

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ



ΤΟΥ ΠΑΝΕΠΙΣΤΗΜΙΟΥ ΙΩΑΝΝΙΝΩΝ

**Πτυχιακή Εργασία του:
ΓΙΑΝΝΕΛΟΥ ΠΑΝΑΓΙΩΤΗ ΑΜ:15445
Μάρτιος 2024**

**Με τίτλο: Cloud Εφαρμογή συλλογής και εκτίμησης κόστους
ενεργειακών μετρήσεων και οι προοπτικές εξέλιξης της ως ένα
εμπορικό προϊόν.**

Επιβλέπων καθηγητή: ΤΖΕΡΙΕΣ ΜΠΕΣΑΡΑΤ

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω τον υπεύθυνο καθηγητή κ.Χρυσόστομο Στύλιο όπως επίσης και τον επιβλέποντα καθηγητή μου κ.Τζερρες Μπεσσαρατ για την πολύτιμη βοήθεια του κατά την διάρκεια διεκπεραίωσης της πτυχιακής μου εργασίας. Επίσης θα ήθελα να ευχαριστήσω των κ.Γρηγόρη Δουμένη που μέσω αυτού έκανα την αρχή στον κόσμο του IoT και έμαθα τις προοπτικές εξέλιξης στην καθημερινότητα μας Τέλος θα ήθελα να ευχαριστήσω την οικογένεια μου για την στήριξη καθόλη την διάρκεια των σπουδών μου.

ΠΕΡΙΕΧΟΜΕΝΑ

Κεφάλαιο 1. Internet of Things

- 1.1 Εισαγωγή
- 1.2 Ιστορία
- 1.3 Εξέλιξη
- 1.4 Παρούσα θέση στην Ελλάδα και τον κόσμο.
- 1.5 Πρακτικές Εφαρμογές

Κεφάλαιο 2. Εφαρμογές IoT (Internet of Things)

- 2.1 Αρχιτεκτονική μιας IoT εφαρμογής
- 2.2 Το IoT στην διαχείριση ενεργειακών πόρων

Κεφάλαιο 3. Περιγραφή Cloud εφαρμογής διαχείρισης ενέργειας σε κτιριακές εγκαταστάσεις-ENERCOST

- 3.1 Εισαγωγή στην εφαρμογή ENERCOST
- 3.2 Ανάλυση επίπεδων αρχιτεκτονικής για την εφαρμογή ENERCOST
- 3.3 Υπηρεσίες cloud και η σημασία τους
- 3.4 Ανάλυση του παρόχου cloud υπηρεσιών Amazon AWS

Κεφάλαιο 4. Ανάλυση Αρχιτεκτονικής και Τεχνολογιών της εφαρμογής ENERCOST

- 4.1 Ανάλυση αρχιτεκτονικής
- 4.2 Ανάλυση τεχνολογιών

Κεφάλαιο 5. Η εφαρμογή ENERCOST ως ένα εμπορικό προϊόν

- 5.1 Ανάλυση απαιτούμενων αλλαγών ή προσθήκες στην υπάρχων serverless αρχιτεκτονική.
- 5.2 Απαραίτητες προσθήκες σε physical και network επίπεδο.
- 5.3 Λοιπές ενέργειες για την επιτυχία του στην αγορά.

ΠΕΡΙΛΗΨΗ

Σκοπός της διπλωματικής, είναι η ανάπτυξη μιας ολοκληρωμένης full-stack εφαρμογής, η οποία έχει την δυνατότητα να λάβει ενεργειακές τιμές δεδομένων κατανάλωσης ηλεκτρικής ενέργειας, φυσικού αερίου και νερού και έπειτα να τα επικοινωνήσει σε μορφή γραφημάτων, ομαδοποιημένα στον χρήστη.

Η εφαρμογή αυτή (με το όνομα ENERCOST) χρησιμοποιεί την cloud υποδομή που παρέχει η Amazon AWS για το κομμάτι του back-end (διαχείρισης δεδομένων) και τις βιβλιοθήκες React-Redux για την υλοποίηση της front-end (διεπαφή χρήστη) εφαρμογής.

Πέραν αυτών, αναλύεται και ο τρόπος που έχει υλοποιηθεί όπως επίσης και για τις περαιτέρω ενέργειες και διαδικασίες που χρειάζονται ώστε να μπορεί εύκολα να μεταμορφωθεί απο μια απλή εφαρμογή διπλωματικής, σε ένα ολοκληρωμένο εμπορικό προϊόν.

ABSTRACT

The purpose of the diploma is the development of an integrated full-stack application, which has the ability to receive energy values of electricity, natural gas and water consumption data and then communicate them in the form of graphs, grouped to the user.

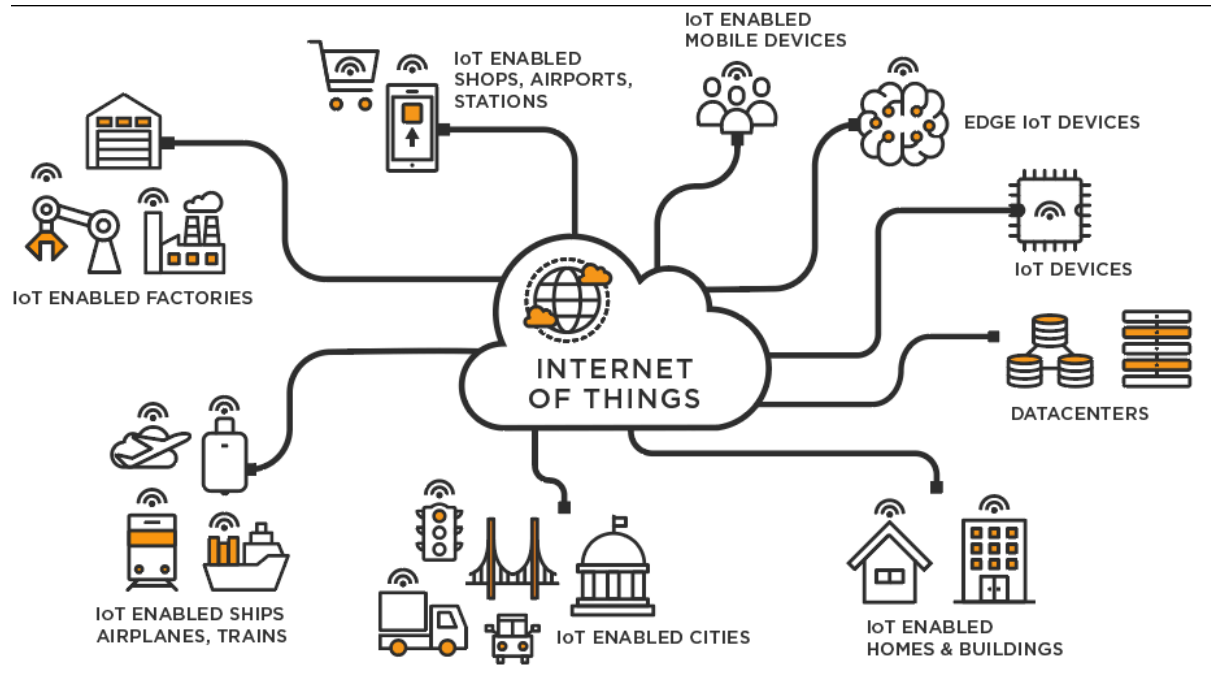
This application (named ENERCOST) uses the cloud infrastructure provided by Amazon AWS for the back-end (data management) part and the React-Redux libraries to implement the front-end (user interface) application.

In addition to these, the way it has been implemented is analyzed as well as the further actions and procedures needed so that it can be easily transformed from a simple diplomatic application to a complete commercial product.

ΔΗΛΩΣΗ ΠΝΕΥΜΑΤΙΚΗΣ ΙΔΙΟΚΤΗΣΙΑΣ

Η παρούσα εργασία αποτελεί προϊόν αποκλειστικά δικής μου προσπάθειας. Όλες οι πηγές που χρησιμοποιήθηκαν περιλαμβάνονται στη βιβλιογραφία και γίνεται ρητή αναφορά σε αυτές μέσα στο κείμενο που έχουν χρησιμοποιηθεί.

Ενότητα 1.1 Εισαγωγή στο IoT



Εικόνα 1.1 (Internet of Things)

Το Διαδίκτυο των Πραγμάτων (IoT) έχει φέρει επανάσταση στον τρόπο με τον οποίο αλληλεπιδρούμε με την τεχνολογία, συνδέοντας συσκευές και επιτρέποντας την απρόσκοπτη επικοινωνία σε παγκόσμια κλίμακα.

Για να κατανοήσουμε τη σημασία του IoT, είναι σημαντικό να εμβαθύνουμε στην ιστορία, την προέλευση και την εξέλιξή του. Σε αυτή την ενότητα θα συνεχίσουμε τη σύντομη ιστορία του IoT. Εξετάζοντας αυτά τα υποστηρικτικά σημεία, μπορούμε να αποκτήσουμε μια ολοκληρωμένη κατανόηση του πώς το IoT έχει μεταμορφώσει τον κόσμο γύρω μας.

Το Διαδίκτυο των Πραγμάτων (IoT) έχει αναδειχθεί γρήγορα ως μια τεχνολογία μετασχηματισμού που συνδέει καθημερινά αντικείμενα με το Διαδίκτυο, επιτρέποντάς τους να επικοινωνούν και να μοιράζονται δεδομένα.

Η προέλευση και η εξέλιξη του IoT εντοπίζονται στις αρχές της δεκαετίας του 1980, όταν εισήχθη για πρώτη φορά η έννοια της σύνδεσης συσκευών και της δυνατότητας επικοινωνίας τους. Ο όρος «Internet of Things» επινοήθηκε από τον Kevin Ashton το 1999, ο οποίος οραματίστηκε έναν κόσμο όπου τα αντικείμενα θα μπορούσαν να συλλέγουν και να μοιράζονται πληροφορίες χωρίς ανθρώπινη παρέμβαση. Έκτοτε, το IoT έχει εξελιχθεί σε διάφορα στάδια, με γνώμονα τις εξελίξεις στην τεχνολογία και την αυξανόμενη ανάγκη για συνδεσιμότητα. Τα πρώτα χρόνια, το IoT επικεντρώθηκε κυρίως στην επικοινωνία από μηχανή με μηχανή και σε βιομηχανικές εφαρμογές. Ωστόσο, με τον πολλαπλασιασμό των ασύρματων τεχνολογιών, όπως το Wifi το Bluetooth το LoRa και άλλες, το IoT επέκτεινε την εμβέλειά του στις καταναλωτικές συσκευές, οδηγώντας στην εμφάνιση έξυπνων κατοικιών και φορητών συσκευών.

Σήμερα, το IoT έχει διαπεράσει σχεδόν όλες τις βιομηχανικές ομάδες, συμπεριλαμβανομένης της υγειονομικής περίθαλψης, των μεταφορών, της γεωργίας, της κτηνοτροφίας και πολλές άλλες προσφέροντας τεράστιες ευκαιρίες για αποτελεσματικότητα, αυτοματοποίηση και βελτιωμένη λήψη αποφάσεων. Η εξέλιξη του IoT έχει ωθηθεί από τις εξελίξεις στην τεχνολογία αισθητήρων, το cloud computing και την ανάλυση δεδομένων, επιτρέποντας τη συλλογή, επεξεργασία και ανάλυση τεράστιων ποσοτήτων δεδομένων που παράγονται από συνδεδεμένες συσκευές. Καθώς το IoT συνεχίζει να εξελίσσεται, αναμένεται να διαδραματίσει κρίσιμο ρόλο στη διαμόρφωση του μέλλοντος της τεχνολογίας και της κοινωνίας, φέρνοντας επανάσταση στις βιομηχανίες και επιτρέποντας την ανάπτυξη έξυπνων πόλεων και αυτόνομων συστημάτων .

1.2 Ιστορική Αναδρομή του IoT

Η προέλευση του Διαδικτύου των Πραγμάτων (IoT) μπορεί να εντοπιστεί στις αρχές της δεκαετίας του 1980, όταν οι ερευνητές άρχισαν να εξερευνούν την ιδέα των διασυνδεδεμένων συσκευών. Ένας από τους ιδρυτές καινοτόμους στον τομέα του IoT είναι ο Mark Weiser, ο οποίος εισήγαγε την έννοια του «πανταχού υπολογισμού» στα τέλη της δεκαετίας του 1980. Ο Weiser οραματίστηκε ένα μέλλον όπου οι υπολογιστές θα ενσωματώνονταν απρόσκοπτα στο φυσικό περιβάλλον, επιτρέποντας στις συσκευές να επικοινωνούν και να συνεργάζονται χωρίς ανθρώπινη παρέμβαση. Οι ιδέες του έθεσαν τα θεμέλια για την ανάπτυξη των τεχνολογιών IoT. Η πρώτη "έξυπνη" συσκευή σας πάει πίσω στις αρχές της δεκαετίας του 1980 όταν ο Ντέιβιντ Νίκολς, μεταπτυχιακός φοιτητής στο τμήμα πληροφορικής του Πανεπιστημίου Carnegie Mellon, έγραψε σε μερικούς φίλους την ιδέα του να παρακολουθεί το περιεχόμενο της αυτόματης μηχανής πώλησης αναψυκτικών από απόσταση.

Όταν κάποιος αγοράζε μια κόκα κόλα, μια ενδεικτική λυχνία για την αντίστοιχη στήλη θα αναβοσβήνει για μερικά δευτερόλεπτα πριν σβήσει ξανά. Όταν μια στήλη ήταν άδεια, το φως παραμένει αναμμένο μέχρι να αντικατασταθούν τα αναψυκτικά. Τοποθετήθηκε μια πλακέτα που ανιχνεύει την κατάσταση καθεμιάς από τις ενδεικτικές λυχνίες. Η πλακέτα ήταν συνδεδεμένη στον κύριο υπολογιστή του τμήματος, ο οποίος ήταν συνδεδεμένος με το ARPANET (πρόδρομο του σημερινού Διαδικτύου). Έπειτα με ένα πρόγραμμα ελέγχεται η κατάσταση του φωτός κάθε στήλης μερικές φορές το δευτερόλεπτο. Εάν μια λυχνία μεταβαλλόταν από το off στο on αλλά μετά έσβησε ξανά λίγα δευτερόλεπτα αργότερα, ήξερε ότι είχε αγοραστεί μια Coca-Cola. Εάν η λυχνία παραμένει αναμμένη περισσότερο από πέντε δευτερόλεπτα, υπέθεσε ότι η στήλη ήταν άδεια. Επίσης επιτρέπει σε οποιονδήποτε ήταν σε υπολογιστή συνδεδεμένο στο ARPANET να έχει πρόσβαση σε πληροφορίες σχετικά με το μηχάνημα.

Ύστερα από αυτό υπήρξαν κι άλλες "έξυπνες" συσκευές 'όπως Ο John Romkey που δημιούργησε μια τοστιέρα που μπορούσε να λειτουργήσει μέσω Διαδικτύου. Συνέδεσε την τοστιέρα με TCP/IP και είχε δύο λειτουργίες: ενεργοποίηση και απενεργοποίηση. Παρόλα αυτά, παρείχε ένα από τα πρώτα συγκεκριμένα παραδείγματα ότι το διαδίκτυο θα μπορούσε να χρησιμοποιηθεί για τη σύνδεση καθημερινών πραγμάτων.

Επίσης, στις αρχές της δεκαετίας του 1990, η περίφημη καφετιέρα Trojan Room στο Πανεπιστήμιο του Κέιμπριτζ έδειξε την πρώτη χρήση μιας δημόσιας κάμερας web. Αντί να ανεβοκατεβενουν πολλές σκάλες για να δουν αν ο καφές τους είχε τελειώσει, οι ερευνητές έστησαν μια κάμερα που στέλνει εικόνες από το δοχείο του καφέ στους υπολογιστές τους μέσω του Διαδικτύου.

Όμως ο όρος "Internet of Things" επινοήθηκε από τον Kevin Ashton, έναν Βρετανό πρωτοπόρο της τεχνολογίας, το 1999. Ο Ashton οραματίστηκε έναν κόσμο όπου τα καθημερινά αντικείμενα θα μπορούσαν να συνδεθούν στο διαδίκτυο και να

επικοινωνούν μεταξύ τους, οδηγώντας σε μια πιο αποτελεσματική και διασυνδεδεμένη κοινωνία. Εμπνευσμένος από τα μικροσίπ, τα οποία είχαν γίνει όλο και περισσότερο μέρος των πιστωτικών καρτών στην Ευρώπη στα μέσα της δεκαετίας του 1990, ο Kevin Ashton είχε την ιδέα να ενσωματώσει μια κεραία RFID απο πιστωτικη καρτα στα ράφια κραγιόν, για να βοηθήσει τους διευθυντές καταστημάτων να παρακολουθούν την ποσότητα του κραγιόν που ήταν ακόμα διαθέσιμο εκεί. Ήταν υπό την πεποίθησή του ότι εάν ένα ασύρματο δίκτυο μπορούσε να συλλέξει δεδομένα σε μια κάρτα, θα μπορούσε επίσης να συλλέξει δεδομένα από ένα σιπ σε μια συσκευασία κραγιόν και να ειδοποιήσει τη διαχείριση για το αν ήταν ή όχι στα ράφια.

Αυτοί και άλλοι καινοτόμοι έπαιξαν καθοριστικό ρόλο στη διαμόρφωση της πρώιμης ανάπτυξης των τεχνολογιών IoT και άνοιξαν το δρόμο για την ταχεία ανάπτυξη και την ευρεία υιοθέτησή τους τα επόμενα χρόνια.

1.3 Η Εξέλιξη του IoT

Η εξέλιξη του Διαδικτύου των Πραγμάτων (IoT) έχει γνωρίσει σημαντικό ορόσημο και τεχνολογικές προόδους τα τελευταία χρόνια. Η εμφάνιση του IoT μπορεί να εντοπιστεί στις αρχές της δεκαετίας του 2000, όταν οι ερευνητές άρχισαν να εξερευνούν την ιδέα της σύνδεσης καθημερινών αντικειμένων με το διαδίκτυο. Από τότε, υπήρξαν πολλά βασικά ορόσημα που έχουν διαμορφώσει την εξέλιξη του IoT. Ένα από τα σημαντικότερα ορόσημα ήταν η ανάπτυξη τεχνολογιών ασύρματης δικτύωσης, όπως το Bluetooth και το Wi-Fi, που επέτρεψαν την απρόσκοπτη συνδεσιμότητα μεταξύ συσκευών.

Αυτές οι εξελίξεις άνοιξαν το δρόμο για τον πολλαπλασιασμό των συσκευών IoT και τη δημιουργία ενός δικτύου διασυνδεδεμένων αντικειμένων.

Ένα άλλο σημαντικό ορόσημο ήταν η εισαγωγή του cloud computing και των big data analytics, που παρείχαν την υποδομή και τα εργαλεία που απαιτούνται για την επεξεργασία και την ανάλυση των τεράστιων ποσοτήτων δεδομένων που παράγονται από συσκευές IoT. Αυτό επέτρεψε την ανάλυση δεδομένων σε πραγματικό χρόνο και τη λήψη αποφάσεων, οδηγώντας σε βελτιωμένη απόδοση και παραγωγικότητα σε διάφορους κλάδους.

Επιπλέον, η έλευση της μηχανικής μάθησης και της τεχνητής νοημοσύνης (AI) έχει φέρει επανάσταση στις δυνατότητες των συσκευών IoT. Οι αλγόριθμοι AI μπορούν πλέον να αναλύουν και να ερμηνεύουν δεδομένα από αισθητήρες IoT, επιτρέποντας στις συσκευές να μαθαίνουν και να προσαρμόζονται στις μεταβαλλόμενες συνθήκες αυτόνομα. Αυτό έχει ανοίξει νέες δυνατότητες σε τομείς όπως η προγνωστική συντήρηση, τα έξυπνα σπίτια και τα αυτόνομα οχήματα. Συμπερασματικά, η εξέλιξη του IoT χαρακτηρίστηκε από σημαντικά ορόσημα και τεχνολογικές προόδους, συμπεριλαμβανομένης της ανάπτυξης τεχνολογιών ασύρματης δικτύωσης, της εισαγωγής του cloud computing και της ανάλυσης μεγάλων δεδομένων και της ενσωμάτωσης αλγορίθμων μηχανικής μάθησης και τεχνητής νοημοσύνης.

Αυτές οι εξελίξεις έχουν ωθήσει τη βιομηχανία του IoT μπροστά και έχουν ανοίξει το δρόμο για ένα μέλλον όπου οι συνδεδεμένες συσκευές παίζουν κεντρικό ρόλο στην καθημερινή μας ζωή.

Το Διαδίκτυο των Πραγμάτων (IoT) έχει φέρει επανάσταση στον τρόπο αλληλεπίδρασης με την τεχνολογία, μετατρέποντας τα τοπικά δίκτυα σε έναν τεράστιο, διασυνδεδεμένο ιστό που εκτείνεται σε όλο τον κόσμο. Αυτή η παγκόσμια επίδραση του IoT είναι εμφανής σε διάφορους τομείς. Ο πολλαπλασιασμός των συσκευών IoT έχει οδηγήσει σε ένα άνευ προηγουμένου επίπεδο συνδεσιμότητας, επιτρέποντας την απρόσκοπτη επικοινωνία και ανταλλαγή δεδομένων μεταξύ συσκευών, ανεξάρτητα από τη γεωγραφική τους θέση. Αυτή η διασύνδεση όχι μόνο ενίσχυσε την αποτελεσματικότητα διαφόρων συστημάτων, αλλά διευκόλυνε επίσης τη συλλογή και ανάλυση τεράστιων ποσοτήτων δεδομένων, οδηγώντας σε μεγάλη βελτίωση στην λήψη αποφάσεων αλλά και τον έλεγχο των συσκευών αυτών.

Τις τελευταίες δεκαετίες, οι συσκευές IoT έχουν εξελιχθεί με γρήγορους ρυθμούς, με αποτέλεσμα εκατομμύρια συνδεδεμένες συσκευές να φέρουν επανάσταση σε διάφορους τομείς.

Από απλούς αισθητήρες, η τεχνολογία IoT προχώρησε, οδηγώντας σε έξυπνες συσκευές για σπίτια, βιομηχανίες και πολλά άλλα. Οι πρώτες συσκευές συνέλεξαν δεδομένα εξ αποστάσεως, αλλά οι εξελίξεις στους υπολογιστές και την επικοινωνία επέτρεψαν την αλληλεπίδραση.

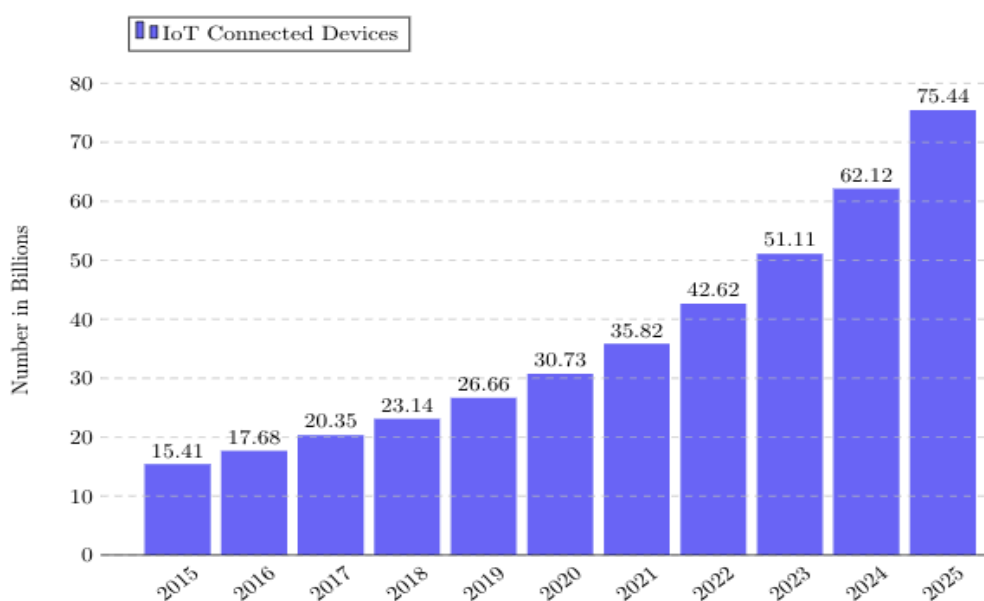
Η ασύρματη τεχνολογία χαμηλής κατανάλωσης, όπως το Zigbee και το BLE, LoRa επέτρεψαν την χρήση έξυπνων συσκευών σε απομακρυσμένα μέρη όπως και επίσης έξυπνες συσκευές με μπαταρία . Η ραγδαία ανάπτυξη της τεχνητής νοημοσύνης ενίσχυσε την "ευφυΐα" των έξυπνων συσκευών

Απ την άλλη το Edge computing ενίσχυσε την επεξεργασία δεδομένων σε πραγματικό χρόνο για αυτόνομα οχήματα και βιομηχανικά συστήματα.

Επίσης καθοριστικό ρόλο για την παγκόσμια ανάπτυξη του Internet of Things, έπαιξε η απελευθέρωση των διευθύνσεων IPv6 το 2011. Αυτή η αλλαγή κατέρριψε τα όρια, αφού πλέον δεν υπάρχει περιορισμός στον αριθμό των διευθύνσεων. Το νέο πρωτόκολλο επιτρέπει τη χρήση 2^{128} διευθύνσεων.

Ή αλλιώς, 666 δισεκατομμύρια διευθύνσεις ανά κόκκο άμμου στο φλοιό της Γης, σε βάθος ενός μιλίου.

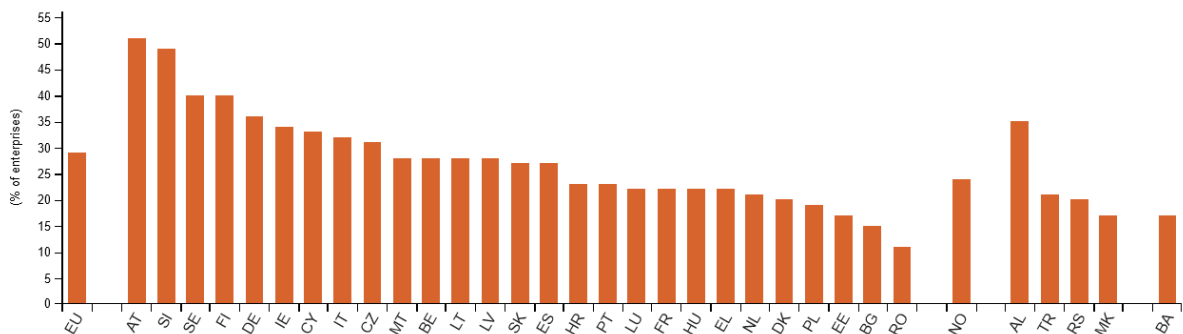
Το Internet of Things συνέχισε να αναπτύσσεται ώσπου το 2015 ονομάστηκε η "Η χρονιά του Internet of Things" καθώς σύμφωνα με την ιστοσελίδα Gartner ήταν η πιο αναπτυσσόμενη τεχνολογία σκαρφαλώνοντας από την 12η θέση που κατείχε το 2011. Στο παρακάτω διάγραμμα, έπειτα από έρευνα στο statista αναφέρεται ότι περίπου 15.4 δισεκατομμύρια συσκευές ήταν συνδεδεμένες το 2015 και αναμένεται ότι ο αριθμός της θα αυξηθεί περίπου στο 75.5 δισεκατομμύρια μέχρι το 2025.



Εικόνα 1.2: (IoT)συνδεδεμένες συσκευές από το 2015 έως το 2025 (σε δις)

Κεφάλαιο 1.4 Η παρούσα θέση του IoT στην Ελλάδα και τον κόσμο.

Παγκοσμίως, κορυφαίες χώρες έχουν αναγνωρίσει την εξέλιξη του IoT ως σημαντικό παράγοντα για την οικονομική ανάπτυξη και τη βελτίωση της ποιότητας ζωής. Η Αμερική, η Κίνα, η Ευρωπαϊκή Ένωση, η Νότια Κορέα, η Ιαπωνία, η Σιγκαπούρη (που θεωρείται πλέον η πιο "έξυπνη χώρα") και πολλές άλλες χώρες επενδύουν στην έρευνα, την ανάπτυξη και την υιοθέτηση της τεχνολογίας αυτής. Εκεί όπου οι εταιρείες εκμεταλλεύονται την ισχύ του IoT για να προσφέρουν καινοτόμες λύσεις, οι κυβερνήσεις διαμορφώνουν πολιτικές πλαισίων για την ασφάλεια, την ιδιωτικότητα και την υποστήριξη της ανάπτυξης του IoT. Στην παρακάτω εικόνα βλέπουμε το ποσοστό υιοθέτησης IoT από μεγάλες επιχειρήσεις στην Ευρωπαϊκή Ένωση το 2021.



Εικόνα 1.3 Ποσοστά χρήσης IoT τεχνολογιών σε επιχειρήσεις ανα ευρωπαϊκή χώρα

Στην Ελλάδα, αν και καθυστερημένα σε σχέση με άλλες ευρωπαϊκές χώρες η εφαρμογή του IoT έχει αρχίσει να παίζει σημαντικό ρόλο έχοντας φτάσει το ποσοστό του 22% των επιχειρήσεων να χρησιμοποιεί IoT τεχνολογίες.

Πλέον όλο και περισσότεροι δήμοι επιλέγουν στα μελλοντικά σας σχέδια λύσεις με έξυπνες συσκευές. Ένα τρανό παράδειγμα είναι ο δήμος Τρικάλων, που έχουν εγκαταστήσει αρκετές έξυπνες συσκευές, που είναι σε θέση να παρακολουθησουν και να διαχειριστούν απομακρυσμένα από ,φωτεινους σηματοδοτες , τον φωτισμο της πόλης, τα απορρίμματα ως και τηλεφροντίδα σε επιλεγμένα άτομα που έχουν άμεση ανάγκη. όπως επίσης και ιδιωτικο.

Επιπλέον, η Ευρωπαϊκή Ένωση έχει αναγνωρίσει τη σημασία του IoT και υποστηρίζει ευρηματικές πρωτοβουλίες μέσω ερευνητικών προγραμμάτων και χρηματοδοτικών μηχανισμών.

Κεφάλαιο 1.5 Πρακτικές εφαρμογές του IoT

Από τη βιομηχανία μέχρι την υγεία, από τις έξυπνες πόλεις μέχρι την διαχείριση ενέργειας και τη γεωργία, το IoT διαδραματίζει ζωτικό ρόλο στη βελτίωση της λειτουργίας, της αποδοτικότητας και της εμπειρίας των χρηστών σε κάθε από αυτούς τους τομείς. Με κάθε έναν να αξιοποιεί την ισχύ του IoT για να βελτιώσει τη λειτουργία, την αποδοτικότητα και την εμπειρία των χρηστών του.

Βιομηχανία: Στη βιομηχανία, το IoT διευκολύνει την επιτήρηση και τη συντήρηση μηχανημάτων και εξοπλισμού. Μέσω αισθητήρων και δικτύων, οι βιομηχανικές μονάδες μπορούν να παρακολουθούν συνθήκες και να εκτελούν προληπτική συντήρηση, μείωση προσωπικού, ενίσχυση της ασφάλειας, ενώ παράλληλα μειώνοντας τις διακοπές στην παραγωγή και αυξάνοντας την παραγωγικότητα.

Υγεία: Στον τομέα της υγείας, το IoT δημιουργεί εφαρμογές για την παρακολούθηση και τη διαχείριση της υγείας. Φορετά αισθητήρια, όπως τα έξυπνα ρολόγια και οι συσκευές παρακολούθησης, παρέχουν στους ασθενείς και τους επαγγελματίες υγείας πραγματικού χρόνου πληροφορίες για την κατάσταση της υγείας μέσω διαδραστικών εφαρμογών.

Έξυπνες Πόλεις: Στο πλαίσιο των έξυπνων πόλεων, το IoT ενσωματώνεται στην υποδομή για την παρακολούθηση και τη διαχείριση των δημόσιων υπηρεσιών. Από την παρακολούθηση της κυκλοφορίας, τον φωτισμό, την ασφάλεια, την στάθμευση μέχρι την έγκαιρη παροχή δεδομένων για τη διαχείριση των απορριμμάτων, το IoT συνεισφέρει στην αποτελεσματικότητα των αστικών περιοχών.

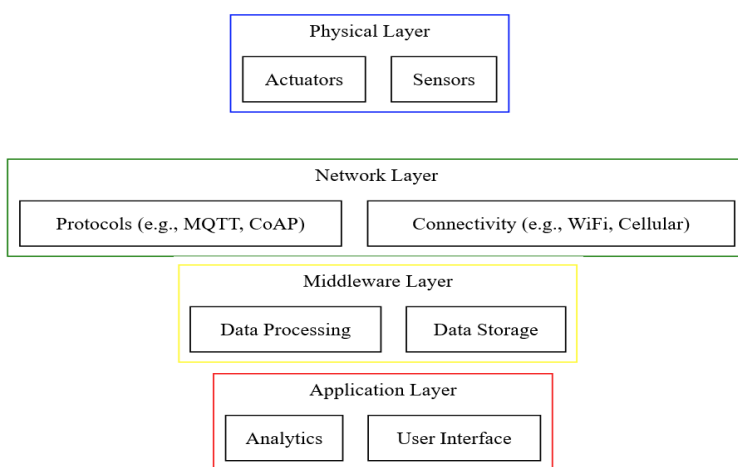
Κτίρια Έξυπνης Τεχνολογίας: Τα έξυπνα κτίρια χρησιμοποιούν το IoT για την αυτοματοποίηση και τη βελτιστοποίηση της λειτουργίας τους. Κάποια από τα πιθανά σενάρια απαρτίζουν αισθητήρες να ελέγχουν τη φωτεινότητα, συστήματα ασφαλείας έλεγχος θερμοκρασίας και κατανάλωσης και πολλά άλλα. Τα δεδομένα από αυτούς, αξιοποιούνται για τη μείωση των ενεργειακών καταναλώσεων, πιθανές μελέτες και έρευνες και σημαντικότερο, τη βελτίωση της άνεσης των κατοίκων.

Γεωργια-Κτηνοτροφία: Στον τομέα της γεωργίας και της κτηνοτροφίας, το IoT επιτρέπει την παρακολούθηση και τον έλεγχο των αγροτικών εκτάσεων με ακρίβεια και αποτελεσματικότητα. Αισθητήρες ανιχνεύουν δεδομένα σχετικά με την υγρασία του εδάφους, την θερμοκρασία και την ποιότητα του αέρα, παρέχοντας πολύτιμες πληροφορίες για την βέλτιστη ανάπτυξη των καλλιεργειών. Έτσι και στην κτηνοτροφία, συμβάλλουν στην βελτίωση της ευημερίας των ζώων και τη διαχείριση των αγροκτημάτων. Αισθητήρες που φοριούνται καταγράφουν την υγεία των ζώων, τη συμπεριφορά τους και τις συνθήκες του περιβάλλοντος, επιτρέποντας στους αγρότες να λαμβάνουν δεδομένα και να παίρνουν αποφάσεις για τη διατροφή, τις παρεμβάσεις υγείας και τη συνολική διαχείριση της φάρμας.

Κεφάλαιο 2.1 Αρχιτεκτονική μιας IoT Εφαρμογής

Έχοντας αναλύσει την ευρεία υιοθέτηση του Internet of Things (IoT) σε παγκόσμιο επίπεδο, αξίζει να εξετάσουμε πιο αναλυτικά τα συστατικά ενός πλήρους συστήματος IoT και πώς αυτά συνεργάζονται για να δημιουργήσουν ολοκληρωμένες εφαρμογές.

Μια πλήρης λύση Internet of Things (IoT) αποτελείται από ένα σύνολο συνδεδεμένων στοιχείων και τεχνολογιών που συνεργάζονται για την κατασκευή και λειτουργία μιας ολοκληρωμένης εφαρμογής. Οι βασικές συνιστώσες ενός πλήρους IoT συστήματος περιλαμβάνουν:



Εικόνα 2.1 Επίπεδα μιας IoT εφαρμογής

- **1.1 Αισθητήρες (Sensors):** Οι αισθητήρες αποτελούν την κινητήρια δύναμη του Internet of Things, ανιχνεύοντας τις φυσικές παραμέτρους του περιβάλλοντος και μετατρέποντας τις σε ψηφιακά δεδομένα. Αυτοί οι αισθητήρες μπορούν να περιλαμβάνουν αισθητήρες θερμοκρασίας, υγρασίας, πίεσης, φωτεινότητας, κίνησης και πολλούς άλλους, και συλλέγουν δεδομένα από το περιβάλλον τους.
- **1.2 Συσκευές Επεξεργασίας Δεδομένων (Data Processing Devices):** Ακολουθώντας τη συλλογή των δεδομένων από τους αισθητήρες, η επόμενη διαδικασία περιλαμβάνει την επεξεργασία αυτών των δεδομένων. Σε αυτή την υποενότητα, οι συσκευές επεξεργασίας αναλαμβάνουν τη μετατροπή και την ανάλυση των ψηφιακών δεδομένων από τους αισθητήρες. Αυτές οι συσκευές μπορούν να είναι μικροελεγκτές, μικροεπεξεργαστές, ή ακόμη και ειδικές μονάδες επεξεργασίας που εκτελούν αλγορίθμους για την εξαγωγή σημαντικών πληροφοριών από τα δεδομένα των αισθητήρων.
Συνήθως απαρτίζονται από ενσωματωμένες πλακέτες που εξειδικεύονται για IoT οικοσυστήματα, με γνώμονες την χαμηλή κατανάλωση, το κόστος, την

υποστήριξη ασύρματο πρωτοκολλων επικοινωνιας όπως ZigBee, wifi, LoRa, NB-IOT και την λήψη-επεξεργασία δεδομένων από αισθητήρες.

- **2 Δίκτυα Επικοινωνίας (Communication Networks):** Τα δίκτυα επιτρέπουν τη μετάδοση των δεδομένων από τους αισθητήρες στην αναλυτική συσκευή ή τον κεντρικό διακομιστή. Τα δίκτυα μπορούν να είναι ενσύρματα (όπως Ethernet) ή ασύρματα (όπως Wi-Fi, Zigbee, LoRa, 5G). Οι συσκευές που απαρτίζουν το συγκεκριμένο κομμάτι μπορεί να διαφέρουν ανάλογα την χρήση.
Παράδειγμα αν οι αισθητήρες χρησιμοποιούν wifi θα χρειαστεί ένα router/modem σε ένα σημείο και αναμεταδότες (κόμβοι) κοντά στους αισθητήρες. Εάν από την άλλη οι αισθητήρες χρησιμοποιούν LoRa πρωτόκολλο χρειαζόμαστε ένα modem/router και κόμβους (gateways) κοντά στους αισθητήρες.
- **3.Επεξεργασία και Ανάλυση Δεδομένων (Data Processing and Analysis):** Το επίπεδο επεξεργασίας δεδομένων της αρχιτεκτονικής IoT αναφέρεται στα στοιχεία λογισμικού και υλικού που είναι υπεύθυνα για τη συλλογή, την ανάλυση και την ερμηνεία δεδομένων από συσκευές IoT. Αυτό το επίπεδο είναι υπεύθυνο για τη λήψη ακατέργαστων δεδομένων από τις συσκευές, την επεξεργασία τους και τη διάθεση τους για περαιτέρω ανάλυση ή δράση. Το επίπεδο επεξεργασίας δεδομένων περιλαμβάνει μια ποικιλία τεχνολογιών και εργαλείων, όπως συστήματα διαχείρισης δεδομένων, πλατφόρμες ανάλυσης και αλγόριθμους μηχανικής μάθησης . Αυτά τα εργαλεία χρησιμοποιούνται για την εξαγωγή σημαντικών πληροφοριών από τα δεδομένα και τη λήψη αποφάσεων με βάση αυτά τα δεδομένα. Παράδειγμα τεχνολογίας που χρησιμοποιείται στο επίπεδο επεξεργασίας δεδομένων είναι μια μεγάλη βάση δεδομένων, η οποία είναι μια "κεντρική αποθήκη" για την αποθήκευση ακατέργαστων δεδομένων από συσκευές IoT.
- **4.Εφαρμογές και Χρήστες (Applications and Users):** Το επίπεδο εφαρμογής και τελευταίο της αρχιτεκτονικής IoT είναι το ανώτερο επίπεδο που αλληλεπιδρά άμεσα με τον τελικό χρήστη. Είναι υπεύθυνο για την παροχή φιλικών προς το χρήστη διεπαφών και λειτουργιών που επιτρέπουν στους χρήστες να έχουν πρόσβαση και να ελέγχουν συσκευές IoT. Αυτό το επίπεδο περιλαμβάνει διάφορα λογισμικά και εφαρμογές όπως εφαρμογές για κινητά, ιστοσελίδες και άλλες διεπαφές χρήστη που έχουν σχεδιαστεί για να αλληλεπιδρούν με την υποκείμενη υποδομή IoT . Περιλαμβάνει επίσης υπηρεσίες ενδιάμεσου λογισμικού που επιτρέπουν σε διαφορετικές συσκευές

και συστήματα IoT να επικοινωνούν και να μοιράζονται δεδομένα απρόσκοπτα. Το επίπεδο εφαρμογής περιλαμβάνει επίσης δυνατότητες ανάλυσης και επεξεργασίας που επιτρέπουν στα δεδομένα να αναλύονται και να μετατρέπονται σε σημαντικές πληροφορίες. Αυτό μπορεί να περιλαμβάνει αλγόριθμους μηχανικής εκμάθησης, εργαλεία οπτικοποίησης δεδομένων και άλλες προηγμένες δυνατότητες ανάλυσης.

Ο συνδυασμός των παραπάνω αποτελεί ένα πλήρες IoT σύστημα που μπορεί να προσφέρει ποικίλες λύσεις, από έξυπνες πόλεις και βιομηχανική αυτοματοποίηση έως και εξειδικευμένες εφαρμογές υγείας, οικονομίας και γεωργίας.

Κεφάλαιο 2.2 Το IoT στην διαχείριση ενεργειακών πόρων.

Στον κόσμο του Διαδικτύου των Πραγμάτων (IoT), η ενεργειακή διαχείριση αποκτά νέα διάσταση και ευρύτερη σημασία. Η εξαιρετική εξέλιξη των συνδεδεμένων συσκευών και των αισθητήρων, σε συνδυασμό με τη δυνατότητα ανταλλαγής δεδομένων μεταξύ αυτών, έχει ανοίξει νέους ορίζοντες για την αποτελεσματική διαχείριση των ενεργειακών πόρων.

Ο κόσμος μας αντιμετωπίζει σημαντικές προκλήσεις στον τομέα της ενεργειακής αποδοτικότητας και αειφορίας. Η αυξανόμενη κατανάλωση ενέργειας, η εξάντληση των παραδοσιακών πηγών και οι επιπτώσεις της ατμοσφαιρικής ρύπανσης απαιτούν άμεσες και καινοτόμες προσεγγίσεις. Ταυτόχρονα, η διαχείριση των πόρων όπως το ηλεκτρικό ρεύμα, το φυσικό αέριο και το νερό απαιτεί αποδοτικότερες πρακτικές για την προστασία τους από την υπερβολική κατανάλωση και την υποβάθμιση.

Σε αυτό το πλαίσιο, η τεχνολογία IoT αναδεικνύει νέες δυνατότητες για την παρακολούθηση, τον έλεγχο και τη βελτιστοποίηση της ενεργειακής κατανάλωσης και της αξιοποίησης των πόρων. Εμπλουτισμένη με αισθητήρες και δυνατότητες ανάλυσης δεδομένων, το IoT παρέχει λύσεις που προσφέρουν αποδοτική και βιώσιμη διαχείριση των ενεργειακών πόρων, βοηθώντας ταυτόχρονα να εξοικονομηθούν πόροι και χρήματα.

Για αυτό τον λόγο, πολλές εταιρίες έχουν εστιάσει και επενδύσει στο συγκεκριμένο τομέα

Ο τρόπος που το IoT επιδρά σε αυτόν τον τομέα, είναι συνήθως με την δημιουργία "έξυπνων" μετρητών, που λαμβάνουν δεδομένα για τους συγκεκριμένους πόρους έχοντας παράλληλα πρόσβαση στο διαδίκτυο ή κάποιο τρόπο μετάδοσης αυτών σε άλλες συσκευές.

Μεμονωμένα προϊόντα όπως "Έξυπνοι μετρητές κατανάλωσης ηλεκτρικού ρεύματος, νερού, καυσίμου αλλά και ολοκληρωμένες λύσεις παρακολούθησης ενός ευρύτερου ενεργειακού οικοσυστήματος. Παρακάτω αναφέρονται μερικές εταιρίες που δραστηριοποιούνται ενεργά στον τομέα αυτό.

- **ABB:** Η ευρέως γνωστή εταιρία προϊόντων ηλεκτρικής τεχνολογίας όπως ηλεκτρολογικό υλικό και συστήματα ελέγχου βιομηχανικών μηχανημάτων. Τον τελευταίο καιρό όμως, έχοντας επενδύσει σημαντικά στον τομέα της διαχείρισης πόρων με τεχνολογίες IoT στην βιομηχανία. Προσφέρει ολοκληρωμένες επαγγελματικές λύσεις παρακολούθησης και διαχείρισης ενεργειακών πόρων. Μάλιστα, λόγω της γρήγορης εξέλιξης και της βαρύτητας του ονόματος της στον τομέα της βιομηχανίας, έχει φτάσει να είναι από τις πιο γνωστές στον κλάδο αυτό.

- **Honeywell** : Άλλη μια μεγάλη αμερικανική εταιρεία που οι τομείς ενασχόλησης της απαρτίζονται από όλων των ειδών αυτοματισμούς εστιασμένους σε βιομηχανικές εγκαταστάσεις και κτιριακές υποδομές.
Η ομπρέλα με τα προϊόντα πάνω στην ανάπτυξη "έξυπνων" λύσεων πάνω στην διαχείριση κτιρίων αποτελούνται από διαχείριση πυρασφάλειας, access control, πίνακες ελέγχου με εστιασμένο λογισμικό και πολλών ειδών αισθητήρες.

Κεφάλαιο 3.1 Εισαγωγή στην εφαρμογή ENERCOST

Στη συνέχεια της ανάλυσης μας για το πώς το Διαδίκτυο των Πραγμάτων (IoT) μεταμορφώνει τον τρόπο που διαχειριζόμαστε τους ενεργειακούς πόρους, προκύπτει μια προκλητική ερώτηση: πώς ακριβώς μπορεί να εφαρμοστεί αυτή η τεχνολογία για να παρέχει άμεσα οφέλη στην καθημερινή ζωή και δραστηριότητα μας; Αυτό το ερώτημα αναδεικνύει την ανάγκη για συγκεκριμένες εφαρμογές που αξιοποιούν το IoT για την παρακολούθηση, τον υπολογισμό και την αποτίμηση της κατανάλωσης και της απόδοσης των πόρων μας.

Στο πλαίσιο αυτό, παρουσιάζεται μια συγκεκριμένη εφαρμογή που αναπτύχθηκε, και έχει να κάνει με τον υπολογισμό της κατανάλωσης ενέργειας, φυσικού αερίου και ύδατος σε πραγματικό χρόνο για κτίρια αλλά και αποθήκευση χρονικών δεδομένων για περαιτέρω ανάλυση.

Αυτή η εφαρμογή επιδιώκει να παρέχει ένα ενεργειακό προφίλ ενός κτιρίου ή μιας μονάδας, μετατρέποντας τα δεδομένα από αισθητήρες και μετρητές σε ευανάγνωστες μετρήσεις και στατιστικές.

Επίσης επιτρέπει την παρακολούθηση πραγματικής καταναλώσεως και την απόδοση των πόρων, βοηθώντας τον χρήστη να προσαρμόσει τις συνήθειές του για μεγαλύτερη αποδοτικότητα. Είναι ένα σαφές παράδειγμα πώς η τεχνολογία του IoT μεταφράζεται σε πρακτικά οφέλη που απευθύνονται απευθείας στον καθημερινό μας τρόπο ζωής και επαγγελματική δραστηριότητα.

Πέρα από την απλή καταγραφή της κατανάλωσης, αυτή η εφαρμογή προσφέρει πολλαπλά οφέλη που αποκαλύπτουν το δυναμικό της τεχνολογίας του IoT στον τομέα της ενεργειακής διαχείρισης. Με τη δυνατότητα πραγματικού χρόνου παρακολούθησης, μπορείτε να αντιλαμβάνεστε την αντίκτυπο των ενεργειακών συνηθειών σας στην κατανάλωση. Αυτό δημιουργεί μια νέα επίγνωση και ευαισθητοποίηση σχετικά με την απόδοση των πόρων και την περιβαλλοντική επίπτωση.

Είναι σημαντικό να αναφέρουμε πως αυτή η εφαρμογή δεν προσφέρει μόνο ευκαιρίες για ενεργειακή αποδοτικότητα, αλλά επιφέρει και οικονομικά οφέλη. Η δυνατότητα να παρακολουθεί ο χρήστης την κατανάλωση σε πραγματικό χρόνο επιτρέπει να εντοπίζει τυχόν απρόβλεπτα φορτία κατανάλωσης, τυχόν διαρροές, παράνομη χρήση του δικτύου του από τρίτους ή ακόμη και ανεξήγητες καταναλώσεις εκτός υπολογισμού, που πιθανόν να τον οδηγήσουν σε αμφισβήτηση των χρεώσεων του εκάστοτε παρόχου.

Με αυτόν τον τρόπο, αυτή η εφαρμογή αποτελεί ένα ενδεικτικό παράδειγμα πώς η τεχνολογία του IoT δεν απλά προσφέρει παρακολούθηση και ανάλυση δεδομένων,

αλλά πραγματικά αλλάζει τον τρόπο που αλληλεπιδρούμε με τους πόρους μας, δημιουργώντας βιώσιμες και εκδοτικές πρακτικές για το μέλλον.

Κεφάλαιο 3.2 Υπηρεσίες cloud και η σημασία τους

Ο όρος "σύννεφο" (cloud) στον τομέα της πληροφορικής αναφέρεται σε μια αλλαγή στον τρόπο που διαχειρίζονται και παρέχονται υπολογιστικοί πόροι.

Η τεχνολογία αυτή έχει επανασχεδιάσει το τοπίο της σύγχρονης τεχνολογίας, παρέχοντας κλιμακούμενη, ανάλογα από την απαίτηση του διαχειριστή, πρόσβαση σε μια εκτεταμένη ποικιλία υπολογιστικών πόρων και υπηρεσιών μέσω του Διαδικτύου. Αυτή η μεταστροφή έχει έναν βαθύ αντίκτυπο στην ανάπτυξη και την υλοποίηση εφαρμογών, προσφέροντας πολλά οφέλη που συμβάλλουν στην αποδοτικότητα.

Ένα απλό παράδειγμα είναι ότι κλασικά για την λειτουργία μιας ολοκληρωμένης εφαρμογής χρειάζεται η ανάλογη υποδομή (servers, βάσεις δεδομένων και άλλες λειτουργίες) και η αντίστοιχη συντήρηση ώστε να διασφαλιστεί η ομαλή τους λειτουργία σε όλες τις πιθανές περιπτώσεις. Αντίστοιχα, σε ένα μοντέλο αρχιτεκτονικής **serverless** όπου η υποδομή της εφαρμογής αποτελείται από ένα σύμπλεγμα cloud υπηρεσιών, πετυχαίνουμε την διασφάλιση των παραπάνω απαιτήσεων από τις υπηρεσίες αυτές.

Παρακάτω αναφέρονται μερικά από τα οφέλη της αρχιτεκτονικής αυτής:

1. **Απλοποίηση Διαχείρισης Servers:** Σε μια serverless αρχιτεκτονική, η διαχείριση των server γίνεται απλούστερη καθώς φροντίζεται από τον πάροχο(των εκάστοτε cloud υπηρεσιών). Αυτό μειώνει τα έξοδα και επιτρέπει στους προγραμματιστές να επικεντρωθούν στη δημιουργία εφαρμογών χωρίς περιορισμούς από τη λειτουργικότητα και συντήρηση των server.
2. **Οικονομία με Δυναμική Χρέωση:** Οι αρχιτεκτονικές serverless προσφέρουν μοντέλο χρέωσης λεγόμενο και ως "πλήρωσε όσο τις χρησιμοποιείς" (pay as you go). Οι προγραμματιστές μπορούν χρεώνονται μόνο για τις υπηρεσίες που χρησιμοποιούν. Ο κώδικας εκτελείται μόνο όταν χρειάζεται, με αυτόματη κλιμάκωση εξαρτώμενη από τις στιγμιαίες απαιτήσεις και κίνησης της εφαρμογής, εξασφαλίζοντας ακριβή διαχείριση κόστους. Αυτό φυσικά δεν σημαίνει ότι δεν προσφέρονται και άλλης μορφής χρεώσεις (όπως δεσμευση ενός server με επιλεγμένα resources αποκλειστικά για σένα).
3. **Ενσωματωμένη Επεκτασιμότητα:** Οι εφαρμογές serverless προσφέρουν αυτόματη επεκτασιμότητα, προσαρμόζοντας τους πόρους αυτόματα σε μεταβαλλόμενα φορτία εργασίας. Αυτή η δυνατότητα αυτόματης κλιμάκωσης εξασφαλίζει ότι οι εφαρμογές serverless μπορούν να ανταποκριθούν σε αιφνιδιαστικές αυξήσεις της χρήσης χωρίς πτώση στην απόδοση, κάτι που συνηθίζεται στις παραδοσιακές αρχιτεκτονικές.

4. **Γρήγορες Αναπτύξεις και Ενημερώσεις:** Η υποδομή serverless επιτρέπει γρήγορες αναπτύξεις και ενημερώσεις. Οι προγραμματιστές μπορούν να ανεβάσουν και να χρησιμοποιήσουν νέο κώδικα χωρίς περιττές διαμορφώσεις ή μεταφορές κώδικα. Οι εφαρμογές αποτελούνται από μεμονωμένες λειτουργίες που επιτρέπουν στους προγραμματιστές να ενημερώνουν λειτουργίες ή να προσθέτουν νέα χαρακτηριστικά με ευελιξία, το μοντέλο αυτό λέγεται και ως "CI/CD" (continuous integration and continuous delivery/continuous deployment).
5. **Εκτέλεση Κώδικα κοντά στον Χρήστη:** Λόγω ότι η εφαρμογή δεν υφίσταται σε συγκεκριμένο server, ο κώδικας μπορεί να εκτελεστεί από οπουδήποτε. Έτσι, είναι δυνατή η εκτέλεση λειτουργιών της εφαρμογής σε servers που βρίσκονται κοντά στον τελικό χρήστη, μειώνοντας τις καθυστερήσεις. Αυτό βελτιώνει σημαντικά την απόκριση της εφαρμογής, καθώς οι αιτήσεις των χρηστών δεν χρειάζεται να ταξιδεύουν μέχρι έναν κεντρικό διακομιστή.

Επίσης πρέπει να αναφέρουμε και κάποια αρνητικά, καθώς η αρχιτεκτονική serverless δεν αποτελεί αντικατάσταση των server-full εφαρμογών, αλλά μια εναλλακτική υλοποίηση εξαρτώμενη από τις ανάγκες της εφαρμογής.

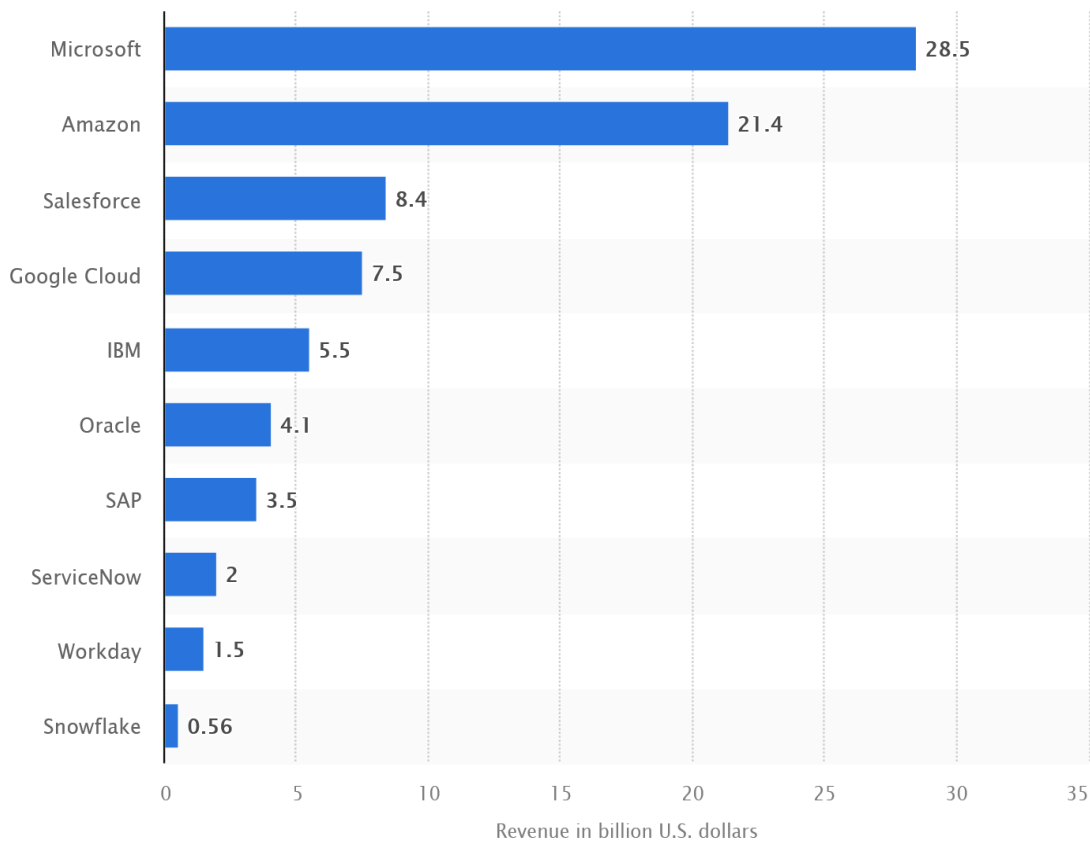
1. **Προβλήματα στον Έλεγχο και Αποσφαλμάτωση:** Κάνει δυσκολότερη την αναπαράσταση του serverless περιβάλλοντος ώστε να καταλάβει κάποιος πώς θα λειτουργήσει ο κώδικας όταν αναπτυχθεί πραγματικά. Η αποσφαλμάτωση είναι πιο πολύπλοκη λόγω της έλλειψης ορατότητας στις διαδικασίες στο παρασκήνιο και της διάσπασης της εφαρμογής σε ξεχωριστές, μικρότερες λειτουργίες. Φυσικά υπάρχουν εργαλεία τα οποία επιτρέπουν στον χρήστη να ανιχνεύει κάθε σημείο των υπηρεσιών που χρησιμοποιεί, βέβαια δεν παύει να χρειάζεται μια εξειδίκευση για να χρησιμοποιηθεί σωστά.
2. **Ασφάλεια:** Υπάρχουν πολλές ανησυχίες για την ασφάλεια και την εμπιστοσύνη του παρόχου που προσφέρει το συνολικό serverless οικοσύστημα. Όταν οι προμηθευτές διαχειρίζονται ολόκληρο το παρασκήνιο, είναι δύσκολο να εξεταστεί πλήρως η ασφάλειά, πράγμα που μπορεί να είναι ειδικά προβληματικό για εφαρμογές που διασφαλίζουν προσωπικά ή ευαίσθητα δεδομένα. Γι Αυτό πρέπει να επιλέγουμε προσεκτικά το τους παρόχους τέτοιων υπηρεσιών και να ψάχνουμε αναλυτικά για την ασφάλεια που προσφέρουν.
3. **Χρέωση:** Οι αρχιτεκτονικές serverless κοστολογούν τον διαχειριστή με χρονοχρέωση των υπηρεσιών. Αυτό περιορίζει τα είδη των εφαρμογών που μπορούν να λειτουργήσουν αποδοτικά σε ένα περιβάλλον serverless, όπως και απαιτεί μια καλή επίγνωση των απαιτήσεων της εφαρμογής που θέλουμε να τρέχει σε αυτό το περιβάλλον. Αν δεν χρησιμοποιηθούν με σύνεση και καλή σχεδίαση, είναι ικανές οι υπηρεσίες αυτές να μας χρεώσουν υπέρογκα ποσά χωρίς την απαιτούμενη αποδοτικότητα.

4. **Εκκίνηση υπηρεσιών:** Επειδή ο κώδικας που τρέχει στο cloud δεν εκτελείται συνεχώς, μπορεί να χρειαστεί να πάρει κάποιο χρόνο να "ξεκινήσει" όταν υπάρχει ανάγκη να χρησιμοποιηθεί. Αυτός ο χρόνος εκκίνησης μπορεί να επηρεάσει την απόδοση. Ωστόσο, αν ένα κομμάτι κώδικα χρησιμοποιείται τακτικά, ο πάροχος cloud θα το διατηρήσει έτοιμο για ενεργοποίηση "warm start." Αν ο κώδικας δεν έχει χρησιμοποιηθεί για κάποιο χρονικό διάστημα και απαιτείται "εκκίνηση" όπως αναφέρθηκε παραπάνω, αυτό ονομάζεται "cold start." Καλό είναι λοιπόν με βάση τις ανάγκες της εφαρμογής και τα χρήματα που διαθέτουμε, να επιλέγουμε ποιά απο τις 2 λειτουργίες μας εξυπηρετεί.

Κεφάλαιο 3.3 Ανάλυση του παρόχου cloud υπηρεσιών Amazon AWS

Συγκεκριμένα αναφέρεται στην προηγούμενη ενότητα μερικά από τα θετικά αλλά και τα αρνητικά σχετικά με την serverless αρχιτεκτονική. Ο καλύτερος τρόπος ώστε να αντλήσουμε τα θετικά και να ελαχιστοποιήσουμε τα αρνητικά, είναι επιλέγοντας ευρέως διαδεδομένους παρόχους cloud υπηρεσιών που θα μας διασφαλίσουν σημαντικούς παράγοντες όπως ασφάλεια, αμεσότητα, υποστήριξη, οικονομία και συνεχή εξέλιξη με υπηρεσίες τελευταίας τεχνολογίας.

Ονομαστικά, οι πιο μεγάλοι πάροχοι cloud υπηρεσιών το 2023 απεικονίζονται στην παρακάτω εικόνα, με βάση τα ετήσια έσοδα σε δις τον χρόνο:

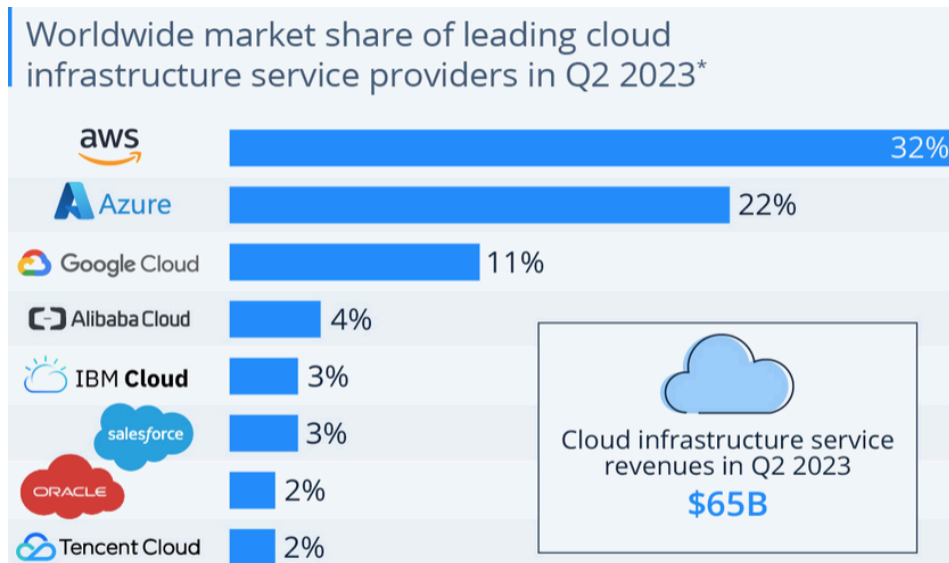


Εικόνα 3.1 ετήσια έσοδα από τους πιο δημοφιλή cloud παρόχους σε δις τον χρόνο

Ενώ όλοι οι παραπάνω πάροχοι λειτουργούν με βάση τις ίδιες αρχές, η σωστή επιλογή εξαρτάται από τις απαιτήσεις της cloud αρχιτεκτονικής που θέλουμε να υπολογίσουμε.

Για την συγκεκριμένη εφαρμογή, επιλέχθηκαν οι cloud υπηρεσίες της Amazon καθώς είναι με διαφορά ο πιο διαδεδομένος πάροχος cloud υπηρεσιών όπως φαίνεται και

στην παρακάτω εικόνα που αναπαριστάται η κεφαλαιαγορά μεταξύ των πιο διάσημων παρόχων το τελευταίο έτος:



Εικόνα 3.2: Market share των βασικών παρόχων cloud υπηρεσιών

Άλλοι λόγοι που προτιμήθηκε σε αντίθεση με άλλους παρόχους είναι οι παρακάτω:

- **Χαμηλό κόστος:** Ένας πολύ βασικός παράγοντας σε ένα ερασιτεχνικό project, είναι η δυνατότητα δοκιμής υπηρεσιών της πλατφόρμας χωρίς οικονομικούς περιορισμούς. Χάρη στην κοστολόγηση με βάση των όγκο δεδομένων και την χρονική κοστομέτρηση, δίνεται η δυνατότητα για περαιτέρω δοκιμές κατα την ανάπτυξη της εφαρμογής, αλλά επίσης προσφέρεται και η ευκολία για πειραματισμό με νέες λειτουργίες και υπηρεσίες που αναπτύσει ο πάροχος.
- **Ευκολία στην χρήση:** Μπορεί η αρχιτεκτονική που ακολουθεί η Amazon να είναι απλή για κάποιον νέο στον κόσμο του cloud, όμως ακολουθώντας το αναλυτικό εγχειρίδιο χρήσης για κάθε υπηρεσία, βίντεο γραφικό υλικό από την ίδια την Amazon, παρουσιάσεις και δοκιμαστικά project ανοιχτού κώδικα πάνω στην κάθε υπηρεσία, η εκμάθηση της πλατφόρμας γίνεται με σταθερους ρυθμούς. Έπειτα, το περιβάλλον χρήστη (user interface) τις περισσότερες φορές είναι πολύ κατατοπιστικό και η εναλλαγή παρακολούθησης από υπηρεσία σε υπηρεσία πολύ γρήγορη. Τέλος, η υπηρεσία υπολογισμού κόστους χρήσης ανα υπηρεσία, σου δίνει ακόμα και παραδείγματα χρήσης ανα υπηρεσία για να υπολογίσεις το ακριβές κόστος του project σου.
- **Μεγάλη κοινότητα:** Βλέποντας το παραπάνω γράφημα, μπορούμε να διακρίνουμε την διάκριση της Amazon οι 2ο πιο επικερδή πάροχο υπηρεσιών cloud. Με αμέτρητους χρήστες, προγραμματιστές και ειδικούς που μοιράζονται τις εμπειρίες και τη γνώση τους, οι χρήστες της AWS μπορούν εύκολα να βρουν πόρους, υποστήριξη και λύσεις για τις προκλήσεις τους στο

διαδίκτυο, καθώς υπάρχουν ακόμη μαθήματα και πιστοποιητικά για τις υπηρεσίες αυτές.

- **Σταθερότητα:** Όντας η πιο ξακουστή επιλογή για μεγάλες επιχειρήσεις αλλά και για το ευρύ κοινό, καταλαβαίνουμε ότι οι υποδομές που προσφέρει είναι ικανές να ανταποκριθούν με ασφάλεια και σταθερότητα στην κάθε ανάγκη και στον κάθε αριθμό χρηστών χωρίς να επηρεάζεται η λειτουργικότητα των εφαρμογών.

Έχοντας υποδομές σε 30 διαφορετικά μέρη του πλανήτη, ενισχύεται η σταθερότητα, καθώς παρέχεται η δυνατότητα αποκέντρωσης και εξισορρόπησης του φόρτου, μειώνοντας τον κίνδυνο αποτυχίας σε μια μόνο περιοχή. Επιπλέον, έχοντας πολλές περιοχές βελτιώνεται η ταχύτητα πρόσβασης, επιτρέποντας στους χρήστες να έχουν γρήγορη πρόσβαση στις υπηρεσίες και τα δεδομένα τους από την πλησιέστερη περιοχή.

Υποστήριξη IoT: Για την υλοποίηση μιας εφαρμογής που συλλέγει και επεξεργάζεται δεδομένα από έξυπνες συσκευές, πολύ σημαντικό κομμάτι είναι η διασύνδεση των συσκευών με την εφαρμογή και η επεξεργασία των δεδομένων αυτών. Η AWS έχει επενδύσει αρκετά στο συγκεκριμένο κομμάτι προσφέροντας υποστήριξη για διασύνδεση μέσω πρωτοκόλλου MQTT αλλά και στην αποθήκευση των δεδομένων αυτών μέσω υπηρεσίας που χρησιμοποιεί μια βάση δεδομένων ειδική για δεδομένα τηλεμετρίας (δεδομένα που προέρχονται από μετρητές).

Κεφάλαιο 3.4 Ανάλυση επίπεδων αρχιτεκτονικής για την εφαρμογή ENERCOST

Όπως αναέραμε και στο δεύτερο κεφάλαιο, η δομή μιας ολοκληρωμένης IoT εφαρμογής, αποτελείται από 4 βασικά επίπεδα. Η εφαρμογή που αναπτύχθηκε στο πλαίσιο της διπλωματικής εργασίας, με τίτλο ENERCOST αποτελεί μια αρχιτεκτονική IoT εφαρμογής.

Παρακάτω αναφέρονται με αναλυτική δομή η λειτουργικότητα του κάθε επιπέδου προς την εφαρμογή:

Στο **Physical Layer**, αναφέρονται η "έξυπνες" συσκευές που έχουν σαν ρόλο την μέτρηση και αποστολή μετρήσεων κατανάλωσης από νερό, φυσικό αέριο και ηλεκτρικό ρεύμα. Το καλό, είναι ότι δεν υπάρχει αυστηρός περιορισμός στην επιλογή των συσκευών αυτών καθώς η εφαρμογή θέτει ως βασική προϋπόθεση την αποστολή των δεδομένων με ένα συγκεκριμένο μορφότυπο JSON μέσω του πρωτοκόλλου MQTT. Με αυτόν τον τρόπο, η εφαρμογή είναι ευέλικτη και προσαρμοστική, επιτρέποντας την ομαλή ένταξη διαφορετικών τύπων συσκευών, εφόσον τηρούν τις καθορισμένες προδιαγραφές επικοινωνίας.

Οι μετρήσεις αυτές σε ένα πραγματικό σενάριο θα λαμβάνονται από έξυπνες συσκευές στις οποίες υπάρχει καταλληλος αισθητήρας για την συγκεκριμένη μέτρηση.

Μόνη προϋπόθεση για την αποστολή των δεδομένων στην εφαρμογή ENERCOST, είναι όπως αναφέρθηκε και παραπάνω η ικανότητα αποστολής δεδομένων μέσω πρωτοκόλλου MQTT.

Βέβαια για λόγους ευκολίας, στο επίπεδο της διπλωματικής, τα δεδομένα αυτά θα αναπαριστούν απο μια συσκευή που θα λειτουργεί σαν "γεννήτρια δεδομένων" η οποία έπειτα θα τα προωθεί στην εφαρμογή ENERCOST μέσω πρωτοκόλλου MQTT.

Στο **Network Layer**, αναφέρονται οι τρόποι διασύνδεσης των έξυπνων συσκευών στο διαδίκτυο αλλά και μεταξύ τους. Οι ποιοι ευρύς διαδεδομένοι είναι μέσω wifi ή μέσω ethernet, βέβαια υπάρχουν και άλλες τεχνολογίες πιο εστιασμένες στην κατανάλωση ενέργειας όπως Zigbee και LoRa. Επίσης αναφέρονται και τα πρωτόκολλα επικοινωνίας με τα οποία στέλνουν δεδομένα, μερικά από αυτά είναι το HTTP, CoAP and MQTT.

Για την εφαρμογή Enercost ο τρόπος διασύνδεσης των συσκευών στο διαδίκτυο δεν παίζει σημαντικό ρόλο, καθώς όπως αναφέραμε και από πάνω, μόνη βασική προϋπόθεση για την αποστολή των δεδομένων είναι η δυνατότητα αποστολής μέσω του πρωτοκόλλου MQTT.

Στο **Middleware Layer**, η εφαρμογή διαχειρίζεται την επεξεργασία, αποθήκευση και διαχείριση των δεδομένων που παράγονται από τις συσκευές IoT. Αυτό περιλαμβάνει τη χρήση υπηρεσιών κατάλληλων για την κάθε παραπάνω απαίτηση. Για την συγκεκριμένη εφαρμογή έχει επιλεγθεί η χρήση cloud υπηρεσιών (τα λεγόμενα cloud services), τα οποία αποτελούν υπολογιστικούς πόρους και εφαρμογές που παρέχονται μέσω του διαδικτύου. Περιλαμβάνουν αποθήκευση δεδομένων, υπολογιστική ισχύ, βάσεις δεδομένων και λογισμικό, επιτρέποντας στους χρήστες να αποκτούν πρόσβαση και να χρησιμοποιούν αυτούς τους πόρους χωρίς την ανάγκη να φτιάξουν και να ανεβάσουν στο ίντερνετ από το μηδέν μια υποδομή. Συγκεκριμένα έχουν χρησιμοποιηθεί οι υπηρεσίες cloud της Amazon ή αλλιώς Amazon Web Services (AWS) δημιουργώντας έτσι μια αρχιτεκτονική χωρίς την ανάγκη ενός server (**serverless architecture**).

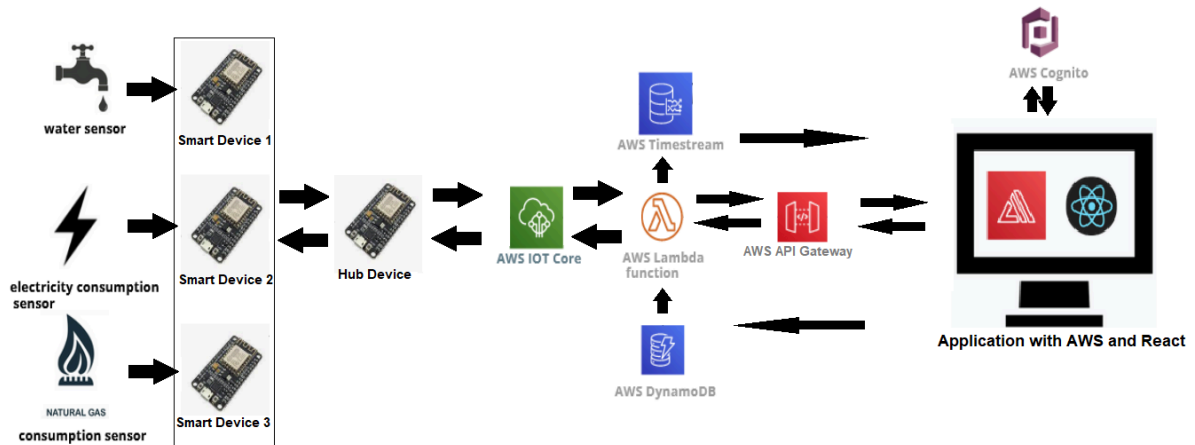
Στο **Application Layer**, η εφαρμογή παρέχει μια διεπαφή χρήστη για την οπτικοποίηση και αλληλεπίδραση με τα δεδομένα που έρχονται από τους αισθητήρες αλλά και στοιχεία για την ταυτότητα τους.

Το κομμάτι αυτό αποκαλούμενο και ως Front-end, έχει δημιουργηθεί για την συγκεκριμένη εφαρμογή χρησιμοποιώντας την γλώσσα προγραμματισμού Javascript (που θεωρείται και η κύρια γλώσσα του front end) μαζί με τεχνολογίες React, Redux, και την υπηρεσία της Amazon AWS Amplify.

Τελειώνοντας με μια συνοπτική περιγραφή της εφαρμογής και των αναγκών της ανα επίπεδο, στα επόμενα κεφάλαια, θα εξηγηθούν αναλυτικά οι παραπάνω όροι και οι χρησιμότητά τους, καθώς και οι λόγοι επιλογής τους. Επίσης, στην συνέχεια θα εμβαθύνουμε κυρίως στα τελευταία 2 επίπεδα (Middleware και Application Layer) καθώς η εφαρμογή Enercost εστιάζει περισσότερο στην ροή των δεδομένων αφού σταλούν από τους αισθητήρες.

4.1 Ανάλυση της αρχιτεκτονικής της Εφαρμογής ENERCOST

Στο παρακάτω διάγραμμα φαίνεται αναλυτικά η αρχιτεκτονική της εφαρμογής, με βάση ότι ειπώθηκε στο υποκεφάλαιο 3.1

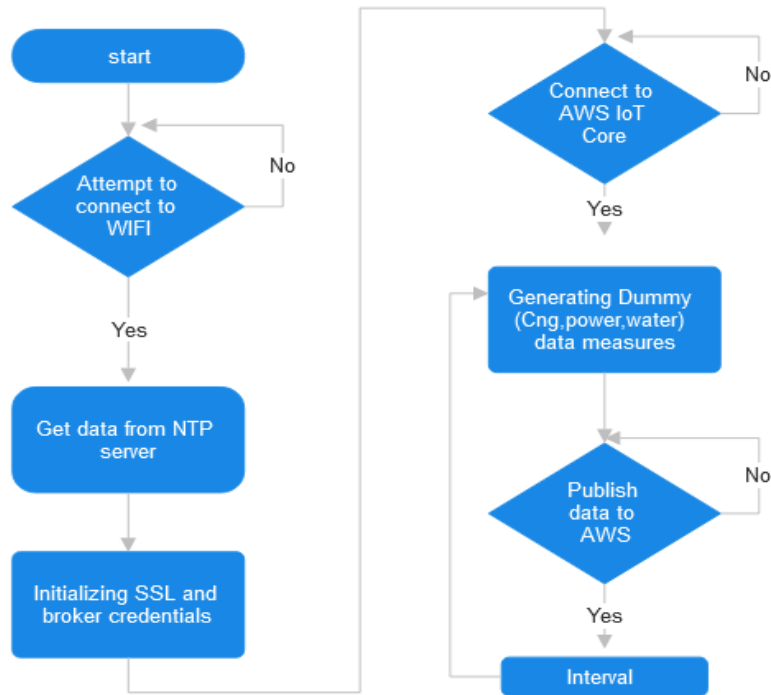


Εικόνα 4.1 Αναπαράσταση αρχιτεκτονικής εφαρμογής ENERCOST

- **Συσκευές:** Όπως είναι εμφανή οι 3 μετρήσεις από τους αισθητήρες επεξεργάζονται στην αντίστοιχη "έξυπνη συσκευή" και έπειτα οι 3 αυτές συσκευές στέλνουν τα δεδομένα σε μια κεντρική συσκευή (Hub device). Η συσκευή αυτή λειτουργεί σαν συλλέκτης δεδομένων των άλλων συσκευών. Έχει σύνδεση με το διαδίκτυο και ο ρόλος αυτής, είναι να προωθεί τα δεδομένα μέσω πρωτοκόλλου MQTT στο cloud όπου βρίσκεται η εφαρμογή ENERCOST.

Καθώς για την συγκεκριμένη εργασία απουσιάζουν στην υλοποίηση τα εικονιζόμενα πίσω από το hub device, δίνονται στο hub device και ο ρόλος του data generator.

Στο παρακάτω διάγραμμα ροής, φαίνεται η διαδικασία για την αποστολή των δεδομένων από το hub device στο aws iot core (cloud). Η υλοποίηση του κώδικα είναι σε γλώσσα C++ πάνω στο Arduino IDE.



Εικόνα 4.2 Διάγραμμα Ροής προγράμματος hub device

Ο τρόπος με τον οποίο προωθούνται οι μετρήσεις στο cloud, είναι μέσω του πρωτοκόλλου JSON (για δομημένη αναπαράσταση δεδομένων) δημιουργώντας μια δομή αντίστοιχη με την εικόνα 2*.

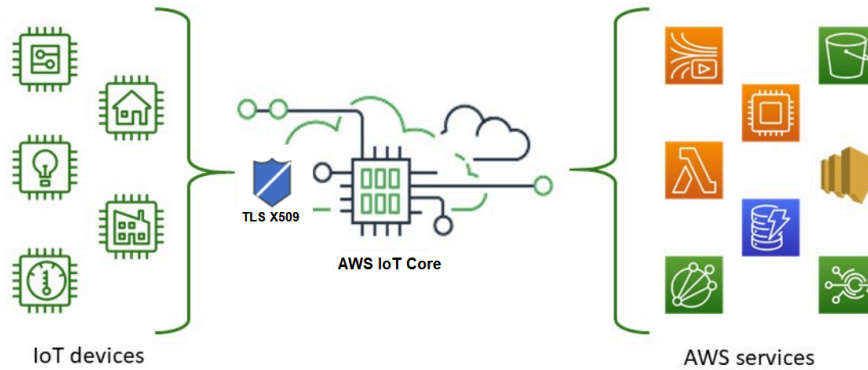
```

{
  "building_id": "B-1",
  "device_id": "ENB1D1",
  "device_loc": "Leof.Alexandras 15 Athens",
  "device_type": "enercost_sensor",
  "watt_value": 1210,
  "watt_price_value": 242,
  "cng_value": 2,
  "cng_price_value": 0.2,
  "water_value": 279,
  "water_price_value": 0.279
}
  
```

Εικόνα 4.3 Δεδομένα μετρήσεων και πληροφορίες συσκευών σε μορφή JSON

Επισημαίνεται ξανά, ότι για τις ανάγκες της υλοποίησης σε επίπεδο διπλωματικής λείπουν από την ολοκληρωμένη αρχιτεκτονική, οι 3 αυτές έξυπνες συσκευές και στον ρόλο του Hub Device έχει προστεθεί και η δημιουργία εικονικών δεδομένων κατανάλωσης.

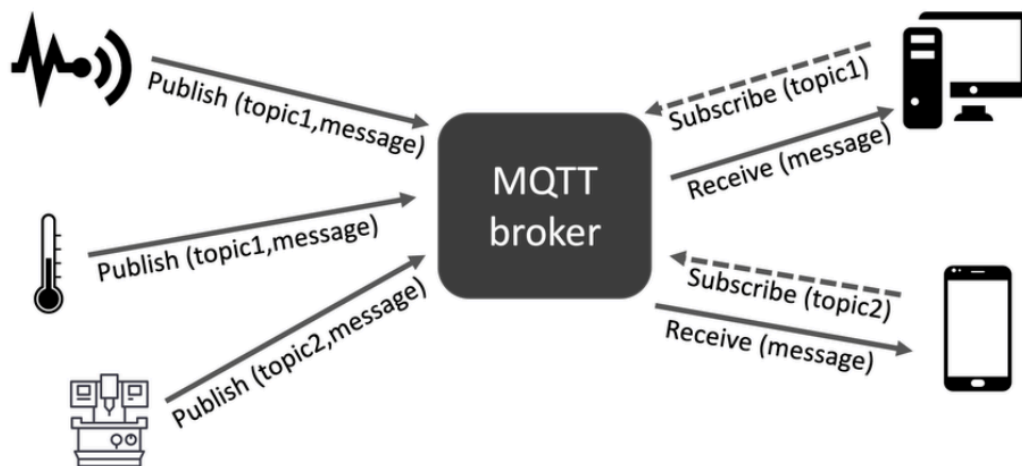
- AWS IoT Core:** Στην συνέχεια, τα δεδομένα που αποστέλλονται μέσω MQTT απο το Hub device, λαμβάνονται από μια υπηρεσία του AWS που ονομάζεται AWS IoT Core. Αυτή η υπηρεσία, χρησιμεύει για την σαν ενδιάμεση πύλη για την λήψη και την αποστολή δεδομένων μεταξύ των έξυπνων συσκευών και του cloud.



Εικόνα 4.4 Σύνδεση μεταξύ συσκευών και AWS εφαρμογών μέσω AWS IoT Core

Για την επικοινωνία των συσκευών μέσω EoT Core, απαιτείται κρυπτογράφηση μέσω TLS ώστε να διασφαλιστεί η ασφαλή μεταφορά των δεδομένων στο cloud. Υπάρχουν πολλοί διαφορετικοί τρόποι να συνδεθεί κανείς συσκευές στο AWS IoT Core (όπως μέσω HTTPS, MQTTS, LoRa), στην περίπτωση της εφαρμογής ENERCOST χρησιμοποιείται το πρωτόκολλο MQTTS (MQTT Secure over TLS).

Ως είναι γνωστό, για την επικοινωνία μέσω πρωτοκόλλου MQTT χρειάζεται ένας διαμοιραστής (mqtt broker) όποιος διαμειράζει τα μηνύματα με βάση την διεύθυνση (topic). Εκείνοι που τα λαμβάνουν, είναι συσκευές ή υπηρεσίες που έχουν επιλέξει να δέχονται μηνύματα από το συγκεκριμένο topic (κάνουν subscribe στο topic). Οι αποστολές των μηνυμάτων χαρακτηρίζονται ως publishers και προωθούν (publish) τα μηνύματα με την αντίστοιχη διεύθυνση στον mqtt broker για διαμοιρασμό. Σημαντικό στην συγκεκριμένη αρχιτεκτονική είναι η άγνοια των αποστολέων και των δέκτων (publishers-subscribers) για την προέλευση ή την κατάληξη των μηνυμάτων, ο μοναδικός που εκτελεί την συγκεκριμένη λειτουργία είναι ο διαχωριστής (mqtt broker)..



Εικόνα 4.5 Διάγραμμα λειτουργίας MQTT

Η υπηρεσία AWS IoT Core παρέχει ενσωματωμένο mqtt broker που συνδέονται οι συσκευές.

Για την διασφάλιση της επικοινωνίας των συσκευών με το cloud με το συγκεκριμένο πρωτόκολλο, απαιτείται η συσκευή (device Hub) να παρέχει 3 διαφορετικά αρχεία:

1. **Πιστοποιητικό Συσκευής:** Πρόκειται για ένα ψηφιακό πιστοποιητικό, μοναδικό για τη συσκευή. Λειτουργεί ως μορφή αναγνώρισης για τη συσκευή σας όταν στο AWS IoT Core. Το πιστοποιητικό αυτό συνήθως εκδίδεται από μια αρχή πιστοποίησης (Certificate Authority, CA) ή μπορεί να είναι αυτο-υπογεγραμμένο.
2. **Ιδιωτικό Κλειδί:** Πρόκειται για το ιδιωτικό κλειδί (private key) που σχετίζεται με το πιστοποιητικό της συσκευής. Πρέπει να φυλάσσεται με ασφάλεια στη συσκευή IoT και δεν πρέπει ποτέ να κοινοποιείται ή να αποκαλύπτεται. Χρησιμοποιείται για την κρυπτογράφηση και αποκρυπτογράφηση δεδομένων κατά τη διαδικασία TLS handshake.
3. **Πιστοποιητικό Αρχής CA:** Το πιστοποιητικό αρχής χρησιμοποιείται για τον έλεγχο της αυθεντικότητας και της εγκυρότητας άλλων πιστοποιητικών εντός του συστήματος AWS IoT, συμπεριλαμβανομένων των πιστοποιητικών συσκευών. Είναι απαραίτητο για τη δημιουργία ασφαλούς επικοινωνίας, καθώς επιτρέπει στις συσκευές να εμπιστεύονται τα πιστοποιητικά των άλλων συσκευών που υπογράφονται από το ίδιο πιστοποιητικό αρχής CA.

Μέσα στην εφαρμογή AWS IoT Core υπάρχει μια ψηφιακή αναπαράσταση (λεγόμενο και ως thing) της συσκευής της οποίας θέλει να συνδεθεί στο cloud.

Για να συνδεθεί μια συσκευή στο cloud, απαιτείται η δημιουργία ενός thing μέσω AWS IoT Core, μαζί δημιουργείται και ένα πιστοποιητικό (certificate) στο οποίο μπορεί ο διαχειριστής να προσθέσει λειτουργίες που επιτρέπεται να πραγματοποιήσει

η συσκευή (όπως για παράδειγμα την δυνατότητα να στέλνει δεδομένα σε συγκεκριμένο topic ή να περιοριστεί η ικανότητα του να δεχθεί εντολές από το cloud).

έπειτα δημιουργούνται τα παραπάνω πιστοποιητικά (CA certificate, Certificate, Private key) τα οποία πρέπει να αποθηκευτούν στην μνήμη της συσκευής, ώστε κατά την σύνδεση με το cloud να μπορέσει να πραγματοποιηθεί η TLS "χειραψία" μεταξύ συσκευής και cloud (AWS IoT Core).

Το άλλο κομμάτι του IoT Core είναι η προώθηση των δεδομένων σε διάφορες υπηρεσίες του AWS οικοσυστήματος. Αυτό επιτυγχάνεται χάρη στα λεγόμενα IoT Core Rules, οι οποίοι είναι κάποιοι προκαθορισμένοι κανόνες για την κατάληξη των δεδομένων αυτών. Είναι ουσιαστικά, σύνδεσμοι μεταξύ Aws IoT Core και των υπόλοιπων υπηρεσιών της AWS.

Στην περίπτωση της εφαρμογής ENERCOST υπάρχει μονάχα ένας τέτοιος κανόνας, ο οποίος προωθεί τα δεδομένα που αποστέλλονται από τις συσκευές με ένα συγκεκριμένο topic (διεύθυνση αποστολής) στην υπηρεσία AWS Lambda, η οποία παίζει σημαντικό ρόλο στην ολική αρχιτεκτονική της εφαρμογής.

AWS Lambda: Η συγκεκριμένη υπηρεσία, επιτρέπει στον διαχειριστή, να εκτελεί κώδικα σε πραγματικό χρόνο ανάλογα με την χρονική στιγμή που χρειάζεται (συνήθως όταν κάποια άλλη υπηρεσία του AWS οικοσυστήματος το απαιτεί).

Βασιζόμενο στην φιλοσοφία εκτέλεσης εφαρμογών χωρίς την φυσική παρουσία μεμονωμένου server (serverless), μπορεί να τρέξει μεμονωμένο κώδικα σε πραγματικό χρόνο χωρίς την ανάγκη για συντήρηση του server όπως και επίσης λόγω της serverless αρχιτεκτονικής, μπορεί να κλιμακώνεται ανάλογα τις ανάγκες και τον φόρτο λειτουργίας (π.χ όταν πολλοί χρήστες εκτελούν ένα request το οποίο τρέχει ένα συγκεκριμένο κομμάτι κώδικα (lambda function)).

Ο κωδικός αυτός, μπορεί να καλέσει (trigger) άλλες υπηρεσίες του AWS όπως API calls, αποστολή δεδομένων σε AWS IoT core, επικοινωνία με βάση δεδομένων αλλά ακόμα και δημιουργώντας ένα ολοκληρωμένο back-end περιβάλλον για την υποστήριξη διαδικτυακών εφαρμογών.

Φυσικά αυτό έρχεται με το αντίστοιχο οικονομικό αντάλλαγμα, γι αυτό απαιτείται ιδιαίτερη προσοχή στην χρήση.

Για τις ανάγκες της εφαρμογής, έχουν δημιουργηθεί πολλά διαφορετικά "κομμάτια κώδικα" (lambda functions) που τρέχουν στο AWS Lambda. Όπως φαίνεται στην παραπάνω εικόνα *1, κάθε βέλος με κατεύθυνση προς και από το εικονίδιο AWS Lambda αντιστοιχεί σε μια κατηγορία από lambda functions. Στην συνέχεια της ανάλυσης του project, θα αναφερθούν όλες οι υπεύθυνες lambda functions καθώς είναι πιο εύκολο να κατανοηθεί η σημασία τους.

Μία από αυτές είναι υπεύθυνη για τον έλεγχο, την εγκυρότητα, την κατάλληλη μορφοποίηση και την μεταφορά των δεδομένων που προέρχονται από την υπηρεσία AWS IoT Core μέσω του αντίστοιχου κανόνα, στην υπηρεσία βάσης δεδομένων AWS Timestream.

- **AWS Timestream:** Το Amazon Timestream είναι μια υπηρεσία βάσης δεδομένων της Amazon AWS που είναι ειδικά σχεδιασμένη για τη διαχείριση και ανάλυση χρονο σειριακών δεδομένων, ανήκει στην κατηγορία βάσεων δεδομένων χρονοσειρών (time series databases). Η συγκεκριμένη κατηγορία είναι ιδανική για εφαρμογές που απαιτούν την αποθήκευση και την ανάλυση δεδομένων που αλλάζουν με την πάροδο του χρόνου, όπως συστήματα αισθητήρων, αποθήκευση μετρήσεων και παρακολούθηση και ανάλυση μετρήσεων, ακόμη και επεξεργασίας μετρήσεων με χρήση τεχνητής νοημοσύνης (A.I).

Το Amazon Timestream, μπορεί να αποθηκεύει μεγάλο όγκο δεδομένων με υψηλή ανάλυση και να εκτελεί πολύπλοκες αναζητήσεις για ανάλυση και εξαγωγή σημαντικών πληροφοριών. Επιπλέον, η υπηρεσία παρέχει εύκολη σύνδεση με άλλες υπηρεσίες της AWS, καθιστώντας την κατάλληλη για πληθώρα εφαρμογών που χρησιμοποιούν τη διαχείριση χρονο σειριακών δεδομένων.

Σημαντικές διαφορές σε σχέση με μια συμβατική βάση δεδομένων SQL είναι:

- **Αποθήκευση Δεδομένων:**
 - **Βάση Χρονοσειράς (TSDB):** Οργανωμένη για τη διαχείριση δεδομένων χρονοσειράς με σήμανση χρόνου, συνήθως με χρονολογική σειρά.
 - **SQL Βάση:** Ακολουθεί δομή πίνακα με στήλες και γραμμές για την αποθήκευση δομημένων δεδομένων.
- **Αποθήκευση Μετρήσεων:**
 - **Βάση Χρονοσειράς (TSDB):** Σχεδιασμένη για αποθήκευση, ανάκτηση και ανάλυση δεδομένων που περιλαμβάνουν χρονικές σήμανσεις, όπως δεδομένα που έρχονται από smart συσκευές.
 - **SQL Βάση:** Δυνατότητα αποθήκευσης για διάφορους τύπους δεδομένων συμπεριλαμβανομένων των χρονικών μετρήσεων χωρίς όμως την ίδια απόδοση.
- **Αναλύση:**
 - **Βάση Χρονοσειράς (TSDB):** Συνήθως περιλαμβάνει ενσωματωμένη υποστήριξη για την αθροισή δεδομένων κατά χρονικά διαστήματα, κρίσιμη για αναλύσεις χρονοσειρών.
 - **SQL Βάση:** Οι αναλύσεις και αθροίσματα μπορεί να απαιτούν πολύπλοκες SQL ερωτήσεις και δεν είναι πάντα βελτιστοποιημένες για αθροισμα χρονικών δεδομένων.
- **Επεκτασιμότητα:**

- **Βάση Χρονοσειράς (TSDB):** Συνήθως επεκτείνεται οριζόντια για να υποστηρίξει υψηλή συχνότητα χρονο σειριακών δεδομένων.
- **SQL Βάση:** Συνήθως εκτείνεται κάθετα για τη διαχείριση αυξημένου φόρτου εργασίας, που μπορεί να απαιτεί περισσότερους πόρους.

Ο τρόπος άντλησης και απεικόνισης των δεδομένων αυτού, γίνεται με παρόμοιο τρόπο όπως και σε μια βάση δεδομένων τύπου SQL, χωρίς όμως να υπάρχει συσχέτιση μεταξύ πινάκων.

Τα πιο συνηθισμένα παραδείγματα χρήσης (use case scenarios) για την επιλογή βάσης χρονοσειράς (όπως η AWS Timestream) είναι για την αποθήκευση δεδομένων τηλεμετρίας και αποθήκευση μετρήσεων πόρων απο υπολογιστικά συστήματα (όπως cpu utilization, ram, disk, and speed...).

city	measure_value::double	measure_name	time
Istanbul	16.5	temp	2020-10-19 00:00:00.000000000
Istanbul	16.28	temp	2020-10-19 01:00:00.000000000
Istanbul	16.08	temp	2020-10-19 02:00:00.000000000
Istanbul	15.64	temp	2020-10-19 03:00:00.000000000
Istanbul	15.86	temp	2020-10-19 04:00:00.000000000
Istanbul	15.77	temp	2020-10-19 05:00:00.000000000
Istanbul	15.98	temp	2020-10-19 06:00:00.000000000
Istanbul	16.07	temp	2020-10-19 07:00:00.000000000
Istanbul	16.72	temp	2020-10-19 08:00:00.000000000
Istanbul	16.93	temp	2020-10-19 09:00:00.000000000

Εικόνα 4.6 Αναπαράσταση δεδομένων θερμοκρασίας στην βάση AWS Timestream

Καθώς η εφαρμογή ENERCOST είναι βασισμένη στην άντληση δεδομένων τηλεμετρίας απο αισθητήρες, είναι πλέον προφανές ο λόγος επιλογής της συγκεκριμένης βάσης για την αποθήκευση και την άντληση των δεδομένων αυτών.

Η Αρχιτεκτονική που ακολουθείται για την αποθήκευση των μετρήσεων της εφαρμογής, απαρτίζεται από μία βάση δεδομένων στο AWS Timestream με

όνομα buildings. Εκεί υπάρχουν διάφοροι πίνακες, όπου ο καθένας είναι υπεύθυνος για την αποθήκευση δεδομένων από το αντίστοιχο κτίριο (παράδειγμα ο πίνακας B-1 περιέχει τις μετρήσεις που προέρχονται από το κτίριο 1 και ού το καθεξής).

Ο κάθε πίνακας περιέχει κάποιες στήλες όπου η κάθε μια αντιστοιχούν σε μια τιμή. Στην παρακάτω εικόνα φαίνονται όλες οι στήλες (οι στήλες είναι ίδιες σε όλα τα κτίρια) για το κτίριο 1.

Rows returned (2)

🔍 Filter

device_loc	device_id	device_type	measure_name	time
Leof.Alexandra s 15 Athens	ENB1D1	enercost_senso r	metrics	2023-09-27 08:50:12.9620 00000
Leof.Alexandra s 15 Athens	ENB1D1	enercost_senso r	metrics	2023-09-27 08:50:32.1010 00000

water_value	watt_price_val ue	water_price_v alue	cng_price_valu e	watt_value	cng_value
151.0	398.2	0.151	0.0	1991.0	0.0
280.0	470.4	0.28	1.0	2352.0	10.0

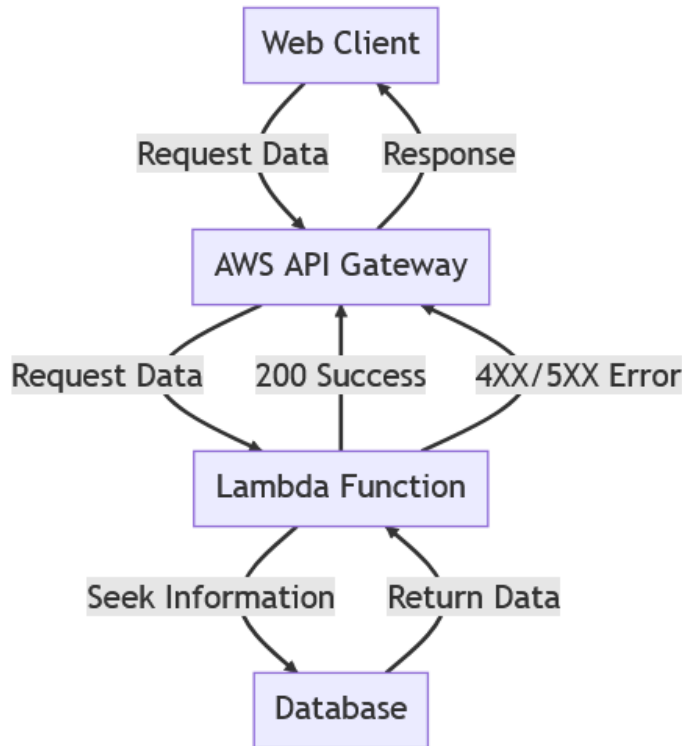
Εικόνα 4.7 Μετρήσεις πίνακα B-1 (building 1) από βάση δεδομένων buildings

- device_loc: Διεύθυνση κτιρίου που είναι εγκατεστημένος ο αισθητήρας.
- device_id: Αναγνωριστικό συσκευής (απευθυνόμαστε στην συσκευή που στέλνει τα δεδομένα στο cloud "hub device")
Στο συγκεκριμένο παράδειγμα η εξήγηση είναι ως εξής **ENB1D1**->{EN= Enercost Sensor, B1= Building 1, D1 = Device 1}
- device_type: Τύπος αισθητήρα.
- measure_name: Τύπος δεδομένων, σχεδόν πάντα αυτή η κατηγορία θα είναι κοινή, υπάρχει όμως το ενδεχόμενο να αλλάξει αν χρειαστεί να γίνει

παρακολούθηση σε κάποιους συγκεκριμένους πόρους του αισθητήρα (π.χ μέτρηση CPU power κλπ).

- **time:** ώρα που έγινε η αποθήκευση (δημιουργείται αυτόματα από την βάση).
 - **water_value,watt_value,cng_vauue:** Η τιμή μέτρησης απο τους αισθητήρες για τις αντίστοιχες μετρήσεις. Ο υπολογισμός στην αντίστοιχη μοναδα μετρησης (παράδειγμα κατανάλωση σε kw/hour) δεν εμφανίζεται αμεσα απο τον αισθητηρα.Ο υπολογισμός υπολογίζεται είτε στην "έξυπνη συσκευή" που παίρνει την μέτρηση από τον αισθητήρα ή στο τελικό "hub-device" που στέλνει τις μετρήσεις στο cloud. Όπως αναφέρθηκε, για την διπλωματικη, αυτές οι μετρήσεις δημιουργούνται στο hub-device καθώς δεν αναπαριστούν φυσικές τιμές.
 - **water,cng,watt_price_value:** Τιμή υπολογισμού κόστους σε χρήματα ανα μέτρηση. Ο υπολογισμός αυτος προκύπτει απο τον πολλαπλασιασμό της τιμης κατανάλωσης με την σταθερή τιμή κοστολόγησης ανα μέτρηση (προκύπτει από την εκάστοτε τιμή κατανάλωσης του εκάστοτε παρόχου). Παράδειγμα αν η τιμή του ρεύματος είναι 0.15 λεπτά ανα kw/h, εάν η τιμή watt_value την προκειμένη χρονική στιγμή είναι 2,5kw/h, τότε προκύπτει τιμή watt_price_value: $0.15 * 2.5 = 0.375$ λεπτά του ευρώ. Η μέτρηση αυτή υπολογίζεται απο το hub-device και αποστέλλεται στο cloud για αποθήκευση μαζί με τις άλλες.
-
- **AWS API Gateway:** Ο τρόπος με τον οποίο τα δεδομένα που βρίσκονται στο cloud αλληλεπιδρούν με τον χρήστη της εφαρμογής (front-end),βρίσκεται στην υπηρεσία AWS API Gateway. Εκτελεί με απλά λόγια, τον ρόλο ενός API το οποίο δέχεται αιτήματα από τον χρήστη (requests) και απαντάει με την ανάλογη απάντηση (response). Όπως φαίνεται και στην εικόνα 1*, υπάρχει αμφίδρομη επικοινωνία μεταξύ της web εφαρμογής και του Api gateway για τον παραπάνω λόγο.

Για να εμφανίσει δεδομένα η web εφαρμογή, χρειάζεται να τα "ζητήσει" απο το cloud οπου είναι αποθηκευμένα. Επειτα,η κατάλληλη lambda function για το συγκεκριμένο request θα αναλάβει να ζητήσει από την αντίστοιχη cloud υπηρεσία (βάση δεδομένων, iot subscription κλπ) την ανάκτηση των δεδομένων αυτών. Σε κάθε περίπτωση θα απαντήσει με μια απάντηση (response) είτε θετική με τα δεδομένα και ένα αναγνωριστικό οτι "όλα πήγαν καλά" status code: 200 ή με σφάλμα ανάλογο της αιτίας (συνήθως status code: 4XX ή 5XX ανάλογα αν η ευθύνη είναι του client ή του server). Μια ροή συνηθισμένη request-response μπορεί να φανει στην παρακάτω εικόνα.



Εικόνα 4.8 Ποή request-response AWS Api Gateway

Συγκεκριμένα, στην εφαρμογή ENERCOST υπάρχουν κάποια paths που ενεργοποιούν τα παρακάτω API requests:

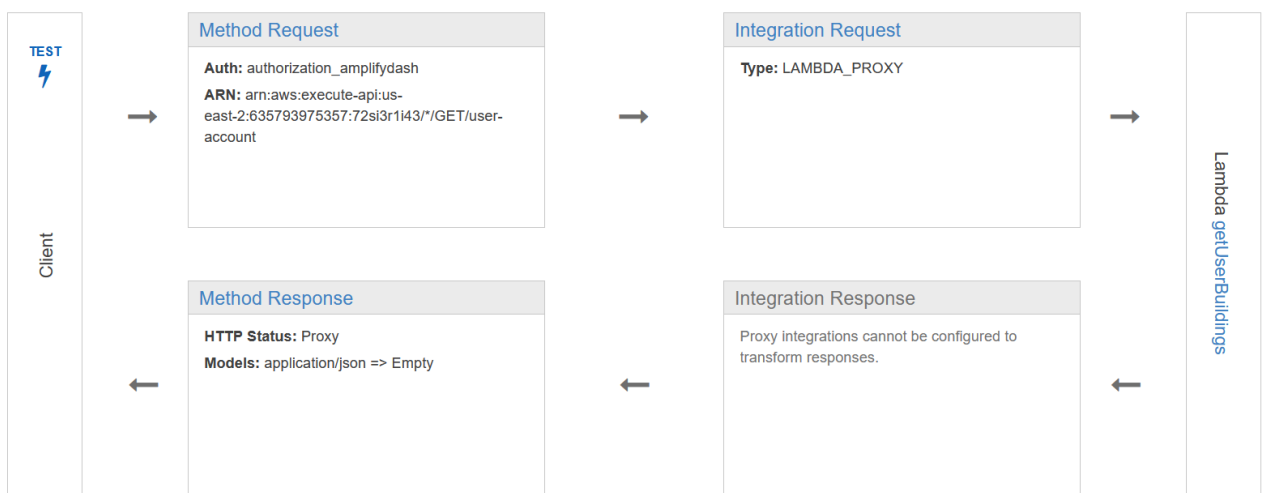
- **/user-account {GET}**: Το συγκεκριμένο path ζητάει απο το API Gateway την επιστροφή όλων των κτιρίων για τον χρήστη ο οποίος είναι συνδεδεμένος. Έπειτα το API gateway καλεί την υπεύθυνη lambda function, και αφού εκεινη εκτελέσει τις απαραίτητες ενέργειες και υπηρεσίες, επιστρέφει πίσω στο API Gateway το αποτέλεσμα και εκεινο με την σειρά του μέσω http response πίσω στον χρήστη. Ίδια φιλοσοφία ακολουθούν και τα παρακάτω.
- **/user-account/building {GET}**: Αίτημα για πληροφορίες κτιρίου. Για το συγκεκριμένο αίτημα, ο χρήστης πρέπει να προσθέσει στο request μια μεταβλητή με το συγκεκριμένο κτίριο που χρειάζεται τα δεδομένα.
- **/user-account/building {PATCH}**: Αίτημα για ανανέωση δεικτων κατανάλωσης κτιρίου. Το συγκεκριμένο αίτημα πραγματοποιείται όταν ο χρήστης επιθυμεί την ανανέωση των τιμών των δεικτών κατανάλωσης (π.χ Απο 0.15ευρω ανα kwh να αλλάξει σε 0.20ευρω ανα kwh, σε περίπτωση υποτίμησης του συμβολαίου).
- **/user-account/building/data {GET}**: Αίτημα για ανάκτηση δεδομένων κατανάλωσης στην αντίστοιχη μονάδα μέτρησης για την καθένα (νερό,

cong,ρεύμα) μαζί με τις αντίστοιχες τιμές σε κατανάλωση σε ευρώ στον καθένα ξεχωριστά.

- **/user-account/info {GET}**: Αίτημα για ανάκτηση πληροφοριών του εκάστοτε συνδεδεμένου χρήστη.

Αξίζει να αναφερθεί, ότι για να ολοκληρωθεί οποιαδήποτε παραπάνω ενέργεια, απαιτείται από τον χρήστη (client) η προσκόμιση ενός token (X-Authorization Token) στο κομμάτι του header κατα την αποστολή του request. Αυτό, προσκομείται κατα την αυθεντικοποίηση του χρήστη στην εφαρμογή.

/user-account - GET - Method Execution



Εικόνα 4.9 Ροή request-response κατά την διάρκεια του request "[website url]/user-account" {GET}

Στην παραπάνω εικόνα φαίνεται

AWS Cognito: Το AWS Cognito είναι μια πλήρως διαχειριζόμενη υπηρεσία που προσφέρεται από το Amazon Web Services, σχεδιασμένη για να αυτοματοποιεί την ταυτοποίηση και την εξουσιοδότηση (authentication/authorization) χρηστών για ολοκληρωμένες εφαρμογές.

Καθώς η εφαρμογή ENERCOST επιτρέπει μόνο σε συγκεκριμένους χρήστες να έχουν δικαίωμα στην προβολή και επεξεργασία των δεδομένων για το συγκεκριμένο κτίριο, χρησιμοποιώντας την υπηρεσία Cognito, παρέχεται στην εφαρμογή ένα ολοκληρωμένο διαδραστικό περιβάλλον διαχείρισης χρηστών λεγόμενο ως User Pool.

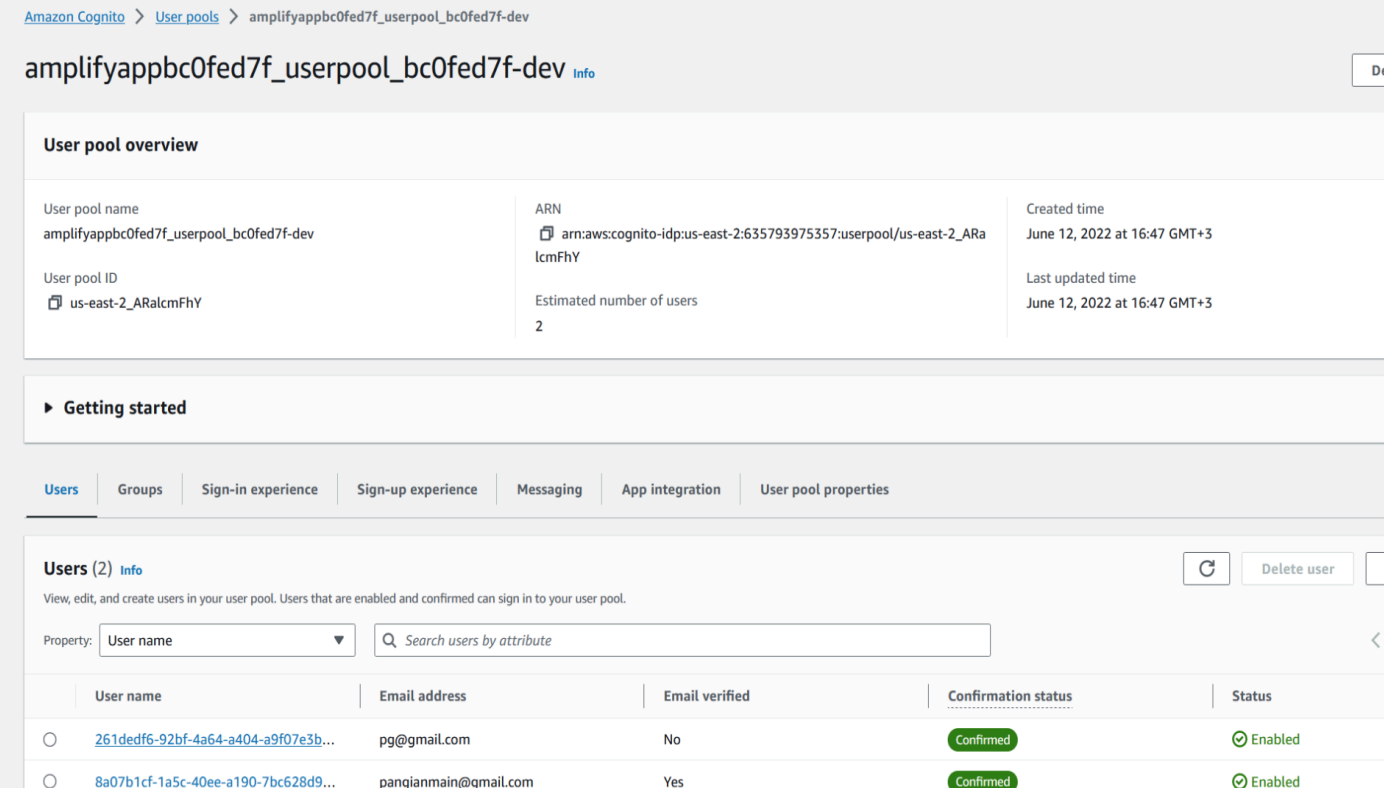
Ουσιαστικά, είναι ένα ευρετήριο χρηστών στο AWS Cognito. Επιτρέπει στους χρήστες να εγγραφούν και να συνδεθούν στην εφαρμογή ή να συνδεθώ με

λογαριασμό σε έναν τρίτο πάροχο υπηρεσιών (όπως Google account, Facebook, LinkedIn...).

Πέρα από την αποθήκευση των στοιχείων των χρηστών, ένα User Pool προσφέρει επιπλέον λειτουργίες, όπως η πολλαπλών παραγόντων ταυτοποίηση (MFA), η ανάκτηση κωδικού πρόσβασης και η επιβεβαίωση δημιουργίας λογαριασμού.

Σε απλά ελληνικά, ένα User Pool είναι ο τόπος όπου το AWS Cognito αποθηκεύει τα στοιχεία των χρηστών και διαχειρίζεται τις διαδικασίες ταυτοποίησης και εξουσιοδότησης για την εφαρμογή.

Στην παρακάτω εικόνα φαίνονται οι χρήστες με τις πληροφορίες τους για την εφαρμογή ENERCOST (δεν υπάρχουν ακόμη πραγματικοί χρήστες καθώς τρέχει πιλοτικά για τις ανάγκες της διπλωματικής).



The screenshot shows the AWS Cognito console interface for a user pool named 'amplifyappbc0fed7f_userpool_bc0fed7f-dev'. The 'Users' tab is active, showing a list of two users. The first user has an email address 'pg@gmail.com' and is confirmed. The second user has an email address 'pangianmain@gmail.com' and is also confirmed. The console includes navigation tabs for 'Users', 'Groups', 'Sign-in experience', 'Sign-up experience', 'Messaging', 'App integration', and 'User pool properties'. A search bar and a 'Delete user' button are also visible.

User name	Email address	Email verified	Confirmation status	Status
261dedf6-92bf-4a64-a404-a9f07e3b...	pg@gmail.com	No	Confirmed	Enabled
8a07b1cf-1a5c-40ee-a190-7bc628d9...	pangianmain@gmail.com	Yes	Confirmed	Enabled

Εικόνα :4.10 User pool για την εφαρμογή ENERCOST.

Για την συγκεκριμένη εφαρμογή, η δημιουργία νέου χρήστη δημιουργείται αποκλειστικά μέσω του διαχειριστή της εφαρμογής, αυτό είναι επίσης εφικτό μέσα από το διαχειριστικό του Cognito user pools, όπου ο χρήστης μπορεί να παρεμβαινει στην διαχείριση των χρηστών.

Επίσης σημαντική, είναι η υπηρεσία του Cognito, Identity Pools. Η χρησιμότητα αυτής είναι να επαναχρησιμοποιω τα στοιχεία εξουσιοδότησης (authorization) του είδη συνδεδεμένου χρήστη στην εφαρμογή (μέσω ενός identity token το οποίο αποστέλλεται στο χρήστη αυτόματα on sign-in μέσω του AWS Cognito),

επιτρέποντας τον να έχει πρόσβαση σε άλλες υπηρεσίες της AWS.

Τα δικαιώματα του κάθε χρήστη με τον συγκεκριμένο τρόπο, περιορίζονται μέσω κάποιων κανόνων προσβασιμότητας που έχει δημιουργήσει ο διαχειριστής, ανάλογα με την ομάδα που ανήκει ο συγκεκριμένος χρήστης (user,tenant,guest...)

Μια υλοποίηση που επιτυγχάνεται με την χρήση αυτής της υπηρεσίας, είναι η δυνατότητα που έχει ο χρήστης να λάβει τιμές σε πραγματικό χρόνο απευθείας από τους αισθητήρες χωρίς την ανάγκη για API call.

Αυτό επιτυγχάνεται με την επαλήθευση της δυνατότητας του συγκεκριμένου χρήστη να χρησιμοποιεί λειτουργίες της υπηρεσίας AWS IoT Core.

Η επαλήθευση αυτή γίνεται με την συσχέτιση του αντίστοιχου identity token που έλαβε ο χρήστης κατά την είσοδο του στην εφαρμογή, με τις αντίστοιχες "άδειες" χρήσης υπηρεσιών AWS που έχουν ανατεθεί στην συγκεκριμένη συνθήκη για την ομάδα χρηστών της εφαρμογής.

Εκεί, ο ρόλος του Cognito Identity Pools, είναι να κάνει τον συνδυαστικό κρίκο για την επαλήθευση και την συσχέτιση αυτής. Δηλαδή, αν ο χρήστης A ανήκει στο group χρηστών 1 που τους έχει επιτραπεί η χρήση των λειτουργιών της υπηρεσίας AWS IoT, τότε θα του επιτραπεί να μπορέσει να λάβει δεδομένα από τους αισθητήρες αυτούς (να κάνει subscribe) σε πραγματικό χρόνο.

Η εφαρμογή των υπηρεσιών του AWS Cognito στο front-end application αλλά και άλλων όπως η συνδρομή (subscription) κατευθύνει από το front-end application μέσω του AWS IoT Core, επιτυγχάνονται μέσω βιβλιοθηκών που προσφέρουν οι υπηρεσίες αυτές.

Για λόγους απλότητας όμως, έχει χρησιμοποιηθεί για την υλοποίηση του ENRCOST μια ακόμη υπηρεσία με το όνομα AWS Amplify. Η υπηρεσία αυτή, διευκολύνει την χρήση πολλών υπηρεσιών της σουίτας από το κομμάτι της front-end εφαρμογής.

AWS Amplify: Ο σκοπός της, είναι να απλοποιήσει τον τρόπο με τον οποίο το κομμάτι του front-end (είτε web εφαρμογή είτε mobile) αλληλεπιδρά με cloud υπηρεσίες της AWS.

Προσφέρει μια εξειδικευμένο περιβάλλον γραμμής εντολών (Amplify CLI) όπου από εκεί μπορείς να δημιουργήσεις το περιβάλλον της front-end εφαρμογής, επιλέγοντας την γλώσσα και τις τεχνολογίες που θέλεις να χρησιμοποιήσεις (Javascript+React για την εφαρμογή ENRCOST). Συνδέεται αυτόματα στο AWS και αποθηκεύει τοπικά τα στοιχεία του AWS λογαριασμού, επιτρέποντας την εύκολη μελλοντική ενσωμάτωση AWS υπηρεσιών στην εφαρμογή.

Επίσης, μπορεί να κάνει και hosting της εφαρμογής σε ειδικούς server στην AWS σουίτα, ανακουφίζοντας τον δημιουργό από την δουλειά του hosting. Εκτός αυτού, υποστηρίζει την δυνατότητα συνεχούς ανάπτυξης (λεγομενο και ως CI/CD) της εφαρμογής με την δυνατότητα της αυτόματης ενσωμάτωσης του project από το

github στον server, αλλά και το κομμάτι του testing του κώδικα σε πραγματικό χρόνο. Επίσης, υποστηρίζει την επιλογή custom domain για την διαδικτυακή εφαρμογή που φιλοξενείται στον server.

Αξίζει να αναφερθεί, ότι το hosting μέσω Amplify φιλοξενείται στην υπηρεσία του AWS S3, αλλά αυτό δεν απασχολεί τον δημιουργό, καθώς το Amplify την χρησιμοποιεί στο παρασκήνιο.

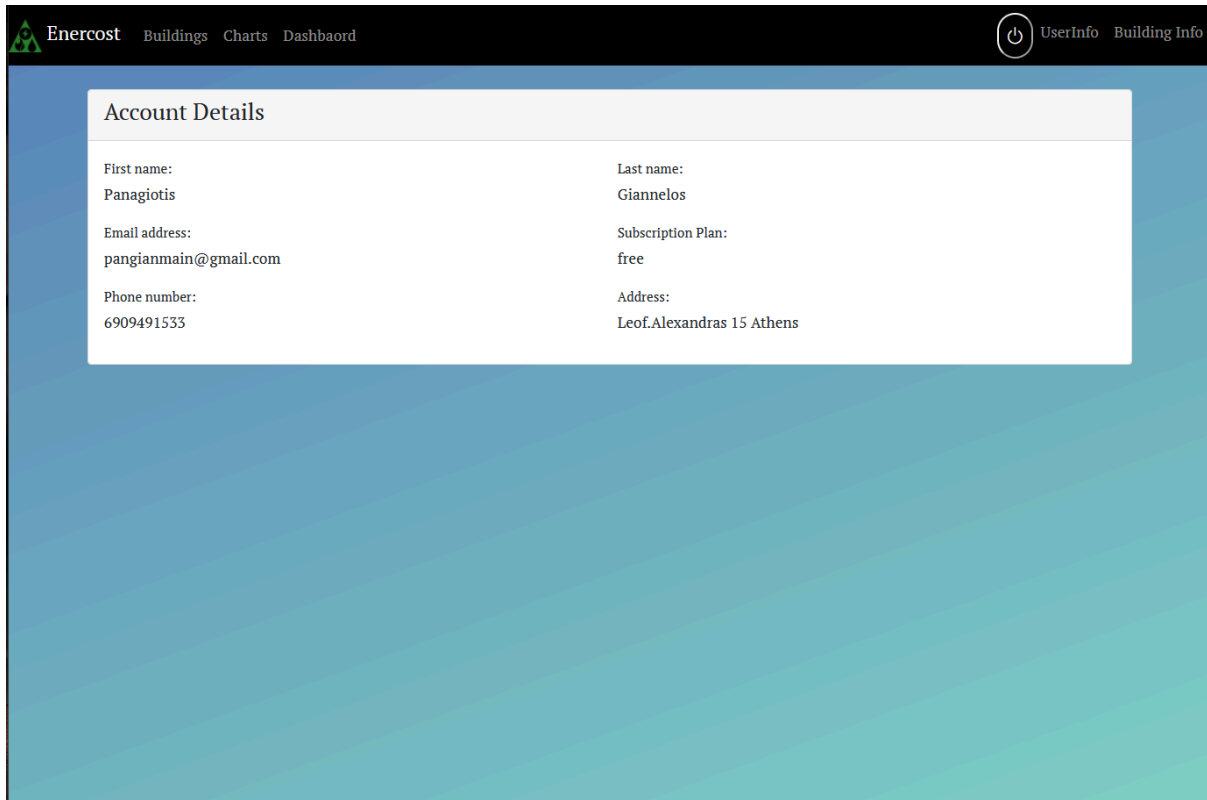
Στην εφαρμογή ENERCOST, ο λόγος χρήσης της υπηρεσίας Amplify, είναι για την εύκολη χρήση των υπηρεσιών Cognito και IoT Core, μέσω των εξειδικευμένων βιβλιοθηκών που προσφέρει στην ανάπτυξη του front-end περιβάλλοντος. Φυσικά, το front-end περιβάλλον αυτό, φιλοξενείται με την σειρά του στον server μέσω της λειτουργίας hosting, διευκολύνοντας έτσι την λειτουργία CI/CD.

FRONT-END interface: Το μόνο σημείο της εφαρμογής από όλες τις τεχνολογίες και υπηρεσίες που αναφέρθηκαν, που έρχεται σε άμεση επαφή ο χρήστης. Σαν κάθε άλλη διαδικτυακή εφαρμογή, είναι υπεύθυνο για την απεικόνιση οποιονδήποτε στοιχείων ζητηθούν από τον χρήστη, όπως και για την αποστολή δεδομένων πίσω στον server (cloud).

Παρακάτω, αναφέρονται μερικά σενάρια χρήσης, όπως και οι αντιστοιχες φωτογραφίες από την αναπαράσταση τους στην front-end εφαρμογή.

- Αυθεντικοποίηση χρήστη:
 - Ο χρήστης επιλέγει το κουμπί Sign-in
 - Μέσω της βιβλιοθήκης Auth της υπηρεσίας AWS Amplify, ελέγχεται αν τα στοιχεία του χρήστη αντιπροσωπεύουν κάποιο χρήστη που υπάρχει στο AWS Cognito/ User pool της εφαρμογής.
 - Αφού βρεθεί χρήστης με τα στοιχεία αυτά, επιστρέφει πίσω στην διεπαφή χρήστη (front-end) το λεγόμενο authentication session, το οποίο περιέχει το access token, ID token, refresh token.
 - Έπειτα, γίνεται ανακατεύθυνση στην κύρια σελίδα της εφαρμογής, όπου εμφανίζονται τα στοιχεία του χρήστη.
 - Αυτόματα, στέλνει αίτημα για ανάκτηση των στοιχείων αυτών μέσω http request στο AWS API Gateway.
 - Το API, καλεί την κατάλληλη AWS Lambda function που είναι υπεύθυνη για την ανάκτηση των στοιχείων αυτών από τον πίνακα της βάση δεδομένων AWS DynamoDB.
 - Η βάση επιστρέφει τα στοιχεία του χρήστη πίσω στην Lambda function, και αυτή στην σειρά της στο API.
 - Αν η διαδικασία ανάκτησης ήταν επιτυχής, επιστρέφονται μέσω του API (http response) στη front-end εφαρμογή τα δεδομένα με τα στοιχεία του χρήστη (με status code 200-success), εάν όχι τότε επιστρέφεται μήνυμα σφάλματος με τον αντίστοιχο κωδικό.

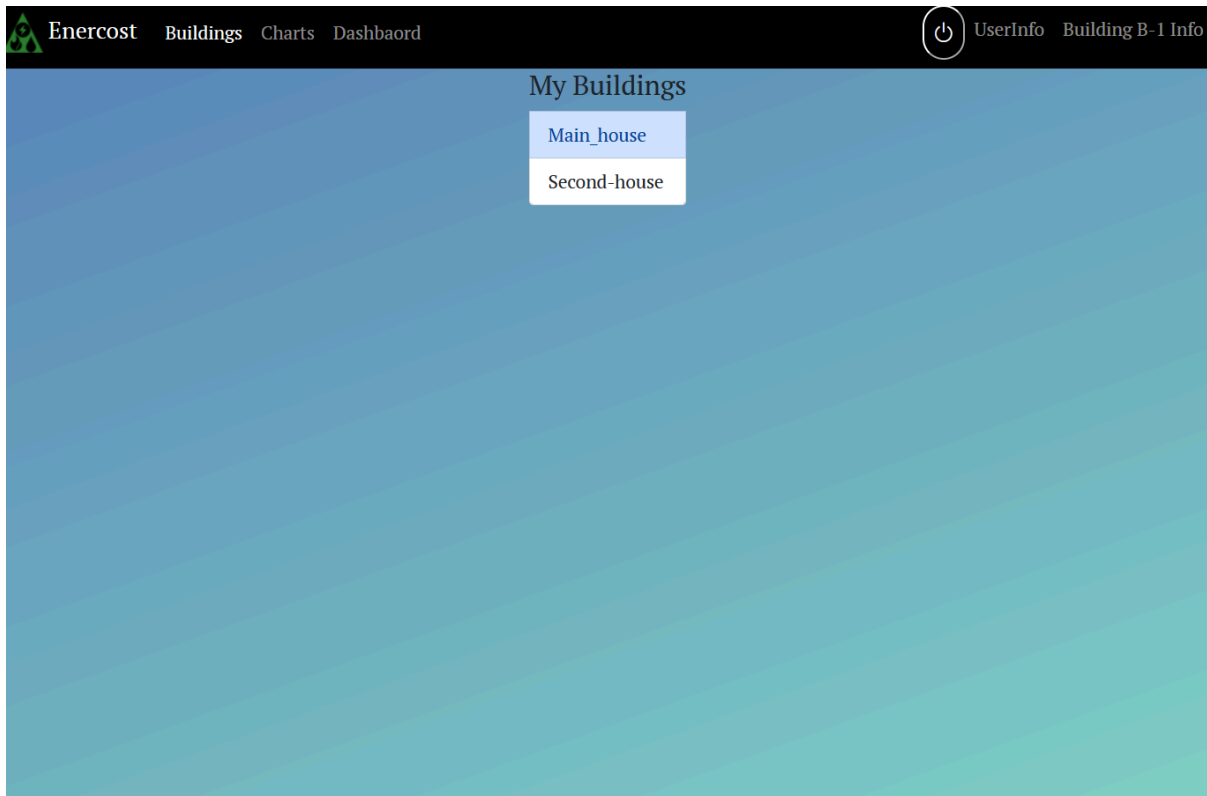
- Τα στοιχεία του χρήστη εμφανίζονται στην σελίδα.



Εικόνα 4.11 Πληροφορίες Χρήστη

- Εμφάνιση και επιλογή κτιρίου:
 - Ο χρήστης επιλέγει να δει τα κτίρια που έχει πρόσβαση πατώντας στο κουμπί menu "Buildings".
 - Αποστέλλεται http request στο AWS API Gateway ζητώντας την επιστροφή των κτιρίων για τον συγκεκριμένο χρήστη. Επίσης, μαζί με το request, προστίθεται στο header το JWT token (που χρησιμεύει στην εγκυρότητα του χρήστη από το API Gateway αλλά και έπειτα από το αντίστοιχο Lambda function).
 - Με την σειρά του το API Gateway καλεί το αντίστοιχο Lambda function...
 - Το οποίο ενημερώνεται από την βάση δεδομένων (DynamoDB) και επιστρέφει τα δεδομένα με το αντίστοιχο response, ανάλογα την απάντηση που θα πάρει από την βάση.
 - Το API επιστρέφει στο front-end σε μορφή JSON την λίστα των κτιρίων που αντιστοιχούν για τον συνδεδεμένο χρήστη.
 - Απεικονίζεται η λίστα με τα κτίρια στον χρήστη.
 - Έπειτα, ο χρήστης επιλέγει ένα συγκεκριμένο κτίριο από την λίστα.

- Τέλος αποθηκεύεται η επιλογή του στο local storage, ώστε να μπορεί εύκολα να ενημερωθεί σε κάθε σημείο της front-end εφαρμογής για την επιλογή κτιρίου που θέλει δεδομένα ο χρήστης (διαγράφεται αυτόματα όταν ο χρήστης αποσυνδεθεί από την εφαρμογή) και έπειτα εμφανίζεται στο πάνω δεξιά σημείο στην οθόνη.

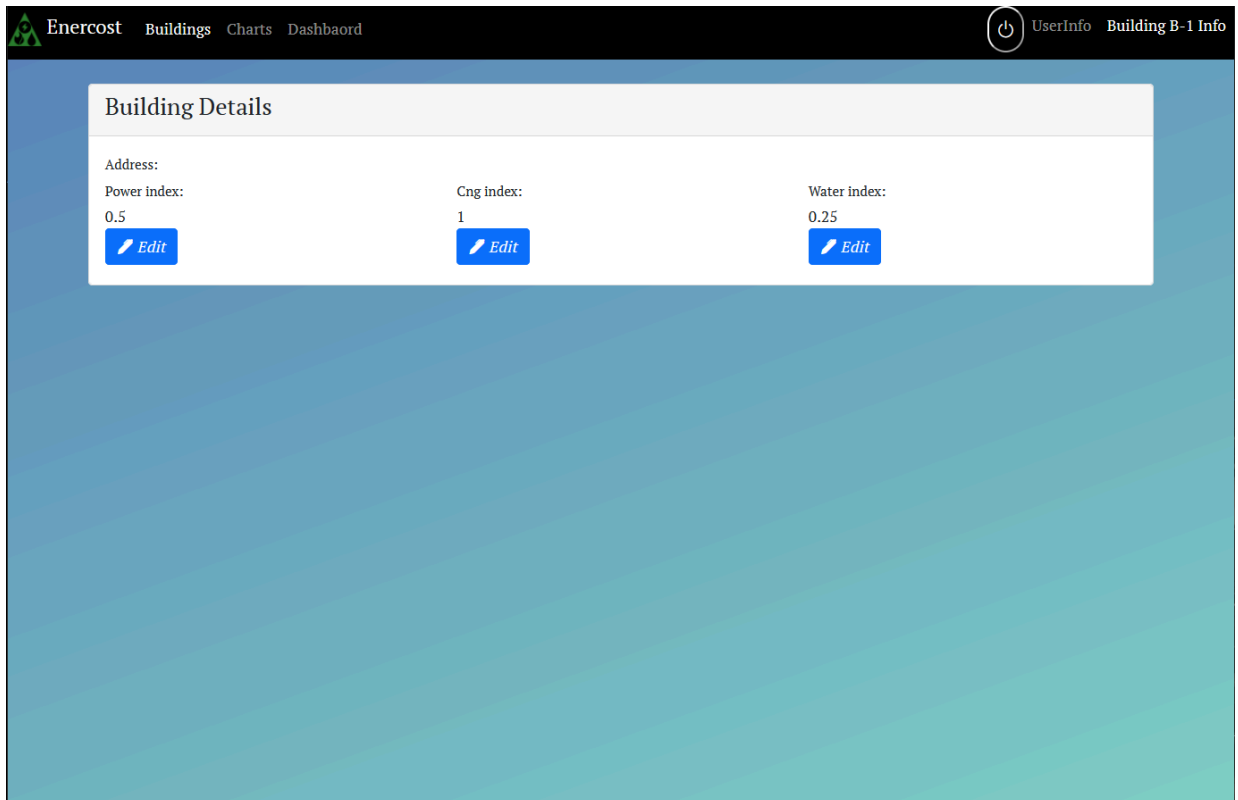


Εικόνα 4.12 Λίστα κτιρίων χρήστη

- Πληροφορίες Κτιρίου
 - Ο χρήστης επιλέγει την επιλογή "Building __ Info"
 - Αποστέλλεται http request στο AWS API Gateway ζητώντας την επιστροφή των πληροφοριών κτιρίου. Επίσης, μαζί με το request, προστίθεται στο header το JWT token (όπως και σε κάθε άλλο request) και το ID του κτιρίου.
 - Με την σειρά του το API Gateway καλεί το αντίστοιχο Lambda function...
 - Το οποίο ενημερώνεται από την βάση δεδομένων (DynamoDB) αν υπάρχει συσχέτιση κτιρίου με χρήστη και επιστρέφει τα δεδομένα με το αντίστοιχο response, ανάλογα την απάντηση που θα πάρει από την βάση.
 - Το API επιστρέφει στο front-end σε μορφή JSON τα δεδομένα..
 - Απεικονίζονται οι πληροφορίες κτιρίου στον χρήστη. Αυτές αποτελούνται από την διεύθυνση του κτιρίου και τις 3 τιμές χρέωσης. Οι

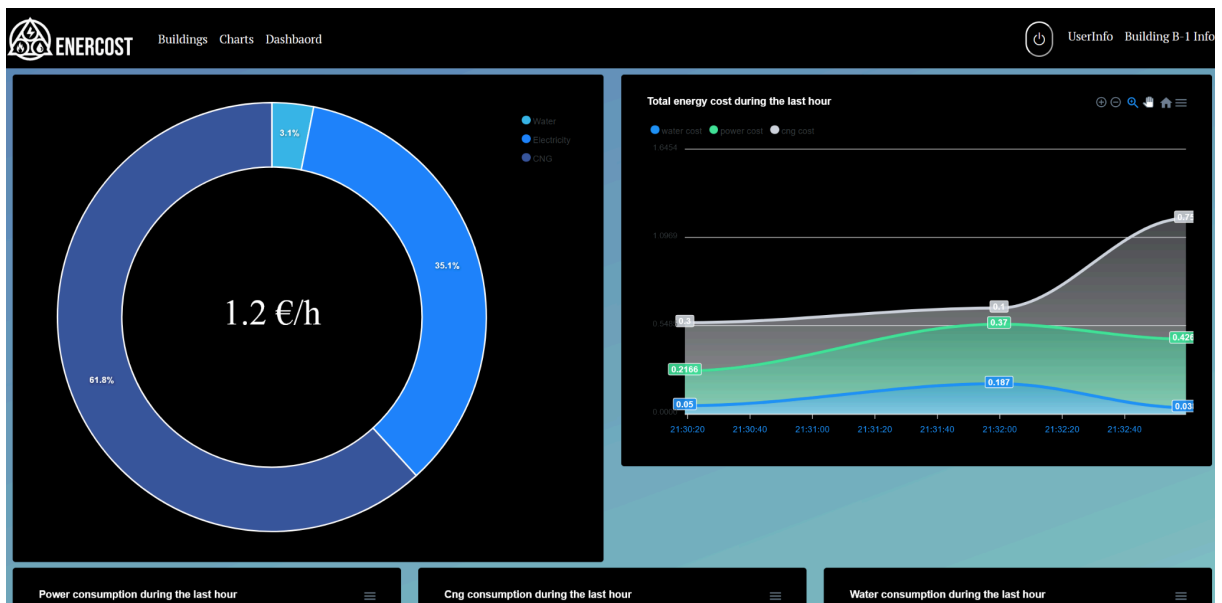
τιμές αυτές αντιστοιχούν σε τιμές χρέωσης ευρώ ανα kw/h για το ρεύμα, ανα m³ για το νερό, και ανα kw/h για το φυσικό αέριο

- Ενημέρωση υπάρχων τιμών χρέωσης κτιρίου:
 - Ο χρήστης, ενώ βρίσκεται στις πληροφορίες κτιρίου, επιλέγει να αλλάξει μια τιμή χρέωσης.
 - Αποστέλλεται http request (PATCH καθώς η αλλαγή θα πραγματοποιηθεί σε συγκεκριμένο σημείο σε υπάρχον κελί του πίνακα) στο AWS API Gateway ζητώντας την επιστροφή των πληροφοριών κτιρίου. Επίσης, μαζί με το request, προστίθεται στο header το JWT token (όπως και σε κάθε άλλο request) και το ID του κτιρίου.
 - Με την σειρά του το API Gateway καλεί το αντίστοιχο Lambda function...
 - Το οποίο ενημερώνεται από την βάση δεδομένων (DynamoDB) αν υπάρχει συσχέτιση κτιρίου με χρήστη και αλλάζει την τιμή ανάλογα.Επειτα επιστρέφει πίσω τις πλέον ενημερωμένες πληροφορίες κτιρίου με το αντίστοιχο response 200(Σε περίπτωση που η διαδικασία δεν ολοκληρωθεί, επιστρέφει το σφάλμα με το αντίστοιχο response).
 - Να αναφερθει, οτι το ίδιο Lambda function στέλνει μέσω mqtt publish, την ανανεωμένη τιμή στο Hub device, το οποίο έχει κάνει subscribe στο αντίστοιχο topic ("**buildings/upd/[building id]**")
 - Στην συνέχεια, αφού ενημερωθει η front-end με τις νέες τιμές, ανανεωνονται με την σειρά τους στην καρτέλα πληροφοριών κτιρίου.

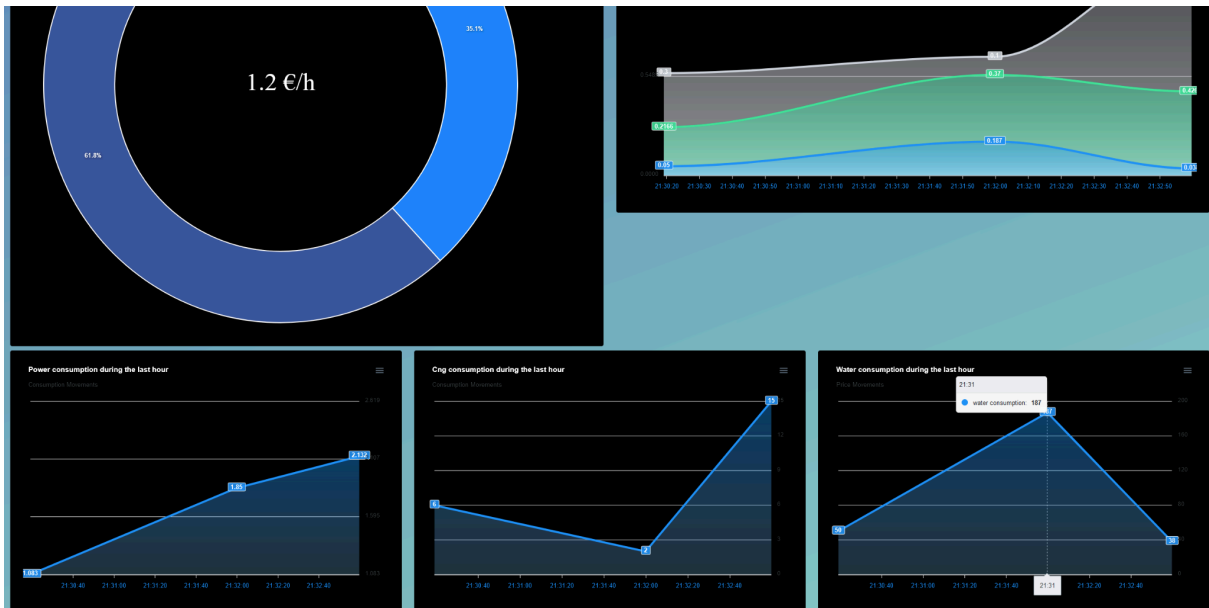


Εικόνα 4.13 Πληροφορίες Κτιρίου

Στις παρακάτω εικόνες, προβάλλεται το γραφικό περιβάλλον (dashboard) που είναι υπεύθυνο για την αναπαράσταση των μετρήσεων στις αντίστοιχες μονάδες μέτρησης όπως επίσης και το σημαντικότερο, την στιγμιαία κατανάλωση σε κόστος.



Εικόνα 4.14: Γραφήματα τιμών σε πραγματικό χρόνο (πάνω μέρος)



Εικόνα 4.15 Γραφήματα τιμών σε πραγματικό χρόνο (κάτω μέρος)

Στις παραπάνω εικόνες, στο πάνω δεξί μέρος απεικονίζεται η συνολική κατανάλωση σε ευρώ ανα ώρα των συνολικών 3ον μετρήσεων (ρεύμα, φυσικό αέριο, νερό), καθώς και ποσοστιαία με την χρήση του γραφήματος τύπου donut. Ακόμη, με το πέρασμα του κέρσορα απο την συγκεκριμένη περιοχή, παρουσιάζεται η συνολική κατανάλωση σε ευρώ ανα ώρα για την συγκεκριμένη μέτρηση.

Στο πάνω αριστερό, παρουσιάζονται σε μορφή γραφήματος το κόστος της κάθε μέτρησης στο διάστημα μιας ώρας, με την κάθε μέτρηση να απέχει διάστημα ανάλογο με την χρονική περίοδο (για την χρήση προσομοίωσης στην διπλωματική, έχει επιλεγθεί το διάστημα του 1 λεπτού).

Στην εικόνα 4.15 παρουσιάζονται άλλα 3 γραφήματα που το καθένα αναπαρηστά μία από τις τιμές κατανάλωσης σε μονάδα ενέργειας των τιμών ρεύματος, φυσικού αερίου και νερού αντίστοιχα. Το μέγιστο διάστημα υποβολής, έχει οριστεί και σε αυτά η μία ώρα.

Δεν υπάρχει ουσιαστικός λόγος για την αναπαράσταση παλαιότερων δεδομένων, καθώς η συγκεκριμένη σελίδα είναι για να παρέχει στον χρήστη την δυνατότητα απεικόνισης live δεδομένων, για ιατρικά δεδομένα, υπάρχει αντίστοιχη σελίδα που θα παρουσιαστεί στην συνέχεια.

Βήματα για την ορθή απεικόνιση των παραπάνω αναφερθέντων δεδομένων που εκτελούνται στο παρασκήνιο:

- Ο χρήστης επιλέγει το κουμπί "Dashboard" απο το menu (αφού έχει επιλεγθεί κάποιο κτίριο από το menu Buildings).

- Αποστέλλεται http request (GET) στο AWS API Gateway ζητώντας την επιστροφή των μετρήσεων κατανάλωσης κόστους και ενέργειας και για τις 3 μετρήσεις (ρεύμα-cng-νερό) για την χρονική περίοδο της τελευταίας ώρας. Επίσης, μαζί με το request, προστίθεται στο header το JWT token (όπως και σε κάθε άλλο request) και το ID του κτιρίου.
- Ταυτόχρονα, η εφαρμογή κάνει απευθείας subscribe μέσω της βιβλιοθήκης AWS Amplify στην συσκευή (hub device) που είναι υπεύθυνη για την αποστολή των δεδομένων στο cloud (όπως και για την δημιουργία των δεδομένων στο επίπεδο της διπλωματικής όπως αναφέρθηκε και στα προηγούμενα κεφάλαια).
Αυτή η ενέργεια, δίνει την δυνατότητα στον χρήστη να λαμβάνει ζωντανά τα δεδομένα, χωρίς να πρέπει πρώτα να καταχωρηθούν στην βάση και να τα λάβει έπειτα κάνοντας Polling (επαναλαμβανόμενα GET requests κάθε μια καθορισμένη χρονική περίοδο).
Είναι σημαντικό να αναφερθεί, ότι αυτή η λειτουργία χρησιμοποιεί μόνο μία cloud υπηρεσία για την ζωντανή μετάδοση των δεδομένων (την λειτουργία mqtt subscription της υπηρεσίας AWS IoT Core) σε αντίθεση με 3 και σημαντικά πιο κοστοβόρες και χρονοβόρες υπηρεσίες (AWS API gateway για το request-response, AWS Lambda function για την διαχείριση του request με την κατάλληλη ενέργεια και επιστροφή των αντίστοιχων πληροφοριών, AWS Timestream για την αναχαίτιση των δεδομένων).
- Απεικόνιση σε ζωντανή ροή των δεδομένων που λαμβάνει μέσω της παραπάνω διαδικασίας στα αντίστοιχα γραφήματα της σελίδας της εφαρμογής (όπως παρουσιάζονται στις παραπάνω 2 εικόνες).

Κεφάλαιο: 5.1 Ανάλυση απαιτούμενων αλλαγών ή προσθήκες στην υπάρχων serverless αρχιτεκτονική.

Για αρχή, να αναφέρουμε πως για την δημιουργία της εφαρμογής, λήφθηκε υπόψη σοβαρά η μελλοντική δυνατότητα, από μια απλή εφαρμογή διπλωματικής, να μπορεί να εξελιχθεί σε ένα εμπορικό προϊόν με ελάχιστες αλλαγές και προσθήκες.

Το παρόν υποκεφάλαιο είναι εστιασμένο στην μετάβαση αυτή ως προς το κομμάτι της εφαρμογής καθεαυτής. Με αυτό, εννοείται το κομμάτι του backend το οποίο έχει φτιαχτεί όλο στο cloud της Aws χρησιμοποιώντας serverless υπηρεσίες, και το front end το οποίο είναι γραμμένο όπως αναφέρθηκε και στα προηγούμενα κεφάλαια σε React-redux.

- **BACK-END:** Οι αλλαγές σε αυτό το κομμάτι, θα είναι ελάχιστες καθώς όντας serverless σημαίνει ότι μπορεί να κάνει scale-up (να μεγαλώσει) αυτομάτως ανάλογα τις απαιτήσεις του trafic που δέχεται. Βέβαια, σε μια περίπτωση προσθήκης multi-tenancy λειτουργίας (δηλαδή χρήστες με διαφορετικά

δικαιώματα μεταξύ τους) η μόνη αλλαγή που θα χρειαζόταν είναι η δημιουργία user groups στην υπηρεσία AWS Cognito, όπου το κάθε group θα έχει τα αντίστοιχα δικαιώματα. Π.χ μπορεί μια ομάδα χρηστών να μην επιτρέπεται να αλλάξουν τους υπάρχοντες δείκτες τιμής ενέργειας, σε αυτή την περίπτωση απλά αφαιρούμε το δικαίωμα των χρηστών αυτών να γράφουν πάνω στην βάση DynamoDB.

Σημαντικό, είναι επίσης η δημιουργία αυτόματης εισαγωγής και fleet control (έλεγχος στόλου) των συνδεδεμένων IoT συσκευών στο cloud μέσω υπάρχοντων λειτουργιών μέσα στο AWS IoT Core που για λόγους απλότητας, δεν προτιμήθηκαν σε επίπεδο διπλωματικής.

Επίσης, σωστό θα είναι να παρακολουθείται η κοστολόγηση του AWS από την ομάδα IT (μέσω της υπηρεσίας AWS Billing), με σκοπό την ανάλυση του κόστους και την προτροπή σε τυχόν αλλαγές, περιορισμούς ή ανατιμολογήσεις του προϊόντος με βάση αυτής.

- **FRONT-END:** Καθώς αυτό αντιπροσωπεύει μόνο το οπτικό κομμάτι, δεν χρειάζεται καμία αλλαγή ως προς την λειτουργικότητα με τα υπάρχοντα δεδομένα. Ός αναφορά το hosting, καθώς τρέχει στο AWS Amplify, το κόστος ανεβαίνει με την χρήση όπως και με τις άλλες serverless υπηρεσίες, εξασφαλίζοντας όμως σταθερή διαθεσιμότητα της ιστοσελίδας.

Επίσης, η υποστήριξη CI/CD του Amplify κάνει την ανάπτυξη νέων λειτουργιών και εκδόσεων κατά πολύ ευκολότερη καθώς έχει την δυνατότητα να κρατά versioning από το Github αλλά και να κάνει fallback (γυρίζει πίσω) σε παλιά version σε περίπτωση που απαιτηθεί. Φυσικά, η εφαρμογή παραμένει ενεργή κατά την διάρκεια του update.

Κάποια πιθανά νέα features της εφαρμογής όταν φτάσει σε επίπεδο παραγωγής, θα μπορούσαν να είναι τα παρακάτω:

- Δημιουργία multi-tenancy, όπως ειπώθηκε και παραπάνω. Δηλαδή δικαιώματα χρηστών ανάλογα με την κατηγορία τους (admins, users), όπως και επίσης δυνατότητα δημιουργίας απλών χρηστών (μόνο με δυνατότητα προβολής των τιμών) αναθέτοντας επιλεκτικές μετρήσεις.
- Προσθήκη ειδοποιήσεων μέσω email ή μηνυμάτων. Κάτι τέτοιο θα ήταν πολύ εύκολα εφικτό με την δημιουργία κάποιων IoT Rules όπου όταν μια τιμή από μια εισερχόμενη μέτρηση στο cloud, ακολουθεί την συνθήκη, μέσω AWS SES για email ή SNS για μηνύματα, ο αντίστοιχος διαχειριστής του κτιρίου να λάβει ειδοποίηση.

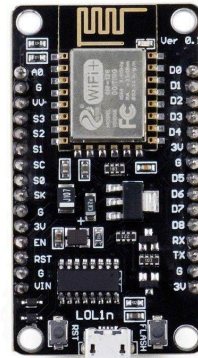
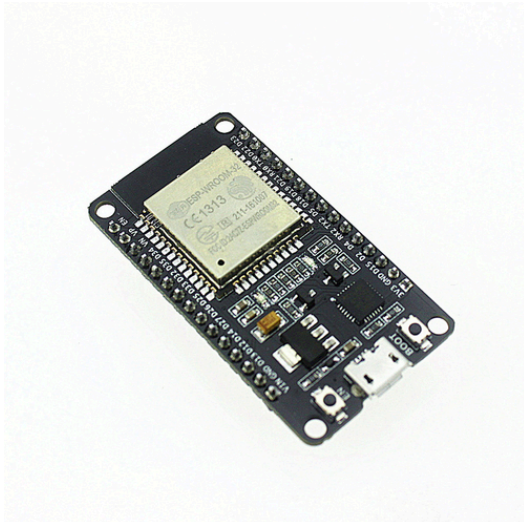
- Δημιουργία mobile εφαρμογής με σκοπό την ευκολότερη πρόσβαση μέσω android/ios λογισμικού. Ξανά, η υπηρεσία της AWS, Amplify επιτρέπει την εύκολη αλληλεπίδραση με το cloud όχι μόνο για της web εφαρμογές, αλλά και για τις mobile καθώς προσφέρει τις ίδιες βιβλιοθήκες, υποστηρίζοντας όλες τις γνωστές γλώσσες προγραμματισμού για mobile εφαρμογές (όπως React native, Kotlin, Flutter, Swift, Java).
- Δυνατότητα "κλωνοποίησης" κάποιον στοιχείων της εφαρμογής με σκοπό την εύκολη δημιουργία παραπλήσιας εφαρμογής για απεικόνιση διαφορετικών δεδομένων παραπλήσιας φύσης (π.χ γραφήματα για διαφορετικές μετρήσεις και αντί για κτίρια χωράφια με καλλιέργειες). Έτσι συμπεριλαμβανομένης και της backend serverless αρχιτεκτονικής, μπορεί να βγει στο εμπόριο παραπάνω από 1 IoT λύση με μικρες αλλαγές.

5.2 Απαραίτητες προσθήκες σε physical και network επίπεδο.

Καθώς όπως αναφέρθηκε, η κύρια αρχιτεκτονική σε επίπεδο συσκευών, βασίζεται σε συσκευές κόμβους εγκατεστημένες σε κυκλώματα ενέργειας (δίκτυο ύδρευσης, ηλεκτρολογικός πίνακας, κύκλωμα φυσικού αερίου), οι οποίες έχουν ενσωματωμένους τους απαιτούμενους αισθητήρες με σκοπό να καταγράψουν την κατανάλωση. Έπειτα, επικοινωνούν με μια κεντρική συσκευή, η οποία και έχει πρόσβαση στο διαδίκτυο προκειμένο να επικοινωνεί με το cloud.

Επανατονίζεται ότι για το επίπεδο της διπλωματικής, οι μετρήσεις αυτές δημιουργούνται (random-generated) απο την κεντρική συσκευή, εξαλείφοντας τις συσκευές "κόμβους".

Μιλώντας για ενα start-up project για αρχή, θα μπορούσαν σαν συσκευές κόμβοι αλλά και κεντρική, να χρησιμοποιηθούν οι ευρέως γνωστές mcu ESP8266 (για τους κόμβους) και ESP32.



Εικόνες 5.1 Στα δεξιά ESP32 στα αριστερά ESP8266

ESP8266: Η χρήση της εν λόγω συσκευής θα καλύπτει το κομμάτι των κόμβων. Βασικά χαρακτηριστικά επιλογής τους, η χαμηλή τιμή τους (κάτω των 10 ευρώ), η ευκολία προγραμματισμού τους, η υποστήριξη πληθώρας αισθητήρων, η χαμηλή κατανάλωση και η σύνδεση τους με WiFi ή πρωτοκόλλου ESP-NOW όπου επιτρέπει την δημιουργία ενός mesh δικτύου με ένα ESP32 σαν κεντρική συσκευή (master) και κόμβους είτε ESP32 ή ESP8266.

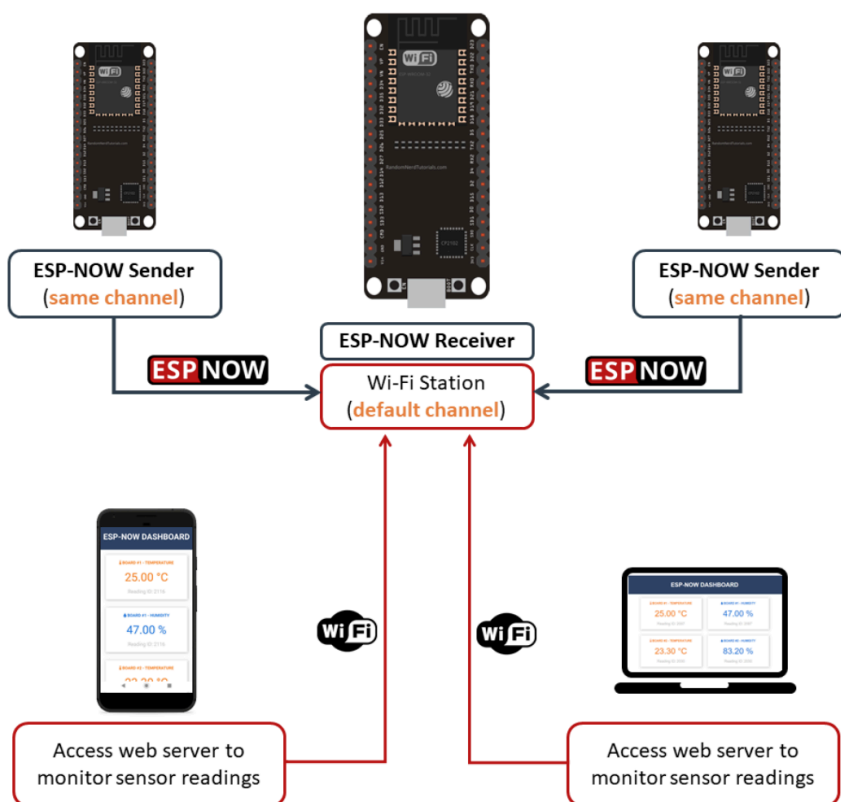
ESP32: Με χρήση ως κεντρικής συσκευής (mother-device), μπορεί εκτός από την επίτευξη της επικοινωνίας με το cloud μέσω mqtt, η επικοινωνία μέσω κλειστού τύπου διασύνδεσης με τα end-devices, μέσω πρωτοκόλλου ESP-NOW. Το πρωτόκολλο αυτό διασφαλίζει ασφαλή επικοινωνία μεταξύ των συσκευών καταγραφής μετρήσεων και την κεντρική συσκευή. Επίσης πολύ σημαντικό, το γεγονός ότι οι τελικές συσκευές κόμβοι (end-devices) δεν χρειάζονται πρόσβαση στο διαδίκτυο, καθώς και επίσης το range που επιτυγχάνεται, είναι πολύ μεγαλύτερο αυτού του κοινού WiFi και φτάνει ακόμη και τα 480 μετρα σε ευθεία βολή.

Βέβαια, ως κεντρική συσκευή η οποία μπορεί να λάβει και να στείλει σε άλλες, είναι απαραίτητη η χρήση της πιο εξελιγμένης συσκευής ESP32 απο το απλό ESP8266. Με κόστος μικρότερο των 20 ευρώ, μεγαλύτερη ισχύ και δυνατότητες από τον προκάτοχό του και δυνατότητα λειτουργίας και ως web server, την καθιστά μια εξαιρετική λύση για το πρόβλημα.

Μάλιστα, χρησιμοποιείται για τον σκοπό του data generator και συσκευής επικοινωνίας με το cloud στο συγκεκριμένο project.

Μια μελλοντική προέκταση χρησιμοποιώντας την συσκευή αυτή για τον ρόλο αυτό, είναι εκμεταλλεύοντας την ικανότητα του ως web server δημιουργώντας μια στατική σελίδα, την οποία θα μπορούν άμεσα οι χρήστες που βρίσκονται στο κτίριο να δουν τις μετρήσεις live μέσα από τοπικό δίκτυο, χωρίς την χρήση σύνδεσης στο διαδίκτυο.

Στην παρακάτω εικόνα φαίνεται μια γρήγορη αναπαράσταση μιας τέτοιας λύσης.

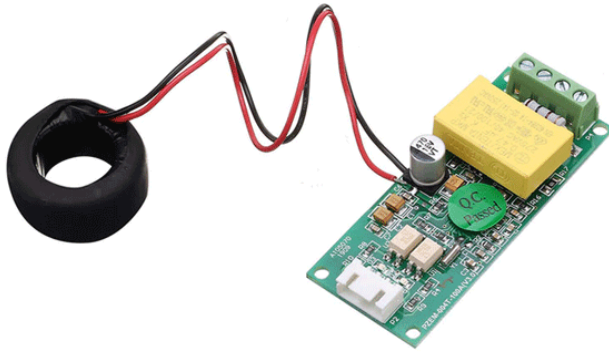


Εικόνα 5.2 Πρόχειρη αναπαράσταση λύσης με ESP-NOW

Τέλος, παρακάτω, αναφέρονται κάποιου είδους πιθανοί αισθητήρες, οι οποίοι θα μπορούσαν να καταγράψουν τις απαιτούμενες μετρήσεις για την εφαρμογή ENERCOST καθώς και προτεινόμενα σημεία τοποθέτησης τους σε ένα κτήριο.

Κατανάλωση Ηλεκτρικής Ενέργειας:

Η επιλογή ενός Αισθητήρα έντασης-τάσης ρεύματος rzem004t (παρακάτω φωτογραφία) συνδεδεμένο με τον ηλεκτρολογικό πίνακα και το ESP8266 για την μέτρηση της έντασης και της τάσης. Το πηνίο που φαίνεται στην φωτογραφία, είναι υπεύθυνο για την μέτρηση της έντασης, καθώς τοποθετείται "αγκαλιάζοντας" το καλώδιο, επίσης άλλα 2 καλώδια (δεν απεικονίζονται στην φωτογραφία) ξεκινούν απο την συσκευή ωστε να μετρήσουν την τάση συνδέοντας το ένα στην φάση και το άλλο στον ουδέτερο.



Ο αισθητήρας τοποθετείται πριν ή σε συγκεκριμένο σημείο μέσα στον ηλεκτρολογικό πίνακα σε περίπτωση που επιθυμείτε ο έλεγχος συγκεκριμένης γραμμής. Κάτι τέτοιο φυσικά, αφήνει περιθώρια εξέλιξης μιας ολοκληρωμένης plug and play συσκευής με ενσωματωμένο τον ESP8266 ή ESP32 (όπως κάνει η παγκοσμίως γνωστή Shelly, οι οποία παράγει συσκευές μέτρησης ρεύματος) η οποία θα έχει την αποκλειστική χρήση κόμβου (end-device) για το προτζεκτ αυτό.

Κατανάλωση Φυσικού Αερίου:

Εάν μιλάμε για την περίπτωση που το κτήριο προμηθεύεται φυσικό αέριο από την ΕΔΑ, απαγορεύεται να εμπλακεί μη εξειδικευμένο προσωπικό στο υπάρχων δίκτυο, όπως και να πειραχτεί σε κάθε περίπτωση ο υπάρχον μετρητής του προμηθευτή. Γι Αυτό, και προτείνεται η εγκατάσταση από εξειδικευμένο συνεργείο, ενός δεύτερου μετρητή ύστερα από τον δημόσιο. Έτσι, μπορεί να ενσωματωθεί μια συσκευή πάνω στο δεύτερο και πλέον ιδιωτικό ρολόι.

Βέβαια, πολλοί μετρητές στην αγορά, έχουν αναμονή για σύνδεση με συσκευή, που μεταφέρουν τα δεδομένα αυτόματα, μια τέτοια λύση, καθιστά αχρείαστη την επιλογή κάποιου είδους αισθητήρα επάνω στον μετρητή, καθώς τα δεδομένα θα μεταφέρονται στο ESP8266/32 απο τον ίδιο τον μετρητή.

Ένα παράδειγμα, είναι ο μετρητής Honeywell/Elster BK-G4, ο οποίος μάλιστα είναι και διαθέσιμος στην Ελληνική αγορά. Διαθέτει εξάρτημα μέτρησης μαγνητικών παλμών κατα την περιστροφή του τελευταίου ψηφίου (απεικονίζεται στην παρακάτω εικόνα)



5.3 Μετρητής Elster BK + Εξάρτημα καταγραφής

Σε περίπτωση που ο δεύτερος μετρητής δεν διαθέτει τέτοια δυνατότητα, υπάρχουν μη επεμβατικές (non-invasive) λύσεις, η οποίες μπορούν να "διαβάσουν" την τελευταία ένδειξη του ταχυμετρου του μετρητή. Μία από αυτές, που χρησιμοποιείται ήδη από μετρητές του εμπορίου, είναι η ενσωμάτωση ενός μαγνητικού αισθητήρα (reed switch) κάτω από το τελευταίο ψηφίο του μετρητή. Καθώς ο τρόπος λειτουργίας των μετρητών αυτόν, προϋποθέτει την ύπαρξη μαγνήτη στο τελευταίο ψηφίο, είναι εύκολο να υπολογιστεί η κάθε περιστροφή, και με έναν non-invasive τρόπο σαν αυτόν. Φυσικά, οι "παλμοί" αυτοί μεταβολής του state του μαγνητικού αισθητήρα, θα διαβάζονται από ένα ESP κόμβο, και θα αποστέλλονται στην κεντρική συσκευή σε κάθε περίπτωση (είτε διαθέτει υποδοχή + εξάρτημα καταγραφής, είτε σε μια περίπτωση που δημιουργηθεί ο καταγραφέας με τον παραπάνω τρόπο).

Κατανάλωση Νερού:

Το κύκλωμα του νερού, είναι πολύ πιο εύκολα διαχειρίσιμο από το κύκλωμα του φυσικού αερίου καθώς επεμβαίνοντας στο κύκλωμα της παροχής νερου(έπειτα από τον δημόσιο μετρητή) είναι σημαντικά λιγότερα επικίνδυνο. Αν και ισχύει και πάλι ο κανόνας της απαγόρευσης σε οποιαδήποτε τροποποίησης στο δημόσιο μετρητή.

Γιαυτό λοιπόν, είναι προτιμότερη η σύνδεση σε σειρά στο κύκλωμα, ενός απλού μετρητή λίτρων ανα ώρα ο οποίος μπορεί να συνδεθεί άμεσα με τον μικροελεγκτή (ESP). Ένα παράδειγμα αυτό, είναι ο YF-S201 , ενώ αν μιλάμε για καταναλώσεις μεγαλύτερες των 30L/λεπτό, ο YF-B7 όπου μάλιστα έχει και την δυνατότητα μέτρησης

και θερμοκρασίας νερού (κάτι που εύκολα προσαρμόζεται και στην εφαρμογή με μικρές τροποποιήσεις). Επίσης να αναφερθεί ότι και οι 2 βρίσκονται σε κόστος κατώτερο των 10 ευρώ!



Εικόνα 5.4 Μετρητής ποσότητας νερού YF-201 δεξιά και YF-B7 αριστερά.

Τέλος, τονίζεται, ότι κάλλιστα θα μπορούσε, αντί για πιο αυτοσχέδιες λύσεις όπως πολλές από τις παραπάνω, να χρησιμοποιηθούν έτοιμες IoT συσκευές οι οποίες συνάπτουν με το πρωτόκολλο MQTT. Όπως η shelly για το ηλεκτρικό ρεύμα και πολλές άλλες εταιρίες με "έξυπνους μετρητές" για νερό και ακόμη και φυσικό αέριο.

Αυτό, με την προϋπόθεση της δυνατότητας πρόσβασης στο διαδίκτυο αυτών, και μια μικρή ανασχεδίαση του κώδικα του central-device (esp32).

Φυσικά, το γεγονός ότι ακολουθείται μια loose coupled (αραιά συνδεδεμένη) αρχιτεκτονική πετυχαίνει την ευκολία σε αλλαγές σαν αυτές με μικρή επεμβατικότητα, αλλά επίσης και δεν αρνείται την ενσωμάτωση μεγαλύτερων αλλαγών όπου οι καταστάσεις το απαιτούν.

Ένα παράδειγμα, είναι οι περίπτωση που σε μια κατάσταση απαιτείται η χρήση συσκευών οι οποίες λειτουργούν με το πρωτόκολλο LoRaWAN. Καθώς οι LoRa network servers υποστηρίζουν MQTT, η ενσωμάτωση με ένα central-device ή ακόμη και απευθείας με το cloud, είναι εφικτές.

5.3 Λοιπές ενέργειες για την επιτυχία του στην αγορά.

Στα παραπάνω υποκεφάλαια, αναφέρθηκαν τρόποι με τους οποίους θα μπορούσε η εφαρμογή να υποστηρίξει πραγματικά σενάρια καταγραφής κατανάλωσης ενέργειας από μια κτιριακή εγκατάσταση. Βέβαια, ένα έτοιμο προς χρήση προϊόν με δυνατότητα προσαρμογής σε περιβάλλοντα με διαφορετικές παραμέτρους, μπορεί να έχει προοπτικές στην αγορά, αλλά πρέπει να ληφθούν υπόψη τα παρακάτω, ώστε να

πάρει την αξία που του αξίζει σαν ολοκληρωμένο πλέον εμπορικό προϊόν.

- **Υποστήριξη:** Όσο σημαντική είναι η υλοποίηση της εγκατάστασης σε μια κτιριακή μονάδα, άλλο τόσο είναι και η εξασφάλιση της ομαλής λειτουργίας των συσκευών, του cloud, και της εξυπηρέτησης των πελατών. Για να πραγματοποιηθούν όλα αυτά, χρειάζεται εξέλιξη στο κομμάτι της πληροφόρησης για την λειτουργία των συσκευών στο συνδεδεμένο δίκτυο.

Αυτό θα μπορούσε να επιτευχθεί εγκαθιστώντας στην συσκευή συλλογής δεδομένων (central-device) η οποία έχει και πρόσβαση στο Internet, μια λίστα με την κατάσταση λειτουργίας των συνδεδεμένων συσκευών, αλλά και της συσκευής κάθε αυτής. Την λίστα αυτή, θα μπορούσε να την προωθήσει στο cloud μέσω mqtt σε διαφορετικό topic. Με αυτό τον τρόπο, θα μπορούν οι μηχανικοί να συλλέξουν δεδομένα ως αναφορά την ανθεκτικότητα του προϊόντος, αλλά και να ελέγξουν αν απαιτείται η επίσκεψη τεχνικού στον χώρο (σε περίπτωση που το ζήτημα δεν μπορεί να επιλυθεί από απόσταση).

Στο κομμάτι του cloud, υπάρχουν έτοιμες λύσεις που προσφέρονται από την AWS, οπότε εάν ο πελάτης διαπιστώσει μη ομαλή λειτουργία στην εφαρμογή, είναι πολύ εύκολο για τους μηχανικούς να ελέγξουν την ροή της πληροφορίας στο cloud (πιθανώς μέσω λογαριασμού διαχείρισης πελατών ο οποίος έχει πρόσβαση στα απαραίτητα δεδομένα για την αντιμετώπιση προβλημάτων) και να πράξουν αναλόγως.

- **Εξέλιξη:** Φυσικά, για να είναι ένα προϊόν κερδοφόρο και το project να μεγαλώσει, χρειάζεται να δοθεί μεγάλη σημασία στο R&D του προϊόντος καθε αυτού, αλλά και η ανάπτυξη παραπλήσιων λύσεων, οι οποίες αποσκοπούν στην επίλυση παρόμοιων προβλημάτων με όσο το δυνατόν μικρότερες τροποποιήσεις στην υπάρχων αρχιτεκτονική.

Επίσης, λαμβάνοντας υπόψη την ραγδαία εξέλιξη της τεχνολογίας γύρω μας, είναι ξεκάθαρο ότι οι σύγχρονες και πρωτοποριακές λύσεις του παρόντος, σε κάποιο χρονικό διάστημα θα πάψουν να είναι ανταγωνιστικές, μοναδικές, ουσιώδης, ασφαλής και λειτουργικές σύμφωνα με τα υψηλά στάνταρ που ξεκίνησαν να ακολουθούν. Αυτό είναι και ένα ακόμη προτέρημα μιας αρχιτεκτονικής της οποίας το software είναι στο cloud. Η συνεχής εξέλιξη της AWS και των υπηρεσιών που χρησιμοποιούνται, αναγκάζουν το project να ακολουθεί τα στάνταρ των τελευταίων τεχνολογιών, αλλά και δίνει την δυνατότητα της δοκιμής νέων εργαλείων που πιθανώς να εκσυγχρονίσουν την εφαρμογή.

- **Εμπορικότητα:** Για την επιτυχη εμπορικοποίηση ενός ολοκληρωμένου προϊόντος, είναι απαραίτητο να καθοριστεί μια στοχευμένη ομάδα πελατών. Στην περίπτωση αυτού του προϊόντος, η στοχευμένη ομάδα αποτελείται από

κτιριακές υποδομές που χρησιμοποιούν έναν από τους τρεις υποστηριζόμενους τύπους κατανάλωσης: νερό, ηλεκτρική ενέργεια, ή φυσικό αέριο. Είναι σαφές ότι το προϊόν μπορεί να λανσαριστεί τόσο για ιδιωτικά κτίρια όσο και για επιχειρήσεις. Ωστόσο, είναι σημαντικό να επικεντρωθεί σε έναν από αυτούς τους δύο τομείς κατά τα πρώτα βήματα του στην αγορά.

Αυτό είναι κρίσιμο κυρίως επειδή η προσέγγιση και η διαχείριση πιθανών προβλημάτων μπορεί να διαφέρει σημαντικά μεταξύ των δύο τομέων, τόσο σε επίπεδο επιχειρηματικής στρατηγικής όσο και σε επίπεδο λύσεων.

- **Διαφήμιση:** Για την αποτελεσματική προώθηση του προϊόντος στην αγορά, είναι απαραίτητη η υλοποίηση μιας στοχευμένης διαφημιστικής εκστρατείας. Αυτή θα πρέπει να σχεδιαστεί με τέτοιο τρόπο ώστε να απευθύνεται στην επιλεγμένη στοχευμένη ομάδα, ενώ παράλληλα να επισημαίνει τα βασικά πλεονεκτήματα και τη μοναδικότητα του προϊόντος. Μερικοί αποτελεσματικοί τρόποι για το συγκεκριμένο προϊόν θα μπορούσαν να είναι:

- Μέσω από social media (ειδικό κανάλι για το προϊόν αυτό ή της εταιρείας) που θα ανεβαίνει υλικό με την υλοποίηση, επεξήγηση της λύσης αλλά ακόμη και διαγωνισμούς (giveaways) με έπαθλο το ίδιο το προϊόν, συνεργασίες και πολλά ακόμη...

- Μέσω demo projects σε κτιριακές δομές (στις οποίες θα εισέρχεται πολύς κόσμος), με σκοπό την απεικόνιση των μετρήσεων αλλά και της κατανάλωσης σε διαδραστικά tablet στους κοινόχρηστους χώρους του κτιρίου.

-Μέσω εξωτερικών συνεργασιών με εταιρείες εγκαταστάσεων επιτυχων λεβητων, ηλεκτρολόγων και υδραυλικών, εργολάβων και εταιριών ανακαίνισης και διακόσμησης.

Και φυσικά, μέσω της καλύτερης διαφήμισης που δεν είναι άλλη από την υπέρβαση των προσδοκιών του εκάστοτε πελάτη, έτσι επιτυγχάνεται η διασφάλιση μιας καλής συνεργασίας που με την σειρά της θα ωθήσει και άλλο κόσμο δίνοντας θετικό feedback.

Βιβλιογραφία

Κεφάλαιο 1

<https://www.dataversity.net/brief-history-internet-things/>

https://ethw.org/Kevin_Ashton

<https://www.elevenfifty.org/blog/the-history-of-the-internet-of-things>

<https://openconnectivity.org/connecting-past-future-origins-evolution-iot/>

<https://www.ibm.com/blog/little-known-story-first-iot-device/>

<https://www.teracomtraining.com/tutorials/teracom-tutorial-how-many-ipv6-addresses-are-there.htm>

<https://www.tibco.com/reference-center/what-is-the-internet-of-things-iot>

Κεφάλαιο 2

https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Use_of_Internet_of_Things_in_enterprises

<https://www.geeksforgeeks.org/architecture-of-internet-of-things-iot/>

<https://new.abb.com/control-systems/features/industrial-IoT-services-people-use-cases>

<https://buildings.honeywell.com/us/en>

Κεφάλαιο 3

<https://aws.amazon.com/lambda/serverless-architectures-learn-more/>

<https://www.cloudflare.com/learning/serverless/why-use-serverless/>

<https://www.statista.com/statistics/1243513/top-10-cloud-vendors-by-revenue-fiscal-quarter-global/>

<https://www.statista.com/chart/18819/worldwide-market-share-of-leading-cloud-infrastructure-service-providers/>

<https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Concepts.RegionsAndAvailabilityZones.html>

<https://docs.aws.amazon.com/iot/latest/developerguide/protocols.html>

Κεφάλαιο 4

<https://aws.amazon.com/what-is/ssl-certificate/>

<https://restfulapi.net/http-status-codes/>

Κεφάλαιο 5

<https://boldringroup.it/en/accessories-for-gas-meters/1554-pulse-generator-for-bk-series-gas-meters-inz61.html>

<https://www.espressif.com/en/solutions/low-power-solutions/esp-now>

Εικόνες

<https://www.sotatek.com/6gs-impact-on-the-future-of-iot/>

https://www.researchgate.net/figure/IoT-Number-of-devices-worldwide-from-2015-to-2025-5_fig1_351075753

https://www.researchgate.net/figure/Use-of-cloud-computing-services-in-2018-Source_fig1_348462051

<https://www.hpcwire.com/2024/01/17/google-cloud-waives-cloud-exit-fees-throws-down-gauntlet-to-aws-and-azure/>

https://www.researchgate.net/figure/Network-sample-representation-of-the-Message-Queue-Telemetry-Transport-MQTT_fig1_341563324

<https://randomnerdtutorials.com/esp32-esp-now-wi-fi-web-server/>

<https://www.google.com/url?sa=i&url=https%3A%2F%2Frandomnerdtutorials.com%2Fesp32-esp-now-wi-fi-web-server%2F&psig=AOvVaw3yVzEQg4UX0RmwHrWmhD1a&ust=1709672900952000&source=images&cd=vfe&opi=89978449&ved=0CBIQjRxqFwoTCJDY4oHC24QDFQAAAAAdAAAAABAE>

<https://www.tinytronics.nl/en/sensors/liquid/yf-b7-water-flow-sensor-with-temperature-sensor-brass-g1-2>

<https://www.hellasdigital.gr/electronics/sensors/humidity-sensor/water-flow-sensor-1-30l-min-2.0mpa-yf-s201/>