



ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ
ΥΠΟΛΟΓΙΣΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Κατασκευή και εγκατάσταση Smart Office

Βίγκλα Νικολέττα
Κωνσταντινίδης Χρήστος

Εισηγητής: Διλιντάς Γεώργιος, Καθηγητής

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Κατασκευή και εγκατάσταση Smart Office

ΚΩΝΣΤΑΝΤΙΝΙΔΗΣ ΧΡΗΣΤΟΣ Α.Μ. 43080

ΒΙΓΚΛΑ ΝΙΚΟΛΕΤΤΑ Α.Μ. 41792

Εισηγητής:

Διλιντάς Γεώργιος, Καθηγητής

Εξεταστική Επιτροπή:

-

Ημερομηνία εξέτασης

-

Κατασκευή και εγκατάσταση Smart Office

ΚΩΝΣΤΑΝΤΙΝΙΔΗΣ ΧΡΗΣΤΟΣ , ΒΙΓΚΛΑ ΝΙΚΟΛΕΤΤΑ

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα πτυχιακή εργασία ολοκληρώθηκε μετά από επίμονες προσπάθειες, σε ένα ενδιαφέρον γνωστικό αντικείμενο, όπως αυτό της επεξεργασίας κειμένου. Την προσπάθειά μας αυτή υποστήριξε ο επιβλέπων καθηγητής μας, τον οποίο θα θέλαμε να ευχαριστήσουμε.

Ακόμα θα θέλαμε να ευχαριστήσουμε τον κ. Στέλιο Βουτσινά για τις πολύτιμες συμβουλές του.

Επίσης, θα θέλαμε να ευχαριστήσουμε όλους τους καθηγητές της σχολής μας καθώς παρά τις δύσκολες οικονομικές συνθήκες, μας δίδαξαν όσα έπρεπε να γνωρίζουμε έτσι ώστε να βγούμε σωστοί στην εργασιακή αγορά.

ΠΕΡΙΛΗΨΗ

Στόχος της παρούσας πτυχιακής εργασίας, είναι η κατασκευή ενός συστήματος που θα πραγματοποιεί αυτόματα τις ενέργειες έτσι ώστε να παρακολουθείται και να προφυλάσσεται με τις απαραίτητες διαδικασίες το υλικό και ο χώρος στον οποίο θα τοποθετηθούν οι αισθητήρες.

Ενός δηλαδή γραφείου, το οποίο εκμεταλλεύεται τις σύγχρονες υπηρεσίες της τεχνολογίας με σκοπό να παρέχει καλύτερη ασφάλεια και πρόληψη για βλάβες σε υλικό.

Η εντολή για την εκτέλεση των εργασιών γίνεται με τη χρήση μικροπλακέτας υπολογιστή τοποθετημένη στο γραφείο των καθηγητών έτσι ώστε να παρακολουθείται άμεσα από την οθόνη στην οποία θα εμφανίζονται όλες οι ειδοποιήσεις και οι πληροφορίες των αισθητήρων.

Η διαδικασία αυτή πραγματοποιείται μέσω ενός ενσύρματου δικτύου στο οποίο θα αποστέλλονται όλες οι πληροφορίες από τους αισθητήρες. Έπειτα τα αποτελέσματα και οι απαραίτητες εργασίες θα εμφανίζονται στον περιηγητή που θα υπάρχει η αντίστοιχη εφαρμογή.

Η συσκευή που κατασκευάστηκε περιγράφεται στο τρίτο κεφάλαιο μπορεί και διαχειρίζεται το εκάστοτε κλιματιστικό μέσω του πίνακα ηλεκτροδότησης των αιθουσών ενώ ειδοποιεί και για παραβιάσεις (όπως άνοιγμα παράθυρων και πορτών).

Στο πρώτο κεφάλαιο περιγράφεται η λειτουργία ενός έξυπνου γραφείου. Αναφέρονται οι τεχνολογίες και οι βασικές λειτουργίες του, αλλά και το πώς μπορούμε να τις ελέγχουμε.

Το δεύτερο κεφάλαιο περιγράφει την μικροπλακέτα που θα χρησιμοποιήσουμε καθώς και οι διαθέσιμες παραμετροποιήσεις που μπορεί να δεχτεί.

Το τρίτο κεφάλαιο ασχολείται με την κατασκευή συστήματος , την συνδεσμολογία μεταξύ μικροπλακέτας και αισθητήρων αλλά και το πως αυτοί θα ειδοποιούν την εφαρμογή που θα τρέχει στον περιηγητή.

ΕΠΙΣΤΗΜΟΝΙΚΗ ΠΕΡΙΟΧΗ: Αρχιτεκτονική Ηλεκτρονικών Υπολογιστών

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: επεξεργαστής, λανθάνουσα μνήμη, διακοπές, assembly, DMA

ABSTRACT

The purpose of this dissertation is to construct a system that automatically performs the actions in order to monitor and preserve with the necessary procedures the material and the space where the sensors will be placed.

One office, which exploits modern technology services to provide better security, and prevention of material damage.

The order is done by using a computer chalk placed in the teacher's office so that it is immediately monitored by the screen where all alerts and sensor information will be displayed.

This process is carried out via a wired network to which all the information from the sensors will be sent. Then the results and necessary tasks will be done to the browser that will have the corresponding application.

The built-in device described in the third chapter can also manage the air conditioner via the panel's electromechanical panel, as well as alerting for violations (such as opening windows and doors).

The first chapter describes the operation of an intelligent office. They mention the technologies and its basic functions, but also how we can control them.

The second chapter describes the microchip that we will use and the available configurations that it can accept.

The third chapter deals with the construction of a system, the connection between microchips and sensors and how they will notify the application that will run on the browser.

SCIENTIFIC AREA: Computer Architecture

KEYWORDS: processor, latency memory, interrupt, assembly, DMA

Κατασκευή και εγκατάσταση Smart Office

ΚΩΝΣΤΑΝΤΙΝΙΔΗΣ ΧΡΗΣΤΟΣ , ΒΙΓΚΛΑ ΝΙΚΟΛΕΤΤΑ

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΥΧΑΡΙΣΤΙΕΣ.....	5
ΠΕΡΙΛΗΨΗ	6
ABSTRACT	7
ΠΕΡΙΕΧΟΜΕΝΑ	9
ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ.....	11
ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ.....	14
ΚΕΦΑΛΑΙΟ 1: ΣΚΟΠΟΣ ΚΑΙ ΔΟΜΗ.....	15
ΚΕΦΑΛΑΙΟ 2:ΕΙΣΑΓΩΓΗ	17
Αυτοματισμός.....	17
Κατηγορίες Αυτοματισμού.....	18
Arduino.....	19
Internet of Things (IoT).....	20
Προσωπικά δεδομένα.....	20
Πολιτική προστασίας προσωπικών δεδομένων	21
Υπηρεσίες Cloud – Εικονικό μηχάνημα (server)	21
Ιστορική Αναφορά	22
Αξιοσημείωτες Αναφορές	24
Περισσότερες δράσεις	24
ΚΕΦΑΛΑΙΟ 3: ΥΛΙΚΑ ΚΑΙ ΜΕΘΟΔΟΙ.....	27
Εισαγωγή.....	27
Υλικά.....	28
Arduino Uno	28
Ethernet Shield	29
Αισθητήρες	30
Θερμοκρασίας / Υγρασίας DHT-11:	31
Ανίχνευσης κίνησης (PIR Sensor):	31
Μαγνητικής Επαφής (B6-MC38):.....	32
4κάναλος ρελες.....	33
Breadboard.....	34

Καλώδια.....	35
Μακέτα	36
Υπηρεσίες – Προγράμματα.....	37
Εικονικό μηχάνημα server σε υποδομή Cloud.....	37
Node-red.....	48
Dweet.io	50
Arduino IDE	50
Συνδεσμολογία και εφαρμογή	51
ΚΕΦΑΛΑΙΟ 4: ΠΡΟΒΛΗΜΑΤΑ ΠΟΥ ΠΡΟΕΚΥΨΑΝ ΚΑΙ ΟΙ ΕΠΙΛΥΣΕΙΣ ΤΟΥΣ.....	62
ΚΕΦΑΛΑΙΟ 5: ΣΥΜΠΕΡΑΣΜΑΤΑ , ΜΕΛΛΟΝΤΙΚΕΣ ΠΡΟΕΚΤΑΣΕΙΣ ΚΑΙ ΛΕΙΤΟΥΡΓΙΕΣ	63
ΜΕΛΟΝΤΙΚΕΣ ΠΡΟΕΚΤΑΣΕΙΣ/ΛΕΙΤΟΥΡΓΙΕΣ.....	63
ΠΑΡΑΡΤΗΜΑ Α: ΚΩΔΙΚΑΣ ARDUINO	65
Κώδικας	65
Workflow	68
Βιβλιογραφία – Πηγές.....	69

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

Εικόνα 1 Αυτοματισμός	17
Εικόνα 2 Arduino sales example	19
Εικόνα 3 Internet of things.....	20
Εικόνα 4 Internet of things Προστασία Δεδομένων	21
Εικόνα 5 Internet of things 1	21
Εικόνα 6 Cloud	22
Εικόνα 7 Amazon ec2	23
Εικόνα 8 Openlab.....	24
Εικόνα 9 Arduino Contest.....	24
Εικόνα 10 Hackster.io	25
Εικόνα 11 Arduino Smart Home Automation	25
Εικόνα 12 Σχεδιάγραμμα της εργασίας μας.....	28
Εικόνα 13 Arduino Uno	29
Εικόνα 14 Ethernet Shield.....	30
Εικόνα 15 Αισθητήρας Θερμοκρασίας Υγρασίας.....	31
Εικόνα 16 Αισθητήρας Ανίχνευσης κίνησης	32
Εικόνα 17 Αισθητήρας Μαγνητικής Επαφής.....	33
Εικόνα 18 4κάναλος Ρελες.....	34
Εικόνα 19 Breadboard.....	35
Εικόνα 20 Καλώδια	35
Εικόνα 21 Η μακέτα μας	36
Εικόνα 22 Iphost.....	37
Εικόνα 23 ProxmoX γραφικό περιβάλλον	37
Εικόνα 24 Σύνδεση στο proxmoX	38
Εικόνα 25 Εκκίνηση διαδικασίας δημιουργίας εικονικού μηχανήματος.....	38
Εικόνα 26 Δημιουργία Εικονικού μηχανήματος βήμα 1	39
Εικόνα 27 Δημιουργία Εικονικού μηχανήματος βήμα 2	40
Εικόνα 28 Δημιουργία Εικονικού μηχανήματος βήμα 3	41

Εικόνα 29 Δημιουργία Εικονικού μηχανήματος βήμα 4	41
Εικόνα 30 Cpu socket	42
Εικόνα 31 Δημιουργία Εικονικού μηχανήματος βήμα 5	42
Εικόνα 32 Δημιουργία Εικονικού μηχανήματος βήμα 6	43
Εικόνα 33 Δημιουργία Εικονικού μηχανήματος βήμα 7	44
Εικόνα 34 Εγκατάσταση Centos 7 βήμα 1.....	44
Εικόνα 35 Εγκατάσταση Centos 7 βήμα 2.....	45
Εικόνα 36 Εγκατάσταση Centos 7 βήμα 3.....	45
Εικόνα 37 Εγκατάσταση Centos 7 βήμα 4.....	46
Εικόνα 38 Εγκατάσταση Centos 7 βήμα 5.....	46
Εικόνα 39 Εγκατάσταση Centos 7 βήμα 6.....	47
Εικόνα 40 Εγκατάσταση Centos 7 βήμα 7.....	47
Εικόνα 41 Εγκατάσταση Centos 7 βήμα 8.....	48
Εικόνα 42 Node-red	48
Εικόνα 43 Centos 7 yum update	49
Εικόνα 44 freeboard	49
Εικόνα 45 dweet.io	50
Εικόνα 46 Arduino IDE	51
Εικόνα 47 Βιβλιοθήκη dht στο Arduino Ide.....	52
Εικόνα 48 Ορισμός των pin που θα χρησιμοποιήσουμε.....	52
Εικόνα 49 Διαχώρηση εισόδων και εξόδων.....	52
Εικόνα 50 Κύριος κορμός κώδικα.....	53
Εικόνα 51 Εισαγωγή βιβλιοθήκης Ethernet.....	53
Εικόνα 52 Ρύθμισης για σύνδεση στο διαδίκτυο.....	53
Εικόνα 53 Σύνδεση στο διαδίκτυο	54
Εικόνα 54 Αποστολή δεδομένων στο dweet.io.....	54
Εικόνα 55 Εισαγωγή στοιχείων στο node-red	55
Εικόνα 56 Τροποποίηση εισόδου από το dweet.io.....	56
Εικόνα 57 Δημιουργία συναρτήσεων για τον διαχωρισμό των εισόδων	56
Εικόνα 58 Παράδειγμα συνάρτησης για την υγρασία	57
Εικόνα 59 Δημιουργία εξόδων και απεικόνιση τους.....	57

Εικόνα 60 Δημιουργία εικονικών led για την απεικόνιση των καταστάσεων των δεδομένων.....	58
Εικόνα 61 Συνάρτηση τροποποίησης χρώματος εικονικού led.....	58
Εικόνα 62 Απεικόνιση ειδοποιήσεων στο γραφικό	59
Εικόνα 63 Συνάρτηση αναμονής παραλαβής δεδομένων	60
Εικόνα 64 Τελικό σχέδιο node-red	61
Εικόνα 65 Τελικό γραφικό node-red	61
Εικόνα 66 Παράδειγμα λανθασμένων δεδομένων από την μαγνητική επαφή.	62
Εικόνα 68 Workflow.....	68

ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ

TCP : Transmission Control Protocol (Πρωτόκολλο Ελέγχου Μεταφοράς)

IP : Internet Protocol.(Διεύθυνση IP)

LAN : Local Area Network (Τοπικό δίκτυο υπολογιστών)

GUI : Graphical user interface (Γραφικό περιβάλλον χρήστη ή γραφική διασύνδεση/διεπαφή χρήστη)

UDP : User Datagram Protocol

ΚΕΦΑΛΑΙΟ 1: ΣΚΟΠΟΣ ΚΑΙ ΔΟΜΗ

Με την εξέλιξη της τεχνολογίας ολοένα και μεγαλώνει η ανάγκη να απλοποιούμε διαδικασίες στην καθημερινότητά μας. Η χρήση της λέξης Smart υποδηλώνει ότι κάτι είναι έξυπνο.

Ίσως είμαστε περισσότερο εξοικειωμένοι με την ορολογία ‘Smart home’ καθώς την συναντάμε συχνότερα. Υπάρχει ωστόσο και η ορολογία Smart office, ένα “έξυπνο” γραφείο, δηλαδή ένας έξυπνος χώρος εργασίας, ο οποίος έχει αυτοματισμούς ειδικά σχεδιασμένους για την εξυπηρέτηση των ανθρώπων που τον χρησιμοποιούν.

Πλεονεκτήματα

Τα οφέλη που δύναται να μας παρέχει η εγκατάσταση ενός smart office μπορούν να συνοψιστούν αλλά δεν περιορίζονται στα παρακάτω:

Εξοικονόμηση ενέργειας

Καθώς οι συσκευές ελέγχονται και η λειτουργία τους μπορεί να προγραμματιστεί σύμφωνα με τις βέλτιστες επιλογές χωρίς να χρειάζεται κάποια επέμβαση από τον χειριστή. Παράδειγμα τέτοιου προγραμματισμού η ενεργοποίηση του συστήματος ψύξης όταν η θερμοκρασία ξεπερνά τους βαθμούς που έχουν οριστεί.

Ασφάλεια

Ο χρήστης έχει επίσης την δυνατότητα να έχει άμεσα εικόνα του χώρου άλλα και άμεση ενημέρωση για την όποια αλλαγή κατάστασης στον χώρο. Για παράδειγμα αν άνοιξε ένα παράθυρο, αν ανιχνεύτηκε κίνηση στον χώρο ή αν υπάρξει μεταβολή στην θερμοκρασία.

Ποιότητα στον χώρο εργασίας οι αυτοματισμοί μας εξοικονομούν χρόνο, ενέργεια και επιτρέπουν στον χρήστη να αποδώσει στο μέγιστο των δυνατοτήτων του.

Έλεγχος

Καθώς ο προγραμματισμός του Smart Office είναι ευέλικτος έχει ως αποτέλεσμα να επιτρέπει στον χρήστη την να ορίσει με τους δικούς του περιορισμούς και να επιλέξει τις ρυθμίσεις που για τον χειρισμό του γραφείου.

Ο χειριστής για παράδειγμα έχει την δυνατότητα να ορίσει τα επιθυμητά επίπεδα υγρασίας και θερμοκρασίας του χώρου ώστε να μπορεί να ενεργοποιηθεί αυτόματα το σύστημα θέρμανσης/ψύξης εξασφαλίζοντας την διατήρηση των επιθυμητών τιμών στον χώρο.

Ακόμη, ο χρήστης μπορεί να έχει άμεση ενημέρωση μέσω email για οποιαδήποτε αλλαγή σχετικά με τις επιτρεπόμενες μεταβλητές στο χώρο. Τέλος, το smart office συνιστά στην εξασφάλιση της ασφάλειας του χώρου μέσω δυνατοτήτων όπως η ανίχνευση της κίνησης και ανίχνευση ανοιχτής πόρτας ή παράθυρου.

Σκοπός

Σκοπός της πτυχιακής είναι η παρουσίαση των μεθόδων διαχείρισης ενός εργασιακού χώρου αυτόματα, διατηρώντας τον έλεγχο ακόμα και από απόσταση. Άμεση ενημέρωση για αλλαγές στην κατάσταση του χώρου. Αρχικά, θα γίνει αναφορά στα υλικά που χρησιμοποιήθηκαν για την υλοποίηση της εργασίας καθώς και θα επεξηγηθεί ο προγραμματισμός αλλά και η εγκατάσταση τους. Στην συνέχεια, θα παρουσιαστούν τα προβλήματα που προέκυψαν κατά την υλοποίηση της εργασίας με τον τρόπο επίλυσης τους. Τέλος, θα ακολουθήσουν προτάσεις για την μελλοντική επέκταση του project.

ΚΕΦΑΛΑΙΟ 2:ΕΙΣΑΓΩΓΗ

Αυτοματισμός

Καθώς ο άνθρωπος ήθελε να απλουστεύσει τα καθημερινά του προβλήματα , ξεκίνησε να ασχολείται με τον αυτοματισμό. Ο αυτοματισμός είναι η δημιουργία ενός προγράμματος ή ενός αλγορίθμου για την επίλυση ενός ή περισσότερων προβλημάτων σε συνεργασία με έναν αυτόνομο μηχανισμό που εκτελεί τον συγκεκριμένο κώδικα χωρίς κάποια ανθρώπινη παρέμβαση.

Ο αυτοματισμός ξεκίνησε να υφίσταται κατά τον 20ο αιώνα, από την μηχανολογία σε συνδυασμό με την ηλεκτρολογία ως μία διαδικασία ελέγχου των διεργασιών και την κατάλληλη επεξεργασία τους για την επίλυση προβλημάτων.



Εικόνα 1 Αυτοματισμός

Ο αυτοματισμός αποτελεί υψηλής σημαντικότητας στοιχείο της βιομηχανίας, καθώς μειώνει σημαντικά την ανάγκη για ανθρώπινη παρέμβαση, αξιοποιώντας υπολογιστές και προγραμματιζόμενους λογικούς ελεγκτές (PLC), για να συγχρονίσουν τη ροή εισόδων από φυσικούς αισθητήρες με τη ροή εντολών προς τους αισθητήρες εξόδου.

Ως είσοδος θεωρείται ένα σήμα, αναλογικό ή ψηφιακό, συλλεγόμενο από κάποιο σημείο του συστήματος.

Αυτά τα αποτελέσματα συνήθως αναπαρίστανται με μαθηματικές συναρτήσεις, τις συναρτήσεις μεταφοράς. Μία συνάρτηση μεταφοράς προσδιορίζει ένα σύστημα και τον τρόπο που μεταβάλλει κάθε σήμα εισόδου. Η έξοδος του συστήματος ονομάζεται «Αναφορά» και ανταποκρίνεται στην τιμή του σήματος κατόπιν ενεργοποίησης των προηγούμενων συναρτήσεων μεταφορών σε αυτήν.

Όταν μια ή περισσότερες μεταβλητές εξόδου ενός συστήματος πρέπει να ακολουθήσουν την τιμή κάποιας αναφοράς που μεταβάλλεται με τον χρόνο, χρειάζεται να προστεθεί ένας ελεγκτής που να χειρίζεται τις τιμές των σημάτων εισόδου έως ότου επιτευχθεί η επιθυμητή έξοδος.

Κατηγορίες Αυτοματισμού

Σύμφωνα με την τρέχουσα τεχνολογία υπάρχουν δύο κατηγορίες ελέγχου: ανοιχτού και κλειστού βρόγχου.

Το σύστημα ανοιχτού βρόγχου είναι το σύστημα στο οποίο η είσοδος στο σύστημα είναι ανεξάρτητη από την έξοδο. Για παράδειγμα, εάν υπάρχει ένα σύστημα το οποίο κάνει έλεγχο στην θερμοκρασία ενός δωματίου και έχει ακόμη μία λειτουργία που μπορεί να ανοιγοκλείνει τα φώτα, και δεν του έχει οριστεί να ελέγχεται η θερμοκρασία για το άνοιγμα των φώτων τότε είναι ανοιχτού βρόγχου.

Εάν όμως το συγκεκριμένο σύστημα έλεγχε την θερμοκρασία και σύμφωνα με κάποια όρια θερμοκρασίας ανοιγόκλεινε τα φώτα, τότε θα θεωρούνταν κλειστού βρόγχου.

Η παρούσα εργασία είναι κατηγορίας κλειστού βρόγχου καθώς πραγματοποιούνται πολλοί παρόμοιοι έλεγχοι.

Arduino

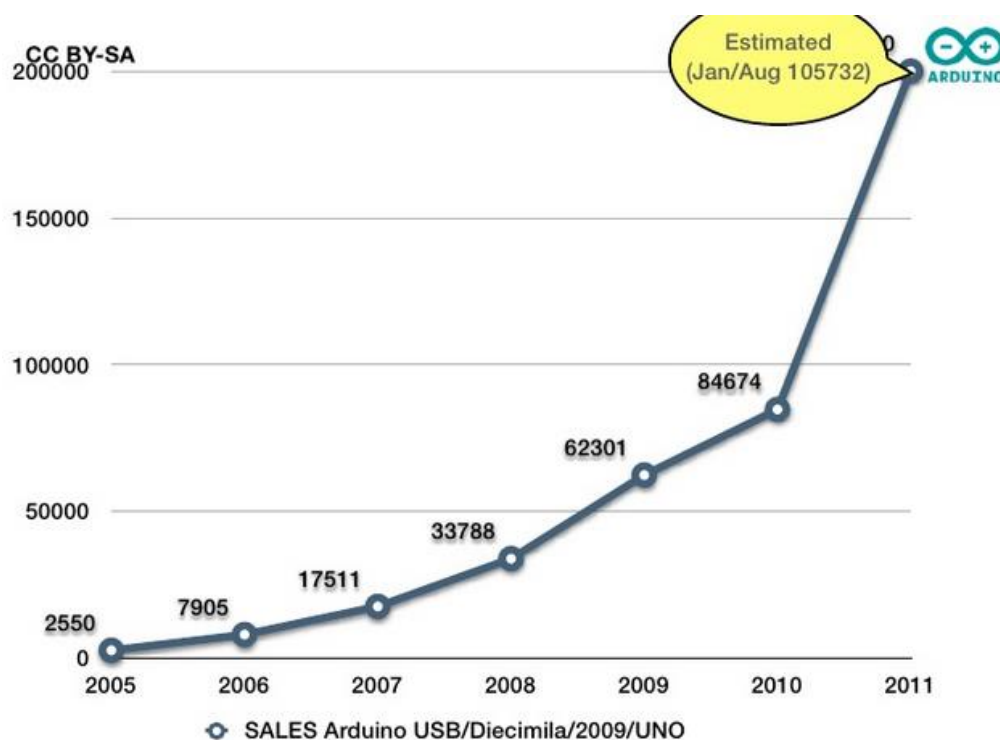
Το Arduino είναι μία πλακέτα που έχει ενσωματωμένο έναν μικροελεγκτή τον οποίον μπορούμε να προγραμματίσουμε έτσι ώστε να πραγματοποιεί οποιαδήποτε ενέργεια επιθυμούμε.

Το Arduino αποτελείται από μία μικρή σε μέγεθος μητρική πλακέτα που περιέχει τον μικροελεγκτή καθώς και εισόδους και εξόδους που μπορούμε να χρησιμοποιήσουμε.

Ο τρόπος προγραμματισμού του συγκεκριμένου μικροελεγκτή γίνεται μέσα από το Android IDE, ένα πρόγραμμα ανοιχτού κώδικα που παρέχεται δωρεάν. Το συγκεκριμένο πρόγραμμα έχει δημιουργηθεί πάνω σε java καθώς και σε άλλες γλώσσες (ανοιχτού κώδικα) προγραμματισμού.

Υπάρχουν πάρα πολλά μοντέλα πλακετών Arduino που διαφέρουν σύμφωνα με τις απαιτήσεις του αγοραστή.

Παρακάτω παρατίθεται ένα παραδειγματικό γράφημα σχετικά με πωλήσεις του Arduino από το 2005 έως το 2011 (το οποίο προέρχεται από το (http://opensourceecology.org/wiki/Arduino_Sales) :



Εικόνα 2 Arduino sales example

Όπως είναι εμφανές , οι πωλήσεις ανεβαίνουν δυναμικά κάθε χρόνο.

Αυτό σημαίνει ότι όλο και περισσότεροι χρήστες – προγραμματιστές έχουν αρχίσει να χρησιμοποιούν την συγκεκριμένη πλακέτα για να μπορέσουν να επιλύσουν τα προβλήματα τους ή ακόμη και των πελατών τους.

Μέσα από έρευνα παρατηρήσαμε ότι πολλές σχολές της Ελλάδας προωθούν τις εργασίες αξιοποιώντας το Arduino. Υπάρχουν αμέτρητες πτυχιακές εργασίες πάνω στον αυτοματισμό ο οποίος έχει θετική συνεισφορά στους φοιτητές καθώς τους

βοηθάει να διευρύνουν τους ορίζοντες του προγραμματισμού συνδέοντάς τον με hardware υλικό.

Internet of Things (IoT)

Το Διαδίκτυο των Πραγμάτων ή Ίντερνετ των Πραγμάτων (αγγλικά: Internet of Things) αποτελεί η συνδεσμολογία όλων των φυσικών (ηλεκτρικών-ηλεκτρονικών) αντικειμένων με το διαδίκτυο. Αυτό πρακτικά σημαίνει ότι μέσα από κατάλληλές διεργασίες μπορούμε να αποστείλουμε δεδομένα στο διαδίκτυο που τα έχουμε λάβει ως εισόδους στο σύστημα μας , και να τα ελέγξουμε από οπουδήποτε αρκεί να είμαστε συνδεδεμένοι στο διαδίκτυο με τον απαραίτητο τρόπο.



Εικόνα 3 Internet of things

Η έννοια "Things" αναφέρεται σε μία ευρεία ποικιλία συσκευών εντελώς διαφορετικά μεταξύ τους. Ο όρος Internet of Things αποδόθηκε την δεκαετία του 1990 από τον Kevin Ashton.

Το IoT αποτελεί την τελευταία λέξη της τεχνολογίας. Ως εκ τούτου, βρίσκεται σε ένα στάδιο συνεχούς αναβάθμισης και βελτίωσης. Ένα ακόμα σημαντικότερο πρόβλημα που αντιμετωπίζει είναι η ασφάλεια των δεδομένων που αποθηκεύει.

Προσωπικά δεδομένα

Τα προσωπικά δεδομένα είναι οι πληροφορίες που επεξεργάζεται το IoT για το σύνολο των συνδεδεμένων συσκευών. Σίγουρα όμως θα πρέπει ο εκάστοτε χρήστης να αισθάνεται σίγουρος ότι οι πληροφορίες και τα προσωπικά του δεδομένα δεν επηρεάζονται.



Εικόνα 6 Cloud

Το Cloud έχει δώσει την δυνατότητα σε διάφορους οργανισμούς και υπηρεσίες να αποθηκεύουν τις εφαρμογές του και τα δεδομένα τους χωρίς να χρειάζονται κάποιον τοπικό υπολογιστή γι' αυτόν το σκοπό.

Κατά συνέπεια σε πολλές περιπτώσεις η συγκεκριμένη λύση εξοικονομεί χρήματα για τις εταιρίες.

Το cloud πολλές φορές δίνει την δυνατότητα σε εφαρμογές να λειτουργούν γρηγορότερα από ότι θα έκαναν σε έναν τοπικό υπολογιστή και δίνει δυνατότητα πάντα για αυξομείωση των πόρων του μηχανήματος για την βέλτιστη απόδοση του σχεδόν άμεσα.

Ιστορική Αναφορά

Από την κυκλοφορία του Amazon EC2 το 2006, η διαθεσιμότητα δικτύων υψηλής χωρητικότητας, χαμηλού κόστους υπολογιστές και συσκευές αποθήκευσης, καθώς και η ευρεία υιοθέτηση του virtualization υλικού, της αρχιτεκτονικής προσανατολισμένης στις υπηρεσίες και της αυτόνομης υπολογιστικής χρησιμότητας οδήγησαν στην ανάπτυξη του cloud χρήση υπολογιστή.



Amazon EC2

Εικόνα 7 Amazon ec2

Αξιοσημείωτες Αναφορές

Περισσότερες δράσεις

Ανά τον κόσμο υπάρχουν πάρα πολλές εταιρίες και σχολές όπου πραγματοποιούνται διαγωνισμοί σχετικά με προγράμματα ανοιχτού κώδικα πάνω σε arduino με διάφορες ανταμοιβές (όπως υποτροφίες κλπ).

Μία από αυτές είναι η openlab που πραγματοποιείται και στη σχολή μας.



Εικόνα 8 Openlab

Για να συμμετάσχει κανείς στον διαγωνισμό θα πρέπει να παραβρεθεί στο μέρος που θα οργανωθεί το οποίο ανακοινώνεται μέσα στην ιστοσελίδα .

Ιδανικές προοπτικές για την συμμετοχή είναι και η ειδίκευση πάνω στο Arduino.

Επίσης υπάρχει και το arduino contest



Εικόνα 9 Arduino Contest

Για να συμμετάσχει κανείς στον συγκεκριμένο διαγωνισμό πρέπει να γράψει κώδικα για arduino που να συμβαδίζει με τους κανόνες και τις οδηγίες που βρίσκονται στο site τους μέσα σε συγκεκριμένο χρονικό διάστημα.

Επίσης , υπάρχει το hackster.io :



Εικόνα 10 Hackster.io

Το οποίο είναι διαγωνισμός δημιουργίας κώδικα arduino σε άμεση συσχέτιση με το gui ή το λειτουργικό της Microsoft. Η υποβολή συμμετοχής στον συγκεκριμένο διαγωνισμό πραγματοποιείται μέσω αίτησης και, στη συνέχεια, θα πρέπει να αποσταλεί ο κώδικας μέχρι ορισμένη ημερομηνία.

Αξιοσημείωτες βρήκαμε τις παρακάτω Δράσεις:



Εικόνα 11 Arduino Smart Home Automation

Εφαρμογή «**Arduino Smart Home Automation**» .Η δυνατότητα σύνδεσης του Arduino και των αποτελεσμάτων του σε μία εφαρμογή σε κινητό μας υπόσχεται πολλές ελπίδες για ανάπτυξη στο μέλλον.

Καθώς η εκδότρια εταιρία Amrhan έχει καταφέρει να δημιουργήσει μία εφαρμογή που θα καλύψει τις ανάγκες του ευρύτερου κοινού.

Δυνατότητες υποστήριξης υλικού:

- Έλεγχος ψηφιακής εξόδου
- 4 αισθητήρες αναλογικής εισόδου
- Έλεγχος μέσω TCP / IP ή Internet
- Έλεγχος με το πρόγραμμα περιήγησης στο Web ή την εφαρμογή Android

Δυνατότητες λογισμικού:

- διακομιστής Web Arduino TCP / IP (LAN)
- Ρύθμιση στο όνομα της συσκευής και στο όνομα του αισθητήρα
- Έξοδος ελέγχου αισθητήρα αυτόματα.

Κατασκευή και εγκατάσταση Smart Office

- Ρύθμιση της διεύθυνσης IP και της θύρας
- Κατάσταση πραγματικού χρόνου

ΚΕΦΑΛΑΙΟ 3: ΥΛΙΚΑ ΚΑΙ ΜΕΘΟΔΟΙ

Εισαγωγή

Σε αυτό το κεφάλαιο θα αναφέρουμε τα υλικά και τους μεθόδους που χρησιμοποιήσαμε για να πραγματοποιήσουμε την εργασία.

Ο γενικός σκοπός της παρούσας εργασίας είναι ο απομακρυσμένος έλεγχος των αιθουσών του εργαστηρίου ή οποιοδήποτε γραφείου ώστε να μπορούμε να διαχειριζόμαστε και να προστατεύουμε εύκολα τις αίθουσες μας.

Θετικά:

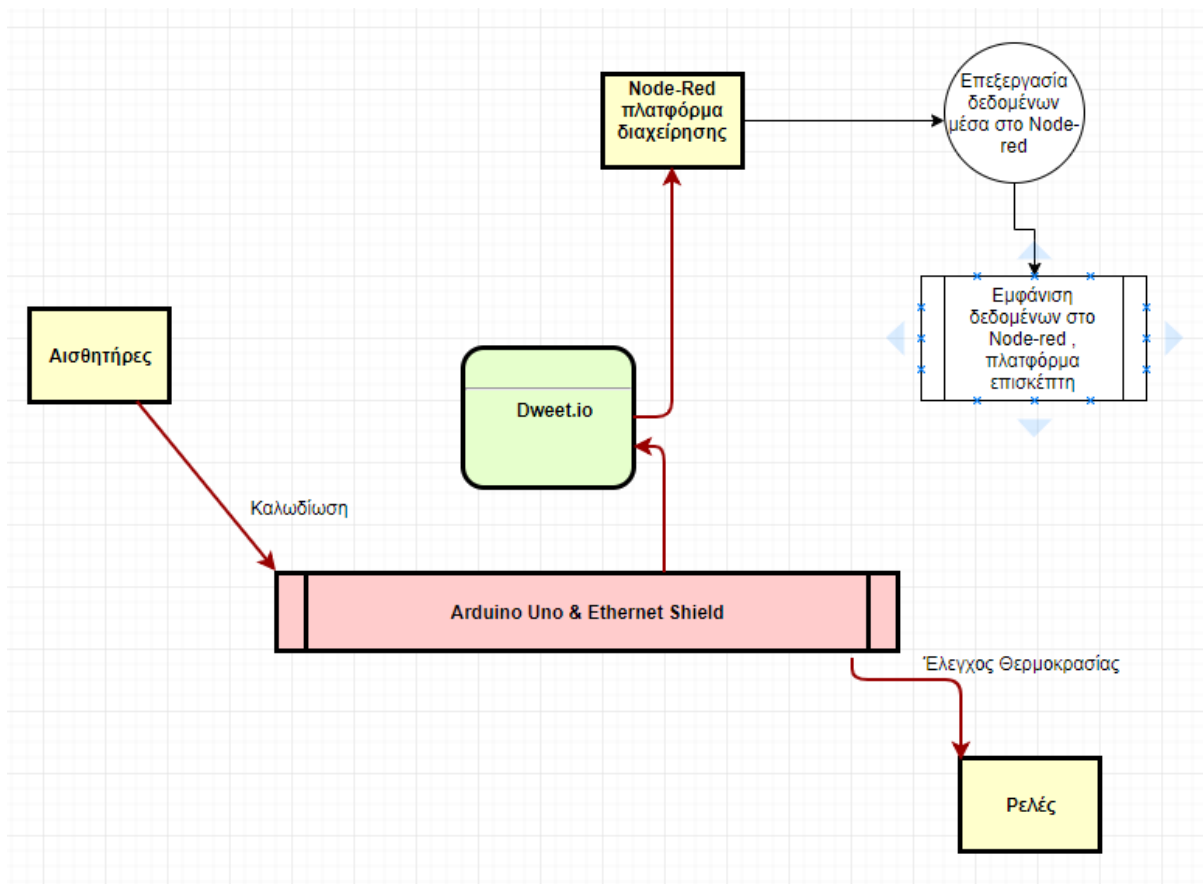
- Εύκολη διαχείριση των απεικονιζόμενων στοιχείων στο node-red
- Εύκολη πρόσβαση μέσα από οποιαδήποτε συσκευή έχει internet και browser (περιηγητή)
- Ακρίβεια δεδομένων
- Μη απώλεια δεδομένων
- Δεν απαιτείται χώρος για την αποθήκευση δεδομένων καθώς όλα λειτουργούν μέσω online υπηρεσιών αποθηκευμένες στο Cloud.
- Αμεσότητα στα αποτελέσματα που έρχονται στην συσκευή μας με πάρα πολύ μικρή καθυστέρηση η οποία είναι όμως λογική (ακολουθεί η αιτιολόγηση παρακάτω στην παρούσα εργασία)

Αρνητικά:

- Ο ολοκληρωτικός έλεγχος βασίζεται στη διαθεσιμότητα του τοπικού δικτύου και κατά πόσο αυτό συνδέεται στο διαδίκτυο
- Εμφάνιση πληροφοριών δημόσια και οι οποίες αποστέλλονται στο dweet.io
- Εξάρτηση υπηρεσίας από εικονικό μηχάνημα , server. Αυτό σημαίνει ότι εάν το μηχάνημα δεν λειτουργεί ή δεν έχει δίκτυο, τότε δεν θα μπορούμε να βλέπουμε-παρατηρούμε τους χώρους στους οποίους έχουμε πραγματοποιήσει την συγκεκριμένη εγκατάσταση.

Ο εξειδικευμένος σκοπός της εργασίας μας είναι αρχικά η παραλαβή δεδομένων από τα διαθέσιμα αισθητήρια που είναι συνδεδεμένα στο Arduino και η μετέπειτα επεξεργασία τους μέσα από τον κώδικα που θα δημιουργήσουμε έτσι ώστε να εκπληρώσουμε τους στόχους της εργασίας.

Ένα μέρος από τα δεδομένα που θα επεξεργαστούν μέσα στο arduino θα αποσταλούν στο dweet.io και τα υπόλοιπα θα χρησιμοποιηθούν για βασικές λειτουργίες και ελέγχους.



Εικόνα 12 Σχεδιάγραμμα της εργασίας μας

Τα δεδομένα που θα αποσταλούν στο dweet.io θα γίνουν επεξεργασία μέσα από το διαχειριστικό panel του node-red. Θα δημιουργηθούν οι κατάλληλες ειδοποιήσεις αλλά και έλεγχοι μέσα στο node-red έτσι ώστε να επιβεβαιώσουμε ότι τα δεδομένα που εμφανίζουμε είναι σωστά.

Τέλος, θα εμφανιστούν δεδομένα και κατάλληλες ειδοποιήσεις μέσα στο panel επισκέπτη του node-red.

Υλικά

Arduino Uno

Το Arduino Uno είναι μία πλακέτα με ενσωματωμένο μικροελεγκτή τον οποίο μπορούμε να προγραμματίσουμε ώστε να πραγματοποιεί οποιαδήποτε ενέργεια

επιθυμούμε. Αποτελείται από μία μικρή σε μέγεθος μητρική πλακέτα που περιέχει τον μικροελεγκτή καθώς και εισόδους και εξόδους που μπορούμε να χρησιμοποιήσουμε.

Ο τρόπος προγραμματισμού του συγκεκριμένου μικροελεγκτή γίνεται μέσα από το Android IDE , ένα πρόγραμμα ανοιχτού κώδικα που παρέχεται δωρεάν από την εκδότρια αρχή την Arduino. Το συγκεκριμένο πρόγραμμα έχει δημιουργηθεί πάνω σε java καθώς και σε άλλες γλώσσες (ανοιχτού κώδικα) προγραμματισμού.

Παρακάτω βλέπουμε το Arduino Uno.



Εικόνα 13 Arduino Uno

Το Arduino Uno είναι η πιο κοινή έκδοση της οικογένειας των μοντέλων Arduino.

Περιέχει τον μικροελεγκτή ATmega328 .

Έχει 14 ψηφιακούς ακροδέκτες εισόδου / εξόδου και 6 αναλογικές εισόδους.

Περιέχει θύρα για σύνδεση με usb στον υπολογιστή καθώς και 16mhz κεραμικό συντονιστή.

Έχει υποδοχή τροφοδοσίας από 7V μέχρι 12V καθώς και ένα reset button για επαναφορά του κυκλώματος .

Ethernet Shield

Το Ethernet Shield είναι μία πλακέτα που δίνει δυνατότητα σύνδεσης του Arduino με δίκτυο είτε αυτό είναι τοπικό είτε δημόσιο.

Λειτουργεί με τροφοδοσία 5v τα οποία λαμβάνει από την πλακέτα του Arduino.

Διαθέτει έναν ελεγκτή Ethernet w5100 με ενσωματωμένη 16kb μνήμη ο οποίος δίνει την δυνατότητα για υλοποίηση είτε TCP είτε UDP προτύπων.

Η ταχύτητα του είναι της μορφής 10/100mb. Υποστηρίζει μέχρι 4 ταυτόχρονες συνδέσεις τύπου Socket.

Διαθέτει ένα βύσμα τύπου RJ45 για το Ethernet καθώς και ένα Reset Button για να μπορούμε να επανεκκινήσουμε το Arduino.



Εικόνα 14 Ethernet Shield

Η πλακέτα έχει μία διαθέσιμη υποδοχή για micro-SD κάρτα (χρησιμοποιώντας την αντίστοιχη βιβλιοθήκη μέσα στο Arduino IDE).

Οι θύρες που δεν μπορούμε να χρησιμοποιήσουμε λόγω χρήσης τους από το Ethernet Shield είναι η 3,11,12,13.

Πάνω στην πλακέτα υπάρχουν διαθέσιμα led τα οποία έχουν τον ρόλο τους :

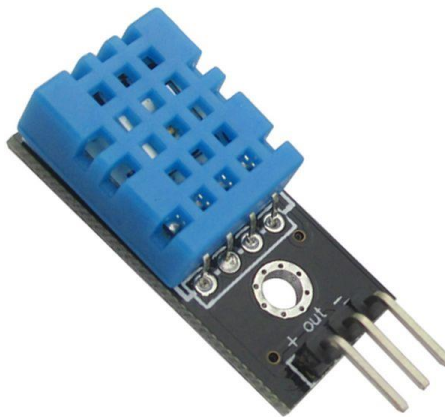
- PWR: Το οποίο μας δείχνει αν η πλακέτα έχει τροφοδοσία
- LINK: Το οποίο μας δείχνει εάν υπάρχει συνδεδεμένο καλώδιο, καθώς αναβοσβήνει όταν η πλακέτα συναλλάσσει δεδομένα
- FULLD: Το οποίο μας δείχνει εάν η σύνδεση είναι πλήρως αμφίδρομη
- 100MB: Το οποίο μας δείχνει εάν η σύνδεση υποστηρίζει ταχύτητα 100mb
- RX: Το οποίο αναβοσβήνει όταν η πλακέτα λαμβάνει δεδομένα
- TX: Το οποίο αναβοσβήνει όταν η πλακέτα στέλνει δεδομένα
- COLL: Το οποίο αναβοσβήνει όταν υπάρξει κάποια εμπλοκή στο δίκτυο

Αισθητήρες

Θερμοκρασίας / Υγρασίας DHT-11:

Ο αισθητήρας DHT-11 είναι ένας απλός και αξιόπιστος αισθητήρας. Καταναλώνοντας λίγο ρεύμα και με μεγάλη ευαισθησία μπορεί να μας δώσει αξιόπιστα αποτελέσματα που μπορούμε να χρησιμοποιήσουμε οπουδήποτε επιθυμούμε.

Το μόνο που αρκεί είναι να έχουμε πάντα εγκατεστημένη την βιβλιοθήκη του “DHT” μέσα στο προγράμματα του Arduino IDE (αναφέρεται παρακάτω)



Εικόνα 15 Αισθητήρας Θερμοκρασίας Υγρασίας

Ανίχνευσης κίνησης (PIR Sensor):

Ένας αισθητήρας PIR χρησιμοποιείται για να αισθανθεί – ανιχνεύσει την κίνηση όλων των αντικειμένων στο χώρο. Συνήθως χρησιμοποιούνται σε συναγερμούς και σε αυτοματοποιημένα συστήματα.

Χρησιμοποιήσαμε τον συγκεκριμένο αισθητήρα για να μπορούμε να έχουμε άμεσα αξιόπιστα αποτελέσματα καθώς και δυνατότητες παραμετροποίησης του αισθητήρα από τους δύο πορτοκαλί διακόπτες που υπάρχουν επάνω στην συσκευή.

Ο αισθητήρας τροφοδοτείται από 5v από το Arduino. Επιπλέον, χρειάζεται μία γείωση και έχει μία έξοδο, που βγάζει το λογικό 1 ή 0, που ως είσοδο στο Arduino μας βοήθησε να γνωρίζουμε εάν υπάρχει κίνηση στον χώρο τον οποίο ελέγχουμε .

Λόγω αδυναμίας εφαρμογής της εργασίας μας σε αίθουσες έχουμε χρησιμοποιήσει έναν αισθητήρα μέσα στην μακέτα που φτιάξαμε.



Εικόνα 16 Αισθητήρας Ανίχνευσης κίνησης

Μαγνητικής Επαφής (B6-MC38):

Η μαγνητική επαφή mc38 είναι από τις πιο αποτελεσματικές χαμηλού κόστους μαγνητικές επαφές που υπάρχουν σχετικά με αυτοματισμούς.

Υπάρχουν δύο καλώδια από τα οποία το ένα χρειάζεται να τροφοδοτηθεί με γείωση και το άλλο να καταλήξει στο Arduino. Διεγείροντας την τάση του ρεύματος κάθε φορά που ο μαγνήτης πλησιάζει την μαγνητική επαφή, λαμβάνουμε την σχετική είσοδο στο Arduino μας από το καλώδιο που έχει εισαχθεί ως είσοδος στο Arduino.



Εικόνα 17 Αισθητήρας Μαγνητικής Επαφής

Για την κατανόηση της λειτουργίας του συγκεκριμένου αισθητήρα μας βοήθησε το βίντεο του «Andr.oid Eric» (https://www.youtube.com/watch?v=aMa7_NYSq3U) σχετικά με την πλατφόρμα του freeboard.io και τον βιβλιοθηκών της. Επίσης, συνέβαλε στο να κατανοήσουμε την λειτουργία του αισθητήρα dht11.

4κάναλος ρελες

Για να τεθεί σε λειτουργία ένας ηλεκτρολογικός ρελές με 4 κανάλια τροφοδοτείται με 5v από το Arduino ενώ παράλληλα χρειάζεται μία γείωση.

Ο διαμοιρασμός του ρεύματος πραγματοποιείται από το αριστερό ριπ της κάθε εξόδου όπου εκεί εισάγεται η κανονική τάση που θέλουμε να δώσουμε. Έπειτα, στο μεσαίο ριπ, βάζουμε την συσκευή που επιθυμούμε να ελέγξουμε και στο τρίτο ριπ εισάγουμε την γείωση.

Υπάρχουν 4 διαθέσιμοι ρελέ, από τους οποίους χρησιμοποιήσαμε τον 1. Ο λόγος που χρησιμοποιήσαμε τον ρελέ ήταν ότι επιθυμούσαμε να εισάγουμε έναν έλεγχο στον οποίο θα ορίζαμε ότι, εάν η θερμοκρασία του δωματίου ήταν πάνω από ένα συγκεκριμένο όριο, τότε θα ενεργοποιούταν το συγκεκριμένο ρελε.

Τα ρελέ που υπάρχουν ενεργοποιούνται όποτε αποστέλλουμε το λογικό 0 σαν είσοδο στο εκάστοτε ριπ του.



Εικόνα 18 4κάναλος Ρελες

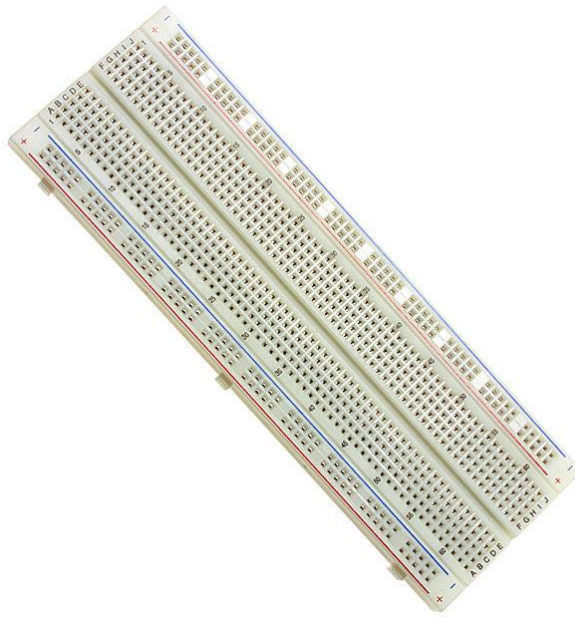
Breadboard

Ένα breadboard είναι μια βάση κατασκευής για την παραγωγή πρωτότυπων ηλεκτρονικών.

Το breadboard το έχουμε κολλήσει με ταινία διπλής όψεως πάνω στην μακέτα.

Μας διευκόλυνε πολύ έτσι ώστε να μπορέσουμε να αραιώσουμε τα καλώδια για να μην υπάρχει οποιοσδήποτε ήχος ή μαγνητική μετάδοση ανάμεσα στα καλώδια που χρησιμοποιήσαμε καθώς δεν είχαν ασφάλεια πάνω στα συγκεκριμένα προβλήματα.

Με παράλληλες συνδέσεις και διαμοιρασμό ρεύματος πάνω στην βάση πέτυχαμε την επέκταση των δυνατοτήτων της πλακέτας με επικοινωνία σε εισόδους και εξόδους από περισσότερα hardware υλικά.



Εικόνα 19 Breadboard

Καλώδια

Κατασκευασμένα για να διευκολύνουν τις δημιουργίες πάνω σε Arduino.

Απλά καλώδια για να πραγματοποιήσουμε μεταφορά εξόδων – εισόδων καθώς και την επέκταση των καλωδίων των αισθητήρων για να μπορέσουμε να κατασκευάσουμε την μακέτα μας.



Εικόνα 20 Καλώδια

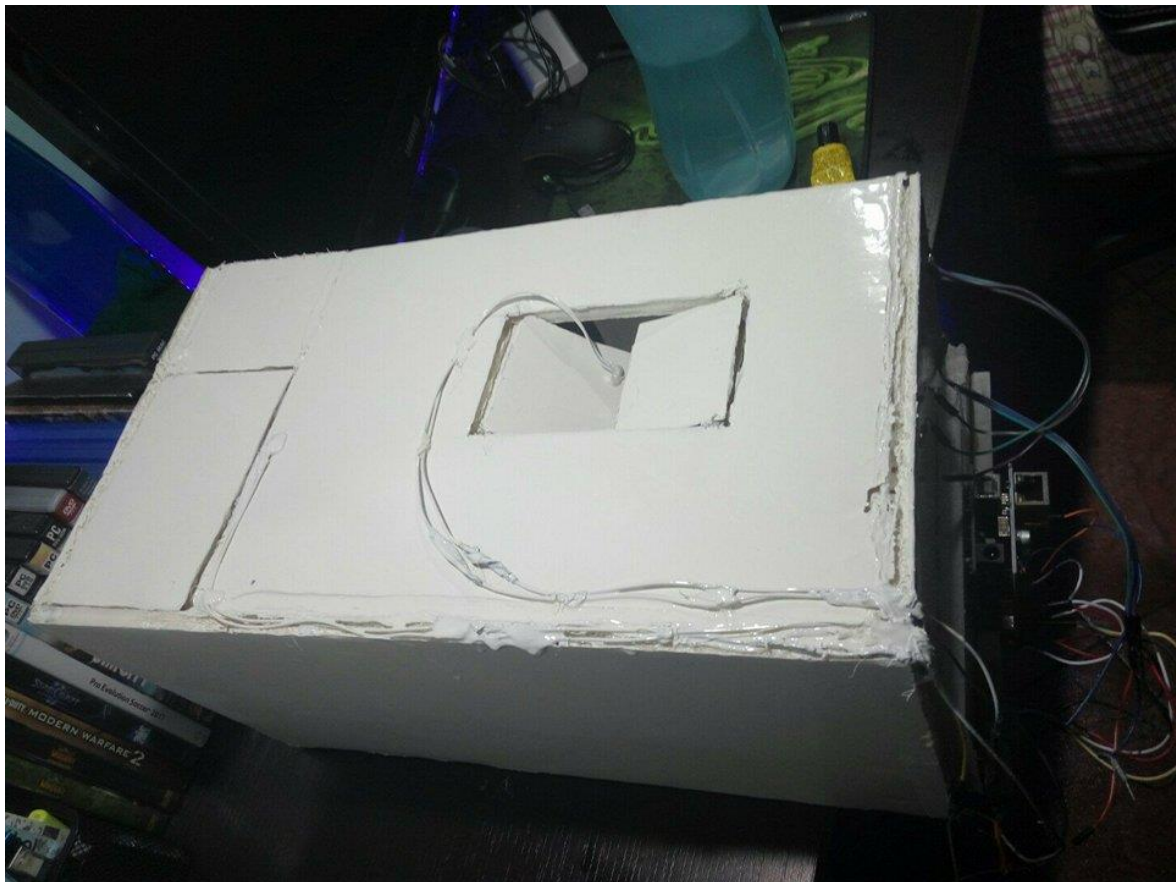
Μακέτα

Για να μπορέσουμε να αναπαραστήσουμε αλλά και να ελέγξουμε την λειτουργικότητα του κώδικα μας αλλά και τον αισθητήρων μας, δημιουργήσαμε μία μακέτα αναπαράστασης μίας αίθουσας με ένα δωμάτιο, ένα παράθυρο και μία πόρτα.

Η μακέτα δημιουργήθηκε από φύλλα συμπιεσμένου φελιζόλ τα οποία κολληθήκαν μεταξύ τους με σιλικόνη.

Τα αισθητήρια κολλήθηκαν με ταινίες διπλής όψεως πάνω στα τοιχώματα του φελιζόλ.

Όπως φαίνεται στην εικόνα, υπάρχουν στη δεξιά πλευρά οι πλακέτες μας τις οποίες κολλήσαμε με ταινίες διπλής όψεως σε ένα επιπρόσθετο φύλλο φελιζόλ.



Εικόνα 21 Η μακέτα μας

Υπηρεσίες – Προγράμματα

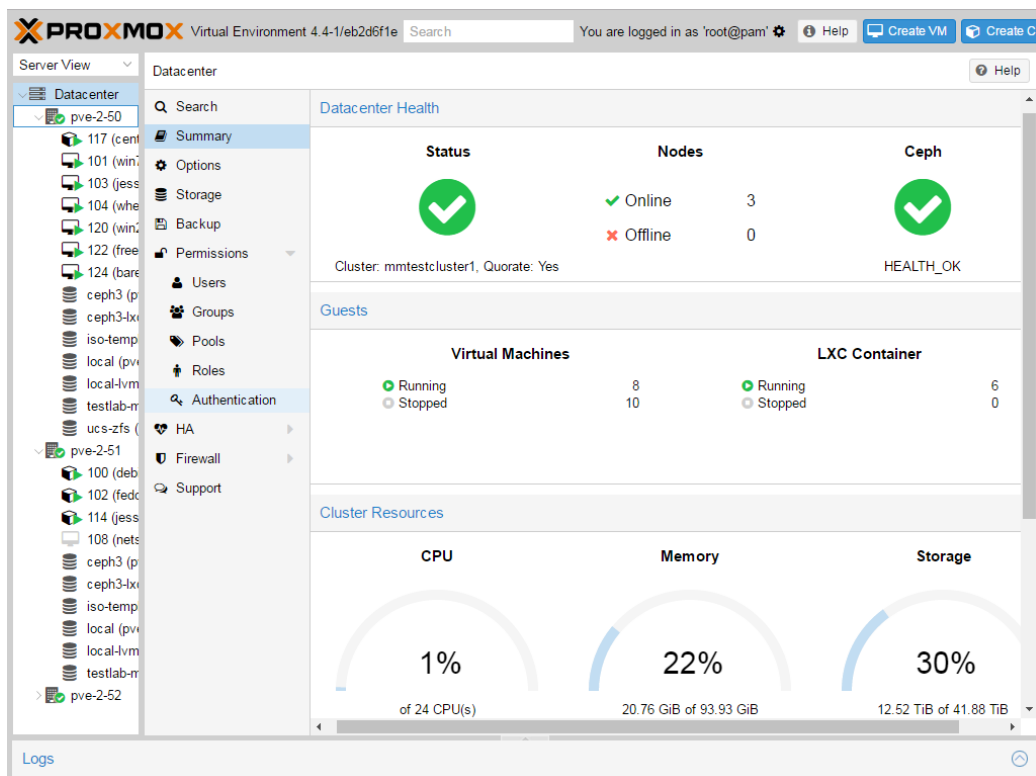
Εικονικό μηχάνημα server σε υποδομή Cloud

Χρησιμοποιήσαμε εικονικό μηχάνημα σε υποδομή cloud , το οποίο χορηγήθηκε δωρεάν από την εταιρία IpHost (<https://www.iphost.net/>).



Εικόνα 22 IpHost

Για να δημιουργήσουμε το εικονικό μηχάνημα , χρειάστηκε να χρησιμοποιήσουμε την υποδομή Cloud της IpHost. Η οποία είναι βασισμένη πάνω σε virtualization Proxmox. Το Proxmox είναι ένα περιβάλλον virtualization server ανοιχτού κώδικα. Πρόκειται για μία διανομή Linux που βασίζεται πάνω σε Debian με έναν τροποποιημένο πυρήνα - kernel του Ubuntu LTS.

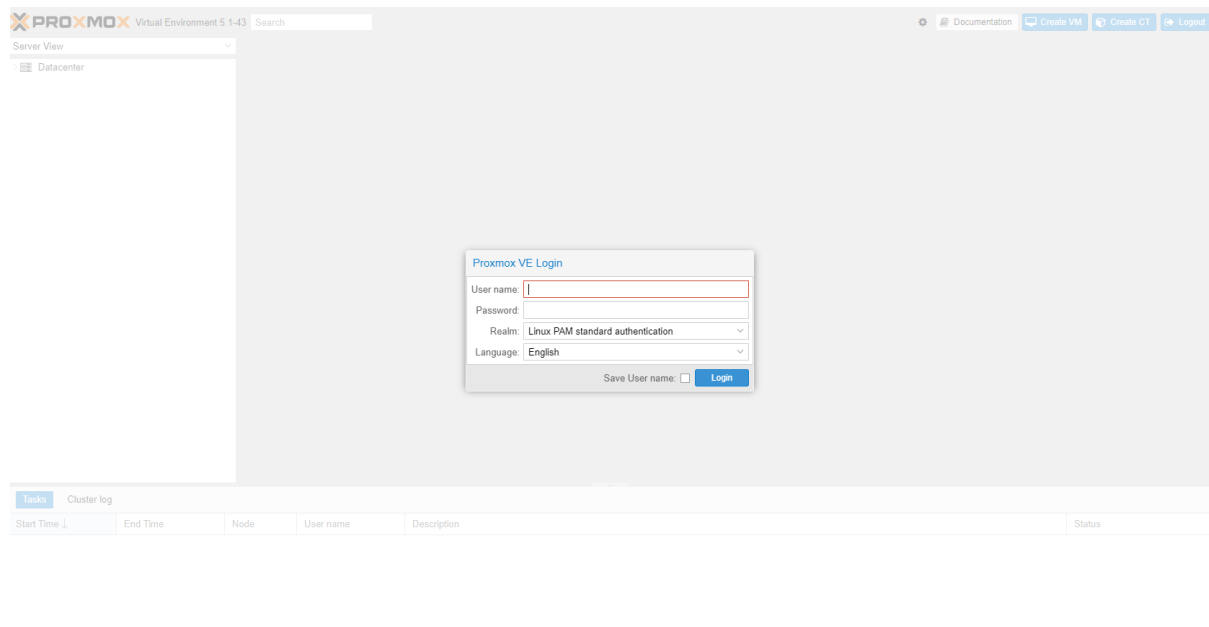


Εικόνα 23 Proxmox γραφικό περιβάλλον

Ο kernel αποτελεί το πιο χαμηλό επίπεδο αφαίρεσης υλικού (επεξεργαστών, της μνήμης και των μονάδων Εισόδου/Εξόδου) και είναι υπεύθυνος για τη διαχείριση των

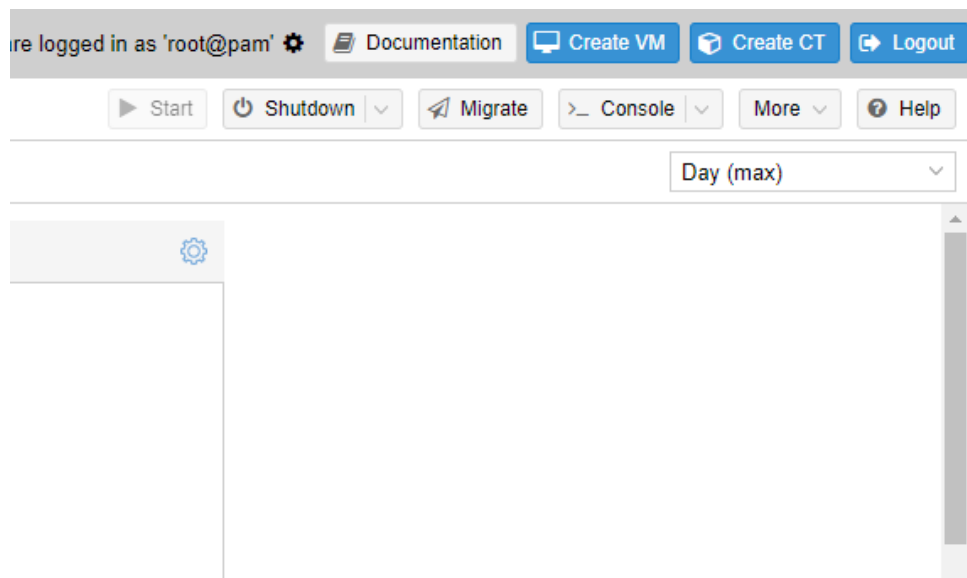
πόρων του συστήματος. Οι εφαρμογές μπορούν μέσω του kernel να μιλήσουν για τα διάφορα στοιχεία του υλικού στο σύστημα.

Τα βήματα που ακολουθήσαμε για να δημιουργήσουμε το μηχάνημα είναι τα παρακάτω. Αρχικά συνδεθήκαμε στο διαχειριστικό panel του Proxmox όπως φαίνεται παρακάτω :



Εικόνα 24 Σύνδεση στο proxmox

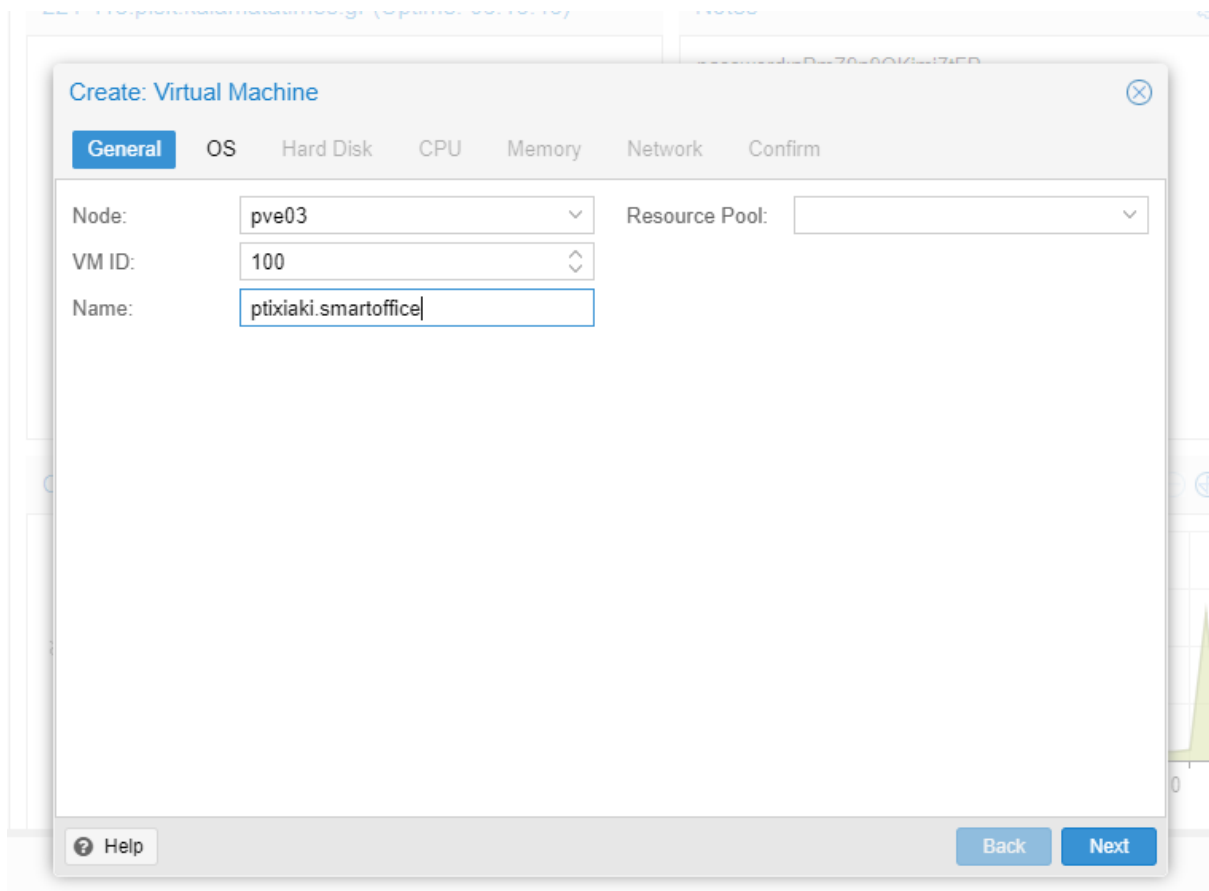
Έπειτα επελέξαμε το κουμπί Create VM, για να ξεκινήσουμε την διαδικασία της δημιουργίας του εικονικού μηχανήματος :



Εικόνα 25 Εκκίνηση διαδικασίας δημιουργίας εικονικού μηχανήματος

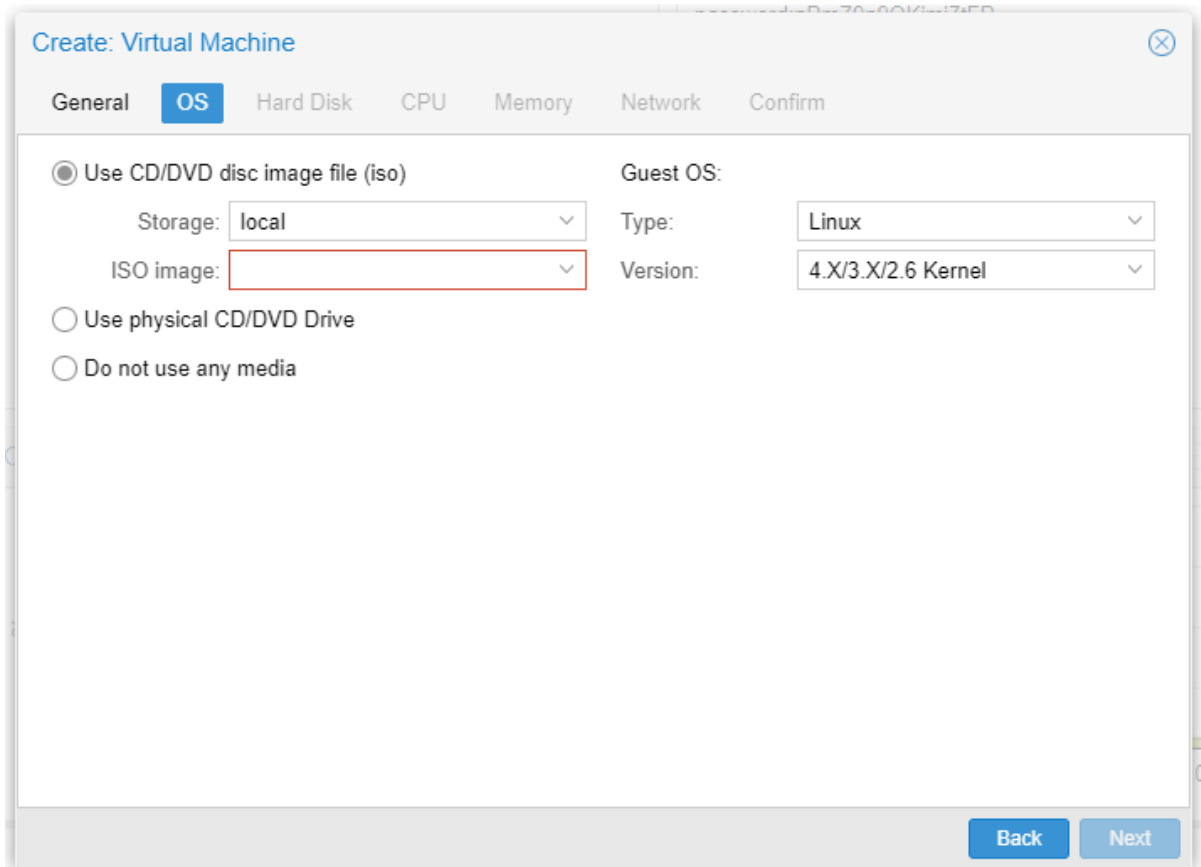
Έπειτα επιλέξαμε Node. Το node είναι ο hypervisor στον οποίο θα δημιουργηθεί το εικονικό μηχάνημα. Hypervisor είναι ένα φυσικό μηχάνημα – server το οποίο έχει συνδεθεί στην υποδομή Cloud - Proxmox της IpHost έτσι ώστε να προσφέρει τους διαθέσιμους πυρήνες του και την μνήμη RAM του μηχανήματος. Επιπλέον, εισάγαμε

τον μοναδικό αριθμό ID του μηχανήματος , και την ονομασία του για να ξεχωρίζει από τα υπόλοιπα εικονικά μηχανήματα.



Εικόνα 26 Δημιουργία Εικονικού μηχανήματος βήμα 1

Πατώντας Next, πήγαμε στο επόμενο βήμα στο οποίο επιλέγουμε το cd με το λειτουργικό που θέλουμε να κάνουμε εγκατάσταση στο μηχάνημα καθώς και το είδος του λειτουργικού. Για την παρούσα εργασία επιλέχθηκε το Linux και με εγκατάσταση του Centos 7. Στο proxmox επιλέξαμε Τύπο: Linux και έκδοση kernel 4.x καθώς η συγκεκριμένη είναι συμβατή για εικονικά μηχανήματα με Centos 7, σύμφωνα με τις οδηγίες του proxmox.



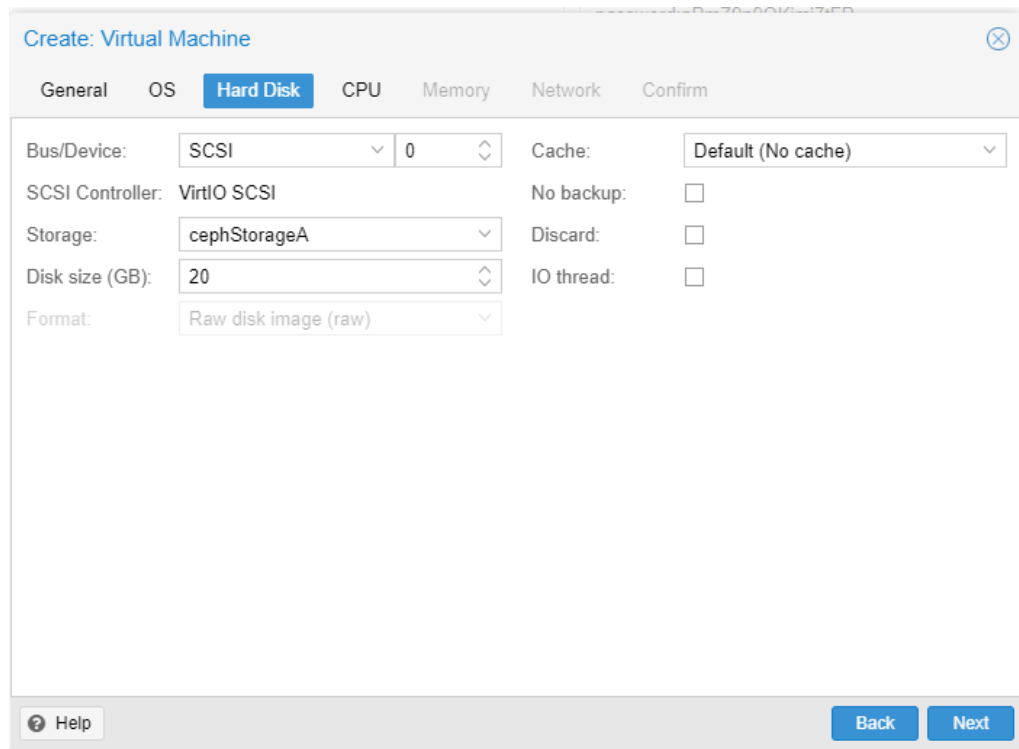
Εικόνα 27 Δημιουργία Εικονικού μηχανήματος βήμα 2

Συνεχίζοντας στο παρακάτω βήμα, επιλέγουμε το είδος του αποθηκευτικού χώρου που θα χρησιμοποιήσουμε καθώς και το μέγεθος. Για την παρούσα εργασία επιλέξαμε αποθηκευτικό χώρο τύπου Ceph και χρησιμοποιήσαμε ένα κομμάτι 20Gb δίσκου.

Ο αποθηκευτικός χώρος τύπου Ceph είναι μία πλατφόρμα (ελεύθερου κώδικα) αποθήκευσης που υλοποιεί την αποθήκευση αντικειμένων σε ένα ενιαίο κατανομημένο σύμπλεγμα δίσκων και παρέχει διεπαφές για αποθήκευση των αντικειμένων, blocks και αρχείων.

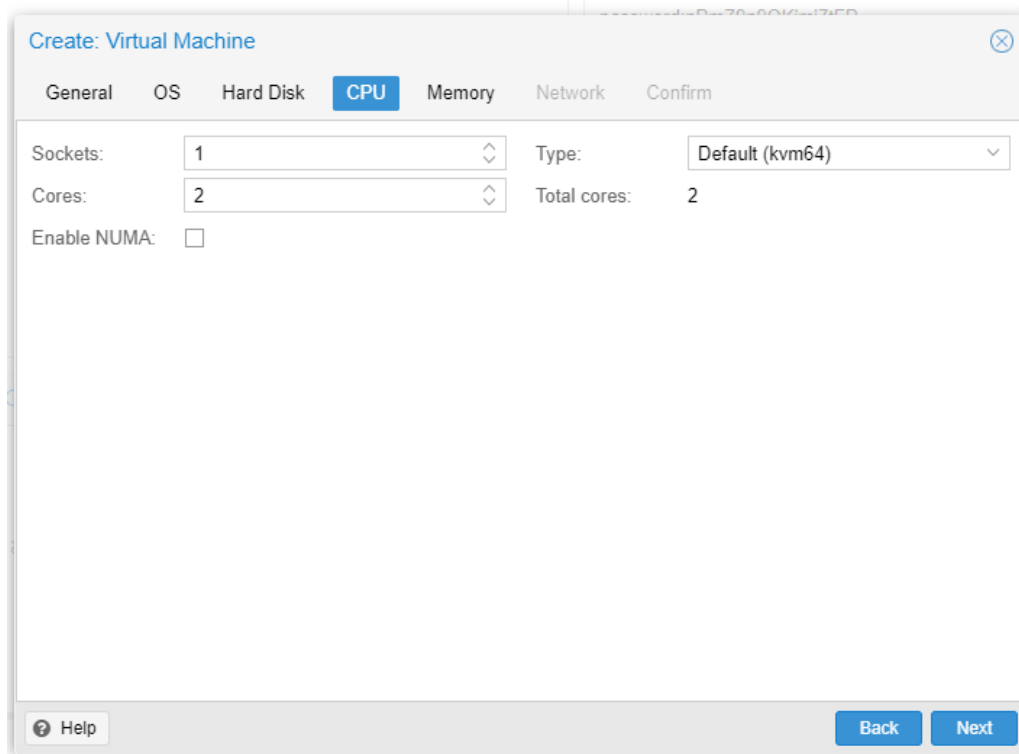
Το Ceph εστιάζει κυρίως στην πλήρη κατανομή της λειτουργίας της Cloud υποδομής χωρίς ένα μοναδικό σημείο αποτυχίας.

Το Ceph αναπαράγει τα δεδομένα του σε πολλούς δίσκους με αποτέλεσμα να το καθιστά ανεκτικό σε σφάλματα χρησιμοποιώντας το δικό του υλικό και δεν απαιτεί ειδική τεχνική υποστήριξη για την διόρθωση των σφαλμάτων.



Εικόνα 28 Δημιουργία Εικονικού μηχανήματος βήμα 3

Συνεχίζοντας παρακάτω, επιλέξαμε το πλήθος των διαθέσιμων πυρήνων που θα έχει το εικονικό μας μηχάνημα. Επιλέξαμε να έχει 2 πυρήνες από οποιοδήποτε socket του μηχανήματος.



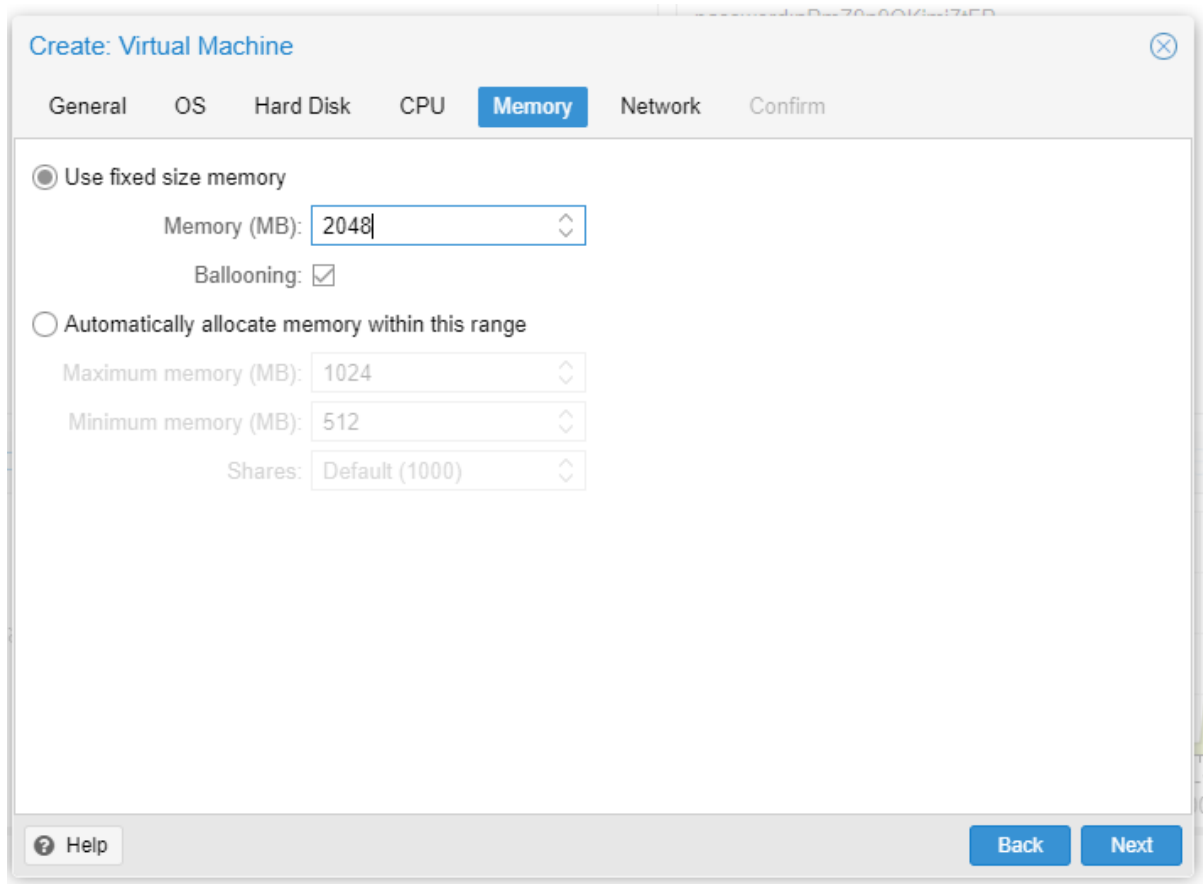
Εικόνα 29 Δημιουργία Εικονικού μηχανήματος βήμα 4

Socket είναι η φυσική μορφή ενός επεξεργαστή όπως φαίνεται παρακάτω:



Εικόνα 30 Cpu socket

Έπειτα προχωρήσαμε στο επόμενο βήμα που είναι η επιλογή της διαθέσιμης μνήμης RAM για το μηχάνημα μας. Επιλέξαμε 2048MB καθώς δεν χρειαζόμασταν περισσότερα σύμφωνα με τις απαιτήσεις του node-red:



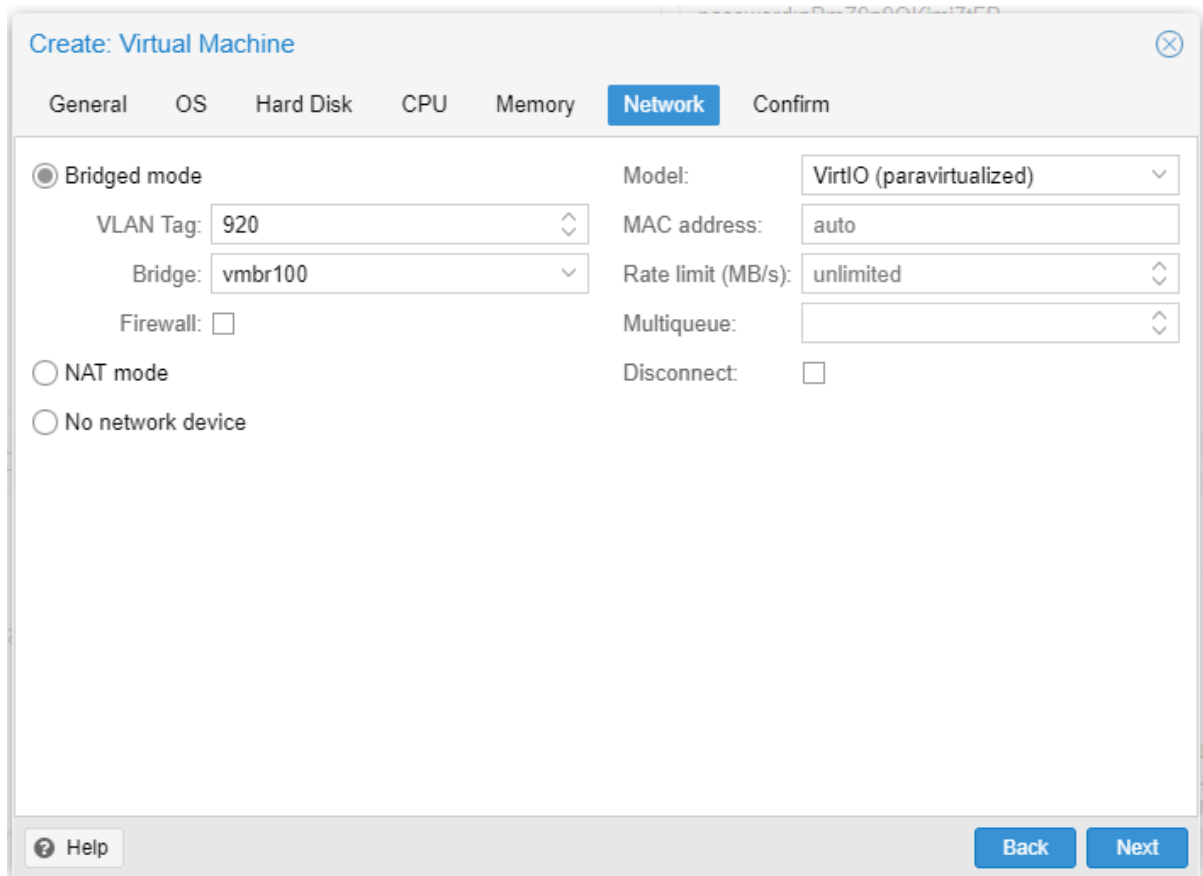
Εικόνα 31 Δημιουργία Εικονικού μηχανήματος βήμα 5

Στη συνέχεια, προχωρήσαμε στη ρύθμιση του δικτύου του μηχανήματος μας.

Έπρεπε να ρυθμίσουμε το vlan σύμφωνα με την Cloud υποδομή της IpHost. Πιο συγκεκριμένα, για να μπορέσουμε να χρησιμοποιήσουμε την ip που μας ανατέθηκε χρειάστηκε να δηλώσουμε το vlan 920.

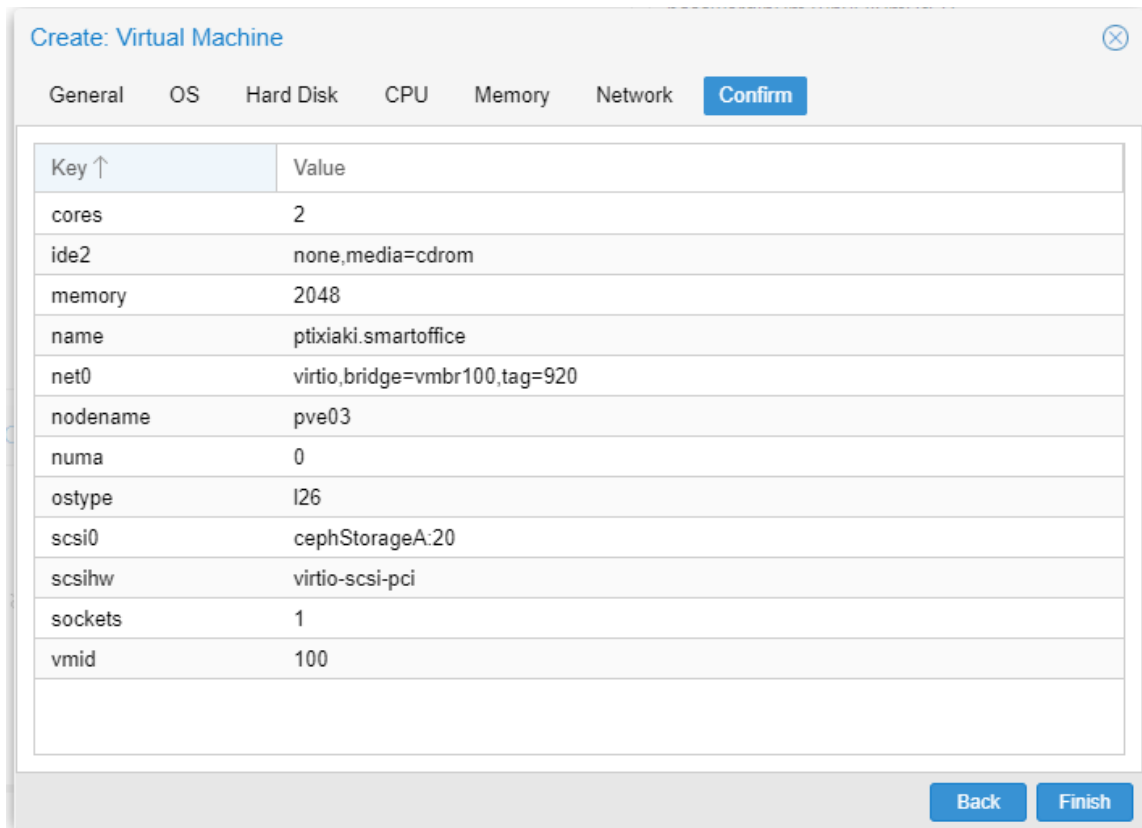
Το vlan σημαίνει virtual lan και ουσιαστικά είναι η τμηματοποίηση (σε επίπεδο IP) ενός τοπικού δικτύου ώστε ο χρήστης ενός τμήματος να μην έχει πρόσβαση σε άλλο τμήμα του δικτύου.

Μετά επιλέξαμε το bridge για να ολοκληρώσουμε την ρύθμιση του δικτύου. Μια γέφυρα δικτύου είναι μια συσκευή δικτύωσης που δημιουργεί ένα ενιαίο συγκεντρωτικό δίκτυο από πολλά δίκτυα επικοινωνιών ή τμήματα δικτύου.



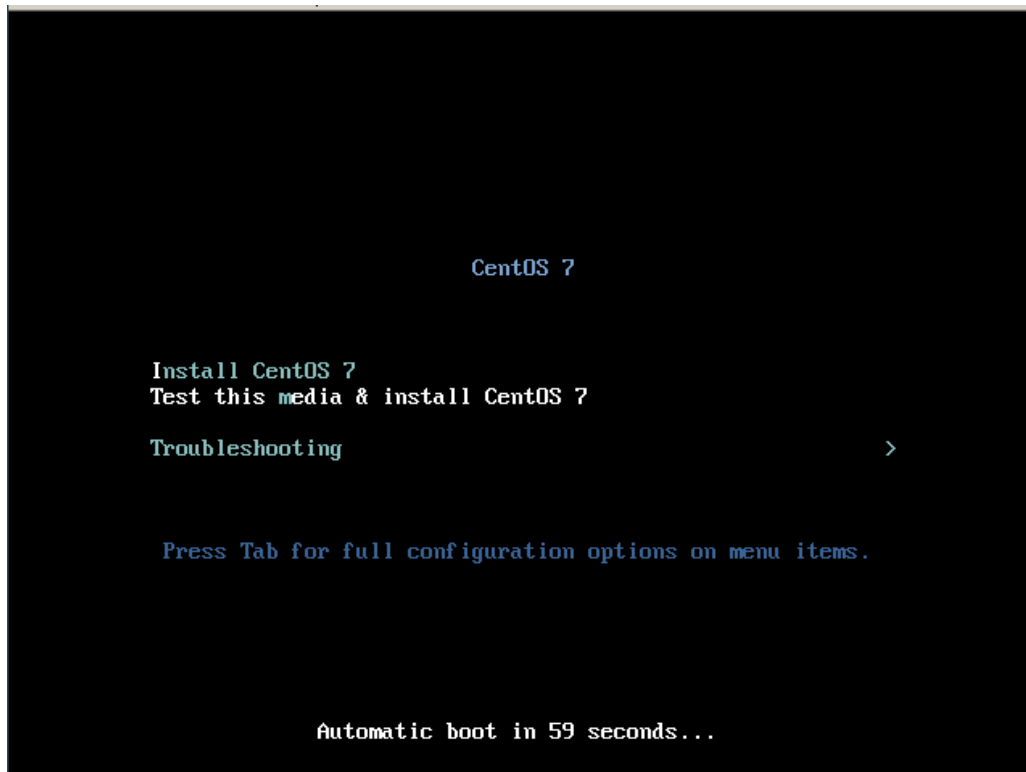
Εικόνα 32 Δημιουργία Εικονικού μηχανήματος βήμα 6

Τέλος πατήσαμε το κουμπί "Finish" για να δημιουργηθεί το μηχανήμα.



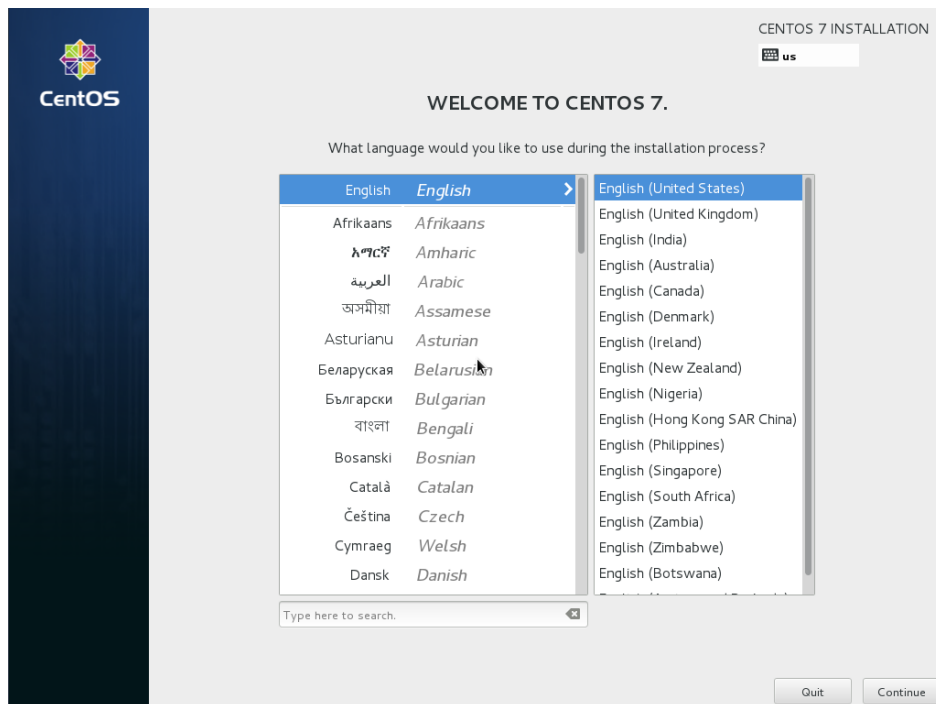
Εικόνα 33 Δημιουργία Εικονικού μηχανήματος βήμα 7

Έπειτα εκκινήσαμε το μηχάνημα και αφού κάναμε boot από το εικονικό cd (iso) που είχαμε εισάγει, επιλέξαμε να εγκαταστήσουμε το centos 7.



Εικόνα 34 Εγκατάσταση Centos 7 βήμα 1

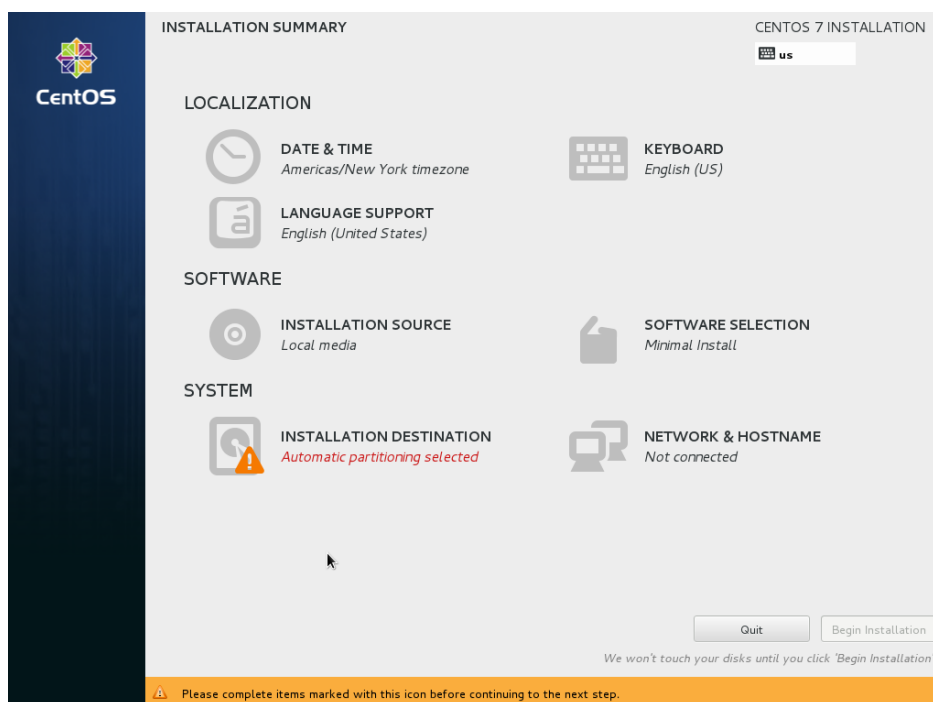
Έπειτα επιλέξαμε γλώσσα εγκατάστασης , τα αγγλικά για διευκόλυνση μας.



Εικόνα 35 Εγκατάσταση Centos 7 βήμα 2

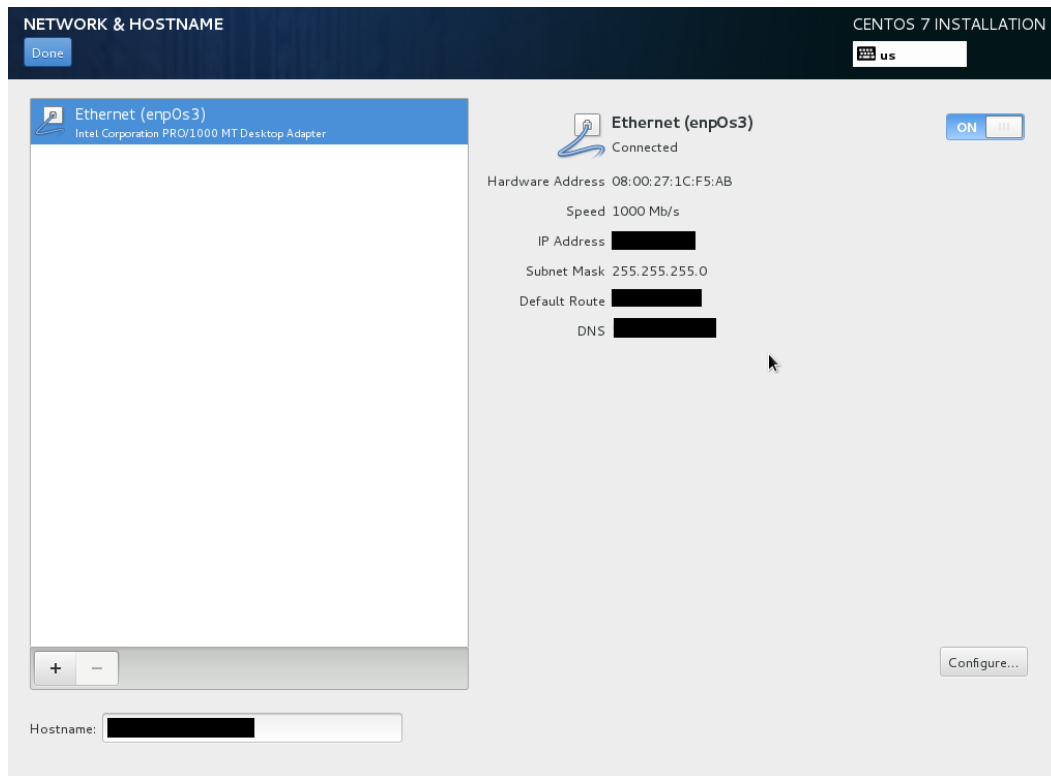
Έπειτα μας εμφανίστηκε ένα γραφικό διάφορων επιλογών για ρυθμίσεις.

Από αυτές χρειάστηκε να ρυθμίσουμε το Installation Destination , Date & Time, Network & Hostname



Εικόνα 36 Εγκατάσταση Centos 7 βήμα 3

Για να ρυθμίσουμε το δίκτυο, σύμφωνα με υποδείξεις της IpHost σχετικά με την ip που θα χρησιμοποιήσουμε, προχωρήσαμε στην αντίστοιχη επιλογή :



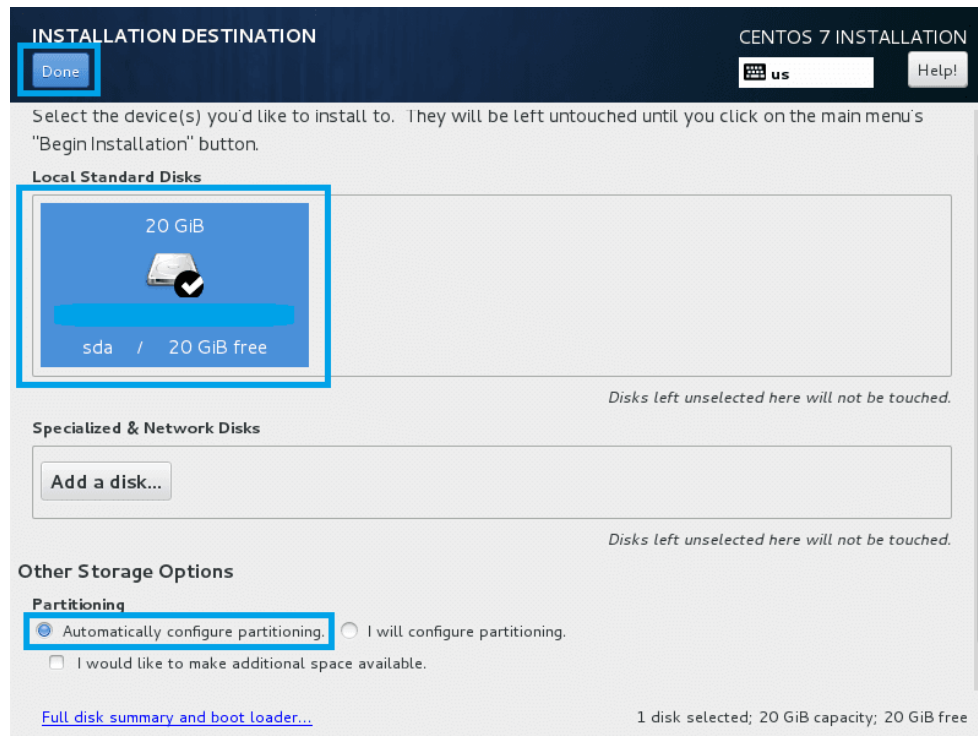
Εικόνα 37 Εγκατάσταση Centos 7 βήμα 4

Έπειτα ρυθμίσαμε την ώρα, πήγαμε στην αντίστοιχη επιλογή και επιλέξαμε Region : Europe , City: Athens



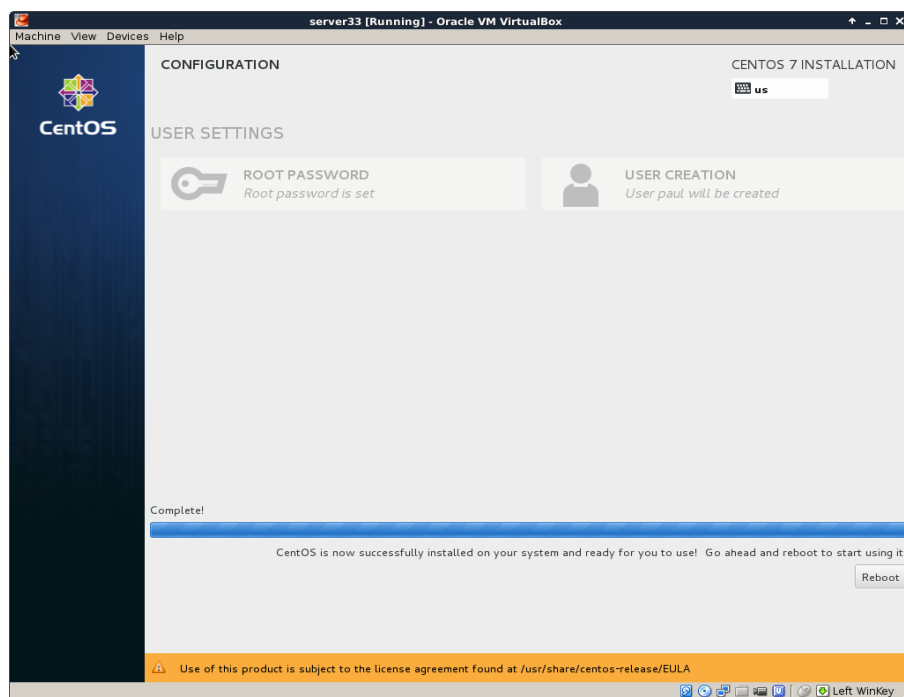
Εικόνα 38 Εγκατάσταση Centos 7 βήμα 5

Έπειτα επιλέξαμε τον δίσκο και του ρυθμίσαμε να φτιάξει αυτόματα τα απαραίτητα κομμάτια δίσκου που απαιτούνται για να μπορέσει το Centos να λειτουργήσει ομαλά.



Εικόνα 39 Εγκατάσταση Centos 7 βήμα 6

Έπειτα πατήσαμε next και πήγαμε στο τελικό βήμα εγκατάστασης όπου ρυθμίσαμε και τον κωδικό για τον χρήστη root του συστήματος.



Εικόνα 40 Εγκατάσταση Centos 7 βήμα 7

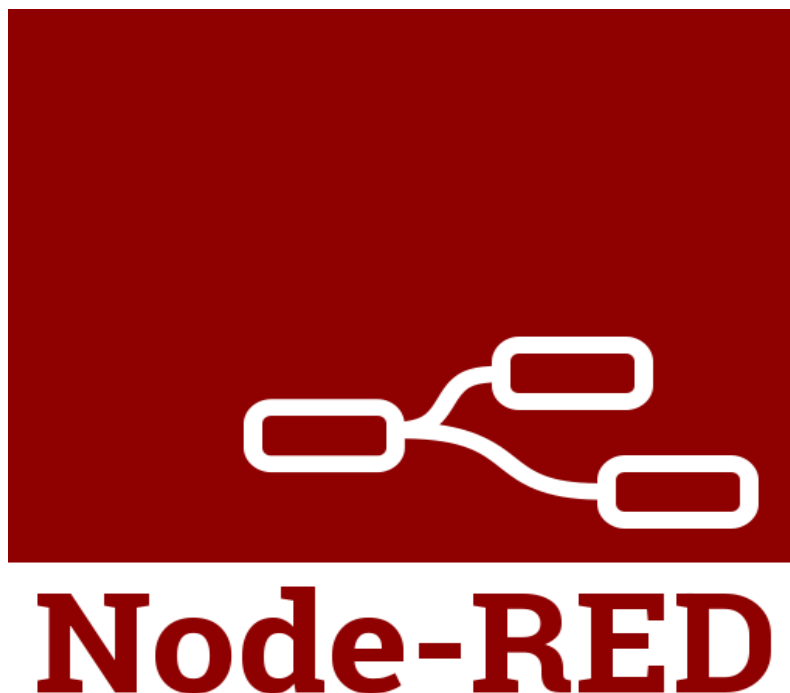
Τέλος έπειτα από μία επιτυχημένη επανεκκίνηση του μηχανήματος, λάβαμε το τερματικό της οθόνης που μας ζητάει τα στοιχεία σύνδεσης :

```
CentOS Linux 7 (Core)  
Kernel 3.10.0-123.el7.x86_64 on an x86_64  
  
localhost login: _
```

Εικόνα 41 Εγκατάσταση Centos 7 βήμα 8

Node-red

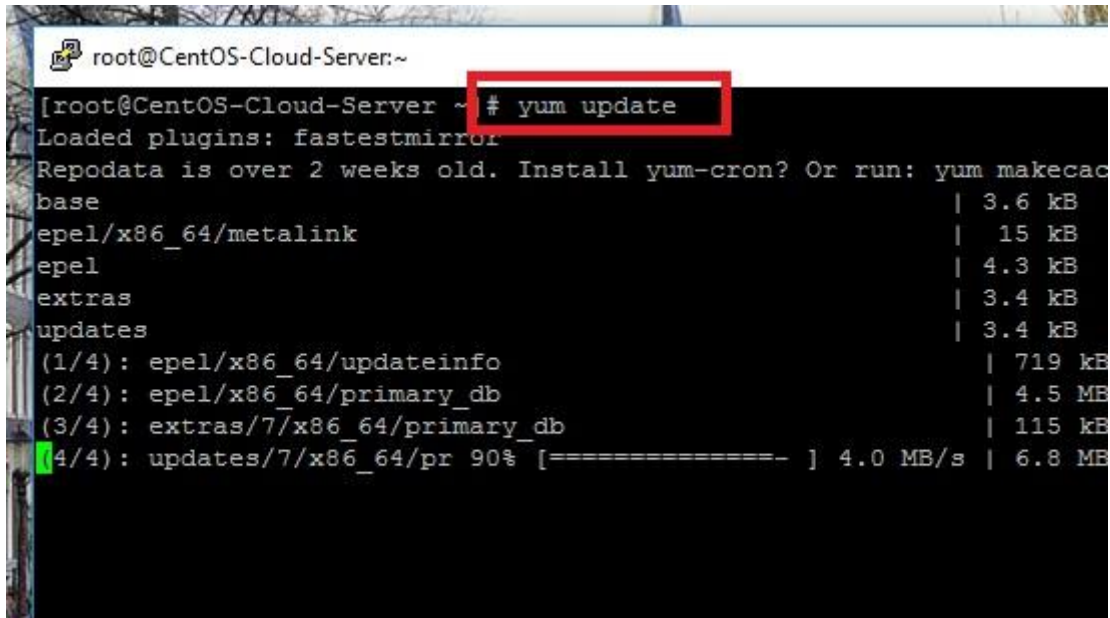
Στο εικονικό μας μηχάνημα πραγματοποιήσαμε εγκατάσταση node-red καθώς είναι ένα από τα προαπαιτούμενα για να μπορέσουμε να ολοκληρώσουμε την εργασία μας.



Εικόνα 42 Node-red

Για να μπορέσουμε να πραγματοποιήσουμε την συγκεκριμένη εγκατάσταση, αρχικά χρειάστηκε να συνδεθούμε ως χρήστης root και να εκτελέσουμε ενημέρωση του συστήματος και των ήδη εγκατεστημένων βιβλιοθηκών με την εντολή «yum update». Το yum update, προέρχεται από την Fedora η οποία παρέχει λογισμικό και τεκμηρίωση μέσω κάποιων ειδικών αρχείων που ονομάζονται RPM packages. Κάθε

RPM package, είναι ένα συμπιεσμένο πρόγραμμα που παρέχει πληροφορίες σχετικές με βιβλιοθήκες. Συνεπώς, εμείς τρέχοντας το yum update του ζητάμε να ελέγξει όλα τα τελευταία rpm πακέτα εάν είναι στην ίδια έκδοση με τα πακέτα του server ώστε, εάν δεν είναι, να εγκαταστήσει τα νέα. Το RPM (RPM Manager Package) είναι ένα δημοφιλές βοηθητικό πρόγραμμα για την εγκατάσταση λογισμικού σε συστήματα που μοιάζουν με Unix, ειδικά το Red Hat Linux.



```
root@CentOS-Cloud-Server:~  
[root@CentOS-Cloud-Server ~]# yum update  
Loaded plugins: fastestmirror  
Repodata is over 2 weeks old. Install yum-cron? Or run: yum makeca  
base | 3.6 kB  
epel/x86_64/metalink | 15 kB  
epel | 4.3 kB  
extras | 3.4 kB  
updates | 3.4 kB  
(1/4): epel/x86_64/updateinfo | 719 kB  
(2/4): epel/x86_64/primary_db | 4.5 MB  
(3/4): extras/7/x86_64/primary_db | 115 kB  
(4/4): updates/7/x86_64/pr 90% [===== ] 4.0 MB/s | 6.8 MB
```

Εικόνα 43 Centos 7 yum update

Μέσα στο οποίο εγκαταστήσαμε την βιβλιοθήκη του freeboard :



Εικόνα 44 freeboard

Το freeboard είναι μία πλατφόρμα εξυπηρέτησης των project που έχουν βασιστεί στο Internet of Things. Πρόκειται για μία πλατφόρμα εύκολη στη χρήση και στην απεικόνιση των δεδομένων. Είναι ανοιχτού κώδικα και πολλοί προγραμματιστές λαμβάνουν δράση για να δημιουργήσουν παραπάνω δυνατότητες για την συγκεκριμένη πλατφόρμα.

Είναι σημαντικό να αναφερθεί ότι, ενώ αρχικά η πλατφόρμα ήταν δωρεάν, κατά την διάρκεια ανάπτυξης της παρούσας πτυχιακής εργασίας δημοσιεύτηκε ενημέρωση ότι οι δωρεάν εκδόσεις θα σταματήσουν να υπάρχουν. Οπότε αναζητήσαμε και βρήκαμε ότι στο node-red μπορούσαμε να εγκαταστήσουμε τις βιβλιοθήκες του freeboard και να προχωρήσουμε κανονικά στην υλοποίηση της εργασίας μας.

Dweet.io

Μέσα από το οποίο πραγματοποιήσαμε αποστολή και παραλαβή δεδομένων από το Arduino προς το dweet.io και από το dweet.io στο node-red.

Το dweet.io χρησιμοποιεί ένα αρκετά απλό πρωτόκολλο TCP με εντολές συμβολοσειρών. Η συσκευή στέλνει ένα μήνυμα GET στον διακομιστή που απαντά με ένα μήνυμα POST που περιέχει τις εντολές και τα δεδομένα που επιθυμούμε να αποστείλουμε στην συγκεκριμένη υπηρεσία.



Εικόνα 45 dweet.io

Είναι μία υπηρεσία στην οποία διάφοροι χρήστες κάνουν dweet για συγκεκριμένα στοιχεία. Σε αυτή την περίπτωση για να πραγματοποιήσεις ένα ανέβασμα δεδομένων στο dweet , αρκεί να αποστείλεις ένα POST request, όπως φαίνεται στο παρακάτω παράδειγμα:

POST <http://dweet.io/dweet/quietly/for/room-a116?temp=10&hum=45&sensor=1&window=0&door=1>

Και σε περίπτωση που θέλουμε να τραβήξουμε δεδομένα για το element room-a116 θα χρειαστεί να κάνουμε ένα GET request :

GET <http://dweet.io/get/latest/dweet/for/room-a116>

Περισσότερα θα εξηγήσουμε στην επόμενη υποενότητα.

Arduino IDE

Ένα πρόγραμμα ανοιχτού πηγαίου κώδικα το οποίο μας αφήνει εύκολα να προγραμματίσουμε κώδικα και να το ανεβάσουμε στο Arduino μας.

Το περιβάλλον έχει γραφτεί σε java καθώς και σε άλλα προγράμματα ανοιχτού κώδικα. Περιέχει ένα πρόγραμμα επεξεργασίας κειμένου για την εγγραφή κώδικα, μια περιοχή μηνυμάτων, μια κονσόλα κειμένου, μια γραμμή εργαλείων με κουμπιά για κοινές λειτουργίες και μια σειρά από μενού. Συνδέεται με το Arduino μέσω υπολογιστή για τη μεταφόρτωση προγραμμάτων και την επικοινωνία μαζί τους.



Εικόνα 46 Arduino IDE

Συνδεσμολογία και εφαρμογή

Εφόσον πραγματοποιήσαμε όλες τις προαναφερόμενες εργασίες – ενέργειες.

Ξεκινήσαμε να φτιάχνουμε την εργασία για να επιτύχουμε τον σκοπό μας.

Αρχικά έπρεπε να φτιάξουμε το arduino έτσι ώστε να λαμβάνει τα δεδομένα και να πραγματοποιεί την επιθυμητή επεξεργασία δεδομένων.

Οπότε συνδέσαμε όλους τους αισθητήρες με το arduino:

- 2 Μαγνητικές επαφές
- 1 Αισθητήρα Κίνησης
- 1 Αισθητήρα Θερμοκρασίας – Υγρασίας

Και έπειτα εισάγαμε και τον Ρελέ που θέλαμε όταν η θερμοκρασία να ανεβαίνει άνω των 35 να ενεργοποιείται μία συγκεκριμένη έξοδο , στην οποία στις κατάλληλες προϋποθέσεις , θα είναι συνδεδεμένο ένα κλιματιστικό.

Έπειτα από αρκετές δοκιμές , και επεξεργασία κώδικα στο arduino ide , καταλήξαμε στον κατάλληλο κώδικα , που φαίνεται παρακάτω:

```
//Εδώ εισάγουμε τις απαραίτητες βιβλιοθήκες που χρειαζόμαστε στον κώδικα  
#include <dht.h>
```

Εικόνα 47 Βιβλιοθήκη dht στο Arduino Ide

Η συγκεκριμένη βιβλιοθήκη είναι για τον αισθητήρα θερμοκρασίας – υγρασίας , καθώς ήταν ο μόνος αισθητήρας που χρειαστήκαμε βιβλιοθήκη μέσα στο arduino ide έτσι ώστε να μπορέσουμε να διαβάσουμε τα δεδομένα του συγκεκριμένου αισθητήρα.

```
//Ορίζουμε τα pin που θα χρησιμοποιήσουμε  
dht DHT;  
#define switchPin 6  
#define switchPin2 7  
#define DHT11_PIN 8  
#define motion_1 9  
#define reles 2
```

Εικόνα 48 Ορισμός των pin που θα χρησιμοποιήσουμε

Έπειτα χρειάστηκε να ορίζουμε τις εισόδους και τις εξόδους που θα έχουμε στο arduino .

Στο pin 6 και 7 είναι οι μαγνητικές επαφές από τις οποίες το 6 είναι η μαγνητική επαφή που εισάγαμε στην πόρτα και το 7 είναι η μαγνητική επαφή που εισάγαμε στο παράθυρο.

Στο pin 8 έχουμε εγκαταστήσει τον αισθητήρα θερμοκρασίας – υγρασίας.

Στο pin 9 έχουμε συνδέσει τον αισθητήρα κίνησης

Στο pin 2 έχουμε συνδέσει το pin του ρελε που θέλουμε να ενεργοποιείται στις κατάλληλες προϋποθέσεις τις θερμοκρασίας.

```
//Εδώ ορίζουμε ποια pin είναι είσοδοι και ποια έξοδοι  
pinMode (motion_1, INPUT);  
pinMode (switchPin, INPUT_PULLUP);  
pinMode (switchPin2, INPUT_PULLUP);  
pinMode (reles, OUTPUT);  
//Ορίζουμε αρχικές τιμές στις εισόδους και στις εξόδους  
digitalWrite (switchPin, HIGH);  
digitalWrite (switchPin2, HIGH);  
digitalWrite (reles, HIGH);
```

Εικόνα 49 Διαχώρηση εισόδων και εξόδων.

Εδώ ορίζουμε τα pin του αισθητήρα κίνησης , τις μαγνητικές επαφές σαν εισόδους καθώς και τον ρελέ σαν έξοδο. Επίσης αρχικοποιούμε τις τιμές στο λογικό 1.

```
void loop()
{
  //Διαβάζουμε δεδομένα από τους αισθητήρες
  int sensor_1 = digitalRead(motion_1);
  int chk = DHT.read11(DHT11_PIN);
  temp = DHT.temperature;
  hum = DHT.humidity;
  magnet1=digitalRead(switchPin);
  magnet2=digitalRead(switchPin2);

  //Εάν η θερμοκρασία είναι παραπάνω από 34 , τότε ενεργοποιούμε το pin του ρελέ που του ορίσαμε στην αρχή
  if (temp > 34) {
    digitalWrite(reles, LOW);
    relestatus = 1;
  } else {
    digitalWrite(reles, HIGH);
    relestatus = 0;
  }
}
```

Εικόνα 50 Κύριος κορμός κώδικα

Παραπάνω έχουμε το κύριο κομμάτι του κώδικα στο οποίο διαβάζουμε τα δεδομένα από όλους τους αισθητήρες και ορίζουμε την έξοδο του ρελέ ανάλογα την θερμοκρασία που λαμβάνουμε από τον αντίστοιχο αισθητήρα.

Το μόνο που μας μένει , πλέον είναι να πραγματοποιήσουμε σύνδεση μέσω του Ethernet shield στο διαδίκτυο και έπειτα να ανεβάσουμε τα δεδομένα στο dweet.io

Αρχικά εισάγαμε την απαραίτητη βιβλιοθήκη για την σύνδεση του arduino στο διαδίκτυο μέσω του Ethernet shield.

```
#include <Ethernet.h>
```

Εικόνα 51 Εισαγωγή βιβλιοθήκης Ethernet

Έπειτα ορίσαμε την διεύθυνση mac του arduino και τις απαραίτητες ρυθμίσεις έτσι ώστε να μπορέσει να συνδεθεί στο διαδίκτυο και να αποστείλει δεδομένα στο dweet.io

```
//Πραγματοποιούμε τις απαραίτητες ρυθμίσεις για την σύνδεση στο διαδίκτυο μέσω του ethernet shield
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
char server[] = "dweet.io"; // name address for Google (using DNS)
IPAddress ip(192, 168, 1, 177);
EthernetClient client;
```

Εικόνα 52 Ρύθμισης για σύνδεση στο διαδίκτυο

Πραγματοποιούμε σύνδεση στο διαδίκτυο , σε περίπτωση που το arduino δεν μπορέσει να πάρει αυτόματα διεύθυνση ip τότε του έχουμε ορίσει να πάρει χειροκίνητα (φαίνεται στην προηγούμενη εικόνα).

Κατασκευή και εγκατάσταση Smart Office

```
//Πραγματοποιούμε σύνδεση στο διαδίκτυο
if (Ethernet.begin(mac) == 0) {
  Serial.println("Failed to configure Ethernet using DHCP");
  //Εάν αποτύχει να πάρει αυτόματα ip μέσω dhcp τότε προσπαθεί να πάρει χειροκίνητα την ip που του ορίσαμε παραπάνω
  Ethernet.begin(mac, ip);
}
//Περιμένουμε ένα δευτερόλεπτο για να επιβεβαιωθεί η σύνδεση
delay(1000);
```

Εικόνα 53 Σύνδεση στο διαδίκτυο

Έπειτα εφόσον έχουν ολοκληρωθεί όλα τα παραπάνω επιτυχώς. Ανοίγουμε επικοινωνία με το dweet.io , στέλνουμε τα δεδομένα , και έπειτα κλείνουμε την επικοινωνία.

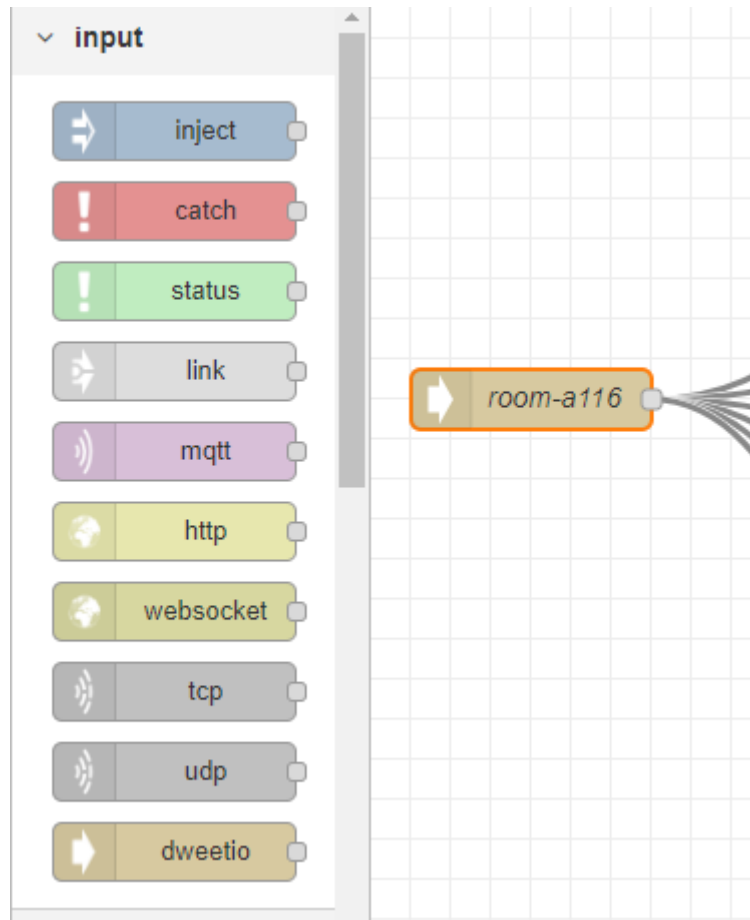
```
//Κάνουμε σύνδεση στο dweet.io
if(client.connect(server, 80)){
  //Στέλνουμε τα δεδομένα στο dweet.io
  client.println("POST /dweet/quietly/for/room-all6?temp="+String(temp)+"&hum="+String(hum)+"&sensor="+S
}
else {
  //Εάν αποτύχει την σύνδεση στο διαδίκτυο τότε εμφανίζουμε το αντίστοιχο μήνυμα στην κονσόλα
  Serial.println("Δεν μπόρεσα να κάνω dweet");
}

//Σε περίπτωση που έχουν ολοκληρωθεί όλα επιτυχώς , τότε κάνουμε αποσύνδεση για να κλείσουμε το session
if (!client.connected()) {
  Serial.println();
  Serial.println("disconnecting.");
}
client.stop();
```

Εικόνα 54 Αποστολή δεδομένων στο dweet.io

Τέλος μας έμεινε η απεικόνιση των δεδομένων μέσα στο node-red.

Ξεκινήσαμε σύροντας τα στοιχεία από την αριστερή στήλη στον κεντρικό πίνακα διαμόρφωσης του node-red.



Εικόνα 55 Εισαγωγή στοιχείων στο node-red

Εισάγαμε ένα στοιχείο εισόδου “dweetio” μέσα στο οποίο τροποποιήσαμε έτσι ώστε να τραβάει δεδομένα από το dweet.io για το αντικείμενο room-a116, όπως φαίνεται στην επόμενη εικόνα.

Edit dweetio in node

Delete Cancel Done

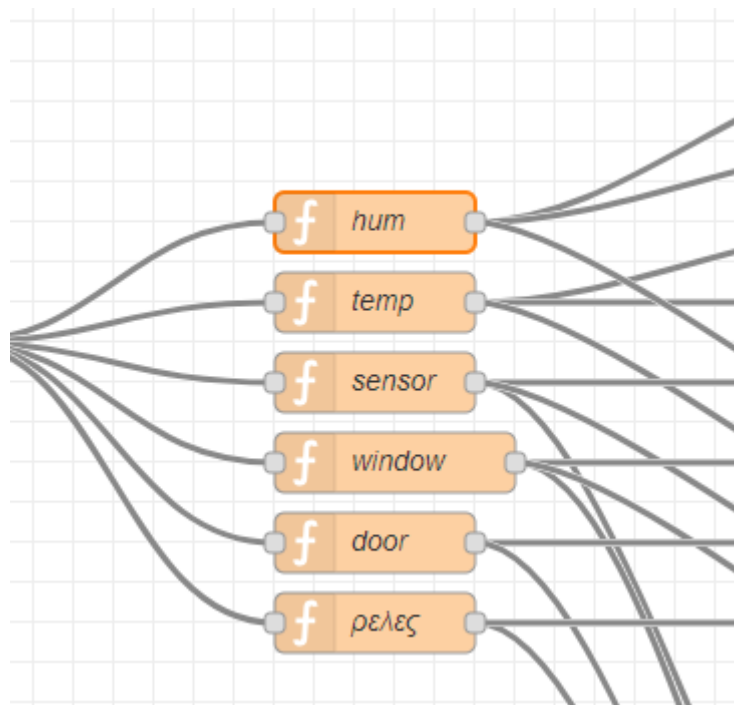
node properties

Thing

Name

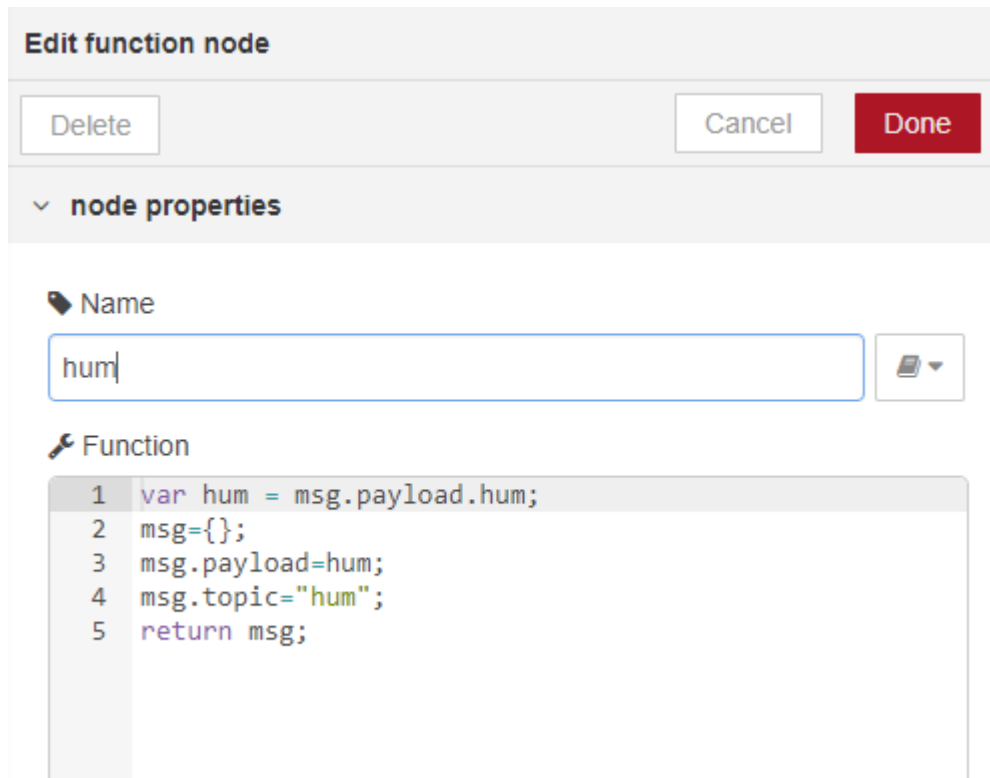
Εικόνα 56 Τροποποίηση εισόδου από το dweet.io

Στη συνέχεια δημιουργήσαμε συναρτήσεις για να μπορέσουμε να ξεχωρίσουμε τα δεδομένα από το εύρος εισόδου που λαμβάνουμε από το dweet.io



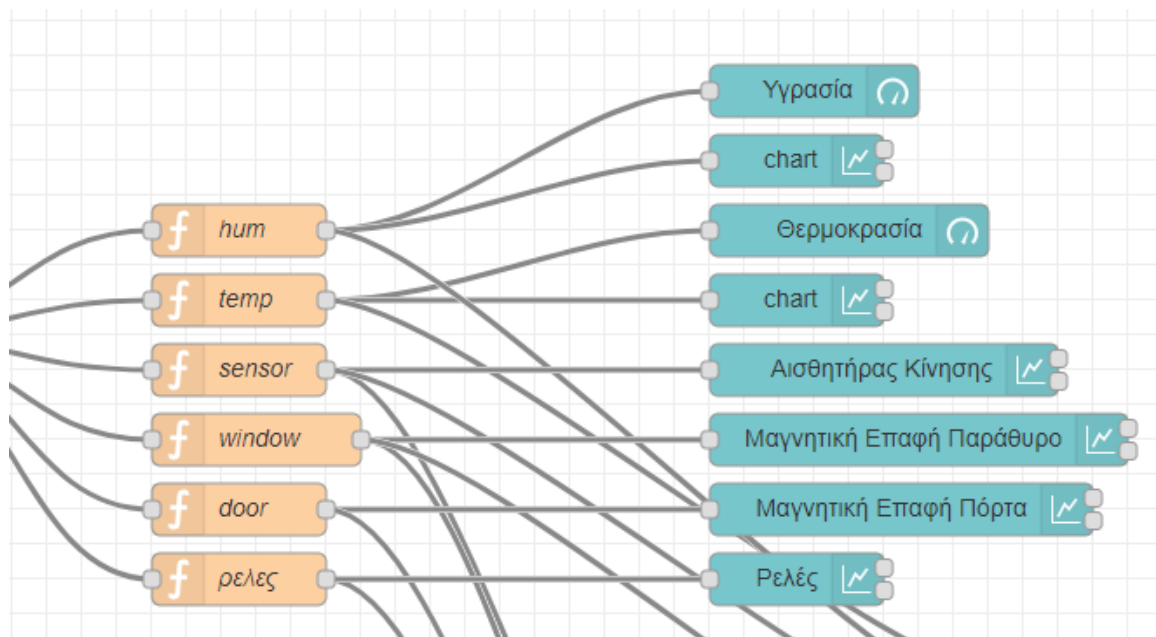
Εικόνα 57 Δημιουργία συναρτήσεων για τον διαχωρισμό των εισόδων

Παρακάτω φαίνεται ένα παράδειγμα για την υγρασία.



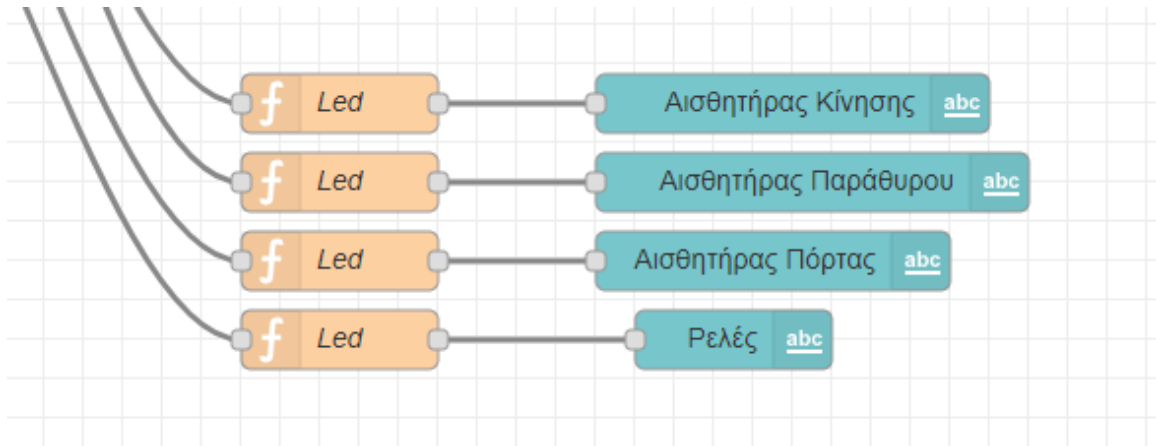
Εικόνα 58 Παράδειγμα συνάρτησης για την υγρασία

Έπειτα προχωρήσαμε στην απεικόνιση των δεδομένων με γραφήματα και σχέδια που θα φανούν παρακάτω.



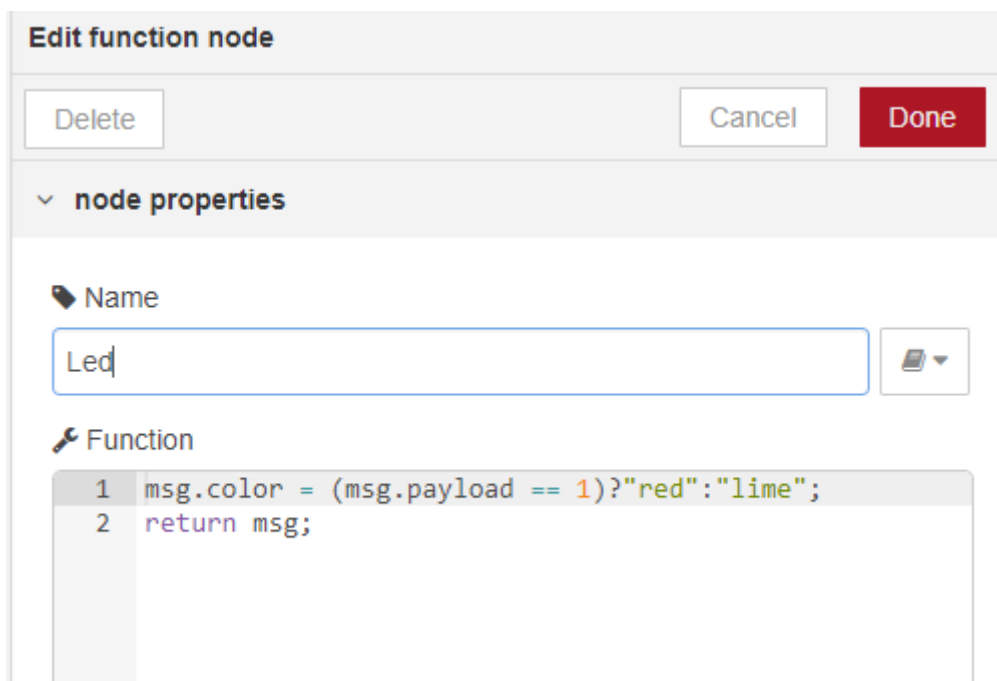
Εικόνα 59 Δημιουργία εξόδων και απεικόνιση τους

Δημιουργήσαμε σχετικές απεικονίσεις εικονικών led για να καταλαβαίνει ο επισκέπτης ευκολότερα τα απεικονιζόμενα δεδομένα.



Εικόνα 60 Δημιουργία εικονικών led για την απεικόνιση των καταστάσεων των δεδομένων

Η συνάρτηση που εισάγαμε για να δημιουργήσουμε το εικονικό led είναι η παρακάτω.

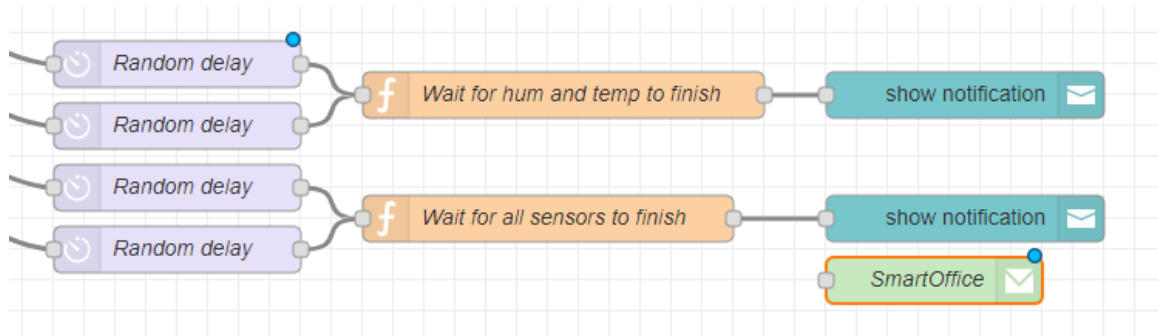


Εικόνα 61 Συνάρτηση τροποποίησης χρώματος εικονικού led

Έπειτα εισάγαμε απεικόνιση ειδοποιήσεων στο γραφικό του node-red .

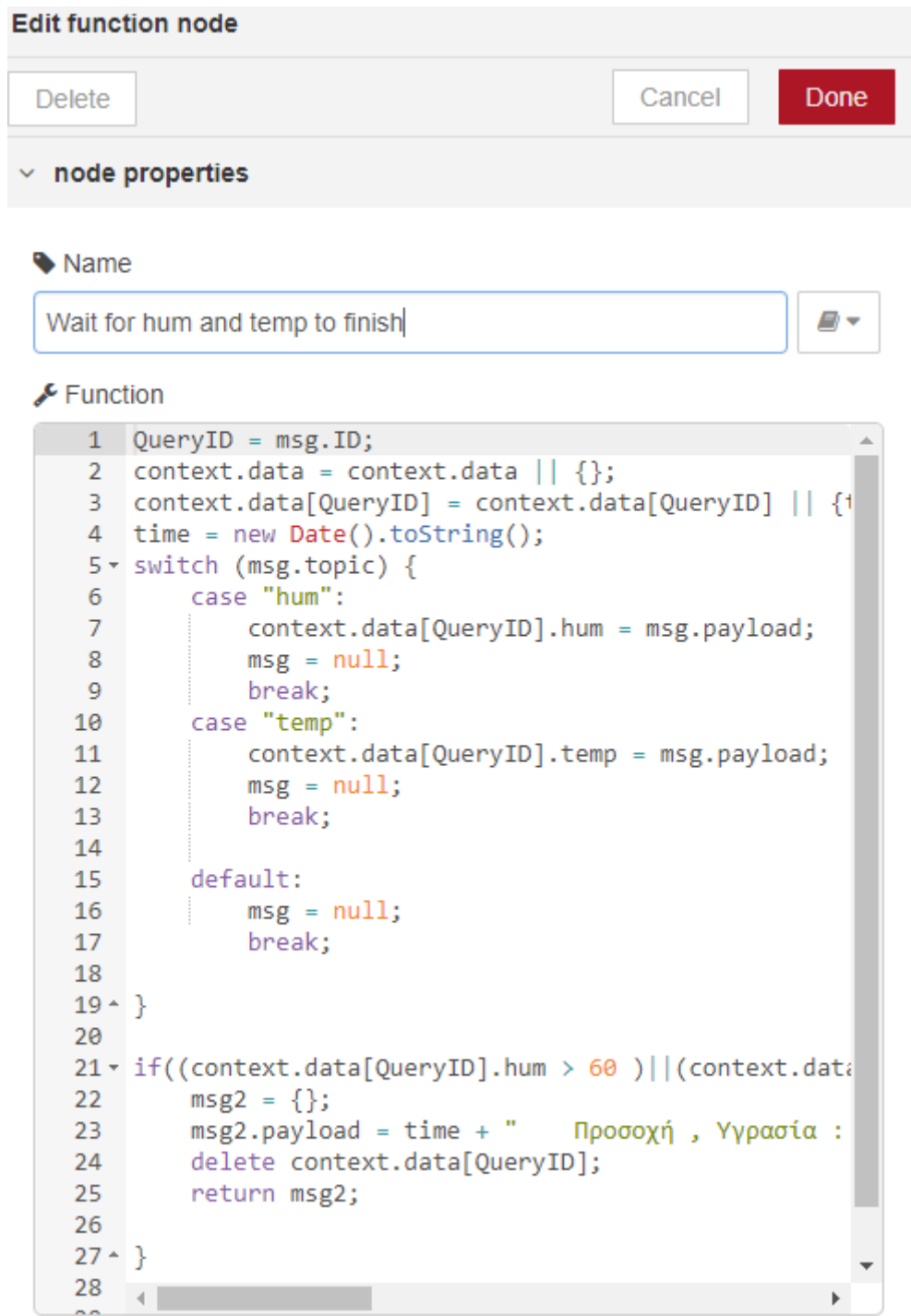
Επίσης έχουμε εισάγει την δυνατότητα αποστολής ειδοποιήσεων σε email , για να ενεργοποιηθεί η συγκεκριμένη δυνατότητα , χρειάζεται απλά να συνδέσουμε το κουτάκι με το "SmartOffice" στην ειδοποίηση που επιθυμούμε να λάβουμε.

Κατασκευή και εγκατάσταση Smart Office



Εικόνα 62 Απεικόνιση ειδοποιήσεων στο γραφικό

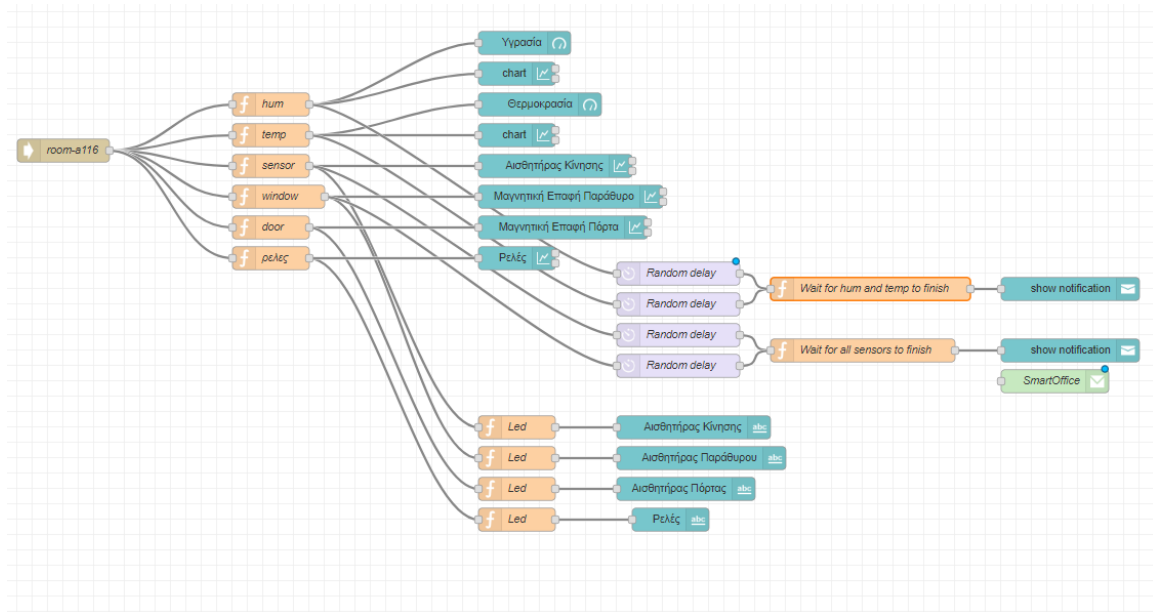
Για να μπορέσουμε να πραγματοποιήσουμε καταλληλότερη απεικόνιση των ειδοποιήσεων εισάγαμε συναρτήσεις , οι οποίες πραγματοποιούσαν αναμονή μέχρι να λάβουν δεδομένα από όλες τις μεταβλητές που ορίσαμε.



Εικόνα 63 Συνάρτηση αναμονής παραλαβής δεδομένων

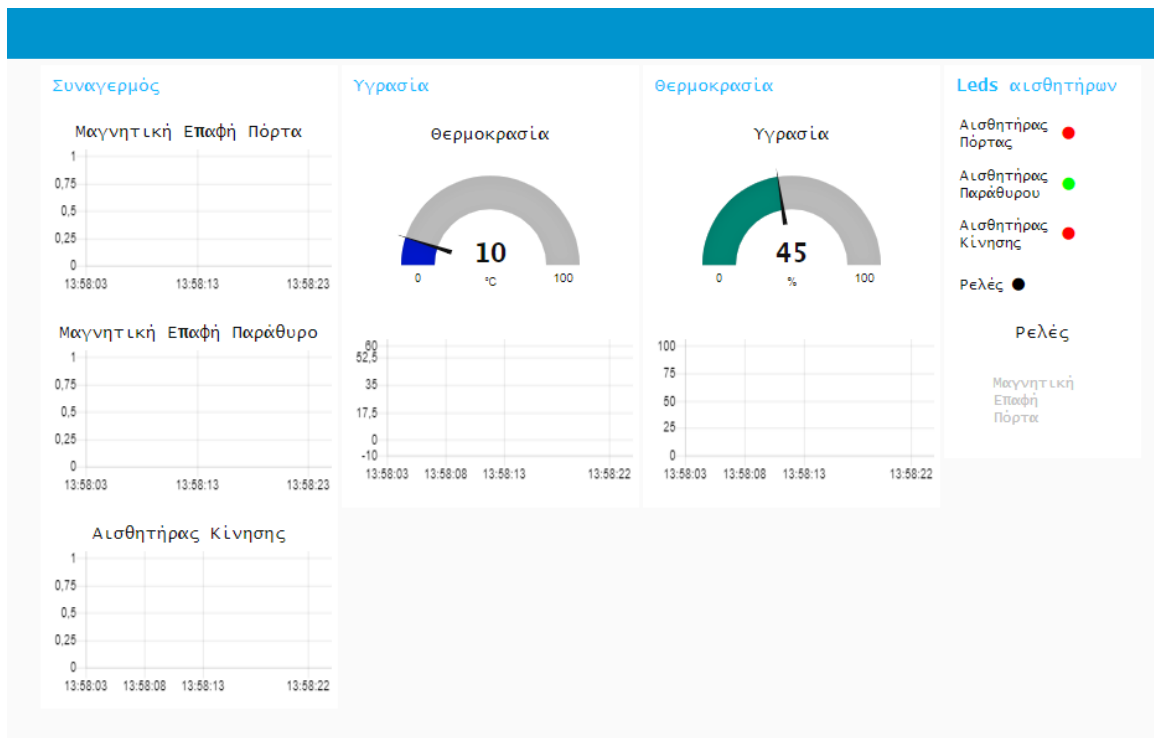
Το τελικό σχέδιο του διαχειριστικού του node-red φαίνεται στην επόμενη εικόνα.

Κατασκευή και εγκατάσταση Smart Office



Εικόνα 64 Τελικό σχέδιο node-red

Το τελικό γραφικό του node-red φαίνεται στην επόμενη εικόνα.



Εικόνα 65 Τελικό γραφικό node-red

ΚΕΦΑΛΑΙΟ 5: ΣΥΜΠΕΡΑΣΜΑΤΑ , ΜΕΛΛΟΝΤΙΚΕΣ ΠΡΟΕΚΤΑΣΕΙΣ ΚΑΙ ΛΕΙΤΟΥΡΓΙΕΣ

Στόχος μας ήταν ο σχεδιασμός ενός έξυπνου γραφείου όπου παραχωρεί στο χρήστη την δυνατότητα να έχει τον έλεγχο της κατάστασης που επικρατεί στο χώρο.

Η μέχρι τώρα πορεία της εργασίας μας δύναται να μας επιτρέψει τα εξής:

Έλεγχο της κατάστασης του χώρου μέσω ιστοσελίδας και ενημέρωση μέσω mail όταν καταγραφεί αλλαγή (με ορίζουμε εξ αρχής στο πρόγραμμα τις μεταβλητές που θα ενεργοποιούν/απενεργοποιούν αυτόματα τους ελεγκτές για να γίνονται οι αντίστοιχες ενέργειες)

Πιο συγκεκριμένα:

-Αποστολή email στον διαχειριστή αν εντοπίστηκε κίνηση στον χώρο.

-Αποστολή email στον διαχειριστή Αν πόρτες/παράθυρα είναι ανοιχτά.

-Έλεγχο υγρασίας/θερμοκρασίας και ρύθμιση τους με την ενεργοποίηση του κλιματιστικού από τιμές που έχουν οριστεί από τον χρήστη.

Όλα τα παραπάνω συνοψίζουν την έως τώρα πορεία της εργασίας μα δεν περιορίζονται σε αυτά καθώς υπογραμμίζεται πως υπάρχουν δυνατότητες για μελλοντική επέκταση.

ΜΕΛΟΝΤΙΚΕΣ ΠΡΟΕΚΤΑΣΕΙΣ/ΛΕΙΤΟΥΡΓΙΕΣ

Ιστοσελίδα

Η ιστοσελίδα ανανεώνεται αυτόματα παρέχοντας μας πληροφορίες για την κατάσταση του χώρου. Επιπρόσθετα θα μπορούσαν να τοποθετηθούν πεδία που αλλάζουν την αρχική τιμή των flags δηλαδή των μεταβλητών που είναι ορισμένες από τον χρήστη για την εκτέλεση μίας εντολής. Για παράδειγμα όταν η θερμοκρασία υπερβαίνει τους 27 βαθμούς πρέπει να ενεργοποιείται ο διακόπτης του κλιματιστικού. Αυτή η τιμή είναι προκαθορισμένη. Προσθέτοντας όμως το πεδίο ώστε ο χρήστης να μπαίνει στην ιστοσελίδα και να αλλάζει αυτή την τιμή, μας δίνεται η δυνατότητα να διαχειριζόμαστε εύκολα την θερμοκρασία του δωματίου χωρίς να χρειάζεται να επεμβούμε στον κώδικα ή να έχουμε γνώσεις προγραμματισμού.

Αισθητήρες

Η προσθήκη αισθητήρων μας δίνει περισσότερες πληροφορίες άρα περισσότερες δυνατότητες αυτοματισμού.

Φωτοδίοδος

Με την χρήση φωτοδίοδου μπορούμε να ελέγχουμε τα επίπεδα φωτισμού ώστε να ρυθμίζονται τα φώτα στον χώρο με dimmer και να εξοικονομούμε ενέργεια. Ακόμα μια επιλογή που έχουμε είναι να γίνεται έλεγχος των περσίδων σύμφωνα με τον ήλιο στον εξωτερικό χώρο του γραφείου. Οι περσίδες μπορούν να λειτουργήσουν ως πηγή ενέργειας με την χρήση ηλιακών πάνελ.

Αισθητήρες καπνού

Σε περίπτωση καπνού γίνεται άμεση ενημέρωση του διαχειριστή και υπάρχει ασφάλεια στον χώρο.

Κάμερες

Πέρα από τους αισθητήρες κίνησης και τις μαγνητικές επαφές η τοποθέτηση καμερών μπορεί πλέον να μας δώσει εικόνα ενώ συνδυαστικά με τα προηγούμενα ενεργοποιεί την δυνατότητα καταγραφής του χώρου σε περίπτωση που παραβιαστεί.

ΠΑΡΑΡΤΗΜΑ Α: ΚΩΔΙΚΑΣ ARDUINO

Κώδικας

//Εδώ εισάγουμε τις απαραίτητες βιβλιοθήκες που χρειαζόμαστε στον κώδικα

```
#include <dht.h>
```

```
#include <Ethernet.h>
```

```
//#include <SPI.h>
```

```
//#include "Phoc.h"
```

//Ορίζουμε τα pin που θα χρησιμοποιήσουμε

```
dht DHT;
```

```
#define switchPin 6
```

```
#define switchPin2 7
```

```
#define DHT11_PIN 8
```

```
#define motion_1 9
```

```
#define reles 2
```

//Ορίζουμε τις αρχικές μεταβλητές μας

```
int state;
```

```
int magnet1;
```

```
int magnet2;
```

```
float temp;
```

```
float hum;
```

```
int relestatus;
```

//Πραγματοποιούμε τις απαραίτητες ρυθμίσεις για την σύνδεση στο διαδίκτυο μέσω του ethernet shield

```
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
```

```
char server[] = "dweet.io"; // name address for Google (using DNS)
```

```
IPAddress ip(192, 168, 1, 177);
```

```
EthernetClient client;
```

```
void setup()
{
  //Εδώ ορίζουμε ποια pin είναι είσοδοι και ποια έξοδοι
  pinMode(motion_1, INPUT);
  pinMode(switchPin, INPUT_PULLUP);
  pinMode(switchPin2, INPUT_PULLUP);
  pinMode(reles, OUTPUT);
  //Ορίζουμε αρχικές τιμές στις εισόδους και στις εξόδους
  digitalWrite(switchPin, HIGH);
  digitalWrite(switchPin2, HIGH);
  digitalWrite(reles, HIGH);

  //Πραγματοποιούμε σύνδεση στο διαδίκτυο
  if (Ethernet.begin(mac) == 0) {
    Serial.println("Failed to configure Ethernet using DHCP");
    //Εάν αποτύχει να πάρει αυτόματα ip μέσω dhcp τότε προσπαθεί να πάρει χειροκίνητα την
    ip που του ορίσαμε παραπάνω
    Ethernet.begin(mac, ip);
  }
  //Περιμένουμε ένα δευτερόλεπτο για να επιβεβαιωθεί η σύνδεση
  delay(1000);
  Serial.begin(9600);
}

void loop()
{
  //Διαβάζουμε δεδομένα από τους αισθητήρες
  int sensor_1 = digitalRead(motion_1);
  int chk = DHT.read11(DHT11_PIN);
  temp = DHT.temperature;
  hum = DHT.humidity;
```

Κατασκευή και εγκατάσταση Smart Office

```
magnet1=digitalRead(switchPin);  
magnet2=digitalRead(switchPin2);
```

//Εάν η θερμοκρασία είναι παραπάνω από 34 , τότε ενεργοποιούμε το pin του ρελέ που του ορίσαμε στην αρχή

```
if (temp > 34) {  
    digitalWrite(reles, LOW);  
    relestatus = 1;  
} else {  
    digitalWrite(reles, HIGH);  
    relestatus = 0;  
}
```

//Κάνουμε σύνδεση στο dweet.io

```
if(client.connect(server, 80)){  
    //Στέλνουμε τα δεδομένα στο dweet.io  
    client.println("POST /dweet/quietly/for/room-  
a116?temp="+String(temp)+"&hum="+String(hum)+"&sensor="+String(sensor_1)+"&door="+  
String(magnet1)+"&window="+String(magnet2)+"&reles="+String(relestatus)+" HTTP/1.1");  
    }  
    else {  
        //Εάν αποτύχει την σύνδεση στο διαδίκτυο τότε εμφανίζουμε το αντίστοιχο μήνυμα στην  
        κονσόλα  
        Serial.println("Δεν μπόρεσα να κάνω dweet");  
    }  
}
```

//Σε περίπτωση που έχουν ολοκληρωθεί όλα επιτυχώς , τότε κάνουμε αποσύνδεση για να κλείσουμε το session

```
if (!client.connected()) {  
    Serial.println();  
    Serial.println("disconnecting.");  
}  
client.stop();
```


Βιβλιογραφία – Πηγές

<https://el.wikipedia.org/wiki/>

<https://www.android.com/>

<https://dweet.io/>

<https://www.youtube.com/>

<http://freeboard.io/>

<https://nodered.org/>

<https://www.proxmox.com/en/>

<http://opensourceecology.org/>