

**ΑΕΙ ΠΕΙΡΑΙΑ Τ.Τ.
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ Τ.Ε.**

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**Μελέτη του Διαδικτύου των Πραγμάτων:
Αρχιτεκτονική, Εφαρμογές και Μελλοντικό όραμα**

**Αριστείδης Σ. Βυζανιάρης
Ευάγγελος Γ. Φαραντάτος**

Εισηγητές: Δημήτριος Καλλέργης

Ζαχαρένια Γαροφαλάκη

**ΑΘΗΝΑ
ΝΟΕΜΒΡΙΟΣ 2016**

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**Μελέτη του Διαδικτύου των Πραγμάτων:
Αρχιτεκτονική, Εφαρμογές και Μελλοντικό όραμα**

**Αριστείδης Σ. Βυζανιάρης
Α.Μ. 35871**

**Ευάγγελος Γ. Φαραντάτος
Α.Μ. 35684**

Εισηγητές:

Δημήτριος Καλλέργης

Ζαχαρένια Γαροφαλάκη

Εξεταστική Επιτροπή:

Ημερομηνία εξέτασης _____

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΩΝ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Οι κάτωθι υπογεγραμμένοι **ΑΡΙΣΤΕΙΔΗΣ ΒΥΖΑΝΙΑΡΗΣ**, του **ΣΤΕΦΑΝΟΥ**, με αριθμό μητρώου **35871** και **ΕΥΑΓΓΕΛΟΣ ΦΑΡΑΝΤΑΤΟΣ**, του **ΓΕΡΑΣΙΜΟΥ**, με αριθμό μητρώου **35684**, φοιτητές του Τμήματος Μηχανικών Η/Υ Συστημάτων Τ.Ε. του Α.Ε.Ι. Πειραιά Τ.Τ. πριν αναλάβουμε την εκπόνηση της Πτυχιακής Εργασίας μας, δηλώνουμε ότι ενημερωθήκαμε για τα παρακάτω:

«Η Πτυχιακή Εργασία (Π.Ε.) αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο του συγγραφέα, όσο και του Ιδρύματος και θα πρέπει να έχει μοναδικό χαρακτήρα και πρωτότυπο περιεχόμενο.

Απαγορεύεται αυστηρά οποιοδήποτε κομμάτι κειμένου της να εμφανίζεται αυτούσιο ή μεταφρασμένο από κάποια άλλη δημοσιευμένη πηγή. Κάθε τέτοια πράξη αποτελεί προϊόν λογοκλοπής και εγείρει θέμα Ηθικής Τάξης για τα πνευματικά δικαιώματα του άλλου συγγραφέα. Αποκλειστικός υπεύθυνος είναι ο συγγραφέας της Π.Ε., ο οποίος φέρει και την ευθύνη των συνεπειών, ποινικών και άλλων, αυτής της πράξης.

Πέραν των όποιων ποινικών ευθυνών του συγγραφέα σε περίπτωση που το Ίδρυμα του έχει απονείμει Πτυχίο, αυτό ανακαλείται με απόφαση της Συνέλευσης του Τμήματος. Η Συνέλευση του Τμήματος με νέα απόφασης της, μετά από αίτηση του ενδιαφερόμενου, του αναθέτει εκ νέου την εκπόνηση της Π.Ε. με άλλο θέμα και διαφορετικό επιβλέποντα καθηγητή. Η εκπόνηση της εν λόγω Π.Ε. πρέπει να ολοκληρωθεί εντός τουλάχιστον ενός ημερολογιακού 6μήνου από την ημερομηνία ανάθεσης της. Κατά τα λοιπά εφαρμόζονται τα προβλεπόμενα στο άρθρο 18, παρ. 5 του ισχύοντος Εσωτερικού Κανονισμού.»

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα πτυχιακή εργασία είναι αφιερωμένη στους εισηγητές μας, Δ. Καλλέργη και Ζ. Γαροφαλάκη, χωρίς τη διαρκή υποστήριξη, καθοδήγηση και υπομονή των οποίων δε θα μπορούσε να πραγματοποιηθεί.

Ευχαριστούμε θερμά τις οικογένειές μας, τους συμφοιτητές και τους φίλους μας που μοιραστήκαμε όλα αυτά τα χρόνια μαζί τους και μας στήριζαν σε όλη μας την πορεία.

ΠΕΡΙΛΗΨΗ

Στις μέρες μας ένας από τους κύριους στόχους του Διαδικτύου είναι η ανάπτυξή του ώστε να παρέχει τη δυνατότητα απόλυτης επικοινωνίας, όχι μόνο μεταξύ των ανθρώπων αλλά και μεταξύ των ανθρώπων με συσκευές και μεταξύ των συσκευών μεταξύ τους, οδηγώντας σε μια machine-to-machine επικοινωνία (M2M), με ή χωρίς την συμβολή της ανθρώπινης παρέμβασης. Εισερχόμαστε έτσι σε μια νέα εποχή πανταχού παρουσίας της επικοινωνίας αυτής, εποχή του Διαδικτύου των Πραγμάτων, Internet of Things, εννοώντας το Διαδίκτυο των Πληροφοριών των Πραγμάτων. Κάθε αντικείμενο θα έχει ένα μοναδικό αναγνωριστικό και θα μπορεί να διευθυνσιοδοτηθεί και να συνδεθεί στο Διαδίκτυο. Έτσι το Internet of Things (IoT) θα σχετίζεται με τις πληροφορίες που μπορεί να συλλέξει ένα αντικείμενο - με τη χρήση αισθητήρων που διαθέτει - και το διαμοιρασμό τους σε μια παγκόσμια κλίμακα, μέσω του Διαδικτύου. Η μεταφορά των πληροφοριών μπορεί να πραγματοποιηθεί με πολλούς τρόπους όπως με τη χρήση ραδιοκυμάτων. Έτσι, τεχνικές ραδιοσυχνοτικής αναγνώρισης (Radio Frequency Identification ή αλλιώς RFID) θα αποτελούν τον βασικό κορμό του IoT. Σημαντική θα είναι και η συμβολή μιας cloud πλατφόρμας η οποία θα είναι υπεύθυνη για την αποθήκευση, την διαχείριση και την επεξεργασία της τεράστιας ποσότητας δεδομένων, ώστε να μπορούν να προβληθούν με έναν κατανοητό τρόπο προς τους τελικούς χρήστες, για την καλύτερη παράσταση και κατανόησή τους.

Η εργασία αυτή αποτελεί μια μικρή εισαγωγική συμβολή στην τεχνολογία IoT. Θα πραγματοποιηθεί μια προσπάθεια ορισμού της τεχνολογίας IoT, ανάλυσης της αρχιτεκτονικής του και ανάπτυξής του σε συνεργασία με το cloud computing για την παροχή υπηρεσιών σε χρήστες. Τέλος, θα παρουσιαστεί μια εικονική αναπαράσταση του IoT σε ένα έξυπνο σπίτι.

ΠΕΡΙΕΧΟΜΕΝΑ

1	ΕΙΣΑΓΩΓΗ	15
1.1	Πρόλογος.....	15
1.2	Σκοπός Πτυχιακής Εργασίας	16
1.3	Δομή της εργασίας	17
2	ΥΠΟΒΑΘΡΟ ΚΑΙ ΥΠΑΡΧΟΥΣΕΣ ΤΕΧΝΟΛΟΓΙΕΣ	19
2.1	Υπολογιστικό Νέφος (Cloud Computing).....	19
2.1.1	Μοντέλα παροχής υπηρεσιών [2]	22
2.1.2	Μοντέλα ανάπτυξης Υπολογιστικού Νέφους [4]	23
2.1.3	Υπάρχουσες Τεχνολογίες	25
2.1.4	Εικονικοποίηση πόρων σε Υπολογιστικό Νέφος.....	33
2.1.5	Πάροχοι Υπολογιστικού Νέφους.....	34
2.2	Διαδίκτυο των Πραγμάτων (Internet of Things)	35
2.2.1	Διαδίκτυο των Πραγμάτων (Internet of Things) και Υπολογιστικό Νέφος (Cloud Computing)	37
2.2.2	Καθοριστικοί Παράγοντες Ανάπτυξης του Διαδικτύου των Πραγμάτων .	39
2.2.3	IoT περιβάλλοντα ανάπτυξης.....	41
2.2.4	Αισθητήρες και Διαδίκτυο των Πραγμάτων.....	42
3	ΕΡΓΑΛΕΙΑ ΚΑΙ ΜΕΘΟΔΟΙ	45
3.1	Εξοπλισμός	46
3.1.1	Arduino Uno [31], [32]	46
3.1.2	Ethernet Shield [32], [33]	48
3.1.3	Shield με Αισθητήρια [34]	49
3.2	Λογισμικό (Software)	51
3.2.1	Arduino IDE [32], [35].....	51
3.2.2	Βιβλιοθήκες [32]	53
3.3	Πλατφόρμα – Υπηρεσία	54
3.3.1	IBM Bluemix	54
3.3.2	Thingspeak	58
4	ΑΠΟΤΕΛΕΣΜΑΤΑ - ΣΥΜΠΕΡΑΣΜΑΤΑ	63
4.1	Οπτικοποίηση Δεδομένων με χρήση της πλατφόρμας IBM Bluemix	63
4.1.1	Σύνδεση της συσκευής με την πλατφόρμα IBM Bluemix και αποστολή δεδομένων	64
4.1.2	Οπτικοποίηση δεδομένων	77
4.2	Οπτικοποίηση Δεδομένων με χρήση της πλατφόρμας Thingspeak.....	86
4.2.1	Σύνδεση της συσκευής με την πλατφόρμα Thingspeak.....	86
4.3	Συμπεράσματα.....	92
5	ΕΠΙΛΟΓΟΣ – ΜΕΛΛΟΝΤΙΚΗ ΕΡΓΑΣΙΑ	95
5.1	ΕΠΙΛΟΓΟΣ.....	95
5.2	Μελλοντική Εργασία.....	95
5.2.1	Εξοπλισμός (Hardware).....	96
5.2.2	Arduino Yun.....	97
5.2.3	Raspberry Pi 2.....	98
5.2.4	Raspberry Pi 3.....	98
5.2.5	Βιομηχανικό Διαδίκτυο των Πραγμάτων [38]	99
5.2.6	Έξυπνο Σπίτι	102

ΛΙΣΤΑ ΕΙΚΟΝΩΝ

Εικόνα 2.1: Υπολογιστικό νέφος.....	19
Εικόνα 2.2: Εικονικοποίηση Πόρων (Virtualization)	22
Εικόνα 2.3: Τα τρία μοντέλα Παροχής Υπηρεσιών(SaaS-PaaS-IaaS).....	23
Εικόνα 2.4: Μοντέλα ανάπτυξης Υπολογιστικού Νέφους.....	24
Εικόνα 2.5: Παράδειγμα Υπηρεσιοκεντρικής Αρχιτεκτονικής	28
Εικόνα 2.6: Σύγκριση δομής IT χωρίς και με τη χρήση SOA.....	29
Εικόνα 2.7: Rest API Σχεδίαση	31
Εικόνα 2.8: Το MQTT και ο ρόλος του στο IoT μέσα από την υπηρεσία IBM Bluemix...	32
Εικόνα 2.9: Τα Συστατικά του Virtualization (Virtualization Layers)	34
Εικόνα 2.10: Internet of Things	37
Εικόνα 2.11: Διαδίκτυο των πραγμάτων και υπολογιστικό νέφος.....	38
Εικόνα 2.12: Παράδειγμα μιας πλατφόρμας ηλεκτρονικών παιχνιδιών που ανιχνεύει την κατάσταση του παίκτη με διάφορους αισθητήρες (κίνησης, πίεσης, κ.α.)	43
Εικόνα 2.13: Μέτρηση περιβαλλοντικών συνθηκών (θερμοκρασία, υγρασία, ταχύτητα του ανέμου, κ.α.)	43
Εικόνα 2.14: Παραδείγματα Ιατρικών αισθητήρων.....	44
Εικόνα 3.1: Σχηματική Απεικόνιση Πειραματικού Μέρους.....	45
Εικόνα 3.2: Arduino Uno – Άνω Όψη	46
Εικόνα 3.3: Arduino Uno – Κάτω Όψη.....	46
Εικόνα 3.4: Ethernet Shield – Άνω Όψη	48
Εικόνα 3.5: Ethernet Shield – Κάτω Όψη.....	48
Εικόνα 3.6: Σύνδεση Arduino – Ethernet Shield.....	49
Εικόνα 3.7: Shield Αισθητήρων	50
Εικόνα 3.8: Διάγραμμα Shield Αισθητήρων	50
Εικόνα 3.9: Αισθητήρας μέτρησης θερμοκρασίας LM35D	50
Εικόνα 3.10: Συνδεσμολογία Sensor shield με Arduino board – Ethernet shield	51
Εικόνα 3.11: Περιβάλλον Arduino IDE.....	52
Εικόνα 3.12: Προσθήκη Βιβλιοθήκης στο Arduino IDE	54
Εικόνα 3.13: Εγγραφή στην πλατφόρμα IBM Bluemix.....	55
Εικόνα 3.14: Δημιουργία Οργανισμού (Organization) και Χώρου Εργασίας (Space)	55
Εικόνα 3.15: Αρχική σελίδα περιβάλλοντος εργασίας.....	56
Εικόνα 3.16: Αρχική σελίδα Recipes – Αναζήτηση “Arduino Uno”	56
Εικόνα 3.17: Αποτέλεσμα αναζήτησης σχετικό με τον σκοπό της εργασίας.....	57
Εικόνα 3.18: Κοινότητα GitHub – παραδείγματα κώδικα	57
Εικόνα 3.19: Επιλογή Registered έκδοσης κώδικα	57
Εικόνα 3.20: Επιλογή dW Answers.....	58
Εικόνα 3.21: dW Answers – Αρχική σελίδα	58
Εικόνα 3.22: Εγγραφή στην πλατφόρμα Thingspeak.....	59
Εικόνα 3.23: Δημιουργία Νέου Καναλιού	60
Εικόνα 3.24: GitHub –Arduino sketches που χρησιμοποιούν την πλατφόρμα Thingspeak	61
Εικόνα 3.25: Κανάλι Thingspeak	61
Εικόνα 4.1: Αναζήτηση εφαρμογών στον κατάλογο Bluemix (Bluemix Catalog)	64
Εικόνα 4.2: Λίστα κατηγοριών εφαρμογών	65

Εικόνα 4.3: Σελίδα εφαρμογής	65
Εικόνα 4.4: Προσθήκη Νέας Συσκευής.....	66
Εικόνα 4.5 Παραμετροποίηση Συσκευής	66
Εικόνα 4.6: Παραμετροποίηση Συσκευής	67
Εικόνα 4.7: Παραμετροποίηση Συσκευής – Προαιρετικά Βήματα	67
Εικόνα 4.8: Προσθήκη Νέας Συσκευής.....	68
Εικόνα 4.9: Ασφάλεια - Authentication Token	69
Εικόνα 4.10: Περίληψη Ενεργειών.....	70
Εικόνα 4.11: Συσκευή Ρτυχιακή όπως απεικονίζεται στην πλατφόρμα	70
Εικόνα 4.12: Χαρακτηριστικά Συσκευής.....	71
Εικόνα 4.13: Παραμετροποίηση Συσκευής – Προαιρετικά Βήματα.....	72
Εικόνα 4.14: Παραμετροποίηση Συσκευής – Προαιρετικά Βήματα.....	72
Εικόνα 4.15: Ταυτοποίηση συσκευής.....	74
Εικόνα 4.16: Συνάρτηση void loop().....	75
Εικόνα 4.17: Συνάρτηση messageArrived – MQTT μήνυμα.....	75
Εικόνα 4.18: Αρχικοποίηση	76
Εικόνα 4.19: Συνάρτηση void setup()	76
Εικόνα 4.20: Υπολογισμός θερμοκρασίας.....	76
Εικόνα 4.21: Πίνακας δεδομένων Json.....	76
Εικόνα 4.22: Απεικόνιση της συσκευής «Ρτυχιακή» στην πλατφόρμα.....	77
Εικόνα 4.23: Απεικόνιση της θερμοκρασίας στην πλατφόρμα	77
Εικόνα 4.24: Κατάλογος υπηρεσιών	78
Εικόνα 4.25: Επιλογή «IoT Real-Time Insights»	78
Εικόνα 4.26: «IoT Real-Time Insights».....	79
Εικόνα 4.27: «Add a data source».....	79
Εικόνα 4.28: «Add new data source»	80
Εικόνα 4.29: Επιλογή «Arduino Setup».....	80
Εικόνα 4.30: Συνέχεια - Επιλογή «Access»	81
Εικόνα 4.31: « Generate API Key ».....	81
Εικόνα 4.32: Επιβεβαίωση ενεργειών	82
Εικόνα 4.33: Add new manage schema	82
Εικόνα 4.34: Message Schema	83
Εικόνα 4.35: Επεξεργασία Data Source	83
Εικόνα 4.36: Προσθήκη data points.....	84
Εικόνα 4.37: Απεικόνιση παραθύρου «Message Schema»	84
Εικόνα 4.38: Κατασκευή γραφήματος	85
Εικόνα 4.39: Επεξεργασία γραφήματος.....	85
Εικόνα 4.40: Ολοκλήρωση επεξεργασίας γραφήματος	86
Εικόνα 4.41: Τελική μορφή - Απεικόνιση δεδομένων.....	86
Εικόνα 4.42: Αρχική σελίδα της πλατφόρμας Thingspeak	87
Εικόνα 4.43: Δημιουργία νέου καναλιού	87
Εικόνα 4.44: Επεξεργασία Καναλιού.....	88
Εικόνα 4.45: Αρχική σελίδα καναλιού	89
Εικόνα 4.46: Δημιουργία API Keys	90
Εικόνα 4.47: Συμπλήρωση πραγματικού API χρήστη	90
Εικόνα 4.48: Αρχικοποίηση	91
Εικόνα 4.49: Συνάρτηση void setup()	91

Εικόνα 4.50: Συνάρτηση void loop()	91
Εικόνα 4.51: Ενημέρωση πλατφόρμας για την τιμή θερμοκρασίας	91
Εικόνα 4.52: Εικονοποίηση δεδομένων	92
Εικόνα 4.53: Επεξεργασία γραφήματος	92
Εικόνα 5.1: Arduino Yun	97
Εικόνα 5.2: Raspberry Pi 2	98
Εικόνα 5.3: Raspberry Pi 3	98
Εικόνα 5.4: Αισθητήρας υγρασίας	102
Εικόνα 5.5: Έξυπνο Σπίτι	104

1 ΕΙΣΑΓΩΓΗ

1.1 Πρόλογος

Τα τελευταία χρόνια η τεχνολογία αναπτύσσεται με ταχύτατους ρυθμούς, υπάρχει μεγαλύτερη ευκολία πρόσβασης στο διαδίκτυο και αξιοποίησης των υπηρεσιών του τόσο από απλούς χρήστες, όσο και από εταιρίες παροχής ψηφιακών υπηρεσιών, με αποτέλεσμα να έχουν αυξηθεί σημαντικά οι ανάγκες σε τεχνολογικές υποδομές. Αυτό οδήγησε στη δημιουργία και την ανάπτυξη των υπηρεσιών Υπολογιστικού Νέφους (Cloud Computing).

Ως Υπολογιστικό Νέφος ορίζεται το μοντέλο που επιτρέπει την αδιάκοπη, ευέλικτη και κατ' απαίτηση (on-demand) πρόσβαση σε ένα πλήθος παραμετροποιήσιμων υπολογιστικών πόρων (π.χ. δίκτυα, διακομιστές, αποθηκευμένα δεδομένα, εφαρμογές και υπηρεσίες) οι οποίοι μπορούν να δεσμευθούν και να απελευθερωθούν με την ελάχιστη δυνατή προσπάθεια.

Ταυτόχρονα, στην αγορά εισέρχονται όλο και περισσότερες συσκευές με τη δυνατότητα σύνδεσης στο Διαδίκτυο (π.χ. οικιακών συσκευών, κλειδαριών πόρτας,καλοριφέρ,κλπ), αλλά και με ενσωματωμένους αισθητήρες για τη συλλογή δεδομένων. Ο όρος «Διαδίκτυο των Πραγμάτων» (Internet of Things – IoT) αναφέρεται στη δυνατότητα διασύνδεσης μηχανών αλλά και διαδικασιών, με σκοπό τη συλλογή δεδομένων από το περιβάλλον με διάφορα μέσα (π.χ. αισθητήρες), την ανταλλαγή δεδομένων και την αξιοποίηση τους σε επιχειρηματικό και επαγγελματικό επίπεδο.

Στην παρούσα πτυχιακή εργασία, εξετάζεται με ποιούς τρόπους μπορεί να επιτευχθεί η αλληλεπίδραση των δύο παραπάνω εννοιών, Cloud Computing και IoT, πώς μπορεί να διευκολύνει την επικοινωνία μεταξύ των ανθρώπων μέσα από αυτοματισμούς, αλλά και σε ποιούς κλάδους θα μπορούσε να εφαρμοστεί. Γίνεται χρήση συγκεκριμένου εξοπλισμού (hardware) με αισθητήρες για τη συλλογή δεδομένων και εικονοποίησή τους με τη χρήση υπηρεσιών νέφους. Με τη σύγκριση των υπηρεσιών που χρησιμοποιήθηκαν και την καταγραφή των αποτελεσμάτων που είχαμε, καταλήγουμε σε συμπεράσματα για τη χρήση των τεχνολογιών και τον τρόπο που θα αποτελέσουν μέρος της καθημερινότητας μας στο άμεσο μέλλον.

1.2 Σκοπός Πτυχιακής Εργασίας

Για την επικοινωνία μεταξύ των ανθρώπων και των συσκευών, καθημερινά χρησιμοποιείται ευρυζωνική σύνδεση στο Διαδίκτυο η οποία είναι πλέον διαθέσιμη σχεδόν παντού. Για να συνδεθεί μια συσκευή με το Διαδίκτυο, θα πρέπει να έχει ένα μοναδικό αναγνωριστικό ώστε να μπορεί να διευθυνσιοδοτηθεί. Αυτό χρειάζεται, διότι ο σκοπός είναι να μοιράσει τις πληροφορίες που περιέχει ή / και συλλέγει με διάφορα μέσα (π.χ. αισθητήρες) , ανάλογα με τα χαρακτηριστικά της, σε μια παγκόσμια κλίμακα.

Σημαντική είναι η συμβολή μιας cloud πλατφόρμας η οποία θα είναι υπεύθυνη για την αποθήκευση, την διαχείριση και την επεξεργασία της τεράστιας ποσότητας δεδομένων, ώστε να μπορούν να προβληθούν με έναν κατανοητό τρόπο προς τους τελικούς χρήστες. Είναι απαραίτητη η εξεύρεση και η επιλογή της κατάλληλης υπηρεσίας που να μας εξυπηρετεί σε αυτό, ανάμεσα σε πληθώρα λύσεων με διάφορα χαρακτηριστικά, τα οποία θα επεξηγηθούν στα αντίστοιχα κεφάλαια και όπου κριθεί χρήσιμο.

Για τις ανάγκες της πτυχιακής εργασίας θα χρησιμοποιηθεί μια συσκευή (shield), δηλαδή ένα ηλεκτρονικό κύκλωμα που παρέχει πρόσθετες δυνατότητες στο χρήστη (π.χ. σύνδεση με το Διαδίκτυο, ανάγνωση δεδομένων από αισθητήρια, αποθήκευση δεδομένων κ.α.) Το συγκεκριμένο shield περιέχει τα παρακάτω αισθητήρια και ο ρόλος του εξηγείται στο αντίστοιχο υποκεφάλαιο:

- Κινητήρας DC: είναι μία διάταξη που χρησιμοποιείται για την μετατροπή της ηλεκτρικής ενέργειας σε μηχανική ενέργεια. Είναι ευρύτατα διαδεδομένη από τα μαγνητόφωνα και τα ηλεκτροκίνητα μέσα μεταφοράς (τρόλεϊ, ηλεκτρικοί σιδηρόδρομοι κ.λπ.) μέχρι τα υποβρύχια.
- Σερβο-κινητήρας: είναι ένας ηλεκτρικός κινητήρας συνεχούς ρεύματος εφοδιασμένος με αισθητήρα προσδιορισμού της θέσης του άξονα περιστροφής του.
- Proximity Sensor: Είναι ηλεκτρονικοί αισθητήρες και λειτουργούν χωρίς να έρχονται σε άμεση επαφή με το αντικείμενο που ανιχνεύουν.
- Αισθητήρας θερμοκρασίας: Οι αισθητήρες της θερμοκρασίας δεν είναι τίποτα άλλο από απλά θερμόμετρα.
- Αισθητήρας Φωτεινότητας: Πρόκειται για έναν οπτικό αισθητήρα, που έχει ως στόχο να μετρήσει τις μεταβολές του φωτός.

Από τα παραπάνω, και για τις ανάγκες της εργασίας, επιλέχθηκε να συνδεθεί ο αισθητήρας θερμοκρασίας και να ληφθούν δεδομένα αποκλειστικά από αυτόν, λόγω της εύκολης κατανόησης της χρήσης και λειτουργίας του από χρήστες χωρίς εμπειρία και εξοικείωση με το αντικείμενο της εργασίας.

Επίσης, θα χρησιμοποιηθεί ένα Arduino Uno¹ board, το οποίο με τον κατάλληλο προγραμματισμό συμβάλλει στην επεξεργασία των δεδομένων που λήφθηκαν από το shield και το διαμοιρασμό τους. Ο προγραμματισμός γίνεται με σύνδεση στον ηλεκτρονικό υπολογιστή μέσω θύρας USB, και εναλλακτικός τρόπος λειτουργίας είναι η χρήση εξωτερικής τροφοδοσίας DC. Η συσκευή ολοκληρώνεται με τη χρήση ενός Ethernet shield για τη σύνδεσή της στο Διαδίκτυο (το Arduino Uno δεν έχει από μόνο του αυτή τη δυνατότητα). Απαραίτητη είναι μια διαθέσιμη θύρα σε κάποια δικτυακή συσκευή (π.χ. δρομολογητής ή DSL modem), και ηλεκτρονικός υπολογιστής.

Θα χρειαστεί να προγραμματίσουμε κατάλληλα μέσω του Arduino IDE τη συσκευή μας ώστε να επικοινωνούν τα shields μεταξύ τους και να μπορεί να συνδεθεί στο Διαδίκτυο με τη χρήση μιας μοναδικής MAC διεύθυνσης.

Η συσκευή εκτός από την αποστολή μετρήσεων και δεδομένων που συλλέγονται από τους αισθητήρες, θα αλληλεπιδρά με υπηρεσία υπολογιστικού νέφους για την αποθήκευση, την ανάλυση και την απεικόνιση των δεδομένων σε πραγματικό χρόνο. Θα γίνει έρευνα για την επιλογή κατάλληλης πλατφόρμας και θα χρησιμοποιηθούν δύο πλατφόρμες με σκοπό τη σύγκριση τους ώστε να οδηγηθούμε σε χρήσιμα συμπεράσματα για τη σχέση Διαδικτύου των Πραγμάτων και Υπολογιστικού Νέφους.

1.3 Δομή της εργασίας

Στο **κεφάλαιο 2** γίνεται εκτενής αναφορά των τεχνολογιών που θα ερευνηθούν. Θα αναλυθεί αρχικά το Υπολογιστικό Νέφος, τα χαρακτηριστικά του, τα μοντέλα παροχής και ανάπτυξης, η ανάγκη και τα πλεονεκτήματα της χρήσης του – με σύντομη αναφορά στις όποιες αδυναμίες παρουσιάζει. Στη συνέχεια επεξηγούνται τα πρωτόκολλα (HTTP, REST, JSON, MQTT) που χρησιμοποιούνται στην παρούσα εργασία, τεχνολογίες όπως το FIWARE και τα εργαλεία που περιέχει, ο όρος της εικονικοποίησης αλλά και της υπηρεσιοκεντρικής αρχιτεκτονικής. Κλείνοντας με ό, τι αφορά το Νέφος, αναφέρονται πάροχοι υπηρεσιών Νέφους και πιο συγκεκριμένα οι πλατφόρμες που θα

¹ <https://www.arduino.cc/en/Main/ArduinoBoardUno>

Αριστείδης Σ. Βυζανιάρης - Ευάγγελος Γ. Φαραντάτος

χρησιμοποιηθούν στο πειραματικό μέρος. Το κεφάλαιο ολοκληρώνεται με την περιγραφή του Διαδικτύου των Πραγμάτων, την επίδρασή του σε διάφορους τομείς της ανθρώπινης δραστηριότητας και τη συσχέτισή του με το Υπολογιστικό Νέφος. Ακόμη, δίνονται παραδείγματα χρήσης και παρουσίας του IoT στην καθημερινότητα, και αναλύονται οι παράγοντες που επηρεάζουν την εξέλιξή του. Το **κεφάλαιο 3**, όπως αναφέρει και ο τίτλος του, έχει σκοπό να παρουσιάσει οτιδήποτε χρησιμοποιήθηκε για να ολοκληρωθεί το πειραματικό μέρος. Αυτό μπορεί να είναι ο εξοπλισμός που χρειάστηκε, το κατάλληλο λογισμικό, οι πλατφόρμες Νέφους που φιλοξενούν την υποδομή αλλά και οτιδήποτε συνέβαλε (ιστοσελίδες προϊόντων, διαδικτυακά forum – κοινότητες, οδηγό χρήσης, παραδείγματα κώδικα κ.α.) στην τροποποίηση και εν τέλει τον συνδυασμό των παραπάνω για το τελικό αποτέλεσμα. Στο **κεφάλαιο 4**, αρχικά θα αναφερθούν όλες οι απαραίτητες τροποποιήσεις που έγιναν για να επιτευχθεί ο στόχος. Αυτό είναι και σε προγραμματιστικό επίπεδο (κώδικας) αλλά και σε τροποποιήσεις που ήταν αναγκαίες προκειμένου οι πλατφόρμες να επικοινωνούν, να λαμβάνουν και να αναπαριστούν γραφικά τα δεδομένα του αισθητήρα θερμοκρασίας. Συνεχίζοντας, θα παρουσιαστούν τα αποτελέσματα της χρήσης της υποδομής, καταλήγοντας σε χρήσιμα συμπεράσματα και συγκρίσεις. Τέλος, στο **κεφάλαιο 5** περιέχονται προτάσεις για μελλοντική εξέλιξη και επέκταση της υποδομής που κατασκευάστηκε, τους τρόπους για να επιτευχθεί αυτό ως το επόμενο βήμα.

2 ΥΠΟΒΑΘΡΟ ΚΑΙ ΥΠΑΡΧΟΥΣΕΣ ΤΕΧΝΟΛΟΓΙΕΣ

2.1 Υπολογιστικό Νέφος (Cloud Computing)

Το Υπολογιστικό Νέφος είναι ένα μοντέλο που δίνει τη δυνατότητα της συνεχούς, εύκολης και υψηλών απαιτήσεων πρόσβασης σε μια κοινόχρηστη συλλογή ρυθμιζόμενων υπολογιστικών πόρων, οι οποίοι τροφοδοτούνται και απελευθερώνονται με ελάχιστη προσπάθεια διαχείρισης παροχής υπηρεσιών. [4]

Δίνει τη δυνατότητα στους χρήστες του, οι οποίοι μπορεί να είναι είτε μεμονωμένοι χρήστες Διαδικτύου, είτε ολόκληρες επιχειρήσεις ή οργανισμοί, να αποθηκεύουν, να επεξεργάζονται και να διαχειρίζονται τα δεδομένα τους. Τα δεδομένα αυτά βρίσκονται σε ένα «νέφος» απόμακρων δικτύων στο οποίο, έχουν πολύ εύκολη πρόσβαση.

Η χρήση της τεχνολογίας του Υπολογιστικού Νέφους προσφέρει μεγάλη ικανοποίηση για τον χρήστη και αυξημένα οφέλη τα οποία είναι: **Error! Reference source not found.**



Υπολογιστικό Νέφος

Εικόνα 2.1: Υπολογιστικό νέφος

Πηγή:https://en.wikipedia.org/wiki/Cloud_computing

- **Αποτελεσματικότητα με μειωμένο κόστος**

Οι εταιρίες και οι δημόσιοι οργανισμοί επενδύουν πολλά χρήματα στην δημιουργία των υποδομών πληροφορικής. Το κόστος συντήρησης και αναβάθμισης είναι ακόμη μεγαλύτερο από την ίδια τη σύσταση της υποδομής. Το Υπολογιστικό Νέφος προσφέρει

υπηρεσίες σε πολύ φθηνότερες τιμές μειώνοντας την ανάγκη για επενδύσεις στην συντήρηση και κάνοντας αναβαθμίσεις.

- **Ανάπτυξη και Κλιμάκωση.**

Όταν μια εταιρεία επιλέξει να αξιοποιήσει το Υπολογιστικό Νέφος, μπορεί να αναπτύξει το σύστημα της και να το κάνει πλήρως λειτουργικό σε πολύ λίγο χρόνο. Επίσης έχει την δυνατότητα να ξεκινήσει με μικρή ανάπτυξη ανάλογα της ανάγκης της και αν χρειαστεί, η ανάπτυξη μπορεί να κλιμακωθεί προκειμένου να ανταποκριθεί σε μεγαλύτερες ανάγκες όπως επίσης και να επιστρέψει στην αρχική της κατάσταση.

- **Αξιοπιστία.**

Οι πάροχοι του Νέφους χρησιμοποιούν κέντρα δεδομένων σε διάφορες τοποθεσίες και για αυτόν τον λόγο μπορούν να ανταπεξέλθουν σε ένα τοπικό πρόβλημα σχετικό με τον πελάτη, προσφέροντας έτσι αξιοπιστία και ποιότητα υπηρεσιών. Επίσης αυτό βοηθάει στην διατήρηση και καλή λειτουργία των εφαρμογών καθώς και στην άμεση αποκατάσταση τους μετά από κάποιο πρόβλημα καταστροφή.

Η χρήση του Υπολογιστικού Νέφους εκτός από τα πλεονεκτήματα που προσφέρει έχει και κάποια χαρακτηριστικά τα οποία προβληματίζουν τον χρήστη. Μερικά από αυτά είναι τα εξής:

- **Ασφάλεια και Προστασία Δεδομένων**

Η ασφάλεια και η προστασία δεδομένων ίσως αποτελούν τα δύο μεγαλύτερα θέματα σχετικά με το Υπολογιστικό Νέφος. Ο χρήστης παραχωρεί δεδομένα και πληροφορίες, οι οποίες είναι ενδεχομένως προσωπικές, ευαίσθητες και απόρρητες. Ο πάροχος του Νέφους είναι υπεύθυνος για την διατήρηση εμπιστευτικότητας και την προστασία των δεδομένων αυτών και γι' αυτό πρέπει να είναι αξιόπιστος.

- **Κενά ασφαλείας.**

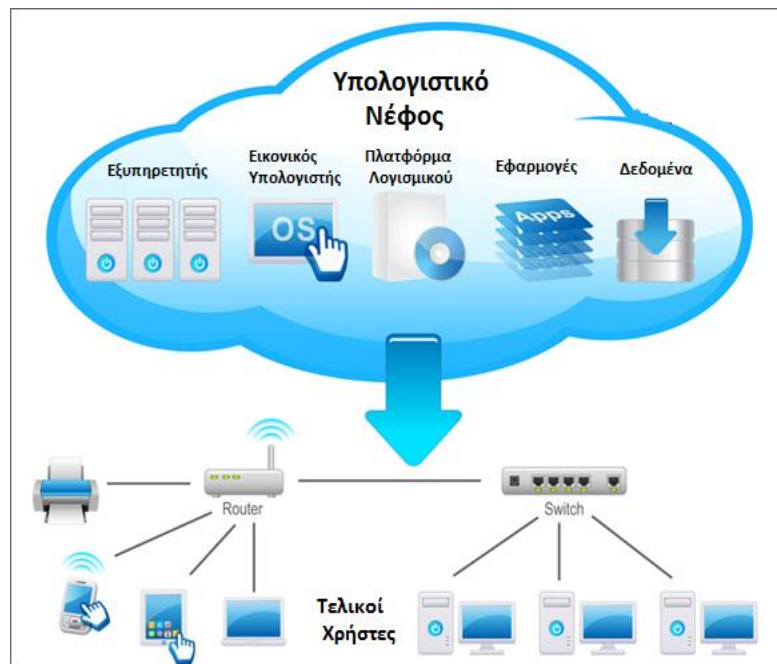
Εφόσον οι πληροφορίες αποθηκεύονται στο Νέφος και η ανταλλαγή τους γίνεται μέσω του διαδικτύου, οι υπηρεσίες που στηρίζονται στο Νέφος είναι εκτεθειμένες σε εξωτερικές απειλές όπως κακόβουλη χρήση και εισβολείς.

- **Τεχνικά Θέματα.**

Από την στιγμή που η σύνδεση στο διαδίκτυο είναι προϋπόθεση για την πρόσβαση στις υπηρεσίες του Νέφους, προβλήματα δικτύωσης και συνδεσιμότητας, τόσο από την πλευρά του παρόχου όσο και από του πελάτη, μπορούν να καταστήσουν μη διαθέσιμες όλες αυτές τις υπηρεσίες.

Πολύ σημαντικό χαρακτηριστικό του cloud είναι το virtualization. Ουσιαστικά είναι ένα πρόσθετο «επίπεδο» ανάμεσα στο hardware του υπολογιστή και το λογισμικό που χρησιμοποιείται σε αυτό το οποίο δίνει μια ιδεατή μορφή των υπολογιστικών πόρων αντί για τη φυσική τους μορφή. Με αυτό τον τρόπο «ξεγελά» το λειτουργικό σύστημα, καθώς πλέον ένα σύνολο εξυπηρετητών είναι ορατό ως ένα ενιαίο σύνολο υπολογιστικών πόρων. Η τεχνική αυτή έχει τις ρίζες της στο partitioning, το οποίο διαιρεί έναν μοναδικό φυσικό server σε πολλαπλούς ιδεατούς. Από τη στιγμή που ο φυσικός server διαιρείται, κάθε ιδεατός μπορεί να «τρέχει» το δικό του λειτουργικό σύστημα και εφαρμογές ανεξάρτητα από τους άλλους. Επιτυγχάνεται λοιπόν εξοικονόμηση πόρων, ο χρήστης δεσμεύει μόνο τους πόρους που θέλει και όποτε το θέλει, ενώ οι υπόλοιποι πόροι διατίθενται σε άλλους χρήστες. Αυτό γίνεται άμεσα, οι πόροι εμφανίζονται ως μη διαθέσιμοι (scale out) και επίσης αποδεσμεύονται γρήγορα για να εμφανιστούν ξανά ως διαθέσιμοι (scale in). Για τον χρήστη, οι διαθέσιμες δυνατότητες για δέσμευση και χρήση δεν είναι δεσμευτικές και μπορούν να αλλάξουν ανά πάσα στιγμή και σε οποιαδήποτε ποσότητα. Από την πλευρά του παρόχου, μικρότερο πλήθος μηχανημάτων σε ένα data center σημαίνει και χαμηλότερη κατανάλωση ενέργειας, τόσο από τους ίδιους τους servers όσο και από τον εξοπλισμό ψύξης που χρησιμοποιείται για να αποφευχθεί φαινόμενο υπερθέρμανσης.

Για μια λύση virtualization χρειάζεται κατάλληλος εξοπλισμός hardware, λογισμικό virtualization (hypervisor), και λογισμικό διαχείρισης. Το βασικό στοιχείο είναι ο hypervisor, το λογισμικό δηλαδή που ξεχωρίζει το λειτουργικό σύστημα και τις εφαρμογές του από τους φυσικούς πόρους και εγκαθίσταται απευθείας στο hardware. Τα ιδεατά συστήματα συνεχίζουν να «νομίζουν» ότι επικοινωνούν απευθείας με το hardware, αλλά στην πραγματικότητα επικοινωνούν με το virtualization layer. Οι απαιτήσεις για αποδοτική χρήση του hardware, η μείωση της ηλεκτρικής κατανάλωσης και των αναγκών ψύξης, γίνονται ολοένα και πιο επιτακτικές. Αυτό έχει ως αποτέλεσμα το virtualization να έχει εδραιωθεί ως λύση σε πολλές επιχειρήσεις και αναμένεται γρήγορα τα περισσότερα συστήματα να γίνουν ιδεατά. [3]



Εικόνα 2.2: Εικονικοποίηση Πόρων (Virtualization)

Πηγή: <https://thecustomizewindows.com/category/computer-and-internet/virtualization/>

2.1.1 Μοντέλα παροχής υπηρεσιών Error! Reference source not found.

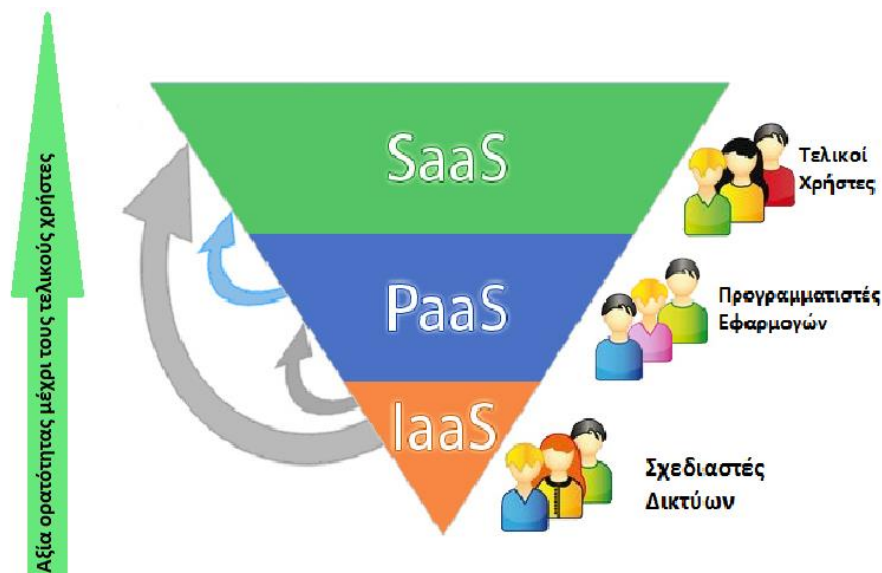
Το Υπολογιστικό Νέφος παραδοσιακά προσφέρεται σύμφωνα με τρία διαφορετικά μοντέλα χρήσης (service models): **Λογισμικό ως Υπηρεσία** (Software as a Service - SaaS), **Πλατφόρμα ως Υπηρεσία** (Platform as a Service - PaaS) και **Υποδομή ως Υπηρεσία** (Infrastructure as a Service - IaaS).

Το **SaaS** είναι το μοντέλο, όπου ο πάροχος του Νέφους δίνει πρόσβαση σε εφαρμογές, που έχουν ήδη αναπτυχθεί στην υποδομή του. Οι εφαρμογές είναι τυπικά διαθέσιμες μέσω διεπαφών διαδικτύου, όπως ένας περιηγητής δικτύου σε συσκευές που καλύπτουν ένα ευρύ φάσμα (π.χ. προσωπικούς υπολογιστές, tablets, smartphones και άλλες φορητές συσκευές). Ο χρήστης δεν μπορεί να επηρεάσει τους διακομιστές, τα λειτουργικά συστήματα ή τους αποθηκευτικούς χώρους και στις περισσότερες περιπτώσεις δεν έχει καθόλου έλεγχο πάνω στην ίδια την εφαρμογή. Το Gmail, το Dropbox, το Yahoo Mail είναι μερικά παραδείγματα ευρέως χρησιμοποιούμενων εφαρμογών SaaS.

Το **PaaS** είναι το μοντέλο, όπου ο πάροχος επιτρέπει στους χρήστες να αναπτύξουν τις δικές τους εφαρμογές σε μία υποδομή νέφους, χρησιμοποιώντας προγραμματιστικές γλώσσες και εργαλεία που έχουν δημιουργηθεί και υποστηρίζονται από τον πάροχο. Ο καταναλωτής δεν διαχειρίζεται ούτε ελέγχει το υφιστάμενο δίκτυο (όπου λειτουργεί η

υποδομή), τους διακομιστές, τα λειτουργικά συστήματα ή τους αποθηκευτικούς χώρους, αλλά μπορεί να ελέγξει τις ίδιες τις εφαρμογές και σε μερικές περιπτώσεις το περιβάλλον των εφαρμογών. Το Google App Engine και το Amazon Web Services είναι σήμερα τα κύρια παραδείγματα για το PaaS.

Το **IaaS** είναι το μοντέλο, όπου ο πάροχος δίνει την δυνατότητα επεξεργασίας και αποθήκευσης των δικτύων και άλλων βασικών υπολογιστικών πόρων απευθείας στον χρήστη. Στη συνέχεια ο χρήστης αναπτύσσει και «τρέχει» το δικό του λογισμικό που περιλαμβάνει λειτουργικά συστήματα και εφαρμογές. Ο καταναλωτής δεν διαχειρίζεται ούτε ελέγχει την υφιστάμενη υποδομή του νέφους αλλά ελέγχει τα λειτουργικά συστήματα, τον χώρο αποθήκευσης, τις εφαρμογές ανάπτυξης και πιθανώς ελέγχει σε μικρότερο βαθμό κάποια στοιχεία όπως το τείχος προστασίας.



Εικόνα 2.3: Τα τρία μοντέλα Παροχής Υπηρεσιών(SaaS-PaaS-IaaS)
Πηγή: <https://qarea.com/articles/cloud-computing-outlook-iaas-paas-and-saas>

2.1.2 Μοντέλα ανάπτυξης Υπολογιστικού Νέφους [4]

Εκτός από τα μοντέλα παροχής υπηρεσιών νέφους (SaaS, PaaS & IaaS), μπορούμε να διακρίνουμε και τέσσερα μοντέλα ανάπτυξης υπηρεσιών (deployment models). Τα τέσσερα μοντέλα είναι τα:



Εικόνα 2.4: Μοντέλα ανάπτυξης Υπολογιστικού Νέφους

Πηγή:<http://bigdata-guide.blogspot.gr/2014/01/types-of-cloud-computing-public-private.html?view=classic>

- **Δημόσιο νέφος (Public cloud):** Η υποδομή του «νέφους» διατίθεται στο ευρύ κοινό και ανήκει σε έναν οργανισμό που πουλάει τις υπηρεσίες. Οι πιο κοινές μορφές του Public cloud είναι προσβάσιμες μέσω Διαδικτύου.
- **Ιδιωτικό νέφος (Private cloud):** η δομή του μοντέλου αυτού λειτουργεί αποκλειστικά για έναν οργανισμό. Μπορεί να την διαχειρίζεται ένας οργανισμός ή ένα τρίτο μέρος και μπορεί να βρίσκεται είτε εκτός είτε εντός των εγκαταστάσεων του οργανισμού. Όπως είναι λογικό, τα private clouds ανταπεξέρχονται καλύτερα στα πρότυπα ασφάλειας των δεδομένων, ακριβώς επειδή είναι ιδιωτικά.
- **Κοινοτικό νέφος (Community cloud):** το μοντέλο αυτό διαμοιράζεται σε διάφορους οργανισμούς και υποστηρίζει μια συγκεκριμένη κοινότητα που έχει κοινά συμφέροντα και κοινές προτεραιότητες. Η διαχείριση του community cloud μπορεί να γίνεται από τους ίδιους τους συμμετέχοντες είτε από τρίτο (πάροχο). Η φυσική θέση της υποδομής μπορεί να είναι είτε εντός είτε εκτός των εγκαταστάσεων.
- **Υβριδικό νέφος (Hybrid cloud):** η υποδομή cloud είναι μια σύνθεση δύο ή περισσότερων cloud (private, community ή public) όπου παραμένουν ως μοναδικές οντότητες. Η μεταξύ τους σύνδεση όμως, επιτυγχάνεται με τυποποιημένη ή αποκλειστική τεχνολογία που επιτρέπει φορητότητα δεδομένων και εφαρμογών.

2.1.3 Υπάρχουσες Τεχνολογίες

2.1.3.1 Υποδομή FIWARE και Intellicloud [5], [6]

Το FIWARE είναι ένα cloud περιβάλλον βασισμένο στο Openstack (λειτουργική υπηρεσία σύννεφου, η οποία αναλύεται παρακάτω). Παρέχει ένα πλούσιο σύνολο από standard APIs (διεπαφές προγραμματισμού εφαρμογών) ανοιχτού κώδικα ώστε να είναι απλούστερη η σύνδεση με το Internet of Things. Επίσης, βοηθάει στη επεξεργασία και την ανάλυση δεδομένων μεγάλης κλίμακας (Big Data), εκπομπή πολυμεσικού περιεχομένου σε πραγματικό χρόνο ή ενσωματώνει εξελιγμένα χαρακτηριστικά διαδραστικότητας με το χρήστη (δυνατότητα σύνδεσης με συσκευές IoT, εφαρμογές διαχείρισης και παρακολούθησης δεδομένων σε 2 ή και 3 διαστάσεις κ.α.)

Το Intellicloud είναι μια πειραματική cloud υποδομή για τον σχεδιασμό Internet εφαρμογών βασισμένων στο cloud. Είναι περιβάλλον ανοιχτού κώδικα, για ερευνητές και προγραμματιστές που στοχεύουν να σχεδιάσουν και να αναπτύξουν μελλοντικές cloud εφαρμογές. Η υποδομή, στα πρότυπα του FI-WARE, προσφέρει υπηρεσίες που περιλαμβάνουν κάποια προορισμένα εργαλεία γενικού σκοπού που ονομάζονται Generic Enablers (GEs), και επεξηγούνται σε ξεχωριστό υποκεφάλαιο. Η σύνθετη χρήση διαφόρων GEs για την υλοποίηση του σκοπού του χρήστη, αποτελεί μια υπηρεσία ειδικού σκοπού (Specific Enabler – SE).

Η υπηρεσία Intellicloud υλοποιήθηκε και χρησιμοποιείται από ερευνητές για πειραματικούς σκοπούς, συμπεριλαμβανομένης της ανάπτυξης Generic Enabler για εξερεύνηση, διαμόρφωση και έλεγχο – testing για το Future Internet Social and Technological Alignment Research (FI - STAR, www.fi-star.eu). Το λειτουργικό OpenStack (Grizzly) datacenter βρίσκεται στο Πολυτεχνείο Κρήτης (TUC) και έχει χωρητικότητα: 128 Cores, 284 GB RAM , 12 TB HDD (24 TB Mirroring). Το Intellicloud θα φιλοξενήσει όλους τους FI-STAR Specific Enablers (SES) για την προώθηση των εξελίξεων της νέας εφαρμογής FI στον τομέα των υπηρεσιών υγειονομικής περίθαλψης και σύννεφου.

2.1.3.2 Openstack/AWS/IBM Bluemix [7]

Στην αγορά, είναι διαθέσιμο ένα πλήθος υπηρεσιών σύννεφου που παρέχεται από εταιρείες με δραστηριότητα στο χώρο της πληροφορικής και της τεχνολογίας. Ο χρήστης, ανάλογα με το αποτέλεσμα που επιθυμεί, τις ανάγκες του σε πόρους αλλά και το χρηματικό ποσό που διαθέτει μπορεί να συγκρίνει και να επιλέξει την κατάλληλη

υπηρεσία για την υλοποίηση του στόχου του. Οι υπηρεσίες σύννεφου μπορεί να διατίθενται ακόμα και χωρίς κανένα κόστος, ή με δωρεάν χρήση και περιορισμό πόρων (π.χ. ανά μήνα) και μπορεί να περιέχουν διαφορετικά εργαλεία μεταξύ τους τα οποία ο χρήστης καλείται να αξιολογήσει και να αξιοποιήσει. Ορισμένες λύσεις υπηρεσιών σύννεφου είναι παρακάτω:

Το Openstack είναι μια λειτουργική υπηρεσία σύννεφου ανοικτού κώδικα που διαχειρίζεται ένα μεγάλο όγκο υπολογιστικών (compute), αποθηκευτικών (storage) και δικτυακών (networking) πόρων μέσω ενός datacenter. Όλα τα παραπάνω διαχειρίζονται από ένα ταμπλό (dashboard) που δίνει στους διαχειριστές (administrators) τον έλεγχο, ενώ εξουσιοδοτεί τους χρήστες του με την παροχή πόρων μέσω ενός web interface.

Το Amazon Web Services (AWS) είναι μια ασφαλής πλατφόρμα υπηρεσιών cloud που προσφέρει υπολογιστική ισχύ, αποθηκευτικό χώρο δεδομένων, διανομή περιεχομένου και άλλες λειτουργίες για να βοηθήσει τις διάφορες επιχειρήσεις να αναπτυχθούν. Εκατομμύρια πελάτες χρησιμοποιούν προϊόντα και λύσεις AWS για την κατασκευή εξελιγμένων εφαρμογών με αυξημένη ευελιξία, επεκτασιμότητα και αξιοπιστία. Το AWS Cloud παρέχει αυτό το ευρύ σύνολο των υπηρεσιών με την εξής χρησιμότητα: on-demand, διαθέσιμο σε δευτερόλεπτα και με τιμολόγηση pay - as - you - go. [8]

Το IBM Bluemix είναι μια πλατφόρμα cloud ως υπηρεσία (PaaS) που αναπτύχθηκε από την IBM. Υποστηρίζει πολλές γλώσσες και υπηρεσίες προγραμματισμού, καθώς και ολοκληρωμένες DevOps ώστε να μπορεί ο χρήστης να χτίσει, να τρέξει, να αναπτύξει και να διαχειριστεί εφαρμογές στο cloud. Το Bluemix βασίζεται στην ανοιχτή τεχνολογία Cloud Foundry και τρέχει σε υποδομές SoftLayer. Το Bluemix επίσης υποστηρίζει πολλές γλώσσες προγραμματισμού, συμπεριλαμβανομένων Java, PHP, Python κ.α. [9]

Εκτός από τις προαναφερθείσες υπηρεσίες, υπάρχουν και πολλές άλλες λύσεις διαθέσιμες, από διαφορετικούς παρόχους υπολογιστικού νέφους (αναφέρονται σε ξεχωριστό υποκεφάλαιο παρακάτω). Απλές δομές αποθήκευσης δεδομένων (π.χ. Apple iCloud, Dropbox, Google Drive κ.α.), σύνθετες πλατφόρμες με εφαρμογές ή και εργαλεία για ανάπτυξη νέων (π.χ. AWS, IBM Bluemix, κ.α.) είναι προσβάσιμες μέσω της δημιουργίας λογαριασμού χρήστη (πάντα με τους περιορισμούς που μπορεί να ορίζει ο πάροχος). Οι παραπάνω λύσεις αναφέρθηκαν λόγω του εύρους και του πλήθους των δυνατοτήτων τους, της δωρεάν χρήσης τους (με περιορισμούς πόρων ανά περίπτωση)

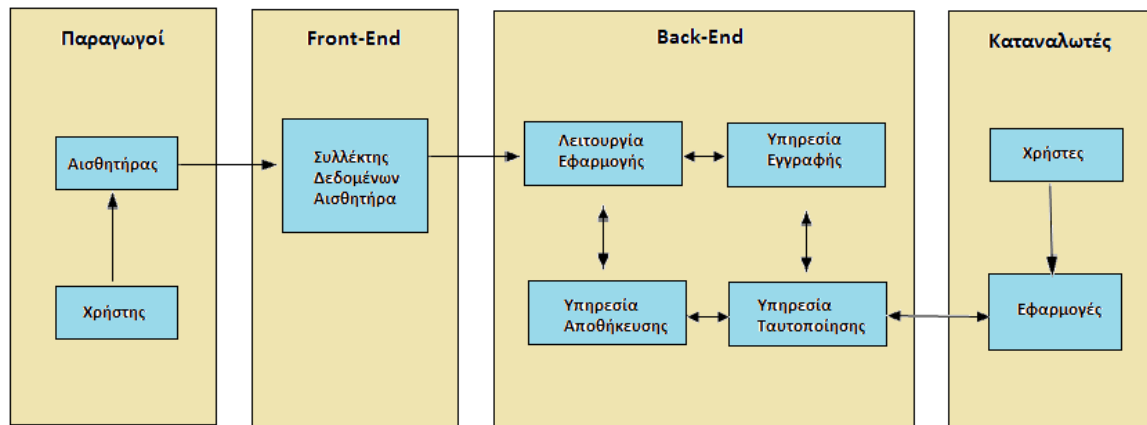
και της κατανοητής δομής τους σε χρήστες χωρίς εμπειρία στον τομέα IoT και υπηρεσιών σύννεφου.

2.1.3.3 Υπηρεσίες Γενικού Σκοπού (Generic Enablers) και Ειδικού Σκοπού (Specific Enablers) [10], [11]

Το FIWARE βασίζεται σε εργαλεία Γενικού Σκοπού (Generic Enablers) τα οποία προσφέρουν επαναχρησιμοποιούμενες λειτουργίες (functions) σε διάφορους τομείς (π.χ. Cloud Enablers, Security Enablers, Data Enablers κ.α.) Διατίθενται από το FIWARE με τη μορφή SaaS και ο χρήστης μπορεί να επιλέξει ένα ή και παραπάνω που εξυπηρετούν το σκοπό του. Για την αντιμετώπιση πιο σύνθετων προβλημάτων είναι δυνατή η χρήση τους συνδυαστικά, οπότε πλέον λειτουργούν σαν ένας Specific Enabler (SE). Η ικανότητα να εξυπηρετούν μια πολλαπλότητα περιοχών χρήσης είναι αυτή που διακρίνει τους Generic Enablers από τους Domain-specific Common Enablers (ή «Specific Enablers»), οι οποίοι είναι κοινοί σε πολλές εφαρμογές, αλλά όλοι τους συγκεκριμένοι σε ένα πολύ περιορισμένο σύνολο περιοχών χρήσης (όπως της υγείας, της ενέργειας κ.α.). Βασικός στόχος του FIWARE είναι ο προσδιορισμός και καθορισμός των GEs, σε συνδυασμό με την ανάπτυξη και επίδειξη εφαρμογών αναφοράς των GEs. Κάθε εφαρμογή ενός GE αποτελείται από ένα σύνολο στοιχείων και θα προσφέρει δυνατότητες και λειτουργίες που μπορούν με ευελιξία να προσαρμοστούν (customization), χρησιμοποιούνται και συνδυάζονται για πολλούς διαφορετικούς τομείς χρήσης, επιτρέποντας την ανάπτυξη προηγμένων και καινοτόμων εφαρμογών και υπηρεσιών διαδικτύου.

2.1.3.4 Υπηρεσιοκεντρική αρχιτεκτονική (Service Oriented Architecture)

Υπηρεσιοκεντρική [12] είναι η αρχιτεκτονική υλοποίησης συστημάτων που βασίζεται στη λογική ότι οποιοδήποτε πρόβλημα μπορεί να αντιμετωπιστεί αποτελεσματικότερα αν το διαιρέσουμε σε μικρότερα υποπροβλήματα (τα οποία συνθέτουν το αρχικό). Το ίδιο ισχύει και στην περίπτωση του IoT, σε συστήματα δηλαδή τα οποία βασίζονται σε υπηρεσίες του διαδικτύου. Η επικοινωνία αυτού του συνόλου υπηρεσιών μπορεί να περιλαμβάνει μια απλή ανταλλαγή δεδομένων, ή το συντονισμό μιας εργασίας από δύο ή παραπάνω υπηρεσίες. [20]

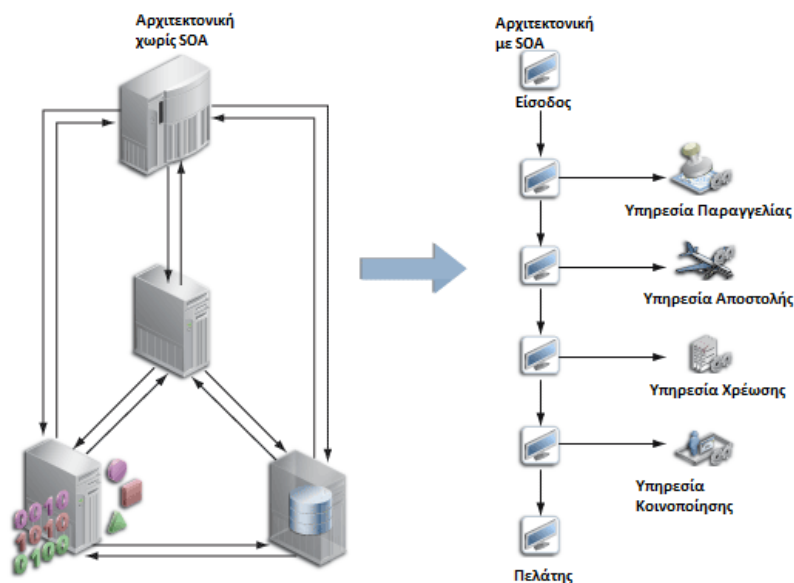


Εικόνα 2.5: Παράδειγμα Υπηρεσιοκεντρικής Αρχιτεκτονικής

Πηγή: Δούζης Κ., (2015) Πολυτεχνείο Κρήτης, Διπλωματική Εργασία: Διαχείριση συσκευών «Διαδικτύου των Πραγμάτων» στο «Υπολογιστικό Νέφος»

Βασικό πλεονεκτήματα αυτής της αρχιτεκτονικής είναι οι επαναχρησιμοποιήσιμες και αυτόνομες υπηρεσίες. Αυτό έχει ως όφελος το χαμηλότερο κόστος για ανάπτυξη και διαχείριση λογισμικού. Αν μια υπηρεσία δεν είναι ανεπτυγμένη ανεξάρτητα, αλλά σύμφωνα με τα πρότυπα των αναγκών συγκεκριμένων συστημάτων, είναι αναμενόμενο να αποτύχει μακροπρόθεσμα. [13][14],[15]

Η υπηρεσιοκεντρική αρχιτεκτονική (SOA) δεν είναι μια τεχνολογία, αλλά μια φιλοσοφία σχεδίασης ανεξάρτητη από κατασκευαστές, προϊόντα και τεχνολογίες. Οι ανάγκες της κάθε βιομηχανίας – επιχείρησης - οργανισμού, διαφέρουν για αυτό και η αγορά «ολοκληρωμένης» δομής SOA από συγκεκριμένο κατασκευαστή εναντιώνεται στη φύση της. Το γεγονός ότι μια συγκεκριμένη σχεδίαση SOA είναι λειτουργική και αποδοτική για έναν οργανισμό δε σημαίνει πως μπορεί να εφαρμοστεί με την ίδια επιτυχία και σε κάποιον άλλο. [16]



Εικόνα 2.6: Σύγκριση δομής IT χωρίς και με τη χρήση SOA

Πηγή : http://docs.oracle.com/cd/E28280_01/doc.1111/e10223/soa_02.htm

2.1.3.5 Πρωτόκολλο επικοινωνίας HTTP και REST API

Το Πρωτόκολλο Μεταφοράς Υπερκειμένου **HTTP** (HyperText Transfer Protocol) λειτουργεί στο επίπεδο Εφαρμογής (Layer 7 – Application Layer) σύμφωνα με το μοντέλο OSI και είναι το πρωτόκολλο μεταφοράς του Web. Το HTTP υλοποιείται σε δυο μέρη: ένα πρόγραμμα πελάτη (φυλλομετρητής - browser) και ένα πρόγραμμα εξυπηρετητή/διακομιστή. Το πρόγραμμα πελάτη και το πρόγραμμα εξυπηρετητή, που εκτελούνται σε διαφορετικά συστήματα, συνομιλούν μεταξύ τους ανταλλάσσοντας μηνύματα HTTP.

Ο πελάτης πραγματοποιεί μια σύνδεση TCP-IP με τον εξυπηρετητή χρησιμοποιώντας το domain name, τη διεύθυνση IP και τον αριθμό της θύρας (port number) που δίνονται στη διεύθυνση. Αν η θύρα δεν έχει καθοριστεί, η προεπιλεγμένη θύρα είναι η 80. Για να πραγματοποιηθεί η σύνδεση, πρέπει να γίνει αποδεκτή από τον διακομιστή. Τα δεδομένα που στέλνονται με ένα HTTP αίτημα ή η απάντηση, είναι σε μορφή και κωδικοποίηση που ορίζεται στο πεδίο header, με την προεπιλεγμένη μορφή να είναι απλό κείμενο με κωδικοποίηση 8 bit.

Το HTTP υποστηρίζει ένα σύνολο μεθόδων που χρησιμοποιούνται με τις βασικότερες να είναι οι **GET** (ο διακομιστής ανακτά τα δεδομένα που ζητούνται από τον πελάτη), **POST** (δημιουργία νέου πόρου στον διακομιστή), **PUT** (ο πελάτης ζητά την ανανέωση/ενημέρωση της κατάστασης ενός υπάρχοντος πόρου) και **DELETE** (αίτημα διαγραφής συγκεκριμένου-ων πόρου-ων από το διακομιστή) [16]

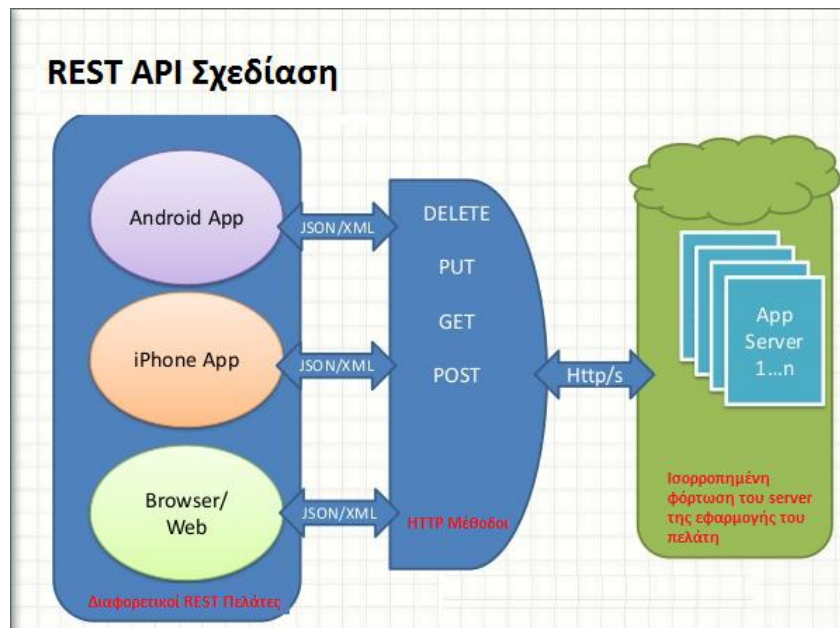
Το **REST** (Representational State Transfer) είναι μια αρχιτεκτονική, ένα σύνολο από κανόνες και αρχές για τη δημιουργία διαδικτυακού λογισμικού και υπηρεσιών. Αποτελεί ένα εξιδανικευμένο μοντέλο για όλες τις διαδράσεις που εκτελούνται στο εσωτερικό μιας διαδικτυακής εφαρμογής και είναι το θεμέλιο της μοντέρνας αρχιτεκτονικής Web. Θα μπορούσε να θεωρηθεί πως είναι η απάντηση στην ανάγκη ύπαρξης ενός μοντέλου για το πώς πρέπει να λειτουργεί το Διαδίκτυο.

Η αρχιτεκτονική REST είναι βασισμένη στο μοντέλο πελάτης-εξυπηρετητής. Για την επικοινωνία μεταξύ πελάτη και εξυπηρετητή γίνεται αποκλειστική χρήση HTTP αιτημάτων και μεθόδων. Ο πελάτης μπορεί να επιλέξει τη μορφή των περιεχομένων της απάντησης από τον εξυπηρετητή, με τις πιο διαδεδομένες μορφές να είναι JSON και XML. Στην αρχιτεκτονική REST είναι βασισμένοι οι Generic Enablers, που αναφέρθηκαν παραπάνω.

Επίσης, η «stateless» ιδιότητα υπηρεσιών REST (δηλαδή «χωρίς κατάσταση») σημαίνει πως ο εξυπηρετητής δε χρειάζεται να γνωρίζει προηγούμενες καταστάσεις για να διεκπεραιώσει ένα αίτημα. Μια υπηρεσία που ο χρήστης ταχτοποιείται και μετά ανακτά μια πληροφορία (άρα ο εξυπηρετητής πρέπει να γνωρίζει πως προηγουμένως έχει γίνει ταυτοποίηση και να κρατά αυτή την κατάσταση) δεν είναι «stateless». Αντιθέτως, στο HTTP αίτημα πρέπει να περιλαμβάνεται ολόκληρη η πληροφορία που χρειάζεται ο εξυπηρετητής για να αποκριθεί. [18]

Ένα **API** (Application Programming Interface) αποτελείται από ένα σύνολο εργαλείων, προτύπων και πρωτοκόλλων για την κατασκευή (προγραμματισμό) εφαρμογών και υπηρεσιών στο διαδίκτυο. Τα APIs προσφέρονται στο κοινό από τους προμηθευτές των υπηρεσιών, έτσι ώστε οι προγραμματιστές να μπορούν να σχεδιάσουν εφαρμογές. Το API καθορίζει, πώς πρέπει να αλληλεπιδρούν τα μέρη – εξαρτήματα, του λογισμικού που χρησιμοποιούνται στον προγραμματισμό γραφικών διεπαφών χρήστη (Graphical User Interface – GUI). Το API πρέπει να διευκολύνει την ανάπτυξη ενός προγράμματος, παρέχοντας όλα τα απαραίτητα «δομικά» στοιχεία που χρειάζεται ο προγραμματιστής για να συνθέσει το πρόγραμμα. Ορισμένα δημοφιλή παραδείγματα APIs είναι το Google Maps API, το οποίο δίνει τη δυνατότητα στους προγραμματιστές να εισάγουν (embed) τους χάρτες Google στις ιστοσελίδες, το YouTube API που δίνει τη δυνατότητα προβολής βίντεο σε εφαρμογές ή ιστοσελίδες, το Twitter API που δίνει πρόσβαση στο κοινωνικό δίκτυο και στα δεδομένα του Twitter, κ.α. [19]

Αριστείδης Σ. Βυζανιάρης - Ευάγγελος Γ. Φαραντάτος



Εικόνα 2.7: Rest API Σχεδίαση

Πηγή: <http://www.slideshare.net/DeviKiranGonuguntla/rest-api-design>

2.1.3.6 Πρωτόκολλο MQTT [20][21]

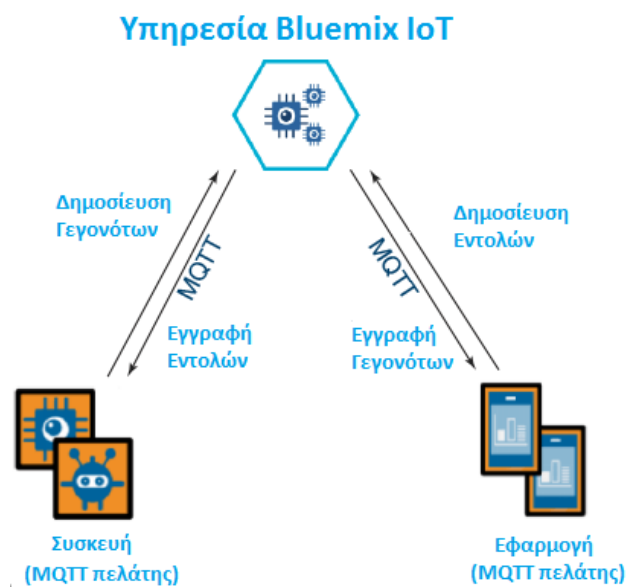
Το MQTT (Message Queueing Telemetry Transport) είναι ένα «ελαφρύ» πρωτόκολλο ταχείας επικοινωνίας μεταξύ συσκευών (machine-to-machine) σχεδιασμένο για το IoT. Χρησιμοποιήθηκε από την IBM (όπου και κατασκευάστηκε το 1999 από τον Andy Stanford-Clark) και ο σκοπός του ήταν να είναι ένα πρωτόκολλο επικοινωνίας για μεταφορά μηνυμάτων του μοντέλου «publish/subscribe». Το μοντέλο αυτό, είναι ένα πρότυπο επικοινωνίας μεταξύ αποστολέων και παραληπτών, ασύγχρονα και ανώνυμα. Παράδειγμα τέτοιας επικοινωνίας είναι όταν ένα πλήθος χρηστών εγγράφονται σε ένα θέμα που τους ενδιαφέρει σε έναν εξυπηρετητή (server), και ειδοποιούνται μέσω ηλεκτρονικού ταχυδρομείου για νέες ενημερώσεις σχετικές με το θέμα αυτό.

Το MQTT είναι σχεδιασμένο για μη αξιόπιστα, χαμηλού εύρους ζώνης (bandwidth) δίκτυα, με υψηλή καθυστέρηση μεταφοράς πακέτων δεδομένων (latency). Η αρχή σχεδιασμού του ήταν να χρησιμοποιεί το ελάχιστο δυνατό bandwidth του δικτύου και πόρων μιας συσκευής (μνήμη, κατανάλωση ενέργειας), και να προσφέρει κατά κάποιο τρόπο αξιοπιστία παράδοσης του μηνύματος σε έναν ή και περισσότερους παραλήπτες. Είναι ιδανικό για χρήση επειδή:

- Είναι ασύγχρονο, με διαφορετικά επίπεδα ποιότητας υπηρεσιών, το οποίο είναι σημαντικό σε περιπτώσεις μη αξιόπιστων συνδέσεων με το Διαδίκτυο.

- Στέλνει μικρού μεγέθους μηνύματα που το κάνουν εύχρηστο σε περιπτώσεις χαμηλού bandwidth.
- Σε περιπτώσεις συσκευών με περιορισμένη μνήμη (όπως το Arduino) δε χρειάζεται περίπλοκο λογισμικό για να εκτελεστεί.

Το πρωτόκολλο MQTT, στην παρούσα εργασία, θα χρησιμοποιηθεί στον κώδικα με τον οποίο προγραμματίζεται το Arduino board για την επικοινωνία του με την πλατφόρμα IBM Bluemix .



Εικόνα 2.8: Το MQTT και ο ρόλος του στο IoT μέσα από την υπηρεσία IBM Bluemix

Πηγή : <http://www.ibm.com/developerworks/cloud/library/cl-mqtt-bluemix-iot-node-red-app/>

2.1.3.7 JSON (JavaScript Object Notation)

Η **JSON** (JavaScript Notation Object) είναι ένα «ελαφρύ» πρότυπο ανταλλαγής δεδομένων, για την επικοινωνία μεταξύ διακομιστή και πελάτη ή ενός διακομιστή με κάποιον άλλο διακομιστή. Είναι εύκολη στη γραφή και την ανάγνωση και εύκολη για τις μηχανές να την αναλύσουν και να την αναπαράγουν. Βασίζεται σε ένα υποσύνολο της γλώσσας προγραμματισμού JavaScript. Η μορφή σύνταξης JSON είναι τελείως ανεξάρτητη αλλά χρησιμοποιεί στοιχεία που είναι οικεία για τους προγραμματιστές της οικογένειας της C όπως C , C ++ , C # , Java , JavaScript , Perl , Python κ.α. Αυτές οι ιδιότητες καθιστούν τη JSON μια ιδανική γλώσσα ανταλλαγής δεδομένων. Η JSON είναι χτισμένη σε δύο δομές :

- Μια συλλογή από ζεύγη ονόματος / τιμής. Σε διάφορες γλώσσες , αυτό υλοποιείται ως ένα αντικείμενο (object), εγγραφή (record), struct, κ.α.

- Μια ταξινομημένη λίστα τιμών. Στις περισσότερες γλώσσες, αυτή πραγματοποιείται ως πίνακας (array), vector, λίστα, ή ακολουθία.

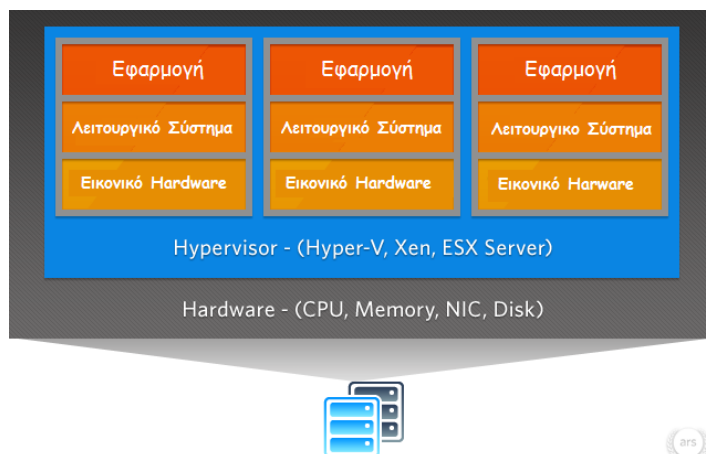
Αυτές είναι δομές δεδομένων καθολικής χρήσης. Σχεδόν όλες οι σύγχρονες γλώσσες προγραμματισμού τις στηρίζουν στη μία ή την άλλη μορφή. Στην παρούσα εργασία, η JSON εξυπηρετεί στην ανταλλαγή δεδομένων του αισθητήρα μέσω της κατασκευής και της πλατφόρμας IBM Bluemix. [22]

2.1.4 Εικονικοποίηση πόρων σε Υπολογιστικό Νέφος

Παραπάνω, αναφέραμε το κύριο χαρακτηριστικό μιας υπηρεσίας υπολογιστικού νέφους που είναι το virtualization. Αναλύσαμε το πώς το hardware διαχωρίζεται από το software (λειτουργικά συστήματα και εφαρμογές), με τη βοήθεια του hypervisor. Τα τρία χαρακτηριστικά του virtualization είναι:

- Οι **Εικονικές Μηχανές** (Virtual Machines) που αποτελούν ένα σύνολο από εικονικούς πόρους (επεξεργαστική ισχύ, μνήμη, αποθηκευτικό χώρο κ.α), ένα φιλοξενούμενο λειτουργικό σύστημα (π.χ. Linux Ubuntu, CentOS, Devian κ.α.) και τις εφαρμογές που εκτελούνται πάνω σε αυτό το λειτουργικό σύστημα.
- Ο **Επιμελητής Εικονικών Μηχανών** (hypervisor), ο οποίος διανέμει τους πόρους στις Εικονικές Μηχανές ανάλογα με τις απαιτήσεις τους.
- Οι **Φυσικοί Πόροι**, που αποτελούν την πραγματική επεξεργαστική ισχύ, μνήμη, αποθηκευτικό χώρο και δικτύωση που διαθέτει η Υποδομή Νέφους.

Ένα όφελος του virtualization, είναι η άμεση διαθεσιμότητα των υπολογιστικών πόρων. Επίσης περιλαμβάνονται τεχνικές όπως «clustering» (η τεχνική κατά την οποία πολλοί υπολογιστές συνδέονται μεταξύ τους και πλέον λειτουργούν ως ενιαία μονάδα) για την αντιμετώπιση προβλημάτων σε ένα φυσικό εξυπηρετητή ή κάποια σύνδεση, χωρίς τη διακοπή λειτουργίας. Επιπλέον η γενική διαχείριση ολόκληρου του περιβάλλοντος γίνεται απλούστερη καθώς π.χ. με ελάχιστο χρόνο και κόπο αυξάνεται η ισχύς της CPU, της μνήμης αλλά και του αποθηκευτικού χώρου που μπορεί να χρησιμοποιεί το σύστημά μας. Με φυσικά συστήματα, αυτό θα ήταν εφικτό μόνο με περίπλοκες και χρονοβόρες ενέργειες. Τέλος, καθώς όλο και περισσότερες επιχειρήσεις χρησιμοποιούν λύσεις ανάκτησης σε περίπτωση καταστροφής (disaster recovery plans), με την τεχνική του virtualization μπορεί να μεταφερθεί αυτούσιο το σύστημα σε εφεδρικό μέρος. [23]



Εικόνα 2.9: Τα Συστατικά του Virtualization (Virtualization Layers)

Πηγή: <http://arstechnica.com/business/2011/02/virtualization-in-the-trenches-with-vmware-part-1-basics-and-benefits/>

2.1.5 Πάροχοι Υπολογιστικού Νέφους

Η Amazon, με πολυετή δράση στον τομέα της πληροφορικής, ανέπτυξε τεχνολογίες cloud με την εμπορική ονομασία Amazon Web Services² (AWS). Σύμφωνα με πρόσφατη έρευνα³, κατέχει το 57% του μεριδίου της αγοράς σε μοντέλο public cloud, ενώ είχε αύξηση 6% σε σχέση με το 2015 στην υιοθέτηση των υπηρεσιών της. Έκτος των υπηρεσιών που διαθέτει στο κοινό, προσφέρει υπηρεσίες αποθήκευσης (Amazon Simple Storage Service), πλατφόρμες (Amazon Elastic Compute Cloud) και βάσεις δεδομένων (Amazon Simple DB). Η Amazon έχει υλοποιήσει την υποδομή Νέφους πάνω σε δική της αρχιτεκτονική.

Η IBM⁴ προσφέρει υπηρεσίες Υπολογιστικού Νέφους SaaS, PaaS, IaaS. Για εικονικοποίηση, προσφέρει την υπηρεσία Websphere, καθώς και πλήθος λύσεων σε περίπτωση καταστροφής (backup, data virtualization, virtualized server recovery, email management). Το σημαντικό χαρακτηριστικό της είναι ότι παρέχει τη δυνατότητα υλοποίησης Νέφους σε τρία βασικά μοντέλα ανάπτυξης: ιδιωτικό, δημόσιο και υβριδικό. Η αρχιτεκτονική πάνω στην οποία έχει βασιστεί η υποδομή της IBM είναι το ήδη γνωστό Openstack.

² <https://aws.amazon.com/>

³ Η έρευνα διεξήχθη τον Ιανουάριο του 2016 ανάμεσα σε 1060 IT professionals, με το 42% αυτών να εκπροσωπεί εταιρείες άνω των 1000 εργαζομένων.

⁴ <http://www-935.ibm.com/services/us/en/it-services/cloud-services/index.html>

Αριστείδης Σ. Βυζανιάρης - Ευάγγελος Γ. Φαραντάτος

Η VMware⁵ είναι ο ηγέτης της εικονικοποίησης. Παρέχει την πλατφόρμα εικονοποίησης VMware vSphere η οποία χρησιμοποιείται από το 44% των ερωτηθέντων σε μοντέλο private cloud. Μαζί με το Openstack, το VMware vCloud Suite έχουν δυναμική ανάπτυξη και είναι σταθερά στο 19% της συνολικής υιοθέτησης.

Η Microsoft⁶ διαθέτει υπηρεσίες Νέφους για επιχειρήσεις και προγραμματιστές. Η πλατφόρμα της, Azure Services Platform, επεκτείνεται συνεχώς και πιο συγκεκριμένα το μοντέλο Azure IaaS στο 17% (από 12% την προηγούμενη χρονιά) και το Azure PaaS στο 13% (από 9%). Επιπλέον, διαθέσιμη στο κοινό είναι η υπηρεσία Microsoft SQL Service, η οποία επιτρέπει την αποθήκευση δεδομένων και σχεσιακών ερωτημάτων πάνω σε αυτά. Φυσικά, υπάρχουν και άλλες πολλές διαδεδομένες υπηρεσίες από γνωστές εταιρείες όπως το iCloud της Apple, το Dropbox, το Google Drive, που προορίζονται κυρίως για αποθήκευση και διαμοιρασμό αρχείων.

2.2 Διαδίκτυο των Πραγμάτων (Internet of Things)

Το Διαδίκτυο έχει αναμφίβολα μια βαθιά επίδραση σε ολόκληρη την κοινωνία και τις βιομηχανίες κατά τη διάρκεια των δύο τελευταίων δεκαετιών. Την ίδια στιγμή που το Διαδίκτυο εξελίσσεται, μια άλλη τεχνολογική επανάσταση έχει αναδείξει τη χρήση των αισθητήρων, των ηλεκτρονικών ετικετών και ενεργοποιητών για τον ψηφιακό εντοπισμό (και όχι μόνο), ώστε κάθε χρήστης να έχει τη δυνατότητα να παρατηρεί και να ελέγχει τα αντικείμενα στο φυσικό κόσμο.

Οι συσκευές που είναι συνδεδεμένες στο διαδίκτυο προβλέπεται να είναι 6,4 δισεκατομμύρια μέχρι το τέλος του 2016 (αύξηση 30% σε σχέση με το προηγούμενο έτος), ενώ μέχρι το τέλος του 2020, ο αριθμός αυτός θα φτάσει τα 20,8 δισεκατομμύρια, σύμφωνα με τη διεθνή εταιρεία ερευνών Gartner [24]. Το μεγαλύτερο μέρος από αυτές δεν είναι υπολογιστές με τη γνωστή μορφή smartphone, laptop, tablet, αλλά είναι πράγματα (things), δηλαδή κάποιο άλλο είδος ηλεκτρικής συσκευής με ενσωματωμένους αισθητήρες (π.χ. μια κάμερα ασφαλείας με αισθητήρες κίνησης και δυνατότητα σύνδεσης στο διαδίκτυο). Αυτές οι συσκευές, είναι σε θέση να επικοινωνούν μέσω σταθερών και κινητών δικτύων. Ως αποτέλεσμα, όλες αυτές οι συνδεδεμένες συσκευές που συνθέτουν το Διαδίκτυο των Πραγμάτων, συλλέγουν πληροφορίες από το

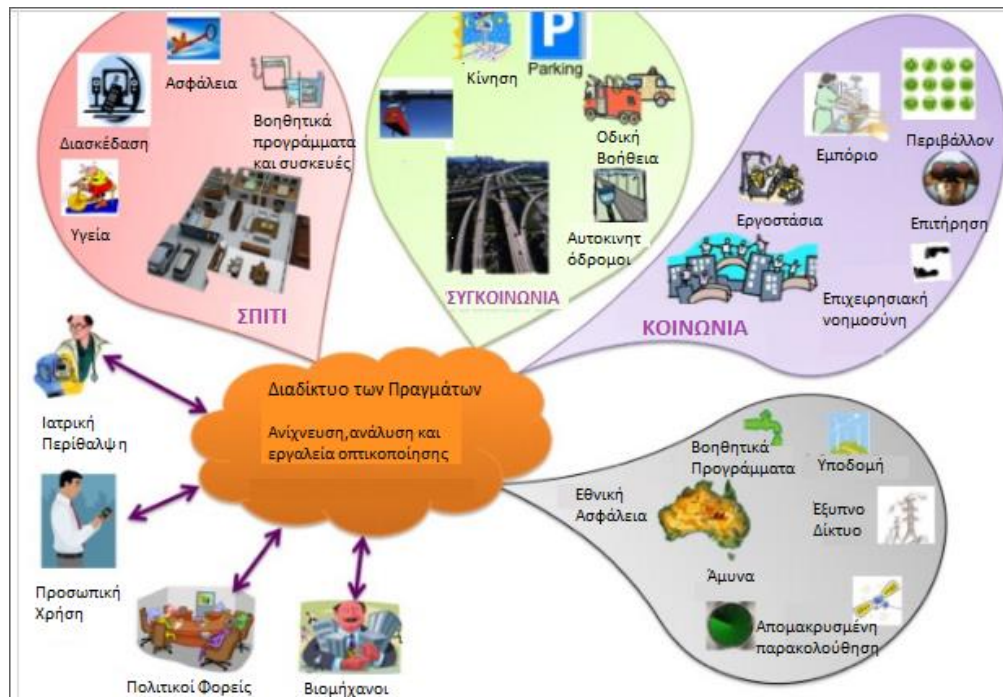
⁵ <http://www.vmware.com/>

⁶ <https://azure.microsoft.com/>

περιβάλλον (φυσικά φαινόμενα, περιβαλλοντολογικές συνθήκες, κ.α.) και την ανθρώπινη δραστηριότητα. Τις πληροφορίες αυτές μπορούν να τις κοινοποιήσουν στον φυσικό κόσμο σε σχεδόν πραγματικό χρόνο, σε ολόκληρο το δίκτυο, με μεγάλο εύρος ζώνης, σε χαμηλό σχετικά κόστος. Στόχος είναι η βελτίωση της ποιότητας της ζωής των ανθρώπων. [24]

Σήμερα, οι περισσότεροι από εμάς έχουμε τουλάχιστον μια συσκευή μόνιμα συνδεδεμένη στο Internet (smartphone ή/και ηλεκτρονικός υπολογιστής) καθώς η ευρυζωνική πρόσβαση έχει γίνει πολύ πιο προσιτή απ' ότι στο παρελθόν. Παράλληλα, η εξέλιξη της τεχνολογίας έχει οδηγήσει στη μείωση του κόστους παραγωγής και του μεγέθους των εξαρτημάτων, ώστε όλο και περισσότερες συσκευές να διαθέτουν ενσωματωμένο Wi-Fi (π.χ. οι σύγχρονες τηλεοράσεις) ή Bluetooth⁷ και προηγμένους αισθητήρες. Η έλευση του πρωτοκόλλου IPv6 (Internet Protocol version 6), επιτρέπει σε όποιο πλήθος συνδεδεμένων συσκευών κι αν κατασκευαστούν μελλοντικά, να έχουν τη δική τους διεύθυνση IP για να είναι δυνατή η επικοινωνία. Οι συσκευές έγιναν μικρότερες σε μέγεθος (υπάρχουν πλέον ακόμα και συσκευές που φοριούνται), ενώ οι επιδόσεις τους βελτιώθηκαν τόσο, ώστε αρκετές συσκευές σήμερα να είναι αποτελεσματικότερες από τους κεντρικούς υπολογιστές του παρελθόντος, οι οποίοι ήταν πολύ μεγαλύτεροι σε όγκο. Αυτοί είναι κάποιοι σημαντικοί λόγοι οι οποίοι συμβάλλουν στην ανάπτυξη και εξέλιξη του Internet of Things. [26]

⁷ <https://www.bluetooth.com/>



Εικόνα 2.10: Internet of Things

Πηγή: <https://www.sciencedirect.com/science/article/pii/S0167739X13000241>

Οι βασικότεροι τομείς που επωφελούνται από τις δυνατότητες που προσφέρει το IoT σε θέματα βελτίωσης ποιότητας, είναι ο κατασκευαστικός / βιομηχανικός τομέας, ο τομέας του περιβάλλοντος, της διαχείρισης ενέργειας, των μέσων μεταφοράς και της υγείας. Επίσης το IoT προσφέρει τη δυνατότητα της αυτοματοποίησης του σπιτιού μας (smarthome). Μπορούμε να χρησιμοποιήσουμε αυτήν την τεχνολογία για να αποφύγουμε φυσικές καταστροφές, να μελετήσουμε την ποιότητα του αέρα/νερού με διάφορους αισθητήρες, να αυτοματοποιήσουμε πλήρως τις κατασκευαστικές διαδικασίες, να βελτιστοποιήσουμε την κατανάλωση ενέργειας σε διάφορες δομές, να ειδοποιείται άμεσα ο γιατρός σε περίπτωση έκτακτης ανάγκης χωρίς την ανθρώπινη παρέμβαση και φυσικά να ελέγχουμε πλήρως όλες τις ηλεκτρικές μας συσκευές, σε πραγματικό χρόνο και απομακρυσμένα.

Τα παραπάνω δημιουργούν έναν τεράστιο όγκο δεδομένων που χρειάζεται διαχείριση. Και σε κάτι τέτοιο θα μπορούσε να συνεισφέρει η χρήση cloud computing, όπως αυτό έχει περιγραφεί σε προηγούμενο κεφάλαιο.

2.2.1 Διαδίκτυο των Πραγμάτων (Internet of Things) και Υπολογιστικό Νέφος (Cloud Computing)

Η χρήση του Διαδικτύου των Πραγμάτων δημιουργεί ένα ανοιχτό, παγκόσμιο δίκτυο που συνδέει τους ανθρώπους, τα δεδομένα και τα πράγματα. Αυτή την χρήση του IoT την αξιοποιεί το σύννεφο για να συνδέσει ευφυή πράγματα, βοηθώντας στην δημιουργία υπηρεσιών. Το υπολογιστικό νέφος δημιουργεί μια παγκόσμια υποδομή για τη δημιουργία νέων υπηρεσιών, που επιτρέπει σε οποιονδήποτε να δημιουργήσει περιεχόμενα και εφαρμογές για κάθε χρήστη παγκοσμίως. Το διαδίκτυο των πραγμάτων συνδέει τα πράγματα σε παγκόσμιο επίπεδο διατηρώντας την ταυτότητά τους σε απευθείας σύνδεση(online). Τα κινητά επιτρέπουν τη σύνδεση με αυτή τη παγκόσμια υποδομή οποτεδήποτε και οπουδήποτε. Το αποτέλεσμα είναι ένα παγκόσμιο προσβάσιμο δίκτυο των πραγμάτων, όπου οι χρήστες και οι καταναλωτές, συμβάλλουν στην δημιουργία και στην αγορά νέων υπηρεσιών. **Error! Reference source not found.**



Εικόνα 2.11: Διαδίκτυο των πραγμάτων και υπολογιστικό νέφος.

Πηγή: <https://www.slideshare.net/PeterNiblett/iot-1906-approaches-for-building-applications-with-the-iiot-cloud>

Ως μέρος αυτής της σύγκλισης μεταξύ του υπολογιστικού νέφους και του διαδικτύου των πραγμάτων, οι εφαρμογές του Internet of Things(IoT) προσφέρονται on-demand μέσω ενός περιβάλλοντος υπολογιστικού νέφους.

Τα ανεπαρκή μέτρα ασφαλείας θα είναι ένα κρίσιμο εμπόδιο για την ανάπτυξη των συστημάτων IoT και την ευρεία υιοθέτηση των εφαρμογών του από τους πελάτες. Οι ήδη υπάρχουσες IT αρχιτεκτονικές ασφαλείας, οι οποίες είναι σχετικά απλές, στο IoT δεν θα είναι επαρκείς. Τα συνδεδεμένα “πράγματα” στο μέλλον θα έχουν περιορισμένους πόρους και δεν θα μπορούν εύκολα ή με οικονομικά αποδοτικό τρόπο να

Αριστέιδης Σ. Βυζανιάρης - Ευάγγελος Γ. Φαραντάτος

αναβαθμιστούν. Προκειμένου να προστατευθούν αυτά τα “πράγματα”, στη διάρκεια του χρόνου, αυξάνεται η σημασία των υπηρεσιών ασφαλείας του υπολογιστικού νέφους με αποδοτική χρήση των πόρων μεταξύ “πραγμάτων-νέφους”.

Με την ανάπτυξη του Διαδικτύου των πραγμάτων, έχουμε συνδέσει την πληροφορική και την επικοινωνία με τα συνδεδεμένα “πράγματα”, συμπεριλαμβανομένης της δυνατότητας ελέγχου των εργασιών τους. Σε τέτοια συστήματα, πολλά τρωτά σημεία της ασφάλειας προέρχονται από τις αλληλεπιδράσεις μεταξύ του κυβερνοχώρου και φυσικού κόσμου. Η προστασία και η ασφάλεια και στους δύο τομείς θα γίνεται ολοένα και πιο απαραίτητη. Σε αντίθεση με τα σημερινά δίκτυα, όπου περιουσιακά στοιχεία βρίσκονται συνήθως πίσω από firewalls και προστατεύονται με έλεγχο πρόσβασης, πολλά πράγματα στην πλατφόρμα του IoT θα λειτουργούν σε απροστάτευτα ή πολύ ευάλωτα περιβάλλοντα (π.χ. οχήματα, αισθητήρες, και ιατρικές συσκευές οι οποίες χρησιμοποιούνται στα σπίτια ή είναι ενσωματωμένες σε ασθενείς). Πολλές εφαρμογές του Διαδικτύου των Πραγμάτων απαιτούν υποστήριξη της κινητικότητας. Τα δεδομένα πρέπει να υποβάλλονται σε επεξεργασία σε "πραγματικό χρόνο", σε μικρό-σύννεφα(Micro-cloud) ή ομίχλη(fog). Τα μικρό-σύννεφα και η υπολογιστική ομίχλη επιτρέπουν σε νέες εφαρμογές και υπηρεσίες να εφαρμόζουν διαφορετικά δεδομένα διαχείρισης και ανάλυσης, επεκτείνοντας με αυτόν τον τρόπο το μοντέλο του Υπολογιστικού νέφους. Το μικρό-σύννεφο και η υπολογιστική ομίχλη πρέπει να έχουν τα ακόλουθα χαρακτηριστικά: **Error! Reference source not found.**

- Ευρεία γεωγραφική κατανομή
- Κινητικότητα
- Πολύ μεγάλο αριθμό κόμβων
- Κυρίαρχο ρόλο της ασύρματης πρόσβασης
- Ισχυρή παρουσία του streaming και εφαρμογές σε πραγματικό χρόνο
- Ετερογένεια

2.2.2 Καθοριστικοί Παράγοντες Ανάπτυξης του Διαδικτύου των Πραγμάτων

Στα πλαίσια των διαρκώς αυξανόμενων απαιτήσεων, η ανάγκη ύπαρξης τεχνολογίας ικανής να ανταποκριθεί, καθίσταται ολοένα και πιο επιτακτική. Μάλιστα, στην εξέλιξη

του IoT, διαδραματίζουν καθοριστικό ρόλο ένα πλήθος κοινωνικών, οικονομικών και περιβαλλοντικών παραγόντων:

- **Ο περιορισμός των φυσικών πόρων.** Ο κόσμος χρειάζεται να καταφέρει ολοένα και περισσότερα, χρησιμοποιώντας λιγότερα, από λιγότερα υλικά, λιγότερη ενέργεια μέχρι λιγότερους φυσικούς πόρους (νερό, φαγητό κ.λπ.). Η αύξηση του πληθυσμού της Γης, έχει αυξήσει και τους περιορισμούς στη χρήση φυσικών πόρων. Η ανάγκη λοιπόν να συμβάλει το IoT στην αύξηση της παραγωγικότητας και στη μείωση των απωλειών σε παγκόσμια κλίμακα είναι αυξανόμενη.
- **Η συνολική οικονομία** προχωρά από την μετά-βιομηχανική εποχή σε μια ψηφιακή οικονομία. Χαρακτηριστικό παράδειγμα είναι η μεταπήδηση από προϊόντοκεντρική οικονομία σε υπηρεσιοκεντρική οικονομία. Καθώς η τεχνολογία αυξάνεται και όλο και περισσότερες διεργασίες αυτοματοποιούνται, το IoT καλό θα ήταν να δημιουργήσει, νέες ευκαιρίες και θέσεις εργασίας.
- **Αλλαγή στα δημογραφικά στοιχεία του κόσμου.** Πολλές χώρες θα χρειαστεί να αντιμετωπίσουν την γήρανση των πληθυσμών, χωρίς να αυξηθούν τα έξοδα. Εδώ, για παράδειγμα, το IoT μπορεί να συμβάλει στην υποστήριξη της καθημερινότητας των ασθενών και των ηλικιωμένων (assisted living) και στη μείωση του κόστους της υγειονομικής περίθαλψης.
- **Κοινωνικοοικονομικές προσδοκίες.** Ο τρόπος ζωής και η άνεση στην καθημερινότητα των ανθρώπων θα αναβαθμιστεί κατακόρυφα με τη χρήση της τεχνολογίας. Οι ίδιες οι πρακτικές που θα χρησιμοποιηθούν στις βιομηχανίες θα εφαρμοστούν και μέσα στις ζωές και τα σπίτια των ανθρώπων.
- **Κλιματικές αλλαγές.** Είναι πλέον επιστημονικά αποδεδειγμένο πως ο αντίκτυπος της ανθρώπινης δραστηριότητας στο περιβάλλον πρέπει να μειωθεί με τη βοήθεια της τεχνολογίας, άρα και του IoT.
- **Ασφάλεια.** Η δημόσια και εθνική ασφάλεια επείγει καθώς η κοινωνία εξελίσσεται αλλά γίνεται και πιο ευάλωτη. Αυτό περιλαμβάνει στη μείωση της εγκληματικότητας αλλά και των θανάτων παγκοσμίως, με τη χρήση διαφορετικών τεχνολογιών.
- **Το φαινόμενο της αστικοποίησης.** Υπάρχει μεγάλη αύξηση των αστικών πληθυσμών και του πλήθους των μεγαλουπόλεων, το οποίο και δημιουργεί

τελείως καινούρια επίπεδα απαιτήσεων από τη δομή μιας πόλης, ώστε να υποστηρίξει το μέγεθος των πληθυσμών. Οι τεχνολογίες IoT αναμένεται να έχουν κύριο ρόλο στη βελτίωση της ζωής των ανθρώπων μέσα στην πόλη (π.χ. στην αναβάθμιση των αστικών συγκοινωνιών). [29]

2.2.3 IoT περιβάλλοντα ανάπτυξης

Πολλές εταιρίες, για να καλύψουν τις ανάγκες που αναλύσαμε παραπάνω, οδηγήθηκαν στην ανάπτυξη πλατφόρμας IoT, δηλαδή ενός συνόλου εργαλείων, που θα παρέχεται στους καταναλωτές ώστε να επεξεργάζονται τα διάφορα δεδομένα που συλλέγουν με τις συσκευές τους.

Σημαντικό βήμα για την ανάπτυξη IoT πλατφόρμας στο Υπολογιστικό Νέφος έκανε η εταιρία IBM, με την ανάπτυξη του IBM Watson IoT⁸ ενώ ανακοίνωσε πως θα διαθέσει \$3 δις τα επόμενα 4 χρόνια για την ανάπτυξη των υπηρεσιών IoT [30]. Ο χρήστης έχει τη δυνατότητα σύνδεσης των συσκευών του, καταγραφής – παρακολούθησης των δεδομένων που συλλέγονται, δημιουργίας γραφημάτων και συντονισμού αυτοματοποιημένων ενεργειών. Η εγγραφή για τη χρήση της υπηρεσίας είναι δωρεάν, και εκτός του βασικού πακέτου χρήσης (περιορισμός χρήσης πόρων/μήνα) είναι δυνατή η αναβάθμιση των υπηρεσιών ανάλογα με τις ανάγκες και το χρηματικό ποσό που μπορεί ο καθένας να διαθέσει.

Η Amazon προσφέρει τη δική της πλατφόρμα AWS IoT⁹. Παρόμοια με τα πρότυπα της IBM, ο χρήστης μπορεί να συνδέσει τις συσκευές του με την πλατφόρμα και να χρησιμοποιήσει τις υπηρεσίες της για να αλληλεπιδράσει με τη συσκευή του, να τις παραμετροποιήσει ώστε να αλληλεπιδρούν μεταξύ τους με αυτοματισμούς και να διαχειριστεί τα δεδομένα που συλλέγονται. Η δημιουργία λογαριασμού είναι δωρεάν και ακολουθείται από επίσης δωρεάν χρήση εντός των ορίων που θέτει η εταιρεία.

Μια χρήσιμη λύση είναι και η πλατφόρμα Thingspeak¹⁰, λόγω της δωρεάν εγγραφής και της φιλικής προς το χρήστη διεπαφής (user interface). Ο χρήστης με απλές παραμετροποιήσεις μπορεί να λαμβάνει και να αποτυπώνει σε πραγματικό χρόνο

⁸ <http://www.ibm.com/internet-of-things/>

⁹ <https://aws.amazon.com/iot/how-it-works/>

¹⁰ <https://thingspeak.com/>

δεδομένα και μετρήσεις από αισθητήρες συσκευών. Επιπλέον έχει τη δυνατότητα να στέλνει αυτοματοποιημένες ειδοποιήσεις, να αποθηκεύει τα δεδομένα και να τα αναλύει με τη βοήθεια του ενσωματωμένου MATLAB(περιβάλλον αριθμητικής υπολογιστικής)¹¹, αλλά είναι και συμβατό με τα διαδεδομένα Arduino Uno και Raspberry Pi (μικρό-υπολογιστής ενιαίων-πινάκων, στο μέγεθος μιας πιστωτικής κάρτας)¹². Η εμπειρία σε περιβάλλοντα IoT δεν κρίνεται απαραίτητη για τους χρήστες της πλατφόρμας, σε αντίθεση με άλλες που ενώ προσφέρουν πληθώρα επιλογών, χρειάζονται μερικές φορές πολύπλοκες παραμετροποιήσεις. Στα παρακάτω κεφάλαια γίνεται επίδειξη της παραμετροποίησης και χρήσης των πλατφορμών IBM και Thingspeak, καθώς και σύγκριση μεταξύ τους.

2.2.4 Αισθητήρες και Διαδίκτυο των Πραγμάτων

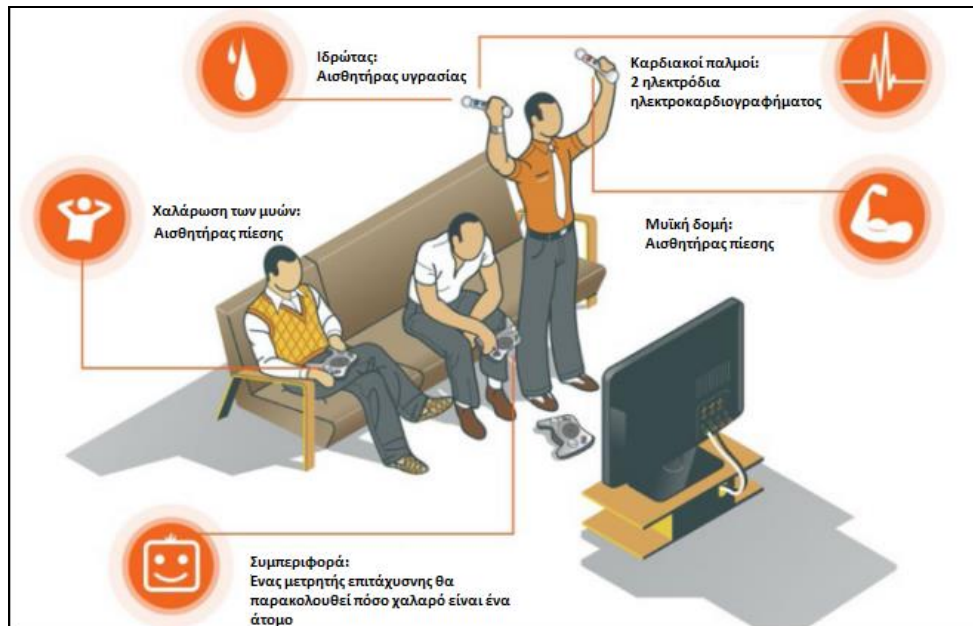
Αναφερόμενοι στο IoT, επισημάνθηκε η παρουσία των αισθητήρων στις διάφορες συσκευές, ώστε να μπορούν να ληφθούν δεδομένα. Οι αισθητήρες είναι απλές ή σύνθετες συσκευές που συνήθως περιλαμβάνουν ένα μετατροπέα που μετατρέπει φυσικές ιδιότητες, όπως είναι η θερμοκρασία, σε ηλεκτρικά σήματα. Σήμερα, υπάρχουν πολλοί διαφορετικοί τύποι αισθητήρων ανάλογα με τις ανάγκες της κάθε περίπτωσης. Η γνωριμία με νέους αισθητήρες αλλά και υπάρχουσες τεχνολογίες έγιναν πια μέρος της καθημερινότητας. Χαρακτηριστικό παράδειγμα είναι η αναγνώριση δαχτυλικού αποτυπώματος, το οποίο μπορεί να ενσωματωθεί στα smartphones. Τα smartphones πλέον περιλαμβάνουν πληθώρα αισθητήρων κίνησης, προσανατολισμού και αναγνώρισης. Σημαντικοί λοιπόν τύποι αισθητήρων είναι:

- Αισθητήρας κίνησης (π.χ. Καταγραφή κίνησης από κάμερες ασφαλείας, Proximity Sensor σε smartphones, Kinect Sensor¹³ σε διάφορα video games, κ.α.)

¹¹ <http://www.mathworks.com/products/matlab/>

¹² <https://www.raspberrypi.org/>

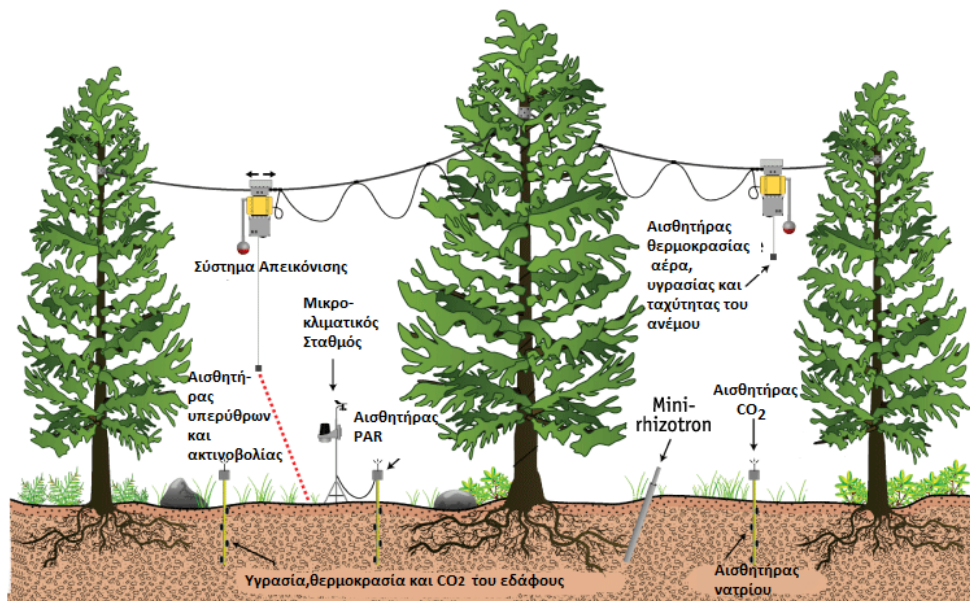
¹³ <https://developer.microsoft.com/en-us/windows/kinect>



Εικόνα 2.12: Παράδειγμα μιας πλατφόρμας ηλεκτρονικών παιχνιδιών που ανιχνεύει την κατάσταση του παίκτη με διάφορους αισθητήρες (κίνησης, πίεσης, κ.α.)

Πηγή: <http://qr.mouser.com/applications/sensor-fusion-iot>

- Περιβαλλοντικός αισθητήρας (π.χ. καταγραφή θερμοκρασίας, υγρασίας, ατμοσφαιρικής πίεσης κ.α.)



Εικόνα 2.13: Μέτρηση περιβαλλοντικών συνθηκών (θερμοκρασία, υγρασία, ταχύτητα του ανέμου, κ.α.)

Πηγή: http://www.environment.ucla.edu/media/images/Fig1_NIMS-lrq.gif

- Ιατρικός αισθητήρας (π.χ. καρδιακός παλμός, ταχύτητα, απόσταση, κ.α.)



Εικόνα 2.14: Παραδείγματα Ιατρικών αισθητήρων.

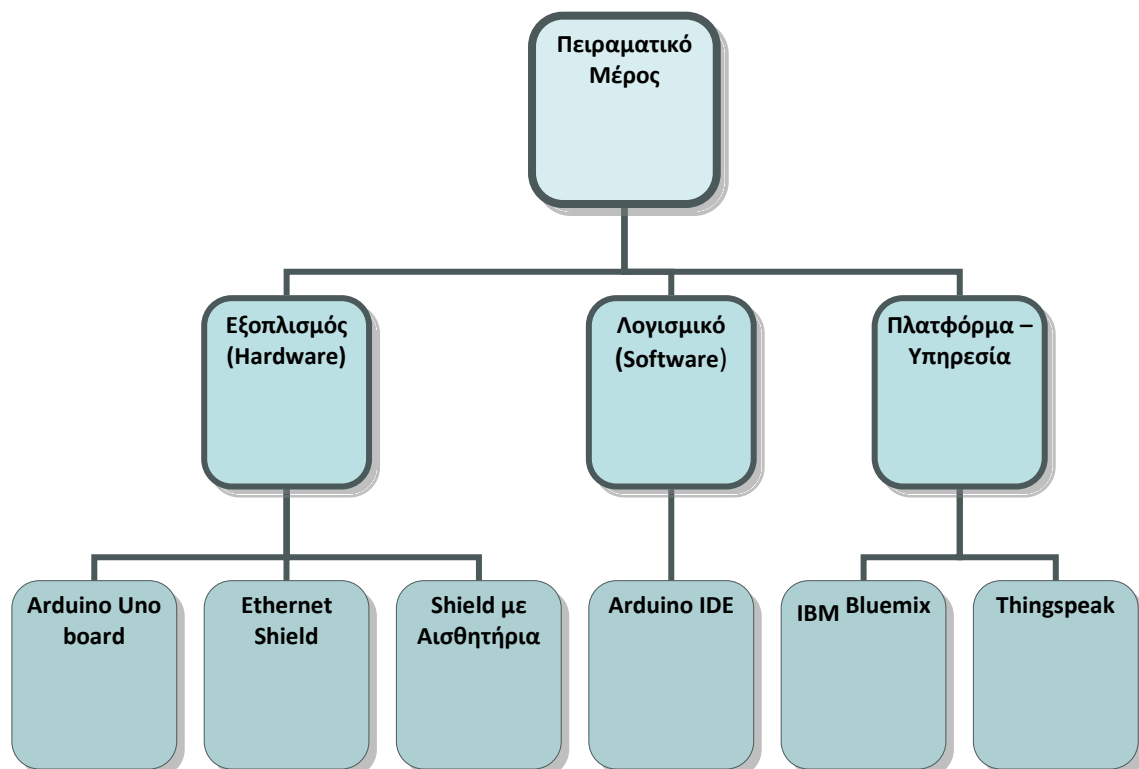
Πηγή: <http://image.slidesharecdn.com/usc-hte-iot-disrupt-healthcare-140322123842-phpapp01/95/disrupting-and-enhancing-healthcare-with-the-internet-of-things-39-638.jpg?cb=1395496096>

3 ΕΡΓΑΛΕΙΑ ΚΑΙ ΜΕΘΟΔΟΙ

Το πειραματικό μέρος της πτυχιακής εργασίας αποτελείται από τρία βασικά κομμάτια:

- Τον **Εξοπλισμό (Hardware)** που χρησιμοποιήθηκε για την κατασκευή του τελικού κυκλώματος.
 - **Arduino Uno board**
 - **Ethernet Shield**
 - **Shield με Αισθητήρια**
- Το **Λογισμικό (Software)** που χρησιμοποιήθηκε για τον προγραμματισμό και την παραγωγή κώδικα.
 - **Arduino IDE**
- Την **Πλατφόρμα – Υπηρεσία** που υποστηρίζει το τελικό κύκλωμα.
 - **IBM Bluemix**
 - **Thingspeak**

Στα παρακάτω υποκεφάλαια θα παρουσιαστεί όχι μόνο ξεχωριστά ο ρόλος τους, αλλά και πώς συνδυάζονται για την επίτευξη του σκοπού της εργασίας.

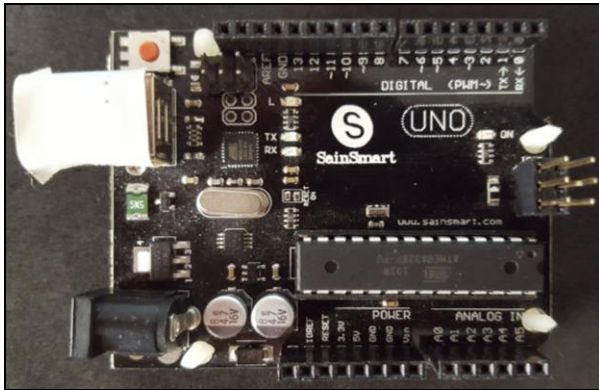


Εικόνα 3.1: Σχηματική Απεικόνιση Πειραματικού Μέρους

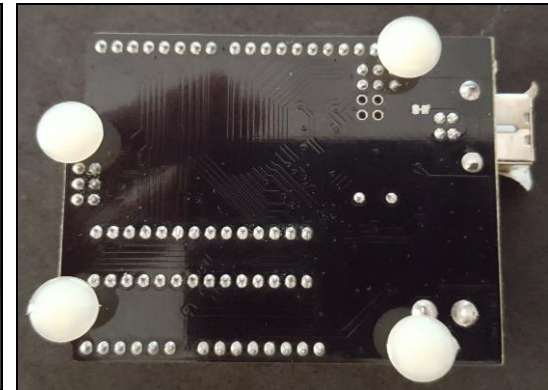
3.1 Εξοπλισμός

3.1.1 Arduino Uno [31], [32]

Βασικό εξάρτημα της κατασκευής που χρησιμοποιήθηκε είναι το Arduino Uno board.



Εικόνα 3.2: Arduino Uno – Άνω Όψη



Εικόνα 3.3: Arduino Uno – Κάτω Όψη

Είναι ένα **ηλεκτρονικό κύκλωμα (board)** που βασίζεται στον μικροελεγκτή ATmega328P. Έχει 14 ψηφιακές εισόδους/εξόδους από τις οποίες 6 μπορούν να προγραμματιστούν ως έξοδοι παλμών μεταβλητού Duty Cycle (PWM-Pulse Width Modulation) αλλά και 6 αναλογικές εισόδους με ανάλυση 10 bits. Διαθέτει ρολόι 16MHz, κουμπί Reset και μπορεί να συνδεθεί με τον Η/Υ μέσω θύρας **USB** ώστε να μπορεί να **τροφοδοτηθεί** με ρεύμα και να **προγραμματιστεί**. Εναλλακτικός τρόπος είναι η **χρήση εξωτερικής τροφοδοσίας AC –to - DC +6 – 20 V** (συνίσταται +7 -12V). Σε κάθε είσοδο ή έξοδο σε λειτουργία 5V περνάει συνεχές ρεύμα 40mA και 50mA σε λειτουργία 3.3V αντίστοιχα. Ο προγραμματισμός επιτυγχάνεται με τη χρήση του λογισμικού **Arduino IDE** το οποίο θα επεξηγηθεί σε ξεχωριστό κεφάλαιο.

Η μνήμη προγράμματος (**Flash ROM**) είναι μεγέθους 32KB από την οποία 0.5 KB είναι ο προεγκατεστημένος bootloader, με τον οποίον είναι δυνατή η μεταφορά (upload) του κώδικα από τον υπολογιστή στον επεξεργαστή μέσω USB χωρίς τη χρήση εξωτερικού hardware . Η επικοινωνία γίνεται με τη χρήση του πρωτοκόλλου STK500. Το υπόλοιπο της μνήμης Flash χρησιμοποιείται για την αποθήκευση των προγραμμάτων που φορτώνονται. Διατίθεται μνήμη **EEPROM** 1KB, για ανάγνωση και εγγραφή δεδομένων (τα οποία και διατηρεί μετά από διακοπή τροφοδοσίας) χρησιμοποιώντας τη βιβλιοθήκη EEPROM. Η μνήμη **SRAM** 2KB η οποία χρησιμοποιείται από το λογισμικό για την αποθήκευση δεδομένων (π.χ. πίνακες, μεταβλητές) κατά τη διάρκεια λειτουργίας, και τα

χάνει όταν η παροχή ρεύματος στο Arduino διακοπεί ή αν γίνει reset (όπως και σε έναν Η/Υ).

Το Arduino διαθέτει 14 θηλυκά pins, αριθμημένα από το 0-13, που μπορούν να οριστούν είτε ως ψηφιακές εισοδοι αλλά και έξοδοι. Ως έξοδοι, αλλάζουν κατάσταση μεταξύ λογικού «1» και «0» βάσει του προγράμματος που έχει φορτωθεί, ώστε να στέλνουν το ανάλογο σήμα στα συνδεδεμένα εξαρτήματα. Ως εισοδοι, δίνεται η δυνατότητα στο πρόγραμμα να δεχθεί πληροφορίες από κάποιο εξάρτημα, διαβάζοντας την κατάσταση του pin. Τα 5 power pins (τροφοδοσίας) έχουν τα εξής χαρακτηριστικά:

- Vin: Η τάση εισόδου του κυκλώματος.
- 5V: Η τάση εξόδου του κυκλώματος.
- 3V3: Τροφοδοσία 3,3V που παράγεται από τον ρυθμιστή του board.
- GND: Τα pin της γείωσης.
- IOREF: Παρέχει την τάση με την οποία λειτουργεί ο μικροελεγκτής.
- pin 0 : Rx (pin 2/RxD of μC)
- pin 1 : Tx (pin 3/TxD of μC)
- pin 2 : General I/O (pin 4/PD2/INT0 of μC)
- pin 3 : General I/O ή PWM (pin 5/PD3/INT1 of μC)
- pin 4 : General I/O (pin 6/PD4/T0 of μC)
- pin 5 : General I/O ή PWM (pin 11/PD5/T1 of μC)
- pin 6 : General I/O ή PWM (pin 12/PD6/AIN0 of μC)
- pin 7 : General I/O (pin 13/PD7/AIN1 of μC)
- pin 8 : General I/O (pin 14/PB0/ICP of μC)
- pin 9 : General I/O ή PWM (pin 15/PB1/OC1 of μC)
- pin 10 : General I/O ή PWM (pin 16/PB2/SS of μC)
- pin 11 : General I/O ή PWM (pin 17/PB3/MOSI of μC)
- pin 12 : General I/O (pin 18/PB4/MISO of μC)
- pin 13 : General I/O (pin 19/PB5/SCK of μC)

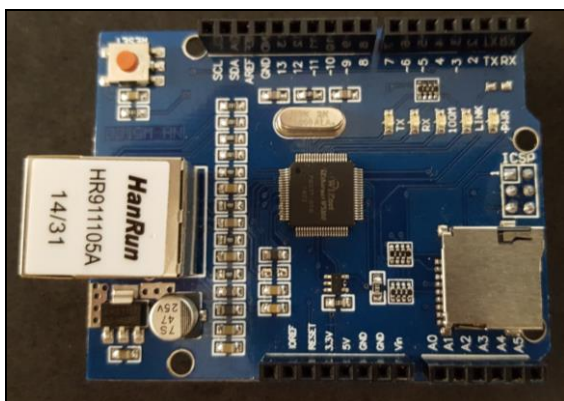
Εκτός από τα παραπάνω, υπάρχει και σειρά 6 pins αναλογικής εισόδου (ANALOG IN, όπως φαίνεται και στην Εικόνα). Με τη χρήση του μετατροπέα Αναλογικού – Ψηφιακού (Analog to Digital Converter – ADC) μπορούν να λειτουργούν ως αναλογική εισοδοι. Στην

τελική κατασκευή χρησιμοποιείται το pin A₀ για την ανάγνωση των δεδομένων του αισθητήρα από το Sensor shield.

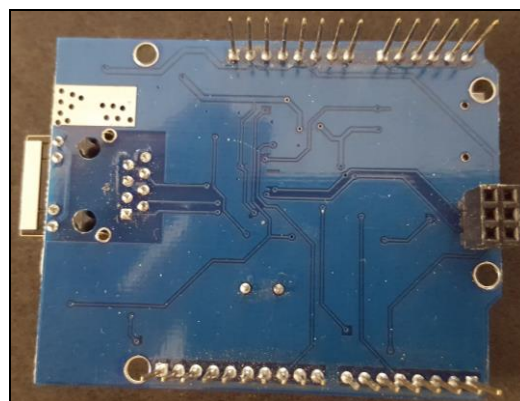
Το Arduino board μπορεί να συνδεθεί με διάφορα **περιφερειακά εξαρτήματα (shields)**, που με την απαραίτητη προϋπόθεση του κατάλληλου προγραμματισμού, του δίνουν πρόσθετες δυνατότητες (π.χ. ενσύρματη / ασύρματη σύνδεση με το Διαδίκτυο, διάφορους τύπους αισθητήρων, camera, κ.α.). Για την παρούσα εργασία, αρκεί η χρήση του **Ethernet shield**, που αποτελεί και την οικονομικότερη λύση σύνδεσης με το Διαδίκτυο, αλλά και ενός **shield αισθητήρων**. Οι δυνατότητες και η χρήση τους θα αναλυθούν στα επόμενα υποκεφάλαια.

3.1.2 Ethernet Shield [32], [33]

Για τη σύνδεση με το Διαδίκτυο απαιτείται η ύπαρξη κάποιας διεπαφής τύπου Ethernet, και καθώς το Arduino δε διαθέτει τέτοιου είδους διεπαφή, θα χρησιμοποιηθεί το Ethernet Shield.



Εικόνα 3.4: Ethernet Shield – Άνω Όψη



Εικόνα 3.5: Ethernet Shield – Κάτω Όψη

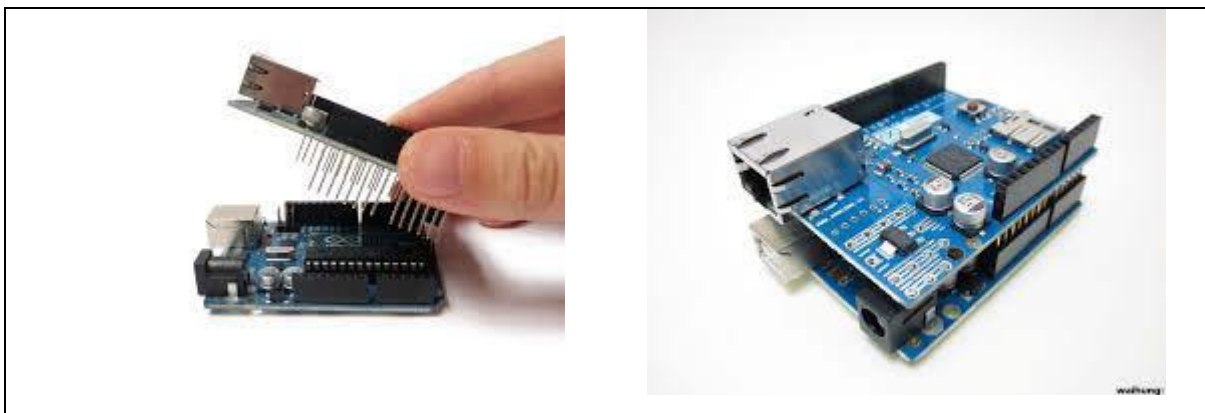
Το Ethernet shield περιέχει το ολοκληρωμένο (ethernet chip) **Wiznet W5100**, ώστε το Arduino να μπορεί να χρησιμοποιήσει τα δικτυακά πρωτόκολλα **TCP** και **UDP**. Η τάση λειτουργίας του shield είναι στα **5V**, η οποία και παρέχεται από το Arduino board. Η υποδοχή του shield δέχεται καλώδιο τύπου **RJ45** – με τη δυνατότητα Power over Ethernet - και διαθέτει ταχύτητες σύνδεσης **10/100 Mb**. Η ανταλλαγή δεδομένων με το Arduino board γίνεται μέσω της θύρας **SPI** και των ψηφιακών pins 10, 11, 12, 14 και 4. Το shield έχει επίσης τη δυνατότητα ταυτόχρονης σύνδεσης με έως 4 sockets. Για τον έλεγχο του shield, ο προγραμματισμός γίνεται χρήση της βιβλιοθήκης **Ethernet library**, για την οποία γίνεται αναλυτικότερη αναφορά σε επόμενο κεφάλαιο. Η ενσωματωμένη υποδοχή κάρτας τύπου **micro-SD** παρέχει τη δυνατότητα αποθήκευσης δεδομένων και για τη διαχείριση της απαιτείται η χρήση της βιβλιοθήκης **SD library** και η επικοινωνία Αριστείδης Σ. Βυζανιάρης - Ευάγγελος Γ. Φαραντάτος

επιτυγχάνεται μέσω του Pin 4. Να σημειωθεί πως λόγω του ότι το Arduino επικοινωνεί με το W5100 και την κάρτα SD μέσω της θύρας SPI (ICSP ετικέτα) μόνο ένα από τα δύο μπορεί να είναι ενεργό. Τέλος, περιέχεται και **ελεγκτής επανεκκίνησης** (reset controller) ο οποίος επανεκκινεί το W5100 αλλά και το Arduino board.

Από τα διάφορα διαγνωστικά **LEDs** που περιέχονται στο shield, θα γίνει αναφορά στα:

- **PWR** - Ένδειξη σωστής τροφοδοσίας ρεύματος)
- **RX** - Αναβοσβήνει κατά την λήψη δεδομένων μέσω της RJ45
- **TX** - Αναβοσβήνει κατά την αποστολή δεδομένων μέσω της RJ45

Το Ethernet shield συνδέεται με το Arduino board όπως φαίνεται στην παρακάτω εικόνα:



Εικόνα 3.6: Σύνδεση Arduino – Ethernet Shield

3.1.3 Shield με Αισθητήρια [34]

Η κατασκευή ολοκληρώνεται με τη χρήση του shield με τα αισθητήρια. Όπως έχουμε αναφέρει και σε προηγούμενο κεφάλαιο, σκοπός είναι να συλλέγουμε δεδομένα από αισθητήρα μέτρησης θερμοκρασίας. Το shield που διατίθεται δεν είναι του εμπορίου, αλλά είναι ένα ηλεκτρονικό κύκλωμα διαφόρων αισθητηρίων που μπορούν να βρεθούν στην αγορά. Το shield διαθέτει τους παρακάτω τύπους αισθητήρων:

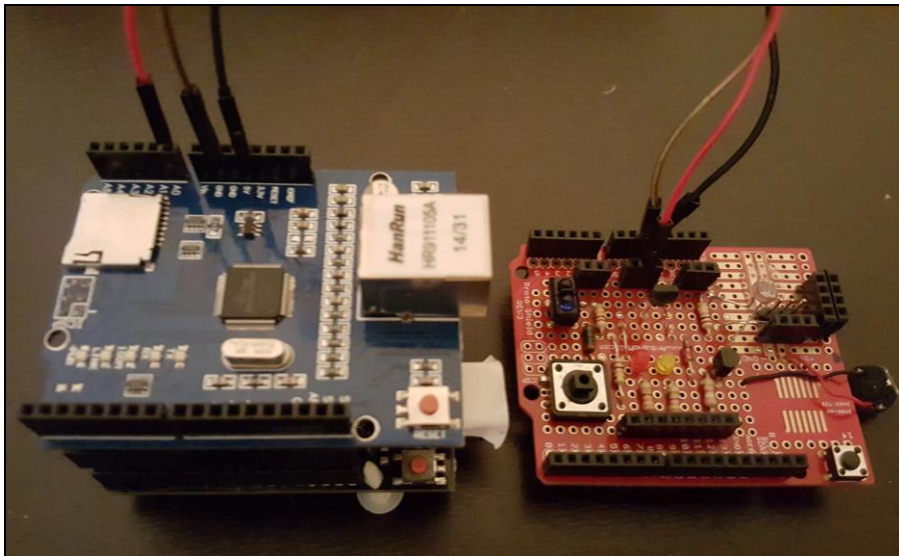
- Αισθητήρας μέτρησης θερμοκρασίας LM35D
- Αισθητήρας φωτεινότητας (φωτοαντίσταση) LDR 5mm¹⁴
- Αισθητήρας εγγύτητας (proximity) TCRT5000¹⁵
- Κινητήρας DC

¹⁴ <http://grobotronics.com/photo-resistor-ldr-5mm.html?sl=el>

¹⁵ <http://pdf1.alldatasheet.com/datasheet-pdf/view/26406/VISHAY/TCRT5000.html>

- **Temperature** = Analog Reading from A₀ / 9.3

Για την αποφυγή του θορύβου που δημιουργεί η γείωση του κυκλώματος δημιουργούμε ένα φίλτρο μέσης τιμής, με το οποίο λαμβάνουμε 20 διαδοχικές μετρήσεις και βρίσκουμε τη μέση τιμή τους. Αυτό θα αποτελέσει κομμάτι του κώδικα με τον οποίο θα προγραμματίσουμε το Arduino και θα αναλυθεί στο κεφάλαιο «Αποτελέσματα – Συμπεράσματα». Τέλος, στην παρακάτω εικόνα προβάλλεται η συνδεσμολογία του shield με τα Arduino board – Ethernet shield, και με αυτόν τον τρόπο θα χρησιμοποιηθεί στην παρούσα εργασία:



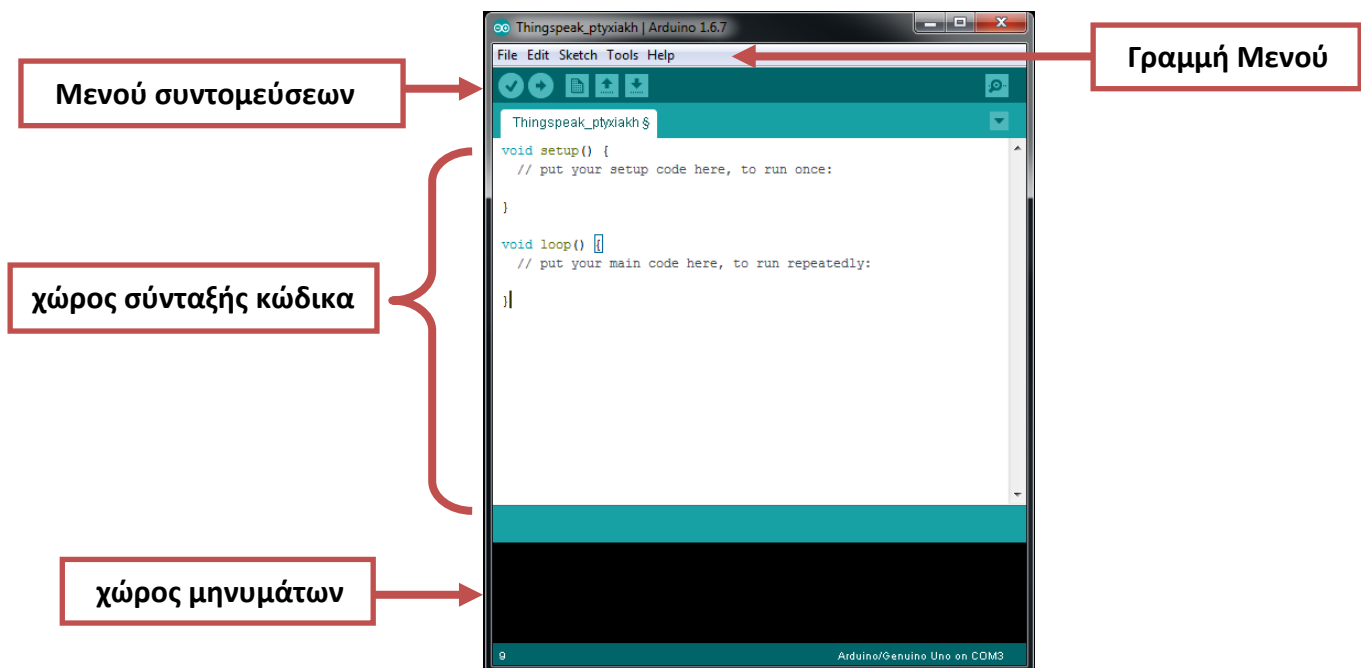
Εικόνα 3.10: Συνδεσμολογία Sensor shield με Arduino board – Ethernet shield

3.2 Λογισμικό (Software)

3.2.1 Arduino IDE [32], [35]

Στο προηγούμενο κεφάλαιο έγινε αναφορά στο πρόγραμμα με το οποίο παράγεται κώδικας και προγραμματίζεται το Arduino. Ο κώδικας ελέγχει την επικοινωνία του board αλλά και τη διαχείριση της επικοινωνίας των περιφερειακών του (όπως Ethernet shield, sensor shield, wi-fi shield κ.α.). Στην παρούσα πτυχιακή εργασία, χρησιμοποιήθηκε το προγραμματιστικό περιβάλλον **Arduino Software IDE v.1.6.7**, το οποίο είναι λογισμικό ανοικτού κώδικα (open-source). Δίνει τη δυνατότητα στο χρήστη να συντάσσει εύκολα και να ελέγχει την ορθότητα του κώδικά του. Επίσης, γίνεται η εγκατάσταση του παραγόμενου προγράμματος στο Arduino μέσω θύρας USB. Τα προγράμματα αυτά ονομάζονται **sketches**. Ο κώδικας που περιέχεται, σε μορφή αναγνώσιμου κειμένου, αποθηκεύεται σε αρχεία με κατάληξη ***.ino**. Το Arduino IDE είναι δομημένο με γλώσσα Αριστείδης Σ. Βυζανιάρης - Ευάγγελος Γ. Φαραντάτος

Java και είναι συμβατό με τα λειτουργικά συστήματα **Microsoft Windows, Apple Mac OS X** (έκδοση 10.7 Lion ή νεώτερη) και **Linux (εκδόσεις 32 & 64 bits)**. Η γλώσσα προγραμματισμού για τα sketches βασίζεται στη γλώσσα **Wiring**, παραλλαγή της C/C++ και έχει τις βασικές δομές της C (βασικές εντολές και συναρτήσεις, σύνταξη, τύποι δεδομένων, τελεστές). Για να μεταγλωττιστεί ο κώδικας, το IDE περιέχει ενσωματωμένο τον λογισμικό μεταγλωττιστή (compiler) **AVR gcc**.



Εικόνα 3.11: Περιβάλλον Arduino IDE

Στην παραπάνω εικόνα, φαίνεται το κύριο περιβάλλον του Arduino IDE, και τα βασικά του μέρη τα οποία είναι:

- **Η Γραμμή Μενού:**
 - File (New, Open, Save, Save As, κ.α.)
 - Edit (Undo, Redo, Cut, Copy κ.α.)
 - Sketch (Verify/Compile, Upload, Include Library κ.α.)
 - Tools (Board, Port, Serial Monitor, κ.α.)
 - Help (Getting Started, Environment κ.α.)
- **Το Μενού συντομεύσεων**
 - Verify – Έλεγχος κώδικα για συντακτικά σφάλματα και μεταγλώττιση (compiling)
 - Upload – Μεταγλώττιση κώδικα και αποστολή του sketch σε συνδεδεμένο board

- New – Δημιουργία νέου αρχείου
- Open – Άνοιγμα υπάρχοντος αρχείου
- Save – Αποθήκευση του περιεχομένου του κώδικα σε *.ino αρχείο
- **Ο χώρος σύνταξης κώδικα**
- **Ο χώρος μηνυμάτων**, όπου ενημερώνεται ο χρήστης για τυχόντα σφάλματα κατά τη μεταγλώττιση, την πρόοδο φόρτωσης του sketch στο board, κ.α.

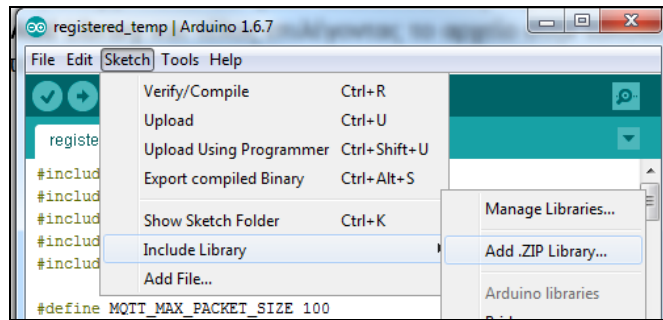
3.2.2 Βιβλιοθήκες [32]

Οι βιβλιοθήκες είναι εργαλεία για την παροχή πρόσθετων δυνατοτήτων στον κώδικα που προγραμματίζεται στο Arduino IDE, για τον έλεγχο του Arduino και των περιφερειακών που μπορεί να χρησιμοποιούνται. Στον κώδικα που χρησιμοποιείται στην εργασία (και που θα αναλυθεί στο κεφάλαιο «Αποτελέσματα – Συμπεράσματα»), θα χρησιμοποιηθούν οι παρακάτω βιβλιοθήκες:

- **Βιβλιοθήκη SPI**, για την επικοινωνία και των έλεγχο του Arduino με τα περιφερειακά του (shields). Είναι ένα πρωτόκολλο που χρησιμοποιούν οι μικροελεγκτές για την ταχεία ανταλλαγή δεδομένων με περιφερειακά σε μικρές αποστάσεις. Η βιβλιοθήκη αυτή είναι ενσωματωμένη στο Arduino IDE. [36]
- **Βιβλιοθήκη Ethernet**, για την επικοινωνία του board με το Ethernet shield. Εξυπηρετεί εισερχόμενες και εξερχόμενες συνδέσεις, με μέγιστο αριθμό τις τέσσερις (4). Παρόμοια με την SPI, παρέχεται από το Arduino IDE αυτόματα. [37]
- **Βιβλιοθήκη MQTT¹⁷**, για τη χρήση του πρωτοκόλλου MQTT, όπως αναφέρθηκε σε προηγούμενο κεφάλαιο. Η συγκεκριμένη βιβλιοθήκη ΔΕΝ περιέχεται στο Arduino IDE και πρέπει να προστεθεί ξεχωριστά.

Το **arduino_1.0.0.zip** αρχείο που θα αποθηκευτεί στον υπολογιστή, πρέπει να φορτωθεί στο Arduino IDE αυτούσιο, χωρίς να γίνει εξαγωγή των αρχείων που περιέχονται (extract/unzip). Αυτό θα γίνει ακολουθώντας τις εντολές **Sketch -> Include Library -> Add .ZIP Library** και τέλος επιλέγοντας το αρχείο στην τοποθεσία που αποθηκεύτηκε στον υπολογιστή.

¹⁷ <https://projects.eclipse.org/projects/technology.paho/downloads>



Εικόνα 3.12: Προσθήκη Βιβλιοθήκης στο Arduino IDE

3.3 Πλατφόρμα – Υπηρεσία

3.3.1 IBM Bluemix

Για την αξιοποίηση των δεδομένων που συλλέγονται από τον αισθητήρα θερμότητας, θα χρησιμοποιηθεί η πλατφόρμα **IBM Bluemix**. Η συγκεκριμένη υπηρεσία Υπολογιστικού Νέφους, υποστηρίζεται από τα πιο δημοφιλή έργα ανοικτού κώδικα (open-source projects) και παρέχει πάνω από 150 υπηρεσίες στο χρήστη. Η χρήση της πλατφόρμας είναι δωρεάν για το δοκιμαστικό διάστημα των 30 ημερών από την εγγραφή, όπου ο χρήστης αξιοποιεί με περιορισμένους πόρους τις διάφορες υπηρεσίες που παρέχει.

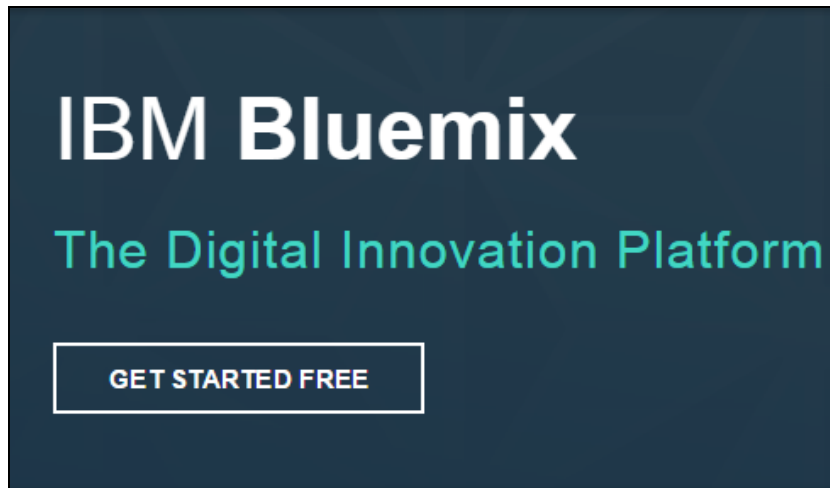
Συγκεκριμένα:

- Πρόσβαση στην πλατφόρμα για τη χρήση και τη δημιουργία εφαρμογών
- 2GB μνήμης
- Χρήση όλων των έτοιμων υπηρεσιών που παρέχονται

Μετά το πέρας των 30 ημερών, μπορεί να συνεχιστεί η χρήση της υπηρεσίας με την προσθήκη πιστωτικής κάρτας, η οποία θα χρεωθεί μόνο αν ο χρήστης ξεπεράσει τα όρια χρήσης (0.5 GB μνήμης / μήνα). Η πρόσβαση στην πλατφόρμα γίνεται μέσω του παρακάτω συνδέσμου:

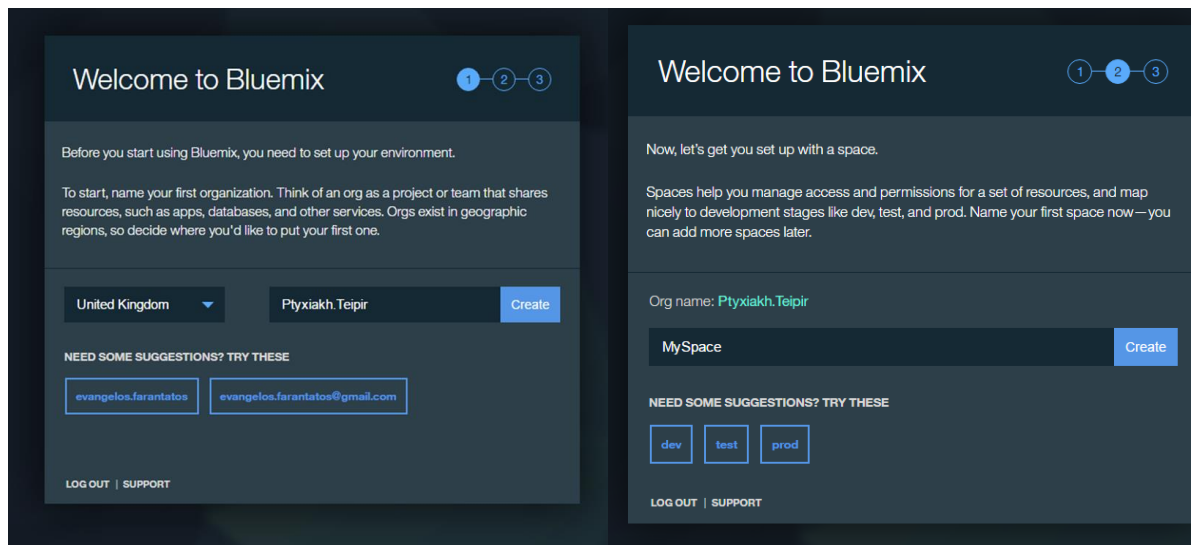
- <https://console.ng.bluemix.net/>

και η εγγραφή μέσω της επιλογής **Get Started Free**, όπως φαίνεται παρακάτω:



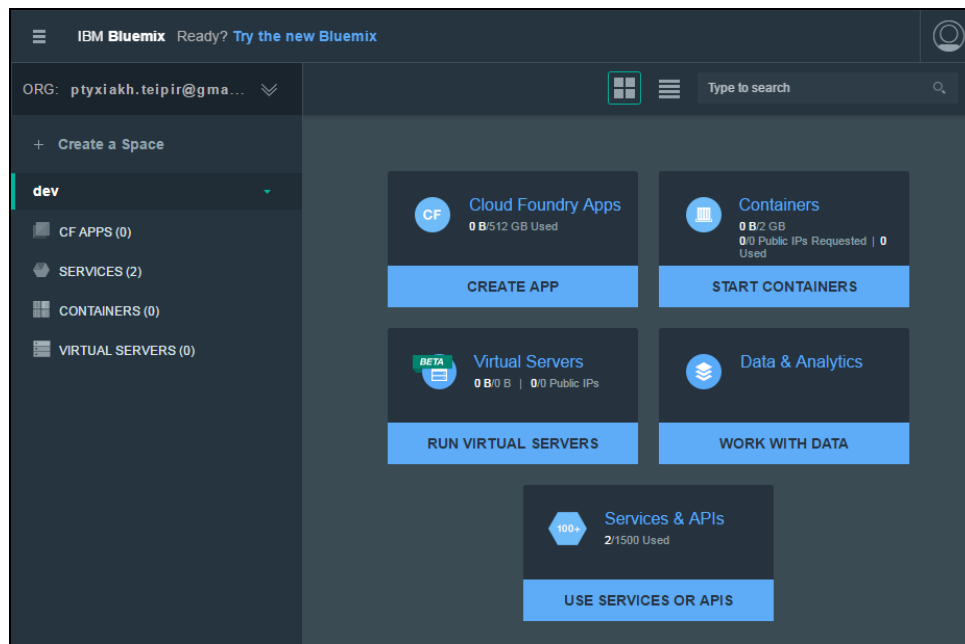
Εικόνα 3.13: Εγγραφή στην πλατφόρμα IBM Bluemix

Στη συνέχεια, ο χρήστης εισάγει τα στοιχεία του στα πεδία που απαιτούνται για τη δημιουργία του λογαριασμού. Κατά την πρώτη σύνδεση (log in), ζητείται να στηθεί το περιβάλλον εργασίας. Αρχικά δημιουργείται ο προσωπικός μας **οργανισμός** (organization), και στη συνέχεια ο **χώρος εργασίας** (space). Η ονοματοδοσία εξαρτάται από το χρήστη, χωρίς κάποιον περιορισμό.



Εικόνα 3.14: Δημιουργία Οργανισμού (Organization) και Χώρου Εργασίας (Space)

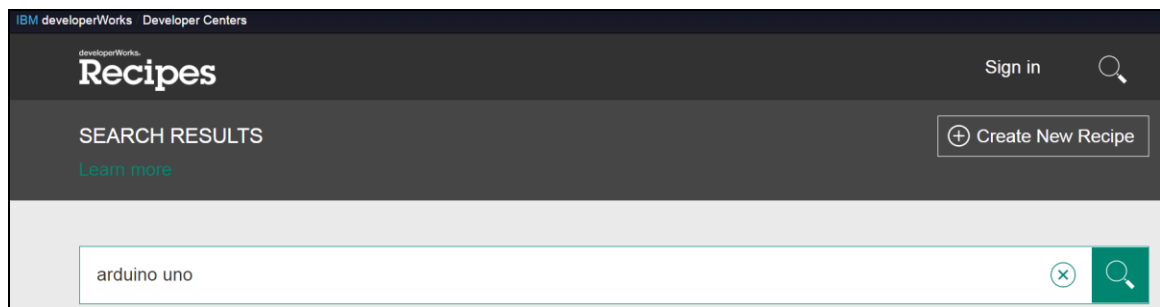
Από αυτό το σημείο και μετά, ο χρήστης μπορεί να επιλέξει τις υπηρεσίες που θα χρησιμοποιήσει για το έργο του, ή και να δημιουργήσει δικές του με τη χρήση των εργαλείων που παρέχονται.



Εικόνα 3.15: Αρχική σελίδα περιβάλλοντος εργασίας

Παράλληλα με την πλατφόρμα, παρέχεται από την IBM ο ιστότοπος **Recipes**¹⁸, ο οποίος περιέχει διάφορα έργα (recipes) που μπορούν να γίνουν με τη χρήση της πλατφόρμας και διάφορου εξοπλισμού, και τους αντίστοιχους οδηγούς (walkthrough) για να υλοποιηθούν.

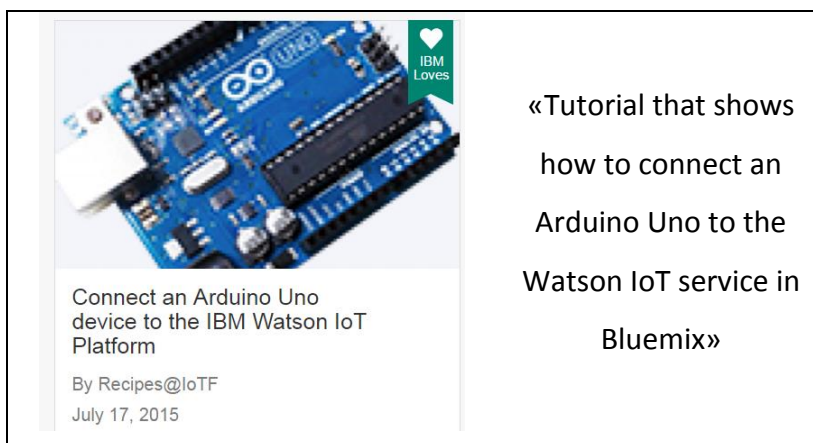
Στην αρχική σελίδα, μπορεί να γίνει αναζήτηση ανάλογα με το αντικείμενο και τον τελικό στόχο, χρησιμοποιώντας λέξεις-κλειδιά (keywords).



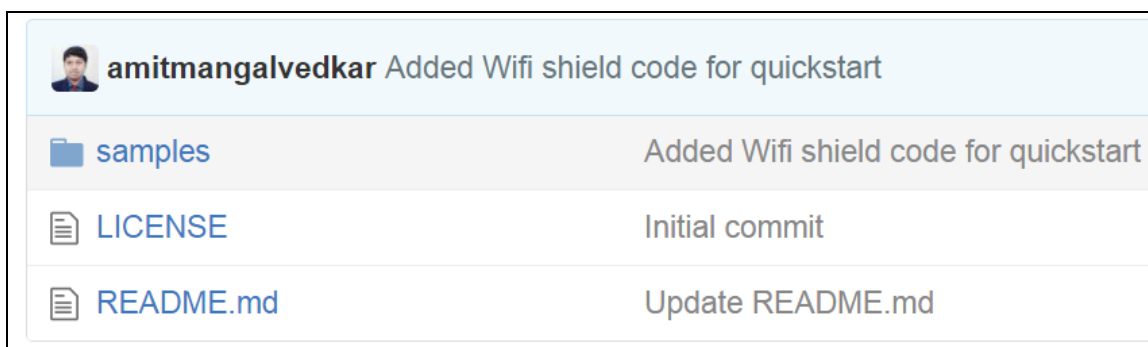
Εικόνα 3.16: Αρχική σελίδα Recipes – Αναζήτηση “Arduino Uno”

Στη λίστα αποτελεσμάτων, επιλέγεται το αποτέλεσμα «Connect an Arduino Uno device to the IBM Watson IoT Platform», μιας και αυτός είναι ο σκοπός της εργασίας.

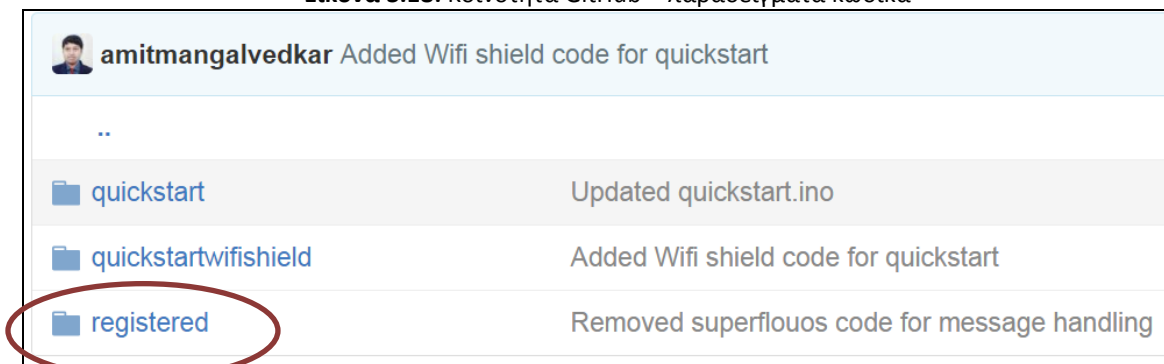
¹⁸ <https://developer.ibm.com/recipes/>



Εικόνα 3.17: Αποτέλεσμα αναζήτησης σχετικό με τον σκοπό της εργασίας. Το συγκεκριμένο recipe, εκτός από την αναφορά στις απαιτήσεις του εξοπλισμού, παρέχει και το sketch που θα χρησιμοποιηθεί για την υλοποίησή του, μέσω του ιστότοπου της κοινότητας **GitHub**¹⁹.



Εικόνα 3.18: Κοινότητα GitHub – παραδείγματα κώδικα



Εικόνα 3.19: Επιλογή Registered έκδοσης κώδικα

Ο κώδικας χρησιμοποιεί τον εσωτερικό αισθητήρα θερμοκρασίας του Arduino board, αλλά μιας και σκοπός είναι η λήψη δεδομένων από το sensor shield, ο κώδικας θα παραμετροποιηθεί κατάλληλα – ανάλυση του κώδικα θα γίνει σε επόμενο κεφάλαιο. Για την κατάλληλη παραμετροποίηση αλλά και υποστήριξη γενικά σχετικά με την χρήση της

¹⁹ <https://github.com/ibm-messaging/iot-arduino>

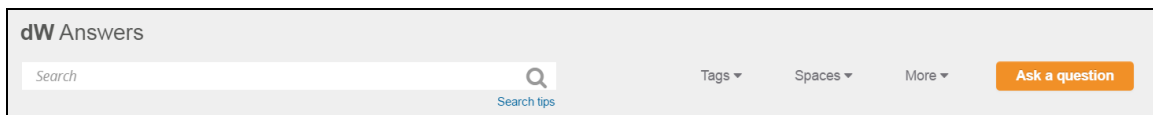
πλατφόρμας και των διαφόρων υπηρεσιών, μπορεί ο χρήστης να ανατρέξει στο φόρουμ «**Developer Works**²⁰» της IBM.

Με δωρεάν εγγραφή, ο χρήστης μπορεί να αναζητήσει επιπρόσθετες πληροφορίες αλλά και να ρωτήσει ειδικούς της IBM σχετικά με κάποιο πρόβλημα ή δυσκολία που αντιμετωπίζει, επιλέγοντας το «**dW Answers**» στο κάτω μέρος της σελίδας:



Εικόνα 3.20: Επιλογή dW Answers

Αρχικά προτείνεται να γίνει αναζήτηση για υπάρχοντα θέματα σχετικά με την ερώτηση ή το πρόβλημα που αντιμετωπίζεται, και αν δεν υπάρχει κάποιο αντίστοιχο αποτέλεσμα να δημιουργηθεί μια νέα ερώτηση επιλέγοντας **Ask a question**.



Εικόνα 3.21: dW Answers – Αρχική σελίδα

3.3.2 Thingspeak

Το **Thingspeak**²¹ είναι μια πλατφόρμα που παρέχει διάφορες υπηρεσίες που απευθύνονται αποκλειστικά στη δημιουργία εφαρμογών IoT. Η πλατφόρμα αυτή προσφέρεται **δωρεάν** για τον χρήστη και απαιτείται μόνο η χρήση ενός απλού λογαριασμού ηλεκτρονικού ταχυδρομείου e-mail για την εγγραφή του σε αυτήν. Προσφέρει τη δυνατότητα συλλογής δεδομένων σε πραγματικό χρόνο, οπτικοποίηση των δεδομένων που συλλέγονται με τη μορφή διαγραμμάτων, την δυνατότητα να δημιουργηθούν προσθήκες (**plugins**) και εφαρμογές για τη συνεργασία με διαδικτυακές υπηρεσίες, κοινωνικά δίκτυα και άλλα APIs. Τα δεδομένα αυτά τα λαμβάνει από διάφορους αισθητήρες ή ενεργοποιητές, όπως **Arduino**, **Raspberry Pi**, **BeagleBone Black** και άλλα.

Το βασικό στοιχείο του ThingSpeak είναι το “**κανάλι Thingspeak**” (Thingspeak Channel). Το κανάλι αυτό αποθηκεύει τα δεδομένα που στέλνουμε στο ThingSpeak και αποτελείται από τα παρακάτω στοιχεία:

²⁰ <http://www.ibm.com/developerworks/>

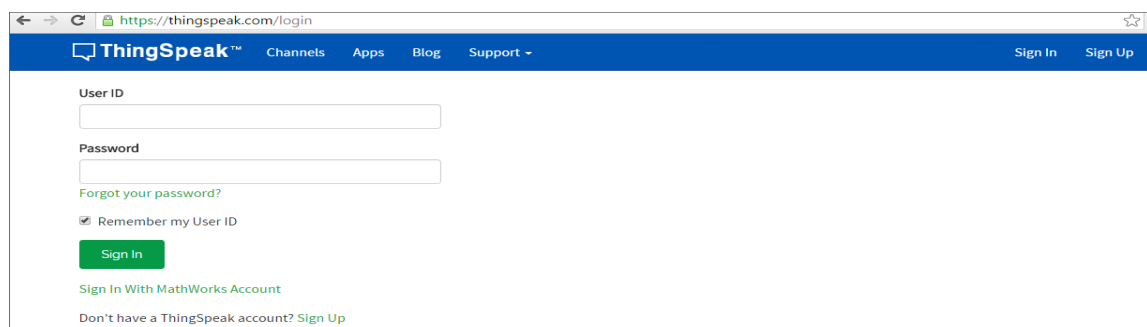
²¹ <https://thingspeak.com/>

- **8 πεδία για την αποθήκευση δεδομένων** οποιουδήποτε τύπου: Αυτά μπορούν να χρησιμοποιηθούν για την αποθήκευση των δεδομένων από έναν αισθητήρα ή από μια ενσωματωμένη συσκευή.
- **3 πεδία τοποθεσίας** (location fields): Μπορούν να χρησιμοποιηθούν για να αποθηκεύσουν το γεωγραφικό πλάτος, μήκος και το υψόμετρο. Αυτά είναι πολύ χρήσιμα για την παρακολούθηση μιας κινητής συσκευής.
- **1 πεδίο κατάστασης** (status field): Ένα σύντομο μήνυμα για να περιγράψει τα δεδομένα που είναι αποθηκευμένα στο κανάλι.

Επίσης το **ThingSpeak support toolbox** επιτρέπει την χρήση της **MATLAB** για την ανάλυση και απεικόνιση δεδομένων που είναι αποθηκευμένα στο thingspeak.com ή σε ιδιωτικές εγκαταστάσεις ThingSpeak. Συγκεκριμένα, μπορούν να εκτελεστούν οι ακόλουθες εργασίες με την MATLAB και το ThingSpeak support toolbox:

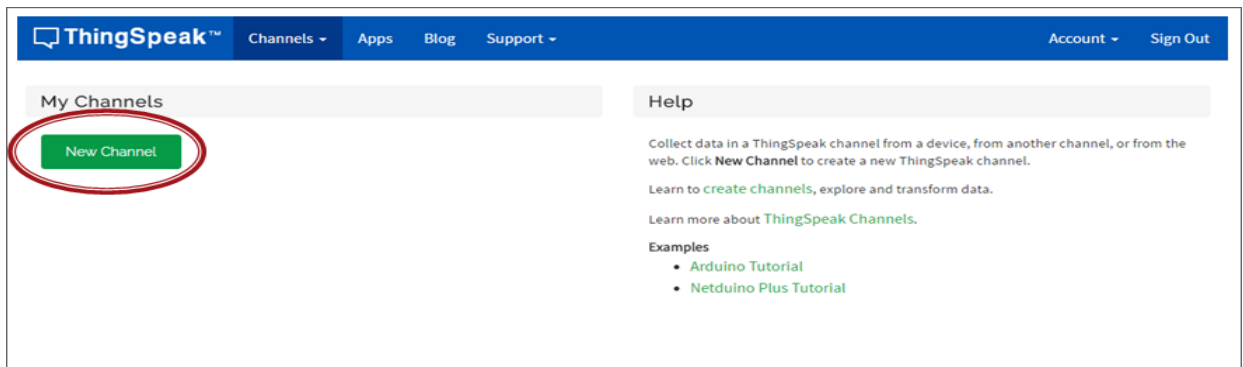
- Να αποκτηθούν πιο πρόσφατα δεδομένα από δημόσια και ιδιωτικά κανάλια του ThingSpeak.
- Να αποκτηθούν ταυτόχρονα δεδομένα από όλα τα πεδία (και τα 8) ενός ThingSpeak καναλιού.
- Να αποκτηθούν τα δεδομένα του καναλιού και του πεδίου κατά τη διάρκεια μιας συγκεκριμένης χρονικής περιόδου.
- Να εγγράφονται τα δεδομένα από την MATLAB σε ένα κανάλι ThingSpeak.
- Να εκτελούνται στατιστικές αναλύσεις των δεδομένων του καναλιού.

Για να χρησιμοποιήσει κάποιος το ThingSpeak, θα πρέπει να **εγγραφεί** και να δημιουργήσει ένα **κανάλι**. Μόλις δημιουργηθεί το κανάλι, μπορούν να σταλούν τα δεδομένα και να αρχίσει η επεξεργασία τους.



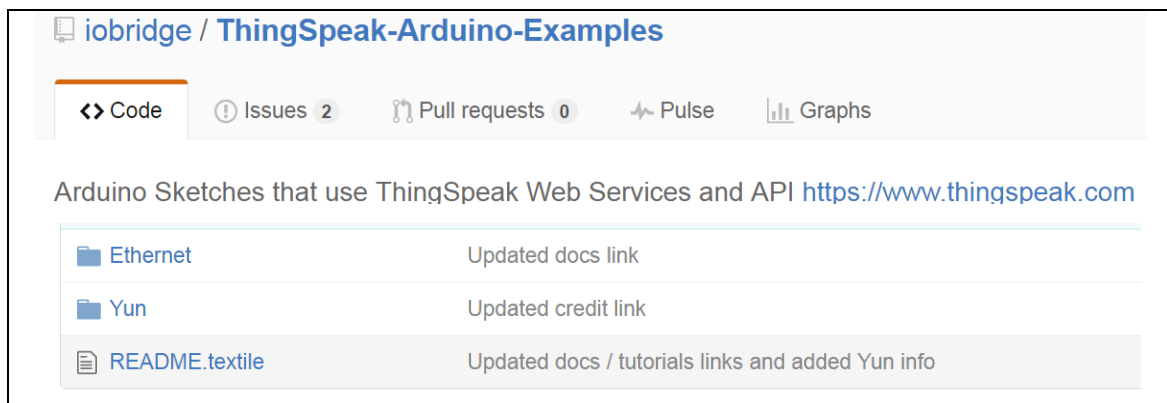
Εικόνα 3.22: Εγγραφή στην πλατφόρμα Thingspeak

Όταν δημιουργείται ένα κανάλι, παράγεται ένα **API κλειδί εγγραφής**(write API key). Το write API key χρησιμοποιείται για την αποστολή δεδομένων στο κανάλι και **το API κλειδί ανάγνωσης** (read API key) χρησιμοποιείται για να διαβάσει τα δεδομένα του καναλιού. Ουσιαστικά το write API key, πρόκειται για ένα **κλειδί ταυτοποίησης** της συσκευής στη πλατφόρμα του Thingspeak, το οποίο είναι και ο μοναδικός τρόπος ταυτοποίησης, καθώς δεν υπάρχει η δυνατότητα χρήσης username-password. Το κλειδί αυτό αποτελείται από αλφαριθμητικούς χαρακτήρες και πρέπει να αποθηκευτεί σε ασφαλές μέρος από τον χρήστη. Σε περίπτωση απώλειας, υπάρχει η δυνατότητα να δημιουργηθεί καινούργιο κλειδί. Στον κώδικα που θα αναλυθεί στα παρακάτω κεφάλαια, θα φανεί πιο συγκεκριμένα ο ρόλος και η χρήση του κλειδιού αυτού.



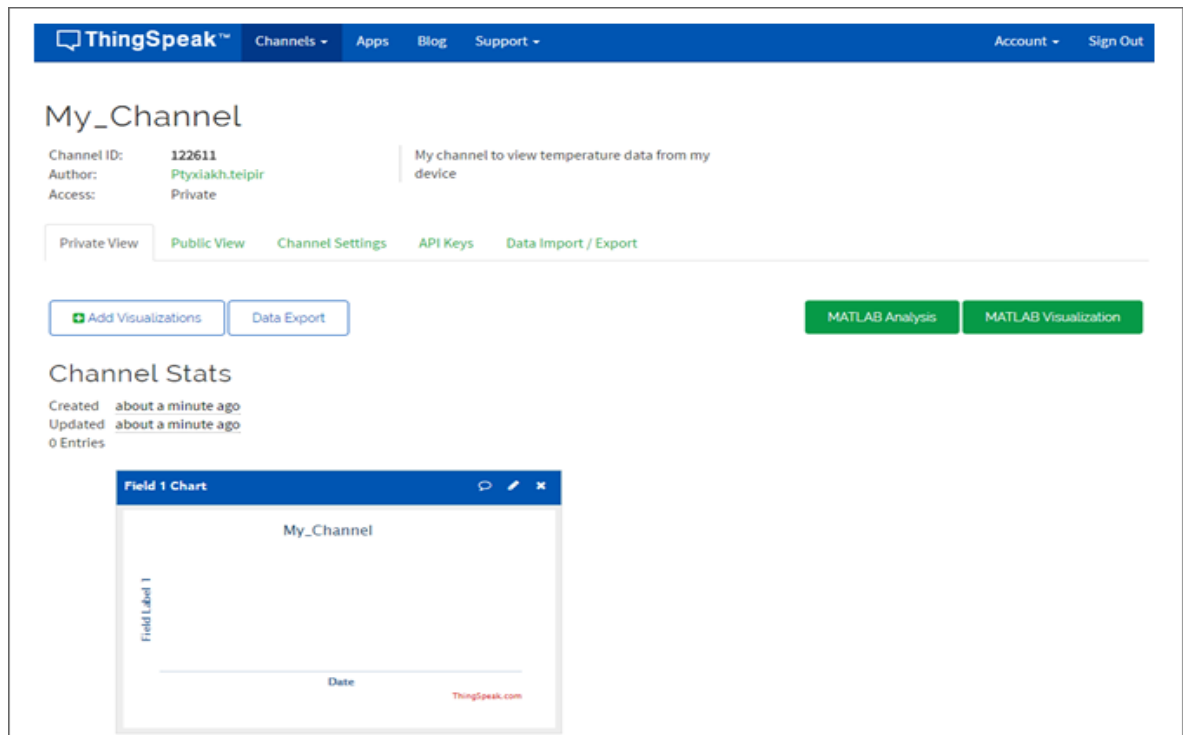
Εικόνα 3.23: Δημιουργία Νέου Καναλιού

Για την επικοινωνία του εξοπλισμού με την πλατφόρμα Thingspeak, έγινε αναζήτηση παραδειγμάτων στον ιστότοπο της κοινότητας **GitHub**²², όπως και παραπάνω στην περίπτωση χρήσης της πλατφόρμας IBM Bluemix. Στον παρακάτω σύνδεσμο, υπάρχει παράδειγμα κώδικα για την ανάγνωση δεδομένων από το A₀ Pin του Arduino board και την αποστολή τους στην πλατφόρμα.



²² <https://github.com/iobridge/ThingSpeak-Arduino-Examples>

Εικόνα 3.24: GitHub –Arduino sketches που χρησιμοποιούν την πλατφόρμα Thingspeak
Όπως και στην περίπτωση της χρήσης IBM Bluemix, ο κώδικας θα παραμετροποιηθεί ανάλογα για την ανάγνωση δεδομένων από το shield και αποστολή τους στην πλατφόρμα όπου και θα γίνει η οπτικοποίηση.



Εικόνα 3.25: Κανάλι Thingspeak

4 ΑΠΟΤΕΛΕΣΜΑΤΑ - ΣΥΜΠΕΡΑΣΜΑΤΑ

Στο προηγούμενο κεφάλαιο, έγινε αναφορά στο πειραματικό μέρος και στις δυνατότητες που προσφέρει στο χρήστη το κάθε εργαλείο που χρησιμοποιήθηκε. Σκοπός, όμως, της εργασίας δεν είναι η χρήση του κάθε εργαλείου αυτόνομα, είτε πρόκειται για εξάρτημα είτε για υπηρεσία. Αντίθετα, η δυνατότητα που έχουν να συνδυαστούν και να λειτουργήσουν ως σύνολο, είναι το αντικείμενο έρευνας της παρούσας εργασίας. Για να επιτευχθεί η επικοινωνία μεταξύ των εργαλείων, χρειάζονται παραμετροποιήσεις στο κάθε εξάρτημα, με τη χρήση της γνώσης που προσφέρει το Διαδίκτυο. Στο παρόν κεφάλαιο, θα γίνει επεξήγηση όλων των απαιτούμενων παραμετροποιήσεων που συνετέλεσαν στο να απεικονίζονται γραφικά τα δεδομένα του αισθητήρα θερμοκρασίας, μέσω της πλατφόρμας IBM Bluemix και Thingspeak. Τέλος, θα παρουσιαστούν τα αποτελέσματα της χρήσης της κάθε πλατφόρμας, σε συνδυασμό με τα υπόλοιπα εργαλεία, και αναφορά των συμπερασμάτων στα οποία οδηγείται ο χρήστης.

4.1 Οπτικοποίηση Δεδομένων με χρήση της πλατφόρμας IBM Bluemix

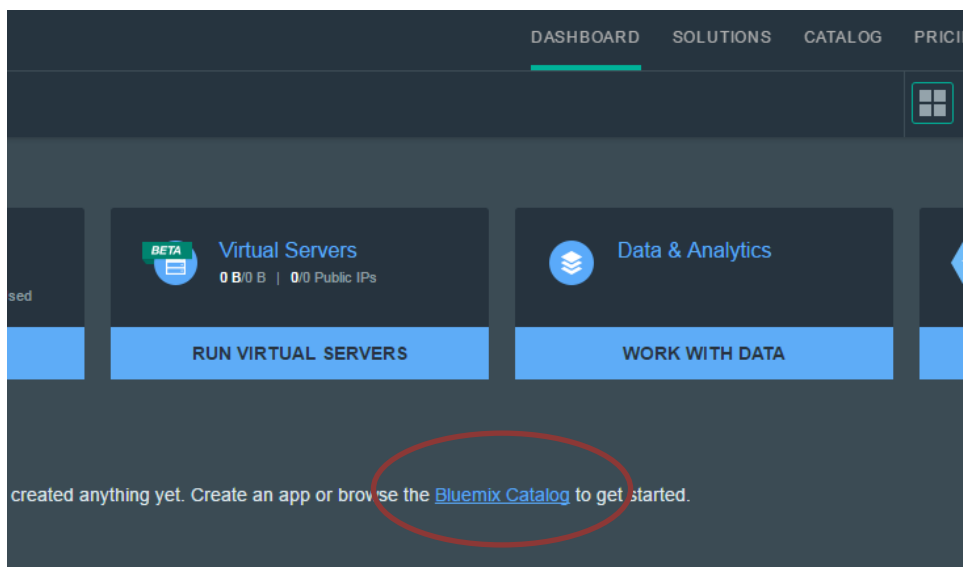
Όπως προαναφέρθηκε, η πλατφόρμα IBM Bluemix προσφέρει ένα πλήθος εφαρμογών (applications), διαφόρων κατηγοριών, τις οποίες μπορεί να χρησιμοποιήσει δωρεάν ο χρήστης (εντός των ορίων που ορίζει ο πάροχος - IBM). Για την οπτικοποίηση των δεδομένων του αισθητήρα, χρειάζεται αρχικά να συνδεθεί η συσκευή με την πλατφόρμα, και στη συνέχεια να γίνει γραφική αναπαράστασή τους. Αυτό επιτυγχάνεται σε δύο μέρη:

- Χρήση εφαρμογής για την επικοινωνία της συσκευής και την αποστολή των δεδομένων – ανάγνωσή τους από την πλατφόρμα σαν αξίες (values)
- Χρήση εφαρμογής για την γραφική αναπαράσταση των δεδομένων που μεταδίδονται

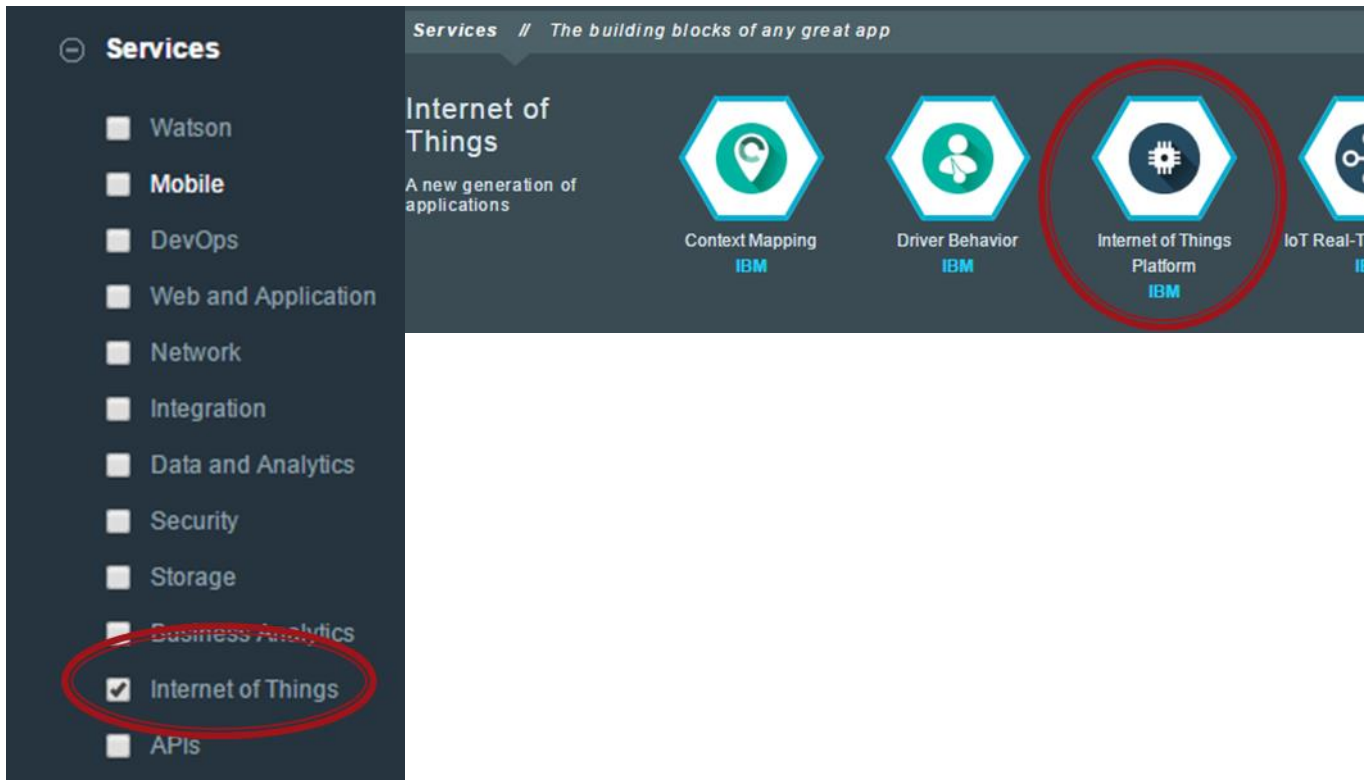
Από την πλευρά του εξοπλισμού, είναι αρκετή η συνδεσμολογία όπως παρουσιάστηκε στο κεφάλαιο 3. Θα χρειαστεί επιπλέον να φορτωθεί στο Arduino board το κατάλληλο sketch, για το οποίο επίσης θα επεξηγηθούν όλες οι απαραίτητες τροποποιήσεις, ανά τμήμα κώδικα.

4.1.1 Σύνδεση της συσκευής με την πλατφόρμα IBM Bluemix και αποστολή δεδομένων

Η εφαρμογή της πλατφόρμας που παρέχει τη δυνατότητα να επικοινωνήσει με τη συσκευή και να λάβει τα δεδομένα του αισθητήρα είναι η «**Internet of Things Platform**». Αφού ο χρήστης έχει εγγραφεί στην πλατφόρμα, σύμφωνα με το κεφάλαιο 3, και έχει συνδεθεί με τα χαρακτηριστικά του (credentials), μπορεί να την αναζητήσει στον «**Bluemix Catalog**», όπως φαίνεται παρακάτω:

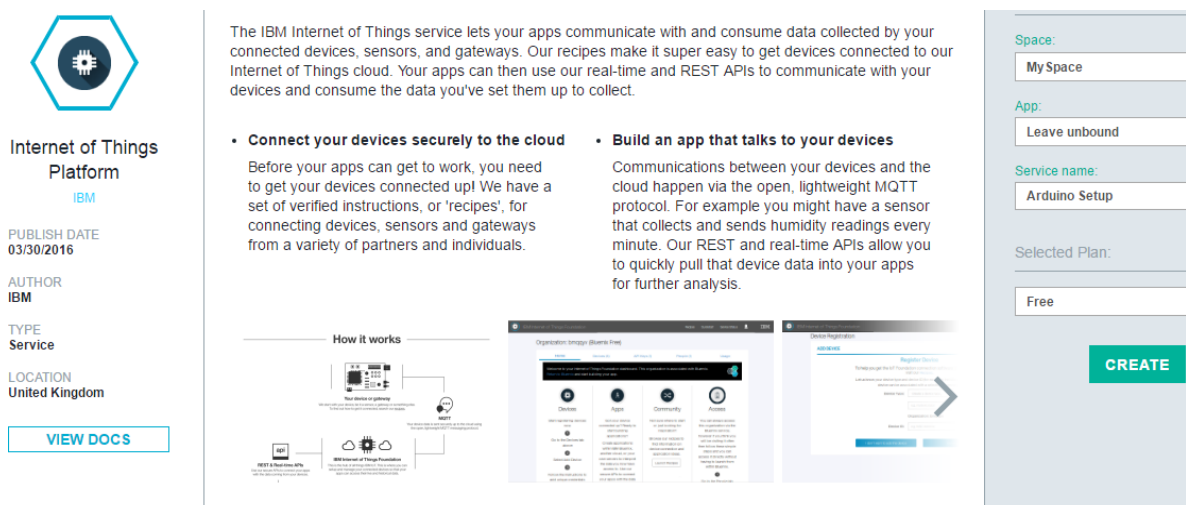


Εικόνα 4.1: Αναζήτηση εφαρμογών στον κατάλογο Bluemix (Bluemix Catalog)
Στην αριστερή πλευρά της οθόνης, υπάρχει μια βοηθητική λίστα αναζήτησης, βάσει κατηγορίας. Εκεί, επιλέγεται η κατηγορία «**Internet of Things**». Από τις διαθέσιμες εφαρμογές, σχετικές με την κατηγορία «Internet of Things», θα επιλεγθεί η «**Internet of Things Platform**», όπως αναφέρθηκε παραπάνω:



Εικόνα 4.2: Λίστα κατηγοριών εφαρμογών

Επιλέγοντας την εφαρμογή, εμφανίζεται η σελίδα της εικόνας 4.3. Στη δεξιά πλευρά, ο χρήστης ορίζει το «χώρο» που θα φιλοξενήσει την εφαρμογή, δίνει όνομα για λόγους εξατομίκευσης και διαλέγει το πλάνο χρήσης. Το δωρεάν πλάνο (free plan) είναι αρκετό για τις ανάγκες τις εργασίας. Στο κάτω μέρος της σελίδας, εξηγείται τι περιέχει το δωρεάν πλάνο, καθώς και οι άλλες διαθέσιμες επιλογές – δωρεάν πλάνο ισούται με έως και 20 καταχωρημένες συσκευές και μέγιστο 100MB για ανταλλαγή δεδομένων.



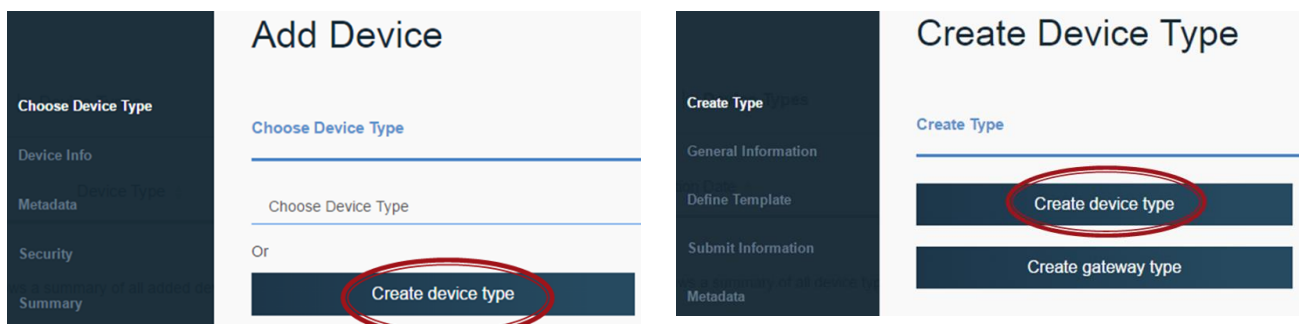
Εικόνα 4.3: Σελίδα εφαρμογής

Μετά τις όποιες τροποποιήσεις, η εφαρμογή προστίθεται στο «χώρο» του χρήστη επιλέγοντας «**Create**». Η σύνδεση της συσκευής, ξεκινά με την εκκίνηση του πίνακα οργάνων (dashboard), από την αρχική σελίδα της εφαρμογής που δημιουργήθηκε. Στη συνέχεια, στην αριστερή πλευρά υπάρχει η κατηγορία «**Devices**», όπου και δίνεται η δυνατότητα να προστεθεί μια νέα συσκευή (δεξί μέρος οθόνης +Add Device)



Εικόνα 4.4: Προσθήκη Νέας Συσκευής

Με το πάτημα της επιλογής, εμφανίζεται ένα νέο παράθυρο στην οθόνη. Η προσθήκη της νέας συσκευής ξεκινά με τη δημιουργία ενός τύπου συσκευής (device type). Ο χρήστης δύναται να ορίσει τον τύπο της συσκευής που θα δημιουργήσει, πληκτρολογώντας ένα όνομα και μια περιγραφή, όπως φαίνεται στις παρακάτω εικόνες. Για να προχωρήσει η διαδικασία, στο τέλος κάθε βήματος υπάρχει στο κάτω δεξιά μέρος της οθόνης η επιλογή «Next» Το όνομα του τύπου της συσκευής που θα χρησιμοποιηθεί εδώ, όπως αναφέρει και η εικόνα, θα χρησιμοποιηθεί για να ταυτοποιηθεί η συσκευή. Αυτό θα γίνει στον κώδικα που θα φορτωθεί στο Arduino, και θα εξηγηθεί παρακάτω.



Εικόνα 4.5: Παραμετροποίηση Συσκευής

General Information ⓘ

Name

The device type name is used to identify the device type uniquely, using a restricted set of characters to make it suitable for API use.

Description

The device type description can be used for a more descriptive way of identifying the device type.

Back **Next**

Εικόνα 4.6: Παραμετροποίηση Συσκευής

Το επόμενο παράθυρο, είναι προαιρετικό γιατί εδώ ο χρήστης μπορεί απλά να εμπλουτίσει τις πληροφορίες της συσκευής του (κατασκευαστής, έκδοση λογισμικού, σειριακός αριθμός κ.α.). Όπως και παραπάνω, η διαδικασία συνεχίζεται επιλέγοντας «Next», όπως και στα επόμενα παράθυρα. Σε οποιαδήποτε περίπτωση, ο χρήστης μπορεί να ανατρέξει για να διορθώσει ή/και να προσθέσει τις παραμέτρους.

Create Device Type ⓘ

Define Template

Use the options below to select attributes for the device type. All of these attributes are optional. They will be used as a template for new devices that are assigned this device type. Attributes you do not define may still be edited individually on devices that are assigned this device type.

<input type="checkbox"/> Serial Number ...	<input type="checkbox"/> Description ...
<input type="checkbox"/> Manufacturer ...	<input type="checkbox"/> Firmware Version ...
<input type="checkbox"/> Model ...	<input type="checkbox"/> Hardware Version ...
<input type="checkbox"/> Class ...	<input type="checkbox"/> Descriptive Location ...

Create Device Type

Submit Information

You did not select any fields in the Define Template step. It is not mandatory to do so, but if you wish to define attributes that will act as a template for new devices that are assigned this device type, you may go back to that step and revise your decision - the fields you select will then appear here.

Create Device Type

Metadata (optional)

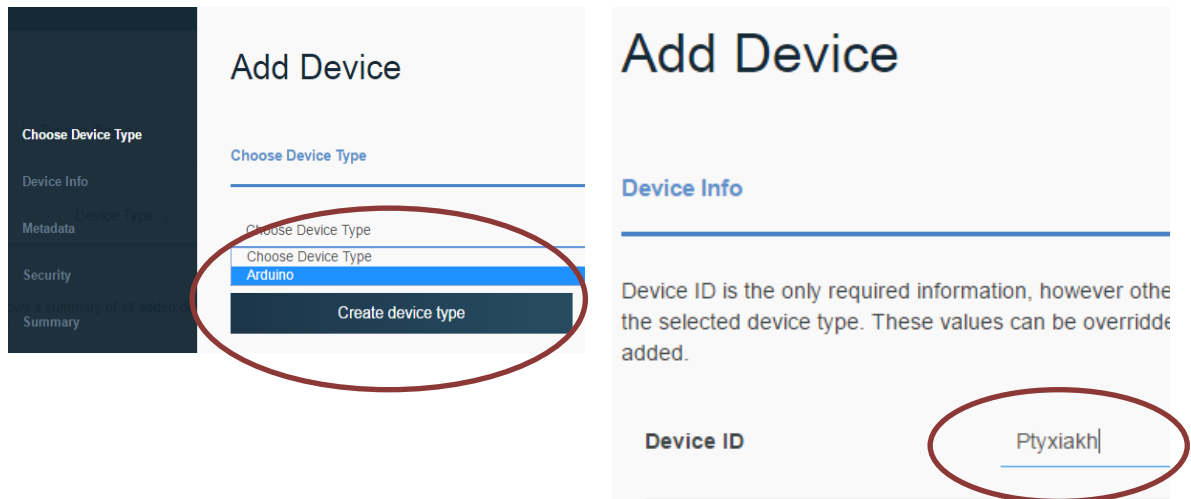
Metadata must be added as JSON; plain text cannot be used.

1 |

Εικόνα 4.7: Παραμετροποίηση Συσκευής – Προαιρετικά Βήματα

Εφόσον ολοκληρώθηκαν τα παραπάνω βήματα, ο τύπος συσκευής που δημιουργήθηκε είναι πλέον διαθέσιμος προς επιλογή, όπως διακρίνεται παρακάτω. Στη συνέχεια, θα Αριστείδης Σ. Βυζανιάρης - Ευάγγελος Γ. Φαραντάτος

πρέπει να ταυτοποιηθεί η συσκευή από τον χρήστη, συμπληρώνοντας το πεδίο Device ID. Το επόμενο παράθυρο είναι επίσης προαιρετικό, όπως επίσης και στη δημιουργία τύπου συσκευής, οπότε η συνέχεια είναι στο παράθυρο με την ένδειξη Security. Αυτό είναι σημαντικό βήμα γιατί θα οριστεί ένας κωδικός ταυτοποίησης (authentication token), ο οποίος είναι μοναδικός για κάθε συσκευή στην πλατφόρμα της IBM και ΔΕΝ είναι ανακτήσιμος.



Εικόνα 4.8: Προσθήκη Νέας Συσκευής

Υπάρχουν οι ακόλουθες επιλογές:

- Είτε ο χρήστης να ορίσει έναν προσωπικό κωδικό (μεταξύ 8-36 χαρακτήρων, που να περιέχει κεφαλαία και πεζά, αριθμητικούς χαρακτήρες και σύμβολα)
- Είτε να δημιουργηθεί αυτόματα ο κωδικός από το σύστημα

Σε κάθε περίπτωση, ο χρήστης έχει την ευθύνη να φυλάξει τον κωδικό και να τον παρέχει όποτε ζητηθεί (θα χρειαστεί να τον περιέχει ο κώδικας, όπως θα εξηγηθεί αργότερα).

Add Device

Security

You have two options:

Auto-generated authentication token

Allow the service to generate an authentication token for you. The token will be 18 characters long and will contain a mix of alphanumeric characters and symbols. The token will be returned to you at the end of the registration process.

Self-provided authentication token

Provide your own authentication token for this device. The token must be between 8 and 36 characters long, and should contain a mix of lower and upper case letters, numbers, and symbols (hyphen, underscore, and period are permitted). The token should be free of repetition, dictionary words, user names, and other predefined sequences.

Provide a token (optional)

Authentication tokens are encrypted before we store them.

We are not able to recover lost authentication tokens. Ensure you make a note of the authentication token after clicking Add.

Εικόνα 4.9: Ασφάλεια - Authentication Token

Ολοκληρώνοντας την παραπάνω διαδικασία, το τελικό παράθυρο παρουσιάζει μια περίληψη των πληροφοριών που ορίσαμε, αλλά και κάποιες επιπλέον τις οποίες πρέπει να αποθηκεύσει ο χρήστης για `να τις χρησιμοποιήσει κατάλληλα (σε περίπτωση που παραπάνω επιλέχθηκε η αυτόματη έκδοση κωδικού, σε αυτό το σημείο εμφανίζεται στο χρήστη). Συγκεκριμένα πρέπει να γνωρίζει:

- Organization ID
- Device Type
- Device ID
- Authentication Token

Add Device

Summary

Please check that all submitted information for this device is correct before adding this device.

Device Type	Arduino
Device ID	Ptyxiakh
Serial Number	-
Manufacturer	-
Model	-
Class	-
Description	-
Firmware Version	-
Hardware Version	-
Descriptive Location	-
Authentication Token	To be generated
Metadata	

Εικόνα 4.10: Περίληψη Ενεργειών

Η συσκευή έχει προστεθεί και είναι ορατή όπως δείχνει η παρακάτω εικόνα:

Devices

Browse | Diagnose | Action | Device Types

<input type="checkbox"/>	Device ID ↓	Device Type ↓	Class ID ↓	Date Added
Results 1-1 of 1				
<input type="checkbox"/>	▲ Ptyxiakh	Arduino	Device	May 20, 2016 12:21:37 AM

Εικόνα 4.11: Συσκευή Ptyxiakh όπως απεικονίζεται στην πλατφόρμα

The screenshot shows a web interface for a device named 'Ptyxiakh'. It features two main sections: 'Your Device Credentials' and 'Connection Information'. Both sections contain key-value pairs of device data. The 'Your Device Credentials' section includes Organization ID, Device Type, Device ID, Authentication Method, and Authentication Token. The 'Connection Information' section includes Device ID, Device Type, Date Added, Added By, and Connection State. A 'Refresh' button is located in the top right corner of the page.

Your Device Credentials	
Organization ID	9lww71
Device Type	Arduino
Device ID	Ptyxiakh
Authentication Method	token
Authentication Token	ScT1Q7tcCR)qi+xdD!

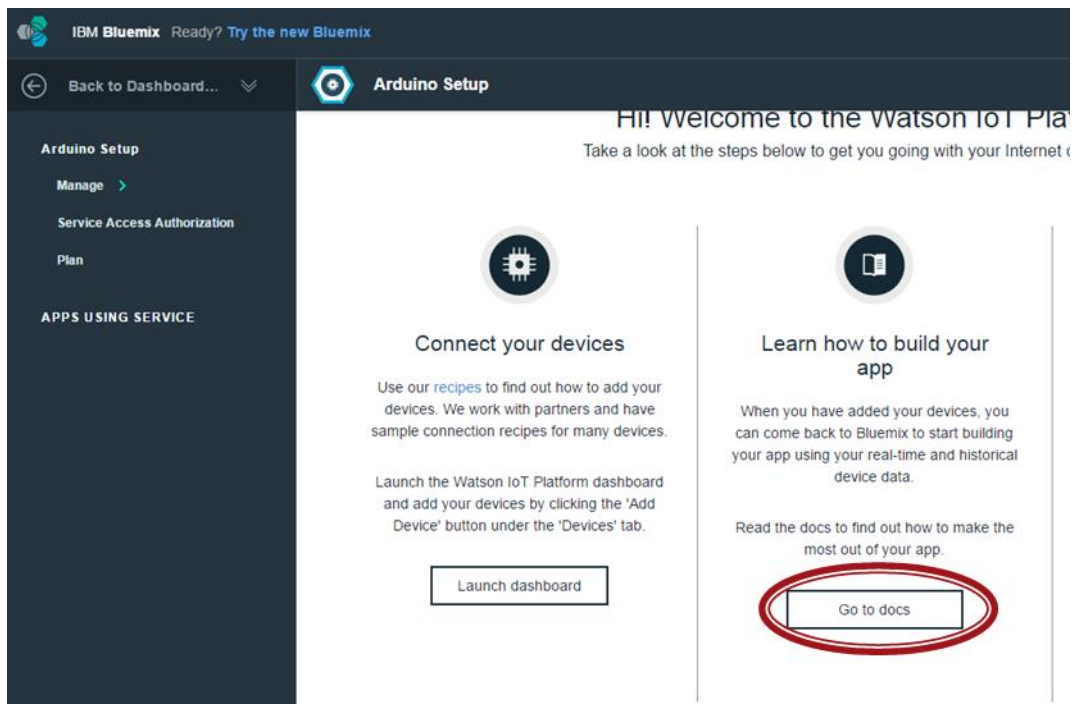
Authentication tokens are non-recoverable. If you misplace this token, you will need to re-register the device to generate a new authentication token.

[Find out how to add these credentials to your device ↗](#)

Connection Information	
Device ID	Ptyxiakh
Device Type	Arduino
Date Added	Friday, May 20, 2016
Added By	evangelos.farantatos@gmail.com
Connection State	Registered Refresh

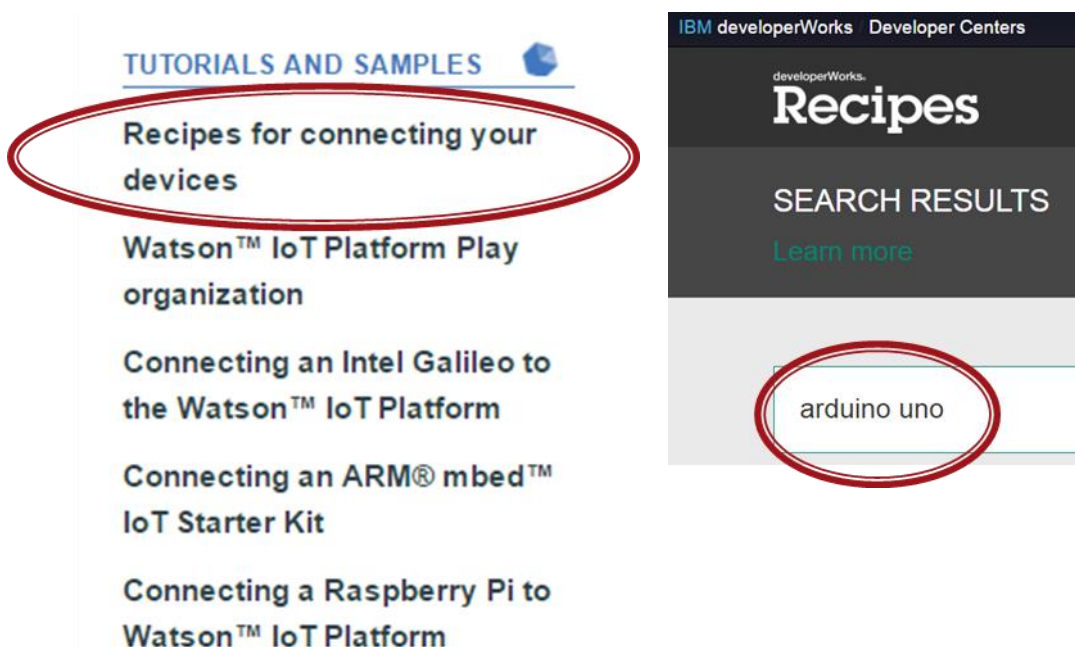
Εικόνα 4.12: Χαρακτηριστικά Συσκευής

Για να επικοινωνήσει η φυσική συσκευή όμως με την πλατφόρμα, θα πρέπει να προγραμματιστεί κατάλληλα το Arduino board. Σε αυτό θα βοηθήσει η ιστοσελίδα «recipes» που αναφέρθηκε στο κεφάλαιο 3. Επιστροφή στην παρακάτω οθόνη και επιλογή «Go to docs», στο μεσαίο μέρος της σελίδας με τίτλο «Learn how to build your app».



Εικόνα 4.13: Παραμετροποίηση Συσκευής – Προαιρετικά Βήματα

Στο δεξί μέρος της νέας καρτέλας που θα ανοίξει, διακρίνεται η επιλογή «Recipes for connecting your devices». Αυτός ο σύνδεσμος οδηγεί στην ιστοσελίδα «Recipes» για την αναζήτηση του κατάλληλου οδηγού που ανταποκρίνεται στις ανάγκες της εργασίας. Στο πεδίο αναζήτησης μπορούν να χρησιμοποιηθούν λέξεις-κλειδιά όπως «Arduino» για να εμφανιστούν τα σχετικά αποτελέσματα.



Εικόνα 4.14: Παραμετροποίηση Συσκευής – Προαιρετικά Βήματα

Από τη λίστα των αποτελεσμάτων που ακολουθεί, κατάλληλο recipe είναι αυτό με τον τίτλο «**Connect an Arduino Uno device to the IBM Watson IoT Platform**». Ο χρήστης Αριστείδης Σ. Βυζανιάρης - Ευάγγελος Γ. Φαραντάτος

μπορεί να ακολουθήσει τα βήματα που περιγράφονται, από το πρώτο στάδιο των προαπαιτούμενων μέχρι και τον κώδικα (όπου δίνεται ένα παράδειγμα χρήσης το οποίο μπορεί να τροποποιηθεί σύμφωνα με τον σκοπό του κάθε χρήστη).

Όπως αναφέρεται στο recipe, από άποψη εξοπλισμού (hardware) θα χρειαστεί το Arduino Uno board, το Ethernet shield και στην περίπτωση της εργασίας το ειδικό shield αισθητήρων που διατίθεται. Η συνδεσμολογία τους είναι αυτή που περιγράφεται στο κεφάλαιο 3, χωρίς περαιτέρω αλλαγές. Για τη σύνδεση με τον ηλεκτρονικό υπολογιστή χρησιμοποιείται καλώδιο USB male – male, το οποίο λειτουργεί και ως τροφοδοσία της συσκευής.

Στο **GitHub**²³, περιέχεται πρόγραμμα σε μορφή κειμένου, το οποίο διαβάζει τιμές θερμοκρασίας από τον εσωτερικό αισθητήρα θερμότητας του Arduino board – παρομοιάζεται με την ανάγνωση δεδομένων ενός αισθητήρα θερμότητας του επεξεργαστή σε έναν Η/Υ.

Αυτός ο αισθητήρας, δεν παρέχει ακριβείς μετρήσεις οπότε χρησιμοποιείται ο αισθητήρας του κεφαλαίου 3. Ο κώδικας θα πρέπει να μεταβληθεί κατάλληλα ώστε η ανάγνωση δεδομένων να γίνεται από το pin A0 του Arduino board. Αρχικά:

- Πρέπει να γίνει εκκίνηση του Arduino IDE
- Αντιγραφή κώδικα (επιλογή κειμένου με τον κέρσορα, και εντολή Copy)
- Επικόλληση κώδικα σε κενό sketch (εντολή Paste)
- Αποθήκευση sketch (Εντολή Save, δίνεται όνομα π.χ. Temperature)registered.ino)

Προαπαιτούμενο για τη λειτουργία του sketch είναι να «φορτωθούν» οι κατάλληλες βιβλιοθήκες όπως εξηγείται στο κεφάλαιο 3. Στη συνέχεια, θα γίνει η παραμετροποίηση του κώδικα για τη σωστή ταυτοποίηση της συσκευής στην πλατφόρμα αλλά και την ανάγνωση – αποστολή δεδομένων. Τα πεδία που πρέπει να αλλαχθούν στον κώδικα είναι τα:

- AUTHTOKEN – αντικαθίσταται με το Device Authorization Token που αποκτήθηκε σε προηγούμενο βήμα

²³ <https://github.com/ibm-messaging/iot-arduino>

- `MS_PROXY` – στη σειρά «`uuhsp.messaging.internetofthings.ibmcloud.com`», αντικαθίσταται το "`uuhsp`" με το `organization ID` (π.χ. `91ww71`).
- `CLIENT_ID` – στη σειρά «`d:uuhsp:iotsample-arduino:00aabbccde03`», αντικαθίσταται το «`iotsample-arduino`» με το `Device Type` (π.χ. `Arduino`) που ορίσαμε, το "`aabbccde03`" με το `Device ID` (π.χ. `Ρτυχιακή`) και το "`uuhsp`" με το `organization ID`.

Παρατίθεται το ανάλογο κομμάτι κώδικα, συμπληρωμένο σύμφωνα με τα στοιχεία του λογαριασμού της παρούσης εργασίας:

```
#define CLIENT_ID "d:91ww71:Arduino:Ptyxiakh"  
#define MS_PROXY "91ww71.messaging.internetofthings.ibmcloud.com"  
#define AUTHTOKEN "ScT1Q7tcCR)qI+xD!"
```

Εικόνα 4.15: Ταυτοποίηση συσκευής

Όπως αναφέρθηκε και στο θεωρητικό μέρος, θα γίνει χρήση πρωτόκολλου επικοινωνίας MQTT μεταξύ της πλατφόρμας και του εξοπλισμού. Αυτό εξυπηρετεί την εγγραφή σύμφωνα με το μοντέλο “publish/subscribe” και τη μετάδοση της πληροφορίας που θέλουμε (τιμή θερμοκρασίας).

```

void loop() {
  int rc = -1;
  if (!client.isConnected()) {
    Serial.print("Connecting using Registered mode with clientid : ");
    Serial.print(CLIENT_ID);
    Serial.print("\tto MQTT Broker : ");
    Serial.print(MS_PROXY);
    Serial.print("\ton topic : ");
    Serial.println(PUBLISH_TOPIC);
    while (rc != 0) {
      rc = ipstack.connect(MS_PROXY, MQTT_PORT); //Σύνδεση πρώτη φορά }
      MQTTPacket_connectData options = MQTTPacket_connectData_initializer;
      options.MQTTVersion = 3;
      options.clientID.cstring = CLIENT_ID;
      options.username.cstring = AUTHMETHOD;
      options.password.cstring = AUTHTOKEN;
      options.keepAliveInterval = 10;
      rc = -1;
      while ((rc = client.connect(options)) != 0); //unsubscribe , αν έχει
προηγηθεί subscribe
      client.unsubscribe(SUBSCRIBE_TOPIC);
      if ((rc = client.subscribe(SUBSCRIBE_TOPIC, MQTT::QOS0,
messageArrived)) != 0) { Serial.print("Subscribe failed with return code
: ");
        Serial.println(rc);
      } else {Serial.println("Subscribed\n"); // Έλεγχος επιτυχίας subscribe
      }
      Serial.println("Subscription tried.....");
      Serial.println("Connected successfully\n");
      Serial.println("Temperature(in C)\tDevice Event (JSON)");
      Serial.println("
"); } //Σε περίπτωση επιτυχούς σύνδεσης, εκτύπωση
κατάλληλων μηνυμάτων στο σειριακό και θερμοκρασίας

```

Εικόνα 4.16: Συνάρτηση void loop()

Η τιμή της θερμοκρασίας (ή γενικά των δεδομένων) στο πρωτόκολλο MQTT περιέχεται στη μεταβλητή payload, όπως φαίνεται και παρακάτω. Παρατίθεται το τμήμα κώδικα:

```

void messageArrived(MQTT::MessageData& md) { \\Καλείται όταν καταφθάνει
ένα μήνυμα από τον server
  Serial.print("\nMessage Received\t");
  MQTT::Message &message = md.message; \\το μήνυμα που μεταφέρεται

  int topicLen = strlen(md.topicName.lenstring.data) + 1;
  \\το όνομα του θέματος στο οποίο εγγράφηκε το μήνυμα

  char * topic = md.topicName.lenstring.data;
  topic[topicLen] = '\0';
  int payloadLen = message.payloadlen + 1; \\η τιμή της θερμοκρασίας
περιέχεται στη μεταβλητή payload
  char * payload = (char*)message.payload;
  payload[payloadLen] = '\0';
  String topicStr = topic;
  String payloadStr = payload;

```

Εικόνα 4.17: Συνάρτηση messageArrived – MQTT μήνυμα

Βασική διαφορά του sketch που παρατίθεται στο σύνδεσμο του Github με το τελικό αποτέλεσμα είναι πως αρχικά χρησιμοποιείται η συνάρτηση getTemp για την ανάγνωση Αριστέιδης Σ. Βυζανιάρης - Ευάγγελος Γ. Φαραντάτος

των δεδομένων του εσωτερικού αισθητήρα. Σε αντίθεση, μπορεί να χρησιμοποιηθεί μια μεταβλητή π.χ. `temperature`, στην οποία μπορεί να αποθηκεύεται η τιμή που διαβάζεται από το `pin A0`. Για την χρήση της παραπάνω λύσης θα γίνει αρχικοποίηση με τον παρακάτω τρόπο:

```
#define pin A0 // (ή αλλιώς int pin=A0;)
int temperature;
```

Εικόνα 4.18: Αρχικοποίηση

Η παραπάνω αρχικοποίηση, τοποθετείται πριν από τη συνάρτηση «`void setup ()`», αντιθέτως το ακόλουθο κομμάτι κώδικα θα πρέπει να περιέχεται μέσα:

```
void setup() {
  Serial.begin(9600); // για το σειριακό μόνιτορ
  Ethernet.begin(mac);
  pinMode(ledPin, OUTPUT);
  delay(1000);
  Serial.println(temperature); // Εκτύπωση τιμής θερμοκρασίας στο σειριακό
  analogReference(INTERNAL); // Εσωτερική τάση αναφοράς 1.1 V
  Serial.println("Ready to start...");
}
```

Εικόνα 4.19: Συνάρτηση `void setup()`

Στη `void loop()` θα περιέχεται ο κώδικας που θα υπολογίζει τη μέση τιμή θερμοκρασίας, όπως αναφέρθηκε στο κεφ. 3.

```
int span = 20; //μέση τιμή 20 τιμών - αρχικοποίηση μετρητή
int analog = 0;
for (int i = 0; i < span; i++) {
  analog = analog + analogRead(pin); // ανάγνωση - άθροιση 20 τιμών από το pin A0
}
analog = analog / 20; //εύρεση μέσης τιμής
temperature = analog/9.3; //εξίσωση μέτρησης θερμοκρασίας
Serial.println(temperature);}
```

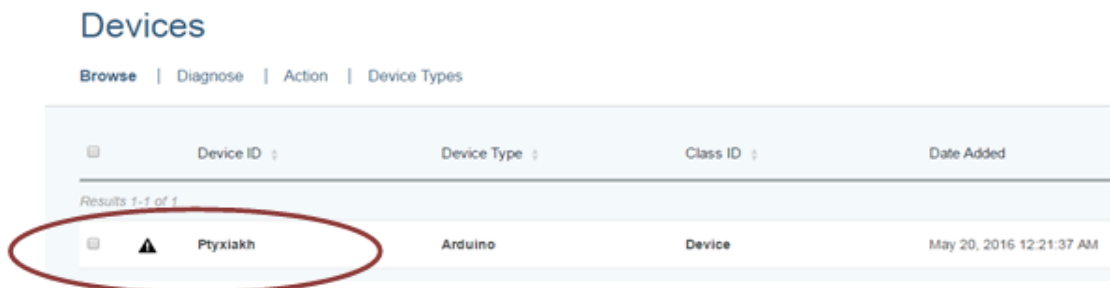
Εικόνα 4.20: Υπολογισμός θερμοκρασίας

Για την εμφάνιση των `json events` στην πλατφόρμα, η τιμή της θερμοκρασίας φορτώνεται σε μεταβλητή τύπου `float` με το όνομα `tempValue`. Γίνεται χρήση της συνάρτησης `dtostrf` για μετατροπή από δεδομένα τύπου `float` σε πίνακα δεδομένων `char` (που χρησιμοποιείται από τη `json`) μιας και δεν υπάρχει υποστήριξη από το λογισμικό του `Arduino` για εκτύπωση δεδομένων `float`.

```
char json[56] = "{\"d\":{\"myName\":\"Arduino Uno\",\"temperature\":\"";
// char buffer[10];
float tempValue = temperature;
dtostrf(tempValue,1,2, &json[43]);
// dtostrf(temperature,1,2, buffer);
```

Εικόνα 4.21: Πίνακας δεδομένων `Json`

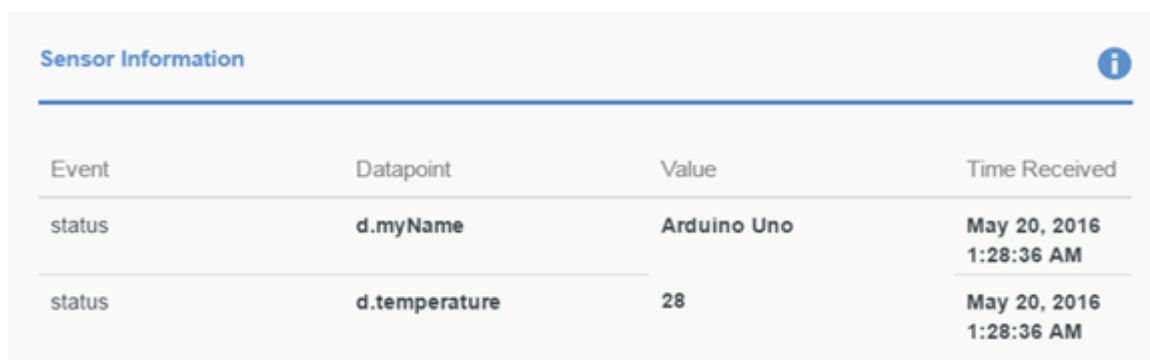
Αφού φορτωθεί επιτυχώς το sketch στο Arduino, τότε πρέπει να συνδεθεί με το Διαδίκτυο, συνδέοντάς το μέσω της θύρας Ethernet του shield, και με τη χρήση του κατάλληλου καλωδίου. Στη συνέχεια, γίνεται έλεγχος στην παρακάτω σελίδα της πλατφόρμας, όπου θα πρέπει να υπάρχει η παρακάτω εικόνα:



Device ID	Device Type	Class ID	Date Added
Ptyxiakh	Arduino	Device	May 20, 2016 12:21:37 AM

Εικόνα 4.22: Απεικόνιση της συσκευής «Ptyxiakh» στην πλατφόρμα

Η συσκευή είναι ορατή στην πλατφόρμα, και με την επιλογή της γίνεται μεταφορά στην παρακάτω σελίδα:



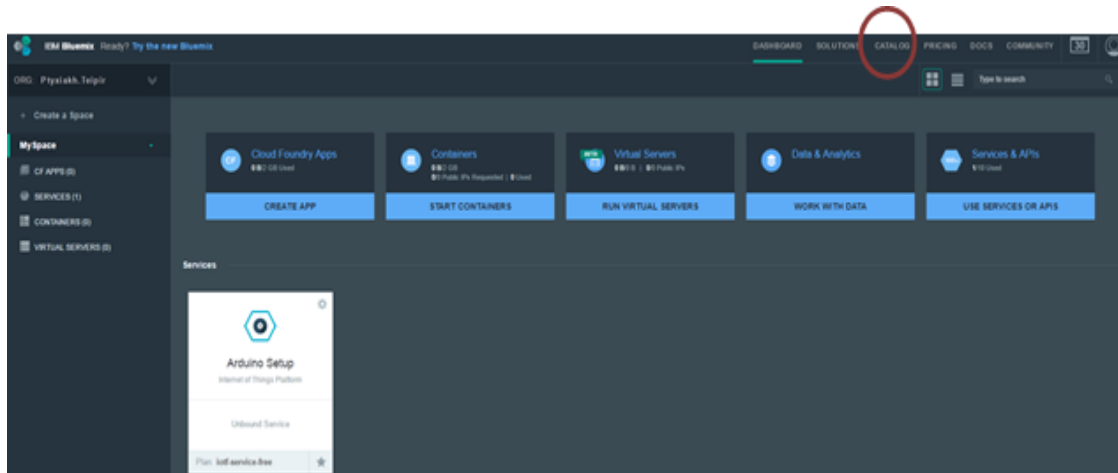
Event	Datapoint	Value	Time Received
status	d.myName	Arduino Uno	May 20, 2016 1:28:36 AM
status	d.temperature	28	May 20, 2016 1:28:36 AM

Εικόνα 4.23: Απεικόνιση της θερμοκρασίας στην πλατφόρμα

Όπως διακρίνεται, η τιμή 28 είναι η τιμή του αισθητήρα που μεταδόθηκε από τη συσκευή στην πλατφόρμα. Η τιμή αυτή ανανεώνεται (στη δεξιά στήλη φαίνονται τα στοιχεία λήψης των δεδομένων – ημερομηνία/ώρα.)

4.1.2 Οπτικοποίηση δεδομένων

Για την οπτικοποίηση των δεδομένων γραφικά (chart) θα χρειαστεί επιπρόσθετη υπηρεσία του Bluemix. Αρχικά ο χρήστης μεταφέρεται στη λίστα με τον κατάλογο των υπηρεσιών:



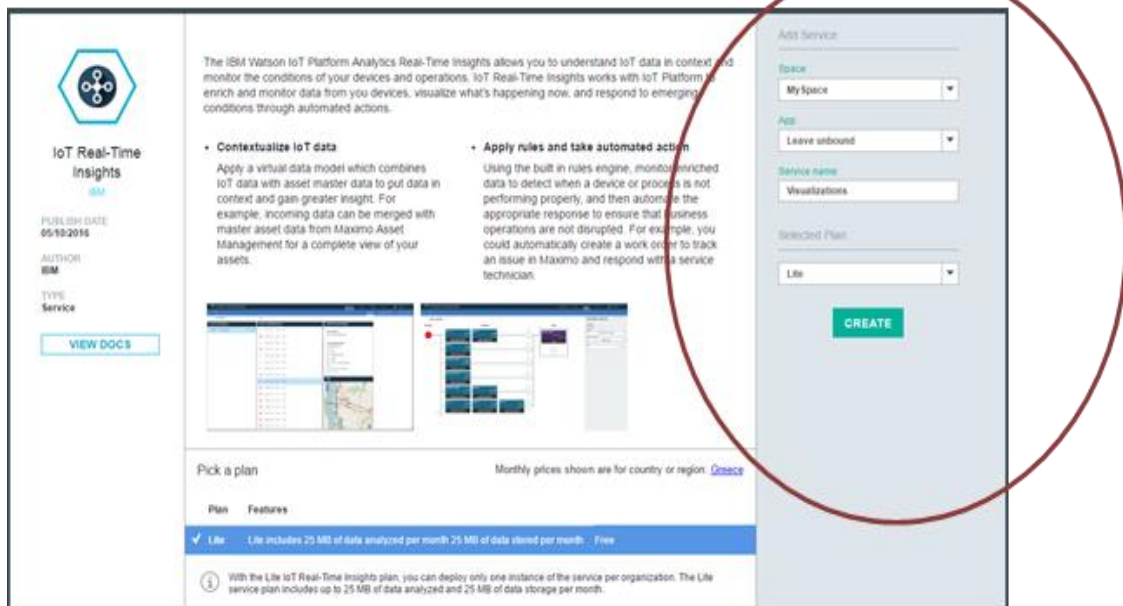
Εικόνα 4.24: Κατάλογος υπηρεσιών

Στη λίστα με τις διαθέσιμες υπηρεσίες «Internet of Things» επιλέγεται η «IoT Real-Time Insights».



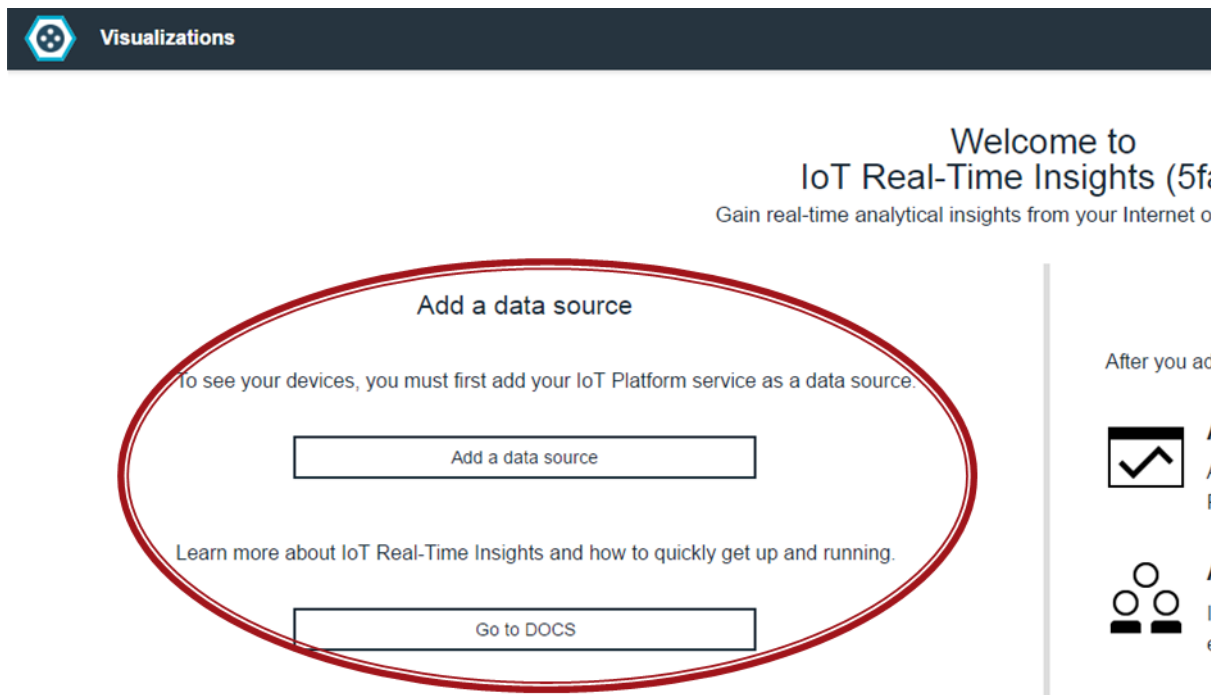
Εικόνα 4.25: Επιλογή «IoT Real-Time Insights»

Ακολουθείται η ίδια διαδικασία (με την πρώτη υπηρεσία) όσον αφορά την ονοματοδοσία και τις απαιτήσεις – το δωρεάν πλάνο καλύπτει τις ανάγκες της εργασίας.



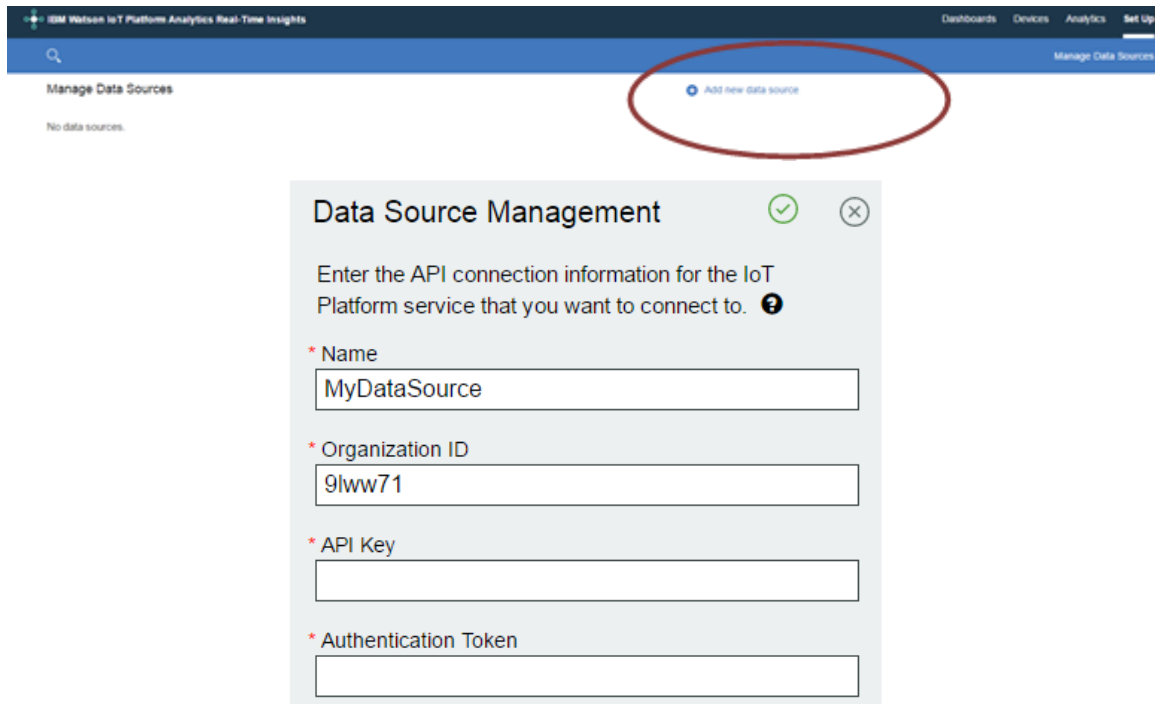
Εικόνα 4.26: «IoT Real-Time Insights»

Συνεχίζοντας, στην αρχική σελίδα της εφαρμογής, επιλέγεται «Add a data source» και ο χρήστης μεταφέρεται σε νέα σελίδα όπου και θα γίνουν παραμετροποιήσεις:



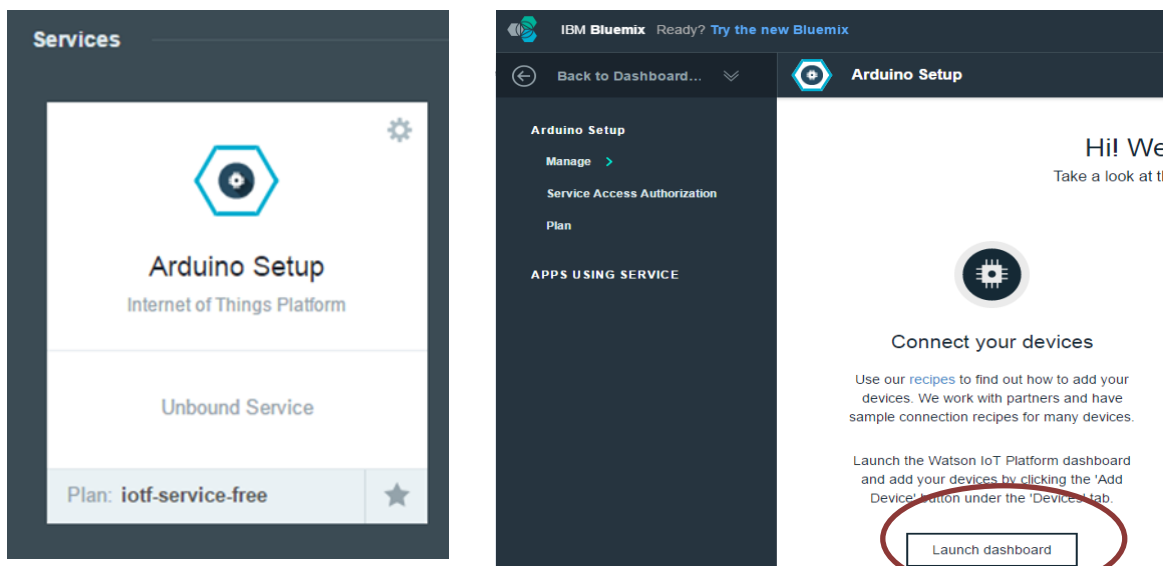
Εικόνα 4.27: «Add a data source»

Το πρώτο βήμα είναι η επιλογή «Add new data source» και συμπλήρωση των κενών πεδίων στη λίστα που εμφανίζεται. Τα πρώτα δύο πεδία είναι γνωστά (Όνομα – στην ευχέρεια του χρήστη, Organization ID – είναι γνωστό από προηγούμενα βήματα):

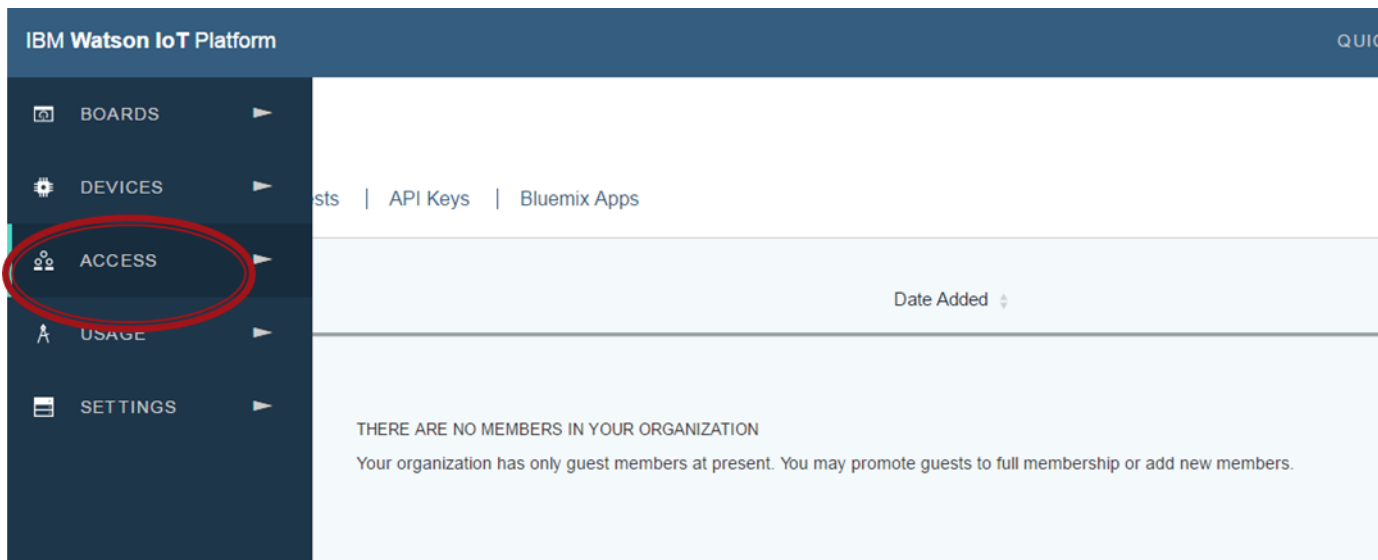


Εικόνα 4.28: «Add new data source»

Για τα δύο τελευταία πεδία, ο χρήστης θα χρειαστεί να μεταφερθεί στη σελίδα του Bluemix Dashboard και να επιλέξει την πρώτη υπηρεσία που δημιουργήθηκε (π.χ. Arduino Setup). Επιλέγοντας «Launch Dashboard», εμφανίζεται στο αριστερό μέρος της οθόνης μια λίστα με την επιλογή «ACCESS» να διακρίνεται.

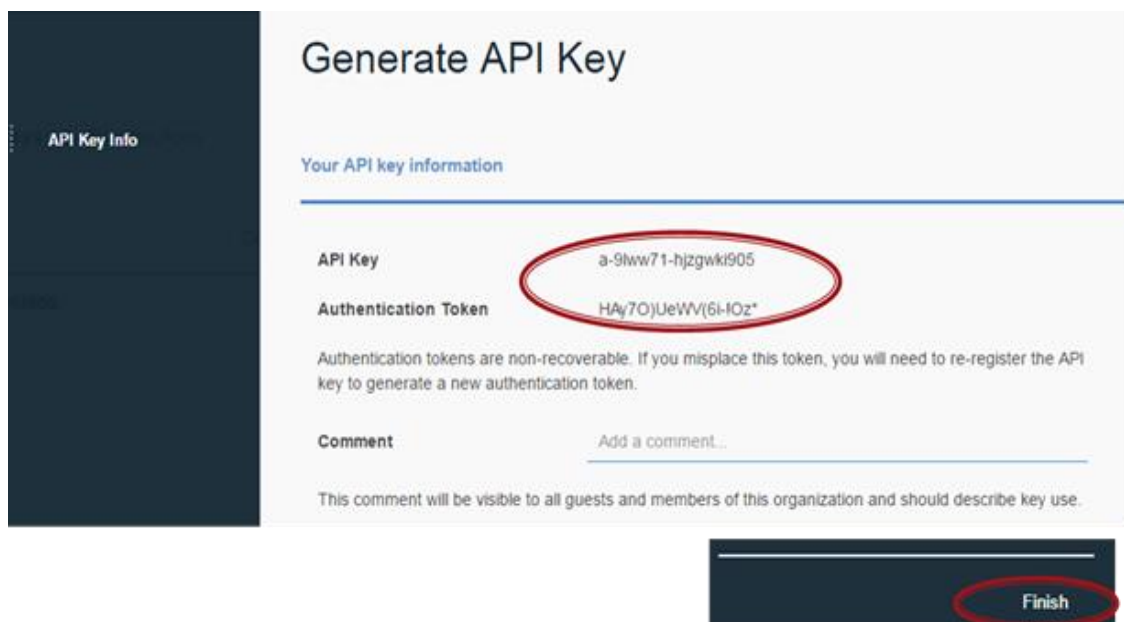


Εικόνα 4.29: Επιλογή «Arduino Setup»



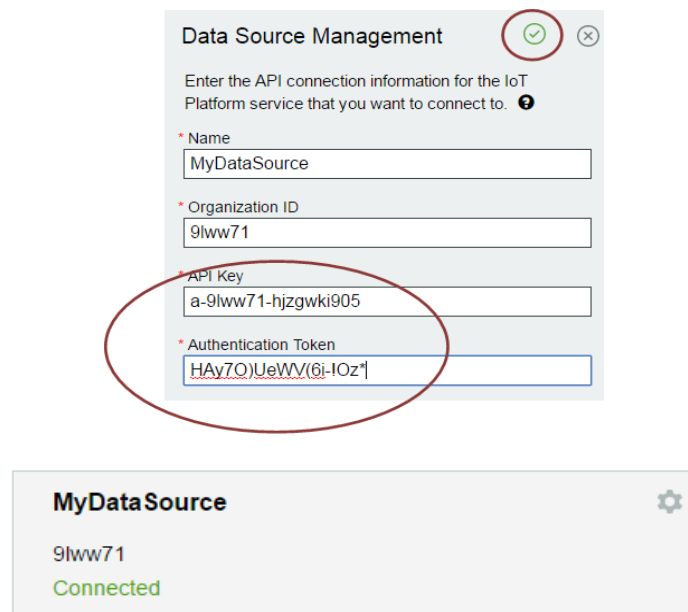
Εικόνα 4.30: Συνέχεια - Επιλογή «Access»

Στην επόμενη οθόνη, επιλέγεται η καρτέλα API Keys και η επιλογή Generate API Key. Αμέσως εμφανίζονται οι πληροφορίες που χρειάζονται για να συμπληρωθούν τα παραπάνω κενά πεδία. Ο χρήστης οφείλει να διατηρήσει τις συγκεκριμένες πληροφορίες σε ασφαλές μέρος, καθώς σε ενδεχόμενη απώλεια είναι μη ανακτήσιμες. Υπάρχει και σχετική προειδοποίηση προς το χρήστη, στην παρακάτω σελίδα:



Εικόνα 4.31: « Generate API Key »

Για να προχωρήσει η διαδικασία, ενημερώνονται τα κενά πεδία και στη συνέχεια επιλέγεται το σύμβολο «✓» για να ενημερωθεί η πλατφόρμα. Επιτυχής και έγκυρη ενημέρωση επιβεβαιώνεται από την εμφάνιση της εικόνας με τη σήμανση «Connected».



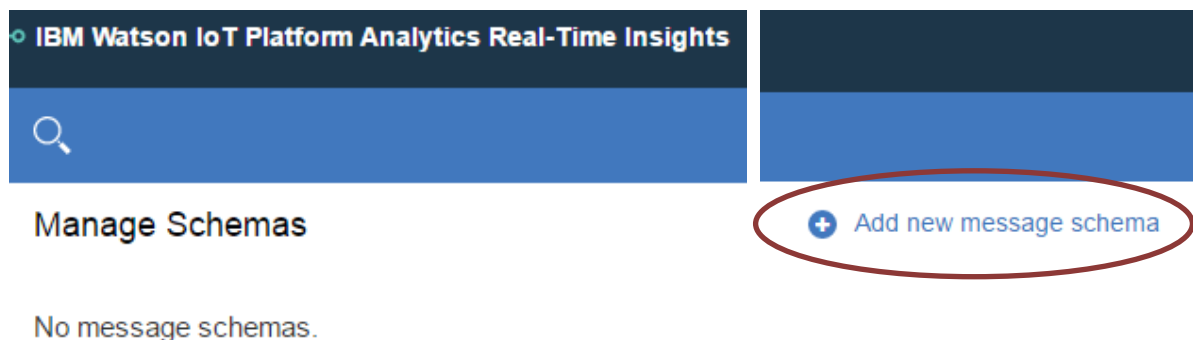
The image shows a 'Data Source Management' dialog box with a green checkmark icon in the top right corner. The dialog contains the following fields:

- Name: MyDataSource
- Organization ID: 9lww71
- API Key: a-9lww71-hjzgwki905
- Authentication Token: HAY7Q)UeWV(6i-IOz*

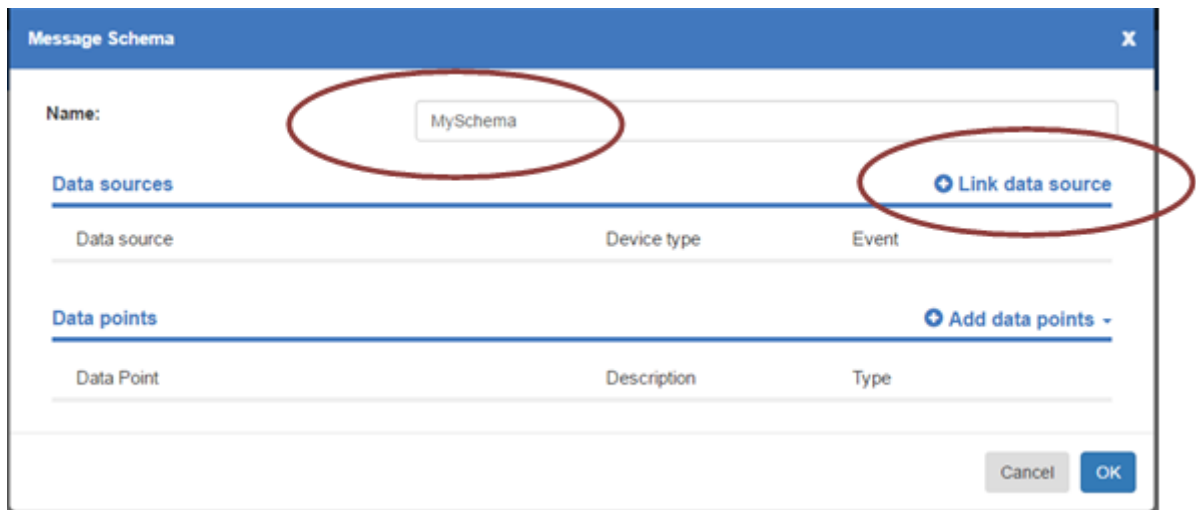
Below the dialog is a card for 'MyDataSource' with a gear icon in the top right corner. The card displays the Organization ID '9lww71' and the status 'Connected' in green text.

Εικόνα 4.32: Επιβεβαίωση ενεργειών

Ακολούθως, επιλέγεται η καρτέλα Manage Schemas και η επιλογή Add new manage schema. Ο χρήστης πληκτρολογεί το όνομα που επιθυμεί στο ανάλογο πεδίο και συνεχίζει επιλέγοντας Link data source. Η παρακάτω διαδικασία συνδέει την πρώτη υπηρεσία που χρησιμοποιήθηκε (Internet of Things Platform) με την IoT Real Time insights προκειμένου η τελευταία να αντλήσει τα δεδομένα από την πρώτη και να τα παραστήσει γραφικά μετά από τις ακόλουθες ενέργειες.

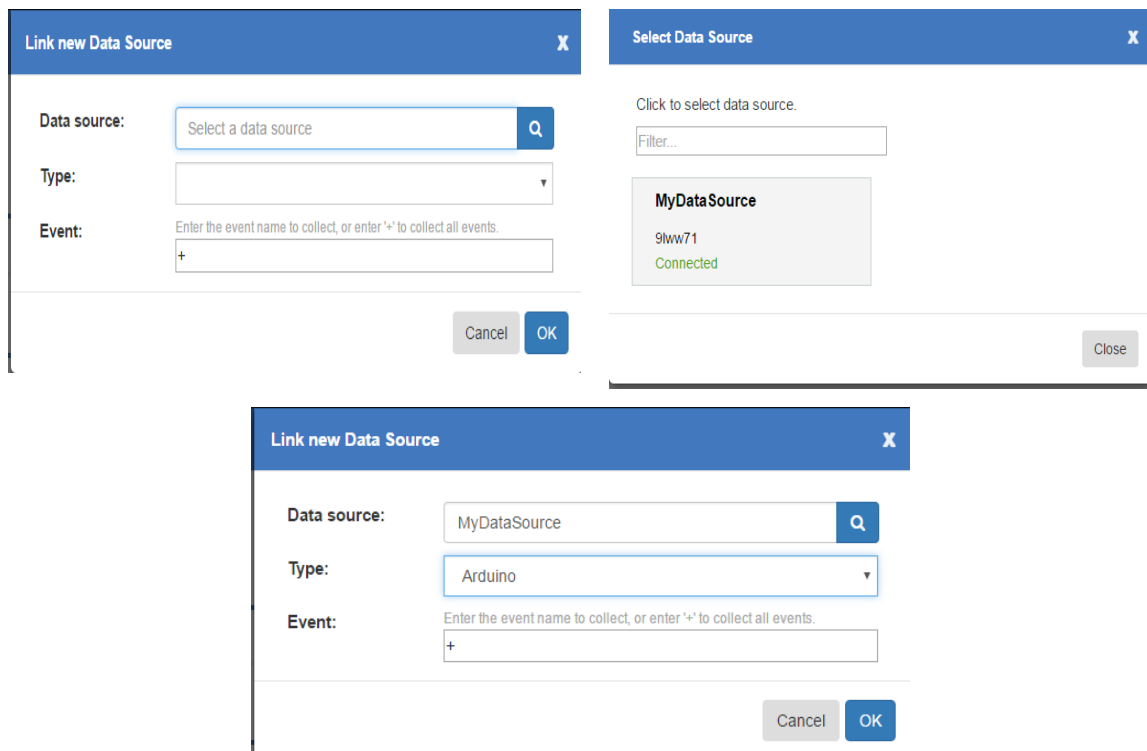


Εικόνα 4.33: Add new manage schema



Εικόνα 4.34: Message Schema

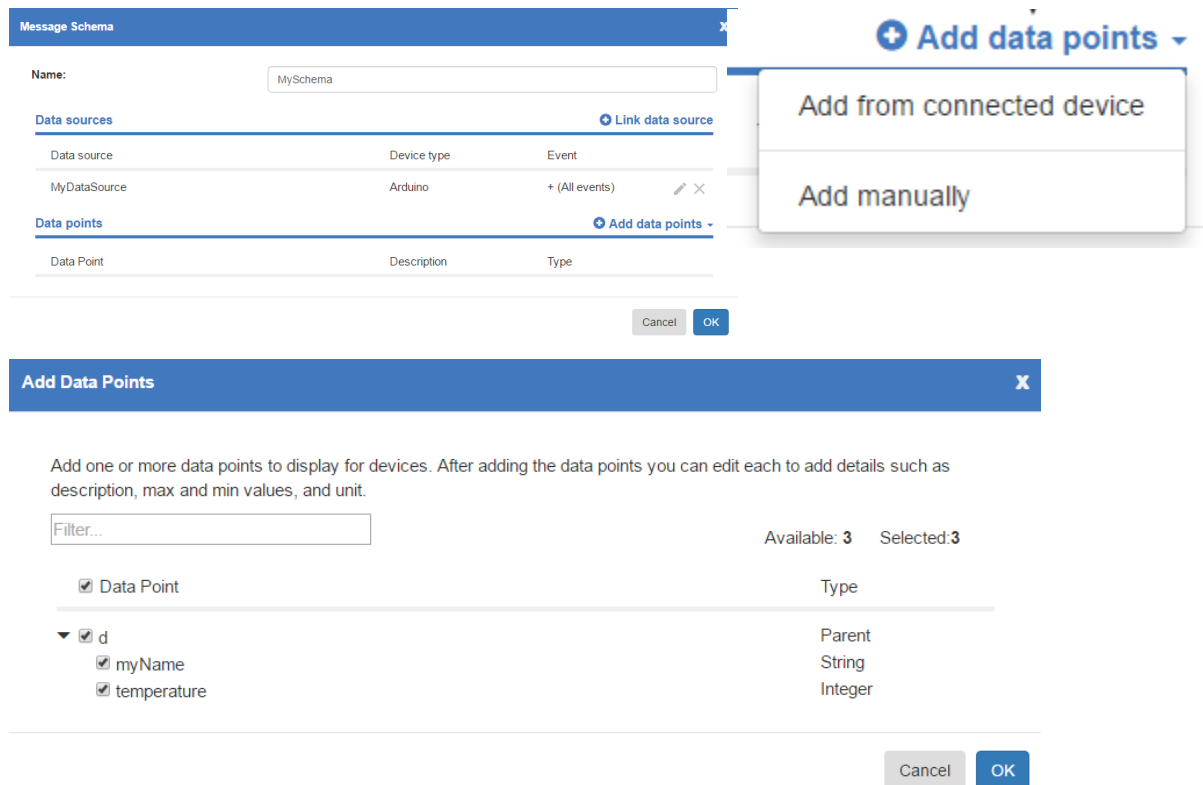
Στο αναδυόμενο παράθυρο, επιλέγεται το data source που δημιουργήθηκε στο πρώτο βήμα (π.χ. MyDataSource). Στο πεδίο Type επιλέγεται το αντίστοιχο device type (π.χ. Arduino) και η διαδικασία ολοκληρώνεται με το κουμπί OK. Το αρχικό παράθυρο θα πρέπει να έχει την παρακάτω μορφή:



Εικόνα 4.35: Επεξεργασία Data Source

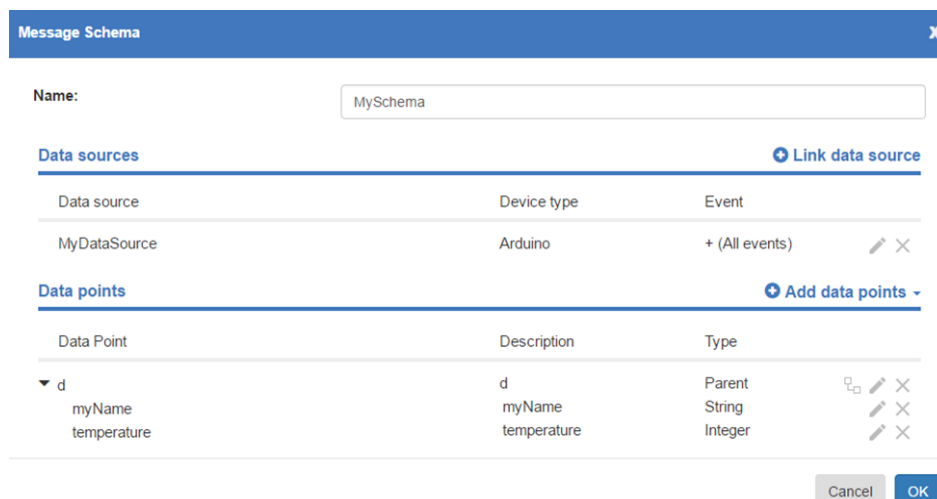
Αφού ολοκληρωθεί η παραπάνω διαδικασία, η επόμενη κίνηση θα είναι να προστεθούν τα data points - πρόκειται για τις πηγές δεδομένων της συσκευής από τα οποία μπορεί να διαλέξει ο χρήστης ποια να εμφανίζονται. Στις παρακάτω εικόνες όπου απεικονίζεται ο τρόπος προσθήκης data points, διακρίνονται οι διαθέσιμες επιλογές για την εργασία

(myName, temperature), καθώς και πληροφορίες όπως ο τύπος της τιμής της μεταβλητής (string και integer αντίστοιχα).



Εικόνα 4.36: Προσθήκη data points

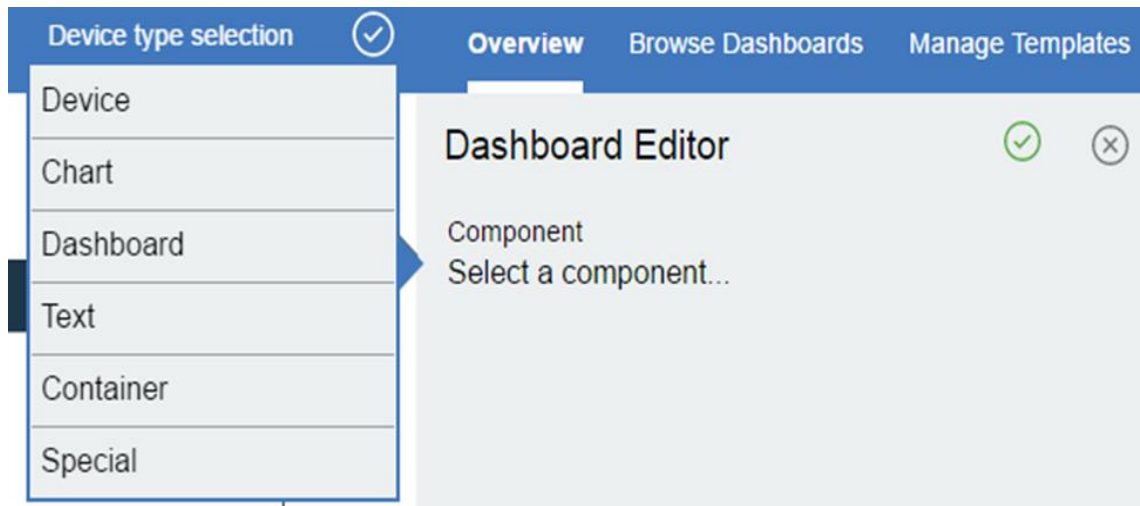
Στο τέλος των παραπάνω παραμετροποιήσεων, το παράθυρο με τίτλο «Message Schema» θα πρέπει να εμφανίζεται κατ’ αντιστοιχία με τις επιλογές του κάθε χρήστη όπως παρακάτω:



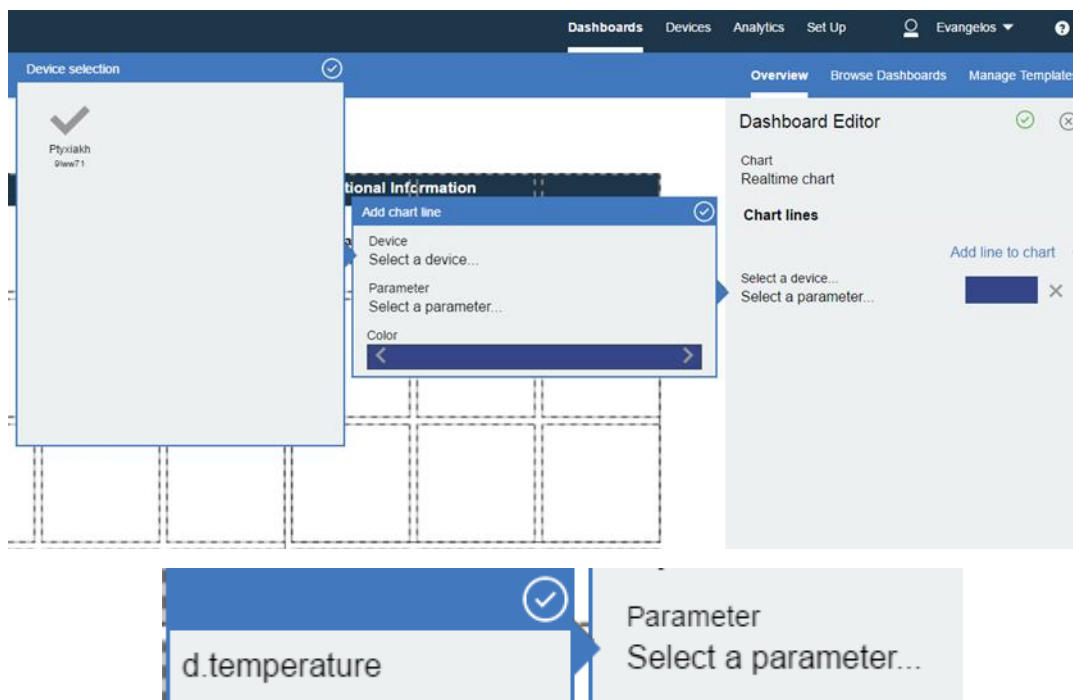
Εικόνα 4.37: Απεικόνιση παραθύρου «Message Schema»

Αυτό που απομένει στο εξής είναι η κατασκευή του γραφήματος. Ο χρήστης πρέπει να μεταφερθεί στο dashboard της υπηρεσίας όπου και θα χρειαστεί να επιλέξει «Add new

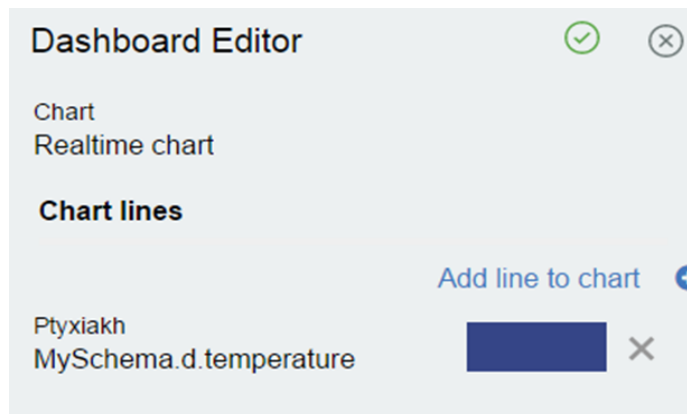
component». Από τις διαθέσιμες επιλογές (π.χ. Device, text, κ.α.), αυτό που ανταποκρίνεται είναι το «chart». Για την παραμετροποίηση του chart, αρχικά επιλέγεται η συσκευή της οποίας τα δεδομένα θα οπτικοποιηθούν (π.χ. Ρτυχιακή), και στη συνέχεια η μεταβλητή (π.χ. d.temperature). Στο τέλος της διαδικασίας, ζητείται επιβεβαίωση των ενεργειών με την επιλογή του εικονιδίου με το πράσινο βέλος.



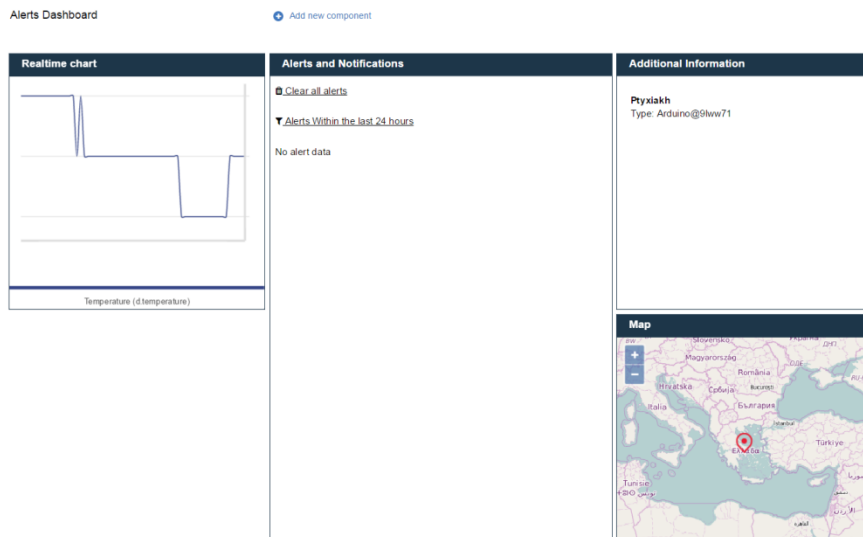
Εικόνα 4.38: Κατασκευή γραφήματος



Εικόνα 4.39: Επεξεργασία γραφήματος



Εικόνα 4.40: Ολοκλήρωση επεξεργασίας γραφήματος
 Η τελική μορφή θα είναι όπως φαίνεται στην παρακάτω εικόνα. Γραφική αναπαράσταση των δεδομένων σε πραγματικό χρόνο στο αριστερό μέρος της οθόνης και στο κάτω δεξιά υπάρχει προκατασκευασμένος ένας χάρτης με την τοποθεσία της συσκευής που αποστέλλει δεδομένα.



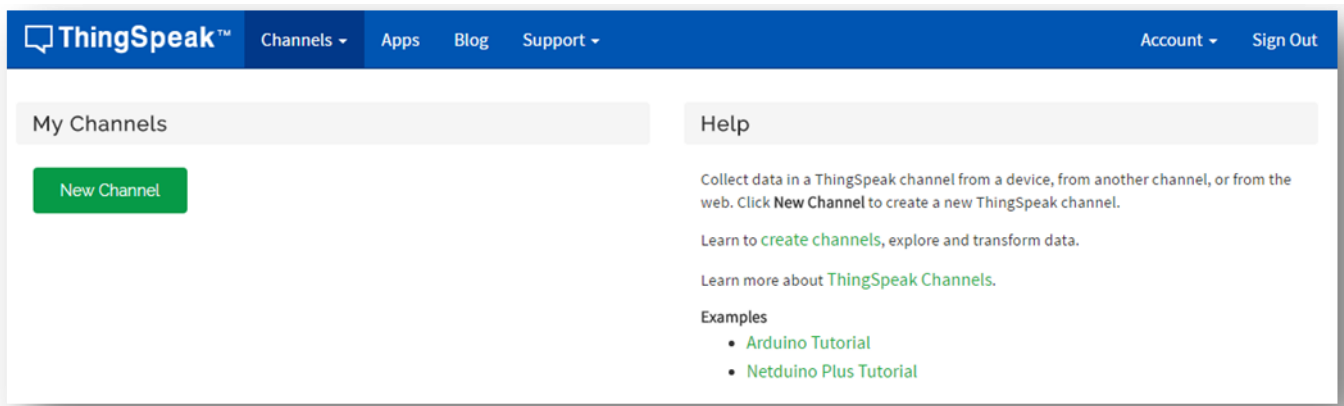
Εικόνα 4.41: Τελική μορφή - Απεικόνιση δεδομένων

4.2 Οπτικοποίηση Δεδομένων με χρήση της πλατφόρμας Thingspeak

Όπως αναφέρθηκε σε προηγούμενο κεφάλαιο, η πλατφόρμα Thingspeak παρέχει διάφορες υπηρεσίες που απευθύνονται αποκλειστικά στη δημιουργία εφαρμογών IoT. Η πλατφόρμα αυτή προσφέρεται δωρεάν για τον χρήστη και απαιτείται μόνο η χρήση ενός έγκυρου λογαριασμού ηλεκτρονικού ταχυδρομείου για την σύνδεση του σε αυτήν.

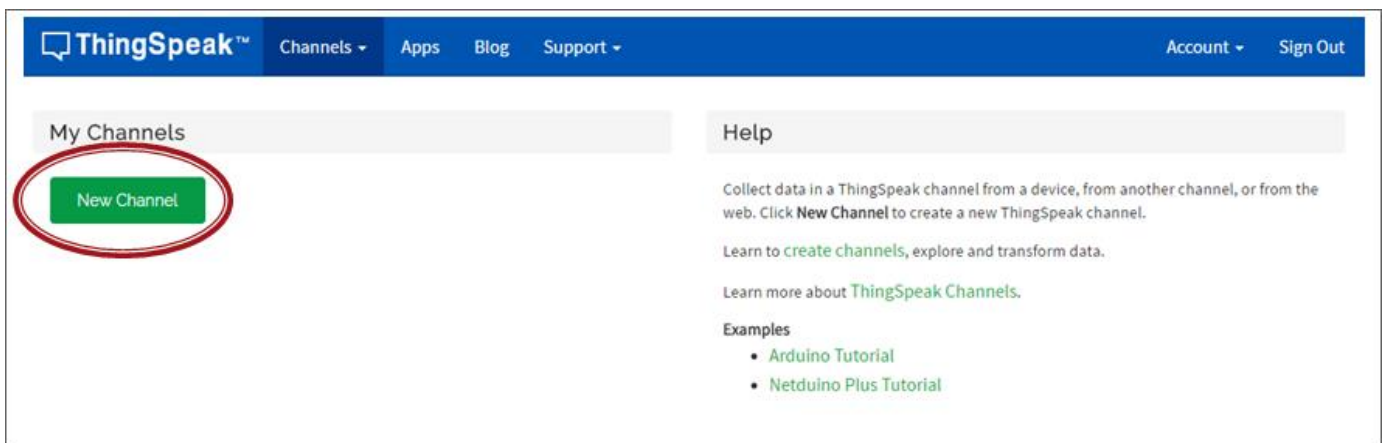
4.2.1 Σύνδεση της συσκευής με την πλατφόρμα Thingspeak.

Για να χρησιμοποιήσει κάποιος το ThingSpeak, θα πρέπει πρώτα να εγγραφεί. Ύστερα ο χρήστης θα συνδεθεί με την αρχική σελίδα της πλατφόρμας η οποία φαίνεται παρακάτω:



Εικόνα 4.42: Αρχική σελίδα της πλατφόρμας Thingspeak

Στην αριστερή πλευρά της οθόνης υπάρχει η επιλογή «**New Channel**» η οποία και επιλέγεται, για να ξεκινήσει η δημιουργία ενός καινούργιου καναλιού όπως φαίνεται και στην παρακάτω εικόνα:



Εικόνα 4.43: Δημιουργία νέου καναλιού

Αφού δημιουργηθεί το κανάλι, εμφανίζεται ένα νέο παράθυρο στο οποίο μπορεί να γίνει η επεξεργασία του καναλιού. Στο αριστερό μέρος της οθόνης στο πεδίο «**Name**» συμπληρώνεται το όνομα που επιθυμεί ο χρήστης να δώσει στο κανάλι το οποίο δημιουργήθηκε. Η συμπλήρωση αυτού του πεδίου είναι απαραίτητη, σε αντίθεση με τα υπόλοιπα πεδία τα οποία μπορούν να παραληφθούν. Στη συνέχεια επιλέγεται στην κάτω δεξιά πλευρά της οθόνης η επιλογή «**Save Channel**» για να γίνει η αποθήκευση των αλλαγών που μόλις πραγματοποιήθηκαν. Εμφανίζεται ένα νέο παράθυρο το οποίο περιέχει περισσότερες πληροφορίες σχετικά με το κανάλι που μόλις δημιουργήθηκε,

ThingSpeak™ Channels Apps Blog Support Account Sign Out

New Channel

Name: My_Channel

Description: My channel to view temperature data from my device

Field 1: Field Label 1

Field 2:

Field 3:

Field 4:

Field 5:

Field 6:

Field 7:

Field 8:

Metadata:

Tags:
(Tags are comma separated)

Make Public:

URL:

Elevation:

Help

ThingSpeak Channel

Channels store all the data that a ThingSpeak application collects. Each channel includes eight fields that can hold any type of data, plus three fields for location data and one for status data. Once you collect data in a channel, you can use ThingSpeak apps to analyze and visualize it.

Channel Settings

- Channel Name:** Enter a unique name for the ThingSpeak channel.
- Description:** Enter a description of the ThingSpeak channel.
- Field:** Check the box to enable the field, and enter a field name. Each ThingSpeak channel can have up to 8 fields.
- Metadata:** Enter information about channel data, including JSON, XML, or CSV data.
- Tags:** Enter keywords that identify the channel. Separate tags with commas.
- Latitude:** Specify the position of the sensor or thing that collects data in decimal degrees. For example, the latitude of the city of London is 51.5072.
- Longitude:** Specify the position of the sensor or thing that collects data in decimal degrees. For example, the longitude of the city of London is -0.1275.
- Elevation:** Specify the position of the sensor or thing that collects data in meters. For example, the elevation of the city of London is 35.052.
- Make Public:** If you want to make the channel publicly available, check this box.
- URL:** If you have a website that contains information about your ThingSpeak channel, specify the URL.
- Video ID:** If you have a YouTube™ or Vimeo® video that displays your channel information, specify the full path of the video URL.

Using the Channel

You can get data into a channel from a device, website, or another ThingSpeak channel. You can then visualize data and transform it using [ThingSpeak Apps](#).

See [Tutorial: ThingSpeak and MATLAB](#) for an example of measuring dew point from a weather station that acquires data from an [Arduino®](#) device.

Save Channel

i Δεν είναι απαραίτητη η συμπλήρωση όλων των πεδίων (Εκτός από το όνομα)

Εικόνα 4.44: Επεξεργασία Καναλιού

όπως το «Channel ID», «Author» και το πότε έγινε η τελευταία ενημέρωση στο κανάλι. Επίσης στην κάτω αριστερή πλευρά του παραθύρου υπάρχει ένα διάγραμμα (με την ονομασία: Field 1 Chart) το οποίο δεν περιέχει κάποιες τιμές ακόμα ,όπως φαίνεται στην εικόνα 4.45

My_Channel

Channel ID: **122611**
Author: Ptyxiakh.teipir
Access: Private

My channel to view temperature data from my device

Private View Public View Channel Settings API Keys Data Import / Export

+ Add Visualizations

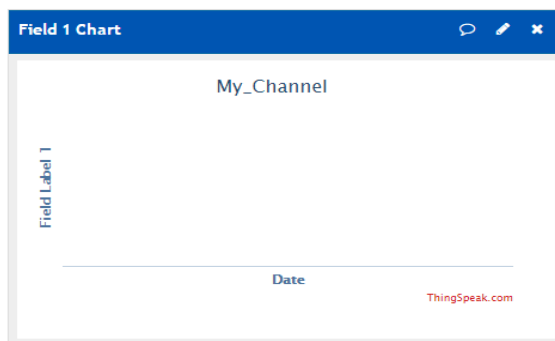
Data Export

MATLAB Analysis

MATLAB Visualization

Channel Stats

Created [about a minute ago](#)
Updated [about a minute ago](#)
0 Entries



Εικόνα 4.45: Αρχική σελίδα καναλιού

Στο ίδιο παράθυρο, πατώντας την επιλογή «**API Keys**» αυτομάτως παράγεται ένα API κλειδί εγγραφής (write API key). Το κλειδί αυτό, πρόκειται για ένα κλειδί ταυτοποίησης της συσκευής στη πλατφόρμα του Thingspeak και για αυτόν τον λόγο πρέπει να αποθηκευτεί σε ασφαλές μέρος από τον χρήστη. Σε περίπτωση απώλειας, υπάρχει η δυνατότητα να ξαναδημιουργηθεί καινούργιο. Η διαδικασία δημιουργίας κλειδιού, φαίνεται στην παρακάτω εικόνα.

The screenshot shows the ThingSpeak interface for a channel named 'My_Channel'. The channel ID is 122611, the author is Ptyxiakh.teipir, and the access is Private. The 'API Keys' tab is selected and circled in red. Below this, the 'Write API Key' section shows a key 'M0P0J9FATTF8QSV4' also circled in red, with a 'Generate New Write API Key' button. The 'Read API Keys' section shows a key 'QM6C8XJVTCLKQ5TK' and a 'Generate New Read API Key' button. On the right, the 'Help' section explains API keys and provides 'API Keys Settings' with instructions for Write, Read, and Note keys. Below the help are three code blocks for channel management: 'Create a Channel' (POST), 'Update a Channel' (PUT), and 'Clear a Channel' (DELETE).

Εικόνα 4.46: Δημιουργία API Keys

Αφού έχουν γίνει οι παραπάνω ενέργειες το επόμενο βήμα που απομένει είναι να συνδεθεί με την πλατφόρμα η συσκευή. Για να επικοινωνήσει η φυσική συσκευή με την πλατφόρμα, θα πρέπει να προγραμματιστεί κατάλληλα το Arduino board. Στο **GitHub**²⁴ βρίσκεται ένα παράδειγμα αποστολής δεδομένων στην πλατφόρμα, από αισθητήρα θερμοκρασίας συνδεδεμένο σε Arduino Uno, το οποίο και θα χρησιμοποιηθεί.

Πριν φορτωθεί ο κώδικας στο Board, θα πρέπει να τροποποιηθούν κάποια σημεία του κώδικα όπως φαίνεται παρακάτω:

```
String writeAPIKey = "QQJJAEG3ILA8CAB3";
```

Εικόνα 4.47: Συμπλήρωση πραγματικού API χρήση

Εδώ αντικαθίσταται το API key του παραδείγματος με το πραγματικό που έχει ήδη δημιουργηθεί ακολουθώντας την παραπάνω διαδικασία. Ακολουθώντας τη λογική που

²⁴ [http://github.com/iobridge/ThingSpeak-Arduino-](http://github.com/iobridge/ThingSpeak-Arduino-Examples/blob/master/Ethernet/Arduino_to_ThingSpeak.ino)

Examples/blob/master/Ethernet/Arduino_to_ThingSpeak.ino

ήδη εξηγήθηκε για τη χρήση της πλατφόρμας IBM Bluemix, θα αρχικοποιηθεί μια μεταβλητή (`temperature`) όπως φαίνεται παρακάτω όπου θα αποθηκεύεται η τιμή της θερμοκρασίας, από το Pin A0:

```
#define pin A0 // (or int pin=A0;)
int temperature;
```

Εικόνα 4.48: Αρχικοποίηση

Στη συνέχεια και μέσα στη συνάρτηση `void setup()` πρέπει να περιέχεται το ακόλουθο κομμάτι κώδικα:

```
void setup()
{
  startEthernet(); // Έναρξη του Ethernet στο Arduino
  Serial.begin(9600); // για το σειριακό μόνιτορ
  analogReference(INTERNAL); // Εσωτερική τάση αναφοράς 1.1 V
  Serial.println("Ready to start...");
}
```

Εικόνα 4.49: Συνάρτηση `void setup()`

Στη συνάρτηση `void loop` πρέπει να προστεθεί ο κώδικας όπως φαίνεται παρακάτω, για την ανάγνωση δεδομένων από το pin A0 και τον υπολογισμό της μέσης τιμής θερμοκρασίας (βλ. Κεφάλαιο 3).

```
void loop()
{
  int span = 20;
  int analog = 0;
  for (int i = 0; i < span; i++) {
    analog = analog + analogRead(pin);
  }
  analog = analog / 20;
  temperature = analog/9.3;
  Serial.println(temperature);
}
```

Εικόνα 4.50: Συνάρτηση `void loop()`

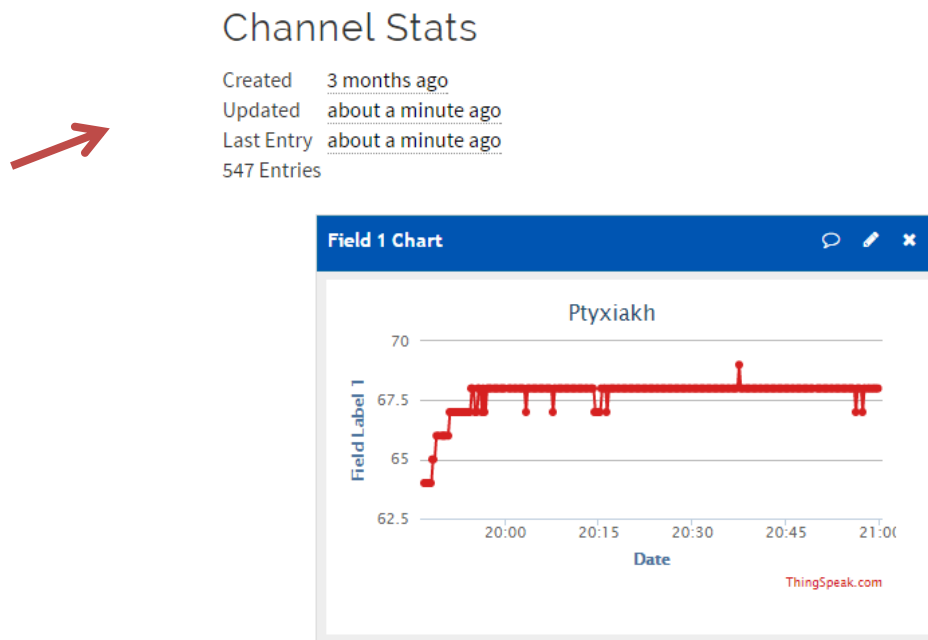
Τέλος, θα πρέπει να γίνει αλλαγή στην παρακάτω γραμμή ώστε οι τιμές που θα αποστέλλονται στην πλατφόρμα να είναι αυτές που περιέχονται στη μεταβλητή `temperature` και όχι απευθείας από το Pin A0 – λόγω των τροποποιήσεων που πραγματοποιήθηκαν παραπάνω.

```
updateThingSpeak("field1="+temperature);
```

Εικόνα 4.51: Ενημέρωση πλατφόρμας για την τιμή θερμοκρασίας

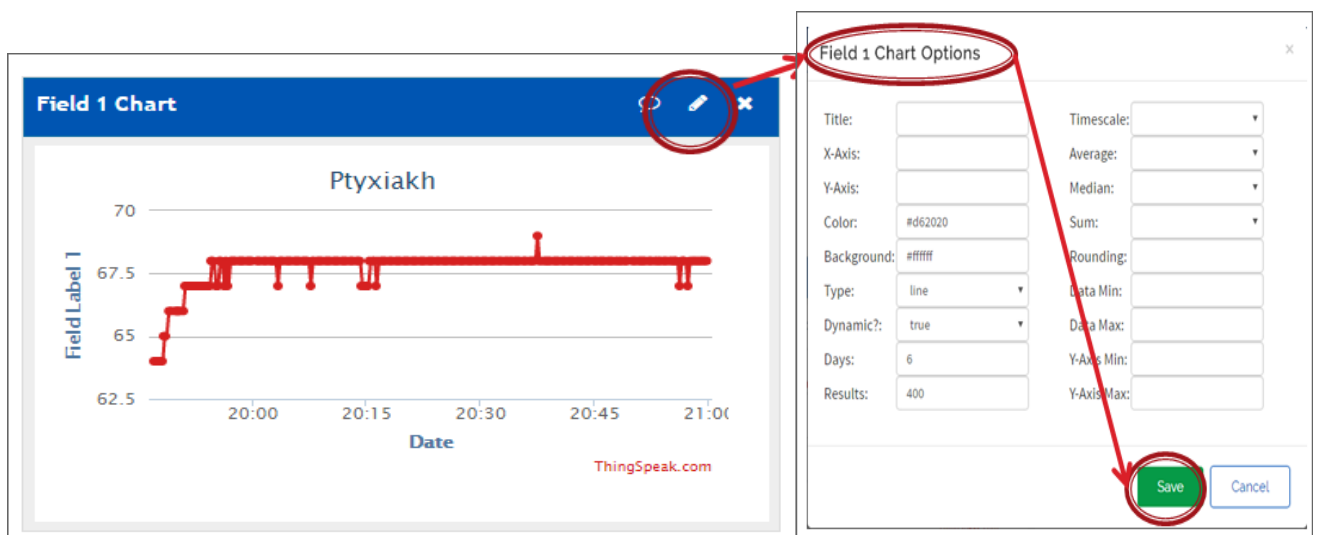
Αφού τροποποιηθεί ο κώδικας, ύστερα φορτώνεται στο Arduino Uno. Το επόμενο βήμα είναι να συνδεθεί το Board μέσω της θύρας Ethernet στο διαδίκτυο. Πατώντας την επιλογή «**Private View**» εμφανίζεται το παράθυρο της εικόνας 4.51, με την διαφορά ότι Αριστέιδης Σ. Βυζανιάρης - Ευάγγελος Γ. Φαραντάτος

το διάγραμμα το οποίο ήταν κενό, τώρα δείχνει τα δεδομένα που στάλθηκαν, όπως φαίνεται παρακάτω:



Εικόνα 4.52: Εικονοποίηση δεδομένων

Επιλέγοντας το σύμβολο με το μολύβι όπως φαίνεται στην εικόνα 4.52 παρακάτω, ο χρήστης έχει την δυνατότητα να μορφοποιήσει το διάγραμμα, αλλάζοντας το χρώμα, τις κεφαλίδες, τις τιμές κλπ.



Εικόνα 4.53: Επεξεργασία γραφήματος

4.3 Συμπεράσματα

Με το πέρας των ενεργειών για την υλοποίηση της υπηρεσίας τόσο στην πλατφόρμα IBM Bluemix όσο και στη Thingspeak, ο χρήστης μπορεί να καταλήξει σε χρήσιμα
Αριστείδης Σ. Βυζανιάρης - Ευάγγελος Γ. Φαραντάτος

συμπεράσματα τα οποία ξεκινούν από το επίπεδο του εξοπλισμού που επιλέχθηκε και ολοκληρώνονται με την οπτικοποίηση των δεδομένων.

Σε επίπεδο εξοπλισμού, η επιλογή του Arduino Uno board σε συνδυασμό με το Ethernet shield μπορεί να φαντάζει ιδανική για μια πρώτη γνωριμία με το IoT και τις τεχνολογίες που χρησιμοποιεί, λόγω της απλής συνδεσμολογίας του και παραμετροποίησης του. Εν αντιθέσει, θα παρουσιάσει πολλούς περιορισμούς σε περίπτωση που θα χρειαζόταν να συνδεθεί με κάποια άλλη πλατφόρμα όπως το AWS καθώς το συγκεκριμένο board δε διαθέτει αρκετή μνήμη runtime για να προχωρήσει τη φάση της ταυτοποίησης της συσκευής στην πλατφόρμα. Για αυτήν την περίπτωση θα χρειαζόταν ένας πιο ισχυρός εξοπλισμός όπως π.χ. το Arduino Yun στο οποίο αναφορά γίνεται σε παρακάτω κεφάλαιο. Επίσης χρήση αναβαθμισμένου εξοπλισμού όπως το Yun που περιέχει συνδεσιμότητα μέσω Ethernet και Wi-Fi θα σήμαινε και τη μη χρήση του Ethernet shield, άρα μικρότερο μέγεθος κατασκευής και απλούστερη συνδεσμολογία.

Όσον αφορά την πλατφόρμα, η Bluemix χρειάστηκε ένα μεγάλο πλήθος παραμετροποιήσεων τόσο για την απλή σύνδεση των δεδομένων στην πλατφόρμα όσο και για την οπτικοποίησή τους. Εν αντιθέσει, στην Thingspeak με απλές παραμετροποιήσεις είχαμε το επιθυμητό αποτέλεσμα. Βασικό όμως στοιχείο είναι πως η μεν Bluemix προσφέρει ένα μεγάλο πλήθος έτοιμων υπο-υπηρεσιών – εφαρμογών τις οποίες ο χρήστης μπορεί να επιλέξει και να συνδυάσει σύμφωνα με τις ανάγκες του. Οι υπηρεσίες αυτές αναφέρονται και στη λήψη δεδομένων από τις διάφορες συσκευές όσο και στην ανάλυση των δεδομένων που συλλέγονται αλλά και την οπτικοποίησή τους. Η Thingspeak ως απλούστερη λύση δε θα μπορούσε να χρησιμοποιηθεί σε μια ευρύτερη κλίμακα και από πλευράς ασφάλειας καθώς οι παράμετροι της Bluemix διασφαλίζουν τα δεδομένα και την επικοινωνία με πιο σύνθετο τρόπο.

5 ΕΠΙΛΟΓΟΣ – ΜΕΛΛΟΝΤΙΚΗ ΕΡΓΑΣΙΑ

5.1 ΕΠΙΛΟΓΟΣ

Η τεχνολογία IoT σε συνδυασμό με τις υπηρεσίες νέφους, μπορούν να προσφέρουν αμέτρητες δυνατότητες στους χρήστες τους οι οποίες δεν μπορούν να περιγραφούν όλες στα πλαίσια μιας πτυχιακής εργασίας. Αρχικά έγινε αναφορά και επεξήγηση των βασικών όρων Υπολογιστικού νέφους και των χαρακτηριστικών του. Αναλύθηκαν τα πρωτόκολλα, οι αρχιτεκτονικές και οι τεχνολογίες που το χαρακτηρίζουν ώστε να γίνει καλύτερη κατανόηση των μερών του και του ρόλου τους στην ύπαρξη και εξέλιξή του. Στη συνέχεια του Θεωρητικού Μέρους επεξηγήθηκε η τεχνολογία IoT, με τους παράγοντες ανάπτυξής του και το σημαντικό ρόλο των αισθητήρων. Στο πρακτικό μέρος, έγινε μια γενική αναφορά όλων των εργαλείων και μεθόδων που χρησιμοποιήθηκαν για την υλοποίηση της υπηρεσίας IoT. Κατόπιν επεξηγήθηκε ο εξοπλισμός που χρησιμοποιήθηκε, σε συνδυασμό με το λογισμικό για την κατασκευή κώδικα και την πλατφόρμα η οποία θα δεχόταν τα δεδομένα και θα τα απεικόνιζε. Σημαντικό ήταν να παρατεθούν και όλοι οι σύνδεσμοι στο Διαδίκτυο με πληροφορίες χρήσιμες για την πορεία της υλοποίησης σε κάθε βήμα και την αποφυγή όλων των προβλημάτων όπως «πού μπορώ να βρω την πλατφόρμα IBM Bluemix;», «που μπορώ να βρω παραδείγματα κώδικα;», «που μπορώ να βρω απαντήσεις στα ερωτήματα σχετικά με την πλατφόρμα που χρησιμοποιώ;» κ.α..Ακολούθως, έγινε αναλυτική περιγραφή όλων των βημάτων που ακολουθήθηκαν για να γίνουν οι απαραίτητες τροποποιήσεις σε επίπεδο πλατφόρμας και κώδικα, ώστε η συσκευή να αποστέλλει τα δεδομένα, και οι πλατφόρμες να τα απεικονίζουν. Αυτό οδήγησε σε χρήσιμα συμπεράσματα τα οποία επεξηγήθηκαν σε ξεχωριστό υποκεφάλαιο. Τέλος, το κεφάλαιο Μελλοντική εργασία είναι μια πρώτη ευκαιρία για τον αναγνώστη να κατανοήσει το μέγεθος του όρου Διαδίκτυο των Πραγμάτων και πως εξελίσσεται συνεχώς, αλλά και πού οδηγεί η συνεχής αυτή εξέλιξη. Από την αύξηση των δυνατοτήτων της παραγωγής σε βιομηχανικό επίπεδο, μέχρι και την διευκόλυνση της καθημερινότητας του ανθρώπου μέσα σε ένα «έξυπνο» σπίτι.

5.2 Μελλοντική Εργασία

Στην παρούσα εργασία, έγινε μελέτη για τη σύνδεση και την επικοινωνία αισθητήρα με πλατφόρμα υπολογιστικού νέφους, σε επίπεδο όμως εισαγωγικό στην τεχνολογία IoT και

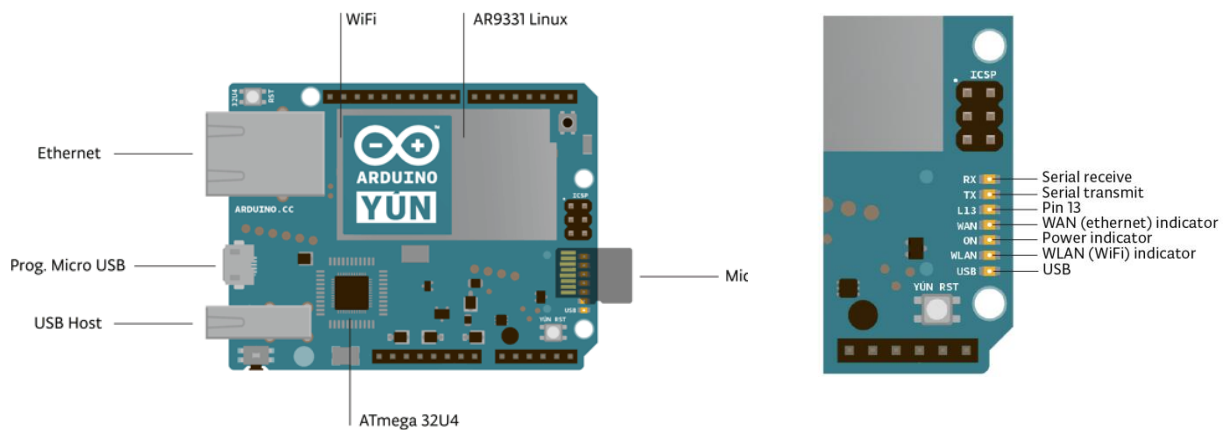
υπηρεσιών νέφους. Χρησιμοποιήθηκε μια μοναδική συσκευή και ένας απλός αισθητήρας μέτρησης θερμοκρασίας, δηλαδή ο όγκος εργασίας κρίνεται διαχειρίσιμος από έναν χρήστη – διαχειριστή. Σκοπός αυτού του κεφαλαίου είναι να παρουσιάσει πώς θα μπορούσε να επεκταθεί σε ευρύτερη κλίμακα αυτή η εργασία, είτε με τη χρήση διαφορετικού/αναβαθμισμένου εξοπλισμού, είτε με τη χρήση διαφορετικής πλατφόρμας νέφους για τη διαχείριση των δεδομένων. Η χρήση λύσεων IoT εκτός από την πειραματική χρήση για την παρούσα εργασία και για εκπαιδευτικό σκοπό, έχει επιρροή και σημαντική παρουσία παγκοσμίως, από την καθημερινή χρήση στη ζωή των ανθρώπων μέχρι και τη βιομηχανία, την ασφάλεια, την επικοινωνία. Θα γίνει αναφορά στον όρο Βιομηχανικό Διαδίκτυο των Πραγμάτων, μαζί με παραδείγματα ως προς την εφαρμογή του. Θα παρουσιαστεί η συμβολή του IoT στη δημιουργία ενός «έξυπνου» σπιτιού, που θα περιέχει διαφορετικούς αισθητήρες οι οποίοι θα συμβάλλουν στην ασφάλεια και τη διαχείριση των ηλεκτρικών συστημάτων του. Για την εξέλιξη και την επέκταση του πρακτικού μέρους της εργασίας θα χρειαζόταν η αναβάθμιση του κάθε μέρους του. Όπως παρουσιάστηκε στο κεφάλαιο 3, τα βασικά μέρη είναι ο Εξοπλισμός (Hardware), το Λογισμικό και η Πλατφόρμα – Υπηρεσία. Πιθανόν, ανάλογα με το νέο στόχο που τίθεται, να χρειαστεί αναβάθμιση σε όλα αυτά τα μέρη ή μόνο σε κάποια π.χ. να χρειαστεί μόνο αλλαγή του εξοπλισμού με πιο σύγχρονο, ενώ η πλατφόρμα να παραμείνει η ίδια. Παρακάτω παρουσιάζεται μια πιθανή αλλαγή εξοπλισμού, ενώ για τις διάφορες πλατφόρμες που είναι διαθέσιμες έχει γίνει αναφορά στο Θεωρητικό Μέρος της εργασίας.

5.2.1 Εξοπλισμός (Hardware)

Μια αλλαγή στον εξοπλισμό θα σημαίνει και αγορά κάποιου προϊόντος της οικογένειας Arduino με επιπρόσθετες λειτουργίες, και με χρονολογία κατασκευής μεταγενέστερης του Arduino Uno. Οι επιπρόσθετες δυνατότητες του εξοπλισμού μπορούν να διευκολύνουν στην επικοινωνία με άλλες συσκευές, στη σύνδεση στο Διαδίκτυο, στην τροφοδοσία, στην χωρητικότητα των δεδομένων που μπορούν να αποθηκευτούν, στο λογισμικό που μπορεί να υποστηριχθεί κ.α.. Εκτός από τα διάφορα boards υπάρχει

μεγάλη ποικιλία σε shields²⁵ που βρίσκονται διαθέσιμα στην αγορά και μπορούν να επεκτείνουν τις δυνατότητες των συμβατών boards. Ακολουθούν κάποια παραδείγματα:

5.2.2 Arduino Yun²⁶



Εικόνα 5.1: Arduino Yun

Το Arduino Yun διαθέτει:

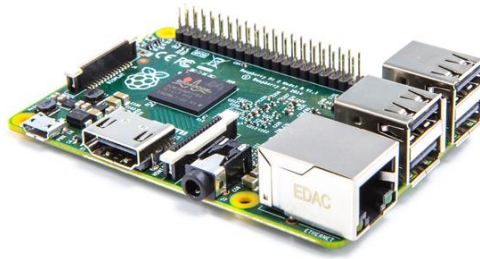
- Υποδοχή για κάρτα SD
- Θύρα Ethernet
- Θύρα USB-Host Connector
- Θύρα Micro USB (λειτουργεί ως τροφοδοσία του Arduino)
- Πομπό Wi-Fi

Το Arduino Yun διαθέτει δύο επεξεργαστές, τον 32U4 και τον AR9331, Ο κύριος επεξεργαστής του είναι ο 32U4 ο οποίος δεν συνδέεται φυσικά με τις θύρες Ethernet, USB-Host Connector και το Wi-Fi. Αυτή την δουλειά την κάνει ο επεξεργαστής AR9331 ο οποίος βρίσκεται σε επικοινωνία με τον 32U4. Οι δύο επεξεργαστές και το Wi-Fi διαθέτουν δικά τους κουμπιά επαννεκίνησης (reset buttons). Επίσης διαθέτει αρκετά LEDs όπως της σύνδεσης του Wi-Fi, του USB, του Ethernet και της τροφοδοσίας όπως φαίνεται στις παραπάνω εικόνες.

²⁵ <https://www.arduino.cc/en/Main/ArduinoShields>

²⁶ <https://www.arduino.cc/en/Main/ArduinoBoardYun>

5.2.3 Raspberry Pi 2²⁷



Εικόνα 5.2: Raspberry Pi 2

Το Raspberry Pi 2 διαθέτει:

- 4 θύρες USB
- Θύρα Ethernet
- Θύρα HDMI
- Θύρα Jack
- Θύρα κάρτας micro-SD

Διαθέτει έναν τετραπύρηνο επεξεργαστή τον ARM Cortex-A7 στα 900MHz, με διαθέσιμη RAM χωρητικότητας 1GB.

5.2.4 Raspberry Pi 3²⁸



Εικόνα 5.3: Raspberry Pi 3

Το Raspberry Pi 3 διαθέτει:

- 4 θύρες USB 2.0
- Θύρα Ethernet
- Θύρα HDMI
- Θύρα Jack
- Θύρα κάρτας micro-SD

²⁷ <https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>

²⁸ <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>

- Θύρα micro USB charger (χρησιμοποιείται ως τροφοδοσία)
- Wi-Fi
- Bluetooth

Διαθέτει έναν τετραπύρηνο επεξεργαστή τον 64-bit Broadcom BCM2837 ARM v8 στα 1.2GHZ, με διαθέσιμη RAM χωρητικότητας 1GB.

5.2.5 Βιομηχανικό Διαδίκτυο των Πραγμάτων [38]

Το **Βιομηχανικό Διαδίκτυο των Πραγμάτων** (Industrial Internet of Things - IIoT) είναι μια νέα τάση με σημαντικές επιπτώσεις για την παγκόσμια οικονομία. Εκτείνεται σε βιομηχανίες που αντιπροσωπεύουν το 62 τοις εκατό του ακαθάριστου εγχώριου προϊόντος (ΑΕΠ), συμπεριλαμβανομένου των ορυχείων, της γεωργίας, του πετρελαίου και του φυσικού αερίου, καθώς και τις επιχειρήσεις κοινής ωφέλειας. Περιλαμβάνει, επίσης εταιρείες που εξαρτώνται από φυσικά αγαθά για τη διεξαγωγή των επιχειρήσεων, όπως οργανώσεις που λειτουργούν νοσοκομεία, αποθήκες και λιμάνια ή που προσφέρουν μεταφορές, logistics και υπηρεσίες υγειονομικής περίθαλψης. Η **λειτουργική αποδοτικότητα** είναι ένα από τα βασικά κλειδιά του IIoT, και οι πρώτες εφαρμογές έχουν επικεντρωθεί σε αυτά τα οφέλη. Με την εισαγωγή της αυτοματοποίησης και των πιο ευέλικτων τεχνικών παραγωγής, οι κατασκευαστές για παράδειγμα, θα μπορούσαν να ενισχύσουν την παραγωγικότητά τους κατά 30 τοις εκατό. Το IIoT εστιάζει στην **προληπτική συντήρηση του εξοπλισμού**, εξοικονομώντας έως και 12 τοις εκατό στις προγραμματισμένες επισκευές, μειώνοντας το συνολικό κόστος συντήρησης μέχρι 30 τοις εκατό και εξαλείφοντας βλάβες έως 70 τοις εκατό. Για παράδειγμα, η Thames Water, η μεγαλύτερη πάροχος υπηρεσιών ύδρευσης και αποχέτευσης στο Ηνωμένο Βασίλειο, χρησιμοποιεί αισθητήρες, ανάλυση στοιχείων και δεδομένων σε πραγματικό χρόνο για να βοηθήσει την εταιρεία να προβλέψει βλάβες του εξοπλισμού και να ανταποκριθεί πιο γρήγορα σε κρίσιμες καταστάσεις, όπως διαρροές ή δυσμενείς καιρικές συνθήκες. Παρά το γεγονός ότι οι εταιρείες αρχίζουν να αξιοποιούν το IIoT ως επιχειρησιακή στρατηγική για αποτελεσματικότητα, προσφέρει επίσης την ευκαιρία σε εταιρείες παραγωγής για την **εισαγωγή νέων ψηφιακών προϊόντων και υπηρεσιών**. Επί της ουσίας το IIoT βοηθάει ώστε να αποφευχθούν προσωρινές ή και μόνιμες διακοπές λειτουργίας, αυξάνοντας έτσι την απόδοση της παραγωγής. Για παράδειγμα, η Apache Corporation, μια εταιρεία παραγωγής πετρελαίου και φυσικού

αερίου, χρησιμοποιεί αυτή την προσέγγιση για την πρόβλεψη αποτυχιών άντλησης πετρελαίου, ώστε να βοηθήσει στην ελαχιστοποίηση της απώλειας παραγωγής. Στελέχη της Apache υποστηρίζουν ότι αν η παγκόσμια βιομηχανία πετρελαίου βελτιώσει την απόδοση της άντλησης ακόμη και 1 τοις εκατό, τότε θα αυξηθεί η παραγωγή πετρελαίου κατά μισό εκατομμύριο βαρέλια την ημέρα και θα αποφέρει στην βιομηχανία επιπλέον 19 δισεκατομμύρια δολάρια τον χρόνο. Η Michelin κατασκεύασε ένα οικοσύστημα από εμπειρογνώμονες για την παροχή νέων υπηρεσιών. Η υπηρεσία χρησιμοποιεί υψηλής τεχνολογίας και υψηλής αφής αισθητήρες για να μειώσει το κόστος των καυσίμων στον στόλο των φορτηγών της. Οι αισθητήρες τοποθετούνται στο εσωτερικό των οχημάτων και συλλέγουν δεδομένα σχετικά με την κατανάλωση καυσίμου, της πίεσης των ελαστικών, τη θερμοκρασία, την ταχύτητα και την τοποθεσία. Αυτά τα δεδομένα στη συνέχεια μεταδίδονται σε μια υπηρεσία cloud της Michelin, και στη συνέχεια οι εμπειρογνώμονες αναλύουν τα δεδομένα και διατυπώνουν συστάσεις προς τους οδηγούς των φορτηγών για το πώς να χρησιμοποιούν λιγότερα καύσιμα κατά την οδήγηση. Το αποτέλεσμα είναι οι διαχειριστές του στόλου των φορτηγών να μπορούν να εξοικονομούν έως και 2 λίτρα καυσίμου για κάθε 100 διανυθέντα χιλιόμετρα. Με την ίδια λογική, οποιαδήποτε εταιρεία η οποία διαθέτει στόλο φορτηγών (φαρμακευτικές εταιρείες, εταιρείες τροφίμων κλπ), χρησιμοποιώντας διάφορους αισθητήρες ανάλογα με τις ανάγκες της, θα μπορούσε να εξοικονομήσει μεγάλα κέρδη. Για παράδειγμα μια φαρμακευτική εταιρεία η οποία μεταφέρει τα φαρμακά της με φορτηγά-ψυγεία, θα μπορούσε με την χρήση αισθητηρών (θερμοκρασίας,υγρασίας) να ελέγχει την θερμοκρασία στο ψυγείο και σε περίπτωση βλάβης ή μεταβολής της θερμοκρασίας να ενημερώνεται σε πραγματικό χρόνο ο διαχειριστής. Με αυτή την διαδικασία το πρόβλημα θα αντιμετωπιστεί εγκαίρως και κάποια φάρμακα τα οποία είναι ευαίσθητα σε μεταβολές της θερμοκρασίας, δεν θα καταστραφούν αποφέροντας ζημία για την εταιρεία.Το ίδιο συμβαίνει και με τις γαλακτοκομικές εταιρείες,τις εταιρείες κρεάτων και λαχανικών οι οποίες θέλουν να διαφυλάξουν την ποιότητα των προϊόντων τους. Σε αυτή την εργασία παρουσιάστηκε η σύνδεση μιας μοναδικής συσκευής στην πλατφόρμα διαχείρισης δεδομένων, αλλά στο συγκεκριμένο παράδειγμα θα υπήρχαν παραπάνω συσκευές (π.χ. μια συσκευή ανά όχημα). Αυτό σημαίνει πως και τα δεδομένα που θα ανταλλάσσονται θα είναι πολλαπλάσια των συσκευών που χρησιμοποιούνται. Η διαδικασία που χρησιμοποιήθηκε στα κεφάλαια 3,4 θα χρειαζόταν να επαναληφθεί -

έστω και μερικώς – για την διαχείριση των συσκευών και των δεδομένων, ή να γίνουν τροποποιήσεις στον κώδικα και στην παραμετροποίηση από πλευράς πλατφόρμας σε περίπτωση αντικατάστασης μιας συσκευής (π.χ. λόγω φυσικής βλάβης). Σε περίπτωση ανάγκης διαχείρισης μεγάλου πλήθους συσκευών, άρα και μεγάλου όγκου δεδομένων, θα παρουσιαζόταν η ανάγκη εξεύρεσης μιας είτε απλούστερης διαδικασίας διαχείρισης για τις πλατφόρμες που αναλύθηκαν, είτε έρευνας αγοράς για μια νέα πλατφόρμα που να καλυπτει αυτή την ανάγκη.

5.2.5.1 Αισθητήρας για διαρροή φυσικού αερίου.

Τα τελευταία δεκαπέντε χρόνια που η χρήση του φυσικού αερίου είναι όλο και πιο εκτεταμένη, οι εκρήξεις από διαρροές είναι υπεύθυνες για εκατοντάδες θανάτους. Η αντικατάσταση των παλαιών αγωγών φυσικού αερίου κοστίζει υπέρογκα ποσά, οπότε μια χρήσιμη εφαρμογή του internet of things θα ήταν η δημιουργία μιας εφαρμογής παρακολούθησης διαρροής του αερίου. Με αυτόν τον τρόπο θα προστατευθούν τυχούσες καταστροφές κτιρίων αλλά και τραυματισμοί ανθρώπων. Για την δημιουργία μιας τέτοιας εφαρμογής θα χρειαστεί ένας αισθητήρας παρακολούθησης διαρροής αερίου (MQ9 gas sensor²⁹), ένα Arduino Yun board καθώς και μια διαδικτυακή υπηρεσία όπου θα αποστέλλονται τα δεδομένα που συλλέγονται από τον αισθητήρα. Το Temboo³⁰ είναι μια υπηρεσία η οποία διαθέτει έναν οδηγό για την δημιουργία μιας εφαρμογής παρακολούθησης διαρροής αερίου.

5.2.5.2 Αισθητήρας παρακολούθησης υγρασίας φυτών.

Μια πολύ χρήσιμη εφαρμογή για κάποιον ο οποίος ασχολείται με την καλλιέργεια είναι η δημιουργία μιας εφαρμογής η οποία θα ενημερώνει για τα ποσοστά της υγρασίας στο χώμα των φυτών. Με την χρήση ενός αισθητήρα υγρασίας (SeeedStudio Grove - Moisture Sensor³¹), ενός Arduino Yun board και της διαδικτυακής υπηρεσίας Amazon Web Services³² είναι εφικτή η δημιουργία μιας τέτοιας εφαρμογής. Όταν ο ενδιαφερόμενος δεν θα μπορεί να βρίσκεται στην περιοχή της καλλιέργειας ώστε να ελέγξει ο ίδιος την υγρασία, η εφαρμογή αυτή θα του δίνει την δυνατότητα να ενημερώνεται από απόσταση (π.χ μέσω email) σε πραγματικό χρόνο, έτσι ώστε να

²⁹ https://www.pololu.com/file/download/MQ9.pdf?file_id=0J314

³⁰ <https://temboo.com/>

³¹ <https://www.seeedstudio.com/Grove-Moisture-Sensor-p-955.html>

³² <https://aws.amazon.com/>

προβεί στις κατάλληλες ενέργειες για να ποτιστούν τα φυτά. Επίσης χρησιμοποιώντας καλύτερο και πιο πολύπλοκο εξοπλισμό θα μπορούσε να χρησιμοποιηθεί σε μεγαλύτερης κλίμακας καλλιέργειες και θερμοκήπια. Με την χρήση του IIoT θα δημιουργηθεί **νέο εργατικό δυναμικό** το οποίο θα αντικαταστήσει ορισμένα καθήκοντα και τις επαναλαμβανόμενες εργασίες, οι οποίες μέχρι πρότινος αντιστέκονταν στον αυτοματισμό. Οι εταιρείες θα πρέπει να ψάξουν για άτομα με δεξιότητες στον τομέα της επιστήμης των δεδομένων, της ανάπτυξης λογισμικού, των μηχανικών ηλεκτρονικών υπολογιστικών συστημάτων, του μάρκετινγκ και των πωλήσεων. Επίσης οι εταιρείες που θα αναπτύξουν και θα προσφέρουν λύσεις IIoT θα πρέπει και να τις υποστηρίξουν. Αυτό σημαίνει να τις εξελίσσουν συνεχώς ώστε να μπορούν να είναι όσο το δυνατόν φιλικές προς το χρήστη. Θα χρειαστούν μηχανικοί για να βρεθούν λύσεις ώστε οι υπηρεσίες να συνεισφέρουν στην παραγωγικότητα των εργαζομένων και στην καλύτερη εξυπηρέτηση των πελατών.



Εικόνα 5.4: Αισθητήρας υγρασίας

5.2.6 Έξυπνο Σπίτι

Η φράση “έξυπνο σπίτι” χρησιμοποιείται για οποιαδήποτε οικία, ενσωματώνει –σε μικρότερο ή μεγαλύτερο βαθμό– τη δυνατότητα ρύθμισης ορισμένων παραμέτρων. Το εγκατεστημένο σύστημα που χρησιμοποιείται, έχει τη δυνατότητα να ρυθμίζει αυτόματα το οικιακό περιβάλλον, σύμφωνα με τις επιθυμίες του ιδιοκτήτη. Για να επιτευχθεί ο στόχος αυτός, θα πρέπει να βρεθεί κάποιος τρόπος ώστε το σύνολο των ηλεκτρικών και ηλεκτρονικών συσκευών να επικοινωνούν μεταξύ τους, λαμβάνοντας και αποστέλλοντας εντολές. Η βασική ιδέα πίσω από το έξυπνο σπίτι και τον κτιριακό αυτοματισμό γενικότερα είναι να τροφοδοτήσουμε μια κατοικία με αισθητήρες και συστήματα ελέγχου και σύμφωνα με αυτά να έχουμε παροχή θερμότητας, κλιματισμού, φωτισμού

Αριστείδης Σ. Βυζανιάρης - Ευάγγελος Γ. Φαραντάτος

και πολλών άλλων υπηρεσιών. Προσαρμόζοντας περαιτέρω τους μηχανισμούς του σπιτιού στις ανάγκες του κατόχου του, το έξυπνο σπίτι μπορεί να παρέχει ένα πιο ασφαλές, πιο άνετο και πιο οικονομικό κατάλυμα. Συγκεκριμένα, ορισμένες από τις λειτουργίες ενός σπιτιού που μπορούν να αυτοματοποιηθούν με στόχο να γίνει εύχρηστο, άνετο και γρήγορο είναι:

- Σύστημα ασφάλειας
- Σύστημα παρακολούθησης
- Σύστημα φωτισμού
- Σύστημα ρύθμισης θερμοκρασίας και υγρασίας
- Σύστημα ανίχνευσης διαρροής αερίου και πλημμύρας
- Σύστημα πυρασφάλειας

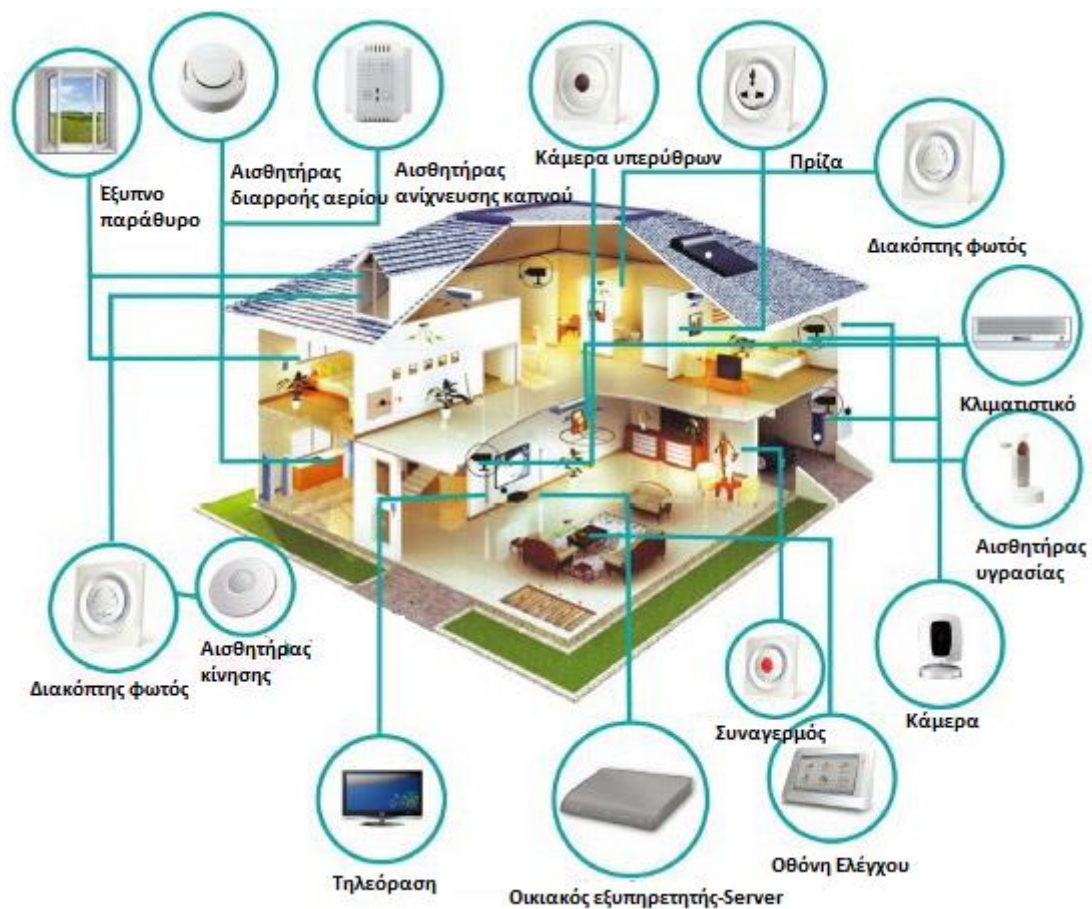
Για να μπορέσει κάποιος να αυτοματοποιήσει τις παραπάνω λειτουργίες του σπιτιού, θα πρέπει να χρησιμοποιήσει κάποιο σύστημα όπως είναι το σύστημα **bus**. Σε γενικές γραμμές ως bus σύστημα χαρακτηρίζεται το σύνολο των στοιχείων εκείνων που επιτρέπουν την συλλογή, μετάδοση και εκτέλεση εντολών ή πληροφοριών με τη χρήση ενός μόνο ζεύγους αγωγών. Το ζεύγος αυτό αποτελεί το καλώδιο του bus (δίαυλος δεδομένων) και μέσα από αυτό διέρχονται όλες οι διαθέσιμες πληροφορίες. Συνεπώς όπου υπάρχει η καλωδίωση bus, είναι διαθέσιμες όλες οι πληροφορίες που κυκλοφορούν σε αυτήν. [39] Τα στοιχεία που μπορεί να χρησιμοποιηθούν για την υλοποίηση ενός συστήματος bus είναι τα ακόλουθα:

- Κεντρική Μονάδα (υποχρεωτικά τουλάχιστον μία)
- Μονάδες Εισόδου
- Μονάδες Εξόδου
- Αισθητήρες
- Ηλεκτρονικοί Διακόπτες
- Κάμερες Ασφαλείας

- Ηλεκτρονικοί Διακόπτες

Πάνω σε αυτό το σύστημα μπορούν να συνδεθούν διάφοροι αισθητήρες, με τους οποίους θα αυτοματοποιήσουμε διάφορες λειτουργίες του σπιτιού. Οι αισθητήρες μετατρέπουν σήματα του φυσικού περιβάλλοντος σε σήματα που μπορούν να μεταφερθούν μέσω του bus (π.χ. θερμοκρασία, φωτεινότητα, κίνηση, κλπ.). Υπάρχουν πολλές κατηγορίες αισθητήρων. Κάποιες από αυτές είναι:

- Αισθητήρες κίνησης-παρουσίας
- Αισθητήρες θερμοκρασίας-υγρασίας
- Αισθητήρες νερού-καπνού
- Αισθητήρες φωτεινότητας
- Φωτοηλεκτρικοί αισθητήρες



Εικόνα 5.5: Έξυπνο Σπίτι

ΑΝΑΦΟΡΕΣ

- [1] Peter Mell - Timothy Grance (2011). U.S. National Institute of Standards and Technology. *The NIST Definition of Cloud Computing*. Διαθέσιμο: <http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf>
- [2] Secovia.eu (2013). South East Europe, Transnational Cooperation Programme. *A brief introduction to Cloud Computing*. Διαθέσιμο: <http://www.secovia.eu/news/secovia-overview-of-cloud-computing>
- [3] John Waters (2007). *Virtualization Definition and Solutions*. Διαθέσιμο: <http://www.cio.com/article/2439494/virtualization/virtualization-definition-and-solutions.html>
- [4] Ιωαννίδου Χ. (2013). *Τεχνολογίες Υπολογιστικού Νέφους & Προστασία Προσωπικών Δεδομένων*, Διπλωματική Εργασία, Διατμηματικό Πρόγραμμα Μεταπτυχιακών Σπουδών «Πληροφορική και Διοίκηση» του Αριστοτελείου Πανεπιστημίου Θεσσαλονίκης. Διαθέσιμο: <http://docplayer.gr/11204664-Tehnologies-y-pologistikoy-nefoys-prostasia-prosopikon-dedomenon.html>
- [5] Fiware.org (2014), *Ιστοσελίδα Υπηρεσίας*. Διαθέσιμο: <https://www.fiware.org/about-us/>
- [6] Intellicloud (2014), *Επίσημη Ιστοσελίδα*. Διαθέσιμο: <http://cloud.intellicloud.tuc.gr/index.php>
- [7] Openstack.org (2016), *Τι είναι το Openstack*; Διαθέσιμο: <http://www.openstack.org/software/>
- [8] Amazon Web Services (2016), *Τι είναι το AWS*. Διαθέσιμο: <https://aws.amazon.com/what-is-aws/>
- [9] IBM (2016), *Ιστοσελίδα Υπηρεσίας*. Διαθέσιμο: <http://www.ibm.com/cloud-computing/bluemix/>
- [10] Frontiercities.freshdesk (2014), *Τι είναι οι Generic Enablers*; Διαθέσιμο: <https://frontiercities.freshdesk.com/support/solutions/articles/5000507301-what-are-generic-enablers-ges->
- [11] Fiware Catalogue (2015), *Κατάλογος Διαθέσιμων GEs FIWARE*. Διαθέσιμο: <http://catalogue.fiware.org/>

- [12] Perrey R. - Lycett M.(2014). Brunel University Uxbridge. *Service-oriented architecture*. Διαθέσιμο: http://ieeexplore.ieee.org/xpl/abstractAuthors.jsp?tp=&arnumber=1210138&url=http%3A%2F%2Fieeexplore.ieee.org%2Fexpls%2Fabs_all.jsp%3Farnumber%3D1210138
- [13] Douglas K. Barry (2016). *Service Architecture*. Διαθέσιμο: <http://www.service-architecture.com/articles/web-services/service-oriented-architecture-soa-definition.html>
- [14] Tom Fuller (2005). *The Good, the Bad, and the Ugly of Service-Oriented Architecture*. Διαθέσιμο: <http://aspalliance.com/707-The-Good-the-Bad-and-the-Ugly-of-ServiceOriented-Architecture.4>
- [15] Opengroup.org (2016), *Χαρακτηριστικά Υπηρεσιοκεντρικής Αρχιτεκτονικής*. Διαθέσιμο: https://www.opengroup.org/soa/source-book/soa/soa_features.htm
- [16] Microsoft.com (2016), *Service Oriented Integration Pattern*. Διαθέσιμο: <https://msdn.microsoft.com/en-us/library/bb833022.aspx#Introduction-to-SOA>
- [17] w3.org (2016), *The Original HTTP as defined in 1991*. Διαθέσιμο: <https://www.w3.org/Protocols/HTTP/AsImplemented.html>
- [18] Michael Jakl (2010). *REST Representational State Transfer*. Διαθέσιμο: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.97.7334&rep=rep1&type=pdf>
- [19] Κολωνιάρη Μ., Χρυσάδκου Ειρ. (2008). *Διεπαφές (API – Application Program Interface)*, Ιόνιο Πανεπιστήμιο Τμήμα Αρχιτεκτονικής – Βιβλιοθηκονομίας Πρόγραμμα Μεταπτυχιακών Σπουδών στην Επιστήμη της Πληροφορίας «Διοίκηση & Οργάνωση Βιβλιοθηκών με έμφαση στις Νέες Τεχνολογίες της Πληροφορίας» Διαθέσιμο: http://users.ionio.gr/~toraki/infotech_met/infotech_met0708_ergasies/3_api_chrisadakou_koloniari.pdf
- [20] Kyle Brown (2014). *CTO for Cloud and Emerging Technologies*. Διαθέσιμο: <http://www.ibm.com/developerworks/cloud/library/cl-bluemix-arduino-iot2/>

- [21] Mqtt.org (2016), *Frequently Asked Questions*. Διαθέσιμο: <http://mqtt.org/faq>
- [22] Json.org (2016), *Introducing JSON*. Διαθέσιμο: <http://www.json.org/>
- [23] Ivisinfo.gr (2016), *Virtualization Technology*. Διαθέσιμο: <http://www.ivisinfo.gr/articles/virtualization-technology.php>
- [24] Gartner.com (2016), *Newsroom Press Release*. Διαθέσιμο: <http://www.gartner.com/newsroom/id/3165317>
- [25] Sciencedirect.com (2013), *Internet of Things (IoT): A vision, architectural elements, and future directions*. Διαθέσιμο: <http://www.sciencedirect.com/science/article/pii/S0167739X13000241>
- [26] samsung.com (2016), *The Internet of Things Smart Home*. Διαθέσιμο: <http://www.samsung.com/gr/discover/new/the-internet-of-things-smart-home/>
- [27] Ovidiu Vermesan-Peter Fries (2013). *Internet of Things: Converging Technologies for Smart Enviroments and Integrated Ecosystems*. Διαθέσιμο: [http://www.internet-of-things-research.eu/pdf/Converging Technologies for Smart Environments and Integrated Ecosystems IERC Book Open Access 2013.pdf](http://www.internet-of-things-research.eu/pdf/Converging_Technologies_for_Smart_Environments_and_Integrated_Ecosystems_IERC_Book_Open_Access_2013.pdf)
- [28] Ovidiu Vermesan-Peter Fries (2014). *Internet of Things-Research and Innovation to Market Deployment*. Διαθέσιμο: [http://www.internet-of-things-research.eu/pdf/IoT-From%20Research%20and%20Innovation%20to%20Market%20Deployment_IERC_Cluster eBook 978-87-93102-95-8 P.pdf](http://www.internet-of-things-research.eu/pdf/IoT-From%20Research%20and%20Innovation%20to%20Market%20Deployment_IERC_Cluster_eBook_978-87-93102-95-8_P.pdf)
- [29] Holler J., Tsiatsis V., Dr. Mulligan C., Karnouskos S., Avesand S., Boyle D. (2014). *From Machine-to Machine to the Internet of Things*
- [30] Ibm.com (2016), *IBM Connects "Internet of Things" to the Enterprise*. Διαθέσιμο: <http://www-03.ibm.com/press/us/en/pressrelease/46453.wss>
- [31] Arduino Board (2016), *Product Overview*, Διαθέσιμο: <https://www.arduino.cc/en/Main/ArduinoBoardUno>
- [32] Γαροφαλάκη Ζ. (2015), *Έλεγχος Ρευματοδότησης Δικτυακών Μονάδων μέσω Συσκευής Arduino*, Μεταπτυχιακή Διατριβή, Τμήμα Η/Υ Συστημάτων ΑΕΙ Πειραιά Τεχνολογικού Τομέα.
- [33] Ethernet Shield (2016), *Product Overview*, Διαθέσιμο: <https://www.arduino.cc/en/Main/ArduinoEthernetShield>
- [34] Έλληνας Ι. (2014), *Εφαρμοσμένα Πληροφοριακά Συστήματα*, Μεταπτυχιακό Πρόγραμμα Σπουδών, Τμήμα Η/Υ Συστημάτων ΑΕΙ Πειραιά Τεχνολογικού Τομέα.

- [35] Arduino Software (2016), Official Download Website, Διαθέσιμο: <https://www.arduino.cc/en/Main/Software>
- [36] SPI Library Overview (2016), Arduino Learning Website, Διαθέσιμο: <https://www.arduino.cc/en/Reference/SPI>
- [37] Ethernet Library Overview (2016), Arduino Learning Website, Διαθέσιμο: <https://www.arduino.cc/en/Reference/Ethernet>
- [38] Accenture.com, (2016), *Driving Unconventional Growth through the Industrial Internet Of Things*, Διαθέσιμο: <https://www.accenture.com/us-en/labs-insight-industrial-internet-of-things>
- [39] Lelectronics.gr, (2016), *Τι είναι το BUS*, Διαθέσιμο: <http://www.lelectronics.gr/cctv/ti-einai-to-bus-p-205.html>