονική Λογισμικού	9
Λογισμικό του Server	9
Λογισμικό στον Client	11
Αρχιτεκτονική 3-Tier	11
Επεξεργασία δεδομένων	12
Μεθοδολογία	14
Ανάλυση Απαιτήσεων	15
Ανάλυση Συστήματος	16
Ανάλυση Δεδομένων	17
Σχεδιασμός	
Σχεδιασμός Δικτύου	18
Σχεδιασμός Βάσης Δεδομένων	19
User Interface	22
Υλοποίηση	
Στάδια Υλοποίησης	24
Client εφαρμογή στο κινητό	24
Android	24
Εγκατάσταση Εργαλείων ανάπτυξης	26
Περιβάλλον ανάπτυξης εφαρμογής Android	28
AndroidManifest.xml	29
MainActivity.java	31
Υλοποίηση Βάσης Δεδομένων	33
Server εφαρμογή	
Διαχείριση	35
Login	36
Σελίδα Διαχείρισης - Admin.php	37
Προσθήκη Νέας Κατηγορίας Σταθμών	38

<b>Τροποποίηση Κατηγορίας Σταθμών</b>	<b>41</b>
<b>Προσθήκη Συσκευής - Σταθμών</b>	<b>43</b>
<b>Τροποποίηση Σταθμού</b>	<b>45</b>
<b>Εφαρμογή προβολής Σταθμού</b>	<b>45</b>
<b>Έλεγχος Συστήματος</b>	
<b>Εργαλεία ανάπτυξης λογισμικού</b>	<b>48</b>
<b>Φάσεις Υλοποίησης και ελέγχου</b>	<b>48</b>
<b>Βιβλιογραφία</b>	<b>50</b>

## Περίληψη

Οι εφαρμογές της Τηλεμετρίας είναι ένας συνδυασμός κλάδων της Πληροφορικής, όπως Προγραμματισμός Συστημάτων και Μικροεπεξεργαστών, Τηλεπικοινωνίες, Βάσεις Δεδομένων, Δίκτυα σε μια ολοκληρωμένη υλοποίηση που ξεκινάει από την απλή καταγραφή και ψηφιοποίηση μιας πληροφορίας και συνεχίζει στη μετάδοσή της σε απόσταση μέσω κάποιου πληροφοριακού συστήματος για την τελική αποθήκευση ή επεξεργασία της. Η εξέλιξη των Τηλεπικοινωνιών και Δικτύων έδωσε τη δυνατότητα αποστολής δεδομένων σε μεγάλες αποστάσεις με μεγάλη ταχύτητα. Με τη συγκεκριμένη εργασία προτείνεται μια υλοποίηση εφαρμογής τηλεμετρίας όπου ένα σύστημα καταγραφής μετρήσεων περιβάλλοντος αποστέλλει ανά τακτά χρονικά διαστήματα σε ένα κεντρικό σύστημα τις μετρήσεις για περαιτέρω επεξεργασία ή αποθήκευση. Η συσκευή η οποία θα συλλέγει τα δεδομένα στη συγκεκριμένη εργασία θα είναι ένα κινητό τηλέφωνο, που θα κάνει την καταγραφή και θα στέλνει τα δεδομένα μέσω ενός δικτύου ευρείας περιοχής σε ένα πληροφοριακό σύστημα. Τέλος το πληροφοριακό σύστημα θα έχει και τη δυνατότητα αποθήκευσης των δεδομένων.

Το καταγραφικό σύστημα αποτελείται από μια έξυπνη συσκευή κινητής τηλεφωνίας η οποία έχει εγκατεστημένο λειτουργικό Android. Η εφαρμογή η οποία εκτελείται για την καταγραφή και αποστολή δεδομένων είναι μια Android εφαρμογή. Τα δεδομένα αποστέλλονται σε μια web εφαρμογή η οποία εκτελείται σε κάποιον web server και τα δεδομένα αποστέλλονται μέσω Διαδικτύου. Όλο το πληροφοριακό σύστημα αποσκοπεί στο να είναι εύκολα επεκτάσιμο και παραμετροποιήσιμο, με μεγάλες δυνατότητες ανάπτυξης και άλλων λειτουργιών.

## **Εισαγωγή**

Η εξέλιξη πολλών παράλληλων επιστημών και κλάδων τους κατά τη διάρκεια των τελευταίων ετών έχουν συντελέσει στο να επιτευχθεί σήμερα η ολοκλήρωση και ο συνδυασμός διαφόρων τεχνολογικών εφαρμογών με σκοπό την καταγραφή και επεξεργασία πληροφοριών. Ο Προγραμματισμός Συστημάτων και Μικροεπεξεργαστών, οι Τηλεπικοινωνίες, οι Βάσεις Δεδομένων είναι κάποιοι ευρύτεροι κλάδοι της Επιστήμης της Πληροφορικής που βρίσκουν άμεση εφαρμογή στον σύγχρονο Ψηφιακό κόσμο και μπορούν να συνδυαστούν έτσι ώστε να αποδώσουν μια ολοκληρωμένη εφαρμογή Ψηφιακής Τεχνολογίας.

Οι εφαρμογές της Τηλεμετρίας θα μπορούσαμε να πούμε ότι είναι ένας συνδυασμός όλων αυτών των κλάδων της Πληροφορικής σε μια ολοκληρωμένη υλοποίηση που ξεκινάει από την απλή καταγραφή και ψηφιοποίηση μιας πληροφορίας και συνεχίζει στη μετάδοσή της σε απόσταση μέσω κάποιου πληροφοριακού συστήματος για την τελική αποθήκευση ή επεξεργασία της. Η εξέλιξη των Τηλεπικοινωνιών και Δικτύων έδωσε μια νέα μεγάλη ώθηση στον τομέα της Τηλεμετρίας. Έδωσε τη δυνατότητα αποστολής δεδομένων σε μεγάλες αποστάσεις με μεγάλη ταχύτητα αλλά κυρίως έκανε δυνατή τη μετάδοση τύπων δεδομένων όπως τα πολυμέσα (κείμενο, εικόνα, βίντεο, ήχος) που μέχρι πριν λίγα χρόνια ήταν απαγορευτικά για τον κόσμο της Τηλεπικοινωνίας.

## **Σκοπός της εργασίας**

Η συγκεκριμένη εργασία αποσκοπεί στη δημιουργία ενός συστήματος καταγραφής δεδομένων από το περιβάλλον που μπορεί να βρει εφαρμογή σε τεράστια κλίμακα στην επιστήμη αλλά και στην καθημερινή μας ζωή. Πρόκειται για ένα σύστημα καταγραφής μετρήσεων περιβάλλοντος ανά τακτά χρονικά διαστήματα και αποστολής του σε ένα κεντρικό σύστημα για περαιτέρω επεξεργασία ή αποθήκευση. Η συσκευή η οποία θα συλλέγει τα δεδομένα στη συγκεκριμένη εργασία θα είναι ένα κινητό τηλέφωνο, που θα κάνει την καταγραφή και θα στέλνει τα δεδομένα μέσω ενός δικτύου ευρείας περιοχής σε ένα πληροφοριακό σύστημα. Τέλος το πληροφοριακό σύστημα θα έχει και τη δυνατότητα αποθήκευσης των δεδομένων σε μια τέτοια μορφή η οποία μπορεί αργότερα να χρησιμοποιηθεί για επεξεργασία και εξαγωγή συμπερασμάτων μέσα από μια βάση δεδομένων (data mining).

Υπάρχουν αρκετά συστήματα τηλεμετρίας τα οποία στηρίζονται στην υλοποίηση ενός ιδιωτικού δικτύου μετάδοσης δεδομένων με τη χρήση δορυφορικών ή ραδιοκυματικών σημάτων. Η υλοποίησες όμως αυτές απαιτούν τεράστιο κόστος κατασκευής, εγκατάστασης και συντήρησης, κάτι το οποίο δεν είναι εφικτό από έναν μικρό οργανισμό ή μια επιχείρηση. Στη δική μας περίπτωση επιχειρούμε να χρησιμοποιήσουμε το ήδη υπάρχον δίκτυο κινητής τηλεφωνίας για τη δημιουργία ενός δικτύου ευρείας περιοχής και την αποστολή και λήψη δεδομένων.

Πέραν του κόστους, άλλο ένα πλεονέκτημα που έχει συγκεκριμένη υλοποίηση είναι η δυνατότητα επέκτασης του συστήματος (scalability) εφόσον εύκολα μπορεί να εγκατασταθεί η εφαρμογή αποστολής δεδομένων σε ένα νέο κινητό τηλέφωνο το οποίο θα αποτελέσει νέο κόμβο του δικτύου.

# 1. Τηλεμετρία

## 1.1 Ορισμός

Σύμφωνα με την εγκυκλοπαίδεια Parygus Larousse Britannica, η τηλεμετρία ορίζεται σαν μια αυτόματη εφαρμογή τηλεπικοινωνιών κατά την οποία οι μετρήσεις οι οποίες συλλέγονται σε ένα απομακρυσμένο ή δύσκολα προσβάσιμο μέρος μεταδίδονται προς ένα κέντρο για επεξεργασία. Όπως ήδη αναφέραμε σε μια τέτοια εφαρμογή αυτοματισμού οι κλάδοι Πληροφορικής που εμπλέκονται είναι ποικίλοι.

Τηλεμετρία είναι η τεχνολογία που επιτρέπει την εξ' αποστάσεως καταγραφή και δημοσιοποίηση μετρήσεων κάποιων μεγεθών που ενδιαφέρουν τον σχεδιαστή του συστήματος τηλεμετρίας. Πρόκειται για σύνθετη λέξη που προέρχεται από το τηλέ που σημαίνει απομακρυσμένα και το μέτρο.

Γενικά ο όρος τηλεμετρία αναφέρεται στη διαδικασία σύναξης και μεταφοράς της πληροφορίας από μια απομακρυσμένη τοποθεσία σε ένα κέντρο ελέγχου. Συνήθως όμως με την τηλεμετρία αναφερόμαστε σε αποστολή των δεδομένων των μετρήσεων σε ένα δέκτη χρησιμοποιώντας ασύρματα μέσα επικοινωνίας, αλλά μπορεί να περιγράψει και ένα σύστημα καταγραφής και μετάδοσης μετρήσεων μέσω τηλεφωνικού δικτύου ή δικτύου υπολογιστών.

## Υλικό και Λογισμικό του Συστήματος

### Απομακρυσμένες Συσκευές

Σε ένα σύστημα τηλεμετρίας υπάρχουν απομακρυσμένες συσκευές εγκατεστημένες στα σημεία παρακολούθησης του περιβάλλοντος οι οποίες μπορούν να συλλέγουν πληροφορίες και είτε να τις αποθηκεύουν προσωρινά πριν τις αποστείλουν ή να τις αποστέλλουν άμεσα σε ένα κεντρικό σύστημα για επεξεργασία και αποθήκευση. Συνεπώς οι συσκευές αυτές θα πρέπει να διαθέτουν κάποια επεξεργαστική ισχύ [Zheng 2006]. Τέτοιες μικροσυσκευές είναι τα smartphones και τα PDAs τα οποία διαθέτουν αρκετή επεξεργαστική ισχύ για να μπορέσουν να διεκπεραιώσουν τις ακόλουθες εργασίες:

- Να συλλέξουν πληροφορίες με τη χρήση κάποιου αισθητήρα συνδεδεμένου με τη συσκευή.
- Να αποθηκεύουν προσωρινά τις πληροφορίες.
- Να συνδεθούν με ένα κεντρικό πληροφοριακό σύστημα για την αποστολή των πληροφοριών.

Η διαδικασία ξεκινάει με τη φάση της Λήψης Δεδομένων από το περιβάλλον. Στη φάση αυτή κάποιος αισθητήρας συλλέγει μετρήσεις από το περιβάλλον σε αναλογική μορφή. Οι μετρήσεις αυτές συνήθως μετατρέπονται απευθείας σε ψηφιακή μορφή από κάποιο ηλεκτρονικό κύκλωμα ενσωματωμένο στον αισθητήρα πριν αποσταλούν στη μικροσυσκευή. Στη συνέχεια μέσω ενός καναλιού επικοινωνίας με τη μικροσυσκευή, η μέτρηση στέλνεται στη μικροσυσκευή. Στα κινητά τηλέφωνα υπάρχουν ενσύρματα και ασύρματα πρωτόκολλα επικοινωνίας για τη σύνδεση της τηλεφωνικής συσκευής με τον αισθητήρα. Ασύρματα πρωτόκολλα είναι οι συνδέσεις Bluetooth και InfraRed ενώ ενσύρματα είναι τα RS-232C και RS 485 [Mohitpour 2003].

Σε κάποιες περιπτώσεις τα δεδομένα που αποθηκεύονται στη συσκευή μπορούν να υποβληθούν σε κάποια επεξεργασία πριν αποσταλούν σε κάποιο κεντρικό πληροφοριακό σύστημα. Τα Smartphones και τα PDAs μπορούν να εκτελέσουν πολύπλοκους υπολογισμούς και να επεξεργαστούν τις μετρήσεις πριν αυτές αποσταλούν για επεξεργασία και αποθήκευση στο κεντρικό σύστημα.



Στο τελευταίο στάδιο, η απομακρυσμένη συσκευή συνδέεται με το κεντρικό πληροφοριακό σύστημα μέσω ενός καναλιού ασύρματης επικοινωνίας. Τα δεδομένα μπορούν να αποσταλούν είτε ανεπεξέργαστα ή αφού υποβληθούν σε κάποια στοιχειώδη επεξεργασία στη μικροσυσκευή. Τέτοια επεξεργασία μπορεί να είναι η διαμόρφωσή τους σε κάποιο συγκεκριμένο format (XML, Fixed Length Record) ή συμπίεση των δεδομένων για ταχύτερη μετάδοση. Η απομακρυσμένη συσκευή θα πρέπει να χρησιμοποιήσει ένα καθορισμένο πρωτόκολλο επικοινωνίας με το κεντρικό πληροφοριακό σύστημα για την εγκατάσταση της επικοινωνίας και την αποστολή των δεδομένων.

### **Υλικό της Συσκευής**

Οι απομακρυσμένες συσκευές οι οποίες θα συλλέγουν τις πληροφορίες και θα τις αποστέλλουν στο κεντρικό σύστημα θα έχουν όπως ήδη προαναφέραμε μια στοιχειώδη επεξεργαστική ισχύ έτσι ώστε να μπορούν να εκτελούν βασικές λειτουργίες όπως ακριβώς ισχύει και με τα τερματικά τα οποία είναι συνδεδεμένα με έναν κεντρικό υπολογιστή στην περίπτωση ενός δικτύου υπολογιστών.

Για να μπορούν οι συσκευές αυτές να εκτελούν τις βασικές λειτουργίες που έχουμε αναφέρει θα πρέπει να έχουν κάποια βασικά χαρακτηριστικά:

- Έναν μικροεπεξεργαστή ο οποίος θα είναι σε θέση να εκτελέσει τον κώδικα ενός προγράμματος.
- Κάποια μορφή Κυρίας Μνήμης όπως ROM (Read Only Memory) ή RAM (Random Access Memory) ή όπως πολύ συχνά χρησιμοποιούνται μια μορφή μνήμης flash. Οι μνήμες ROM συνήθως περιέχουν ρυθμίσεις του λειτουργικού συστήματος και εντολές ενσωματωμένες από τον κατασκευαστή της συσκευής ενώ οι μνήμη RAM μπορεί να χρησιμοποιηθεί κυρίως για την εκτέλεση ενός προγράμματος και την προσωρινή αποθήκευση δεδομένων.
- Ένα σύστημα χρονισμού το οποίο μπορεί να παρέχει πληροφορίες για την ημερομηνία και ώρα που τα δεδομένων συλλέχθηκαν.
- Ένα σύστημα Εισόδου / Εξόδου (I/O) για τη μεταφορά των δεδομένων από και προς τη συσκευή.
- Ένα σύστημα επικοινωνίας με κάποιο δίκτυο έτσι ώστε τα δεδομένα να μπορούν να αποσταλούν κάπου απομακρυσμένα. Στην περίπτωση των κινητών τηλεφώνων το σύστημα χρησιμοποιεί το ασύρματο δίκτυο κινητής τηλεφωνίας για την αποστολή των δεδομένων σε ένα απομακρυσμένο πληροφοριακό κέντρο [Mohitpour 2003].

Τέτοιες συσκευές στο παρελθόν ήταν τα Pocket PC ή τα PDA (Personal Digital Assistants PDA). Σήμερα οι συσκευές οι οποίες κυριαρχούν στο χώρο της κινητής τηλεφωνίας είναι τα smartphones ή αλλιώς «έξυπνα κινητά». Επιπλέον οι δυνατότητες που παρέχουν σήμερα οι υπηρεσίες κινητής τηλεφωνίας, επιτρέπουν σε μια τέτοια συσκευή να αποστέλλει και να λαμβάνει δεδομένα μέσω του Internet. Ουσιαστικά είναι συσκευές που συνδέονται ασύρματα με άλλους υπολογιστές μέσω των διαδεδομένων πρωτοκόλλων επικοινωνίας του Internet και ως εκ τούτου μπορούν να αποστέλλουν πληροφορίες σε ένα απομακρυσμένο κεντρικό υπολογιστή ο οποίος είναι επίσης συνδεδεμένος με το Internet. Στην πραγματικότητα η επικοινωνία αυτή εξυπηρετείται από την ύπαρξη κεντρικών διακομιστών των εταιριών κινητής τηλεφωνίας οι οποίοι λειτουργούν ως Πύλες (Gateway) ανάμεσα στο δίκτυο της κινητής τηλεφωνίας και του Internet. Τα δεδομένα από μια συσκευή αποστέλλονται ως πακέτα μέσω του δικτύου της κινητής τηλεφωνίας και κατευθύνονται σε ένα κέντρο το οποίο αναλαμβάνει να τα μετατρέψει σε TCP/IP πακέτα και να τα κατευθύνει στο Διαδίκτυο.

Οι υπηρεσίες του Internet έγιναν διαθέσιμες στις συσκευές και τα δίκτυα κινητής τηλεφωνία κατά τη 2<sup>η</sup> Γενιά κινητών τηλεφώνων (2G). Οι συσκευές όμως στα πρώτα στάδια είχαν πολύ περιορισμένες δυνατότητες με αποτέλεσμα πολλές εφαρμογές Διαδικτύου να υλοποιούνται σε πρωτόκολλα τα οποία δημιουργήθηκαν για να εξυπηρετήσουν την κινητή τηλεφωνία όπως ήταν το WAP και το iMode. Σήμερα τα κινητά τηλέφωνα έχουν δυνατότητες περίπου όσες και ένα απλό PC. Υπάρχουν κινητά τηλέφωνα των οποίων οι επεξεργαστές τους τρέχουν στα 500 MHz έως και 2 GHz.

Τα σημερινά κινητά τηλέφωνα είναι εφοδιασμένα με εσωτερική ή εξωτερική μνήμη πολλών megabyte. Επίσης διαθέτουν έγχρωμες οθόνες υψηλής ανάλυσης με γραφικά περιβάλλοντα διεπαφής με το χρήστη κινητής τηλεφωνίας. Τα περισσότερα πλέον κινητά διαθέτουν κάμερες CCD υψηλής ανάλυσης για τη λήψη φωτογραφιών και video. Σε κάποιες περιπτώσεις τα κινητά έχουν ενσωματωμένους δέκτες GPS ή μπορούν να συνδεθούν εξωτερικά με κάποιο δέκτη μέσω του πρωτοκόλλου σύνδεσης Bluetooth. Τέλος έχουν πρόσβαση στο Internet ή σε intranets και extranets με διάφορους τρόπους όπως μέσω τοπικών δικτύων Wi-Fi, ή Ασύρματων Δικτύων Ευρείας Περιοχής (Wireless Wide-Area Networks - WWANs) ή μέσω του δικτύου κινητής τηλεφωνίας 3<sup>ης</sup> Γενιάς (3G) και 4<sup>ης</sup> Γενιάς (4G).

## Αρχιτεκτονική Λογισμικού

Η δημιουργία ενός συστήματος τηλεμετρίας όπως το περιγράψαμε προϋποθέτει την ανάπτυξη μιας κατανεμημένης εφαρμογής λογισμικού για τη μετάδοση δεδομένων από τις απομακρυσμένες συσκευές προς το κεντρικό υπολογιστικό σύστημα. Η κυρίαρχη τεχνολογία στις εφαρμογές λογισμικού σε δίκτυα υπολογιστών είναι η αρχιτεκτονική Client-Server (Πελάτη-Εξυπηρετητή).

“Η αρχιτεκτονική Client-server είναι μια αρχιτεκτονική δικτύων που διαχωρίζει την εφαρμογή-πελάτη (συνήθως ένα γραφικό περιβάλλον διαχείρισης της εφαρμογής από τον εξυπηρετητή). Κάθε στιγμιότυπο της εφαρμογής – πελάτη μπορεί να στέλνει κάποια μορφή αίτησης προς μια εφαρμογή-εξυπηρετητή” [Warnock 1988]. Μια άλλη διάκριση μεταξύ των δύο μερών μιας client server εφαρμογής είναι ότι καθώς τα δεδομένα ανταλλάσσονται μεταξύ δύο υπολογιστικών συστημάτων, στην περίπτωση του εξυπηρετητή τα δεδομένα φυλάσσονται και επεξεργάζονται σε ένα σύστημα με μεγάλη επεξεργαστική ισχύ ενώ στην περίπτωση της εφαρμογής πελάτη μπορεί να χρησιμοποιηθεί ένα απλό τερματικό όπως ένα απλό PC. Συνήθως το σύστημα που αποτελεί τον server αναμένει από τον client κάποιο αίτημα [Harold 2004].

Η client – server αρχιτεκτονική έγινε πολύ δημοφιλής στα τέλη της δεκαετίας του 80. Σύμφωνα με τον Warnock “η client - server αρχιτεκτονική είναι ευέλικτη, βασισμένη σε μηνύματα, και χρησιμοποιεί τμηματικό προγραμματισμό με σκοπό τη βελτίωση της χρηστικότητας, ευελιξίας, συμβατότητας και επεκτασιμότητας σε σύγκριση με τα συστήματα mainframe” [Warnock 1988]. Ένα παράδειγμα εφαρμογής του Client Server είναι το Internet και οι υπηρεσίες του. Ο Παγκόσμιος Ιστός και οι ιστοσελίδες είναι ένα χαρακτηριστικό παράδειγμα υλοποίησης αρχιτεκτονικής client-server. Το μεγαλύτερο πλεονέκτημα της τεχνολογίας του Internet είναι ότι μπορεί να υποστηρίξει απεριόριστους clients και σημεία παραλαβής δεδομένων. Στην περίπτωση του συστήματος παρακολούθησης, μπορεί να γίνει η παρακολούθηση και ο έλεγχος οποιουδήποτε σημείου που έχει πρόσβαση στο Internet εύκολα και αποδοτικά. Τα δεδομένα μπορούν να μεταδοθούν και να παρουσιαστούν οπουδήποτε μειώνοντας το κόστος ανάπτυξης ειδικού δικτύου και την πολυπλοκότητα μιας απομακρυσμένης τηλεπικοινωνιακής σύνδεσης.

Στην περίπτωση μιας client εφαρμογής μέσω του Internet δεν απαιτείται η εγκατάσταση κανενός ειδικού λογισμικού παρά μόνο ενός φυλλομετρητή ο οποίος θα παρέχει το γραφικό περιβάλλον διαχείρισης στον χειριστή της εφαρμογής. Αξίζει να σημειωθεί ότι στις περιπτώσεις όπου η

εφαρμογή είναι διαδικτυακή, μπορεί να λειτουργήσει σε οποιαδήποτε πλατφόρμα και λειτουργικό σύστημα. Τέτοιες εφαρμογές υλοποιούνται συνήθως με τη χρήση των τεχνολογιών Java ή .Net που είναι προσανατολισμένες στην ανάπτυξη τέτοιων συστημάτων [Business Wire, 2000].

### **Λογισμικό στον Server**

Γενικά στην αρχιτεκτονική client-server κάθε υπολογιστής που συνδέεται στο δίκτυο αναλαμβάνει το ρόλο είτε του πελάτη ή του εξυπηρετητή χωρίς να είναι απαραίτητο να παραμείνει στο ρόλο αυτό για πάντα. Μπορεί ένας υπολογιστής οποίος λειτουργεί ως server σε μια δεδομένη στιγμή, να λειτουργεί ως client σε κάποια άλλη χρονική στιγμή. Το λογισμικό που είναι εγκατεστημένο σε ένα υπολογιστή ο οποίος λειτουργεί ως server αναλαμβάνει να εξυπηρετεί πολλούς υπολογιστές ανταλλάσσοντας δεδομένα με αυτούς. Συνήθως ο server είναι ένας υπολογιστής με μεγάλη επεξεργαστική ισχύ έτσι ώστε να μπορούν να εξυπηρετήσουν πολλούς υπολογιστές πελάτες ταυτόχρονα. Υπάρχει συνήθως ένας περιορισμός για το πλήθος των υπολογιστών που μπορούν να συνδεθούν ταυτόχρονα με κάποιον server για την αποστολή και λήψη δεδομένων και εξαρτάται από τις προδιαγραφές της εφαρμογής που είναι εγκατεστημένη για την εξυπηρέτηση των υπολογιστών πελατών.

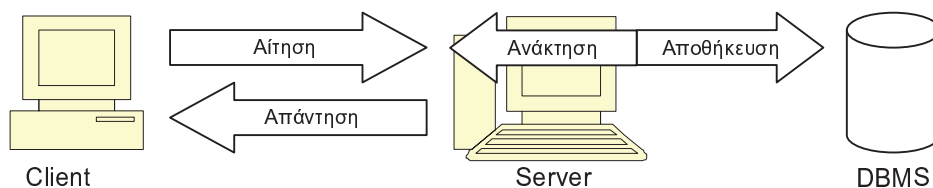
Γενικά η εφαρμογή που εκτελείται στον server είναι παθητική με την έννοια ότι αναμένει αιτήσεις από τους clients για την αποστολή ή λήψη δεδομένων. Σε άλλες περιπτώσεις όπως στην περίπτωση των web services, ο server είναι αυτός που αποστέλλει την αίτηση στον client. Η εφαρμογή server πέραν της λειτουργίας της παραλαβής αιτήσεων και αποστολής δεδομένων στους clients, μπορεί να επιτελεί και άλλες λειτουργίες όπως την αποθήκευση δεδομένων σε Βάσεις Δεδομένων, την επεξεργασία τους και την εξαγωγή συμπερασμάτων ή την διαχείριση συμβάντων όπως σημάτων κινδύνου [Harold 2004].

## Λογισμικό στον Client

Στις προηγούμενες παραγράφους παρουσιάστηκαν οι διάφορες συσκευές που μπορούν να αποτελέσουν τις client συσκευές και τα βασικά χαρακτηριστικά τους. Όπως περιγράψαμε οι συσκευές με ρόλο client στην client-server αρχιτεκτονική αποστέλλουν αιτήσεις σε κάποια εφαρμογή server και περιμένουν για κάποια απάντηση από τον server. Οι εφαρμογές Client μπορούν να εκτελούνται σε τερματικά συνδεδεμένα με το δίκτυο όπως είναι τα απλά PC. Παρόλα αυτά, οι σύγχρονες πλατφόρμες ανάπτυξης λογισμικού υποστηρίζουν τη δημιουργία εφαρμογών οι οποίες μπορούν να εκτελούνται σε άλλες συσκευές όπως είναι οι μικροσυσκευές, τα έξυπνα τηλέφωνα, τα PDA και τα οποία αναφέρονται με τον όρο Ελαφριά (thin) client. Ένα thin client μπορεί να είναι είτε ένας υπολογιστής δικτύου χωρίς σκληρό δίσκο σχεδιασμένος έτσι ώστε να είναι αρκετά μικρός ή ένα κινητό τηλέφωνο. Αυτές οι συσκευές χρησιμοποιούνται μόνο για να στέλνουν δεδομένα σε κάποιον server για την περεταίρω επεξεργασία. Οι κυριότερες πλατφόρμες ανάπτυξης λογισμικού για μικρές συσκευές και κινητά τηλέφωνα είναι η J2ME για απλά Java-enabled κινητά και η ανάπτυξη λογισμικού για Android ή iOS για smartphones.

## Αρχιτεκτονική 3-Tier

Το μοντέλο Client/Server αναφέρεται και ως αρχιτεκτονική 2-tier διότι αποτελείται από δύο μέρη τον Client και τον Server. Στις περιπτώσεις που τα δεδομένα αποθηκεύονται σε κάποια βάση δεδομένων σε ένα ξεχωριστό server, το μοντέλο αποτελείται από τρία διαφορετικά συστατικά. Οι εφαρμογές Client στέλνουν δεδομένα στην εφαρμογή server η οποία αποθηκεύει τα δεδομένα σε ένα ξεχωριστό σύστημα Διαχείριση Βάσεων Δεδομένων (Database Management System - DBMS). Το ξεχωριστό αυτό σύστημα είναι υπεύθυνο για τη διαχείριση και τη διανομή των δεδομένων. Το συγκεκριμένο μοντέλο αποτελείται από τρία συστατικά και ονομάζεται αρχιτεκτονική 3-tier. Η προσέγγιση αυτή είναι καλύτερη από το απλό client/server model επειδή απομονώνει τη λειτουργία της αποθήκευσης και διαχείρισης των δεδομένων από τη λειτουργία της επεξεργασίας τους με αποτέλεσμα να απελευθερώνεται σημαντικός χρόνος και πόροι όσον αφορά τον server παραλαβής και επεξεργασίας των δεδομένων. Επιπλέον το μοντέλο 3-tier αυξάνει την συμβατότητα και την επεκτασιμότητα όλου του πληροφοριακού συστήματος αφού η διαχείριση των δεδομένων αποδεδεσμεύεται από τη συγκεκριμένη εφαρμογή και εκτελείται από ένα σύστημα που μπορεί να είναι συμβατό και διασυνδεδεμένο και με άλλες εφαρμογές [Edelstein 1994].



**Σχήμα 1. 3 – Tier Μοντέλο**

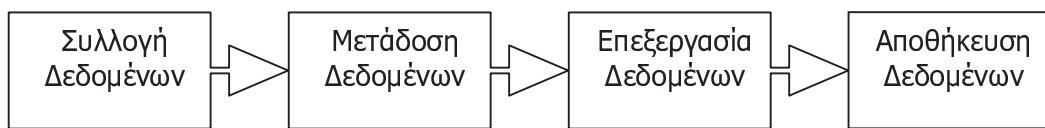
Η σύνδεση και ανταλλαγή δεδομένων με το Σύστημα Διαχείρισης της ΒΔ μπορεί να γίνει με τη χρήση ειδικών οδηγών για κάθε μια βάση δεδομένων. Οι οδηγοί αυτοί είναι ειδικές εφαρμογές λογισμικού οι οποίες έχουν δημιουργηθεί για κάθε ένα σύστημα ΒΔ και λειτουργούν σαν μεσάζον λογισμικό (middleware) μεταξύ της εφαρμογής και της ΒΔ. Στην περίπτωση όμως που για κάποιο λόγο γίνει μεταφορά των δεδομένων σε κάποιο άλλο σύστημα ΒΔ τότε θα πρέπει να αλλάξει και ο πηγαίος κώδικας της εφαρμογής server που στέλνει τα δεδομένα προς αποθήκευση στη ΒΔ. Αυτό όμως δεν συμβαδίζει με τη λογική της αρχιτεκτονικής 3-tier στην οποία λειτουργία της επεξεργασίας των δεδομένων θα πρέπει να είναι ανεξάρτητη από τη λειτουργία της αποθήκευσης και διαχείρισης της ΒΔ. Στην περίπτωση αυτή αντί των συγκεκριμένων οδηγών μπορεί να χρησιμοποιηθεί κάποιο πρωτόκολλο ανοικτής διασύνδεσης με Βάσεις Δεδομένων όπως είναι το πρότυπο Open Database Connectivity (ODBC). Έτσι η ίδια εφαρμογή μπορεί να συνδεθεί χωρίς να γίνει καμία αλλαγή με μια βάση δεδομένων που είναι υλοποιημένη σε διάφορες πλατφόρμες όπως την MySQL ή τον MS SQL Server.

### **Επεξεργασία Δεδομένων**

Στην περίπτωση των συστημάτων τηλεμετρίας υπάρχουν τρεις βασικές κατηγορίες δεδομένων που θα πρέπει να διατηρούνται ανάλογα με τη λειτουργία του συστήματος. Μια από αυτές τις κατηγορίες είναι τα δεδομένα που λαμβάνονται σε πραγματικό χρόνο και συλλέγονται σε ανεπεξέργαστη μορφή από τις απομακρυσμένες συσκευές πριν αποσταλούν στην εφαρμογή server για επεξεργασία. Ο δεύτερος τύπος πληροφοριών είναι τα ιστορικά δεδομένα τα οποία αποθηκεύονται σε μια βάση σε χρονολογική σειρά συνοδευόμενα και από την ημερομηνία και ώρα που συλλέχθηκαν. Τα δεδομένα αυτά μπορούν είτε να αποθηκευτούν μετά από κάποια επεξεργασία η να αποθηκευτούν σε ανεπεξέργαστη μορφή και να χρησιμοποιηθούν για την εξαγωγή συμπερασμάτων στο μέλλον. Τέλος στη βάση δεδομένων μπορεί να αποθηκευτούν και

δεδομένα τα οποία έχουν προκύψει από στατιστική ανάλυση και επεξεργασία των ιστορικών δεδομένων[Dai 2004].

Η συνολική διαδικασία της συλλογής και διαχείρισης δεδομένων ξεκινά με την σύλληψη των μετρήσεων από τις απομακρυσμένες συσκευές και τη μετάδοσή τους σε έναν server για επεξεργασία. Οι βασικές φάσεις της διαδικασίας περιγράφονται στο παρακάτω σχήμα:



**Σχήμα 2.** Στάδια Διαχείρισης Δεδομένων

Στο πρώτο στάδιο τα δεδομένα περισυλλέγονται από διάφορες πηγές που είναι συνδεδεμένες σε ένα δίκτυο μετάδοσης πληροφοριών έτσι ώστε να υπάρχει η δυνατότητα αποστολής των δεδομένων σε ένα κεντρικό σύστημα για περαιτέρω επεξεργασία. Συνήθως οι απομακρυσμένες μικρές συσκευές χρησιμοποιούν κάποιον αισθητήρα προσαρτημένο πάνω τους για τη συλλογή δεδομένων όπως για παράδειγμα έναν αισθητήρα θερμοκρασίας [Parkinson 1996]. Τα δεδομένα μπορούν να αποθηκευτούν προσωρινά στη μνήμη της συσκευής και στη συνέχεια να αποσταλούν στον κεντρικό σταθμό. Η συχνότητα με την οποία γίνεται η δειγματοληψία επηρεάζει τον όγκο της πληροφορίας που θα στέλνεται στο σταθμό. Θα πρέπει να ληφθεί υπόψη ότι πολλά δεδομένα θα στέλνονται ταυτόχρονα από πολλές συσκευές και ο server θα πρέπει να είναι σε θέση να διαχειριστεί όλα τα εισερχόμενα δεδομένα.

Τα δεδομένα αποστέλλονται σε ένα κεντρικό σύστημα και παρέχουν πληροφορίες για την παρακολούθηση του συστήματος σε πραγματικό χρόνο. Παρόλα αυτά τα δεδομένα αποθηκεύονται παράλληλα σε ένα σύστημα διαχείρισης δεδομένων για μελλοντική χρήση. Η στατιστική επεξεργασία των αποθηκευμένων δεδομένων μπορεί να οδηγήσει σε πολύ σημαντικές πληροφορίες. Τα ιστορικά δεδομένα αν και πολλά σε ποσότητα παρέχουν ελάχιστη πληροφορία στη μορφή που αποθηκεύονται. Η διατήρηση μεγάλων βάσεων δεδομένων όμως δίνει τη δυνατότητα για ανάλυση των δεδομένων αυτών με σύγχρονες τεχνικές επεξεργασίας όπως είναι οι τεχνικές data mining. Χρησιμοποιώντας τέτοιες τεχνικές μπορούν να αποκαλυφθούν πληροφορίες και συμπεράσματα που ήταν κρυμμένα χωρίς την κατάλληλη επεξεργασία. Τέτοιες

τεχνικές όπως sequential patterns, association rules και classifications μπορούν να αποκαλύψουν μοντέλα πρόβλεψης γεγονότων και συμπεριφορών [Fayyad 1995].

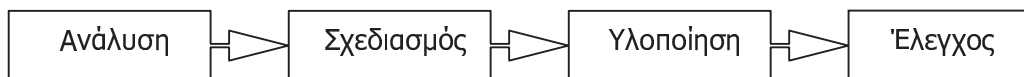
## Ανάλυση και Σχεδιασμός

### Μεθοδολογία

Το σύστημα μας αναπτύχθηκε σε πέντε στάδια:

1. Ανάλυση Απαιτήσεων
2. Ανάλυση Συστήματος
3. Σχεδιασμός
4. Υλοποίηση
5. Έλεγχος και Αξιολόγηση

Ξεκινώντας από τη φάση της ανάλυσης των απαιτήσεων, οι στόχοι λειτουργίας του συστήματος προσδιορίστηκαν αρχικά όσον αφορά τη λειτουργικότητά του. Τα αποτελέσματα από αυτή την ανάλυση χρησιμοποιήθηκαν για τον καθορισμό των προδιαγραφών και των χαρακτηριστικών του συστήματος που θα ικανοποιούν τις παραπάνω απαιτήσεις. Μετά την ανάλυση ακολουθεί η φάση του σχεδιασμού του συστήματος όπου εκεί εξετάστηκαν οι διάφορες εναλλακτικές προτάσεις από τη σκοπιά των τεχνολογιών που θα πρέπει να χρησιμοποιηθούν για την ικανοποίηση των απαιτήσεων και την υλοποίηση του συστήματος. Κατά τη διάρκεια της υλοποίησης, οι τεχνολογίες που προτάθηκαν στη φάση του σχεδιασμού χρησιμοποιούνται για την κατασκευή του συστήματος και παράγονται οι εφαρμογές λογισμικού που θα εξυπηρετήσουν το σύστημα. Τέλος κατά τον έλεγχο και την αξιολόγηση του συστήματος, εξετάζεται η λειτουργικότητα του συστήματος και ο βαθμός στον οποίο ικανοποιούνται οι αρχικές απαιτήσεις.



Σχήμα 3. Μεθοδολογία



## Ανάλυση Απαιτήσεων

Στο πρώτο στάδιο ανάπτυξης του συστήματος γίνεται καταγραφή των απαιτήσεων του συστήματος από τρεις διαφορετικές σκοπιές:

- Λειτουργικές Απαιτήσεις
- Απαιτήσεις Δεδομένων
- Απαιτήσεις Περιβάλλοντος Διεπαφής (User interface)

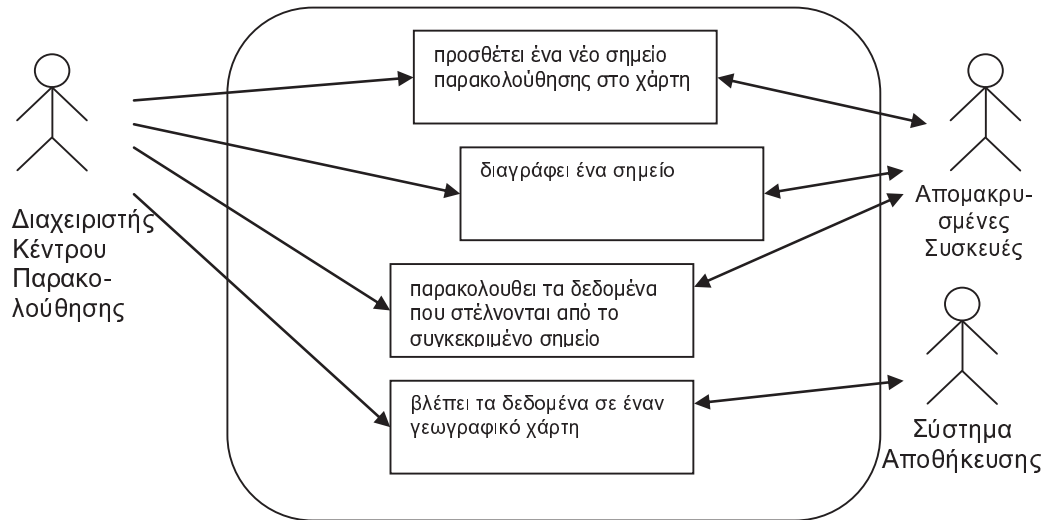
Ο βασικός στόχος του συστήματος είναι η παρακολούθηση σε πραγματικό χρόνο κάποιων περιβαλλοντικών δεδομένων όπως η θερμοκρασία από ένα δίκτυο απομακρυσμένων σταθμών και η αποστολή των δεδομένων αυτών σε έναν κεντρικό εξυπηρετητή όπου θα μπορούν τα εισερχόμενα δεδομένα να εμφανίζονται σε μια οθόνη. Στη συγκεκριμένη υλοποίηση τα δεδομένα συλλέγονται από έξυπνα κινητά τηλέφωνα και αποστέλλονται μέσω ενός δικτύου κινητής τηλεφωνίας σε έναν εξυπηρετητή μέσω του Διαδικτύου. Τα δεδομένα που αποστέλλονται στον υπολογιστή σε τακτά χρονικά διαστήματα ανάλογα με τη συχνότητα δειγματοληψίας αποθηκεύονται σε μια βάση δεδομένων. Οι απαιτήσεις όσον αφορά τη λειτουργία της εφαρμογής εξαρτώνται από τις απαιτήσεις του χρήστη του συστήματος. Οι απαιτήσεις αυτές μπορούν να ποικίλουν αλλά το ζητούμενο είναι δημιουργηθεί μια υποδομή η οποία θα μπορεί να εξυπηρετήσει καταρχήν τις ελάχιστες απαιτήσεις αλλά θα μπορεί επίσης να υποστηρίξει μελλοντικά και ότι άλλες απαιτήσεις προκύψουν. Στη συγκεκριμένη περίπτωση οι ελάχιστες απαιτήσεις στη λειτουργία τους συστήματος είναι:

- Online μετάδοση των δεδομένων από το κινητό τηλέφωνο προς τον εξυπηρετητή μέσω mobile Internet.
- Παρακολούθηση των δεδομένων μέσα από μια web εφαρμογή και διάθεση των δεδομένων σε χρήστες του Παγκόσμιου Ιστού σε πραγματικό χρόνο.
- Αποθήκευση των εισερχόμενων δεδομένων κατά χρονολογική σειρά σε μια κεντρική βάση δεδομένων η οποία μπορεί να χρησιμοποιηθεί αργότερα για ανάκτηση των δεδομένων και μελλοντική επεξεργασία.

Όσον αφορά τις λειτουργίες του διαχειριστή της web εφαρμογής, θα πρέπει να είναι σε θέση να ολοκληρώσει τις ακόλουθες λειτουργίες:

- Να προσθέσει ένα νέο σημείο παρακολούθησης στο χάρτη.
- Να μπορεί να διαγράψει ένα σημείο.
- Να μπορεί να δει τα δεδομένα σε έναν γεωγραφικό χάρτη.

Οι παραπάνω λειτουργίες περιγράφονται από το ακόλουθο Διάγραμμα Use Case.



Σχήμα 4 Use Case Διάγραμμα

## Ανάλυση Συστήματος

Σύμφωνα με τις απαιτήσεις που περιγράφηκαν παραπάνω, για την εξυπηρέτησή τους απαιτείται η εγκατάσταση ενός δικτύου ευρείας περιοχής (Wide Area Network – WAN) το οποίο θα αποτελείται από πολλές απομακρυσμένες συσκευές κινητής τηλεφωνίας και ένα κεντρικό υπολογιστή – server για τη λήψη των δεδομένων από τις συσκευές. Οι συσκευές θα συλλέγουν τα δεδομένα από το περιβάλλον και θα τα στέλνουν μέσω του δικτύου στον κεντρικό υπολογιστή. Το συγκεκριμένο δίκτυο WAN θα πρέπει να έχει δύο βασικά χαρακτηριστικά. Πρώτον θα πρέπει να εξυπηρετεί διαρκώς και αδιάλειπτα τη μετάδοση δεδομένων και κατά δεύτερον θα πρέπει να εξυπηρετεί όλες τις γεωγραφικές περιοχές, ακόμα και αυτές στις οποίες δεν υπάρχει εύκολη πρόσβαση. Σε μια τέτοια περίπτωση η μετάδοση δεδομένων μπορεί να γίνει μόνο με εναέρια και ασύρματα μέσα αφού η πρόσβαση σε ενσύρματη υποδομή είναι πολύ δύσκολη και οικονομικά ασύμφορη λόγω του κόστους κατασκευής. Παλαιότερα αντίστοιχες υλοποιήσεις απαιτούσαν τη δημιουργία σταθμών αποστολής και λήψης ραδιοσημάτων ή τη χρήση δορυφορικής σύνδεσης. Η μεγάλη εξέλιξη στο χώρο της κινητής τηλεφωνίας έδωσε τη δυνατότητα μετάδοσης δεδομένων μέσω των δικτύων κινητής τηλεφωνίας και σήμερα η γεωγραφική κάλυψη των δικτύων αυτών φτάνει σχεδόν το 100% της επικράτειας μιας χώρας.

Στα σύγχρονα δίκτυα κινητής τηλεφωνίας τα πακέτα δεδομένων από μια συσκευή κινητής τηλεφωνίας αποστέλλονται από τη συσκευή προς την κεραία κινητής τηλεφωνίας που εξυπηρετεί τη συγκεκριμένη περιοχή και από εκεί το κέντρο μεταγωγής των πακέτων. Τα πακέτα που αφορούν το Διαδίκτυο στέλνονται σε ένα σύστημα διακομιστών το οποίο λειτουργεί ως Πύλη (Internet gateway) μεταξύ του δικτύου κινητής τηλεφωνίας και του Διαδικτύου. Έτσι τα δεδομένα ανάλογα με τη διεύθυνση IP φεύγουν από την Πύλη προς τον προορισμό τους και καταλήγουν στη διεύθυνση του υπολογιστή παραλήπτη.

### **Ανάλυση Δεδομένων**

Σε ένα δίκτυο συσκευών που θα αποστέλλουν δεδομένα θερμοκρασιών, θα πρέπει να προσδιορίζεται η θέση του κάθε σταθμού μέτρησης. Έτσι για κάθε συσκευή η οποία είναι τοποθετημένη σε ένα γεωγραφικό σημείο πρέπει να προσδιορίζεται το χαρακτηριστικό όνομα της συσκευής καθώς και η γεωγραφική της θέση με βάση το γεωγραφικό της μήκος και πλάτος. Οι συσκευές αυτές θα πρέπει να έχουν και ένα όνομα το οποίο θα τις χαρακτηρίζει μοναδικά και επιπλέον θα συνοδεύει τα δεδομένα που αποστέλλονται έτσι ώστε η εφαρμογή που τρέχει στο κεντρικό υπολογιστή και λαμβάνει τα δεδομένα να μπορεί να γνωρίζει από ποιο σταθμό μέτρησης προήλθαν.

Τα δεδομένα που αποστέλλονται από τις συσκευές στο συγκεκριμένο παράδειγμα είναι μετρήσεις όσον αφορά την θερμοκρασία. Συνεπώς η θερμοκρασία θα πρέπει να αποστέλλεται στον εξυπηρετητή συνοδευμένη από το όνομα της συσκευής-σταθμού. Στην απλή περίπτωση που τα δεδομένα απλώς εμφανίζονται σε μια κεντρική οθόνη και παρακολουθούνται από έναν χειριστή, οι πληροφορίες που περιγράφηκαν είναι αρκετές. Στην περίπτωση όμως που τα δεδομένα προορίζονται για μελλοντική χρήση τότε πρέπει να αποθηκεύονται σε μια βάση δεδομένων κατά χρονολογική σειρά άφιξης. Συνεπώς εκτός από τις μετρήσεις της θερμοκρασίας καθώς και του κωδικού της συσκευής, απαιτείται και ημερομηνία και ώρα παραλαβής των δεδομένων από τον κεντρικό υπολογιστή. Η πληροφορία αυτή είναι απαραίτητη στην περίπτωση που ο χειριστής του συστήματος θέλει να ανακτήσει μελλοντικά τις μετρήσεις αυτές για κάποια συγκεκριμένη περίοδο και να δει την εξέλιξη των δεδομένων αυτών με το πέρασμα του χρόνου.

Συμπερασματικά λοιπόν θα λέγαμε ότι οι απαραίτητες πληροφορίες που θα πρέπει να είναι αποθηκευμένες στη βάση δεδομένων είναι οι εξής:

- Οι γεωγραφικές συντεταγμένες της απομακρυσμένης συσκευής κινητής τηλεφωνίας.
- Το όνομα της συσκευής.
- Η θερμοκρασία.
- Η ημερομηνία και ώρα λήψης των δεδομένων.

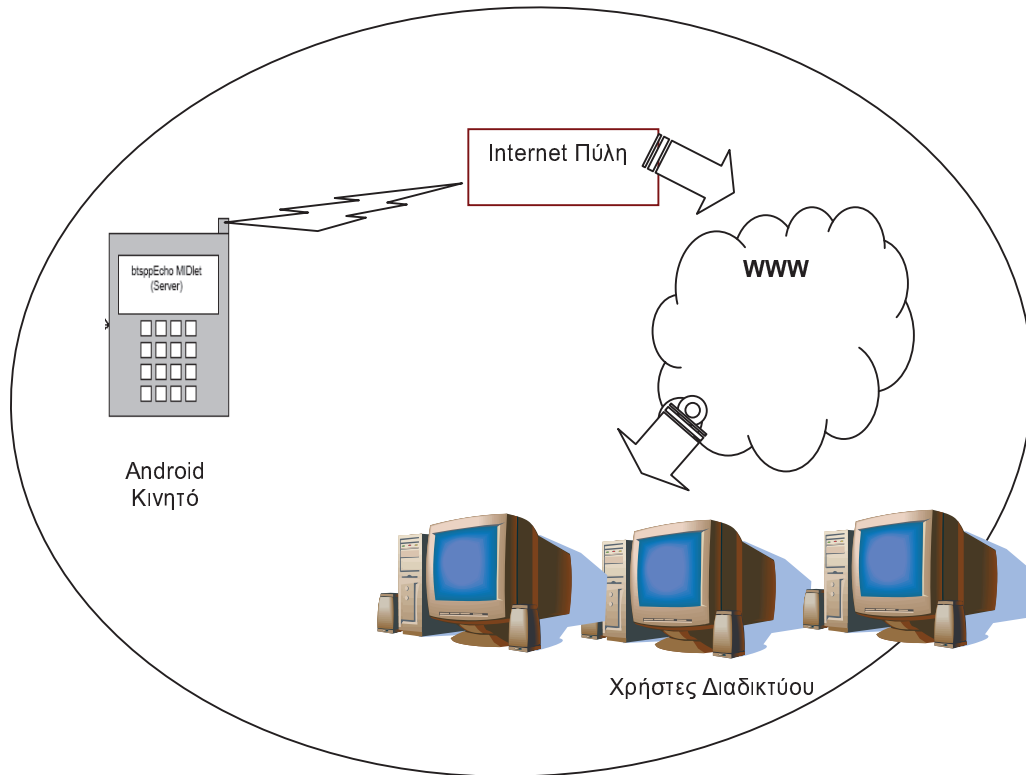
## **Σχεδιασμός**

### **Σχεδιασμός Δικτύου**

Όπως έχουμε ήδη περιγράψει το σύστημα μας βασίζεται στην ανάπτυξη ενός δικτύου ευρείας περιοχής (WAN) που χρησιμοποιεί ως μέσο μετάδοσης των δεδομένων το Internet και ως φυσικό μέσο το ασύρματο δίκτυο της κινητής τηλεφωνίας. Η Client – Server αρχιτεκτονική θα χρησιμοποιηθεί για την ανάπτυξη του λογισμικού που θα εκτελείται στις συσκευές και τον κεντρικό υπολογιστή. Μια client εφαρμογή θα αναπτυχθεί για να εκτελείται στις συσκευές κινητής τηλεφωνίας. Η εφαρμογή αυτή θα αναπτυχθεί για το λειτουργικό σύστημα Android.

Από την άλλη πλευρά, η server εφαρμογή θα αναπτυχθεί σε PHP και θα αφορά στην ανάπτυξη μιας web εφαρμογής που θα επιτρέπει ταυτόχρονα τη διαχείριση των σταθμών, την περισυλλογή των δεδομένων αλλά και την ανάρτηση των δεδομένων σε έναν ιστότοπο που θα είναι προσβάσιμος από τους χρήστες του Παγκόσμιου Ιστού. Η client εφαρμογή θα στέλνει δεδομένα σε μια διεύθυνση του Παγκόσμιου Ιστού η οποία θα ανήκει στον server του συστήματος και θα λαμβάνονται από την αντίστοιχη εφαρμογή που θα τρέχει σε αυτόν. Στην πραγματικότητα τα δεδομένα αυτά στέλνονται από τη συσκευή της κινητής τηλεφωνίας στον server δια μέσου ενός Gateway Server που ανήκει στην εταιρία κινητής τηλεφωνίας που εξυπηρετεί το δίκτυο στο οποίο ανήκουν οι τηλεφωνικές συσκευές. Ο server αυτός που λειτουργεί σαν Πύλη προωθεί τα δεδομένα στην IP διεύθυνση μέσω του Internet.

Η συνολική εικόνα του δικτύου εμφανίζεται στο Σχήμα



Σχήμα 5. Αποστολή δεδομένων κινητού μέσω Πύλης

### Σχεδιασμός Βάσης Δεδομένων

Όπως έχουμε ήδη προαναφέρει η αρχιτεκτονική του συστήματος στηρίζεται στο μοντέλο 3-tier. Τα εισερχόμενα δεδομένα αποστέλλονται από μια client εφαρμογή και λαμβάνονται από μια server εφαρμογή για να προβληθούν στην οθόνη από τη web εφαρμογή. Ένα Σχεσιακό Σύστημα Διαχείρισης Βάσεων Δεδομένων (Relational Database Management System – RDBMS) θα είναι επίσης εγκατεστημένο στην πλευρά του server για την αποθήκευση των εισερχόμενων δεδομένων.

Μια από τις οντότητες για την οποία κρατώνται πληροφορίες στη Βάση Δεδομένων είναι η συσκευή-σταθμός. Για κάθε συσκευή θα πρέπει να καταχωρείται η κατηγορία της, η γεωγραφική της θέση με βάση το γεωγραφικό μήκος και πλάτος. Επίσης θα πρέπει να υπάρχει και ένα όνομα

για κάθε συσκευή-σταθμό που θα τη χαρακτηρίζει μοναδικά. Η οντότητα της συσκευής υλοποιείται με τη χρήση ενός πίνακα που καλείται tblPlace. Το πεδίο περιέχει τον κωδικό της συσκευής και είναι το πρωτεύον κλειδί του πίνακα γιατί χαρακτηρίζει μοναδικά κάθε εγγραφή. Το πεδίο mapXY περιέχει το γεωγραφικό μήκος και το γεωγραφικό πλάτος της θέσης της συσκευής, το πεδίο address τη ταχυδρομική διεύθυνση, το category την κατηγορία του σταθμού και το description την περιγραφή του.

<b>tblPlace</b>
ID
MapXY
Address
Category
Title
Description

Επίσης η κατηγορία στην οποία ανήκει ο σταθμός αποτελεί μια ξεχωριστή οντότητα η οποία αναπαρίσταται από τον πίνακα tblCategory. Ο πίνακας έχει πρωτεύον κλειδί το πεδίο Id, την περιγραφή της κατηγορίας (Description) και αναλυτικές λεπτομέρειες της κατηγορίας (Details). Επίσης αν είναι υποκατηγορία κάποιας μεγαλύτερης κατηγορίας σταθμών τότε αυτό προσδιορίζεται από το πεδίο hypercat.

<b>tblCategory</b>
ID
Description
Hypercat
Details

Επιπλέον τα δεδομένα που αποστέλλονται από τη συσκευή μέτρησης είναι επίσης μια οντότητα του συστήματος. Τα δεδομένα αυτά αποτελούνται από την τιμή της θερμοκρασίας. Επίσης τα δεδομένα αποθηκεύονται μαζί με το όνομα-κωδικό της συσκευής που τα έστειλε καθώς και την ημερομηνία και ώρα που λήφθηκαν. Η οντότητα της μέτρησης υλοποιείται με τον πίνακα tblData.

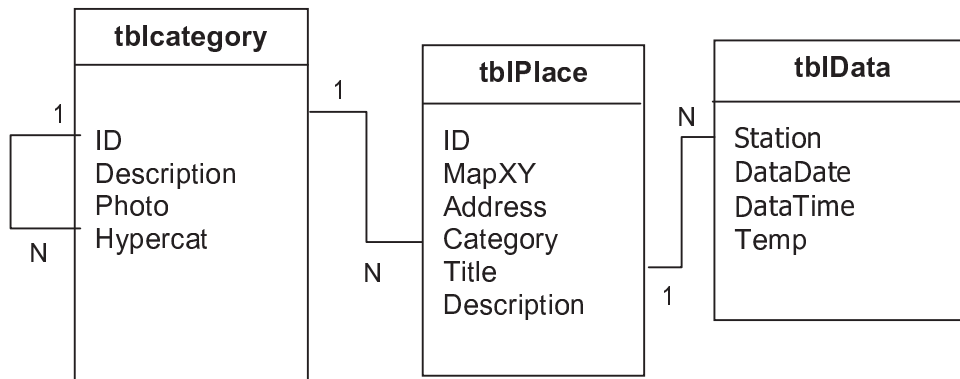
Το πεδίο Station περιέχει το κωδικό όνομα της συσκευής που έστειλε τα δεδομένα και αποτελεί εξωτερικό κλειδί προς τον πίνακα tblPlace. Τα πεδία DataDate και DateTime αντίστοιχα περιέχουν την ημερομηνία και ώρα που παραλήφθηκαν τα δεδομένα ενώ το πεδίο Temp περιέχει τη θερμοκρασία. Το πρωτεύον κλειδί του πίνακα είναι ο συνδυασμός Station, DataDate και DateTime.

<b>tblData</b>
Station DataDate DateTime Temp

Τέλος υπάρχει ένας πίνακας tblMember ο οποίος αποθηκεύει τα στοιχεία των διαχειριστών του συστήματος.

<b>tblMember</b>
Id FirstName LastName Address Phone Email Username Password UserType

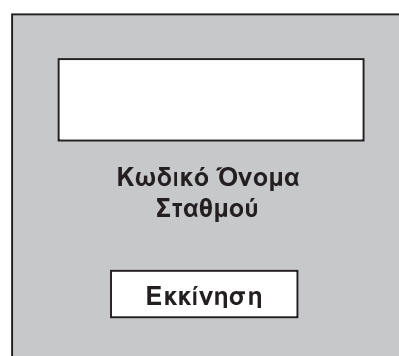
Το Διάγραμμα Οντοτήτων Σχέσεων περιγράφει το μοντέλο της βάσης δεδομένων:



**Σχήμα 6.** Διάγραμμα Οντοτήτων Σχέσεων

### User Interface

Οι απαιτήσεις όσον αφορά το σχεδιασμό του UI χωρίζονται σε δύο μέρη. Το μεν πρώτο αφορά το UI της client εφαρμογής που θα εκτελείται στη συσκευή κινητής τηλεφωνίας. Το UI της εφαρμογής αυτής είναι σχετικά απλό διότι οι διαδικασίες χειρισμού της είναι ελάχιστες και αφορούν στην αρχικοποίησή της κατά την έναρξη λειτουργίας της. Καταρχήν θα πρέπει να υπάρχει η δυνατότητα προσδιορισμού του κωδικού της συσκευής ο οποίος θα συνοδεύει τα δεδομένα. Το παρακάτω σχήμα περιγράφει τη βασική οθόνη της client εφαρμογής.



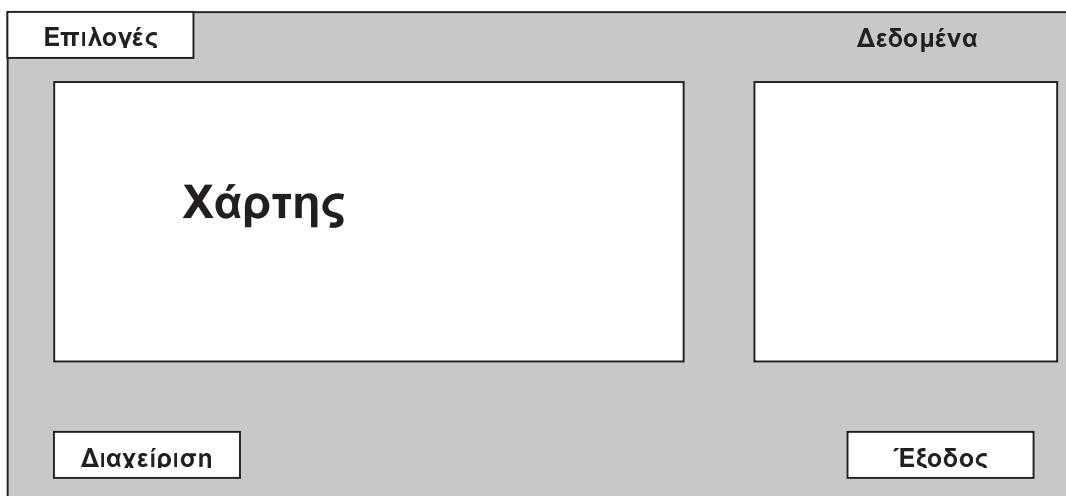


## Σχήμα 7

Από την άλλη πλευρά οι λειτουργίες της web εφαρμογής είναι πολύ περισσότερες και απαιτούν ένα πιο ολοκληρωμένο γραφικό περιβάλλον χειρισμού της εφαρμογής. Ένα περιβάλλον διεπαφής είναι στην πραγματικότητα ένα σύστημα Διεπαφής Ανθρώπου – Μηχανής (Human Machine Interface - HMI).

Οι βασικές λειτουργίες είναι αυτές της παρακολούθησης των δεδομένων, της προσθήκης μιας νέας συσκευής μέτρησης καθώς και η δυνατότητα διαγραφής της.

Η κεντρική οθόνη αποτελείται από έναν χάρτη στον οποίο εμφανίζεται η υπό παρακολούθηση περιοχή καθώς και τα σημεία από τα οποία αποστέλλονται δεδομένα. Έτσι τα δεδομένα εμφανίζονται και σε μια λίστα κατά χρονολογική σειρά. Η κεντρική οθόνη περιγράφεται από το παρακάτω σχήμα.



Σχήμα 8 Οθόνη Παρακολούθησης

Πέρα από την κεντρική οθόνη υπάρχουν και βοηθητικές φόρμες για τις διαχειριστικές λειτουργίες όπως είναι η εισαγωγή και διαγραφή των συσκευών μέτρησης και αποστολής δεδομένων. Έτσι θα πρέπει να υπάρχει μια φόρμα για την εισαγωγή ενός νέου σημείου μέτρησης στο χάρτη και στη βάση δεδομένων. Για το νέο αυτό αντικείμενο θα πρέπει να προσδιορίζεται ο κωδικός του καθώς και η γεωγραφική του θέση έτσι ώστε αυτό να εμφανίζεται και στο κατάλληλο σημείο στο χάρτη.

## Υλοποίηση

### Στάδια Υλοποίησης

Η υλοποίηση του συστήματος απαιτεί τη δημιουργία 4 ξεχωριστών ενοτήτων:

- Η υλοποίηση μιας client εφαρμογής η οποία εγκαθιστά μια επικοινωνία με τον server και αποστέλλει δεδομένα από το κινητό. Η εφαρμογή αυτή υλοποιήθηκε για Android περιβάλλον.
- Η υλοποίηση μιας web εφαρμογής η οποία λαμβάνει τα δεδομένα από το κινητό και τα προβάλλει στην οθόνη και τα αποθηκεύει με τη βοήθεια ενός συστήματος ΒΔ. Η εφαρμογή αυτή υλοποιήθηκε ως server side εφαρμογή με τη χρήση PHP και MySQL καθώς και τη χρήση javascript για τον Google Map.
- Η υλοποίηση μιας web εφαρμογής η οποία αναλαμβάνει τη διαχείριση των σταθμών, τη δημιουργία και κατάργησή τους. Η εφαρμογή αυτή υλοποιήθηκε ως server side εφαρμογή με τη χρήση PHP και MySQL.
- Η δημιουργία μιας βάσης δεδομένων για την αποθήκευση πληροφοριών. Η βάση δεδομένων αναπτύχθηκε σε MySQL DBMS και η εφαρμογή στον κεντρικό υπολογιστή αποθηκεύει τα εισερχόμενα δεδομένα καθώς και πληροφορίες για τις συσκευές μέτρησης σε πίνακες της βάσης.

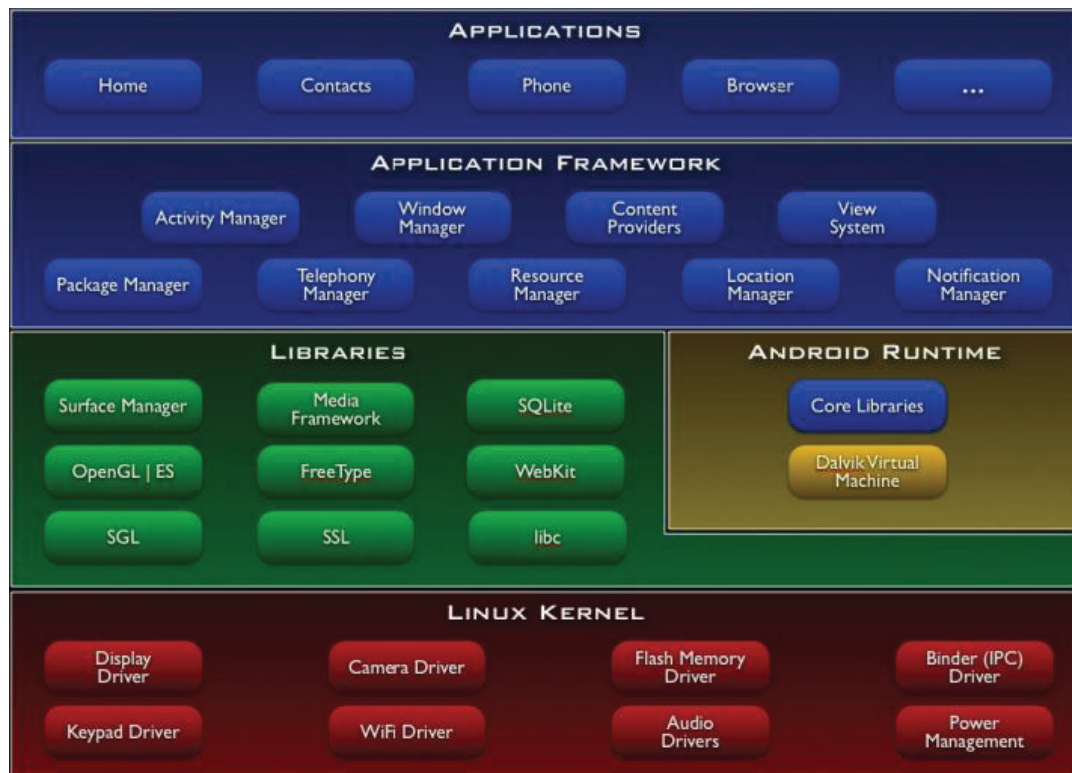
### Client εφαρμογή στο κινητό

Όπως έχουμε ήδη περιγράψει, η εφαρμογή που θα είναι εγκατεστημένη και θα εκτελείται σε μια συσκευή κινητής τηλεφωνίας, λαμβάνει τα δεδομένα από έναν αισθητήρα συνδεδεμένο με την έξυπνη συσκευή και τα αποστέλλει στον server μέσω μιας σύνδεσης TCP/IP μέσω mobile Internet. Στην πραγματικότητα η εφαρμογή αυτή θα πρέπει με κάποια ρουτίνα να διαβάζει τις τιμές αυτές από το εξωτερικό περιβάλλον αλλά στη συγκεκριμένη υλοποίηση έχει δημιουργηθεί μια συνάρτηση παραγωγής τυχαίων αριθμών οι οποίοι προσομοιώνουν τα δεδομένα θερμοκρασιών. Η εφαρμογή γράφτηκε με τη χρήση των εργαλείων Eclipse IDE για Java, Android SDK και ADT (Android Development Tools) Plug-in για το Eclipse

### Android

Το Android είναι ένα λειτουργικό σύστημα που βασίζεται στον πυρήνα του kernel. Το project αφορά την ανάπτυξη του συστήματος Android ονομάζεται Android Open Source Project

(AOSP) και το οποίο εισήγαγε η Google. Παρέχει ένα επίπεδο αφαιρετικότητας ανάμεσα στο υλικό και τον χρήστη που δίνει την δυνατότητα στον χρήστη να χρησιμοποιεί τους πόρους του συστήματος προς όφελος του με μια διεπαφή περισσότερο κατανοητή προς τον άνθρωπο. Το Android, αποτελείται από μία στοίβα τμημάτων (software stack) με καθορισμένους ρόλους τα οποία και βλέπουμε στην Εικόνα .



**Σχήμα 9. Στοίβα συστατικών του Android**

Στο κατώτερο επίπεδο βρίσκεται ο πυρήνας του Linux. Πάνω από τον πυρήνα του Linux βρίσκονται οι βασικές βιβλιοθήκες του όπως γραφικών, βάσεων δεδομένων, η βιβλιοθήκη της γλώσσας προγραμματισμού C και άλλα. Το επίπεδο αυτό προσφέρει τα στοιχεία που χρειάζονται για την υλοποίηση του Android Runtime, του βασικού μηχανισμού, δηλαδή, που απαιτείται για την εκτέλεση των εφαρμογών που αναπτύσσονται για το περιβάλλον του Android. Το Android Runtime περιλαμβάνει την εικονική μηχανή (virtual machine) που μεταφράζει κι εκτελεί τις εφαρμογές Android καθώς και την υλοποίηση των βασικών βιβλιοθηκών της Java. Το προτελευταίο επίπεδο είναι το ονομαζόμενο Application Framework. Το επίπεδο αυτό προσφέρει στον προγραμματιστή μια πληθώρα δυνατοτήτων σχετιζόμενων με το λειτουργικό σύστημα και την συσκευή που το φιλοξενεί. Τέλος, το ανώτερο επίπεδο της στοίβας είναι το επίπεδο όπου βρίσκονται οι εφαρμογές για το περιβάλλον του Android. Σε αυτό το επίπεδο έχει αναπτυχθεί η εφαρμογή που περιγράφεται στην εργασία.

## Εγκατάσταση εργαλείων ανάπτυξης

Τα βασικά εργαλεία που χρειάζονται για την ανάπτυξη της εφαρμογής στις αντίστοιχες εκδόσεις του λειτουργικού στο οποίο θα αναπτυχθεί η εφαρμογή είναι τα εξής:

- Java Development Kit
- Eclipse IDE για Java
- Android SDK
- ADT (Android Development Tools) Plug-in για το Eclipse

Όλα τα απαραίτητα εργαλεία είναι διαθέσιμα δωρεάν. Αρχικά γίνεται εγκατάσταση του Java Development Kit. Το JDK περιλαμβάνει ένα σύνολο εργαλείων ανάπτυξης προγραμμάτων για την γλώσσα προγραμματισμού Java όπως είναι ο java (φορτωτής Java εφαρμογών), ο javac (ο compiler για Java κώδικα σε Java bytecodes), ο jar (ο γνωστός Java archiver) και άλλα. Το JDK είναι το υποσύνολο του Java SDK που είναι απαραίτητο για τον προγραμματισμό και την εκτέλεση εφαρμογών σε Java. Το αρχείο εγκατάστασης είναι διαθέσιμο από τον ιστότοπο της Oracle.

Στη συνέχεια θα πρέπει να γίνει η εγκατάσταση του Eclipse IDE (Integrated Development Environment). Πρόκειται για ένα ολοκληρωμένο περιβάλλον ανάπτυξης εφαρμογών που διευκολύνει την ανάπτυξη εφαρμογών αν και δεν είναι απαραίτητη η χρήση του αφού η όλη εφαρμογή θα μπορούσε να γραφεί με τη χρήση text editor. Με την υποστήριξη των plug-in το Eclipse μπορεί να επεκτείνει τις λειτουργίες που προσφέρει. Έτσι για την ανάπτυξη των Android εφαρμογών γίνεται εγκατάσταση του ADT Plug-in που αναφέρουμε στη συνέχεια. Εναλλακτικά μπορούμε να χρησιμοποιήσουμε και το Android Studio.

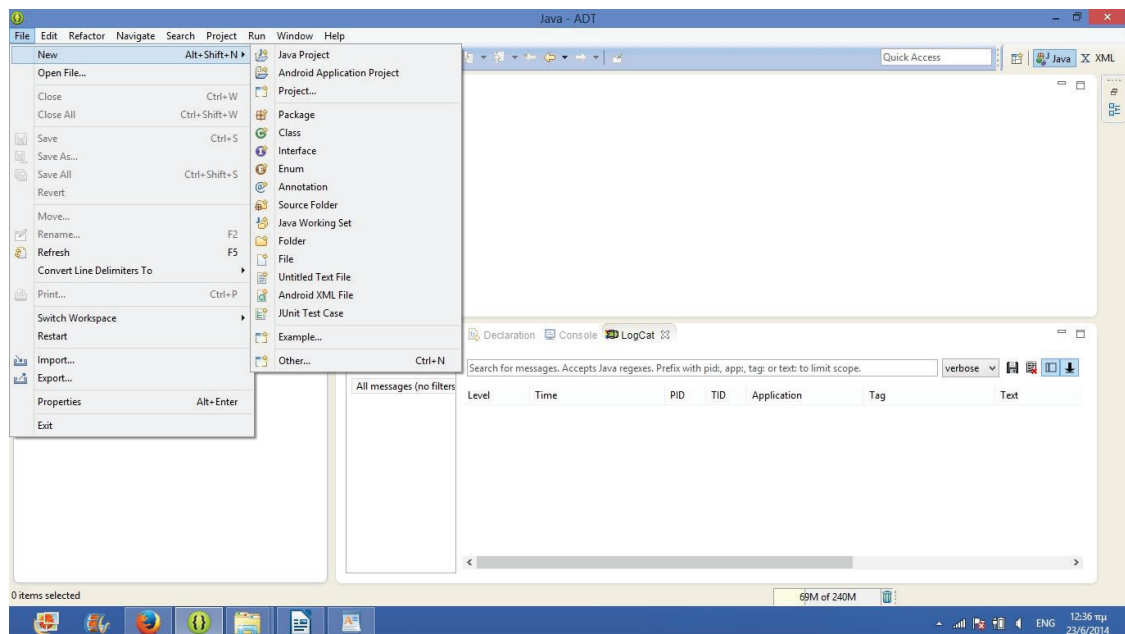
Μετά την εγκατάσταση του Eclipse, ακολουθεί η εγκατάσταση του Android SDK , απαραίτητο για την ανάπτυξη εφαρμογών για Android. Περιλαμβάνει τις απαραίτητες βιβλιοθήκες για την ανάπτυξη εφαρμογών Android, παραδείγματα εφαρμογών με τον πηγαίο κώδικά τους και επίσης έναν εξομοιωτή συσκευών smartphone (*Android Virtual Device* - AVD). Το SDK είναι διαθέσιμο στον ιστότοπο της Google στην περιοχή Google developer.

Τέλος η εγκατάσταση του ADT (Android Development Tools) Plug-in για το Eclipse δίνει τη δυνατότητα στους προγραμματιστές να σχεδιάσουν τις εφαρμογές Android με γραφικό τρόπο μέσα από ένα περιβάλλον παρόμοιο με το Visual Studio και επίσης δίνει τη δυνατότητα να

δημιουργηθεί ένα καινούριο project Android με την σωστή δομή και αρχεία αυτοματοποιημένα εντός του περιβάλλοντος.

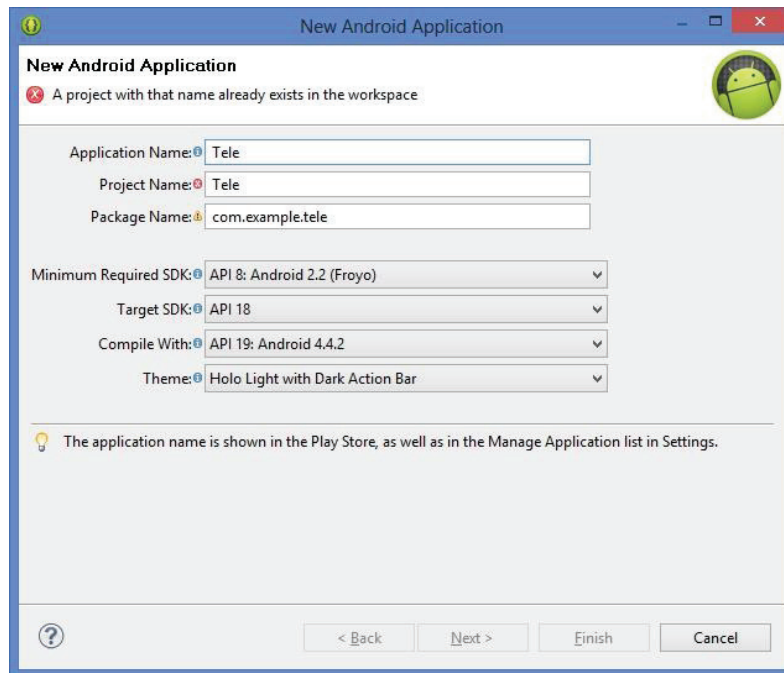
## Περιβάλλον Ανάπτυξης Εφαρμογής Android

Ξεκινώντας από το Eclipse, μέσα στο περιβάλλον ανάπτυξης της εφαρμογής επιλέγουμε την εκκίνηση ενός νέου Android project.

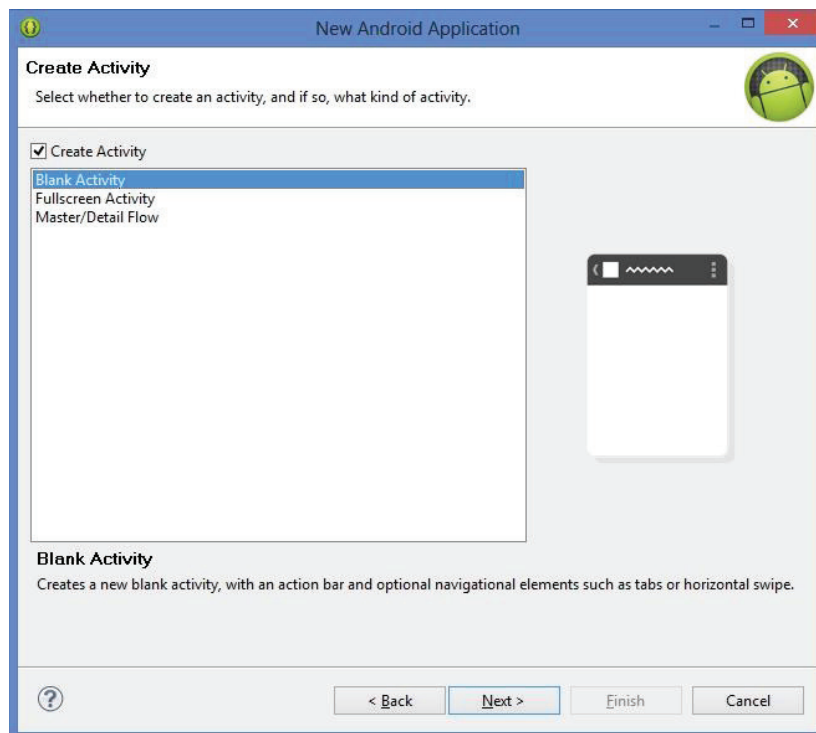


Εικόνα 1

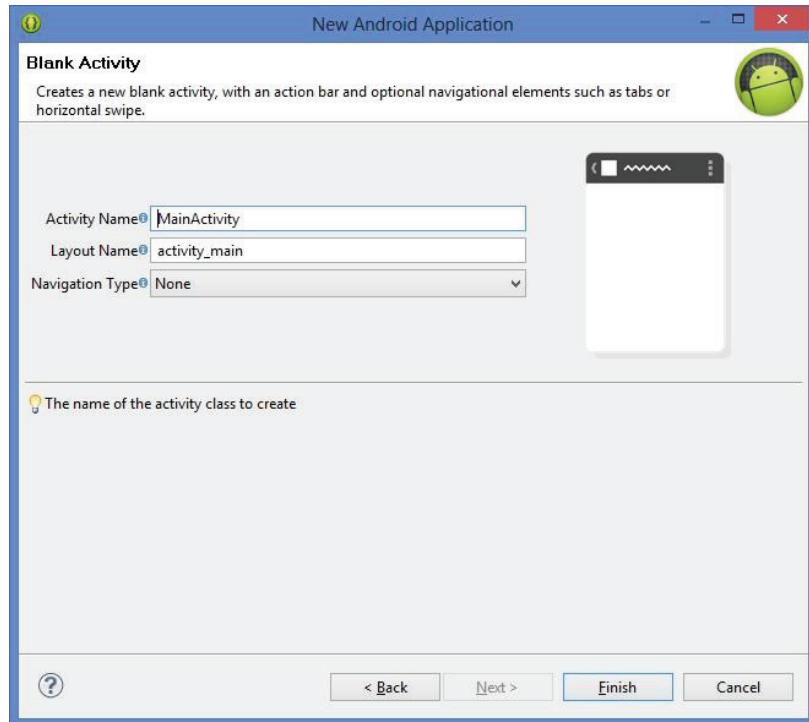
Επιλέγοντας την κενή κεντρική activity με όνομα MainActivity, στη συνέχεια δημιουργούνται αυτόματα όλα τα απαραίτητα αρχεία για την ανάπτυξη της εφαρμογής.



Εικόνα 2



Εικόνα 3



Εικόνα 4

Ένα Android project περιέχει όλα τα αρχεία του πηγαίου κώδικα. Συνοψίζοντας τη δομή των καταλόγων και τα βασικά αρχεία που αυτοί περιέχουν σε ένα Android project, αυτά είναι:

## AndroidManifest.xml

Το manifest file περιγράφει τα βασικά χαρακτηριστικά της εφαρμογής και ορίζει κάθε ένα από τα συστατικά.

Ένα από τα πιο βασικά στοιχεία του αρχείου manifest είναι το στοιχείο `<uses-sdk>` το οποίο δηλώνει την συμβατότητα της εφαρμογής με διαφορετικές εκδόσεις του Android. Όσο μικρότερη είναι η υποστηριζόμενη έκδοση τόσο μεγαλύτερη είναι η συμβατότητα με περισσότερες εκδόσεις. Στη συγκεκριμένη εφαρμογή το `AndroidManifest.xml` περιέχει τα εξής:

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
  package="com.example.tele"  
  android:versionCode="1"  
  android:versionName="1.0" >
```

```
  <uses-sdk  
    android:minSdkVersion="8"  
    android:targetSdkVersion="18" />
```

```

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

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

</manifest>

```

Ένας από τους σημαντικότερους καταλόγους είναι ο κατάλογος `src/` ο οποίος περιέχει τον πηγαίο κώδικα και κυρίως την κλάση `Activity` η οποία εκτελείται όταν ξεκινάει η εφαρμογή.

Επίσης ο κατάλογος των `resources (res/)` είναι από τους σημαντικούς για την ανάπτυξη καταλόγους διότι περιέχει την περιγραφή των πόρων της εφαρμογής. Ένας από τους υποκαταλόγους είναι ο `layout/`. Αυτός περιέχει το `layout` της εφαρμογής. Στη συγκεκριμένη εφαρμογή, το `tele manifest` είναι το αρχείο που περιγράφει το `layout` και τα βασικά συστατικά του:

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal" >

    <EditText
        android:id="@+id/edit_message"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:hint="@string/edit_message" />

        <Button
            android:id="@+id/start"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Start"/>

</LinearLayout>

```



Σύμφωνα με τα παραπάνω η εφαρμογή αποτελείται από δύο βασικά στοιχεία. Το ένα είναι ένα Start button το οποίο όπως θα δούμε εκκινεί την αποστολή δεδομένων. Επίσης το edit πεδίο είναι αυτό στο οποίο ο διαχειριστής εισάγει την κωδική ονομασία της συσκευής.

Επίσης ο κατάλογος values/ περιέχει πληροφορίες για μεταβλητές και τιμές μεταβλητών που χρησιμοποιούνται στην εφαρμογή. Οι μεταβλητές που χρησιμοποιούνται περιέχονται στο αρχείο strings.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>

    <string name="app_name">Tele</string>
    <string name="action_settings">Settings</string>
    <string name="edit_message">Type StationID</string>

</resources>
```

## MainActivity.java

Η MainActivity είναι η εναρκτήρια κλάση του project. Η συνάρτηση onCreate αφού εκκινήσει την λειτουργία της εφαρμογής, κατασκευάζει το layout της εφαρμογής και προσθέτει έναν listener στο start button της εφαρμογής. Όταν γίνει κλικ στο κουμπί καλείται η συνάρτηση SummaryAsyncTask().execute(). Η συνάρτηση αυτή εκτελεί μια άλλη λειτουργία ασύγχρονα.

Η ασύγχρονη εκτέλεση κάποιας λειτουργίας υλοποιείται με τη χρήση thread. Η χρήση της κλάσης AsyncTask δίνει τη δυνατότητα εκτέλεσης κατάλληλων και ευκολων νημάτων πολυνημάτωσης (threads). Στην προκειμένη περίπτωση η χρήση πολυνημάτωσης ήταν ενδεδειγμένη διότι η λειτουργία που εκτελείται παράλληλα είναι η σύνδεση και αποστολή κάποιων μεταβλητών μέσω μιας http σύνδεσης. Όταν πατηθεί το start button, η εφαρμογή εκτελεί το παράλληλο thread με τη χρήση της συνάρτησης doInBackground(). Η συνάρτηση καλεί την postData() η οποία αναλαμβάνει μέσα σε ένα loop να διαβάζει δεδομένα από το εξωτερικό περιβάλλον όπως τη θερμοκρασία περιβάλλοντος. Στην υλοποίηση η καταγραφή θερμοκρασίας προσομοιώνεται με τη χρήση μιας συνάρτησης παραγωγής ψευδο-τυχαίων ακεραίων, την int getTemperature(). Στη συνέχεια εγκαθιστά μια http σύνδεση με τον ιστότοπο που περιέχει την web εφαρμογή για την αποθήκευση και προβολή των δεδομένων και αποστέλλει ένα ζεύγος τιμών για τις μεταβλητές station και temp στη σελίδα getdata.php.

```

public class MainActivity extends Activity {

    String StationID;
    private EditText editStationID;
    private Button button;

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

        editStationID = (EditText) findViewById(R.id.edit_message);
        button = (Button) findViewById(R.id.start);
        button.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View v) {
                StationID = editStationID.getText().toString();
                new SummaryAsyncTask().execute((Void) null);
            }
        });
    }
}

class SummaryAsyncTask extends AsyncTask<Void, Void, Boolean> {
    private int getTemperature()
    {
        Random r = new Random();
        int i1 = r.nextInt(60) - 20;
        return i1;
    }
    private void postData(String StationID) {

        HttpClient httpclient = new DefaultHttpClient();
        HttpPost httppost = new HttpPost("http://www.smptelemetry.comxa.com/getData.php");
        for(;;)
        try {
            Thread.sleep(1000);
            ArrayList<NameValuePair> nameValuePairs = new ArrayList<NameValuePair>(4);
            String temp;
            temp=Integer.toString(getTemperature());
            nameValuePairs.add(new BasicNameValuePair("Station", StationID));
            nameValuePairs.add(new BasicNameValuePair("Temp", temp));
            httppost.setEntity(new UrlEncodedFormEntity(nameValuePairs));
            HttpResponse response = httpclient.execute(httppost);
        }
        catch(Exception e)
        {
            Log.e("log_tag", "Error: "+e.toString());
        }
    }
}

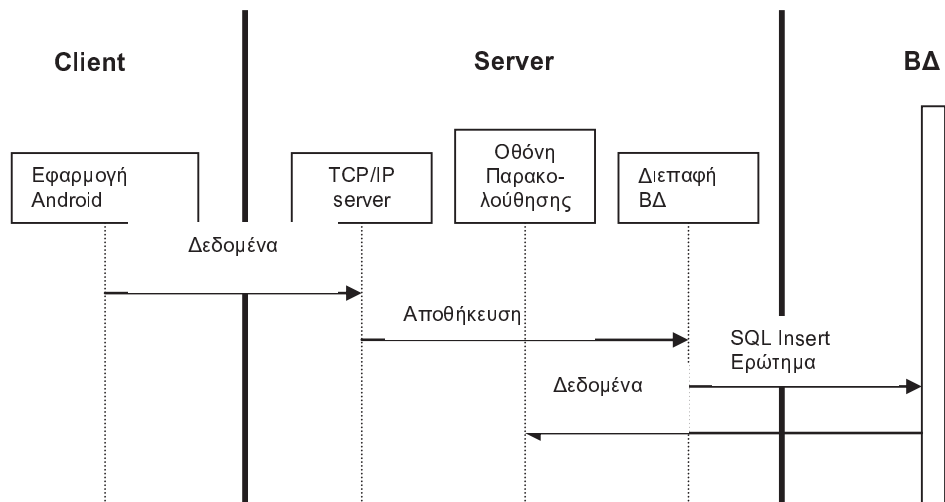
```

```
@Override
protected Boolean doInBackground(Void... params) {
    postData(StationID);
    return null;
}
}
```

## Υλοποίηση Βάσης Δεδομένων

Όπως έχουμε ήδη αναφέρει η server εφαρμογή είναι συνδεδεμένη με ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων όπου τα δεδομένα αποθηκεύονται σε μια Σχεσιακή ΒΔ. Τα εισερχόμενα δεδομένα αποθηκεύονται στη Βάση για μελλοντική χρήση και επεξεργασία αμέσως μόλις ληφθούν από τη σύνδεση με τις client εφαρμογές. Η server εφαρμογή συνδέεται με το σύστημα ΒΔ και αποθηκεύει τα δεδομένα εκτελώντας κατάλληλες SQL εντολές στους αντίστοιχους πίνακες. Όπως είδαμε η σύνδεση με τη ΒΔ γίνεται χρησιμοποιώντας έναν driver σύνδεσης με τη ΒΔ.

Στη συγκεκριμένη υλοποίηση το σύστημα διαχείρισης της ΒΔ είναι η πλατφόρμα MySQL. Η MySQL που αποτελείται από ένα σύστημα διαχείρισης σχεσιακών βάσεων και από έναν server της βάσης, αναπτύχθηκε από την εταιρία λογισμικού MySQL AB. Υποστηρίζει λειτουργίες multi-threading και multi-user. Είναι επίσης απόλυτα συμβατή με την γλώσσα ερωτημάτων standard SQL (Structured Query Language) και με μια μεγάλη γκάμα πλατφόρμων. Είναι επίσης σημαντικό ότι ανήκει στην κατηγορία λογισμικού Ανοιχτού Κώδικα (Open Source) με ελεύθερα δικαιώματα εγκατάστασης και χρήσης υπό την άδεια GNU General Public License (GPL). Στην πραγματικότητα η MySQL χρησιμοποιεί ένα υβριδικό σύστημα άδειας. Οι προγραμματιστές μπορούν να χρησιμοποιήσουν την MySQL χωρίς χρέωση υπό την άδεια GPL ή μπορούν να την αγοράσουν με τα εμπορικά δικαιώματα της MySQL AB με τον όρο ότι δεν θα υπόκεινται στους όρους του GPL. Οι προγραμματιστές είναι υποχρεωμένοι να αγοράσουν τα εμπορικά δικαιώματα της MySQL αν αυτή δεν διατίθεται με εφαρμογή που δεν ανήκει στην κατηγορία Ανοιχτού Κώδικα.



Σχήμα 10 Διάγραμμα Ακολουθιών

Για την ανάπτυξη της ΒΔ χρησιμοποιήθηκε το γραφικό περιβάλλον του εργαλείου PHPMYAdmin. Το PHPMYAdmin είναι ένα web based εργαλείο που παρέχει ένα γραφικό περιβάλλον εργασίας για τη δημιουργία μιας ΒΔ και τη δημιουργία των πινάκων και των πεδίων τους καθώς επίσης και την εισαγωγή, τροποποίηση και διαγραφή δεδομένων και την εκτέλεση SQL ερωτημάτων. Επίσης μπορούν να δημιουργηθούν χρήστες και να οριστούν τα δικαιώματά τους σε κάθε πίνακα.

Στη συγκεκριμένη Βάση δημιουργήθηκαν οι πίνακες: tblPlace και tblCategory για την αποθήκευση των πληροφοριών σχετικών με τις συσκευές μέτρησης και ο tblData για την αποθήκευση των εισερχόμενων δεδομένων.

## Server Εφαρμογή

Η εφαρμογή συλλογής και παρουσίασης των δεδομένων είναι μια web based εφαρμογή υλοποιημένη με PHP και Javascript. Αποτελείται από δύο βασικά τμήματα:

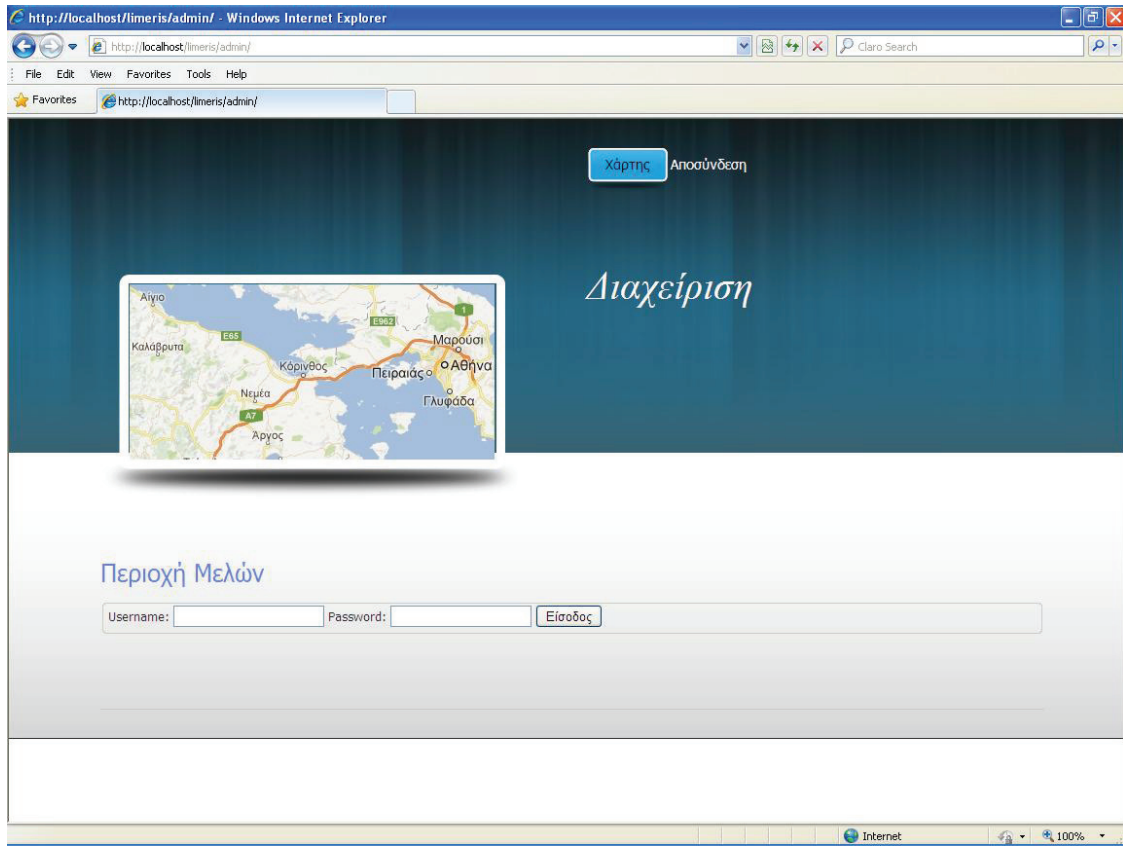
- Το τμήμα διαχείρισης των στοιχείων των συσκευών – σταθμών (π.χ. κωδικό όνομα σταθμού, γεωγραφικές συντεταγμένες, κατηγορία κα)
- Το τμήμα της προβολής των σταθμών και των δεδομένων σε μια ιστοσελίδα.

Οι λειτουργίες της web εφαρμογής απαιτούν τη σύνδεση με τη βάση δεδομένων. Αυτό επιτυγχάνεται από το αρχείο dbinfo.php το οποίο αντλεί τα στοιχεία πρόσβασης στη βάση από το αρχείο database.php.

## Διαχείριση

Η Διαχείριση των σταθμών γίνεται από μια ιστοσελίδα διαχείρισης με έλεγχο πρόσβασης. Στην περιοχή \admin, η σελίδα index.php αναλαμβάνει τη διαδικασία του login. Σελίδα περιέχει μια φόρμα για τη συμπλήρωση του username και password του διαχειριστή.

```
<form id="form1" method="post" action="loginadmin.php">
<fieldset>
<label for="inputtext1">Username:</label>
<input id="inputtext1" type="text" name="inputtext1" value="" size="20" />
<label for="inputtext2">Password:</label>
<input id="inputtext2" type="password" name="inputtext2" value="" size="20" />
<input id="inputsubmit1" type="submit" name="inputsubmit1" value="Είσοδος" />
</fieldset>
</form>
```



Εικόνα 5 Login

## Login

Τα στοιχεία username και password αποστέλλονται στη σελίδα loginadmin.php όπου εκτελείται το κατάλληλο select ερώτημα στον πίνακα tblmember για να διαπιστωθεί αν ο χρήστης είναι administrator.

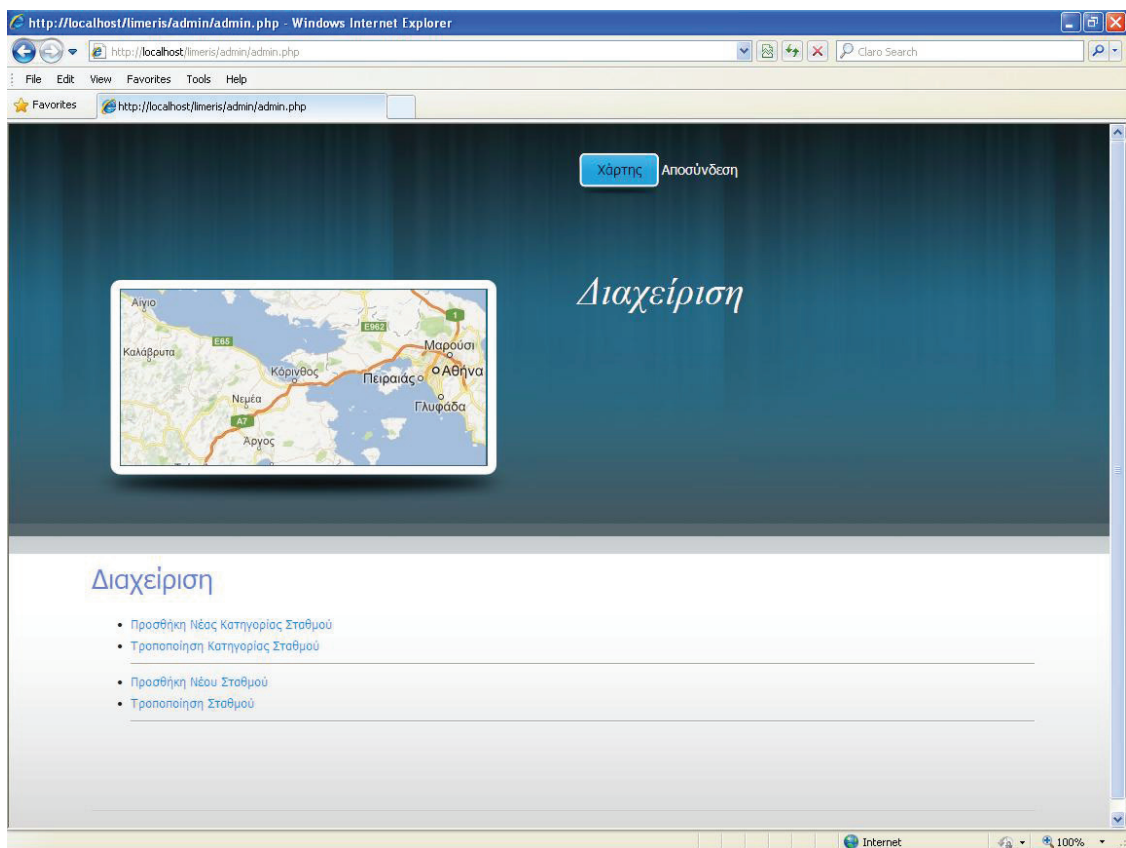
```
$username = addslashes($_POST['inputtext1']);  
$password=addslashes($_POST['inputtext2']);
```

```
$query=mysql_query("select * from member where username='".$username.'");
```

Στην περίπτωση που ο χρήστης είναι διαχειριστής, κατευθύνεται στη σελίδα admin.php.

## Σελίδα Διαχείρισης – Admin.php

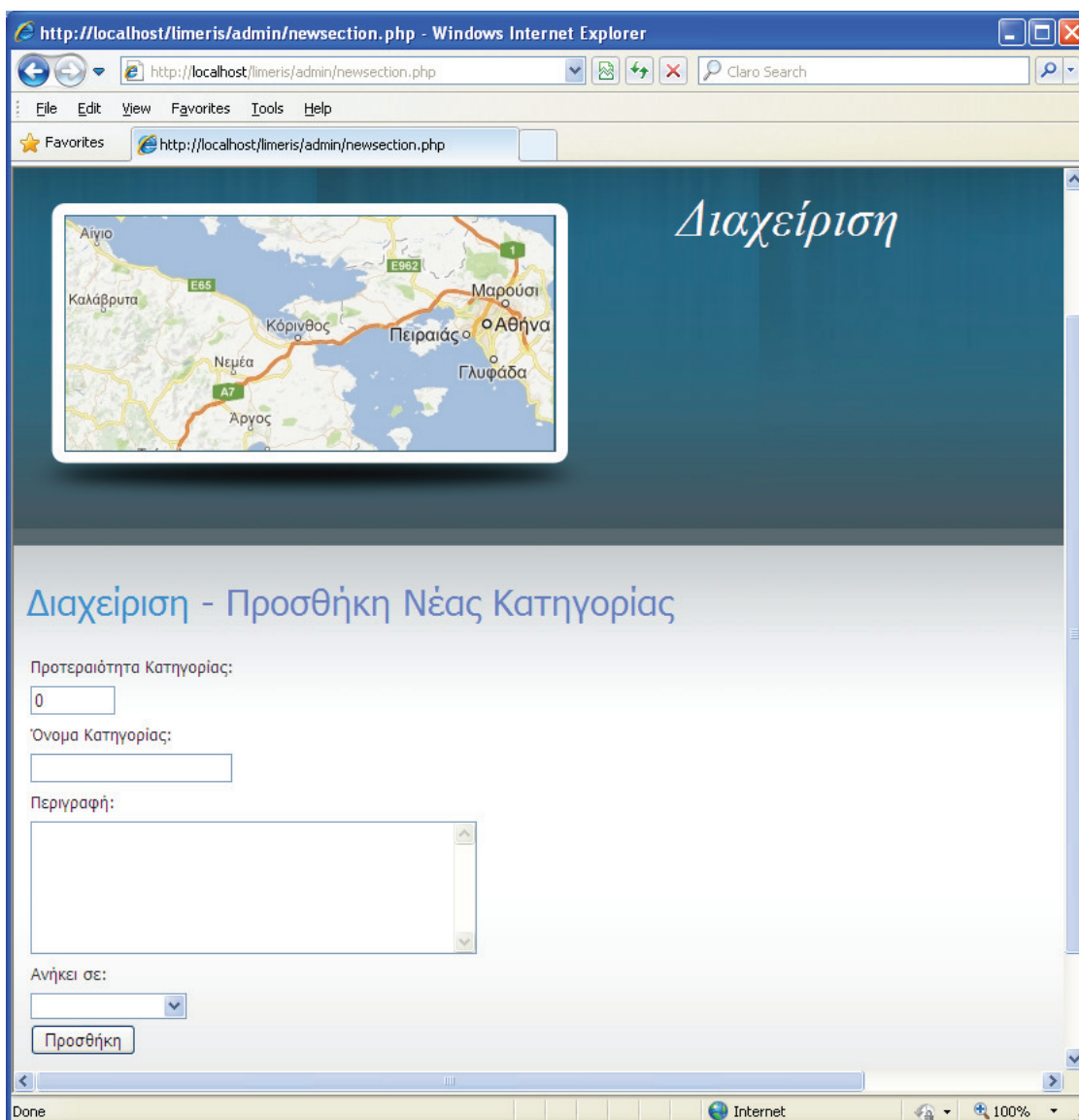
Στη σελίδα αυτή ο διαχειριστής μπορεί να προσθέσει μια νέα κατηγορία σταθμών ή να τροποποιήσει τις ήδη υπάρχουσες και επίσης μπορεί να προσθέσει τροποποιήσει ή διαγράψει συσκευές σταθμούς.



Εικόνα 6

## Προσθήκη Νέας Κατηγορίας Σταθμών

Στη προσθήκη σταθμών, υπάρχει μια φόρμα εισαγωγής του ονόματος και της περιγραφής της κατηγορίας. Επίσης υπάρχει η δυνατότητα αλλαγής της προτεραιότητας κατηγοριών και επίσης υπάρχει η δυνατότητα κάποια κατηγορία να ανήκει σε μια άλλη μεγαλύτερη κατηγορία σταθμών. Τα στοιχεία της φόρμας αποστέλλονται στη σελίδα `addsection.php`.



Εικόνα 7



Τα στοιχεία παραλαμβάνονται από τη σελίδα addsection.php και με τον παρακάτω κώδικα προστίθενται στον πίνακα tblcategory.

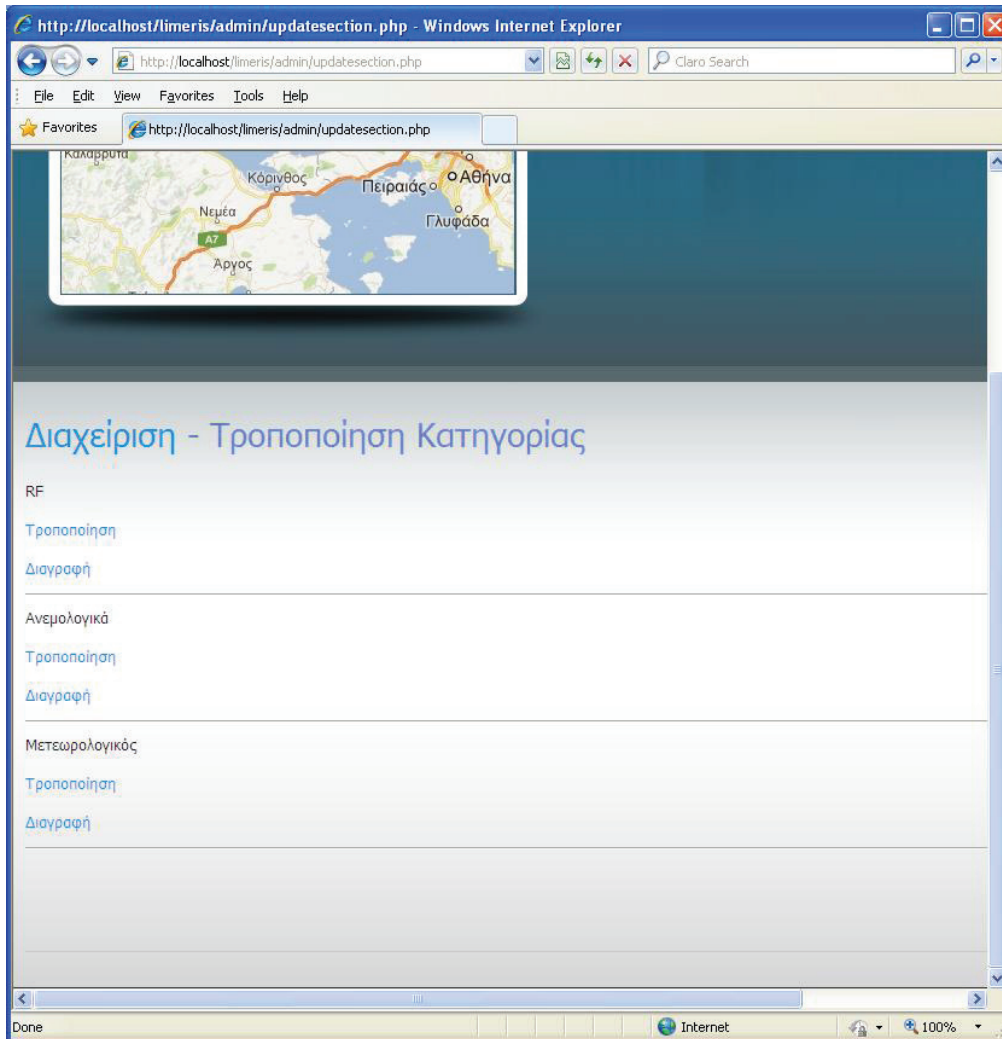
```
if (!(isset($_SESSION['usertype']))) || ($_SESSION['usertype'] != "admintp"){
header( 'Location: ../index.php');
}
include 'dbinfo.php';

$details = $_POST['details' ];
$priority=$_POST['priority'];
$description = $_POST['description' ];
$hypercat=$_POST['hypercat'];

mysql_query("insert into tblcategory(priority,details,description,hypercat)
values(".$priority.", ".$details.", ".$description.", ".$hypercat.)");
mysql_close($link);
```

## Τροποποίηση Κατηγορίας σταθμών

Αντίστοιχα για την τροποποίηση μιας ήδη υπάρχουσας κατηγορίας σταθμών υπάρχει η επιλογή τροποποίησης κατηγοριών η οποία ανοίγει τη σελίδα updatesection.php.



Εικόνα 8

Στη σελίδα αυτή εμφανίζονται όλες οι κατηγορίες ακολουθούμενες από την επιλογή Τροποποίηση / Διαγραφή. Στην περίπτωση που επιλεχθεί η διαγραφή της κατηγορίας τότε το id της κατηγορίας αποστέλλεται στη σελίδα deletesection.php όπου διαγράφεται η κατηγορία σταθμών.

```

$id=$_GET['section'];
include 'dbinfo.php';
mysql_query("delete from tblcategory where id=".$id);

```

Αντίστοιχα, με την επιλογή «Τροποποίηση», αποστέλλεται το id της κατηγορίας στη σελίδα altersection.php όπου υπάρχει μια φόρμα συμπλήρωσης των στοιχείων προς τροποποίηση. Τα στοιχεία αποστέλλονται στη σελίδα changesection.php όπου εκτελείται ένα ερώτημα update αφού γίνει πρώτα ο έλεγχος αν η κατηγορία υπάρχει.

```

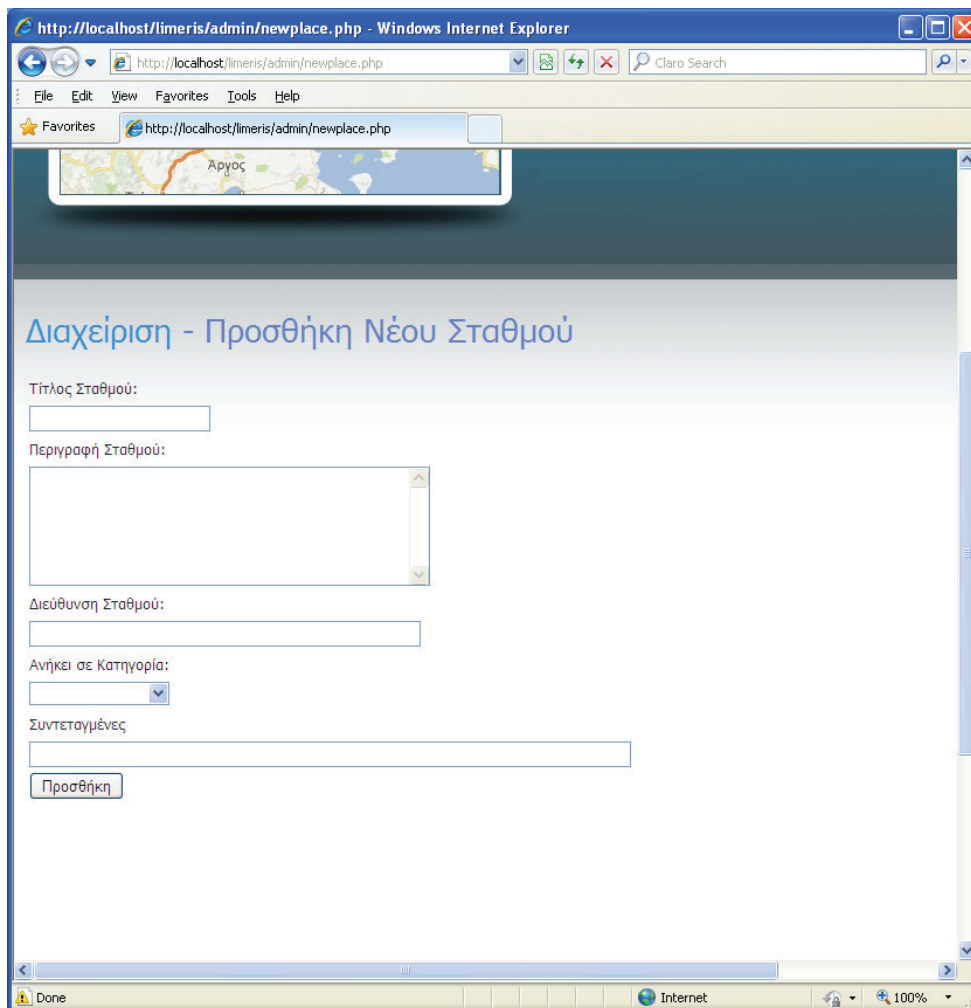
$priority = $_POST['priority'];
$details = $_POST['details'];
$description = $_POST['description'];
$hypercat=$_POST['hypercat'];
$id=$_POST['id'];
include 'dbinfo.php';

$query=mysql_query("select * from tblcategory where description='".$description.'" and id<>".$id);
if (!(mysql_num_rows($query)==0))
{
    mysql_close($link);
    ?>
    <script language="javascript">
    window.alert("Η κατηγορία υπάρχει");
    window.location="admin.php";
    </script>
<?php }
mysql_query("update tblcategory set
priority=".$priority.",details='".$details."',description='".$description."',hypercat='".$hypercat.'" where
id=".$id);
mysql_close($link);

```

## Προσθήκη Συσκευής – Σταθμού

Η σελίδα newplace.php δίνει τη δυνατότητα στο διαχειριστή να προσθέσει έναν καινούριο σταθμό. Αποτελείται από μια φόρμα εισαγωγής στοιχείων καθώς και από έναν χάρτη Google Map όπου ο διαχειριστής μπορεί να σημειώσει στο χάρτη τη θέση της συσκευής-σταθμού.



Εικόνα 9

Ο διαχειριστής συμπληρώνει το όνομα του σταθμού, τη διεύθυνση, την περιγραφή και την κατηγορία.

Με τη χρήση του API της Google για τους χάρτες Google Map, η εφαρμογή εντοπίζει τις συντεταγμένες που ο διαχειριστής κάνει κλικ στο χάρτη και ενημερώνει το πεδίο Συντεταγμένες. Στον html κώδικα έχει προστεθεί το εξής τμήμα το οποίο ορίζει έναν χάρτη με id map\_canvas:

```
<DIV id="map_canvas" style="WIDTH: 500px; HEIGHT: 300px"></DIV>
```

Η αρχικοποίηση του χάρτη επιτυγχάνεται με την κλήση της συνάρτησης initialize() κατά στην αρχή του σώματος της σελίδας με τον εξής κώδικα JavaScript:

```
function initialize() {
  if (GBrowserIsCompatible()) {
    var map = new GMap2(document.getElementById("map_canvas"));
    map.setCenter(new GLatLng(39.0618, 22.0605),3);
    //Αντικείμενο zoom control
    map.addControl(new GLargeMapControl());
    //Αντικείμενο type control
    var mapControl = new GMapTypeControl();
    map.addControl(mapControl);
    GEvent.addListener(map, "click", function() {
      var center = map.getCenter();
      var photoxy=document.getElementById("photoXY");
      photoxy.value=center.toString();
    });
  }
}
```

Ο παραπάνω κώδικας ορίζει έναν χάρτη, του οποίου το κέντρο ορίζεται από τη συνάρτηση map.setCenter() και στον οποίο προστίθενται τα εργαλεία για το zoom και τον τύπο του χάρτη με τη συνάρτηση του addControl(). Επίσης με τη συνάρτηση addListener προστίθεται μια παρακολούθηση του συμβάντος click στο χάρτη η οποία όταν γίνει κλικ στο χάρτη μεταφέρει τις συντεταγμένες στο πεδίο των συντεταγμένων στη φόρμα ( photoXY ). Όταν ο διαχειριστής πατήσει το κουμπί Εισαγωγή τα στοιχεία αποστέλλονται στη σελίδα addplace.php και προστίθενται στον πίνακα tblPlace.

```
$description = $_POST['description'];
$category=$_POST['hypercat'];
$title = $_POST['title'];
$address=$_POST['address'];
$mapXY= trim($_POST['photoXY']);

include 'dbinfo.php';

mysql_query("insert into tblplace(description,title,address,mapXY,category)
values('".$description."','".$title."','".$address."','".$mapXY."','".$category."");

mysql_close($link);
```

## Τροποποίηση σταθμού

Κατά την τροποποίηση ενός σταθμού ο διαχειριστής μέσα από τη σελίδα `updateplace.php` επιλέγει το σταθμό που πρόκειται να τροποποιήσει. Στη συνέχεια το `id` του σταθμού που επιλέχθηκε αποστέλλεται στη σελίδα `alterplace.php` όπου εκεί εμφανίζονται μέσα σε μια φόρμα τα στοιχεία του σταθμού που πρέπει να αλλαχθούν. Τα στοιχεία αποστέλλονται στη σελίδα `changeplace.php` όπου εκτελείται ο κατάλληλος κώδικας.

```
$description = $_POST['description'];
$category=$_POST['hypercat'];
$title = $_POST['title'];
$id=$_POST['id'];
include 'dbinfo.php';
$address=$_POST['address'];
$mapXY=$_POST['photoXY'];
```

```
mysql_query("update tblplace set
description='".$description."',address='".$address."',title='".$title."',mapXY='".$mapXY."',category='".$category.'" where id='".$id.'");
mysql_close($link);
```

## Εφαρμογή προβολής σταθμών

Η προβολή των σταθμών και των μετρήσεων υλοποιείται από τις `php` σελίδες `index.php` και `data.php`. Στη σελίδα `Index.php` υπάρχει ένας χάρτης Google ο οποίος εμφανίζει του σταθμούς αποστολής δεδομένων. Η εμφάνιση των σταθμών γίνεται με την εκτέλεση του παρακάτω κώδικα `javascript`:

```
var allmarkers = [];
function initialize() {
    if (GBrowserIsCompatible()) {
        var map = new GMap2(document.getElementById("map_canvas"));
        map.setCenter(new GLatLng<?php echo $mapXY.' '.$cityzoom;?>);
        map.addControl(new GLargeMapControl());
        map.addControl(new GOverviewMapControl());
        map.addControl(new GMapTypeControl());
        map.setMapType(G_NORMAL_MAP);
        // Create a base icon for all of our markers that specifies the
        // shadow, icon dimensions, etc.
        var baseIcon = new GIcon(G_DEFAULT_ICON);
        baseIcon.shadow = "http://www.google.com/mapfiles/shadow50.png";
        baseIcon.iconSize = new GSize(50, 50);
        baseIcon.shadowSize = new GSize(37, 34);
        baseIcon.iconAnchor = new GPoint(9, 34);
        baseIcon.infoWindowAnchor = new GPoint(9, 2);

        // Creates a marker whose info window displays the letter corresponding
        // to the given index.
```

```

function createMarker(point,placename,description,address)
{
    var letteredIcon = new GIcon(baseIcon);
        letteredIcon.image = "images/folder.png";
    // Set up our GMarkerOptions object
    markerOptions = { icon:letteredIcon };
    var marker = new GMarker(point, markerOptions);
    GEvent.addListener(marker, "click", function() {
        marker.openInfoWindowHtml("<HTML><BODY><p><a target='_blank'
href='data.php?station="+placename+">Station:"+placename+"</a></p><p>+description+"</p><p>Διεύ
θυσση:"+address+"</p></BODY></HTML>");

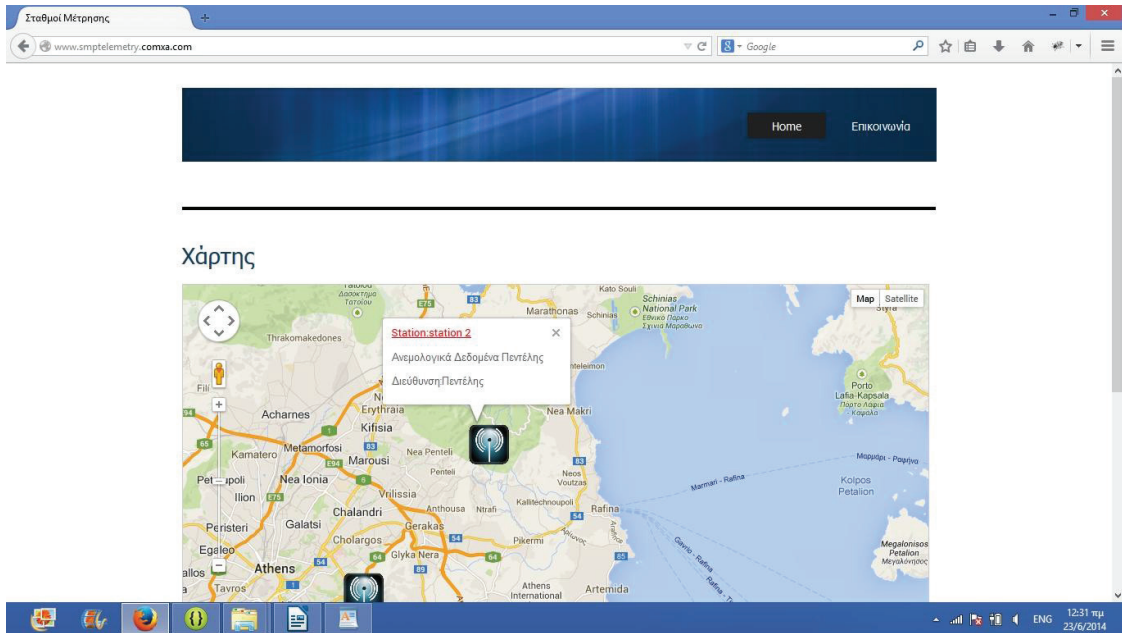
    });
    return marker;
}

// Add markers to the map at random locations
    i=0;
    <?php $places=mysql_query("select * from tblplace ");
        $j=0;
        while($j < mysql_num_rows($places))
        {?>
var latlng = new GLatLng<?php echo mysql_result($places,$j,"mapXY");?>;
    var placename = '<?php echo mysql_result($places,$j,"title");?>';
    var description = '<?php echo mysql_result($places,$j,"description");?>';
    var address = '<?php echo mysql_result($places,$j,"address");?>';
    allmarkers[i]=createMarker(latlng,placename,description,address );
    map.addOverlay(allmarkers[i]);
        allmarkers[i].show();
    i++;
    <?php $j++;}?>
        }
}

</SCRIPT>

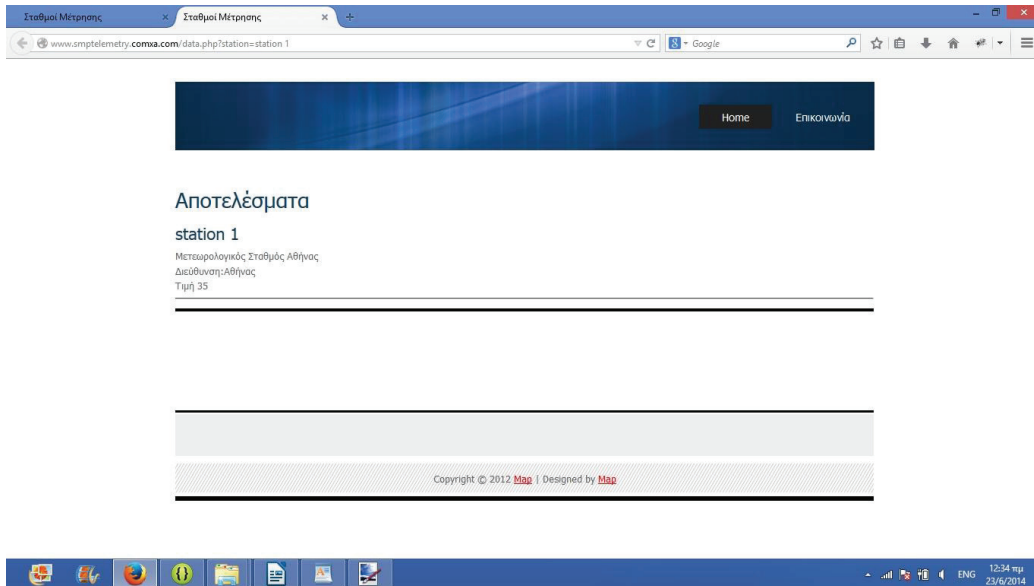
```

Αρχικά ορίζεται ένα διάνυσμα με όνομα allmarkers για τα σημεία που εμφανίζονται στο χάρτη. Η συνάρτηση initialize() αρχικοποιεί τον χάρτη θέτοντας το κέντρο, το zoom και τα controls για το zoom και τον τύπο του χάρτη. Επίσης ορίζει ένα base\_icon τύπου GIcon. Στη συνέχεια εκτελώντας ένα select query με τη χρήση κώδικα php προς τον πίνακα των σταθμών tblplace, δημιουργείται για κάθε έναν σταθμό ένα marker με την κλήση της συνάρτησης createMarker(). Για κάθε ένα marker που δημιουργείται προστίθενται και μια παρακολούθηση συμβάντων με την addListener(). Όταν γίνεται κλικ σε κάποιο marker τότε ανοίγει ένα μικρό παράθυρο με τα στοιχεία του σταθμού και ένα link προς τη σελίδα data.php.



Εικόνα 10

Η σελίδα data.php δέχεται το κωδικό όνομα της συσκευής αποστολής δεδομένων και στη συνέχεια εκτελεί ένα select ερώτημα για τις πρόσφατες θερμοκρασίες του συγκεκριμένου σταθμού.



Εικόνα 11



## Έλεγχος Συστήματος

### Εργαλεία Ανάπτυξης Λογισμικού

Όπως είδαμε το σύστημα υλοποιήθηκε με την κατασκευή τριών μερών:

- Την client εφαρμογή που θα εγκατασταθεί στις συσκευές κινητής τηλεφωνίας.
- Την server εφαρμογή που είναι εγκατεστημένη σε έναν server του Διαδικτύου. Τη Βάση Δεδομένων του συστήματος.

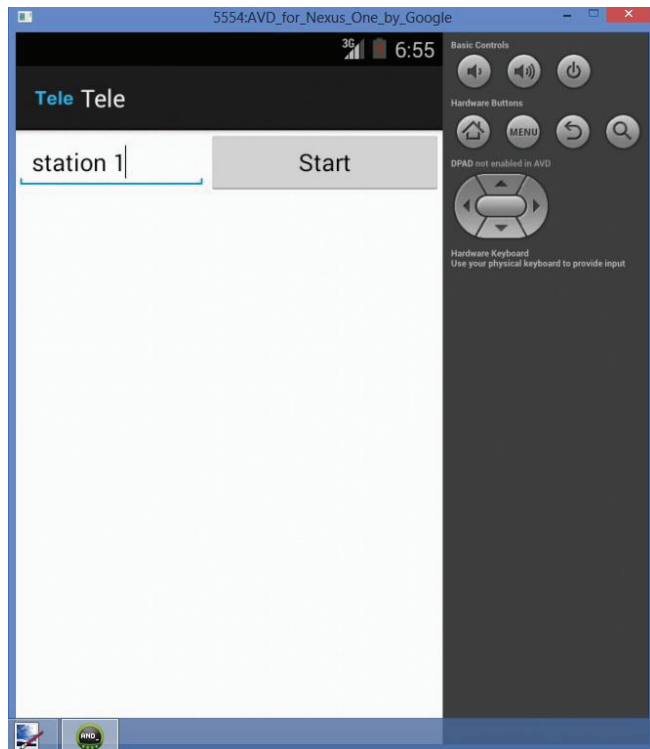
Η client εφαρμογή για κινητά τηλέφωνα αναπτύχθηκε για Android με τη χρήση του Android SDK.

Όσον αφορά την εφαρμογή στον server, αυτή αναπτύχθηκε χρησιμοποιώντας PHP ενώ η Βάση Δεδομένων αναπτύχθηκε σε MySQL 5. Για την κατασκευή της χρησιμοποιήθηκε το web-based εργαλείο phpMyAdmin.

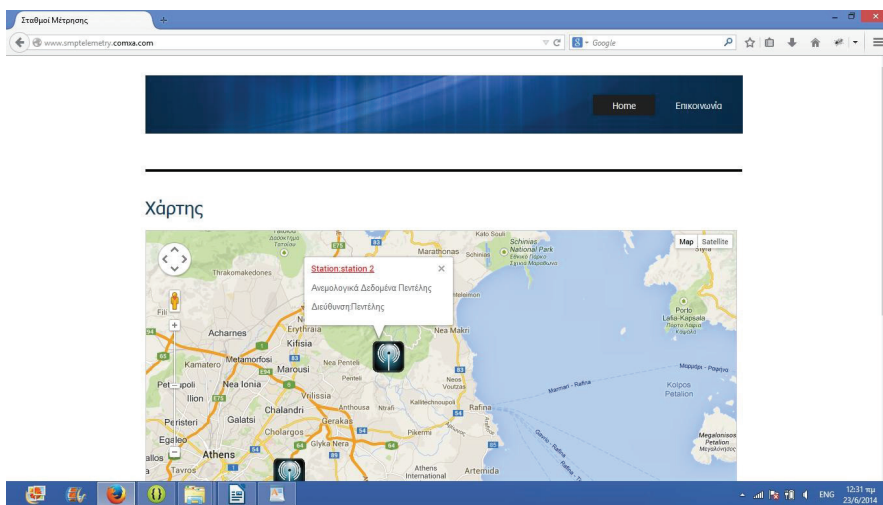
### Φάσεις Υλοποίησης και Ελέγχου

Αρχικά δημιουργήθηκε η ΒΔ σύμφωνα με τις απαιτήσεις που είχαν προκύψει από τη φάση της Ανάλυσης και του Σχεδιασμού. Στη συνέχεια δημιουργήθηκαν οι πίνακες tblPlace, tblCategory και tblData χρησιμοποιώντας το εργαλείο phpMyAdmin.

Στη συνέχεια δημιουργήθηκε η client εφαρμογή με τη χρήση του Android SDK. Η εφαρμογή παράγει δεδομένα με τη χρήση μιας γεννήτρια τυχαίων αριθμών και τα αποστέλλει στη server εφαρμογή.



Εικόνα 12



Εικόνα 13

## Βιβλιογραφία

1. Pei Zheng, Lionel Ni, “Smart Phone and Next Generation Mobile Computing”, Morgan Kaufmann, 2006
2. Mo Mohitpour, Jason Szabo, Thomas Van Hardeveld, *Pipeline Operation & Maintenance*, ASME Press New York 2003 (p519)
3. Warnock, *Programmable controllers*. Prentice Hall, 1988
4. Elliotte Rusty Harold, “Java Network Programming”, O'Reilly, 2004
5. *Telemetry Technologies to Deploy Aeris.net's MicroBurst Technology; Wireless Data Messaging Service Enables Data Monitoring Applications*, Business Wire, May 19, 2000
6. Edelstein, Herb. *Unraveling Client/Server Architecture*. *DBMS* 7, 5 May 1994 (p34).
7. Honghua Dai, Ramakrishnan Srikant, PAKDD, Chengqi Zhang, “Advances in knowledge discovery and data mining”, Springer, 2004
8. Parkinson 1996
9. Usama Fayyad and Evangelos Simoudis, *Knowledge Discovery and Data Mining tutorial*, 1995.