

**Μελέτη δομής, οργάνωσης και διασύνδεσης συστημάτων IoT και
αρμονικής συνεργασίας των πρωτοκόλλων επικοινωνίας.
Πρόταση διδασκαλίας της υλοποίησης εφαρμογών με Node red**

Η Διπλωματική Εργασία
παρουσιάστηκε ενώπιον
του Διδακτικού Προσωπικού του
Πανεπιστημίου Αιγαίου

Σε Μερική Εκπλήρωση
των Απαιτήσεων για το Δίπλωμα του
Μηχανικού Πληροφοριακών και Επικοινωνιακών Συστημάτων

της
Άννας Μαμουζέλου
ΧΕΙΜΕΡΙΝΟ ΕΞΑΜΗΝΟ 2017-2018

Η ΤΡΙΜΕΛΗΣ ΕΠΙΤΡΟΠΗ ΔΙΔΑΣΚΟΝΤΩΝ ΕΠΙΚΥΡΩΝΕΙ
ΤΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
ΤΗΣ ΑΝΝΑΣ ΜΑΜΟΥΖΕΛΟΥ:

Εργίνα Καβαλλιεράτου, Επιβλέπων
Τμήμα Μηχανικών Πληροφοριακών και
Επικοινωνιακών Συστημάτων

Ανδρέας Παπασαλούρος, Μέλος
Τμήμα Μηχανικών Πληροφοριακών και
Επικοινωνιακών Συστημάτων

Ευστάθιος Σταματάτος, Μέλος
Τμήμα Μηχανικών Πληροφοριακών και
Επικοινωνιακών Συστημάτων

ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ
ΧΕΙΜΕΡΙΝΟ ΕΞΑΜΗΝΟ 2018

ΠΕΡΙΛΗΨΗ

Στην παρούσα διπλωματική εργασία επιλέχτηκε να αξιοποιηθούν τεχνολογίες, πρωτόκολλα και δίκτυα επικοινωνιών που είναι σε άμεση εφαρμογή και θα επεκταθούν στο άμεσο μέλλον.

Οι τεχνολογίες αυτές, προσαρμόστηκαν στη σύγχρονη κατοικία της οποίας οι συσκευές είναι συμβατικού (παλαιού) τύπου, δηλαδή δεν διαθέτουν interface IoT, όπως θα έχουν οι μελλοντικές. Έτσι οι προτάσεις μας είναι εφαρμόσιμες δίνοντας στην κατοικία άμεση λειτουργική εφαρμογή.

Αν και το σύνολο πραγματοποιήθηκε σε φορητό υπολογιστή κάτω από το λειτουργικό windows, εντούτοις για λόγους οικονομίας και οικολογίας, επιλέχθηκε ως μόνιμα συνδεδεμένο MQTT broker το μικροϋπολογιστικό σύστημα Raspberry Pi 3 που συνδυάζει τα πλεονεκτήματα της πολύ χαμηλής κατανάλωσης (~5w) και του χαμηλού κόστους απόκτησης.

Λαμβάνοντας υπόψη τα πιο πάνω, η παρούσα εργασία εκπονήθηκε με σκοπό να διδάξει πως μέσα από τις εφαρμογές των IoT μπορούμε να έρθουμε σε επαφή (να αγγίξουμε) το φυσικό περιβάλλον επιλέγοντας τον κατάλληλο αισθητήρα. Να διδάξει πως το φυσικό μέγεθος μετατρέπεται σε αντίστοιχο ηλεκτρικό σήμα και στη συνέχεια σε αριθμητικό μέγεθος, το οποίο μεταφέρεται, «ταξιδεύει», μέσα στο διαδίκτυο σε μορφή πακέτων του πρωτοκόλλου TCP/IP που περιέχουν ως χρήσιμο φορτίο ASCII χαρακτήρες και οδηγούνται στη διεπαφή (interface) με τον τελικό χρήστη, σε εικονοποιημένη μορφή με αξιοποίηση του NODE-RED.

Για να επιτευχθούν όλα αυτά στη διδασκαλία του μαθήματος της εκπαίδευσης στην πληροφορική, παρουσιάζεται αριθμός αισθητήρων και διδάσκεται ο τρόπος λειτουργίας τους, δηλαδή πως ο καθένας από αυτούς πρέπει να συνδέεται για να «αισθάνεται» το προς διερεύνηση αντίστοιχο φυσικό μέγεθος.

Δηλαδή πως το φυσικό μέγεθος μετατρέπεται σε ηλεκτρικό σήμα, πως λαμβάνεται στην κατάλληλα οργανωμένη είσοδο του μικροελεγκτή, το οποίο αξιοποιούμε με το κατάλληλα οργανωμένο λογισμικό, γραμμένο στη γλώσσα προγραμματισμού που επιλέχθηκε γι' αυτό το σκοπό. Είναι απαραίτητο να διδαχθούν οι κατάλληλες γλώσσες προγραμματισμού που είναι συμβατές με μικροσυστήματα. Στη συγκεκριμένη εφαρμογή επιλέχθηκαν οι γλώσσες C/C++, PYTHON (MicroPython), και LUA.

Στη συνέχεια αξιοποιήθηκαν οι τεχνολογίες διαδικτυακής επικοινωνίας έχοντας ως πυρήνα το πρωτόκολλο TCP/IP.

Τη διαχείριση/διανομή των πληροφοριών, στην παρούσα εργασία, αναλαμβάνει ο MQTT broker που αν και στη διάρκεια της μελέτης και ανάπτυξης ενεργοποιήθηκε στο φορητό υπολογιστή (laptop), εντούτοις, για λόγους που ήδη αναφέρθηκαν, θα τρέχει σε πλακέτα μικροϋπολογιστή Raspberry Pi 3, επειδή με αυτόν το τρόπο ο broker, μπορεί να διατηρηθεί σε συνεχή αδιάλειπτη λειτουργία.

Στη συνέχεια παρουσιάζεται η δομή διεπαφής, (επικοινωνία) ανθρώπου – μηχανής σε φορητές συσκευές, όπως smartphones και tablets, οργανώνοντας το αντίστοιχο γραφιστικό περιβάλλον αξιοποιώντας την τεχνολογία λογισμικού Node-Red.

Σκοπός είναι η παρούσα εργασία να εμπεριέχει έμπρακτη διδασκαλία της πληροφορικής στο πεδίο του Διαδικτύου των Πραγμάτων.

Για αυτό το λόγο εστιάζουμε στην επιλογή αισθητήρων που αλληλεπιδρούν με το περιβάλλον και μας παρέχουν τις απαραίτητες πληροφορίες των φυσικών μεγεθών με τη μορφή ηλεκτρικών σημάτων.

Παρουσιάζουμε πως αξιοποιούνται οι διατάξεις μικροελεγκτών τύπου Arduino, NodeMCU ESP8266 και ESP32 που από την έρευνά μας διαπιστώθηκε ότι έχουν τη μέγιστη εξάπλωση. Εφαρμόζουμε την τεχνολογία PUB/SUB του πρωτοκόλλου MQTT αξιοποιώντας την υλοποίηση Mosquitto. Δείχνουμε ποιες είναι οι συνιστώμενες γλώσσες προγραμματισμού για αυτό το σκοπό, καθώς επίσης και το συνιστώμενο εργαλείο Node-Red που συνδέει τη γραφιστική απεικόνιση στην οθόνη των φορητών συσκευών, με τη φυσική λειτουργία (action), των συστημάτων IoT.

© 2018

της

Άννας Μαμουζέλου

Τμήμα Μηχανικών Πληροφοριακών και Επικοινωνιακών Συστημάτων

ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ

ABSTRACT

In this diploma thesis we have chosen to use technologies, protocols and communication networks that are in direct use and will expand in the future topicality.

These technologies have been adapted to a modern home whose devices are of a conventional (old) type, they do not have an IoT interface, as will in future. So our proposals are applicable by giving the house immediate operational and value.

Although the set was made on a laptop computer under the windows, however, for reasons of economy and ecology, the Raspberry Pi 3 microcomputer, combining the advantages of very low consumption (5w) and low cost acquisition, was chosen as a permanently connected MQTT broker. These technologies have been adapted to a modern home whose devices are of a conventional (old) type, they do not have an IoT interface, as will the future. So our proposals are applicable by giving the house immediate operational and value.

Although the set was made on a laptop computer under the windows, however, for reasons of economy and ecology, the Raspberry Pi 3 microcomputer, combining the advantages of very low consumption (5w) and low cost acquisition, was chosen as a permanently connected MQTT broker.

Considering the above, this paper was designed to teach that through IoT applications we can touch the natural environment by selecting the appropriate sensor. To teach that physical size is transformed into a corresponding electrical signal and then to a numeric size that is transported in the Internet in TCP/IP packet format containing ASCII characters and is led to the interface with the final user, in pictographic form.

In order to achieve this in teaching computer literacy, a number of sensors must be presented and learned how each of them will correspond and "feel" the physical size to be explored.

That is, the physical size is transformed into an electrical signal, as is obtained at the properly organized microcontroller input, which we use with properly organized software written in the programming language chosen for this purpose. It is essential to learn the appropriate programming languages that are compatible with microsystems. In this application, C/C ++, PYTHON (MicroPython), and LUA is used.

Then, the Internet communication technologies have been exploited, having as core the TCP/IP protocol.

The management/distribution of information in the present work is undertaken by the MQTT broker which, although for the purposes of the survey, was activated on the laptop, however, for reasons already mentioned, will run on a Raspberry Pi 3 microprocessor because it the way the broker can be kept in continuous uninterrupted operation.

The human-to-machine interface for portable devices such as smartphones and tablets was then presented, organizing the corresponding graphical environment utilizing Node-Red software technology.

The aim is to present a practical teaching of informatics.

For this reason we focused on the choice of sensors that interact with the environment and provide us with the necessary information of physical sizes in the form of electrical signals.

We have shown how the microcontrollers of the Arduino AVR, NodeMCU ESP8266 and ESP32 microcontrollers are exploited, which are the most expandable in our research. We show what are the recommended programming languages for this purpose, as well as the recommended Node-Red tool that links (connects), graphic design to the portable device display with the action (physical operation) of the IOT systems.

© 2018

Anna Mamouzelou

Department of Information and Communication Systems Engineering

UNIVERSITY OF THE AEGEAN

ΕΥΧΑΡΙΣΤΙΕΣ - ΑΦΙΕΡΩΣΕΙΣ

Ευχαριστώ την καθηγήτρια μου κ. Εργίνα Καβαλλιεράτου που μου ανάθεσε την παρούσα διπλωματική εργασία καθώς και τον καθηγητή-Αναλυτή Συστημάτων κ. Χρήστο Σακαρίκα για την αμέριστη υποστήριξη και συμβουλές που μου πρόσφερε καθ' όλη τη διάρκεια της εργασίας.

Επίσης, θα ήθελα να ευχαριστήσω τις φίλες μου Ειρήνη Ζώβα και Βασιλική Δαμάλα για την βοήθεια τους.

Τέλος, θα ήθελα να εκφράσω την ευγνωμοσύνη μου στους γονείς μου Μιχάλη και Βασιλική, το σύζυγό μου Χριστόφορο και τα παιδιά μου Ανδρέα, Βασιλική και Μιχάλη για όλη τη στήριξη, τη συμπαράσταση και την κατανόησή τους, καθ' όλη τη διάρκεια των σπουδών μου.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΠΕΡΙΛΗΨΗ.....	iii
ABSTRACT.....	v
ΕΥΧΑΡΙΣΤΙΕΣ - ΑΦΙΕΡΩΣΕΙΣ.....	vii
ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ.....	viii
ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ.....	ix
ΚΕΦΑΛΑΙΟ 1 - ΕΙΣΑΓΩΓΗ.....	1
ΚΕΦΑΛΑΙΟ 2 - Συνδρομητική Εγγραφή (Subscription).....	23
ΚΕΦΑΛΑΙΟ 3 - Έκδοση Δεδομένων (Publishing).....	26
ΚΕΦΑΛΑΙΟ 4 - Το κλασικό Arduino ως εξυπηρετητής στο MQTT.....	29
ΚΕΦΑΛΑΙΟ 5 - Έκδοση και Συνδρομή MQTT με server ESP8266.....	40
ΚΕΦΑΛΑΙΟ 6 - Ο μικροΕλεγκτής ESP32.....	43
ΚΕΦΑΛΑΙΟ 7 - Μεθοδολογία Λειτουργίας.....	53
ΚΕΦΑΛΑΙΟ 8 - ΣΥΜΠΕΡΑΣΜΑΤΑ.....	55
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	56
ΠΑΡΑΡΤΗΜΑ I - Αισθητήρες.....	60
ΠΑΡΑΡΤΗΜΑ II- MQTT.....	62
ΠΑΡΑΡΤΗΜΑ III- LUA.....	66
ΠΑΡΑΡΤΗΜΑ IV- Παράρτημα κεφ.2.....	73
ΠΑΡΑΡΤΗΜΑ V- Παράρτημα κεφαλαίου 3.....	84
ΠΑΡΑΡΤΗΜΑ VI- Παράρτημα κεφαλαίου 4.....	92
ΠΑΡΑΡΤΗΜΑ VII Παράρτημα κεφαλαίου 5.....	103
ΠΑΡΑΡΤΗΜΑ VIII-Παράρτημα κεφαλαίου 6.....	117
ΠΑΡΑΡΤΗΜΑ VIII-Παράρτημα Node-Red.....	129
ΣΥΝΤΟΜΟ ΒΙΟΓΡΑΦΙΚΟ ΣΗΜΕΙΩΜΑ.....	134

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

<u>A/A</u>	<u>Τίτλος σήματος</u>	<u>σελίδα</u>
1.	Πίνακας 1.1: Τα χαρακτηριστικά του RaspberryPi3	7

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

A/A	Τίτλος σχήματος	σελίδα
2.	Σχήμα 1.1: Συνολικό έργο	1
3.	Σχήμα 1.2: τα μέρη που απαρτίζουν τις ασκήσεις	2
4.	Σχήμα 1.3: Η επικοινωνία	3
5.	Σχήμα 1.4: Η ακριβής ροή των πακέτων επικοινωνίας	4
6.	Σχήμα 1.5: Κέντρο του εσωτερικού δικτύου είναι ο δρομολογητής του διαδικτύου	5
7.	Σχήμα 1.6: Πλακέτα RaspberryPi3	6
8.	Σχήμα 1.7: Οι λειτουργίες στην πλακέτα RaspberryPi3	8
9.	Σχήμα 1.8: Πλακέτα NodeMCU ESP 8266.	
10.	Σχήμα 1.9: NodeMCU pinout	
11.	Σχήμα 1.10: Η κλασική πλακέτα Arduino με επεξεργαστή AVR των 8 bit	
12.	Σχήμα 1.11: Πλακέτα ESP8266 σε σχεδίαση τύπου Arduino	11
13.	Σχήμα 1.12: Πλακέτα ESP32 στην κλασική μορφή Arduino	12
14.	Σχήμα 1.13: Οι servers στέλνουν στον broker πακέτα πληροφορίας μέσω του δρομολογητή.	13
15.	Σχήμα 1.14: Οι συνδρομητές λαμβάνουν πακέτα πληροφορίας	14
16.	Σχήμα 1.15: Αισθητήρας θερμοκρασίας-υγρασίας DHT 11	14
17.	Σχήμα 1.16: Η σύνδεση του αισθητήρα DHT 11	15
18.	Σχήμα 1.17: Αισθητήρας ανίχνευσης αέριου νέφους	16
19.	Σχήμα 1.18: ο αισθητήρας PIR	17
20.	Σχήμα 1.19: Αισθητήρας ανίχνευσης πλημμύρας	19
21.	Σχήμα 1.20: Παράθυρο εποπτείας σειριακής πόρτας IDE	20
22.	Σχήμα 1.21: Πλακέτα 4 ρελλέ με οπτική απομόνωση	21
23.	Σχήμα 1.22: Η εικόνα περιλαμβάνει τις συσκευές εξόδου καθώς και τις εποπτεύουσες συσκευές	22
24.	Σχήμα 2.1: Μπλόκ διάγραμμα εξυπηρετητή εγγραφής	24
25.	Σχήμα 2.2: Η απεικόνιση του πίνακα ελέγχου (Dashboard) του NODE-RED	25
26.	Σχήμα 2.3: Η απεικόνιση των ροών στον συντάκτη NODE-RED	26
27.	Σχήμα 3.1: έκδοση δεδομένων με Arduino like D1 board	27
28.	Σχήμα 3.2: Απεικόνιση του εικονικού πίνακα ελέγχου	28
29.	Σχήμα 3.3: Απεικόνιση των ροών στον editor NODE-RED	29
30.	Σχήμα 4.1: Η κλασική πλακέτα Arduino UNO.	30
31.	Σχήμα 4.2: Η πλακέτα Ethernet	31
32.	Σχήμα 4.3: Αισθητήρας θερμοκρασίας και υγρασίας χαμηλού κόστους	31

33. Σχήμα 4.4: Φωτοαντίσταση για τη μέτρηση του φωτός	32
34. Σχήμα 4.5: Αισθητήρας μέτρησης της υγρασίας στο έδαφος	
35. Σχήμα 4.6: Διάταξη ρελλέ για τη διαχείριση φορτίων στην έξοδο	32
36. Σχήμα 4.7: Τα μέρη που απαρτίζουν την άσκηση	33
37. Σχήμα 4.8 Το σχέδιο σε μπλοκ	34
38. Σχήμα 4.9 Η σύνδεση της φωτοαντίστασης	35
39. Σχήμα 4.10 Arduino+Ethernet pub Server	40
40. Σχήμα 4.11: Απεικόνιση ροών βελτιωμένου πίνακα NODE-RED Dashboard	
41. Σχήμα 4.12: Βελτιωμένος πίνακας ενδείξεων που εμπεριέχει γραφιστική γραμμική απεικόνιση	41
42. Σχήμα 5.1 Απεικόνιση των μερών της άσκησης 5	
43. Σχήμα 5.2 Ο πίνακας εποπτείας και ελέγχου	42
44. Σχήμα 5.3: διάγραμμα (block) της οργάνωσης των διατάξεων για την άσκηση 5	43
45. Σχήμα 5.4 Οι ροές των κόμβων Node-Red για την υλοποίηση της άσκησης 5	44
46. Σχήμα 6.1: Η Arduino like πλακέτα με τον ESP 32.	
47. Σχήμα 6.2 : Η εσωτερική δομή του ESP32	46
48. Σχήμα 6.2.1: Σύγκριση του ESP 8266 με τον ESP 32 [Espressif]	
49. Σχήμα 6.3 : Ηλεκτρονικό διάγραμμα πλακέτας ESP 32	
50. Σχήμα 6.4 :ESP32 DEV KIT DEV BOARD	
51. Σχήμα 6.5:ESP 32 DEV BOARD BACKPLANE	
52. Σχήμα 6.6: εικονίζονται τα μέρη που συνθέτουν την άσκηση του κεφαλαίου 6	
53. Σχήμα 6.7 : Διάγραμμα σε μπλοκ εργασίας κεφαλαίου	
54. Σχήμα 6.8: Ομάδα εικονικών διακοπών που ελέγχουν οικιακές συσκευές	
55. Σχέδιο 6.9 Η πλήρης απεικόνιση του πίνακα εποπτείας και ελέγχου.	
56. Σχήμα 6.10 Οι ροές για την έκδοσης (Publish) των χειρισμών στοNODE-RED.	
57. Σχήμα 6.11 οι ροές εγγραφής (Subscription) στα επιλεγμένα θέματα με NODE-RED	
58. Σχήμα 6.12 :Η οργάνωση (layout) του πίνακα εποπτείας και ελέγχου	
59. Σχήμα 7.1 Το σύνολο της εργασίας	

Παρουσίαση του διδακτικού σκοπού της διπλωματικής εργασίας

Η παρούσα διπλωματική εργασία είναι μία διδακτική προσέγγιση η οποία συνδυάζει την μάθηση με την έρευνα τη μελέτη και την κατασκευή στο πεδίο των τεχνολογιών που υπηρετούν το Διαδίκτυο των Πραγμάτων. Οι μαθητές εμπλέκονται σε δραστηριότητες στο πλαίσιο των οποίων η μάθηση είναι ευχάριστη διαδικασία (edutainment) και σκοπός της είναι οι μαθητές να αναγνωρίσουν τους σύγχρονους μικροελεγκτές και τις δυνατότητές τους, να μελετήσουν να σχεδιάσουν να κατασκευάσουν σε διάτρητη πλακέτα (breadboard) τα αντίστοιχα κυκλώματα και να συνεργαστούν για τον προγραμματισμό της κάθε εφαρμογής. Επιδιώκουμε ολοκλήρωση των κατασκευών Arduino ώστε η λειτουργία τους να λάβει μορφή, λειτουργία και συμπεριφορά και κατά συνέπεια οπτικοποιώντας την με αποτέλεσμα την μακροχρόνια παραμονή της στην μνήμη με τρόπο δημιουργικό, ευχάριστο και μέσα σ' ένα κλίμα συνεργασίας αλληλεπιδρώντας με τους συμμαθητές τους και τον εκπαιδευτικό.

(**Λέξεις κλειδιά:** μικροελεγκτής, Arduino, edutainment, οπτικοποίηση, κλίμα συνεργασίας)

1. Εισαγωγή

Το παρόν διδακτικό έργο βασίζεται στην ιδέα της μάθησης με μελέτη και κατασκευή και στην άποψη ότι οι μαθητές μαθαίνουν καλύτερα όταν εμπλακούν με την γνώση, αλληλεπιδρώντας με αυτήν και με τους συμμαθητές τους σε κλίμα συνεργασίας και διαλόγου ανταλλάσσοντας ιδέες, απόψεις και ξεφεύγοντας από την κλασσική διδασκαλία με την χρήση μόνον βιβλίου και εισήγησης από τον καθηγητή η οποία οδηγεί σε μία δασκαλοκεντρική και παθητική διδασκαλία με τους μαθητές να γίνονται παθητικοί δέκτες και να εξαναγκάζονται στην επίπονη διαδικασία της απομνημόνευσης.

Οι κατασκευές τις οποίες καλούνται να υλοποιήσουν οι μαθητές έχουν άμεση πρακτική εφαρμογή στο πεδίο του Διαδικτύου των πραγμάτων. Οι μαθητές αρχικά μέσω συζητήσεων στην τάξη εξοικειώνονται με τις βασικές έννοιες, όπως επεξεργασία, προγραμματισμός, σχεδιασμός γλώσσα προγραμματισμού, έλεγχος λειτουργίας, βελτιστοποίηση και νιώθουν οικεία απέναντι σε αυτές και κατόπιν εμπλέκονται με λογικές και πρακτικές δραστηριότητες σε μία προσπάθεια υλοποίησης της κατασκευής και συγγραφή του προγράμματος τους καλλιεργώντας με αυτόν τον τρόπο την φαντασία τους και την δημιουργική τους σκέψη.

2. Ταυτότητα της εργασίας

2.1 Δημιουργός της εργασίας

Άννα Μαμουζέλου

2.2 Τίτλος της εργασίας

Μελέτη δομής, οργάνωσης και διασύνδεσης συστημάτων IoT και αρμονικής συνεργασίας των πρωτοκόλλων επικοινωνίας.

Πρόταση διδασκαλίας της υλοποίησης εφαρμογών με Node red

2.3 Διάρκεια Διδακτικής Ύλης

Μια εξαμηνιαία εκπαιδευτική περίοδος

2.4 Τάξη Εφαρμογής

Το διδακτικό αντικείμενο απευθύνεται σε μαθητές της Β' τάξης των Λυκείων στα πλαίσια ομαδοσυνεργατικής δραστηριότητας της διδακτικής ενότητας Εφαρμοσμένη Πληροφορική Τεχνολογία

2.5 Σκοπός & Στόχοι

2.5.1 Σκοπός της εργασίας

Ο σκοπός της παρούσας εργασίας είναι να δείξει πως μπορούν οι μαθητές του Λυκείου – ιδιαίτερα όσοι ακολουθούν Τεχνολογική κατεύθυνση- να επιλέξουν μια εφαρμογή στο πεδίο του Διαδικτύου των πραγμάτων, να συλλέξουν το απαραίτητο υλικό και τα αντίστοιχα εργαλεία λογισμικού, ώστε να υλοποιήσουν ένα πλήρες χρηστικό έργο, όπως είναι εφαρμογές εποπτείας και ελέγχου μέσω διαδικτύου αξιοποιώντας σύγχρονες τεχνολογίες Υλικού και Λογισμικού.

Στη συνέχεια με την ολοκλήρωση να αναδείξουμε τις δυνατότητες του μέσω γραφικού προγραμματισμού σε εικονικό πίνακα εποπτείας και ελέγχου.

2.5.2 Διδακτικοί Στόχοι

Γνωστικοί Στόχοι

1. Οι μαθητές να κατανοήσουν την λειτουργία ενός μικροελεγκτή.
2. Να αναγνωρίζουν αισθητήρες ανάλογα με τη χρήση τους
3. Να ονοματίζουν τα μέρη ενός ηλεκτρονικού κυκλώματος.
4. Να κατασκευάζουν ένα ηλεκτρονικό κύκλωμα.
5. Να αναγνωρίζουν τα μέρη μιας πλήρους κατασκευής (πλακέτες, αισθητήρες κτλ).
6. Να εμπλακούν στην διαδικασία συνδεσμολογίας ενός κυκλώματος.
7. Να αναπτύξουν δεξιότητες προγραμματισμού των εξυπηρετητών.
8. Να ενισχύσουν και να οικοδομήσουν την προϋπάρχουσα γνώση τους για το software και το hardware.

Κοινωνικό- Συναισθηματικοί Στόχοι

1. Να αναπτύξουν πνεύμα συνεργασίας λειτουργώντας μέσα σε ομάδα.
2. Να αυξηθεί η αυτοεκτίμηση τους.
3. Να ισχυροποιηθούν οι μεταξύ τους σχέσεις μέσα από την συνεργασία για την υλοποίηση κοινών στόχων.
4. Να καλλιεργηθεί η δημιουργικότητα και η φαντασία τους.
5. Να μάθουν ο ένας από τις γνώσεις του άλλου.

Ψυχοκινητικοί Στόχοι

1. Ανάπτυξη κινητικών δεξιοτήτων των μαθητών.
2. Βελτίωση της λεπτής κινητικότητας των μαθητών.
3. Καλλιέργεια οπτικό-κινητικού συντονισμού.

3. Μάθηση με κατασκευή – Θεωρητική προσέγγιση

3.1. Θεωρίες Μάθησης

Το παρόν διδακτικό σενάριο συνδέεται με την ιδέα του εποικοδομισμού (Piaget, 1974) και πιο συγκεκριμένα του κατασκευαστικού εποικοδομισμού όπου οι μαθητές εμπλέκονται με την νέα γνώση μέσω της συνεργατικής έρευνας, μελέτης, σχεδιασμού και κατασκευής, αλληλεπιδρούν με αυτές και «μαθαίνουν πώς να μαθαίνουν» δημιουργώντας έργα που έχουν προσωπικό νόημα για τους ίδιους (Papert, 1993). Οι μαθητές έρχονται σε επαφή με την νέα γνώση σ' ένα μαθησιακό περιβάλλον μαθητοκεντρικό όπου τους δίνει την δυνατότητα να εμπλέκονται στην διαδικασία επίλυσης προβλημάτων που σχετίζονται με τον πραγματικό κόσμο, υποστηρίζει την κοινωνική αλληλεπίδραση, ενθαρρύνει την ενεργοποίηση των μαθητών και τους δίνει κίνητρα να εμπλακούν στην μαθησιακή δραστηριότητα συμβάλλοντας έτσι στην οπτικοποίηση της νέας γνώσης και συνεπώς στην αβίαστη αποθήκευση των νέων πληροφοριών στην μακροχρόνια μνήμη τους.

3.2. Η συμβολή του Piaget-Vygotsky

Το μοντέλο του Papert (1980) και Papert (1993) οδήγησε στον σχεδιασμό εποικοδομικών περιβαλλόντων μάθησης συνδυάζοντας τις θεωρίες του Piaget και του Papert σχετικά με τον κατασκευαστικό εποικοδομισμό και την τεχνητή νοημοσύνη. Σύμφωνα με τον Piaget και τον Papert ο μαθητής συμμετέχοντας ενεργά σε μαθησιακές διαδικασίες και δρώντας γύρω από αυτές οικοδομεί την γνώση, παράγει νέα γνώση και αναδομεί την παλιά δημιουργώντας κάτι που έχει νόημα γι' αυτόν. Σύμφωνα με τις κοινωνικοπολιτισμικές θεωρίες του (Vygotsky, 1962) και του (Bruner, 1997) οι μαθητές δρουν σε συνεργατικά περιβάλλοντα μάθησης και εμπλέκονται σε συνεργατικές δραστηριότητες που αναπτύσσουν την σκέψη και που ο ρόλος του εκπαιδευτικού είναι υποστηρικτικός (scaffolding) (Harel, 1991, Nardi, 1996) και διαμεσολαβητικός. Ο εκπαιδευτικός καθοδηγεί τους μαθητές στην μαθησιακή διαδικασία όσον αφορά τα είδη

αλληλεπίδρασης με τα οποία έρχονται αντιμέτωποι και στην μάθηση. Ο συνδυασμός της εποικοδομικής θεωρίας του Piaget και τις κοινωνικοπολιτισμικής θεωρίας του Vygotsky οδηγεί στην επιλογή για σχεδίαση εκπαιδευτικού λογισμικού όπου δίνει την ευκαιρία στους μαθητές για οικοδόμηση των γνώσεων τους ατομικά (individualistic construction) και κοινωνικής αλληλεπίδρασης μέσω συνεργατικών δραστηριοτήτων.

3.3 Εποικοδομική προσέγγιση

Ο εποικοδομισμός σύμφωνα με τον Piaget είναι η οικοδόμηση νέων γνώσεων πάνω σε ήδη υπάρχουσες γνώσεις, εμπειρίες και η αναδόμηση των ήδη υπάρχουσών. Στόχος είναι η δημιουργία περιβαλλόντων μάθησης όπου τα παιδιά θα εμπλέκονται με αντικείμενα και κατασκευές που έχουν νόημα γι' αυτά. Ο μαθητής μαθαίνει φτιάχνοντας (learning by making), ενθαρρύνεται γι' αυτό και δημιουργεί μέσω της κατασκευής την έννοια του εποικοδομισμού που πηγάζει από τις σχετικές θεωρίες. Η μάθηση γίνεται πιο αποτελεσματική και οι μαθητές μαθαίνουν να κάνουν συλλογισμούς με τρόπο φυσικό και αβίαστα σε αντίθεση με την παραδοσιακή εκπαίδευση. Ο εποικοδομισμός συνδέεται με την έννοια του brocolage (μαστόρεμα) σύμφωνα με το οποίο ο μαθητής καθοδηγείται μόνος του από την πορεία της δουλειάς του και επαναδιαπραγματεύεται τον σχεδιασμό του κάθε φορά μέχρι να φτάσει εκεί που θέλει χωρίς να προσκολλάται σ' έναν αρχικό σχεδιασμό. Οι μαθητές καθώς εμπλέκονται με την κατασκευή παράγουν νέες ιδέες, τις μοιράζονται με τους συμμαθητές τους και κάθε φορά τις επαναδιαπραγματεύονται, τις διαμορφώνουν και με αυτόν τον τρόπο οικοδομούν την γνώση φτιάχνοντας κατασκευές με προσωπικό νόημα (kafai & Resnick, 1996, Constructionism in practice). Οι μαθητές μαθαίνουν ανακαλύπτοντας καθώς κατασκευάζουν. Σύμφωνα με τον Piaget η μάθηση μέσω ανακάλυψης έχει πολλαπλά πλεονεκτήματα αφού αυξάνει τις διανοητικές ικανότητες, συμβάλλει στην καλλιέργεια της σκέψης, της συλλογιστικής και δημιουργικής σκέψης, τις αποτελεσματικότητας της κατανόησης και της ελαχιστοποίησης της αποστήθισης.

4. Διδακτική Μεθοδολογία

4.1. Ομαδοσυνεργατική διδασκαλία μέσω της έρευνας μελέτης σχεδιασμού και κατασκευής

Η ομαδοσυνεργατική διδασκαλία συνδέεται άμεσα με την «ζώνη επικείμενης ανάπτυξης» του Vygotsky (1978) όπου τόνισε ότι είναι το κενό ανάμεσα στο τι μπορεί να μάθει ένα παιδί μόνο του και του τι μπορεί να μάθει με την βοήθεια κάποιου μεγαλύτερου και πιο έμπειρου. Τα συνεργατικά περιβάλλοντα μάθησης δίνουν την ευκαιρία σε μαθητές και εκπαιδευτικούς να εμπλακούν (Bruner, 1986), να επικοινωνήσουν και γενικότερα να αλληλεπιδράσουν με κύριο σκοπό την κατανόηση (Edwards & Mercer (1987)) η οποία μαζί με την μάθηση έχει κοινωνικές, πολιτιστικές και επικοινωνιακές διαστάσεις. Οι ομαδοσυνεργατικές δραστηριότητες παίζουν

σημαντικό ρόλο στην επίλυση προβλημάτων, στην κατανόηση εννοιών και συνδέονται με την κοινωνικοποίηση των μαθητών, την αύξηση της παραγωγικότητας τους και την ενίσχυση της αυτοεκτίμησης τους.

Ως τέτοια δραστηριότητα διδάσκεται στους μαθητές, να πειραματιστούν, να εκφραστούν (Resnick & Silverman 2005) και να κατανοήσουν τις βασικές αρχές προγραμματισμού και συνδεσμολογίας του υλικού. Για την κατασκευή χρησιμοποιήθηκαν σαν τεχνολογικά εργαλεία Arduino λογισμικό και συμβατό υλικό που αποτελείται από σύγχρονους ισχυρούς, αλλά φτηνούς και προσιτούς επεξεργαστές και αισθητήρες τα οποία παρέχουν ένα εύχρηστο περιβάλλον προγραμματισμού δίνοντας κίνητρο στους μαθητές να εκφράσουν την δημιουργικότητα τους με εύκολο τρόπο, να εξερευνήσουν και να εμπλακούν ενεργά τόσο στην σχεδίαση όσο και στην κατασκευή οικοδομώντας πιο αποτελεσματικά τις γνώσεις τους (Papert, 1991). Οι μαθητές αλληλεπιδρούν με οντότητες/αντικείμενα και δίνουν σε αυτά συμπεριφορές τις οποίες τις παρατηρούν ζωντανά να συμβαίνουν και δεν βλέπουν απλά τιμές στην οθόνη του ηλεκτρονικού υπολογιστή. Στην προσπάθεια τους να υλοποιήσουν μία προγραμματιζόμενη κατασκευή εμπλέκονται σε τέσσερα στάδια εργασίας τα οποία μπορούν να επαναλαμβάνονται κυκλικά ή να επικαλύπτονται.

Στάδιο εμπλοκής: Σχετίζεται με την διατύπωση του προβλήματος και τον καταϊγισμό ιδεών και συζήτηση από τους μαθητές για την μορφή της κατασκευής τους και για τις συμπεριφορές της.

Στάδιο πειραματισμού: Οι μαθητές πειραματίζονται με επεξεργαστές και λογισμικό με στόχο να μάθουν και να εξοικειωθούν με τον τρόπο λειτουργίας τους.

Στάδιο διερεύνησης: Οι μαθητές συζητούν ξανά και επαναπροσδιορίζουν τις ιδέες τους με βάση τις εμπειρίες τους από το προηγούμενο στάδιο.

Στάδιο σύνθεσης και δημιουργίας: Οι μαθητές ξεκινούν την υλοποίηση της κατασκευής τους συνθέτοντας τα επιμέρους υλικά και κυκλώματα για να δώσουν την τελική μορφή σε αυτήν. Ακολούθως αρχίζουν με κλιμακωτή ανάπτυξη τον προγραμματισμό τους.

Στάδιο αξιολόγησης: Οι μαθητές αξιολογούν την δουλειά τους εκφράζοντας τις απόψεις τους σχετικά με την επίτευξη των προσδοκώμενων στόχων, συζητούν και στέκονται κριτικά απέναντι στην κατασκευή τους.

Στις μέρες μας οι δραστηριότητες μοντελοποίησης κερδίζουν έδαφος στα σχολεία της Πρωτοβάθμιας και Δευτεροβάθμιας Εκπαίδευσης, η παιδαγωγική αξία τους είναι μεγάλη (Κυνηγός & Φράγκου, 2000) και αυξάνουν τα κίνητρα για μάθηση των Τεχνολογικών (Lawhead et al.) και των Φυσικών επιστημών. Τα επιλεγμένα περιβάλλοντα λογισμικού συνδυάζουν την θεωρία του Piaget (εποικοδομική θεωρία) και του Vygotsky (κοινωνικοπολιτισμική θεωρία) και είναι μία καλή επιλογή για την υλοποίηση παιδαγωγικών και διδακτικών δραστηριοτήτων.

4.2. Οι Τεχνολογίες του Διαδικτύου των Πραγμάτων

Η επιλογή του κατάλληλου εκπαιδευτικού υλικού με το οποίο οι μαθητές θα εμπλακούν σε δραστηριότητες/κατασκευές εξαρτώνται από τις προσωπικές προτιμήσεις του μαθητή, του εκπαιδευτικού, την ηλικία των μαθητών αλλά και τους εκπαιδευτικούς στόχους που θα πρέπει να επιτευχθούν κάθε φορά. Η παραπάνω σειρά των ασκήσεων IoT επιλέχθηκε με σκοπό την δημιουργία ενός περιβάλλοντος κοινωνικής αλληλεπίδρασης και εποικοδομητικής μάθησης (constructive learning) όπου οι μαθητές εμπλέκονται στην επίλυση προβλημάτων ανοιχτών όπου η υλοποίηση τους μπορεί να προσεγγιστεί με διάφορους τρόπους και συνδέονται άμεσα με πραγματικά προβλήματα όπως η δημιουργία μίας αυτοματοποιημένης οντότητας (λειτουργίας) για την απομακρυσμένη εποπτεία και των έλεγχο εργαστηρίων, κατοικιών κλπ. Οι σύγχρονες κατασκευές στο πεδίο των IoT συνδέονται με το «μαστόρεμα της γνώσης»(constructionism) καθώς σύμφωνα και με τον Papert οι μαθητές κατασκευάζουν την γνώση συμμετέχοντας στην κατασκευή είτε χειρωνακτικά είτε προγραμματιστικά με αποτέλεσμα να την οικοδομούν αποτελεσματικότερα.

4.3. Μειονεκτήματα των IoT

Υψηλό πεδίο τεχνολογικής κατάρτισης.

Απαιτείται υψηλή τεχνολογική κατάρτιση και διδακτική εμπειρία, καθώς επίσης αντίστοιχους πάγκους εργασίας που θα εξοπλίζονται με υπολογιστή. Απαιτεί μαθητική τάξη με υψηλή προσήλωση.

Χρονικοί περιορισμοί: Οι δραστηριότητες IoT απαιτούν χρόνο ο οποίος δεν καλύπτεται από τα ωρολόγια προγράμματα. Ο χρόνος σχετίζεται με την εξοικείωση των μαθητών με τα ηλεκτρονικά μέρη και τα προγράμματα-εργαλεία, την οργάνωση της διδασκαλίας, τα φύλλα εργασίας, το υποστηρικτικό υλικό, την οργάνωση του χώρου του εργαστηρίου και σαφώς με τον προσωπικό ρυθμό μάθησης του κάθε μαθητή.

Φυσικοί περιορισμοί υλικού: Σχετίζονται με τις εκπομπές ακτινοβολιών, την ύπαρξη δικτύου και ελεύθερου χώρου δοκιμών οι οποίες δεν έχουν πάντα την επιθυμητή ακρίβεια καθώς και με την πηγή ενέργειας σε ρεύμα (μπαταρία) που πρέπει να ανανεώνεται συχνά.

4.4. Ψυχαγωγική εκπαίδευση (edutainment) και IoT

Η ψυχαγωγική εκπαίδευση (edutainment) συνδυάζει το παιχνίδι με την μάθηση με σκοπό η μάθηση να γίνει μία διασκεδαστική εμπειρία και κατά συνέπεια αποτελεσματικότερη και ευκολότερη (Lund & Nielsen, 2002). Ο μαθητής εμπλέκεται με δραστηριότητες κατά τις οποίες αλληλεπιδρά είτε με τον ηλεκτρονικό υπολογιστή είτε με κάποια συσκευή (τεχνούργημα) όπως για παράδειγμα μ' ένα σύστημα IoT με σκοπό το δημιουργήμα του να του προσφέρει κάτι, όπως για παράδειγμα μία ηθική ικανοποίηση ή ένα βραβείο. Αυτή ακριβώς η προσφορά είναι και το κίνητρο για την εμπλοκή του στις δραστηριότητες. Το edutainment συνδέεται με το βίωμα αφού ο μαθητής βιωματικά κατανοεί τις διδασκόμενες έννοιες και παράγει νέα γνώση. Η εκπαιδευτική

διαδικασία στις σύγχρονες εφαρμοσμένες τεχνολογίες, συνδυάζει την εκπαίδευση με την μάθηση κάνοντας την πιο διασκεδαστική και προσιτή στους μαθητές καθώς δίνει κίνητρο στους μαθητές να εμπλακούν ενεργά στην επίλυση αυθεντικών προβλημάτων καλλιεργώντας την δημιουργική τους σκέψη, την φαντασία και παίρνοντας άμεση ανατροφοδότηση. Οι μαθητές διερευνούν, ανακαλύπτουν και παράγουν καινοτόμες ιδέες έχοντας ως σύμμαχο τους την τεχνολογία και τα εργαλεία που αυτή τους παρέχει. Η εκπαιδευτική διαδικασία εμπλέκει τους μαθητές σε διαθεματικές δραστηριότητες αφού συνδυάζει διάφορες επιστήμες και υποστηρίζει την βιωματική μάθηση έτσι ώστε οι μαθητές να μαθαίνουν μέσα από τις εμπειρίες τους καθώς εμπλέκονται σε δραστηριότητες / κατασκευές οι οποίες αρκετές φορές προσομοιώνουν συμπεριφορές που εμφανίζονται στην καθημερινή ζωή. Ολοκληρώνοντας, πρέπει να αναφέρουμε την κοινωνική διάσταση της εκπαιδευτικής διαδικασίας σε χρηστικές εφαρμογές, αφού ενθαρρύνεται η επικοινωνία, ο διάλογος, η ανταλλαγή ιδεών και η αντιμετώπιση συγκρούσεων στα πλαίσια της συνεργασίας.

4.5. Οργάνωση Διδασκαλίας

Το σενάριο έχει σχεδιαστεί με τέτοιο τρόπο ώστε να μπορεί να υλοποιηθεί στο εργαστήριο πληροφορικής του σχολείου. Για την υλοποίηση των δραστηριοτήτων οι μαθητές δουλεύουν άλλοτε ατομικά όπου ο κάθε μαθητής κάθεται στον δικό του υπολογιστή και άλλοτε σε ομάδες 2 ατόμων. Το σύνολο των μαθητών είναι οι παρόντες στην τάξη. Οι υπολογιστές του εργαστηρίου είναι 10 σε σχήμα «Π» και όλοι οι μαθητές έχουν οπτική επαφή με τον κλασσικό πίνακα και τον διαδραστικό πίνακα του εργαστηρίου. Για να μπορεί να υλοποιηθεί το διδακτικό σενάριο είναι υποχρεωτική η ύπαρξη πίνακα, μαρκαδόρου, προτζέκτορα, διαδραστικού πίνακα για την παρουσίαση βίντεο και δραστηριοτήτων καθώς και η εγκατάσταση των λογισμικού για τον προγραμματισμό των Arduino και των παρουσιάσεων για τη διδασκαλία της γλώσσας προγραμματισμού, κατασκευής διαγραμμάτων και η ύπαρξη Internet μέσω του Πανελληνίου Σχολικού Δικτύου.

4.6. Ο ρόλος του Εκπαιδευτικού

Ο εκπαιδευτικός καθοδηγεί όλη τη μαθησιακή διαδικασία και έχει υποστηρικτικό και διαμεσολαβητικό ρόλο, βοηθάει τους μαθητές του να βρουν το ρυθμό τους, τους δίνει κίνητρα και τους παρέχει ενίσχυση όπου χρειάζεται, ενισχύει την συνεργασία και βοηθάει στην διαχείριση της ομάδας. Ο εκπαιδευτικός πρέπει να προετοιμάζεται για να είναι όσο πιο αποτελεσματικός μπορεί σχεδιάζοντας το υλικό του, έντυπο ή ψηφιακό/οπτικοακουστικό υλικό με τις διδασκόμενες έννοιες παρέχοντας βοήθεια στον μαθητή, ώστε να δουλέψει πάνω σε αυτό το υλικό με στόχο την απομνημόνευση των διδασκόμενων εννοιών στην μακροχρόνια μνήμη. Η διατήρηση των πληροφοριών στην μνήμη για μεγάλο χρονικό διάστημα μπορεί να γίνει με την

χρήση βίντεο, εικόνας και γενικότερα οπτικοακουστικού υλικού το οποίο βοηθάει και αναπτύσσει τον αισθητηριακό κόσμο των μαθητών.

4.7. Εκτιμώμενη Διάρκεια

Το διδακτικό έργο σχεδιάστηκε για να υλοποιηθεί σε εξαμηνιαία βάση. Ο χρόνος υλοποίησης του μπορεί να είναι μεταβλητός αφού εξαρτάται από το γνωστικό επίπεδο των μαθητών που εφαρμόζεται κάθε φορά, τον χρόνο και τον ρυθμό που χρειάζεται για να κατανοήσουν τις διδασκόμενες έννοιες και να υλοποιήσουν τις δραστηριότητες αλλά και από τις μεταξύ τους σχέσεις όσον αφορά στην ικανότητα τους να συνεργάζονται, να δημιουργούν ομάδες και να συζητούν στα πλαίσια ενός κλίματος διαλόγου. Σημαντικό είναι να αναφερθεί ότι εξαιτίας της κούρασης υπάρχει χαμένος χρόνος διδασκαλίας από την διάσπαση προσοχής των παιδιών με αποτέλεσμα, η διδακτική ώρα να τελειώσει χωρίς να έχουν επιτευχθεί όλοι οι προβλεπόμενοι στόχοι. Τέλος, εκτός από τον γνωστό θόρυβο της τάξης (διάσπαση προσοχής, φασαρία, άσχετες συζητήσεις κτλ) ας μην ξεχνάμε και τα τεχνικά προβλήματα που εμφανίζονται και σχετίζονται με το υλικό, λογισμικό και την σύνδεση στο διαδίκτυο που απαιτούνται για την υλοποίηση του σεναρίου. Σε όλα τα παραπάνω μπορεί να οφείλεται η αύξηση των διδακτικών ωρών για την υλοποίηση του σεναρίου. Όλα τα παραπάνω είναι θέματα που πρέπει να απασχολούν τον εκπαιδευτικό κατά τον αναστοχασμό του στο τέλος κάθε διδακτικής ενότητας.

4.8. Πρόβλεψη δυσκολιών στο Διδακτικό έργο

- 1 Οι δυσκολίες στην κατανόηση εννοιών και δομών
- 2 Οι δυσκολίες στην εξοικείωση με τον εξοπλισμό
- 3 Οι δυσκολίες στον σχεδιασμό της κατασκευής
- 4 Οι δυσκολίες στην σύνδεση του εξοπλισμού
- 5 Οι δυσκολίες στον προγραμματισμό των συμπεριφορών

5. Ανάπτυξη του διδακτικού έργου

5.1 Το Επιστημονικό Περιεχόμενο

Το διδακτικό έργο είναι βασισμένο στην μάθηση μέσα από την κατασκευή και στοχεύει στην ενεργή συμμετοχή των μαθητών και την εμπλοκή τους με την γνώση με σκοπό την παραγωγή νέας γνώσης και την ένταξη τους μέσα σε ένα κοινωνικό σύνολο με συγκεκριμένους ρόλους καθώς εμπλέκονται με ομαδοσυνεργατικές δραστηριότητες για την επίτευξη του έργου τους.

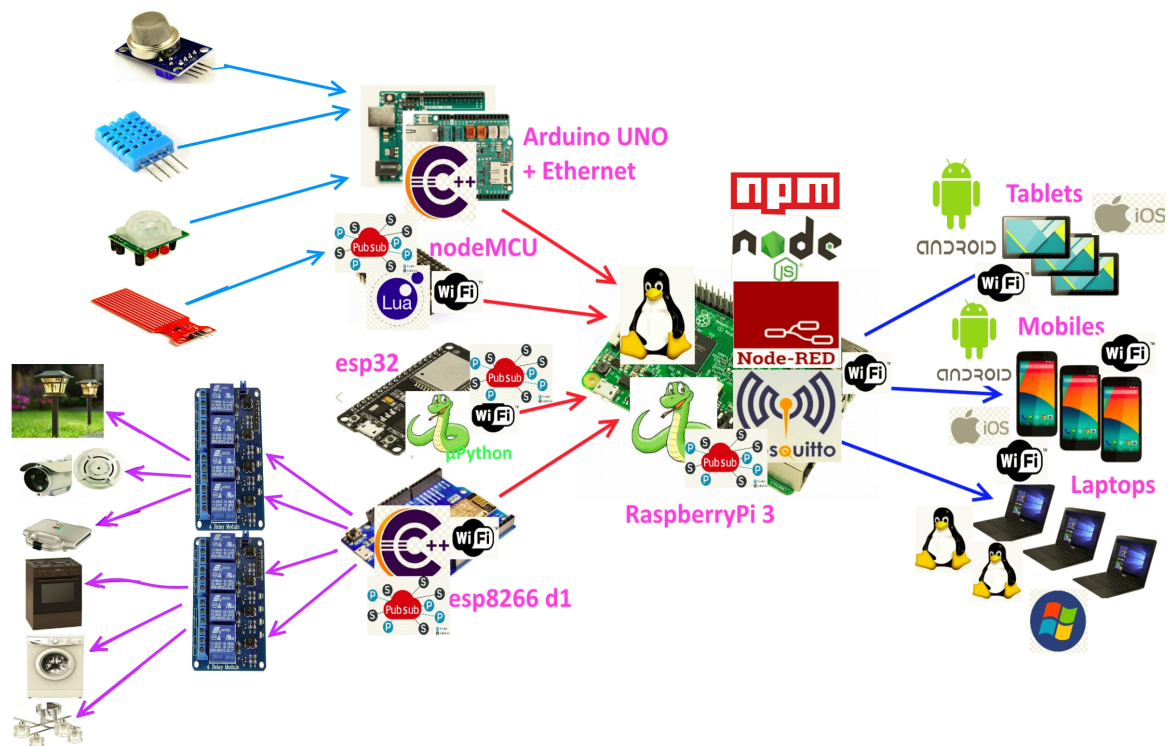
ΚΕΦΑΛΑΙΟ 1 - ΕΙΣΑΓΩΓΗ

Ο σκοπός, η δομή, ο εξοπλισμός και τα προγράμματα της εργασίας μας

Στη παρούσα εργασία έγινε προσπάθεια να έχει τεχνολογικό αλλά και παιδαγωγικό χαρακτήρα.

Θα επιχειρήσουμε να δείξουμε το πώς και το γιατί, την ανάγκη και τη λύση, το ερώτημα και την απάντηση στο πεδίο του Διαδικτύου των πραγμάτων.

Το τελικό αποτέλεσμα θα φτάσει να εικονίζεται στο ακόλουθο σχήμα



Σχήμα 1.1: Συνολικό έργο

Στην παρούσα εργασία επιδιώξαμε τη μελέτη, το σχεδιασμό και την υλοποίηση ενός πλήρους συστήματος με αναφορά στο Διαδίκτυο των Πραγμάτων.

Πέραν της βασικής λειτουργικότητας θελήσαμε να εμπεριέχει διδακτικό περιεχόμενο που θα διευκολύνει στη Διδακτική της Σύγχρονης Τεχνολογίας.

Οι βασικές δομικές μονάδες είναι ο Broker, ο Server, οι Clients και ο Router (δρομολογητής) που τις διασυνδέει μέσω ασύρματου δικτύου WiFi ή ενσύρματου δικτύου Ethernet.

Ο broker MQTT υλοποιείται με την εφαρμογή mosquitto που τρέχει σε υπολογιστική μονάδα RaspberryPi 3. Αυτή με τη σειρά της, συνδέεται σε Internet Router που τρέχει DHCP για να εκχωρεί IP διευθύνσεις στο ιδιωτικό υποδίκτυο 192.168.1.0 - 192.168.1.255.

Ο Server υλοποιείται σε μικροπολογιστικό σύστημα αποτελούμενο από πλακέτα nodeMCU ESP 8266 ή Arduino D1 ή ESP 32 ή Arduino Uno με διεπαφή Ethernet και οι εφαρμογές που τρέχουν σε αυτούς είναι γραμμένες κατ' αρχάς σε C/C++ με την προσπάθεια κάποιες να μεταφερθούν σε microPython και άλλες σε LUA.(μελλοντικές βελτιώσεις)

Οι Clients είναι μικροπολογιστικά συστήματα συνήθως smartphones ή tablets (χωρίς να αποκλείεται κανονικό PC με λειτουργικό Windows ή Linux), με δυνατότητα γραφιστικής απεικόνισης σε οθόνη LCD ή άλλου τύπου.

Η γραφιστική διεπαφή υλοποιείται με την εφαρμογή Node-Red που τρέχει στο RaspberryPi 3. Τα ανωτέρω συνδυαζόμενα μας επιτρέπουν την εποπτεία και τον έλεγχο των ηλεκτρικών συσκευών μιας σύγχρονης κατοικίας. Δεδομένου ότι οι μέχρι τώρα λειτουργούσες συσκευές είναι συμβατικής τεχνολογίας, δηλαδή δε διαθέτουν διεπαφή (interface) IoT, θεωρήσαμε πως θα ήταν προτιμότερο να τις προσαρμόσουμε στην εφαρμογή μας σχεδιάζοντας ενδιάμεση διεπαφή ικανή να διαχειρίζεται ισχύ μέσω ηλεκτρονόμων (ρελλέ). Όλα απεικονίζονται στο σχήμα 1-2.



Σχήμα 1.2: τα μέρη που απαρτίζουν τις ασκήσεις

Στο παραπάνω σχήμα φαίνονται οι συσκευές που επιλέχτηκαν για να αξιοποιηθούν στην παρούσα εργασία. Πυρήνας των λειτουργιών είναι ο μικροϋπολογιστής RaspberryPi3. Ως

εξυπηρετητές εισόδου και εξόδου θα αξιοποιηθούν πλακέτες μικροϋπολογιστών που οι μαθητές μπορούν να προμηθευτούν με ευκολία. Αυτό προέκυψε από την έρευνά μας, επειδή διαπιστώσαμε ότι οι διατάξεις αυτές είναι οι ευρύτερα αξιοποιούμενες.

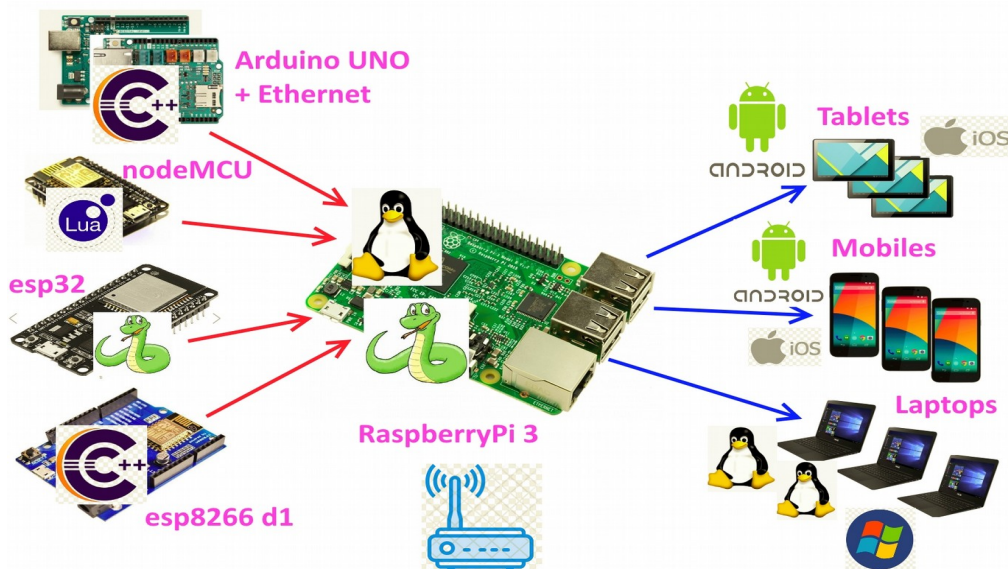
Το τρίτο σκέλος της εργασίας περιλαμβάνει τις συσκευές επικοινωνίας, όπως tablets, smartphones και laptops.

Όλες οι συσκευές εισόδου (εξυπηρετητές) εξασφαλίζουν την πληροφορία και την εκδίδουν στο broker. Κάθε πληροφορία διακρίνεται από τις υπόλοιπες επειδή έχει διαφορετική θέση προσωρινής διαχείρισης στη μνήμη του μεσίτη. Τη θέση αυτή την ονομάζουμε θέμα (Topic). Τα θέματα τα δημιουργεί μόνον ο κάθε εξυπηρετητής εισόδου που θέλει να στείλει πληροφορίες και όχι ο εξυπηρετητής συνδρομής που θέλει να τις λάβει.

Κάθε πελάτης που επιθυμεί να παίρνει τις πληροφορίες για ένα θέμα, (ή και περισσότερα θέματα), πρέπει να εγγράψει συνδρομή σε αυτό. Μόλις ο μεσίτης δεχθεί (παραλάβει) τη νέα πληροφορία για το συγκεκριμένο θέμα, τότε την εκπέμπει προς τον συνδρομητή που τη χρειάζεται. Μπορεί να είναι ένας ή περισσότεροι οι παραλήπτες της πληροφορίας, εφόσον όλοι έχουν εγγράψει σε αυτό συνδρομή.

Πολλοί εκδότες μπορούν να δηλώνουν τα ξεχωριστά θέματά τους ο καθένας και να τα παραλαμβάνουν ένας ή περισσότεροι πελάτες.

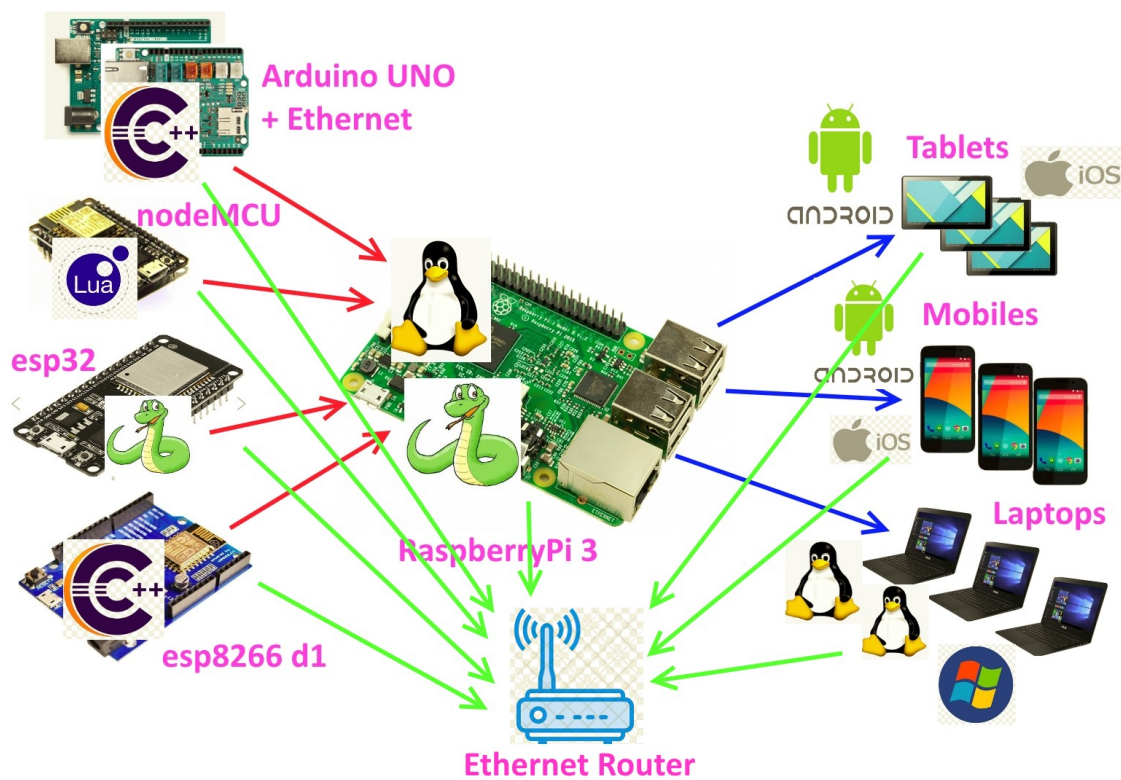
Η λειτουργία Pub/Sub, δηλαδή έκδοση/συνδρομή σε πληροφορία θεμάτων από πολλούς εξυπηρετητές προς πολλούς πελάτες εικονίζεται στο παρακάτω σχήμα.



Σχήμα 1.3: Η επικοινωνία

Η μεσολάβηση του δικτύου.

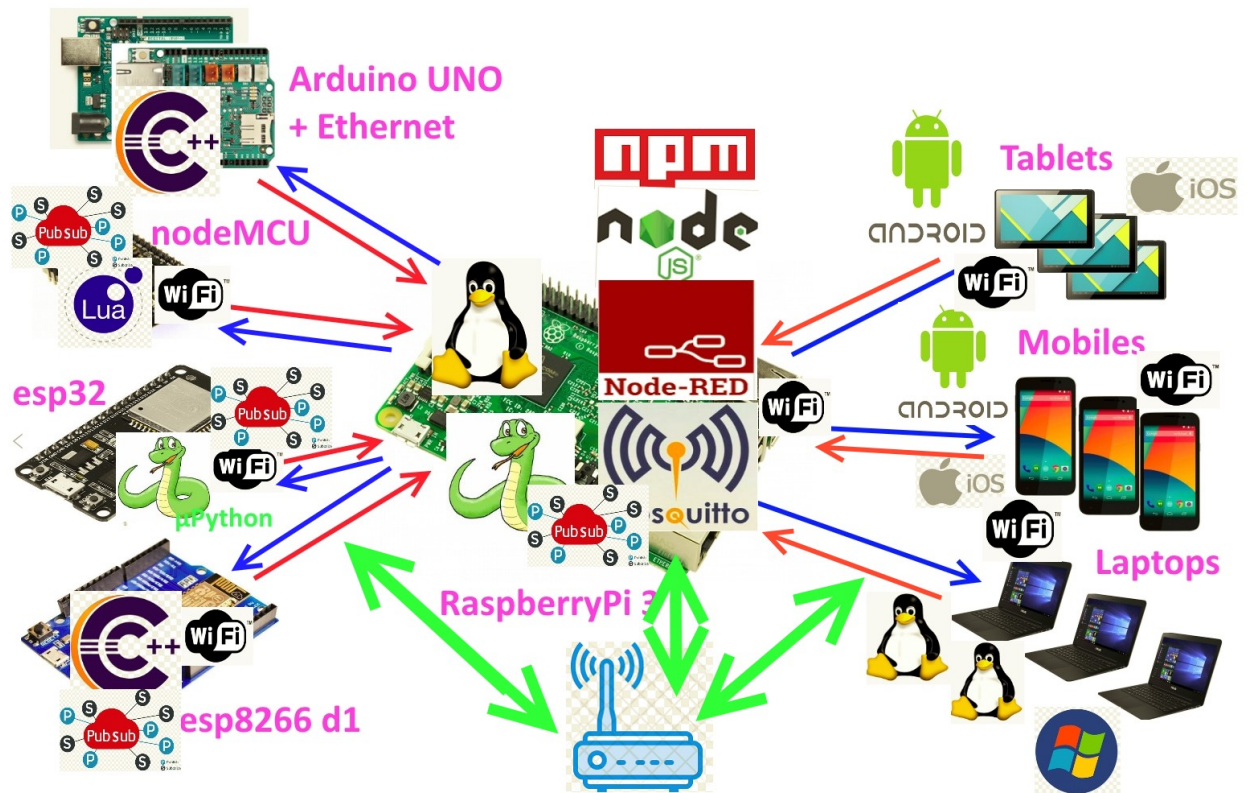
Είναι προφανής η ανάγκη ύπαρξης δικτύου IP στο οποίο να συνδέονται και να επικοινωνούν οι διατάξεις μεταξύ τους. Τον πυρήνα του δικτύου μας, (και κάθε άλλου ιδιωτικού δικτύου στην περιοχή 192.168.X.X), αποτελεί ο δρομολογητής του διαδικτύου(Internet Router). Όλες οι διατάξεις συνδέονται με τον πυρήνα του δικτύου. Η εφαρμογή DHCP που τρέχει μόνιμα στο δρομολογητή, εκχωρεί διεύθυνση IP σε κάθε υπολογιστική μονάδα που θα συνδεθεί ασύρματα ή ενσύρματα σε αυτόν. Η πραγματική πορεία της επικοινωνίας φαίνεται στο σχήμα που ακολουθεί με πράσινα βέλη.



Σχήμα 1.4: Η ακριβής ροή των πακέτων επικοινωνίας

Στο σχήμα 1-4 απεικονίζονται επίσης, οι γλώσσες προγραμματισμού και τα λειτουργικά συστήματα που αξιοποιούνται στην εφαρμογή μας.

Στο ακόλουθο σχήμα 1-5 δείχνουμε με τα διπλά βέλη τον τρόπο επικοινωνίας του συνόλου των συσκευών μέσω του δρομολογητή του δικτύου.



Σχήμα 1.5: Κέντρο του εσωτερικού δικτύου είναι ο δρομολογητής του διαδικτύου

Στο αριστερό μέρος (εξυπηρετητές), αξιοποιούνται κυρίως οι γλώσσες προγραμματισμού C και C++ κάτω από την εφαρμογή Arduino IDE.

Η microPython, υποσύνολο της γλώσσας Python έχει επίσης κατά την άποψή μας ευρύτατο μέλλον στις εφαρμογές IoT, επειδή, εκτός από την απλότητά της, είναι εντεταγμένη και διδάσκεται στους μαθητές του Λυκείου.

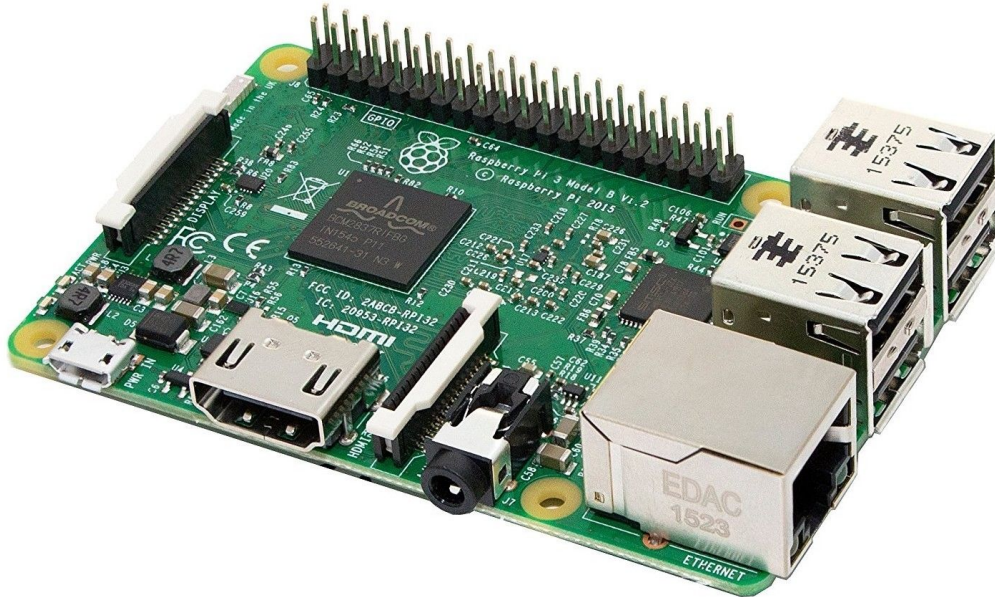
Θεωρήσαμε πως εφαρμογές της στο πεδίο των IoT, όπως είναι οι αυτοματισμοί και η ρομποτική, θα δείξουν άμεσα τη χρησιμότητά της.

Τέλος, όταν αρχίσαμε την έρευνά μας στο τι θα ήταν χρήσιμο να διδαχθεί στο πεδίο της πληροφορικής στη μέση εκπαίδευση, διαπιστώσαμε πως η γλώσσα προγραμματισμού LUA (σελήνη) αν και στο διεθνή χώρο είναι πολύ γνωστή στη χώρα μας είναι σχεδόν άγνωστη.

Αφορμή μας έδωσε η πλακέτα NodeMCU ESP8266 που ενσωματώνει μεταφραστή LUA και λόγω της απλότητάς της κάνει τη δημιουργία εφαρμογών, εύκολη και γρήγορη.

Ο broker

Ο broker όπως αναφέρθηκε τρέχει σε πλακέτα RaspberryPi 3 που φαίνεται στο σχήμα 5



Σχήμα 11.6: Πλακέτα RaspberryPi3

Το συγκεκριμένο RaspberryPi3 είναι εξοπλισμένο με τετραπύρηνο επεξεργαστή ARM (V7, 64 bit) που τρέχει στα 1.2 GHz, διαθέτει μνήμη RAM 1GB και τις απαραίτητες διεπαφές, (interfaces), για σύνδεση πληκτρολογίου, οθόνης κλπ. Ως λειτουργικό σύστημα τρέχει LINUX(έκδοση Jessie). Κάτω από το λειτουργικό LINUX για την εφαρμογή μας ενεργοποιούμε τον MQTT Broker στη ελαφρά υλοποίηση ελεύθερου λογισμικού Mosquitto. Επίσης αξιοποιούμε το Node-Red για την υλοποίηση του γραφιστικού περιβάλλοντος εποπτείας και ελέγχου των εφαρμογών μας.

Προφανώς για να λειτουργήσει εκτός της κύριας πλακέτας απαιτούνται:

4. Οθόνη
5. Τροφοδοτικό 5V/3A.
6. HDMI to VGA διεπαφή
7. Πληκτρολόγιο
8. Ποντίκι

Ο πίνακας που ακολουθεί περιλαμβάνει τα χαρακτηριστικά του RaspberryPi3.

	Raspberry Pi 3 Model B	Raspberry Pi 2 Model B
Processor Chipset	Broadcom BCM2837 64Bit Quad Core Processor powered Single Board Computer running at 1.2GHz	Broadcom BCM2836 32Bit Quad Core Processor powered Single Board Computer running at 900MHz
GPU	Videocore IV	Videocore IV
Processor Speed	QUAD Core @1.2 GHz	QUAD Core @900 MHz
RAM	1GB SDRAM @ 400 MHz	1GB SDRAM @ 400 MHz
Storage	MicroSD	MicroSD
USB 2.0	4x USB Ports	4x USB Ports
Max Power Draw/voltage	2.5A @ 5V	1.8A @ 5V
GPIO	40 pin	40 pin
Ethernet Port	Yes	Yes
WiFi	Built in	No
Bluetooth LE	Built in	No

Πίνακας 1.1: Τα χαρακτηριστικά του RaspberryPi3

Παρουσίαση και Ανάλυση Raspberry Pi 3

Θα εστιάσουμε στο **Raspberry Pi 3** αιτιολογώντας πρώτα από όλα, την επιλογή του ως broker Host και στη συνέχεια στις εφαρμογές που τρέχουν σε αυτό.

Ο εξυπηρετητής του μεσίτη πρέπει να τρέχει διαρκώς και αδιάλειπτα. Πρέπει δηλαδή να είναι απλός, φτηνός και να καταναλώνει μικρά ποσά ενέργειας. Πρέπει όμως να διαθέτει την υπολογιστική ισχύ για να τρέχει τις απαραίτητες εφαρμογές. Το **Raspberry Pi 3** καλύπτει όλα τα παραπάνω χωρίς να έχει ανάγκη ιδιαίτερης ψυκτικής μονάδας. Όταν ρυθμιστεί στις απαραίτητες λειτουργίες του, το μόνο που απαιτεί είναι η τροφοδοσία των 5-6 W περίπου. Όλα τα άλλα μπορούν να αφαιρεθούν.

Έτσι διαθέτουμε έναν αξιοπρεπή Broker Server που τρέχει mosquitto, κάτω από μια σταθερή έκδοση Linux! {όλα τα προγράμματα δωρεάν}.

Για να εγκατασταθούν οι εφαρμογές απαιτείται ο «φορτωτής» **npm (node-package-manager)**.

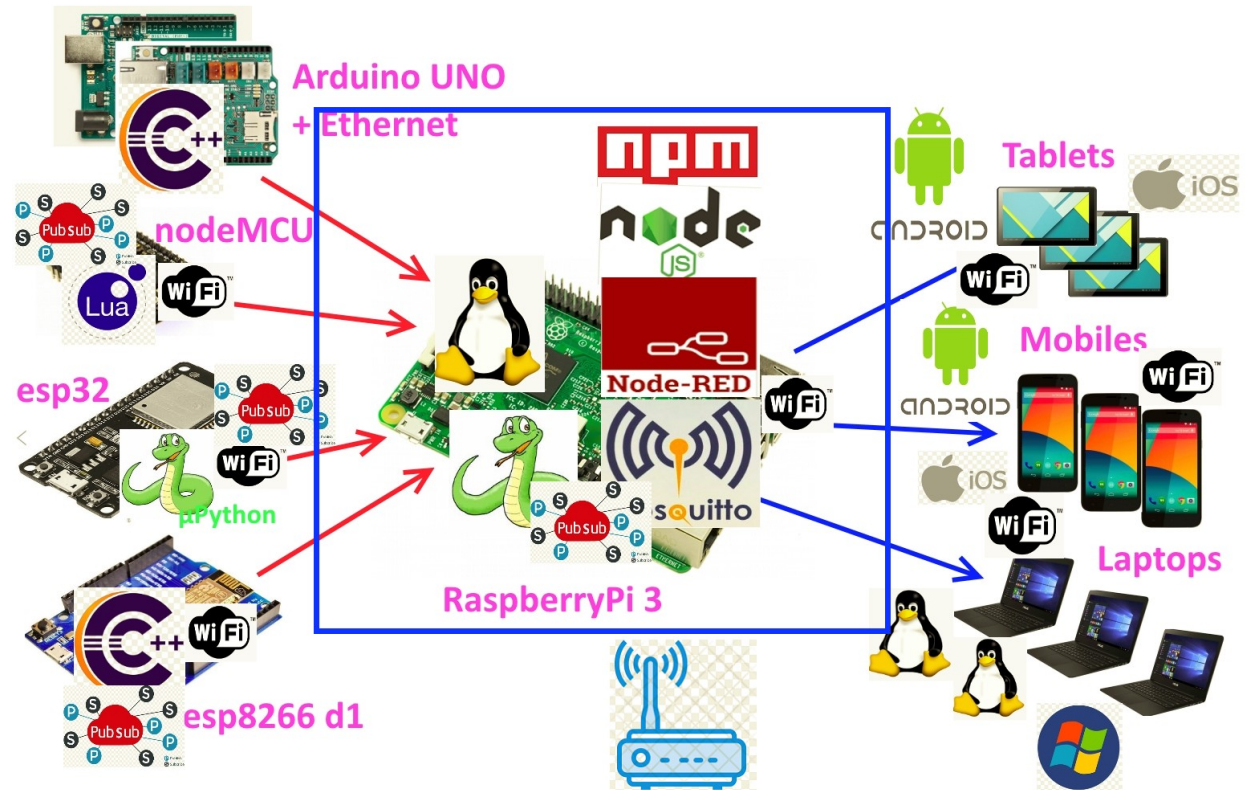
Όλες οι εφαρμογές του NODE-RED είναι γραμμένες σε JAVASCRIPT.

Έτσι για να τρέξουν σε εφαρμογή που δεν είναι φυλλομετρητής (browser), είναι απαραίτητη η εφαρμογή node.js που στην ουσία είναι ένας μεταφραστής JAVASCRIPT (Runtime).

Ακολούθως αξιοποιείται το NODE-RED για τη γραφιστική απεικόνιση των μεγεθών που πρέπει να διαχειριστούμε.

Αν τρέξουμε Python και αξιοποιήσουμε τις βιβλιοθήκες raho Pub/Sub μπορούμε να υλοποιήσουμε ένα ακόμη σύστημα εκδότη/πελάτη στο ίδιο το **Raspberry Pi 3**

Όλα τα ανωτέρω απεικονίζονται στο κεντρικό ορθογώνιο του σχήματος 7.



Σχήμα 1.7: Οι λειτουργίες στην πλακέτα RaspberryPi3

Ο πυρήνας

Όπως αναφέραμε στον πρόλογο, κύριο μέλημα της εργασίας μας δεν ήταν να προσθέσουμε έναν ακόμη αυτοματισμό κατοικίας, αλλά να δειχθεί το παιδαγωγικό μέρος της, ενώ παράλληλα θα μας ανάγκαζε να εμπλακούμε με όλα αυτά αποκομίζοντας τεράστιο όγκο γνώσης.

Στο σχήμα 7 απεικονίζεται το σύνολο των προγραμμάτων που τρέχουν στην εργασία μας ώστε οι μαθητές μας να έχουν πλήρη εικόνα της πληθώρας των συσκευών, των δικτύων, των πρωτοκόλλων, των λειτουργικών συστημάτων και των εφαρμογών που συνεργάζονται αρμονικά για να πετύχουμε το σκοπό μας.

Στον πυρήνα της εργασίας μας, RaspberryPi3 εγκαταστάθηκαν και λειτούργησαν τα παρακάτω:

-Το λειτουργικό σύστημα Linux (Jessie)

- Η γλώσσα προγραμματισμού Python και pyPub/pySub για να μπορούμε να οργανώσουμε το ίδιο το RaspberryPi3 πέραν των υπολοίπων λειτουργιών του και ως εκδότη/συνδρομητή στο MQTT.
- Ο MQTT Broker στην ελαφρά ελεύθερη υλοποίηση mosquitto.
- Ο loader npm για να εγκαταστήσουμε εφαρμογές μεταξύ των οποίων το Nodejs και το Node-Red
- Το Nodejs ως προϋπόθεση για να τρέξει το Node-Red και οι εφαρμογές σε javascript .
- Το Node-Red ως διαδικτυακή γραφιστική εφαρμογή-συνδέτης, μεταξύ της εποπτείας από τον χρήστη και των λειτουργιών (actions), στις συσκευές IoT.

Οι εξυπηρετητές

Οι εξυπηρετητές πρέπει να διαθέτουν τα εξής:

- Τρόπο επεξεργασίας της εφαρμογής
- Τρόπο συλλογής δεδομένων από το περιβάλλον
- Τρόπο επίδρασης προς το περιβάλλον
- Τρόπο διασύνδεσης με το διαδίκτυο

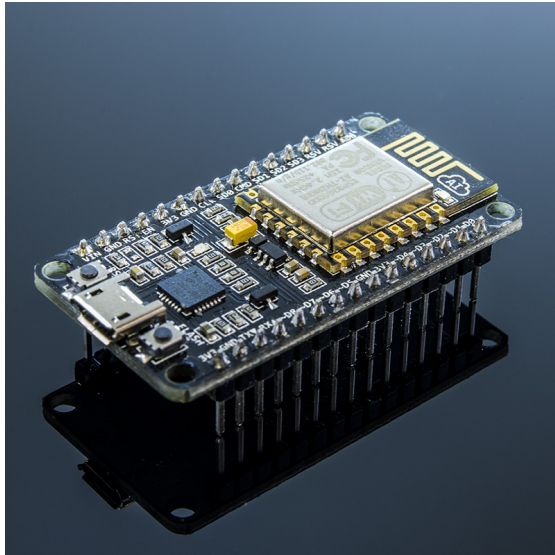
Ο server NodeMCU ESP8266

Ο server υλοποιείται στην πλακέτα NodeMCU ESP8266 και στους ακροδέκτες του συστήματος συνδέονται αισθητήρες ως είσοδοι φυσικών μεγεθών και ηλεκτρικά κυκλώματα διαχείρισης ισχύος για την ενεργοποίηση –απενεργοποίηση ηλεκτρικών συσκευών ως έξοδοι.

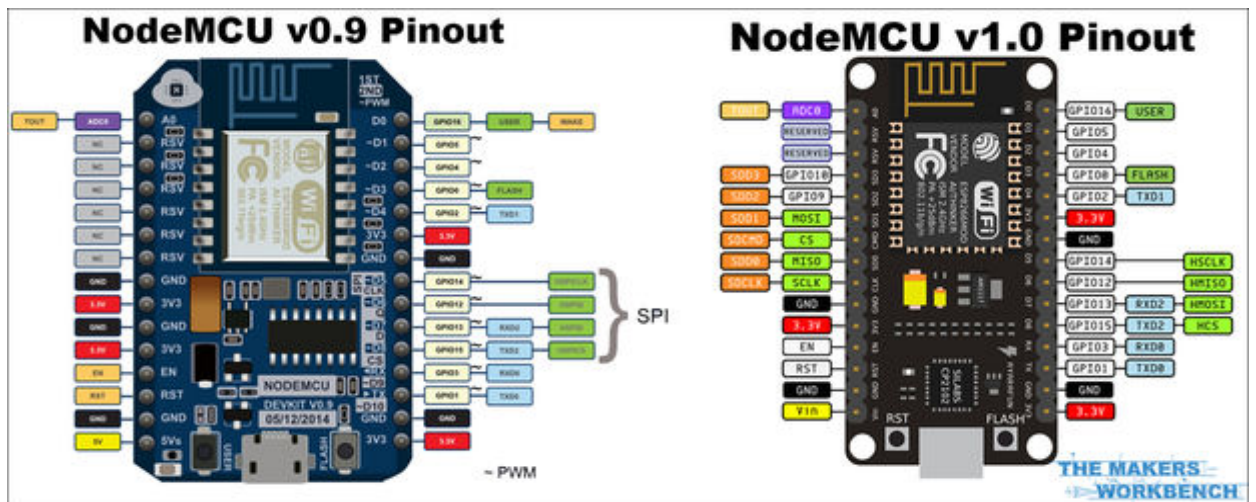
Ρόλος του server είναι να:

1. Τρέχει την εφαρμογή που είναι γραμμένη σε γλώσσα προγραμματισμού C/C++/mPython
- 6 Τρέχει το πλήρες stack (στοίβα) του πρωτοκόλλου TCP/IP
- 7 Τρέχει το πρωτόκολλο ασύρματης επικοινωνίας WiFi για διασύνδεση με τον Internet router
- 8 Ελέγχει τις εισόδους για τη λήψη των φυσικών μεγεθών
- 9 Ελέγχει τις εξόδους για την ενεργοποίηση απενεργοποίηση των διασυνδεδεμένων ηλεκτρικών συσκευών.

Η απεικόνιση του είναι στο σχ.1-8



Σχήμα 1.8: Πλακέτα NodeMCU ESP 8266.



Σχήμα 1.9: NodeMCU pinout

Ακολουθεί η απεικόνιση τριών πλακετών τύπου Arduino.

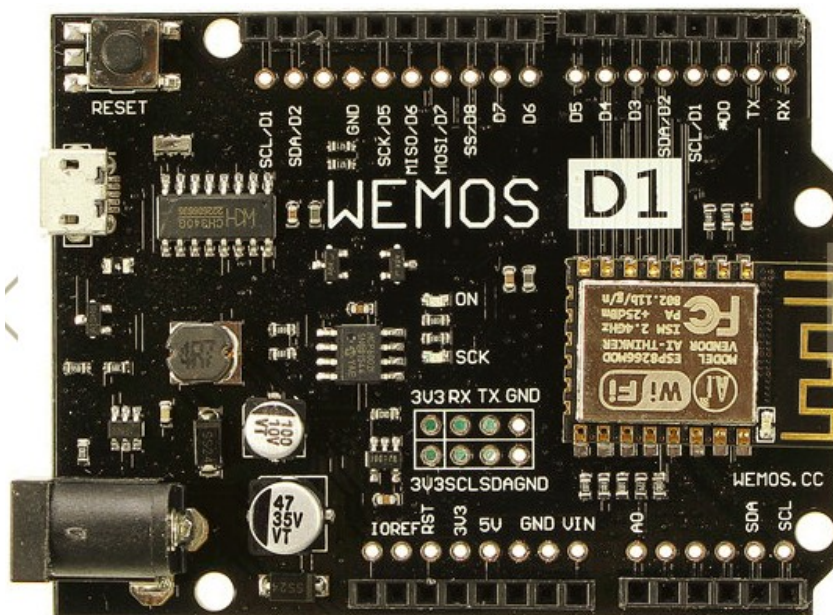
Η πρώτη classic Arduino Uno έχει ως κεντρικό μικροελεγκτή AVR των 8 bit, που τρέχει στα 16 MHz, με 32 KB Flash memory και ανάλογη υπολογιστική ισχύ.

Σήμερα αποτελεί το minimum για την είσοδο στα IoT.



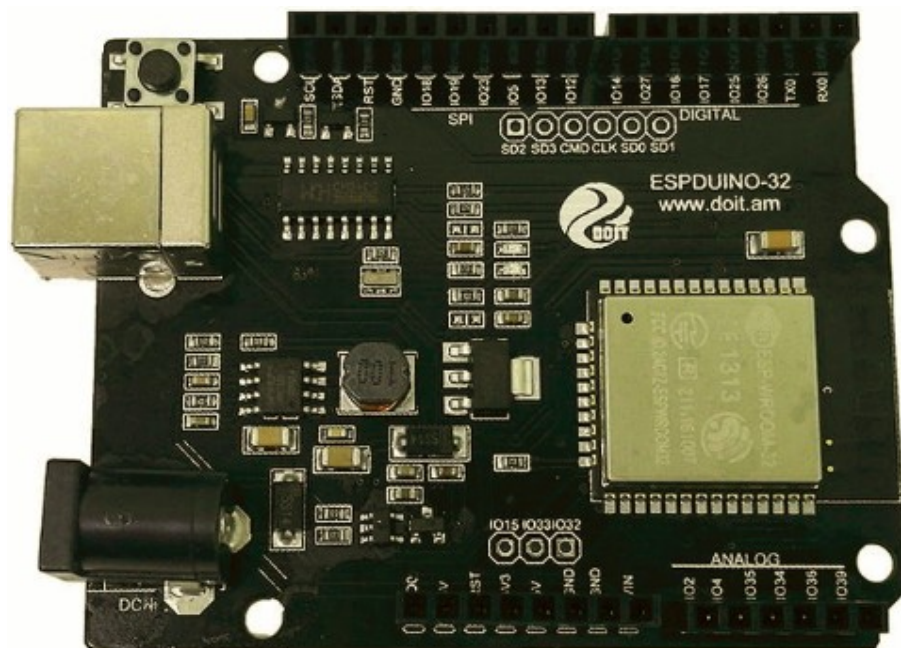
Σχήμα 1.10: Η κλασική πλακέτα Arduino με επεξεργαστή AVR των 8 bit

Η Arduino like πλακέτα D1 που ακολουθεί έχει ως κεντρική μονάδα επεξεργασίας τον ESP8266 των 32 bit. Η λειτουργία της είναι ταχύτερη, το Uploading επίσης γρήγορο, αλλά το μόνο μειονέκτημά της είναι οι περιορισμένες ψηφιακές εισόδους καθώς και η μοναδική αναλογική (A0)!



Σχήμα 1.11: Πλακέτα ESP8266 σε σχεδίαση τύπου Arduino

Στη συνέχεια απεικονίζεται η πλακέτα ESP32. Η πλακέτα αυτή πολλαπλασιάζει τις δυνατότητες του ESP8266 έχοντας ενσωματωμένους επεξεργαστές για WiFi και Bluetooth και 6 αναλογικές θύρες.



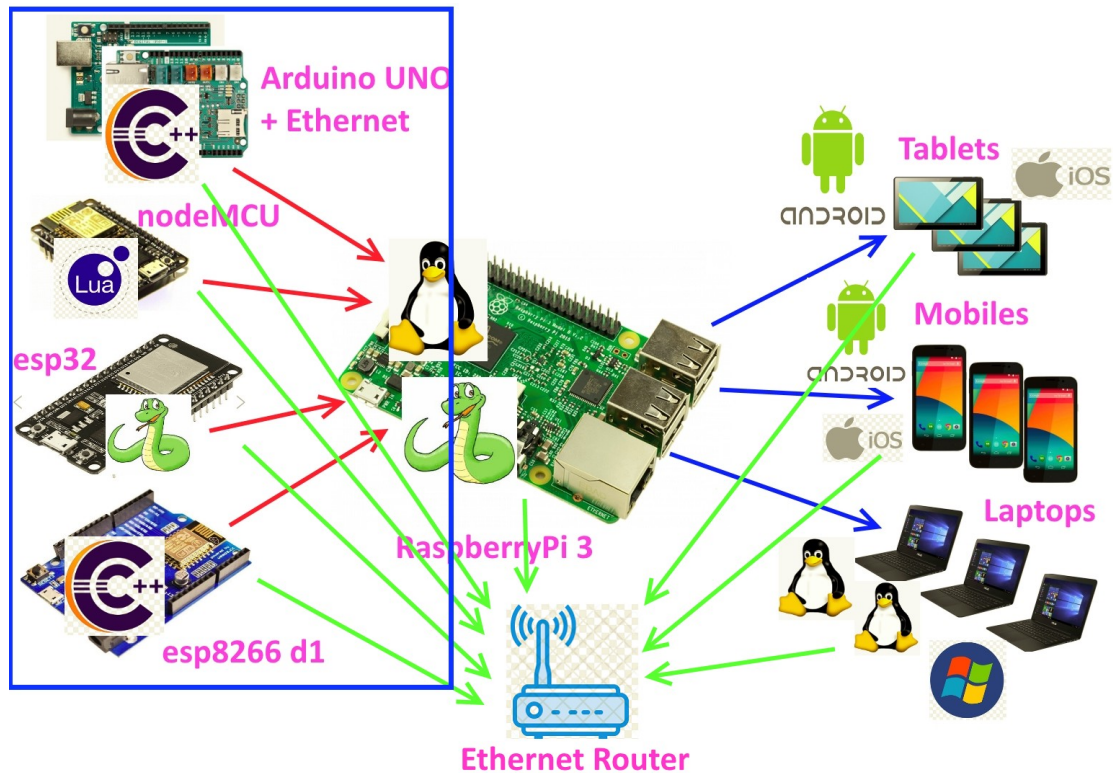
Σχήμα 1.12: Πλακέτα ESP32 στην κλασική μορφή Arduino

Arduino

Έτσι στην περίπτωση της πλακέτας Arduino, (που μας χορηγήθηκε από το Πανεπιστήμιο) προσθέσαμε πλακέτα-διεπαφή Ethernet. Αυτή απαιτεί ενσύρματη σύνδεση με καλώδιο Ethernet σε μία από τις εισόδους του μεταγωγέα (Switch) που είναι ενσωματωμένος στον διαδικτυακό δρομολογητή (Internet Router).

Ο εξυπηρετητής Arduino τρέχει εφαρμογή γραμμένη σε C/C++, ενσωματώνει τη βιβλιοθήκη Pub/Sub και διαθέτει τις κλασικές αναλογικές και ψηφιακές θύρες για τη διασύνδεση με το περιβάλλον μέσω αισθητήρων ή ηλεκτρονόμων (ρελλέ) ισχύος.

Στο σχήμα 1-13 (εντός ορθογωνίου), φαίνονται οι εξυπηρετητές που αξιοποιούμε.

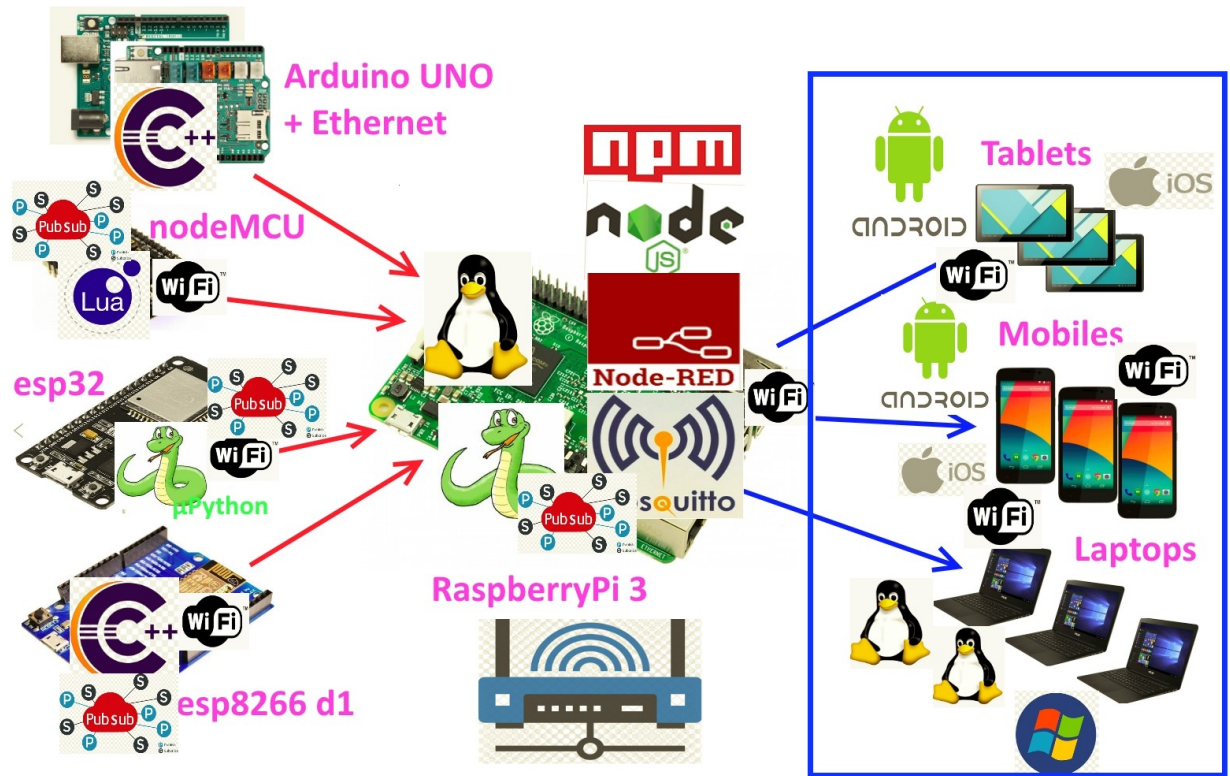


Σχήμα 1.13: Οι servers στέλνουν στον broker πακέτα πληροφορίας μέσω του δρομολογητή.

Οι πελάτες-συνδρομητές (Clients)

Στο δεξιό μέρος της εικόνας εμφανίζονται οι συσκευές εποπτείας και τηλεχειρισμού.

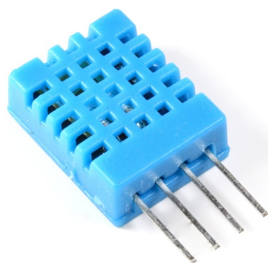
Οι Clients είναι συνήθως φορητές συσκευές που έχουν δυνατότητα απεικόνισης σε οθόνη LCD. Κάθε μια τους τρέχει το λειτουργικό για το οποίο είναι σχεδιασμένη με δυνατότητα να τρέχει οποιονδήποτε browser αν και για λόγους πλησιέστερης συμβατότητας με το Android και τα Windows, προτιμάται το Chrome. Ως εσωτερική υποδομή υπονοείται η στοίβα του πρωτοκόλλου TCP/IP και η διαδικτυακή Data link λειτουργία με WiFi για το ασύρματο ή το Ethernet για το ενσύρματο δίκτυο.



Σχήμα 1.14: Οι συνδρομητές λαμβάνουν πακέτα πληροφορίας

Οι αισθητήρες

Στη παρούσα εργασία επιλέχθηκαν απλοί αισθητήρες χαμηλού κόστους ευρείας εφαρμογής ώστε να είναι προσιτοί στους μαθητές. Για τη μέτρηση θερμοκρασίας και υγρασίας επιλέχθηκε ο DHT 11 σχ. 15 με τα εξής χαρακτηριστικά:



Σχήμα 1.15: Αισθητήρας θερμοκρασίας-υγρασίας DHT 11

- 1.DHT 11 series numeric humiture sensor.
- 2.Humidity measuring range: 20%~90%RH(0—50°Ctemperature compensation).
- 3.Temperature measuring range: 0~+50°C.
- 4.Humidity measurement accuracy: ±5.0%RH.

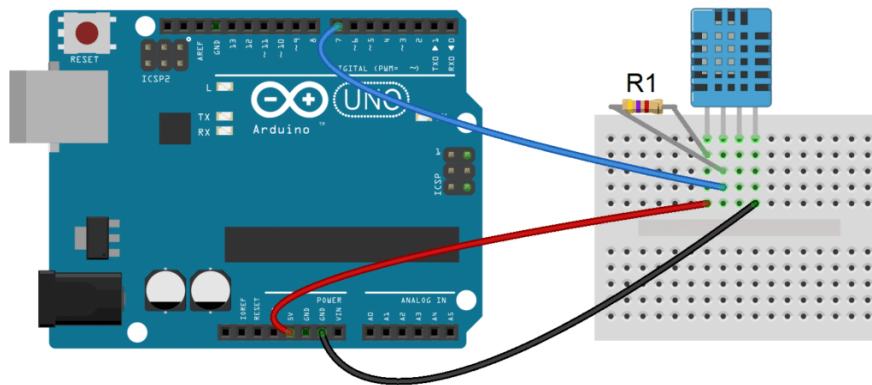
5. Temperature measurement accuracy: $\pm 2.0^{\circ}\text{C}$.

6. response time: $< 5\text{s}$.

7. low power consumption.

8. Size: $2.3\text{cm} \times 1.2\text{cm} \times 0.5\text{cm}$.

Τρόπος σύνδεσης και κώδικας



Σχήμα 1.16: Η σύνδεση του αισθητήρα DHT 11

Ο κώδικας λειτουργίας σε C/C++

```
//+++++=====SENSORS=====+++++  
//
```

```
float hum = dht.readHumidity(); // διαβάσε θερμοκρασία  
//  
float temp = dht.readTemperature(); // διάβασε υγρασία  
//  
// κάνε έλεγχο, αν όχι τύπωσε μήνυμα  
if (isnan(hum) || isnan(temp)) {  
    Serial.println("Failed to read from DHT sensor!");  
}
```

Αισθητήρας ανίχνευσης επικίνδυνων αερίων

Με τον αισθητήρα MQ-2 Gas sensor ανιχνεύεται ομάδα επικίνδυνων αερίων που συνήθως συνυπάρχουν εντός της κατοικίας και η διαρροή τους στον εσωτερικό ζωτικό χώρο μπορεί να προκαλέσει βλάβες στα άτομα που τα εισπνέουν, καθώς επίσης και εκρήξεις. Κάποια από αυτά τα αέρια είναι άοσμα και προκαλούν δηλητηρίαση ή ασφυξία ιδιαίτερα σε μικρά παιδιά. Είναι ο λόγος για τον οποίο θεωρείται απαραίτητη η παρουσία ένας τέτοιου αισθητήρα μέσα στη κατοικία που κλείνει ερμητικά για να αποφεύγονται θερμικές απώλειες.



Σχήμα 1.17: Αισθητήρας ανίχνευσης αέριου νέφους

MQ-2 Gas sensor

Ο αισθητήρας λειτουργεί όταν βρεθεί σε περιβάλλον νέφους αποτελούμενο από έστω και μία από τις κατωτέρω αέριες χημικές ενώσεις.

- 1.LPG
- 2.Butane
- 3.Propane
- 4.Methane
- 5.Alcohol
- 6.Hydrogen

Το επίπεδο ενεργοποίησης συναγερμού είναι ρυθμίσιμο μέσω ποτενσιομέτρου.

Η τάση τροφοδοσίας είναι 5 V

Ο κώδικας λειτουργίας σε C/C++

```

//++++++=====+++++ GAS SENSOR ++++++
//
float GasLeak = analogRead(GasLeakA0);
Serial.println(GasLeak);
static char gasLeakage[7]; //δημιουργία σειράς χαρακτήρων
dtostrf(GasLeak, 6, 2, gasLeakage);
//----- Έλεγχος κατωφλίου
if (Gas > sensorThres)
{
Serial.print("GasLeakage Exceeded Threshold: ");
Serial.println(Gas);
}
else{
Serial.print("GasLeakage Lower Than Threshold: ");
Serial.println(Gas);
}

```

Αισθητήρας ανίχνευσης κίνησης

Η διάταξη HC-SR501 PIR αξιοποιεί αισθητήρα υπέρυθρης ακτινοβολίας με δυνατότητα να ανιχνεύει θερμά σώματα. Η «ευαισθησία» του και ο χρόνος ενεργοποίησης είναι ρυθμιζόμενα. Στη κατοικία ανιχνεύεται η παρουσία παρείσακτου εισβολέα.



Σχήμα 1.18: ο αισθητήρας PIR

Τα χαρακτηριστικά του αισθητήρα

HC-SR501 PIR Motion Sensor

1. Working voltage: DC5V to 20V
2. Static consumption: 65 microamps
3. level output: 3.3V high, low 0V
4. Time delay: Adjustable (0.3 seconds to 18 seconds)
5. Blockade of the time: 0.2 seconds
6. Trigger: L can not be duplicated, H can be repeated, the default value of H
7. Sensing range: less than 120 degrees cone angle less than 7 m
8. Working temperature: -15 to +70 degrees
9. PCB Dimensions: 32 * 24mm ,screw pitch 28mm ,screw aperture 2mm, sensor lens Dimensions: (diameter): 23mm (default)
10. Material: Mixture

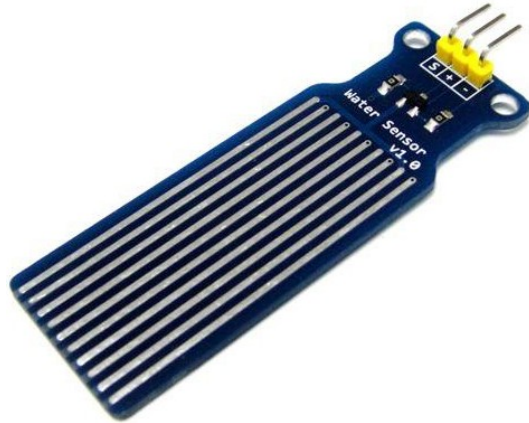
Ο κώδικας λειτουργίας του αισθητήρα PIR σε C/C++

```
//+++++ PIR ++++++  
//  
salert_value=digitalRead(salert); // διάβασε την πόρτα  
if(salert_value==HIGH) // αν στέλνει HIGH στο Arduino  
{  
  static char alert[7]="HIGH";  
  //  
  digitalWrite(salarm,LOW); // ενεργοποίηση ηχητικό σήμα  
  Serial.println("Intruder In the House!");  
  delay(50);  
  digitalWrite(salarm,HIGH); // σταμάτησε τον ήχο  
  salert_value=LOW;  
}
```

Αισθητήρας αντίχρεσης πλημμύρας

Αισθητήρας ανίχνευσης πλημμύρας στο εσωτερικό των πατωμάτων ή στην επιφάνεια των πατωμάτων.

Συνήθως τοποθετείται επαπτόμενος του πατώματος



Σχήμα 1.19: Αισθητήρας ανίχνευσης πλημμύρας

Όταν οι αγώγιμες μπάρες καλυφθούν με υγρά (συνήθως νερό) τότε η αγωγιμότητα μεταξύ τους αυξάνεται και η μείωση της αντίστασης ελαττώνει την τάση εισόδου.

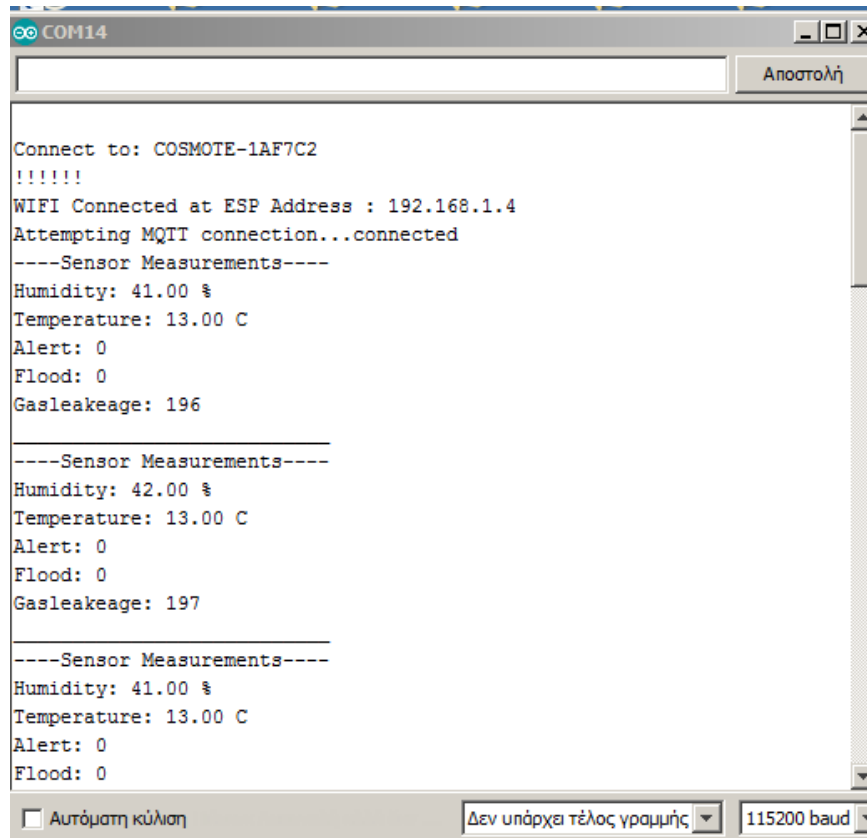
Αυτό ανιχνεύεται στο pin εισόδου του εξυπηρετητή που με τη σειρά του ενεργοποιεί σήμα συναγερμού.

Ο κώδικας λειτουργίας σε C/C++

```
//+++++ FLOODING SENSOR ++++++  
//  
kflood_value=digitalRead(kflood); // διάβασε την πόρτα  
if(kflood_value==HIGH) // αν είναι HIGH στο Arduino  
{  
  static char flood[7]="HIGH";  
  //  
  digitalWrite(salarm,LOW);  
  Serial.println("Floor Flooding In the House!");  
  delay(50);  
  digitalWrite(salarm,HIGH);  
  kflood_value=LOW;
```

}

Η λειτουργία του εξυπηρετητή στην οθόνη σειριακής εποπτείας του Arduino IDE



The screenshot shows the serial monitor window in the Arduino IDE. The window title is 'COM14'. The text displayed in the monitor is as follows:

```
Connect to: COSMOTE-1AF7C2
!!!!!!
WIFI Connected at ESP Address : 192.168.1.4
Attempting MQTT connection...connected
----Sensor Measurements----
Humidity: 41.00 %
Temperature: 13.00 C
Alert: 0
Flood: 0
Gasleakeage: 196

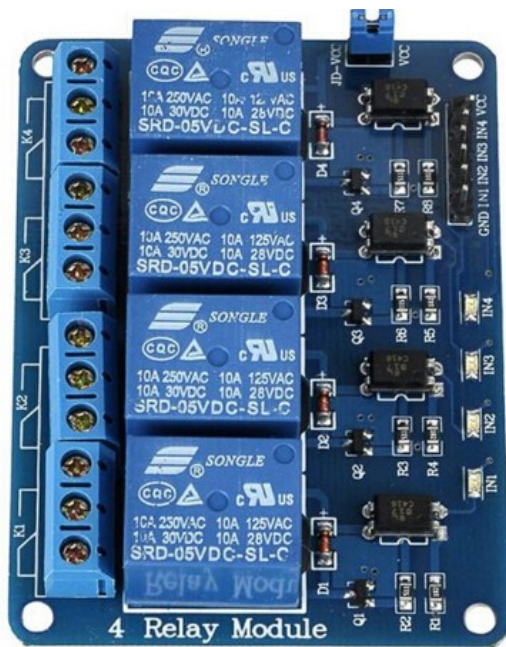
----Sensor Measurements----
Humidity: 42.00 %
Temperature: 13.00 C
Alert: 0
Flood: 0
Gasleakeage: 197

----Sensor Measurements----
Humidity: 41.00 %
Temperature: 13.00 C
Alert: 0
Flood: 0
```

At the bottom of the window, there is a checkbox for 'Αυτόματη κύλιση' (Auto scroll) which is unchecked. To its right is a dropdown menu with the text 'Δεν υπάρχει τέλος γραμμής' (No line ending) and a baud rate dropdown menu set to '115200 baud'.

Σχήμα 1.20:Παράθυρο εποπτείας σειριακής πόρτας IDE

Συσκευές εξόδου

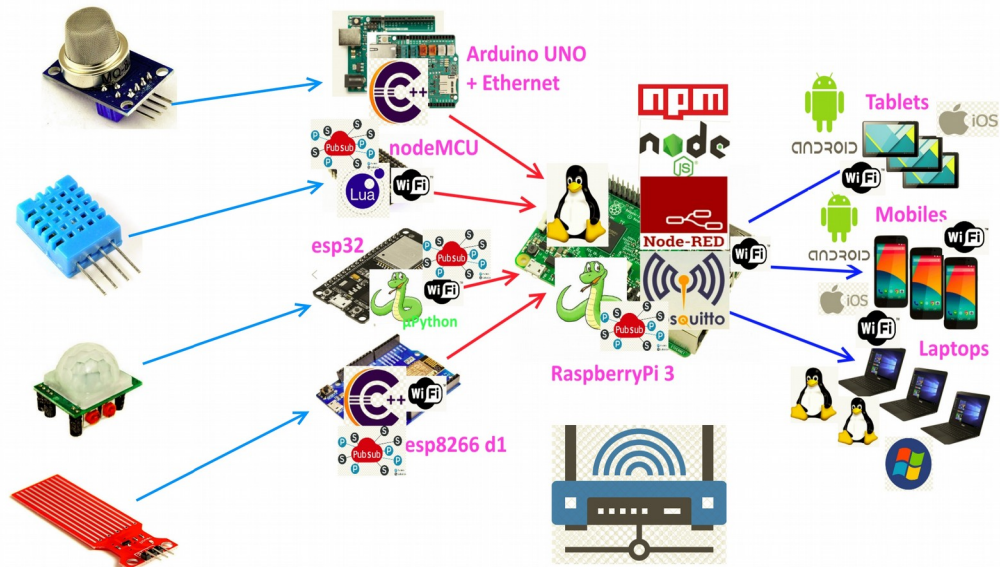


Σχήμα 1.21: Πλακέτα 4 ρελλέ με οπτική απομόνωση

Ο Internet Router



Internet Router με διεπαφή Ethernet Switch 4 εισόδων που τρέχει DHCP για εκχώρηση διευθύνσεων και NAT για τη μετάφραση εσωτερικών προς εξωτερικές διευθύνσεις. Στη συνέχεια δείχνουμε και τις συσκευές εισόδου (αισθητήρες).



Σχήμα 1.22.: Η εικόνα περιλαμβάνει τις συσκευές εξόδου καθώς και τις εποπτεύουσες συσκευές

ΚΕΦΑΛΑΙΟ 2 - Συνδρομητική Εγγραφή (Subscription)

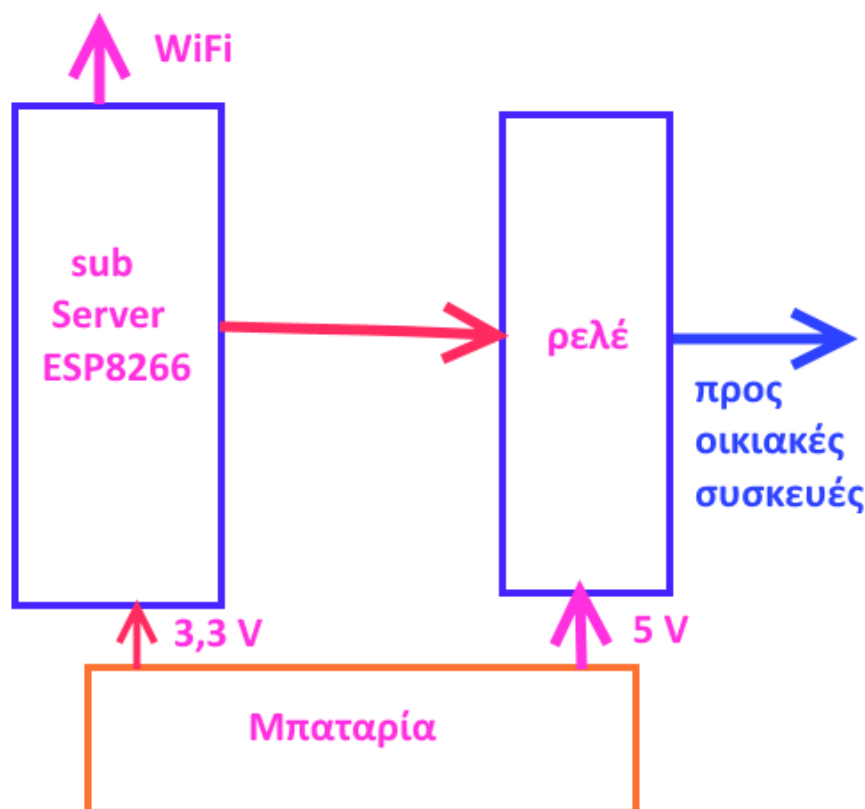
Στο κεφάλαιο 1 αναφέραμε το γενικό σκοπό και τα μέρη (υλικό και λογισμικό) της εργασίας μας. Στο τρέχον κεφάλαιο θα δείξουμε την πρώτη μας πλήρη εφαρμογή που ο εξυπηρετητής εγγράφεται στον broker σε 8 θέματα για το χειρισμό 8 οικιακών συσκευών μέσω του διαδικτύου. Θα αξιοποιήσουμε την πλακέτα NodeMCU8266 και θα συνδέσουμε σε αυτήν συστοιχία 8 ρελλέ ως διεπαφές εξόδου προκειμένου να χειριστούμε οικιακές συσκευές που απαιτούν υψηλές τάσεις και μεγάλα ρεύματα.

Επειδή ο μικροελεγκτής είναι εξαιρετικά ευαίσθητος υπάρχει οπτικός απομονωτής σε κάθε έξοδό του, μεταξύ θύρας εξόδου και ρελλέ, πάνω στην πλακέτα των ρελλέ.

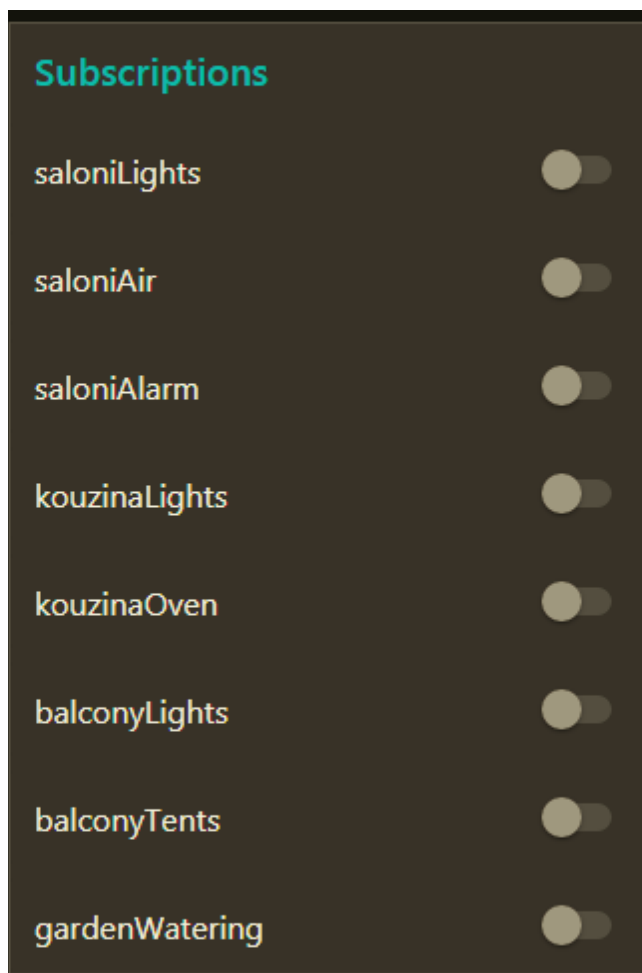
Εξυπακούεται, ότι για να πετύχουμε το στόχο μας που είναι ο χειρισμός συμβατικών συσκευών μέσω του διαδικτύου, πρέπει να παρεμβάλουμε την επαφοδέσμη των ρελλέ στην ηλεκτρική τροφοδοσία ώστε να λειτουργούν ως διακόπτες ON-OFF.

Ο χειριστής των συσκευών βρίσκεται σε απόσταση και χειρίζεται εικονικούς διακόπτες στην οθόνη ταμπλέτας, έξυπνου κινητού ή υπολογιστή. Μοναδική προϋπόθεση είναι η φορητή συσκευή να συνδέεται στο διαδίκτυο και να τρέχει φυλλομετρητή(browser).

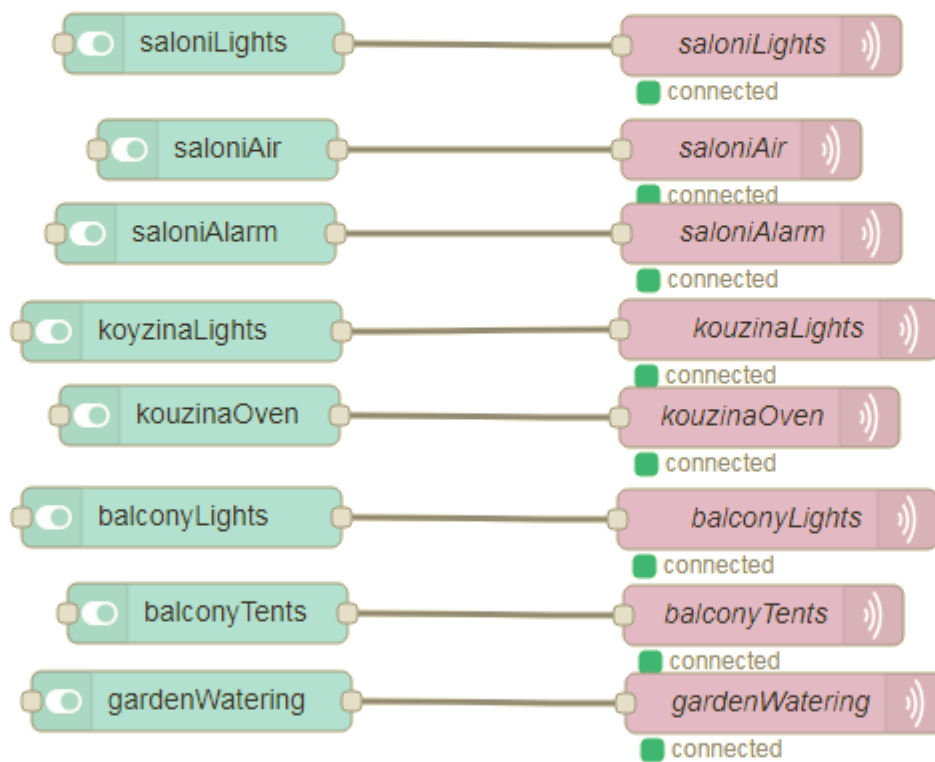
Ακολουθεί η απεικόνιση των ροών των κόμβων στο συντάκτη του NODE-RED.



Σχήμα 2.1: Μπλόκ διάγραμμα εξυπηρετητή εγγραφής



Σχήμα 2.2: Η απεικόνιση του πίνακα ελέγχου (Dashboard) του NODE-RED



Σχήμα 2.3: Η απεικόνιση των ροών στον συντάκτη NODE-RED

Ο κώδικας που τρέχει στον εξυπηρετητή αναπτύχθηκε σε γλώσσα C/C++.
Στο παράρτημα κεφάλαιο 2

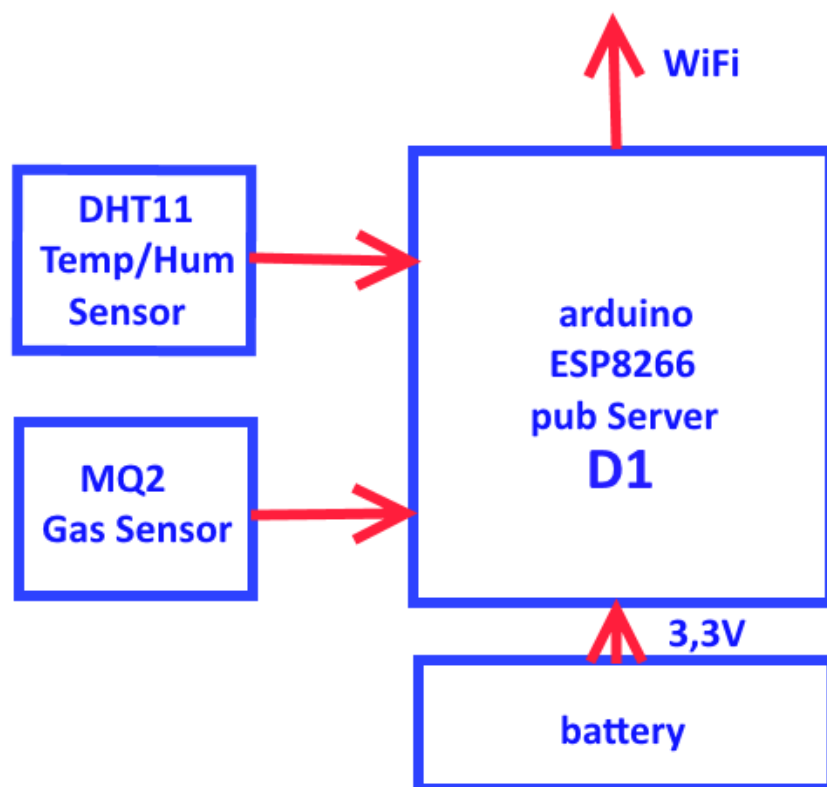
ΚΕΦΑΛΑΙΟ 3 - Έκδοση Δεδομένων (Publishing)

Στο κεφάλαιο 3 θα εστιάσουμε στη λειτουργία έκδοσης των μηνυμάτων. Δηλαδή ο εξυπηρετητής θα εκδίδει (Publish) τις τιμές των φυσικών μεγεθών, όπως θερμοκρασία, σχετική υγρασία κλπ που μετράει μέσω των αισθητήρων.

Ως εξυπηρετητής αξιοποιήθηκε η πλακέτα D1 που έχει δομή Arduino.

Λογισμικό υποστήριξης είναι το Arduino IDE και C/C++

Η δομή σε block διάγραμμα φαίνεται παρακάτω



Σχήμα 3.1: έκδοση δεδομένων με Arduino like D1 board

Όπως φαίνεται τόσο στην εικόνα όσο και στο σχηματικό διάγραμμα αξιοποιήθηκαν οι αισθητήρες DHT11 για μετρήσεις θερμοκρασίας και σχετικής υγρασίας καθώς και ο αισθητήρας εντοπισμού διαρροής επικίνδυνων αερίων.

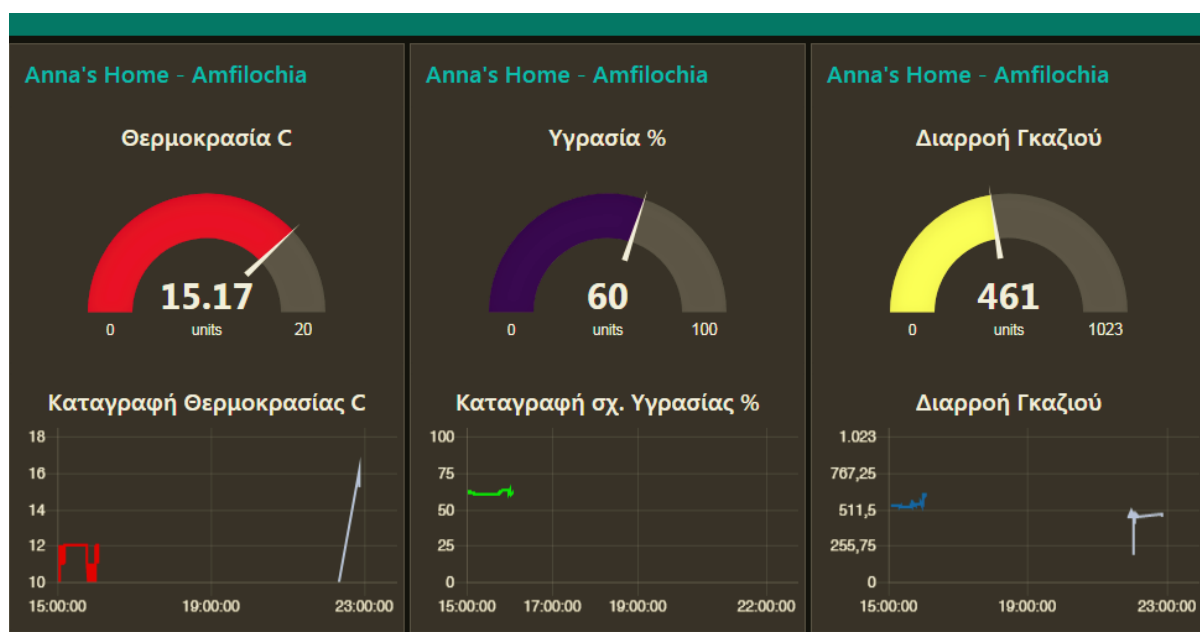
Αξιοποιήσαμε επίσης αισθητήρα PIR για τον εντοπισμό παραβίασης χώρου και προσπαθούμε να το κάνουμε να στέλνει μηνύματα e-mail. (δεν έχει υλοποιηθεί)

Οι τιμές των φυσικών μεγεθών (είναι συνήθως είναι τύπου float), μετατρέπονται σε σειρά χαρακτήρων (έτσι απαιτεί το πρωτόκολλο), και μεταφέρονται στο αντίστοιχο θέμα (Topic) που έχει δηλωθεί και «κατοικεί» στον broker.

Ο αντίστοιχος κώδικας είναι:

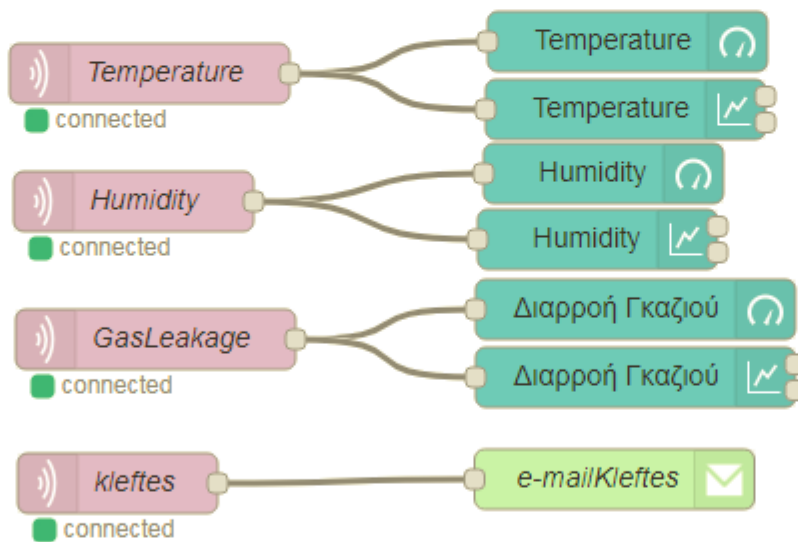
```
client.publish("home/temp", temperatureTemp);  
client.publish("home/humid", humidityTemp);  
client.publish("home/gas", gasLeakage);  
client.publish("home/light", light);  
client.publish("home/pir", pir);
```

Στη συνέχεια υπάρχει ο πίνακας ελέγχου, (dashboard) του NODE_RED, όπου εκτός της απεικόνισης των τρεχουσών τιμών συνυπάρχουν και καταγραφείς. Οι καταγραφείς αυτοί εκτείνονται στο χρόνο, ώστε να έχουμε το πλήρες ιστορικό των μεταβολών των τιμών που έχουν τα παρατηρούμενα φυσικά μεγέθη.



Σχήμα 3.2: Απεικόνιση του εικονικού πίνακα ελέγχου

Είναι προφανές ότι ο ανωτέρω πίνακας εποπτείας & ελέγχου σχεδιάστηκε και υλοποιήθηκε από ένα σύνολο ροών του NODE-RED που απεικονίζονται στο επόμενο σχήμα..



Σχήμα 3.3: Απεικόνιση των ροών στον editor NODE-RED

Οδηγίες για να τρέξει η εφαρμογή του κεφαλαίου

{ Πρώτα τρέχουμε σε(command mode) τον MQTT server (mosquitto -v),στον υπολογιστή μας
 Στη συνέχεια σε νέο παράθυρο CMD mode ενεργοποιούμε το (node-red) και τον browser
 (localhost:1880)

Με τον IP scanner εντοπίζουμε τις IP διευθύνσεις του MQTT broker
 και του ESPserver, καθώς και των υπολοίπων Android συσκευών

Τοποθετούμε την IP διεύθυνση στον κώδικά του εξυπηρετητή κάνουμε διεργασία
 και στη συνέχεια τον ανεβάζουμε στη Flash του εξυπηρετητή
 Φορτώνουμε τις ροές και

Ενεργοποιούμε το dashboard.}

{ο κώδικας του κεφαλαίου στο παράρτημα κεφ 3}

ΚΕΦΑΛΑΙΟ 4 - Το κλασικό Arduino ως εξυπηρετητής στο MQTT

Αναφέραμε πως ρόλος μας στην παρούσα εργασία είναι η παιδαγωγική παρουσίαση των μερών που την αποτελούν.

Για αυτό το λόγο θα παρουσιάσουμε εκτενέστερα, τόσο τις σκέψεις μας όσο και τη διαδοχή των βημάτων ώστε κατακτώντας κάθε ενδιάμεσο στόχο, να φτάσουμε στο σκοπό μας.

Έτσι ως μέλος των διατάξεων εισόδου στην εργασία μας θα αξιοποιήσουμε την παλιά κλασική πλακέτα Arduino UNO, παραβλέποντας και ξεπερνώντας τις αδυναμίες της που είναι:

Ανυπαρξία WiFi

Μικρή αποθηκευτική μνήμη (32K- bootstrap)

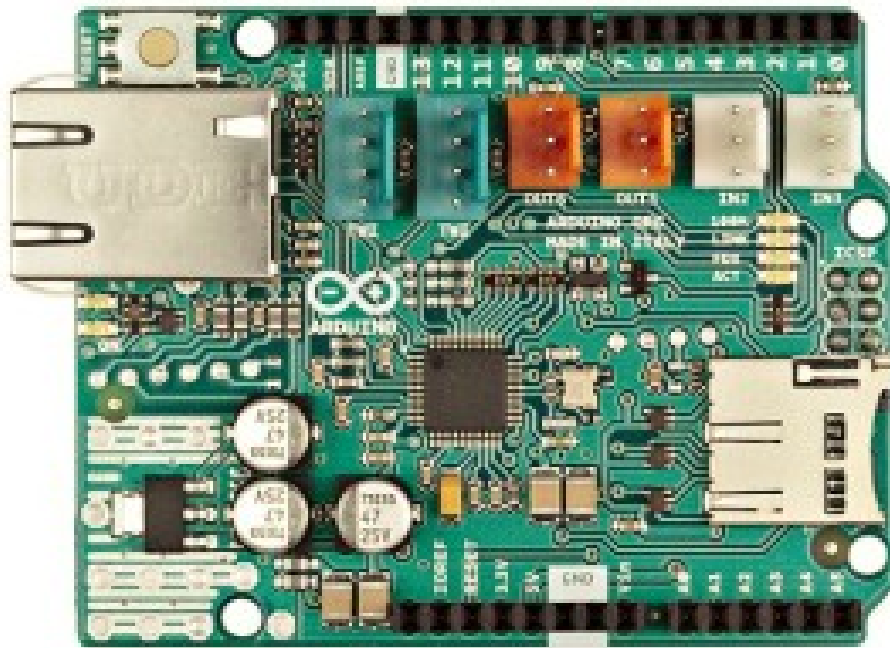
Μικρή ισχύς επεξεργασίας (AVR 32- 8 bit RISC)

Έτσι θα αξιοποιήσουμε την πλακέτα διεπαφής Ethernet που θα συνδεθεί με τον δρομολογητή του διαδικτύου με καλώδιο επίσης Ethernet.

Αυτό μας δεσμεύει εν μέρει, αφού μας στερεί την ανεξαρτησία του ασύρματου WiFi και πρέπει να τοποθετηθεί αρκετά κοντά στο δρομολογητή(router).

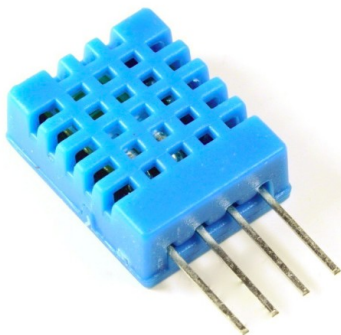


Σχήμα 4.1: Η κλασική πλακέτα Arduino UNO.



Σχήμα 4.2: Η πλακέτα Ethernet

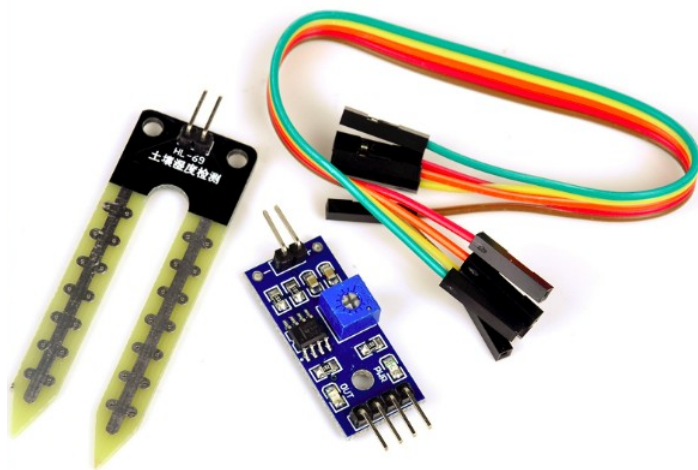
Ως συσκευές εισόδου επιλέξαμε απλούς αισθητήρες (μέρος των οποίων μας εκχώρησε το Πανεπιστήμιο) και για έξοδο όπου απαιτείται διαχείριση υψηλών τάσεων, επιλέξαμε ομάδα ηλεκτρονόμων (ρελλέ).



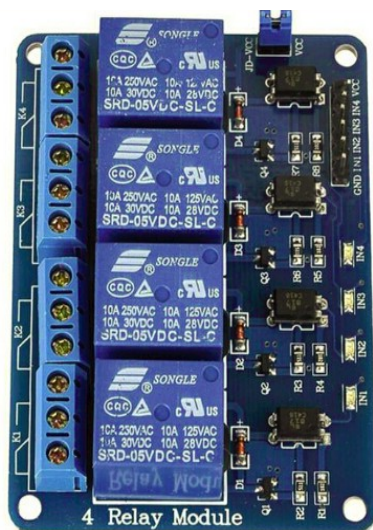
Σχήμα 4.3: αισθητήρας θερμοκρασίας και υγρασίας χαμηλού κόστους



Σχήμα 4.4: Φωτοαντίσταση για τη μέτρηση του φωτός



Σχήμα 4.5: Αισθητήρας μέτρησης της υγρασίας στο έδαφος



Σχήμα 4.6: Διάταξη ρελέ για τη διαχείριση φορτίων στην έξοδο

Η σκέψη μας

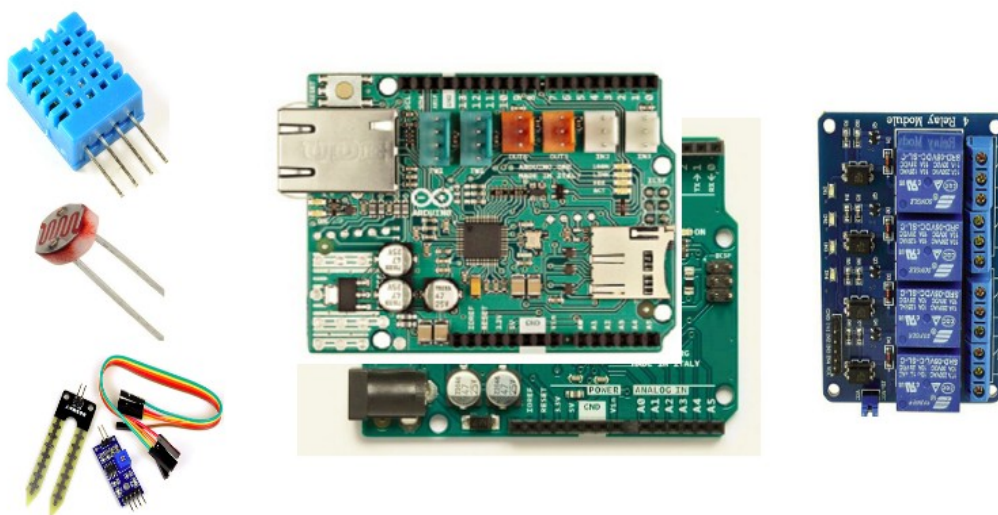
Θελήσαμε να οργανώσουμε την εποπτεία και τον έλεγχο των εξωτερικών χώρων μιας κατοικίας. Έτσι μετράμε την εξωτερική θερμοκρασία και τη σχετική υγρασία και τα εκδίδουμε στον broker που θα τα «εκπέμψει» σε όποιον συνδρομητή ενδιαφέρεται για αυτά. (προφανώς ο ένοικος).

Μετρούμε τη φωτεινή ροή που προσπίπτει στην επιφάνεια φωτοαντίστασης και της μεταβάλλει την τιμή. Έτσι ο εξυπηρετητής γνωρίζει πότε είναι ημέρα για να σβήσει τα φώτα του μπαλκονιού και να κατεβάσει την τέντα, και το αντίστροφο. Πότε δηλαδή είναι σκοτάδι για να ανάψει τα εξωτερικά φώτα και να ανεβάσει την αντηλιακή τέντα.

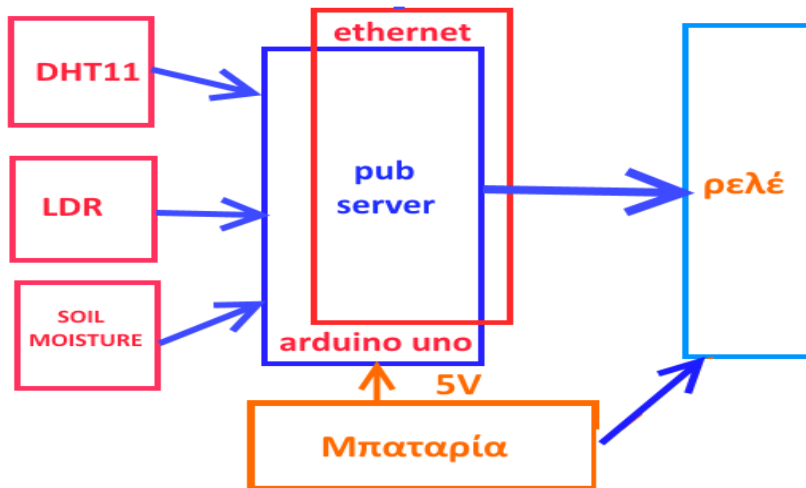
Βέβαια, για αυτές του τις ενέργειες πρέπει να ενημερώνει και τον ένοικο (και οποιονδήποτε άλλον έχει πρόσβαση). {Ενδεχομένως, [σε νεώτερη, βελτιωμένη έκδοση], να ζητάει την έγκρισή του πριν προβεί στην πράξη του}.

Ο σχεδιασμός μας

Δείχνουμε τη δομή του υποέργου μας σε εικόνες που τις συμπληρώνει το χειροποίητο σχέδιό μας.



Σχήμα 4.7: Τα μέρη που απαρτίζουν την άσκηση



Σχήμα 4.8 Το σχέδιο σε μπλοκ

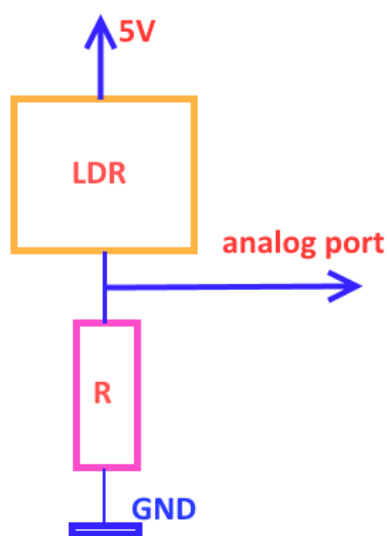
Το πρόγραμμα

Θέλουμε να δείξουμε στους μαθητές μας (μελλοντικούς σχεδιαστές συστημάτων IoT), ότι δεν αρχίζουμε μονομιάς, αλλά υλοποιούμε την εφαρμογή μας σταδιακά, αρχίζοντας από τα απλούστερα μέρη προς τα συνθετότερα.

Επιλέγουμε τη φωτοαντίσταση.

Με πολύμετρο στην κλίμακα αντιστάσεων μετρούμε την ωμική τιμή της στο σκοτάδι και τη βρίσκουμε $\sim 100 \text{ K}\Omega$ και στο φως όπου η τιμή της είναι $< 1 \text{ K}\Omega$.

Συνδέουμε σε σειρά με τον ακροδέκτη της φωτοαντίστασης μια αντίσταση $4,7 \text{ K}\Omega$, $\frac{1}{4} \text{ W}$ όπως στο σχήμα και το ανάμεσά τους σημείο σύνδεσης στην αναλογική πόρτα A2.



Σχήμα 4.9 Η σύνδεση της φωτοαντίστασης

Ο επι μέρους κώδικας

Αισθητήρας φωτός

Πρώτα διαβάζει την τιμή της αναλογικής πόρτας.

Αν η φωτεινή ροή είναι υψηλή (ημέρα) τότε σβήνει τα εξωτερικά φώτα και κατεβάζει την τέντα.

Βέβαια ο χρόνος δεν είναι επαρκής για το ανεβακατέβασμα της τέντας.

Παρατήρηση: Αν εφαρμοστεί κανονικά, τότε πρέπει ο χρόνος να μετρηθεί με ακρίβεια.

```
int dayLight=analogRead(dayLightPin);
//
Serial.print("dayLight : ");
Serial.println(dayLight);
if(dayLight>700){
    digitalWrite(balconyLightPin,HIGH);
    digitalWrite(tentUpPin,LOW);
    delay(1000); //Ο χρόνος να ρυθμιστεί ανάλογα
    digitalWrite(tentUpPin,HIGH);
    //
    Serial.println("SunLight->Tent Down");
    //digitalWrite(lightLEDPin,LOW);
}
else {
    digitalWrite(balconyLightPin,LOW);
    digitalWrite(tentDownPin,LOW);
    delay(1000); //Ο χρόνος να ρυθμιστεί ανάλογα
    digitalWrite(tentDownPin,HIGH);
    Serial.println("Dark->Tent Up");
    digitalWrite(lightLEDPin,HIGH);
}
```

Όταν ο κώδικας λειτουργήσει και τα ρελλέ ανοιγοκλείνουν σωστά, προχωρούμε στον επόμενο αισθητήρα.

Αισθητήρας υγρασίας εδάφους

Μετράται στην ουσία η αγωγιμότητα του εδάφους, ανάλογη της υγρασίας που περιέχει.

Η μέγιστη τιμή είναι 1023 (ξηρασία) και περίπου 0 (πλήρες υγρό με άλατα).

Στον κώδικα ορίζουμε ενδιαμέσες τιμές που είναι ρυθμίσιμες.

Αν η υγρασία είναι χαμηλή τότε ποτίζουμε ανοίγοντας τη βρύση, αν όχι, τότε η ηλεκτροβάννα της βρύσης κλείνει.

Ο κώδικας

```
moistureValue = analogRead(moisturePin);
if(moistureValue >= 700){
  digitalWrite(wateringPin,LOW);
  digitalWrite(moistsoilLEDPin,HIGH);
  Serial.print("Low moisture (DRY!):");
  Serial.println(moistureValue);
}
if((moistureValue < 800) && (moistureValue >= 600)){
  digitalWrite(wateringPin,LOW);
  digitalWrite(moistsoilLEDPin,HIGH);
  Serial.print("Medium moisture (MEDIUM !):");
  Serial.println(moistureValue);
}
if(moistureSensorValue < 600){
  digitalWrite(wateringPin,HIGH);
  digitalWrite(moistsoilLEDPin,LOW);
  Serial.print("Normal moisture (NORMAL!):");
  Serial.println(moistureSensorValue);
}
```

Το ίδιο θα επαναληφθεί και με τον αισθητήρα DH11, χωρίς όμως να σημειωθεί εδώ δεδομένου ότι θα εμφανίζεται στον πλήρη κώδικα.

```

//ANNA MAMOYZELOY
//PANEPISTHMIO AIGAIΟΥ
// ARDUINO PRE DESIGN
//

int dayLightPin=1;    //A1
int moisturePin= 2;   //A2
int moistsoilLEDPin = 4; //D4
int lightLEDPin     = 5; //D5
//

int tentUpPin       = 7;
int tentDownPin     = 8;
int balconyLightPin = 9;
int wateringPin     = 10;

int moistureSensorValue; // για την τιμή της υγρασίας εδάφους

void setup() {
Serial.begin(9600);

pinMode(moistsoilLEDPin,OUTPUT);//Ανάβει με την υγρασία
pinMode(lightLEDPin, OUTPUT);//Ανάβει με το φως
pinMode(balconyLightPin,OUTPUT);//Ανάβει τα φώτα του μπαλκονιού
pinMode(tentUpPin , OUTPUT); //Ανεβάζει τέντες τη νύχτα
pinMode(tentDownPin, OUTPUT);//Κατεβάζει τέντες τη μέρα
pinMode(wateringPin, OUTPUT);//Πότισμα φυτών του κήπου

//Set Relays to OFF
digitalWrite(balconyLightPin,HIGH);
digitalWrite(tentUpPin,HIGH);
digitalWrite(tentDownPin,HIGH);
digitalWrite(wateringPin,HIGH);
}

void loop() {
//

int dayLight=analogRead(dayLightPin);

```

```

Serial.print("dayLight : ");
Serial.println(dayLight);
if(dayLight>700){
    digitalWrite(balconyLightPin,HIGH);
    digitalWrite(tentUpPin,LOW);
    delay(1000); //Ο χρόνος να ρυθμιστεί ανάλογα
    digitalWrite(tentUpPin,HIGH);
    //
    Serial.println("SunLight->Tent Down");
    //digitalWrite(lightLEDPin,LOW);
}
else {
    digitalWrite(balconyLightPin,LOW);
    digitalWrite(tentDownPin,LOW);
    delay(1000); //Ο χρόνος να ρυθμιστεί ανάλογα
    digitalWrite(tentDownPin,HIGH);
    Serial.println("Dark->Tent Up");
    digitalWrite(lightLEDPin,HIGH);
}

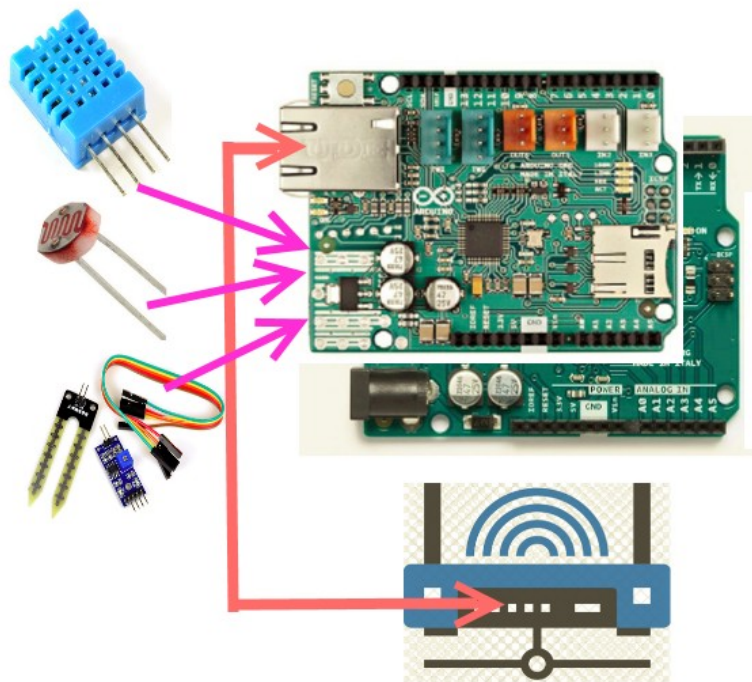
int moistureValue = analogRead(moisturePin);
//
if(moistureValue >= 700){

    //
    digitalWrite(wateringPin,LOW);
    digitalWrite(moistsoilLEDPin,HIGH);
    Serial.print("Low moisture (DRY!):");
    Serial.println(moistureValue);
}
if((moistureValue < 800) && (moistureValue >= 600)){
    //
    digitalWrite(wateringPin,LOW);
    digitalWrite(moistsoilLEDPin,HIGH);
    Serial.print("Medium moisture (MEDIUM !):");
    Serial.println(moistureValue);
}

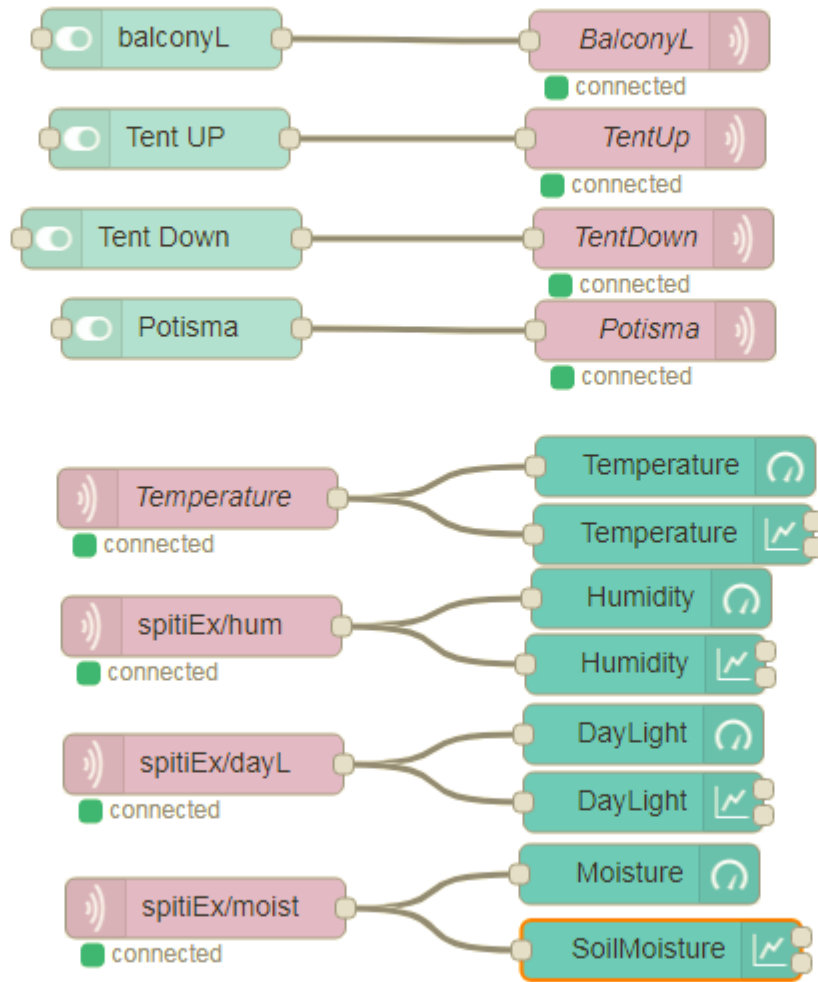
```



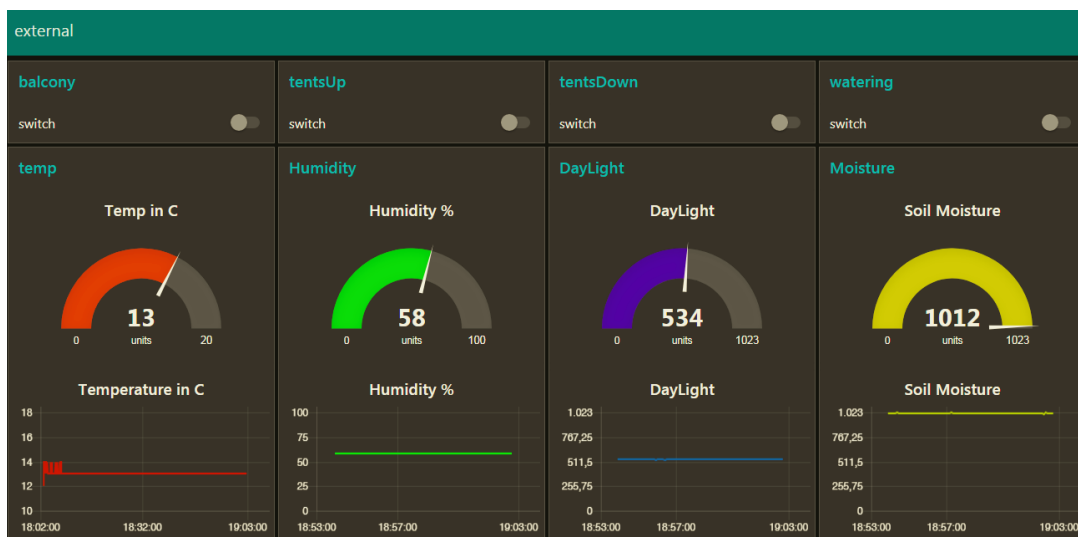
```
if(moistureValue < 600){  
  //  
  digitalWrite(wateringPin,HIGH);  
  digitalWrite(moistsoilLEDPin,LOW);  
  Serial.print("Normal moisture (NORMAL!):");  
  Serial.println(moistureValue);  
}  
delay (3000);  
}
```



Σχήμα 4.10 Arduino+Ethernet pub Server



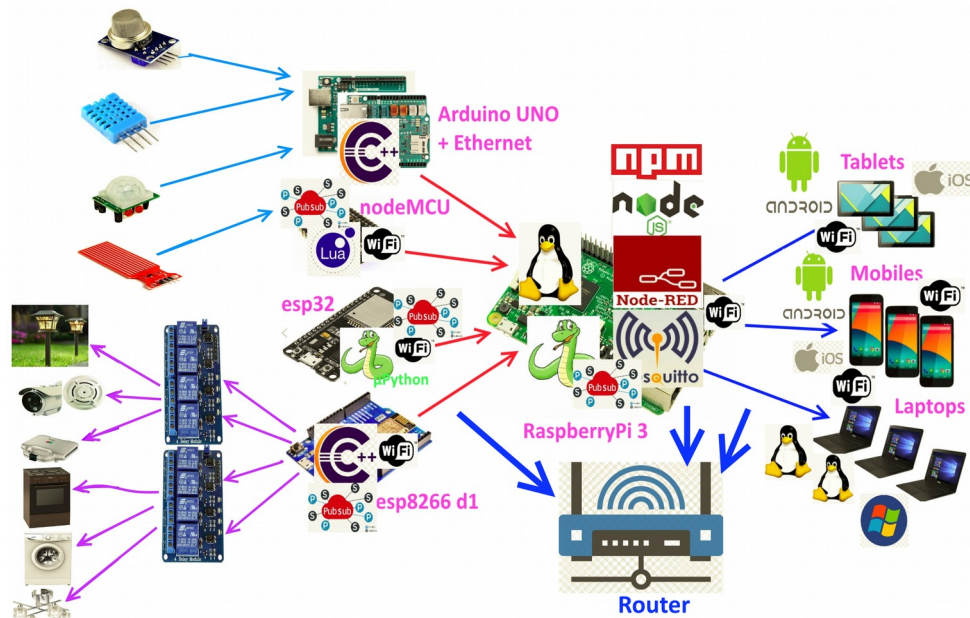
Σχήμα 4.11: Απεικόνιση ροών βελτιωμένου πίνακα NODE-RED Dashboard



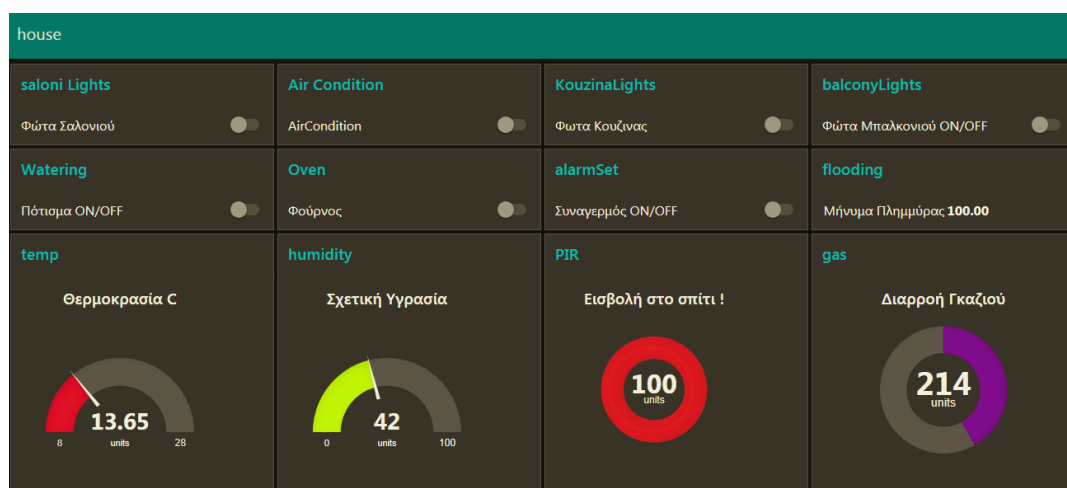
Σχήμα 4.12: Βελτιωμένος πίνακας ενδείξεων που εμπεριέχει γραφιστική γραμμική απεικόνιση

ΚΕΦΑΛΑΙΟ 5 - Έκδοση και Συνδρομή MQTT με server ESP8266

Αρχικός σχεδιασμός ήταν να αξιοποιήσουμε την πλακέτα ESP32, όμως λόγω καθυστέρησης θα αξιοποιήσουμε ως σύγχρονη έκδοχή την πλακέτα nodeMCU ESP8266 με γλώσσα προγραμματισμού C/C++, και (λίγο αργότερα), να προσαρμόσουμε τη λειτουργία σε mPython και LUA.



Σχήμα 5.1 Απεικόνιση των μερών της άσκησης 5



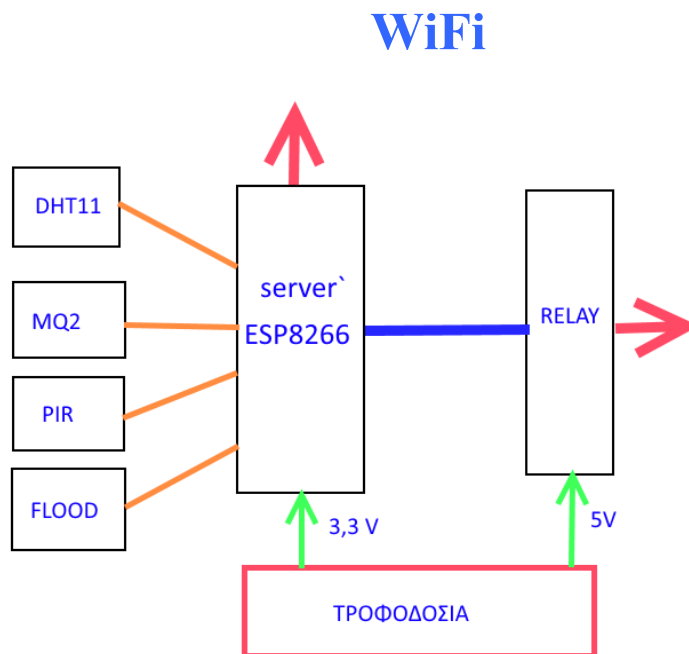
Σχήμα 5.2 Ο πίνακας εποπτείας και ελέγχου

Από ότι δείχνουμε στον παραπάνω πίνακα εποπτείας και ελέγχου, μπορούμε να ελέγξουμε: Τα φώτα του σαλονιού

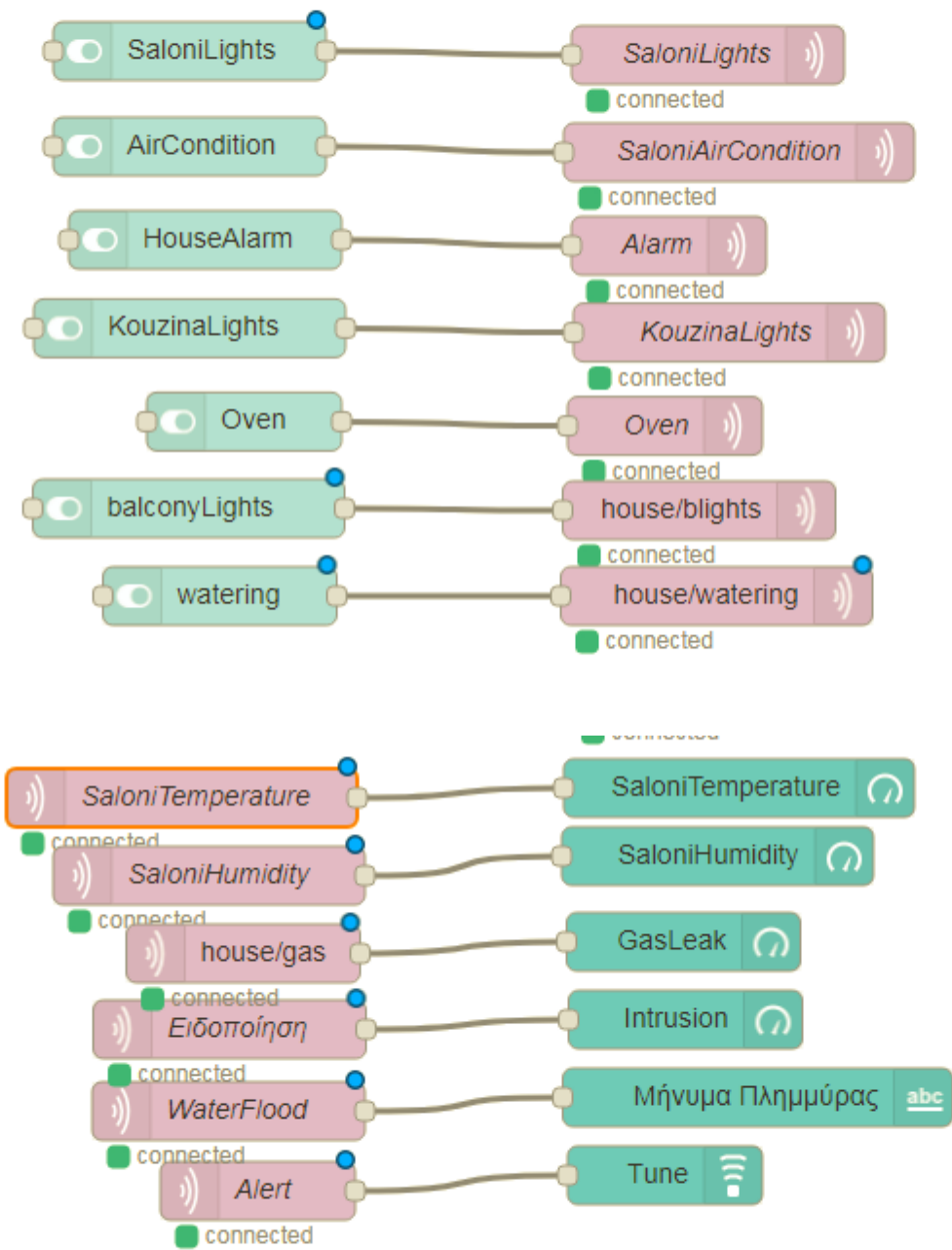
Το λειτουργία του κλιματιστικού
Τα φώτα της κουζίνας
Τα φώτα του μπαλκονιού
Την ηλεκτροβάννα ποτίσματος
Το φούρνο της κουζίνας
Να ενεργοποιήσουμε και απενεργοποιήσουμε το σύστημα συναγερμού

Να πληροφορηθούμε για:
Αν πλημμύρισε το δάπεδο
Τη θερμοκρασία στο εσωτερικό του σπιτιού
Τη σχετική υγρασία στο εσωτερικό του σπιτιού
Την εισβολή παρείσακτου επισκέπτη
Τη διαρροή γκαζιού στον εσωτερικό χώρο του σπιτιού

Το διάγραμμα φαίνεται στο ακόλουθο σχήμα



Σχήμα 5.3: διάγραμμα (block) της οργάνωσης των διατάξεων για την άσκηση 5



Σχήμα 5.4 Οι ροές των κόμβων Node-Red για την υλοποίηση της άσκησης 5

{Ο κώδικας στο παράρτημα κεφ 5}

ΚΕΦΑΛΑΙΟ 6 - Ο μικροΕλεγκτής ESP32

Στο κεφάλαιο αυτό θα παρουσιαστεί μια πλήρης εφαρμογή ελέγχου σύγχρονης κατοικίας μεν, με συμβατικές οικιακές συσκευές δε!

Δηλαδή οι συσκευές δεν έχουν προσαρμοσμένη διεπαφή IoT όπως συμβαίνει στην πλειονότητα των συσκευών αυτή τη στιγμή.

Στόχος μας είναι να δείξουμε στους μαθητές μας ότι δεν είναι καθόλου ανέφικτο να χρησιμοποιηθεί η τρέχουσα τεχνολογία και στη χώρα μας. Αλλά σήμερα!

Θα αξιοποιηθεί η Arduino like πλακέτα που εξοπλίζεται με τον πληρέστερο και ισχυρότερο επεξεργαστή ESP 32, (μικροελεγκτή) που υπάρχει αυτή τη στιγμή, σε τιμή εξαιρετικά προσιτή. Έτσι μπορεί καθένας να τον προμηθευτεί.

Επειδή η «έξοδός του» στην αγορά είναι πρόσφατη, δεν υπάρχουν πλήρεις βιβλιοθήκες που να τον υποστηρίζουν κάτω από το Arduino IDE και τη γλώσσα C/C++ .

Αυτό μας στοίχισε σε χρόνο, γιατί έπρεπε να «ανιχνεύσουμε» δυνατότητες με δοκιμή και λάθος, {χωρίς ακόμη να τον κάψουμε, αφού εξακολουθεί να λειτουργεί}, ελπίζοντας πως το αποτέλεσμα θα δικαιώσει όλους μας.

Ο ESP32 είναι πάνω από 100 φορές ισχυρότερος από τον AVR AT 328 που εξοπλίζει το κλασικό Arduino, διαθέτει WiFi και Bluetooth για διασύνδεση με το διαδίκτυο και άλλες συσκευές, διαθέτει τεράστια μνήμη RAM και μνήμη FLASH, ενσωματώνει τη στοίβα TCP/IP, διαθέτει πληθώρα ψηφιακών αλλά και αναλογικών θυρών.

Όλα αυτά με ελάχιστη κατανάλωση!

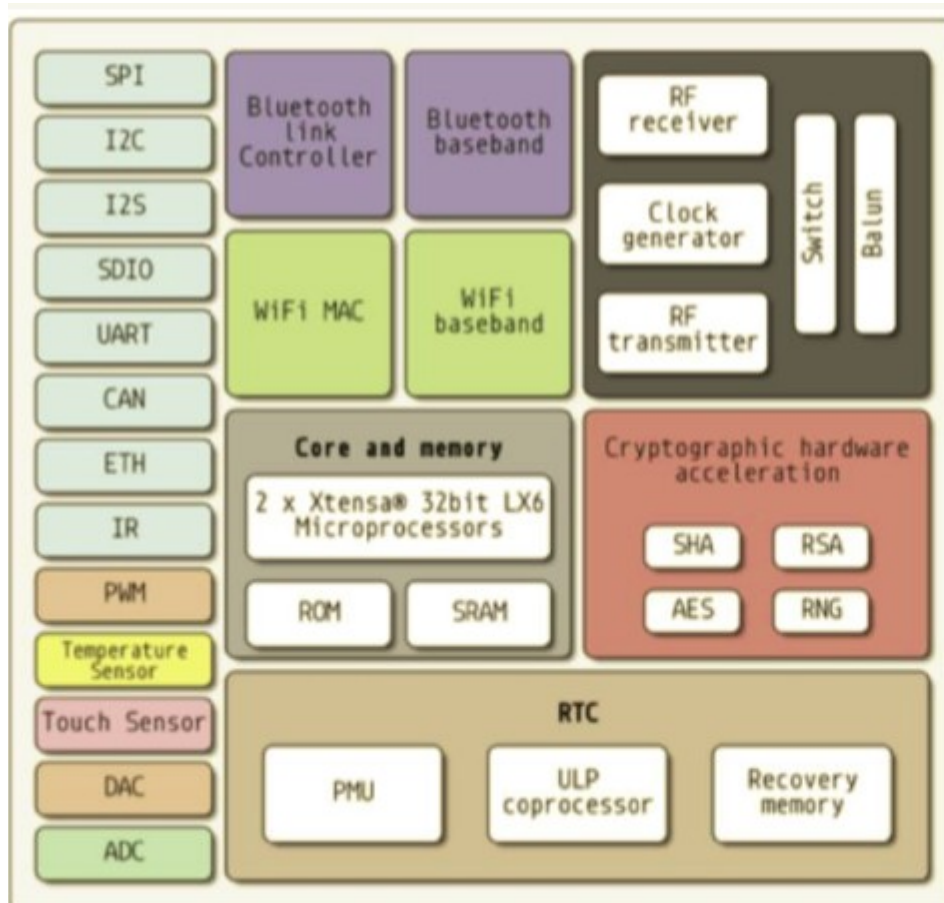
Έτσι νομίζουμε πως η μαθησιακή επένδυση των σχολείων μας σε αυτόν, αξίζει τον κόπο.

Στην εργασία αυτή υλοποιήθηκε η εφαρμογή αξιοποιώντας το IDE του Arduino και η γλώσσα C/C++ . Όμως η μέχρι τώρα έρευνα και μελέτη μας, έδειξαν ότι πρέπει να περάσουμε σε γλώσσα προγραμματισμού microPython, ίσως και σε LUA.

Είναι κάτι που θα επιχειρήσουμε πολύ σύντομα.



Σχήμα 6.1: Η Arduino like πλακέτα με τον ESP 32.



Σχήμα 6.2 : Η εσωτερική δομή του ESP32

Specifications	ESP8266	ESP32
MCU	Xtensa® Single-Core 32-bit L106	Xtensa® Dual-Core 32-bit LX6 600 DMIPS
802.11 b/g/n Wi-Fi	Yes, HT20	Yes, HT40
Bluetooth	None	Bluetooth 4.2 and below
Typical Frequency	80 MHz	160 MHz
SRAM	160 kBytes	512 kBytes
Flash	SPI Flash , up to 16 MBytes	SPI Flash , up to 16 MBytes
GPIO	17	36
Hardware / Software PWM	None / 8 Channels	1 / 16 Channels
SPI / I2C / I2S / UART	2/1/2/2	4/2/2/2
ADC	10-bit	12-bit
CAN	None	1
Ethernet MAC Interface	None	1
Touch Sensor	None	Yes
Temperature Sensor	None	Yes
Working Temperature	- 40°C ~ 125°C	- 40°C ~ 125°C

Σχήμα 6.2.1: Σύγκριση του ESP 8266 με τον ESP 32 [Espressif]

Κύρια χαρακτηριστικά του ESP32 (Key Features):

- 240 MHz dual core Tensilica LX6 microcontroller with 600 DMIPS
- Integrated 520 KB SRAM
- Integrated 802.11 b/g/n HT40 Wi-Fi transceiver, baseband, stack and LwIP
- Integrated dual mode Bluetooth (classic and BLE)
- 16 MB flash, memory-mapped to the CPU code space
- 2.3V to 3.6V operating voltage
- 40°C to +125°C operating temperature
- Onboard PCB antenna / IPEX connector for external antenna

34 x GPIO:

- 3 x UARTs, including hardware flow control
- 3 x SPI
- 2 x I2S
- 18 x ADC input channels
- 2 x DAC

- 2 x I2C
- PWM/timer input/output available on every GPIO pin
- OpenOCD debug interface with 32 kB TRAX buffer
- SDIO master/slave 50 MHz
- Supports external SPI flash up to 16 MB
- SD-card interface support

Security Related:

WEP, WPA/WPA2 PSK/Enterprise

Hardware accelerated encryption: AES/SHA2/Elliptical Curve Cryptography/RSA-4096

Performance:

Supports sniffer, Station, SoftAP and Wi-Fi direct mode

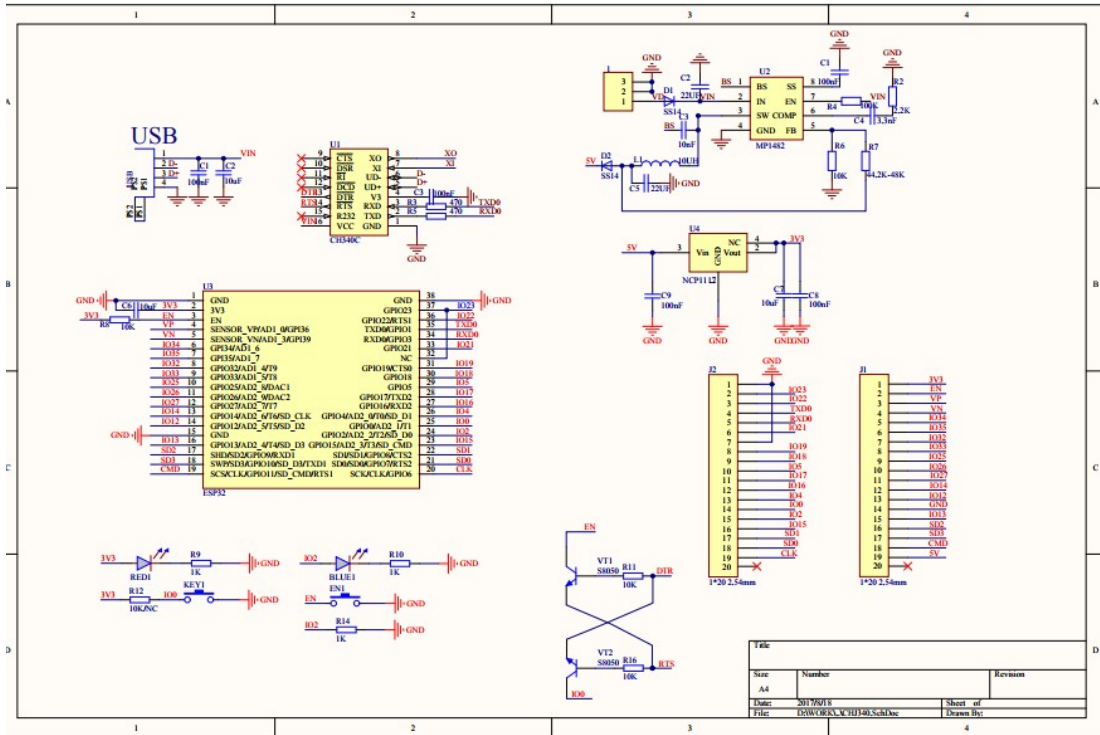
Max data rate of 150 Mbps@11n HT40, 72 Mbps@11n HT20, 54 Mbps@11g, and 11 Mbps@11b

Maximum transmit power of 19.5 dBm@11b, 16.5 dBm@11g, 15.5 dBm@11n

Minimum receiver sensitivity of -97 dBm

135 Mbps UDP sustained throughput

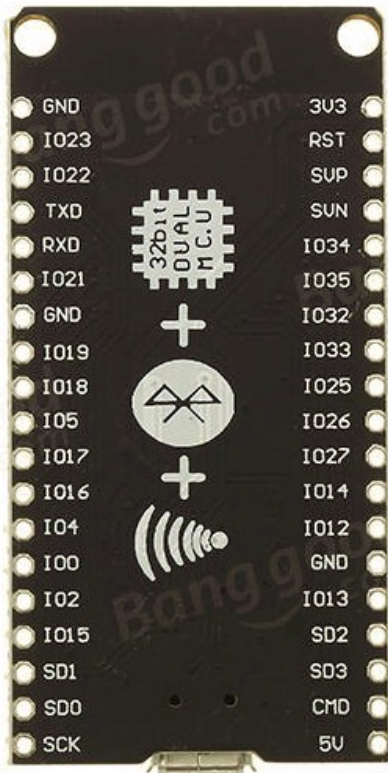
5 μ A power consumption in Deep-sleep



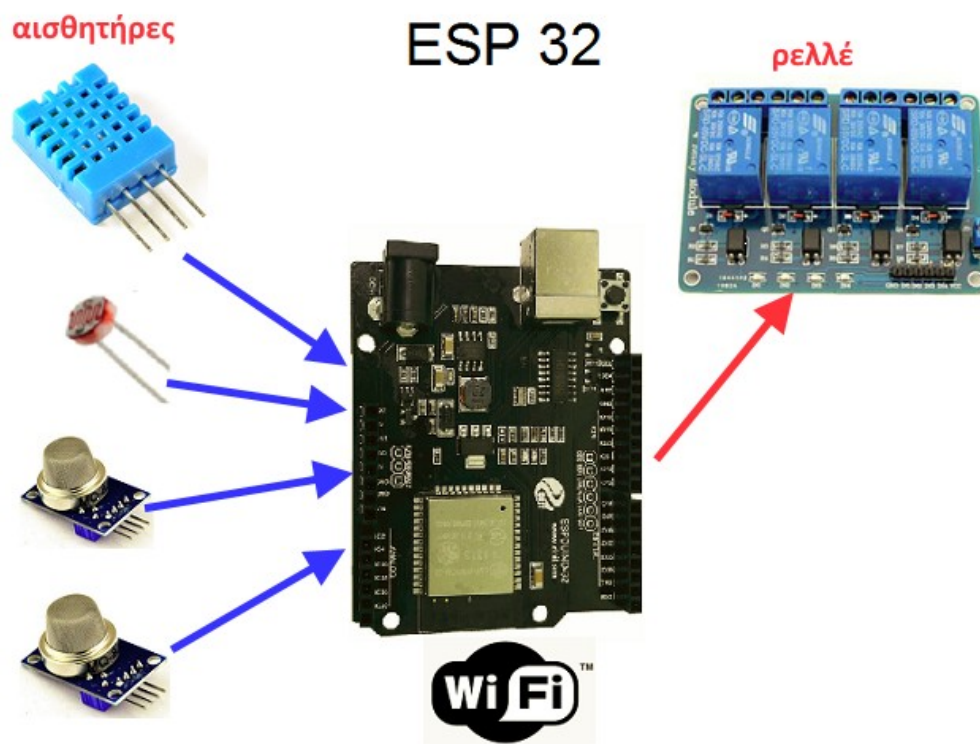
Σχήμα 6.3 : Ηλεκτρονικό διάγραμμα πλακέτας ESP 32



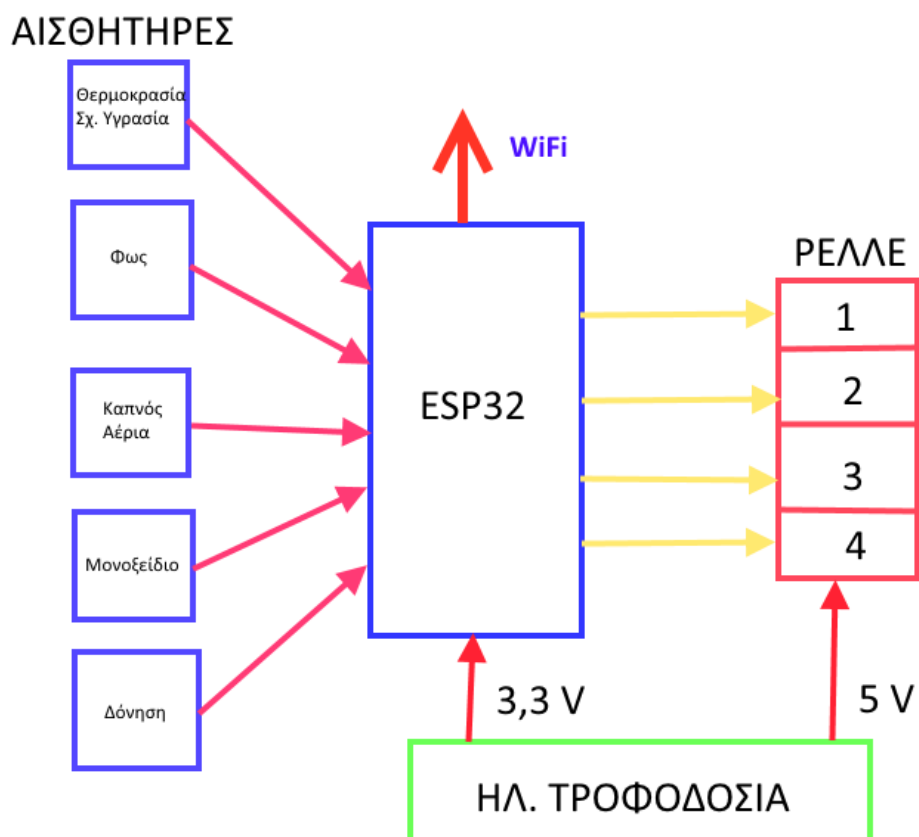
Σχήμα 6.4 :ESP32 DEV KIT DEV BOARD



Σχήμα 6.5:ESP 32 DEV BOARD BACKPLANE

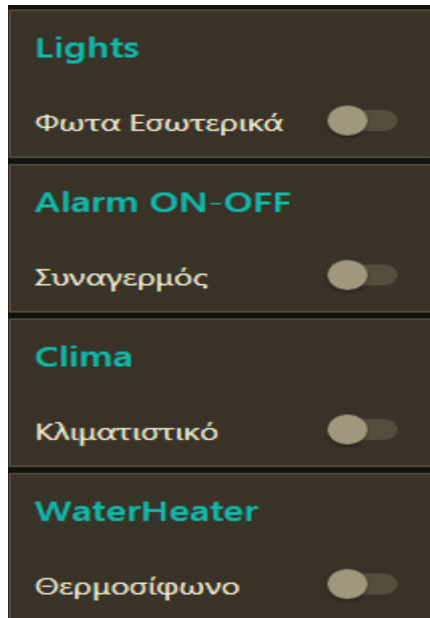


Σχήμα 6.6: εικονίζονται τα μέρη που συνθέτουν την άσκηση του κεφαλαίου 6



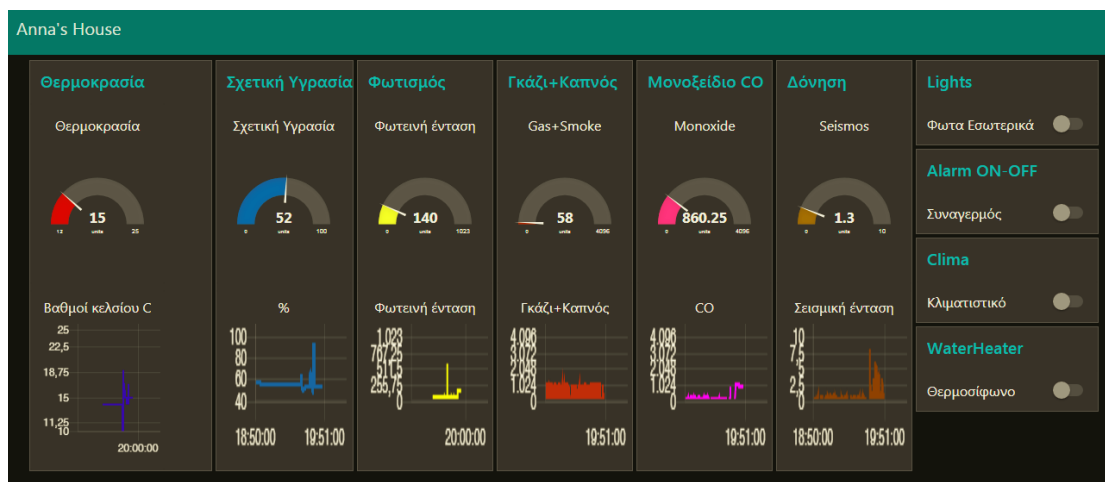
Σχήμα 6.7 : Διάγραμμα σε μπλοκ εργασίας κεφαλαίου

Στην εργασία περιελήφθησαν μία ομάδα εικονικών διακοπών που ελέγχουν τη λειτουργία των συνηθέστερων συσκευών που υπάρχουν σε μια κατοικία, χωρίς όμως να σημαίνει ότι δεν έχουν εφαρμογή στη βιοτεχνία, τους ποικίλους χώρους εργασίας ή οπουδήποτε κριθεί αναγκαίο. Ο έλεγχος γίνεται μέσω οποιασδήποτε συσκευής που μπορεί να τρέξει ένα σύγχρονο φυλλομετρητή και να συνδεθεί με οποιονδήποτε τρόπο στο διαδίκτυο από οποιοδήποτε σημείο του πλανήτη.



Σχήμα 6.8: Ομάδα εικονικών διακοπών που ελέγχουν οικιακές συσκευές

Παράλληλα «εκδίδονται» πληροφορίες που αποκτώνται με τη βοήθεια κατάλληλων αισθητήρων που με τη σειρά τους, συνδέονται στις εισόδους (κυρίως τις αναλογικές), του μικροελεγκτή. Ταυτόχρονα καταγράφονται χρονικά τα γεγονότα σε αντίστοιχες καμπύλες. Η απεικόνισή τους γίνεται με γραφιστικό τρόπο και φαίνεται στο σχέδιο 6.9

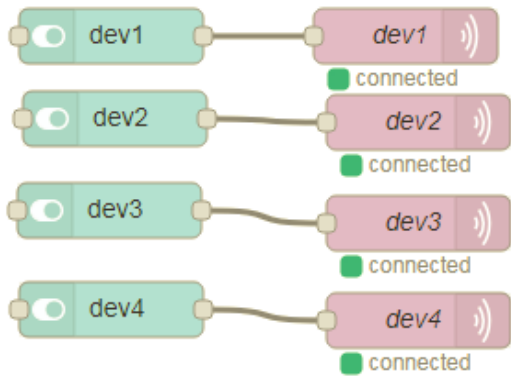


Σχέδιο 6.9 Η πλήρης απεικόνιση του πίνακα εποπτείας και ελέγχου.

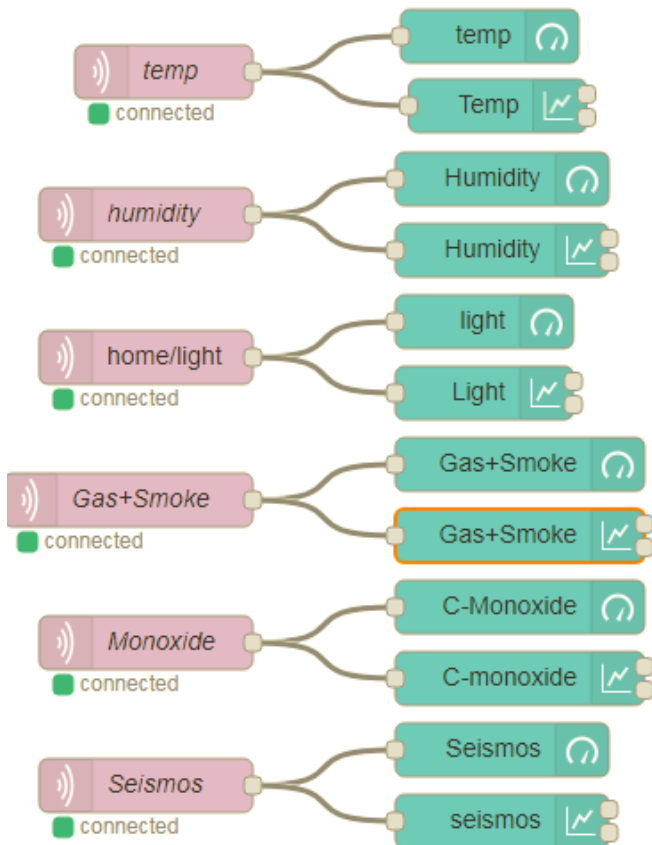
Επιδιώξαμε την εποπτεία σε θέματα απλά όπως η θερμοκρασία και η σχετική υγρασία, αλλά και πιο σύνθετα όπως ξαφνικός φωτισμός σε ανέλπιστο χρόνο, διαρροή στον εσωτερικό χώρο επικίνδυνων

αερίων, ύπαρξη καπνού, ύπαρξη μονοξειδίου του άνθρακα και τελικώς την ενημέρωσή μας και ταυτόχρονη καταγραφή της έντασης τυχόντος σεισμού στο συγκεκριμένο χώρο.

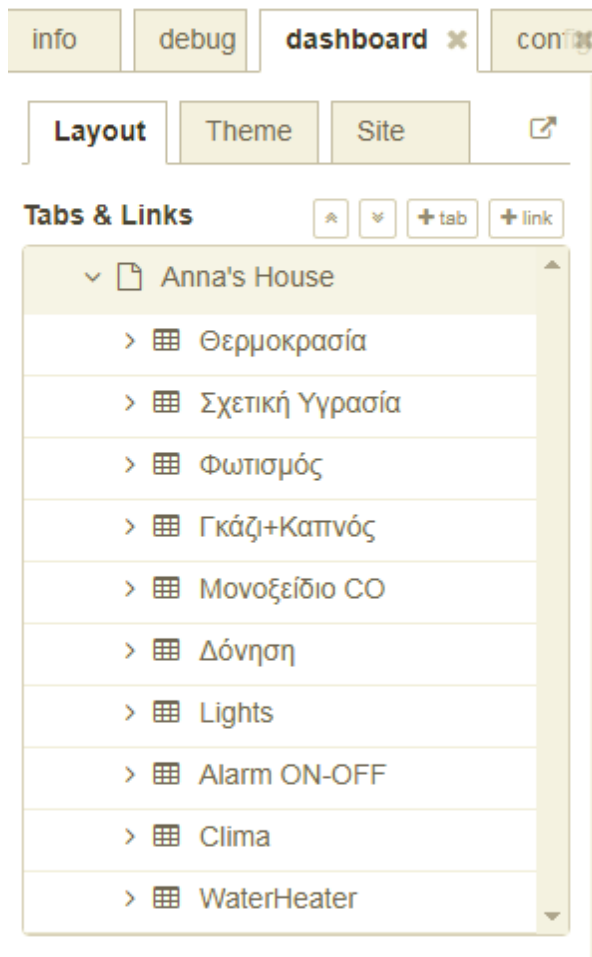
Η απεικόνιση των ροών NODE-RED φαίνεται στο σχήμα 6.10 & 6.11



Σχήμα 6.10 Οι ροές για την έκδοση (Publish) των χειρισμών στο NODE-RED.



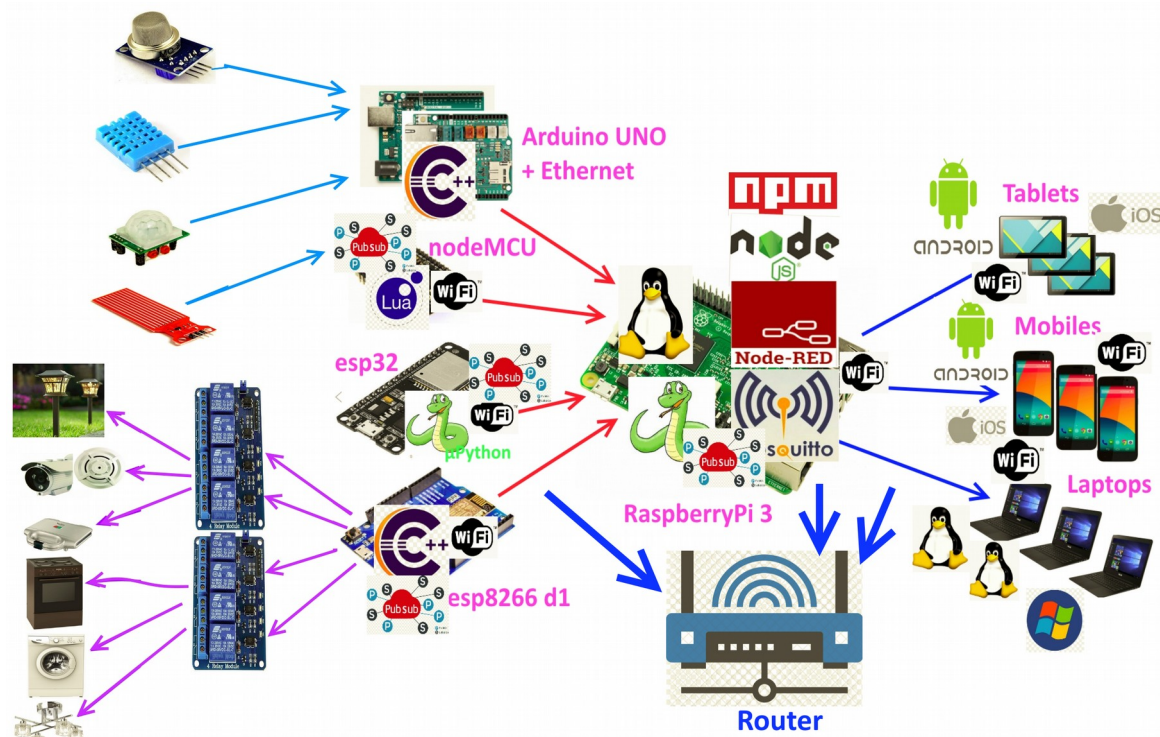
Σχήμα 6.11 οι ροές εγγραφής (Subscription) στα επιλεγμένα θέματα με NODE-RED.



Σχήμα 6.12 :Η οργάνωση (layout) του πίνακα εποπτείας και ελέγχου

ΚΕΦΑΛΑΙΟ 7 - Μεθοδολογία Λειτουργίας

Η παρακάτω εικόνα (σχήμα 7.1) δείχνει το σύνολο του έργου.



Σχήμα 7.1 Το σύνολο της εργασίας

Η διαδοχή των ενεργειών για τη λειτουργία των ασκήσεων

Απαιτούμενες διατάξεις και συσκευές

Φορητός υπολογιστής Windows

Πλακέτα Raspberry Pi3 Linux, Mosquitto, Node-Red

Internet Router WiFi, DHCP, NAT

Server NodeMCU ESP8266, IP stack, WiFi, Application

Τροφοδοτούνται και ενεργοποιούνται όλες οι συσκευές με την εξής σειρά:

Router -> PC -> RaspberryPi3 -> Server

Όλες οι διατάξεις πρέπει να συνδεθούν στο δίκτυο του συγκεκριμένου Router δίνοντας ssid και password.

Τότε λογικά το DHCP θα τους εκχωρήσει τις ακόλουθες IP διευθύνσεις:

Router: 192.168.1.1 ως Gateway

PC : 192.168.1.2

Raspberry3 : 192.168.1.3 (ενδιαφέρει επειδή εδώ τρέχει το mosquito Broker)

Server : 192.168.1.4

Αυτό πρέπει να διαπιστωθεί τρέχοντας το IPScan στο φορητό υπολογιστή.

Η διεύθυνση που ενδιαφέρει είναι εκείνη που εκχωρείται στο RaspberryPi3 το οποίο τρέχει το Mosquitto Broker.

Στον κώδικα ενσωματώθηκε το ssid και το password του συγκεκριμένου Internet Router που συνοδεύει την εφαρμογή. Σε περίπτωση αντικατάστασής του πρέπει να αλλάξουν με τα δεδομένα του νέου Router.

Στον κώδικα της τρέχουσας εργασίας, η διεύθυνση του Raspberry3 που τρέχει το Mosquitto Broker είναι: 192.168.1.3.

Αν τρέχοντας το IPscan η διεύθυνση του Raspberry3 είναι διαφορετική τότε πρέπει στον κώδικα να προστεθεί η νέα, να ξαναγίνει μετάφραση και φόρτωση του νέου κώδικα.

Άλλος τρόπος για να δούμε την IP διεύθυνση είναι να δώσουμε σε command mode: **ip addr disp**

Προσοχή : Αν άλλη φορητή συσκευή όπως π.χ. το κινητό τηλέφωνο είχε ενεργή σύνδεση με το δρομολογητή μέσω WiFi, τότε πρέπει πριν αρχίσουμε την ενεργοποίηση της άσκησης να το διακόψουμε.

Ακολούθως μετά το στήσιμο της εφαρμογής μπορεί να επανενεργοποιηθεί και να του εκχωρηθεί από το DHCP νέα IP διεύθυνση.

Στο Raspberry3 ανοίγουμε παράθυρο Command Mode και τρέχουμε: **mosquitto -v**.

Ακολούθως σε νέο παράθυρο CM τρέχουμε : **node-red**

Αφήνουμε τα παράθυρα ενεργά.

Στο browser του Rpi καλούμε : **localhost:1880** για να ενεργοποιηθεί το σύστημα ανάπτυξης Node-Red και η εφαρμογή μας.

Στο browser του PC καθώς και σε εκείνον των φορητών συσκευών (smartphones και tablets) καλούμε τη διεύθυνση : 192.168.1.X:1880/ui για να ενεργοποιήσουμε σε αυτές τη γραφιστική διεπαφή του χρήστη δημιουργημένη από την εφαρμογή Node-Red.

(όπου X, ο αριθμός της IP που παίρνουμε από το IPscan)

Τώρα μπορούμε να παρακολουθούμε και να ελέγχουμε τις συσκευές της κατοικίας.

ΚΕΦΑΛΑΙΟ 8 - ΣΥΜΠΕΡΑΣΜΑΤΑ

Διαλέξαμε να υλοποιήσουμε το παρόν διδακτικό έργο γιατί θεωρήσαμε ότι η πρακτική εμπλοκή των μαθητών με την γνώση μέσω της έρευνας, μελέτης σχεδιασμού και κατασκευής θα τους βοηθήσει στην βαθύτερη κατανόηση της πληροφορίας στην μακροχρόνια μνήμη τους , θα καλλιεργήσει την δημιουργική τους σκέψη και τις κοινωνικές δεξιότητες τους. Εξάλλου, ότι τα παιδιά μπορούν να κάνουν μαζί σήμερα, μπορούν να κάνουν μόνα τους αύριο (Vygotsky, 1962)). Οι τεχνολογικές κατασκευές ενισχύουν το ενδιαφέρον των παιδιών για την μάθηση και με ευχάριστο τρόπο τα εμπλέκουν με βασικές έννοιες της πληροφορικής τόσο σε επίπεδο Hardware όσο και σε επίπεδο Software. Με τις χρηστικές ηλεκτρονικές κατασκευές επιδιώκουμε τον εντυπωσιασμό των μαθητών ώστε να ενθαρρυνθούν και να οδηγηθούν στην απόκτηση της γνώσης ψηφιακών και κοινωνικών δεξιοτήτων κάτι που αποτελεί κύριο συστατικό του επαγγελματικού και κοινωνικού τους βίου.

BIBΛΙΟΓΡΑΦΙΑ

- [1] Κυνηγός, Χ. και Φράγκου, Σ. (2000), *Παιδαγωγική Αξιοποίηση της Τεχνολογίας Ελέγχου στη Τάξη*, Στο: Β.Ι. Κόμης (επιμ.): Πρακτικά του 2ου Πανελληνίου Συνεδρίου με Διεθνή Συμμετοχή “Οι Τεχνολογίες της Πληροφορίας και της Επικοινωνίας στην Εκπαίδευση”, Πάτρα, 83-91.
- [2] Bruner, J.,(1986). *Actual minds, possible worlds*. Cambridge, MA: Harvard University Press.xi,201 pages.
- [3] Bruner J.(1997). Πράξεις νοήματος, Ελληνικά Γράμματα.
- [4] Edwards, D. ,and Mercer, N.(1987).Common knowledge. *The development of understanding in the classroom*. London: Methuen.
- [5] Harel, I.,(1991). *Children Designers. Interdisciplinary Constructions for Learning and Knowing Mathematics in a Computer-Rich School*. Ablex Publishing Corporation, Norwood, New Jersey.
- [6] Kafai, Y., & Resnick, M.(1996). *Constructionism in Practice: Designing, Thinking and Learning a Digital World*. Hillsdale, NS: Erlbaum Associates.
- [7] Papert, S.,(1980), Νοητικές θύελλες : *Παιδιά, ηλεκτρονικοί υπολογιστές και δυναμικές ιδέες*, Εκδόσεις Οδυσσέας (Ελληνική Μετάφραση 1991).
- [8] Papert,S.,(1993). *The children’s Machine: Rethinking Schools in the Age of the Computers*, Basic Books.
- [9] Piaget, J., (1974), *To Understand Is To Invent*. N.Y.: Basic Books.
- [10] Vygotsky, L.S.,(1962). *Thought and language* Cambridge, MA: The MIT Press. Published originally in Russian in 1934.
- [11] Vygotsky, L. (1978). *Interaction Between. Learning and Development*. In Gauvain &. Cole (Eds) Readings on the Development of Children. New York: Scientific.
- [12] Lund,H.H., & Nielsen, S.(2002). *An Edutainment Robotics Survey. In Proceedings of the Third International Symposium on Human and Artificial Intelligence Systems: The Dynamic Systems Approach for Embodiment and Sociality*, Fukui.Papert, S. (1991) *.Situating Constructionism*. In S.Papert and I.Harel (eds.) Constructionism, Norwood, NJ, Ablex Publishing Corporation.
- [13] Resnick, M. and Silverman, B. (2005), *Some Reflections on Designing Construction Kits for Kids*. Proceedings of Interaction Design and Children conference, Boulder, CO, ανακτήθηκε 2/1/2008 από <http://ilk.media.mit.edu/papers/IDC-2005.pdf>
- [14] Brazi,Rossano : Getting Started with Arduino 3rd edition, Make
Jeremy Blum :**Exploring Arduino®: Tools and Techniques for Engineering Wizardry** John Wiley & Sons, Inc.
- [15] Arduino Bots and Gadgets, Make

- [16] Making Things Talk, Third Edition ,Make
- [17] [Don Wilcher :Arduino Electronics Blueprints .Make](#)
- [18] [Simon Monk](#):Programming Arduino: Getting Started with Sketches, Second Edition (Tab books)
- [19] [Mark Geddes](#):Arduino Project Handbook: 25 Practical Projects to Get You Started
- [20] [Jody Culkin](#) and [Eric Hagan](#): Learn Electronics with Arduino: An Illustrated Beginner's Guide to Physical Computing (Make)
- [21] [Gaston C. Hillar :MQTT Essentials - A Lightweight IoT Protocol, Pact](#)
- [22] [Building Real-time Mobile Solutions With Mqtt and IBM Messagesight](#)
IBM Redbooks
- [23] [Pradeeka Seneviratne](#): ESP8266 Robotics Projects: DIY Wi-Fi controlled robots
- [24] [Marco Schwartz: ESP8266 Internet of Things Cookbook: Over 50 recipes to help you master ESP8266 functionality Pact](#)
- [25] [ESP8266: Programming NodeMCU Using Arduino IDE - Get Started With ESP8266 \(Internet Of Things, IOT, Projects In Internet Of Things, Internet Of Things for Beginners, NodeMCU Programming, ESP8266\)](#)
UpSkill Learning
- [26] [Shantanu Bhadoria and Ruben Oliva Ramos: Raspberry Pi 3 Home Automation Projects: Bringing your home to life using Raspberry Pi 3, Arduino, and ESP8266](#)
- [27] [Catalin Batrinu ESP8266 Home Automation Projects: Leverage the power of this tiny WiFi chip to build exciting smart home projects](#)
- [28] [Praveenkumar Natarajan](#) :How to program ESP8266 in lua: With Nodemcu dev kit
- [29] [Richard Blum](#) :ESP8266 Arduino IDE Guide: Internet of Things with ESP8266(Nodemcu)
- [30] ESP8266 Get Started with ESP8266 Programming NodeMCU Using Arduino IDE ,UpSkill Learning
- [31] Learning, UpSkill (2016-06-20). ESP8266: Programming NodeMCU Using Arduino IDE - Get Started With ESP8266 (Internet Of Things, IOT, Projects In Internet Of Things, Internet Of Things for Beginners, NodeMCU Programming, ESP8266)
- [32] David Leithauser: Remote Sensor Monitoring by Radio with Arduino: Detecting Intruders, Fires, Flammable and Toxic Gases, and Other Hazards at a Distance
- [33] Leithauser, David: Remote Sensor Monitoring by Radio with Arduino: Detecting Intruders, Fires, Flammable and Toxic Gases, and Other Hazards at a Distance , Leithauser Research.
- [34] Authors, Instructables (2011-01-17). Arduino Projects Instructables.com. Kindle Edition.
- [35] Instructable: How To Smell Pollutants Author: liseman License: Attribution-ShareAlike
- [36] Authors, Instructables (2011-01-17). Arduino Projects Instructables.com. Kindle Edition.
- [37] Elton Stoneman: Hive Succinctly Synconfusion 2016
- [38] <http://noderedguide.com/nr-lecture-1/>

Μονάδα Joystick

Αισθητήρας φλόγας

Επιταχυνσιόμετρο τριπλού άξονα MMA7361

Ψηφιακός αισθητήρας κίνησης υπερύθρων

Sharp GP2Y0A21 αισθητήρας απόστασης (10-80cm)

Αισθητήρας υγρασίας εδάφους

ΠΑΡΑΡΤΗΜΑ ΙΙ- MQTT

Εισαγωγή στο MQTT

Το MQTT είναι ένα ελαφρύ πρωτόκολλο ανταλλαγής μηνυμάτων για έκδοση / εγγραφή που έχει σχεδιαστεί για τηλεμετρία M2M (από μηχανή σε μηχανή) σε περιβάλλοντα χαμηλού εύρους ζώνης.



Σχήμα Μ.1 : έκδοση-συνδρομή

Σχεδιάστηκε από τον Andy Stanford-Clark (IBM) και την Arlen Nipper το 1999 για τη σύνδεση των συστημάτων τηλεμετρίας πετρελαιαγωγών μέσω δορυφόρου.

Αν και ξεκίνησε να λειτουργεί ως πρωτόκολλο ιδιωτικό, κυκλοφόρησε το έτος 2010 δωρεάν και έγινε πρότυπο του οργανισμού OASIS το 2014.

Το MQTT εξελίσσεται γρήγορα ως ένα από τα κύρια πρωτόκολλα ανάπτυξης εφαρμογών του Διαδικτύου των πραγμάτων IOT

Υπάρχουν δύο εκδόσεις του MQTT.

Το αρχικό MQTT, το οποίο σχεδιάστηκε το 1999 και έχει χρησιμοποιηθεί εδώ και πολλά χρόνια.

Έχει σχεδιαστεί για λειτουργία σε αξιόπιστα δίκτυα TCP / IP.

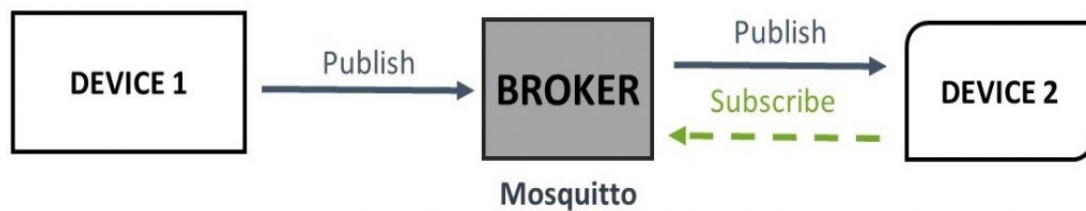
Η προδιαγραφή βρίσκεται: [MQTT V3.1](#) και θα μπορείτε να βρείτε μια πιο λεπτομερή επισκόπηση της [δομής πακέτων MQTT](#).

Το MQTT-SN που καθορίστηκε περίπου το 2013 και έχει σχεδιαστεί για να λειτουργεί με πρωτόκολλα μη αξιόπιστης μετάδοσης όπως είναι το UDP.

Το MQTT-SN δεν φαίνεται να είναι πολύ δημοφιλές σήμερα. Και η προδιαγραφή δεν έχει αλλάξει για αρκετά χρόνια, αλλά πιστεύουμε ότι θα αλλάξει καθώς αρχίζουν να πολλαπλασιάζονται οι εφαρμογές IOT που τρέχουν πάνω στον ακραίο κορμό των δικτύων.

Λειτουργία του MQTT

Το MQTT χρησιμοποιεί ένα μοντέλο δημοσίευσης / εγγραφής (Pub/Sub) το οποίο απαιτεί τη χρήση κεντρικού μεσίτη (Broker), όπως φαίνεται στο παρακάτω διάγραμμα:



Σχήμα Μ.2:η συσκευή 1 εκδίδει, ο μεσίτης το διαχειρίζεται και η συσκευή 2 το παραλαμβάνει.

Οι πελάτες δεν έχουν διευθύνσεις όπως στα συστήματα ηλεκτρονικού ταχυδρομείου και τα μηνύματα δεν αποστέλλονται στους πελάτες.

Αντ' αυτού τα μηνύματα δημοσιεύονται για μεσολάβηση σε ένα θέμα (Topic).

Το έργο ενός μεσίτη MQTT είναι να φιλτράρει τα μηνύματα με βάση το θέμα και στη συνέχεια να τα διανείμει στους συνδρομητές.

Ένας πελάτης μπορεί να λάβει αυτά τα μηνύματα δηλώνοντας το συγκεκριμένο θέμα στον συγκεκριμένο μεσίτη.

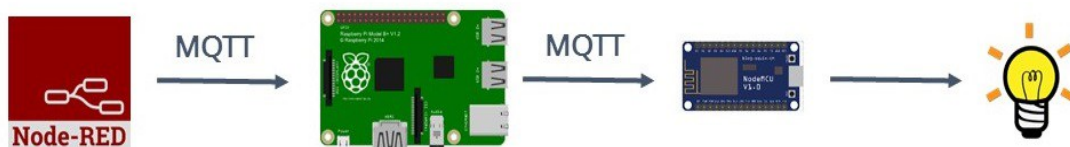
Σε αυτό το μοντέλο δεν υπάρχει άμεση σύνδεση μεταξύ εκδότη και συνδρομητή.

Το μοντέλο είναι παρόμοιο με το ραδιόφωνο ή την τηλεόραση. Όταν ένας τηλεοπτικός ή ραδιοφωνικός σταθμός εκπέμπει σε ένα κανάλι, τότε την πληροφορία θα λάβουν οι ακροατές / θεατές που συντονίζονται σε αυτό.

Ωστόσο, σε αντίθεση με το μοντέλο εκπομπής τηλεόρασης / ραδιοφώνου στο MQTT, όλοι οι πελάτες μπορούν να γίνουν Publishers, δηλαδή να δημοσιεύσουν (εκπομπή) και Subscribers, δηλαδή να εγγραφούν για να λαμβάνουν τις εκπομπές των άλλων.

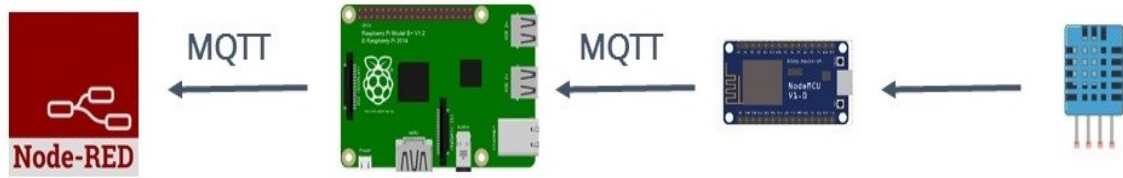
Συνήθως αποστέλλεται πακέτο μηνύματος από πελάτη, (όπως η εφαρμογή Node-Red) για να ελέγξει η λειτουργία απομακρυσμένης συσκευής, όπως για παράδειγμα τα φώτα του σαλονιού στη κατοικία μας.

Ο αποστολέας του μηνύματος ονομάζεται εκδότης (Publisher) και ο παραλήπτης (Subscriber)



Σχήμα Μ.3 NODE-RED εκδότης μηνύματος ESP8266 συνδρομητής παραλήπτης μηνύματος

Μπορεί κατ' αντιστοιχία να διαβάσουν τα δεδομένα από έναν αισθητήρα και να δημοσιευτούν σε έναν πελάτη όπως στην εικόνα το Node-Red.



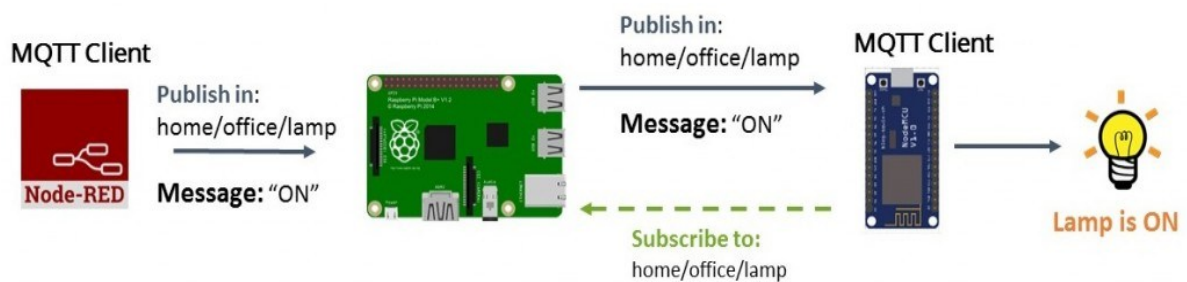
Σχήμα Μ.4 ESP8266 εκδότης NODE-RED συνδρομητής-παραλήπτης

Τα Θέματα Topics- Διευθυνσιοδότηση MQTT

Τα θέματα (Topics) είναι αυτά που συνδέουν τον εκδότη και τον συνδρομητή.

Στο MQTT δεν υπάρχει επίσημη δομή και ένας εκδότης είναι ελεύθερος να επιλέξει τα δικά του ονόματα και δομή.

Ωστόσο, υπάρχουν κανόνες.



Σχήμα Μ.5 : έκδοση/ παραλαβή θέματος

Τα θέματα δημιουργούνται από τον συνδρομητή και τον εκδότη και δεν εκχωρούνται εκ των προτέρων από τον μεσίτη Broker.

Όσο ένα όνομα θέματος ακολουθεί τα πρότυπα ονομασίας, είναι αποδεκτό από τον Broker.

Το όνομα του θέματος αποτελείται από μια συμβολοσειρά στον κώδικα UTF-8 και πρέπει να περιλαμβάνει τουλάχιστον ένα χαρακτήρα.

Τα θέματα μπορούν να δημιουργηθούν με έναν ιεραρχικό τρόπο (επίπεδα) χρησιμοποιώντας μια εμπρόσθια κάθετο ως διαχωριστή.

Τα απλά παραδείγματα θεμάτων είναι:

θέμα = saloni, θέμα =koyzina, θέμα =balkoni ή

θέμα = saloni/fota, θέμα =koyzina/oven, θέμα =balkoni/tents

Τα δημοσιευμένα μηνύματα

Όταν ένας πελάτης δημοσιεύει ένα μήνυμα σε ένα θέμα, τότε ο μεσίτης θα διανείμει αυτό το μήνυμα σε οποιονδήποτε συνδεδεμένο πελάτη που έχει εγγραφεί στο συγκεκριμένο θέμα.

Μόλις το μήνυμα αποστέλλεται στους πελάτες αυτούς, ο μεσίτης το διαγράφει.

Εάν κανένας πελάτης δεν έχει εγγραφεί στο θέμα ή δεν είναι συνδεδεμένος, τότε το μήνυμα διαγράφεται από το μεσίτη. Σε γενικές γραμμές ο μεσίτης δεν αποθηκεύει μηνύματα εκτός από την αποθήκευση μηνυμάτων στον μεσίτη / διακομιστή.

Συνδέσεις μεσίτη MQTT

Το MQTT χρησιμοποιεί το πρωτόκολλο TCP/IP για να συνδεθεί με τον μεσίτη. Το πρωτόκολλο TCP είναι ένα πρωτόκολλο επικύρωσης της σύνδεσης με διόρθωση σφαλμάτων και εγγυάται ότι τα πακέτα παραλαμβάνονται κατά σειρά.

Ωστόσο, αυτό δεν σημαίνει ότι τα μηνύματα δεν μπορούν να χαθούν.

Επίπεδα ποιότητας υπηρεσιών MQTT QoS

Τα δίκτυα είναι από τη φύση τους αναξιόπιστα, έτσι το MQTT επιτρέπει την επιλογή 3 επιπέδων ποιότητας υπηρεσιών QoS ανάλογα με τις απαιτήσεις της εφαρμογής. Εάν η εφαρμογή μπορεί να ανεχτεί και να λειτουργήσει με απώλεια μηνυμάτων, τότε μπορεί να επιλεγεί το χαμηλότερο επίπεδο QoS 0.

Διαφορετικά, θα χρειαστεί η χρήση QoS επιπέδου 1 (τουλάχιστον μία φορά) ή επιπέδου QoS 2 (το πολύ μία φορά).

Όσο υψηλότερο είναι το επίπεδο QoS, τόσο μεγαλύτερο είναι το κόστος των μηνυμάτων σε μέγεθος και χρονική διάρκεια εκπομπής.

Πελάτες MQTT

Επειδή οι πελάτες του MQTT δεν έχουν διευθύνσεις όπως αυτές του ηλεκτρονικού ταχυδρομείου, ή του τηλεφωνικού δικτύου, δεν χρειάζεται να αντιστοιχίζετε διευθύνσεις σε πελάτες.

Μεσίτες ή διακομιστές MQTT

Ο αρχικός όρος ήταν μεσίτης, αλλά πλέον τυποποιήθηκε ως διακομιστής. Παρόλα αυτά θα διαπιστώσετε ότι χρησιμοποιούνται σήμερα και οι δύο όροι.

Υπάρχουν πολλοί μεσίτες MQTT που μπορείτε να χρησιμοποιήσετε τόσο για δοκιμές όσο και για πραγματικές εφαρμογές.

Για ερασιτεχνική εφαρμογή δημοφιλέστερος είναι το [Mosquitto](#) και για σοβαρότερες εμπορικές εφαρμογές το [HiveMQ](#).

Το Mosquitto είναι ένας δωρεάν μεσίτης ανοιχτού κώδικα MQTT που λειτουργεί με Windows και Linux.

Εάν δεν θέλετε να εγκαταστήσετε και να διαχειριστείτε τον δικό σας μεσίτη, μπορείτε να χρησιμοποιήσετε έναν μεσίτη δοκιμών όπως το test.mosquitto.org ή iot.eclipse.org

Ασφάλεια του MQTT

Το MQTT υποστηρίζει διάφορους μηχανισμούς ελέγχου ταυτότητας και ασφάλειας δεδομένων. Είναι σημαντικό να σημειωθεί ότι αυτοί οι μηχανισμοί ασφαλείας έχουν διαμορφωθεί στον μεσίτη MQTT και εξαρτάται από τον πελάτη να συμμορφώνεται με τους μηχανισμούς που υπάρχουν.

Η έκδοση 5 του MQTT

Η επόμενη έκδοση του MQTT θα είναι η έκδοση 5 που την ώρα που γράφεται αυτό το κείμενο (Δεκέμβριος 2017) δεν έχει κυκλοφορήσει.

ΠΑΡΑΡΤΗΜΑ ΙΙΙ- LUA

Η γλώσσα προγραμματισμού Lua

Η Lua είναι μια μικρή γλώσσα προγραμματισμού γραμμένη σε C. Ξεκίνησε ως εσωτερικό έργο το 1993 από τους Roberto Ierusalimschy, Luiz Henrique de Figueiredo και Waldemar Cele.

Το NodeMCU είναι ένα πακέτο υλικολογισμικού βασισμένο σε Lua για το ESP8266, οπότε είναι σημαντικό να γνωρίζετε τα βασικά στοιχεία της Lua για να γράψετε τις δικές σας εφαρμογές για το ESP8266.

Μεταβλητές

Στη Lua, αν και δεν έχουμε τύπους δεδομένων, έχουμε τρεις τύπους βάσει του εύρους της μεταβλητής. Το πεδίο σημαίνει ότι μια μεταβλητή μπορεί να είναι είτε γενικής είτε τοπικής εμβέλειας.

Γενικές μεταβλητές (Global variables): Όλες οι μεταβλητές θεωρούνται γενικές (εκτός εάν δηλώνονται ως τοπικές).

Τοπικές μεταβλητές (Local variables): Όταν ο τύπος ορίζεται ως τοπικός για μια μεταβλητή, το πεδίο εφαρμογής του περιορίζεται από τις λειτουργίες εντός του πεδίου εφαρμογής του

```
local pin = 3  
local test = "It works!"
```

Πεδία πίνακα (Table fields): Αυτός είναι ένας ειδικός τύπος μεταβλητής που μπορεί να περιλάβει οτιδήποτε εκτός από το μηδέν.

Η Lua είναι case-sensitive. Έτσι μια μεταβλητή που ονομάζεται SENSOR είναι διαφορετική από την Sensor ή τη sensor.

Τύποι δεδομένων, γνωστός και ως Τύπος τιμής

Η Lua είναι μια δυναμική γλώσσα, δηλαδή οι μεταβλητές δεν έχουν τύπους. Τύπο έχουν μόνο οι τιμές. Οι τιμές μπορούν να αποθηκευτούν σε μεταβλητές, να περάσουν ως παράμετροι και να επιστραφούν ως αποτελέσματα.

Ο κατάλογος των τύπων δεδομένων για τις τιμές δίνονται παρακάτω.

Τύπος	Περιγραφή
String	Αρχεία χαρακτήρων
αριθμός	Αντιπροσωπεύει πραγματικούς αριθμούς (διπλής ακρίβειας κινητής υποδιαστολής)
boolean	Περιλαμβάνει αληθείς True και ψευδείς False τιμές.
Function	Μια μέθοδος που γράφεται στο Lua
nil	Δεν υπάρχουν δεδομένα αποθηκευμένα στη μεταβλητή
table, userdata και thread	Δεν θα καλύψουμε αυτούς τους 3 τύπους στην παρούσα εργασία

Εδώ είναι μια απεικόνιση των τύπων:

```
print(type("Hello World!")) -- string
print(type(7))              -- number
print(type(true))          -- boolean
print(type(print))         -- function
print(type(nil))           -- nil
```

Όταν εργάζεστε με το NodeMCU στο ESP8266, θα δείτε την τιμή nil να εμφανίζεται μια φορά. Απλώς σημαίνει ότι μια μεταβλητή δεν έχει οριστεί. Επίσης, εάν θέλετε να διαγράψετε μια μεταβλητή, απλά ορίστε αυτήν τη μεταβλητή στην τιμή μηδέν.

Σχόλια

Τα σχόλια είναι απλό κείμενο που εξηγεί πώς λειτουργεί ο κώδικας. Οτιδήποτε χαρακτηρίζεται ως σχόλιο αγνοείται από την ενότητα ESP. Τα σχόλια αρχίζουν με δύο παύλες: -. Υπάρχουν δύο τύποι σχολίων:

Σχόλιο μιας γραμμής

```
print("Hello World!") -- That's how you make a comment
```

```
--[[
print("Hello World!") this is a multi line comment
--]]
```

Σχόλια πολλών γραμμών

A = 1
B = 2

Αριθμητικοί τελεστές

Operator	Example	Result
+	A + B	3
-	A - B	-1
*	A * B	2
/	B / A	2
%	B % A	0
^	B^2	4
-	-A	-1

Συσχετικοί τελεστές

Operator	Example	Result
==	(A == B)	not true
~=	(A ~= B)	true
>	(A > B)	not true
<	(A < B)	true
>=	(A >= B)	not true
<=	(A <= B)	true

Λογικοί τελεστές

Operator	Example	Result
and	(A and B)	false

or	(A or B)	true
not	!(A and B)	true

Τελεστής διαδοχής

Φανταστείτε ότι έχετε 2 τελεστές

```
a = "Hello "
b = "World!"
```

Τελεστές (Operatorς)

Ο τελεστής είναι ένα σύμβολο που λέει στον διερμηνέα να εκτελεί συγκεκριμένους μαθηματικές ή λογικές πράξεις. Η γλώσσα Lua είναι πλούσια σε ενσωματωμένους τελεστές και παρέχει τον ακόλουθο τύπο τελεστών:

Αριθμητικοί τελεστές

Σχεσιακοί τελεστές

Λογικοί τελεστές

Διάφοροι τελεστές

Για όλους τους παρακάτω πίνακες και παραδείγματα σε αυτήν την ενότητα υποθέστε ότι έχετε δύο μεταβλητές: A που αποθηκεύει τον αριθμό 1 και μια μεταβλητή B που αποθηκεύει τον αριθμό 2.

Αριθμητικοί τελεστές

Παράδειγμα αποτελεσμάτων χρήστη

+ A + B 3

- A-B-1

* A * B2

/ B / A 2

% B% A 0

$\wedge B \wedge 2 4$

$--A'1$

Σχεσιακοί τελεστές

Παράδειγμα αποτελεσμάτων χρήστη

$== (A == B)$ δεν είναι αλήθεια

$\sim (A \sim B)$ αληθής

$> (A > B)$ δεν είναι αληθές

$< (A < B)$ αληθές

$>= (A >= B)$ δεν είναι αληθές

$<= (A <= B)$ true

Λογικοί χειριστές

Παράδειγμα αποτελεσμάτων χρήστη

και (A και B) ψευδείς

ή (A ή B) είναι αληθές

όχι! (A και B) αληθινό

Διαχειριστής συνενώσεως

Τώρα φανταστείτε ότι έχετε δύο νέες μεταβλητές:

Operator	Example	Result
..	a..b	"Hello World!"

Βρόχοι

While

For

```
--While Loop
while boolean_value
do
  -- executes this code while is true
end

-- or For Loop
for init,max/min value, increment
do
  -- executes this code while is true
end
```

Ένας βρόχος μας επιτρέπει να εκτελέσουμε ένα μπλοκ κώδικα πολλές φορές για όσο διάστημα η συνθήκη (boolean_value) είναι αληθής.

if... else

αν οι δηλώσεις ... άλλου είναι ένα από τα πιο σημαντικά εργαλεία κωδικοποίησης για την προσθήκη ελέγχου στο πρόγραμμά σας. αν ... άλλες δηλώσεις χρησιμοποιούνται ως εξής:

```
if boolean_value then
  -- if the boolean_value is true
else
  -- if the boolean_value is false
end
```

Η χρήση τους είναι ακριβώς όπως τις περιγράφουν οι λέξεις: Εάν πληρούν μια συγκεκριμένη συνθήκη (boolean_value = true), τότε ο κώδικας μέσα στην εντολή if εκτελείται. Εάν η συνθήκη είναι ψευδής (boolean_value = false), εκτελείται ο κώδικας μέσα στην εντολή else.

Συναρτήσεις (Functions)

Οι συναρτήσεις είναι εξαιρετικός τρόπος οργάνωσης του κώδικα σας. Αν θέλετε να κάνετε κάτι πολλές φορές, αντί να επαναλάβετε τον κωδικό σας αρκετές φορές, δημιουργείτε μια ξεχωριστή συνάρτηση που μπορείτε να καλέσετε και να εκτελέσετε οποτεδήποτε.

Έτσι δημιουργείτε μια νέα συνάρτηση που παίρνει μία παράμετρο (π.χ. τη θερμοκρασία σε Kelvin) και μετατρέπει αυτή τη θερμοκρασία σε Κελσίου και Φαρενάιτ:

```
function displayTemperature(kelvin)
  celsius = kelvin - 273.15
  print("Temperature in Celsius is: ", celsius)

  fahrenheit = (celsius*9/5+32)
  print("Temperature in Fahrenheit is: ", fahrenheit)
end

k = 294 --temperature in Kelvin
displayTemperature(k) -- calls function
```

ΠΑΡΑΡΤΗΜΑ IV- Παράρτημα κεφ.2

Κώδικας που τρέχει στον Server του Κεφαλαίου 2

```
/*****
```

```
ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ  
ANNA ΜΑΜΟΥΖΕΛΟΥ ΑΜ 16088  
Διπλωματική εργασία
```

```
Πρώτα τρέχουμε σε(command mode) το MQTT server (mosquitto -v),στον υπολογιστή μας  
στη συνέχεια (node-red) και το browser (localhost:1880)
```

```
Με τον IP scanner εντοπίζουμε τις IP διευθύνσεις του MQTT broker  
και του ESPserver, καθώς και των υπολοίπων Android συσκευών
```

```
Τοποθετούμε την IP διεύθυνση στον κώδικά μας και κάνουμε διερμηνευση
```

```
*****/
```

```
//
```

```
// Συμπεριλαμβάνουμε τις απαραίτητες βιβλιοθήκες
```

```
//
```

```
#include <ESP8266WiFi.h>
```

```
#include <PubSubClient.h>
```

```
#include "DHT.h"
```

```
//
```

```
// Ενεργοποιούμε τον ανάλογο αισθητήρα θερμοκρασίας-υγρασίας
```

```
#define DHTTYPE DHT11 // DHT 11
```

```
//
```

```
// Σύνδεση του Server ESP8266 στο Router
```

```
//
```

```
//const char* ssid = "COSMOTE-XXXXXX"; //vivliothiki
```

```
//
```

```
//const char* password = "XXXXXXXXXXXXXXXX"; //vivliothiki
```

```
//
```

```
const char* mqtt_server = "192.168.1.4"; //---- Η διεύθυνση IP στην οποία τρέχει ο broker
```

```
//
```

```
int salert_value;;
```

```
int kflood_value;
```

```
static char alert[7];
```

```

    static char flood[7];
//
// Αρχικοποίηση του espClient
//
WiFiClient espClient;
PubSubClient client(espClient);
//
//Πόρτες ελεγχόμενες απο πελάτες-συνδρομητές (subscribing)
//
const int slights = 16; // Φώτα σαλονιού
const int sair = 5; // Κλιματιστικό/Θερμαντικό σώμα
const int salarm = 4; // Θέση συναγερμού στο ενεργό/απενεργοποίηση
const int klights = 0; // Φώτα κουζίνας
const int koven = 2; // Φούρνος κουζίνας ενεργοποίηση/απενεργοποίηση
const int blights = 14; // Φώτα μπαλκονιού
const int tents = 12; // τέντες
const int watering = 13; // Πότισμα
//
long now = millis();
long lastMeasure = 0;
int sensorThres=400;
//
//----- functions prototyping, Εδώ δηλώνονται οι συναρτήσεις
//
void setup_wifi();
void callback(String topic, byte* message, unsigned int length);
void reconnect();
//
//+++++++ MAIN SETUP ++++++
//
void setup() {
    pinMode(slights, OUTPUT);
    pinMode(sair, OUTPUT);
    pinMode(salarm, OUTPUT);
    pinMode(klights, OUTPUT);
    pinMode(koven, OUTPUT);
    pinMode(blights, OUTPUT);

```

```

pinMode(tents, OUTPUT);
pinMode(watering, OUTPUT);
//
digitalWrite(slights, HIGH);
digitalWrite(sair, HIGH);
digitalWrite(salarm, HIGH);
digitalWrite(klights, HIGH);
digitalWrite(koven, HIGH);
digitalWrite(blights, HIGH);
digitalWrite(tents, HIGH);
digitalWrite(watering,HIGH);
//
Serial.begin(115200);
setup_wifi();
client.setServer(mqtt_server, 1883);
client.setCallback(callback);
}
//
//+++++++END SETUP ++++++
//
//+++++ MAIN LOOP ++++++
//
void loop() {

if (!client.connected()) {
    reconnect();
}
if(!client.loop())
    client.connect("ESP8266Client");

    now = millis();
//----- Publishes EVERYTHING every 15 seconds
if (now - lastMeasure > 15000) { // tested also with 30000
    lastMeasure = now;
}
}

```

```

//
// ++++++ FUNCTIONS ++++++
+++++
//
// Σύνδεση του ESP8266 με το Router
//
void setup_wifi() {
  delay(10);
  //----- Πρώτα συνδεόμαστε με το δίκτυο WiFi
  Serial.println();
  Serial.print("Connect to: ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print("!"); // Η τρέχουσα απόπειρα σύνδεσης απέτυχε
  }
  Serial.println("");
  Serial.print("WIFI Connected at ESP Address : ");
  Serial.println(WiFi.localIP());
}
//
//+++++ CALLBACK FUNCTION ++++++
+++++
//
void callback(String topic, byte* message, unsigned int length) {
  Serial.print("Message arrived on topic: ");
  Serial.print(topic);
  Serial.print(". Message: ");

  String messageTemp;

  for (int i = 0; i < length; i++) {
    Serial.print((char)message[i]);
    messageTemp += (char)message[i];
  }
  Serial.println();
}

```

```

//
//+++++++ SALONI-> LIGHTS ON-OFF CONTROL ++++++
//
if(topic=="saloni/lights"){
  Serial.print("Changing Saloni Lights to ");
  if(messageTemp == "ON"){
    digitalWrite(slights, LOW);
    Serial.print("ACTIVE");
  }
  else if(messageTemp == "OFF"){
    digitalWrite(slights, HIGH);
    Serial.print("INACTIVE");
  }
}
//
//+++++++ SALONI-> AIR CONDITION ON-OFF CONTROL ++++++
//
else if(topic=="saloni/air"){
  Serial.print("Changing Saloni air condition device to ");
  if(messageTemp == "ON"){
    digitalWrite(sair, LOW);
    Serial.print("ACTIVE");
  }
  else if(messageTemp == "OFF"){
    digitalWrite(sair, HIGH);
    Serial.print("INACTIVE");
  }
}
//
//+++++++ SALONI-> ALARM ON-OFF CONTROL ++++++
//
else if(topic=="saloni/alarm"){
  Serial.print("Changing Saloni Alarm device to ");
  if(messageTemp == "ON"){
    digitalWrite(salarm, LOW);
    Serial.print("ACTIVE");
  }
}

```

```

    }
    else if(messageTemp == "OFF"){
        digitalWrite(salarm, HIGH);
        Serial.print("INACTIVE");
    }
}
//
// +++++++ KOUZINA-> LIGHTS ON-OFF CONTROL +++++++
//
else if(topic=="kouzina/lights"){
    Serial.print("Changing Kouzina Lights to ");
    if(messageTemp == "ON"){
        digitalWrite(klights, LOW);
        Serial.print("ACTIVE");
    }
    else if(messageTemp == "OFF"){
        digitalWrite(klights, HIGH);
        Serial.print("INACTIVE");
    }
}
//
// +++++++ KOUZINA-> OVEN ON-OFF CONTROL +++++++
//
else if(topic=="kouzina/oven"){
    Serial.print("Changing Kouzina Oven device to ");
    if(messageTemp == "ON"){
        digitalWrite(koven, LOW);
        Serial.print("ACTIVE");
    }
    else if(messageTemp == "OFF"){
        digitalWrite(koven, HIGH);
        Serial.print("INACTIVE");
    }
}
//
// +++++++ BALCONY LIGHTS ON-OFF CONTROL +++++++
//

```

```

else if(topic=="extern/blights"){
  Serial.print("Changing Kouzina Oven device to ");
  if(messageTemp == "ON"){
    digitalWrite(blights, LOW);
    Serial.print("ACTIVE");
  }
  else if(messageTemp == "OFF"){
    digitalWrite(blights, HIGH);
    Serial.print("INACTIVE");
  }
}
//
// ++++++++ TENTS ON-OFF CONTROL ++++++++
//
else if(topic=="extern/tents"){
  Serial.print("Changing Kouzina Oven device to ");
  if(messageTemp == "ON"){
    digitalWrite(tents, LOW);
    Serial.print("ACTIVE");
  }
  else if(messageTemp == "OFF"){
    digitalWrite(tents, HIGH);
    Serial.print("INACTIVE");
  }
}
//
// ++++++++ POTISMA ON-OFF CONTROL ++++++++
//
else if(topic=="extern/watering"){
  Serial.print("Changing Kouzina Oven device to ");
  if(messageTemp == "ON"){
    digitalWrite(watering, LOW);
    Serial.print("ACTIVE");
  }
  else if(messageTemp == "OFF"){
    digitalWrite(watering, HIGH);
    Serial.print("INACTIVE");
  }
}

```



```

    }
  }
  Serial.println();
}
//
//+++++++END DEVICES CONTROL+++++++
+++++++
//
//+++++++ RECONNECT ++++++++
//
// Σύνδεση του ESP8266 με τον MQTT broker
// Change the function below if you want to subscribe to more topics with your ESP8266
//
void reconnect() {
//----- Loop until we're reconnected
  while (!client.connected()) {
    Serial.print("Attempting MQTT connection...");
//----- Attempting to connect

    if (client.connect("ESP8266Client")) {
      Serial.println("connected");
      // Subscribe or resubscribe to a topic
      // You can subscribe to more topics (to control more LEDs in this example)
//+++++++ SUBSCRIPTION ++++++++
      client.subscribe("saloni/lights");
      client.loop();
      client.subscribe("saloni/air");
      client.loop();
      client.subscribe("saloni/alarm");
      client.loop();
      client.subscribe("kouzina/lights");
      client.loop();
      client.subscribe("kouzina/oven");
      client.loop();
      client.subscribe("extern/blights");
      client.loop();
      client.subscribe("extern/tents");

```

```

client.loop();
client.subscribe("extern/watering");

} else {
  Serial.print("failed, rc=");
  Serial.print(client.state());
  Serial.println(" try again in 5 seconds");
  // Wait 5 seconds before retrying
  delay(5000);
}
}
}

```

Παράρτημα κεφ 2

Ο κώδικας του NODE-RED του dashboard, κεφάλαιο 2

```

[{"id":"98e5a171.3add4","type":"mqtt
out","z":"16891922.0aa4b7","name":"saloniLights","topic":"saloni/lights","qos":"0","retain":"","
broker":"15d11240.d1909e","x":533,"y":100,"wires":[]},
{"id":"26178dac.d2abf2","type":"ui_switch","z":"16891922.0aa4b7","name":"saloniLights","label":
"saloniLights","group":"6a8fcf9f.8b781","order":0,"width":0,"height":0,"passthru":true,"decou
ple":"false","topic":"saloni/lights","style":"","onvalue":"ON","onvalueType":"str","onicon":"","o
ncolor":"","offvalue":"OFF","offvalueType":"str","officon":"","offcolor":"","x":254,"y":99,"wires
":["98e5a171.3add4"]},
{"id":"6a94bc3b.488274","type":"ui_switch","z":"16891922.0aa4b7","name":"saloniAir","label":
"saloniAir","group":"6a8fcf9f.8b781","order":0,"width":0,"height":0,"passthru":true,"decouple":
"false","topic":"saloni/air","style":"","onvalue":"ON","onvalueType":"str","onicon":"","oncolor":
","offvalue":"OFF","offvalueType":"str","officon":"","offcolor":"","x":261,"y":152,"wires":
["a45f32de.f354b"]},
{"id":"c53c1afc.8ac3d8","type":"ui_switch","z":"16891922.0aa4b7","name":"saloniAlarm","label":
"saloniAlarm","group":"6a8fcf9f.8b781","order":0,"width":0,"height":0,"passthru":true,"decou
ple":"false","topic":"saloni/alarm","style":"","onvalue":"ON","onvalueType":"str","onicon":"","on
color":"","offvalue":"OFF","offvalueType":"str","officon":"","offcolor":"","x":250,"y":194,"wires
":["8b7ac906.f91e28"]},
{"id":"8990c499.5cf548","type":"ui_switch","z":"16891922.0aa4b7","name":"koyzinaLights","la
bel":"kouzinaLights","group":"6a8fcf9f.8b781","order":0,"width":0,"height":0,"passthru":true,"de

```

```

couple":"false","topic":"kouzina/lights","style":"","onvalue":"ON","onvalueType":"str","onicon":
":"","oncolor":"","offvalue":"OFF","offvalueType":"str","officon":"","offcolor":"","x":243,"y":243,
"wires":[["e035e1ff.c4724"]]},
{"id":"9e2a9838.c5fa78","type":"ui_switch","z":"16891922.0aa4b7","name":"kouzinaOven","lab
el":"kouzinaOven","group":"6a8fcf9f.8b781","order":0,"width":0,"height":0,"passthru":true,"dec
ouple":"false","topic":"kouzina/oven","style":"","onvalue":"ON","onvalueType":"str","onicon":"","
oncolor":"","offvalue":"OFF","offvalueType":"str","officon":"","offcolor":"","x":252,"y":285,"
wires":[["f24c613c.c8133"]]}, {"id":"a45f32de.f354b","type":"mqtt
out","z":"16891922.0aa4b7","name":"saloniAir","topic":"saloni/air","qos":0,"retain":"","broker
":"15d11240.d1909e","x":523,"y":152,"wires":[]}, {"id":"8b7ac906.f91e28","type":"mqtt
out","z":"16891922.0aa4b7","name":"saloniAlarm","topic":"saloni/alarm","qos":0,"retain":"","
broker":"15d11240.d1909e","x":533,"y":194,"wires":[]}, {"id":"e035e1ff.c4724","type":"mqtt
out","z":"16891922.0aa4b7","name":"kouzinaLights","topic":"kouzina/lights","qos":0,"retain":
":"","broker":"15d11240.d1909e","x":542,"y":242,"wires":[]}, {"id":"f24c613c.c8133","type":"mqtt
out","z":"16891922.0aa4b7","name":"kouzinaOven","topic":"kouzina/oven","qos":1,"retain":"","
broker":"15d11240.d1909e","x":532,"y":286,"wires":[]}, {"id":"3e005fcd.d8707","type":"mqtt
out","z":"16891922.0aa4b7","name":"balconyLights","topic":"extern/blights","qos":1,"retain":
":"","broker":"15d11240.d1909e","x":541,"y":337,"wires":[]}, {"id":"ec877cab.39458","type":"mqtt
out","z":"16891922.0aa4b7","name":"balconyTents","topic":"extern/tents","qos":0,"retain":"","
broker":"15d11240.d1909e","x":534,"y":385,"wires":[]}, {"id":"be78ee28.4cf29","type":"mqtt
out","z":"16891922.0aa4b7","name":"gardenWatering","topic":"extern/watering","qos":0,"retai
n":"","broker":"15d11240.d1909e","x":545,"y":429,"wires":[]},
{"id":"b0f64d4c.d1e62","type":"ui_switch","z":"16891922.0aa4b7","name":"balconyLights","lab
el":"balconyLights","group":"6a8fcf9f.8b781","order":0,"width":0,"height":0,"passthru":true,"dec
ouple":"false","topic":"extern/blights","style":"","onvalue":"ON","onvalueType":"str","onicon":"","
oncolor":"","offvalue":"OFF","offvalueType":"str","officon":"","offcolor":"","x":243,"y":336,"
wires":[["3e005fcd.d8707"]]},
{"id":"40966cae.5a1be4","type":"ui_switch","z":"16891922.0aa4b7","name":"balconyTents","lab
el":"balconyTents","group":"6a8fcf9f.8b781","order":0,"width":0,"height":0,"passthru":true,"dec
ouple":"false","topic":"extern/tents","style":"","onvalue":"ON","onvalueType":"str","onicon":"","
oncolor":"","offvalue":"OFF","offvalueType":"str","officon":"","offcolor":"","x":256,"y":384,"wi
res":[["ec877cab.39458"]]},
{"id":"298801d3.ee10ee","type":"ui_switch","z":"16891922.0aa4b7","name":"gardenWatering","
label":"gardenWatering","group":"6a8fcf9f.8b781","order":0,"width":0,"height":0,"passthru":true
,"decouple":"false","topic":"extern/watering","style":"","onvalue":"ON","onvalueType":"str","oni
con":"","oncolor":"","offvalue":"OFF","offvalueType":"str","officon":"","offcolor":"","x":248,"y"
:428,"wires":[["be78ee28.4cf29"]]}, {"id":"15d11240.d1909e","type":"mqtt-

```

```
broker", "z": "", "broker": "localhost", "port": "1883", "clientId": "", "usetls": false, "compatmode": true, "
keepalive": "60", "cleansession": true, "willTopic": "", "willQos": "0", "willPayload": "", "birthTopic": "
", "birthQos": "0", "birthPayload": ""},
{"id": "6a8fc9f.8b781", "type": "ui_group", "z": "", "name": "Subscriptions", "tab": "474279dc.ddb3c8
", "order": 1, "disp": true, "width": "6"},
{"id": "474279dc.ddb3c8", "type": "ui_tab", "z": "", "name": "subscriptions", "icon": "dashboard", "orde
r": 2}]
```

ΠΑΡΑΡΤΗΜΑ V- Παράρτημα κεφαλαίου 3

```
*****/
// Arduino like WeMos D1, ESP8266 board
// Συμπεριλαμβάνουμε τις απαραίτητες βιβλιοθήκες
//
#include <ESP8266WiFi.h>
#include <PubSubClient.h>
#include "DHT.h"
//
// Ενεργοποιούμε τον ανάλογο αισθητήρα θερμοκρασίας-υγρασίας
#define DHTTYPE DHT11 // DHT 11
//
// Σύνδεση του Server ESP8266 στο Router
//
//const char* ssid = "COSMOTE-XXXXXX"; //vivliothiki
//
//const char* password = "XXXXXXXXXX"; //vivliothiki
//
const char* mqtt_server = "192.168.1.4"; //---- Η διεύθυνση IP στην οποία τρέχει ο broker
//
    int PIR_value;
    // static char light[7];
    // static char PIR_value[7];
//
// Αρχικοποίηση του espClient
//
WiFiClient espClientD1;
PubSubClient client(espClientD1);
//
//Πόρτες ελεγχόμενες απο πελάτες-συνδρομητές (subscribing)
//
//Πόρτες αισθητήρων για τιμές δημοσίευσης (publishing)
//
const int DHTPin = D5; // Αισθητήρας Θερμοκρασίας και υγρασίας
```

```

const int GasPin = A0; // Ανιχνευτής διαρροής γκαζιού
const int LightPin = D3; //LDR
const int PIRPin = D4; // Ενεργοποίηση PIR συναγερμού απο εισβολέα
//
int light=0;
//
long now = millis();
long lastMeasure = 0;
int sensorThres=400;
//
//----- functions prototyping, Εδώ δηλώνονται οι συναρτήσεις
//
DHT dht(DHTPin, DHTTYPE); //----- Αρχικοποίηση του DHT sensor.
void setup_wifi();
void callback(String topic, byte* message, unsigned int length);
void reconnect();
//
//+++++++ MAIN SETUP ++++++
//
void setup() {
//
pinMode(GasPin, INPUT);
pinMode(PIRPin, INPUT);
pinMode(LightPin, INPUT);

dht.begin();

Serial.begin(115200);
setup_wifi();
client.setServer(mqtt_server, 1883);
client.setCallback(callback);
}
//
//+++++++END SETUP ++++++

//+++++++ MAIN LOOP ++++++
//

```

```

void loop() {

  if (!client.connected()) {
    reconnect();
  }
  if (!client.loop())
    client.connect("ESP8266Client");

  now = millis();
  //----- Publishes EVERYTHING every 15 seconds
  if (now - lastMeasure > 15000) { // tested also with 30000
    lastMeasure = now;
  }
  //
  //+++++=====SENSORS=====+++++
  //
  float hum = dht.readHumidity(); //Celsius (the default)
  float temp = dht.readTemperature();

  // Check if any reads failed and exit early (to try again).
  if (isnan(hum) || isnan(temp)) {
    Serial.println("Failed to read from DHT sensor!");
    return;
  }
  //
  //+++++ PIR +++++
  //
  float PIR_value=digitalRead(PIRPin); // Reading sensor value from pin 13
  if(PIR_value==HIGH) // Checking if PIR sensor sends a HIGH signal to Arduino
  {
    static char alert[7]="HIGH";
    //
    Serial.println("Intruder In the House!");
    delay(50);
    PIR_value=LOW;
  }

  //+++++===== GAS SENSOR +++++

```

```

//
float GasLeak = analogRead(GasPin);
Serial.print("Pin A0: ");
Serial.println(GasLeak);

int Gas=int(GasLeak); // make int to char
static char gasLeakage[7];
dtostrf(GasLeak, 6, 2, gasLeakage);
//----- Checks if it has reached the threshold value
if (Gas> sensorThres)
{
//digitalWrite(redLed, HIGH);
//digitalWrite(greenLed, LOW);
//tone(salarm, 1000, 200);
Serial.print("GasLeakage Exceeded Threshold: ");
Serial.println(Gas);
}
else
{
// digitalWrite(redLed, LOW);
//digitalWrite(greenLed, HIGH);
Serial.print("GasLeakage Lower Than Threshold: ");
Serial.println(Gas);
// noTone(salarm);
}
delay(10);
//
//+++++++END OF GAS SENSOR ++++++
//
// Computes temperature values in Celsius
float hic = dht.computeHeatIndex(temp, hum, false);
static char temperatureTemp[7];
dtostrf(hic, 6, 2, temperatureTemp);

static char humidityTemp[7];

```



```

    dtostrf(hum, 6, 2, humidityTemp);
//
// ++++++ PUBLISHING VALUES IN char mode ++++++
//
client.publish("home/temp", temperatureTemp);
client.publish("home/humid", humidityTemp);
client.publish("home/gas", gasLeakage);
client.publish("home/light", light);
client.publish("home/pir", pir);

Serial.print(" Humidity: ");
Serial.print(hum);
Serial.print(" Temperature: ");
Serial.print(temp);
Serial.print(" *C ");
Serial.print(" Gasleakeage: ");
Serial.println(gasLeakage);
Serial.print(" Light: ");
Serial.println(light);
}
}
//
// ++++++ FUNCTIONS ++++++
+++++
//
// Σύνδεση του ESP8266 με το Router
//
void setup_wifi() {
    delay(10);
//----- Πρώτα συνδεόμαστε με το δίκτυο WiFi
    Serial.println();
    Serial.print("Connect to: ");
    Serial.println(ssid);
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print("!"); // Η τρέχουσα απόπειρα σύνδεσης απέτυχε

```

```

}
Serial.println("");
Serial.print("WIFI Connected at ESP Address : ");
Serial.println(WiFi.localIP());
}
//
//+++++++ CALLBACK FUNCTION ++++++++
+++++++
//
void callback(String topic, byte* message, unsigned int length) {
Serial.print("Message arrived on topic: ");
Serial.print(topic);
Serial.print(". Message: ");

String messageTemp;

for (int i = 0; i < length; i++) {
Serial.print((char)message[i]);
messageTemp += (char)message[i];
}
Serial.println();
}
//
//+++++++END DEVICES CONTROL+++++++
+++++++
//
//+++++++ RECONNECT ++++++++
//
// Σύνδεση του ESP8266 με τον MQTT broker
// Change the function below if you want to subscribe to more topics with your ESP8266
//
void reconnect() {
//----- Loop until we're reconnected
while (!client.connected()) {
Serial.print("Attempting MQTT connection...");
//----- Attempting to connect

```

```

if (client.connect("ESP8266Client")) {
  Serial.println("connected");}
// Subscribe or resubscribe to a topic
// You can subscribe to more topics (to control more LEDs in this example)
//+++++++ SUBSCRIPTIONS IF EXIST ++++++
else {
  Serial.print("failed, rc=");
  Serial.print(client.state());
  Serial.println(" try again in 5 seconds");
  // Wait 5 seconds before retrying
  delay(5000);
}
}
}

```

Παράρτημα κεφ 3

Κεφαλαιο 3: ο κώδικας ροών NODE-RED

```

[{"id":"ca42e13b.9550a","type":"mqtt
in","z":"16891922.0aa4b7","name":"Temperature","topic":"home/temp","qos":"1","broker":"ecbfd5f.966ae
","x":299,"y":283,"wires":[["d8ec9260.b1e9f","cd68c1b9.4913c"]]},
{"id":"671de03f.ba52b","type":"ui_gauge","z":"16891922.0aa4b7","name":"Humidity","group":"3c78b1cb.
086f6e","order":0,"width":0,"height":0,"gtype":"gage","title":"Υγρασία
%","label":"units","format":"{{value}}","min":0,"max":"100","colors":
["#1e03b1","#0b006a","#ca3838"],"seg1":"","seg2":"","x":526,"y":333,"wires":[]},
{"id":"fa6c0cc6.9f39a","type":"ui_gauge","z":"16891922.0aa4b7","name":"Διαρροή
Γκαζιού","group":"b2f1e432.e53af8","order":0,"width":0,"height":0,"gtype":"gage","title":"Διαρροή
Γκαζιού","label":"units","format":"{{value}}","min":0,"max":"1023","colors":
["#ffff80","#ffff80","#ca3838"],"seg1":"","seg2":"","x":542,"y":401,"wires":[]},
{"id":"8949a772.8ff238","type":"mqtt
in","z":"16891922.0aa4b7","name":"Humidity","topic":"home/hum","qos":"2","broker":"ecbfd5f.966ae","
x":291,"y":347,"wires":[["671de03f.ba52b","bb1a663b.974318"]]},{"id":"7d9034f0.d97e8c","type":"mqtt
in","z":"16891922.0aa4b7","name":"GasLeakage","topic":"home/gas","qos":"1","broker":"ecbfd5f.966ae",
"x":302,"y":416,"wires":[["fa6c0cc6.9f39a","9e2243ac.43bca"]]},{"id":"8373071b.6e3d48","type":"mqtt
in","z":"16891922.0aa4b7","name":"kleftes","topic":"home/kleftes","qos":"1","broker":"ecbfd5f.966ae","x
":283,"y":488,"wires":[["7f0ddef2.b9604"]]},

```

```

{"id":"d8ec9260.b1e9f","type":"ui_gauge","z":"16891922.0aa4b7","name":"Temperature","group":"9bfd8bc.083fc8","order":0,"width":0,"height":0,"gtype":"gage","title":"Θερμοκρασία C","label":"units","format":"{{value}}","min":"0","max":"20","colors":["#b4012e","#e60045","#ca3838"],"seg1":"","seg2":"","x":538,"y":267,"wires":[]},
{"id":"cd68c1b9.4913c","type":"ui_chart","z":"16891922.0aa4b7","name":"Temperature","group":"9bfd8bc.083fc8","order":0,"width":0,"height":0,"label":"Καταγραφή Θερμοκρασίας C","chartType":"line","legend":"false","xformat":"HH:mm:ss","interpolate":"linear","nodata":"","dot":false,"ymin":"10","ymax":"18","removeOlder":1,"removeOlderPoints":"","removeOlderUnit":"3600","cutout":0,"useOneColor":false,"colors":["#cf0322","#aec7e8","#ff7f0e","#2ca02c","#98df8a","#d62728","#ff9896","#9467bd","#c5b0d5"],"useOldStyle":false,"x":537,"y":301,"wires":[[],[]]},
{"id":"bb1a663b.974318","type":"ui_chart","z":"16891922.0aa4b7","name":"Humidity","group":"3c78b1cb.086f6e","order":0,"width":0,"height":0,"label":"Καταγραφή σχ. Υγρασίας %","chartType":"line","legend":"false","xformat":"HH:mm:ss","interpolate":"linear","nodata":"","dot":false,"ymin":"0","ymax":"100","removeOlder":1,"removeOlderPoints":"","removeOlderUnit":"3600","cutout":0,"useOneColor":false,"colors":["#2dce04","#aec7e8","#ff7f0e","#2ca02c","#98df8a","#d62728","#ff9896","#9467bd","#c5b0d5"],"useOldStyle":false,"x":523,"y":366,"wires":[[],[]]},
{"id":"9e2243ac.43bca","type":"ui_chart","z":"16891922.0aa4b7","name":"Διαρροή Γκαζιού","group":"b2f1e432.e53af8","order":0,"width":0,"height":0,"label":"Διαρροή Γκαζιού","chartType":"line","legend":"false","xformat":"HH:mm:ss","interpolate":"linear","nodata":"","dot":false,"ymin":"0","ymax":"1023","removeOlder":1,"removeOlderPoints":"","removeOlderUnit":"3600","cutout":0,"useOneColor":false,"colors":["#1f77b4","#aec7e8","#ff7f0e","#2ca02c","#98df8a","#d62728","#ff9896","#9467bd","#c5b0d5"],"useOldStyle":false,"x":542,"y":435,"wires":[[],[]]},{ "id":"7f0ddef2.b9604","type":"e-mail","z":"16891922.0aa4b7","server":"smtp.gmail.com","port":"465","secure":true,"name":"csakarikas@gmail.com","dname":"e-mailKleftes","x":531.5,"y":486,"wires":[]},{ "id":"ecbfd5f.966ae","type":"mqtt-broker","z":"","broker":"192.168.1.4","port":"1883","clientid":"","usetls":false,"compatmode":true,"keepalive":"60","cleansession":true,"willTopic":"","willQos":"0","willPayload":"","birthTopic":"","birthQos":"0","birthPayload":""},{ "id":"3c78b1cb.086f6e","type":"ui_group","z":"","name":"Anna's Home - Amfilochia","tab":"5b2017c5.25b848","order":6,"disp":true,"width":"6"},
{"id":"b2f1e432.e53af8","type":"ui_group","z":"","name":"Anna's Home - Amfilochia","tab":"5b2017c5.25b848","order":7,"disp":true,"width":"6"},
{"id":"9bfd8bc.083fc8","type":"ui_group","z":"","name":"Anna's Home - Amfilochia","tab":"5b2017c5.25b848","order":5,"disp":true,"width":"6"},
{"id":"5b2017c5.25b848","type":"ui_tab","z":"","name":"external","icon":"dashboard","order":1}

```

ΠΑΡΑΡΤΗΜΑ VI- Παράρτημα κεφαλαίου 4

Κεφάλαιο 4

Ο πλήρης Κώδικας που τρέχει στον εξυπηρετητή πλακέτας Arduino/Ethernet

```
/*
 * ARDUINO CLASSIC SERVER PUB/SUB USING MQTT
 *
ANNA MAMOUCZELOU
UNIVERSITY OF AEGEAN
*/
#include <Ethernet.h> //Ethernet.h library
#include "PubSubClient.h" //PubSubClient.h Library from Knolleary
#include "DHT.h" //DHT.h library
//
#define CLIENT_ID "Ardu"
#define PUBLISH_DELAY 3000 // that is 3 seconds interval
#define DHTTYPE DHT11
// Analog Pins
#define dayLightPin A1 //A1
#define moisturePin A2 //A2
//Digital Pins
#define DHTPIN 2
#define tentUpPin 7
#define tentDownPin 8
#define balconyLightPin 9
#define wateringPin 10
#define moistureLEDPin 13
//
//
String ip = "";
```

```

bool pir = LOW;
bool startsend = HIGH;// flag for sending at startup
int lightstatus; //contains LDR reading
int moisture;
uint8_t mac[6] = {0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F};
//
EthernetClient ethClient;
PubSubClient mqttClient;
DHT dht(DHTPIN, DHTTYPE);
//
void callback(char* topic, byte* payload, unsigned int length);
long previousMillis;
//
void setup() {
  // setup serial communication
  Serial.begin(9600);
  while (!Serial) {};
  Serial.println(F("Arduino with MQTT PUB/SUB"));
  Serial.println();

  //
  pinMode(DHTPIN, INPUT_PULLUP);
  //
  pinMode(moistureLEDPin, OUTPUT);//Ανάβει με την υγρασία
  pinMode(balconyLightPin,OUTPUT);//Ανάβει τα φώτα του μπαλκονιού
  pinMode(tentUpPin , OUTPUT);//Ανεβάζει τέντες τη νύχτα
  pinMode(tentDownPin, OUTPUT);//Κατεβάζει τεντες τη μέρα
  pinMode(wateringPin, OUTPUT);//Πότισμα φυτών του κήπου
  //Set Relays to OFF at Starting
  digitalWrite(balconyLightPin,HIGH);//Relays ACTIVE LOW!
  digitalWrite(tentUpPin, HIGH);
  digitalWrite(tentDownPin, HIGH);
  digitalWrite(wateringPin, HIGH);
  //
  // setup ethernet communication using DHCP
  if (Ethernet.begin(mac) == 0) {
    Serial.println(F("Unable to configure Ethernet using DHCP"));
  }
}

```

```

    for (;;)
    }
//
    Serial.println(F("Ethernet configured via DHCP"));
    Serial.print("IP address: ");
    Serial.println(Ethernet.localIP());
    Serial.println();
//convert ip Array into String
    ip = String (Ethernet.localIP()[0]);
    ip = ip + ".";
    ip = ip + String (Ethernet.localIP()[1]);
    ip = ip + ".";
    ip = ip + String (Ethernet.localIP()[2]);
    ip = ip + ".";
    ip = ip + String (Ethernet.localIP()[3]);
//
    Serial.println(ip);
//
// setup mqtt client
//
    mqttClient.setClient(ethClient);
    // mqttClient.setServer("test.mosquitto.org", 1883); //If use public broker
// Must Be IP address, where BROKER RUNS !!!
    mqttClient.setServer("192.168.1.3", 1883); //If use local broker
//
    Serial.println(F("MQTT client configured"));
    mqttClient.setCallback(callback);
// setup DHT sensor
    dht.begin();
    Serial.println(F("DHT sensor initialized"));
    Serial.println();
    Serial.println(F("Ready to send data"));
    previousMillis = millis();
    mqttClient.publish("spitiEx/ip", ip.c_str());
}
//
//+++++ END SETUP +++++

```

```

//
void loop() {

  lightstatus = analogRead(A0);//Reads an LDR
  pir = digitalRead(7);//Reads a PIR sensor
  // it's time to send new data?
  if (millis() - previousMillis > PUBLISH_DELAY) {
    sendData();

    previousMillis = millis();
  }
  mqttClient.loop();

  delay(5000);
}
//
// ++++++ END LOOP ++++++
//

void sendData() {
//
  char msgBuffer[20];
//
  //Reading Sensor Values
//
  float hum = dht.readHumidity();
  float temp = dht.readTemperature();
  float dayL=analogRead(dayLightPin);
  float moist = analogRead(moisturePin);
//
  //Printing Sensor Values
//
  Serial.print("Temperature: ");
  Serial.print(temp);
  Serial.println(" C");
  Serial.print("Humidity: ");
  Serial.print(hum);

```



```

Serial.println(" %");
Serial.print("DayLight: ");
Serial.println(dayL);
Serial.print("Moisture: ");
Serial.println(moist);
//
//Publishing Sensor Values
//
Serial.println("=====Publishing=====");
//
if (mqttClient.connect(CLIENT_ID)) {
  mqttClient.publish("spitiEx/temp", dtostrf(temp, 6, 2, msgBuffer));
  Serial.println("Temperature.. Published!");
  mqttClient.publish("spitiEx/hum", dtostrf(hum, 6, 2, msgBuffer));
  Serial.println("Humidity.. Published!");
  mqttClient.publish("spitiEx/dayL", dtostrf(dayL, 6, 2, msgBuffer));
  Serial.println("DayLight.. Published!");
  mqttClient.publish("spitiEx/moist", dtostrf(moist,6, 2, msgBuffer));
  Serial.println("Moisture.. Published! ");
//
//Subscribing to MQTT Topics
Serial.println("+++++++");
//
Serial.println("Subscribing to Topics");
delay(100);
//
mqttClient.subscribe("spitiEx/balconyL");
mqttClient.subscribe("spitiEx/tentUp");
mqttClient.subscribe("spitiEx/tentDn");
mqttClient.subscribe("spitiEx/watering");

if (startsend) {
  mqttClient.publish("spitiEx/ip", ip.c_str());
  startsend = LOW;
}
}
}

```

```

//
//===== Getting Subscription Commands =====
//
void callback(char* topic, byte* payload, unsigned int length) {
  char msgBuffer[20];
  //
  Serial.print("Message arrived [");
  //
  String strPayload = String((char*)payload); // convert to string
  Serial.println(strPayload);
  Serial.print("Message arrived [");
  Serial.print(topic);
  Serial.print("] "); //MQTT_BROKER
  for (int i = 0; i < length; i++) {
    Serial.print((char)payload[i]);
  }
  Serial.println();
  Serial.println(payload[0]);
  //
  // Examine only the first character of the message
  //
  // Lights On Balcony
  //
  if (topic=="spitiEx/balconyL"){
    if (payload[0] == 49){          //Message "1" in ASCII (turn output ON)
      digitalWrite(balconyLightPin,LOW); //Lights ON
      Serial.println("Balcony Lights ON!");
    }
    else if (payload[0] == 48){    //Message "0" in ASCII (turn output OFF)
      digitalWrite(balconyLightPin,HIGH);
      Serial.println("Balcony Lights OFF!");
    }
    else if (payload[0] == 50){
      mqttClient.publish("spitiEx/ip", ip.c_str());// publish IP nr
    }
    else {
      Serial.println("Unknown value");
    }
  }
}

```

```

    // mqttClient.publish("spitiEx/mess", "Syntax Error");
  }
}

// Tent Up
if (topic=="spitiEx/tentUp"){
  if (payload[0] == 49){          //Message "1" in ASCII (turn output ON)
    digitalWrite(tentUpPin,LOW);    //Tents Up ON
    Serial.println("Tents Up ON!");
  }
  else if (payload[0] == 48){      //Message "0" in ASCII (turn output OFF)
    digitalWrite(tentUpPin,HIGH);    //Tents Up OFF
    Serial.println("Tents Up OFF!");
  }
}

// Tent Down
if (topic=="spitiEx/tentDn"){
  if (payload[0] == 49){          //Message "1" in ASCII (turn output ON)
    digitalWrite(tentDownPin,LOW);    //Tents Down ON
    Serial.println("Tents Down ON!");
  }
  else if (payload[0] == 48){      //Message "0" in ASCII (turn output OFF)
    digitalWrite(tentDownPin,HIGH);    //Tents Down OFF
    Serial.println("Tents Down OFF!");
  }
}

//Watering
if (topic=="spitiEx/watering"){
  if (payload[0] == 49){          //Message "1" in ASCII (turn output ON)
    digitalWrite(wateringPin,LOW);    //Potisma ON
    Serial.println("Potisma ON!");
  }
  else if (payload[0] == 48){      // Message "0" in ASCII (turn output OFF)
    digitalWrite(wateringPin,HIGH);    //Potisma OFF
    Serial.println("Potisma OFF!");
  }
}
}
}

```

```
// }
//+++++ END OF PROGRAM +++++
++
```

Παράρτημα: κεφάλαιο 4

Το αρχείο δημιουργίας του dashboard σε Json

```
[{"id":"8abc6cac.c9de","type":"mqtt
out","z":"16891922.0aa4b7","name":"BalconyL","topic":"spitiEx/balconyL","qos":"1","retain":"","
broker":"b4558ce1.b9fae","x":539.5,"y":90,"wires":[]},{ "id":"2500ae5a.0a6342","type":"mqtt
out","z":"16891922.0aa4b7","name":"TentUp","topic":"spitiEx/tentUp","qos":"1","retain":"","bro
ker":"b4558ce1.b9fae","x":537.5,"y":139,"wires":[]},{ "id":"b23b71ac.39965","type":"mqtt
out","z":"16891922.0aa4b7","name":"TentDown","topic":"spitiEx/tentDn","qos":"1","retain":"","
broker":"b4558ce1.b9fae","x":541.5,"y":189,"wires":[]},{ "id":"8095f60c.4f1458","type":"mqtt
out","z":"16891922.0aa4b7","name":"Potisma","topic":"spitiEx/watering","qos":"1","retain":"","
broker":"b4558ce1.b9fae","x":542.5,"y":235,"wires":[]},
{"id":"d3d35688.36b148","type":"ui_switch","z":"16891922.0aa4b7","name":"balconyL","label":
"switch","group":"2f3ceba.e732214","order":0,"width":0,"height":0,"passthru":true,"decouple":"f
alse","topic":"spitiEx/balconyL","style":"","onvalue":"ON","onvalueType":"str","onicon":"","onco
lor":"","offvalue":"OFF","offvalueType":"str","officon":"","offcolor":"","x":295.5,"y":89,"wires
":["8abc6cac.c9de"]},
{"id":"9eed39bd.ac7f68","type":"ui_switch","z":"16891922.0aa4b7","name":"Tent
UP","label":"switch","group":"2dba42b0.0ee88e","order":0,"width":0,"height":0,"passthru":true,"
decouple":"false","topic":"spitiEx/tentUp","style":"","onvalue":"ON","onvalueType":"str","onico
n":"","oncolor":"","offvalue":"OFF","offvalueType":"str","officon":"","offcolor":"","x":299.5,"y"
:139,"wires":["2500ae5a.0a6342"]},
{"id":"87c8357d.e79498","type":"ui_switch","z":"16891922.0aa4b7","name":"Tent
Down","label":"switch","group":"bf958bfc.fbce8","order":0,"width":0,"height":0,"passthru":true
,"decouple":"false","topic":"spitiEx/tentDn","style":"","onvalue":"ON","onvalueType":"str","onico
n":"","oncolor":"","offvalue":"OFF","offvalueType":"str","officon":"","offcolor":"","x":295.5,"y"
:189,"wires":["b23b71ac.39965"]},
{"id":"8f80289a.105818","type":"ui_switch","z":"16891922.0aa4b7","name":"Potisma","label":"s
witch","group":"bcaa2e58.085e4","order":0,"width":0,"height":0,"passthru":true,"decouple":"fals
e","topic":"spitiEx/watering","style":"","onvalue":"ON","onvalueType":"str","onicon":"","oncolo
r":"","offvalue":"OFF","offvalueType":"str","officon":"","offcolor":"","x":305.5,"y":234,"wires":
["8095f60c.4f1458"]},{ "id":"556686dc.431568","type":"mqtt
```

```

in","z":"16891922.0aa4b7","name":"Temperature","topic":"spitiEx/temp","qos":"1","broker":"b4
558ce1.b9fae","x":313.5,"y":319,"wires":[["653cae00.c3a0e","6f9d0a8e.075984"]]},
{"id":"f03970e5.a0d01","type":"ui_gauge","z":"16891922.0aa4b7","name":"Humidity","group":"
67d3dc05.b64234","order":0,"width":0,"height":0,"gtype":"gage","title":"Humidity
%","label":"units","format":"{{value}}","min":0,"max":"100","colors":
["#00b500","#04e10a","#ca3838"],"seg1":"","seg2":"","x":540.5,"y":369,"wires":[]},
{"id":"b6f88337.f9cc2","type":"ui_gauge","z":"16891922.0aa4b7","name":"DayLight","group":"f
25305cb.1c5ac8","order":0,"width":0,"height":0,"gtype":"gage","title":"DayLight
","label":"units","format":"{{value}}","min":0,"max":"1023","colors":
["#00b500","#392bbb","#ca3838"],"seg1":"","seg2":"","x":536.5,"y":437,"wires":[]},
{"id":"c8cf58c6.9b6928","type":"ui_gauge","z":"16891922.0aa4b7","name":"Moisture","group":
"a75a351.7198ec8","order":0,"width":0,"height":0,"gtype":"gage","title":"Soil Moisture
","label":"units","format":"{{value}}","min":0,"max":"1023","colors":
["#00b500","#5b648a","#cebb33"],"seg1":"","seg2":"","x":534.5,"y":507,"wires":[]},
{"id":"6489dbdc.add6e4","type":"mqtt
in","z":"16891922.0aa4b7","name":"","topic":"spitiEx/hum","qos":"2","broker":"b4558ce1.b9fae
","x":315.5,"y":383,"wires":[["f03970e5.a0d01","846c54da.415f08"]]},
{"id":"a1127905.7ce048","type":"mqtt
in","z":"16891922.0aa4b7","name":"","topic":"spitiEx/dayL","qos":"2","broker":"b4558ce1.b9fae
","x":316.5,"y":452,"wires":[["b6f88337.f9cc2","abb737c8.6cadb8"]]},
{"id":"fc5bad98.43d97","type":"mqtt
in","z":"16891922.0aa4b7","name":"","topic":"spitiEx/moist","qos":"2","broker":"b4558ce1.b9fa
e","x":317.5,"y":524,"wires":[["c8cf58c6.9b6928","481cd4c3.a6f8ec"]]},
{"id":"653cae00.c3a0e","type":"ui_gauge","z":"16891922.0aa4b7","name":"Temperature","group
":"c9525699.a2e268","order":0,"width":0,"height":0,"gtype":"gage","title":"Temp in
C","label":"units","format":"{{value}}","min":0,"max":"20","colors":
["#00b500","#d35112","#ca3838"],"seg1":"","seg2":"","x":552.5,"y":303,"wires":[]},
{"id":"6f9d0a8e.075984","type":"ui_chart","z":"16891922.0aa4b7","name":"Temperature","grou
p":"c9525699.a2e268","order":0,"width":0,"height":0,"label":"Temperature in
C","chartType":"line","legend":"false","xformat":"HH:mm:ss","interpolate":"linear","nodata":"","
dot":false,"ymin":"10","ymax":"18","removeOlder":1,"removeOlderPoints":"","removeOlderUnit
":"3600","cutout":0,"useOneColor":false,"colors":
["#bd2f15","#aec7e8","#ff7f0e","#2ca02c","#98df8a","#d62728","#ff9896","#9467bd","#c5b0d5
"],"useOldStyle":false,"x":551.5,"y":337,"wires":[[],[]]},
{"id":"846c54da.415f08","type":"ui_chart","z":"16891922.0aa4b7","name":"Humidity","group":"
67d3dc05.b64234","order":0,"width":0,"height":0,"label":"Humidity
%","chartType":"line","legend":"false","xformat":"HH:mm:ss","interpolate":"linear","nodata":"","

```

```

"dot":false,"ymin":0,"ymax":100,"removeOlder":1,"removeOlderPoints":"","removeOlderUnit":3600,"cutout":0,"useOneColor":false,"colors":
["#2dce04","#aec7e8","#ff7f0e","#2ca02c","#98df8a","#d62728","#ff9896","#9467bd","#c5b0d5
"],"useOldStyle":false,"x":537.5,"y":402,"wires":[[],[]]},
{"id":"abb737c8.6cadb8","type":"ui_chart","z":16891922.0aa4b7,"name":"DayLight","group":
f25305cb.1c5ac8,"order":0,"width":0,"height":0,"label":"DayLight","chartType":"line","legend":
"false","xformat":"HH:mm:ss","interpolate":"linear","nodata":"","dot":false,"ymin":0,"ymax":
1023,"removeOlder":1,"removeOlderPoints":"","removeOlderUnit":3600,"cutout":0,"useOneC
olor":false,"colors":
["#1f77b4","#aec7e8","#ff7f0e","#2ca02c","#98df8a","#d62728","#ff9896","#9467bd","#c5b0d5
"],"useOldStyle":false,"x":536.5,"y":471,"wires":[[],[]]},
{"id":"481cd4c3.a6f8ec","type":"ui_chart","z":16891922.0aa4b7,"name":"SoilMoisture","grou
p":"a75a351.7198ec8","order":0,"width":0,"height":0,"label":"Soil
Moisture","chartType":"line","legend":"false","xformat":"HH:mm:ss","interpolate":"linear","nod
ata":"","dot":false,"ymin":0,"ymax":1023,"removeOlder":1,"removeOlderPoints":"","remove
OlderUnit":3600,"cutout":0,"useOneColor":false,"colors":
["#b0b81b","#aec7e8","#ff7f0e","#2ca02c","#98df8a","#d62728","#ff9896","#9467bd","#c5b0d5
"],"useOldStyle":false,"x":545.5,"y":545,"wires":[[],[]]},{
"mqtt-broker","z":"","broker":"192.168.1.2","port":"1883","clientid":"","usetls":false,"compatmode":tru
e,"keepalive":"60","cleansession":true,"willTopic":"","willQos":"0","willPayload":"","birthTopic
":"","birthQos":"0","birthPayload":""},
{"id":"2f3ceba.e732214","type":"ui_group","z":"","name":"balcony","tab":"11556ba.5bcb394","o
rder":1,"disp":true,"width":"6"},
{"id":"2dba42b0.0ee88e","type":"ui_group","z":"","name":"tentsUp","tab":"11556ba.5bcb394","
order":2,"disp":true,"width":"6"},
{"id":"bf958bfc.fbece8","type":"ui_group","z":"","name":"tentsDown","tab":"11556ba.5bcb394",
"order":3,"disp":true,"width":"6"},
{"id":"bcaa2e58.085e4","type":"ui_group","z":"","name":"watering","tab":"11556ba.5bcb394","o
rder":4,"disp":true,"width":"6"},
{"id":"67d3dc05.b64234","type":"ui_group","z":"","name":"Humidity","tab":"11556ba.5bcb394"
,"order":6,"disp":true,"width":"6"},
{"id":"f25305cb.1c5ac8","type":"ui_group","z":"","name":"DayLight","tab":"11556ba.5bcb394",
"order":7,"disp":true,"width":"6"},
{"id":"a75a351.7198ec8","type":"ui_group","z":"","name":"Moisture","tab":"11556ba.5bcb394",
"order":8,"disp":true,"width":"6"},
{"id":"c9525699.a2e268","type":"ui_group","z":"","name":"temp","tab":"11556ba.5bcb394","ord
er":5,"disp":true,"width":"6"},

```

```
{"id":"11556ba.5bcb394","type":"ui_tab","z":"","name":"external","icon":"dashboard","order":1}  
]
```

ΠΑΡΑΡΤΗΜΑ VII Παράρτημα κεφαλαίου 5

Κεφάλαιο 5

Ο κώδικας του Server ESP8266

/*****

ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ
ANNA ΜΑΜΟΥΖΕΛΟΥ AM 16088

Διπλωματική εργασία

ΚΕΦΑΛΑΙΟ 5

Πρώτα τρέχουμε σε(command mode) το MQTT server (mosquitto -v),στον υπολογιστή μας στη συνέχεια (node-red) και το browser (localhost:1880)

Με τον IP scanner εντοπίζουμε τις IP διευθύνσεις του MQTT broker

και του ESPserver, καθώς και των υπολοίπων Android συσκευών

Τοποθετούμε την IP διεύθυνση στον κώδικά μας και κάνουμε διερμηνευση

*****/

//

// Συμπεριλαμβάνουμε τις απαραίτητες βιβλιοθήκες

//

#include <ESP8266WiFi.h>

#include <PubSubClient.h>

#include "DHT.h"

//

// Ορίζουμε τον τύπο του αισθητήρα θερμοκρασίας-υγρασίας

#define DHTTYPE DHT11 // DHT 11

//

// Σύνδεση του Server ESP8266 στο Router

//

//const char* ssid = "XXXXXXXXXXXXX"; //vivliothiki

//

//const char* password = "XXXXXXXXXXXXX"; //vivliothiki

//

const char* mqtt_server = "192.168.1.4"; //---- Η διεύθυνση IP στην οποία τρέχει ο broker

//


```

int salert_value;
int kflood_value;
static char alert[7];
static char flood[7];
//
// Αρχικοποίηση του espClient
//
WiFiClient espClient;
PubSubClient client(espClient);
//
//Πόρτες ελεγχόμενες απο πελάτες-συνδρομητές (subscribing)
//
const int slights = 16; // Φώτα σαλονιού
const int sair = 5; // Κλιματιστικό/Θερμαντικό σώμα
const int salarm = 4; // Θέση συναγερμού στο ενεργό/απενεργοποίηση
const int klights = 0; // Φώτα κουζίνας
const int koven = 2; // Φούρνος κουζίνας ενεργοποίηση/απενεργοποίηση
const int blights = 5; // Φώτα μπαλκονιού 9
const int watering =5; // Ηλεκτροβάνα για Πότισμα10
//
//Πόρτες αισθητήρων για τιμές δημοσίευσης (publishing)
//
const int DHTPin = 14; // Αισθητήρας Θερμοκρασίας και υγρασίας
const int GasLeakA0 = A0; // Ανιχνευτής διαρροής γκαζιού
const int salert = 13; // Ενεργοποίηση συναγερμού απο εισβολέα
const int kflood = 15; // Διαρροή νερού, πλημμύρα δαπέδου
//
long now = millis();
long lastMeasure = 0;
int sensorThres=400;
//
//----- functions prototyping, Εδώ δηλώνονται οι συναρτήσεις
//
DHT dht(DHTPin, DHTTYPE); //----- Αρχικοποίηση του DHT sensor.
void setup_wifi();
void callback(String topic, byte* message, unsigned int length);
void reconnect();

```

```

//
//+++++++ MAIN SETUP ++++++
//
void setup() {
  pinMode(slights, OUTPUT);
  pinMode(sair, OUTPUT);
  pinMode(salarm, OUTPUT);
  pinMode(klights, OUTPUT);
  pinMode(koven, OUTPUT);
  pinMode(blights, OUTPUT);
  pinMode(watering, OUTPUT);
  //+++++++
  pinMode(DHTPin, INPUT);
  pinMode(GasLeakA0,INPUT);
  pinMode(salert, INPUT);
  pinMode(kflood, INPUT);
  // Αρχικοποίηση των ρελέ σε θέση OFF
  digitalWrite(slights, HIGH);
  digitalWrite(sair, HIGH);
  digitalWrite(salarm, HIGH);
  digitalWrite(klights, HIGH);
  digitalWrite(koven, HIGH);
  digitalWrite(blights, HIGH);
  digitalWrite(watering,HIGH);
  //
  dht.begin();
  Serial.begin(115200);
  setup_wifi();
  client.setServer(mqtt_server, 1883);
  client.setCallback(callback);
}
//
//+++++++END SETUP ++++++
//
//+++++++ MAIN LOOP ++++++
//
void loop() {

```

```

if(!client.connected()) {
    reconnect();
}
if(!client.loop())
    client.connect("ESP8266Client");

    now = millis();
//----- Publishes EVERYTHING every 10 seconds
    if (now - lastMeasure > 10000) { // δοκιμάστηκε και με 30000
        lastMeasure = now;
//
//+++++++=====SENSORS=====+++++++
//
//+++++++ DHT11 Sensor ++++++
//
// Διαβάζει Σχετική Υγρασία %
float hum = dht.readHumidity();
// Διαβάζει Θερμοκρασία σε βαθμούς Κελσίου
float temp = dht.readTemperature();
// Αποτυχία
if (isnan(hum) || isnan(temp)) {
    Serial.println("Failed to read from DHT sensor!");
    }
//
//+++++++ PIR ++++++
//
salert_value=digitalRead(salert); // Έλεγχος του αισθητήρα PIR
float XPIR=100.0;;
if (salert_value==HIGH){
    float XPIR=100.0;
    Serial.println("Intrudor In the House!");
    }
    static char PIR[7];
    dtostrf(XPIR, 6, 2,PIR);
//
//+++++++ FLOODING SENSOR ++++++

```

```

//
kfloat_value=digitalRead(kfloat); // Διαβάζει πλημμύρα
float xfloat;
if(kfloat_value==HIGH){ // Έλεγχος πλημμύρας
  xfloat=100.0;
  Serial.println("Floor Flooding In the House!");
}
else{
  float xfloat=0.0;
}
static char flood[7];
dtostrf(xfloat, 6, 2, flood);
//
//+++++=====+++++ GAS SENSOR ++++++
//
float gasLeak = analogRead(GasLeakA0);
//
//----Έλεγχος αν η διαρροή υπερβαίνει το όριο
if (gasLeak> sensorThres){
  Serial.print("GasLeakage Exceeded Threshold: ");
  Serial.println(gasLeak);
}
else{
  Serial.print("GasLeakage Lower Than Threshold: ");
  Serial.println(gasLeak);
}
static char gasLeakage[7];
dtostrf(gasLeak, 6, 2, gasLeakage);
//
//+++++END OF GAS SENSOR ++++++
//
// Διάβασμα Θερμοκρασίας-Υγρασίας
float hic = dht.computeHeatIndex(temp, hum, false);
static char temperatureTemp[7];
dtostrf(hic, 6, 2, temperatureTemp);
//
static char humidityTemp[7];

```

```

    dtostrf(hum, 6, 2, humidityTemp);
//
// ++++++ Δημοσίευση σε char mode ++++++
//
    client.publish("house/temp", temperatureTemp);
    client.publish("house/humid", humidityTemp);
    client.publish("house/PIR", PIR);
    client.publish("house/flood", flood);
    client.publish("house/gas", gasLeakage);

    Serial.print("Humidity: ");
    Serial.print(hum);
    Serial.print(" Temperature: ");
    Serial.println(temp);
    Serial.print(" *C ");
    Serial.print("Alert: ");
    Serial.println(alert);
    Serial.print("Flood: ");
    Serial.println(flood);
    Serial.print(" Gasleakeage: ");
    Serial.println(gasLeakage);
}
}
//
// ++++++ FUNCTIONS ++++++
+++++
//
// Σύνδεση του ESP8266 με το Router
//
void setup_wifi() {
    delay(10);
//----- Πρώτα συνδεόμαστε με το δίκτυο WiFi
    Serial.println();
    Serial.print("Connect to: ");
    Serial.println(ssid);
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {

```

```

    delay(500);
    Serial.print("!"); // Η τρέχουσα απόπειρα σύνδεσης απέτυχε
}
Serial.println("");
Serial.print("WIFI Got ESP Address : ");
Serial.println(WiFi.localIP());
}
//
//+++++++ CALLBACK FUNCTION ++++++++
++++
//
void callback(String topic, byte* message, unsigned int length) {
    Serial.print("Message arrived on topic: ");
    Serial.print(topic);
    Serial.print(". Message: ");

    String messageTemp;

    for (int i = 0; i < length; i++) {
        Serial.print((char)message[i]);
        messageTemp += (char)message[i];
    }
    Serial.println();
//
//+++++++ ΦΩΤΑ ΣΑΛΟΝΙΟΥ ON-OFF CONTROL ++++++++
//
if(topic=="home/slights"){
    Serial.print("Changing Saloni Lights to ");
    if(messageTemp == "ON"){
        digitalWrite(slights, LOW); // Τα ρελλά είναι ACTIVE LOW!
        Serial.print("ACTIVE");
    }
    else if(messageTemp == "OFF"){
        digitalWrite(slights, HIGH);
        Serial.print("INACTIVE");
    }
}
}
}

```

```

//
// +++++++ ΦΩΤΑ ΚΟΥΖΙΝΑΣ ON-OFF CONTROL +++++++
//
else if(topic=="home/klaights"){
  Serial.print("Changing KITCHEN lights device to ");
  if(messageTemp == "ON"){
    digitalWrite(sair, LOW);
    Serial.print("ACTIVE");
  }
  else if(messageTemp == "OFF"){
    digitalWrite(sair, HIGH);
    Serial.print("INACTIVE");
  }
}

//
// +++++++ AIR-CONDITION ON-OFF CONTROL +++++++
//
else if(topic=="home/air"){
  Serial.print("Changing AIR CONDITION device to ");
  if(messageTemp == "ON"){
    digitalWrite(salarm, LOW);
    Serial.print("ACTIVE");
  }
  else if(messageTemp == "OFF"){
    digitalWrite(salarm, HIGH);
    Serial.print("INACTIVE");
  }
}

//
// +++++++ ΕΝΕΡΓΟΠΟΙΗΣΗ ΣΥΝΑΓΕΡΜΟΥ ON/OFF CONTROL +++++++
//
else if(topic=="home/alarmsSet"){
  Serial.print("Changing BunglarAlarm to ");
  if(messageTemp == "ON"){
    digitalWrite(klights, LOW);
    Serial.print("ACTIVE");
  }
}

```

```

    }
    else if(messageTemp == "OFF"){
        digitalWrite(klights, HIGH);
        Serial.print("INACTIVE");
    }
}
//
// +++++++ ΦΟΥΡΝΟΣ ON-OFF CONTROL +++++++
//
else if(topic=="home/oven"){
    Serial.print("Changing Kouzina Oven device to ");
    if(messageTemp == "ON"){
        digitalWrite(koven, LOW);
        Serial.print("ACTIVE");
    }
    else if(messageTemp == "OFF"){
        digitalWrite(koven, HIGH);
        Serial.print("INACTIVE");
    }
}
//
// +++++++ ΦΩΤΑ ΜΠΑΛΚΟΝΙΟΥ ON-OFF CONTROL +++++++
//
else if(topic=="home/balconyL"){
    Serial.print("Changing Balcony Lights to ");
    if(messageTemp == "ON"){
        digitalWrite(blights, LOW);
        Serial.print("ACTIVE");
    }
    else if(messageTemp == "OFF"){
        digitalWrite(blights, HIGH);
        Serial.print("INACTIVE");
    }
}
//
// +++++++ ΠΟΤΙΣΜΑ ON-OFF CONTROL +++++++
//

```



```

else if(topic=="home/watering"){
  Serial.print("Changing Watering device to ");
  if(messageTemp == "ON"){
    digitalWrite(watering, LOW);
    Serial.print("ACTIVE");
  }
  else if(messageTemp == "OFF"){
    digitalWrite(watering, HIGH);
    Serial.print("INACTIVE");
  }
}
//
Serial.println();
}
//
//++++++END DEVICES CONTROL++++++
//
//++++++ RECONNECT ++++++
//
// Σύνδεση του ESP8266 με τον MQTT broker και εγγραφή στα θέματα
//
void reconnect() {
//----- Επανάληψη μέχρι να συνδεθείς
while (!client.connected()) {
  Serial.print("Attempting MQTT connection...");
//
  if (client.connect("ESP8266Client")) {
    Serial.println("Connected");
//
//++++++ SUBSCRIPTION ++++++
//
    client.subscribe("home/slights");
    client.subscribe("home/klights");
    client.subscribe("home/air");
    client.subscribe("home/alarmSet");
    client.subscribe("home/oven");
    client.subscribe("home/balconyL");

```

```

    client.subscribe("home/watering");
//
    } else {
        Serial.print("failed, rc=");
        Serial.print(client.state());
        Serial.println(" try again in 5 seconds");
        // Wait 5 seconds before retrying
        delay(5000);
    }
}
}
}

```

Παράρτημα κεφάλαιο 5

Κεφάλαιο 5

Ο κώδικας NODE-RED

```

[{"id":"2d036e57.ea8de2","type":"ui_switch","z":"16891922.0aa4b7","name":"AirCondition","label":"AirCondition","group":"feac0250.84ac7","order":0,"width":0,"height":0,"passthru":true,"decouple":"false","topic":"house/air","style":"","onvalue":"ON","onvalueType":"str","onicon":"","oncolor":"","offvalue":"OFF","offvalueType":"str","officon":"","offcolor":"","x":279.99993896484375,"y":85,"wires":[[{"d5479e07.73ffb"}]}, {"id":"ada981e9.1aa64","type":"ui_switch","z":"16891922.0aa4b7","name":"HouseAlarm","label":"Συναγερμός ON/OFF","group":"17b84430.85a71c","order":0,"width":0,"height":0,"passthru":true,"decouple":"false","topic":"house/alarmSet","style":"","onvalue":"ON","onvalueType":"str","onicon":"","oncolor":"","offvalue":"OFF","offvalueType":"str","officon":"","offcolor":"","x":287.99993896484375,"y":133,"wires":[[{"f39e0b97.9b49e8"}]}, {"id":"d5479e07.73ffb","type":"mqtt out","z":"16891922.0aa4b7","name":"SaloniAirCondition","topic":"house/air","qos":"2","retain":"","broker":"616ef29f.e4189c","x":561.9999389648438,"y":88,"wires":[]}, {"id":"f39e0b97.9b49e8","type":"mqtt out","z":"16891922.0aa4b7","name":"Alarm","topic":"house/alarmSet","qos":"2","retain":"","broker":"616ef29f.e4189c","x":525.9999389648438,"y":136,"wires":[]}, {"id":"8f4378d8.7f44a8","type":"mqtt

```

```

in", "z": "16891922.0aa4b7", "name": "SaloniTemperature", "topic": "house/temp", "qos": "1", "broker":
": "616ef29f.e4189c", "x": 272.99993896484375, "y": 373, "wires": [ ["51e31350.1def1c"] ] },
{ "id": "51e31350.1def1c", "type": "ui_gauge", "z": "16891922.0aa4b7", "name": "SaloniTemperature",
, "group": "71c2ac00.1e3fa4", "order": 0, "width": "5", "height": "5", "gtype": "gage", "title": "Θερμοκρα
σία C", "label": "units", "format": " {{value}} ", "min": "8", "max": "28", "colors":
["#00b500", "#e6e600", "#ca3838"], "seg1": "", "seg2": "", "x": 558.9999389648438, "y": 368, "wires":
[] },
{ "id": "63152a46.7f5d34", "type": "ui_gauge", "z": "16891922.0aa4b7", "name": "SaloniHumidity", "g
roup": "6f5730e2.0bb26", "order": 0, "width": "5", "height": "5", "gtype": "gage", "title": "Σχετική
Υγρασία", "label": "units", "format": " {{value}} ", "min": 0, "max": "100", "colors":
["#00b500", "#e6e600", "#ca3838"], "seg1": "", "seg2": "", "x": 547.9999389648438, "y": 403, "wires":
[] }, { "id": "b0416e3e.37136", "type": "mqtt
in", "z": "16891922.0aa4b7", "name": "SaloniHumidity", "topic": "house/humid", "qos": "2", "broker": "
616ef29f.e4189c", "x": 286.99993896484375, "y": 413, "wires": [ ["63152a46.7f5d34"] ] },
{ "id": "883d14b.08cd3e8", "type": "mqtt
in", "z": "16891922.0aa4b7", "name": "Alert", "topic": "saloni/alert", "qos": "2", "broker": "616ef29f.e4
189c", "x": 311.99993896484375, "y": 575, "wires": [ ["b9a556c8.8826e8"] ] },
{ "id": "b9a556c8.8826e8", "type": "ui_audio", "z": "16891922.0aa4b7", "name": "Tune", "group": "73b
e94ae.d68bcc", "voice": "0", "always": "", "x": 520.9999389648438, "y": 567.0000610351562, "wires":
[] },
{ "id": "d8bf62f4.6c2b4", "type": "ui_switch", "z": "16891922.0aa4b7", "name": "KouzinaLights", "lab
el": "Φωτα
Κουζίνας", "group": "8763aef1.ed79f", "order": 0, "width": 0, "height": 0, "passthru": true, "decouple": "f
alse", "topic": "house/klights", "style": "", "onvalue": "ON", "onvalueType": "str", "onicon": "", "oncolor
": "", "offvalue": "OFF", "offvalueType": "str", "officon": "", "offcolor": "", "x": 279.99993896484375, "
y": 178.39996337890625, "wires": [ ["b8300006.75011"] ] },
{ "id": "8ea9a2ee.029c2", "type": "ui_switch", "z": "16891922.0aa4b7", "name": "Oven", "label": "Φού
ρνος", "group": "1d0a4bcb.d004c4", "order": 0, "width": 0, "height": 0, "passthru": true, "decouple": "fals
e", "topic": "house/oven", "style": "", "onvalue": "ON", "onvalueType": "str", "onicon": "", "oncolor": "",
"offvalue": "OFF", "offvalueType": "str", "officon": "", "offcolor": "", "x": 307.99993896484375, "y": 2
26.4000244140625, "wires": [ ["ca270c75.68196"] ] }, { "id": "b8300006.75011", "type": "mqtt
out", "z": "16891922.0aa4b7", "name": "KouzinaLights", "topic": "house/klights", "qos": "1", "retain": "
", "broker": "616ef29f.e4189c", "x": 556.9999389648438, "y": 180.60003662109375, "wires": [ ] },
{ "id": "ca270c75.68196", "type": "mqtt out", "z": "16891922.0aa4b7", "name": "Oven
", "topic": "house/oven", "qos": "2", "retain": "", "broker": "616ef29f.e4189c", "x": 523.9999389648438
, "y": 228.60009765625, "wires": [ ] }, { "id": "19c6b6a1.a2b109", "type": "mqtt
in", "z": "16891922.0aa4b7", "name": "WaterFlood", "topic": "house/flood", "qos": "2", "broker": "616e

```

```

f29f.e4189c","x":296.99993896484375,"y":534,"wires":[["dcbfcd21.2a15f"]]},
{"id":"9748f03f.9cc68","type":"mqtt
in","z":"16891922.0aa4b7","name":"Ειδοποίηση","topic":"house/PIR","qos":"1","broker":"616ef
29f.e4189c","x":297.5,"y":492,"wires":[["334477cd.e82f78"]]},
{"id":"76bd274b.2fdd28","type":"mqtt
in","z":"16891922.0aa4b7","name":"","topic":"house/gas","qos":"1","broker":"616ef29f.e4189c",
"x":304.5,"y":453,"wires":[["b55dc9b9.2604d8"]]},
{"id":"b55dc9b9.2604d8","type":"ui_gauge","z":"16891922.0aa4b7","name":"GasLeak","group":
"71f8188c.01ed28","order":0,"width":0,"height":0,"gtype":"donut","title":"Διαρροή
Γκαζιού","label":"units","format":"{{value}}","min":0,"max":512,"colors":
["#00b500","#e6e600","#ca3838"],"seg1":"","seg2":"","x":530.5,"y":447,"wires":[]},
{"id":"334477cd.e82f78","type":"ui_gauge","z":"16891922.0aa4b7","name":"Intrusion","group":
"73be94ae.d68bcc","order":0,"width":5,"height":5,"gtype":"donut","title":"Εισβολή στο σπίτι
!","label":"units","format":"{{value}}","min":0,"max":100,"colors":
["#00b500","#e6e600","#ca3838"],"seg1":"","seg2":"","x":531.5,"y":487,"wires":[]},
{"id":"dcbfcd21.2a15f","type":"ui_text","z":"16891922.0aa4b7","group":"2fc38dd9.3c1812","ord
er":0,"width":0,"height":0,"name":"","label":"Μήνυμα
Πλημμύρας","format":"{{msg.payload}}","layout":"row-left","x":568.5,"y":527,"wires":[]},
{"id":"977637ce.10fca8","type":"ui_switch","z":"16891922.0aa4b7","name":"balconyLights","la
bel":"Φώτα Μπαλκονιού
ON/OFF","group":"e53837b1.788248","order":0,"width":0,"height":0,"passthru":true,"decouple":
"false","topic":"house/blights","style":"","onvalue":"true","onvalueType":"bool","onicon":"","on
color":"","offvalue":"false","offvalueType":"bool","officon":"","offcolor":"","x":279.5,"y":271,"wi
res":[["a22120ae.c84ad"]]},
{"id":"8ebfc037.81e13","type":"ui_switch","z":"16891922.0aa4b7","name":"watering","label":"Π
ότισμα
ON/OFF","group":"a0e890e6.40c95","order":0,"width":0,"height":0,"passthru":true,"decouple":f
alse,"topic":"house/watwring","style":"","onvalue":"true","onvalueType":"bool","onicon":"","on
color":"","offvalue":"false","offvalueType":"bool","officon":"","offcolor":"","x":295.5,"y":316,"
wires":[["39bc0e3c.f354d2"]]},{"id":"a22120ae.c84ad","type":"mqtt
out","z":"16891922.0aa4b7","name":"","topic":"house/blights","qos":"1","retain":"","broker":"61
6ef29f.e4189c","x":541.5,"y":272,"wires":[]},{"id":"39bc0e3c.f354d2","type":"mqtt
out","z":"16891922.0aa4b7","name":"","topic":"house/watering","qos":"1","retain":"","broker":"6
16ef29f.e4189c","x":550.5,"y":316,"wires":[]},
{"id":"feac0250.84ac7","type":"ui_group","z":"","name":"Air Condition
","tab":"6b48c07c.3b098","order":2,"disp":true,"width":"6"},
{"id":"17b84430.85a71c","type":"ui_group","z":"","name":"alarmSet","tab":"6b48c07c.3b098","

```

```
order":7,"disp":true,"width":"6"},{"id":"616ef29f.e4189c","type":"mqtt-  
broker","z":"","broker":"localhost","port":"1883","clientId":"","usetls":false,"compatmode":true,"  
keepalive":"60","cleansession":true,"willTopic":"","willQos":"0","willPayload":"","birthTopic":  
"","birthQos":"0","birthPayload":""},  
{"id":"71c2ac00.1e3fa4","type":"ui_group","z":"","name":"temp","tab":"6b48c07c.3b098","order  
":9,"disp":true,"width":"6"},  
{"id":"6f5730e2.0bb26","type":"ui_group","z":"","name":"humidity","tab":"6b48c07c.3b098","or  
der":10,"disp":true,"width":"6"},  
{"id":"73be94ae.d68bcc","type":"ui_group","z":"","name":"PIR","tab":"6b48c07c.3b098","order"  
":11,"disp":true,"width":"6"},  
{"id":"8763aef1.ed79f","type":"ui_group","z":"","name":"KouzinaLights","tab":"6b48c07c.3b09  
8","order":3,"disp":true,"width":"6"},  
{"id":"1d0a4bcb.d004c4","type":"ui_group","z":"","name":"Oven","tab":"6b48c07c.3b098","orde  
r":6,"disp":true,"width":"6"},  
{"id":"71f8188c.01ed28","type":"ui_group","z":"","name":"gas","tab":"6b48c07c.3b098","order":  
12,"disp":true,"width":"6"},  
{"id":"2fc38dd9.3c1812","type":"ui_group","z":"","name":"flooding","tab":"6b48c07c.3b098","o  
rder":8,"disp":true,"width":"6"},  
{"id":"e53837b1.788248","type":"ui_group","z":"","name":"balconyLights","tab":"6b48c07c.3b0  
98","order":4,"disp":true,"width":"6"},  
{"id":"a0e890e6.40c95","type":"ui_group","z":"","name":"Watering","tab":"6b48c07c.3b098","or  
der":5,"disp":true,"width":"6"},  
{"id":"6b48c07c.3b098","type":"ui_tab","z":"","name":"house","icon":"dashboard","order":1}]
```

ΠΑΡΑΡΤΗΜΑ VIII-Παράρτημα κεφαλαίου 6

ΠΑΡΑΡΤΗΜΑ ΚΕΦ.6

Κεφάλαιο 6

Ο κώδικας που τρέχει στον εξυπηρετητή ESP32

```
/*
 * Πανεπιστήμιο Αιγαίου
 * Άννα Μαμουζέλου
 */
#include <WiFi.h>
#include <PubSubClient.h>
#include "DHT.h"
//
// Δήλωση SSID και PASSWORD
//
//const char* ssid = "XXXXXXXXXXXX"; //vivliothiki
//const char* password = "XXXXXXXXXXXX"; //vivliothiki
//
const char* ssid = "XXXXXXXXXXXX"; //my router at university
const char* password = "XXXXXXXXXXXX"; // my router an university
/* εύρεση IP διεύθυνσης στο PC/raspberry που τρέχει ο MQTT Server
   για WINDOWS "ipconfig"
   για Linux "ifconfig"
   ή ipscan
 */
const char* mqtt_server = "192.168.1.2";
//
// Ορίζουμε τις εισόδους
//
#define DHTPIN 5
#define DHTTYPE DHT11
```

```

#define lightpin 35
#define gassmokepin 34
#define monoxidepin 36
#define seismospin 38
//
DHT dht(DHTPIN, DHTTYPE);
//
// Δήλωση-Αρχικοποίηση μεταβλητών
//
float temperature = 0;
float humidity = 0;
float light = 0;
float gassmoke = 0;
float monoxide = 0;
float seismos = 0;
//
//Δημιουργία PubSubClient client
//
WiFiClient espClient;
PubSubClient client(espClient);
//
// 8 ελεγχόμενες συσκευές δήλωση των PORTS
//
const char dev1 = 14;
const char dev2 = 27;
const char dev3 = 16;
const char dev4 = 17;
//
//Δήλωση θεμάτων συσκευών εισόδου (DEVICE TOPICS)
//
#define TEMP_TOPIC "home/temp"
#define HUMID_TOPIC "home/humid"
#define LIGHT_TOPIC "home/light"
#define GASSMOKE_TOPIC "home/gassmoke"
#define MONOXIDE_TOPIC "home/monoxide"
#define SEISMOS_TOPIC "home/seismos"
//

```

```

// Δήλωση θεμάτων συσκευών εξόδου (DEVICE TOPICS)
//
#define DEV_TOPIC1 "home/dev1" /* 1=on, 0=off */
#define DEV_TOPIC2 "home/dev2" /* 1=on, 0=off */
#define DEV_TOPIC3 "home/dev3" /* 1=on, 0=off */
#define DEV_TOPIC4 "home/dev4" /* 1=on, 0=off */
//
// Δήλωση message Arrays
//
long lastMsg = 0;
char msg1[20];
char msg2[20];
char msg3[20];
char msg4[20];
char msg5[20];
char msg6[20];
//
//===== Function Prototyping =====
void receivedCallback(char* topic, byte* payload, unsigned int length);
void mqttconnect();
//
//===== Starting SETUP Function =====
//
void setup() {
  Serial.begin(115200);
  // Αρχικά επιδιώκεται σύνδεση στο WiFi
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print("!");
  }
  // Δήλωση των PORT ως έξοδοι

```



```

pinMode(dev1, OUTPUT);
pinMode(dev2, OUTPUT);
pinMode(dev3, OUTPUT);
pinMode(dev4, OUTPUT);
//
// Ρεσέτ των Ρελλέ(Active LOW!)
digitalWrite(dev1, HIGH);
digitalWrite(dev2, HIGH);
digitalWrite(dev3, HIGH);
digitalWrite(dev4, HIGH);
//
//Δήλωση των PORT ως είσοδοι
pinMode(lightpin, INPUT);
pinMode(gassmokepin,INPUT);
pinMode(monoxidepin,INPUT);
pinMode(seismospin, INPUT);
//
Serial.println("");
Serial.println("WiFi connected");
Serial.println("IP address: ");
Serial.println(WiFi.localIP());

// MQTT server με IPAddress
client.setServer(mqtt_server, 1883);
//καλείται όταν λαμβάνει μήνυμα απο τον μεσίτη
client.setCallback(receivedCallback);
// Εναρξη του DHT
dht.begin();
}
//
// Βρόχος διαρκούς επανάληψης
//
void loop() {
// Αν αποσυνδεθεί επεχείρησε επανασύνδεση
if (!client.connected()) {
  mqttconnect();
}
}

```

```

//Διαρκής Ακρόαση για εισερχόμενα μηνύματα απο το μεσίτη
client.loop();
// Επανάληψη μετρήσεων και έκδοση κάθε 3 secs
long now = millis();
if (now - lastMsg > 3000) {
    lastMsg = now;
// Διάβασε τοDHT11 και μετέτρεψε την τιμή σε string
    humidity=dht.readHumidity();
    if (!isnan(humidity)) {
        snprintf (msg1, 20, "%lf", humidity);
    }
//
    temperature = dht.readTemperature();
    if (!isnan(temperature)) {
        snprintf (msg2, 20, "%lf", temperature);
    }
//
    light=analogRead(lightpin);
    light=light/4; //divide by 4
    if (!isnan(light)) {
        snprintf (msg3, 20, "%lf", light);
    }
//
    gassmoke=analogRead(gassmokepin);
    gassmoke=gassmoke/4; //divide by 4
    if (!isnan(gassmoke)) {
        snprintf (msg4, 20, "%lf", gassmoke);
    }
//
    monoxide=analogRead(monoxidepin);
    monoxide=monoxide/4; //divide by 4
    if (!isnan(monoxide)) {
        snprintf (msg5, 20, "%lf", monoxide);
    }
//
    seismos=analogRead(seismospin);
    seismos=seismos/100; //divide by 100

```

```

    if (!isnan(seismos)) {
        snprintf (msg6, 20, "%lf", seismos);
    }
// Έκδοση μηνυμάτων
    client.publish(TEMP_TOPIC, msg2);
    client.publish(HUMID_TOPIC, msg1);
    client.publish(LIGHT_TOPIC, msg3);
    client.publish(GASSMOKE_TOPIC, msg4);
    client.publish(MONOXIDE_TOPIC, msg5);
    client.publish(SEISMOS_TOPIC, msg6);
}
}
//
//===== ΑΚΡΟΑΣΗ ΛΗΨΗΣ ΑΠΟ ΤΟ ΜΕΣΙΤΗ =====
//
void receivedCallback(char* topic, byte* payload, unsigned int length) {
    Serial.print("Message received: ");
    Serial.println(topic);
    String xtopic=topic;
    Serial.print("payload: ");

    if(xtopic=="home/dev1"){
        // Serial.print("payload: ");
        for (int i = 0; i < length; i++) {
            Serial.print((char)payload[i]);
        }
        Serial.println();
        // Λήψη '1' -> ON
        if ((char)payload[0] == '1') {
            digitalWrite(dev1, LOW);
        } else {
            // Λήψη '0' -> OFF
            digitalWrite(dev1, HIGH);
        }
    }
}
//
if(xtopic=="home/dev2"){

```

```

for (int i = 0; i < length; i++) {
  Serial.print((char)payload[i]);
}
Serial.println();
// Λήψη '1' -> ON
if ((char)payload[0] == '1') {
  digitalWrite(dev2, LOW);
} else {
  // Λήψη '0' -> OFF
  digitalWrite(dev2, HIGH);
}
}
//
if(xtopic=="home/dev3"){
for (int i = 0; i < length; i++) {
  Serial.print((char)payload[i]);
}
Serial.println();
// we got '1' -> ON
if ((char)payload[0] == '1') {
  digitalWrite(dev3, LOW);
} else {
  // we got '0' -> OFF
  digitalWrite(dev3, HIGH);
}
}
//
if(xtopic=="home/dev4"){
for (int i = 0; i < length; i++) {
  Serial.print((char)payload[i]);
}
Serial.println();
// Λάβαμε '1' -> ON, τα PELE EINAI activeLOW!
if ((char)payload[0] == '1') {
  digitalWrite(dev4, LOW);
} else {
  // Λάβαμε '0' -> OFF

```

```

    digitalWrite(dev4, HIGH);
}
}
}
//
// =====Σύνδεση στο MQTT=====
//
void mqttconnect() {
    // Επανάλαβε μέχρι τη σύνδεση
    while (!client.connected()) {
        Serial.print("MQTT connecting ...");
        // Δήλωση αυτού του client ID
        String clientId = "ESP32Client";
        //
        if (client.connect(clientId.c_str())) {
            Serial.println("connected");
            //
            // Εγγραφή των θεμάτων συνδρομής
            //
            client.subscribe(DEV_TOPIC1);
            client.subscribe(DEV_TOPIC2);
            client.subscribe(DEV_TOPIC3);
            client.subscribe(DEV_TOPIC4);
            //
        } else {
            Serial.print("failed, status code =");
            Serial.print(client.state());
            Serial.println("try again in 5 seconds");
            // Αναμονή 5 seconds για επανάληψη
            delay(5000);
        }
    }
}
}
}

```

Κεφαλαίο 6

Ο κώδικας σε δομή Json για τη δημιουργία του πλήρους πίνακα εποπτείας και ελέγχου για την εφαρμογή του κεφαλαίου 6

```
[{"id":"fd6a021d.68f61","type":"mqtt
in","z":"16891922.0aa4b7","name":"temp","topic":"home/temp","qos":"2","broker":"2653ba68.f69056"
,"x":90,"y":40,"wires":[["be8a4fd1.9cac7","802533fa.d283"]]}, {"id":"9cacf174.3a5dc","type":"mqtt
out","z":"16891922.0aa4b7","name":"dev1","topic":"home/dev1","qos":"2","retain":"","broker":"2653b
a68.f69056","x":610,"y":60,"wires":[]},
{"id":"7c8dcfd3.c5251","type":"ui_switch","z":"16891922.0aa4b7","name":"dev1","label":"Φωτα
Εσωτερικά","group":"cc16677e.75a048","order":1,"width":4,"height":1,"passthru":true,"decouple":
"false","topic":"home/dev1","style":"","onvalue":1,"onvalueType":"str","onicon":"","oncolor":"","off
value":0,"offvalueType":"str","officon":"","offcolor":"","x":450,"y":60,"wires":
[["9cacf174.3a5dc"]]},
{"id":"be8a4fd1.9cac7","type":"ui_gauge","z":"16891922.0aa4b7","name":"temp","group":"60d88697.
b7d128","order":5,"width":3,"height":4,"gtype":"gage","title":"Θερμοκρασία","label":"units","form
at":"{{value}}","min":12,"max":25,"colors":
["#b50036","#e10f04","#ca3838"],"seg1":"","seg2":"","x":273,"y":20,"wires":[]},
{"id":"6620c50e.214a7c","type":"ui_switch","z":"16891922.0aa4b7","name":"dev2","label":"Συναγερμ
ός
","group":"2d267244.194b6e","order":3,"width":4,"height":1,"passthru":true,"decouple":"false","to
pic":"home/dev2","style":"","onvalue":1,"onvalueType":"str","onicon":"","oncolor":"","offvalue":0,
"offvalueType":"str","officon":"","offcolor":"","x":452,"y":105,"wires":[["d581024f.4f272"]]},
{"id":"d581024f.4f272","type":"mqtt
out","z":"16891922.0aa4b7","name":"dev2","topic":"home/dev2","qos":1,"retain":"","broker":"2653b
a68.f69056","x":617,"y":107,"wires":[]}, {"id":"15543b2c.5c6fd5","type":"mqtt
in","z":"16891922.0aa4b7","name":"humidity","topic":"home/humid","qos":"2","broker":"2653ba68.f6
9056","x":80,"y":120,"wires":[["cb313170.fe6c6","2c7cf05c.600a6"]]},
{"id":"cb313170.fe6c6","type":"ui_gauge","z":"16891922.0aa4b7","name":"Humidity","group":"bd360
962.2e6ed8","order":0,"width":3,"height":4,"gtype":"gage","title":"Σχετική
Υγρασία","label":"units","format":"{{value}}","min":0,"max":100,"colors":
["#00b500","#0080c0","#ca3838"],"seg1":"","seg2":"","x":280,"y":100,"wires":[]},
{"id":"25fd2403.2eba5c","type":"mqtt
in","z":"16891922.0aa4b7","name":"","topic":"home/light","qos":0,"broker":"2653ba68.f69056","x":
80,"y":200,"wires":[["5d9c0792.d5f098","4155c08.7429d4"]]},
{"id":"5d9c0792.d5f098","type":"ui_gauge","z":"16891922.0aa4b7","name":"light","group":"65fb6007.
990e8","order":0,"width":3,"height":4,"gtype":"gage","title":"Φωτεινή
```

```

ένταση", "label": "units", "format": "{value}", "min": 0, "max": 1023, "colors":
["#ffff80", "#e6e600", "#ca3838"], "seg1": "", "seg2": "", "x": 270, "y": 180, "wires": []},
{"id": "447d9c8.6359264", "type": "ui_switch", "z": "16891922.0aa4b7", "name": "dev3", "label": "Κλιματισ
τικό", "group": "a6fcd5db.ac72e8", "order": 5, "width": 4, "height": 1, "passthru": true, "decouple": "false",
"topic": "home/dev3", "style": "", "onvalue": "1", "onvalueType": "str", "onicon": "", "oncolor": "", "offvalue": "
0", "offvalueType": "str", "officon": "", "offcolor": "", "x": 449, "y": 155, "wires": [{"48e0711.579cc9"}]},
{"id": "d90d743d.b25e58", "type": "ui_switch", "z": "16891922.0aa4b7", "name": "dev4", "label": "Θερμοσί
φωνο", "group": "d6b3b1c2.460ee", "order": 6, "width": 4, "height": 1, "passthru": true, "decouple": "false"
, "topic": "home/dev6", "style": "", "onvalue": "1", "onvalueType": "str", "onicon": "", "oncolor": "", "offvalue":
"0", "offvalueType": "str", "officon": "", "offcolor": "", "x": 450, "y": 209, "wires": [{"55ccfcab.9483f4"}]},
{"id": "dd9e3c10.335df", "type": "ui_gauge", "z": "16891922.0aa4b7", "name": "Seismos", "group": "4a17bf
52.3a8f7", "order": 0, "width": 3, "height": 4, "gtype": "gage", "title": "Seismos", "label": "units", "format": "
{value}", "min": 0, "max": 10, "colors":
["#804000", "#e6e600", "#ca3838"], "seg1": "", "seg2": "", "x": 280, "y": 420, "wires": []},
{"id": "93e723bd.bbf6e", "type": "ui_gauge", "z": "16891922.0aa4b7", "name": "C-
Monoxide", "group": "8db424.9c35fbe", "order": 0, "width": 3, "height": 4, "gtype": "gage", "title": "Mono
xide", "label": "units", "format": "{value}", "min": 0, "max": 4096, "colors":
["#ff00ff", "#e6e600", "#ca3838"], "seg1": "", "seg2": "", "x": 290, "y": 340, "wires": []},
{"id": "1eb71e20.69c8a2", "type": "ui_gauge", "z": "16891922.0aa4b7", "name": "Gas+Smoke", "group": "fc
e4917b.f9292", "order": 0, "width": 3, "height": 4, "gtype": "gage", "title": "Gas+Smoke", "label": "units", "
format": "{value}", "min": 0, "max": 4096, "colors":
["#ad3107", "#e6e600", "#ca3838"], "seg1": "", "seg2": "", "x": 290, "y": 260, "wires": []},
{"id": "e5e621bb.fff3c", "type": "mqtt
in", "z": "16891922.0aa4b7", "name": "Gas+Smoke", "topic": "home/gassmoke", "qos": "0", "broker": "2653b
a68.f69056", "x": 70, "y": 280, "wires": [{"1eb71e20.69c8a2", "c0458e9d.60a92"}]},
{"id": "d42e4cb0.2cf4b", "type": "mqtt
in", "z": "16891922.0aa4b7", "name": "Monoxide", "topic": "home/monoxide", "qos": "0", "broker": "2653ba
68.f69056", "x": 80, "y": 360, "wires": [{"93e723bd.bbf6e", "66bc83f4.b5134c"}]},
{"id": "f3b2e99.fdf8218", "type": "mqtt
in", "z": "16891922.0aa4b7", "name": "Seismos", "topic": "home/seismos", "qos": "0", "broker": "2653ba68.f
69056", "x": 80, "y": 440, "wires": [{"dd9e3c10.335df", "db041d40.46da4"}]},
{"id": "48e0711.579cc9", "type": "mqtt
out", "z": "16891922.0aa4b7", "name": "dev3", "topic": "home/dev3", "qos": "2", "retain": "", "broker": "2653b
a68.f69056", "x": 617, "y": 162, "wires": []}, {"id": "55ccfcab.9483f4", "type": "mqtt
out", "z": "16891922.0aa4b7", "name": "dev4", "topic": "home/dev4", "qos": "2", "retain": "", "broker": "2653b
a68.f69056", "x": 617, "y": 215, "wires": []},
{"id": "802533fa.d283", "type": "ui_chart", "z": "16891922.0aa4b7", "name": "Temp", "group": "60d88697.b

```

7d128", "order":0, "width":3, "height":4, "label": "Βαθμοί κελσίου
C", "chartType": "line", "legend": "false", "xformat": "HH:mm:ss", "interpolate": "linear", "nodata": "", "dot": false, "ymin": "10", "ymax": "25", "removeOlder": 1, "removeOlderPoints": "", "removeOlderUnit": "3600", "cutout": 0, "useOneColor": false, "colors":
["#152bbd", "#aec7e8", "#ff7f0e", "#2ca02c", "#98df8a", "#d62728", "#ff9896", "#9467bd", "#c5b0d5"], "useOldStyle": false, "x": 277.5, "y": 59, "wires": [[], []]},
{ "id": "2c7cf05c.600a6", "type": "ui_chart", "z": "16891922.0aa4b7", "name": "Humidity", "group": "bd360962.2ebed8", "order":0, "width":3, "height":4, "label": "%", "chartType": "line", "legend": "false", "xformat": "HH:mm:ss", "interpolate": "linear", "nodata": "", "dot": false, "ymin": "", "ymax": "", "removeOlder": 1, "removeOlderPoints": "", "removeOlderUnit": "3600", "cutout": 0, "useOneColor": false, "colors":
["#1f77b4", "#aec7e8", "#ff7f0e", "#2ca02c", "#98df8a", "#d62728", "#ff9896", "#9467bd", "#c5b0d5"], "useOldStyle": false, "x": 280, "y": 140, "wires": [[], []]},
{ "id": "db041d40.46da4", "type": "ui_chart", "z": "16891922.0aa4b7", "name": "seismos", "group": "4a17bf52.3a8f7", "order":0, "width":3, "height":4, "label": "Σεισμική
ένταση", "chartType": "line", "legend": "false", "xformat": "HH:mm:ss", "interpolate": "linear", "nodata": "", "dot": false, "ymin": "0", "ymax": "10", "removeOlder": 1, "removeOlderPoints": "", "removeOlderUnit": "3600", "cutout": 0, "useOneColor": false, "colors":
["#804000", "#aec7e8", "#ff7f0e", "#2ca02c", "#98df8a", "#d62728", "#ff9896", "#9467bd", "#c5b0d5"], "useOldStyle": false, "x": 280, "y": 460, "wires": [[], []]},
{ "id": "66bc83f4.b5134c", "type": "ui_chart", "z": "16891922.0aa4b7", "name": "C-
monoxide", "group": "8db424.9c35fbc", "order":0, "width":3, "height":4, "label": "CO", "chartType": "line", "legend": "false", "xformat": "HH:mm:ss", "interpolate": "linear", "nodata": "", "dot": false, "ymin": "0", "ymax": "4096", "removeOlder": 1, "removeOlderPoints": "1023", "removeOlderUnit": "3600", "cutout": 0, "useOneColor": false, "colors":
["#ff80ff", "#aec7e8", "#ff7f0e", "#2ca02c", "#98df8a", "#d62728", "#ff9896", "#9467bd", "#c5b0d5"], "useOldStyle": false, "x": 290, "y": 380, "wires": [[], []]},
{ "id": "c0458e9d.60a92", "type": "ui_chart", "z": "16891922.0aa4b7", "name": "Gas+Smoke", "group": "fce4917b.f9292", "order":0, "width":3, "height":4, "label": "Γκάζι+Καπνός", "chartType": "line", "legend": "false", "xformat": "HH:mm:ss", "interpolate": "linear", "nodata": "", "dot": false, "ymin": "0", "ymax": "4096", "removeOlder": 1, "removeOlderPoints": "1023", "removeOlderUnit": "3600", "cutout": 0, "useOneColor": false, "colors":
["#af3823", "#aec7e8", "#ff7f0e", "#2ca02c", "#98df8a", "#d62728", "#ff9896", "#9467bd", "#c5b0d5"], "useOldStyle": false, "x": 290, "y": 300, "wires": [[], []]},
{ "id": "4155c08.7429d4", "type": "ui_chart", "z": "16891922.0aa4b7", "name": "Light", "group": "65fb6007.990e8", "order":0, "width":3, "height":4, "label": "Φωτεινή
ένταση", "chartType": "line", "legend": "false", "xformat": "HH:mm:ss", "interpolate": "linear", "nodata": "", "dot": false, "ymin": "0", "ymax": "1023", "removeOlder": 1, "removeOlderPoints": "", "removeOlderUnit": "3600", "cutout": 0, "useOneColor": false, "colors":
["#af3823", "#aec7e8", "#ff7f0e", "#2ca02c", "#98df8a", "#d62728", "#ff9896", "#9467bd", "#c5b0d5"], "useOldStyle": false, "x": 290, "y": 300, "wires": [[], []]},
{ "id": "4155c08.7429d4", "type": "ui_chart", "z": "16891922.0aa4b7", "name": "Light", "group": "65fb6007.990e8", "order":0, "width":3, "height":4, "label": "Φωτεινή
ένταση", "chartType": "line", "legend": "false", "xformat": "HH:mm:ss", "interpolate": "linear", "nodata": "", "dot": false, "ymin": "0", "ymax": "1023", "removeOlder": 1, "removeOlderPoints": "", "removeOlderUnit": "3600", "cutout": 0, "useOneColor": false, "colors":
["#af3823", "#aec7e8", "#ff7f0e", "#2ca02c", "#98df8a", "#d62728", "#ff9896", "#9467bd", "#c5b0d5"], "useOldStyle": false, "x": 290, "y": 300, "wires": [[], []]},


```
00","cutout":0,"useOneColor":false,"colors":
["#ffff00","#aec7e8","#ff7f0e","#2ca02c","#98df8a","#d62728","#ff9896","#9467bd","#c5b0d5"],"use
OldStyle":false,"x":270,"y":220,"wires":[[],[]],{"id":"2653ba68.f69056","type":"mqtt-
broker","z":"","broker":"192.168.1.2","port":"1883","clientId":"","usetls":false,"compatmode":true,"kee
palive":"60","cleansession":true,"willTopic":"","willQos":"0","willPayload":"","birthTopic":"","birthQo
s":"0","birthPayload":""},
{"id":"cc16677e.75a048","type":"ui_group","z":"","name":"Lights","tab":"6b48c07c.3b098","order":7,"
disp":true,"width":"4"},
{"id":"60d88697.b7d128","type":"ui_group","z":"","name":"Θερμοκρασία","tab":"6b48c07c.3b098","o
rder":1,"disp":true,"width":"4"},{"id":"2d267244.194b6e","type":"ui_group","z":"","name":"Alarm
ON-OFF","tab":"6b48c07c.3b098","order":8,"disp":true,"width":"4"},
{"id":"bd360962.2ebed8","type":"ui_group","z":"","name":"Σχετική
Υγρασία","tab":"6b48c07c.3b098","order":2,"disp":true,"width":"3"},
{"id":"65fb6007.990e8","type":"ui_group","z":"","name":"Φωτισμός","tab":"6b48c07c.3b098","order":
3,"disp":true,"width":"3"},
{"id":"a6fcd5db.ac72e8","type":"ui_group","z":"","name":"Clima","tab":"6b48c07c.3b098","order":9,"
disp":true,"width":"4"},
{"id":"d6b3b1c2.460ee","type":"ui_group","z":"","name":"WaterHeater","tab":"6b48c07c.3b098","orde
r":10,"disp":true,"width":"4"},
{"id":"4a17bf52.3a8f7","type":"ui_group","z":"","name":"Δόνηση","tab":"6b48c07c.3b098","order":6,"
disp":true,"width":"3"},{"id":"8db424.9c35fbe","type":"ui_group","z":"","name":"Μονοξείδιο
CO","tab":"6b48c07c.3b098","order":5,"disp":true,"width":"3"},
{"id":"fce4917b.f9292","type":"ui_group","z":"","name":"Γκάζί+Καπνός","tab":"6b48c07c.3b098","or
der":4,"disp":true,"width":"3"},{"id":"6b48c07c.3b098","type":"ui_tab","z":"","name":"Anna's
House","icon":"dashboard","order":1}]
```

ΠΑΡΑΡΤΗΜΑ VIII- Παράρτημα NODE-RED

Εισαγωγή

Το Node-RED είναι ένα ισχυρό εργαλείο για την ανάπτυξη εφαρμογών του Ίντερνετ των πραγμάτων (IoT) με έμφαση στην απλοποίηση της "καλωδίωσης" των μπλοκ κωδικών για την εκτέλεση εργασιών. Χρησιμοποιεί την προσέγγιση του οπτικού προγραμματισμού που επιτρέπει στους προγραμματιστές να συνδέουν μαζί προκαθορισμένα μπλοκ κώδικα, γνωστά ως «κόμβοι», για να εκτελέσουν μια εργασία. Οι συνδεδεμένοι κόμβοι, συνήθως ένας συνδυασμός κόμβων εισόδου, κόμβων επεξεργασίας και κόμβων εξόδου. Όταν συνδέονται μεταξύ τους, τότε λέμε ότι σχηματίζουν μια ροή.

Αρχικά αναπτύχθηκαν ως έργο ανοιχτού κώδικα στην IBM στα τέλη του 2013, για να ανταποκριθούν στην ανάγκη τους να συνδέουν γρήγορα το υλικό και τις συσκευές με υπηρεσίες ιστού και άλλο λογισμικό - ως ένα είδος διασυνδέτη για το IoT - γρήγορα εξελίχθηκε ως γενικό προγραμματιστικό IoT εργαλείο. Είναι σημαντικό ότι το Node-RED έχει αναπτύξει γρήγορα μια σημαντική και αυξανόμενη βάση χρηστών και μια ενεργή κοινότητα προγραμματιστών που συμβάλλουν με νέους κόμβους οι οποίοι επιτρέπουν στους προγραμματιστές να επαναχρησιμοποιούν τον κώδικα Node-RED για μια μεγάλη ποικιλία εργασιών.

Παρόλο που το Node-RED σχεδιάστηκε αρχικά για να λειτουργεί με το Διαδίκτυο των Πράξεων, δηλαδή συσκευές που αλληλεπιδρούν και ελέγχουν τον πραγματικό κόσμο, όπως έχει εξελιχθεί, έχει καταστεί χρήσιμο για μια σειρά εφαρμογών.

Παρουσιάζονται ορισμένα παραδείγματα Node-RED, τόσο για να σας δώσουν μια αίσθηση των δυνατοτήτων του όσο και για να εισαγάγουν τις βασικές έννοιες. Θα συζητήσουμε επίσης την ιστορία του Node-RED και την προέλευσή του για να σας δώσουμε μια ιδέα για το τι είναι καλό το Node-RED και σε ποιους τύπους εργασιών αποδίδεται καλύτερα σε μια πιο παραδοσιακή γλώσσα προγραμματισμού και στα συνήθη περιβάλλοντα.

Μέχρι το τέλος αυτού του κειμένου, θα πρέπει να έχετε μια κατανόηση υψηλού επιπέδου για το Node-RED, το μοντέλο προγραμματισμού του και μερικούς απλούς προ-κατασκευασμένους κόμβους καθώς και μια κατανόηση της προέλευσης και της καλύτερης χρήσης του Node-RED.

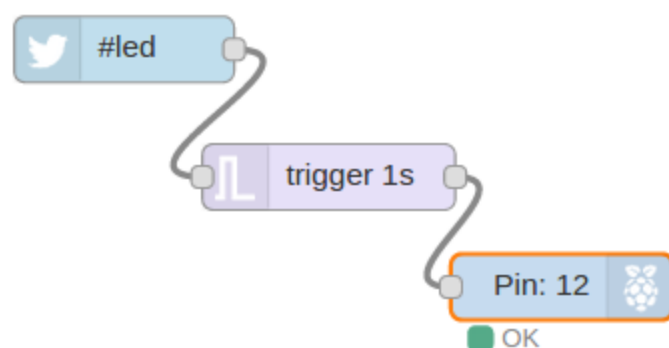
Γνωριμία με το Node-RED

Ας ξεκινήσουμε εξετάζοντας μερικά απλά παραδείγματα των εργασιών που μπορείτε να κάνετε με το Node-RED. Ο στόχος μας είναι να σας δώσουμε μια αίσθηση των δυνατοτήτων του μέσα από μερικά παραδείγματα.

Η πρώτη ροή-παράδειγμα θα λάβει ένα μήνυμα από μια τροφοδοσία Twitter που περιέχει ένα hashtag . Θα ενεργοποιηθεί και θα απενεργοποιηθεί μια λυχνία LED όταν έρθει ένα νέο tweet. Για αυτό το παράδειγμα, ας υποθέσουμε ότι το NODE-RED τρέχει σε ένα Raspberry Pi - μια κοινή κατάσταση για πολλούς χρήστες Node-RED.

Παράδειγμα Χρήση του Twitter για τον έλεγχο ενός Raspberry Pi

Παρουσιάζεται μια ροή Node-RED για να ελέγξετε το Twitter και να δείτε εάν ανιχνεύεται ένα hashtag #led, και ενεργοποιείται ένα LED που συνδέεται με ένα ακροδέκτη του RPi που φαίνεται παρακάτω στο σχήμα. Όπως μπορείτε να δείτε, είναι πολύ απλό. Αποτελείται από 3 συνδεδεμένους κόμβους - έναν κόμβο twitter στα αριστερά, έναν κόμβο ενεργοποίησης στη μέση και έναν κόμβο Raspberry Pi (gpio node) στα δεξιά. (το gpio σημαίνει Εισαγωγή / Έξοδος Γενικού Σκοπού και είναι το όνομα που δίνεται στον γενικό κόμβο Pupil Raspberry που χειρίζεται το IO). Οι κόμβοι, που αντιπροσωπεύονται από θυρίδες με στρογγυλεμένες γωνίες, συνήθως έχουν ένα εικονίδιο που αντιπροσωπεύει τον τύπο τους, π.χ. το εικονίδιο πτηνών Twitter στον πρώτο κόμβο και ένα όνομα κειμένου που μπορεί να οριστεί από τον προγραμματιστή. Κάθε ένας από αυτούς τους τρεις κόμβους ενσωματώνεται στο εργαλείο οπτικού προγραμματισμού Node-RED και μπορεί να μεταφερθεί από την παλέτα κόμβου στον κύριο χώρο εργασίας. Εν συνεχεία «ενσύρματοι» μαζί με τη σύνδεση των καρτελών εξόδου τους στην καρτέλα εισόδου του επόμενου κόμβου στη ροή (οι μικρές γκρι μπάλες στα δεξιά ή στα αριστερά



των κουτιών κόμβων.)

Σχήμα: Μια απλή ροή 3 κόμβων που χρησιμοποιεί ετικέτα Twitter για να ενεργοποιήσει μια λυχνία LED σε ένα Raspberry Pi.

Ο κόμβος Twitter είναι ένας ενσωματωμένος κόμβος στο Node-RED και κρύβει όλη την πολυπλοκότητα της χρήσης του API Twitter. Μπορεί να ρυθμιστεί με τα διαπιστευτήρια λογαριασμού ενός χρήστη και μια ποικιλία από συμβολοσειρές αναζήτησης. Στην περίπτωσή μας απλώς ψάχνει για το hashtag '#led'.

Όταν ο κόμβος Twitter βλέπει την ετικέτα είτε σε ένα τιτίβισμα στον χρήστη, είτε στο δημόσιο tweet feed, δημιουργεί ένα νέο μήνυμα με τις λεπτομέρειες του tweet, το οποίο είναι προς τα εμπρός στον επόμενο κόμβο της ροής.

Στο παράδειγμά μας, αφού η έξοδος του κόμβου του Twitter συνδέεται με την είσοδο του κόμβου ενεργοποίησης, ο κόμβος ενεργοποίησης λαμβάνει ένα μήνυμα με τις λεπτομέρειες του τιτιβίσματος.

Ο κόμβος ενεργοποίησης είναι ένας άλλος ενσωματωμένος κόμβος στο Node-RED και η προεπιλεγμένη συμπεριφορά του είναι να περιμένει οποιοδήποτε μήνυμα στην είσοδο του. Όταν λαμβάνει ένα μήνυμα, «ενεργοποιεί» και στέλνει ένα μήνυμα στην έξοδο του με την τιμή «1» ως το σώμα του μηνύματος. Στη συνέχεια περιμένει 1 δευτερόλεπτο και στέλνει ένα δεύτερο μήνυμα με την τιμή "0" στο σώμα του μηνύματος.

Δεδομένου ότι ο κόμβος ενεργοποίησης συνδέεται με το grionode, το οποίο ελέγχει τους ακροδέκτες εισόδου / εξόδου ή IO στο Raspberry Pi, παίρνει αυτά τα δύο μηνύματα, σε διάστημα 1 δευτερολέπτου, στις εισόδους του.

Και πάλι, ο grionode είναι ένας ενσωματωμένος κόμβος στο Node-RED που κρύβει τις λεπτομέρειες της ρύθμισης των ακίδων IO στο PI. Όταν παίρνει ένα μήνυμα με τιμή "1" στο σώμα, ανεβάζει το δυναμικό στην ακίδα IO ψηλά, και όταν λάβει ένα μήνυμα με το σώμα του "0", κατεβάζει το δυναμικό στο pin χαμηλά. Σε αυτό το παράδειγμα, ο grionode έχει ρυθμιστεί ώστε να ελέγχει το pin 12 .

Εάν έχετε συνδέσει το RPi σας με ένα LED που είναι συνδεδεμένο στον ακροδέκτη 12, το grionode που ανεβάνει στα 5V για 1 δευτερόλεπτο και στη συνέχεια χαμηλώνει, θα αναγκάσει το LED να αναβοσβήσει για 1 δευτερόλεπτο.

Αν σκεφτείτε τι συμβαίνει εδώ, είναι πραγματικά ένα απίστευτα ισχυρό παράδειγμα του Node-RED και του Διαδικτύου των πραγμάτων (IoT). Συνδέει μεν το 'Twitterverse' με ένα PI και ένα LED, χωρίς όμως να απαιτεί προγραμματισμό - όλες οι εργασίες δηλαδή, εκτελούνται από προκατασκευασμένους κόμβους στο Node-RED. Το μόνο που χρειάζονται είναι απλώς διαμόρφωση. Αυτό δείχνει πραγματικά τη δύναμη του Node-RED όταν τρέχει σε μια απλή συσκευή, όπως το Raspberry PI.

Ολοκληρώνοντας θα λέγαμε:

Το Node-RED είναι ένα εργαλείο προγραμματισμού βασισμένο σε ροές με έναν οπτικό επεξεργαστή που σας επιτρέπει να συνδέσετε κόμβους για να δημιουργήσετε αυτές τις ροές. Στην περίπτωση αυτή, 3 κόμβοι δημιουργούν την πρώτη μας ροή. Τα γεγονότα πραγματικού κόσμου, π.χ. με # hashtag, μετατρέπονται σε μηνύματα, τα οποία "ρέουν" κατά μήκος των καλωδίων

μεταξύ των κόμβων. Οι κόμβοι επεξεργάζονται τα μηνύματα στις εισόδους τους και, κατά συνέπεια, στέλνουν μηνύματα στις εξόδους τους και στον επόμενο κόμβο της ροής. Σε αυτό το απλό παράδειγμα, ψάχνουμε για ένα tweet με hashtag #led και το χρησιμοποιούμε για να ενεργοποιήσουμε ένα μήνυμα σε έναν κόμβο Raspberry Pi που προκαλεί τον ακροδέκτη IO 12 να ανεβάσει υψηλό δυναμικό για 1 δευτερόλεπτο και να αναβοσβήνει για 1 δευτερόλεπτο το συνδεδεμένο LED.

Αυτό το παράδειγμα, είναι κυρίως να σας δείξει τα είδη των ενεργειών που μπορούμε να κάνουμε με το Node-RED.

Η περιγραφή του κώδικά της ροής φαίνεται στη συνέχεια:

```
[{"id":"5392b19a.ac6d5","type":"twitter
in","twitter":"","tags":["#led"],"user":"true","name":"","topic":"tweets","x":226,"y":162,"z":"4539e432.bac61c","
wires":[["d5c9e9c9.2a3618"]]}, {"id":"21fdbdaa.de0242","type":"rpi-gpio
out","name":"","pin":"12","set":"","level":"0","out":"out","x":424.99998474121094,"y":270,"z":"4539e432.bac61c"
,"wires":[]},
{"id":"d5c9e9c9.2a3618","type":"trigger","op1":"1","op2":"0","op1type":"val","op2type":"val","duration":"1","ex
tend":false,"units":"s","name":"","x":319,"y":220,"z":"4539e432.bac61c","wires":[["21fdbdaa.de0242"]]}]
```

ΣΥΝΤΟΜΟ ΒΙΟΓΡΑΦΙΚΟ ΣΗΜΕΙΩΜΑ

ANNA MAMOYZEΛΟΥ

Ονομάζομαι Άννα Μαμουζέλου του Μιχαήλ. Είμαι 35 χρονών. Μένω σε μια επαρχιακή πόλη την Αμφιλοχία Αιτωλοακαρνανίας. Είμαι παντρεμένη και μητέρα τριών παιδιών ηλικίας 10, 8 και 3 ετών. Αποφοίτησα από το Ενιαίο Λύκειο Αμφιλοχίας και στη συνέχεια σπούδασα στο Τμήμα Εφαρμογών Πληροφορικής Διοίκησης & Οικονομίας του ΤΕΙ Μεσολογγίου. Κατά τη διάρκεια των σπουδών μου έδειξα άμεσα ενδιαφέρον για τον τομέα της εκπαίδευσης στο μάθημα επιλογής επέλεξα τη διδακτική της πληροφορικής και θέμα πτυχιακής εργασίας «Architecture of web based Teaching & Learning systems». Το αντικείμενο της παρούσας διπλωματικής εργασίας ήταν η μελέτη και η παρουσίαση της αρχιτεκτονικής των συστημάτων μάθησης και διδασκαλίας δια μέσου του διαδικτύου (Internet), καθώς επίσης και η υλοποίηση ενός τέτοιου συστήματος διδασκαλίας.

Μετά την αποφοίτηση μου απασχολήθηκα στο Κέντρο Πληροφορικής Futurekids στο παράρτημα Αμφιλοχίας για ένα (1) σχολικό έτος καθώς επίσης και στο Κέντρο Ξένων Γλωσσών Χρύσα Καραχάλιου στο Λουτρό Αμφιλοχίας για πέντε (5) σχολικά έτη και δίδαξα πληροφορική σε παιδιά ηλικίας 7 έως 14 ετών. Στη συνέχεια απασχολήθηκα στο Ενιαίο Λύκειο Αμφιλοχίας ως ωρομίσθια στην ειδικότητα ΠΕ20 στο πρόγραμμα ενισχυτικής διδασκαλίας για ένα (1) σχολικό έτος. Απασχολήθηκα για τρία (3) συνεχόμενα έτη στο Γραφείο Δημάρχου Δήμου Αμφιλοχίας ως Ειδικός Συνεργάτης του Δημάρχου και στη συνέχεια έχω απασχοληθεί στο Δήμο Αμφιλοχίας με Πεντάμηνη Σύμβαση στο πρόγραμμα Ψηφιοποίηση Δεδομένων με την ειδικότητα ΤΕ Πληροφορικής και με μια πεντάμηνη και μία οχτάμηνη σύμβαση με την ειδικότητα ΤΕ Πληροφορικής στα πλαίσια του προγράμματος προώθησης της απασχόλησης μέσω του προγράμματος Κοινωφελούς Εργασίας του ΟΑΕΔ.