

Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης



Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Εφαρμοσμένης Πληροφορικής &
Πολυμέσων



Πτυχιακή Εργασία

Τίτλος:

Ανάπτυξη πλατφόρμας Ασύρματου Δικτύου Αισθητήρων (WSN) για την συλλογή περιβαλλοντικών δεδομένων και επεξεργασία με εφαρμογή Android.

Βασιλικός Γεώργιος (ΑΜ:1362)

Επιβλέπων καθηγητής: κ.Βλησίδης Ανδρέας

**Επιτροπή αξιολόγησης: κ.Βλησίδης Ανδρέας
κ.Μιαουδάκης Ανδρέας
κ.Σιδέρης Ανάργυρος**

Ημερομηνία παρουσίασης: 26/6/2013

Ευχαριστίες

Η παρούσα πτυχιακή εργασία εκπονήθηκε από το φοιτητή Βασιλικό Γεώργιο του Τμήματος Εφαρμοσμένης Πληροφορικής και Πολυμέσων του ΤΕΙ Ηρακλείου το ακαδημαϊκό έτος 2012 – 2013 υπό την επίβλεψη του καθηγητή του τμήματος κ. Βλησίδα Ανδρέα. Στον κ. Βλησίδα οφείλω τις θερμές μου ευχαριστίες για την καθοδήγηση και την υποστήριξη του καθ' όλη τη διάρκεια διεκπεραίωσης της παρούσας πτυχιακής.

Abstract

The preparation of this thesis is to succeed with the help of technology to collect environmental data in real time, wherever we are, so we then able to edit them. Developing dissertation done using a circuit which uses sensors to make the collection of environmental measurements, send data to a server and then displayed in the user's mobile phone with android application function or system computer (tablet, laptop, etc.).

Σύνοψη

Η εκπόνηση αυτής της πτυχιακής εργασίας έχει ως σκοπό να καταφέρουμε με την βοήθεια της τεχνολογίας να συλλέξουμε περιβαλλοντικά δεδομένα σε πραγματικό χρόνο ,οπουδήποτε κι αν βρισκόμαστε ,ώστε να έχουμε έπειτα την δυνατότητα να τα επεξεργαστούμε .Η ανάπτυξη της πτυχιακής εργασίας γίνεται με την χρήση ενός κυκλώματος το οποίο χρησιμοποιεί αισθητήρες ώστε να γίνει η συλλογή των περιβαλλοντικών μετρήσεων ,αποστέλλονται τα δεδομένα σε ένα διακομιστή και από εκεί εμφανίζονται στο κινητό τηλέφωνο του χρήστη με την λειτουργία εφαρμογής android ή στο υπολογιστικό του σύστημα(tablet,laptop,κτλ).

Πίνακας Περιεχομένων

Ευχαριστίες.....	2
Abstract	3
Σύνοψη	4
Πίνακας Περιεχομένων.....	5
Πίνακας Εικόνων	6
1 Εισαγωγή	8
1.1 Περίληψη.....	8
1.2 Κίνητρο για την Διεξαγωγή της Εργασίας	8
1.3 Σκοπός και Στόχοι Εργασίας	8
1.4 Δομή εργασίας	9
2 Μεθοδολογία Υλοποίησης	9
2.1 Θεωρία.....	9
3 Σχέδιο Δράσης για την εκπόνηση της εργασίας	10
3.1 State of the Art	10
3.2 Σημαντικοί Στόχοι Για την Ολοκλήρωση της Πτυχιακής	11
3.3 Χρονοδιάγραμμα Εργασίας	12
4 Κύριο μέρος Πτυχιακής.....	13
4.1 Περιβάλλον Σχεδίασης, Υλικό – Λογισμικό.....	13
4.1.1 Arduino uno.....	13
4.1.2 Αισθητήρας θερμοκρασίας Lm35.....	14
4.1.3 Αισθητήρας θερμοκρασίας – Υγρασίας DHT11.....	15
4.1.4 Κινητό τηλέφωνο SmartPhone (android).....	16
4.1.5 Arduino 1.0.1.....	17
4.1.6 Processing 2.0b5.....	18
4.1.7 Xampp-win32-01.8.1-VC9.....	19
4.1.8 Eclipse 4.2.1(Android).....	19
4.2 Σχεδιασμός Υλοποίησης.....	20
4.2.1 1ο Βήμα.....	21
4.2.2 2ο Βήμα.....	21
4.2.3 3ο Βήμα.....	22
4.2.4 4ο Βήμα.....	22
5 Υλοποίηση.....	22
5.1 Υλοποίηση 1ου Βήματος.....	22
5.1.1 Εκκίνηση λογισμικού Arduino 1.0.1.....	24
5.2 Υλοποίηση 2ου Βήματος.....	26
5.2.1 Εκκίνηση λογισμικού Xampp.....	27
5.2.2 Εκκίνηση λογισμικού Proseccing 2.0b5.....	33
5.2.3 Γλώσσα συγγραφής σεναρίων PHP.....	36
5.3 Υλοποίηση 3ου Βήματος.....	39
5.3.1 Δημιουργία αρχείου PHP.....	39
5.3.2 Δημιουργία εφαρμογής Android	40
6 Επίλογος.....	55
6.1 Μελλοντική εργασία και επεκτάσεις.....	55
Βιβλιογραφία.....	56
Παράρτημα.....	57
Παρουσίαση Πτυχιακής.....	58

Πίνακας εικόνων

Εικόνα 3.1: Χρονοδιάγραμμα Εργασίας.....	12
Εικόνα 4.1: Ηλεκτρονική πλατφόρμα Arduino Uno.....	14
Εικόνα 4.2: Αισθητήρας θερμοκρασίας LM35.....	15
Εικόνα 4.3: Αισθητήρας Υγρασίας – Θερμοκρασίας DHT11.....	16
Εικόνα 4.4: Κινητό τηλέφωνο SmartPhone(android).....	17
Εικόνα 4.5: Arduino 1.0.1.....	18
Εικόνα 4.6: Processing 2.0b5.....	18
Εικόνα 4.7: Xampp.....	19
Εικόνα 4.8: Eclipse.....	20
Εικόνα 4.9: Android Developer Tools.....	20
Εικόνα 4.10: Πρότυπο πλατφόρμας WSN και επεξεργασίας με εφαρμογή android.....	21
Εικόνα 5.1: Συνδεσμολογία LM35.....	23
Εικόνα 5.2: Συνδεσμολογία DHT11.....	23
Εικόνα 5.3: Σύνδεση πλακέτας- H/Y μέσω USB.....	24
Εικόνα 5.4: Εικονίδιο Arduino.....	24
Εικόνα 5.5: Κενό φύλλο εργασίας Arduino 1.0.1.....	25
Εικόνα 5.6: Κώδικας προγραμματισμού ηλεκτρονικής πλακέτας Arduino σε γλώσσα Wiring.....	26
Εικόνα 5.7: Εικονίδιο λογισμικού Xampp.....	27
Εικόνα 5.8: Πίνακας ελέγχου Xampp.....	27
Εικόνα 5.9: Πίνακας ελέγχου βάσης δεδομένων.....	28
Εικόνα 5.10: Επιλογή βάσης δεδομένων.....	29
Εικόνα 5.11: Φύλλο εντολών SQL.....	29
Εικόνα 5.12: Κώδικας δημιουργίας πίνακα.....	30
Εικόνα 5.13: Πίνακας “temp1”.....	30
Εικόνα 5.14: Πίνακας ελέγχου πλατφόρμας ηλεκτρονικού ταχυδρομείου.....	31
Εικόνα 5.15: Μενού διαχείρισης τοπικών χρηστών.....	32
Εικόνα 5.16: Μενού ιδιοτήτων χρήστη.....	32
Εικόνα 5.17: Δημιουργία ψευδώνυμου e-mail.....	33
Εικόνα 5.18: Προσθήκη Ψευδώνυμου e-mail.....	33
Εικόνα 5.19: Εικονίδιο λογισμικού Proseccing.....	34
Εικόνα 5.20: Φύλλο εργασίας λογισμικού Proseccing.....	34
Εικόνα 5.21: Κώδικας επικοινωνίας ηλεκτρονικής πλακέτας Arduino uno με βάση δεδομένων.....	35
Εικόνα 5.22: Εμφάνιση μετρήσεων σε ιστοσελίδα.....	36
Εικόνα 5.23: Κώδικας καταχώρισης στοιχείων για πρόσβαση στην Βάση Δεδομένων.....	36
Εικόνα 5.24: Κώδικας δημιουργίας ιστοσελίδας.....	37
Εικόνα 5.25: Μήνυμα συναγερμού.....	38
Εικόνα 5.26: Κώδικας για την επικοινωνία ΒΔ- Android.....	39
Εικόνα 5.27: Εικονίδιο λογισμικού Eclipse.....	40
Εικόνα 5.28: Βασικό παράθυρο Eclipse.....	40
Εικόνα 5.29: Παράθυρο εγκατάστασης νέου λογισμικού.....	41
Εικόνα 5.30: Εισαγωγή αποθήκης.....	42

Εικόνα 5.31: Εισαγωγή λογισμικού ADT.....	42
Εικόνα 5.32: Λίστα λογισμικών.....	43
Εικόνα 5.33: Παράθυρο όρων και αδειών.....	43
Εικόνα 5.34: Διαχείριση λογισμικών ανάπτυξης Android.....	44
Εικόνα 5.35: Εικονικό SmartPhone.....	45
Εικόνα 5.36: Προσθήκη Εικονικού SmartPhone.....	46
Εικόνα 5.37: Παράθυρο ιδιοτήτων εικονικού SmartPhone.....	47
Εικόνα 5.38: Ονομασία Project.....	48
Εικόνα 5.39: Προορισμός αποθήκευσης project.....	48
Εικόνα 5.40: Κώδικας δραστηριότητας οθόνης.....	49
Εικόνα 5.41: Κώδικας για την δημιουργία εφαρμογής Android.....	50
Εικόνα 5.42: Κώδικας για την δημιουργία εφαρμογής Android.....	51
Εικόνα 5.43: Κώδικας δραστηριότητας οθόνης.....	52
Εικόνα 5.44: Κώδικας για την δημιουργία εφαρμογής Android.....	53
Εικόνα 5.45: Ένδειξη δεδομένων εικονικού SmartPhone.....	54
Εικόνα 5.46: Ένδειξη δεδομένων εικονικού SmartPhone	54

1 Εισαγωγή

Αντικείμενο της παρούσας πτυχιακής εργασίας είναι η ανάπτυξη πλατφόρμας Ασύρματου Δικτύου Αισθητήρων (WSN) για την συλλογή περιβαλλοντικών δεδομένων και επεξεργασία με εφαρμογή Android. Αυτή η τεχνολογία μας παρέχει την δυνατότητα να αποκτήσουμε σε πραγματικό χρόνο διάφορα περιβαλλοντικά δεδομένα τα οποία μπορούμε να τα επεξεργαστούμε άμεσα χωρίς να είναι απαραίτητο να είμαστε στο ίδιο γεωγραφικό σημείο από το οποίο έγινε η συλλογή τους. Επίσης η πλατφόρμα αυτή έχει την επιπλέον δυνατότητα να καταχωρεί τα δεδομένα σε μια βάση δεδομένων ώστε να μπορούμε να ανακτήσουμε παλαιότερες μετρήσεις καθώς και να στέλνει αυτόματα ένα ηλεκτρονικό μήνυμα συναγερμού σε έναν ή περισσότερους παραλήπτες.

1.1 Περίληψη

Στην πτυχιακή αυτή θα χρησιμοποιήσουμε κάποιες γλώσσες προγραμματισμού για να προγραμματίσουμε το υλικό και το λογισμικό με βάση τα οποία θα αναπτυχθεί η πλατφόρμα. Το υλικό, το λογισμικό και οι γλώσσες προγραμματισμού αναφέρονται αναλυτικά στα επόμενα κεφάλαια.

Αρχικά θα αξιολογήσουμε τις δυνατότητες χαμηλού επιπέδου της ηλεκτρονική πλατφόρμας Arduino καθώς και των αισθητήρων LM35 και DHT11 ώστε να συλλέξουμε τα δεδομένα τα οποία μας ενδιαφέρουν. Έπειτα με τη χρήση του κατάλληλου λογισμικού που αναφέρετε σε επόμενο κεφάλαιο, ενός ηλεκτρονικού υπολογιστή ο οποίος λειτουργεί σαν διακομιστής και ενός ασύρματου δικτύου θα αποθηκεύσουμε τα δεδομένα μας σε μια βάση δεδομένων. Μετά θα δημιουργήσουμε την εφαρμογή Android την οποία και θα εγκαταστήσουμε στο κινητό τηλέφωνο. Τέλος θα δούμε με ποιο λογισμικό και ποιες γλώσσες προγραμματισμού θα καταφέρουμε να κάνουμε εφικτή την επικοινωνία ανάμεσα στους αισθητήρες, την ηλεκτρονική πλατφόρμα, τον διακομιστή, την βάση δεδομένων, το κινητό τηλέφωνο και το ηλεκτρονικό ταχυδρομείο.

1.2 Κίνητρο για την Διεξαγωγή της Εργασίας

Είναι χρονοβόρο και δαπανηρό το να συλλέγεις διάφορες περιβαλλοντικές πληροφορίες, να τις επεξεργάζεσαι και να χρειάζεται να βγάζεις διάφορα συμπεράσματα η ακόμα και να επεμβαίνεις σε αυτές. Η πτυχιακή αυτή στοχεύει στην ελαχιστοποίηση του χρόνου και του κόστους των περιβαλλοντικών μετρήσεων καθώς και στην πιθανή πρόβλεψη τους με την βοήθεια της τεχνολογίας.

1.3 Σκοπός και Στόχοι Εργασίας

Οι στόχοι αυτής της εργασίας είναι οι εξής:

- Η μετατροπή του αναλογικού σήματος των μετρήσεων θερμοκρασίας και υγρασίας σε ψηφιακό.

- Η δημιουργία ενός διακομιστή και μιας βάσης δεδομένων.
- Ο προγραμματισμός του υλικού και του λογισμικού ώστε να διασφαλίζετε η αυτόματη καταχώριση των μετρήσεων στην βάση δεδομένων.
- Η επικοινωνία και πρόσβαση στα δεδομένα της βάσης από το κινητό τηλέφωνο με την χρήση εφαρμογής Android.
- Η επικοινωνία και πρόσβαση στα δεδομένα της βάσης από τον ηλεκτρονικό υπολογιστή.
- Η αποστολή ηλεκτρονικού μηνύματος συναγερμού σε περίπτωση ακραίας περιβαλλοντικής μέτρησης.

1.4 Δομή εργασίας

Τα κεφάλαια καθώς και το θέμα με το οποίο ασχολείται το κάθε κεφάλαιο στο υπόλοιπο της εργασίας είναι δομημένο ως εξής:

2 Μεθοδολογία Υλοποίησης

Στην παρούσα πτυχιακή εργασία θα χρησιμοποιηθεί η ηλεκτρονική πλατφόρμα Arduino uno πάνω στην οποία θα συνδεθούν ο αισθητήρας LM35 για την μέτρηση της θερμοκρασίας και ο αισθητήρας DHT11 για την μέτρηση της υγρασίας. Για την επικοινωνία των αισθητήρων με την ηλεκτρονική πλακέτα καθώς και την μετατροπή του αναλογικού σήματος σε ψηφιακό θα χρησιμοποιηθεί το λογισμικό Arduino 1.0.1 και η γλώσσα προγραμματισμού C++. Για την δημιουργία διακομιστή θα χρησιμοποιηθεί ένας ηλεκτρονικός υπολογιστής τύπου Laptop μοντέλο Pavilion dv6 της εταιρίας Hewlett-Packard με λειτουργικό σύστημα Windows 7 καθώς και το λογισμικό Xampp-win32-1.8.1-VC9 το οποίο ενσωματώνει έναν εικονικό εξυπηρετητή με τη χρήση του λογισμικού Apache, μια εικονική βάση δεδομένων με τη χρήση του λογισμικού MySQL και μια πλατφόρμα ηλεκτρονικού ταχυδρομείου με τη χρήση του λογισμικού Mercury. Για την επικοινωνία και την αυτόματη καταχώριση των στοιχείων στην βάση δεδομένων θα χρησιμοποιηθεί το λογισμικό Processing 2.0b5 , η γλώσσα προγραμματισμού C++ και η γλώσσα συγγραφής σεναρίων PHP. Τέλος για την ανάπτυξη της εφαρμογής Android και της επικοινωνίας της με την βάση δεδομένων θα χρησιμοποιηθεί το λογισμικό Eclipse 4.2.1 , η γλώσσα προγραμματισμού JAVA και η γλώσσα συγγραφής σεναρίων PHP.

2.1 Θεωρία

Τα ασύρματα δίκτυα αισθητήρων αποτελούν μια από τις πολλά υποσχόμενες τεχνολογίες στην παροχή υποδομής για διεργασίες παρακολούθησης κατάστασης. Η μεγάλη εξάπλωση μικρών ασύρματων συσκευών, επιτρέπει την προσπέλαση της πληροφορίας οποιαδήποτε στιγμή και από παντού. Με τα ασύρματα δίκτυα αισθητήρων μπορούν να δημιουργηθούν επεκτάσιμα περιβάλλοντα παρακολούθησης, καθώς η προσθήκη κόμβων είναι άμεση.

3 Σχέδιο Δράσης για την Εκπόνηση της Εργασίας

Η υλοποίηση της πτυχιακής εργασίας θα υλοποιηθεί ακολουθώντας τις τελευταίες εξελίξεις της τεχνολογίας στην κατασκευή πλατφόρμας Ασύρματου Δικτύου Αισθητήρων, όπως περιγράφεται παρακάτω στο υποκεφάλαιο State of the Art.

3.1 State of the Art

Τις τελευταίες δεκαετίες η ανάπτυξη της τεχνολογίας των αισθητήρων άνοιξε εντελώς νέα πεδία για τις σύγχρονες τεχνολογικές εφαρμογές. Η συλλογή δεδομένων από το περιβάλλον σε συνδυασμό με την ανάπτυξη της μικροηλεκτρονικής έδωσε ώθηση στον αυτοματισμό και την επιστημονική παρατήρηση. Επιπλέον, οι δυνατότητες των αισθητήρων έδωσαν το κίνητρο για να διερευνηθεί η προοπτική της καταγραφής και επεξεργασίας δεδομένων τους από απόσταση, εκμεταλλεύοντας την παράλληλη ανάπτυξη των ασύρματων επικοινωνιών. Η συγχώνευση της τεχνολογίας των αισθητήρων, των υπολογιστικών μικροσυσκευών και των ασύρματων επικοινωνιών οδήγησε σε μία νέα καινοτομία, στα ασύρματα δίκτυα αισθητήρων (Wireless Sensor Networks - WSNs). Τα ασύρματα δίκτυα αισθητήρων (WSNs) αποτελούν σήμερα ένα πολλά υποσχόμενο πεδίο έρευνας στον τομέα των Ασύρματων Επικοινωνιών, δεδομένου του εκτενούς εύρους εφαρμογών που μπορούν να υποστηρίξουν. Συνιστούν μία νέα τάξη υπολογιστικών συστημάτων, που διευρύνει την ανθρώπινη δυνατότητα αλληλεπίδρασης με το φυσικό περιβάλλον. Ενώ μέχρι σήμερα τα ενσύρματα δίκτυα αισθητήρων χρησιμοποιούνταν για πολλούς σκοπούς, η ανάπτυξη της τεχνολογίας και των μικροηλεκτρονικών συστημάτων γενικά στον τομέα των ασύρματων επικοινωνιών έκαναν εφικτή και πιο οικονομική την χρήση των ασύρματων μέσων επίσης. Σίγουρα η χρήση των ασύρματων μέσων παρέχει μεγαλύτερη ευελιξία, ευκολία στη χρήση και γρήγορη ανάπτυξη των δικτύων αισθητήρων, παρόλο που σε κάποιες κρίσιμες εφαρμογές μπορεί να μειωθεί η αξιοπιστία της μεταφοράς δεδομένων.

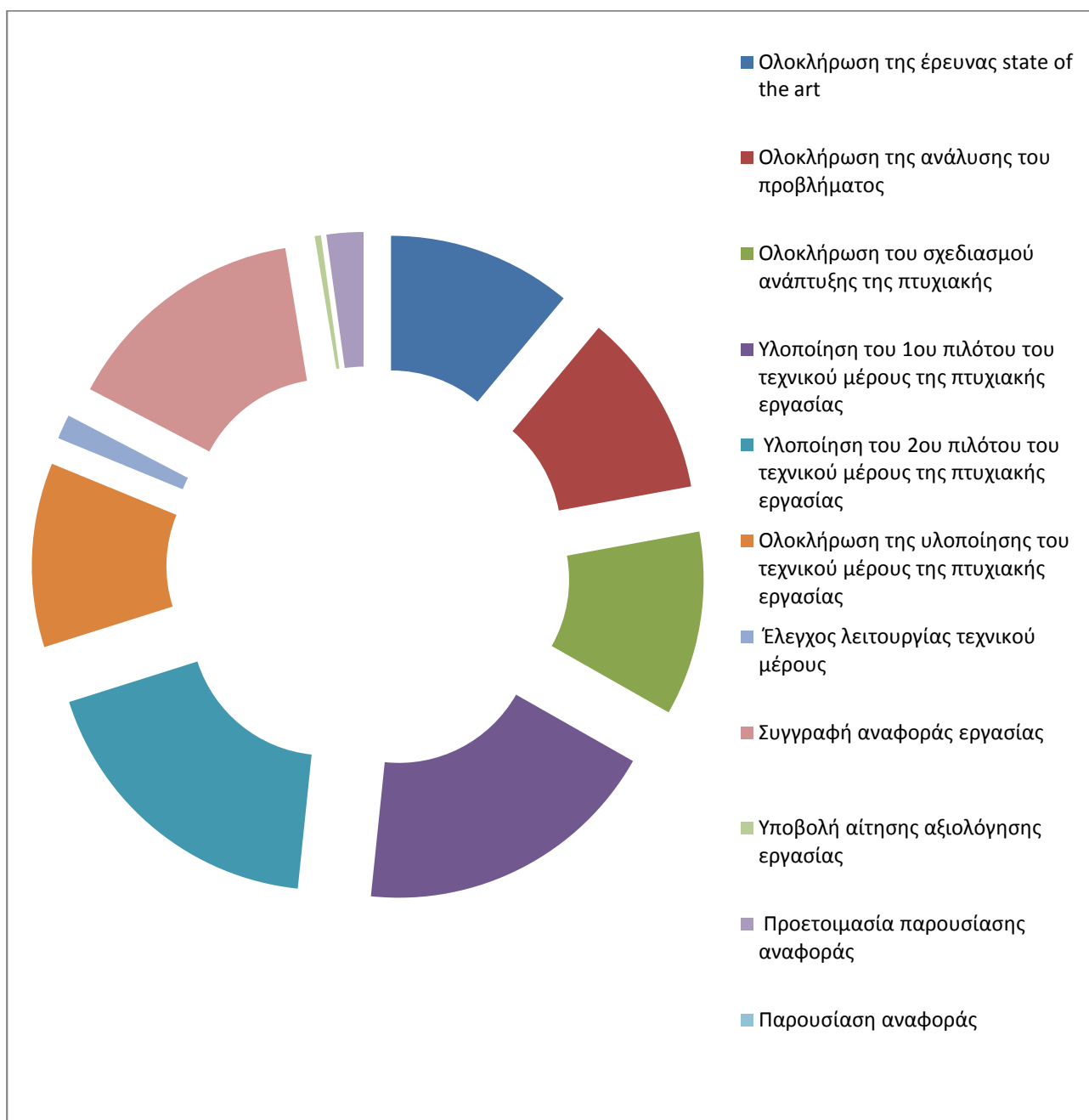
Αναλυτικότερα, τα Ασύρματα Δίκτυα Αισθητήρων (WSNs) αποτελούνται από ένα ή περισσότερα sink (ή base station) και από μερικές δεκάδες ή χιλιάδες κόμβους αισθητήρες (sensor nodes), οι οποίοι διασκορπίζονται σε ένα χώρο. Οι κόμβοι αυτοί συλλέγουν πληροφορίες από το περιβάλλον και ανάλογα με την εφαρμογή, είτε επεξεργάζονται τις πληροφορίες και τις στέλνουν, είτε τις στέλνουν χωρίς καμιά επεξεργασία. Οι κόμβοι αυτοί, συνήθως πρέπει να αισθάνονται τη θερμοκρασία, το φως, τη δόνηση, τον ήχο, την ακτινοβολία κ.α. Οι πληροφορίες αυτές “ταξιδεύουν” μέσα στο δίκτυο, έχοντας σαν τελικό προορισμό τους κόμβους sink. Ανάλογα με την εφαρμογή, τα sink ενδέχεται να αποστείλουν κάποια υποερωτήματα (queries) προς τους κόμβους, με σκοπό να μαζέψουν χρήσιμες πληροφορίες.

Το σημαντικό πλεονέκτημα, το οποίο παρέχει η εγκατάσταση ενός τέτοιου δικτύου είναι ότι δεν απαιτείται η εκ των προτέρων γνώση της τοπολογίας του. Η δυνατότητα αυτή επιτρέπει την ταχεία ανάπτυξη δικτύων αυτού του τύπου σε δύσβατες ή ακατάλληλες για τον άνθρωπο περιοχές. Οι κόμβοι είναι χαμηλού κόστους και χαμηλής κατανάλωσης και έχουν τη δυνατότητα να επικοινωνούν σε μικρές αποστάσεις, να εκτελούν περιορισμένη τοπική επεξεργασία δεδομένων και να «αισθάνονται» διαφόρων ειδών ερεθίσματα στην περιοχή εφαρμογής τους.

3.2 Σημαντικοί Στόχοι για την Ολοκλήρωση της Πτυχιακής

• Ολοκλήρωση της έρευνας state of the art	30
• Ολοκλήρωση της ανάλυσης του προβλήματος	30
• Ολοκλήρωση του σχεδιασμού ανάπτυξης της πτυχιακής	30
• Υλοποίηση του 1 ^{ου} πιλότου του τεχνικού μέρους της πτυχιακής εργασίας	50
• Υλοποίηση του 2 ^{ου} πιλότου του τεχνικού μέρους της πτυχιακής εργασίας	50
• Ολοκλήρωση της υλοποίησης του τεχνικού μέρους της πτυχιακής εργασίας	30
• Έλεγχος λειτουργίας τεχνικού μέρους	4
• Συγγραφή αναφοράς εργασίας	40
• Υποβολή αίτησης αξιολόγησης εργασίας	1
• Προετοιμασία παρουσίασης αναφοράς	6
• Παρουσίαση αναφοράς	1

3.3 Χρονοδιάγραμμα Εργασίας



Εικόνα 3.1: Χρονοδιάγραμμα Εργασίας

4 Κύριο μέρος Πτυχιακής

Ο στόχος αυτής της πτυχιακής είναι να αναπτύξουμε μια πλατφόρμα ασύρματου δικτύου αισθητήρων (WSN) για την συλλογή και επεξεργασία περιβαλλοντικών δεδομένων έτσι ώστε, να μπορούμε να έχουμε πρόσβαση στα δεδομένα μας με την χρήση μιας εφαρμογής Android.

4.1 Περιβάλλον Σχεδίασης, Υλικό - Λογισμικό

Το λειτουργικό σύστημα που θα χρησιμοποιηθεί είναι το Windows 7. Το υλικό που απαιτείτε είναι το εξής:

- Ηλεκτρονικός Υπολογιστής .
- Ηλεκτρονική πλακέτα Arduino uno.
- Αισθητήρας θερμοκρασίας LM35.
- Αισθητήρας υγρασίας DHT11 .
- Καλώδιο σύνδεσης περιφερικών τύπου USB.
- Κινητό τηλέφωνο τύπου SmartPhone με λειτουργικό Android.

Το λογισμικό που απαιτείτε είναι το εξής:

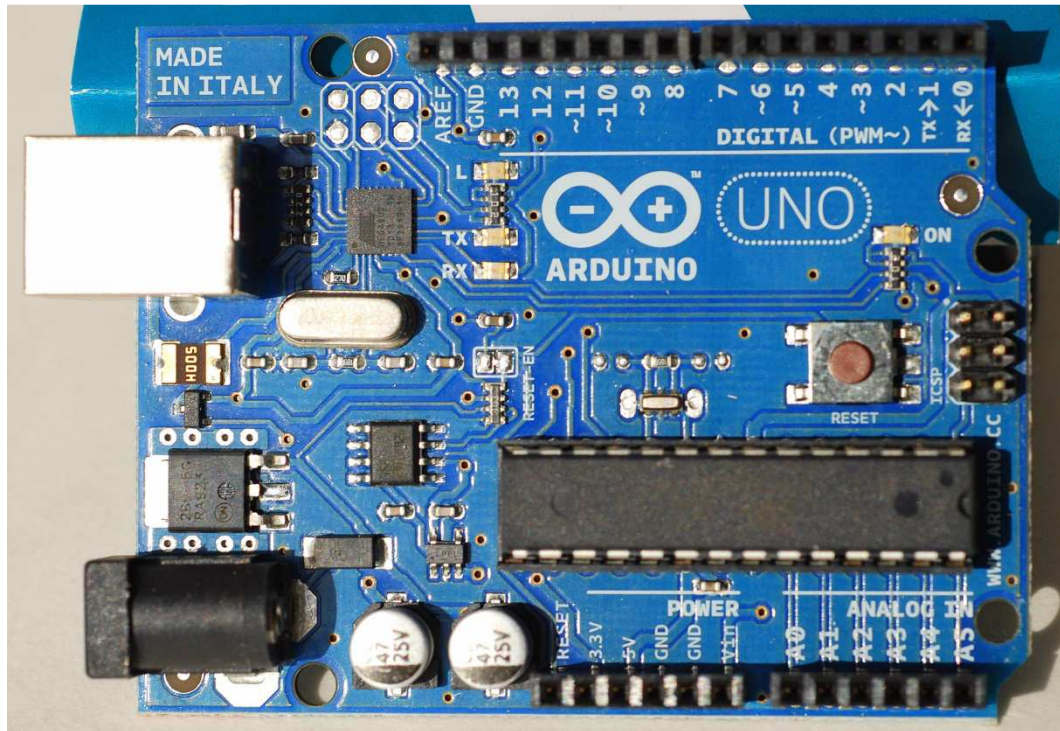
- Arduino 1.0.1
- Processing 2.0b5
- Xampp-win32-01.8.1-VC9 (Apache, MySQL, Mercury)
- Eclipse 4.2.1(Android)

Και οι γλώσσες προγραμματισμού και συγγραφής σεναρίων είναι οι εξής:

- C++
- JAVA
- PHP
- SQL
- Wiring

4.1.1 Arduino uno

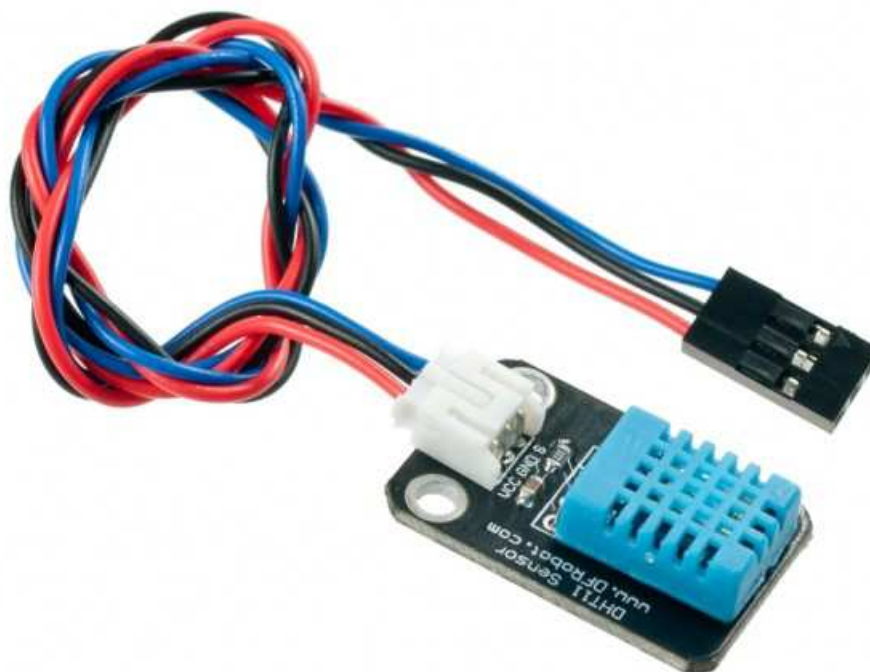
Το Arduino uno είναι μια υπολογιστική πλατφόρμα βασισμένη σε μια απλή μητρική πλακέτα με ενσωματωμένο μικροελεγκτή και εισόδους/εξόδους, και η οποία μπορεί να προγραμματιστεί με τη γλώσσα Wiring (ουσιαστικά πρόκειται για τη C++ με κάποιες μετατροπές). Το Arduino μπορεί να χρησιμοποιηθεί για την ανάπτυξη ανεξάρτητων διαδραστικών αντικειμένων αλλά και να συνδεθεί με υπολογιστή μέσω προγραμμάτων σε Processing, Max/MSP, Pure Data, SuperCollider. Οι περισσότερες εκδόσεις του Arduino μπορούν να αγοραστούν προ-συναρμολογημένες, το διάγραμμα και πληροφορίες για το υλικό είναι ελεύθερα διαθέσιμα για αυτούς που θέλουν να συναρμολογήσουν το Arduino μόνοι τους.



Εικόνα 4.1: Ηλεκτρονική πλατφόρμα Arduino Uno

4.1.2 Αισθητήρας θερμοκρασίας Lm35

Το LM35 είναι ένας αισθητήρας θερμοκρασίας, ακριβείας, σε ολοκληρωμένη μορφή. Η τάση εξόδου από τον αισθητήρα είναι γραμμικά ανάλογη της θερμοκρασίας Celsius. Έτσι το LM35 έχει ένα πλεονέκτημα σε σχέση με όλους εκείνους τους αισθητήρες στους οποίους η τάση εξόδου είναι ανάλογη της θερμοκρασίας Kelvin, αφού ο χρήστης δεν χρειάζεται να αφαιρεί μια σταθερή αλλά μεγάλη τάση για να προσαρμοστεί στην κλίμακα Celsius. Το LM35 δεν χρειάζεται κανένα εξωτερικό ρυθμιστικό κύκλωμα για να προσφέρει ακρίβεια $(\pm) 1/4$ C σε θερμοκρασία δωματίου και $(\pm) 3/4$ C σε φουλ κλίμακα θερμοκρασίας από -55 σε +150 C. Η μικρή σύνθετη αντίσταση εξόδου, η γραμμικότητά της και το εσωτερικό ρυθμιστικό κύκλωμα κάνουν πολύ εύκολη την προσαρμογή του αισθητήρα στο μικροελεγκτή. Μπορεί να πάρει απλή τροφοδοσία ή συμμετρική. Καταναλώνει πολύ μικρό ρεύμα 60 μ A και έτσι αναπτύσσει πολύ μικρή θερμοκρασία μόλις 0.1C.



Εικόνα 4.3: Αισθητήρας Υγρασίας – Θερμοκρασίας DHT11

4.1.4 Κινητό τηλέφωνο SmartPhone (android)

Με τα σύγχρονα κινητά τηλέφωνα μπορούμε πλέον να κάνουμε πολλά περισσότερα πράγματα από το να τηλεφωνούμε και να γράφουμε μηνύματα SMS. Τα λεγόμενα Smartphone (αγγλική λέξη που σημαίνει «έξυπνο τηλέφωνο») καθιερώθηκαν ως η νέα τάση της μόδας. Κινητά τηλέφωνα με πλήρεις δυνατότητες πρόσβασης στο Internet και με εξαιρετικά ευρύ φάσμα λειτουργιών, όπως σε ένα φορητό υπολογιστή, τα οποία λειτουργούν με λειτουργικό σύστημα και επιτρέπουν την εγκατάσταση εφαρμογών πολυμέσων. Ο χειρισμός των περισσότερων Smartphone γίνεται συνήθως μέσω μιας οθόνης αφής (Touchscreen), ενώ μερικά από αυτά διαθέτουν και μικρών διαστάσεων πληκτρολόγιο QWERTZ, όπως ένας υπολογιστής.

Το Android είναι λειτουργικό σύστημα για συσκευές κινητής τηλεφωνίας το οποίο τρέχει τον πυρήνα του λειτουργικού Linux. Αρχικά αναπτύχθηκε από την Google και αργότερα από την Open Handset Alliance. Επιτρέπει στους

κατασκευαστές λογισμικού να συνθέτουν κώδικα με την χρήση της γλώσσας προγραμματισμού Java, ελέγχοντας την συσκευή μέσω βιβλιοθηκών λογισμικού ανεπτυγμένων από την Google.



Εικόνα 4.4: Κινητό τηλέφωνο SmartPhone(android)

4.1.5 Arduino 1.0.1

Το Arduino 1.0.1 είναι ένα περιβάλλον προγραμματισμού ανοικτού κώδικα. Μπορεί να εγκατασταθεί στα λειτουργικά συστήματα των Windows, Mac OS X και Linux και η γλώσσα προγραμματισμού που χρησιμοποιεί ονομάζεται Wiring (ουσιαστικά πρόκειται για τη C++ με κάποιες μετατροπές).



Εικόνα 4.5: Arduino 1.0.1

4.1.6 Processing 2.0b5

Το processing είναι ένα περιβάλλον προγραμματισμού ανοικτού κώδικα και χρησιμοποιείτε περισσότερο από τους ανθρώπους που θέλουν να δημιουργήσουν εικόνες, κινούμενες εικόνες και αλληλεπιδράσεις. Αρχικά αναπτύχθηκε για να διδάξει βασικές αρχές του προγραμματισμού ηλεκτρονικών υπολογιστών σε οπτικό περιβάλλον, αργότερα όμως εξελίχθηκε σε εργαλείο για τη δημιουργία επαγγελματικών εργασιών. Σήμερα, υπάρχουν δεκάδες χιλιάδες φοιτητές, καλλιτέχνες, σχεδιαστές, ερευνητές και ερασιτέχνες που χρησιμοποιούν το processing για εκπαίδευση, δημιουργία προτύπων και παραγωγή προγραμμάτων.



Εικόνα 4.6: Processing 2.0b5

4.1.7 Xampp-win32-01.8.1-VC9

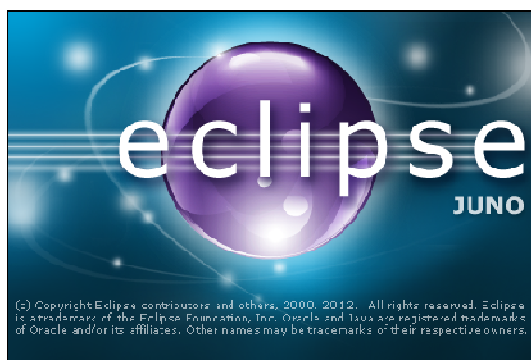
Το XAMPP είναι ένα πακέτο προγραμμάτων ελεύθερου λογισμικού, λογισμικού ανοικτού κώδικα και ανεξαρτήτου πλατφόρμας το οποίο περιέχει τον εξυπηρετητή ιστοσελίδων http Apache, την βάση δεδομένων MySQL, την πλατφόρμα ηλεκτρονικού ταχυδρομείου Mercury και ένα διερωτηματολόγιο για κώδικα γραμμένο σε γλώσσες προγραμματισμού PHP και Perl. Επίσης οι σχεδιαστές του XAMPP προόριζαν το λογισμικό ως εργαλείο ανάπτυξης και δοκιμής ιστοσελίδων τοπικά στον υπολογιστή χωρίς να είναι απαραίτητη η σύνδεση στο διαδίκτυο.



Εικόνα 4.7: Xampp

4.1.8 Eclipse 4.2.1(Android)

Το Eclipse αποτελεί ένα SDK (Software Development Kit), ένα ολοκληρωμένο δηλαδή περιβάλλον μέσα από το οποίο μπορούμε να γράψουμε και να εκτελέσουμε κώδικα. Το περιβάλλον αυτό καθώς και όλα τα υπόλοιπα προγράμματα που χρειάζονται για να εκτελέσουμε κώδικα σε Java ή/και C/C++ είναι ελεύθερης διανομής (freeware) και ανοικτού κώδικα (open source). Επιπλέον αποτελεί το ανερχόμενο περιβάλλον ανάπτυξης κώδικα, καθώς χρησιμοποιείται από ολοένα και περισσότερους χρήστες αλλά και εταιρείες. Η ανάπτυξη της εφαρμογής Android γίνεται μέσω του Eclipse ή άλλων συμβατών προγραμμάτων. Στην παρούσα πτυχιακή εργασία θα χρησιμοποιήσουμε το Android Developer Tools (ADT) το οποίο είναι πρόσθετο λογισμικό του Eclipse ώστε να δημιουργήσουμε την εφαρμογή μας και να είναι συμβατή με το λογισμικό Android το οποίο χρησιμοποιεί το SmartPhone.



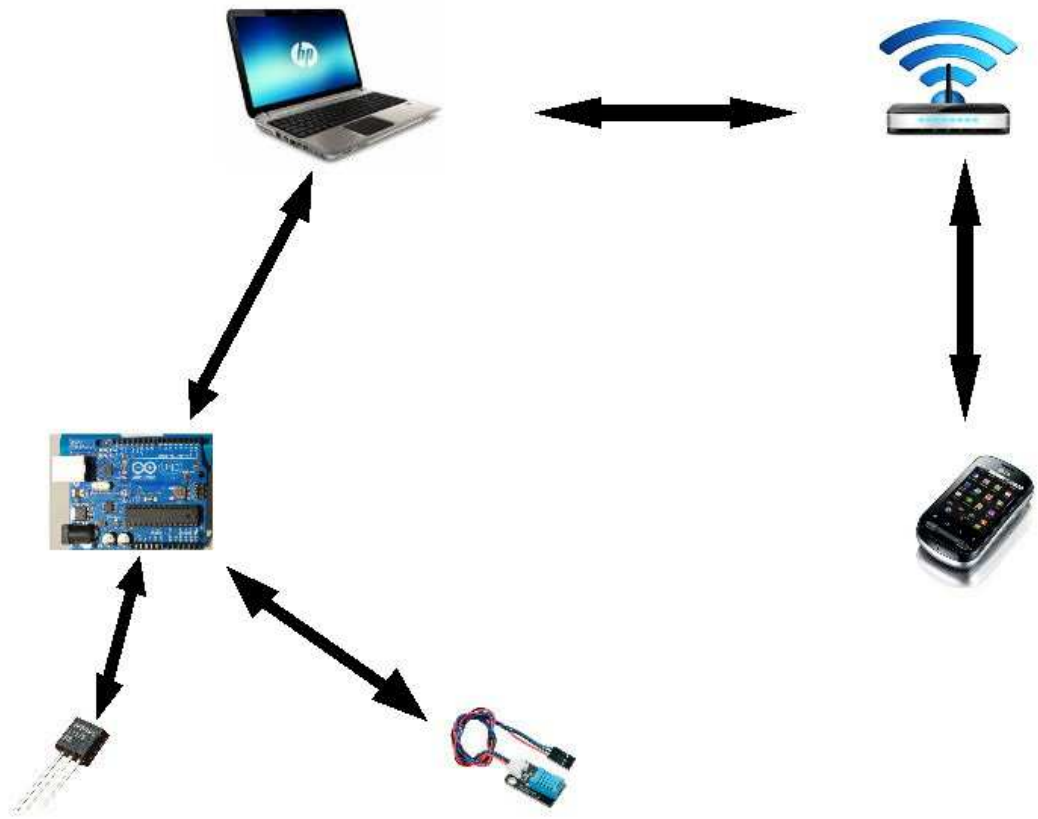
Εικόνα 4.8: Eclipse



Εικόνα 4.9: Android Developer Tools

4.2 Σχεδιασμός Υλοποίησης

Σε αυτό το κεφάλαιο θα σχεδιάσουμε και θα αναλύσουμε τον τρόπο υλοποίησης της πτυχιακής εργασίας. Εφόσον έχουμε στην κατοχή μας όλα αυτά που αναφέραμε στο προηγούμενο κεφάλαιο, τα οποία μας είναι απαραίτητα για την υλοποίηση του πρακτικού μέρους της πτυχιακής εργασίας, θα ξεκινήσουμε να αναπτύξουμε την πλατφόρμα του ασύρματου δικτύου αισθητήρων όπως φαίνεται στην εικόνα 4.10 διασπώντας τον σχεδιασμό μας σε κομμάτια.



Εικόνα 4.10: Πρότυπο πλατφόρμας WSN και επεξεργασίας με εφαρμογή android

4.2.1 1^ο Βήμα

Αρχικά συνδέουμε τους αισθητήρες LM35 και DHT11 στην ηλεκτρονική πλατφόρμα Arduino uno. Έπειτα με την χρήση του καλωδίου σύνδεσης περιφερειακών τύπου USB πραγματοποιούμε την τροφοδοσία της ηλεκτρονικής πλατφόρμας και την επικοινωνία της με το λειτουργικό Arduino 1.0.1 το οποίο εγκαθιστούμε στον ηλεκτρονικό υπολογιστή. Με την γλώσσα προγραμματισμού Wiring θα προγραμματίσουμε την ηλεκτρονική πλακέτα Arduino uno ώστε να επικοινωνήσει με τους αισθητήρες και να συλλέξουμε τα δεδομένα που θέλουμε.

4.2.2 2^ο Βήμα

Με την χρήση του λογισμικού Xampp θα δημιουργήσουμε έναν εικονικό εξυπηρετητή- διακομιστή ο οποίος μας παρέχει την δυνατότητα να έχουμε πρόσβαση σε μια βάση δεδομένων που θα χρησιμοποιήσουμε για την αποθήκευση των δεδομένων μας και επίσης την δημιουργία μίας εικονικής πλατφόρμας ηλεκτρονικού ταχυδρομείου που θα χρησιμοποιήσουμε για να στέλνουμε ένα μήνυμα συναγερμού σε περίπτωση που υπάρχει μια ακραία μέτρηση από τους αισθητήρες μας.

Η επικοινωνία του διακομιστή με την ηλεκτρονική πλατφόρμα Arduino uno, η αυτόματη καταχώρηση των στοιχείων στην βάση δεδομένων μας, η αποστολή

μηνύματος συναγερμού και η παρουσίαση των δεδομένων στο πρόγραμμα περιήγησης στο διαδίκτυο (browser) θα επιτευχθεί με την διαμεσολάβηση των λογισμικών Arduino 1.0.1, Proseccing 2.0b5, Xampp ,τις γλώσσες προγραμματισμού C++, SQL , την γλώσσα συγγραφής σεναρίων PHP και την παροχή υπηρεσιών διαδικτύου(Internet).

4.2.3 3^ο Βήμα

Το επόμενο στάδιο περιλαμβάνει την δημιουργία της εφαρμογής Android, την εγκατάσταση της στο κινητό τηλέφωνο καθώς και την επικοινωνία της με την βάση δεδομένων μας. Αυτό σημαίνει ότι η εφαρμογή Android θα πρέπει να ενσωματώνει την επικοινωνία της με το διαδίκτυο, την αδειοδότηση και επικοινωνία της με τον διακομιστή και την βάση δεδομένων, την αντιγραφή των στοιχείων από την βάση δεδομένων, την εμφάνιση τους στην οθόνη του κινητού τηλεφώνου και τέλος την αποστολή ενός μηνύματος συναγερμού σε περίπτωση που υπάρξει κάποια ακραία μέτρηση των αισθητήρων.

Όλα αυτά θα υλοποιηθούν με την χρήση του λογισμικού Eclipse 4.2.1, Android Developer Tools, Xampp, την γλώσσα προγραμματισμού JAVA, την γλώσσα συγγραφής σεναρίων PHP και την παροχή υπηρεσιών διαδικτύου(Internet).

4.2.4 4^ο Βήμα

Τέλος θα πρέπει να συντονίσουμε τα τρία προηγούμενα στάδια και να τα ενώσουμε με τέτοιο τρόπο ώστε να έχουμε το τελικό αποτέλεσμα στο πρακτικό τμήμα της πτυχιακής εργασίας. Στο επόμενο κεφάλαιο θα υλοποιήσουμε και θα αναπτύξουμε αναλυτικά την λύση που έχουμε σχεδιάσει στις προηγούμενες ενότητες.

5 Υλοποίηση

Αρχικά εφόσον έχουμε στην κατοχή μας το υλικό που είναι απαραίτητο για την εκπόνηση της πτυχιακής εργασίας, εγκαθιστούμε στον ηλεκτρονικό υπολογιστή όλα τα προγράμματα του λογισμικού που εμπλέκονται στην πτυχιακή εργασία και τα οποία αναφέραμε στο κεφάλαιο 4.1.

5.1 Υλοποίηση 1^ο Βήματος

Ξεκινάμε την υλοποίηση της πλατφόρμας συνδέοντας τους αισθητήρες LM35 και DHT11 με την ηλεκτρονική πλακέτα του Arduino υπο βάζοντας τους ακροδέκτες τους, στην πλακέτα με τον τρόπο που φαίνονται στις εικόνες 5.1 ,5.2.

Εικόνα 5.2: Συνδεσμολογία DHT11

Στη συνέχεια συνδέουμε την πλακέτα με τον ηλεκτρονικό υπολογιστή με την βοήθεια του καλωδίου σύνδεσης περιφερικών τύπου USB (εικόνα 5.3). Μόλις γίνει η σύνδεση ανάβει η λυχνία της τροφοδοσίας στην ηλεκτρονική πλακέτα.



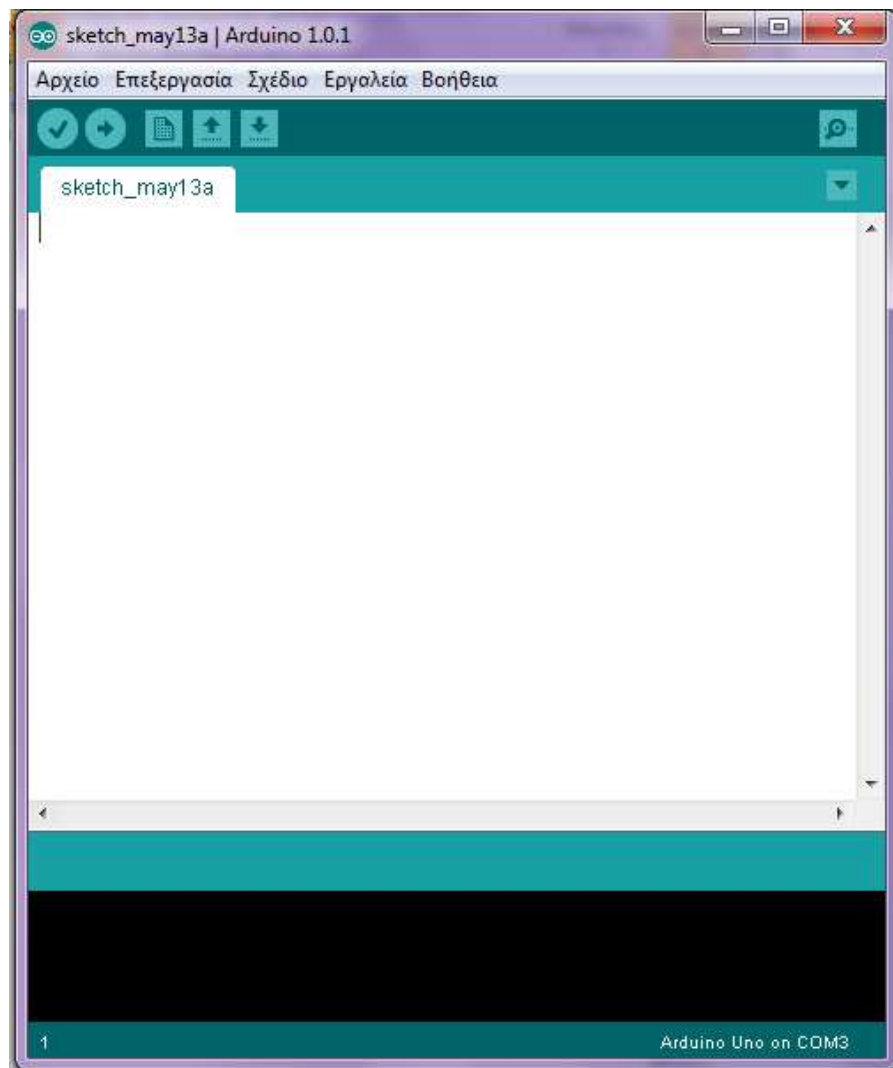
Εικόνα 5.3: Σύνδεση πλακέτας- Η/Υ μέσω USB

5.1.1 Εκκίνηση λογισμικού Arduino 1.0.1

Το επόμενο βήμα είναι να ξεκινήσει το λογισμικό Arduino 1.0.1. κάνοντας διπλό κλικ στο εικονίδιο Arduino (εικόνα 5.4). Όταν τρέξει η εφαρμογή θα εμφανιστεί στην οθόνη του υπολογιστή όπως φαίνεται στην εικόνα 5.5 όπου ανοίγει αυτόματα ένα κενό φύλλο εργασίας στο οποίο μπορούμε αμέσως να γράψουμε κώδικα στην γλώσσα προγραμματισμού Wiring.



Εικόνα 5.4: Εικονίδιο Arduino



Εικόνα 5.5: Κενό φύλλο εργασίας Arduino 1.0.1

Πριν ξεκινήσουμε να γράψουμε τον κώδικα θα πρέπει να πάμε στην μπάρα που υπάρχει πάνω από το φύλλο εργασίας και να επιλέξουμε: **Εργαλεία > Πλακέτα > Arduino uno** για να επιλέξουμε την πλακέτα με την οποία θα δουλέψουμε. Έπειτα επιλέγουμε ξανά **Εργαλεία > Σειριακή Θύρα** και διαλέγουμε την Σειριακή Θύρα όπου έχουμε συνδέσει το καλώδιο USB από την ηλεκτρονική πλακέτα στον ηλεκτρονικό υπολογιστή. Στην περίπτωση μας είναι η σειριακή θύρα COM3. Στην συνέχεια αρχίζουμε να γράφουμε τον κώδικα σε γλώσσα προγραμματισμού Wiring στο φύλλο εργασίας (εικόνα 5.6). Όταν γράφουμε τον κώδικα επιλέγουμε πάλι στην μπάρα επάνω από το φύλλο εργασίας **Σχέδιο > Επαλήθευση / Μεταγλώττιση** για να ελέγξει το πρόγραμμα αν υπάρχουν συντακτικά ή λειτουργικά λάθη. Εάν υπάρχουν μας τα επισημαίνει με έντονο χρώμα. Εάν δεν υπάρχουν επιλέγουμε πάλι **Αρχείο > Φόρτωση** και αρχίζει το φόρτωμα του προγράμματος που γράψαμε. Για να δούμε τα αποτελέσματα του προγράμματος στην οθόνη του υπολογιστή επιλέγουμε **Εργαλεία > Σειριακή Οθόνη** όπου εμφανίζετε μια οθόνη στην οποία αναγράφονται τα δεδομένα που παίρνουμε από τους αισθητήρες, δηλαδή η θερμοκρασία σε βαθμούς Κελσίου και η Υγρασία σε ποσοστό %. Τέλος ο κώδικας που έχουμε γράψει και φαίνετε στην εικόνα 5.6 ουσιαστικά προγραμματίζει την ηλεκτρονική πλακέτα

Arduino uno έτσι ώστε να αναγνωρίζει τους αισθητήρες που έχουμε συνδέσει σε αυτή, να παίρνει τα δεδομένα από τις μετρήσεις τους και να τα μετατρέπει στην μορφή που εμείς θέλουμε δηλαδή την θερμοκρασία σε βαθμούς Κελσίου και την υγρασία σε ποσοστό % και τέλος να τα εμφανίζει στην σειριακή οθόνη του προγράμματος. Η διαδικασία αυτή επαναλαμβάνετε κάθε 5 δευτερόλεπτα. Για να αποθηκεύσουμε το αρχείο μας επιλέγουμε **Αρχείο > Αποθήκευση** και αποθηκεύουμε στον φάκελο που θέλουμε και με το όνομα που θέλουμε το αρχείο.

```

1  #include <stdio.h>
2  #include <math.h>
3  #include <dht11.h>
4
5  dht11 DHT11;
6  #define DHT11PIN 2
7  const unsigned int SENSOR_PIN = A0;
8  const unsigned int BAUD_RATE = 9600;
9
10 double Thermokrasia (int RawADC){
11 double Temp;
12 Temp = analogRead(SENSOR_PIN);
13 Temp = (5.0 * Temp * 100.0)/1024.0;
14 return Temp;
15 }
16 void setup() {
17   Serial.begin(BAUD_RATE);
18   pinMode(SENSOR_PIN, INPUT);
19   pinMode(2, INPUT);
20 }
21
22 void loop() {
23
24   int temp = int(Thermokrasia(analogRead(SENSOR_PIN)));
25   int hum = DHT11.read(DHT11PIN);
26   int HUM=(int)DHT11.humidity;
27   int buf[]={temp,HUM};
28   int i;
29   for (i=0;i<2;i++)
30     Serial.println(buf[i]);
31   delay(5000);
32 }

```

Εικόνα 5.6: Κώδικας προγραμματισμού ηλεκτρονικής πλακέτας Arduino σε γλώσσα Wiring

5.2 Υλοποίηση 2^{οο} Βήματος

Στην παρούσα φάση θα χρησιμοποιήσουμε το λογισμικό Xampp για την δημιουργία του εικονικού εξυπηρετητή και της βάσης δεδομένων. Επίσης θα χρησιμοποιήσουμε το λογισμικό Proseccing 2.0b5 το οποίο μεσολαβεί για την

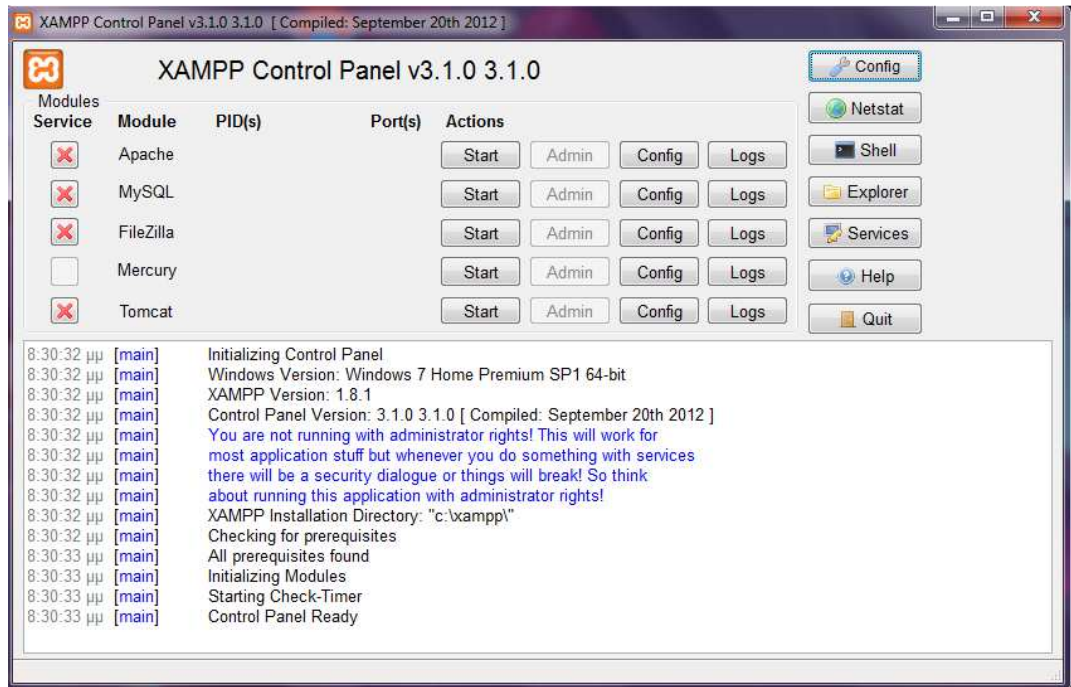
επικοινωνία ανάμεσα στην βάση δεδομένων και του λογισμικού Arduino 1.0.1 καθώς και στην αυτόματη καταχώριση των στοιχείων στη βάση. Τέλος θα δημιουργηθεί ένας εικονικός λογαριασμός ηλεκτρονικού ταχυδρομείου καθώς και μία ιστοσελίδα στην οποία θα μπορούμε να βλέπουμε σε μορφή πίνακα τα δεδομένα μας.

5.2.1 Εκκίνηση λογισμικού Xampp

Ανοίγουμε το λογισμικό Xampp πατώντας διπλό κλικ στο εικονίδιο Xampp Control Panel (εικόνα 5.7). Όταν τρέξει η εφαρμογή θα εμφανιστεί στην οθόνη του υπολογιστή ένα παράθυρο όπως φαίνεται στην εικόνα 5.8



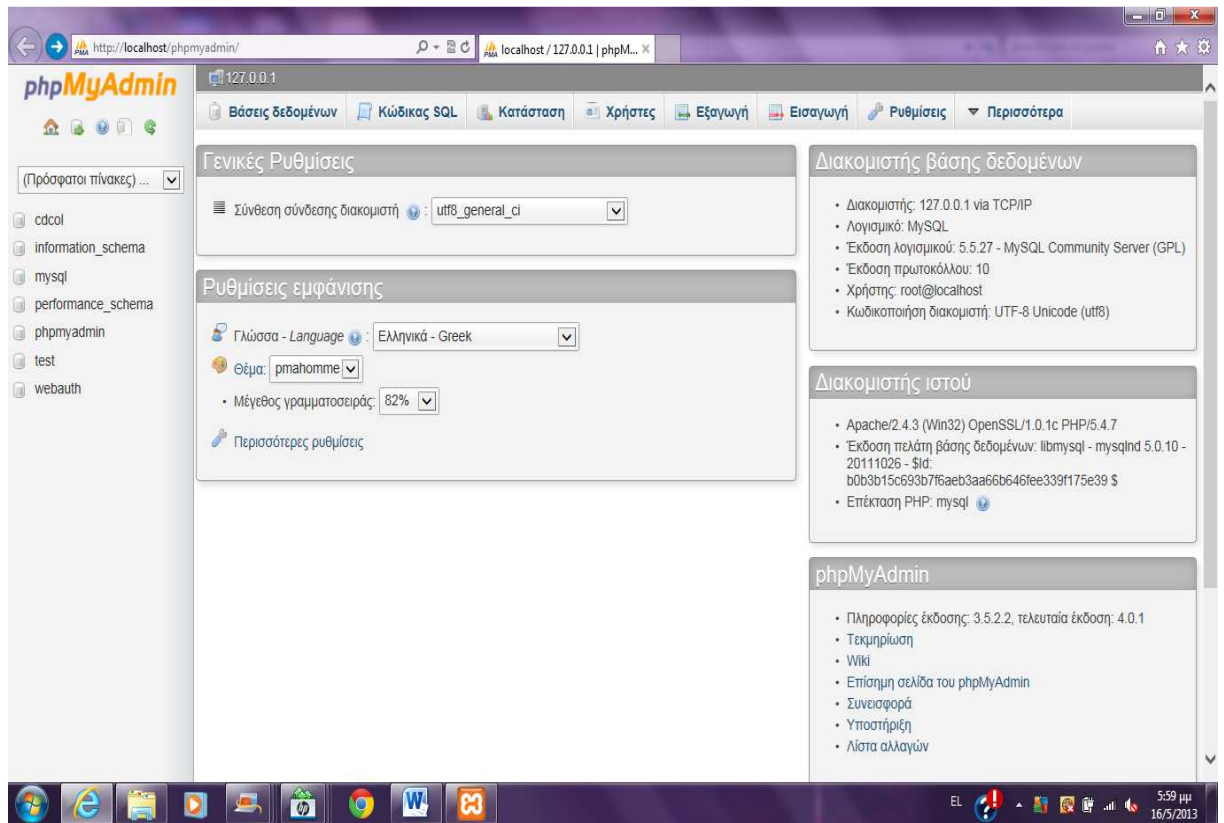
Εικόνα 5.7: Εικονίδιο λογισμικού Xampp



Εικόνα 5.8: Πίνακας ελέγχου Xampp

Κάνουμε κλικ στο κουμπί **Start** το οποίο είναι στην ίδια ευθεία με την ετικέτα **Apache** καθώς και την ετικέτα **MySQL** και **Mercury** αντίστοιχα, δηλαδή στο 1^ο, 2^ο, και 4^ο κουμπί της στήλης **Actions** που φαίνεται στην εικόνα 5.8. Με αυτόν τον τρόπο

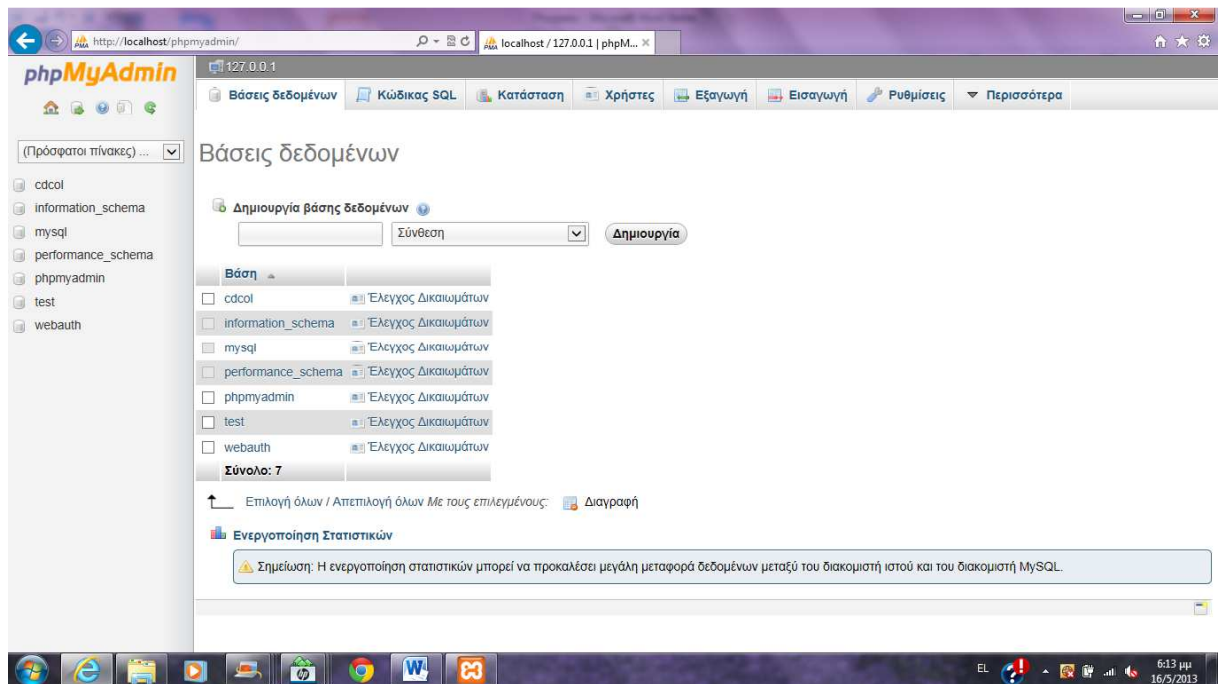
επιτυγχάνουμε την εκκίνηση του εξυπηρετητή, της βάσης δεδομένων και του ηλεκτρονικού ταχυδρομείου. Πατώντας το κουμπί Admin της MySQL ανοίγει μία σελίδα στο πρόγραμμα περιήγησης στο διαδίκτυο (browser) η οποία μας παραπέμπει στην διαχείριση της βάσης δεδομένων (εικόνα 5.9).



Εικόνα 5.9: Πίνακας ελέγχου βάσης δεδομένων

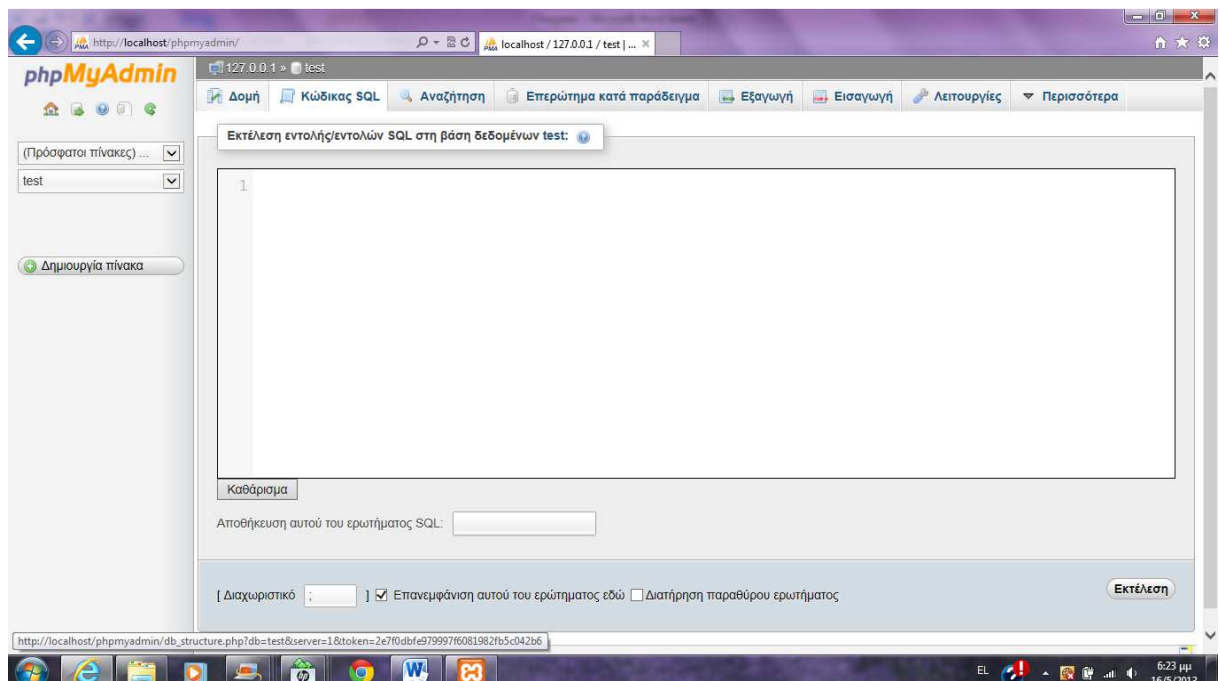
Στην αριστερή στήλη του παραθύρου βλέπουμε τις βάσεις δεδομένων που υπάρχουν, στο κέντρο βλέπουμε κάποιες καρτέλες οι οποίες μας παρέχουν την δυνατότητα να κάνουμε διάφορες ρυθμίσεις, και στα δεξιά του παραθύρου βλέπουμε ορισμένες καρτέλες με διάφορες πληροφορίες του συστήματος. Τέλος στο πάνω μέρος του παραθύρου υπάρχει μία σειρά από διαφορετικές καρτέλες οι οποίες μας προσφέρουν διάφορες λειτουργίες- υπηρεσίες.

Επιλέγοντας την καρτέλα **Βάσεις δεδομένων** εμφανίζονται όλες οι βάσεις δεδομένων που υπάρχουν στο λογισμικό καθώς και η δυνατότητα να δημιουργήσουμε ή να διαγράψουμε κάποια βάση δεδομένων (εικόνα 5.10).



Εικόνα 5.10: Επιλογή βάσης δεδομένων

Επιλέγουμε την βάση δεδομένων με όνομα “test”. Αμέσως μετά επιλέγουμε την διπλανή καρτέλα **Κώδικας SQL** (εικόνα 5.11), γράφουμε κώδικα σε γλώσσα SQL (εικόνα 5.12) και πατάμε εκτέλεση.



Εικόνα 5.11: Φύλλο εντολών SQL

```

1 CREATE TABLE temp1 (
2     id INT(10) NOT NULL AUTO_INCREMENT,
3     PRIMARY KEY(id),
4     Time TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
5     Celsius INT(3) NULL,
6     Hum INT(3) NULL
7 )
    
```

Εικόνα 5.12: Κώδικας δημιουργίας πίνακα

Ο κώδικας της εικόνας 5.12 δημιουργεί ένα πίνακα με όνομα “**temp1**” ο οποίος έχει τέσσερις στήλες με τις εξής ετικέτες: «id», «Time», «Celsius», «Hum». Η στήλη «id» είναι και το πρωτεύον κλειδί του πίνακα που σημαίνει ότι όλες τις οι τιμές θα είναι μοναδικές και επίσης θα αυξάνετε η τιμή της αυτόματα κάθε φορά που υπάρχει μια νέα εγγραφή. Η στήλη «Time» και αυτή καταχωρεί δεδομένα αυτόματα κάθε φορά που υπάρχει μία νέα εγγραφή, δηλαδή θα καταχωρεί την ημερομηνία και την ώρα που έγινε η κάθε εγγραφή στον πίνακα. Τέλος οι στήλες «Celsius» και «Hum» θα είναι κενές και θα καταχωρείτε σε αυτές ένας ακέραιος, θα είναι ουσιαστικά τα δεδομένα από την θερμοκρασία και την υγρασία που θα παίρνουμε από τους αισθητήρες.

Όταν εκτελεστούν οι εντολές του παραπάνω κώδικα θα έχει δημιουργηθεί στην βάση δεδομένων “**test**” ο πίνακας “**temp1**” με τα χαρακτηριστικά που προαναφέραμε (εικόνα 5.13).

The screenshot shows the phpMyAdmin interface for a database named 'test'. The table 'temp1' is selected, and its structure is displayed in a table format. The columns are:

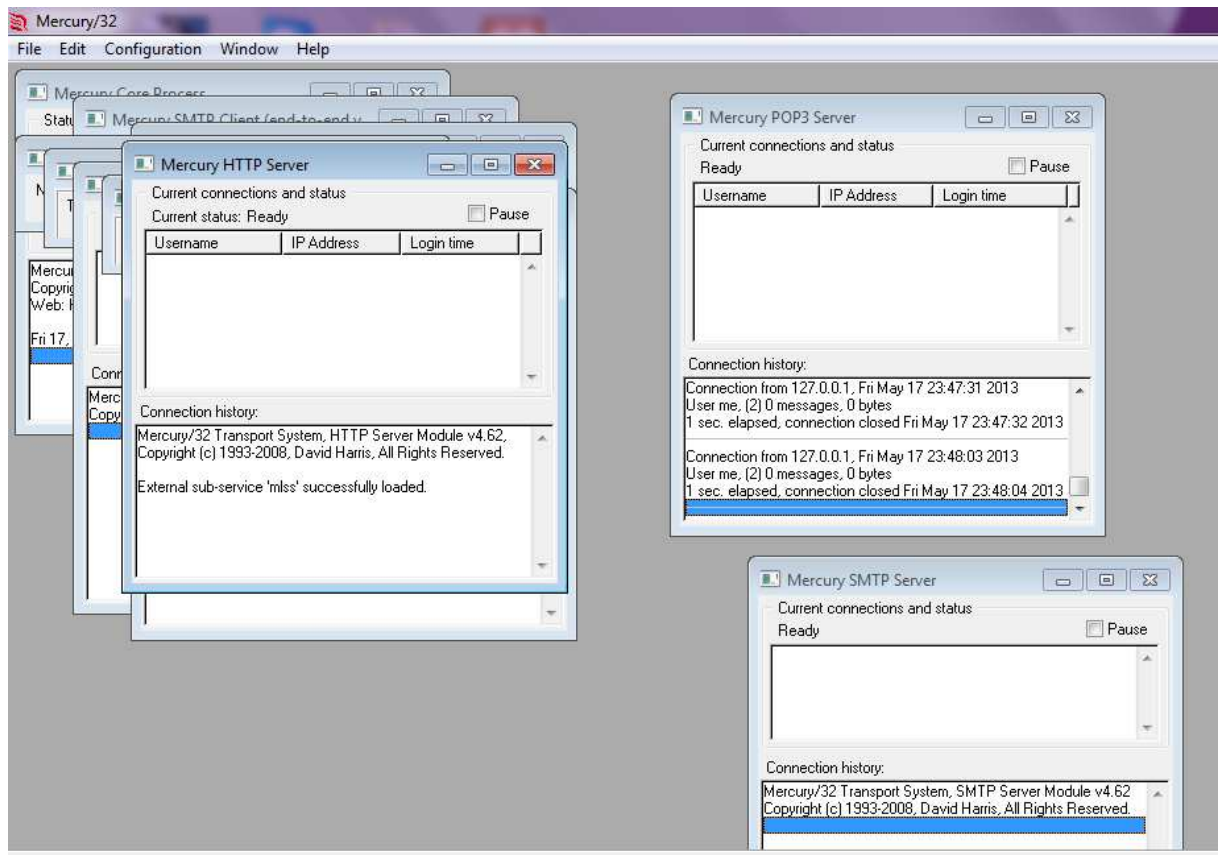
#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα	Ενέργεια
1	id	int(10)			Όχι	Καμία	AUTO_INCREMENT	Αλλαγή Διαγραφή Περισσότερα
2	Time	timestamp			Όχι	CURRENT_TIMESTAMP		Αλλαγή Διαγραφή Περισσότερα
3	Celsius	int(3)			Ναι	NULL		Αλλαγή Διαγραφή Περισσότερα
3	Hum	int(3)			Ναι	NULL		Αλλαγή Διαγραφή Περισσότερα

Below the table structure, there are statistics for the table:

Χρήση χώρου	Στατιστικά Εγγραφών
Δεδομένα: 16 KB	Μορφοποίηση: Compact
Ευρετήριο: 0 B	Σύνθεση: latin1_swedish_ci
Σύνολο: 16 KB	Επόμενη αυτόματη αρίθμηση: 75
	Δημιουργία: 02 Μαρ 2013 στις 00:40:30

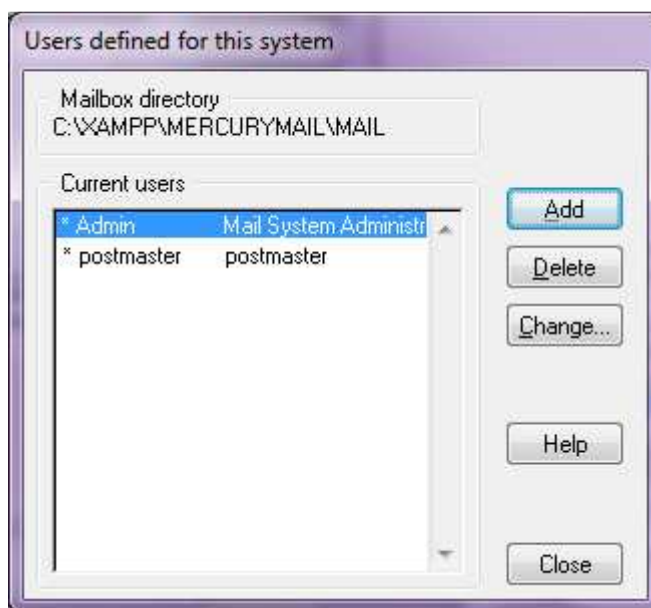
Εικόνα 5.13: Πίνακας “temp1”

Επιστρέφοντας στο αρχικό παράθυρο του Χαμpp κάνουμε κλικ στο κουμπί Admin της πλατφόρμας ηλεκτρονικού ταχυδρομείου Mercury. Εμφανίζεται ένα παράθυρο (εικόνα 5.14) το οποίο είναι ο πίνακας ελέγχου της πλατφόρμας.

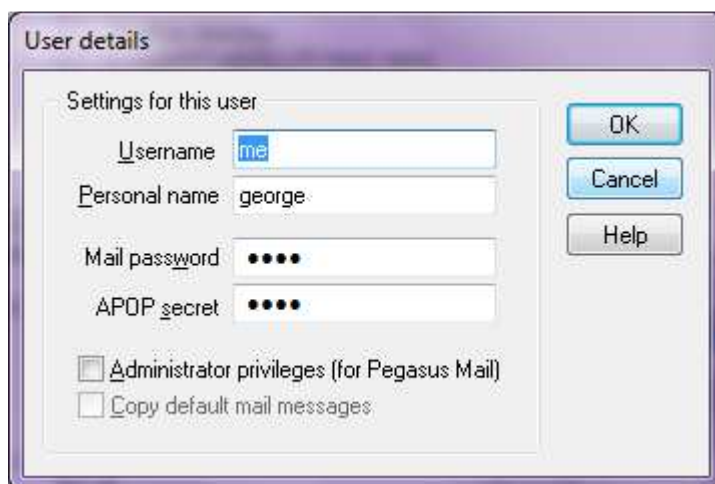


Εικόνα 5.14: Πίνακας ελέγχου πλατφόρμας ηλεκτρονικού ταχυδρομείου

Θα δημιουργήσουμε έναν εικονικό λογαριασμό e-mail ο οποίος θα μας χρησιμεύσει στο να δέχεται τα μηνύματα συναγεμίου σε περίπτωση ακραίων μετρήσεων από τους αισθητήρες μας. Για να δημιουργήσουμε τον λογαριασμό αυτό ακολουθούμε τα εξής βήματα: Στο μενού που βρίσκετε στο επάνω μέρος του παραθύρου επιλέγουμε **Configuration > Manage local users**. Ανοίγει ένα νέο παράθυρο (εικόνα 5.15) πατάμε κλικ στο κουμπί Add και εμφανίζεται ένα άλλο παράθυρο (εικόνα 5.16) στο οποίο συμπληρώνουμε τα πεδία που υπάρχουν ώστε να δημιουργηθεί ο λογαριασμός e-mail. Όταν συμπληρώσουμε τα κενά πεδία πατάμε ok, κλείνει το παράθυρο και το άλλο παράθυρο πατάμε close για να κλείσει και αυτό με τη σειρά του.

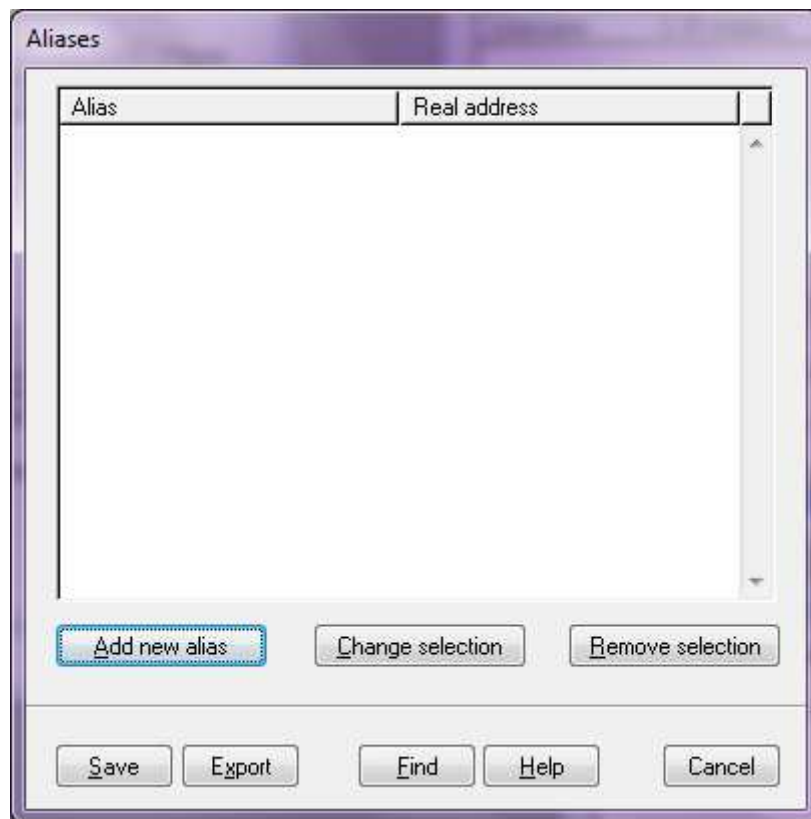


Εικόνα 5.15: Μενού διαχείρισης τοπικών χρηστών



Εικόνα 5.16: Μενού ιδιοτήτων χρήστη

Επιλέγουμε πάλι από το μενού **Configuration > Aliases**, εμφανίζεται ένα παράθυρο (εικόνα 5.17). Κάνουμε κλικ στο κουμπί **Add new alias**, εμφανίζεται ένα νέο παράθυρο στο οποίο συμπληρώνουμε τα κενά πεδία (εικόνα 5.18), έπειτα πατάμε **ok**, κλείνει το παράθυρο και στο άλλο παράθυρο πατάμε **save** το οποίο κλείνει και αυτό με τη σειρά του. Με αυτόν τον τρόπο δημιουργήσαμε το όνομα που θέλουμε να φαίνεται στο e-mail μας.



Εικόνα 5.17: Δημιουργία ψευδώνυμου e-mail



Εικόνα 5.18: Προσθήκη Ψευδώνυμου e-mail

5.2.2 Εκκίνηση λογισμικού Proseccing 2.0b5

Ανοίγουμε το λογισμικό Proseccing 2.0b5 πατώντας διπλό κλικ στο εικονίδιο Proseccing (εικόνα 5.19). Όταν τρέξει η εφαρμογή θα εμφανιστεί στην οθόνη του υπολογιστή ένα παράθυρο όπως φαίνεται στην εικόνα 5.20 όπου ανοίγει αυτόματα ένα κενό φύλλο εργασίας στο οποίο μπορούμε αμέσως να γράψουμε κώδικα στην γλώσσα προγραμματισμού Wiring.



Εικόνα 5.19: Εικονίδιο λογισμικού Proseccing



Εικόνα 5.20: Φύλλο εργασίας λογισμικού Proseccing

Ξεκινάμε να γράψουμε τον κώδικα. Οι εντολές του κώδικα της εικόνας 5.21 κάνουν το εξής: Ορίζουν από ποια πόρτα θα πάρει το πρόγραμμα τα δεδομένα (θέλουμε την ίδια πόρτα με αυτή που χρησιμοποιεί η ηλεκτρονική πλακέτα και το λογισμικό Arduino δηλαδή την COM3), δημιουργούν μία συνθήκη η οποία διαβάζει

και αποθηκεύει τα δεδομένα, αποκτά πρόσβαση στην βάση δεδομένων “test” και τέλος καταχωρεί τα δεδομένα στην βάση.

```

1  import de.bezier.data.sql.*;
2  import processing.serial.*;
3
4  Serial myPort;
5  float value,a,b;
6  String[] message=new String[5];
7  MySQL dbconnection;
8
9  void setup() {
10     myPort = new Serial(this, "COM3", 9600);
11 }
12 void draw(){
13     serialEvent();
14 }
15 void serialEvent()
16 {
17     for(int i=0;i<2;i++)
18     {delay(5000);
19     message[i] = myPort.readStringUntil(13);
20         if(message[i] != null)
21         {
22             value = float(message[i]);
23         if (i==0) {
24             a=value;
25             println("a=");
26         }
27         if (i==1){
28             b=value;
29             println("b=");
30         }
31         println(value);
32     }
33 }
34     String user    = "root";
35     String pass    = "";
36     String database = "test";
37     dbconnection = new MySQL( this, "localhost", database, user, pass );
38
39     if ( dbconnection.connect() ) {
40
41         dbconnection.execute( "INSERT INTO temp1 (Celsius,Hum) VALUES ('"+a+"','"+b+"')";
42         println();
43         println("ΑΠΟΣΤΟΛΗ ΔΕΔΟΜΕΝΩΝ!");
44     }
45 }

```

Εικόνα 5.21: Κώδικας επικοινωνίας ηλεκτρονικής πλακέτας Arduino υπο με βάση δεδομένων

Όταν τελειώσουμε με τον κώδικα επιλέγουμε από το μενού του Proseccing **sketch > run**, γίνεται έλεγχος του κώδικα και φόρτωση του προγράμματος. Εάν είναι επιτυχής τότε κάθε μέτρηση που γίνεται από του αισθητήρες καταχωρείτε στην βάση δεδομένων μας. Τέλος επιλέγουμε **File > save** και αποθηκεύουμε το αρχείο μας.

5.2.3 Γλώσσα συγγραφής σεναρίων PHP

Με την βοήθεια της γλώσσας συγγραφής σεναρίων PHP θα δημιουργήσουμε μία ιστοσελίδα στην οποία θα εμφανίζετε ένας πίνακας όπου θα καταχωρούνται οι τιμές από τις μετρήσεις των αισθητήρων (εικόνα 5.22) οι οποίες θα έχουν πριν καταχωρηθεί στην βάση δεδομένων. Όταν υπάρχει κάποια ακραία μέτρηση τότε η τιμή αυτή θα εμφανίζεται με έντονο χρώμα στον πίνακα και θα αποστέλλετε αυτομάτως ένα μήνυμα συναγερμού μέσω ηλεκτρονικού ταχυδρομείου.

ΗΜΕΡΟΜΗΝΙΑ	ΘΕΡΜΟΚΡΑΣΙΑ	ΥΓΡΑΣΙΑ
2013-05-10 12:04:37	23 °C	40 %
2013-05-10 12:04:47	24 °C	40 %
2013-05-10 12:06:53	24 °C	40 %
2013-05-10 12:07:03	23 °C	40 %
2013-05-10 12:07:13	66 °C	40 %
2013-05-10 12:07:23	26 °C	42 %
2013-05-10 12:07:33	27 °C	43 %
2013-05-10 12:07:43	29 °C	40 %
2013-05-10 12:07:54	28 °C	40 %
2013-05-10 12:08:04	27 °C	40 %
2013-05-10 12:08:14	26 °C	40 %
2013-05-10 12:08:24	26 °C	47 %
2013-05-10 12:32:10	24 °C	0 %
2013-05-10 12:32:21	24 °C	0 %
2013-05-10 12:32:31	24 °C	0 %
2013-05-10 12:32:41	24 °C	0 %
2013-05-10 12:32:51	24 °C	0 %
2013-05-10 12:33:56	24 °C	43 %
2013-05-10 12:34:06	24 °C	43 %
2013-05-10 12:34:16	24 °C	44 %
2013-05-10 12:34:26	24 °C	43 %
2013-05-10 12:34:36	24 °C	43 %
2013-05-10 12:34:46	24 °C	43 %

Εικόνα 5.22: Εμφάνιση μετρήσεων σε ιστοσελίδα

Για να έχουμε το αποτέλεσμα της εικόνας 5.22 θα πρέπει να γράψουμε κώδικα σε γλώσσα συγγραφής σεναρίων PHP. Επομένως πηγαίνουμε στον φάκελο που έχουμε εγκαταστήσει το λογισμικό Xampp και ακολουθούμε το εξής μονοπάτι: **C:\ > xampp > htdocs > example**. Στον φάκελο example δημιουργούμε ένα νέο έγγραφο κειμένου πατώντας δεξί κλικ > Δημιουργία > Έγγραφο κειμένου. Ανοίγουμε το έγγραφο κειμένου και γράφουμε τον κώδικα (εικόνα 5.23).

```

1  <?php
2  $host = "localhost";
3  $username = "root";
4  $password = "";
5  $database = "test";
6
7  mysql_connect($host, $username, $password) or die (mysql_error());
8  ?>

```

Εικόνα 5.23: Κώδικας καταχώρισης στοιχείων για πρόσβαση στην Βάση Δεδομένων

Με αυτόν τον κώδικα αποθηκεύουμε στις μεταβλητές τα στοιχεία που χρειάζονται για να μπορούμε να έχουμε πρόσβαση στην Βάση Δεδομένων. Αποθηκεύουμε το αρχείο και δημιουργούμε ένα νέο έγγραφο κειμένου στο οποίο θα γράψουμε πάλι κώδικα (εικόνα 5.24).

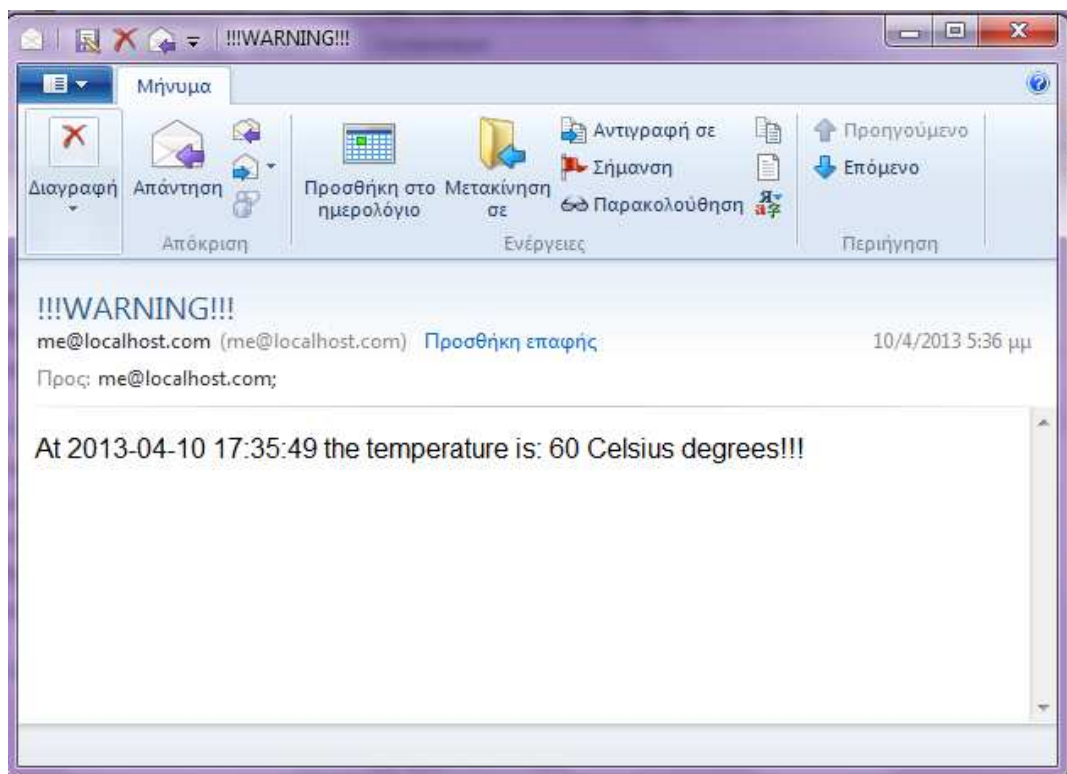
```

1 <html>
2 <body background=":\xampp\htdocs\example\app.png"
3 <table border="1" cellspacing="2" cellpadding="2" bgcolor="999999" align="center">
4 <tr bgcolor="#6297c5">
5 <th><font face="Arial"> ΗΜΕΡΟΜΗΝΙΑ </font></th>
6 <th><font face="Arial"> ΘΕΡΜΟΚΡΑΣΙΑ </font></th>
7 <th><font face="Arial"> ΥΓΡΑΣΙΑ </font></th>
8 </tr>
9 <?php
10 include "db_test_temperature.php";
11
12 echo('<meta http-equiv="refresh" content="5">');//αναζωωση
13
14
15 mysql_connect($host,$username,$password);
16 @mysql_select_db($database) or die("Δεν είναι δυνατή η επιλογή της βάσης δεδομένων.");
17 //αποστολη e-mail
18 $t=mysql_query("SELECT * FROM temp1 where Celsius>25");
19
20 while($r=mysql_fetch_assoc($t)
21 {
22 $p=($r["Celsius"]);
23 $a1=date('Y-m-d H:i:s');
24 $a2=($r["Time"]);
25 if (strtotime($a2) >= strtotime('+1 hour -5 seconds'))
26 $mailsent=mail("me@localhost.com","!!!WARNING!!!","At $a2 the temperature is: $p Celsius degrees!!!","me@localhost.com");
27 }
28 $sql = "SELECT * FROM `temp1`";
29 $rresult=mysql_query($sql);
30 $num=mysql_numrows($rresult);
31
32 mysql_close();
33 $i=0;
34 while ($i < $num) {
35 $Time=mysql_result($rresult,$i,"Time");
36 $Celsius=mysql_result($rresult,$i,"Celsius");
37 $Hum=mysql_result($rresult,$i,"Hum");
38
39 echo "<tr>";
40 echo "<td align=center>$Time</td>";
41 if($Celsius>25)
42 echo "<td align=center bgcolor=\"\#00ff00\"><b>$Celsius °C</b></td>";
43 else
44 echo "<td align=center><b>$Celsius °C</b></td>";
45 echo "<td align=center>$Hum &lt;/td>";
46 echo "</tr>";
47 $i++;
48 }
49 </table>
50 </body>
51 </html>

```

Εικόνα 5.24: Κώδικας δημιουργίας ιστοσελίδας

Οι εντολές αυτού του κώδικα δημιουργούν μια ιστοσελίδα στην οποία εμφανίζετε ένας πίνακας με τρεις στήλες: «Ημερομηνία», «Θερμοκρασία», «Υγρασία». Έπειτα ανακαλείτε το προηγούμενο πρόγραμμα που δημιουργήσαμε ,παίρνουμε τα στοιχεία που χρειάζονται και συνδεόμαστε στην Βάση Δεδομένων. Όταν συνδεθούμε στην ΒΔ ελέγχουμε μια μία τις εγγραφές ανά γραμμή. Αν υπάρχει κάποια ακραία μέτρηση στέλνεται ένα μήνυμα συναγερμού στο ηλεκτρονικό μας ταχυδρομείο (εικόνα 5.25). Στη συνέχεια αντιγράφονται όλες οι τιμές από τον πίνακα της Βάσης Δεδομένων και καταχωρούνται μια μία στον πίνακα της ιστοσελίδας που δημιουργήσαμε. Καθώς γίνονται οι καταχωρίσεις πραγματοποιείται έλεγχος ξανά για ακραίες μετρήσεις και εάν υπάρχουν τονίζετε το κελί της τιμής με έντονο χρώμα. Η ιστοσελίδα πραγματοποιεί αυτήν την διαδικασία που προαναφέραμε κάθε 5 δευτερόλεπτα ώστε να προστίθενται οι νέες μετρήσεις που πραγματοποιούνται με το πέρασμα του χρόνου. Τέλος αποθηκεύουμε το αρχείο. Για να μπορέσουμε να δούμε το αποτέλεσμα του κώδικα (εικόνα 5.22) ανοίγουμε το πρόγραμμα περιήγησης στο διαδίκτυο και γράφουμε την εξής διεύθυνση:
http://192.168.1.2/example/color_test.php



Εικόνα 5.25: Μήνυμα συναγερμού

5.3 Υλοποίηση 3^{ου} Βήματος

Σε αυτό το σημείο θα δημιουργήσουμε μία εφαρμογή Android με το πρόγραμμα Eclipse την οποία εν συνεχεία θα την εγκαταστήσουμε στο κινητό τηλέφωνο, καθώς και ένα αρχείο PHP το οποίο θα καταστήσει εφικτή την λειτουργία της σύνδεσης ανάμεσα στην Βάση Δεδομένων και στην εφαρμογή Android.

5.3.1 Δημιουργία αρχείου PHP

Το αρχείο PHP που θα δημιουργήσουμε θα πρέπει να περιέχει τις εξής δυνατότητες: Να έχει πρόσβαση στην βάση δεδομένων, να πραγματοποιεί έναν έλεγχο για ακραίες μετρήσεις και να στέλνει ένα μήνυμα συναγερμού με το ηλεκτρονικό ταχυδρομείο, καθώς επίσης κατά την ανάκληση του αρχείου από την εφαρμογή Android να έχει μία έξοδο όπου θα περιέχει τον πίνακα με τα στοιχεία από την βάση δεδομένων, έτσι ώστε να τα αντιγράψει η εφαρμογή και να τα εμφανίζει στην οθόνη του κινητού τηλεφώνου.

Σύμφωνα με τα παραπάνω για να επιτύχουμε το αποτέλεσμα που θέλουμε θα ακολουθήσουμε τα βήματα ως εξής: Πηγαίνουμε στον φάκελο που έχουμε εγκαταστήσει το λογισμικό Xampp και ακολουθούμε το εξής μονοπάτι: **C:\ > xampp > htdocs > example**. Στον φάκελο example δημιουργούμε ένα νέο έγγραφο κειμένου πατώντας δεξί κλικ > Δημιουργία > Έγγραφο κειμένου. Ανοίγουμε το έγγραφο κειμένου και γράφουμε τον κώδικα (εικόνα 5.26).

```

1 <?php
2 mysql_connect("localhost","root","");
3 mysql_select_db("test");
4 $t=mysql_query("SELECT * FROM temp1 where Celsius>25");
5
6 while($r=mysql_fetch_assoc($t))
7 {
8     $p=($r["Celsius"]);
9     $a1=date('Y-m-d H:i:s');
10    $a2=($r["Time"]);
11    if (strtotime($a2) >= strtotime('+1 hour -5 seconds'))
12    $mailsent=mail("me@localhost.com", "!!!WARNING!!!", "At $a2 the temperature is: $p Celsius degrees!!!", "me@localhost.com");
13 }
14
15     $q=mysql_query("SELECT * FROM temp1");
16 while($row=mysql_fetch_assoc($q))
17     $json_output[]=$row;
18 print(json_encode($json_output));
19 mysql_close();
20 ?>

```

Εικόνα 5.26: Κώδικας για την επικοινωνία ΒΔ- Android

Οι εντολές του κώδικα της εικόνας 5.26 πραγματοποιούν την σύνδεση καθώς και την πρόσβαση στην βάση δεδομένων, τον έλεγχο για ακραίες μετρήσεις και την

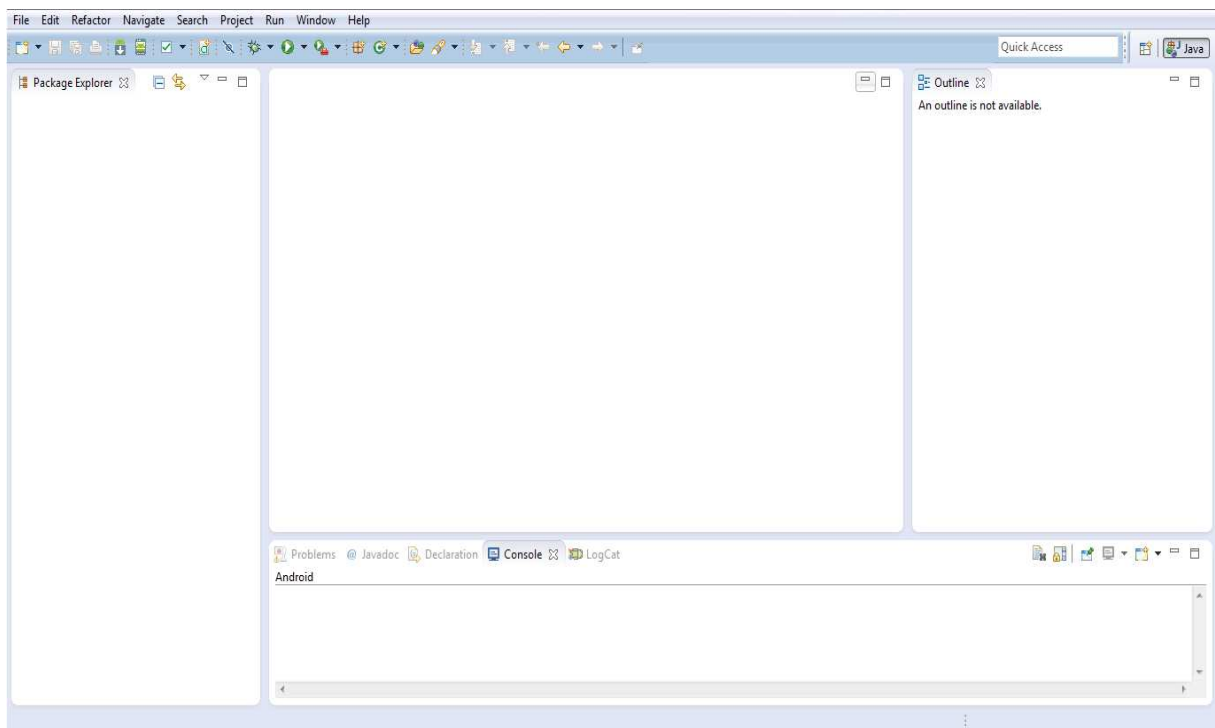
αποστολή μηνύματος συναγερμού με την χρήση του ηλεκτρονικού ταχυδρομείου, καθώς επίσης μία έξοδος όπου θα περιέχει τον πίνακα με τα στοιχεία από την βάση δεδομένων, έτσι ώστε να τα αντιγράψει η εφαρμογή Android όταν ανακαλεί το αρχείο και να τα εμφανίζει στην οθόνη του κινητού τηλεφώνου. Τέλος αποθηκεύουμε το αρχείο μας με όνομα « GetUsers.php».

5.3.2 Δημιουργία εφαρμογής Android

Για την δημιουργία της εφαρμογής Android θα χρησιμοποιήσουμε το λογισμικό Eclipse. Για να ανοίξουμε το Eclipse πατάμε διπλό κλικ στο εικονίδιο της εφαρμογής (εικόνα 5.27). Όταν τρέξει το πρόγραμμα ανοίγει ένα παράθυρο με διάφορα πάνελ (εικόνα 5.28)



Εικόνα 5.27: Εικονίδιο λογισμικού Eclipse

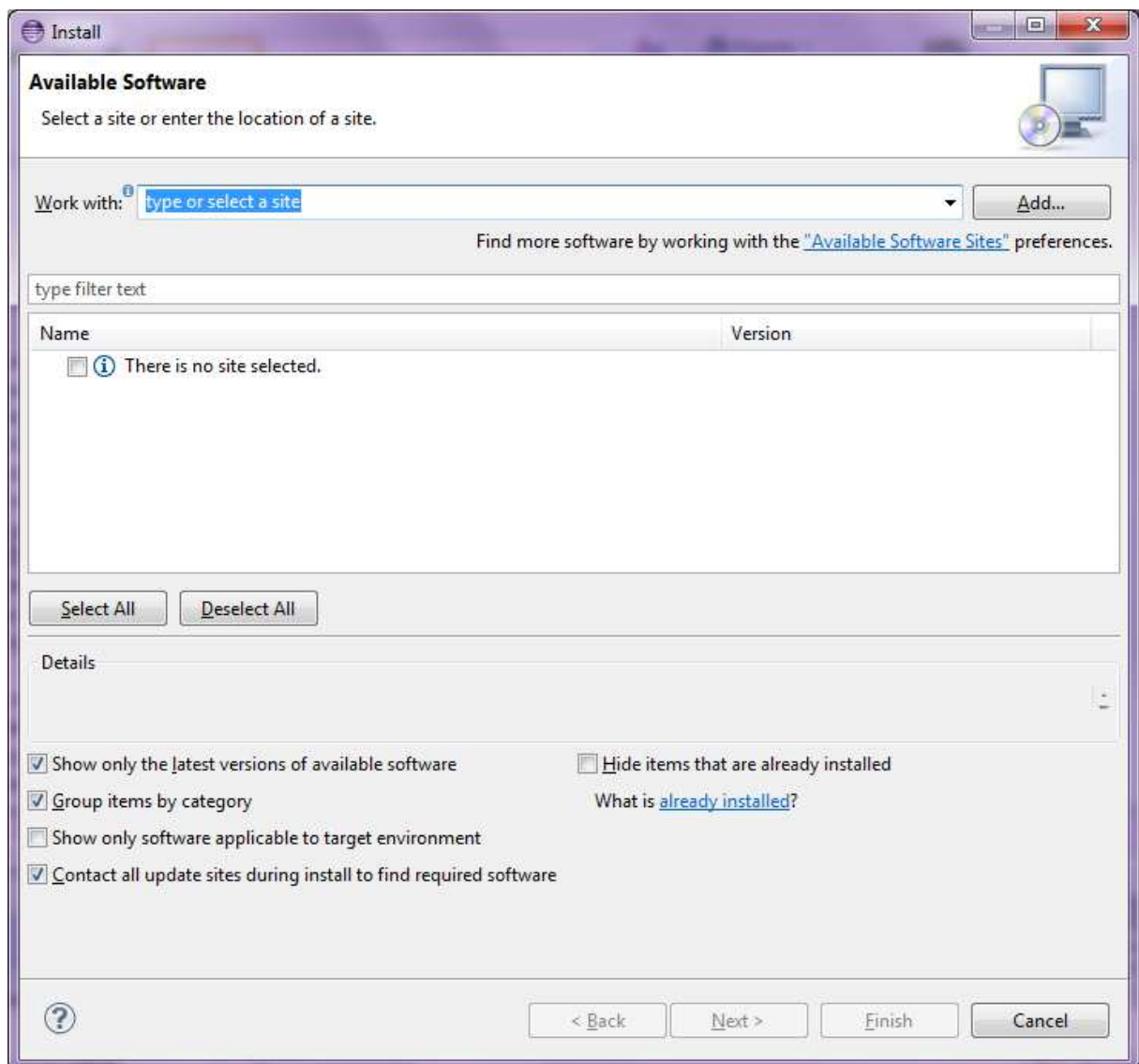


Εικόνα 5.28: Βασικό παράθυρο Eclipse

Στο επάνω μέρος βλέπουμε μια μπάρα με διάφορα μενού και από κάτω ακριβώς μια άλλη μπάρα με διάφορα κουμπιά συντομίας ορισμένων βασικών λειτουργιών. Το πάνελ που βρίσκετε στο αριστερό μέρος μας βοηθάει να βλέπουμε και να διαχειριζόμαστε τα Project που διαθέτουμε. Το κεντρικό και μεγαλύτερο πάνελ είναι αυτό που χρησιμοποιούμε για να γράψουμε κώδικα και σε κάποιες περιπτώσεις μας δίνετε η δυνατότητα να κάνουμε και αντικειμενοστραφή

προγραμματισμό. Στα δεξιά του κεντρικού πάνελ υπάρχει ένα άλλο το οποίο περιέχει τα διάφορα αντικείμενα, μεταβλητές, κλάσεις κτλ που περιέχει το κάθε Project το οποίο επεξεργαζόμαστε κάθε φορά. Τέλος κάτω από το κεντρικό πάνελ βρίσκετε αυτό το οποίο αναλόγως την ετικέτα που επιλέγουμε μας δίνει και διαφορετικές πληροφορίες. Κάποιες από αυτές είναι για παράδειγμα τα διάφορα errors και warnings που υπάρχουν στον κώδικα μας ή τα αποτελέσματα του προγράμματος σε μια σειριακή οθόνη μετά το τρέξιμο του κτλ.

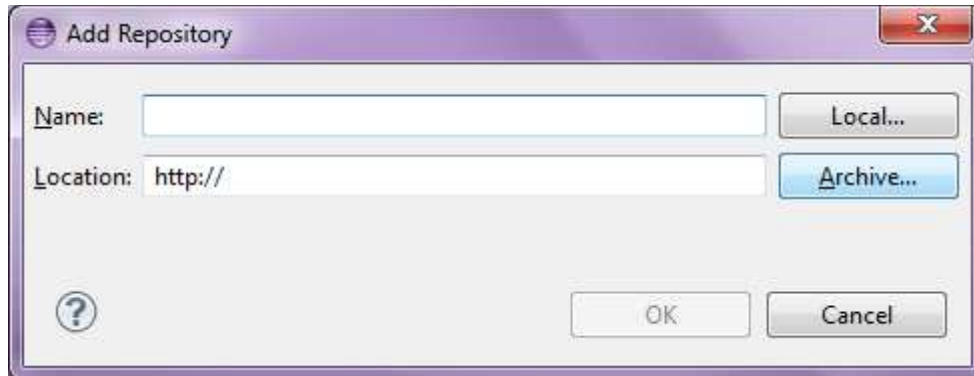
Αυτό που θα πρέπει να κάνουμε προτού δημιουργήσουμε το project μας είναι ενσωματώσουμε το λογισμικό ADT (Android Developer Tools) στο Eclipse ώστε να μπορούμε να μετατρέψουμε την εφαρμογή μας σε εφαρμογή Android. Άρα το πρώτο βήμα που κάνουμε είναι να επιλέξουμε από την μπάρα των βασικών λειτουργιών Help > Install New Software όπου εν συνεχεία θα εμφανιστεί το παράθυρο της εικόνας 5.29.



Εικόνα 5.29: Παράθυρο εγκατάστασης νέου λογισμικού

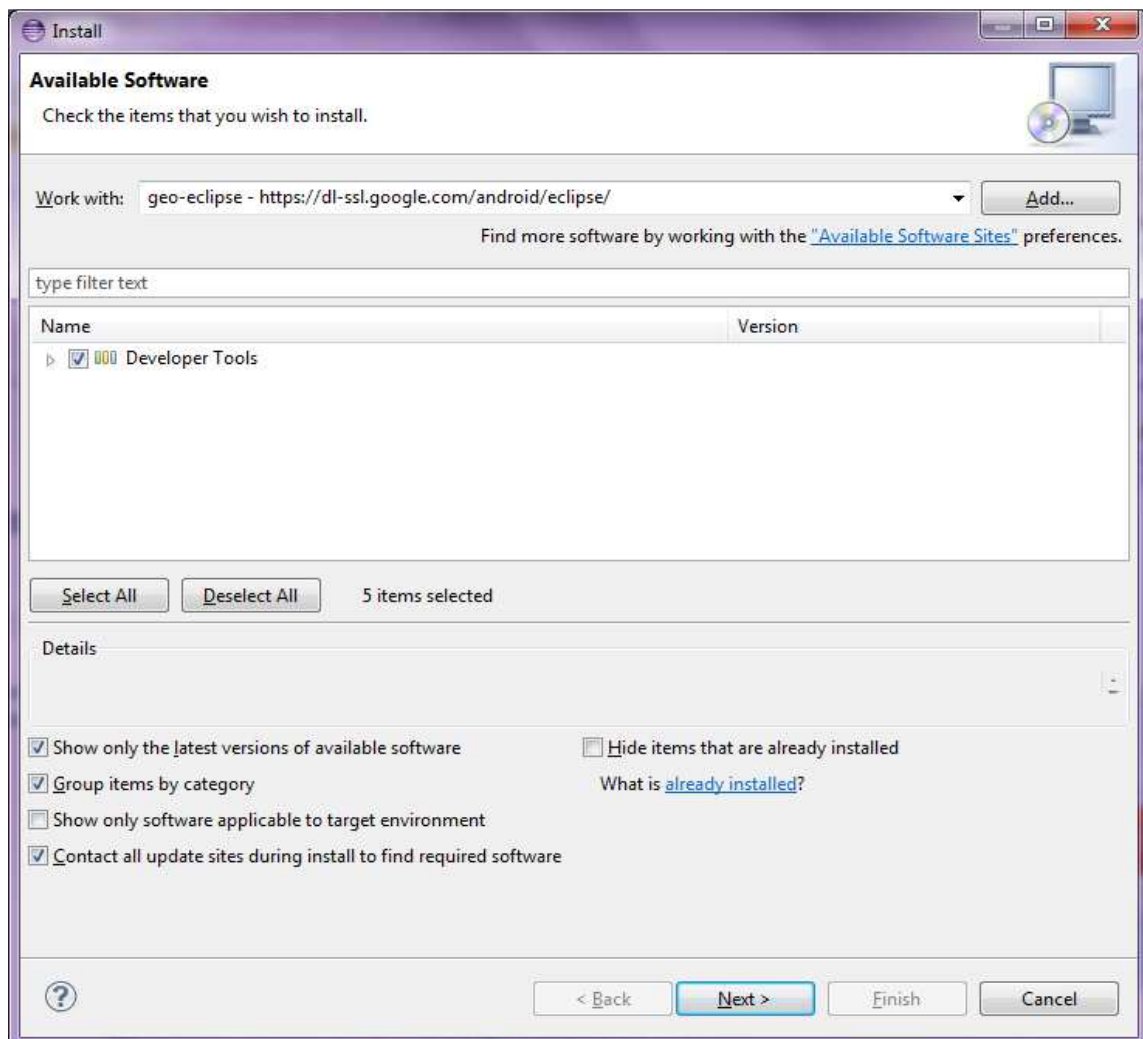
Σε αυτό το παράθυρο πατάμε κλικ στο κουμπί **add** που βρίσκετε πάνω δεξιά. Εμφανίζεται ένα νέο παράθυρο (εικόνα 5.30) και στο text box **Location** γράφουμε τον

εξής σύνδεσμο <https://dl-ssl.google.com/android/eclipse/> τον οποίο έχουμε βρει στην ιστοσελίδα <http://developer.android.com> που είναι και η επίσημη ιστοσελίδα προγραμματισμού Android και πατάμε το κουμπί ok.



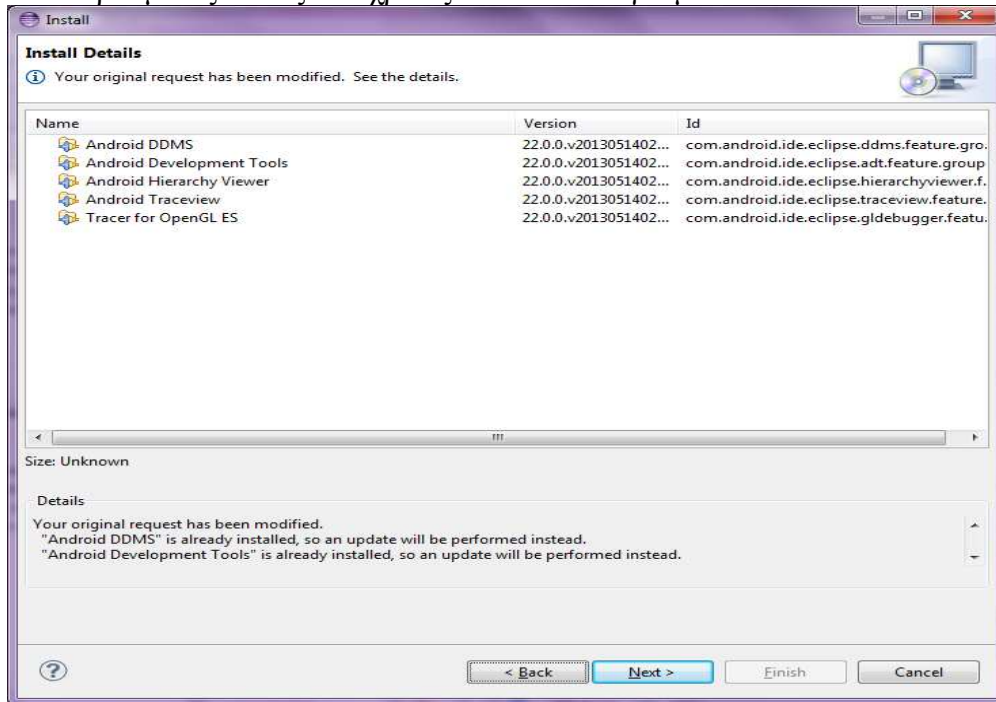
Εικόνα 5.30: Εισαγωγή αποθήκης

Επιστρέφοντας αυτόματα στο προηγούμενο παράθυρο, διαλέγουμε την επιλογή **Developer Tools** και πατάμε **next** (εικόνα 5.31).

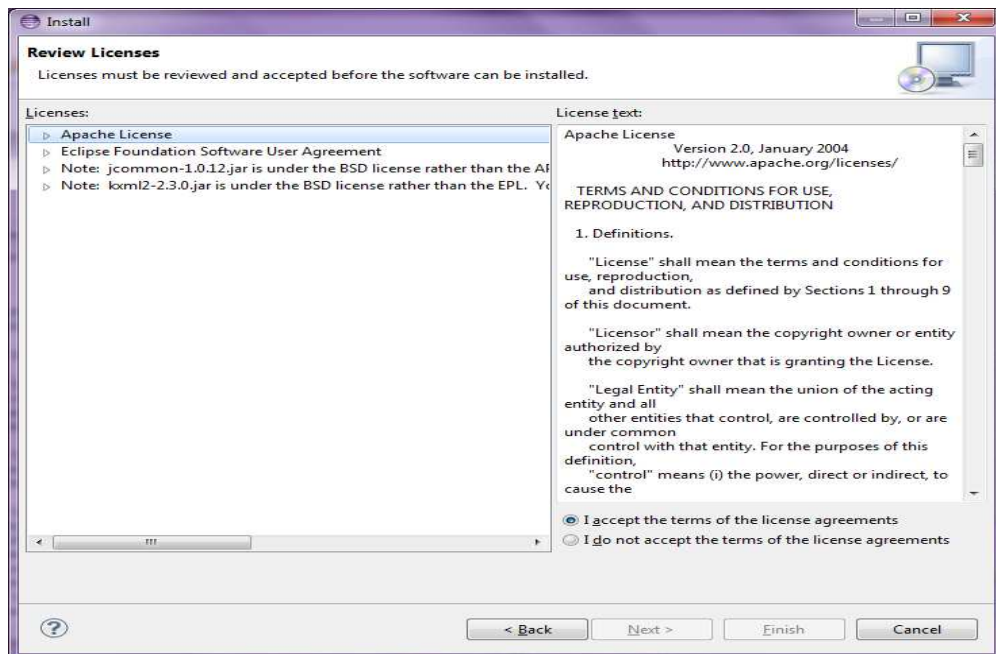


Εικόνα 5.31: Εισαγωγή λογισμικού ADT

Ανοίγει ένα νέο παράθυρο (εικόνα 5.32) στο οποίο μας δείχνει ποιά λογισμικά περιέχονται στο πακέτο ADT επιλέγουμε ξανά **next**, εμφανίζετε επίσης ένα άλλο παράθυρο (εικόνα 5.33) στο οποίο πρέπει να δεχτούμε τους όρους που αναγράφονται ώστε να πάρουμε τις άδειες που χρειάζεται και επιλέγουμε **Finish**.



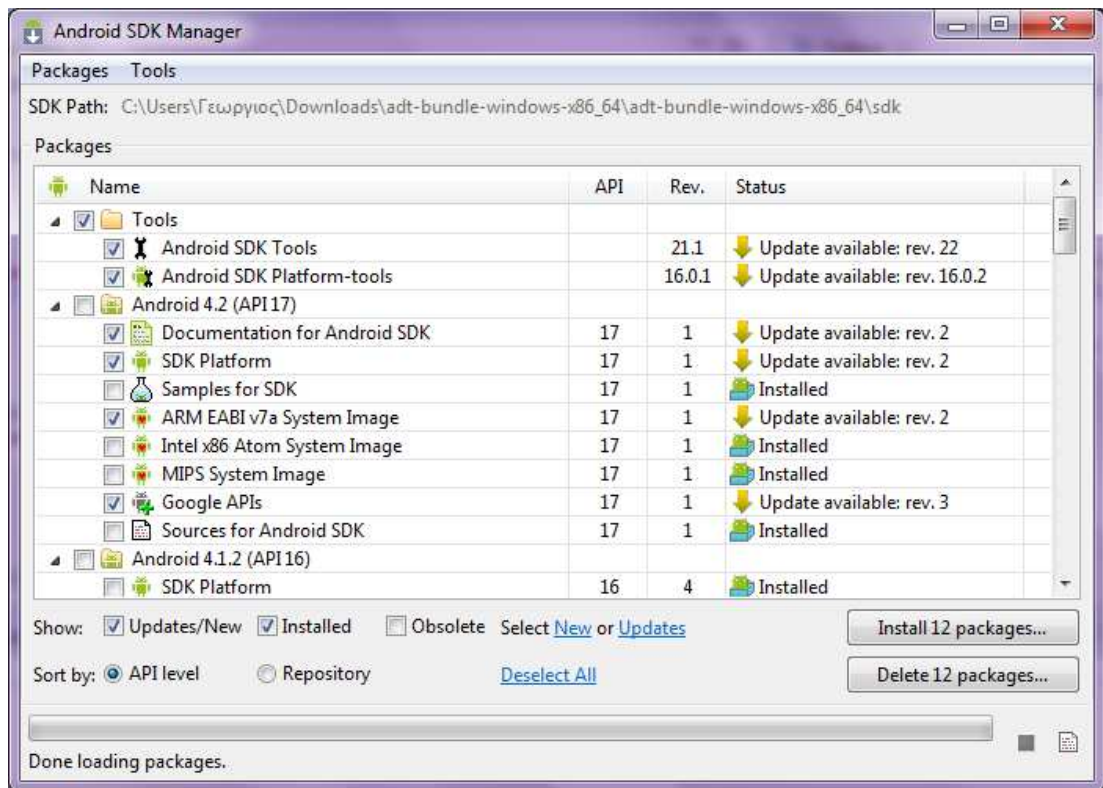
Εικόνα 5.32: Λίστα λογισμικών



Εικόνα 5.33: Παράθυρο όρων και αδειών

Εφόσον έχουμε εγκαταστήσει το λογισμικό ADT επιλέγουμε από την μπάρα των βασικών λειτουργιών Window > Android SDK Manager και εμφανίζετε ένα

παράθυρο διαχείρισης λογισμικών ανάπτυξης Android (εικόνα 5.34). Επιλέγουμε να εγκαταστήσουμε όλα τα πακέτα που υπάρχουν διαθέσιμα.



Εικόνα 5.34: Διαχείριση λογισμικών ανάπτυξης Android

Το επόμενο βήμα που πρέπει να κάνουμε είναι να δημιουργήσουμε ένα εικονικό SmartPhone (εικόνα 5.35) ώστε να πειραματιζόμαστε σε αυτό προτού εγκαταστήσουμε την εφαρμογή στο πραγματικό μας SmartPhone.

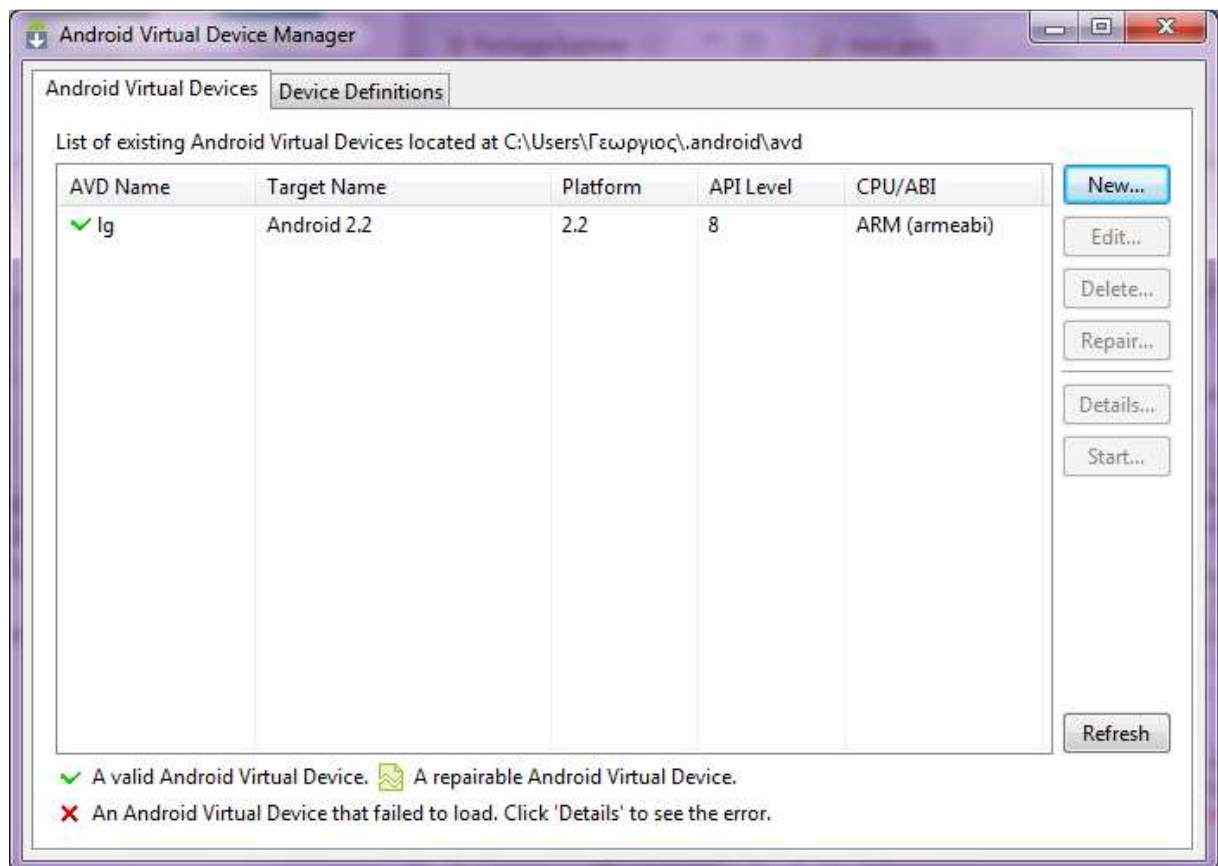


Εικόνα 5.35: Εικονικό SmartPhone

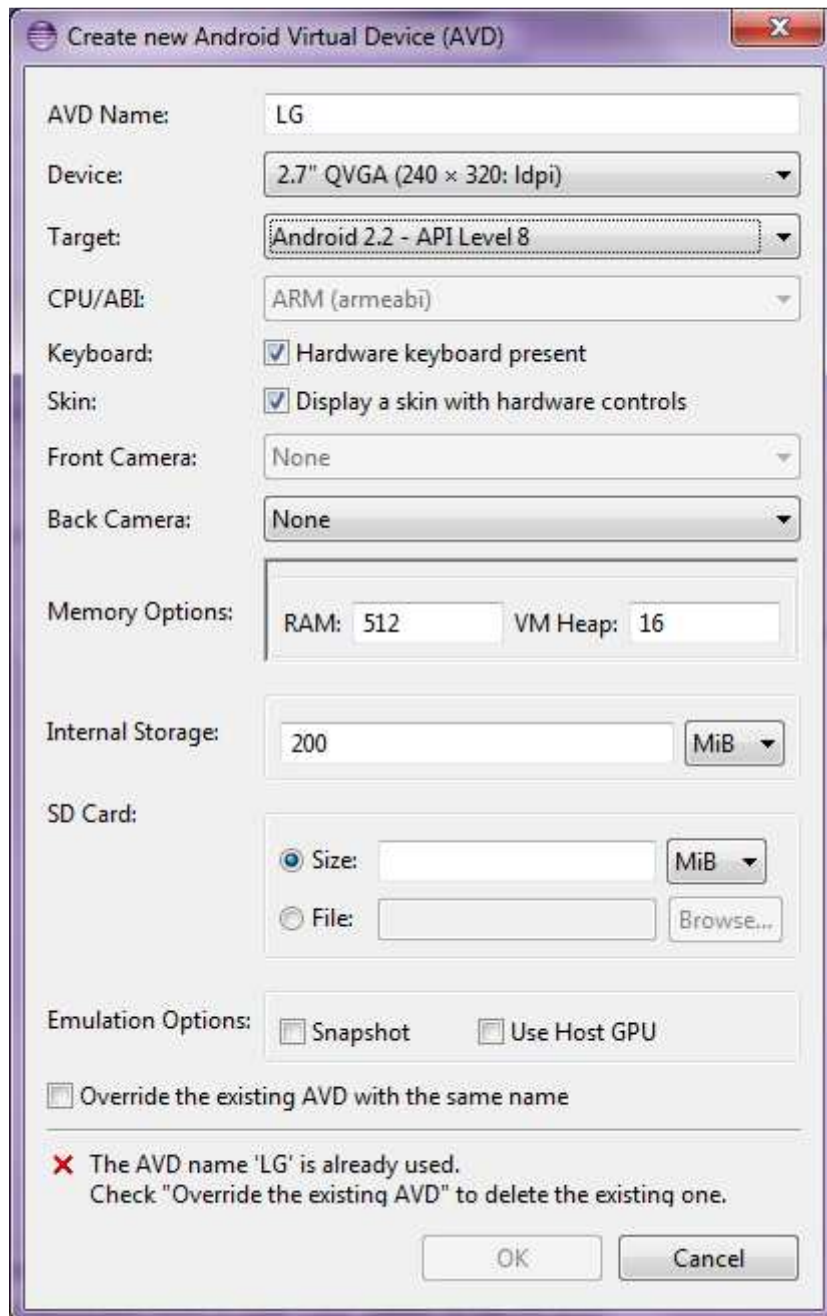
Για να το πραγματοποιήσουμε αυτό ακολουθούμε τα εξής βήματα:
Επιλέγουμε από την μπάρα βασικών λειτουργιών **Window > Android Virtual Device Manager** στο παράθυρο που εμφανίζετε (εικόνα 5.36) επιλέγουμε **New** και στο νέο παράθυρο που ανοίγει (εικόνα 5.37) κάνουμε τις εξής ρυθμίσεις :

- AVD Name: LG «όνομα εικονικής συσκευής»
- Device: 2.7'' QVGA (240 x 320 Idpi) «μέγεθος οθόνης»
- Target: Android 2.2 – API Level 8 «έκδοση λογισμικού»

Όλα τα υπόλοιπα τα αφήνουμε ως έχουν και πατάμε ok. Με αυτόν τον τρόπο έχουμε ολοκληρώσει την ενσωμάτωση του λογισμικού ADT και μπορούμε πλέον να ξεκινήσουμε το Project για την δημιουργία της εφαρμογής.

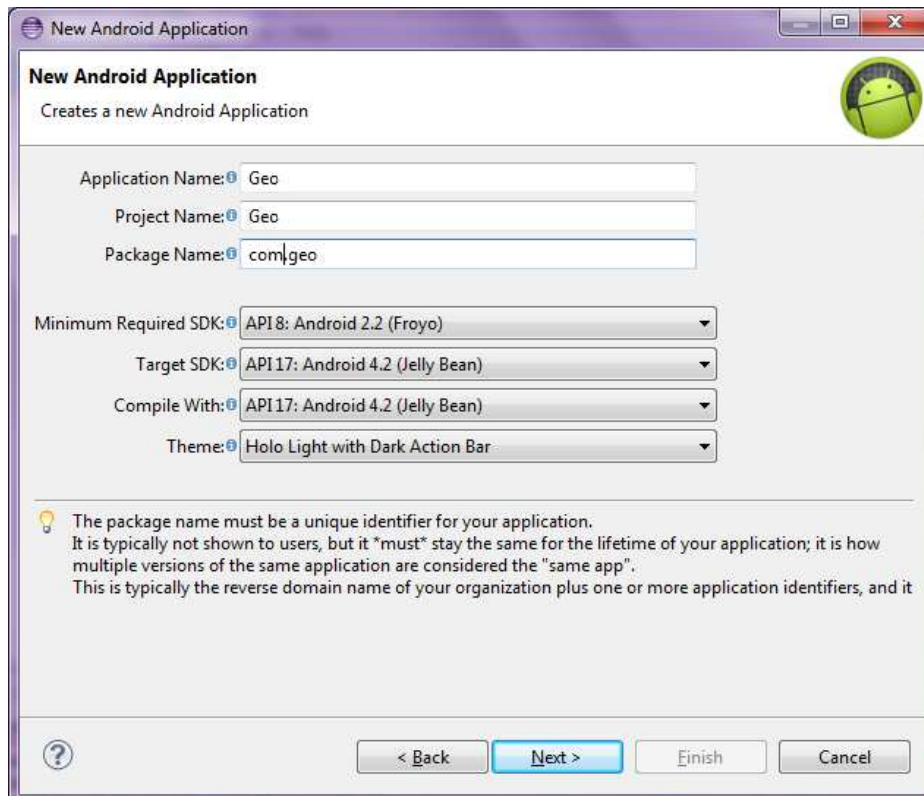


Εικόνα 5.36: Προσθήκη Εικονικού SmartPhone

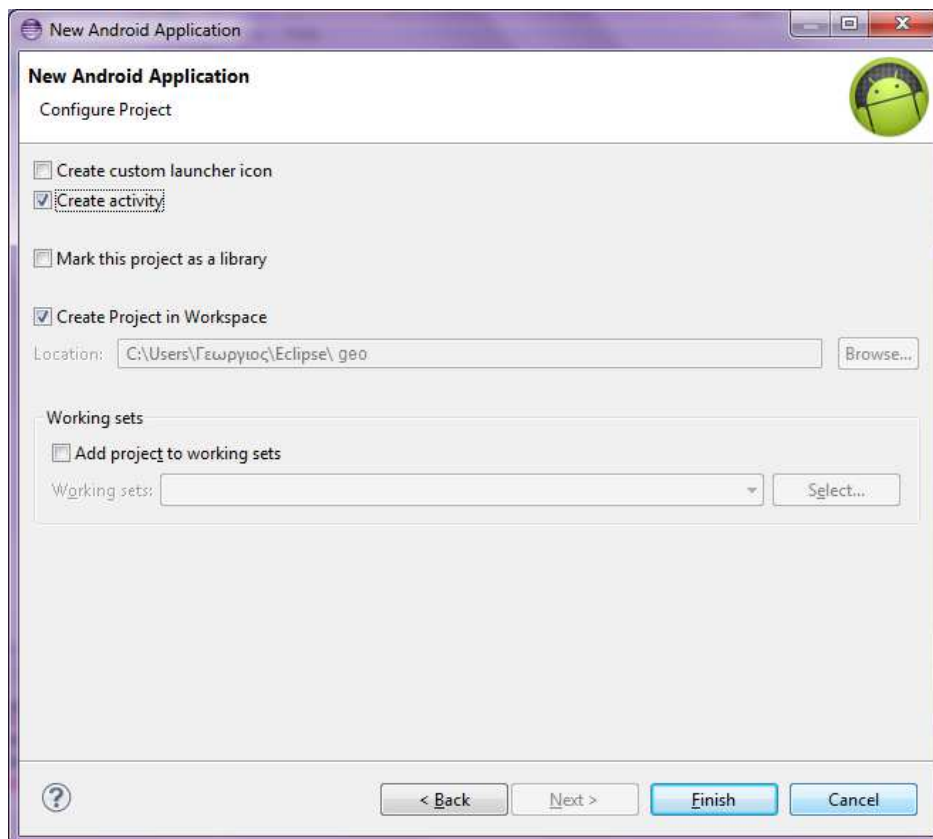


Εικόνα 5.37: Παράθυρο ιδιοτήτων εικονικού SmartPhone

Για να δημιουργήσουμε ένα νέο project επιλέγουμε **File > New > Other > Android Application Project** όπου εμφανίζετε ένα παράθυρο (εικόνα 5.38), δίνουμε όνομα πχ «geo» και πατάμε **next**. Στην επόμενη καρτέλα (εικόνα 5.39) επιλέγουμε τον προορισμό που επιθυμούμε να αποθηκευτεί το project και πατάμε **Finish**.



Εικόνα 5.38: Ονομασία Project



Εικόνα 5.39: Προορισμός αποθήκευσης project

Το project έχει δημιουργηθεί και μπορούμε να το δούμε στο πάνελ που βρίσκετε στο δεξιό μέρος του παραθύρου. Πατάμε διπλό κλικ πάνω στο project και επιλέγουμε **res > layout > activity_geo1.xml**. Βλέπουμε αυτομάτως στο κεντρικό πάνελ στην καρτέλα **activity_geo1.xml** τον κώδικα της εικόνας 5.40

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:orientation="vertical"
4      android:layout_width="fill_parent"
5      android:layout_height="fill_parent"
6  >
7
8      <TextView
9          android:layout_width="match_parent"
10         android:layout_height="wrap_content"
11         android:text="@string/hello_world"/>
12
13 </LinearLayout>

```

Εικόνα 5.40: Κώδικας δραστηριότητας οθόνης

Αυτός ο κώδικας είναι ο προγραμματισμός που γίνεται για τις δραστηριότητες της οθόνης δηλαδή αυτό που θα βλέπει ο χρήστης στην οθόνη του Smartphone. Εμείς θα μετατρέψουμε τον κώδικα αυτό ως εξής: Θα διαγράψουμε την εντολή **android:text="@string/hello_world"** και στην θέση της θα προσθέσουμε τις εξής εντολές:

- **android:id="@+id/textView1"**
- **android:scrollbarAlwaysDrawVerticalTrack="false"**
- **android:textColor="#000000"**

Η πρώτη εντολή δίνει όνομα `textView1` στο αντικείμενο τύπου `text View` για να μπορέσουμε να το χρησιμοποιήσουμε αργότερα στον κώδικα. Με την χρήση του αντικειμένου `textView1` θα έχουμε την δυνατότητα να εμφανίζουμε στην οθόνη `SmartPhone` τα δεδομένα που θα παίρνουμε από την βάση δεδομένων . Η δεύτερη εντολή προσθέτει κυλιόμενες κατακόρυφες μπάρες σε περίπτωση που είναι πάρα πολλά τα δεδομένα και δεν χωράνε στην οθόνη ,να έχουμε την δυνατότητα να σύρουμε προς τα επάνω η προς τα κάτω την οθόνη για να εμφανιστούν όλα τα δεδομένα. Τέλος η τρίτη εντολή κάνει τα γράμματα να φαίνονται με μαύρο χρώμα.

Έπειτα πατάμε διπλό κλικ πάνω στο project και επιλέγουμε **src > com.geo > Geo1.java**. Στο κεντρικό πάνελ πλέον μπορούμε να γράψουμε τον κώδικα ο οποίος θα δημιουργήσει την εφαρμογή Android. Γράφουμε τον κώδικα σε γλώσσα προγραμματισμού Java όπως φαίνετε στις εικόνες 5.41 , 5.42.

```

1 package com.new2;//ΟΝΟΜΑΣΙΑ ΠΑΚΕΤΟΥ
2
3 import java.io.BufferedReader;//ΜΕΘΟΔΟΣ ΠΟΥ ΔΙΑΒΑΖΕΙ ΧΑΡΑΚΤΗΡΕΣ
4 import java.io.InputStream; //Η ΒΑΣΙΚΗ ΜΕΘΟΔΟΣ ΟΛΩΝ ΤΩΝ ΕΙΣΟΔΩΝ ΤΗΣ JAVA
5 import java.io.InputStreamReader;// ΜΕΘΟΔΟΣ ΜΕΤΑΤΡΟΠΗΣ ΤΩΝ bytes ΣΕ ΧΑΡΑΚΤΗΡΕΣ
6 import java.util.ArrayList;// ΜΕΘΟΔΟΣ ΓΙΑ ΤΙΣ ΛΙΣΤΕΣ ΠΙΝΑΚΩΝ
7
8 import org.apache.http.HttpEntity;//ΜΕΘΟΔΟΣ ΑΙΤΗΣΗΣ Hypertext Transfer Protocol (HTTP)
9 import org.apache.http.HttpResponse;// ΜΕΘΟΔΟΣ ΑΠΟΚΡΙΣΗΣ HTTP
10 import org.apache.http.NameValuePair;//ΜΕΘΟΔΟΣ ΕΓΚΛΕΙΣΜΟΥ ΧΑΡΑΚΤΗΡΑ Η ΖΕΥΓΟΥΣ ΤΙΜΩΝ
11 import org.apache.http.client.HttpClient;//ΜΕΘΟΔΟΣ ΕΠΙΚΟΙΝΩΝΙΑΣ CLIENT-HTTP
12 import org.apache.http.client.entity.UrlEncodedFormEntity;//ΜΕΘΟΔΟΣ ΚΑΤΑΛΟΓΟΥ URL ΚΩΔΗΚΟΠΟΗΜΕΝΩΝ ΖΕΥΓΩΝ
13 import org.apache.http.client.methods.HttpPost;//ΜΕΘΟΔΟΣ ΓΙΑ ΤΗΝ ΑΟΣΤΟΛΗ ΜΙΑΣ ΑΙΤΗΣΗΣ POST HTTP
14 import org.apache.http.impl.client.DefaultHttpClient;//ΜΕΘΟΔΟΣ ΠΡΟΕΠΙΛΟΓΗΣ HTTP CLIENT
15 import org.apache.http.message.BasicNameValuePair;//ΜΕΘΟΔΟΣ ΕΓΚΛΕΙΣΜΟΥ ΧΑΡΑΚΤΗΡΑ Η ΖΕΥΓΟΥΣ ΤΙΜΩΝ
16 import org.json.JSONArray;//ΜΕΘΟΔΟΣ ΕΥΡΕΤΗΡΙΟΥ ΑΚΟΛΟΥΘΙΑΣ ΤΙΜΩΝ
17 import org.json.JSONException;//ΜΕΘΟΔΟΣ ΥΠΟΔΕΙΞΗΣ ΠΡΟΒΛΗΜΑΤΩΝ
18 import org.json.JSONObject; //ΜΕΘΟΔΟΣ ΤΡΟΠΟΠΟΙΗΣΗΣ ΤΙΜΩΝ
19
20 import android.app.Activity;//ΜΕΘΟΔΟΣ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ ΑΛΗΛΕΠΙΔΡΑΣΗΣ ΜΕ ΤΟ ΧΡΗΣΤΗ
21 import android.os.Bundle;//ΜΕΘΟΔΟΣ ΑΝΤΙΣΤΟΙΧΙΑΣ STRING ΣΕ ΔΙΑΦΟΡΟΥΣ ΤΥΠΟΥΣ ΔΕΔΟΜΕΝΩΝ
22 import android.os.Handler;//ΜΕΘΟΔΟΣ ΕΙΣΑΓΩΓΗΣ HANDLER ΓΙΑ ΕΠΕΞΕΡΓΑΣΙΑ ΜΥΝΗΜΑΤΩΝ ΚΑΙ ΑΝΤΙΚΕΙΜΕΝΩΝ
23 import android.text.method.ScrollingMovementMethod;//ΜΕΘΟΔΟΣ ΛΕΙΤΟΥΡΓΙΑΣ ΚΥΛΗΣΗΣ ΚΕΙΜΕΝΟΥ
24 import android.util.Log;//ΜΕΘΟΔΟΣ ΑΡΧΕΙΟΥ ΚΑΤΑΓΡΑΦΗΣ ΕΞΟΔΟΥ
25 import android.widget.LinearLayout;//ΜΕΘΟΔΟΣ ΡΥΘΜΙΣΗΣ ΓΡΑΜΜΙΚΗΣ ΔΙΑΤΑΞΗΣ
26 import android.widget.TextView;//ΜΕΘΟΔΟΣ ΕΜΦΑΝΙΣΗΣ ΚΕΙΜΕΝΟΥ ΣΤΟ ΧΡΗΣΤΗ
27
28 public class New3 extends Activity {//ΚΛΑΣΗ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ New3
29
30     String key;//ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
31     TextView txt;//ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
32     Handler handler = new Handler();//ΔΗΛΩΣΗ HANDLER
33     Runnable Refresh;//ΔΗΛΩΣΗ ΕΠΑΝΑΛΥΨΗΣ-ΑΝΑΝΕΩΣΗΣ
34     @Override
35     protected void onCreate(Bundle savedInstanceState) {{//ΣΥΝΑΡΤΗΣΗ onCreate ΔΗΜΙΟΥΡΓΕΙΤΑΙ ΑΥΤΟΜΑΤΑ
36         super.onCreate(savedInstanceState);// ΑΚΟΛΟΥΘΙΑ ΚΛΗΣΕΩΝ
37         setContentView(R.layout.activity_new3);// ΕΜΦΑΝΙΣΗ ΤΗΣ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ new3
38
39         Bundle extras = getIntent().getExtras();//ΜΕΤΑΦΟΡΑ ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΤΗΝ "MainActivity"
40         if (extras != null) {
41             key = extras.getString("met");
42         }
43
44         LinearLayout rootLayout = new LinearLayout(getApplicationContext()); //ΧΡΗΣΗΜΟΠΟΙΗΣΗ ΤΗΣ ΓΡΑΜΜΙΚΗΣ ΔΙΑΤΑΞΗΣ
45         txt = new TextView(getApplicationContext()); //ΑΝΑΘΕΣΗ ΣΤΟ txt ΤΟ TextView ΤΗΣ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ geol
46         rootLayout.addView(txt); //ΚΑΤΑΧΩΡΗΣΗ ΤΟΥ txt ΣΤΗΝ ΓΡΑΜΜΙΚΗ ΔΙΑΤΑΞΗ
47         setContentView(rootLayout); //ΕΜΦΑΝΙΣΗ ΤΗΣ ΓΡΑΜΜΙΚΗΣ ΔΙΑΤΑΞΗΣ
48
49         txt.setText("Connecting..."); //ΚΑΤΑΧΩΡΗΣΗ ΧΑΡΑΚΤΗΡΩΝ ΣΤΟ txt ΚΑΙ ΕΜΦΑΝΙΣΗ ΣΤΗΝ ΟΘΟΝΗ
50
51         Refresh = new Runnable() {{//ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
52             public void run() {{//ΔΗΛΩΣΗ ΣΥΝΑΡΤΗΣΗΣ
53
54                 txt.setText(getServerData(key)); //ΕΜΦΑΝΙΣΗ ΑΠΟΤΕΛΕΣΜΑΤΟΣ ΤΗΣ ΣΥΝΑΡΤΗΣΗΣ getServerData ΣΤΗΝ ΟΘΟΝΗ
55                 txt.setMovementMethod(new ScrollingMovementMethod());//ΧΡΗΣΗΜΟΠΟΙΗΣΗ ΚΥΛΗΣΗΣ ΚΕΙΜΕΝΟΥ
56
57                 handler.postDelayed(Refresh, 5000);//ΚΑΤΑΧΩΡΗΣΗ ΑΝΑΝΕΩΣΗΣ ΑΝΑ 5"
58             }
59         };
60         handler.post(Refresh);//ΑΝΑΝΕΩΣΗ
61     }

```

Εικόνα 5.41: Κώδικας για την δημιουργία εφαρμογής Android

```

62 private String getServerData(String returnString) {
63     InputStream is = null;
64     String result = "";
65     try{
66         HttpClient httpclient = new DefaultHttpClient();
67         HttpPost httppost = new HttpPost(key );
68         HttpResponse response = httpclient.execute(httppost);
69         HttpEntity entity = response.getEntity();
70         is = entity.getContent();
71     }catch(Exception e){
72         Log.e("log_tag", "Error in http connection "+e.toString());
73     }
74     try{
75         BufferedReader reader = new BufferedReader(new InputStreamReader(is,"iso-8859-1"),8);
76         StringBuilder sb = new StringBuilder();
77         String line = null;
78         while ((line = reader.readLine()) != null) {
79             sb.append(line + "\n");
80         }
81         is.close();
82         result=sb.toString();
83     }catch(Exception e){
84         Log.e("log_tag", "Error converting result "+e.toString());
85     }
86     //parse json data
87     try{
88         JSONArray jArray = new JSONArray(result);
89         for(int i=0;i<jArray.length();i++){
90             JSONObject json_data = jArray.getJSONObject(i);
91             Log.i("log_tag"," id: "+json_data.getInt("id")+", Time: "+json_data.getString("Time")+
92                 ", Celsius: "+json_data.getInt("Celsius")+", Hum: "+json_data.getInt("Hum"));
93             returnString += "\n\t" + jArray.getJSONObject(i);
94         }
95     }catch(JSONException e){
96         Log.e("log_tag", "Error parsing data "+e.toString());
97     }
98     return returnString;
99 }

```

Εικόνα 5.42: Κώδικας για την δημιουργία εφαρμογής Android

Οι εντολές αυτού του κώδικα κάνουν την εφαρμογή να συνδέετε μέσω του αρχείου «GetUsers.php» στην βάση δεδομένων, να παίρνει τα δεδομένα του πίνακα «temp1» από το αρχείο «GetUsers.php», να τα αποθηκεύει σε κάποιες μεταβλητές που έχουμε ορίσει και να τα εμφανίζει μέσω του «TextView1» στην οθόνη του SmartPhone. Όλη αυτή η διαδικασία επαναλαμβάνετε κάθε 5 δευτερόλεπτα.

Το επόμενο βήμα που πρέπει να κάνουμε είναι να δημιουργήσουμε ένα Android Application Project με την ίδια διαδικασία που δημιουργήσαμε το **Geo** και το οποίο θα ονομάσουμε **New 2**. Μόλις δημιουργήσουμε το project μπορούμε να το δούμε στο πάνελ που βρίσκετε στο δεξιό μέρος του παραθύρου. Πατάμε διπλό κλικ πάνω στο project και επιλέγουμε **res > layout**, πατάμε δεξί κλικ και **επιλέγουμε New>Android Activity** και επιλέγουμε το project **Geo**. Με αυτόν τον τρόπο καταφέραμε να συνδέσουμε τα δύο project μεταξύ τους. Στην καρτέλα **activity_main.xml** γράφουμε τον κώδικα της εικόνας 5.43

```
activity_main.xml ☒
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity" >

    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="156dp"
        android:text="@string/Connect" />

    <EditText
        android:id="@+id/editText1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_above="@+id/button1"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="62dp"
        android:ems="10"
        android:inputType="text"
        android:text="http://192.168.1.2/example/GetUsers.php" >

        <requestFocus />
    </EditText>

</RelativeLayout>
```

Εικόνα 5.43: Κώδικας δραστηριότητας οθόνης

Έπειτα πατάμε διπλό κλικ πάνω στο project και επιλέγουμε **src > com> New2 > MainActivity.java**. Στο κεντρικό πάνελ πλέον μπορούμε να γράψουμε τον κώδικα σε γλώσσα προγραμματισμού Java όπως φαίνετε στην εικόνα 5.44

```

MainActivity.java
package com.new2; //ΟΝΟΜΑΣΙΑ ΠΑΚΕΤΟΥ

import android.os.Bundle; //ΜΕΘΟΔΟΣ ΑΝΤΙΣΤΟΙΧΙΑΣ STRING ΣΕ ΔΙΑΦΕΡΟΥΣ ΤΥΠΟΥΣ ΔΕΔΟΜΕΝΩΝ

public class MainActivity extends Activity { //ΚΛΑΣΗ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ MainActivity

    EditText mEditText; //ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
    String key=""; //ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
    Button mButton; //ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ

    @Override
    protected void onCreate(Bundle savedInstanceState) { //ΣΥΝΑΡΤΗΣΗ onCreate ΔΗΜΙΟΥΡΓΕΙΤΑΙ ΑΥΤΟΜΑΤΑ
        super.onCreate(savedInstanceState); // ΑΚΟΛΟΥΘΙΑ ΚΛΗΣΕΩΝ
        setContentView(R.layout.activity_main); // ΕΜΦΑΝΗΣΗ ΤΗΣ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ new3

        mEditText=(EditText)findViewById(R.id.editText1); //ΑΝΤΙΣΤΟΙΧΙΑ ΜΕΤΑΒΛΗΤΗΣ
        mButton=(Button)findViewById(R.id.button1); //ΑΝΤΙΣΤΟΙΧΙΑ ΜΕΤΑΒΛΗΤΗΣ
        mButton.setOnClickListener(// ΜΕ ΤΟ ΠΑΤΗΜΑ ΤΟΥ ΚΟΥΜΠΙΟΥ ΠΕΡΝΕΙ ΟΤΙ ΓΡΑΦΕΙ ΤΟ EDIT TEXT,
            //ΤΟ ΑΠΟΘΗΚΕΥΕΙ ΣΤΗΝ ΜΕΤΑΒΛΗΤΗ met ΚΑΙ ΤΟ ΣΤΕΛΝΕΙ ΣΤΗΝ ΔΡΑΣΤΗΡΙΟΤΗΤΑ New3,
            //ΑΝΟΙΓΕΙ ΤΗΝ ΔΡΑΣΤΗΡΙΟΤΗΤΑ New3
            new View.OnClickListener(){
                public void onClick(View view)
                {
                    key=mEditText.getText().toString();

                    Intent i = new Intent(getApplicationContext(), New3.class);
                    i.putExtra("met",key);
                    startActivity(i);
                }
            });
    }
}

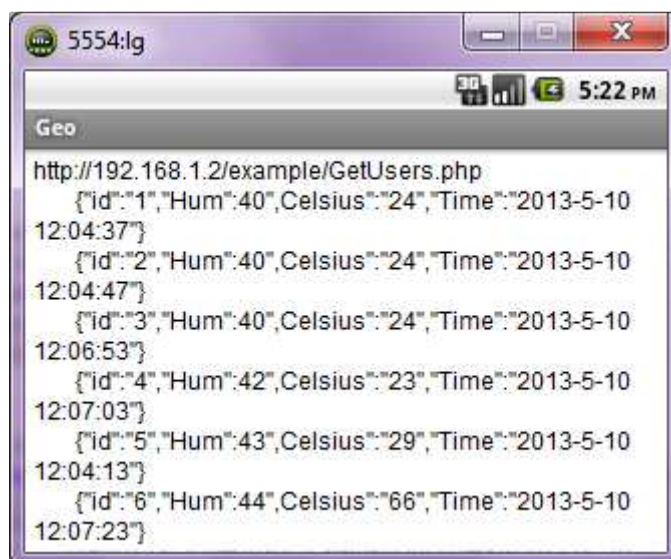
```

Εικόνα 5.44: Κώδικας για την δημιουργία εφαρμογής Android

Για να δούμε εάν λειτουργεί σωστά η εφαρμογή επιλέγουμε από την μπάρα βασικών λειτουργιών **Run > Run**. Το αποτέλεσμα φαίνεται στις εικόνες 5.45, 5.46.



Εικόνα 5.45: Ένδειξη δεδομένων εικονικού SmartPhone



Εικόνα 5.46: Ένδειξη δεδομένων εικονικού SmartPhone

6 Επίλογος

Αφού ολοκληρώθηκε όλη η πλατφόρμα επιτυχώς αξίζει να σημειωθεί ότι ένα μεγάλο ποσοστό των πληροφοριών που χρειάστηκαν για την υλοποίηση της πτυχιακής εργασίας βρέθηκε μέσα από το διαδίκτυο διότι τα περισσότερα προγράμματα που χρησιμοποιήθηκαν δεν μου ήταν γνωστά. Αυτό που παρατήρησα όμως με μεγάλη μου έκπληξη και ευχαρίστηση είναι ότι κατάφερα σε πολύ μικρό χρονικό διάστημα να κατανοήσω την λειτουργία τους ,παρατηρώντας ότι όλα βασίζονται σε γλώσσες προγραμματισμού και γενικά στην ίδια τακτική και νοοτροπία με την οποία διδάχθηκα όλα αυτά τα χρόνια στο ΑΤΕΙ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΠΟΛΥΜΕΣΩΝ.

Τα προβλήματα που προέκυψαν προερχόταν κυρίως από τις ελλείψεις γνώσεις μου στο κομμάτι της σύνδεσης όλων των λογισμικών μεταξύ τους, έτσι ώστε να προγραμματιστούν και να λειτουργήσουν αρμονικά το ένα με το άλλο και να έχω στο τέλος το αποτέλεσμα το οποίο απαιτούσε η συγκεκριμένη πτυχιακή εργασία. Ένα άλλο σημαντικό πρόβλημα ήταν το ότι η πηγή πληροφοριών προέρχεται στο μεγαλύτερο ποσοστό της από το διαδίκτυο και μάλιστα στη αγγλική γλώσσα λόγω του ότι δεν υπάρχει δωρεάν η αντίστοιχη βιβλιογραφία .

Η χρησιμότητα αυτής της πτυχιακής εργασίας ποικίλει αν αναλογιστεί κανείς τα θετικά πράγματα που μπορούν να προσφέρουν στην κοινωνία και στην βιομηχανία η ενασχόληση και η ανάπτυξη τέτοιου είδους εφαρμογών. Τέτοιες παρεμφερής εφαρμογές βλέπουμε και αναπτύσσονται ευρέως στο εξωτερικό π.χ. στην γεωργία, σε υπηρεσίες για επιτήρηση πυρκαγιών, σε μηχανές που χρησιμοποιούνται στην βιομηχανία ,σε σύστημα άρδευσης κτλ

Ο λόγος που επέλεξα αυτήν την πτυχιακή εργασία ήταν για να γνωρίσω πως θα μπορέσω να υλοποιήσω τις θεωρητικές και πρακτικές γνώσεις που έχω αποκομίσει από την σχολή μας και για να μάθω μία νέα τεχνολογία ,συνεχώς εξελισσόμενη στις μέρες μας ,τις εφαρμογές Android.

6.1 Μελλοντική εργασία και επεκτάσεις

Η πλατφόρμα αυτή μπορεί να επεκταθεί σε διάφορους τομείς. Ένα παράδειγμα θα ήταν να προστεθούν επιπλέον αισθητήρες οι οποίοι με κάποιο πρόσθετο υλικό θα έχουν την δυνατότητα να συνδέονται ασύρματα στην πλατφόρμα για να μπορούμε έτσι να καλύψουμε μία μεγάλη έκταση και να την ελέγξουμε (πχ ένα δάσος). Η ενσωμάτωση διαφορετικών αισθητήρων όπως είναι οι αισθητήρες μέτρησης θερμοκρασίας , υγρασίας, ατμοσφαιρικής πίεσης, ταχύτητας , διεύθυνσης του ανέμου κτλ θα μπορούσε να μετατρέψει την εφαρμογή σε ένα μετεωρολογικό σταθμό. Επίσης μία μελλοντική εξέλιξη της πλατφόρμας θα ήταν να χρησιμοποιηθεί σε διάφορα συστήματα αυτοματισμού όπως για παράδειγμα σε ένα σύστημα αυτόματης άρδευσης ή σε αυτοματισμούς που αφορούν την κεντρική διαχείριση διαφόρων συστημάτων μιας κατοικίας (έξυπνο σπίτι) κτλ.

Βιβλιογραφία

- [1] <http://developer.android.com/reference/packages.html>
- [2] <http://www.tutorialspoint.com/java/>
- [3] <http://www.w3schools.com>
- [4] <http://wiring.org.co/learning/basics/>
- [5] <http://arduino.cc/en/Tutorial/HomePage>
- [6] <http://processing.org/learning/basics/>
- [7] <http://el.wikipedia.org>
- [8] <http://www.apachefriends.org/en/xampp.html>
- [9] <http://www.tutorialspoint.com/eclipse/>
- [10] <http://www.tutorialspoint.com/mysql/index.htm>
- [11] <http://www.tutorialspoint.com/php/index.htm>
- [12] <http://www.tutorialspoint.com/html/index.htm>
- [13] <http://www.tutorialspoint.com/javascript/index.htm>
- [14] <http://www.tutorialspoint.com/cplusplus/index.htm>
- [15] <http://www.welcom-project.ipet.gr>
- [16] <http://www.mediamarkt.gr/mp/article/Smartphone,872008.html>
- [17] <http://greek.gsm-gprs-modem.com>

Παράρτημα

Παρουσίαση Πτυχιακής

Ανάπτυξη πλατφόρμας Ασύρματου Δικτύου Αισθητήρων (WSN) για την συλλογή περιβαλλοντικών δεδομένων και επεξεργασία με εφαρμογή Android

Βασιλικός Γεώργιος AM : 1362

Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Εφαρμοσμένης Πληροφορικής και Πολυμέσων

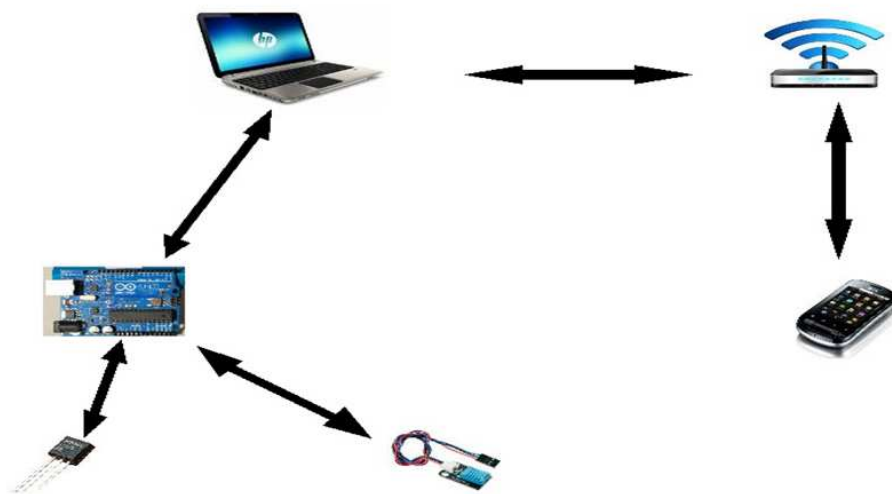
Επιβλέπων καθηγητής: Βλησίδης Ανδρέας



Εισαγωγή

Αντικείμενο της παρούσας πτυχιακής εργασίας είναι η ανάπτυξη πλατφόρμας Ασύρματου Δικτύου Αισθητήρων (WSN) για την συλλογή περιβαλλοντικών δεδομένων και επεξεργασία με εφαρμογή Android. Αυτή η τεχνολογία μας παρέχει την δυνατότητα να αποκτήσουμε σε πραγματικό χρόνο διάφορα περιβαλλοντικά δεδομένα τα οποία μπορούμε να τα επεξεργαστούμε άμεσα χωρίς να είναι απαραίτητο να είμαστε στο ίδιο γεωγραφικό σημείο από το οποίο έγινε η συλλογή τους. Επίσης η πλατφόρμα αυτή έχει την επιπλέον δυνατότητα να καταχωρεί τα δεδομένα σε μια βάση δεδομένων ώστε να μπορούμε να ανακτήσουμε παλαιότερες μετρήσεις καθώς και να στέλνει αυτόματα ένα ηλεκτρονικό μήνυμα συναγερμού σε έναν ή περισσότερους παραλήπτες.

Δικτιο WSN

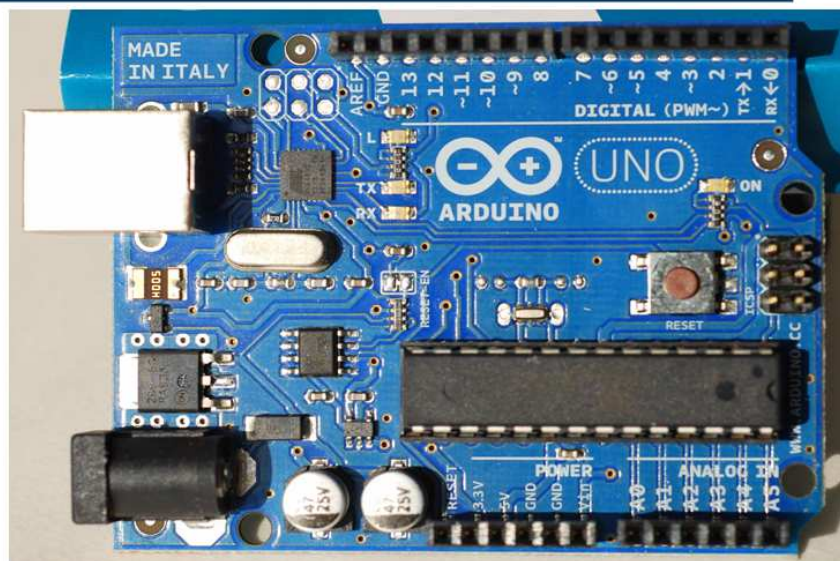


Διαχωρισμός δικτυου σε τμηματα

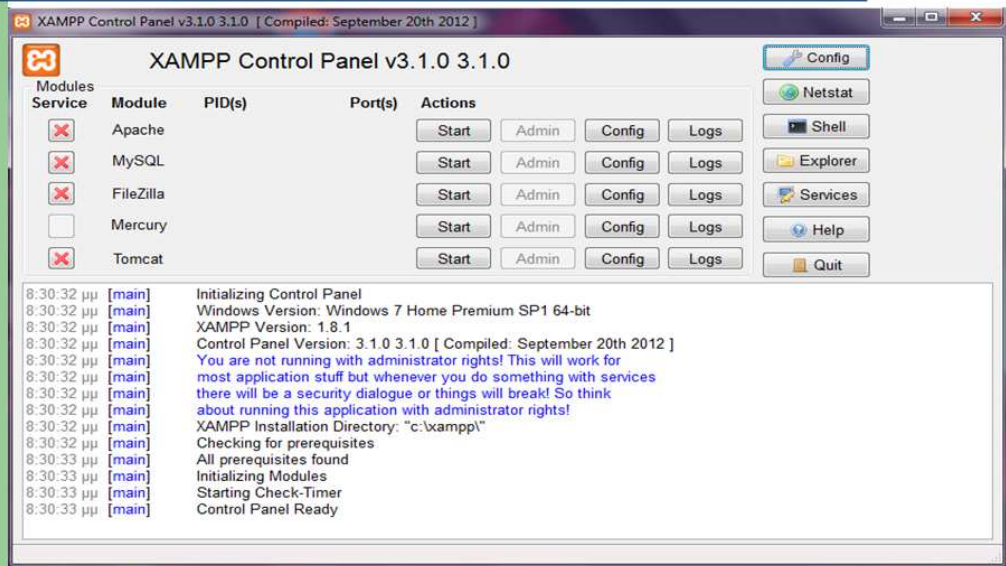
Για να καταφέρουμε να υλοποιήσουμε το WSN θα το διασπάσουμε σε τέσσερα τμήματα:

1. Δημιουργία πλατφόρμας αισθητήρων
2. Δημιουργία διακομιστή και βάσης δεδομένων
3. Δημιουργία εφαρμογής Android
4. Επικοινωνία όλων των τμημάτων μεταξύ τους

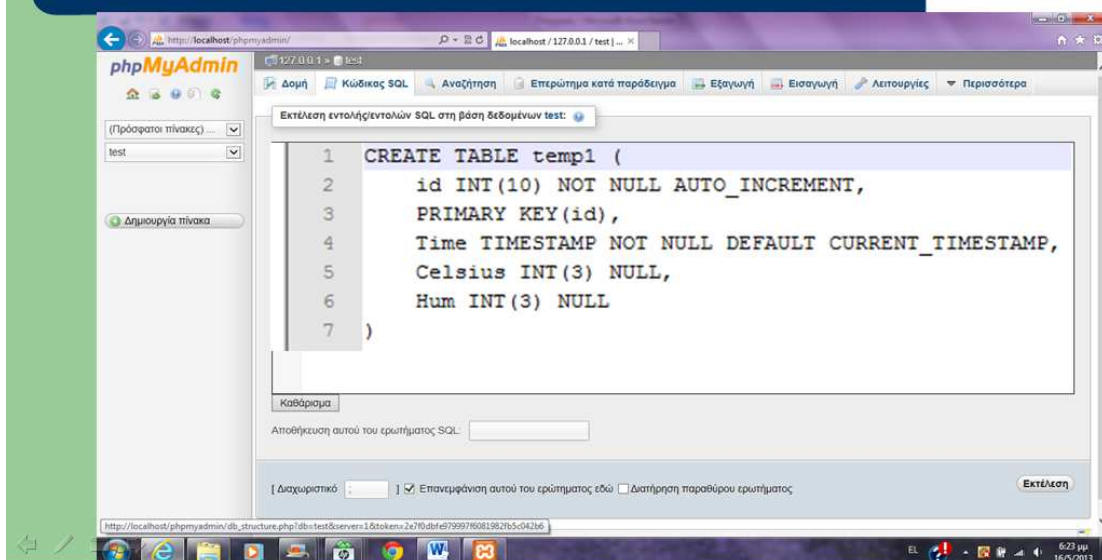
Δημιουργία Πλατφόρμας Αισθητήρων



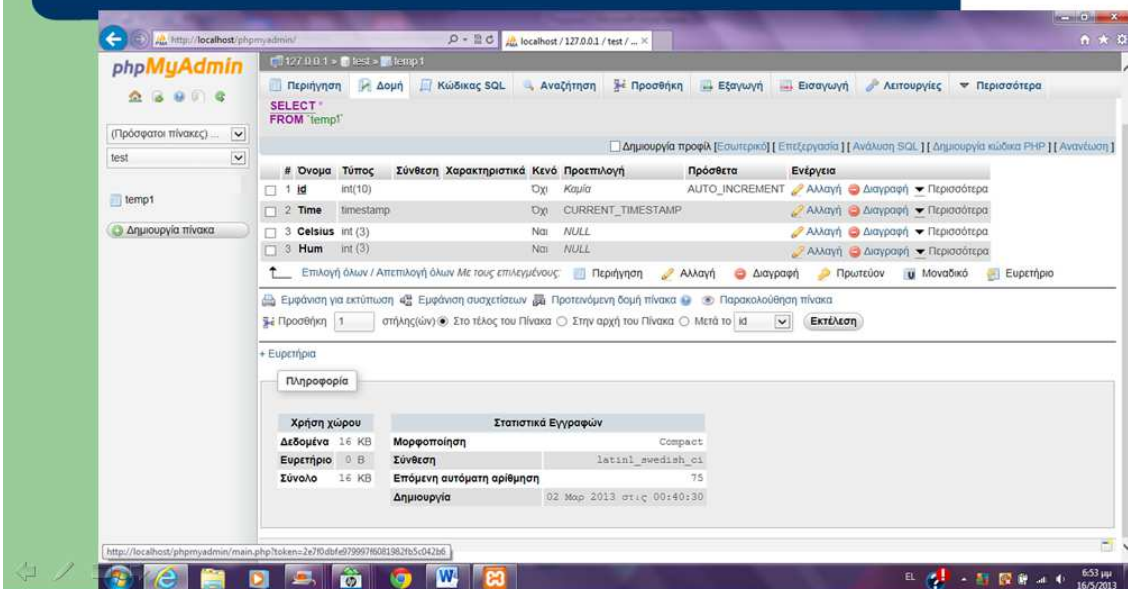
Δημιουργία Διακομιστή και ΒΔ



Δημιουργία Διακομιστή και ΒΔ



Δημιουργία Διακομιστή και ΒΔ



Δημιουργία εφαρμογής Android



Επικοινωνία όλων των τμημάτων μεταξύ τους

Το λογισμικό που απαιτείτε είναι το εξής:

- Arduino 1.0.1
- Processing 2.0b5
- Xampp-win32-01.8.1-VC9 (Apache, MySQL, Mercury)
- Eclipse 4.2.1 (Android)




Επικοινωνία όλων των τμημάτων μεταξύ τους

Οι γλώσσες προγραμματισμού και συγγραφής
σεναρίων είναι οι εξής:

- C++
- JAVA
- PHP
- SQL
- Wiring




Συλλογή περιβαλλοντικών δεδομένων



1.0
ARDUINO
AN OPEN PROJECT WRITTEN, DEBUGGED AND SUPPORTED
BY MASSIMO BANZI, DAVID CUARTIELLES, TOM IGOE,
GIANLUCA MARTINO AND DAVID MELLIS
BASED ON PROCESSING BY CASEY REAS AND BEN FRY

```
1 #include <stdio.h> //ΒΙΒΛΙΟΘΗΚΗ C
2 #include <math.h> // ΜΑΘΗΜΑΤΙΚΗ ΒΙΒΛΙΟΘΗΚΗ
3 #include <dht11.h> // ΒΙΒΛΙΟΘΗΚΗ ΑΙΣΘΗΤΗΡΑ DHT11
4 dht11 DHT11; // ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
5 #define DHT11PIN 2 // ΔΗΛΩΣΗ ΣΤΑΘΕΡΑΣ ΑΚΡΟΔΕΚΤΗ
6 const unsigned int SENSOR_PIN = A0; //ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ ΑΚΡΟΔΕΚΤΗ
7 const unsigned int BAUD_RATE = 9600; // ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ ΡΥΘΜΟΥ ΜΕΤΑΔΟΣΗΣ
8 double Thermokrasia (int RawADC){ //ΔΗΜΗΟΥΡΓΙΑ ΣΥΝΑΡΤΗΣΗΣ "Thermokrasia" ΓΙΑ ΝΑ ΜΕΤΑΤΡΕΠΕΙ ΤΗΝ ΤΙΜΗ ΠΟΥ ΣΤΕΛΕΝΕΙ Ο ΑΙΣΘΗΤΗΡΑΣ ΣΕ ΒΑΘΜΟΥΣ ΚΕΛΣΙΟΥ
9 double Temp;
10 Temp = analogRead(SENSOR_PIN);
11 Temp = (5.0 * Temp * 100.0)/1024.0;
12 return Temp;
13 }
14 void setup() { //ΕΚΚΙΝΗΣΗ ΚΥΡΙΟΥΣ ΠΡΟΓΡΑΜΜΑΤΟΣ
15 Serial.begin(BAUD_RATE); //ΕΚΚΙΝΗΣΗ ΡΥΘΜΟΥ ΜΕΤΑΔΟΣΗΣ
16 pinMode(SENSOR_PIN, INPUT); //ΔΗΛΩΣΗ ΑΚΡΟΔΕΚΤΗ ΟΠΟΥ ΛΑΜΒΑΝΟΝΤΑΙ ΤΑ ΔΕΔΟΜΕΝΑ ΑΠΟ ΤΟΝ ΑΙΣΘΗΤΗΡΑ LM35
17 pinMode(2, INPUT); //ΔΗΛΩΣΗ ΑΚΡΟΔΕΚΤΗ ΟΠΟΥ ΛΑΜΒΑΝΟΝΤΑΙ ΤΑ ΔΕΔΟΜΕΝΑ ΑΠΟ ΤΟΝ ΑΙΣΘΗΤΗΡΑ DHT11
18 }
19 void loop() { //ΣΥΝΑΡΤΗΣΗ ΕΠΑΝΑΛΥΨΗΣ
20
21 int temp = int(Thermokrasia(analogRead(SENSOR_PIN)));//ΔΗΛΩΣΗ ΤΗΣ ΜΕΤΑΒΛΗΤΗΣ "temp",ΔΙΑΒΑΣΜΑ ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΤΟΝ ΑΙΣΘΗΤΗΡΑ LM35, ΜΕΤΑΤΡΟΠΗ ΜΕΣΩ ΤΗΣ ΣΥΝΑΡΤΗΣΗΣ
22 Thermokrasia ΣΕ ΒΑΘΜΟΥΣ ΚΕΛΣΙΟΥ,ΜΕΤΑΤΡΟΠΗ ΤΟΥ DOUBLE ΣΕ INT ΚΑΙ ΑΠΟΔΟΣΗ ΤΙΜΗΣ ΣΤΗΝ ΜΤΑΒΛΗΤΗ temp
23 int hum = DHT11.read(DHT11PIN);//ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ "hum",ΔΙΑΒΑΣΜΑ ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΤΟΝ ΑΙΣΘΗΤΗΡΑ DHT11(ΕΚΚΙΝΗΣΗ ΑΙΣΘΗΤΗΡΑ),ΑΠΟΔΟΣΗ ΤΙΜΗΣ ΣΤΗΝ ΜΕΤΑΒΛΗΤΗ hum
24 int HUM=(int)DHT11.humidity;// ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ "HUM",ΜΕΤΑΤΡΟΠΗ ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΤΟΝ ΑΙΣΘΗΤΗΡΑ ΣΕ ΠΟΣΟΣΤΟ ΥΓΡΑΣΙΑΣ %,ΜΕΤΑΤΡΟΠΗ ΤΙΜΗΣ ΣΕ INT,ΑΠΟΔΟΣΗ ΤΙΜΗΣ ΣΤΗΝ
25 //ΜΕΤΑΒΛΗΤΗ HUM
26
27 int buf[]={temp,HUM};//ΔΗΛΩΣΗ ΠΙΝΑΚΑ "buf" ΚΑΙ ΓΕΜΙΣΜΑ
28 int i; //ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
29 for (i=0;i<2;i++)//ΣΥΝΘΗΚΗ FOR ΓΙΑ ΝΑ ΔΙΑΒΑΣΟΥΜΕ ΤΟΝ ΠΙΝΑΚΑ ΑΚΕΡΑΙΩΝ
30 Serial.println(buf[i]);//ΕΚΤΥΠΩΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ ΣΤΗΝ ΣΥΡΙΑΚΗ ΕΞΟΔΟ
31
32 delay(5000);//ΚΑΘΥΣΤΕΡΗΣΗ 5 ΔΕΠΤΕΡΟΛΕΠΤΩΝ
33 }
```

Καταχώρηση δεδομένων στη ΒΔ



Processing

An open project initiated by Ben Fry and Casey Reas

© 2004-2011 Ben Fry and Casey Reas
© 2001-2004 Massachusetts Institute of Technology

```
1 import de.bezier.data.sql.*;//ΒΙΒΛΙΟΘΗΚΗ ΓΙΑ ΤΗΝ ΕΠΙΚΟΙΝΩΝΙΑ ΜΕ ΤΗΝ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ
2 import processing.serial.*;//ΒΙΒΛΙΟΘΗΚΗ ΓΙΑ ΝΑ ΔΙΑΒΑΖΕΙ ΤΑ ΔΕΔΟΜΕΝΑ ΑΠΟ ΤΗΝ ΣΕΙΡΙΑΚΗ ΠΟΡΤΑ
3
4 Serial myPort; // ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
5
6 float value,a,b;// ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΩΝ
7 String[] message=new String[5];//ΔΗΛΩΣΗ ΠΙΝΑΚΑ ΧΑΡΑΚΤΗΡΩΝ
8 MySQL dbconnection;//ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ
9
10 void setup() {//ΕΝΑΡΧΗ ΠΡΟΓΡΑΜΜΑΤΟΣ
11     myPort = new Serial(this, "COM3", 9600);//ΔΗΛΩΣΗ ΣΕΙΡΙΑΚΗΣ ΠΟΡΤΑΣ ΚΑΙ ΡΥΘΜΟΥ ΜΕΤΑΔΟΣΗΣ
12 }
13
14 void draw(){//ΕΝΑΡΧΗ ΕΠΑΝΑΛΑΜΒΑΝΟΜΕΝΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ
15     serialEvent();//ΣΥΝΑΡΤΗΣΗ
16 }
17
18 void serialEvent(){//ΔΡΩΣΗ ΣΥΝΑΡΤΗΣΗΣ ΠΟΥ ΔΙΑΒΑΖΕΙ ΤΑ ΔΕΔΟΜΕΝΑ ΑΠΟ ΤΗΝ ΣΕΙΡΙΑΚΗ ΠΟΡΤΑ ΤΟΥ ARDUINO 1.0.1
19 {
20
21     for(int i=0;i<2;i++){//ΣΥΝΘΗΚΗ
22         {delay(5000);//ΚΑΘΥΣΤΗΡΗΣΗ 5 ΔΕΠΤΕΡΟΔΕΠΤΩΝ
23         message[i] = myPort.readStringUntil(13);//ΚΑΤΑΧΡΗΣΗ ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΤΗΝ COM3 ΣΤΟΝ ΠΙΝΑΚΑ ΧΑΡΑΚΤΗΡΩΝ message ΜΕΧΡΙ ΝΑ ΤΕΛΕΙΩΣΕΙ ΤΟ ΑΡΧΕΙΟ ΑΠΟΣΤΟΛΗΣ
24             if(message[i] != null){//ΣΥΝΘΗΚΗ ΜΕ ΒΑΣΗ ΤΗΝ ΟΠΟΙΑ ΠΕΡΝΟΥΜΕ ΤΑ ΔΕΔΟΜΕΝΑ ΤΩΝ ΜΕΤΡΗΣΕΩΝ ΑΠΟ ΤΟΥΣ ΑΙΣΘΗΤΗΡΕΣ ΚΑΙ ΤΑ ΚΑΤΑΧΡΟΥΜΕ ΣΕ ΜΕΤΑΒΛΗΤΕΣ
25                 {
26                     value = float(message[i]);//ΑΠΟΔΟΣΗ ΤΙΜΗΣ ΣΤΗΝ ΜΕΤΑΒΛΗΤΗ value
27
28                     if (i==0){//ΣΥΝΘΗΚΗ
29                         a=value;//ΑΠΟΔΟΣΗ ΤΙΜΗΣ ΣΤΗΝ ΜΕΤΑΒΛΗΤΗ a
30                         println("a=");//ΕΚΤΥΠΩΣΗ ΤΙΜΗΣ
```

```

31 }
32 if (i==1)//ΣΥΝΘΗΚΗ
33     b=value;//ΑΠΟΔΩΣΗ ΤΙΜΗΣ ΣΤΗΝ ΜΕΤΑΒΛΗΤΗ b
34     println("b=");//ΕΚΤΥΠΩΣΗ ΤΙΜΗΣ
35 }
36
37     println(value);//ΕΚΤΥΠΩΣΗ ΤΙΜΗΣ
38 }
39
40 }
41
42 String user    = "root";//ΔΗΛΩΣΗ ΣΤΟΙΧΕΙΩΝ
43 String pass    = "";//ΔΗΛΩΣΗ ΣΤΟΙΧΕΙΩΝ
44 String database = "test";//ΔΗΛΩΣΗ ΣΤΟΙΧΕΙΩΝ
45
46 dbconnection = new MySQL( this, "localhost", database, user, pass );//ΣΥΝΔΕΣΗ ΣΤΗΝ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ ΤΟΥ SERVER "localhost" ΜΕ ΤΑ
47                                     //ΣΤΟΙΧΕΙΑ ΠΟΥ ΔΗΛΩΘΗΚΑΝ ΠΑΡΑΠΑΝΩ
48
49 if ( dbconnection.connect() ){//ΣΥΝΘΗΚΗ
50
51     dbconnection.execute( "INSERT INTO temp1 (Celsius,Hum) VALUES ('"+a+"','"+b+"');");//ΚΑΤΑΧΩΡΗΣΗ ΤΙΜΩΝ ΣΤΗΝ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ "temp1"
52                                     //ΣΤΙΣ ΣΤΗΛΕΣ "Celsius" ΚΑΙ "Hum"
53     println();//ΕΚΤΥΠΩΣΗ
54     println("ΑΠΟΣΤΟΛΗ ΔΕΔΟΜΕΝΩΝ!");//ΕΚΤΥΠΩΣΗ
55
56 }
57 }

```

Εμφάνιση δεδομένων με PHP

ΗΜΕΡΟΜΗΝΙΑ	ΘΕΡΜΟΚΡΑΣΙΑ	ΥΓΡΑΣΙΑ
2013-05-10 12:04:37	23 °C	40 %
2013-05-10 12:04:47	24 °C	40 %
2013-05-10 12:06:53	24 °C	40 %
2013-05-10 12:07:03	23 °C	40 %
2013-05-10 12:07:13	66 °C	40 %
2013-05-10 12:07:23	26 °C	42 %
2013-05-10 12:07:33	27 °C	43 %
2013-05-10 12:07:43	29 °C	40 %
2013-05-10 12:07:54	28 °C	40 %
2013-05-10 12:08:04	27 °C	40 %
2013-05-10 12:08:14	26 °C	40 %
2013-05-10 12:08:24	26 °C	47 %
2013-05-10 12:32:10	24 °C	0 %
2013-05-10 12:32:21	24 °C	0 %
2013-05-10 12:32:31	24 °C	0 %
2013-05-10 12:32:41	24 °C	0 %
2013-05-10 12:32:51	24 °C	0 %
2013-05-10 12:33:56	24 °C	43 %
2013-05-10 12:34:06	24 °C	43 %
2013-05-10 12:34:16	24 °C	44 %
2013-05-10 12:34:26	24 °C	43 %
2013-05-10 12:34:36	24 °C	43 %
2013-05-10 12:34:46	24 °C	43 %

```

1 <html>
2 <body background=C:\xampp\htdocs\example\app.png //ΕΠΙΛΟΓΗ ΕΙΚΟΝΑΣ ΓΙΑ ΦΟΝΤΟ
3 <table border="1" cellspacing="2" cellpadding="2" bgcolor="999999" align="center" //ΠΥΘΜΗΕΙΣ ΠΙΝΑΚΑ
4 <tr bgcolor="#6297c5" // ΧΡΩΜΑ ΓΙΑ ΦΟΝΤΟ ΤΟΥ ΠΙΝΑΚΑ
5 <th><font face="Arial"> ΗΜΕΡΟΜΗΝΙΑ </font></th> //ΕΠΙΛΟΓΗ ΓΡΑΜΜΑΤΟΣΕΙΡΑΣ ΕΜΦΑΝΙΣΗΣ ΚΑΙ ΟΝΟΜΑΣΙΑ ΕΤΙΚΕΤΑΣ ΤΗΣ ΣΤΗΛΗΣ ΤΟΥ ΠΙΝΑΚΑ
6 <th><font face="Arial"> ΘΕΡΜΟΚΡΑΣΙΑ </font></th> //ΕΠΙΛΟΓΗ ΓΡΑΜΜΑΤΟΣΕΙΡΑΣ ΕΜΦΑΝΙΣΗΣ ΚΑΙ ΟΝΟΜΑΣΙΑ ΕΤΙΚΕΤΑΣ ΤΗΣ ΣΤΗΛΗΣ ΤΟΥ ΠΙΝΑΚΑ
7 <th><font face="Arial"> ΥΓΡΑΣΙΑ </font></th> //ΕΠΙΛΟΓΗ ΓΡΑΜΜΑΤΟΣΕΙΡΑΣ ΕΜΦΑΝΙΣΗΣ ΚΑΙ ΟΝΟΜΑΣΙΑ ΕΤΙΚΕΤΑΣ ΤΗΣ ΣΤΗΛΗΣ ΤΟΥ ΠΙΝΑΚΑ
8 </tr>
9
10 <?php
11 include "db_test_temperature.php"; //ΕΝΣΩΜΑΤΩΣΗ ΑΡΧΕΙΟΥ "db_test_temperature.php"
12
13 echo('<meta http-equiv="refresh" content="5">');//ΑΝΑΝΕΩΣΗ ΣΕΛΙΔΑΣ
14
15
16 mysql_connect($host,$username,$password);//ΣΥΝΔΕΞΗ ΣΤΟΝ ΔΙΑΚΟΜΙΣΤΗ
17 @$mysql_select_db($database) or die("Δεν είναι δυνατή η επιλογή της βάσης δεδομένων.");//ΣΥΝΔΕΞΗ ΣΤΗ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ
18
19 //ομοιοση e-mail
20 $t=mysql_query("SELECT * FROM temp1 where Celsius>25"); //ΕΠΙΛΟΓΗ ΓΡΑΜΜΗΣ ΠΟΥ Η ΘΕΡΜΟΚΡΑΣΙΑ ΕΙΝΑΙ ΜΕΓΑΛΥΤΕΡΗ ΤΩΝ 25 ΒΑΘΜΩΝ ΚΕΛΣΙΟΥ
21
22 while($r=mysql_fetch_assoc($t))
23 {
24 $p={$r["Celsius"]};
25 $a1=date('Y-m-d H:i:s');
26 $a2={$r["Time"]};
27 if (strtotime($a2) >= strtotime('+1 hour -5 seconds'))
28 $mailto=mail("me@localhost.com","!!!WARNING!!!","At $a2 the temperature is: $p Celsius degrees!!!","me@localhost.com");
29 //ΕΑΝ ΥΠΑΡΧΕΙ ΤΙΜΗ ΜΕΓΑΛΥΤΕΡΗ ΑΠΟ 25 ΒΑΘΜΟΥΣ ΚΕΛΣΙΟΥ ΚΑΙ Η ΗΜΕΡΟΜΗΝΙΑ ΚΑΙ ΩΡΑ ΤΗΣ ΜΕΤΡΗΣΗΣ ΕΙΝΑΙ ΜΕΧΡΙ ΚΑΙ 5' ΠΡΙΝ ΑΠΟ ΤΗΝ ΤΡΕΧΩΣ ΩΡΑ ΤΟΤΕ
30 //ΣΤΕΙΛΕ Ε-MAIL ΣΤΟΝ "me@localhost.com" ΜΕ ΘΕΜΑ "!!!WARNING!!!" ΚΑΙ ΚΕΙΜΕΝΟ "At $a2(ΩΡΑ) the temperature is: $p(ΘΕΡΜΟΚΡΑΣΙΑ) Celsius degrees!!!" ΑΠΟ ΤΟΝ "me@
31
32 $sql = "SELECT * FROM `temp1`";//ΚΑΤΑΧΩΡΗΣΗ ΑΙΤΗΜΑΤΟΣ ΣΤΗΝ ΜΕΤΑΒΛΗΤΗ
33 $result=mysql_query($sql);//ΑΠΟΙΟΤΗΛΗ ΑΙΤΗΜΑΤΟΣ ΣΤΗΝ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΑΠΟΘΗΚΕΥΣΗ ΑΠΑΝΤΗΣΗΣ ΣΤΗΝ ΜΕΤΑΒΛΗΤΗ
34
35 $sql = "SELECT * FROM `temp1`";//ΚΑΤΑΧΩΡΗΣΗ ΑΙΤΗΜΑΤΟΣ ΣΤΗΝ ΜΕΤΑΒΛΗΤΗ
36 $result=mysql_query($sql);//ΑΠΟΙΟΤΗΛΗ ΑΙΤΗΜΑΤΟΣ ΣΤΗΝ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΑΠΟΘΗΚΕΥΣΗ ΑΠΑΝΤΗΣΗΣ ΣΤΗΝ ΜΕΤΑΒΛΗΤΗ
37 $num=mysql_numrows($result); //ΑΠΟΘΗΚΕΥΣΗ ΠΛΗΘΟΥΣ ΓΡΑΜΜΩΝ ΤΟΥ ΑΠΟΤΕΛΕΣΜΑΤΟΣ
38
39 mysql_close();//ΚΛΕΙΣΙΜΟ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΣ
40
41 $i=0;//ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
42 while ($i < $num) //ΩΣΤΟ Η ΤΙΜΗ ΤΗΣ ΜΕΤΑΒΛΗΤΗΣ ΕΙΝΑΙ ΜΙΚΡΟΤΕΡΗ ΑΠΟ ΤΟ ΠΛΗΘΟΣ ΤΩΝ ΓΡΑΜΜΩΝ ΤΟΥ ΠΙΝΑΚΑ
43 {
44 $Time=mysql_result($result,$i,"Time");//ΚΑΤΑΧΩΡΗΣΗ ΤΙΜΗΣ ΣΤΗΝ ΜΕΤΑΒΛΗΤΗ
45 $Celsius=mysql_result($result,$i,"Celsius");//ΚΑΤΑΧΩΡΗΣΗ ΤΙΜΗΣ ΣΤΗΝ ΜΕΤΑΒΛΗΤΗ
46 $Hum=mysql_result($result,$i,"Hum");//ΚΑΤΑΧΩΡΗΣΗ ΤΙΜΗΣ ΣΤΗΝ ΜΕΤΑΒΛΗΤΗ
47
48 echo "<tr>";
49 echo "<td align=center>$Time</td>";//ΚΑΤΑΧΩΡΗΣΗ ΤΙΜΗΣ ΣΤΟΝ ΠΙΝΑΚΑ
50 if($Celsius>25)//ΕΑΝ Η ΤΙΜΗ ΕΙΝΑΙ ΜΕΓΑΛΥΤΕΡΗ ΤΟΥ 25
51 echo "<td align=center bgcolor=#00ff00><b>$Celsius °C</b></td>";//ΑΛΛΑΓΗ ΧΡΩΜΑΤΟΣ ΣΤΟ ΦΟΝΤΟ ΚΑΙ ΚΑΤΑΧΩΡΗΣΗ ΤΙΜΗΣ ΣΤΟΝ ΠΙΝΑΚΑ
52 else
53 echo "<td align=center><b>$Celsius °C</b></td>";//ΚΑΤΑΧΩΡΗΣΗ ΤΙΜΗΣ ΣΤΟΝ ΠΙΝΑΚΑ
54 echo "<td align=center>$Hum %</td>";//ΚΑΤΑΧΩΡΗΣΗ ΤΙΜΗΣ ΣΤΟΝ ΠΙΝΑΚΑ
55 echo "</tr>";
56 $i++;
57 }
58
59 ?>
60
61 </table>
62 </body>
63 </html>

```

Είσοδος στην ΒΔ μέσω PHP

```
1 <?php
2 $host = "localhost";
3 $username = "root";
4 $password = "";
5 $database = "test";
6
7 mysql_connect($host, $username, $password) or die (mysql_error());
```

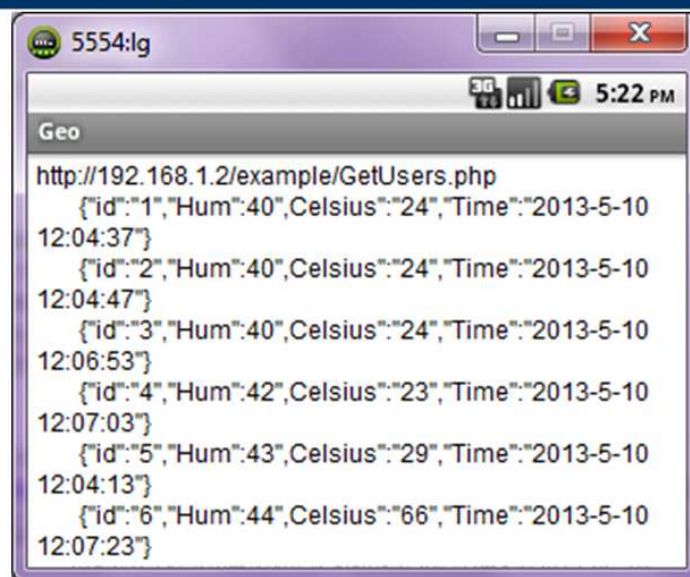
Επικοινωνία ΒΔ με εφαρμογή Android

```
1 <?php
2 mysql_connect("localhost","root",""); //ΣΥΝΔΕΞΗ ΜΕ ΤΟΝ ΔΙΑΚΟΜΙΣΤΗ
3 mysql_select_db("test"); //ΣΥΝΔΕΞΗ ΜΕ ΤΗΝ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ
4 $t=mysql_query("SELECT * FROM temp1 where Celsius>25"); ///ΕΠΙΛΟΓΗ ΓΡΑΜΜΗΣ ΠΟΥ Η ΘΕΡΜΟΚΡΑΣΙΑ ΕΙΝΑΙ ΜΕΓΑΛΥΤΕΡΗ ΤΩΝ 25 ΒΑΘΜΩΝ ΚΕΛΣΙΟΥ
5
6
7 while($r=mysql_fetch_assoc($t))
8 {
9     $p={$r["Celsius"]};
10    $a1=date('Y-m-d H:i:s');
11    $a2={$r["Time"]};
12    if (strtotime($a2) >= strtotime('+1 hour -5 seconds'))
13    $m=mail("me@localhost.com","!!!WARNING!!!","At $a2 the temperature is: $p Celsius degrees!!!","me@localhost.com");
14    ///ΕΑΝ ΥΠΑΡΧΕΙ ΤΙΜΗ ΜΕΓΑΛΥΤΕΡΗ ΑΠΟ 25 ΒΑΘΜΟΥΣ ΚΕΛΣΙΟΥ ΚΑΙ Η ΗΜΕΡΟΜΗΝΙΑ ΚΑΙ ΩΡΑ ΤΗΣ ΜΕΤΡΗΣΗΣ ΕΙΝΑΙ ΜΕΧΡΙ ΚΑΙ 5" ΠΡΙΝ ΑΠΟ ΤΗΝ ΤΡΕΧΟΥΣΑ ΩΡΑ ΤΟΤΕ
15    //ΣΤΕΙΛΕ Ε-MAIL ΣΤΟΝ "me@localhost.com" ΜΕ ΘΕΜΑ "!!!WARNING!!!"
16    ///ΚΑΙ ΚΕΙΜΕΝΟ "At $a2 (ΩΡΑ) the temperature is: $p(ΘΕΡΜΟΚΡΑΣΙΑ) Celsius degrees!!!" ΑΠΟ ΤΟΝ "me@localhost.com"
17
18    $q=mysql_query("SELECT * FROM temp1"); //ΑΠΟΣΤΟΛΗ ΑΙΤΗΜΑΤΟΣ ΣΤΗΝ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΑΠΟΘΗΚΕΥΗ ΑΠΑΝΤΗΣΗΣ ΣΤΗΝ ΜΕΤΑΒΑΗΤΗ
19    while($row=mysql_fetch_assoc($q) //ΣΥΝΘΗΚΗ ΜΕ ΒΑΣΗ ΣΤΗΝ ΟΠΟΙΑ ΟΣΟ ΥΠΑΡΧΟΥΝ ΔΕΔΟΜΕΝΑ ΣΤΟΝ ΠΙΝΑΚΑ temp1 ΑΠΟΘΗΚΕΥΟΝΤΑΙ ΣΕ ΜΙΑ ΜΕΤΑΒΑΗΤΗ
20    //ΚΑΙ Η ΜΕΤΑΒΑΗΤΗ ΑΥΤΗ ΤΥΠΩΝΕΤΕ
21    $json_output[]=$row;
22    print(json_encode($json_output));
23    mysql_close(); //ΚΛΕΙΣΙΜΟ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ
24 >
```

Δημιουργία εφαρμογής Android



Δημιουργία εφαρμογής Android



```

1 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
2   xmlns:tools="http://schemas.android.com/tools"
3   android:layout_width="match_parent"
4   android:layout_height="match_parent"
5   android:paddingBottom="@dimen/activity_vertical_margin"
6   android:paddingLeft="@dimen/activity_horizontal_margin"
7   android:paddingRight="@dimen/activity_horizontal_margin"
8   android:paddingTop="@dimen/activity_vertical_margin"
9   tools:context=".MainActivity" >
10
11   <Button
12     android:id="@+id/button1"
13     android:layout_width="wrap_content"
14     android:layout_height="wrap_content"
15     android:layout_alignParentBottom="true"
16     android:layout_centerHorizontal="true"
17     android:layout_marginBottom="156dp"
18     android:text="@string/Connect" />
19
20   <EditText
21     android:id="@+id/editText1"
22     android:layout_width="wrap_content"
23     android:layout_height="wrap_content"
24     android:layout_above="@+id/button1"
25     android:layout_centerHorizontal="true"
26     android:layout_marginBottom="62dp"
27     android:ems="10"
28     android:inputType="text"
29     android:text="http://192.168.1.2/example/GetUsers.php" >
30
31     <requestFocus />
32   </EditText>
33 </RelativeLayout>

```

```

1 package com.new2;//ΟΝΟΜΑΣΙΑ ΠΑΚΕΤΟΥ
2
3 import android.os.Bundle;//ΜΕΘΟΔΟΣ ΑΝΤΙΣΤΟΙΧΙΑΣ STRING ΣΕ ΔΙΑΦΟΡΟΥΣ ΤΥΠΟΥΣ ΔΕΔΟΜΕΝΩΝ
4 import android.app.Activity;//ΜΕΘΟΔΟΣ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ ΑΔΗΛΕΠΙΔΡΑΣΗΣ ΜΕ ΤΟ ΧΡΗΣΤΗ
5 import android.content.Intent;//ΜΕΘΟΔΟΣ ΠΕΡΙΓΡΑΦΗΣ ΛΕΙΤΟΥΡΓΙΩΝ ΠΟΥ ΠΡΕΠΕΙ ΝΑ ΕΚΤΕΛΕΣΤΟΥΝ
6 import android.view.View;//ΜΕΘΟΔΟΣ ΠΟΥ ΑΠΟΤΕΛΕΙ ΤΟ ΒΑΣΙΚΟ ΔΟΜΙΚΟ ΣΤΟΙΧΕΙΟ ΓΙΑ ΣΤΟΙΧΕΙΑ ΔΙΕΠΡΑΨΗΣ ΧΡΗΣΤΗ
7 import android.widget.Button;//ΜΕΘΟΔΟΣ ΕΜΦΑΝΙΣΗΣ ΚΟΥΜΠΙΟΥ ΣΤΟ ΧΡΗΣΤΗ
8 import android.widget.EditText;//ΜΕΘΟΔΟΣ ΕΜΦΑΝΙΣΗΣ ΠΛΑΙΣΙΟΥ ΚΕΙΜΕΝΟΥ ΣΤΟ ΧΡΗΣΤΗ
9
10 public class MainActivity extends Activity {//ΚΛΑΣΗ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ MainActivity
11
12     EditText mEditText;//ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
13     String key="";//ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
14     Button mButton;//ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
15     @Override
16     protected void onCreate(Bundle savedInstanceState) {//ΣΥΝΑΡΤΗΣΗ onCreate ΔΗΜΙΟΥΡΓΕΙΤΑΙ ΑΥΤΟΜΑΤΑ
17         super.onCreate(savedInstanceState);// ΑΚΟΛΟΥΘΙΑ ΚΛΗΣΕΩΝ
18         setContentView(R.layout.activity_main);// ΕΜΦΑΝΙΣΗ ΤΗΣ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ new3
19
20         mEditText=(EditText)findViewById(R.id.editText1);//ΑΝΤΙΣΤΟΙΧΙΑ ΜΕΤΑΒΛΗΤΗΣ
21         mButton=(Button)findViewById(R.id.button1);//ΑΝΤΙΣΤΟΙΧΙΑ ΜΕΤΑΒΛΗΤΗΣ
22         mButton.setOnClickListener{// ΜΕ ΤΟ ΠΑΤΗΜΑ ΤΟΥ ΚΟΥΜΠΙΟΥ ΠΕΡΝΕΙ ΟΤΙ ΓΡΑΦΕΙ ΤΟ EDIT TEXT,
23             //ΤΟ ΑΠΟΘΗΚΕΥΕΙ ΕΤΗΝ ΜΕΤΑΒΛΗΤΗ met ΚΑΙ ΤΟ ΣΤΕΛΝΕΙ ΕΤΗΝ ΔΡΑΣΤΗΡΙΟΤΗΤΑ New3,
24             //ΑΝΟΙΓΕΙ ΤΗΝ ΔΡΑΣΤΗΡΙΟΤΗΤΑ New3
25             new View.OnClickListener() {
26                 public void onClick(View view)
27                 {
28                     key=mEditText.getText().toString();
29                     Intent i = new Intent(getApplicationContext(), New3.class);
30                     i.putExtra("met", key);
31                     startActivity(i);
32                 }
33             });

```

```

1 <?xml version="1.0" encoding="utf-8"?> //ΕΚΔΟΣΗ ΚΑΙ ΚΩΔΙΚΟΠΟΙΗΣΗ
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android" //ΑΝΟΙΓΜΑ ΓΡΑΜΜΙΚΗ ΔΙΑΤΑΞΗ ΟΘΟΝΗΣ
3     android:orientation="vertical" //ΚΑΘΕΤΗ ΔΙΑΤΑΞΗ ΣΤΟΙΧΕΙΩΝ
4     android:layout_width="fill_parent" //ΡΥΘΜΗΣΗ ΠΛΑΤΟΥΣ
5     android:layout_height="fill_parent" //ΡΥΘΜΗΣΗ ΥΨΟΥΣ
6 >
7
8 <TextView //ΑΝΤΙΚΕΙΜΕΝΟ ΤΥΠΟΥ TEXTVIEW
9     android:id="@+id/textView1" //ΟΝΟΜΑΣΙΑ ΑΝΤΙΚΕΙΜΕΝΟΥ
10    android:layout_width="match_parent" //ΡΥΘΜΗΣΗ ΠΛΑΤΟΥΣ
11    android:layout_height="wrap_content" //ΡΥΘΜΗΣΗ ΥΨΟΥΣ
12    android:scrollbarAlwaysDrawVerticalTrack="false" //ΕΙΣΑΓΩΓΗ ΚΑΘΕΤΗΣ ΜΠΑΡΑΣ ΚΥΛΗΣΗΣ
13    android:textColor="#000000"/> //ΕΠΙΛΟΓΗ ΧΡΩΜΑΤΟΣ ΓΡΑΜΜΑΤΟΣΕΙΡΑΣ
14
15 </LinearLayout>//ΚΛΕΙΣΙΜΟ ΓΡΑΜΜΙΚΗ ΔΙΑΤΑΞΗ ΟΘΟΝΗΣ

```

```

1 package com.new2;//ΟΝΟΜΑΣΙΑ ΠΑΚΕΤΟΥ
2
3 import java.io.BufferedReader;//ΜΕΘΟΔΟΣ ΠΟΥ ΔΙΑΒΑΖΕΙ ΧΑΡΑΚΤΗΡΕΣ
4 import java.io.InputStream; //Η ΒΑΣΙΚΗ ΜΕΘΟΔΟΣ ΟΛΩΝ ΤΩΝ ΕΙΣΟΔΩΝ ΤΗΣ JAVA
5 import java.io.InputStreamReader;// ΜΕΘΟΔΟΣ ΜΕΤΑΤΡΟΠΗΣ ΤΩΝ bytes ΣΕ ΧΑΡΑΚΤΗΡΕΣ
6 import java.util.ArrayList;// ΜΕΘΟΔΟΣ ΓΙΑ ΤΙΣ ΔΙΣΤΕΣ ΠΙΝΑΚΩΝ
7
8 import org.apache.http.HttpEntity;//ΜΕΘΟΔΟΣ ΑΙΤΗΣΗΣ Hypertext Transfer Protocol (HTTP)
9 import org.apache.http.HttpResponse;// ΜΕΘΟΔΟΣ ΑΠΟΚΡΙΣΗΣ HTTP
10 import org.apache.http.NameValuePair;//ΜΕΘΟΔΟΣ ΕΓΚΛΕΙΣΜΟΥ ΧΑΡΑΚΤΗΡΑ Η ΖΕΥΓΟΥΣ ΤΙΜΩΝ
11 import org.apache.http.client.HttpClient;//ΜΕΘΟΔΟΣ ΕΠΙΚΟΙΝΩΝΙΑΣ CLIENT-HTTP
12 import org.apache.http.client.entity.UrlEncodedFormEntity;//ΜΕΘΟΔΟΣ ΚΑΤΑΛΟΓΟΥ URL ΚΩΔΗΚΟΠΟΗΜΕΝΩΝ ΖΕΥΓΩΝ
13 import org.apache.http.client.methods.HttpPost;//ΜΕΘΟΔΟΣ ΓΙΑ ΤΗΝ ΑΟΣΤΟΛΗ ΜΙΑΣ ΑΙΤΗΣΗΣ POST HTTP
14 import org.apache.http.impl.client.DefaultHttpClient;//ΜΕΘΟΔΟΣ ΠΡΟΕΠΙΛΟΓΗΣ HTTP CLIENT
15 import org.apache.http.message.BasicNameValuePair;//ΜΕΘΟΔΟΣ ΕΓΚΛΕΙΣΜΟΥ ΧΑΡΑΚΤΗΡΑ Η ΖΕΥΓΟΥΣ ΤΙΜΩΝ
16 import org.json.JSONArray;//ΜΕΘΟΔΟΣ ΕΥΡΕΤΗΡΙΟΥ ΑΚΟΛΟΥΘΙΑΣ ΤΙΜΩΝ
17 import org.json.JSONException;//ΜΕΘΟΔΟΣ ΥΠΟΔΕΙΞΗΣ ΠΡΟΒΛΗΜΑΤΩΝ
18 import org.json.JSONObject; //ΜΕΘΟΔΟΣ ΤΡΟΠΟΠΟΙΗΣΗΣ ΤΙΜΩΝ
19
20 import android.app.Activity;//ΜΕΘΟΔΟΣ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ ΑΛΗΛΕΠΙΔΡΑΣΗΣ ΜΕ ΤΟ ΧΡΗΣΤΗ
21 import android.os.Bundle;//ΜΕΘΟΔΟΣ ΑΝΤΙΣΤΟΙΧΙΑΣ STRING ΣΕ ΔΙΑΦΟΡΟΥΣ ΤΥΠΟΥΣ ΔΕΔΟΜΕΝΩΝ
22 import android.os.Handler;//ΜΕΘΟΔΟΣ ΕΙΣΑΓΩΓΗΣ HANDLER ΓΙΑ ΕΠΕΞΕΡΓΑΣΙΑ ΜΥΝΗΜΑΤΩΝ ΚΑΙ ΑΝΤΙΚΕΙΜΕΝΩΝ
23 import android.text.method.ScrollingMovementMethod;//ΜΕΘΟΔΟΣ ΛΕΙΤΟΥΡΓΙΑΣ ΚΥΛΗΣΗΣ ΚΕΙΜΕΝΟΥ
24 import android.util.Log;//ΜΕΘΟΔΟΣ ΑΡΧΕΙΟΥ ΚΑΤΑΓΡΑΦΗΣ ΕΞΟΔΟΥ
25 import android.widget.LinearLayout;//ΜΕΘΟΔΟΣ ΡΥΘΜΗΣΗΣ ΓΡΑΜΜΙΚΗΣ ΔΙΑΤΑΞΗΣ
26 import android.widget.TextView;//ΜΕΘΟΔΟΣ ΕΜΦΑΝΙΣΗΣ ΚΕΙΜΕΝΟΥ ΣΤΟ ΧΡΗΣΤΗ
27
28 public class New3 extends Activity {//ΚΛΑΣΗ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ New3
29
30     String key;//ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
31     TextView txt;//ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
32     Handler handler = new Handler();//ΔΗΛΩΣΗ HANDLER

```

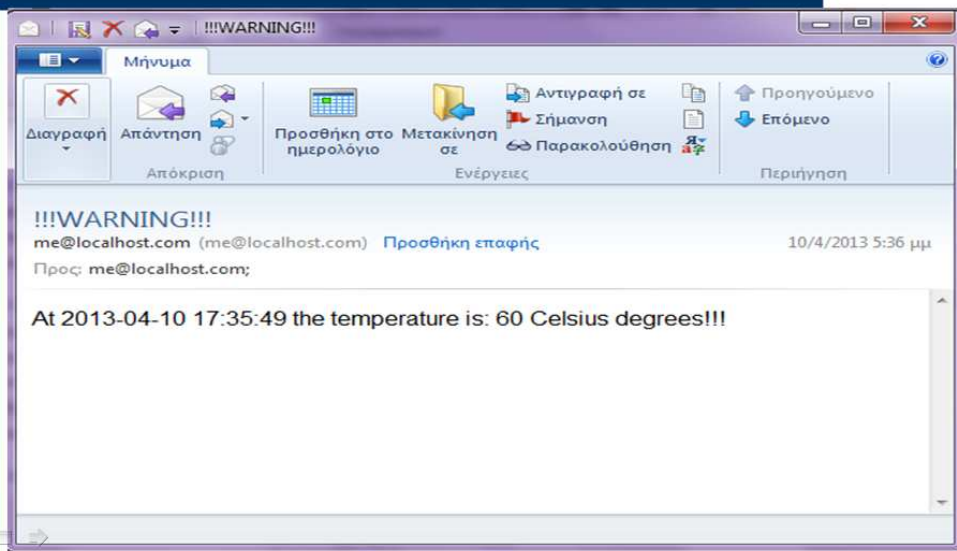


```

33 Runnable Refresh;//ΔΗΛΩΣΗ ΕΠΑΝΑΛΥΨΗΣ-ΑΝΑΝΕΩΣΗΣ
34 @Override
35 protected void onCreate(Bundle savedInstanceState) { //ΣΥΝΑΡΤΗΣΗ onCreate ΔΗΜΙΟΥΡΓΕΙΤΑΙ ΑΥΤΟΜΑΤΑ
36     super.onCreate(savedInstanceState); // ΑΚΟΛΟΥΘΙΑ ΚΛΗΣΕΩΝ
37     setContentView(R.layout.activity_new3); // ΕΜΦΑΝΗΣΗ ΤΗΣ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ new3
38
39     Bundle extras = getIntent().getExtras(); //ΜΕΤΑΦΟΡΑ ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΤΗΝ "MainActivity"
40     if (extras != null) {
41         key = extras.getString("met");
42     }
43
44     LinearLayout rootLayout = new LinearLayout(getApplicationContext()); //ΧΡΗΣΗΜΟΠΟΙΗΣΗ ΤΗΣ ΓΡΑΜΜΙΚΗΣ ΔΙΑΤΑΞΗΣ
45     txt = new TextView(getApplicationContext()); //ΑΝΑΘΕΣΗ ΣΤΟ txt ΤΟ TextView ΤΗΣ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ geol
46     rootLayout.addView(txt); //ΚΑΤΑΧΩΡΗΣΗ ΤΟΥ txt ΣΤΗΝ ΓΡΑΜΜΙΚΗ ΔΙΑΤΑΞΗ
47     setContentView(rootLayout); //ΕΜΦΑΝΗΣΗ ΤΗΣ ΓΡΑΜΜΙΚΗΣ ΔΙΑΤΑΞΗΣ
48
49     txt.setText("Connecting..."); //ΚΑΤΑΧΩΡΗΣΗ ΧΑΡΑΚΤΗΡΩΝ ΣΤΟ txt ΚΑΙ ΕΜΦΑΝΗΣΗ ΣΤΗΝ ΟΘΟΝΗ
50
51     Refresh = new Runnable() { //ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
52         public void run() { //ΔΗΛΩΣΗ ΣΥΝΑΡΤΗΣΗΣ
53
54             txt.setText(getServerData(key)); //ΕΜΦΑΝΗΣΗ ΑΠΟΤΕΛΕΣΜΑΤΟΣ ΤΗΣ ΣΥΝΑΡΤΗΣΗΣ getServerData ΣΤΗΝ ΟΘΟΝΗ
55             txt.setMovementMethod(new ScrollingMovementMethod()); //ΧΡΗΣΗΜΟΠΟΙΗΣΗ ΚΥΛΗΣΗΣ ΚΕΙΜΕΝΟΥ
56
57             handler.postDelayed(Refresh, 5000); //ΚΑΤΑΧΩΡΗΣΗ ΑΝΑΝΕΩΣΗΣ ΑΝΑ 5"
58         }
59     };
60     handler.post(Refresh); //ΑΝΑΝΕΩΣΗ
61 }
62
63 private String getServerData(String returnString) { //ΔΗΛΩΣΗ ΣΥΝΑΡΤΗΣΗΣ getServerData
64
65     InputStream is = null; //ΔΗΛΩΣΗ ΜΕΤΑΒΛΗΤΗΣ
66 } catch (Exception e) { // ΥΠΟΔΕΙΞΗ ΠΡΟΒΛΗΜΑΤΟΣ
67     Log.e("log_tag", "Error in http connection "+e.toString()); //ΕΜΦΑΝΗΣΗ ΠΡΟΒΛΗΜΑΤΟΣ ΣΤΟ ΑΡΧΕΙΟ ΚΑΤΑΓΡΑΦΗΣ ΕΞΟΔΟΥ
68 }
69 }
70
71 try{ //ΜΕΤΑΤΡΟΠΗ ΑΠΟΚΡΗΜΗΣ ΑΙΤΗΜΑΤΟΣ ΣΥΝΔΕΣΗΣ ΣΕ String
72     BufferedReader reader = new BufferedReader(new InputStreamReader(is,"iso-8859-1"),8);
73     StringBuilder sb = new StringBuilder();
74     String line = null;
75     while ((line = reader.readLine()) != null) {
76         sb.append(line + "\n");
77     }
78     is.close();
79     result=sb.toString();
80 } catch (Exception e) { // ΥΠΟΔΕΙΞΗ ΠΡΟΒΛΗΜΑΤΟΣ
81     Log.e("log_tag", "Error converting result "+e.toString()); //ΕΜΦΑΝΗΣΗ ΠΡΟΒΛΗΜΑΤΟΣ ΣΤΟ ΑΡΧΕΙΟ ΚΑΤΑΓΡΑΦΗΣ ΕΞΟΔΟΥ
82 }
83
84 try{ //ΑΝΤΙΓΡΑΦΗ ΤΩΝ json (JavaScript Object Notation) ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΤΟ ΑΡΧΕΙΟ GetUsers.php
85     JSONArray jArray = new JSONArray(result);
86     for(int i=0;i<jArray.length();i++){
87         JSONObject json_data = jArray.getJSONObject(i);
88         Log.i("log_tag"," id: "+json_data.getInt("id")+", Time: "+json_data.getString("Time")+
89             ", Celsius: "+json_data.getInt("Celsius")+", Hum: "+json_data.getInt("Hum")
90         );
91
92         returnString += "\n\t" + jArray.getJSONObject(i); //ΚΑΤΑΧΩΡΗΣΗ ΑΠΟΤΕΛΕΣΜΑΤΟΣ ΣΤΗ ΜΕΤΑΒΛΗΤΗ
93     }
94
95 } catch (JSONException e) { // ΥΠΟΔΕΙΞΗ ΠΡΟΒΛΗΜΑΤΟΣ
96     Log.e("log_tag", "Error parsing data "+e.toString()); //ΕΜΦΑΝΗΣΗ ΠΡΟΒΛΗΜΑΤΟΣ ΣΤΟ ΑΡΧΕΙΟ ΚΑΤΑΓΡΑΦΗΣ ΕΞΟΔΟΥ
97 }
98 return returnString; //ΕΠΙΣΤΡΟΦΗ ΤΙΜΗΣ ΣΤΗΝ ΣΥΝΑΡΤΗΣΗ getServerData ΚΑΙ ΕΜΦΑΝΗΣΗ ΑΠΟΤΕΛΕΣΜΑΤΟΣ ΣΤΗΝ ΟΘΟΝΗ
99 }

```

Μήνυμα συναγερμού επικίνδυνων τιμών



Μελλοντική Ανάπτυξη

Η πλατφόρμα αυτή μπορεί να επεκταθεί σε διάφορους τομείς. Ένα παράδειγμα θα ήταν να προστεθούν επιπλέον αισθητήρες οι οποίοι με κάποιο πρόσθετο υλικό θα έχουν την δυνατότητα να συνδέονται ασύρματα στην πλατφόρμα για να μπορούμε έτσι να καλύψουμε μία μεγάλη έκταση και να την ελέγξουμε (πχ ένα δάσος). Η ενσωμάτωση διαφορετικών αισθητήρων όπως είναι οι αισθητήρες μέτρησης θερμοκρασίας, υγρασίας, ατμοσφαιρικής πίεσης, ταχύτητας, διεύθυνσης του ανέμου κτλ θα μπορούσε να μετατρέψει την εφαρμογή σε ένα μετεωρολογικό σταθμό. Επίσης μία μελλοντική εξέλιξη της πλατφόρμας θα ήταν να χρησιμοποιηθεί σε διάφορα συστήματα αυτοματισμού όπως για παράδειγμα σε ένα σύστημα αυτόματης άρδευσης ή σε αυτοματισμούς που αφορούν την κεντρική διαχείριση διάφορων συστημάτων μιας κατοικίας (έξυπνο σπίτι) κτλ.



Ερωτήσεις;



Ευχαριστώ!



Περίληψη Π.Ε. σε στυλ δημοσίευσης
Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης



Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Εφαρμοσμένης Πληροφορικής &
Πολυμέσων



Πτυχιακή Εργασία

Τίτλος:

Ανάπτυξη πλατφόρμας Ασύρματου Δικτύου Αισθητήρων (WSN) για την συλλογή περιβαλλοντικών δεδομένων και επεξεργασία με εφαρμογή Android.

Βασιλικός Γεώργιος (ΑΜ:1362)

1 Εισαγωγή

Αντικείμενο της παρούσας πτυχιακής εργασίας είναι η ανάπτυξη πλατφόρμας Ασύρματου Δικτύου Αισθητήρων (WSN) για την συλλογή περιβαλλοντικών δεδομένων και επεξεργασία με εφαρμογή Android. Αυτή η τεχνολογία μας παρέχει την δυνατότητα να αποκτήσουμε σε πραγματικό χρόνο διάφορα περιβαλλοντικά δεδομένα τα οποία μπορούμε να τα επεξεργαστούμε άμεσα χωρίς να είναι απαραίτητο να είμαστε στο ίδιο γεωγραφικό σημείο από το οποίο έγινε η συλλογή τους. Επίσης η πλατφόρμα αυτή έχει την επιπλέον δυνατότητα να καταχωρεί τα δεδομένα σε μια βάση δεδομένων ώστε να μπορούμε να ανακτήσουμε παλαιότερες μετρήσεις καθώς και να στέλνει αυτόματα ένα ηλεκτρονικό μήνυμα συναγερμού σε έναν ή περισσότερους παραλήπτες.

2 Περίληψη

Στην πτυχιακή αυτή θα χρησιμοποιήσουμε κάποιες γλώσσες προγραμματισμού για να προγραμματίσουμε το υλικό και το λογισμικό με βάση τα οποία θα αναπτυχθεί η πλατφόρμα. Το υλικό, το λογισμικό και οι γλώσσες προγραμματισμού αναφέρονται αναλυτικά στα επόμενα κεφάλαια.

Αρχικά θα αξιοποιήσουμε τις δυνατότητες χαμηλού επιπέδου της ηλεκτρονική πλατφόρμας Arduino καθώς και των αισθητήρων LM35 και DHT11 ώστε να συλλέξουμε τα δεδομένα τα οποία μας ενδιαφέρουν. Έπειτα με τη χρήση του κατάλληλου λογισμικού που αναφέρετε σε επόμενο κεφάλαιο, ενός ηλεκτρονικού

υπολογιστή ο οποίος λειτουργεί σαν διακομιστής και ενός ασύρματου δικτύου θα αποθηκεύσουμε τα δεδομένα μας σε μια βάση δεδομένων. Μετά θα δημιουργήσουμε την εφαρμογή Android την οποία και θα εγκαταστήσουμε στο κινητό τηλέφωνο. Τέλος θα δούμε με ποιο λογισμικό και ποιες γλώσσες προγραμματισμού θα καταφέρουμε να κάνουμε εφικτή την επικοινωνία ανάμεσα στους αισθητήρες, την ηλεκτρονική πλατφόρμα, τον διακομιστή, την βάση δεδομένων, το κινητό τηλέφωνο και το ηλεκτρονικό ταχυδρομείο.

3 Θεωρία

Τα ασύρματα δίκτυα αισθητήρων αποτελούν μια από τις πολλά υποσχόμενες τεχνολογίες στην παροχή υποδομής για διεργασίες παρακολούθησης κατάστασης. Η μεγάλη εξάπλωση μικρών ασύρματων συσκευών, επιτρέπει την προσπέλαση της πληροφορίας οποιαδήποτε στιγμή και από παντού. Με τα ασύρματα δίκτυα αισθητήρων μπορούν να δημιουργηθούν επεκτάσιμα περιβάλλοντα παρακολούθησης, καθώς η προσθήκη κόμβων είναι άμεση.

4 Περιβάλλον Σχεδίασης, Υλικό - Λογισμικό

Το λειτουργικό σύστημα που θα χρησιμοποιηθεί είναι το Windows 7. Το υλικό που απαιτείτε είναι το εξής:

- Ηλεκτρονικός Υπολογιστής .

- Ηλεκτρονική πλακέτα Arduino uno.
- Αισθητήρας θερμοκρασίας LM35.
- Αισθητήρας υγρασίας DHT11 .
- Καλώδιο σύνδεσης περιφερικών τύπου USB.
- Κινητό τηλέφωνο τύπου SmartPhone με λειτουργικό Android.

Το λογισμικό που απαιτείτε είναι το εξής:

- Arduino 1.0.1
- Processing 2.0b5
- Xampp-win32-01.8.1-VC9 (Apache, MySQL, Mercury)
- Eclipse 4.2.1(Android)

Και οι γλώσσες προγραμματισμού και συγγραφής σεναρίων είναι οι εξής:

- C++
- JAVA
- PHP
- SQL
- Wiring

5 Arduino uno

Το Arduino uno είναι μια υπολογιστική πλατφόρμα βασισμένη σε μια απλή μητρική πλακέτα με ενσωματωμένο μικροελεγκτή και εισόδους/εξόδους, και η οποία μπορεί να προγραμματιστεί με τη γλώσσα Wiring (ουσιαστικά πρόκειται για τη C++ με κάποιες μετατροπές). Το Arduino μπορεί να χρησιμοποιηθεί για την ανάπτυξη ανεξάρτητων διαδραστικών αντικειμένων αλλά και να συνδεθεί με υπολογιστή μέσω προγραμμάτων σε Processing, Max/MSP, Pure Data, SuperCollider. Οι περισσότερες εκδόσεις του Arduino μπορούν να αγοραστούν προ-

συναρμολογημένες, το διάγραμμα και πληροφορίες για το υλικό είναι ελεύθερα διαθέσιμα για αυτούς που θέλουν να συναρμολογήσουν το Arduino μόνοι τους.

6 Αισθητήρας θερμοκρασίας Lm35

Το LM35 είναι ένας αισθητήρας θερμοκρασίας, ακριβείας, σε ολοκληρωμένη μορφή. Η τάση εξόδου από τον αισθητήρα είναι γραμμικά ανάλογη της θερμοκρασίας Celsius. Έτσι το LM35 έχει ένα πλεονέκτημα σε σχέση με όλους εκείνους τους αισθητήρες στους οποίους η τάση εξόδου είναι ανάλογη της θερμοκρασίας Kelvin, αφού ο χρήστης δεν χρειάζεται να αφαιρεί μια σταθερή αλλά μεγάλη τάση για να προσαρμοστεί στην κλίμακα Celsius. Το LM35 δεν χρειάζεται κανένα εξωτερικό ρυθμιστικό κύκλωμα για να προσφέρει ακρίβεια (+-) 1/4 C σε θερμοκρασία δωματίου και (+-)3/4 C σε φουλ κλίμακα θερμοκρασίας από -55 σε +150 C. Η μικρή σύνθετη αντίσταση εξόδου, η γραμμικότητά της και το εσωτερικό ρυθμιστικό κύκλωμα κάνουν πολύ εύκολη την προσαρμογή του αισθητήρα στο μικροελεγκτή. Μπορεί να πάρει απλή τροφοδοσία ή συμμετρική. Καταναλώνει πολύ μικρό ρεύμα 60 μ A και έτσι αναπτύσσει πολύ μικρή θερμοκρασία μόλις 0.1C .

7 Αισθητήρας θερμοκρασίας – Υγρασίας DHT11

Ο ψηφιακός αισθητήρας θερμοκρασίας και υγρασίας DHT11 είναι ένας σύνθετος αισθητήρας ο οποίος περιέχει μια βαθμολογημένη

παραγωγή ψηφιακών σημάτων της θερμοκρασίας και της υγρασίας. Ο αισθητήρας περιλαμβάνει πολύ καλή αίσθηση των υγρών συστατικών και της μέτρησης της θερμοκρασίας. Είναι συνδεδεμένος με έναν υψηλής απόδοσης οκτάμπιτο μικροελεγκτή. Διαθέτει ενιαία τμηματική διεπαφή καλωδίων, και βοηθάει στην ολοκλήρωση συστημάτων γρήγορα και εύκολα. Έχει μικρό μέγεθος, χαμηλής ισχύος κατανάλωση και απόσταση μετάδοσης σημάτων μέχρι 20 μέτρα.

Χαρακτηριστικά:

- Τάση παροχής ηλεκτρικού ρεύματος: +5 V
- Σειρά θερμοκρασίας: 0-50 °C λάθος ± 2 °C
- Σειρά υγρασίας: 20-90% λάθος $\pm 5\%$ RH
- Ακολουθία γραμμών διεπαφών: VCC, GND, S
- Μέγεθος : 3 x 2 cm

8 Κινητό τηλέφωνο SmartPhone (android)

Με τα σύγχρονα κινητά τηλέφωνα μπορούμε πλέον να κάνουμε πολλά περισσότερα πράγματα από το να τηλεφωνούμε και να γράφουμε μηνύματα SMS. Τα λεγόμενα Smartphone (αγγλική λέξη που σημαίνει «έξυπνο τηλέφωνο») καθιερώθηκαν ως η νέα τάση της μόδας. Κινητά τηλέφωνα με πλήρεις δυνατότητες πρόσβασης στο Internet και με εξαιρετικά ευρύ φάσμα λειτουργιών, όπως σε ένα φορητό υπολογιστή, τα οποία λειτουργούν με λειτουργικό σύστημα και επιτρέπουν την εγκατάσταση εφαρμογών πολυμέσων. Ο χειρισμός των περισσότερων Smartphone γίνεται συνήθως μέσω μιας οθόνης αφής (Touchscreen), ενώ μερικά από αυτά

διαθέτουν και μικρών διαστάσεων πληκτρολόγιο QWERTZ, όπως ένας υπολογιστής.

Το Android είναι λειτουργικό σύστημα για συσκευές κινητής τηλεφωνίας το οποίο τρέχει τον πυρήνα του λειτουργικού Linux. Αρχικά αναπτύχθηκε από την Google και αργότερα από την Open Handset Alliance. Επιτρέπει στους κατασκευαστές λογισμικού να συνθέτουν κώδικα με την χρήση της γλώσσας προγραμματισμού Java, ελέγχοντας την συσκευή μέσω βιβλιοθηκών λογισμικού ανεπτυγμένων από την Google.

9 Arduino 1.0.1

Το Arduino 1.0.1 είναι ένα περιβάλλον προγραμματισμού ανοικτού κώδικα. Μπορεί να εγκατασταθεί στα λειτουργικά συστήματα των Windows, Mac OS X και Linux και η γλώσσα προγραμματισμού που χρησιμοποιεί ονομάζεται Wiring (ουσιαστικά πρόκειται για τη C++ με κάποιες μετατροπές).

10 Processing 2.0b5

Το processing είναι ένα περιβάλλον προγραμματισμού ανοικτού κώδικα και χρησιμοποιείτε περισσότερο από τους ανθρώπους που θέλουν να δημιουργήσουν εικόνες, κινούμενες εικόνες και αλληλεπιδράσεις. Αρχικά αναπτύχθηκε για να διδάξει βασικές αρχές του προγραμματισμού ηλεκτρονικών υπολογιστών σε οπτικό περιβάλλον, αργότερα όμως εξελίχθηκε σε εργαλείο για τη δημιουργία επαγγελματικών εργασιών. Σήμερα, υπάρχουν δεκάδες χιλιάδες φοιτητές, καλλιτέχνες, σχεδιαστές, ερευνητές και ερασιτέχνες που χρησιμοποιούν το processing για

εκπαίδευση, δημιουργία προτύπων και παραγωγή προγραμμάτων.

11 Xampp-win32-01.8.1-VC9

Το XAMPP είναι ένα πακέτο προγραμμάτων ελεύθερου λογισμικού, λογισμικού ανοικτού κώδικα και ανεξαρτήτου πλατφόρμας το οποίο περιέχει τον εξυπηρετητή ιστοσελίδων http Apache, την βάση δεδομένων MySQL, την πλατφόρμα ηλεκτρονικού ταχυδρομείου Mercury και ένα διερμηνέα για κώδικα γραμμένο σε γλώσσες προγραμματισμού PHP και Perl. Επίσης οι σχεδιαστές του XAMPP προόριζαν το λογισμικό ως εργαλείο ανάπτυξης και δοκιμής ιστοσελίδων τοπικά στον υπολογιστή χωρίς να είναι απαραίτητη η σύνδεση στο διαδίκτυο.

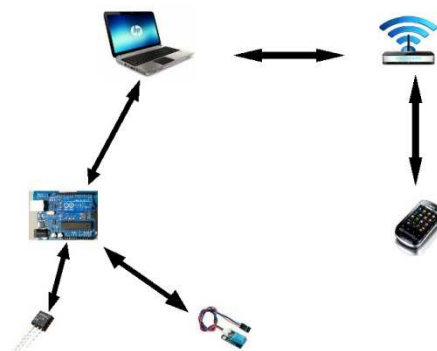
12 Eclipse 4.2.1 (Android)

Το Eclipse αποτελεί ένα SDK (Software Development Kit), ένα ολοκληρωμένο δηλαδή περιβάλλον μέσα από το οποίο μπορούμε να γράψουμε και να εκτελέσουμε κώδικα. Το περιβάλλον αυτό καθώς και όλα τα υπόλοιπα προγράμματα που χρειάζονται για να εκτελέσουμε κώδικα σε Java ή/και C/C++ είναι ελεύθερης διανομής (freeware) και ανοικτού κώδικα (open source). Επιπλέον αποτελεί το ανερχόμενο περιβάλλον ανάπτυξης κώδικα, καθώς χρησιμοποιείται από ολοένα και περισσότερους χρήστες αλλά και εταιρείες. Η ανάπτυξη της εφαρμογής Android γίνεται μέσω του Eclipse ή άλλων συμβατών προγραμμάτων. Στην παρούσα πτυχιακή εργασία θα χρησιμοποιήσουμε το Android Developer Tools (ADT) το οποίο είναι

πρόσθετο λογισμικό του Eclipse ώστε να δημιουργήσουμε την εφαρμογή μας και να είναι συμβατή με το λογισμικό Android το οποίο χρησιμοποιεί το SmartPhone.

13 Σχεδιασμός Υλοποίησης

Σε αυτό το κεφάλαιο θα σχεδιάσουμε και θα αναλύσουμε τον τρόπο υλοποίησης της πτυχιακής εργασίας. Εφόσον έχουμε στην κατοχή μας όλα αυτά που αναφέραμε στο προηγούμενο κεφάλαιο, τα οποία μας είναι απαραίτητα για την υλοποίηση του πρακτικού μέρους της πτυχιακής εργασίας, θα ξεκινήσουμε να αναπτύξουμε την πλατφόρμα του ασύρματου δικτύου αισθητήρων όπως φαίνεται στην εικόνα 1 διασπώντας τον σχεδιασμό μας σε κομμάτια.



Εικόνα 1: Πρότυπο πλατφόρμας WSN και επεξεργασίας με εφαρμογή android

1^ο Βήμα

Αρχικά συνδέουμε τους αισθητήρες LM35 και DHT11 στην ηλεκτρονική πλατφόρμα Arduino uno. Έπειτα με την χρήση του καλωδίου σύνδεσης περιφερειακών τύπου USB πραγματοποιούμε την τροφοδοσία της ηλεκτρονικής πλατφόρμας και την επικοινωνία της με το λειτουργικό Arduino 1.0.1 το οποίο εγκαθιστούμε

στον ηλεκτρονικό υπολογιστή. Με την γλώσσα προγραμματισμού Wiring θα προγραμματίσουμε την ηλεκτρονική πλακέτα Arduino υπο όστε να επικοινωνήσει με τους αισθητήρες και να συλλέξουμε τα δεδομένα που θέλουμε.

2^ο Βήμα

Με την χρήση του λογισμικού Xampp θα δημιουργήσουμε έναν εικονικό εξυπηρετητή- διακομιστή ο οποίος μας παρέχει την δυνατότητα να έχουμε πρόσβαση σε μια βάση δεδομένων που θα χρησιμοποιήσουμε για την αποθήκευση των δεδομένων μας και επίσης την δημιουργία μίας εικονικής πλατφόρμας ηλεκτρονικού ταχυδρομείου που θα χρησιμοποιήσουμε για να στέλνουμε ένα μήνυμα συναγερμού σε περίπτωση που υπάρχει μια ακραία μέτρηση από τους αισθητήρες μας.

Η επικοινωνία του διακομιστή με την ηλεκτρονική πλατφόρμα Arduino υπο, η αυτόματη καταχώρηση των στοιχείων στην βάση δεδομένων μας, η αποστολή μηνύματος συναγερμού και η παρουσίαση των δεδομένων στο πρόγραμμα περιήγησης στο διαδίκτυο (browser) θα επιτευχθεί με την διαμεσολάβηση των λογισμικών Arduino 1.0.1, Processing 2.0b5, Xampp ,τις γλώσσες προγραμματισμού C++, SQL , την γλώσσα συγγραφής σεναρίων PHP και την παροχή υπηρεσιών διαδικτύου(Internet).

3^ο Βήμα

Το επόμενο στάδιο περιλαμβάνει την δημιουργία της εφαρμογής Android, την εγκατάσταση της στο κινητό τηλέφωνο καθώς και

την επικοινωνία της με την βάση δεδομένων μας. Αυτό σημαίνει ότι η εφαρμογή Android θα πρέπει να ενσωματώνει την επικοινωνία της με το διαδίκτυο, την αδειοδότηση και επικοινωνία της με τον διακομιστή και την βάση δεδομένων, την αντιγραφή των στοιχείων από την βάση δεδομένων, την εμφάνιση τους στην οθόνη του κινητού τηλεφώνου και τέλος την αποστολή ενός μηνύματος συναγερμού σε περίπτωση που υπάρξει κάποια ακραία μέτρηση των αισθητήρων.

Όλα αυτά θα υλοποιηθούν με την χρήση του λογισμικού Eclipse 4.2.1, Android Developer Tools, Xampp, την γλώσσα προγραμματισμού JAVA, την γλώσσα συγγραφής σεναρίων PHP και την παροχή υπηρεσιών διαδικτύου(Internet).

4ο Βήμα

Τέλος θα πρέπει να συντονίσουμε τα τρία προηγούμενα στάδια και να τα ενώσουμε με τέτοιο τρόπο ώστε να έχουμε το τελικό αποτέλεσμα στο πρακτικό τμήμα της πτυχιακής εργασίας.

14 Βιβλιογραφία

- [1] <http://welcom-project.ipet.gr>
- [2] <http://arduino.cc>
- [3] <http://processing.org>
- [4] <http://el.wikipedia.org>
- [5] <http://www.mediamarkt.gr/mp/article/Smartphone,872008.h>

