



Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης  
Σχολή Τεχνολογικών Εφαρμογών  
Τμήμα Μηχανικών Πληροφορικής



Τίτλος πτυχιακής  
εργασίας:  
ΥΛΟΠΟΙΗΣΗ WSN ΜΕ  
ARDUINO ΚΑΙ RASPBERRY

Κιουρτσιδάκης Νικόλαος (ΑΜ: 3363)

Ραπτάκης Πολυχρόνιος (ΑΜ:3203)

Επιβλέπων καθηγητής: Παναγιωτάκης Σπυρίδων

ΗΡΑΚΛΕΙΟ ΙΑΝΟΥΑΡΙΟΣ 2018

**Copyright © Νίκος Κιουρτσιδάκης – Πολυχρόνιος Ραπτάκης, 2018**

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Η έγκριση της πτυχιακής εργασίας από το Τμήμα Μηχανικών Πληροφορικής του ΤΕΙ Κρήτης δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων των συγγραφέων και του Τμήματος.

### **Ευχαριστίες**

Θα θέλαμε να ευχαριστήσουμε τις οικογένειες μας, που μας στήριξαν όλα αυτά χρόνια στις σπουδές μας και μας προσέφεραν τα απαραίτητα εφόδια για την ολοκλήρωσή τους.

Επίσης αισθανόμαστε την ανάγκη να ευχαριστήσουμε τον καθηγητή μας κύριο Παναγιωτάκη Σπυρίδων για την υπομονή του και την καθοδήγηση που μας προσέφερε.

## **Περίληψη**

Στην παρούσα πτυχιακή εργασία έχει αναπτυχθεί ένα δίκτυο ασύρματων αισθητήρων το οποίο συλλέγει, αποθηκεύει και παρουσιάζει τα δεδομένα σε πραγματικό χρόνο με τη βοήθεια της πλατφόρμας Arduino και Raspberry.

Τα δεδομένα, όπως θερμοκρασία, υγρασία, απόσταση, ένταση φωτός και κίνηση συλλέγονται από αισθητήρες τοποθετημένους πάνω σε Arduino, αποστέλλονται στο Raspberry, όπου αποθηκεύονται και παρουσιάζονται με γραφήματα.

**Abstract**

In this thesis has been developed a wireless sensors network, which collect, store and preset the data in real time with the help of the Arduino and Raspberry platform.

Data such as temperature, humidity, distance, light intensity and motion are collected by sensors mounted on Arduino, sent to Raspberry, where they are stored and displayed in graphs.

## Πίνακας Περιεχομένων:

Πίνακας Περιεχομένων:.....	6
Πίνακας Εικόνων:.....	8
ΚΕΦΑΛΑΙΟ 1. Εισαγωγή .....	11
1.1. Εισαγωγή.....	11
1.2. Δομή.....	12
ΚΕΦΑΛΑΙΟ 2. θεωρητικό Υπόβαθρο.....	13
2.1. Τεχνολογία αισθητήρων .....	13
2.1.1. Εισαγωγή.....	13
2.1.2 Ανιχνευτές Κίνησης & Παρουσίας (Occupancy sensors).....	14
2.1.3 Έξυπνοι Αισθητήρες.....	15
2.2 Δίκτυα Ασύρματων Αισθητήρων .....	17
2.2.1. Ασύρματα δίκτυα αισθητήρων (WSN) .....	19
2.2.2. Περιγραφή του ασύρματου κόμβου αισθητήρα.....	21
2.2.3. Διαφοροποίηση των WSN σε σχέση με τα υπόλοιπα δίκτυα .....	25
2.2.4. Τοπολογίες WSN.....	27
2.3. Εφαρμογές Δικτύων Αισθητήρων.....	30
2.3.1. Υφιστάμενες εφαρμογές .....	30
2.3.2. Στρατιωτικές Εφαρμογές .....	31
2.3.3. Περιβαλλοντολογικές Εφαρμογές.....	31
2.3.4. Ανίχνευση δασικών πυρκαγιών.....	32
2.3.5. Ανίχνευση πλημμυρών .....	32
2.3.6 Γεωργία Ακρίβειας (Precision Agriculture) .....	33
2.3.7 Εφαρμογές Υγείας.....	33
2.3.8. Οικιακές Εφαρμογές.....	33
2.3.8.1. Αυτοματισμός σπιτιού .....	33
2.3.9. Αλληλεπιδραστικά Μουσεία .....	34
2.3.10. Παρακολούθηση και ανίχνευση οχημάτων .....	35
ΚΕΦΑΛΑΙΟ 3. Τεχνολογίες Υλοποίησης .....	36
3.1. HTML5 .....	36
3.2. Javascript.....	37

3.3. CSS3.....	38
3.4. jQuery.....	39
3.5. Bootstrap.....	39
3.6. Grafana.....	40
3.7. Arduino IDE .....	42
3.8. Python .....	43
3.9. Raspbian.....	44
3.10. Apache HTTP εξυπηρετητής .....	45
3.11. MySQL .....	46
3.12. Raspberry Pi .....	46
3.12.1. Raspberry Pi 3 .....	49
3.13. Arduino .....	53
Κεφάλαιο 4. Υλοποίηση Δικτύου Αισθητήρων.....	56
4.1. Ανάλυση Προβλήματος .....	56
4.1.1. Απαιτήσεις Συστήματος .....	57
4.2. Hardware που χρησιμοποιήθηκε .....	57
4.3. RF Modules NRF24L01 & NRF24L01+ .....	59
4.4. Arduino με NRF24L01 .....	60
4.4.1. Κώδικας που χρησιμοποιήθηκε για τη σύνδεση arduino και nrf24l01: .....	60
4.5. Raspberry Pi 3 με NRF24L01 .....	61
4.5.1. Κώδικας που χρησιμοποιήθηκε για το Raspberry Pi 3, για να λαμβάνει τα δεδομένα και να τα καταχωρεί στη σχεσιακή βάση: .....	62
4.6. Nodes – Συνδεσμολογία - Κώδικας .....	64
4.6.1. Node με αισθητήρα μέτρησης απόστασης .....	64
4.6.2. Συνδεσμολογία για το node του αισθητήρα απόστασης:.....	68
4.6.3. Κώδικας που χρησιμοποιήθηκε για τον αισθητήρα απόστασης: .....	69
4.7. Αισθητήρας Ανίχνευσης Κίνησης.....	72
4.7.1. Συνδεσμολογία για το node του αισθητήρα κίνησης: .....	73
4.7.2. Κώδικας που χρησιμοποιήθηκε για τον αισθητήρα κίνησης:.....	73
4.8. Αισθητήρας Θερμοκρασίας/Υγρασίας .....	77
4.8.1. Συνδεσμολογία για το node του αισθητήρα θερμοκρασίας/υγρασίας:.....	80
4.8.2. Κώδικας που χρησιμοποιήθηκε για τον αισθητήρα θερμοκρασίας/υγρασίας: .....	80
4.9. Αισθητήρας Βροχής .....	84

4.9.1. Συνδεσμολογία για το node του αισθητήρα βροχής: .....	86
4.9.2. Κώδικας που χρησιμοποιήθηκε για τον αισθητήρα βροχής:.....	86
4.10. Αισθητήρας φωτός .....	89
4.10.1. Συνδεσμολογία για node του αισθητήρα φωτός:.....	90
4.10.2. Κώδικας που χρησιμοποιήθηκε για τον αισθητήρα φωτός:.....	90
4.11. Σύνδεση Raspberry Pi 3– Διαδίκτυο .....	93
4.12. Χρήστες .....	94
4.12.1. Χρήστης Διαχειριστής.....	94
4.12.2. Χρήστης Αναζήτησης .....	94
4.13. Περιγραφές χρήσης (Use Case Descriptions) .....	95
4.14. Σχεδιασμός Οθονών (Grafana) .....	96
4.15. Ενσωμάτωση Portal .....	98
4.16. Σενάρια χρήσης/Εκσφαλμάτωση .....	99
4.16.1. Εκσφαλμάτωση Οθονών.....	99
Κεφάλαιο 5. Συμπεράσματα & Μελλοντικές Επεκτάσεις .....	110
5.1. Συμπεράσματα.....	110
5.2. Μελλοντικές Επεκτάσεις.....	110
Βιβλιογραφία .....	111

## Πίνακας Εικόνων:

Εικόνα 1 - Ασύρματα Δίκτυα Αισθητήρων.....	17
Εικόνα 2 - Αρχιτεκτονική WSN.....	21
Εικόνα 3 - Δομικά στοιχεία κόμβου-αισθητήρα.....	22
Εικόνα 4 - Αρχιτεκτονική κόμβου-αισθητήρα .....	23
Εικόνα 5 - Τοπολογία Σημείο προς Σημείο .....	27
Εικόνα 6 - Τοπολογία Δέντρου .....	28
Εικόνα 7 - Τοπολογία Αστέρα .....	28
Εικόνα 8 - Τοπολογία Πλέγματος .....	29
Εικόνα 9 - Εφαρμογές δικτύων Αισθητήρων .....	30
Εικόνα 10 - HTML5 .....	36
Εικόνα 11 - Javascript.....	37
Εικόνα 12 - CSS3.....	38
Εικόνα 13 - jQuery.....	39
Εικόνα 14 - Bootstrap .....	39
Εικόνα 15 - Grafana .....	40



Εικόνα 16 - Grafana Graphs - Dashboard .....	41
Εικόνα 17 – Arduino IDE .....	42
Εικόνα 18 – Arduino IDE .....	42
Εικόνα 19 - Python .....	43
Εικόνα 20 - Raspbian.....	44
Εικόνα 21 - Apache Web Server.....	45
Εικόνα 22 – MySQL .....	46
Εικόνα 23 - Raspberry pi & pi 2.....	48
Εικόνα 24 - Raspberry Pi 3 .....	49
Εικόνα 25 - Raspberry Pi 3 GPIO .....	51
Εικόνα 26 - Raspberry Pi Heatsinks.....	52
Εικόνα 27 - Σύγκριση Pi2 & Pi3 .....	52
Εικόνα 28 - Arduino .....	53
Εικόνα 29 - Συνδεσμολογία - Επικοινωνία .....	56
Εικόνα 30 - Τα modules NRF24L01 και L01+ .....	59
Εικόνα 31 - Συνδεσμολογία arduino – nrf24l01 .....	60
Εικόνα 32 - Συνδεσμολογία raspberry Pi 3 – nrf24l01 .....	61
Εικόνα 33 – Συνδεσμολογία Raspberry Pi .....	63
Εικόνα 34 - Αρχή των Sonar .....	65
Εικόνα 35 - HC-SR04.....	66
Εικόνα 36 - Διαστάσεις Συσκευής HC-SR04.....	67
Εικόνα 37 - Διαγράμματα Χρονισμού HC-SR04.....	68
Εικόνα 38 – Σύνδεση Arduino-αισθητήρα απόστασης .....	68
Εικόνα 39 - Arduino - Αισθητήρας απόστασης.....	71
Εικόνα 40 - HC-SR501 PIR Motion Detector .....	72
Εικόνα 41 - Σύνδεση Arduino - HC-SR501.....	73
Εικόνα 42 - Arduino – Αισθητήρας κίνησης.....	76
Εικόνα 43 - DHT.....	77
Εικόνα 44 - Διαδικασία επικοινωνίας Arduino-DHT.....	79
Εικόνα 45 - Συνδεσμολογία arduino – DHT .....	80
Εικόνα 46 - Arduino - Αισθητήρας θερμοκρασίας.....	84
Εικόνα 47 - Αισθητήρας Βροχής .....	85
Εικόνα 48 - Συνδεσμολογία arduino - αισθητήρα βροχής .....	86
Εικόνα 49 - Arduino - Αισθητήρας βροχής .....	88
Εικόνα 50 – Αισθητήρας φωτός.....	89
Εικόνα 51 - Συνδεσμολογία arduino - αισθητήρα φωτός .....	90
Εικόνα 52 - Arduino - Αισθητήρας φωτός .....	92
Εικόνα 53 - Raspberry Pi 3 .....	93
Εικόνα 54 - Κεντρική σελίδα του portal.....	97
Εικόνα 55 – Οθόνη με τιμές ενός αισθητήρα.....	98
Εικόνα 56 - Κεντρική οθόνη ηλεκτρονικής πύλης .....	100
Εικόνα 57 - Οθόνη αισθητήρα θερμοκρασίας/υγρασίας.....	100
Εικόνα 58 - Οθόνη αισθητήρα απόστασης.....	101
Εικόνα 59 - Οθόνη αισθητήρα φωτός .....	101

Εικόνα 60 - Οθόνη αισθητήρα βροχής .....	102
Εικόνα 61 - Οθόνη αισθητήρα κίνησης .....	102
Εικόνα 62 – Οθόνες (smartphone) αισθητήρα θερμοκρασίας .....	103
Εικόνα 63 – Οθόνη (smartphone) αισθητήρα βροχής .....	104
Εικόνα 64 - Οθόνη παραμετροποίησης grafana.....	104
Εικόνα 65 - Οθόνες αισθητήρα ανίχνευσης κίνησης .....	105
Εικόνα 66 - Εικόνες αισθητήρα DHT .....	107
Εικόνα 67 - Οθόνες μετρήσεις αισθητήρων .....	109

## ΚΕΦΑΛΑΙΟ 1. Εισαγωγή

### 1.1. Εισαγωγή

Τα Ασύρματα Δίκτυα Αισθητήρων (WSN), είναι σήμερα ένας ταχύτατα αναπτυσσόμενος και πολλά υποσχόμενος τομέας στην έρευνα των ασύρματων επικοινωνιών, λόγω της τεράστιας γκάμας εφαρμογών που μπορεί να έχουν. Τα Ασύρματα Δίκτυα Αισθητήρων είναι ένα καινούριο κομμάτι υπολογιστικών μονάδων, που ερευνά την δυνατότητα του ανθρώπου να αλληλοεπιδρά με περιβάλλον γύρω του. Τα χαρακτηριστικά που διαφοροποιούν τα Ασύρματα Δίκτυα Αισθητήρων από τα ad hoc δίκτυα είναι η δυνατότητα αίσθησης του περιβάλλοντος, η αυτόνομη λειτουργία τους χωρίς την ανθρώπινη παρουσία, το χαμηλό τους κόστος και το μικρό τους μέγεθος. Τα χαρακτηριστικά αυτά δίνουν το έναυσμα για νέες εφαρμογές και δημιουργούν νέες προοπτικές. Ωστόσο, κάποια από τα χαρακτηριστικά αυτά περιορίζουν τα Ασύρματα Δίκτυα Αισθητήρων σε μικρά ενεργειακά αποθέματα, μνήμη, υπολογιστική ισχύ και μικρή ισχύ εκπομπής.

Όλα τα παραπάνω γεννούν νέα αντικείμενα έρευνας που σκοπό έχουν τη μεγιστοποίηση της απόδοσης του δικτύου. Ένα από τα αντικείμενα αυτά είναι η συνολική συνδεσιμότητα του δικτύου, όσον αφορά την σωστή επικοινωνία ανάμεσα στους κόμβους και της μεταφοράς των πληροφοριών σχετικά με το αντικείμενο παρατήρησης. Η ολοένα και αυξανόμενη χρησιμότητα των Ασύρματων Δικτύων Αισθητήρων και η ανάγκη για συνδεσιμότητα αποτέλεσαν το έναυσμα για να ξεκινήσουμε την παρούσα πτυχιακή εργασία.

Σκοπός της πειραματικής διαδικασίας ήταν η δημιουργία ενός ασύρματου δικτύου αισθητήρων το οποίο αποτελείτο από 5 αισθητήρες. Έναν αισθητήρα θερμοκρασίας-υγρασίας, έναν αισθητήρα απόστασης, έναν αισθητήρα βροχής, έναν αισθητήρα κίνησης και έναν αισθητήρα φωτός/φωτιάς. Οι αισθητήρες είναι συνδεδεμένοι με πλακέτες arduino. Η μετάδοση των τιμών των αισθητήρων και η επικοινωνία επιτυγχάνεται μέσω ασύρματων αναμεταδοτών nrf24l01. Οι τιμές των αισθητήρων αποθηκεύονται σε μία σχεσιακή βάση δεδομένων η οποία φιλοξενείται σε πλατφόρμα raspberry. Κίνητρο μας για την διεξαγωγή της παρούσας διπλωματικής εργασίας είναι η εξοικείωση μας με τεχνολογίες αιχμής και η απόκτηση δεξιοτήτων για τον σχεδιασμό και

ανάπτυξη ηλεκτρονικών πυλών και εφαρμογών για πλατφόρμες όπως το Arduino και το Raspberry. Πιο συγκεκριμένα θα εξοικειωθούμε με τεχνολογίες όπως το πλαίσιο εφαρμογής Bootstrap, το CCS3, η γλώσσα HTML5, το Grafana, την Javascript, την JQuery, την σχεσιακή βάση δεδομένων mysql 5 και το περιβάλλον ανάπτυξης Arduino ide.

## **1.2. Δομή**

Στο δεύτερο κεφάλαιο της εργασίας περιγράφεται η τεχνολογία των αισθητήρων. Πιο συγκεκριμένα το τι είναι αισθητήρας, πως λειτουργεί, τι μετράει, τι είναι ευφυή δίκτυο, τι είναι δίκτυο αισθητήρων και τι είναι το διαδίκτυο των πραγμάτων.

Στα επόμενα δύο κεφάλαιο παρουσιάζονται οι τεχνολογίες και τα υλικά τα οποία χρησιμοποιήθηκαν στην πειραματική διαδικασία. Δηλαδή, περιγράφονται οι τεχνολογίες html5, bootstrap, javascript, jquery, grafana, mysql. Επιπλέον, περιγράφονται υλικά τα οποία χρησιμοποιήθηκαν όπως η πλακέτα raspberry και arduino, ο ασύρματος αναμεταδότης nrf24l01 και αισθητήρες θερμοκρασίας-υγρασίας, κίνησης, απόστασης, βροχής και φωτός/φωτιάς.

Στο επόμενο κεφάλαιο, παρουσιάζονται όλες οι φάσεις ανάπτυξης της ηλεκτρονικής πύλης. Από την συλλογή των απαιτήσεων έως την εκσφαλμάτωση και την εκτέλεση δοκιμών. Το πέμπτο κεφάλαιο είναι το τελευταίο. Σε αυτό παρουσιάζονται συμπεράσματα και προτείνονται βελτιώσεις και μελλοντικές προεκτάσεις.

## **ΚΕΦΑΛΑΙΟ 2. Θεωρητικό Υπόβαθρο**

### **2.1. Τεχνολογία αισθητήρων**

#### **2.1.1. Εισαγωγή**

Αισθητήρας (sensor), ονομάζεται μια ηλεκτρονική διάταξη, η οποία αντιλαμβάνεται τα φυσικά μεγέθη του περιβάλλοντος στο οποίο βρίσκεται και μας δίνει μια μετρήσιμη έξοδο. Οι αισθητήρες καθιστούν ικανές τις συσκευές που χρησιμοποιούμε καθημερινά, να βλέπουν (πχ κάμερα), να ακούνε (πχ μικρόφωνο) και μετρούν καθετί στο φυσικό κόσμο γύρω τους. Αυτό οι αισθητήρες το καταφέρνουν χρησιμοποιώντας τις κατάλληλες διεπαφές (interfaces), ώστε να μπορούν να μετατρέπουν τα αναλογικά σήματα του φυσικού κόσμου σε ηλεκτρικά σήματα, κατανοητά από τον οποιοδήποτε ηλεκτρονικό εξοπλισμό.

Τα τελευταία χρόνια, η τεράστια εξέλιξη στον χώρο της τεχνολογίας, απαιτεί τη μείωση των διαστάσεων των ηλεκτρονικών συσκευών και την ενσωμάτωση σε αυτές όσον το δυνατόν περισσότερων δυνατοτήτων. Εξ' αιτίας των εξελίξεων αυτών, οι απαιτήσεις ανάγκασαν και τον κλάδο των ερευνητών που ασχολούνται με τους αισθητήρες, να τους συρρικνώσουν ώστε να μπορούν με ευκολία να ενσωματωθούν οπουδήποτε. Έτσι σήμερα έχουμε τους μικροαισθητήρες, οι οποίοι είναι εξαιρετικά μικροί αισθητήρες μερικών μικρόμετρων ο καθένας. Η εξέλιξη όμως δεν επηρέασε τους αισθητήρες μόνο στις διαστάσεις τους, αλλά τους επηρέασε σημαντικά και στην μείωση της ενέργειας που καταναλώνουν και στην ευφυΐα που ενσωματώνουν ως αυτόνομες συσκευές. Όλα τα παραπάνω συντέλεσαν στην τεράστια αύξηση της χρήσης των αισθητήρων σε πληθώρα εφαρμογών.

Αν προσπαθούσαμε να κατηγοριοποιήσουμε τους αισθητήρες, ώστε να έχουμε μια πιο ξεκάθαρη εικόνα για την ευρεία γκάμα που προσφέρεται θα λαμβάναμε υπόψη μας διάφορους παράγοντες. Αρχικά, ένας αισθητήρας μπορεί να κατηγοριοποιηθεί ανάλογα το φυσικό μέγεθος που έχει σχεδιαστεί να μετράει, έτσι θα έχουμε τους αισθητήρες ακτινοβολίας, τους μηχανικούς, θερμικούς, χημικούς και μαγνητικούς. Μια άλλη κατηγοριοποίηση που θα μπορούσαμε κάνουμε

στους αισθητήρες είναι αν και σε τι ποσοστό χρησιμοποιούν κάποια βοηθητική πηγή ενέργειας ή σε άλλα ιδιαίτερα χαρακτηριστικά τους που τους διαφοροποιούν από άλλους αισθητήρες όπως είναι οι οπτικοί αισθητήρες, οι βιοαισθητήρες, οι ανιχνευτές κίνησης, τα επιταχυνσιόμετρα, οι αισθητήρες θερμοκρασίας, υπερήχων, υπερύθρων και άλλα ιδιαίτερα χαρακτηριστικά που μπορεί να έχουν κατά περίπτωση που χρησιμοποιούνται.

Πλέον οι αισθητήρες χρησιμοποιούνται σε πληθώρα εφαρμογών, από πλυντήρια, ψυγεία και κουζίνες μέχρι αυτοκίνητα, αεροσκάφη και διαστημόπλοια. Ενώ τομείς όπως η ιατρική και η στρατιωτική βιομηχανία, χρησιμοποιούν την αιχμή της τεχνολογίας των αισθητήρων. Το πλέον αντιπροσωπευτικό παράδειγμα που χρησιμοποιούνται αισθητήρες είναι τα έξυπνα κινητά τηλέφωνα. Στην πλειοψηφία τους χρησιμοποιούν δεκάδες αισθητήρες το καθένα, όπως γυροσκόπιο, επιταχυνσιόμετρο, βαρόμετρο, μαγνητόμετρο, gps, φωτόμετρο, αισθητήρα υπερύθρων, θερμοκρασίας, καρδιακών παλμών, δαχτυλικών αποτυπωμάτων, πεδόμετρο, αισθητήρα αναγνώρισης ίριδας, μικρόφωνο, κάμερα και άλλους. Έχοντας υπόψη όλα τα παραπάνω και καταλαβαίνοντας την τεράστια ζήτηση στην αγορά των αισθητήρων, η βιομηχανία των αισθητήρων για να ανταποκριθεί στην κάλυψη των αναγκών και στην ανταγωνιστικότητα προσπαθεί να παράγει αισθητήρες χαμηλού κόστους, αυξημένης λειτουργικότητας και ευφυέστερους. Έχοντας οι εταιρείες στα χέρια τους τέτοιας ποιότητας αισθητήρες, μπορούν να κάνουν αλματώδεις προόδους και να μπουν σε νέες καινοτόμες τεχνολογικές προκλήσεις, προς όφελος των ιδίων αλλά και των καταναλωτών.

### **2.1.2 Ανιχνευτές Κίνησης & Παρουσίας (Occupancy sensors)**

Η ανίχνευση κίνησης και η ανίχνευση παρουσίας σε ένα χώρο είναι ανάγκη που προκύπτει τα τελευταία χρόνια από την ανάπτυξη των συστημάτων ασφαλείας και των συστημάτων αυτοματισμού κατοικίας. Η ανίχνευση ανθρώπινης παρουσίας σε ένα χώρο πρέπει να γίνεται με αξιόπιστο τρόπο για να μην υπάρχουν ψεύτικες ειδοποιήσεις και σύγχυση στην λειτουργία του συστήματος. Έτσι κρίνεται απαραίτητη η ανάπτυξη αισθητήρων παρουσίας (presence detectors), οι οποίοι θα μπορούν με απόλυτη βεβαιότητα να δίνουν σωστές εξόδους για την παρουσία ή μη ανθρώπου (ή και ζώου) σε ένα χώρο. Σε αυτό το σημείο πρέπει να διευκρινιστεί και η διαφορά ανάμεσα στον αισθητήρα παρουσίας και τον αισθητήρα κίνησης. Ο αισθητήρας κίνησης είναι

τεχνολογικά απλούστερος και ανιχνεύει έναν άνθρωπο ή ένα αντικείμενο μόνο αν αυτό κινηθεί, αν απλά υπάρχει στο χώρο χωρίς να κινείται το αγνοεί. Ο αισθητήρας παρουσίας όμως είναι πιο σύνθετος τεχνολογικά και μπορεί να ανιχνεύσει την ανθρώπινη παρουσία σε ένα χώρο χωρίς να υπάρξει κίνηση, αυτό το πετυχαίνει για παράδειγμα με τη διαφορά θερμοκρασίας που έχει το ανθρώπινο σώμα σε σχέση με τον υπόλοιπο χώρο. Όλα αυτά μπορούν να χρησιμοποιηθούν κυρίως για λόγους ασφαλείας και για λόγους εξοικονόμησης ενέργειας. Ένα προφανές παράδειγμα είναι η ειδοποίηση ενός συναγερμού αν αντιληφθεί παρουσία σε ένα χώρο που προστατεύει. Όσον αφορά την εξοικονόμηση ενέργειας σε ένα κτήριο, όταν από το σύστημα, μέσω των αισθητήρων παρουσίας, δεν ανιχνευθεί ανθρώπινη παρουσία σε ένα χώρο, μπορεί να σβήσουν τα φώτα, να κλείσει ο κλιματισμός να σβήσει μια αναμμένη κουζίνα που έχει ξεχασθεί.

Για την ανίχνευση ανθρώπινης παρουσίας και ανάλογα την εφαρμογή που θα έχει ο κάθε αισθητήρας, η ανίχνευση μπορεί να επιτευχθεί με διάφορους τρόπους καθένας από τους οποίους παίρνει ερεθίσματα από άλλες ιδιότητες του ανθρώπινου σώματος, όπως τη θερμοκρασία, το βάρος ή ήχους που παράγονται.

Ενδεικτικά κάποιοι τέτοιοι αισθητήρες είναι:

1. Μαγνητικοί διακόπτες, που αντιλαμβάνονται αν είναι ανοιχτά ή κλειστά τα κουφώματα ενός χώρου
2. Αισθητήρες πίεσης τύπου χαλιού, που ανιχνεύουν την πίεση στο πάτωμα από το περπάτημα ανθρώπου.
3. Αισθητήρες πίεσης αέρα, για την ανίχνευση στην αλλαγή της ροής του αέρα μέσα σε ένα χώρο.
4. Αισθητήρες θραύσης, οι οποίοι αντιλαμβάνονται το σπάσιμο κάποιου τζαμιού σε κούφωμα.
5. Ανιχνευτές κραδασμών για τοίχους, οι οποίοι μπορούν να αντιληφθούν δονήσεις που μπορεί να προκληθούν από το σπάσιμο κάποιου τοίχου.

### **2.1.3 Έξυπνοι Αισθητήρες**

Έξυπνος αισθητήρας (smart sensor) είναι μια ηλεκτρονική διάταξη, η οποία αποτελείται από έναν αισθητήρα και ένα κύκλωμα επεξεργασίας των δεδομένων που “διαβάζει” ο αισθητήρας και στη

συνέχεια βάση αυτών δίνει το σήμα εξόδου. Ένας έξυπνος αισθητήρας πρέπει να περιλαμβάνει τα κάτωθι χαρακτηριστικά:

1. Να έχει επικοινωνία με μία ή παραπάνω συσκευές.
2. Να είναι ικανός να παίρνει αποφάσεις χρησιμοποιώντας δεδομένα που έχει συλλέξει ο ίδιος ή του παρέχονται από άλλους.
3. Να εκτελεί λογική επεξεργασία των δεδομένων.

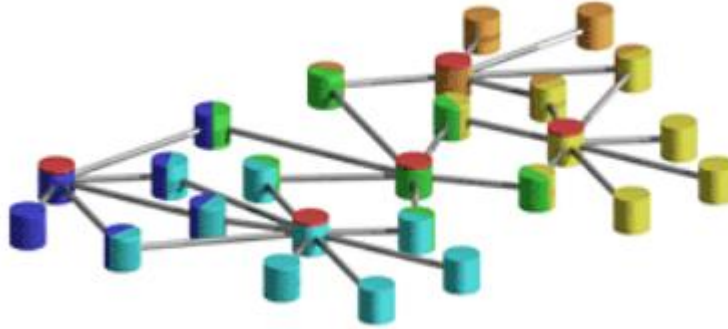
Πέρα από τις παραπάνω ιδιότητες, ένας έξυπνος αισθητήρας έχει κι άλλες ιδιαίτερες ιδιότητες που τον κάνουν να διαφέρει από τους υπόλοιπους ολοκληρωμένους αισθητήρες (integrated sensors). Οι ιδιότητες αυτές σύμφωνα με τους Hauptmann & Giachino:

- Πολυαίσθηση: ποιοτικότερα αποτελέσματα από έναν αισθητήρα χάρις την χρήση άλλων αισθητήρων, οι οποίοι παρακολουθούν τις ανεπιθύμητες εξαρτημένες μεταβλητές.
- Επικοινωνία: αυτόνομη επικοινωνία του αισθητήρα με το περιβάλλον του.
- Δομική αντιστάθμιση: Προβλέπονται τα κατάλληλα δομικά χαρακτηριστικά σε συνάρτηση με την προβλεπόμενη χρήση του.
- Αυτοβαθμονόμηση: αυτόματη προσαρμογή της ευαισθησίας και της αντιστάθμισης του αισθητήρα σύμφωνα με την προβλεπόμενη χρήση του.
- Αυτοέλεγχος: Η ικανότητα που έχει ο αισθητήρας να ελέγχει τη λειτουργικότητά του.
- Ολοκληρωμένη επεξεργασία σήματος: Η πλήρης επεξεργασία του συλλεγόμενου σήματος εντός του αισθητήρα.

Όσο η αγορά αναπτύσσεται, τόσο οι απαιτήσεις για πιο ευφυείς αισθητήρες αυξάνονται και οι εφαρμογές που χρησιμοποιούν έξυπνους αισθητήρες καθημερινά πληθαίνουν. Οι βασικοί τομείς που καινοτομούν με τη χρήση αισθητήρων είναι η αυτοκινητοβιομηχανία, τα έξυπνα κτήρια, έλεγχος της ποιότητας παραγωγής και η ανάπτυξη διεπαφών για άτομα με ειδικές ανάγκες.



## 2.2 Δίκτυα Ασύρματων Αισθητήρων



Εικόνα 1 - Ασύρματα Δίκτυα Αισθητήρων

Η ραγδαία εξέλιξη της τεχνολογίας και η αυξανόμενη χρήση ασύρματων συσκευών κατά τις τελευταίες δεκαετίες έχει οδηγήσει στην αναθεώρηση του τρόπου αντιμετώπισης των ασύρματων δικτύων. Σε αντίθεση με τα παραδοσιακά ενσύρματα δίκτυα, τα ασύρματα δίκτυα χαρακτηρίζονται από την έλλειψη κεντρικής διαχείρισης, τη δυσκολία προσέγγισης και φυσικής παρουσίας στο δίκτυο, την αναξιопιστία του ασύρματου διαύλου, την κινητικότητα και τους περιορισμούς των πόρων των κόμβων. Επομένως, στα ασύρματα δίκτυα εμφανίζεται επιτακτική η ανάγκη για εξασφάλιση αυτόνομης λειτουργίας. Για να χαρακτηριστεί ένα σύστημα αυτόνομο πρέπει να διαθέτει τα παρακάτω χαρακτηριστικά:

- i. Το σύστημα πρέπει να ελέγχει τις εσωτερικές και εξωτερικές του λειτουργίες, χωρίς εξωτερική παρέμβαση.
- ii. Το σύστημα πρέπει να είναι σε θέση να προσαρμόζεται στις μεταβολές του περιβάλλοντος στο οποίο βρίσκεται, κάνοντας τις απαραίτητες τροποποιήσεις για τη βελτιστοποίηση της λειτουργίας του.
- iii. Το σύστημα πρέπει να είναι διαρκώς ενήμερο ώστε να είναι σε θέση να αντιλαμβάνεται έγκαιρα τις μεταβολές και να αντιδρά καταλλήλως.

Τα ασύρματα δίκτυα που διαθέτουν αυτόνομη λειτουργία συναντώνται στη βιβλιογραφία ως ασύρματα αυτό-οργανούμενα δίκτυα (Wireless Self-Organizing Networks, WSON). Στην ευρύτερη κατηγορία των WSON ανήκουν τα ασύρματα ad hoc δίκτυα, δηλαδή δίκτυα τα οποία

σχηματίζονται οπουδήποτε και οποτεδήποτε, δίχως την ανάγκη ύπαρξης κάποιας υποδομής. Οι τρεις μεγάλες κατηγορίες WSON δικτύων είναι οι ακόλουθες:

### **Ασύρματα δίκτυα αισθητήρων**

Τα ασύρματα δίκτυα αισθητήρων (Wireless Sensor Networks, WSN) αποτελούνται από ένα πλήθος ασύρματων αισθητήρων που είναι επιφορτισμένοι με την παρακολούθηση ενός ή περισσότερων φυσικών φαινομένων πραγματοποιώντας μετρήσεις των κατάλληλων φυσικών μεγεθών. Τα δίκτυα αυτά είναι ομοιογενή, αφού αποτελούνται από κόμβους με πανομοιότυπες λειτουργίες και δυνατότητες. Οι κόμβοι διαθέτουν περιορισμένους ενεργειακούς και υπολογιστικούς πόρους. Τα δίκτυα αυτά μπορεί να είναι μεγάλης κλίμακας με εκατοντάδες ή χιλιάδες κόμβους. Η παρούσα διπλωματική εργασία θα ασχοληθεί με αυτήν την κατηγορία δικτύων. Για αυτό θα γίνει εκτενέστερη αναφορά σε αυτά σε επόμενη ενότητα.

### **Ασύρματα χαοτικά δίκτυα**

Τα ασύρματα χαοτικά δίκτυα (Wireless Mesh Networks, WMN) χρησιμοποιούνται για την παροχή υπηρεσιών Διαδικτύου σε δυσμενές περιβάλλον. Ταξινομούνται σε τρεις βασικές κατηγορίες: τα δίκτυα υποδομής WMN (infrastructure WMN), τα δίκτυα πελατών WMN (client WMN) και τα υβριδικά δίκτυα WMN (hybrid WMN). Οι κόμβοι των δικτύων υποδομής WMN σχηματίζουν ένα μικρό δίκτυο παροχής υπηρεσιών και πρέπει να ανταποκρίνονται στην υψηλή ζήτηση για εξυπηρέτηση των WMN δικτύων πελατών. Τα WMN δίκτυα πελατών πρέπει να εγγυώνται την εξασφαλισμένη σύνδεση των κόμβων-πελατών τους με το δίκτυο υποδομής, παρά την ετερογένεια στους τρόπους σύνδεσης. Τα υβριδικά δίκτυα WMN αποτελούνται και από κόμβους-υποδομής και από κόμβους-πελάτες.

## **Δίκτυα με ανοχή σε καθυστέρηση/αποσύνδεση**

Τα δίκτυα με ανοχή σε καθυστέρηση/αποσύνδεση (Delay/Disruption Tolerant Networks, DTN) αποτελούν μια κατηγορία ασύρματων δικτύων που αναπτύσσονται σε δυναμικό περιβάλλον με αντίξοες συνθήκες. Χαρακτηρίζονται από συχνές διακοπές επικοινωνίας μεταξύ των κόμβων, ενώ δεν είναι πάντοτε εφικτή η σύνδεση του κόμβου-πηγής με τον κόμβο - προορισμό. Στα δίκτυα αυτά, οι κόμβοι διαθέτουν περιορισμένους ενεργειακούς, επεξεργαστικούς και αποθηκευτικούς πόρους.

### **2.2.1. Ασύρματα δίκτυα αισθητήρων (WSN)**

Ένα δίκτυο ασύρματο αισθητήρων (WSN) είναι μια υποδομή, η οποία αποτελείται από στοιχεία αίσθησης (μέτρησης), υπολογισμού και επικοινωνίας και παρέχει τη δυνατότητα στο διαχειριστή του δικτύου να παρατηρεί γεγονότα και φαινόμενα που συμβαίνουν σε ένα συγκεκριμένο περιβάλλον και να αντιδρά καταλλήλως. Ο επιτηρητής του δικτύου είναι συνήθως κάποιος κοινωνικός, κυβερνητικός, εμπορικός ή βιομηχανικός οργανισμός. Τα σημερινά ασύρματα δίκτυα αισθητήρων αποτελούνται από μεγάλο αριθμό φθηνών συσκευών που διασυνδέονται μεταξύ τους με χρήση ασύρματης επικοινωνίας χαμηλής ισχύος. Στόχος ενός WSN δικτύου είναι η παρακολούθηση ενός ή περισσότερων φυσικών φαινομένων που λαμβάνουν χώρα εντός μιας περιοχής ενδιαφέροντος. Τα ασύρματα δίκτυα αισθητήρων αναπτύσσονται σε περιοχές στις οποίες η ανθρώπινη παρέμβαση είναι δύσκολη ή έχει μεγάλο κόστος, παρόλα αυτά η λειτουργία τους θα πρέπει να είναι αυτόνομη. Σύμφωνα με την IBM, για να είναι ένα σύστημα αυτόνομο πρέπει να διαθέτει τα παρακάτω χαρακτηριστικά:

1. Αυτο-προστασία (self-protecting)
2. Αυτο-βελτιστοποίηση (self-optimizing)
3. Αυτο-ίαση (self-healing)
4. Αυτο-ρύθμιση (self-configurable)

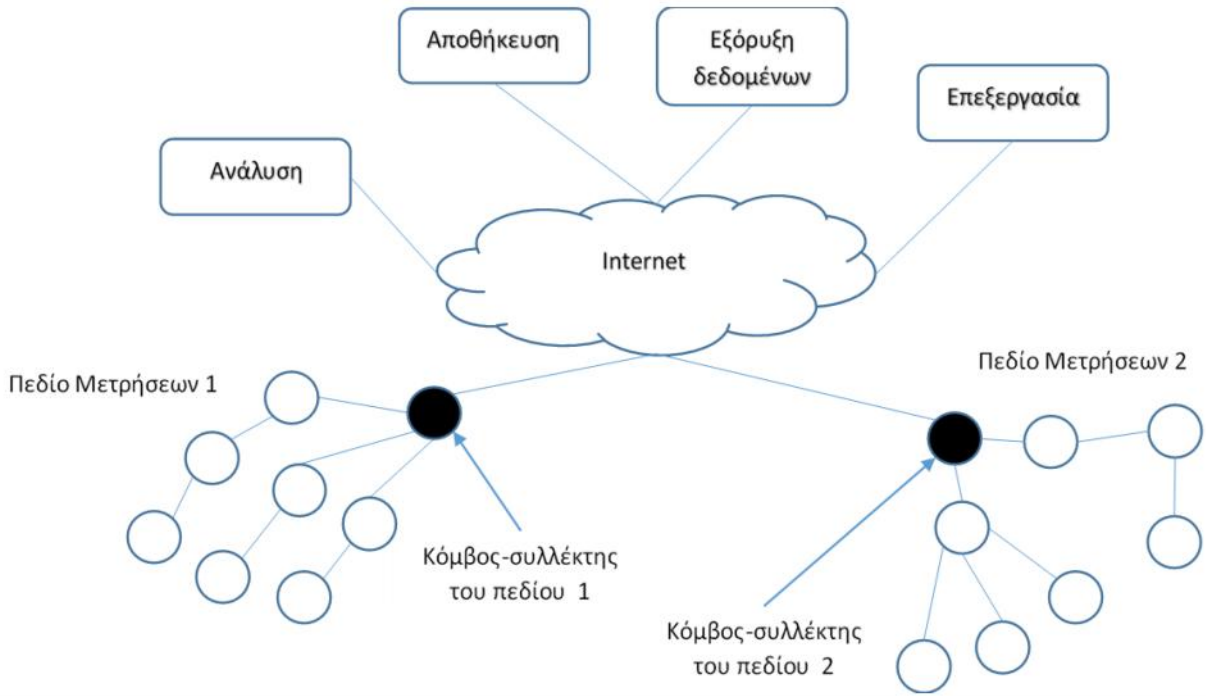
Η ιδιότητα της αυτο-ρύθμισης εξασφαλίζει ότι ένα WSN ρυθμίζει δίχως εξωτερική παρέμβαση τις παραμέτρους που απαιτούνται για την ομαλή λειτουργία του. Επιπλέον, η ιδιότητα της αυτο-

βελτιστοποίησης δίνει στο ασύρματο δίκτυο αισθητήρων την δυνατότητα προσαρμογής των παραμέτρων λειτουργίας ώστε η συνολική λειτουργία του δικτύου να παραμένει βέλτιστη.

Η ιδιότητα της αυτο-προστασίας συνεπάγεται ότι ένα WSN έχει την ικανότητα πρόληψης και αντιμετώπισης των μεταβολές, οι οποίες ενδεχομένως απειλούν την ομαλή λειτουργία του. Τέλος, η ιδιότητα της αυτο-ίασης επιτρέπει σε ένα WSN να διαγνώσει πιθανή διαταραχή της λειτουργίας του, την αιτία που την προκάλεσε αλλά και τρόπους επαναφοράς της βέλτιστης λειτουργίας του. Η δομή ενός WSN είναι σχετικά απλή. Ένα WSN απαρτίζεται από ένα σύνολο ασύρματων κόμβων-αισθητήρων (Wireless Sensor, WS) έκαστος των οποίων σε κατάλληλες χρονικές στιγμές διεξάγει μετρήσεις των φυσικών μεγεθών που ενδιαφέρουν. Οι πληροφορίες οι οποίες συγκεντρώνονται από τους κόμβους ενός δικτύου ασύρματων αισθητήρων αποθηκεύονται συνήθως στον κόμβο-συλλέκτη (sink node) του WSN. Συνήθως, το node αυτό έχει μεγαλύτερη υπολογιστική ισχύ και ενεργειακούς πόρους από τους υπόλοιπους κόμβους του δικτύου. Χρησιμοποιώντας τεχνικές ασύρματης επικοινωνίας οι κόμβοι αισθητήρες μπορούν να επικοινωνούν μεταξύ τους.

Όμως, λόγω των πεπερασμένων ενεργειακών πόρων που διαθέτουν οι κόμβοι ενός WSN, είναι ανέφικτη η απευθείας επικοινωνία όλων των κόμβων του δικτύου με τον κόμβο-συλλέκτη. Για το λόγο αυτό, υιοθετείται συνήθως μια λειτουργία προώθησης των μηνυμάτων πληροφορίας βήμα προς βήμα από τον αρχικό κόμβο-αποστολέα μέχρι τον κόμβο - συλλέκτη. Επομένως, κάθε ασύρματος κόμβος, εκτός από μέτρηση κάποιου ή κάποιων μεγεθών, πρέπει να είναι σε θέση να επεξεργάζεται δεδομένα, να αποθηκεύει δεδομένα και να επικοινωνεί με άλλους κόμβους.

Δηλαδή, οι κόμβοι ενός WSN αναλαμβάνουν ένα διπλό ρόλο: αφενός, λειτουργούν ως πηγές πληροφορίας που προέρχεται από τον αισθητήρα τους και, αφετέρου, λειτουργούν ως ενδιάμεσοι κόμβοι-αναμεταδότες μηνυμάτων άλλων κόμβων του δικτύου. Τέλος, μέσω του κόμβου-συλλέκτη επιτυγχάνεται και σύνδεση, συνήθως μέσω του Διαδικτύου, του ασύρματου δικτύου με το διαχειριστή του συστήματος, ο οποίος αξιοποιεί τη λαμβανόμενη πληροφορία.



Εικόνα 2 - Αρχιτεκτονική WSN

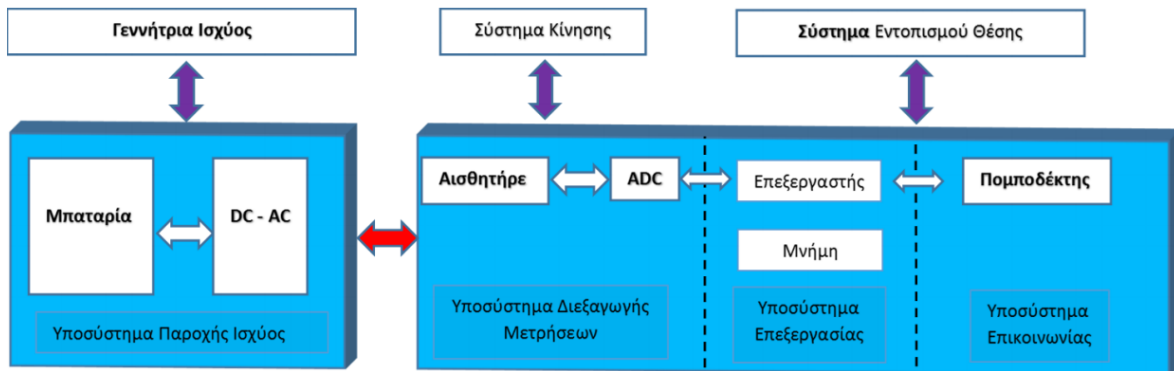
### 2.2.2. Περιγραφή του ασύρματου κόμβου αισθητήρα

Η εξέλιξη της τεχνολογίας της μικροηλεκτρονικής που έχει συντελεστεί κατά τις τελευταίες δεκαετίες επιτρέπει τη σχεδίαση και παραγωγή ολοκληρωμένων κυκλωμάτων και συσκευών με σχετικά μεγάλες υπολογιστικές δυνατότητες, μικρό όγκο και χαμηλό κόστος παραγωγής. Η κατηγορία των συστημάτων αυτών καλείται MEMS ( Micro-Electro-Mechanical Systems). Μια από τις πλέον αντιπροσωπευτικές χρήσεις των συστημάτων MEMS αφορά τη δημιουργία ασύρματων κόμβων αισθητήρων (Wireless Sensors, WS).

Στη συνέχεια, παρουσιάζονται τα δομικά στοιχεία ενός ασύρματου κόμβου. Προκειμένου κάθε κόμβος-αισθητήρας να πραγματοποιήσει τις απαραίτητες λειτουργίες της μέτρησης φυσικών μεγεθών, της επεξεργασίας δεδομένων και της επικοινωνίας με άλλους κόμβους χρειάζεται μια πηγή ενέργειας. Αυτή η πηγή είναι συνήθως μια μπαταρία, για την οποία συνήθως δεν υπάρχει η δυνατότητα αντικατάστασης. Αυτό έχει ως αποτέλεσμα οι κόμβοι-αισθητήρες να έχουν περιορισμένα ενεργειακά αποθέματα.

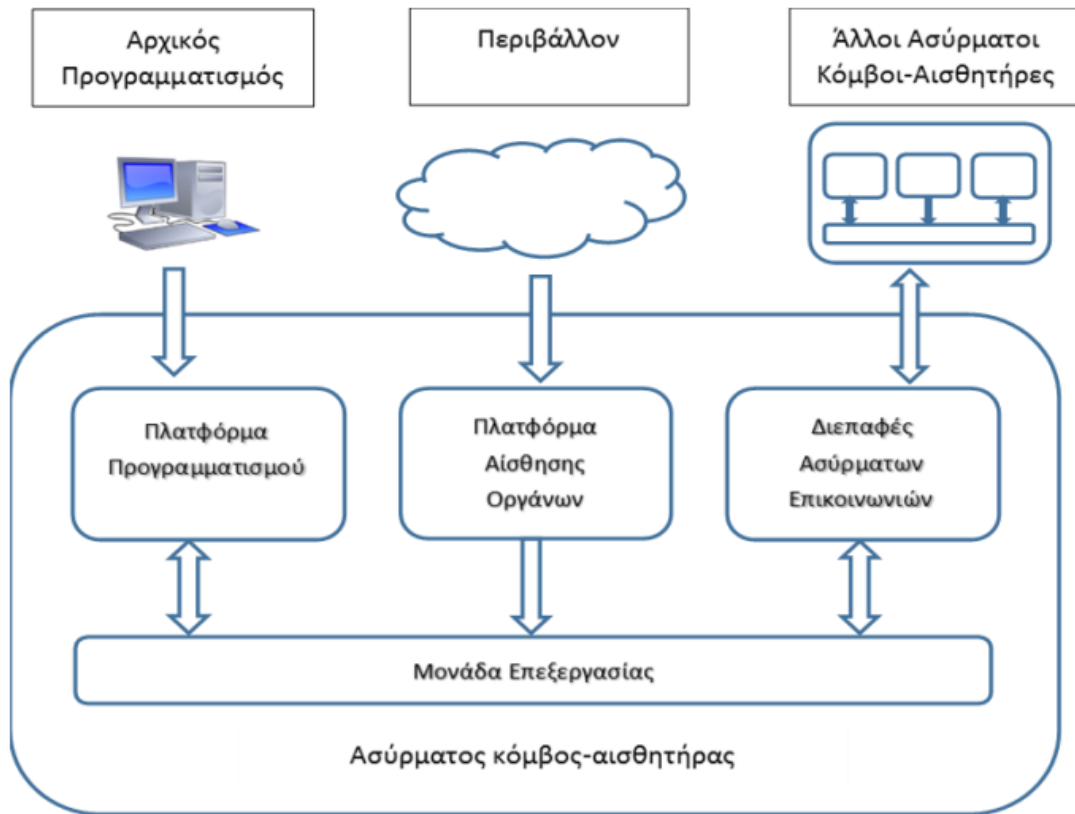
Σπανιότερα, ο κόμβος-αισθητήρας διαθέτει ηλιακό πάνελ, μέσω του οποίου επιτυγχάνεται μετατροπή της ηλιακής ενέργειας σε ηλεκτρική και στη συνέχεια γίνεται επαναφόρτιση της μπαταρίας. Ακόμα, όμως, και σε μια τέτοια περίπτωση, η παράταση της διάρκειας ζωής του κόμβου που επιτυγχάνεται δεν αλλάζει το χαρακτηριστικό της έλλειψης ενεργειακών πόρων των κόμβων-αισθητήρων. Επίσης, καθώς οι κόμβοι συνεργάζονται μεταξύ τους για τη μεταφορά πληροφορίας προς τον κόμβο-συλλέκτη, κάθε κόμβος-συλλέκτης διαθέτει μνήμη για την προσωρινή αποθήκευση δεδομένων, επεξεργαστή για την κατάλληλη επεξεργασία των δεδομένων και επιπλέον, έναν πομποδέκτη περιορισμένης εμβέλειας και κεραία.

Αντίστοιχα με το είδος της εφαρμογής που εξυπηρετεί το ασύρματο δίκτυο αισθητήρων, οι κόμβοι μπορεί να διαθέτουν πρόσθετες μονάδες. Συγκεκριμένα, είναι πιθανό να διαθέτουν ένα σύστημα εντοπισμού θέσης (location finding system). Επίσης, μπορεί να διαθέτουν σύστημα κίνησης, ώστε να μπορούν να μετακινηθούν σε μικρές αποστάσεις καθώς και να μεταβάλλουν τον προσανατολισμό τους. Οι μονάδες αυτές, πάντως, είναι προαιρετικές και δεν αποτελούν τη γενική περίπτωση.



Εικόνα 3 - Δομικά στοιχεία κόμβου-αισθητήρα

Η αρχιτεκτονική ενός ασύρματου κόμβου αποτελείται από τέσσερις δομικές μονάδες. Αυτά είναι (1) η μονάδα προγραμματισμού (2) η μονάδα αισθητήρων (3) η μονάδα διεπαφών ασύρματης επικοινωνίας και (4) η μονάδα επεξεργασίας.



Εικόνα 4 - Αρχιτεκτονική κόμβου-αισθητήρα

Μέσω της πλατφόρμας προγραμματισμού, πραγματοποιείται ο αρχικός προγραμματισμός του ασύρματου κόμβου, αλλά υπάρχει και δυνατότητα αναβάθμισης του λογισμικού του. Η πλατφόρμα αισθητήρων περιλαμβάνει όλα τα απαραίτητα μετρητικά όργανα που ανιχνεύουν το ή τα υπό παρακολούθηση φυσικά φαινόμενα. Τα όργανα αυτά, μπορεί να μετρούν φυσικά μεγέθη όπως φωτεινότητα, ένταση υπεριώδους ακτινοβολίας, υγρασία, ένταση σεισμικών δονήσεων, θερμοκρασία, ηλεκτρομαγνητικά πεδία, και άλλα.

Οι διεπαφές ασύρματης επικοινωνίας αποτελούνται συνήθως από πομποδέκτη και ενισχυτή εκπομπής. Μέσω αυτών των διεπαφών, καθίσταται εφικτή η επικοινωνία του κόμβου με το υπόλοιπο δίκτυο αισθητήρων. Τέλος, η μονάδα επεξεργασίας, είναι άρρηκτα συνδεδεμένη με τη λειτουργία του κόμβου, καθώς επεξεργάζεται τις μετρήσεις που λαμβάνονται από άλλους κόμβους και τις μετρήσεις που λαμβάνονται από τον ίδιο τον κόμβο, σύμφωνα με τις εντολές της μονάδας προγραμματισμού. Επίσης, είναι η μονάδα που φροντίζει για την εφαρμογή του πρωτοκόλλου επικοινωνίας εκτελώντας υπολογισμούς και λαμβάνοντας αντίστοιχες αποφάσεις.

Στη γενική περίπτωση, για τη λειτουργία ενός ασύρματου δικτύου αισθητήρων απαιτείται μεγάλος κατά κανόνα αριθμός ασύρματων κόμβων. Αυτό έχει ως συνέπεια την ανάγκη ελαχιστοποίησης του κόστους παραγωγής των κόμβων. Με βάση το σκοπό χρήσης των ασύρματων κόμβων, τα τεχνικά χαρακτηριστικά τους, παρουσιάζουν αρκετές διαφοροποιήσεις που οδηγούν στην ένταξή τους σε δύο κατηγορίες με κριτήριο τις δυνατότητές τους.

Η πρώτη κατηγορία αποτελείται από κόμβους χαμηλών υπολογιστικών δυνατοτήτων. Οι κόμβοι αυτής της κατηγορίας διαθέτουν μικροεπεξεργαστή σχετικά χαμηλής συχνότητας λειτουργίας (4MHz – 16MHz). Διαθέτουν, επίσης, μνήμη τύπου flash χωρητικότητας 8kB έως 512 kB και μνήμη τυχαίας προσπέλασης χωρητικότητας 0.5kB έως 64kB. Πιθανές τιμές της φέρουσας συχνότητας που χρησιμοποιείται για την επικοινωνία μεταξύ των κόμβων είναι τα 868MHz για την Ευρώπη και τα 915MHz στην Αμερική. Μια ακόμη πιθανή τιμή της φέρουσας συχνότητας είναι τα 2.4GHz, η οποία είναι επιτρεπτή σε παγκόσμιο επίπεδο. Οι ρυθμοί μετάδοσης που επιτυγχάνουν οι κόμβοι κυμαίνονται μεταξύ 10kbps και 250kbps. Εμπορικοί κόμβοι που ανήκουν σε αυτήν την κατηγορία είναι οι Telos/TMode, IRIS, EYES και Mica/Mica2/MicaZ.

Η δεύτερη κατηγορία αποτελείται από κόμβους υψηλότερων υπολογιστικών δυνατοτήτων. Οι κόμβοι αυτής της κατηγορίας διαθέτουν επεξεργαστές με συχνότητες που μπορούν να φθάσουν ακόμα και τα 400MHz. Αντίστοιχα, η μνήμη αποθήκευσης των κόμβων αυτών μπορεί να φθάσει μέχρι και τα 32MB και η μνήμη τυχαίας προσπέλασης τα 64MB. Οι φέρουσες συχνότητες και ο ρυθμός μετάδοσης δεδομένων κινούνται στο ίδιο εύρος τιμών, σε σχέση με τους κόμβους της προηγούμενης κατηγορίας. Εμπορικοί κόμβοι υψηλών υπολογιστικών δυνατοτήτων είναι οι IMote/IMote2, Stargate και Netbridge NB-100. Οι ασύρματοι κόμβοι τροφοδοτούνται από μπαταρίες περιορισμένης χωρητικότητας. Ο περιορισμός αυτός διαφοροποιεί τα δίκτυα αισθητήρων από τα υπόλοιπα δίκτυα και πρέπει να λαμβάνεται πολύ σοβαρά υπόψη κατά το σχεδιασμό πρωτοκόλλων λειτουργίας.



### 2.2.3. Διαφοροποίηση των WSN σε σχέση με τα υπόλοιπα δίκτυα

Τα Ασύρματων Δικτύων Αισθητήρων διαφέρουν από τα κλασικά δίκτυα λόγω κάποιων ιδιαίτερων χαρακτηριστικών που έχουν, αυτά είναι:

- a) Στα Ασύρματων Δικτύων Αισθητήρων υπάρχουν σημαντικοί περιορισμοί όσον αφορά την παροχή ενέργειας, τον αποθηκευτικό χώρο για τα δεδομένα, την υπολογιστική ισχύ του επεξεργαστή τους και το εύρος ζώνης της εκπομπής τους. Αντιθέτως, στα κλασικά ασύρματα δίκτυα η διαχείριση της κινητικότητας και η λειτουργία της δρομολόγησης πακέτων γίνονται με σκοπό το βέλτιστο δυνατό QoS. Ακόμα, λόγω της ανθρώπινης παρουσίας και της εγκατάστασής τους σε χώρους εύκολα προσβάσιμους, η κατανάλωση ενέργειας δεν μας απασχολεί και τόσο, διότι η πηγή ενέργειας μπορεί με ευκολία να αντικατασταθεί ή να επαναφορτιστεί ανά πάσα στιγμή. Από την άλλη, οι κόμβοι των Ασύρματων Δικτύων Αισθητήρων είναι σχεδιασμένοι για να λειτουργούν σε περιβάλλοντα χωρίς ανθρώπινη παρουσία, οπότε όλες οι λειτουργίες εκτελούνται με γνώμονα την χαμηλή κατανάλωση ενέργειας και την εξοικονόμηση πόρων για την όσο το δυνατόν μεγαλύτερη διάρκεια λειτουργίας χωρίς ανθρώπινη παρέμβαση.
- b) Ένα άλλο χαρακτηριστικό που διαφοροποιεί τα Ασύρματων Δικτύων Αισθητήρων από τα κλασικά δίκτυα είναι το μοντέλο επικοινωνίας που χρησιμοποιούν. Στα κλασικά δίκτυα ο χρήστης συνδέεται σε έναν κόμβο και ζητάει κάποια υπηρεσία από άλλο κόμβο και έτσι διαμορφώνεται κατά κύριο λόγο η επικοινωνιακή κίνηση σε ένα τέτοιου είδους δίκτυο. Στα Ασύρματων Δικτύων Αισθητήρων όμως η επικοινωνία των κόμβων μοιάζει περισσότερο με των κατανεμημένων συστημάτων. Ο χρήστης σε αυτή την περίπτωση δεν ενδιαφέρεται για τα αποτελέσματα κάποιου συγκεκριμένου κόμβου, αλλά για αποτελέσματα που έχουν παραχθεί από την συνεργασία των κόμβων και κατόπιν επεξεργασίας των αρχικών τιμών που έχουν συλλεχθεί. Αυτό έχει ως συνέπεια, σε ένα τέτοιο δίκτυο να μην υπάρχουν συγκεκριμένοι κόμβοι που μας δίνουν τα δεδομένα, ούτε ο χρήστης να μπορεί να επιλέξει από ποιόν κόμβο θα αντλεί τα δεδομένα του, σε ένα τέτοιο δίκτυο υπάρχει μόνο ο χρήστης και το σύνολο του δικτύου. Οπότε σε αυτή την περίπτωση, το δίκτυο δεν παρέχει ουσιαστικά σύνδεση μεταξύ τμημάτων του δικτύου, αλλά υπηρεσίες προς τον τελικό χρήστη.

- c) Στα Ασύρματων Δικτύων Αισθητήρων οι κόμβοι παραμένουν στη θέση τους μετά την τοποθέτησή τους, ενώ σπάνια υπάρχουν κινούμενοι κόμβοι.
- d) Τέλος, στα Ασύρματων Δικτύων Αισθητήρων δημιουργείται το φαινόμενο του πλεονασμού, αυτό συμβαίνει διότι οι κόμβοι των δικτύων αυτών στέλνουν τα δεδομένα τους με χαμηλό ρυθμό μετάδοσης.

	WSN	Ασύρματα Ad hoc δίκτυα
Αριθμός κόμβων	Μεγάλος: Εκατοντάδες έως χιλιάδες κόμβοι, ή και περισσότεροι	Μικρός μέχρι μέσος
Πυκνότητα κόμβων	Υψηλή	Σχετικά χαμηλή
Πλεονασμός δεδομένων	Υψηλός	Περιορισμένος
Τροφοδότηση ισχύος	Μη επαναφορτιζόμενη λειτουργία, αναντικατάστατες μπαταρίες	Επαναφορτιζόμενη λειτουργία και/ή αντικατάσταση μπαταριών
Ρυθμός δεδομένων	Χαμηλός: 1-100 kbps	Υψηλός
Κινητικότητα των κόμβων	Χαμηλή	Πιθανότατα υψηλή κινητικότητα
Κατεύθυνση της ροής δεδομένων	Κυρίως μονοκατευθυντήρια ροή: ασύρματοι κόμβοι → sink	Δικατευθυντήρια: από άκρο σε άκρο
Προώθηση πακέτων	Πολλοί κόμβοι σε έναν: κατεύθυνση προσανατολισμένη στα δεδομένα (data centric)	Από άκρο σε άκρο: κατεύθυνση προσανατολισμένη στη διεύθυνση (address centric)
Φύση αίτησης	Βασισμένη στην κατάσταση (attribute based)	Βασισμένη στον κόμβο
Μετάδοση αιτήσεων	Πολύ-εκπομπή (broadcast)	Πολύ-εκπομπή ή hop by hop
Διευθυνσιοδότηση	Έλλειψη γενικού identification	Χρήση γενικού identification (Global ID)
Ενεργό Duty cycle	Χαμηλό, έως και 1%	Υψηλό

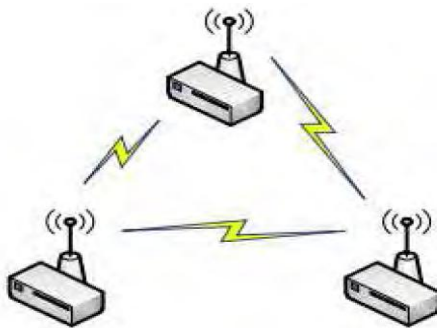
## 2.2.4. Τοπολογίες WSN

Με τον όρο τοπολογία δικτύου, εννοούμε τον τρόπο της φυσικής διασύνδεσης των κόμβων, δηλαδή, την αρχιτεκτονική διασύνδεση. Η ανάπτυξη και εξάπλωση των WSN έχει οδηγήσει τις παραδοσιακές τοπολογίες δικτύων σε νέες κατευθύνσεις. Πολλές από τις σύγχρονες εφαρμογές αισθητήρων, απαιτούν εναλλακτικές λύσεις δικτύωσης που μειώνουν το κόστος και την πολυπλοκότητα, ενώ παράλληλα βελτιώνουν την αξιοπιστία του συστήματος. Οι πιο σημαντικές τοπολογίες που χρησιμοποιούνται στα WSN είναι:

- Τοπολογία Σημείο προς Σημείο (Point-to-Point)
- Τοπολογία Δέντρου (Tree)
- Τοπολογία Αστέρα (Star)
- Τοπολογία Πλέγματος (Mesh)

### Τοπολογία Σημείο προς Σημείο (Point-to-Point)

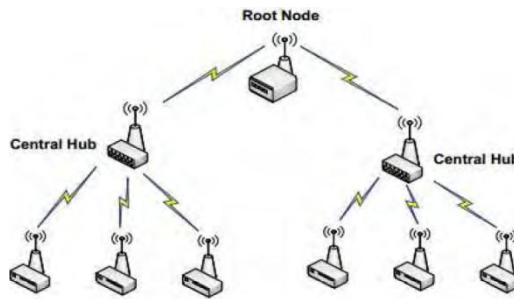
Η τοπολογία Σημείο-προς-Σημείο, επιτρέπει σε κάθε κόμβο να επικοινωνεί απευθείας με κάποιον άλλο κόμβο χωρίς τη μεσολάβηση κάποιου κεντρικού επικοινωνιακού σημείου. Κάθε κόμβος-σημείο μπορεί να λειτουργεί τόσο σαν πελάτης (client) όσο και σαν διακομιστής (server) για τους υπόλοιπους κόμβους του δικτύου.



Εικόνα 5 - Τοπολογία Σημείο προς Σημείο

### Τοπολογία Δέντρου (Tree)

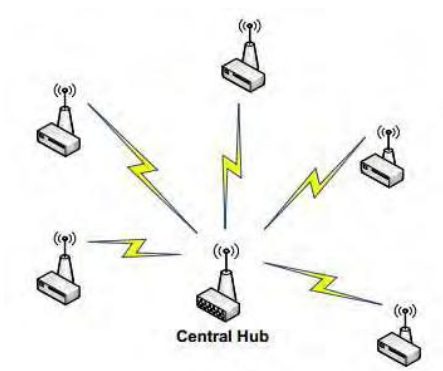
Η τοπολογία Δέντρου, χρησιμοποιεί ένα κεντρικό σημείο με την ονομασία Κόμβος-Ρίζα ως κύριο δρομολογητή της επικοινωνίας. Ένα επίπεδο κάτω από τον Κόμβο-Ρίζα στην ιεραρχία βρίσκεται ο Κεντρικός Κόμβος και ακόμα πιο χαμηλά οι υπόλοιποι κόμβοι του δικτύου. Κάθε Κεντρικός Κόμβος σχηματίζει δίκτυο τοπολογίας Αστέρα με κάποιους από τους υπόλοιπους κόμβους. Η τοπολογία Δέντρου μπορεί να θεωρηθεί σαν υβριδική τοπολογία των τοπολογιών Σημείο-προς-Σημείο και Αστέρα.



Εικόνα 6 - Τοπολογία Δέντρου

### Τοπολογία Αστέρα (Star)

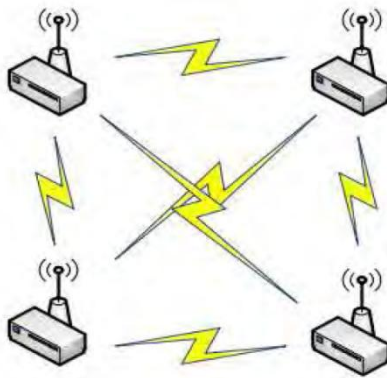
Στην τοπολογία αστέρα υπάρχει μια κεντρική συσκευή (υπολογιστής ή δρομολογητής) η οποία καλείται και κεντρικός κόμβος η οποία είναι συνδεμένη με όλες τις συσκευές του δικτύου (Σταθμούς εργασίας (Workstations), Διακομιστές (Servers) , εκτυπωτές κλπ) και ελέγχει την επικοινωνία αυτών.



Εικόνα 7 - Τοπολογία Αστέρα

### Τοπολογία Πλέγματος (Mesh)

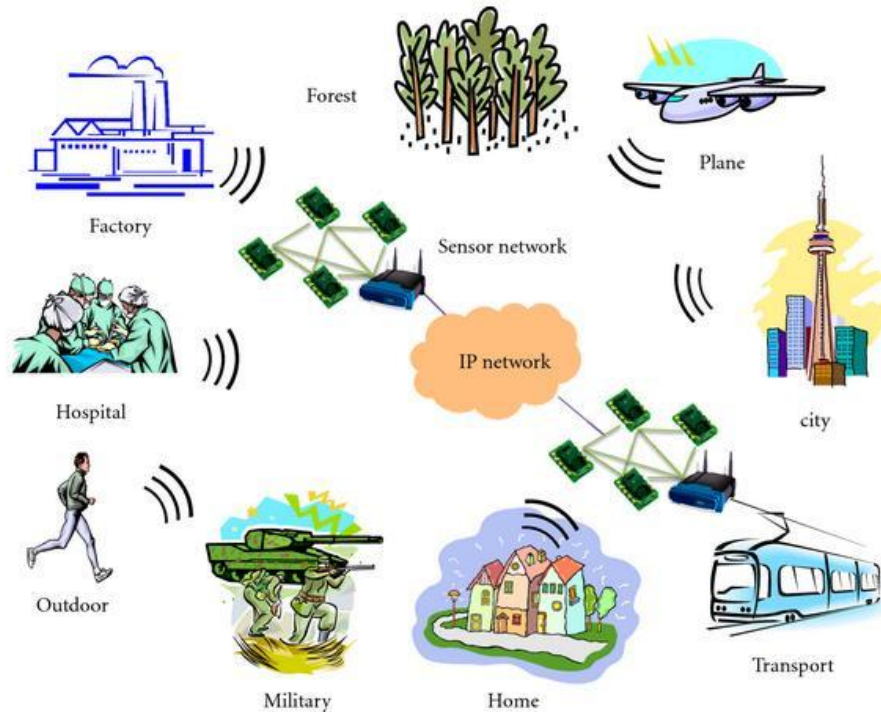
Η τοπολογία πλέγματος, επιτρέπει στα δεδομένα να «πηδούν» από κόμβο σε κόμβο, με αποτέλεσμα την αυτο-επούλωση της λειτουργίας του δικτύου σε περιπτώσεις σφαλμάτων. Κάθε κόμβος μπορεί να επικοινωνήσει με οποιονδήποτε άλλο, αφού τα δεδομένα δρομολογούνται από κόμβο σε κόμβο μέχρι να καταλήξουν στην επιθυμητή τοποθεσία του δικτύου. Αυτό το είδος τοπολογίας είναι το πιο σύνθετο και υλοποιείται σε διαφορετικές εκδοχές από τις εταιρίες που ασχολούνται με την ανάπτυξη WSN.



Εικόνα 8 - Τοπολογία Πλέγματος

Οι βασικές τοπολογίες που παρουσιάστηκαν εδώ δεν είναι οι μόνες που χρησιμοποιούνται στα σύγχρονα WSN. Ανάλογα με τις απαιτήσεις κάθε εφαρμογής ενδέχεται να υλοποιηθούν παραλλαγές ή συνδυασμοί τους, για την επίτευξη του καλύτερου αποτελέσματος.

## 2.3. Εφαρμογές Δικτύων Αισθητήρων



Εικόνα 9 - Εφαρμογές δικτύων Αισθητήρων

### 2.3.1. Υφιστάμενες εφαρμογές

Τα WSN συνήθως αποτελούνται από διάφορους τύπους αισθητήρων, κάποιοι από τους οποίους είναι θερμικοί, υπέρυθρων, μαγνητικοί, σεισμικοί, οπτικοί. Οι αισθητήρες αυτοί έχουν την ικανότητα να παρατηρούν μια μεγάλη γκάμα περιβαλλοντικών φαινομένων και μεταβολών, μερικά εκ των οποίων είναι τα παρακάτω:

- Υγρασία
- Πίεση
- Θερμοκρασία
- Ένταση φωτός
- Κίνηση οχημάτων
- Ταχύτητα
- Θόρυβος
- Κατεύθυνση

Οι αισθητήρες των δικτύων μπορούν να πραγματοποιούν συνεχή ανίχνευση, ανίχνευση μια θέσης, ανίχνευση ενός συμβάντος, κινήσεις αντικειμένων και πολλά άλλα. Οι τομείς που χρησιμοποιούνται οι αισθητήρες είναι πολλοί, αν τους ομαδοποιούσαμε θα ήταν εφαρμογές για οικίες, για το περιβάλλον, για εμπορική χρήση, για στρατιωτική χρήση, για το χώρο της υγείας, για την αντιμετώπιση καταστροφών και πολλά άλλα.

### **2.3.2. Στρατιωτικές Εφαρμογές**

Ο στρατός, ειδικά στις χώρες του εξωτερικού, χρησιμοποιεί πάντοτε την τελευταία λέξη της τεχνολογίας, ώστε να έχει εκτός των άλλων και την τεχνολογική υπεροχή σε σχέση με τους αντιπάλους. Τα δίκτυα ασύρματων αισθητήρων βοηθούν τον στρατό να αντιλαμβάνεται πράγματα εξ αποστάσεως χωρίς την παρουσία φαντάρων. Ακόμα τα δίκτυα αισθητήρων είναι εγκατεστημένα σε οπτικά συστήματα και βοηθούν να ελέγχουν, να παρακολουθούν, να αναγνωρίζουν και να στοχεύουν εχθρικά σημεία στο περιβάλλον τους, προσθέτοντας τους ευφυΐα και αυτονομία. Τα πλεονεκτήματα των Ασύρματων Δικτύων Αισθητήρων είναι η εύκολη και γρήγορη τοποθέτηση του σε συνθήκες πολέμου, η χωρίς λάθη λειτουργία του κάτω από δύσκολες συνθήκες και η απρόσκοπτη λειτουργία τους σε περίπτωση που ένα κομμάτι του δικτύου καταστραφεί, λόγω της πυκνής τοποθέτησής τους οι εναπομείναντες κόμβοι εξακολουθούν να λειτουργούν. Πέραν όμως της εγκατάστασης των αισθητήρων σε οπτικά συστήματα, χρησιμοποιούνται ακόμα και για την παρακολούθηση και φύλαξη εγκαταστάσεων, εξοπλισμού, πυρομαχικών, αναγνώριση εδάφους καθώς και μια πιθανής Πυρηνικής επίθεσης.

### **2.3.3. Περιβαλλοντολογικές Εφαρμογές**

Τα ασύρματα δίκτυα αισθητήρων βρίσκουν ευρεία εφαρμογή στην παρακολούθηση του περιβάλλοντος, συμβάλλοντας στην εποπτεία των επιβλαβών περιβαλλοντικών μεταβολών, βοηθώντας στην πρόληψη και την καταστολή τους. Με τα δίκτυα αισθητήρων σε ένα δάσος μπορούμε να ελέγχουμε τις μεταβολές στη χλωρίδα και την πανίδα, ώστε να είμαστε σε θέση να αντιμετωπίσουμε μια καταστροφή, η οποία προκαλείται, είτε από φυσικά αίτια, είτε από τον ανθρώπινο παράγοντα, για παράδειγμα ανίχνευση μιας φωτιάς. Αντίστοιχα και στη θάλασσα, ελέγχοντας μια πιθανή μόλυνση των υδάτων ή τη συμπεριφορά των θαλάσσιων οργανισμών.

Επιπλέον τα δίκτυα αισθητήρων χρησιμοποιούνται για την παρατήρηση μετεωρολογικών φαινομένων και γεωγραφική έρευνα.

#### **2.3.4. Ανίχνευση δασικών πυρκαγιών**

Ένα ασύρματο δίκτυο αισθητήρων είναι το πλέον κατάλληλο μέσο για την πρόληψη μια δασικής πυρκαγιάς και την έγκαιρη ειδοποίηση των αρμοδίων πριν εξαπλωθεί ανεξέλεγκτα. Οι αισθητήρες έχουν την δυνατότητα να δώσουν πληροφορίες για το μέγεθος και το ακριβές σημείο που έχει ξεσπάσει μια πυρκαγιά. Χάρη την εξελιγμένη ασύρματη επικοινωνία που διαθέτουν, μπορούν να καλύψουν μεγάλες εκτάσεις σε δάση και βουνά απροβλημάτιστα, παρακάμπτοντας εμπόδια, πυκνή βλάστηση και δύσβατα σημεία. Ακόμα, για την ενεργειακή τους κάλυψη εκμεταλλεύονται την ηλιακή ενέργεια και με τη βοήθεια φωτοβολταϊκών συστημάτων μπορούν να είναι ενεργειακά αυτόνομοι για πολλά χρόνια, έτσι ώστε να μην χρειάζεται ανθρώπινη παρέμβαση σχεδόν ποτέ.

#### **2.3.5. Ανίχνευση πλημμυρών**

Για την ανίχνευση πλημμυρών μπορούν να χρησιμοποιηθούν διάφοροι τύποι αισθητήρων και μοντέλων. Αρχικά θα μπορούσε να γίνει πρόβλεψη μιας πλημμύρας αναλύοντας τα μετεωρολογικά δεδομένα που έχουν προηγουμένως συλλεχθεί. Αν όμως μια πλημμύρα είναι ήδη σε εξέλιξη πρέπει εγκαίρως να ανιχνευθεί για να ληφθούν τα απαραίτητα μέτρα ασφαλείας. Ανίχνευση μια πλημμύρας μπορεί να γίνει με διάφορους τύπους αισθητήρων, ένας τύπος αισθητήρας είναι ο αισθητήρας βροχόπτωσης, ο οποίος μετρά την ένταση της βροχής, σύμφωνα με τις σταγόνες που δέχεται στην επιφάνεια του. Ένας άλλος τύπος αισθητήρα είναι ο αισθητήρας μέτρησης του ύψους της βροχής, έτσι μπορούμε να έχουμε εικόνα του όγκου του νερού που πέφτει σε μια περιοχή. Ακόμα ένας παρόμοιος αισθητήρας χρησιμοποιείται για να μας δείξει το ύψος του νερού που συγκεντρώνεται σε ένα σημείο, για παράδειγμα σε ένα ποτάμι ή μια λίμνη, έτσι ώστε να μπορούμε έγκαιρα να γνωρίζουμε αν υπάρξει υπερχειλίση και κατά συνέπεια πλημμύρα της περιοχής.



### **2.3.6 Γεωργία Ακρίβειας (Precision Agriculture)**

Στην περίπτωση της γεωργίας μπορούν να χρησιμοποιηθούν αισθητήρες οι οποίοι κάνουν μετρήσεις, τα δεδομένα των οποίων αναλύονται και δίνουν στους γεωργούς χρήσιμες πληροφορίες για την εξέλιξη της παραγωγής τους. Οι αισθητήρες αυτοί μπορούν να μετρούν τιμές για τα συστατικά του εδάφους φύτευσης, για την ποιότητα του νερού ποτίσματος, τις συνθήκες του περιβάλλοντος που αναπτύσσεται η παραγωγή και το επίπεδο μόλυνσης του αέρα. Τα δεδομένα αυτά αναλύονται από τους γεωπόνους και στη συνέχεια ανάλογα με τα αποτελέσματα, ακολουθούνται οι κατάλληλες μέθοδοι για μια υγιείς παραγωγή.

### **2.3.7 Εφαρμογές Υγείας**

Στο χώρο της υγείας οι αισθητήρες έχουν παίξει σημαντικό ρόλο στην ανάπτυξη ιατρικών εργαλείων, στην φαρμακοβιομηχανία, αλλά και στην αύξηση της ποιότητας ζωής των ασθενών. Όσον αφορά τα την ιατρική τα τελευταία χρόνια, με τη σημαντική συμβολή των αισθητήρων, έχουν αναπτυχθεί σύγχρονα εξεταστικά συστήματα που βοηθούν στην πρόληψη και την αντιμετώπιση ασθενειών. Στη φαρμακοβιομηχανία οι αισθητήρες βοηθούν στην γραμμή παραγωγής των φαρμάκων και στον έλεγχο ποιότητας τους μετά την παραγωγή. Ενώ οι ασθενείς έχουν καλύτερη περίθαλψη και μεγαλύτερη άνεση στην μετέπειτα ζωή τους, ειδικά αυτοί που αντιμετωπίζουν προβλήματα κινητικότητας και χρόνιες παθήσεις. Τέλος οι αισθητήρες έχουν βοηθήσει πολύ στην ανάπτυξη ειδών για άτομα με ειδικές ανάγκες και στην παρακολούθηση εξ αποστάσεων ασθενών.

### **2.3.8. Οικιακές Εφαρμογές**

#### **2.3.8.1. Αυτοματισμός σπιτιού**

Ένα σπίτι δε θα μπορούσε να μείνει χωρίς αισθητήρες, είναι ο χώρος που οι περισσότεροι περνάνε το μεγαλύτερο μέρος της ημέρας τους, οπότε οι ευκολίες που προσφέρουν οι αισθητήρες δε θα μπορούσαν να λείπουν από αυτό. Οι αισθητήρες σε ένα σπίτι χρησιμοποιούνται κυρίως για τρεις λόγους, πρώτον ασφάλεια, δεύτερον εξοικονόμηση ενέργειας και τρίτον προσφέρουν ευκολία και άνεση στην καθημερινότητα. Επιγραμματικά να αναφέρουμε κάποια παραδείγματα, ο συναγερμός μας χρησιμοποιεί αισθητήρες κίνησης για να ανιχνεύσει την πιθανή είσοδο διαρρηκτών. Ακόμα

χρησιμοποιούμε αισθητήρες παρουσίας, ούτως ώστε αν δεν υπάρχει κανείς στο σπίτι να κλείνουν τα ξεχασμένα φώτα για να εξοικονομούμε ενέργεια. Τέλος οι αισθητήρες είναι πλέον ενσωματωμένοι σε όλες τις ηλεκτρικές μας συσκευές, από την ηλεκτρική σκούπα μέχρι την τηλεόραση, έτσι μας προσφέρουν επιπλέον υπηρεσίες και ανέσεις στην οικιακή καθημερινότητα, για παράδειγμα η ηλεκτρική σκούπα μπορεί να σκουπίζει μόνη της χάρις τους αισθητήρες που ανιχνεύουν τους χώρους που κινείται και τα εμπόδια που συναντάει.

### **2.3.9. Αλληλεπιδραστικά Μουσεία**

Στα σύγχρονα μουσεία η τεχνολογία εφαρμόζεται σε πολλούς τομείς, από την συντήρηση των εκθεμάτων μέχρι τη διαχείριση των επισκεπτών και των εγκαταστάσεων. Σχετικά με τη διαχείριση του κτηρίου, οι αισθητήρες μπορούν να εφαρμοστούν, όπως αναφέραμε και παραπάνω, για στην εξοικονόμηση ενέργειας, για παράδειγμα αισθητήρες φωτός μετρούν την ένταση του φωτός στους χώρους του μουσείου και αναλόγως ανοίγουν και κλείνουν τις περσίδες για να υπάρχει πάντα μια σταθερή ένταση φωτός στο χώρο. Πέραν όμως αυτής της χρήσης, οι αισθητήρες μπορούν να χρησιμοποιηθούν και στην αλληλεπίδραση των επισκεπτών και των εκθεμάτων. Για παράδειγμα όταν θα εισέρχεται ένας επισκέπτης στο μουσείο το έξυπνο εισιτήριο του θα φέρει πληροφορίες για την εθνικότητα του και κατά συνέπεια και για την γλώσσα ξενάγησης που επιθυμεί. Αυτό θα είχε ως αποτέλεσμα, όταν ο επισκέπτης στέκεται με μια οθόνη πληροφοριών ή μια προθήκη με εκθέματα, το έξυπνο εισιτήριο του να “διαβάζεται” και οι πληροφορίες και οι ετικέτες που συνοδεύουν τα εκθέματα, να μεταφράζονται στην γλώσσα που επιθυμεί και κατανοεί καλύτερα.

### **2.3.10. Παρακολούθηση και ανίχνευση οχημάτων**

Οι αισθητήρες και κατ' επέκταση τα ασύρματα δίκτυα αισθητήρων χρησιμοποιούνται για την διαχείριση των οχημάτων και της κυκλοφορίας. Αισθητήρες τοποθετημένοι σε δρόμους, μετρούν τον αριθμό των οχημάτων που περνούν και μετά από επεξεργασία των δεδομένων αποφασίζεται αν τα υπόλοιπα οχήματα θα οδηγηθούν από άλλο δρόμο για να μην υπάρξει μπουτιλιάρισμα. Παράλληλα, επιγραφές στο δρόμο ειδοποιούν τους οδηγούς για την αλλαγή κατεύθυνσης, τα όρια ταχύτητας και τον χρόνο που θα χρειαστούν μέχρι την άφιξη στον προορισμό τους. Αισθητήρες όμως, μπορούν να χρησιμοποιηθούν και σε χώρους στάθμευσης οχημάτων, ώστε να είναι πλήρως αυτοματοποιημένος, σε τέτοιο βαθμό που να μην χρειάζεται ανθρώπινη παρουσία για την λειτουργία του.

## ΚΕΦΑΛΑΙΟ 3. Τεχνολογίες Υλοποίησης

Σε αυτήν την ενότητα παρουσιάζονται οι τεχνολογίες οι οποίες χρησιμοποιήθηκαν για την υλοποίηση της πτυχιακής εργασίας. Πιο συγκεκριμένα, χρησιμοποιήθηκε η γλώσσα HTML5, η γλώσσα javascript, η γλώσσα CSS3, η γλώσσα jquery, το πλαίσιο εφαρμογής bootstrap, το Grafana, η βάση δεδομένων mysql, ένας Apache web server, η πλατφόρμα Arduino και η πλατφόρμα Raspberry Pi.

### 3.1. HTML5



Εικόνα 10 - HTML5

Η HTML (HyperText Markup Language - Γλώσσα σήμανσης υπερκειμένου) είναι η γλώσσα η οποία χρησιμοποιείται για την δημιουργία και σχεδιασμό ιστοσελίδων. Οι περιηγητές (για παράδειγμα internet Explorer, Mozilla, Chrome κ.α.) αναγνωρίζουν και μεταφράζουν τα στοιχεία της HTML. Όμως, όπως αναφέρθηκε και νωρίτερα στο κείμενο, ο όρος HTML5 αναφέρεται σε ένα σύνολο τεχνολογιών και όχι απλά σε μια νέα έκδοση της γλώσσας σήμανσης HTML. Η HTML5 είναι η πιο πρόσφατη έκδοση της HTML. Το σημαντικό με αυτήν την έκδοση όμως, είναι ότι επιτρέπει στους

προγραμματιστές να δημιουργήσουν mobile web sites (ιστότοπους συμβατούς με κινητές συσκευές) τα οποία να έχουν ταυτόχρονα πολλές από τις ιδιότητες των εφαρμογών που συναντάμε στις διάφορες πλατφόρμες κινητών, ανοίγοντας νέους ορίζοντες έτσι στην παρουσία του διαδικτύου στην όλη εμπειρία με τα κινητά τηλέφωνα.

### **3.2. Javascript**



Εικόνα 11 - Javascript

Η Javascript, είναι ερμηνευόμενη (interpreted) γλώσσα προγραμματισμού η οποία έχει ως περιβάλλον εκτέλεσης τον περιηγητή. Η Javascript μπορεί να διαχειριστεί με δυναμικό τρόπο τα περιεχόμενα και την διεπαφή μιας HTML σελίδας και επιτρέπει την αλληλεπίδραση με τον χρήστη. Επιπλέον, η πλειοψηφία των διεπαφών και λειτουργιών της HTML5 μπορούν να κληθούν μόνο μέσω Javascript.

### 3.3. CSS3



Εικόνα 12 - CSS3

Η CSS (Cascading Style Sheets) αποτελεί γλώσσα προσδιορισμού εμφάνισης φύλλων (style sheet language). Ανάλογα με τη σύνταξη της, καθορίζεται η εμφάνιση ενός ιστοτόπου. Στην εφαρμογή που αναπτύχθηκε χρησιμοποιείται η CSS3 (η τελευταία έκδοση της CSS). Κατά την ανάπτυξη της εφαρμογής χρησιμοποιήθηκε στην σχεδίαση της διεπαφής της.

### **3.4. jQuery**



Εικόνα 13 - jQuery

Η jQuery χρησιμοποιείται για την δυναμική διαχείριση των δεδομένων και της διεπαφής μιας ιστοσελίδας HTML. Είναι από τις πιο δημοφιλείς βιβλιοθήκες της javascript.

### **3.5. Bootstrap**



Εικόνα 14 - Bootstrap

Το Bootstrap είναι ένα ελεύθερο λογισμικό, το οποίο ενσωματώνει εργαλεία για την κατασκευή ιστοσελίδων και εφαρμογών διαδικτύου. Το Bootstrap παρέχει διάφορα έτοιμα στυλ προς χρήση τα οποία μπορούν να εξελιχθούν περαιτέρω με τα ενσωματωμένα εργαλεία του. Ενώ παρέχει CSS και HTML και επεκτάσεις javascript για τη μορφοποίηση των σελίδων, τα κουμπιά πλοήγησης, τα στοιχεία περιβάλλοντος και για τις γραμματοσειρές του.

Για χρησιμοποιηθούν τα παραπάνω στην κατασκευή μια σελίδας HTML, πρέπει να γίνει λήψη του πακέτου εγκατάστασης του Bootstrap και των CSS στυλ του. Ενώ όλες οι επεκτάσεις του, όπως JQuery και Javascript, πρέπει να ληφθούν εκ νέου και να εγκατασταθούν στο ήδη εγκατεστημένο βασικό πακέτο.

### **3.6. Grafana**



Εικόνα 15 - Grafana

Το Grafana είναι η πιο δημοφιλής ανοιχτού κώδικα εφαρμογή για οπτικοποίηση μεγάλου εύρους δεδομένων. Δίνει τη δυνατότητα της παρουσίασης χρονοσειρών σε μορφή φιλική προς το χρήστη αλλά και εύκολα παραμετροποιήσιμη από τον διαχειριστή. Το Grafana χρησιμοποιείται κυρίως σε διαδικτυακές εφαρμογές (π.χ. web analytics). Η τελευταία έκδοση της Grafana (η οποία χρησιμοποιήθηκε στην πτυχιακή) είναι η 2.6 η οποία παρουσιάστηκε τον Δεκέμβριο του 2015.

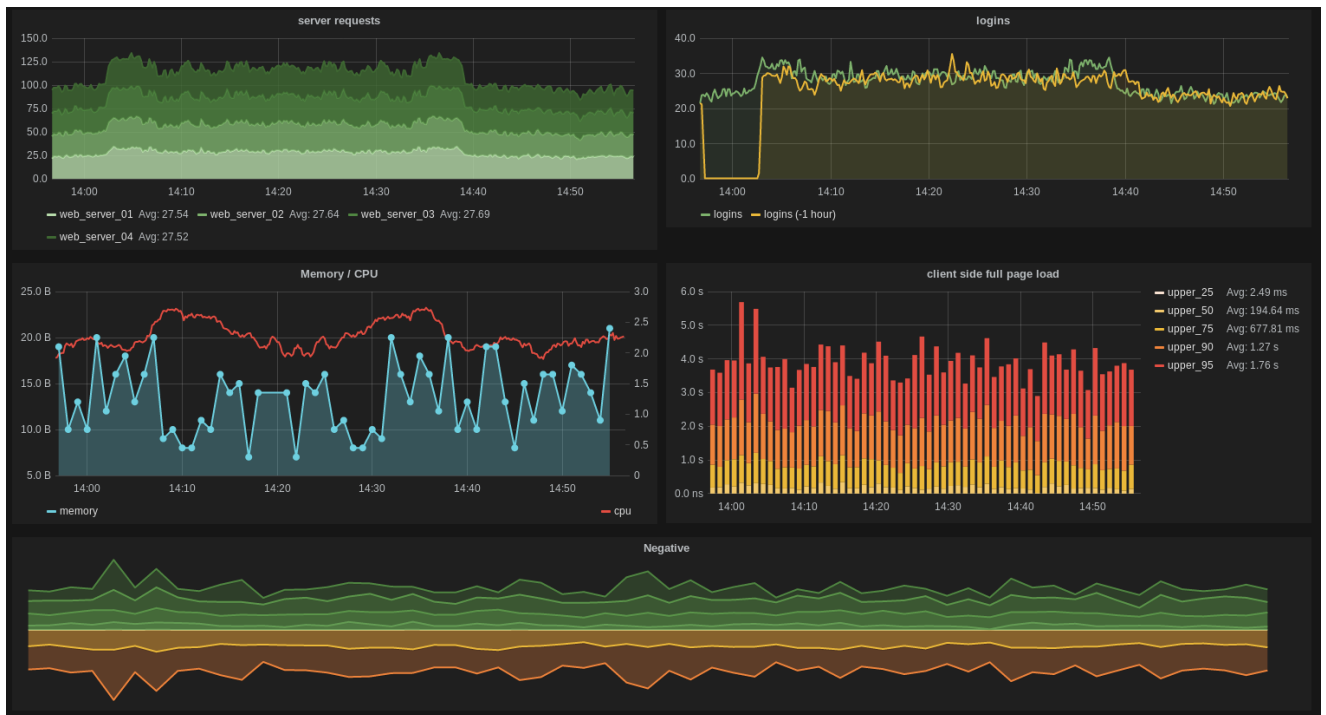
Η Grafana διαθέτει pluggable panels (πίνακες) και data sources (πηγές δεδομένων) οι οποίες επιτρέπουν επεκτασιμότητα των εφαρμογών. Η Grafana υποστηρίζει σύνολα δεδομένων όπως



Graphite, InfluxDB, OpenTSDB, KairosDB, Prometheus, και σε επόμενες εκδόσεις θα υποστηρίζει και SQL Server.

Τα κυριότερα χαρακτηριστικά της Grafana είναι τα παρακάτω

- Γρήγορα και ευέλικτα γραφήματα τα οποία δημιουργούνται στην πλευρά του χρήστη με μία σειρά από επιλογές
- Επέλεξε περιοχή για μεγέθυνση
- Πολλαπλές Y-Axes
- Μπάρες, Γραμμές, Σημεία
- Thresholds, Logarithmic scales (Λογαριθμικές κλίμακες)
- Y-axis μορφή (bytes, milliseconds, κ.α.)
- Προβολή ή μεταβολή των γραφημάτων σε πλήρης οθόνη
- Πλήρης έλεγχος στην σχεδίαση της κάθε χρονοσειράς
- Δυνατότητα ανάμιξης διαφορετικών τύπων γράφων lines, bars και points
- Ανάμεικτες stacked σειρές με απομονωμένες σειρές.
- Δυνατότητα εξαγωγή του γραφήματος σε εικόνα τύπου png. (μέσω UI ή HTTP API)



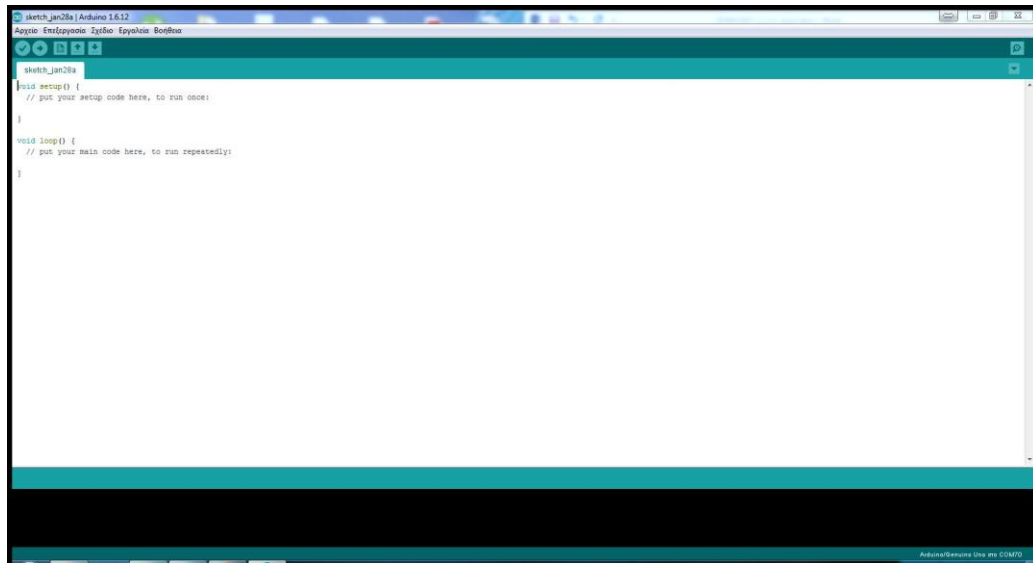
Εικόνα 16 - Grafana Graphs - Dashboard

### 3.7. Arduino IDE



Εικόνα 17 – Arduino IDE

Το Arduino IDE είναι το πρόγραμμα με το οποίο συντάσσουμε και αποστέλλουμε τον κώδικα στις πλακέτες Arduino. Είναι συμβατό με σχεδόν όλα τα γνωστά λειτουργικά συστήματα και παρέχει ένα άνετο και λειτουργικό γραφικό περιβάλλον, οποίο είναι φιλικό και εύκολα χρησιμοποιήσιμο και από τους πιο αρχάριους χρήστες.



Εικόνα 18 – Arduino IDE

Ενσωματώνει πολλά χαρακτηριστικά, όπως “σειριακό μόνιτορ” το οποίο μας δείχνει τις τιμές που εξάγει η πλακέτα μας, βιβλιοθήκες με έτοιμες συναρτήσεις για τα διάφορα πρόσθετα που

συνδέουμε στις πλακέτες Arduino, στην προκειμένη περίπτωση χρησιμοποιήσαμε τις βιβλιοθήκες για τους αισθητήρες μας και για τα τις κεραίες που χρησιμοποιήσαμε.

### **3.8. Python**



Εικόνα 19 - Python

Η Python είναι η γλώσσα που χρησιμοποιήσαμε για να γράψουμε τον κώδικα που χρησιμοποιήσαμε στο Raspberry για να συλλέγει και να καταχωρεί τα δεδομένα στη σχεσιακή βάση δεδομένων.

Είναι μια γλώσσα προγραμματισμού υψηλού επιπέδου που δημιουργήθηκε το 1990, από τον Ολλανδό Γκβίντο βαν Ρόσσουμ. Τα κύρια χαρακτηριστικά της είναι η αναγνωσιμότητα του κώδικα της, η ευκολία χρήσης της από τους προγραμματιστές, η ταχύτητα εκμάθησης της, το πλήθος των βιβλιοθηκών της και το απλό συντακτικό της, χάρις στο οποίο μπορούν να εκφραστούν έννοιες με λιγότερες γραμμές κώδικα, σε σχέση με άλλες γλώσσες προγραμματισμού.

Μπορεί να εγκατασταθεί στα περισσότερα λειτουργικά συστήματα, ενώ η διαχείρισή της γίνεται από τον οργανισμό Python Software Foundation. Και η διανομή της είναι δωρεάν με την άδεια Python Software Foundation License.

### 3.9. Raspbian



Εικόνα 20 - Raspbian

Το Raspbian είναι το λειτουργικό σύστημα που χρησιμοποιήσαμε για να εγκατασταθεί και να τρέξει το Raspberry Pi 3.

Είναι ένα λειτουργικό σύστημα για το Raspberry Pi που βασίζεται στο Debian. Υπάρχουν διάφορες εκδόσεις του Raspbian, συμπεριλαμβανομένου του Raspbian Stretch και του Raspbian Jessie. Από το 2015 έχει κηρυχθεί επίσημα από το Ίδρυμα Raspberry Pi, ως το κύριο λειτουργικό σύστημα για την οικογένεια ηλεκτρονικών υπολογιστών Raspberry Pi. Το Raspbian δημιουργήθηκε από τους Mike Thompson και Peter Green. Η αρχική κατασκευή ολοκληρώθηκε τον Ιούνιο του 2012. Το λειτουργικό σύστημα βρίσκεται ακόμη υπό ενεργό ανάπτυξη. Το Raspbian είναι εξαιρετικά βελτιστοποιημένο για τους επεξεργαστές ARM χαμηλής απόδοσης της σειράς Raspberry Pi.

Το Raspbian χρησιμοποιεί το PIXEL, το Pi Improved Xwindows Environment, το ελαφρύ ως κύριο περιβάλλον επιφάνειας εργασίας από την τελευταία ενημέρωση. Αποτελείται από ένα τροποποιημένο περιβάλλον επιφάνειας εργασίας LXDE και το διαχειριστή παραθύρων Openbox με ένα νέο θέμα και λίγες άλλες αλλαγές. Η διανομή αποστέλλεται με ένα αντίγραφο του προγράμματος υπολογιστικής άλγεβρας Mathematica και μια έκδοση του Minecraft που ονομάζεται Minecraft Pi καθώς και μια ελαφριά έκδοση του Chrome από την τελευταία έκδοση.

### 3.10. Apache HTTP εξυπηρετητής



Εικόνα 21 - Apache Web Server

Κάθε ιστότοπος που επισκεπτόμαστε φιλοξενείται σε ένα εξυπηρετητή (server). Ένας τέτοιος εξυπηρετητής είναι και ο Apache server. Όταν πληκτρολογούμε μια διεύθυνση ιστότοπου στο πρόγραμμα περιήγησης (browser), αυτό επικοινωνεί με τον εξυπηρετητή με το πρωτόκολλο HTTP. Σε αυτή την περίπτωση η δουλειά του εξυπηρετητή είναι να παράξει την ιστοσελίδα και να τη στείλει στο πρόγραμμα περιήγησης προς εμφάνιση της στον πελάτη (client).

Πρέπει να σημειώσουμε ότι ο Apache server είναι λογισμικό ανοιχτού κώδικα και κυκλοφορεί υπό την άδεια της apache, ενώ τρέχει στα γνωστότερα λειτουργικά συστήματα και συνεργάζεται με συστήματα διαχείρισης Βάσης Δεδομένων όπως MySQL και Oracle. Η πρώτη του έκδοση κυκλοφόρησε το 1993 από τον Robert McCool και στην Ελλάδα κατέχει σχεδόν το 50% του μεριδίου της αγοράς.

### **3.11. MySQL**



Η MySQL είναι ένα από τα γνωστότερα RDBMS (relational database management system), ένα από τα γνωστότερα δηλαδή συστήματα διαχείρισης σχεσιακών βάσεων δεδομένων και χρησιμοποιεί την SQL (Structured Query Language), που είναι η δημοφιλέστερη γλώσσα για την προθήκη, την πρόσβαση και την επεξεργασία των δεδομένων μιας Βάσης Δεδομένων.

Τα κύρια χαρακτηριστικά της MySQL είναι η ταχύτητα, η αξιοπιστία και η ευελιξία της. Λειτουργεί σε περιβάλλον Windows, Unix και Linux και είναι ανοιχτού κώδικα, ούτως ώστε μπορεί να ληφθεί από τον οποιοδήποτε και να προσαρμοστεί στις ανάγκες του.

### **3.12. Raspberry Pi**

Το Raspberry Pi αποτελεί έναν υπολογιστή σε μέγεθος πιστωτικής κάρτας. Αναπτύχθηκε από την Raspberry Pi Foundation και σκοπό είχε την προώθηση της διδασκαλίας της επιστήμης των υπολογιστών σε σχολεία. Είναι μια συσκευή η οποία παρουσιάζει πολλές δυνατότητες και είναι ικανή για λειτουργίες που συνήθως τις κάνει ένας προσωπικός υπολογιστής, όπως προβολή ταινιών, παιχνίδια, πλοήγηση στο διαδίκτυο κλπ.

Λόγω του μικρού μεγέθους και των υψηλών του δυνατοτήτων του, χρησιμοποιείται σε πολλά project. Ο λόγος επιλογής αυτής της πλακέτας (board), ήταν ότι αποτελεί ένα σύγχρονο

υπολογιστικό σύστημα με πολλές λειτουργικότητες και χαμηλό κόστος. Έχουν κυκλοφορήσει δύο μοντέλα raspberry pi (Model A & Model B) ενώ πρόσφατα κυκλοφόρησε και μία βελτίωση του δεύτερου μοντέλου (Model B+).

Όπως παρατηρούμε στο παρακάτω σχήμα, η διαφορά των δύο μοντέλων είναι ότι το Model B+ διαθέτει θύρα Ethernet, μία παραπάνω θύρα USB 2.0 και 256 MB μνήμης Ram περισσότερο. Για την διπλωματική εργασία επιλέχθηκε το Raspberry Pi Model B+.

Το raspberry pi διαθέτει μία SD Card η οποία χρησιμοποιείται για την αποθήκευση αρχείων. Επίσης η SD Card είναι απαραίτητη στην Boot διαδικασία καθώς όλα τα απαραίτητα αρχεία για την διαδικασία αυτή αποθηκεύονται σε ένα FAT32 partition της κάρτας.

Όπως μπορεί να παρατηρήσει κανείς στο raspberry pi, υπάρχει ένας Header ο οποίος διαθέτει 40 ακίδες. Πρόκειται για τον P1 Header. Συνολικά, στο board υπάρχουν ακόμη πέντε Headers αλλά δεν είναι ενεργοποιημένοι.

Ο P1 Header είναι γνωστός και ως expansion header gpio connector P1. Σε αυτό τον Header πέραν από κάποιες ακίδες τροφοδοσίας 5V, 3.3V και GND, οι υπόλοιπες ακίδες έχουν κάποιες συγκεκριμένες λειτουργίες που επιτρέπουν την επικοινωνία και σύνδεση του raspberry pi με κάποιες άλλες συσκευές. Η τροφοδοσία στο raspberry επιτυγχάνεται μέσω μιας mini usb θύρας που υπάρχει στο board ή μέσω των 5V και GND pins.

Τα χαρακτηριστικά του είναι τα παρακάτω:

- 1 θύρα HDMI (για την σύνδεση με την οθόνη)
- θύρες USB (μπορούν να χρησιμοποιηθούν για ποντίκι και πληκτρολόγιο)
- 1 micro usb (για την τροφοδοσία)
- 1 θύρα ethernet (για σύνδεση στο τοπικό δίκτυο)
- 1 κάρτα SD (παίζει το ρόλο του αποθηκευτικού μέσου, όχι μόνο για τα αρχεία σας αλλά και για το λειτουργικό που θα τρέξει)
- 1 rca video έξοδος (σε περίπτωση που δεν υπάρχει οθόνη με HDMI)
- 1 αναλογική έξοδος (για τα ηχεία)

- 1 micro usb (για την τροφοδοσία)



Εικόνα 23 - Raspberry pi & pi 2

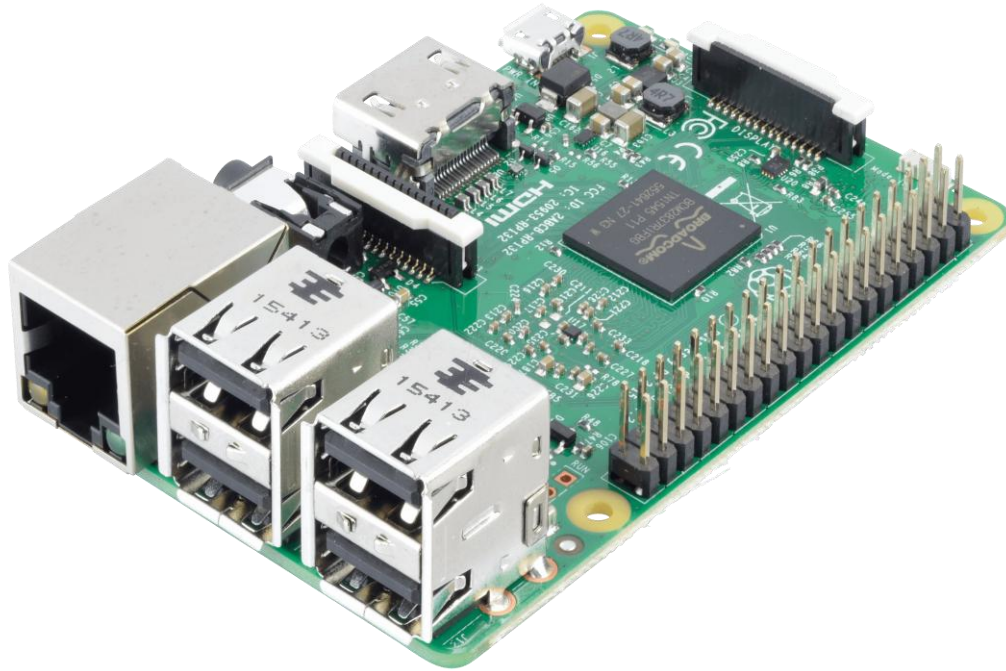
Πιο αναλυτικά, τα χαρακτηριστικά του Raspberry pi 2 σε σχέση με 1 είναι. Το πρώτο χαρακτηριστικό είναι ότι έχει αλλάξει ο επεξεργαστής και είναι ο BCM2836 ο οποίος αποτελείται από τέσσερις πυρήνες και είναι στα 900MHz. Επιπλέον, η cache του επεξεργαστή ήταν 16KB και αναβαθμίστηκε σε 32KB. Οπότε αυτόματα δίνει αρκετή δύναμη έτσι ώστε να βελτιωθούν λειτουργίες όπως η περιήγηση στο διαδίκτυο και να υπάρχουν πολλές παράλληλες λειτουργίες (multithread) όπως για παράδειγμα σε άλλον επεξεργαστή να τρέχει το γραφικό περιβάλλον και σε άλλον η επεξεργασία των δεδομένων από την sd κάρτα. Επίσης, διπλασιάστηκε η μνήμη RAM η οποία είναι πλέον 1GB. Σύμφωνα με τον Eben Upton η ταχύτητα έχει εξαπλασιαστεί CEO!

Μια ακόμη σημαντική διαφορά, είναι ότι η αρχιτεκτονική του νέου επεξεργαστή είναι ARMv7 η οποία επιτρέπει να δημιουργηθούν διανομές λειτουργικού συστήματος όπως ubuntu και όλα τα λειτουργικά που μπορεί να έχει ένας επεξεργαστής ARM. Επίσης αίσθηση έχει προκαλέσει ότι το Raspberry Pi 2 θα είναι συμβατό με τα Windows 10.



Όσον αφορά τη συμβατότητα, οι είσοδοι-έξοδοι γενικού σκοπού παραμένουν οι ίδιες. Επίσης έχει 4 USB οι οποίες όμως παρέχουν περισσότερο ρεύμα και είναι δυνατόν να συνδεθούν συσκευές που έχουν μεγαλύτερες απαιτήσεις σε ρεύμα.

### 3.12.1. Raspberry Pi 3



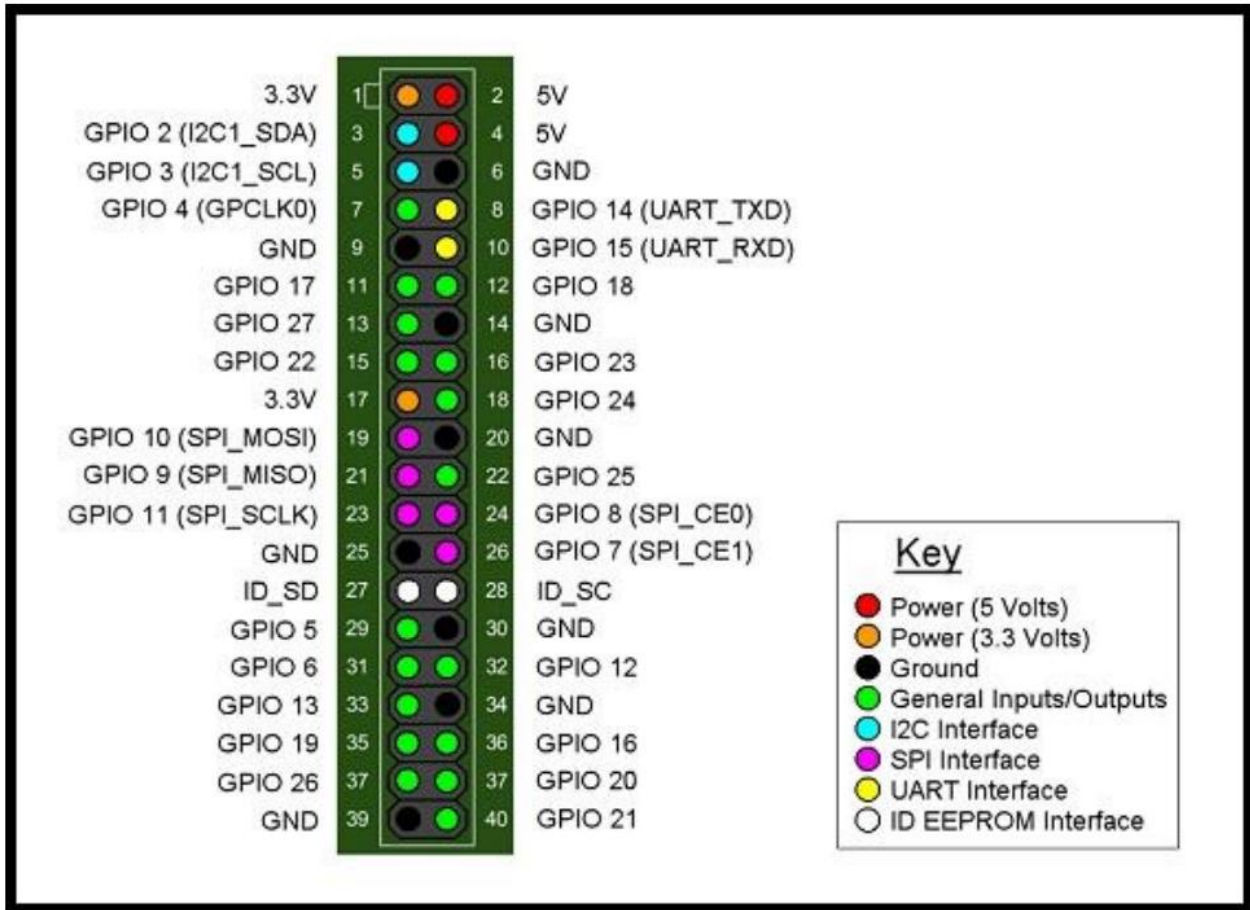
Εικόνα 24 - Raspberry Pi 3

Το Raspberry Pi 3 χρησιμοποιεί ένα System – On – Chip Broadcom BCM2837, το οποίο περιλαμβάνει έναν τετραπύρνο επεξεργαστή ARM Cortex - A53, ο οποίος «τρέχει» στα 1.2 GHz, είναι 64-bit και διαθέτει μία κοινή L2 cache (κρυφή μνήμη) του μεγέθους των 512 KB. Η υπολογιστική ισχύς του Raspberry Pi 2B υπολογίζεται σαν ένας Athlon Thunderbird με ταχύτητα 1.1 GHz και πολύ καλύτερα γραφικά, ενώ του Raspberry Pi 3 υπολογίζεται ότι είναι 50% γρηγορότερη απ' αυτή του 2B . Η κάρτα γραφικών είναι η Broadcom VideoCore IV, αλλά στο συγκεκριμένο μοντέλο έχει γρηγορότερες ταχύτητες απ' αυτήν των 250 MHz των προηγούμενων γενιών. Είναι ικανή να απεικονίσει την εικόνα στις τυπικές αναλύσεις που έχουν οι σημερινές τηλεοράσεις, δηλαδή HD και Full HD. Επίσης, πρέπει να σημειώσουμε ότι διαθέτει 1 GB μνήμη

τύπου RAM, RJ45 Ethernet θύρα καθώς και 2.4 Ghz Wi-fi (με ρυθμό μετάδοσης 600 Mbit/s), αλλά και Bluetooth 4.1 (με ρυθμό μετάδοσης 24 Mbit/s).

Τα περιφερειακά, όπως το ποντίκι και το πληκτρολόγιο μπορούν αν συνδεθούν σε κάθε μία απ' τις 4 θύρες USB, οι οποίες μπορούν να γίνουν περισσότερες αν χρησιμοποιήσουμε USB hub. Όσον αφορά την κατανάλωση, θέλει είσοδο ρεύματος στα 2.5 A, ενώ υπολογίζεται ότι η κατανάλωση είναι 4.0 W. Τέλος, όπως όλα τα Raspberry, έχει ειδική θύρα για την κάμερα που συνδέεται με καλωδιωταινία, καθώς κι έξοδο HDMI για σύνδεση με την οθόνη μας.

Η GPIO (θύρες εισόδου - εξόδου γενικής χρήσεως) της τρίτης γενιάς έχει 40 ακροδέκτες. Έχει ακροδέκτες που δίνουν τάση της τάξεως των 5 V, ακροδέκτες που δίνουν τάση της τάξεως των 3.3 V, ακροδέκτες που χρησιμοποιούνται για γείωση, καθώς και γενικές θύρες εισόδου – εξόδου. Επίσης, πρέπει να σημειώσουμε ότι έχει ακόμη θύρες για I2C, SPI, UART και ID EEPROM αν και τις συγκεκριμένες θύρες στην κατασκευή μας δεν θα τις χρησιμοποιήσουμε. Κάθε ακροδέκτης της GPIO αντιστοιχεί και σε μία διαφορετική ονομασία, γεγονός που μας βοηθάει στον προγραμματισμό.



Εικόνα 25 - Raspberry Pi 3 GPIO

Κατά γενική ομολογία τα Raspberry Pi, άρα και η τρίτη γενιά τους δεν χρειάζονται απαγωγείς θερμότητας. Αυτό συμβαίνει, επειδή το τσιπ που χρησιμοποιούν είναι το ανάλογο με αυτό ενός κινητού τηλεφώνου και γι' αυτόν τον λόγο δεν είναι σύνηθες να υπερθερμανθεί εκτός αν θέλουμε να πάρουμε το 100% του Raspberry ή αν εφαρμόσουμε την τεχνική του overclocking, το οποίο με τη σειρά του παρέχει τη δυνατότητα στο Raspberry Pi να «τρέχει» με ταχύτητες που είναι 1000 MHz ARM, 500 MHz πυρήνα, 600 MHz SDRAM δηλαδή σχεδόν κοντά στην τρίτη γενιά.

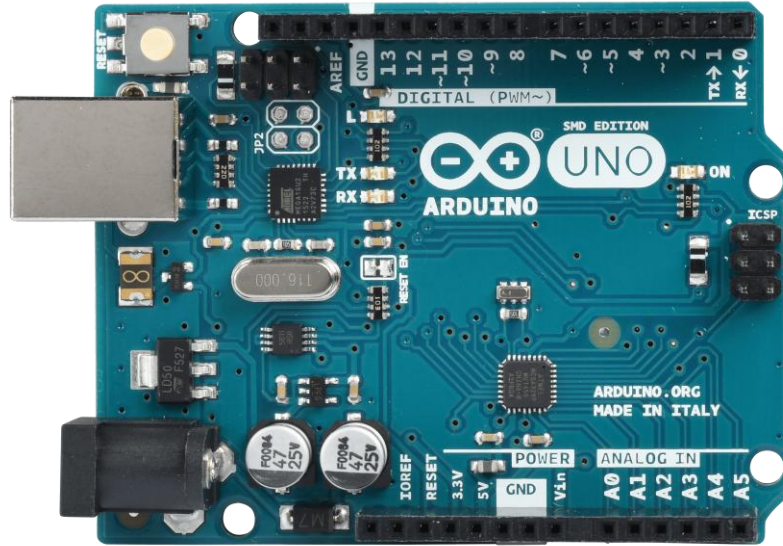


Εικόνα 26 - Raspberry Pi Heatsinks

Board	Raspberry Pi 2 Model B	Raspberry Pi 3 Model B
Processor	Broadcom BCM2836	Broadcom <b>BCM2837</b>
CPU Core	Quadcore ARM Cortex-A7, 32Bit	Quadcore <b>ARM Cortex-A53, 64Bit</b>
Clock Speed	900 MHz	1.2GHz (Roughly 50% faster than Pi2)
RAM	1 GB	1 GB
GPU	250 MHz VideoCore IV®	<b>400 MHz</b> VideoCore IV®
Network Connectivity	1 x 10 / 100 Ethernet (RJ45 Port)	1 x 10 / 100 Ethernet (RJ45 Port)
Wireless Connectivity	None	<b>802.11n wireless LAN (WiFi) and Bluetooth 4.1</b>
USB Ports	4 x USB 2.0	4 x USB 2.0
GPIOs	2 x 20 Pin Header	2 x 20 Pin Header
Camera Interface	15-pin MIPI	15-pin MIPI
Display Interface	DSI 15 Pin / HDMI Out / Composite RCA	DSI 15 Pin / HDMI Out / Composite RCA
Power Supply (Current Capacity)	1.8 A	<b>2.5 A</b>

Εικόνα 27 - Σύγκριση Pi2 & Pi3

### 3.13. Arduino



Εικόνα 28 - Arduino

Το arduino αποτελεί μια πλατφόρμα μικροελεγκτή ανοιχτού κώδικα. Πρόκειται, ουσιαστικά, για μια μητρική πλακέτα με ενσωματωμένο μικροελεγκτή και εισόδους και εξόδους, αναλογικές και ψηφιακές. Είναι κατάλληλο για κατασκευές ψηφιακών συσκευών και διαδραστικών αντικειμένων που ελέγχουν και μετράνε ψηφιακά διάφορα μεγέθη. Ο προγραμματισμός του μικροελεγκτή πραγματοποιείται σε γλώσσα Arduino, η οποία βασίζεται στη Wiring και μοιάζει αρκετά στη δομή και στη σύνταξη της με τη γλώσσα C . Ο μικροελεγκτής ATmega328 έχει προεγκατεστημένο ένα πρόγραμμα εκκίνησης που επιτρέπει να φορτώνονται οι κώδικες στην πλακέτα χωρίς να χρειάζεται εξωτερική πλακέτα προγραμματισμού. Η επικοινωνία με τον μικροελεγκτή γίνεται με χρήση του πρωτοκόλλου STK500. Το περιβάλλον ανάπτυξης είναι βασισμένο και αναπτυγμένο σε Processing και μπορεί είτε να αξιοποιηθεί μόνο του είτε να επικοινωνήσει με πλήθος λογισμικού όπως είναι το MaxMSP, Flash, Processing και πολλά άλλα. Επίσης, τα σχέδια κατασκευής του είναι ανοιχτού κώδικα, όπως και όλα τα προγράμματα που συνοδεύουν τα shields του ή τις διάφορες εφαρμογές που πραγματοποιούν οι χρήστες του, επομένως έχουμε να κάνουμε με ένα αρκετά πλούσιο πεδίο πειραματισμού, μία αρκετά ενεργή κοινότητα και πλούσια γκάμα είτε συναρμολογημένων shields είτε βιβλιοθήκες και εξαρτήματα που έχουν δημιουργηθεί και δοκιμαστεί από την κοινότητά του. Η εφαρμογή μας σχεδιάστηκε με βάση το μοντέλο της πλατφόρμας Arduino Uno, ο οποίος έχει σαν

βάση τον μικροελεκτή ATmega328. Έχει 14 συνδέσμους (pins) ψηφιακών εισόδων/εξόδων (6 εξ αυτών μπορούν να χρησιμοποιηθούν και σαν PWM έξοδοι), 6 αναλογικές εισόδους, ένα κρυσταλλικό ρολόι συχνότητας 16MHz, USB επικοινωνία για σύνδεση με τον υπολογιστή, ένα καλώδιο τροφοδοσίας, μια ICSP κεφαλή και ένα κουμπί επανεκκίνησης (reset button). Περιέχει όλα όσα χρειάζονται για τον έλεγχο του μικροελεγκτή.

## Τροφοδοσία

Το Arduino Uno μπορεί να τροφοδοτηθεί είτε από την θύρα USB είτε με εξωτερική τροφοδοσία. Η εξωτερική τροφοδοσία μπορεί να είναι ή με την χρήση μετασχηματιστή από εναλλασσόμενη τάση σε συνεχή ή με την χρήση μπαταριών. Ο μετασχηματιστής συνδέεται μέσω βύσματος 2,1 χιλιοστών με τον θετικό πόλο στο κέντρο, στο βύσμα που βρίσκεται ενσωματωμένο στο Arduino. Η πλακέτα μπορεί να λειτουργήσει με εξωτερική τροφοδοσία από 6 έως 20V. Αν όμως τροφοδοτηθεί με λιγότερο από 7 V ο σύνδεσμος τροφοδοσίας 5 V που διαθέτει η πλακέτα ίσως να μην μπορέσει να παράγει 5V και η πλακέτα να είναι ασταθής. Αν χρησιμοποιηθούν περισσότερα από 12 V για τροφοδοσία ο ρυθμιστής τάσης μπορεί να υπερθερμανθεί και να χαλάσει την πλακέτα. Επομένως, προκύπτει πως η προτεινόμενη τροφοδοσία κυμαίνεται από 7 έως 12 V. Οι σύνδεσμοι τροφοδοσίας έχουν ως εξής:

- Vin: Είναι ίση με την τάση τροφοδοσίας που έχει εφαρμοστεί στο Arduino. Μπορεί να τροφοδοτήσει εξωτερικές πλακέτες.
- 5V: Προσφέρει τάση 5V που έχει προκύψει από τον ρυθμιστή τάσης της πλακέτας σε περίπτωση που η τροφοδοσία είναι μεγαλύτερη των 5V.
- 3,3V: Προσφέρει 3,3V που προέρχονται από τον ρυθμιστή τάσης της πλακέτας. Το μεγαλύτερο ρεύμα που καταναλώνεται είναι 500mA.
- GND: Σύνδεσμος γείωσης. Μνήμη Ο μικροελεγκτής ATmega328 διαθέτει μνήμη 32KB (όπου το 0.5 KB χρησιμοποιούνται από τον φορτωτή εκκίνησης (bootloader)).

Διαθέτει επίσης 2KB μνήμη SRAM και 1KB μνήμη EEPROM. Είσοδοι και Έξοδοι κάθε ένα από τις 14 συνδέσεις του Arduino μπορεί να λειτουργήσει σαν είσοδος ή σαν έξοδος κάνοντας χρήση της

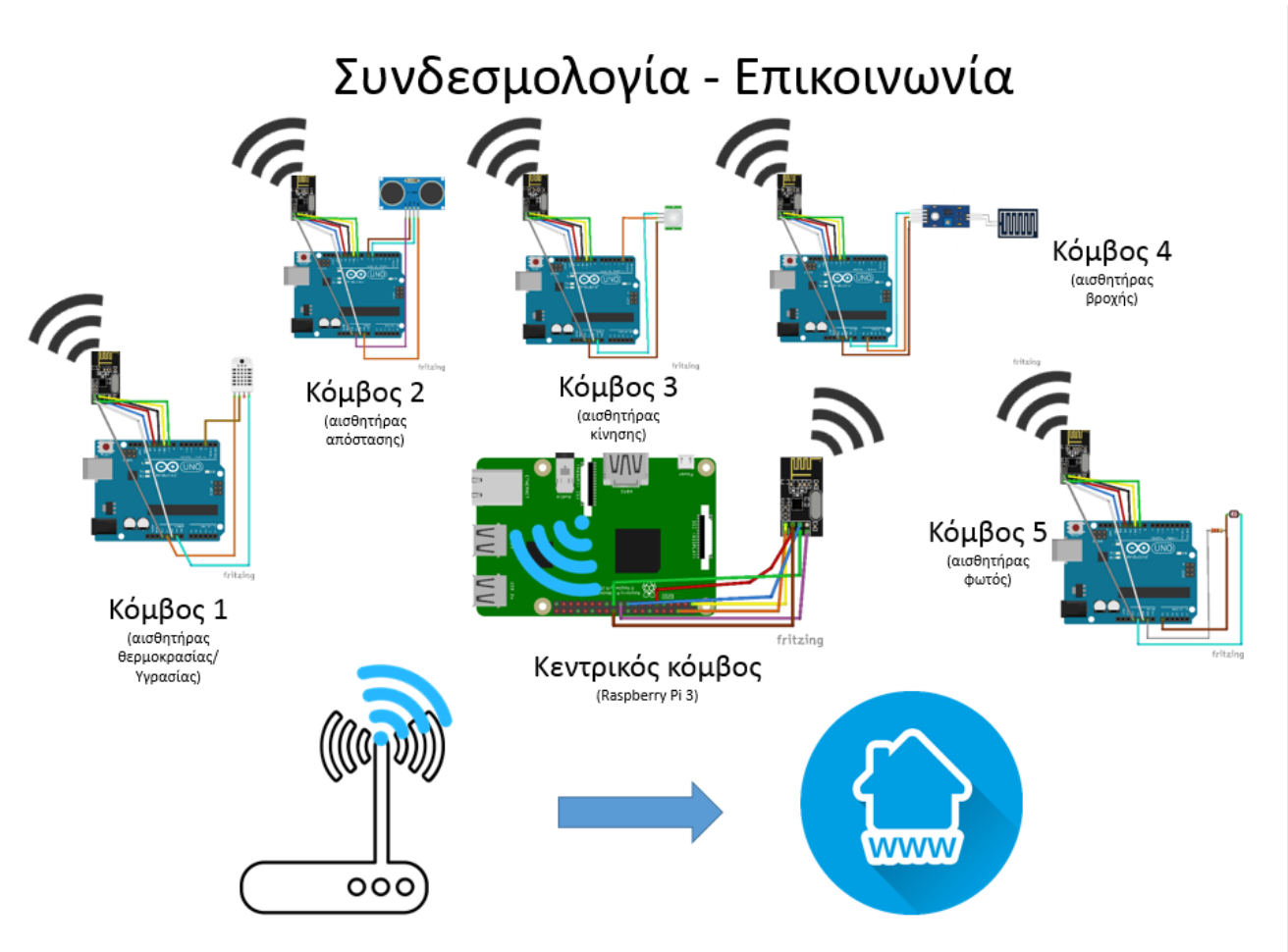
κατάλληλης συνάρτησης από την αντίστοιχη βιβλιοθήκη. Λειτουργούν στα 5V και μπορεί να προσφέρει ή να καταναλώσει το μέγιστο 40mA. Διαθέτουν, επίσης, εσωτερική αντίσταση που κυμαίνεται από 20-50kΩ. Μπορεί οι συνδέσεις του να λειτουργούν σαν γενικού σκοπού, παρόλα αυτά κάποια από αυτά έχουν επιπρόσθετες ιδιότητες.

- Σειριακή επικοινωνία (Serial): 0 (RX) και 1 (TX). Αυτοί οι ακροδέκτες μπορούν να χρησιμοποιηθούν για να αποστείλουν (TX) ή να λάβουν (RX) TTL σειριακά δεδομένα. Αυτά συνδέονται με τα αντίστοιχα του ATmega8U2.
- Εξωτερικές διακοπές (External Interrupts). Οι ακροδέκτες 2 και 3 μπορούν να χρησιμοποιηθούν έτσι ώστε να διαπιστωθεί μια διακοπή σε χαμηλή τιμή, θετική ή αρνητική ακμή ή μια αλλαγή σε μία τιμή, με την χρήση της κατάλληλης συνάρτησης του λογισμικού.
- PWM: Οι ακροδέκτες 3,5,6,9,10,11 έχουν τη δυνατότητα PWM εξόδου μεγέθους 8 bit με την χρήση της αντίστοιχης συνάρτησης.
- SPI: Οι ακροδέκτες 10, 11, 12, 13 υποστηρίζουν την SPI επικοινωνία, όπως θα αναλύσουμε εκτενέστερα παρακάτω.
- LED: Υπάρχει ενσωματωμένο led που υποστηρίζεται από τον ακροδέκτη 13. Όταν το pin μεταβαίνει σε κατάσταση HIGH, το led ανάβει, ενώ όταν μεταβεί σε κατάσταση LOW, σβήνει.
- TWI: Το A4 ή SDA ακροδέκτη και το A5 ή SCL ακροδέκτη υποστηρίζουν την TWI επικοινωνία χρησιμοποιώντας τη βιβλιοθήκη Wire. Η πλατφόρμα του UNO διαθέτει επίσης 6 αναλογικές εισόδους (A0 έως A5) η κάθε μία από αυτές έχει ανάλυση 10-bit, δηλαδή 1024 διαφορετικές τιμές. Προεπιλεγμένα, μπορούν να μετρήσουν τιμές από 0 έως 5 V. Μέσω όμως του ακροδέκτη AREF και της αντίστοιχης συνάρτησης, το οποίο ρυθμίζει την τάση αναφοράς, είναι δυνατόν να αλλάξει το πάνω όριο του εύρους αυτού. Επικοινωνία Το Uno διαθέτει μεγάλη ποικιλία δυνατοτήτων επικοινωνίας με έναν ηλεκτρονικό υπολογιστή, με άλλη πλατφόρμα Arduino ή κάποιον άλλον μικροελεγκτή. Ο Atmega328 προσφέρει UART σειριακή επικοινωνία μέσω των ακροδεκτών RX και TX όπως αναφέραμε προηγουμένως. Ο ATmega16U2 μετατρέπει την σειριακή επικοινωνία σε USB με αποτέλεσμα μία εικονική θύρα για το λογισμικό του υπολογιστή. Τα LED RX και TX στην πλακέτα του Arduino ανάβουν κάθε φορά που επικοινωνεί μέσω της θύρας USB.



## Κεφάλαιο 4. Υλοποίηση Δικτύου Αισθητήρων

### 4.1. Ανάλυση Προβλήματος



Εικόνα 29 - Συνδεσμολογία - Επικοινωνία

Σκοπός της πτυχιακής είναι να φτιαχτεί ένα portal (ηλεκτρονική πύλη) και ένα δίκτυο αισθητήρων το οποίο θα συλλέγει σε πραγματικό χρόνο δεδομένα από διάφορους αισθητήρες (π.χ. αισθητήρα βροχής, αισθητήρα φωτός, αισθητήρα κίνησης, αισθητήρα απόστασης και αισθητήρα θερμοκρασίας/υγρασίας) και θα τα παρουσιάζει σε μια ιστοσελίδα υπό μορφή τιμών και γραφημάτων που θα ανανεώνονται δυναμικά σε πραγματικό χρόνο. Κατόπιν συζήτησης με τον επιβλέποντα καθηγητή, η βιβλιοθήκη γραφημάτων η οποία θα χρησιμοποιηθεί είναι η grafana. Τα δεδομένα θα συλλέγονται από αισθητήρες, οι οποίοι θα είναι συνδεδεμένοι με πλακέτες arduino. Η μετάδοση των δεδομένων από τους αισθητήρες θα πραγματοποιείται μέσω του RF modules



nrf24l01. Ο web server (εξυπηρετητής) ο οποίος θα φιλοξενεί την ηλεκτρονική πύλη θα είναι εγκαταστημένος σε μια πλακέτα raspberry. Επιπλέον, η πλακέτα raspberry θα φιλοξενεί και τη βάση δεδομένων (mysql), όπου θα αποθηκεύονται οι τιμές των αισθητήρων.

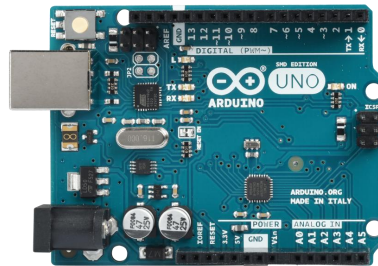
#### 4.1.1. Απαιτήσεις Συστήματος

Η ηλεκτρονική πύλη και η εφαρμογή θα πρέπει να είναι φιλική προς το χρήστη, να επιτρέπεται η διασύνδεση των δεδομένων, να είναι συμβατό με κινητές συσκευές, τα γραφήματα να υλοποιηθούν με το πλαίσιο εφαρμογής grafana και η μετάδοση των τιμών να επιτυγχάνεται με rf modules. Η ηλεκτρονική πύλη να δίνει τη δυνατότητα προβολής τόσο ιστορικών δεδομένων όσο και δεδομένων σε πραγματικό χρόνο.

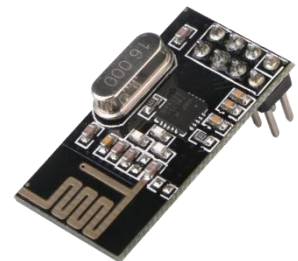
#### 4.2. Hardware που χρησιμοποιήθηκε



Raspberry Pi 3



5x Arduino UNO



6x nrf24l01



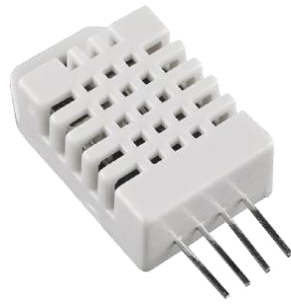
Αισθητήρας απόστασης



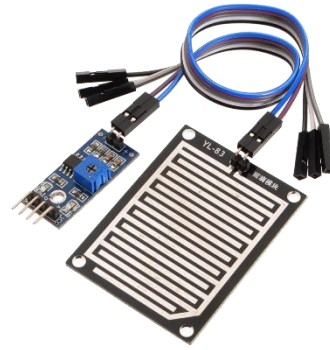
Αισθητήρας φωτός



Αισθητήρας κίνησης



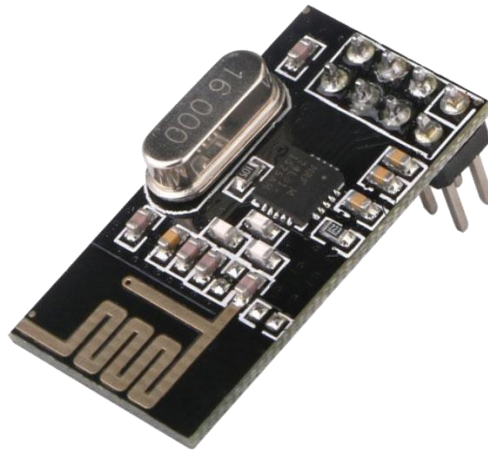
Αισθητήρας  
θερμοκρασίας/υγρασίας



Αισθητήρας βροχής

### 4.3. RF Modules NRF24L01 & NRF24L01+

Τα RF modules της εταιρείας Nordic semiconductors είναι RF συσκευές χαμηλής κατανάλωσης και μικρής εμβέλειας (περίπου 60 μέτρα). Συνδυάζουν RF πομποδέκτη, RF synthesizer και λειτουργούν στα 2.4 GHz. Υποστηρίζουν διασύνδεση SPI υψηλής ταχύτητας μετάδοσης και αποτελούν ένα πολύ αποδοτικό μέσο ασύρματης επικοινωνίας μεταξύ των μικροελεγκτών που χρησιμοποιήσαμε. Τα πιο σημαντικά χαρακτηριστικά τους, είναι η χαμηλή τους τιμή και η υψηλή τους απόδοση. Παρ' όλο που είναι αρκετά δύσκολα στον χειρισμό στην αρχή, η εξοικείωση με τις λειτουργίες τους είναι εύκολη. Για την κατασκευή αποκτήσαμε δυο διαφορετικά είδη modules, τα NRF24L01 και NRF24L01+. Εκτός από τη διαφορετική θέση των pins τάσης και γείωσης, τα modules αυτά λειτουργούν με τον ίδιο τρόπο.



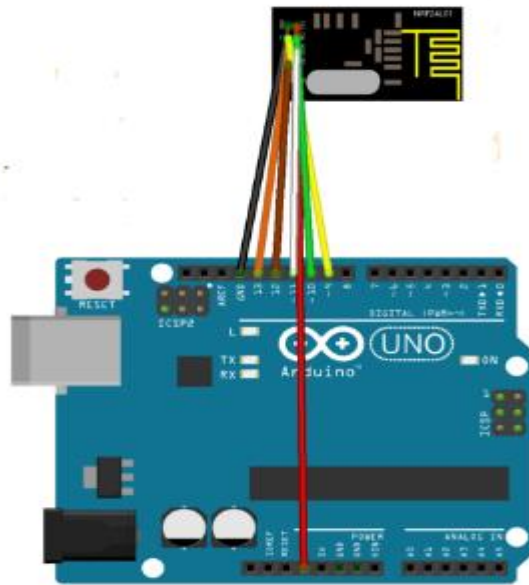
Εικόνα 30 - Τα modules NRF24L01 και L01+

Επικοινωνία με τον μικροελεγκτή Τα modules αυτά έχουν 8 ακροδέκτες. Το NRF24L01 έχει 10, αλλά, όπως φαίνεται στην παραπάνω εικόνα, έχει διπλά ground και Vcc pins. Οι υπόλοιποι ακροδέκτες είναι οι CE, IRQ και οι τέσσερις SPI ακροδέκτες (SCK, CSN, MISO, MOSI). Ο ακροδέκτης CE χρησιμοποιείται για να ελέγχει τη λήψη και αποστολή δεδομένων όταν το module βρίσκεται σε TX ή RX mode (transmit-receive mode). Ο ακροδέκτης IRQ χρησιμοποιείται για interrupts. Οι τέσσερις SPI ακροδέκτες είναι οι: SCK, MISO, MOSI και CSN. Η σύνδεση που προτείνεται για τα modules, σύμφωνα με διάφορους οδηγούς και κώδικες, είναι η παρακάτω. Με εξαίρεση τα SPI pins, τα οποία προφανώς συνδέονται στα προκαθορισμένα pins (10-13), ο ακροδέκτης CE συνδέεται στο pin 9. Όπως θα δούμε αργότερα βέβαια, η κατασκευή του κώδικα που υποστηρίζει

τη λειτουργία αυτών των μονάδων δε συγκεκριμενοποιεί (και συνεπώς δεν περιορίζει από άποψη επιλογής) τα pins στα οποία θα συνδεθούν οι ακροδέκτες CE και CSN.

#### 4.4. Arduino με NRF24L01

Συνδεσμολογία Arduino με το nrf24l01:



Εικόνα 31 - Συνδεσμολογία arduino – nrf24l01

##### 4.4.1. Κώδικας που χρησιμοποιήθηκε για τη σύνδεση arduino και nrf24l01:

Ο κώδικας για τον έλεγχο καλής λειτουργίας και την αποστολή μηνυμάτων είναι ο παρακάτω:

```
void setup()
{
  pinMode(8, OUTPUT);

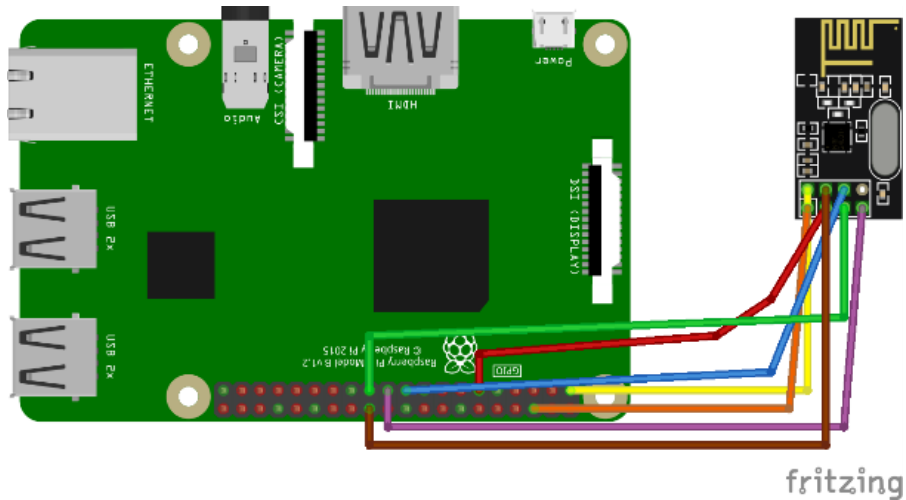
  delay(1000);

  radio.begin();
  radio.openReadingPipe(1,pipe);
  radio.startListening();;
}//--(end setup )---
```

```
void loop()  
{  
  if ( radio.available() )  
  {  
    bool done = false;  
    while (!done)  
    {  
      done = radio.read( received, sizeof(received) );  
  
      if (received[0]==0)  
        digitalWrite(8, LOW);  
      else  
        digitalWrite(8, HIGH);  
    }  
  }  
}
```

#### 4.5. Raspberry Pi 3 με NRF24L01

Συνδεσμολογία για το Raspberry Pi με το nrf24l01:



Εικόνα 32 - Συνδεσμολογία raspberry Pi 3 – nrf24l01

#### 4.5.1. Κώδικας που χρησιμοποιήθηκε για το Raspberry Pi 3, για να λαμβάνει τα δεδομένα και να τα καταχωρεί στη σχεσιακή βάση:

```
import RPi.GPIO as GPIO # eisagwgi vivliothikon
from lib_nrf24 import NRF24
import time
import spidev
import time
import os
import sys
sys.path.append('/usr/local/lib/python2.7/dist-packages/MYSQL_python-1.2.5-py2.7-
linux-armv7l.egg')
import pymysql

db = # stoixia vaseis dedomenwn pou tha kataxorountai ta dedomena twn aisthithirwn
pymysql.connect(host='localhost',port=3306,user='root',passwd='root',db='wsn')
cursor = db.cursor()

GPIO.setmode(GPIO.BCM)
# ekinisi kai rythmiseis nrf24l01
pipes = [[0xE8, 0xE8, 0xF0, 0xF0, 0xE1], [0xF0, 0xF0, 0xF0, 0xF0, 0xE1]]

radio = NRF24(GPIO, spidev.SpiDev())
radio.begin(0, 17)

radio.setPayloadSize(32)
radio.setChannel(0x76)
radio.setDataRate(NRF24.BR_1MBPS)
radio.setPALevel(NRF24.PA_MIN)

radio.setAutoAck(True)
radio.enableDynamicPayloads()
radio.enableAckPayload()

radio.openWritingPipe(pipes[0])
radio.openReadingPipe(1, pipes[1])
radio.printDetails()
# radio.startListening()

message = list("RADIO IS LISTENING")
# synthiki diavasmatos kai kataxwrish dedomenwn sth vash
while len(message) < 32:
    message.append(0)

while(1):
    start = time.time()
    radio.write(message)
    print("Sent the message: {}".format(message))
    radio.startListening()

    while not radio.available(0):
        time.sleep(1 / 100)
        if time.time() - start > 2:
            print("Timed out.")
            break

    receivedMessage = []
```

```
radio.read(receivedMessage, radio.getDynamicPayloadSize())
print("Received: {}".format(receivedMessage))

print("Translating the receivedMessage into unicode characters")
string = ""
for n in receivedMessage:
    # Decode into standard unicode set
    if (n >= 32 and n <= 126):
        string += chr(n)
print("Out received message decodes to: {}".format(string))
if string.find("WSN") > -1:
print "testing"
    sql = "insert into wsn (sensorval) values ('" + string + "'"
    print sql
    try:
        print sql
        cursor.execute(sql)
        db.commit()
    except Exception , e:
        print "exception"
        print ('DB exception : %s' %e)
time.sleep(5)
radio.stopListening()
time.sleep(1)
```



Εικόνα 33 – Συνδεσμολογία Raspberry PI

## **4.6. Nodes – Συνδεσμολογία - Κώδικας**

### **4.6.1. Node με αισθητήρα μέτρησης απόστασης**

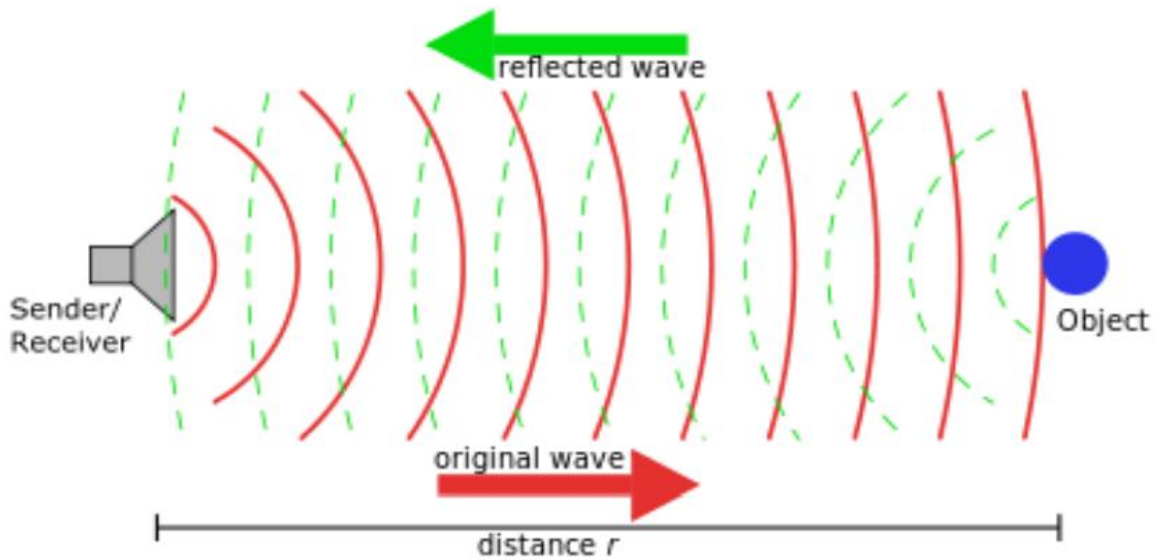
Υπέρηχος ονομάζεται εκείνο το μηχανικό κύμα, το οποίο έχει συχνότητα παραπάνω από 20 kHz και κατ' επέκταση δεν είναι δυνατό να γίνει αντιληπτό από το ανθρώπινο αυτί . Ωστόσο δεν έχει καμία άλλη διαφορά με τους κανονικούς ήχους στις ιδιότητές του. Οι συσκευές που εφαρμόζουν υπέρηχους χρησιμοποιούνται ως επί το πλείστον στον εντοπισμό αντικειμένων και στον υπολογισμό της απόστασης απ' αυτά. Επίσης, λειτουργούν από 20 kHz έως και κάποια GHz. Η επιστήμη του ήχου ξεκινάει απ' τα χρόνια του Πυθαγόρα, ο οποίος έκανε σημαντικές παρατηρήσεις σχετικά με το ύψος των ήχων, δηλαδή το 600 π.Χ.

Η βασική αρχή πίσω από κάθε συσκευή υπερήχων είναι η ίδια. Το μεταδιδόμενο σήμα είναι ένας παλμός υπερηχητικής ενέργειας, ο οποίος συνήθως παράγεται από κρύσταλλο, το οποίο δονείται εξαιτίας του πιεζοηλεκτρικού φαινομένου. Μετά από κάθε τέτοιο «ξέσπασμα» παλμών, η ηλεκτρονική συσκευή περιμένει το επιστρεφόμενο σήμα μέσα σ' ένα αποδεκτό χρονικό διάστημα. Ο ανακλώμενος παλμός επιστρέφει στον κρύσταλλο, τον οποίο και πάλι παράγοντας έτσι μία ηλεκτρική ώση, την οποία η κάθε συσκευή χρησιμοποιεί με τον τρόπο που είναι σχεδιασμένη. Στην φύση, διάφορα ζώα χρησιμοποιούν υπερήχους. Για παράδειγμα οι νυχτερίδες μπορούν να ανιχνεύσουν συχνότητες που φτάνουν μέχρι τα 200 kHz και με τον τρόπο αυτό βρίσκουν την πορεία τους ή και το «θύμα» τους. Επίσης, οι σκύλοι καθώς και πολλά έντομα μπορούν να ακούσουν υπερήχους.

Γι' αυτό και στο εμπόριο κυκλοφορούν πολλές συσκευές υπερήχων, οι οποίες διαφημίζονται ότι μπορούν να διώξουν μακριά διάφορους τύπους ζώων, αλλά δεν έχει αποδειχθεί επιστημονικά η χρήση τους. Την ίδια ώρα και ο άνθρωπος χρησιμοποιεί τους υπέρηχους σε πολλά πεδία. Κύριος κλάδος χρήσης υπερήχων είναι η Ιατρική, όπου χρησιμοποιούνται κυρίως για την απεικόνιση των εσωτερικών οργάνων του ανθρώπινου σώματος. Η πιο γνωστή και σημαντική ίσως εφαρμογή τους είναι η απεικόνιση του εμβρύου στις έγκυες γυναίκες. Επιπλέον, οι υπέρηχοι χρησιμοποιούνται ως διαγνωστική μέθοδος, για την ανίχνευση όγκων και οιδημάτων, καθώς και για εξωσωματική λιθοτριψία, αλλά και για τις ανάγκες της οδοντιατρικής επιστήμης. Άλλες εφαρμογές είναι οι συγκολλήσεις, οι καθαρισμοί και οι μη καταστροφικοί έλεγχοι, που χρησιμοποιούνται στις



βιομηχανίες κυρίως για την ανίχνευση ρωγμών. Απ' τις πιο διαδεδομένες χρήσεις υπερήχων είναι το σόναρ, το οποίο είναι συσκευή εντοπισμού αντικειμένων υποβρυχίως και χρησιμοποιείται κυρίως από τα πλοία του Πολεμικού Ναυτικού. Το σόναρ, χρησιμοποιεί την βασική αρχή των υπερήχων η οποία παρουσιάστηκε παραπάνω και φαίνεται στην εικόνα παρακάτω. Στη συνέχεια για να υπολογίσουμε την απόσταση απ' το αντικείμενο που ανιχνεύτηκε θα πρέπει να ξέρουμε τον χρόνο που έκανε το κύμα, καθώς και την ταχύτητα του ήχου στο συγκεκριμένο μέσο (νερό στην προκειμένη περίπτωση) που εξαρτάται κι απ' την θερμοκρασία.



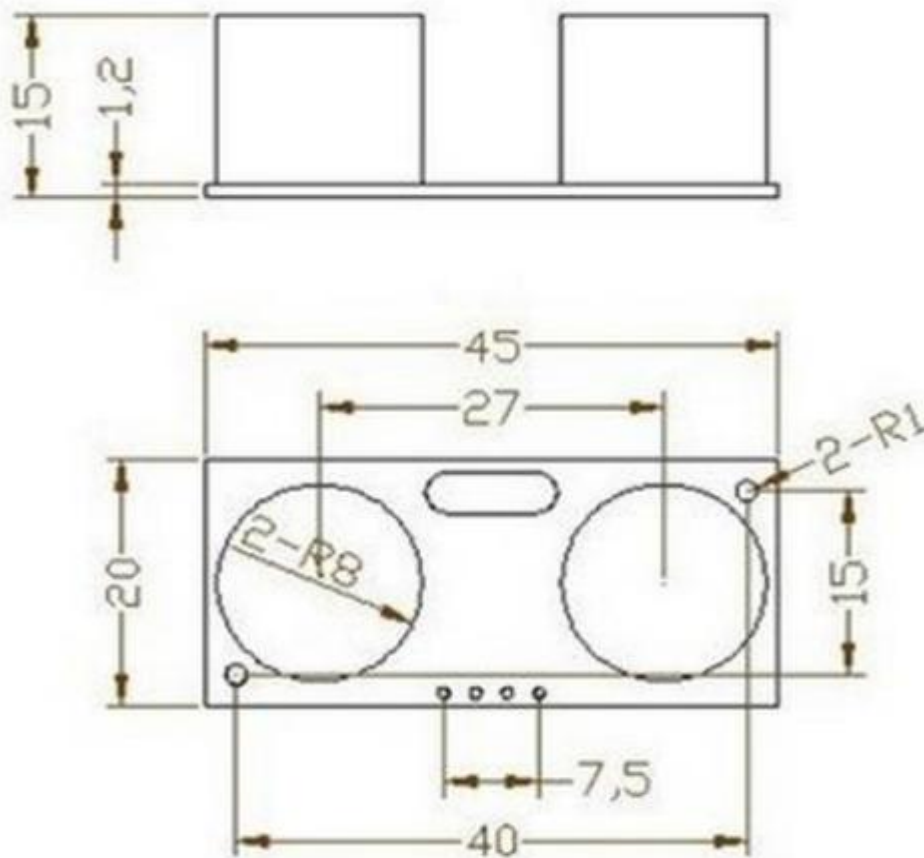
Εικόνα 34 - Αρχή των Sonar



Εικόνα 35 - HC-SR04

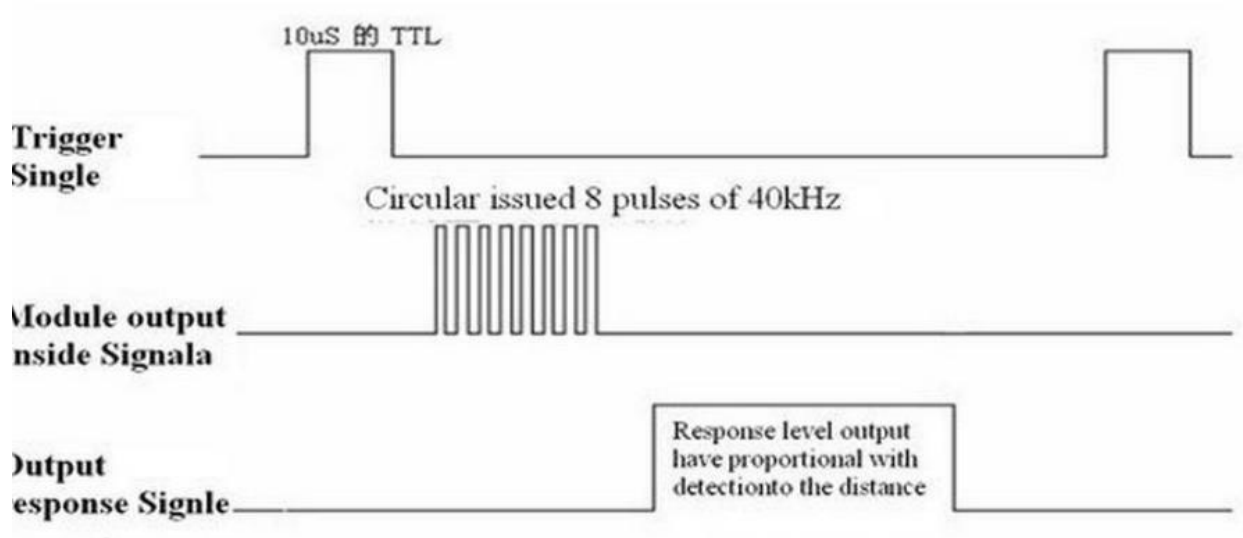
Η συσκευή υπερήχων που θα χρησιμοποιήσουμε στην κατασκευή μας είναι η HC - SR04 η οποία διαθέτει τέσσερις ακροδέκτες. Οι ακροδέκτες είναι οι εξής : ground (GND), Echo Pulse Output (ECHO), Trigger Pulse Input (TRIG), 5V Supply (Vcc). Δίνουμε τάση στη συσκευή χρησιμοποιώντας τον ακροδέκτη Vcc και το γειώνουμε με το GND. Με την βοήθεια του Raspberry Pi στέλνουμε ένα σήμα στον ακροδέκτη TRIG, το οποίο με την σειρά του δημιουργεί μία σειρά παλμών υπερήχων απ' την συσκευή. Τα κύματα αυτά προσπίπτουν πάνω στα αντικείμενα που υπάρχουν σχετικά κοντά στη συσκευή και αντανακλώνται. Μ' αυτόν τον τρόπο τα ανακλώμενα κύματα γυρνούν στην συσκευή μας και μόλις τα ανιχνεύσει ο αισθητήρας μας, στέλνει ένα σήμα τάσεως 5V απ' τον ακροδέκτη ECHO πίσω στην πλακέτα μας.

Ο συγκεκριμένος ακροδέκτης θα παραμείνει «χαμηλά», δηλαδή δεν θα δίνει τάση, μέχρι ο αισθητήρας να ανιχνεύσει το ανακλώμενο κύμα.



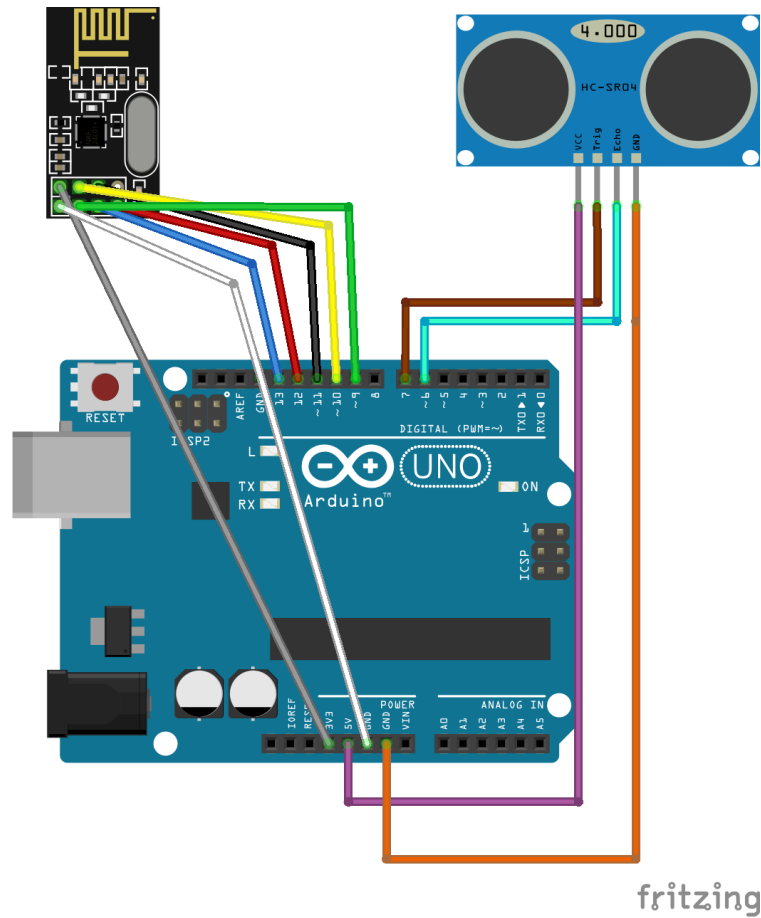
Εικόνα 36 - Διαστάσεις Συσκευής HC-SR04

Ο αισθητήρας, τον οποίο χρησιμοποιούμε απαιτεί έναν παλμό εισόδου διάρκειας 10 $\mu$ s, που προκαλεί με την σειρά του την εκπομπή 8 παλμών στα 40kHz συχνότητας ο καθένας.



Εικόνα 37 - Διαγράμματα Χρονισμού HC-SR04

#### 4.6.2. Συνδεσμολογία για το node του αισθητήρα απόστασης:



Εικόνα 38 – Σύνδεση Arduino-αισθητήρα απόστασης

### 4.6.3. Κώδικας που χρησιμοποιήθηκε για τον αισθητήρα απόστασης:

```
#include <nRF24L01.h> //eisagogi aparaititon vivliothikon
#include <printf.h>
#include <RF24.h>
#include <RF24_config.h>
#include<SPI.h>
#include<RF24.h>

RF24 radio(9, 10);
#define trigPin 7 //arxikopoihsh twv pins pou tha xrisimopoihthoun
#define echoPin 6

void setup(void) {
    radio.begin(); //ekkinisi tou nrf24l01 kai rythmiseis tou
    radio.setPALevel(RF24_PA_MAX);
    radio.setChannel(0x76);
    radio.openWritingPipe(0xF0F0F0F0E1LL);
    radio.enableDynamicPayloads();
    radio.powerUp();

    Serial.begin(9600);
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);

    const char text[] = "WSN HSR04 SENSOR STARTED"; //dilwsi xaraktirwn
    radio.write(&text, sizeof(text));
}

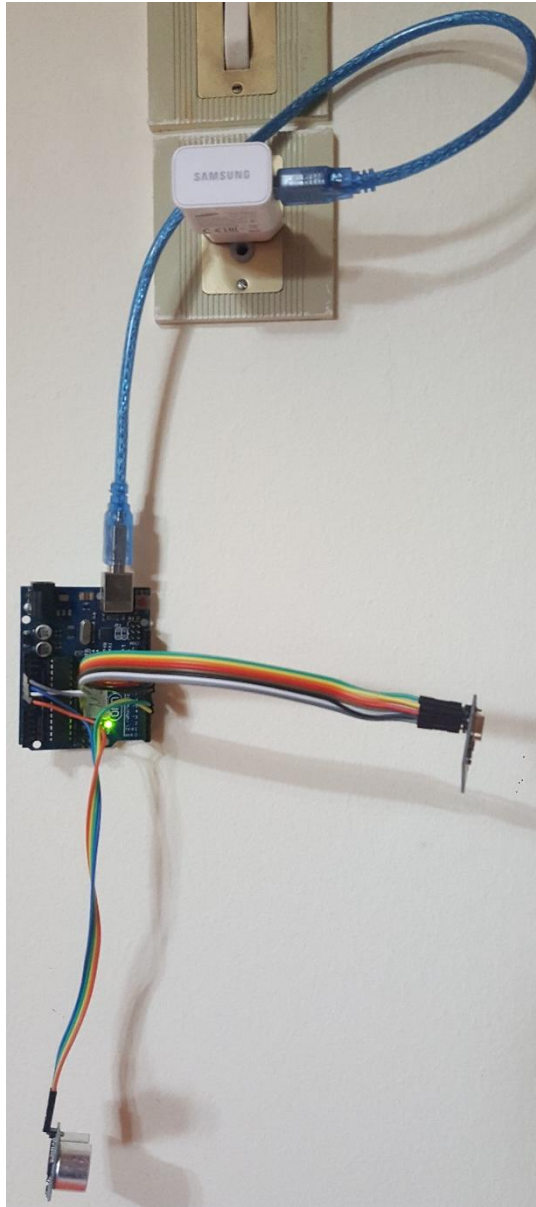
void loop(void) {

    delay(1000); //kathisterisi enos second

    float duration, distance; //dilwsi timwn diarkeia kai aostasi
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH); //arxikopoihsh timvvn high kai low
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    duration = pulseIn(echoPin, HIGH);
    distance = (duration/2) / 29.1; // ypologismos apostasis
    if (distance < 4) { //synartisi if else gia ta oria metrisis tou aisthitira
    }
    else {

    }
    if (distance >= 200 || distance <= 0) {
        Serial.println("Out of range");
        const char text[] = "WSN OUT OF RANGE"; //an o aisthitiras einai ekstos orion
        metrisis tipwnei "WSN OUT OF RANGE"
        radio.write(&text, sizeof(text)); //edw stelnete h timi tis metrisis mesw tou
        nrf24l01
    }
}
```

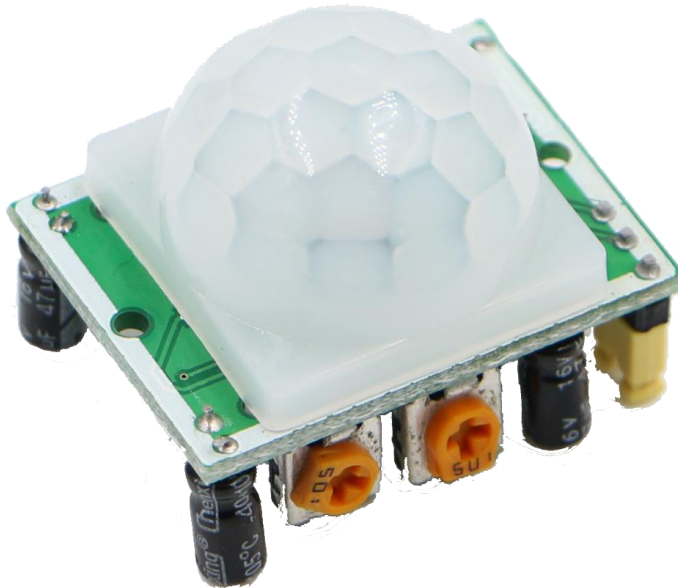
```
else {
    char dist[10];
    dtostrf(distance, 4, 2, dist); //parakatw typwnetai sth seiriaki othoni h
    apostasy kai kapoio synodeutiko keimeno.
    char text[28] = "WSN DISTANCE IS : ";
    text[19] = dist[0];
    text[20] = dist[1];
    text[21] = dist[2];
    text[22] = dist[3];
    text[24] = dist[4];
    text[25] = ' ';
    text[26] = 'c';
    text[27] = 'm';
    radio.write(&text, sizeof(text));
    Serial.print(distance);
    Serial.println(" cm");
}
delay(5000); //kathisterisi 5 seconds mexri thn epomenh metrish
}
```



Εικόνα 39 - Arduino - Αισθητήρας απόστασης

#### 4.7. Αισθητήρας Ανίχνευσης Κίνησης

Ο αισθητήρας έχει την ικανότητα αναγνώρισης διαταράξεων στο περιβάλλον γύρω του μέσω εκπομπής χαμηλής έντασης ηλεκτρομαγνητικών κυμάτων. Για τις ανάγκες της διπλωματικής εργασίας έχει χρησιμοποιηθεί ο αισθητήρας HC-SR501 PIR Motion Detector.

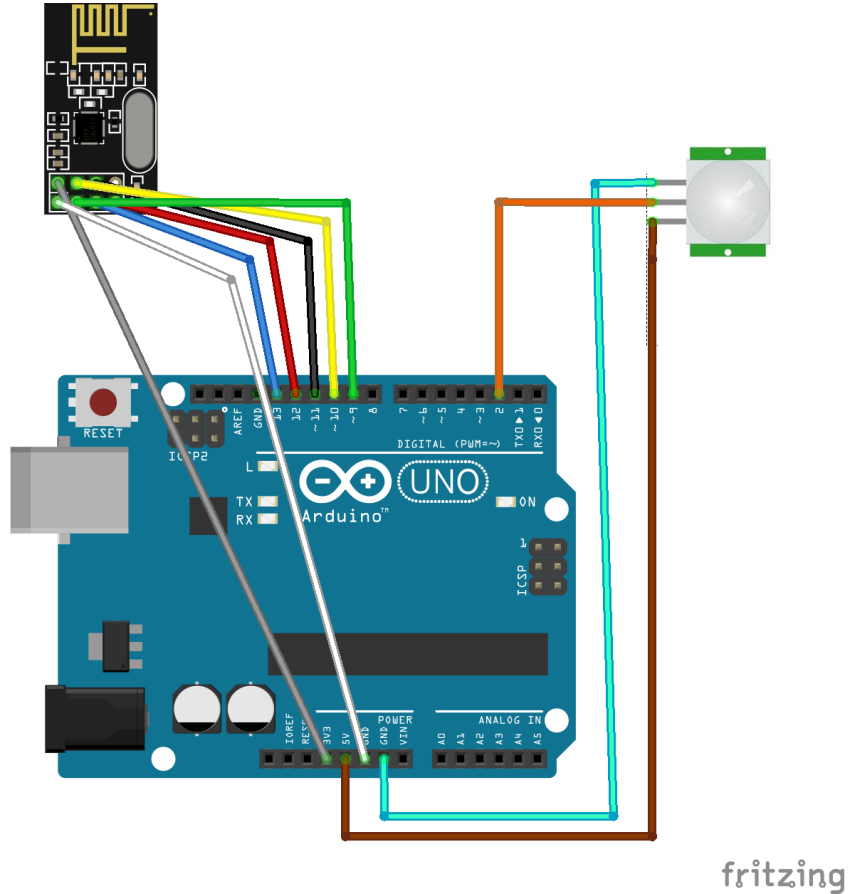


Εικόνα 40 - HC-SR501 PIR Motion Detector

Όπως φαίνεται κι από την παραπάνω εικόνα, κάποιες ρυθμίσεις του αισθητήρα γίνονται απευθείας στο ηλεκτρονικό κομμάτι. Υπάρχει δυνατότητα καθορισμού της «ευαισθησίας» του αισθητήρα, δηλαδή στα πόσα μέτρα να γίνεται η ανίχνευση κίνησης και κυμαίνεται από 3 – 7 μέτρα. Επίσης, υπάρχει δυνατότητα καθορισμού διάρκειας της αναγνώρισης κίνησης, αποτρέποντας την αναγνώριση δεύτερης κίνησης πριν το πέρας του χρόνου. Ο ελάχιστος χρόνος διάρκειας είναι 3s και ο μέγιστος 5 λεπτά. Τέλος μπορούμε να καθορίσουμε κάθε πότε γίνεται επανεκκίνηση του ενσωματωμένου χρονομέτρου καθυστέρησης, δηλαδή μετά από ανίχνευση κίνησης ή μετά το πέρας του χρόνου που διαρκεί η κίνηση.



#### 4.7.1. Συνδεσμολογία για το node του αισθητήρα κίνησης:



Εικόνα 41 - Σύνδεση Arduino - HC-SR501

#### 4.7.2. Κώδικας που χρησιμοποιήθηκε για τον αισθητήρα κίνησης:

```
#include <nRF24L01.h> //eisagogi aparaititon vivliothikon
#include <printf.h>
#include <RF24.h>
#include <RF24_config.h>

#include<SPI.h>
#include<RF24.h>

RF24 radio(9, 10);
//dilwsi timwn
int calibrationTime = 30;

long unsigned int lowIn;

long unsigned int pause = 5000;
```

```

boolean lockLow = true;
boolean takeLowTime;

int pirPin = 7; //dilwsi twv pins pou tha xrisimopoihthoun
int ledPin = 8;
int buzPin = 6;

void setup(void) {
  radio.begin(); //ekkinisi tou nrf24l01 kai rythmiseis tou
  radio.setPALevel(RF24_PA_MAX);
  radio.setChannel(0x76);
  radio.openWritingPipe(0xF0F0F0E1LL);
  radio.enableDynamicPayloads();
  radio.powerUp();

  Serial.begin(9600);
  pinMode(pirPin, INPUT);
  digitalWrite(pirPin, LOW);
  pinMode(buzPin, OUTPUT);
  //dilwv pinaka xaraktirwn me minima
  const char text[] = "WSN PIR SENSOR CALIBRATING";
  radio.write(&text, sizeof(text));
  //Kalimprarisma aisthitira kinisis
  Serial.print("calibrating sensor ");
  for(int i = 0; i < calibrationTime; i++){
    Serial.print(".");
    delay(1000);
  }
  Serial.println(" done");
  Serial.println("SENSOR ACTIVE");
  const char text1[] = "WSN PIR SENSOR ACTIVE";
  radio.write(&text1, sizeof(text1));
  delay(50);
}

void loop(void) {
  //synartisi if me thn opoia diavazetai h timi tou aisthitira kai typwnetai to
  antistoixο minima gia to an yparxei kinisi h oxι
  delay(1000);

  if(digitalRead(pirPin) == HIGH) {

    digitalWrite(buzPin, HIGH);
    delay(100);

    digitalWrite(buzPin, LOW);
    delay(100);

    digitalWrite(buzPin, HIGH);
    delay(100);

    if(lockLow) {
      lockLow = false;
      Serial.println("---");
      Serial.print("WSN Motion Detected");
    }
  }
}

```

```
    const char text2[] = "Motion Detected";
    radio.write(&text2, sizeof(text2));
    Serial.print(millis()/1000);
    Serial.println(" sec");
    delay(50);
  }
  takeLowTime = true;
}

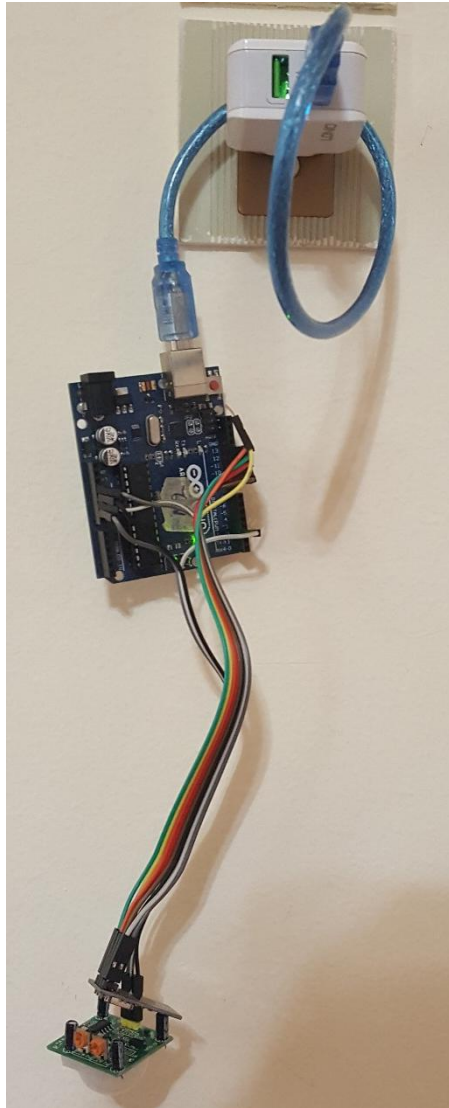
if(digitalRead(pirPin) == LOW){

  digitalWrite(buzPin, LOW);

  if(takeLowTime){
    lowIn = millis();
    takeLowTime = false;
  }

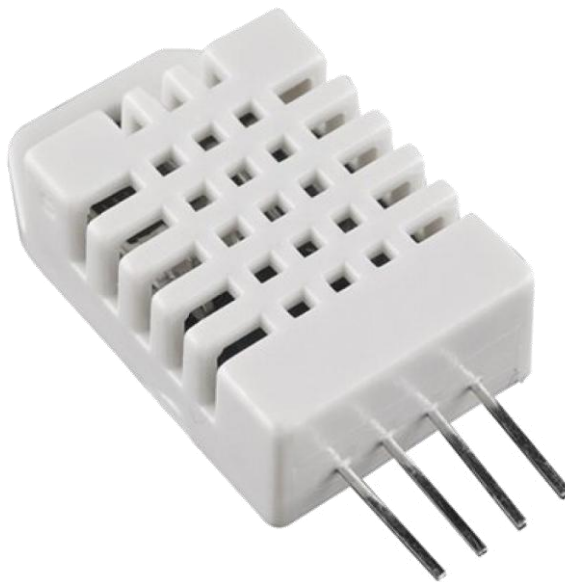
  if(!lockLow && millis() - lowIn > pause){

    lockLow = true;
    Serial.print("Motion Stopped");
    const char text3[] = "WSN Motion Stopped";
    radio.write(&text3, sizeof(text3));
    Serial.print((millis() - pause)/1000);
    Serial.println(" sec");
    delay(50);
  }
}
}
```



Εικόνα 42 - Arduino – Αισθητήρας κίνησης

#### **4.8. Αισθητήρας Θερμοκρασίας/Υγρασίας**



Εικόνα 43 - DHT

Ο αισθητήρας DHT είναι ένας φθηνός και αξιόπιστος αισθητήρας. Χρησιμοποιεί έναν πυκνωτή για την υγρασία και ένα θερμίστορ για να μετρήσει τον περιβάλλον αέρα. Όλα αυτά συνδέονται με έναν υψηλής επίδοσης 8-bit μικροελεγκτή που προσφέρει πολύ καλή ποιότητα και ταχύτητα στα αποτελέσματα.

Γενικά, ο αισθητήρας είναι μικρός σε μέγεθος, έχει αρκετά χαμηλή κατανάλωση ισχύος και είναι αρκετά ακριβής στις μετρήσεις. Έχει τρεις ακροδέκτες, ένα για την τροφοδοσία, ένα για την γείωση και ένα για τα δεδομένα.

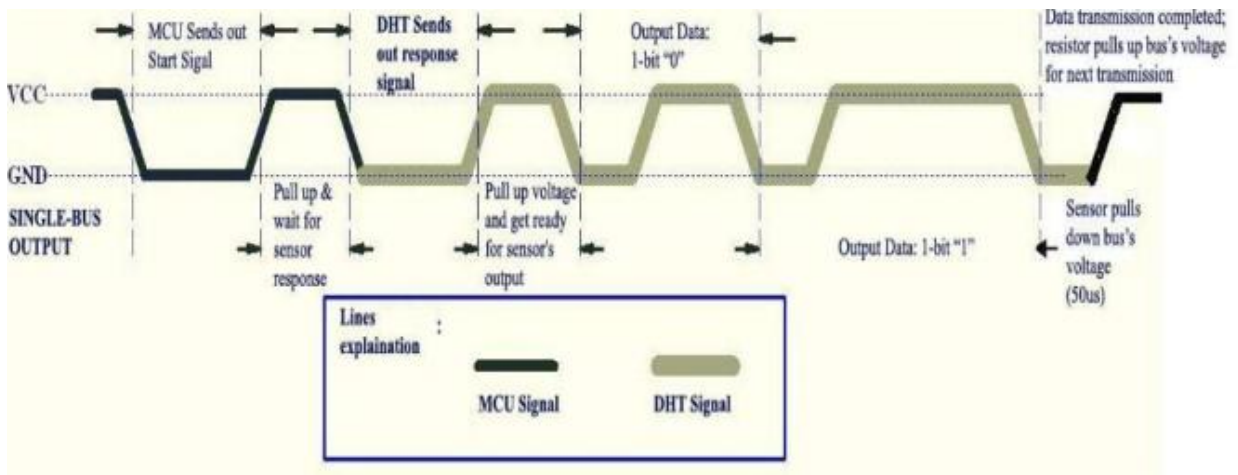
**Τεχνικά Χαρακτηριστικά:**

	Υγρασία	Θερμοκρασία
Τάση	3-5.5Volt	
Έξοδος	Ψηφιακή	
Αισθητήρας	Resistive-Type Humidity	NTC temperature component
Εύρος	20-80% Rh	0-50°C
Ακρίβεια	5%	±2°C
Δειγματοληψία	1Hz	

Η διαδικασία της επικοινωνίας ξεκινάει όταν το Arduino στέλνει πρώτο σήμα, ο αισθητήρας μεταβαίνει από την χαμηλής-κατανάλωσης λειτουργία σε κανονική λειτουργία περιμένοντας το Arduino να τελειώσει με την αποστολή. Μόλις αυτό πραγματοποιηθεί ο αισθητήρας στέλνει πίσω στο Arduino ένα σήμα δεδομένων απάντησης των 40bit που περιλαμβάνουν τις ανάλογες πληροφορίες για την υγρασία και την θερμοκρασία. Συγκεκριμένα:

- 8 bit για την τιμή της υγρασίας
- 8 bit για την αέραια τιμή της υγρασίας
- 8 bit για την τιμή της θερμοκρασίας
- 8 bit για την αέραια τιμή της θερμοκρασίας
- 8 bit για το bit ελέγχου ισοτιμίας των δεδομένων.

Χωρίς το αρχικό σήμα από το Arduino ο αισθητήρας δεν στέλνει δεδομένα. Όταν τα δεδομένα σταλούν από τον αισθητήρα, αυτός μεταβαίνει και πάλι στην χαμηλής - κατανάλωσης λειτουργία και περιμένει νέο σήμα.



Εικόνα 44 - Διαδικασία επικοινωνίας Arduino-DHT





```

// ce, csn pins
RF24 radio(9, 10);
//dilwsi timwn
char * floatToString(char * outstr, double val, byte precision, byte widthp, int
length){
    char temp[length];
    byte i;
//ipologismos tou syntelsti stroggilopoihsis kai tou klasmatikou pollaplasiasiati
    double roundingFactor = 0.5;
    unsigned long mult = 1;
    for (i = 0; i < precision; i++)
    {
        roundingFactor /= 10.0;
        mult *= 10;
    }

    temp[0]='\0';
    outstr[0]='\0';

    if(val < 0.0){
        strcpy(outstr,"-\0");
        val = -val;
    }

    val += roundingFactor;

    strcat(outstr, itoa(int(val),temp,10));
    if( precision > 0) {
        strcat(outstr, ".\0");
        unsigned long frac;
        unsigned long mult = 1;
        byte padding = precision -1;
        while(precision--)
            mult *=10;

        if(val >= 0)
            frac = (val - int(val)) * mult;
        else
            frac = (int(val)- val ) * mult;
        unsigned long frac1 = frac;

        while(frac1 /= 10)
            padding--;

        while(padding--)
            strcat(outstr,"0\0");

        strcat(outstr,itoa(frac,temp,10));
    }

    if ((widthp != 0)&&(widthp >= strlen(outstr))){
        byte J=0;
        J = widthp - strlen(outstr);

        for (i=0; i< J; i++) {
            temp[i] = ' ';

```

```

    }

    temp[i++] = '\0';
    strcat(temp, outstr);
    strcpy(outstr, temp);
}

return outstr;
}

void setup() {
    radio.begin(); //ekkinisi tou nrf24l01 kai rythmiseis tou
    radio.setPALevel(RF24_PA_MAX);
    radio.setChannel(0x76);
    radio.openWritingPipe(0xF0F0F0F0E1LL);
    radio.enableDynamicPayloads();
    radio.powerUp();

    Serial.begin(9600);

    Serial.println("DHTxx test!");
    dht.begin();
    Serial.println("DHTxx test!");
    const char text[] = "WSN DTH SENSOR STARTED";
    radio.write(&text, sizeof(text));
    Serial.println("DHTxx test!");
}

void loop() {
    //anamoni 1 second anamesa stis metriseis
    delay(1000);

    //diavasma igrasias
    float h = dht.readHumidity();
    //diavasma thermokrasias se vathoum kelsiou
    float t = dht.readTemperature();
    //diavasma thermokrasias se vathmous Farenait
    float f = dht.readTemperature(true);
    //elegxos gia sfalma diavasmatos timwn, an yparxei sfalma xanaprospathei
    if (isnan(h) || isnan(t) || isnan(f)) {
        Serial.println("Failed to read from DHT sensor!");
        return;
    }
    //ypologismos tou heat index se Farenait
    float hif = dht.computeHeatIndex(f, h);
    //Ypologisos tou heat index se Kelsiou
    float hic = dht.computeHeatIndex(t, h, false);
    //ektypwsi twn timwn pou ipologistikan sthn seiriaki othoni kai apostoli tous
    mesw nrf24l01
    Serial.print("Humidity: ");
    Serial.print(h);
    Serial.print(" %\t");
    Serial.print("Temperature: ");
    Serial.print(t);
    Serial.print(" *C ");
    Serial.print(f);
}

```

```

Serial.print(" *F\t");
Serial.print("Heat index: ");
Serial.print(hic);
Serial.print(" *C ");
Serial.print(hif);
Serial.println(" *F");

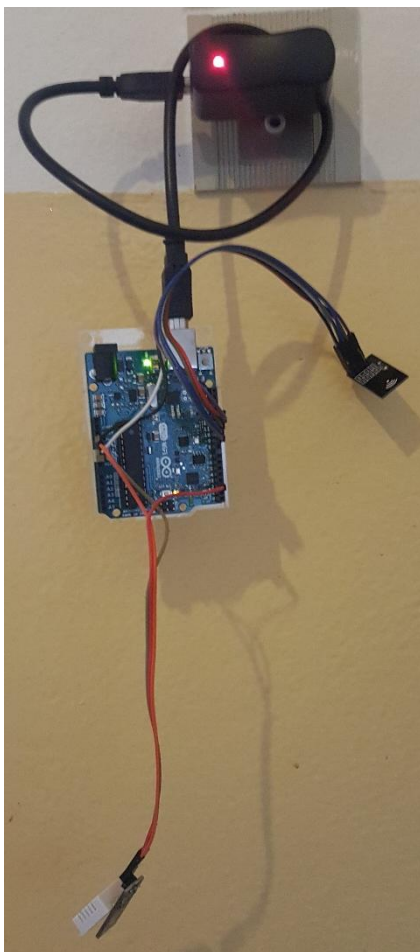
char hum[10];

dtostrf(h, 4, 2, hum);
char text[28] = "WSN HUMIDITY IS : ";
text[19] = hum[0];
text[20] = hum[1];
text[21] = hum[2];
text[22] = hum[3];
text[24] = hum[4];
text[25] = ' ';
text[26] = '%';

radio.write(&text, sizeof(text));
delay(1000);
char temp[10];

dtostrf(t, 4, 2, temp);
char text1[30] = "WSN TEMPERATURE IS : ";
text1[22] = temp[0];
text1[23] = temp[1];
text1[24] = temp[2];
text1[25] = temp[3];
text1[26] = temp[4];
text1[27] = ' ';
text1[28] = '*';
text1[29] = 'C';
radio.write(&text1, sizeof(text1));
delay(1000);
char heatidx[10];
dtostrf(hic, 4, 2, heatidx);
char text2[30] = "WSN HEAT INDEX IS : ";
text2[21] = heatidx[0];
text2[22] = heatidx[1];
text2[23] = heatidx[2];
text2[24] = heatidx[3];
text2[25] = heatidx[4];
text2[26] = ' ';
text2[27] = '*';
text2[28] = 'C';
radio.write(&text2, sizeof(text2));
delay(1000); //kathisterisi 1 second.
}

```

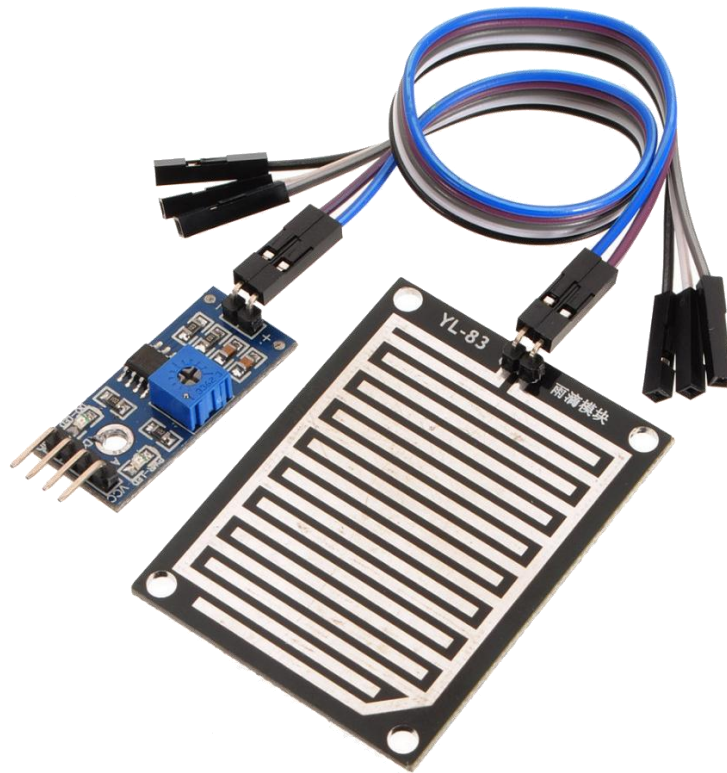


Εικόνα 46 - Arduino - Αισθητήρας θερμοκρασίας

#### ***4.9. Αισθητήρας Βροχής***

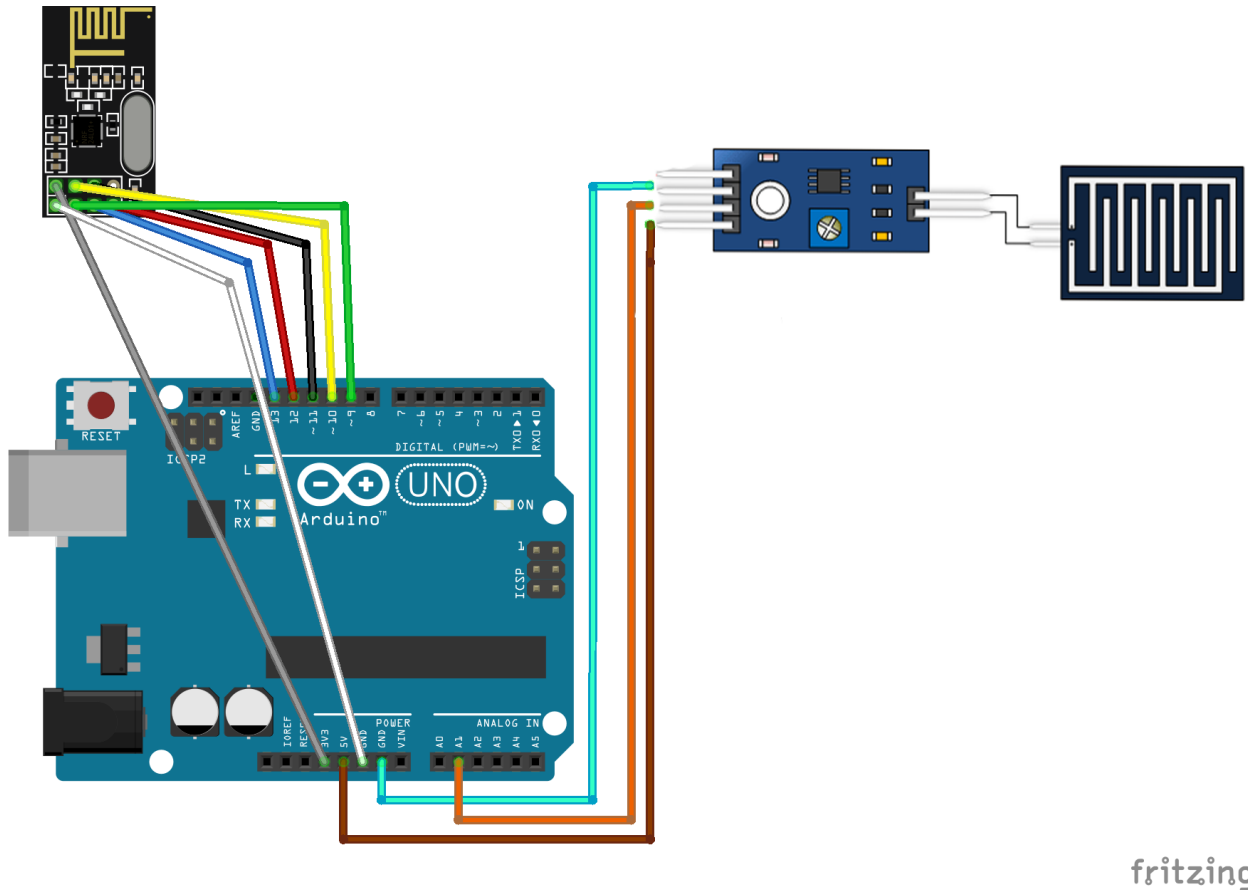
Το αισθητήριο βροχής, τροφοδοτείται με 5V και ουσιαστικά είναι μια μεταβλητή αντίσταση η οποία μεταβάλλεται με την ύπαρξη υγρασίας ή σταγόνων πάνω στην ειδικά διαμορφωμένη επιφάνειά του, δίνοντας μας δύο εξόδους (Vout): μια αναλογική (A0) και μία ψηφιακή (D0).

Για την αναγνώριση της κατάστασης του καιρού, πέραν της φωτεινότητας είναι απαραίτητο να χρησιμοποιηθεί και ένας αισθητήρας βροχής που ουσιαστικά εντοπίζει αν υφίσταται και πόσον δυνατή είναι. Ο αισθητήρας που χρησιμοποιήθηκε φαίνεται στην παρακάτω εικόνα.



Εικόνα 47 - Αισθητήρας Βροχής

#### 4.9.1. Συνδεσμολογία για το node του αισθητήρα βροχής:



Εικόνα 48 - Συνδεσμολογία arduino - αισθητήρα βροχής

#### 4.9.2. Κώδικας που χρησιμοποιήθηκε για τον αισθητήρα βροχής:

```
#include <nRF24L01.h> //eisagogi aparaititon vivliothikon
#include <printf.h>
#include <RF24.h>
#include <RF24_config.h>
#include <SPI.h>
#include <RF24.h>

RF24 radio(9, 10); //arxiopoihsh tw'n pins kai tw'n timvn pou tha xrisimopoihthoun
int rainPin = A0;
int thresholdValue = 500; //to orio tis timis tou aisthitira pou exoume thesei

void setup(void) {
    radio.begin(); //ekkinisi tou nrf24l01 kai rythmiseis tou
    radio.setPALevel(RF24_PA_MAX);
}
```

```

radio.setChannel(0x76);
radio.openWritingPipe(0xF0F0F0F0E1LL);
radio.enableDynamicPayloads();
radio.powerUp();
pinMode(rainPin, INPUT);
Serial.begin(9600);
const char text[] = "WSN RAIN SENSOR STARTED";//dilwsi xaraktirwn
radio.write(&text, sizeof(text));

}

void loop(void){

    delay(1000); //kathisterisi enos second

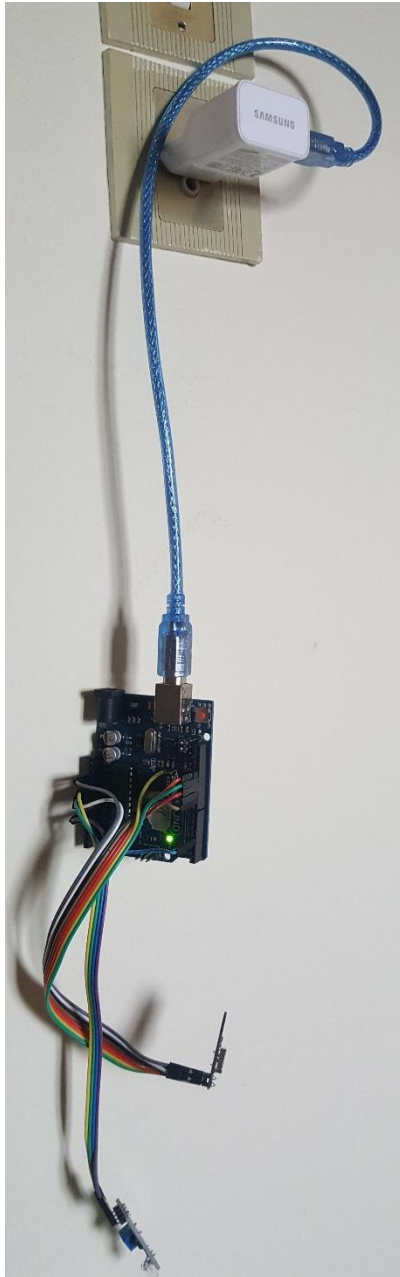
    //diavasma tou aisthitira sthn analog A0
    int sensorReading = analogRead(A0);
    Serial.print(sensorReading);
    const char text1[] = "WSN ITS WET"; //arxikopoihsh pinakwn metavlitwn me
    minima pou theloume na emfanisoume
    const char text2[] = "WSN ITS DRY";

    if(sensorReading < thresholdValue){ //synartisi if pou elegxei to orio pou
    exoume thesei kai analogws typwnei an o aisthotiras einai stegnos h ygros kai
    typwneo to antistoixo minima
        Serial.println("WSN ITS WET");
        radio.write(&text1, sizeof(text1));
    }
    else {
        Serial.println("WSN ITS DRY");
        radio.write(&text2, sizeof(text2));
    }

    delay(15000); //kathisterisi 15 seconds

}

```



Εικόνα 49 - Arduino - Αισθητήρας βροχής



#### 4.10. Αισθητήρας φωτός



Εικόνα 50 – Αισθητήρας φωτός

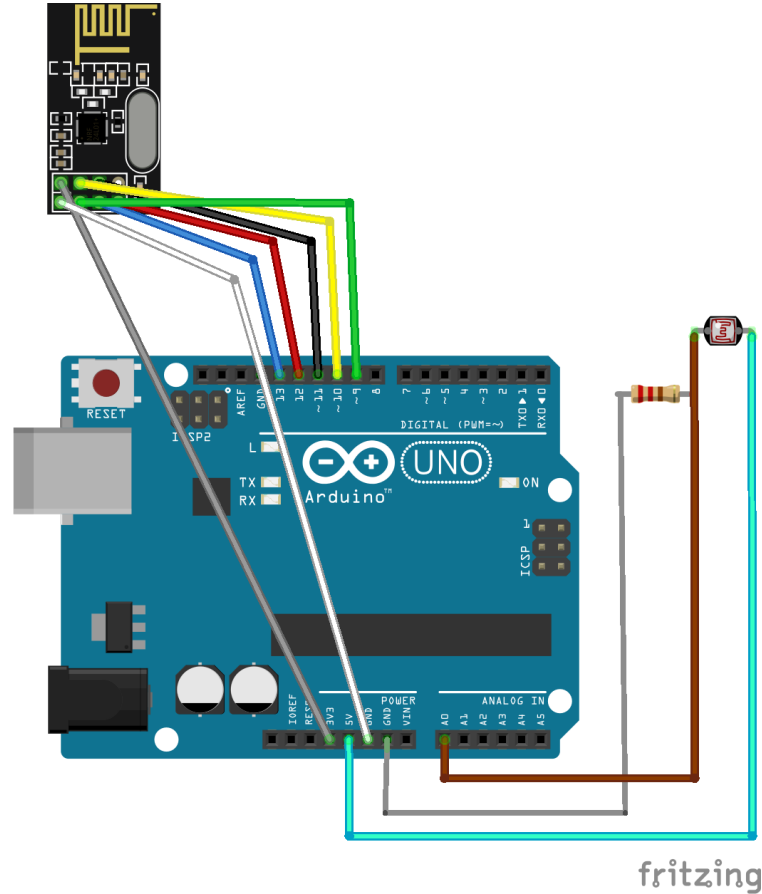
Ο αισθητήρας φωτός αποτελείται από μια φωτοδίοδο και μια ηλεκτρονική διάταξη με LM393. Ενώ ενσωματώνει ποτενσιόμετρο για την ρύθμιση της ευαισθησίας του.

Χρησιμοποιούμε την ψηφιακή έξοδο του αισθητήρα η οποία λειτουργεί ως εξής. Όταν η στάθμη του φωτός στο περιβάλλον είναι κάτω από το όριο που έχει τεθεί, τότε η ψηφιακή έξοδος του αισθητήρα δίνει υψηλή τιμή 1, ενώ όταν η στάθμη του φωτός στο περιβάλλον είναι πάνω από το όριο που έχει τεθεί η ψηφιακή έξοδος του αισθητήρα δίνει χαμηλή τιμή 0.

Η τάση λειτουργίας του κυμαίνεται από 3,3V έως 5V.

Η συνδεσιμότητα του γίνεται με 3 καλώδια pin 1 A0 αναλογική έξοδος, pin 2 DO ψηφιακή έξοδος, pin 3 GND για την τροφοδοσία του και pin 4 VCC επίσης για την τροφοδοσία του.

#### 4.10.1. Συνδεσμολογία για node του αισθητήρα φωτός:



Εικόνα 51 - Συνδεσμολογία arduino - αισθητήρα φωτός

#### 4.10.2. Κώδικας που χρησιμοποιήθηκε για τον αισθητήρα φωτός:

```
#include <nRF24L01.h> //eisagogi aparaititon vivliothikon
#include <printf.h>
#include <RF24.h>
#include <RF24_config.h>
#include<SPI.h>
#include<RF24.h>

// ce, csn pins
RF24 radio(9, 10);
// elaxisti kai megisti timi pou diavazei o aisthitiras
const int sensorMin = 0; // elaxissti
const int sensorMax = 1024; // megisti

void setup(void) {
  radio.begin();//ekkinisi tou nrf24l01 kai rythmiseis tou
  radio.setPALevel(RF24_PA_MAX);
  radio.setChannel(0x76);
  radio.openWritingPipe(0xF0F0F0F0E1LL);
  radio.enableDynamicPayloads();
}
```

```

radio.powerUp();

Serial.begin (9600);

const char text[] = "WSN FLAME SENSOR STARTED"; //dilwsi pinaka xaraktirwn me
minima
radio.write(&text, sizeof(text));
}

void loop(void) {

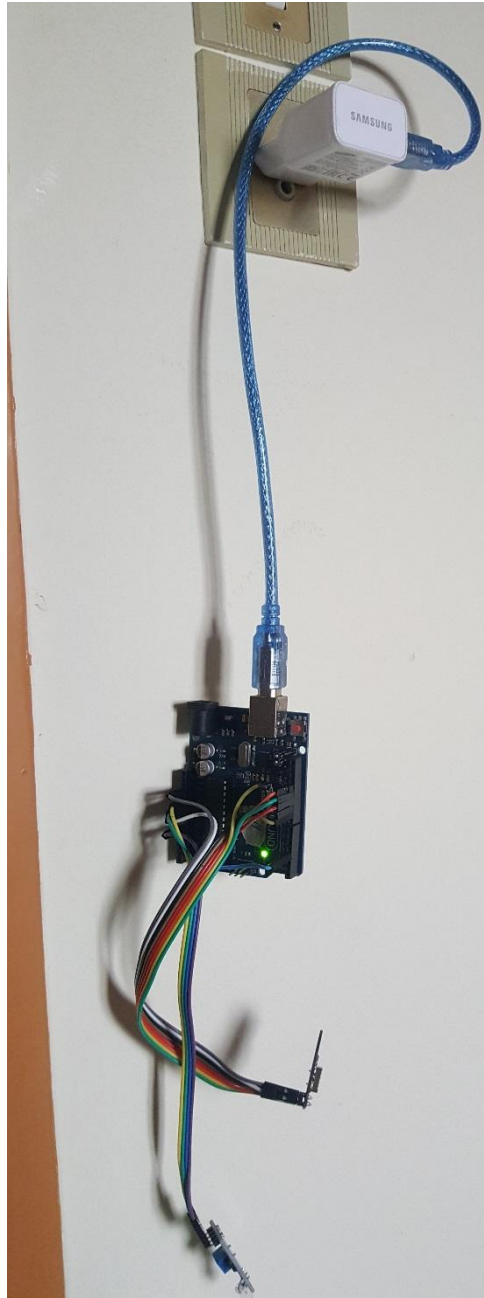
    delay(1000);
    // diavazoume tin timi tou aisthitira stin analog A0
    int sensorReading = analogRead(A0);
    int range = map(sensorReading, sensorMin, sensorMax, 0, 3);
    char text1[] = "WSN CLOSE FIRE";// pinakes xaraktirwn me minimata pou tha
emfanizontai
    char text2[] = "WSN DISTANT FIRE";
    char text3[] = "WSN NO FIRE";
    // synthiki switch me thn opoia, analoga thn timi tou aisthitira typwnei to
antistixo minima kai parallila to stelnei me to nrf24l01
    switch (range) {
    case 0:
        Serial.println("*** Close Fire ***");

        radio.write(&text1, sizeof(text1));
        break;
    case 1:
        Serial.println("*** Distant Fire ***");

        radio.write(&text2, sizeof(text2));
        break;
    case 2:
        Serial.println("No Fire");

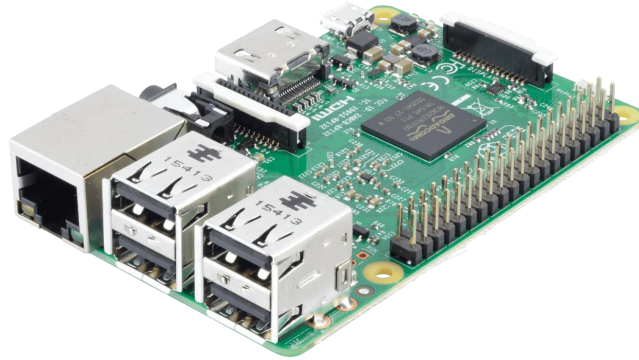
        radio.write(&text3, sizeof(text3));
        break;
    }
    delay(10000); // anamoni 10 second mexri thn epomeini metrisi
}

```



Εικόνα 52 - Arduino - Αισθητήρας φωτός

#### 4.11. Σύνδεση Raspberry Pi 3- Διαδίκτυο



Εικόνα 53 - Raspberry Pi 3

Όπως προαναφέραμε, στο Raspberry Pi 3 είναι εγκατεστημένος ένας webserver ο οποίος είναι υπεύθυνος για τις html σελίδες που παρουσιάζουν τα γραφήματα που εξάγονται από τα δεδομένα των αισθητήρων. Επίσης είναι εγκατεστημένος και ένας sql server στον οποίο είναι η βάση δεδομένων μας, όπου αποθηκεύονται οι μετρήσεις των αισθητήρων.

Για να έχει ο χρήστης πρόσβαση στα παραπάνω και να μπορεί απομακρυσμένα να δει τις σελίδες με τα γραφήματα και να έχει πλήρη εικόνα των αποτελεσμάτων και τη διαχείρισή αυτών, είναι απαραίτητη η πρόσβαση του Raspberry Pi 3 στο διαδίκτυο. Αυτό επιτυγχάνεται με τη σύνδεση του Raspberry Pi 3 στο οικιακό μας modem/router και τις απαραίτητες ρυθμίσεις σε αυτό, ώστε να μπορεί ο χρήστης να έχει πρόσβαση. Πιο συγκεκριμένα αφού συνδέσουμε το Raspberry Pi 3 στο modem/router μας μέσω της θύρας Ethernet, ανοίγουμε τις κατάλληλες πόρτες, ώστε να έχουμε κίνηση από τον "έξω" κόσμο προς το raspberry. Επιπλέον, λόγω του ότι δεν έχουμε στατική ip από τον πάροχό μας χρησιμοποιούμε την υπηρεσία dyndns, η οποία μας δίνει μια διεύθυνση της μορφής myname.dyndns.org, πίσω από την οποία "κρύβεται" κάθε φορά η δυναμική ip μας. Έτσι εμείς χρησιμοποιούμε τη σταθερή αυτή διεύθυνση χωρίς να μας νοιάζει ποια είναι η δυναμική ip μας κάθε φορά.

## 4.12. Χρήστες

### 4.12.1. Χρήστης Διαχειριστής

Ο Χρήστης Διαχειριστής πραγματοποιεί ελέγχους κατά την λειτουργία της εφαρμογής (έλεγχος για την ενημέρωση της εφαρμογής με νέα δεδομένα). Είναι παθητικός υποχρεωτικός δράστης.

<b>Description</b>	Ο Χρήστης Διαχειριστής πραγματοποιεί ελέγχους κατά την λειτουργία της εφαρμογής (έλεγχος για την ενημέρωση της εφαρμογής με νέα δεδομένα). Είναι παθητικός υποχρεωτικός δράστης.
<b>Aliases</b>	Καμία
<b>Inherits</b>	Καμία
<b>Actor Type</b>	Παθητικός χρήστης . Υποχρεωτικός, δουλεύει στο back end
<b>Contact Person</b>	
<b>Contact Details</b>	

### 4.12.2. Χρήστης Αναζήτησης

Ο Χρήστης Αναζήτησης πραγματοποιεί την αναζήτηση των τιμών των αισθητήρων είτε συνολικά είτε ανά αισθητήρα. Είναι ενεργός δράστης.

<b>Description</b>	Ο Χρήστης Αναζήτησης πραγματοποιεί την αναζήτηση για καιρικά δεδομένα είτε ανά αισθητήρα είτε βάσει ημερομηνιών. Είναι ενεργός δράστης.
--------------------	---

<b>Aliases</b>	Καμία
<b>Inherits</b>	Καμία
<b>Actor Type</b>	Ενεργός χρήστης - δρα στο front end της εφαρμογής.
<b>Contact Person</b>	
<b>Contact Details</b>	

### ***4.13. Περιγραφές χρήσης (Use Case Descriptions)***

#### ***Αναζήτηση Δεδομένων***

Η εφαρμογή επιτρέπει στον χρήστη να αναζητάει τις μετρήσεις των αισθητήρων. Ο χρήστης αναζητεί δεδομένα είτε βάσει αισθητήρα, είτε βάσει ημερομηνιών. Επιπλέον έχει τη δυνατότητα να επιλέγει το διάστημα της ανανέωσης των τιμών του αισθητήρα.

#### **Actors:**

Χρήστης Αναζήτηση

#### **Preconditions:**

1. Το portal να λειτουργεί εύρυθμα και να είναι επικαιροποιημένο με τις τιμές των αισθητήρων

#### **Scenario Text:**

2. Ο χρήστης επιλέγει αισθητήρα.
3. Εμφανίζονται τα γραφήματα για τις επιλογές του χρήστη.

#### **Alternative Courses:**

4. Ο χρήστης επιλέγει ημερομηνία/ες.
5. Ο χρήστης επιλέγει διάστημα ανανέωσης.

#### **Extends:**

None

**User Interfaces:**

localhost/\*.php

**Constraints:**

**Questions:**

**Notes:**

**Authors:**

**Source Documents:**

#### ***4.14. Σχεδιασμός Οθονών (Grafana)***

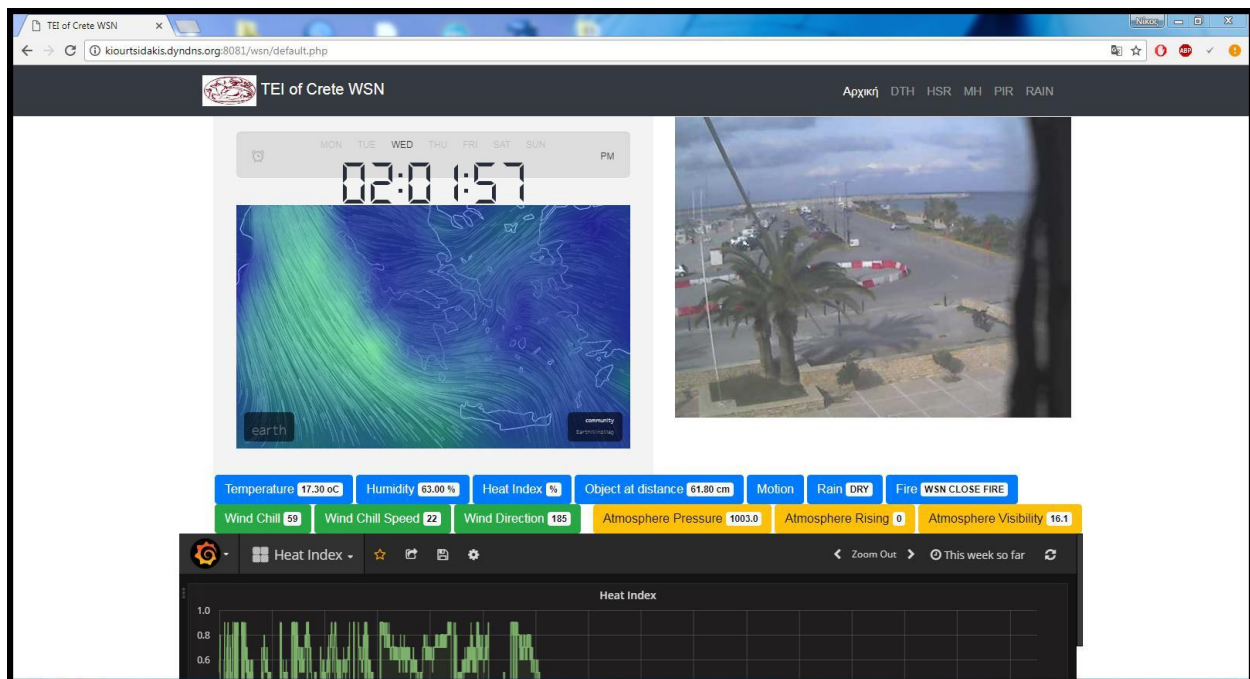
Η οθόνη του ιστοτόπου θα χωρίζεται σε δυο διαφορετικές ενότητες. Στην πρώτη ενότητα βρίσκεται το μενού του ιστοτόπου και στην δεύτερη εμφανίζονται τα γραφήματα και οι τρέχουσες τιμές των αισθητήρων. Από το μενού, ο χρήστης έχει τη δυνατότητα να αναζητήσει και να εμφανίσει τις τιμές του κάθε αισθητήρα ξεχωριστά. Οι ενότητες έχουν στοίχιση στο κέντρο και είναι οριοθετημένες η μία κάτω από την άλλη. Με τη χρήση του πλαισίου εφαρμογής bootstrap, η οθόνη είναι συμβατή όλους τους τύπους των συσκευών (π.χ. κινητές συσκευές, tablets) και όλες τις διαστάσεις. Ενδεικτικά παρακάτω είναι το σχήμα της οθόνης:



Ενότητα 1 (Top Menu)

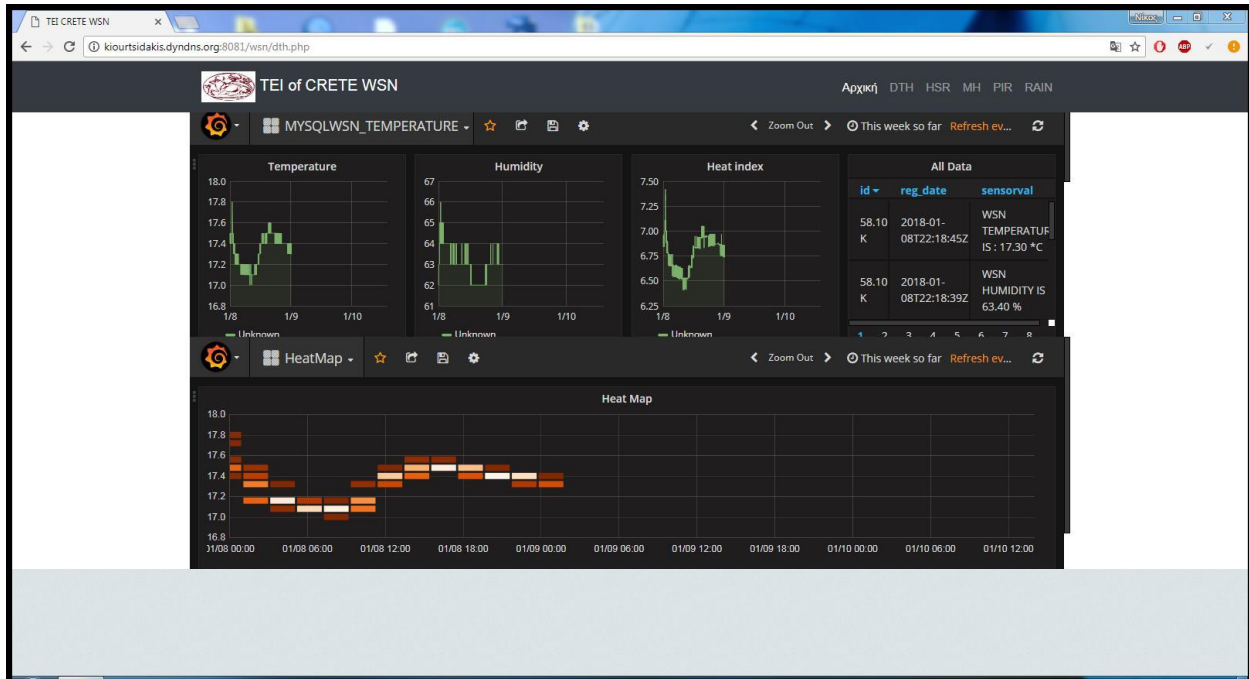
Ενοτήτα 2 (Grafana Γραφήματα)

Η σχεδίαση της οθόνης βασίστηκε στο starter template του πλαισίου Bootstrap. Η τελική οθόνη του portal είναι η παρακάτω:



Εικόνα 54 - Κεντρική σελίδα του portal

Παρακάτω παρουσιάζονται οι οθόνες, που παρουσιάζουν τις τιμές των αισθητήρων.



Εικόνα 55 – Οθόνη με τιμές ενός αισθητήρα

#### 4.15. Ενσωμάτωση Portal

Στην παρακάτω ενότητα παρουσιάζονται γενικές οδηγίες σχετικά με την εγκατάσταση και παραμετροποίηση της ηλεκτρονικής πύλης και του δικτύου των αισθητήρων.

Τα βήματα για την απρόσκοπτη λειτουργία της ηλεκτρονικής πύλης είναι τα παρακάτω:

1. εκκίνηση όλων των arduino (έλεγχος συνδεσμολογιών)
2. εκκίνηση του raspberry (έλεγχος συνδεσμολογιών)
3. εκκίνηση της υπηρεσίας του apache web server
4. εκκίνηση της υπηρεσίας mysql
5. εκκίνηση του python script στο raspberry το οποίο είναι υπεύθυνο για την λήψη των τιμών των αισθητήρων και την εισαγωγή στην βάση δεδομένων
6. εκκίνηση του διακομιστή του grafana
7. πλοήγηση στην κεντρική σελίδα της ηλεκτρονικής πύλης

Πιο συγκεκριμένα, τα αρχεία \*.ino είναι εγκατεστημένα στις πλακέτες arduino και μεταδίδουν τα δεδομένα του κάθε αισθητήρα μέσω του rf module στην πλακέτα του raspberry που φιλοξενείται τόσο ο διακομιστής όσο και η σχεσιακή βάση δεδομένων.

Όλα τα αρχεία \*.php είναι υπεύθυνα για την ανάγνωση και παρουσίαση των δεδομένων της βάσης δεδομένων στην ηλεκτρονική πύλη με δυναμικό τρόπο.

Το αρχείο rasp1.py είναι υπεύθυνο για τη λήψη των τιμών των αισθητήρων μέσω rf modules και την εισαγωγή τους στη βάση δεδομένων mysql.

Η βάση δεδομένων αποτελείται από έναν πίνακα στον οποίο αποθηκεύονται οι τιμές των αισθητήρων (ημερομηνία/τιμή).

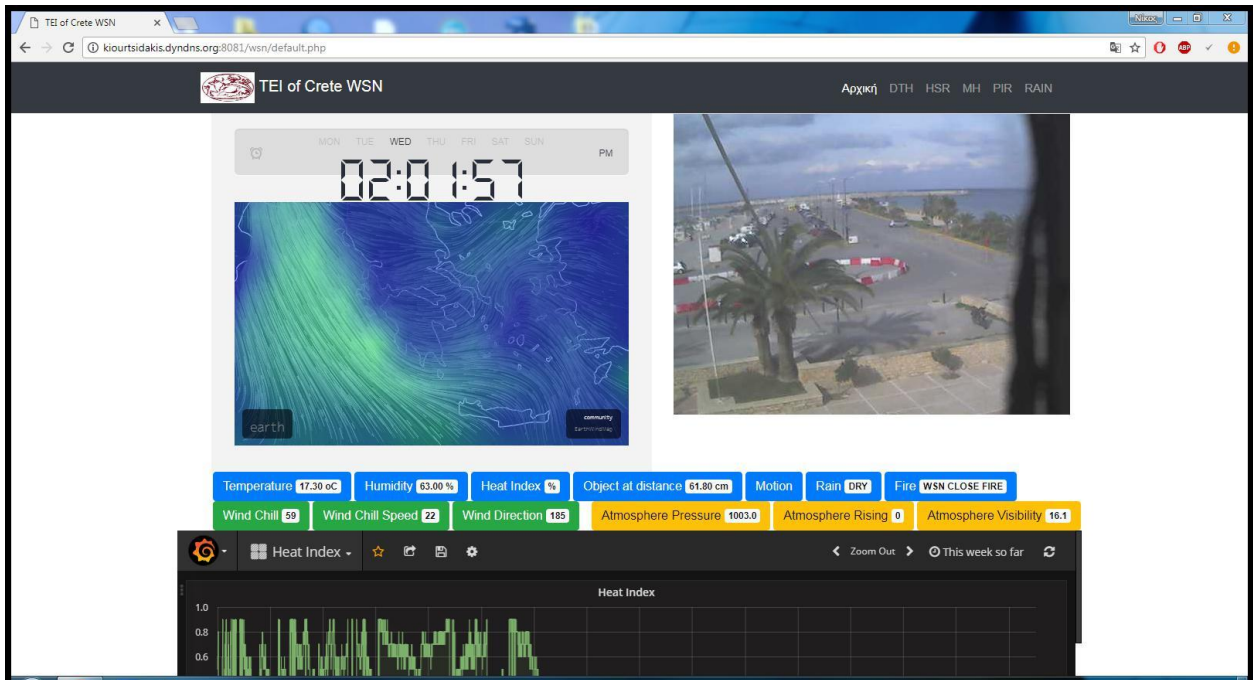
#### ***4.16. Σενάρια χρήσης/Εκσφαλμάτωση***

##### **4.16.1. Εκσφαλμάτωση Οθονών**

Για την καλή λειτουργία της οθόνης πραγματοποιήθηκαν δοκιμές σε όλους τους γνωστούς περιηγητές Internet Explorer 8 έως 11, Mozilla Firefox και Google Chrome στον υπολογιστή μας. Με τη χρήση των εργαλείων Google Chrome Developer Tools ελέγχθηκε εάν είναι συμβατή η διεπαφή με κινητές συσκευές και με προσανατολισμό Portrait και Landscape.

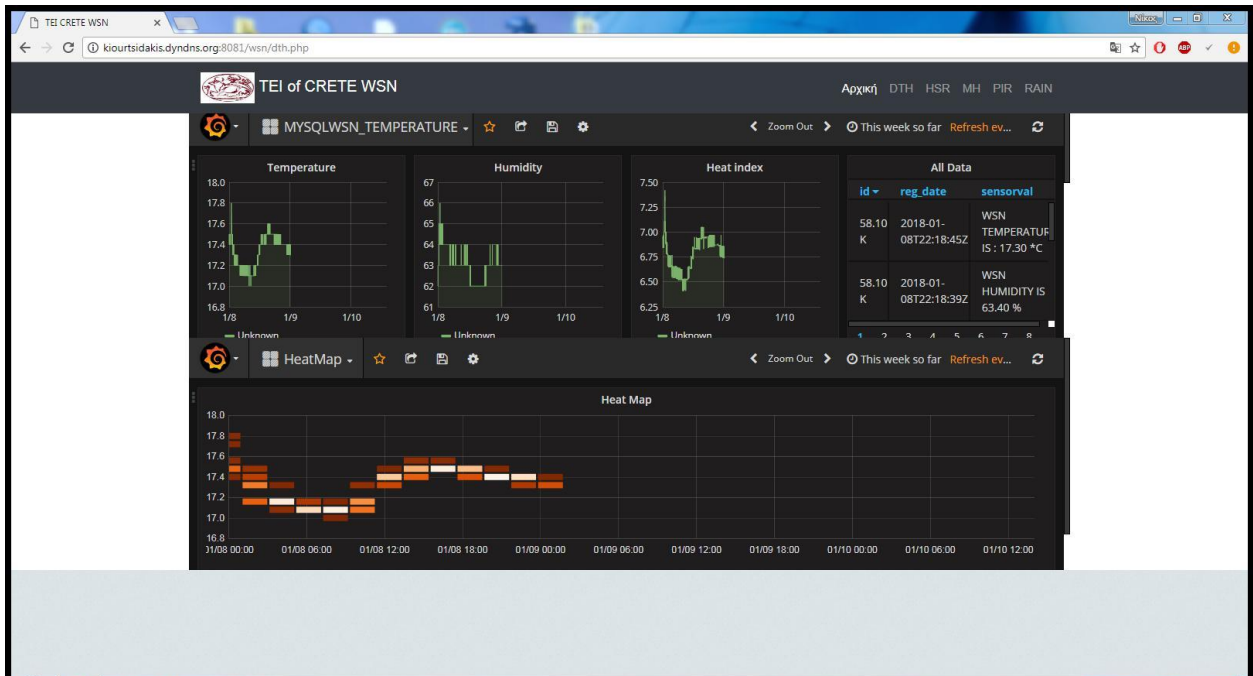
Παρακάτω παρουσιάζονται ενδεικτικά όλες οι οθόνες του συστήματος.

Στην παρακάτω εικόνα παρουσιάζεται η αρχική οθόνη της ηλεκτρονικής πύλης. Όπου στο πάνω μέρος είναι το μενού, στη συνέχεια η κεντρική οθόνη χωρίζεται σε δύο μέρη. Στα αριστερά παρουσιάζεται η ημερομηνία, η ώρα και η ταχύτητα των ανέμων ενώ στο δεξί μέρος παρουσιάζεται σε πραγματικό χρόνο το λιμάνι του Ρεθύμνου.



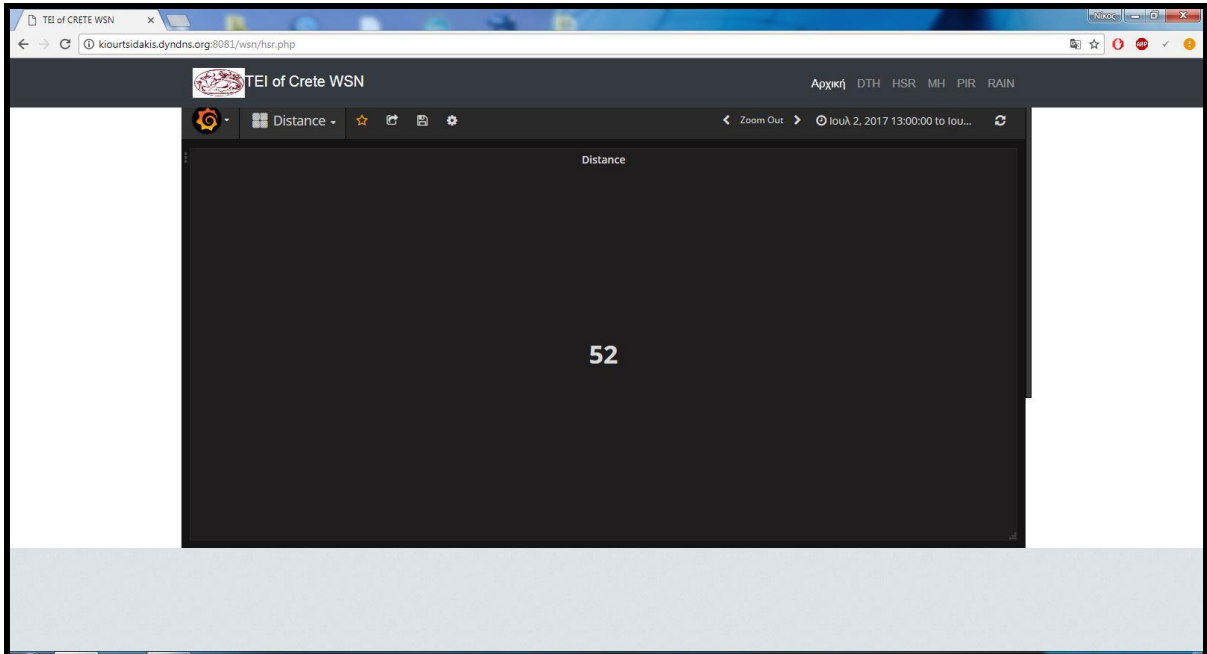
Εικόνα 56 - Κεντρική οθόνη ηλεκτρονικής πύλης

Στην παρακάτω οθόνη παρουσιάζονται οι τιμές από τον αισθητήρα θερμοκρασίας.



Εικόνα 57 - Οθόνη αισθητήρα θερμοκρασίας/υγρασίας

Στην παρακάτω οθόνη παρουσιάζεται η τελευταία μέτρηση από τον αισθητήρα απόστασης.



Εικόνα 58 - Οθόνη αισθητήρα απόστασης

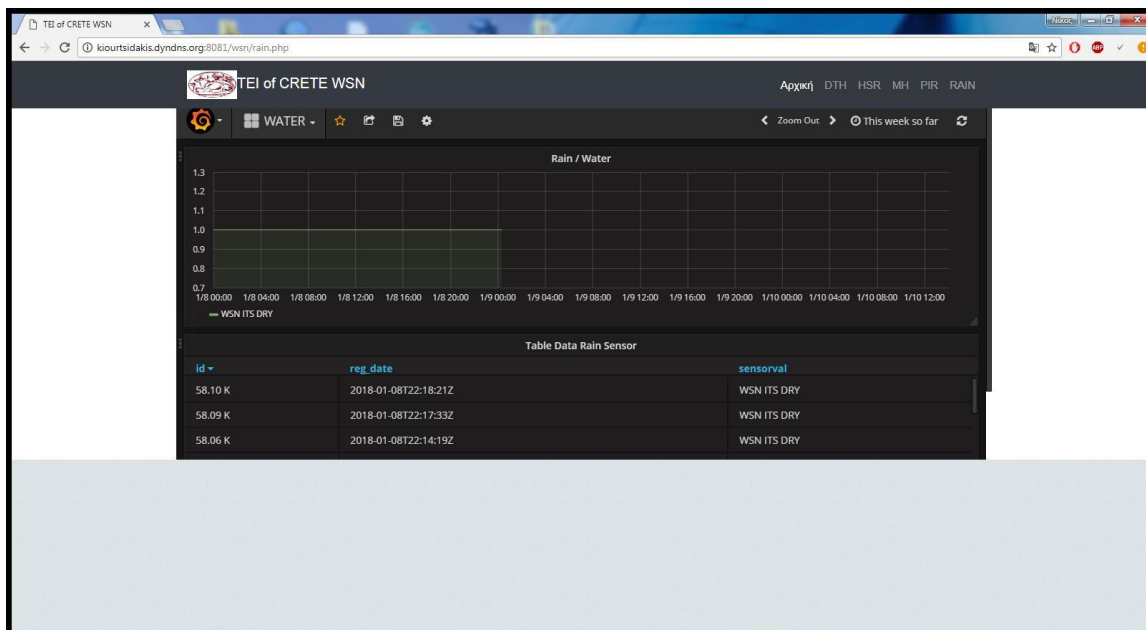
Στην παρακάτω οθόνη παρουσιάζονται οι μετρήσεις από τον αισθητήρα φωτιάς.

The screenshot shows a web browser window displaying the TEI of Crete WSN interface. The URL is [kiourtsidakis.dyndns.org:8081/wsn/mh.php](http://kiourtsidakis.dyndns.org:8081/wsn/mh.php). The page title is "TEI of CRETE WSN". The main content area is dark with a table titled "FIRE". The table has three columns: "Time", "Metric", and "Value". The table contains five rows of data, all with "WSN CLOSE FIRE" as the metric and "-" as the value. There is an "ADD ROW" button at the bottom of the table. The interface includes a navigation menu with "Αρχική", "DTH", "HSR", "MH", "PIR", and "RAIN". There are also icons for settings, home, and refresh, and a "Zoom Out" button.

Time	Metric	Value
2017-12-30 11:21:26	WSN CLOSE FIRE	-
2017-12-30 11:20:25	WSN CLOSE FIRE	-
2017-12-30 11:19:37	WSN CLOSE FIRE	-
2017-12-30 11:18:06	WSN CLOSE FIRE	-
2017-12-30 11:17:35	WSN CLOSE FIRE	-

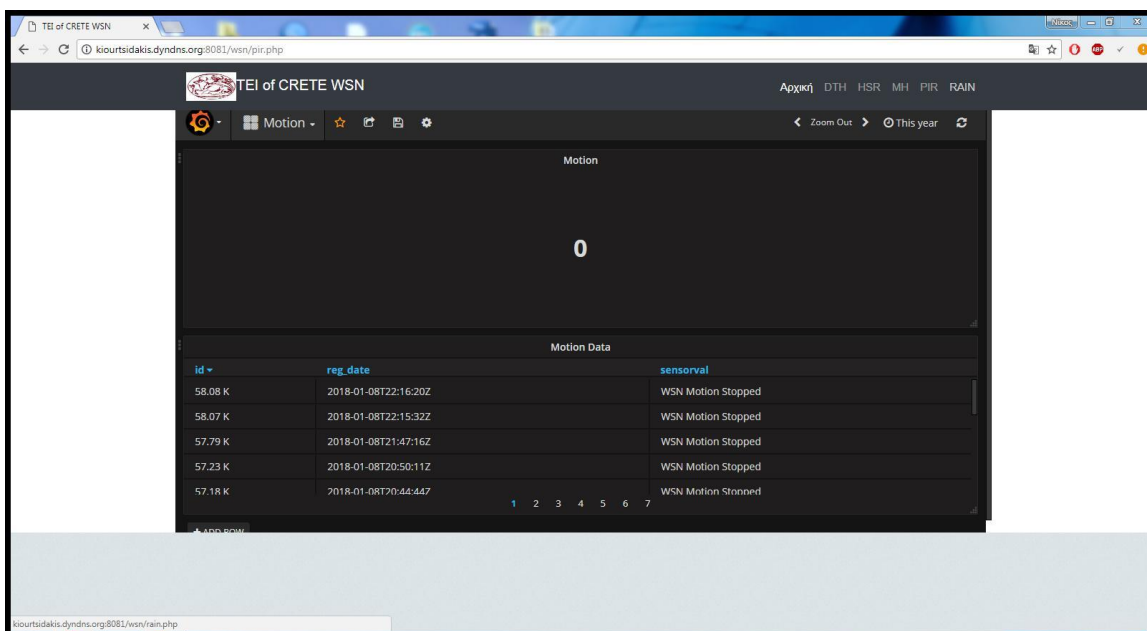
Εικόνα 59 - Οθόνη αισθητήρα φωτός

Στην παρακάτω εικόνα παρουσιάζονται οι μετρήσεις του αισθητήρα βροχής.



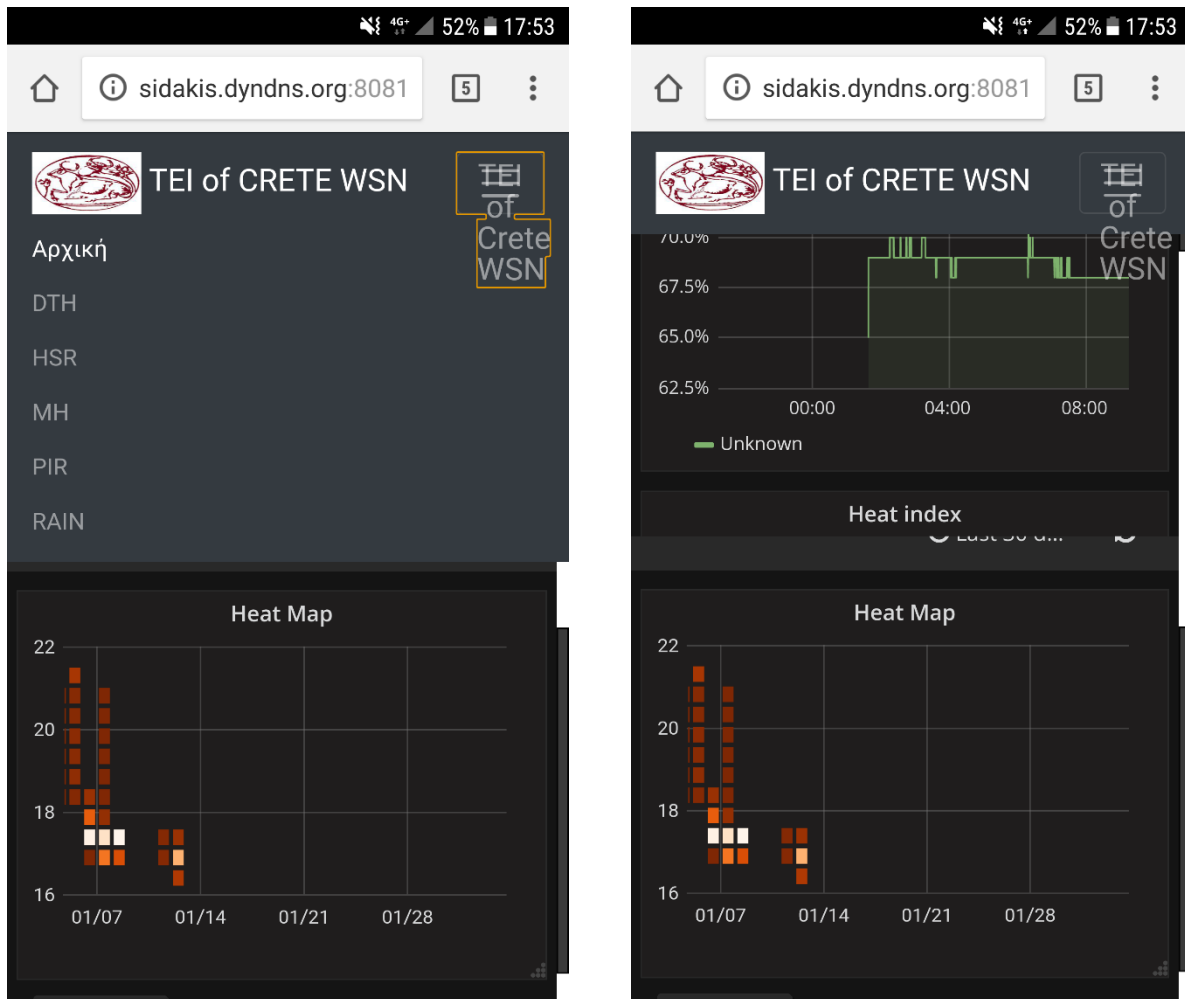
Εικόνα 60 - Οθόνη αισθητήρα βροχής

Στην παρακάτω εικόνα παρουσιάζονται οι μετρήσεις του αισθητήρα κίνησης.

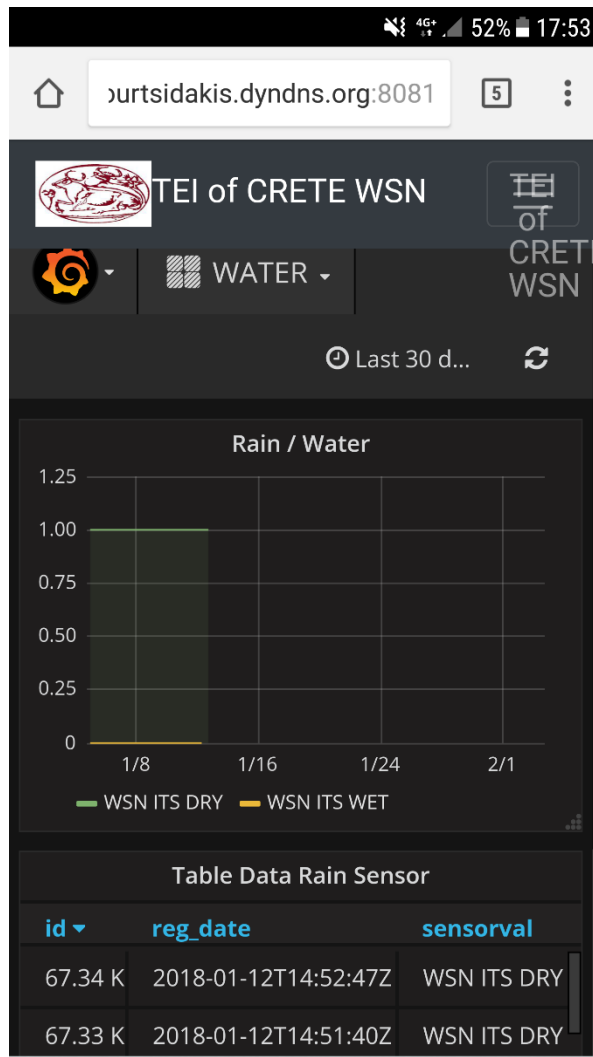


Εικόνα 61 - Οθόνη αισθητήρα κίνησης

Το portal είναι συμβατό και με κινητές συσκευές και προσαρμόζεται ικανοποιητικά σε όλα τα μεγέθη οθόνης, παρακάτω μπορείτε να δείτε μερικά στιγμιότυπα από την οθόνη κινητού τηλεφώνου:

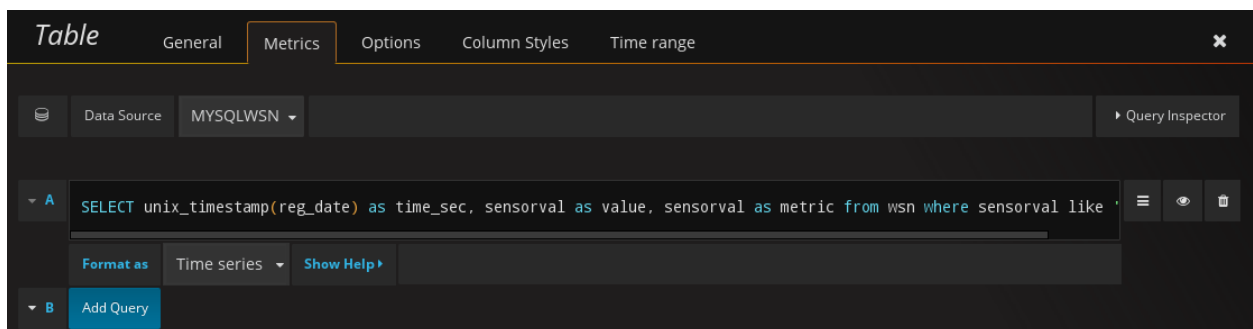


Εικόνα 62 – Οθόνες (Smartphone) αισθητήρα θερμοκρασίας



Εικόνα 63 – Οθόνη (smartphone) αισθητήρα βροχής

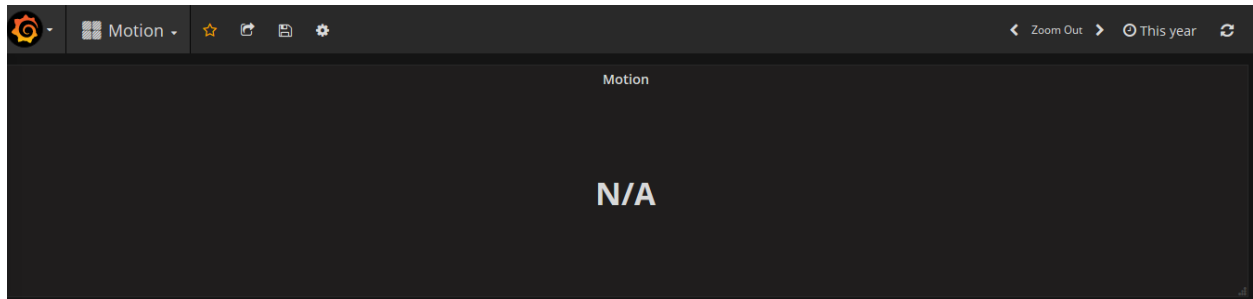
Στην παρακάτω οθόνη παρουσιάζεται το ερώτημα sql το οποίο επιστρέφει δεδομένα αισθητήρα σε κατάλληλη μορφή ώστε να παρουσιαστούν από το πλαίσιο εφαρμογής grafana.



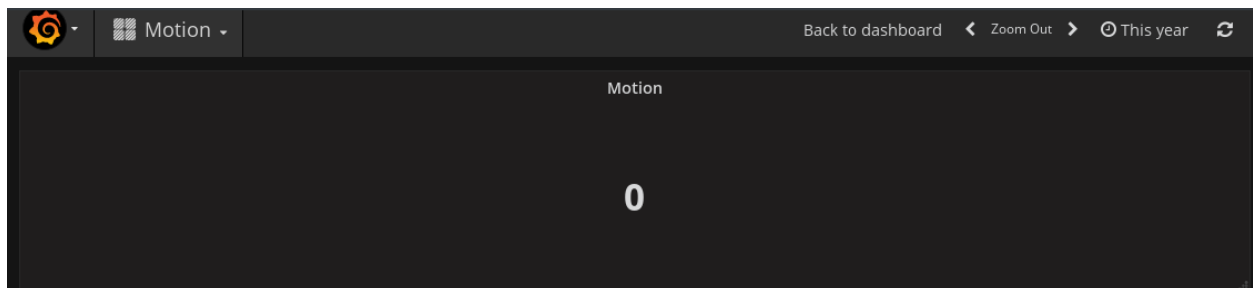
Εικόνα 64 - Οθόνη παραμετροποίησης grafana



Στις παρακάτω οθόνες παρουσιάζονται οι μετρήσεις του αισθητήρα κίνησης και το sql ερώτημα το οποίο χρησιμοποιείται.



```
SELECT
  max(UNIX_TIMESTAMP(reg_date)) as time_sec,
  sensorval = 'motion detected' as value,
  sensorval as metric
FROM wsn
WHERE $__timeFilter(reg_date) and sensorval like '%otion%'
ORDER BY reg_date ASC
```



A screenshot of a dashboard titled "Motion Data" showing a table with 5 rows. Below the table is a "Table" configuration panel with tabs for "General", "Metrics", "Options", "Column Styles", and "Time range". The "Metrics" tab is active. At the bottom, there is a "Data Source" dropdown set to "default" and a "Query Inspector" showing the SQL query used to generate the data.

id	reg_date	sensorval
5.52 K	2017-12-21T21:43:54Z	WSN Motion Stopped
5.52 K	2017-12-21T21:42:54Z	WSN Motion Stopped
5.52 K	2017-12-21T21:42:18Z	WSN Motion Stopped
5.52 K	2017-12-21T21:40:45Z	WSN Motion Stopped
1.66 K	2017-09-12T19:01:57Z	WSN Motion Stopped

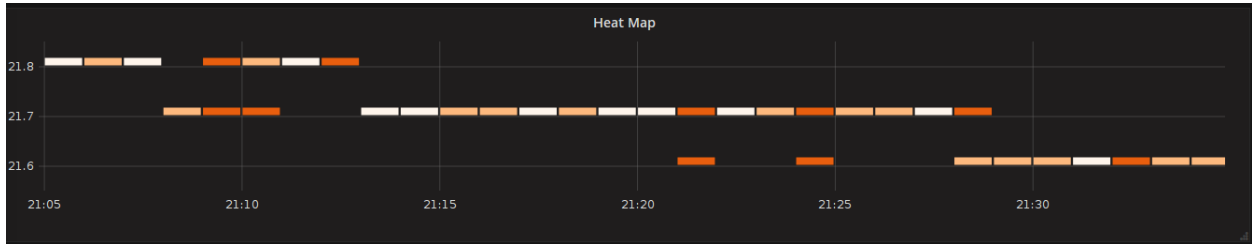
**Table** | General | **Metrics** | Options | Column Styles | Time range | ✕

Data Source: default | Query Inspector

```
SELECT * from wsn where sensorval like '%otion%'
```

Εικόνα 65 - Οθόνες αισθητήρα ανίχνευσης κίνησης

Στις παρακάτω οθόνες παρουσιάζονται μετρήσεις από τον αισθητήρα DHT και το sql query το οποίο χρησιμοποιείται.



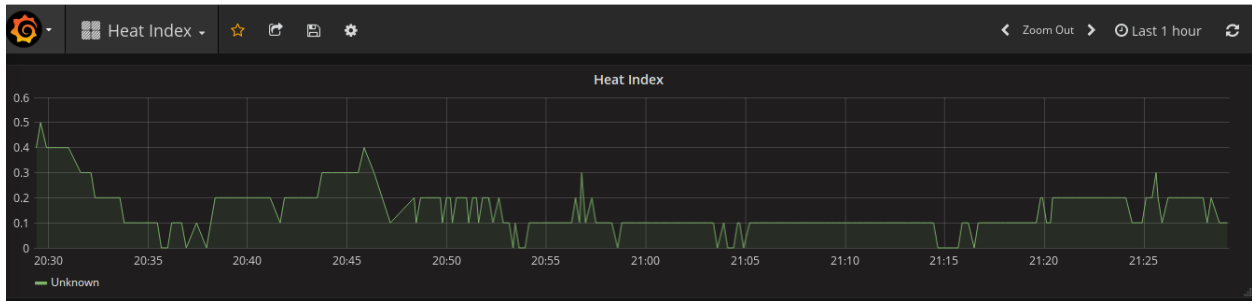
**Heatmap** | General | Metrics | Axes | Display | Time range | X

Data Source: MYSQLWSN | Query Inspector

```
SELECT
  UNIX_TIMESTAMP(reg_date) as time_sec,
  convert(substring(sensorval,22,4), decimal(9,2)) as value
FROM wsn
WHERE $__timeFilter(reg_date) and sensorval like '%Temperature%'
ORDER BY reg_date ASC
```

Format as: Time series | Show Help

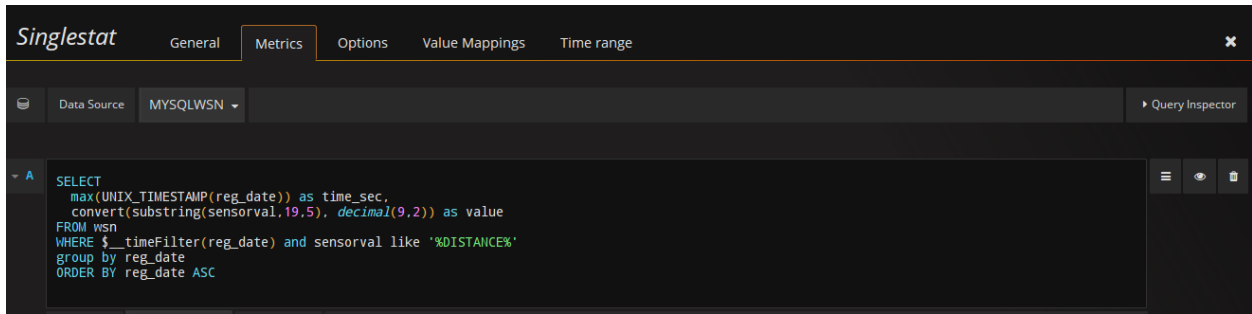
Add Query



Data Source: MYSQLWSN | Query Inspector

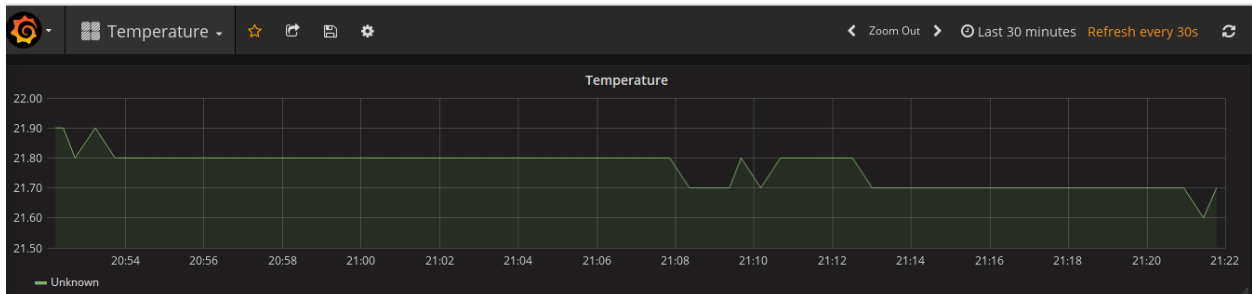
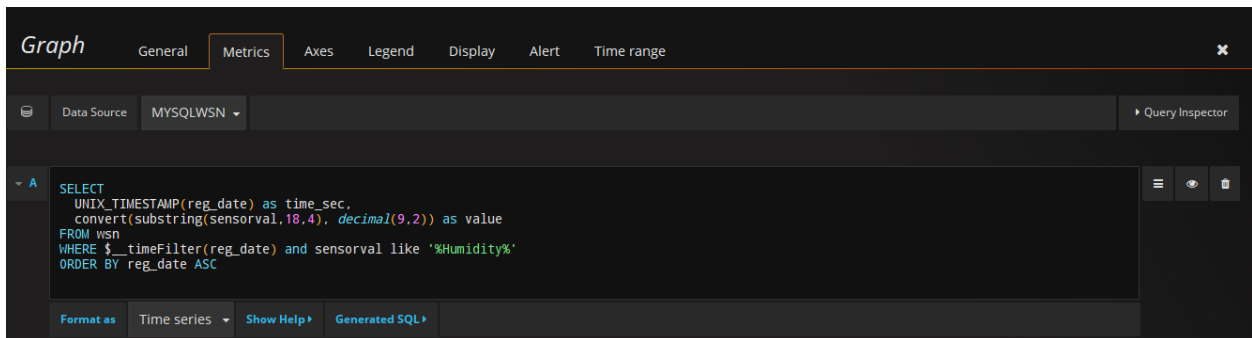
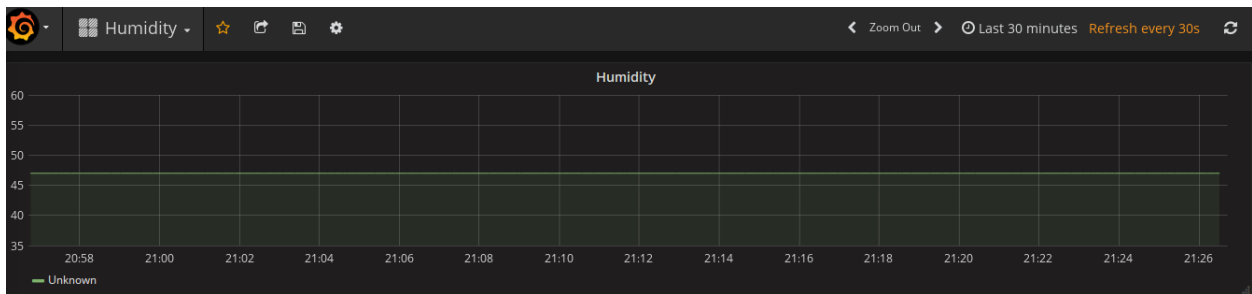
```
SELECT
  UNIX_TIMESTAMP(reg_date) as time_sec,
  convert(substring(sensorval,21,4), decimal(9,2)) as value
FROM wsn
WHERE $__timeFilter(reg_date) and sensorval like '%Humidity%'
ORDER BY reg_date ASC
```

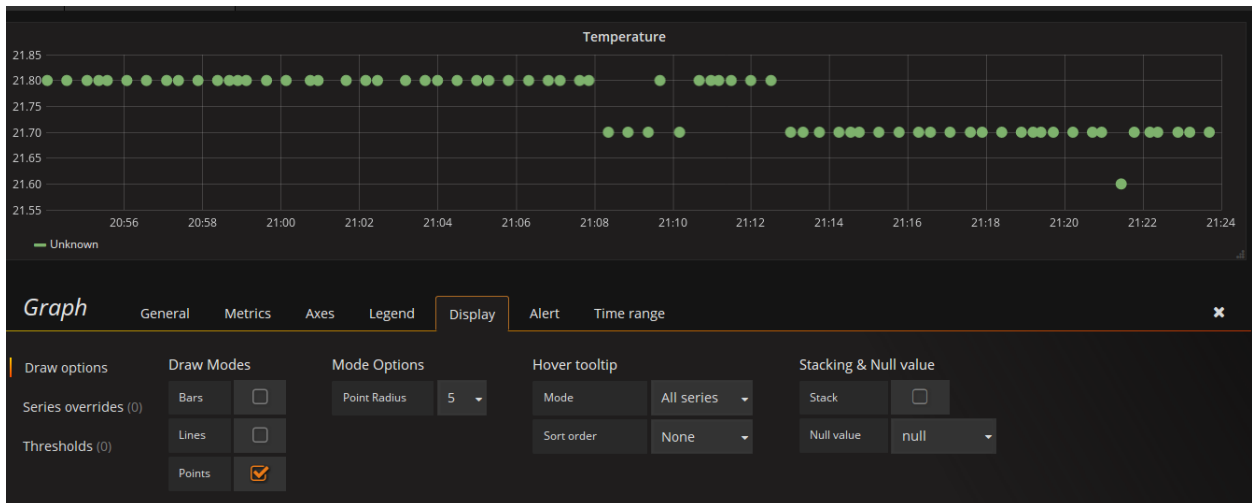
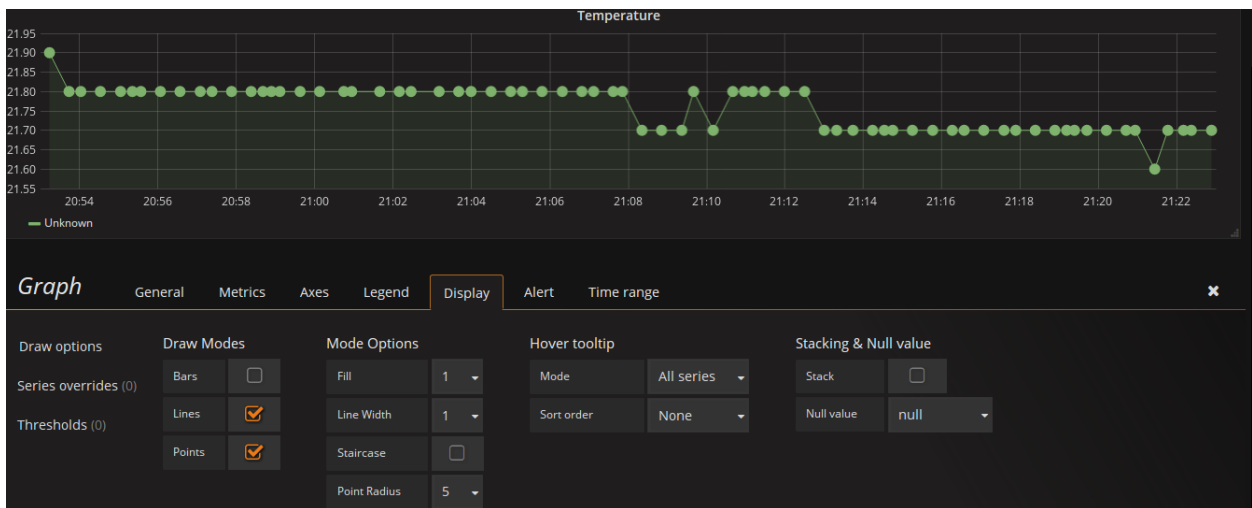
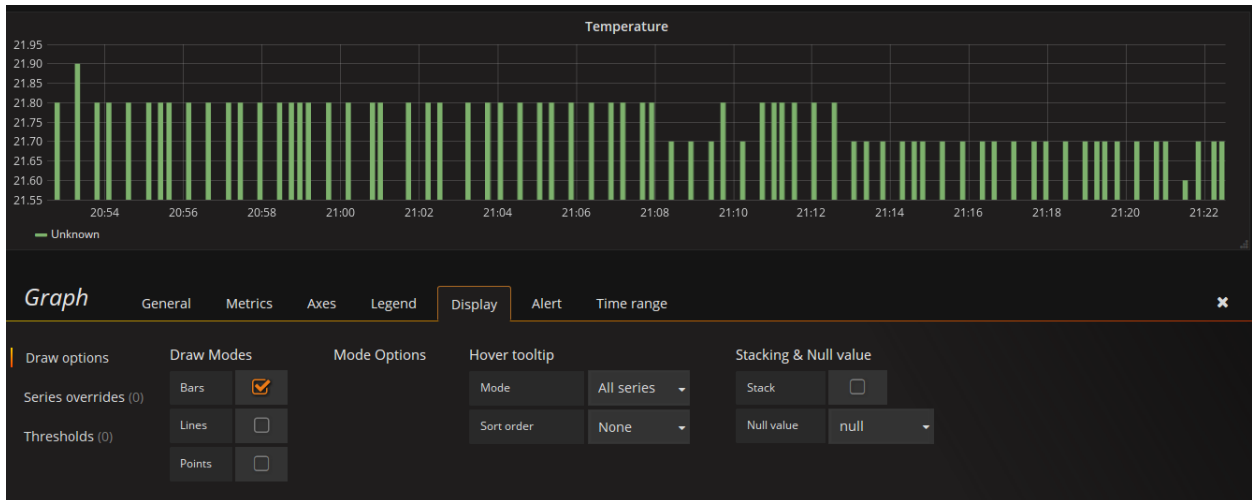
Format as: Time series | Show Help



Εικόνα 66 - Εικόνες αισθητήρα DHT

Στις παρακάτω οθόνες παρουσιάζονται μετρήσεις αισθητήρων και τα sql ερωτήματα τα οποία χρησιμοποιούνται καθώς και διαφορετικά είδη γραφημάτων.



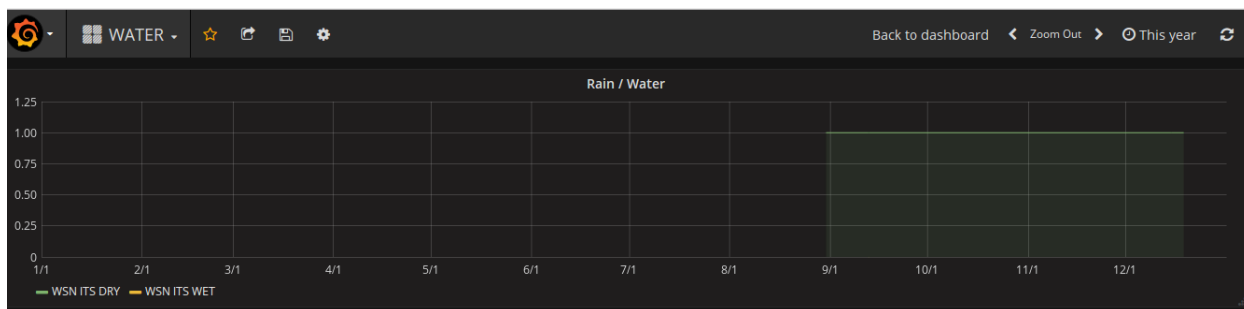


**Graph** General Metrics Axes Legend Display Alert Time range ✕

Data Source: MYSQLWSN Query Inspector

```

SELECT
  UNIX_TIMESTAMP(reg_date) as time_sec,
  convert(substring(sensorval,22,4), decimal(9,2)) as value
FROM wsn
WHERE $__timeFilter(reg_date) and sensorval like '%Temperature%'
ORDER BY reg_date ASC
    
```



**Graph** General Metrics Axes Legend Display Alert Time range

Data Source: MYSQLWSN

```

SELECT
  UNIX_TIMESTAMP(reg_date) as time_sec,
  sensorval = 'WSN ITS DRY' as value,
  sensorval as metric
FROM wsn
WHERE $__timeFilter(reg_date) and (sensorval like '%DRY%' or sensorval like '%WET%')
ORDER BY reg_date ASC
    
```

Εικόνα 67 - Οθόνες μετρήσεις αισθητήρων

## **Κεφάλαιο 5. Συμπεράσματα & Μελλοντικές Επεκτάσεις**

### **5.1. Συμπεράσματα**

Με το πέρας της πτυχιακής μας εργασίας ολοκληρώθηκε μία ηλεκτρονική πύλη η οποία αναλαμβάνει την εύπεπτη παρουσίαση των τιμών σε πραγματικό και μη χρόνο ενός δικτύου αισθητήρων.

Τα δεδομένα παρουσιάζονται σε φιλική προς το χρήστη μορφή με τη βοήθεια τεχνολογιών αιχμής όπως το πλαίσιο εφαρμογής Grafana, το Bootstrap, το Css3, η γλώσσα προγραμματισμού HTML5 και η βάση δεδομένων mysql.

Επιπλέον, με την ολοκλήρωση της πτυχιακής μας εργασίας εξοικειωθήκαμε και αποκτήσαμε δεξιότητες με τεχνολογίες αιχμής όπως το πλαίσιο εφαρμογής Bootstrap, το CCS3, η γλώσσα HTML5, το Grafana, την Javascript, την JQuery, την σχεσιακή βάση δεδομένων mysql 5 και το περιβάλλον ανάπτυξης Arduino ide.

### **5.2. Μελλοντικές Επεκτάσεις**

Όσον αφορά την επεκτασιμότητα του δικτύου αισθητήρων, θα μπορούσαμε μελλοντικά να αυξήσουμε τον αριθμό των αισθητήρων, ώστε να έχουμε αποτελέσματα μεγαλύτερης ακρίβειας ανά είδος μέτρησης. Έτσι θα είχαμε μεγαλύτερη περιοχή κάλυψης λόγω του μεγαλύτερου αριθμού αισθητήρων και τα αποτελέσματά μας θα ήταν πιο ποιοτικά λόγω του μεγαλύτερου αριθμού μετρήσεων που θα είχαμε και της εξαγωγής του μέσου όρου αυτών.

Μια άλλη αναβάθμιση που θα μπορούσε να γίνει στο δίκτυο αισθητήρων είναι η ενεργειακή τους αυτονομία. Τώρα μπορούν να τροφοδοτηθούν είτε από το δίκτυο, με 220V με τους κατάλληλους μετασχηματιστές που τροφοδοτούν τις πλακέτες, είτε από μπαταρίες των 9V που με τους κατάλληλους αντάπτορες. Τα nodes όμως θα μπορούσαν να γίνουν ενεργειακά αυτόνομα με την προσθήκη της κατάλληλης διάταξης ενός φωτοβολταϊκού πάνελ στο κάθε node, το οποίο θα φορτίζει τις μπαταρίες του και έτσι θα μπορεί να τροφοδοτείται απρόσκοπτα για πάρα πολύ καιρό.

## Βιβλιογραφία

- [1] K. Sohraby, D. Minoli, T. Znati, “Wireless sensor networks: Technology, Protocols, and Applications”, Wiley-Interscience, 2007, pp. 273-281
- [2] M. Ilyas, I. Mahgoub, “Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems”, CRC Press, Boca Raton, 2004
- [3] Ι.Ν.Αβαριτσιώτης, Τεχνολογία Αισθητήρων & Μικροσυστημάτων, Εκδόσεις ΕΜΠ, Αθήνα 2003
- [4] [https://www.w3schools.com/html/html5\\_intro.asp](https://www.w3schools.com/html/html5_intro.asp)
- [5] <https://www.javascript.com/>
- [6] <http://www.css3.info/>
- [7] <https://jquery.com/>
- [8] <https://el.wikipedia.org/wiki/Bootstrap>
- [9] <https://grafana.com/>
- [10] <https://www.raspberrypi.org/>
- [11] <https://www.arduino.cc/>
- [12] <https://en.wikipedia.org/wiki/Ultrasound>
- [13] <http://www.modmypi.com/blog/hc-sr04-ultrasonic-range-sensor-on-the-raspberry-pi>
- [14] <https://www.mpja.com/download/31227sc.pdf>
- [15] DHT-22, <http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Sensors/Weather/RHT03.pdf>
- [16] <https://randomnerdtutorials.com/guide-for-rain-sensor-fc-37-or-yl-83-with-arduino/>
- [17] [https://www.sparkfun.com/datasheets/Components/SMD/nRF24L01Pluss\\_Preliminary\\_Product\\_Specification\\_v1\\_0.pdf](https://www.sparkfun.com/datasheets/Components/SMD/nRF24L01Pluss_Preliminary_Product_Specification_v1_0.pdf)
- [18] <https://el.wikipedia.org/wiki/Python>