



ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΑΚΩΝ
ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ ΣΠΟΥΔΩΝ
"Τεχνολογίες και Διοίκηση Πληροφορικής: Δίκτυα Υπολογιστών και Επικοινωνιών"

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
«Πλατφόρμες ανάπτυξης εφαρμογών Διαδικτύου των Αντικειμένων: Μία κριτική
επισκόπηση»

Αναστασίου Νικόλαος
Α.Μ: 117027

Επιβλέπων:
Χρήστος Γκουμόπουλος

ΣΑΜΟΣ
ΣΕΠΤΕΜΒΡΙΟΣ 2019

«Πλατφόρμες ανάπτυξης εφαρμογών Διαδικτύου των Αντικειμένων: Μία κριτική επισκόπηση»

Η Διπλωματική Εργασία
παρουσιάστηκε ενώπιον
του Διδακτικού Προσωπικού του
Πανεπιστημίου Αιγαίου

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

του

ΝΙΚΟΛΑΟΥ ΑΝΑΣΤΑΣΙΟΥ

ΕΑΡΙΝΟ ΕΞΑΜΗΝΟ 2019

Η ΤΡΙΜΕΛΗΣ ΕΠΙΤΡΟΠΗ ΔΙΔΑΣΚΟΝΤΩΝ ΕΠΙΚΥΡΩΝΕΙ
ΤΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
ΤΟΥ ΝΙΚΟΛΑΟΥ ΑΝΑΣΤΑΣΙΟΥ

[ΟΝΟΜΑ ΔΙΔΑΣΚΟΝΤΑ] , Επιβλέπων
Τμήμα Μηχανικών Πληροφοριακών και
Επικοινωνιακών Συστημάτων

Ημερομηνία

[ΟΝΟΜΑ ΔΙΔΑΣΚΟΝΤΑ], Μέλος
Τμήμα Μηχανικών Πληροφοριακών και
Επικοινωνιακών Συστημάτων

[ΟΝΟΜΑ ΔΙΔΑΣΚΟΝΤΑ], Μέλος
Τμήμα Μηχανικών Πληροφοριακών και
Επικοινωνιακών Συστημάτων

ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ

ΕΑΡΙΝΟ ΕΞΑΜΗΝΟ 2019

ΠΕΡΙΛΗΨΗ

Το *Διαδίκτυο των Αντικειμένων* (*Internet of Things, IoT*) αποτελεί πλέον μια εξέχουσα τεχνολογία παγκοσμίως. Το Διαδίκτυο των αντικειμένων ορίζεται ως ένα σύστημα αλληλένδετων υπολογιστικών συσκευών, ψηφιακών μηχανών, αντικειμένων και ανθρώπων, το οποίο ταυτόχρονα χρησιμοποιεί την υποδομή του Διαδικτύου ως τη ραχοκοκαλιά του συστήματος επικοινωνίας, για να δημιουργήσει μια έξυπνη αλληλεπίδραση μεταξύ των συστατικών του. Ο γενικός στόχος είναι η διασύνδεση του φυσικού με τον ψηφιακό κόσμο. Συνεπώς, ο φυσικός κόσμος μετρίεται από αισθητήρες και μεταφράζεται σε επεξεργάσιμα δεδομένα. Στην παρούσα διπλωματική εργασία πραγματοποιείται μια κριτική επισκόπηση του πεδίου έρευνας που αφορά στις πλατφόρμες ανάπτυξης εφαρμογών Διαδικτύου των Αντικειμένων. Η εργασία προτείνει ένα πλαίσιο αξιολόγησης των διαφορετικών προσεγγίσεων, το οποίο έχει καταγραφεί μέσα από τη μελέτη της τρέχουσας ερευνητικής στάθμης (*state-of-the-art*).

Οι *πλατφόρμες ανάπτυξης εφαρμογών Διαδικτύου των Αντικειμένων* (*IoT πλατφόρμες*) αντιπροσωπεύουν το λογισμικό που επιτρέπει στο Διαδίκτυο των Αντικειμένων να είναι λειτουργικό. Το λογισμικό αυτό χρησιμοποιεί τα δεδομένα που συλλέγονται από τους αισθητήρες για να παράγει χρήσιμη πληροφορία η οποία αναλύεται και χρησιμοποιείται από τις εφαρμογές. Οι κατασκευαστές και οι προμηθευτές των IoT πλατφορμών υποστηρίζουν διαφορετικούς τύπους αρχιτεκτονικών ανάλογα με τις υπηρεσίες που παρέχουν. Για παράδειγμα, μια αρχιτεκτονική μπορεί να υλοποιηθεί με τη μορφή μιας cloud πλατφόρμας μέσω της οποίας υποστηρίζονται η διαχείριση των συνδεδεμένων συσκευών, καθώς και η ανάπτυξη και η χρήση εφαρμογών. Στο πλαίσιο της εργασίας θα γίνει η μελέτη και η ανάλυση δεκατεσσάρων διαφορετικών περιπτώσεων IoT πλατφορμών βάσει της σχετικής βιβλιογραφίας, θα ορισθεί ένα πλαίσιο αξιολόγησης των προσεγγίσεων που θα εξεταστούν και θα πραγματοποιηθεί η καταγραφή των ευρημάτων της αξιολόγησης ως προς τους άξονες που περιλαμβάνει το συγκεκριμένο πλαίσιο.

© 2019

του ΝΙΚΟΛΑΟΥ ΑΝΑΣΤΑΣΙΟΥ

Τμήμα Μηχανικών Πληροφοριακών και Επικοινωνιακών Συστημάτων

ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ

ABSTRACT

The IoT concept has faced prominent criticism around the globe. What does IoT do is basically foresee the general embed of many things while using the Internet as the backbone of the network in order to create a clever interaction with people and devices over the Internet. The main target is to connect the natural with the digital world. In this dissertation a critical overview of the research field is made that refers to the application development platforms of the IoT. This thesis suggests an assessment framework of different approaches that has been recorded through the study of (state-of-the-art).

The application development platforms of the IoT is the software that makes the IoT function. This software uses the data which are gathered by sensors in order to create useful information which is analyzed and used from the applications. The manufactures and the suppliers of the IoT platforms support different types of architecture depend on the services they provide. For instance, an architecture can be implemented with the form of a cloud platform that both supports the control of already connected devices and the development and usage of applications based on them. In this thesis a study and an analysis of fourteen different cases of IoT platforms based on the relevant bibliography will be occurred. Moreover, a framework for assessing approaches to be consider will be created. Furthermore, a recording of the assessment findings will be held towards the evaluation framework

© 2019

NIKOLAOS ANASTASIOU

Department of Information and Communication Systems Engineering

UNIVERSITY OF THE AEGEAN

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΚΕΦΑΛΑΙΟ 1 – ΕΙΣΑΓΩΓΗ	16
1.1 Πλατφόρμες Internet of Things.....	16
1.2 Στόχοι και μεθοδολογία	16
1.3 Συμβολή της διπλωματικής.....	17
1.4 Διάρθρωση της διπλωματικής.....	18
ΚΕΦΑΛΑΙΟ 2 – ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ INTERNET OF THINGS.....	19
2.1 Ορισμός του Internet of Things	20
2.2 Κατηγορίες Αρχιτεκτονικής	20
2.3 Βασικά στοιχεία IoT αρχιτεκτονικής	22
2.4 Δομικά στοιχεία Internet of Things.....	25
2.5 Πρότυπα (standards) Internet of Things.....	27
2.6 Μελλοντικές προκλήσεις Internet of Things	35
ΚΕΦΑΛΑΙΟ 3 – ΠΛΑΤΦΟΡΜΕΣ INTERNET OF THINGS	36
3.1 Ανάπτυξη IoT πλατφορμών	36
3.2 Χαρακτηριστικά IoT πλατφορμών	37
3.2.1 Συνδεσιμότητα	38
3.2.2 Διαχείριση συσκευής	38
3.2.3 Αποθήκευση Δεδομένων	38
3.2.4 Επεξεργασία & Διαχείριση ενεργειών	39
3.2.5 Ανάλυση στοιχείων	39
3.2.6 Απεικόνιση δεδομένων	40
3.2.7 Ασφάλεια	40
3.2.8 Επιπρόσθετα εργαλεία	41
3.2.9 Εξωτερικές διεπαφές.....	41
3.3 Παράδειγμα κατανόησης χαρακτηριστικών	41
3.4 Παράγοντες διαφοροποίησης IoT πλατφορμών	44
3.4.1 Τεχνολογικό υπόβαθρο.....	44
3.4.2 Πεδίο εστίασης τελικού πελάτη	45
3.4.3 Προσέγγιση υλοποίησης	46
3.5 Κριτήρια Επιλογής IoT πλατφόρμας	47
ΚΕΦΑΛΑΙΟ 4 – ΑΠΑΙΤΗΣΕΙΣ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΠΛΑΤΦΟΡΜΩΝ – ΚΡΙΤΗΡΙΑ ΑΞΙΟΛΟΓΗΣΗΣ	50
4.1 Integration - APIs.....	50

4.2 Ασφάλεια	51
4.3 Πρωτόκολλα Επικοινωνίας	51
4.4 Ανάλυση Δεδομένων	51
4.5 Απεικόνιση Δεδομένων	52
4.6 Hosting	53
4.7 Push Notifications	53
4.8 Ενεργοποίηση συμβάντων (Trigger)	53
4.9 Αποθήκευση Δεδομένων	54
4.10 Αποθήκευση BLOB	54
4.11 Kit Ανάπτυξης Λογισμικού (SDKs)	55
4.12 Geolocation	56
4.13 Τιμολόγηση IoT Πλατφορμών	57
4.14 Υποστηριζόμενες γλώσσες προγραμματισμού	57
4.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων	58
4.16 Ενσωμάτωση IoT Hardware πλατφορμών	59
ΚΕΦΑΛΑΙΟ 5 – ΠΑΡΟΥΣΙΑΣΗ IoT ΠΛΑΤΦΟΡΜΩΝ	61
5.1 Kaa Enterprise IoT Platform	61
5.1.1 Integration - APIs	61
5.1.2 Ασφάλεια	62
5.1.3 Πρωτόκολλα Επικοινωνίας	63
5.1.4 Ανάλυση Δεδομένων	63
5.1.5 Απεικόνιση Δεδομένων	63
5.1.6 Hosting	64
5.1.7 Push Notifications	64
5.1.8 Ενεργοποίηση Συμβάντων (Trigger Actions)	65
5.1.9 Αποθήκευση Δεδομένων	66
5.1.10 Αποθήκευση Blob	67
5.1.11 Kit Ανάπτυξης Λογισμικού (SDKs)	67
5.1.12 Geolocation	67
5.1.13 Πλάνα τιμολόγησης	68
5.1.14 Υποστηριζόμενες Γλώσσες Προγραμματισμού	68
5.1.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων	69
5.1.16 Ενσωμάτωση με Hardware πλατφόρμες	69
5.1.17 Συμπεράσματα	69
5.2 IBM Watson IoT platform	69

5.2.1 Integration - APIs.....	70
5.2.2 Ασφάλεια	71
5.2.3 Πρωτόκολλα Επικοινωνίας.....	71
5.2.4 Ανάλυση δεδομένων	71
5.2.5 Απεικόνιση δεδομένων	72
5.2.6 Hosting.....	73
5.2.7 Push Notifications.....	73
5.2.8 Ενεργοποίηση Συμβάντων (Trigger Actions)	74
5.2.9 Αποθήκευση Δεδομένων	74
5.2.10 Αποθήκευση Blob.....	75
5.2.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)	75
5.2.12 Geolocation	76
5.2.13 Πλάνα Τιμολόγησης	76
5.2.14 Υποστηριζόμενες γλώσσες Προγραμματισμού	76
5.2.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων	77
5.2.16 Ενσωμάτωση με Hardware πλατφόρμες	78
5.2.17 Συμπεράσματα	78
5.3 Microsoft Azure IoT platform.....	78
5.3.1 Integration – APIs.....	80
5.3.2 Ασφάλεια	80
5.3.3 Πρωτόκολλα Επικοινωνίας.....	81
5.3.4 Ανάλυση Δεδομένων.....	81
5.3.5 Απεικόνιση Δεδομένων.....	82
5.3.6 Hosting.....	83
5.3.7 Push Notifications.....	83
5.3.8 Ενεργοποίηση συμβάντων (Trigger Actions)	83
5.3.9 Αποθήκευση Δεδομένων	84
5.3.10 Αποθήκευση BLOB	84
5.3.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)	85
5.3.12 Geolocation	86
5.3.13 Πλάνα Τιμολόγησης	86
5.3.14 Υποστηριζόμενες γλώσσες προγραμματισμού	86
5.3.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων	86
5.3.16 Ενσωμάτωση με Hardware πλατφόρμες.....	86
5.3.17 Συμπεράσματα	87

5.4 WSo2 IoT Platform	87
5.4.1 Integration - APIs	89
5.4.2 Ασφάλεια	89
5.4.3 Πρωτόκολλα επικοινωνίας	90
5.4.4 Ανάλυση Δεδομένων	90
5.4.5 Απεικόνιση Δεδομένων	91
5.4.6 Hosting	92
5.4.7 Push Notifications	93
5.4.8 Ενεργοποίηση συμβάντων (Trigger)	93
5.4.9 Αποθήκευση Δεδομένων	93
5.4.10 Αποθήκευση Blob	94
5.4.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)	94
5.4.12 Geolocation	94
5.4.13 Πλάνα Τιμολόγησης	94
5.4.14 Υποστηριζόμενες γλώσσες προγραμματισμού	94
5.4.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων	94
5.4.16 Ενσωμάτωση με Hardware πλατφόρμες	94
5.4.17 Συμπεράσματα	95
5.5 AWS IoT platform	95
5.5.1 Integration - API	97
5.5.2 Ασφάλεια	97
5.5.3 Πρωτόκολλα Επικοινωνίας	98
5.5.4 Ανάλυση Δεδομένων	99
5.5.5 Απεικόνιση Δεδομένων	100
5.5.6 Hosting	100
5.5.7 Push Notifications	100
5.5.8 Ενεργοποίηση συμβάντων (Trigger Actions)	101
5.5.9 Αποθήκευση Δεδομένων	102
5.5.10 Αποθήκευση Blob	102
5.5.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)	102
5.5.12 Geolocation	103
5.5.13 Πλάνα Τιμολόγησης	103
5.5.14 Υποστηριζόμενες γλώσσες προγραμματισμού	103
5.5.14 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων	103
5.5.16 Ενσωμάτωση Hardware πλατφορμών	103

5.5.17 Συμπεράσματα	104
5.6 ThingSpeak IoT platform	105
5.6.1 Integration - API.....	105
5.6.2 Ασφάλεια	106
5.6.3 Πρωτόκολλα Επικοινωνίας	106
5.6.4 Απεικόνιση και Ανάλυση Δεδομένων	106
5.6.6 Hosting.....	107
5.6.7 Push Notifications.....	107
5.6.8 Ενεργοποίηση συμβάντων (Trigger).....	108
5.6.9 Αποθήκευση Δεδομένων	108
5.6.10 Αποθήκευση Blob.....	109
5.6.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)	109
5.6.12 Geolocation	109
5.6.13 Πλάνα Τιμολόγησης	109
5.6.13 Υποστηριζόμενες γλώσσες προγραμματισμού	109
5.6.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων	109
5.6.16 Ενσωμάτωση Hardware Πλατφορμών.....	110
5.6.17 Συμπεράσματα	110
5.7 Google Cloud IoT Platform	111
5.7.1 Cloud IoT Core API's.....	112
5.7.2 Ασφάλεια	113
5.7.3 Πρωτόκολλα Επικοινωνίας	113
5.7.4 Ανάλυση δεδομένων	114
5.7.5 Απεικόνιση δεδομένων	114
5.7.6 Hosting.....	114
5.7.7 Push Notifications.....	114
5.7.8 Ενεργοποίηση Συμβάντων	115
5.7.9 Αποθήκευση Δεδομένων	115
5.7.10 Αποθήκευση Blob.....	115
5.7.10 Κιτ Ανάπτυξης Λογισμικού (SDKs)	116
5.7.12 Geolocation API.....	116
5.7.13 Πλάνα Τιμολόγησης	116
5.7.14 Υποστηριζόμενες γλώσσες προγραμματισμού	116
5.7.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων	116
5.7.16 Ενσωμάτωση με Hardware πλατφόρμες.....	117

5.7.17 Συμπεράσματα	117
5.8 Carriots IoT platform	118
5.8.1 Integration – APIs.....	119
5.8.2 Ασφάλεια	119
5.8.3 Πρωτόκολλα Επικοινωνίας	120
5.8.4 Ανάλυση - Απεικόνιση.....	120
5.8.6 Hosting.....	120
5.8.7 Push Notifications.....	120
5.8.8 Ενεργοποίηση Συμβάντων	120
5.8.9 Αποθήκευση Δεδομένων	120
5.8.10 Αποθήκευση Blob.....	121
5.8.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)	121
5.8.12 Geolocation	121
5.8.13 Πλάνα Τιμολόγησης	121
5.8.14 Υποστηριζόμενες Γλώσσες Προγραμματισμού	121
5.8.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων	121
5.8.16 Ενσωμάτωση Hardware πλατφορμών	122
5.8.17 Συμπεράσματα	122
5.9 SiteWhere IoT platform	123
5.9.1 Integration - APIs.....	124
5.9.2 Ασφάλεια δεδομένων	124
5.9.3 Πρωτόκολλα Επικοινωνίας	125
5.9.4 Ανάλυση Δεδομένων.....	125
5.9.5 Απεικόνιση των δεδομένων	125
5.9.6 Hosting.....	126
5.9.7 Push Notifications.....	126
5.9.8 Ενεργοποίηση Συμβάντων (Trigger Actions)	126
5.9.9 Αποθήκευση Δεδομένων	126
5.9.10 Αποθήκευση Blob.....	127
5.9.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)	127
5.9.12 Geolocation	127
5.9.13 Πλάνα Τιμολόγησης	128
5.9.14 Υποστηριζόμενες Γλώσσες Προγραμματισμού	128
5.9.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων	128
5.9.16 Ενσωμάτωση Hardware Πλατφορμών.....	128

5.9.17 Συμπεράσματα	128
5.10 Thinger.io Platform	129
5.10.1 Integration - APIs	129
5.10.2 Ασφάλεια Δεδομένων	129
5.10.3 Πρωτόκολλα Επικοινωνίας.....	130
5.10.4 Ανάλυση – Απεικόνιση Δεδομένων.....	130
5.10.6 Hosting	130
5.10.7 Push Notifications	131
5.10.8 Ενεργοποίηση Συμβάντων (Trigger Actions)	131
5.10.9 Αποθήκευση Δεδομένων	131
5.10.10 Αποθήκευση Blob.....	131
5.10.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)	132
5.10.12 Geolocation	132
5.10.13 Πλάνα Τιμολόγησης.....	132
5.10.14 Υποστηριζόμενες Γλώσσες Προγραμματισμού.....	132
5.10.15 Ενσωμάτωση με πλατφόρμες τρίτων παρόχων.....	132
5.10.16 Ενσωμάτωση Hardware πλατφορμών	132
5.10.17 Συμπεράσματα	132
5.11 Ubidots Platform	133
5.11.1 Integration – APIs	133
5.11.2 Ασφάλεια.....	134
5.11.3 Πρωτόκολλα Επικοινωνίας.....	135
5.11.4 Απεικόνιση Δεδομένων	135
5.11.5 Ανάλυση Δεδομένων	136
5.11.6 Hosting	137
5.11.7 Push Notifications	137
5.11.8 Ενεργοποίηση Συμβάντων (Trigger Actions)	138
5.11.9 Αποθήκευση Δεδομένων	138
5.10.10 Αποθήκευση Blob.....	139
5.11.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)	139
5.11.12 Geolocation	139
5.11.13 Πλάνα τιμολόγησης	140
5.11.14 Υποστηριζόμενες Γλώσσες Προγραμματισμού.....	140
5.11.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων	140
5.11.16 Ενσωμάτωση Hardware πλατφορμών	141

5.11.17 Συμπεράσματα	141
5.12 Oracle IoT Platform	142
5.12.1 Integration - APIs	142
5.12.2 Ασφάλεια Συσκευών	142
5.12.3 Πρωτόκολλα Επικοινωνίας	142
5.12.4 Ανάλυση Δεδομένων	142
5.12.5 Απεικόνιση Δεδομένων	143
5.12.6 Hosting	143
5.12.7 Push Notifications	143
5.12.8 Ενεργοποίηση Συμβάντων (Trigger)	144
5.12.9 Αποθήκευση Δεδομένων	144
5.12.10 Αποθήκευση Blob	144
5.12.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)	145
5.12.12 Geolocation	145
5.12.13 Πλάνα Τιμολόγησης	145
5.12.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων	145
5.12.16 Ενσωμάτωση Hardware πλατφορμών	145
5.12.17 Συμπεράσματα	146
5.13 ThingWorx Platform	146
5.13.1 Integration - APIs	147
5.13.2 Ασφάλεια	147
5.13.3 Πρωτόκολλα Επικοινωνίας	147
5.13.4 Απεικόνιση Δεδομένων	147
5.13.5 Ανάλυση Δεδομένων	148
5.13.6 Hosting	148
5.13.7 Push Notifications	149
5.13.8 Ενεργοποίηση Συμβάντων (Trigger)	149
5.13.9 Αποθήκευση Δεδομένων	149
5.13.10 Αποθήκευση Blob	150
5.13.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)	150
5.13.12 Geolocation	150
5.13.13 Πλάνα Τιμολόγησης	150
5.13.14 Υποστηριζόμενες Γλώσσες Προγραμματισμού	150
5.13.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων	151
5.13.16 Ενσωμάτωση Hardware πλατφορμών	151

5.13.17 Συμπεράσματα	151
5.14 ThingsBoard Platform.....	151
5.14.1 Integration API	153
5.14.2 Ασφάλεια.....	153
5.13.3 Πρωτόκολλα Επικοινωνίας.....	153
5.14.4 Απεικόνιση Δεδομένων	153
5.14.5 Ανάλυση Δεδομένων	154
5.14.6 Hosting	154
5.14.7 Push Notifications	154
5.14.8 Ενεργοποίηση Συμβάντων (Trigger).....	154
5.14.9 Αποθήκευση Δεδομένων	155
5.14.10 Αποθήκευση Blob.....	155
5.14.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)	155
5.14.12 Geolocation API	155
5.14.13 Πλάνα τιμολόγησης	155
5.14.14 Υποστηριζόμενες Γλώσσες Προγραμματισμού	156
5.14.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων	156
5.14.16 Ενσωμάτωση Hardware πλατφορμών	156
ΚΕΦΑΛΑΙΟ 6 – ΔΟΚΙΜΗ ΠΛΑΤΦΟΡΜΩΝ.....	157
6.1 Υλικό Δοκιμών	157
6.2 Δοκιμή Πλατφόρμας Ubidots	160
6.2.1 Δημιουργία Λογαριασμού.....	160
6.2.2 Προσθήκη Συσκευής	161
6.2.3 Παραμετροποίηση Arduino IDE.....	162
6.2.4 Φόρτωση Κώδικα	163
6.2.5 Αναγνώριση Συσκευής – Απεικόνιση Δεδομένων.....	165
6.2.6 Δημιουργία Πινάκων (Dashboards)	166
6.2.7 Ενεργοποίηση Συμβάντων (Trigger Actions)	170
6.2.8 Χρήση Mobile Εφαρμογής.....	171
6.2.9 Έλεγχος MQTT Επικοινωνίας	172
6.2.10 Συμπεράσματα	174
6.3 Δοκιμή Πλατφόρμας ThingSpeak	175
6.3.1 Διαδικασία Δοκιμής.....	175
6.3.2 Ενσωμάτωσής με MATLAB	180
6.3.3 Χρήση Mobile Εφαρμογής.....	181

6.3.4 Συμπεράσματα.....	181
6.4 Δοκιμή Πλατφόρμας ThingWorx	182
6.4.1 Βήμα 1.....	182
6.4.2 Βήμα 2.....	183
6.4.3 Βήμα 3.....	183
6.4.4 Βήμα 4.....	184
6.4.5 Βήμα 5.....	184
6.4.6 Βήμα 6.....	186
6.4.7 Έλεγχος Δοκιμής	186
ΚΕΦΑΛΑΙΟ 7 - ΑΞΙΟΛΟΓΗΣΗ ΠΛΑΤΦΟΡΜΩΝ.....	189
7.1 Διαδικασία.....	189
7.2 Συζήτηση	195
ΑΝΑΦΟΡΕΣ.....	200

ΚΕΦΑΛΑΙΟ 1 – ΕΙΣΑΓΩΓΗ

1.1 IoT Πλατφόρμες

Η ταχεία εξάπλωση του Διαδικτύου των αντικειμένων (Internet of Things, IoT) καθώς και ο αυξανόμενος αριθμός των συσκευών και των αισθητήρων τα οποία απαρτίζουν το IoT οικοσύστημα θέτουν διάφορα ερωτήματα σχετικά με την αποτελεσματική επεξεργασία του μεγάλου όγκου των δεδομένων που παράγονται από αυτές τις συσκευές. Επίσης, κατά την προσπάθεια ενσωμάτωσης μια πληθώρας διαφορετικών αντικειμένων - πραγμάτων σε ένα παγκόσμιο δίκτυο όπως το Διαδίκτυο (Internet), προκύπτουν ζητήματα σχετικά με την επεκτασιμότητα, την ετερογένεια, τη διαλειτουργικότητα και την τυποποίηση. Ως απάντηση στα ερωτήματα αυτά, τα τελευταία χρόνια έχουν δημιουργηθεί πολλές πλατφόρμες διαχείρισης των δεδομένων που παράγονται από το Διαδίκτυο των αντικειμένων. Γενικότερα, οι πλατφόρμες ανάπτυξης εφαρμογών Διαδικτύου των Αντικειμένων (IoT πλατφόρμες) αντιπροσωπεύουν το λογισμικό που επιτρέπει στο Διαδίκτυο των Αντικειμένων να είναι λειτουργικό. Οι πλατφόρμες αυτές συμβάλλουν στην διαχείριση, επεξεργασία, ανάλυση και απεικόνιση των δεδομένων καθώς και στις μεταβολές αυτών. Επίσης, στοχεύουν στην διασφάλιση της ασφαλούς επικοινωνίας μεταξύ των αντικειμένων, είτε πρόκειται για ανθρώπους, συσκευές αισθητήρων ή υπηρεσίες, και θεωρούνται αναπόσπαστο μέρος οποιασδήποτε IoT αρχιτεκτονικής.

Οι IoT πλατφόρμες μπορούν να διαφοροποιηθούν με διάφορα κριτήρια όπως για παράδειγμα ο τρόπος με τον οποίο αναλύουν τα δεδομένα, τις βάσεις δεδομένων που χρησιμοποιούν για την αποθήκευση των δεδομένων (π.χ. MySQL ή MongoDB), τα πρωτόκολλα επικοινωνίας όπως το MQTT ή το CoAP, την τιμολόγηση των υπηρεσιών, το υποστηριζόμενο υλικό, την δυνατότητα ενεργοποίησης συμβάντων βάσει αλλαγών στα δεδομένα και πολλά άλλα. Ο αριθμός των πλατφορμών που θα μπορούσε να χρησιμοποιηθεί για την υλοποίηση εφαρμογών IoT μπορεί να είναι αρκετά μεγάλος, επομένως μια σύγκριση και επισκόπηση των πλατφορμών οι οποίες διατίθενται στην αγορά μπορεί να αποτελέσει χρήσιμη πηγή πληροφοριών. Σκοπός αυτής της εργασίας είναι να παρουσιάσει μια σύντομη περιγραφή και ανάλυση των χαρακτηριστικών που προσφέρουν οι IoT πλατφόρμες καθώς και η σύγκριση αυτών.

1.2 Στόχοι και μεθοδολογία

Κύριος στόχος της συγκεκριμένης εργασίας είναι ο εντοπισμός και η καταγραφή των βασικών λειτουργιών και των χαρακτηριστικών που θα πρέπει να έχει μία ολοκληρωμένη IoT πλατφόρμα ώστε να μπορεί να

ικανοποιήσει τις απαιτήσεις των χρηστών της. Επιπλέον, γίνεται μία προσπάθεια εντοπισμού νέων δυνατοτήτων καθώς και των ελλείψεων που παρουσιάζονται στις σύγχρονες πλατφόρμες. Μέσω του εντοπισμού των ελλείψεων μπορούν να γίνουν αντιληπτές οι βελτιώσεις τόσο στα εργαλεία όσο και στις υπηρεσίες που προσφέρουν οι πλατφόρμες. Η μεθοδολογία η οποία ακολουθήθηκε βασίζεται αρχικά στην μελέτη και στην ανάλυση δεκατεσσάρων IoT πλατφορμών του, η επιλογή των οποίων βασίστηκε στη σχετική βιβλιογραφία. Ακολούθως της μελέτης και της ανάλυσης, δημιουργήθηκε ένα πλαίσιο αξιολόγησης των διαφορετικών χαρακτηριστικών και λειτουργιών που παρουσιάζουν οι IoT πλατφόρμες. Τέλος, καταγράφηκαν τα ευρήματα της αξιολόγησης ως προς τους άξονες που περιλαμβάνει το πλαίσιο αξιολόγησης καθώς και οι ερευνητικές προκλήσεις του παρόντος και του μέλλοντος στο συγκεκριμένο πεδίο.

1.3 Συμβολή της διπλωματικής

Η παρούσα διατριβή συμβάλει στην καταγραφή των στοιχείων εκείνων τα οποία καθιστούν την ύπαρξη των IoT πλατφορμών, αναπόσπαστο κομμάτι του Διαδικτύου των Αντικειμένων (Internet of Things). Η εκτενής περιγραφή και ανάλυση των δομικών αυτών στοιχείων μπορεί να αποτελέσει κριτήριο για τον τρόπο επιλογής μιας IoT πλατφόρμας. Για την καλύτερη κατανόηση του ρόλου και της χρησιμότητας των IoT πλατφορμών πραγματοποιήθηκε η παρουσίαση δεκατεσσάρων πλατφορμών με τέτοιο τρόπο ώστε να μπορέσει ο αναγνώστης να αντιληφθεί τόσο τα εργαλεία εκείνα και τα χαρακτηριστικά τα οποία διαφοροποιούν τις IoT πλατφόρμες μεταξύ τους όσο και τα κοινά χαρακτηριστικά αυτών. Επίσης, δημιουργήθηκε ένας συγκριτικός πίνακας των πλατφορμών μέσω του οποίου κατηγοριοποιούνται όλες οι υπηρεσίες και οι λειτουργίες αυτών, κάτι το οποίο συμβάλει στην καλύτερη απεικόνιση των διαφορών και των ομοιοτήτων.

Επιπλέον, στο έκτο κεφάλαιο παρουσιάζεται και αναλύεται η προσπάθεια δοκιμής τριών πλατφορμών με τέτοιο τρόπο, ώστε να μπορέσει ο αναγνώστης να προχωρήσει στις δικές του δοκιμές, χρησιμοποιώντας αντίστοιχο εξοπλισμό με σκοπό την κατανόηση των υπηρεσιών που παρέχονται από τις πλατφόρμες. Σαφώς και υπάρχει πληθώρα παρόμοιων οδηγιών στο Διαδίκτυο. Ως αποτέλεσμα όλων των παραπάνω εξάγονται τα συμπεράσματα για τους άξονες που μπορούν και πρέπει να βελτιωθούν καθώς επίσης και για την διαδικασία επιλογής μιας IoT πλατφόρμας βασισμένη στις ανάγκες του χρήστη.

1.4 Διάρθρωση της διπλωματικής

Η διάρθρωση της διπλωματικής είναι η ακόλουθη:

Στο πρώτο κεφάλαιο και την εισαγωγή, πραγματοποιείται μία γενική περιγραφή των IoT πλατφορμών, καταγράφονται κάποια σημαντικά γνωρίσματα σχετικά με τη διατριβή και που αποσκοπεί. Αναφέρουμε τους λόγους στους οποίους βασίστηκε η επιλογή των πλατφορμών και τον ρόλο που διαδραματίζουν στο IoT οικοσύστημα, όπως και τη μεθοδολογία που ακολουθήσαμε.

Στο δεύτερο κεφάλαιο, γίνεται μία αναφορά στις βασικές έννοιες και στα δομικά στοιχεία από τα οποία απαρτίζεται το Διαδίκτυο των αντικειμένων.

Στο τρίτο κεφάλαιο καταγράφουμε τα βασικά χαρακτηριστικά και τα εργαλεία που θα πρέπει να υπάρχουν σε κάθε IoT πλατφόρμα. Επίσης σε αυτό το κεφάλαιο δίνεται και ένα παράδειγμα κατανόησης των βασικών λειτουργιών μιας πλατφόρμας.

Στο τέταρτο κεφάλαιο καταγράφουμε τις υπηρεσίες και τα χαρακτηριστικά που υπάρχουν ή απουσιάζουν για κάθε μια από τις δεκατέσσερις πλατφόρμες που εξετάστηκαν

Στο πέμπτο κεφάλαιο παρουσιάζεται η σε βάθος ανάλυση όλων των λειτουργιών που εμπεριέχονται στις πλατφόρμες βάσει του συγκριτικού πίνακα αξιολόγησης.

Στο έκτο κεφάλαιο, παρουσιάζεται η δοκιμή τριών πλατφορμών, συλλέγοντας τιμές θερμοκρασίας και υγρασίας από τους αισθητήρες που χρησιμοποιήθηκαν και απεικόνιση των δεδομένων στην πλατφόρμα μέσω του Διαδικτύου.

Στο έβδομο κεφάλαιο, γίνεται η αξιολόγηση των πλατφορμών, έπειτα παρουσιάζεται ένα συγκριτικός πίνακας με όλα τα χαρακτηριστικά των αξόνων σε κάθε πλατφόρμα και μετά ακολουθεί μία συζήτηση σχετικά με την εξαγωγή χρήσιμων πληροφοριών για τους άξονες και τα προβλήματα/ελλείψεις που προκύπτουν καθώς και κάποιες μελλοντικές προκλήσεις.

ΚΕΦΑΛΑΙΟ 2 – ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ ΠΕΔΙΟΥ ΙoT

Το IoT αποτελεί μία τεχνολογική επανάσταση κατά την οποία, ηλεκτρονικές συσκευές, άνθρωποι και διάφορα άλλα συστήματα συνδέονται μεταξύ τους με τέτοιο τρόπο, ώστε να μπορούν να επικοινωνούν μέσω του Διαδικτύου. Με τον όρο "πράγματα – αντικείμενα" αναφερόμαστε σε οποιαδήποτε ζωντανή οντότητα όπως οι άνθρωποι και τα ζώα καθώς και σε οποιεσδήποτε μη ζωντανές οντότητες πάνω στη γη. Το διαδίκτυο των αντικειμένων, συνδυάζει τον πραγματικό κόσμο με τον εικονικό κόσμο της ψηφιακής τεχνολογίας [1], βάσει του οποίου επηρεάζεται άμεσα η κοινωνική, προσωπική και επαγγελματική ζωή των ανθρώπων. Υπάρχουν διάφοροι τομείς και περιβάλλοντα στα οποία το IoT μπορεί να διαδραματίσει έναν αξιοσημείωτο ρόλο και να βελτιώσει την ποιότητα της ζωής, όπως η υγειονομική περίθαλψη (healthcare), οι μεταφορές (transportation) καθώς και η αντιμετώπιση έκτακτων αναγκών σε φυσικές καταστροφές (natural disasters).



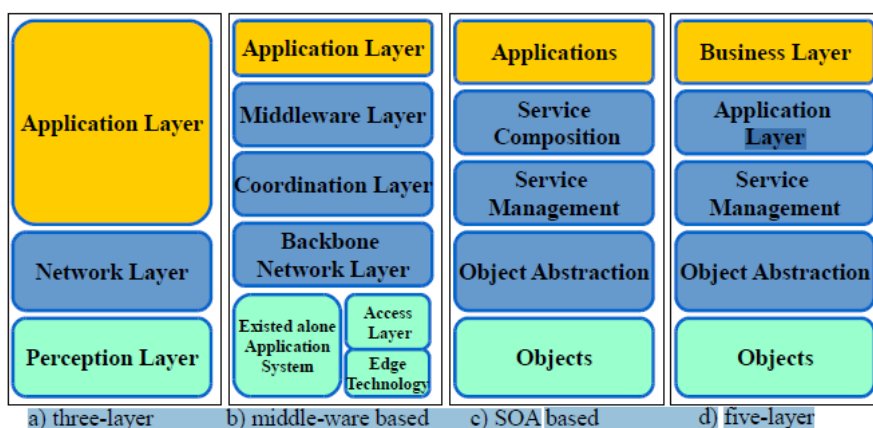
Εικόνα 2-1: Οικοσύστημα Διαδικτύου των Αντικειμένων

2.1 Ορισμός του IoT

Αν θέλαμε να δώσουμε ένα ορισμό για το Διαδίκτυο των αντικειμένων, θα το ορίζαμε ως ένα σύστημα αλληλένδετων υπολογιστικών συσκευών, ψηφιακών μηχανών, αντικειμένων, ζώων ή προσώπων που διαθέτουν μοναδικά αναγνωριστικά στοιχεία (UID) μέσω των οποίων συλλέγονται δεδομένα από τους αισθητήρες που βρίσκονται ενσωματωμένοι πάνω τους, χωρίς την ανθρώπινη παρέμβαση. Ωστόσο, η επιτυχία του Διαδικτύου των αντικειμένων εξαρτάται σε μεγάλο βαθμό, από μια καλά σχεδιασμένη αρχιτεκτονική, με στόχο να φέρει την έννοια του IoT πιο κοντά στην πραγματικότητα.

2.2 Αρχιτεκτονικά Μοντέλα

Το IoT πρέπει να είναι ικανό να διασυνδέει δισεκατομμύρια ή τρισεκατομμύρια ετερογενή αντικείμενα μέσω του Διαδικτύου, και για τον λόγο αυτόν γεννιέται μία κρίσιμη ανάγκη, η οποία στηρίζεται στην δημιουργία μιας ευέλικτης και πολυεπίπεδης αρχιτεκτονικής. Ο συνεχώς αυξανόμενος αριθμός των προτεινόμενων αρχιτεκτονικών δεν έχει ακόμη συγκλίνει σε κάποιο συγκεκριμένο μοντέλο αναφοράς. Από τα προτεινόμενα μοντέλα που υπάρχουν [7], το βασικό μοντέλο αφορά μία αρχιτεκτονική τριών επιπέδων, η οποία περιλαμβάνει το επίπεδο Δικτύου (Network Layer), το επίπεδο εφαρμογής (Application Layer) και το φυσικό επίπεδο (Perception Layer). Στην Εικόνα 2.2 απεικονίζονται ορισμένες κοινές αρχιτεκτονικές, μεταξύ των οποίων και το μοντέλο της αρχιτεκτονικής των πέντε επιπέδων.



Εικόνα 2-2: Κατηγορίες επιπέδων IoT αρχιτεκτονικών

Η αρχιτεκτονική των πέντε επιπέδων αποτελείται από τα εξής στρώματα: Αντικειμένων (Objects), Αφαίρεσης Αντικειμένων (Object Abstraction layer), Διαχείρισης Υπηρεσιών (Service Management Layer), Εφαρμογών (Application Layer) και Επιχειρηματικό (Business). Μια σύντομη περιγραφή για το κάθε ένα ακολουθεί.

1. **Επίπεδο Αντικειμένων (Objects Layer)**

Το πρώτο επίπεδο, το οποίο αναφέρεται ως αυτό των αντικειμένων (Objects), αντιπροσωπεύει τους φυσικούς αισθητήρες του IoT, οι οποίοι αποσκοπούν στην συλλογή (collection) καθώς και στην επεξεργασία της πληροφορίας [7]. Περιλαμβάνονται τόσο οι αισθητήρες (sensors) όσο και οι ενεργοποιητές (actuators), με σκοπό την εκτέλεση διαφορετικών λειτουργιών, όπως η αναζήτηση θέσης (querying location), θερμοκρασίας (temperature), κίνησης (motion), κραδασμών (vibration), υγρασίας (humidity) και άλλων πολλών. Τυποποιημένοι μηχανισμοί plug-and-play χρησιμοποιούνται στο συγκεκριμένο επίπεδο, για την παραμετροποίηση – διαμόρφωση (configuration) ετερογενών αντικειμένων. Επίσης, το επίπεδο των αντικειμένων ψηφιοποιεί και μεταφέρει τα δεδομένα στο επόμενο επίπεδο, το ονομαζόμενο Abstraction Layer μέσω ασφαλών καναλιών (secure channels).

2. **Επίπεδο Object Abstraction**

Στο συγκεκριμένο επίπεδο, έχουμε την μεταφορά των δεδομένων που παράγονται από το προηγούμενο στρώμα, στο αμέσως επόμενο το οποίο είναι το επίπεδο διαχείρισης υπηρεσιών (Service Management Layer). Τα δεδομένα μπορούν να μεταφερθούν μέσω διαφόρων τεχνολογιών όπως RFID, 3G, GSM, UMTS, WiFi, Bluetooth Low Energy, ZigBee και πολλών άλλων.

3. **Επίπεδο Διαχείρισης Υπηρεσιών (Service Management Layer)**

Στο επίπεδο διαχείρισης υπηρεσιών ή αλλιώς Middleware Layer, έχουμε την αντιστοίχιση (pairing), μιας υπηρεσίας με τον αιτούντα (requester), βάσει διευθύνσεων και ονομάτων. Επίσης, παρέχεται η δυνατότητα στους προγραμματιστές εφαρμογών IoT, να συνεργάζονται με ετερογενή αντικείμενα χωρίς να λαμβάνεται υπόψη μια συγκεκριμένη πλατφόρμα υλικού. Επιπλέον, σε αυτό το στρώμα έχουμε την επεξεργασία των δεδομένων τα οποία έχουν συλλεχθεί, την λήψη αποφάσεων καθώς και την παροχή των απαιτούμενων υπηρεσιών.

4. **Επίπεδο Εφαρμογής (Application Layer)**

Στο συγκεκριμένο στρώμα, παραδίδονται οι υπηρεσίες οι οποίες έχουν ζητηθεί από τους πελάτες. Για παράδειγμα, το επίπεδο εφαρμογής απεικονίζει στο πελάτη δεδομένα τα οποία έχουν ζητηθεί, όπως η μέτρηση της θερμοκρασίας και της υγρασίας. Η σημασία αυτού του στρώματος για το IoT είναι ότι έχει την ικανότητα να παρέχει υψηλής ποιότητας έξυπνες υπηρεσίες για να καλύψει τις ανάγκες των πελατών. Επίσης, καλύπτει μια μεγάλη γκάμα υπηρεσιών όπως, το έξυπνο σπίτι (smart home), το έξυπνο κτίριο (smart building), την μεταφορά (transportation), το βιομηχανικό

αυτοματισμό (industrial automation) καθώς και την έξυπνη υγειονομική περίθαλψη (smart healthcare).

5. Επιχειρηματικό Επίπεδο (Business Layer)

Το συγκεκριμένο επίπεδο διαχειρίζεται τις συνολικές δραστηριότητες και υπηρεσίες του IoT συστήματος. Οι ευθύνες αυτού του επιπέδου είναι η δημιουργία ενός επιχειρηματικού μοντέλου (business model), απεικόνισης γραφημάτων και διαγραμμάτων ροής, τα οποία βασίζονται στα ληφθέντα δεδομένα από το επίπεδο εφαρμογής. Επίσης, σχετίζεται με τον σχεδιασμό, την ανάλυση, την εφαρμογή, την αξιολόγηση στοιχείων σχετιζόμενα με το IoT σύστημα. Ακόμη, ένα άλλο σημαντικό στοιχείο σε αυτό το επίπεδο είναι η διαδικασία λήψης αποφάσεων, βασισμένων στην ανάλυση μεγάλου όγκου δεδομένων. Επιπλέον, η παρακολούθηση και διαχείριση των υποκείμενων τεσσάρων στρωμάτων επιτυγχάνεται σε αυτό το στρώμα. Στο επίπεδο αυτό συγκρίνεται η έξοδος (output) κάθε στρώματος με το αναμενόμενο αποτέλεσμα για την ενίσχυση της ποιότητας των υπηρεσιών και τη διατήρηση της μυστικότητας των χρηστών.

2.3 Βασικά συστατικά στοιχεία IoT αρχιτεκτονικής

Σε αυτή την ενότητα περιγράφονται τα βασικά στοιχεία μιας IoT αρχιτεκτονικής. Όλα τα στοιχεία αυτής της αρχιτεκτονικής [16] περιγράφονται με τον τρόπο από κάτω προς τα πάνω όπως ακριβώς απεικονίζονται στην Εικόνα 2.3.

- Αισθητήρας (Sensor)

Ο αισθητήρας ορίζεται ως ένα στοιχείο υλικού, το οποίο καταγράφει πληροφορίες σχετικά με το φυσικό περιβάλλον [1], απαντώντας σε ένα φυσικό ερέθισμα όπως η θερμότητα, το φως, ο ήχος, η πίεση, ο μαγνητισμός ή μια συγκεκριμένη κίνηση. Για παράδειγμα, για την καταγραφή της θερμοκρασίας μέσα σε ένα δωμάτιο, θα πρέπει να τοποθετηθεί ο ανάλογος αισθητήρας. Οι αισθητήρες μεταδίδουν τις καταγεγραμμένες πληροφορίες χρησιμοποιώντας ηλεκτρικά σήματα προς τις συσκευές στις οποίες είναι συνδεδεμένοι. Αυτή η σύνδεση μπορεί να πραγματοποιηθεί με καλώδιο ή ασύρματα.

- Ενεργοποιητής (Actuator)

Ο ενεργοποιητής είναι ένα στοιχείο υλικού, όπου κατά μία έννοια χειρίζεται το φυσικό περιβάλλον. Οι ενεργοποιητές δέχονται εντολές από την συσκευή στην οποία βρίσκονται συνδεδεμένοι και μεταφράζουν αυτά τα ηλεκτρικά σήματα σε κάποιος είδος φυσικής δράσης. Για παράδειγμα, ένας ενεργοποιητής, ανοίγοντας ή κλείνοντας έναν εξαερισμό σε ένα δωμάτιο, ενεργεί στο φυσικό

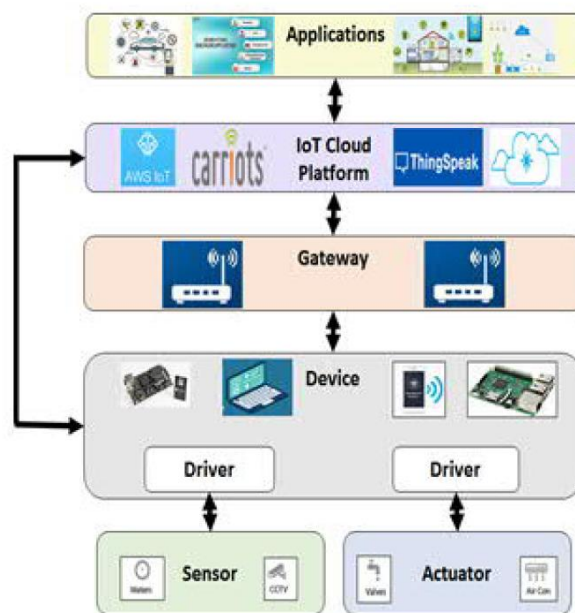
περιβάλλον επηρεάζοντας την υγρασία του δωματίου. Παρόμοια με τους αισθητήρες, η σύνδεση με την συσκευή μπορεί να είναι ενσύρματη ή ασύρματη.

- Συσκευή (Device)

Μία συσκευή είναι ένα στοιχείο υλικού το οποίο συνδέεται ασύρματα ή ενσύρματα με τους αισθητήρες και τους ενεργοποιητές, μπορεί όμως να τις διαθέτει και ενσωματωμένες. Οι συσκευές διαθέτουν επεξεργαστή και αποθηκευτικό χώρο για την εκτέλεση λογισμικού καθώς και την επικοινωνία με το IoT Middleware ή την IoT πλατφόρμα. Αποτελούν το σημείο εισόδου του φυσικού περιβάλλοντος στον ψηφιακό κόσμο. Σε κάθε συσκευή εκτελείται ειδικό λογισμικό (driver), το οποίο επιτρέπει την αλληλεπίδραση με τους αισθητήρες και τους ενεργοποιητές. Οι συσκευές μπορούν να είναι αυτοτελείς (self-contained) ή να συνδέονται με ένα άλλο μεγαλύτερο σύστημα.

- Πύλη (Gateway)

Σε περίπτωση που μία συσκευή δεν είναι σε θέση να συνδεθεί απευθείας με άλλα συστήματα, η επικοινωνία πραγματοποιείται μέσω μίας πύλης (Gateway). Μια πύλη παρέχει τις απαιτούμενες τεχνολογίες και μηχανισμούς ώστε να πραγματοποιηθεί η επικοινωνία μεταξύ διαφορετικών πρωτοκόλλων και τεχνολογιών. Προωθεί την επικοινωνία μεταξύ συσκευών και άλλων συστημάτων. Όταν μια πύλη λάβει ένα μήνυμα σε δυαδική μορφή (binary) από την συσκευή, το μεταφράζει σε μία πιο κοινή μορφή όπως για παράδειγμα την JSON και το προωθεί στο ανάλογο σύστημα μέσω ενός IP δικτύου.



Εικόνα 2-3: Στοιχεία IoT Αρχιτεκτονικής

- *IoT Πλατφόρμα*

Το συγκεκριμένο στοιχείο χρησιμοποιείται ως ένα στρώμα ενσωμάτωσης (integration layer) για τα διάφορα είδη αισθητήρων, ενεργοποιητών, συσκευών και εφαρμογών. Είναι κυρίως υπεύθυνο για την λήψη των δεδομένων από τις συνδεδεμένες συσκευές, την επεξεργασία των λαμβανόμενων δεδομένων, την παροχή των δεδομένων στις συνδεδεμένες εφαρμογές καθώς και τον έλεγχο των συσκευών. Μια συσκευή μπορεί να επικοινωνεί απευθείας με την IoT πλατφόρμα εάν υποστηρίζει μια κατάλληλη τεχνολογία επικοινωνίας όπως η IP επικοινωνία καθώς και ένα αντίστοιχο πρωτόκολλο μεταφοράς, όπως το HTTP, το CoAP ή το MQTT. Διαφορετικά, η συσκευή θα μπορέσει να επικοινωνήσει μέσω μιας πύλης (gateway).

- *Εφαρμογή (Application)*

Το στοιχείο της εφαρμογής αντιπροσωπεύει το λογισμικό που χρησιμοποιεί το IoT Middleware για να αποκτήσει πρόσβαση στον φυσικό κόσμο και να εκτελέσει διάφορες ενέργειες [16]. Αυτό επιτυγχάνεται λαμβάνοντας δεδομένα μέσω των αισθητήρων ή χρησιμοποιώντας τους ενεργοποιητές. Για παράδειγμα, ένα σύστημα λογισμικού μέσω του οποίου ελέγχεται η θερμοκρασία ενός κτιρίου αντιπροσωπεύει μια εφαρμογή συνδεδεμένη σε ένα Middleware.

2.4 Δομικά στοιχεία

Η κατανόηση των δομικών στοιχείων του IoT βοηθά στην απόκτηση μιας καλύτερης εικόνας για την λειτουργία του IoT. Παρακάτω αναλύονται τα έξι βασικά στοιχεία τα οποία απαιτούνται για την λειτουργικότητα του IoT [7], όπως απεικονίζονται στην Εικόνα 2.4.

➤ Αναγνώριση (Identification)

Η αναγνώριση – ταυτοποίηση είναι ένα στοιχείο ζωτικής σημασίας ώστε να πραγματοποιηθεί η αντιστοίχιση ονόματος και υπηρεσίας. Στο IoT διατίθενται πολλές μέθοδοι αναγνώρισης, όπως οι ηλεκτρονικοί κωδικοί προϊόντων (EPC) καθώς και οι ubiquitous κωδικοί (uCode). Επίσης, σημαντική είναι η διαφοροποίηση μεταξύ του αναγνωριστικού (ID) του αντικειμένου και της διεύθυνσης του (IP Address). Το αναγνωριστικό αντικειμένου (ID) αναφέρεται στο όνομα του, ενώ η διεύθυνση του (ip address) στο στοιχείο εκείνο που το διαφοροποιεί εντός ενός δικτύου επικοινωνιών. Οι μέθοδοι αναγνώρισης χρησιμοποιούνται για την παροχή σαφούς ταυτότητας του αντικειμένου εντός του δικτύου.

➤ Ανίχνευση (Sensing)

Η ανίχνευση σχετίζεται με την συλλογή δεδομένων από τα διάφορα αντικείμενα εντός δικτύου, τα οποία αποστέλλονται σε μια βάση δεδομένων. Τα δεδομένα που συλλέγονται αναλύονται για να ληφθούν συγκεκριμένες ενέργειες βάσει των απαιτούμενων υπηρεσιών. Στην ανίχνευση παίρνουν μέρος διάφορες συσκευές όπως οι έξυπνοι αισθητήρες (smart sensors), ενεργοποιητές (actuators) ή φορητές συσκευές ανίχνευσης (wearable devices). Υπολογιστές μονής πλακέτας (Single Board Computers) με ενσωματωμένους αισθητήρες και ενσωματωμένες λειτουργίες TCP / IP χρησιμοποιούνται ως IoT συσκευές.

➤ Επικοινωνία (Communication)

Οι τεχνολογίες επικοινωνίας IoT συνδέουν ετερογενή αντικείμενα ώστε να επιτευχθεί η παράδοση έξυπνων υπηρεσιών. Οι IoT κόμβοι (nodes) θα πρέπει να λειτουργούν με χαμηλή ισχύ (low power). Παραδείγματα πρωτοκόλλων επικοινωνίας που χρησιμοποιούνται για το IoT είναι το, WiFi, Bluetooth, IEEE 802.1, Z-wave και το LTE-Advanced. Ορισμένες τεχνολογίες επικοινωνίας που χρησιμοποιούνται είναι το RFID και το Near Field Communication (NFC).



Εικόνα 2.4: Δομικά Στοιχεία του IoT

➤ **Επεξεργασία – Υπολογισμός (Computation)**

Οι μονάδες επεξεργασίας όπως μικροελεγκτές και μικροεπεξεργαστές, σε συνδυασμό με τις εφαρμογές λογισμικού αντιπροσωπεύουν τον «εγκέφαλο» του IoT. Διάφορες πλατφόρμες υλικού έχουν αναπτυχθεί, πάνω στις οποίες μπορούν να τρέξουν IoT εφαρμογές. Μερικές από αυτές είναι οι : Arduino, UDOO, Friendly ARM, Intel Galileo, Raspberry PI, Gadgeteer, BeagleBone. Επιπλέον, υπάρχουν αρκετές πλατφόρμες λογισμικού οι οποίες χρησιμοποιούνται για να παρέχουν λειτουργίες IoT. Επίσης, πλατφόρμες βασισμένες σε Cloud τεχνολογίες, αποτελούν ένα ακόμη σημαντικό υπολογιστικό μέρος του IoT. Οι συγκεκριμένες πλατφόρμες, επιτρέπουν την επεξεργασία μεγάλου όγκου δεδομένων σε πραγματικό χρόνο.

➤ **Υπηρεσίες (Services)**

Συνολικά, οι υπηρεσίες IoT μπορούν να κατηγοριοποιηθούν σε τέσσερις κατηγορίες:

- **Identity-related services:** Οι συγκεκριμένες υπηρεσίες σχετίζονται με την αναγνώριση – ταυτοποίηση των αντικειμένων, και θεωρούνται από τις πιο βασικές και σημαντικές υπηρεσίες, διότι χρησιμοποιούνται από άλλους τύπους υπηρεσιών.
- **Information Aggregation Services:** Μέσω των υπηρεσιών αυτών επιτυγχάνεται η συλλογή των πληροφοριών μέσω των μετρήσεων όπου έχουν παρθεί από τους διάφορους αισθητήρες.
- **Collaborative – Aware Services:** Βάσει των ληφθέντων δεδομένων, λαμβάνονται διάφορες αποφάσεις και πραγματοποιούνται βάσει αυτών διάφορες ενέργειες.
- **Ubiquitous Services:** Στοχεύουν στην παροχή υπηρεσιών, οπουδήποτε (anywhere) και οποτεδήποτε (anytime)

➤ **Semantics**

Ορίζεται ως η ικανότητα εξαγωγής έξυπνης γνώσης από διαφορετικές μηχανές με στόχο την παροχή των απαιτούμενων υπηρεσιών. Επίσης, περιλαμβάνει την ανακάλυψη και την χρήση πόρων και πληροφοριών για την μοντελοποίηση.

2.5 Πρότυπα (standards)

Πλέον υπάρχουν αρκετά πρότυπα (standards) αναφορικά με το IoT μέσω των οποίων διευκολύνεται και απλοποιείται κατά έναν βαθμό η δουλειά των προγραμματιστών και των παροχών υπηρεσιών (service providers). Έχουν δημιουργηθεί διαφορετικές ομάδες για την παροχή πρωτοκόλλων στο IoT, συμπεριλαμβανομένων των προσπαθειών που διεξάγονται από Παγκόσμιους οργανισμούς όπως το World Wide Web Consortium (W3C), το Internet Engineering Task Force (IETF), το Institute of Electrical and Electronics Engineers (IEEE) καθώς και το European Telecommunications Standards Institute (ETSI). Στον παρακάτω Πίνακα 2.1 απεικονίζεται μια σύνοψη των πιο σημαντικών πρωτοκόλλων. Τα πρωτόκολλα αυτά κατατάσσονται σε τέσσερις κατηγορίες: πρωτόκολλα εφαρμογής (application protocols), πρωτόκολλα ανίχνευσης υπηρεσιών (service discovery protocols), πρωτόκολλα υποδομής (infrastructure protocols) και πρωτόκολλα επιρροής (influential protocols). Δεν απαιτείται να υπάρξει συνδυασμός όλων αυτών των πρωτοκόλλων ώστε να παραδοθεί μια συγκεκριμένη εφαρμογή.

Application Protocol		DDS	CoAP	AMQP	MQTT	MQTT-NS	XMPP	HTTP REST
Service Discovery		mDNS				DNS-SD		
Infrastructure Protocols	Routing Protocol	RPI						
	Network Layer	6LoWPAN					IPv4/IPv6	
	Link Layer	IEEE 802.15.4						
	Physical Layer	LTE-A	EPCglobal		IEEE 802.154		Z-WAVE	
Influential Protocols		IEEE 1888.3, IPSec					IEEE 1905.1	

Πίνακας 2-1: ΠΡΟΤΥΠΑ (STANDARDS) στο IOT

2.5.1 Πρωτόκολλα Εφαρμογής

1) Constrained Application Protocol (CoAP) : Η ομάδα εργασίας του Internet Engineering Task Force (IETF), δημιούργησε το συγκεκριμένο πρωτόκολλο, το οποίο ανήκει στην κατηγορία των πρωτοκόλλων εφαρμογής. Ορίζεται ως ένα πρωτόκολλο μεταφοράς ιστού (web transfer protocol), το οποίο βασίζεται σε REST αρχιτεκτονική. Το REST αντιπροσωπεύει έναν απλούστερο τρόπο ανταλλαγής δεδομένων μεταξύ clients και servers, πάνω από HTTP. Βάσει του σχεδιασμού του το CoAP μπορεί να μεταφραστεί εύκολα σε HTTP με κύριο στόχο την ενοποίηση με το διαδίκτυο [23]. Σε αντίθεση με το REST, το CoAP είναι από προεπιλογή UDP (όχι TCP) το οποίο το καθιστά πιο κατάλληλο για εφαρμογές IoT. Αφορά κυρίως μικροσκοπικές ηλεκτρονικές συσκευές χαμηλής ισχύος και περιορισμένων πόρων οι οποίες θα πρέπει να

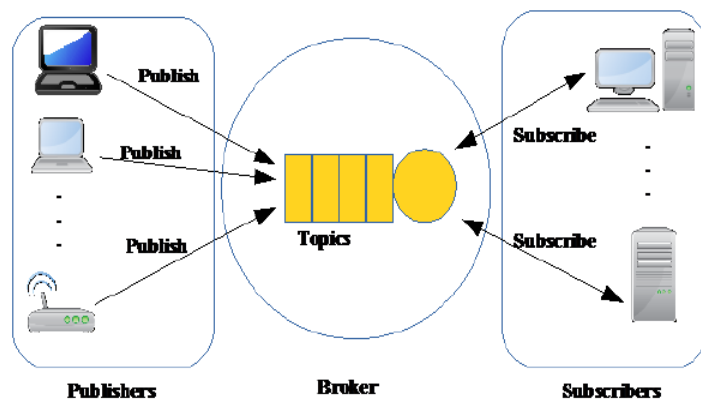
επικοινωνούν διαδικαστικά μέσω του Διαδικτύου. Είναι στοχευμένο για μικρούς αισθητήρες οι οποίοι θα πρέπει να ελέγχονται και να επιτηρούνται από απόσταση.

Το CoAP χρησιμοποιεί τέσσερις τύπους μηνυμάτων: επιβεβαίωση (confirmable), μη επιβεβαίωση (non-confirmable), επαναφορά (reset) και αναγνώριση (acknowledgement). Το CoAP όπως και στο HTTP, χρησιμοποιεί μεθόδους όπως GET, PUT, POST και DELETE για να επιτύχει λειτουργίες όπως η δημιουργία (Create), ανάκτηση (Retrieve), ενημέρωση (UPDATE) και διαγραφή (DELETE). Για παράδειγμα η μέθοδος GET μπορεί να χρησιμοποιηθεί από τον SERVER ώστε να αιτηθεί δεδομένα σχετικά με την θερμοκρασία. Ο client στέλνει πίσω τα δεδομένα, διαφορετικά απαντά με έναν κωδικό κατάστασης (status code) για να δηλώσει ότι δεν έχουν βρεθεί τα ζητούμενα δεδομένα. Το CoAP χρησιμοποιεί μια απλή και μικρή μορφή για την κωδικοποίηση μηνυμάτων. Μερικά από τα σημαντικά χαρακτηριστικά που παρέχει το CoAP περιλαμβάνουν:

- **Παρατήρηση πόρων (Recourse observation):** Συνδρομές κατά παραγγελία για την παρακολούθηση πόρων με τη χρήση μηχανισμού δημοσίευσης / εγγραφής (publish/subscribe).
- **Block-wise resource transport:** Δυνατότητα ανταλλαγής συγκεκριμένων δεδομένων από τον πομποδέκτη μεταξύ client και server, με στόχο την μικρότερη κατανάλωση του εύρους ζώνης της επικοινωνίας.
- **Interacting with HTTP:** Ευελιξία επικοινωνίας με πολλές συσκευές, επειδή η κοινή αρχιτεκτονική με το REST επιτρέπει στο CoAP να αλληλοεπιδρά εύκολα με το HTTP διαμέσου ενός proxy.
- **Security:** Πρόκειται για ένα ασφαλές πρωτόκολλο το οποίο βασίζεται στο DTLS (datagram transport layer security) και εγγυάται την ακεραιότητα και την εμπιστευτικότητα των ανταλλασσόμενων μηνυμάτων.

2) Message Queue Telemetry Transport (MQTT) : Πρόκειται για ένα πρωτόκολλο ανταλλαγής μηνυμάτων το οποίο δημιουργήθηκε από τους Andy Stanford-Clark της IBM και Arlen Nipper της Arcom το 1999 και τυποποιήθηκε το 2013. Δεν πρόκειται για πρωτόκολλο αρχιτεκτονικής client/server όπου ο client αποστέλλει ένα αίτημα στον server και αυτός του απαντά. Όπως απεικονίζεται στην Εικόνα 2.5, υλοποιείται σε λογική publisher/subscriber [7] όπου ο client εκδηλώνει το αίτημα του για διάφορα δεδομένα (subscriber) και ο διακομιστής του απαντά μόνον όταν υπάρχει κάτι διαθέσιμο το οποίο έχει ληφθεί από άλλον client (publisher). Το MQTT αποτελείται από τρεις οντότητες, τον εκδότη (publisher), τον συνδρομητή (subscriber) και μια ενδιάμεση οντότητα τον broker. Μία συσκευή θα εγγραφεί ως συνδρομητής ώστε να λαμβάνει πληροφορίες για συγκεκριμένα θέματα από τον broker (ενδιάμεση οντότητα) όταν οι εκδότες (publishers) δημοσιεύουν θέματα ενδιαφέροντος. Με απλά λόγια, ο εκδότης μεταδίδει τις πληροφορίες στους συνδρομητές μέσω του broker. Επιπλέον, ο broker έχει άποψη όλων των

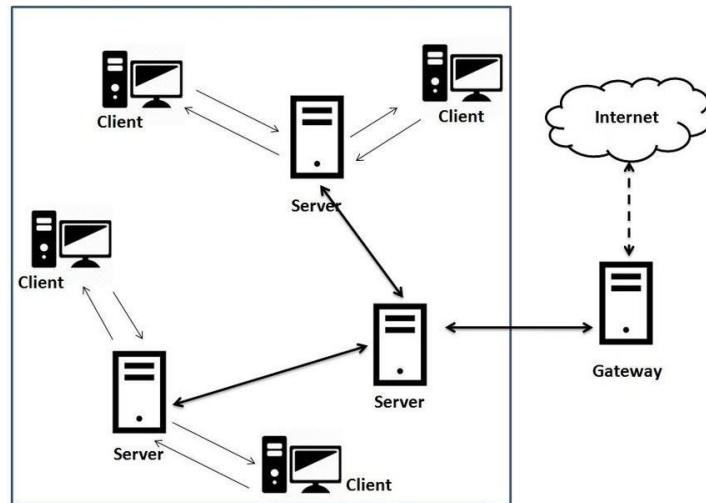
συνδρομητών του συστήματος καθώς επίσης μέσω αυτού επιτυγχάνεται η ασφάλεια ελέγχοντας την εξουσιοδότηση (authorization) των εκδοτών και των συνδρομητών.



Εικόνα 2-5: Αρχιτεκτονική MQTT Πρωτοκόλλου

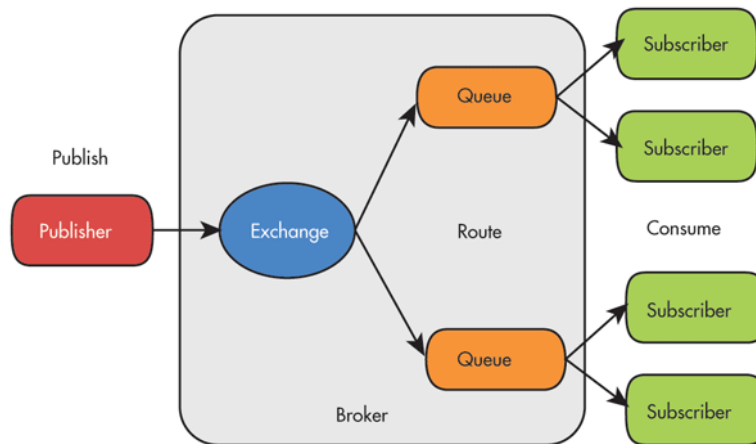
Το MQTT χρησιμοποιείται από πολλών ειδών εφαρμογές που σχετίζονται με την υγειονομική περίθαλψη, την μέτρηση ενέργειας καθώς και τις ειδοποιήσεις που λαμβάνονται μέσω του Facebook. Ως εκ τούτου το MQTT αντιπροσωπεύει ένα ιδανικό πρωτόκολλο ανταλλαγής μηνυμάτων για IoT και M2M επικοινωνίες. Είναι σε θέση να παρέχει δρομολόγηση σε μικρές συσκευές χαμηλής κατανάλωσης και χαμηλής μνήμης οι οποίες λειτουργούν σε δίκτυα χαμηλού εύρους ζώνης.

3) Extensible Messaging and Presence Protocol (XMPP) : Πρόκειται για ένα σύνολο ανοιχτών πρωτοκόλλων άμεσης επικοινωνίας (instant messaging), βασισμένα στην XML. Χρησιμοποιείται για συνομιλία στην οποία λαμβάνουν μέρος ταυτόχρονα πολλοί χρήστες, για φωνητικές κλήσεις, βιντεοκλήσεις και τηλεπικοινωνίες. Επιτρέπει στους χρήστες να επικοινωνούν μεταξύ τους στέλνοντας άμεσα μηνύματα (instant messages), κάνοντας χρήση του Διαδικτύου ανεξάρτητα από το λειτουργικό σύστημα που χρησιμοποιούν. Το XMPP επιτρέπει σε Instant Messaging εφαρμογές να επιτύχουν έλεγχο ταυτότητας (authentication), έλεγχο πρόσβασης (access control), κρυπτογράφηση από άκρο σε άκρο (end to end encryption) καθώς και συμβατότητα με άλλα πρωτόκολλα (compatibility). Ένα μοναδικό χαρακτηριστικό του συστήματος αυτού, το οποίο απεικονίζεται στην Εικόνα 2.6 είναι αυτό των μεταφορών, επίσης γνωστές ως πύλες (gateways), οι οποίες επιτρέπουν στους χρήστες την πρόσβαση σε δίκτυα βασισμένα σε άλλα πρωτόκολλα.



Εικόνα 2-6: Δομή XMPP πρωτοκόλλου

4) Advanced Message Queuing Protocol (AMQP) : Το πρωτόκολλο προηγμένης ουράς μηνυμάτων, είναι ανοικτού προτύπου πρωτόκολλο επιπέδου εφαρμογής (open standard), του οποίου το κύριο χαρακτηριστικό είναι οι ουρές. Στο AMQP απαιτείται ένα αξιόπιστο πρωτόκολλο μεταφοράς όπως το TCP για την ανταλλαγή μηνυμάτων. Στοχεύει στην μη-απώλεια μηνυμάτων. Οι επικοινωνίες διεκπεραιώνονται από δύο βασικά στοιχεία όπως εμφανίζονται στην Εικόνα 2.7: τις ανταλλαγές και τις ουρές.



Εικόνα 2-7: Αρχιτεκτονική Πρωτοκόλλου AMQP

Οι ανταλλαγές χρησιμοποιούνται για την δρομολόγηση των μηνυμάτων στις κατάλληλες ουρές. Η δρομολόγηση μεταξύ ανταλλαγών και ουρών μηνυμάτων βασίζεται σε ορισμένους προκαθορισμένους κανόνες και προϋποθέσεις. Τα μηνύματα μπορούν να αποθηκευτούν στις ουρές μηνυμάτων και στη συνέχεια να αποσταλούν στους παραλήπτες. Εκτός από την επικοινωνία από σημείο σε σημείο (point to point), το AMQP υποστηρίζει επίσης το μοντέλο επικοινωνίας δημοσίευσης / εγγραφής (publish/

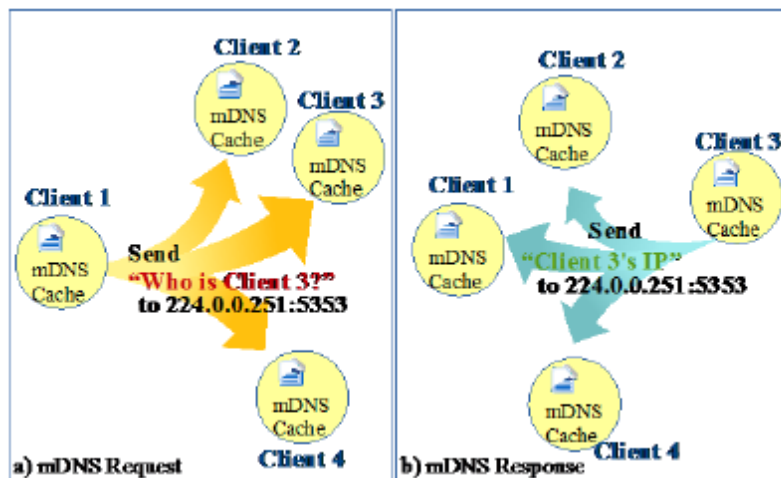
subscribe). Το AMQP ορίζει δύο τύπους μηνυμάτων: Τα λιτά μηνύματα (bare messages) τα οποία παρέχονται από τον αποστολέα και τα σχολιασμένα μηνύματα (annotated messages) τα οποία εμφανίζονται στον δέκτη.

5) Data Distribution Service (DDS) : Η υπηρεσία διανομής δεδομένων είναι ένα πρωτόκολλο δημοσίευσης-εγγραφής (publish/subscribe) για επικοινωνίες M2M, δηλαδή επικοινωνίας συσκευών με άλλες συσκευές σε πραγματικό χρόνο [6]. Σε αντίθεση με άλλα πρωτόκολλα αρχιτεκτονικής publish/subscribe όπως το MQTT ή το AMQP, βασίζεται σε μια αρχιτεκτονική χωρίς μεσάζοντα (broker-less) και χρησιμοποιεί τεχνολογία multicasting για να προσφέρει άριστη ποιότητα εξυπηρέτησης (QoS) και υψηλή αξιοπιστία στις εφαρμογές. Το DDS ορίζει δύο επίπεδα: το Data-Centric Publish-Subscribe (DCPS) και το Data-Local Reconstruction Layer (DLRL). Το πρώτο είναι υπεύθυνο για την παροχή πληροφοριών σε συνδρομητές (subscribers). Από την άλλη πλευρά το DLRL, είναι ένα προαιρετικό στρώμα και χρησιμεύει ως διαπαφή με τις λειτουργίες DCPS.

Η ροή των δεδομένων στο DCPS περιλαμβάνει πέντε οντότητες: (1) Τον εκδότη (publisher) ο οποίος διαδίδει τα δεδομένα, (2) το DataWriter που χρησιμοποιείται από την εφαρμογή για να αλληλοεπιδράσει με τον εκδότη, (3) Συνδρομητής που λαμβάνει δημοσιευμένα δεδομένα και τα παραδίδει στην εφαρμογή, (4) DataReader που χρησιμοποιείται από τον Συνδρομητή για πρόσβαση στα ληφθέντα δεδομένα και (5) Θέμα (Topic) που προσδιορίζεται από έναν τύπο δεδομένων και ένα όνομα. Τα θέματα σχετίζονται με τους DataWriters στους DataReaders. Η μετάδοση δεδομένων επιτρέπεται σε έναν τομέα DDS που είναι ένα εικονικό περιβάλλον για συνδεδεμένες εφαρμογές δημοσίευσης και εγγραφής.

2.5.2 Πρωτόκολλα Ανίχνευσης Υπηρεσιών

1) Multicast DNS (mDNS) : Μία βασική υπηρεσία για κάποιες εφαρμογές IoT όπως το chatting είναι η επίλυση ονομάτων. Το mDNS είναι μια τέτοια υπηρεσία που μπορεί να επιτελέσει τον ρόλο του διακομιστή DNS [7]. Το mDNS είναι ευέλικτο λόγω του ότι ο χώρος ονομάτων του DNS χρησιμοποιείται τοπικά χωρίς την ανάγκη για επιπλέον διαμόρφωση (configuration). Το mDNS είναι η κατάλληλη επιλογή για ενσωματωμένες συσκευές που βασίζονται στο Internet λόγω των γεγονότων ότι: α) δεν υπάρχει ανάγκη για επιπλέον παραμετροποίηση για την διαχείριση των συσκευών και β) είναι σε θέση να συνεχίσει να λειτουργεί εάν συμβεί κάποια αποτυχία στην υποδομή. Το mDNS ρωτά τα ονόματα αποστέλλοντας ένα μήνυμα IP multicast σε όλους τους κόμβους του τοπικού τομέα. Με αυτό το ερώτημα, ζητείται από συσκευές που έχουν το όνομα να απαντήσουν ξανά. Όταν το μηχανήμα-στόχος λαμβάνει το όνομά του, απαντά με ένα multicast μήνυμα που περιέχει την IP διεύθυνσή του, όπως απεικονίζεται στην Εικόνα 2-8. Όλες οι συσκευές στο δίκτυο που λαμβάνουν αυτό το μήνυμα ενημερώνουν την τοπική τους μνήμη cache χρησιμοποιώντας το όνομα και τη διεύθυνση IP.



Εικόνα 2-8: Πρωτόκολλο mDNS

2) DNS service discovery (DNS-SD) : Χρησιμοποιώντας αυτό το πρωτόκολλο, οι clients έχουν την δυνατότητα να ανακαλύψουν ένα σύνολο επιθυμητών υπηρεσιών σε ένα συγκεκριμένο δίκτυο κάνοντας χρήση τυπικών DNS μηνυμάτων. Ομοίως με το mDNS, δεν απαιτείται επιπλέον παραμετροποίηση για την σύνδεση των συσκευών.

Ουσιαστικά, το DNS-SD χρησιμοποιεί το mDNS για την αποστολή πακέτων DNS σε συγκεκριμένες διευθύνσεις πολυεκπομπής (multicast) μέσω του UDP. Υπάρχουν δύο βασικά βήματα για την διαδικασία του Service Discovery: (1) εύρεση ονομάτων κεντρικών υπολογιστών των απαιτούμενων υπηρεσιών, όπως εκτυπωτές και αντιστοίχιση των IP διευθύνσεων με τα ονόματα των κεντρικών υπολογιστών τους χρησιμοποιώντας το mDNS. Η εύρεση ονομάτων είναι σημαντική επειδή οι διευθύνσεις IP ενδέχεται να αλλάξουν, ενώ τα ονόματα όχι. Χρησιμοποιώντας το DNS-SD, τα ονόματα των συσκευών στο δίκτυο μπορούν να διατηρούνται σταθερά όσο το δυνατόν περισσότερο ώστε να διατηρείται η εμπιστοσύνη και η αξιοπιστία.

Το Διαδίκτυο των Αντικειμένων χρειάζεται κάποιο είδος αρχιτεκτονικής χωρίς να εξαρτάται από μηχανισμούς διαμόρφωσης - παραμετροποίησης. Σε μια τέτοια αρχιτεκτονική, έξυπνες συσκευές μπορούν να ενταχθούν στην πλατφόρμα ή να την αφήσουν χωρίς να επηρεάσουν τη συμπεριφορά ολόκληρου του συστήματος. Το mDNS και το DNS-SD μπορούν να βοηθήσουν σε αυτόν τον τρόπο ανάπτυξης.

2.5.3 Πρωτόκολλα Υποδομής

1) 6Low PAN : Τα Ασύρματα δίκτυα προσωπικής περιοχής χαμηλής κατανάλωσης (Low power Wireless Personal Area Networks) στα οποία βασίζονται πολλές από τις επικοινωνίες του IoT, έχουν κάποια ειδικά χαρακτηριστικά, όπως περιορισμένο μέγεθος πακέτου (π.χ. μέγιστο 127 bytes), διάφορα μήκη

διευθύνσεων καθώς και περιορισμένο εύρος ζώνης. Έτσι, υπήρξε ανάγκη να δημιουργηθεί ένα στρώμα προσαρμογής που να ταιριάζει με τα πακέτα IPv6 σύμφωνα με τις προδιαγραφές του IEEE 802.15.4. Η ομάδα εργασίας IETF 6LoWPAN ανέπτυξε ένα τέτοιο πρότυπο το 2007. Το 6LoWPAN είναι η προδιαγραφή των υπηρεσιών που απαιτείται από το IPv6 μέσω ασυρμάτων δικτύων χαμηλής ισχύος για τη διατήρηση ενός δικτύου IPv6. Το συγκεκριμένο πρότυπο παρέχει συμπίεση κεφαλίδας (header compression) ώστε να μειωθεί το φορτίο κατά την μετάδοση καθώς και κατακερματισμό (fragmentation) με στόχο την κάλυψη των απαιτήσεων των τιμών MTU στο IPv6. Υπάρχει η δυνατότητα συμπίεσης της IPv6 κεφαλίδας σε 2 bytes.

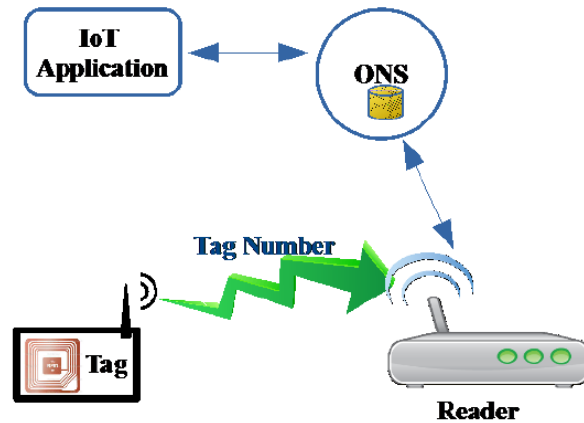
2) Bluetooth Low Energy: Το συγκεκριμένο πρωτόκολλο ή αλλιώς και Bluetooth Smart, χρησιμοποιεί μικρή εμβέλεια με χαμηλή ισχύ, ώστε να μπορεί να λειτουργήσει για μεγαλύτερο χρονικό διάστημα (ακόμη και για χρόνια) σε σύγκριση με τις προηγούμενες εκδόσεις του. Η κάλυψη της εμβέλειας (περίπου 100 μέτρα) είναι δέκα φορές μεγαλύτερη του κλασικού Bluetooth, ενώ η καθυστέρηση είναι 15 φορές μικρότερη. Το BLE μπορεί να λειτουργήσει με ισχύ μετάδοσης μεταξύ 0,01 mW έως 10 mW. Με αυτά τα χαρακτηριστικά, το BLE είναι ένας καλός υποψήφιος για εφαρμογές IoT. Το πρότυπο BLE αναπτύχθηκε γρήγορα από τους κατασκευαστές smartphone και είναι τώρα διαθέσιμο στα περισσότερα μοντέλα. Η σκοπιμότητα της χρήσης αυτού του προτύπου αφορά κυρίως στην επικοινωνία οχήματος με όχημα (vehicle-to-vehicle) καθώς και στα ασύρματα δίκτυα αισθητήρων. Σε σύγκριση με το ZigBee, το BLE είναι πιο αποδοτικό όσον αφορά την κατανάλωση ενέργειας.

Το BLE επιτρέπει στις συσκευές να λειτουργούν ως masters ή slaves. Ο μηχανισμός ανακάλυψης λειτουργεί μέσω της αποστολής μηνυμάτων από τους slaves σε συγκεκριμένα κανάλια τα οποία σαρώνονται από τους masters. Εκτός από την ώρα που οι δύο συσκευές ανταλλάσσουν δεδομένα, για το υπόλοιπο της ώρας βρίσκονται σε λειτουργία αναμονής.

3) EPC global : Ο Ηλεκτρονικός Κωδικός Προϊόντος (EPC) είναι ένας μοναδικός αριθμός ταυτοποίησης ο οποίος αποθηκεύεται σε ετικέτα RFID και χρησιμοποιείται κυρίως για τον εντοπισμό αντικειμένων. Ο EPC global είναι ο οργανισμός ο οποίος είναι υπεύθυνος για την ανάπτυξη του EPC καθώς και για την διαχείριση των τεχνολογιών EPC και RFID. Η βασική αρχιτεκτονική χρησιμοποιεί τεχνολογίες RFID με βάση το Διαδίκτυο μαζί με φτηνές ετικέτες RFID και αναγνώστες για την ανταλλαγή των πληροφοριών.

Αυτή η αρχιτεκτονική αναγνωρίζεται ως μια πολλά υποσχόμενη τεχνική για το μέλλον του IoT λόγω του ανοίγματος (openness), της επεκτασιμότητας του (scalability), της διαλειτουργικότητας (interoperability) και της αξιοπιστίας (reliability) πέρα από την υποστήριξή του στις πρωταρχικές απαιτήσεις του IoT, όπως στα αναγνωριστικά αντικειμένων (objects IDs) και την ανακάλυψη υπηρεσιών (service discovery). Το σύστημα RFID μπορεί να χωριστεί σε δύο κύρια στοιχεία: (1) τον αναμεταδότη ραδιοκυμάτων (tag) και

τον αναγνώστη ετικετών. Η ετικέτα από την άλλη πλευρά αποτελείται από δύο στοιχεία: ένα τσιπ για την αποθήκευση της ταυτότητας του αντικειμένου και μια κεραία για να επιτρέψει στο τσιπ να επικοινωνεί με τον αναγνώστη ετικετών χρησιμοποιώντας ραδιοκύματα. Ο αναγνώστης ετικετών δημιουργεί ένα πεδίο ραδιοσυχνότητας για τον εντοπισμό αντικειμένων μέσω ανακλώμενων ραδιοκυμάτων της ετικέτας. Η λειτουργία RFID στέλνει τον αριθμό της ετικέτας στον αναγνώστη ετικετών χρησιμοποιώντας ραδιοκύματα, όπως φαίνεται στην Εικόνα 2.9.



Εικόνα 2-9: Ηλεκτρονικός Κωδικός Προϊόντος (EPC)

Μετά από αυτό, ο αναγνώστης διαβιβάζει αυτόν τον αριθμό σε μια συγκεκριμένη εφαρμογή υπολογιστή που ονομάζεται Object-Naming Services (ONS). Ένα ONS αναζητά τα στοιχεία της ετικέτας από μια βάση δεδομένων, όπως το πότε και πού κατασκευάστηκε.

4) Z-Wave: Πρόκειται για ένα πρωτόκολλο ασύρματης επικοινωνίας καθώς και χαμηλής κατανάλωσης το οποίο χρησιμοποιείται κυρίως από εφαρμογές αυτοματισμού σε έξυπνα σπίτια. Το Z-Wave έχει πεδίο κάλυψης περίπου 30 μέτρων από σημείο σε σημείο. Το Z-Wave λειτουργεί σε ζώνες ISM(περίπου 900 MHz) και επιτρέπει ταχύτητα μετάδοσης 40 kbps. Οι πρόσφατες εκδόσεις υποστηρίζουν έως και 200 kbps.

ΚΕΦΑΛΑΙΟ 3 – ΙoT ΠΛΑΤΦΟΡΜΕΣ

Η ταχεία εξάπλωση του IoT σε συνδυασμό με τον αυξανόμενο αριθμό συσκευών και συστατικών - εξαρτημάτων από τα οποία αποτελείται ένα ασύρματο δίκτυο αισθητήρων (Wireless sensor networks) εγείρουν ερωτήματα για τον τρόπο μέσω του οποίου θα υπάρξει αποτελεσματική επεξεργασία του τόσο μεγάλου όγκου δεδομένων που παράγονται από αυτά τα δίκτυα. Τα τελευταία χρόνια έχουν δημιουργηθεί πολλαπλές IoT πλατφόρμες. Οι πλατφόρμες αυτές συμβάλλουν στη διαχείριση, στην επεξεργασία και στην απεικόνιση των δεδομένων που ανταλλάσσονται σε IoT περιβάλλον καθώς και στην εξαγωγή σημαντικών πληροφοριών. Οι IoT πλατφόρμες μπορούν να διαφέρουν ως προς τον τρόπο που αποθηκεύονται τα δεδομένα, στον τρόπο που επεξεργάζονται τα δεδομένα, τα πρωτόκολλα επικοινωνίας που χρησιμοποιούν, το υλικό (hardware) που υποστηρίζεται, την αδειοδότηση που παρέχεται, την τιμολόγηση των υπηρεσιών και πολλά άλλα. Επίσης οι IoT πλατφόρμες μπορούν να ορισθούν και ως το ενδιάμεσο λογισμικό (middleware) καθώς και η υποδομή η οποία επιτρέπει στους τελικούς χρήστες να αλληλοεπιδρούν με τα έξυπνα αντικείμενα.

3.1 Διάταξη IoT πλατφορμών

Μία IoT πλατφόρμα μπορεί να εγκατασταθεί μέσω μιας cloud υποδομής ή μπορεί να εγκατασταθεί τοπικά σε έναν υπολογιστή ή διακομιστή βάσει της επιλογής του χρήστη. Οι IoT πλατφόρμες οι οποίες βασίζονται σε cloud αρχιτεκτονική (cloud based) παρέχονται συνήθως με την μορφή πλατφόρμας ως υπηρεσίας (PaaS) ή λογισμικού ως υπηρεσίας (SaaS) [3]. Οι PaaS πλατφόρμες παρέχουν υπηρεσίες υποδομής και cloud λειτουργίες στις IoT συσκευές. Αυτές οι υπηρεσίες περιλαμβάνουν την αποθήκευση δεδομένων, τα πρωτόκολλα επικοινωνίας και ασφάλειας, την υποστήριξη κ.λπ. Αυτό σημαίνει ότι στο περιβάλλον με το οποίο αλληλοεπιδρά ο χρήστης για την κατασκευή των εφαρμογών του, δεν χρειάζεται από τον ίδιο να φροντίσει για την διαμόρφωση και την συντήρηση της υποδομής. Αναφορικά με τις Software as a Service (SaaS) πλατφόρμες, πρόκειται για εφαρμογές οι οποίες φιλοξενούνται σε κάποιον κεντρικό υπολογιστή και εστιάζουν περισσότερο στην παροχή εργαλείων για την κατασκευή εφαρμογών, οι οποίες επικεντρώνονται στον τελικό χρήστη με την παροχή του τελικού προϊόντος το οποίο είναι έτοιμο προς χρήση. Στην περίπτωση που η πλατφόρμα διαμορφώνεται σε μια cloud υποδομή, ο χρήστης δεν ασχολείται ούτε με τον τρόπο της αποθήκευσης των δεδομένων ούτε με τους υπολογιστικούς πόρους. Τα πλεονεκτήματα των τοπικά διαμορφωμένων πλατφορμών, σε ορισμένες περιπτώσεις μπορούν να εμφανισθούν και ως μειονεκτήματα, όπως η δυνατότητα προσαρμογής των ρυθμίσεων, η επιλογή ενός συστήματος αποθήκευσης δεδομένων, ο απόλυτος έλεγχος της ανάπτυξης των εφαρμογών καθώς και η ασφάλεια των δεδομένων.

3.2 Χαρακτηριστικά IoT πλατφορμών

Οι IoT πλατφόρμες αποτελούν το κεντρικό κομμάτι μιας IoT αρχιτεκτονικής, βάσει του οποίου συνδέεται ο εικονικός και ο πραγματικός κόσμος καθώς και η επικοινωνία μεταξύ των αντικειμένων (objects) και των χρηστών. Στην πιο απλή της μορφή, μια IoT πλατφόρμα αφορά απλώς την δυνατότητα σύνδεσης μεταξύ των αντικειμένων και συλλογής δεδομένων. Σε μια πιο εξελιγμένη μορφή, μια πλατφόρμα αποτελείται από μια ποικιλία σημαντικών δομικών στοιχείων όπως απεικονίζονται και στον Πίνακα 3-1 και είναι τα εξής [29] : συνδεσιμότητα (connectivity), διαχείριση συσκευής (device management), βάση δεδομένων (database), επεξεργασία & διαχείριση ενεργειών (processing & action management), ανάλυση δεδομένων (analytics), απεικόνιση (visualization), εξωτερικές διεπαφές (external interfaces), πρόσθετα εργαλεία (additional tools) και ασφάλεια (security). Πέραν αυτών των σημαντικών δομικών στοιχείων, σε επόμενα κεφάλαια στα οποία θα γίνει ανάλυση και σύγκριση συγκεκριμένων IoT πλατφορμών, θα εξετάσουμε και άλλα χαρακτηριστικά.

Μέρος αποθήκευσης του συνόλου των δεδομένων Βάση Δεδομένων	Εξωτερικές διεπαφές APIs, SDKs και gateways τα οποία δρουν ως διεπαφές για εξωτερικά συστήματα (ERP, CRM)		Πρόστασία της επικοινωνίας μεταξύ των αντικειμένων και των δεδομένων των χρηστών Ασφάλεια Επικοινωνίας
	Ανάλυση δεδομένων Αλγόριθμοι για προηγμένους υπολογισμούς	Πρόσθετα εργαλεία Περαιτέρω εργαλεία ανάπτυξης (π.χ. εργαλεία αναφοράς)	
	Απεικόνιση Δεδομένων Γραφική απεικόνιση δεδομένων αισθητήρων (σε πραγματικό χρόνο)		
	Επεξεργασία & διαχείριση ενεργειών Κανόνες που επιτρέπουν δράσεις (σε πραγματικό χρόνο) με βάση τα εισερχόμενα δεδομένα αισθητήρων και συσκευών		
	Διαχείριση Συσκευής Εργαλείο Backend για τη διαχείριση της κατάστασης της συσκευής, την απομακρυσμένη ανάπτυξη λογισμικού και τις ενημερώσεις		
	Επικοινωνία Στοιχεία που εξασφαλίζουν σταθερή συνδεσιμότητα αντικειμένων και κατάλληλες μορφές δεδομένων		

Πίνακας 3-1: Χαρακτηριστικά IoT Πλατφορμών

3.2.1 Συνδεσιμότητα

Κάθε IoT πλατφόρμα περιέχει ένα στρώμα συνδεσιμότητας. Αφορά τη λειτουργία κατά την οποία διαφορετικά πρωτόκολλα και διαφορετικές μορφές δεδομένων μπορούν να συμπεριληφθούν σε μια διασύνδεση "λογισμικού". Αυτό είναι απαραίτητο για να διασφαλιστεί ότι όλες οι συσκευές μπορούν να αλληλοεπιδρούν μεταξύ τους και ότι τα δεδομένα διαβάζονται σωστά. Έχοντας όλα τα δεδομένα που

συλλέγονται συγκεντρωμένα σε ένα μέρος και σε μία μορφή δεδομένων, υπάρχει η δυνατότητα για την παρακολούθηση, την διαχείριση και την ανάλυση των IoT συσκευών. Για παράδειγμα θα πρέπει να δημιουργηθούν βιβλιοθήκες για μεμονωμένες συσκευές οι οποίες στέλνουν αναλογικά σήματα όπως ένας αισθητήρας πίεσης ώστε να μπορούν να ενσωματωθούν στην πλατφόρμα, καθώς επίσης και δεδομένα τα οποία αποστέλλονται από άλλων ειδών συσκευές όπως ένα smartphone ή άλλων ειδών φορητές συσκευές.

Οι προηγμένες συσκευές συνήθως παρέχουν ένα API το οποίο αποτελεί τη διεπαφή επικοινωνίας με την πλατφόρμα.

3.2.2 Διαχείριση συσκευής

Η μονάδα (module) διαχείρισης συσκευών μιας IoT πλατφόρμας εξασφαλίζει ότι τα συνδεδεμένα αντικείμενα λειτουργούν και επικοινωνούν σωστά καθώς και ότι το λογισμικό και οι εφαρμογές τους ενημερώνονται και εκτελούνται χωρίς να υπάρχει κάποια δυσλειτουργία [29]. Οι εργασίες που εκτελούνται σε αυτήν την ενότητα περιλαμβάνουν την προετοιμασία των συσκευών (device provisioning), την απομακρυσμένη παραμετροποίηση, τη διαχείριση και τις ενημερώσεις του υλικολογισμικού / λογισμικού καθώς και την αντιμετώπιση των προβλημάτων. Καθώς χιλιάδες ή και εκατομμύρια διαφορετικές συσκευές λαμβάνουν μέρος σε μια IoT υλοποίηση, μαζικές ενέργειες θα πρέπει να γίνονται με αυτοματοποιημένο τρόπο με στόχο την μείωση του κόστους

3.2.3 Αποθήκευση Δεδομένων

Η αποθήκευση των δεδομένων αποτελεί κεντρικό κομμάτι σε μια IoT πλατφόρμα. Η διαχείριση των δεδομένων που παράγονται από τις IoT συσκευές φέρνει νέες απαιτήσεις σε κατηγορίες όπως:

- **Ποσότητα**, τον όγκο των δεδομένων που μπορούν να αποθηκευτούν.
- **Ποικιλία**, διαφορετικές συσκευές και διαφορετικοί τύποι αισθητήρων παράγουν πολύ διαφορετικές μορφές δεδομένων.
- **Ταχύτητα**, πολλές περιπτώσεις εφαρμογών IoT απαιτούν την ανάλυση δεδομένων ροής για λήψη άμεσων αποφάσεων.
- **Ακρίβεια**, σε ορισμένες περιπτώσεις, οι αισθητήρες παράγουν ανακριβή δεδομένα.

Πλεονεκτήματα μπορεί να επιφέρει η δυνατότητα για την αποθήκευση των δεδομένων βασισμένη σε cloud αρχιτεκτονική. Θα πρέπει να είναι κλιμακωτή για μεγάλα δεδομένα και επίσης να είναι σε θέση να αποθηκεύει δομημένα (SQL) και μη δομημένα δεδομένα (NoSQL). Ορισμένες πλατφόρμες προσφέρουν επίσης την άμεση εισαγωγή και εξαγωγή δεδομένων από κόμβους IoT σε μορφές Extensible Markup Language (XML) ή σε μορφότυπο CSV.

3.2.4 Επεξεργασία & Διαχείριση ενεργειών

Η έννοια του IoT δεν συνίσταται μόνο στη συλλογή δεδομένων από τους διάφορους κόμβους και αντικείμενα, αλλά σχετίζεται και με την επεξεργασία τους καθώς και εκτέλεση ενεργειών σε ορισμένες ανιχνεύσιμες αλλαγές. Από την στιγμή που τα δεδομένα έχουν συλλεχθεί και αποθηκευτεί στην βάση δεδομένων, υπάρχει η δυνατότητα για την εκτέλεση ενεργειών οι οποίες σχετίζονται με αυτά τα δεδομένα. Μια ενεργοποίηση συμβάντος με βάση έναν κανόνα (rule-based) επιτρέπει την απόδοση "έξυπνων" ενεργειών που βασίζονται σε συγκεκριμένα δεδομένα τα οποία συλλέγονται μέσω των IoT αισθητήρων.

Οι συνθήκες για την κλήση των ενεργειών μπορεί να ποικίλουν από πολύ απλές, όπως ο καθορισμός μιας τιμής όριο για ένα σύνολο δεδομένων, καθώς και σε πλήρως προσαρμοσμένες που επιτρέπουν σε ένα χρήστη να διαμορφώσει την συνθήκη βάσει κάποιας γλώσσας προγραμματισμού (SQL, MATLAB, JavaScript) [19]. Οι ενέργειες ενδέχεται να περιλαμβάνουν την αποστολή αιτήματος HTTP POST σε έναν καθορισμένο URL, αλλάζοντας την εσωτερική κατάσταση της συνδεδεμένης συσκευής. Ένας άλλος τρόπος ενεργοποίησης και εκτέλεσης μιας ενέργειας είναι η ενσωμάτωση εξωτερικών υπηρεσιών, όπως το ηλεκτρονικό ταχυδρομείο, ή υπηρεσία σύντομων μηνυμάτων (SMS), ή τα κοινωνικά δίκτυα. Αυτές οι υπηρεσίες μπορούν επίσης να χρησιμοποιηθούν για προγραμματισμένες ειδοποιήσεις εάν βέβαια υποστηρίζεται η λειτουργία του χρονοπρογραμματισμού (scheduling).

Σε ένα έξυπνο σπίτι, για παράδειγμα, μπορεί να οριστεί μια ενεργοποίηση συμβάντος-δράσης, έτσι ώστε όλα τα φώτα να σβήνουν όταν ένα άτομο βγαίνει από το σπίτι. Η τεχνική στην οποία βασίζεται η πραγματοποίηση της ενέργειας αυτής έρχεται συχνά με τη μορφή ενός μοντέλου IF-This-Then-That rule (IFTTT). Για παράδειγμα, αν το σήμα GPS υποδεικνύει ότι το smartphone του ατόμου είναι περισσότερο από 5 μέτρα μακριά από το σπίτι του, τότε απενεργοποιούνται όλα τα φώτα.

3.2.5 Ανάλυση δεδομένων

Σε πολλές περιπτώσεις απαιτείται πολύπλοκη και λεπτομερής ανάλυση των δεδομένων τα οποία συλλέγονται, ώστε να αξιοποιηθεί στο έπακρο η ροή των IoT δεδομένων. Σε ένα έξυπνο σπίτι, για παράδειγμα, μέσω της ανάλυσης μπορούν να παραχθούν αλγόριθμοι που επιτρέπουν στην IoT πλατφόρμα να μάθει ποιος συνδυασμός φωτισμού και θέρμανσης προτιμάται από τον χρήστη σε ποια ώρα της ημέρας και σε σχέση με τις εξωτερικές καιρικές συνθήκες.

Οι αναλύσεις οι οποίες βασίζονται σε cloud αρχιτεκτονική των αποθηκευμένων δεδομένων ή των δεδομένων σε πραγματικό χρόνο είναι ένα άλλο παράδειγμα προηγμένων χαρακτηριστικών. Η ανάλυση μπορεί να γίνει με την ενσωμάτωση γλωσσών προγραμματισμού ή γλωσσών σεναρίων (scripting or computing languages). Με την ανάλυση των δεδομένων, ο χρήστης μπορεί να λάβει χρήσιμες πληροφορίες

σχετικά με την κατάσταση των IoT συσκευών και των διαφόρων μετρήσεων και βάσει αυτών να επικαλεστεί αντίστοιχες ενέργειες.

3.2.6 Απεικόνιση δεδομένων

Ένα από τα βασικά χαρακτηριστικά των πλατφορμών είναι και η απεικόνιση των δεδομένων. Τα δεδομένα μπορούν να παρουσιασθούν στον τελικό χρήστη με διαφόρους τρόπους, όπως για παράδειγμα μέσω γραφημάτων, στατικών ή δυναμικών. Διαφορετικοί τύποι γραφημάτων μπορούν να προβληθούν σε όλες τις πλατφόρμες, από απλά γραφήματα γραμμών ή ράβδων όπως επίσης και σε πιο περίπλοκες 3D απεικονίσεις [19]. Μέσω συσκευών οι οποίες αποστέλλουν πληροφορίες σχετικά με την τοποθεσία τους, η γραφική απεικόνιση μπορεί να πάει ένα βήμα πιο πέρα σε συνδυασμό με πλατφόρμες που υποστηρίζουν γεωγραφική κατανομή (geolocation), απεικονίζοντας κάθε συσκευή βάσει της τοποθεσίας της σε πραγματικό χρόνο. Τα γραφήματα και οι πίνακες μπορούν να τοποθετηθούν σε dashboards τα οποία έχει δημιουργήσει ο χρήστης, παρέχοντας μια πιο προσαρμόσιμη προβολή των δεδομένων. Ο πίνακας εργαλείων απεικόνισης (dashboard) ο οποίος διατίθεται στον διαχειριστή της IoT πλατφόρμας συχνά περιλαμβάνεται στα εργαλεία που παρέχει μια προηγμένη IoT πλατφόρμα. Οι πλατφόρμες μπορούν να παρέχουν εργαλεία και επιλογές για πιο προσωποποιημένες απεικονίσεις, δημιουργώντας plugins και widgets χρησιμοποιώντας HTML, CSS, JavaScript ή άλλα ιδιόκτητα εργαλεία.

3.2.7 Ασφάλεια

Σε ένα IoT περιβάλλον, τα διάφορα αντικείμενα τα οποία απαρτίζουν αυτό το περιβάλλον είναι ευάλωτα σε μεγάλο αριθμό επιθέσεων και επομένως είναι κρίσιμο να διασφαλιστεί η ασφάλειά τους σε διαφορετικά επίπεδα [29]. Επίσης, η προστασία των δεδομένων που ανταλλάσσονται μεταξύ των χρηστών αποτελεί ένα σημαντικό στοιχείο.

Διάφορες πλατφόρμες επιτρέπουν στους χρήστες να καθορίζουν τα δικαιώματα πρόσβασης στα δεδομένα. Ο έλεγχος ταυτότητας των συσκευών γίνεται κατά κύριο λόγο με την παροχή κλειδιών APIs για κάθε συσκευή. Αυτά τα κλειδιά, πρέπει να παρουσιαστούν ως μέρος κάθε κλήσης API, προκειμένου να ανακτήσουν ή να αποθηκεύσουν δεδομένα στην πλατφόρμα. Επίσης, ένας ασφαλής τρόπος επαλήθευσης της ταυτότητας μπορεί να επιτευχθεί με την χρήση πιστοποιητικών X.509 (AWS IoT), ή με ιδιόκτητες μεθόδους ελέγχου ταυτότητας.

Η επικοινωνία προς και από τις πλατφόρμες μπορεί να εξασφαλιστεί μέσω του πρωτοκόλλου HTTPS. Ορισμένες πλατφόρμες (π.χ. Azure, AWS IoT) απαιτούν την κρυπτογράφηση της κυκλοφορίας μεταξύ των τελικών σημείων της πλατφόρμας μέσω του πρωτοκόλλου TLS (Transport Layer Security). Η κρυπτογράφηση της επικοινωνίας προς και από τις IoT πλατφόρμες μπορεί να είναι δύσκολη λόγω της ποικιλομορφίας των αντικειμένων στο IoT περιβάλλον.

Πολλά κρυπτοσυστήματα και πρωτόκολλα ασφαλείας μπορεί να θεωρούνται ασφαλή και ισχυρά, αλλά μπορεί να μην είναι κατάλληλα για συσκευές με περιορισμένη υπολογιστική ισχύ και χωρητικότητα.

3.2.8 Επιπρόσθετα εργαλεία

Οι προηγμένες IoT πλατφόρμες προσφέρουν συχνά ένα πρόσθετο σύνολο εργαλείων για τον προγραμματιστή και τον διαχειριστή της IoT εφαρμογής. Τα εργαλεία ανάπτυξης επιτρέπουν στον προγραμματιστή IoT εφαρμογών να κάνει διάφορες δοκιμές. Ένα παράδειγμα είναι ένα εργαλείο διαχείρισης το οποίο καθορίζει ποιος έχει πρόσβαση σε ποια συσκευή και σε ποια δεδομένα. Ένα άλλο εργαλείο σχετίζεται με την αναφορά (reporting), που επιτρέπει την εξαγωγή δεδομένων (π.χ. σε μορφή .csv ή Json).

3.2.9 Εξωτερικές διεπαφές

Σε επιχειρήσεις που έχουν ενεργοποιήσει το Διαδίκτυο των αντικειμένων είναι σημαντικό να καταφέρουν να το ενσωματώσουν με τα υπάρχοντα συστήματα ERP, τα εργαλεία διαχείρισης και το υπόλοιπο ευρύτερο περιβάλλον πληροφορικής που διαθέτουν. Πακέτα ανάπτυξης λογισμικού (SDKs) καθώς και τα APIs αποτελούν το κλειδί για την ενσωμάτωση συστημάτων και εφαρμογών τρίτων κατασκευαστών.

3.3 Παράδειγμα κατανόησης χαρακτηριστικών

Στην συνέχεια παρουσιάζεται ένα παράδειγμα βάσει των παραπάνω στοιχείων, στο οποίο αναλύονται τα πλεονεκτήματα που παρέχει το Διαδίκτυο των αντικειμένων στον χρήστη ενός πλυντηρίου ρούχων [32].

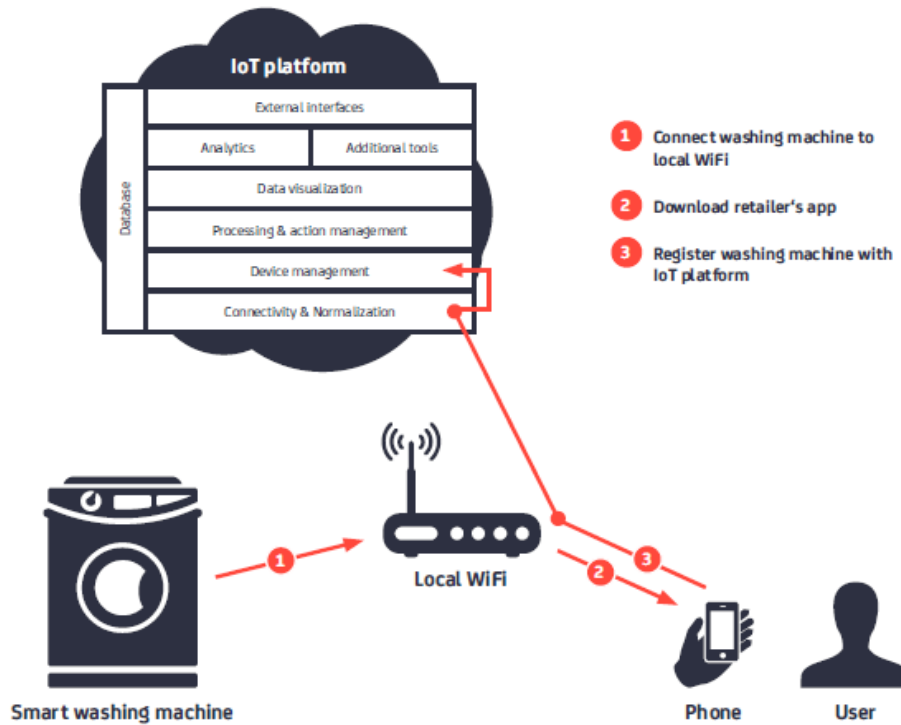
Τα πλεονεκτήματα είναι τα εξής:

- Η συσκευή σταματάει αυτόματα πριν από κάποια βλάβη η οποία θα μπορούσε να προκαλέσει ζημιά στην ροή της λειτουργίας. Εδώ υπάρχει το στοιχείο της προβλέψιμης συντήρησης.
- Ο χρήστης ενημερώνεται για τα προβλήματα και έχει τον έλεγχο του πλυντηρίου του μέσω μιας εφαρμογής smartphone, ανεξάρτητα από την τοποθεσία του
- Μπορεί να αποσταλεί ένας ειδικός εξυπηρέτησης πελατών ή ένας τεχνικός. Με βάση σημαντικά στοιχεία, όπως ο τύπος πλυντηρίου, τα δεδομένα απόδοσης, το ιστορικό και προηγούμενα προβλήματα, μπορεί να ληφθεί άμεσα μια απόφαση.

Στο συγκεκριμένο παράδειγμα έχουμε τέσσερα κύρια στοιχεία όπως απεικονίζονται και στην Εικόνα 3.1:

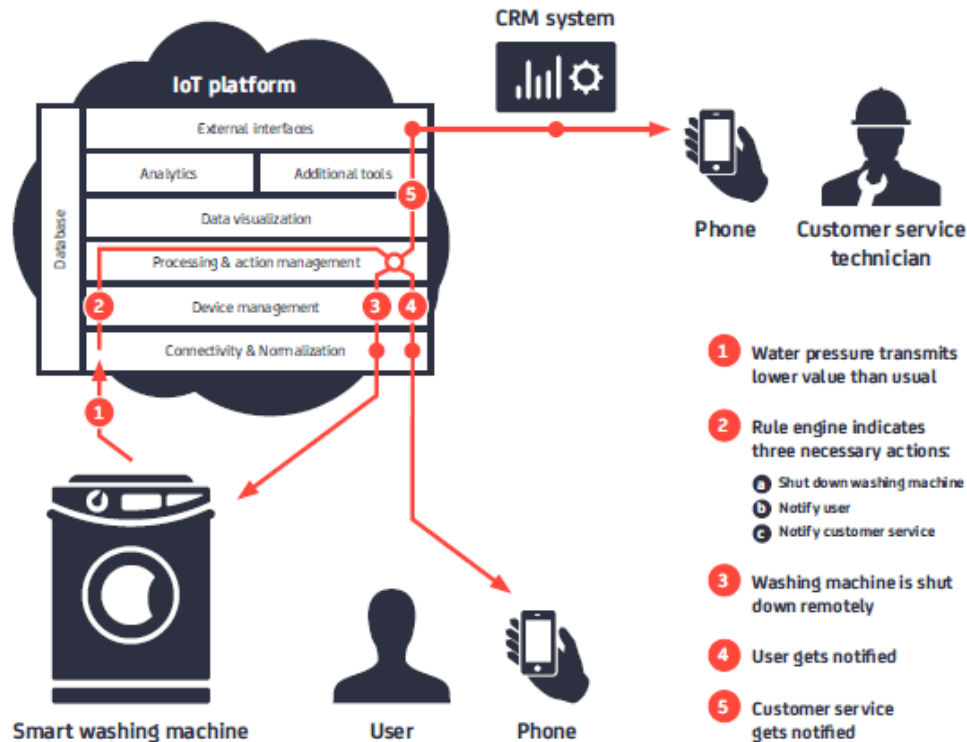
1. *Σύνδεση του πλυντηρίου*: Με την ενεργοποίηση της συσκευής και μέσω ασύρματης πρόσβασης (WiFi) η οποία παρέχεται από το πλυντήριο, μέσω ενός API, η επικοινωνία πλυντηρίου και της IoT πλατφόρμας είναι πλέον εφικτή. Από το σημείο αυτό και μετά ο χρήστης είναι έτοιμος να εκμεταλλευτεί όλες τις δυνατότητες που του παρέχονται. Το μόνο που έχει να κάνει πλέον ο

χρήστης είναι να συνδέσει το πλυντήριο στο τοπικό WiFi και να κατεβάσει την εφαρμογή smartphone που παρέχεται. Στη συνέχεια ο χρήστης συνδέει το πλυντήριο με την IoT πλατφόρμα καταγράφοντας το μηχάνημα στο σύστημα διαχείρισης της πλατφόρμας. Το πλυντήριο στέλνει τώρα συνεχή μηνύματα σε πραγματικό χρόνο μέσω του WiFi του σπιτιού στην IoT πλατφόρμα.



Εικόνα 3-1: Επικοινωνία χρήστη και συσκευής μέσω της πλατφόρμας

2. *Πρόληψη διαρροής νερού:* Το πλυντήριο είναι εξοπλισμένο με αισθητήρες που αναλύουν την κατανάλωση ενέργειας, την πίεση του νερού, τη δόνηση του μηχανήματος και άλλα. Με βάση την λήψη μη κανονικών τιμών στην ροή δεδομένων σε πραγματικό χρόνο μέσω των αισθητήρων, η μηχανή σταματά αυτόματα πριν προκαλέσει οποιαδήποτε ζημιά. Ένα τέτοιο σενάριο θα ήταν μια περίπτωση διαρροής νερού. Η Εικόνα 3.2 δείχνει πώς η IoT πλατφόρμα σταματά γρήγορα το πλυντήριο, αποτρέποντας έτσι πιθανή διαρροή νερού. Τόσο ο χρήστης όσο και ο τεχνικός εξυπηρέτησης πελατών λαμβάνουν ταυτόχρονα ειδοποίηση για να αποφασίσουν πώς να αντιμετωπίσουν αυτό το πρόβλημα.



Εικόνα 3-2: Ειδοποίηση χρήστη για τυχόν πρόβλημα

Στην συνέχεια μέσω της πλατφόρμας ενεργοποιούνται ειδικοί μηχανισμοί βασισμένοι σε κανόνες και εκτελούνται ενέργειες όπως στην περίπτωση πτώσης της πίεσης του νερού, μέσω της IoT πλατφόρμας, απενεργοποιείται η συσκευή και ενημερώνονται μέσω μηνυμάτων, τόσο ο χρήστης όσο και το τμήμα εξυπηρέτηση πελατών της εταιρείας κατασκευής. Σημαντικό στοιχείο αποτελεί η λήψη δεδομένων σε πραγματικό χρόνο.

3. *Υπηρεσία πελατών:* Είτε πρόκειται για μια περίπτωση διαρροής νερού είτε για ηλεκτρονική βλάβη, το πλυντήριο ρούχων μέσω της IoT σύνδεσης βοηθά τον εκπρόσωπο εξυπηρέτησης πελατών να έχει μια πολύ βαθύτερη κατανόηση του πραγματικού προβλήματος, ακόμη και αν είναι μακριά.

3.4 Παράγοντες διαφοροποίησης IoT πλατφορμών

Εκτιμάται ότι υπάρχουν περισσότερες από τριακόσιες IoT πλατφόρμες μέχρι σήμερα και ο αριθμός τους αυξάνεται σχεδόν κάθε εβδομάδα. Υπάρχουν τρεις κύριοι παράγοντες για τη διαφοροποίηση των IoT

πλατφορμών: Το τεχνολογικό υπόβαθρο (technological depth), ο τομέας εστίασης (segment-focus) και η προσέγγιση προσαρμογής (customization approach).

3.4.1 Τεχνολογικό υπόβαθρο

Η ανάπτυξη μιας καλά ολοκληρωμένης IoT πλατφόρμας με όλα τα παραπάνω χαρακτηριστικά και η ολοκλήρωση με μια σειρά προτύπων διαδικτύου απαιτεί αρκετά χρόνια εργασίας. Επομένως υπάρχουν διαφορετικά επίπεδα τεχνολογικού υποβάθρου για τις IoT πλατφόρμες [32]:

- **Επίπεδο 1**

Connectivity platform. Οι πιο απλές IoT πλατφόρμες λειτουργούν ως συλλέκτες δεδομένων και παρέχουν έναν απλό δίαυλο ανταλλαγής μηνυμάτων.

- **Επίπεδο 2**

Action platform. Οι πλατφόρμες σε αυτό το επίπεδο όχι μόνο διαχειρίζονται τη σύνδεση αλλά και επιτρέπουν την ενεργοποίηση ενεργειών που βασίζονται σε συγκεκριμένα συμβάντα. Μέσω αυτών των πλατφορμών για παράδειγμα, μπορεί να ενεργοποιηθεί ο φωτισμός σε συγκεκριμένο χώρο όταν ο ειδικός αισθητήρας ανιχνεύσει κίνηση.

- **Επίπεδο 3**

Full-scale platform. Οι πιο προηγμένες πλατφόρμες υπερβαίνουν τη συνδεσιμότητα και τη δράση, παρέχοντας περισσότερες λειτουργίες, επιτρέποντας εξωτερικές διεπαφές και υποστηρίζοντας μια μεγάλη ποικιλία πρωτοκόλλων και προτύπων. Αυτές οι πλατφόρμες έρχονται επίσης με προηγμένες λύσεις βάσεων δεδομένων που επιτρέπουν την επεκτασιμότητα σε πολλές συσκευές καθώς και ανάλυση μεγάλου όγκου δεδομένων.

3.4.2 Πεδίο εστίασης τελικού πελάτη

Σε αντίθεση με τον χρήστη της συσκευής του πλυντηρίου που αναλύθηκε παραπάνω, υπάρχουν διαφορετικές απαιτήσεις ως προς την επιλογή μιας IoT πλατφόρμας από μία εταιρεία που επιδιώκει την διαχείριση δεδομένων από αισθητήρες οι οποίοι βρίσκονται εγκαταστημένοι σε βιομηχανικά μηχανήματα. Οι μεγάλες διαφορές περιλαμβάνουν διαφορετική υποστήριξη συσκευών και πρωτοκόλλων, διαφορετικό είδος απαιτούμενης ανάλυσης και απεικόνισης, διαφορετικές εξωτερικές διεπαφές, καθώς και διαφορετικό είδος υποδομής ασφάλειας. Ως εκ τούτου, ορισμένες εταιρείες σχεδιασμού και κατασκευής IoT πλατφορμών επικεντρώνονται σε συγκεκριμένους τομείς τελικών χρηστών όπως το Smart Home ή Smart City, ενώ λιγότερες είναι αυτές που εστιάζουν ταυτόχρονα τόσο σε τομείς B2C (Business to consumer) όσο και σε βιομηχανικούς.

- Smart Home: Οι πλατφόρμες Smart Home υποστηρίζουν πρότυπα οικιακής συνδεσιμότητας όπως WiFi, Zigbee, Z-wave και Bluetooth. Συχνά έρχονται με προ-κατασκευασμένες οπτικές εφαρμογές και είναι σχεδιασμένες για την παρακολούθηση και τον έλεγχο συσκευών στο σπίτι.
- Connected Car: Οι πλατφόρμες αυτοκινήτων λειτουργούν με τα παραδοσιακά πρότυπα για το αυτοκίνητο καθώς και το πρωτόκολλο επικοινωνίας οχήματος-οχήματος (V2V) επόμενης γενιάς. Προσφέρουν την ενσωμάτωση της ψυχαγωγίας και δίνουν ιδιαίτερη προσοχή στον τομέα της ασφάλειας καθώς η οποιαδήποτε παρέμβαση από μη εξουσιοδοτημένο χρήστη, αυτή η πλατφόρμα μπορεί γρήγορα να γίνει θανάσιμη για τον οδηγό του οχήματος. Οι πλατφόρμες ενσωματώνουν επίσης υπηρεσίες τηλεματικής όπως η διαχείριση στόλου.
- Smart Retail: Οι πλατφόρμες λιανικής πώλησης που διαχειρίζονται το χαρτοφυλάκιο προϊόντων ενός εμπόρου λιανικής πώλησης τυπικά πρέπει να υποστηρίζουν μια πολύ μεγάλη ποικιλία διαφορετικών συσκευών. Η ενσωμάτωση με τις υπηρεσίες της επιχείρησης (π.χ. λογισμικό CRM) είναι επίσης ένα σημαντικό κριτήριο των συγκεκριμένων πλατφορμών.
- Smart City: Οι περιπτώσεις IoT πλατφορμών στην χρήση έξυπνων πόλεων, όπως η έξυπνη στάθμευση ή η διαχείριση αποβλήτων, συχνά βασίζονται σε δίκτυα χαμηλής κατανάλωσης όπως mesh δίκτυα ή δίκτυα ευρείας ζώνης χαμηλής ισχύος (LPWAN). Οι πλατφόρμες βελτιστοποιούνται επίσης για να συνεργάζονται με υπηρεσίες χαρτογράφησης (π.χ. Google maps) και τοπικές οδικές πληροφορίες.
- Industrial: Οι βιομηχανικές IoT πλατφόρμες παρέχουν ειδικές πύλες (gateways) για να ενσωματωθούν σε υπάρχοντα συστήματα SCADA και αυτοματισμού. Η ενισχυμένη ασφάλεια έχει επίσης μεγάλη σημασία καθώς οι εταιρείες φοβούνται να αποκαλύψουν ακούσια ευαίσθητα δεδομένα σε πελάτες ή ανταγωνιστές.
- Other: Άλλες πλατφόρμες είναι ειδικά σχεδιασμένες για τομείς όπως η έξυπνη γεωργία ή υγεία.

3.4.3 Προσέγγιση προσαρμογής

Ο τρίτος σημαντικός παράγοντας βάσει του οποίου μπορούν να διαφοροποιηθούν οι IoT πλατφόρμες είναι ο τρόπος ενσωμάτωσής τους με τις ανάγκες και το περιβάλλον των συστημάτων που διέπουν μια επιχείρηση ή έναν οργανισμό. Υπάρχουν πλατφόρμες οι οποίες διαθέτουν ένα καθορισμένο σύνολο χαρακτηριστικών τα οποία είναι προ-εγκατεστημένα χωρίς τη δυνατότητα για περαιτέρω παραμετροποίηση, οι οποίες χρησιμοποιούνται με συγκεκριμένα πρότυπα και APIs.

Από την άλλη μεριά υπάρχουν διαθέσιμες στην αγορά IoT πλατφόρμες, οι οποίες είναι αρκετά προσαρμόσιμες με τις μεταβαλλόμενες ανάγκες μιας επιχείρησης ή ενός οργανισμού. Είναι αρκετά

σημαντικό μια εταιρεία η οποία κατασκευάζει μια IoT πλατφόρμα να διαθέτει τους απαραίτητους μηχανισμούς ώστε να μπορέσει να ενσωματώσει τις νέες τεχνολογικές απαιτήσεις μιας εταιρείας στις δυνατότητες της πλατφόρμας.

3.5 Κριτήρια Επιλογής IoT πλατφόρμας

Οι εταιρείες οι οποίες αποφασίζουν να ενσωματώσουν μια IoT πλατφόρμα στις υπηρεσίες τους, έρχονται αντιμέτωπες με το στοιχείο της επιλογής της κατάλληλης πλατφόρμας βάσει των αναγκών τους. Παρακάτω αναλύονται βασικά κριτήρια ως προς την αξιολόγηση μιας IoT πλατφόρμας [31].

Επεκτασιμότητα (Scalability)

Όταν εξετάζουμε μια IoT πλατφόρμα η οποία δεν βασίζεται σε τοπικούς διακομιστές (server less), αλλά υποστηρίζεται από μια cloud υποδομή, εξετάζεται αν υπάρχει η δυνατότητα για αυτόματη κλιμάκωση και εξυπηρέτηση ενός αυξανόμενου φορτίου, χωρίς να υπάρχει απαίτηση ενεργειών από πλευράς του τελικού χρήστη. Ορισμένα IoT πλαίσια (frameworks) παρέχουν πιο εξελιγμένες και λεπτομερέστερες επιλογές κλιμάκωσης, οι οποίες είναι πιο αποτελεσματικές αλλά απαιτούν εξειδικευμένη ομάδα ή προηγμένες λειτουργίες.

Αξιοπιστία (Reliability)

Σε ένα παραγωγικό περιβάλλον τις περισσότερες φορές, δεν υπάρχει ο απαιτούμενος χρόνος ώστε να διερευνηθούν τα αίτια που προκάλεσαν μια πλατφόρμα να μην μπορεί να ανταποκριθεί. Το σύστημα θα πρέπει να διαθέτει δυνατότητες αποκατάστασης ώστε να υποστηρίζονται τα υψηλά επίπεδα αξιοπιστίας της παραγωγής. Κατά την επιλογή μιας IoT πλατφόρμας, θα πρέπει να δίνεται προσοχή στο πώς μπορεί να επιτευχθεί η αξιοπιστία της. Συνήθως είναι ένας συνδυασμός αρχιτεκτονικής και άλλων λειτουργικών παραγόντων.

Προσαρμογή (Customization)

Μια προσαρμοσμένη λύση IoT πλατφόρμας θα πρέπει να προσφέρει ενσωματωμένη λειτουργικότητα, εκτεταμένες βιβλιοθήκες, APIs και επιλογές ενσωμάτωσης με άλλες πλατφόρμες ή υπηρεσίες. Μέσω open-source λύσεων παρέχεται η ελευθερία απεριόριστης προσαρμογής (customization) και ρύθμισης (tuning) του προϊόντος ώστε να ανταποκρίνεται σε συγκεκριμένες ανάγκες.

Λειτουργίες (Operations)

Η διαχείριση μιας IoT πλατφόρμας αποτελεί έναν σημαντικό παράγοντα. Σε αντίθεση με τις IoT πλατφόρμες οι οποίες βασίζονται σε cloud υποδομή, αν όχι σχεδόν όλες, οι περισσότερες λειτουργίες δεν είναι εμφανείς στον τελικό χρήστη. Στην περίπτωση μιας προσαρμοσμένης (custom) πλατφόρμας απαιτείται περισσότερη προσπάθεια από την ομάδα που την διαχειρίζεται. Σημαντικό πλεονέκτημα μπορεί να αποτελέσει ένας πίνακας ελέγχου (dashboard) στον οποίο εμφανίζονται τα γενικά στατιστικά στοιχεία

του συστήματος και πληροφορίες για την ομαλή λειτουργία των υπηρεσιών [33], την κατάσταση κάθε συγκεκριμένης ενότητας (module), με την επιλογή για ειδοποιήσεις σε περίπτωση έκτακτης ανάγκης.

Πρωτοκόλλα Επικοινωνίας (Protocols)

Αξίζει να εξεταστεί εκ των προτέρων ποια πρωτόκολλα υποστηρίζονται από την IoT πλατφόρμα η οποία έχει επιλεγεί. Θα πρέπει επίσης να υποστηρίζονται πρωτόκολλα όπως το MQTT, το πιο ευρέως χρησιμοποιούμενο πρωτόκολλο το οποίο αποτελεί πρότυπο στην IoT τεχνολογία. Το συγκεκριμένο πρωτόκολλο υιοθετείται ευρέως από τους περισσότερους IoT παρόχους, συμπεριλαμβανομένων των AWS, Azure και IBM Watson. Επιπρόσθετη ευελιξία επιτυγχάνεται εάν η IoT λύση υποστηρίζει API μέσω WebSockets, REST και CoAP.

Hardware-agnostic

Η βιομηχανία κατασκευής υλικού εξελίσσεται εξαιρετικά γρήγορα και μπορούμε πλέον να παρατηρήσουμε πόσο μικρές και ταυτόχρονα ισχυρές συσκευές έχουν δημιουργηθεί. Οι περισσότερες από αυτές τις συσκευές σήμερα μπορούν να εκτελούν λειτουργικά συστήματα, καθιστώντάς τα εξαιρετικά ευέλικτα και ικανά να εκτελούν προγράμματα γραμμένα σε οποιαδήποτε γλώσσα. Πριν την επιλογής μιας IoT πλατφόρμας, αξίζει να ελεγχθεί ποιες συσκευές υλικού υποστηρίζονται, αν έχει ενσωματωμένες εφαρμογές για συσκευές περιορισμένης μνήμης καθώς και υπολογιστικής ισχύς.

Υποστήριξη (Support)

Η υποστήριξη είναι μια από τις υπηρεσίες που προσφέρονται με επιπλέον κόστος από τις εταιρείες που αναπτύσσουν IoT πλατφόρμες, εξασφαλίζοντας ότι οι πιο πρόσφατες ενημερώσεις και τα ζητήματα ασφαλείας αντιμετωπίζονται εγκαίρως. Υπάρχουν αρκετές επιλογές για την διαμόρφωση της οργάνωσης σχετικά με την υποστήριξη μιας IoT πλατφόρμας όπως: η πρόσληψη ή εκπαίδευση μιας ειδικής ομάδας, χρήση συμφωνιών υποστήριξης με την εταιρεία από την οποία έχει αγοραστεί η πλατφόρμα ή η στήριξη της κοινότητας σε περίπτωση πλατφόρμας ανοιχτού κώδικα. Ένα μείγμα των παραπάνω είναι ίσως το πιο αποτελεσματικό. Ακόμη και αν δεν απαιτείται σύντομα η οποιαδήποτε μορφής υποστήριξη, αυτή η επιλογή θα πρέπει να διερευνηθεί εκ των προτέρων, ώστε να αποφευχθεί η αντιμετώπιση ενός προϊόντος που δεν λειτουργεί ή δεν υποστηρίζεται.

Αρχιτεκτονική και Τεχνολογία (Architecture & Technology)

Η σημασία της αρχιτεκτονικής και των τεχνολογιών που χρησιμοποιούνται στην πλατφόρμα είναι θεμελιώδης καθώς η μελλοντική διατήρησή της εξαρτάται από τον αρχικό σχεδιασμό της. Η πλατφόρμα θα πρέπει να χρησιμοποιεί πλαίσια, εργαλεία και γλώσσες που να έχουν ποιότητα παραγωγής. Για παράδειγμα μια αρχιτεκτονική βασισμένη σε μικρό-υπηρεσίες (micro-services) πρέπει να έχει εξεταστεί μεταξύ άλλων. Ενώ είναι πολύπλοκη ως προς την λειτουργία της, παρέχει την μεγαλύτερη ευελιξία για την αντικατάσταση οποιασδήποτε υπηρεσίας, διατηρώντας παράλληλα το υπόλοιπο σύστημα άθικτο.

Ασφάλεια (Security)

Η ασφάλεια στο διαδίκτυο αντικειμένων αποτελεί ένα από τα πιο καυτά θέματα. Το βασικό πρόβλημα είναι η απουσία βιομηχανικών προτύπων παρόλο που η ασφάλεια αφορά κάθε συνιστώσα του IoT οικοσυστήματος - συσκευές, πλατφόρμες, εφαρμογές και χρήστες. Ο έλεγχος των χαρακτηριστικών ασφαλείας μιας υπάρχουσας IoT πλατφόρμας κρίνεται απαραίτητη. Για παράδειγμα θα πρέπει να υποστηρίζονται πρωτόκολλα όπως της ασφάλειας επιπέδου μεταφορών (TLS) για επικοινωνία με συσκευές και εφαρμογές καθώς και έλεγχος ταυτότητας μεταξύ συσκευών και χρηστών.

Αξίζει επίσης να εξεταστεί εάν η πλατφόρμα διαθέτει δυνατότητες εξουσιοδότησης (authorization), τι είδους διαχείριση κύκλου ζωής δεδομένων χρησιμοποιεί, εάν παρέχει μια επιλογή κρυπτογραφημένης αποθήκευσης των δεδομένων και τέλος, αν υπάρχουν διαθέσιμα έγγραφα ελέγχου ασφαλείας (audit documents).

Κόστος (Cost)

Διαφορετικοί πάροχοι IoT συστημάτων παρέχουν διαφορετικά μοντέλα τιμολόγησης. Στις περισσότερες περιπτώσεις, μια πλατφόρμα ανοιχτού κώδικα μπορεί να χρησιμοποιηθεί δωρεάν. Παρακάτω παρουσιάζονται τα πιο κοινά μοντέλα τιμολόγησης για το IoT:

- πληρωμή ανά κόμβο / έτος
- πληρωμή ανά ενεργό συσκευή
- πληρωμή ανά μήνυμα
- πληρωμή για επιπλέον χαρακτηριστικά
- πληρωμή για υποστήριξη

Η τιμή συχνά δεν περιλαμβάνει το κόστος της υποδομής για τη φιλοξενία (hosting) της IoT λύσης που έχει επιλεγεί.

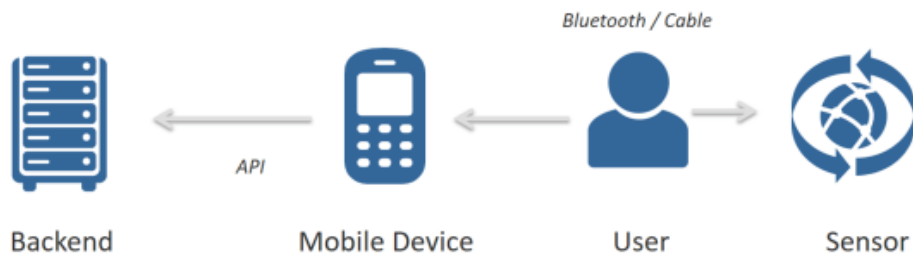
ΚΕΦΑΛΑΙΟ 4 – ΚΡΙΤΗΡΙΑ ΑΞΙΟΛΟΓΗΣΗΣ ΙoT ΠΛΑΤΦΟΡΜΩΝ

Στο συγκεκριμένο κεφάλαιο παρουσιάζονται και αναλύονται συνοπτικά τα κριτήρια βάσει των οποίων αξιολογήθηκαν οι 14 πλατφόρμες οι οποίες θα αναλυθούν λεπτομερώς στο επόμενο κεφάλαιο. Τα κριτήρια σχετίζονται με εργαλεία και υπηρεσίες τα οποία παρέχονται από τις IoT πλατφόρμες με στόχο την ικανοποίηση των αναγκών μιας επιχείρησης ή ενός οργανισμού. Η υποστήριξη ή όχι τέτοιων εργαλείων και υπηρεσιών στις IoT πλατφόρμες καθώς και η διαφορετικότητα που παρουσιάζεται, όπως για παράδειγμα στο κριτήριο της ασφάλειας, σχετικά με το αν υποστηρίζεται το API Key ή όχι, ή στο κριτήριο των πρωτοκόλλων επικοινωνίας, σχετικά με το ποια πρωτόκολλα υποστηρίζονται (MQTT, COAP, STOMP), δημιουργούν μια βάση για τη σύγκριση των IoT πλατφορμών. Η επιλογή των κριτηρίων προκύπτει τόσο από την γενικότερη μελέτη του ρόλου των πλατφορμών στο IoT περιβάλλον καθώς και στο ότι είναι τα στοιχεία εκείνα στα οποία δίνουν έμφαση οι κατασκευαστές αυτών. Επίσης, τα προτεινόμενα κριτήρια συνδέονται άρρηκτα με τα στοιχεία που παρουσιάστηκαν στο Κεφάλαιο 3 μιας και αποτελούν αναπόσπαστο κομμάτι αυτών. Για παράδειγμα, το κριτήριο της προσαρμογής (customization) που αναφέρθηκε στο προηγούμενο κεφάλαιο, συνδέεται άμεσα με το κριτήριο της **Ενσωμάτωσης με πλατφόρμες Τρίτων Παρόχων**. Λαμβάνοντας, λοιπόν, υπόψη τα παραπάνω, θεωρούμε το προτεινόμενο σύνολο κριτηρίων ως ένα επαρκές σύνολο για τη συγκριτική μελέτη μεταξύ των πλατφορμών. Τέλος, θα πρέπει να τονίσουμε ότι εξαιτίας της ύπαρξης εκατοντάδων πλατφορμών στην αγορά, η διαδικασία της επιλογής είναι αρκετά περίπλοκη και εγκυμονεί αρκετούς κινδύνους.

4.1 Integration APIs

Μια διεπαφή προγραμματισμού εφαρμογής (Application Programming Interface, API) είναι ένα σύνολο ρουτινών, πρωτοκόλλων και εργαλείων τα οποία συνεργάζονται για την κατασκευή εφαρμογών λογισμικού και σαν αποτέλεσμα καθορίζουν τον τρόπο αλληλεπίδρασης των διαφόρων στοιχείων. Στις περισσότερες των περιπτώσεων των IoT πλατφορμών γίνεται χρήση από RESTful APIs τα οποία παρέχουν στους προγραμματιστές υλικού και λογισμικού μια τυποποιημένη αρχιτεκτονική για την αξιόπιστη σύνδεση των **ετερογενών** στοιχείων ενός IoT συστήματος.

Με το όρο "integration" αναφερόμαστε στον τρόπο κατά τον οποίον επιτυγχάνεται η διασύνδεση όλων των μερών τα οποία απαρτίζουν ένα IoT οικοσύστημα όπως για παράδειγμα η επικοινωνία ενός IoT αισθητήρα με την εφαρμογή απεικόνισης των δεδομένων τα οποία έχουν συλλεχτεί από αυτόν τον αισθητήρα. Η διασύνδεση αυτή στηρίζεται στα IoT APIs [3], τα οποία αποτελούν τα σημεία αλληλεπίδρασης μεταξύ μιας IoT συσκευής και άλλων στοιχείων ενός του IoT περιβάλλοντος όπως φαίνεται και στην Εικόνα 4.1.



Εικόνα 4-1: IoT APIs

4.2 Ασφάλεια

Άλλη μια σημαντική παράμετρος για μια IoT πλατφόρμα είναι η παροχή ασφαλούς επικοινωνίας βασισμένη σε πρωτόκολλα όπως το SSL / TLS για την μεταφορά των δεδομένων μεταξύ συσκευών και εφαρμογών, καθώς και τον απαιτούμενο έλεγχο ταυτότητας για την επικοινωνία συσκευών και χρηστών [27]. Είναι επίσης πολύ σημαντικό να εξασφαλισθεί η ανταλλαγή των δεδομένων μέσω μιας κρυπτογραφημένης επικοινωνίας. Θα πρέπει να υπάρχει ένα αρκετά ασφαλές περιβάλλον το οποίο θα μπορέσει να υπερασπιστεί τα δεδομένα του τελικού χρήστη από όλα τα είδη επιθέσεων, όπως: backdoors, επιθέσεις DDoS, ευπάθειες ιστού και κακόβουλα προγράμματα.

4.3 Πρωτόκολλα Επικοινωνίας

Ένας από τους βασικούς παράγοντες ενός IoT οικοσυστήματος είναι τα πρωτόκολλα επικοινωνίας. Μέσω των πρωτοκόλλων αυτών επιτυγχάνεται η λήψη των δεδομένων από τις IoT συσκευές οι οποίες έχουν ενσωματωμένους αισθητήρες και στην συνέχεια αποστέλλονται στους διακομιστές [23]. Στις περισσότερες των περιπτώσεων κατά την ανάλυση των πλατφορμών, θα συναντήσουμε πρωτόκολλα όπως είναι το MQTT (Message Queue Telemetry Transport), το AMQP (Advanced Message Queuing Protocol) και το CoAP (Constrained Application Protocol).

4.4 Ανάλυση Δεδομένων

Κάθε IoT πλατφόρμα θα πρέπει να διαθέτει εργαλεία τα οποία θα καθιστούν εύκολη την συλλογή, την επεξεργασία και την ανάλυση των δεδομένων ροής σε πραγματικό χρόνο, ώστε να παρέχεται έγκαιρη γνώση και λήψη αποφάσεων βάσει των νέων πληροφοριών [33]. Οι αναλύσεις των πληροφοριών θα πρέπει να μπορούν να εφαρμοστούν στα IoT δεδομένα για τη δημιουργία πινάκων ελέγχου (Dashboards), αναφορών, απεικονίσεων και διαφόρων ειδοποιήσεων, για την παρακολούθηση της υγείας και της κατάστασης των συνδεδεμένων συσκευών καθώς και για την προβολή των μετρήσεων των αισθητήρων.

Ένα τέτοιο εργαλείο είναι το Amazon Kinesis, Εικόνα 4.1, το οποίο επιτρέπει την επεξεργασία και την διαχείριση μεγάλου όγκου δεδομένων.



Εικόνα 4-1: Εργαλείο Ανάλυσης Δεδομένων

4.5 Απεικόνιση Δεδομένων

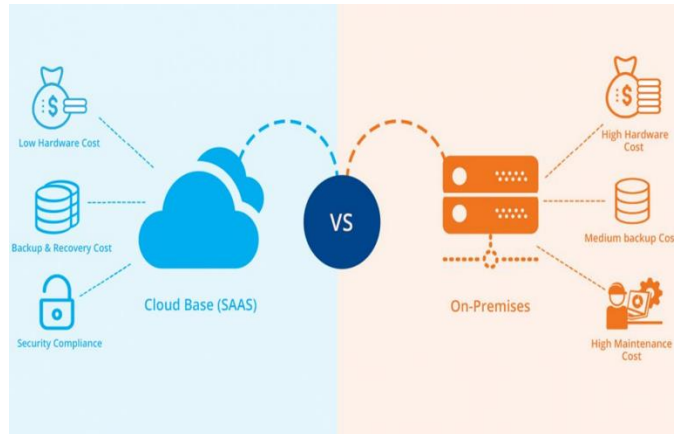
Τα διάφορα εργαλεία απεικόνισης των δεδομένων τα οποία συλλέγονται από τους αισθητήρες παρέχουν στον χρήστη πλεονεκτήματα ως προς την ανάλυση αυτών με αποτέλεσμα να φιλτράρονται και να εξάγονται σημαντικότερες πληροφορίες. Ένα εργαλείο απεικόνισης δεδομένων θα μπορούσε να αποτελείται από ένα πλούσιο σύνολο από widgets, όπως διαγράμματα, χάρτες, πίνακες κ.λπ. Τα συγκεκριμένα widgets μπορούν να χρησιμοποιηθούν για την απεικόνιση διαφορετικών τύπων δεδομένων όπως στατιστικών στοιχείων, γεωγραφικών εντολών, μεταδεδομένων (metadata), φίλτρων, ενημερώσεων λογισμικού και άλλων. Στην Εικόνα 4.2 απεικονίζεται ένα τέτοιο εργαλείο.



Εικόνα 4-2: Εργαλείο απεικόνισης δεδομένων

4.6 Hosting

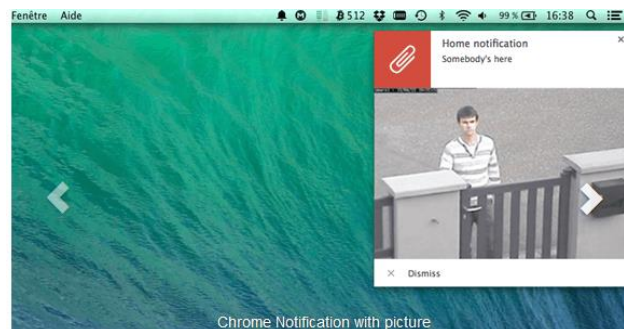
Ένας ακόμη παράγοντας που καθορίζει την επιλογή μιας IoT πλατφόρμας και συμμετέχει στα κριτήρια αξιολόγησης είναι το εάν ο χρήστης μπορεί να αποφασίσει για τον τόπο της διάταξης της IoT λύσης ή εάν η διάταξη περιορίζεται στους διακομιστές παρόχων μέσω μιας cloud υποδομής.



4.7 Push Notifications

Με τον όρο "push notifications" αναφερόμαστε στον μηχανισμό εκείνο μέσω του οποίου αποστέλλονται ειδοποιήσεις στους χρήστες για διάφορα γεγονότα τα οποία συνέβησαν και θεωρούνται κρίσιμα για την ομαλή λειτουργία ενός IoT συστήματος. Στόχος είναι να πραγματοποιηθεί η αποστολή μιας ειδοποίησης μέσω διαφόρων τρόπων όπως κάποιου sms ή email σε οποιαδήποτε συσκευή που διαθέτει ο χρήστης όπως έναν προσωπικό φορητό ή επιτραπέζιο υπολογιστή, ένα κινητό τηλέφωνο ή ένα tablet.

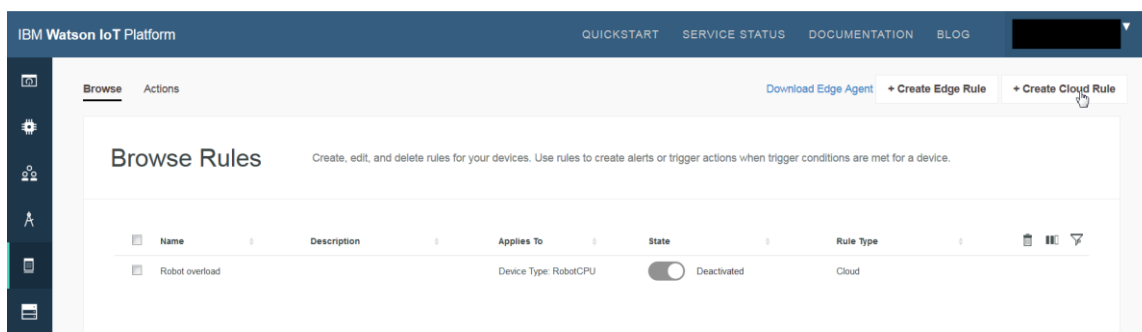
Στην Εικόνα 4.3, παρατηρούμε ένα παράδειγμα στο οποίο ο χρήστης λαμβάνει μία ειδοποίηση μέσω email στον υπολογιστή του διότι κάποιος επισκέπτης βρίσκεται στο σπίτι του. Η επικοινωνία μεταξύ των εφαρμογών που αποστέλλουν τις ειδοποιήσεις και των συσκευών του χρήστη γίνεται μέσω APIs.



Εικόνα 4-3: Παράδειγμα ειδοποίησης

4.8 Ενεργοποίηση συμβάντων (Trigger)

Μία ακόμη σημαντική παράμετρος αξιολόγησης είναι η διαθεσιμότητα εργαλείων μέσω των οποίων τα δεδομένα τα οποία λαμβάνονται και αναλύονται από τις IoT συσκευές, θα μπορούν να ενεργοποιούν την εκτέλεση ενός συμβάντος. Μηχανισμοί οι οποίοι βασίζονται σε συγκεκριμένους κανόνες (Rule engines) χρησιμοποιούν τα δεδομένα τα οποία έχουν ληφθεί από τις συσκευές για την παραγωγή εύστοχων και επαναλαμβανόμενων ενεργειών [20]. Στην Εικόνα 4.4 αποτυπώνεται ο μηχανισμός της IoT πλατφόρμας IBM Watson όπου γίνεται απενεργοποίηση της συσκευής λόγω αυξημένων τιμών στην CPU.



Εικόνα 4-4: Ενεργοποίηση Συμβάντος

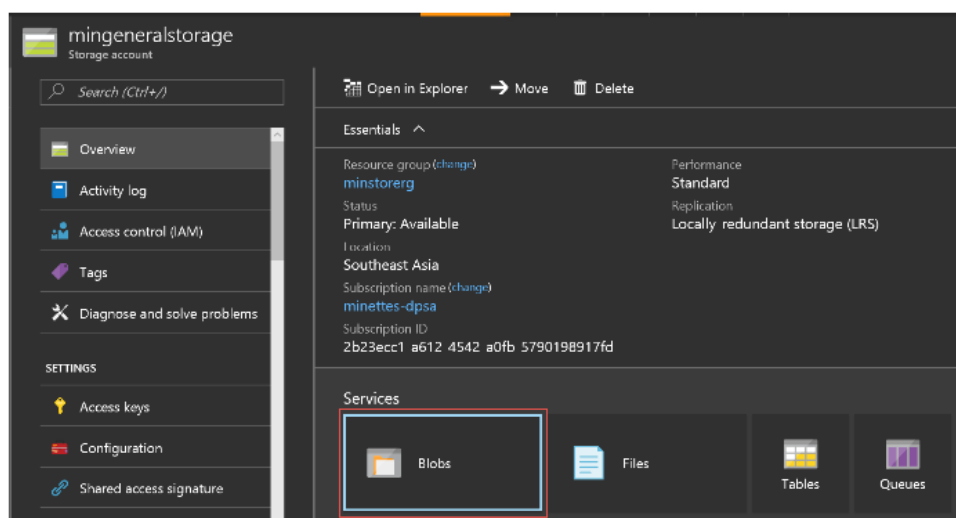
4.9 Αποθήκευση Δεδομένων

Εξ αιτίας της πολυπλοκότητας των IoT υλοποιήσεων, οι πλατφόρμες θα πρέπει να χρησιμοποιούν συνδυασμό τεχνολογιών σχετικά με την αποθήκευση των δεδομένων. Η πρόσβαση στα δεδομένα θα πρέπει να είναι γρήγορη και να υποστηρίζει την αναζήτηση συγκεκριμένων δεδομένων σε πραγματικό χρόνο. Οι τεχνολογίες αποθήκευσης που υιοθετούνται για αυτά τα δεδομένα θα πρέπει να υποστηρίζουν ταυτόχρονες αναγνώσεις και εγγραφές, να περιλαμβάνουν ευρετήρια που μπορούν να παραμετροποιηθούν ώστε να βελτιστοποιούν την πρόσβαση στα δεδομένα και την εκτέλεση των ερωτημάτων. Οι τεχνολογίες αποθήκευσης δεδομένων που υιοθετούνται συχνά από τις IoT υλοποιήσεις περιλαμβάνουν βάσεις δεδομένων NoSQL [26], όπως τις **MongoDB**, **Apache HBase**, **Apache Cassandra**, **AWS DynamoDB** και άλλες.

4.10 Αποθήκευση BLOB

Με τον όρο "blob" αναφερόμαστε σε ένα ακρωνύμιο το οποίο σημαίνει μεγάλο δυαδικό αντικείμενο (binary large object), το οποίο συνήθως περιλαμβάνει μεγάλα αρχεία τα οποία δεν είναι δομημένα όπως εικόνες, βίντεο, αρχεία μουσικής, αρχεία αντιγράφων ασφαλείας κ.λπ. Μπορεί να οριστεί και ως μια συλλογή δυαδικών δεδομένων που αποθηκεύονται ως μία ενιαία μονάδα σε ένα σύστημα διαχείρισης

βάσεων δεδομένων [26]. Το συγκεκριμένο είδος αποθήκευσης αποτελείται από δύο κατηγορίες: 1) αυτήν που ονομάζεται «ζεστή πρόσβαση» για δεδομένα τα οποία είναι συχνά προσβάσιμα και 2) την «κρύα πρόσβαση» όπου η πρόσβαση στα δεδομένα δεν πολύ συχνή. Αρκετές IoT πλατφόρμες διαθέτουν αυτό το χαρακτηριστικό όπως για παράδειγμα η IoT πλατφόρμα της Microsoft, Εικόνα 4.5, στην οποία απεικονίζεται η επιλογή και η διαδικασία κατά την οποία ο χρήστης μπορεί να επιλέξει τον συγκεκριμένο τύπο αποθήκευσης δεδομένων.



Εικόνα 4.5: Αποθήκευση blob δεδομένων (Microsoft Azure IoT)

4.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)

Οι πιο σύγχρονες IoT πλατφόρμες θα πρέπει να διαθέτουν SDKs, δηλαδή ένα σύνολο εργαλείων ανάπτυξης τα οποία θα επιτρέπουν σε έναν χρήστη ή προγραμματιστή να δημιουργήσει μια εφαρμογή για μια συγκεκριμένη πλατφόρμα όπως μια Android ή iOS συσκευή. Η πλατφόρμα AWS, για παράδειγμα, διαθέτει SDKs τα οποία βοηθούν τον χρήστη να συνδέσει εύκολα και γρήγορα διάφορες συσκευές όπως ένα κινητό τηλέφωνο με την πλατφόρμα. Τα SDKs των IoT πλατφορμών θα πρέπει να περιλαμβάνουν βιβλιοθήκες ανοιχτού κώδικα, οδηγούς προγραμματιστών με δείγματα και οδηγούς μεταφόρτωσης, ώστε να μπορούν οι χρήστες να δημιουργούν καινοτόμα IoT προϊόντα ή λύσεις. Στην Εικόνα 4.6 απεικονίζεται ένα δείγμα ενός AWS IoT SDK σε γλώσσα Python [2], το οποίο επιτρέπει στους προγραμματιστές να γράφουν Python scripts ώστε να μπορούν οι συσκευές τους να έχουν πρόσβαση στην πλατφόρμα AWS IoT μέσω του πρωτοκόλλου MQTT.

```

# Import SDK packages
from AWSIoTPythonSDK.MQTTLib import AWSIoTMQTTClient

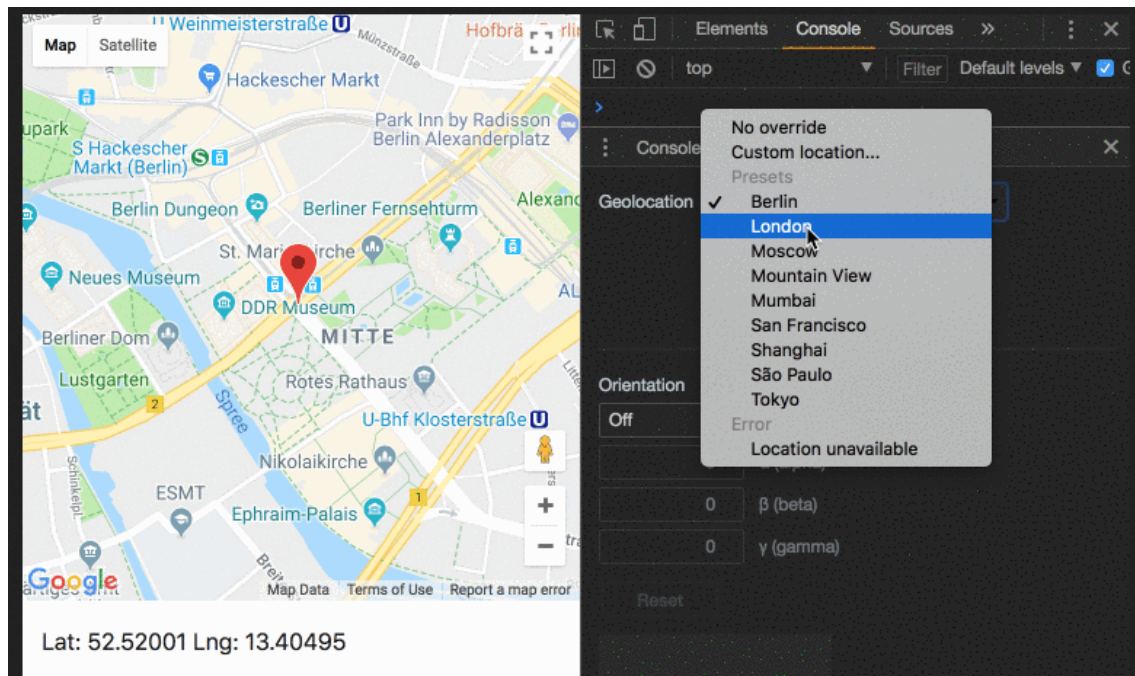
# For certificate based connection
myMQTTClient = AWSIoTMQTTClient("myClientID")
# For Websocket connection
# myMQTTClient = AWSIoTMQTTClient("myClientID", useWebsocket=True)
# Configurations
# For TLS mutual authentication
myMQTTClient.configureEndpoint("YOUR.ENDPOINT", 8883)
# For Websocket
# myMQTTClient.configureEndpoint("YOUR.ENDPOINT", 443)
# For TLS mutual authentication with TLS ALPN extension
# myMQTTClient.configureEndpoint("YOUR.ENDPOINT", 443)
myMQTTClient.configureCredentials("YOUR/ROOT/CA/PATH", "PRIVATE/KEY/PATH", "CERTIFICATE/PATH")
# For Websocket, we only need to configure the root CA
# myMQTTClient.configureCredentials("YOUR/ROOT/CA/PATH")
myMQTTClient.configureOfflinePublishQueueing(-1) # Infinite offline Publish queueing
myMQTTClient.configureDrainingFrequency(2) # Draining: 2 Hz
myMQTTClient.configureConnectDisconnectTimeout(10) # 10 sec
myMQTTClient.configureMQTTOperationTimeout(5) # 5 sec
...

```

Εικόνα 4.6: Python SDKs για την AWS IoT πλατφόρμα

4.12 Geolocation

Η αναζήτηση και η γνώση της γεωγραφικής τοποθέτησης των IoT συσκευών αποτελεί ένα σημαντικό ζήτημα για τις εταιρείες. Για παράδειγμα μια εταιρεία θα πρέπει να γνωρίζει ανά πάση στιγμή την γεωγραφική θέση του εταιρικού της στόλου. Είναι λοιπόν σημαντικό κριτήριο επιλογής για μια IoT πλατφόρμα να διαθέτει τεχνολογίες εντοπισμού θέσης των συσκευών. Η αναζήτηση της γεωγραφικής θέσης γίνεται μέσω GPS, το οποίο όμως δεν αντιπροσωπεύει μια καθολική λύση, διότι απαιτεί μεγάλη κατανάλωση ενέργειας από τις IoT συσκευές καθώς επίσης πολλές φορές ο εντοπισμός σε μέρη όπου υπάρχουν πολλά κτίρια όπως σε μεγάλα αστικά περιβάλλοντα είναι δύσκολο να επιτευχθεί. Για παράδειγμα η IoT πλατφόρμα της Google χρησιμοποιεί Geolocation APIs όπως φαίνεται στην Εικόνα 4.7 αντί του GPS.

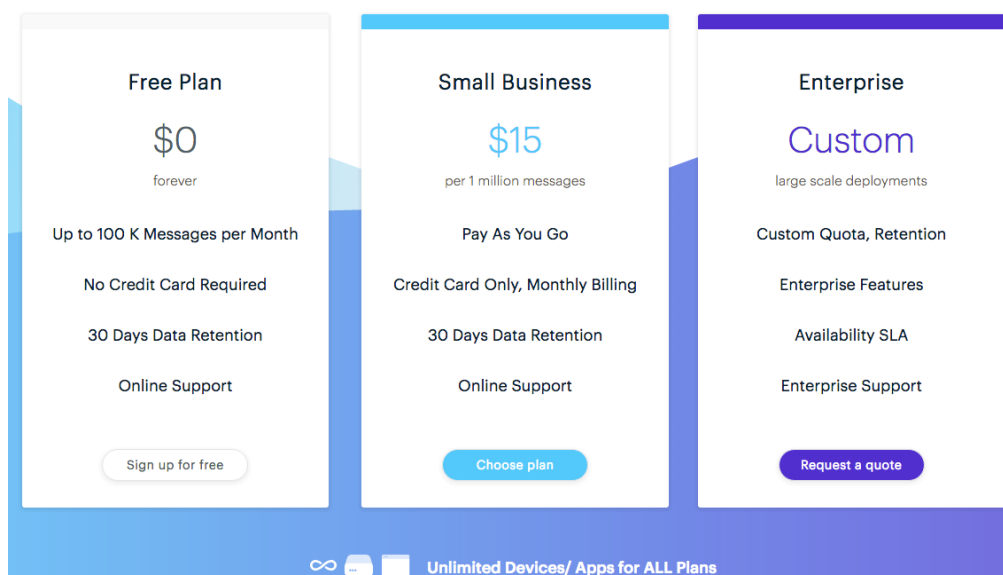


Εικόνα 4.7: Geolocation APIs

Τα APIs γεωγραφικής κατανομής ορίζουν διεπαφές υψηλού επιπέδου οι οποίες διαθέτουν πληροφορίες τοποθεσίας, όπως γεωγραφικό πλάτος και γεωγραφικό μήκος.

4.13 Τιμολόγηση IoT Πλατφορμών

Πολλές εταιρείες κατασκευής και παροχής IoT πλατφορμών διαθέτουν διαφορετικές πολιτικές τιμολόγησης. Εταιρείες όπως η AWS, η Microsoft, η Google και η IBM δημοσιεύουν τις τιμές της εκάστοτε πλατφόρμας στο διαδίκτυο [2], ώστε να είναι δυνατή η έρευνα των διαφορετικών μοντέλων τιμολόγησης και η σύγκριση του συνολικού κόστους. Οι περισσότερες των εταιρειών παρέχουν στους υποψήφιους πελάτες του δοκιμαστικές περιόδους χρήσης των υπηρεσιών που παρέχουν. Επίσης, για παράδειγμα, η IBM και η Google διαθέτουν ένα απλό μοντέλο τιμολόγησης, το οποίο βασίζεται στον όγκο των δεδομένων που ανταλλάσσονται όπως φαίνεται και στην Εικόνα 4.8.



Εικόνα 4.8: Πλάνα Τιμολόγησης IoT Πλατφορμών

4.14 Υποστηριζόμενες γλώσσες προγραμματισμού

Αναφορικά με τις γλώσσες προγραμματισμού που υποστηρίζουν οι IoT πλατφόρμες τόσο σχετικά με την συγγραφή προσαρμοσμένου κώδικα από την πλευρά του διακομιστή (server) όσο και για την συγγραφή κώδικα από την πλευρά του πελάτη (client) υπάρχει μεγάλη ποικιλία η οποία αποτελείται από γλώσσες όπως: Java, C++, JavaScript, Python, Go και πολλές άλλες [33]. Βέβαια υπάρχει μεγάλη διαφοροποίηση ως προς την χρήση μιας συγκεκριμένης γλώσσας προγραμματισμού μέσα σε ένα IoT περιβάλλον, Εικόνα 4.9.



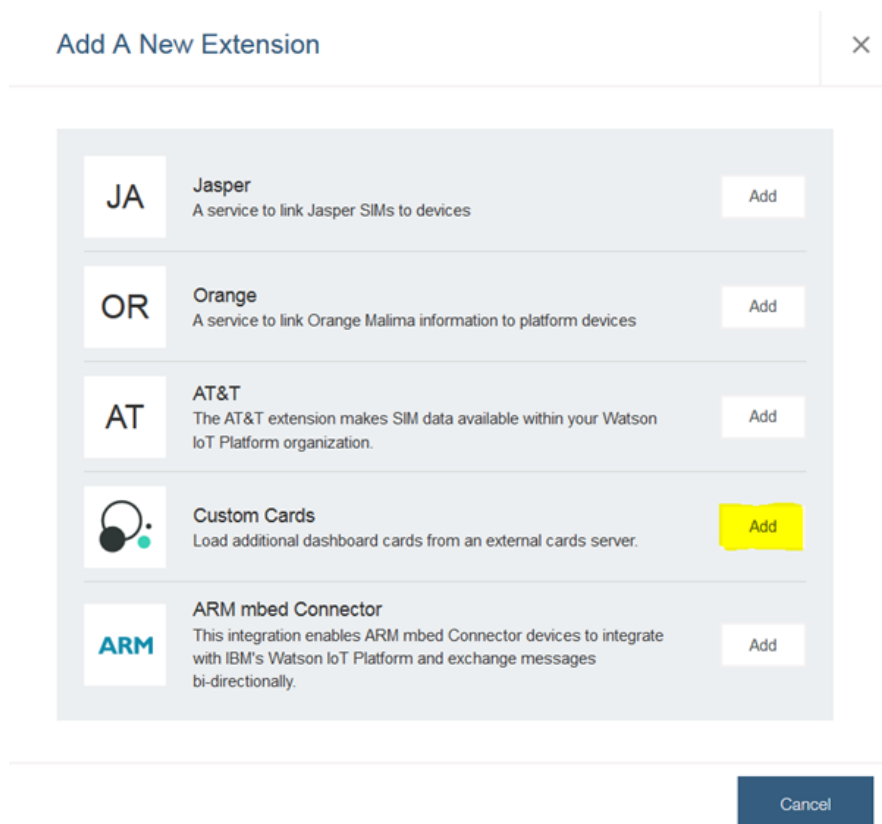
Εικόνα 4.9: Υποστηριζόμενες Γλώσσες Προγραμματισμού IoT

Για παράδειγμα, σε επίπεδο συσκευής όπου η υπολογιστική ισχύ είναι συνήθως περιορισμένη, η γλώσσα C προτιμάται διότι είναι ιδανική για συγγραφή κώδικα χαμηλού επιπέδου, (δηλ. κώδικας πλησίον του

στρώματος υλικού) επειδή δεν απαιτείται πολλή ισχύ επεξεργασίας. Επίσης, η C προτιμάται και για τον προγραμματισμό μικροελεγκτών όπως το Arduino και το Raspberry. Από την άλλη μεριά η Java βρίσκει ευρεία χρήση στο IoT οικοσύστημα. Χρησιμοποιώντας κώδικα που εκτελείται σε JVM (Java Virtual Machine), ο οποίος μπορεί διαθέτει το χαρακτηριστικό της μεταφοράς σε οποιαδήποτε επεξεργαστή, την καθιστούν ιδανική γλώσσα για την δημιουργία IoT εφαρμογών.

4.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων

Ως προς την αξιολόγηση και την επιλογή μιας IoT πλατφόρμας ένας ακόμη σημαντικός παράγοντας είναι η διαθεσιμότητα APIs για την διασύνδεση με εξωτερικές πλατφόρμες που σχετίζονται με επιχειρησιακά συστήματα όπως ERP και CRM, υπηρεσίες ιστού όπως Microsoft Azure και Amazon web services καθώς και πλατφόρμες κοινωνικών μέσων όπως LinkedIn, Twitter και Facebook. Για παράδειγμα η πλατφόρμα IBM Watson μέσω REST API μπορεί να διασυνδεθεί με τη πλατφόρμα Jasper της Cisco (<https://www.jasper.com/>) μέσω της οποίας μπορεί να επιτευχθεί η διαχείριση SIM (Security Identity Module) συσκευών, Εικόνα 4.10.



Εικόνα 4.10: Διασύνδεση με πλατφόρμες τρίτων παρόχων

Σε αυτή την περίπτωση θα μπορεί ο χρήστης μέσω του dashboard της IBM πλατφόρμας να παρακολουθεί στοιχεία για τις IoT συσκευές όπως την κατάσταση της συσκευής, το πρόγραμμα χρεώσεων, τη χρήση μηνυμάτων ανά μία μέρα, τη χρήση μηνυμάτων SMS, τη χρήση μηνυμάτων φωνής, τα όρια υπέρβασης και άλλα.

4.16 Ενσωμάτωση IoT Hardware πλατφορμών

Σε αρκετές περιπτώσεις για την επίτευξη μιας IoT υλοποίησης θα χρειαστεί να υπάρξει επικοινωνία μεταξύ μας IoT πλατφόρμας με μία πλατφόρμα υλικού στην οποία θα βρίσκονται ενσωματωμένοι οι διάφοροι αισθητήρες. Με τον όρο "πλατφόρμες υλικού" αναφερόμαστε σε συσκευές όπως μικροελεγκτές ή SoC (system on a chip) μονάδες οι οποίες αποτελούν απαραίτητο στοιχείο για την ανάπτυξη IoT λύσεων. Σε μία από αυτές τις πλατφόρμες υλικού συγκαταλέγεται και το Arduino. Το Arduino όπως απεικονίζεται και στην Εικόνα 4.11 είναι ένας μικροελεγκτής μονής πλακέτας, δηλαδή μια απλή μητρική πλακέτα ανοικτού κώδικα με ενσωματωμένο μικροελεγκτή και εισόδους/εξόδους, η οποία μπορεί να προγραμματιστεί με τη γλώσσα C++. Με την έλευση και την ανάπτυξη του IoT κατασκευάστηκαν πλατφόρμες Arduino με ειδικά χαρακτηριστικά για την υποστήριξη IoT υλοποιήσεων όπως το Arduino 101 (αναπτύχθηκε με την Intel), το MKR1000, το Arduino WiFi Rev 2 και το MKR Vidor 4000 το οποίο αποτελεί την πρώτη πλακέτα Arduino βασισμένη σε chip FPGA.



Εικόνα 4.11: Arduino

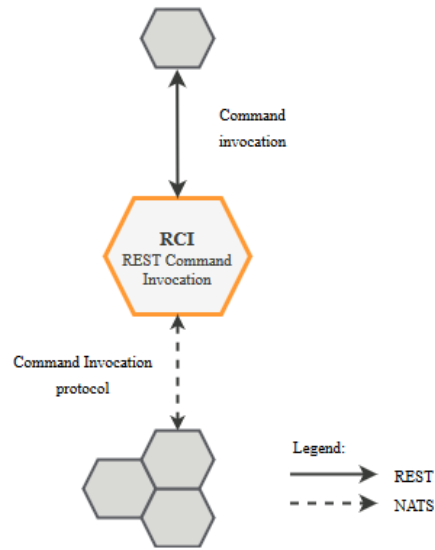
Παρόμοιες πλατφόρμες όπως το Arduino είναι και το Raspberry, το ESP8266, το Banana Pi, το NodeMCU, το Intel Edison και πολλά άλλα.

ΚΕΦΑΛΑΙΟ 5 – ΠΑΡΟΥΣΙΑΣΗ ΙΟΤ ΠΛΑΤΦΟΡΜΩΝ

Στη διπλωματική αυτή έχουν επιλεγθεί 14 διαφορετικές πλατφόρμες, οι οποίες θα συγκριθούν βάσει των κριτηρίων του Κεφαλαίου 4 και θα αναλυθούν βάσει κοινών χαρακτηριστικών. Εξαιτίας του μεγάλου αριθμού πλατφορμών που υπάρχουν και έχουν αναλυθεί στη βιβλιογραφία, η επιλογή των δεκατεσσάρων πλατφορμών που παρουσιάζονται παρακάτω βασίστηκε σε στοιχεία που σχετίζονται με τη δημοφιλία, όπως για παράδειγμα πλατφόρμες σαν την Kaa, Azure IoT και Thingworx, την ενσωμάτωση χαρακτηριστικών που περιεγράφηκαν στο Κεφάλαιο 4, την παροχή δυνατότητας δοκιμών όπως η ThingSpeak και η Ubidots, και γενικότερα με την οπτική της τεχνολογίας αιχμής (state of the art) στο γενικότερο IoT περιβάλλον. Ο Πίνακας 5.1 περιλαμβάνει τις πλατφόρμες που έχουν επιλεγθεί και παρέχει στοιχεία όπως η επίσημη ιστοσελίδα της πλατφόρμας (URL), το έτος κυκλοφορίας (release year) καθώς και αυτό της δημοφιλίας (Ranking), το οποίο βασίζεται στην επισκεψιμότητα της επίσημης ιστοσελίδας και έχει μετρηθεί μέσω του συστήματός Alexa.com. Το Alexa Traffic Ranking είναι ένας μηχανισμός σχετικής κατάταξης των site σε παγκόσμια κλίμακα. Ουσιαστικά πρόκειται για είναι ένα μέτρο δημοτικότητας του ιστότοπου. Κατατάσσει εκατομμύρια ιστοσελίδες με τη σειρά της δημοτικότητας, με μικρότερη τιμή να σημαίνει περισσότερο δημοφιλές. Η κατάταξη υπολογίζεται χρησιμοποιώντας μεθοδολογία που συνδυάζει την επισκεψιμότητα των επισκεπτών κατά τη διάρκεια των τελευταίων τριών μηνών.

Πίνακας 5.1 Προφίλ των IoT πλατφορμών της μελέτης

PLATFORM	URL	RANKING	RELEASE YEAR
Kaa	https://www.kaaproject.org/	497.906	2014
IBM Watson	https://www.ibm.com/watson	701	2016
Azure IoT	https://azure.microsoft.com/en-us/services/iot-hub/	33	2016
WSo2	https://wso2.com/	82.156	2006
AWS IoT	https://aws.amazon.com/iot-core/	11	2015
ThingSpeak	https://thingspeak.com/	50.126	2010
Google IoT	https://cloud.google.com/solutions/iot/	1	2017
Carriots	https://www.altairsmartworks.com/	2.327.849	2011
SiteWhere	https://sitewhere.io/	877.292	2013
Thinger.io	https://thinger.io/	309.049	2015
Ubidots	https://ubidots.com/	127.562	2013
Oracle	https://www.oracle.com/internet-of-things/	403	2016
ThingWorx	https://www.ptc.com/en/products/iiot	16.406	2013
ThingsBoard	https://thingsboard.io/	161.828	2016



Εικόνα 5.1: Υπηρεσία κλήσης εντολών REST (RCI)

Για την εσωτερική επικοινωνία, οι υπηρεσίες Kaa χρησιμοποιούν κυρίως REST APIs και πρωτόκολλα ανταλλαγής μηνυμάτων που εκτελούνται μέσω του συστήματος μηνυμάτων NATS. Το RCI εφαρμόζει το πρωτόκολλο κλήσης εντολών (Command Allocation Protocol) (12 / CIP) για την προώθηση εντολών στα τελικά σημεία.

5.1.2 Ασφάλεια

Από προεπιλογή, η επικοινωνία της Kaa πλατφόρμας με τις συσκευές διασφαλίζεται μέσω των πρωτοκόλλων TLS ή DTLS. Οι συσκευές επαληθεύουν τον διακομιστή (server) ελέγχοντας το πιστοποιητικό του TLS. Ο Kaa διακομιστής υποστηρίζει μια ποικιλία μεθόδων για τον έλεγχο ταυτότητας συσκευών, όπως τα προκαθορισμένα κλειδιά (preshared keys) για τον έλεγχο των πιστοποιητικών TLS από την πλευρά της συσκευής [15]. Επίσης, μέσω της πλατφόρμας διατίθεται ευέλικτη διαχείριση της διάρκειας των διαπιστευτηρίων (credentials), η οποία επιτρέπει την παροχή, αναστολή / ακύρωση των διαπιστευτηρίων της συσκευής.

5.1.3 Πρωτόκολλα Επικοινωνίας

Η συνδεσιμότητα αφορά όλα τα μηνύματα τα οποία ανταλλάσσονται μεταξύ της πλατφόρμας και των συσκευών. Αφορά τον τρόπο βάσει του οποίου οι συσκευές συνδέονται στην πλατφόρμα για να εκτελούν διαφορετικές λειτουργίες. Η πλατφόρμα Kaa υποστηρίζει "lightweight" πρωτόκολλα IoT για την σύνδεση των συσκευών, όπως το MQTT και το CoAP. Η πλατφόρμα επιτρέπει την κατασκευή εφαρμογών οι οποίες

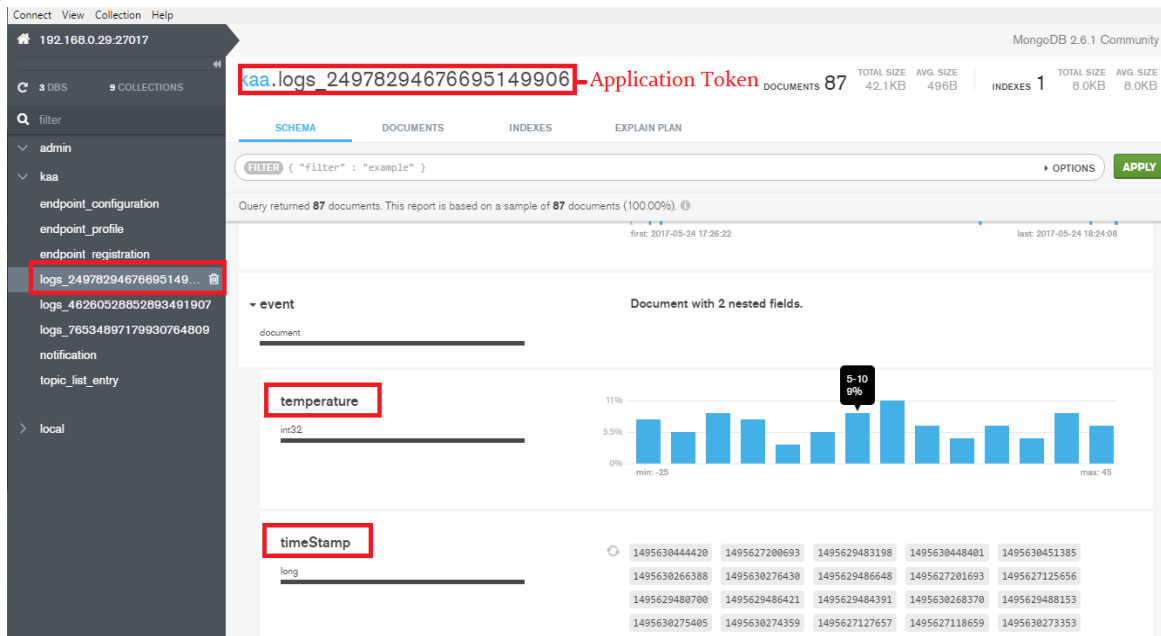
μπορούν να λειτουργούν με οποιοδήποτε τύπο σύνδεσης δικτύου. Το MQTT είναι το προεπιλεγμένο πρωτόκολλο που χρησιμοποιείται. Το πρωτόκολλο Kaa είναι χτισμένο πάνω από το MQTT και το CoAP, και ορίζει βασικούς κανόνες επικοινωνίας μεταξύ της πλατφόρμας και των συσκευών. Το πρωτόκολλο είναι πλήρως ανοικτό, ασύγχρονο, και επιτρέπει αυθαίρετες μορφές μηνυμάτων.

5.1.4 Ανάλυση Δεδομένων

Η συγκεκριμένη πλατφόρμα παρέχει πολλές δυνατότητες σχετικά με την επεξεργασία των συλλεγόμενων δεδομένων [19]. Διαθέτει προσαρμογείς συλλογής δεδομένων (collection adapters) που επιτρέπουν την αποστολή δεδομένων σε διάφορες βάσεις δεδομένων ή συστήματα ανάλυσης δεδομένων. Ακατέργαστα, μη δομημένα δεδομένα μπορούν επίσης να μετατραπούν σε καλά δομημένες χρονικές σειρές, κατάλληλες για ανάλυση, ανάλυση προτύπων, απεικόνιση, χαρτογράφηση κλπ. Μέσω των χρονολογικών σειρών (time series) που παρέχει η πλατφόρμα, η ανάλυση των δεδομένων γίνεται αρκετά ευέλικτη. Εκτός από την εμφάνιση των τιμών για μια χρονολογική σειρά, π.χ. θερμοκρασία, η πλατφόρμα επιτρέπει στους χρήστες να δημιουργούν ετικέτες που επιτρέπουν την προβολή ορισμένων πρόσθετων δεδομένων, όπως η τοποθεσία, η ένταση του φωτός, η υγρασία κλπ.

5.1.5 Απεικόνιση Δεδομένων

Το στοιχείο απεικόνισης δεδομένων της πλατφόρμας περιλαμβάνει ένα πλούσιο σύνολο widgets, όπως διαγράμματα, χάρτες, πίνακες κ.λπ. Τα συγκεκριμένα widgets μπορούν να χρησιμοποιηθούν για την απεικόνιση διαφορετικών τύπων δεδομένων όπως στατιστικών στοιχείων, γεωγραφικών εντολών, μεταδεδομένων, φίλτρων, ενημερώσεων λογισμικού και άλλων [19]. Όλα τα widgets είναι διαμορφώσιμα και επιτρέπουν την παραμετροποίηση τόσο της πηγής των δεδομένων τους καθώς και της οπτικής αναπαράστασης. Στην Εικόνα 5.2 παρουσιάζεται ένα παράδειγμα απεικόνισης δεδομένων τα οποία έχουν εισαχθεί από την MongoDB βάση δεδομένων όπου είναι αποθηκευμένα.



Εικόνα 5.2: Απεικόνιση Δεδομένων

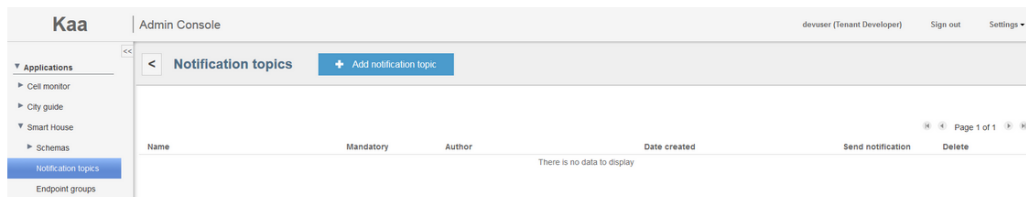
5.1.6 Hosting

Πέραν της cloud υποδομής που διατίθεται για την χρήση των διαφόρων υπηρεσιών της πλατφόρμας, παρέχεται και η δυνατότητα εγκατάστασης σε μία εκ των παρακάτω Linux διανομών [35]:

- Ubuntu and Debian systems
- Red Hat/CentOS/Oracle 5 or Red Hat 6 systems

5.1.7 Push Notifications

Στην πλατφόρμα διατίθεται σύστημα ειδοποιήσεων μέσω του οποίου αποστέλλονται μηνύματα στα τελικά σημεία (endpoints). Σε αντίθεση με τα δεδομένα διαμόρφωσης που αντιπροσωπεύουν την επιθυμητή κατάσταση ενός τελικού σημείου δηλ. μιας IoT συσκευής, οι ειδοποιήσεις χρησιμοποιούνται ως κλήσεις για μια ενέργεια που έλαβε μέρος στο τελικό σημείο. Για παράδειγμα, μια ειδοποίηση μπορεί να αναγκάσει μια Kaa συσκευή να εμφανίσει ένα μήνυμα UI (μια ειδοποίηση χρήστη).



To create a notification topic, click the **Add notification topic** button.

Εικόνα 5.3: Παραμετροποίηση Ειδοποιήσεων

Το σύστημα ειδοποιήσεων της πλατφόρμας οργανώνεται βάσει θεμάτων (topics) όπως διακρίνεται στην Εικόνα 5.3. Κάθε θέμα μπορεί να συσχετιστεί με μία ή περισσότερες ομάδες τελικών σημείων. Για να ληφθεί μια ειδοποίηση, το τελικό σημείο πρέπει να ανήκει σε μία ή περισσότερες ομάδες που σχετίζονται με το αντίστοιχο θέμα ειδοποίησης.

5.1.8 Ενεργοποίηση Συμβάντων (Trigger Actions)

Η διαχείριση των διαφόρων ρυθμίσεων που παρέχονται από την πλατφόρμα είναι απαραίτητη για τον έλεγχο της συμπεριφοράς της συσκευής, τη διαχείριση παραμέτρων σχετικά με την επεξεργασία δεδομένων, των διαφόρων αναλύσεων, και άλλων λειτουργιών. Η πλατφόρμα Kaa επιτρέπει τη εφαρμογή όλων αυτών των λειτουργιών παρέχοντας τη δυνατότητα διαχείρισης παραμέτρων που λειτουργεί με αυθαίρετες δομές δεδομένων (arbitrary data).

Το προεπιλεγμένο πρωτόκολλο διαχείρισης ρυθμίσεων της Kaa υποστηρίζει και τις λειτουργίες **push and pull**. Με άλλα λόγια, μια συσκευή μπορεί να εγγραφεί (subscribe) ώστε να ειδοποιείται κάθε φορά που συμβαίνει κάποια αλλαγή ή εναλλακτικά μπορεί περιοδικά να κάνει το λεγόμενο **poll** ώστε να ενημερώνεται για τις αλλαγές οι οποίες συμβαίνουν.

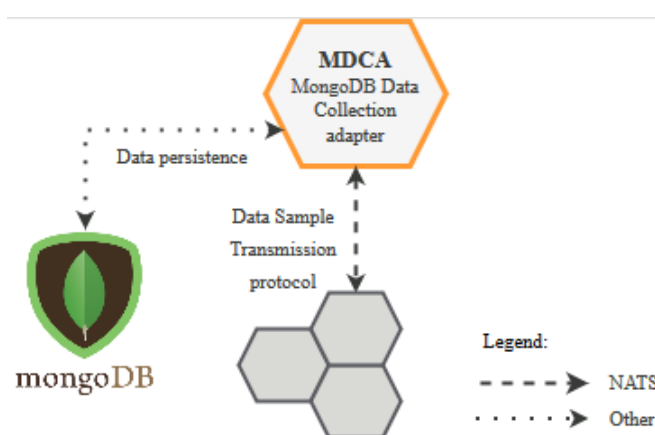
Όταν η συσκευή αλλάξει την τρέχουσα διαμόρφωσή της (configuration), η Kaa δημιουργεί ένα συμβάν ότι κάποια αλλαγή πραγματοποιήθηκε. Αυτά τα συμβάντα μπορούν να χρησιμοποιηθούν για την ενεργοποίηση διαφορετικών ενεργειών για παράδειγμα αποστολής ειδοποιήσεων μέσω κινητού τηλεφώνου καθώς και διαδικασιών αυτοματοποίησης εντός και εκτός της πλατφόρμας.

5.1.9 Αποθήκευση Δεδομένων

Η Kaa πλατφόρμα διαθέτει έναν προσαρμογέα συλλογής δεδομένων βασισμένο σε MongoDB, ο οποίος ονομάζεται MDCA, και η κύρια λειτουργία του είναι να λαμβάνει δείγματα δεδομένων από την συσκευή (endpoint), τα οποία αποθηκεύονται στην MongoDB [2]. Κάθε δείγμα δεδομένων αποθηκεύεται ως ξεχωριστή εγγραφή. Το δείγμα δεδομένων αποθηκεύεται μαζί με τα επόμενα μεταδεδομένα (metadata):

- serverTimestamp, ώρα άφιξης μηνύματος στον προσαρμογέα MDCA.
- appVersionName, όνομα έκδοσης εφαρμογής μιας συσκευής που παράγει ένα δείγμα δεδομένων
- endpointID, το αναγνωριστικό μιας συσκευής (endpoint) που παρήγαγε ένα δείγμα δεδομένων.

Το MDCA υποστηρίζει μια σειρά διεπαφών για την εκτέλεση του ρόλου. Οι βασικές υποστηριζόμενες διεπαφές συνοψίζονται στο ακόλουθο Εικόνα 5-2.



Εικόνα 5.2 Διεπαφές MCDA

Η υψηλή διαθεσιμότητα ενός συμπλέγματος (cluster) επιτυγχάνεται με την ανάπτυξη της βάσης δεδομένων SQL σε λειτουργία HA. Η Kaa υποστηρίζει επίσημα τη MariaDB και την PostgreSQL ως ενσωματωμένες βάσεις δεδομένων SQL αυτήν τη στιγμή.

5.1.10 Αποθήκευση Blob

Μέσω της πλατφόρμας επιτρέπεται η συλλογή και η αποθήκευση τόσο δομημένων όσο και μη δομημένων δεδομένων όπως απλοί αριθμοί, κείμενο απλό ή σύνθετο καθώς και πίνακες (arrays) ή ένθετα αντικείμενα (nested objects).

5.1.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)

Προκειμένου να εξοικονομηθεί χρόνος για τους προγραμματιστές η πλατφόρμα Kaa προσφέρει SDKs τελικών σημείων. Ένα SDK τελικού σημείου είναι μια βιβλιοθήκη που παρέχει επικοινωνία, ταξινόμηση δεδομένων και άλλες λειτουργίες που διατίθενται στην Kaa πλατφόρμα για έναν συγκεκριμένο τύπο SDK. Κάθε τύπος SDK έχει σχεδιαστεί για να ενσωματώνεται στην εφαρμογή του πελάτη. Στον ακόλουθο Πίνακα 5-1 παρέχονται πληροφορίες σχετικά με τις υποστηριζόμενες πλατφόρμες και SDKs.

Platform	C	C++	Objective-C	Java
Linux	Supported	Supported		Supported
Windows		Supported		Supported
QNX Neutrino RTOS	Supported			
Generic Desktop				Supported
Android				Supported
iOS			Supported	
Raspberry Pi	Supported	Supported		
Intel Edison		Supported		
Beaglebone	Supported	Supported		
Samsung Artik 5		Supported		

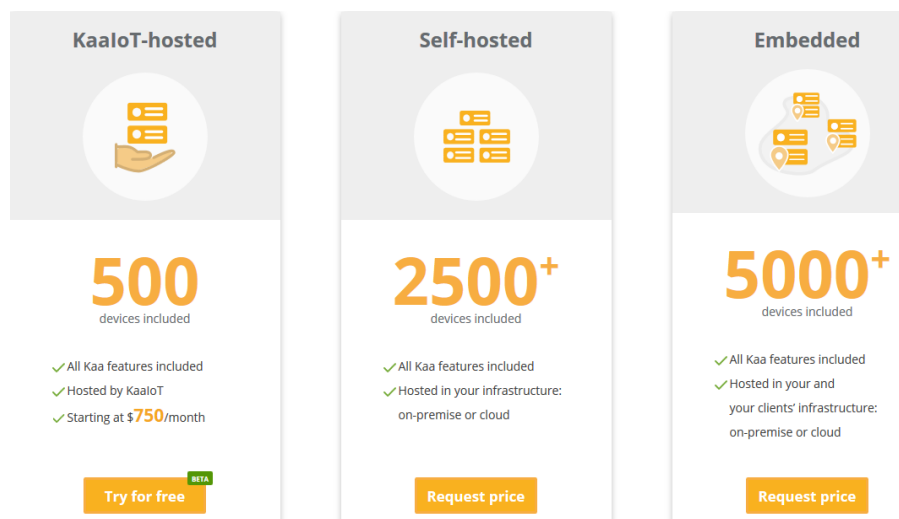
Πίνακας 5.1: Υποστηριζόμενες πλατφόρμες και SDKs

5.1.12 Geolocation

Με την χρήση της υπηρεσίας Google Maps και των κατάλληλων APIs, υποστηρίζεται η δυνατότητα αναγνώρισης ή εκτίμηση της πραγματικής γεωγραφικής θέσης ενός IoT αντικειμένου

5.1.13 Πλάνα τιμολόγησης

Η πλατφόρμα προσφέρει διάφορα ευέλικτα πλάνα τιμολόγησης τόσο για τους προγραμματιστές όσο και για τις εταιρείες [16]. Η τιμολόγηση όπως απεικονίζεται και στην Εικόνα 5.4, βασίζεται κυρίως στον αριθμό των τελικών συσκευών τα οποία διασυνδέονται με την πλατφόρμα καθώς και με την υποδομή εγκατάστασης (on premise or cloud).



Εικόνα 5.4 Επιλογές συνδρομής - τιμολόγησης

5.1.14 Υποστηριζόμενες Γλώσσες Προγραμματισμού

Η πλατφόρμα Kaa αποτελείται από SDKs διακομιστή (server) και τελικού σημείου (endpoint). Ο διακομιστής Kaa υλοποιεί το back-end τμήμα της πλατφόρμας, τις διεπαφές ενοποίησης και προσφέρει διάφορες δυνατότητες διαχείρισης. Ένα SDK τελικού σημείου είναι μια βιβλιοθήκη που παρέχει επικοινωνία, ταξινόμηση δεδομένων και άλλες λειτουργίες που είναι διαθέσιμες στο Kaa για συγκεκριμένο τύπο ενός τελικού σημείου (π.χ. βασισμένο σε Java, C, C++, Object-C) [16]. Αυτό το SDK μπορεί να χρησιμοποιηθεί για τη δημιουργία Kaa endpoints, τα οποία είναι οποιοδήποτε λογισμικό που χρησιμοποιεί τη λειτουργικότητα της πλατφόρμας και είναι εγκατεστημένο σε ορισμένες συνδεδεμένες συσκευές.

5.1.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων

Μέσω της χρήσης REST APIs, επιτυγχάνεται η επικοινωνία με πλατφόρμες και υπηρεσίες τρίτων παρόχων όπως οι Amazon web services και Microsoft Azure

5.1.16 Ενσωμάτωση με Hardware πλατφόρμες

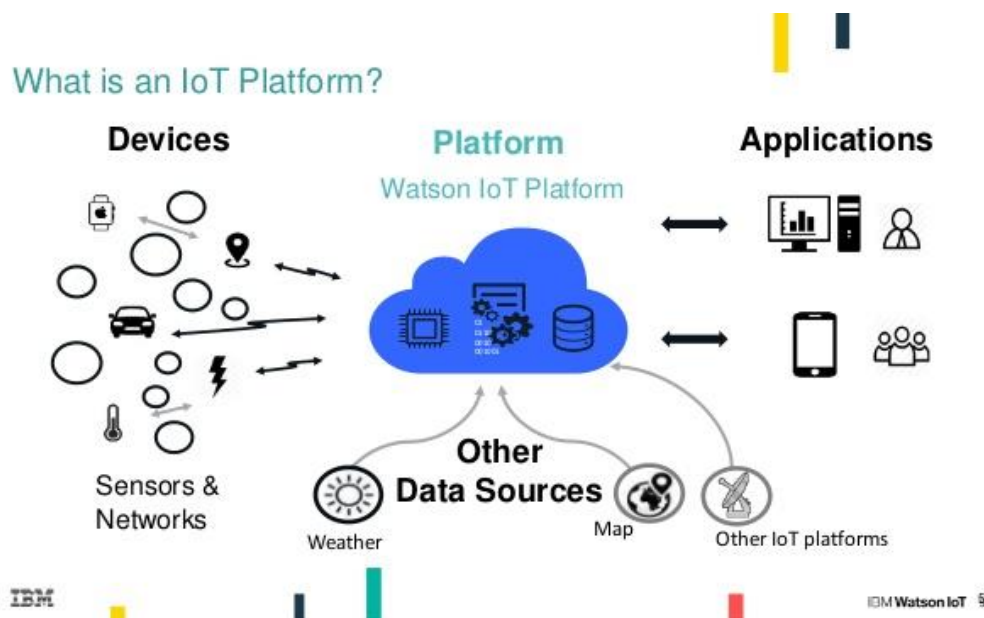
Μέσω των SDKs που παρέχονται από την πλατφόρμα, μπορούν να κατασκευαστούν εφαρμογές για hardware πλατφόρμες όπως είναι οι: Intel Edison, Raspberry Pi, Samsung Artik, Beaglebone και άλλες.

5.1.17 Συμπεράσματα

Η πλατφόρμα Kaa IoT είναι μία από τις λίγες επιλογές πλατφορμών ανοιχτού κώδικα στην αγορά σήμερα η οποία χαρακτηρίζεται ως "business - ready". Η Kaa είναι μια αυτόνομη IoT πλατφόρμα που μπορεί να μεταφορτωθεί με την άδεια Apache 2.0, καθιστώντας την εντελώς δωρεάν για χρήση. Η Kaa μπορεί να λειτουργεί τόσο ως πύλη (gateway) των IoT συσκευών όσο και ως μεσίτης μηνυμάτων (message broker), μεταφέροντας δεδομένα από μια σειρά ετερογενών συσκευών [24]. Επίσης, περιλαμβάνει μια αρχιτεκτονική βασισμένη σε "microservices" επιτρέποντας σε κάθε στοιχείο της πλατφόρμας να προσαρμόζεται ή να αντικαθίσταται με λογισμικό τρίτου κατασκευαστή.

5.2 IBM Watson IoT platform

Η πλατφόρμα IBM Watson IoT (<https://www.ibm.com/internet-of-things>) (Εικόνα 5.5) είναι μια πλήρως διαχειριζόμενη cloud υπηρεσία μέσω της οποίας μπορούν να παραχθούν με απλό τρόπο διάφορες ενέργειες από τις IoT συσκευές, οι οποίες είναι διασυνδεδεμένες με την πλατφόρμα. Μέσω μιας απλής καταγραφής και σύνδεσης της IoT συσκευής, είτε πρόκειται για έναν αισθητήρα, μία πύλη (gateway) ή κάτι άλλο μπορεί να ξεκινήσει η αποστολή δεδομένων με ασφάλεια στην cloud υποδομή της πλατφόρμας χρησιμοποιώντας το πρωτόκολλο ανταλλαγής μηνυμάτων MQTT [2]. Παρέχεται η δυνατότητα για ρύθμιση και διαχείριση των συσκευών, χρησιμοποιώντας τον πίνακα ελέγχου (dashboard) ή τα ασφαλή API's, έτσι ώστε οι εφαρμογές να έχουν πρόσβαση και να χρησιμοποιούν σε πραγματικό χρόνο τα δεδομένα. Τα APIs RESTful βοηθούν στη σύνδεση των δεδομένων που προέρχονται από τις συσκευές με το IBM Bluemix, σημείο όπου οι κατασκευάζονται οι διάφορες εφαρμογές.



Εικόνα 5.5: IBM Watson IoT πλατφόρμα

Βασικά χαρακτηριστικά της πλατφόρμας είναι τα εξής:

- Ο τρόπος εγγραφής της συσκευής (Device Registration)
- Ευέλικτη διασύνδεση (Scalable connectivity)
- Ασφαλής επικοινωνία (Security of communication)
- Αποθήκευση δεδομένων (Storage of data)
- Ισχυρός web πίνακας ελέγχου (web dashboard)
- Παροχή υποστήριξης (Provided support)

5.2.1 Integration - APIs

Η υπηρεσία IBM Watson IoT επιτρέπει στις διάφορες εφαρμογές να επικοινωνούν και να επεξεργάζονται τα δεδομένα που συλλέγονται από τις συνδεδεμένες συσκευές, τους αισθητήρες και τις πύλες. Οι εφαρμογές μπορούν να χρησιμοποιούν REST APIs [14] σε πραγματικό χρόνο για να επικοινωνούν με τις IoT συσκευές και να διαχειρίζονται τα διάφορα δεδομένα. Υπάρχουν αρκετά APIs για την ανάπτυξη κώδικα για συσκευές, πύλες (gateways) και εφαρμογές που συνδέονται με την πλατφόρμα IBM Watson IoT. Τα HTTP APIs προστατεύονται με βασικό HTTP έλεγχο ταυτότητας, δημιουργώντας ένα κλειδί API χρησιμοποιώντας τον πίνακα ελέγχου της πλατφόρμας. Τα διάφορα HTTP APIs και η χρήση τους είναι η εξής:

- **Security API:** Διαχείριση του ελέγχου ταυτότητας και της εξουσιοδότησης των χρηστών
- **Data Management API:** Οργάνωση και ενσωμάτωση των δεδομένων που εισέρχονται και εξέρχονται από την πλατφόρμα Watson IoT.
- **Organization API:** Δημιουργία και διαγραφή συσκευών, έλεγχος της χρήσης, της κατάστασης της υπηρεσίας καθώς και διάγνωση των προβλημάτων σύνδεσης συσκευών.
- **Messaging API:** Αποστολή εντολών χρησιμοποιώντας το HTTP

5.2.2 Ασφάλεια

Η ασφάλεια θεωρείται ένα από τα κύρια κριτήρια για την επιλογή μιας IoT πλατφόρμας [17]. Η IBM Watson IoT πλατφόρμα υποστηρίζει πρωτόκολλα ασφαλείας όπως το TLS v1.2, το οποίο εξασφαλίζει ότι οι αλληλεπιδράσεις των IoT συσκευών έχουν πιστοποιηθεί και κρυπτογραφηθεί [27]. Η πλατφόρμα επίσης παρέχει την δυνατότητα για την διαμόρφωση πολλαπλών επιπέδων ασφαλείας ώστε να προστατεύσει τα δεδομένα από μη εξουσιοδοτημένη αναζήτηση και να περιορίσει τις λειτουργίες διαχείρισης σε συγκεκριμένους χρήστες. Τα επίπεδα είναι τα παρακάτω:

- Web application server
- System-level security
- Collection-level security
- Document-level security
- Encryption

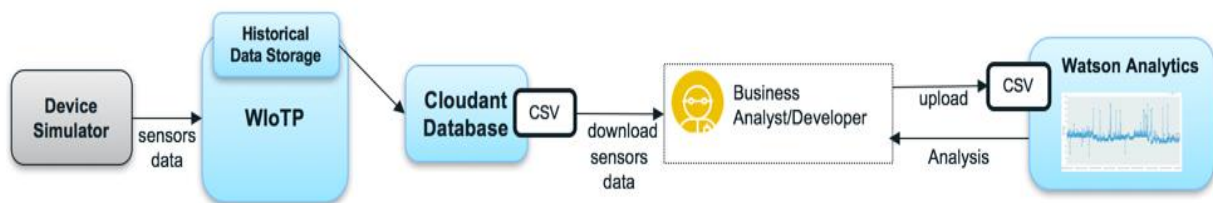
5.2.3 Πρωτόκολλα Επικοινωνίας

Η πλατφόρμα IBM Watson IoT υποστηρίζει οποιοδήποτε περιεχόμενο επιτρέπεται από το πρωτόκολλο MQTT [10]. Το MQTT είναι το κύριο πρωτόκολλο που χρησιμοποιούν οι συσκευές και εφαρμογές για να επικοινωνούν με την πλατφόρμα IBM Watson IoT. Το MQTT είναι ένα πρωτόκολλο μεταφοράς και εγ-

γραφής μηνυμάτων, το οποίο έχει σχεδιαστεί για την αποτελεσματική ανταλλαγή δεδομένων σε πραγματικό χρόνο μεταξύ αισθητήρων και φορητών συσκευών. Το MQTT υποστηρίζεται μέσω TCP και Web-Sockets. Η ασφαλής MQTT επικοινωνία επιτυγχάνεται μέσω της χρήσης κατάλληλων διαπιστευτηρίων και κλειδιών APIs.

5.2.4 Ανάλυση δεδομένων

Χρησιμοποιώντας την πλατφόρμα IBM Watson IoT σε συνδυασμό με το Watson Analytics (WA), παρέχεται οι δυνατότητα στους χρήστες απεικονίσουν και να αναλύουν τα δεδομένα που αποστέλλονται από τις συσκευές που είναι συνδεδεμένες στην πλατφόρμα. Στην Εικόνα 5.3 παρουσιάζεται η διαδικασία απεικόνισης των δεδομένων της πλατφόρμας Watson IoT χρησιμοποιώντας το Watson Analytics (WA) ως εργαλείο ανάλυσης.

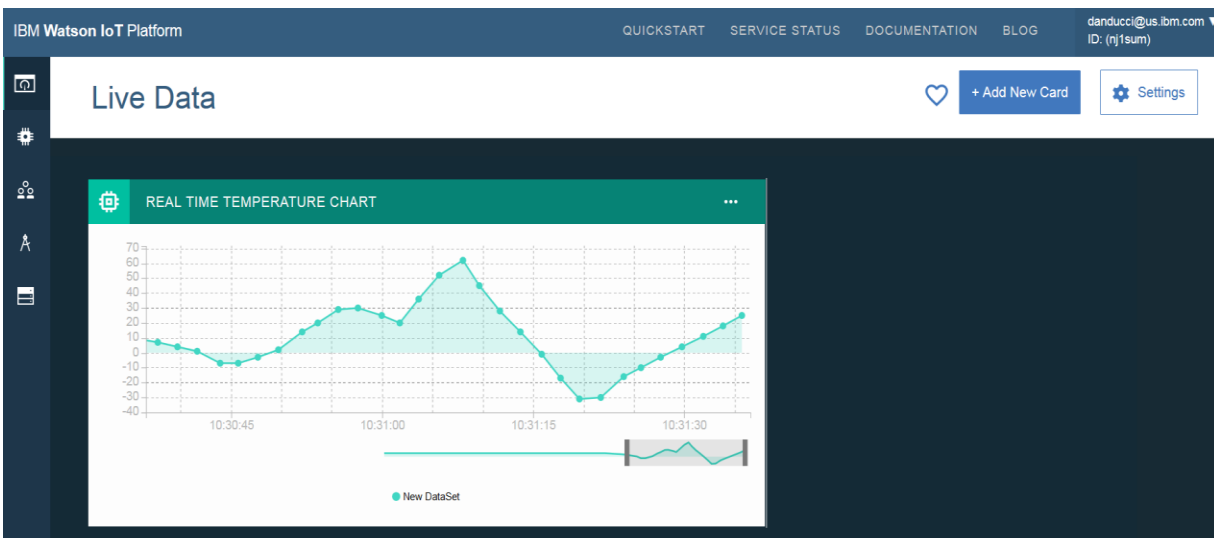


Εικόνα 5.3: Εργαλείο Ανάλυσης Watson Analytics (WA)

Τα δεδομένα της συσκευής που αποστέλλονται στην πλατφόρμα Watson IoT μπορούν να συλλεχθούν και να αποθηκευτούν στο IBM Cloud χρησιμοποιώντας την υπηρεσία IBM Cloudant για IBM Cloud NoSQL DB. Για την συλλογή των δεδομένων, πρέπει πρώτα να έχει συνδεθεί η πλατφόρμα Watson IoT με την υπηρεσία IBM Cloudant [16]. Μετά τη συλλογή δεδομένων, γίνεται η εξαγωγή των δεδομένων σε ένα αρχείο CSV. Στην συνέχεια γίνεται μεταφόρτωση του αρχείου στην WA, όπου μπορούν να απεικονιστούν και να αναλυθούν τα δεδομένα της συσκευής. Τα δεδομένα συσκευής αποθηκεύονται σε ημερήσιες, εβδομαδιαίες ή μηνιαίες βάσεις δεδομένων του IBM Cloudant, ανάλογα με το διάστημα που έχει ρυθμιστεί.

5.2.5 Απεικόνιση δεδομένων

Μέσω της πλατφόρμας παρέχεται η δυνατότητα απεικόνισης των real-time δεδομένων, Εικόνα 5.6 καθώς και των ιστορικών δεδομένων τα οποία αποστέλλονται από τις εγγεγραμμένες συσκευές στην IBM Watson IoT πλατφόρμα.



Εικόνα 5.6: Real-time απεικόνιση δεδομένων

Για την απεικόνιση των δεδομένων η συγκεκριμένη πλατφόρμα κάνει χρήση Πινάκων και Καρτών. Μέσω των δυνατοτήτων που προσφέρονται από αυτά τα δύο εργαλεία μπορούν να δημιουργηθούν προσαρμοσμένα περιβάλλοντα απεικόνισης δεδομένων βασισμένα στις ανάγκες του χρήστη. Επίσης μπορούν να κατασκευαστούν:

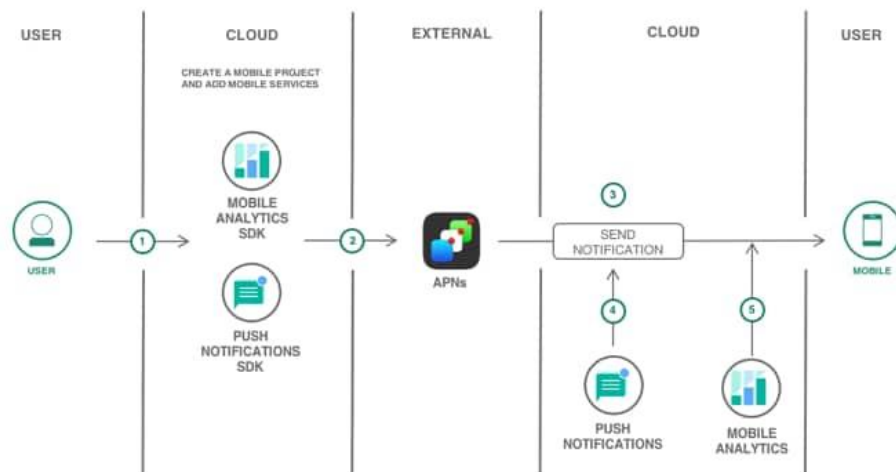
- Γραφήματα απεικόνισης για τα δεδομένα πραγματικού χρόνου (real-time) από τις συσκευές
- Δημιουργία μετρητών για την απεικόνιση φυσικών μεγεθών όπως η ταχύτητα ενός οχήματος και η θερμοκρασία
- Δημιουργία διαγραμμάτων για την εμφάνιση των τρεχουσών τιμών.

5.2.6 Hosting

Η πλατφόρμα IBM Watson IoT είναι μια υπηρεσία που διατίθεται μόνο στο IBM Cloud.

5.2.7 Push Notifications

Η υπηρεσία Push Notifications παρέχει την δυνατότητα για την αποστολή και την ειδοποιήσεων μέσω κινητού τηλεφώνου ή άλλης φορητής συσκευής χρησιμοποιώντας SDKs και REST APIs για τη διαμόρφωση, την παρακολούθηση και την αποστολή των ειδοποιήσεων. Μέσω της πλατφόρμας υποστηρίζεται η αποστολή ειδοποιήσεων τόσο σε Android όσο και iOS συσκευές όπως απεικονίζεται στην Εικόνα 5.4.

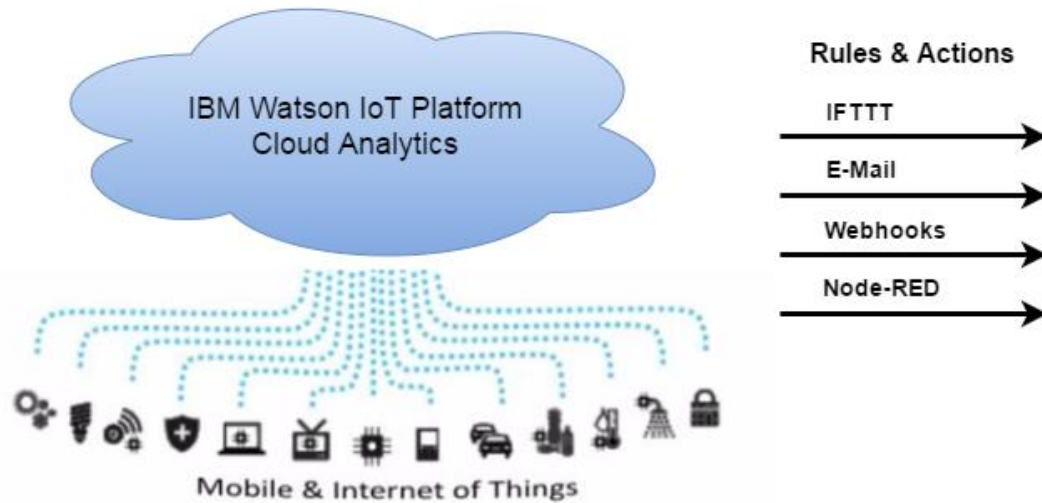


Εικόνα 5.4: Αποστολή ειδοποιήσεων σε iOS συσκευές

Επίσης, η υπηρεσία IBM Push Notifications επεκτείνει την ικανότητά της μέσω της απεικόνισης γραφημάτων για τον όγκο των ειδοποιήσεων που αποστέλλονται, κατηγοριοποιώντας ανάλογα με το είδος των συσκευών και το είδος των ειδοποιήσεων (email, sms).

5.2.8 Ενεργοποίηση Συμβάντων (Trigger Actions)

Μέσω της πλατφόρμας IBM Watson IoT μπορούν οι χρήστες να ορίσουν διάφορους κανόνες και ενέργειες οι οποίες ενεργοποιούνται από τα δεδομένα της IoT συσκευής. Οι κανόνες που ορίζονται από το χρήστη εφαρμόζονται στα δεδομένα ροής σε πραγματικό χρόνο για τον εντοπισμό των συνθηκών που χρειάζονται προσοχή. Ο μηχανισμός δράσης παρέχει τη δυνατότητα να ορισθούν αυτοματοποιημένες διαδικασίες στις συνθήκες ανίχνευσης, όπως η αποστολή ενός μηνύματος ηλεκτρονικού ταχυδρομείου, η ενεργοποίηση ενός IFTTT κανόνα, η εκτέλεση μιας ροής εργασίας ή η χρήση webhooks, Εικόνα 5.7, για σύνδεση σε μια ποικιλία υπηρεσιών ιστού. Τα Webhooks είναι ένας τρόπος με τον οποίο οι εφαρμογές μπορούν να στέλνουν αυτοματοποιημένα μηνύματα ή πληροφορίες σε άλλες εφαρμογές. Τέλος, τα δεδομένα σε πραγματικό χρόνο εμφανίζονται επίσης σε ένα διαμορφωμένο πίνακα ελέγχου για μια γρήγορη προβολή της τοποθεσίας των δεδομένων, των μετρήσεων και των ειδοποιήσεων για τις συσκευές IoT.



Εικόνα 5.7: Διαδικασία Ενεργοποίησης Συμβάντων

5.2.9 Αποθήκευση Δεδομένων

Για την αποθήκευση των δεδομένων η πλατφόρμα χρησιμοποιεί τη βάση δεδομένων IBM Cloudant NoSQL η οποία είναι μια πλήρως οργανωμένη βάση δεδομένων NoSQL ως υπηρεσία (DBaaS). Τα πλεονεκτήματα της συγκεκριμένης βάσης είναι πως μπορεί λειτουργεί χωρίς διακοπή και χειρίζεται μια μεγάλη ποικιλία τύπων δεδομένων όπως JSON, πλήρες κείμενο(full-text) και χωρικά δεδομένα (geospatial). Η Cloud NoSQL DB [16] βάση δεδομένων μπορεί να χειρίζεται ταυτόχρονες αναγνώσεις και εγγραφές και την ίδια στιγμή να παρέχει υψηλή διαθεσιμότητα και ανθεκτικότητα δεδομένων. Αυτή η υπηρεσία είναι συμβατή με το CouchDB και είναι προσβάσιμη μέσω μιας απλής διεπαφής HTTP για μοντέλα κινητών και web εφαρμογών. Έτσι, η σύνδεση με την υπηρεσία βάσης δεδομένων Cloudant [16] στην πλατφόρμα Watson IoT παρέχει στους χρήστες ένα ισχυρό και αξιόπιστο μέσο αποθήκευσης και πρόσβασης στα ιστορικά δεδομένα των συσκευών για περαιτέρω επεξεργασία και ανάλυση. Τα δεδομένα των συσκευών μπορούν να αποθηκευτούν σε ημερήσιες, εβδομαδιαίες ή μηνιαίες βάσεις.

5.2.10 Αποθήκευση Blob

Τα δεδομένα που συλλαμβάνονται από τις IoT συσκευές παράγονται σε συνδυασμό με μορφές δεδομένων, συμπεριλαμβανομένων δομημένων, ημιδομημένων και αδόμητων δεδομένων [36]. Αυτά τα δεδομένα ενδέχεται να περιλαμβάνουν διακριτές μετρήσεις αισθητήρων, μεταδεδομένα για την υγεία των συσκευών ή μεγάλα αρχεία για εικόνες ή βίντεο.

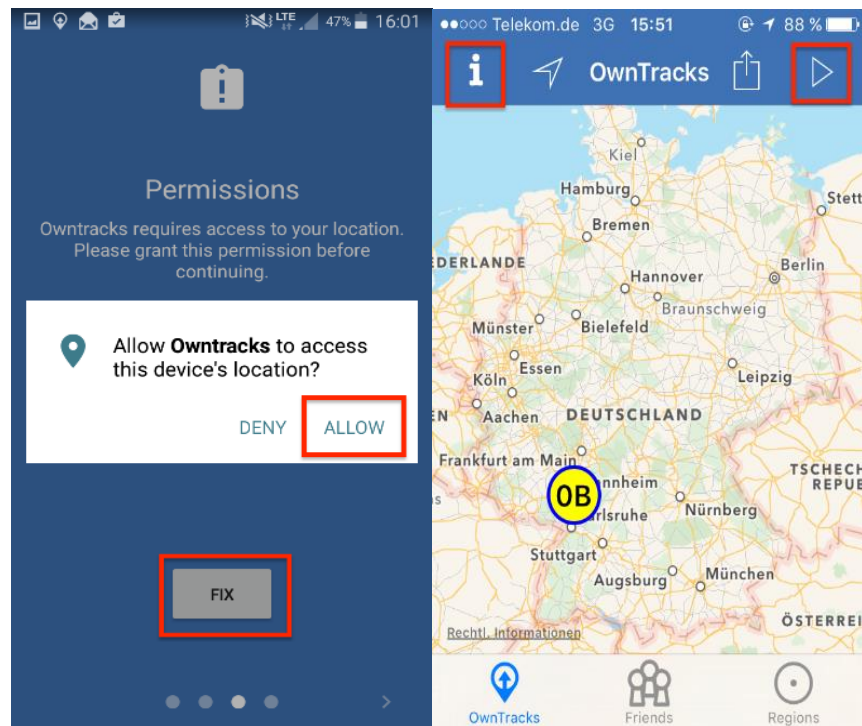
5.2.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)

Τα SDKs τα οποία υποστηρίζονται από την πλατφόρμα IBM Watson είναι τα ακόλουθα [16]:

- Android SDK
- Go SDK
- Java SDK
- Node.js SDK
- Python SDK
- Ruby SDK
- .NET SDK
- OpenWhisk SDK
- Salesforce SDK
- Swift SDK
- Unity SDK

5.2.12 Geolocation

Μέσω της ενσωμάτωσης της υπηρεσίας Owntracks (<https://owntracks.org/>) στην πλατφόρμα επιτυγχάνεται ο εντοπισμός θέσης των IoT συσκευών [36]. Το Owntracks είναι μία εφαρμογή εντοπισμού θέσης για iOS και Android συσκευές όπως απεικονίζεται στην Εικόνα 5.8.



Εικόνα 5.8: Χρήση εφαρμογής Owntracks για τον εντοπισμό θέσης των συσκευών

5.2.13 Πλάνα Τιμολόγησης

Η τιμολογιακή πολιτική της πλατφόρμας σχετικά με την χρέωση των υπηρεσιών, βασίζεται στον αριθμό των συσκευών που έχουν συνδεθεί με την πλατφόρμα, του όγκου των πληροφοριών που ανταλλάσσονται καθώς και με την αποθήκευση δεδομένων.

5.2.14 Υποστηριζόμενες γλώσσες Προγραμματισμού

Η πλατφόρμα IBM Watson IoT υποστηρίζει προγραμματιστικές γλώσσες όπως java, php, Python, Node.js, Go και άλλες [10]. Οι χρήστες μπορούν να δημιουργήσουν και να διαχειριστούν εφαρμογές στην πλατφόρμα Watson IoT χρησιμοποιώντας τις παρακάτω υποστηριζόμενες γλώσσες:

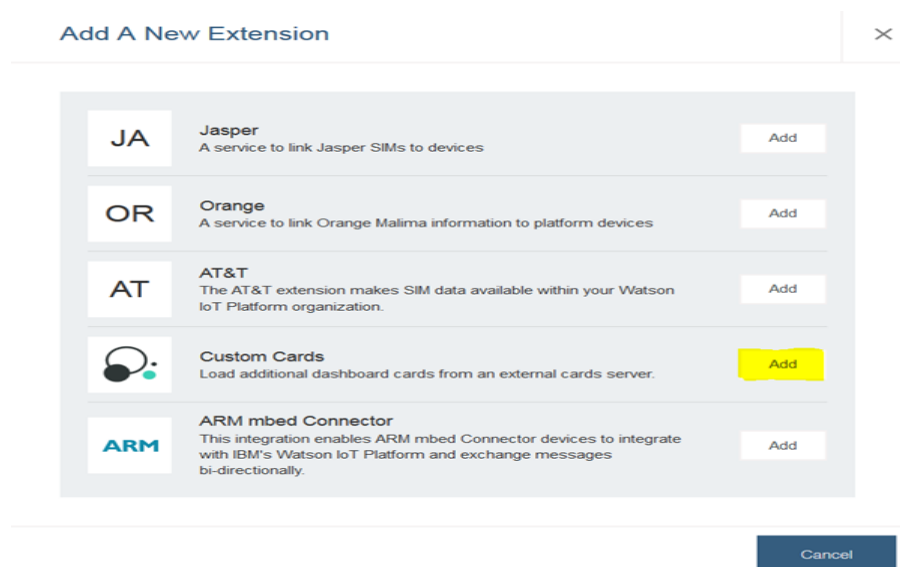
- Python
- Java
- C#
- Node.js

Επίσης οι χρήστες μπορούν να δημιουργήσουν και να διαχειριστούν συσκευές μέσω των παρακάτω υποστηριζόμενων προγραμματιστικών γλωσσών:

- Python
- Java
- C#
- Embedded C
- Mbed C++

5.2.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων

Ως μία υπηρεσία IBM Cloud, η πλατφόρμα Watson IoT μπορεί να ενσωματωθεί σε άλλες υπηρεσίες που φιλοξενούνται στο IBM Cloud. Η ενσωμάτωση εξωτερικής υπηρεσίας επιτρέπει την πρόσβαση σε δεδομένα και λειτουργίες από υπηρεσίες τρίτων όπως για παράδειγμα την Jasper πλατφόρμα [36]. Η πλατφόρμα IBM Watson μέσω REST API μπορεί να διασυνδεθεί με τη πλατφόρμα Jasper της Cisco (<https://www.jasper.com/>) μέσω της οποίας μπορεί να επιτευχθεί η διαχείριση SIM (Security Identity Module) συσκευών. Εικόνα 5.9. Το Jasper ενσωματώνεται στον πίνακα (dashboard) της πλατφόρμας Watson IoT, καθιστώντας δυνατή τη διαχείριση των συσκευών Jasper μέσω του dashboard της πλατφόρμας Watson IoT Platform.



Εικόνα 5.9: Διασύνδεση με πλατφόρμες τρίτων παρόχων

5.12.16 Ενσωμάτωση με Hardware πλατφόρμες

Μέσω των SDKs που παρέχονται από την πλατφόρμα, μπορούν να κατασκευαστούν εφαρμογές για hardware πλατφόρμες όπως είναι οι: ARM mbed, Texas Instruments, Raspberry Pi, Arduino Uno και άλλες [16].

5.12.17 Συμπεράσματα

Η IBM Watson είναι μια ισχυρή πλατφόρμα, η οποία υποστηρίζεται από την υβριδική πλατφόρμα Bluemix της IBM [1]. Παρέχει εύκολες στην χρήση εφαρμογές και διεπαφές για την ανάπτυξη IoT υπηρεσιών κάτι το οποίο την καθιστούν προσιτή προς τους αρχάριους χρήστες. Κάποια από τα πλεονεκτήματα είναι ανταλλαγή δεδομένων σε πραγματικό χρόνο, η ασφαλής επικοινωνία καθώς και η παροχή γνωστικών συστημάτων (cognitive systems), τα οποία συνεργάζονται με τους χρήστες τους σε ένα επίπεδο γνωστικά συμβατό με αποτέλεσμα να παρέχουν εξατομικευμένες και προσαρμοσμένες υπηρεσίες. Τέλος, ως μειονέκτημα χαρακτηρίζεται το υψηλό κόστος των υπηρεσιών.

5.3 Microsoft Azure IoT platform

Η Microsoft παρέχει την υπηρεσία Microsoft Azure (<https://azure.microsoft.com/en-us/services/iot-hub/>), η οποία αποτελεί μια ολοκληρωμένη πλατφόρμα με υπηρεσίες πάνω στις οποίες μπορούν να αναπτυχθούν συστήματα και εφαρμογές του Διαδικτύου των αντικειμένων, συλλέγοντας αποθηκεύοντας και επεξεργάζονται δεδομένα. Οι υπηρεσίες ευφυών (intelligent) συστημάτων βασίζονται στο Microsoft Azure βοηθά τους οργανισμούς να συνδέουν, να διαχειρίζονται, να συλλέγουν και να μετατρέπουν δεδομένα που παράγονται από διάφορες συσκευές σε πολύτιμες πληροφορίες [10]. Ο πυρήνας της IoT υπηρεσίας που παρέχει η Microsoft είναι η cloud πλατφόρμα Microsoft Azure. Παρέχει τη συνδεσιμότητα εκατομμυρίων συσκευών και αισθητήρων μέσω των IoT εφαρμογών, καθώς και δυνατότητες όπως, η απομακρυσμένη πρόσβαση, η παρακολούθηση και η παραμετροποίηση των συνδεδεμένων συσκευών. Παρέχεται επίσης η ανάλυση μεγάλου όγκου δεδομένων καθώς και μεγάλη γκάμα εργαλείων σχετικά με την ανάπτυξη και την ολοκλήρωση IoT εφαρμογών. Σχετικά με το χαρτοφυλάκιο των Azure IoT προϊόντων υπάρχουν διαθέσιμες, δύο μορφής κατηγοριών τεχνολογικές λύσεις ως προ την δημιουργία μιας IoT εφαρμογής:

1) *Platform as a Service (PaaS)*: μπορεί κάποιος να δημιουργήσει την δική του εφαρμογή χρησιμοποιώντας οποιαδήποτε από τις ακόλουθες υπηρεσίες [16]:

- Azure IoT solution accelerators, οι οποίες αποτελούν μια συλλογή έτοιμων λύσεων για επιχειρήσεις, οι οποίες επιτρέπουν την επιτάχυνση της ανάπτυξης προσαρμοσμένων λύσεων IoT [39] ή
- Azure Digital Twins service: η οποία επιτρέπει την μοντελοποίηση του φυσικού περιβάλλοντος για την δημιουργία IoT λύσεων

2) *Software as a Service (SaaS)*: μέσω της Azure IoT Central υποδομής, παρέχεται η δυνατότητα για την δημιουργία μιας IoT επιχειρηματικής λύσης, χωρίς ο ίδιος οργανισμός – επιχείρηση να διαθέτει τους απαιτούμενους πόρους. Το Azure IoT Central είναι μια λύση χωρίς κώδικα (codeless) IoT μέσω της οποίας μπορούν να δημιουργηθούν μοντέλα συσκευών, πίνακες ελέγχου (dashboards) και κανόνες μέσα σε λίγα λεπτά.

Ο πυρήνας μια εφαρμογής IoT αποτελείται από τα ακόλουθα υποσυστήματα :

- 1) τις συσκευές οι οποίες έχουν τη δυνατότητα να επικοινωνούν με ασφάλεια με την cloud υποδομή για την αποστολή και λήψη δεδομένων,
- 2) το hub ή gateway, το οποίο παρεμβάλλεται μεταξύ των συσκευών και της cloud υποδομής και σχετίζεται με την ανταλλαγή των δεδομένων και την διαχείριση των συσκευών,
- 3) τους επεξεργαστές ροής (stream processors), είναι υπεύθυνοι για διαδικασίες όπως η αποθήκευση των δεδομένων και
- 4) μια διεπαφή χρήστη για την απεικόνιση των δεδομένων καθώς επίσης και τη διευκόλυνση σχετικά με την διαχείριση των συσκευών.

Συνίσταται από την Microsoft η χρήση της υπηρεσίας Azure IoT Hub ως πύλη cloud. Ο διανομέας (hub)IoT προσφέρει ασφαλή σύνδεση, τηλεμετρία δηλαδή την εξ υποστάσεως συλλογή δεδομένων καθώς και την αμφίδρομη επικοινωνία με τις συσκευές, συμπεριλαμβανομένης της διαχείρισης συσκευών με δυνατότητες εντολών και ελέγχου. Επιπλέον, ο διανομέας IoT μπορεί να χρησιμοποιηθεί και για την αποθήκευση των μεταδεδομένων (metadata). Για την σύνδεση ενός μεγάλου συνόλου συσκευών συνιστάται η χρήση της υπηρεσίας Azure IoT Hub (DPS). Το DPS επιτρέπει την ταξινόμηση και την καταχώρηση των συσκευών. Συνίσταται επίσης η χρήση των SDK Hub Azure IoT Hub για την ενεργοποίηση της ασφαλούς σύνδεσης των συσκευών και την αποστολή δεδομένων στο cloud [39].

Η επεξεργασία ροής επεξεργάζεται μεγάλες ροές δεδομένων και αξιολογεί κανόνες για αυτές τις ροές. Για την επεξεργασία ροής προτείνεται η χρήση του Analytics Azure Stream για εφαρμογές IoT που απαιτούν πολύπλοκη επεξεργασία κανόνων.

5.3.1 Integration – APIs

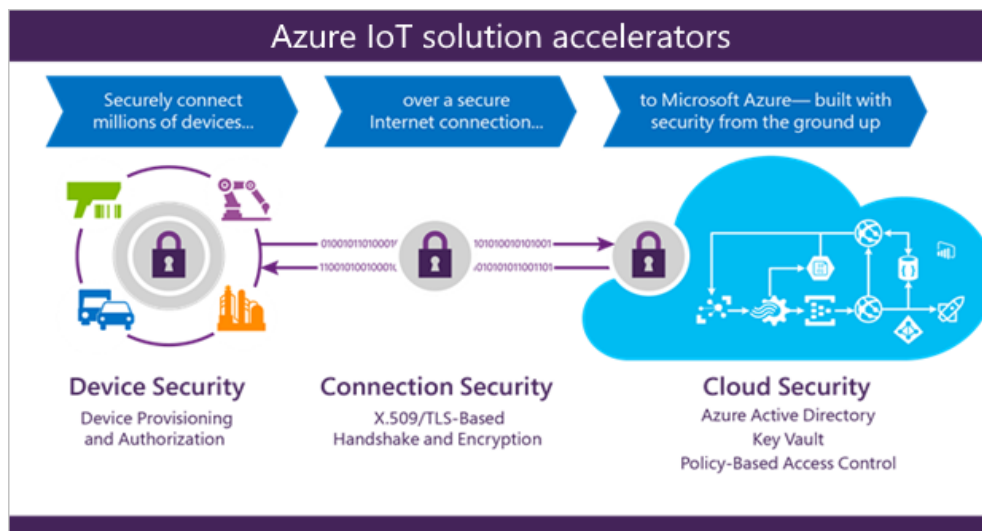
Τα REST APIS που διατίθενται στην πλατφόρμα για προσφέρουν πρόσβαση στις συσκευές, στις υπηρεσίες ανταλλαγής μηνυμάτων καθώς και την προετοιμασία των πόρων στο IoT Hub [16]. Επίσης, μπορεί ο χρήστης να έχει πρόσβαση σε υπηρεσίες μηνυμάτων από μια υπηρεσία IoT που εκτελείται στην Azure υποδομή ή απευθείας μέσω του Διαδικτύου από οποιαδήποτε εφαρμογή που μπορεί να στείλει ένα αίτημα HTTPS και να λάβει μια απάντηση HTTPS [19]. Ακόμη μέσω των APIs παρέχονται δυνατότητες όπως:

- Διαχείριση των "device twins", ανάκτηση και ενημέρωση των ιδιοτήτων των "device twins" καθώς και η επίκληση άμεσων μεθόδων προς τις συσκευές. Τα "device twins" είναι έγγραφα σε μορφή JSON τα οποία αποθηκεύουν πληροφορίες σχετικές με την κατάσταση των συσκευών, συμπεριλαμβανομένων των μεταδεδομένων, παραμετροποιήσεων και διαφόρων άλλων συνθηκών [39].
- Χρήση των APIs για την διαχείριση των ταυτοτήτων των συσκευών στο μητρώο συσκευών
- Διαχείριση εργασιών όπως τον προγραμματισμό ή την ακύρωση μιας ενέργειας.

5.3.2 Ασφάλεια

Για την ασφαλή επικοινωνία μεταξύ των IoT συσκευών και των IoT εφαρμογών η Microsoft παρέχει συστήματα για τον εντοπισμό και την πρόληψη εισβολών καθώς και εργαλεία που συμβάλλουν στον μετριασμό (mitigate) των απειλών [39]. Ο έλεγχος ταυτότητας πολλαπλών παραγόντων (multi-factor) παρέχει ένα επιπλέον επίπεδο ασφάλειας για τους τελικούς χρήστες. Το Azure IoT Hub μαζί με τους "IoT solution accelerators" Εικόνα 5.10 προσφέρουν μια πλήρως διαχειριζόμενη υπηρεσία που επιτρέπει την αξιόπιστη και ασφαλή αμφίδρομη επικοινωνία μεταξύ συσκευών IoT και υπηρεσιών Azure όπως Azure Machine

Learning και Azure Stream Analytics χρησιμοποιώντας τα διαπιστευτήρια ασφαλείας ανά συσκευή και τον έλεγχο πρόσβασης.



Εικόνα 5.10: Azure IoT solution accelerators

Στις συσκευές παρέχεται ένα μοναδικό κλειδί ταυτότητας το οποίο μπορεί να χρησιμοποιηθεί από την υποδομή IoT για επικοινωνία με τη συσκευή ενώ είναι σε λειτουργία. Η διαδικασία είναι γρήγορη και εύκολη στη ρύθμιση. Το παραγόμενο κλειδί με αναγνωριστικό συσκευής που έχει επιλεγεί από τον χρήστη αποτελεί τη βάση ενός διακριτικού που χρησιμοποιείται σε όλες τις επικοινωνίες μεταξύ της συσκευής και του Azure IoT Hub.

Επιπλέον χαρακτηριστικά ασφαλείας σύνδεσης περιλαμβάνουν:

- Η διαδρομή επικοινωνίας μεταξύ των συσκευών και του Azure IoT Hub, ή μεταξύ των πύλων και του Azure IoT Hub, εξασφαλίζεται χρησιμοποιώντας το πρότυπο TLS (Transport Layer Security) και το πρωτόκολλο X.509.
- Για την προστασία των συσκευών από ανεπιθύμητες εισερχόμενες συνδέσεις, το Azure IoT Hub δεν ανοίγει καμία σύνδεση με τη συσκευή. Από την μεριά της συσκευής ενεργοποιούνται όλες οι συνδέσεις.

5.3.3 Πρωτόκολλα Επικοινωνίας

Για την επικοινωνία των συσκευών υπάρχουν διαθέσιμα αρκετά πρωτόκολλα [31]. Θα πρέπει να ληφθεί υπόψη ότι το πρωτόκολλο επικοινωνίας ορίζει τον τρόπο με τον οποίο μεταφέρονται τα ωφέλιμα φορτία (payloads), χωρίς να καθορίζει το σχήμα ή την μορφή τους. Για παράδειγμα η επικοινωνία μπορεί να πραγματοποιηθεί μέσω του πρωτοκόλλου AMQP, αλλά η κωδικοποίηση των δεδομένων μπορεί να γίνει μέσω του Apache Avro ή του JSON. Όπως αναφέραμε προηγουμένως το Azure IoT Hub μπορεί να συνδέσει και ελέγξει εκατομμύρια συσκευές. Το Azure IoT Hub παρέχει υποστήριξη για το AMQP 1.0, το MQTT

3.1.111 και το HTTP 1.1 πάνω από πρωτόκολλο TLS. Η υποστήριξη για πρόσθετα πρωτόκολλα πέρα από τα AMQP, MQTT και HTTP μπορεί να εφαρμοστεί χρησιμοποιώντας ένα μοντέλο προσαρμοσμένης πύλης πρωτοκόλλου (gateway adaption). Το CoAP είναι ένα παράδειγμα πρωτοκόλλου που μπορεί να χρησιμοποιήσει αυτό το μοντέλο.

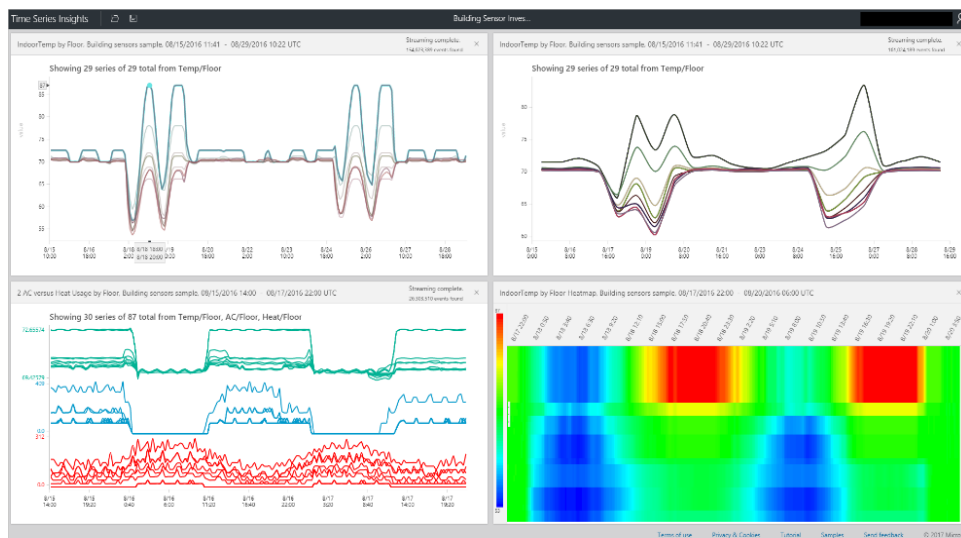
5.3.4 Ανάλυση Δεδομένων

Η ανάλυση των δεδομένων επιτυγχάνεται μέσω της εφαρμογής Azure Stream Analytics (ASA). Πρόκειται για μία υπηρεσία μέσω της οποίας πραγματοποιούνται σύνθετα ερωτήματα σε προσαρμοσμένες χρονικές περιόδους και πολλαπλές ροές. Το Azure Stream Analytics παρέχει τη δυνατότητα στους χρήστες να αποκτήσουν την πλήρη αξία των δεδομένων που παράγονται από τις IoT συσκευές. Επίσης, με το Azure Streaming Analytics (ASA) μπορεί να πραγματοποιηθεί διεξαγωγή σύνθετης επεξεργασίας συμβάντων (Complex Event Processing) πιο κοντά στις περιοχές όπου παράγονται τα δεδομένα με σκοπό την:

- Ανάλυση των δεδομένων των αισθητήρων σε σχεδόν πραγματικό χρόνο με στόχο την ανίχνευση διαφόρων ανωμαλιών [31].
- Αποστολή και ανάλυση των δεδομένων ανεξάρτητα από τη συνδεσιμότητα δικτύου

5.3.5 Απεικόνιση Δεδομένων

Το Azure Time Series Insights (TSI) είναι μια υπηρεσία ανάλυσης, αποθήκευσης και απεικόνισης των δεδομένων. Όλα τα δεδομένα μέσω της TSI υποδομής, αποθηκεύονται σε SSD δίσκους, γεγονός που διασφαλίζει ότι τα δεδομένα είναι γρήγορα προσπελάσιμα για περαιτέρω ανάλυση. Μέσω του Azure Time Series Insights παρέχεται απεικόνιση των δεδομένων από διαφορετικές χρονολογικές σειρές, διάφοροι πίνακες προβολών καθώς και χάρτες θερμότητας (heatmaps). Ένας χάρτης θερμότητας είναι μια γραφική



Εικόνα 5.11: Υπηρεσία Απεικόνισης Δεδομένων TSI

αναπαράσταση δεδομένων στα οποία οι τιμές αντιπροσωπεύονται από τα χρώματα, Εικόνα 5.11. Επιτρέπουν την παρακολούθηση και την καταγραφή των χρηστών με βάση τις κινήσεις του ποντικιού τους. Επίσης μέσω του Azure Time Series Insights προσφέρεται μία βιβλιοθήκη ελέγχου JavaScript, η οποία διευκολύνει την ενσωμάτωση της απεικόνισης των δεδομένων σε προσαρμοσμένες εφαρμογές. Η TSI (Time Series Insights) πραγματοποιεί τέσσερις βασικές εργασίες.

- Πρώτον, είναι πλήρως ενσωματωμένο με cloud gateways όπως το Azure IoT Hub και Azure Event Hubs.
- Δεύτερον, διαχείριση της αποθήκευσης των δεδομένων με τέτοιον τρόπο ώστε να διασφαλιστεί άμεσα η εύκολη και γρήγορη πρόσβαση στα δεδομένα. Τα δεδομένα διατηρούνται έως και 400 ημέρες.
- Τρίτον, παρέχει απεικόνιση εκτός πλαισίου απεικόνιση (out-of-the-box) μέσω του TSI explorer
- Τέταρτον, παρέχει υπηρεσία ερωτήματος όπου μέσω της χρήσης API πραγματοποιείται ενσωμάτωση των δεδομένων σε προσαρμοσμένες εφαρμογές

5.3.6 Hosting

Το Azure IoT Hub είναι μια διαχειριζόμενη πλατφόρμα η οποία φιλοξενείται (hosted) στο σύννεφο (cloud) και λειτουργεί ως κεντρικός κόμβος για την αμφίδρομη επικοινωνία μεταξύ των IoT εφαρμογών και των συσκευών. Το Azure IoT Hub μπορεί να χρησιμοποιηθεί για την δημιουργία και υποστήριξη IoT λύσεων βασισμένο σε αξιόπιστες και ασφαλείς επικοινωνίες μεταξύ εκατομμυρίων συσκευών IoT και back-end λύσεων που φιλοξενούνται στο cloud.

5.3.7 Push Notifications

Μέσω του Azure Event Grid, το οποίο αποτελεί μια πλατφόρμα διαχείρισης συμβάντων, παρέχεται η δυνατότητα ενημέρωσης σε πραγματικό χρόνο για τις αλλαγές που συμβαίνουν στις διάφορες εφαρμογές που φιλοξενούνται στην Azure IoT πλατφόρμα [39]. Το Grid Event, χειρίζεται ενσωματωμένα Azure Events από τις υπηρεσίες Azure καθώς και προσαρμοσμένα συμβάντα από τις εφαρμογές ενημερώνοντας τον χρήστη σε πραγματικό χρόνο.

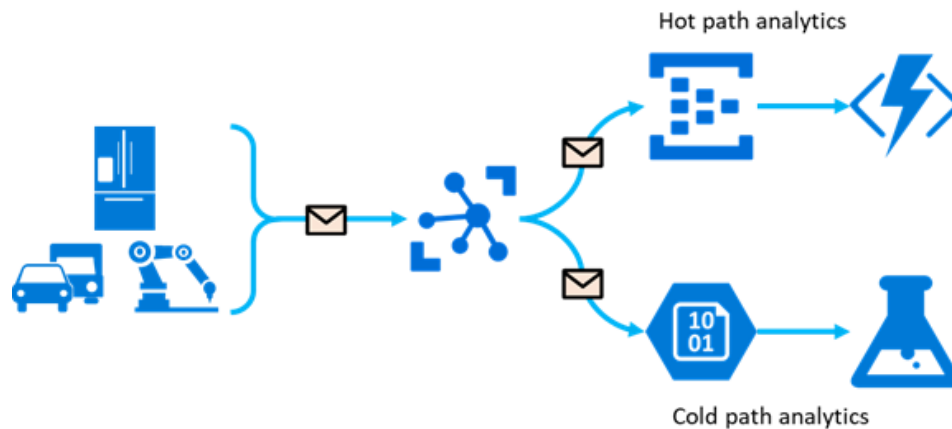
5.3.8 Ενεργοποίηση συμβάντων (Trigger Actions)

Το Azure IoT Hub ενσωματώνεται με την υπηρεσία Azure Event Grid μέσω της οποίας μπορούν να αποστέλλονται διάφορες ειδοποιήσεις συμβάντων που έλαβαν μέρος, σε άλλες υπηρεσίες και μέσω αυτών των ειδοποιήσεων να ενεργοποιούνται μεταγενέστερες διαδικασίες. Επίσης μπορούν να παραμετροποιηθούν κατάλληλα οι επιχειρηματικές εφαρμογές ώστε να λαμβάνουν γνώση για τα γεγονότα

που συνέβησαν στο Azure IoT Hub, ώστε να μπορούν να αντιδρούν σε κρίσιμα συμβάντα με αξιόπιστο και ασφαλή τρόπο [39]. Για παράδειγμα, μπορεί να δημιουργηθεί μια εφαρμογή που να ενημερώνει μια βάση δεδομένων, και να παρέχει ειδοποίηση μέσω ηλεκτρονικού ταχυδρομείου κάθε φορά που μια νέα συσκευή IoT είναι καταχωρείται στο Azure IoT Hub. Το Azure Event Grid είναι μια πλήρως διαχειριζόμενη υπηρεσία δρομολόγησης συμβάντων που χρησιμοποιεί ένα μοντέλο δημοσίευσης-εγγραφής (publish-subscribe).

5.3.9 Αποθήκευση Δεδομένων

Μέσω ενός IoT περιβάλλοντος μπορούν να δημιουργηθούν σημαντικές ποσότητες δεδομένων, ανάλογα με τον αριθμό των συσκευών που βρίσκονται συνδεδεμένες, τη συχνότητα αποστολής των δεδομένων καθώς και το μέγεθος του ωφέλιμου φορτίου (payload). Συνηθίζεται τα δεδομένα να χωρίζονται σε "ζεστά" και "κρύα" δεδομένα όπως απεικονίζεται στην Εικόνα 5.5. Στην πρώτη περίπτωση περιέχονται τα πιο πρόσφατα δεδομένα. Συνιστάται η χρήση του Azure Cosmos DB για την πρώτη επιλογή και η λύση του Azure Blob για την δεύτερη επιλογή [31].



Εικόνα 5.5: Αποθήκευση δεδομένων

Οι λύσεις της "ζεστής" αποθήκευσης αξιολογήθηκαν με βάση τα ακόλουθα κριτήρια. Αυτά τα κριτήρια δεν θα ισχύουν για κάθε λύση IoT αλλά σχεδιάστηκαν για να είναι γενικά εφαρμόσιμα σε όλες τις λύσεις:

- Ασφάλεια: Θα πρέπει να υποστηρίζεται κρυπτογράφηση και έλεγχος ταυτότητας
- Απλότητα: Θα πρέπει να υπάρχει τεκμηρίωση και μια καλά καθορισμένη αρχιτεκτονική.
- Εκτέλεση: Η ανάγνωση και η εγγραφή στη βάση δεδομένων θα πρέπει να είναι γρήγορη και να κλιμακώνεται σε πολλές ταυτόχρονες αναγνώσεις και εγγραφές
- Ευελιξία: Η βάση δεδομένων θα πρέπει να υποστηρίζει την αποθήκευση gigabytes σε terabytes.
- Δυνατότητα Ερωτήματος: Η βάση δεδομένων θα πρέπει να έχει τις δυνατότητες αναζήτησης που απαιτούνται για τη συνολική IoT λύση.

- Τιμή: Η βάση δεδομένων θα πρέπει να είναι προσιτή σχετικά με τις δυνατότητες αποθήκευσης.

5.3.10 Αποθήκευση BLOB

Μέσω της υπηρεσίας Azure Blob Storage επιτυγχάνεται η αποθήκευση τεράστιων ποσοτήτων μη δομημένων δεδομένων. Τα μη δομημένα δεδομένα είναι δεδομένα που δεν τηρούν συγκεκριμένο μοντέλο δεδομένων όπως κείμενο ή δυαδικά δεδομένα [39]. Η αποθήκευση Blob έχει σχεδιαστεί για:

- Παρουσίαση εικόνων ή εγγράφων απευθείας σε ένα πρόγραμμα περιήγησης.
- Αποθήκευση αρχείων για κατανεμημένη πρόσβαση
- Streaming βίντεο και ήχου.
- Καταγραφή αρχείων (log files).
- Αποθήκευση δεδομένων για δημιουργία αντιγράφων ασφαλείας και επαναφορά, την αποκατάσταση καταστροφών και την αρχειοθέτηση.

Οι χρήστες ή οι εφαρμογές-πελάτες μπορούν να έχουν πρόσβαση σε αντικείμενα στο Storage Blob μέσω HTTP / HTTPS, από οπουδήποτε στον κόσμο. Τα αντικείμενα στην αποθήκευση Blob είναι προσβάσιμα μέσω του Azure Storage REST API, του Azure PowerShell, του Azure CLI ή μιας βιβλιοθήκης client Azure Storage.

5.3.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)

Τα SDK της Azure IoT συσκευής αντιπροσωπεύουν ένα σύνολο στοιχείων τα οποία μπορούν να χρησιμοποιηθούν σε συσκευές ή πύλες για να απλοποιήσουν τη συνδεσιμότητα με το Azure IoT Hub. Τα SDK της συσκευής μπορούν να χρησιμοποιηθούν για την υλοποίηση ενός IoT client με το οποίο διευκολύνεται η συνδεσιμότητα με την cloud υποδομή. Παρέχονται βιβλιοθήκες οι οποίες καθιστούν δυνατή τη σύνδεση μιας ετερογενούς σειράς συσκευών και πυλών σε μια λύση που βασίζεται σε Azure IoT υποδομή. Τα SDK της συσκευής Azure IoT είναι ανοιχτού κώδικα. Ενώ μπορούν να απλοποιήσουν την συνδεσιμότητα με το Azure IoT Hub, ωστόσο είναι προαιρετικά και δεν απαιτούνται στην περίπτωση όπου οι προγραμματιστές της συσκευής επιλέξουν να συνδεθούν χρησιμοποιώντας υπάρχοντα πλαίσια και υποστηριζόμενα πρότυπα πρωτοκόλλου. Υπάρχουν δύο κατηγορίες κιτ ανάπτυξης λογισμικού (SDK) [31].

- **SDKs Συσκευών (IoT Hub Device SDKs)**
- **SDKs Υπηρεσιών (IoT Hub Service SDKs)**, επιτρέπουν την δημιουργία εφαρμογών υποστήριξης (backend) για τη διαχείριση των IoT συσκευών

Οι γλώσσες προγραμματισμού που υποστηρίζονται περιλαμβάνουν τις εξής:

- C
- Python

- Node.js
- Java
- .NET

5.3.12 Geolocation

Οι χάρτες Azure είναι μια συλλογή γεωχωρικών υπηρεσιών (geospatial services) και παρέχουν δεδομένα χαρτογράφησης για τις IoT συσκευές. Οι χάρτες Azure αποτελούνται από REST APIs για την απόδοση χαρτών σε πολλαπλά στυλ και δορυφορικές εικόνες καθώς επίσης για την αναζήτηση διευθύνσεων, τόπων και σημείων. Επιπλέον, παρέχουν πληροφορίες μέσω γεωγραφικών αναλύσεων (geospatial analytics).

5.3.13 Πλάνα Τιμολόγησης

Για τον λόγο ότι κάθε IoT λύση είναι διαφορετική, η Azure IoT Hub πλατφόρμα προσφέρει διάφορες επιλογές σχετικά με την τιμολογιακή πολιτική. Το Azure IoT Hub προσφέρει δύο ειδών κατηγορίες τιμολόγησης, την βασική (basic) και την τυπική (standard), οι οποίες διαφέρουν στον αριθμό των λειτουργιών που υποστηρίζουν. Εάν για παράδειγμα η IoT λύση βασίζεται στη συλλογή δεδομένων από τις συσκευές και στην ανάλυση αυτών τότε η βασική βαθμίδα είναι πιθανώς η σωστή επιλογή. Εάν όμως υπάρχει ανάγκη για χρήση πιο εξειδικευμένων εφαρμογών και παραμετροποιήσεων όπως για παράδειγμα ο απομακρυσμένος έλεγχος των IoT συσκευών, τότε ενδεχομένως να πρέπει να γίνει χρήση της τυπικής βαθμίδας. Κάθε κατηγορία του IoT Hub διατίθεται σε τρία μεγέθη, ανάλογα με τον όγκο των δεδομένων που ανταλλάσσονται ανά ημέρα [16]. Αυτά τα μεγέθη αναγνωρίζονται αριθμητικά ως 1, 2 και 3. Για παράδειγμα, κάθε μονάδα ενός διανομέα επιπέδου 1 μπορεί να χειριστεί 400.000 μηνύματα την ημέρα, ενώ μια μονάδα επιπέδου 3 μπορεί να χειριστεί 300 εκατομμύρια. Η χρέωση ωστόσο γίνεται σε μηνιαία βάση.

5.3.14 Υποστηριζόμενες γλώσσες προγραμματισμού

Η πλατφόρμα Microsoft Azure IoT υποστηρίζει προγραμματιστικές γλώσσες όπως .NET, UWP, Java, C, NodeJS, Ruby, Android, iOS και άλλες [16].

5.3.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων

Μέσα από την βιβλιογραφία και την επίσημη ιστοσελίδα της πλατφόρμας, δεν αναφέρεται κάπου η ενσωμάτωση καθώς και η υποστήριξη με υπηρεσίες τρίτων παρόχων.

5.3.16 Ενσωμάτωση με Hardware πλατφόρμες

Η συγκεκριμένη πλατφόρμα υποστηρίζει ένα μεγάλο εύρος hardware πλατφορμών για την ανάπτυξη IoT λύσεων όπως οι: Intel, Raspberry, FreeScale, Texas Instrument και άλλες πολλές [16].

5.3.17 Συμπεράσματα

Η IoT πλατφόρμα της Microsoft παρέχει πλήθος λειτουργιών και υπηρεσιών με στόχο την ανάπτυξη ολοκληρωμένων και επεκτάσιμων IoT λύσεων. Προσφέρει μια ποικιλία δομικών στοιχείων όπως το Azure IoT Edge, το Stack, το Hub, τα Digital Twins καθώς και χάρτες για τον εντοπισμό των συσκευών. Σημαντικό στοιχείο αποτελεί η μεγάλη γκάμα των προεγκατεστημένων εφαρμογών που διαθέτει [14]. Πλεονέκτημα αποτελεί ότι πρόκειται για μια ανοικτού κώδικα πλατφόρμα, παρέχοντας διαλειτουργικότητα και ισχυρές λειτουργικές δυνατότητες. Ωστόσο ως μειονέκτημα χαρακτηρίζεται η πολυπλοκότητα της τιμολόγησης των υπηρεσιών.

5.4 WSO2 IoT Platform

Η εταιρεία WSO2 <https://wso2.com/> η οποία ειδικεύεται στην ανάπτυξη εφαρμογών ανοικτού κώδικα έχει κυκλοφορήσει στην αγορά την νέα έκδοση του IoT WSO2 διακομιστή μέσω της οποίας προσφέρονται διάφορες λύσεις προς τις εταιρείες οι οποίες επιθυμούν να εκμεταλλευτούν τα πλεονεκτήματα του Ίντερνετ των αντικειμένων (IoT) [33]. Όλα τα προϊόντα WSO2 είναι 100% ανοιχτού κώδικα, παρέχουν την απαραίτητη ευελιξία και έχουν σχεδιαστεί με γνώμονα τον ψηφιακό μετασχηματισμό. Η νέα έκδοση του WSO2 IoT Server διαχειρίζεται με τέτοιον τρόπο ένα IoT οικοσύστημα προκειμένου να αναπτύξει και να διαχειριστεί εφαρμογές, επιτυγχάνοντας ασφαλούς πρόσβασης στα δεδομένα και προσφέροντας τη δυνατότητα ανάλυσης από διάφορες συσκευές [37]. Επιπλέον, ξεχωρίζει λόγω της συμβατότητάς της με άλλα συστήματα τρίτων χάρη στην εκτεταμένη χρήση ανοικτών προτύπων επικοινωνίας. Η τελευταία έκδοση του διακομιστή WSO2 IoT εισάγει μια σειρά από καινοτόμες υπηρεσίες:

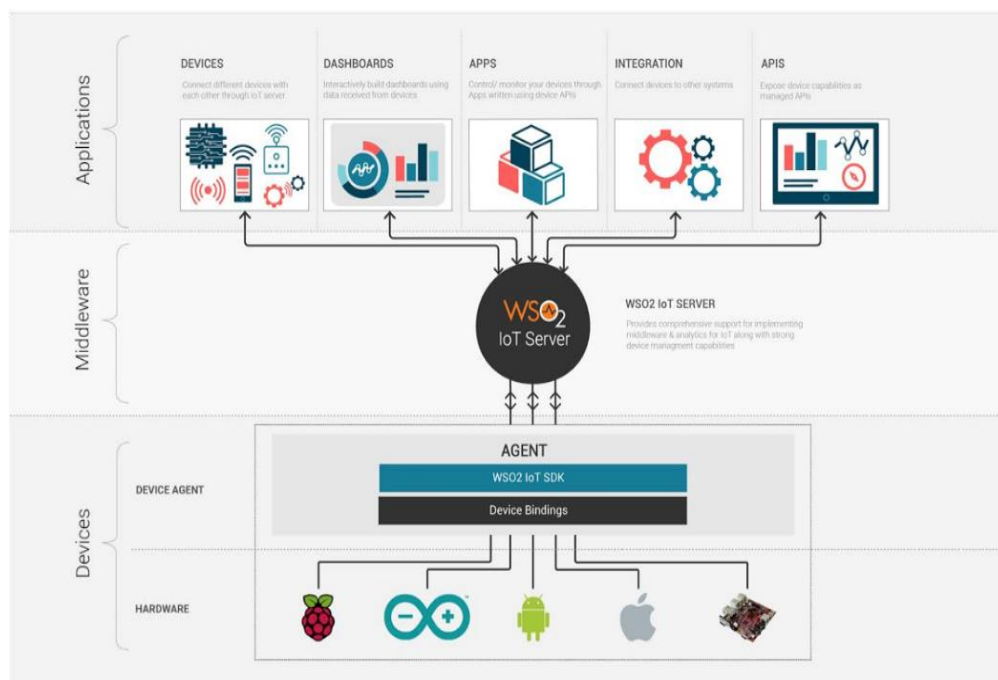
- Με την υπηρεσία ενημέρωσης WSO2, μπορούν να πραγματοποιούνται σταθερές ή περιοδικές βελτιώσεις στο προϊόντα και να εκτελείται μια συνεχής συντήρηση.
- Προσφέρει βελτιωμένα προφίλ προϊόντων προκειμένου να υλοποιούνται περισσότερο ευέλικτες και κλιμακούμενες εφαρμογές.
- Έχει την δυνατότητα να προσφέρει geofencing και διάφορες συναφείς ειδοποιήσεις, όλα σε πραγματικό χρόνο.
- Σχεδιάζει επίσης σενάρια IoT και Enterprise Mobility Manager (EMM) με προσαρμοσμένα χαρακτηριστικά.

Η αρχιτεκτονική του διακομιστή IoT μπορεί να αναλυθεί σε δύο κύρια τμήματα όπως απεικονίζεται στην Εικόνα 5.6.

Η **Πλατφόρμα διαχείρισης συσκευών (DM)** διαχειρίζεται τόσο τις φορητές συσκευές όσο και τις IoT συσκευές. Η διαχείριση των φορητών συσκευών γίνεται μέσω του Διαχειριστή Συσκευών Mobile Device Manager (MDM) WSO2, το οποίο επιτρέπει σε οργανισμούς να ασφαλίζουν, να διαχειρίζονται και να παρακολουθούν συσκευές Android, iOS και Windows (π.χ. smartphones, συσκευές iPod touch και tablet PC), ανεξαρτήτως του φορέα παροχής υπηρεσιών ή τον οργανισμό [33].

Ο **IoT Server** επικεντρώνεται κυρίως στη διαχείριση των συσκευών IoT, οι οποίες λειτουργούν στην κορυφή του WSO2 CDMF. Η πλατφόρμα υποστηρίζει διαφόρους τύπους συσκευών όπως το Android Sense, το Raspberry Pi, το Arduino Uno και πολλά άλλα [37].

Η πλατφόρμα WSO2 IoT συνεργάζεται με την πλατφόρμα διαχείρισης συσκευών WSO2 και την πλατφόρμα Middleware. Όχι μόνο οι συσκευές θα έχουν τη δυνατότητα εκτέλεσης διαφορετικών λειτουργιών, αλλά τα δεδομένα που συλλέγονται κατά την εκτέλεση αυτών των λειτουργιών θα αναλυθούν για να παρέχουν χρήσιμες πληροφορίες στον τελικό χρήστη χρησιμοποιώντας την πλατφόρμα Middleware WSO2.



Εικόνα 5.6: Αρχιτεκτονική πλατφόρμας WSO2

5.4.1 Integration - APIs

Μέσω της διεπαφής χρήστη η οποία παρέχεται από τον διακομιστή WSO2 IoT, μπορούν να εκτελεστούν διάφορες λειτουργίες σε μια IoT συσκευή, να ομαδοποιηθούν οι συσκευές, να εφαρμοσθούν πολιτικές ασφαλείας και πολλά άλλα. Οι ίδιες εργασίες μπορούν να επιτευχθούν χρησιμοποιώντας τα API του WSO2 IoT Server. Όλες οι δυνατότητες διαχείρισης των συσκευών προσφέρονται μέσω REST APIs με χρήση OAuth2 [24]. Τα API's είναι πλήρως ασφαλή. Όταν ένας χρυσωτής προσπαθήσει να πραγματοποιήσει μια οποιαδήποτε λειτουργία σε μια συσκευή, πραγματοποιούνται δύο επίπεδα εξουσιοδότησης:

- 1) Ελέγχει αν ο χρήστης είναι εξουσιοδοτημένος να έχει πρόσβαση στα API.
- 2) Ελέγχει εάν ο χρήστης είναι εξουσιοδοτημένος να πραγματοποιήσει την λειτουργία στην επιλεγμένη συσκευή.

5.4.2 Ασφάλεια

Κατά την χρήση του IoT WSO2 διακομιστή και την επικοινωνία του με τις διάφορες IoT συσκευές ανταλλάσσονται και αποθηκεύονται εμπιστευτικές πληροφορίες. Γι' αυτό τον λόγο η ασφάλεια κρίνεται απαραίτητη. Επιπλέον, η ασφάλεια κρίνεται απαραίτητη όταν ο διακομιστής WSO2 IoT ενσωματώνεται με άλλες εφαρμογές τρίτων παρόχων. Ο διακομιστής WSO2 IoT παρέχει τα ακόλουθα πρωτόκολλα ασφαλείας για την πιστοποίηση και την εξουσιοδότηση χρηστών [33] και συσκευών όπως απεικονίζονται στην Εικόνα 5.12:



Authentication & Authorization

Εικόνα 5.12 Πρωτόκολλα ασφαλείας για την πιστοποίηση και την εξουσιοδότηση χρηστών

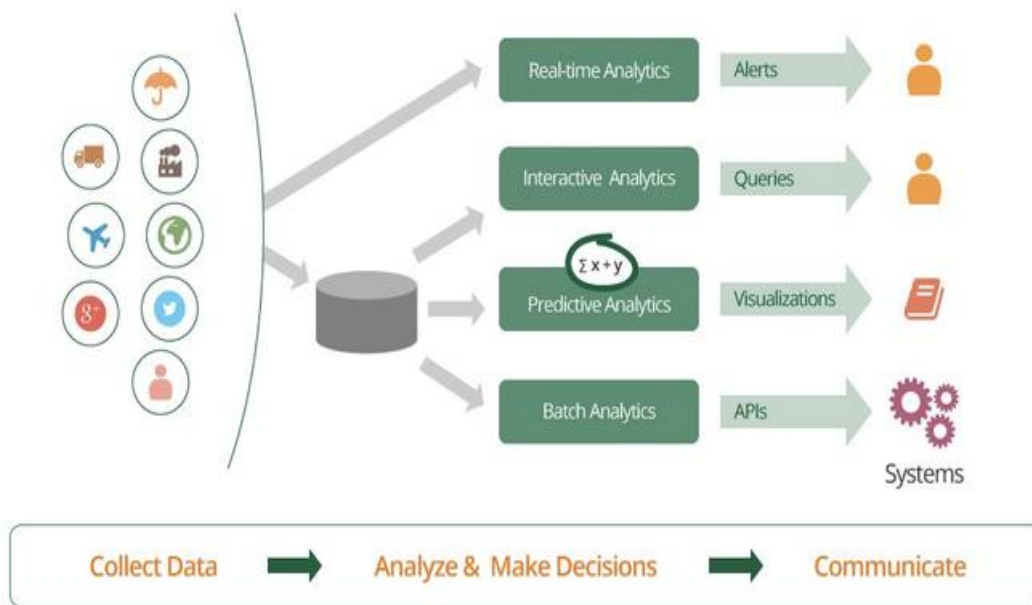
Ο διαχειριστής WSO2 API υποστηρίζει την ασφάλεια επιπέδου μεταφοράς (transport level). Μπορεί εύκολα να παραμετροποιηθεί για την ενεργοποίηση του TLS και την απενεργοποίηση και υποστήριξη του SSL. Αυτό είναι σημαντικό λόγω ενός σφάλματος (Poodle Attack) στο πρωτόκολλο SSL που θα μπορούσε να εκθέσει κρίσιμα δεδομένα κρυπτογραφημένα μεταξύ των υπολογιστών-πελατών και των διακομιστών. Σε σύγκριση με το SSL, το TLS χρησιμοποιεί βελτιωμένους κρυπτογραφικούς αλγόριθμους για να εξασφαλίσει την ασφάλεια της επικοινωνίας.

5.4.3 Πρωτόκολλα επικοινωνίας

Για την αμφίδρομη επικοινωνία μεταξύ των συσκευών και του IoT διακομιστή παρέχοντα διάφορα πρωτόκολλα επικοινωνίας. Για παράδειγμα, ο διακομιστής WSO2 IoT επικοινωνεί με τις συσκευές iOS μέσω του πρωτοκόλλου APNS (Apple Push Notification Service) και οι συσκευές επικοινωνούν με το διακομιστή χρησιμοποιώντας το HTTP πρωτόκολλο. Από προεπιλογή, ο διακομιστής WSO2 IoT χρησιμοποιεί πρωτόκολλα όπως το MQTT [24], XMPP, FCM και APNS.

5.4.4 Ανάλυση Δεδομένων

Η ανάλυση των δεδομένων που συλλέγονται από τις συσκευές είναι σημαντική για κάθε επιχείρηση. Ως εκ τούτου, ο διακομιστής WSO2 IoT συνδυάζει από προεπιλογή την ανάλυση δεδομένων σε πραγματικό χρόνο, μέσω διαδραστικής ανάλυσης καθώς και προγνώσεων (machine learning) σε μια ολοκληρωμένη πλατφόρμα με στόχο την υποστήριξη των πολλαπλών απαιτήσεων [40]. Χρησιμοποιώντας τον WSO2 IoT Server, μπορεί ο χρήστης να γράψει τις δικές του μεθόδους για την συλλογή των δεδομένων από τις συσκευές και να προχωρήσει στην ανάλυση των δεδομένων. Το WSO2 IoT Analytics Framework συνδυάζει την ανάλυση των δεδομένων σε πραγματικό χρόνο μέσω μιας ολοκληρωμένης πλατφόρμας όπως φαίνεται και στην Εικόνα 5.13.

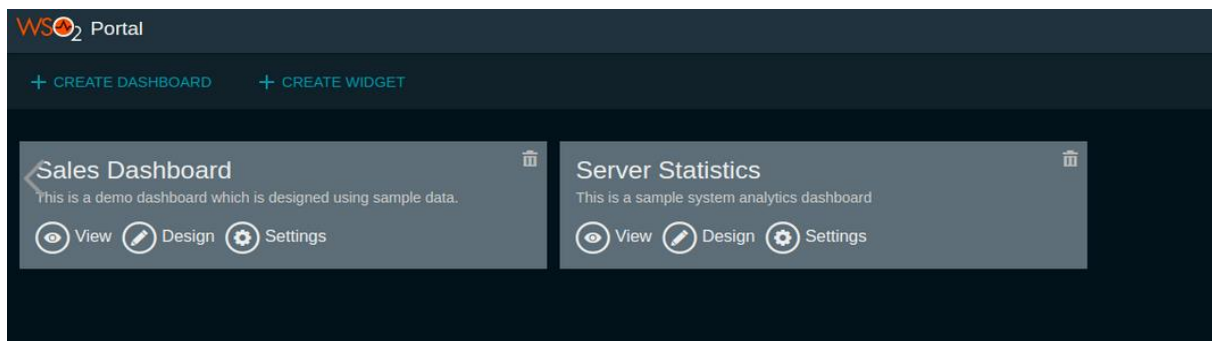


Εικόνα 5.13: Ανάλυση δεδομένων μέσω του WSO2 IoT Analytics Framework

Ως μέρος της πλατφόρμας ανάλυσης WSO2 IoT Server, ο WSO2 Data Analytics Server (WSO2 DAS) και ο WSO2 Complex Event Processor (WSO2 CEP) εισάγουν μια ενιαία λύση με την ικανότητα δημιουργίας συστημάτων και εφαρμογών που συλλέγουν και αναλύουν δεδομένα τόσο σε πραγματικό χρόνο όσο και σε προγενέστερο [40].

5.4.5 Απεικόνιση Δεδομένων

Μέσω του Dashboard, Εικόνα 5.14 που παρέχεται από την πλατφόρμα, μπορούν να απεικονισθούν τα αποτελέσματα των αναλύσεων που πραγματοποιήθηκαν μέσα από το εργαλείο WSO2 SP (WSO2 Stream Processor) σε διάφορες γραφικές παραστάσεις.

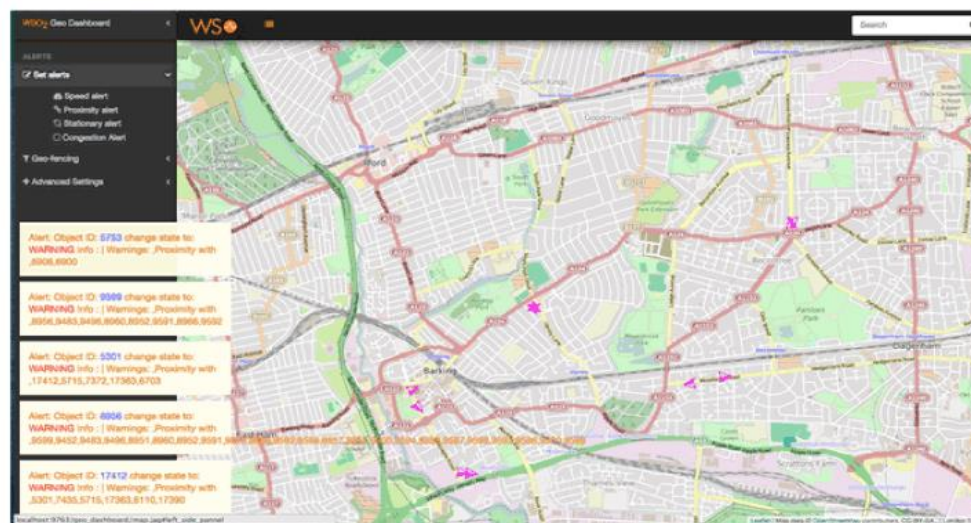


Εικόνα 5.14: Εργαλείο Απεικόνισης Δεδομένων

Μπορούν να εκτελεσθούν οι παρακάτω ενέργειες όπως [33]:

- Σχεδίαση πινάκων ελέγχου (Dashboards)
- Δημιουργία γραφικών στοιχείων (widgets)
- Δημιουργία προσαρμοσμένων γραφικών στοιχείων
- Έλεγχος πρόσβασης σε συγκεκριμένα dashboards
- Διαχείριση γραφικών στοιχείων (widgets)

Επίσης, μέσω της πλατφόρμας ανάλυσης WSO2 παρέχονται δυνατότητες για την απεικόνιση των αναλυμένων πληροφοριών με πολλούς τρόπους. Τα αποτελέσματα μπορούν να δημοσιευτούν σε πίνακες εργαλείων (παραδείγματα που φαίνονται στην Εικόνα 5.15) που περιέχουν διάφορους τύπους μικροεφαρμογών (gadgets).



Εικόνα 5.15: Απεικόνιση δεδομένων μέσω πινάκων

5.4.6 Hosting

Η πλατφόρμα WSO2 μπορεί να εγκατασταθεί ως εξής [37]:

- Σε διακομιστές Linux, Windows και Solaris
- Ανάπτυξη σε cloud υποδομές συμπεριλαμβανομένων των Amazon EC2, της Microsoft Azure και του Google Compute Engine
- Υβριδική ή ιδιωτική ανάπτυξη σε cloud πλατφόρμες όπως OpenStack, Suse Cloud, Eucalyptus, Amazon Virtual Private Cloud και Apache Stratos.

5.4.7 Push Notifications

Οι ειδοποιήσεις push χρησιμοποιούνται για την αποστολή μηνυμάτων σε συσκευές. Εάν έχει προηγηθεί η κατάλληλη ρύθμιση του διακομιστή WSO2 IoT, μπορεί να προωθήσει (push) τις ειδοποιήσεις στις συσκευές μέσω του Firebase Cloud Messaging (FCM) ή του Apple Push Notification (APNS), η ειδοποίηση αποστέλλεται στις συσκευές [40]. Όταν πολλές συσκευές ταυτόχρονα προσπαθήσουν να επικοινωνήσουν με το διακομιστή, υπάρχει η δυνατότητα ο διακομιστής να τεθεί εκτός λειτουργίας. Για να αποφευχθεί αυτή η κατάσταση, ο διακομιστής WSO2 IoT μπορεί να διαχειριστεί έναν συγκεκριμένο αριθμό συσκευών που επικοινωνούν με το διακομιστή σε μία δεδομένη χρονική στιγμή. Η αποστολή των ειδοποιήσεων γίνεται σε καθορισμένα χρονικά διαστήματα. Με αυτόν τον τρόπο επιτυγχάνεται, ότι σε μία δεδομένη στιγμή μόνο ένας καθορισμένος αριθμός συσκευών θα μπορεί να επικοινωνεί με τον διακομιστή.

5.4.8 Ενεργοποίηση συμβάντων (Trigger)

Η πλατφόρμα WSO2 προσφέρει την δυνατότητα εκτέλεσης συμβάντων σε συνάρτηση των πληροφοριών που έχουν συλλεχθεί. Χρησιμοποιώντας εργαλεία όπως το WSO2 CEP, το WSO2 BAM και το WSO2 Business Process Server μπορούν να επιτευχθούν τα εξής [33]:

- Ενεργοποίηση σύνθετων ενεργειών
- Προειδοποιήσεις
- Παρακολούθηση της προόδου των ενεργειών
- Κατανόηση των αποτελεσμάτων των ενεργειών και των παραγόντων επιτυχίας.

5.4.9 Αποθήκευση Δεδομένων

Η WSO2 IoT πλατφόρμα παρέχει την δυνατότητα για την δημιουργία scripts, μέσω των οποίων επιτυγχάνεται η εγκατάσταση και η διαμόρφωση διαφόρων βάσεων δεδομένων όπως MySQL, H2, Oracle, MSSQL και PostgreSQL [37]. Κάθε μία από αυτές τις βάσεις δεδομένων υποστηρίζει διαδικασίες αποθήκευσης, οι οποίες επιτρέπουν την ενσωμάτωση των δεδομένων μιας επιχείρησης μέσα στη βάση δεδομένων μέσω API, παρέχοντας έναν ισχυρό μηχανισμό αλληλεπίδρασης με μια σχεσιακή βάση δεδομένων. Επειδή αυτές οι διαδικασίες αποθηκεύονται σε μια προκατασκευασμένη μορφή μέσα στην ίδια τη βάση δεδομένων, η ταχύτητα εκτέλεσης είναι πολύ γρήγορη. Η προεπιλεγμένη βάση δεδομένων είναι η H2. Επίσης μπορούν να εγκατασταθούν και να παραμετροποιηθούν βάσεις δεδομένων από άλλους προμηθευτές, όπως MySQL, Oracle, PostgreSQL και Microsoft SQL Servers, χρησιμοποιώντας τα scripts που παρέχονται από το WSO2 για την εγκατάσταση και τη διαμόρφωση των σχεσιακών βάσεων δεδομένων.

5.4.10 Αποθήκευση Blob

Μέσω του διακομιστή αποθήκευσης WSO2 Storage server [37] προσφέρεται η δομημένη και μη δομημένη αποθήκευση δεδομένων.

5.4.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)

Τα κιτ ανάπτυξης λογισμικού (SDK) περιέχουν τα απαραίτητα εργαλεία για τη δημιουργία μιας εφαρμογής που χρησιμοποιεί το λειτουργικό σύστημα για να καλέσει το τοπικό API [33]. Από προεπιλογή, η δημιουργία SDK είναι ενεργοποιημένη για Java και Android.

5.4.12 Geolocation

Οι υπηρεσίες βάσει τοποθεσίας (LBS) χρησιμοποιούν γεωγραφικές τοποθεσίες σε πραγματικό χρόνο ή γεωχωρικά δεδομένα με στόχο την παρακολούθηση των IoT συσκευών [37]. Η επέκταση geo στο WSO2 IoT Server παρέχει πληροφορίες σε πραγματικό χρόνο σχετικά με τα γεωχωρικά αντικείμενα. Επεξεργάζεται χωρικά δεδομένα από μια εξωτερική πηγή συμβάντων και αναλύει / χειρίζεται αυτά τα δεδομένα για να παράγει σημαντικές πληροφορίες στους τελικούς χρήστες. Μέσω αυτής της αλληλεπίδρασης μπορεί να δημιουργηθεί μια ποικιλία ειδοποιήσεων και προειδοποιήσεων.

5.4.13 Πλάνα Τιμολόγησης

Το κόστος των υπηρεσιών της πλατφόρμας δεν διατίθεται στην επίσημη ιστοσελίδα.

5.4.14 Υποστηριζόμενες γλώσσες προγραμματισμού

Στην επίσημη ιστοσελίδα της πλατφόρμας γίνεται αναφορά σε μία νέα γλώσσα προγραμματισμού με την ονομασία Ballerina [37]. Πρόκειται για μία γλώσσα η οποία διευκολύνει την διασύνδεση εφαρμογών και υπηρεσιών σε όλα τα είδη σεναρίων ενσωμάτωσης.

5.4.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων

Μέσω APIs επιτυγχάνεται η διασύνδεση με πλατφόρμες τρίτων παρόχων όπως οι SAP, Salesforce, Twitter και Paypal

5.4.16 Ενσωμάτωση με Hardware πλατφόρμες

Ο διακομιστής IoT WSo2 είναι χτισμένος (built on) πάνω στο WSO2 CDMF framework το οποίο παρέχει τη δυνατότητα διασύνδεσης με ένα μεγάλο εύρος συσκευών υλικού, όπως κινητές συσκευές με λειτουργικό

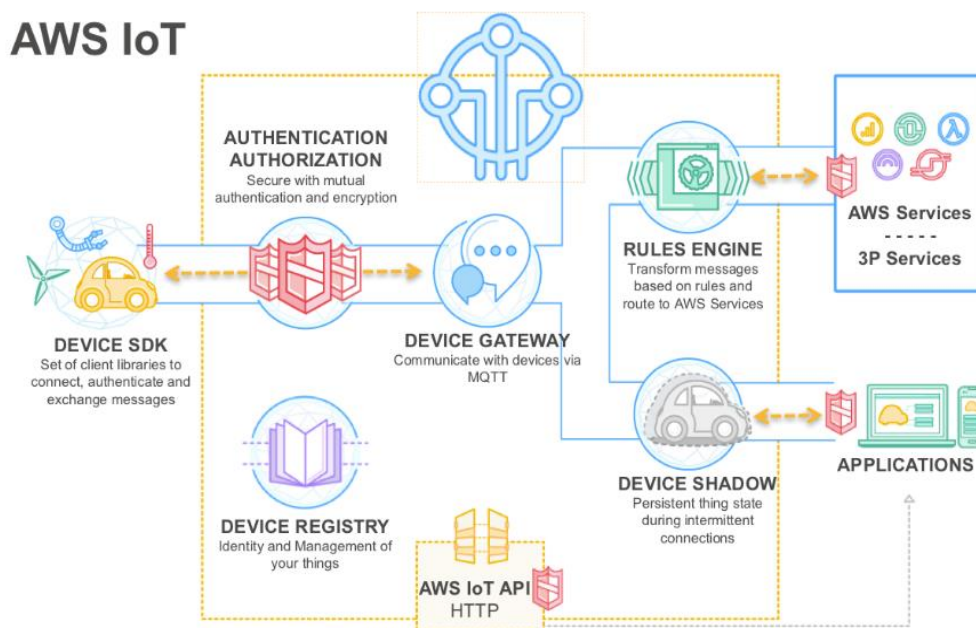
Android, Microsoft Windows ή iOS, μικροελεγκτές όπως Raspberry Pi και Arduino καθώς επίσης και την δυνατότητα με Drones όπως για παράδειγμα το 3D Robotics IRIS [37].

5.4.17 Συμπεράσματα

Η WSO2 IoT πλατφόρμα προσφέρει μια ολοκληρωμένη, ασφαλή και ανοικτού κώδικα (open-source) λύση για τη διαχείριση IoT συσκευών. Η αρθρωτή (modular) και προσαρμόσιμη (customizable) αρχιτεκτονική από την πλευρά του διακομιστή (server) καθιστούν την πλατφόρμα μια ιδανική λύση για την ανάπτυξη και τη διαχείριση IoT εφαρμογών [33]. Ο διακομιστής WSO2 βασίζεται σε μία open – source πλατφόρμα η οποία μπορεί να επεκταθεί για να υποστηρίξει οποιοδήποτε τύπο συσκευής, πρωτόκολλα και μορφές δεδομένων. Ως αποτέλεσμα, οι χρήστες μπορούν να ενσωματώσουν στο διακομιστή τις δικές τους IoT συσκευές.

5.5 AWS IoT platform

Το AWS IoT (Εικόνα 5.16) είναι μια cloud πλατφόρμα διαχείρισης IoT υποδομών, η οποία αποτελεί μέρος μιας μεγάλης υπηρεσίας cloud computing που προσφέρεται από την American Amazon.com, Inc. Η Amazon είναι ένας από τους μεγαλύτερους παρόχους υπηρεσιών cloud computing στη σημερινή αγορά και πολλές μεγάλες εταιρείες βασίζονται στις υπηρεσίες που προσφέρει. Οι cloud υπηρεσίες του Amazon είναι διαθέσιμες αυτήν τη στιγμή σε 12 γεωγραφικές περιοχές σε όλο τον κόσμο, που τροφοδοτούνται από τελενταίας τεχνολογίας κέντρα δεδομένων (datacenters).



Εικόνα 5.16: Πλατφόρμα AWS IoT

Ακολουθεί μια επεξήγηση των βασικών στοιχείων της πλατφόρμας AWS IoT:

- **Things** - Τα πράγματα είναι εφαρμογές, συνδεδεμένες συσκευές ή φυσικά αντικείμενα που υποστηρίζονται από μια cloud εφαρμογή του Amazon.
- **Thing Registry** - Επίσης αναφέρεται ως μητρώο συσκευών. Οργανώνει τα χαρακτηριστικά και τις ιδιότητες των καταλόγων που σχετίζονται με το κάθε πράγμα. Κάθε αντικείμενο μπορεί να συσχετιστεί με μέχρι τρία προσαρμοσμένα γνωρίσματα (attributes).
- **Thing shadow** - Επίσης αναφέρεται ως σκιά συσκευής. "Οι Σκιάσεις Συσκευών είναι μια εικονική έκδοση (JSON) για κάθε πράγμα - αντικείμενο που περιλαμβάνει την τελευταία αναφερθείσα κατάσταση του αντικειμένου και την επιθυμητή μελλοντική κατάσταση, ακόμα και όταν το συγκριμένο πράγμα δεν είναι συνδεδεμένο." [38].
- **Message broker** - Είναι μια υπηρεσία δημοσίευσης / εγγραφής (publish / subscribe) η οποία επιτρέπει την αποστολή και λήψη μηνυμάτων από και προς το AWS IoT. Είναι υπεύθυνη για την

εξακρίβωση της ταυτότητας των συσκευών, την ασφαλή λήψη των δεδομένων από την συσκευή καθώς και την τήρηση των δικαιωμάτων πρόσβασης που έχουν τεθεί σε συσκευές που χρησιμοποιούν πολιτικές ασφαλείας.

Σε αντίθεση με άλλες πλατφόρμες που αναλύθηκαν, η πλατφόρμα AWS IoT χρησιμεύει αποκλειστικά ως πλατφόρμα για αμφίδρομη επικοινωνία μεταξύ των αντικειμένων και του AWS Cloud. Για την αποθήκευση δεδομένων, απεικονίσεις, την επεξεργασία δεδομένων κλπ., χρησιμοποιούνται άλλες υπηρεσίες AWS. Στις περισσότερες περιπτώσεις η πλατφόρμα δεν χρησιμοποιείται ανεξάρτητα και λειτουργεί ως πύλη (gateway) για τις συσκευές που επικοινωνούν με άλλες υπηρεσίες AWS. Το AWS IoT περιέχει “Rule Engine”, το οποίο προσφέρει τη δυνατότητα αλληλεπίδρασης με αυτές τις υπηρεσίες. Οι κανόνες (rules) που αξιολογούν τα εισερχόμενα μηνύματα γράφονται σε μια γλώσσα που βασίζεται σε SQL, η οποία δίνει στον χρήστη τη δυνατότητα να προσαρμόσει πλήρως αυτούς τους κανόνες. Υπηρεσίες AWS που είναι άμεσα ενσωματωμένες στην πλατφόρμα AWS IoT μέσω του Rule Engine είναι [19] :

- Το Amazon CloudWatch, για τη λήψη μετρήσεων, την ενημέρωση αλλαγής καταστάσεων και καταγραφή.
- Amazon DynamoDB, για την εγγραφή δεδομένων σε διαχειριζόμενες βάσεις δεδομένων NoSQL.
- Amazon Elastic search, για αναζήτηση δεδομένων, αναλύσεις και απεικονίσεις (μέσω του ενσωματωμένου plugin open source Kibana).
- Amazon Kinesis, για την επεξεργασία δεδομένων ροής σε πραγματικό χρόνο.
- Amazon Simple Storage Service (S3), για την αποθήκευση δεδομένων.
- Amazon Simple Notification Service (SNS), για την αποστολή ειδοποιήσεων.
- Amazon Lambda, για την εκτέλεση προσαρμοσμένου κώδικα σε Java, Node.js ή Python στους εικονικούς διακομιστές του Amazon.

5.5.1 Integration - API

Το πλεονέκτημα της χρήσης μιας σκιάς συσκευής (device shadow) σε εφαρμογές IoT είναι ότι αυτές οι εφαρμογές μπορούν να παρέχουν πάντα διαθέσιμο API για αλληλεπίδραση με τις συσκευές, ακόμη και όταν οι συσκευές δεν είναι διαθέσιμες. Επιπλέον, οι εφαρμογές μπορούν να ορίσουν την επιθυμητή μελλοντική κατάσταση μιας συσκευής χωρίς να υπολογίζουν την τρέχουσα κατάσταση της συσκευής, συγκρίνοντας τη διαφορά μεταξύ της επιθυμητής και της τελευταίας αναφερθείσας κατάστασης [40]. Το AWS IoT υποστηρίζει δύο τρόπους ενημέρωσης μια σκιάς συσκευής - με RESTful API μέσω πρωτοκόλλου HTTP και με MQTT publish / subscribe [19] μηνύματα έγγραφα για τις λειτουργίες UPDATE, GET και DELETE ονομάζονται Έγγραφα αιτημάτων και έχουν τη μορφή αρχείου JSON.

5.5.2 Ασφάλεια

Κάθε συνδεδεμένη συσκευή χρειάζεται πιστοποίηση για να μπορέσει να αποστείλει και να λάβει δεδομένα, δηλαδή να έρθει σε επικοινωνία με τον message broker. Αφού τα δεδομένα φτάσουν στον μεσίτη μηνυμάτων, οι μηχανισμοί ασφάλειας του cloud της AWS, όπως το AWS Identity and Access Management (IAM), προστατεύουν τα δεδομένα καθώς μετακινούνται μεταξύ του AWS IoT και άλλων συσκευών ή υπηρεσιών AWS. Όλη η κίνηση προς και από το AWS IoT πρέπει να κρυπτογραφείται μέσω του Transport Layer Security (TLS) [19]. Το TLS χρησιμοποιείται για την εξασφάλιση της εμπιστευτικότητας των πρωτοκόλλων MQTT και HTTP που υποστηρίζονται από το AWS IoT. Το AWS IoT υποστηρίζει αρκετές σουίτες κρυπτογράφησης TLS.

Το AWS IoT υποστηρίζει τρεις τύπους για τον έλεγχο ταυτότητας. Όταν χρησιμοποιείτε το πρωτόκολλο HTTP, γίνεται χρήση ταυτοτήτων μέσω του IAM (χρήστες, ομάδες, ρόλοι) και του Amazon Cognito. Όταν χρησιμοποιείτε το MQTT, γίνεται χρήση X.509 πιστοποιητικών [20]. Οι χρήστες, οι ομάδες και οι ρόλοι του IAM αποτελούν τον βασικό μηχανισμό διαχείρισης της ταυτότητας και της πιστοποίησης ταυτότητας στο AWS. Η ταυτότητα του Amazon Cognito επιτρέπει την διαχείριση και την πιστοποίηση της ταυτότητας μέσω άλλων δημοφιλών παρόχων όπως η σύνδεση με Amazon, Facebook ή Google.

5.5.3 Πρωτόκολλα Επικοινωνίας

Μέσω της AWS IoT πλατφόρμας επιτρέπεται στις συσκευές οι οποίες έχουν πρόσβαση στο Διαδίκτυο να συνδεθούν στο Cloud AWS με αποτέλεσμα να μπορέσουν οι διάφορες AWS εφαρμογές να αλληλοεπιδρούν με τις συσκευές που έχουν συνδεθεί. Οι κοινές εφαρμογές του IoT οικοσυστήματος της Amazon μπορούν να συλλέγουν και να επεξεργάζονται δεδομένα από τις διάφορες συσκευές ή να παρέχουν στους χρήστες να ελέγχουν από απόσταση μια συσκευή. Οι συσκευές αναφέρουν την κατάστασή τους δημοσιεύοντας μηνύματα, σε μορφή JSON, σε θέματα (topics) MQTT. Κάθε θέμα (topic) MQTT έχει ένα ιεραρχικό όνομα που προσδιορίζει τη συσκευή της οποίας η κατάσταση θα πρέπει να ενημερώνεται.

Στο MQTT, η λέξη "θέμα" αναφέρεται σε μια συμβολοσειρά UTF-8 που χρησιμοποιεί ο μεσίτης (broker) για να φιλτράρει μηνύματα για κάθε συνδεδεμένη συσκευή. Το θέμα αποτελείται από ένα ή περισσότερα επίπεδα θεμάτων. Κάθε επίπεδο θέματος διαχωρίζεται με μια προς τα εμπρός κάθετο (διαχωριστικό επίπεδο θέματος) όπως απεικονίζεται στην Εικόνα 5.7.



Εικόνα 5.7: Διαχωριστικό επίπεδο θέματος

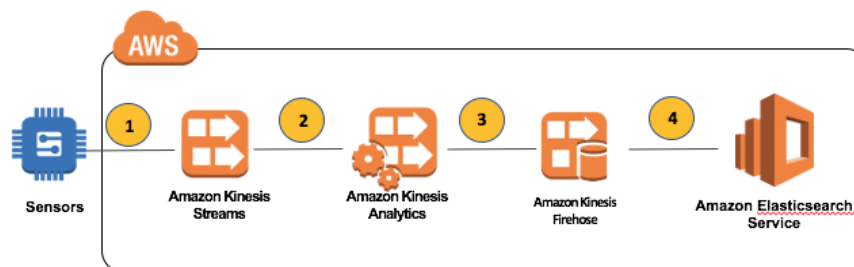
Όταν δημοσιεύεται ένα μήνυμα σε ένα θέμα MQTT, το μήνυμα αποστέλλεται στον μεσίτη μηνυμάτων AWS IoT MQTT, ο οποίος είναι υπεύθυνος για την αποστολή όλων των μηνυμάτων που δημοσιεύονται σε ένα θέμα MQTT σε όλους τους πελάτες που έχουν εγγραφεί στο συγκεκριμένο θέμα. Ο μεσίτης μηνυμάτων υποστηρίζει τη χρήση του πρωτοκόλλου MQTT για δημοσίευση και εγγραφή (publish - subscribe) και το πρωτόκολλο HTTPS μόνο για δημοσίευση (publish). Και τα δύο πρωτόκολλα υποστηρίζονται μέσω IP έκδοσης 4 (IPv4) και IP έκδοσης 6 (IPv6). Επίσης, υποστηρίζεται και MQTT μέσω του πρωτοκόλλου WebSocket.

5.5.4 Ανάλυση Δεδομένων

Το Amazon Kinesis Data Analytics καθιστά τον ευκολότερο τρόπο για την ανάλυση των δεδομένων ροής, την απόκτηση πληροφοριών και κατά συνέπεια την ανταπόκριση στις ανάγκες των χρηστών σε πραγματικό χρόνο. Το Amazon Kinesis Data Analytics [31] μειώνει την πολυπλοκότητα της δημιουργίας, διαχείρισης και ενσωμάτωσης εφαρμογών συνεχούς ροής με άλλες υπηρεσίες AWS. Οι χρήστες SQL μπορούν εύκολα να αναζητήσουν δεδομένα ροής ή να δημιουργήσουν ολόκληρες εφαρμογές συνεχούς ροής χρησιμοποιώντας προσαρμοσμένα στις ανάγκες τους πρότυπα. Οι προγραμματιστές της Java μπορούν να δημιουργήσουν γρήγορα εξελιγμένες εφαρμογές συνεχούς ροής, χρησιμοποιώντας βιβλιοθήκες Java Open Source και ενσωματώσεις AWS για τη μετατροπή και την ανάλυση δεδομένων σε πραγματικό χρόνο.

Ακόμη ένα ενδιαφέρον χαρακτηριστικό αποτελεί η υπηρεσία Amazon Elasticsearch μέσω της οποίας επιτυγχάνεται η επεξεργασία και η ανάλυση των δεδομένων καταγραφής (logs) που έχουν συλλεχθεί από τις συνδεδεμένες συσκευές. Η Amazon Elasticsearch Service είναι μια πλήρως διαχειριζόμενη υπηρεσία που παρέχει APIs και δυνατότητες ανάλυσης σε πραγματικό χρόνο. Παρέχεται ανάλυση μη δομημένων (unstructured) και ημιδομημένων (semi-structured) αρχείων καταγραφής (logs) τα οποία παράγονται από τις συνδεδεμένες IoT συσκευές. Η υπηρεσία Amazon Elastic υπηρεσία αφού αρχικά κατηγοριοποιήσει τα δεδομένα τα καθιστά διαθέσιμα για ανάλυση σε σχεδόν πραγματικό χρόνο (λιγότερο από ένα δευτερόλεπτο).

Εν συνεχεία μπορεί να γίνει χρήση της ενσωματωμένη υπηρεσίας **Kibana** για την απεικόνιση των δεδομένων και την εκτέλεση κρίσιμων αναλύσεων όπως την αναγνώριση διακοπής (outages) των υπηρεσιών και άλλων προβλημάτων. Μέσω της γεωχωρικής ανάλυσης (geospatial analysis) μπορεί να γίνει εντοπισμός της γεωγραφικής περιοχής όπου συνέβη το πρόβλημα. Στην Εικόνα 5.8 απεικονίζεται μια επισκόπηση της διασύνδεσης των υπηρεσιών που παρέχονται από την πλατφόρμα σχετικά με την ανάλυση των δεδομένων.



Εικόνα 5.8: Διαδικασία Ανάλυσης Δεδομένων

5.5.5 Απεικόνιση Δεδομένων

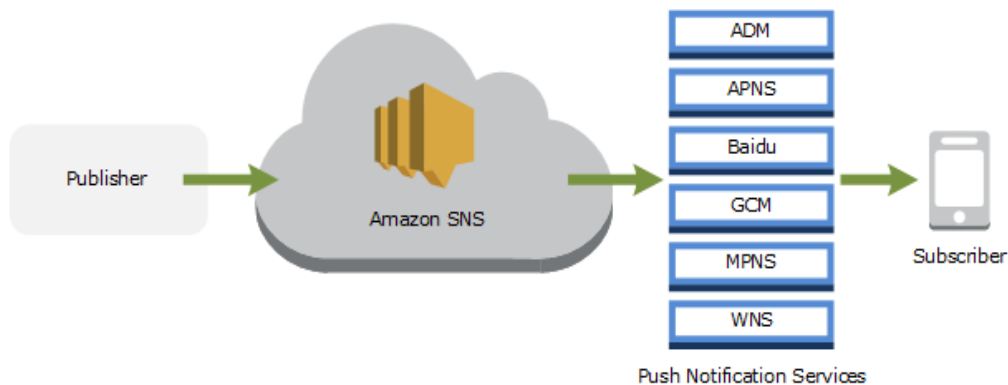
Οι απεικονίσεις των δεδομένων που λαμβάνονται μέσω των Ιοτ αντικειμένων δεν ενσωματώνονται στην ίδια την πλατφόρμα. Το Amazon ακολουθεί μια διαφορετική προσέγγιση μέσω του Amazon Elasticsearch Service, επιτρέποντας την παρακολούθηση σε πραγματικό χρόνο, την αναζήτηση και την απεικόνιση οποιωνδήποτε δεδομένων σε κεντρικό επίπεδο, όχι μόνο από το AWS IoT [19]. Η Amazon Elasticsearch Service είναι μια υπηρεσία AWS που παρέχει την μηχανή αναζήτησης ανοιχτού κώδικα Elasticsearch στο AWS Cloud. Ενσωματώνει την πλατφόρμα απεικόνισης ανοιχτού κώδικα Kibana (έκδοση 3 και 4). Το Kibana προσφέρει διάφορους τρόπους για την εμφάνιση και την ανάλυση δεδομένων σε πραγματικό χρόνο. Για παράδειγμα, μπορεί να απεικονίσει δεδομένα σε πίνακες ή γραμμές καθώς επίσης μπορεί να εμφανίσει γεωγραφικά δεδομένα. Αυτά τα διαγράμματα και widgets μπορούν να ταξινομηθούν σε πίνακες ελέγχου. Επιπλέον, όλα τα δεδομένα μπορούν να εξαχθούν σε μορφή CSV.

5.5.6 Hosting

Η ανάπτυξη και η διαχείριση της συγκεκριμένης πλατφόρμας πραγματοποιείται μόνο από την cloud υποδομή που παρέχεται από την Amazon.

5.5.7 Push Notifications

Η υπηρεσία Amazon Simple Notification Service (Amazon SNS) είναι μια υπηρεσία ιστού η οποία επιτρέπει στις εφαρμογές, στους τελικούς χρήστες και τις Ιοτ συσκευές να στέλνουν και να λαμβάνουν ειδοποιήσεις από και προς την πλατφόρμα η οποία είναι εγκατεστημένη στην cloud υποδομή (Εικόνα 5.10).



Εικόνα 5.10 : Υπηρεσία Amazon Simple Notification Service (Amazon SNS)

Στην υπηρεσία Amazon SNS, υπάρχουν δύο τύποι πελατών (clients), οι εκδότες (publishers) και οι συνδρομητές (subscribers). Οι εκδότες επικοινωνούν ασύγχρονα με τους συνδρομητές δημιουργώντας και στέλλοντας ένα μήνυμα σε ένα θέμα (topic), το οποίο είναι ένα λογικό σημείο πρόσβασης και ένα κανάλι επικοινωνίας. Οι συνδρομητές (π.χ. διακομιστές ιστού, διευθύνσεις ηλεκτρονικού ταχυδρομείου, ουρές Amazon SQS, λειτουργίες AWS Lambda) λαμβάνουν το μήνυμα ή την ειδοποίηση μέσω ενός από τα υποστηριζόμενα πρωτόκολλα (Amazon SQS, HTTP / S, email, SMS, Lambda) όταν έχουν εγγραφεί στο συγκεκριμένο θέμα (topic).

5.5.8 Ενεργοποίηση συμβάντων (Trigger Actions)

Η AWS IoT πλατφόρμα χρησιμοποιεί κανόνες (rules) βάσει των οποίων ενεργοποιούνται διάφορα συμβάντα. Ένα συμβάν ή μία ενέργεια μπορεί να ισοδυναμεί με την εγγραφή δεδομένων σε μία βάση δεδομένων DynamoDB ή να επικαλεστεί μία ενέργεια Lambda [34]. Οι ενέργειες οι οποίες υποστηρίζονται είναι οι ακόλουθες:

- **cloudwatchMetric** to capture a CloudWatch metric.
- **dynamoDB** to write data to a DynamoDB database.
- **dynamoDBv2** to write data to a DynamoDB database.
- **elasticsearch** to write data to an Amazon Elasticsearch Service domain.
- **firehose** to write data to an Amazon Kinesis Data Firehose stream.
- **iotAnalytics** to send data to an AWS IoT Analytics channel.
- **iotEvents** to send data to an AWS IoT Events input.
- **kinesis** to write data to a Kinesis stream.
- **lambda** to invoke a Lambda function.

- **republish** to republish the message on another MQTT topic.
- **s3** to write data to an Amazon S3 bucket.
- **salesforce** to write a message to a Salesforce IoT input stream.
- **sns** to write data as a push notification.
- **sqs** to write data to an SQS queue.
- **stepFunctions** to start execution of a Step Functions state machine.

Ο μηχανισμός κανόνων (Rule Engine) της AWS IoT πλατφόρμας μπορεί να κάνει πολλαπλές προσπάθειες για να εκτελέσει μια ενέργεια σε περίπτωση σφάλματος. Εάν όλες οι προσπάθειες αποτύχουν, το μήνυμα απορρίπτεται και το σφάλμα είναι διαθέσιμο στα αρχεία καταγραφής (logs) της πλατφόρμας CloudWatch. Επίσης, μπορεί ο χρήστης να ορίσει μια ενέργεια σφάλματος για κάθε κανόνα που γίνεται επίκληση μετά από μια αποτυχία.

5.5.9 Αποθήκευση Δεδομένων

Η αποθήκευση των δεδομένων επιτυγχάνεται μέσω του Amazon DynamoDB. Πρόκειται για μία υπηρεσία βάσεων δεδομένων NoSQL.

5.5.10 Αποθήκευση Blob

Μέσω της υπηρεσίας Amazon S3 (Amazon Simple Storage Service) [34] επιτυγχάνεται αποθήκευση και ανάκτηση οποιασδήποτε ποσότητας και μορφής δεδομένων οποιαδήποτε στιγμή, από οπουδήποτε στο διαδίκτυο. Μέσω του Amazon S3 τα δεδομένα αποθηκεύονται ως αντικείμενα (objects) μέσα σε "κουβάδες" (buckets) όπως φαίνεται και στην Εικόνα 5.9



Εικόνα 5.9: Διαδικασία αποθήκευσης δεδομένων ως αντικείμενα

Ένα αντικείμενο αποτελείται από ένα αρχείο και προαιρετικά από οποιοδήποτε μεταδεδομένα (metadata) που περιγράφουν αυτό το αρχείο.

5.5.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)

Τα Κιτ ανάπτυξης λογισμικού που προσφέρει η AWS IoT πλατφόρμα, περιλαμβάνουν βιβλιοθήκες ανοιχτού κώδικα, οδηγούς προγραμματιστών με δείγματα και οδηγούς μεταφόρτωσης, ώστε να μπορούν

οι χρήστες να δημιουργούν καινοτόμα προϊόντα ή λύσεις IoT με την δυνατότητα της επιλογής των πλατφορμών υλικού.

5.5.12 Geolocation

Μέσα από τη βιβλιογραφία καθώς και την επίσημη ιστοσελίδα της πλατφόρμας δεν είναι ξεκάθαρος ο τρόπος κατά τον οποίο πραγματοποιείται ο εντοπισμός θέσης των IoT συσκευών που έχουν διασυνδεθεί με την πλατφόρμα.

5.5.13 Πλάνα Τιμολόγησης

Η χρέωση των υπηρεσιών της AWS πλατφόρμας, βασίζεται στο μοντέλο Pay-as-you-go. Μέσω του συγκεκριμένου μοντέλου επιτυγχάνεται η εύκολη προσαρμογή στις μεταβαλλόμενες επιχειρηματικές ανάγκες χωρίς να δημιουργείται υπερβολική χρέωση. Οι υπηρεσίες AWS πληρώνονται με βάση την πραγματική χρήση των πόρων και κάθε υπηρεσία έχει διαφορετικούς όρους τιμολόγησης. Οι τιμές του AWS IoT βασίζονται στον αριθμό των μηνυμάτων που έχουν δημοσιευτεί και παραδοθεί. Το Amazon προσφέρει μια δωρεάν συνδρομή που λήγει μετά από 12 μήνες. Οι δυνατότητες της δωρεάν συνδρομής ποικίλλουν από υπηρεσία σε υπηρεσία, ενώ το AWS IoT προσφέρει 250.000 δωρεάν μηνύματα ανά μήνα [16]. Ωστόσο, υπάρχει η απαίτηση, ο χρήστης να εισαγάγει πληροφορίες της πιστωτικής κάρτας του κατά τη διάρκεια μιας εγγραφής.

5.5.14 Υποστηριζόμενες γλώσσες προγραμματισμού

Οι γλώσσες προγραμματισμού οι οποίες υποστηρίζονται από την πλατφόρμα είναι οι εξής:

- Python 3.7
- Node v8.10.0
- Java
- C
- C++
- Οποιαδήποτε γλώσσα υποστηρίζει την εισαγωγή βιβλιοθηκών C

5.5.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων

Μέσα από τη βιβλιογραφία καθώς και την επίσημη ιστοσελίδα της πλατφόρμας δεν είναι ξεκάθαρος ο τρόπος κατά τον οποίο πραγματοποιείται η ενσωμάτωση της πλατφόρμας με υπηρεσίες τρίτων παρόχων.

5.5.16 Ενσωμάτωση Hardware πλατφορμών

Μέσω της χρήσης του AWS IoT SDK παρέχεται η δυνατότητα για την διασύνδεση διαφόρων συσκευών υλικού (<https://devices.amazonaws.com/search?page=2&sv=iot>) με την πλατφόρμα [34]. Έχει αναπτυχθεί σε γλώσσα C για συσκευές με περιορισμένους πόρους ή συστήματα σε chip (SoC) με διαθέσιμη μνήμη 256 KB ή περισσότερο. Το SDK απλοποιεί τη διαδικασία σύνδεσης των συσκευών στην πλατφόρμα AWS IoT, εφαρμόζοντας τις απαιτήσεις ασφάλειας για την σύνδεση με τον message broker της υπηρεσίας AWS IoT καθώς και την παροχή βασικών λειτουργιών publish / subscribe. Το SDK υποστηρίζει επί του παρόντος την C για και την JavaScript.

Όλες οι υπηρεσίες AWS μπορούν να διαχειρίζονται μέσω της διασύνδεσης γραμμής εντολών AWS (CLI), η οποία είναι μια ενοποιημένη διασύνδεση γραμμής εντολών για το AWS, η οποία διατίθεται για συστήματα Windows, Mac και Linux. Παρέχει εργαλεία για τη διαμόρφωση, τον έλεγχο και την αυτοματοποίηση των υπηρεσιών AWS μέσω scripts.

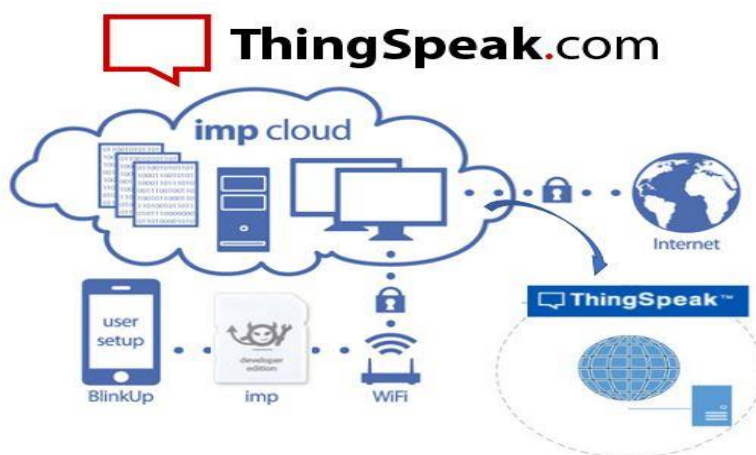
5.5.17 Συμπεράσματα

Η AWS IoT πλατφόρμα μπορεί να περιγράφει ως μία γενική λύση για την ανάπτυξη ενός IoT οικοσυστήματος. Με άλλα λόγια μπορεί να χαρακτηριστεί ως μία σουίτα, η οποία αποτελείται από τέσσερα διαφορετικά δομικά στοιχεία IoT λογισμικού:

- (1) το AWS IoT Core το οποίο επιτρέπει στις συνδεδεμένες συσκευές να αλληλοεπιδρούν με τις cloud εφαρμογές και άλλες συσκευές.
- (2) το AWS IoT Device Management, επιτρέπει στους χρήστες να ενσωματώνουν, να οργανώνουν, να παρακολουθούν και να διαχειρίζονται εξ αποστάσεως τις IoT συσκευές IoT.
- (3) το AWS IoT Device Defender, μέσω του οποίου ελέγχονται συνεχώς οι διάφορες αλλαγές – παραμετροποιήσεις στο IoT περιβάλλον.
- (4) το AWS IoT Analytics, το οποίο διευκολύνει την ανάλυση μεγάλου όγκου δεδομένων. Ουσιαστικά πρόκειται για μία ολοκληρωμένη πλατφόρμα η οποία παρέχει ασφάλεια, ευελιξία, λειτουργικότητα σε χαμηλό κόστος σε σύγκριση με τον ανταγωνισμό.

5.6 ThingSpeak IoT platform

Η ThingSpeak (<https://thingspeak.com/>) είναι μια ανοιχτή IoT πλατφόρμα η οποία βασίζεται σε public cloud αρχιτεκτονική. Η συγκεκριμένη πλατφόρμα επιτρέπει τη συλλογή, ανάλυση και επεξεργασία δεδομένων σε πραγματικό χρόνο μέσω ενός Open API [1]. Μέσω διαφόρων εφαρμογών και plugins υπάρχει η δυνατότητα για αποθήκευση των δεδομένων, απεικόνιση αυτών σε διάφορες μορφές, παρακολούθηση και ενσωμάτωση των δεδομένων του χρήστη με διάφορες πλατφόρμες τρίτων παρόχων όπως το Arduino, το Twilio, το Twitter, το ThingHTTP, το MATLAB [41]. Το κύριο στοιχείο της πλατφόρμας ThingSpeak είναι το "**κανάλι**". Τα κανάλια αποθηκεύουν όλα τα δεδομένα που συλλέγει μια εφαρμογή ThingSpeak. Κάθε κανάλι περιλαμβάνει οκτώ πεδία που μπορούν να χωρέσουν οποιοδήποτε τύπο δεδομένων σε αριθμητική ή αλφαριθμητική μορφή, συν τρία πεδία για δεδομένα θέσης (υψόμετρο, γεωγραφικό πλάτος και μήκος) και ένα για δεδομένα κατάστασης.



Πηγή: <https://thingspeak.com/>

5.6.1 Integration - API

Η ThingSpeak είναι μια IoT πλατφόρμα που χρησιμοποιεί κανάλια για την αποθήκευση δεδομένων που αποστέλλονται από εφαρμογές ή συσκευές. Από την στιγμή που θα δημιουργηθεί το κανάλι υπάρχει πλέον η δυνατότητα για την αποστολή και την ανάκτηση δεδομένων προς και από το κανάλι. Επίσης, υπάρχει η δυνατότητα για την δημιουργία δημοσίων καναλιών, για την κοινή χρήση των δεδομένων. Χρησιμοποιώντας API REST [13] κλήσεις όπως GET, POST, PUT και DELETE, μπορείτε να δημιουργηθεί ένα κανάλι και να ενημερώνεται η ροή του, να γίνεται ενημέρωση σε ένα υπάρχον κανάλι, να διαγραφεί μια ροή καναλιών καθώς και η διαγραφή ενός ολόκληρου καναλιού [41]. Μπορείτε επίσης να γίνει χρήση της μεθόδου MQTT Publish για την ενημέρωση μιας ροής καναλιών καθώς και η MQTT Subscribe για την λήψη μηνυμάτων όταν ένα κανάλι ενημερώνεται.

5.6.2 Ασφάλεια

Τα κανάλια μέσω των οποίων ανταλλάσσονται τα δεδομένα χωρίζονται σε δύο κατηγορίες: την δημόσια (public) και τα ιδιωτικά. Από προεπιλογή, ένα κανάλι είναι ιδιωτικό και απαιτεί ένα κλειδί API ανάγνωσης ώστε κάποιος να αποκτήσει πρόσβαση στην ροή (feed) του καναλιού. Ο ιδιοκτήτης του καναλιού έχει την δυνατότητα να κάνει ένα κανάλι κοινό, δίνοντας σε άλλους χρήστες τη δυνατότητα χρήσης της ροής του καναλιού χωρίς κάποιο κλειδί ανάγνωσης [41]. Μέσω των APIs κλειδιών επιτρέπεται η εγγραφή και η ανάγνωση των δεδομένων, χρησιμοποιώντας τον κώδικα API του ThingSpeak ή τον κώδικα MATLAB. Το κλειδί API αποτελείται από μια συμβολοσειρά 16 ψηφίων η οποία δημιουργείται αυτόματα, ταυτόχρονα με την δημιουργία ενός καναλιού. Τα κλειδιά APIs ανάγνωσης μπορούν να δημιουργηθούν εκ νέου στις ρυθμίσεις που παρέχονται για την παραμετροποίηση των καναλιών, κάτι το οποίο επιτρέπει την ταυτόχρονη ύπαρξη πολλαπλών κλειδιών API's για διαφορετικές εφαρμογές. Τα κλειδιά APIs πρέπει να καθορίζονται σε κάθε αίτημα HTTP. Το κλειδί API ενός χρήστη χρησιμοποιείται για κλήσεις API, όπως η δημιουργία ενός νέου καναλιού.

5.6.3 Πρωτόκολλα Επικοινωνίας

Το MQTT είναι ένα κοινό πρωτόκολλο που χρησιμοποιείται στα συστήματα IoT για τη σύνδεση συσκευών και αισθητήρων χαμηλού επιπέδου (low-level). Το MQTT χρησιμοποιείται για τη μετάδοση σύντομων μηνυμάτων από και προς έναν μεσίτη (broker). Η πλατφόρμα ThingSpeak παρέχει έναν μεσίτη MQTT, ώστε οι συσκευές να μπορούν να στέλνουν μηνύματα στο ThingSpeak [13].

5.6.4 Απεικόνιση και Ανάλυση Δεδομένων

Για την απεικόνιση και την ανάλυση δεδομένων καθώς και για την εκτέλεση μιας ενέργειας σε συνάρτηση κάποιου συμβάντος η ThingSpeak πλατφόρμα χρησιμοποιεί τις λεγόμενες *εφαρμογές (apps)*. Οι απεικονίσεις των μεταφορτωμένων (uploaded) δεδομένων μπορούν να πραγματοποιηθούν μέσω ενσωματωμένων γραφημάτων ή μέσω της MATLAB εφαρμογής απεικόνισης. Αυτή η εφαρμογή επιτρέπει στον χρήστη να δημιουργήσει μια διαδραστική ή στατική απεικόνιση, χρησιμοποιώντας λειτουργίες MATLAB με φιλικό προς τον χρήστη τρόπο [21]. Ο κώδικας MATLAB μπορεί να δημιουργηθεί μέσω ενός εκ των έτοιμων προτύπων που υπάρχουν διαθέσιμα καθώς και να επεξεργαστεί στον ενσωματωμένο επεξεργαστή κώδικα. Ένας άλλος τρόπος προβολής προσαρμοσμένων (custom) απεικονίσεων μπορεί να παραμετροποιηθεί μέσω διαφόρων Plugins. Τα συγκεκριμένα plugins επιτρέπουν στο χρήστη να χρησιμοποιεί γλώσσες HTML, CSS και JavaScript.

Ο κώδικας MATLAB μπορεί επίσης να εκτελεστεί στην εφαρμογή ανάλυσης δεδομένων MATLAB, η οποία μπορεί να χρησιμοποιηθεί για διάφορους υπολογισμούς, καθώς και μετασχηματισμούς

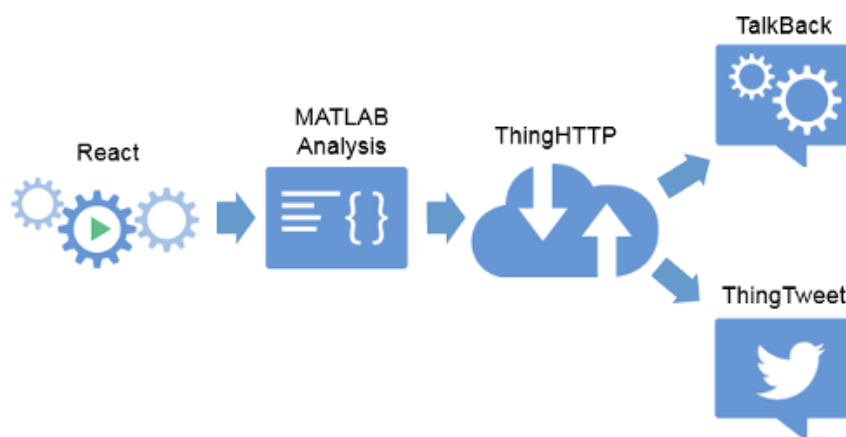
(transformation) των δεδομένων κ.λπ. Ένα παράδειγμα είναι ο υπολογισμός ενός μέσου όρου μιας επιλεγμένης τροφοδοσίας ή η εύρεση και η αντικατάσταση ελλειπουσών τιμών στα δεδομένα.

5.6.6 Hosting

Η ανάπτυξη (deployment) της συγκεκριμένης πλατφόρμας μπορεί να πραγματοποιηθεί τόσο σε cloud υποδομή όσο και με την εγκατάσταση του Thingspeak διακομιστή τοπικά σε Linux διανομή [21].

5.6.7 Push Notifications

Το ThingSpeak όπως αναφέραμε και προηγουμένως παρέχει διάφορες εφαρμογές (apps) για τη μετατροπή ή την απεικόνιση δεδομένων ή την ενεργοποίηση μιας κατάστασης. Στο παρακάτω παράδειγμα αναλύεται ο τρόπος μέσω του οποίου παρακολουθείται η δραστηριότητα του "καναλιού" ThingSpeak χρησιμοποιώντας κάποιες από τις εφαρμογές του ThingSpeak όπως: το MATLAB Analysis, το ThingHTTP, το TalkBack και το ThingTweet [41]. Εξετάζουμε ένα σενάριο (Εικόνα 5.11) όπου ένας αισθητήρας πίεσης παρακολουθεί την πίεση του λέβητα. Ο αισθητήρας πίεσης τροφοδοτεί συνεχώς τα δεδομένα στο κανάλι ThingSpeak. Αν το κανάλι σταματήσει να λαμβάνει τα δεδομένα πίεσης, αμέσως αποστέλλεται ειδοποίηση σχετικά με το συμβάν.



Εικόνα 5.11: Παράδειγμα αποστολής ειδοποίησης

5.6.8 Ενεργοποίηση συμβάντων (Trigger)

Η πλατφόρμα παρέχει έναν ασυνήθιστο τρόπο αντίδρασης και εκτέλεσης διαφόρων ενεργειών στην περίπτωση που συμβούν διάφορες αλλαγές στα δεδομένα, μέσω ενσωματωμένων εφαρμογών οι οποίες αλληλοεπιδρούν με την υπηρεσία κοινωνικής δικτύωσης Twitter [21]. Οι συγκεκριμένες εφαρμογές επιτρέπουν σε έναν χρήστη να συνδέσει τον προσωπικό του λογαριασμό στο ThingSpeak με τον προσωπικό του λογαριασμό στο Twitter με αποτέλεσμα να ακούσει (listen) και να αντιδράσει (react) στα εισερχόμενα

tweets (εφαρμογή TweetControl) ή να στείλει ένα νέο tweet ως ειδοποίηση για αλλαγές οι οποίες έχουν συμβεί στα δεδομένα (ThingTweet).

- **Εφαρμογή React app:** Η εφαρμογή React app συνεργάζεται με άλλες εφαρμογές για την εκτέλεση ενεργειών όταν τα δεδομένα τα οποία έχουν ληφθεί μέσω των καναλιών πληρούν μια συγκεκριμένη προϋπόθεση. Ο χρήστης μπορεί να επιλέξει μεταξύ διαφορετικών τύπων συνθηκών και ενεργειών. Παρόλο που ένας χρήστης μπορεί να επιλέξει έναν τύπο συνθηκών μέσα από μια λίστα συνθήκες, παρέχεται επίσης και η δυνατότητα για την παραμετροποίηση πιο περίπλοκων ενεργοποιητών μέσω της εφαρμογής ανάλυσης MATLAB
- **Εφαρμογή ThingHTTP:** Η εφαρμογή ThingHTTP επιτρέπει την επικοινωνία συσκευών με υπηρεσίες ιστού (web services), API's καθώς και με οποιεσδήποτε συσκευές με δυνατότητα HTTP. Επιτρέπει την αποστολή αίτησης (request) HTTP το οποίο ενεργοποιείται από άλλες εφαρμογές ThingSpeak.

5.6.9 Αποθήκευση Δεδομένων

Η πλατφόρμα ThingSpeak αποθηκεύει όλες τις πληροφορίες που αποστέλλονται από τις συσκευές σε μια κεντρική τοποθεσία σε cloud υποδομή, ώστε να μπορεί ο χρήστης να έχει εύκολη πρόσβαση στα δεδομένα. Τα δεδομένα προστατεύονται μέσω API κλειδιών τα οποία είναι υπό τον έλεγχο των χρηστών. Επίσης, υπάρχει η δυνατότητα ενσωμάτωσης της πλατφόρμας με συστήματα παραγωγής μεγάλης κλίμακας όπως το Azure Blob και Azure SQL [16].

5.6.10 Αποθήκευση Blob

Με την χρήση του MATLAB υποστηρίζεται η διαχείριση των δεδομένων με χρονική σήμανση (time-stamped) και μη δομημένα δεδομένα από πολλές πηγές, συμπεριλαμβανομένων υπηρεσιών αποθήκευσης cloud (π.χ. AWS S3, Azure Blob).

5.6.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)

Μέσα από τη βιβλιογραφία καθώς και την επίσημη ιστοσελίδα της πλατφόρμας δεν περιγράφεται η υποστήριξη Android και iOS SDKs.

5.6.12 Geolocation

Ο εντοπισμός θέσης των συνδεδεμένων συσκευών πραγματοποιείται με την χρήση της υπηρεσίας Google Maps.

5.6.13 Πλάνα Τιμολόγησης

Το ThingSpeak διατίθεται ως δωρεάν υπηρεσία για μη εμπορικές υλοποιήσεις (<3 εκατομμύρια μηνύματα / έτος ή ~ 8.200 μηνύματα / ημέρα). Για μεγαλύτερες υλοποιήσεις - έργα ή εμπορικές εφαρμογές, προσφέρονται τέσσερις διαφορετικοί τύποι ετήσιων αδειών: Standard, Academic, Student and Home [41].

5.6.14 Υποστηριζόμενες γλώσσες προγραμματισμού

Στην συγκεκριμένη πλατφόρμα υποστηρίζεται μόνο η γλώσσα Matlab [21].

5.6.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων

Η ThingSpeak πλατφόρμα συνεργάζεται με την εταιρεία MathWorks και παρέχει ενσωματωμένη υποστήριξη από την υψηλού επιπέδου γλώσσα προγραμματισμού MATLAB προσανατολισμένη στον αριθμητικό υπολογισμό [41]. Η συνδεσιμότητα με το MathWorks επιτρέπει στους χρήστες να αξιοποιήσουν τις δυνατότητες του MATLAB για απεικόνιση και ανάλυση των μεταφορτωμένων δεδομένων χωρίς να απαιτείται η αγορά άδειας (licensing) MATLAB. Επίσης, η πλατφόρμα διαθέτει εργαλειοθήκη υποστήριξης για το MATLAB, για την ανάλυση και να απεικόνιση των δεδομένων που είναι αποθηκευμένα στην πλατφόρμα.

5.6.16 Ενσωμάτωση Hardware Πλατφορμών

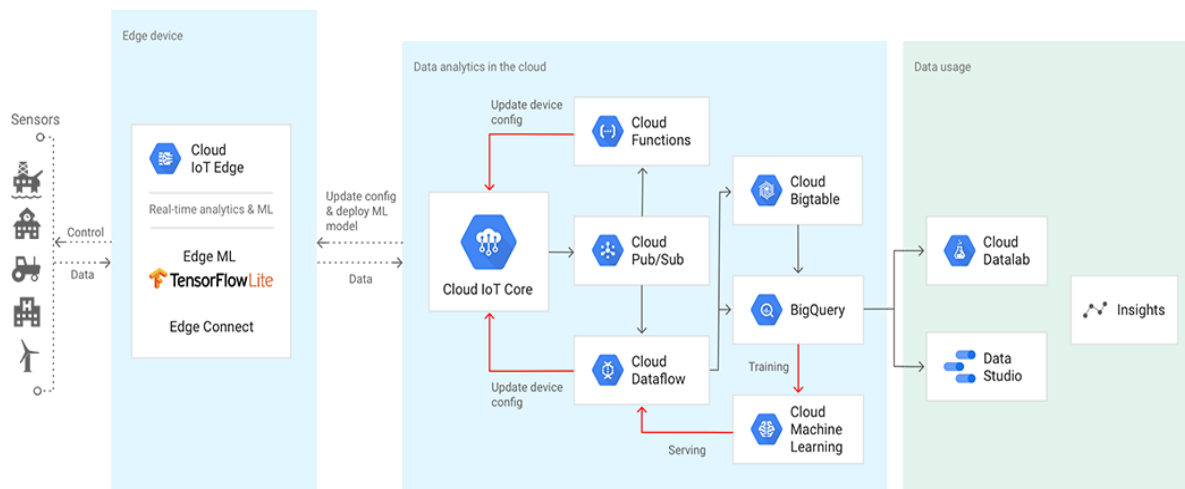
Αυτή η IoT πλατφόρμα υποστηρίζει αρκετές hardware συσκευές, όπως Arduino, Raspberry Pi, ESP, Particle και άλλες [16].

5.6.17 Συμπεράσματα

Το Thingspeak είναι μια πλατφόρμα ανοικτού κώδικα (open source) η οποία επιτρέπει την συλλογή και την αποθήκευση των δεδομένων που συλλέγονται από τους αισθητήρες στο cloud περιβάλλον. Παρέχει διάφορες εφαρμογές όπου σε συνδυασμό με το Matlab επιτυγχάνεται η ανάλυση και η απεικόνιση των δεδομένων. Μπορείτε να συνεργαστεί με διάφορες πλατφόρμες υλικού όπως το Arduino, το Raspberry Pi, το Beaglebone και άλλα. Επίσης, μπορούν να δημιουργηθούν ξεχωριστά "κανάλια" για την αποθήκευση των δεδομένων. Επιπλέον πλεονεκτήματα αποτελούν η δωρεάν φιλοξενία (hosting) στο cloud καθώς η δυνατότητα πρόσθετων λειτουργιών για γλώσσες προγραμματισμού όπως η Ruby, η Node.js και η Python. Ωστόσο ως μειονέκτημα χαρακτηρίζεται ο περιορισμένος όγκος των δεδομένων που μπορούν να σταλθούν μέσω των APIs.

5.7 Google Cloud IoT Platform

Το Google Cloud IoT (<https://cloud.google.com/solutions/iot/>), Εικόνα 5.12, είναι ένα πλήρες σύνολο εργαλείων για τη σύνδεση, επεξεργασία, αποθήκευση και ανάλυση δεδομένων τόσο στο σημείο το οποίο βρίσκονται εγκατεστημένες οι συσκευές (edge/on-premises) όσο και στην cloud υποδομή [42]. Η πλατφόρμα αποτελείται από κλιμακούμενες, πλήρως διαχειριζόμενες υπηρεσίες cloud με στόχο την ικανοποίηση των IoT αναγκών. Μέσω του Cloud IoT Core, σε συνδυασμό με άλλες υπηρεσίες στην πλατφόρμα Cloud IoT, παρέχεται μια ολοκληρωμένη λύση για τη συλλογή, επεξεργασία, ανάλυση και απεικόνιση των δεδομένων IoT σε πραγματικό χρόνο, για τη βελτίωση της λειτουργικής αποτελεσματικότητας.



Εικόνα 5.12: Cloud IoT Αρχιτεκτονική

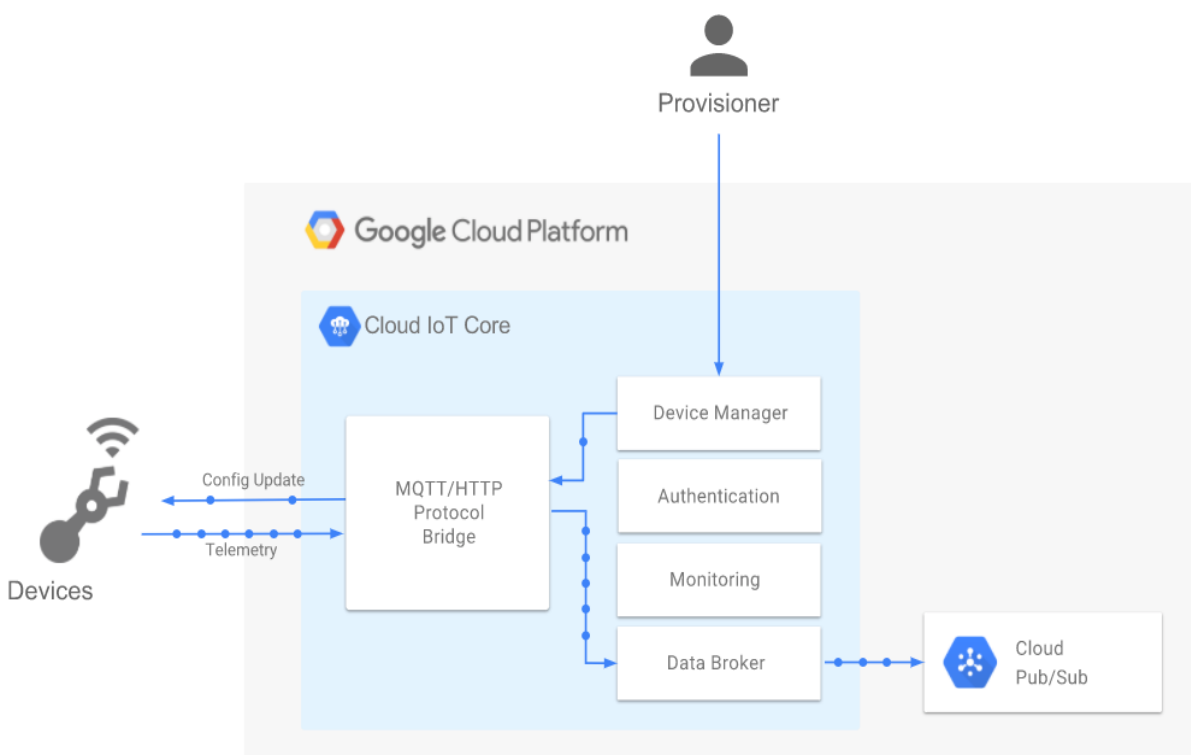
Η IoT πλατφόρμα της Google, προσφέρει λειτουργίες όπως:

- Παροχή τεράστιου αποθηκευτικού χώρου.
- Αποτελεσματικότητα και ευελιξία.
- Ανάλυση μεγάλου όγκου δεδομένων.
- Παροχή υπηρεσιών μέσω ενός πλήρως προστατευμένου και ευφυούς IoT περιβάλλοντος.

Τα κύρια συστατικά του Cloud IoT Core είναι ο διαχειριστής συσκευών (device manager) και τα πρωτόκολλα επικοινωνίας μέσω των οποίων διασυνδέονται οι IoT συσκευές με την υπόλοιπη υποδομή [16]:

- **Device Manager**, μέσω του διαχειριστή συσκευών επιτυγχάνεται η καταχώρηση των συσκευών και η δυνατότητα να μπορούν να παραμετροποιηθούν.
- **Protocol bridges**, δύο πρωτόκολλα επικοινωνίας (MQTT και HTTP), τα οποία μπορούν να χρησιμοποιήσουν οι συσκευές για να συνδεθούν με το Google Cloud Platform.

Τα δεδομένα που συλλέγονται από την συσκευή προωθούνται στην υπηρεσία Cloud Pub / Sub, η οποία στη συνέχεια μπορεί να χρησιμοποιηθεί για να ενεργοποιήσει τις διάφορες Cloud λειτουργίες. Μπορείτε επίσης να πραγματοποιηθεί και ανάλυση ροής μέσω του Cloud Dataflow. Η Εικόνα 5.13 συνοψίζει τα στοιχεία της υπηρεσίας και τη ροή δεδομένων:



Εικόνα 5.13: Διαδικασία ροής δεδομένων

5.7.1 Cloud IoT Core API's

Το συγκεκριμένο API αποτελείται από δύο REST σύνολα, το **cloudiot** και το **cloudiotdevice**. Οι **cloudIoT** μέθοδοι, διευκολύνουν τις εργασίες διαχείρισης των συσκευών όπως η ενημέρωση των ρυθμίσεων των συσκευών και η λήψη των καταστάσεων συσκευών. Από την άλλη μεριά οι μέθοδοι **cloudiotdevice** διευκολύνουν την επικοινωνία της συσκευής μέσω του πρωτοκόλλου HTTP [14]. Μέσω αυτών των μεθόδων μπορούν να ληφθούν ρυθμίσεις σχετικές με την παραμετροποίηση των συσκευών καθώς και με την ρύθμιση της κατάστασης της συσκευής. Η τρέχουσα έκδοση του API Core Cloud IoT είναι v1.

5.7.2 Ασφάλεια

Η ασφάλεια αποτελεί κρίσιμο ζήτημα κατά την ανάπτυξη και τη διαχείριση IoT συσκευών. Το Cloud IoT Core προσφέρει τα ακόλουθα χαρακτηριστικά ασφαλείας [42]:

- Ο έλεγχος ταυτότητας ανά συσκευή κάνοντας χρήση δημόσιου / ιδιωτικού κλειδιού μέσω JSON Web Tokens (JWTs, RFC 7519). Αυτό περιορίζει το αντίκτυπο μιας επίθεσης, επειδή θα επηρεάσει μόνο μια μόνο συσκευή και όχι τις υπόλοιπες.
- Υποστήριξη αλγορίθμων RSA ή Elliptic Curve για την επαλήθευση υπογραφών (signatures), με εφαρμογή για ισχυρών κλειδιών.
- TLS 1.2, χρησιμοποιώντας πιστοποιητικά (απαιτείται για το MQTT).
- Η πρόσβαση στο API Core Cloud IoT ελέγχεται από τους ρόλους και τα δικαιώματα του Cloud Identity and Access Management (IAM).

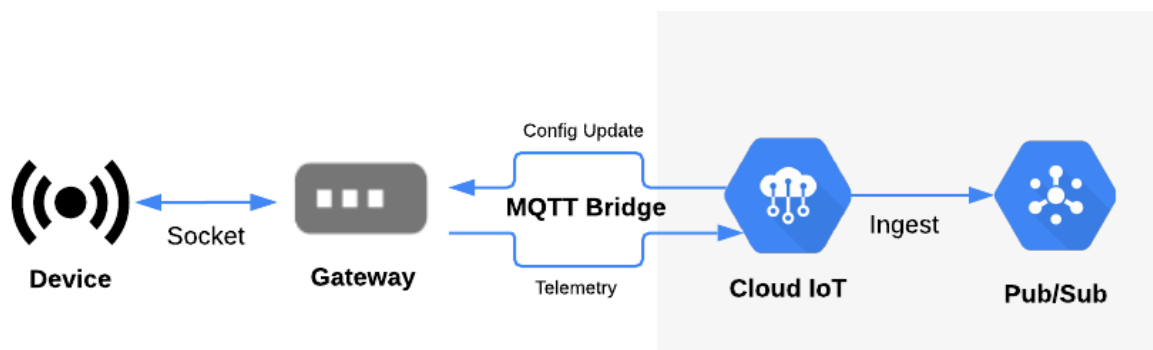
5.7.3 Πρωτόκολλα Επικοινωνίας

Η διασύνδεση των συσκευών με την Cloud IoT Core υποδομή πραγματοποιείται μέσω των πυλών (gateways), οι οποίες εκτελούν διάφορα καθήκοντα όπως:

- Επικοινωνία με την Cloud IoT Core υποδομή
- Επικύρωση των διαπιστευτηρίων της συσκευής
- Αποθήκευση και επεξεργασία δεδομένων
- Ρύθμιση της κατάστασης της συσκευής
- Μετάφραση Πρωτοκόλλων

Η Cloud IoT Core πλατφόρμα, υποστηρίζει επικοινωνία μέσω των πρωτοκόλλων MQTT και HTTP [14].

Η Εικόνα 5.14 παρέχει μια επισκόπηση της αρχιτεκτονικής για την χρήση του πρωτοκόλλου MQTT.



Εικόνα 5.14: Επικοινωνία μέσω πρωτοκόλλου MQTT

5.7.4 Ανάλυση δεδομένων

Μέσω του BigQuery το οποίο είναι μια RESTful web υπηρεσία μέσω του οποίου πραγματοποιείται ανάλυση μεγάλων συνόλων δεδομένων. Μπορεί να γίνει ανάλυση τόσο σε δεδομένα συνεχούς ροής όσο και σε δεδομένα τα οποία βρίσκονται ήδη αποθηκευμένα [42]. Το BigQuery παρέχει εξωτερική πρόσβαση στην τεχνολογία **Dremel** το οποίο αποτελεί ένα καταναμημένο σύστημα που αναπτύχθηκε από την Google

για την αλληλεπιδραστική αναζήτηση μεγάλων συνόλων δεδομένων. Επίσης μέσω του BigQuery παρέχονται λειτουργίες όπως:

- Δημιουργία και διαγραφή πινάκων με δεδομένα τα οποία είναι μορφής CSV ή JSON και έχουν εισαχθεί Google Storage.
- Το BigQuery μπορεί να χρησιμοποιηθεί από την εφαρμογή Google Apps Script καθώς με REST APIs.
- Διαμοιρασμός αναλύσεων και αναφορών με διάφορες ομάδες ατόμων ανά τον κόσμο.

5.7.5 Απεικόνιση δεδομένων

Η απεικόνιση των δεδομένων επιτυγχάνεται μέσω της εφαρμογής Google Data Studio [42]. Μέσω της συγκεκριμένης υπηρεσίας παρέχονται διάφορα widgets τα οποία προσφέρουν διάφορες επιλογές προσαρμογής και διαδραστικούς ελέγχους.

5.7.6 Hosting

Η ανάπτυξη και η διαχείριση της πλατφόρμας πραγματοποιείται μόνο μέσω της Google cloud υποδομής.

5.7.7 Push Notifications

Μέσω της υπηρεσίας Cloud Pub / Sub, η οποία είναι μια πλήρως διαχειριζόμενη υπηρεσία μηνυμάτων σε πραγματικό χρόνο, παρέχεται η αποστολή και η λήψη μηνυμάτων μεταξύ ανεξάρτητων εφαρμογών [14].

5.7.8 Ενεργοποίηση Συμβάντων

Μέσω της υπηρεσίας Google Cloud Functions παρέχεται η δυνατότητα δημιουργίας προσαρμοσμένων σεναρίων (scripts) τα οποία μπορούν να εφαρμοστούν για την εκτέλεση ενεργειών όταν κάποιο συμβάν λαμβάνει μέρος. Για παράδειγμα μπορούν να χρησιμοποιηθούν για την αποστολή ειδοποιήσεων, τη διόρθωση μη έγκυρων δεδομένων ή την επίκληση άλλων APIs [16].

5.7.9 Αποθήκευση Δεδομένων

Μέσω της Google IoT πλατφόρμας προσφέρεται μία σειρά λύσεων αποθήκευσης μη δομημένων δεδομένων (blob), όπως εικόνες ή ροές βίντεο. Η αποθήκευση τόσο των επεξεργασμένων όσο και των μη επεξεργασμένων δεδομένων επιτυγχάνεται με την χρήση δομημένων βάσεων δεδομένων όπως το Cloud Datastore και το Firebase Realtime. Το Cloud Datastore [42] είναι μια κλιμακούμενη βάση δεδομένων NoSQL για την αποθήκευση των εφαρμογών και των δεδομένων αυτών. Το Cloud Datastore υποστηρίζει με αυτόματο τρόπο τη δυνατότητα της συνεχούς αντιγραφής των δεδομένων (replacation) σε άλλο σύστημα αποθήκευσης (storage) στον ίδιο ή σε απομακρυσμένο χώρο. Επίσης, το Cloud Datastore παρέχει

μια ισχυρή μηχανή αναζήτησης που επιτρέπει την αναζήτηση των δεδομένων βάσει πολλαπλών ιδιοτήτων καθώς και την ταξινόμηση ανάλογα των αναγκών των χρηστών.

Το Firebase Realtime είναι μια βάση δεδομένων η οποία φιλοξενείται σε cloud υποδομή. Τα δεδομένα αποθηκεύονται ως μορφή JSON και συγχρονίζονται σε πραγματικό χρόνο σε κάθε συνδεδεμένη συσκευή. Όταν δημιουργείτε εφαρμογές πολλαπλών πλατφορμών με χρήση των SDK iOS, Android και JavaScript, όλες οι συσκευές μοιράζονται ένα στιγμιότυπο βάσης δεδομένων πραγματικού χρόνου και λαμβάνουν αυτόματα ενημερώσεις με τα νεότερα δεδομένα.

5.7.10 Αποθήκευση Blob

Η πλατφόρμα υποστηρίζει τόσο δομημένα όσο και μη δομημένα δεδομένα. Για παράδειγμα η τρέχουσα κατάσταση μιας IoT συσκευής αποτελείται από μη δομημένα δεδομένα τα οποία ρέουν μόνο στην κατεύθυνση από την συσκευή προς την cloud υποδομή [42].

5.7.10 Κιτ Ανάπτυξης Λογισμικού (SDKs)

Το SDK της Cloud IoT πλατφόρμας αποτελείται από βιβλιοθήκες πελάτη γραμμένες σε γλώσσα C κάτι το οποίο επιτρέπει στους προγραμματιστές να συνδέουν, να παρέχουν και να διαχειρίζονται με ασφάλεια συσκευές που είναι συνδεδεμένες Cloud IoT Core [16]. Το κιτ στοχεύει σε περιορισμένες εφαρμογές ενέργειας και μεγέθους, όπως κινητές συσκευές που λειτουργούν με μπαταρία και οι οποίες λειτουργούν ως συσκευές παρακολούθησης αποθεμάτων ή έξυπνες οικιακές συσκευές Wi-Fi με περιορισμένη μνήμη flash ROM (μνήμη μόνο για ανάγνωση).

5.7.12 Geolocation API

Με την χρήση της υπηρεσίας Google Maps επιτυγχάνεται η ανακάλυψη της γεωγραφικής τοποθεσίας των IoT συσκευών οι οποίες έχουν διασυνδεθεί με την πλατφόρμα χωρίς την χρήση GPS. Με μια μόνο γραμμή κώδικα, μια συσκευή ή ένας αισθητήρας διασκορπισμένος σε ένα μπορεί να έχει πρόσβαση στη γεωχωρική βάση δεδομένων των κυψελωτών δικτύων της Google χρησιμοποιώντας το API γεωγραφικής κατανομής της υπηρεσίας Google Maps [42]. Εναλλακτικά, μπορεί να γίνει των APIs της υπηρεσίας Google Maps σε συνδυασμό με υπάρχοντα συστήματα GPS, με αποτέλεσμα να υπάρξει μεγαλύτερη ακρίβεια στα δεδομένα θέσης. Παρομοίως, μπορεί να γίνει χρήση του **Timezone API** για την μετατροπή της UTC ζώνης σε τοπική ώρα. Επιπρόσθετα, μπορεί να γίνει χρήση και του **Elevation API** ώστε να υπολογιστεί η υψομετρική διαφορά από την επιφάνεια της θάλασσας.

5.7.13 Πλάνα Τιμολόγησης

Η τιμολόγηση των Google IoT υπηρεσιών βασίζεται στον όγκο των δεδομένων που ανταλλάσσονται. Τα πρώτα 250MB μηνιαίως είναι δωρεάν, αλλά στη συνέχεια υπάρχει ένας κλιμακωτός πίνακας τιμών με βάση τα συνολικά μηνιαία δεδομένα.

5.7.14 Υποστηριζόμενες γλώσσες προγραμματισμού

Οι υποστηριζόμενες γλώσσες προγραμματισμού είναι οι: Go, Java, .NET, Node.js, php, Python, Ruby.

5.7.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων

Παρέχει ενσωμάτωση με άλλες υπηρεσίες της Google [42].

5.7.16 Ενσωμάτωση με Hardware πλατφόρμες

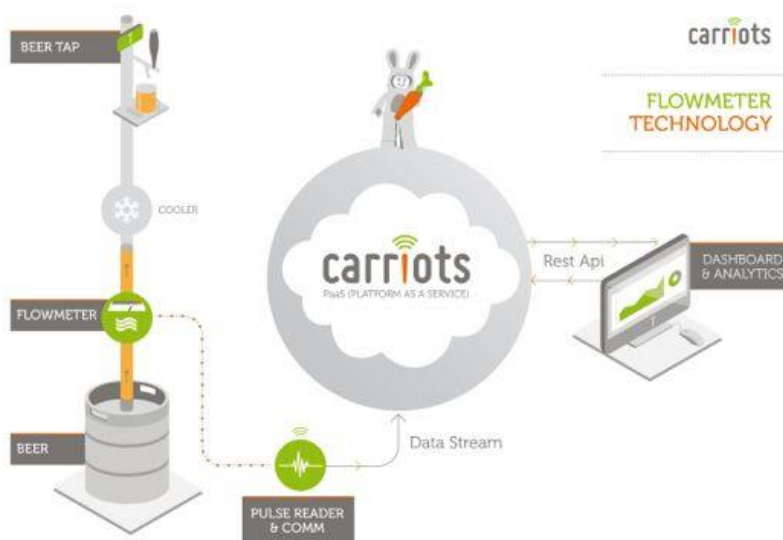
Υπάρχει μία μεγάλη ποικιλία hardware πλατφορμών το οποίο υποστηρίζεται από την πλατφόρμα, με στόχο την κατασκευή IoT εφαρμογών. Τα συνηθισμένα παραδείγματα πλατφορμών περιλαμβάνουν μικροελεγκτές όπως το Beaglebone το Raspberry Pi, το Arduino, το Feather Adafruit και άλλα [16].

5.7.17 Συμπεράσματα

Το Cloud IoT Core είναι μια πλήρως διαχειριζόμενη πλατφόρμα η οποία επιτρέπει την σύνδεση, την διαχείριση και την λήψη δεδομένων από εκατομμύρια συσκευές που μπορούν να βρίσκονται οπουδήποτε. Το Cloud IoT Core, σε συνδυασμό με άλλες υπηρεσίες της Google Cloud πλατφόρμα, παρέχει μια ολοκληρωμένη λύση για τη συλλογή, επεξεργασία, ανάλυση και απεικόνιση των IoT δεδομένων IoT σε πραγματικό χρόνο. Εστιάζει στην γρήγορη και με εύκολο τρόπο ανάπτυξη των IoT εφαρμογών. Η τιμολόγηση βασίζεται στην χρέωση ανά λεπτό, κάτι το οποίο την καθιστά φθηνότερη σε σχέση με άλλες πλατφόρμες. Ωστόσο, εντοπίζονται και μειονεκτήματα όπως ο περιορισμός στην ενσωμάτωση εφαρμογών τρίτων παρόχων καθώς και οι περιορισμένες επιλογές ως προς τις γλώσσες προγραμματισμού.

5.8 Carriots IoT platform

Η Carriots (<https://www.altairsmartworks.com/>) της Altair Engineering προσφέρει μια IoT πλατφόρμα (IoT) από άκρο σε άκρο (end to end), σχεδιασμένη για τις σημερινές ανάγκες της βιομηχανίας. Είναι μία πλατφόρμα η οποία έχει σχεδιαστεί για M2M (machine to machine) επικοινωνίες. Επιτυγχάνει την ανάπτυξη εφαρμογών IoT και παρέχει απλή επεκτασιμότητα καθώς επίσης επιτρέπει στους χρήστες να συλλέγουν και να αποθηκεύουν τα δεδομένα από τις συνδεδεμένες συσκευές και να δημιουργούν ισχυρές εφαρμογές [43]. Η ανοιχτή αρχιτεκτονική, η ενσωμάτωση των πλατφορμών λογισμικού HyperWorks και Carriots της Altair για σχεδιασμό και διαχείριση συσκευών με γνώμονα την προσομοίωση επιτρέπει μια ολοκληρωμένη λύση IoT η οποία περιλαμβάνει, cloud Computing Υψηλών Επιδόσεων και ανάλυση μεγάλου όγκου δεδομένων (Big Data analytics) με στόχο την υποστήριξη κάθε IoT υλοποίησης. Τα REST APIs και Groovy SDKs [5] διατίθενται για ανάπτυξη εφαρμογών ιστού. Η μορφή δεδομένων που υποστηρίζεται είναι JSON και XML. Τα δεδομένα αποθηκεύονται στην πλατφόρμα και απαιτούνται κλειδιά πρόσβασης για την πρόσβαση σε αυτήν.



Πηγή: <https://www.altairsmartworks.com/>

Στην πλατφόρμα συμπεριλαμβάνονται τα εξής χαρακτηριστικά:

- **Διαχείριση συσκευών:** Απομακρυσμένη συντήρηση, έλεγχος και αλληλεπίδραση με τις συσκευές ανεξάρτητα από την τοποθεσία τους.
- **Ακροατές (Listeners):** Ενέργειες όπως η λήψη δεδομένων, η αποθήκευση, όταν μια συσκευή συνδέεται ή αποσυνδέεται, όλα αυτά τα συμβάντα επιτρέπουν την αντίδραση με δομές IF-THEN-ELSE.
- **Κανόνες (Rules):** Σύνθετα σενάρια (scripts), επαναχρησιμοποιούμενα κομμάτια κώδικα. Οι κανόνες Groovy script αποθηκεύονται ώστε να χρησιμοποιηθούν σε ακροατές.

- **Ενεργοποιητές (Triggers):** Προωθεί (Push) δεδομένα σε ένα εξωτερικό σύστημα
- **Μηχανή εφαρμογής SDK:** Οι ακροατές και οι κανόνες εκτελούνται από αυτό το SDK
- **Εξαγωγή δεδομένων:** ενσωμάτωση με άλλα συστήματα, εξαγωγή αρχείων, προώθηση (push) δεδομένων σε άλλες βάσεις δεδομένων ή χρήση REST API για διαχείριση εξερχόμενων δεδομένων.
- **Προσαρμοσμένες ειδοποιήσεις:** Δημιουργία alarms στην περίπτωση που προκληθεί η οποιαδήποτε δυσλειτουργία.
- **Διαχείριση κλειδιών API:** Η πρόσβαση API ελέγχεται από κλειδιά API
- **Διαχείριση χρηστών:** Οι χρήστες μπορούν να διαχειρίζονται άλλους λογαριασμούς χρηστών για τη διαχείριση έργων ή για τον καθορισμό δικαιωμάτων
- **Προσαρμοσμένος πίνακας ελέγχου:** Δημιουργία προσαρμοσμένου πίνακα ελέγχου με σκοπό την διαχείριση όλων των οντοτήτων της πλατφόρμας, χρησιμοποιώντας API REST.

5.8.1 Integration – APIs

Η Carriots πλατφόρμα επιτρέπει την συνδεσιμότητα με οποιαδήποτε συσκευή και την διαχείριση της ροής δεδομένων μέσω REST API [2].

5.8.2 Ασφάλεια

Η επικοινωνία μέσω των REST APIs επιτυγχάνεται μέσω των πρωτοκόλλων HTTP ή HTTPS. Το τελευταίο συνιστάται ώστε να εξασφαλίζεται η κρυπτογράφηση της σύνδεσης μεταξύ των εφαρμογών / των συσκευών και της πλατφόρμας [43].

Η Carriots IoT πλατφόρμα χρησιμοποιεί το πρωτόκολλο MQTT για να στέλνει ροές και να ανταλλάσσει πληροφορίες με τις IoT συσκευές. Το MQTT βασίζεται στο TCP ως πρωτόκολλο μεταφοράς του, πράγμα που σημαίνει ότι από προεπιλογή η σύνδεση δεν χρησιμοποιεί κρυπτογραφημένη επικοινωνία. Για την κρυπτογράφηση ολόκληρης της επικοινωνίας MQTT [16], οι περισσότεροι MQTT Brokers επιτρέπουν στους clients να χρησιμοποιούν TLS αντί για απλό TCP. Η θύρα TCP / IP 1883 προορίζεται αποκλειστικά για χρήση με το MQTT και η θύρα TCP / IP 8883 επίσης προορίζεται για τη χρήση του MQTT [43] μέσω TLS.

5.8.3 Πρωτόκολλα Επικοινωνίας

Η Carriots πλατφόρμα για την αποστολή και ανάκτηση δεδομένων από τις IoT συσκευές χρησιμοποιεί το HTTP και το MQTT ως πρωτόκολλα επικοινωνίας.

5.8.4 Ανάλυση - Απεικόνιση

Το Carriots Analytics είναι μια cloud πλατφόρμα ανάλυσης δεδομένων για εφαρμογές IoT. Παρέχει ασφαλή και γρήγορη απεικόνιση σε πραγματικό χρόνο. Τα κύρια χαρακτηριστικά είναι τα εξής [3]:

- Ενσωμάτωση διαδραστικών χαρτών.
- Κοινή χρήση διαγραμμάτων και πινάκων ελέγχου (dashboards) μέσω διαδικτύου.
- Εύκολη εξαγωγή μιας ενιαίας σελίδας ή πλήρους αναφοράς σε μορφή .pdf ή σε μορφή PowerPoint.
- Παρουσίαση γραφημάτων μέσω Κοινωνικών δικτύων όπως το Facebook, το LinkedIn, το Twitter, το Tumblr, το Pinterest
- Πλήρεις δυνατότητες GIS: Υποστήριξη δεδομένων Geo JSON.

5.8.6 Hosting

Παρέχεται η δυνατότητα της on-premise ανάπτυξης, δηλαδή της εγκατάστασης σε κάποια τοπική υποδομή καθώς και η ανάπτυξη σε cloud υποδομές.

5.8.7 Push Notifications

Κατά την διάρκεια μη κανονικών καταστάσεων η πλατφόρμα διαθέτει μηχανισμούς δημιουργίας ειδοποιήσεων. Η δημιουργία και η διαχείριση προσαρμοσμένων συναγερμών (alarms) χρησιμοποιούνται για την αναγνώριση και την απόρριψη κάποιας ασυνήθιστης ροής.

5.8.8 Ενεργοποίηση Συμβάντων

Η ενεργοποίηση συμβάντων μπορεί να πραγματοποιηθεί μέσω της υπηρεσίας IFTTT (if this, then that) καθώς και με την χρήση προσαρμοσμένων APIs [43].

5.8.9 Αποθήκευση Δεδομένων

Η πλατφόρμα διαθέτει την ικανότητα αποθήκευσης μεγάλου όγκου δεδομένων διαμέσου μιας αρχιτεκτονικής η οποία βασίζεται σε δομή χωρίς σχηματισμό (schema less structure). Με αυτόν τον τρόπο παρέχει στα IoT προγράμματα την ευελιξία να διαχειρίζονται ετερογενή δεδομένα από διαφορετικές συσκευές. Επίσης, γίνεται χρήση NoSQL βάσης δεδομένων.

5.8.10 Αποθήκευση Blob

Με την χρήση της υπηρεσίας Google Maps επιτυγχάνεται η ανακάλυψη της γεωγραφικής τοποθεσίας των IoT συσκευών

5.8.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)

Η πλατφόρμα αυτή αναπτύσσεται μέσω της γλώσσας προγραμματισμού Groovy [5]. Η Groovy είναι μια δυναμική γλώσσα που στοχεύει στη βελτίωση της παραγωγικότητας των προγραμματιστών χάρη σε μια συνοπτική, οικεία και εύκολη σύνταξη. Διαθέτει επίσης και πρόσθετα χαρακτηριστικά εμπνευσμένα από γλώσσες όπως Python, Ruby ή Smalltalk.

Το Carriots SDK περιλαμβάνει ένα σύνολο βασικών βιβλιοθηκών για την διαχείριση των οντοτήτων της πλατφόρμας καθώς επίσης και ένα σύνολο εργαλείων για προσαρμοσμένη χρήση. Οι πολιτικές ασφαλείας της Java παρέχουν ασφάλεια εντός του οικοσυστήματος SDK [16]. Ο σκοπός της πλατφόρμας Carriots είναι να εξασφαλίσει την ενθυλάκωση (encapsulation) και την ασφάλεια με την εκτέλεση αυθαίρετου κώδικα. Οι Προγραμματιστές μπορούν να δημιουργήσουν τον δικό τους πλήρη πίνακα ελέγχου πάνω από το υφιστάμενο Carriots REST API στη γλώσσα της επιλογής τους.

Επιπλέον, το HTTP / HTTPS RESTful API μπορεί να χρησιμοποιηθεί για να ωθήσει και να τραβήξει (push and pull) XML ή JSON κωδικοποιημένα δεδομένα.

5.8.12 Geolocation

Υποστηρίζεται ο εντοπισμός της γεωγραφικής θέσης των συσκευών με την χρήση της υπηρεσίας Google Maps.

5.8.13 Πλάνα Τιμολόγησης

Η Carriots πλατφόρμα προσφέρει δωρεάν δοκιμαστική πρόσβαση σε όλες τις υπηρεσίες ώστε να μπορέσουν οι χρήστες να αναπτύξουν πλήρως τις δικές τους εφαρμογές IoT, οι οποίες βέβαια μπορούν να κλιμακωθούν μέσω κάποιας εμπορικής συμφωνίας [43].

5.8.14 Υποστηριζόμενες Γλώσσες Προγραμματισμού

Η Carriots πλατφόρμα έχει αναπτυχθεί (develop) βασισμένη στην τεχνολογία της γλώσσας Groovy [5] (<http://www.groovy-lang.org/>). Η Groovy είναι μια δυναμική γλώσσα για την εικονική μηχανή Java που βασίζεται στα δυνατά σημεία της Java, αλλά έχει πρόσθετα χαρακτηριστικά ισχύος εμπνευσμένα από γλώσσες όπως Python και Ruby.

5.8.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων

Η πλατφόρμα διαθέτει APIs για την ενσωμάτωση με εφαρμογές και πλατφόρμες τρίτων παρόχων όπως οι παρακάτω:

- Matlab
- Google Cloud Messaging (GCM)
- Facebook
- If This Then That (IFTTT)

5.8.16 Ενσωμάτωση Hardware πλατφορμών

Οποιοδήποτε εμπορικό, ανοιχτό (open) ή ειδικά σχεδιασμένο υλικό είναι δυνητικά συμβατό με την Carriots πλατφόρμα [16]. Το μόνο που χρειάζεται είναι η σύνδεση στο Internet για την επικοινωνία με ένα REST API. Το συμβατό δοκιμαζόμενο υλικό περιλαμβάνει:

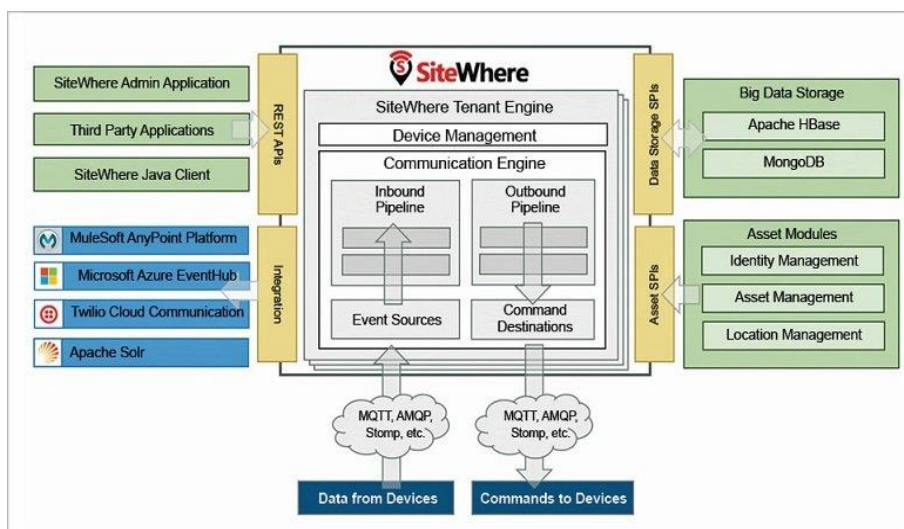
- Arduino
- Raspberry Pi
- Beagle Bone
- Cubie Board
- TST Gate
- TST Mote
- Cloudgate
- Nanode

5.8.17 Συμπεράσματα

Η Carriots IoT πλατφόρμα επιτρέπει την δημιουργία IoT Εφαρμογών εξοικονομώντας χρόνο και κόστος. Βασισμένη στην cloud αρχιτεκτονική Platform as a Service (PasS), παρέχονται λειτουργίες, όπως η απομακρυσμένη διαχείριση και ο έλεγχος των συσκευών, η καταγραφή δραστηριότητας (logging) η ενεργοποίηση προσαρμοσμένων ειδοποιήσεων και η εξαγωγή δεδομένων. Με την χρήση RESTful API λαμβάνουν τα δεδομένα από τις συσκευές και ενημερώνονται για την τρέχουσα κατάσταση αυτών. Η αποθήκευση των δεδομένων επιτυγχάνεται μέσω της NoSQL βάσης η οποία επιτρέπει την αποθήκευση μεγάλου όγκου δεδομένων (big data). Ωστόσο ως μειονέκτημα χαρακτηρίζεται το λιγότερο φιλικό προς τον χρήστη περιβάλλον της πλατφόρμας.

5.9 SiteWhere IoT platform

Το SiteWhere είναι μια πλατφόρμα ανοικτού κώδικα για την δημιουργία IoT εφαρμογών. Παρέχει μια υποδομή βασισμένη σε microservices, η οποία περιλαμβάνει τα βασικά χαρακτηριστικά που απαιτούνται για την κατασκευή και την ανάπτυξη εφαρμογών IoT [44]. Η υποδομής και τα microservices αναπτύσσονται μέσω του Kubernetes, επιτρέποντας την επιτόπου (on-premise) ανάπτυξη ή σε κάποια cloud υποδομή.



Εικόνα 5.15: Αρχιτεκτονική SiteWhere πλατφόρμας

Η SiteWhere πλατφόρμα περιλαμβάνει τα εξής χαρακτηριστικά όπως φαίνονται και στην Εικόνα 5.15:

- Εκτέλεση ταυτόχρονα πολλαπλών IoT εφαρμογών
- Σύνδεση συσκευών μέσω MQTT, AMQP, Stomp και άλλων πρωτοκόλλων
- Προσθήκη συσκευών μέσω REST υπηρεσιών και self-registration
- Ενσωμάτωση με third-party εφαρμογές όπως το Mule AnyPoint
- Η προεπιλεγμένη βάση δεδομένων είναι η MongoDB
- Απεικόνιση δεδομένων μέσω του Grafana
- Πλατφόρμα Twilio για ειδοποιήσεις μέσω SMS μέσω του cloud
- Apache Spark για ανάλυση ροής

Η πλατφόρμα Sitewhere, έχει σχεδιαστεί με τέτοιο τρόπο ώστε να μπορεί να επωφεληθεί από τις τεχνολογίες αιχμής προκειμένου να ανταπεξέρχεται αποτελεσματικά σε διάφορα IoT έργα. Αντί του να επιλέγει να χρησιμοποιεί μια μονολιθική αρχιτεκτονική, το SiteWhere περιλαμβάνει μια πλήρως καταναεμημένη αρχιτεκτονική χρησιμοποιώντας το Kubernetes ως βασική υποδομή και μια ποικιλία microservices για την κατασκευή των συστημάτων [1]. Αυτή η προσέγγιση επιτρέπει την προσαρμογή και την κλιμάκωση σε ένα τέτοιο επίπεδο, έτσι ώστε το σύστημα να μπορεί να προσαρμόζεται σε πολλές πιθανές περιπτώσεις χρήσης IoT. Η πλατφόρμα SiteWhere είναι χτισμένη με μια προσέγγιση πλαισίου (framework) χρησιμοποιώντας

σαφώς καθορισμένα APIs έτσι ώστε οι νέες τεχνολογίες να μπορούν εύκολα να ενσωματωθούν καθώς εξελίσσεται το IoT οικοσύστημα.

Επίσης, η SiteWhere πλατφόρμα αποτελείται από microservices βασισμένα σε JAVA, τα οποία είναι χτισμένα ως Docker images και έχουν αναπτυχθεί σε Kubernetes για την επίτευξη της εννοχρήστρωσης.

5.9.1 Integration - APIs

Οι περισσότερες βασικές λειτουργίες που παρέχονται από τα APIs του SiteWhere είναι προσβάσιμες εξωτερικά μέσω υπηρεσιών REST. Χρησιμοποιώντας τις REST υπηρεσίες, μια εξωτερική οντότητα μπορεί να δημιουργήσει, να προβάλει, να ενημερώσει ή να διαγράψει διάφορα αντικείμενα από το σύστημα. Όλες οι κλήσεις REST υπόκεινται σε έλεγχο ταυτότητας και χρησιμοποιούν το μοντέλο δικαιωμάτων για να επαληθεύσουν ότι ο πιστοποιημένος χρήστης είναι εξουσιοδοτημένος για τη λειτουργία. Το SiteWhere περιλαμβάνει μια λειτουργική έκδοση του Swagger μέσω του οποίου προσθέτετε μια διεπαφή χρήστη γύρω από τις υπηρεσίες REST [44]. Χρησιμοποιώντας τη διεπαφή Swagger, οι χρήστες μπορούν να εκτελέσουν αλληλεπιδραστικά τις κλήσεις REST έναντι μιας εκτελούμενης εγκατάστασης του SiteWhere.

5.9.2 Ασφάλεια δεδομένων

Για να επιτευχθεί ο έλεγχος ταυτότητας από τις IoT συσκευές προς την cloud υποδομή, οι συσκευές IoT μπορούν να καταχωρηθούν στο Sitewhere μέσω στοιχείων δομής JSON, τα οποία περιέχουν ένα αναγνωριστικό υλικού το οποίο είναι ένα μοναδικό αναγνωριστικό για τη συσκευή, ένα διακριτικό που καθορίζει συγκεκριμένα τον τύπο της συσκευής που καταχωρείται και ένα διακριτικό που καθορίζει τον τόπο στον οποίο θα συσχετιστεί η συσκευή [27]. Όταν μια συσκευή εγγράφεται με το Sitewhere η πλατφόρμα στέλνει πίσω μια απάντηση στη συσκευή που επιβεβαιώνει την εγγραφή της. Έτσι, το αναγνωριστικό χρησιμοποιείται για τον έλεγχο ταυτότητας της συσκευής.

5.9.3 Πρωτόκολλα Επικοινωνίας

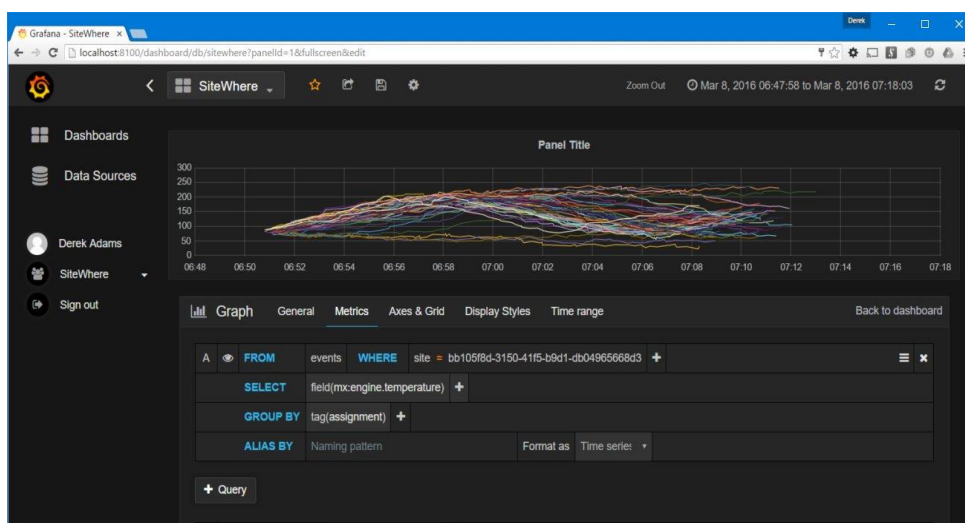
Η SiteWhere πλατφόρμα έχει σχεδιαστεί ώστε να παρέχει ευελιξία στον τρόπο που στέλνει και λαμβάνει τα δεδομένα από τις διάφορες συσκευές που απαρτίζουν την IoT υποδομή. Από προεπιλογή, η πλατφόρμα έχει ρυθμιστεί να επικοινωνεί μέσω του MQTT πρωτοκόλλου έτσι ώστε να επιτυγχάνεται υποστήριξη πολλών κωδικοποιήσεων καθώς και η αλληλεπίδραση με διαφόρους τύπους συσκευών. Τα πρωτόκολλα όπως το MQTT και το AMQP επιτρέπουν την αποδοτικότερη συνδεσιμότητα και την αμφίδρομη επικοινωνία. Επίσης υποστηρίζει την μορφή JSON για το payload στα περισσότερα πρωτόκολλα, συμπεριλαμβανομένων επικοινωνιών σε MQTT, AMQP και WebSockets [1].

5.9.4 Ανάλυση Δεδομένων

Η ανάλυση των δεδομένων επιτυγχάνεται μέσω της εφαρμογής Apache Spark. Το Apache Spark είναι ένας μηχανισμός επεξεργασίας δεδομένων μεγάλης κλίμακας. Η SiteWhere πλατφόρμα παρέχει μια σύνδεση για τα δεδομένα ροής στο Apache Spark μέσω σύνδεσης Hazelcast [44]. Το Apache Spark μπορεί να χρησιμοποιήσει τα δεδομένα από το SiteWhere για την εφαρμογή αλγορίθμων μηχανικής μάθησης (machine learning) ή την εξαγωγή δεδομένων.

5.9.5 Απεικόνιση των δεδομένων

Η SiteWhere IoT πλατφόρμα χρησιμοποιεί την εφαρμογή Grafana για την απεικόνιση των δεδομένων (Εικόνα 5.17). Το Grafana είναι ένα εργαλείο απεικόνισης δεδομένων το οποίο έχει ενσωματωμένη τη δυνατότητα σύνδεσης με την InfluxDB βάση δεδομένων [44]. Από την στιγμή που θα υπάρξει η κατάλληλη παραμετροποίηση για την διασύνδεση του InfluxDB και του Grafana, είναι πλέον δυνατή η δημιουργία ενός προσαρμοσμένου πίνακα ελέγχου για την προβολή των SiteWhere δεδομένων σε γραφική μορφή



Εικόνα 5.17: Απεικόνιση Δεδομένων

5.9.6 Hosting

Η SiteWhere πλατφόρμα καθώς και τα microservices αναπτύσσονται και διαχειρίζονται χρησιμοποιώντας την Kubernetes ως πλατφόρμα υποδομής. Αυτό επιτρέπει την ανάπτυξη σε σχεδόν οποιαδήποτε cloud υποδομή (π.χ. Microsoft Azure, AWS, Google Cloud, OpenShift). Επίσης παρέχεται και η δυνατότητα της on-premise ανάπτυξης, δηλαδή την εγκατάσταση σε κάποια τοπική υποδομή.

5.9.7 Push Notifications

Μόλις εγγραφούν στο σύστημα της πλατφόρμας, οι συσκευές μπορούν να αναφέρουν οποιονδήποτε αριθμό ή τύπο συμβάντων στην πλατφόρμα, η οποία με τη σειρά της αποθηκεύει τα συμβάντα. Οι τύποι συμβάντων περιλαμβάνουν ενημερώσεις τοποθεσίας, μετρήσεις αισθητήρων και άλλα δεδομένα που έχουν ληφθεί ή ειδοποιήσεις (alerts) σε περίπτωση έκτακτων συμβάντων [26]. Οι συσκευές έχουν επίσης τη δυνατότητα να λαμβάνουν διάφορες εντολές που αποστέλλονται από την πλατφόρμα SiteWhere.

5.9.8 Ενεργοποίηση Συμβάντων (Trigger Actions)

Μέσω της πλατφόρμας αποστέλλονται ειδοποιήσεις προς τις συσκευές, ώστε να υποδείξουν ότι κάποιο γεγονός έλαβε μέρος. Μια ειδοποίηση περιλαμβάνει έναν τύπο ειδοποίησης, ένα μήνυμα, ένα επίπεδο ειδοποίησης και άλλες πληροφορίες που παρέχουν το πλαίσιο ως προς την κατάσταση που συνέβη. Οι ειδοποιήσεις μπορούν να χρησιμοποιηθούν για την ενεργοποίηση άλλων συμβάντων [24]. Για παράδειγμα, ένας συναγερμός πυρκαγιάς μπορεί να αποστείλει μια ειδοποίηση ανίχνευσης καπνού.

5.9.9 Αποθήκευση Δεδομένων

Σχετικά με την αποθήκευση των δεδομένων μέσω της SiteWhere πλατφόρμας υποστηρίζονται οι εξής τεχνολογίες:

- **Την MongoDB** η οποία είναι μια βάση δεδομένων NoSQL που μπορεί να χρησιμοποιηθεί για την αποθήκευση των δεδομένων στην SiteWhere πλατφόρμα. Είναι πολύ γρήγορη και εύκολη στην εγκατάσταση, έχει εξαιρετική απόδοση και δεν απαιτεί τους υπολογιστικούς πόρους της HBase.
- **Την Apache HBase** η οποία είναι μία κατακευκτική, μη σχεσιακή βάση δεδομένων. Η πλατφόρμα SiteWhere χρησιμοποιεί ένα προσαρμοσμένο σχήμα της συγκεκριμένης βάσης δεδομένων, για την αποθήκευση διαφόρων συμβάντων των συσκευών, σε συγκεκριμένα χρονικά διαστήματα, επιτρέποντας την γρήγορη πρόσβαση σε αυτά με βάση την ακριβή στιγμή που έγιναν.
- **Την influxDB**, η πλατφόρμα SiteWhere χρησιμοποιεί μια υβριδική προσέγγιση όπου τα δεδομένα διαχείρισης συσκευών αποθηκεύονται στην MongoDB βάση ενώ τα δεδομένα συμβάντων των συσκευών αποθηκεύονται στο InfluxDB. Με αυτήν την προσέγγιση, εργαλεία όπως το Grafana μπορούν να χρησιμοποιηθούν για την απεικόνιση των δεδομένων του SiteWhere.

5.9.10 Αποθήκευση Blob

Δεν γίνεται κάποια αναφορά για την υποστήριξη μη δομημένων δεδομένων.

5.9.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)

Η πλατφόρμα παρέχει ένα SDK Android το οποίο είναι διαθέσιμο στη διεύθυνση github.com/site-where/site-where-android-sdk, και παρέχει βασικές λειτουργίες και υπηρεσίες επικοινωνίας όπως MQTT και REST. Οι εφαρμογές Android μπορούν εύκολα να κατασκευαστούν χρησιμοποιώντας αυτό το SDK. Επίσης ένας Java agent είναι διαθέσιμος στο github.com/site-where/site-where-tools, το οποίο παρέχει ένα cli πρόγραμμα για την δημιουργία μιας προσαρμοσμένης επικοινωνίας με την πλατφόρμα [44].

5.9.12 Geolocation

Η πλατφόρμα SiteWhere υποστηρίζει την αποθήκευση τοποθεσιών που σχετίζονται με μια συνδεδεμένη συσκευή ως δεδομένα συμβάντων (event data). Αυτό βοηθά στη γεωγραφική χαρτογράφηση της συσκευής κάθε φορά που πραγματοποιείται μία αλλαγή στην τοποθεσία. Η αποστολή της τοποθεσίας της συσκευής είναι της μορφής JSON, όπως ακριβώς εμφανίζεται παρακάτω:

```
{
  "type": "DeviceLocation",
  "originator": "device",
  "deviceToken": "mydevicetoken",
  "request": {
    "latitude": "33.75",
    "longitude": "-84.39",
    "elevation": "1",
    "updateState": true,
    "eventDate": "2018-11-03T19:40:03.390Z"
  }
}
```

5.9.13 Πλάνα Τιμολόγησης

Η SiteWhere πλατφόρμα είναι ανοικτού κώδικα και παρέχεται δωρεάν τόσο για ιδιωτική όσο και για εμπορική χρήση. Επίσης η ομάδα του SiteWhere προσφέρει δωρεάν βασική υποστήριξη [44].

5.9.14 Υποστηριζόμενες Γλώσσες Προγραμματισμού

Δεν γίνεται αναφορά στην βιβλιογραφία.

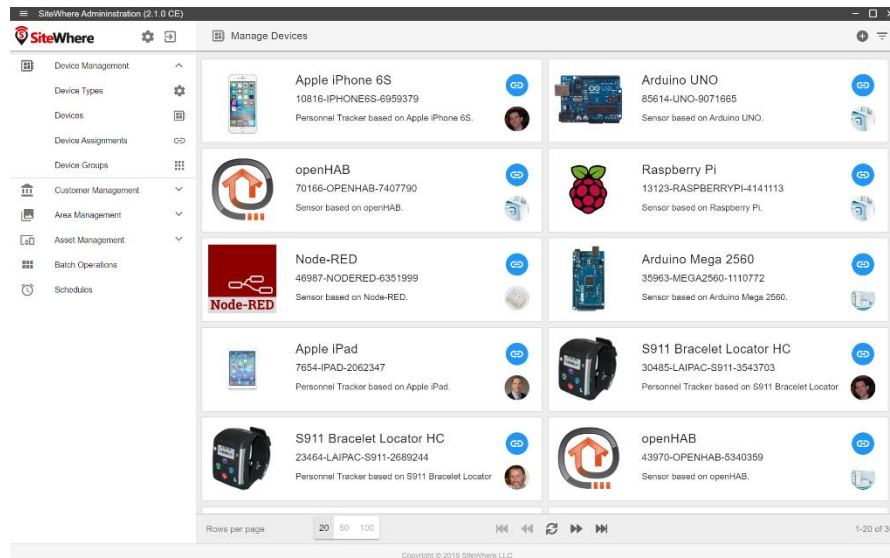
5.9.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων

Η πλατφόρμα μπορεί να λειτουργήσει και ως middleware έτσι ώστε να καταφέρει να ωθήσει και να τραβήξει (push and pull) δεδομένα από εξωτερικά συστήματα [24]. Αυτό επιτρέπει στο SiteWhere να ενσωματώνει δεδομένα που συλλαμβάνονται από άλλα συστήματα με αποτέλεσμα να επιτυγχάνονται λειτουργίες όπως η ανάλυση δεδομένων. Κάποιες από αυτές τις πλατφόρμες είναι οι:

- Mule AnyPoint
- openHAB
- Apache Spark
- Grafana

5.9.16 Ενσωμάτωση Hardware Πλατφορμών

Η SiteWhere πλατφόρμα παρέχει κιτ ανάπτυξης λογισμικού (SDKs), μέσω των οποίων επιτυγχάνεται η ενσωμάτωση των συσκευών από δημοφιλείς πλατφόρμες υλικού όπως απεικονίζονται στην παρακάτω Εικόνα 5.17. Αυτό επιτρέπει στους χρήστες να εστιάζουν στην ανάπτυξη IoT εφαρμογών αντί να αναλώνουν χρόνο ως προς την αντιμετώπιση προβλημάτων που σχετίζονται με την υποδομή (infrastructure) [44].



Εικόνα 5.17 Ενσωμάτωση με πλατφόρμες υλικού

5.9.17 Συμπεράσματα

Η SiteWhere είναι μία ακόμη open-source πλατφόρμα, η οποία επιτρέπει την ταχεία ανάπτυξη εφαρμογών για το Διαδίκτυο των αντικειμένων. Επίσης, υποστηρίζει πολλαπλά πρωτόκολλα επικοινωνίας, συμπεριλαμβανομένων των MQTT, AMQP, Stomp και άλλων. Επιπλέον, επιτρέπει την ενσωμάτωση υπηρεσιών τρίτων παρόχων όπως το Azure, το Apache καθώς και την ενσωμάτωση με πλατφόρμες υλικού όπως το Arduino, Raspberry Pi, Android, iOS, Intel και άλλες πολλές. Πέραν των παραπάνω παρέχει δυνατότητες όπως η διαχείριση των IoT συσκευών, αποθήκευση μεγάλου όγκου δεδομένων, υποστήριξη επικοινωνίας, καθώς και ανάλυση των πληροφοριών που συλλέγονται από τις συσκευές.

5.10 Thinger.io Platform

Η Thinger.io είναι μια IoT πλατφόρμα ανοιχτού κώδικα για το Διαδίκτυο των αντικειμένων, η οποία παρέχει μια cloud υποδομή για την διασύνδεση των αντικειμένων ενός IoT περιβάλλοντος [49]. Η συγκεκριμένη πλατφόρμα στοχεύει κυρίως σε ένα εύρος χρηστών οι οποίοι θέλουν να δημιουργήσουν IoT λύσεις από την αρχή (from scratch). Η πλατφόρμα μπορεί να αναπτυχθεί ελεύθερα σε οποιαδήποτε cloud υποδομή χρησιμοποιώντας την έκδοση Ubuntu Core, καθώς και να συνδεθεί με άλλους μικροελεγκτές ή συστήματα σε πραγματικό χρόνο μέσω ενός API REST.

Τα κύρια χαρακτηριστικά της πλατφόρμας είναι η πλήρης διαχείριση συσκευών, η αλληλεπίδραση την οποία παρέχει μέσω του API, διάφορα dashboards για απεικόνιση δεδομένων σε πραγματικό χρόνο καθώς και η ασφάλεια για την αποθήκευση και την εξαγωγή των δεδομένων.

5.10.1 Integration - APIs

Ένα από τα κύρια χαρακτηριστικά της πλατφόρμας Thinger.io είναι ότι επιτρέπει την ανακάλυψη των πόρων που διαθέτει η συσκευή η οποία συνδέεται με την πλατφόρμα. Ένας πόρος μπορεί να είναι μια ένδειξη ενός αισθητήρα, όπως η θερμοκρασία, η υγρασία ή η πίεση. Ένας πόρος μπορεί επίσης να είναι οποιοδήποτε στοιχείο που μπορεί να προκαλέσει μία ενέργεια, όπως ένα ρελέ ή ένας κινητήρας. Αλλά γενικά, κάθε πόρος συσκευής ισοδυναμεί σαν μια λειτουργία επανάκλησης η οποία μπορεί να καλείται μέσω του API Rest [30].

5.10.2 Ασφάλεια Δεδομένων

Η αλληλεπίδραση και η επικοινωνία των συσκευών με την πλατφόρμα θα πρέπει με κάποιο τρόπο να πιστοποιηθεί και να εξασφαλιστεί ότι κανείς μη εξουσιοδοτημένος χρήστης έχει πρόσβαση σε αυτή. Αυτό επιτυγχάνεται με την χρήση ενός διακριτικού πρόσβασης (access token) το οποίο παράγεται από τον συνδυασμό του ονόματος χρήστη (username) και του κωδικού πρόσβασης (password). Αυτό το είδος εξουσιοδότησης παρέχει πρόσβαση σε όλους τους πόρους του λογαριασμού ενός χρήστη, ώστε να μπορεί να ρυθμίσει τις παραμέτρους των διαφόρων συνδεδεμένων συσκευών κ.λπ. [30]. Ωστόσο, η χρονική ισχύς αυτής της εξουσιοδότησης είναι αρκετά μικρή αλλά ανανεώνεται αυτόματα από το πρόγραμμα περιήγησης.

5.10.3 Πρωτόκολλα Επικοινωνίας

Οι διάφορες συσκευές μπορούν να διασυνδεθούν με την πλατφόρμα μέσω των τριών πρωτοκόλλων που υποστηρίζει, τα οποία είναι το HTTP, το CoAP και το MQTT. Κάθε ένας από αυτά τα πρωτόκολλα έχει τα υπέρ και τα κατά [30].

HTTP: Το συγκεκριμένο πρωτόκολλο τρέχει πάνω από TCP επικοινωνία και υπάρχουν αρκετές διαθέσιμες βιβλιοθήκες. Ωστόσο, το http εισάγει περισσότερα overhead.

CoaP: Το CoAP επικοινωνεί πάνω από το UDP πρωτόκολλο, και είναι σχεδιασμένο ειδικά για IoT εφαρμογές καθώς επίσης εισάγει ελάχιστα overhead. Το συγκεκριμένο πρωτόκολλο παρέχεται από την πλατφόρμα σε συνδυασμό με το DTLS (Datagram Transport Layer Security)

MQTT: Το MQTT επικοινωνεί πάνω από το TCP. Το MQTT είναι επίσης ελαφρύτερο από το HTTP, αλλά δεν είναι δυνατό να αναπαραχθούν όλες οι λειτουργίες που υποστηρίζονται από τα άλλα πρωτόκολλα.

5.10.4 Ανάλυση – Απεικόνιση Δεδομένων

Ένας πίνακας ελέγχου (dashboard) είναι ένα γραφικό περιβάλλον εργασίας μέσω του οποίου δίνεται η δυνατότητα στον χρήστη να διαχειρίζεται την εμφάνιση με την οποία απεικονίζονται οι πληροφορίες. Μέσω της Thinger.io πλατφόρμας, μπορούν οι χρήστες να διαμορφώσουν τους πίνακες ελέγχου με διαφορετικά γραφικά στοιχεία, να διαμορφώσουν τη διάταξη, τη διάσταση, το χρώμα και τις πηγές από τις οποίες αντλούνται τα δεδομένα.[49] Οι πίνακες ελέγχου (dashboards) μπορούν να εμφανίζουν πληροφορίες σε πραγματικό χρόνο από τις συσκευές σας οι οποίες είναι συνδεδεμένες με την πλατφόρμα (χρησιμοποιώντας websockets) ή να χρησιμοποιούν ιστορικές πληροφορίες οι οποίες βρίσκονται αποθηκευμένες (data buckets).

5.10.6 Hosting

Η εγκατάσταση ενός Thinger.io διακομιστή (server) μπορεί να πραγματοποιηθεί με την χρήση ενός Ubuntu client με πακέτα Ubuntu Snap για διαφορετικές πλατφόρμες όπως x86, x64, arm64 ή armhf [24]. Επίσης, μπορεί να εγκατασταθεί στην Amazon Cloud υποδομή (AWS), με την χρήση του Amazon Machine Image (AMI).

5.10.7 Push Notifications

Υποστηρίζεται μέσω REST API.

5.10.8 Ενεργοποίηση Συμβάντων (Trigger Actions)

Η Thinger.io πλατφόρμα επιτρέπει την δυνατότητα διασύνδεσης με πλατφόρμες τρίτων παρόχων όπως την IFTTT (If This Then That) η οποία είναι μία από τις πιο ενδιαφέρουσες εφαρμογές αυτοματοποίησης

συμβάντων βάσει των συνθηκών που έχει ορίσει ο χρήστης. Για παράδειγμα, η μεταβολή της θερμοκρασίας στην περιοχή όπου είναι εγκατεστημένη η IoT συσκευή, μπορεί να ενεργοποιήσει την αποστολή ενός ηλεκτρονικού μηνύματος στον χρήστη, πράξη η οποία υποδηλώνει ένα συμβάν.

5.10.9 Αποθήκευση Δεδομένων

Η Thinger.io πλατφόρμα χρησιμοποιεί data buckets "δοχεία δεδομένων" τα οποία είναι ένα είδος εικονικής αποθήκευσης, στα οποία μπορούν να διατηρηθούν πληροφορίες με χρονολογική σειρά, όπως η θερμοκρασία ή η υγρασία. Επίσης είναι δυνατό να χρησιμοποιηθούν και για την αποθήκευση οποιοδήποτε άλλου συμβάντος, όπως για παράδειγμα η ανίχνευση κίνησης (ανοίγει η πόρτα του γκαράζ), προειδοποιήσεις για τις τιμές της θερμοκρασίας, και ούτω καθεξής [49]. Αυτά τα δεδομένα μπορούν να χρησιμοποιηθούν για την καταγραφή πληροφοριών σε πίνακες ελέγχου (dashboards) ή μπορούν ακόμη να εξαχθούν σε διαφορετικές μορφές για offline επεξεργασία. Για την δημιουργία Data bucket απαιτούνται διάφοροι παράμετροι όπως οι παρακάτω:

- **Bucket Id:** Μοναδικό αναγνωριστικό.
- **Bucket name:** Χρήση ενός αντιπροσωπευτικού ονόματος
- **Bucket description:** περιγραφή με περισσότερες λεπτομέρειες
- **Enabled:** Ενεργοποίηση ή απενεργοποίηση της λειτουργίας

5.10.10 Αποθήκευση Blob

Μέσω της πλατφόρμας παρέχεται η υποστήριξη αποθήκευσης και διαχείρισης μη δομημένων δεδομένων, όπως αρχεία, φωτογραφίες και εικόνες. Επισημαίνεται, ότι στην ορολογία του SQL, τα μη δομημένα δεδομένα έχουν την ονομασία BLOB (δυναμικό μεγάλο αντικείμενο).

5.10.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)

Υποστηρίζεται μόνο Android SDK.

5.10.12 Geolocation

Ο εντοπισμός θέσης των IoT συσκευών γίνεται με την χρήση widgets και την υπηρεσία Google Maps και τα δεδομένα είναι πραγματικού χρόνου (real-time).

5.10.13 Πλάνα Τιμολόγησης

Η πλατφόρμα διατίθεται για δωρεάν χρήση με αρκετούς περιορισμούς όπως ο αριθμός των συνδεδεμένων συσκευών. Για την προσθήκη επιπλέον συσκευών και λειτουργιών υπάρχει χρέωση σε μηνιαία βάση.

5.10.14 Υποστηριζόμενες Γλώσσες Προγραμματισμού

Η γλώσσα η οποία υποστηρίζεται ως server και client side είναι η C++.

5.10.15 Ενσωμάτωση με πλατφόρμες τρίτων παρόχων

Η Πλατφόρμα διαθέτει ένα ανοιχτό API και έναν διαχειριστή τελικών σημείων που απλοποιεί πολύ αυτή τη διαδικασία.

5.10.16 Ενσωμάτωση Hardware πλατφορμών

Η συγκεκριμένη πλατφόρμα χαρακτηρίζεται ως "hardware agnostic" δηλαδή δεν απαιτείται κάποια εξάρτηση από συγκεκριμένο υλικό για την διασύνδεση της συσκευής με την πλατφόρμα. Επομένως, είναι δυνατή η σύνδεση οποιασδήποτε συσκευής όπως Arduino, Raspberry Pi, Sigfox και άλλων με την πλατφόρμα [24]. Ως "hardware agnostic" προσφέρει ένα υψηλό επίπεδο συμβατότητας καθώς επίσης διαθέτει ένα μεγάλο εύρος βιβλιοθηκών για τις συσκευές αυτές.

5.10.17 Συμπεράσματα

Το κύριο πλεονέκτημα από τη χρήση αυτής της πλατφόρμας, εκτός από το γεγονός ότι είναι ανοιχτού κώδικα, είναι η δυνατότητα αμφίδρομης επικοινωνίας με τις συσκευές, σε πραγματικό χρόνο, χρησιμοποιώντας REST-APIs διεπαφές. Επιπλέον, η Thinger.io πλατφόρμα παρέχει βιβλιοθήκες για τη σύνδεση διάφορων IoT συσκευών όπως το Arduino, το ESP8266, το ESP32, το LinkitOne, το Texas Instruments CC3200, το Libellium Waspmotes, το Raspberry Pi κ.λπ. Οι βιβλιοθήκες αυτές παρέχουν έναν ολοκληρωμένο τρόπο σύνδεσης των συσκευών και την αποστολή πληροφοριών στο cloud, χωρίς την εμπλοκή σύνθετων πρωτοκόλλων IoT.

5.11 Ubidots Platform

Το Ubidots (<https://ubidots.com/platform/>) αποτελεί άλλη μια ενδιαφέρουσα IoT πλατφόρμα που παρέχει πολλές υπηρεσίες. Πρώτα από όλα, υποστηρίζει αρκετές συσκευές, όπως το Arduino, το Raspberry, το Particle, το Espressif και πολλά άλλα. Επιπλέον, παρέχει διαφορετικά είδη υπηρεσιών, τα οποία κυμαίνονται από τη συνδεσιμότητα των συσκευών με την πλατφόρμα μέχρι την απεικόνιση δεδομένων και άλλα όπως:

- SDK – APIs, Το Ubidots παρέχει ένα SDK για διαφορετικές συσκευές για να απλοποιήσει τη διαδικασία ενσωμάτωσης μεταξύ της ίδιας της συσκευής και της πλατφόρμας [45]. Επιπλέον, έχει πολλά APIs τα οποία μπορούν να χρησιμοποιηθούν για να αλληλοεπιδράσουν με αυτήν την πλατφόρμα.
- Υποστηρίζονται αρκετά πρωτόκολλα, όπως το MQTT και το HTTP.
- Συνθετικές μεταβλητές που εφαρμόζουν μαθηματικούς τύπους στα δεδομένα.
- Αποθήκευση δεδομένων.
- Απεικόνιση – Ανάλυση δεδομένων.

5.11.1 Integration – APIs

Η σύνδεση και η επικοινωνία των διαφόρων συσκευών με την cloud υποδομή που παρέχει η Ubidots πλατφόρμα μπορεί να επιτευχθεί με την χρήση πάνω από 200 αποδεδειγμένων από τους χρήστες βιβλιοθηκών καθώς και με την χρήση SDKs, μέσω διαφόρων πρωτοκόλλων επικοινωνίας τα οποία υποστηρίζονται από την πλατφόρμα όπως το HTTP και το MQTT [18].

Το HTTP είναι ένα πρωτόκολλο αίτησης / απόκρισης (request - response), το οποίο σημαίνει ότι κάθε αίτημα που γίνεται από τον χρήστη απαντάται από το διακομιστή. Αυτή η απάντηση περιλαμβάνει έναν αριθμό (κωδικό απόκρισης) και ένα σώμα (body). Για παράδειγμα, όταν εκτελείται ένα αίτημα για την ανάκτηση ενός αρχείου σε μια ιστοσελίδα δημιουργείται ένα αίτημα GET. Εάν το αίτημα είναι σωστό, ο διακομιστής θα επιστρέψει συνήθως έναν κωδικό απάντησης 200, μαζί με το αρχείο που ζητήθηκε (σώμα). Κάτω από την ίδια λογική, όταν μια συσκευή στέλνει ένα αίτημα GET στον εξυπηρετητή του Ubidots (δηλ. Ζητώντας την τελευταία τιμή μιας μεταβλητής), τότε ο διακομιστής στέλνει μια απάντηση με την κατάσταση του αιτήματος (κωδικός απόκρισης) και το αρχείο που ζητήθηκε (body). Το Ubidots δέχεται δεδομένα με την μορφή JavaScript Object Notation ή JSON [45]. Το JSON είναι ένας τύπος δεδομένων HTTP, πρόκειται ουσιαστικά για μία συλλογή ζευγών ονόματος / τιμής. Ένα παράδειγμα ενός τύπου δεδομένων JSON που δέχεται το Ubidots μπορεί να αναφερθεί παρακάτω:

```
{"temperature": {"value":10, "timestamp": 1534881387000, "context": {"machine": "1st floor"}}}
```

Ένα τυπικό αίτημα HTTP στην Ubidots πλατφόρμα θα πρέπει να οριστεί ως εξής:

```
POST {PATH} HTTP/1.1<CR><LN>  
Host: {HOST}<CR><LN>
```

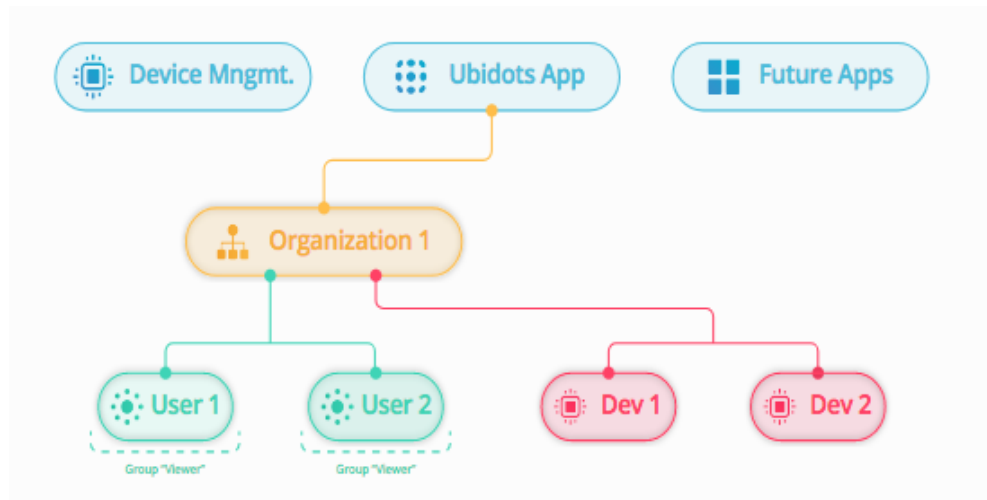
```
User-Agent: {USER_AGENT}<CR><LN>
X-Auth-Token: {TOKEN}<CR><LN>
Content-Type: application/json<CR><LN>
Content-Length: {PAYLOAD_LENGTH}<CR><LN><CR><LN>
{PAYLOAD}
<CR><LN>
```

Μέσω του Ubidots REST API επιτρέπεται η προγραμματική πρόσβαση σε διάφορους πόρους, συμπεριλαμβανομένων των συσκευών, μεταβλητών, τιμών και πολλών άλλων. Για την αλληλεπίδραση με το Ubidots REST API, συνιστάται η χρήση ενός HTTPS client (Curl, HTTPie, Postman).

5.11.2 Ασφάλεια

Η ασφάλεια αποτελεί ένα σημαντικό ζήτημα σχετικά με τα δεδομένα τα οποία αποστέλλονται από τους αισθητήρες στην Ubidots πλατφόρμα. Η Ubidots είναι μια βελτιωμένη πλατφόρμα που έχει σχεδιαστεί με γνώμονα την εύκολη χρήση και γρήγορη ανάπτυξη με βέλτιστες πρακτικές ασφάλειας [45] ενσωματωμένες όπως:

- HTTP μέσω SSL κρυπτογράφησης, μέσω του SSL δημιουργείται ασφαλή κρυπτογραφημένη σύνδεση μεταξύ των διακομιστών και των συσκευών.
- MQTT μέσω TLS κρυπτογράφησης, Το Ubidots υποστηρίζει το πρωτόκολλο MQTT, για την μεταφορά μηνυμάτων σε συνδυασμό κρυπτογράφησης TLS. Το TLS (Security Layer Security) παρέχει ένα ασφαλές κανάλι επικοινωνίας μεταξύ ενός πελάτη και ενός διακομιστή.
- Έλεγχος ταυτότητας βάσει αναγνωριστικού (Token-based Authentication).
- Διαχείριση πολλαπλών χρηστών, μέσω της άδειας (license) που παρέχεται από την Ubidots, επιτρέπεται η δυνατότητα του ελέγχου του ποιος μπορεί να αποκτήσει πρόσβαση στα δεδομένα της συσκευής σε διαφορετικά επίπεδα όπως απεικονίζεται στην Εικόνα 5.18.



Εικόνα 5.18 Έλεγχος πρόσβασης χρηστών

5.11.3 Πρωτόκολλα Επικοινωνίας

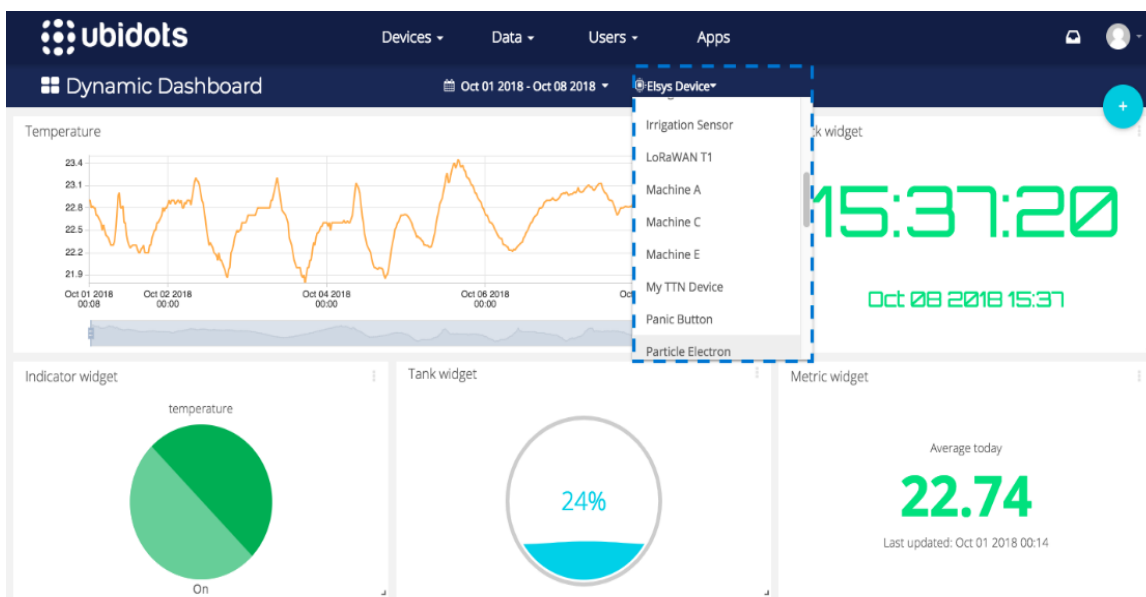
Η Ubidots πλατφόρμα για την αποστολή και ανάκτηση δεδομένων από τις συσκευές υλικού χρησιμοποιεί το HTTP και το MQTT ως πρωτόκολλα επικοινωνίας [18].

5.11.4 Απεικόνιση Δεδομένων

Τα Dashboards (στατικά και δυναμικά) είναι διεπαφές χρηστών για την οργάνωση και παρουσίαση των δεδομένων μιας συσκευής και των στοιχείων που προκύπτουν από τα δεδομένα. Τα Dashboards περιέχουν widget που εμφανίζουν τα δεδομένα ως διαγράμματα, δείκτες, στοιχεία ελέγχου, πίνακες, γραφήματα και άλλα μεγέθη, σχήματα και φόρμες. Μέσω της πλατφόρμας Ubidots ο χρήστης μπορεί να δημιουργήσει στατικά και δυναμικά dashboards [45].

Τα στατικά dashboards χρησιμοποιούνται για την απεικόνιση δεδομένων από προκαθορισμένες συσκευές και μεταβλητές. Οι στατικοί πίνακες (dashboards) ελέγχου εμφανίζουν ανά πάσα στιγμή την ίδια συσκευή και δεδομένα.

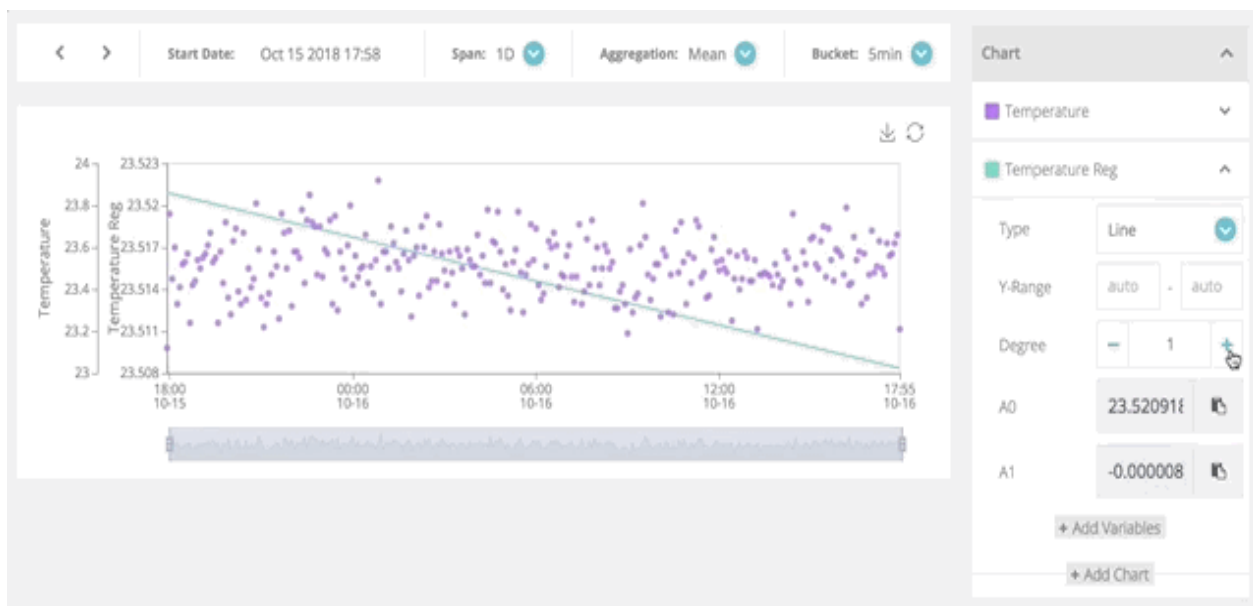
Τα δυναμικά Dashboards χρησιμοποιούν τεχνολογία master / slave για την απεικόνιση δεδομένων από διαφορετικές συσκευές και μεταβλητές, χρησιμοποιώντας ένα αναπτυσσόμενο μενού στη γραμμή ρυθμίσεων (Εικόνα 5.19). Επιλέγοντας διαφορετικές συσκευές από τη γραμμή ρυθμίσεων, θα γίνει ανανέωση του δυναμικού πίνακα ελέγχου (dashboard) με την υποκείμενη συσκευή και τις μεταβλητές που συγκεντρώνουν την εμφάνιση κάθε widget σύμφωνα με τη συσκευή που έχει επιλεγθεί.



Εικόνα 5.19: Δυναμικό Dashboard απεικόνισης δεδομένων

5.11.5 Ανάλυση Δεδομένων

Μέσω της εφαρμογής Data Explorer (Εικόνα 5.20) η οποία παρέχεται από τη Ubidots, είναι μια συλλογή εργαλείων ανάλυσης δεδομένων που βοηθούν στον εντοπισμό και την ανάλυση τάσεων, ανωμαλιών και συσχετισμών μεταξύ των μεταβλητών σε μια εφαρμογή. Μέσω του Data Explorer ο χρήστης μπορεί να πραγματοποιήσει:



Εικόνα 5.20: Ανάλυση Δεδομένων

- **Ιστορική πλοήγηση δεδομένων (Historical data navigation):** Ανάλυση δεδομένων σε διαφορετικές χρονικές περιόδους
- **Ανάλυση συσχέτισης (Correlation Analysis):** Σύγκριση δύο ή περισσότερων μεταβλητών στον ίδιο πίνακα ή σε πολλά διαγράμματα
- **Ανάλυση παλινδρόμησης (Regression Analysis):** Εφαρμογή αλγορίθμων παλινδρόμησης για των γραμμών τάσεων και πληροφοριών.
- **Ανίχνευση ανωμαλιών (Anomaly Detection)**

5.11.6 Hosting

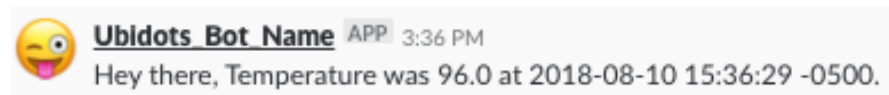
Η ανάπτυξη και η διαχείριση της πλατφόρμας πραγματοποιείται μέσω της cloud υποδομής, ωστόσο παρέχεται και η δυνατότητα της on-premise ανάπτυξης, δηλαδή της εγκατάστασης σε κάποια τοπική υποδομή μέσω ιδιόκτητων διακομιστών [45].

5.11.7 Push Notifications

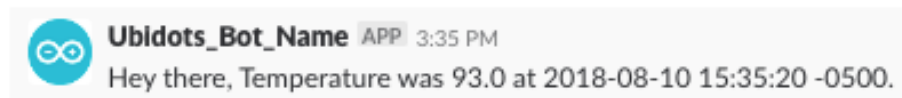
Μέσω της πλατφόρμας υποστηρίζεται η δυνατότητα δημιουργίας συμβάντων, και η αποστολή ειδοποιήσεων όταν κάποιο γεγονός λάβει μέρος. Οι προεγκατεστημένες ενσωματώσεις της Ubidots πλατφόρμας περιλαμβάνουν:

- Ειδοποιήσεις ηλεκτρονικού ταχυδρομείου.
- Ειδοποιήσεις SMS.
- Ειδοποιήσεις φωνητικών κλήσεων.
- Ειδοποιήσεις "χαλάρωσης" (Slack Notifications), η αποστολή ειδοποιήσεων και ενημερώσεων μέσω του Slack παρέχει έγκαιρη ενημέρωση σε όλα τα μέλη μιας ομάδας εργασίας την ίδια στιγμή (Εικόνα 5.21).

Slack Events with Emoji:



Slack Events with Image:



Εικόνα 5.21: Slack Notifications

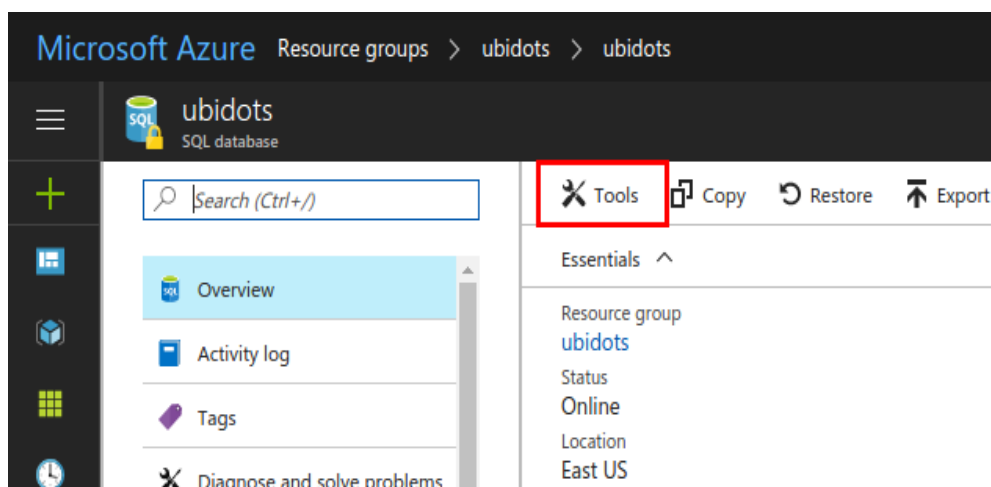
- Ειδοποίηση Geofence, το Geofencing είναι ένα λογισμικό το οποίο, μέσω GPS, ορίζει γεωγραφικά όρια. Όταν μια συσκευή εισέρχεται ή εξέρχεται από μια καθορισμένη περιοχή, μπορεί να ενεργοποιηθεί μια ειδοποίηση (ή συμβάν) χρησιμοποιώντας την Ubidots πλατφόρμα.

5.11.8 Ενεργοποίηση Συμβάντων (Trigger Actions)

Στην συγκεκριμένη πλατφόρμα παρέχεται η δυνατότητα ενεργοποίησης συμβάντων όταν καθορισμένα όρια (thresholds) έχουν ξεπεραστεί. Η διαδικασία βασίζεται στην δομή IFTTT (*if this, then that*) [18].

5.11.9 Αποθήκευση Δεδομένων

Μέσω της πλατφόρμας παρέχεται η δυνατότητα για την λήψη και τον υπολογισμό μεγάλου όγκου αποθηκευμένων δεδομένων για οποιαδήποτε χρονικό διάστημα είτε πρόκειται μιας ημέρας, ενός μήνα ή ενός λεπτού. Με το πρότυπο κράτησης δεδομένων 2 ετών, το Ubidots Cloud δίνει στους χρήστες ένα χώρο αποθήκευσης και οργάνωσης δεδομένων για πρόσθετες πληροφορίες, όπως ανίχνευση ανωμαλιών ή προγνωστική συντήρηση. Επίσης, μέσω της πλατφόρμας παρέχεται η δυνατότητα διαχείρισης των αποθηκευμένων δεδομένων μέσω της υπηρεσίας που παρέχεται από την πλατφόρμα Azure IoT (Εικόνα 5.18) . Η αποθήκευση γίνεται σε μια SQL βάση δεδομένων [45].



Εικόνα 5.18 Αποθήκευση δεδομένων στο Azure IoT

5.10.10 Αποθήκευση Blob

Δεν αναφέρεται στην βιβλιογραφία ή στην επίσημη ιστοσελίδα της πλατφόρμας η υποστήριξη Blob data.

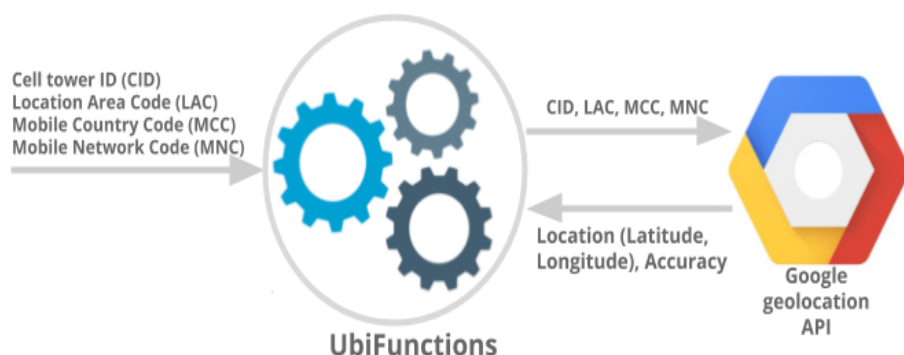
5.11.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)

Μέσω της πλατφόρμας παρέχονται περισσότερες από 200 βιβλιοθήκες και SDKs μέσω των οποίων επιτυγχάνεται η σύνδεση των συσκευών με την cloud υποδομή της πλατφόρμας, με την χρήση πρωτοκόλλων όπως το HTTP, MQTT, TCP και UDP. Μερικά από τα SDKs περιλαμβάνουν [45]:

- **To Ubidots Android GPS Tracker SDK**, παρέχει την δυνατότητα στους προγραμματιστές να ενσωματώσουν δεδομένα θέσης
- **To Ubidots Python SDK**, με στόχο την αυτοματοποίηση εφαρμογών

5.11.12 Geolocation

Στην συγκεκριμένη πλατφόρμα, η αναγνώριση ή η εκτίμηση της πραγματικής γεωγραφικής θέσης ενός αντικειμένου επιτυγχάνεται με τον συνδυασμό του Geolocation API της Google και του UbiFunction όπως φαίνεται στην Εικόνα 5.18.



Εικόνα 5.18: Γεωγραφική θέση ενός αντικειμένου

Μέσω του συγκεκριμένου API εξαλείφεται η ανάγκη για την χρήση ενσωματωμένου τσιπ GPS πάνω στην συσκευή. Από την άλλη με τον όρο UbiFunction αναφερόμαστε σε ένα κομμάτι κώδικα οριζόμενο από τον χρήστη το οποίο εκτελείται κατ' απαίτηση κάνοντας μια αίτηση HTTP GET είτε POST [48]. Ουσιαστικά με την χρήση των UbiFunctions, επιτυγχάνεται η λήψη δεδομένων από APIs τρίτων παρόχων.

5.11.13 Πλάνα τιμολόγησης

Η τιμολογιακή πολιτική της πλατφόρμας σχετίζεται με τον αριθμό των συνδεδεμένων συσκευών καθώς και του όγκου των δεδομένων που ανταλλάσσονται μεταξύ της πλατφόρμας και των συσκευών. Πρόκειται για μία ευέλικτη τιμολόγηση, βασισμένη στις ανάγκες του IoT έργου η οποία κατηγοριοποιείται σε τέσσερα διαφορετικά επίπεδα (Developer, IoT Lab, Industrial, Scale) συσκευών. Για την προσθήκη επιπλέον συσκευών και λειτουργιών υπάρχει χρέωση σε μηνιαία βάση.

5.11.14 Υποστηριζόμενες Γλώσσες Προγραμματισμού

Η πλατφόρμα διαθέτει την υποστήριξη γλωσσών προγραμματισμού όπως Python και C, C++, Javascript και άλλων με στόχο την ανάπτυξη εφαρμογών που σχετίζονται με την πλευρά των χρηστών.

5.11.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων

Η πλατφόρμα Ubidots επιτρέπει στους χρήστες την δυνατότητα προσαρμοσμένων APIs μέσω του περιβάλλοντος Node.js, κάτι το οποίο επεκτείνει τις λύσεις των συνδέσεων πέρα από τα όρια της αρχιτεκτονικής

και των δυνατοτήτων του Ubidots. Μέσω αυτού επιτυγχάνεται επέκταση των δυνατοτήτων σχετικά με την παρακολούθηση και την ανάλυση δεδομένων των εφαρμογών με ενσωμάτωση APIs όπως Weather Underground, Watson της IBM, Google Locations, Zapier και πολλά άλλα.

5.11.16 Ενσωμάτωση Hardware πλατφορμών

Η πλατφόρμα Ubidots υποστηρίζει μια πληθώρα πλατφορμών υλικού με στόχο την ανάπτυξη IoT υλοποιήσεων (<https://ubidots.com/docs/devices/#hologram>) όπως οι: Intel Edison, NodeMCU ESP8266, Raspberry Pi, BeagleBone Green Wireless και άλλες πολλές [18].

5.11.17 Συμπεράσματα

Η Ubidots πλατφόρμα στοχεύει στην συλλογή των δεδομένων που παράγονται από τους εγκατεστημένους κόμβους, και δίνει την δυνατότητα στον χρήστη να αποθηκεύει αυτές τις πληροφορίες στην cloud υποδομή οι οποίες είναι προσβάσιμες μέσω ενός συνδέσμου URL. Παρέχει μικροεφαρμογές (widgets) κατάλληλες για την ανάλυση και την απεικόνιση των πληροφοριών. Επίσης, χαρακτηρίζεται ως μια απλή με καλά τεκμηριωμένα APIs πλατφόρμα, την οποία την καθιστούν ένα ισχυρό, κλιμακωτό σύστημα χαμηλού κόστους.

5.12 Oracle IoT Platform

Η IoT πλατφόρμα της Oracle παρέχει μια ολοκληρωμένη λύση με στόχο την ανάπτυξη μιας κλιμακωτής και οικονομικά αποδοτικής αρχιτεκτονικής IoT, η οποία επιτρέπει στους οργανισμούς [48] :

- Την ανάπτυξη IoT εφαρμογών.
- Την διαχείριση και την ανάλυση μεγάλων όγκων δεδομένων από τις IoT συσκευές.
- Ενσωμάτωση και αυτοματοποίηση.
- Προστασία των δεδομένων από άκρο σε άκρο.
- Βελτιστοποίηση και καινοτομία μέσω της ενσωμάτωσης με εφαρμογές της Oracle.

5.12.1 Integration - APIs

Με την χρήση REST APIs επιτυγχάνεται η επικοινωνία μεταξύ των IoT συσκευών και της πλατφόρμας καθώς και η ενσωμάτωση με άλλες υπηρεσίες της Oracle.

5.12.2 Ασφάλεια Συσκευών

Η Oracle Internet of Things Cloud πλατφόρμα παρέχει διάφορα εργαλεία για την διαχείριση της ασφάλειας των συνδεδεμένων IoT συσκευών. Για να διασφαλιστεί η ασφάλεια των επικοινωνιών και της ροής δεδομένων του IoT δικτύου, η υπηρεσία Oracle Internet of Things Cloud Service υλοποιεί τον έλεγχο ταυτότητας OAuth2 πριν από την επικοινωνία με οποιαδήποτε συσκευή ή λογισμικό, αποδεικνύοντας την προέλευση των δεδομένων και εξασφαλίζοντας ότι όλα τα στοιχεία της ροής δεδομένων αποτελούν μέρος του δικτύου [20]. Η ασφάλεια επιπέδου μεταφοράς μέσω του HTTPS χρησιμοποιείται για όλες τις επικοινωνίες εξασφαλίζοντας ότι τα δεδομένα δεν είναι ευάλωτα σε τρίτους.

5.12.3 Πρωτόκολλα Επικοινωνίας

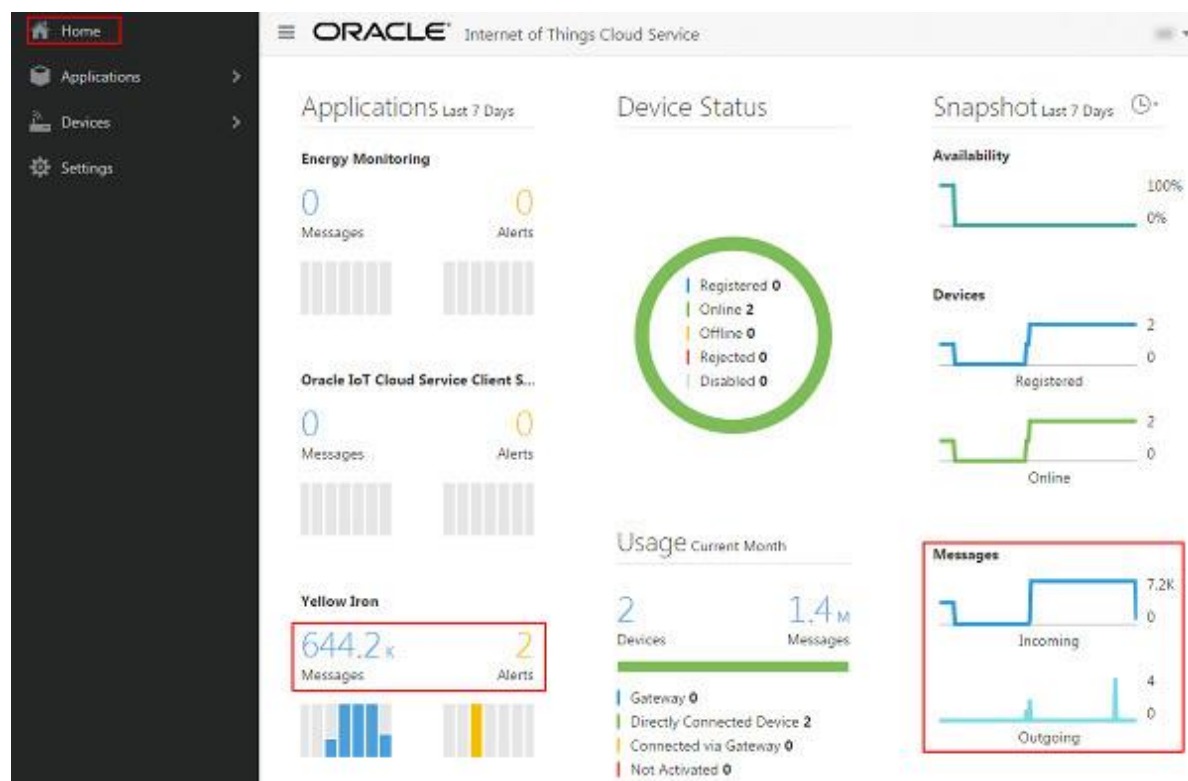
Υποστηρίζει πρωτόκολλα επικοινωνίας όπως το MQTT και το HTTPS.

5.12.4 Ανάλυση Δεδομένων

Η IoT πλατφόρμα της Oracle παρέχει την δυνατότητα για την ανάλυση ροών σε πραγματικό χρόνο. Διαθέτει ένα ευρύ φάσμα εργαλείων για επιχειρηματικές αναλύσεις και αναλύσεις βάσεων δεδομένων. Η ανάλυση των ροών δεδομένων σε πραγματικό χρόνο βασίζεται στο φιλτράρισμα (filtering) και στην συσχέτιση (correlation) των συμβάντων [2]. Επίσης, οι αναλυόμενες ροές δεδομένων μπορούν να αποστέλλονται στις ενσωματωμένες cloud υπηρεσίες ή σε επιχειρησιακές εφαρμογές για περαιτέρω επεξεργασία. Η παραπάνω διαδικασία πραγματοποιείται μέσω της Oracle Big Data Cloud εφαρμογής.

5.12.5 Απεικόνιση Δεδομένων

Για την απεικόνιση των δεδομένων η πλατφόρμα χρησιμοποιεί δυναμικούς πίνακες ελέγχου (dashboards), στους οποίους αποτυπώνονται δεδομένα και ειδοποιήσεις σε πραγματικό χρόνο [48]. Μέσω της απεικόνισης επιτυγχάνεται η προβολή βασικών δεικτών απόδοσης, στατιστικά στοιχεία για τον μέσο χρόνο μεταξύ αποτυχιών και άλλες πληροφορίες.



5.12.6 Hosting

Η πλατφόρμα εκμεταλλεύεται την υποδομή cloud της Oracle για βελτιστοποιημένη απόδοση των IoT εφαρμογών.

5.12.7 Push Notifications

Οι ειδοποιήσεις είναι μια πλήρως διαχειριζόμενη υπηρεσία δημοσίευσης-εγγραφής (publish-subscribe) που επιτρέπει την προώθηση των μηνυμάτων με μεγάλη κλίμακα και αξιοπιστία. Για παράδειγμα, με την ενσωματωμένη υποστήριξη για παρακολούθηση υποδομών Oracle Cloud, οι χρήστες μπορούν να λαμβάνουν ειδοποιήσεις άμεσα είτε μέσω ηλεκτρονικού ταχυδρομείου είτε μέσω προσαρμοσμένης υποστήριξης HTTP [16].

5.12.8 Ενεργοποίηση Συμβάντων (Trigger)

Μέσω της πλατφόρμας υποστηρίζεται η δυνατότητα ενεργοποίησης συμβάντων και η αποστολή ειδοποιήσεων μεταξύ των συσκευών. Η διαδικασία αυτή στηρίζεται σε έναν ενσωματωμένο μηχανισμό κανόνων (Rules engine) [48]. Όταν πληρείται μια προϋπόθεση ενός κανόνα, τότε ενεργοποιείται η σχετική ειδοποίηση ή ένα συμβάν. Οι κανόνες παρακολούθησης μπορούν να κατηγοριοποιηθούν ως εξής:

- Κανόνες βάσει τοποθεσίας: Για παράδειγμα, μπορεί ο χρήστης να παρακολουθεί πότε μια IoT συσκευή εγκαταλείπει τον τόπο που της έχει οριστεί και μέσω αυτού του κανόνα να δημιουργείται ένα περιστατικό.
- Κανόνες βάσει κατωφλίων (threshold): Οι κανόνες κατωφλίου βασίζονται σε τιμές αισθητήρων ή σε δείκτη KPI (δείκτης βασικών επιδόσεων). Οι κανόνες κατωφλίου χρησιμοποιούνται για την παρακολούθηση των τιμών των αισθητήρων, όπως επίπεδα καυσίμου και τιμές θερμοκρασίας.

5.12.9 Αποθήκευση Δεδομένων

Η υπηρεσία διαχείρισης μεγάλου όγκου δεδομένων της Oracle (Oracle Storage Cloud Service) απλοποιεί την αποθήκευση και την ανάλυση διαφόρων δεδομένων από τις IoT συνδεδεμένες συσκευές. Η Oracle ενσωματώνει απρόσκοπτα την αποθήκευση σε βάσεις δεδομένων όπως Hadoop ή NoSQL [16]. Επίσης, η ενσωμάτωση με την υπηρεσία Oracle Storage Cloud Service επιτρέπει την αποθήκευση ροών δεδομένων χρησιμοποιώντας αρχεία μορφοποιημένα σε CSV ή JSON.

5.12.10 Αποθήκευση Blob

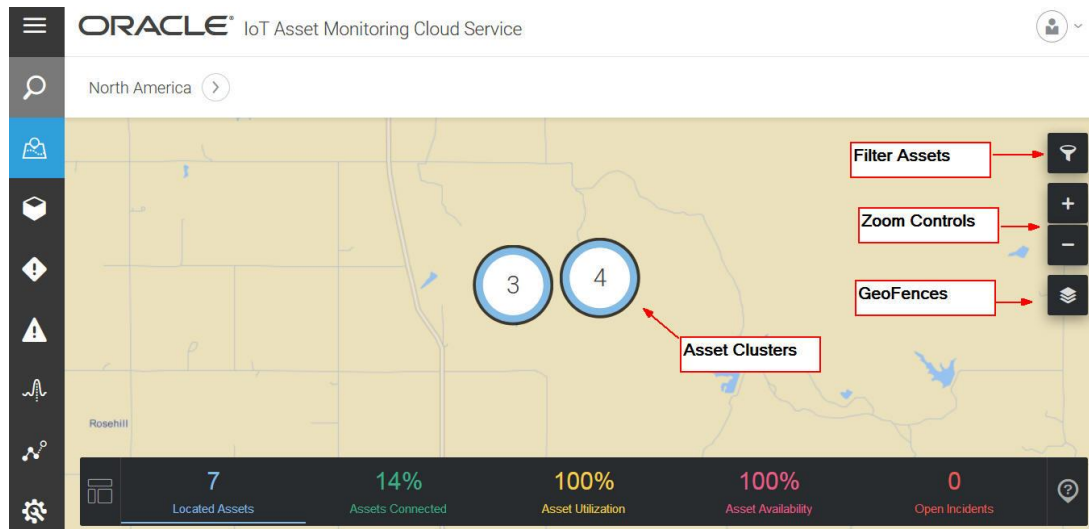
Η Oracle Multimedia εφαρμογή επιτρέπει στην Oracle Database να αποθηκεύει, να διαχειρίζεται και να ανακτά εικόνες, ήχο, βίντεο ή άλλα ετερογενή δεδομένα πολυμέσων.

5.12.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)

Παρέχεται η δυνατότητα στους προγραμματιστές να δημιουργήσουν εύκολα εφαρμογές και υπηρεσίες χρησιμοποιώντας έναν μεγάλο αριθμό SDKs τόσο για κινητές συσκευές όσο και πλατφόρμες: C Posix, Windows, Java, Android, Javascript ή iOS.

5.12.12 Geolocation

Μέσω της υπηρεσίας Oracle IoT Asset Monitoring Cloud επιτυγχάνεται η παρακολούθηση της τοποθεσίας των IoT συσκευών. Μέσω χαρτών πραγματοποιείται ο εντοπισμός της φυσικής τοποθεσίας των συσκευών. Οι IoT συσκευές μπορούν να εμφανίζονται ανεξάρτητα ή συγκεντρωμένα [48].



5.12.13 Πλάνα Τιμολόγησης

Η υπηρεσία πληρωμής βασίζεται στο μοντέλο Pay as You Go. Δεν υπάρχει κάποια δέσμευση εκ των προτέρων και η χρήση των υπηρεσιών μετριέται ανά ώρα και χρεώνεται μόνο για τους πόρους που καταναλώνονται.

5.12.14 Υποστηριζόμενες Γλώσσες Προγραμματισμού

Τόσο στο back-end της πλατφόρμας όσο και στο front-end γίνεται χρήση της γλώσσας Java.

5.12.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων

Η ενσωμάτωση με υπηρεσίες και πλατφόρμες τρίτων παρόχων επιτυγχάνεται με την χρήση REST APIs. Μια ενσωμάτωση ορίζεται ως ένα σύνολο ροών για την αποστολή δεδομένων σε μια εξωτερική εφαρμογή.

5.12.16 Ενσωμάτωση Hardware πλατφορμών

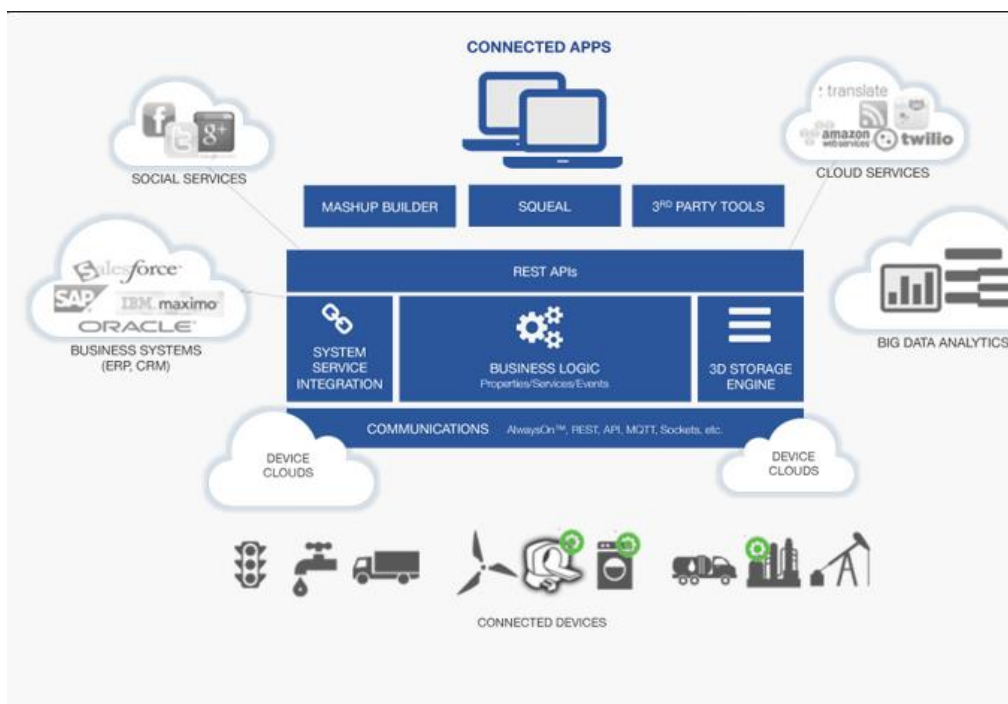
Η πλατφόρμα ενσωματώνεται με hardware πλατφόρμες όπως οι: Raspberry Pi, Arduino και άλλες.

5.12.17 Συμπεράσματα

Η Oracle IoT αποτελεί μια ολοκληρωμένη IoT πλατφόρμα η οποία επιτρέπει την επικοινωνία μεταξύ IoT συσκευών και των Oracle cloud υπηρεσιών μέσω των στοιχείων της Java (π.χ. Java, Java Card και Java Embedded), προσφέροντας συγκεκριμένες λύσεις Oracle όπως το Big Data Analytics και διαχείριση μεγάλου όγκου δεδομένων [48]. Συγκεκριμένα, η πλατφόρμα επιτρέπει την από άκρη σε άκρη (end to end) ασφάλεια, την ενσωμάτωση επιχειρηματικά συστήματα πληροφορικής, όπως ο Σχεδιασμός Επιχειρησιακών Πόρων (ERP).

5.13 ThingWorx Platform

Η πλατφόρμα ThingWorx (Εικόνα 5.24) είναι μια πλήρης, ολοκληρωμένη τεχνολογική πλατφόρμα ειδικά σχεδιασμένη για την υλοποίηση IoT λύσεων. Παρέχει εργαλεία και τεχνολογίες μέσω των οποίων οι επιχειρήσεις μπορούν να αναπτύξουν και να υλοποιήσουν γρήγορα ισχυρές IoT εφαρμογές. Η πλατφόρμα ThingWorx περιλαμβάνει συγκεκριμένη λειτουργικότητα σχεδιασμένη για βιομηχανική χρήση του IoT, συμπεριλαμβανομένης της επεκτασιμότητας και της ασφάλειας. Επίσης, η συγκεκριμένη πλατφόρμα προσφέρει διάφορα εργαλεία για την διαχείριση των δεδομένων και των εφαρμογών όπως το Composer, το Mashup Builder, μηχανή αναζήτησης και άλλα. Το εργαλείο Composer παρέχει ένα περιβάλλον μοντελοποίησης κατάλληλο για δοκιμές σχεδιασμού [46]. Το Mashup Builder προσφέρει την δυνατότητα δημιουργίας πινάκων ελέγχων (Dashboards). Επιπλέον, η Thingworx πλατφόρμα χρησιμοποιεί μία μηχανή αναζήτησης γνωστή ως SQUEAL, την οποία οι χρήστες χρησιμοποιούν για την ανάλυση, το φιλτράρισμα των δεδομένων και την αναζήτηση αρχείων.



Εικόνα 5.24: Αρχιτεκτονική Πλατφόρμας

5.13.1 Integration - APIs

Οι συσκευές και τα δεδομένα είναι προσπελάσιμα μέσω REST API [2].

5.13.2 Ασφάλεια

Η πλατφόρμα ThingWorx έχει σχεδιαστεί με γνώμονα την ασφάλεια των δεδομένων που ανταλλάσσονται μεταξύ των συσκευών και της πλατφόρμας. Επίσης, προσφέρει πολλαπλές επιλογές ελέγχου ταυτότητας ώστε να αυξήσει την ασφάλεια των IoT εφαρμογών. Μέσω της κρυπτογραφημένης επικοινωνίας κάνοντας χρήση του πρωτοκόλλου TLS καθώς και των ελέγχων πρόσβασης βασισμένων σε ρόλους, το ThingWorx ενσωματώνει μια σειρά χαρακτηριστικών ασφαλείας που μπορούν να αξιοποιηθούν στη διαδικασία ανάπτυξης των IoT εφαρμογών [46].

5.13.3 Πρωτόκολλα Επικοινωνίας

Μέσω του πρωτοκόλλου MQTT, μικρές ηλεκτρονικές συσκευές και αισθητήρες αποκτούν συνδεσιμότητα και μπορούν να επικοινωνούν με την ThingWorx πλατφόρμα [14]. Η διαδικασία αυτή επιτυγχάνεται με την εγκατάσταση μιας MQTT επέκτασης (extension).

5.13.4 Απεικόνιση Δεδομένων

Μέσω των Πινάκων ελέγχου (dashboards) τα οποία είναι προεγκατεστημένες εφαρμογές, επιτρέπουν την εύκολη απεικόνιση των δεδομένων και μπορούν να εξάγουν σημαντικές πληροφορίες για τους χρήστες. Υπάρχουν πολλές διαφορετικές περιπτώσεις χρήσης για την εφαρμογή ενός πίνακα ελέγχου. Για παράδειγμα, μπορεί να κατασκευαστεί ένα ταμπλό για την παρακολούθηση και τον έλεγχο των οικιακών συσκευών όπως θερμοστάτες, συναγερμοί καπνού, κλειδαριές και άλλα). Σε κάθε περίπτωση χρήσης, οι πίνακες ελέγχου δίνουν στον τελικό χρήστη τη δυνατότητα να ελέγχει δυναμικά τα δεδομένα, προβάλλοντας, μοιράζοντας και αναδιατάσσοντας τις πληροφορίες όπως ακριβώς ο ίδιος επιθυμεί. Για την υλοποίηση ενός πίνακα ελέγχου στην πλατφόρμα ThingWorx, απαιτούνται τέσσερα σημαντικά βήματα:

- 1) Δημιουργία μικροεφαρμογών (Gadgets)
- 2) Δημιουργία βιβλιοθήκης μικροεφαρμογών (Gadget Library mashup)
- 3) Δημιουργία Πίνακα Ελέγχου (Dashboard)
- 4) Ρύθμιση ορατότητας (visibility) και απεικόνισης (permissions)

5.13.5 Ανάλυση Δεδομένων

Η πλατφόρμα διαθέτει το εργαλείο ThingWorx Analytics, το οποίο έχει σχεδιαστεί ειδικά για να αντιμετωπίζει τις προκλήσεις που σχετίζονται με τον όγκο, την ταχύτητα και την ποικιλία των IoT δεδομένων. Το ThingWorx Analytics χρησιμοποιεί εξελιγμένη τεχνολογία τεχνητής νοημοσύνης (artificial intelligence) και μηχανικής μάθησης (machine learning) για την παροχή αξιόπιστων γνώσεων σε

πραγματικό χρόνο [9]. Επίσης, παρέχει την δυνατότητα αυτοματοποίησης πολύπλοκων διαδικασιών ανάλυσης. Επιπλέον προσφέρει:

- Μετατροπή των δεδομένων μέσω φιλικών προς το χρήστη διεπαφών, που επιτρέπουν πολύπλοκες δυνατότητες ανάλυσης για μη εξειδικευμένους χρήστες.
- Παροχή προβλέψεων και συστάσεων σε επιχειρησιακές λειτουργίες με στόχο την βελτίωση της λήψης αποφάσεων.
- Εντοπισμό στις αλλαγές λειτουργίας σε πραγματικό χρόνο, ελαχιστοποιώντας το χρόνο μέχρι την ανάλυση των δεδομένων.
- Βελτιστοποίηση των αποτελεσμάτων απόδοσης χρησιμοποιώντας προσομοιωτικές αναλύσεις μέσω των οποίων επιτυγχάνεται διάγνωση των προβλημάτων.

5.13.6 Hosting

Η εγκατάσταση και η διαχείριση [46] της πλατφόρμας μπορεί να πραγματοποιηθεί με τους παρακάτω τρόπους:

- Επί τόπου εγκατάσταση (on-premise), διαχείριση του λογισμικού ThingWorx σε ιδιόκτητους διακομιστές.
- Ανάπτυξη, φιλοξενία και διαχείριση σε κάποια cloud υποδομή όπως η Microsoft Azure.
- Εγκατάσταση σε Windows και Linux διακομιστές.

5.13.7 Push Notifications

Οι ειδοποιήσεις (alerts) παρέχουν μια μέθοδο για τη δημιουργία συμβάντων στην πλατφόρμα ThingWorx. Όταν ληφθεί κάποια ειδοποίηση η οποία βασίζεται σε κάποια προϋπόθεση – κανόνα, τότε το σύστημα ενεργοποιεί ένα συμβάν για την συγκεκριμένη συνθήκη [3]. Για παράδειγμα, εάν ένας χρήστης παρακολουθεί τις περιοχές θερμοκρασίας των μηχανών, μπορεί να ρυθμιστεί μια ειδοποίηση όταν η θερμοκρασία υπερβεί την επιθυμητή τιμή. Η ειδοποίηση ενός συμβάντος μπορεί να αποσταλεί στον χρήστη μέσω ηλεκτρονικού ταχυδρομείου (email) ή μηνύματος κειμένου (sms). Υπάρχουν οκτώ τύποι ειδοποιήσεων μερικοί από τους οποίους είναι:

- DateTime
- Integer
- EqualTo
- NotEqualTo

5.13.8 Ενεργοποίηση Συμβάντων (Trigger)

Η ενεργοποίηση συμβάντων γίνεται μέσω της λειτουργίας η οποία ονομάζεται ως: Συμβάντα Δεδομένων (Data Service Events). Η συγκεκριμένη υπηρεσία έχει τις εξής επιλογές:

- ServiceInvokeCompleted, Το συμβάν αυτό εμφανίζεται στην περίπτωση όπου έχει γίνει ενημέρωση δεδομένων σε έναν διακομιστή και τα δεδομένα αυτά θα πρέπει να αποσταλούν σε κάποια εφαρμογή.
- AllDataChanged, Το συμβάν αυτό ενεργοποιείται στην περίπτωση της ενημέρωσης ότι κάποια αλλαγή στα δεδομένα πραγματοποιήθηκε.

5.13.9 Αποθήκευση Δεδομένων

Η συγκεκριμένη πλατφόρμα διαθέτει διάφορους τύπους βάσεων δεδομένων όπως [14] :

- Ενσωματωμένες βάσεις δεδομένων για τις οποίες δεν απαιτείται κάποια πρόσθετη εγκατάσταση ή ρύθμιση.
- PostgreSQL, η οποία είναι η συνιστάμενη βάση δεδομένων για τα συστήματα παραγωγής και παρέχει τα εξής πλεονεκτήματα:
 - Είναι σε θέση να διαχειριστεί αποτελεσματικότερα μεγαλύτερες ποσότητες δεδομένων και να αποθηκεύσει περισσότερα δεδομένα χωρίς να επηρεάσει την απόδοση της πλατφόρμας.
 - Εγκαθίσταται ξεχωριστά από την πλατφόρμα.
 - Ευκολότερη ανάκτηση δεδομένων με πρόσβαση στη βάση δεδομένων από μια εξωτερική εφαρμογή (ξεχωριστή από την πλατφόρμα ThingWorx) χρησιμοποιώντας απλές εντολές SQL.
 - Υψηλή διαθεσιμότητα (High Availability)

5.13.10 Αποθήκευση Blob

Η λειτουργία υποστηρίζεται μέσω της υπηρεσίας Azure Blob Storage.

5.13.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)

Μέσω της πλατφόρμας υποστηρίζονται διάφορες βιβλιοθήκες λογισμικού που επιτρέπουν στους χρήστες να δημιουργήσουν το δική τους προσαρμοσμένη εφαρμογή, διαθέσιμη για διάφορες πλατφόρμες και γλώσσες προγραμματισμού - Java, C, .NET, iOS, Android [31].

5.13.12 Geolocation

Μέσω της μικροεφαρμογής (Widget) Google Map, παρέχεται ο έλεγχος και η δυνατότητα σε έναν χρήστη να παρακολουθεί την γεωγραφική τοποθεσία μιας IoT συσκευής η οποία έχει συνδεθεί με την πλατφόρμα καθώς και να εισάγει επιλεγμένες τοποθεσίες για νέες συσκευές [46]. Επίσης, μπορεί να ενημερώνεται για το ιστορικό της τοποθεσίας μιας συσκευής ή για την τρέχουσα θέση στην οποία βρίσκεται. Το συγκεκριμένο Widget συνεργάζεται μόνο με τους χάρτες της Google και απαιτείται ειδική άδεια χρήσης.

5.13.13 Πλάνα Τιμολόγησης

Η ThingWorx πλατφόρμα διατίθεται μόνο με συνδρομή. Αυτό προσφέρει στους πελάτες τα πλεονεκτήματα της ταχύτερης πρόσβασης σε νέες λειτουργίες και βελτιώσεις. Δυστυχώς, δεν παρέχονται πληροφορίες σχετικά με τις τιμές και οι πληροφορίες τιμολόγησης δεν είναι διαθέσιμες.

5.13.14 Υποστηριζόμενες Γλώσσες Προγραμματισμού

Σχετικά με την υποστήριξη γλωσσών προγραμματισμού υποστηρίζεται ένα μεγάλος εύρος. Συγκεκριμένα οι γλώσσες που χρησιμοποιούνται για την εγγραφή προσαρμοσμένου κώδικα από την πλευρά του πελάτη (client side) είναι οι C, Java και .Net

5.13.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων

Μια βασική πτυχή της αρχιτεκτονικής και του σχεδιασμού της πλατφόρμας ThingWorx είναι η δυνατότητα επέκτασης (extensibility) η οποία επιτρέπει την διασύνδεση με πλατφόρμες τρίτων παρόχων. Μέσω των επεκτάσεων μπορούν εύκολα να προστεθούν νέες λειτουργίες στην πλατφόρμα. Μερικές από τις διαθέσιμες επεκτάσεις που περιλαμβάνει η πλατφόρμα είναι: το Twitter, το Twillio, το Google Translate, το QR Code και άλλα [14].

5.13.16 Ενσωμάτωση Hardware πλατφορμών

Κάποιες από τις πλατφόρμες υλικού που υποστηρίζονται με στόχο την ανάπτυξη IoT εφαρμογών είναι οι εξής:

- Intel Galileo
- Intel Galileo Gen 2
- Intel Edison
- NXP iMX6UltraLite Evaluation Kit
- Raspberry Pi

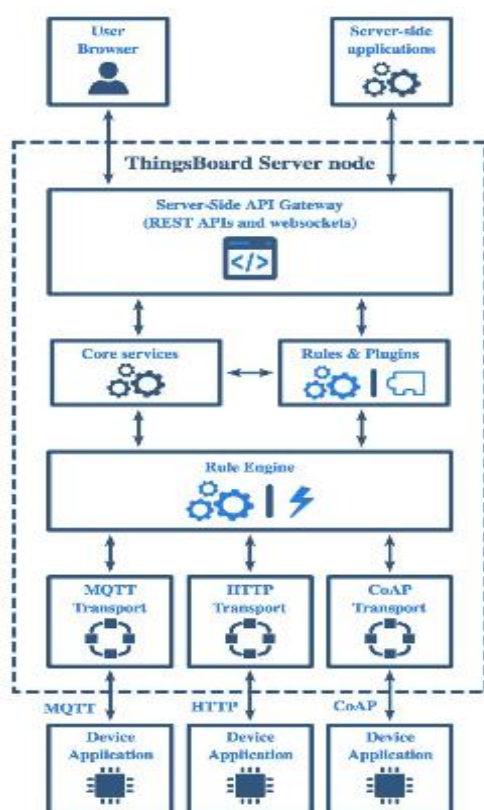
5.13.17 Συμπεράσματα

Η πλατφόρμα ThingWorx αποτελεί μία δημοφιλή επιλογή στον τομέα των IoT Cloud πλατφορμών. Επίσης, η πλατφόρμα παρέχει μία υποδομή η οποία βασίζεται σε μοντέλα όπως το M2M (Machine to Machine), το IaaS (infrastructure as a service) καθώς και του DDDM (Data-driven decision making) μέσω του οποίου η λήψη αποφάσεων βασίζεται στα δεδομένα τα οποία ανταλλάσσονται και όχι μόνο στην παρατήρηση αυτών. Ένα από τα βασικά πλεονεκτήματα της πλατφόρμας είναι το ότι διαθέτει το χαρακτηριστικό του "υπολογισμού της έντασης δεδομένων" (Data-intensive computing) για τη διεκπεραίωση μεγάλων όγκων δεδομένων, συνήθως σε μέγεθος terabytes ή petabytes. Άλλα πλεονεκτήματα είναι η εύκολη συνδεσιμότητα με ηλεκτρονικές συσκευές όπως αισθητήρες, RFID καθώς επίσης και η εύκολη διαχείριση συσκευών. Ωστόσο πέραν των πλεονεκτημάτων υπάρχουν και μειονεκτήματα όπως ο περιορισμένος αριθμός των συνδεδεμένων συσκευών.

5.14 ThingsBoard Platform

Όπως παρουσιάζεται και στον επίσημο δικτυακό τόπο (<https://thingsboard.io/>) πρόκειται για μια IoT πλατφόρμα open-source μέσω της οποίας επιτυγχάνεται η συλλογή, η επεξεργασία, η απεικόνιση, η διαχείριση των συσκευών καθώς και των δεδομένων που ανταλλάσσονται μεταξύ αυτών. Παρέχει τη συνδεσιμότητα των συσκευών μέσω πρότυπων πρωτοκόλλων IoT όπου το MQTT, το CoAP, το HTTP και υποστηρίζει την ανάπτυξη τόσο σε cloud υποδομές όσο και σε ιδιωτικές εγκαταστάσεις (on-premises) [47]. Η πλατφόρμας ThingsBoard συνδυάζει την επεκτασιμότητα, την ανοχή σφαλμάτων και την απόδοση.

Η γενική αρχιτεκτονική της πλατφόρμας παρουσιάζεται στην Εικόνα 5.19:



Εικόνα 5.19: Αρχιτεκτονική πλατφόρμας ThingsBoard

- **Devices**, η πλατφόρμα υποστηρίζει πρωτόκολλα όπως το MQTT, το CoAP και το HTTP για τη συνδεσιμότητα των συσκευών. Επίσης, παρέχεται και η παραμετροποίηση των υφιστάμενων πρωτοκόλλων.
- **Rule Engine**, ο μηχανισμός αυτός επιτρέπει την επεξεργασία των μηνυμάτων από τις συσκευές και την ενεργοποίηση ρυθμιζόμενων συμβάντων (modules) που ονομάζονται Plugins

- **Core Services**, μέσω των οποίων επιτυγχάνεται η διαχείριση των οντοτήτων
- **Service-side API Gateway**, Websockets and REST API calls.

5.14.1 Integration API

Η Thingsboard πλατφόρμα μπορεί να λειτουργήσει ως υπηρεσία ιστού (web service) RESTful κάτι το οποίο μας επιτρέπει να κάνουμε κλήσεις (calls) REST και να συλλέγουμε τα δεδομένα από την πλατφόρμα. Για παράδειγμα θα μπορούσαμε αν θέλαμε να χρησιμοποιήσουμε όλα τα δεδομένα που συλλέχθηκαν από την πλατφόρμα σε έναν αλγόριθμο μηχανικής μάθησης (machine learning algorithm) [24]. Για να υποβάλετε αιτήματα, χρειάζονται ορισμένες πιστοποιήσεις.

5.14.2 Ασφάλεια

Σχετικά με την διασφάλιση των δεδομένων, η συγκεκριμένη πλατφόρμα διαθέτει διάφορα επίπεδα ασφαλείας. Αρχικά, επιτρέπει την δυνατότητα εκτέλεσης ενός MQTT διακομιστή μέσω SSL πρωτοκόλλου. Υποστηρίζονται "one-way" και "two-way" SSL. Για την ενεργοποίηση του, κρίνεται απαραίτητη η εγκατάσταση ενός έγκυρου πιστοποιητικού SSL [47]. Επίσης, για την διασύνδεση των εφαρμογών που εκτελούνται στις απομακρυσμένες συσκευές με τον διακομιστή της πλατφόρμας απαιτείται η χρήση διαπιστευτηρίων (credentials). Υπάρχουν δύο τύποι διαπιστευτηρίων που υποστηρίζονται:

- **Access Tokens**, ο έλεγχος ταυτότητας βάσει αναγνωριστικού μπορεί να χρησιμοποιηθεί σε μη κρυπτογραφημένη ή μονόδρομη λειτουργία SSL.
- **X.509 Certificates**, ο έλεγχος ταυτότητας που βασίζεται σε πιστοποιητικό X.509 χρησιμοποιείται σε αμφίδρομη λειτουργία SSL.

5.13.3 Πρωτόκολλα Επικοινωνίας

Επιτρέπει τη συνδεσιμότητα των συσκευών μέσω πρότυπων IoT πρωτοκόλλων όπως το MQTT, το CoAP και το HTTP.

5.14.4 Απεικόνιση Δεδομένων

Η ThingsBoard πλατφόρμα επιτρέπει την δημιουργία και την παραμετροποίηση προσαρμοσμένων πινάκων ελέγχου (Dashboards). Κάθε Πίνακας ελέγχου μπορεί να ενσωματωθεί με πολλαπλά widgets τα οποία απεικονίζουν δεδομένα από πολλαπλές συσκευές IoT [30]. Επιπλέον, είναι δυνατή η προσαρμογή των

εργαλείων απεικόνισης μέσω των widgets καθώς επίσης και η δημιουργία νέων widget απεικόνισης με την χρήση HTML, CSS και Javascript.

5.14.5 Ανάλυση Δεδομένων

Η ανάλυση των δεδομένων επιτυγχάνεται μέσω της υπηρεσίας Apache Spark, η οποία είναι μια ενοποιημένη μηχανή ανάλυσης για επεξεργασία δεδομένων μεγάλης κλίμακας.

5.14.6 Hosting

Η πλατφόρμα μπορεί να εγκατασταθεί και να λειτουργήσει τοπικά σε κάποια πλατφόρμα υλικού όπως το Raspberry PI αλλά και σε ισχυρούς cloud διακομιστές. Επίσης υποστηρίζεται και η χρήση εικόνων docker με στόχο την απλοποίηση της εγκατάστασης. Τέλος διατίθενται και εναλλακτικές επιλογές εγκατάστασης όπως μέσω Windows και Linux λειτουργικών συστημάτων [47].

5.14.7 Push Notifications

Μέσω της πλατφόρμας υποστηρίζεται η δυνατότητα αποστολής ειδοποιήσεων για κάποιο συμβάν το οποίο έλαβε μέρος. Για παράδειγμα για την αποστολή μιας ειδοποίησης μέσω ηλεκτρονικού ταχυδρομείου θα πρέπει πρώτα να πραγματοποιηθεί η εγκατάσταση ενός email plugin. Ο μηχανισμός κανόνων (Rule Engine) επιτρέπει να επεξεργασία των μηνυμάτων από τις απομακρυσμένες συσκευές. Αυτός ο μηχανισμός παρέχει την δυνατότητα δημιουργίας κάποιων πολύ χρήσιμες λειτουργιών, όπως για παράδειγμα: την αποστολή ενός μηνύματος ηλεκτρονικού ταχυδρομείου όταν μια ιδιότητα της συσκευής αλλάξει, την δημιουργία συναγερμού (alarm) όταν κάποια τιμή υπερβαίνει ένα συγκεκριμένο όριο, την προώθηση των δεδομένων σε κάποιο εξωτερικό διακομιστή RESTful καθώς πολλά άλλα με άλλους κανόνες και την χρήση διαφόρων plugins [24]. Ο μηχανισμός κανόνων αποτελείται από τρία βασικά στοιχεία: τα φίλτρα (filters), τον επεξεργαστή (Processor) και την δράση (Action). Ανάλογα με την εφαρμογή, κάθε στοιχείο ενδέχεται να απαιτεί συγκεκριμένες ρυθμίσεις πριν να μπορέσει να χρησιμοποιηθεί.

5.14.8 Ενεργοποίηση Συμβάντων (Trigger)

Μέσω του ThingsBoard Rule Engine παρέχεται η δυνατότητα για τη σύνθετη επεξεργασία συμβάντων [47]. Ακολουθούν μερικές περιπτώσεις χρήσης:

- Δημιουργία / Ενημέρωση / Εκκαθάριση συναγερμών βάσει καθορισμένων συνθηκών.

- Ενεργοποίηση συμβάντων βασισμένα στην κατάσταση συσκευής. Για παράδειγμα, δημιουργία ειδοποιήσεων για τον αν η συσκευή είναι συνδεδεμένη ή απενεργοποιημένη.
- Api calls σε εξωτερικά συστήματα.

5.14.9 Αποθήκευση Δεδομένων

Η ThingsBoard πλατφόρμα παρέχει την χρήση βάσης δεδομένων SQL (PostgreSQL συνιστάται) ή NoSQL (Cassandra). Από προεπιλογή χρησιμοποιεί ενσωματωμένη HSQLDB βάση η οποία είναι πολύ βολική για σκοπούς αξιολόγησης ή ανάπτυξης. Επίσης, παρέχει την δυνατότητα αποθήκευσης αρχείων δυαδικού περιεχομένου (binary data) [24].

5.14.10 Αποθήκευση Blob

Παρέχεται η δυνατότητα αποθήκευσης δυαδικού περιεχομένου (αρχεία) στην Βάση δεδομένων.

5.14.11 Κιτ Ανάπτυξης Λογισμικού (SDKs)

Μέχρι και την έκδοση 2.3 δεν υποστηρίζονται SDKs, ωστόσο από την έκδοση 2.4 θα παρέχεται η δυνατότητα mobile SDKs.

5.14.12 Geolocation API

Για τον εντοπισμό της γεωγραφικής θέσης των συσκευών, η πλατφόρμα χρησιμοποιεί την υπηρεσία Google maps και ένα ιχνηλάτη (tracker) της εταιρείας Yerpzon Asset. Η Yerpzon παρέχει κορυφαίες τεχνολογικές λύσεις παρακολούθησης και δεδομένων αισθητήρων για την κάλυψη τόσο ιδιωτικών όσο και επιχειρηματικών αναγκών [47]. Διαθέτει την δική της πλατφόρμα για την απεικόνιση και διαχείριση των συσκευών και των δεδομένων. Η Yerpzon παρέχει ένα API ενοποίησης χρησιμοποιώντας μεθόδους RESTful, οι οποίες μπορούν να χρησιμοποιηθούν για να λάβουν τα δεδομένα που ο ιχνηλάτης έστειλε στην πλατφόρμα. Το Thingsboard παρέχει χάρτες μέσω widget για την απεικόνιση της τοποθεσίας, της διαδρομής και παρακολούθησης, οπότε αυτό που χρειάζεται να γίνει είναι να ληφθούν τα δεδομένα από το Yerpzon μέσω του API και να εμφανισθούν τα δεδομένα στα γραφικά στοιχεία χάρτη.

5.14.13 Πλάνα τιμολόγησης

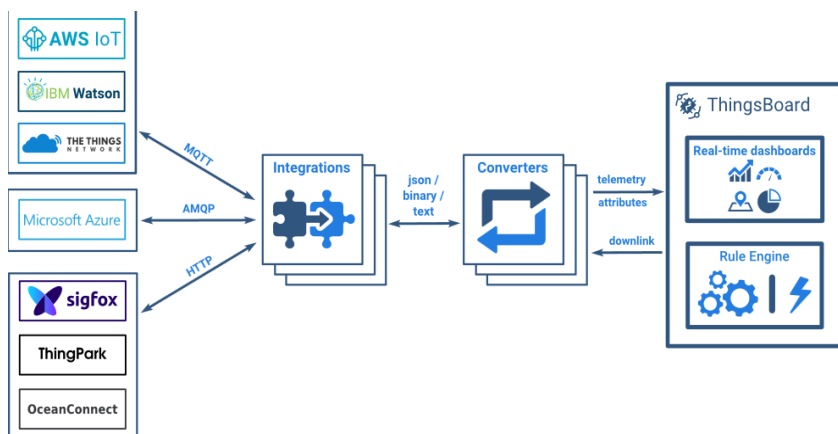
Η πλατφόρμα διαθέτει δύο κατηγορίες σχετικά με την τιμολόγηση των υπηρεσιών, την βασική έκδοση (Community Edition) και την έκδοση με πρόσθετες λειτουργίες (Professional Edition) [30].

5.14.14 Υποστηριζόμενες Γλώσσες Προγραμματισμού

Το back-end της πλατφόρμας είναι γραμμένο σε γλώσσα Java καθώς επίσης χρησιμοποιούνται και κάποιες μικρο-υπηρεσίες (micro-services) βασισμένες σε Node.js. Από την άλλη μεριά το front-end σε Angular JS πλαίσιο (framework).

5.14.15 Ενσωμάτωση με πλατφόρμες Τρίτων Παρόχων

Η ThingsBoard πλατφόρμα υποστηρίζει δύο κύρια πρωτόκολλα ενσωμάτωσης (integration): το HTTP και το MQTT (Εικόνα 5.20). Για παράδειγμα, το SigFox backend χρησιμοποιεί το HTTP για την προώθηση δεδομένων (push data) στην ThingsBoard πλατφόρμα ή σε οποιοδήποτε άλλο σύστημα. Από την άλλη πλευρά πλατφόρμες τρίτων παρόχων όπως το AWS IoT, το IBM Watson και το Azure Event Hub επιτρέπουν την εγγραφή (subscribe) στην τροφοδοσία δεδομένων από συσκευές μέσω του MQTT [30]. Παρόμοια, μερικές πλατφόρμες όπως η LoRaWAN και το NB IoT επιτρέπουν ταυτόχρονα τις διεπαφές HTTP και MQTT.



Εικόνα 5.20: Υποστηριζόμενα πρωτόκολλα επικοινωνίας

5.14.16 Ενσωμάτωση Hardware πλατφορμών

Υπάρχει μία μεγάλη ποικιλία hardware πλατφορμών το οποίο υποστηρίζεται από την πλατφόρμα, με στόχο την κατασκευή IoT εφαρμογών. Τα συνηθισμένα παραδείγματα πλατφορμών περιλαμβάνουν μικροελεγκτές όπως το Raspberry Pi, το Arduino, το Xiaomi MI και άλλα.

5.14.17 Συμπεράσματα

Η ThingsBoard είναι μια πλατφόρμα ανοιχτού κώδικα η εγκατάσταση της οποίας μπορεί να πραγματοποιηθεί τοπικά σε κάποιο ιδιόκτητο διακομιστή (server) ή μέσω της cloud υποδομής η οποία παρέχεται μέσω της πλατφόρμας. Μέσω της χρήσης μηχανισμού κανόνων (Rules Engine) πραγματοποιείται η ενεργοποίηση συμβάντων και η δημιουργία ειδοποιήσεων. Ως προς την απεικόνιση δεδομένων παρέχει αρκετές μικροεφαρμογές (widgets) οι οποίες βοηθούν στην δημιουργία αρκετά προσαρμόσιμων πινάκων ελέγχων (dashboards). Επίσης, η κλιμάκωση της πλατφόρμας μπορεί να επιτευχθεί με την προσθήκη όμοιων διακομιστών σε κατάσταση αναμονής (hot standby servers), αντιγράφοντας τα δεδομένα δε NoSQL βάση.

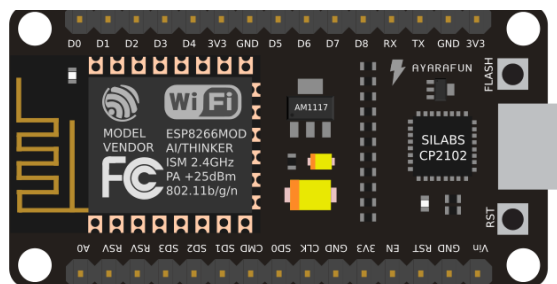
ΚΕΦΑΛΑΙΟ 6 – ΔΟΚΙΜΗ ΠΛΑΤΦΟΡΜΩΝ

Στο συγκεκριμένο κεφάλαιο περιγράφεται αναλυτικά η διαδικασία των δοκιμών που πραγματοποιήθηκαν για τις εξής IoT πλατφόρμες: Ubidots, Thingspeak και ThingWorx. Το κύριο σενάριο των δοκιμών αφορά την εγκατάσταση ενός αισθητήρα σε έναν μικροελεγκτή (Arduino esp8266), ο οποίος αισθητήρας λαμβάνει τιμές θερμοκρασίας και υγρασίας από το περιβάλλον και τις αποστέλλει με την χρήση APIs και πρωτοκόλλων επικοινωνίας όπως το MQTT και το CoAP στις αναφερθείσες πλατφόρμες. Επίσης, παρουσιάζονται σενάρια κατά τα οποία δοκιμάζονται εργαλεία απεικόνισης και ανάλυσης των δεδομένων καθώς και η δημιουργία προσαρμοσμένων Πινάκων Ελέγχου (Dashboards) με την χρήση widgets. Επιπλέον, έχει πραγματοποιηθεί δοκιμή σχετικά με την ενεργοποίηση συμβάντων (Trigger Actions) και αποστολή ειδοποιήσεων (Push Notifications). Ακόμη, έγινε εγκατάσταση των mobile apps που προσφέρουν οι εν λόγω πλατφόρμες σε android συσκευή, με αποτέλεσμα την χρήση των λειτουργιών που προσφέρουν οι πλατφόρμες μέσω της κινητής συσκευής.

6.1 Υλικό Δοκιμών

Για τις δοκιμές έγινε χρήση των παρακάτω στοιχείων:

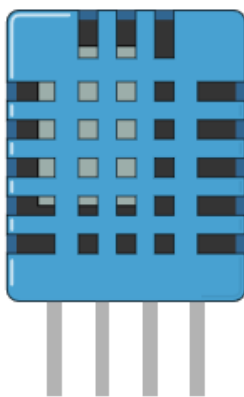
1. NodeMCU – ESP8266: Το NodeMCU (Node MicroController Unit), Εικόνα 6.1, αποτελεί ένα περιβάλλον ανάπτυξης ανοιχτού λογισμικού και υλικού το οποίο έχει κατασκευαστεί γύρω από ένα σύστημα βασισμένο σε ένα ολοκληρωμένο κύκλωμα (SoC) το οποίο ονομάζεται ESP8266. Το ESP8266, έχει σχεδιαστεί και κατασκευαστεί από την εταιρεία Espressif Systems, και αποτελείται από όλα εκείνα τα στοιχεία ενός σύγχρονου ηλεκτρονικού υπολογιστή όπως: CPU, RAM, δικτύωση (WiFi), και ακόμη και ένα μοντέρνο λειτουργικό σύστημα και SDK.



Εικόνα 6.1: Πλακέτα NodeMCU – ESP8266

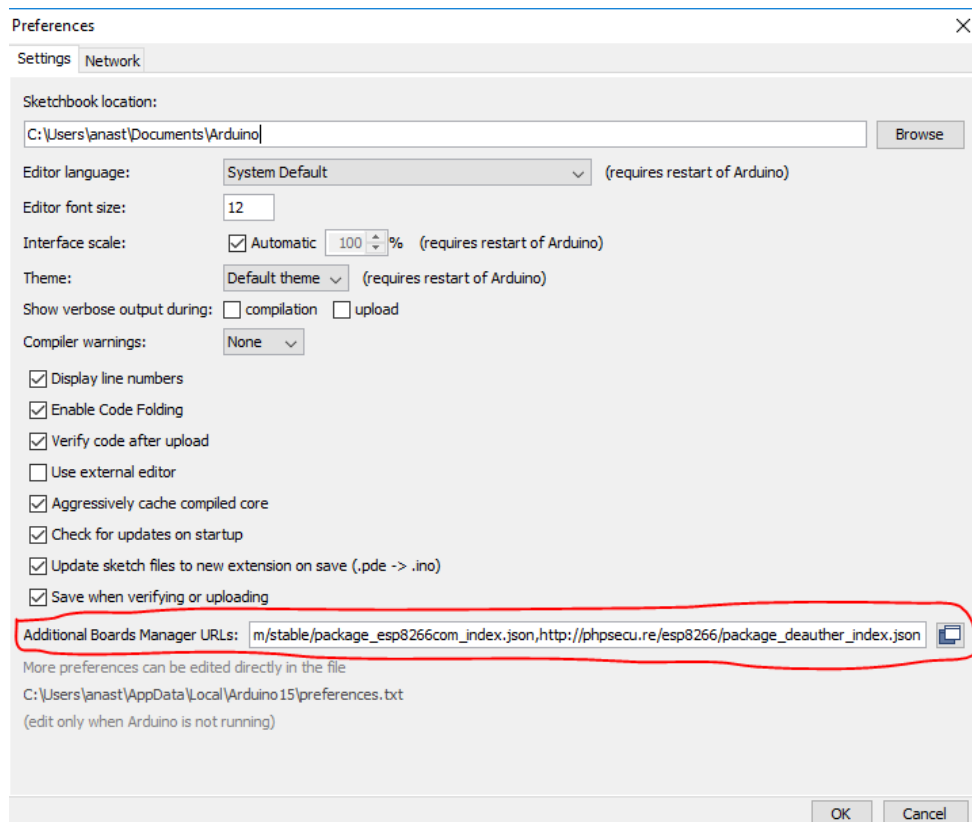
Η πλακέτα μπορεί να προγραμματιστεί χρησιμοποιώντας «σκίτσα» (sketches) Lua, Micropython και Arduino. Με τον όρο «σκίτσο» αναφερόμαστε στο μέρος του κώδικα που φορτώνεται και τρέχει στην συσκευή. Η τροφοδοσία του **NodeMCU – ESP8266** και η επικοινωνία του με τον ηλεκτρονικό υπολογιστή για την αποστολή και παραλαβή δεδομένων προγραμματισμού ή της παρακολούθησης των αποτελεσμάτων γίνεται με την χρήση ενός βύσματος Micro-USB. Όλα τα παραπάνω το καθιστούν ιδανικό εργαλείο για την υλοποίηση IoT έργων.

2. Αισθητήρας DHT11: Συνδυάζοντας τόσο αισθητήρα θερμοκρασίας όσο και υγρασίας, το DHT11 (Εικόνα 6.2) επιτρέπει τη βασική παρακολούθηση των περιβαλλοντικών συνθηκών. Με μια κλίμακα μέτρησης 0-50 ° C για την θερμοκρασία και 20-95% για την εργασία, το καθιστά ιδανικό για τις περισσότερες βασικές IoT εφαρμογές. Παρόλο που έχει χαμηλότερη ακρίβεια από έναν DHT22 ή DS18B20, το χαμηλό κόστος του καθιστά ιδανικό για βασικές υλοποιήσεις. Οι μετρήσεις γίνονται δεκτές μέσω ενός μοναδικού ακροδέκτη δεδομένων και δέχεται τροφοδοσία τόσο 3.3V όσο και 5V DC.



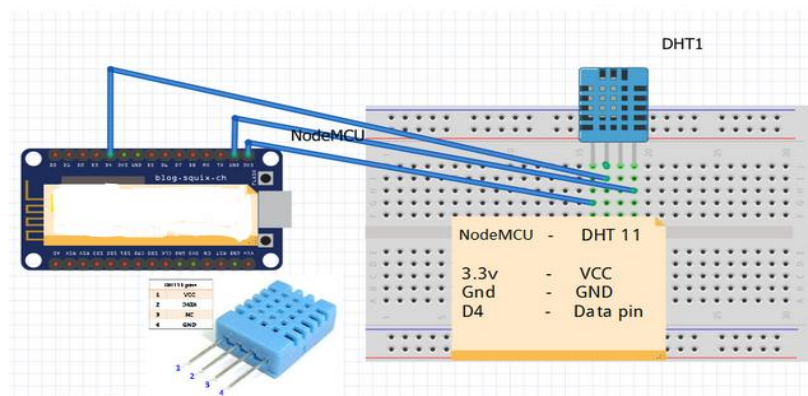
Εικόνα 6.2: Αισθητήρας DHT1 θερμοκρασίας και υγρασίας

3. Λογισμικό Arduino (C++) : Σχεδιασμένο κυρίως για χρήση μικροελεγκτών Arduino, το συγκεκριμένο IDE (Integrated Development Environment) περιλαμβάνει επίσης υποστήριξη για πλακέτες (boards) τρίτων κατασκευαστών όπως το **NodeMCU - ESP8266**. Η εγκατάσταση της συγκεκριμένης πλακέτας γίνεται μέσω του διαχειριστικού μενού που παρέχει το συγκεκριμένο λογισμικό της πλακέτας, συμπεριλαμβάνοντας απλά τη διεύθυνση URL του πηγαίου κώδικα (Εικόνα 6.3). Τα «σκίτσα» γράφονται σε μια παραλλαγή της γλώσσας C ++ και υπάρχει μεγάλη ποικιλία οδηγιών για τη δημιουργία και λειτουργία «σκίτσων» Arduino στην **NodeMCU - ESP8266** συσκευή.

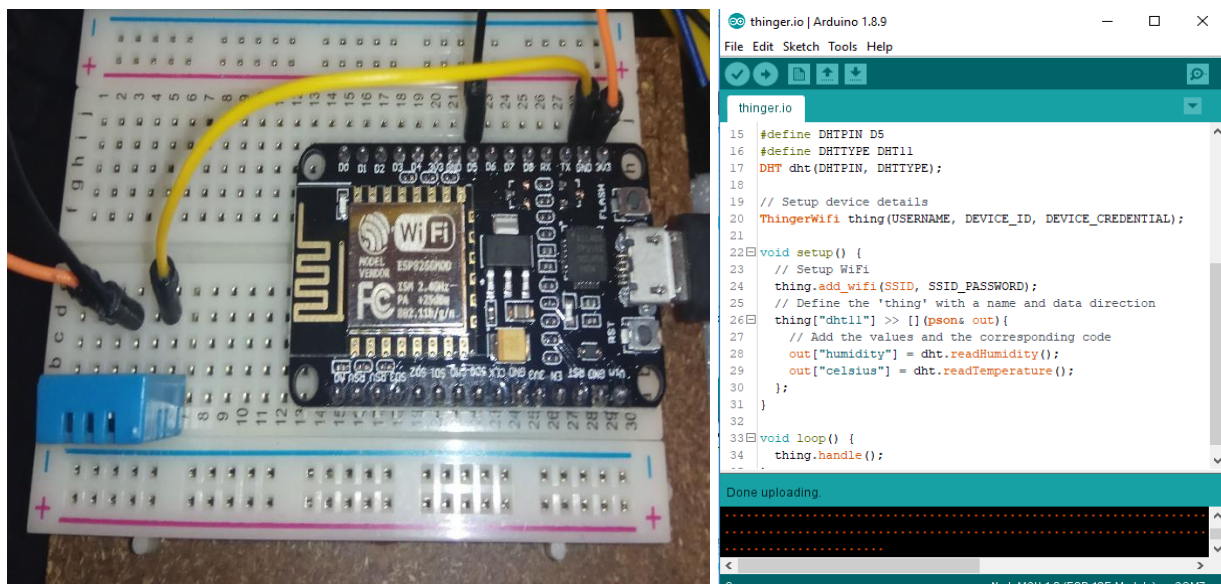


Εικόνα 6.3: Εγκατάσταση ESP8266 στο Arduino IDE

Η τελική μορφή των παραπάνω στοιχείων καθώς και η συνδεσμολογία απεικονίζονται στις παρακάτω Εικόνες 6.4 και 6.5.



Εικόνα 6.4: Συνδεσμολογία



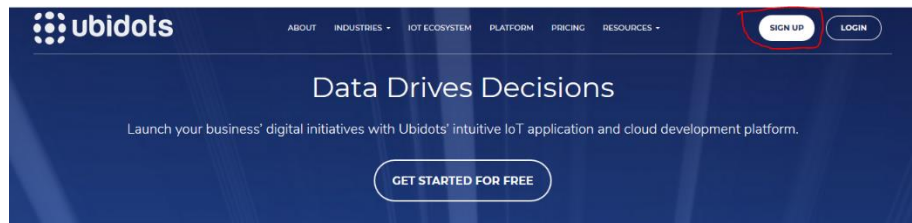
Εικόνα 6.5: Υλικό Δοκιμών

6.2 Δοκιμή Πλατφόρμας Ubidots

Η πλατφόρμα Ubidots όπως αναφέρθηκε και σε προηγούμενο κεφάλαιο επιτρέπει σε IoT συσκευές να αποστέλλουν δεδομένα στην πλατφόρμα μέσω των πρωτοκόλλων MQTT και HTTP. Για τις δοκιμές χρησιμοποιήθηκε η συσκευή NodeMCU και ένας αισθητήρας DHT11 για την μέτρηση τιμών θερμοκρασίας και υγρασίας καθώς και η αποστολή και η απεικόνιση των δεδομένων στην πλατφόρμα. Τα βήματα της δοκιμής αναλύονται παρακάτω

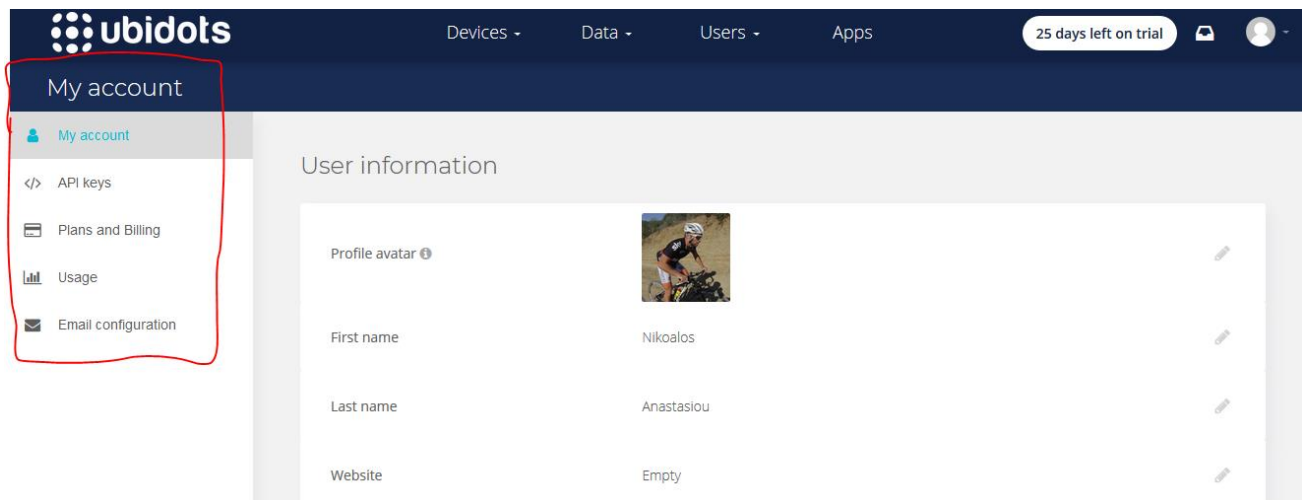
6.2.1 Δημιουργία Λογαριασμού

Μέσω της ιστοσελίδας <https://ubidots.com/> πραγματοποιήθηκε η δημιουργία του λογαριασμού όπως φαίνεται και στην Εικόνα 6.6. Η Ubidots διαθέτει 2 τύπους έκδοσης λογαριασμού, μία δωρεάν για διδακτικούς σκοπούς και μια εμπορική έκδοση για προγραμματιστές και επιχειρήσεις όπου η χρέωση γίνεται ανάλογα με τον αριθμό των συνδεδεμένων συσκευών και του όγκου των δεδομένων που αποστέλλονται μεταξύ των συσκευών και της πλατφόρμας. Στην συγκεκριμένη περίπτωση έγινε επιλογή της δωρεάν έκδοσης στην οποία διατίθενται 5000 μονάδες (credits), οι οποίες σχετίζονται με τον όγκο των δεδομένων που αποστέλλονται.



Εικόνα 6.6 Δημιουργία Λογαριασμού

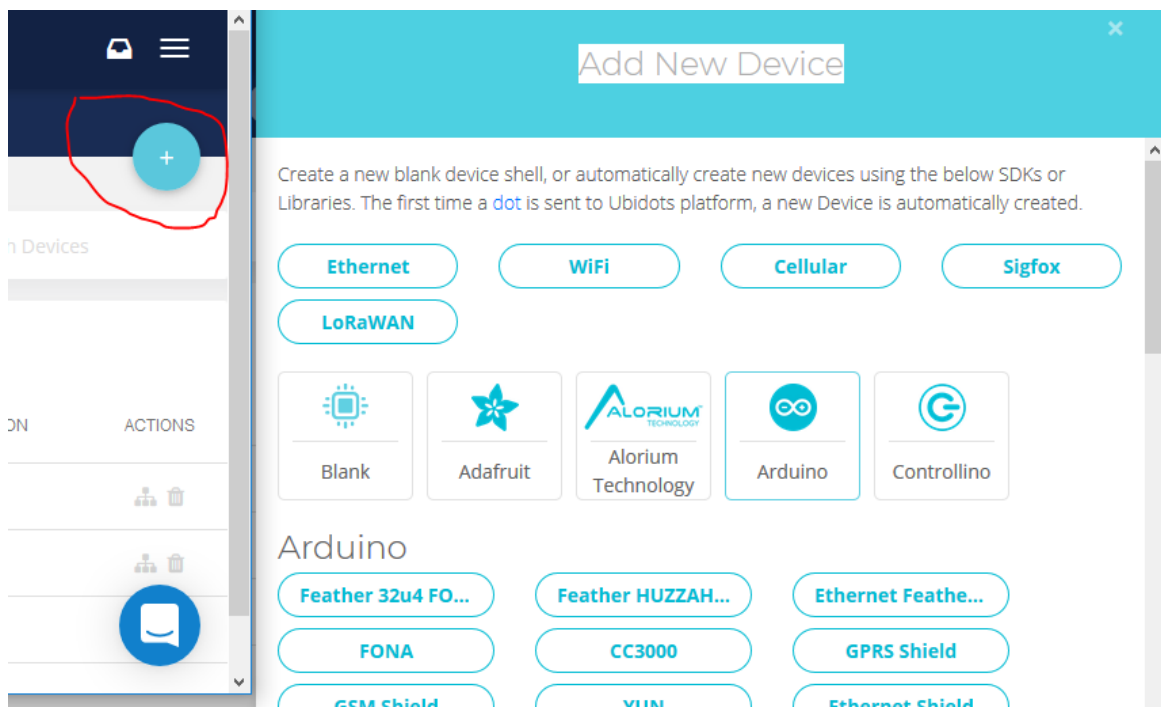
Από το μενού **My account** (Εικόνα 6.7) μπορούμε να δούμε και να παραμετροποιήσουμε διάφορα στοιχεία για τον λογαριασμό μας, όπως το πλάνο τιμολόγησης τον όγκο των δεδομένων που έχουν αποσταλεί και άλλα.



Εικόνα 6.7: Στοιχεία λογαριασμού

6.2.2 Προσθήκη Συσκευής

Επόμενο βήμα είναι η προσθήκη της IoT συσκευής στην πλατφόρμα. Κατά την προσθήκη μιας νέας συσκευής ανοίγει ένα μενού (Εικόνα 6.8) από το οποίο μπορεί να γίνει επιλογή συσκευής διαφόρων κατασκευαστών καθώς και η επιλογή του μέσου διασύνδεσης όπως Ethernet, WiFi, Cellular και άλλων.

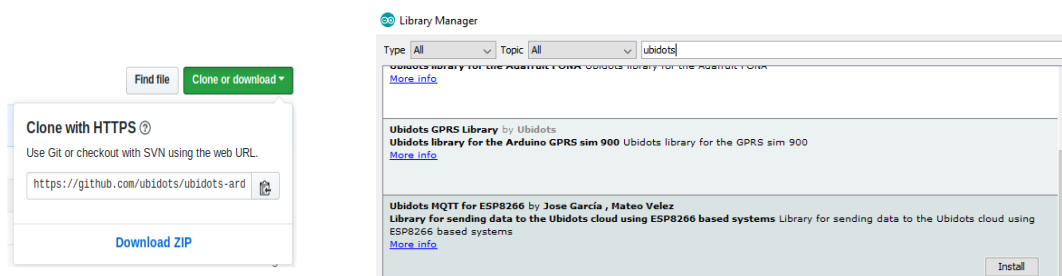


Εικόνα 6.8: Διαδικασία προσθήκης συσκευής

Για να ολοκληρωθεί η προσθήκη της συσκευής θα πρέπει να γίνουν ορισμένες ενέργειες στο Arduino IDE, όπως η φόρτωση του κώδικα, η επιλογή της πλακέτας NodeMCU καθώς και η εγκατάσταση των ανάλογων βιβλιοθηκών. Οι συγκεκριμένες ενέργειες παρουσιάζονται στο παρακάτω βήμα.

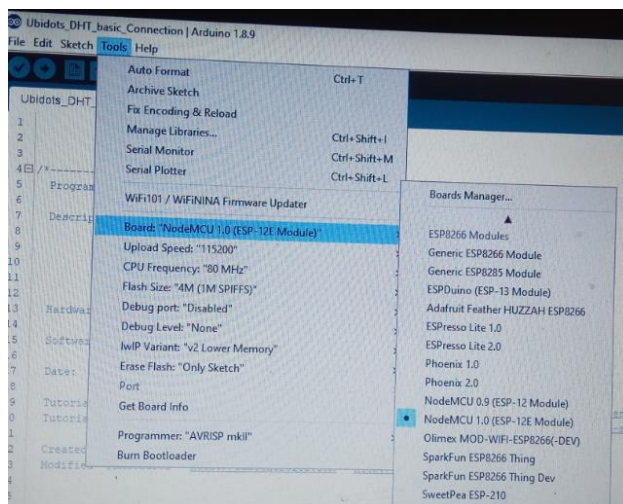
6.2.3 Παραμετροποίηση Arduino IDE

Η πλατφόρμα Ubidots, όπως και άλλες εφαρμογές λογισμικού καθώς και IoT πλατφόρμες, διαθέτουν προ-διαμορφωμένες (pre- configured) βιβλιοθήκες έτοιμες να εγκατασταθούν στο λογισμικό Arduino IDE. Για παράδειγμα, η βιβλιοθήκη Ubidots περιέχει πολλές μεθόδους για την αποστολή (SEND) και τη λήψη (GET) δεδομένων προς και από την πλατφόρμα. Η εγκατάσταση της συγκεκριμένης βιβλιοθήκης μπορεί να γίνει με δύο τρόπους όπως απεικονίζονται στην Εικόνα 6.9.



Εικόνα 6.9: Επιλογές εγκατάστασης βιβλιοθήκης

Η εγκατάσταση μπορεί ακόμη μέσω του Github αποθετηρίου (repository) ή απευθείας από το Arduino IDE επιλέγοντας **Sketch/Program > Include Library > Library Manager**, αναζήτηση της πλατφόρμας "Ubidots" και επιλογή εγκατάσταση. Στην συνέχεια θα πρέπει να γίνει η επιλογή της πλακέτας NodeMCU μέσα από το μενού **Tools > Board > NodeMCU 1.0** όπως απεικονίζεται και παρακάτω (Εικόνα 6.10).



Εικόνα 6.10: Επιλογή πλακέτας NodeMCU

6.2.4 Φόρτωση Κώδικα

Για τις ανάγκες του συγκεκριμένου σεναρίου, δηλαδή την λήψη τιμών θερμοκρασίας και υγρασίας από τον αισθητήρα DHT11 και την αποστολή των δεδομένων στην πλατφόρμα, χρησιμοποιήθηκε ο παρακάτω κώδικας (Εικόνα 6.11).

```
#include "DHT.h"
#define DHTPIN 14
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

#include "UbidotsMicroESP8266.h"
#define TOKEN "BBFF-HQwFylk2XILQmhpnj5ljDtnfTWgF4" // Put here your Ubidots TOKEN
#define WIFISSID "unsecure"
#define PASSWORD "xxxxxxxxxx"

Ubidots client(TOKEN);
unsigned long lastMillis = 0;

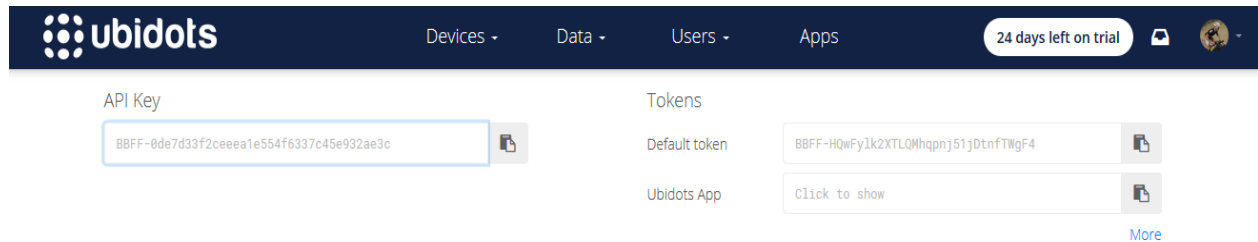
void setup() {
  Serial.begin(115200);
  dht.begin();
  delay(10);
  client.wifiConnection(WIFISSID, PASSWORD);
}

void loop() {
  if (millis() - lastMillis > 10000) { //every 10S
    float humedad = dht.readHumidity();
    float temperatura = dht.readTemperature();

    lastMillis = millis();
    client.add("humidity", humedad);
    client.add("temperature", temperatura);
    client.sendAll(true);
  }
}
```

Εικόνα 6.11: Κώδικας για την αποστολή των δεδομένων

Στον συγκεκριμένο κώδικα, στο πρώτο κομμάτι έχουν ενσωματωθεί οι βιβλιοθήκες που αφορούν τον αισθητήρα DHT11 θερμοκρασίας και υγρασίας (DHT.h) και την πλατφόρμα Ubidots (UbidotsMicroESP8266.h). Στο δεύτερο κομμάτι ορίζουμε τις παραμέτρους της σύνδεσης της IoT συσκευής με την πλατφόρμα όπως το SSID, τον κωδικό του ασυρμάτου δικτύου καθώς και το **TOKEN** (#define TOKEN) που μας δίνεται από την πλατφόρμα κατά την δημιουργία του λογαριασμού (Εικόνα 6.12).



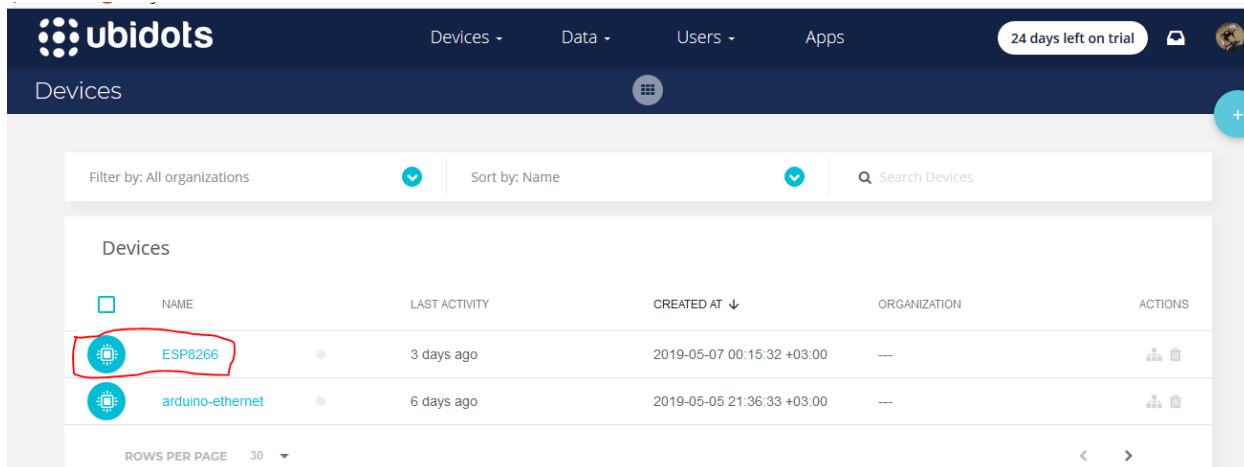
Εικόνα 6.12: Ubidots TOKEN

Κάθε αίτημα προς την πλατφόρμα απαιτεί την χρήση του συγκεκριμένου **TOKEN**. Όλες οι API κλήσεις γίνονται βάσει αυτού του **TOKEN**. Όπως φαίνεται και στην Εικόνα 6.12 υπάρχουν δύο τύποι κλειδιών στον Ubidots λογαριασμό του κάθε χρήστη:

- **TOKEN:** Προσωρινό κλειδί που χρησιμοποιείται στα API αιτήματα
- **API key:** Μοναδικό και αμετάβλητο κλειδί που χρησιμοποιείται μόνο για τη δημιουργία TOKENS

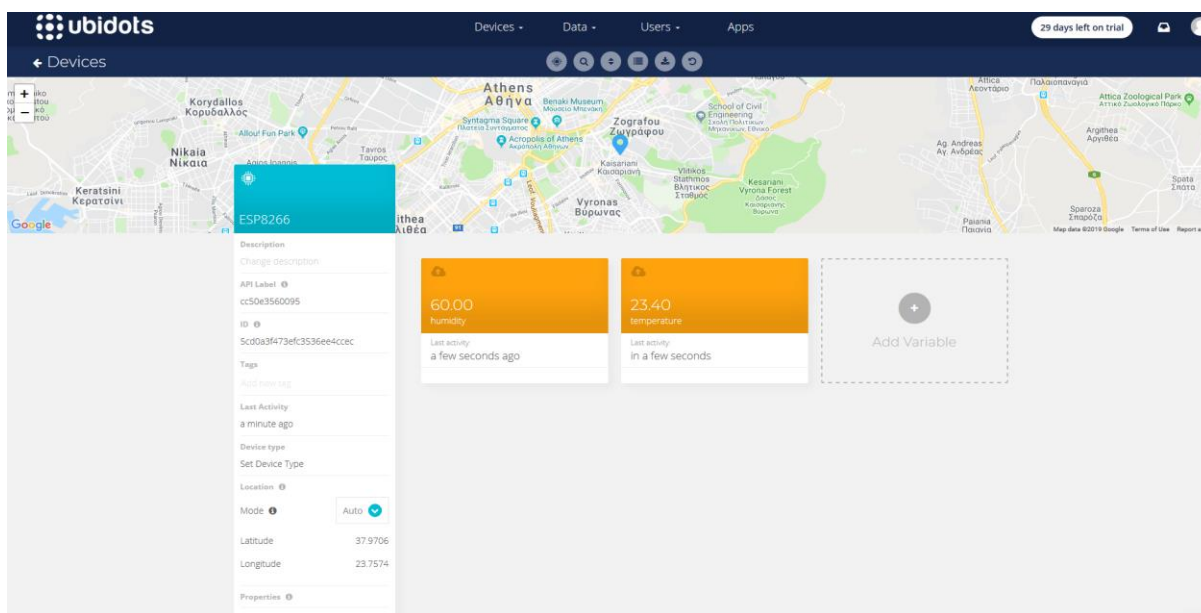
6.2.5 Αναγνώριση Συσκευής – Απεικόνιση Δεδομένων

Εάν όλα τα παραπάνω βήματα έχουν γίνει σωστά θα πρέπει πλέον η συσκευή να έχει εμφανιστεί στον κατάλογο **Devices** όπως φαίνεται στην Εικόνα 6.13.



Εικόνα 6.13: Εγγραφή συσκευής στην πλατφόρμα

Επιλέγοντας την συγκεκριμένη συσκευή, μπορούμε να δούμε τις μεταβλητές τις οποίες περιλαμβάνει και τις τιμές αυτών, που στην περίπτωσή μας είναι η θερμοκρασία και η υγρασία. Επίσης, απεικονίζονται στοιχεία όπως η γεωγραφική τοποθεσία της συσκευής με την χρήση Geolocation API, καθώς και η χρονική διάρκεια κατά την οποία η συσκευή είναι συνδεδεμένη. Τα στοιχεία αυτά απεικονίζονται στην Εικόνα 6.14.

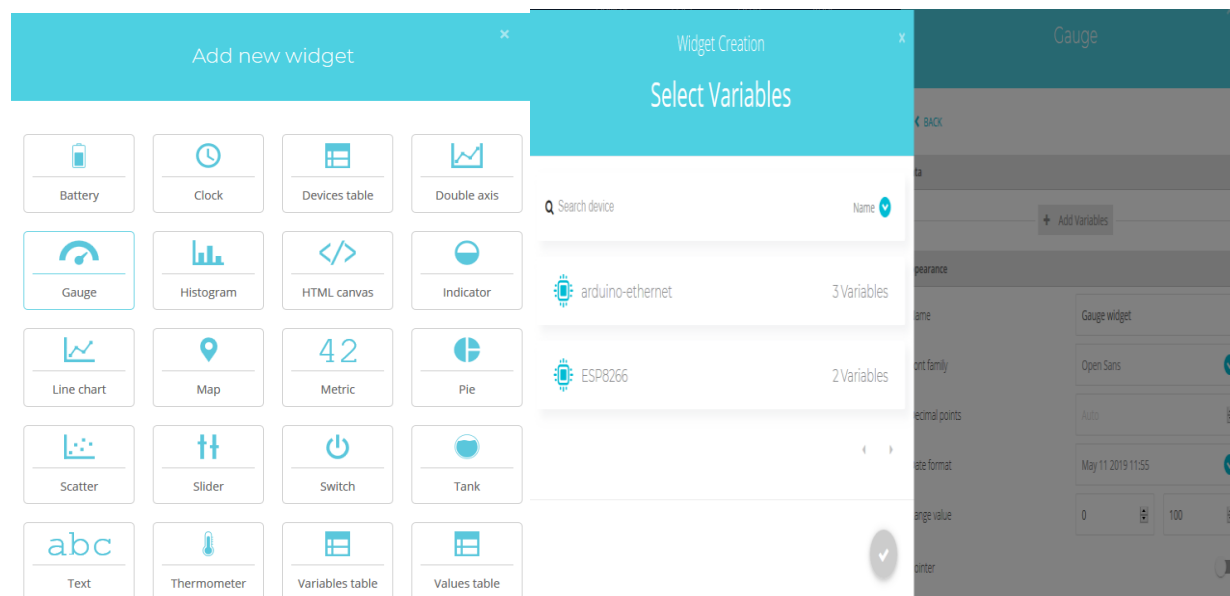


Εικόνα 6.14: Απεικόνιση παραμέτρων συσκευής

Πλέον είμαστε σε θέση να χρησιμοποιήσουμε όλες τις δυνατότητες – λειτουργίες που μας παρέχει η πλατφόρμα όπως η δημιουργία προσαρμοσμένων πινάκων (Dashboards), ανάλυση δεδομένων, ειδοποιήσεων μέσω email ή sms, ενεργοποίηση συμβάντων και άλλων πολλών τα οποία αναλύονται παρακάτω.

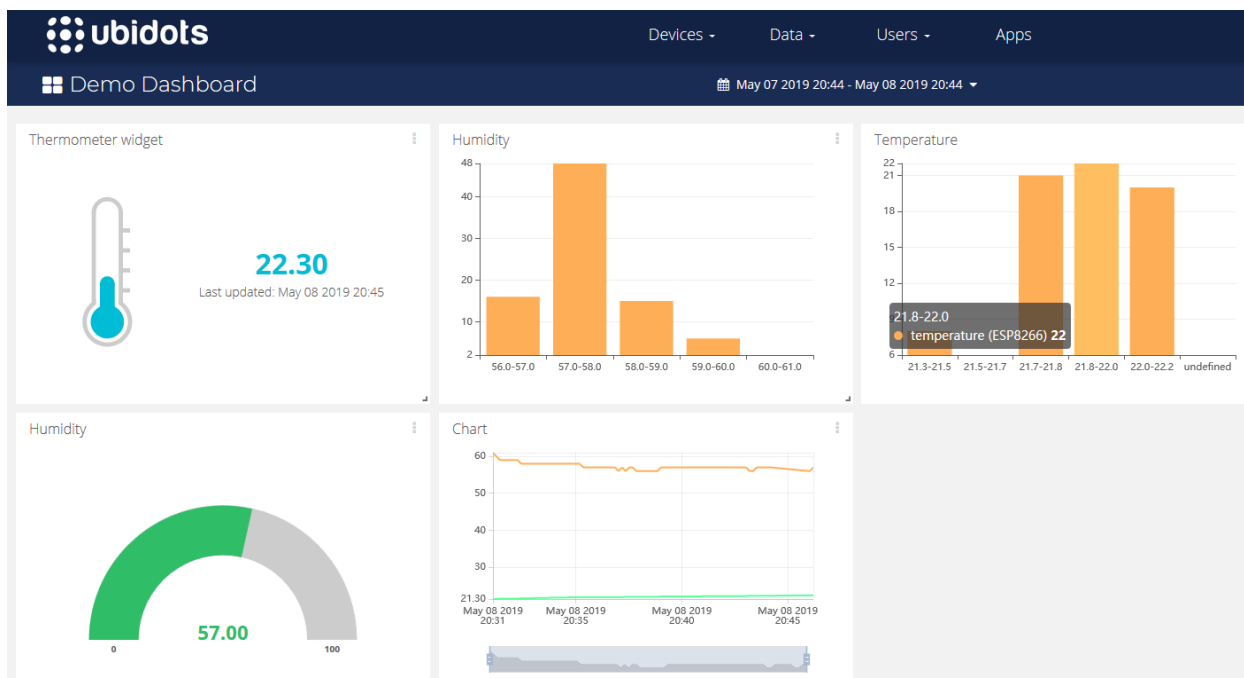
6.2.6 Δημιουργία Πινάκων (Dashboards)

Μέσω της πλατφόρμας Ubidots μπορούμε να δημιουργήσουμε προσαρμοσμένα στις ανάγκες μας **Dashboards**, με τη χρήση διαφόρων Widgets που παρέχονται από την πλατφόρμα, μέσω των οποίων θα απεικονίζονται οι τιμές των μεταβλητών της επιλογής μας όπως διακρίνουμε στην Εικόνα 6.15.



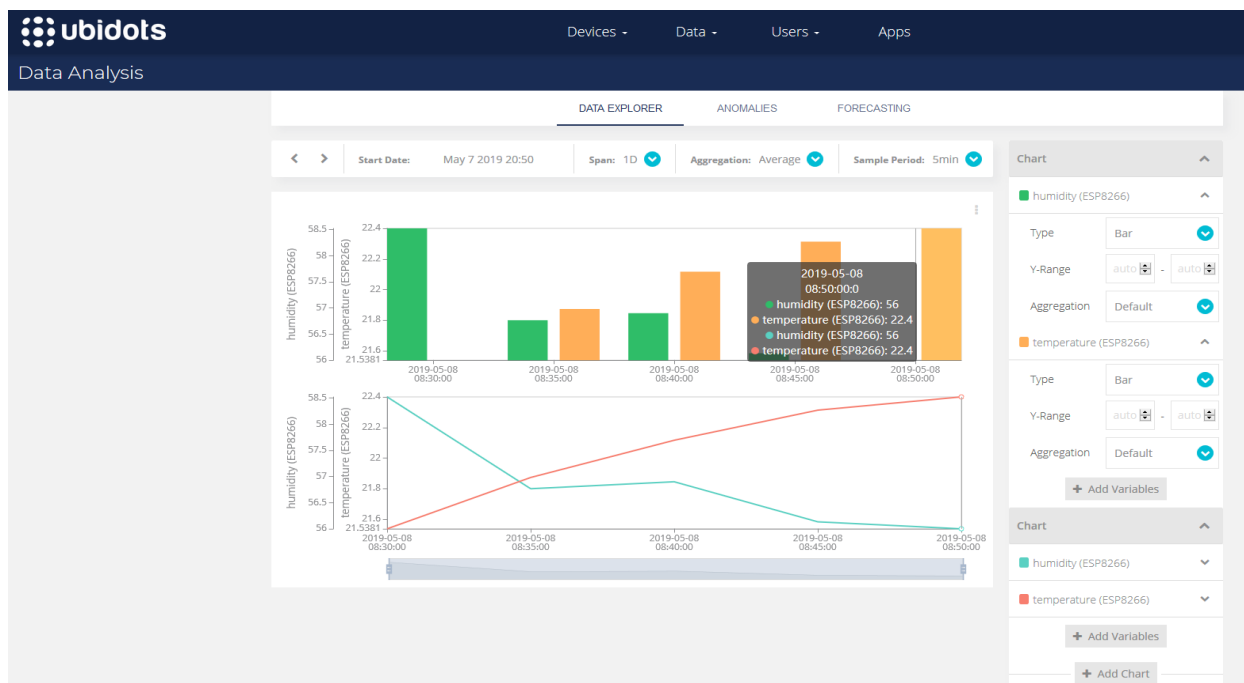
Εικόνα 6.15: Δημιουργία Widget

Μετά την ολοκλήρωση των απαιτούμενων ενεργειών σχετικά με την επιλογή και παραμετροποίηση των widgets, το αποτέλεσμα θα είναι πλέον διαφορετικό στην αρχική σελίδα της πλατφόρμας (Εικόνα 6.16). Στην Εικόνα διακρίνονται τέσσερα διαφορετικά **Widgets** (Gauge, Thermometer, Histogram, Line chart) τα οποία μας βοηθούν στην καλύτερη απεικόνιση των δεδομένων.



Εικόνα 6.16: Δημιουργία προσαρμοσμένου Dashboard

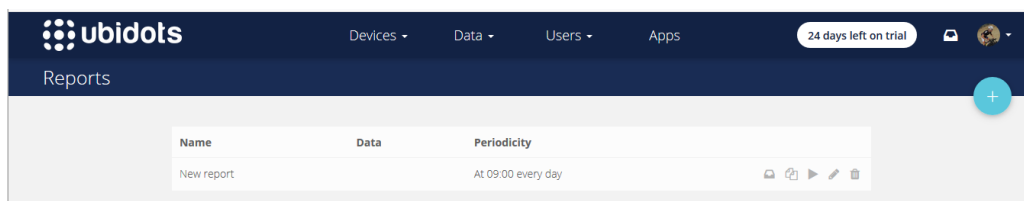
Ένα άλλο σημαντικό στοιχείο είναι η ανάλυση των δεδομένων και η εξαγωγή σημαντικών πληροφοριών. Στην Εικόνα 6.17 διακρίνουμε ένα διάγραμμα κατά το οποίο παρουσιάζεται η διακύμανση των τιμών των μεταβλητών θερμοκρασίας και υγρασίας για συγκεκριμένα χρονικά διαστήματα.



Εικόνα 6.17: Ανάλυση Δεδομένων

Βάσει των δεδομένων που συλλέγονται μπορούμε να προχωρήσουμε στην δημιουργία αναφορών για συγκεκριμένα χρονικά διαστήματα και την αποστολή αυτών στο προσωπικό μας email. Η διαδικασία έχει ως εξής:

- Επιλέγοντας **Data > Reports**, μας εμφανίζεται το παρακάτω μενού (Εικόνα 6.18) κατά το οποίο μπορούμε να επιλέξουμε το όνομα της αναφοράς καθώς και το χρονικό διάστημα το οποίο θα στέλνεται όπως για παράδειγμα κάθε μέρα στις 9:00 το βράδυ.

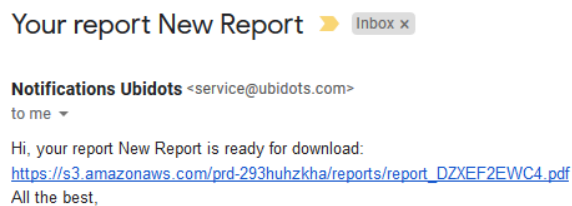


Εικόνα 6.18: Δημιουργία αναφοράς

- Στην συνέχεια ορίζουμε τις παραμέτρους (Όνομα αναφοράς, Φορμάτ, Μήνυμα κειμένου, Παραλήπτες) όπως απεικονίζονται παρακάτω (Εικόνα 6.19).

Εικόνα 6.19: Ορισμός παραμέτρων αναφοράς

Από την στιγμή που θα ορισθούν οι παράμετροι θα μπορούμε να λαμβάνουμε στο email μας την αναφορά (Εικόνα 6.20) στο χρονικό διάστημα το οποίο έχουμε επιλέξει.



Εικόνα 6.20: Αποστολή αναφοράς

Η αναφορά είναι σε μορφή pdf όπως απεικονίζεται παρακάτω (Εικόνα 6.21).

Page 1 of 3

New report

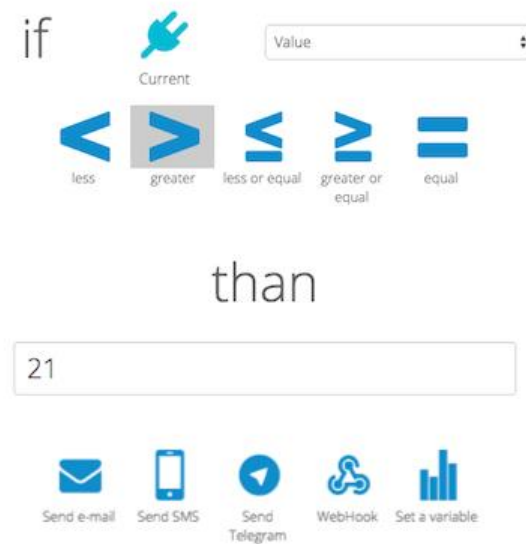
Hour and date: 2019-05-08 20:47:32 EEST
Period: 2019-05-08 00:00:00 EEST to 2019-05-08 23:59:59 EEST

Date	ESP8266 temperature(raw)	ESP8266 humidity(raw)
2019-05-08 20:31:05	21.3	61.0
2019-05-08 20:31:15	21.3	60.0
2019-05-08 20:31:25	21.4	59.0
2019-05-08 20:31:35	21.4	59.0
2019-05-08 20:31:45	21.4	59.0
2019-05-08 20:31:55	21.4	59.0
2019-05-08 20:32:05	21.4	59.0
2019-05-08 20:32:15	21.4	59.0
2019-05-08 20:32:25	21.5	58.0
2019-05-08 20:32:35	21.5	58.0
2019-05-08 20:32:45	21.5	58.0
2019-05-08 20:32:55	21.5	58.0
2019-05-08 20:33:06	21.6	58.0
2019-05-08 20:33:40	21.6	58.0
2019-05-08 20:33:50	21.7	58.0
2019-05-08 20:34:00	21.7	58.0
2019-05-08 20:34:10	21.7	58.0
2019-05-08 20:34:20	21.7	58.0

Εικόνα 6.21: Αναφορά σε μορφή PDF

6.2.7 Ενεργοποίηση Συμβάντων (Trigger Actions)

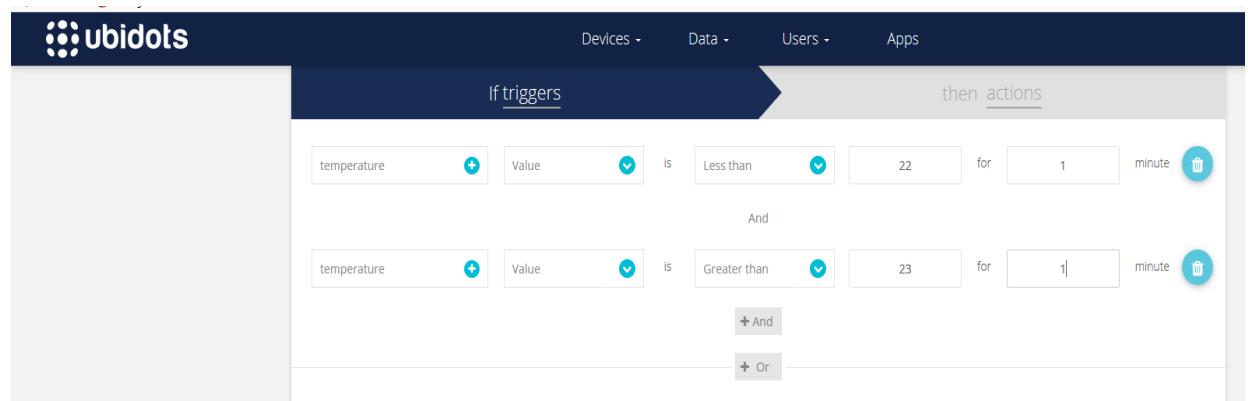
Μέσω των δεδομένων που συλλέγονται από τις συσκευές και αποστέλλονται στην πλατφόρμα μπορούν να εκτελεσθούν συγκεκριμένες ενέργειες, όπως για παράδειγμα η αποστολή μας ειδοποίησης (notification) μέσω email μέσω της οποίας θα γίνεται ενημέρωση ότι κάποιες τιμές ξεπέρασαν τα κανονικά επίπεδα ή ακόμη και η ενεργοποίηση μηχανισμών για την αποφυγή καταστροφών. Η διαδικασία έχει της μορφής **if** **than else** όπως φαίνεται στην Εικόνα 6.22.



Εικόνα 6.22: Ενεργοποίηση Συμβάντων

Η διαδικασία παραμετροποίησης έχει ως εξής:

- Επιλέγοντας: **Data > Events**, επιλέγουμε δημιουργία νέου συμβάντος (event).
- Στο μενού που μας εμφανίζεται ορίζουμε τον κανόνα που επιθυμούμε (Εικόνα 6.23).



Εικόνα 6.23: Δημιουργία Κανόνα

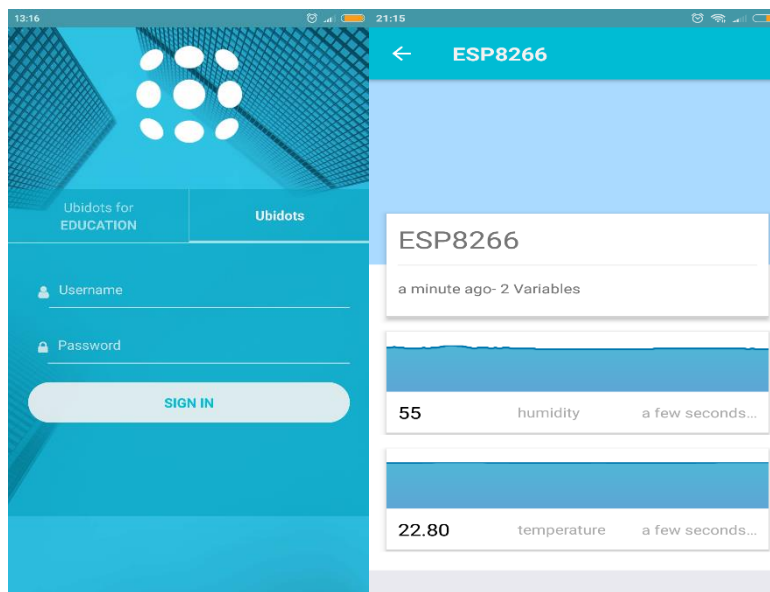
Στον συγκεκριμένο κανόνα έχουμε ορίσει ότι, εάν η τιμή της θερμοκρασίας είναι μικρότερη των 22 βαθμών ή μεγαλύτερη των 23 βαθμών τότε να εκτελεσθεί η ενέργεια η οποία σχετίζεται με την αποστολή ειδοποίησης μέσω email (Εικόνα 6.24).

The screenshot shows the Ubidots web interface. At the top, there's a navigation bar with 'ubidots' logo and links for 'Devices', 'Data', 'Users', and 'Apps'. Below this, there's a tabbed interface with 'If triggers' and 'then actions'. The 'then actions' tab is active, showing an 'ACTIVE TRIGGER' section with a 'BACK TO NORMAL' link. The main configuration area is for an email alert. It has a 'Subject' field with a dropdown menu showing 'Variable name' and 'alert!'. Below that is a 'Message' field with a dropdown menu showing 'Hey there, Variable name was Last value at Trigger timestamp'. There's also a 'Email Address' field with the value 'anastasiou1982@gmail.com' and a note 'Add comma separated emails'.

Εικόνα 6.24: Αποστολή Ειδοποίησης

6.2.8 Χρήση Mobile Εφαρμογής

Για την πλατφόρμα Ubidots διατίθεται mobile εφαρμογή για συσκευές android και iOS. Στην συγκεκριμένη περίπτωση έγινε λήψη και εγκατάσταση της εφαρμογής σε android κινητό. Η χρήση της εφαρμογής είναι αρκετά απλή, αρκεί να βάλουμε τα στοιχεία του λογαριασμού μας. Από την στιγμή που θα συνδεθούμε μπορούμε πλέον να δούμε τις συσκευές οι οποίες είναι συνδεδεμένες (Εικόνα 6.25) και τις τιμές που έχουν ληφθεί.

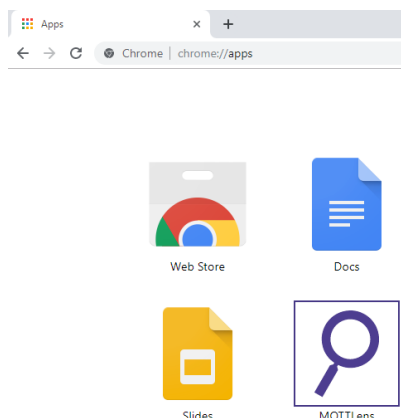


Εικόνα 6.25: Χρήση Mobile εφαρμογής για android συσκευή

6.2.9 Έλεγχος MQTT Επικοινωνίας

Στο παρακάτω παράδειγμα παρουσιάζεται η εξομίωση μιας MQTT επικοινωνίας (publish / subscribe) μεταξύ ενός **MQTT Broker**, στην περίπτωσή μας του **Ubidots Broker** και ενός client, στην περίπτωσή μας το **MQTTLens** το οποίο είναι μια εφαρμογή του Google Chrome και η οποία θα δημοσιεύσει (publish) **MQTT topics** (υγρασία, θερμοκρασία, πίεση) στον **Ubidots Broker**. Στο MQTT πρωτόκολλο, η λέξη "topic" αναφέρεται σε μια συμβολοσειρά UTF-8 που χρησιμοποιεί ο **broker** για να φιλτράρει μηνύματα για κάθε συνδεδεμένο πελάτη (client). Η διαδικασία έχει ως εξής:

- Εγκατάσταση της εφαρμογής (Εικόνα 6.26) MQTTLens στον Chrome



Εικόνα 6.26: Εγκατάσταση της εφαρμογής MQTTLens στον Chrome

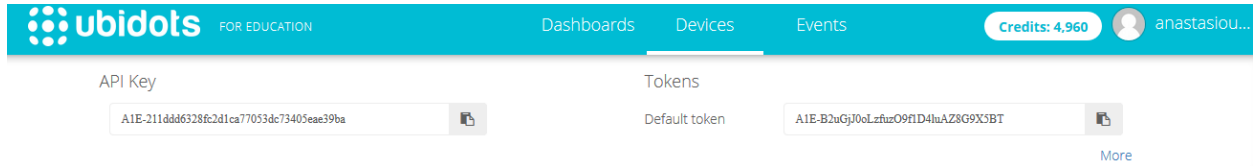
- Παραμετροποίηση του MQTTLens ορίζοντας το **hostname** και την **TCP Port** του **MQTT Broker** (<https://ubidots.com/docs/hw/#mqtt>) με τις τιμές: things.ubidots.com – 1883.

A screenshot of the 'Add a new Connection' dialog box in the MQTTLens application. The dialog has a title bar with a close button. It contains several sections: 'Connection Details' with fields for 'Connection name' (ubidots), 'Connection color scheme' (a green bar), 'Hostname' (tcp:// things.ubidots.com), and 'Port' (1883). There is a 'Client ID' field with a generated ID and a 'Generate a random ID' button. Below that are checkboxes for 'Clean Session' and 'Automatic Connection', and a 'Keep Alive' field set to 120 seconds. The 'Credentials' section at the bottom has fields for 'Username' (A1E-B2uGjJ0oLzfuzO9f1D4luAZ8G9X5BT) and 'Password' (masked with dots).

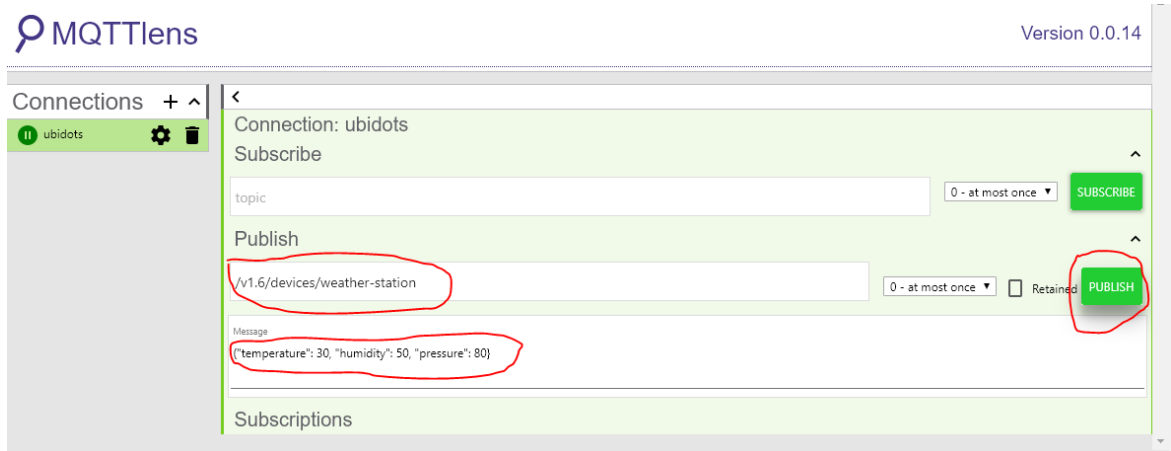
Εικόνα 6.27: Συμπλήρωση στοιχείων Username και Password

Στο πεδία **Username** και **Password** συμπληρώνουμε την τιμή του **Default token** την οποία την λαμβάνουμε από τον λογαριασμό μας (Εικόνα 6.27) στην ιστοσελίδα:

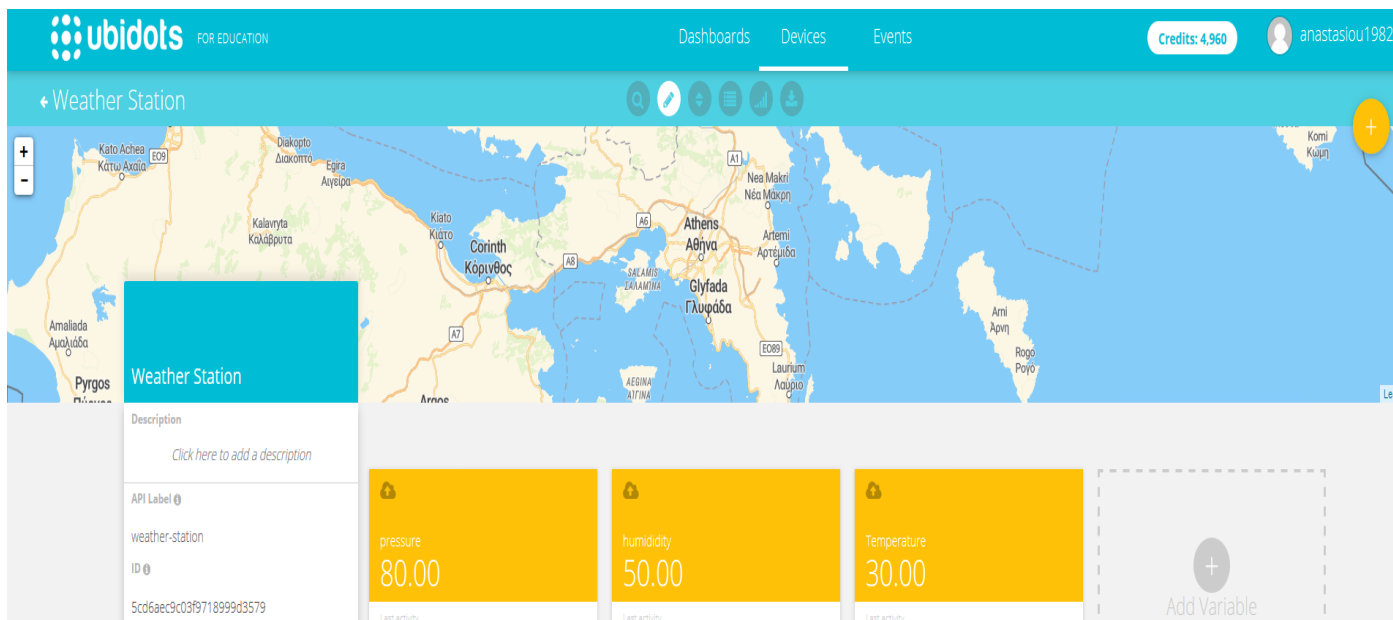
<https://app.ubidots.com/ubi/datasources>



- Αποστολή τιμών για τις μεταβλητές (θερμοκρασία, υγρασία και πίεση) προς τον MQTT Broker κάνοντας **PUBLISH** ένα **MQTT Topic**



Εάν όλα έχουν γίνει σωστά θα πρέπει να λάβουμε στην πλατφόρμα, τις μεταβλητές και τις τιμές που έχουμε ορίσει στην **MQTTlens** εφαρμογή .



6.2.10 Συμπεράσματα

Από τα στάδια των δοκιμών συμπεραίνουμε ότι πρόκειται για μία πλατφόρμα η οποία παρέχει αρκετές λειτουργίες σχετικά με την απεικόνιση και την ανάλυση των δεδομένων. Επίσης, είναι φανερό ότι πρόκειται για μία πλατφόρμα αρκετά παραμετροποιήσιμη με τέτοιον τρόπο ώστε να μπορεί να διαχειριστεί από έναν απλό χρήστη. Διαθέτει αρκετά αναλυτικούς οδηγούς σχετικά με την ανάπτυξη διαφόρων υπηρεσιών όπως η ενσωμάτωση με πλατφόρμες τρίτων κατασκευαστών και η δημιουργία συμβάντων. Όλες οι δοκιμές πραγματοποιήθηκαν μέσω ενός εύχρηστου και διαδραστικού περιβάλλοντος, χωρίς να απαιτείται η εμπλοκή με κάποια γλώσσα προγραμματισμού. Τέλος, αξίζει να αναφερθεί πως παρέχεται η δυνατότητα χρήσης όλων των υπηρεσιών της πλατφόρμας, με μέγιστη χρονική διάρκεια τριάντα ημερών.

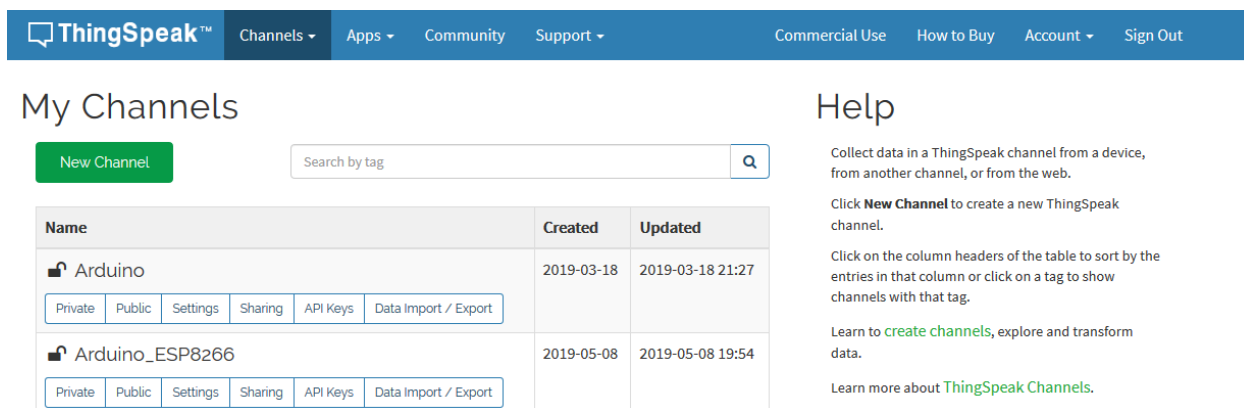
6.3 Δοκιμή Πλατφόρμας ThingSpeak

Για την δοκιμή των λειτουργιών της ThingSpeak πλατφόρμας έχει χρήση του ίδιου υλικού όπως και στην προηγούμενη πλατφόρμα (Ubidots). Το σενάριο όπως και πριν σχετίζεται με την λήψη τιμών θερμοκρασίας και υγρασίας από τον αισθητήρα DHT11, ο οποίος είναι εγκατεστημένος στην NodeMCU πλακέτα και την αποστολή των δεδομένων στην πλατφόρμα. Η διαφοροποίηση είναι ο τρόπος επικοινωνίας της συσκευής με την πλατφόρμα, οι λειτουργίες (features) της πλατφόρμας καθώς και στοιχεία όπως ο κώδικας και οι βιβλιοθήκες. Η διαδικασία περιλαμβάνει τα εξής βήματα:

- Δημιουργία λογαριασμού στο Thingspeak.
- Δημιουργία καναλιού με δύο πεδία (θερμοκρασία και υγρασία).
- Δημιουργία API κλειδιού.
- Λήψη της βιβλιοθήκης ThingSpeak (<https://github.com/mathworks/thingspeak-arduino>).
- Λήψη της βιβλιοθήκης για τον αισθητήρα DHT11 (<https://github.com/adafruit/DHT-sensor-library>).

6.3.1 Διαδικασία Δοκιμής

Το πρώτο βήμα για την διαδικασία της δοκιμής είναι η δημιουργία ενός λογαριασμού στην ιστοσελίδα https://thingspeak.com/users/sign_up, η οποία συνδέεται με την εταιρεία MathWorks, η οποία είναι υπεύθυνη για την ανάπτυξη των προϊόντων MATLAB και Simulink. Η αρχική σελίδα που θα συναντήσουμε απεικονίζεται παρακάτω (Εικόνα 6.28).



The screenshot shows the ThingSpeak website interface. At the top is a navigation bar with links: Channels, Apps, Community, Support, Commercial Use, How to Buy, Account, and Sign Out. Below the navigation bar is the 'My Channels' section, which includes a 'New Channel' button and a search bar. A table lists the user's channels:

Name	Created	Updated
Arduino	2019-03-18	2019-03-18 21:27
Arduino_ESP8266	2019-05-08	2019-05-08 19:54

Each channel row has buttons for Private, Public, Settings, Sharing, API Keys, and Data Import / Export. To the right of the table is a 'Help' section with instructions on how to create a new channel and use the table's sorting and filtering features.

Εικόνα 6.28: Αρχική σελίδα ThingSpeak

Επόμενο βήμα όπως φαίνεται και στην Εικόνα 6.29 είναι η δημιουργία ενός καναλιού.

New Channel

Name

Description

Field 1 ☒

Field 2 ☒

Field 3 ☐

Field 4 ☐

Field 5 ☐

Field 6 ☐

Field 7 ☐

Field 8 ☐

Metadata

Tags
(Tags are comma separated)

Link to External Site

Link to GitHub

Εικόνα 6.29: Δημιουργία Καναλιού

Τα κανάλια αποθηκεύουν όλα τα δεδομένα που συλλέγει μια εφαρμογή ThingSpeak. Κάθε κανάλι μπορεί να χωρέσει έως οκτώ πεδία, στην συγκεκριμένη περίπτωση έχουμε δημιουργήσει δύο πεδία (θερμοκρασία και υγρασία). Αφού ληφθούν τα δεδομένα σε ένα κανάλι, τότε μπορεί να γίνει χρήση διαφόρων εφαρμογών της ThingSpeak πλατφόρμας για την ανάλυση και την απεικόνιση αυτών. Αφού ολοκληρώσουμε την παραμετροποίηση θα μπορούμε πλέον να δούμε να εμφανίζεται στον κατάλογο **Channels**, το κανάλι που έχουμε δημιουργήσει, το οποίο στην συγκεκριμένη περίπτωση είναι το Arduino_ESP8266 και απεικονίζεται παρακάτω (εικόνα 6.30).

My Channels

[New Channel](#)

Search by tag

Name	Created	Updated
Arduino Private Public Settings Sharing API Keys Data Import / Export	2019-03-18	2019-03-18 21:27
Arduino_ESP8266 Private Public Settings Sharing API Keys Data Import / Export	2019-05-08	2019-05-08 19:54

Εικόνα 6.30: Εμφάνιση καναλιού στην πλατφόρμα

Επιλέγοντας το κανάλι που έχουμε δημιουργήσει, μάς εμφανίζονται διάφορα στοιχεία (Εικόνα 6.31) όπως το ID του καναλιού, ο δημιουργός του καναλιού, τα APIs κλειδιά τα οποία χρησιμοποιούνται για την ανάγνωση των δεδομένων και άλλα.

Arduino_ESP8266

Channel ID: **776844**

Author: **nanastas**

Access: Public

[Private View](#)

[Public View](#)

[Channel Settings](#)

[Sharing](#)

[API Keys](#)

[Data Import / Export](#)

Εικόνα 6.31: Ιδιότητες Καναλιού

Τα APIs κλειδιά (Εικόνα 6.32) επιτρέπουν την εγγραφή ή την ανάγνωση δεδομένων σε ένα κανάλι. Δημιουργούνται αυτόματα κατά την διαδικασία δημιουργίας ενός νέου καναλιού.

Arduino_ESP8266

Channel ID: 776844

Author: nanastas

Access: Public

[Private View](#)[Public View](#)[Channel Settings](#)[Sharing](#)[API Keys](#)

Write API Key

Key

LT6VD583CGUI395N

Generate New Write API Key

Read API Keys

Key

I7R8U89WARYNXW67

Note

Save Note

Delete API Key

Generate New Read API Key

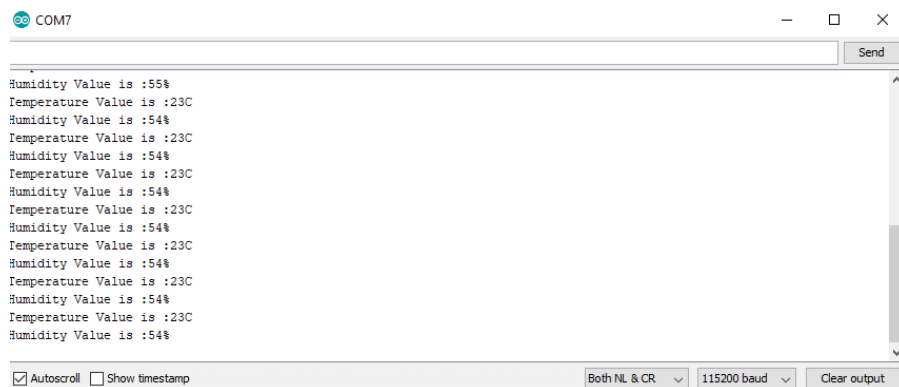
Εικόνα 6.32: Δημιουργία APIs κλειδιών

Πλέον έχουμε όλες τις απαραίτητες πληροφορίες για την παραμετροποίηση και την φόρτωση του κώδικα στην πλακέτα. Ο κώδικας ο οποίος απεικονίζεται παρακάτω (Εικόνα 6.33) χρησιμοποιήθηκε στην συγκεκριμένη περίπτωση για την αποστολή των δεδομένων από τον αισθητήρα DHT11 στην πλατφόρμα.



Εικόνα 6.33: Κώδικας ThingSpeak

Φορτώνοντας τον κώδικα, βλέπουμε στην Εικόνα 6.34 τις τιμές που λαμβάνονται από τον αισθητήρα, οι οποίες θα πρέπει να απεικονίζονται και στην πλατφόρμα (Εικόνα 6.35). Στον παραπάνω κώδικα θα πρέπει να ορίσουμε τα στοιχεία του ασυρμάτου δικτύου (**ssid**, **password**) καθώς και τις τιμές των πεδίων του καναλιού (**myChannelNumber**, **myWriteAPIKey**).



Εικόνα 6.34: Λήψη τιμών υγρασίας και θερμοκρασίας

Arduino_ESP8266

Watch Tweet Like 0 Share

Channel ID: 776844

Author: nanastas

Access: Public

Private View

Public View

Channel Settings

Sharing

API Keys

Data Import / Export

+ Add Visualizations

+ Add Widgets

Export recent data

MATLAB Analysis

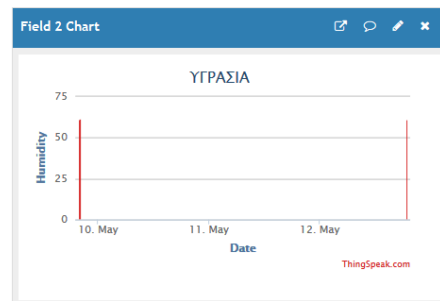
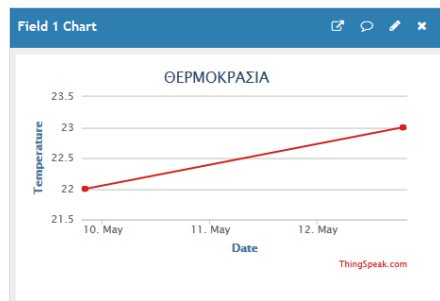
MATLAB Visualization

Channel Stats

Created: 3 days ago

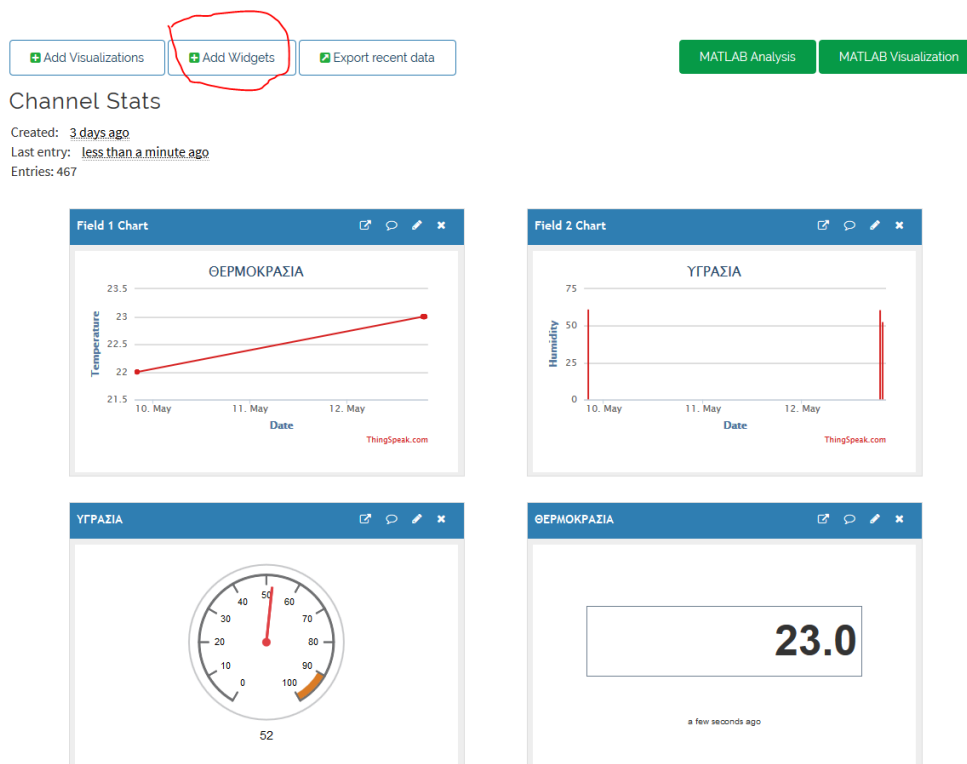
Last entry: less than a minute ago

Entries: 457



Εικόνα 6.35: Απεικόνιση τιμών στην Πλατφόρμα

Με την προσθήκη widgets, μπορούμε να προσαρμόσουμε την απεικόνιση των δεδομένων όπως φαίνεται στην Εικόνα 6.36.



Εικόνα 6.36: Προσθήκη Widgets

6.3.2 Ενσωμάτωση με MATLAB

Επίσης, με την χρήση MATLAB κώδικα μπορούμε να προχωρήσουμε σε περισσότερο εξατομικευμένη απεικόνιση δεδομένων και ανάλυση στοιχείων όπως φαίνεται στις παρακάτω Εικόνες 6.37 και 6.38.

Name

ΧΡΗΣΗ ΙΣΤΟΓΡΑΜΑΤΟΣ ΠΑΡΑΛΛΑΓΩΝ ΔΕΔΟΜΕΝΩΝ

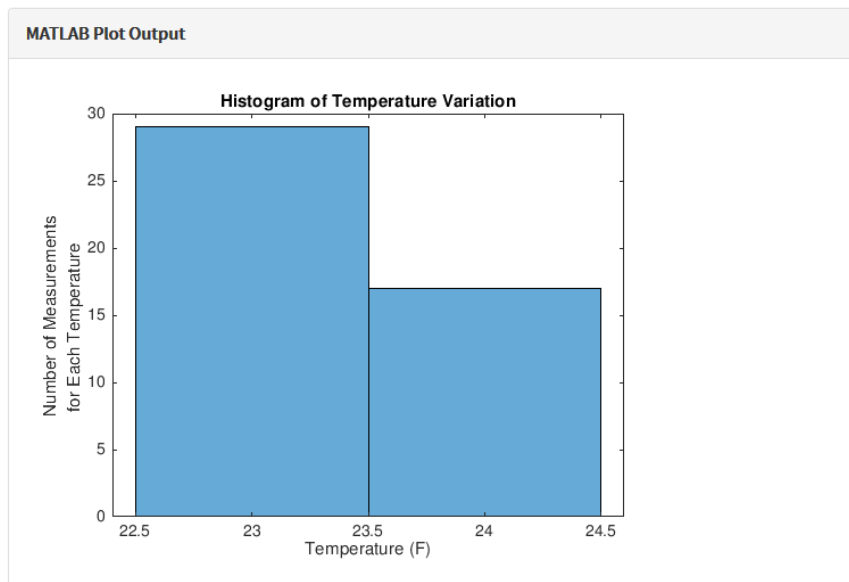
MATLAB Code

```
1 % Read temperature for the last 10 hours from a ThingSpeak channel and
2 % visualize temperature variations using the MATLAB HISTOGRAM function.
3 % Channel ID to read data from
4 readChannelID = 776844;
5
6 % Temperature Field ID
7 TemperatureFieldID = 1;
8
9 % Channel Read API Key
10 % If your channel is private, then enter the read API
11 % Key between the '' below:
12 readAPIKey = 'I7R8U89WARYNXW67';
13
14 % Get temperature data from field 4 for the last 10 hours = 10 x 60
15 % minutes. Learn more about the THINGSPEAKREAD function by going to
16 % the Documentation tab on the right side pane of this page.
17
18 tempF = thingSpeakRead(readChannelID,'Fields',TemperatureFieldID,...
19 'NumMinutes',10*60, 'ReadKey',readAPIKey);
20
21 histogram(tempF);
22 xlabel('Temperature (F)');
23 ylabel('Number of Measurements\nnewline for Each Temperature');
24 title('Histogram of Temperature Variation');
25
```

Save and Run

Save

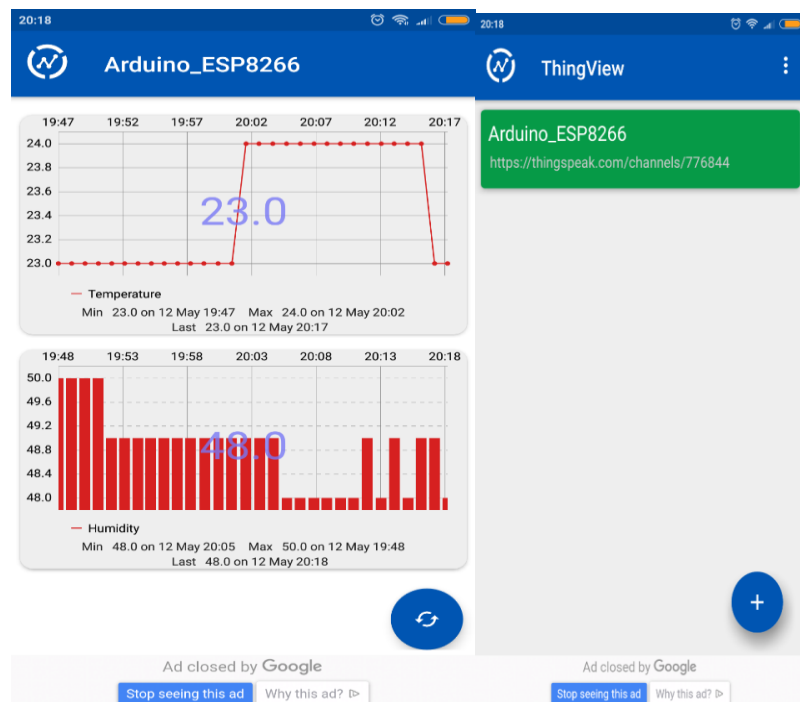
Εικόνα 6.37: MATLAB ΚΩΔΙΚΑΣ



Εικόνα 6.38: Αποτελέσματα MATLAB κώδικα

6.3.3 Χρήση Mobile Εφαρμογής

Για την ThingSpeak πλατφόρμα διατίθεται mobile εφαρμογή για συσκευές android και iOS. Στην συγκεκριμένη περίπτωση έγινε λήψη και εγκατάσταση της εφαρμογής σε android κινητό. Η χρήση της εφαρμογής είναι αρκετά απλή, αρκεί να βάλουμε τα στοιχεία του λογαριασμού μας. Από την στιγμή που θα συνδεθούμε μπορούμε πλέον να δούμε τις συσκευές οι οποίες είναι συνδεδεμένες (Εικόνα 6.39) και τις τιμές που έχουν ληφθεί.



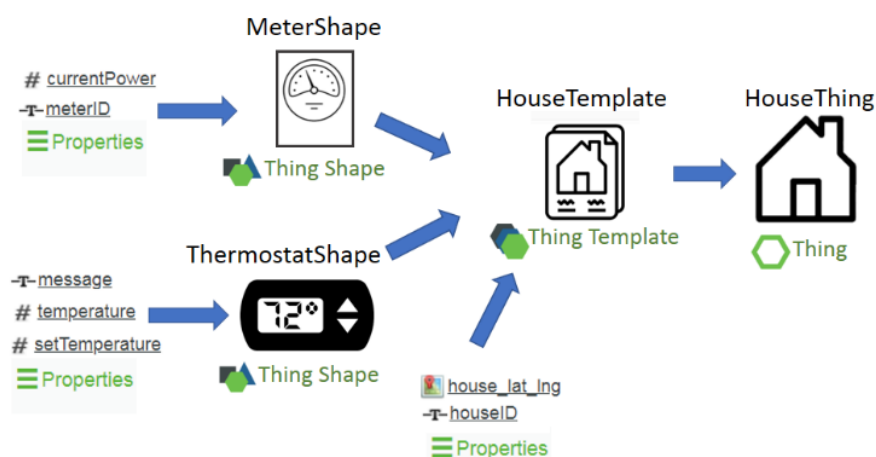
Εικόνα 6.39: Απεικόνιση Δεδομένων σε Android Συσκευή

6.3.4 Συμπεράσματα

Κατά την διεξαγωγή των δοκιμών με την συγκεκριμένη πλατφόρμα το πρώτο στοιχείο το οποίο προκαλεί εντύπωση είναι η ευχρηστία της ενσωμάτωσης με συσκευές υλικού όπως το Arduino μέσω του ThingSpeak API. Ένα από τα πλεονεκτήματα της πλατφόρμας είναι η δωρεάν παροχή φιλοξενίας για τα κανάλια δεδομένων. Όπως προαναφέρθηκε, αυτά τα κανάλια μπορεί να είναι ιδιωτικά ή δημόσια. Ουσιαστικά πρόκειται για μία πλατφόρμα αρκετά εύκολη στη χρήση για νεοφερμένους στην συγκεκριμένη τεχνολογία, ωστόσο μπορεί να παραμετροποιηθεί σε βάθος με την χρήση MATLAB κώδικα. Τέλος, κάποια από τα μειονεκτήματα που παρατηρήθηκαν αφορούν την μη γνώση της διάρκειας όπου τα δεδομένα αποθηκεύονται στους διακομιστές της πλατφόρμας καθώς και η καθυστέρηση των 15 δευτερολέπτων μεταξύ της μεταφόρτωσης των δεδομένων και της ενημέρωσης του καναλιού.

6.4 Δοκιμή Πλατφόρμας ThingWorx

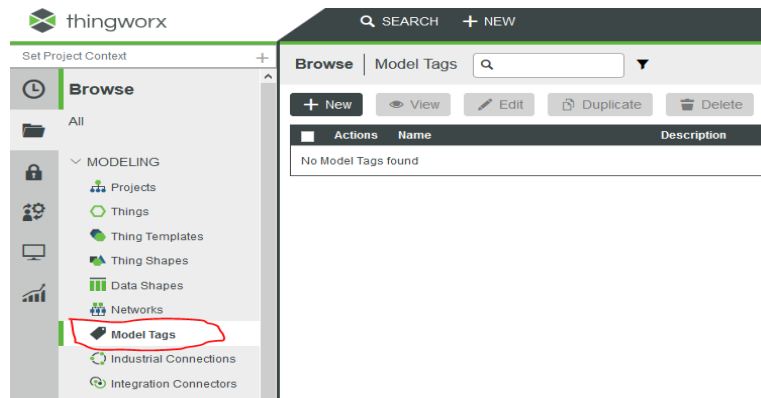
Στην συγκεκριμένη δοκιμή, δεν χρησιμοποιήθηκε κάποια συσκευή υλικού όπως στις δύο προηγούμενες περιπτώσεις δοκιμών, αλλά χρησιμοποιήθηκε ο ThingWorx διακομιστής, ο οποίος αποτελεί τον πυρήνα της πλατφόρμας, επιτρέποντας τη δημιουργία ολοκληρωμένων IoT λύσεων. Η δοκιμή βασίστηκε στην ανάπτυξη μίας IoT εφαρμογής χρησιμοποιώντας τα βασικά δομικά στοιχεία της πλατφόρμας. Επισημαίνεται ότι το συγκεκριμένο περιβάλλον διατίθεται για δοκιμαστική χρήση περιόδου τριών μηνών. Ο σχεδιασμός της εφαρμογής βασίζεται στο μοντέλο επαναχρησιμοποίησης των δομικών στοιχείων, επιτυγχάνοντας πιο ευέλικτες εφαρμογές. Μια εφαρμογή ThingWorx αποτελείται από αντικείμενα (things), το καθένα από τα οποία βασίζεται σε ένα πρότυπο (ThingTemplate) μέσω του οποίου ορίζονται τα κοινά χαρακτηριστικά και οι συμπεριφορές για ένα σύνολο οντοτήτων. Η εφαρμογή αφορά ένα σπίτι (Εικόνα 6.40) που περιλαμβάνει έναν θερμοστάτη, έναν ηλεκτρικό μετρητή και έναν προσομοιωτή δεδομένων αισθητήρων. Θα παρουσιάσουμε τον τρόπο λήψης, αποθήκευσης, ανάλυσης και απεικόνισης των δεδομένων



Εικόνα 6.40: Thingworx IoT Εφαρμογή

6.4.1 Βήμα 1 (Model Tags)

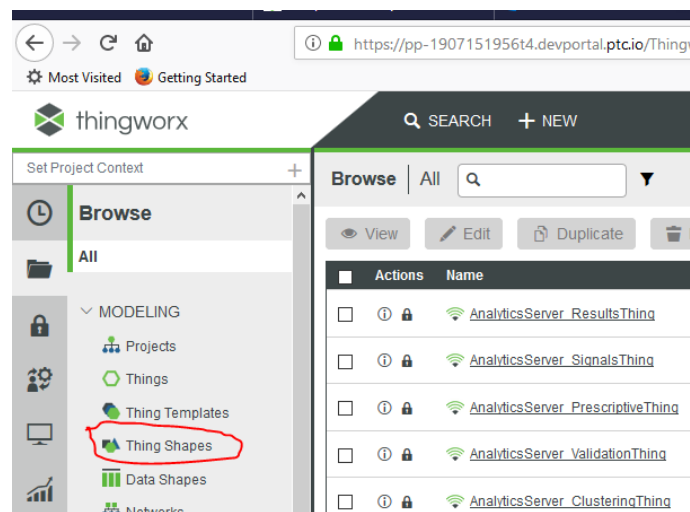
Αρχικά συνίσταται η δημιουργία ετικετών μοντέλων (Model Tags) για την καλύτερη οργάνωση της εργασίας. Οι ετικέτες μοντέλου είναι όροι που χρησιμοποιούνται για την κατηγοριοποίηση και την ομαδοποίηση αντικειμένων σε όλη την πλατφόρμα. Μέσω του εργαλείου Thingworx Composer, όπου ουσιαστικά γίνονται όλες οι ενέργειες, επιλέγουμε την δημιουργία νέου model tag (Εικόνα 6.41).



Εικόνα 6.41: Δημιουργία model tag

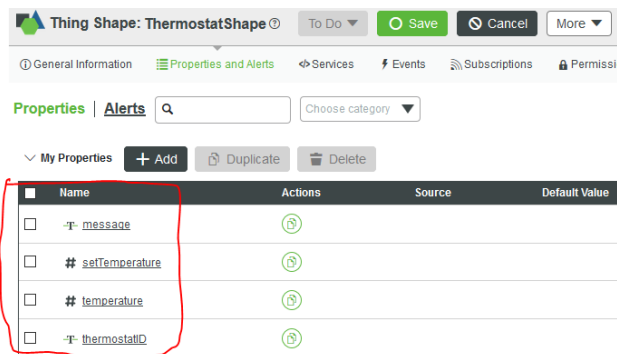
6.4.2 Βήμα 2 (Thing Shapes)

Στην συνέχεια δημιουργούμε τα Thing Shapes (Εικόνα 6.42) τα οποία είναι συστατικά που περιέχουν Ιδιότητες και Υπηρεσίες. Αν θέλαμε να το προσεγγίσουμε σε όρους προγραμματισμού της Java, είναι παρόμοια με μια διεπαφή. Στην συγκεκριμένη περίπτωση τα Thing Shapes αφορούν έναν ηλεκτρικό μετρητή και ένα θερμοστάτη.



Εικόνα 6.42: Δημιουργία Thingshape

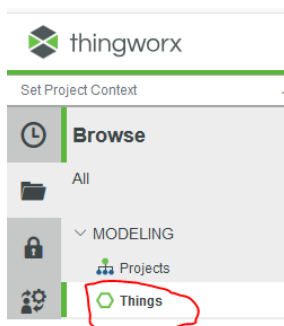
Εν συνέχεια ορίζουμε τις τέσσερις ιδιότητες του θερμοστάτη οι οποίες απεικονίζονται στην Εικόνα 6.43.



Εικόνα 6.43: Δημιουργία ιδιοτήτων

6.4.3 Βήμα 3 (Δημιουργία αντικειμένου)

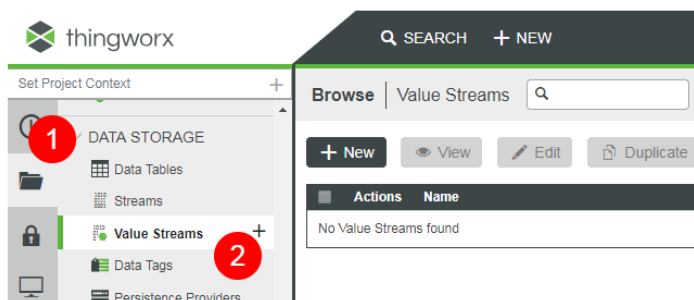
Δημιουργία αντικειμένου, η οποία πραγματοποιείται μέσω του ThingWorx Composer (Εικόνα 6.44). Το συγκεκριμένο αντικείμενο αντιπροσωπεύει ένα μεμονωμένο σπίτι



Εικόνα 6.44: Δημιουργία Αντικειμένου

6.4.4 Βήμα 4 (Αποθήκευση)

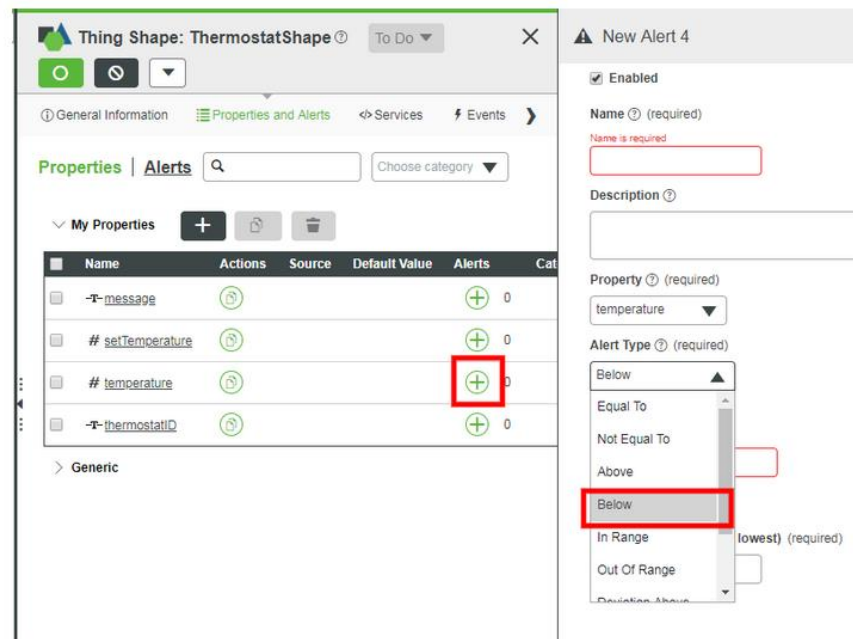
Στο συγκεκριμένο βήμα έχουμε τη δημιουργία μιας οντότητας Αποθήκευσης, για την καταγραφή των μεταβαλλόμενων τιμών των ιδιοτήτων. Αν και παρέχονται διάφοροι τρόποι αποθήκευσης των δεδομένων στον διακομιστή, στην περίπτωσή μας γίνεται χρήση της επιλογής "Value Stream" (Εικόνα 6.45), η οποία είναι ένας γρήγορος και εύκολος τρόπος για την αποθήκευση δεδομένων χρονοσειρών (times - series data).



Εικόνα 6.45: Αποθήκευση Δεδομένων

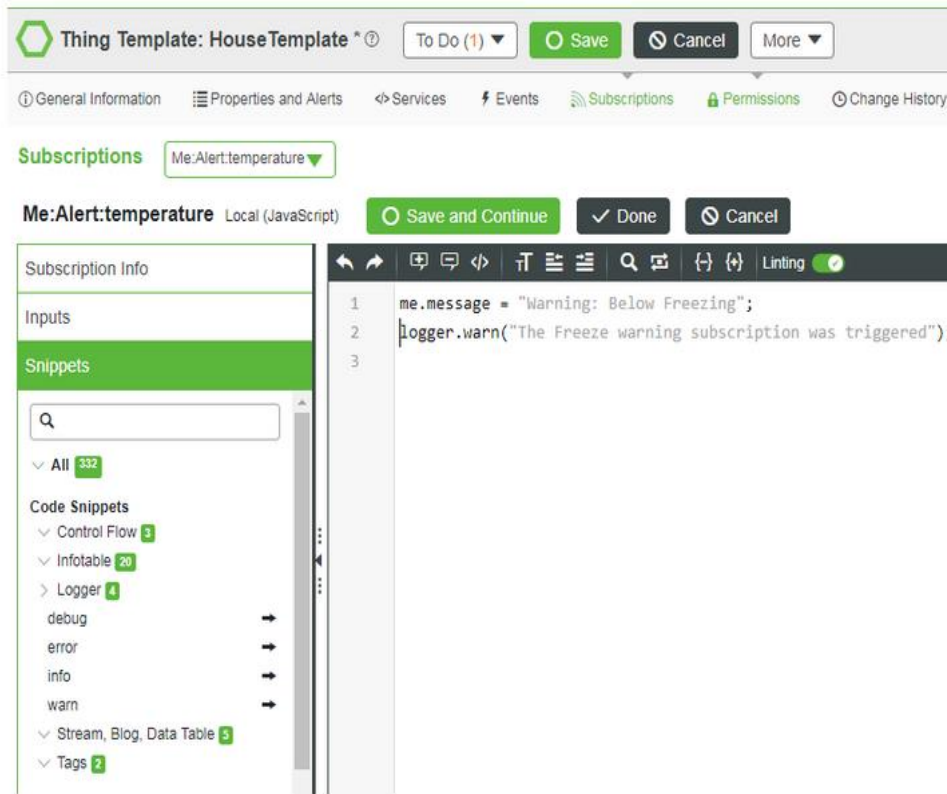
6.4.5 Βήμα 5 (Notifications)

Μέσω της πλατφόρμας υποστηρίζεται η διαδικασία δημιουργίας ειδοποιήσεων. Ένα συμβάν (event) είναι ένα προσαρμοσμένο μήνυμα που δημοσιεύεται από ένα αντικείμενο (thing), συνήθως όταν αλλάζει η τιμή μιας ιδιότητας. Στην συγκεκριμένη περίπτωση δημιουργήθηκε μια ειδοποίηση, στην περίπτωση όπου υπάρξει μεταβολή στην τιμή της θερμοκρασίας του θερμοστάτη (Εικόνα 6.46).



Εικόνα 6.46: Δημιουργία Ειδοποιήσεων

Εν συνεχεία δημιουργήθηκε ένα κομμάτι Javascript κώδικα (Εικόνα 6.47), μέσω του οποίου ορίζεται και αποστέλλεται ένα μήνυμα προειδοποίησης στο αρχείο καταγραφής (log) του ThingWorx διακομιστή.

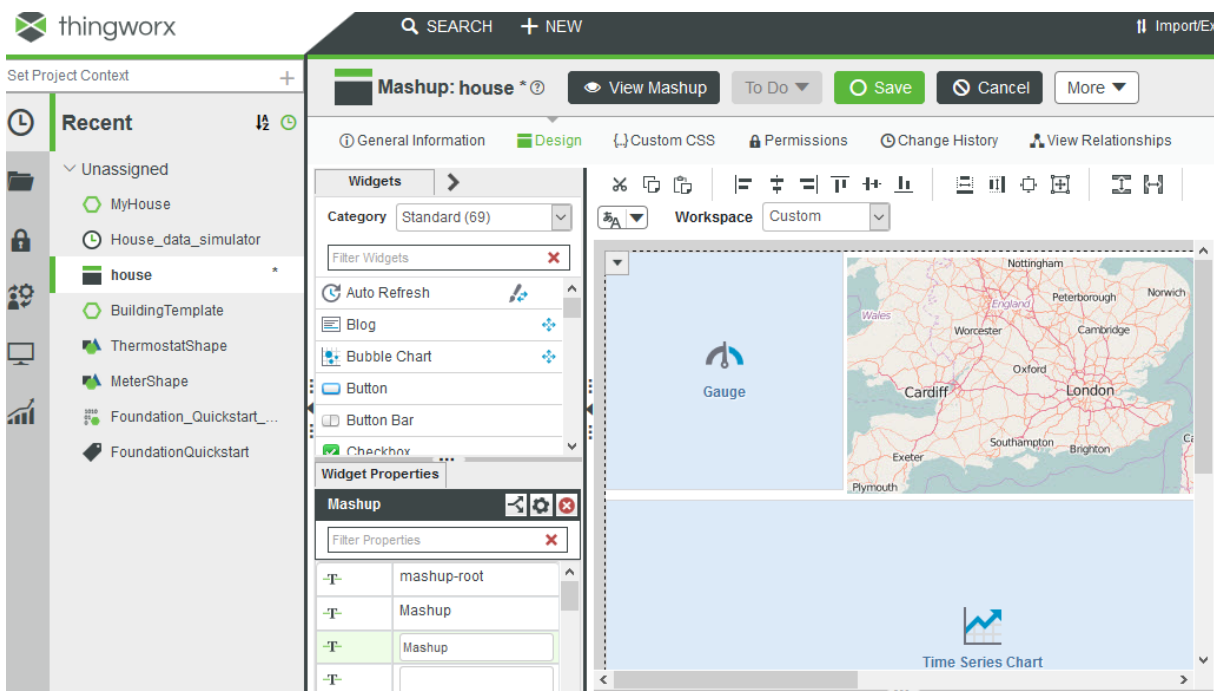


Εικόνα 6.47: Δημιουργία κώδικα μηνύματος ειδοποίησης

6.4.6 Βήμα 6 (Dynamic Dashboards)

Στην συνέχεια αναλύεται η διαδικασία δημιουργίας δυναμικών Dashboards και η προσθήκη widgets για την απεικόνιση των δεδομένων.

Μέσω της πλατφόρμας ThingWorx μπορεί ο χρήστης να δημιουργήσει προσαρμοσμένες εφαρμογές οι οποίες εμφανίζουν και αλληλοεπιδρούν με δεδομένα από πολλές πηγές. Αυτές οι εφαρμογές ιστού ονομάζονται Mashups και δημιουργούνται χρησιμοποιώντας το Mashup Builder. Ο Mashup Builder είναι ο τόπος όπου δημιουργείται η εφαρμογή, μεταφέροντας Widgets όπως διαγράμματα, χάρτες, κουμπιά σε καμβά (Εικόνα 6.48). Όλα τα στοιχεία διεπαφής χρήστη στην εφαρμογή είναι Widgets.

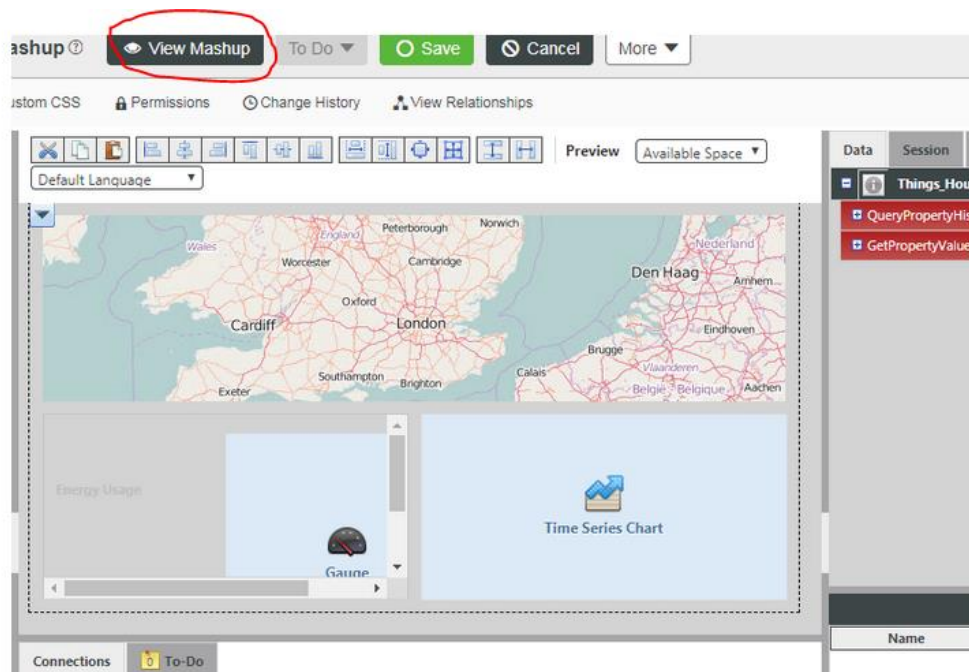


Εικόνα 6.48: Δημιουργία Dashboards

Για την συγκεκριμένη δοκιμή, δημιουργήθηκαν τρία widgets, μέσω των οποίων απεικονίζονται, ένας μετρητής (Gauge), ένα γράφημα χρονοσειρών (time series chart) και ένας χάρτης (Open Street Map) για την γεωγραφική απεικόνιση.

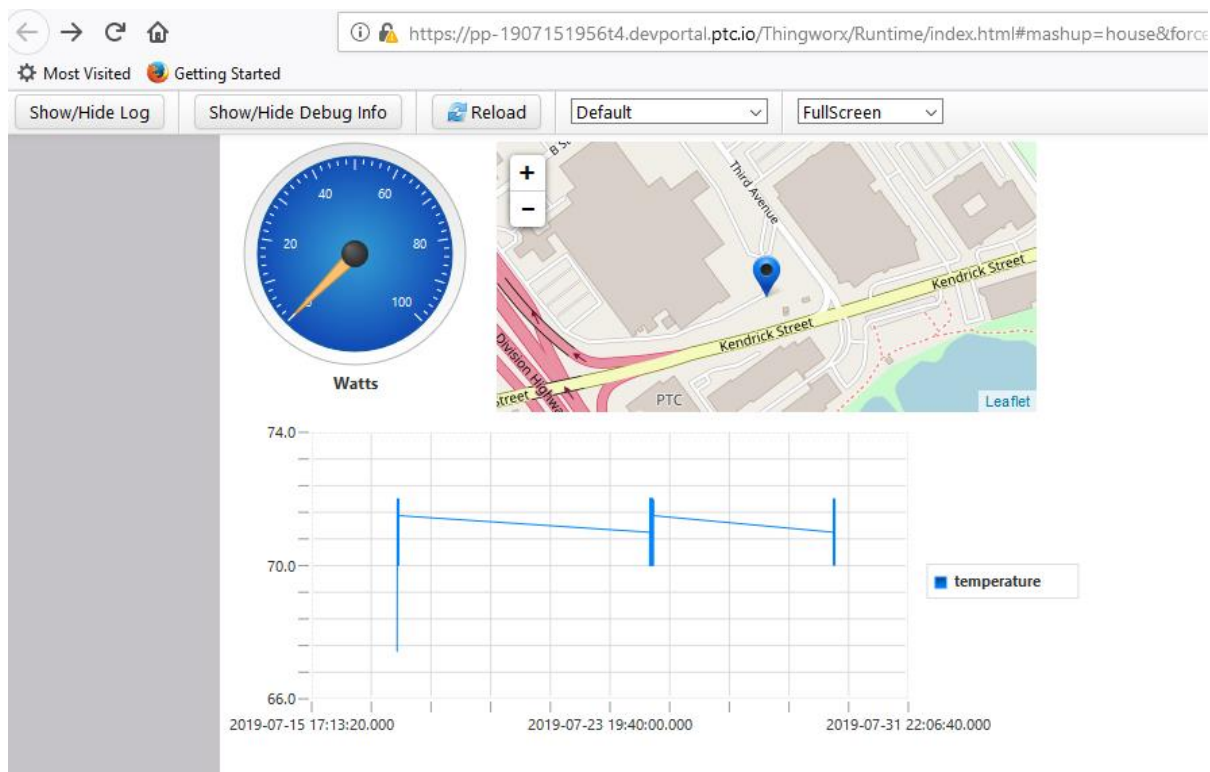
6.4.7 Έλεγχος Δοκιμής

Για τον έλεγχο και την έναρξη της εφαρμογής, επιλέγουμε το εικονίδιο Mashup (Εικόνα 6.49)



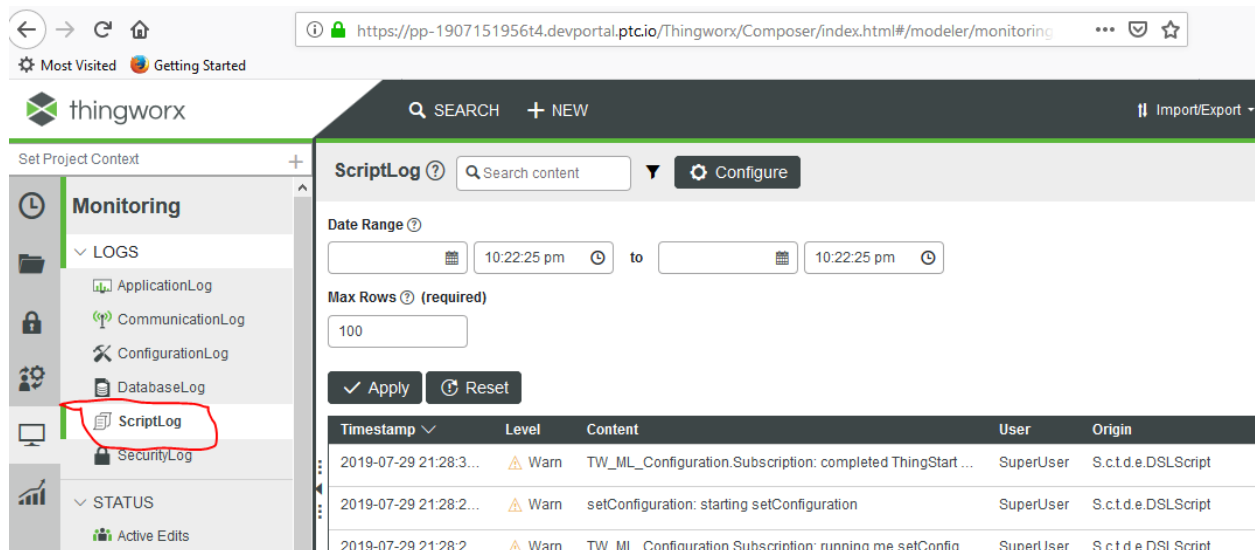
Εικόνα 6.49: Έναρξη Εφαρμογής

Τα αποτελέσματα απεικονίζονται παρακάτω (Εικόνα 6.50)



Εικόνα 6.50: Απεικόνιση Αποτελεσμάτων

Η ενημέρωση για τις αλλαγές που έλαβαν μέρος, απεικονίζονται μέσω του monitoring εργαλείου που διαθέτει η πλατφόρμα και συγκεκριμένα της επιλογής ScriptLog (Εικόνα 6.51).



Εικόνα 6.51: Απεικόνιση Αρχείου Καταγραφής

6.4.8 Συμπεράσματα

Το περιβάλλον ανάπτυξης και διαχείρισης της ThingWorx πλατφόρμας παρέχει ευελιξία, γρήγορη διασύνδεση μεταξύ των αντικειμένων και της πλατφόρμας καθώς και μια πληθώρα εργαλείων για την δημιουργία και την ανάπτυξη ολοκληρωμένων εφαρμογών IoT. Σημαντικό πλεονέκτημα αποτελούν οι προεγκατεστημένες επεκτάσεις και τα widgets που διατίθενται. Επίσης, η κεντρική διαχείριση όλων των λειτουργιών και των υπηρεσιών μέσω του γραφικού περιβάλλοντος που διαθέτει, την καθιστούν αρκετά προσιτή στον απλό χρήστη.

ΚΕΦΑΛΑΙΟ 7 - ΑΞΙΟΛΟΓΗΣΗ ΠΛΑΤΦΟΡΜΩΝ

7.1 Διαδικασία

Αφού ολοκληρώθηκε η καταγραφή των εργαλείων, των υπηρεσιών και γενικά των δυνατοτήτων που παρέχει η κάθε μια από τις IoT πλατφόρμες που παρουσιάστηκαν στα προηγούμενα κεφάλαια, επόμενο βήμα είναι η συγκριτική αξιολόγησή τους. Ο τρόπος βάσει του οποίου γίνεται η αξιολόγηση, στηρίζεται στην χρήση ενός πίνακα στον οποίο εμφανίζονται όλα τα τεχνικά χαρακτηριστικά των πλατφορμών. Πέραν βέβαια των παραπάνω, ο συγκεκριμένος πίνακας απεικονίζει και τις τάσεις που υπάρχουν στην ανάπτυξη των IoT πλατφορμών και των στοιχείων που τις απαρτίζουν όπως για παράδειγμα τα πρωτόκολλα επικοινωνίας, ο τρόπος αποθήκευσης των δεδομένων καθώς και η ενσωμάτωση υπηρεσιών τρίτων κατασκευαστών. Θα πρέπει ωστόσο να επισημανθεί ότι δεν γίνεται αναφορά στο ποια ενδεχομένως είναι η καλύτερη πλατφόρμα, διότι αυτό σχετίζεται με τις ανάγκες και τις απαιτήσεις των χρηστών αλλά κυρίως σε θέματα τα οποία μπορούν να θεωρηθούν ως πλεονέκτημα ή μειονέκτημα. Φυσικά, αυτές οι πλατφόρμες μοιράζονται ένα κοινό έδαφος, αλλά παρουσιάζουν επίσης σημαντικές διαφορές όσον αφορά την αρχιτεκτονική και τον τρόπο με τον οποίο μπορούν να υλοποιηθούν οι IoT υπηρεσίες. Έχοντας τα παραπάνω υπόψη έχουν δημιουργηθεί οι παρακάτω συγκριτικοί πίνακες αξιολόγησης των δεκατεσσάρων πλατφορμών, στους οποίους αποτυπώνονται οι ομοιότητες και διαφορές των χαρακτηριστικών, τα οποία απαρτίζουν τα κριτήρια αξιολόγησης. Θα πρέπει να τονισθεί ότι αν και τα κριτήρια αξιολόγησης που παρουσιάστηκαν στα προηγούμενα κεφάλαια είναι δεκαέξι στον αριθμό, στους πίνακες παρουσιάζονται ως δεκαοκτώ μιας και το κριτήριο σχετικά με τις υποστηριζόμενες γλώσσες προγραμματισμού έχει διαχωρισθεί σε δύο μέρη (server and client languages) καθώς επίσης γίνεται και αναφορά στο εάν η πλατφόρμα είναι open source ή όχι.

Platforms	Integration	Protocols	Open	Hosting	Security	Analytics
Kaa	REST API, NATS	MQTT, CoAP, HTTP	Yes	On-premise or Cloud	TLS, DTLS RSA key 2048 bits, AES key 256 bits	Real-Time Apache Storm, Hadoop
SiteWhere	REST API, Self-Reg	MQTT, AMQP, Stomp	Yes	On-premise, Cloud, Kubernetes	Link Encryption (SSL)	Apache Spark
ThingSpeak	ThingSpeak API, REST	HTTP, MQTT	Yes	Cloud, Standalone Server	SSLv2, SSLv3, and TLSv1, API Keys	MATLAB Production Server
ThingsBoard	REST API	HTTP, MQTT, CoAP	Yes	Cloud, Standalone Server and via Docker	Access Tokens, X.509 Certificates	Apache Spark
Thinger.io	REST API	HTTP, MQTT, CoAP	Yes	Cloud, Instable package Ubuntu	SSL/TLS support	Weka Data Mining
IBM Watson	REST API, HTTP API	Lightweight MQTT	No	Cloud	TLS/SSL, IPSEC, CAs	Watson Analytics (WA), Real-time analytics (IBM IoT Real-Time Insights)
AWS	HTTP/HTTPS API	MQTT, HTTP, WebSockets	No	Cloud	TLS cipher suites	AWS Kinesis, AWS Lambda
Oracle	REST API	MQTT	No	Cloud	TLS, trusted certificate, OATH2	Oracle Big Data Cloud
MS Azure	REST APIs	HTTP, AMQP, MQTT	No	Cloud	Security Tokens, X.509 certificates	HDInsight, Real-time analytics
Carriots	REST API	HTTP, MQTT	Yes	Cloud	SSL	Dynamic Dashboards
WSO2	REST API	HTTP, MQTT, XMPP	Yes	Cloud hosted	Single Sign-On (SSO), SAML2, and Kerberos KDC	WSO2 DAS
Ubidots	REST API	HTTP, MQTT	No	Cloud	SSL, TLS, Token-based Auth	Distimo, Google Analytics, Data Explorer app
ThingWorx	REST API	MQTT, COAP	No	Cloud, on premise, or a hybrid of the two.	TLS, AES 128, PKI, FIPS chipers	Predictive analytics (ThingWorx, ML Real-time (ParStream DB)
Google IoT	REST, CLOUDIOT API	HTTP, MQTT	No	Cloud	TLS, Per-device public/private key auth	BigQuery, Cloud Datalab

Πίνακας 7-1: Χαρακτηριστικά IoT πλατφορμών (μέρος 1/3)

Platforms	Visualization	Data storage	BLOB	Trigger	Push Notifications	Mobile SDK
Kaa	Visualization, Widgets	MongoDB, Cassandra, Hadoop, Oracle NoSQL	Yes, Raw and unstructured data	Yes, Push and Pull	Yes, via sms	Android, iOS
SiteWhere	Electron, Grafana	MongoDB, Influx DB, Apache Hbase	N/A	Yes, via third party application Mule Any Point	Yes, via API	Android, iOS
ThingSpeak	GUI, Dynamic dashboard via widgets	Azure SQL	Azure Blob	Yes, via ThingSpeak apps, IFTTT	Using web services like Twitter and Twilio	N/A
ThingsBoard	Dynamics dashboards and custom widgets	SQL (PostgreSQL) or NoSQL (Cassandra)	Yes	Yes, via ThingsBoard Rule Engine	Yes, via Rule Engine, alarms and email	Only from version 2.4
Thinger.io	Dynamic dashboard via widgets	Data Buckets	Yes, via Protoson Library	Yes for boolean behaviors (>,<=,!)	Yes, via API	Android
IBM Watson	Visualization charts, Gauges, bar charts	IBM Cloudant NoSQL Database	Yes	Basic real-time rules and actions (OR-AND). Webhooks, IFTTT	Yes, via APIs	Android, iOS
AWS	AWS IoT Dashboard, Amazon QuickSight, QlikView	Amazon Simple Storage Service (S3)	Yes	Yes, function for AWS Lambda	Amazon Simple Notification Service (SNS)	iOS through AWS IoT API
Oracle	Dynamic Dashboards	Oracle Database, NoSQL Database	Unstructured data in the form of audio, video and images	Yes, Rules engine	Yes, via email	Android, iOS
MS Azure	Azure Time Series Insights, Power BI, Cortana	Azure SQL, DocumentDB, Storage Blob	Azure Blob Storage	Yes, via Event Grid	Yes, via Event Grid	Android, iOS
Carriots	Via Azure services, Nobodha	NoSQL Big Data Base	N/A	Yes	Yes, via APIs	Android
WSo2	WSo2 SP Dashboard	WSo2 Data Services Server(DSS)MSSQL, DB2, Oracle, MySQL	Multi-tenant structured and unstructured data	Firestore Cloud Messaging or Apple Push Notification	Firestore Cloud Messaging or Apple Push Notification	Android
Ubidots	Dynamic and custom Dashboards	Microsoft Azure, SQL	N/A	Yes, custom triggers	Yes, via sms, email, Slack Notification	Android, iOS
ThingWorx	Widgets, Dashboards (ThingWorx SQUEAL)	PostgreSQL, embedded database	Yes	Yes, custom triggers	Telegram App using Rest API Call extension	Android, iOS
Google IoT	Qlik, iCharts, zoomdata, Looker, tableau	Cloud Bigtable, Cloud SQL, mongoDB	Unstructured data, any format	Yes, via Cloud Function	GCM (Google Cloud Messaging)	Android

Πίνακας 7-2: Χαρακτηριστικά IoT πλατφορμών (μέρος 2/3)

Platforms	External platforms	Hardware Supported	Server Lang.	Client Lang.	Geolocation	Pricing Plans
Kaa	Amazon web services, Microsoft Azure	Intel Edison, Raspberry Pi, LeafLabs, BeagleBone	Java	Python, Java, Scala, C++, Ruby	Yes, Google Maps	Trial period, Monthly, Yearly
SiteWhere	Mule AnyPoint Platform, OpenHAB, Microsoft Azure	Arduino, Raspberry Pi, Intel and others platforms	N/A	SDK Android	Yes, Google Maps	Free account or other paid plans
ThingSpeak	AWS, MATLAB	Arduino®, ESP-8266, Particle and Raspberry Pi	C, C++	MATLAB	Yes, Google Maps	Commercial, non-commercial, different licenses
ThingsBoard	AWS IoT, IBM Watson, Azure Event Hub	Raspberry Pi, Arduino and others	Java	C++	Yes, Google and Yepzon Asset	Community Edition and Professional Edition)
Thinger.io	Amazon web services, Github, IFTTT	Arduino, ESP8266, Raspberry Pi, Intel Edison	C++	C++	Yes, via Google maps	Free account or other paid plans
IBM Watson	Twitter, Node-RED, Jasper, Google BigQuery	ARM mbed, Texas Instruments, Raspberry Pi, Arduino Uno	Java, C++ and Prolog	Python, Java, Node.js C++, C Sharp	Owntracks, Google maps	Related to number of device, data traffic and data storage
AWS	N/A	SEGGER Microcontroller, Texas Instruments, Libelium and others	C++, Ruby	Python, Java, .NET, C++, Ruby, PHP	N/A	Connection time, number of messages, and rules engine actions
Oracle	Yes, via APIs	Raspberry, Arduino and other Microcontrollers	Java	Java, Javascript	Yes, Asset Monitoring Cloud	Pay as You Go Model
MS Azure	N/A	Intel, Raspberry Pi2, Free-scale, Texas Instruments	C, Python, Java, .NET	Net and UWP, Java, C, NodeJS	Yes, Azure Maps	Number of devices and messages per days
Carriots	Twitter, Dropbox, Sigfox	Arduino, Raspberry Pi, Cloudgate	Groovy - Java	N/A	Yes, via Google Maps	Free, or paid plans
WSo2	Business (SAP, FIX, HL7) Salesforce, Twitter, PayPal	Raspberry Pi, Arduino Uno, v3D Robotics Drone, etc.	Java, C	C++	Location-based services (LBS)	Free, or paid plans
Ubidots	Yes, via UBIParser	Android, Arduino, ARM mbed and Electric Imp	N/A	Python, C, and more	Yes, Google Geolocation API, Geofencing	Pay as You Grow
ThingWorx	Yes via API such as, Azure, AWS, Twitter, Twillio	Raspberry Pi, Intel Edison, Intel Galileo, etc.	N/A	SDKs through various programming languages (Java, C, .NET)	Yes, via Google Maps	Free, Professional, Enterprise
Google IoT	Only Google proprietary services	Beaglebone, Raspberry Pi, Arduino, Particle, etc.	C#, Go, Java, Node.js, PHP, Python, Ruby	Java, NODE.JS, Python	Yes, via Google Maps	Trial or paid plans, Pay-as-you-go

Πίνακας 7-3: Χαρακτηριστικά IoT πλατφορμών (μέρος 3/3)

7.2 Συζήτηση

Αρχικά παρατηρούμε ότι στο κομμάτι της **ενσωμάτωσης (integration) και των APIs**, τα οποία αποτελούν το σημείο αλληλεπίδρασης μεταξύ μιας IoT συσκευής και της πλατφόρμας καθώς και άλλων στοιχείων εντός του IoT δικτύου, επικρατεί σχεδόν σε όλες τις περιπτώσεις η επιλογή των REST APIs διότι παρέχουν έναν απλούστερο τρόπο διασύνδεσης των εφαρμογών και των "Αντικειμένων". Στην περίπτωση της Google IoT πλατφόρμας ονομάζεται "cloud iot API" διότι αποτελείται από δύο σύνολα REST πόρων (resources). Επίσης, υπάρχει και η επιλογή του MQTT API όπως στην περίπτωση της πλατφόρμας ThingSpeak για την περίπτωση όπου υπάρχει περιορισμένο εύρος ζώνης (bandwidth). Εν συνεχεία, στο κομμάτι **των πρωτοκόλλων που χρησιμοποιούνται για την ανταλλαγή των μηνυμάτων** από και προς τις συσκευές, όλες οι πλατφόρμες υποστηρίζουν τουλάχιστον δύο από τα τέσσερα επικρατέστερα πρωτόκολλα τα οποία είναι το MQTT, το CoAP, το AMQP και το HTTP. Ωστόσο το MQTT αποτελεί το μοναδικό πρωτόκολλο το οποίο υποστηρίζεται από όλες τις πλατφόρμες. Πέραν των τεσσάρων αυτών πρωτοκόλλων, η πλατφόρμα SiteWhere υποστηρίζει την χρήση ενός μη διαδεδωμένου πρωτοκόλλου όπως το Stomp, για την ανταλλαγή των μηνυμάτων.

Στο διαχωρισμό μεταξύ **open-source και non open-source**, περίπου οι μισές εξ αυτών όπως η Kaa, η SiteWhere, η WSo2, η ThingSpeak είναι open-source, επιτρέποντας στους προγραμματιστές να έχουν μεγαλύτερο βαθμό ελέγχου σε ολόκληρο το σύστημα, τον πηγαίο κώδικα, τους μηχανισμούς συνδεσιμότητας και ασφαλείας.

Εν συνεχεία, στο πεδίο το οποίο σχετίζεται με **την αποθήκευση των δεδομένων** παρατηρούμε μία πληθώρα επιλογών. Το Διαδίκτυο των αντικειμένων (IoT) παράγει τεράστιες ποσότητες δεδομένων, όπως δεδομένα ροής, δεδομένα χρονοσειρών, δεδομένα αισθητήρων κ.λπ. Η αποτελεσματική διαχείριση αυτών των δεδομένων απαιτεί τη χρήση μιας βάσης δεδομένων. Οι open-source πλατφόρμες όπως για παράδειγμα η Kaa και η Sitewhere χρησιμοποιούν για την αποθήκευση των δεδομένων open-source βάσεις δεδομένων, όπως η Apache Cassandra, η οποία είναι ένα ελεύθερο και ανοικτού κώδικα σύστημα διαχείρισης βάσεων δεδομένων NoSQL, η MongoDB και η InfluxDB η οποία είναι και αυτή μια βάση δεδομένων ανοιχτού κώδικα, η οποία προσφέρει δυνατότητες όπως γρήγορη αποθήκευση και ταχεία ανάκτηση δεδομένων χρονοσειρών (times series database). Από την άλλη μεριά, πλατφόρμες όπως η AWS, η Azure και η IBM Watson χρησιμοποιούν βάσεις δεδομένων τις οποίες έχουν αναπτύξει και τις υποστηρίζουν οι ίδιες. Κάποιες από αυτές είναι η Cloudant NoSQL Database της IBM, η Azure SQL η οποία είναι μια σχεσιακή βάση δεδομένων γενικής χρήσης και η επιλογή της Amazon με την υπηρεσία Amazon Simple Storage Service (S3). Επιπλέον, σχεδόν όλες οι πλατφόρμες υποστηρίζουν την **αποθήκευση δεδομένων "BLOB"**

τα οποία αποτελούν μη επεξεργασμένα δεδομένα (RAW or unstructured) όπως αρχεία εικόνων, ήχου ή άλλων πολυμέσων.

Τόσο για την αλληλεπίδραση των συσκευών με τις IoT πλατφόρμες καθώς και για την ανταλλαγή των δεδομένων μεταξύ αυτών θα πρέπει να εξασφαλιστεί η χρήση ενός **ασφαλούς καναλιού επικοινωνίας**. Βάσει του συγκριτικού πίνακα αξιολόγησης, παρατηρούμε ότι όλες οι πλατφόρμες υποστηρίζουν το πρωτόκολλο TLS (Transport Layer Security) μέσω του οποίου επιτυγχάνεται η κρυπτογράφηση των δεδομένων. Επίσης, πέραν του TLS κάποιες πλατφόρμες όπως η Kaa υποστηρίζουν και την χρήση του DTLS (Datagram Transport Layer Security) πρωτοκόλλου με στόχο την αποτροπή της υποκλοπής (eavesdropping), της παραποίησης (tampering) ή της πλαστογράφησης (forgery) των μηνυμάτων. Επιπλέον της κρυπτογράφησης, στις περισσότερες των περιπτώσεων υποστηρίζονται και άλλα επίπεδα ασφαλείας, όπως ο έλεγχος ταυτότητας των συσκευών με την χρήση X.509 πιστοποιητικών (certificates) στην περίπτωση της AWS IoT πλατφόρμας, την χρήση OAuth2 tokens για την επικοινωνία μεταξύ των APIs στην περίπτωση της Samsung Artik IoT πλατφόρμας καθώς και την χρήση APIs κλειδιών στην περίπτωση της IBM Watson IoT πλατφόρμας. Στην περίπτωση της SiteWhere πλατφόρμας παρατηρούμε την χρήση του ενονομαζόμενου Spring Security Framework, το οποίο είναι ένα ισχυρό και εξαιρετικά προσαρμόσιμο JAVA πλαίσιο ελέγχου ταυτότητας και ελέγχου πρόσβασης.

Τα δεδομένα που συλλέγονται από τους αισθητήρες θα πρέπει να αναλύονται και να απεικονίζονται με έξυπνο τρόπο, ώστε να μπορούν να εξαχθούν σημαντικές πληροφορίες μέσω αυτών. Οι πλατφόρμες διαθέτουν διάφορα εργαλεία – υπηρεσίες για **ανάλυση δεδομένων** σε πραγματικό χρόνο (real-time), αναλύσεις δεδομένων καταγραφής (logs), αναλύσεις δεδομένων που βασίζονται σε τεχνικές στατιστικής και μηχανικής μάθησης (machine learning) καθώς και στην ανάλυση δεδομένων "παρτίδας" (batch data) τα οποία αφορούν ένα συσσωρευμένο σύνολο δεδομένων, μία εργασία η οποία εκτελείται σε προγραμματισμένες χρονικές περιόδους και μπορεί να διαρκεί για αρκετές ώρες ή ημέρες.

Σχετικά με την **απεικόνιση των δεδομένων** η οποία παίζει καθοριστικό ρόλο και στην ανάλυση αυτών, οι πλατφόρμες διαθέτουν διάφορα εργαλεία μέσω των οποίων μπορούν να δημιουργηθούν τόσο στατικά αλλά και δυναμικά γραφήματα. Τα γραφήματα μπορούν να είναι από απλά γραμμικά έως και τρισδιάστατα όπως στην περίπτωση της ThingSpeak πλατφόρμας με την χρήση του Matlab. Σχετικά με τα εργαλεία απεικόνισης μπορεί να υπάρξει κάποια κατηγοριοποίηση ανάλογα με την χρήση τους, όπως για παράδειγμα στην περίπτωση της πλατφόρμας SiteWhere και του εργαλείου Grafana το οποίο χρησιμοποιείται για την απεικόνιση των δεδομένων που σχετίζονται με την κατάσταση (system health, CPU) των συσκευών και στην περίπτωση της Azure πλατφόρμας με το εργαλείο Power BI το οποίο μπορεί να αντλήσει δεδομένα από διάφορες πηγές όπως το Excel και το Google Analytics.

Αναφορικά με την **δυνατότητα ενεργοποίησης συμβάντων (Trigger Actions)** και την **αποστολή ειδοποιήσεων (Push Notifications)**, παρατηρούμε από τους συγκριτικούς πίνακες αξιολόγησης ότι πρόκειται για δύο λειτουργίες οι οποίες υποστηρίζονται από όλες τις πλατφόρμες μέσω ενσωματωμένων εργαλείων (Amazon Simple Notification Service) όσο και τρίτων κατασκευαστών (IFTTT, Webhooks).

Συνεχίζοντας την διαδικασία αξιολόγησης ένα ακόμη κριτήριο στο οποίο διαφοροποιούνται οι πλατφόρμες είναι αυτό της ανάπτυξης – εγκατάστασης. Για παράδειγμα κάποιες πλατφόρμες όπως η IBM Watson, η AWS, η Azure και άλλες μπορούν να αναπτυχθούν μόνο σε cloud υποδομή, σε αντίθεση με πλατφόρμες όπως η Kaa, η Thingspeak και η SiteWhere, παρέχουν την δυνατότητα ανάπτυξης των υπηρεσιών τους σε αυτόνομους διακομιστές (standalone servers) πέραν των cloud υποδομών. Επίσης, υπάρχει και η δυνατότητα ενός υβριδικού μοντέλου (cloud, on premise) στην περίπτωση της Thingworx πλατφόρμας.

Συνεχίζοντας την διαδικασία αξιολόγησης ένα ακόμη κριτήριο στο οποίο διαφοροποιούνται οι πλατφόρμες είναι αυτό της **ανάπτυξης – εγκατάστασης (Hosting)**. Για παράδειγμα κάποιες πλατφόρμες όπως η IBM Watson, η AWS, η Azure και άλλες μπορούν να αναπτυχθούν μόνο σε cloud υποδομή, σε αντίθεση με πλατφόρμες όπως η Kaa, η Thingspeak και η SiteWhere, παρέχουν την δυνατότητα ανάπτυξης των υπηρεσιών τους σε αυτόνομους διακομιστές (standalone servers) πέραν των cloud υποδομών. Επίσης, υπάρχει και η δυνατότητα ενός υβριδικού μοντέλου (cloud, on premise) στην περίπτωση της Thingworx πλατφόρμας.

Ο μεγαλύτερος όγκος των πλατφορμών υποστηρίζει την **δυνατότητα παρακολούθησης και εντοπισμού των IoT συσκευών** μέσω της χρήσης GPS, ωστόσο παρατηρούμε ότι σχεδόν όλες χρησιμοποιούν API Χαρτών Google (Geolocation API) για να βελτιώσουν την ακρίβεια της τοποθεσίας σε υπάρχουσες συσκευές GPS και να διασφαλίσουν ότι τα δεδομένα θέσης είναι διαθέσιμα ακόμα και αν το GPS αποτύχει.

Αναφορικά με τα **Κιτ Ανάπτυξης Λογισμικού (SDKs)**, όλες οι πλατφόρμες παρέχουν την συγκεκριμένη δυνατότητα ώστε να μπορούν οι προγραμματιστές να δημιουργούν εφαρμογές για κινητές συσκευές (mobile apps). Για παράδειγμα η πλατφόρμα AWS υποστηρίζει Android και iOS SDKs.

Από προγραμματιστικής πλευράς, οι γλώσσες που υποστηρίζονται από τις Internet of Things πλατφόρμες τόσο σχετικά με την **εγγραφή προσαρμοσμένου κώδικα από τη πλευρά του διακομιστή (server)** όσο και για την **εγγραφή κώδικα από την πλευρά του πελάτη (client)** υπάρχει μεγάλη ποικιλία η οποία αποτελείται από γλώσσες όπως: η Java, η C++, η JavaScript, η Python, η Go και πολλές άλλες

Επίσης, στο κομμάτι της **διασύνδεσης με υπηρεσίες τρίτων παρόχων (third-party-services)**, παρατηρούμε ότι οι περισσότερες των πλατφορμών μπορούν και αλληλοεπιδρούν με τρίτα συστήματά με την χρήση κυρίως REST APIs, όπως για παράδειγμα η Kaa με τις web υπηρεσίες της Amazon και η

SiteWhere με την εφαρμογή Grafana για την απεικόνιση δεδομένων. Ωστόσο, πλατφόρμες όπως η AWS IoT και η Google χρησιμοποιούν εξ ολοκλήρου υπηρεσίες δικής του ανάπτυξης.

Πέραν της διασύνδεσης με υπηρεσίες τρίτων εταιρειών, στο στοιχείο της επικοινωνίας με **πλατφόρμες υλικού όπως είναι το** Arduino και το Raspberry Pi, κυρίως δηλαδή μικροελεγκτές, οι περισσότερες εξ αυτών διαθέτουν αυτήν την δυνατότητα, όπως η πλατφόρμα Thingier.io η οποία χαρακτηρίζεται ως "hardware agnostic" δηλαδή δεν απαιτείται κάποια εξάρτηση από συγκεκριμένο υλικό για την διασύνδεση του με του μικροελεγκτή με την πλατφόρμα

Τέλος, αναφορικά με το κομμάτι **της τιμολόγησης των υπηρεσιών** που προσφέρουν οι IoT πλατφόρμες, διαπιστώνουμε μέσα από τον συγκριτικό πίνακα αξιολόγησης ότι από όλες παρέχεται μια περίοδος χρήσης και δοκιμής αυτών με μηδαμινό κόστος. Αναφορικά με την χρέωση των υπηρεσιών η IBM και η Google διαθέτουν ένα απλό μοντέλο τιμολόγησης το οποίο βασίζεται στον όγκο των μηνυμάτων και των δεδομένων που ανταλλάσσονται. Η IBM προσθέτει επιπλέον κόστος για υπηρεσίες οι οποίες σχετίζονται με την ανάλυση των δεδομένων. Η Microsoft διαθέτει και αυτή ένα πιο απλό μοντέλο βάσει του αριθμού των μηνυμάτων που αποστέλλονται / λαμβάνονται από το IoT Hub. Η AWS πλατφόρμα έχει το πιο περίπλοκο μοντέλο τιμών το οποίο βασίζεται στον χρόνο σύνδεσης, στον αριθμό των μηνυμάτων και στα αιτήματα (requests) τα οποία αποστέλλονται από τις συσκευές.

Η τεχνολογία των IoT πλατφορμών έρχεται αντιμέτωπη με μία σειρά προκλήσεων οι οποίες θα πρέπει να αντιμετωπιστούν. Παραδείγματα κάποιων εκ των βασικών προκλήσεων αναλύονται παρακάτω και περιλαμβάνουν τα εξής:

Διαθεσιμότητα

Η διαθεσιμότητα θα πρέπει να πραγματοποιείται τόσο σε επίπεδο υλικού όσο και σε επίπεδο λογισμικού ώστε να παρέχονται οπουδήποτε και οποτεδήποτε οι IoT υπηρεσίες για τους διάφορους χρήστες. Η διαθεσιμότητα του λογισμικού αναφέρεται στην ικανότητα των IoT εφαρμογών να παρέχουν υπηρεσίες για όλους σε διαφορετικά μέρη ταυτόχρονα. Η διαθεσιμότητα υλικού αναφέρεται στην ύπαρξη συσκευών που είναι συμβατές με τις λειτουργίες και τα πρωτόκολλα του Διαδικτύου.

Αξιοπιστία

Η αξιοπιστία αφορά την ορθή λειτουργία του συστήματος βάσει των προδιαγραφών του. Η αξιοπιστία στοχεύει στην αύξηση του ποσοστού επιτυχίας της παροχής των IoT υπηρεσιών. Πρόκειται για την διαθεσιμότητα των πληροφοριών και των υπηρεσιών με την πάροδο του χρόνου. Το κρίσιμο μέρος είναι το δίκτυο επικοινωνίας το οποίο πρέπει να είναι ανθεκτικό στις αστοχίες προκειμένου να πραγματοποιηθεί η αξιόπιστη διανομή των πληροφοριών.

Απόδοση

Η αξιολόγηση της απόδοσης των IoT υπηρεσιών είναι μια μεγάλη πρόκληση, καθώς εξαρτάται από την απόδοση πολλών συνιστωσών καθώς και από τις επιδόσεις των βασικών τεχνολογιών. Το IoT, όπως και άλλα συστήματα, πρέπει να αναπτύσσει και να βελτιώνει συνεχώς τις υπηρεσίες ώστε να ανταποκρίνεται στις απαιτήσεις των χρηστών. Οι IoT συσκευές πρέπει να παρακολουθούνται και να αξιολογούνται για να παρέχουν την καλύτερη δυνατή απόδοση.

Διαχείριση

Η μελλοντική διασύνδεση δισεκατομμυρίων ή τρισεκατομμυρίων έξυπνων συσκευών φέρνει αντιμέτωπους τους παρόχους υπηρεσιών με ζητήματα που σχετίζονται με τη διαχείριση αυτών των συσκευών και αφορούν ζητήματα όπως, η παραμετροποίηση, η απόδοση και η ασφάλεια. Η διαχείριση των IoT συσκευών μπορεί να αποτελέσει έναν αποτελεσματικό παράγοντα για την ανάπτυξη των IoT εφαρμογών.

Επεκτασιμότητα

Η κλιμάκωση του IoT αναφέρεται στην ικανότητα προσθήκης νέων συσκευών, υπηρεσιών και λειτουργιών για τους χρήστες χωρίς αυτό να έχει ως αποτέλεσμα να επηρεαστεί αρνητικά η ποιότητα των υφιστάμενων υπηρεσιών. Η προσθήκη νέων λειτουργιών και η υποστήριξη νέων συσκευών δεν είναι εύκολη υπόθεση, ειδικά με την παρουσία διαφορετικών πλατφορμών υλικού και πρωτοκόλλων επικοινωνίας.

Διαλειτουργικότητα

Η διαλειτουργικότητα αποτελεί μια ακόμη πρόκληση για τα IoT συστήματα λόγω της ανάγκης να διαχειριστούν έναν μεγάλο αριθμό ετερογενών καταστάσεων οι οποίες ανήκουν σε διαφορετικές πλατφόρμες. Η διαλειτουργικότητα είναι ένας παράγοντας ο οποίος θα πρέπει να εξεταστεί τόσο από τους προγραμματιστές εφαρμογών όσο και από τους κατασκευαστές IoT συσκευών ώστε να διασφαλιστεί η παροχή υπηρεσιών για όλους τους χρήστες ανεξάρτητα από τις προδιαγραφές της πλατφόρμας υλικού που χρησιμοποιούν. Για παράδειγμα, τα περισσότερα από τα smartphones σήμερα υποστηρίζουν κοινές τεχνολογίες επικοινωνιών όπως το WiFi, το NFC και το 4G ώστε να μπορέσουν να εγγραφούν τη διαλειτουργικότητα σε διάφορα σενάρια.

Ασφάλεια – Εμπιστευτικότητα

Η ασφάλεια παρουσιάζεται ως ακόμη μια σημαντική πρόκληση για τις IoT υλοποιήσεις λόγω της έλλειψης κοινού προτύπου και αρχιτεκτονικής. Σε ετερογενή δίκτυα, όπως στην περίπτωση του IoT, δεν είναι εύκολο να διασφαλιστούν καταστάσεις, όπως η ασφάλεια και το απόρρητο των χρηστών. Από τη στιγμή που η βασική λειτουργικότητα του IoT βασίζεται στην ανταλλαγή πληροφοριών μεταξύ δισεκατομμυρίων ή και

τρισεκατομμυρίων αντικειμένων μέσω του Διαδικτύου, η διασφάλιση της προστασίας των δεδομένων, αποτελεί ίσως τον σημαντικότερο παράγοντα.

ΚΕΦΑΛΑΙΟ 8 – ΣΥΜΠΕΡΑΣΜΑΤΑ

Στόχος της διπλωματικής εργασίας ήταν να παράσχει μια αναφορά ικανή να περιγράψει τον ρόλο που διαδραματίζουν οι IoT πλατφόρμες μέσα στο περιβάλλον του Διαδικτύου των Αντικειμένων καθώς να παρουσιάσει και να αναλύσει κάποια από τα σημαντικότερα χαρακτηριστικά και τις λειτουργίες αυτών. Προκειμένου να γίνει αυτό πραγματοποιήθηκε η μελέτη και η σύγκριση δεκατεσσάρων πλατφορμών, οι οποίες βασίστηκαν σε κριτήρια αξιολόγησης που ορίστηκαν. Για την καλύτερη κατανόηση της χρήσης κάποιων από των χαρακτηριστικών – δυνατοτήτων των πλατφορμών, πραγματοποιήθηκε η δοκιμή τριών εξ αυτών. Τόσο κατά την παρουσίαση όσο και μέσα από τις δοκιμές των πλατφορμών, εξάχθηκαν διάφορα συμπεράσματα σχετικά με τις συνιστώσες που πλαισιώνουν τις πλατφόρμες.

Ένα από τα βασικά συμπεράσματα που εξάχθηκε αφορά την έλλειψη τυποποίησης με αποτέλεσμα να έχει δημιουργηθεί ένα ετερογενές τοπίο σχετικά με την ανάπτυξη και τις δυνατότητες που παρέχουν οι πλατφόρμες. Ως αποτέλεσμα αυτής της ανομοιογένειας, η σύγκριση και η επιλογή μιας εκ των πλατφορμών αποτελούν μια δύσκολη και χρονοβόρα διαδικασία. Όχι μόνο χρησιμοποιούνται διαφορετικές έννοιες και τεχνολογίες, αλλά και η ορολογία δεν είναι σαφώς καθορισμένη [1].

Ένα ακόμη από τα συμπεράσματα που εξήχθησαν αφορά στο ότι πολλές από διαθέσιμες IoT πλατφόρμες είναι ανοικτού κώδικα (open source) γεγονός που σημαίνει ότι θα είναι ευκολότερο για τους προγραμματιστές να μπορούν να δοκιμάσουν νέες ιδέες εφαρμογής χωρίς να χρειάζεται να πληρώσουν για τις διάφορες υπηρεσίες. Αυτό οδηγεί στην ελεύθερη ροή ιδεών και συνεπώς σε αύξηση της καινοτομίας στο Διαδίκτυο των Αντικειμένων. Επίσης, έχουν συσταθεί πολλές κοινοπραξίες και συμβούλια τα τελευταία χρόνια για τη δομή του πλαισίου σχετικά με τις IoT αρχιτεκτονικές, καθώς επίσης και για τη διαμόρφωση προτύπων που μπορούν να χρησιμοποιηθούν σε εφαρμογές και πλατφόρμες του Διαδικτύου των Αντικειμένων, επιτυγχάνοντας έτσι τη μέγιστη διαλειτουργικότητα και υψηλότερο επίπεδο ασφάλειας για το υλικό και το λογισμικό.

Επιπλέον, τα χαρακτηριστικά ασφαλείας που αντιπροσωπεύουν αυτές τις πλατφόρμες αποτελούν και θα συνεχίσουν να αποτελούν μια κρίσιμη παράμετρο για την επιλογή της καταλληλότερης λύσης [27]. Οι άνθρωποι εξακολουθούν να μην αισθάνονται άνετα χρησιμοποιώντας εφαρμογές που βασίζονται στο Διαδίκτυο των Αντικειμένων από το φόβο ότι τα ευαίσθητα και προσωπικά δεδομένα που συλλέγονται, παρακολουθούνται, χωρίς να γνωρίζουν ότι η συλλογή δεδομένων αποτελεί σημαντικό μέρος μιας IoT εφαρμογής για να λειτουργήσει.

Ένα ακόμη σημαντικό ζήτημα το οποίο αποτελεί πλεονέκτημα για τους διαφόρους χρήστες, είναι η δυνατότητα παροχής δοκιμαστικών εκδόσεων που προσφέρονται από τις διάφορες εταιρείες ανάπτυξης

πλατφορμών, με στόχο την χρήση των υπηρεσιών και των λειτουργιών που προσφέρουν οι πλατφόρμες. Επίσης, η ύπαρξη ενημερωμένων και αναλυτικών οδηγιών χρήσης των διαφόρων δυνατοτήτων που παρέχονται από τις επίσημες ιστοσελίδες των πλατφορμών αποτελεί σημαντικό προνόμιο σε διάφορες καταστάσεις. Χωρίς την ύπαρξη των παραπάνω θα ήταν αδύνατον να πραγματοποιηθεί η δοκιμή των πλατφορμών.

Όπως παρατηρήσαμε όλες οι πλατφόρμες πλαισιώνονται από ένα μεγάλο πλήθος διαθέσιμων εργαλείων. Φυσικά αυτό δεν σημαίνει πως δεν υπάρχουν περιθώρια βελτίωσης των υφιστάμενων εργαλείων. Συνεπώς δεν πρέπει να θεωρείται ότι έχουν καλυφθεί όλες οι ανάγκες επειδή έχει ενσωματωθεί ένα εργαλείο, αφού με την περαιτέρω ανάπτυξη και προσθήκη νέων υπηρεσιών μπορούν οι χρήστες να έχουν επιπλέον διευκολύνσεις. Γι' αυτό πρέπει να γίνεται συνεχής αναζήτηση για βελτίωση του συνόλου των λειτουργιών και των χαρακτηριστικών των πλατφορμών, βασισμένη στην ανατροφοδότηση που λαμβάνεται από του εν δυνάμει χρήστες.

Θα πρέπει να επισημανθεί, ότι μια από τις μεγαλύτερες προκλήσεις που αντιμετωπίζει το Διαδίκτυο των Αντικειμένων, και όπως φαίνεται θα αποτελέσει στοιχείο μελλοντικών ερευνών, αφορά το θέμα της διαλειτουργικότητας (Interoperability) τόσο σε επίπεδο υλικού όσο και λογισμικού τα οποία εμπλέκονται στις IoT πλατφόρμες [51]. Σε γενικές γραμμές, η διαλειτουργικότητα μπορεί να οριστεί ως μέτρο του βαθμού στο οποίο διαφορετικά συστήματα, οργανισμοί και / ή άτομα μπορούν να συνεργαστούν για την επίτευξη ενός κοινού στόχου. Ωστόσο, η διαλειτουργικότητα είναι πολύπλοκη και υπάρχουν πολλές πτυχές. Τα "αντικείμενα" εξακολουθούν να μην είναι εντελώς διαλειτουργικά και ενδέχεται να περιορίζονται από διάφορους παράγοντες, όπως τα διάφορα πρωτόκολλα επικοινωνίας. Αυτό έχει ως συνέπεια, όλο και περισσότερες εταιρείες να επιλέγουν να δημιουργούν τη δική τους IoT πλατφόρμα από την αρχή (from the scratch), έτσι ώστε να μην χρειάζεται να ανησυχούν για τη διαλειτουργικότητα των συσκευών τους που περιορίζονται από μια προ-προγραμματισμένη IoT πλατφόρμα [50]. Ως εκ τούτου, το προφανές επόμενο βήμα για να παραμείνουν στο παιχνίδι οι εκάστοτε IoT πλατφόρμες, θα είναι να παρέχουν απόλυτη διαλειτουργικότητα χρησιμοποιώντας πρότυπα που έχουν προταθεί από το συμβούλιο και την κοινοπραξία του Διαδικτύου των Αντικειμένων (IoT Council and Consortium) και επίσης να επιτρέπουν στις επιχειρήσεις να έχουν την πλήρη ελευθερία να επιλέξουν τους δικούς τους διακομιστές σχετικά με την αποθήκευση και την επεξεργασία των δεδομένων [51].

Κλείνοντας, είναι πλέον σίγουρο ότι οι IoT πλατφόρμες ως μέρος του Διαδικτύου των Αντικειμένων, θα εξακολουθούν να αποτελούν ένα βασικό ερευνητικό πεδίο στον τομέα των υπολογιστών και της πληροφορικής εξ αιτίας του ετερογενούς μείγματος των διαφόρων τεχνολογιών από τις οποίες πλαισιώνονται.

ΑΝΑΦΟΡΕΣ

- [1] Guth, J., Breitenbücher, U., Falkenthal, M., Fremantle, P., Kopp, O., Leymann, F., & Reinfurt, L. (2018). A detailed analysis of IoT platform architectures: concepts, similarities, and differences. In *Internet of Everything* (pp. 81-101). Springer, Singapore.
- [2] Ray, P. P. (2016). A survey of IoT cloud platforms. *Future Computing and Informatics Journal*, 1(1-2), 35-46.
- [3] Mineraud, J., Mazhelis, O., Su, X., & Tarkoma, S. (2016). A gap analysis of Internet-of-Things platforms. *Computer Communications*, 89, 5-16.
- [4] Abdmeziem, M. R., Tandjaoui, D., & Romdhani, I. (2016). Architecting the internet of things: state of the art. In *Robots and Sensor Clouds* (pp. 55-75). Springer, Cham.
- [5] Mineraud, J., Mazhelis, O., Su, X., & Tarkoma, S. (2015). Contemporary internet of things platforms. *arXiv preprint arXiv:1501.07438*.
- [6] Botta, A., De Donato, W., Persico, V., & Pescapé, A. (2016). Integration of cloud computing and internet of things: a survey. *Future generation computer systems*, 56, 684-700.
- [7] Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE communications surveys & tutorials*, 17(4), 2347-2376.
- [8] Ngu, A. H., Gutierrez, M., Metsis, V., Nepal, S., & Sheng, Q. Z. (2016). IoT middleware: A survey on issues and enabling technologies. *IEEE Internet of Things Journal*, 4(1), 1-20.
- [9] Zdravković, M., Trajanović, M., Sarraipa, J., Jardim-Gonçalves, R., Lezoche, M., Aubry, A., & Panetto, H. (2016, February). Survey of Internet-of-Things platforms. In *6th International Conference on Information Society and Technology, ICIST 2016* (Vol. 1, pp. 216-220).
- [10] Toran Mercadé, D. (2018). *Comparison of different Internet of Things platforms* (Bachelor's thesis, Universitat Politècnica de Catalunya).
- [11] Mazhelis, O., & Tyrväinen, P. (2014, March). A framework for evaluating Internet-of-Things platforms: Application provider viewpoint. In *WF-IoT* (pp. 147-152).
- [12] Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE communications surveys & tutorials*, 17(4), 2347-2376.
- [13] Sruthi Mohan, A survey on IoT platforms. VIT University (2016)
- [14] Nakhuva, B., & Champaneria, T. (2015). Study of various internet of things platforms. *International Journal of Computer Science & Engineering Survey*, 6(6), 61-74.
- [15] Dumitru, R. L. (2017). IoT Platforms: Analysis for Building Projects. *Informatica Economica*, 21(2).
- [16] Preeti Agarwal, Mansaf Alam, Jamia Millia Islamia, New Delhi, India Investigating IoT Middleware Platforms for Smart Application Development

- [17] Goumopoulos, C., Kameas, A., Hagraas, H., Callaghan, V., Gardner, M., Minker, W., ... & Meliones, A. (2008, October). Atraco: Adaptive and trusted ambient ecologies. In *2008 Second IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshops* (pp. 96-101). IEEE.
- [18] Cabra, J., Castro, D., Colorado, J., Mendez, D., & Trujillo, L. (2017, June). An IoT approach for wireless sensor networks applied to e-health environmental monitoring. In *2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)* (pp. 578-583). IEEE.
- [19] Branislav Smik, (2016) The web-based platforms for IoT (Internet of Things) management, Masaryk University Faculty of Informatics (Bachelor's thesis).
- [20] Fernandes, E., Rahmati, A., Jung, J., & Prakash, A. (2018, February). Decentralized action integrity for trigger-action iot platforms. In *22nd Network and Distributed Security Symposium (NDSS 2018)*.
- [21] Maureira, M. A. G., Oldenhof, D., & Teernstra, L. (2011). ThingSpeak—an API and Web Service for the Internet of Things. *World Wide Web*.
- [22] Apoorv Gupta, Review of IoT Market and Open Source Technologies in IoT, *International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)* Vol 5, Issue 2, (2018)
- [23] Naik, N. (2017, October). Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP. In *2017 IEEE international systems engineering symposium (ISSE)* (pp. 1-7). IEEE.
- [24] Christos S. Tranoris, Open Source Software solutions implementing a reference IoT architecture from the Things and Edge to the Cloud. Technical Report (2018)
- [25] Luis Bustamante, A., Patricio, M. A., & Molina, J. M. (2019). Thinger. io: An Open Source Platform for Deploying Data Fusion Applications in IoT Environments. *Sensors*, 19(5), 1044.
- [26] Nikolaos Gkatzios, Big Data Analytics for the Cloud Master Thesis (2017)
- [27] Díaz López, D., Blanco Uribe, M., Santiago Cely, C., Tarquino Murgueitio, D., Garcia Garcia, E., Nespoli, P., & Gómez Mármol, F. (2018). Developing Secure IoT Services: A Security-Oriented Review of IoT Platforms. *Symmetry*, 10(12), 669.
- [28] [Theodoor van der Klaauw, IoT Platforms for Cities: A Comparative Survey. The Academy for smarter Communities (2019)
- [29] Henrique Hilleshein, IoT Platform Analysis for Educational Purposes IoT-Ticket. Häme University of Applied Sciences –HAMK (2018)
- [30] Eduardo Natan Bitencourt, Willian Pereira IoT Centralization and Management Applying ThingsBoard Platform. Häme University of Applied Sciences –HAMK (2018)
- [31] Hejazi, H., Rajab, H., Cinkler, T., & Lengyel, L. (2018, January). Survey of platforms for massive IoT. In *2018 IEEE International Conference on Future IoT Technologies (Future IoT)* (pp. 1-8). IEEE.

- [32] White paper, IOT PLATFORMS The central backbone for the Internet of Things. IOT ANALYTICS (2015)
- [33] Ganguly, P. (2016, January). Selecting the right IoT cloud platform. In *2016 International Conference on Internet of Things and Applications (IOTA)* (pp. 316-320). IEEE.
- [33] Fremantle, P. (2015). A reference architecture for the internet of things. *WSO2 White paper*.
- [34] <https://docs.aws.amazon.com/whitepapers/latest/aws-overview/internet-of-things-services.html>
- [35] <https://docs.kaaproject.org>
- [36] <https://cloud.ibm.com/docs>
- [37] <https://wso2.com/products>
- [38] <https://www.oracle.com/internet-of-things/>
- [39] <https://azure.microsoft.com/en-us/services/iot-hub/>
- [40] Sajith Ravindra. Everything You Need to Know About the WSO2 Analytics Platform *White paper*.
- [41] <https://www.mathworks.com/help/thingspeak/>
- [42] <https://cloud.google.com/solutions/iot/>
- [43] Alvaro Everlet. (2015). CARRIOTS TECHNICAL PRESENTATION White paper.
- [44] <https://sitewhere.io/docs/2.0.0/>
- [45] <https://ubidots.com/>
- [46] <https://www.ptc.com/en/products/iiot>
- [47] <https://thingsboard.io/>
- [48] <https://www.oracle.com/internet-of-things/>
- [49] <https://www.thinger.io/>
- [50] Mukherjee, S. (2016). Ranking System for IoT Industry Platform.
- [51] Soursos, S., Žarko, I. P., Zwickl, P., Gojmerac, I., Bianchi, G., & Carrozzo, G. (2016, June). Towards the cross-domain interoperability of IoT platforms. In *2016 European conference on networks and communications (EuCNC)* (pp. 398-402). IEEE.