



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ, ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ & ΔΙΚΤΥΩΝ

*Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για
Ασύρματα Δίκτυα Αισθητήρων*

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Νικολάου Θεόφилου

Αν. Καθ. Σταμούλης Γεώργιος
Επιβλέπων καθηγητής

Καθ. Χούστη Αικατερίνη
Μέλος Επιτροπής Εξέτασης

Καθ. Τασιούλας Λεάνδρος
Μέλος Επιτροπής Εξέτασης

Βόλος, Νοέμβριος 2006



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΒΙΒΛΙΟΘΗΚΗ & ΚΕΝΤΡΟ ΠΛΗΡΟΦΟΡΗΣΗΣ
ΕΙΔΙΚΗ ΣΥΛΛΟΓΗ «ΓΚΡΙΖΑ ΒΙΒΛΙΟΓΡΑΦΙΑ»**

Αριθ. Εισ.: 5079/1

Ημερ. Εισ.: 19-09-2007

Δωρεά: Συγγραφέα

Ταξιθετικός Κωδικός: Δ

004.6

ΝΙΚ

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

(this page is left intentionally blank)

ΠΕΡΙΛΗΨΗ

Σκοπός της εργασίας είναι η ανάπτυξη μιας διαδικτυακής διεπαφής (web based interface) για την απεικόνιση των δεδομένων που συλλέγονται από ένα δίκτυο ασύρματων αισθητήρων. Η απεικόνιση των δεδομένων γίνεται είτε με την μορφή γραφημάτων είτε με την μορφή πινάκων.

Η λειτουργία του συστήματος που υλοποιήθηκε συνοψίζεται στα εξής:

Τα δεδομένα που συλλέγονται από τους κόμβους αισθητήρες αποθηκεύονται σε μια βάση δεδομένων. Ο χρήστης, μέσω της γραφικής διεπαφής, υποβάλλει το αίτημά του για εμφάνιση δεδομένων που συμφωνούν με ορισμένα κριτήρια που έχει επιλέξει (δεδομένα συγκεκριμένου κόμβου, συγκεκριμένης χρονικής περιόδου κ.α.) και ο εξυπηρετητής συλλέγει τις απαραίτητες πληροφορίες από τη βάση δεδομένων που αφορούν αυτό το αίτημα του χρήστη. Τέλος, τα δεδομένα – αποτελέσματα εμφανίζονται στο χρήστη με μορφή γραφημάτων ή πινάκων.

Λέξεις Κλειδιά:

Ασύρματα δίκτυα αισθητήρων, σύστημα καταγραφής δεδομένων, διαδικτυακή διεπαφή, γραφήματα, γράφημα πραγματικού χρόνου

Abstract

The aim of the project is the development of a web based interface to be used for the manipulation and presentation of data gathered from a wireless sensors' network. Data are displayed using either graphs or tables.

Functionality of the implemented system can be summarized as follows:

Data that are gathered from the sensors – nodes are stored in a database. The user submits a request through the web based graphical user interface for data that meet certain criteria (such as to originate from a specific node, during a specific time period etc) to be displayed. The web server, then, collects the information specified by the user request from the database server. Finally, data are displayed in a graphical form either as graphs or live graphs and tables.

Keywords

Wireless sensor networks, WSN, data monitoring application, web interface, graphs, live graph

ΕΥΧΑΡΙΣΤΙΕΣ

Ευχαριστώ τον επιβλέποντα καθηγητή μου Σταμούλη Γεώργιο και τον καθηγητή Παναγιώτη Κίκιρα για την βοήθεια και καθοδήγηση που μου προσέφεραν καθ' όλη τη διάρκεια εκπόνησης της παρούσας εργασίας καθώς και τους: Φώτη Λούκο, Ελένη Παπαϊωάννου και Νίκο Τζιρίτα που όντας μέλη της, ίδιας με εμένα, ομάδας υλοποιήσαμε ένα σύστημα στα πλαίσια ενός μαθήματος, στο οποίο βασίστηκε αυτό της παρούσας εργασίας.

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

(this page is left intentionally blank)

Περιεχόμενα

ΠΕΡΙΕΧΟΜΕΝΑ	7
ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ	8
1 ΕΙΣΑΓΩΓΗ	10
1.1 ΑΝΤΙΚΕΙΜΕΝΟ ΕΡΓΑΣΙΑΣ.....	10
1.2 ΔΙΑΡΘΡΩΣΗ ΕΡΓΑΣΙΑΣ.....	10
2 ΑΣΥΡΜΑΤΑ ΔΙΚΤΥΑ ΑΙΣΘΗΤΗΡΩΝ	11
2.1 ΓΕΝΙΚΑ.....	11
2.2 ΕΦΑΡΜΟΓΕΣ.....	13
3. Η ΕΦΑΡΜΟΓΗ	15
3.1 Το ΜΟΝΤΕΛΟ ΑΝΑΠΤΥΞΗΣ.....	15
3.2 ΑΝΑΛΥΣΗ.....	16
3.2.1 Ανάλυση με χρήση διαγραμμάτων UML.....	18
3.3 ΣΧΕΔΙΑΣΗ.....	31
3.3.1 Αρχιτεκτονική.....	31
3.4 ΥΛΟΠΟΙΗΣΗ.....	34
3.4.1 Επιμέρους τμήματα.....	34
3.4.2 Η δομή της βάσης δεδομένων.....	35
3.4.3 Μετασχηματισμοί.....	36
3.5 ΈΛΕΓΧΟΣ ΕΦΑΡΜΟΓΗΣ – ΔΟΚΙΜΑΣΤΙΚΗ ΛΕΙΤΟΥΡΓΙΑ.....	39
4 ΠΑΡΟΥΣΙΑΣΗ ΣΥΣΤΗΜΑΤΟΣ	41
4.1 ΚΑΡΤΕΛΑ ΠΡΩΤΗ: ΕΝΑΣ ΚΟΜΒΟΣ – ΕΝΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΟ.....	41
4.2 ΚΑΡΤΕΛΑ ΔΕΥΤΕΡΗ: ΕΝΑΣ ΚΟΜΒΟΣ – ΠΟΛΛΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ.....	42
4.3 ΚΑΡΤΕΛΑ ΤΡΙΤΗ: ΠΟΛΛΟΙ ΚΟΜΒΟΙ – ΕΝΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΟ.....	43
4.4 ΚΑΡΤΕΛΑ ΤΕΤΑΡΤΗ: ΓΡΑΦΗΜΑ ΠΡΑΓΜΑΤΙΚΟΥ ΧΡΟΝΟΥ.....	45
4.5 ΚΑΡΤΕΛΑ ΠΕΜΠΤΗ: ΕΡΩΤΗΣΕΙΣ ΣΤΗΝ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ.....	45
5 ΠΡΟΒΛΗΜΑΤΑ, ΠΡΟΟΠΤΙΚΕΣ, ΣΥΜΠΕΡΑΣΜΑΤΑ	47
5.1 ΠΡΟΒΛΗΜΑΤΑ.....	47
5.2 ΠΡΟΟΠΤΙΚΕΣ – ΔΥΝΑΤΟΤΗΤΕΣ ΕΠΕΚΤΑΣΗΣ.....	48
5.3 Η ΑΥΤΟΝΟΜΗ ΕΦΑΡΜΟΓΗ.....	49
5.4 ΕΠΙΛΟΓΟΣ.....	49
ΑΝΑΦΟΡΕΣ	50
ΠΑΡΑΡΤΗΜΑ	53

Πίνακας εικόνων

Πίνακας 1 : Συνοπτική παράθεση χαρακτηριστικών των πιο.....	15
γνωστών μοντέλων κύκλου ζωής λογισμικού	15
Εικόνα 1 : Το γενικό μοντέλο κύκλου ζωής λογισμικού που χρησιμοποιήθηκε	16
Εικόνα 2 : Το διάγραμμα ροής δεδομένων (ΔΡΔ) του συστήματος	17
Εικόνα 3 : 1 ^η περίπτωση χρήσης - εμφάνιση γραφήματος στην πρώτη καρτέλα	20
Εικόνα 4 : Το διάγραμμα δραστηριοτήτων για την πρώτη καρτέλα.....	20
Εικόνα 5 : 2 ^η περίπτωση χρήσης – εμφάνιση γραφημάτων	22
στην δεύτερη καρτέλα	22
Εικόνα 6 : Το διάγραμμα δραστηριοτήτων για την δεύτερη καρτέλα	22
Εικόνα 7 : 3 ^η περίπτωση χρήσης – εμφάνιση γραφήματος στην τρίτη καρτέλα	24
Εικόνα 8 : Το διάγραμμα δραστηριοτήτων για την τρίτη καρτέλα.....	24
Εικόνα 9 : 4 ^η περίπτωση χρήσης – εμφάνιση γραφήματος.....	26
πραγματικού χρόνου στην τέταρτη καρτέλα	26
Εικόνα 10 : Το διάγραμμα δραστηριοτήτων για την τέταρτη καρτέλα	26
Εικόνα 11 : 5 ^η περίπτωση χρήσης – εμφάνιση πίνακα τιμών στην	28
πέμπτη καρτέλα.....	28
Εικόνα 12 : Το διάγραμμα δραστηριοτήτων για την πέμπτη καρτέλα	28
Εικόνα 13 : Γενική περίπτωση χρήσης του συστήματος	30
Εικόνα 14 : Διάγραμμα δραστηριοτήτων για την περίπτωση.....	30
χρήσης του συστήματος.....	30
Εικόνα 15 : Η αρχιτεκτονική τριών επιπέδων.....	32
Εικόνα 16 : Η αρχιτεκτονική του συστήματος που υλοποιήθηκε.....	33
Πίνακας 2 : Ο πίνακας q1	35
Πίνακας 3 : Ο πίνακας q2	35
Πίνακας 4 : Τα αρχεία της εφαρμογής.....	39
Εικόνα 17 : Η πρώτη καρτέλα.....	41
Εικόνα 18 : Δείγμα γραφήματος που εμφανίζεται στην πρώτη καρτέλα.....	42
Εικόνα 19 : Η δεύτερη καρτέλα	42
Εικόνα 20 : Έχοντας επιλέξει πολλά χαρακτηριστικά	43
Εικόνα 21 : Η τρίτη καρτέλα.....	44
Εικόνα 22 : Δείγμα γραφήματος της τρίτης καρτέλας.....	44
Εικόνα 23 : Το γράφημα πραγματικού χρόνου της τέταρτης καρτέλας	45

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

Εικόνα 24 : Η καρτέλα επρωτήσεων προς την βάση	46
Εικόνα 25 : Δείγμα του πίνακα της τελευταίας καρτέλας με δεδομένα.....	46

1 Εισαγωγή

1.1 Αντικείμενο Εργασίας

Τα Ασύρματα Δίκτυα Αισθητήρων (Wireless Sensor Networks, WSN) αποτελούνται από μερικούς έως πολλούς, μικρού μεγέθους κόμβους με περιορισμένη ενεργειακή αυτονομία και υπολογιστικές δυνατότητες, κατανομημένους σε μια ορισμένη γεωγραφική περιοχή. Τέτοια δίκτυα χρησιμοποιούνται σε πληθώρα εφαρμογών, όπως ενδεικτικά είναι η παρακολούθηση του περιβάλλοντος, η γεωργία ακριβείας και διάφορες στρατιωτικές εφαρμογές. Συνήθως, μετά την τοποθέτησή τους τα ασύρματα δίκτυα απλά συλλέγουν δεδομένα λειτουργώντας αυτόνομα χωρίς ανθρώπινη παρέμβαση, καθ' όλη τη διάρκεια της “ζωής” τους (η διάρκεια της ενεργειακής πηγής των κόμβων). Έτσι, ο κάθε αισθητήρας κόμβος συλλέγει δεδομένα από το περιβάλλον και τα στέλνει μέσω των γειτονικών του κόμβων (multi-hop routing) σε ένα κόμβο πύλη (sink ή gateway) ο οποίος έχει απεριόριστο αποθηκευτικό χώρο και ενέργεια.

Σε ένα δίκτυο αισθητήρων είναι προφανής η ανάγκη για παρακολούθηση και επεξεργασία των δεδομένων που στέλνονται από τους αισθητήρες είτε σε πραγματικό χρόνο, είτε αν αυτά έχουν πρώτα αποθηκευτεί σε μια βάση δεδομένων. Με άλλα λόγια, για να υφίσταται χρησιμότητα του δικτύου αισθητήρων πρέπει, με κάποιο τρόπο, τα δεδομένα που συλλέγονται, να εμφανίζονται και να μπορούν να επεξεργαστούν από τον εκάστοτε χρήστη για να εξαχθούν χρήσιμα συμπεράσματα. Σκοπός της παρούσας εργασίας ήταν η υλοποίηση ενός συστήματος για αυτόν το σκοπό, με κύριο στόχο την χρησιμότητα και την ευχρηστία από τον τελικό χρήστη.

1.2 Διάρθρωση Εργασίας

Η εργασία ξεκινά με μια εισαγωγή του αναγνώστη στα ασύρματα δίκτυα αισθητήρων (wireless sensor networks) στο κεφάλαιο 2. Στο κεφάλαιο 3 αναλύεται η μέθοδος ανάπτυξης της εφαρμογής (ανάλυση απαιτήσεων, αρχιτεκτονική κλπ.). Η εφαρμογή που υλοποιήθηκε παρουσιάζεται στο κεφάλαιο 4. Το κεφάλαιο 5 αναφέρεται στα συμπεράσματα και στις μελλοντικές προοπτικές βελτίωσης που έχει η διαδυσκτακή εφαρμογή και η αυτόνομη εφαρμογή που ξεκίνησε να υλοποιείται. Τέλος, στο παράρτημα υπάρχει ο κώδικας της εφαρμογής.

2 Ασύρματα Δίκτυα Αισθητήρων

2.1 Γενικά

Η ανάπτυξη των δικτύων αισθητήρων απαιτεί τεχνολογία από τρεις διαφορετικούς ερευνητικούς τομείς [1]: αυτούς των αισθητήρων, των επικοινωνιών και της επεξεργασίας (συμπεριλαμβανομένου του υλικού, του λογισμικού και των διαφόρων αλγορίθμων). Κατά συνέπεια, η πρόοδος είτε συνδυαστικά είτε χωριστά σε κάθε μια από αυτές τις τεχνολογικές περιοχές, έχει οδηγήσει την έρευνα στα δίκτυα αισθητήρων.

Τα τελευταία χρόνια, οι εξελίξεις στην τεχνολογία των ασύρματων επικοινωνιών και της ηλεκτρονικής κατέστησαν δυνατή την ανάπτυξη χαμηλού κόστους, χαμηλής κατανάλωσης, πολλαπλών χρήσεων αισθητήρων που είναι μικροσκοπικοί σε μέγεθος, λειτουργούν χωρίς ανθρώπινη παρέμβαση και επικοινωνούν μεταξύ τους σε μικρές αποστάσεις. Ένα ασύρματο δίκτυο αισθητήρων είναι ένα δίκτυο που αποτελείται από έναν μεγάλο αριθμό από τέτοιους κόμβους αισθητήρες καθένας από τους οποίους ενσωματώνει εξοπλισμό συγκεκριμένης επεξεργαστικής, επικοινωνιακής και αποθηκευτικής ισχύος και επικοινωνούν ασύρματα μεταξύ τους [2]. Η θέση των κόμβων δεν χρειάζεται να είναι προσχεδιασμένη. Αυτό από την μια επιτρέπει τυχαία τοποθέτηση σε μια, για παράδειγμα, δυσπρόσιτη περιοχή, από την άλλη όμως, σημαίνει μεγαλύτερη δυσκολία στην υλοποίηση αποδοτικών πρωτοκόλλων και αλγορίθμων.

Στις μέρες μας, ένα τυπικό ασύρματο δίκτυο αισθητήρων χαρακτηρίζεται από τα εξής:

- Μεγάλο πλήθος ασύρματων κόμβων.
- Σχετικά μεγάλη πυκνότητα στην τοποθέτησή των κόμβων.
- Ανοχή σε πιθανές αποτυχίες λειτουργίας ορισμένων κόμβων.
- Συχνή αναπροσαρμογή της τοπολογίας του δικτύου.
- Περιορισμένη ενέργεια στους κόμβους.

Οι κόμβοι συνήθως είναι διάσπαρτοι σε μια συγκεκριμένη περιοχή με απόσταση δεκάδων ποδιών ο ένας από τον άλλων [3]. Μπορούν να τοποθετηθούν στο έδαφος, τον αέρα, κάτω από το νερό, μέσα σε ζωντανούς οργανισμούς, σε οχήματα και στο

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

εσωτερικό κτιρίων ή κατασκευών. Κάθε ένας από αυτούς του κόμβους έχει την δυνατότητα να συγκεντρώνει δεδομένα και να τα δρομολογεί πίσω σε έναν κεντρικό κόμβο (sink, gateway) χρησιμοποιώντας πολλαπλών αλμάτων μη-δομημένης (multi-hop ad-hoc or infrastructureless) αρχιτεκτονικής [2] [4] [5] [6]. Ο κεντρικός αυτός κόμβος μπορεί να συνδέεται απευθείας με κάποιο υπολογιστικό σύστημα ή να στέλνει τα δεδομένα που έχει συγκεντρώσει μέσω δορυφόρου ή κάποιου άλλου τρόπου (υπάρχουσα τηλεφωνική υποδομή, δίκτυο οπτικών ινών κλπ) σε κάποιο απομακρυσμένο υπολογιστικό σύστημα. Η διάρκεια ζωής ενός κόμβου εξαρτάται ουσιαστικά μόνο από την διάρκεια ζωής της ενεργειακής του πηγής (συνήθως μπαταρία) καθώς τις περισσότερες φορές είναι αυτή που εξαντλείται πολύ πριν παρουσιαστεί κάποια βλάβη λειτουργίας στα κυκλώματα του κόμβου. Για το θέμα αυτό υπάρχουν δύο προσεγγίσεις σήμερα. Η μια αφορά τον εξοπλισμό των αισθητήρων με μια επαναφορτιζόμενη ενεργειακή πηγή και η κυρίαρχη επιλογή είναι η χρήση μπαταριών υψηλής πυκνότητας [7] [8]. Η άλλη εναλλακτική λύση για αυτή την προσέγγιση είναι η χρησιμοποίηση των κυψελών καυσίμου (fuel cells) οι οποίες αποτελούν καθαρές ενεργειακές πηγές ιδιαίτερα υψηλής πυκνότητας [9]. Εντούτοις, ακόμη δεν υπάρχουν σε μορφή κατάλληλη για πρακτική χρήση στους κόμβους αισθητήρες. Η δεύτερη προσέγγιση αφορά την συλλογή ενέργειας που είναι διαθέσιμη στο περιβάλλον [4]. Παράλληλα με την χρήση ηλιακών κυψελών και φωτοβολταϊκών που ήδη χρησιμοποιούνται σε φορητές συσκευές, όπως, ηλεκτρονικοί υπολογιστές τσέπης (calculators) υπάρχει ένας αριθμός προτάσεων για την μετατροπή δονήσεων σε ηλεκτρική ενέργεια [10].

Τα δεδομένα που μπορεί να συγκεντρώνει ένας κόμβος αισθητήρας και κατά συνέπεια ένα ασύρματο δίκτυο αισθητήρων περιορίζονται μόνο από τα κυκλώματα αισθητήρων (sensor boards) που αυτοί περιέχουν και περιλαμβάνουν μεταξύ άλλων μετρήσεις για την υγρασία, το φως, τον ήχο, την μαγνήτιση, την επιτάχυνση, την θερμοκρασία, την τάση της ενεργειακής τους πηγής ή διαφόρων ειδών ακτινοβολίες [11]. Η λίστα με τους τύπους δεδομένων που μπορεί να συγκεντρώσει ένας αισθητήρας θα συνεχίσει να αυξάνει καθώς τεχνολογικές εξελίξεις επιτρέπουν την μέτρηση ακόμα και τοξικών ουσιών στο χώμα, το νερό ή τον αέρα, ίσως ακόμα και γεύσεων.

2.2 Εφαρμογές

Η έρευνα για τα δίκτυα αισθητήρων ήταν, όπως και για πολλές άλλες τεχνολογίες, αρχικά παρακινούμενη από στρατιωτικές εφαρμογές [1] [12] [13] [14]. Εντούτοις, η διαθεσιμότητα χαμηλού κόστους αισθητήρων και επικοινωνιακών δικτύων έχουν οδηγήσει στην ανάπτυξη πολλών άλλων πιθανών εφαρμογών.

Έτσι, σήμερα, τα δίκτυα αισθητήρων μπορούν να βρουν εφαρμογή σε ένα μεγάλο πλήθος δραστηριοτήτων.

Για παράδειγμα, μπορούν να χρησιμοποιηθούν για την παρατήρηση του περιβάλλοντός μας [15], φαινομένων και ζωντανών οργανισμών σε αυτό καθώς και την αλληλεπίδραση αυτών μεταξύ τους και με το περιβάλλον. Έτσι, μπορεί να πραγματοποιηθεί παρατήρηση της σεισμικής δραστηριότητας κάποιου ωκεανού [16], των καιρικών και άλλων φυσικών φαινομένων και να συγκεντρωθούν νέα στοιχεία, σημαντικά για την περιβαλλοντική επιστήμη, όπως, κλιματολογικά μοντέλα και αρκετά δεδομένα για την προειδοποίηση, αντιμετώπιση ή αντίδραση σε μια φυσική καταστροφή (π.χ. επικείμενη πλημμύρα ή εκδήλωση ακραίων καιρικών φαινομένων). Για τους ίδιους λόγους τέτοια ασύρματα δίκτυα αισθητήρων χρησιμοποιούνται και στην γεωργία ακριβείας [17]. Παράλληλα, η δυνατότητα των κόμβων να τοποθετούνται στην περιοχή ενδιαφέροντος ακόμα και με απομακρυσμένη ρίψη, τους καθιστά ιδανικούς για την έρευνα και εξερεύνηση απομακρυσμένων, δυσπρόσιτων ή επικίνδυνων τοποθεσιών, όπως για παράδειγμα το εσωτερικό ενός ηφαιστείου ή μια σπηλιά μεγάλου βάθους, όπου δεν έχουν γίνει στοιχειώδεις μελέτες για αυτούς ακριβώς τους λόγους.

Επίσης, τα ασύρματα δίκτυα αισθητήρων μπορούν να έχουν στρατιωτική χρήση [18] [19]. Ενδεικτικές εφαρμογές είναι μεταξύ άλλων: η παρακολούθηση μιας περιοχής για εχθρικές κινήσεις, η πληροφόρηση για το πεδίο της μάχης και η κατάδειξη ενός πιθανού στόχου [20]. Ένα ασύρματο δίκτυο αισθητήρων μπορεί, ακόμα, να ανιχνεύσει πιθανές απειλές, για παράδειγμα την παρουσία εχθρικού υποβρυχίου ή χημικές, βιολογικές ή πυρηνικές επιθέσεις [21].

Παράλληλα, τέτοια δίκτυα μπορούν να χρησιμοποιηθούν σε εφαρμογές υγείας με την παρακολούθηση ασθενών και την βοήθεια ατόμων με κινητικά προβλήματα και σε οικιακές εφαρμογές, όπως η παρακολούθηση δωματίων για πιθανούς κινδύνους (διαρροές αερίου, φωτιά κλπ.) και χρήση τους ως αντικλεπτικό σύστημα, ενώ έχει

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

γίνει μελέτη για την δημιουργία ενός “έξυπνου” παιδικού σταθμού που χρησιμοποιεί δίκτυα αισθητήρων για εκπαιδευτικούς σκοπούς [22].

Τέλος, χρήση τέτοιων δικτύων γίνεται και στην βιομηχανία [23] παρέχοντας βοήθεια, ενδεικτικά, στην κατασκευαστική αυτοματοποίηση, στην παρακολούθηση της χωρητικότητας μιας αποθήκης, στον ποιοτικό έλεγχο προϊόντων, στην αναγκαιότητα συντήρησης και αλλαγής εξοπλισμού ή στον έλεγχο και παρατήρηση οικοδομημάτων και κατασκευών.

3. Η εφαρμογή

3.1 Το μοντέλο ανάπτυξης

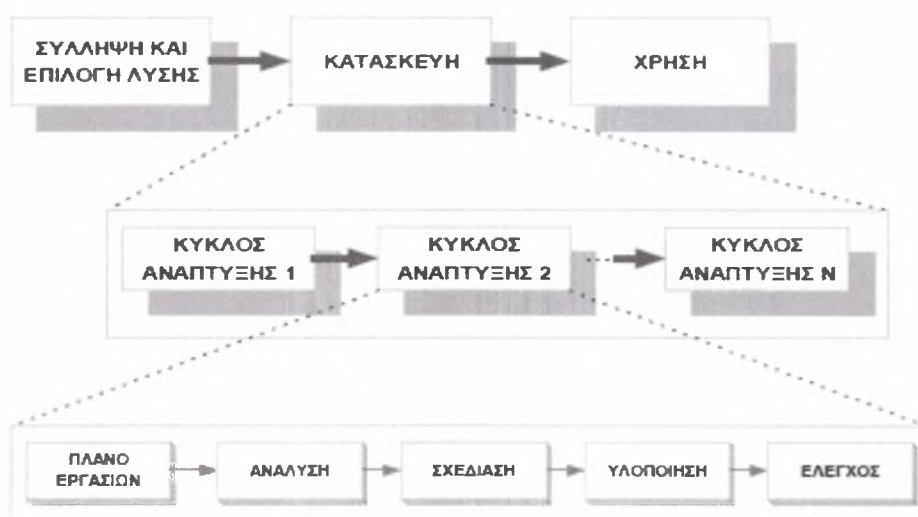
Τα σύγχρονα μοντέλα ζωής λογισμικού προσπαθούν να δώσουν μια γενική κατεύθυνση εφαρμογής των υπάρχουσών ιδεών, αφήνοντας σημαντικούς βαθμούς ελευθερίας στον προγραμματιστή που τα ακολουθεί. Η περιγραφή τέτοιων μοντέλων περιέχει μόνο τις γενικές κατευθύνσεις για αυτό χαρακτηρίζονται ως γενικά και δεν είναι άρρηκτα συνδεδεμένη με κάποια μεθοδολογία ανάπτυξης λογισμικού [24].

Το μοντέλο ανάπτυξης λογισμικού που υιοθετήθηκε για την εφαρμογή κινείται στα ίδια πλαίσια και μπορεί να χαρακτηριστεί ως απόγονος πολλών από τα παλαιότερα μοντέλα (μοντέλο καταρράκτη, σπειροειδές μοντέλο, μοντέλο προτυποποίησης κ.ά.) καθώς συνδυάζει αρκετά χαρακτηριστικά τους όπως επιτάσσει η σύγχρονη τάση στην ανάπτυξη λογισμικού από ότι φαίνεται και στον πίνακα (*Πίνακας 1*) [24].

Μοντέλο	Μέγεθος εφαρμογών	Μεταβολές στις απαιτήσεις	Προσαρμοστικότητα στον κατασκευαστή	Διάδοση
Καταρράκτη	Μικρό έως μεσαίο	Ανεπιθύμητες	Καμία	Μεγάλη με τάση μείωσης
Πρωτοτυποποίησης	Μικρό ως μεσαίο	Δεκτές	Μικρή	Μικρή με τάση αύξησης
Λειτουργικής επαύξησης	Μεσαίο ως μεγάλο	Ανεπιθύμητες	Καμία	Μικρή με τάση μείωσης
Σπειροειδές	Μεσαίο ως μεγάλο	Δεκτές	Αρκετή	Μικρή με τάση μείωσης
Πίδακα	Οποιοδήποτε	Δεκτές	Αρκετή	Μικρή
Γενικό	Οποιοδήποτε	Δεκτές	Μεγάλη	Μικρή με ισχυρές τάσεις αύξησης

Πίνακας 1 : Συνοπτική παράθεση χαρακτηριστικών των πιο γνωστών μοντέλων κύκλου ζωής λογισμικού

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων



Εικόνα 1 : Το γενικό μοντέλο κύκλου ζωής λογισμικού που χρησιμοποιήθηκε

Στο σχήμα (*Εικόνα 1*) [24] παρουσιάζεται το μοντέλο που χρησιμοποιήθηκε. Το γενικό πλαίσιο του μοντέλου αυτού περιλαμβάνει τις φάσεις σύλληψης, κατασκευής και λειτουργίας. Κάθε μια από αυτές αναλύεται σε επιμέρους εργασίες, σύμφωνα με τα χαρακτηριστικά του εκάστοτε περιβάλλοντος. Ιδιαίτερα η γενική φάση της κατασκευής αναλύεται σε “κύκλους ανάπτυξης” καθένας εκ των οποίων προσθέτει νέα χαρακτηριστικά και λειτουργίες στο υπό κατασκευή λογισμικό. Τα επιμέρους βήματα μέσα σε κάθε κύκλο ανάπτυξης μοιάζουν με τα βήματα του μοντέλου του καταρράκτη, μόνο που δεν εφαρμόζονται για ολόκληρο το σύστημα, αλλά για το μικρό μέρος του που κατασκευάζεται στον εν λόγω κύκλο, όπως στο μοντέλο της προτυποποίησης. Για την εκκίνηση κάθε κύκλου ανάπτυξης είναι πιθανόν να έχει προηγηθεί ανάλυση ρίσκου και σκοπιμότητας όπως στο σπειροειδές μοντέλο αλλά δεν είναι απαραίτητο. Ζητήματα πως αλληλουχία των ενεργειών, ακριβής καθορισμός των κύκλων ανάπτυξης κ.ά., αφήνονται στη διακριτική ευχέρεια του κάθε αναλυτή - προγραμματιστή από τον οποίο και καθορίζονται σύμφωνα με τις ιδιαιτερότητες κάθε περίπτωσης.

3.2 Ανάλυση

3.2.1 Ανάλυση απαιτήσεων

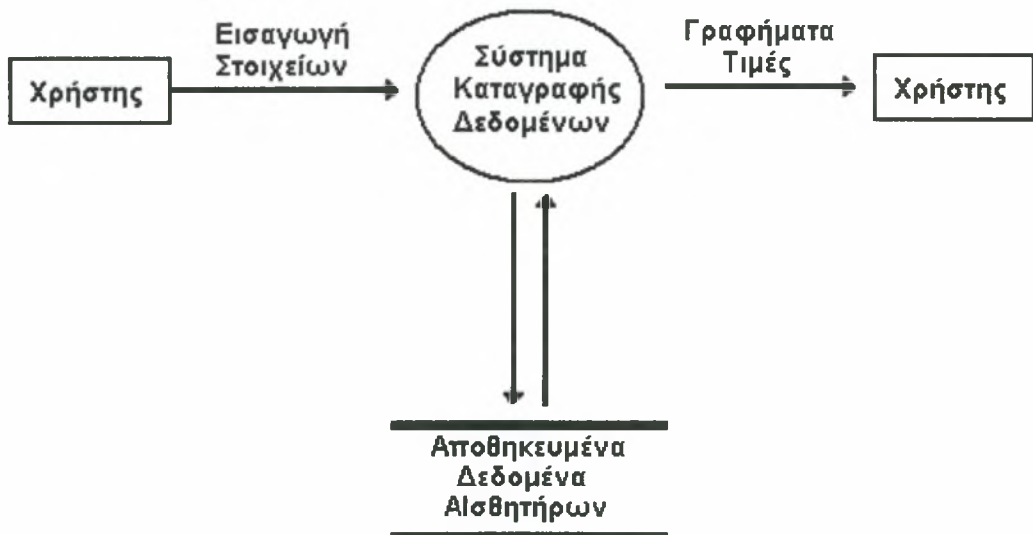
Το πρώτο στάδιο του μοντέλου ανάπτυξης λογισμικού που υιοθετήθηκε αφορά την ανάλυση των απαιτήσεων του συστήματος. Κατά την ανάλυση των απαιτήσεων εντοπίζονται για πρώτη φορά οι απαιτήσεις από το λογισμικό και ορίζονται

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

λειτουργίες και γεγονότα τα οποία πρέπει να εκτελεστούν έτσι ώστε εκπληρωθούν οι στόχοι του συστήματος.

Αποτέλεσμα των εργασιών που εκτελούνται στη φάση αυτή είναι ένα σύνολο απαιτήσεων από το λογισμικό οι οποίες περιγράφονται με μορφή διαγραμμάτων παρακάτω.

Το πρωταρχικό διάγραμμα ροής δεδομένων του συστήματος που προέκυψε από την παραπάνω αρχική φάση ανάλυσης παρουσιάζεται στην *Εικόνα 2*. Εκεί βλέπουμε πως ο χρήστης εισάγει στο σύστημα τα στοιχεία που επιθυμεί, το σύστημα επεξεργάζεται αυτά τα στοιχεία και ανακτά δεδομένα από τα αποθηκευμένα δεδομένα των αισθητήρων και επιστρέφει στον χρήστη γραφήματα και τιμές δεδομένων.



Εικόνα 2: Το διάγραμμα ροής δεδομένων (ΔΡΔ) του συστήματος

Σε αυτό το στάδιο παρουσιάστηκε η ανάγκη ύπαρξης διαφόρων τρόπων απεικόνισης των δεδομένων. Έτσι, υιοθετήθηκε η έννοια των καρτελών απεικόνισης από όπου ο χρήστης επιλέγει τον τρόπο απεικόνισης των δεδομένων.

Η βασική λειτουργία του συστήματος, όπως αναφέρθηκε παραπάνω, είναι η εμφάνιση δεδομένων, που έχουν σταλεί από τους αισθητήρες και έχουν αποθηκευτεί, με μορφή γραφημάτων και πινάκων τιμών. Οι δραστηριότητες που περιλαμβάνονται στη βασική αυτή λειτουργία είναι οι εξής:

- Ο χρήστης μεταβαίνει στην καρτέλα που αφορά τον τρόπο απεικόνισης δεδομένων που επιθυμεί.

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

- Στη συνέχεια επιλέγει τον κόμβο ή τους κόμβους, το χαρακτηριστικό ή τα χαρακτηριστικά, την χρονική περίοδο και κάποια άλλα στοιχεία ανάλογα με την καρτέλα στην οποία βρίσκεται και πατάει το αντίστοιχο κουμπί εμφάνισης.
- Αν τα στοιχεία που έχει εισάγει ο χρήστης είναι έγκυρα και υπάρχουν δεδομένα που συμφωνούν με τα κριτήρια που έχει θέσει, αυτά θα εμφανιστούν με την μορφή που αντιστοιχεί στην καρτέλα που βρίσκεται.
- Σε διαφορετική από την παραπάνω περίπτωση εμφανίζεται μήνυμα λάθους ή δεν εμφανίζονται δεδομένα.
- Σε κάθε περίπτωση ο χρήστης μπορεί να ξεκινήσει την διαδικασία από την αρχή ή να μεταβεί σε διαφορετική καρτέλα.

3.2.1 Ανάλυση με χρήση διαγραμμάτων UML

Το επόμενο στάδιο της ανάλυσης αφού έχουν προσδιοριστεί οι απαιτήσεις είναι η περιγραφή τους με διαγράμματα. Για τον σκοπό αυτό χρησιμοποιήθηκε η γλώσσα UML [25] και συγκεκριμένα οι τύποι διαγραμμάτων δραστηριοτήτων και περιπτώσεων χρήσης (activity και use case diagrams αντίστοιχα) που αυτή προσφέρει.

Τα διαγράμματα δραστηριοτήτων περιγράφουν το πως οι δραστηριότητες που εμφανίζονται στο σύστημα συντονίζονται και σχετίζονται μεταξύ τους. Στα διαγράμματα αυτά παρουσιάζεται η αλληλουχία των δραστηριοτήτων που εκτελούνται ακόμα και αν αυτές είναι υπό συνθήκη ή παράλληλες.

Τα διαγράμματα περιπτώσεων χρήσης περιγράφουν το πως οι χρήστες ενός συστήματος σχετίζονται με τις περιπτώσεις χρήσης του. Οι περιπτώσεις χρήσης αναπαριστούν την λειτουργικότητα ενός συστήματος, όπως αυτή παρουσιάζεται σε όσους αλληλεπιδρούν με το σύστημα αυτό.

Έγινε διάκριση έξι περιπτώσεων χρήσης της εφαρμογής: πέντε για κάθε μια καρτέλα απεικόνισης και μια για όλο το σύστημα. Για κάθε μία από αυτές παρουσιάζονται τα διαγράμματα δραστηριοτήτων και περιπτώσεων χρήσης.

Καρτέλα πρώτη

Η πρώτη περίπτωση χρήσης αφορούσε την πρώτη καρτέλα απεικόνισης της εφαρμογής και αναλύεται ως εξής:

1. Τίτλος περίπτωσης χρήσης

Χρήση πρώτης καρτέλας

2. Σύντομη Περιγραφή

Ο χρήστης έχει μεταβεί στην πρώτη καρτέλα του συστήματος και καλείται να εισάγει τα στοιχεία που επιθυμεί για να εμφανίσει τα δεδομένα που αντιστοιχούν στα στοιχεία αυτά υπό τη μορφή γραφήματος ενός κόμβου για ένα χαρακτηριστικό. Τα στοιχεία είναι κωδικός κόμβου, χαρακτηριστικό, χρονική περίοδος.

3. Ροή γεγονότων

3.1 Κύρια ροή

Το σύστημα επαληθεύει την εγκυρότητα και διαθεσιμότητα των δεδομένων που συμφωνούν με τα κριτήρια που έθεσε ο χρήστης και του εμφανίζει το γράφημα ενός κόμβου για ένα χαρακτηριστικό.

3.2 Δευτερεύουσες Ροές

Δεν υπάρχουν για αυτή την περίπτωση χρήσης.

3.3 Εναλλακτικές Ροές

A-1: Τα στοιχεία που έχουν εισαχθεί δεν είναι έγκυρα ή δεν υπάρχουν δεδομένα που να συμφωνούν με τα κριτήρια που έθεσε ο χρήστης. Θα εμφανιστεί ένα μήνυμα σφάλματος και ο χρήστης μπορεί να δοκιμάσει να εισάγει ξανά νέα στοιχεία οπότε η περίπτωση χρήσης ξεκινά πάλι ή να μεταβεί σε διαφορετική καρτέλα απεικόνισης και η περίπτωση χρήσης τερματίζει.

4. Μη λειτουργικές Απαιτήσεις

Δεν υπάρχουν για αυτή την περίπτωση χρήσης.

5. Κατάσταση Εισόδου

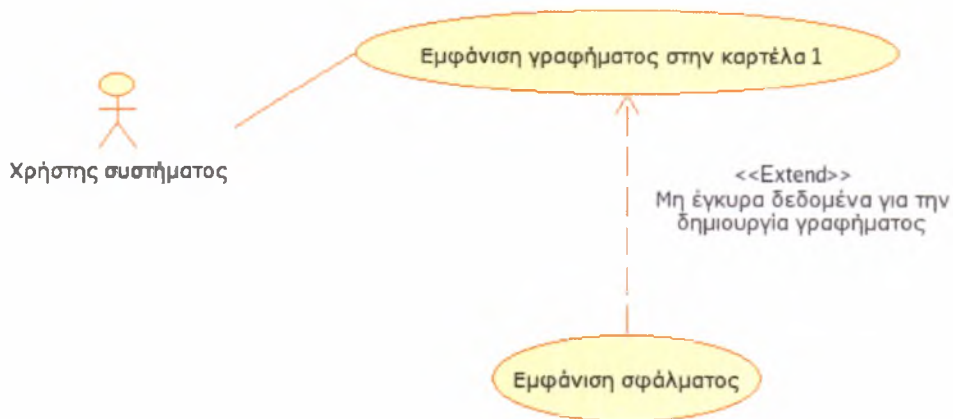
Ο χρήστης πρέπει να βρίσκεται στην πρώτη καρτέλα απεικόνισης.

6. Κατάσταση Εξόδου

Έχει /Δεν έχει εμφανιστεί το γράφημα.

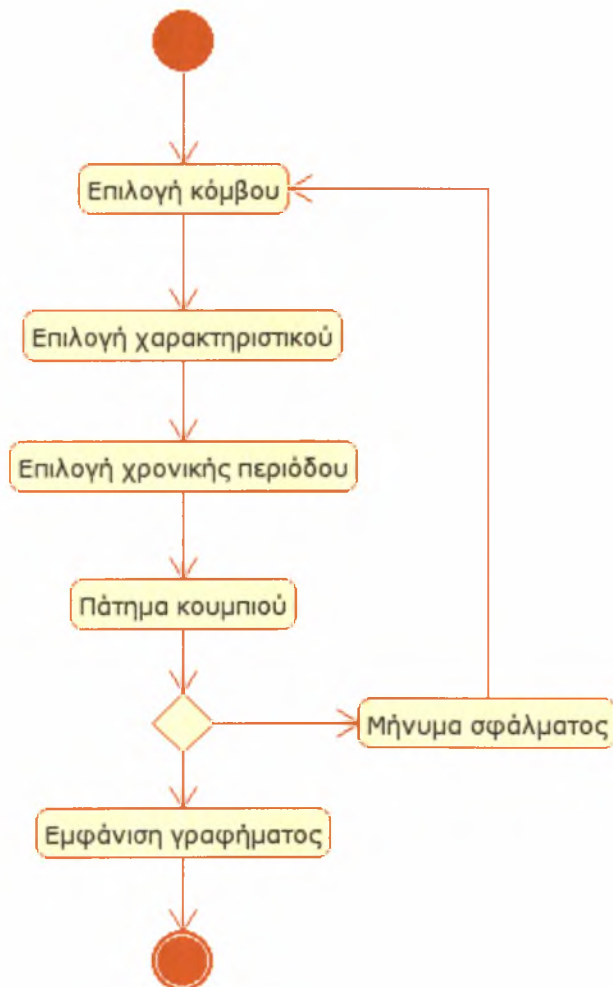
Το διάγραμμα για την συγκεκριμένη περίπτωση χρήσης είναι το ακόλουθο:

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων



Εικόνα 3: 1^η περίπτωση χρήσης - εμφάνιση γραφήματος στην πρώτη καρτέλα

Το διάγραμμα δραστηριοτήτων για την πρώτη καρτέλα είναι το εξής:



Εικόνα 4 : Το διάγραμμα δραστηριοτήτων για την πρώτη καρτέλα

Καρτέλα δεύτερη

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

Η δεύτερη περίπτωση χρήσης αφορούσε την δεύτερη καρτέλα απεικόνισης της εφαρμογής και αναλύεται ως εξής:

1. Τίτλος περίπτωσης χρήσης

Χρήση δεύτερης καρτέλας

2. Σύντομη Περιγραφή

Ο χρήστης έχει μεταβεί στην δεύτερη καρτέλα του συστήματος και καλείται να εισάγει τα στοιχεία που επιθυμεί για να εμφανίσει τα δεδομένα που αντιστοιχούν στα στοιχεία αυτά υπό τη μορφή γραφημάτων ενός κόμβου για πολλά χαρακτηριστικά (ένα γράφημα ανά χαρακτηριστικό). Τα στοιχεία είναι κωδικός κόμβου, χαρακτηριστικά, χρονική περίοδος.

3. Ροή γεγονότων

3.1 Κύρια ροή

Το σύστημα επαληθεύει την εγκυρότητα και διαθεσιμότητα των δεδομένων που συμφωνούν με τα κριτήρια που έθεσε ο χρήστης και του εμφανίζει τα γραφήματα ενός κόμβου για όσα χαρακτηριστικά έχει επιλέξει, το ένα κάτω από το άλλο.

3.2 Δευτερεύουσες Ροές

Δεν υπάρχουν για αυτή την περίπτωση χρήσης.

3.3 Εναλλακτικές Ροές

A-1: Τα στοιχεία που έχουν εισαχθεί δεν είναι έγκυρα ή δεν υπάρχουν δεδομένα που να συμφωνούν με τα κριτήρια που έθεσε ο χρήστης. Θα εμφανιστεί ένα μήνυμα σφάλματος και ο χρήστης μπορεί να δοκιμάσει να εισάγει ξανά νέα στοιχεία οπότε η περίπτωση χρήσης ξεκινά πάλι ή να μεταβεί σε διαφορετική καρτέλα απεικόνισης και η περίπτωση χρήσης τερματίζει.

4. Μη λειτουργικές Απαιτήσεις

Δεν υπάρχουν για αυτή την περίπτωση χρήσης.

5. Κατάσταση Εισόδου

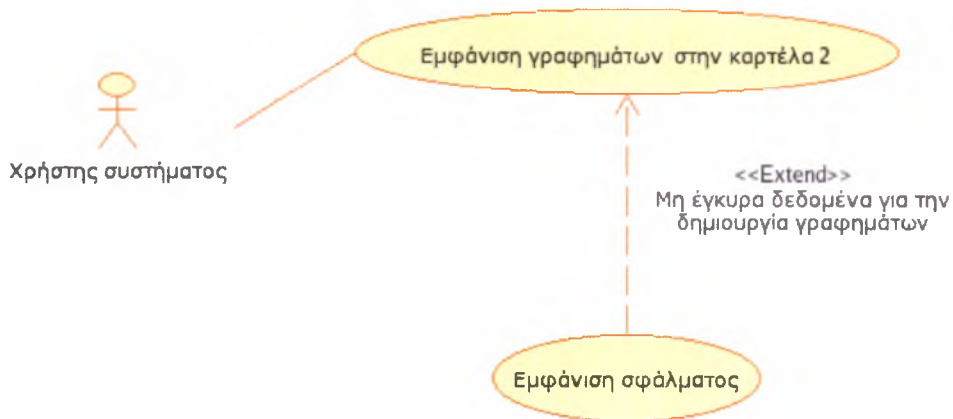
Ο χρήστης πρέπει να βρίσκεται στην δεύτερη καρτέλα απεικόνισης.

6. Κατάσταση Εξόδου

Έχουν /Δεν έχουν εμφανιστεί τα γραφήματα.

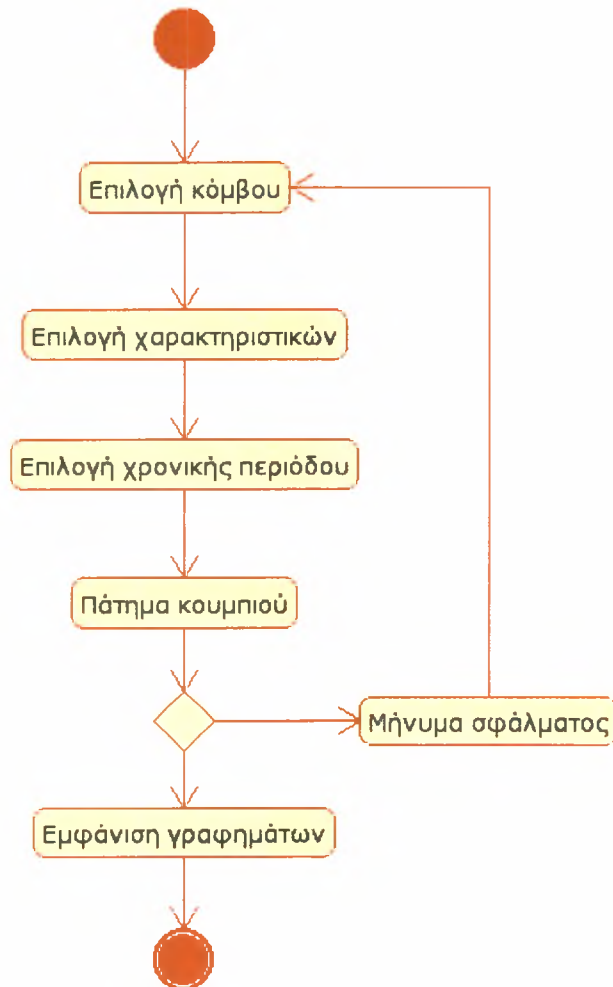
Το διάγραμμα για την συγκεκριμένη περίπτωση χρήσης είναι το ακόλουθο:

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων



Εικόνα 5 : 2^η περίπτωση χρήσης – εμφάνιση γραφημάτων στην δεύτερη καρτέλα

Το διάγραμμα δραστηριοτήτων για την δεύτερη καρτέλα είναι το εξής:



Εικόνα 6 : Το διάγραμμα δραστηριοτήτων για την δεύτερη καρτέλα

Καρτέλα τρίτη

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

Η τρίτη περίπτωση χρήσης αφορούσε την τρίτη καρτέλα απεικόνισης της εφαρμογής και αναλύεται ως εξής:

1. Τίτλος περίπτωσης χρήσης

Χρήση τρίτης καρτέλας

2. Σύντομη Περιγραφή

Ο χρήστης έχει μεταβεί στην τρίτη καρτέλα του συστήματος και καλείται να εισάγει τα στοιχεία που επιθυμεί για να εμφανίσει τα δεδομένα που αντιστοιχούν στα στοιχεία αυτά υπό τη μορφή γραφήματος ενός χαρακτηριστικού για πολλούς κόμβους. Τα στοιχεία είναι κωδικοί κόμβων, χαρακτηριστικό, χρονική περίοδος.

3. Ροή γεγονότων

3.1 Κύρια ροή

Το σύστημα επαληθεύει την εγκυρότητα και διαθεσιμότητα των δεδομένων που συμφωνούν με τα κριτήρια που έθεσε ο χρήστης και του εμφανίζει το γράφημα ενός χαρακτηριστικού για πολλούς κόμβους.

3.2 Δευτερεύουσες Ροές

Δεν υπάρχουν για αυτή την περίπτωση χρήσης.

3.3 Εναλλακτικές Ροές

A-1: Τα στοιχεία που έχουν εισαχθεί δεν είναι έγκυρα ή δεν υπάρχουν δεδομένα που να συμφωνούν με τα κριτήρια που έθεσε ο χρήστης. Θα εμφανιστεί ένα μήνυμα σφάλματος και ο χρήστης μπορεί να δοκιμάσει να εισάγει ξανά νέα στοιχεία οπότε η περίπτωση χρήσης ξεκινά πάλι ή να μεταβεί σε διαφορετική καρτέλα απεικόνισης και η περίπτωση χρήσης τερματίζει.

4. Μη λειτουργικές Απαιτήσεις

Δεν υπάρχουν για αυτή την περίπτωση χρήσης.

5. Κατάσταση Εισόδου

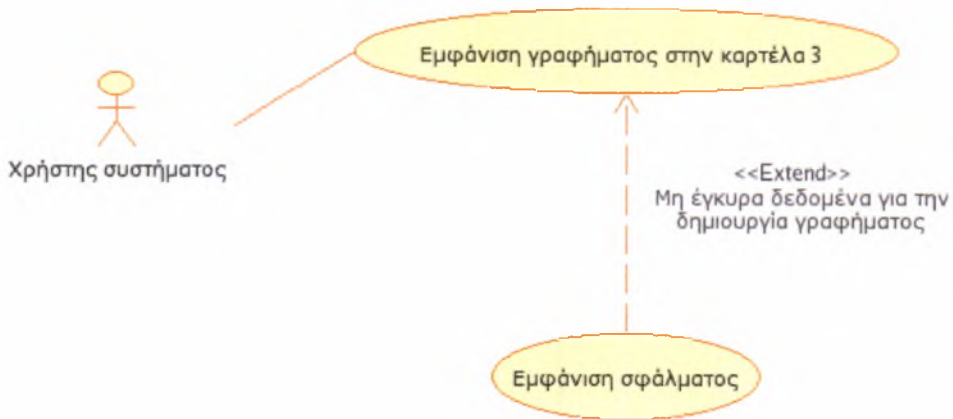
Ο χρήστης πρέπει να βρίσκεται στην τρίτη καρτέλα απεικόνισης.

6. Κατάσταση Εξόδου

Έχει /Δεν έχει εμφανιστεί το γράφημα.

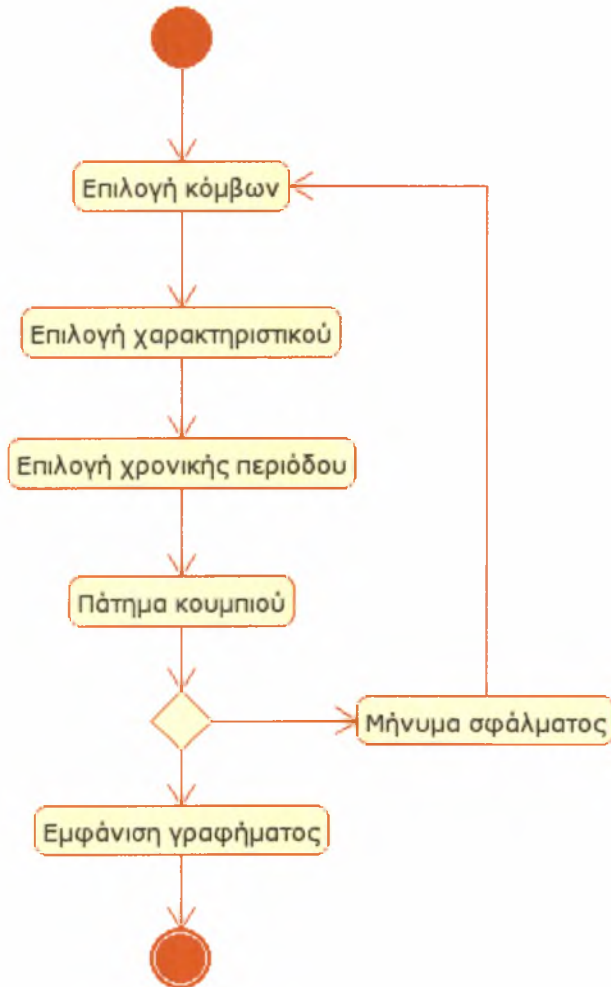
Το διάγραμμα για την συγκεκριμένη περίπτωση χρήσης είναι το ακόλουθο:

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων



Εικόνα 7 : 3^η περίπτωση χρήσης – εμφάνιση γραφήματος στην τρίτη καρτέλα

Το διάγραμμα δραστηριοτήτων για την τρίτη καρτέλα είναι το εξής:



Εικόνα 8 : Το διάγραμμα δραστηριοτήτων για την τρίτη καρτέλα

Καρτέλα τέταρτη

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

Η τέταρτη περίπτωση χρήσης αφορούσε την τέταρτη καρτέλα απεικόνισης της εφαρμογής και αναλύεται ως εξής:

1. Τίτλος περίπτωσης χρήσης

Χρήση τέταρτης καρτέλας

2. Σύντομη Περιγραφή

Ο χρήστης έχει μεταβεί στην τέταρτη καρτέλα του συστήματος και αφού περιμένει να φορτώσει το Java applet βλέπει σε γράφημα πραγματικού χρόνου τα δεδομένα τριών κόμβων για ένα χαρακτηριστικό. Αν επιθυμεί μπορεί να επιλέξει άλλους κόμβους ή άλλο χαρακτηριστικό.

3. Ροή γεγονότων

3.1 Κύρια ροή

Το σύστημα επαληθεύει την διαθεσιμότητα των δεδομένων από την βάση δεδομένων και του εμφανίζει το γράφημα πραγματικού χρόνου για τρεις κόμβους και ενός χαρακτηριστικού.

3.2 Δευτερεύουσες Ροές

S-1: Αν ο χρήστης επιλέξει διαφορετικούς κόμβους για απεικόνιση πρέπει να επιλέξει το ίδιο ή κάποιο άλλο χαρακτηριστικό για να αρχίσει το γράφημα να εμφανίζει τα νέα δεδομένα.

S-2: Αν ο χρήστης επιλέξει διαφορετικό χαρακτηριστικό το γράφημα ξεκινά να εμφανίζει τα νέα δεδομένα.

3.3 Εναλλακτικές Ροές

A-1: Τα στοιχεία που έχουν εισαχθεί δεν είναι έγκυρα ή δεν υπάρχουν δεδομένα ή υπάρχει κάποιο πρόβλημα με την σύνδεση του applet στην βάση δεδομένων. Δεν θα εμφανιστεί καθόλου γράφημα και ο χρήστης μπορεί να δοκιμάσει να εισάγει ξανά νέα στοιχεία ή να μεταβεί σε διαφορετική καρτέλα απεικόνισης και η περίπτωση χρήσης τερματίζει.

A-2: Υπάρχει κάποιο πρόβλημα με την εκτέλεση του applet. Ο χρήστης μπορεί να δει τα μηνύματα σφάλματος που αναφέρει το περιβάλλον εκτέλεσης της Java ή να μεταβεί σε διαφορετική καρτέλα απεικόνισης και η περίπτωση χρήσης τερματίζει.

4. Μη λειτουργικές Απαιτήσεις

Δεν υπάρχουν για αυτή την περίπτωση χρήσης.

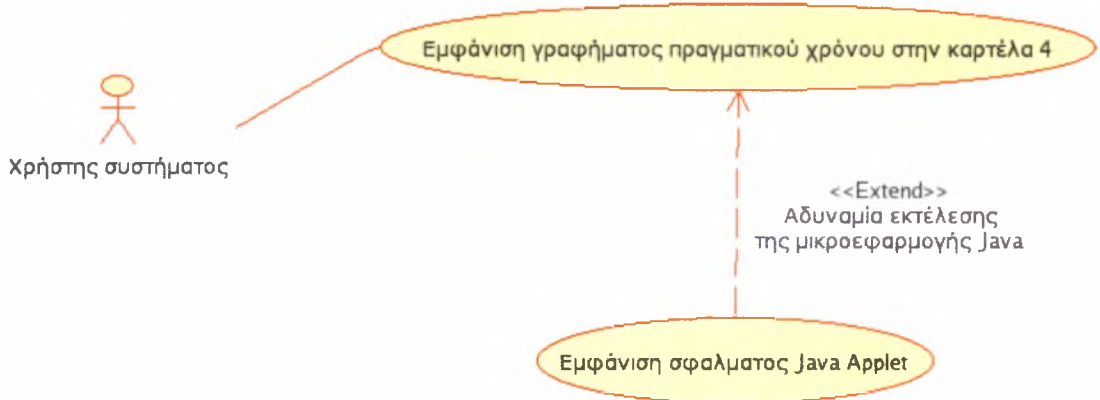
5. Κατάσταση Εισόδου

Ο χρήστης πρέπει να βρίσκεται στην τέταρτη καρτέλα απεικόνισης και να έχει δυνατότητα εκτέλεσης Java applets.

6. Κατάσταση Εξόδου

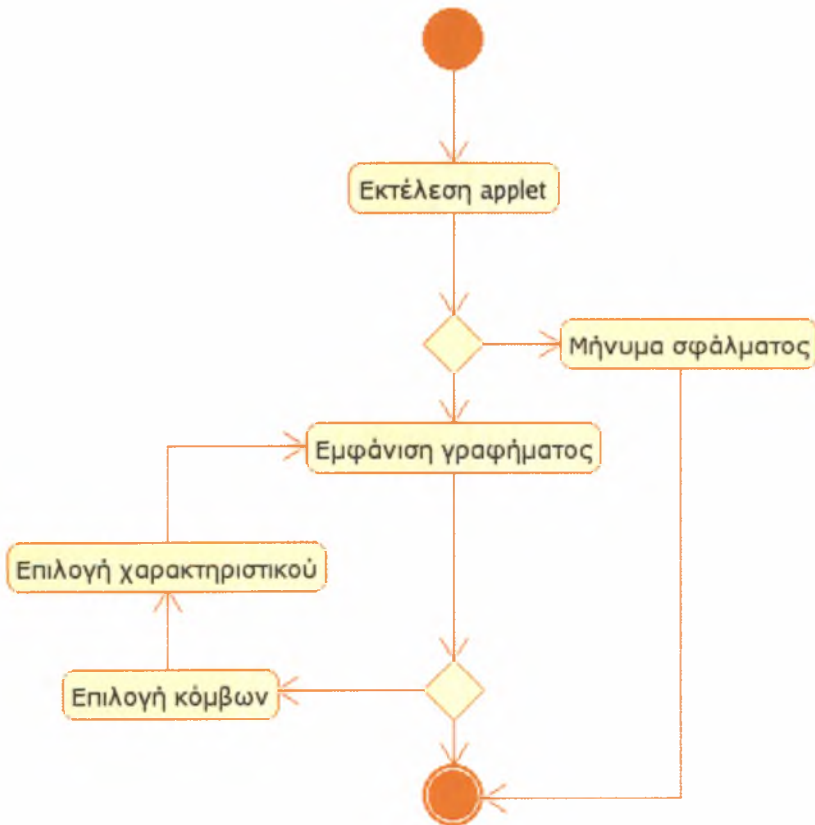
Έχει /Δεν έχει εμφανιστεί το γράφημα πραγματικού χρόνου.

Το διάγραμμα για την συγκεκριμένη περίπτωση χρήσης είναι το ακόλουθο:



Εικόνα 9 : 4^η περίπτωση χρήσης – εμφάνιση γραφήματος πραγματικού χρόνου στην τέταρτη καρτέλα

Το διάγραμμα δραστηριοτήτων για την τέταρτη καρτέλα είναι το εξής:



Εικόνα 10 : Το διάγραμμα δραστηριοτήτων για την τέταρτη καρτέλα

Καρτέλα πέμπτη

Η πέμπτη περίπτωση χρήσης αφορούσε την πέμπτη καρτέλα απεικόνισης της εφαρμογής και αναλύεται ως εξής:

1. Τίτλος περίπτωσης χρήσης

Χρήση πέμπτης καρτέλας

2. Σύντομη Περιγραφή

Ο χρήστης έχει μεταβεί στην πέμπτη καρτέλα του συστήματος και καλείται να εισάγει τα στοιχεία που επιθυμεί για να εμφανίσει τα δεδομένα που αντιστοιχούν στα στοιχεία αυτά υπό τη μορφή πίνακα τιμών. Τα στοιχεία είναι κωδικοί κόμβου, χαρακτηριστικό, χρονική περίοδος, λογική έκφραση για το χαρακτηριστικό (σχέση, τιμή), τρόπος ταξινόμησης (βάση συγκεκριμένου πεδίου, αύξουσα / φθίνουσα σειρά).

3. Ροή γεγονότων

3.1 Κύρια ροή

Το σύστημα επαληθεύει την εγκυρότητα και διαθεσιμότητα των δεδομένων που συμφωνούν με τα κριτήρια που έθεσε ο χρήστης και του εμφανίζει τον πίνακα τιμών

3.2 Δευτερεύουσες Ροές

Δεν υπάρχουν για αυτή την περίπτωση χρήσης.

3.3 Εναλλακτικές Ροές

A-1: Τα στοιχεία που έχουν εισαχθεί δεν είναι έγκυρα ή δεν υπάρχουν δεδομένα που να συμφωνούν με τα κριτήρια που έθεσε ο χρήστης. Θα εμφανιστεί ένας κενός πίνακας τιμών και ο χρήστης μπορεί να δοκιμάσει να εισάγει ξανά νέα στοιχεία οπότε η περίπτωση χρήσης ξεκινά πάλι ή να μεταβεί σε διαφορετική καρτέλα απεικόνισης και η περίπτωση χρήσης τερματίζει.

4. Μη λειτουργικές Απαιτήσεις

Δεν υπάρχουν για αυτή την περίπτωση χρήσης.

5. Κατάσταση Εισόδου

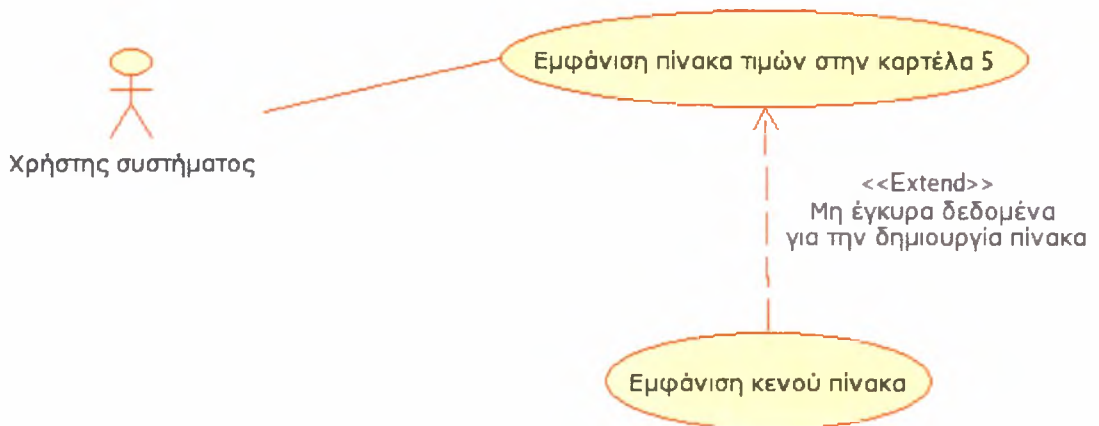
Ο χρήστης πρέπει να βρίσκεται στην πέμπτη καρτέλα απεικόνισης.

6. Κατάσταση Εξόδου

Έχει /Δεν έχει δεδομένα ο πίνακας τιμών.

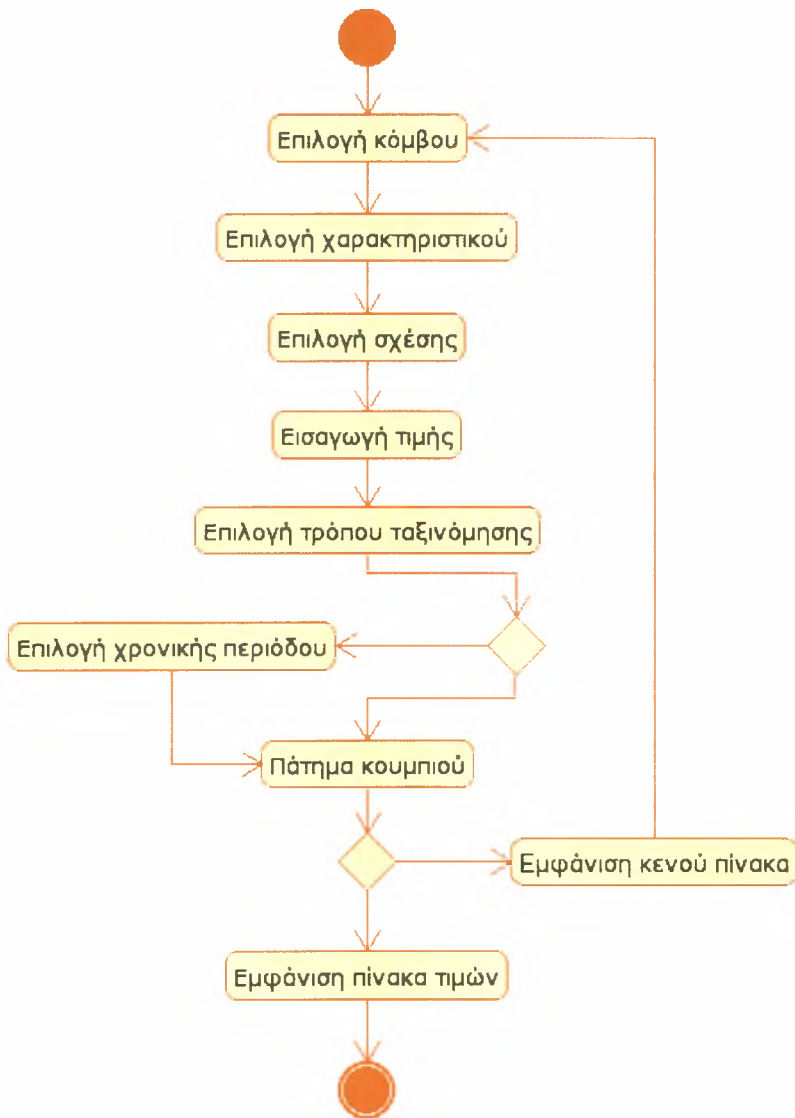
Το διάγραμμα για την συγκεκριμένη περίπτωση χρήσης είναι το ακόλουθο:

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων



Εικόνα 11 : 5^η περίπτωση χρήσης – εμφάνιση πίνακα τιμών στην πέμπτη καρτέλα

Το διάγραμμα δραστηριοτήτων για την πέμπτη καρτέλα είναι το εξής:



Εικόνα 12 : Το διάγραμμα δραστηριοτήτων για την πέμπτη καρτέλα

Γενική χρήση του συστήματος

Η τελευταία περίπτωση χρήσης αφορούσε την γενική χρήση της εφαρμογής και αναλύεται ως εξής:

1. Τίτλος περίπτωσης χρήσης

Γενική χρήση του συστήματος

2. Σύντομη Περιγραφή

Ο χρήστης μεταβαίνει στην ιστοσελίδα της εφαρμογής και βρίσκεται κατευθείαν στην πρώτη καρτέλα. Μπορεί είτε να την χρησιμοποιήσει ή να μεταβεί σε κάποια άλλη καρτέλα απεικόνισης.

3. Ροή γεγονότων

3.1 Κύρια ροή

Αν ο χρήστης παραμείνει σε αυτήν τη καρτέλα μεταβαίνουμε στην περίπτωση χρήσης της πρώτης καρτέλας.

3.2 Δευτερεύουσες Ροές

Δεν υπάρχουν για αυτή την περίπτωση χρήσης.

3.2 Εναλλακτικές Ροές

A-1: Ο χρήστης μεταβαίνει σε άλλη καρτέλα απεικόνισης οπότε μεταβαίνουμε στην περίπτωση χρήσης της συγκεκριμένης καρτέλας.

4. Μη λειτουργικές Απαιτήσεις

Δεν υπάρχουν για αυτή την περίπτωση χρήσης.

5. Κατάσταση Εισόδου

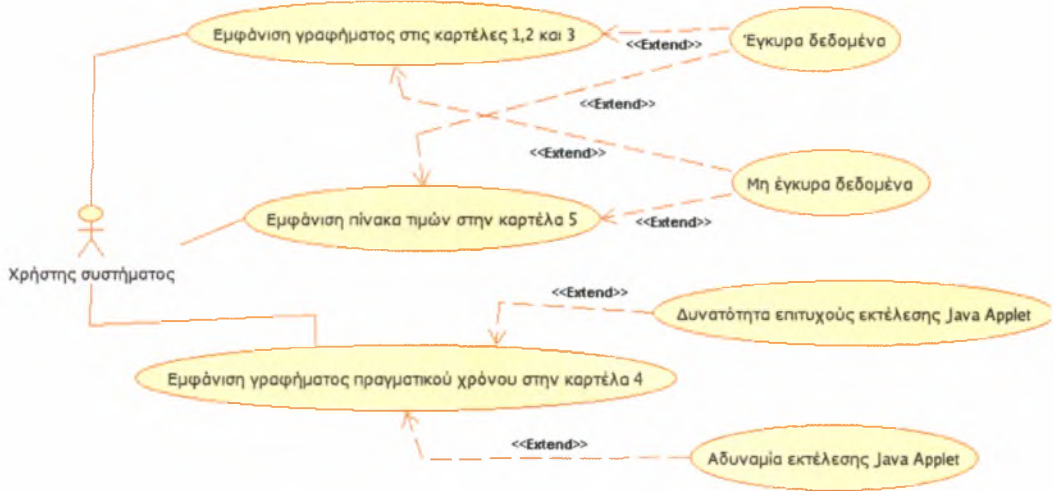
Δεν υπάρχει για αυτή την περίπτωση χρήσης.

6. Κατάσταση Εξόδου

Δεν υπάρχει για αυτή την περίπτωση χρήσης.

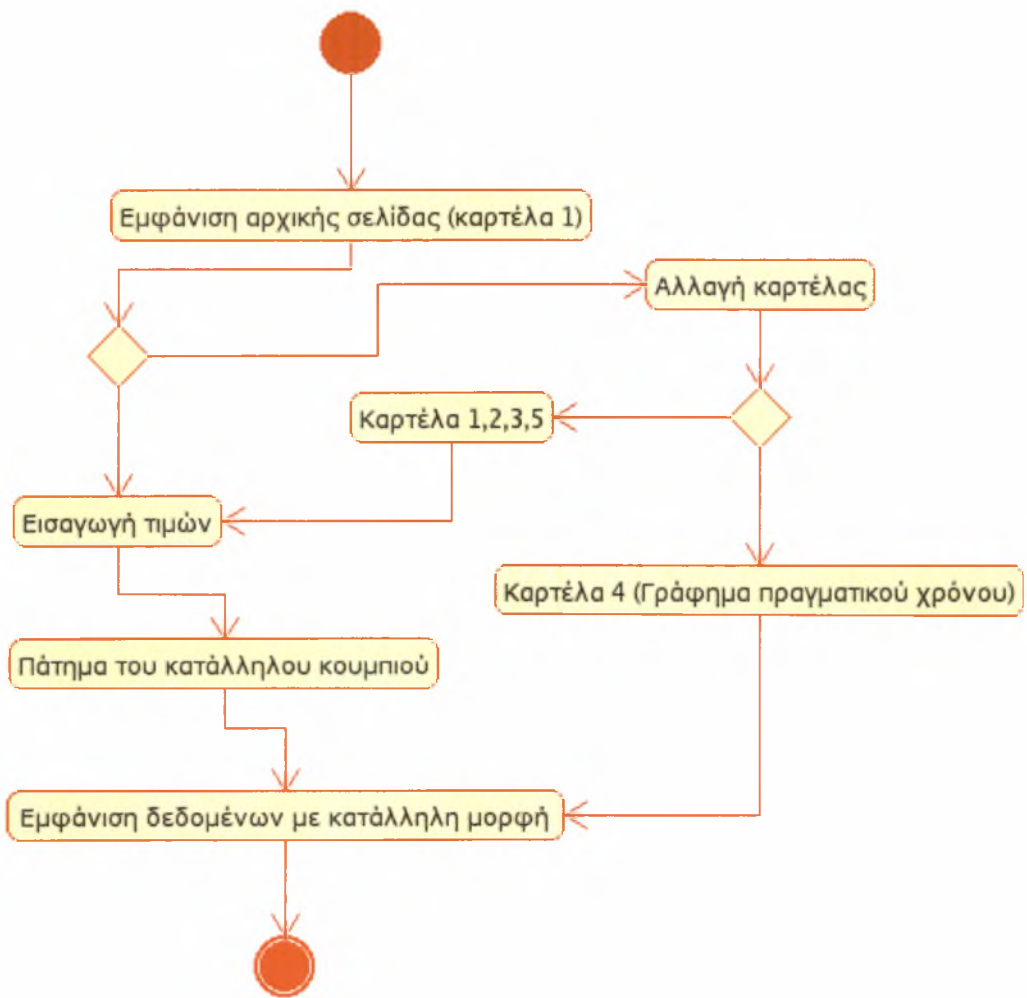
Το διάγραμμα για την συγκεκριμένη περίπτωση χρήσης του συστήματος είναι το ακόλουθο:

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων



Εικόνα 13 : Γενική περίπτωση χρήσης του συστήματος

Το διάγραμμα δραστηριοτήτων για την συγκεκριμένη περίπτωση χρήσης είναι το εξής:



**Εικόνα 14 : Διάγραμμα δραστηριοτήτων για την περίπτωση
χρήσης του συστήματος**

3.3 Σχεδίαση

Έχοντας αναλύσει τις απαιτήσεις και λειτουργίες του συστήματος και λαμβάνοντάς αυτές υπόψη μας περνάμε στο στάδιο της σχεδίασης του συστήματος το οποίο ακολουθεί με την παρουσίαση της αρχιτεκτονικής που ακολουθήθηκε.

3.3.1 Αρχιτεκτονική

Ο σκοπός της εργασίας, όπως παρουσιάστηκε και στην περίληψη, είναι η εμφάνιση των δεδομένων που συλλέγονται από τους αισθητήρες μέσα από μια διαδικτυακή διεπαφή (internet based interface), δηλαδή, μια διεπαφή την οποία ο χρήστης βλέπει σαν ιστοσελίδα. Έτσι, μπορεί να βρίσκεται σε εντελώς διαφορετικό γεωγραφικό χώρο από ότι το δίκτυο αισθητήρων και εντούτοις να έχει πρόσβαση στα δεδομένα που συλλέγονται από αυτό.

Η πιο γνωστή αρχιτεκτονική τα τελευταία είκοσι χρόνια – κυρίως σε εφαρμογές που περιλαμβάνουν βάσεις δεδομένων – είναι αυτή του πελάτη - εξυπηρετητή (client - server). Σε αυτή την αρχιτεκτονική, ο πελάτης στέλνει ένα αίτημα (request) για δεδομένα στον εξυπηρετητή και αυτός επιστρέφει την απάντηση (response), την οποία επεξεργάζεται ο πελάτης και εμφανίζει στο χρήστη τα αποτελέσματα. Το πρόβλημα με αυτή την προσέγγιση είναι ότι η εμφάνιση των δεδομένων και η επεξεργασία τους γίνεται από το ίδιο πρόγραμμα, το πρόγραμμα - πελάτη. Αν υπάρχουν πολλαπλά κανάλια διάχυσης της πληροφορίας ή συχνή αλλαγή στη μορφή παρουσίασης, τότε θα πρέπει να αλλάζει κάθε φορά η εφαρμογή πελάτη.

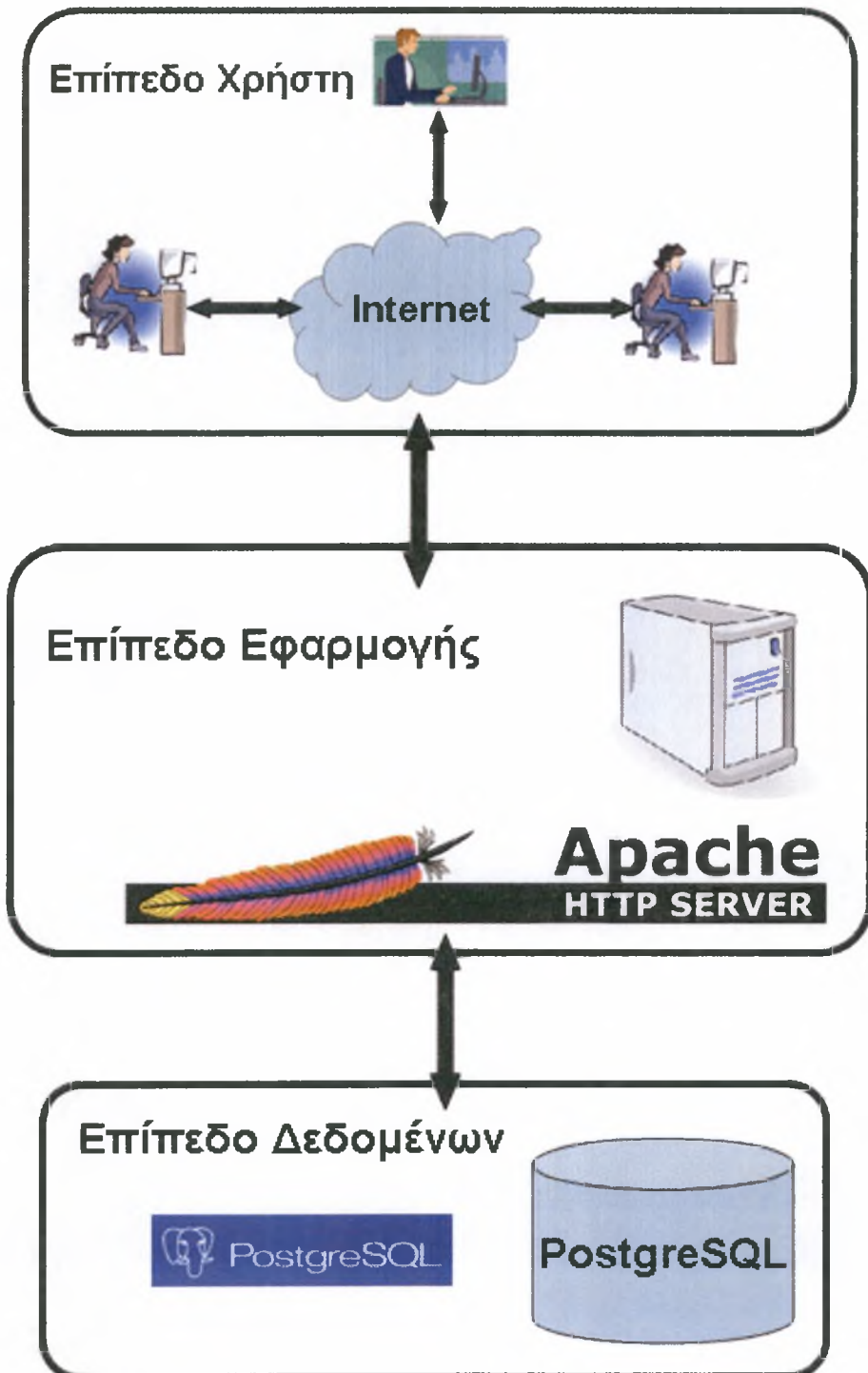
Τα τελευταία χρόνια με την εμφάνιση του διαδικτύου, έχει επικρατήσει σε διαδικτυακές εφαρμογές η αρχιτεκτονική τριών επιπέδων (3-tier architecture) η οποία τοποθετεί σε διαφορετικά εννοιολογικά επίπεδα τη λογική του προγράμματος που επεξεργάζεται τα δεδομένα (application logic) από τον τρόπο και μέσο που γίνεται η παρουσίαση (presentation). Αυτή είναι η προσέγγιση και για την παρούσα εργασία.

Η συγκεκριμένη αρχιτεκτονική αποτελείται από (*Εικόνα 15*):

1. Επίπεδο χρήστη (Client Layer) – Το επίπεδο χρήστη παρέχει την διεπαφή μέσα από την οποία οι χρήστες έχουν πρόσβαση στο σύστημα.
2. Επίπεδο εφαρμογής (Application Layer) – Το επίπεδο εφαρμογής εμπεριέχει την βασική λειτουργικότητα του συστήματος.

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

3. Επίπεδο δεδομένων (Data Layer) – Το επίπεδο δεδομένων είναι το τμήμα της εφαρμογής που αφορά την αποθήκευση και διαχείριση των δεδομένων.



Εικόνα 15 : Η αρχιτεκτονική τριών επιπέδων.

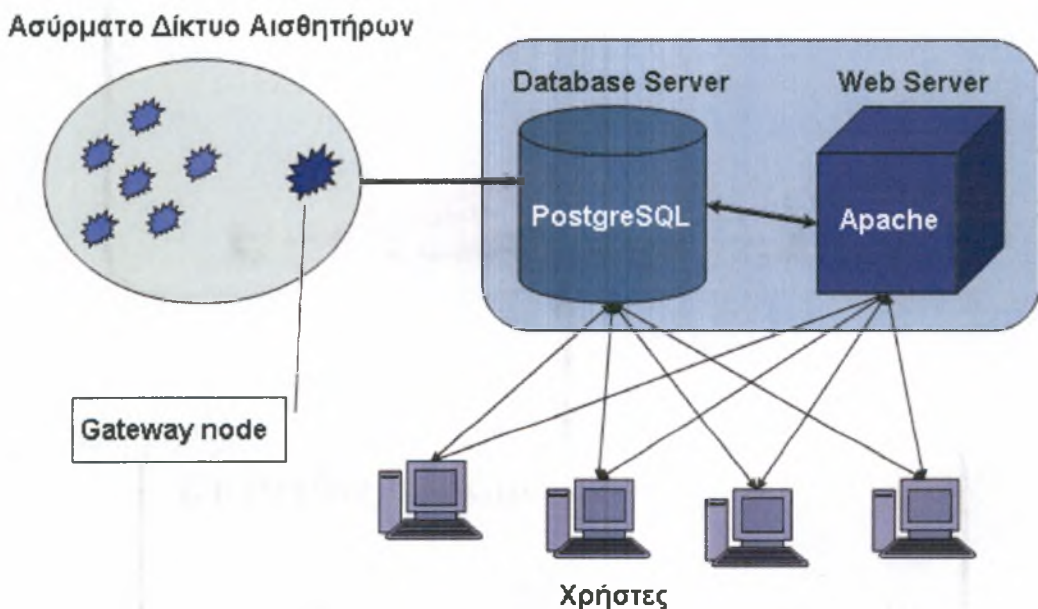
Η εφαρμογή που υλοποιήθηκε, για τα τρία παραπάνω επίπεδα περιλαμβάνει:

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

- Το επίπεδο χρήστη αποτελείται από σελίδες γραμμένες σε HTML [26] και PHP [27].
- Το επίπεδο εφαρμογής χρησιμοποιεί την γλώσσα PHP και τις διάφορες βιβλιοθήκες γραφημάτων καθώς και μικροεφαρμογές Java (applets) [28]. Χρησιμοποιώντας αυτές μπορεί να ανακτήσει δεδομένα και να στείλει ερωτήματα στη βάση δεδομένων.
- Το επίπεδο δεδομένων χρησιμοποιεί την βάση δεδομένων PostgreSQL [29] για να διαχειρίζεται και να ανακτά δεδομένα και να θέτει ερωτήματα.

Σε αυτή την αρχιτεκτονική ένας φυλλομετρητής ιστού (web browser) δρα ως πελάτης, ένας εξυπηρετητής ιστού (web server) υλοποιεί την λογική της εφαρμογής και ένας εξυπηρετητής βάσεων δεδομένων (database server), που μπορεί να βρίσκεται στο ίδιο μηχάνημα με τον εξυπηρετητή ιστού ή σε διαφορετικό, χειρίζεται λειτουργίες της βάσης δεδομένων.

Η αρχιτεκτονική της εφαρμογής (*Εικόνα 16*) στην πράξη είναι η εξής:



Εικόνα 16 : Η αρχιτεκτονική του συστήματος που υλοποιήθηκε

Τα δεδομένα του δικτύου αισθητήρων μέσω του κόμβου πύλη (gateway node) αποθηκεύονται σε μια βάση δεδομένων από όπου ανακτώνται από τον εξυπηρετητή ιστού και χρησιμοποιούνται για να δημιουργηθούν τα γραφήματα που εμφανίζονται στον τελικό χρήστη. Στην περίπτωση της απεικόνισης σε πραγματικό χρόνο τα

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

δεδομένα ανακτώνται μέσα από την μικροεφαρμογή Java κατευθείαν από την βάση δεδομένων χωρίς την διαμεσολάβηση του εξυπηρετητή ιστού.

Όπως αναφέρθηκε παραπάνω, το σύστημα αποτελείται από δυο βασικά μέρη:

- 1) Εξυπηρετητής Βάσης Δεδομένων (DataBase Server): περιέχει τα δεδομένα που συγκεντρώνονται από τους αισθητήρες. Σε αυτόν είναι συνδεδεμένος και ο κόμβος πύλη (gateway node) ο οποίος είναι υπεύθυνος να συγκεντρώνει τα δεδομένα από όλο το δίκτυο αισθητήρων τα οποία στην συνέχεια με κατάλληλο λογισμικό προωθούνται στην βάση δεδομένων.
- 2) Εξυπηρετητής ιστού: περιέχει το web περιβάλλον και εμφανίζει στον χρήστη μια διεπαφή σε μορφή ιστοσελίδας μέσα από την οποία μπορεί να πάρει τις πληροφορίες που επιθυμεί.

Τα δύο παραπάνω συστήματα μπορούν είτε να βρίσκονται στο ίδιο υπολογιστικό σύστημα, είτε σε διαφορετικά υπολογιστικά συστήματα πράγμα που κάνει την εφαρμογή πιο ευέλικτη.

3.4 Υλοποίηση

3.4.1 Επιμέρους τμήματα

Στην υλοποίηση της παραπάνω αρχιτεκτονικής χρησιμοποιήθηκε λογισμικό ανοιχτού κώδικα (open source). Συγκεκριμένα χρησιμοποιήθηκαν οι εξής γλώσσες προγραμματισμού, βιβλιοθήκες και εργαλεία:

- Εξυπηρετητής διαδυκτίου (web server): *Apache 2* [30]
- Βάση δεδομένων (database server) : *postgreSQL* [29]
- Κατασκευή γραφημάτων: *JpGraph* PHP graph library [31] και *GD* graphics library [32].
- Δημιουργία της μικροεφαρμογής απεικόνισης δεδομένων σε πραγματικό χρόνο: *JfreeChart* Java chart library [33] και γλώσσα προγραμματισμού *Java* [34].
- Δημιουργία διεπαφής και λειτουργιών: συνδυασμός *PHP 5* [27] (με *gd* και *pgsql* modules), *HTML* [26], *javascript* [35] και *CSS* [36] για μορφοποίηση της ιστοσελίδας.
- Από την πλευρά των αισθητήρων οι εφαρμογές: *tinyOS* [37] ενώ κάθε κόμβος είχε προεγκατεστημένο το πρωτόκολλο surge της Crossbow [38] . Το tinyOS

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

είναι το λειτουργικό σύστημα που τρέχουν οι κόμβοι ενώ το surge είναι το πρωτόκολλο το οποίο εξασφαλίζει την ενεργειακά επαρκή (efficient) λειτουργία των κόμβων. Για την συλλογή των δεδομένων χρησιμοποιήθηκε η εφαρμογή serial forwarder που έχει αναπτυχθεί για συγκέντρωση των δεδομένων από τους αισθητήρες.

3.4.2 Η δομή της βάσης δεδομένων

Η βάση δεδομένων είναι πολύ απλή καθώς αποτελείται από δυο πίνακες. Η μορφή των πινάκων είναι παρόμοια με αυτή που χρησιμοποιεί η *TinyDB* [39] και παρουσιάζεται και εξηγείται παρακάτω:

Όνομα Πεδίου	Τύπος Δεδομένων	Επεξήγηση
result_time	timestamp (ημερομηνία & ώρα)	Ημερομηνία & ώρα που αποθηκεύτηκε η εγγραφή
epoch	integer (ακέραιος)	Χρόνος από μια δεδομένη χρονική στιγμή σε δευτερόλεπτα
nodeid	integer (ακέραιος)	Ο αριθμός - ταυτότητα του κόμβου αισθητήρα
temp	integer (ακέραιος)	Θερμοκρασία
light	integer (ακέραιος)	Φως
accel x	integer (ακέραιος)	Επιτάχυνση στον άξονα X
accel y	integer (ακέραιος)	Επιτάχυνση στον άξονα Y
mag x	integer (ακέραιος)	Μαγνητισμός στον άξονα X
mag y	integer (ακέραιος)	Μαγνητισμός στον άξονα Y
humidity	integer (ακέραιος)	Υγρασία
voltage	integer (ακέραιος)	Τάση ρεύματος ενεργειακής πηγής

Πίνακας 2 : Ο πίνακας q1

Όνομα Πεδίου	Τύπος Δεδομένων	Επεξήγηση
result_time	timestamp (ημερομηνία & ώρα)	Ημερομηνία & ώρα που αποθηκεύτηκε η εγγραφή
epoch	real (πραγματικός)	Χρόνος από μια δεδομένη χρονική στιγμή σε δευτερόλεπτα
nodeid	real (πραγματικός)	Ο αριθμός - ταυτότητα του κόμβου αισθητήρα
temp	real (πραγματικός)	Θερμοκρασία
light	real (πραγματικός)	Φως
accel x	real (πραγματικός)	Επιτάχυνση στον άξονα X
accel y	real (πραγματικός)	Επιτάχυνση στον άξονα Y
mag x	real (πραγματικός)	Μαγνητισμός στον άξονα X
mag y	real (πραγματικός)	Μαγνητισμός στον άξονα Y
humidity	real (πραγματικός)	Υγρασία
voltage	real (πραγματικός)	Τάση ρεύματος ενεργειακής πηγής

Πίνακας 3 : Ο πίνακας q2

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

Η διαφορά αυτών των δυο πινάκων είναι ότι ο ένας (*q1*) περιέχει δεδομένα όπως έρχονται από τους αισθητήρες σε raw μορφή ενώ ο άλλος (*q2*) τα ίδια δεδομένα που έχουν μετατραπεί σε μορφή αναγνωρίσιμη από τον άνθρωπο. Η διαδικασία της μετατροπής γίνεται με την χρήση μιας “σκανδάλης” (trigger) που ενεργοποιείται στην βάση δεδομένων κάθε φορά που γίνεται μια εγγραφή στον πίνακα *q1*: τα δεδομένα της εγγραφής μετατρέπονται με κάποιο συγκεκριμένο τύπο και τα δεδομένα που προκύπτουν αποθηκεύονται στον πίνακα *q2*.

3.4.3 Μετασχηματισμοί

Οι τύποι μετασχηματισμού των δεδομένων θερμοκρασίας, επιτάχυνσης στους άξονες X και Y, μαγνητισμού στους άξονες X και Y, υγρασίας και τάσης ρεύματος είναι οι ακόλουθοι (η τιμή του φωτός δεν απαιτεί μετασχηματισμό):

Μετασχηματισμός για την θερμοκρασία :

$$R = \frac{10000 * (1023 - temp)}{temp} \\ \frac{1}{0.00130705 + (0.000214381 * \ln(R)) + (0.000000093 * (\ln(R))^3)} - 273.15$$

όπου “temp” η τιμή που έχει στείλει ο αισθητήρας για την θερμοκρασία.

Μετασχηματισμός για την επιτάχυνση στον άξονα X :

$$1.0 - \left(\frac{666 - accel_x}{197} \right)$$

όπου “accel_x” η τιμή που έχει στείλει ο αισθητήρας για την επιτάχυνση στον άξονα X.

Μετασχηματισμός για την επιτάχυνση στον άξονα Y :

$$1.0 - \left(\frac{666 - accel_y}{197} \right)$$

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

όπου “*accel_y*” η τιμή που έχει στείλει ο αισθητήρας για την επιτάχυνση στον άξονα Y.

Μετασχηματισμός για τον μαγνητισμό στον άξονα X :

$$\frac{mag_x}{(1.023 * 2.262 * 3.2)}$$

όπου “*mag_x*” η τιμή που έχει στείλει ο αισθητήρας για τον μαγνητισμό στον άξονα X.

Μετασχηματισμός για τον μαγνητισμό στον άξονα Y :

$$\frac{mag_y}{(1.023 * 2.262 * 3.2)}$$

όπου “*mag_y*” η τιμή που έχει στείλει ο αισθητήρας για τον μαγνητισμό στον άξονα Y.

Μετασχηματισμός για την υγρασία :

$$R = -4.0 + (0.0405 * humidity) - (0.0000028 * humidity^2)$$
$$(((0.0098 * temp) - 38.4) - 25.0) * (0.01 + 0.00008 * humidity) + R$$

όπου “*temp*” η τιμή που έχει στείλει ο αισθητήρας για την θερμοκρασία και “*humidity*” η τιμή που έχει στείλει ο αισθητήρας για την υγρασία.

Μετασχηματισμός για την τάση του ρεύματος της ενεργειακής πηγής :

$$\frac{1252.352}{voltage}$$

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

όπου “voltage” η τιμή που έχει στείλει ο αισθητήρας για την τάση του ρεύματος της ενεργειακής πηγής.

Σε γλώσσα SQL [40] η συνάρτηση των παραπάνω μετασχηματισμών είναι η εξής:

```
CREATE FUNCTION transform() RETURNS OPAQUE AS '
DECLARE
-----
TRmag_x REAL;

TRmag_y REAL;

TRaccel_y REAL;
tmpAcc_y REAL;

TRaccel_x REAL;
tmpAcc_x REAL;

TRhumidity REAL;
tmpTemp_for_Humidity REAL;

TRtemp REAL;
tmpTemp REAL;

TRlight REAL;

TRvoltage REAL;
-----
BEGIN
-----
TRmag_x := NEW.mag_x / (1.023*2.262*3.2);

TRmag_y := NEW.mag_y / (1.023*2.262*3.2);

tmpAcc_y := 666 - NEW.accel_y;
TRaccel_y := 1.0 - (tmpAcc_y/197);

tmpAcc_x := 666 - NEW.accel_x;
TRaccel_x := 1.0 - (tmpAcc_x/197);

tmpTemp_for_Humidity := (0.0098 * NEW.temp) - 38.4;
TRhumidity := -4.0 + (0.0405 * NEW.humidity) - (0.0000028 *
NEW.humidity * NEW.humidity);
TRhumidity := (tmpTemp_for_Humidity - 25.0) * (0.01 + 0.00008 *
NEW.humidity) + TRhumidity;

tmpTemp := 10000 * (1023 - NEW.temp) / NEW.temp;
TRtemp := 1 / (0.001307050 + (0.000214381 * ln(tmpTemp)) +
(0.000000093 * power(ln(tmpTemp),3)));
TRtemp := TRtemp - 273.15;

TRlight := NEW.light;

TRvoltage := 1252.352 / NEW.voltage;
-----
INSERT INTO q2 values
(NEW.result_time,NEW.epoch,NEW.nodeid,TRtemp,TRlight,TRaccel_x,TRacce
l_y,TRmag_x,TRmag_y,TRhumidity,TRvoltage);
RETURN NEW;
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```
END;  
' LANGUAGE 'plpgsql';
```

Αντίστοιχα, η εντολή που δημιουργεί την “σκανδάλη” (trigger) που ενεργοποιείται με κάθε προσθήκη εγγραφής στον πίνακα *q1* είναι η εξής:

```
CREATE TRIGGER transform_trigger AFTER INSERT ON q1 FOR EACH ROW  
EXECUTE PROCEDURE transform();
```

Τα αρχεία που δημιουργήθηκαν στα πλαίσια της εφαρμογής παρουσιάζονται συνοπτικά στον ακόλουθο πίνακα:

Όνομα Αρχείου	Σύντομη περιγραφή
<i>connect2db.php</i>	Παρέχει τη σύνδεση με την βάση δεδομένων
<i>footer.php</i>	Δημιουργεί το κάτω τμήμα της σελίδας
<i>graph1.php</i>	Δημιουργεί ένα απλό γράφημα
<i>graph2.php</i>	Δημιουργεί ένα γράφημα με τα δεδομένα πολλαπλών κόμβων
<i>header.php</i>	Δημιουργεί το επάνω τμήμα της σελίδας
<i>index.php</i>	Το αρχικό αρχείο της εφαρμογής
<i>tools.php</i>	Αρχείο με βοηθητικές συναρτήσεις και μεταβλητές
<i>view_tab1.php</i>	Δημιουργεί την σελίδα για την καρτέλα 1
<i>view_tab2.php</i>	Δημιουργεί την σελίδα για την καρτέλα 2
<i>view_tab3.php</i>	Δημιουργεί την σελίδα για την καρτέλα 3
<i>view_tabQDB.php</i>	Δημιουργεί την σελίδα για την καρτέλα “Ερωτήσεων στη Βάση Δεδομένων”
<i>view_tabRT.php</i>	Δημιουργεί την σελίδα για την καρτέλα “Εμφάνισης Δεδομένων σε Πραγματικό Χρόνο”
<i>sensorStyle.css</i>	Χρησιμοποιείται στην μορφοποίηση της σελίδας
<i>LiveGraph.jar</i>	Java Applet για την εμφάνιση δεδομένων σε πραγματικό χρόνο

Πίνακας 4 : Τα αρχεία της εφαρμογής

3.5 Έλεγχος εφαρμογής – Δοκιμαστική λειτουργία

Για τον έλεγχο της εφαρμογής αρχικά χρησιμοποιήθηκαν δοκιμαστικά δεδομένα τα οποία δημιουργήθηκαν τυχαία και αποθηκεύτηκαν στην βάση δεδομένων

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

χειροκίνητα. Στην συνέχεια, έγινε χρήση της εφαρμογής για διαφορετικές περιπτώσεις και σε όλες τις καρτέλες απεικόνισης χρησιμοποιώντας διάφορους συνδυασμούς εισαγόμενων από τον χρήστη στοιχείων. Σε αυτή την φάση το γράφημα πραγματικού χρόνου αποτελούνταν από ευθείες γραμμές καθώς δεν υπήρχε μεταβολή στα δεδομένα με την πάροδο του χρόνου ενώ τα υπόλοιπα γραφήματα των χαρακτηριστικών των κόμβων ήταν τυχαία γραφήματα. Τέλος, η εφαρμογή δοκιμάστηκε χρησιμοποιώντας τους πραγματικούς αισθητήρες Mica2 της Crossbow [38] για διάφορα σενάρια χρήσης. Σε αυτό το στάδιο όλες οι δοκιμές ήταν επιτυχείς και η εφαρμογή παρουσίασε την αναμενόμενη λειτουργία.

4 Παρουσίαση συστήματος

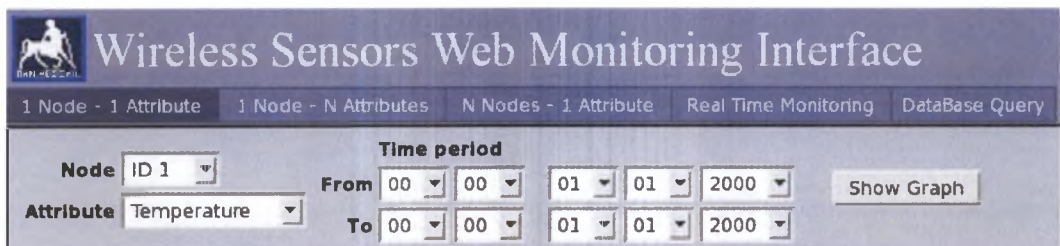
Η διεπαφή της εφαρμογής αποτελείται από πέντε “καρτέλες” (tabs) οι οποίες αντιστοιχούν σε πέντε τρόπους απεικόνισης των δεδομένων:

1. Ένας συγκεκριμένος κόμβος με ένα συγκεκριμένο χαρακτηριστικό (1 Node – 1 Attribute)
2. Ένας συγκεκριμένος κόμβος με πολλά χαρακτηριστικά του (1 Node – N Attributes)
3. Πολλοί κόμβοι με ένα συγκεκριμένο χαρακτηριστικό τους (N Nodes – 1 Attribute)
4. Εμφάνιση δεδομένων σε γράφημα πραγματικού χρόνου (Real Time Monitoring)
5. Εμφάνιση δεδομένων σε μορφή πίνακα με ερώτηση στην βάση δεδομένων (Database Query).

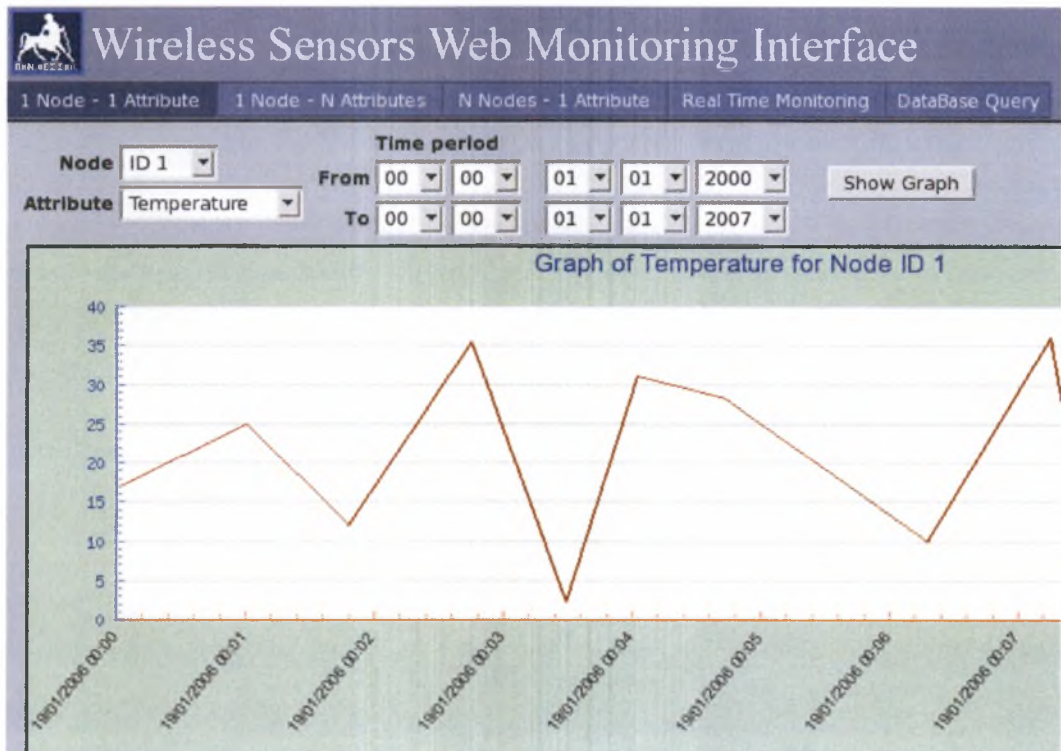
Γενικά οι επιλογές του χρήστη σε όλες τις καρτέλες, πλην της απεικόνισης των δεδομένων σε πραγματικό χρόνο αφορούν τον κόμβο, το χρονικό διάστημα για το οποίο επιθυμεί να δει δεδομένα και το χαρακτηριστικό που επιθυμεί από τα διαθέσιμα που είναι τα εξής: φως, τάση ρεύματος ενεργειακής πηγής, θερμοκρασία, υγρασία, μαγνήτιση στους άξονες X και Y και επιτάχυνση στους άξονες X και Y (light, voltage, temperature, humidity, magnitude X / Y, acceleration X / Y)

4.1 Καρτέλα πρώτη: ένας κόμβος – ένα χαρακτηριστικό

Σε αυτή τη καρτέλα, όπως φαίνεται και από την *Εικόνα 17*, ο χρήστης καλείται να επιλέξει τον κόμβο, το χαρακτηριστικό, καθώς και την χρονική περίοδο που επιθυμεί. Κατόπιν επιλέγει “Show Graph” και εμφανίζεται το γράφημα των τιμών του χαρακτηριστικού για τον συγκεκριμένο κόμβο και για το χρονικό διάστημα που έχει επιλέξει (*Εικόνα 18*).



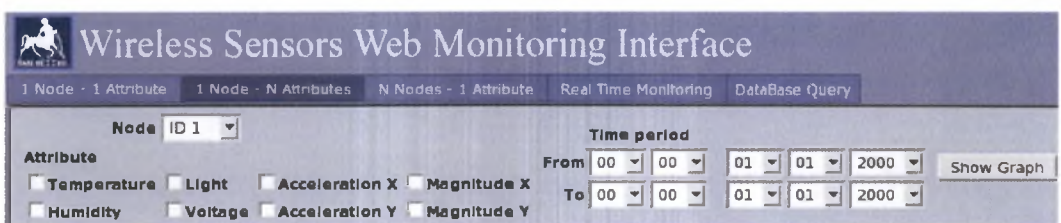
Εικόνα 17 : Η πρώτη καρτέλα



Εικόνα 18 : Δείγμα γραφήματος που εμφανίζεται στην πρώτη καρτέλα

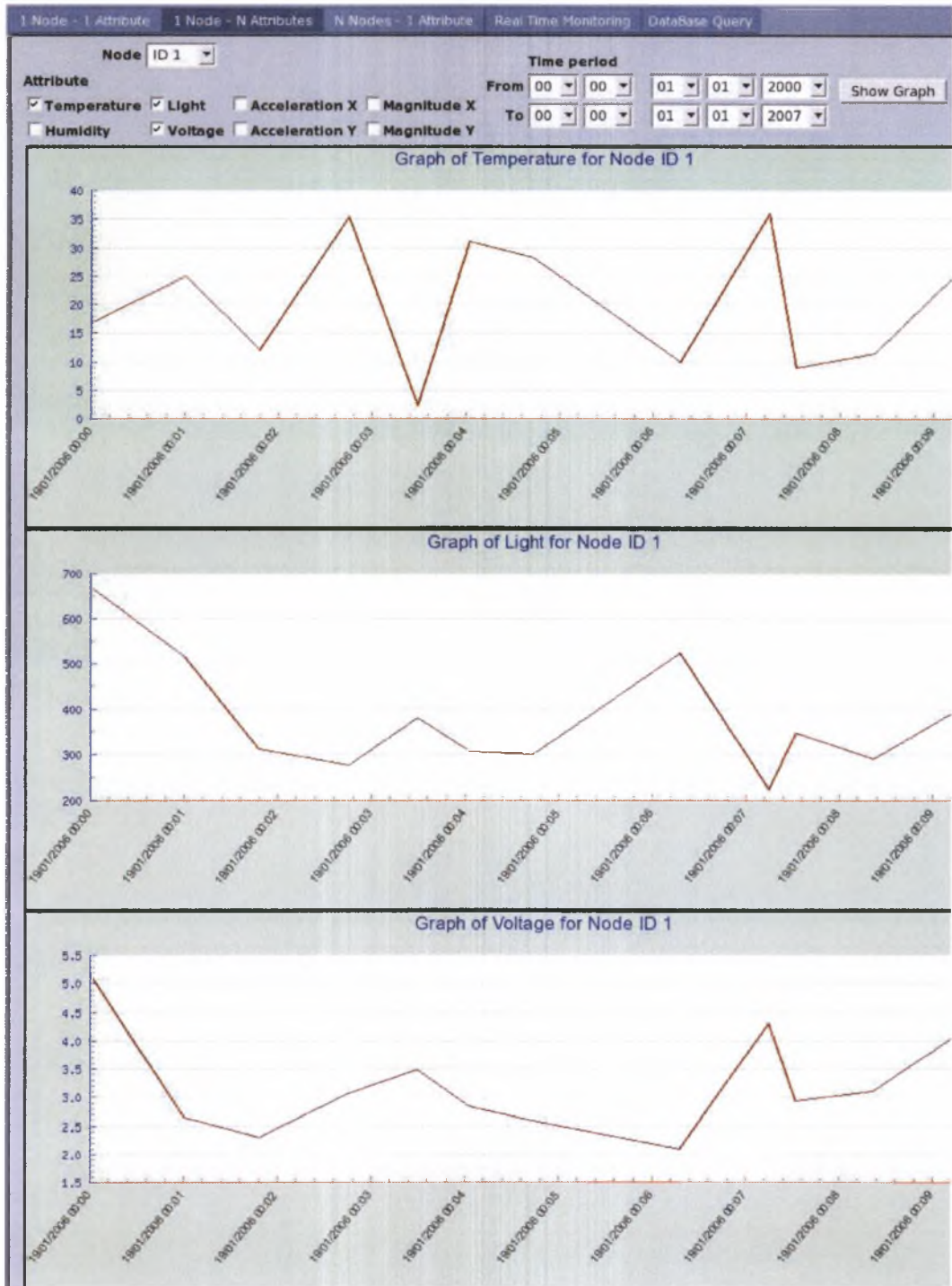
4.2 Καρτέλα δεύτερη: ένας κόμβος – πολλά χαρακτηριστικά

Η δεύτερη καρτέλα, όπως φαίνεται και από την *Εικόνα 19*, είναι παρόμοια οπτικά και λειτουργικά με την πρώτη καρτέλα, εκτός από το γεγονός ότι ο χρήστης μπορεί να επιλέξει περισσότερα από ένα χαρακτηριστικά ταυτόχρονα και να εμφανιστεί ξεχωριστό γράφημα για κάθε ένα από αυτό σε μια σελίδα (*Εικόνα 20*).



Εικόνα 19 : Η δεύτερη καρτέλα

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων



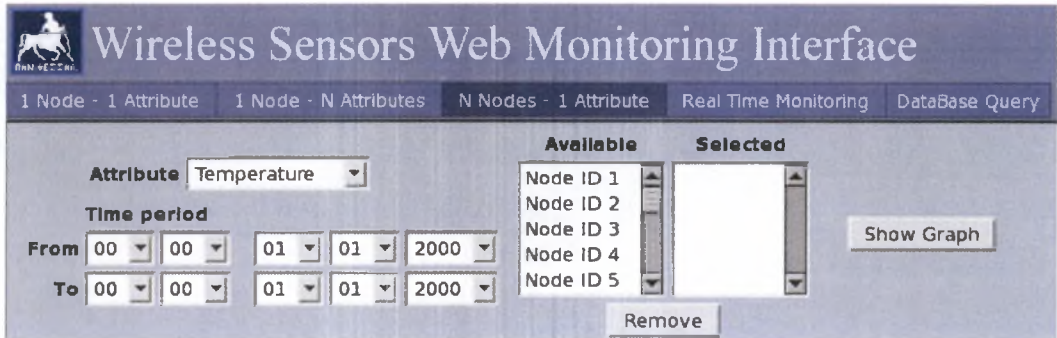
Εικόνα 20 : Έχοντας επιλέξει πολλά χαρακτηριστικά

4.3 Καρτέλα τρίτη: πολλοί κόμβοι – ένα χαρακτηριστικό

Σε αυτή την καρτέλα ο χρήστης μπορεί να επιλέξει κάποιο χαρακτηριστικό και παραπάνω από έναν κόμβο αισθητήρα και να δει τις τιμές των επιλεγμένων κόμβων για το συγκεκριμένο χαρακτηριστικό σε ένα γράφημα. Έτσι, μπορεί εύκολα να

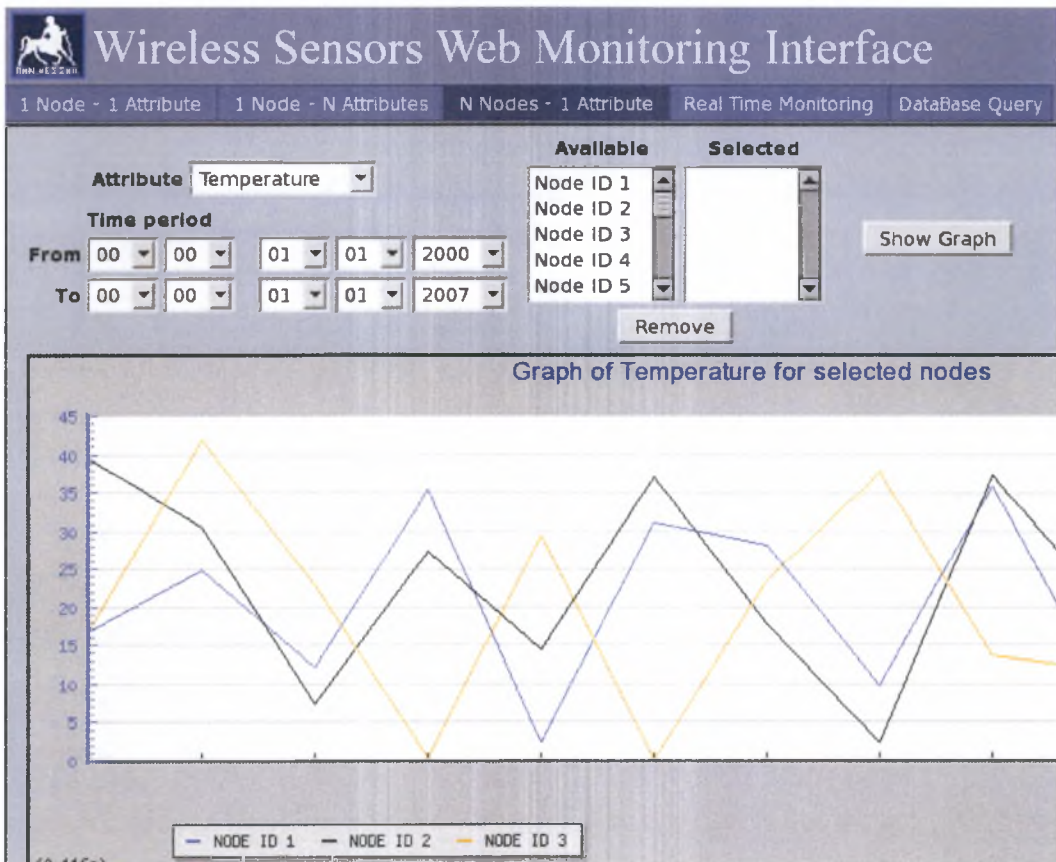
Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

συγκρίνει τις τιμές από τους διάφορους κόμβους. Ο τρόπος εμφάνισης και η λειτουργία της τρίτης καρτέλας παρουσιάζεται στις **Εικόνες 21** και **22**.



Εικόνα 21 : Η τρίτη καρτέλα

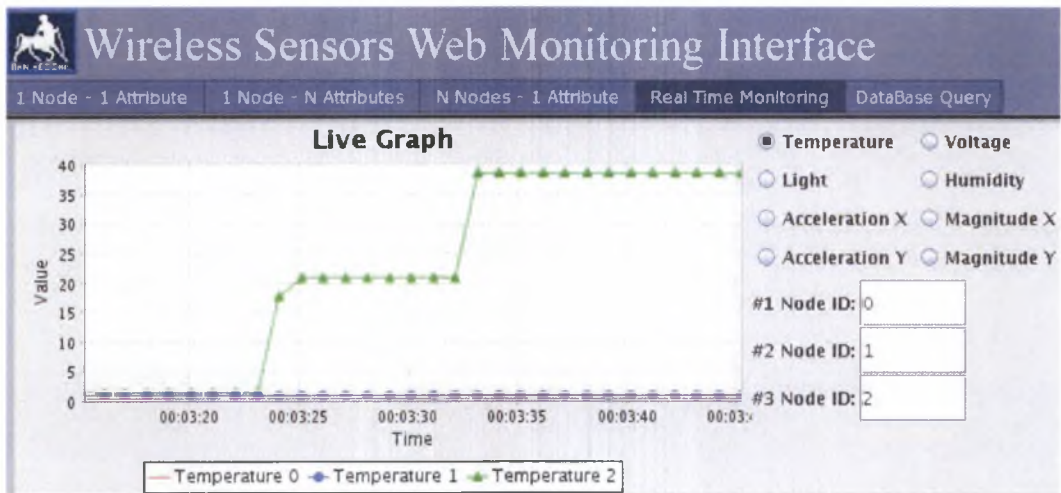
Το πλήθος των κόμβων που μπορεί να επιλέξει ο χρήστης είναι δεκαπέντε αλλά επειδή το γράφημα γίνεται όλο και πιο πυκνό όσο αυξάνεται ο αριθμός των κόμβων συνιστάται να επιλέγονται μέχρι πέντε κόμβοι.



Εικόνα 22 : Δείγμα γραφήματος της τρίτης καρτέλας

4.4 Καρτέλα τέταρτη: Γράφημα Πραγματικού Χρόνου

Σε αυτή τη καρτέλα ο χρήστης μπορεί με χρήση της τεχνολογίας των μικροεφαρμογών Java (Java Applets) να βλέπει τα δεδομένα που αποθηκεύονται στην βάση δεδομένων σε πραγματικό χρόνο. Αρχικά απαιτείται κάποιος χρόνος για να φορτώσει η μικροεφαρμογή και στην συνέχεια εμφανίζεται ένα γράφημα που εμφανίζει δεδομένα για τρεις κόμβους, όπως παρουσιάζεται στην *Εικόνα 23*. Ο χρήστης έχει την δυνατότητα να επιλέξει ένα από τα χαρακτηριστικά και να αλλάξει τους κόμβους για τους οποίους θα βλέπει δεδομένα. Το γράφημα πραγματικού χρόνου έχει την δυνατότητα να απεικονίζει δεδομένα από τρεις διαφορετικούς κόμβους ταυτόχρονα.



Εικόνα 23 : Το γράφημα πραγματικού χρόνου της τέταρτης καρτέλας

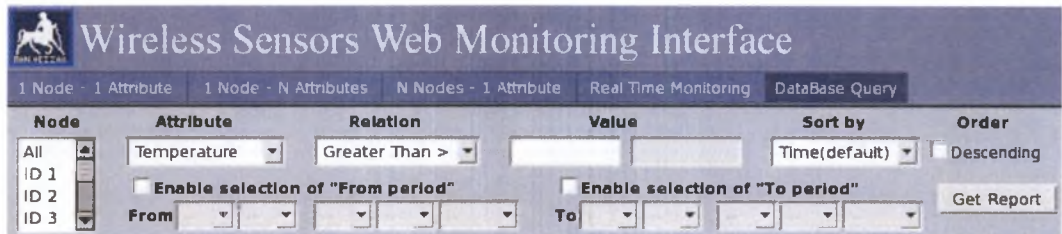
4.5 Καρτέλα πέμπτη: Ερωτήσεις στην βάση δεδομένων

Σε αυτή την καρτέλα ο χρήστης μπορεί να επιλέξει τον κόμβο, το χαρακτηριστικό, την σχέση, την τιμή, το χρονικό διάστημα καθώς και τον τρόπο που θα εμφανιστούν τα δεδομένα, να θέσει ένα ερώτημα στην βάση με τις τιμές που έχει καθορίσει και να ανακτήσει όλες τις εγγραφές που υπάρχουν στην βάση και ταιριάζουν στα κριτήρια που έχει θέσει. Αναλυτικότερα:

Αρχικά, ο χρήστης μπορεί να επιλέξει έναν, πολλούς ή όλους τους κόμβους για να εμφανίσει δεδομένα για αυτούς. Στη συνέχεια επιλέγει το χαρακτηριστικό, την σχέση που θα έχει αυτό (μεγαλύτερο, μικρότερο ή μεταξύ) και την τιμή του. Στην περίπτωση που επιλέξει να αναζητήσει εγγραφές μεταξύ κάποιων τιμών τότε μπορεί να εισάγει τιμή και στα δυο πεδία της φόρμας, διαφορετικά μόνο στο ένα. Κατόπιν,

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

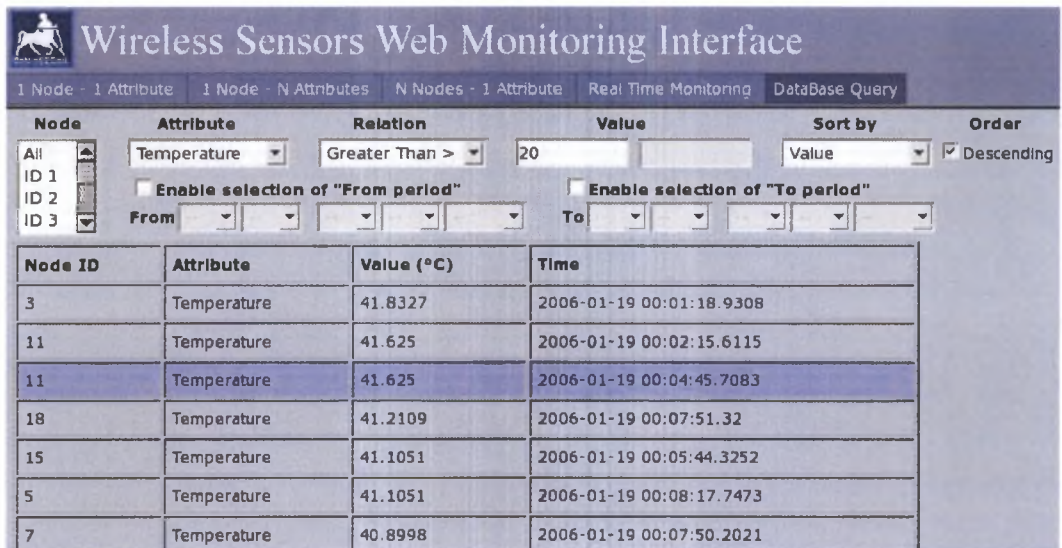
επιλέγει αν θέλει την χρονική περίοδο έτσι ώστε να ανακτήσει δεδομένα από την πρώτη εγγραφή μέχρι μια ορισμένη ημερομηνία, από μια ορισμένη ημερομηνία μέχρι την πιο πρόσφατη εγγραφή ή μεταξύ δύο ορισμένων ημερομηνιών. Τέλος, επιλέγει τον τρόπο που θέλει να ταξινομηθούν τα δεδομένα (ανά κόμβο, ανά χρονική στιγμή ή ανά τιμή) και αν αυτή η ταξινόμηση θα είναι φθίνουσα (η εξ ορισμού ταξινόμηση είναι αύξουσα). Τα παραπάνω φαίνονται στην **Εικόνα 24**.



The screenshot shows the 'Wireless Sensors Web Monitoring Interface' with a navigation bar containing '1 Node - 1 Attribute', '1 Node - N Attributes', 'N Nodes - 1 Attribute', 'Real Time Monitoring', and 'DataBase Query'. The main form has several sections: 'Node' with a list (All, ID 1, ID 2, ID 3), 'Attribute' set to 'Temperature', 'Relation' set to 'Greater Than >', 'Value' with two empty input fields, 'Sort by' set to 'Time(default)', and 'Order' set to 'Descending'. There are checkboxes for 'Enable selection of "From period"' and 'Enable selection of "To period"', and a 'Get Report' button.

Εικόνα 24 : Η καρτέλα ερωτήσεων προς την βάση

Πατώντας στο κουμπί “Get Report” εμφανίζεται ένας πίνακας με όλες τις εγγραφές που ταιριάζουν με τα κριτήρια αναζήτησης που έθεσε ο χρήστης. Χρησιμοποιώντας την λειτουργία εκτύπωσης του προγράμματος περιήγησης μπορεί να γίνει εκτύπωση του πίνακα ενώ η γραμμή της εγγραφής στην οποία μετακινείται ο δείκτης του ποντικιού τονίζεται με διαφορετικό χρώμα όπως φαίνεται στην **Εικόνα 25**.



The screenshot shows the same interface as Figure 24, but with the 'Get Report' button clicked. The 'Value' field now contains '20'. Below the form, a table displays the results. The table has four columns: 'Node ID', 'Attribute', 'Value (°C)', and 'Time'. The table contains 7 rows of data. The second row (Node ID 11) is highlighted in blue, indicating it is the current selection.

Node ID	Attribute	Value (°C)	Time
3	Temperature	41.8327	2006-01-19 00:01:18.9308
11	Temperature	41.625	2006-01-19 00:02:15.6115
11	Temperature	41.625	2006-01-19 00:04:45.7083
18	Temperature	41.2109	2006-01-19 00:07:51.32
15	Temperature	41.1051	2006-01-19 00:05:44.3252
5	Temperature	41.1051	2006-01-19 00:08:17.7473
7	Temperature	40.8998	2006-01-19 00:07:50.2021

Εικόνα 25 : Δείγμα του πίνακα της τελευταίας καρτέλας με δεδομένα

5 Προβλήματα, Προοπτικές, Συμπεράσματα

5.1 Προβλήματα

Ένα από τα αρχικά προβλήματα που προέκυψαν κατά τη διαδικασία ανάπτυξης της εφαρμογής αφορούσε την εγκατάσταση της βιβλιοθήκης παραγωγής γραφημάτων JpGraph [31]. Στην περίπτωση που το σύστημα στο οποίο εγκαθίσταται η εφαρμογή χρησιμοποιεί λειτουργικό σύστημα Linux η βιβλιοθήκη εσφαλμένα αναζητά γραμματοσειρές του συστήματος σε διαδρομή που χρησιμοποιούνταν παλαιότερα. Η λύση ήταν η δημιουργία συνδέσμων (softlinks) μεταξύ της καινούριας θέσης όπου το σύστημα έχει εγκατεστημένες τις γραμματοσειρές και της παλαιάς. Το πρόβλημα λογικά θα λυθεί σε κάποια νεότερη έκδοση της βιβλιοθήκης.

Ένα άλλο πρόβλημα με την συγκεκριμένη βιβλιοθήκη είναι ότι στο γράφημα ενός χαρακτηριστικού πολλών κόμβων δεν υπάρχει δυνατότητα εμφάνισης ημερομηνίας και χρόνου στο κάτω μέρος του γραφήματος. Αυτό οφείλεται εν μέρει σε αδυναμία χειρισμού και συνδυασμού των δεδομένων ανά χρονική στιγμή από την βιβλιοθήκη και εν μέρει στην φύση των δεδομένων καθώς δυο εγγραφές στην βάση δεδομένων δεν μπορούν να έχουν ίδια τιμή στο πεδίο χρονικής περιόδου (timestamp).

Πρόβλημα προέκυψε και αντιμετωπίστηκε και στην ρύθμιση της βάσης δεδομένων για απομακρυσμένη σύνδεση έτσι ώστε να μπορεί να ανακτά δεδομένα από αυτήν ένας εξυπηρετητής ιστού που δεν βρίσκεται στο ίδιο υπολογιστικό σύστημα. Το πρόβλημα λύθηκε με την μετατροπή των κατάλληλων ρυθμίσεων.

Παρόμοιο πρόβλημα αντιμετωπίστηκε επιτυχώς με την εύρεση των κατάλληλων ρυθμίσεων των επιλογών ασφάλειας του περιβάλλοντος εκτέλεσης της Java (Java Runtime Environment) καθώς οι εξ' ορισμού ρυθμίσεις είναι πολύ αυστηρές και δεν επέτρεπαν στο Java applet να συνδεθεί στην βάση δεδομένων για να ανακτήσει δεδομένα για την απεικόνιση του γραφήματος πραγματικού χρόνου.

Όπως αναφέρθηκε, για την δημιουργία του applet για την απεικόνιση γραφήματος πραγματικού χρόνου χρησιμοποιήθηκε η βιβλιοθήκη JfreeChart [33] η οποία στην ουσία δεν υποστηρίζει δυνατότητα γραφήματος πραγματικού χρόνου αλλά αυτόματης ανανέωσης του γραφήματος σε περίπτωση που τα δεδομένα αλλάζουν. Έτσι, αν τα δεδομένα αποθηκεύονται στη βάση δεδομένων με πάρα πολύ γρήγορους ρυθμούς (της τάξης του δέκατου του δευτερολέπτου) ενδέχεται να μην εμφανιστεί κάποια τιμή στο γράφημα. Ένας τόσο γρήγορος ρυθμός αποθήκευσης δεδομένων, όμως, από τρεις

συγκεκριμένους κόμβους (ο αριθμός των κόμβων που εμφανίζονται ταυτόχρονα στο γράφημα) είναι πολύ απίθανος.

Τέλος, επειδή το applet για την απεικόνιση πραγματικού χρόνου ενδέχεται να ανακτά δεδομένα από έναν εξυπηρετητή βάσης δεδομένων που μπορεί να βρίσκεται σε αρκετή απόσταση και με πιθανή χαμηλή σύνδεση, τα δεδομένα ίσως να έχουν μια καθυστέρηση από την στιγμή που αποθηκεύτηκαν στη βάση μέχρι την στιγμή που θα εμφανιστούν στο γράφημα της τάξης λίγων δευτερολέπτων το οποίο, όμως, θεωρείται ανεκτό σφάλμα για την συγκεκριμένη εφαρμογή.

5.2 Προοπτικές – Δυνατότητες Επέκτασης

Οι προοπτικές που έχει το σύστημα καταγραφής δεδομένων που υλοποιήθηκε είναι πολλές καθώς μπορεί να βρει πρακτική εφαρμογή σε αρκετούς τομείς, όπως αυτοί αναφέρθηκαν παραπάνω (υποκεφάλαιο 2.2). Θεωρητικά μπορεί να χρησιμοποιηθεί, όπου χρησιμοποιούνται αισθητήρες που μετράνε μεγέθη σαν αυτά που μπορεί και χειρίζεται η εφαρμογή (φως, θερμοκρασία, επιτάχυνση, υγρασία, μαγνητισμό, επιτάχυνση). Ενδεικτικά αναφέρονται εφαρμογές ελέγχου περιβαλλοντικών συνθηκών, παρατήρησης καιρού ή εφαρμογή στην γεωργία ακριβείας.

Το σύστημα που αναπτύχθηκε, παρόλες τις δυνατότητες που ενσωματώνει, δεν παύει να έχει περιθώρια βελτίωσης και επέκτασης. Οι παρακάτω προτάσεις για βελτίωση, παρόλο που δεν στάθηκε δυνατόν να υλοποιηθούν λόγω κυρίως έλλειψης χρόνου, είναι αρκετά σημαντικές:

- Συνδυασμός του συστήματος που υλοποιήθηκε με κάποιο υπάρχον σύστημα GIS (Geographic Information System) ή υλοποίηση δυνατοτήτων ενός GIS συστήματος. Με αυτόν τον τρόπο ο χρήστης θα μπορεί να έχει και μια χωρική απεικόνιση των κόμβων και των δεδομένων τους.
- Υλοποίηση λειτουργίας “συναγερμών”, δηλαδή ύπαρξης μηχανισμού ο οποίος μέσα από κάποια διεπαφή θα δημιουργεί λογικές εκφράσεις του τύπου “Εάν η θερμοκρασία ξεπεράσει τους 70 βαθμούς Κελσίου” και με βάση αυτές θα εκτελεί κάποια λειτουργία όταν μία, μερικές ή όλες γίνουν αληθείς, όπως, η αποστολή κάποιου ηλεκτρονικού μηνύματος ή SMS στον χρήστη ή η εμφάνιση κάποιου μηνύματος στην οθόνη.
- Υλοποίηση δυνατότητας ασύρματου προγραμματισμού των κόμβων (over the air programming) η οποία θα καθιστούσε δυνατό τον επαναπρογραμματισμό

του λογισμικού κάποιου κόμβου χωρίς την ανάκτηση του από το σημείο που έχει τοποθετηθεί.

5.3 Η αυτόνομη εφαρμογή

Η φύση της εφαρμογής που υλοποιήθηκε (διαδικτυακή - web based) καθιστά δύσκολη την υλοποίηση των παραπάνω επεκτάσεων. Για τον λόγο αυτό, καθώς και για τα πλεονεκτήματα που προσφέρει μια αυτόνομη εφαρμογή (περισσότερες δυνατότητες υλοποίησης, επεξεργαστική ισχύ κλπ) στα πλαίσια της παρούσας εργασίας ξεκίνησε και η δημιουργία μιας αυτόνομης εφαρμογής η οποία ακόμη βρίσκεται στο στάδιο της ανάπτυξης και θα μπορεί με πιο εύκολο τρόπο να ενσωματώνει τις λειτουργίες της διαδικτυακής εφαρμογής που αναπτύχθηκε καθώς και επεκτάσεις αυτών. Σε μια αυτόνομη εφαρμογή μπορεί κάποιος χρησιμοποιώντας γλώσσα προγραμματισμού υψηλού επιπέδου να ενσωματώσει πολύ περισσότερες λειτουργίες και δυνατότητες από ότι μπορεί να υλοποιήσει σε μια εφαρμογή διαδικτύου. Ενδεικτικά αναφέρονται: χρήση διαφορετικών ειδών αισθητήρων μέσω επιλογής του τύπου των αισθητήρων, χωρική απεικόνιση των κόμβων με χρήση συστήματος GIS, ασύρματο προγραμματισμό αισθητήρων, ύπαρξη λειτουργίας “συναγερμών”, εξαγωγή στατιστικών στοιχείων για τα δεδομένα και απεικόνισή τους με διάφορους τρόπους, ύπαρξη βοήθειας υπό μορφή “wizard” που θα καθοδηγεί βήμα – βήμα τον χρήστη για όποια λειτουργία θέλει να εκτελέσει, απεικόνιση των συνδέσεων των κόμβων – αισθητήρων μεταξύ τους, γραφική διεπαφή με υποστήριξη για πάνω από μια γλώσσες. Στόχος της δημιουργίας της αυτόνομης εφαρμογής είναι η υλοποίηση των παραπάνω, και όχι μόνο, λειτουργιών.

5.4 Επίλογος

Τα ασύρματα δίκτυα αισθητήρων είναι μια σχετικά νέα αλλά πολλά υποσχόμενη τεχνολογία. Η ανάπτυξη και εξέλιξή της θα βοηθήσει τον άνθρωπο σε πολλούς τομείς είτε βελτιώνοντας την ζωή του είτε δίνοντας του πληροφορίες για το περιβάλλον του και βοηθώντας τον να το κατανοήσει και να το μελετήσει. Για την εποικοδομητική χρήση των δικτύων αισθητήρων, όμως, απαιτείται κάποιος αποδοτικός τρόπος καταγραφής, επεξεργασίας και απεικόνισης των δεδομένων που συλλέγονται. Η εφαρμογή που υλοποιήθηκε είναι μια προσέγγιση προς αυτή την κατεύθυνση, την εξαγωγή χρήσιμων συμπερασμάτων και πληροφοριών για τον κόσμο που μας περιβάλλει.

Αναφορές

- [1] C. Chong, S.P.Kumar “Sensor Networks: Evolution, Opportunities, and Challenges” *Proceedings of the IEEE*, vol. 91, no. 8, August 2003.
- [2] Estrin, D.; Govindan, R.; Heidemann, J. ; Kumar, S., “Next century challenges: Scalable coordination in sensor networks.”
MOBICOM, 1999, Seattle, pg 263-270.
- [3] C. Intanagonwiwat, R. Govindan, and D. Estrin, “Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks,”
Proc. ACM MobiCom '00, Boston, MA, 2000, pp. 56–67.
- [4] Rabaey, J.M.; Ammer, M.J.; da Silva, J.L., Jr.; Patel, D.; et al, “PicoRadio supports ad hoc ultra-low power wireless networking.”
Computer, July 2000, vol.33, (no.7):42-8.
- [5] Hill, J.; Szewczyk, R.; Woo, A.; Culler, D.; Hollar, S.; and Pister. K., “System architecture directions for networked sensors.”
ASPLOS, Nov. 2000, pg 93-104
- [6] Ye, W.; Heidemann, J.; and Estrin, D., “An Energy-Efficient MAC Protocol for Wireless Sensor Networks.”
INFOCOM 2002, New York, June, 2002.
- [7] Fuller, T.F.; Doyle, M.; Newman, J., “Simulation and optimization of the dual lithium ion insertion cell.”
Journal of the Electrochemical Society, Jan. 1994, vol.141, (no.1): 1-10.
- [8] Linden, D., “Handbook of batteries.” 2nd ed.
New York: McGraw-Hill, 1995.
- [9] Faroque, M.; Maru, H.C., “Fuel cells-the clean and efficient power generators.”
Proceedings of the IEEE, Dec. 2001, vol.89, (no.12):1819-29.
- [10] Meninger, S.; Mur-Miranda, J.O.; Amirtharajah, R.; Chandrakasan, A.; et al, “Vibration-to-electric energy conversion.”
IEEE Transactions on VLSI Systems, Feb. 2001, vol.9, (no.1): 64-76.
- [11] P.Kikiras, J.N.Avaritsiotis, “Unattended Ground Sensor Network for Force Protection” in *Journal of Battlefield Technology*, Vol 7. No 3 November 2004.
- [12] The SOSUS Project : Available on line at
(<http://www.globalsecurity.org/intell/systems/sosus.htm>)

- [13] The COBRA BRASS Project: *Available on line at*
(<http://www.globalsecurity.org/space/systems/cobrabrass.htm>)
- [14] S. Kumar and D. Shepherd, “SensIT: Sensor information technology for the warfighter,” in *Proc. 4th Int. Conf. on Information Fusion, 2001*
- [15] D. Steere, A. Baptista, D. McNamee, C. Pu, and J. Walpole, “Research challenges in environmental observation and forecasting systems,” in *Proc. 6th Int. Conf. Mobile Computing and Networking (MOBICOMM), 2000*, pp. 292–299.
- [16] C. E. Nishimura and D. M. Conlon, “IUSS dual use: Monitoring whales and earthquakes using SOSUS,” *Mar. Technol. Soc. J.*, vol. 27, no. 4, pp. 13–21, 1994.
- [17] Cerpa, A.; Elson, J.; Estrin, D.; Girod, L.; Hamilton M.; and Zhao, J., “Habitat monitoring: Application driver for wireless communications technology.” ACM SIGCOMM Workshop on Data Communications in Latin America and the Caribbean, Costa Rica, April 2001
- [18] (1995) The cooperative engagement capability.
[Online] <http://techdigest.jhuapl.edu/td1604/APLteam.pdf>
- [19] Y. Carts-Powell. (2000, Apr.) Unattended ground sensors stop and analyze the roses. *OE Rep.*
[Online] <http://www.spie.org/web/ocf/april/apr00/cover2.html>
- [20] C.Y. Chong, S. Mori, and K. C. Chang, “Distributed multitarget multisensor tracking,” in *Multitarget Multisensor Tracking: Advanced Applications*, Y. Bar-Shalom, Ed. Norwood, MA: Artech House, 1990, pp. 247–295.
- [21] R. Hills. (2001, July/Aug.) Sensing for danger. *Sci. Technol. Rep.*
[Online] <http://www.llnl.gov/str/JulAug01/Hills.html>
- [22] Mani Srivastava, Richard Muntz, and Miodrag Potkonjak, “Smart Kindergarten: Sensor-based Wireless Networks for Smart Developmental Problem-solving Environments,” in *The Seventh Annual International Conference on Mobile Computing and Networking 2001*, July 2001, pp.132 – 138.
- [23] K. Lee, “Wireless sensing and IEEE 1451,” presented at the Sensor Conf./Expo 2001, Chicago, IL.
- [24] Β. Βεσκούκης “Τεχνολογία Λογισμικού Ι” , ΕΑΠ, ΠΑΤΡΑ, 2000

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

Ιστοσελίδες:

- [25] <http://www.uml.org/>
- [26] <http://www.w3.org/MarkUp/>
- [27] <http://www.php.net/>
- [28] <http://java.sun.com/applets/>
- [29] <http://www.postgresql.org/>
- [30] <http://www.apache.org/>
- [31] <http://www.aditus.nu/jpgraph/>
- [32] <http://www.boutell.com/gd/>
- [33] <http://www.jfree.org/jfreechart/>
- [34] <http://java.sun.com/>
- [35] <http://www.mozilla.org/js/>
- [36] <http://www.w3.org/Style/CSS/>
- [37] <http://www.tinyos.net/>
- [38] http://www.xbow.com/wireless_home.aspx
- [39] <http://telegraph.cs.berkeley.edu/tinydb/>
- [40] <http://www.postgresql.org/docs/8.1/static/sql.html>

Παράρτημα

Ακολουθεί ο πηγαίος κώδικας των αρχείων της εφαρμογής.

sensorStyle.css

```
body {
  font-family: Verdana, Arial, Helvetica, sans-serif;
  font-size: 12px;
  font-weight: bold;
  margin: 0px;
  padding: 0px;
  background-color: #2d3851;
}

h1 {
  font-family: Times New Roman, serif;
  font-size: 32px;
  font-weight: normal;
  line-height: 0px;
  color: #e9f2fc;
}

h2 {
  font-family: Verdana, Arial, Helvetica, sans-serif;
  font-size: 12px;
  font-weight: bold;
  color: #333333;
}

p {
  font-family: Verdana, Arial, Helvetica, sans-serif;
  font-size: 12px;
  font-weight: normal;
  line-height: 18px;
  color: #333333;
}

a {
  font-family: Verdana, Arial, Helvetica, sans-serif;
  font-size: 12px;
  font-weight: normal;
  text-decoration: underline;
  color: #0000f0;
}

a:visited {
  text-decoration: underline;
  color: #000090;
}

a.navlink {
  font-family: Verdana, Arial, Helvetica, sans-serif;
  font-size: 12px;
  font-weight: normal;
  text-decoration: none;
  color: #e9f2fc;
}

a.navlink:hover {
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```
background-color: rgb(105, 75, 219);
}

a.navlink:visited {
text-decoration: none;
color: #e9f2fc;
}

td.navcell:hover {
background-color: rgb(105, 75, 219);
}

tr.resultRow:hover {
background-color: rgb(150, 170, 200);
}

li {
font-family: Verdana, Arial, Helvetica, sans-serif;
font-size: 12px;
font-weight: normal;
line-height: 18px;
padding-bottom: 7px;
color: #333333;
}

.navcell {
padding: 3px 1px 3px 1px;
}

.update {
font-family: Verdana, Arial, Helvetica, sans-serif;
font-size: 10px;
font-weight: normal;
line-height: 14px;
color: #e9f2fc;
}

.copyright {
font-family: Verdana, Arial, Helvetica, sans-serif;
font-size: 10px;
font-weight: normal;
line-height: 10px;
color: #464c64;
}
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

connect2db.php

```
<?
/*
 * connects to DataBase and defines the $dbcon
 */
if( !isset( $dbcon ) ) {
    global $dbcon;

    $dbcon=pg_pconnect("host=localhost port=5432 dbname=sensors
user=sensor password=sensor")
    or die('Could not connect to database. Check settings : ' .
pg_last_error($dbcon));
}
?>
```


Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

graph1.php

```
<?php
include ("./src/jpgraph.php");
include ("./src/jpgraph_line.php");
include ("tools.php");

$ydata = array();
$xdata = array();

$fDate = $_GET['fY'] . "-" . $_GET['fM'] . "-" . $_GET['fD'] . " " .
$_GET['fHour'] . ":" . $_GET['fMin'] . ":00";
$tDate = $_GET['tY'] . "-" . $_GET['tM'] . "-" . $_GET['tD'] . " " .
$_GET['tHour'] . ":" . $_GET['tMin'] . ":00";

$attr = strtolower(attrib2($_GET['attribute']));

$quer = "select * from " . TABLE . " where nodeid=" . $_GET['nodeid']
. " and result_time between '" . $fDate . "' and '" . $tDate . "'";
//TABLE is defined in tools.php

$result = pg_query($quer);
$row = pg_fetch_array($result);
array_push($ydata,$row[$attr]);
$startPlotDate = $row['result_time'];
$startTime = strtotime( $startPlotDate);
array_push($xdata,$startTime);

while($row = pg_fetch_array($result)){
    array_push($ydata,$row[$attr]);
    $tempDate=$row['result_time'];
    $tempTime=strtotime( $tempDate);
    array_push($xdata,$tempTime);
    $row1=$row;
}

$stopPlotDate = $row1['result_time'];
$stopTime = strtotime( $stopPlotDate);

function TimeCallback( $aVal) {
    return Date ('d/m/Y H:i', $aVal);
}

$adjstart=$startTime;
$adjend=$stopTime;
$dd = $stopTime - $startTime;
$dd = $dd / 10;
$graph = new Graph(960, 350);
$graph->SetMargin(60,30,40 ,100);

// Now specify the X-scale explicit but let the Y-scale be auto-
scaled
$graph->SetScale( "intlin",0,0 , $adjstart,$adjend);
//$graph->SetScale("textlin");
$graph->title->Set("Graph of " . attrib($_GET['attribute']) . " for
Node ID " . $_GET['nodeid'] );
$graph->title->SetColor('navy');
$graph->title->SetFont(FF_ARIAL,FS_NORMAL,13);
// Setup the callback and adjust the angle of the labels
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```
$graph->xaxis->SetPos('min');
$graph->xaxis-> SetFont(FF_ARIAL,FS_NORMAL,8);
$graph->xaxis-> SetLabelFormatCallback( 'TimeCallback');
$graph->xaxis-> SetLabelAngle(50);

// Set the labels every 5min (i.e. 300seconds) and minor ticks every
minute
$graph->xaxis-> scale->ticks->Set ($dd,$dd/5);

$line = new LinePlot($ydata,$xdata);
$line->SetWeight(2);
$line->SetColor( 'darkred');
$graph->Add( $line);
$graph->SetMarginColor('darkgreen@0.8');
$graph->SetColor('white');
$graph->yaxis->SetColor('blue','darkblue');
$graph->xaxis->SetColor('red','black');
$graph->Stroke();
?>
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

graph2.php

```
<?php
include ("./src/jpgraph.php");
include ("./src/jpgraph_line.php");
include ("tools.php");
$colors=array('blue','black','orange','red','yellow',
              'turquoise','green','violet','aqua','cyan',
              'magenta','hotpink','khaki','purple','sienna');

$nodesFromGET = $_GET['selNodes'];
$nodesFromGET = rtrim($nodesFromGET, ",");

$nodes = explode(",", $nodesFromGET);

foreach($nodes as $key=>$value){
    $ydata{(integer)$value}=array();
}

$fDate = $_GET['fY'] . "-" . $_GET['fM'] . "-" . $_GET['fD'] . " " .
$_GET['fHour'] . ":" . $_GET['fMin'] . ":00";
$tDate = $_GET['tY'] . "-" . $_GET['tM'] . "-" . $_GET['tD'] . " " .
$_GET['tHour'] . ":" . $_GET['tMin'] . ":00";

$attr = strtolower(attrib2($_GET['attribute']));

$query = "select * from " . TABLE . " where result_time between '" .
$fDate . "' and '" . $tDate . "' ";
//TABLE is defined in tools.php

$result = pg_query($query);

while($row = pg_fetch_array($result)){
    array_push($ydata{(integer)$row['nodeid']}, $row[$attr]);
    $rowl=$row;
}

$gJpgBrandTiming=true;

// Create the graph. These two calls are always required
$graph = new Graph(960, 350);
$graph->SetScale("textlin");

// Create the linear plot
foreach($nodes as $key=>$value){
    $lineplot{$value}=new LinePlot($ydata{$value});
    $lineplot{$value}->SetLegend("NODE ID ".$value);
    $lineplot{$value}->SetColor($colors[$key]);
    $graph->Add($lineplot{$value});
    // Add the plot to the graph
}

$graph->img->SetMargin(40,20,40,80);
$graph->title->Set("Graph of " . attrib($_GET['attribute']) . " for
selected nodes");
$graph->title->SetColor('navy');
$graph->title->SetFont(FF_ARIAL, FS_NORMAL, 13);
$graph->yaxis->title->SetFont(FF_FONT1, FS_BOLD);
$graph->xaxis->title->SetFont(FF_FONT1, FS_BOLD);
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```
$graph->xaxis->HideLabels();  
$graph->xaxis->HideTicks(true,false);  
$graph->xaxis->SetPos('min');  
  
$graph->SetColor('white');  
  
$graph->yaxis->SetColor("blue");  
$graph->yaxis->SetWeight(2);  
  
$graph->legend->SetPos(0.1,0.95,'left','bottom');  
$graph->legend->SetColumns(8);  
  
// Display the graph  
$graph->Stroke();  
?>
```


Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

index.php

```
<?
require_once ("tools.php");

isset($_GET['v']) ? ($v = $_GET['v']) : (isset($_POST['v']) ? $v =
$_POST['v'] : $v = VIEW1);

require_once ("header.php");

switch ($v){

    case VIEW1:
        require_once("view_tab1.php");
        break;

    case VIEW2:
        require_once("view_tab2.php");
        break;

    case VIEW3:
        require_once("view_tab3.php");
        break;

    case VIEWRT:
        require_once("view_tabRT.php");
        break;

    case VIEWQDB:
        require_once("view_tabQDB.php");
        break;
}

require_once ("footer.php");
?>
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

tools.php

```
<?

//open the database connection
require_once("connect2db.php");

define("TABLE", "q2");

define("DatFile", "../.../connect.dat");
$fileLine = file(DatFile);

define("SERVER_FROM_FILE", $fileLine[0]);
define("DATABASE_FROM_FILE", $fileLine[1]);
define("USER_FROM_FILE", $fileLine[2]);
define("PASS_FROM_FILE", $fileLine[3]);

//Returns the attributes' names
function attrib($attr){
    $attribs =
array("Temperature", "Light", "AccelerationX", "MagnitudeX", "Humidity", "
Voltage", "AccelerationY", "MagnitudeY");
    return ($attribs[$attr]);
}

//Returns the attributes' table fields names
function attrib2($attr){
    $attribs =
array("temp", "light", "accel_x", "mag_x", "humidity", "voltage", "accel_y"
, "mag_y");
    return ($attribs[$attr]);
}

//Returns the attributes' measurement units
function units($attr) {
    $units = array(" (B°C)", " (lux)", " (g)", "", "", " (volt)", "", "");
    return ($units[$attr]);
}

function show_array($array) {
    echo('?');
    foreach ($array as $key => $value) {
        echo($key);
        echo('=');
        echo($value);
        echo('&');
    }
}
?>
```


Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

view_tab1.php

```
<!-- Navigation -->
<tr>
  <td align="left">
    <table cellspacing="1" cellpadding="2" border="0"
bgcolor="#354463" width="70%">
      <tr>
        <td bgcolor="#354463" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEW1">1 Node - 1
Attribute</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEW2">1 Node - N
Attributes</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEW3">N Nodes - 1
Attribute</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEWRT">Real Time
Monitoring</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEWQDB">DataBase Query</a></td>
      <!-- For extra navigation tabs uncomment below and maybe
change width above
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="pasilda.html">Links</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="pasilda.html">Contact</a></td>
      -->
    </tr>
  </table>
</td>
</tr>

<!-- Page Middle -->
<tr>
  <td>
    <table cellspacing="1" cellpadding="0" border="1"
bgcolor="#354463" width="100%">
      <tr>
        <td bgcolor="#bfc4cb" style="padding-left: 5px">
          <!-- FORM BEGIN-->
          <form name="input" action="index.php"
method="post">
            <table> <!-- Tab 1 main table -->
              <tr>
                <td>
                  <table> <!-- Node and attribute table -->
                    <tr>
                      <td align="right">Node </td>
                      <?
                        $result=pg_query("select distinct nodeid from "
. TABLE . " order by nodeid asc");
                        //TABLE is defined in tools.php
                        ?>
                    <td>
                      <select name="nodeid">
                        <?
                          while($row = pg fetch array($result)){
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

```

?>
                                                                    <option
value="<?=$row['nodeid']?>"
<?if(isset($_POST['nodeid'])&&$_POST['nodeid']==$row['nodeid']){print
"selected";}?>>ID <?=$row['nodeid']?></option>
                                                                    <?}?>
                                                                    </select>
    </td>
</tr>
<tr>
    <td align="right">Attribute</td>
    <td>
        <select name="attribute">
                                                                    <? $attribs =
array("Temperature"=>"0",
                                                                    "Light"=>"1",
                                                                    "Acceleration X"=>"2",
                                                                    "Magnitude X"=>"3",
                                                                    "Humidity"=>"4",
                                                                    "Voltage"=>"5",
                                                                    "Acceleration Y"=>"6",
                                                                    "Magnitude Y"=>"7");
                                                                    foreach
($attribs as $key => $value) {
                                                                    ?>
                                                                    <option value="<?=$value?>"
<?if(isset($_POST['attribute'])&&$_POST['attribute']==$value){print
"selected";}?>><?=$key?></option><?}?>
                                                                    </select>
                                                                    </td>
                                                                    </tr>
                                                                    </table> <!-- Node and attribute table -->
</td>
<td>
    <table> <!-- Time period table -->
    <tr>
        <td></td>
        <td colspan="6">Time period</td>
    </tr>
    <tr>
        <td align="right">From</td>
        <td> <!-- hour -->
                                                                    <select name="fHour">
                                                                    <?
                                                                    for($i=0;$i<24;$i++){
                                                                    if($i<10){?>
                                                                    <option
value="<?=$i?>"<?if(isset($_POST['fHour'])&&$_POST['fHour']==$i){prin
t "selected";}?>>0<?=$i?></option>
                                                                    <?}
                                                                    else
                                                                    <option
value="<?=$i?>"<?if(isset($_POST['fHour'])&&$_POST['fHour']==$i){prin
t "selected";}?>><?=$i?></option>
                                                                    <?}
                                                                    else{?>
                                                                    <option

```


Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

```

<select name="fM">
    <?
    for($i=1;$i<13;$i++){
        if($i<10){?>
            <option
value="<?=$i?>"<?if(isset($_POST['fM'])&&$_POST['fM']==$i){print
"selected";}?>>0<?=$i?></option>
            <?>
            else{?>
                <option
value="<?=$i?>"<?if(isset($_POST['fM'])&&$_POST['fM']==$i){print
"selected";}?>><?=$i?></option>
                <?>
            }
        }?>
    }>
</td> <!-- month -->
<td> <!-- year -->
    <select name="fY">
        <?
        for($i=2000;$i<2008;$i++){?>
            <option
value="<?=$i?>"<?if(isset($_POST['fY'])&&$_POST['fY']==$i){print
"selected";}?>><?=$i?></option>
            <?}?>
        }>
    }>
</td> <!-- year -->
</tr>
<tr>
    <td align="right">To</td>
    <td> <!-- hour -->
        <select name="tHour">
            <?
            for($i=0;$i<24;$i++){
                if($i<10){?>
                    <option
value="<?=$i?>"<?if(isset($_POST['tHour'])&&$_POST['tHour']==$i){prin
t "selected";}?>>0<?=$i?></option>
                    <?>
                    else
                if($i>=10&&$i<24){?>
                    <option
value="<?=$i?>"<?if(isset($_POST['tHour'])&&$_POST['tHour']==$i){prin
t "selected";}?>><?=$i?></option>
                    <?>
                    else{?>
                        <option
value="<?=$i?>"<?if(isset($_POST['tHour'])&&$_POST['tHour']==$i){prin
t "selected";}?>>00</option>
                        <?>
                    }
                }?>
            }>
        }>
    }>
</select>

```


Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

view_tab2.php

```
<SCRIPT LANGUAGE="JavaScript">
<!-- Begin
function checkSelection() {
    if (!document.input.temp.checked &&
        !document.input.light.checked &&
        !document.input.accelX.checked &&
        !document.input.magnX.checked &&
        !document.input.humid.checked &&
        !document.input.volt.checked &&
        !document.input.accelY.checked &&
        !document.input.magnY.checked) {

        alert("Please select at least one attribute!");
    }
}
// End -->
</script>

<!-- Navigation -->
<tr>
<td align="left">
    <table cellspacing="1" cellpadding="2" border="0"
bgcolor="#354463" width="70%">
    <tr>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEW1">1 Node - 1
Attribute</a></td>
        <td bgcolor="#354463" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEW2">1 Node - N
Attributes</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEW3">N Nodes - 1
Attribute</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEWRT">Real Time
Monitoring</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEWQDB">DataBase Query</a></td>
        <!-- For extra navigation tabs uncomment below and maybe
change width above
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="pasilda.html">Links</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="pasilda.html">Contact</a></td>
        -->
    </tr>
    </table>
</td>
</tr>

<!-- Page Middle -->
<tr>
<td>
    <table cellspacing="1" cellpadding="0" border="1"
bgcolor="#354463" width="100%">
    <tr>
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```
<td bgcolor="#bfc4cb" style="padding-left: 5px">
<!-- FORM BEGIN-->
    <form name="input" action="index.php"
method="post">
        <table> <!-- Tab 2 main table -->
            <tr>
                <td>
                    <table>
                        <tr>
                            <td align="right">Node </td>
                            <?
                                $result=pg_query("select distinct nodeid from " .
TABLE . " order by nodeid asc");
                                //TABLE is defined in tools.php
                                ?>
                            <td colspan="3">
                                <select name="nodeid">
                                    <?
                                        while($row = pg_fetch_array($result)){
                                            ?>
                                                <option value="<?=$row['nodeid']?>"
<?if(isset($_POST['nodeid'])&&$POST['nodeid']==$row['nodeid']){print
"selected";}?>>ID <?=$row['nodeid']?></option>
                                            <?}
                                            ?>
                                        </select>
                                    </td>
                                </tr>
                                <tr>
                                    <td colspan="4">Attribute</td>
                                </tr>
                                <tr>
                                    <td><input type="checkbox" name="temp"
value="1"<?if(isset($_POST['temp'])) {print
"checked";}?>>Temperature</td>
                                    <td><input type="checkbox" name="light"
value="1"<?if(isset($_POST['light'])) {print "checked";}?>>Light</td>
                                    <td><input type="checkbox" name="accelX"
value="1"<?if(isset($_POST['accelX'])) {print
"checked";}?>>Acceleration X</td>
                                    <td><input type="checkbox" name="magnX"
value="1"<?if(isset($_POST['magnX'])) {print "checked";}?>>Magnitude
X</td>
                                </tr>
                                <tr>
                                    <td><input type="checkbox" name="humid"
value="1"<?if(isset($_POST['humid'])) {print
"checked";}?>>Humidity</td>
                                    <td><input type="checkbox" name="volt"
value="1"<?if(isset($_POST['volt'])) {print
"checked";}?>>Voltage</td>
                                    <td><input type="checkbox" name="accely"
value="1"<?if(isset($_POST['accely'])) {print
"checked";}?>>Acceleration Y</td>
                                    <td><input type="checkbox" name="magnY"
value="1"<?if(isset($_POST['magnY'])) {print "checked";}?>>Magnitude
Y</td>
                                </tr>
                            </table>
                        </td>
                    </tr>
                </table>
            </td>
        </table>
    </td>
</pre>
```


Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```

<table> <!-- Time period table -->
  <tr>
    <td></td>
    <td colspan="6">Time period</td>
  </tr>
  <tr>
    <td align="right">From</td>
    <td> <!-- hour -->
      <select name="fHour">
        <?
          for($i=0;$i<24;$i++){
            if($i<10){?>
              <option
value="<?=$i?"<?if(isset($_POST['fHour'])&&$_POST['fHour']==$i){prin
t "selected";}?>>0<?=$i?"</option>
              <?>
            else
if($i>=10&&$i<24){?>
              <option
value="<?=$i?"<?if(isset($_POST['fHour'])&&$_POST['fHour']==$i){prin
t "selected";}?>><?=$i?"</option>
              <?>
            else{?>
              <option
value="<?=$i?"<?if(isset($_POST['fHour'])&&$_POST['fHour']==$i){prin
t "selected";}?>>00</option>
              <?>
            }
          }
        ?>
      </select>
    </td> <!-- hour -->
    <td> <!-- minute -->
      <select name="fMin">
        <?
          for($i=0;$i<60;$i++){
            if($i==0){?>
              <option
value="<?=$i?"<?if(isset($_POST['fMin'])&&$_POST['fMin']==$i){print
"selected";}?>>00</option>
              <?>
            else
if($i<10){?>
              <option
value="<?=$i?"<?if(isset($_POST['fMin'])&&$_POST['fMin']==$i){print
"selected";}?>>0<?=$i?"</option>
              <?>
            else{?>
              <option
value="<?=$i?"<?if(isset($_POST['fMin'])&&$_POST['fMin']==$i){print
"selected";}?>><?=$i?"</option>
              <?>
            }
          }
        ?>
      </select>
    </td> <!-- minute -->

```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```

<td>&nbsp;&nbsp;&nbsp;</td>
<td> <!-- day -->
<select
name="fD">
<?
for($i=1;$i<32;$i++){
if($i<10){?>
<option
value="<?=$i?>"<?if(isset($_POST['fD'])&&$_POST['fD']==$i){print
"selected";}?>><?=$i?></option>
<?>
else{?>
<option
value="<?=$i?>"<?if(isset($_POST['fD'])&&$_POST['fD']==$i){print
"selected";}?>><?=$i?></option>
<?>
}
?>
</select>
</td> <!-- day -->
<td> <!-- month -->
<select name="fM">
<?
for($i=1;$i<13;$i++){
if($i<10){?>
<option
value="<?=$i?>"<?if(isset($_POST['fM'])&&$_POST['fM']==$i){print
"selected";}?>><?=$i?></option>
<?>
else{?>
<option
value="<?=$i?>"<?if(isset($_POST['fM'])&&$_POST['fM']==$i){print
"selected";}?>><?=$i?></option>
<?>
}
?>
</select>
</td> <!-- month -->
<td> <!-- year -->
<select name="fY">
<?
for($i=2000;$i<2008;$i++){?>
<option
value="<?=$i?>"<?if(isset($_POST['fY'])&&$_POST['fY']==$i){print
"selected";}?>><?=$i?></option>
<?}?>
</select>
</td> <!-- year -->
</tr>
<tr>
<td align="right">To</td>
<td> <!-- hour -->
<select name="tHour">
<?

```


Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```

for($i=1;$i<32;$i++){
    if($i<10){?}

    <option
value="<?=$i?>"<?if(isset($_POST['tD'])&&$_POST['tD']==$i){print
"selected";}?>>0<?=$i?></option>
                                                                    <?}
                                                                    else{?>

    <option
value="<?=$i?>"<?if(isset($_POST['tD'])&&$_POST['tD']==$i){print
"selected";}?>><?=$i?></option>
                                                                    <?}

                                                                    }
                                                                    ?>
                                                                </select>

</td> <!-- day -->
<td> <!-- month -->
    <select name="tM">
                                                                    <?

for($i=1;$i<13;$i++){
    if($i<10){?}

    <option
value="<?=$i?>"<?if(isset($_POST['tM'])&&$_POST['tM']==$i){print
"selected";}?>>0<?=$i?></option>
                                                                    <?}
                                                                    else{?>

    <option
value="<?=$i?>"<?if(isset($_POST['tM'])&&$_POST['tM']==$i){print
"selected";}?>><?=$i?></option>
                                                                    <?}

                                                                    }
                                                                    ?>
                                                                </select>

</td> <!-- month -->
<td> <!-- year -->
    <select name="tY">
                                                                    <?

for($i=2000;$i<2008;$i++){?}
                                                                    <option
value="<?=$i?>"<?if(isset($_POST['tY'])&&$_POST['tY']==$i){print
"selected";}?>><?=$i?></option>
                                                                    <?}?>
                                                                </select>

</td> <!-- year -->
</tr>
</table> <!-- Time period table -->
</td>
<td align="center" style="padding-left: 20px">
    <input type="submit" value="Show Graph"
onClick="checkSelection();">
    <input type="hidden" name="v" value="VIEW2">
    <input type="hidden" name="img" value="1">
    <!-- So as to stay in this tab-->
</td>
</tr>
</table> <!-- Tab 2 main table -->

```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```
</form>
<!-- FORM END-->
<!-- Graph table here (or graph picture?)-->
<?if($_POST['img']=='1'){
    $postTemp1=array_slice($_POST,0,1,true);
    $postTemp2=array_slice($_POST,-12,12,true);
    $plotsAr = array();
    if(isset($_POST['temp']))){
        $arrayTemp=array_merge($postTemp1,$postTemp2);
        $sar = array("attribute"=>0);
        $arrayTemp=array_merge($arrayTemp,$sar);
        array_push($plotsAr,$arrayTemp);
    }
    if(isset($_POST['light']))){
        $arrayLight=array_merge($postTemp1,$postTemp2);
        $sar = array("attribute"=>1);
        $arrayLight=array_merge($arrayLight,$sar);
        array_push($plotsAr,$arrayLight);
    }
    if(isset($_POST['accelX']))){
        $arrayAccelX=array_merge($postTemp1,$postTemp2);
        $sar = array("attribute"=>2);
        $arrayAccelX=array_merge($arrayAccelX,$sar);
        array_push($plotsAr,$arrayAccelX);
    }
    if(isset($_POST['magnX']))){
        $arrayMagnX=array_merge($postTemp1,$postTemp2);
        $sar = array("attribute"=>3);
        $arrayMagnX=array_merge($arrayMagnX,$sar);
        array_push($plotsAr,$arrayMagnX);
    }
    if(isset($_POST['humid']))){
        $arrayHumid=array_merge($postTemp1,$postTemp2);
        $sar = array("attribute"=>4);
        $arrayHumid=array_merge($arrayHumid,$sar);
        array_push($plotsAr,$arrayHumid);
    }
    if(isset($_POST['volt']))){
        $arrayVolt=array_merge($postTemp1,$postTemp2);
        $sar = array("attribute"=>5);
        $arrayVolt=array_merge($arrayVolt,$sar);
        array_push($plotsAr,$arrayVolt);
    }
    if(isset($_POST['accelY']))){
        $arrayAccelY=array_merge($postTemp1,$postTemp2);
        $sar = array("attribute"=>6);
        $arrayAccelY=array_merge($arrayAccelY,$sar);
        array_push($plotsAr,$arrayAccelY);
    }
    if(isset($_POST['magnY']))){
        $arrayMagnY=array_merge($postTemp1,$postTemp2);
        $sar = array("attribute"=>7);
        $arrayMagnY=array_merge($arrayMagnY,$sar);
        array_push($plotsAr,$arrayMagnY);
    }

    foreach ($plotsAr as $key => $value) {
        ?>&nbsp;   &nbsp;   <br>  
    <?}  
    }?>  
</td>  
</tr>
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

view_tab3.php

```
<SCRIPT LANGUAGE="JavaScript">
<!-- Begin

function moveOver() {
    var boxLength = document.choiceForm.choiceBox.length;
    var selectedItem = document.choiceForm.available.selectedIndex;
    var selectedText =
document.choiceForm.available.options[selectedItem].text;
    var selectedValue =
document.choiceForm.available.options[selectedItem].value;
    var i;
    var isNew = true;
    if (boxLength != 0) {
        for (i = 0; i < boxLength; i++) {
            thisitem = document.choiceForm.choiceBox.options[i].text;
            if (thisitem == selectedText) {
                isNew = false;
                break;
            }
        }
    }
    if (isNew) {
        newoption = new Option(selectedText, selectedValue, false,
false);
        document.choiceForm.choiceBox.options[boxLength] = newoption;
    }
    document.choiceForm.available.selectedIndex=-1;
}

function selectNodes() {
var boxLength = document.choiceForm.choiceBox.length;
    for (i = 0; i < boxLength; i++) {
        document.choiceForm.choiceBox.options[i].selected = 1;
        document.choiceForm.selNodes.value +=
document.choiceForm.choiceBox.options[i].value +",";
    }
}

function removeMe() {
    var boxLength = document.choiceForm.choiceBox.length;
    arrSelected = new Array();
    var count = 0;
    for (i = 0; i < boxLength; i++) {
        if (document.choiceForm.choiceBox.options[i].selected) {
            arrSelected[count] =
document.choiceForm.choiceBox.options[i].value;
        }
        count++;
    }
    var x;
    for (i = 0; i < boxLength; i++) {
        for (x = 0; x < arrSelected.length; x++) {
            if (document.choiceForm.choiceBox.options[i].value ==
arrSelected[x]) {
                document.choiceForm.choiceBox.options[i] = null;
            }
        }
    }
}
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

```
        boxLength = document.choiceForm.choiceBox.length;
    }
}

/*function saveMe() {
    var strValues = "";
    var boxLength = document.choiceForm.choiceBox.length;
    var count = 0;
    if (boxLength != 0) {
        for (i = 0; i < boxLength; i++) {
            if (count == 0) {
                strValues = document.choiceForm.choiceBox.options[i].value;
            }
            else {
                strValues = strValues + "," +
document.choiceForm.choiceBox.options[i].value;
            }
            count++;
        }
    }
    if (strValues.length == 0) {
        alert("You have not made any selections");
    }
    else {
        alert("Here are the values you've selected:\r\n" + strValues);
    }
}*/
// End -->
</script>

<?
    isset($_GET['pg']) ? ($pg = $_GET['pg']) : (isset($_POST['pg']) ?
$pg = $_POST['pg'] : $pg = 1);
    $hrefprefix = "index.php?v=".$v;
    $result = pg_query("select distinct nodeid from " . TABLE);
    //TABLE is defined in tools.php
?>

<!-- Navigation -->
<tr>
    <td align="left">
        <table cellspacing="1" cellpadding="2" border="0"
bgcolor="#354463" width="70%">
            <tr>
                <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEW1">1 Node - 1
Attribute</a></td>
                <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEW2">1 Node - N
Attributes</a></td>
                <td bgcolor="#354463" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEW3">N Nodes - 1
Attribute</a></td>
                <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEWRT">Real Time
Monitoring</a></td>
                <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEWQDB">DataBase Query</a></td>
            <!-- For extra navigation tabs uncomment below and maybe
change width above
                <td bgcolor="#5f7797" class="navcell" align="center"><a
```


Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```
class="navlink" href="pasilda.html">Links</a></td>
  <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="pasilda.html">Contact</a></td>
  -->
</tr>
</table>
</td>
</tr>

<!-- Page Middle -->
<tr>
  <td>
    <table cellspacing="1" cellpadding="0" border="1"
bgcolor="#354463" width="100%">
      <tr>
        <td bgcolor="#bfc4cb" style="padding-left: 5px">
          <!-- FORM BEGIN-->
          <form name="choiceForm" action="index.php" method="post">
            <table> <!-- Tab 3 main table -->
              <tr>
                <td>
                  <table>
                    <tr>
                      <td align="right">Attribute</td>
                      <td>
                        <select name="attribute">
                          <? $attribs = array("Temperature"=>"0",
                            "Light"=>"1",
                            "Acceleration X"=>"2",
                            "Magnititude X"=>"3",
                            "Humidity"=>"4",
                            "Voltage"=>"5",
                            "Acceleration Y"=>"6",
                            "Magnititude Y"=>"7");

                          foreach ($attribs as $key => $value) {
                            <?>
                              <option value="<?=$value?>"
<?if(isset($_POST['attribute'])&&$_POST['attribute']==$value){print
"selected";}>?><?=$key?></option><?}>
                                </select>
                              </td>
                            </tr>
                          <tr>
                            <td colspan="2">
                              <table> <!-- Time period table -->
                                <tr>
                                  <td></td>
                                  <td colspan="6">Time period</td>
                                </tr>
                                <tr>
                                  <td align="right">From</td>
                                  <td> <!-- hour -->
                                    <select
name="fHour">
                                      <?
for($i=0;$i<24;$i++){
if($i<10){?>
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```

        <option
value="<?=$i?"<?if(isset($_POST['fHour'])&&$_POST['fHour']==$i){prin
t "selected";}?>>0<?=$i?"</option>
                                                    <?}
                                                    else

if($i>=10&&$i<24){?>

        <option
value="<?=$i?"<?if(isset($_POST['fHour'])&&$_POST['fHour']==$i){prin
t "selected";}?>><?=$i?"</option>
                                                    <?}
                                                    else{?>

        <option
value="<?=$i?"<?if(isset($_POST['fHour'])&&$_POST['fHour']==$i){prin
t "selected";}?>>00</option>
                                                    <?}
                                                    }
                                                    ?>
                                                    </select>

</td> <!-- hour -->
<td> <!-- minute -->
    <select name="fMin">
                                                    <?

for($i=0;$i<60;$i++){
if($i==0){?>

        <option
value="<?=$i?"<?if(isset($_POST['fMin'])&&$_POST['fMin']==$i){print
"selected";}?>>00</option>
                                                    <?}
                                                    else

if($i<10){?>

        <option
value="<?=$i?"<?if(isset($_POST['fMin'])&&$_POST['fMin']==$i){print
"selected";}?>>0<?=$i?"</option>
                                                    <?}
                                                    else{?>

        <option
value="<?=$i?"<?if(isset($_POST['fMin'])&&$_POST['fMin']==$i){print
"selected";}?>><?=$i?"</option>
                                                    <?}
                                                    }
                                                    ?>
                                                    </select>

</td> <!-- minute -->
<td>&nbsp;&nbsp;&nbsp;</td>
<td> <!-- day -->
                                                    <select
name="fD">
                                                    <?

for($i=1;$i<32;$i++){
if($i<10){?>

<option

```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```

value="<?=$i?>"<?if(isset($_POST['fD'])&&$_POST['fD']==$i){print
"selected";}>>0<?=$i?></option>
                                                                 <?>
                                                                 else{?>

    <option
value="<?=$i?>"<?if(isset($_POST['fD'])&&$_POST['fD']==$i){print
"selected";}>><?=$i?></option>
                                                                 <?>
                                                                 }
                                                                 ?>
                                                                 </select>
    </td> <!-- day -->
    <td> <!-- month -->
        <select name="fM">
                                                                 <?>

    for($i=1;$i<13;$i++){
        if($i<10){?>
                                                                 <option
value="<?=$i?>"<?if(isset($_POST['fM'])&&$_POST['fM']==$i){print
"selected";}>>0<?=$i?></option>
                                                                 <?>
                                                                 else{?>

    <option
value="<?=$i?>"<?if(isset($_POST['fM'])&&$_POST['fM']==$i){print
"selected";}>><?=$i?></option>
                                                                 <?>
                                                                 }
                                                                 ?>
                                                                 </select>
    </td> <!-- month -->
    <td> <!-- year -->
        <select name="fY">
                                                                 <?>

    for($i=2000;$i<2008;$i++){?>

    <option
value="<?=$i?>"<?if(isset($_POST['fY'])&&$_POST['fY']==$i){print
"selected";}>><?=$i?></option>
                                                                 <?>?>
                                                                 </select>
    </td> <!-- year -->
</tr>
<tr>
    <td align="right">To</td>
    <td> <!-- hour -->
        <select name="tHour">
                                                                 <?>

    for($i=0;$i<24;$i++){

    if($i<10){?>

    <option
value="<?=$i?>"<?if(isset($_POST['tHour'])&&$_POST['tHour']==$i){prin
t "selected";}>>0<?=$i?></option>
                                                                 <?>

```


Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```

    <option
value="<?=$i?"<?if(isset($_POST['tD'])&&$_POST['tD']==$i){print
"selected";}>><?=$i?"></option>
                                                                    <?>
                                                                    }
                                                                    ?>
                                                                    </select>

    </td> <!-- day -->
    <td> <!-- month -->
        <select name="tM">
                                                                    <?>

for($i=1;$i<13;$i++){
    if($i<10){?>

    <option
value="<?=$i?"<?if(isset($_POST['tM'])&&$_POST['tM']==$i){print
"selected";}>>>0<?=$i?"></option>
                                                                    <?>
                                                                    else{?>

<option
value="<?=$i?"<?if(isset($_POST['tM'])&&$_POST['tM']==$i){print
"selected";}>>><?=$i?"></option>
                                                                    <?>
                                                                    }
                                                                    ?>
                                                                    </select>

    </td> <!-- month -->
    <td> <!-- year -->
        <select name="tY">
                                                                    <?>

for($i=2000;$i<2008;$i++){?>
                                                                    <option
value="<?=$i?"<?if(isset($_POST['tY'])&&$_POST['tY']==$i){print
"selected";}>>><?=$i?"></option>
                                                                    <?>?>
                                                                    </select>

        </td> <!-- year -->
    </tr>
</table> <!-- Time period table -->
</td>
</tr>
</table>
</td>
<td>
    <table> <!-- Available & selected Nodes Table -->
    <tr>
        <td align="center">Available</td>
        <td align="center">Selected</td>
    </tr>
    <tr>
        <td>
            <select name="available" size=5
onchange="moveOver();"
                <?while($row = pg_fetch_array($result)){?>
                    <option value="<?=$row['nodeid']?">
<?if(isset($_POST['nodeid'])&&$_POST['nodeid']==$row['nodeid']){print
"selected";}>>>Node ID <?=$row['nodeid']?"></option>

```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

```
        <?}?>
        </select>
    </td>
    <td>
        <select multiple name="choiceBox" style="width:
93px;" size=5 >
            </select>
        </td>
    </tr>
    <tr>
        <td colspan="2" align="center">
            <input type="button" value="Remove"
onclick="removeMe();" >
        </td>
    </tr>
</table> <!-- Available & selected Nodes Table -->
</td>
<td align="center" style="padding-left: 20px">
    <input type="submit" value="Show Graph"
onClick="selectNodes();" >
    <input type="hidden" name="v" value="VIEW3">
    <input type="hidden" name="img" value="1">
        <input type="hidden"
name="selNodes">
        <!-- So as to stay in this tab-->
    </td>
</tr>
</table>
</form>
<!-- FORM END-->
<?if($_POST['img']=='1'){?>
    &nbsp;&nbsp;&nbsp;
    <?}?>
</td>
</tr>
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

view_tabQDB.php

```
<SCRIPT LANGUAGE="JavaScript">
function createSQLQuery() {

    var attribs = new Array(
    "temp", "light", "accel_x", "mag_x", "humidity", "voltage", "accel_y", "mag_
y" );

    var sqlq, attribute, value1, value2, relation, nodeid, sort;
    var len = document.queryForm.nodeid.length;
    var i;
    var fHour, fMin, fD, fM, fY, tHour, tMin, tD, tM, tY, fDate, tDate;

    sqlq = "select all nodeid, ";
    attribute = attribs[document.queryForm.attribute.selectedIndex];

    sqlq += "result_time, " + attribute + " from q2 where ";

    if (document.queryForm.relation.selectedIndex == 2) {
        document.forms[0].elements['value2'].disabled = false;
        value1 = document.queryForm.value1.value;
        value2 = document.queryForm.value2.value;
        sqlq += attribute + ">" + value1 + " AND " + attribute + "<" +
value2;
        if(!document.queryForm.nodeid[0].selected)
            sqlq += " AND ";
    }
    else {
        document.forms[0].elements['value2'].disabled = true;
        value1 = document.queryForm.value1.value;
        relation =
document.queryForm.relation.options[document.queryForm.relation.selec
tedIndex].value;
        sqlq += attribute + relation + value1;
        if(!document.queryForm.nodeid[0].selected)
            sqlq += " AND ";
    }

    if (!document.queryForm.nodeid[0].selected) {

        //nodeid =
document.queryForm.nodeid.options[document.queryForm.nodeid.selectedI
ndex].value;
        //sqlq += "nodeid=" + nodeid + " AND ";
        sqlq += "(";
        for (i=1; i<len; i++) {
            if (document.queryForm.nodeid[i].selected) {
                nodeid = document.queryForm.nodeid[i].value;
                sqlq += "nodeid=" + nodeid + " OR ";
            }
        }
        sqlq = sqlq.substring(0, (sqlq.length)-4);
        sqlq += ")";
    }

    //Adding the from and to period selection
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```
fHour = document.queryForm.fHour.value;
fMin = document.queryForm.fMin.value;
fD = document.queryForm.fD.value;
fM = document.queryForm.fM.value;
fY = document.queryForm.fY.value;
tHour = document.queryForm.tHour.value;
tMin = document.queryForm.tMin.value;
tD = document.queryForm.tD.value;
tM = document.queryForm.tM.value;
tY = document.queryForm.tY.value;
fDate = fY + "-" + fM + "-" + fD + " " + fHour + ":" + fMin +
":00";
tDate = tY + "-" + tM + "-" + tD + " " + tHour + ":" + tMin +
":00";
if (document.queryForm.EnableFromPeriod.checked) {
    if (document.queryForm.EnableToPeriod.checked) {
        sqlq += " and result_time between '" + fDate + "' and '" +
tDate + "'";
    }
    else {
        sqlq += " and result_time > '" + fDate + "'";
    }
}
else if (document.queryForm.EnableToPeriod.checked) {
    sqlq += " and result_time < '" + tDate + "'";
}
//

sort = document.queryForm.sort.value;
sqlq += " order by ";
if (sort == 0)
    sqlq += "result_time";
if (sort == 1)
    sqlq += "nodeid";
if (sort == 2)
    sqlq += attribute;

if (document.queryForm.desc.checked)
    sqlq += " desc";

//document.queryForm.queryLabel.value = sqlq;
document.queryForm.toQuery.value = sqlq
//document.queryForm.queryLabel.visible = false;
//document.queryForm.query.value = sqlq;
//alert(sqlq);
}

function enableFromPeriod(){
    if (document.queryForm.EnableFromPeriod.checked) {
        document.queryForm.fHour.disabled=false;
        document.queryForm.fMin.disabled=false;
        document.queryForm.fD.disabled=false;
        document.queryForm.fM.disabled=false;
        document.queryForm.fY.disabled=false;
    }
    else {
        document.queryForm.fHour.disabled=true;
        document.queryForm.fMin.disabled=true;
        document.queryForm.fD.disabled=true;
        document.queryForm.fM.disabled=true;
        document.queryForm.fY.disabled=true;
    }
}
```


Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

```
}
}

function enableToPeriod(){
  if (document.queryForm.EnableToPeriod.checked) {
    document.queryForm.tHour.disabled=false;
    document.queryForm.tMin.disabled=false;
    document.queryForm.tD.disabled=false;
    document.queryForm.tM.disabled=false;
    document.queryForm.tY.disabled=false;
  }
  else {
    document.queryForm.tHour.disabled=true;
    document.queryForm.tMin.disabled=true;
    document.queryForm.tD.disabled=true;
    document.queryForm.tM.disabled=true;
    document.queryForm.tY.disabled=true;
  }
}
</SCRIPT>

<!-- Navigation -->
<tr>
  <td align="left">
    <table cellspacing="1" cellpadding="2" border="0"
    bgcolor="#354463" width="70%">
      <tr>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
        class="navlink" href="index.php?v=VIEW1">1 Node - 1
        Attribute</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
        class="navlink" href="index.php?v=VIEW2">1 Node - N
        Attributes</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
        class="navlink" href="index.php?v=VIEW3">N Nodes - 1
        Attribute</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
        class="navlink" href="index.php?v=VIEWRT">Real Time
        Monitoring</a></td>
        <td bgcolor="#354463" class="navcell" align="center"><a
        class="navlink" href="index.php?v=VIEWQDB">DataBase Query</a></td>
        <!-- For extra navigation tabs uncomment below and maybe
        change width above
        <td bgcolor="#5f7797" class="navcell" align="center"><a
        class="navlink" href="pasilda.html">Links</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
        class="navlink" href="pasilda.html">Contact</a></td>
        -->
      </tr>
    </table>
  </td>
</tr>

<!-- Page Middle -->
<tr>
  <td>
    <table cellspacing="1" cellpadding="0" border="1"
    bgcolor="#354463" width="100%">
      <tr>
        <td bgcolor="#bfc4cb" style="padding-left: 5px">
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```

<!-- FORM BEGIN-->
<form name="queryForm" action="index.php" method="post">
  <table border="0"> <!-- Tab QDB main table -->
    <tr>
      <td align="center">Node</td>
      <td align="center">Attribute</td>
      <td align="center">Relation</td>
      <td align="center">Value</td>
      <td align="center">Sort by</td>
      <td align="center">Order</td>
      <td rowspan="3" align="center" style="padding-left:
20px">
        <input type="submit" value="Get Report"
onClick="createSQLQuery();" />
        <input type="hidden"
name="v" value="VIEWQDB">
          <input type="hidden" name="table" value="1">
          <!-- So as to stay in this tab-->
          <input type="hidden" name="toQuery" value="">
        </td>
    </tr>
    <tr>
      <td rowspan="2">
        <select name="nodeid" size=4
onChange="createSQLQuery();" multiple>
          <? $result=pg_query("select distinct nodeid from " .
TABLE . " order by nodeid asc");?>
          //TABLE is defined in tools.php
          <option
value="All">All</option>
          <?
          while($row = pg_fetch_array($result)){
            <option
value="<?=$row['nodeid']?>"
<?if(isset($_POST['nodeid']) && $_POST['nodeid']==$row['nodeid']){print
"selected";}?>>ID <?=$row['nodeid']?></option>
            <?}?>
          </select>
        </td>
      <td valign="top" style="padding-left: 20px">
        <select name="attribute" onChange="createSQLQuery();">
          <? $attribs =
array("Temperature"=>"0",
          "Light"=>"1",
          "Acceleration X"=>"2",
          "Magnitude
X"=>"3",
          "Humidity"=>"4",
          "Voltage"=>"5",
          "Acceleration
Y"=>"6",
          "Magnitude
Y"=>"7");
          <? $attribs =
          foreach ($attribs
as $key => $value) {
            <?

```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```

value="<?=$value?>"
<?if(isset($_POST['attribute'])&&$_POST['attribute']==$value){print
"selected";}><?=$key?></option><?}?>
</select>
</td>
<td valign="top" style="padding-left: 20px">
<select name="relation" onChange="createSQLQuery();">
<?=$relation =
array("Greater Than >"=>">",
"Lower Than <"=>"<",
"Between"=>"><");
// "Equal To ="=>"=");

foreach ($relation as $rkey => $rvalue) {
?>
<option
value="<?=$rvalue?>"
<?if(isset($_POST['relation'])&&$_POST['relation']==$rvalue){print
"selected";}>><?=$rkey?></option><?}?>
</select>
</td>
<td valign="top" style="padding-left: 20px">
<!-- <textarea name="queryLabel" rows="1"
cols="80"><?print($_POST['queryLabel']);?></textarea><br><br>-->
<input type="text" name="value1" size="8"
onKeyPress="createSQLQuery();"
value="<?print($_POST['value1']);?>">&nbsp;
<input type="text"
name="value2" size="8" onKeyPress="createSQLQuery();"
value="<?print($_POST['value2']);?>">
</td>
<td valign="top" style="padding-left: 20px">
<select name="sort" onChange="createSQLQuery();">
<?=$sort =
array("Time (default)"=>"0",
"Node ID"=>"1",
"Value"=>"2");

foreach ($sort as $rkey => $rvalue) {
?>
<option
value="<?=$rvalue?>"
<?if(isset($_POST['sort'])&&$_POST['sort']==$rvalue){print
"selected";}>><?=$rkey?></option><?}?>
</select>
</td>
<td valign="top" style="font-weight: normal">
<input type="checkbox" name="desc"
<?if(isset($_POST['desc'])&&$_POST['desc']=="on"){print
"checked";}>>Descending</input>
</td>
</tr>
<tr>
<td colspan="5" style="padding-left: 20px">
<table border="0" cellspacing="0"> <!-- Time period
table -->
<tr>
<td colspan="8">
<input type="checkbox" name="EnableFromPeriod"
onChange="createSQLQuery(); enableFromPeriod();"

```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```
<?if(isset($_POST['EnableFromPeriod'])&&$_POST['EnableFromPeriod']=="on"){print "checked";}?>>Enable selection of "From period"</input>
</td>
<td colspan="8">
    <input type="checkbox" name="EnableToPeriod"
onChange="createSQLQuery(); enableToPeriod();"
<?if(isset($_POST['EnableToPeriod'])&&$_POST['EnableToPeriod']=="on")
{print "checked";}?>>Enable selection of "To period"</input>
</td>
</tr>
<tr>
<td align="right">From</td>
<td> <!-- hour -->
<select name="fHour"
<?if(!(isset($_POST['EnableFromPeriod'])&&$_POST['EnableFromPeriod']="on")){print "disabled=\"true\"";}?>>
<option value="-
1"<?if(isset($_POST['fHour'])&&$_POST['fHour']==-1){print
"selected";}?>>--</option>
<?
for($i=0;$i<24;$i++){
if($i<10){?>
<option
value="<?=$i?>"<?if(isset($_POST['fHour'])&&$_POST['fHour']==$i){prin
t "selected";}?>>0<?=$i?></option>
<?>
else
if($i>=10&&$i<24){?>
<option
value="<?=$i?>"<?if(isset($_POST['fHour'])&&$_POST['fHour']==$i){prin
t "selected";}?>><?=$i?></option>
<?>
else{?>
<option
value="<?=$i?>"<?if(isset($_POST['fHour'])&&$_POST['fHour']==$i){prin
t "selected";}?>>00</option>
<?>
}
?>
</select>
</td> <!-- hour -->
<td> <!-- minute -->
<select name="fMin"
<?if(!(isset($_POST['EnableFromPeriod'])&&$_POST['EnableFromPeriod']="on")){print "disabled=\"true\"";}?>>
<option value="-
1"<?if(isset($_POST['fMin'])&&$_POST['fMin']==-1){print
"selected";}?>>--</option>
<?
for($i=0;$i<60;$i++){
if($i==0){?>
<option
value="<?=$i?>"<?if(isset($_POST['fMin'])&&$_POST['fMin']==$i){print
"selected";}?>>00</option>
<?>
else
if($i<10){?>
```


Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```

                <select name="tM"
<?if(!(isset($_POST['EnableToPeriod']))&&$_POST['EnableToPeriod']=="on
")){print "disabled=\"true\"";}?>>
                <option value="-
1"<?if(isset($_POST['tM']))&&$_POST['tM']==-1){print "selected";}?>>--
</option>
                <?
                for($i=1;$i<13;$i++){
                    if($i<10){?>
                <option
value="<?=$i?>"<?if(isset($_POST['tM']))&&$_POST['tM']==$i){print
"selected";}?>>0<?=$i?></option>
                <?>
                else{?>
                <option
value="<?=$i?>"<?if(isset($_POST['tM']))&&$_POST['tM']==$i){print
"selected";}?>><?=$i?></option>
                <?>
                }
                ?>
                </select>
                </td> <!-- month -->
                <td> <!-- year -->
                <select name="tY"
<?if(!(isset($_POST['EnableToPeriod']))&&$_POST['EnableToPeriod']=="on
")){print "disabled=\"true\"";}?>>
                <option value="-
1"<?if(isset($_POST['tY']))&&$_POST['tY']==-1){print "selected";}?>>--
</option>
                <?
                for($i=2000;$i<2008;$i++){?>
                <option
value="<?=$i?>"<?if(isset($_POST['tY']))&&$_POST['tY']==$i){print
"selected";}?>><?=$i?></option>
                <?>?>
                </select>
                </td> <!-- year -->
                </tr>
                </table> <!-- Time period table -->
                </td>
                </tr>
                <tr>
                <?if($_POST['table']=='1'){?>
                <table width="700" cellspacing="2" cellpadding="5"
border="1">
                <tr>
                <td><strong>Node ID</strong></td>
                <td><strong>Attribute</strong></td>
                <td><strong>Value<? print units($_POST['attribute']);
?></strong></td>
                <td><strong>Time</strong></td>
                </tr>
                <?
                $query = $_POST['toQuery'];
                $result = pg_query($query);
                $row =

```


Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

```
pg_fetch_array($result);
                                $attr =
strtolower(attrib2($_POST['attribute']));
                                $at =
attrib($_POST['attribute']);
                                // $nodeid = $_POST['nodeid'];
                                ?>
                                <tr class="resultRow">
weight: normal"><?print($row['nodeid']);?></td>
                                <td style="font-
weight: normal"><?print($at);?></td>
                                <td style="font-
weight: normal"><?print($row[$attr]);?></td>
                                <td style="font-
weight: normal"><?print($row['result_time']);?></td>
                                </tr>
                                <?
                                while($row =
pg_fetch_array($result)){
                                ?>
                                <tr class="resultRow">
weight: normal"><?print($row['nodeid']);?></td>
                                <td style="font-
weight: normal"><?print($at);?></td>
                                <td style="font-
weight: normal"><?print($row[$attr]);?></td>
                                <td style="font-
weight: normal"><?print($row['result_time']);?></td>
                                </tr>
                                <?
                                }
                                }?>
                                </table>
                                </tr>
                                </table> <!-- Tab QDB main table -->
                                </form>
                                <!-- FORM END-->
                                </td>
                                </tr>
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

view_tabRT.php

```
<!-- Navigation -->
<tr>
  <td align="left">
    <table cellspacing="1" cellpadding="2" border="0"
bgcolor="#354463" width="70%">
      <tr>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEW1">1 Node - 1
Attribute</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEW2">1 Node - N
Attributes</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEW3">N Nodes - 1
Attribute</a></td>
        <td bgcolor="#354463" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEWRT">Real Time
Monitoring</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="index.php?v=VIEWQDB">DataBase Query</a></td>
      <!-- For extra navigation tabs uncomment below and maybe
change width above
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="pasilda.html">Links</a></td>
        <td bgcolor="#5f7797" class="navcell" align="center"><a
class="navlink" href="pasilda.html">Contact</a></td>
      -->
    </tr>
  </table>
</td>
</tr>

<!-- Page Middle -->
<tr>
  <td>
    <table cellspacing="1" cellpadding="0" border="1"
bgcolor="#354463" width="100%">
      <tr>
        <td bgcolor="#bfc4cb" style="padding-left: 5px">
          <table>
            <tr>
              <td>
                <APPLET ARCHIVE="LiveGraph.jar,jfreechart-
1.0.1.jar,jcommon-1.0.0.jar,postgresql-8.1-405.jdbc3.jar"
CODE="Applet7" width=750 height=260 ALT="You should see an applet,
not this text.">
                  <PARAM NAME="host"
VALUE="<?print(SERVER_FROM_FILE);?>">
                  <PARAM NAME="database"
VALUE="<?print(DATABASE_FROM_FILE);?>">
                  <PARAM NAME="username"
VALUE="<?print(USER_FROM_FILE);?>">
                  <PARAM NAME="password"
VALUE="<?print(PASS_FROM_FILE);?>">
                </APPLET>
              </td>
            </tr>
          </table>
          <td style="font-size: 12px; font-weight: normal;">
```


Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

Applet7.java (από το οποίο προέρχεται το αρχείο LiveGraph.jar)

```
import java.awt.Dimension;
import java.awt.GridLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.BoxLayout;
import javax.swing.ButtonGroup;
import javax.swing.JLabel;
import javax.swing.JRadioButton;
import javax.swing.JTextField;
import org.jfree.chart.ChartFactory;
import org.jfree.chart.ChartPanel;
import org.jfree.chart.JFreeChart;
import org.jfree.chart.axis.ValueAxis;
import org.jfree.chart.plot.XYPlot;
import org.jfree.chart.renderer.xy.DefaultXYItemRenderer;
import org.jfree.data.time.Millisecond;
import org.jfree.data.time.TimeSeries;
import org.jfree.data.time.TimeSeriesCollection;
import org.jfree.data.xy.XYDataset;

import java.sql.*;
import javax.swing.JApplet;
import javax.swing.JPanel;
import javax.swing.Timer;

public class Applet7 extends JApplet {

    /** The time series data. */
    private TimeSeries series[]=new TimeSeries[3];

    private double lastValue0 = 0.5;
    private double lastValue1 = 1.0;
    private double lastValue2 = 1.5;

    String host, database, username, password;

    int nodeZeroID = 0; int nodeOneID = 1; int nodeTwoID = 2;

    JLabel nodeZeroWarning, nodeOneWarning, nodeTwoWarning;

    JRadioButton radioButtonHumidity,    radioButtonVoltage,
        radioButtonTemperature, radioButtonLight,
        radioButtonAccel_x,    radioButtonAccel_y,
        radioButtonMag_x,    radioButtonMag_y;

    JTextField nodeZeroTextField, nodeOneTextField, nodeTwoTextField;

    /* Variables for the postgresql statement */
    Statement st;
    String sql1 = "SELECT result_time,";
    String sql2 = "temp";
    String sql3 = ",nodeid FROM q2 ORDER BY result_time DESC LIMIT 1";
    String sql = sql1 + sql2 + sql3;
    ResultSet rs;

    public Applet7() {
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```
        TimeSeriesCollection dataset[] = new TimeSeriesCollection[3];

        this.series[0] = new TimeSeries("Temperature 0",
Millisecond.class);
        this.series[1] = new TimeSeries("Temperature 1",
Millisecond.class);
        this.series[2] = new TimeSeries("Temperature 2",
Millisecond.class);
        dataset[0] = new TimeSeriesCollection(this.series[0]);
        dataset[1] = new TimeSeriesCollection(this.series[1]);
        dataset[2] = new TimeSeriesCollection(this.series[2]);
        //see TimeSeries setMaximumItemCount function...

        JFreeChart chart = createChart(dataset[0]);
        final ChartPanel chartPanel = new ChartPanel(chart);

        XYPlot plot = chart.getXYPlot();
        plot.setDataset(1, dataset[1]);
        plot.setDataset(2, dataset[2]);
        plot.setRenderer(1, new DefaultXYItemRenderer());
        plot.setRenderer(2, new DefaultXYItemRenderer());

        //The code for the controls on the right side of the applet
panel
        JPanel rightPanel = new JPanel();
        rightPanel.setLayout(new
BoxLayout(rightPanel, BoxLayout.PAGE_AXIS));

        //rightPanel.setBorder(BorderFactory.createLineBorder(Color.black));

        JPanel nodesPanel = new JPanel(new GridLayout(3, 4));
        JLabel nodeZeroLabel = new JLabel("#1 Node ID:");
        nodeZeroTextField = new JTextField("0", 3);
        nodeZeroWarning = new JLabel("<html><FONT
COLOR=RED><B>Invalid</B></FONT></html>");
        nodeZeroWarning.setVisible(false);
        nodeZeroLabel.setFont(new java.awt.Font("Arial", 0, 12));
        nodeZeroTextField.setFont(new java.awt.Font("Arial", 0, 12));
        nodeZeroWarning.setFont(new java.awt.Font("Arial", 0, 12));
        //////////////////////////////////////////////////
        JLabel nodeOneLabel = new JLabel("#2 Node ID:");
        nodeOneTextField = new JTextField("1", 3);
        nodeOneWarning = new JLabel("<html><FONT
COLOR=RED><B>Invalid</B></FONT></html>");
        nodeOneWarning.setVisible(false);
        nodeOneLabel.setFont(new java.awt.Font("Arial", 0, 12));
        nodeOneTextField.setFont(new java.awt.Font("Arial", 0, 12));
        nodeOneWarning.setFont(new java.awt.Font("Arial", 0, 12));
        //////////////////////////////////////////////////
        JLabel nodeTwoLabel = new JLabel("#3 Node ID:");
        nodeTwoTextField = new JTextField("2", 3);
        nodeTwoWarning = new JLabel("<html><FONT
COLOR=RED><B>Invalid</B></FONT></html>");
        nodeTwoWarning.setVisible(false);
        nodeTwoLabel.setFont(new java.awt.Font("Arial", 0, 12));
        nodeTwoTextField.setFont(new java.awt.Font("Arial", 0, 12));
        nodeTwoWarning.setFont(new java.awt.Font("Arial", 0, 12));
        //////////////////////////////////////////////////
        nodesPanel.add(nodeZeroLabel);
nodesPanel.add(nodeZeroTextField); nodesPanel.add(nodeZeroWarning);
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```
nodesPanel.add(nodeOneLabel);
nodesPanel.add(nodeOneTextField); nodesPanel.add(nodeOneWarning);
nodesPanel.add(nodeTwoLabel);
nodesPanel.add(nodeTwoTextField); nodesPanel.add(nodeTwoWarning);

nodesPanel.setPreferredSize(new Dimension(230,100));
nodesPanel.setMaximumSize(new Dimension(230,100));

JPanel attributesPanel = new JPanel(new GridLayout(4,2));
attributesPanel.setPreferredSize(new Dimension(230,110));
attributesPanel.setMaximumSize(new Dimension(230,110));
radioButtonHumidity = new JRadioButton("Humidity");
radioButtonHumidity.setName("Humidity");
radioButtonHumidity.setFont(new java.awt.Font("Arial", 0, 12));
//System.out.println(radioButtonHumidity.getName());
radioButtonVoltage = new JRadioButton("Voltage");
radioButtonVoltage.setName("Voltage");
radioButtonVoltage.setFont(new java.awt.Font("Arial", 0, 12));
radioButtonTemperature = new JRadioButton("Temperature");
radioButtonTemperature.setName("Temperature");
radioButtonTemperature.setFont(new java.awt.Font("Arial", 0,
12));
radioButtonTemperature.setSelected(true);
//the applet begins showing temperature graph..
radioButtonLight = new JRadioButton("Light");
radioButtonLight.setName("Light");
radioButtonLight.setFont(new java.awt.Font("Arial", 0, 12));
radioButtonAccel_x = new JRadioButton("Acceleration X");
radioButtonAccel_x.setName("Acceleration X");
radioButtonAccel_x.setFont(new java.awt.Font("Arial", 0, 12));
radioButtonAccel_y = new JRadioButton("Acceleration Y");
radioButtonAccel_y.setName("Acceleration Y");
radioButtonAccel_y.setFont(new java.awt.Font("Arial", 0, 12));
radioButtonMag_x = new JRadioButton("Magnitude X");
radioButtonMag_x.setName("Magnitude X");
radioButtonMag_x.setFont(new java.awt.Font("Arial", 0, 12));
radioButtonMag_y = new JRadioButton("Magnitude Y");
radioButtonMag_y.setName("Magnitude Y");
radioButtonMag_y.setFont(new java.awt.Font("Arial", 0, 12));

radioButtonHumidity.setActionCommand("humidity");
radioButtonVoltage.setActionCommand("voltage");
radioButtonTemperature.setActionCommand("temp");
radioButtonLight.setActionCommand("light");
radioButtonAccel_x.setActionCommand("accel_x");
radioButtonAccel_y.setActionCommand("accel_y");
radioButtonMag_x.setActionCommand("mag_x");
radioButtonMag_y.setActionCommand("mag_y");

attributesListener attribsListen = new attributesListener();
radioButtonHumidity.addActionListener(attribsListen);
radioButtonVoltage.addActionListener(attribsListen);
radioButtonTemperature.addActionListener(attribsListen);
radioButtonLight.addActionListener(attribsListen);
radioButtonAccel_x.addActionListener(attribsListen);
radioButtonAccel_y.addActionListener(attribsListen);
radioButtonMag_x.addActionListener(attribsListen);
radioButtonMag_y.addActionListener(attribsListen);

ButtonGroup group = new ButtonGroup();
group.add(radioButtonHumidity);
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```
group.add(radioButtonVoltage);
    group.add(radioButtonTemperature); group.add(radioButtonLight);
    group.add(radioButtonAccel_x);
group.add(radioButtonAccel_y);
    group.add(radioButtonMag_x);          group.add(radioButtonMag_y);

    attributesPanel.add(radioButtonTemperature);
attributesPanel.add(radioButtonVoltage);
    attributesPanel.add(radioButtonLight);
attributesPanel.add(radioButtonHumidity);
    attributesPanel.add(radioButtonAccel_x);
attributesPanel.add(radioButtonMag_x);
    attributesPanel.add(radioButtonAccel_y);
attributesPanel.add(radioButtonMag_y);

    rightPanel.add(attributesPanel);
    rightPanel.add(nodesPanel);
    //////////////////////////////////

    getContentPane().add(chartPanel);
    getContentPane().add(rightPanel, java.awt.BorderLayout.EAST);

    chartPanel.setPreferredSize(new java.awt.Dimension(500, 270));
    //new Applet4.DataGenerator().start();
    Applet7.DataGenerator app = new Applet7.DataGenerator();
    app.start();
}

private JFreeChart createChart(final XYDataset dataset) {
    final JFreeChart result = ChartFactory.createTimeSeriesChart(
        "Live Graph", "Time", "Value", dataset,
        true, true, false);
    final XYPlot plot = result.getXYPlot();
    ValueAxis axis = plot.getDomainAxis();
    //axis.setAutoRange(true);
    axis.setFixedAutoRange(30000.0); // 60 seconds
    axis = plot.getRangeAxis();
    axis.setRange(-30.0, 50.0);
    axis.setAutoRange(true);
    return result;
}

/**
 * Adds an observation to the 'zero' node time series.
 *
 * @param y the attribute value.
 */
private void addZeroObservation(double y) {
    if ( y != -1000) {
        lastValue0 = y;
    }
    series[0].add(new Millisecond(), lastValue0);
}

/**
 * Adds an observation to the 'one' node time series.
 *
 * @param y the attribute value.
 */
private void addOneObservation(double y) {
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```
        if ( y != -1000) {
            lastValue1 = y;
        }
        series[1].add(new Millisecond(), lastValue1);
    }

    /**
     * Adds an observation to the 'two' node time series.
     *
     * @param y the attribute value.
     */
    private void addTwoObservation(double y) {
        if ( y != -1000) {
            lastValue2 = y;
        }
        series[2].add(new Millisecond(), lastValue2);
    }

    boolean isNumber(String str) {
        String Digits = "0123456789";
        for (int i=0; i<str.length(); i++) {
            if (Digits.indexOf(str.charAt(i)) == -1) {
                return false;
            }
        }
        return true;
    }

    class attributesListener implements ActionListener {
        public void actionPerformed(ActionEvent e) {
            sql2 = e.getActionCommand();
            series[0].clear();
            series[1].clear();
            series[2].clear();
            if (isNumber(nodeZeroTextField.getText())) {
                nodeZeroID =
Integer.valueOf(nodeZeroTextField.getText());
                nodeZeroWarning.setVisible(false);
            }
            else
                nodeZeroWarning.setVisible(true);

            if (isNumber(nodeOneTextField.getText())) {
                nodeOneID = Integer.valueOf(nodeOneTextField.getText());
                nodeOneWarning.setVisible(false);
            }
            else
                nodeOneWarning.setVisible(true);

            if (isNumber(nodeTwoTextField.getText())) {
                nodeTwoID = Integer.valueOf(nodeTwoTextField.getText());
                nodeTwoWarning.setVisible(false);
            }
            else
                nodeTwoWarning.setVisible(true);

            JRadioButton radioButtonSelected =
(JRadioButton)e.getSource();
            series[0].setKey(radioButtonSelected.getName() + " " +
nodeZeroID);
```


Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων για Ασύρματα Δίκτυα Αισθητήρων

```
        series[1].setKey(radioButtonSelected.getName() + " " +
nodeOneID);
        series[2].setKey(radioButtonSelected.getName() + " " +
nodeTwoID);

    }
}

/**
 * The data generator.
 */
class DataGenerator extends Timer implements ActionListener {

    /**
     * Constructor.
     */
    DataGenerator() {
        super(1000, null); //Timer that ticks every 1000
milliseconds
        addActionListener(this);
    }

    /**
     * Adds a new free/total memory reading to the dataset.
     *
     * @param event the action event.
     */
    public void actionPerformed(ActionEvent event) {
        host = getParameter("host");
        database = getParameter("database");
        username = getParameter("username");
        password = getParameter("password");

        try {
            Class.forName("org.postgresql.Driver");
        }
        catch(ClassNotFoundException e) {
            System.err.print("ClassNotFoundException: ");
            System.err.println(e.getMessage());
        }
        try {
            String url = "jdbc:postgresql://" + host + "/" +
database;

            Connection conn = DriverManager.getConnection(url,
username, password);

            st = conn.createStatement();
            rs = st.executeQuery(sql1+sql2+sql3);
            rs.next();

            if (rs.getInt(3) == nodeZeroID) {
                addZeroObservation(rs.getDouble(2));
                addOneObservation(-1000);
                addTwoObservation(-1000);
            }
            else if (rs.getInt(3) == nodeOneID) {
                addZeroObservation(-1000);
                addOneObservation(rs.getDouble(2));
                addTwoObservation(-1000);
            }
        }
    }
}
```

Ολοκληρωμένο Σύστημα Καταγραφής Δεδομένων
για Ασύρματα Δίκτυα Αισθητήρων

```
    }
    else if (rs.getInt(3) == nodeTwoID) {
        addZeroObservation(-1000);
        addOneObservation(-1000);
        addTwoObservation(rs.getDouble(2));
    }
    else {
        addZeroObservation(-1000);
        addOneObservation(-1000);
        addTwoObservation(-1000);
    }
} //try
catch(SQLException e) {
    System.err.print("SQLException: ");
    System.err.println(e.getMessage());
    e.printStackTrace();
}
catch(Exception e) {
    System.err.print("Exception: ");
    System.err.println(e.getMessage());
    e.printStackTrace();
}
} //ActionPerformed

} //class DataGenerator
}
```