

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**« Σύνθεση νέων τεχνολογιών IoT για την υλοποίηση
συστήματος παρακολούθησης περιβαλλοντικών
παραμέτρων σε διαφορετικές τοποθεσίες »**



**Του φοιτητή Πολατίδη Χρήστου
Αρ. Μητρώου: 2020262**

Ημερομηνία 9/24

**Επιβλέπων
Κοκκώνης Γεώργιος
Βαθμίδα Επίκουρος Καθηγητής**

Τίτλος Δ.Ε. Σύνθεση νέων τεχνολογιών IoT για την υλοποίηση συστήματος παρακολούθησης περιβαλλοντικών παραμέτρων σε διαφορετικές τοποθεσίες

Κωδικός Δ.Ε. 24169

Όνοματεπώνυμο φοιτητή ΠΟΛΑΤΙΔΗΣ ΧΡΗΣΤΟΣ

Όνοματεπώνυμο εισηγητή ΚΟΚΚΩΝΗΣ ΓΕΩΡΓΙΟΣ

Ημερομηνία ανάληψης Δ.Ε. 27/3/24

Ημερομηνία περάτωσης Δ.Ε. 9/24

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Πολατίδη Χρήστου που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

«Αφιερώνω αυτή την εργασία στη γιαγιά μου Σοφία Πολατίδου, την οποία έχασα φέτος και στο Άγιο Πνεύμα που ακόμα με ανέχεται.»

Πρόλογος

Η επιλογή να ασχοληθώ με αυτό το θέμα διπλωματικής εργασίας έγινε λόγω του ενδιαφέροντος μου για τις τεχνολογίες Internet of Things (IoT). Μου άρεσε το γεγονός ότι συνδυάζει τους τομείς των **δικτύων**, του **προγραμματισμού** και των **ηλεκτρονικών**. Μια πρώτη επαφή με το αντικείμενο είχα στο 2^ο μεταπτυχιακό μου στο τμήμα πληροφορικής του ΑΠΘ. Ακόμα με βοήθησε στην επιλογή των τεχνολογιών, οι διαλέξεις στο μάθημα ελεύθερη επιλογή όπου είχα τη δυνατότητα να καταγράψω τις τεχνολογίες που έχουν επικρατήσει στην αγορά, καθώς και τις απαιτήσεις ενός συστήματος IoT στη βιομηχανία. Η χρήση και ειδικότερα η εξοικείωση με τις σύγχρονες τεχνολογίες IoT με έκανε να εκτιμήσω περαιτέρω την ευελιξία και την αποδοτικότητα αυτών των εργαλείων και μέσα από την υλοποίηση ενός τέτοιου συστήματος να βελτιώσω τις δεξιότητές μου έτσι ώστε να μπορώ να ασχοληθώ περεταίρω επαγγελματικά ή και ερευνητικά σε ένα άλλο επίπεδο σε αυτό το τομέα της πληροφορικής.

Περίληψη

Η διπλωματική εργασία αναφέρεται στην αναζήτηση συγχρόνων τεχνολογιών και τη σύνθεση τους για την υλοποίηση ενός συστήματος Internet of Things (IoT) που παρακολουθεί περιβαλλοντικές παραμέτρους σε διαφορετικές τοποθεσίες, σε πραγματικό χρόνο. Το σύστημα συλλέγει και να αναλύει δεδομένα από αισθητήρες για τη θερμοκρασία, την υγρασία, τη φωτεινότητα, τον θόρυβο και τη ποιότητα αέρα. Η Βιβλιογραφική έρευνα ανέδειξε τα κατάλληλα όρια για τις παραπάνω παραμέτρους και έτσι προτάθηκε και υλοποιήθηκε ένας αυτόματος μηχανισμός βαθμολόγησης του περιβάλλοντος.

Έμμεσος στόχος είναι να συγκρίνει ανταγωνιστικές τεχνολογίες και πρωτόκολλα και να αναδείξει σύγχρονες λύσεις. Πρωτόκολλα όπως το MQTT, ο agent Telegraf, το λογισμικό Grafana, η nosql βάση δεδομένων Fluxdb και ο Mosquitto broker είναι μερικές από τις τεχνολογίες που συνθέτουν το σύστημα. Ακόμα δοκιμάστηκαν διαφορετικές λύσεις για τα ζητήματα δικτύωσης όπως η χρήση του IPv6 και του tunneling έτσι ώστε να προταθούν οι καλύτερες λύσεις.

Η ανάλυση των δεδομένων που συλλέγονται βοήθησαν στην εξαγωγή συμπερασμάτων για τα περιβάλλοντα που αφορούσαν. Οι τεχνολογίες που αξιοποιήθηκαν επιτρέπουν την εκτίμηση και βελτίωση της ποιότητας του περιβάλλοντος, καθιστώντας το σύστημα πολύτιμο εργαλείο για διαφορετικούς επαγγελματικούς τομείς όπως η βιομηχανία, η υγεία και οι έξυπνες πόλεις.

Η διπλωματική εργασία εκτός από την τεχνική περιγραφή και την υλοποίηση του συστήματος, περιλαμβάνει και μια εκτενή ανάλυση των προκλήσεων και των προοπτικών της τεχνολογίας IoT, υπογραμμίζοντας τη σημασία της ασφάλειας δεδομένων, της διαλειτουργικότητας των συσκευών και της βελτιωμένης διαχείρισης των δεδομένων στην νέα ψηφιακή εποχή.

Λέξεις-κλειδιά: Διαδίκτυο των πράγματος; MQTT; Telegraf; Fluxdb; Grafana; IPv6; Δικτύωση; Ενσωματωμένα Συστήματα; Περιβάλλον;

«IoT technologies Synthesis for Weather environment monitoring in multiple places»

«POLATIDIS CHRISTOS»

Abstract

The thesis refers to the search for modern technologies and their synthesis for the implementation of an Internet of Things (IoT) system that monitors environmental parameters in different locations, in real time. The system collects and analyzes data from sensors for temperature, humidity, brightness, noise and air quality. Literature research indicated the appropriate thresholds for the above parameters and thus an automatic environmental scoring mechanism was proposed and implemented.

The indirect objective is to compare competing technologies and protocols and to highlight state-of-the-art solutions. Protocols such as MQTT, the Telegraf agent, Grafana software, nosql database Fluxdb and Mosquitto broker are some of the technologies that make up the system. Still different solutions for the networking issues such as the use of IPv6 and tunneling were tested so that the best solutions could be proposed.

The analysis of the data collected helped to draw conclusions about the environments involved. The technologies exploited allow the assessment and improvement of the quality of the environment, making the system a valuable tool for different professional sectors such as industry, health and smart cities.

In addition to the technical description and the implementation of the system, the thesis includes an extensive analysis of the challenges and perspectives of IoT technology, highlighting the importance of data security, device interoperability and improved data management in the new digital era.

Keywords: Internet of things; MQTT; Telegraf; Fluxdb; Grafana; IPv6; Networking; Embedded systems; environment;

Ευχαριστίες

Σε αυτό το σημείο θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες σε αυτούς που βοήθησαν να υλοποιηθεί αυτή η διπλωματική εργασία :

- Τον Δρ. Κοκκώνη Γεώργιο επιβλέποντα καθηγητή της εργασίας μου ο οποίος μου έδειξε εμπιστοσύνη δίνοντας μου ελευθερία στην επιλογή θέματος. Ακόμα μου έδωσε σημαντικές ιδέες για την επέκταση της εργασίας και μου παρείχε σημαντική υποστήριξη σε όλη τη διαδικασία της ερευνάς.
- Τον Δρ. Περικλή Χατζημίσιο καθηγητή του τμήματος Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων για τις ιδέες και τη προοπτική που μου έδωσε έτσι ώστε να ασχοληθώ με το αντικείμενο του Internet of things.

Περιεχόμενα

Πρόλογος.....	iv
Περίληψη.....	v
Abstract	vi
Ευχαριστίες.....	vii
Περιεχόμενα.....	viii
Κατάλογος Εικόνων	x
Κατάλογος Πινάκων	xii
Συνοτομογραφίες	xii
ΕΙΣΑΓΩΓΗ.....	1
1. INTERNET OF THINGS.....	1
1.1 Τί είναι το Internet Of Things	1
1.2 Πορεία του IOT στο χρόνο	1
1.3 Αρχιτεκτονική IOT	2
1.4 Πρωτόκολλα επικοινωνίας IOT	4
1.4.1 MQTT (Message Queue Telemetry Transport).....	4
1.4.2 CoAP (Constrained Application Protocol).....	5
1.4.3 HTTP (HyperText Transfer Protocol)	5
1.4.4 XMPP (Extensible Messaging and Presence Protocol).....	5
1.4.5 Zigbee.....	6
1.4.6 Z-wave.....	6
1.4.7 Bluetooth	7
1.4.8 LoRaWAN.....	8
1.4.9 IEEE 802.11 ah.....	8
1.4.10 Cellular (Κυψελοειδής Δίκτυα):.....	9
1.5 Σύγκριση πρωτοκόλλων IoT	10
1.6 Εφαρμογές και Παραδείγματα Χρήσης του IoT	11
1.6.1 Έξυπνες Πόλεις (Smart Cities).....	12
1.6.2 Έξυπνα Σπίτια (Smart Homes).....	12
1.6.3 Βιομηχανικό IoT (IIoT):.....	13
1.6.4 Υγειονομική Περίθαλψη(Healthcare)	14
1.6.5 Γεωργία(Agricultural IoT).....	14
1.7 Προκλήσεις και προοπτικές στον Τομέα του IoT	15
2. ΠΑΡΑΚΟΛΟΥΘΗΣΗ ΠΕΡΙΒΑΛΛΟΝΤΟΣ ΓΡΑΦΕΙΟΥ	16

2.1	Επίπεδα αξιολόγησης γραφείου	16
2.1.1	Θερμοκρασία.....	16
2.1.2	Υγρασία.....	17
2.1.3	Ποιότητα του αέρα	18
2.1.4	Φωτεινότητα(Luminosity).....	21
2.1.4	Ακουστικός θόρυβος(Noise dbA)	22
3	ΜΕΛΕΤΗ ΣΥΣΤΗΜΑΤΟΣ ΙΟΤ.....	24
3.1	Μελέτη περίπτωσης του συστήματος.....	24
3.2	Απαιτήσεις συστήματος	25
3.3	Σχεδιασμός του συστήματος ΙοΤ.....	25
3.3.1	Σχεδιασμός στο Cloud.....	26
3.3.2	Σχεδιασμός τοπικά.....	26
4	ΥΛΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ	28
4.1	Επιλογή Hardware & Software	28
4.2	Συνδεσμολογία	29
4.3	Προγραμματισμός ESP32	30
4.4	Εγκατάσταση Mqtt Broker.....	36
4.4.1	Εγκατάσταση Cloud broker.....	36
4.4.2	Εγκατάσταση Local Mosquitto broker	38
4.5	Εγκατάσταση Telegraf agent.....	39
4.6	Εγκατάσταση Influxdb	40
4.7	Εγκατάσταση Grafana	42
4.7.1	Grafana cloud	42
4.7.2	Grafana local	43
4.7.3	Βαθμολόγηση περιβάλλοντος.....	50
4.7.3	Alerts	52
4.8	Βοηθητικά εργαλεία και ρυθμίσεις δικτύου	54
4.8.1	Port forward.....	54
4.8.2	IPv6 Filter.....	57
4.8.3	Mqtt Analyzer.....	58
4.8.4	Expose broker (Tunnel).....	58
4.9	Troubleshooting.....	61
4.10	Subscribe & λήψη αποφάσεων.....	62
5	ΑΠΟΤΕΛΕΣΜΑΤΑ	67

5.1	Σενάριο 1 : Σπίτι – Γραφείο – Εξωτερικός χώρος	67
5.2	Σενάριο 2 : Μεγάλο Γραφείο 2 ατόμων – Μικρό Γραφείο 5 ατόμων – Εξωτερικός χώρος	71
6	ΣΥΜΠΕΡΑΣΜΑΤΑ.....	73
7	ΜΕΛΛΟΝΤΙΚΗ ΕΡΕΥΝΑ	74
	ΒΙΒΛΙΟΓΡΑΦΙΑ.....	75

Κατάλογος Εικόνων

Εικόνα 1	IoT history [4].....	2
Εικόνα 2	Αρχιτεκτονική δικτύου IoT [9]	3
Εικόνα 3	Αρχιτεκτονική λειτουργίας IoT [10].....	3
Εικόνα 4	MQTT πρωτόκολλο [12]	4
Εικόνα 5	Zigbee on Smart home [17]	6
Εικόνα 6	Δίκτυο z-wave για smart home [18]	7
Εικόνα 7	Εφαρμογές Bluetooth στο IoT [20].....	7
Εικόνα 8	Αρχιτεκτονική LoRaWan [22].....	8
Εικόνα 9	Αρχιτεκτονική IEEE 802.11.ah [23]	9
Εικόνα 10	Εφαρμογές IoT μέσω δικτύου 5G [25]	9
Εικόνα 11	IoT σε έξυπνες πόλεις [30]	12
Εικόνα 12	IoT σε έξυπνα σπίτια [32].....	13
Εικόνα 13	Η εξέλιξη του IIoT [34]	13
Εικόνα 14	Οφέλη χρήσης IoT στον τομέα healthcare [36].....	14
Εικόνα 15	Οφέλη του Agricultural IoT [37]	15
Εικόνα 16	Τα κατώφλια θερμοκρασίας στο Grafana	17
Εικόνα 17	Τα κατώφλια υγρασίας στο Grafana	18
Εικόνα 18	AQI στην Αγία Σοφία [45]	19
Εικόνα 19	Κατώφλια ποιότητας του αέρα στο Grafana.....	21
Εικόνα 20	Κατώφλια φωτεινότητας στο Grafana	22
Εικόνα 21	Επίπεδα ακουστικού θορύβου[54]	23
Εικόνα 22	Κατώφλια θορύβου στο Grafana	23
Εικόνα 23	Γραφείο εργασίας 2 ατόμων 100TM.....	24
Εικόνα 24	Γραφείο εργασίας 5 ατόμων 60TM.....	25
Εικόνα 25	Αρχιτεκτονική cloud	26
Εικόνα 26	Αρχιτεκτονική local	27
Εικόνα 27	Συνδεσμολογία Esp32	29
Εικόνα 28	Συνδεσμολογία στο άκρο	30
Εικόνα 29	Calibration photoresistor.....	34
Εικόνα 30	Χρήση EMQX Broker	36
Εικόνα 31	Mosquitto Broker - MQTT Explorer.....	37
Εικόνα 32	Mosquitto κρυπτογράφηση Passwd	38
Εικόνα 33	Εκκίνηση Telegraf	39
Εικόνα 34	Telegraf Configuration.....	40

Εικόνα 35 Data Explorer	41
Εικόνα 36 Flux query	41
Εικόνα 37 Mqtt Data source	42
Εικόνα 38 Panel query.....	42
Εικόνα 39 Dashboard cloud Mqtt.....	43
Εικόνα 40 Influxdb configuration	44
Εικόνα 41 Query builder	45
Εικόνα 42 Panel configuration	45
Εικόνα 43 Thresholds.....	46
Εικόνα 44 Label transformation	46
Εικόνα 45 Query combination.....	47
Εικόνα 46 Compare tab	47
Εικόνα 47 Work inside tab	48
Εικόνα 48 Average tab	48
Εικόνα 49 Clock panel plugin.....	49
Εικόνα 50 Memory usage query.....	49
Εικόνα 51 Broker Health tab	50
Εικόνα 52 Βαθμολογία σημείου ενδιαφέροντος	52
Εικόνα 53 Alert configuration.....	52
Εικόνα 54 Ram alert	53
Εικόνα 55 Discord channel alert	54
Εικόνα 56 Port Forward	55
Εικόνα 57 Server static local ip.....	56
Εικόνα 58 Ipv6 Filter.....	57
Εικόνα 59 Mosquitto testing.....	57
Εικόνα 60 Mqtt Analyzer	58
Εικόνα 61 Ngrok service	59
Εικόνα 62 Localto service	59
Εικόνα 63 Αρχιτεκτονική Localtonet.....	59
Εικόνα 64 Localtunnel service	60
Εικόνα 65 Σύνδεση στο Local Grafana από το internet.....	60
Εικόνα 66 Wireshark.....	61
Εικόνα 67 Nmap.....	61
Εικόνα 68 Mosquitto Log.....	62
Εικόνα 69 Subscribe λειτουργία.....	62
Εικόνα 70 Λειτουργία subscribe	66
Εικόνα 71 Σύγκριση θερμοκρασίας.....	67
Εικόνα 72 Σύγκριση Υγρασίας	68
Εικόνα 73 Σύγκριση Φωτεινότητας	69
Εικόνα 74 Σύγκριση Ποιότητας του αέρα	69
Εικόνα 75 Σύγκριση Θορύβου	70
Εικόνα 76 Σύγκριση δείκτη Overall Score	70
Εικόνα 77 Παρακολούθηση παραμέτρων σε συσχέτιση με κάποιο γεγονός.....	71
Εικόνα 78 Σενάριο 2	72
Εικόνα 79 Tab Average Last 24h.....	72

Κατάλογος Πινάκων

Πίνακας 1 Σύγκριση πρωτοκόλλων εφαρμογής του IoT [11, 13-15]	10
Πίνακας 2 Σύγκριση πρωτοκόλλων δικτύου εφαρμογών IoT [26-28]	11
Πίνακας 3 Αισθητήρες μέτρησης ποιότητας αέρα	20

Συντομογραφίες

Δ.Ε.	Διπλωματική Εργασία
ΔΠΠΑΕ	Διεθνές Πανεπιστήμιο Ελλάδος
IoT	Internet Of Things
AQI	Air Quality Index
VOCs	Volatile Organic Compounds
MQTT	Message Queuing Telemetry Transport
HTTP	Hypertext Transfer Protocol
COAP	Constrained Application Protocol
XMPP	Extensible Messaging And Presence Protocol
IIoT	Industrial Internet Of Things
XML	Extensible Markup Language

ΕΙΣΑΓΩΓΗ

Η εργασία έχει σκοπό τη μελέτη, το σχεδιασμό και την υλοποίηση ενός συστήματος IoT το οποίο θα παρακολουθεί το περιβάλλον σε συγκεκριμένα σημεία. Ειδικότερα συλλέγονται τιμές για τη θερμοκρασία, την υγρασία, τη φωτεινότητα, τον ακουστικό θόρυβο και την ποιότητα του αέρα. Το σύστημα επιτρέπει την οπτικοποίηση της κατάστασης του περιβάλλοντος στα σημεία ενδιαφέροντος έτσι ώστε να μπορούμε να συγκρίνουμε σε πραγματικό και παρελθοντικό χρόνο τα διαφορετικά περιβάλλοντα. Το κάθε περιβάλλον λαμβάνει μια βαθμολογία σύμφωνα με τις τιμές των παραπάνω παραμέτρων. Ακόμα υπάρχει η λειτουργία subscribe όπου δημιουργείτε ένας διαδραστικός αυτοματισμός ανάλογα με τις τιμές των παραμέτρων που συλλέγουμε.

1. INTERNET OF THINGS

1.1 Τί είναι το Internet Of Things

Το Internet of Things (IoT) αναφέρεται σε ένα δίκτυο συσκευών στις οποίες είναι συνδεδεμένοι αισθητήρες, λογισμικό και άλλες τεχνολογίες με σκοπό τη σύνδεση τους και την ανταλλαγή δεδομένων με άλλες συσκευές και συστήματα μέσω του διαδικτύου. Αυτές οι συσκευές μπορούν να συλλέγουν και να ανταλλάσσουν δεδομένα, επιτρέποντας την παρακολούθηση, τον έλεγχο και τη διαχείριση ενός περιβάλλοντος σε πραγματικό χρόνο. [\[1\]](#)

1.2 Πορεία του IOT στο χρόνο

Η ιδέα του Internet of Things (IoT) ξεκίνησε στα τέλη της δεκαετίας του 1990, όταν ο Kevin Ashton, ερευνητής και συνιδρυτής του Auto-ID Center στο MIT, χρησιμοποίησε για πρώτη φορά τον όρο το 1999 για να περιγράψει ένα σύστημα όπου φυσικά αντικείμενα θα ήταν συνδεδεμένα στο διαδίκτυο μέσω αισθητήρων RFID.

Με την εξέλιξη της τεχνολογίας, το IoT γνώρισε ραγδαία ανάπτυξη κατά τη διάρκεια της δεκαετίας του 2000 και 2010, κυρίως λόγω της μείωσης του κόστους των αισθητήρων και της αύξησης της υπολογιστικής ισχύος και της χωρητικότητας των δικτύων. Προς το τέλος της δεκαετίας του 2010, το IoT είχε ήδη αρχίσει να εφαρμόζεται σε διάφορους τομείς, όπως η υγειονομική περίθαλψη, η γεωργία, οι έξυπνες πόλεις και η βιομηχανία (Industrial IoT).

Τα τελευταία χρόνια, η πρόοδος στις τεχνολογίες 5G, η ανάπτυξη της τεχνητής νοημοσύνης (AI) και των μεγάλων δεδομένων (Big Data) έχουν δώσει νέα ώθηση στο IoT, ενισχύοντας τις δυνατότητές του για την παροχή πιο σύνθετων και εξατομικευμένων υπηρεσιών σε πραγματικό χρόνο. Το IoT πλέον δεν αφορά μόνο τη συνδεσιμότητα, αλλά και την έξυπνη λήψη αποφάσεων, τη βελτίωση της αποδοτικότητας και τη δημιουργία νέων επιχειρηματικών μοντέλων. Οι προβλέψεις δείχνουν ότι το IoT θα συνεχίσει να επεκτείνεται, με δισεκατομμύρια συσκευές να αναμένεται να είναι συνδεδεμένες μέχρι το 2030, μεταμορφώνοντας τον τρόπο που οι άνθρωποι ζουν και εργάζονται. [\[2-3\]](#)



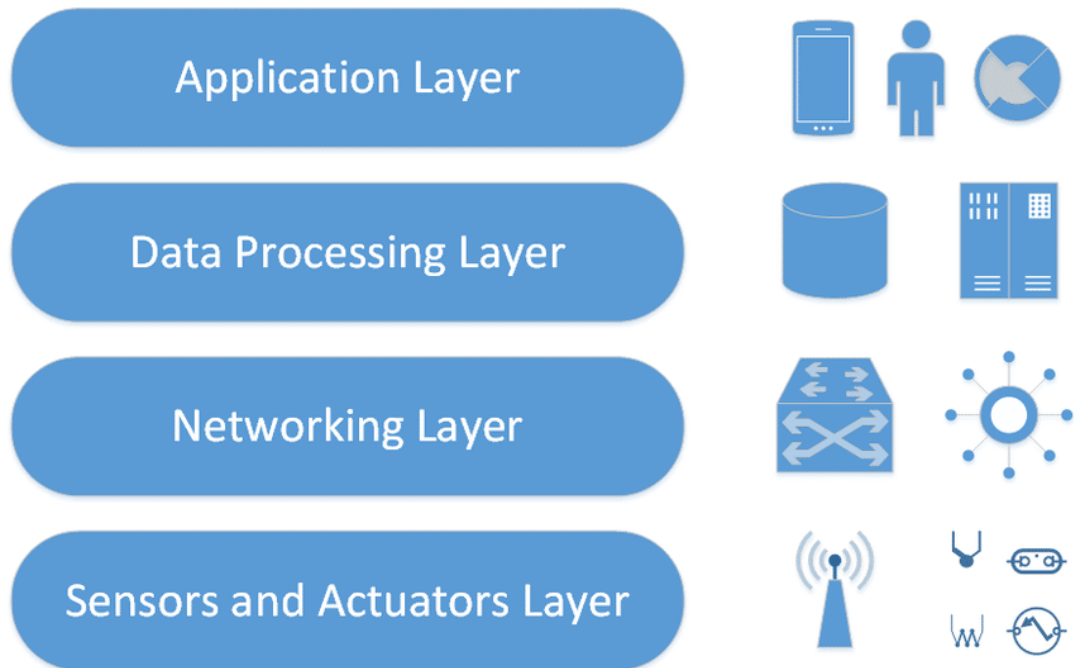
Εικόνα 1 IoT history [4]

1.3 Αρχιτεκτονική IOT

Για την περιγραφή της αρχιτεκτονικής του Internet of Things (IoT), μπορούμε να αναλύσουμε τα κύρια επίπεδα που συνθέτουν το σύστημα IoT.

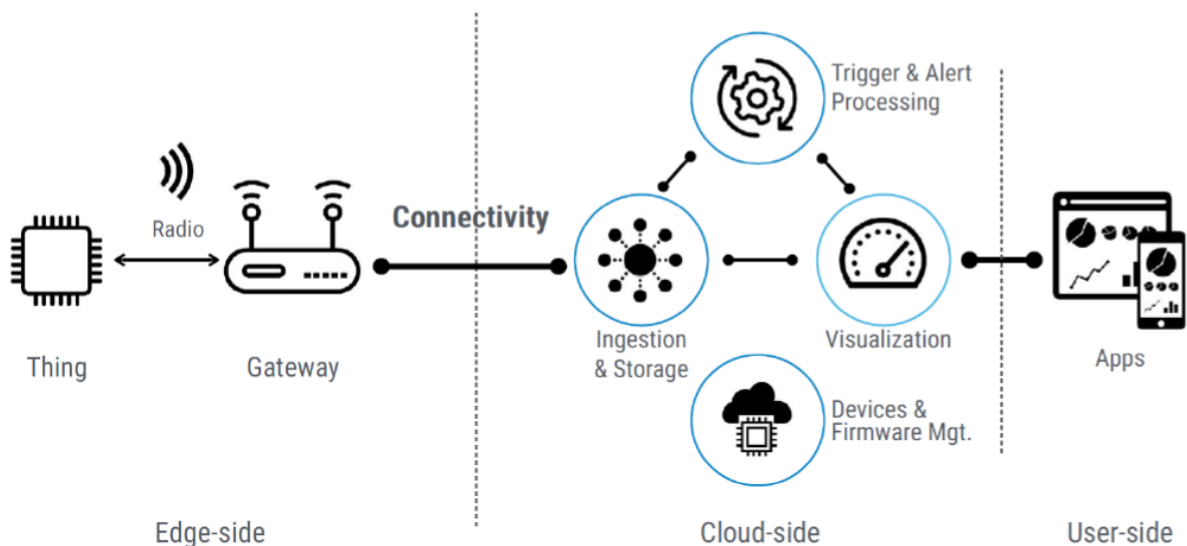
- I. **Επίπεδο Ανίχνευσης (Sensing Layer):** Το επίπεδο ανίχνευσης αποτελεί το θεμέλιο του IoT. Εδώ ενσωματώνονται αισθητήρες και ενεργοποιητές (actuators) που συλλέγουν δεδομένα από το φυσικό περιβάλλον. Οι αισθητήρες μπορούν να μετρούν παραμέτρους όπως θερμοκρασία, υγρασία, φως, πίεση, κίνηση, και άλλα. Τα δεδομένα που συλλέγονται στη συνέχεια προωθούνται σε άλλα επίπεδα για επεξεργασία και ανάλυση.[5]
- II. **Επίπεδο Δικτύου (Network Layer):** Το επίπεδο δικτύου είναι υπεύθυνο για τη διασύνδεση των συσκευών και τη μεταφορά των δεδομένων που συλλέγονται από το επίπεδο ανίχνευσης. Αυτό περιλαμβάνει πρωτόκολλα επικοινωνίας και τεχνολογίες όπως Wi-Fi, Bluetooth, Zigbee, και δίκτυα κινητής τηλεφωνίας (π.χ., 4G/5G). Ο σκοπός είναι να διασφαλιστεί ότι τα δεδομένα μπορούν να μεταφερθούν αξιόπιστα και με ασφάλεια προς το επόμενο επίπεδο.[6]
- III. **Επίπεδο Επεξεργασίας Δεδομένων (Data Processing Layer):** Στο επίπεδο αυτό πραγματοποιείται η ανάλυση των δεδομένων που έχουν συλλεχθεί. Συχνά περιλαμβάνει τεχνολογίες cloud computing ή edge computing, ανάλογα με το πού πραγματοποιείται η επεξεργασία. Τα δεδομένα υποβάλλονται σε επεξεργασία, φιλτράρονται, και αποθηκεύονται για μελλοντική χρήση ή ανάλυση. Αυτό το επίπεδο παρέχει τη δυνατότητα λήψης αποφάσεων σε πραγματικό χρόνο, ανάλογα με τα δεδομένα που επεξεργάζονται.[7]
- IV. **Επίπεδο Εφαρμογών (Application Layer):** Το τελευταίο επίπεδο της αρχιτεκτονικής αφορά τις εφαρμογές που χρησιμοποιούν τα επεξεργασμένα δεδομένα για να προσφέρουν συγκεκριμένες υπηρεσίες. Αυτές οι εφαρμογές μπορεί να περιλαμβάνουν έξυπνες πόλεις, έξυπνα σπίτια, βιομηχανικό IoT, και άλλες εφαρμογές που αξιοποιούν τις δυνατότητες του IoT για τη βελτίωση της καθημερινής ζωής και της επιχειρηματικής αποδοτικότητας.[8]

Στην παρακάτω εικόνα 2 βλέπουμε τα τέσσερα επίπεδα που αναφέρθηκαν παραπάνω.



Εικόνα 2 Αρχιτεκτονική δικτύου IoT [9]

Μια ακόμη απεικόνιση της αρχιτεκτονικής IoT είναι η παρακάτω (εικόνα 3). Εδώ έχουμε το επίπεδο **edge** όπου βρίσκουμε αισθητήρες, μικροελεγκτές και gateways. Είναι το επίπεδο όπου συλλέγονται τα δεδομένα του συστήματος. Κεντρικά βλέπουμε το επίπεδο **Cloud-side** αποθηκεύεται και επεξεργάζεται η πληροφορία που έχουμε συλλέξει. Στα δεξιά βλέπουμε το επίπεδο **User-side** όπου γίνεται η οπτικοποίηση των δεδομένων και των αποτελεσμάτων της διαδικασίας της επεξεργασίας τους.



Εικόνα 3 Αρχιτεκτονική λειτουργίας IoT [10]

1.4 Πρωτόκολλα επικοινωνίας ΙΟΤ

Τα πρωτόκολλα επικοινωνίας αποτελούν το βασικό συστατικό για τη διασύνδεση των συσκευών ΙοΤ, επιτρέποντας την αποστολή δεδομένων με αποτελεσματικότητα και αξιοπιστία. Παρακάτω θα αναφερθούμε στα κυριότερα. Τα πρώτα τέσσερα είναι επιπέδου εφαρμογής και τα υπόλοιπα έξι είναι επιπέδου δικτύου.

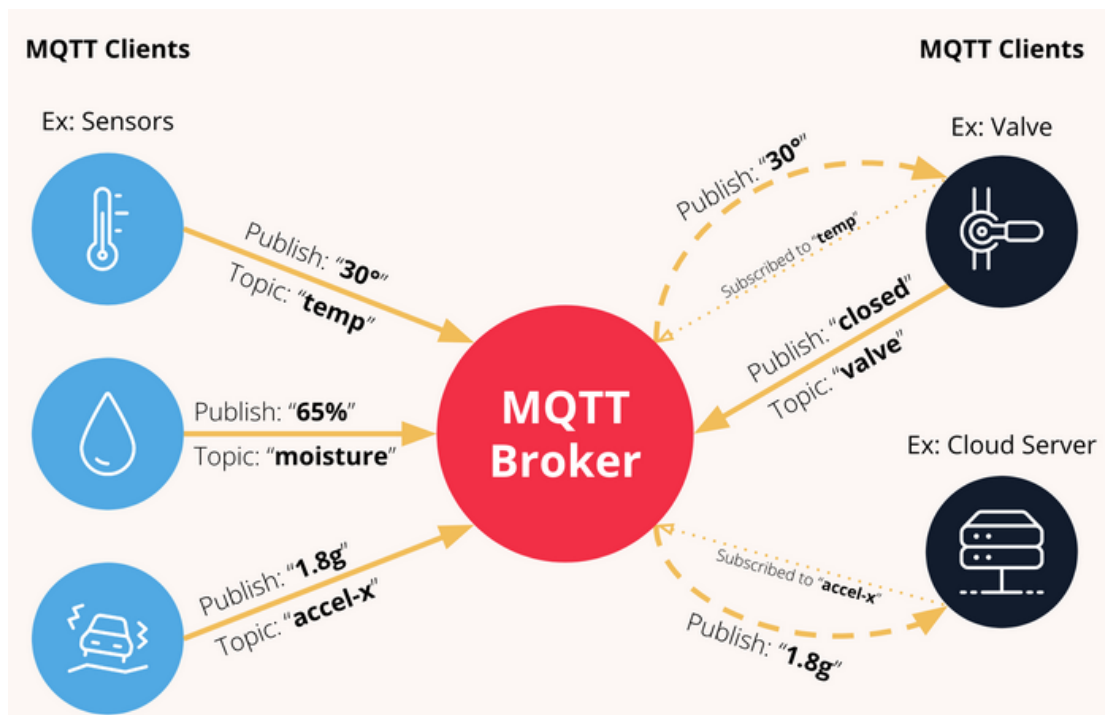
1.4.1 MQTT (Message Queue Telemetry Transport)

Το MQTT είναι ένα ελαφρύ πρωτόκολλο μηνυμάτων που χρησιμοποιείται ευρέως στο ΙοΤ για την επικοινωνία μεταξύ συσκευών που έχουν περιορισμένους πόρους. Σχεδιάστηκε για να παρέχει αξιόπιστη μετάδοση μηνυμάτων σε περιβάλλοντα χαμηλού εύρους ζώνης, υψηλής καθυστέρησης, ή ασταθούς δικτύου.

Χρησιμοποιεί αρχιτεκτονική publisher-subscriber, όπου οι συσκευές (publishers) στέλνουν μηνύματα σε ένα κεντρικό broker, και άλλες συσκευές (subscribers) λαμβάνουν αυτά τα μηνύματα αν είναι εγγεγραμμένες σε συγκεκριμένα θέματα. Είναι μια αρχιτεκτονική παρόμοια με τη γνωστή μας client-server.

Συχνά χρησιμοποιείται σε εφαρμογές όπου απαιτείται γρήγορη και αξιόπιστη επικοινωνία με μικρό όγκο δεδομένων, όπως σε αισθητήρες, έξυπνες οικιακές συσκευές και συστήματα παρακολούθησης.[\[11\]](#)

Στη παρακάτω εικόνα 4 βλέπουμε τους αισθητήρες στα αριστερά να κάνουν publish τις τιμές τους στα αντίστοιχα topics στον MQTT Broker. Ακόμα βλέπουμε στα δεξιά ένας actuator να έχει κάνει subscribe στο topic temp παίρνοντας τη τιμή της θερμοκρασίας. Ακόμα επιστρέφει κάνοντας publish τη κατάσταση του “closed”. Κάτω δεξιά βλέπουμε έναν cloud server που αποθηκεύει τις τιμές που παίρνει στο συγκεκριμένο topic “accel-x”, κάνοντας subscribe.



Εικόνα 4 MQTT πρωτόκολλο [\[12\]](#)

1.4.2 CoAP (Constrained Application Protocol)

Το CoAP είναι ένα πρωτόκολλο ειδικά σχεδιασμένο για συσκευές με περιορισμένους πόρους σε δίκτυα IoT. Βασίζεται στο πρωτόκολλο REST (Representational State Transfer) και επιτρέπει τη διαχείριση πόρων μέσω της αρχιτεκτονικής client-server.

Προσφέρει λειτουργίες παρόμοιες με αυτές του HTTP, αλλά με χαμηλότερες απαιτήσεις σε εύρος ζώνης και κατανάλωση ενέργειας. Είναι ιδανικό για χρήση σε μικρά συστήματα όπου είναι σημαντική η εξοικονόμηση ενέργειας. Εφαρμόζεται κυρίως σε συσκευές που χρειάζονται αποδοτική επικοινωνία, όπως σε έξυπνες μετρήσεις, συστήματα αυτοματισμού και παρακολούθησης. [\[13\]](#)

1.4.3 HTTP (HyperText Transfer Protocol)

Το HTTP είναι το βασικό πρωτόκολλο του Παγκόσμιου Ιστού και χρησιμοποιείται ευρέως και στο IoT, κυρίως λόγω της ευκολίας του στη χρήση και της ευρείας υποστήριξης. Αν και δεν είναι τόσο ελαφρύ όσο τα MQTT και CoAP, χρησιμοποιείται σε εφαρμογές όπου η συνδεσιμότητα είναι καλή και οι περιορισμοί πόρων είναι λιγότερο σημαντικοί.

Το HTTP είναι ένα πρωτόκολλο που βασίζεται σε αρχιτεκτονική client-server, όπου ο πελάτης (client) κάνει αιτήσεις σε διακομιστές (servers) για την πρόσβαση σε δεδομένα ή υπηρεσίες. Είναι κατάλληλο για IoT εφαρμογές που απαιτούν αλληλεπίδραση με web-based υπηρεσίες, όπως smart home εφαρμογές που ελέγχονται μέσω web interfaces. [\[14\]](#)

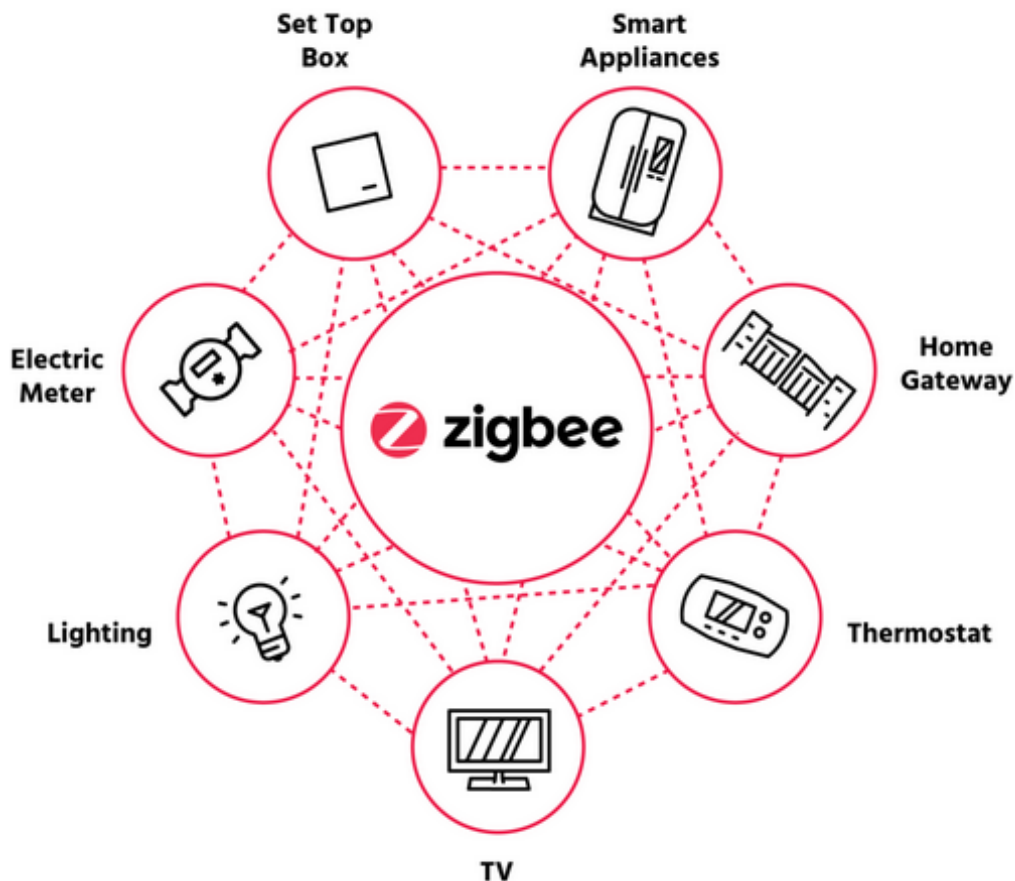
1.4.4 XMPP (Extensible Messaging and Presence Protocol)

Το XMPP είναι ένα πρωτόκολλο επικοινωνίας βασισμένο σε XML, αρχικά σχεδιασμένο για ανταλλαγή μηνυμάτων και πληροφοριών παρουσίας σε πραγματικό χρόνο. Λόγω της επεκτασιμότητας και της ευελιξίας του, έχει προσαρμοστεί και για χρήση στο IoT. Το XMPP ακολουθεί μια αρχιτεκτονική client-server, επιτρέποντας στις συσκευές να συνδέονται σε κεντρικούς διακομιστές για την ανταλλαγή δεδομένων. Χρησιμοποιεί τεχνολογίες όπως το TLS για την ασφάλεια των επικοινωνιών.

Η φύση του XML το καθιστά πιο βαρύ σε σύγκριση με άλλα ελαφριά πρωτόκολλα, όπως το MQTT και το CoAP, γεγονός που μπορεί να αυξήσει τις απαιτήσεις σε εύρος ζώνης και πόρους, ιδιαίτερα σε περιβάλλοντα με περιορισμένες δυνατότητες. Παρόλα αυτά, το XMPP παραμένει μια ισχυρή επιλογή για εφαρμογές που απαιτούν ασφάλεια και επεκτασιμότητα σε πραγματικό χρόνο. [\[15\]](#)

1.4.5 Zigbee

Το Zigbee είναι ένα πρωτόκολλο επικοινωνίας που προορίζεται για χαμηλής ισχύος, μικρής εμβέλειας δίκτυα, και χρησιμοποιείται κυρίως σε συσκευές IoT που απαιτούν χαμηλή κατανάλωση ενέργειας και αξιοπιστία. Το Zigbee λειτουργεί στην μπάντα των 2,4 GHz και είναι ιδανικό για εφαρμογές όπως οικιακός αυτοματισμός, έξυπνοι μετρητές και βιομηχανικός έλεγχος. Το κύριο πλεονέκτημα του Zigbee είναι η δυνατότητά του να δημιουργεί μεγάλα mesh δίκτυα, επιτρέποντας σε πολλές συσκευές να συνδεθούν και να επικοινωνούν μεταξύ τους, ακόμα και σε μεγάλες αποστάσεις μέσω της αναμετάδοσης δεδομένων.[\[16\]](#) Στη παρακάτω εικόνα 5 βλέπουμε ένα παράδειγμα εφαρμογής του πρωτοκόλλου Zigbee σε smart home.



Εικόνα 5 Zigbee on Smart home [\[17\]](#)

1.4.6 Z-wave

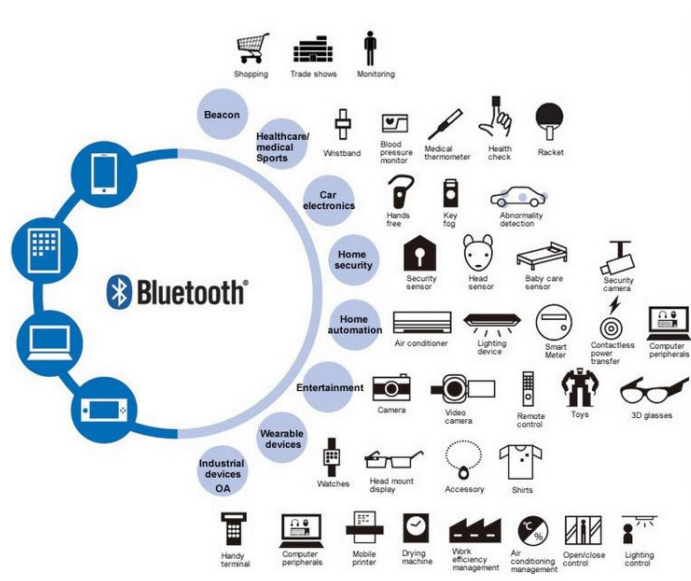
Το Z-Wave είναι παρόμοιο με το Zigbee, αλλά χρησιμοποιεί διαφορετική συχνότητα (κυρίως στα 908 MHz στις ΗΠΑ) και έχει σχεδιαστεί για οικιακές εφαρμογές. Το Z-Wave παρέχει χαμηλότερο εύρος ζώνης από το Zigbee, αλλά αντισταθμίζεται με μεγαλύτερη εμβέλεια και καλύτερη διείσδυση μέσα από τοίχους και εμπόδια, γεγονός που το καθιστά δημοφιλές για εφαρμογές έξυπνου σπιτιού, όπως έλεγχος φωτισμού, ασφάλεια και HVAC.[\[16\]](#) Στην εικόνα 6 βλέπουμε την ιεραρχία των συσκευών σε ένα δίκτυο z-wave σε εφαρμογή ενός smart home.



Εικόνα 6 Δίκτυο z-wave για smart home [18]

1.4.7 Bluetooth

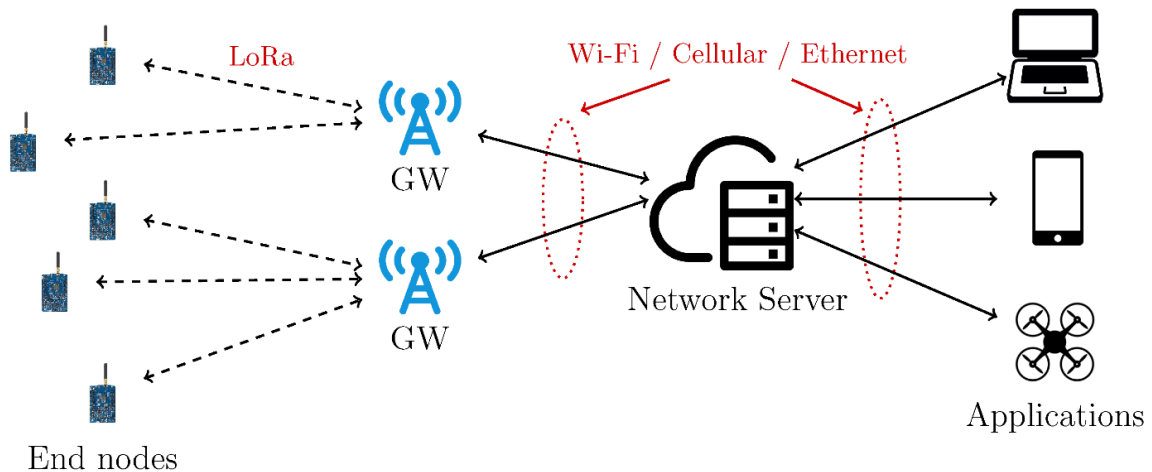
Το Bluetooth είναι ευρέως γνωστό για τη χρήση του σε προσωπικές συσκευές, αλλά έχει επεκταθεί και σε εφαρμογές IoT με την ανάπτυξη του Bluetooth Low Energy (BLE). Το BLE προσφέρει χαμηλή κατανάλωση ενέργειας και είναι κατάλληλο για σύντομες αποστάσεις, όπως wearables, έξυπνες κλειδαριές και συσκευές υγείας. Η πρόσφατη έκδοση Bluetooth 5.0 προσφέρει βελτιώσεις σε ταχύτητα, εμβέλεια και χωρητικότητα δεδομένων, καθιστώντας το ακόμη πιο ευέλικτο για IoT εφαρμογές. [19] Στην εικόνα 7 βλέπουμε κάποιες εφαρμογές του πρωτοκόλλου Bluetooth στο IoT.



Εικόνα 7 Εφαρμογές Bluetooth στο IoT [20]

1.4.8 LoRaWAN

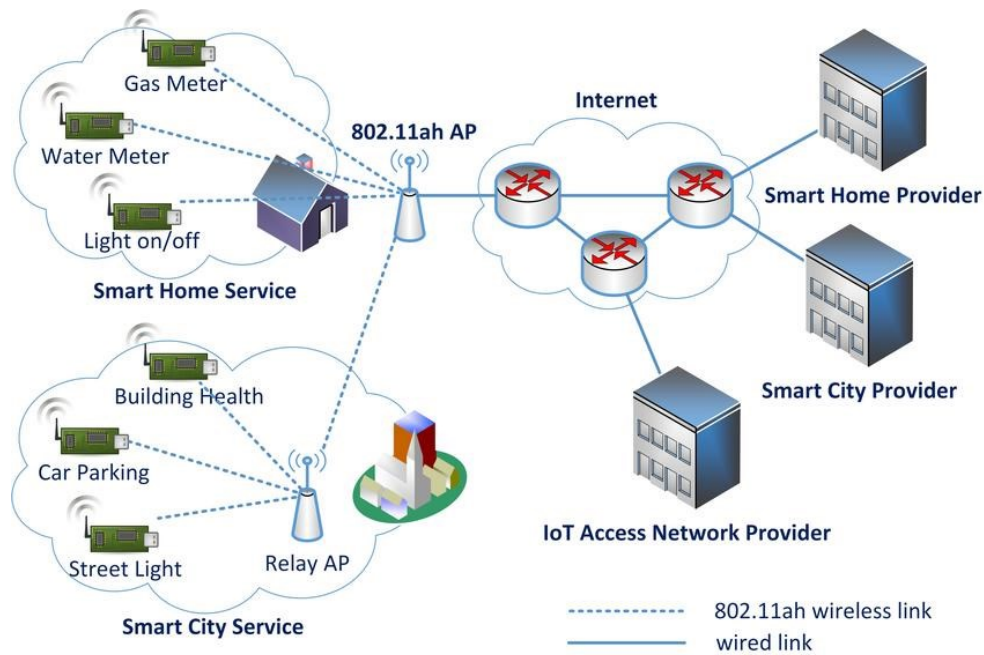
Το LoRaWAN (Long Range Wide Area Network) είναι ένα πρωτόκολλο που χρησιμοποιείται για ασύρματη επικοινωνία χαμηλής ισχύος σε μεγάλα δίκτυα IoT. Είναι ιδανικό για εφαρμογές που απαιτούν μετάδοση δεδομένων σε μεγάλες αποστάσεις με χαμηλή κατανάλωση ενέργειας, όπως έξυπνες πόλεις, γεωργία και περιβαλλοντική παρακολούθηση. Μπορεί να καλύψει μεγάλες αποστάσεις, κάποιες φορές μέχρι και δεκάδες χιλιόμετρα.[\[21\]](#) Στη παρακάτω εικόνα 8 βλέπουμε την αρχιτεκτονική ενός δικτύου LoRaWAN.



Εικόνα 8 Αρχιτεκτονική LoRaWan [\[22\]](#)

1.4.9 IEEE 802.11ah

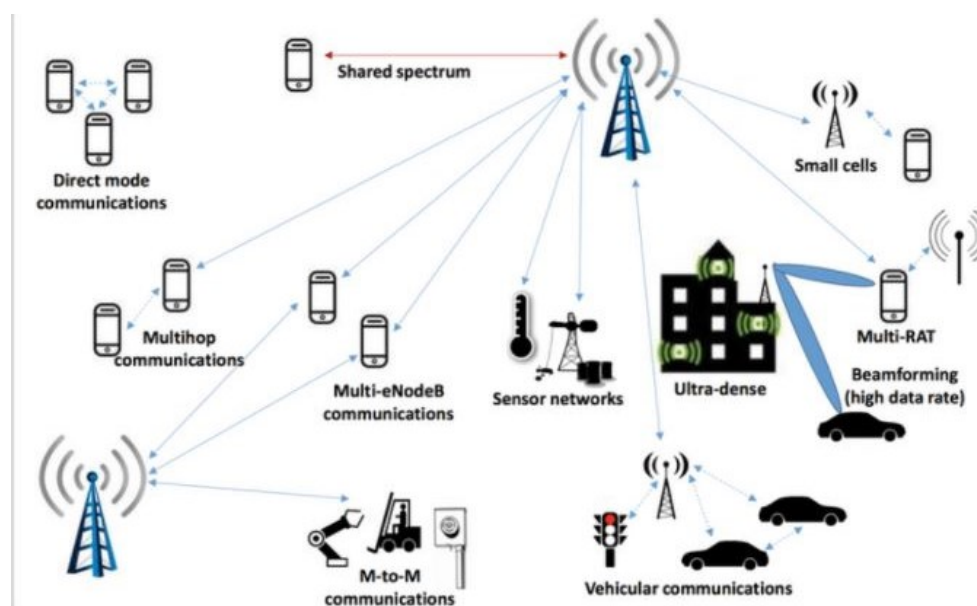
Το IEEE 802.11ah, επίσης γνωστό ως Wi-Fi HaLow, είναι μια επέκταση του προτύπου Wi-Fi (802.11) σχεδιασμένη ειδικά για εφαρμογές IoT. Λειτουργεί στη ζώνη των 900 MHz, η οποία προσφέρει καλύτερη διείσδυση μέσα από εμπόδια και μεγαλύτερη εμβέλεια σε σχέση με τις παραδοσιακές ζώνες συχνοτήτων Wi-Fi (2.4 GHz και 5 GHz). Αυτό το καθιστά ιδανικό για εφαρμογές IoT που απαιτούν χαμηλή κατανάλωση ενέργειας και σύνδεση σε μεγάλες αποστάσεις, όπως σε έξυπνα σπίτια, βιομηχανικούς αισθητήρες και αγροτική παρακολούθηση. Το 802.11ah υποστηρίζει επίσης μεγαλύτερο αριθμό συσκευών ανά σημείο πρόσβασης, κάτι που είναι κρίσιμο για περιβάλλοντα με υψηλή πυκνότητα συσκευών IoT.[\[23\]](#)



Εικόνα 9 Αρχιτεκτονική IEEE 802.11.ah [23]

1.4.10 Cellular (Κυψελοειδής Δίκτυα):

Τα κυψελοειδή δίκτυα, όπως το 4G LTE και το νεότερο 5G, χρησιμοποιούνται στο IoT για εφαρμογές που απαιτούν μεγάλη εμβέλεια και αξιόπιστη σύνδεση σε πραγματικό χρόνο, όπως αυτοκίνητα, drones, και απομακρυσμένη παρακολούθηση. Τα 5G δίκτυα, με την υψηλή ταχύτητα και την ελάχιστη καθυστέρηση, αναμένεται να επιταχύνουν σημαντικά την υιοθέτηση του IoT σε βιομηχανίες όπως η υγεία, η βιομηχανία και οι έξυπνες πόλεις. Το αρνητικό τους στοιχείο είναι η μεγάλη κατανάλωση ενέργειας σε σχέση με τα λιγότερο ενεργοβόρα πρωτόκολλα που είναι σχεδιασμένα για εφαρμογές IoT. [24] Στη παρακάτω εικόνα 10 βλέπουμε ένα δίκτυο 5G το οποίο εξυπηρετεί εφαρμογές IoT.



Εικόνα 10 Εφαρμογές IoT μέσω δικτύου 5G [25]

1.5 Σύγκριση πρωτοκόλλων IoT

Σε αυτό το σημείο θα συγκρίνουμε τα παραπάνω πρωτόκολλα. Στον παρακάτω πίνακα βλέπουμε τη σύγκριση των πρωτοκόλλων εφαρμογής.

Πίνακας 1 Σύγκριση πρωτοκόλλων εφαρμογής του IoT [11, 13-15]

Πρωτόκολλο	Μοντέλο Επικοινωνίας	Υποστήριξη QoS	Μηχανισμός Ασφάλειας	Ιδανική Χρήση
MQTT	Publish-Subscribe	Ναι	TLS/SSL Κρυπτογράφηση	Βιομηχανικός Αυτοματισμός, IoT συσκευές
CoAP	Request-Response	Μέτρια	DTLS για ασφαλή UDP επικοινωνία	Έξυπνες πόλεις, Έξυπνη Γεωργία
HTTP	Request-Response	Όχι	HTTPS (TLS/SSL)	Web Ενσωματώσεις, Διαχείριση συσκευών
XMPP	Ανοιχτή Συνεδρία	Μέτρια	TLS/SSL για ασφαλή συνδέσεις	Πραγματικού χρόνου επικοινωνία, Κοινωνικά δίκτυα

Στον παρακάτω πίνακα βλέπουμε τη σύγκριση των πρωτοκόλλων δικτύου που χρησιμοποιούνται σε εφαρμογές IoT.

Πίνακας 2 Σύγκριση πρωτοκόλλων δικτύου εφαρμογών IoT [26-28]

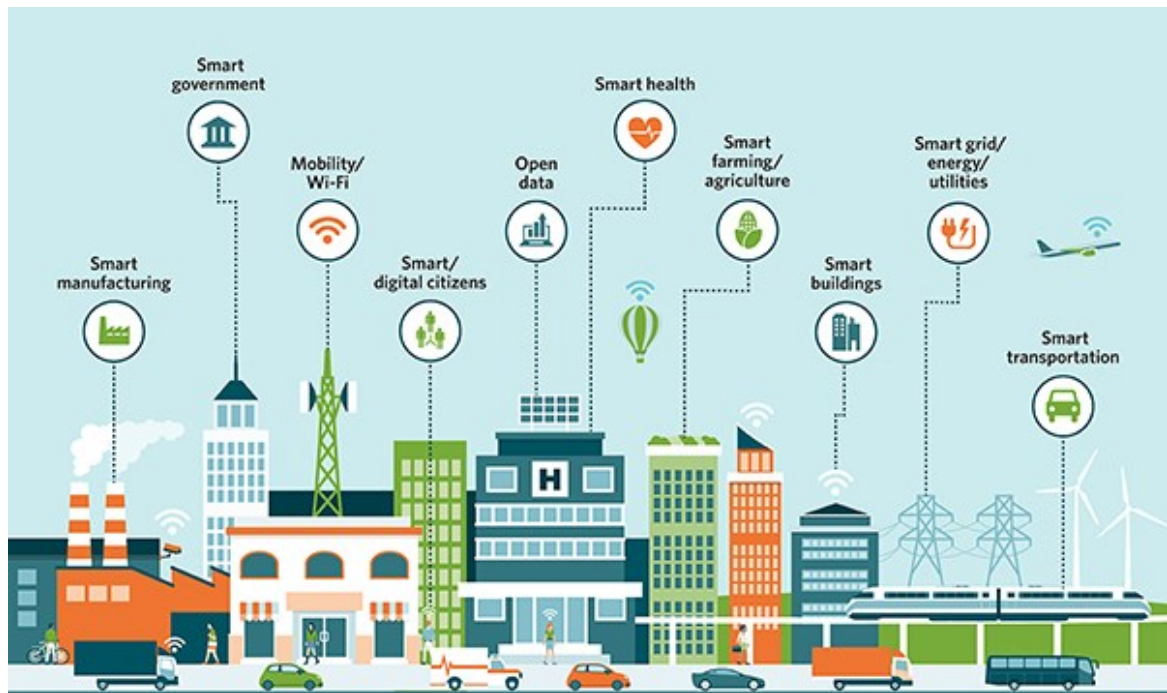
Τεχνολογία	Συχνότητα	Εμβέλεια	Ταχύτητα Δεδομένων	Κατανάλωση Ενέργειας	Προτεινόμενες Εφαρμογές
Zigbee	800 & 900 MHz, 2.4 GHz	Μέχρι 100 μέτρα	250 kbps	Χαμηλή	Έξυπνα σπίτια, αισθητήρες
Z-Wave	908.4 MHz	Μέχρι 100 μέτρα	9.6-100 kbps	Χαμηλή	Οικιακός αυτοματισμός
Bluetooth 5 LE	2.4 GHz	Μέχρι 400 μέτρα	Μέχρι 2 Mbps	Χαμηλή	Wearables, έξυπνες συσκευές
LoRaWAN	868 MHz/915 MHz	Πάνω από 10 km	50 kbps	Πολύ χαμηλή	Έξυπνες πόλεις, γεωργία
IEEE 802.11ah	900 MHz	Μέχρι 1 km	Μέχρι 347 Mbps	Χαμηλή	Έξυπνα σπίτια, βιομηχανικά δίκτυα
Cellular 5G	600-700 MHz ή 1-6 GHz ή 24-40 GHz	Μεγάλη εμβέλεια μεχρι 600m	Μέχρι 20 Gbps	Υψηλή	Κινητά, οχήματα, ευρεία κάλυψη

1.6 Εφαρμογές και Παραδείγματα Χρήσης του IoT

Το IoT αναπτύσσεται συνεχώς και παρατηρείται διεύρυνση του σε διάφορους τομείς. Σε αυτή την ενότητα θα αναφερθούμε σε εφαρμογές της τεχνολογίας IoT.

1.6.1 Έξυπνες Πόλεις (Smart Cities)

Η χρήση αισθητήρων και IoT συσκευών για την παρακολούθηση και τη διαχείριση της κυκλοφορίας, τη βελτίωση της διαχείρισης απορριμμάτων και την αυτοματοποίηση του φωτισμού των δρόμων είναι τα στοιχεία που έχει μια έξυπνη πόλη. Ακόμα υπάρχουν εφαρμογές πρόβλεψης των καιρικών συνθηκών και παρακολούθησης της ποιότητας του περιβάλλοντος. Τα οφέλη της χρήσης της τεχνολογίας IoT στις έξυπνες πόλεις είναι η μείωση του κόστους ενέργειας, η βελτίωση της δημόσιας ασφάλειας και η μείωση των εκπομπών CO₂. [29] Στη παρακάτω εικόνα 11 βλέπουμε εφαρμογές χρήσης IoT τεχνολογίας στα smart cities.



Εικόνα 11 IoT σε έξυπνες πόλεις [30]

1.6.2 Έξυπνα Σπίτια (Smart Homes)

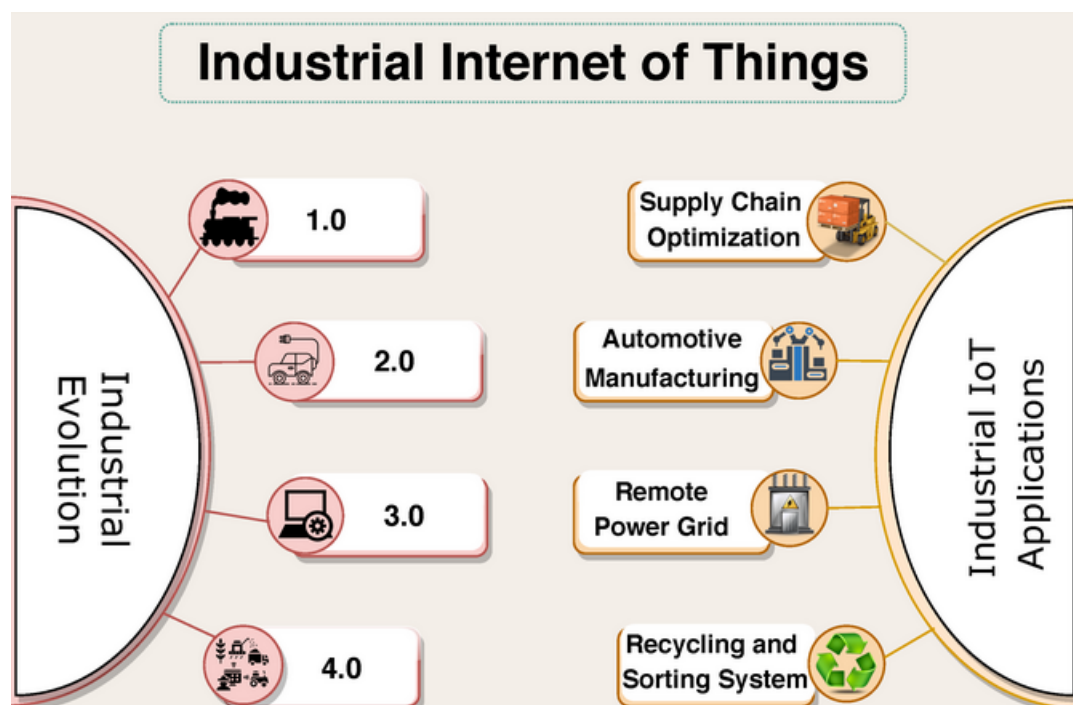
Συσκευές όπως έξυπνοι θερμοστάτες, έξυπνες κλειδαριές, αυτόματος φωτισμός και συστήματα ασφαλείας μπορούν παρέχουν νέες υπηρεσίες. Τα οφέλη που παρέχονται είναι η αυξημένη ασφάλεια, η βελτιωμένη ενεργειακή απόδοση και η αυτοματοποίηση καθημερινών εργασιών, οδηγώντας σε μεγαλύτερη άνεση, εξοικονόμηση χρόνου και πόρων. [31] Στη παρακάτω εικόνα 12 βλέπουμε εφαρμογές χρήσης IoT τεχνολογίας στα smart homes.



Εικόνα 12 IoT σε έξυπνα σπίτια [32]

1.6.3 Βιομηχανικό IoT (IIoT):

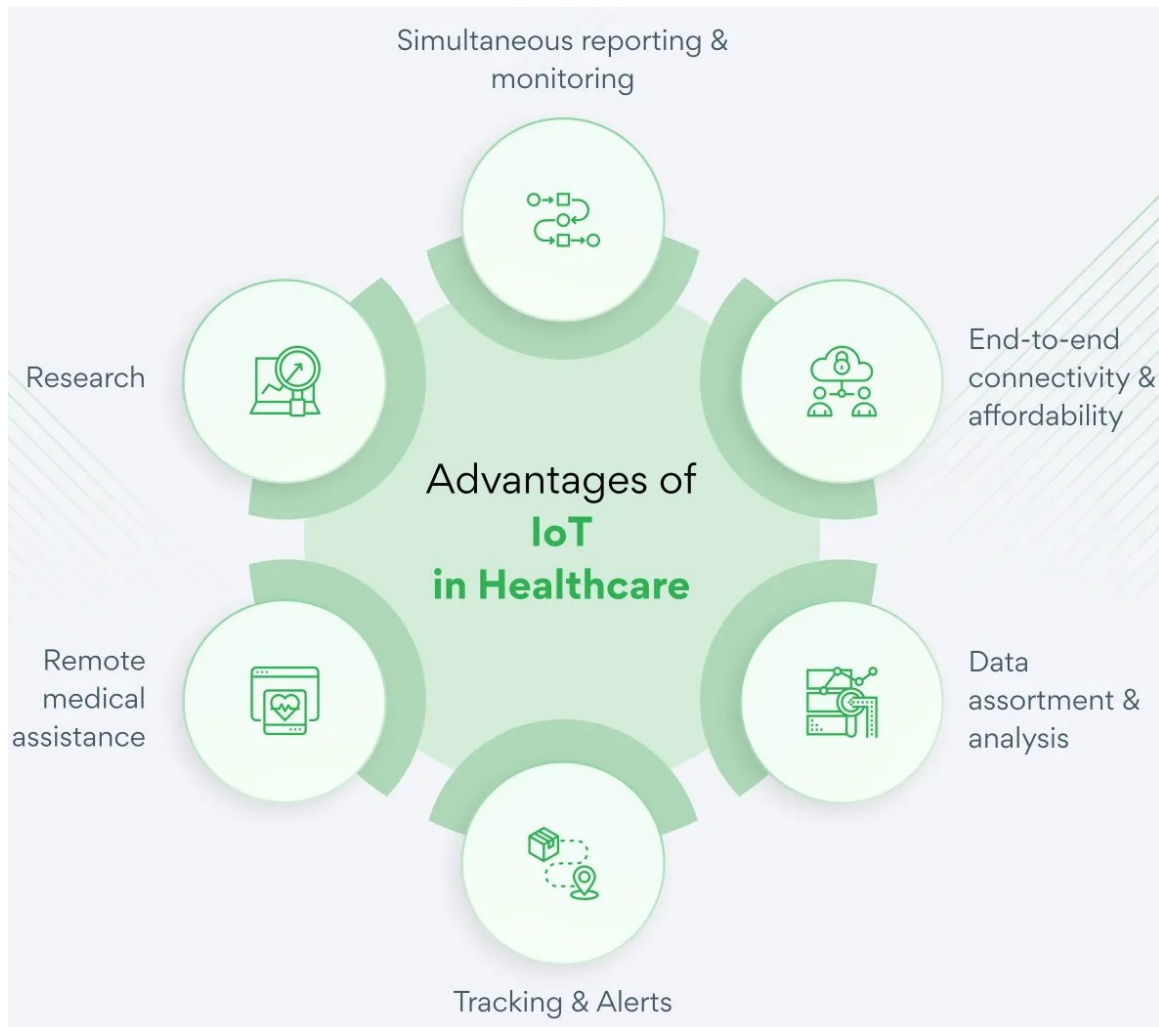
Αισθητήρες και actuators(κινητήρες-ενεργοποιητές) παρέχουν σε πραγματικού χρόνου αυτοματισμούς στην παραγωγική διαδικασία, επιτρέποντας την ταχύτερη αντίδραση σε προβλήματα και την βελτιστοποίηση των εργασιών. Η βελτίωση της αποδοτικότητας, η μείωση του κόστους παραγωγής, και η αποτελεσματική συντήρηση είναι κάποια από τα οφέλη των εφαρμογών IIoT στη βιομηχανία. [33] Στη παρακάτω εικόνα 13 βλέπουμε την εξέλιξη του IIoT με τις αντίστοιχες εφαρμογές.



Εικόνα 13 Η εξέλιξη του IIoT [34]

1.6.4 Υγειονομική Περίθαλψη(Healthcare)

Συσκευές που φορούν χρήστες παρακολουθούν τα βιομετρικά τους δεδομένα, όπως ο ρυθμός της καρδιάς, η αρτηριακή πίεση και το ποσοστό οξυγόνου. Έτσι επιτρέπεται η καλύτερη εποπτεία ασθενών απομακρυσμένα. Η ικανότητα για προληπτική φροντίδα, η βελτιωμένη διαχείριση της υγείας των ασθενών και μείωση του κόστους υγειονομικής φροντίδας είναι μερικά από τα οφέλη της χρησιμοποίησης της τεχνολογίας IoT σε αυτόν το τομέα. [35] Στη παρακάτω εικόνα 14 βλέπουμε τα οφέλη των εφαρμογών της τεχνολογίας IoT στον τομέα healthcare.

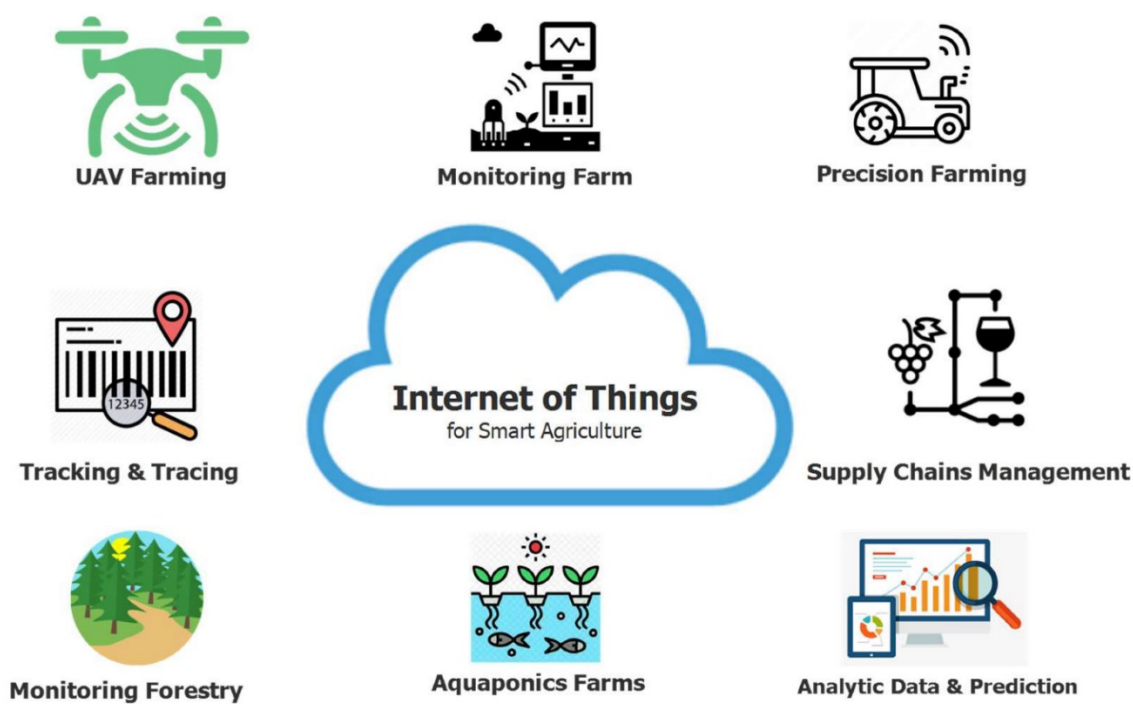


Εικόνα 14 Οφέλη χρήσης IoT στον τομέα healthcare [36]

1.6.5 Γεωργία(Agricultural IoT)

Οι εφαρμογές IoT στη γεωργία (Agricultural IoT) επιτρέπουν στους αγρότες να αυξήσουν την αποδοτικότητα, να μειώσουν το κόστος και να βελτιώσουν την παραγωγικότητα των καλλιεργειών τους μέσω προηγμένων τεχνολογιών. Χρησιμοποιώντας αισθητήρες και δίκτυα επικοινωνίας, το IoT μπορεί να παρέχει πραγματικού χρόνου δεδομένα για παράγοντες όπως η υγρασία του εδάφους, η θερμοκρασία, η επίπεδα θρεπτικών ουσιών και η υγεία των φυτών. Αυτά τα δεδομένα επιτρέπουν την ακριβή διαχείριση των πόρων, όπως νερό και λιπάσματα,

και βοηθούν στην πρόληψη των ασθενειών και την αύξηση της ποιότητας και ποσότητας της σοδειάς. Το IoT επίσης ενισχύει τη βιωσιμότητα μέσω της εξοικονόμησης πόρων και της βελτιστοποίησης της αγροτικής διαδικασίας, παρέχοντας αποδοτικότητα και προσαρμοσμένη προσέγγιση στη σύγχρονη γεωργία. [37] στην εικόνα 15 βλέπουμε τα οφέλη των εφαρμογών της τεχνολογίας IoT στον τομέα Agriculture.



Εικόνα 15 Οφέλη του Agricultural IoT [37]

1.7 Προκλήσεις και προοπτικές στον Τομέα του IoT

Το Internet of Things αντιμετωπίζει σημαντικές **προκλήσεις** που μπορεί να επηρεάσουν την ευρύτερη καθιέρωση του. Κυρίαρχο θέμα αποτελεί η ασφάλεια και η προστασία της ιδιωτικότητας, καθώς οι συσκευές IoT συλλέγουν και μεταδίδουν ευαίσθητα δεδομένα, τα οποία μπορούν να γίνουν στόχος κυβερνοεπιθέσεων. Επιπλέον, η διαλειτουργικότητα μεταξύ διαφορετικών συσκευών και πλατφορμών παραμένει ένα σημαντικό εμπόδιο, καθώς η έλλειψη ενιαίων προτύπων μπορεί να περιορίσει την αποτελεσματική επικοινωνία. Ακόμα η διαχείριση του τεράστιου όγκου των δεδομένων που παράγουν οι IoT συσκευές απαιτεί προηγμένα συστήματα ανάλυσης και αποθήκευσης, αυξάνοντας την πολυπλοκότητα και το κόστος των IoT λύσεων.

Παρά τις προκλήσεις, οι **προοπτικές** του IoT είναι εντυπωσιακές και διαρκώς εξελίσσονται. Η εισαγωγή της τεχνολογίας 5G υπόσχεται να επιταχύνει την ανάπτυξη του IoT, προσφέροντας αυξημένες ταχύτητες, μειωμένη καθυστέρηση και βελτιωμένη συνδεσιμότητα. Η τεχνητή νοημοσύνη και η μηχανική μάθηση παίζουν κεντρικό ρόλο, επιτρέποντας την αυτοματοποίηση και την ανάλυση των δεδομένων. Αυτές οι τεχνολογίες βοηθούν στην αποτελεσματική συντήρηση και τη βελτίωση της απόδοσης των συστημάτων. Έτσι όλοι οι παραπάνω τομείς του IoT που αναφέρθηκαν γίνονται αποδοτικότεροι.

2. ΠΑΡΑΚΟΛΟΥΘΗΣΗ ΠΕΡΙΒΑΛΛΟΝΤΟΣ ΓΡΑΦΕΙΟΥ

Μια ακόμα εφαρμογή της τεχνολογίας IoT είναι η παρακολούθηση του περιβάλλοντος ενός χώρου. Στη συγκεκριμένη εργασία θα υλοποιηθεί ένα σύστημα παρακολούθησης πολλών χώρων ταυτόχρονα. Αρχικά θα καταγράψουμε τα επίπεδα για κάθε ένα από τα στοιχεία που θα παρακολουθούμε.[\[38\]](#) Οι παράμετροι που θα καταγράφονται είναι :

1. Θερμοκρασία
2. Υγρασία
3. Ποιότητα αέρα
4. Ακουστικός θόρυβος
5. Φωτεινότητα

Με την επεξεργασία των δεδομένων που θα συλλέγονται, θα αξιολογείται σε πραγματικό χρόνο το περιβάλλον ενός γραφείου και θα συγκρίνεται με ένα άλλο γραφείο. Συνολικά θα μπορούμε να παρακολουθούμε 3 ταυτόχρονα περιβάλλοντα. Παρακάτω θα καταγράψουμε τα επίπεδα αξιολόγησης για τις 5 παραμέτρους. Θα γίνει βαθμονόμηση τους από το 1εως το 5.

2.1 Επίπεδα αξιολόγησης γραφείου

Παρακάτω θα αναλύσουμε αυτές τις παραμέτρους και θα ορίσουμε κατώφλια που θα μας βοηθήσουν στην αξιολόγηση του περιβάλλοντος γραφείου.

2.1.1 Θερμοκρασία

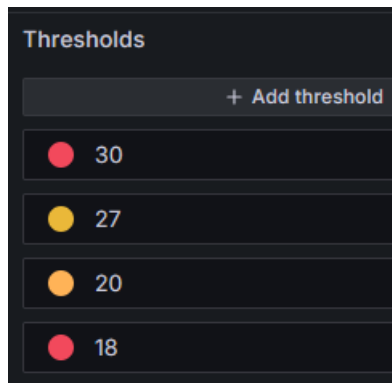
Ξεκινώντας από τη **θερμοκρασία** έχουμε 5 κατώφλια :

1. Πολύ κακό: Θερμοκρασίες πάνω από 30°C θεωρούνται δυσάρεστες και μπορούν να επηρεάσουν την απόδοση και την ευεξία των εργαζομένων. Εδώ εντάσσουμε τις θερμοκρασίες που ξεπερνούν τους 30°C.
2. Κακό: Θερμοκρασίες από 27°C έως 30°C. Σε αυτό το εύρος, οι εργαζόμενοι αρχίζουν να νιώθουν δυσφορία, και η παραγωγικότητα μπορεί να μειωθεί.
3. Μέτριο: Θερμοκρασίες από 24.5°C έως 27°C. Αυτές οι θερμοκρασίες θεωρούνται ανεκτές, αλλά μπορεί να μην προσφέρουν την ιδανική άνεση για όλους.
4. Καλό: Θερμοκρασίες από 22°C έως 24.5°C. Αυτό το εύρος θεωρείται το βέλτιστο για τη θερμική άνεση σε ένα γραφείο, σύμφωνα με τα πρότυπα της ASHRAE.
5. Πολύ καλό: Θερμοκρασίες από 20°C έως 22°C. Αυτές οι θερμοκρασίες θεωρούνται πολύ άνετες, ειδικά για τους εργαζόμενους που ντύνονται με βαρύτερα ρούχα τον χειμώνα.[\[39-40\]](#)

Έτσι η αντιστοίχιση με τους βαθμούς αξιολόγησης είναι η παρακάτω. Αυτή η αντιστοίχιση θα μας βοηθήσει όταν θα προγραμματίσουμε το λογισμικό ανάλυσης και οπτικοποίησης των δεδομένων.

- 0-17°C → 1: Πολύ κακή (Άβολη και πολύ κρύα για τα περισσότερα περιβάλλοντα γραφείου)
- 18-20°C → 3: Μέτρια (Αποδεκτό, αλλά μπορεί να αισθάνεται κρύο)
- 21-22°C → 5: Πολύ καλό (Ιδανικό για άνεση και παραγωγικότητα)
- 23-24°C → 4: Καλό (Εξακολουθεί να είναι άνετο για τους περισσότερους ανθρώπους)
- 25-27°C → 3: Μέτρια (Ζεστή, αλλά διαχειρίσιμη)
- 28-30°C → 2: Κακή (Αρχίζει να γίνεται άβολη και μπορεί να μειώσει την παραγωγικότητα)
- 30+°C → 1: Πολύ κακή (Πολύ υψηλή θερμοκρασία, άβολη για τους περισσότερους ανθρώπους)

Στη παρακάτω εικόνα 16 βλέπουμε τα αντίστοιχα κατώφλια για τη θερμοκρασία όπως τα ορίσαμε στο λογισμικό παρουσίασης των δεδομένων Grafana.



Εικόνα 16 Τα κατώφλια θερμοκρασίας στο Grafana

2.1.2 Υγρασία

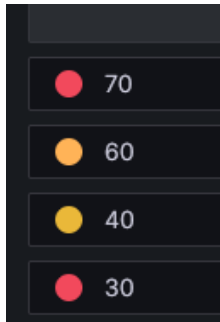
Συνεχίζουμε με την **υγρασία**. Έχουμε λοιπόν 5 κατώφλια :

1. 0-29%: Πολύ ξηρό, πιθανό να προκαλέσει δυσφορία, αναπνευστικά προβλήματα και αυξημένο στατικό ηλεκτρισμό.
2. 30-39%: Κάτω από το βέλτιστο, αλλά αποδεκτό. Μπορεί ακόμα να προκαλέσει ξηρότητα σε ορισμένα άτομα.
3. 40-50%: Ιδανικό εύρος. Σε αυτό το εύρος έχουμε ευεξία και παραγωγικότητα.
4. 51-60%: Εξακολουθεί να είναι άνετο, αλλά η υψηλότερη υγρασία μπορεί να αρχίσει να προκαλεί δυσφορία ή ήπια προβλήματα.
5. 61%+: Υπερβολική υγρασία, που οδηγεί σε δυσφορία, πιθανή ανάπτυξη μούχλας και προβλήματα με την ποιότητα του αέρα. [\[41-42\]](#)

Για την υγρασία λοιπόν έχουμε την παρακάτω αντιστοίχιση των κατωφλίων. Με αυτά τα όρια θα γράψουμε τον κώδικα ελέγχου της υγρασίας.

- < 30%: Πολύ χαμηλή υγρασία (1,0) - δυσάρεστη, πολύ ξηρή.
- 30-39%: Χαμηλή υγρασία (2,0) - κάτω από τη βέλτιστη, αλλά ακόμα αποδεκτή.
- 40-50%: Βέλτιστη υγρασία (5,0) - ιδανική για άνεση και παραγωγικότητα.
- 51-60%: Υψηλότερη υγρασία (4,0) - εξακολουθεί να είναι άνετη αλλά λιγότερο ιδανική.
- 61-70%: Υψηλότερη υγρασία (3,0) .
- > 70%: Υπερβολική υγρασία (1,0) - πιθανόν να προκαλέσει δυσφορία και προβλήματα ποιότητας του αέρα.

Στη παρακάτω εικόνα 17 βλέπουμε τα αντίστοιχα κατώφλια για τη θερμοκρασία όπως τα ορίσαμε στο λογισμικό παρουσίασης των δεδομένων Grafana.



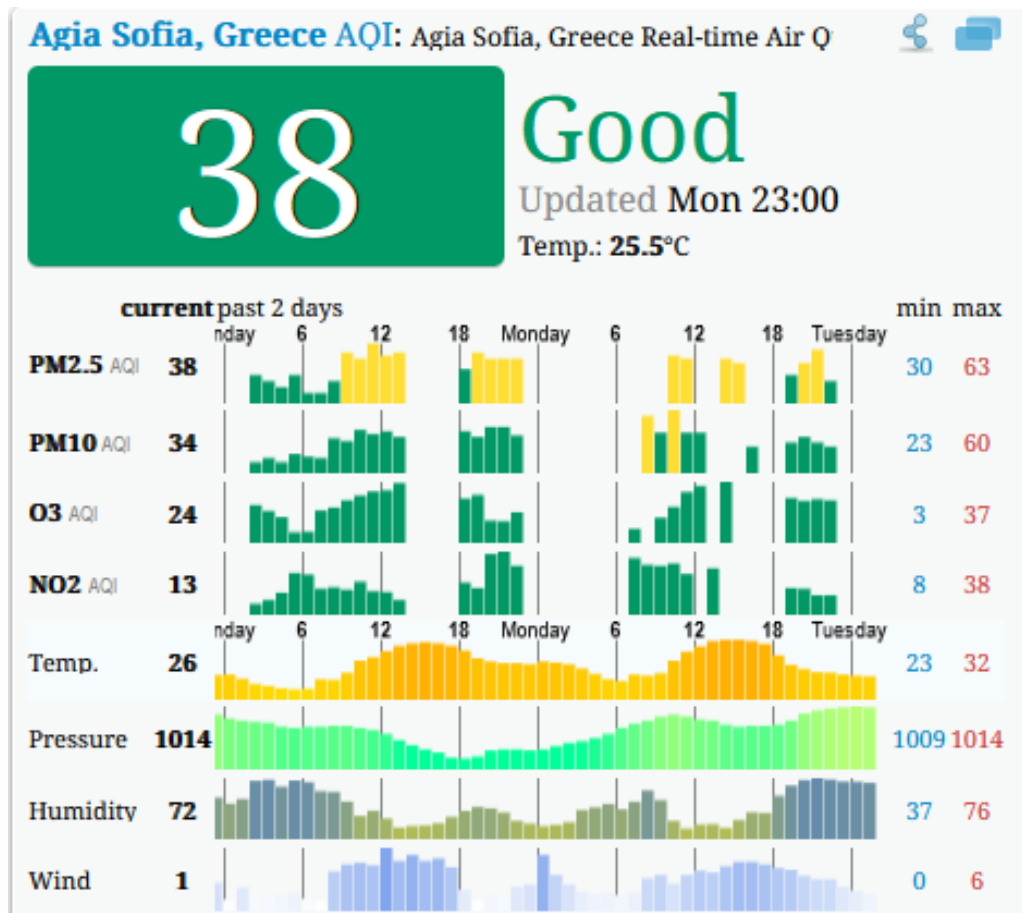
Εικόνα 17 Τα κατώφλια υγρασίας στο Grafana

2.1.3 Ποιότητα του αέρα

Η επόμενη παράμετρος είναι η **ποιότητα του αέρα**. Η βιβλιογραφία αναφέρει το AIR QUALITY INDEX(AQI). Μια βαθμολογία δηλαδή που προκύπτει από τη μέτρηση διαφορετικών παραγόντων. Για τους **εξωτερικούς χώρους** οι παράμετροι που έχουν επικρατήσει και έχουν καθοριστική συνεισφορά στη ποιότητα του αέρα είναι :

1. Μικροσωματίδια (PM2.5 και PM10): Αυτά είναι μικρά σωματίδια ή σταγονίδια στον αέρα που μπορεί να εισπνευστούν και να προκαλέσουν προβλήματα στην αναπνευστική υγεία.
2. Διοξείδιο του θείου (SO₂): Προέρχεται κυρίως από την καύση ορυκτών καυσίμων σε εργοστάσια και οχήματα, και μπορεί να προκαλέσει προβλήματα στους πνεύμονες και να επηρεάσει τους ασθματικούς.
3. Διοξείδιο του αζώτου (NO₂): Εκπέμπεται από κινητήρες οχημάτων και εγκαταστάσεις καύσης καυσίμων και μπορεί να επηρεάσει την αναπνευστική υγεία.
4. Όζον (O₃): Το όζον στο επίπεδο του εδάφους δημιουργείται όταν οι εκπομπές από βιομηχανίες και οχήματα αντιδρούν με τη ηλιακή ακτινοβολία. Είναι ερεθιστικό για τους πνεύμονες και μπορεί να προκαλέσει ή να επιδεινώσει αναπνευστικά προβλήματα.
5. Η πίεση ατμόσφαιρας (Pressure) έχει έμμεση σημασία γιατί επηρεάζει τις καιρικές συνθήκες και μπορεί να διαμορφώσει τη συγκέντρωση και την κατανομή ατμοσφαιρικών ρύπων. Μετρείται συνήθως σε millibars ή inches of mercury και είναι κρίσιμη για την κατανόηση και την πρόβλεψη μεταβολών στον καιρό.[\[43-45\]](#)

Στην παρακάτω εικόνα βλέπουμε το AQI από τον μετεωρολογικό σταθμό που βρίσκεται πλησίον της εκκλησίας της Αγίας σοφίας στη πόλη της Θεσσαλονίκης.



Εικόνα 18 AQI στην Αγία Σοφία [45]

Για τους **εσωτερικούς χώρους** όμως το AQI προκύπτει από άλλες παραμέτρους. Οι συνηθέστεροι παράμετροι που προκύπτουν απτή βιβλιογραφία είναι :

1. Μικροσωματίδια (PM2.5 και PM10): Τα μικροσωματίδια μπορούν να προέρχονται από εξωτερικές πηγές αλλά και από εσωτερικές δραστηριότητες όπως εκτυπώσεις, μαγείρεμα και καπνός.
2. Διοξείδιο του άνθρακα (CO2): Υψηλά επίπεδα CO2 μπορεί να δείχνουν κακή αερισμό και είναι σημαντικός δείκτης για την ανάγκη βελτίωσης της ποιότητας αέρα.
3. Τοξικές αέρια εκπομπές (VOCs, Volatile Organic Compounds): Πηγές VOCs περιλαμβάνουν γραφική ύλη, καθαριστικά, αποσμητικά χώρου και άλλα υλικά που εκλύουν αέρια.
4. Υγρασία: Η υγρασία πρέπει να διατηρείται σε κατάλληλα επίπεδα για να αποφεύγεται η ανάπτυξη μούχλας και να διασφαλίζεται άνετη αναπνοή.
5. Θερμοκρασία. Υψηλές θερμοκρασίες μπορούν να επιταχύνουν τις χημικές αντιδράσεις που παράγουν όζον στην ατμόσφαιρα. Η συνεχής παρακολούθηση της θερμοκρασίας είναι σημαντική για την ακριβή πρόβλεψη και διαχείριση της ποιότητας του αέρα. [46-48]

Μετά από ερευνά στην αγορά(2024), οι πιο οικονομικού αισθητήρες για τις παραπάνω παραμέτρους φαίνονται στον παρακάτω πίνακα 3.

Πίνακας 3 Αισθητήρες μέτρησης ποιότητας αέρα

A/A	Μοντέλο Αισθητήρα	Παράμετρος μέτρησης	Τιμή (€)
1	SGP30	VOCs	25
2	SDS-011	PM2.5, PM10	50
3	SEN0471	NO2	200
4	Gravity Sensor SEN0470	SO4	200
5	O3 12C SENO321	OZONE	100
6	MQ-135	CO2, CO, alcohol, acetone, toluene, NH4	7

Στη συγκεκριμένη εργασία θα υλοποιούν 3 σταθμοί μέτρησης παραμέτρων. Το κόστος λοιπόν για τον κάθε σταθμό είναι υψηλό και για αυτό το λόγο θα χρησιμοποιηθεί ο αισθητήρας MQ135 για τη μέτρηση του διοξειδίου του άνθρακα(CO2).

Για τη παράμετρο λοιπόν του **CO2** έχουμε 5 κατώφλια :

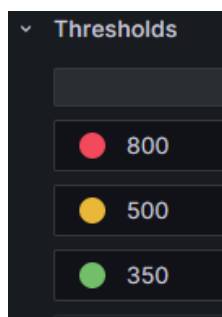
1. Επίπεδο CO2 < 350 ppm (Εξαιρετικό): Τα επίπεδα CO2 κάτω από 350 ppm υποδεικνύουν εξαιρετική ποιότητα αέρα. Η συγκέντρωση CO2 είναι κοντά στη φυσική τιμή του εξωτερικού αέρα (~400 ppm) και δείχνει ότι ο αερισμός είναι επαρκής. Οι συνθήκες είναι ιδανικές για την ευημερία και την παραγωγικότητα των ανθρώπων, με ελάχιστο κίνδυνο επιπτώσεων στην υγεία ή τη γνωστική απόδοση.
2. Επίπεδο CO2 351 - 500 ppm (Πολύ Καλό): Στο εύρος των 351 έως 500 ppm, η ποιότητα του αέρα είναι πολύ καλή και δείχνει επαρκή αερισμό και χαμηλή συγκέντρωση CO2. Αυτά τα επίπεδα είναι σχεδόν ιδανικά για εσωτερικούς χώρους και δεν αναμένονται σοβαρές επιπτώσεις στην υγεία ή την άνεση. Ωστόσο, εάν η συγκέντρωση CO2 διατηρηθεί σε αυτό το επίπεδο για μακροχρόνιο διάστημα, ίσως είναι χρήσιμος ο περιοδικός αερισμός.
3. Επίπεδο CO2 501 - 800 ppm (Μέτριο): Στα 501 έως 800 ppm, η ποιότητα του αέρα θεωρείται μέτρια. Αυτή η συγκέντρωση CO2 μπορεί να αρχίσει να επηρεάζει ελαφρώς την ευημερία και την απόδοση, με πιθανά συμπτώματα όπως ήπια κόπωση ή μείωση της συγκέντρωσης. Συνιστάται η αύξηση του αερισμού για να διατηρηθεί η ποιότητα του αέρα σε άνετα επίπεδα.
4. Επίπεδο CO2 801 - 1000 ppm (Κακό): Επίπεδα CO2 μεταξύ 801 και 1000 ppm δείχνουν κακή ποιότητα αέρα και πιθανές αρνητικές επιπτώσεις στην υγεία και την ευημερία. Στα επίπεδα αυτά, μπορεί να παρατηρήσετε πονοκεφάλους, κόπωση και μειωμένη συγκέντρωση. Η βελτίωση του αερισμού είναι απαραίτητη για να αποφευχθεί η υποβάθμιση της ποιότητας του αέρα και να διατηρηθεί η άνεση του χώρου.
5. Επίπεδο CO2 > 1000 ppm (Πολύ Κακό): Όταν η συγκέντρωση CO2 ξεπερνά τα 1000 ppm, η ποιότητα του αέρα είναι πολύ κακή και ενδέχεται να προκαλέσει σοβαρές επιπτώσεις στην υγεία, όπως πονοκεφάλους, υπνηλία και σημαντική μείωση της

γνωστικής απόδοσης. Αυτό το επίπεδο CO₂ δείχνει ότι ο αερισμός είναι ανεπαρκής και απαιτείται άμεση αύξηση του αερισμού για να αποκατασταθεί η ποιότητα του αέρα και να προστατευτεί η υγεία των ανθρώπων στον χώρο.[\[49-51\]](#)

Αυτές οι περιγραφές παρέχουν μια σαφή εικόνα των συνθηκών ποιότητας αέρα και των σχετικών επιπτώσεων με βάση τα επίπεδα CO₂. Παρακάτω θα καταγράψουμε την αντιστοίχιση των κατώφλιων. Με αυτά τα όρια θα γράψουμε τον κώδικα ελέγχου της ποιότητας του αέρα.

- < 350 ppm: Βαθμολογία 5 (Εξαιρετικό)
- 351 - 500 ppm: Βαθμολογία 4 (Πολύ Καλό)
- 501 - 800 ppm: Βαθμολογία 3 (Μέτριο)
- 801 - 1000 ppm: Βαθμολογία 2 (Κακό)
- > 1000 ppm: Βαθμολογία 1 (Πολύ Κακό)

Στη παρακάτω εικόνα 19 βλέπουμε τα αντίστοιχα κατώφλια για τη ποιότητα του αέρα όπως τα ορίσαμε στο λογισμικό παρουσίασης των δεδομένων Grafana. Μπορούμε να επιλέξουμε το χρώμα και το είδος της γραμμής διαχωρισμού του ορίου της παραμέτρου που μετράμε.



Εικόνα 19 Κατώφλια ποιότητας του αέρα στο Grafana

2.1.4 Φωτεινότητα(Luminosity)

Η φωτεινότητα θα μετρηθεί σε ποσοστό %. 0% είναι το απολυτό σκοτάδι και 100% είναι το πολύ δυνατό φως. Για τη φωτεινότητα έχουμε τα 5 κατώφλια :

1. Ιδανικός Φωτισμός, Επίπεδο Φωτισμού: 40% - 60% του μέγιστου φωτισμού. Αυτό το επίπεδο φωτισμού είναι ιδανικό για εργασία σε γραφείο. Παρέχει αρκετή φωτεινότητα για ανάγνωση, εργασία σε υπολογιστή και άλλες δραστηριότητες χωρίς καταπόνηση των ματιών. Διασφαλίζει άνεση και υψηλή παραγωγικότητα.
2. Πολύ Καλός Φωτισμός, Επίπεδο Φωτισμού: 30% - 39% ή 61% - 70% του μέγιστου φωτισμού. Πολύ καλός φωτισμός για τις περισσότερες εργασίες γραφείου. Μπορεί να είναι ελαφρώς χαμηλότερος ή υψηλότερος από το ιδανικό, αλλά γενικά διατηρείται η άνεση και η παραγωγικότητα χωρίς να προκαλείται σημαντική καταπόνηση των ματιών.
3. Μέτριος Φωτισμός, Επίπεδο Φωτισμού: 20% - 29% ή 71% - 80% του μέγιστου φωτισμού. Μέτριος φωτισμός που είναι αποδεκτός για σύντομες περιόδους εργασίας. Ωστόσο, είτε πολύ χαμηλός είτε πολύ υψηλός, μπορεί να προκαλέσει καταπόνηση των ματιών και ενδεχομένως να μειώσει την παραγωγικότητα με την πάροδο του χρόνου.
4. Κακός Φωτισμός, Επίπεδο Φωτισμού: 10% - 19% ή 81% - 90% του μέγιστου φωτισμού. Κακός φωτισμός που μπορεί να οδηγήσει σε καταπόνηση των ματιών, πονοκεφάλους και

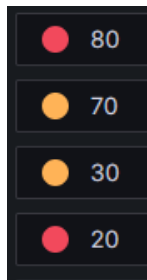
δυσφορία κατά την εργασία. Ανεπαρκής για μακροχρόνια εργασία σε γραφείο, είτε λόγω χαμηλού φωτισμού είτε λόγω υπερβολικής φωτεινότητας.

5. Πολύ Κακός Φωτισμός, Επίπεδο Φωτισμού: 0% - 9% ή 91% - 100% του μέγιστου φωτισμού. Πολύ κακός φωτισμός, είτε λόγω υπερβολικής φωτεινότητας (υπερβολική αντανάκλαση, θάμπωση) είτε λόγω σχεδόν πλήρους σκοταδιού. Αυτό το επίπεδο φωτισμού είναι ακατάλληλο για οποιαδήποτε εργασία γραφείου και απαιτεί άμεση προσαρμογή.[\[52-53\]](#)

Για την φωτεινότητα λοιπόν έχουμε την παρακάτω αντιστοίχιση των κατώφλιων

- Βαθμολογία 5: 40% - 60%
- Βαθμολογία 4: 30% - 39% ή 61% - 70%
- Βαθμολογία 3: 20% - 29% ή 71% - 80%
- Βαθμολογία 2: 10% - 19% ή 81% - 90%
- Βαθμολογία 1: 0% - 9% ή 91% - 100%

Στη παρακάτω εικόνα 20 βλέπουμε τα αντίστοιχα κατώφλια για τη φωτεινότητα όπως τα ορίσαμε στο λογισμικό παρουσίασης των δεδομένων Grafana



Εικόνα 20 Κατώφλια φωτεινότητας στο Grafana

2.1.4 Ακουστικός θόρυβος(Noise dbA)

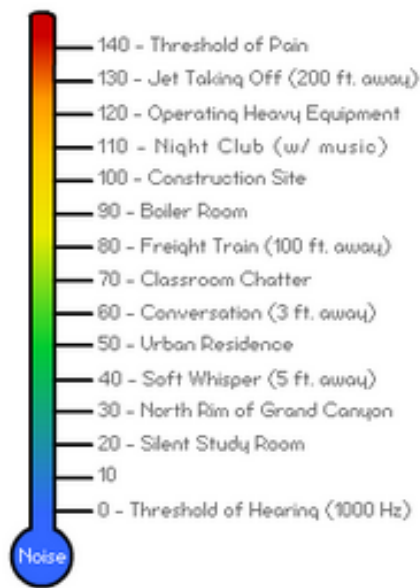
Η κατάλληλη στάθμη θορύβου σε περιβάλλον γραφείου είναι κρίσιμη για την άνεση και την παραγωγικότητα των εργαζομένων. Ο θόρυβος μπορεί να επηρεάσει αρνητικά την ικανότητα συγκέντρωσης και να προκαλέσει στρες. Τα επίπεδα θορύβου σε dB(A) είναι τα παρακάτω 5:

1. Ιδανικά Επίπεδα Θορύβου, Επίπεδα Θορύβου: 30 - 40 dB(A). Το ιδανικό επίπεδο θορύβου για ένα γραφείο. Σε αυτό το επίπεδο, ο θόρυβος είναι ελάχιστος και δεν προκαλεί καμία ενόχληση ή διάσπαση προσοχής, εξασφαλίζοντας μέγιστη παραγωγικότητα και άνεση.
2. Πολύ Καλό Επίπεδο Θορύβου, Επίπεδα Θορύβου: 41 - 50 dB(A). Πολύ καλό επίπεδο θορύβου που είναι αποδεκτό για τις περισσότερες εργασίες γραφείου. Μπορεί να υπάρξουν μικρές διαταραχές, αλλά δεν επηρεάζει σημαντικά την παραγωγικότητα.
3. Μέτριο Επίπεδο Θορύβου, Επίπεδα Θορύβου: 51 - 60 dB(A). Μέτριο επίπεδο θορύβου που μπορεί να είναι αποδεκτό για κάποιες δραστηριότητες, αλλά σε μακροχρόνια βάση μπορεί να προκαλέσει μείωση της συγκέντρωσης και της παραγωγικότητας.

4. Κακό Επίπεδο Θορύβου, Επίπεδα Θορύβου: 61 - 70 dB(A). Κακό επίπεδο θορύβου που προκαλεί συχνές διαταραχές, δυσκολία συγκέντρωσης και αυξημένο στρες στους εργαζομένους. Ακατάλληλο για συνεχή εργασία σε περιβάλλον γραφείου.
5. Πολύ Κακό Επίπεδο Θορύβου, Επίπεδα Θορύβου: >70 dB(A). Πολύ υψηλά επίπεδα θορύβου που καθιστούν το περιβάλλον εργασίας εξαιρετικά δυσάρεστο. Ο θόρυβος σε αυτό το επίπεδο μπορεί να προκαλέσει σημαντική μείωση της παραγωγικότητας, δυσφορία και σοβαρό στρες.[\[54-55\]](#)

Στην παρακάτω εικόνα 21 βλέπουμε κάποια προτεινόμενα επίπεδα θορύβου.

Typical Sound Levels (dBA)

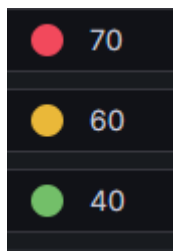


Εικόνα 21 Επίπεδα ακουστικού θορύβου [\[54\]](#)

Για τον θόρυβο λοιπόν έχουμε την παρακάτω αντιστοίχιση των κατωφλίων

- Βαθμολογία 5: 30 - 40 dB(A)
- Βαθμολογία 4: 41 - 50 dB(A)
- Βαθμολογία 3: 51 - 60 dB(A)
- Βαθμολογία 2: 61 - 70 dB(A)
- Βαθμολογία 1: >70 dB(A)

Στη παρακάτω εικόνα 22 βλέπουμε τα αντίστοιχα κατώφλια για τον θόρυβο όπως τα ορίσαμε στο λογισμικό παρουσίασης των δεδομένων Grafana



Εικόνα 22 Κατώφλια θορύβου στο Grafana

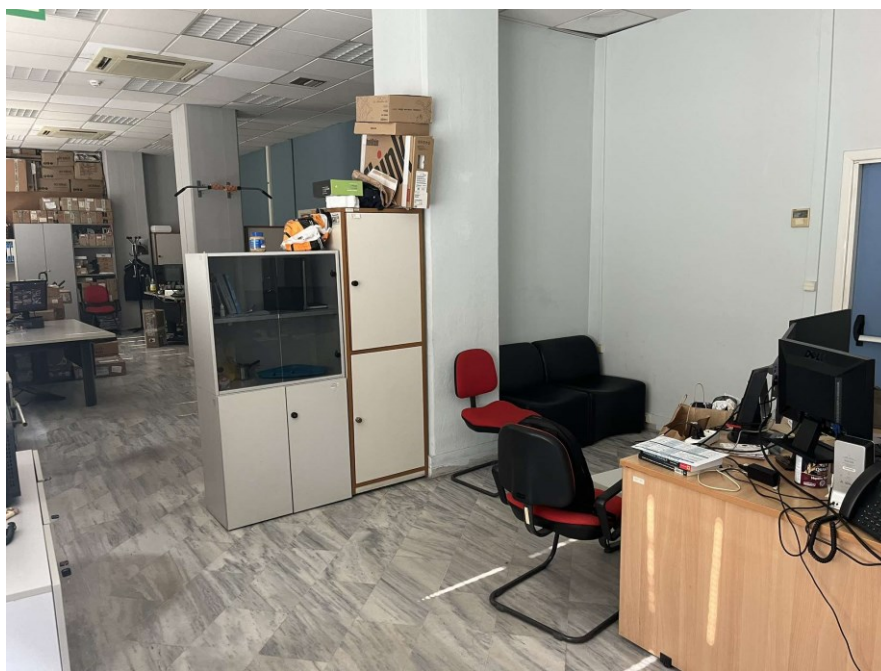
3 ΜΕΛΕΤΗ ΣΥΣΤΗΜΑΤΟΣ ΙΟΤ

Σε αυτό το κεφάλαιο θα δούμε τη μελέτη του συστήματος ΙοΤ για τη παρακολούθηση του περιβάλλοντος. Θα υλοποιηθούν 3 σταθμοί που θα συλλέγουν δεδομένα και θα τοποθετηθούν σε διαφορετικά σημεία. Οι παράμετροι που θα καταγράφονται είναι ίδιοι σε αυτά τα 3 παράλληλα περιβάλλοντα και είναι οι πέντε που αναφέρθηκαν στο προηγούμενο κεφάλαιο : Θερμοκρασία, Υγρασία, Ποιότητα αέρα(Συγκέντρωση CO₂), Θόρυβος και Φωτεινότητα.

3.1 Μελέτη περίπτωσης του συστήματος

Το πρώτο σενάριο θα μελετήσει τρία σημεία ενδιαφέροντος:

- Το σπίτι(ο χώρος του γραφείου είναι δωμάτιο 15TM και διαμένει ένας άνθρωπος)
- Τον εσωτερικό χώρο του γραφείου όπως βλέπουμε στην εικόνα 23(το γραφείο είναι 100TM και εργάζονται 2 άτομα)
- Τον εξωτερικό χώρο του γραφείου(ο εξωτερικός χώρος έξω από το παράθυρο του γραφείου στο κέντρο της Θεσσαλονίκης στην οδό Ερμού, στις εγκαταστάσεις της εταιρείας ΟΤΕ Α.Ε. στον 4^ο όροφο)



Εικόνα 23 Γραφείο εργασίας 2 ατόμων 100TM

Παρακολουθώντας αυτά τα τρία περιβάλλοντα παράλληλα θα μπορούμε ένα δούμε ενδιαφέροντα στοιχεία για αυτούς τους χώρους. Σε κάθε ένα από αυτά τα σημεία θα συλλέγουμε τις πληροφορίες περιβάλλοντος και θα κατευθύνουμε τα δεδομένα έτσι ώστε να αναλύονται και να οπτικοποιούνται. Από την ανάλυση των δεδομένων θα προκύψουν ενδιαφέροντα στοιχεία και θα μπορούμε να απαντήσουμε σε ερωτήματα όπως πχ :

Είναι καλύτερο το περιβάλλον της τηλεργασίας στο σπίτι από το περιβάλλον γραφείου?

Το περιβάλλον του γραφείου μας προστατεύει από κάποια στοιχεία του εξωτερικού περιβάλλοντος?

Στο δεύτερο σενάριο της εργασίας θα αντικατασταθεί ο χώρος του σπιτιού από έναν χώρο γραφείου όπου εργάζονται 5 άτομα(εικόνα 24). Το μέγεθος αυτού του γραφείου είναι 60TM.



Εικόνα 24 Γραφείο εργασίας 5 ατόμων 60TM

3.2 Απαιτήσεις συστήματος

Για τις ανάγκες του συστήματος θα χρειαστούν τρεις μικροελεγκτές (ένας για κάθε σημείο) στον οποίο θα συνδεθούν οι αισθητήρες συλλογής των δεδομένων. Για τη θερμοκρασία & υγρασία θα χρησιμοποιηθεί ο **DHT11**, για τη φωτεινότητα το **photoresistor**, για τη ποιότητα του αέρα ο **MQ135** και για το θόρυβο ο αισθητήρας **MAX4466**. Ακόμα θα χρειαστεί να εγκατασταθεί ένας **MQTT broker** για τα εκτελεί το πρωτόκολλο MQTT και ένα λογισμικό απεικόνισης των δεδομένων. Το **Grafana** θα χρησιμοποιηθεί για την απεικόνιση των δεδομένων. Ο MQTT broker μπορεί να εγκατασταθεί σε κάποιο τερματικό στο χώρο της εγκατάστασης ή να χρησιμοποιηθεί μια cloud λύση. Το ίδιο ισχύει και για το λογισμικό Grafana. Σε αυτή την εργασία θα υλοποιηθούν και οι δυο λύσεις(Τοπική εγκατάσταση ή Cloud).

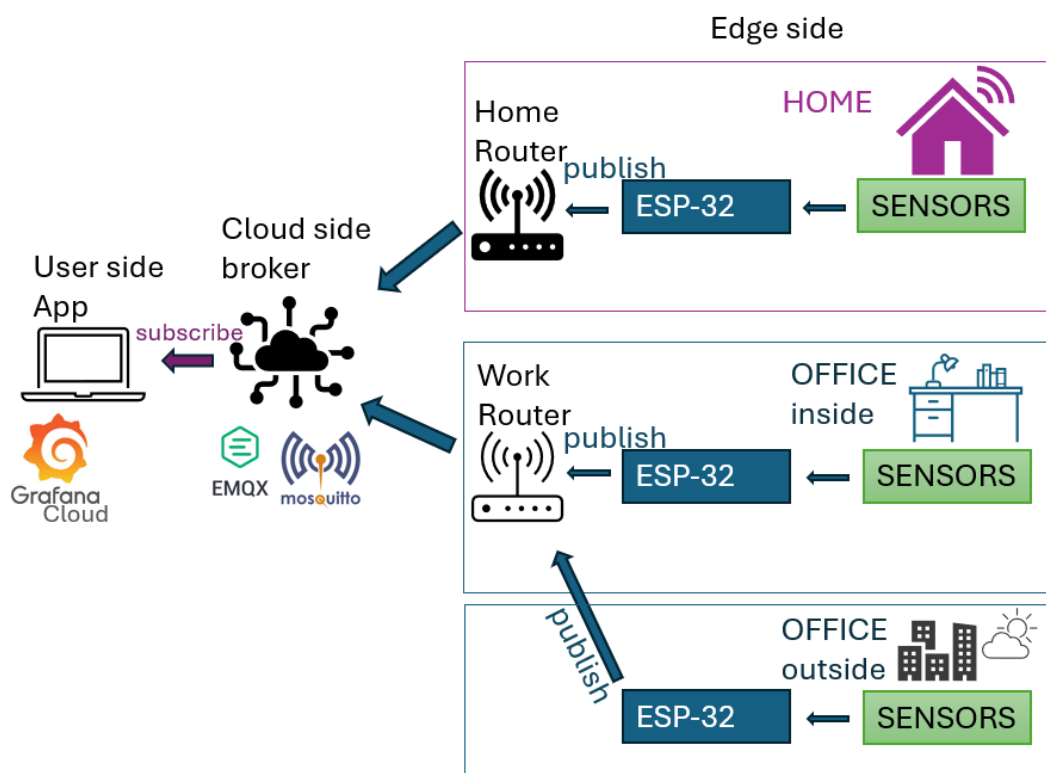
3.3 Σχεδιασμός του συστήματος IoT

Η επιλογή των δυο διαφορετικών προσεγγίσεων(Cloud, Local) έγινε με σκοπό να αναδειχθούν οι διαφορές ανάμεσα στις τοπολογίες και η εξοικείωση με τις διαφορετικού τύπου τεχνολογίες. Η **cloud** λύση προσφέρει μεγαλύτερη ευελιξία και ευκολία στην υλοποίηση αλλά έχει περιορισμούς στις ελεύθερες εκδόσεις. Επίσης υπάρχουν ζητήματα ασφάλειας καθώς το σύστημα λειτουργεί σε χώρο στο cloud και όχι στις εγκαταστάσεις της εταιρείας. Από την άλλη πλευρά, η **local** λύση προσφέρει αυξημένο έλεγχο των δεδομένων και της ασφάλειας, καθώς η επεξεργασία και αποθήκευση γίνεται σε Hardware που εγκαθίσταται στο χώρο τις εταιρείας και μπορεί να υπάγεται στη πολιτική ασφάλειας της εταιρείας μειώνοντας την εξάρτηση από την έξω κόσμο και τις ανησυχίες για ιδιωτικότητα. Η λύση Local απαιτεί συγκεκριμένο Hardware, με ότι συνεπάγεται αυτό(συντήρηση, τροφοδοσία, φύλαξη). Ακόμα χρειάζεται επιπλέον δικτυακό σχεδιασμό και παραμετροποίηση (Routing, IP Addressing, Tunneling, Port Forwarding κτλ) έτσι ώστε το σύστημα να λειτουργεί αρμονικά με ασφάλεια. Η λύση Local

λοιπόν μας δίνει περισσότερη ιδιοτικότητα και για την εγκατάσταση της παρέχονται πακέτα Docker τα οποία μας βοηθούν να εγκαταστήσουμε εύκολα το λογισμικό που θα χρειαστούμε. Ακόμα στη λύση Local μας παρέχονται καλύτερες υπηρεσίες στις ελεύθερες εκδόσεις των λογισμικών που θα χρησιμοποιήσουμε όπως το Grafana και το Fluxdb.

3.3.1 Σχεδιασμός στο Cloud

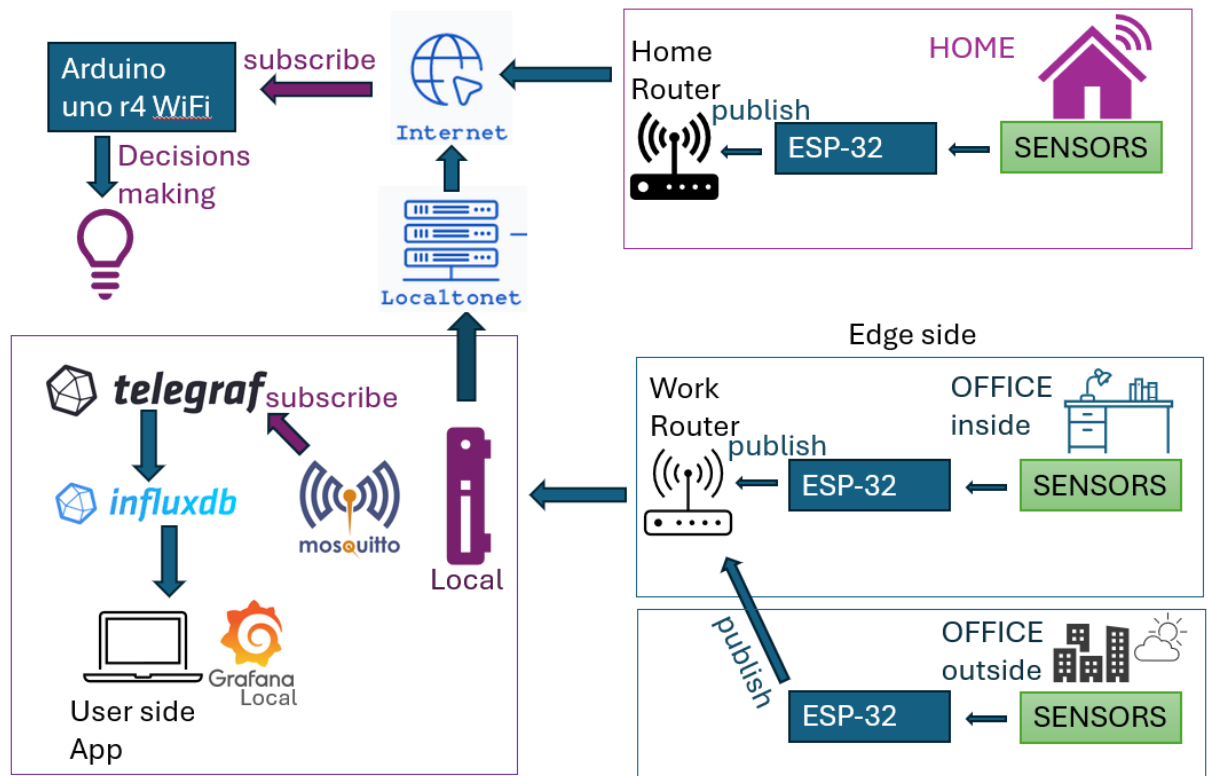
Θα εγκαταστήσουμε λοιπόν έναν μικροελεγκτή ESP32 στο χώρο του σπιτιού και πάνω του θα συνδέσουμε τους αισθητήρες DHT11, MQ135, MAX4466 & ένα photoresistor. Ο ESP32 αρχικά τροφοδοτήθηκε με συνεχές ρεύμα 5V από τη πόρτα USB του Router αλλά στη συνέχεια προέκυψε ότι ήταν ανεπαρκές και αντικαταστάθηκε με τροφοδοτικό που παρείχε περισσότερο ρεύμα(2A). Ακόμα θα επικοινωνεί ασύρματα με τον router μέσω WiFi στα 2.4Ghz και έτσι θα εκτελεί τη λειτουργία publish του πρωτοκόλλου MQTT. Στο χώρο του γραφείου θα εγκατασταθούν δύο ESP32 στους οποίους θα συνδεθούν δυο αισθητήρες DHT, δύο photoresistors, δυο MQ135 και δύο MAX4466. Κάθε esp32 θα παίρνει μετρήσεις για τον εσωτερικό και για τον εξωτερικό χώρο του γραφείου. Οι ESP32 αρχικά τροφοδοτήθηκαν με 5V τον υπολογιστή που είναι εγκατεστημένος ο Mqtt Broker. Το κύκλωμα δούλεψε σωστά μέχρι τους 2-3 αισθητήρες. Όταν εγκαταστάθηκε και ο τέταρτος χρειάστηκε και αλλαγή τροφοδοτικού με περισσότερο ρεύμα. Έτσι λοιπόν εγκαταστάθηκε τροφοδοτικό 5V 2A. Το δίκτυο πρόσβασης του router στο σπίτι είναι τεχνολογίας ADSL 24M και στο γραφείο VDSL 50M. Για την cloud λύση χρησιμοποιήθηκαν ο cloud broker EMQX και εναλλακτικά ο Mosquitto mqtt broker. Για την οπτικοποίηση των δεδομένων χρησιμοποιήθηκε το λογισμικό Grafana στο cloud. Στη παρακάτω εικόνα 25 παρατηρούμε την αρχιτεκτονική που εφαρμόστηκε για τη λύση του Cloud.



Εικόνα 25 Αρχιτεκτονική cloud

3.3.2 Σχεδιασμός τοπικά

Σε αυτή την ενότητα θα αναλύσουμε την εγκατάσταση τοπικά σε Hardware που θα εγκαταστήσουμε εμείς. Η διαφορά με τον παραπάνω (Cloud) σχεδιασμό είναι στην εγκατάσταση του MQTT broker και του λογισμικού απεικόνισης Grafana. Σε αυτή η λύση χρησιμοποιήθηκε ένας H/Y i7-6700 με 8gb ram στον οποίο εγκαταστάθηκε λειτουργικό windows 10. Σε αυτό τον υπολογιστή εγκαταστάθηκε ο **MQTT broker Mosquitto** και το **Grafana**. Ακόμα εγκαταστάθηκαν ο agent **Telegraf** και η βάση δεδομένων **Fluxdb**. Σε αυτή τη λύση θα μπορούμε να αποθηκεύουμε τα δεδομένα στη βάση έτσι ώστε να μπορούμε να εξάγουμε περισσότερα συμπεράσματα όπως για παράδειγμα ο μέσος όρος του τελευταίου 24ωρου. Ακόμα εγκαταστάθηκε το εργαλείο **Ngrok** έτσι ώστε να γίνεται expose ο broker στο internet και να μπορεί το σημείο στο σπίτι να κάνει publish τα δεδομένα του. Εναλλακτική λύση είναι το εργαλείο **Localto** το οποίο μας παρέχει τη δυνατότητα να κάνουμε Expose τον τοπικό μας broker στο internet. Το εργαλείο Localto μας παρείχε περισσότερες υπηρεσίες στην ελεύθερη έκδοση του. Στη παρακάτω εικόνα 26 βλέπουμε την αρχιτεκτονική του συστήματος με εγκατάσταση τοπικά.



Εικόνα 26 Αρχιτεκτονική local

4 ΥΛΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ

4.1 Επιλογή Hardware & Software

Τα υλικά(Hardware) που χρησιμοποιήθηκαν είναι :

- 3X ESP32 WROOM
- Arduino uno r4 WiFi
- 9X resistors 220Ohm
- 3X RGB Leds
- 3X resistors 2k, 47k & 10k
- 6X resistors 1k
- 3X DHT11
- 3X PHOTORESISTOR
- 3X MQ135
- 3X MAX4466
- 3X Raster
- 2X router ADSL/VDSL
- H/Y i7-6700 8Gb ram

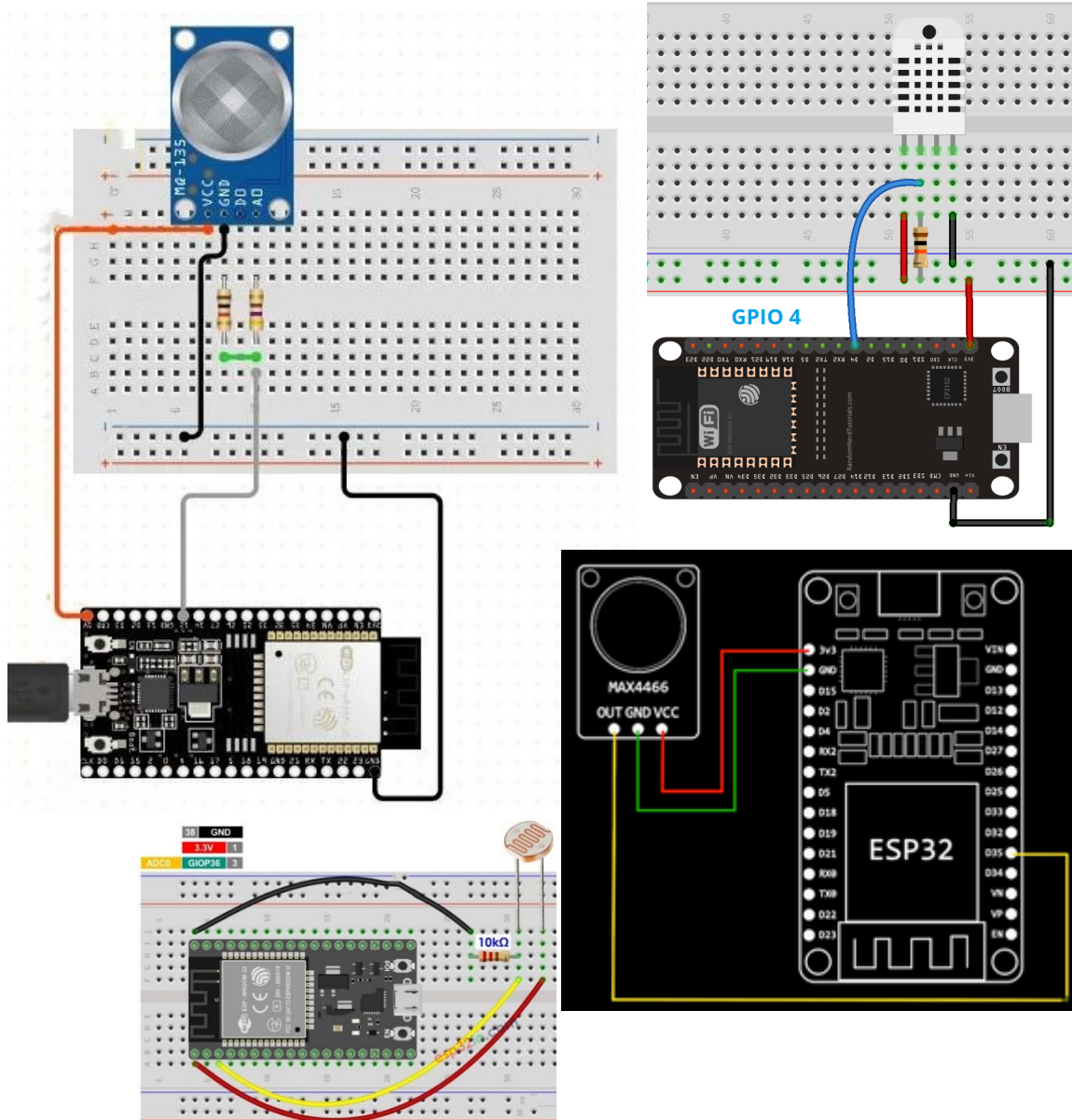
Το λογισμικό που χρησιμοποιήθηκε είναι:

- Cloud & Local brokers Mosquitto, EMQX
- Grafana cloud & Local
- Ngrok service, Localto & local tunnel
- Arduino IDE
- Telegraf agent
- Fluxdb (NoSQL)
- Wireshark
- Nmap

Τα παραπάνω υλικά(Hardware) αναφέρονται στην κατασκευή τριών σταθμών συλλογής δεδομένων, ενός Server και δυο Router που θα εξυπηρετούν τη δικτύωση. Οι αισθητήρες και μικροελεγκτές που χρησιμοποιήθηκαν είναι οι πιο οικονομικοί στην αγορά(τη χρονική στιγμή που έγινε η έρευνα). Ενδεικτικά μπορούμε να αναφέρουμε ότι το κόστος ανά σταθμό είναι περίπου 45€ χωρίς το τροφοδοτικό. Αντί του υπολογιστή i7(ο οποίος υπήρχε στη παρούσα εργασία) θα μπορούσε να χρησιμοποιηθεί ένα raspberry pi καθώς ο mosquitto broker δεν έχει υψηλές απαιτήσεις hardware για να λειτουργεί σωστά.

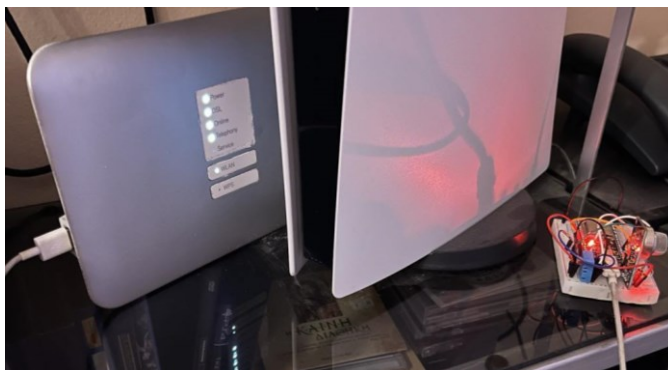
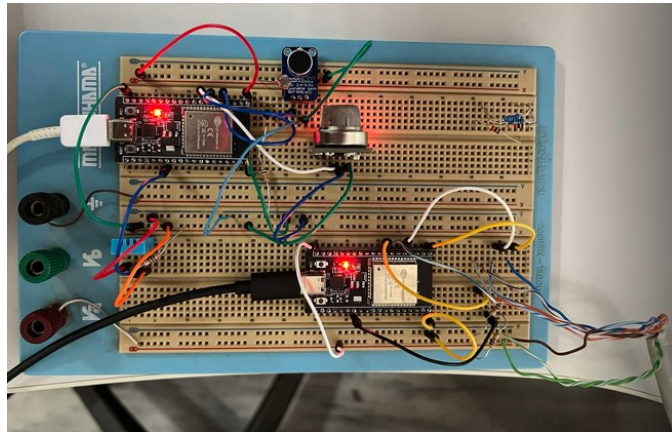
4.2 Συνδεσμολογία

Στη παρακάτω εικόνα 27 βλέπουμε τη συνδεσμολογία για τον MQ135, το photoresistor, τον MAX4466 και για τον DHT11.



Εικόνα 27 Συνδεσμολογία Esp32

Για κάθε αισθητήρα θα συνδέσουμε τις αντίστοιχες αντιστάσεις έτσι ώστε είτε να προστατεύσουμε τον ESP32, είτε να προσαρμόσουμε τη τάση λειτουργίας από τα 5V στα 3,3V. Παρακάτω (εικόνα 28) βλέπουμε τη συνδεσμολογία στα τρία σημεία ενδιαφέροντος.



Εικόνα 28 Συνδεσμολογία στο άκρο

4.3 Προγραμματισμός ESP32

Σε αυτή την ενότητα θα ασχοληθούμε με τον προγραμματισμό των μικροελεγκτών. Για το σημείο στο σπίτι ο κώδικας είναι ο παρακάτω. Αρχικά καλούμε τις βιβλιοθήκες των αισθητήρων και της mqtt λειτουργίας. Ακόμα ορίζουμε τα pins σύνδεσης των αισθητήρων. Έπειτα καθορίζουμε το όνομα ασύρματου δικτύου SSID και τον κωδικό πρόσβασης.

```
#include <WiFi.h>
#include <PubSubClient.h>
#include <MQUnifiedsensor.h>
#include "DHT.h"
#define DHTPIN 4 // Digital pin connected to the DHT sensor
#define DHTTYPE DHT11 // DHT 11
DHT dht(DHTPIN, DHTTYPE);
#define LIGHT_SENSOR_PIN 34
#define MQ135_PIN 33
#define DB_SENSOR 32
// WiFi Home
const char *ssid = "Jungle"; // Enter your WiFi name
const char *password = "XXXXZZZ"; // Enter WiFi password
```

Στη συνέχεια ορίζουμε τα Topics, στα οποία είναι case-sensitive. Έχουμε δυο topics για τη θερμοκρασία και υγρασία, ένα για τη ποιότητα του αέρα(CO2), ένα για τη φωτεινότητα και ένα για το θόρυβο. Ακόμα θα χρησιμοποιήσουμε credentials(username, password) για τον broker. Έπειτα δηλώνουμε τη διεύθυνση του broker και τη πόρτα του. Με τη συνάρτηση millis

θα εκτελέσουμε παράλληλα δυο λειτουργίες έτσι ώστε η μια να μην επηρεάζει την άλλη. Η μια λειτουργία είναι η διατήρηση της επικοινωνίας με τον broker και θα γίνεται ανά 2 δευτερόλεπτα και η άλλη είναι η λειτουργία ανάγνωσης των αισθητήρων και η δημοσίευση(Publish) των τιμών τους στον broker. Ανάγνωση των αισθητήρων θα γίνεται ανά 30 δευτερόλεπτα καθώς δεν υπάρχει ανάγκη να γίνεται πιο συχνά καθώς τα μεγέθη που μετράμε δεν έχουν τόσο άμεσες μεταβολές. Με αυτό το τρόπο εξοικονομούμε όγκο δεδομένων και κατανάλωση ενέργειας.

```
// MQTT Broker
const char *mqtt_broker = "de2.localto.net";
const char *topicTHome = "Omiros/Home/temperature";
const char *topicHHome = "Omiros/Home/humidity";
const char *topicAHome = "Omiros/Home/air";
const char *topicAudioHome = "Omiros/Home/audio";
const char *topicLightHome = "Omiros/Home/light";

const char *mqtt_username = "perseas";
const char *mqtt_password = "ZZZXXXX";
const int mqtt_port = 10855;
WiFiClient espClient;
PubSubClient client(espClient);

//milis time events
const long eventTime_clientLoop = 2000;
const long eventTime_readSensors = 30000;
unsigned long previousTime_clientLoop,previousTime_readSensors = 0;
const int sampleWindow = 50; // Sample window width in mS (50 mS = 20Hz)
```

Όπως παρατηρούμε στον παραπάνω κώδικα χρησιμοποιούμε τη διεύθυνση του host: de2.localto.net και τη πόρτα 10855. Αυτά τα στοιχεία έχουν προκύψει από τη χρήση της υπηρεσίας localto με την οποία κάνουμε expose τον τοπικό μας broker στη internet. Ακόμα ορίζουμε και τη συχνότητα δειγματοληψίας για τη μέτρηση του θορύβου σε 50mSec -> 20Hz. Παρακάτω έχουμε τη συνάρτηση setup όπου ορίζουμε το data rate της σειριακής επικοινωνίας στα 115200bps. Έπειτα εκκινούμε τον αισθητήρα DHT11 όπως φαίνεται παρακάτω.

```
void setup() {
// Set software serial baud to 115200;
Serial.begin(115200);
dht.begin();
connectToWiFi();
```

Στη συνέχεια έχουμε τη λειτουργία σύνδεσης με το WiFi δίκτυο και τον broker. Δημιουργούμε τη μέθοδο connect ToWiFi(); και την εκτελούμε. Εδώ έχουμε κάποιες ρουτίνες ελέγχου με τα αντίστοιχα μηνύματα σφάλματος σε περίπτωση αποτυχίας της σύνδεσης. Ακόμα εμφανίζουμε τη local ip που παίρνει ο esp32 και την ένταση του σήματος Wifi σε db.

```
void connectToWiFi(){
// connecting to a WiFi network
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
delay(500);
Serial.println("Connecting to WiFi..");
```

```

}
Serial.print("Connected, IP address: ");
Serial.print(WiFi.localIP());
Serial.print(", RSSI: ");
Serial.println(WiFi.RSSI());
}

```

Αμέσως μετά ακολουθεί η σύνδεση στον mqtt broker. Εδώ δημιουργούμε ένα client id χρησιμοποιώντας τη mac address του esp32 και με το username & password κάνουμε τη σύνδεση. Σε περίπτωση λάθους τυπώνουμε το μήνυμα λάθους που μας επιστρέφεται.

```

//connecting to a mqtt broker
client.setServer(mqtt broker, mqtt port);
client.setCallback(callback);
while (!client.connected()) {
  String client id = "esp32-client-";
  client_id += String(WiFi.macAddress());
  Serial.printf("The client %s connects to the public mqtt broker\n", client_id.c_str());
  if (client.connect(client_id.c_str(), mqtt_username, mqtt_password)) {
    Serial.println("Perseas MQTT broker connected");
  } else {
    Serial.print("failed with state ");
    Serial.print(client.state());
    delay(2000);
  }
}
}

```

Παρακάτω έχει οριστεί η συνάρτηση callback σε περίπτωση που θέλουμε να κάνει subscribe ο esp32 σε κάποιο topic.

```

void callback(char *topicTHome, byte *payload, unsigned int length) {
  Serial.print("Message arrived in topicTHome: ");
  Serial.println(topicTHome);
  Serial.print("Message:");
  for (int i = 0; i < length; i++) {
    Serial.print((char) payload[i]);
  }
  Serial.println();
  Serial.println("-----");
}

```

Μπαίνουμε λοιπόν στη συνάρτηση loop. Αρχικά ορίζουμε τη παράλληλη λειτουργία των δυο κύριων λειτουργιών μας με τη συνάρτηση millis. Η client.loop(); Διατηρεί τη σύνδεση με το broker.

```

void loop() {
  unsigned long currentTime = millis();

  //Client Loop delay 2secs
  if (currentTime - previousTime_clientLoop >= eventTime_clientLoop)
  {
    client.loop();
  }
}

```

```

previousTime_clientLoop = currentTime;
}

//Read Sensors delay 20secs
if (currentTime - previousTime_readSensors >= eventTime_readSensors)
{
    ReadSensorsAndPublish();
    previousTime_readSensors = currentTime;
}
}

```

Η συνάρτηση ReadSensorsAndPublish(); Αφού ελέγξει ότι η σύνδεση WiFi και η σύνδεση με τον broker είναι ενεργή, διαβάζει τους τέσσερις αισθητήρες και καταχωρεί τις τιμές τους σε μεταβλητές. Ο MQ135 μας επιτρέπει με κάποιους υπολογισμούς να μετρήσουμε τις τιμές για τα CO2, CO, Alcohol, Acetone, NH4 και Toluene. Σε αυτή την εργασία έχουμε επιλέξει τη μέτρηση της παραμέτρου CO2.

```

void ReadSensorsAndPublish(){
    // Ελέγξε αν η σύνδεση WiFi χάθηκε και προσπάθησε να συνδεθείς ξανά
    if (WiFi.status() != WL_CONNECTED) {
        Serial.println("WiFi connection lost, attempting to reconnect...");
        connectToWiFi(); // Προσπάθησε να συνδεθείς στο WiFi ξανά
    }
    // Ελέγξε αν η σύνδεση με τον broker MQTT χάθηκε και προσπάθησε να επανασυνδεθείς
    if (!client.connected()) {
        reconnectMqtt(); // Προσπάθησε να επανασυνδεθείς στον broker MQTT
    }
    float h = dht.readHumidity();
    // Read temperature as Celsius (the default)
    float t = dht.readTemperature();
    // Check if any reads failed and exit early (to try again).
    if (isnan(h) || isnan(t)) {
        Serial.println(F("Failed to read from DHT sensor!"));
        return;
    }
    int CO2 = analogRead(MQ135_PIN); // Sensor will read CO2 concentration
    int lightPercentage = (analogRead(LIGHT_SENSOR_PIN)*100)/4095;
}

```

Για τη μέτρηση της φωτεινότητας κάνουμε calibration όπως φαίνεται στην εικόνα 29 βάζοντας ένα φακό στον αισθητήρα και κρύβοντας τον στο σκοτάδι. Έτσι βλέπουμε τη χαμηλότερη και τη μεγαλύτερη τιμή που μπορεί να πάρει ο photoresistor. Το εύρος τιμών λοιπόν από το απόλυτο σκοτάδι μέχρι το φως είναι 0-4095. Έτσι θα υπολογίσουμε τη ποσοστιαία % φωτεινότητα.

```

Analog Value = 4095 => Very bright Analog Value = 0 => Dark
Analog Value = 4095 => Very bright Analog Value = 0 => Dark
Analog Value = 4095 => Very bright Analog Value = 0 => Dark
Analog Value = 4095 => Very bright Analog Value = 0 => Dark
Analog Value = 4095 => Very bright Analog Value = 0 => Dark

```

Εικόνα 29 Calibration photoresistor

Για τη μέτρηση του θορύβου θα χρησιμοποιήσουμε τον παρακάτω κώδικα. Θα συλλέξουμε λοιπόν τις τιμές του αισθητήρα MAX4466 και θα υπολογίζουμε την τιμή Peak to Peak. Έπειτα θα μετατρέψουμε σε db είτε με τη συνάρτηση map είτε υπολογίζοντας το λογάριθμο.

```

///db audio
unsigned long startMillis = millis(); // Start of sample window
unsigned int peakToPeak = 0; // peak-to-peak level
unsigned int signalMax = 0;
unsigned int signalMin = 4095;
unsigned int sample;
int numSamples =50;
for (int i = 0; i < numSamples; i++) {
  sample = analogRead(DB_SENSOR);
  if (sample < 4095) {
    if (sample > signalMax) {
      signalMax = sample;
    }
    if (sample < signalMin) {
      signalMin = sample;
    }
  }
  delayMicroseconds(100); // Καθυστέρηση για σταθερό ρυθμό δειγματοληψίας
}
Serial.print(" sample: ");
Serial.println(sample);
peakToPeak = signalMax - signalMin; // max - min = peak-peak amplitude
Serial.print(" peakToPeak: ");
Serial.println(peakToPeak);
float audio = map(peakToPeak,70,4095,30,95); // Convert to dB

```

Αφού λοιπόν τυπώσουμε τις τιμές τους στο serial monitor για να ξέρουμε ότι δουλεύει σωστά θα κάνουμε publish στα αντίστοιχα topics. Τέλος τυπώνουμε τη κατάσταση σύνδεσης με το broker για τον έλεγχο της συνδεσιμότητας μαζί του.

```

Serial.print(F("Humidity: "));
Serial.print(h);
Serial.print(F("% Temperature: "));
Serial.print(t);
Serial.print(F("°C "));
Serial.print("CO2: ");
Serial.print(CO2);
Serial.print(" Audio : ");
Serial.print(audio);

```

```

Serial.print(" Light : ");
Serial.print(lightPercentage);
Serial.println("%");
// Publish to MQTT topic

String temperature = String(t);
String humidity = String(h);
String air = String(CO2);
String Audio = String(audio);
String lightP = String(lightPercentage);
client.publish(topicTHome, temperature.c_str());
client.publish(topicHHome, humidity.c_str());
client.publish(topicAHome, air.c_str());
client.publish(topicAudioHome, Audio.c_str());
client.publish(topicLightHome, lightP.c_str());

//print the state of the connection
if (client.state()==0){
  Serial.println("Connected to Broker");}
else{
  Serial.print("No connection with broker. State : ");
  Serial.println(client.state());}
}

```

Τέλος δημιουργούμε τη μέθοδο `reconnectMqtt()`; η οποία είναι απαραίτητη έτσι ώστε να επανέρχεται η σύνδεση με τον broker όταν αυτή χάνεται. Αυτή η μέθοδος βοηθάει στη σταθερότητα της συνδεσιμότητας του συστήματος. Παρατηρήθηκε ότι χωρίς αυτή χρειάζονταν κάποια reset στους Esp32 περιοδικά. Αυτή η μέθοδος εκτελείτε κάθε φορά που θέλουμε να κάνουμε publish τις τιμές των αισθητήρων.

```

void reconnectMqtt() {
  while (!client.connected()) {
    Serial.println("Attempting MQTT connection...");
    if (client.connect("ESP32Client", mqtt_username, mqtt_password)) {
      Serial.println("Connected to MQTT broker");
    } else {
      Serial.print("Failed to connect to MQTT broker, rc=");
      Serial.print(client.state());
      Serial.println(" Retrying in 5 seconds...");
      delay(5000);
    }
  }
}

```

Για τα υπόλοιπα δυο σημεία στο χώρο του γραφείου ο προγραμματισμός είναι παρόμοιος. Οι βασικές αλλαγές που χρειάζονται έχουν να κάνουν με τη συνδεσιμότητα σε διαφορετικό δίκτυο. Εδώ συνδεόμαστε στη τοπική ip του mosquitto mqtt broker η οποία είναι η 192.168.1.7. ακόμα πρέπει να αλλάξουμε τα ονόματα των topics και της πόρτας. Για λόγους ασφαλείας θα χρησιμοποιήσουμε την τυχαία πόρτα 5555 πάρα την 1883 που είναι η συνήθης πόρτα για mqtt brokers. Ο mqtt broker θα ρυθμιστεί έτσι ώστε να απαιτείται η χρήση credentials για τη σύνδεση σε αυτόν. Στον παρακάτω κώδικα παρατηρούμε το διαφορετικό ασύρματο δίκτυο και τα διαφορετικά topics τα οποία αφορούν το χώρο του γραφείου.

```

// WiFi work
const char *ssid = "DATA"; // Enter your WiFi name
const char *password = "XXXZZZZZ"; // Enter WiFi password
// MQTT Broker
const char *mqtt_broker = "192.168.1.7";
//inside sensors topics
const char *topicTHome = "Omiros/Work/temperatureIn";
const char *topicHHome = "Omiros/Work/humidityIn";
const char *topicAHome = "Omiros/Work/airIn";
const char *topicTolueneWorkIn = "Omiros/Work/airTolueneIn";
const char *topicNH4WorkIn = "Omiros/Work/NH4In";
const char *topicLightWorkIn = "Omiros/Work/lightIn";
const char *mqtt_username = "perseas";
const char *mqtt_password = "veronica";
const int mqtt_port = 55555;

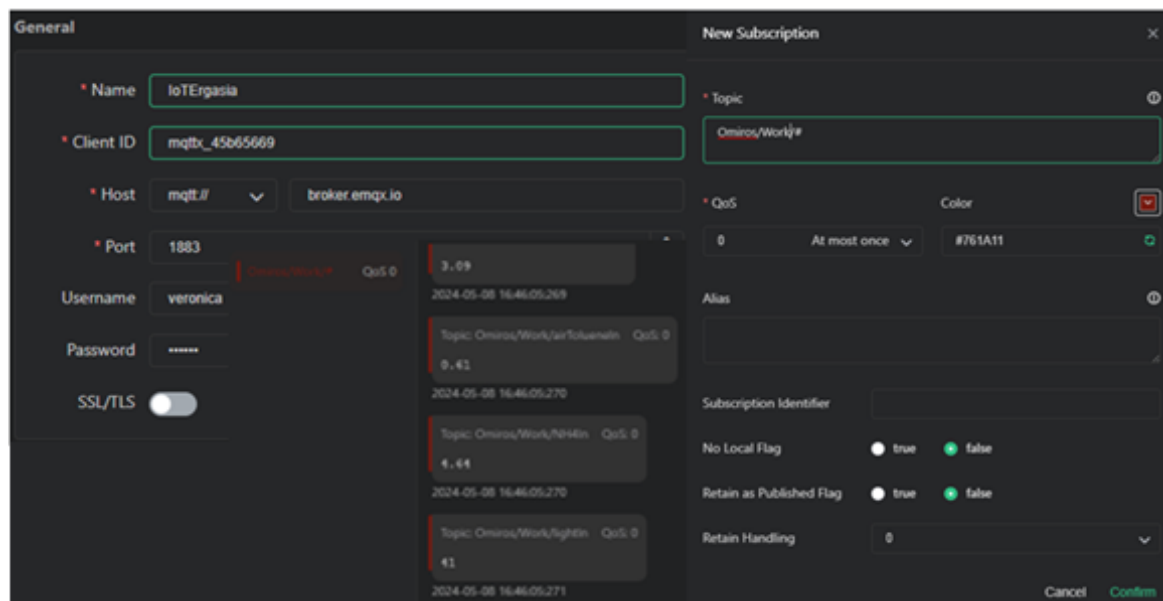
```

4.4 Εγκατάσταση Mqtt Broker

Σε αυτή την ενότητα θα ασχοληθούμε με την εγκατάσταση του MQTT Broker. Υλοποιήθηκαν οι δυο λύσεις : cloud και local. Για τη λύση του cloud δοκιμάστηκαν οι πιο γνωστοί online brokers που υπάρχουν ενώ για τη λύση local ο mosquitto broker.

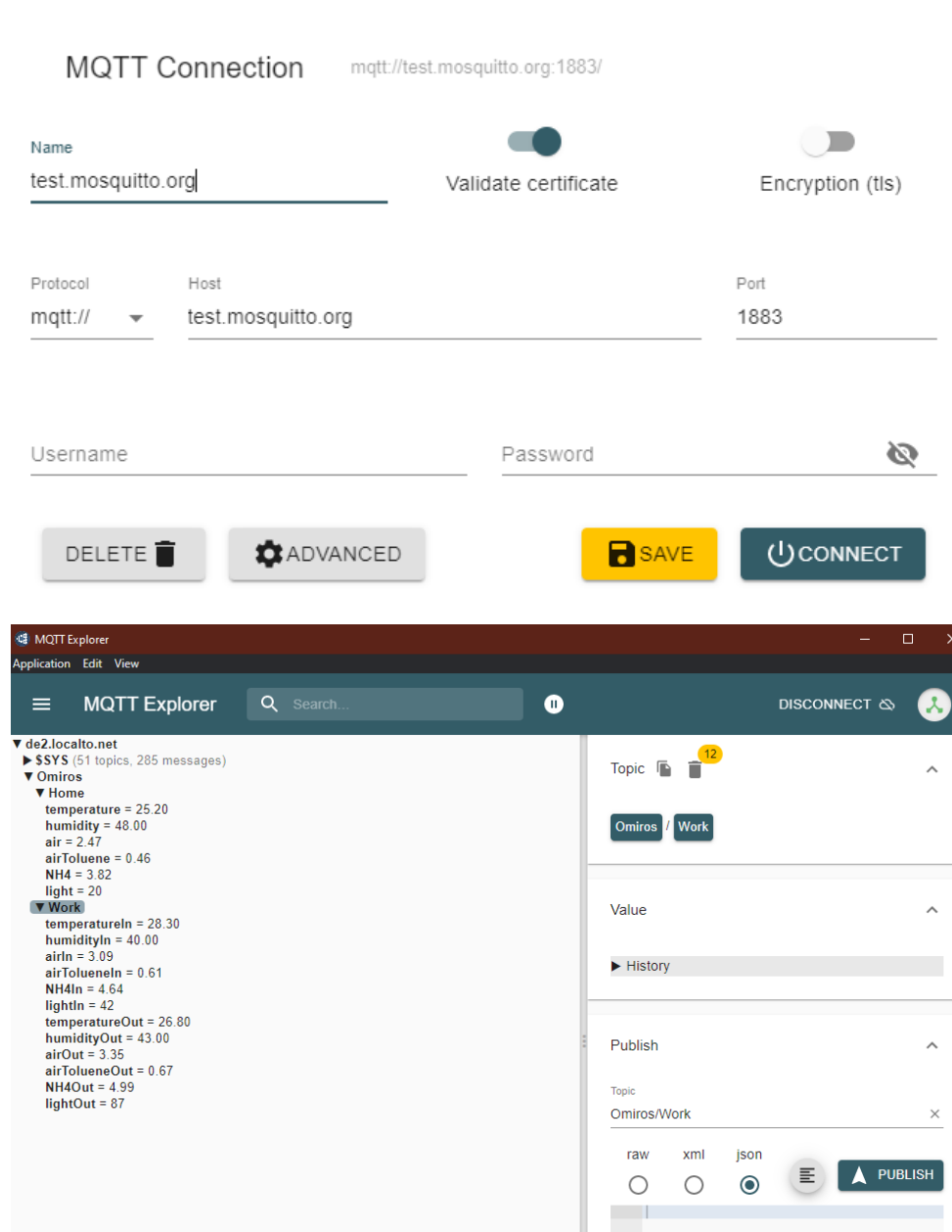
4.4.1 Εγκατάσταση Cloud broker

Για το σενάριο όπου χρησιμοποιούμε κάποιον έτοιμο cloud broker δεν απαιτείται εγκατάσταση τοπικά. Για τον EMQX broker μπορούμε να χρησιμοποιήσουμε το εργαλείο mqttx με το οποίο μπορούμε να ελέγξουμε τη συνδεσιμότητα και να κάνουμε publish & subscribe σε δικά μας topics. Παρακάτω(εικόνα 30) βλέπουμε τα στοιχεία σύνδεσης του cloud broker EMQX καθώς και τη διαδικασία subscribe σε όλα τα topics που αφορούν τα σημεία στο χώρο εργασίας.



Εικόνα 30 Χρήση EMQX Broker

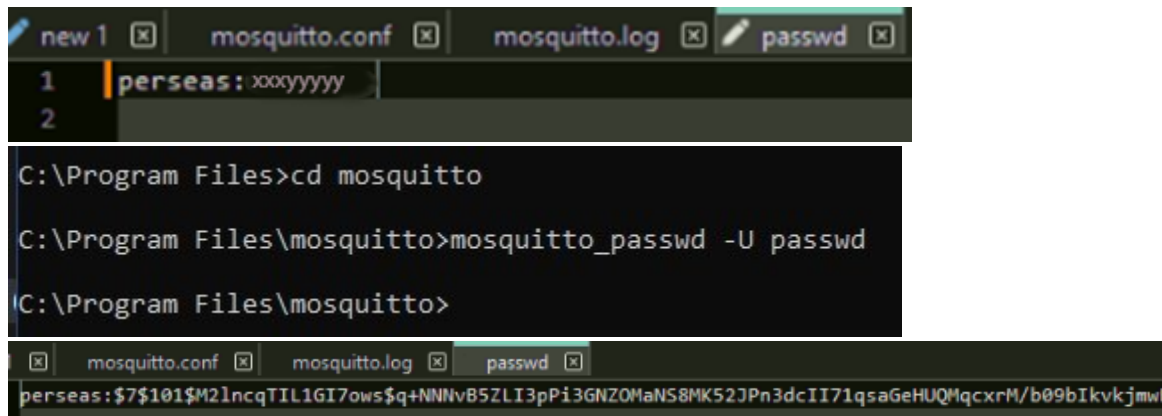
Εναλλακτικά μπορούμε να χρησιμοποιήσουμε τον mosquitto broker cloud με το client λογισμικό mqtt explorer. Παρακάτω(εικόνα 31) βλέπουμε τα στοιχεία σύνδεσης του cloud broker Mosquitto καθώς και τη διαδικασία subscribe σε όλα τα topics που έχουμε δημιουργήσει. Όπως και στο προηγούμενο εργαλείο μπορούμε να κάνουμε publish σε κάποιο topic.



Εικόνα 31 Mosquitto Broker - MQTT Explorer

4.4.2 Εγκατάσταση Local Mosquitto broker

Η εγκατάσταση του broker mosquitto τοπικά μπορεί να γίνει είτε με χρήση docker είτε μέσω του installer που βρίσκουμε στη σελίδα του mosquitto. Μετά την εγκατάσταση πρέπει να ρυθμίσουμε το αρχείο mosquitto.conf που βρίσκεται στο φάκελο της εγκατάστασης. Ανοίγοντας το αρχείο βλέπουμε τις οδηγίες με μορφή σχολίων. Θα δημιουργήσουμε ένα αρχείο password το οποίο θα έχει τα credentials του broker. Για το password θα κρυπτογραφήσουμε χρησιμοποιώντας τον cli του mosquitto. Ορίσουμε τον χρήστη με τον κωδικό του και έπειτα τρέχουμε το αρχείο κρυπτογράφησης με την παρακάτω εντολή και βλέπουμε ότι έχει κρυπτογραφήσει τον κωδικό.(εικόνα 32)



```
new 1 x mosquitto.conf x mosquitto.log x passwd x
1 |perseas: xxxxyyyy
2 |
C:\Program Files>cd mosquitto
C:\Program Files\mosquitto>mosquitto_passwd -U passwd
C:\Program Files\mosquitto>
mosquitto.conf x mosquitto.log x passwd x
perseas:$7$101$M2lncqTIL1GI7ows$q+NNNvB5ZLI3pPi3GNZOMaNS8MK52JPn3dcII71qsaGeHUQMqcxrM/b09bIkvkjmw
```

Εικόνα 32 Mosquitto κρυπτογράφηση Password

Παρακάτω βλέπουμε το configuration του broker. Αρχικά Ορίζουμε τη διαδρομή του αρχείου χρηστών και κωδικών και αποτρέπουμε τη σύνδεση ανώνυμων χρηστών. Έπειτα ορίζουμε τη πόρτα 55555 ανοιχτή για τον broker μας για συνδέσεις ipv4 & ipv6. Στη συνέχεια ορίζουμε τη διαδρομή για το αρχείο log και τα στοιχεία που θέλουμε να καταγράψει έτσι ώστε να μπορούμε να ελέγχουμε θέματα βλαβών και ασφαλείας.

```
password_file C:\Program Files\mosquitto\passwd
```

```
listener 55555
```

```
allow_anonymous false
```

```
connection_messages true
```

```
log_type error
```

```
log_dest file C:\Program Files\mosquitto\log\mosquitto.log
```

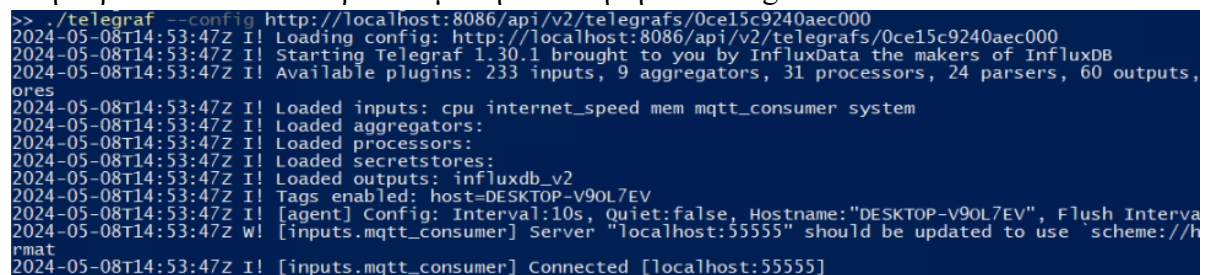
4.5 Εγκατάσταση Telegraf agent

Το επόμενο βήμα αφορά την εγκατάσταση του Telegraf τοπικά. Ο Telegraf είναι ένας agent που μας βοηθάει να συνδέσουμε τη ροή δεδομένων ενός mqtt broker προς μια βάση δεδομένων. Η εγκατάσταση του telegraf γίνεται μέσα από ένα powershell που τρέχουμε στον υπολογιστή που βρίσκεται ο mqtt broker. Φτιάχνουμε ένα token λοιπόν και εκκινούμε τον agent Telegraf με τις παρακάτω εντολές

```
$env:INFLUX_TOKEN = "4-FegfmyEFW6YIBJd5FyBY_Si_Sqw0enFIX_yLto9O4jE7xwiFGFp934qCRY6Cj9IKpe9niBPOYgq7TYMpYPeQ=="
```

```
./telegraf --config http://localhost:8086/api/v2/telegrafs/0ce15c9240aec000
```

Στη παρακάτω εικόνα 33 βλέπουμε την εκκίνηση του Telegraf.



```
>> ./telegraf --config http://localhost:8086/api/v2/telegrafs/0ce15c9240aec000
2024-05-08T14:53:47Z I! Loading config: http://localhost:8086/api/v2/telegrafs/0ce15c9240aec000
2024-05-08T14:53:47Z I! Starting Telegraf 1.30.1 brought to you by InfluxData the makers of InfluxDB
2024-05-08T14:53:47Z I! Available plugins: 233 inputs, 9 aggregators, 31 processors, 24 parsers, 60 outputs,
ores
2024-05-08T14:53:47Z I! Loaded inputs: cpu internet_speed mem mqtt_consumer system
2024-05-08T14:53:47Z I! Loaded aggregators:
2024-05-08T14:53:47Z I! Loaded processors:
2024-05-08T14:53:47Z I! Loaded secretstores:
2024-05-08T14:53:47Z I! Loaded outputs: influxdb_v2
2024-05-08T14:53:47Z I! Tags enabled: host=DESKTOP-V90L7EV
2024-05-08T14:53:47Z I! [agent] Config: Interval:10s, Quiet:false, Hostname:"DESKTOP-V90L7EV", Flush Interval:10s
2024-05-08T14:53:47Z W! [inputs.mqtt_consumer] Server "localhost:55555" should be updated to use `scheme://h
rmat
2024-05-08T14:53:47Z I! [inputs.mqtt_consumer] Connected [localhost:55555]
```

Εικόνα 33 Εκκίνηση Telegraf

ακόμα πρέπει να συντάξουμε το configuration file του Telegraf έτσι ώστε να ορίσουμε τη πηγή, τον προορισμό και τη μετατροπή του τύπου των δεδομένων μας που προκύπτουν από τα topics που έχουμε δημιουργήσει. Ειδικότερα θα ορίσουμε την έξοδο του Telegraf η οποία θα είναι το bucket της fluxdb στη πόρτα 8086 με το αντίστοιχο token. Στο πεδίο inputs ορίζουμε τον mqtt broker και την αντίστοιχη πόρτα. Εδώ μπορούμε να βάλουμε την external ip του router που ανήκει ο broker μας ή να ορίσουμε τη τοπική ip του υπολογιστή στην οποία θα τρέχει ο broker. Στη πρώτη περίπτωση απαιτείται port forward. Επίσης μπορούμε να ορίσουμε το localhost στη πόρτα 55555 που τρέχει ο broker. Έπειτα ορίζουμε όλα τα topics για τα οποία θέλουμε να έχουμε καταγραφή. Θα χρειαστούμε και κάποια στοιχεία για την υγεία του συστήματος που βρίσκεται ο broker οπότε προσθέτουμε τη κατανάλωση cpu, memory καθώς και τη ποιότητα σύνδεσης του internet.(Εικόνα 34)

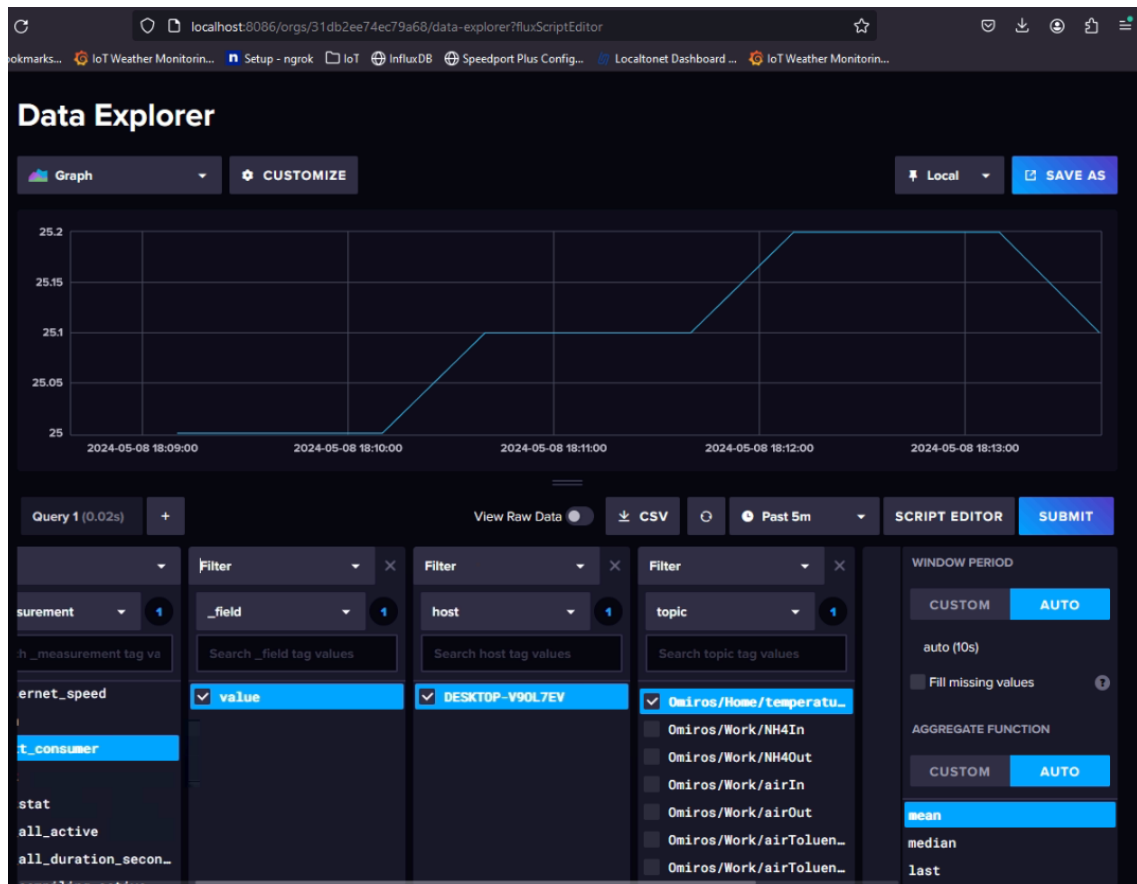
Telegraf Weather IoT

```
1 # Configuration for telegraf agent
2 [[agent]]
3   interval = "10s"
4   round_interval = true
5   metric_batch_size = 1000
6   metric_buffer_limit = 10000
7   collection_jitter = "0s"
8   flush_interval = "10s"
9   flush_jitter = "0s"
10  precision = ""
11  hostname = ""
12  omit_hostname = false
13
14  [[outputs.influxdb_v2]]
15    urls = ["http://localhost:8086"]
16    token = "$INFLUX_TOKEN"
17    organization = "PERSEAS IoT"
46    username = "perseas"
47    password = "xxxxxxxx"
48  data_format = "value"
49  data_type = "float"
50
51  [[inputs.mqtt_consumer.fieldpass]]
52    field = "value"
53    converter = "float"
54
55  [[inputs.mem]]
56  # no configuration needed
20 [[inputs.system]]
21 [[inputs.cpu]]#cpu monitoring
22 [[inputs.internet_speed]]
23 #[[inputs.net]]
24 [[inputs.mqtt_consumer]]
25   servers = ["localhost:55555"]
26   topics = [
27     "Omiros/Home/temperature",
28     "Omiros/Home/humidity",
29     "Omiros/Home/air",
30     "Omiros/Home/audio",
31     "Omiros/Home/light",
32     "Omiros/Work/temperatureIn",
33     "Omiros/Work/humidityIn",
34     "Omiros/Work/airIn",
35     "Omiros/Work/audioIn",
36     "Omiros/Work/lightIn",
37     "Omiros/Work/temperatureOut",
38     "Omiros/Work/humidityOut",
39     "Omiros/Work/airOut",
40     "Omiros/Work/audioOut",
41     "Omiros/Work/lightOut"
42   ]
```

Εικόνα 34 Telegraf Configuration

4.6 Εγκατάσταση Influxdb

Το επόμενο βήμα είναι η εγκατάσταση του influxdb. Αφού κατεβάσουμε τα αρχεία που χρειάζονται εγκαθιστούμε τη βάση δεδομένων influxdb με εντολές μέσω του powershell και εκκινούμε την υπηρεσία στον τοπικό υπολογιστή που βρίσκεται και ο broker με τον agent. Εκκινούμε τη βάση δεδομένων : /influxd και στη συνέχεια μέσω του browser προηγούμενα στο περιβάλλον της influxdb στη πόρτα 8086. Εδώ ορίζουμε τη σύνδεση με το telegraf μέσω ενός bucket. Σε αυτό το περιβάλλον μπορούμε να τρέχουμε ερωτήματα στη βάση και να δούμε την οπτικοποίηση τους επιλέγοντας μέσα από ένα εύχρηστο low code περιβάλλον.(Εικόνα 35)



Εικόνα 35 Data Explorer

Επιλέγοντας script editor μπορούμε να εξαγάγουμε τον κώδικα sql ή flux(εικόνα 36) έτσι ώστε να τον χρησιμοποιήσουμε στο Grafana στο επόμενο στάδιο δημιουργίας dashboard.

```

Query 1 (0.02s) + View Raw Data
1 from(bucket: "telegraf")
2   |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
3   |> filter(fn: (r) => r["_measurement"] == "mqtt_consumer")
4   |> filter(fn: (r) => r["_field"] == "value")
5   |> filter(fn: (r) => r["host"] == "DESKTOP-V90L7EV")
6   |> filter(fn: (r) => r["topic"] == "Omiros/Home/temperature")
7   |> aggregateWindow(every: v.windowPeriod, fn: mean, createEmpty: false)
8   |> yield(name: "mean")

```

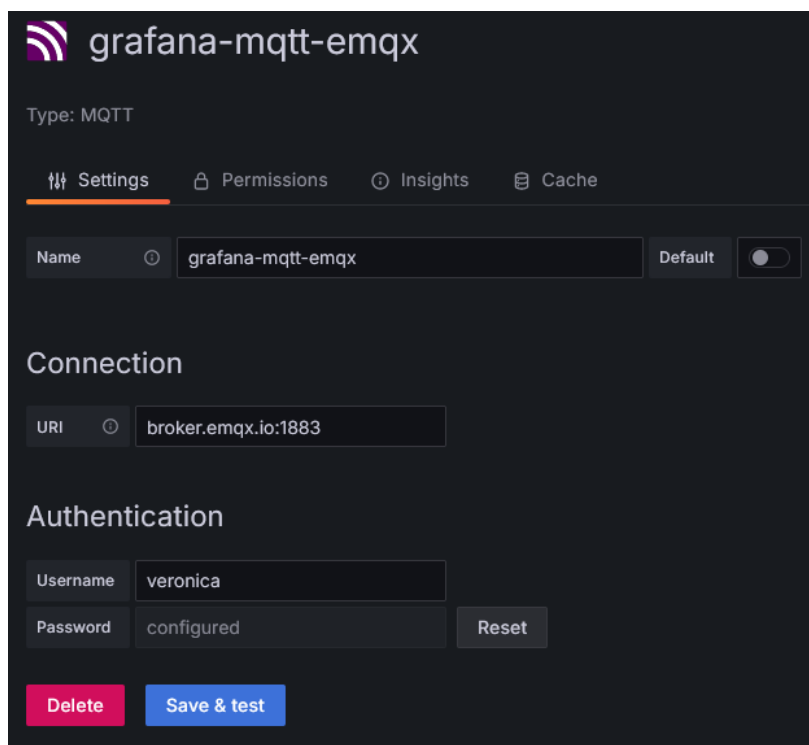
Εικόνα 36 Flux query

4.7 Εγκατάσταση Grafana

Το Grafana είναι ένα λογισμικό οπτικοποίησης δεδομένων με πολλές δυνατότητες. Για την εγκατάσταση του Grafana μπορούμε να χρησιμοποιήσουμε docker ή stand alone install. Υπάρχει η έκδοση cloud και η τοπική εγκατάσταση που ακούει στη πόρτα 3000.

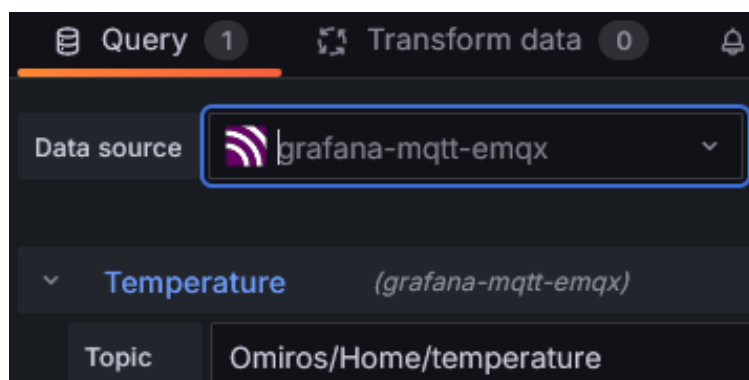
4.7.1 Grafana cloud

Για την υλοποίηση της cloud λύσης χρησιμοποιήθηκε η cloud έκδοση στην οποία μπορούμε να ορίσουμε τον cloud mqtt broker σαν data source. Έτσι τα δεδομένα οπτικοποιούνται σε πραγματικό χρόνο. Στη παρακάτω εικόνα 37 βλέπουμε τον ορισμό του data source στον cloud mqtt broker EMQX στη πόρτα 1883.



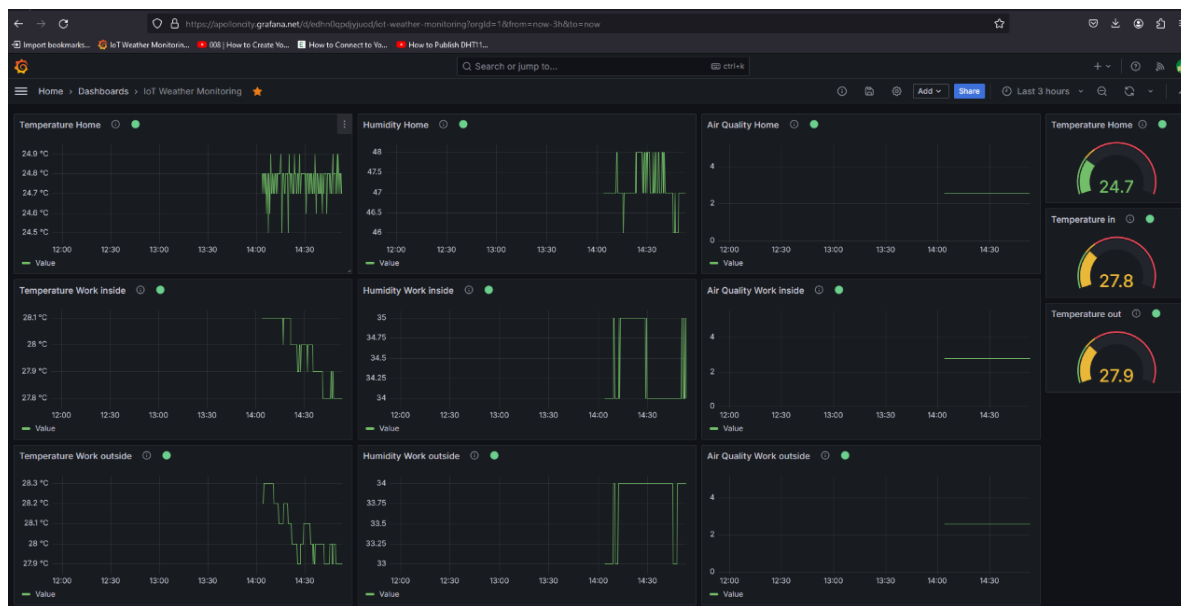
Εικόνα 37 Mqtt Data source

Το επόμενο βήμα είναι να δημιουργήσουμε το dashboard στο οποίο θα απεικονίζονται τα δεδομένα ροής. Δημιουργούμε τα γραφήματα ανά topic. Στη παρακάτω εικόνα 38 βλέπουμε τη δήλωση του topic Omiros/Home/temperature που αφορά τη θερμοκρασία στο περιβάλλον του σπιτιού.



Εικόνα 38 Panel query

Με αυτό το τρόπο χτίζουμε το dashboard αναπτύσσοντας τα topics. Στη παρακάτω εικόνα 39 βλέπουμε τα αντίστοιχα γραφήματα για τα topics θερμοκρασίας, υγρασίας και ποιότητας αέρα. Στο δεξιό μέρος χρησιμοποιούμε το element Gauge έτσι ώστε να εμφανίσουμε τη τελευταία τιμή που παίρνουμε από τους αισθητήρες.



Εικόνα 39 Dashboard cloud Mqtt

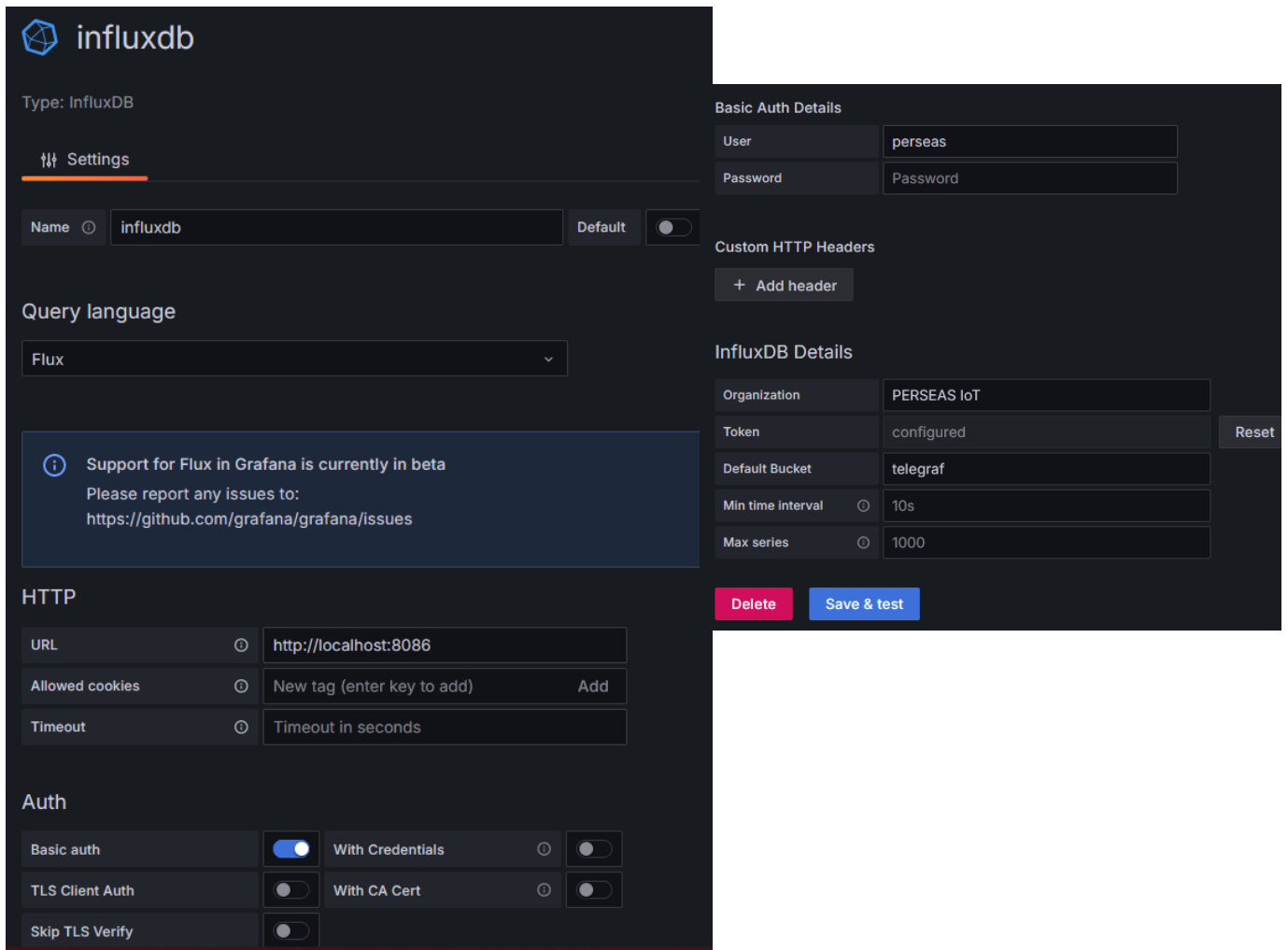
Η υλοποίηση cloud έχει το πλεονέκτημα ότι δεν απαιτεί εξοπλισμό hardware και μας δίνει την ευκολία να έχουμε πρόσβαση από οποιοδήποτε δίκτυο στο internet κάνοντας είσοδο στο λογαριασμό μας στο Grafana όπως φαίνεται στο παρακάτω link.

<https://apolloncity.grafana.net/d/edhn0qpdjyud/iot-weather-monitoring?orgId=1&from=now-24h&to=now>

Βέβαια η έκδοση του cloud έχει κάποιους περιορισμούς στην ελεύθερη της έκδοση όπως και ο συνδυασμός της με τις λύσεις cloud του fluxdb.

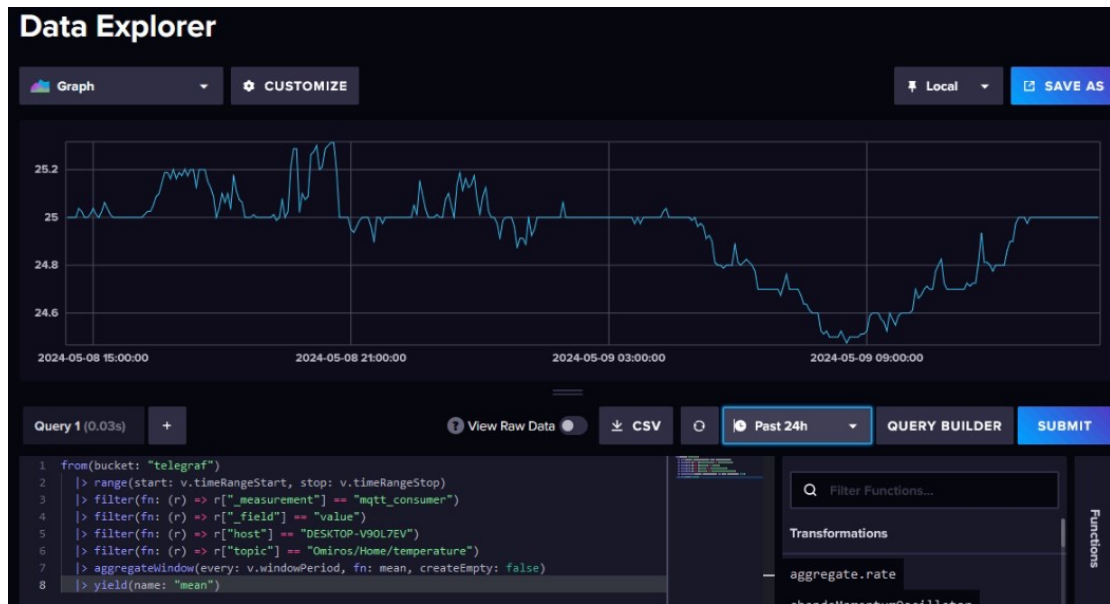
4.7.2 Grafana local

Ιδιαίτερο ενδιαφέρον παρουσιάζει η εγκατάσταση του Grafana και των telegraf & fluxdb τοπικά σε ένα υπολογιστή που θα παίζει το ρόλο του server. Για την τοπική εγκατάσταση λοιπόν αφού κατεβάσουμε και εγκαταστήσουμε τα αρχεία για windows λειτουργικό ανοίγουμε το Grafana μέσω του localhost στη πόρτα 3000. Δημιουργούμε ένα χρήστη με τον κωδικό του και ορίζουμε ένα datasource(εικόνα 40). Εδώ θα ορίσουμε τη τοπική βάση δεδομένων fluxdb που εγκαταστήσαμε σε προηγούμενο βήμα. Μπορούμε να επιλέξουμε τη γλώσσα της βάσης δεδομένων. Επιλέχτηκε η γλώσσα flux έναντι της sql καθώς μας δίνει την ευκαιρία εξοικείωσης με μια νέα τεχνολογία. Ακόμα δηλώνουμε τα στοιχεία του Telegraf(bucket, credentials, url, port). Από αυτό το σημείο μπορούμε να δημιουργούμε ερωτήματα flux και να τα οπτικοποιούμε στο dashboard που θα δημιουργήσουμε. Στο Grafana μπορούμε να αντιγράψουμε ένα dashboard καθώς μας δίνεται το αρχείο json έτσι ώστε να το εισάγουμε και να το επαναχρησιμοποιήσουμε αλλού.



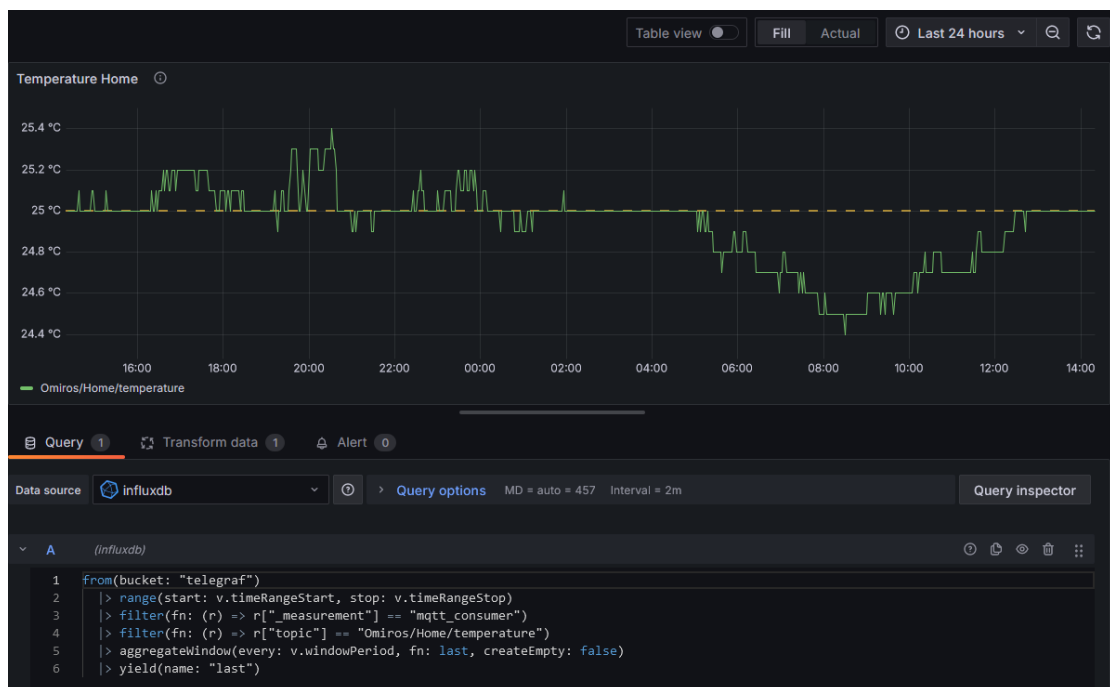
Εικόνα 40 Influxdb configuration

Με τον ίδιο τρόπο λοιπόν όπως και με τη cloud έκδοση χτίζουμε το dashboard τοποθετώντας τα elements της αρεσκείας μας. Με τη βοήθεια του Data explorer το οποίο ακούει στη πόρτα 8086 φτιάχνουμε τα ερωτήματα flux όπως φαίνεται στη παρακάτω εικόνα 41.



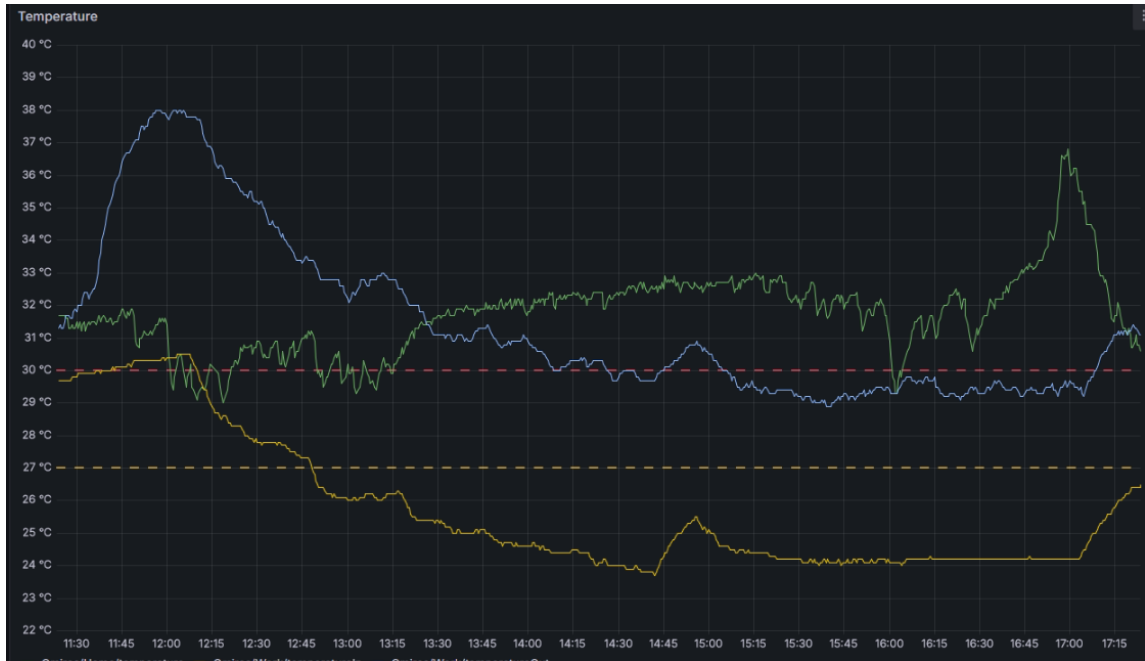
Εικόνα 41 Query builder

Στη συνέχεια μεταφέρουμε το ερώτημα στο element του dashboard όπως φαίνεται παρακάτω.(εικόνα 42)



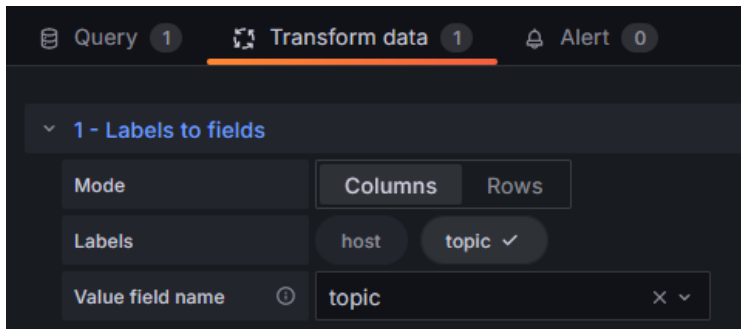
Εικόνα 42 Panel configuration

Μπορούμε να προσθέσουμε κάποια όρια σε συγκεκριμένες τιμές όπως φαίνεται στη εικόνα 16 στο κεφάλαιο 2.1.1. Στην παρακάτω εικόνα 43 βλέπουμε τα όρια(thresholds) στο γράφημα της θερμοκρασίας.



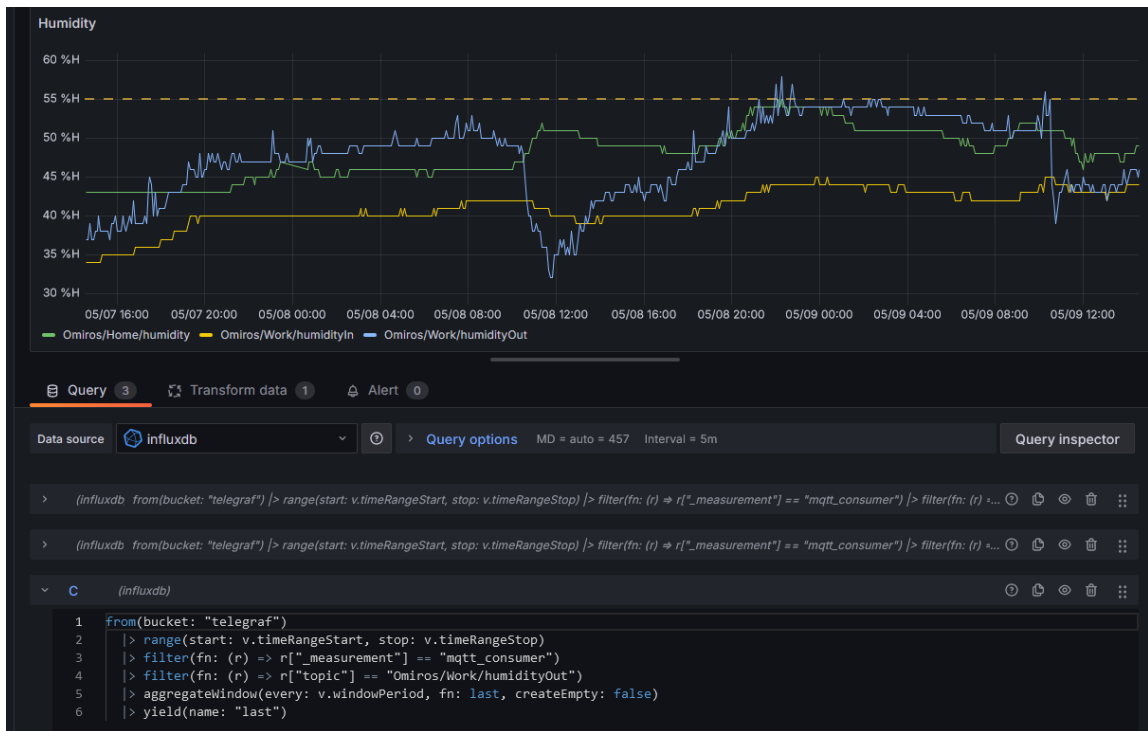
Εικόνα 43 Thresholds

Ακόμα μπορούμε να ορίσουμε τη κεφαλίδα, τα σχόλια, τη μονάδα μέτρησης και τη βαθμονόμηση του γραφήματος. στη παρακάτω εικόνα 44 θέλουμε να εμφανίσουμε το topic σαν legend στο γράφημα μας.



Εικόνα 44 Label transformation

Επίσης μας παρέχεται η δυνατότητα να δημιουργήσουμε γραφήματα συνδυάζοντας ερωτήματα στη βάση δεδομένων. Με αυτό το τρόπο δημιουργούμε γραφήματα που συγκρίνουν τις τιμές από τα τρία διαφορετικά περιβάλλοντα που παρακολουθούμε. Στη παρακάτω εικόνα 45 φαίνεται η διακύμανση της υγρασίας τις τελευταίες δυο μέρες. Στο κάτω μέρος βλέπουμε τα τρία ερωτήματα flux στη βάση δεδομένων.



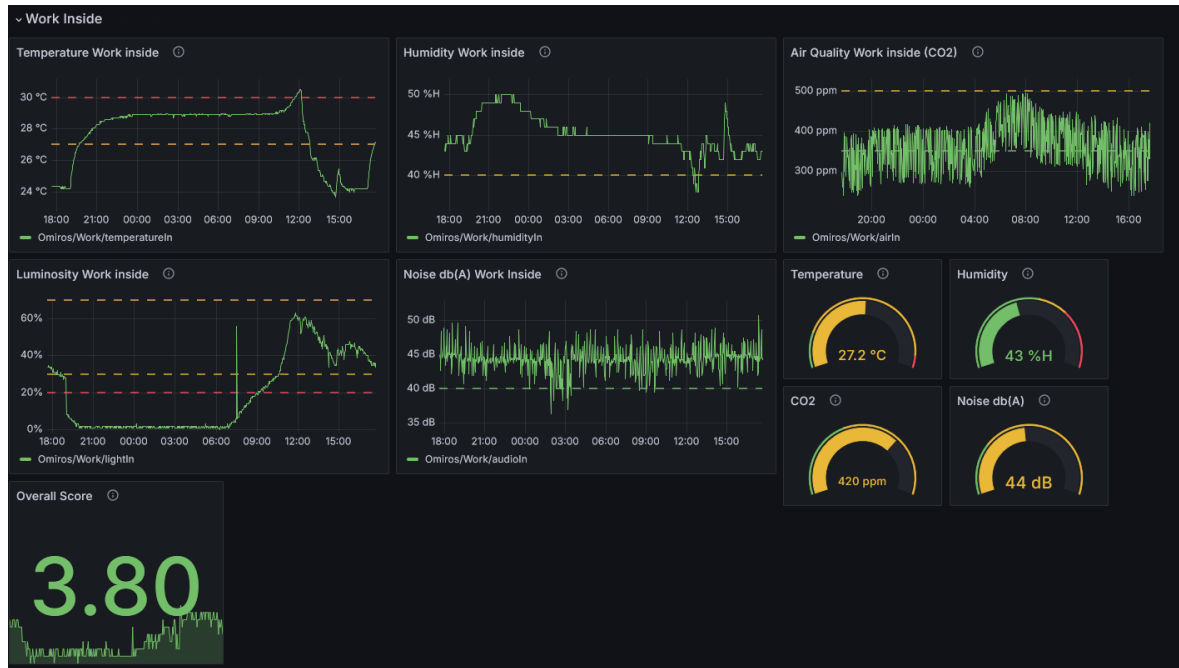
Εικόνα 45 Query combination

Μπορούμε να δημιουργήσουμε tabs στα οποία θα χωρίσουμε τις κατηγορίες που θέλουμε να εμφανίσουμε. Στη παρακάτω εικόνα 46 βλέπουμε τη σύγκριση των τριών περιβαλλόντων (Tab COMPARE) σε όλες τις κατηγορίες. (Θερμοκρασία, υγρασία, φωτεινότητα, ποιότητα αέρα, θόρυβος).



Εικόνα 46 Compare tab

Τα tabs μας βοηθούν να οργανώσουμε σωστά το dashboard. Στη παρακάτω εικόνα 47 βλέπουμε τη κατάσταση του περιβάλλοντος στον εσωτερικό χώρο του γραφείου. Ακόμα βλέπουμε τα υπόλοιπα tabs τα οποία αφορούν τα υπόλοιπα περιβάλλοντα που καταγράφουμε. Ακόμα υπάρχει το tab σύγκρισης και το tab μέσου όρου του τελευταίου 24ωρου. Στη κορυφή βλέπουμε το tab broker health που μας παρέχει πληροφορίες του συστήματος που έχουμε εγκαταστήσει τους servers και του broker.



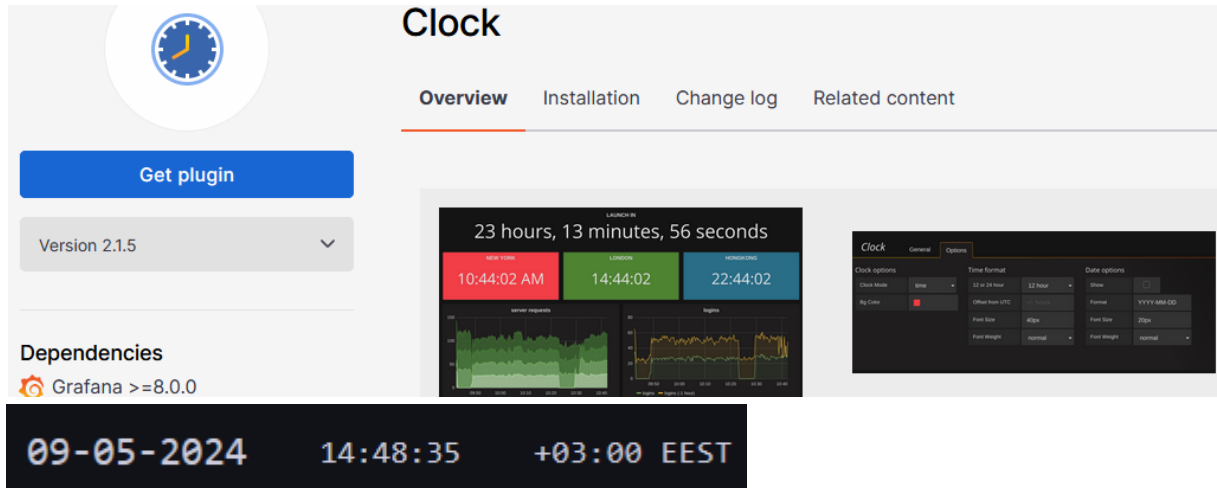
Εικόνα 47 Work inside tab

Στη παρακάτω εικόνα 48 βλέπουμε το element stat με το οποίο εμφανίζουμε τον μέσο όρο(average) των παραμέτρων του τελευταίου 24ωρου.



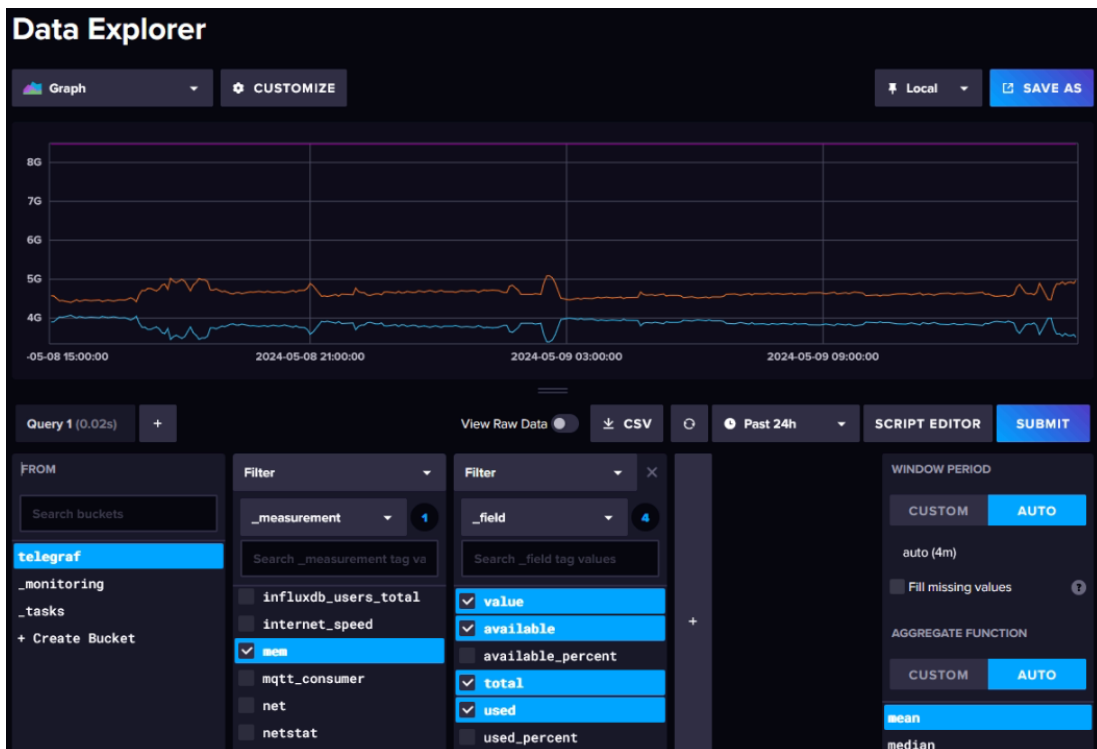
Εικόνα 48 Average tab

Το Grafana μας παρέχει πολλά elements στη βιβλιοθήκη του και ακόμα μας παρέχει τη δυνατότητα να εισάγουμε και επιπρόσθετα. Στη συγκεκριμένη εργασία χρησιμοποιήθηκε το element clock όπως φαίνεται στη παρακάτω εικόνα 49. Αυτό το στοιχείο το εισάγουμε σαν plugin στο Grafana dashboard.



Εικόνα 49 Clock panel plugin

Ιδιαίτερο ενδιαφέρον παρουσιάζει η παρακολούθηση του συστήματος που είναι εγκατεστημένες όλες αυτές οι τεχνολογίες. Με τη βοήθεια του data explorer μπορούμε να παρακολουθήσουμε στοιχεία ευστάθειας του συστήματος. Στη παρακάτω εικόνα βλέπουμε τη παρακολούθηση στοιχείων της μνήμης Ram.(εικόνα 50)



Εικόνα 50 Memory usage query

Επιλέχτηκαν λοιπόν οι τιμές :

- κατανάλωσης πόρων επεξεργαστή καθώς και η συνολική κατανάλωση του
- χρήση της μνήμης Ram
- ταχύτητα σύνδεσης σε Download & Upload
- latency & jitter

Παρακολουθώντας την υγεία του συστήματος μπορούμε να εντοπίσουμε πιθανές δυσλειτουργίες ή επιθέσεις στο σύστημα μας. Για παράδειγμα μια επίθεση Dos θα κατανάλωνε πόρους από τη μνήμη έτσι ώστε να θέσει εκτός λειτουργείας τους server του συστήματος. Στη παρακάτω εικόνα 51 βλέπουμε τα γραφήματα του tab Broker Health.



Εικόνα 51 Broker Health tab

4.7.3 Βαθμολόγηση περιβάλλοντος

Σε αυτή την ενότητα θα υπολογίσουμε τη βαθμολογία για το κάθε περιβάλλον. Η κάθε παράμετρος έχει αντιστοιχιστεί όπως είδαμε στην ενότητα 2.1 με βαθμολογία 1 έως 5. Στο Grafana λοιπόν θα δημιουργήσουμε ένα element στο οποίο θα υπολογίσουμε τη τελική βαθμολογία. Με τον παρακάτω κώδικα διαβάζουμε τα δεδομένα χρονοσειράς από τη βάση δεδομένων της Fluxdb καταγράφουμε τη βαθμολογία για τη θερμοκρασία.

```
// Θερμοκρασία
```

```
temp = from(bucket: "telegraf")
```

```
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
```

```
  |> filter(fn: (r) => r["_measurement"] == "mqtt_consumer")
```

```
  |> filter(fn: (r) => r["topic"] == "Omiros/Work/temperatureOut")
```

```
  |> aggregateWindow(every: v.windowPeriod, fn: last, createEmpty: false)
```

```
  |> map(fn: (r) => ({
```

```
    _time: r._time,
```

```
    temperature:
```

```

    if r._value <= 17.0 then 1.0
    else if r._value <= 20.0 then 3.0
    else if r._value <= 22.0 then 5.0
    else if r._value <= 24.0 then 4.0
    else if r._value <= 27.0 then 3.0
    else if r._value <= 30.0 then 2.0
    else 1.0
  )))
  |> keep(columns: ["_time", "temperature"])

```

Ο κώδικας για τις υπόλοιπες 4 παραμέτρους είναι παρόμοιος και αφού καταγράψουμε τις βαθμολογίες για τις 5 παραμέτρους υπολογίζουμε τη τελική βαθμολογία του σημείου ενδιαφέροντος όπως φαίνεται στον παρακάτω κώδικα. Η βαθμολογία δίνεται από τον τύπο

Εξίσωση 1 Ο τύπος υπολογισμού της βαθμολογίας του σημείου ενδιαφέροντος

$$\text{Overall Score} = \frac{(\text{Θερμοκρασία} + \text{Υγρασία} + \text{Ποιότητα Αέρα} + \text{Θόρυβος} + \text{Φωτεινότητα})}{5}$$

```

overall_combined = join(
  tables: {temp_humidity_air_light: temp_humidity_air_light, noise: noise},
  on: ["_time"],
  method: "inner"
)
|> map(fn: (r) => ({
  _time: r._time,
  overall_score: (
    (r.temperature +
     r.humidity +
     r.air +
     r.light +
     r.noise) / 5.0
  )
}))
|> yield(name: "overall_score") // Αυτό το yield εμφανίζει μόνο το συνολικό σκορ

```

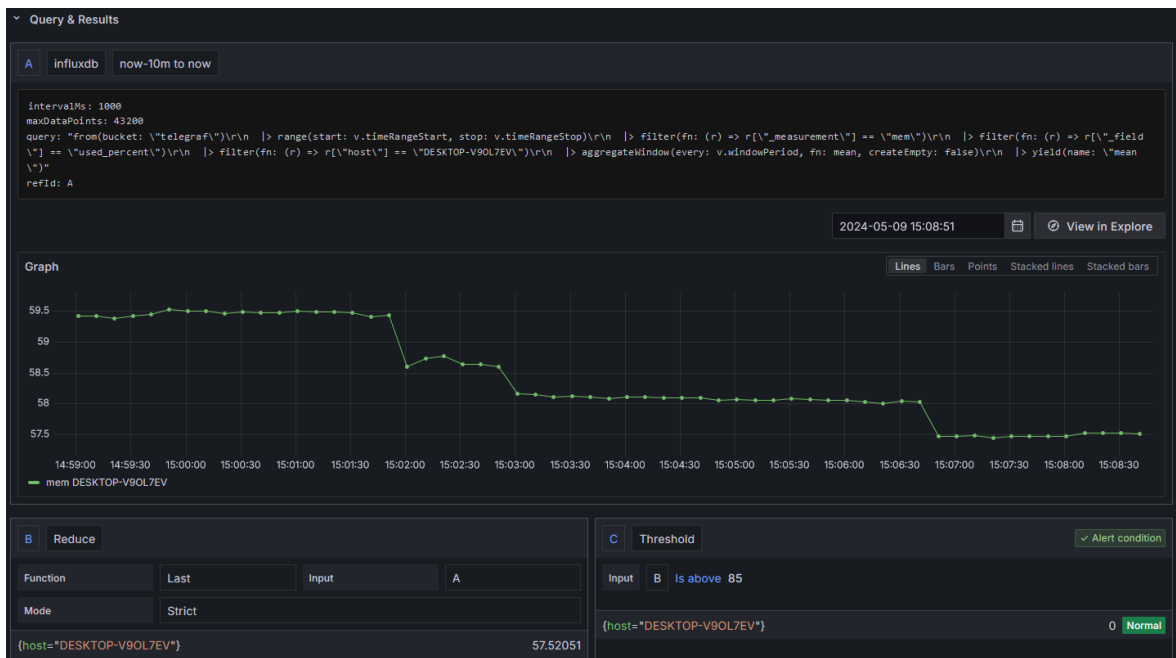
Στη παρακάτω εικόνα 52 βλέπουμε το γράφημα της βαθμολογίας των τριών περιβαλλόντων ταυτόχρονα. Εδώ για παράδειγμα το γραφείο In έχει τη καλύτερη βαθμολογία σε σύγκριση με το γραφείο B2B και σε σύγκριση με τον εξωτερικό χώρο.



Εικόνα 52 Βαθμολογία σημείου ενδιαφέροντος

4.7.3 Alerts

Σε αυτό το σημείο θα αναφερθούμε στη λειτουργία ειδοποιήσεων που μπορούμε να υλοποιήσουμε μέσω του λογισμικού Grafana. Στη παρακάτω εικόνα 53 βλέπουμε τον ορισμό ενός alert που αφορά τη μνήμη Ram. Το όριο που έχουμε θέσει είναι το 85% κατανάλωσης των πόρων της μνήμης.



Εικόνα 53 Alert configuration

Αυτά τα alerts μπορούν να κατευθύνονται σε διαφορετικούς προορισμούς. Μπορούμε λοιπόν να χρησιμοποιήσουμε τα παρακάτω :

- DingDing
- Discord
- Email
- Google Chat
- Kafka
- Line
- Microsoft Teams
- Opsgenie
- Pagerduty
- Prometheus Alertmanager
- Pushover
- Sensu Go
- Slack
- Telegram
- Threema
- VictorOps
- Webhook
- Cisco Webex Teams
- WeCom
- Zenduty

Στη συγκεκριμένη εργασία υλοποιήθηκε το alert στο discord. Δημιουργούμε λοιπόν ένα webhook στο discord και το θέτουμε στο Grafana όπως φαίνεται παρακάτω(εικόνα 54). Μπορούμε να θέσουμε τη περίοδο αποστολής ειδοποίησης, εδώ είναι τα 5 λεπτά.

The screenshot displays the Grafana alert configuration interface, divided into two main sections: '3. Set evaluation behavior' and 'Contact points'.

3. Set evaluation behavior
Define how the alert rule is evaluated. [Need help?](#)

Folder
Select a folder to store your rule.
MemRule

Evaluation group
Rules within the same group are evaluated concurrently over the same time interval.
Perseas Alert

All rules in the selected group are evaluated every 5m.

Pending period
Period in which an alert rule can be in breach of the condition until the alert rule fires.
5m

Pause evaluation

Contact points
Choose how to notify your contact points when an alert instance fires

Create contact point

Name *
Omiros

Integration
Discord

Webhook URL
https://discord.com/api/webhooks/1231030411891834940/r55eQ8dOc6GrrBb...

> Optional Discord settings

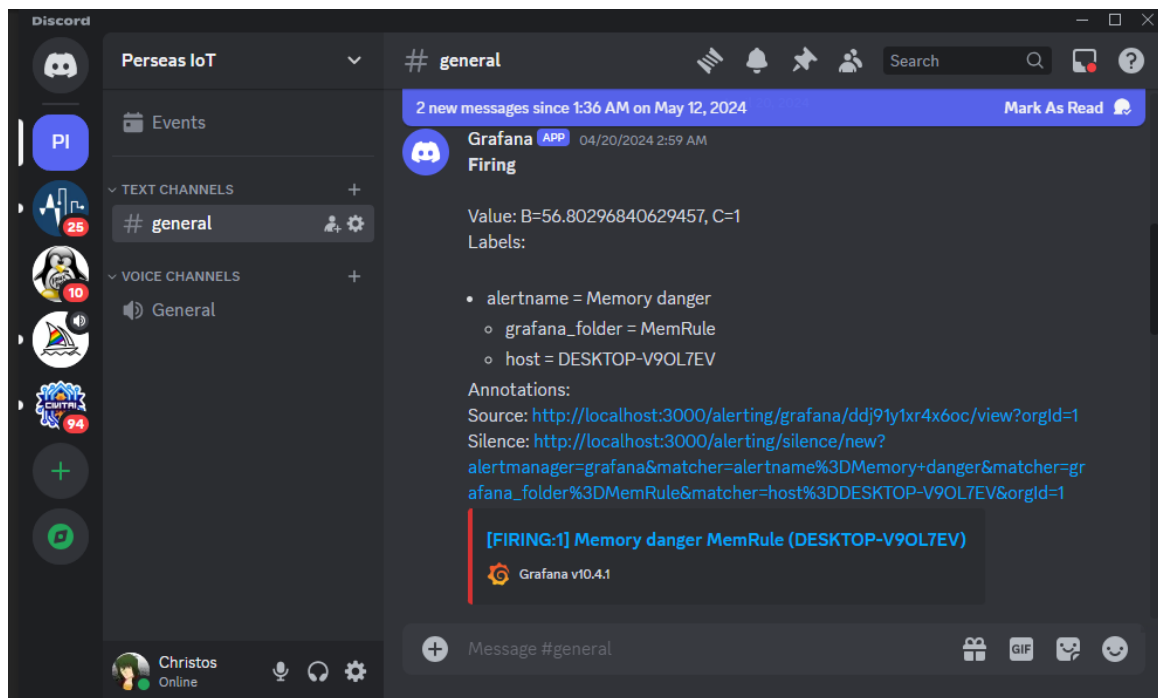
> Notification settings
 Disable resolved message
Disable the resolve message [OK] that is sent when alerting state returns to false

+ Add contact point integration

Save contact point Cancel

Εικόνα 54 Ram alert

Ακόμα πρέπει να δημιουργήσουμε ένα κανάλι αφιερωμένο για αυτά τα alerts στο discord. Στη παρακάτω εικόνα 55 βλέπουμε ένα alert που έχει έρθει για τη κατάσταση της μνήμης Ram στο κανάλι PerseasIoT που δημιουργήθηκε για αυτό το σκοπό.



Εικόνα 55 Discord channel alert

4.8 Βοηθητικά εργαλεία και ρυθμίσεις δικτύου

Τα σημεία ενδιαφέροντος που έχουμε επιλέξει βρίσκονται σε διαφορετικά δίκτυα. Αυτό συνεπάγεται τη χρήση ρυθμίσεων και εργαλείων δικτύου έτσι ώστε να έχουμε τη ζητούμενη επικοινωνία με την απαιτούμενη ασφάλεια.

4.8.1 Port forward

Για να μπορέσει να επικοινωνήσει ο Esp32 στο χώρο του σπιτιού με τον broker που είναι εγκατεστημένος σε άλλο δίκτυο στο γραφείο πρέπει να γίνουν κάποιες ρυθμίσεις δικτύου στον router του γραφείου. Όπως φαίνεται στη παρακάτω εικόνα 56 δημιουργούμε έναν κανόνα όπου ορίζουμε port forward στη πόρτα 55555 από την κίνηση που έρχεται από τη vds1 γραμμή προς την local ip του υπολογιστή που έχουμε εγκαταστήσει τον broker. Έτσι ο esp32 θα συνδέεται στον broker χρησιμοποιώντας την external ip του router στον οποίον είναι συνδεδεμένος ο broker.

Settings for activating the port

Enable	<input checked="" type="checkbox"/>
Name	MQTT BROKER
Protocol	TCP
WAN Host Start IP Address	<input type="text"/> . <input type="text"/> . <input type="text"/> . <input type="text"/>
WAN Host End IP Address	<input type="text"/> . <input type="text"/> . <input type="text"/> . <input type="text"/>
WAN Connection	Internet_VDSL
WAN Start Port	<input type="text" value="55555"/> (1-65535)
WAN End Port	<input type="text" value="55555"/> (1-65535)
Enable MAC Mapping	<input type="checkbox"/>
LAN Host IP Address	<input type="text" value="192"/> . <input type="text" value="168"/> . <input type="text" value="1"/> . <input type="text" value="7"/>
LAN Host Start Port	<input type="text" value="55555"/> (1-65535)
LAN Host End Port	<input type="text" value="55555"/> (1-65535)

Εικόνα 56 Port Forward

Ακόμα πρέπει να ορίσουμε κανόνα για το windows defender firewall. Με τη παρακάτω εντολή επιτρέπουμε την εισερχόμενη κίνηση στη πόρτα 55555

```
New-NetFirewallRule -DisplayName "Allow MQTT Traffic" -Direction Inbound -Protocol TCP -LocalPort 55555 -Action Allow
```

Επίσης μπορούμε να προωθήσουμε την κίνηση που έρχεται στην local ip του υπολογιστή που έχουμε εγκαταστήσει τον broker προς τη πόρτα 55555 που ακούει η υπηρεσία mosquitto mqtt.

```
netsh interface portproxy add v4tov4 listenport=55555 listenaddress=192.168.1.7 connectport=55555 connectaddress=127.0.0.1
```

Για να διατηρήσουμε σταθερή τη local ip του server μας μπορούμε να τη δηλώσουμε χρησιμοποιώντας τη mac address στον router όπως φαίνεται παρακάτω(Εικόνα 57). Ακόμα χρησιμοποιούμε σταθερή ip στις ρυθμίσεις των windows.

▼ DHCP [What is DHCP?](#)

DHCP off
 DHCP on

Assign addresses in the range from
to

Validity period for the addresses

DHCP relay

Reserved IP Address

MAC Address ▼	IP Address ▼	
30-9C-23-7C-87-40	192.168.1.7	<input type="button" value="x"/> delete

Manually add device

Obtain an IP address automatically
 Use the following IP address:

IP address:
Subnet mask:
Default gateway:

Obtain DNS server address automatically
 Use the following DNS server addresses:

Preferred DNS server:
Alternate DNS server:

Εικόνα 57 Server static local ip

4.8.2 IPv6 Filter

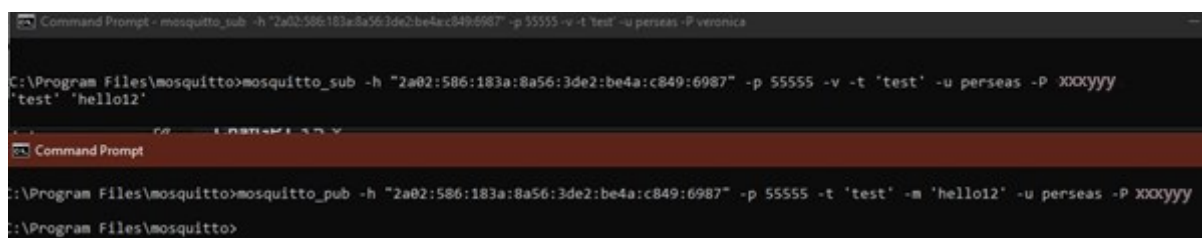
Ένας άλλος τρόπος να συνδεθούμε στο broker από το internet είναι χρησιμοποιώντας την ipv6 του server. Ενεργοποιούμε λοιπόν τη ipv6 στον υπολογιστή που έχουμε εγκαταστήσει το broker και ορίζουμε την ipv6 αυτή σαν προορισμό στον κανόνα ipv6 filter για εισερχόμενη κίνηση στη πόρτα 55555.(εικόνα 58)

Settings for the IPv6 Filter

Enable	<input checked="" type="checkbox"/>
Protocol	TCP and UDP
Name	Broker
Incoming Traffic	Internet_VDSL
Outgoing Traffic	LAN
Start Source IPv6 Address	::
End Source IPv6 Address	::
Start Destination IPv6 Address	2a02:586:183a:8ab7:7ef1:4587:f8b0::
End Destination IPv6 Address	2a02:586:183a:8ab7:7ef1:4587:f8b0::
Start Source port	1 (1-65535)
End Source port	65535 (1-65535)
Start Destination port	55555 (1-65535)
End Destination port	55555 (1-65535)
Mode	Permit

Εικόνα 58 IPv6 Filter

Στη παρακάτω εικόνα 59 βλέπουμε τη δοκιμαστική σύνδεση όπου ο ένας client κάνει publish και ο άλλος client κάνει subscribe μέσω της σύνδεσης στον broker από την ipv6 που έχει ο υπολογιστής που εγκαταστάθηκε ο broker.



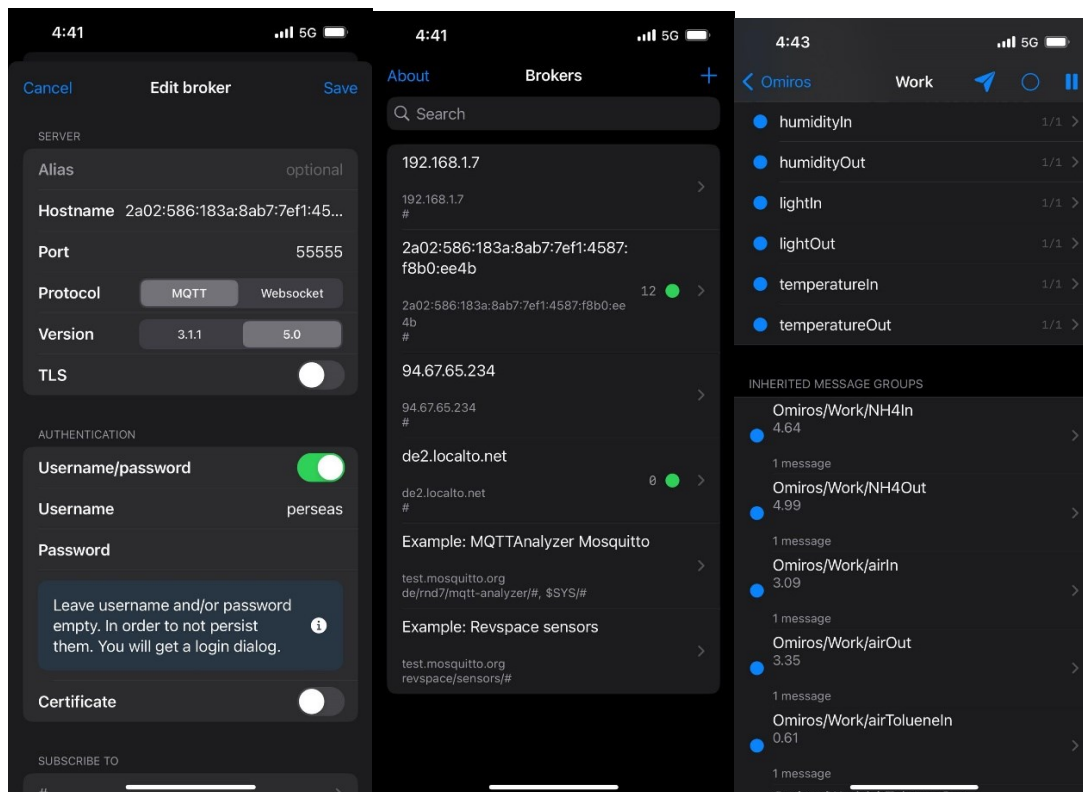
```
Command Prompt - mosquitto_sub -h "2a02:586:183a:8a56:3de2:be4a:c849:6987" -p 55555 -v -t test -u perseas -P xxxxyy
C:\Program Files\mosquitto>mosquitto_sub -h "2a02:586:183a:8a56:3de2:be4a:c849:6987" -p 55555 -v -t 'test' -u perseas -P xxxxyy
'test' 'hello12'

Command Prompt
C:\Program Files\mosquitto>mosquitto_pub -h "2a02:586:183a:8a56:3de2:be4a:c849:6987" -p 55555 -t 'test' -m 'hello12' -u perseas -P xxxxyy
C:\Program Files\mosquitto>
```

Εικόνα 59 Mosquitto testing

4.8.3 Mqtt Analyzer

Σε αυτό το σημείο θα αναφερθούμε στο εργαλείο διαχείρισης mqtt Analyzer για κινητές συσκευές. Στη παρακάτω εικόνα 60 βλέπουμε τη ρύθμιση της σύνδεσης στην ipν6 του server μας(αριστερά). Ακόμα βλέπουμε τη δυνατότητα σύνδεσης σε πολλούς brokers ταυτόχρονα(κέντρο). Τέλος βλέπουμε τα topics με τις τιμές που επιστρέφουν(δεξιά).



Εικόνα 60 Mqtt Analyzer

4.8.4 Expose broker (Tunnel)

Εναλλακτικά με το port forward και του ipν6 filtering μπορούμε να χρησιμοποιήσουμε κάποιες υπηρεσίες οι οποίες θα μας επιτρέψουν να κάνουμε τον broker μας διαθέσιμο στο internet. Στη παρακάτω εικόνα 61 βλέπουμε δυο τέτοιες υπηρεσίες με τις οποίες επιτυγχάνεται η παραπάνω λειτουργία χωρίς να χρειάζεται κάποια δικτυακή ρύθμιση. Αυτή η λύση είναι πιο απλή παρακάμπτοντας πιθανά προβλήματα CG Nat, firewalls που μπορεί να προκύψουν από τον πάροχο του internet. Η υπηρεσία Ngrok στη συγκεκριμένη εικόνα προωθεί τη κίνηση της πόρτας 1883(mosquitto broker) στο host 2.tcp.eu.ngrok.io στη πόρτα 16354 και έτσι μπορούμε να συνδεθούμε στο broker από το internet από οποιοδήποτε δίκτυο βλέπει το internet.

```

Command Prompt - ngrok tcp 1883

ngrok

Take our ngrok in production survey! https://forms.gle/aXiBFwzEA36DudFn6

Session Status      online
Account             apolloncity@gmail.com (Plan: Free)
Version             3.8.0
Region              Europe (eu)
Latency             49ms
Web Interface       http://127.0.0.1:4040
Forwarding          tcp://2.tcp.eu.ngrok.io:16354 -> localhost:1883

Connections         ttl      opn      rt1      rt5      p50      p90
                   0        0        0.00    0.00    0.00    0.00

```

Εικόνα 61 Ngrok service

Εναλλακτικά η υπηρεσία της localto δίνει λιγότερους περιορισμούς από το Ngrok και επιλέχτηκε να φιλοξενεί τον broker μας. Στη παρακάτω εικόνα 62 βλέπουμε τη προώθηση της κίνησης localhost της πόρτας του broker 55555 στο host που μας παρέχει η υπηρεσία.

```

[ Session Status: Connected ]

IP/Url      Protocol Type  Client IP  Client Port  Ping  Status
-----
de2.localto.net:37577  TCP           127.0.0.1  55555       99    OK

Token: Default                                     Plan: Free

Localtonet Version v5.3  New Version available

```

Εικόνα 62 Localto service

Ο περιορισμός της υπηρεσίας localto έχει να κάνει με την χρήση των δεδομένων. Στη παρακάτω εικόνα 63 βλέπουμε τη χρήση που έχουμε κάνει 20 μέρες περίπου με συνεχή λειτουργία του broker. Ακόμα παρατηρούμε την αρχιτεκτονική της υπηρεσίας. Κατά τη διαδικασία εγγραφής στην υπηρεσία μας παρέχεται ένα ευχρηστο εργαλείο μέτρησης του ογκου των δεδομένων που θα χρειαστούμε.



Εικόνα 63 Αρχιτεκτονική Localtonet

Ένα ακόμη εργαλείο tunnel είναι το localtunnel. Αυτό το εργαλείο απαιτεί εγκατάσταση του node.js και μας επιτρέπει να μπορούμε να δώσουμε τους server μας στο internet. Μετά την εγκατάσταση του node.js εγκαθιστούμε το localtunnel όπως φαίνεται παρακάτω (εικόνα 64) και το εκκινούμε για τη porta 3000. Αμέσως μας εμφανίζεται ένα link από το οποίο μπορούμε να δούμε το grafana από το internet.

```
C:\Windows\system32\cmd.exe - "node" "C:\Users\PERSEAS\AppData\Roaming\npm\node_modules\localtunnel\bin\lt.js" --port 3000
Your environment has been set up for using Node.js 22.1.0 (x64) and npm.

C:\Users\PERSEAS>npm install -g localtunnel

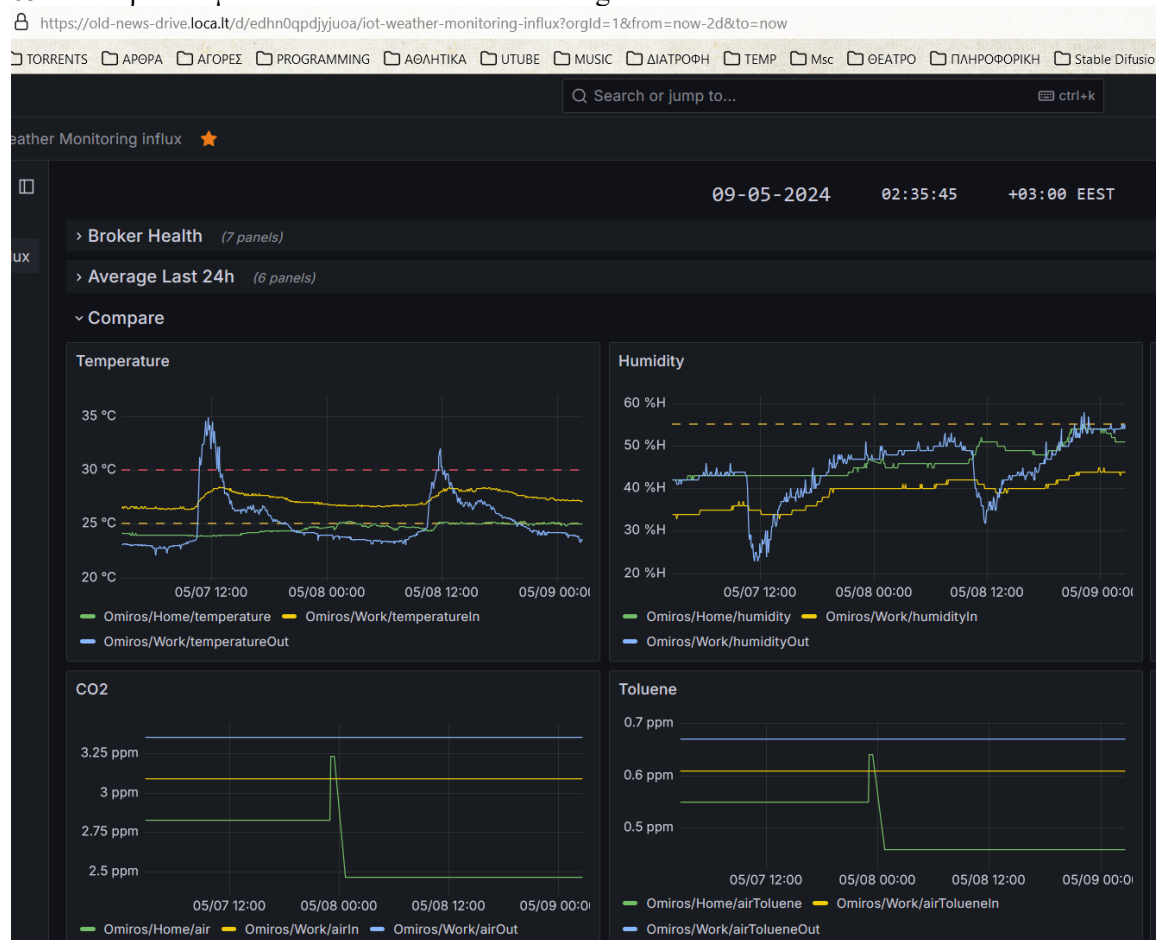
added 22 packages in 6s

3 packages are looking for funding
  run `npm fund` for details

C:\Users\PERSEAS>lt --port 3000
your url is: https://old-news-drive.loca.lt
```

Εικόνα 64 Localtunnel service

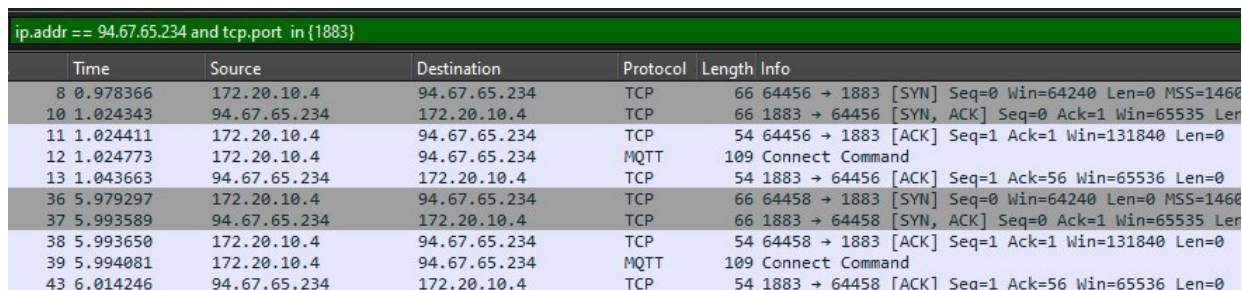
Στο επόμενο βήμα μας ζητείται την eternal ip του router που βρίσκεται ο broker και έπειτα γίνεται η σύνδεση. Το συγκεκριμένο εργαλείο είναι πολύ χρήσιμο αν θέλουμε να έχουμε πρόσβαση σε μια υπηρεσία που τρέχει σε κάποια porta στο localhost. Στη παρακάτω εικόνα 65 βλέπουμε το local grafana από το internet.



Εικόνα 65 Σύνδεση στο Local Grafana από το internet

4.9 Troubleshooting

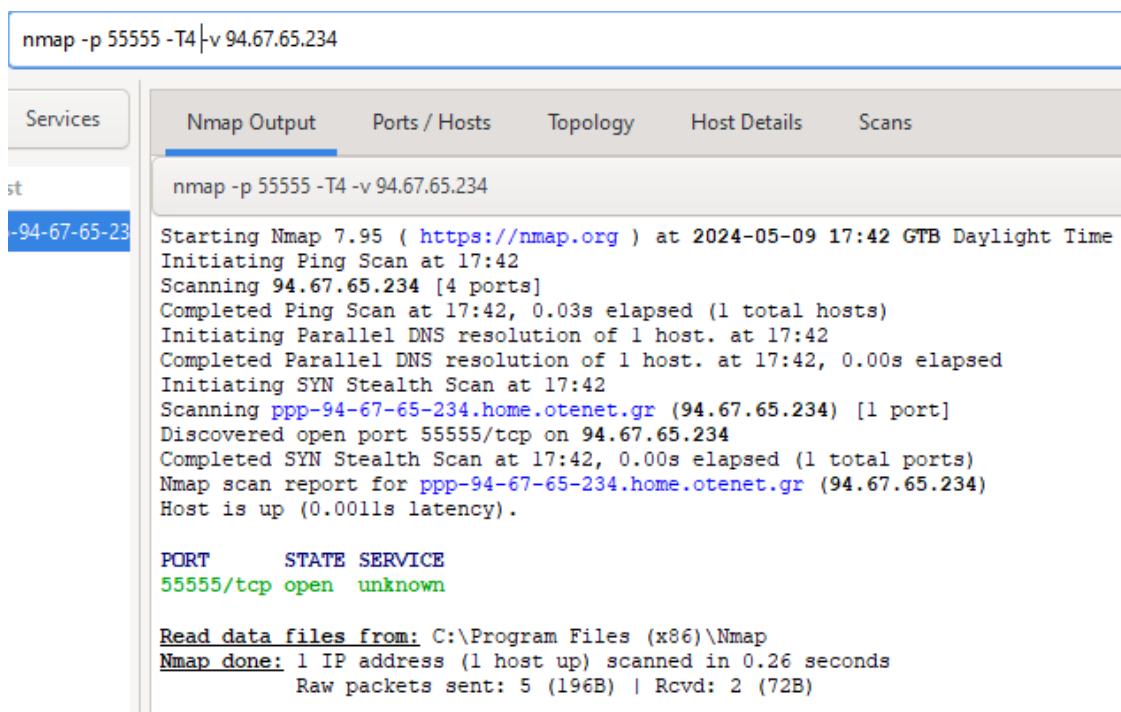
Για την αντιμετώπιση προβλημάτων χρησιμοποιήθηκαν εργαλεία δικτύου όπως **wireshark** & **nmap**. Στη παρακάτω εικόνα 66 βλέπουμε τη προσπάθεια σύνδεσης σε επίπεδο tcp στη πόρτα 1883 του mosquito broker.



Time	Source	Destination	Protocol	Length	Info
8 0.978366	172.20.10.4	94.67.65.234	TCP	66	64456 → 1883 [SYN] Seq=0 Win=64240 Len=0 MSS=1460
10 1.024343	94.67.65.234	172.20.10.4	TCP	66	1883 → 64456 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0
11 1.024411	172.20.10.4	94.67.65.234	TCP	54	64456 → 1883 [ACK] Seq=1 Ack=1 Win=131840 Len=0
12 1.024773	172.20.10.4	94.67.65.234	MQTT	109	Connect Command
13 1.043663	94.67.65.234	172.20.10.4	TCP	54	1883 → 64456 [ACK] Seq=1 Ack=56 Win=65536 Len=0
36 5.979297	172.20.10.4	94.67.65.234	TCP	66	64458 → 1883 [SYN] Seq=0 Win=64240 Len=0 MSS=1460
37 5.993589	94.67.65.234	172.20.10.4	TCP	66	1883 → 64458 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0
38 5.993650	172.20.10.4	94.67.65.234	TCP	54	64458 → 1883 [ACK] Seq=1 Ack=1 Win=131840 Len=0
39 5.994081	172.20.10.4	94.67.65.234	MQTT	109	Connect Command
43 6.014246	94.67.65.234	172.20.10.4	TCP	54	1883 → 64458 [ACK] Seq=1 Ack=56 Win=65536 Len=0

Εικόνα 66 Wireshark

Στη παρακάτω εικόνα 67 ελέγχουμε αν η πόρτα 55555 που έχουμε ορίσει είναι ανοιχτή για εισερχόμενες συνδέσεις.



```
nmap -p 55555 -T4 -v 94.67.65.234
```

Services	Nmap Output	Ports / Hosts	Topology	Host Details	Scans
st	<pre>nmap -p 55555 -T4 -v 94.67.65.234</pre>				

```
Starting Nmap 7.95 ( https://nmap.org ) at 2024-05-09 17:42 GTB Daylight Time
Initiating Ping Scan at 17:42
Scanning 94.67.65.234 [4 ports]
Completed Ping Scan at 17:42, 0.03s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 17:42
Completed Parallel DNS resolution of 1 host. at 17:42, 0.00s elapsed
Initiating SYN Stealth Scan at 17:42
Scanning ppp-94-67-65-234.home.otenet.gr (94.67.65.234) [1 port]
Discovered open port 55555/tcp on 94.67.65.234
Completed SYN Stealth Scan at 17:42, 0.00s elapsed (1 total ports)
Nmap scan report for ppp-94-67-65-234.home.otenet.gr (94.67.65.234)
Host is up (0.0011s latency).

PORT      STATE SERVICE
55555/tcp open  unknown

Read data files from: C:\Program Files (x86)\Nmap
Nmap done: 1 IP address (1 host up) scanned in 0.26 seconds
Raw packets sent: 5 (196B) | Rcvd: 2 (72B)
```

Εικόνα 67 Nmap

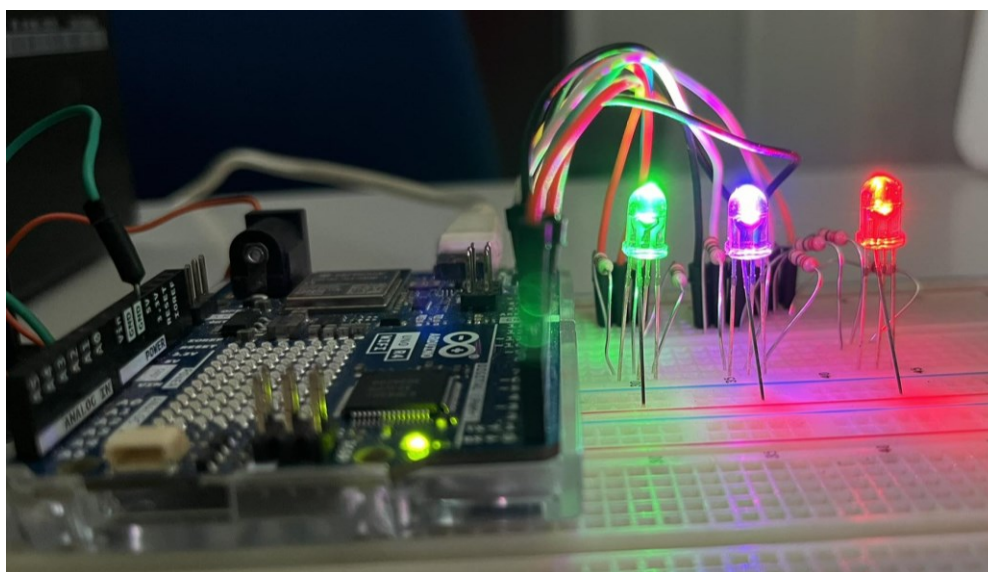
Ακόμα χρησιμοποιήθηκαν τα logs του Telegraf και του Mosquitto έτσι ώστε να επιλυθούν προβλήματα. Στη παρακάτω εικόνα 68 βλέπουμε το log του mosquitto όπου ακουει συνδέσεις ipv4 & ipv6 στη πορτα 55555 και στη συνεχεια βλεπουμε τη δοκιμαστικη συνδεση σε επιπεδο ipv6 με το μηνυμα στο topic test. Στη συνεχεια βλεπουμε το αιτημα για subscribe στα topics των αισθητηρων από το το Telegraf.

```
mosquitto.log - Notepad
File Edit Format View Help
1714556973: Client ESP32Client already connected, closing old connection.
1714556974: Client ESP32Client already connected, closing old connection.
1714556987: mosquitto version 2.0.18 starting
1714556987: Config loaded from C:\Program Files\mosquitto\mosquitto.conf.
1714556987: Opening ipv6 listen socket on port 55555.
1714556987: Opening ipv4 listen socket on port 55555.
1714556987: mosquitto version 2.0.18 running
1714556988: New connection from 2a02:586:183a:8a56:cc87:d8e8:722f:a1fe:5293 on port 55555.
1714556988: New client connected from 2a02:586:183a:8a56:cc87:d8e8:722f:a1fe:5293 as auto-C24A1C25-A16C-13BE-0AC!
1714556988: No will message specified.
1714556988: Sending CONNACK to auto-C24A1C25-A16C-13BE-0AC5-F2E5108932EB (0, 0)
1714556988: Received SUBSCRIBE from auto-C24A1C25-A16C-13BE-0AC5-F2E5108932EB
1714556988:   'test' (QoS 0)
1714556988: auto-C24A1C25-A16C-13BE-0AC5-F2E5108932EB 0 'test'
1714556988: Sending SUBACK to auto-C24A1C25-A16C-13BE-0AC5-F2E5108932EB
1714556990: New connection from 192.168.1.1:44877 on port 55555.
1714556990: New client connected from 192.168.1.1:44877 as Telegraf-Consumer-Mu11F (p2, c1, k60, u'perseas').
1714556990: No will message specified.
1714556990: Sending CONNACK to Telegraf-Consumer-Mu11F (0, 0)
1714556990: Received SUBSCRIBE from Telegraf-Consumer-Mu11F
1714556990:   Omiros/Home/airToluene (QoS 0)
1714556990: Telegraf-Consumer-Mu11F 0 Omiros/Home/airToluene
1714556990:   Omiros/Home/NH4 (QoS 0)
1714556990: Telegraf-Consumer-Mu11F 0 Omiros/Home/NH4
1714556990:   Omiros/Work/humidityOut (QoS 0)
1714556990: Telegraf-Consumer-Mu11F 0 Omiros/Work/humidityOut
1714556990:   Omiros/Work/NH4Out (QoS 0)
1714556990: Telegraf-Consumer-Mu11F 0 Omiros/Work/NH4Out
1714556990:   Omiros/Work/airTolueneOut (QoS 0)
1714556990: Telegraf-Consumer-Mu11F 0 Omiros/Work/airTolueneOut
1714556990:   Omiros/Work/lightOut (QoS 0)
```

Εικόνα 68 Mosquitto Log

4.10 Subscribe & λήψη αποφάσεων

Έχοντας στήσει το παραπάνω σύστημα παρακολούθησης του περιβάλλοντος μπορούμε να δημιουργήσουμε αυτοματισμούς σύμφωνα με τις ανάγκες μας. Ενδεικτικά με τη χρήση ενός Arduino r4 WiFi θα λάβουμε τις τιμές της θερμοκρασίας από τα τρία περιβάλλοντα και θα εμφανίσουμε το κατάλληλο χρώμα σε τρία rgb leds. Αντί για leds θα μπορούσαμε να είχαμε συστήματα ψύξης/θέρμανσης, αφύγρανσης, καθαρισμού αέρα ή αυτομάτου φωτισμού. Στη παρακάτω εικόνα 69 βλέπουμε τη συνδεσμολογία των τριών rgb leds με τις αντιστάσεις 220ohm. Αριστερά βλέπουμε την ένδειξη για το σπίτι(<25), στο κέντρο για τον εσωτερικό χώρο του γραφείου(25-30) και δεξιά για τον εξωτερικό χώρο(>30).



Εικόνα 69 Subscribe λειτουργία

Ξεκινώντας την ανάλυση του κώδικα παρατηρούμε ότι το Arduino r4 χρησιμοποιεί διαφορετικές βιβλιοθήκες τη σύνδεση WiFi και τη λειτουργία mqtt. Στη συνέχεια ορίζουμε τα pins σύνδεσης των RGB leds και τα credentials του WiFi δικτύου και broker.

```
#include "WiFiS3.h"
#include <ArduinoMqttClient.h>

int redPin = 2;
int greenPin = 3;
int bluePin = 4;
int redPinIn = 5;
int greenPinIn = 6;
int bluePinIn = 7;
int redPinOut = 8;
int greenPinOut = 9;
int bluePinOut = 10;
byte mac[6];
// WiFi work
char ssid[] = "DATA"; // Enter your WiFi name
char pass[] = "xxxxyyyzzz"; // Enter WiFi password
char mqtt_user[] = "perseas";
char mqtt_pass[] = "xxxxyyyy";
```

Έπειτα ορίζουμε τα στοιχεία σύνδεσης του broker ανάλογα το δίκτυο που βρισκόμαστε, καθώς και τα topics στα οποία θέλουμε να κάνουμε subscribe. Παρακάτω θέτουμε τα pins των led σε λειτουργία εξόδου και εκκινούμε τη σύνδεση WiFi. Αφού τυπώσουμε τις πληροφορίες που μας ενδιαφέρουν κάνουμε σύνδεση στον mqtt broker.

```
WiFiClient wifiClient;
MqttClient mqttClient(wifiClient);
float temperature = 0.0;
float temperatureIn = 0.0;
float temperatureOut = 0.0;
const char broker[] = "192.168.1.7";
int port = 55555;
const char subscribe_topic[] = "Omiros/Home/temperature";
const char subscribe_topicIn[] = "Omiros/Work/temperatureIn";
const char subscribe_topicOut[] = "Omiros/Work/temperatureOut";
void setup() {
  pinMode(redPin, OUTPUT);
  pinMode(greenPin, OUTPUT);
  pinMode(bluePin, OUTPUT);
  pinMode(redPinIn, OUTPUT);
  pinMode(greenPinIn, OUTPUT);
  pinMode(bluePinIn, OUTPUT);
  pinMode(redPinOut, OUTPUT);
  pinMode(greenPinOut, OUTPUT);
  pinMode(bluePinOut, OUTPUT);
```

```

// Create serial connection and wait for it to become available.
Serial.begin(9600);
while (!Serial) {
  ;
}
// Connect to WiFi
Serial.print("Attempting to connect to WPA SSID: ");
Serial.println(ssid);
while (WiFi.begin(ssid, pass) != WL_CONNECTED) {
  // failed, retry
  Serial.print(".");
  delay(5000);
}
Serial.print("Connected, IP address: ");
Serial.print(WiFi.localIP());
Serial.print(", RRSI: ");
Serial.println(WiFi.RSSI());
Serial.print(" MAC address : ");
WiFi.macAddress(mac);
for (int i=1;i<5;i++){
  Serial.print(mac[i],HEX);Serial.print(":");
}Serial.println();
// You can provide a username and password for authentication
mqttClient.setUsernamePassword(mqtt_user, mqtt_pass);
Serial.print("Attempting to connect to the MQTT broker.");
while (!mqttClient.connect(broker, port)) {
  Serial.print("MQTT connection failed! Error code = ");
  Serial.println(mqttClient.connectError());
  delay(4000);
}
Serial.println("You're connected to the MQTT broker!");

```

Έπειτα λαμβάνουμε τις τιμές από τα topics της θερμοκρασίας και τις ελέγχουμε. Έχουμε τρεις περιπτώσεις :

- <25 -> πράσινο
- 25-30 -> μωβ
- >30 -> κόκκινο

Εμφανίζουμε λοιπόν το αντίστοιχο χρώμα δίνοντας τις κατάλληλες τιμές στα RGB leds, για τα τρία διαφορετικά περιβάλλοντα.

```

Serial.print("Subscribing to topic: ");
Serial.println(subscribe topic);
Serial.println(subscribe topicIn);
Serial.println(subscribe_topicOut);
// subscribe to a topic

```

```

mqttClient.subscribe(subscribe_topic);
mqttClient.subscribe(subscribe_topicIn);
mqttClient.subscribe(subscribe_topicOut);
Serial.println("Waiting for messages on topic: ");
Serial.println(subscribe_topic);
Serial.println(subscribe_topicIn);
Serial.println(subscribe_topicOut);
}
void loop() {
  int messageSize = mqttClient.parseMessage();

  if (messageSize) {
    // we received a message, print out the topic and contents
    String topicRecieved = mqttClient.messageTopic();
    Serial.print("Received a message with topic :");
    Serial.print(topicRecieved);
    Serial.print(", length ");
    Serial.print(messageSize);
    Serial.print(" bytes:");
    if(topicRecieved=="Omiros/Home/temperature"){
      String message = mqttClient.readStringUntil('\n');
      temperature = atof(message.c_str());
      Serial.println(message);
      if (temperature <= 25 ) {
        digitalWrite(redPin, HIGH);//green
        digitalWrite(greenPin, HIGH);
        digitalWrite(bluePin, LOW);
      }
    }
    else if(temperature > 30){
      digitalWrite(redPin, LOW);//red
      digitalWrite(greenPin, HIGH);
      digitalWrite(bluePin, HIGH);
    }
    else {
      digitalWrite(redPin, LOW);//magenta
      digitalWrite(greenPin, LOW);
      digitalWrite(bluePin, HIGH);
    }
  }
  else if(topicRecieved == "Omiros/Work/temperatureIn"){
    String messageIn = mqttClient.readStringUntil('\n');
    temperatureIn = atof(messageIn.c_str());
    Serial.println(messageIn);
    if (temperatureIn <= 25 ) {
      digitalWrite(redPinIn, HIGH);//green
      digitalWrite(greenPinIn, HIGH);
      digitalWrite(bluePinIn, LOW);
    }
  }
}

```

```

else if(temperatureIn > 30){
  digitalWrite(redPinIn, LOW); //red
  digitalWrite(greenPinIn, HIGH);
  digitalWrite(bluePinIn, HIGH);
}
else {
  digitalWrite(redPinIn, LOW); //magenta
  digitalWrite(greenPinIn, LOW);
  digitalWrite(bluePinIn, HIGH);
}
}
else if(...
}
delay(1000);
}

```

Στην παρακάτω εικόνα 70 βλέπουμε το serial monitor του Arduino στο οποίο έχει γίνει σύνδεση και έχουμε τυπώσει :

- local ip
- ένταση σήματος RRSI
- firmware
- mac address
- τα topics
- οι τιμές των topics σε πραγματικό χρόνο

```

Attempting to connect to WPA SSID: DATA
Connected, IP address: 192.168.1.10, RRSI: -40, firmware: 0.4.1, MAC address : 54:75:C3:AB:
Attempting to connect to the MQTT broker.You're connected to the MQTT broker!
Subscribing to topic: Omiros/Home/temperature
Omiros/Work/temperatureIn
Omiros/Work/temperatureOut
Waiting for messages on topic:
Omiros/Home/temperature
Omiros/Work/temperatureIn
Omiros/Work/temperatureOut
Received a message with topic :Omiros/Work/temperatureOut, length 5 bytes:24.60
Received a message with topic :Omiros/Home/temperature, length 5 bytes:24.50
Received a message with topic :Omiros/Work/temperatureIn, length 5 bytes:26.80

```

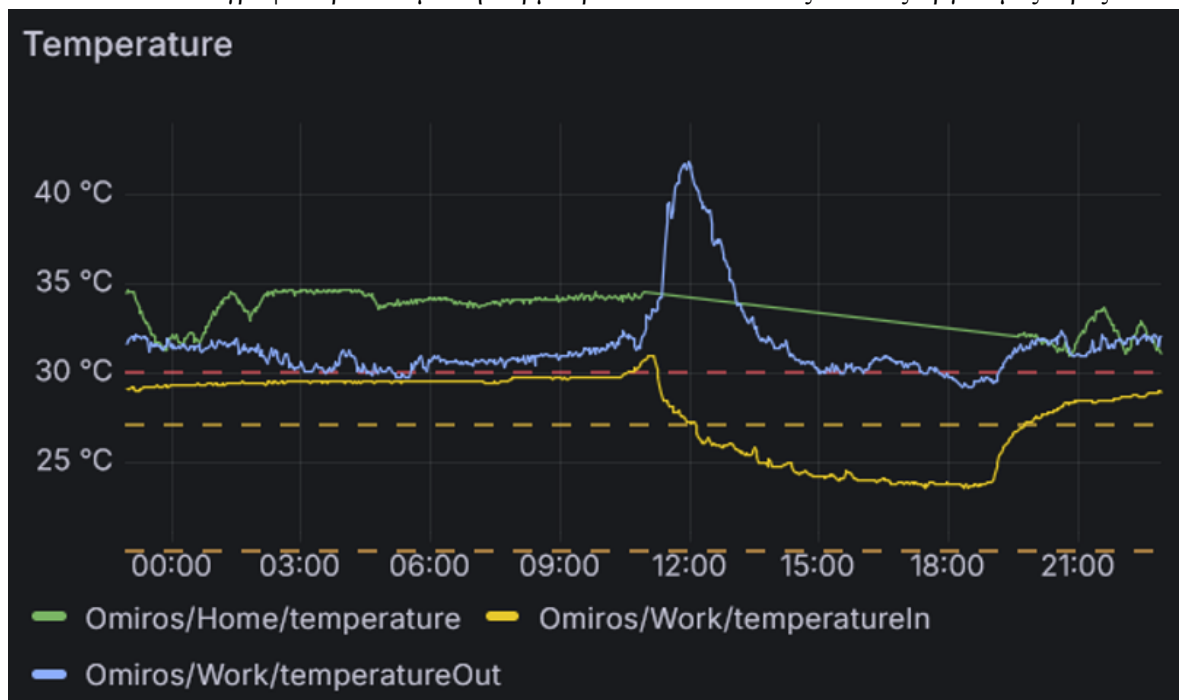
Εικόνα 70 Λειτουργία subscribe

5 ΑΠΟΤΕΛΕΣΜΑΤΑ

Το σύστημα IoT που υλοποιήθηκε έμεινε σε λειτουργία τους τελευταίους 5 μήνες και εξυπηρέτησε τα δυο σενάρια που αναφέρθηκαν στην ενότητα 3.1. Αυτό το διάστημα εγκαταστάθηκε μια οθόνη προβολής των γραφημάτων έτσι ώστε οι εργαζόμενοι να βλέπουν σε πραγματικό χρόνο τη κατάσταση του περιβάλλον που εργάζονται. Τον τελευταίο μήνα δόθηκε η ευκαιρία να μπορούν να δουν από τον προσωπικό τους υπολογιστή την ανάλυση των δεδομένων για το περιβάλλον εργασίας τους. Αυτό τους έδωσε την ευκαιρία να έχουν μια αντικειμενική άποψη για το χώρο που εργάζονται και να ενδιαφέρονται για τη διατήρηση του καλύτερου περιβάλλοντος.

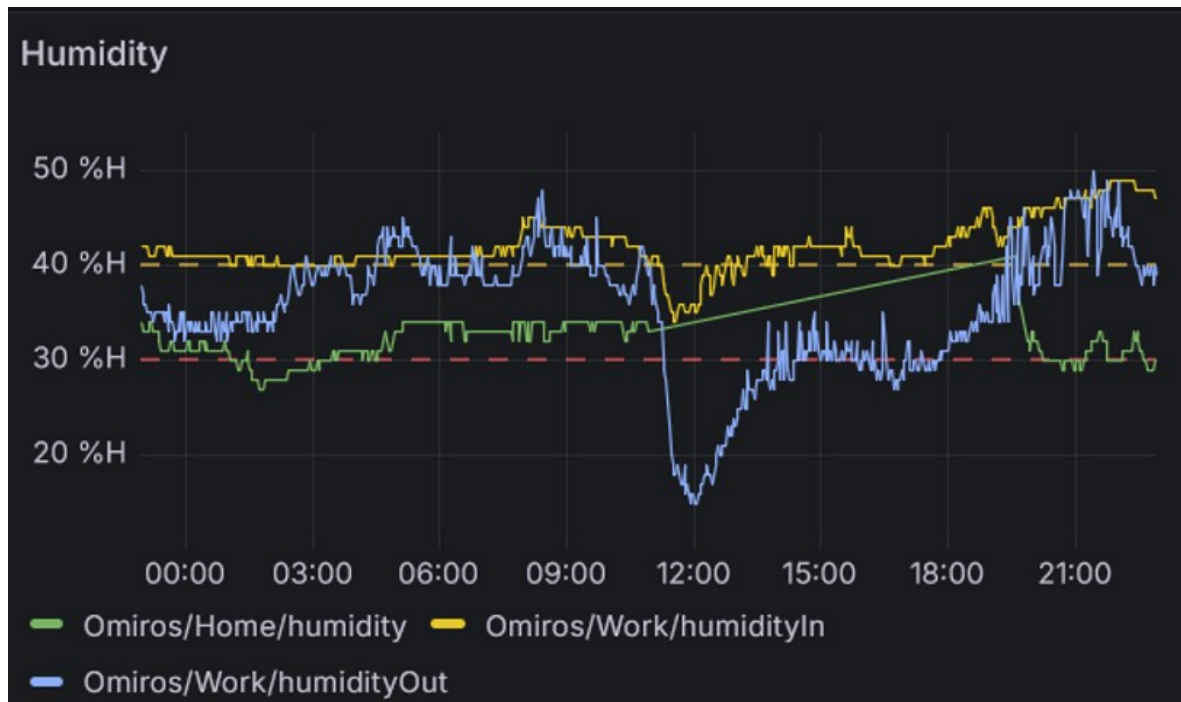
5.1 Σενάριο 1 : Σπίτι – Γραφείο – Εξωτερικός χώρος

Το πρώτο σενάριο αφορά τη σύγκριση του χώρου του σπιτιού, του εξωτερικού χώρου και το χώρο του γραφείου. Τα παρακάτω γραφήματα αφορούν το τελευταίο 24ωρο μιας μέρας τον Αύγουστο. Παρατηρώντας το tab σύγκρισης Compare(εικόνα 71) βλέπουμε ότι ο χώρος του σπιτιού είναι πιο ζεστός όλη τη μέρα σε σχέση με το χώρο εργασίας. Είναι σταθερα πάνω από του 30° ενώ στο γραφείο βλέπουμε τη θερμοκρασία κοντα στους 25° τις εργασιμες ωρες.



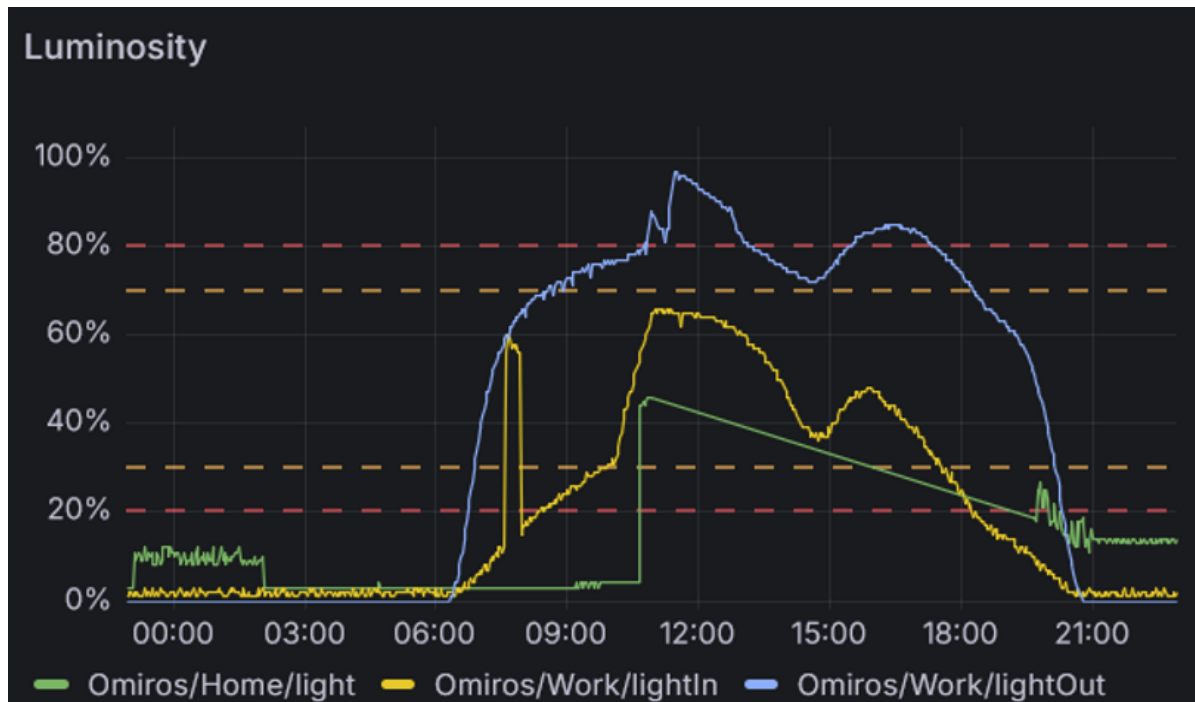
Εικόνα 71 Σύγκριση θερμοκρασίας

Για την υγρασία παρατηρούμε(εικόνα 72) ότι το γραφείο έχει περίπου 40% υγρασία στη διάρκεια της ημέρας ενώ το σπίτι έχει λιγότερη(περίπου 30%). Ο εξωτερικός χώρος ακολουθεί μια διακύμανση όπου η υγρασία μειώνεται τη διάρκεια της ημέρας και αυξάνεται τη νύχτα.



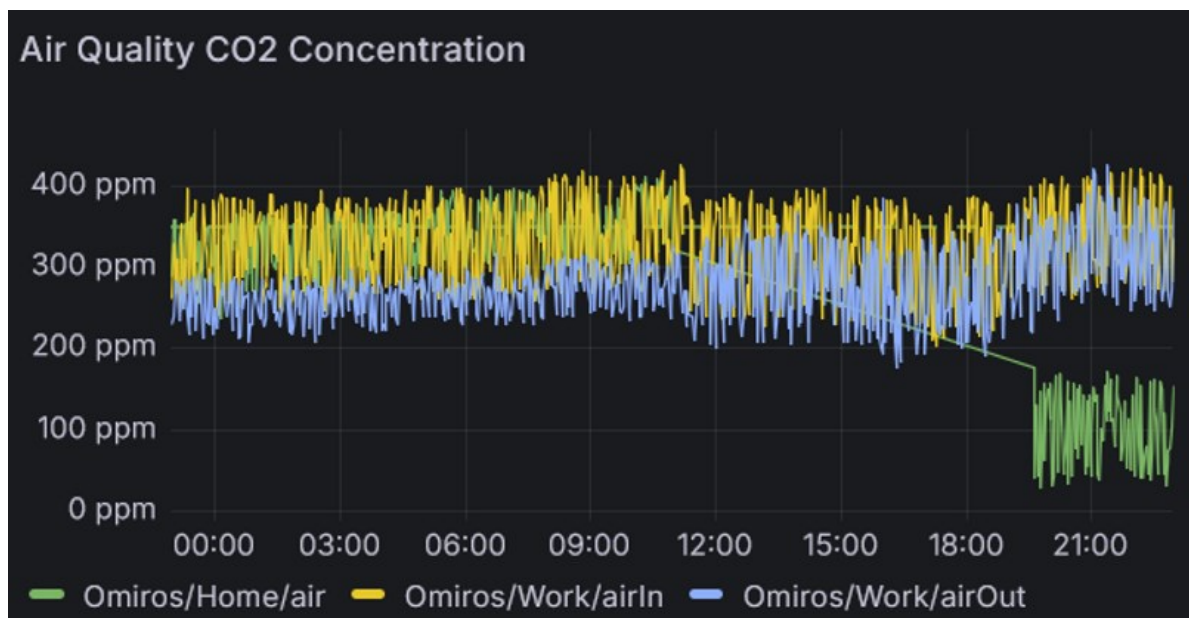
Εικόνα 72 Σύγκριση Υγρασίας

Για τη παράμετρο της φωτεινότητας(εικόνα 73) παρατηρούμε ότι το γραφείο δεν έχει φως το βράδυ και ακολουθεί τη φωτεινότητα του εξωτερικού χώρου. Με αυτή τη παράμετρο μπορούμε να ελέγξουμε το φωτισμό τις ώρες που δεν είναι εργάσιμες εξοικονομώντας ενέργεια. Τις εργάσιμες ώρες παρατηρούμε ότι το γραφείο έχει σωστά επίπεδα φωτισμού με μεγαλύτερη φωτεινότητα από το σπίτι. Ο Εξωτερικός χώρος έχει πολύ υψηλά επίπεδα φωτισμού και δεν ενδείκνυται για εργασία τις ώρες της ημέρας. Οι μοναδικές ώρες που ο εξωτερικός χώρος έχει σωστό επίπεδο φωτισμού έτσι ώστε να είναι ευχάριστος για εργασία είναι η ανατολή και το ηλιοβασίλεμα. Η παράμετρος της φωτεινότητας επίσης μπορεί να αναδείξει ότι κοιμόμαστε αργά το βράδυ στο σπίτι και αν αυτό συνεχίζεται να ορίσουμε κάποια αντίστοιχη ειδοποίηση καθώς δεν είναι καλό για την υγεία.



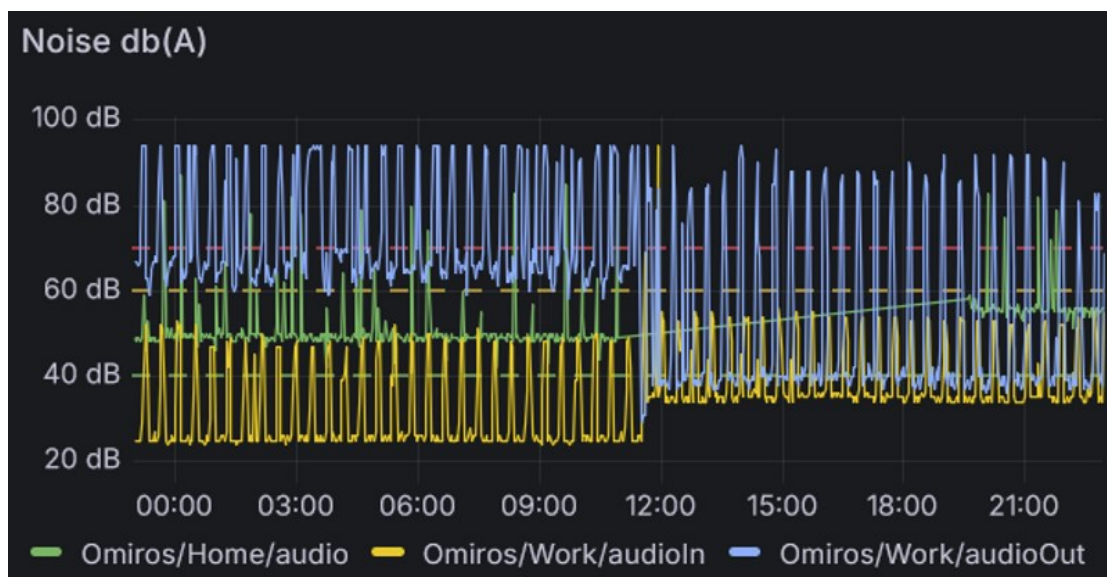
Εικόνα 73 Σύγκριση Φωτεινότητας

Για την ποιότητα του αέρα παρατηρούμε(εικόνα 74) τις υψηλότερες τιμές συγκέντρωσης CO₂ τις έχει το γραφείο περίπου 350ppm ενώ ο εξωτερικός χώρος όπως είναι αναμενόμενο έχει λιγότερη συγκέντρωση 250ppm. Ο χώρος του σπιτιού έχει και αυτός μικρότερη συγκέντρωση από το γραφείο αλλά μεγαλύτερη από τον εξωτερικό χώρο.



Εικόνα 74 Σύγκριση Ποιότητας του αέρα

Για το θόρυβο βλέπουμε(εικόνα75) ότι ο εξωτερικός χώρος έχει υψηλά επίπεδα θορύβου και μετά είναι ο χώρος του σπιτιού με αποδεκτά επίπεδα θορύβου. Καλύτερος χώρος για εργασία είναι το γραφείο με τα χαμηλότερο επίπεδο θορύβου.



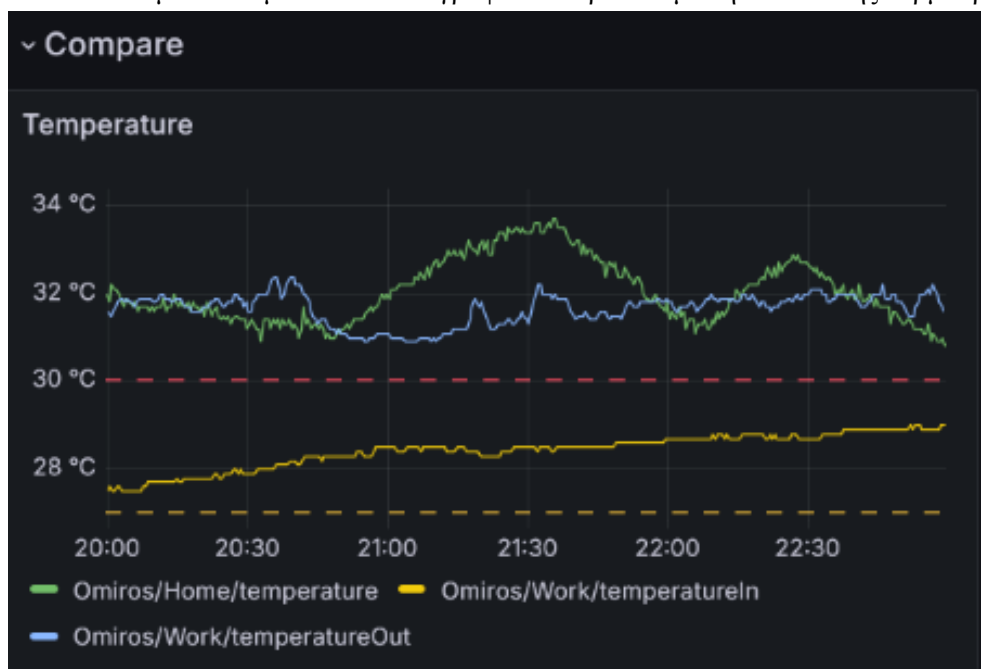
Εικόνα 75 Σύγκριση Θορύβου

Έτσι λοιπόν για το τελευταίο 24ωρο βλέπουμε(εικόνα 76) ότι το γραφείο είναι ο καλύτερος χώρος εργασίας, έπειτα ακολουθεί το σπίτι με βαθμολογία 2.6 στα 5 και τελευταίος έρχεται ο εξωτερικός χώρος.



Εικόνα 76 Σύγκριση δείκτη Overall Score

Το σύστημα που έχει υλοποιηθεί μας δίνει τη δυνατότητα να αναλύσουμε τα δεδομένα σε όποια χρονική διάρκεια θέλουμε. Για παράδειγμα μπορούμε να παρακολουθήσουμε τις τελευταίες 3 ώρες μετά από κάποια αλλαγή που κάναμε στο χώρο έτσι ώστε να δούμε το αποτέλεσμα στις συνθήκες του περιβάλλοντος. Στην εικόνα 77 βλέπουμε το γράφημα για τις τελευταίες 3 ώρες όπου κλείσαμε το κλιματιστικό στο γραφείο και βλέπουμε την άνοδο της θερμοκρασίας εκεί.



Εικόνα 77 Παρακολούθηση παραμέτρων σε συσχέτιση με κάποιο γεγονός

Συμπερασματικά θα λέγαμε ότι το περιβάλλον του γραφείου είναι καλύτερο από την τηλεργασία στο σπίτι όπως επίσης και από την εργασία στον εξωτερικό χώρο. Ακόμα βλέπουμε ότι ο χώρος του γραφείου μας προστατεύει από υψηλές θερμοκρασίες, υψηλή φωτεινότητα, υψηλό επίπεδο θορύβου και ανεπιθύμητα επίπεδα υγρασίας. Για τη ποιότητα του αέρα όμως χρειάζεται να αερίζουμε το χώρο καθώς παρατηρείται μεγαλύτερη συγκέντρωση CO₂ σε σχέση με τον εξωτερικό χώρο.

5.2 Σενάριο 2 : Μεγάλο Γραφείο 2 ατόμων – Μικρό Γραφείο 5 ατόμων – Εξωτερικός χώρος

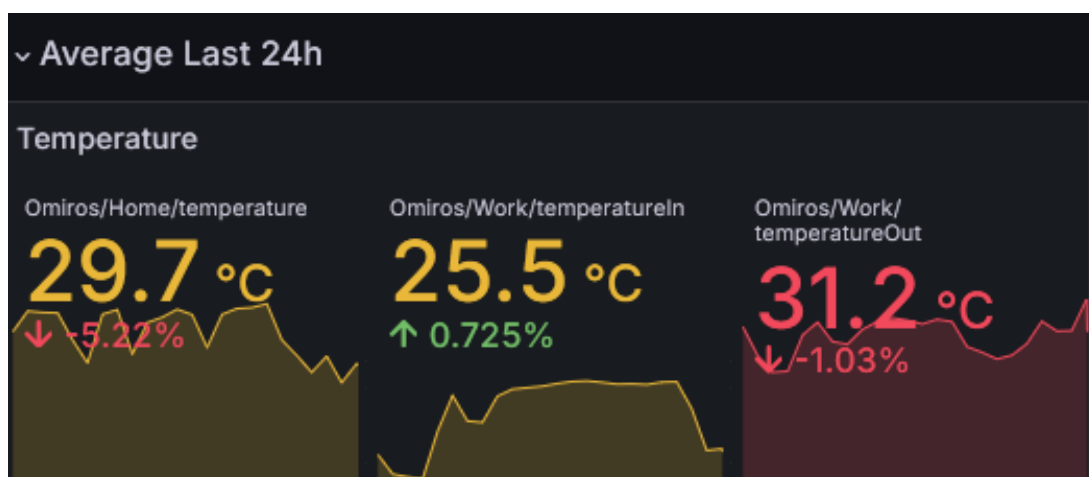
Το δεύτερο σενάριο αφορά τη σύγκριση του χώρου του μεγάλου γραφείου 2 ατόμων, του εξωτερικού χώρου και του χώρου του μικρότερου γραφείου των 5 ατόμων. Η μέτρηση αφορά το τελευταίο 24ωρο της 4ης Σεπτεμβρίου. Θα αναφέρουμε συνοπτικά κάποιες παρατηρήσεις. Παρατηρούμε(εικόνα 78) λοιπόν ότι στις εργάσιμες ώρες το γραφείο με τα 2 άτομα έχει χαμηλότερη θερμοκρασία σε σχέση με εκείνο με τα 5 άτομα. Για την υγρασία έχουμε διακύμανση για τις εργάσιμες ώρες και για τα δύο γραφεία και παρατηρείται χαμηλότερη υγρασία στον εξωτερικό χώρο. Η φωτεινότητα στο γραφείο με τα 5 άτομα είναι αρκετά υψηλή και πλησιάζει τον εξωτερικό χώρο. Αντίθετα η φωτεινότητα στο γραφείο με τα 2 άτομα είναι στα ιδανικά επίπεδα. Για την ποιότητα του αέρα βλέπουμε ότι ο εξωτερικός χώρος παρουσιάζει μεγαλύτερη συγκέντρωση CO₂ σε σχέση με το γραφείο 2 ατόμων ενώ το γραφείο με τα 5 άτομα έχει τη χαμηλότερη συγκέντρωση CO₂. Αυτό συμβαίνει γιατί στο γραφείο με τα 2 άτομα δεν αερίστηκε ο χώρος σε αντίθεση με το άλλο γραφείο όπου ένα παράθυρο ήταν πάντα

ανοιχτό. Επακόλουθο του ανοιχτού παραθύρου είναι να έχουμε περισσότερο θόρυβο στο γραφείο με τα 5 άτομα σε αντίθεση με τα 2 όπου ήταν κλειστό. Όπως αναμενόταν έχουμε τον μεγαλύτερο θόρυβο στον εξωτερικό χώρο. Συγκεντρωτικά βλέπουμε ότι με σκορ 3.2 το γραφείο με τα 2 άτομα είναι καλύτερο σε σχέση με το γραφείο με τα 5 άτομα όπου παίρνει σκορ 2.4. Ο εξωτερικός χώρος έρχεται τελευταίος με βαθμολογία 2 στα 5.



Εικόνα 78 Σενάριο 2

Ακόμα μπορούμε να παρατηρήσουμε συγκεκριμένα το τελευταίο 24ωρο από τα γραφήματα του tab Average Last 24h(εικόνα 79). Εκεί βλέπουμε ότι η θερμοκρασία στο γραφείο που είναι κλειστά τα παράθυρα είναι χαμηλότερη από το γραφείο που συνήθως έχει ανοιχτό παράθυρο και η τιμή της πλησιάζει τη τιμή του εξωτερικού χώρου. Το συγκεκριμένο tab είναι προγραμματισμένο να μας δίνει τιμές για τις 5 παραμέτρους ενδιαφέροντος μαζί με τη τελική βαθμολογία του χώρου ενδιαφέροντος.



Εικόνα 79 Tab Average Last 24h

Κλείνοντας θα λέγαμε ότι το περιβάλλον του γραφείου με τα δύο άτομα έχει καλύτερο περιβάλλον από αυτό με τα 5 άτομα και από τον εξωτερικό χώρο. Έχει καλύτερες τιμές στη θερμοκρασία, την υγρασία, τη φωτεινότητα και το θόρυβο. Χρειάζεται καλύτερο εξαερισμό όμως καθώς έχει τη μεγαλύτερη συγκέντρωση από το γραφείο με τα 5 άτομα που αερίζεται καλύτερα.

6 ΣΥΜΠΕΡΑΣΜΑΤΑ

Με το πέρας αυτού του ταξιδιού μπορούμε να επισημάνουμε κάποια ενδιαφέροντα συμπεράσματα που προέκυψαν. Η τεχνολογία του IoT απαιτεί τη σύνθεση πολλών τεχνολογιών όπως τα δίκτυα, η ασφάλεια, ο προγραμματισμός και τα ηλεκτρονικά. Στο πλαίσιο αυτό, η αρχιτεκτονική του IoT επιτρέπει τη δυνατότητα προσφοράς υπηρεσιών μέσω διάφορων πλατφορμών και εφαρμογών, ενσωματώνοντας τεχνολογίες από απλές low code λύσεις έως πιο σύνθετο low level προγραμματισμό, προσφέροντας έτσι μεγάλη ευελιξία και προσαρμοστικότητα.

Η δικτυακή λειτουργία των συστημάτων IoT μπορεί να υλοποιηθεί μέσω διάφορων τεχνικών όπως tunneling στο επίπεδο της εφαρμογής ή μέσω port forwarding και IPv6 στο επίπεδο του δικτύου, επιτρέποντας την ασφάλη και αποτελεσματική διακίνηση δεδομένων. Η τοπική εγκατάσταση και η χρήση του IPv6 που εφαρμόστηκε στο σύστημα IoT της εργασίας ήταν μια προσέγγιση προσανατολισμένη στις ανάγκες συστημάτων της σύγχρονης βιομηχανίας.

Στο πεδίο των ενσωματωμένων συστημάτων, οι μικροελεγκτές και οι αισθητήρες παίζουν κρίσιμο ρόλο. Ο σχεδιασμός τους απαιτεί λεπτομερή γνώση των διαθέσιμων βιβλιοθηκών και των πλατφορμών, ώστε να διασφαλίζεται η διαλειτουργικότητα μεταξύ διαφορετικών συστημάτων και συσκευών.

Αναφορικά με τις λύσεις cloud, αυτές παρέχουν τη δυνατότητα εύκολης υλοποίησης και διαχείρισης, ωστόσο συνοδεύονται από ορισμένους περιορισμούς στις δωρεάν εκδόσεις. Αντιθέτως, οι τοπικές λύσεις προσφέρουν μεγαλύτερη αυτονομία και ελευθερία στη χρήση, αλλά απαιτούν επενδύσεις σε ειδικό hardware. Τα πρωτόκολλα IoT όπως το MQTT δεν απαιτούν ιδιαίτερη υπολογιστική ισχύ και έτσι έχουν χαμηλές απαιτήσεις στο hardware.

Σημαντικός είναι επίσης ο έλεγχος και η διαχείριση της κατανάλωσης ενέργειας, καθώς και η ανάπτυξη αυτοματοποιημένων διαδικασιών για τον έλεγχο και την επαναφορά των λειτουργιών των συσκευών και των αισθητήρων, εξασφαλίζοντας τη συνεχή και αξιόπιστη λειτουργία τους. Στο επίπεδο του Server / Broker είναι αναγκαίο να υπάρχει αυτόματος έλεγχος των λειτουργιών του server καθώς και της υγείας του, ιδιαίτερα στη περίπτωση της τοπικής εγκατάστασης. Οι τεχνολογία IoT μας παρέχει χρήσιμες και εύκολα υλοποιήσιμες λύσεις ελέγχου των πόρων του hardware που χρησιμοποιούμε.

Τέλος, η ασφάλεια αποτελεί έναν κρίσιμο τομέα και υπάρχει η ανάγκη για ακριβή σχεδιασμό που να ισορροπεί μεταξύ προστασίας και της αποδοτικής λειτουργικότητας. Στον γενικότερο σχεδιασμό του συστήματος είναι επίσης σημαντικό να αφιερωθεί χρόνος στην επιλογή των κατάλληλων και σύγχρονων τεχνολογιών με το επαρκές documentation. Η σύνθεση όλων αυτών των τεχνολογιών IoT δε παρουσιάζει ιδιαίτερη πολυπλοκότητα και η ενασχόληση με αυτό το τομέα είναι μια καλή ευκαιρία για να καλλιεργήσουμε τις δεξιότητές μας σε σύγχρονα ζητήματα τεχνολογίας.

7 ΜΕΛΛΟΝΤΙΚΗ ΕΡΕΥΝΑ

Η διπλωματική εργασία ανέδειξε ενδιαφέροντα ζητήματα για μελλοντική έρευνα.

- Ένα από αυτά είναι η μοντελοποίηση και η προτυποποίηση ενός συστήματος βαθμολόγησης εσωτερικών χώρων γραφείου. Στη παραπάνω έρευνα θα είχε ενδιαφέρον η δημιουργία ενός αυτοματοποιημένου συστήματος με sensors & actuators που θα παρείχε υψηλής ποιότητας περιβάλλον εργασίας σε ένα κτήριο. Οι τεχνολογίες IoT μπορούν να βοηθήσουν στη έρευνα για τη σχεδίαση σύγχρονων κτηρίων τα οποία θα είναι ασφαλή, φιλικά προς το περιβάλλον με χαμηλή κατανάλωση ενέργειας και ευχάριστο για εργασία.
- Ακόμα θα είχε ενδιαφέρον η υλοποίηση ενός παρόμοιου συστήματος με αυτό της διπλωματικής εργασίας προσανατολισμένο σε βιομηχανικούς χώρους, χώρους γυμναστικής, σχολείων και νοσοκομείων όπου η κάθε περίπτωση έχει διαφορετικές ανάγκες και ιδιαιτερότητες.
- Η χρήση τεχνητής νοημοσύνης σε συστήματα IoT θα βοηθούσε στην ανάλυση των δεδομένων έτσι ώστε να χρησιμοποιηθούν λειτουργίες πρόβλεψης.
- Με την εξοικείωση με όλες τις τεχνολογίες IoT που χρησιμοποιήθηκαν σε αυτή την εργασία θα είχε ενδιαφέρον η περαιτέρω έρευνα στο τομέα της αγροτικής τεχνολογίας IoT (Agricultural IoT). Η βελτιστοποίηση της υδροληψίας, η εφαρμογή αισθητήρων για έλεγχο ασθενειών, τα αυτοματοποιημένα ρομποτικά συστήματα και η χρήση δορυφορικών δεδομένων για περιβαλλοντική πρόβλεψη είναι μερικές από τις προκλήσεις αυτού του τομέα.
- Σχεδιασμός και υλοποίηση συστήματος κινητών σταθμών μέτρησης ποιότητας του εξωτερικού περιβάλλοντος μιας πόλης. Για παράδειγμα να τοποθετούνται σε επαγγελματικά οχήματα που κινούνται στη πόλη(υπηρεσίες delivery ή μεταφορές), κινητοί σταθμοί(μικροελεγκτές και αισθητήρες) έτσι ώστε να έχουμε μέτρηση της ποιότητας του εξωτερικού περιβάλλοντος σε διαφορετικά σημεία της πόλης. Η οπτικοποίηση αυτής της πληροφορίας μπορεί να γίνει μέσα από έναν δυναμικό χάρτη.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7), 1645–1660. <https://doi.org/10.1016/j.future.2013.01.010>
2. Hammi, B., Khatoun, R., Zeadally, S., Fayad, A., & Khoukhi, L. (2017). Internet of Things (IoT) Technologies for Smart Cities. *IET Networks*, 7. <https://doi.org/10.1049/iet-net.2017.0163>
3. Liu, Z., Li, Z., Zhang, Y., Mutukumira, A. N., Feng, Y., Cui, Y., Wang, S., Wang, J., & Wang, S. (2024). Comparing Business, Innovation, and Platform Ecosystems: A Systematic Review of the Literature. *Biomimetics*, 9, 216. <https://doi.org/10.3390/biomimetics9040216>
4. Evolution of Internet of Things (IoT)—A Brief History. (2022, September 26). Bytebeam. <https://bytebeam.io/blog/a-brief-history-of-internet-of-things/>
5. Xu, L. D., He, W., & Li, S. (2014). Internet of Things in Industries: A Survey. *IEEE Transactions on Industrial Informatics*, 10(4), 2233–2243. *IEEE Transactions on Industrial Informatics*. <https://doi.org/10.1109/TII.2014.2300753>
6. Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Communications Surveys & Tutorials*, 17(4), 2347–2376. *IEEE Communications Surveys & Tutorials*. <https://doi.org/10.1109/COMST.2015.2444095>
7. Botta, A., de Donato, W., Persico, V., & Pescapé, A. (2016). Integration of Cloud computing and Internet of Things: A survey. *Future Generation Computer Systems*, 56, 684–700. <https://doi.org/10.1016/j.future.2015.09.021>
8. Define IoT - IEEE Internet of Things. (n.d.). Retrieved September 2, 2024, from <https://iot.ieee.org/definition.html>
9. hiotron. (2019, July 5). IoT Architecture Layers. HIOTRON. <https://www.hiotron.com/iot-architecture-layers/>
10. Periklis Chatzimisios Internet of Things (& 5G) Applications and technologies. International Hellenic University.
11. Ketshabetswe, L. K., Zungeru, A. M., Mangwala, M., Chuma, J. M., & Sigweni, B. (2019). Communication protocols for wireless sensor networks: A survey and comparison. *Heliyon*, 5(5), e01591. <https://doi.org/10.1016/j.heliyon.2019.e01591>
12. What is MQTT? (n.d.). Twilio. Retrieved September 2, 2024, from <https://www.twilio.com/en-us/blog/what-is-mqtt>
13. CoAP - Constrained Application Protocol. (n.d.). Radiocrafts. Retrieved September 2, 2024, from <https://radiocrafts.com/technologies/coap-constrained-application-protocol/>
14. Guinand, D., Trifa, V., & Wilde, E. (2010). A resource oriented architecture for the Web of Things. 2010 Internet of Things (IOT), 1–8. <https://doi.org/10.1109/IOT.2010.5678452>
15. <https://kr-labs.com.ua/books/Oreilly-XMPP-The-Definitive-Guide-May-2009.pdf>. (n.d.). Retrieved September 2, 2024, from <https://kr-labs.com.ua/books/Oreilly-XMPP-The-Definitive-Guide-May-2009.pdf>
16. Z-Wave vs. Zigbee: What’s the Difference. (n.d.). Retrieved September 2, 2024, from <https://reolink.com/blog/z-wave-vs-zigbee/>

17. Zigbee | Complete IOT Solution. (n.d.). CSA-IOT. Retrieved September 2, 2024, from <https://csa-iot.org/all-solutions/zigbee/>
18. Hierarchy of the Z-Wave system. (n.d.). Retrieved September 2, 2024, from https://smarterhome.sk/en/blog/hierarchy-of-the-z-wave-system_39.html
19. Hosni, S. H. A. (2017). Bluetooth Low Energy: A Survey. *International Journal of Computer Applications*, 162(1), 27–32. <https://doi.org/10.5120/ijca2017913248>
20. AumRaj. (2022, November 16). New IoT Platform for Wireless Device- Bluetooth Low Energy. Aumraj. <https://aumraj.com/new-iot-platform-for-wireless-device-bluetooth-low-energy/>
21. Saban, M., Aghzout, O., Medus, L. D., & Rosado, A. (2021). Experimental Analysis of IoT Networks Based on LoRa/LoRaWAN under Indoor and Outdoor Environments: Performance and Limitations. *IFAC-PapersOnLine*, 54(4), 159–164. <https://doi.org/10.1016/j.ifacol.2021.10.027>
22. Codeluppi, G., Cilfone, A., Davoli, L., & Ferrari, G. (2020). LoRaFarM: A LoRaWAN-Based Smart Farming Modular IoT Architecture. *Sensors*, 20(7), Article 7. <https://doi.org/10.3390/s20072028>
23. Adame, T., Bel, A., Bellalta, B., Barcelo, J., & Oliver, M. (2014). IEEE 802.11AH: The WiFi approach for M2M communications. *IEEE Wireless Communications*, 21(6), 144–152. IEEE Wireless Communications. <https://doi.org/10.1109/MWC.2014.7000982>
24. What is 5G Technology and What Does 5G Mean for IoT? (n.d.). Telenor IoT. Retrieved September 2, 2024, from <https://iot.telenor.com/technologies/connectivity/5g/>
25. Chakraborty, S. (2021, December 9). IoT Development Platforms Empowered by 5G Network—Open Cloudware. <https://opencloudware.com/iot-development-platforms-empowered-by-5g-network/>, <https://opencloudware.com/iot-development-platforms-empowered-by-5g-network/>
26. Papathanasiou, S. (2021, March 8). Let’s talk: IoT Wireless Network Protocols for Air Quality Monitors. See The Air. <https://seetheair.org/2021/03/08/lets-talk-iot-wireless-network-protocols-for-air-quality-monitors/>
27. Valerio, P. (2018, April 11). Top Wireless Standards For IoT Devices. *IoT Times*. <https://iot.eetimes.com/top-wireless-standards-for-iot-devices/>
28. The 3G4G Blog. (n.d.). Retrieved September 2, 2024, from <https://blog.3g4g.co.uk/search?updated-max=2017-09-01T11:15:00-07:00&max-results=20&start=303&by-date=false&m=0>
29. Zhou, M., Prodromou, A., Wang, R., Yang, H., Qian, D., & Tullsen, D. (2020). Temperature-Aware DRAM Cache Management—Relaxing Thermal Constraints in 3-D Systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 39(10), 1973–1986. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. <https://doi.org/10.1109/TCAD.2019.2927528>
30. Gautam, S. (2021, February 9). IoT in Smart Cities: How IoT is Raising the Standard of City Living. *Get My Parking*. <https://blog.getmyparking.com/2021/02/09/iot-in-smart-cities-how-iot-is-raising-the-standard-of-city-living/>
31. Vardakis, G., Hatzivasilis, G., Koutsaki, E., & Papadakis, N. (2024). Review of Smart-Home Security Using the Internet of Things. *Electronics*, 13(16), Article 16. <https://doi.org/10.3390/electronics13163343>

32. IoT | Smart Home and Smart City—Javatpoint. (n.d.). Wwww.Javatpoint.Com. Retrieved September 2, 2024, from <https://www.javatpoint.com/iot-smart-home-and-smart-city>
33. What is the Internet of Things (IoT)? | IBM. (2023, May 12). <https://www.ibm.com/topics/internet-of-things>
34. Choudhary, K., Gaba, G. S., Butun, I., & Kumar, P. (2020). MAKE-IT—A Lightweight Mutual Authentication and Key Exchange Protocol for Industrial Internet of Things. *Sensors*, 20(18), Article 18. <https://doi.org/10.3390/s20185166>
35. Forgan, B. (n.d.). Council Post: How IoT Is Transforming Healthcare. *Forbes*. Retrieved September 3, 2024, from <https://www.forbes.com/councils/forbestechcouncil/2021/03/31/how-iot-is-transforming-healthcare/>
36. IoT in Healthcare: Applications, benefits & Challenges. (n.d.). Peerbits. Retrieved September 3, 2024, from <https://www.peerbits.com/blog/internet-of-things-healthcare-applications-benefits-and-challenges.html>
37. IO, V. (2023, March 15). The Basics of Smart Farming: How Is IoT Used in Agriculture? VITALITY IO, Inc. <https://vitality.io/how-is-iot-used-in-agriculture/>
38. Brink, H. W., Loomans, M. G. L. C., Mobach, M. P., & Kort, H. S. M. (2022). A systematic approach to quantify the influence of indoor environmental parameters on students' perceptions, responses, and short-term academic performance. *Indoor Air*, 32(10), e13116. <https://doi.org/10.1111/ina.13116>
39. ISO standards in the area of the Ergonomics of the Physical Environment—OSHWiki | European Agency for Safety and Health at Work. (n.d.). Retrieved September 3, 2024, from <https://oshwiki.osha.europa.eu/en/themes/iso-standards-area-ergonomics-physical-environment>
40. Government of Canada, C. C. for O. H. and S. (2024, July 26). CCOHS: Thermal Comfort for Office Work. https://www.ccohs.ca/oshanswers/phys_agents/thermal_comfort.html
41. Buildings | Free Full-Text | Indoor Environmental Quality and Comfort in Offices: A Review. (n.d.). Retrieved September 3, 2024, from <https://www.mdpi.com/2075-5309/13/10/2490>
42. Reiteration of Existing OSHA Policy on Indoor Air Quality: Office Temperature/Humidity and Environmental Tobacco Smoke | Occupational Safety and Health Administration. (n.d.). Retrieved September 3, 2024, from <https://www.osha.gov/laws-regs/standardinterpretations/2003-02-24>
43. Table 1 Outdoor air quality (OAQ) parameters. (n.d.). ResearchGate. Retrieved September 3, 2024, from https://www.researchgate.net/figure/Outdoor-air-quality-OAQ-parameters_tbl1_326306113
44. Alonso-Blanco, E., Gómez-Moreno, F. J., Díaz-Ramiro, E., Fernández, J., Coz, E., Yagüe, C., Román-Cascón, C., Narros, A., Borge, R., & Artíñano, B. (2023). Real-Time Measurements of Indoor–Outdoor Exchange of Gaseous and Particulate Atmospheric Pollutants in an Urban Area. *International Journal of Environmental Research and Public Health*, 20(19), 6823. <https://doi.org/10.3390/ijerph20196823>
45. project, T. W. A. Q. I. (n.d.). Agia Sofia, Greece Air Pollution: Real-time Air Quality Index. Aqicn.Org. Retrieved September 3, 2024, from <https://aqicn.org/city/greece/agia-sofia/>
46. Santos, P. G. (2024, March 4). How to monitor indoor air quality | Nanoenvi IAQ. Envira. <https://envira.global/nanoenvi-iaq/>

47. Pipal, A., & Taneja, A. (2023). Measurements of Indoor Air Quality: Science and Applications (pp. 1–35). https://doi.org/10.1007/978-981-19-1550-5_90-1
48. Indoor air quality guidelines from across the world: An appraisal considering energy saving, health, productivity, and comfort—ScienceDirect. (n.d.). Retrieved September 3, 2024, from <https://www.sciencedirect.com/science/article/pii/S0160412023004002>
49. Lazovic, I., Stevanovic, Z., Jovasevic-Stojanovic, M., Zivkovic, M., & Banjac, M. (2016). Impact of CO₂ concentration on indoor air quality and correlation with relative humidity and indoor air temperature in school buildings in Serbia. *Thermal Science*, 20(suppl. 1), 297–307. <https://doi.org/10.2298/TSCI150831173L>
50. Palacios, J., Steele, K., Tan, Z., & Zheng, S. (2021). Human health and productivity outcomes of office workers associated with indoor air quality: A systematic review (SSRN Scholarly Paper 3881998). <https://doi.org/10.2139/ssrn.3881998>
51. Sey, M. S. L., Schmaltz, L., Al-Haddad, M. A., DeWitt, J. M., Calley, C. S. J., Juan, M., Lasisi, F., Sherman, S., McHenry, L., Imperiale, T. F., & LeBlanc, J. K. (2015). Effectiveness and safety of serial endoscopic ultrasound–guided celiac plexus block for chronic pancreatitis. *Endoscopy International Open*, 3(1), E56–E59. <https://doi.org/10.1055/s-0034-1377919>
52. Boyce, P., Veitch, J., Newsham, G., Myer, M., & Hunter, C. (2003). Lighting Quality and Office Work: A Field Simulation Study. <https://doi.org/10.4224/20393390>
53. The IESNA lighting handbook by Mark Stanley Rea | Open Library. (n.d.). Retrieved September 3, 2024, from https://openlibrary.org/books/OL58383M/The_IESNA_lighting_handbook
54. Occupational Noise Exposure—Overview | Occupational Safety and Health Administration. (n.d.). Retrieved September 3, 2024, from <https://www.osha.gov/noise>
55. Berglund, B., Lindvall, T., Schwela, D. H., & Team, W. H. O. O. and E. H. (1999). Guidelines for community noise. <https://iris.who.int/handle/10665/66217>