



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΑΙΓΑΙΟΥ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΑΚΩΝ
ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

*Βιομηχανική παρακολούθηση συστημάτων
ψύξης, θέρμανσης και κλιματισμού (HVAC), με
τη χρήση του Διαδικτύου των Πραγμάτων.*

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΤΟΥ
Νικολάου Ρέκκα

Επιβλέπων:

Μαλιάτσος Κωνσταντίνος

Μέλη εξεταστικής επιτροπής:

Σκούτας Δημήτριος

Εμμανουήλ Καλλίγερος

Σάμος, Ιούλιος 2023

Η σελίδα αυτή είναι σκόπιμα λευκή.

Περίληψη

Στις μέρες μας οι περισσότεροι άνθρωποι σκέφτονται το Διαδίκτυο των Πραγμάτων (IoT), ως έναν τρόπο σύνδεσης συσκευών στα σπίτια ή τα γραφεία τους με το διαδίκτυο, ώστε να μπορούν να ελέγχονται από απόσταση. Ωστόσο, οι πιθανές εφαρμογές του IoT εκτείνονται πολύ πέρα από ένα έξυπνο σπίτι (Smart Home), με κάποιες από τις πιθανές εφαρμογές να είναι εμπορικά και βιομηχανικά περιβάλλοντα (IIoT - Industrial IoT). Συγκεκριμένα, το IoT έχει τη δυνατότητα να φέρει επανάσταση στον τρόπο με τον οποίο παρακολουθούνται και ελέγχονται τα συστήματα Heating, ventilation, and air conditioning (HVAC).

Τα συστήματα HVAC σε συνδυασμό με το Διαδίκτυο των Πραγμάτων, μπορούν να προσφέρουν μια σειρά από οφέλη σε σχέση με τα παραδοσιακά συστήματα. Για παράδειγμα, μπορούν να επιτρέπουν την παρακολούθηση της χρήσης ενέργειας σε πραγματικό χρόνο, πράγμα που σημαίνει ότι η ενέργεια μπορεί να χρησιμοποιηθεί πιο αποτελεσματικά. Επιπλέον, μπορούν να παρέχουν ειδοποιήσεις όταν τα συστήματα αυτά (HVAC) χρειάζονται συντήρηση (predictive maintenance), πράγμα που σημαίνει ότι μπορούν να διατηρούνται σε βέλτιστη κατάσταση πιο εύκολα και με σημαντικά λιγότερο κόστος. Τέλος, μπορούν να χρησιμοποιηθούν για τη δημιουργία πιο άνετων εργασιακών περιβαλλόντων, ρυθμίζοντας αυτόματα τα επίπεδα θερμοκρασίας και υγρασίας με βάση τις ανάγκες των χρηστών.

Η εφαρμογή συστημάτων HVAC με δυνατότητες που προσφέρει το IoT, βρίσκεται ακόμα σε αρχικά στάδια. Ωστόσο, υπάρχει ήδη μια σειρά από εταιρείες που προσφέρουν προϊόντα και υπηρεσίες σε αυτόν τον τομέα, όπως επίσης και μεγάλο ερευνητικό ενδιαφέρον. Στο μέλλον, είναι πιθανό τα συστήματα HVAC με την χρήση του IoT να γίνουν πιο ευρέως αναγνωρισμένα, καθώς τα οφέλη που προσφέρουν είναι πολλαπλά.

Στην παρούσα διπλωματική εργασία έγινε ανάπτυξη δύο πρωτότυπων πλακετών, χρησιμοποιώντας του μικροελεγκτές (MCU) ESP8266 και ESP32 για την συλλογή δεδομένων που αφορούν την εσωτερική θερμοκρασία του θαλάμου ψύξης, την κατάσταση της πόρτας του θαλάμου (OPEN/CLOSE state), καθώς και την ένταση, την πίεση και την θερμοκρασία του συμπιεστή (Compressor Unit). Επιπλέον αναπτύχθηκε μια διαδικτυακή διεπαφή για την οπτικοποίηση των δεδομένων αυτών σε πραγματικό χρόνο, όπως επίσης και διαγράμματα (Charts) για κάθε τύπο δεδομένων που προαναφέρθηκε.

© 2023

του

Νικολάου Ρέκκα

Τμήμα Μηχανικών Πληροφοριακών και Επικοινωνιακών Συστημάτων
ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ

Η σελίδα αυτή είναι σκόπιμα λευκή.

Abstract

These days most people think of the Internet of Things (IoT) as a way to connect devices in their homes or offices to the internet so they can be controlled remotely. However, the potential applications of IoT extend far beyond a Smart Home, with some of the potential applications being commercial and industrial environments (IIoT - Industrial IoT). In particular, the IoT has the potential to revolutionize the way Heating, Ventilation, and Air Conditioning (HVAC) systems are monitored and controlled.

HVAC systems combined with the Internet of Things can offer a number of benefits over traditional systems. For example, they can enable real-time monitoring of energy use, meaning that energy can be used more efficiently. In addition, they can provide alerts when these systems (HVAC) need maintenance (predictive maintenance), which means they can be kept in optimal condition more easily and at significantly less cost. Finally, they can be used to create more comfortable work environments by automatically adjusting temperature and humidity levels based on user needs.

The implementation of IoT enabled HVAC systems is still in its early stages. However, there are already a number of companies offering products and services in this area, as well as considerable research interest. In the future, it is likely that HVAC systems using IoT will become more widely recognized, as the benefits they offer are manifold.

In this thesis, two prototype boards were developed, using ESP8266 and ESP32 microcontrollers (MCU) to collect data regarding the internal temperature of the cooling chamber, the state of the chamber door (OPEN/CLOSE state), as well as the current, pressure and temperature of the compressor (Compressor Unit). In addition, an online interface was developed for the visualization of these data in real time, as well as charts for each type of data mentioned above.

Η σελίδα αυτή είναι σκόπιμα λευκή.

Υπεύθυνη Δήλωση πρωτοτυπίας διπλωματικής εργασίας

Είμαι συγγραφέας αυτής της διπλωματικής εργασίας και δηλώνω ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω αναφέρει τις πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επίσης βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά ειδικά για τη συγκεκριμένη διπλωματική εργασία.

Υπογραφή

Νικόλαος Ρέκκας.

Ευχαριστίες

Η επαφή μου με το Διαδίκτυο των Πραγμάτων και τα Δίκτυα Αισθητήρων ξεκίνησε μέσα από το προπτυχιακό πρόγραμμα σπουδών. Αυτά τα μαθήματα μου κέντρισαν το ενδιαφέρον και μου έδωσαν όλες τις απαραίτητες βάσεις για να συνεχίσω. Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου Κωσταντίνο Μαλιάτσο, ο οποίος με εμπιστεύτηκε και μου έδωσε την ευκαιρία να αναλάβω ένα θέμα το οποίο πραγματικά με ενδιέφερε. Τέλος θα ήθελα να ευχαριστήσω τους γονείς μου και τους φίλους μου οι οποίοι με βοήθησαν και στάθηκαν δίπλα μου όλα αυτά τα χρόνια κατά την διάρκεια των σπουδών μου.

Πίνακας περιεχομένων

1. Εισαγωγή	6
1.1 Κίνητρο	6
1.2 Στόχοι Διπλωματικής	6
1.3 Δομή Διπλωματικής	6
2. Διαδίκτυο των Πραγμάτων - IoT	8
2.1 Εισαγωγή	8
2.2 Ορισμός	9
2.3 Ιστορική Αναδρομή	9
2.4 Αρχιτεκτονική	11
2.4 Χαρακτηριστικά	14
2.5 Απαιτήσεις	15
2.6 Περιπτώσεις Χρήσης	16
3. Industrial IoT - IIoT	19
3.1 Εισαγωγή	19
3.2 Ορισμός	20
3.3 Δομικά Στοιχεία του IIoT	21
3.4 Περιπτώσεις Χρήσης του IIoT	25
3.5 Εφαρμογές του IIoT	26
3.5.1 Περίπτωση Χρήσης: HVAC (Heating Ventilation & Air Conditioning)	27
4. Σχεδίαση Εφαρμογής	28
4.1 Εισαγωγή	28
4.2 IoT Development Boards	28
4.2.1 Microcontrollers	28
4.2.2 Single Board Computers	30
4.3 Τεχνολογίες Ασύρματων Επικοινωνιών	32
4.3.1 Μικρής Εμβέλειας	32
4.3.2 Μεγάλης Εμβέλειας (και χαμηλής ισχύος)	35
4.4 Μοτίβα Ενδιάμεσου Λογισμικού (Pub/Sub)	38
4.4.1 MQTT - Message Queuing Telemetry Transport	38
4.4.2 XMPP - eXtensible Messaging and Presence Protocol	40
4.4.3 AMQP - Advanced Message Queuing Protocol	40
4.5 Έννοιες Σχεδίασης Λογισμικού	41

4.5.1 API (Application Programming Interface)	41
4.5.2 Τύποι API	42
4.5.3 Πρωτόκολλα και Αρχιτεκτονική API	42
4.5.4 REST API	43
4.6 Βάσεις Δεδομένων	44
4.6.1 Σχεσιακές Βάσεις Δεδομένων	44
4.6.2 Μη-Σχεσιακές Βάσεις Δεδομένων (NoSQL)	45
4.6.3 Βάσεις Δεδομένων Χρονοσειρών (TSDB)	46
4.7 Ολοκληρωμένα Περιβάλλοντα Ανάπτυξης για IoT εφαρμογές	47
5. Υλοποίηση Εφαρμογής	49
5.1 Απαιτήσεις Συστήματος	49
5.2 Αρχιτεκτονική Συστήματος	50
5.3 Ανάπτυξη Πρωτοτύπων	52
5.3.1 Prototype board 1: Cooling Chamber	53
5.3.2 Prototype board 2: Compressor Unit	59
5.4 Cloud Υποδομή	66
5.4.1 Backend	67
5.4.2 Frontend	70
5.4.3 Middleware	71
5.4.4 MQTT Broker	72
6. Αποτελέσματα	74
7. Συμπεράσματα / Μελλοντική Εργασία	75
7.1 Συμπεράσματα	75
7.2 Μελλοντική Εργασία	75
Βιβλιογραφία	78

Λίστα Εικόνων

Εικόνα 1 Περιπτώσεις χρήσης του Internet of Things.	8
Εικόνα 2: Ιστορία του Internet of Things	10
Εικόνα 3: Αρχιτεκτονική 3 επιπέδων του IoT	11
Εικόνα 4: Αρχιτεκτονική 5 επιπέδων του IoT	13
Εικόνα 5: Περιπτώσεις χρήσης του IoT	16
Εικόνα 6: Διαφορετικές εφαρμογές του IIoT και IoT	19
Εικόνα 7: Επίπεδα υπολογιστικής λειτουργίας του IIoT	23
Εικόνα 8 : Ασύρματα Πρωτόκολλα επικοινωνίας χαμηλής εμβέλειας	33
Εικόνα 9: Ασύρματα Πρωτόκολλα επικοινωνίας υψηλής εμβέλειας	35
Εικόνα 10: Παράδειγμα της αρχιτεκτονικής του πρωτοκόλλου MQTT	39
Εικόνα 11: Μοντέλο επικοινωνίας REST API	43
Εικόνα 12: Αρχιτεκτονική προτεινόμενου συστήματος	50
Εικόνα 13 ESP8266 Development Board Pinout	53
Εικόνα 14: Αισθητήρας θερμοκρασίας DS18B2	54
Εικόνα 15: Μαγνητική επαφή.	55
Εικόνα 16: Λειτουργία μαγνητικής επαφής	55
Εικόνα 17: Prototype Board 1 (Cooling Chamber): Pictorial Diagram	56
Εικόνα 18: Prototype Board 1 (Cooling Chamber): Pictorial Diagram	57
Εικόνα 19: ESP32 Development Board Pinout	60
Εικόνα 20: Αισθητήρας θερμοκρασίας NTC 10K	61
Εικόνα 21: Αισθητήρας ρεύματος	62
Εικόνα 22: Αισθητήρας πίεσης	63
Εικόνα 23: Prototype Board 2 (Compressor Unit): Pictorial Diagram	64
Εικόνα 24: Prototype Board 2 (Compressor Unit): Schematic Diagram	65
Εικόνα 25: User Interface (UI)	67

Λίστα Πινάκων

Πίνακας 1: Χαρακτηριστικά από εμπορικά Development Boards	29
Πίνακας 2: Χαρακτηριστικά δημοφιλών/εμπορικών Single Board Computers	31
Πίνακας 3: Prototype Board 1 (Cooling Chamber): Firmware Setup	59
Πίνακας 4: Prototype Board 2 (Compressor Unit): Firmware Setup	66
Πίνακας 5: Μοντέλα Δεδομένων και αντίστοιχα πεδία.	68
Πίνακας 6: API Endpoints	69

Ακρωνύμια

IoT	Internet of Things
IIoT	Industrial Internet of Things
DCS	Distributed Control Systems
RFID	Radio Frequency Identification
OTA	Over the Air Update
OSI	Open Systems Interconnection
MCU	Microcontroller Unit
SBC	Single Board Computers
HAT	Hardware Attached on Top
BLE	Bluetooth Low Energy
NTC	Negative Temperature Coefficient
UI	User Interface
PCB	Printed Circuit Board
API	Application Programming Interface
MQTT	Message Queuing Telemetry Transport
XMPP	eXtensible Messaging and Presence Protocol
AMQP	Advanced Message Queuing Protocol
HVAC	Heating Ventilation and Air Conditioning
LTE	Long-Term Evolution
AWS	Amazon Web Services
LoRaWAN	Long Range Wide Area Network
NB-IoT	Narrowband Internet of things
SQL	Structured Query Language
IDE	Integrated Development Environment
TSDB	Timeseries Database
GPIO	General Purpose Input/Output
ADR	Adaptive Data Rate
TLS	Transport Layer Security
ISM	Industrial Scientific Medical

1. Εισαγωγή

1.1 Κίνητρο

Καθώς εξελίσσονται διάφορες τεχνολογίες αιχμής στο τομέα των Πληροφοριακών Συστημάτων - κυρίως τεχνολογίες όπως το Διαδίκτυο των Πραγμάτων (IoT), τα Μεγάλα Δεδομένα και η Τεχνητή Νοημοσύνη - είναι σημαντικό να βρεθούν καινοτόμοι τρόποι για την εφαρμογή και ενσωμάτωση αυτών των τεχνολογιών σε κρίσιμα για την βιομηχανία συστήματα, όπως τα συστήματα Θέρμανσης, Εξαερισμού και Κλιματισμού (HVAC - Heating, Ventilation and Air Conditioning). Τα βιομηχανικά HVAC συστήματα αποτελούν την ραχοκοκαλιά της σύγχρονης υποδομής των βιομηχανιών και η αποτελεσματική λειτουργία τους είναι ένας σημαντικός παράγοντας για την απρόσκοπτη λειτουργία αυτών των μηχανημάτων όσο και για την εξοικονόμηση ενέργειας αυτών.

1.2 Στόχοι Διπλωματικής

Η παρούσα διπλωματική εστιάζει στο πως μπορεί να γίνει ενσωμάτωση της τεχνολογίας του Διαδικτύου των Πραγμάτων στα HVAC συστήματα. Στόχος είναι η ανάπτυξη μιας ολοκληρωμένης λύσης για την παρακολούθηση σε πραγματικό χρόνο, καθώς και η συλλογή των δεδομένων αυτών για τα μηχανήματα HVAC. Η ανάπτυξη της εφαρμογής αυτής στην παρούσα διπλωματική εργασία αποτελείται από 1) δύο πρωτότυπες 'έξυπνες' συσκευές και 2) το πληροφοριακό σύστημα/cloud υποδομή όπου έχει υλοποιηθεί η Web εφαρμογή μέσω της οποίας ο χρήστης μπορεί να δει τις κρίσιμες για τα μηχανήματα αυτά μετρικά (metrics).

1.3 Δομή Διπλωματικής

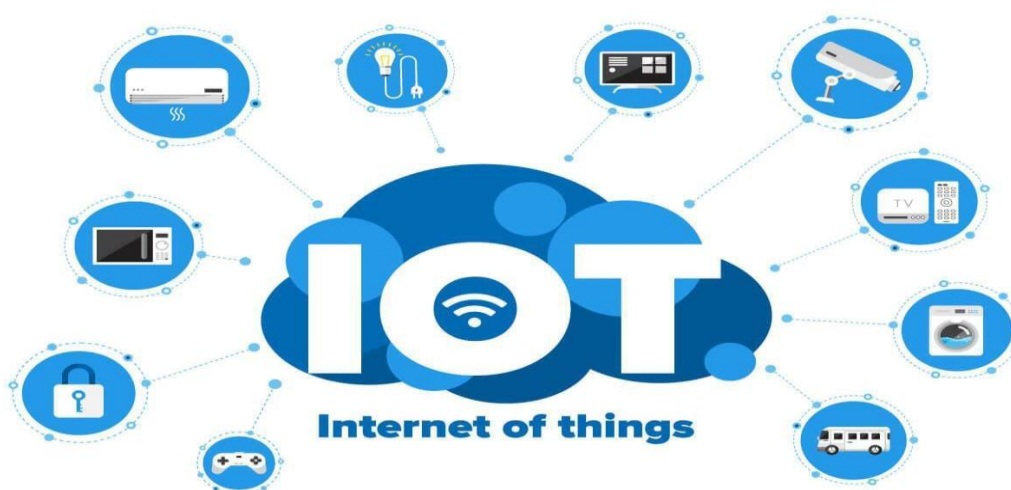
Η δομή των κεφαλαίων είναι η εξής: στο Κεφάλαιο 2, γίνεται μια εισαγωγή σε διάφορες έννοιες που αφορούν το Διαδίκτυο των Πραγμάτων, όπως ορισμοί, αρχιτεκτονικές, απαιτήσεις και διάφορες περιπτώσεις χρήσεις. Στο Κεφάλαιο 3, γίνεται αναφορά στο Βιομηχανικό Διαδίκτυο των Πραγμάτων με έμφαση τα δομικά στοιχεία του, όπως και σε περιπτώσεις χρήσεις του στην βιομηχανία, ενώ τελικά αναφέρονται τα HVAC μηχανήματα τα οποία τα οποία αποτελούν μια τέτοια περίπτωση χρήσης με έμφαση στη συγκεκριμένη που μελετάται στην παρούσα διπλωματική εργασία. Στο Κεφάλαιο 4, αναλύονται όλες οι τεχνολογίες οι οποίες έπρεπε να ληφθούν υπόψιν, πριν γίνει η υλοποίηση της εφαρμογής. Στο κεφάλαιο 5, παρουσιάζεται ενδελεχώς όλα τα βήματα που χρειάστηκαν - στα επιμέρους κομμάτια που απαρτίζουν το προτεινόμενο σύστημα - για να υλοποιηθεί τελικά η εφαρμογή. Στο Κεφάλαιο 6, αναφέρονται τα αποτελέσματα από την υλοποίηση της εφαρμογής.

Τέλος, στο Κεφάλαιο 7, πραγματοποιείται μια σύνοψη της εργασίας και γίνεται μια συνολική αναφορά σε διάφορα σημεία της εφαρμογής τα οποία θα χρειαζόνταν βελτίωση, σε μελλοντικό χρόνο, τόσο σε ερευνητικό επίπεδο όσο και σε επίπεδο ενός enterprise προϊόντος.

2. Διαδίκτυο των Πραγμάτων - IoT

2.1 Εισαγωγή

Το Διαδίκτυο των Πραγμάτων (IoT) είναι μια από τις νεότερες τεχνολογίες στο κλάδο των ΤΠΕ, η οποία φαίνεται να έχει προσελκύσει πολύ ενδιαφέρον τα τελευταία χρόνια, καθώς έχει την δυνατότητα να μετασχηματίσει διάφορους κλάδους. Το Διαδίκτυο των Πραγμάτων αναφέρεται στην διασύνδεση φυσικών αντικειμένων όπως: συσκευές, οχήματα, κτίρια και άλλα, τα οποία έχουν ενσωματωμένα πάνω τους αισθητήρες, λογισμικό και κάποια δικτυακή διεπαφή για την διασύνδεση τους στο δίκτυο. Αυτές οι 'έξυπνες' συσκευές έχουν του δυνατότητα να συλλέγουν και να ανταλλάσσουν δεδομένα μέσω του διαδικτύου, καθώς και μεταξύ τους (Machine-to-Machine - M2M), επιτρέποντας έτσι την αυτοματοποίηση των διαδικασιών, την βελτιωμένη λήψη αποφάσεων που σκοπό έχει την μέγιστη αποτελεσματικότητα. Σε αυτό το κεφάλαιο θα γίνει μια εκτενής επισκόπηση του Διαδικτύου των Πραγμάτων, συμπεριλαμβανομένου του ορισμού/ων, των χαρακτηριστικών, των απαιτήσεων αλλά και κάποιων περιπτώσεων χρήσης του. Έπειτα θα αναπτυχθούν κάποιες από τις υπάρχουσες τεχνολογίες επικοινωνίας, διάφοροι μικροελεγκτές, όπως επίσης και πρωτόκολλα επιπέδου εφαρμογής (application layer) σύμφωνα με το OSI Model.



Εικόνα 1 Περιπτώσεις χρήσης του Internet of Things.

Πηγή : <https://www.globalsign.com/en-sg/blog/what-internet-things-and-how-does-it-work>

2.2 Ορισμός

Οι ορισμοί για το Διαδίκτυο των Πραγμάτων ποικίλουν ανάλογα με την πηγή αλλά και το πλαίσιο. Παρακάτω παρατίθενται κάποιοι από τους ορισμούς σύμφωνα με κάποια survey και ερευνητικά άρθρα.

1. Η εργασία [\[1\]](#) “Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications” των Al-Fuqaha κ.α. ορίζει το IoT ως «ένα δίκτυο διασυνδεδεμένων αντικειμένων, το καθένα ενσωματωμένο με αισθητήρες, ενεργοποιητές και άλλες συσκευές που ανταλλάσσουν δεδομένα μέσω του Διαδικτύου».
2. Στην εργασία [\[2\]](#) «Internet of Things (IoT) Security: Current Status, Challenges and Prospective Measures», ο Rwan κ.α.. ορίζουν το IoT ως «μια συλλογή πολλών διασυνδεδεμένων αντικειμένων, υπηρεσιών, ανθρώπων και συσκευών που μπορούν να επικοινωνούν, να μοιράζονται δεδομένα και πληροφορίες για την επίτευξη ενός κοινού στόχου σε διαφορετικούς τομείς και εφαρμογές».
3. Στην εργασία [\[3\]](#) «The Internet of Things: A survey» των Atzori κ.α. ορίζουν το IoT ως «την διάχυτη παρουσία γύρω μας από μια ποικιλία πραγμάτων ή αντικειμένων, αισθητήρων, ενεργοποιητών, κινητών τηλεφώνων που μέσα από μοναδικές διευθύνσεις, είναι σε θέση να αλληλοεπιδρούν με τους γείτονες τους για την επίτευξη κοινών στόχων».

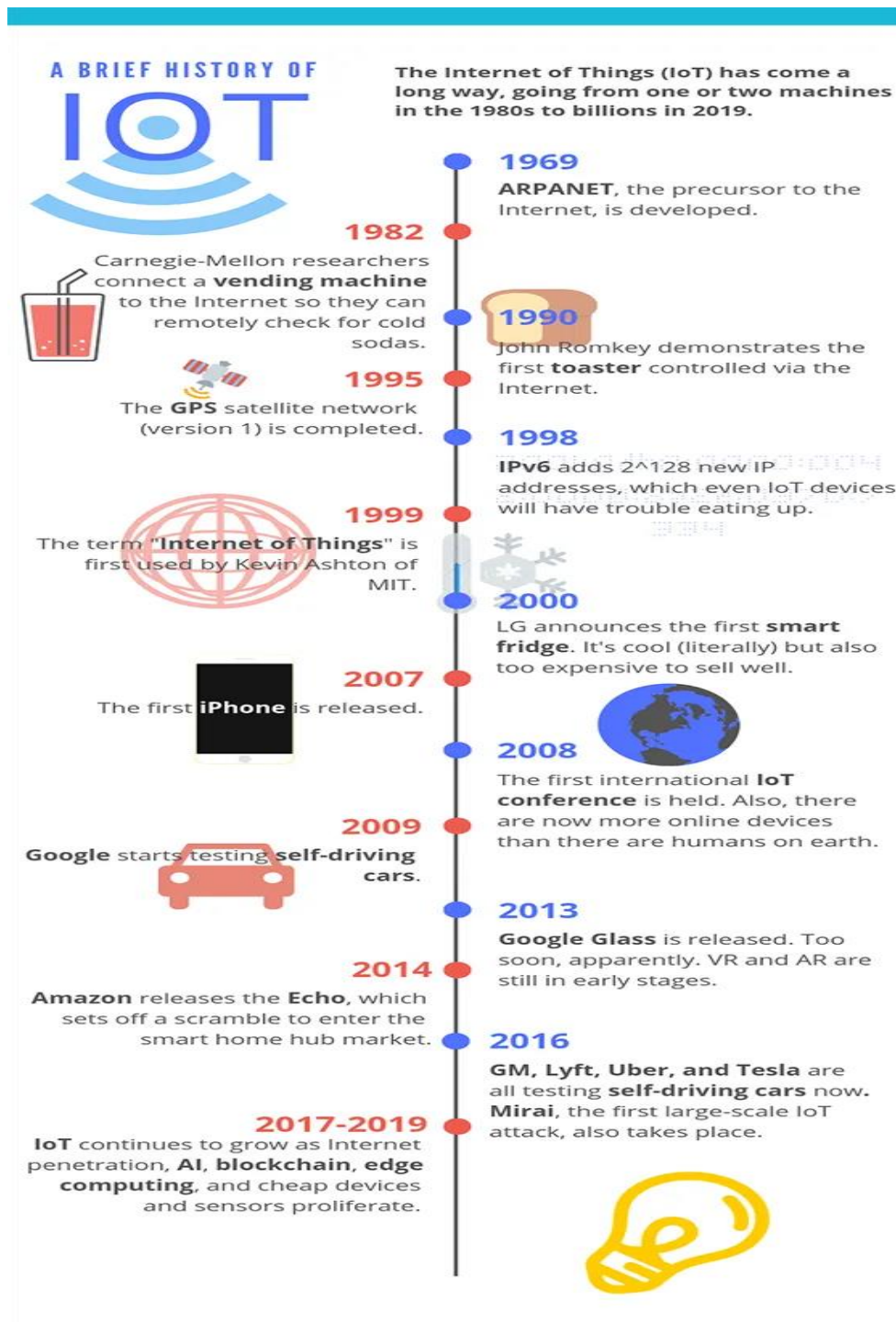
Συνολικά, το IoT αναφέρεται στην διασύνδεση φυσικών αντικειμένων με αισθητήρες, ενεργοποιητές και τεχνολογίες επικοινωνίας που αποσκοπούν στην ανταλλαγή δεδομένων, τη δυνατότητα ευφυών υπηρεσιών και την παρακολούθηση/έλεγχο φυσικών περιβαλλόντων.

2.3 Ιστορική Αναδρομή

Η έννοια του Διαδικτύου των Πραγμάτων - σύμφωνα με τους παραπάνω ορισμούς- , δεν είναι καινούργια. Οι πρώτες συνδεδεμένες συσκευές στο Διαδίκτυο, ήταν οι αυτόματοι πωλητές προϊόντων (vending machines) και τα ATM, τα οποία δημιουργήθηκαν περίπου την δεκαετία του 1980. Ο πρώτος αυτόματος πωλητής ήταν της Coca-Cola (1982), ενώ το πρώτο ATM ξεκίνησε να λειτουργεί στην Νέα Υόρκη (1969). Ο πρώτος που εισήγαγε την έννοια του διαδικτύου των πραγμάτων ήταν ο βρετανός επιστήμονας στο τομέα της πληροφορικής Kevin Ashton το 1999. Ο Ashton υποστήριξε την εισαγωγή της τεχνολογίας του RFID (Radio-frequency identification) σε προϊόντα, με σκοπό την παρακολούθηση τους στην εφοδιαστική αλυσίδα. Στην Εικόνα 2 φαίνεται συνοπτικά η ιστορία του διαδικτύου των πραγμάτων.

Αξιοσημείωτο είναι επίσης, το γεγονός ότι αυτή την στιγμή υπάρχουν περισσότερες από 14 δισεκατομμύρια [4] IoT συσκευές παγκοσμίως, με τον αριθμό των συσκευών να συνεχίζει να αυξάνεται κατακόρυφα για τα επόμενα χρόνια.

Πηγή : <https://www.iottechrends.com/history-of-iot/>

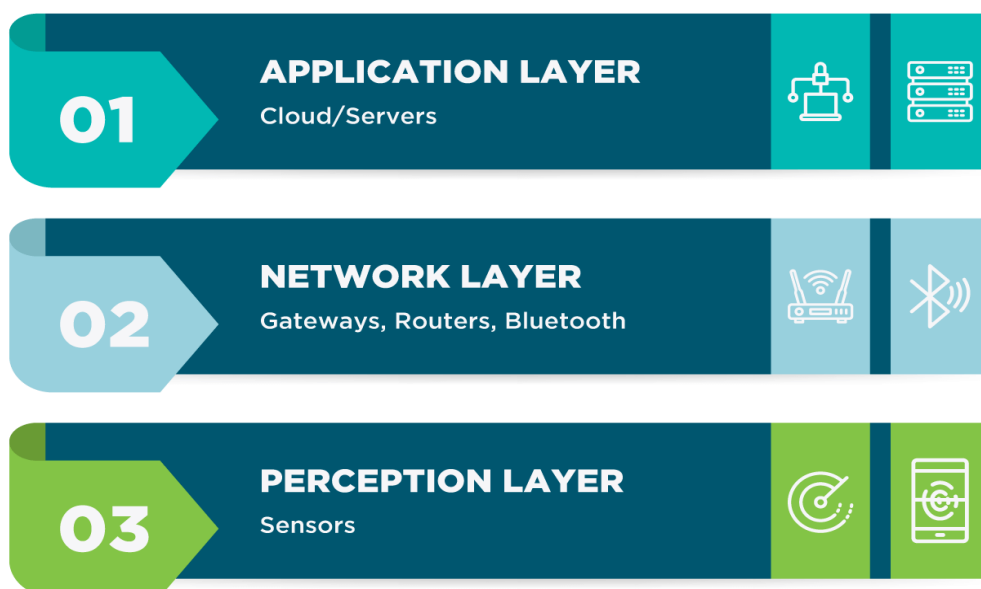


Εικόνα 2: Ιστορία του Internet of Things

2.4 Αρχιτεκτονική

Λόγω της εξελισσόμενης φύσης του IoT - τόσο των συσκευών, όσο και των αισθητήρων - δεν υπάρχει μία ενιαία αρχιτεκτονική για όλες τις περιπτώσεις χρήσης αυτής της τεχνολογίας. Έτσι βλέπουμε να υπάρχουν διαφορετικές απόψεις σχετικά με τον αριθμό των επιπέδων στο IoT. Ωστόσο σύμφωνα με αρκετούς ερευνητές [2] η διασθρωματωμένη αρχιτεκτονική του IoT λειτουργεί σε κυρίως τρία επίπεδα. Τα επίπεδα αυτά είναι 1) το επίπεδο αντίληψης, 2) το επίπεδο δικτύου και τέλος 3) το επίπεδο εφαρμογής.

THREE-LAYER ARCHITECTURE



Εικόνα 3: Αρχιτεκτονική 3 επιπέδων του IoT

Πηγή : <https://jelvix.com/blog/iot-architecture-layers>

Επίπεδο Αντίληψης (Perception Layer)

Το επίπεδο αντίληψης - όπως φαίνεται και από την εικόνα 3 - ονομάζεται και αλλιώς επίπεδο αισθητήρων. Ο σκοπός αυτού του επιπέδου είναι να αποκτήσει τα δεδομένα από το περιβάλλον με τη βοήθεια αισθητήρων. Το επίπεδο αυτό είναι υπεύθυνο για την ανίχνευση, τη συλλογή και την τοπική επεξεργασία των δεδομένων, όπως επίσης και για την πραγματοποίηση κατάλληλων ενεργειών πριν γίνει μετάδοση αυτών των δεδομένων στο επίπεδο δικτύου.

Επίπεδο Δικτύου (Network Layer)

Το επίπεδο δικτύου εξυπηρετεί τη λειτουργία της μετάδοσης και της δρομολόγησης των δεδομένων από και σε διαφορετικούς κόμβους και συσκευές IoT μέσω του Διαδικτύου. Σε αυτό το επίπεδο, πλατφόρμες υπολογιστικού νέφους, πύλες δικτύου (Gateways), συσκευές μεταγωγής (Switches) και δρομολόγησης (Routers) κ.λπ. λειτουργούν χρησιμοποιώντας κάποιες από τις πιο πρόσφατες τεχνολογίες μετάδοσης όπως το WiFi, LTE, Bluetooth, 3G, Zigbee, LoRaWAN κ.λπ. Τα gateways χρησιμεύουν - ως διαμεσολαβητές (υπό μια έννοια access points) μεταξύ διάφορων οντοτήτων του IoT - για την συγκέντρωση, το φιλτράρισμα, και τη μετάδοση των δεδομένων από και προς διαφορετικές IoT συσκευές.

Επίπεδο Εφαρμογής (Application Layer)

Το επίπεδο εφαρμογής εγγυάται την αυθεντικότητα, την ακεραιότητα, και την εμπιστευτικότητα των δεδομένων. Επίσης το επίπεδο εφαρμογής αφορά τη συγκέντρωση, επεξεργασία, ανάλυση, απόκτηση γνώσης, εφαρμογή ευφυών αλγορίθμων και συστημάτων απόφασης.

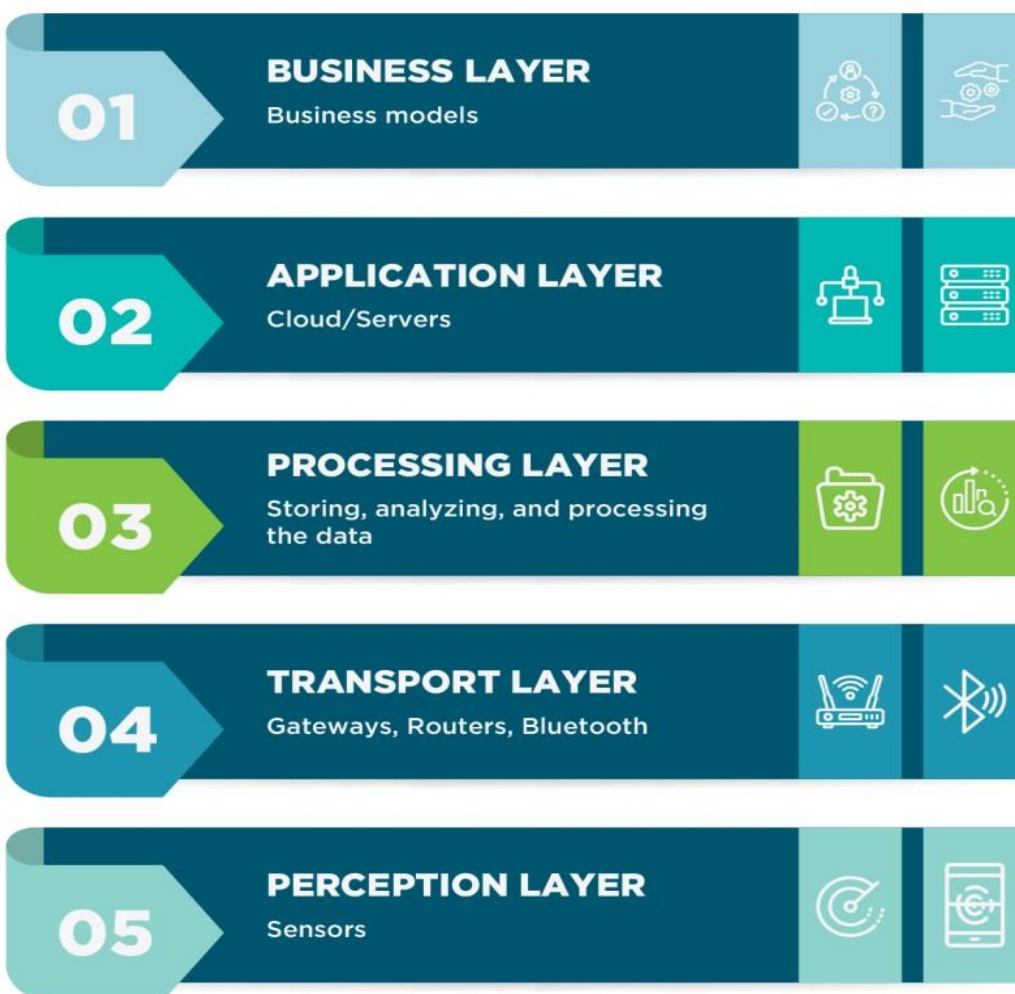
Το επίπεδο εφαρμογής είναι αυτό που βλέπουν οι χρήστες. Αυτό θα μπορούσε να είναι μια εφαρμογή για τον έλεγχο μιας συσκευής ή μιας ομάδας συσκευών σε ένα οικοσύστημα έξυπνου σπιτιού ή ένας πίνακας εργαλείων που δείχνει την κατάσταση των συσκευών που αποτελούν μέρος ενός συστήματος.

Εκτός από την αρχιτεκτονική τριών επιπέδων, υπάρχει και αυτή των 5. Η αρχιτεκτονική 5 επιπέδων θεωρείται ως η καλύτερη προτεινόμενη αρχιτεκτονική IoT. Η αρχιτεκτονική αυτή μπορεί να θεωρηθεί μια επέκταση της βασικής αρχιτεκτονικής. Τα δύο επιπλέον επίπεδα που αποτελούν κομμάτι της αρχιτεκτονικής αυτής είναι το Processing Layer και το Business Layer. Παρακάτω αναλύονται τα 2 επιμέρους επίπεδα.

Επίπεδο επεξεργασίας (Processing Layer)

Στο επίπεδο επεξεργασίας συγκεντρώνονται όλα τα δεδομένα που παρέχονται από τα επίπεδα αντίληψης μέσω ενός δικτύου. Όλες οι πληροφορίες αποθηκεύονται και αναλύονται. Είναι ένα από τα πιο σημαντικά επίπεδα του IoT επειδή λαμβάνει αποφάσεις με βάση την ανάλυση δεδομένων. Αυτό το επίπεδο τοποθετείται μεταξύ των επιπέδων Δικτύου και Εφαρμογών στον λογικό σχεδιασμό της αρχιτεκτονικής του IoT.

FIVE-LAYER ARCHITECTURE



Εικόνα 4: Αρχιτεκτονική 5 επιπέδων του IoT

Πηγή : <https://jelvix.com/blog/iot-architecture-layers>

Επιχειρηματικό Επίπεδο (Business Layer)

Το Business Layer, είναι το υψηλότερο επίπεδο στην αρχιτεκτονική 5 επιπέδων IoT, διαδραματίζει κρίσιμο ρόλο στη διαχείριση και λειτουργία εφαρμογών, υπηρεσιών και υποδομών IoT. Είναι υπεύθυνο για την εκτέλεση των επιχειρηματικών μοντέλων IoT, τη διαχείριση της αλληλεπίδρασης μεταξύ των υπηρεσιών και το χειρισμό των αλυσίδων αξίας για τις εκάστοτε υπηρεσίες και προϊόντα. Με τα δεδομένα να έχουν υποστεί επεξεργασία και αναλυθεί από τα κατώτερα επίπεδα, το Business Layer ενημερώνει τη λήψη στρατηγικών αποφάσεων. Επιπλέον, διέπει την εφαρμογή πρωτοκόλλων ασφαλείας και πολιτικών απορρήτου, διασφαλίζοντας την προστασία των ευαίσθητων δεδομένων και διατηρώντας το απόρρητο των χρηστών. Τέλος, διαχειρίζεται τις διεπαφές για την αλληλεπίδραση των χρηστών, παρέχοντας ένα μέσο για τους χρήστες να εμπλακούν με το σύστημα IoT.

2.4 Χαρακτηριστικά

Το Διαδίκτυο των Πραγμάτων έχει αρκετά χαρακτηριστικά τα οποία το διακρίνουν από άλλες τεχνολογίες. Κάποια από αυτά τα χαρακτηριστικά είναι :

- **Πανταχού Παρουσία (Ubiquity):** οι συσκευές IoT μπορούν να βρεθούν σχεδόν σε κάθε τομέα, συμπεριλαμβανομένης της υγειονομικής περίθαλψης, των μεταφορών, της γεωργίας, της ενέργειας και πολλούς άλλους τομείς.
- **Συνδεσιμότητα (Connectivity):** οι συσκευές IoT είναι συνδεδεμένες στο δίκτυο και μπορούν να επικοινωνούν μεταξύ τους, επιτρέποντας την ανταλλαγή και ανάλυση δεδομένων σε πραγματικό χρόνο.
- **Αυτόνομη Λειτουργία (Autonomous Operation):** Η ικανότητα των συστημάτων και συσκευών IoT να λειτουργούν αυτόνομα και να λαμβάνουν αποφάσεις χωρίς ανθρώπινη παρέμβαση αναφέρεται ως αυτόνομη λειτουργία. Αυτή είναι μια σημαντική δυνατότητα IoT (Internet of Things) που επιτρέπει μια ποικιλία νέων εφαρμογών και υπηρεσιών.
- **Ευφυΐα (Intelligence):** Οι συσκευές IoT είναι εξοπλισμένες με αισθητήρες και λογισμικό που τους επιτρέπουν να συλλέγουν και μετά από την επεξεργασία των δεδομένων είτε τοπικά είτε συνήθως στο cloud, καθίστανται έξυπνες και ικανές να λαμβάνουν αποφάσεις.
- **Διαλειτουργικότητα (Interoperability):** Οι συσκευές IoT μπορούν να επικοινωνούν μεταξύ τους ανεξάρτητα από τον κατασκευαστή και άρα το υλικό (hardware), το λειτουργικό σύστημα ή το πρωτόκολλο επικοινωνίας του
- **Ασφάλεια (Security):** Οι συσκευές IoT πρέπει να είναι ασφαλείς – κάτι το οποίο δεν ισχύει συνήθως και είναι μια από τις προκλήσεις για τον τομέα του IoT - για την προστασία ευαίσθητων δεδομένων και την αποτροπή μη εξουσιοδοτημένης πρόσβασης.

- **Επεκτασιμότητα (Scalability):** Τα δίκτυα IoT μπορούν να αυξηθούν ή να μειωθούν ανάλογα με τον αριθμό των συσκευών και την ποσότητα των δεδομένων που υποβάλλονται σε επεξεργασία.
- **Δυναμικές Αλλαγές (Dynamic Changes):** οι συσκευές αυτές πρέπει να μπορούν να αλλάζουν την κατάστασή τους δυναμικά, π.χ. να μπορούν να απενεργοποιούνται (δηλαδή, να μην καταναλώνουν ενέργεια) και να ενεργοποιούνται. Ακόμα μπορεί να αλλάζει το γενικό πλαίσιο των συσκευών αυτών όπως π.χ. η τοποθεσία τους.

2.5 Απαιτήσεις

Το Διαδίκτυο των Πραγμάτων τόσο σε επίπεδο δικτυακό, όσο και σε επίπεδο υλικού απαιτεί διάφορα στοιχεία για να μπορέσει να λειτουργεί σωστά. Κάποια από αυτά είναι:

- **Αισθητήρες (Sensors):** οι συσκευές IoT έχουν ενσωματωμένους αισθητήρες οι οποίοι συλλέγουν πληροφορίες από τον φυσικό κόσμο σε πραγματικό χρόνο. Κάποια δεδομένα που μπορεί μια IoT συσκευή να συλλέξει είναι π.χ. θερμοκρασία, υγρασία κτλ.
- **Συνδεσιμότητα (Connectivity):** οι συσκευές αυτές πρέπει να είναι σε θέση να επικοινωνούν με το διαδίκτυο, ώστε τα δεδομένα αυτά να μπορούν να χρησιμοποιηθούν για περαιτέρω χρήσεις, όπως την αποθήκευση και την ανάλυση τους.
- **Αποθήκευση Δεδομένων (Data Storage):** η αποθήκευση των δεδομένων που παράγονται - τα οποία ονομάζονται και Big Data λόγω του μεγάλου όγκου δεδομένων από τις πολλές μετρήσεις που έρχονται από πάρα πολλές εγκατεστημένες συσκευές - πρέπει να αποθηκευτούν, ώστε μετέπειτα να μπορεί να γίνει ανάλυση τους για εξαγωγή σημασιολογικού περιεχομένου.
- **Μεγάλη Διάρκεια Μπαταρίας (Long Battery Life):** για συσκευές οι οποίες δεν έχουν σταθερή παροχή ρεύματος από το δίκτυο, η ενεργειακή απόδοση είναι ένα μια μετρική η οποία είναι πολύ σημαντική. Για τον παραπάνω λόγο θα πρέπει να γίνεται σωστός σχεδιασμός τόσο σε επίπεδο λογισμικού όσο και σε επίπεδο υλικού, ώστε να αξιοποιείται με αποδοτικό τρόπο η μπαταρία.

Εκτός από τα παραπάνω, πρέπει να επισημανθεί πως όπως προαναφέρθηκε και στο [2.1.2](#) η επεκτασιμότητα, διαλειτουργικότητα και η ασφάλεια είναι και αυτές κάποιες από τις απαιτήσεις που έχουν τα IoT συστήματα.

2.6 Περιπτώσεις Χρήσης

Το Διαδίκτυο των Πραγμάτων έχει πολλές περιπτώσεις χρήσης σε διάφορους κλάδους. Στην εικόνα 5 φαίνεται μια πλήρη ταξινόμηση του IoT στα σημαντικά πεδία εφαρμογής. Κύριοι τομείς εφαρμογής είναι η υγειονομική περίθαλψη, το περιβάλλον, οι έξυπνες πόλεις, οι εμπορικές και βιομηχανικές υποδομές. Στη συνέχεια πραγματοποιείται μια σύντομη περιγραφή για κάθε πεδίο εφαρμογής [\[1\]](#) [\[6\]](#) [\[7\]](#).



Εικόνα 5: Περιπτώσεις χρήσης του IoT

Πηγή : <https://www.semanticscholar.org/paper/Internet-of-Things%3A-A-Survey-on-Enabling-Protocols%2C-AI-Fugaha-Guizani/84f25ddd053e414f239b91552410dab0adbaedad>

Έξυπνη Υγεία (Smart Healthcare)

Η χρήση του Διαδικτύου των Πραγμάτων στην υγειονομική περίθαλψη, έχει την δυνατότητα να αλλάξει εντελώς τον τρόπο παροχής υγειονομικής περίθαλψης, παρακολούθησης και θεραπείας των ασθενών όχι μόνο στα νοσοκομεία αλλά και μετά στα σπίτια τους. Οι ασθενείς και οι επαγγελματίες υγείας πλέον με τη χρήση των IoT συσκευών μπορούν να παρακολουθούν ένα μεγάλο εύρος ζωτικών δεικτών, συμπεριλαμβανομένων των καρδιακών παλμών, της αρτηριακής πίεσης και των επιπέδων γλυκόζης κ.α. Οι γιατροί αντίστοιχα, μπορούν και αυτοί, να παρακολουθούν τα δεδομένα αυτά σε πραγματικό χρόνο και να τα χρησιμοποιούν για να αλλάξουν πιθανώς κάποια δοσολογία στα φάρμακα ή να αποφασίσουν για άλλες θεραπείες. Από τα παραπάνω μπορούμε να συμπεράνουμε πως η φροντίδα των ασθενών γίνεται εξατομικευμένη, ενώ αποφεύγονται σε αρκετές περιπτώσεις οι άσκοπες επισκέψεις στο νοσοκομείο και τέλος αυτό οδηγεί σε μικρότερες δαπάνες για την υγειονομική περίθαλψη τους. Κάποιες από τις προκλήσεις σε αυτόν το τομέα είναι η ασφάλεια/απόρρητο των δεδομένων και η διαλειτουργικότητα των διαφορετικών συστημάτων και συσκευών.

Έξυπνο Σπίτι (Smart Home)

Αναμφισβήτητα, μία από τις πιο γνωστές και δημοφιλείς περιπτώσεις εφαρμογών του IoT είναι το Έξυπνο Σπίτι. Οι εφαρμογές αυτές έκαναν το IoT γνωστό στο ευρύ κοινό. Η ιδέα του Έξυπνου Σπιτιού είναι η δυνατότητα των ιδιοκτητών ή των ενοίκων να παρακολουθούν και να ελέγχουν από απόσταση τις οικιακές τους συσκευές. Κάποια παραδείγματα είναι ο έξυπνος φωτισμός, οι κάμερες ασφαλείας, οι έξυπνοι θερμοστάτες και άλλες συσκευές, οι οποίες μπορούν να λειτουργούν και ελέγχονται εξ αποστάσεως μέσω κάποιας εφαρμογής που έχει διεπαφή χρήστη (π.χ. android app σε smartphone). Το πλήθος των προσιτών τιμών αυξάνεται με αποτέλεσμα η πρόσβαση στις έξυπνες οικιακές συσκευές να γίνεται πιο εύκολη, καθιστώντας πιο προσιτό για τους ιδιοκτήτες κατοικιών να ενσωματώσουν το IoT στους χώρους τους. Και σε αυτή την περίπτωση χρήσης του Διαδικτύου των Πραγμάτων αξίζει να αναφερθούμε σε κάποιες προκλήσεις και κινδύνους που ενέχουν αυτά τα συστήματα. Μία σημαντική πρόκληση φαίνεται να είναι και εδώ η διαλειτουργικότητα των συσκευών IoT, μιας και αρκετές φορές για να καλυφθεί μια μεγάλη περιπτώσιολογία χρήσης (π.χ. έξυπνου φωτισμού, ρολών για τα παράθυρα, συστήματα ασφαλείας, αισθητήρια χώρου κτλ...) πρέπει να γίνει αγορά συσκευών από διαφορετικούς κατασκευαστές, οι οποίες όμως δεν ελέγχονται μέσω μιας ενιαίας πλατφόρμας, καθιστώντας τις έτσι δύσχρηστες. Τέλος το μεγαλύτερο πρόβλημα και στα έξυπνα σπίτια, είναι το θέμα της ασφάλειας και ιδιωτικότητας των συσκευών/εφαρμογών αυτών.

Έξυπνες Πόλεις (Smart Cities)

Με περισσότερο από το 50% του παγκόσμιο πληθυσμού να κατοικεί στα αστικές περιοχές, είναι φυσικό επακόλουθο η τεχνολογία του IoT να έχει θέση στην ανάπτυξη Έξυπνων Πόλεων. Οι έξυπνες πόλεις σκοπό έχουν να αναβαθμίσουν την ποιότητα ζωής των πολιτών, μέσω καλύτερων υποδομών και δημόσιων υπηρεσιών. Κάποια παραδείγματα από τις εφαρμογές των έξυπνων πόλεων είναι οι εξής; 1) Έξυπνο σύστημα μεταφοράς, το οποίο περιλαμβάνει έξυπνα συστήματα δημόσιων μεταφορών, προγράμματα κοινής χρήσης ποδηλάτων, όπως και έξυπνες λύσεις στάθμευσης. 2) Έξυπνη διαχείριση ενέργειας, η οποία περιλαμβάνει χρήση έξυπνων δικτύων διανομής (smart grids), ενεργειακά αποδοτικά συστήματα φωτισμού τα οποία μπορούν να ελέγχονται απομακρυσμένα και μόνο όταν χρειάζονται. 3) Έξυπνη διαχείριση απορριμμάτων, όπου μέσα από κατάλληλους αισθητήρες παράγονται δεδομένα τα οποία μετέπειτα αναλύονται για την παρακολούθηση των επιπέδων αποβλήτων στους κάδους, με στόχο την βελτιστοποίηση της συλλογής των απορριμμάτων. Τέλος μία ακόμα περίπτωση είναι αυτή της 4) Έξυπνης διαχείρισης του νερού, η οποία περιλαμβάνει αισθητήρες που μετρούν σε πραγματικό χρόνο την ποιότητα του νερού και άλλων μετρικών (π.χ. πίεση) μέσω των οποίων μπορούν να εξαχθούν χρήσιμα συμπεράσματα όπως εάν υπάρχει διαρροή νερού.

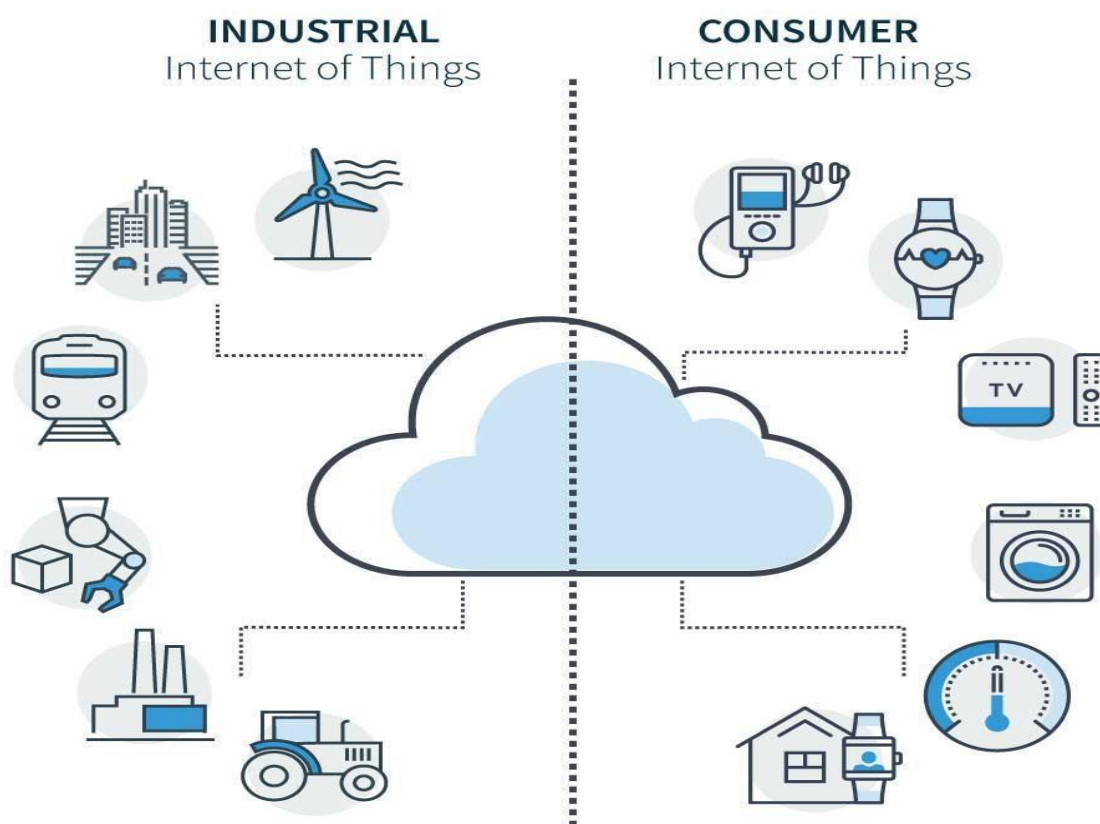
Μερικές ακόμα περιπτώσεις χρήσεις του Διαδικτύου των Πραγμάτων είναι οι ακόλουθες: 1) Φορητές Συσκευές (wearable technologies) όπως το έξυπνο ρολόι (smart watch), 2) Βιομηχανικό IoT (ή IIoT) και τέλος η 3) Έξυπνη Γεωργία.

3. Industrial IoT - IIoT

3.1 Εισαγωγή

Το Industrial Internet of Things (IIoT) είναι μια ειδική κατηγορία IoT που επικεντρώνεται σε εφαρμογές και περιπτώσεις χρήσης στη σύγχρονη βιομηχανία και τον 'έξυπνο' κατασκευαστικό τομέα (smart manufacturing). Το IIoT, που χρησιμοποιείται στο πλαίσιο της 4ης Βιομηχανικής Επανάστασης ή αλλιώς και Industry 4.0, μπορεί να θεωρηθεί ως ένα σύνθετο σύστημα που απαρτίζεται από πολλές/διαφορετικές τεχνολογίες.

Το IIoT φέρνει επανάσταση στις βιομηχανίες, επιτρέποντας νέα επίπεδα αποδοτικότητας, παραγωγικότητας και καινοτομίας στις παραγωγικές διαδικασίες. Πολλοί οργανισμοί αναγνωρίζουν τη σημασία του IIoT. Για παράδειγμα, η Accenture αναφέρει ότι το IIoT θα μπορούσε να προσθέσει 14,2 τρισεκατομμύρια δολάρια στην παγκόσμια οικονομία μέχρι το 2030 (Accenture, 2015) [8]. Επιπλέον, μια μελέτη που διεξήχθη από την IDC δείχνει ότι οι παγκόσμιες δαπάνες για το IIoT αναμένεται να φτάσουν τα 1,1 τρισεκατομμύρια δολάρια έως το 2023, καταδεικνύοντας την αυξανόμενη σημασία του IIoT σε μια σειρά από διαφορετικούς τομείς (IDC, 2020).



Εικόνα 6: Διαφορετικές εφαρμογές του IIoT και IoT

Πηγή : <https://www.smartrek.io/dissecting-the-industrial-iiot/>

3.2 Ορισμός

Στη συγκεκριμένη ενότητα θα γίνει μια εκτενής καταγραφή για τους ορισμούς του IIoT. Με βάση τους προηγούμενους ορισμούς για το IoT και επεκτείνοντας τους, ένας ορισμός για το IIoT είναι ο εξής: το Βιομηχανικό Διαδίκτυο των Πραγμάτων (IIoT) είναι η χρήση των τεχνολογιών του IoT προσανατολισμένες σε βιομηχανικά περιβάλλοντα. Ο παραπάνω ορισμός είναι μια απλουστευμένη εκδοχή για τον ορισμό του IIoT. Επειδή μια τόσο απλουστευμένη εκδοχή για τον ορισμό του IIoT είναι ανεπαρκής, οι ερευνητές [9] εισήγαγαν δύο χαρακτηριστικά τα οποία είναι ουσιώδη για να δοθεί ένας ακριβής ορισμός. Τα χαρακτηριστικά αυτά είναι: 1) τα είδη των τεχνολογιών που χρησιμοποιούνται σε ένα περιβάλλον IIoT και 2) οι διακριτοί στόχοι και σκοποί των οποίων την επίτευξη υπηρετούν οι τεχνολογίες. Με βάση τα παραπάνω, είναι εύκολο να ξεχωρίσουμε τις έξυπνες συσκευές που χρησιμοποιούνται στο IoT, από αυτές που χρησιμοποιούνται στο IIoT. Ο ορισμός των

συγγραφέων για το IIoT τελικά είναι ο εξής: «Βιομηχανικό Διαδίκτυο των Πραγμάτων: ένα σύστημα που περιλαμβάνει δικτυωμένα έξυπνα αντικείμενα, κυβερνοφυσικά συστήματα, γενικές τεχνολογίες πληροφοριών και προαιρετικά cloud και edge πλατφόρμες, που επιτρέπουν σε πραγματικό χρόνο, την ευφυή και αυτόνομη, πρόσβαση, συλλογή, ανάλυση, επικοινωνία και ανταλλαγή διεργασιών, προϊόντων ή/και υπηρεσιών, εντός του βιομηχανικού περιβάλλοντος, ώστε να βελτιστοποιηθεί η συνολική αξία παραγωγής. Αυτή η αξία περιλαμβάνει: τη βελτίωση της παράδοσης προϊόντων και υπηρεσιών, τη μείωση του κόστους εργασίας, τη μείωση της ενεργειακής κατανάλωσης και τη μείωση του κύκλου κατασκευής ανά παραγγελία.»

Άλλοι ορισμοί που υπάρχουν σε ερευνητικά άρθρα και όχι μόνο είναι οι εξής:

- 1) Σύμφωνα με την Amazon [10] : το IIoT «συνδέει/συνενώνει μαζί μηχανήματα, υπολογιστικό νέφος, ανάλυση δεδομένων και ανθρώπους για να βελτιώσει την απόδοση και την παραγωγικότητα των βιομηχανικών διαδικασιών. Με το IIoT οι βιομηχανίες μπορούν να ψηφιοποιήσουν διαδικασίες, να μεταμορφώσουν τα επιχειρηματικά μοντέλα και να βελτιώσουν την απόδοση και την παραγωγικότητα, μειώνοντας παράλληλα τις σπατάλες. Οι εταιρείες που δραστηριοποιούνται σε τομείς όπως οι κατασκευές, η ενέργεια, η γεωργία, οι μεταφορές, εργάζονται σε έργα IoT που συνδέουν δισεκατομμύρια συσκευές, προσφέροντας αξία σε διάφορες περιπτώσεις χρήσης όπως την ανάλυση πρόβλεψης και ποιότητας συντήρησης, την παρακολούθηση της κατάστασης περιουσιακών στοιχείων και τέλος την βελτιστοποίηση διαδικασιών.»
- 2) Ο ορισμός που υπάρχει στο wikipedia [11] είναι ο εξής: «Το βιομηχανικό διαδίκτυο των πραγμάτων (IIoT) αναφέρεται σε διασυνδεδεμένους αισθητήρες, όργανα και άλλες συσκευές που είναι δικτυακά συνδεδεμένα με βιομηχανικές εφαρμογές υπολογιστών, συμπεριλαμβανομένου του κατασκευαστικού τομέα και του τομέα της διαχείρισης ενέργειας. Αυτή η συνδεσιμότητα επιτρέπει τη συλλογή, ανταλλαγή και ανάλυση δεδομένων, διευκολύνοντας ενδεχόμενες βελτιώσεις που αφορούν την

παραγωγικότητα και την αποδοτικότητα καθώς και άλλα οικονομικά οφέλη. Το IIoT είναι μία εξέλιξη ενός κατανεμημένου συστήματος ελέγχου (DCS - distributed control system) που επιτρέπει υψηλότερο βαθμό αυτοματοποίησης χρησιμοποιώντας το υπολογιστικό νέφος για την βελτίωση και την βελτιστοποίηση των ελέγχων διαδικασίας»

3.3 Δομικά Στοιχεία του IIoT

Απαραίτητο για να μπορεί να υπάρξει το Βιομηχανικό Διαδίκτυο των Πραγμάτων είναι η ύπαρξη μιας ολοκληρωμένης υποδομής. Για τον λόγο αυτό θα γίνει αναφορά στα δομικά μέρη που επιτρέπουν την ύπαρξη του IIoT. Εν' συντομία αυτά είναι 1) Αισθητήρες και Ενεργοποιητές (Sensors/Actuators), 2) Edge Συσκευές, 3) Πλατφόρμες υπολογιστικού νέφους (Cloud Computing Platforms), 4) Πρωτόκολλα επικοινωνίας και 5) Μηχανισμοί Ασφαλείας.

Αισθητήρες και Ενεργοποιητές

Οι **αισθητήρες** είναι στοιχεία των έξυπνων συσκευών και χρησιμοποιούνται για την μέτρηση διαφορετικών πτυχών του φυσικού κόσμου όπως π.χ. η θερμοκρασία, η υγρασία και η κίνηση. Τα δεδομένα από του αισθητήρες μπορούν να χρησιμοποιηθούν για την 1) απόκτηση πληροφοριών, 2) για τη βελτίωση της λήψης αποφάσεων, 3) την βελτιστοποίηση διαδικασιών και 4) την αυτοματοποίηση εργασιών σε διάφορους κλάδους. Οι αισθητήρες IoT μπορούν να κατηγοριοποιηθούν με διάφορους τρόπους, μεταξύ των οποίων:

- *Τύπος της φυσικής ποσότητας που μετράται:* ήχος, φως, θερμοκρασία, ηλεκτρομαγνητική ακτινοβολία, επιτάχυνση, ταχύτητα και χρώμα, είναι μερικά μόνο παραδείγματα τύπων αισθητήρων IoT με βάση το τι μετρούν.
- *Χρήση ενέργειας:* ανάλογα με το εάν χρειάζονται ή όχι μια εξωτερική πηγή ενέργειας, οι αισθητήρες IoT ταξινομούνται ως ενεργοί ή παθητικοί αισθητήρες. Οι IoT-enabled RFID κάρτες είναι ένα παράδειγμα παθητικών αισθητήρων.
- *Μέθοδος μέτρησης:* Οι αισθητήρες IoT χρησιμοποιούν διαφορετικούς μηχανισμούς μέτρησης και μπορούν να κατηγοριοποιηθούν, σε ηλεκτρικούς αισθητήρες, οπτικούς αισθητήρες, χημικούς αισθητήρες και γυροσκοπικούς αισθητήρες.
- *Ακρίβεια:* όσον αφορά το εύρος μέτρησης, οι αισθητήρες IoT μπορούν να ταξινομηθούν ως αισθητήρες υψηλής και χαμηλής ακρίβειας. Οι αισθητήρες IoT υψηλής ακρίβειας χρησιμοποιούνται συνήθως σε ελεγχόμενα περιβάλλοντα και μπορούν να μετρήσουν πολύ μικρές αλλαγές, ενώ οι αισθητήρες χαμηλής ακρίβειας μπορούν να μετρήσουν μόνο μεγάλες αλλαγές.

- **Τύπος εξόδου:** με βάση τον τύπο εξόδου που παράγουν, οι αισθητήρες IoT μπορούν να κατηγοριοποιηθούν σε αναλογικούς ή ψηφιακούς αισθητήρες. Οι αναλογικοί αισθητήρες έχουν μια συνεχώς μεταβαλλόμενη έξοδο - για παράδειγμα, μια αντίσταση που μεταβάλλεται συνεχώς ανάλογα με τη θερμοκρασία (thermistor). Ενώ οι ψηφιακοί αισθητήρες ψηφιοποιούν το σήμα και στέλνουν κβαντισμένη την πληροφορία (σε bits) πίσω.

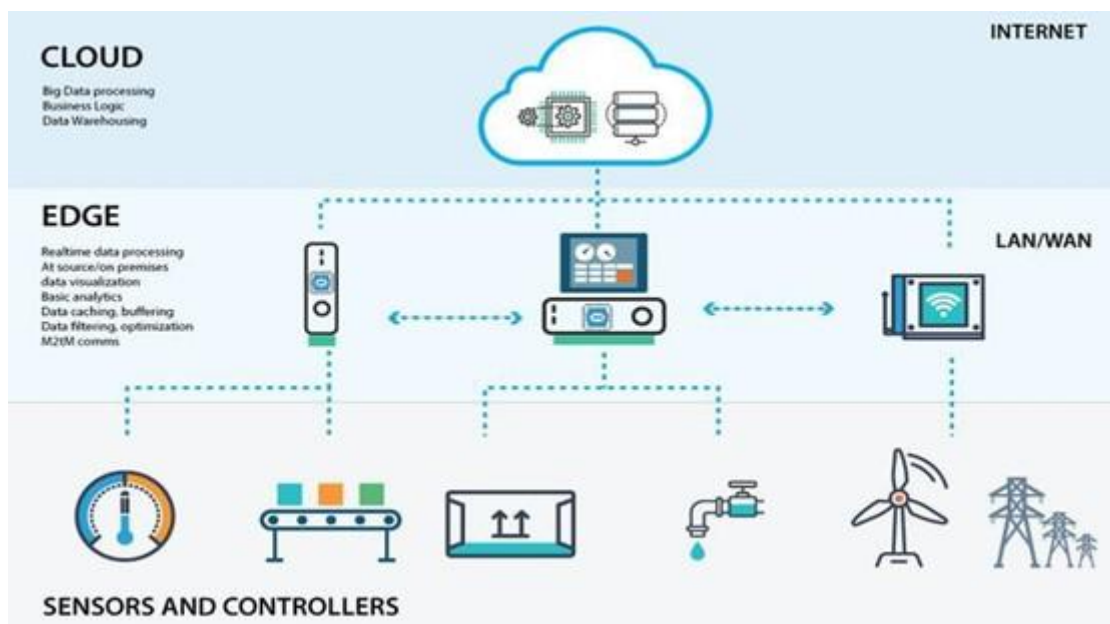
Κάποιο ενδεικτικοί τύποι αισθητήρων είναι εξής: 1) αισθητήρες ροής, 2) αισθητήρες δύναμης, 3) αισθητήρες υγρασίας, 4) αισθητήρες πίεσης, 5) φωτοηλεκτρικοί αισθητήρες, 6) αισθητήρες στάθμης νερού, 7) αισθητήρες υπερήχων.

Οι **ενεργοποιητές** είναι εξαρτήματα που κινούν ή ελέγχουν τον μηχανισμό ενός συστήματος μέσω της εφαρμογής / backend. Οι αισθητήρες ανιχνεύουν το περιβάλλον και στη συνέχεια δημιουργούν σήματα ελέγχου για τους ενεργοποιητές, οι οποίοι μετατρέπουν την ηλεκτρική ενέργεια σε μηχανική κίνηση. Ένας σερβοκινητήρας είναι ένα παράδειγμα ενεργοποιητή, ο οποίος μπορεί να μετακινηθεί σε μία δεδομένη θέση γωνιακή ή γραμμική. Κάποιοι τύποι διαφορετικών ενεργοποιητών είναι οι εξής: 1) υδραυλικοί ενεργοποιητές, 2) πνευματικοί ενεργοποιητές, 3) ηλεκτρικοί ενεργοποιητές, 4) θερμικοί/μαγνητικοί ενεργοποιητές, 5) μηχανικοί ενεργοποιητές, 6) μαλακοί (soft) ενεργοποιητές.

Edge Συσκευές

Το Edge computing είναι ένα κατανεμημένο υπολογιστικό πρότυπο αρχιτεκτονικής που φέρνει τον υπολογισμό και την αποθήκευση δεδομένων πιο κοντά στις πηγές δεδομένων [\[12\]](#). Ο κύριος λόγος για την χρήση υπολογιστικών συστημάτων στην άκρη του δικτύου (Edge computing) είναι το ισοζύγιο μεταξύ καθυστέρησης, κόστους και αξιοπιστίας. Η καθυστέρηση χαλάει την ποιότητα εμπειρίας των χρηστών, ενώ σε περιβάλλοντα όπως αυτά για τα οποία θίγουμε (δηλ. βιομηχανικά) είναι κρίσιμης σημασίας. Ακόμα το εύρος ζώνης μπορεί να είναι σε αρκετές περιπτώσεις αρκετά ακριβό εάν π.χ. χρησιμοποιούνται δεδομένα κινητής τηλεφωνίας, επομένως είναι λογικό να γίνεται χρήση των πόρων των edge συσκευών για την επεξεργασία/αποθήκευση των παραγόμενων δεδομένων τοπικά. Επίσης για κρίσιμες εφαρμογές όπου η αξιοπιστία είναι εξαιρετικά σημαντική και η συνδεσιμότητα δικτύου δεν μπορεί να είναι εγγυημένη, στο edge μοντέλο λειτουργίας διασφαλίζεται ότι οι συσκευές θα συνεχίσουν να λειτουργούν κάνοντας τις απαιτούμενες εργασίες τοπικά, αντί να βασίζομαστε στην επεξεργαστική ισχύ που παρέχει το cloud.

Οι Edge συσκευές μπορούν να οριστούν ως κομμάτια υλικού (hardware) τα οποία συνδέονται σε μία πλατφόρμα edge με σκοπό την συλλογή, μετάδοση, επεξεργασία, αποθήκευση δεδομένων. Για αυτό το σκοπό συνήθως χρησιμοποιούνται SBC's (Single Board Computer), διότι έχουν πολύ μεγαλύτερες επεξεργαστικές δυνατότητες από ότι ένας μικροελεγκτής. Κάποια edge devices ενδεικτικά είναι τα εξής: 1) Raspberry Pi, 2) Nvidia Jetson, 3) Orange Pi, 4) Intel NUC.



Εικόνα 7: Επίπεδα υπολογιστικής λειτουργίας του IIoT

Πηγή : http://digit-project.eu/fileadmin/documents/Workshops/April_2019/GR_Themen-Info_3_Edge_Computing_5G_gre.pdf

Πλατφόρμες Υπολογιστικού Νέφους (Cloud Computing Platforms)

Το υπολογιστικό νέφος είναι μια αρχιτεκτονική που επιτρέπει την on-demand πρόσβαση σε μία κοινόχρηστη υποδομή με υπολογιστικούς πόρους, όπως δίκτυα, διακομιστές, χώρους αποθήκευσης, εφαρμογές και υπηρεσίες, που παρέχονται μέσω του Διαδικτύου. Προσφέρει μια ευέλικτη και επεκτάσιμη υποδομή για αποθήκευση δεδομένων, επεξεργασία και ανάπτυξη εφαρμογών, εξαλείφοντας έτσι την ανάγκη συντήρησης και αναβάθμισης του εξοπλισμού - η οποία είναι αρκετά κοστοβόρα - από τους εκάστοτε οργανισμούς.

Από τα παραπάνω μπορούμε να διακρίνουμε την απαίτηση για cloud-based IoT υποδομές και λύσεις γενικότερα. Το υπολογιστικό νέφος συνδέει IoT συσκευές - οι οποίες συλλέγουν και μεταδίδουν δεδομένα - στους αντίστοιχους cloud-based servers. Οι έξυπνες συσκευές έχουν την δυνατότητα - μέσω αυτών των υποδομών - να παραμετροποιούνται, να ελέγχονται και να παρακολουθούνται σε πραγματικό χρόνο εξ αποστάσεως, καθώς και να ενσωματώνονται/διασυνδέονται με άλλες cloud υπηρεσίες. Ακόμα οι τεράστιες ποσότητες παραγόμενων δεδομένων από τις IoT συσκευές, μπορούν να αποθηκευτούν και να υποβληθούν σε επεξεργασία στο Cloud.

Κάποιες από τις πιο δημοφιλείς πλατφόρμες υπολογιστικού νέφους για IoT οικοσυστήματα είναι οι εξής:

1. Amazon Web Services (AWS) IoT Core [13]: είναι μια λύση που προσφέρει η Amazon. Η υποδομή αυτή είναι μια διαχειριζόμενη υπηρεσία cloud για ασφαλή και κλιμακούμενη σύνδεση και διαχείριση συσκευών IoT. Προσφέρει ακόμα εύκολη διαχείριση των συσκευών αυτών, κατανάλωση (data ingestion) και αποθήκευση δεδομένων, μηχανή κανόνων και ενοποίηση με άλλες υπηρεσίες της AWS.

2. Microsoft Azure IoT Hub [\[14\]](#): Το Azure IoT Hub είναι μια πλήρως διαχειριζόμενη υπηρεσία cloud από τη Microsoft που επιτρέπει αμφίδρομη επικοινωνία και διαχείριση μεταξύ συσκευών IoT και συστημάτων υποστήριξης. Προσφέρει παροχή (provisioning) συσκευών, κατανάλωση δεδομένων τηλεμετρίας και ενσωμάτωση τους με άλλες υπηρεσίες της Azure.
3. Google Cloud IoT Core [\[15\]](#): Το Google Cloud IoT Core είναι μια επεκτάσιμη και ασφαλής cloud-based IoT πλατφόρμα που επιτρέπει τη σύνδεση, τη διαχείριση και την απορρόφηση δεδομένων από συσκευές IoT. Παρέχει εγγραφή συσκευής, διαχείριση κατάστασης συσκευής και ενοποίηση με άλλες υπηρεσίες Google Cloud.
4. Oracle IoT Cloud Service [\[16\]](#): Η υπηρεσία Oracle IoT Cloud παρέχει μια ολοκληρωμένη πλατφόρμα για ασφαλή σύνδεση, διαχείριση και ανάλυση συσκευών και δεδομένων IoT. Περιλαμβάνει υπηρεσίες όπως εγγραφή συσκευής, οπτικοποίηση δεδομένων και ενοποίηση με άλλες υπηρεσίες της Oracle Cloud.

Πρωτόκολλα επικοινωνίας

Τα πρωτόκολλα επικοινωνίας διαδραματίζουν, ίσως, τον πιο κρίσιμο ρόλο στην αποτελεσματική και αξιόπιστη ανταλλαγή δεδομένων μεταξύ των συσκευών IoT. Καθώς ο αριθμός των συνδεδεμένων συσκευών συνεχίζει να αυξάνεται εκθετικά, η ανάγκη για προτυποποιημένα πρωτόκολλα γίνεται όλο και πιο σημαντική. Τα πρωτόκολλα επικοινωνίας IoT ορίζουν τους κανόνες και τη μορφοποίηση (data format) για τη μετάδοση και τη λήψη

δεδομένων, διασφαλίζοντας τη διαλειτουργικότητα, την επεκτασιμότητα και την ασφάλεια στα οικοσυστήματα του IoT. Στο επόμενο κεφάλαιο, θα γίνει μια εκτενής καταγραφή και ανάλυση των πρωτοκόλλων επικοινωνίας τα οποία διευκολύνουν την σύνδεση και την ανταλλαγή δεδομένων μεταξύ των έξυπνων συσκευών.

Μηχανισμοί Ασφαλείας

Οι μηχανισμοί ασφαλείας διαδραματίζουν κρίσιμο ρόλο στη διασφάλιση της ακεραιότητας, της εμπιστευτικότητας και της διαθεσιμότητας των συστημάτων IoT και πολύ περισσότερο των Industrial Internet of Things (IIoT) συστημάτων. Καθώς το IIoT συνεχίζει να επεκτείνεται και να φέρνει επανάσταση στις βιομηχανικές διαδικασίες, η προστασία των συνδεδεμένων συσκευών, δικτύων και δεδομένων γίνεται πρωταρχικής σημασίας. Τα μοναδικά χαρακτηριστικά των περιβαλλόντων IIoT απαιτούν ισχυρά μέτρα ασφαλείας για την προστασία από πιθανές απειλές και ευπάθειες στον κυβερνοχώρο. Κάποια χαρακτηριστικά ασφαλείας για τα IIoT συστήματα είναι τα εξής: 1) ο έλεγχος ταυτότητας, ο οποίος επαληθεύει την ταυτότητα των συσκευών, των χρηστών, και των οντοτήτων εντός του οικοσυστήματος, 2) η κρυπτογράφηση είναι ένας άλλος ουσιαστικός μηχανισμός ασφαλείας που προστατεύει τη μετάδοση και την αποθήκευση δεδομένων, 3) οι μηχανισμοί ελέγχου πρόσβασης διαδραματίζουν και αυτοί έναν ζωτικό ρόλο στη ρύθμιση των αδειών χρηστών και συσκευών στα συστήματα/υποδομές IIoT, 4) τα συστήματα ανίχνευσης και πρόληψης εισβολών είναι ένα από τα πιο κρίσιμα στοιχεία της ασφαλείας των IIoT συστημάτων και τέλος 5) οι ασφαλείς πρακτικές OTA (Over-the-Air update) αναβάθμισης του firmware.

3.4 Περιπτώσεις Χρήσης του IIoT

Το Industrial Internet of Things (IIoT) έχει 'ξεκλειδώσει' μια πληθώρα περιπτώσεων χρήσης, μετασχηματίζοντας κλάδους της βιομηχανίας και αυξάνοντας την αποτελεσματικότητα και την παραγωγικότητα. Σε αυτήν την ενότητα, θα εξετάσουμε ορισμένες σημαντικές περιπτώσεις χρήσης του IIoT και τον πιθανό αντίκτυπό τους σε βιομηχανικές διαδικασίες και συστήματα.

1. *Προγνωστική Συντήρηση (Predictive Maintenance)*: Η προγνωστική συντήρηση αξιοποιεί τα δομικά στοιχεία του IIoT για την παρακολούθηση του εξοπλισμού και τη συλλογή δεδομένων σε πραγματικό χρόνο για την αξιολόγηση της απόδοσης του. Στη συνέχεια, εφαρμόζονται προηγμένοι αλγόριθμοι ανάλυσης των δεδομένων αυτών για τον εντοπισμό ανωμαλιών και την πρόβλεψη πιθανών αστοχιών, επιτρέποντας προληπτικές ενέργειες συντήρησης [\[17\]](#). Αυτή η προσέγγιση βοηθά στη μείωση του χρόνου διακοπής λειτουργίας, στη βελτιστοποίηση των προγραμμάτων συντήρησης (καλύτερος χρονοπρογραμματισμός από του υπεύθυνους συντήρησης αυτών των συστημάτων / μηχανημάτων) και στη βελτίωση της συνολικής αποτελεσματικότητας του εξοπλισμού.
2. *Παρακολούθηση περιουσιακών στοιχείων και διαχείριση αποθέματος (Asset Tracking and Inventory Management)*: Το IIoT επιτρέπει την παρακολούθηση σε πραγματικό χρόνο των περιουσιακών στοιχείων (assets) και του αποθέματος σε όλη την αλυσίδα εφοδιασμού. Με την ανάπτυξη αισθητήρων και ετικετών RFID, οι εταιρείες μπορούν να αποκτήσουν καλύτερη άποψη για την θέση, την κατάσταση και την κίνηση των αγαθών τους. Αυτό ενισχύει την αποτελεσματικότητα, μειώνει τις απώλειες και επιτρέπει την καλύτερη διαχείριση του αποθέματος.
3. *Ποιοτικός Έλεγχος και Βελτιστοποίηση Διαδικασιών (Quality Control and Process Optimization)*: Το IIoT διευκολύνει τη συνεχή παρακολούθηση και τον έλεγχο των διαδικασιών παραγωγής, επιτρέποντας τον ποιοτικό έλεγχο και τη βελτιστοποίηση σε πραγματικό χρόνο. Οι αισθητήρες που είναι ενσωματωμένοι σε μηχανήματα και εξοπλισμό συλλέγουν δεδομένα για διάφορες παραμέτρους διαδικασίας, επιτρέποντας άμεσες προσαρμογές και προληπτικό εντοπισμό ζητημάτων ποιότητας. Αυτό βοηθά στη βελτίωση της ποιότητας των προϊόντων, στη μείωση της σπατάλης και στη βελτιστοποίηση της χρήσης των πόρων.
4. *Διαχείριση και Βελτιστοποίηση Ενέργειας (Energy Management and Optimization)*: Το IIoT παρέχει πληροφορίες σε μοτίβα κατανάλωσης ενέργειας, επιτρέποντας στους οργανισμούς να παρακολουθούν και να βελτιστοποιούν τη χρήση ενέργειας σε πραγματικό χρόνο. Με την ενσωμάτωση συσκευών παρακολούθησης της ενέργειας, οι εταιρείες μπορούν να εντοπίσουν τομείς ενεργειακής αναποτελεσματικότητας, να εφαρμόσουν δράσεις αύξησης της ενεργειακής αποτελεσματικότητας και να μειώσουν το περιβαλλοντικό τους αποτύπωμα.

Οι παραπάνω περιπτώσεις χρήσης καταδεικνύουν τις τεράστιες δυνατότητες του IIoT στις βιομηχανικές λειτουργίες, στη βελτιστοποίηση της χρήσης πόρων, στη μείωση του κόστους, στη βελτίωση των εμπειριών των πελατών και στην προώθηση βιώσιμων πρακτικών. Αξιοποιώντας τη δύναμη του IIoT, οι οργανισμοί μπορούν να ξεκλειδώσουν νέα επίπεδα αποτελεσματικότητας/παραγωγικότητας, να αποκτήσουν χρήσιμες γνώσεις και να επιτύχουν ανταγωνιστικό πλεονέκτημα στο δυναμικό βιομηχανικό τοπίο το οποίο συνεχώς αλλάζει.

3.5 Εφαρμογές του IIoT

Οι εφαρμογές, στο πλαίσιο του IIoT, αναφέρονται στους ευρύτερους τομείς όπου χρησιμοποιούνται τεχνολογίες του IIoT. Οι εφαρμογές IIoT αντιπροσωπεύουν τους διάφορους τομείς όπου η ενοποίηση συνδεδεμένων συσκευών, αισθητήρων, ανάλυσης δεδομένων και αυτοματισμού μπορεί να επιφέρει σημαντικές βελτιώσεις στην αποτελεσματικότητα και την παραγωγικότητα. Παρακάτω θα γίνει σύντομα μια περιγραφή τριών εφαρμογών του IIoT, διερευνώντας έτσι τις επιπτώσεις καθώς και τις δυνατότητες του σε αυτούς τους πολύ σημαντικούς τομείς της βιομηχανίας.

1. *Κατασκευή και Παραγωγή (Manufacturing and Production)* [\[18\]](#): Το IIoT αναδιαμορφώνει τις διαδικασίες παραγωγής, βελτιστοποιεί την αποδοτικότητα και επιτρέπει την πιο έξυπνη λήψη αποφάσεων (decision-making). Τα έξυπνα εργοστάσια (Smart Factories) αξιοποιούν το IIoT για να συνδέσουν μηχανές, αισθητήρες και συστήματα, δημιουργώντας ένα οικοσύστημα όπου τα δεδομένα πραγματικού χρόνου τροφοδοτούν τις κατάλληλες υπηρεσίες ανάλυσης δεδομένων και τεχνητής νοημοσύνης (Data Analytics and AI/ML) με σκοπό την αυτοματοποίηση και τη βελτιστοποίηση των διαδικασιών. Το IIoT επιτρέπει την προγνωστική συντήρηση, τον ποιοτικό έλεγχο σε πραγματικό χρόνο και την παραγωγή βάσει ζήτησης, οδηγώντας σε μειωμένο χρόνο διακοπής λειτουργίας, βελτιωμένη παραγωγικότητα και βελτιωμένη συνολική απόδοση.
2. *Ενέργεια και υπηρεσίες κοινής ωφέλειας (Energy and Utilities)*: ένας ακόμα πολύ σημαντικός κλάδος είναι αυτός της ενέργειας και των υπηρεσιών κοινής ωφέλειας. Το IIoT υπόσχεται να μετασχηματίσει τους παραπάνω τομείς και ιδιαίτερα στο πλαίσιο του έξυπνου δικτύου διανομής (Smart Grid). Το έξυπνο δίκτυο διανομής χρησιμοποιεί το IIoT για τη συλλογή δεδομένων σε πραγματικό χρόνο από αισθητήρες που τοποθετούνται σε όλο το ηλεκτρικό δίκτυο, επιτρέποντας την αποτελεσματική διανομή ενέργειας, εξισορρόπηση φορτίου και απόκριση ζήτησης. Το IIoT διευκολύνει την ενσωμάτωση των ανανεώσιμων πηγών ενέργειας, βελτιώνει την αξιοπιστία του δικτύου και παρέχει στους καταναλωτές πληροφορίες σχετικά με τη χρήση της ενέργειας τους. Αυτές οι εφαρμογές οδηγούν σε βελτιστοποιημένη κατανάλωση ενέργειας και μειωμένο κόστος.
3. *Μεταφορές και Logistics*: Το IIoT φαίνεται να ενισχύει την ελεγχιμότητα, την ασφάλεια και την αποτελεσματικότητα της αλυσίδας εφοδιασμού. Τα συνδεδεμένα οχήματα, τα έξυπνα συστήματα διαχείρισης στόλου και η παρακολούθηση σε πραγματικό χρόνο επιτρέπουν στους οργανισμούς να παρακολουθούν την απόδοση των οχημάτων, να βελτιστοποιούν τις διαδρομές και να βελτιώνουν τις λειτουργίες της εφοδιαστικής αλυσίδας. Οι λύσεις παρακολούθησης περιουσιακών στοιχείων (asset monitoring) που βασίζονται στο IIoT διασφαλίζουν την ακριβή παρακολούθηση της αποστολής, ελαχιστοποιούν τις απώλειες και βελτιώνουν την ευρωστία της εφοδιαστικής αλυσίδας.

3.5.1 Περίπτωση Χρήσης: HVAC (Heating Ventilation & Air Conditioning)

Όπως έχει γίνει είδη ξεκάθαρο, ένας από τους σκοπούς της παρούσας διπλωματικής εργασίας είναι να γίνουν αντιληπτά τα οφέλη που μπορούν να προσφέρουν οι νέες τεχνολογίες του τομέα των Πληροφοριακών Συστημάτων στα HVAC συστήματα. Τα συστήματα αυτά είναι κρίσιμα δομικά στοιχεία σε διάφορους βιομηχανικούς κλάδους - όπως Data Centers, κτιριακές υποδομές(Νοσοκομεία, Ξενοδοχεία), λιανικό εμπόριο, Εφοδιαστική αλυσίδα κτλ - και η εισαγωγή λύσεων που προσφέρει το IIoT θα βοηθήσει στον ψηφιακό μετασχηματισμό (digitalization) τους με σκοπό την βέλτιστη λειτουργία τους και την καλύτερη απόδοση τους. Παρακάτω φαίνεται για κάθε περίπτωση χρήσης του IIoT πως επηρεάζονται τα HVAC συστήματα.

Παρακολούθηση και έλεγχος σε πραγματικό χρόνο (Real-time Monitoring and Control): Το IIoT επιτρέπει την παρακολούθηση και τον έλεγχο των συστημάτων HVAC σε πραγματικό χρόνο, επιτρέποντας στους διαχειριστές των εγκαταστάσεων να συλλέγουν δεδομένα σχετικά με τη θερμοκρασία, την υγρασία, την ποιότητα του αέρα και την κατανάλωση ενέργειας. Αυτά τα δεδομένα σε πραγματικό χρόνο τους δίνουν τη δυνατότητα να λαμβάνουν τεκμηριωμένες αποφάσεις και να αντιμετωπίζουν έγκαιρα τυχόν αποκλίσεις ή ζητήματα που αφορούν την απόδοση του συστήματος.

Προγνωστική συντήρηση και απομακρυσμένα διαγνωστικά (Predictive Maintenance and Remote Control): Το IIoT διευκολύνει την προγνωστική συντήρηση για συστήματα HVAC συλλέγοντας και αναλύοντας δεδομένα από συνδεδεμένους αισθητήρες και συσκευές. Με

την παρακολούθηση διαφόρων παραμέτρων, όπως οι κραδασμοί, η θερμοκρασία και η απόδοση του συστήματος, το IIoT επιτρέπει τον εντοπισμό πιθανών αστοχιών εξοπλισμού ή αναγκών συντήρησης προτού οδηγήσουν σε δαπανηρές βλάβες. Κάποιες από τις πιο δαπανηρές βλάβες αφορούν την διαρροή φρέον και την δυσλειτουργία του συμπιεστή του μηχανήματος.

Ενεργειακή απόδοση και βελτιστοποίηση (Energy Efficiency and Optimization): Το IIoT διαδραματίζει ζωτικό ρόλο στη βελτίωση/βελτιστοποίηση της ενεργειακής απόδοσης των συστημάτων HVAC. Οι συνδεδεμένοι αισθητήρες και ενεργοποιητές παρέχουν πολύτιμες πληροφορίες για τα μοτίβα κατανάλωσης ενέργειας, επιτρέποντας στους διαχειριστές εγκαταστάσεων να εντοπίσουν ευκαιρίες για εξοικονόμηση ενέργειας και να εφαρμόσουν στοχευμένα μέτρα για καλύτερη ενεργειακή απόδοση.

4. Σχεδίαση Εφαρμογής

4.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα γίνει μια εκτενής καταγραφή από διάφορες τεχνολογίες αιχμής σε όλα τα επίπεδα, δηλαδή από μικροελεγκτές με τις κατάλληλες μονάδες ανάπτυξης τους (dev boards) μέσω των οποίων δημιουργήθηκαν οι πρωτότυπες έξυπνες συσκευές, τις βάσεις δεδομένων που χρησιμοποιούνται και συγκεκριμένα Time Series DBs, μέχρι μοτίβα ενδιάμεσου λογισμικού (middleware) τα οποία είναι απαραίτητα για την ανάπτυξη IoT Cloud υποδομών. Με την ανάλυση των τεχνολογιών αυτών έγινε ευκολότερη η επιλογή αυτών που χρησιμοποιήθηκαν για την ανάπτυξη της αρχιτεκτονικής με κριτήριο τις λειτουργικές απαιτήσεις της εφαρμογής.

4.2 IoT Development Boards

Η εμφάνιση του Διαδικτύου των Πραγμάτων (IoT) έχει πυροδοτήσει μια τεράστια ζήτηση για πλακέτες ανάπτυξης. Αυτές χρησιμεύουν ως κρίσιμα εργαλεία για την δημιουργία πρωτοτύπων και χωρίζονται σε πλακέτες ανάπτυξης με μονάδες μικροελεγκτών (MCU) όσο και σε υπολογιστές ενιαίας πλακέτας (SBC). Αυτές οι μονάδες ανάπτυξης παρέχουν μια σταθερή βάση για τη δημιουργία πρωτοτύπων, τη δοκιμή και την ανάπτυξη λύσεων IoT συστημάτων.

4.2.1 Microcontrollers

Οι μονάδες (Development Boards) μικροελεγκτών (MCU) αποτελούν τη ραχοκοκαλιά των περισσότερων εφαρμογών IoT λόγω της χαμηλής κατανάλωσης ενέργειας, της σχέση κόστους-αποτελεσματικότητας και των δυνατοτήτων ενσωμάτωσής (integration capabilities) τους. Οι μονάδες αυτές προσφέρουν πολλά πλεονεκτήματα τα οποία θα εξετάσουμε παρακάτω:

- Ταχεία Πρωτοτυποποίηση και Ανάπτυξη (Rapid Prototyping and Development): Τα συγκεκριμένα development boards παρέχουν ουσιαστικά μια προσβάσιμη πλατφόρμα για γρήγορη δημιουργία πρωτοτύπων. Αυτές οι πλακέτες - οι οποίες είναι εξοπλισμένες με κρίσιμα εξαρτήματα όπως μικροελεγκτή (MCU), αισθητήρες, μονάδες συνδεσιμότητας (connectivity modules) και διεπαφές εισόδου/εξόδου (I/O) - προσφέρουν ένα έτοιμο προς χρήση περιβάλλον για τον χρήστη.

- Υποστήριξη Οικοσυστήματος και Κοινότητα (Ecosystem Support and Community) : Οι μονάδες ανάπτυξης IoT συχνά συνοδεύονται από ένα οικοσύστημα το οποίο προσφέρει εκτεταμένη υποστήριξη, συμπεριλαμβάνοντας τον πολύ σημαντικό παράγοντα της κοινότητας. Οι κατασκευαστές και οι κοινότητες προσφέρουν ολοκληρωμένη τεκμηρίωση, βιβλιοθήκες, παραδείγματα κώδικα και εκπαιδευτικά προγράμματα, διευκολύνοντας τους προγραμματιστές να ξεκινήσουν με τα έργα τους. Το οικοσύστημα υποστήριξης προωθεί τη συνεργασία, την ανταλλαγή γνώσεων και την ικανότητα αξιοποίησης των υπαρχόντων πόρων, επιταχύνοντας τη διαδικασία ανάπτυξης.
- Ενσωμάτωση/Ολοκλήρωση αισθητήρων και συνδεσιμότητας (Sensor Integration and Connectivity): Οι πλακέτες ανάπτυξης IoT έχουν σχεδιαστεί για να ενσωματώνουν 1) διεπαφές αισθητήρων και 2) μια πληθώρα επιλογών συνδεσιμότητας απαραίτητες για τις εφαρμογές του IoT καθώς συνήθως διαθέτουν αναλογικές και ψηφιακές εισόδους, πρωτόκολλα σειριακής επικοινωνίας (όπως UART, SPI και I2C) και ασύρματες δυνατότητες επικοινωνίας (όπως Wi-Fi, Bluetooth ή LoRa). Αυτά τα χαρακτηριστικά επιτρέπουν την απρόσκοπτη ενσωμάτωση διαφόρων αισθητήρων, διευκολύνοντας την απόκτηση δεδομένων και τη συνδεσιμότητα με άλλες συσκευές ή δίκτυα.

IoT Development Board	Microcontroller/Processor	Connectivity	Interfaces	Additional Features
ESP32	Xtensa Dual-Core 32-bit LX6 with 600 DMIPS	Wi-Fi, Ethernet, Bluetooth	SPI, I2C, I2S, UART, ADC, GPIO, PWM, CAN	Dual Core, Low Power, Micropython support
Arduino Uno	ATmega328P	X	SPI, I2C, ADC, GPIO, PWM	Extensive community support
Adafruit Feather MO with LoRa Radio	ATSAMD21G18	LoRa	SPI, I2C, ADC, GPIO, PWM	Small form factor, modular
Particle Photon:Wi-Fi	STM32F205RG Y6 120Mhz ARM Cortex M3	Wi-Fi, Bluetooth	SPI, I2C, I2S, ADC, GPIO, PWM, CAN	Cloud connectivity, RTOS, OTA
Pycom WiPy	ESP32 chipset	Wi-Fi, Bluetooth	SPI, I2C, I2S, ADC, GPIO, PWM, CAN	Micropython support

Πίνακας 1: Χαρακτηριστικά από εμπορικά Development Boards

Στον Πίνακα 1 φαίνονται κάποιες γνωστές μονάδες ανάπτυξης και πρωτοτυποποίησης για IoT, τα χαρακτηριστικά τους καθώς και κάποιες ακόμα πληροφορίες.

4.2.2 Single Board Computers

Οι υπολογιστές ενιαίας πλακέτας (SBC) παρέχουν μεγάλη επεξεργαστική ισχύ και ευελιξία, καθιστώντας τους κατάλληλους για πολύπλοκες εφαρμογές IoT όπως και για edge/fog computing. Τα Single Board Computers προσφέρουν πολλά πλεονεκτήματα τα οποία θα εξετάσουμε παρακάτω:

- **Ευέλικτη και Ισχυρή υπολογιστική ισχύ (Versatile and Powerful Computing):** Τα SBCs παρέχουν πολύ μεγάλη υπολογιστική ισχύ και πόρους, δεδομένης της τιμής τους, του μεγέθους τους και της ενεργειακής τους κατανάλωσης, επιτρέποντας την εκτέλεση εξελιγμένων εφαρμογών IoT, όπως επίσης και την υλοποίηση διαφορετικών αρχιτεκτονικών (π.χ. edge/fog). Συχνά διαθέτουν πολυπύρηνους επεξεργαστές, μεγάλη μνήμη RAM, επιλογές αποθήκευσης και προηγμένες δυνατότητες γραφικών (GPU accelerators κτλ). Αυτά τα χαρακτηριστικά είναι ζωτικής σημασίας για το χειρισμό μεγάλου και σύνθετου όγκου δεδομένων που χρειάζεται επεξεργασία, την εκτέλεση αλγορίθμων τεχνητής νοημοσύνης και την υποστήριξη εφαρμογών πολυμέσων (multimedia).
- **Επεκτασιμότητα και Προσαρμοστικότητα (Expansion and Customization):** Τα SBCs παρέχουν υποδοχές επέκτασης (HATs - Hardware attached on top), επιτρέποντας στους προγραμματιστές να συνδέσουν πρόσθετες μονάδες ή περιφερειακά. Αυτές οι επιλογές επέκτασης επιτρέπουν την προσαρμογή και εξατομίκευση, δίνοντας τη δυνατότητα στους προγραμματιστές να ενσωματώνουν εξειδικευμένες μονάδες υλικού, διεπαφές ειδικές για την κάλυψη συγκεκριμένων απαιτήσεων εφαρμογών IoT.
- **Υποστήριξη λειτουργικού συστήματος:** Οι μονάδες ανάπτυξης IoT που βασίζονται σε SBC προσφέρουν συμβατότητα με ένα ευρύ φάσμα λειτουργικών συστημάτων, συμπεριλαμβανομένων των διανομών Linux, Android και Windows IoT Core. Αυτή η ευελιξία επιτρέπει στους προγραμματιστές να επιλέξουν ένα λειτουργικό σύστημα που ευθυγραμμίζεται με τις απαιτήσεις του έργου τους, να αξιοποιήσει τις υπάρχουσες βιβλιοθήκες λογισμικού και να επωφεληθεί από ένα τεράστιο οικοσύστημα εφαρμογών και εργαλείων ανάπτυξης.

Κάποια θετικά ακόμα είναι η έμφαση που δίνεται στο κόστος αυτών όπως επίσης και μικρό μέγεθος (small footprint) τους. Στο παρακάτω πίνακα φαίνονται κάποιοι από τους πιο δημοφιλείς υπολογιστές ενιαίας πλακέτας.

SBCs	Processor	Memory	Ports	Connectivity	Power
Raspberry Pi 4 Model B	Broadcom BCM2711 SoC featuring quad-core Cortex-A72 64-bit @1.5GHz	1GB, 2GB, 4GB or 8GB RAM	40-pin GPIO connector, 2xmicro- HDMI (up to 4K@60Hz), 2 x USB 3.0, 2 x USB 2.0, 2-lane MIPI DSI display port, 2-lane MIPI CSI camera port, microSD slot, 3.5 mm jack	2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE, Gigabit Ethernet	5V/3A power via USB-C port and GPIO; Power over Ethernet (PoE) capable
Banana Pi M5	Quad-core Amlogic S905X3 with Cortex-A53 @2.0GHz	4GB RAM	40-pin GPIO connector, 1 x HDMI 2.1 (up to 4K@60Hz), 3.5 mm jack, 4 x USB 3.0, microSD slot	Gigabit Ethernet; No onboard Wi-Fi or Bluetooth	5V/3A via USB-C
Orange Pi 5	Rockchip RK3588S octa-core 64-bit processor with quad-core A76+ (2.4GHz)	4GB, 8GB, 16GB or 32GB DDR4 RAM	26-pin GPIO connector, HDMI 2.1 (up to 8K@60Hz), 1 x USB 3.0 Type A, 2 x USB 2.0 Type A, USB-C 3.1 DisplayPort 1.4, M.2 M- KEY socket expansion slot, 3 x MIPI connectors for camera or display, microSD card slot	Gigabit Ethernet; No onboard Wi-Fi or Bluetooth	5V/3A via USB-C

Πίνακας 2: Χαρακτηριστικά δημοφιλών/εμπορικών Single Board Computers

4.3 Τεχνολογίες Ασύρματων Επικοινωνιών

Οι τεχνολογίες ασύρματων επικοινωνιών έχουν αποτελέσει ουσιαστικό καταλύτη στον συνεχώς εξελισσόμενο τομέα της τεχνολογίας, επιτρέποντας στις συσκευές να ανταλλάσσουν δεδομένα χωρίς τη χρήση καλωδίων. Αυτή η καινοτομία έχει φέρει σημαντική επανάσταση στον τρόπο με τον οποίο επικοινωνούμε και έχει ανοίξει το δρόμο για την ταχεία ανάπτυξη του Διαδικτύου των Πραγμάτων (IoT).

Οι τεχνολογίες ασύρματης επικοινωνίας αποτελούν ακρογωνιαίο λίθο των συστημάτων IoT καθώς διευκολύνουν την απρόσκοπτη και πανταχού παρούσα επικοινωνία μεταξύ των συσκευών. Αυτές οι τεχνολογίες κυμαίνονται από τις καθιερωμένες όπως το Wi-Fi, το Bluetooth και το Zigbee έως τις αναδυόμενες όπως το Long Range (LoRa/LoRaWAN), το Narrowband IoT (NB-IoT) και τις ειδικές εκδόσεις του 5G για τις IoT εφαρμογές. Καθεμία από αυτές τις τεχνολογίες προσφέρει ξεχωριστές δυνατότητες όσον αφορά τον ρυθμό δεδομένων, το εύρος επικοινωνίας, την κατανάλωση ενέργειας και την τοπολογία δικτύου, καθιστώντας τις περισσότερο ή λιγότερο κατάλληλες για συγκεκριμένες εφαρμογές IoT.

Η κατανόηση των διαφορετικών τεχνολογιών ασύρματης επικοινωνίας και των αντίστοιχων δυνατών και αδύνατων σημείων τους είναι απαραίτητη για το σχεδιασμό και την ανάπτυξη αποτελεσματικών και αποδοτικών συστημάτων IoT. Η σύγκλιση αυτών των τεχνολογιών στο τοπίο του IoT υπόσχεται να επαναπροσδιορίσει την αλληλεπίδρασή μας με τον φυσικό κόσμο, δημιουργώντας ένα πιο συνδεδεμένο, αποτελεσματικό και έξυπνο οικοσύστημα. Στη συνέχεια θα εξεταστούν κάποια από τα πιο δημοφιλή σε σχέση με την εφαρμοσιμότητα τους πρωτόκολλα ασύρματων επικοινωνιών, ενώ η κατηγοριοποίηση τους θα γίνει με βάση την εμβέλεια.

4.3.1 Μικρής Εμβέλειας

Οι τεχνολογίες μικρής εμβέλειας επιτρέπουν τη συνδεσιμότητα μεταξύ πολλών συσκευών εντός περιορισμένης γεωγραφικής περιοχής. Αυτές οι ασύρματες τεχνολογίες μικρής εμβέλειας, όπως το Bluetooth BLE, το Zigbee και το Wi-Fi, χρησιμεύουν για την τοπική σύνδεση σε διαφορετικά περιβάλλοντα, από την wearable τεχνολογία έως τα συστήματα οικιακού αυτοματισμού. Κάθε μία από αυτές τις τεχνολογίες παρουσιάζει μοναδικά χαρακτηριστικά σχετικά με τους ρυθμούς μεταφοράς δεδομένων, την κατανάλωση ενέργειας, τις τοπολογίες δικτύου και τις λειτουργικές συχνότητες.



Εικόνα 8 : Ασύρματα Πρωτόκολλα επικοινωνίας χαμηλής εμβέλειας

Bluetooth Low Energy

Το Bluetooth χαμηλής ενέργειας—το οποίο είναι επίσης γνωστό ως Bluetooth 4.0 ή Bluetooth Smart—είναι η έκδοση της τεχνολογίας Bluetooth που έχει σχεδιαστεί ειδικά για το IoT. Όπως υποδηλώνει το όνομά του, αυτή είναι μια έκδοση φιλική προς την ισχύ και τους πόρους καθώς έχει σχεδιαστεί για να λειτουργεί σε συσκευές χαμηλής κατανάλωσης που συνήθως λειτουργούν για μικρά χρονικά διαστήματα (sleep mode), είτε συλλέγοντας ενέργεια είτε με τροφοδοσία από μικρή σε μέγεθος μπαταρία. Ένα από τα κύρια πλεονεκτήματα του Bluetooth είναι ότι υπάρχει εδώ και δεκαετίες, έτσι υπάρχουν δισεκατομμύρια συσκευές με δυνατότητα σύνδεσης Bluetooth. Επίσης, το Bluetooth είναι ένα καλά καθιερωμένο και αναγνωρισμένο πρότυπο-πρωτόκολλο ασύρματης επικοινωνίας με πολλούς προμηθευτές οι οποίοι προσφέρουν υποστήριξη και διαλειτουργικότητα, γεγονός που την καθιστά ιδανική τεχνολογία. Η τεχνολογία Bluetooth είναι ιδανική για συσκευές χαμηλής κατανάλωσης όπως κόμβους σε ασύρματα δίκτυα αισθητήρων (WSN) που αφορούν το IIoT και M2M.

Σε αντίθεση με το κλασικό Bluetooth, το οποίο διατηρεί συνεχή σύνδεση, το BLE λειτουργεί χρησιμοποιώντας ένα connectionless μοντέλο το οποίο αναφέρεται συχνά σαν 'broadcast mode'. Οι συσκευές επικοινωνούν χρησιμοποιώντας μικρά πακέτα δεδομένων σε σύντομα χρονικά διαστήματα, ενώ μετά επιστρέφουν σε κατάσταση αναμονής (standby mode), μειώνοντας έτσι σημαντικά τη χρήση ενέργειας. Επιπλέον, το BLE χρησιμοποιεί μια απλοποιημένη διαδικασία σύζευξης, η οποία συμβάλλει επίσης στην αποδοτικότητά του. Ακόμα το BLE υποστηρίζει πολλαπλές τοπολογίες δικτύου, συμπεριλαμβανομένων point-to-point, star και mesh, καθιστώντας το ευέλικτο για διαφορετικές απαιτήσεις εφαρμογών.

ZigBee

Το ZigBee, ένα πρωτόκολλο επικοινωνίας υψηλού επιπέδου (Application/Network Layer), διαδραματίζει κεντρικό ρόλο στην ασύρματη μεταφορά δεδομένων χαμηλής ταχύτητας και μικρής εμβέλειας και αποτελεί βασικό άξονα στον τομέα του Internet of Things (IoT).

Βασισμένο στο πρότυπο IEEE 802.15.4 για ασύρματα προσωπικά δίκτυα (WPAN), το ZigBee φημίζεται για τη χαμηλή κατανάλωση ενέργειας, καθιστώντας το ιδανική επιλογή για συσκευές που λειτουργούν με μπαταρία σε περιορισμένη χωρική εμβέλεια. Το πρωτόκολλο ZigBee λειτουργεί στη ζώνη των 2,4 GHz, 915MHz και 868Mhz, με ταχύτητες δεδομένων 250 kbps. Επιπλέον, παρέχει συμμετρικά κλειδιά κρυπτογράφησης 128-bit για ασφαλή επικοινωνία, αποτρέποντας έτσι μη εξουσιοδοτημένη πρόσβαση και παραβιάσεις δεδομένων. Το δίκτυο μπορεί να φιλοξενήσει σημαντικό αριθμό συσκευών (έως 65.000 κόμβους), γεγονός που δείχνει την επεκτασιμότητα του ZigBee.

Τα εξέχοντα χαρακτηριστικά του ZigBee - χαμηλή κατανάλωση ενέργειας, δικτύωση πλέγματος (mesh networking), ασφάλεια και επεκτασιμότητα - το καθιστούν μια εξαιρετική επιλογή για πληθώρα εφαρμογών. Αυτά κυμαίνονται από τον οικιακό αυτοματισμό και την έξυπνη ενέργεια έως την υγειονομική περίθαλψη, υπογραμμίζοντας τον ευέλικτο ρόλο του ZigBee στο οικοσύστημα IoT. Ως εκ τούτου, η κατανόηση των βασικών χαρακτηριστικών και λειτουργιών του ZigBee είναι πρωταρχικής σημασίας για την αξιοποίηση των δυνατοτήτων του για την ανάπτυξη αποτελεσματικών και ισχυρών λύσεων σε περιβάλλοντα IoT.

WiFi

Το Wi-Fi, έχει μια πανταχού παρουσία σαν τεχνολογία στον τομέα των ασύρματων επικοινωνιών και έχει αναδειχθεί ως θεμελιώδης πυλώνας για την ανάπτυξη εφαρμογών Internet of Things (IoT) σε τοπικά δίκτυα (LAN). Βασισμένο στην οικογένεια προτύπων IEEE 802.11, το Wi-Fi διευκολύνει την ψηφιακή επικοινωνία αξιοποιώντας ραδιοσυχνότητες, συνηθέστερα 2,4 GHz και 5 GHz. Οι υψηλοί ρυθμοί μεταφοράς δεδομένων του Wi-Fi, που συνήθως κυμαίνονται από 54 Mbps έως αρκετά Gbps ανάλογα με το συγκεκριμένο πρότυπο που χρησιμοποιείται (π.χ. 802.11g, 802.11n, 802.11ac ή το πιο πρόσφατο 802.11ax, γνωστό και ως Wi-Fi 6), το καθιστά εξαιρετική επιλογή για εφαρμογές που απαιτούν τη μετάδοση σημαντικών όγκων δεδομένων ή ροή πολυμέσων υψηλής ευκρίνειας.

Επιπλέον, το Wi-Fi υποστηρίζει διάφορες τοπολογίες δικτύου, συμπεριλαμβανομένης της λειτουργίας υποδομής (infrastructure mode) και της λειτουργίας ad hoc. Η λειτουργία υποδομής, η πιο κοινή ρύθμιση, περιλαμβάνει συσκευές (σταθμούς) που επικοινωνούν μέσω ενός κεντρικού σημείου πρόσβασης, συνήθως ενός δρομολογητή Wi-Fi. Η λειτουργία ad hoc επιτρέπει την άμεση επικοινωνία μεταξύ συσκευών χωρίς κεντρικό σημείο πρόσβασης. Αυτές οι δυνατότητες καθιστούν το Wi-Fi εξαιρετικά ευέλικτο για διαφορετικές ρυθμίσεις IoT. Επίσης, ενώ η κατανάλωση ενέργειας του Wi-Fi είναι σχετικά υψηλότερη σε σύγκριση με τεχνολογίες χαμηλής κατανάλωσης όπως το Zigbee ή το Bluetooth Low Energy, οι εξελίξεις στην προτυποποίηση νέων εκδοχών όπως το Wi-Fi HaLow [\[19\]](#) (βασισμένο στο 802.11ah) στοχεύουν στην αντιμετώπιση αυτού του περιορισμού, παρέχοντας Wi-Fi μεγαλύτερης εμβέλειας και χαμηλής κατανάλωσης

6LoWPAN

Η εισαγωγή του 6LoWPAN (IPv6 over Low-Power Wireless Personal Area Networks), το οποίο είναι πρωτόκολλο Layer 2/3 του μοντέλου OSI, αντιπροσωπεύει μια σημαντική πρόοδο στον τομέα των επικοινωνιών IoT. Το 6LoWPAN είναι ένα ανοιχτό πρότυπο που ορίζεται από την Internet Engineering Task Force (IETF) που διευκολύνει τη μετάδοση πακέτων IPv6 μέσω ασύρματων δικτύων χαμηλής κατανάλωσης και χαμηλής ταχύτητας, όπως αυτά που βασίζονται στο πρότυπο IEEE 802.15.4. Ένα καθοριστικό χαρακτηριστικό του 6LoWPAN είναι η ικανότητά του να προσφέρει IP συνδεσιμότητα σε συσκευές περιορισμένων πόρων (resource-constrained devices). Επιτρέπει την απρόσκοπτη ενσωμάτωση συσκευών χαμηλής κατανάλωσης με μεγαλύτερα δίκτυα, διευρύνοντας έτσι το εύρος και την ευελιξία των εφαρμογών IoT. Από τεχνική άποψη, το 6LoWPAN χρησιμοποιεί συμπίεση κεφαλίδας για να ελαχιστοποιήσει το overhead που συνήθως σχετίζεται με την επικοινωνία IPv6. Μειώνει το μέγεθος της κεφαλίδας IPv6 από 40 byte σε μόλις 2 byte, καθιστώντας την πιο κατάλληλη για συσκευές χαμηλής κατανάλωσης με περιορισμένο μέγεθος πακέτου. Όσον αφορά την τοπολογία δικτύου, το 6LoWPAN υποστηρίζει δικτύωση αστέρα και πλέγματος (mesh/star networking topology).

4.3.2 Μεγάλης Εμβέλειας (και χαμηλής ισχύος)

Η έλευση του Διαδικτύου των Πραγμάτων (IoT) έχει προκαλέσει μια αλλαγή σε αρχιτεκτονικές και δομές ασύρματων επικοινωνιών, υπογραμμίζοντας την ανάγκη για τεχνολογίες ικανές να υποστηρίζουν συνδεσιμότητα μεγάλης εμβέλειας. Η διασφάλιση αξιόπιστης μεταφοράς δεδομένων σε εκτεταμένες αποστάσεις είναι ζωτικής σημασίας για την επιτυχή εφαρμογή πολλών εφαρμογών IoT, όπως οι έξυπνες πόλεις, η γεωργία ακριβείας και τα δίκτυα αισθητήρων ευρείας περιοχής. Τεχνολογίες όπως το LoRaWAN, το Sigfox και το Narrowband IoT (NB-IoT), που λειτουργούν υπό την ομπρέλα των δικτύων ευρείας περιοχής χαμηλής ισχύος (LPWAN), έχουν σχεδιαστεί με αυτόν ακριβώς τον στόχο. Αυτές οι τεχνολογίες στοχεύουν να παρέχουν μια ισορροπία μεταξύ της εμβέλειας, της κατανάλωσης ενέργειας και των ρυθμών δεδομένων, ευθυγραμμισμένη με τις απαιτήσεις διαφόρων περιπτώσεων χρήσης IoT.



Εικόνα 9: Ασύρματα Πρωτόκολλα επικοινωνίας υψηλής εμβέλειας

LoRaWAN

Το LoRaWAN (Long Range Wide Area Network) βρίσκεται στην πρώτη γραμμή των τεχνολογιών ασύρματης επικοινωνίας μεγάλης εμβέλειας, παίζοντας έναν καθοριστικό ρόλο για την πρόοδο του Διαδικτύου των Πραγμάτων (IoT). Ανεπτυγμένο από την LoRa Alliance, το LoRaWAN είναι ένα πρωτόκολλο επιπέδου ελέγχου πρόσβασης πολυμέσων (MAC) σχεδιασμένο για δημόσια δίκτυα μεγάλης κλίμακας. Χτισμένο πάνω στο LoRa (Long Range), μια ψηφιακή ασύρματη τεχνολογία επικοινωνίας δεδομένων, το LoRaWAN χαρακτηρίζεται ξεκάθαρα από τις δυνατότητες χαμηλής κατανάλωσης και μεγάλης εμβέλειας, καθιστώντας το κατάλληλο για συσκευές IoT που λειτουργούν με μπαταρία σε ένα δίκτυο ευρείας περιοχής. Λειτουργεί σε διάφορες ζώνες χωρίς την απαίτηση άδειας (unlicensed bands) παγκοσμίως, όπως 868 MHz στην Ευρώπη και 915 MHz στη Βόρεια Αμερική.

Η αρχιτεκτονική επικοινωνίας του LoRaWAN τυπικά διαμορφώνεται σε μια τοπολογία star-of-stars, όπου οι πύλες αναμεταδίδουν μηνύματα μεταξύ των τελικών συσκευών και ενός κεντρικού διακομιστή δικτύου. Αυτή η τοπολογία ενισχύει την επεκτασιμότητα και την ευρωστία του δικτύου. Τα τεχνικά χαρακτηριστικά του LoRaWAN περιλαμβάνουν τον προσαρμοστικό ρυθμό δεδομένων (ADR), όπου ο διακομιστής δικτύου ελέγχει τον ρυθμό δεδομένων μεμονωμένων κόμβων, βελτιστοποιώντας την κατανάλωση ενέργειας και ενισχύοντας τη χωρητικότητα του δικτύου. Επιπλέον, παρέχει αμφίδρομη επικοινωνία με επιβεβαιωμένους και μη επιβεβαιωμένους τύπους μηνυμάτων. Όσον αφορά την ασφάλεια, το LoRaWAN χρησιμοποιεί δύο επίπεδα κρυπτογράφησης: ένα για το ωφέλιμο φορτίο και ένα άλλο για το δίκτυο. Και τα δύο επίπεδα χρησιμοποιούν κρυπτογράφηση AES, παρέχοντας ένα ισχυρό πλαίσιο ασφαλείας για την αποτροπή μη εξουσιοδοτημένης πρόσβασης και παραβιάσεων δεδομένων. Συνοπτικά, το LoRaWAN, με τα χαρακτηριστικά μεγάλης εμβέλειας, χαμηλής κατανάλωσης και την κλιμακούμενη αρχιτεκτονική του, χρησιμεύει ως σημαντικός παράγοντας για εφαρμογές IoT σε διάφορους τομείς, συμπεριλαμβανομένων των έξυπνων πόλεων, της γεωργίας και του βιομηχανικού αυτοματισμού.

Sigfox

Ένα ακόμη βασικό πρωτόκολλο των LPWAN δικτύων, είναι το Sigfox, το οποίο έχει αναδειχθεί ως μια εξέχουσα λύση για εφαρμογές IoT μεγάλης εμβέλειας και χαμηλής κατανάλωσης ενέργειας. Η Sigfox ιδρύθηκε το 2010 και είναι ένας παγκόσμιος πάροχος υπηρεσιών επικοινωνίας που έχει αναπτύξει τη δική του αποκλειστική τεχνολογία για την επικοινωνία των IoT συσκευών με την δικιά του cloud υποδομή. Χαρακτηριζόμενο από την απλότητα και την αποτελεσματικότητά του, το Sigfox έχει σχεδιαστεί με μινιμαλιστική προσέγγιση, όπου ο στόχος είναι να μειωθεί στο ελάχιστο το κόστος και η κατανάλωση ενέργειας των συσκευών. Το Sigfox χρησιμοποιεί μια απλή τοπολογία star δικτύου, όπου οι τελικές συσκευές στέλνουν μηνύματα απευθείας στους σταθμούς βάσης και αυτοί οι σταθμοί βάσης προωθούν τα μηνύματα στο Sigfox Cloud.

Η τεχνολογία του προσφέρει μια δυνατότητα αμφίδρομης επικοινωνίας, αν και εστιάζεται στην επικοινωνία ανοδικής ζεύξης από τις συσκευές στο cloud. Η Sigfox λειτουργεί στις μη αδειοδοτημένες βιομηχανικές, επιστημονικές και ιατρικές ζώνες ραδιοσυστημάτων (ISM – industrial, scientific, medical), οι οποίες διαφέρουν ανά χώρα, όπως 868 MHz στην Ευρώπη και 902 MHz στις Ηνωμένες Πολιτείες. Το δίκτυο χρησιμοποιεί διαμόρφωση Ultra-Narrow Band (UNB) για ραδιοεπικοινωνία, επιτρέποντας εξαιρετική κάλυψη ακόμη και με μετάδοση χαμηλής ισχύος, με τυπική εμβέλεια δεκάδων χιλιομέτρων σε αγροτικές περιοχές και αρκετών χιλιομέτρων σε αστικές περιοχές. Ένα από τα καθοριστικά τεχνικά χαρακτηριστικά του Sigfox είναι το μικρό μέγεθος ωφέλιμου φορτίου, που περιορίζεται στα 12 byte ανά μήνυμα για την ανοδική σύνδεση και 8 byte ανά μήνυμα για την κατερχόμενη ζεύξη, διατηρώντας έτσι την κατανάλωση ενέργειας χαμηλή.

Επιπλέον, για τη βελτιστοποίηση της χωρητικότητας του δικτύου και της διάρκειας ζωής της μπαταρίας της συσκευής, ο αριθμός των μηνυμάτων που μπορεί να στείλει μια συσκευή ανά ημέρα είναι επίσης περιορισμένος. Όσον αφορά την ασφάλεια, το Sigfox χρησιμοποιεί αριθμούς ακολουθίας και έλεγχο ταυτότητας για να αποτρέψει επιθέσεις επανάληψης και να διασφαλίσει την αυθεντικότητα των μηνυμάτων. Συνοπτικά, η Sigfox, με την απλοποιημένη και ενεργειακά αποδοτική προσέγγισή της, είναι ένας σημαντικός διευκολυντής για εφαρμογές IoT, ιδιαίτερα εκείνες που απαιτούν μεγάλη διάρκεια ζωής μπαταρίας και κάλυψη ευρείας περιοχής, όπως έξυπνη μέτρηση, παρακολούθηση του περιβάλλοντος (environmental monitoring) και παρακολούθηση περιουσιακών στοιχείων.

Narrowband IoT (NB-IoT)

Αναπτύχθηκε από το 3rd Generation Partnership Project (3GPP), το Narrowband IoT (NB-IoT) έχει αναδειχθεί ως βασική τεχνολογία IoT στη σφαίρα των δικτύων κινητής τηλεφωνίας ενισχύοντας σημαντικά την εμβέλεια και την αποτελεσματικότητα για την ανάπτυξη λύσεων σε περιβάλλοντα Internet of Things (IoT). Ως τεχνολογία Low Power Wide Area Network (LPWAN), το NB-IoT είναι ειδικά προσαρμοσμένο για συσκευές IoT που απαιτούν μεγάλη διάρκεια ζωής μπαταρίας, οικονομικά αποδοτική συνδεσιμότητα και βαθιά διείσδυση σε εσωτερικούς χώρους. Το NB-IoT λειτουργεί σε μια στενή ζώνη συχνοτήτων 200 kHz, η οποία, όπως υποδηλώνει το όνομα, είναι στενότερη από αυτή που χρησιμοποιούν οι περισσότερες τεχνολογίες κινητής τηλεφωνίας. Όσον αφορά την κάλυψη, το NB-IoT χρησιμοποιεί επανάληψη μεταδόσεων (relays) για να βελτιώσει το link budget, προσφέροντας εξαιρετική κάλυψη σε δύσκολα περιβάλλοντα, όπως βαθιά σε εσωτερικούς χώρους ή σε υπόγειες τοποθεσίες. Αυτό καθιστά το NB-IoT ιδανικό για εφαρμογές όπως η έξυπνη μέτρηση και η παρακολούθηση του περιβάλλοντος (environmental monitoring).

Από άποψη ασφάλειας, το NB-IoT επωφελείται από τους αποδεδειγμένους μηχανισμούς ασφαλείας των δικτύων LTE, συμπεριλαμβανομένου του ελέγχου ταυτότητας συσκευής, της ακεραιότητας δεδομένων και της κρυπτογράφησης. Συνοπτικά, το NB-IoT, με τη μεγάλη διάρκεια ζωής της μπαταρίας, τη βαθιά κάλυψη και την ασφαλή και αποτελεσματική συνδεσιμότητα, διαδραματίζει κρίσιμο ρόλο στην ενεργοποίηση εφαρμογών IoT σε δίκτυα κινητής τηλεφωνίας. Η πλήρης κατανόηση των τεχνικών χαρακτηριστικών και των λειτουργικών αρχών του είναι θεμελιώδης για τους ερευνητές και τους επαγγελματίες που εργάζονται στο τοπίο του IoT.

4.4 Μοτίβα Ενδιάμεσου Λογισμικού (Pub/Sub)

Η ανάπτυξη του Διαδικτύου των Πραγμάτων (IoT) που είχε ως αποτέλεσμα μια χωρίς προηγούμενο ενοποίηση συσκευών και συστημάτων, που είχε ως αποτέλεσμα τα ετερογενή αυτά οικοσυστήματα να χρειάζονται αποτελεσματικές μεθοδολογίες επικοινωνίας. Το Middleware, λειτουργεί ως γεφύρωμα μεταξύ αυτών των διαφορετικών στοιχείων. Στο πλαίσιο του ενδιάμεσου λογισμικού, τα πρότυπα επικοινωνίας διαδραματίζουν κεντρικό ρόλο στην ενορχήστρωση της απρόσκοπτης ανταλλαγής πληροφοριών σε όλο το δίκτυο IoT. Ένα τέτοιο μοτίβο με επιρροή είναι το μοντέλο Publish/Subscribe (Pub/Sub), το οποίο εφαρμόζεται ευρέως σε πολλά πρωτόκολλα. Το μοτίβο Pub/Sub αναφέρεται σε τέσσερα κυρίαρχα πρωτόκολλα ενδιάμεσου λογισμικού – MQTT, XMPP, AMQP και DDS – και δίνει έμφαση στην εφαρμογή τους που είναι προσανατολισμένη σε IoT περιβάλλοντα.

Το μοντέλο Publish/Subscribe (Pub/Sub) είναι ένα ξεχωριστό και ευρέως διαδεδομένο μοτίβο ενδιάμεσου λογισμικού που αντιμετωπίζει την ανάγκη για ασύγχρονη, αποσυνδεδεμένη (decoupled) επικοινωνία σε κατανεμημένα συστήματα. Η ουσία του μοντέλου Pub/Sub έγκειται στην απλότητα και την αποτελεσματικότητά του: ενορχηστρώνει την επικοινωνία μεταξύ αποστολέων (publishers) και δεκτών (subscribers) μηνυμάτων, χωρίς να απαιτεί από αυτούς να αλληλεπιδρούν άμεσα μεταξύ τους. Το μοντέλο Pub/Sub βασίζεται εγγενώς σε γεγονότα (event-driven): οι αποστολείς διαδίδουν μηνύματα, κατηγοριοποιημένα σε topics, χωρίς να γνωρίζουν κανένα συνδρομητή. Αντίθετα, οι δέκτες εκδηλώνουν ενδιαφέρον για μία ή περισσότερες κατηγορίες μηνυμάτων, λαμβάνοντας αντίστοιχα μηνύματα χωρίς να γνωρίζουν τους αποστολείς. Ένα κεντρικό στοιχείο, που αναφέρεται ως διαμεσολαβητής (broker), επιτρέπει την επικύρωση μηνυμάτων, την επικοινωνία και την δρομολόγηση μεταξύ αποστολέων και δεκτών.

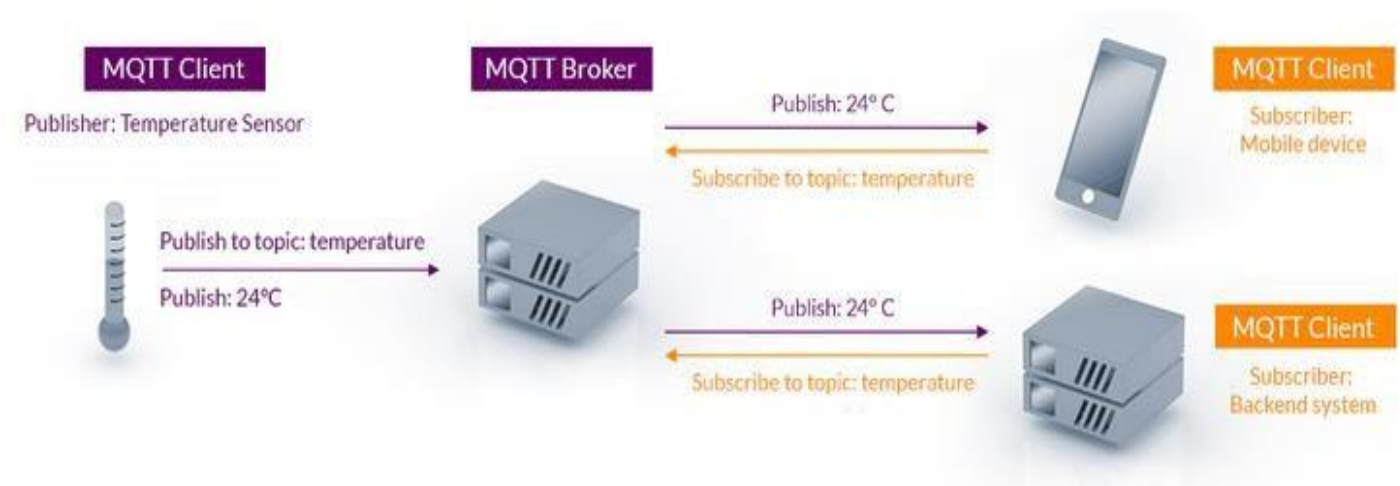
Στο πλαίσιο του Internet of Things (IoT), το μοντέλο Pub/Sub αποδεικνύεται ιδιαίτερα ωφέλιμο λόγω της επεκτασιμότητας και της ανοχής του σε σφάλματα. Η ασύγχρονη φύση του μοντέλου επιτρέπει την αντιμετώπιση των προβλημάτων της διακοπτόμενης συνδεσιμότητας, κυμαινόμενη καθυστέρηση μεταφοράς και περιορισμούς ισχύος, που είναι χαρακτηριστικές προκλήσεις στα οικοσυστήματα IoT. Επιπλέον, επιτρέπει σε μεγάλο αριθμό συσκευών IoT να επικοινωνούν αποτελεσματικά, επιτρέποντας την επεκτασιμότητα (scalability).

4.4.1 MQTT - Message Queuing Telemetry Transport

Το πρωτόκολλο MQTT αναπτύχθηκε το 1999 από τον Andy Stanford-Clark της IBM και τον Arlen Nipper της Arcom και έχει καθιερωθεί από τότε ως πρωτόκολλο βιομηχανικού προτύπου στη σφαίρα του Διαδικτύου των Πραγμάτων (IoT) λόγω των “ελαφρών” και αποτελεσματικών χαρακτηριστικών του. Παρέχει μια ισχυρή λύση για την παράδοση μηνυμάτων σε αναξιόπιστα ή υψηλής καθυστέρησης δίκτυα, που αποτελούν συνθήκες στα IoT περιβάλλοντα. Στον πυρήνα του, το MQTT λειτουργεί χρησιμοποιώντας το μοτίβο Pub/Sub μέσω του πρωτοκόλλου TCP/IP, διασφαλίζοντας αξιόπιστη μεταφορά δεδομένων.

Το μοντέλο βασίζεται σε έναν κεντρικό broker για τη διαχείριση της διανομής μηνυμάτων, αφαιρώντας την πολυπλοκότητα της άμεσης επικοινωνίας από συσκευή σε συσκευή. Αυτή η αποσύνδεση του αποστολέα (publisher) και του παραλήπτη (subscriber) συμβάλλει στην επεκτασιμότητα του πρωτοκόλλου και στην ανοχή σφαλμάτων.

MQTT Publish / Subscribe Architecture



Εικόνα 10: Παράδειγμα της αρχιτεκτονικής του πρωτοκόλλου MQTT

Πηγή: <https://theautomationblog.com/how-to-setup-an-mqtt-system/>

Κάποια τεχνικά χαρακτηριστικά του MQTT για τα οποία είναι δημοφιλές είναι τα ακόλουθα:

- Επίπεδα ποιότητας υπηρεσίας (QoS): Το MQTT προσφέρει τρία επίπεδα QoS για να ισορροπήσει στο ισοζύγιο μεταξύ της διασφάλισης παράδοσης μηνυμάτων και τη χρήση των πόρων του δικτύου. Το QoS 0 (το πολύ μία φορά) παρέχει την ταχύτερη παράδοση χωρίς επιβεβαίωση, το QoS 1 (τουλάχιστον μία φορά) εγγυάται την παράδοση αλλά μπορεί να οδηγήσει σε διπλότυπα και το QoS 2 το οποίο προσφέρει το υψηλότερο επίπεδο υπηρεσιών στο MQTT, διασφαλίζει ότι κάθε μήνυμα παραδίδεται ακριβώς μια φορά στους προβλεπόμενους παραλήπτες.
- Minimal Protocol Overhead: Το πρωτόκολλο MQTT έχει σχεδιαστεί με ελάχιστο αποτύπωμα, με σταθερό μέγεθος κεφαλίδας των δύο byte. Αυτός ο ελαφρύς σχεδιασμός βελτιστοποιεί τη χρήση του δικτύου και την κατανάλωση πόρων της συσκευής, καθιστώντας το MQTT μια εξαιρετική επιλογή για συσκευές IoT με περιορισμένους πόρους.
- Last Will and Testament (LWT): Σε περίπτωση απροσδόκητης αποσύνδεσης, ο MQTT broker μπορεί να δημοσιεύσει ένα προκαθορισμένο μήνυμα LWT εκ μέρους του αποσυνδεδεμένου publisher, παρέχοντας έτσι έναν μηχανισμό για άλλες συσκευές στο δίκτυο να αναγνωρίζουν και να ανταποκρίνονται σε βλάβες της συσκευής.

- **Security Enabled:** Το MQTT διευκολύνει την κρυπτογράφηση μηνυμάτων χρησιμοποιώντας TLS (Transport Layer Security) - το οποίο είναι ένα πρωτόκολλο κρυπτογράφησης που παρέχει ασφάλεια στην επικοινωνία, προσφέροντας ιδιωτικότητα και ακεραιότητα των δεδομένων - και τον έλεγχο ταυτότητας πελατών χρησιμοποιώντας σύγχρονα πρωτόκολλα ελέγχου ταυτότητας, όπως το OAuth.

4.4.2 XMPP - eXtensible Messaging and Presence Protocol

Το Extensible Messaging and Presence Protocol (XMPP), αρχικά γνωστό ως Jabber, ιδρύθηκε από την κοινότητα ανοιχτού κώδικα Jabber το 1999 και αργότερα επισημοποιήθηκε από την ομάδα Internet Engineering Task Force (IETF) το 2004. Το XMPP είναι ένα σύνολο ανοιχτών τεχνολογιών για ανταλλαγή άμεσων μηνυμάτων, συνομιλία πολλών μερών, φωνητικές κλήσεις και βιντεοκλήσεις, συνεργασία, ελαφρύ ενδιάμεσο λογισμικό, διανομή περιεχομένου και γενικευμένη δρομολόγηση δεδομένων XML. Το XMPP είναι ένα αποκεντρωμένο πρωτόκολλο, που σημαίνει ότι δεν απαιτείται κεντρικός διακομιστής για την επικοινωνία μεταξύ των πελατών. Αντίθετα, ένα δίκτυο XMPP αποτελείται από ανεξάρτητους διακομιστές που είναι διασυνδεδεμένοι σε ένα συνενωμένο/συσσωματωμένο (federated) μοντέλο.

Τα βασικά τεχνικά χαρακτηριστικά του XMPP περιλαμβάνουν: 1) *Αποκεντρωμένη Αρχιτεκτονική*, εξαλείφοντας την ανάγκη για κεντρικό διακομιστή και μειώνοντας έτσι τον κίνδυνο ενός μόνο σημείου αστοχίας (single point of failure), 2) *Ασφάλεια*, από προεπιλογή, το XMPP υποστηρίζει Transport Layer Security (TLS) και Simple Authentication and Security Layer (SASL), προσφέροντας ασφαλή έλεγχο ταυτότητας και κρυπτογράφηση δεδομένων, 3) *Επεκτασιμότητα*, το XMPP έχει σχεδιαστεί για να είναι επεκτάσιμο και υποστηρίζει ένα ευρύ φάσμα επεκτάσεων, επιτρέποντας μια ευέλικτη προσαρμογή σε διαφορετικές απαιτήσεις εφαρμογών, 4) *Πληροφορίες παρουσίας*, εκτός από την ανταλλαγή μηνυμάτων, το XMPP επιτρέπει στους χρήστες να μεταδίδουν την παρουσία τους (κατάσταση στο διαδίκτυο) και να λαμβάνουν ενημερώσεις σχετικά με την παρουσία άλλων χρηστών.

4.4.3 AMQP - Advanced Message Queuing Protocol

Το Advanced Message Queuing Protocol (AMQP) είναι ένα ανοιχτό πρωτόκολλο επιπέδου εφαρμογής (application layer) για ενδιάμεσο λογισμικό προσανατολισμένο σε μηνύματα, που ορίστηκε για πρώτη φορά το 2004. Τα καθοριστικά χαρακτηριστικά του AMQP είναι ο προσανατολισμός μηνυμάτων (message orientation), οι ουρές αναμονής/εξυπηρέτησης, η δρομολόγηση (συμπεριλαμβανομένου σημείου σε σημείο και δημοσίευσης-και-εγγραφή), η αξιοπιστία και η ασφάλεια. Το AMQP παρέχει ένα ισχυρό πλαίσιο για ευέλικτη επικοινωνία μεταξύ εφαρμογών και συσκευών, καθιστώντας το μια στρατηγική επιλογή για λύσεις στο Internet of Things (IoT), όπου διαφορετικές εφαρμογές και συσκευές συχνά χρειάζονται να επικοινωνούν αποτελεσματικά.

Τα βασικά τεχνικά χαρακτηριστικά του AMQP περιλαμβάνουν: 1) *Προσανατολισμός μηνυμάτων*: σημαίνει ότι η επικοινωνία βασίζεται στην αποστολή και λήψη διακριτών μηνυμάτων παρά στη δημιουργία μιας συνεχούς ροής δεδομένων. Αυτό επιτρέπει μια πιο ευέλικτη και αξιόπιστη ανταλλαγή δεδομένων μεταξύ κατανεμημένων συστημάτων (event-driver distributed architecture), 2) *Ουρά (Queueing)*: Το AMQP παρέχει δυνατότητα ουρών αναμονής που διασφαλίζουν ότι τα μηνύματα αποθηκεύονται και προωθούνται σωστά, διασφαλίζοντας αξιόπιστη παράδοση μηνυμάτων ακόμη και σε περίπτωση αστοχίας του συστήματος, 3) *Δρομολόγηση*: επιτρέπει ευέλικτα σχήματα δρομολόγησης μηνυμάτων, συμπεριλαμβανομένων μοντέλων όπως point-to-point και publish-subscribe, 4) *Ασφάλεια*: το AMQP παρέχει ασφαλή παράδοση μηνυμάτων με υποστήριξη TLS και έλεγχο ταυτότητας (SASL) για κρυπτογραφημένες συνδέσεις και έλεγχο ταυτότητας, 5) *Διαλειτουργικότητα*: δεδομένου ότι το AMQP είναι πρωτόκολλο ενσύρματου επιπέδου (wire-level protocol), επιτρέπει τη διαλειτουργικότητα μεταξύ client και broker ανεξάρτητα από την πλατφόρμα ή τη γλώσσα. Αυτή η λειτουργικότητα του wire-level protocol επιτρέπει σε πολλαπλές εφαρμογές μέσα σε ένα δίκτυο να λειτουργούν απρόσκοπτα.

4.5 Έννοιες Σχεδίασης Λογισμικού

Στο δυναμικό τοπίο του Software Engineering, οι διεπαφές προγραμματισμού εφαρμογών (API) αποτελούν ίσως το σημαντικότερο στοιχείο για την αλληλεπίδραση μεταξύ διαφορετικών οντοτήτων λογισμικού. Το αναπτυσσόμενο πεδίο του Διαδικτύου των Πραγμάτων προσθέτει ένα νέο επίπεδο πολυπλοκότητας στις αρχές σχεδιασμού λογισμικού λόγω των μοναδικών χαρακτηριστικών και απαιτήσεων των εφαρμογών σε αυτό.

Τα API διαδραματίζουν κεντρικό ρόλο στα οικοσυστήματα IoT διευκολύνοντας την επικοινωνία μεταξύ διαφορετικών συσκευών και υπηρεσιών και ορίζουν πώς πρέπει να αλληλεπιδρούν τα στοιχεία λογισμικού, παρέχοντας έναν δομημένο τρόπο πρόσβασης των εφαρμογών σε υπηρεσίες, αισθητήρες, ενεργοποιητές ή πόρους σε ένα σύστημα IoT.

4.5.1 API (Application Programming Interface)

Η διεπαφή προγραμματισμού εφαρμογών (API), επιτρέπει τη σύνδεση μεταξύ υπολογιστών ή προγραμμάτων υπολογιστή. Είναι μια διεπαφή λογισμικού που προσφέρει υπηρεσίες σε άλλο λογισμικό για τη βελτίωση των απαιτούμενων λειτουργιών. Τα τελευταία χρόνια, το API έχει κερδίσει δημοτικότητα στην αγορά για σχεδόν κάθε εφαρμογή Ιστού. Το API δίνει τη δυνατότητα στις επιχειρήσεις να ανοίγουν τα δεδομένα και να παρέχουν διαλειτουργικότητα στις εφαρμογές τους προς εξωτερικούς προγραμματιστές (third-party developers) με σκοπό την ανάπτυξη νέων εφαρμογών και την επέκταση υπαρχουσών.

4.5.2 Τύποι API

Οι διεπαφές προγραμματισμού εφαρμογών (APIs) ταξινομούνται σε τρεις τύπους με βάση τους σκοπούς και τις εφαρμογές τους:

Remote API

Τα απομακρυσμένα API έχουν σχεδιαστεί για να αλληλεπιδρούν μέσω ενός δικτύου επικοινωνιών. Με τον όρο απομακρυσμένη, εννοούμε ότι οι πόροι που χειρίζονται το API βρίσκονται κάπου “έξω” από τον υπολογιστή που υποβάλλει το αίτημα. Επειδή το πιο ευρέως χρησιμοποιούμενο δίκτυο επικοινωνιών είναι το διαδίκτυο, τα περισσότερα API έχουν σχεδιαστεί με βάση τα πρότυπα ιστού. Ο πιο διαδεδομένος τύπος API ο οποίος χρησιμοποιείται στην εφαρμογές ιστού είναι το Web API και αναλύεται παρακάτω.

Web API

Το Web API είναι μια διεπαφή ανοιχτού κώδικα η οποία μπορεί να προσπελαστεί εύκολα χρησιμοποιώντας το πρωτόκολλο HTTP. Μπορεί να αξιοποιηθεί από έναν μεγάλο αριθμό συσκευών όπως: Smartphones, Tablets ή Laptops. Ένα Web API μπορεί να αναπτυχθεί χρησιμοποιώντας διάφορες τεχνολογίες όπως Java και ASP.NET, που παρέχουν ανώτερη απόδοση και ταχύτερη ανάπτυξη υπηρεσιών. Ωστόσο, καθώς τα Web API έχουν σχεδιαστεί για κατανεμημένες υπηρεσίες, είναι ελαφριά και έχουν περιορισμούς στο εύρος ζώνης.

Local API

Είναι API που κάνει διαθέσιμες υπηρεσίες του λειτουργικού (OS) σε εφαρμογές στο πεδίο χρήστη. Για παράδειγμα, τα API .NET της Microsoft, το TAPI (API τηλεφωνίας) και τα API πρόσβασης στη βάση δεδομένων είναι μορφές τοπικών API.

4.5.3 Πρωτόκολλα και Αρχιτεκτονική API

Τα πρωτόκολλα και οι αρχιτεκτονικές API αποτελούν τη βάση του τρόπου με τον οποίο δομούνται και επικοινωνούν τα APIs. Με βάση τον σκοπό, ένα API ακολουθεί μια σειρά από πρωτόκολλα και πρότυπα. Παρακάτω είναι μερικά βασικά πρωτόκολλα και η σημαντικότερη/δημοφιλέστερη αρχιτεκτονική API:

HTTP/HTTPS: είναι τα βασικά πρωτόκολλα που χρησιμοποιούνται από τα Web API. Παρέχουν ένα σύνολο μεθόδων (όπως GET, POST, PUT, DELETE) που καθορίζουν τον τύπο της λειτουργίας ή της ενέργειας που πρέπει να εκτελεστεί.

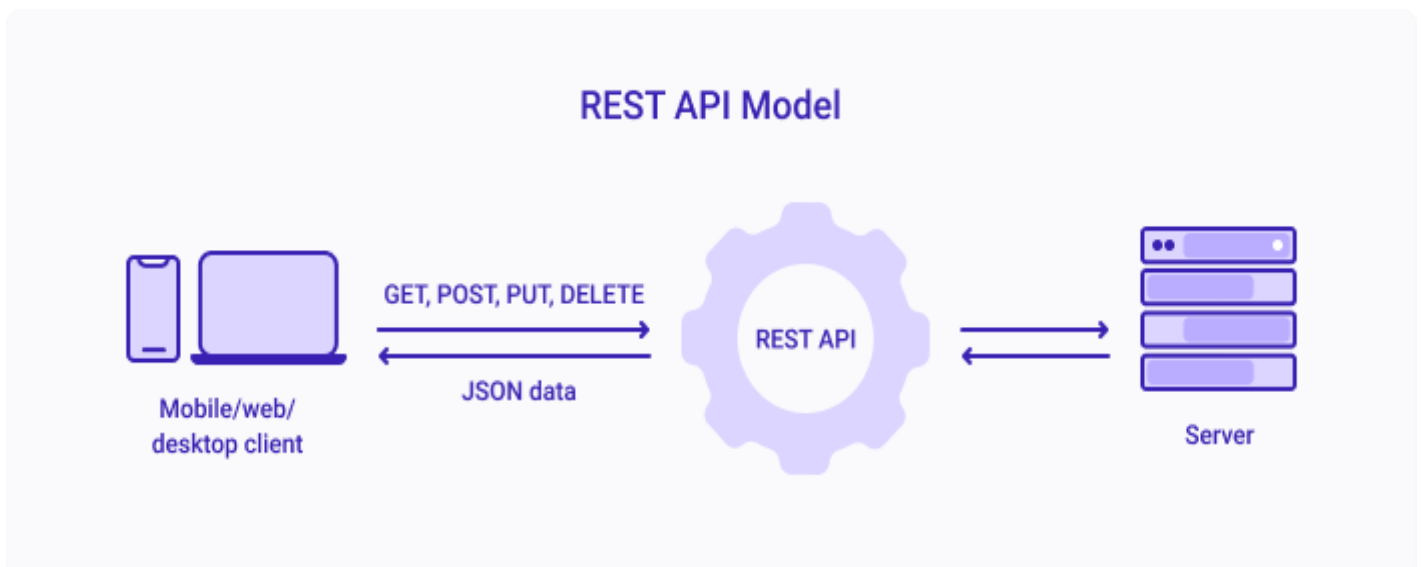
SOAP (Simple Object Access Protocol): Το SOAP είναι μια προδιαγραφή πρωτοκόλλου ανταλλαγής μηνυμάτων για την ανταλλαγή δομημένων πληροφοριών κατά την υλοποίηση υπηρεσιών Ιστού με χρήση XML. Μπορεί να λειτουργήσει 'πάνω' από διάφορα πρωτόκολλα όπως HTTP, SMTP κ.λπ. Τα API SOAP είναι εξαιρετικά επεκτάσιμα, ευέλικτα και υποστηρίζουν ισχυρά μέτρα ασφαλείας, καθιστώντας τα κατάλληλα για εφαρμογές κατασκευασμένες να λειτουργούν σε εταιρικό περιβάλλον (από επιχειρήσεις, κυβερνητικούς φορείς, κλπ).

JSON-RPC και XML-RPC: Πρόκειται για πρωτόκολλα remote procedure call (RPC) που κωδικοποιούνται σε JSON ή XML. Είναι πιο απλά από το SOAP αλλά προσφέρουν και λιγότερες δυνατότητες.

REST (Representational State Transfer): Το REST είναι μια αρχιτεκτονική για δικτυακές εφαρμογές μέσω HTTP/HTTPS. Τα REST API (ή τα RESTful API) χρησιμοποιούν τυπικές μεθόδους HTTP.

4.5.4 REST API

Το REST API είναι ένα API που ακολουθεί ένα σύνολο κανόνων για την επικοινωνία μιας εφαρμογής και υπηρεσιών μεταξύ τους. Το REST API αναφέρεται ως RESTful API και έχει αναδειχθεί ως το de facto πρότυπο για το σχεδιασμό web-based APIs λόγω της απλότητας, της αποτελεσματικότητας και της συμβατότητας του με το πρωτόκολλο HTTP. Παρά ορισμένους περιορισμούς, η αρχιτεκτονική REST προσφέρει σημαντικά πλεονεκτήματα και αναμένεται να συνεχίσει να διαδραματίζει ζωτικό ρόλο στη σχεδίαση και την ανάπτυξη διαδικτυακών υπηρεσιών.



Εικόνα 11: Μοντέλο επικοινωνίας REST API

Πηγή: <https://apidocs.tegsoft.com/>

Κάποιες από τις βασικές αρχές που τηρεί το REST είναι οι ακόλουθες:

- **Client-Server Architecture** : τα RESTful API διατηρούν έναν σαφή διαχωρισμό μεταξύ του πελάτη και του διακομιστή, βελτιώνοντας τη φορητότητα της διεπαφής πελάτη- χρήστη σε πολλές πλατφόρμες και ενισχύοντας την επεκτασιμότητα των στοιχείων του διακομιστή.
- **Stateless Interactions** : Κάθε αίτημα από έναν πελάτη προς έναν διακομιστή πρέπει να περιέχει όλες τις πληροφορίες που απαιτούνται για την κατανόηση και την επεξεργασία του αιτήματος. Αυτό διασφαλίζει ότι ο διακομιστής δεν χρειάζεται να διατηρεί κάποια συγκεκριμένη κατάσταση μεταξύ των αιτημάτων.
- **Cacheable Data**: Οι αποκρίσεις (responses) από τον διακομιστή μπορούν να αποθηκευτούν προσωρινά, επιτρέποντας στους πελάτες να επαναχρησιμοποιήσουν προηγούμενες αποκρίσεις για να βελτιώσουν την αποτελεσματικότητα και να μειώσουν το φόρτο του διακομιστή.
- **Uniform Interface**: Τα API REST έχουν μια συνεπή και τυποποιημένη δομή, η οποία απλοποιεί την αρχιτεκτονική και ενισχύει την ορατότητα (visibility) και τη φορητότητα.

Κεντρική θέση στο σχεδιασμό ενός RESTful API είναι η έννοια των πόρων, δηλαδή συγκεκριμένες πηγές πληροφοριών που μπορούν να αναγνωριστούν από μοναδικές διευθύνσεις URL. Οι τυπικές μέθοδοι HTTP (GET, POST, PUT, DELETE) χρησιμοποιούνται για τον χειρισμό αυτών των πόρων. Για παράδειγμα, ένα αίτημα GET χρησιμοποιείται για την ανάκτηση ενός πόρου, ένα αίτημα POST για τη δημιουργία ενός νέου πόρου, ένα αίτημα PUT για την ενημέρωση ενός υπάρχοντος πόρου και ένα αίτημα DELETE για την κατάργηση ενός πόρου.

4.6 Βάσεις Δεδομένων

Το IoT, που χαρακτηρίζεται από ένα δίκτυο διασυνδεδεμένων φυσικών αντικειμένων - έξυπνων συσκευών, έχει αναδιαμορφώσει τον τρόπο με τον οποίο παράγονται, συλλέγονται και αναλύονται τα δεδομένα. Οι συσκευές IoT παράγουν συνεχώς δεδομένα, αυξάνοντας την ανάγκη για βάσεις δεδομένων που μπορούν αποτελεσματικά να απορροφούν, να αποθηκεύουν και να αναλύουν τεράστιες ποσότητες χρονοσειρών. Αυτά τα συστήματα διαχείρισης δεδομένων κυμαίνονται από παραδοσιακές βάσεις δεδομένων SQL έως πιο πρόσφατες βάσεις δεδομένων NoSQL και χρονοσειρών, καθεμία από τις οποίες ταιριάζει σε συγκεκριμένες ανάγκες του οικοσυστήματος IoT.

4.6.1 Σχεσιακές Βάσεις Δεδομένων

Οι βάσεις δεδομένων SQL, οι οποίες αναφέρονται και ως σχεσιακές βάσεις δεδομένων, έχουν καθιερωθεί και χρησιμοποιούνται ευρέως. Οι βάσεις αυτές χρησιμοποιούν μια δομημένη προσέγγιση στη διαχείριση δεδομένων, καθιστώντας τις κατάλληλες για εφαρμογές που απαιτούν πολύπλοκα ερωτήματα και συναλλαγές.

Στο τοπίο του IoT, οι εξέχουσες DBMS (database management systems) για σχεσιακές βάσεις δεδομένων περιλαμβάνουν τις MySQL και PostgreSQL.

MySQL: είναι μια από τις πιο δημοφιλείς open-source βάσεις δεδομένων για την αξιοπιστία της, καθώς επίσης και μια πολύ καλή επιλογή για εφαρμογές IoT που απαιτούν συνέπεια και ακεραιότητα δεδομένων. Υποστηρίζει πολύπλοκα ερωτήματα, καθιστώντας την χρήσιμη σε εφαρμογές όπως η διαχείριση της εφοδιαστικής αλυσίδας που βασίζεται στο IoT ή τα συστήματα έξυπνων κατοικιών.

PostgreSQL: Προσφέροντας προηγμένες δυνατότητες, όπως τύπους που καθορίζονται από το χρήστη, η PostgreSQL είναι μια άλλη ευρέως χρησιμοποιούμενη βάση δεδομένων SQL. Υπερέχει σε περιπτώσεις όπου οι εφαρμογές IoT πρέπει να χειρίζονται σύνθετους τύπους δεδομένων ή να χρησιμοποιούν γεωχωρικά δεδομένα, όπως υπηρεσίες που βασίζονται σε τοποθεσία.

4.6.2 Μη-Σχεσιακές Βάσεις Δεδομένων (NoSQL)

Οι βάσεις δεδομένων NoSQL (όχι μόνο SQL) εμφανίστηκαν για να αντιμετωπίσουν τους περιορισμούς των παραδοσιακών βάσεων δεδομένων SQL, ειδικά στην αντιμετώπιση άκαμπτων σχημάτων - εισάγοντας δηλαδή πιο ευέλικτα μοντέλα δεδομένων - και περιορισμών επεκτασιμότητας. Μεταξύ των διαφορετικών τύπων βάσεων δεδομένων NoSQL, οι βάσεις δεδομένων εγγράφων (document based) και key-value προσφέρουν ευελιξία, οριζόντια επεκτασιμότητα και γρήγορες λειτουργίες για την πρόσβαση στα αποθηκευμένα δεδομένα. Αυτά τα χαρακτηριστικά ευθυγραμμίζονται με τις απαιτήσεις διαχείρισης των υψηλού όγκου, ετερογενών δεδομένων που είναι τυπικά σε σενάρια IoT. Αξίζει να αναφερθούν η MongoDB, η Redis και η Amazon DynamoDB.

MongoDB: είναι μια από τις πιο γνωστές βάσεις δεδομένων εγγράφων NoSQL και είναι γνωστή για το ευέλικτο μοντέλο δεδομένων της. Σε αντίθεση με τις βάσεις δεδομένων SQL, οι οποίες απαιτούν ένα προκαθορισμένο σχήμα, η MongoDB υποστηρίζει ένα δυναμικό σχήμα, επιτρέποντας την αποθήκευση και τη διαχείριση διαφορετικών δομών δεδομένων. Επιπλέον, η MongoDB παρέχει ισχυρή οριζόντια επεκτασιμότητα, εξασφαλίζοντας υψηλή απόδοση καθώς αυξάνεται ο όγκος δεδομένων. Προσφέρει επίσης αποτελεσματική και γρήγορη προσβασιμότητα (indexing) στα αποθηκευμένα δεδομένα. Αυτός ο συνδυασμός ευελιξίας, επεκτασιμότητας και ταχύτητας καθιστά την MongoDB ιδανική επιλογή για εφαρμογές IoT μεγάλης κλίμακας, όπως βιομηχανικά συστήματα παρακολούθησης ή εκτεταμένα δίκτυα αισθητήρων

Redis (Remote Dictionary Server): είναι μια βάση δεδομένων key-value ανοιχτού κώδικα, η οποία θεωρείται καλή για την απόδοση και την ευελιξία της. Υποστηρίζει μια ποικιλία δομών δεδομένων, συμπεριλαμβανομένων συμβολοσειρών, λιστών, συνόλων, ταξινομημένων συνόλων και κατακερματισμών, προσφέροντας σημαντική ευελιξία στο χειρισμό δεδομένων IoT. Επιπλέον, η Redis διαθέτει ενσωματωμένα χαρακτηριστικά αναπαραγωγής, επεξεργασίας συναλλαγών και υψηλής διαθεσιμότητας, καθιστώντας την κατάλληλη για εφαρμογές IoT σε πραγματικό χρόνο.

Amazon DynamoDB: Η Amazon DynamoDB είναι μια πλήρως διαχειριζόμενη υπηρεσία βάσης δεδομένων NoSQL που προσφέρεται από το Amazon Web Services (AWS). Η DynamoDB παρέχει απρόσκοπτη επεκτασιμότητα, επιτρέποντάς του να χειρίζεται περισσότερα από 10 τρισεκατομμύρια αιτήματα την ημέρα και να υποστηρίζει πάνω από 20 εκατομμυρίων αιτήματα ανά δευτερόλεπτο. Αυτό το επίπεδο απόδοσης και επεκτασιμότητας είναι συχνά απαραίτητο σε εφαρμογές IoT μεγάλης κλίμακας όπου παράγονται συνεχώς τεράστιες ποσότητες δεδομένων. Η ενσωματωμένη ασφάλεια, η δημιουργία αντιγράφων ασφαλείας και επαναφοράς, η προσωρινή αποθήκευση στη μνήμη και οι δυνατότητες αναπαραγωγής (replication) της σε πολλαπλές περιοχές (multi-Region) καθιστούν την DynamoDB μια αξιόπιστη επιλογή για τη διαχείριση δεδομένων IoT, ιδιαίτερα για χρήστες που ήδη λειτουργούν στο οικοσύστημα AWS.

4.6.3 Βάσεις Δεδομένων Χρονοσειρών (TSDB)

Στον κόσμο των Big Data, η αποτελεσματική επεξεργασία και ανάλυση δεδομένων χρονοσειρών αποτελεσματικά είναι μια σημαντική πρόκληση. Τα δεδομένα χρονοσειρών, που χαρακτηρίζονται από διαδοχικά σημεία δεδομένων, είναι πανταχού παρόντα στις σύγχρονες εφαρμογές, από δεδομένα οικονομικών στοιχείων και αρχεία καταγραφής των server έως αναγνώσεις αισθητήρων IoT. Οι βάσεις δεδομένων χρονοσειρών (TSDB) έχουν σχεδιαστεί ειδικά για την αποθήκευση, την ανάκτηση και την επεξεργασία αυτού του τύπου δεδομένων.

InfluxDB: αναπτύχθηκε από την InfluxData, η InfluxDB είναι μία TSDB ανοιχτού κώδικα που δημιουργήθηκε ειδικά για το χειρισμό μετρήσεων και συμβάντων (events). Είναι γραμμένη σε Go και βελτιστοποιημένη για υψηλές ταχύτητες εγγραφής και εκτέλεσης ερωτημάτων/αναζητήσεων, καθιστώντας το ιδανικό για εφαρμογές που απαιτούν αποτελεσματική επεξεργασία δεδομένων χρονοσειρών σε πραγματικό χρόνο.

Η InfluxDB χρησιμοποιεί μια γλώσσα ερωτημάτων που μοιάζει με SQL, η οποία διευκολύνει τους προγραμματιστές που είναι ήδη εξοικειωμένοι με την SQL. Υποστηρίζει επίσης ένα ευέλικτο σχήμα (schemaless), επιτρέποντας την αποτελεσματική διαχείριση δεδομένων χρονοσειρών.

TimescaleDB: είναι μια ακόμη αξιόλογη TSDB ανοιχτού κώδικα, που αναπτύχθηκε ως επέκταση της PostgreSQL. Αξιοποιώντας την ευρωστία και την ευελιξία της PostgreSQL, η TimescaleDB εισάγει εγγενείς λειτουργίες και βελτιστοποιήσεις προσανατολισμένες στον χρόνο, καθιστώντας την κατάλληλη για δεδομένα χρονοσειρών. Η κύρια δύναμη της TimescaleDB έγκειται στην πλήρη υποστήριξη SQL, διευκολύνοντας την εκτέλεση σύνθετων ερωτημάτων. Επιπλέον, η TimescaleDB προσφέρει αυτόματη κατάτμηση σε χρόνο και χώρο, η οποία βοηθά στη βελτίωση της απόδοσης των ερωτημάτων και της διαχείρισης δεδομένων.

Από την άλλη πλευρά, η TimescaleDB ενδέχεται να μην παρέχει το ίδιο επίπεδο απόδοσης εγγραφής με άλλα TSDB για δεδομένα χρονοσειρών υψηλής συχνότητας. Επιπλέον, οι επιλογές κλιμάκωσής της, αν και είναι διαθέσιμες, ενδέχεται να απαιτούν μεγαλύτερη λειτουργική πολυπλοκότητα.

Prometheus: αναπτύχθηκε αρχικά από την SoundCloud και είναι ένα ισχυρό ανοιχτού κώδικα TSDB και σύστημα παρακολούθησης που έχει υιοθετηθεί από το Cloud Native Computing Foundation. Ο Prometheus έχει σχεδιαστεί ειδικά για την 1) παρακολούθηση και την ειδοποίηση (alerting) για διάφορες περιπτώσεις χρήσης - καθιστώντας την ένα πολύτιμο εργαλείο για τους DevOps - και 2) για περιβάλλοντα αναφοράς πραγματικού χρόνου.

Ο Prometheus χρησιμοποιεί ένα μοντέλο έλξης (pull-based) για τη συλλογή μετρήσεων, δηλαδή ο Prometheus Server αντλεί μετρήσεις από την εκάστοτε υποδομή/εφαρμογές, αντί αυτές να στέλνουν μετρήσεις σε αυτόν. Επίσης χρησιμοποιεί μια ευέλικτη γλώσσα ερωτημάτων (queries) που ονομάζεται PromQL, επιτρέποντας την αποτελεσματική συγκέντρωση και ανάλυση δεδομένων. Παρέχει επίσης ένα πολυδιάστατο μοντέλο δεδομένων και ισχυρή υποστήριξη σε μεταδεδομένα. Ο σχεδιασμός του ως σύστημα παρακολούθησης οδηγεί σε ορισμένους περιορισμούς όταν χρησιμοποιείται αποκλειστικά ως TSDB.

4.7 Ολοκληρωμένα Περιβάλλοντα Ανάπτυξης για IoT εφαρμογές

Στον δυναμικό τομέα της ανάπτυξης εφαρμογών για το Διαδίκτυο των Πραγμάτων (IoT), τα Ολοκληρωμένα Περιβάλλοντα Ανάπτυξης (IDEs) χρησιμεύουν ως απαραίτητα εργαλεία τα οποία υποστηρίζουν το σχεδιασμό, τη δοκιμή και την ανάπτυξη εφαρμογών. Μεταξύ της ποικιλίας των διαθέσιμων IDE, δύο πλατφόρμες ξεχωρίζουν από άποψη δημοτικότητας και ευρείας υιοθέτησης: το Arduino IDE και το Visual Studio Code (VSCode) με την προσθήκη του PlatformIO.

Arduino IDE

Το Arduino IDE είναι μια πλατφόρμα γραμμένη σε Java και αναπτύχθηκε ως ένα εργαλείο για τον προγραμματισμό των μικροελεγκτών Arduino, οι οποίοι έχουν γίνει συνώνυμοι με το IoT και τη δημιουργία πρωτοτύπων. Παρέχει μια απλή και φιλική προς το χρήστη διεπαφή με δυνατότητες όπως ένα πρόγραμμα επεξεργασίας κειμένου για τη σύνταξη κώδικα, μια κονσόλα μηνυμάτων για την ανατροφοδότηση από το σύστημα και μια κονσόλα κειμένου για έξοδο προγράμματος.

Ένα από τα κύρια χαρακτηριστικά του Arduino IDE είναι η υποστήριξή του σε γλώσσες προγραμματισμού C και C++, οι οποίες χρησιμοποιούνται συνήθως για τον προγραμματισμό μικροελεγκτών λόγω των δυνατοτήτων και της αποτελεσματικότητάς τους σε προγραμματισμό χαμηλού επιπέδου (low level programming languages). Απλοποιεί επίσης τη διαδικασία μεταγλώττισης και αποστολής κώδικα στους μικροελεγκτές,

καθιστώντας το κατάλληλη επιλογή για αρχάριους στον τομέα της ανάπτυξης εφαρμογών IoT. Επιπλέον, το IDE υποστηρίζει μια μεγάλη ποικιλία από πλακέτες Arduino και υλικό τρίτων (διάφορους άλλους μικροελεγκτές όπως π.χ. ESP8266), παρέχοντας έτσι μεγάλη ευελιξία.

Visual Studio Code (VSCode) με PlatformIO

Το Visual Studio Code (VSCode), αναπτύχθηκε από τη Microsoft και έχει αναδειχθεί ως ένα ισχυρό και ευέλικτο πρόγραμμα επεξεργασίας για διάφορες γλώσσες προγραμματισμού. Αν και δεν είναι εγγενώς IDE για την ανάπτυξη IoT εφαρμογών, η λειτουργικότητά του μπορεί να βελτιωθεί σημαντικά με επεκτάσεις, καθιστώντας το ένα ελκυστικό εργαλείο. Μια αξιοσημείωτη επέκταση είναι το PlatformIO, ένα οικοσύστημα ανοιχτού κώδικα για την ανάπτυξη IoT.

Το PlatformIO επεκτείνει το VSCode σε ένα πλήρως ανεπτυγμένο IDE προσανατολισμένο για την ανάπτυξη IoT εφαρμογών, προσφέροντας ισχυρές δυνατότητες όπως έξυπνη ολοκλήρωση κώδικα, linting, εντοπισμό σφαλμάτων. Υποστηρίζει επίσης ένα ευρύ φάσμα μικροελεγκτών, συμπεριλαμβανομένων των Arduino, ESP32 και STM32, μεταξύ άλλων και είναι συμβατό με πολλές γλώσσες, όπως C, C++ και Python (MicroPython).

Συμπερασματικά, τόσο το Arduino IDE όσο και το VSCode με PlatformIO προσφέρουν ένα σύνολο χαρακτηριστικών που καλύπτουν τις διαφορετικές ανάγκες των προγραμματιστών. Η επιλογή μεταξύ αυτών των πλατφορμών εξαρτάται από διάφορους παράγοντες, όπως η πολυπλοκότητα του έργου, η προτιμώμενη γλώσσα προγραμματισμού και το συγκεκριμένο υλικό που χρησιμοποιείται. Κατανοώντας τα δυνατά σημεία και τις λειτουργίες αυτών των IDE, οι προγραμματιστές μπορούν να πλοηγηθούν πιο αποτελεσματικά στο περιβάλλον της ανάπτυξης IoT εφαρμογών.

5. Υλοποίηση Εφαρμογής

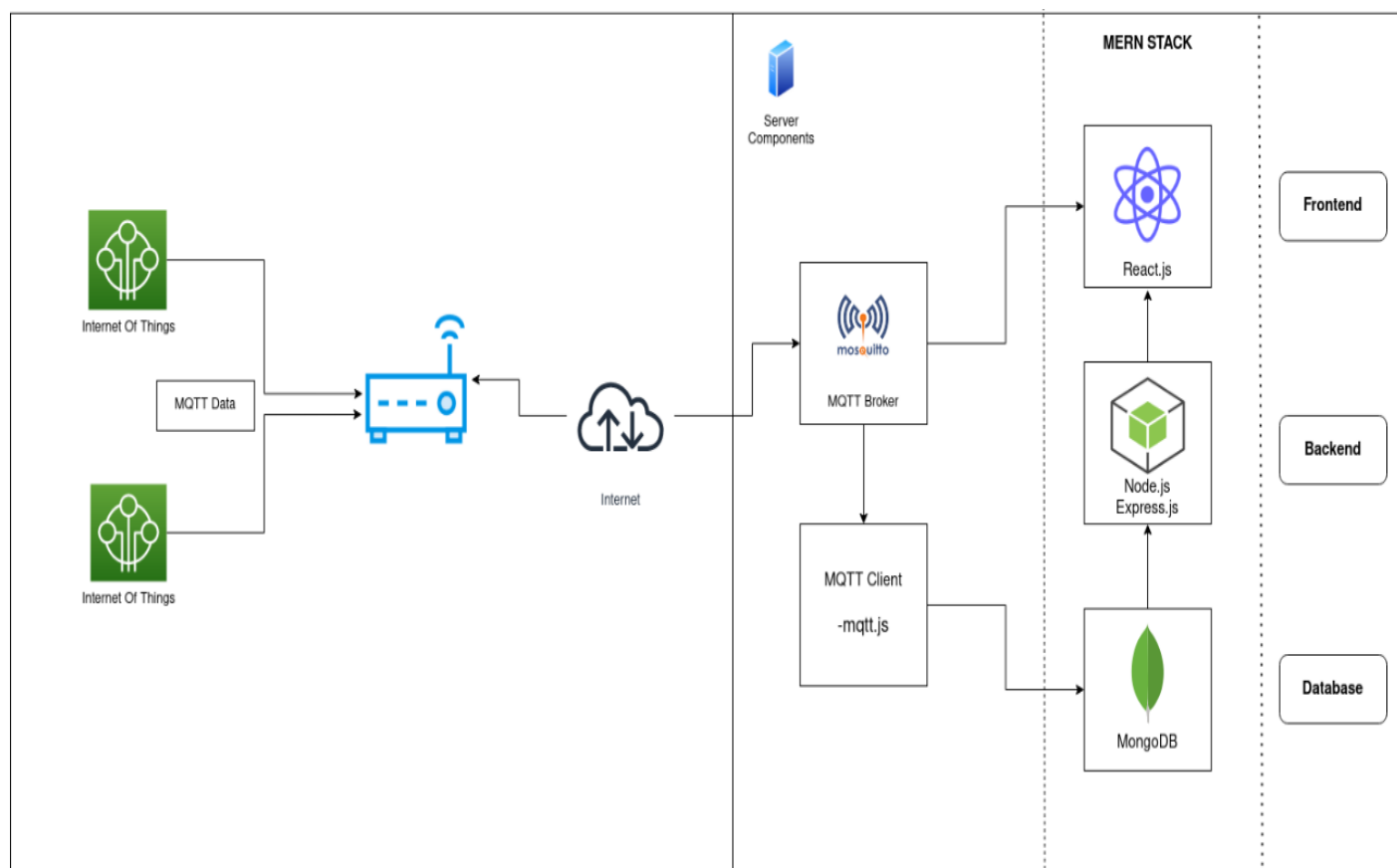
5.1 Απαιτήσεις Συστήματος

Η ιδέα για την ανάπτυξη της εφαρμογής ήταν ο σχεδιασμός μια φιλικής διεπαφής μέσω της οποίας ο χρήστης του συστήματος θα είχε την δυνατότητα να βλέπει πραγματικού χρόνου τιμές από αισθητήρες καθώς και διαγράμματα τα οποία απεικονίζουν τις τιμές από τους αισθητήρες για κάποια συγκεκριμένη ημερομηνία. Ακόμα έπρεπε να δημιουργηθούν οι πρωτότυπες έξυπνες συσκευές οι οποίες είναι υπεύθυνες για την συλλογή των δεδομένων από τους αισθητήρες και την αποστολή τους στην cloud-native εφαρμογή. Κάποιες από τις απαιτήσεις (λειτουργικές και μη) είναι:

1. Επιλογή κατάλληλων αισθητήρων
2. Επιλογή κατάλληλων Development Board
 - 2.1. Χαρακτηριστικά όπως αριθμός GPIO, ADC pin, καθώς και συνδεσιμότητα
3. Επιλογή κατάλληλων βιβλιοθηκών
4. Εύκολη εγκατάσταση των συσκευών σε νέους χώρους
 - 4.1. Δυναμική εισαγωγή των WiFi Credential
 - 4.2. Μικρό footprint
 - 4.3. Prototype Board (και όχι Breadboard)
5. Αποδοτική αξιοποίηση των πόρων για κάθε μικροελεγκτή
6. Επιλογή κατάλληλων τεχνολογιών για την ανάπτυξη της Cloud υποδομής
 - 6.1. Frontend
 - 6.2. Backend
 - 6.3. Middleware
7. Επιλογή Βάσης Δεδομένων
8. Επιλογή message broker
9. Επιλογή IDE

5.2 Αρχιτεκτονική Συστήματος

Στην παρακάτω ενότητα θα γίνει μια ανάλυση της αρχιτεκτονικής του συστήματος που αναπτύχθηκε στα πλαίσια της διπλωματικής εργασίας, με έμφαση στο πως επικοινωνούν τα επιμέρους στοιχεία του συστήματος. Η αρχιτεκτονική συνδυάζει δύο διαφορετικούς μικροελεγκτές για να διαβάζει δεδομένα από τους αντίστοιχους αισθητήρες, τα οποία μεταδίδονται μέσω WiFi, χρησιμοποιώντας το πρωτόκολλο MQTT και έπειτα αποθηκεύονται και προβάλλονται μέσω ενός UI από την cloud υποδομή. Η Cloud υποδομή - περιλαμβάνει διάφορα στοιχεία - χρησιμοποιεί την τεχνολογική στοίβα MERN (MongoDB, Express.js, React.js, Node.js), μαζί με ένα στοιχείο ενδιάμεσου λογισμικού (MQTT Node.js client) που διευκολύνει την επικοινωνία μεταξύ των μικροελεγκτών και του cloud. Η στοίβα MERN είναι υπεύθυνη για τη δημιουργία API για το frontend, για την οπτικοποίηση των δεδομένων από τους αισθητήρες, ενώ μια λειτουργία παρακολούθησης σε πραγματικό χρόνο είναι



Εικόνα 12: Αρχιτεκτονική προτεινόμενου συστήματος

Λήψη Δεδομένων από τους Μικροελεγκτές:

Η αρχιτεκτονική ξεκινά με τους δύο μικροελεγκτές (prototype boards), ο καθένας από τους οποίους είναι εξοπλισμένος με τους αντίστοιχους αισθητήρες για την απόκτηση των απαραίτητων δεδομένων. Αυτοί οι μικροελεγκτές είναι υπεύθυνοι για την ανάγνωση των δεδομένων από τους αισθητήρες και την προετοιμασία τους για την μετάδοση στην cloud υποδομή. Το πρώτο prototype board αφορά την ανάγνωση δεδομένων, συγκεκριμένα την θερμοκρασία και την κατάσταση της πόρτας (OPEN/CLOSE) του ψυκτικού θαλάμου (Cooling Chamber). Το δεύτερο prototype board, είναι υπεύθυνο για την ανάγνωση δεδομένων από τον συμπιεστή (Compressor Unit) των ψυκτικών μηχανημάτων, και συγκεκριμένα την ένταση του ρεύματος, την θερμοκρασία και τέλος την πίεση του ψυκτικού υγρού.

Πρωτόκολλο MQTT για την μετάδοση των δεδομένων:

Για τη μετάδοση των δεδομένων, για κάθε αισθητήρα από τα prototype boards αποτελεσματικά και αξιόπιστα, η αρχιτεκτονική χρησιμοποιεί το πρωτόκολλο MQTT. Το MQTT - όπως προαναφέρθηκε - είναι ένα ελαφρύ πρωτόκολλο ανταλλαγής μηνυμάτων σχεδιασμένο για συσκευές με περιορισμένους πόρους και αναξιόπιστες συνδέσεις δικτύου. Ακολουθεί ένα μοντέλο δημοσίευσης-συνδρομής, όπου οι αισθητήρες ενεργούν ως publishers μηνυμάτων και το middleware στο νέφος ενεργεί ως subscriber. Οι μικροελεγκτές δημοσιεύουν δεδομένα αισθητήρα ως μηνύματα MQTT, τα οποία στη συνέχεια αποστέλλονται μέσω Wi-Fi στον Mosquitto [\[25\]](#) MQTT Broker ο οποίος έχει γίνει deploy στην cloud υποδομή.

Στοιχεία υποδομής νέφους:

Η cloud υποδομή αποτελείται από διάφορα στοιχεία που διευκολύνουν την αποθήκευση, την επεξεργασία και την οπτικοποίηση δεδομένων. Τα βασικά συστατικά περιλαμβάνουν:

1. *MQTT Broker*: ο οποίος είναι υπεύθυνος για τη λήψη των μηνυμάτων MQTT από τους μικροελεγκτές και τη διανομή τους στους αντίστοιχους συνδρομητές εντός του cloud. Λειτουργεί ως κεντρικός κόμβος για την απορρόφηση δεδομένων.
2. *Middleware (MQTT Node.js Client)*: είναι ένα στοιχείο ενδιάμεσου λογισμικού, που υλοποιείται χρησιμοποιώντας έναν MQTT πελάτη, η υλοποίηση του οποίου έχει γίνει σε Node.js, λαμβάνει τα μηνύματα MQTT από τον broker και τα επεξεργάζεται πριν τα αποθηκεύσει σε μια βάση δεδομένων. Αυτό το ενδιάμεσο λογισμικό λειτουργεί ως γέφυρα μεταξύ του MQTT Broker και της βάσης δεδομένων.
3. *Βάση δεδομένων (MongoDB)*: όπως προαναφέρθηκε, ο πελάτης MQTT Node.js αποθηκεύει τα δεδομένα από τους αισθητήρες σε μια βάση δεδομένων MongoDB. Η MongoDB είναι μια βάση δεδομένων εγγράφων NoSQL που παρέχει ευελιξία και επεκτασιμότητα για το χειρισμό μεγάλου όγκου δεδομένων. Επιτρέπει την αποτελεσματική αποθήκευση, ανάκτηση και αναζήτηση των δεδομένων.

Επίσης εδώ αξίζει να αναφερθεί πως η επιλογή αυτής της βάσης έγινε, γιατί είναι από τα κύρια στοιχεία της τεχνολογικής στοίβας MERN. Επίσης από την έκδοση MongoDB 5.0 υποστηρίζονται εγγενώς τα δεδομένα χρονοσειρών.

Οπτικοποίηση δεδομένων στο Frontend:

Το frontend αποτελεί ένα στοιχείο του MERN Stack, το οποίο υλοποιήθηκε με χρήση της React.js και παρέχει μια φιλική προς το χρήστη διεπαφή για την οπτικοποίηση των δεδομένων από τους αισθητήρες. Στη διεπαφή γίνονται αιτήματα API (συγκεκριμένα GET requests) για την ανάκτηση συγκεκριμένων δεδομένων από τη βάση δεδομένων, με βάση τις εισόδους του χρήστη, που είναι ημερομηνία (π.χ. από 10/6/23).

Τα δεδομένα που ανακτώνται εμφανίζονται στη συνέχεια σε γραφήματα, επιτρέποντας στους χρήστες να αναλύουν και να ερμηνεύουν τα δεδομένα που προέρχονται από τους αισθητήρες.

Παρακολούθηση σε πραγματικό χρόνο:

Για να ενεργοποιηθεί η παρακολούθηση σε πραγματικό χρόνο των δεδομένων από τους αισθητήρες, το frontend ενσωματώνει έναν MQTT client. Αυτός ο πελάτης εγγράφεται στα σχετικά θέματα MQTT και λαμβάνει ενημερώσεις κάθε φορά που δημοσιεύονται νέα δεδομένα σε αυτά τα θέματα. Στη συνέχεια, τα δεδομένα εμφανίζονται σε πραγματικό χρόνο στο frontend, παρέχοντας στους χρήστες live πληροφορίες.

Η αρχιτεκτονική που περιγράφεται παραπάνω παρουσιάζει μια αποτελεσματική και επεκτάσιμη προσέγγιση για την απόκτηση, τη μετάδοση, την αποθήκευση και την οπτικοποίηση δεδομένων αισθητήρων σε μια εφαρμογή IoT. Στις επόμενες υποενότητες γίνεται αναφορά στην υλοποίηση τόσο των πρωτότυπων πλακετών όσο και στην cloud υποδομή.

5.3 Ανάπτυξη Πρωτοτύπων

Σε αυτή την υποενότητα θα γίνει λεπτομερής καταγραφή της διαδικασίας που ακολουθήθηκε για την ανάπτυξη των δύο πρωτοτύπων. Το πρώτο πρωτότυπο αφορά τον θάλαμο ψύξης, ενώ το δεύτερο αφορά το συμπιεστή των μηχανημάτων HVAC. Για κάθε ένα από αυτά θα καταγραφεί ο σκοπός, τα εξαρτήματα που χρησιμοποιήθηκαν, το σχήμα τους όπως θα δοκιμαζόταν σε ένα breadboard, τα ηλεκτρονικά διαγράμματα (schematics), οι βιβλιοθήκες και τέλος ο κώδικας σε επίπεδο συναρτήσεων (για συντομία) και τέλος το πρωτότυπο αφού έχουν κολληθεί σε αυτό όλα τα εξαρτήματα.

5.3.1 Prototype board 1: Cooling Chamber

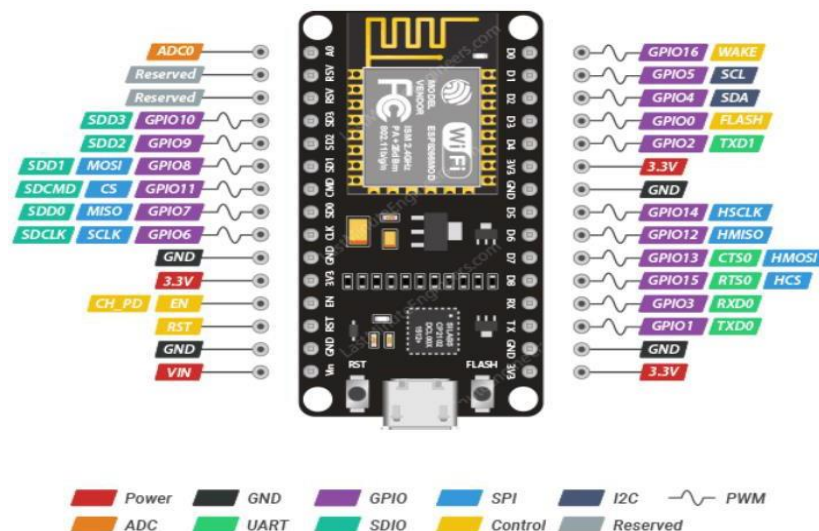
Αυτή η πλακέτα σχεδιάστηκε με σκοπό να καταγράφει και να μεταδίδει σε πραγματικό χρόνο μετρικές που αφορούν την κατάσταση του ψυκτικού θαλάμου, όπως η εσωτερική θερμοκρασία του και η κατάσταση της πόρτας. Με αυτό το τρόπο ο χρήστης έχει την δυνατότητα να έχει απομακρυσμένα την επίγνωση της κατάστασης του ψυκτικού θαλάμου. Η επιλογή του development board έχει ως βάση το μικροελεγκτή ESP8266 και συγκεκριμένα ονομάζεται ESP8266 Lua NodeMCU, ενώ οι αισθητήρες για την απόκτηση των δεδομένων είναι ο αισθητήρας θερμοκρασίας DS18B20 και η μαγνητική επαφή για τον έλεγχο της πόρτας.

Εξαρτήματα

Παρακάτω αναγράφονται οι ονομασίες των εξαρτημάτων και η ποσότητα αυτών.

1. ESP8266 Lua NodeMCU (x1)
2. DS18B20 temperature sensor - waterproof version (x1)
3. Magnetic Reed Switch (x1)
4. 4.7k Ohm Resistor (x1)
5. 10k Ohm Resistor (x1)
6. Prototyping Board 50x70mm Double-Sided (x1)
7. Pin Headers 1x40 Female 2.54mm

ESP8266 Lua NodeMCU



Εικόνα 13 ESP8266 Development Board Pinout

Πηγή: <https://lastminuteengineers.com/esp8266-pinout-reference/>

DS18B20 Temperature Sensor

Ο αισθητήρας θερμοκρασίας DS18B20 είναι ένας ψηφιακός one-wire αισθητήρας θερμοκρασίας. Αυτό σημαίνει ότι απαιτεί απλώς ένα καλώδιο δεδομένων για να επικοινωνήσει με τον ESP8266. Επίσης ο συγκεκριμένος ψηφιακός αισθητήρας επιτρέπει την σύνδεση πολλών αντίστοιχων αισθητηρίων στο ίδιο καλώδιο δεδομένων, με αποτέλεσμα να γίνεται εξοικονόμηση των διαθέσιμων GPIO (General Purpose Input/Output). Για την περίπτωση της συγκεκριμένης πλακέτας ανάπτυξης χρησιμοποιήθηκε η αδιάβροχη εκδοχή του DS18B20.



Εικόνα 14: Αισθητήρας θερμοκρασίας DS18B2

Πηγή: <https://grobotronics.com/ds18b20-el.html>

Χαρακτηριστικά

- Sensor Type: Temperature
- Typical Input Voltage: 3-5 VDC
- Operating Current: 1.5mA
- Interface: Digital
- Communication Protocol: Dallas 1-Wire
- Temperature Range: -55...125 °C

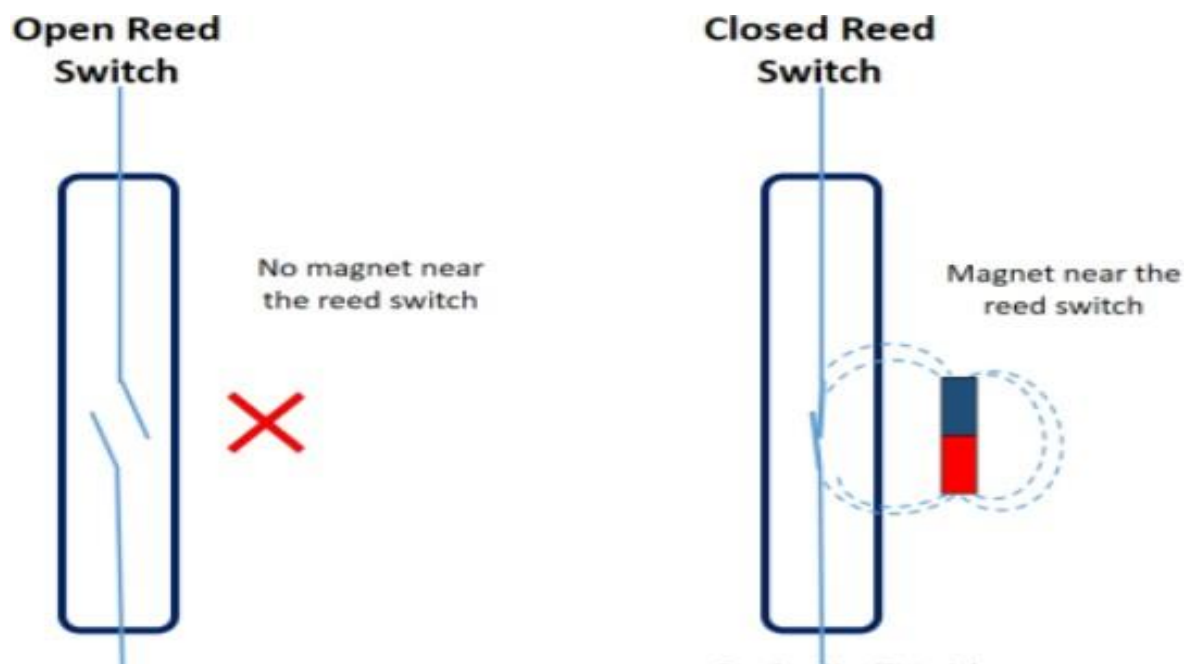
Magnetic Reed Switch

Ένας διακόπτης μαγνητικής επαφής λειτουργεί όπως ένας κανονικός διακόπτης με κουμπί, αλλά αντ' αυτού χρησιμοποιεί το μαγνητικό πεδίο για την αλλαγή κατάστασης (ανοιχτό/κλειστό). Ο διακόπτης αυτός περικλείεται από ένα πλαστικό κουτί, ώστε να μπορεί να εφαρμόζετε εύκολα σε μια πόρτα ή ένα παράθυρο για να εντοπίζετε η κατάσταση του (αν είναι ανοιχτό ή κλειστό). Το ηλεκτρικό κύκλωμα είναι κλειστό όταν ένας μαγνήτης βρίσκεται κοντά στον διακόπτη — πόρτα κλειστή. Όταν ο μαγνήτης είναι πολύ μακριά από τον διακόπτη - πόρτα ανοιχτή - το κύκλωμα είναι ανοιχτό. Στα παρακάτω σχήματα φαίνονται: 1) το αισθητήριο και 2) η λειτουργία του.



Εικόνα 15: Μαγνητική επαφή.

Πηγή: <https://grobotronics.com/ds18b20-el.html>



Εικόνα 16: Λειτουργία μαγνητικής επαφής

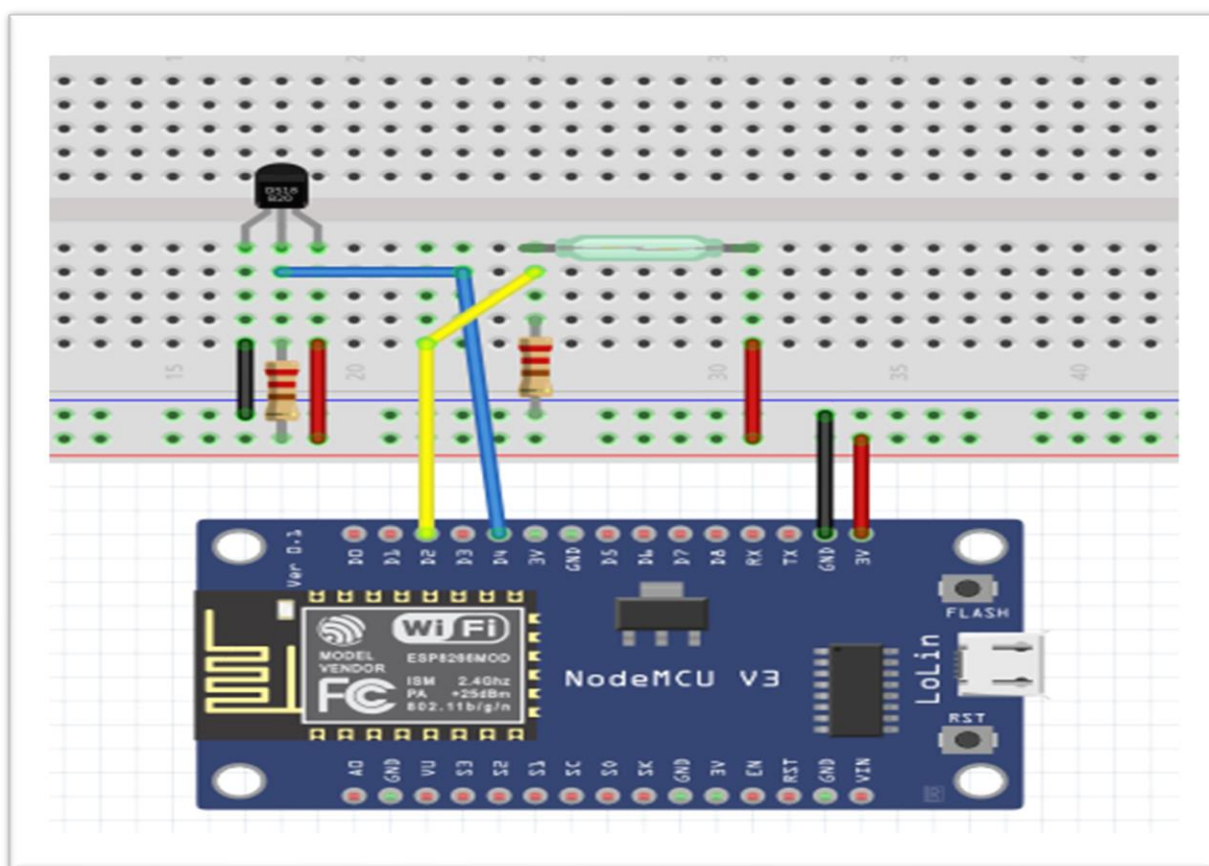
Πηγή: <https://randomnerdtutorials.com/esp32-door-status-telegram/>

Χαρακτηριστικά:

- Sensor Type: Reed Relay
- Interface: Digital
- Contact configuration: SPST-NO
- Switched current: 500mA
- Switched voltage: MAX: 110v

Breadboard Schematic (Pictorial)

Παρακάτω φαίνεται η προσομοίωση του ηλεκτρονικού κυκλώματος που έγινε με την χρήση του λογισμικού Fritzing. Στη φωτογραφία φαίνονται τα αισθητήρια, και η συνδεσμολογία με τον ESP8266. Ο αισθητήρας θερμοκρασίας είναι ο κλασσικός και όχι ο αδιάβροχος επειδή δεν υπήρχε το αντίστοιχο στοιχείο στις βιβλιοθήκες του fritzing όπως και αντίστοιχα για τον άλλο αισθητήρα.

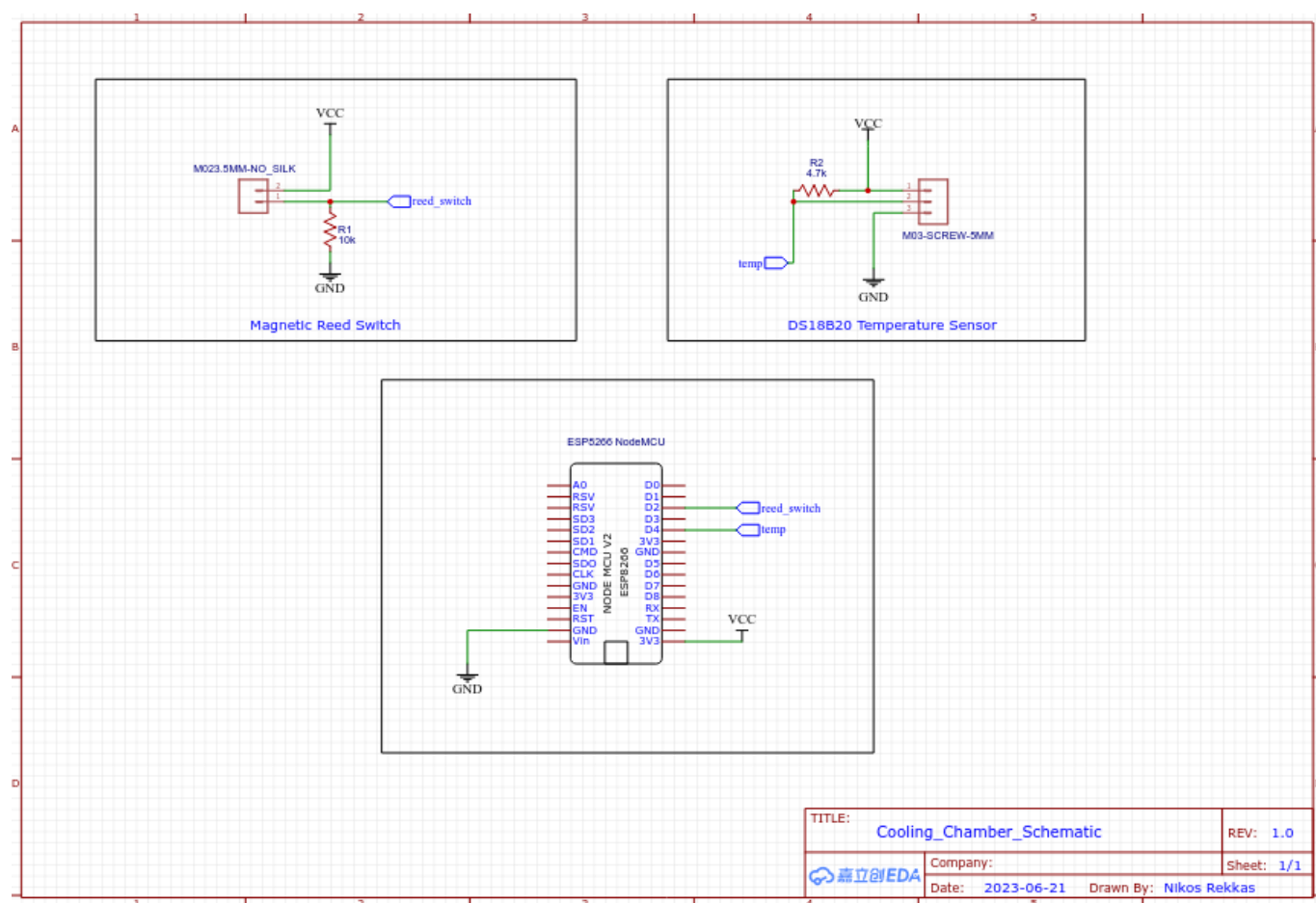


Εικόνα 17: Prototype Board 1 (Cooling Chamber): Pictorial Diagram

Circuit Diagram

Παρακάτω φαίνεται το ηλεκτρονικό κύκλωμα το οποίο σχεδιάστηκε με την βοήθεια του λογισμικού της EasyEDA.

Η συνδεσμολογία για τον αισθητήρα θερμοκρασίας DS18B20 έγινε σύμφωνα με το datasheet (φύλλο δεδομένων) [26] που παρέχεται από τον κατασκευαστή του και η τροφοδοσία του έγινε από την γραμμή δεδομένων (parasite mode). Για τον αισθητήρα magnetic reed switch, η συνδεσμολογία έγινε σύμφωνα με αυτό το άρθρο [27] .



Εικόνα 18: Prototype Board 1 (Cooling Chamber): Pictorial Diagram

Βιβλιοθήκες

Παρακάτω αναγράφονται οι βιβλιοθήκες που χρειάστηκε να εγκατασταθούν και λίγες πληροφορίες για την κάθε βιβλιοθήκη ξεχωριστά:

1. **One Wire** by Paul Stoffregen: Η βιβλιοθήκη One Wire χρησιμοποιείται για την επικοινωνία με συσκευές που ακολουθούν το πρωτόκολλο επικοινωνίας "One Wire". Χρησιμοποιείται συνήθως για τη σύνδεση αισθητήρων θερμοκρασίας, τσιπ μνήμης και άλλων περιφερειακών.
2. **Dallas Temperature** by Miles Burton: έχει σχεδιαστεί ειδικά για αισθητήρες θερμοκρασίας που κατασκευάζονται από την Dallas Semiconductor, όπως ο DS18B20. Αυτή η βιβλιοθήκη λειτουργεί σε συνδυασμό με τη βιβλιοθήκη One Wire για την ανάγνωση τιμών θερμοκρασίας από αυτούς τους αισθητήρες. Απλοποιεί τη διαδικασία ανάγνωσης δεδομένων.
3. **PubSubClient**: Η βιβλιοθήκη PubSubClient είναι μια βιβλιοθήκη πελάτη MQTT. Η βιβλιοθήκη PubSubClient επιτρέπει στη συσκευή να δημοσιεύει μηνύματα σε έναν MQTT broker και να εγγραφεί στα αντίστοιχα θέματα για να λαμβάνει μηνύματα. Παρέχει λειτουργίες για τη σύνδεση με έναν MQTT broker, τη δημοσίευση μηνυμάτων και την εγγραφή σε θέματα, διευκολύνοντας την εφαρμογή της επικοινωνίας MQTT στα έργα σας.
4. **WiFiManager**: Η βιβλιοθήκη WiFiManager απλοποιεί τη διαδικασία της παραμετροποίησης και της διαχείρισης συνδέσεων Wi-Fi σε μια πλακέτα ανάπτυξης. Επιτρέπει δηλαδή στους χρήστες να εισάγουν εύκολα τα Wi-Fi διαπιστευτήριά τους χρησιμοποιώντας μια διεπαφή ιστού (web interface). Η βιβλιοθήκη χειρίζεται την αποθήκευση και την ανάκτηση των διαπιστευτηρίων Wi-Fi και συνδέεται αυτόματα στο διαμορφωμένο δίκτυο. Το WiFiManager είναι ιδιαίτερα χρήσιμο κατά την ανάπτυξη συσκευών IoT που πρέπει να συνδεθούν σε διαφορετικά δίκτυα Wi-Fi, χωρίς να απαιτείται επαναπρογραμματισμός.

Αυτές οι βιβλιοθήκες παρέχουν βασικές λειτουργίες για το πρωτότυπο, όπως πρωτόκολλα επικοινωνίας, ολοκλήρωση αισθητήρων, συνδεσιμότητα δικτύου και ανταλλαγή μηνυμάτων MQTT.

Συναρτήσεις λειτουργίας συσκευής

Παρακάτω στον πίνακα παρουσιάζονται οι συναρτήσεις που αναπτύχθηκαν. Ολόκληρος ο κώδικας μπορεί να βρεθεί στο αντίστοιχο Github Repository [\[24\]](#).

<code>void mqtt_reconnect()</code>	Η συνάρτηση αυτή είναι υπεύθυνη για την προσπάθεια σύνδεσης με τον MQTT Broker κάθε 5 sec.
<code>float getTemperature()</code>	Καταγράφει την τιμή της θερμοκρασίας από τον αισθητήρα και την επιστρέφει σαν πραγματική τιμή απλής ακριβείας (float) τιμή.
<code>bool checkTempSensorStatus()</code>	Η συνάρτηση είναι υπεύθυνη για τον έλεγχο σύνδεσης του αισθητήρα θερμοκρασίας και επιστρέφει μια boolean τιμή ανάλογα την κατάσταση του.
<code>bool checkDoorStatus()</code>	Έλεγχος της κατάστασης της πόρτας και επιστροφή boolean τιμής ανάλογα με τον αν πόρτα είναι ανοιχτή (true) ή κλειστή (false).
<code>void setup()</code>	Σε αυτή τη συνάρτηση γίνεται: 1) Αρχική παραμετροποίηση για το Serial Monitor (εργαλείο το οποίο βοηθάει στην προβολή μηνυμάτων που προέρχονται από τον μικροελεγκτή), για τον αισθητήρα DS18B20, και ρύθμιση των κατάλληλων GPIO, 2) ρύθμιση WiFiManager, 3) ρύθμιση MQTT Broker.
<code>void loop()</code>	Η συνάρτηση αυτή είναι υπεύθυνη 1) για τον συνεχή έλεγχο της σύνδεσης στο wifi, 2) αποστολή των δεδομένων από τα αισθητήρες στα κατάλληλα mqtt topic, 3) αναμονή 3 sec αναμονή 3 sec.

Πίνακας 3: Prototype Board 1 (Cooling Chamber): Firmware Setup

5.3.2 Prototype board 2: Compressor Unit

Αυτή η πλακέτα σχεδιάστηκε με σκοπό να καταγράφει και να μεταδίδει σε πραγματικό χρόνο μετρικές που αφορούν την κατάσταση του συμπιεστή του μηχανήματος, όπως η θερμοκρασία και πίεση του ψυκτικού υγρού καθώς και την ένταση του ρεύματος. Με αυτό το τρόπο ο χρήστης έχει την δυνατότητα να έχει απομακρυσμένα την επίγνωση της κατάστασης του συμπιεστή του μηχανήματος. Εδώ αξίζει να σημειωθεί πως είναι η 'καρδιά' του μηχανήματος αφού το συγκεκριμένο εξάρτημα είναι υπεύθυνο για την κυκλοφορία του ψυκτικού υγρού στο κύκλωμα.

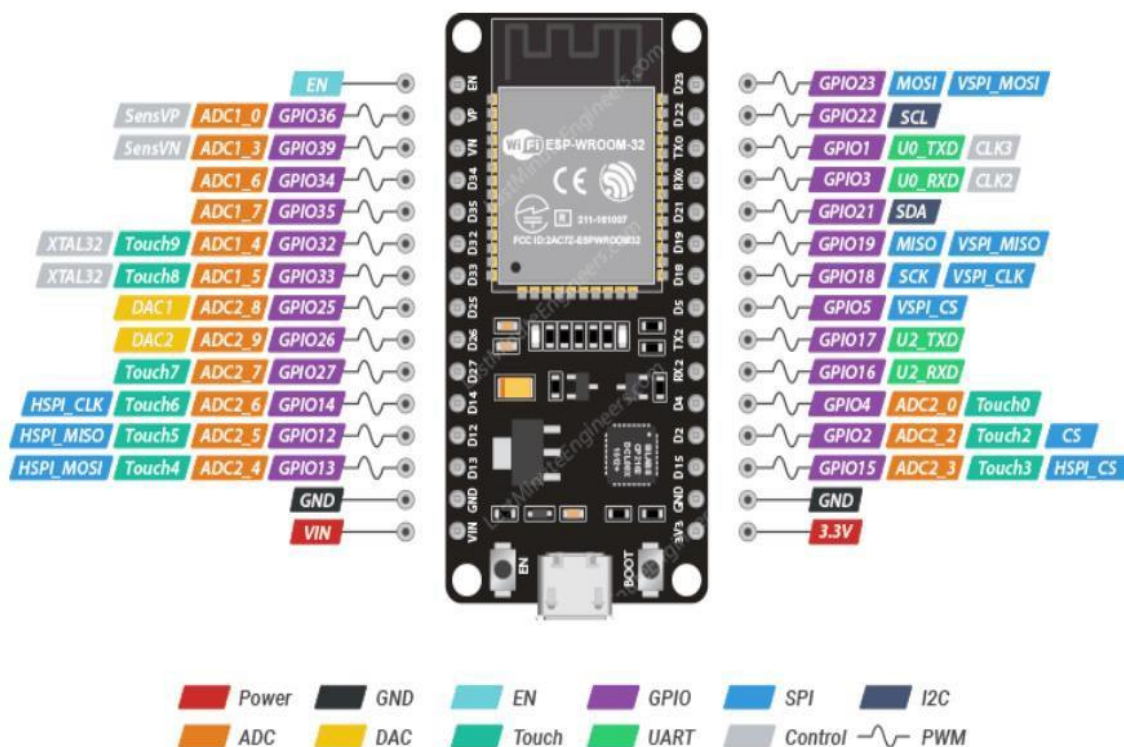
Η επιλογή του development board έγινε με βάση το μικροελεγκτή ESP32 και συγκεκριμένα ονομάζεται ESP32 Development Board - DEVKIT V1, ενώ οι αισθητήρες για την απόκτηση των δεδομένων είναι ένας αναλογικός αισθητήρας θερμοκρασίας (thermistor), ο CT-Clamp sensor για την μέτρηση της έντασης και τέλος ένας αισθητήρας πίεσης (Pressure Transducer). Η επιλογή του συγκεκριμένου μικροελεγκτή έγινε, διότι όλοι οι αισθητήρες είναι αναλογικοί και ο ESP8266 που χρησιμοποιήθηκε στην πρώτη πρωτότυπη πλακέτα μπορεί να υποστηρίξει μόνο έναν αναλογικό αισθητήρα. Ταυτόχρονα ο ESP32 έχει σημαντικά περισσότερες επεξεργαστικές δυνατότητες σε σχέση με τον ESP8266.

Εξαρτήματα

Παρακάτω αναγράφονται οι ονομασίες των εξαρτημάτων και η ποσότητα αυτών.

1. ESP32 Development Board - DevKit V1 (x1)
2. NTC 10k Thermistor Probe Bolt (x1)
3. Non-Invasive Current Sensor 30A (x1)
4. Pressure Transducer Sensor - 1/8 NPT Thread Stainless Steel (500PSI) (x1)
5. Universal Τροφοδοτικό 5v DC power supply (x1)
6. 10k Ohm Resistor (x5)
7. 10μF Electrolytic Capacitor (x1)
8. Prototyping Board 70x90mm Double-Sided (x1)
9. Pin Headers 1x40 Female 2.54mm

ESP32 Development Board - DevKit V1



Εικόνα 19: ESP32 Development Board Pinout

Πηγή: https://lk.linkedin.com/posts/embed-workshop_esp32-dev-kit-v1-pinout-activity-7065278747326787585-7ES8

NTC 10K Thermistor

Ο NTC (Negative Temperature Coefficient) thermistor είναι ένας τύπος αντίστασης που αλλάζει σημαντικά την αντίστασή του με τη θερμοκρασία. Η ονομασία 10K υποδεικνύει ότι ο thermistor έχει αντίσταση 10 kilo-ohms σε θερμοκρασία δωματίου (περίπου 25 βαθμοί Κελσίου). Όταν η θερμοκρασία αυξάνεται, η αντίσταση μειώνεται. γι' αυτό ονομάζεται αρνητικός συντελεστής θερμοκρασίας.



Εικόνα 20: Αισθητήρας θερμοκρασίας NTC 10K

Πηγή: <https://grobotronics.com/ntc-10k-thermistor-digital-temperature-probe-bolt.html?sl=en>

Χαρακτηριστικά

- Sensor Type: Temperature
- Operating Current: 1.5mA
- Interface: Analog
- Temperature Range: -55...125 °C
- Thread width: M8

Non-Invasive Current Sensor

Ένας μη επεμβατικός αισθητήρας ρεύματος, χρησιμοποιείται για τη μέτρηση του ρεύματος που ρέει μέσα από ένα καλώδιο χωρίς να χρειάζεται να έρθει σε άμεση επαφή ή να κοπεί το κάποιο καλώδιο ή να διακοπεί το κύκλωμα. Λειτουργεί με βάση την αρχή της ηλεκτρομαγνητικής επαγωγής. Όταν το ρεύμα διέρχεται από ένα καλώδιο, δημιουργεί ένα μαγνητικό πεδίο γύρω από το καλώδιο. Ο αισθητήρας ανιχνεύει αυτό το μαγνητικό πεδίο και το μετατρέπει σε ηλεκτρικό σήμα ανάλογο με το ρεύμα. Αυτοί οι τύποι αισθητήρων χρησιμοποιούνται συνήθως σε συστήματα παρακολούθησης ισχύος, συστήματα προστασίας από υπέρταση και φορτιστές μπαταριών.



Εικόνα 21: Αισθητήρας ρεύματος

Πηγή: <https://grobotronics.com/non-invasive-current-sensor-30a.html?sl=en>

Χαρακτηριστικά

- Sensor Type: Current
- Interface: Analog
- Input Current: 0-30A AC
- Output Mode: 0~1V DC
- Working Temperature: -25 ~ +70 °C

Pressure Transducer Sensor

Ένας μετατροπέας πίεσης ή ένας αισθητήρας πίεσης είναι μια συσκευή που μετατρέπει την πίεση σε αναλογικό ηλεκτρικό σήμα, συχνά μέσω της χρήσης φυσικής παραμόρφωσης λόγω της πίεσης που ασκείται. Υπάρχουν διάφοροι τύποι αισθητήρων πίεσης που λειτουργούν βάσει διαφορετικών αρχών (όπως πιεζοαντιστικοί, χωρητικοί, ηλεκτρομαγνητικοί κ.λπ.) και χρησιμοποιούνται σε ένα ευρύ φάσμα εφαρμογών. Αυτά περιλαμβάνουν βιομηχανικούς αυτοματισμούς, συστήματα ελέγχου υγρών/αερίων, όργανα καιρού, αεροσκάφη, αυτοκίνητα και ηλεκτρονικά είδη ευρείας κατανάλωσης.

Χαρακτηριστικά

- Sensor Type: Pressure
- Interface: Analog
- Working Voltage: 5VDC
- Output Voltage: 0.5 ~ 4.5 VDC
- Sensor Material: Carbon Steel Alloy
- Working Current: ≤ 10 mA
- Working Pressure Range: 0-500 PSI

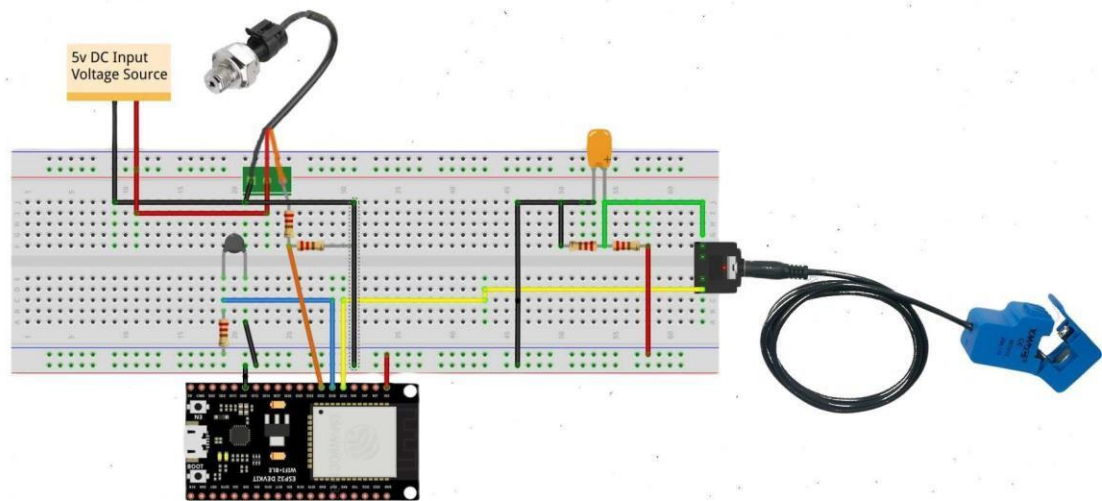


Εικόνα 22: Αισθητήρας πίεσης

Πηγή: <https://www.robofun.ro/lichide/senzor-presiune-transductor-g1-4-inch-1mpa.html>

Breadboard Schematic (Pictorial)

Παρακάτω φαίνεται η προσομοίωση του ηλεκτρονικού κυκλώματος που έγινε με την χρήση του λογισμικού Fritzing. Στην φωτογραφία έχουν προστεθεί με την χρήση λογισμικού Photoshop οι αισθητήρες πίεσης και μέτρησης της έντασης του ρεύματος και αυτό γιατί δεν υπήρχαν ενσωματωμένοι στο fritzing. Στη φωτογραφία φαίνονται τα αισθητήρια, και η συνδεσμολογία με τον ESP32.

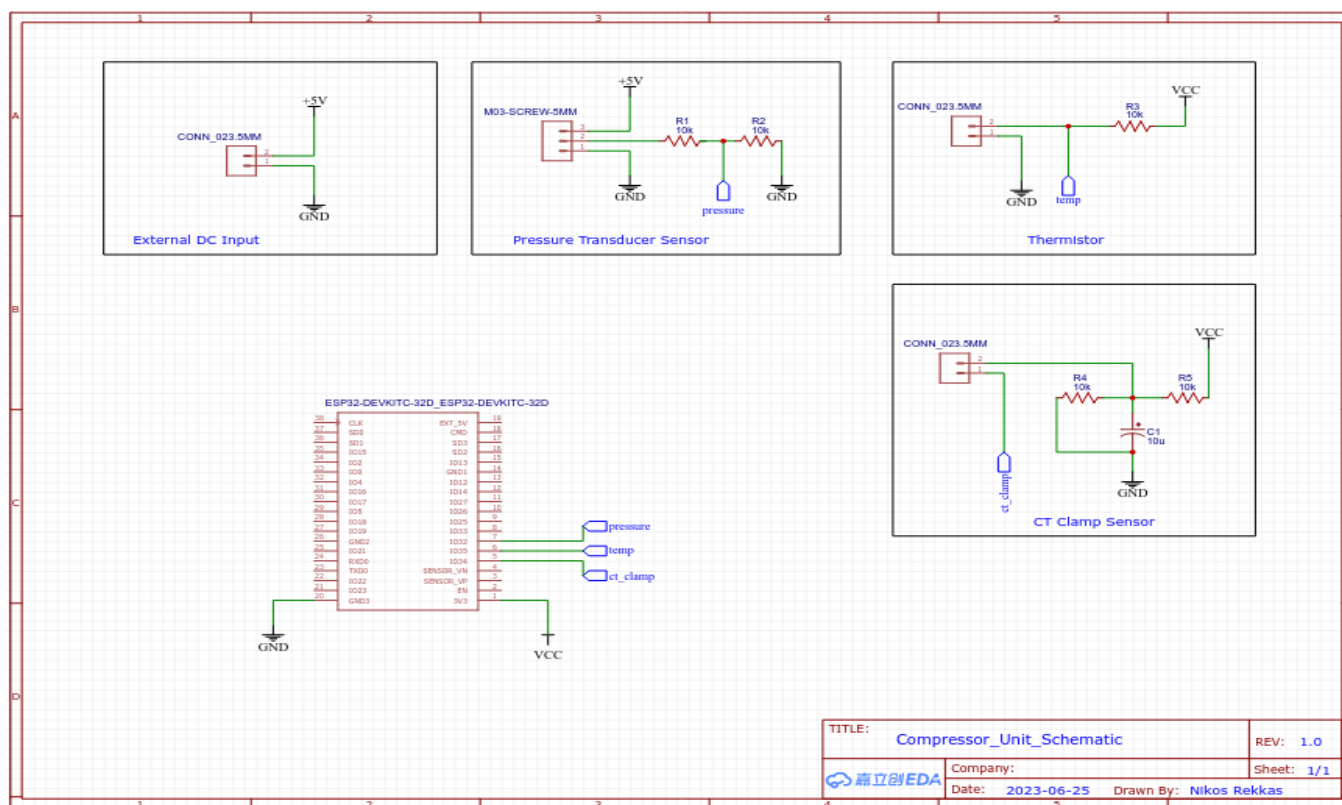


Εικόνα 23: Prototype Board 2 (Compressor Unit): Pictorial Diagram

Circuit Diagram

Παρακάτω φαίνεται το ηλεκτρονικό κύκλωμα το οποίο σχεδιάστηκε και αυτό με την βοήθεια του λογισμικού της EasyEDA.

Το κύκλωμα του αισθητήρα θερμοκρασίας (Thermistor NTC 10K), προσομοιώνει ένα κύκλωμα διαιρέτη τάσης [\[28\]](#), για αυτό υπάρχει η αντίσταση που φαίνεται και στην Εικόνα 24. Ο αισθητήρας πίεσης έχει έξοδο στην γραμμή πληροφορίας 0.5 με 4.5 volt ανάλογα την πίεση. Ο ESP32 μπορεί να 'διαβάσει' τάση από τα analog I/O μέχρι 3.3 volt. Για τον λόγο αυτό υλοποιήθηκε ένα κύκλωμα διαιρέτη τάσης, με σκοπό οι τιμές της τάσης εξόδου από τον αισθητήρα να μπορεί να τις διαβάσει ο ESP32. Τέλος για το κύκλωμα του αισθητήρα έντασης ρεύματος, η συνδεσμολογία έγινε σύμφωνα με το άρθρο [\[29\]](#).



Εικόνα 24: Prototype Board 2 (Compressor Unit): Schematic Diagram

Βιβλιοθήκες

Οι βιβλιοθήκες που χρειάστηκε να εγκατασταθούν είναι σημαντικά λιγότερες, διότι οι αισθητήρες είναι όλοι αναλογικοί. Επίσης οι βιβλιοθήκες που εγκαταστάθηκαν για μετάδοση των mqtt μηνυμάτων είναι ίδια, όπως και για την εισαγωγή των WiFi credentials. Η μόνη βιβλιοθήκη που διαφοροποιεί την εγκατάσταση σε σχέση με την προηγούμενη πρωτότυπη πλακέτα είναι η βιβλιοθήκη EmonLib. Παρακάτω αναγράφονται κάποιες πληροφορίες για την βιβλιοθήκη EmonLib.

EmonLib by OpenEnergyMonitor: Η βιβλιοθήκη EmonLib είναι ένα χρήσιμο εργαλείο για εφαρμογές παρακολούθησης και μέτρησης ενέργειας. Παρέχει λειτουργίες και αλγόριθμους για τον υπολογισμό της κατανάλωσης ενέργειας και της χρήσης ενέργειας με βάση τις εισόδους από αισθητήρες ρεύματος και τάσης.

Συναρτήσεις λειτουργίας συσκευής

Παρακάτω στον πίνακα παρουσιάζονται οι συναρτήσεις που αναπτύχθηκαν. Ολόκληρος ο κώδικας μπορεί να βρεθεί επίσης στο αντίστοιχο Github Repository [\[24\]](#).

<code>void mqtt_reconnect()</code>	Η συνάρτηση αυτή είναι υπεύθυνη για την προσπάθεια σύνδεσης με τον MQTT Broker κάθε 5 sec.
<code>float getTemperature()</code>	Καταγράφει την τιμή της θερμοκρασίας από τον αισθητήρα και την επιστρέφει σαν πραγματική τιμή απλής ακριβείας (float).
<code>double getCurrent()</code>	Καταγράφει την τιμή από τον αισθητήρα έντασης του ρεύματος και την επιστρέφει σαν double τιμή.
<code>double getPressure()</code>	Καταγράφει την τιμή από τον αισθητήρα πίεσης και την επιστρέφει σαν πραγματική τιμή διπλής ακριβείας (double).
<code>void setup()</code>	Σε αυτή τη συνάρτηση γίνεται: 1) Αρχική παραμετροποίηση για το Serial Monitor, ρύθμιση των κατάλληλων GPIO, 2) ρύθμιση WiFiManager, 3) ρύθμιση MQTT Broker.
<code>void loop()</code>	Η συνάρτηση αυτή είναι υπεύθυνη 1) για τον συνεχή έλεγχο της σύνδεσης στο wifi, 2) αποστολή των δεδομένων από τα αισθητήρια στα κατάλληλα mqtt topic, 3) αναμονή 3 sec.

Πίνακας 4: Prototype Board 2 (Compressor Unit): Firmware Setup

5.4 Cloud Υποδομή

Η αρχιτεκτονική της cloud υποδομής βασίζεται στη στοίβα MERN (MongoDB, Express.js, React.js και Node.js), μαζί με πρόσθετα στοιχεία όπως το mqtt_client.js για το ενδιαμέσο λογισμικό και έναν διαμεσολαβητή (broker) MQTT.



Εικόνα 25: User Interface (UI)

5.4.1 Backend

Οι κύριες τεχνολογίες που επέτρεψαν την ανάπτυξη του backend είναι η Node.js και η Express.js, μαζί με κάποιες ακόμα βιβλιοθήκες όπως η dotenv και η mongoose.

Node.js είναι ένα runtime περιβάλλον χτισμένο με την V8 Javascript engine και επιτρέπει την εκτέλεση Javascript κώδικα από την πλευρά του διακομιστή (server-side). Η Node.js παρέχει στους developers ένα ενοποιημένο σύστημα που βασίζεται στην Javascript για την δημιουργία επεκτάσιμων και υψηλής απόδοσης εφαρμογών.

Express.js είναι ένα μινιμαλιστικό framework για εφαρμογές ιστού το οποίο βασίζεται στην Node.js. Η Express.js προσεγγίζει με απλό τρόπο την ανάπτυξη εφαρμογών ιστού που αποτέλεσμα έχει την εύκολη δημιουργία APIs. Ακόμα παρέχει βασικές δυνατότητες, όπως δρομολόγηση, υποστήριξη ενδιάμεσου λογισμικού, ενώ εξακολουθεί να προσφέρει την ευελιξία για την ενσωμάτωση πρόσθετων βιβλιοθηκών.

Η δομή του του backend φαίνεται παρακάτω.

```

- backend
  |- models
    | |- ampere.js
    | |- door.js
    | |- pressure.js
    | |- temp.js
  |- routes
    | |- routes.js
- server.js

```

Στο κύριο φάκελο “backend” υπάρχουν όλα τα απαραίτητα αρχεία. Ο φάκελος models περιέχει τα μοντέλα δεδομένων/σχήματα για τις διαφορετικές οντότητες: “ampere.js”, “door.js”, “pressure.js” και “temp.js”. Ο φάκελος “routes” περιέχει την λογική για κάθε ένα από τα διαθέσιμα API. Τέλος το “server.js” αρχείο είναι το σημείο εκκίνησης της εφαρμογής (του server) ενώ ταυτόχρονα περιέχει όλα τα απαραίτητα configurations.

Μοντέλα Δεδομένων

Για κάθε αισθητήριο δημιουργήθηκε το αντίστοιχο μοντέλο δεδομένων. Σε κάθε Model Schema περιέχονται τρία πεδία. Τα κοινά πεδία είναι δύο σε όλα τα μοντέλα δεδομένων, αυτά είναι το topic (π.χ CompressorUnit/Temperature ή CoolingChamber/DoorStatus) και η χρονοσφραγίδα (timestamp). Το πεδίο που δεν είναι κοινό, είναι η τιμή, για το “temp.js” το πεδίο ονομάζεται temperature, για το “ampere.js” είναι το ampere, για το “pressure.js” το pressure και τέλος για το “door.js” το isOpen.

Model Schema	Additional Fields
Ampere	topic String Required ampere Number Required timestamp Date Required
Door	topic String Required isOpen Boolean Required timestamp Date Required
Pressure	topic String Required pressure Number Required timestamp Date Required
Temperature	topic String Required temperature Number Required timestamp Date Required

Πίνακας 5: Μοντέλα Δεδομένων και αντίστοιχα πεδία.

Ο Πίνακας παρέχει μια σαφή επισκόπηση των μοντέλων δεδομένων που χρησιμοποιήθηκαν για την διαμόρφωση της βάσης δεδομένων, μαζί με τα πρόσθετα πεδία τους. Κάθε σειρά αντιπροσωπεύει ένα σχήμα μοντέλου, με την πρώτη στήλη να είναι το όνομα του και την δεύτερη να είναι τα πεδία που υπάρχουν σε κάθε ένα από αυτά. Για κάθε σχήμα μοντέλου, ο πίνακας παραθέτει το όνομα του πεδίου, τον τύπο δεδομένων και εάν είναι υποχρεωτικό το πεδίο.

APIs

Όπως προαναφέρθηκε στο αρχείο “routes.js” αναπτύχθηκαν διάφορα API endpoints για την ανάκτηση δεδομένων που σχετίζονται με μετρήσεις θερμοκρασίας, πίεσης, έντασης ρεύματος και κατάσταση πόρτας. Η λειτουργικότητα του περιγράφεται ως εξής:

- Το αρχείο εισάγει τα απαιτούμενα μοντέλα για να μπορεί να αλληλοεπιδράσει με την βάση δεδομένων.
- Ορίζονται πολλές διαδρομές (routes) GET που αντιστοιχούν σε διαφορετικές λειτουργίες ανάκτησης δεδομένων.
- Σε κάθε διαδρομή (route) ο κώδικας εκτελεί λειτουργίες ώστε τελικά να επιστρέψει σε μορφή json τα κατάλληλα αποτελέσματα.

API Endpoint	Description
GET /coolingChamber/temperature/:id	Ανακτά δεδομένα θερμοκρασίας για το θάλαμο ψύξης με βάση μια συγκεκριμένη ημερομηνία.
GET /coolingChamber/doorStatus/:id	Ανακτά δεδομένα για την κατάσταση την πόρτας στο θάλαμο ψύξης με βάση μια συγκεκριμένη ημερομηνία.
GET /compressorUnit/temperature/:id	Ανακτά δεδομένα θερμοκρασίας για τη μονάδα συμπίεστη με βάση μια συγκεκριμένη ημερομηνία.
GET /compressorUnit/pressure/:id	Ανακτά δεδομένα πίεσης για τη μονάδα συμπίεστη με βάση μια συγκεκριμένη ημερομηνία.
GET /compressorUnit/ampere/:id	Ανακτά δεδομένα έντασης ρεύματος για τη μονάδα συμπίεστη με βάση μια συγκεκριμένη ημερομηνία.

Πίνακας 6: API Endpoints

5.4.2 Frontend

Για την δημιουργία του frontend χρησιμοποιήθηκε η React.js βιβλιοθήκη που είναι υλοποιημένη σε Javascript. Η React, είναι μια δηλωτική (δηλαδή περιγράφεται η τελική κατάσταση της επιθυμητής διεπαφής χρήστη), αποτελεσματική και ευέλικτη βιβλιοθήκη που αναπτύχθηκε από τη Facebook το 2013. Έχει αποκτήσει σημαντική δημοτικότητα λόγω της αποτελεσματικότητας της στη δημιουργία διαδραστικών διεπαφών χρήστη για single-page εφαρμογές. Η βασική ιδέα στη React περιστρέφεται γύρω από την δημιουργία επαναχρησιμοποιήσιμων στοιχείων (components) για το user interface (UI), τα οποία βοηθούν στη διατήρηση του κώδικα σε μεγάλα project. Κάποια από τα βασικά στοιχεία της React είναι τα εξής: 1) React Components, 2) State Management, 3) React Router, 4) API Requests, 5) Testing, 6) Styling.

Παρακάτω αναλύονται τα αρχεία της εφαρμογής του frontend. Ξεκινώντας με το “App.js” που είναι το κύριο αρχείο της εφαρμογής, το οποίο ξεκινάει πρώτα να κάνει render το στοιχείο Dashboard. Το αρχείο “Dashboard.js” αξιοποιεί διάφορα χαρακτηριστικά, πρώτα γίνεται η εισαγωγή των βιβλιοθηκών “react-icons” για την διακόσμηση του UI - και συγκεκριμένα των ‘καρτών’ που προβάλλονται σε πραγματικό χρόνο τα δεδομένα από τους αισθητήρες -και τη βιβλιοθήκη MQTT για την εγγραφή στα αντίστοιχα topics.

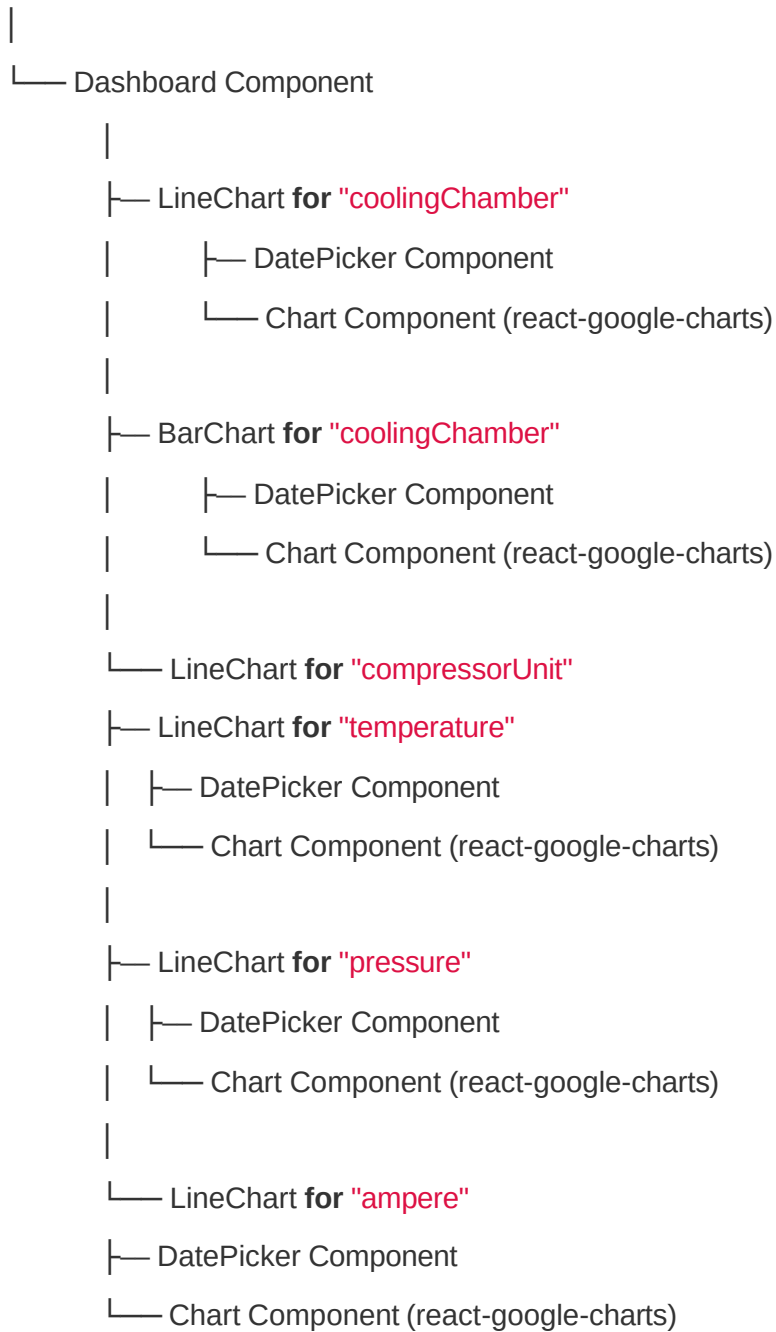
Συγκεκριμένα γίνεται σύνδεση στο MQTT Broker τοπικά και έπειτα η εγγραφή στα παρακάτω topic:

- ‘CoolingChamber/Temperature’
- ‘CoolingChamber/DoorStatus’
- ‘CompressorUnit/Temperature’
- ‘CompressorUnit/Ampere’
- ‘CompressorUnit/Pressure’

Τέλος ανάλογα την κάρτα την οποία επιλέξει ο χρήστης προβάλλονται τα αντίστοιχα διαγράμματα χρονοσειρών. Η συνάρτηση η οποία είναι υπεύθυνη να κάνει το conditional template είναι η `getSelectedScreen()`. Εάν επιλεγθεί το ‘coolingChamber’ card επιστρέφει ένα LineChart (γραμμικό διάγραμμα) και ένα BarChart (ραβδοειδές διάγραμμα), ενώ εάν επιλεγθεί το ‘compressorUnit’ card επιστρέφει τρία LineChart.

Τέλος τα αρχεία ‘LineChart.js’ και ‘BarChart.js’ είναι υπεύθυνα για την εμφάνιση των Line και Bar διαγραμμάτων με βάση τα δεδομένα που λαμβάνονται από τον server και συγκεκριμένα από τις ημερομηνίες που επιλέγει ο χρήστης. Τα component αυτά δέχονται δύο ορίσματα: 1) `api`, το endpoint από το οποίο θα γίνει η λήψη των δεδομένων και 2) `chartOptions`, που είναι οι επιλογές διαμόρφωσης της εμφάνισης/περιεχομένου των γραφημάτων. Και σε αυτά τα component γίνεται conditional template αλλά σε αυτή την περίπτωση γίνεται για τον έλεγχο των διαθέσιμων τιμών. Αν δεν υπάρχουν τιμές για την επιλεγθείσα ημερομηνία τότε προβάλλετε ένα μήνυμα το οποίο ειδοποιεί τον χρήστη ότι δεν υπάρχουν διαθέσιμα δεδομένα για εκείνη την ημέρα. Παρακάτω γίνεται παρουσίαση της δένδροειδούς δομής των components της εφαρμογής.

App Component



5.4.3 Middleware

Στα πλαίσια του Web Development, το ενδιαμέσο λογισμικό ή αλλιώς Middleware είναι λογισμικό το οποίο συνδέει διαφορετικά συστήματα ή στοιχεία λογισμικού μεταξύ τους. Το αρχείο 'mqttClient.js' είναι ουσιαστικά ένα ενδιαμέσο λογισμικό MQTT που είναι υπεύθυνο για να 'ακούει' μηνύματα από διαφορετικά topics από τον MQTT Broker και στην συνέχεια να τα αποθηκεύει στη βάση δεδομένων MongoDB. Η βιβλιοθήκη 'mqtt' για Node.js, χρησιμοποιήθηκε για να υλοποιηθούν οι παραπάνω λειτουργίες. Η βιβλιοθήκη [mqtt](#) είναι μια βιβλιοθήκη πελάτη για το πρωτόκολλο MQTT, γραμμένη σε Javascript για την Node.js και για τους περιηγητές/φυλλομετρητές των συμβατικών υπολογιστών (η ίδια βιβλιοθήκη χρησιμοποιήθηκε στο frontend). Παρακάτω γίνεται ανάλυση της λογικής του 'mqttClient.js'.

1. *Εισαγωγή βιβλιοθηκών και μοντέλων δεδομένων*: Στην αρχή γίνεται εισαγωγή της βιβλιοθήκης 'mqtt', 'mongoose' και των κατάλληλων μοντέλων/σχημάτων δεδομένων για την αλληλεπίδραση με την mongoose
2. *Σύνδεση με την MongoDB*: δημιουργείται μια σύνδεση με την βάση δεδομένων MongoDB χρησιμοποιώντας την βιβλιοθήκη 'mongoose'.
3. *Σύνδεση και Εγγραφή στον MQTT Broker*: δημιουργείται μια σύνδεση με τον MQTT Broker και γίνεται εγγραφή στα αντίστοιχα topics που σχετίζονται με τον θάλαμο ψύξης και τη μονάδα συμπίεστη.
4. *Χειρισμός εισερχόμενων μηνυμάτων MQTT*: όταν φθάνει ένα μήνυμα ενεργοποιείται η συνάρτηση client.on('message') και με βάση το topic για κάθε εισερχόμενο μήνυμα δημιουργείται ένα νέο έγγραφο χρησιμοποιώντας το κατάλληλο μοντέλο και τέλος αποθηκεύεται στη βάση δεδομένων MongoDB.

Το αρχείο αυτό είναι κρίσιμο στοιχείο της εφαρμογής, διότι επιτρέπει την διατήρηση ενός ιστορικού αρχείου δεδομένων για κάθε αισθητήρα στην βάση δεδομένων. Τα δεδομένα αυτά είναι εύκολο να ανακτηθούν από το Backend με σκοπό την δημιουργία APIs και τέλος την οπτικοποίηση αυτών στο Frontend.

5.4.4 MQTT Broker

Το Mosquitto είναι ένας από τους πιο δημοφιλής MQTT Broker. Το Mosquitto είναι μέρος του Eclipse Foundation και είναι ένα έργο iot.eclipse.org. Το MQTT έχει σχεδιαστεί για ανταλλαγή μηνυμάτων (μικρού όγκου) χρησιμοποιώντας το μοντέλο δημοσίευσης/συνδρομής (Pub/Sub). Χρησιμοποιείται συχνά σε αισθητήρες χαμηλής ισχύος, κινητές συσκευές και ενσωματωμένα συστήματα. Στον Mosquitto MQTT Broker μπορούν να διαμορφωθούν αρκετές ρυθμίσεις. Αυτές οι ρυθμίσεις καθορίζονται στο mosquitto.conf αρχείο. Κάποιες από αυτές που υλοποιήθηκαν είναι εξής:

- **persistence_true**: ο broker αποθηκεύει πληροφορίες για τους subscribers, μηνύματα σε ουρά (queue messages) για επαναφορά σε περίπτωση επανεκκίνησης του Broker.
- **persistence_location** : είναι η ρύθμιση της τοποθεσίας που αποθηκεύονται τα παραπάνω δεδομένα που αφορούν τους subscribers. Η εντολή λέει να αποθηκεύονται τα δεδομένα στο /var/lib/mosquitto.
- **log_dest** : η ρύθμιση της τοποθεσίας που αποθηκεύονται τα logs που παράγονται από τον broker. Η εντολή λέει να αποθηκεύονται τα δεδομένα στο /var/log/mosquitto.log
- **listener 1883**: η επιλογή αυτή επιτρέπει τις εισερχόμενες συνδέσεις στην πόρτα 1883 (mqtt protocol). Η συγκεκριμένη ρύθμιση αφορά μη κρυπτογραφημένες συνδέσεις με το πρωτόκολλο MQTT.

- **listener 8080:** αντίστοιχα με παραπάνω, μόνο που στην συγκεκριμένη περίπτωση η πόρτα αυτή είναι για μη κρυπτογραφημένες συνδέσεις που αφορούν το επικοινωνιακό πρωτόκολλο WebSocket. Το πρωτόκολλο αυτό παρέχει κανάλια επικοινωνίας, πλήρους αμφίδρομης λειτουργίας μέσω μιας ενιαίας σύνδεση TCP/IP. Η συγκεκριμένη επιλογή έγινε διότι ο mqtt client στο frontend μπορούσε να 'μιλήσει' με τον broker μέσω websockets.
- **protocol websockets:** εδώ καθορίζεται ότι η πόρτα 8080 θα πρέπει να χρησιμοποιεί το πρωτόκολλο WebSocket. Όπως προαναφέρθηκε αυτή η λειτουργία είναι ιδιαίτερα χρήσιμη για την ενεργοποίηση της επικοινωνίας με MQTT, απευθείας από Web εφαρμογές

6. Αποτελέσματα

Η εφαρμογή στο σύνολο της λειτουργεί πολύ καλά και φαίνεται να έχει επιτευχθεί ο στόχος της παρακολούθησης - μέσω των έξυπνων συσκευών - σε πραγματικό χρόνο του θαλάμου ψύξης και του συμπιεστή του μηχανήματος των συστημάτων HVAC. Εδώ αξίζει να αναφερθεί ότι λόγω τεχνικής δυσκολίας οι έξυπνες συσκευές δεν εγκαταστάθηκαν σε κάποιο μηχάνημα. Για να γίνει όμως έλεγχος της σωστής λειτουργίας των έξυπνων συσκευών έγιναν κάποια πειράματα για κάθε αισθητήρα. Ο μόνος αισθητήρας ο οποίος δεν εξετάστηκε είναι ο αισθητήρας πίεσης. Για τον συγκεκριμένο αισθητήρα σε μηδενική πίεση - δηλαδή σε κατάσταση ηρεμίας - έβγαζε στην έξοδο του τιμή τάσης λίγο μικρότερη από την προβλεπόμενη (0.5VDC), ενώ ταυτόχρονα ο ESP32 φάνηκε να μην δύναται να διαβάσει με μεγάλη ακρίβεια τάσεις οι οποίες είναι τόσο χαμηλές (<0.4 VDC). Παρά την μικρή αστοχία για τον αισθητήρα πίεσης, οι αισθητήρες θερμοκρασίας λειτούργησαν με τον αναμενόμενο τρόπο. Για να είναι σίγουρη η σωστή λειτουργία τους έγινε cross validation, δηλαδή στο dashboard του frontend παρατηρήθηκε ότι και οι δύο αισθητήρες και ο αναλογικός και ο ψηφιακός αισθητήρας θερμοκρασίας - στον ίδιο χώρο - παρουσίαζαν σχεδόν πανομοιότυπες τιμές. Τέλος για τον αισθητήρα κατανάλωσης ρεύματος, για να εξεταστεί η σωστή λειτουργία του αλλά και για να γίνει το callibration στην αρχή (στο κώδικα του firmware) χρησιμοποιήθηκε μια αμπεροτσιμπίδα για την μέτρηση της έντασης του ρεύματος.

7. Συμπεράσματα / Μελλοντική Εργασία

7.1 Συμπεράσματα

Σκοπός αυτής της διπλωματικής εργασίας ήταν να υλοποιηθεί μια end-to-end λύση για την παρακολούθηση σε πραγματικό χρόνο των συστημάτων ψύξης θέρμανσης και κλιματισμού. Έτσι αναπτύχθηκαν δύο πρωτότυπες έξυπνες συσκευές, όπως και ένα web app σε σκοπό την παρακολούθηση σε πραγματικό χρόνο των συστημάτων HVAC. Οι μικροελεγκτές που επιλέχθηκαν είναι από τους πιο γνωστούς τόσο στο ερασιτεχνικό κοινό που ασχολείται με τις κατασκευές ως hobby, όσο και σε επαγγελματικές/εμπορικές IoT εφαρμογές. Η τεχνολογική στοίβα που επέλεξα για την ανάπτυξη της διεπαφής του χρήστη, όσο και του Server είναι μια από τις πιο γνωστές. Η επιλογή αυτών των τεχνολογιών έγινε με σκοπό να διαπιστωθεί εάν είναι κατάλληλες για την ανάπτυξη εφαρμογών 'προσανατολισμένων' στο Διαδίκτυο των Πραγμάτων.

Ακόμα αξίζει να σημειωθεί πως σε επίπεδο υλικού (hardware), η δεύτερη πλακέτα ανάπτυξης θα έπρεπε να είχε ακόμα έναν αισθητήρα θερμοκρασίας και έναν πίεσης για να μετράει την πίεση του ψυκτικού υγρού στην είσοδο του συμπιεστή. Η υλοποίηση δεν έγινε όμως γιατί ήταν αρκετά πιο ακριβή. Η επιλογή όμως του μικροελεγκτή έγινε με σκοπό κάποια μετέπειτα υλοποίηση να υποστηρίζει την εισαγωγή των προαναφερθέντων αισθητηρίων. Ακόμα λόγω του αυξημένου κόστους αυτής της υλοποίησης νόημα έχει η εγκατάσταση αυτών των έξυπνων συσκευών σε μεγάλα HVAC συστήματα, μία μετρική που καθορίζει τα 'μεγάλα' είναι η ιπποδύναμη του συμπιεστή.

Τέλος το αποτέλεσμα της ανάπτυξης των έξυπνων συσκευών, όπως και του frontend/backend είναι η ύπαρξη **πληροφορίας**. Η πληροφορία αυτή μπορεί να είναι πραγματικού χρόνου ή πληροφορία μέσα στο χρόνο. Έτσι οι χρήστες - τόσο οι end users, δηλαδή οι κάτοχοι των μηχανημάτων αυτών όσο και οι τεχνικοί - μπορούν να καταλαβαίνουν με καλύτερο τρόπο την 'συμπεριφορά' αυτών των μηχανημάτων και να δρουν πιο άμεσα.

7.2 Μελλοντική Εργασία

Η υλοποίηση του συστήματος, με όλα τα μέρη που το απαρτίζουν, είναι μια ρεαλιστική λύση για την παρακολούθηση των μηχανημάτων HVAC. Για μια εμπορικά έτοιμη λύση με υψηλότερο επίπεδο τεχνολογικής ετοιμότητας, που θα μπορούσε να είναι προϊόν και να

απευθυνθεί σε διάφορους κλάδους της βιομηχανίας, θα προτεινόταν οι εξής αλλαγές, ξεκινώντας από τα χαμηλότερα στρώματα που είναι οι έξυπνες συσκευές, καταλήγοντας στην Cloud υποδομή:

Hardware

- *Compressor Unit*: για αυτήν την έξυπνη συσκευή όπως προαναφέρθηκε θα χρειαζόταν να γίνει εισαγωγή ενός ακόμα αισθητήρα θερμοκρασίας(Liquid / Vapor Line) και ενός ακόμα αισθητήρα πίεσης (Low / High Pressure).
- *CoolingChamber*: για μεγαλύτερη ασφάλεια σε επίπεδο επεξεργαστικών πόρων, θα γινόταν επιλογή του ESP32 αντί του ESP8266, ενώ ακόμα θα γινόταν επιλογή ενός αισθητήρα υγρασίας για τον ψυκτικό θάλαμο.
- *PCB Design*: όπως είναι αναμενόμενο ένας ηλεκτρονικό προϊόν στην τελική μορφή του πρέπει να σε μορφή τυπωμένης πλακέτας (PCB). Ένα θετικό της κατασκευής των έξυπνων συσκευών σε μορφή PCB είναι το μικρότερο footprint της τελικής πλακέτας, όπως επίσης και η χαμηλότερη τιμή του τελικού προϊόντος.
- *FreeRTOS*: για την αποδοτικότερη διαχείριση των πόρων των συσκευών θα γινόταν επιλογή του συγκεκριμένου λειτουργικού συστήματος, το οποίο είναι προσανατολισμένο για την εφαρμογή του σε μικροελεγκτές, ώστε να μπορεί να υπάρξει σωστός χρονοπρογραμματισμός των απαραίτητων διεργασιών.
- *IP68 Enclosure*: Το σύστημα βαθμονόμησης IP (Ingress Protection) είναι ένα διεθνές πρότυπο που δείχνει το επίπεδο αντοχής στη σκόνη και το νερό. Άρα ένα κουτί με το πρότυπο IP68 είναι απαραίτητο και για τις δύο έξυπνες συσκευές, για να καλύπτει το ηλεκτρονικό κύκλωμα το οποίο είναι εκτεθειμένο σε εξωτερικό χώρο (Compressor Unit) ή σε εσωτερικό που όμως και εκεί υπάρχει μεγάλη υγρασία (Cooling Chamber).

Cloud Infrastructure

- *Εισαγωγή ρόλων χρήστη*: ύπαρξη Logic/Register διαδικασίας για τους χρήστες με διαφορετικά interfaces για τον καθένα, π.χ. ο τεχνικός θα μπορούσε να έχει πρόσβαση και να βλέπει όλα τα μηχανήματα τα οποία είναι υπεύθυνος για την συντήρησής τους, ενώ ο ιδιοκτήτης να βλέπει μόνο τα δικά του.
- *Maps Feature*: να μπορεί να βλέπει ο τεχνικός και ο χρήστης ένας στίγμα στο χάρτη με την τοποθεσία του κάθε μηχανήματος.
- *Better UI/UX*: καλύτερος σχεδιασμός της web εφαρμογής (user friendly)
- *Επανασχεδιασμός της Βάσης Δεδομένων*.
- *Docker*: μετατροπή όλων των επιμέρους στοιχείων του κώδικα, δηλαδή frontend/backend/middleware/database σε docker images. Τα πλεονεκτήματα αυτής της επιλογής είναι: 1) ενσωμάτωση όλων των dependencies και βιβλιοθηκών σε ένα απομονωμένο περιβάλλον (container), 2) ευκολία ανάπτυξης και deploy της εφαρμογής σε διαφορετικά περιβάλλοντα (π.χ. Windows/Linux/macOS), 3) καλύτερη διαχείριση των πόρων.

- Kubernetes: διαχείρισης των εικόνων αυτών μέσω του Container Orchestration εργαλείου Kubernetes για την εύκολη επεκτασιμότητα της εφαρμογής.

Predictive Maintenance

Το endpoint αυτής της εφαρμογής θα ήταν η προγνωστική συντήρηση των μηχανημάτων αυτών μέσω της Τεχνητής Νοημοσύνης. Πάνω σε αυτό τον τομέα μελετήθηκαν αρκετά ερευνητικά άρθρα [\[20\]](#) [\[21\]](#) [\[22\]](#) [\[23\]](#) τα οποία αναδεικνύουν της σημαντικότητα της ανίχνευσης λαθών/ανωμαλιών σε αυτό τον τύπο μηχανημάτων. Η έγκαιρη ανίχνευση ανώμαλης συμπεριφοράς και ειδοποίηση αντίστοιχα στους τεχνικούς θα μπορούσε δυνητικά να προσφέρει:

1. Γρηγορότερη και καλύτερη εξυπηρέτηση στον πελάτη.
2. Καλύτερη σχέση μεταξύ πελατών – τεχνικών.
3. Χαμηλότερη ενεργειακή κατανάλωση.
4. Μείωση του εκτός χρόνου λειτουργίας των μηχανημάτων.
5. Αύξηση του μέσου χρόνου ζωής των μηχανημάτων.
6. Σημαντικά οικονομικά οφέλη που προκύπτουν από τα παραπάνω.

Τέλος για το Predictive Maintenance στα HVAC συστήματα θα είχε ερευνητικό ενδιαφέρον η χρήση της Τεχνητής Νοημοσύνης και ιδιαίτερα σε συνδυασμό με edge computing όπως γίνεται στο federated learning, δηλαδή οι αλγόριθμοι της τεχνητής νοημοσύνης να τρέχουν κατευθείαν στο Edge και όχι στο Cloud. Κάποιες πολύ δημοφιλείς βιβλιοθήκες που θα μπορούσαν να έχουν κάποιο αντίκτυπο για περεταίρω έρευνα είναι οι εξής: 1) TinyML, 2) Edge Impulse, 3) Tensorflow Lite.

Βιβλιογραφία

- [1] 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 and Tutorials*, 17(4), 2347–2376.
<https://doi.org/10.1109/comst.2015.2444095>
- [2] Mahmoud, R., Yousuf, T., Aloul, F., & Zualkernan, I. A. (2015). Internet of things (IoT) security: Current status, challenges and prospective measures. *IEEE*.
<https://doi.org/10.1109/icitst.2015.7412116>
- [3] Atzori, L., Iera, A., & Morabito, G. (2010). The Internet of Things: A survey. *Computer Networks*, 54(15), 2787–2805. <https://doi.org/10.1016/j.comnet.2010.05.010>
- [4] *IoT connected devices worldwide 2019-2030* | Statista. (2023). Statista.
<https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/>
- [5] Hassan, R., Qamar, F., Hasan, M. K., Aman, A. H. M., & Ahmed, A. S. (2020). Internet of Things and its Applications: A Comprehensive survey. *Symmetry*, 12(10), 1674.
<https://doi.org/10.3390/sym12101674>
- [6] Gharaibeh, A., Salahuddin, M. A., Hussini, S. J., Khreishah, A., Khalil, I., Guizani, M., & Al-Fuqaha, A. (2017). Smart Cities: A survey on data management, security, and enabling technologies. *IEEE Communications Surveys and Tutorials*, 19(4), 2456–2501.
<https://doi.org/10.1109/comst.2017.2736886>
- [7] Accenture. (2015). Industrial Internet of Things Will Boost Economic Growth but Greater Government and Business Action Needed to.
<https://newsroom.accenture.com/news/industrial-internet-of-things-will-boost-economic-growth-but-greater-government-and-business-action-needed-to-fulfill-its-potential-finds-accenture.htm>
- [8] Boyes, H., Hallaq, B., Cunningham, J., & Watson, T. (2018). The industrial internet of things (IIoT): An analysis framework. *Computers in Industry*, 101, 1–12.
<https://doi.org/10.1016/j.compind.2018.04.015>
- [9] *Industrial IoT | Digital Transformation | AWS IoT*. (n.d.). Amazon Web Services, Inc.
<https://aws.amazon.com/iot/solutions/industrial-iiot/>

- [10] Wikipedia contributors. (2023). Industrial internet of things. *Wikipedia*. https://en.wikipedia.org/wiki/Industrial_internet_of_things
- [11] Zhang, T., Li, Y., & Chen, C. P. (2021). Edge computing and its role in Industrial Internet: Methodologies, applications, and future directions. *Information Sciences*, 557, 34–65. <https://doi.org/10.1016/j.ins.2020.12.021>
- [12] AWS IoT Core. (n.d.). Amazon Web Services, Inc. <https://aws.amazon.com/iot-core/>
- [13] IoT Hub | Microsoft Azure. (n.d.). <https://azure.microsoft.com/en-us/products/iot-hub>
- [14] Cloud IoT Core | Google Cloud. (n.d.). Google Cloud. <https://cloud.google.com/iot-core>
- [15] Oracle Internet of Things Cloud Service - Get started. (2022). Oracle Help Center. <https://docs.oracle.com/en/cloud/paas/iot-cloud/index.html>
- [16] Zonta, T., Da Costa, C. A., Da Rosa Righi, R., De Lima, M. J., Da Trindade, E. S., & Li, G. (2020). Predictive maintenance in the industry 4.0: A systematic literature review. *ELSEVIER*, 150, 106889. <https://doi.org/10.1016/j.cie.2020.106889>
- [17] Gilchrist, A. (2016). *Industry 4.0: the industrial internet of things*. <https://cds.cern.ch/record/2204749/>
- [18] Wi-Fi CERTIFIED HaLow | Wi-Fi Alliance. (n.d.). <https://www.wi-fi.org/discover-wi-fi/wi-fi-certified-halow>
- [19] [Payload](#) | Sigfox build. (n.d.). <https://build.sigfox.com/payload>
- [20] Borda, D., Bergagio, M., Amerio, M., Masoero, M. C., Borchellini, R., & Papurello, D. (2023). Development of anomaly detectors for HVAC systems using machine learning. *Processes*, 11(2), 535. <https://doi.org/10.3390/pr11020535>
- [21] Munir, M., Erkel, S., Dengel, A., & Ahmed, S. (2017). Pattern-Based Contextual Anomaly Detection in HVAC Systems. *IEEE*. <https://doi.org/10.1109/icdmw.2017.150>

- [22] Bouabdallaoui, Y., Lafhaj, Z., Yim, P., Ducoulombier, L., & Bennadji, B. (2021). Predictive Maintenance in Building Facilities: A Machine Learning-Based approach. *Sensors*, 21(4), 1044. <https://doi.org/10.3390/s21041044>
- [23] Susto, G. A., Schirru, A., Pampuri, S., McLoone, S., & Beghi, A. (2015). Machine Learning for Predictive Maintenance: A multiple classifier approach. *IEEE Transactions on Industrial Informatics*, 11(3), 812–820. <https://doi.org/10.1109/tii.2014.2349359>
- [24] Nikos Rekkas. nikosrk/HVAC_IoT. GitHub. https://github.com/nikosrk/HVAC_IoT
- [25] Eclipse Mosquitto. (2018, January 8). Eclipse Mosquitto. <https://mosquitto.org/>
- [26] DS18B20. (2019). <https://www.analog.com/>. Retrieved September 28, 2023, from <https://www.analog.com/media/en/technical-documentation/data-sheets/ds18b20.pdf>
- [27] Santos, R., & Santos, R. (2021, September 30). ESP8266 NodeMCU Door Status Monitor with Email Notifications (IFTTT) | Random Nerd Tutorials. Random Nerd Tutorials. <https://randomnerdtutorials.com/door-status-monitor-using-the-esp8266/>
- [28] Thermistor. (2012, July 29). Adafruit Learning System. <https://learn.adafruit.com/thermistor/using-a-thermistor>
- [29] Xavier, D. (2019, July 6). DIY Home Energy Monitor: ESP32 + CT Sensors + Emonlib. Simply Explained. <https://simplyexplained.com/blog/Home-Energy-Monitor-ESP32-CT-Sensor-Emonlib/>