

Ατομική Διπλωματική Εργασία

**ΜΗΧΑΝΗ ΠΑΡΑΓΩΓΗΣ ΚΩΔΙΚΑ ΓΙΑ ΛΗΨΗ ΜΕΤΡΗΣΕΩΝ ΜΕΣΩ
ΠΛΑΤΦΟΡΜΩΝ INTERNET OF THINGS**

Ευάγγελος Αριστοδήμου

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2018

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μηχανή Παραγωγής Κώδικα για λήψη μετρήσεων μέσω πλατφόρμων Internet of Things

Ευάγγελος Αριστοδήμου

Επιβλέπων Καθηγητής

Γεωργία Καπιτσάκη

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου

Μάιος 2018

Ευχαριστίες

Θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες στην επιβλέπουσα καθηγήτρια μου Δρ. Γεωργία Καπιτσάκη για την εξαιρετική συνεργασία που είχαμε τα τελευταία δύο εξάμηνα, καθώς και για τη βοήθεια σε όλες τις φάσεις ανάπτυξης της παρούσας διπλωματικής εργασίας.

Επίσης θα ήθελα να εκφράσω ένα μεγάλο ευχαριστώ στους φίλους και στην οικογένεια μου για την ψυχολογική στήριξη που μου παρείχαν καθ' όλη τη χρονική διάρκεια εκπόνησης και ολοκλήρωσης αυτής της εργασίας.

Περίληψη

Στις μέρες μας, οι εφαρμογές γύρω από το πλαίσιο του Internet of Things (Διαδίκτυο των Πραγμάτων) έχουν τεράστια ανάπτυξη. Με τον όρο Internet of Things εννοούμε διάφορα αντικείμενα της καθημερινότητας, τα οποία χρησιμοποιούν αισθητήρες για να λαμβάνουν μετρήσεις από το περιβάλλον στο οποίο βρίσκονται, με σκοπό την αυτοματοποίηση διαδικασιών ή συλλογή πληροφοριών.

Με χρήση διαφόρων πλατφορμών μικροελεγκτών όπως είναι το Arduino και το Raspberry Pi διευκολύνεται η ανάπτυξη εφαρμογών αυτού του είδους. Οι μικροελεγκτές υποστηρίζουν ένα μεγάλο εύρος αισθητήρων όπως φωτεινότητας, θερμοκρασίας, υγρασίας, κίνησης και ήχου, και επιτρέπουν τη δημιουργία απλών αλλά και περίπλοκων συστημάτων για λήψη μετρήσεων από το περιβάλλον τους. Παρ' όλα αυτά, σήμερα υπάρχουν διάφοροι τύποι μικροελεγκτών και ο αριθμός των αισθητήρων που μπορούν να χρησιμοποιηθούν είναι μεγάλος. Έτσι, η χρήση τους γίνεται κάπως περίπλοκη, αφού απαιτείται μια συγκεκριμένη διαδικασία που πρέπει να ακολουθήσει κάποιος για να καταφέρει να επιτύχει αυτό που επιθυμεί. Η διαδικασία αυτή περιλαμβάνει τη συναρμολόγηση του κυκλώματος των αισθητήρων και τη συγγραφή κώδικα ο οποίος θα εγκατασταθεί στην πλατφόρμα μικροελεγκτή.

Στο πλαίσιο της παρούσας διπλωματικής εργασίας, δημιουργήθηκε ένας μηχανισμός παραγωγής κώδικα, ο οποίος αυτοματοποιεί την πιο πάνω διαδικασία, προσφέροντας στο χρήστη ένα πιο εύκολο τρόπο χρήσης μικροελεγκτών και αισθητήρων. Ο μηχανισμός αυτός, βάσει των επιλογών του χρήστη, παράγει ένα αρχείο κώδικα το οποίο μπορεί να εγκατασταθεί στο μικροελεγκτή για λήψη μετρήσεων, μαζί με οδηγίες συναρμολόγησης του κυκλώματος και βοηθητικά διαγράμματα.

Σε αυτή την αναφορά θα παρουσιαστεί η αρχιτεκτονική του συστήματος SensoMan και θα γίνει αναλυτική περιγραφή της μηχανής παραγωγής κώδικα.

Περιεχόμενα

Κεφάλαιο 1: Εισαγωγή.....	1
1.1 Κίνητρο Διπλωματικής Εργασίας	2
1.2 Ανασκόπηση συστήματος SensoMan	4
1.2.1 Γενική Περιγραφή	4
1.2.2 Οι φάσεις του συστήματος.....	4
1.2.3 Τεχνολογίες συστήματος.....	5
1.3 Δομή Διπλωματικής Εργασίας	7
Κεφάλαιο 2: Υπόβαθρο.....	9
2.1 Διαδίκτυο των Πραγμάτων	10
2.2 Πλατφόρμα μικροελεγκτή	11
2.3 Αισθητήρας	14
2.4 RESTful υπηρεσίες διαδικτύου	15
2.5 Σύστημα OpenHAB.....	17
Κεφάλαιο 3: Υλικό.....	20
3.1 Πλατφόρμες μικροελεγκτών.....	21
3.1.1 Πλατφόρμα μικροελεγκτή Arduino	21
3.1.2 Πλατφόρμα μικροελεγκτή Raspberry Pi.....	24
3.1.3 Πλατφόρμα μικροελεγκτή Beaglebone Black.....	27
3.1.4 Πλατφόρμα μικροελεγκτή Intel Galileo.....	29
3.2 Αισθητήρες.....	30
3.2.1 Αισθητήρας κίνησης PIR	30
3.2.2 Αισθητήρας φωτεινότητας LDR	31
3.2.3 Αισθητήρας θερμοκρασίας DS18B20	32
3.2.4 Αισθητήρας θερμοκρασίας TMP36	33
3.2.5 Αισθητήρας ήχου	34
3.2.6 Αισθητήρας υγρασίας FC-28.....	34
3.3 Επιπλέον εξαρτήματα	35
3.3.1 Wi-Fi Shield CC3000	36
3.3.2 Ethernet Shield	36
3.3.3 Wireless card και αντένες Wi-Fi	37
Κεφάλαιο 4: Σύστημα SensoMan	38
4.1 Αρχιτεκτονική Συστήματος	39
4.2 Βάση Δεδομένων	42
4.3 SensoMan RESTful υπηρεσία	46
4.4 Υποσυστήματα του SensoMan	50
4.4.1 Διαδικτυακή Εφαρμογή SensoMan.....	51
4.4.2 Κινητή Εφαρμογή SensoMan.....	56
4.4.3 Μηχανή Κανόνων	56
4.4.4 Κοινωνικό Δίκτυο	57
Κεφάλαιο 5: Μηχανή Παραγωγής Κώδικα	58
5.1 Περιγραφή	59
5.2 Διαδραστική Φόρμα Επιλογών	61

5.3	Συναρμολόγηση Αρχείου Κώδικα	65
5.3.1	Κώδικας για Arduino και Intel Galileo	69
5.3.2	Κώδικας για Raspberry Pi	74
5.3.3	Κώδικας για Beaglebone Black	77
5.4	Επιπλέον Λειτουργίες	79
5.4.1	Λίστα Υλικού	80
5.4.2	Οδηγίες συναρμολόγησης κυκλώματος – εγκατάστασης κώδικα.....	81
5.4.3	Βοηθητικά Διαγράμματα	82
5.4.4	Αποθήκευση αρχείων	84
Κεφάλαιο 6: Σενάρια Χρήσης και Αξιολόγηση Μηχανής Παραγωγής Κώδικα		86
6.1	Σενάρια Χρήσης.....	87
6.1.1	Πρώτο Σενάριο Χρήσης.....	87
6.1.2	Δεύτερο Σενάριο Χρήσης.....	95
6.2	Σύνοψη Σεναρίων.....	101
6.3	Αξιολόγηση.....	103
Κεφάλαιο 7: Συμπεράσματα και Μελλοντικές Επεκτάσεις.....		104
7.1	Συμπεράσματα	105
7.2	Μελλοντικές Επεκτάσεις.....	106
Βιβλιογραφία		108

Κεφάλαιο 1

Εισαγωγή

Εισαγωγή	1
1.1 Κίνητρο Διπλωματικής Εργασίας	2
1.2 Ανασκόπηση συστήματος SensoMan	4
1.2.1 Γενική Περιγραφή	4
1.2.2 Οι φάσεις του συστήματος	4
1.2.3 Τεχνολογίες συστήματος	5
1.3 Δομή Διπλωματικής Εργασίας	7

1.1 Κίνητρο Διπλωματικής Εργασίας

Στις μέρες μας, μέσω της ραγδαίας ανάπτυξης της τεχνολογίας, ο άνθρωπος κατάφερε να φέρει στη επιφάνεια επιτεύγματα που φάνταζαν αδύνατα στο παρελθόν. Συνεχώς, μέσω καινοτομικών ιδεών, παράγονται νέα προϊόντα και υπηρεσίες που στόχο έχουν να κάνουν τη ζωή του ανθρώπου πιο εύκολη και πιο ευχάριστη. Οι έννοιες «έξυπνα σπίτια» και «έξυπνες συσκευές» είναι όροι οι οποίοι ακούγονται συχνά σήμερα και αποτελούν μέρος της ζωής μας.

Με τον όρο έξυπνο σπίτι εννοούμε το σπίτι το οποίο περιλαμβάνει έξυπνες συσκευές οι οποίες με χρήση αισθητήρων λαμβάνουν πληροφορίες από το περιβάλλον στο οποίο βρίσκονται και επικοινωνούν μεταξύ τους μέσω του Δικτύου των Πραγμάτων (Internet of Things), δίνοντας την δυνατότητα απομακρυσμένου ελέγχου τους από το χρήστη, συλλογή μετρήσεων για εξαγωγή συμπερασμάτων και αυτοματοποίηση διαδικασιών χωρίς τη συγκατάθεση του ανθρώπου.

Ο αριθμός των έξυπνων συσκευών αυξάνεται ολοένα και περισσότερο στις μέρες μας και οι δυνατότητες τους είναι αμέτρητες αφού μπορούν να εμφανιστούν παντού σε διάφορους χώρους μέσα σε ένα σπίτι. Μερικά παραδείγματα τέτοιου είδους συσκευών αποτελούν τα έξυπνα ψυγεία, οι καφετιέρες, τα πλυντήρια, οι θερμοστάτες, οι φούρνοι καθώς και άλλες ηλεκτρικές συσκευές. Οι συσκευές αυτές μπορούν να συνδεθούν στο διαδίκτυο και ο χρήστης μπορεί να λαμβάνει πληροφορίες για διάφορες μετρήσεις και να τις θέτει σε λειτουργία μέσω εφαρμογής στο κινητό του.

Παράλληλα, η ευρεία εξάπλωση και διασημότητα πλατφόρμων μικροελεγκτών όπως είναι το Arduino, το Raspberry Pi, το BeagleBone Black και το Intel Galileo επιτρέπουν την δημιουργία κυκλωμάτων αισθητήρων, τα οποία μπορούν να χρησιμοποιηθούν σε ποικίλες εφαρμογές είτε για ερευνητικούς είτε για πρακτικούς σκοπούς. Υποστηρίζουν ένα μεγάλο φάσμα αισθητήρων οι οποίοι μπορούν να λάβουν μετρήσεις και να δώσουν μια σφαιρική εικόνα του περιβάλλοντος.

Υπάρχουν διάφοροι αισθητήρες οι οποίοι διαφέρουν στο μέγεθος τους, στο τύπο τους και στην ακρίβεια των μετρήσεων των οποίων λαμβάνουν. Βασικοί τύποι αισθητήρων αποτελούν οι αισθητήρες κίνησης, φωτεινότητας, θερμοκρασίας, ήχου και υγρασίας. Οι αισθητήρες αυτοί, μπορούν να τοποθετηθούν σε πλατφόρμες μικροελεγκτών και να αποτελέσουν μέρος απλών αλλά και πολύπλοκων συστημάτων.

Παρ' όλα αυτά, η συναρμολόγηση και λειτουργία τέτοιου είδους συστημάτων μπορεί να αποτελέσει μια χρονοβόρα αλλά και πολύπλοκη διαδικασία λόγω του μεγάλου αριθμού επιλογών μικροελεγκτών και αισθητήρων. Η διαδικασία αυτή αποτελείται από την γνώση του υλικού το οποίο χρησιμοποιείται, τη συναρμολόγηση του κυκλώματος των αισθητήρων και τη δημιουργία κώδικα, ο οποίος θα αποτελέσει την γέφυρα επικοινωνίας μεταξύ του μικροελεγκτή και των αισθητήρων.

Στην ουσία, ο κώδικας αυτός δίνει την δυνατότητα στο μικροελεγκτή να γνωρίζει σε ποιες εισδοχές (pins) είναι συνδεδεμένος ο κάθε αισθητήρας, να γνωρίζει το είδος του (αναλογικός ή ψηφιακός) και να μπορεί να τον ενεργοποιεί ανά τακτά χρονικά διαστήματα ώστε να λαμβάνει μετρήσεις από το περιβάλλον και τέλος να εμφανίζει αυτές τις μετρήσεις στο χρήστη με τρόπο που τις καταλαβαίνει.

Έτσι, λόγω του ότι αυτή η διαδικασία είναι χρονοβόρα και διαφέρει ανάλογα με το υλικό που επιλέγεται να χρησιμοποιηθεί, αποφασίστηκε στα πλαίσια της παρούσας διπλωματικής εργασίας να δημιουργηθεί ένα επιπλέον κομμάτι στο σύστημα SensoMan το οποίο να αυτοματοποιεί την πιο πάνω διαδικασία. Συγκεκριμένα, δημιουργήθηκε ένας μηχανισμός παραγωγής κώδικα, ο οποίος επιτρέπει στο χρήστη μέσω απλής διαδικασίας να επιλέξει το υλικό που θέλει να χρησιμοποιήσει για τη συναρμολόγηση του κυκλώματος του και ο μηχανισμός παράγει ένα αρχείο κώδικα το οποίο μπορεί να εγκατασταθεί στο μικροελεγκτή και να κάνει τις λειτουργίες που πρέπει. Παράλληλα, ο μηχανισμός εμφανίζει στο χρήστη το υλικό το οποίο θα χρειαστεί για την δημιουργία του κυκλώματος, οδηγίες συναρμολόγησης και εγκατάστασης, καθώς και βοηθητικά διαγράμματα. Ο μηχανισμός παραγωγής κώδικα θα επεξηγηθεί αναλυτικότερα σε επόμενο κεφάλαιο.

1.2 Ανασκόπηση συστήματος SensoMan

1.2.1 Γενική Περιγραφή

Το σύστημα SensoMan αποτελεί μια διαδικτυακή εφαρμογή η οποία δημιουργήθηκε στα πλαίσια παλιών διπλωματικών εργασιών και συνεχίζει να αναβαθμίζεται μέχρι σήμερα. Το σύστημα αυτό έχει ως στόχο να δώσει στους χρήστες του ένα ολοκληρωμένο τρόπο διαχείρισης πλατφορμών μικροελεγκτών, αισθητήρων και μετρήσεων που λαμβάνονται από αυτούς.

Συγκεκριμένα, ο κάθε χρήστης μπορεί να έχει στο σύστημα το δικό του λογαριασμό και να διαχειρίζεται τους δικούς του μικροελεγκτές οι οποίοι μπορεί να είναι τύπου Arduino, Raspberry Pi, BeagleBone Black και Intel Galileo. Οι μικροελεγκτές αυτοί λόγω του μεγέθους και της φορητότητας τους μπορούν να τοποθετηθούν σε διάφορες τοποθεσίες, σε εσωτερικούς, ακόμα και εξωτερικούς χώρους και να συμβάλουν στην δημιουργία διαφόρων εφαρμογών.

Επιπρόσθετα, το σύστημα υποστηρίζει την παρακολούθηση αισθητήρων διαφόρων τύπων, οι οποίοι μπορούν να τοποθετηθούν στους προαναφερθέντες μικροελεγκτές και να αποτελέσουν μέρος ενός κυκλώματος αισθητήρων, λαμβάνοντας πληροφορίες από το περιβάλλον τους. Το σύστημα εμφανίζει τις μετρήσεις των αισθητήρων μέσω γραφικών παραστάσεων ώστε ο χρήστης να μπορεί να παρακολουθεί εύκολα την κατάσταση ενός χώρου. Για παράδειγμα, ο χρήστης μπορεί να δει τις γραφικές παραστάσεις που σχηματίζονται από μετρήσεις αισθητήρων θερμοκρασίας μέσα στο σπίτι του και το σύστημα να τον ενημερώνει αν παρουσιαστεί κάποια ανεπιθύμητη κατάσταση.

1.2.2 Οι φάσεις του συστήματος

Η πρώτη φάση του συστήματος δημιουργήθηκε από τη φοιτήτρια Στέλλα Κωνσταντίνου και αφορούσε τη διαχείριση αισθητήρων, οι οποίοι είναι

τοποθετημένοι σε πλατφόρμα μικροελεγκτή Arduino. Έπειτα το σύστημα επεκτάθηκε από το φοιτητή Φίλιππο Αζίζ, ο οποίος δημιούργησε και πρόσθεσε στο σύστημα μια μηχανή κανόνων. Μέσω αυτής της μηχανής, ο χρήστης μπορεί να εισάγει στο σύστημα λογικούς κανόνες τύπου IF – ELSE, οι οποίοι ανάλογα με τις μετρήσεις που λαμβάνουν συγκεκριμένοι αισθητήρες, αυτοματοποιούν κάποιες ενέργειες, όπως η ενεργοποίηση ή απενεργοποίηση συσκευών. Ακολούθως, στα πλαίσια της διπλωματικής εργασίας της φοιτήτριας Αθηνάς Παφίτου, δημιουργήθηκε μια Android εφαρμογή, η οποία αποτελεί ανεξάρτητο κομμάτι του συστήματος SensoMan. Η εφαρμογή αυτή επιτρέπει στους χρήστες του συστήματος να παρακολουθούν την κατάσταση των αισθητήρων τους, οι οποίοι βρίσκονται τοποθετημένοι σε πλατφόρμες μικροελεγκτών Arduino και Raspberry Pi. Τέλος, το σύστημα επεκτάθηκε από το φοιτητή Άγγελο Ευσταθίου, δημιουργώντας ένα κοινωνικό δίκτυο, το οποίο επιτρέπει στους χρήστες του SensoMan να ενωθούν με άλλους χρήστες και να βλέπουν τις μετρήσεις των δικών τους αισθητήρων.

Όπως αναφέρθηκε και προηγουμένως, στα πλαίσια της παρούσας διπλωματικής εργασίας δημιουργήθηκε και προστέθηκε στο σύστημα μια μηχανή παραγωγής κώδικα, η οποία στο παρόν στάδιο υποστηρίζει τέσσερα είδη μικροελεγκτών και έξι διαφορετικούς αισθητήρες.

1.2.3 Τεχνολογίες συστήματος

Για την δημιουργία της διαδικτυακής εφαρμογής SensoMan χρησιμοποιήθηκαν διάφορες τεχνολογίες διαδικτύου, όπου η κάθε μια έχει το δικό της ρόλο στην λειτουργία του συστήματος.

Αρχικά, για την δόμηση και το περιεχόμενο των σελίδων της διαδικτυακής εφαρμογής χρησιμοποιήθηκε η γλώσσα σήμανσης HTML5 (Hypertext Markup Language) [1] η οποία αποτελεί την τελευταία (ως σήμερα) έκδοση της HTML. Η συγκεκριμένη γλώσσα χρησιμοποιεί ετικέτες (tags) της μορφής <όνομα ετικέτας>, οι οποίες

περικλείουν το περιεχόμενο της ιστοσελίδας και με αυτό τον τρόπο σχηματίζεται η δομή της.

Για την δημιουργία και εισαγωγή δυναμικού περιεχομένου χρησιμοποιήθηκε η γλώσσα PHP (Hypertext Preprocessor) [2], η οποία έχει τη δυνατότητα να παράγει περιεχόμενο σε πραγματικό χρόνο και να το εισάγει δυναμικά στη σελίδα που ζητάει ο χρήστης κάθε φορά. Αξίζει να σημειωθεί ότι το δυναμικό περιεχόμενο αυτό μπορεί να διαφέρει από χρήστη σε χρήστη λόγω των επιλογών και αναγκών του καθενός κάθε φορά.

Επιπρόσθετα, για την διαχείριση των ενεργειών του χρήστη χρησιμοποιήθηκε η γλώσσα προγραμματισμού JavaScript [3]. Η γλώσσα αυτή τρέχει στην πλευρά του πελάτη (client-side) και παρακολουθεί αν εγερθεί κάποιο γεγονός (event), είτε από το πάτημα ενός κουμπιού, είτε από την αλλαγή κάποιας κατάστασης ενός στοιχείου από το χρήστη και κάνει τις ανάλογες ενέργειες για κάθε περίπτωση. Ακόμα, η JavaScript χρησιμοποιείται για να καλεστούν οι διάφορες υπηρεσίες διαδικτύου του συστήματος, οι οποίες εξηγούνται σε επόμενη παράγραφο.

Για την αποθήκευση των πληροφοριών και των δεδομένων που χρειάζεται το σύστημα για να λειτουργήσει χρησιμοποιήθηκε η βάση δεδομένων MySQL [4]. Η συγκεκριμένη βάση δεδομένων είναι ανοικτού κώδικα (open source) και είναι γνωστή λόγω της αποδοτικότητας, ασφάλειας και επεκτασιμότητας της. Η βάση δεδομένων του συστήματος SensoMan βρίσκεται σε εσωτερικό εξυπηρετητή του πανεπιστημίου.

Τέλος, το σύστημα SensoMan λειτουργεί μέσω υπηρεσιών διαδικτύου της μορφής REST (RESTful Web Services) [5]. Οι εν λόγω υπηρεσίες χρησιμοποιούνται για την ανάκτηση και αποθήκευση δεδομένων από και προς τη βάση δεδομένων του SensoMan. Οι υπηρεσίες αυτές χρησιμοποιούν URIs (Uniform Resource Identifiers) για την πρόσβαση τους σε πόρους (resources) του συστήματος και μέσω του πρωτοκόλλου επικοινωνίας HTTP (Hypertext Transfer Protocol) στέλνονται διάφορες αιτήσεις (requests) της μορφής GET, POST, PUT και DELETE. Τα περιεχόμενα των

μηνυμάτων που αποστέλλονται μπορεί να έχει διάφορες μορφές (MIME types) όπως για παράδειγμα HTML, XML, απλό κείμενο, PDF, JPEG και JSON. Παρ' όλα αυτά τα μηνύματα του συστήματος SensoMan χρησιμοποιούν μόνο τη μορφή JSON για την αποστολή των δεδομένων τους. Αξίζει να αναφερθεί ότι οποιαδήποτε ενέργεια των υπηρεσιών διαδικτύου είναι stateless, δηλαδή είναι αυτόνομη και εκτελείται διαφορετικά σε κάθε περίπτωση, χωρίς ο εξυπηρετητής να θυμάται ενέργειες που έγιναν στο παρελθόν. Έτσι, αυτό κάνει τις υπηρεσίες διαδικτύου να είναι γρήγορες και να μπορούν να δίνουν διαφορετικές πληροφορίες ανάλογα με κάθε σενάριο.

1.3 Δομή Διπλωματικής Εργασίας

Συνολικά η παρούσα διπλωματική εργασία απαρτίζεται από επτά κεφάλαια. Στο πρώτο κεφάλαιο έγινε μια γενική περιγραφή του συστήματος SensoMan και αναφέρθηκαν οι φάσεις που έχει περάσει το σύστημα μέχρι να φτάσει στην υφιστάμενη κατάσταση του. Παρουσιάστηκε το πρόβλημα το οποίο οδήγησε στην υλοποίηση της παρούσας διπλωματικής εργασίας και αναφέρθηκαν οι διάφορες τεχνολογίες που χρησιμοποιήθηκαν για την δημιουργία της διαδικτυακής εφαρμογής.

Στο δεύτερο κεφάλαιο θα γίνει επεξήγηση διαφόρων ορισμών και εννοιών γύρω από τον κόσμο των αισθητήρων και του Δικτύου των Πραγμάτων (Internet of Things). Συγκεκριμένα θα επεξηγηθούν οι ορισμοί: Δίκτυο των Πραγμάτων (Internet of Things), μικροελεγκτής, αισθητήρας και RESTful υπηρεσίες διαδικτύου. Επίσης θα γίνει σύντομη περιγραφή του συστήματος openHAB, το οποίο αποτελεί παρόμοιο σύστημα με το SensoMan.

Στο τρίτο κεφάλαιο θα γίνει περιγραφή του υλικού (hardware) που χρησιμοποιείται στο σύστημα SensoMan όπως οι διάφορες πλατφόρμες μικροελεγκτών και αισθητήρες που υποστηρίζονται από τη μηχανή παραγωγής κώδικα.

Στο τέταρτο κεφάλαιο θα γίνει εκτενής περιγραφή του συστήματος SensoMan, όπως η αρχιτεκτονική του και τα διάφορα κομμάτια που το απαρτίζουν.

Στο πέμπτο κεφάλαιο θα περιγραφεί η μηχανή παραγωγής κώδικα, η οποία δημιουργήθηκε στα πλαίσια της παρούσας διπλωματικής εργασίας. Συγκεκριμένα, θα γίνει μια γενική εισαγωγή και επεξήγηση του προβλήματος το οποίο επιλύει, θα αναλυθεί η δομή της και θα επεξηγηθούν οι διάφορες λειτουργίες τις οποίες προσφέρει.

Στο έκτο κεφάλαιο, θα παρουσιαστούν δύο σενάρια χρήσης της μηχανής παραγωγής κώδικα και έπειτα θα γίνει αξιολόγηση τους.

Στο έβδομο και τελευταίο κεφάλαιο, θα διεξαχθούν κάποια συμπεράσματα και θα παρουσιαστούν μελλοντικές επεκτάσεις που μπορούν να αναβαθμίσουν το σύστημα ώστε να προσφέρει περισσότερες λειτουργίες.

Κεφάλαιο 2

Υπόβαθρο

Υπόβαθρο	9
2.1 Διαδίκτυο των Πραγμάτων	10
2.2 Πλατφόρμα μικροελεγκτή	11
2.3 Αισθητήρας	14
2.4 RESTful υπηρεσίες διαδικτύου	15
2.5 Σύστημα OpenHAB.....	17

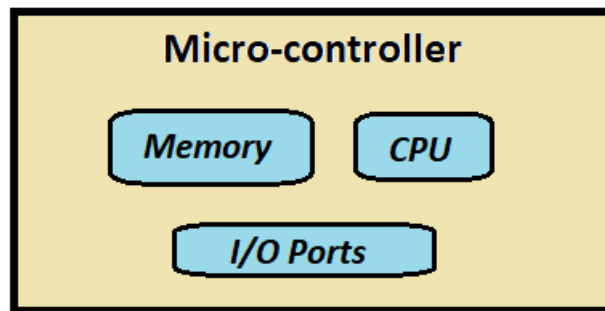
Βασική ιδέα πίσω από το Διαδίκτυο των Πραγμάτων (Internet of Things) είναι οτιδήποτε υπάρχει, να συνδεθεί στο διαδίκτυο. Αυτή η άποψη αποσκοπεί στο να αλλάξει την ζωή του ανθρώπου, κάνοντας την καλύτερη, πιο ευχάριστη και πιο δημιουργική. Πολλοί υποστηρίζουν ότι αν αυτή η ιδέα γίνει πραγματικότητα θα μπορεί ο άνθρωπος να αυτοματοποιήσει διάφορες διαδικασίες οι οποίες δεν θα χρειάζονται την δική του παρέμβαση, έτσι θα μπορεί να επικεντρώνεται στα δικά του ενδιαφέροντα και εργασίες χωρίς να σπαταλάει το χρόνο του σε αχρείαστες δραστηριότητες.

Ένα απλό παράδειγμα είναι το εξής: το ξυπνητήρι είναι ενωμένο στο διαδίκτυο και επικοινωνεί με την καφετιέρα και το θερμοστάτη για το νερό του μπάνιου. Πριν κτυπήσει το ξυπνητήρι το πρωί, ενεργοποιεί την καφετιέρα για να φτιάξει καφέ και δίνει εντολή στο θερμοστάτη να ζεστάνει το νερό. Έτσι, όταν ο άνθρωπος ξυπνήσει το πρωί, θα έχει έτοιμο καφέ και ζεστό νερό χωρίς να κάνει οποιαδήποτε ενέργεια. Αυτό το απλό παράδειγμα είναι ένα από τα πολλά που μπορούν να υπάρξουν όσο αυξάνεται η διάδοση του διαδικτύου των πραγμάτων.

Σήμερα, συνηθίζεται να λέγεται ότι οι συσκευές οι οποίες μπορούν να συνδεθούν στο διαδίκτυο ονομάζονται «έξυπνες συσκευές» (smart devices) και το σπίτι το οποίο εμπεριέχει αυτές τις συσκευές ονομάζεται «έξυπνο σπίτι» (smart home). Παρομοίως, αυτή η ιδέα μπορεί να εφαρμοστεί και σε άλλους χώρους, δημιουργώντας «έξυπνα νοσοκομεία», «έξυπνα αυτοκίνητα», «έξυπνες πόλεις» και «έξυπνη γεωργία».

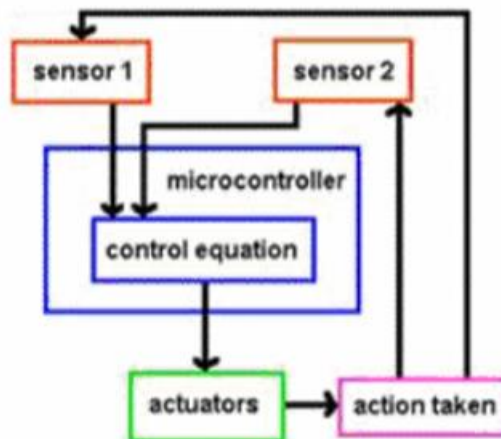
2.2 Πλατφόρμα μικροελεγκτή

Ένα βασικό κομμάτι του Διαδικτύου των Πραγμάτων είναι η πλατφόρμα μικροελεγκτή. Μια πλατφόρμα μικροελεγκτή είναι ένα προγραμματιζόμενο ολοκληρωμένο σύστημα το οποίο διαθέτει επεξεργαστή, μνήμη, διάφορα περιφερειακά κυκλώματα καθώς επίσης και εισδοχές εισόδου/εξόδου για επικοινωνία με εξωτερικές συσκευές.



Εικόνα 2.2: Πλατφόρμα Μικροελεγκτή

Οι εξωτερικές συσκευές μπορεί να είναι οποιαδήποτε συσκευή ή αισθητήρας, ο οποίος υποστηρίζεται από το μικροελεγκτή. Λόγω της ευελιξίας και της φορητότητας της πλατφόρμας μικροελεγκτή, μπορεί να χρησιμοποιηθεί ως μέρος συστήματος μικρής ή μεγάλης κλίμακας. Μπορεί να προγραμματιστεί με πολλούς διαφορετικούς τρόπους προσφέροντας κάθε φορά διαφορετικές δυνατότητες. Το πρόγραμμα αποθηκεύεται μόνιμα στη μνήμη προγράμματος και εκτελείται συνεχώς, όσο η πλατφόρμα μικροελεγκτή είναι συνδεδεμένη σε κάποια πηγή ενέργειας.



Εικόνα 2.3: Βρόχος λειτουργίας ενός μικροελεγκτή [8]

Στην εικόνα 2.3 φαίνεται διαγραμματικά ο βρόχος ελέγχου που εκτελεί μια πλατφόρμα μικροελεγκτή. Αρχικά οι αισθητήρες οι οποίοι είναι ενσωματωμένοι στην πλατφόρμα μικροελεγκτή λαμβάνουν πληροφορία από το εξωτερικό περιβάλλον και

στέλνουν σήματα στις θύρες εισόδου του μικροελεγκτή. Τα σήματα αυτά μπορεί να έχουν τη μορφή λογικών ψηφιακών καταστάσεων (0 ή 1) ή αναλογικά σήματα τα οποία στη συνέχεια μετατρέπονται σε ψηφιακά. Ακολουθώντας, ο μικροελεγκτής συλλέγει και αναλύει τα δεδομένα μέσω αλγορίθμων (προγράμματος). Τέλος, ανάλογα με την περίπτωση αποστέλλει σήματα σε εξωτερικές συσκευές ή ενεργοποιητές (actuators) μέσω θυρών εξόδου, οι οποίοι με τη σειρά τους κάνουν τις δικές τους ενέργειες.

Μερικά παραδείγματα εφαρμογών στα οποία μπορεί να λάβει μέρος μια πλατφόρμα μικροελεγκτή είναι τα εξής:

- **Συστήματα αυτοματισμών:** Συστήματα τα οποία συνδυάζουν αισθητήρες μαζί με ενεργοποιητές (actuators) με στόχο να αυτοματοποιήσουν διαδικασίες χωρίς την συγκατάθεση του ανθρώπου.
- **Ηλεκτρονικές και ηλεκτρικές συσκευές:** Συσκευές οι οποίες βρίσκονται σε εσωτερικούς και εξωτερικούς χώρους και δίνουν δυνατότητα απομακρυσμένης πρόσβασης.
- **Συστήματα συλλογής δεδομένων:** Συστήματα τα οποία χρησιμοποιούν αισθητήρες με στόχο τη συλλογή δεδομένων και μετρήσεων από το περιβάλλον στο οποίο βρίσκονται και καταγραφή τους σε αρχεία (log files) ή βάση δεδομένων (database).

Υπάρχουν διάφορες πλατφόρμες μικροελεγκτών οι οποίες διαφέρουν στα χαρακτηριστικά και στις δυνατότητες τις οποίες προσφέρουν. Για παράδειγμα, μπορούμε να έχουμε μικροελεγκτές με διαφορετικές ταχύτητες επεξεργαστή, διαφορετική χωρητικότητα μνήμης και διαφορετικές θύρες εισόδου/εξόδου. Οι θύρες εισόδου/εξόδου μπορεί να διαφέρουν στο αριθμό τους και στο είδος τους, δηλαδή αν είναι αναλογικής ή ψηφιακής φύσεως.

Λόγω της μεγάλης δημοσιότητας τους τα τελευταία χρόνια, έχουν κατασκευαστεί πλατφόρμες μικροελεγκτών από διαφορετικές εταιρίες. Επικρατέστερες στην αγορά

αποτελούν οι Arduino, Raspberry Pi, BeagleBone Black και Intel Galileo, οι οποίες θα αναλυθούν στην συνέχεια σε επόμενο κεφάλαιο.

2.3 Αισθητήρας

Αισθητήρας ονομάζεται η συσκευή που λαμβάνει πληροφορία από το περιβάλλον στο οποίο βρίσκεται, ανιχνεύοντας ένα φυσικό μέγεθος και παράγει μια μετρήσιμη έξοδο. Μερικά παραδείγματα αισθητήρων αποτελούν οι αισθητήρες κίνησης, φωτεινότητας, πίεσης, θερμοκρασίας, ήχου και υγρασίας.

Στις μέρες μας, οι αισθητήρες είναι μέρος της ζωής του ανθρώπου και χρησιμοποιούνται παντού, σε διάφορους χώρους και αντικείμενα. Για παράδειγμα, μπορεί να βρίσκονται σε αυτοκίνητα, σε ανελκυστήρες, σε μηχανές, στην ιατρική, στην ρομποτική, στην γεωργία, σε κινητά τηλέφωνα, σε φορητούς υπολογιστές και σε ότι μπορεί κανείς να φανταστεί. Ο συνδυασμός πολλών αισθητήρων δημιουργεί κυκλώματα αισθητήρων, τα οποία βοηθούν στα να αυτοματοποιούνται διαδικασίες, καθώς και στη συλλογή δεδομένων για ανάλυση και εξαγωγή συμπερασμάτων.

Μερικά από τα χαρακτηριστικά των αισθητήρων είναι τα εξής:

- **Εύρος:** Τα όρια στα οποία ο αισθητήρας λειτουργεί αξιόπιστα.
- **Ακρίβεια:** Πόσο ακριβής είναι η μέτρηση του αισθητήρα έναντι της πραγματικής.
- **Σφάλμα:** Η διαφορά ανάμεσα στην μετρούμενη τιμή και την πραγματική.
- **Απόκριση:** Ο χρόνος που απαιτείται για να λάβει την τελική τιμή η έξοδος.
- **Καθυστερήση:** Ο χρόνος αλλαγής της εξόδου ως προς την είσοδο.
- **Χρόνος ζωής:** Ο εκτιμώμενος χρόνος λειτουργίας του αισθητήρα στα πλαίσια των προδιαγραφών του.

2.4 RESTful υπηρεσίες διαδικτύου

Υπηρεσίες διαδικτύου ονομάζονται οποιεσδήποτε υπηρεσίες – λειτουργικότητες παρέχονται στο χρήστη μέσω του διαδικτύου. Στην περίπτωση των RESTful (Representational State Transfer) υπηρεσιών διαδικτύου, βασική αρχιτεκτονική αρχή είναι ο χαρακτηρισμός όλων των υπηρεσιών και στοιχείων που χρησιμοποιούνται ως πόροι του συστήματος (resources) και η πρόσβαση τους γίνεται μέσω URIs (Uniform Resource Identifiers).

Ένα URI αποτελείται από το πρωτόκολλο επικοινωνίας (scheme) που χρησιμοποιείται, τη διεύθυνση του εξυπηρετητή (host) που προσφέρει το πόρο που ζητείται, το μονοπάτι (path) που βρίσκεται ο πόρος εσωτερικά του εξυπηρετητή, ο πόρος (resource) και κάποιες φορές το ερώτημα (query), το οποίο μπορεί να περιλαμβάνει κάποιες παραμέτρους για προσαρμογή του τελικού αποτελέσματος προς το χρήστη.



Εικόνα 2.4: Uniform Resource Identifier

Στην εικόνα 2.4 πιο πάνω φαίνεται ένα παράδειγμα URI το οποίο αντιπροσωπεύει την ανάκτηση του αρχείου (πόρου) `rfc2396.txt`, μέσω του πρωτοκόλλου επικοινωνίας `http` από ένα εξυπηρετητή.

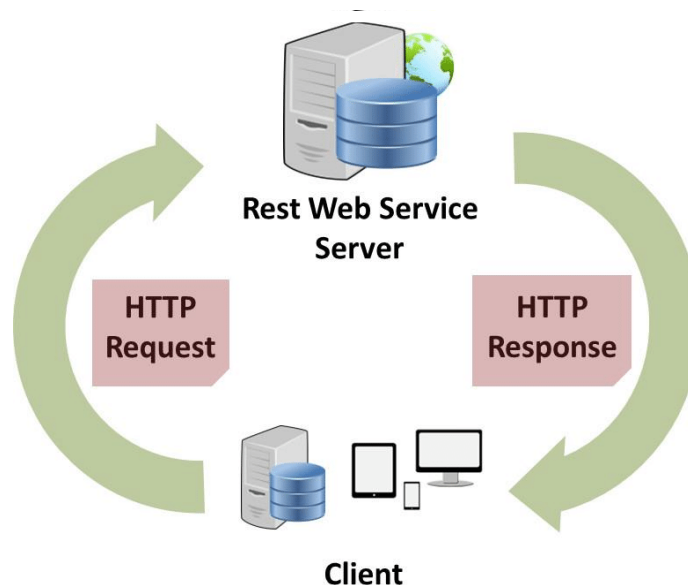
Άλλο ένα χαρακτηριστικό των RESTful υπηρεσιών διαδικτύου, σε αντίθεση με άλλα είδη, είναι η έννοια της stateless πρόσβασης. Δηλαδή, ο εξυπηρετητής, ο οποίος αναλαμβάνει να εκτελέσει μια υπηρεσία διαδικτύου, δεν χρειάζεται κάποιο ιστορικό του χρήστη ή προηγούμενων ενεργειών που έγιναν στο σύστημα για να φέρει εις πέρας την λειτουργία αυτή, ούτε αποθηκεύει την νέα κατάσταση του συστήματος. Αυτό το γεγονός, κάνει τις REST υπηρεσίες διαδικτύου να είναι πολύ γρήγορες και

αξιόπιστες, αφού μπορούν, χωρίς επιπλέον πληροφορίες, να προσαρμοστούν στις ανάγκες της κάθε περίπτωσης ξεχωριστά.

Για την μεταφορά δεδομένων από και προς τον εξυπηρετητή χρησιμοποιείται το πρωτόκολλο μεταφοράς δεδομένων στον παγκόσμιο ιστό HTTP (Hypertext Transfer Protocol). Μέσω του συγκεκριμένου πρωτοκόλλου δίνεται η επιλογή να χρησιμοποιηθούν όλες οι μέθοδοι που προσφέρει όπως GET για ανάκτηση κάποιου πόρου, PUT για δημιουργία κάποιου νέου πόρου, POST για ενημέρωση κάποιου υφιστάμενου και DELETE για διαγραφή του.

Οι RESTful υπηρεσίες διαδικτύου προσφέρουν την δυνατότητα να μεταφερθούν διάφοροι τύποι δεδομένων (MIME types), όπως HTML, JSON, XML, PDF, απλό κείμενο και JPEG. Τα μηνύματα που ανταλλάσσονται στο σύστημα SensoMan περιλαμβάνουν δεδομένα της μορφής JSON (JavaScript Object Notation). Ο τύπος αυτός χαρακτηρίζεται από δεδομένα της μορφής ζευγών κλειδιού και τιμής (key-value pairs) και χρησιμοποιείται επανειλημμένα σε πολλές εφαρμογές διαδικτύου αφού είναι πολύ εύκολη η ανάγνωση και επεξεργασία του.

Στην εικόνα 2.5 φαίνεται ο κύκλος εκτέλεσης μιας υπηρεσίας διαδικτύου.



Εικόνα 2.5: Κύκλος εκτέλεσης RESTful υπηρεσίας διαδικτύου [28]

Αρχικά ο χρήστης (Client) στέλνει ένα αίτημα HTTP (HTTP request) για πρόσβαση σε κάποιο πόρο του συστήματος στο εξυπηρετητή ο οποίος προσφέρει τις RESTful υπηρεσίες διαδικτύου. Έπειτα, αφού ο εξυπηρετητής εκτελέσει την ανάλογη υπηρεσία επιστρέφει πίσω στο χρήστη του συστήματος την απάντηση (HTTP response) στο αίτημα του. Σημειώνεται ότι τα μηνύματα που μεταφέρονται εμπεριέχουν δεδομένα τα οποία έχουν μια μορφή από αυτές που αναφέρθηκαν σε προηγούμενη παράγραφο. Όταν τελειώσει αυτός ο κύκλος εκτέλεσης της υπηρεσίας, όπως αναφέρθηκε και προηγουμένως, δεν αποθηκεύεται κάποιο ιστορικό για την ενέργεια που έγινε.

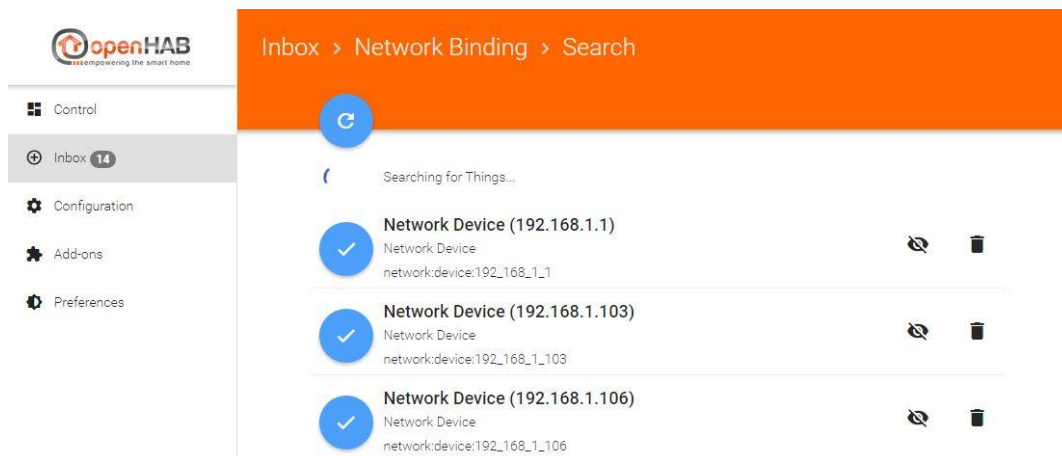
Σε αυτό το σημείο αξίζει να σημειωθεί ότι οι RESTful υπηρεσίες του συστήματος SensoMan χρησιμοποιούνται για την επικοινωνία του συστήματος, καθώς και εξωτερικών συσκευών (όπως πλατφόρμες μικροελεγκτών), με τη βάση δεδομένων. Προσφέρουν τη δυνατότητα να αποθηκευτούν νέα δεδομένα στη βάση δεδομένων, να γίνει επεξεργασία υφιστάμενων καθώς επίσης και να διαγραφεί μόνιμα κάποιο από αυτά. Παραδείγματα χρήσης των RESTful υπηρεσιών του συστήματος αποτελούν η προσθήκη νέας πλατφόρμας μικροελεγκτή σε κάποιο χρήστη, η προσθήκη νέου αισθητήρα σε κάποιο μικροελεγκτή, η αλλαγή του κωδικού πρόσβασης κάποιου χρήστη, η διαγραφή κάποιου αρχείου του χρήστη κ.α.

2.5 Σύστημα OpenHAB

Σε αυτό το υποκεφάλαιο θα παρουσιαστεί εν συντομία ο τρόπος λειτουργίας του συστήματος openHAB [9], το οποίο είναι ένα παρόμοιο σύστημα με το SensoMan. Το openHAB (open Home Automation Bus) είναι μια πλατφόρμα ανοικτού κώδικα, τεχνολογικού περιεχομένου για αυτοματοποίηση διαδικασιών σε ένα έξυπνο σπίτι. Παρέχει ένα ενιαίο τρόπο διαχείρισης και ελέγχου έξυπνων συσκευών, οι οποίες ονομάζονται “items” από το σύστημα και τοποθετούνται σε διαφορετικές τοποθεσίες μέσα σε ένα σπίτι. Αυτή τη στιγμή η πλατφόρμα openHAB υποστηρίζει ένα μεγάλο εύρος συσκευών και λόγω του ότι είναι ανοικτού κώδικα, συνεχίζει να αναπτύσσεται από μεγάλη κοινότητα ανθρώπων.

Ο τρόπος λειτουργίας και χειρισμού ενός item στο σύστημα openHAB, απαιτεί μια βασική διαδικασία την οποία ο χρήστης πρέπει να ακολουθήσει. Αρχικά, για την εισαγωγή ενός νέου item, χρειάζεται να εγκατασταθεί το κατάλληλο “binding” για αυτό. Τα bindings είναι στην ουσία οι διάφορες κατηγορίες έξυπνων συσκευών που υποστηρίζονται από το σύστημα. Μέσω της καρτέλας περιήγησης “Add-ons”, ο χρήστης κατευθύνεται σε μια νέα σελίδα όπου παρουσιάζονται όλα τα bindings, σε μορφή λίστας, και από εκεί μπορεί να εγκαταστήσει όποιο επιθυμεί.

Όλα τα εγκατεστημένα bindings εμφανίζονται στο “Inbox” του χρήστη. Πατώντας σε κάποιο, το σύστημα κάνει αναζήτηση και εμφανίζει όλες τις διαθέσιμες συσκευές οι οποίες είναι συνδεδεμένες στο διαδίκτυο (κάτω από το συγκεκριμένο binding) και βρίσκονται εντός του δικτύου (subnet) που είναι εγκατεστημένο το openHAB. Οι συνδεδεμένες συσκευές αποκτούν ως αρχικό όνομα την IP διεύθυνση την οποία έχουν στο δίκτυο. Ο χρήστης μπορεί να μετονομάσει την κάθε συσκευή, δίνοντας της ένα όνομα για να την ξεχωρίζει και να την προσθέσει στη λίστα με τα αντικείμενα του. Στην εικόνα 2.6 φαίνεται ένα στιγμιότυπο οθόνης με όλες τις διαθέσιμες συσκευές που βρήκε το σύστημα openHAB για ένα συγκεκριμένο παράδειγμα.



Εικόνα 2.6: Διαθέσιμες συσκευές

Κάθε συσκευή του χρήστη μπορεί να ορίσει κάποια κανάλια (channels) τα οποία βοηθούν στην επικοινωνία του συστήματος με τη συσκευή. Το κάθε ένα από αυτά

παρέχει μια διαφορετική λειτουργικότητα και μπορεί να χρησιμοποιηθεί ως μέσο αποστολής μηνυμάτων από και προς τη συσκευή. Για παράδειγμα, το κανάλι “Online” δείχνει την κατάσταση της συσκευής και μέσω αυτού ο χρήστης μπορεί να την ενεργοποιήσει και απενεργοποιήσει.

Ακόμα μια δυνατότητα την οποία παρέχει το σύστημα openHAB είναι η προσθήκη κανόνων (rules). Οι κανόνες συνδυάζουν τις λειτουργίες δύο ή περισσότερων συσκευών και βοηθούν στην αυτοματοποίηση διαδικασιών σε ένα έξυπνο σπίτι. Είναι της μορφής IF-ELSE, όπου το IF ορίζει μια συνθήκη και το ELSE μια ενέργεια, έτσι αν ικανοποιείται η συνθήκη τότε εκτελείται η ανάλογη ενέργεια.

Συνοπτικά, το σύστημα openHAB βοηθάει στο χειρισμό και απομακρυσμένο έλεγχο έξυπνων συσκευών, οι οποίες τοποθετούνται σε ένα έξυπνο σπίτι. Ο τρόπος με τον οποίο δουλεύει και οι δυνατότητες τις οποίες προσφέρει μοιάζουν με αυτές του συστήματος SensoMan. Εντούτοις το σύστημα SensoMan επικεντρώνεται στην υποστήριξη πλατφόρμων μικροελεγκτών και αισθητήρων με σκοπό την συγκέντρωση και ανάλυση μετρήσεων, οι οποίες λαμβάνονται από το περιβάλλον.

Κεφάλαιο 3

Υλικό

Υλικό.....	20
3.1 Πλατφόρμες μικροελεγκτών.....	21
3.1.1 Πλατφόρμα μικροελεγκτή Arduino	21
3.1.2 Πλατφόρμα μικροελεγκτή Raspberry Pi.....	24
3.1.3 Πλατφόρμα μικροελεγκτή Beaglebone Black.....	27
3.1.4 Πλατφόρμα μικροελεγκτή Intel Galileo.....	29
3.2 Αισθητήρες.....	30
3.2.1 Αισθητήρας κίνησης PIR	30
3.2.2 Αισθητήρας φωτεινότητας LDR.....	31
3.2.3 Αισθητήρας θερμοκρασίας DS18B20	32
3.2.4 Αισθητήρας θερμοκρασίας TMP36	33
3.2.5 Αισθητήρας ήχου	34
3.2.6 Αισθητήρας υγρασίας FC-28.....	34
3.3 Επιπλέον εξαρτήματα	35
3.3.1 Wi-Fi Shield CC3000	36
3.3.2 Ethernet Shield	36
3.3.3 Wireless card και αντένες Wi-Fi	37

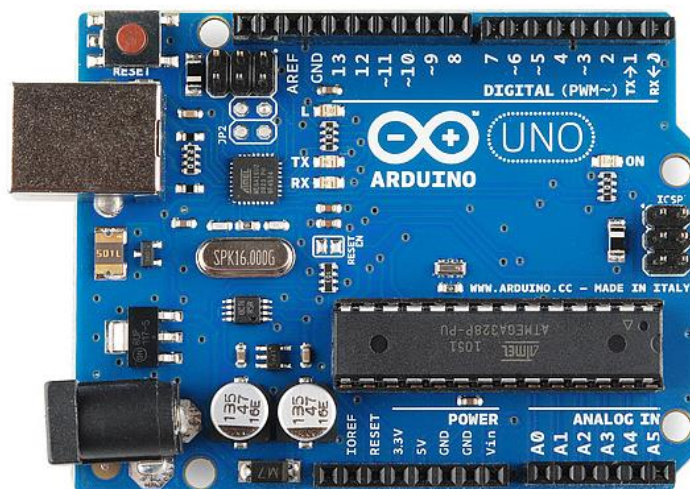
Σε αυτό το κεφάλαιο θα περιγραφεί το υλικό (hardware) που χρησιμοποιήθηκε στα πλαίσια της παρούσας διπλωματικής εργασίας.

3.1 Πλατφόρμες μικροελεγκτών

Για δημιουργία του μηχανισμού παραγωγής κώδικα χρησιμοποιήθηκαν τέσσερις βασικές πλατφόρμες μικροελεγκτών: Arduino, Raspberry Pi, Beaglebone Black και Intel Galileo. Οι συγκεκριμένες πλατφόρμες παρέχουν την δυνατότητα προσθήκης αισθητήρων καθώς και προγραμματισμού τους από το χρήστη όμως διαφέρουν στα χαρακτηριστικά τους, τα οποία θα αναλυθούν στη συνέχεια.

3.1.1 Πλατφόρμα μικροελεγκτή Arduino

Το Arduino [11] είναι μια από τις δημοφιλέστερες πλατφόρμες μικροελεγκτών σήμερα. Είναι πλατφόρμα ανοικτού κώδικα, χαμηλού κόστους που χαρακτηρίζεται από την ευκολία χρήσης και επεκτασιμότητας της. Μπορεί να χρησιμοποιηθεί για να φτιαχτούν μικρά αλλά και πολύπλοκα συστήματα που στόχο έχουν να αυτοματοποιήσουν διαδικασίες ή να λάβουν μετρήσεις από το περιβάλλον μέσω αισθητήρων.

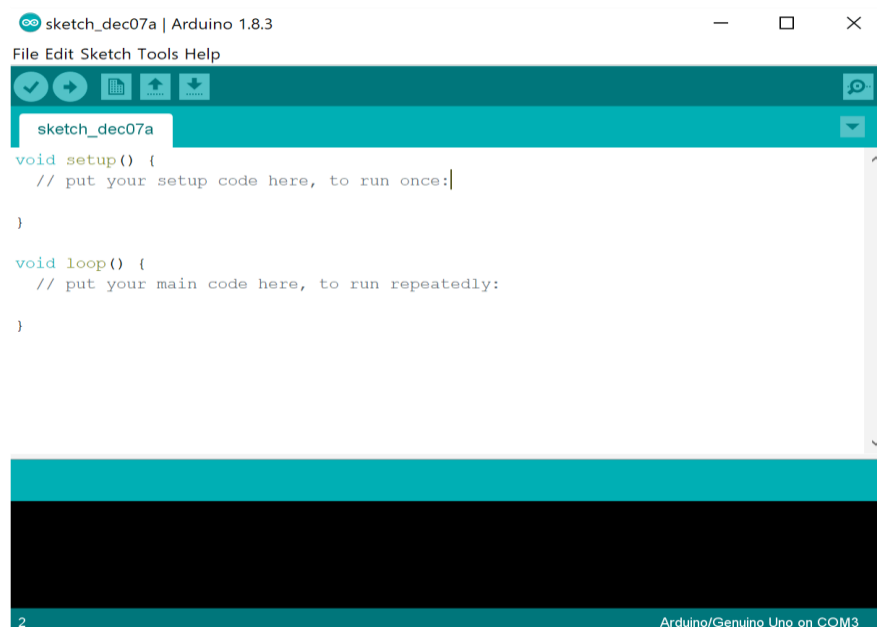


Εικόνα 3.1: Πλατφόρμα μικροελεγκτή Arduino Uno [11]

Λόγω του ότι η πλατφόρμα αυτή είναι ανοικτού κώδικα υποστηρίζεται από μια μεγάλη κοινότητα ατόμων παγκοσμίως και χρησιμοποιείται για εκπαιδευτικούς ή και ψυχαγωγικούς σκοπούς από μαθητές, καθηγητές, προγραμματιστές, επαγγελματίες καθώς και απλούς χρήστες.

Για τον προγραμματισμό του Arduino χρησιμοποιείται η γλώσσα προγραμματισμού Wiring, που στην ουσία είναι μια απλοποιημένη μορφή της C++ καθώς και ένα σύνολο βιβλιοθηκών υλοποιημένες επίσης στην ίδια γλώσσα. Ο προγραμματιστής του Arduino μπορεί να γράψει προγράμματα μέσω του λογισμικού Arduino IDE [12] το οποίο επιτρέπει την μεταγλώττιση και εγκατάσταση του κώδικα απευθείας στην πλατφόρμα. Το λογισμικό αυτό υποστηρίζεται από όλα τα λειτουργικά συστήματα (cross-platform), όπως Windows, Macintosh και Linux.

Στην πιο κάτω εικόνα (εικόνα 3.2) φαίνεται ένα στιγμιότυπο οθόνης από το Arduino IDE.



Εικόνα 3.2: Arduino IDE [12]

Υπάρχουν διάφορες πλατφόρμες τύπου Arduino οι οποίες διαφέρουν στα χαρακτηριστικά, στο μέγεθος και στην τιμή τους. Μερικά από τα χαρακτηριστικά μιας πλατφόρμας μικροελεγκτή Arduino είναι η ταχύτητα του επεξεργαστή, ο αριθμός των εισδοχών (pins), η χωρητικότητα της μνήμης καθώς και τα επιπλέον στοιχεία που μπορεί να υπάρχουν ενσωματωμένα, όπως θύρα mini USB και θύρα για SD κάρτα μνήμης.

Οι εισδοχές προσφέρουν τη δυνατότητα να συνδεθούν εξωτερικές συσκευές, όπως και αισθητήρες και διαφέρουν ανάλογα με τη λειτουργία τους. Οι εισδοχές της πλατφόρμας μικροελεγκτή Arduino είναι οι εξής:

- **GND:** Εισδοχές για γείωση του κυκλώματος.
- **5V και 3.3V:** Εισδοχές για ηλεκτροδότηση των εξωτερικών συσκευών και αισθητήρων.
- **Analog in:** Εισδοχές για είσοδο αναλογικού σήματος
- **Digital I/O:** Εισδοχές για είσοδο και έξοδο ψηφιακού σήματος.
- **PWM:** Εισδοχές οι οποίες χρησιμοποιούνται σαν τις κανονικές εισδοχές εισόδου και εξόδου ψηφιακού σήματος, όπως επίσης και για προσομοίωση εξόδου αναλογικού σήματος (Pulse-Width Modulation).

Τα ονόματα των πιο γνωστών τύπων Arduino είναι Arduino Uno, Arduino Mega, Arduino Nano και Arduino Yun. Η δημοφιλέστερη πλατφόρμα είναι η Arduino Uno η οποία περιλαμβάνει 14 εισδοχές για είσοδο και έξοδο ψηφιακού σήματος, 5 εισδοχές για είσοδο αναλογικού σήματος, 3 εισδοχές για γείωση (Ground - GND), 2 εισδοχές για ηλεκτροδότηση εξωτερικών συσκευών και αισθητήρων και 6 εισδοχές PWM για προσομοίωση εξόδου αναλογικού σήματος.

Στο πίνακα 3.1 φαίνονται τα χαρακτηριστικά (ταχύτητα επεξεργαστή, αριθμός εισδοχών, χωρητικότητα μνήμης, επιπλέον στοιχεία και τιμή) των προαναφερθέντων τύπων Arduino.

Name	CPU speed	Analog pins	Digital pins	Memory	Extras	Price
Uno	16MHz	6 input	14 I/O 6 PWM	32KB flash memory	USB connection DC Power jack	\$22
Mega 2560	16MHz	16 input	54 I/O 15 PWM	256KB flash memory	USB connection DC Power jack	\$38.50
Nano	16MHz	8 input	16 I/O 6 PWM	32KB flash memory	Mini USB connection	\$22
Yun	16MHz 400MHz	12 input	20 I/O 7 PWM	32KB flash memory 64MB RAM DDR2	Micro SD card reader USB 2.0 Ethernet port Wi-Fi support	\$68.20

Πίνακας 3.1: Χαρακτηριστικά πλατφόρμων Arduino

3.1.2 Πλατφόρμα μικροελεγκτή Raspberry Pi

Η πλατφόρμα μικροελεγκτή Raspberry Pi [14] κατασκευάστηκε στο Ηνωμένο Βασίλειο από την εταιρία Raspberry Pi Foundation, με σκοπό να αναβαθμίσει την εκπαίδευση στην επιστήμη των υπολογιστών. Μπορεί να χρησιμοποιηθεί και στη δημιουργία κυκλωμάτων αισθητήρων, αυτοματοποίησης διαδικασιών και ρομποτικών συστημάτων. Λόγω του μικρού κόστους, φορητότητας και μικρού μεγέθους της έγινε ευρέως γνωστή και χρησιμοποιείται παγκόσμια σε πολλές εφαρμογές.



Εικόνα 3.3: Πλατφόρμα μικροελεγκτή Raspberry Pi 2, Model B

Η πλατφόρμα μικροελεγκτή Raspberry Pi, σε αντίθεση με το Arduino, μπορεί να χρησιμοποιηθεί σαν ένας μικρός ηλεκτρονικός υπολογιστής. Αυτό οφείλεται στο γεγονός ότι έχει γρηγορότερο επεξεργαστή της τάξης των GHz, μεγαλύτερη χωρητικότητα μνήμης, δυνατότητα σύνδεσης με εξωτερική οθόνη μέσω θύρας HDMI, δυνατότητα σύνδεσης πληκτρολογίου και ποντικιού και δυνατότητα εφαρμογής λειτουργικού συστήματος Linux.

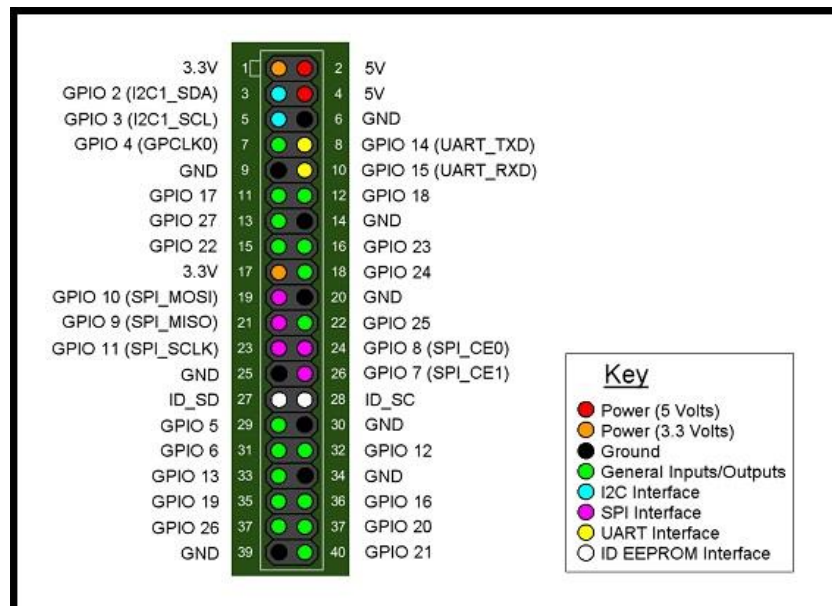
Οι πιο γνωστές εκδόσεις του λειτουργικού συστήματος Linux που μπορούν να εγκατασταθούν στην πλατφόρμα είναι το Raspbian, το οποίο είναι βασισμένο στο λειτουργικό σύστημα Debian και το Pidora, το οποίο είναι βασισμένο στο Fedora. Μια πιο απλοποιημένη έκδοση λειτουργικού συστήματος αποτελεί το NOOBS (New Out Of Box Software) το οποίο προορίζεται για αρχάριους χρήστες που θέλουν να πειραματιστούν. Η εγκατάσταση λειτουργικού συστήματος στην πλατφόρμα γίνεται μέσω κάρτας SD.

Φυσικά, η συγκεκριμένη πλατφόρμα είναι πιο αργή από ένα μοντέρνο ηλεκτρονικό υπολογιστή αλλά αποτελεί ολοκληρωμένο ηλεκτρονικό σύστημα το οποίο προσφέρει αξιόλογες δυνατότητες. Αξίζει να αναφερθεί ότι λόγω του μεγέθους της (μέγεθος μιας πιστωτικής κάρτας) δεν καταναλώνει αρκετή ενέργεια.

Όπως και το Arduino, έτσι και το Raspberry Pi χωρίζεται σε διάφορα μοντέλα, τα οποία αντιπροσωπεύουν την εξέλιξη της πλατφόρμας κατά την πάροδο του χρόνου. Αρχικά, η πρώτη γενιά περιλαμβάνει τρεις πλατφόρμες, την πλατφόρμα Raspberry Pi 1 Model B η οποία χρονολογείται το 2012, την πλατφόρμα Raspberry Pi 1 Model A, η οποία αποτελεί μια πιο απλοποιημένη μορφή της προηγούμενης και την πλατφόρμα Raspberry Pi Model B+, η οποία είχε βελτιωμένο σχεδιασμό καθώς και καλύτερα χαρακτηριστικά. Έπειτα, ακολουθεί η δεύτερη γενιά η οποία περιλαμβάνει την πλατφόρμα Raspberry Pi 2, η οποία περιλάμβανε βελτιωμένα χαρακτηριστικά όπως επεξεργαστή και περισσότερη μνήμη RAM. Τέλος, η τρίτη γενιά περιλαμβάνει δύο πλατφόρμες, την Raspberry Pi 3 Model B (2016) και την Raspberry Pi Model B+ (2018).

Στο σύστημα SensoMan χρησιμοποιήθηκε η πλατφόρμα Raspberry Pi 2, Model B [15] η οποία φαίνεται στην εικόνα 3.3. Τα χαρακτηριστικά της συγκεκριμένης πλατφόρμας είναι τα εξής: 900MHz επεξεργαστής (ARM Cortex-A7), 1GB RAM, 4 θύρες USB, 40 GPIO (General Purpose Input Output) εισδοχές, θύρα HDMI, θύρα ethernet για σύνδεση στο διαδίκτυο, 3.5mm audio jack και θύρα για micro SD κάρτα.

Στην εικόνα 3.4 φαίνονται διαγραμματικά οι εισδοχές GPIO της συγκεκριμένης πλατφόρμας, οι οποίες χρησιμοποιούνται για είσοδο και έξοδο ψηφιακού σήματος, οι εισδοχές ηλεκτροδότησης 3.3V και 5V και οι εισδοχές γείωσης του κυκλώματος (GND).



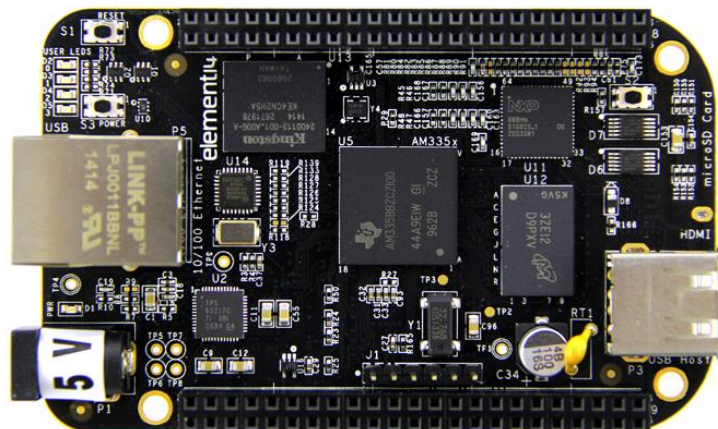
Εικόνα 3.4: Διάγραμμα εισδοχών Raspberry Pi

Σημειώνεται ότι ένα από τα μειονεκτήματα του Raspberry Pi σε σύγκριση με το Arduino είναι ότι οι εισδοχές (pins) του χρησιμοποιούνται μόνο για είσοδο και έξοδο ψηφιακού σήματος καθώς δεν υπάρχουν εισδοχές αναλογικού σήματος. Για το λόγο αυτό, αν ο χρήστης επιθυμεί να συνδέσει εξωτερική συσκευή ή αισθητήρα αναλογικού σήματος, θα χρειαστεί ένα μετατροπέα ψηφιακού-αναλογικού σήματος (digital to analog converter).

3.1.3 Πλατφόρμα μικροελεγκτή Beaglebone Black

Η πλατφόρμα μικροελεγκτή Beaglebone Black [16], όπως και το Raspberry Pi προσομοιώνει ένα μικρό ηλεκτρονικό υπολογιστή και κατασκευάστηκε για εκπαιδευτικούς σκοπούς. Είναι πλατφόρμα ανοικτού κώδικα που στόχο έχει να διαδοθεί στα σχολεία και πανεπιστήμια για να διδάξει τις δυνατότητες του υλικού και λογισμικού ανοικτού κώδικα.

Κατασκευάστηκε από την εταιρία Texas Instruments σε συνεργασία με την Digi-Key και Newark element14. Είναι εύκολη στη χρήση της και έρχεται με εγκατεστημένη έκδοση λειτουργικού συστήματος Linux, το Angstrom.



Εικόνα 3.5: Πλατφόρμα μικροελεγκτή Beaglebone Black

Στην εικόνα 3.5 φαίνεται η πλατφόρμα μικροελεγκτή Beaglebone Black, όπως επίσης και κάποια από τα βασικά στοιχεία της. Η πλατφόρμα έχει θύρα USB για σύνδεση με ηλεκτρονικό υπολογιστή, θύρα ethernet για σύνδεση στο διαδίκτυο μέσω καλωδίου, θύρα micro HDMI για σύνδεση σε εξωτερικές οθόνες, και 92 εισδοχές για σύνδεση με εξωτερικές συσκευές ή αισθητήρες.

Βασικά χαρακτηριστικά της πλατφόρμας είναι 1GHz επεξεργαστής, 512MB DDR3 RAM, 3D ενισχυτή γραφικών (3D graphics accelerator) καθώς και 2 PRU 32-bit

microcontrollers. Λόγω των απίστευτων χαρακτηριστικών και δυνατοτήτων της, η πλατφόρμα χρησιμοποιείται πολύ στη βιομηχανία για κατασκευή ρομποτικών συστημάτων, ανάλυση δεδομένων και αναγνώριση αντικειμένων στο περιβάλλον.

Ένα από τα βασικά πλεονεκτήματα της είναι ότι περιλαμβάνει εισδοχές εισόδου αναλογικού σήματος, κάτι το οποίο έλειπε από την πλατφόρμα μικροελεγκτή Raspberry Pi, με αποτέλεσμα να μην χρειάζεται επιπλέον υλικό κατά τη συναρμολόγηση κυκλωμάτων αναλογικών αισθητήρων.

P9					P8			
Function	Physical Pins		Function		Function	Physical Pins		Function
DGND	1	2	DGND		DGND	1	2	DGND
VDD 3.3 V	3	4	VDD 3.3 V		MMC1_DAT6	3	4	MMC1_DAT7
VDD 5V	5	6	VDD 5V		MMC1_DAT2	5	6	MMC1_DAT3
SYS 5V	7	8	SYS 5V		GPIO_66	7	8	GPIO_67
PWR_BTN	9	10	SYS_RESET		GPIO_69	9	10	GPIO_68
UART4_RXD	11	12	GPIO_60		GPIO_45	11	12	GPIO_44
UART4_TXD	13	14	EHRPWM1A		EHRPWM2B	13	14	GPIO_26
GPIO_48	15	16	EHRPWM1B		GPIO_47	15	16	GPIO_46
SPI0_CS0	17	18	SPI0_D1		GPIO_27	17	18	GPIO_65
I2C2_SCL	19	20	I2C_SDA		EHRPWM2A	19	20	MMC1_CMD
SPI0_DO	21	22	SPI0_SLCK		MMC1_CLK	21	22	MMC1_DAT5
GPIO_49	23	24	UART1_TXD		MMC1_DAT4	23	24	MMC1_DAT1
GPIO_117	25	26	UART1_RXD		MMC1_DAT0	25	26	GPIO_61
GPIO_115	27	28	SPI1_CS0		LCD_VSYNC	27	28	LCD_PCLK
SPI1_DO	29	30	GPIO_112		LCD_HSYNC	29	30	LCD_AC_BIAS
SPI1_SCLK	31	32	VDD_ADC		LCD_DATA14	31	32	LCD_DATA15
AIN4	33	34	GND_ADC		LCD_DATA13	33	34	LCD_DATA11
AIN6	35	36			LCD_DATA12	35	36	LCD_DATA10
AIN2	37	38	AIN3		LCD_DATA8	37	38	LCD_DATA9
AIN0	39	40	AIN1		LCD_DATA6	39	40	LCD_DATA7
GPIO_20	41	42	ECAPWMO		LCD_DATA4	41	42	LCD_DATA5
DGND	43	44	DGND		LCD_DATA2	43	44	LCD_DATA3
DGND	45	46	DGND		LCD_DATA0	45	46	LCD_DATA1

LEGEND	
Power, Ground, Reset	
Digital Pins	
PWM Output	
1.8 Volt Analog Inputs	
Shared I2C Bus	
Reconfigurable Digital	

Εικόνα 3.6: Διάγραμμα εισδοχών πλατφόρμας Beaglebone Black

Στην εικόνα 3.6 [17] φαίνονται διαγραμματικά οι 92 εισδοχές της πλατφόρμας, όπως επίσης και οι λειτουργίες τους. Με κόκκινο χρώμα αντιστοιχούν οι εισδοχές για ηλεκτροδότηση εξωτερικών συσκευών και αισθητήρων, με πράσινο χρώμα οι εισδοχές εισόδου και εξόδου ψηφιακού σήματος, με μωβ οι εισδοχές προσομοίωσης αναλογικής εξόδου και με γαλάζιο οι εισδοχές εισόδου αναλογικού σήματος.

3.1.4 Πλατφόρμα μικροελεγκτή Intel Galileo

Η πλατφόρμα μικροελεγκτή Intel Galileo [18] (εικόνα 3.7) είναι η τελευταία πλατφόρμα μικροελεγκτή η οποία μελετήθηκε και χρησιμοποιήθηκε στα πλαίσια της παρούσας διπλωματικής εργασίας. Πρόκειται για την πρώτη πλατφόρμα βασισμένη στην αρχιτεκτονική της Intel και εμπεριέχει τον επεξεργαστή Intel Quark SoC X1000.



Εικόνα 3.7: Πλατφόρμα μικροελεγκτή Intel Galileo

Σχεδιάστηκε με τρόπο ώστε να είναι συμβατή με όλα τα shields του Arduino Uno. Αυτό σημαίνει ότι περιλαμβάνει την ίδια ακριβώς διάταξη εισδοχών όπως το Arduino Uno, δηλαδή 14 εισδοχές εισόδου-εξόδου ψηφιακού σήματος, 6 εισδοχές εισόδου αναλογικού σήματος και 6 εισδοχές οι οποίες προσομοιώνουν έξοδο αναλογικού σήματος (PWM). Επίσης, μπορεί να προγραμματιστεί μέσω του Arduino IDE, το οποίο κάνει την χρήση της συγκεκριμένης πλατφόρμας εξαιρετικά εύκολη και ο προγραμματισμός της μπορεί να γίνει σε όλα τα λειτουργικά συστήματα (Windows, OS, Linux).

Υπάρχουν δύο εκδόσεις της συγκεκριμένης πλατφόρμας, η Intel Galileo Gen 1 και η Intel Galileo Gen 2. Και οι δύο εκδόσεις την συγκεκριμένης πλατφόρμας έχουν παρόμοια χαρακτηριστικά, όπως τον επεξεργαστή Intel Quark X1000, ο οποίος τρέχει στα 400MHz, 256MB DDR3 RAM και 8MB ενσωματωμένη μνήμη (flash memory). Τέτοιου είδους χαρακτηριστικά κάνουν τη πλατφόρμα αρκετά πιο ισχυρή σε σύγκριση

με την Arduino Uno. Τέλος, το ενσωματωμένο ρολόι πραγματικού χρόνου (real time clock) το οποίο είναι ενσωματωμένο στο Intel Galileo βοηθάει στο να διατηρείται ακριβής μέτρηση του χρόνου χωρίς να υπάρχει συνδεδεμένη πηγή ενέργειας ή διαδίκτυο.

3.2 Αισθητήρες

Αισθητήρας, όπως αναφέρθηκε και σε προηγούμενο κεφάλαιο, είναι μια συσκευή η οποία ανιχνεύει ένα φυσικό μέγεθος και παράγει μια μετρήσιμη έξοδο. Σε αυτό το υποκεφάλαιο θα αναλυθούν 6 αισθητήρες οι οποίοι υποστηρίζονται από το μηχανισμό παραγωγής κώδικα. Σημειώνεται ότι οι αισθητήρες αυτοί δεν χρησιμοποιούνται σε όλες τις πλατφόρμες μικροελεγκτών το ίδιο. Μπορεί κάποιος από αυτούς να μην υποστηρίζονται καθόλου ή να αλλάζει η διαδικασία κατά την οποία λαμβάνουν μετρήσεις από το περιβάλλον. Περισσότερες λεπτομέρειες για αυτό θα δοθούν σε επόμενο κεφάλαιο.

3.2.1 Αισθητήρας κίνησης PIR



Εικόνα 3.8: Αισθητήρας κίνησης PIR

Ο PIR (Passive Infrared) αισθητήρας [19] (εικόνα 3.8) είναι ψηφιακός αισθητήρας ο οποίος ανιχνεύει κίνηση στο περιβάλλον το οποίο βρίσκεται μέσω υπέρυθρης

ακτινοβολίας (infrared light) η οποία εκπέμπεται από αντικείμενα τα οποία βρίσκονται στο πεδίο ορατότητας του. Το πεδίο ορατότητας του αισθητήρα φτάνει τα 6 μέτρα και έχει γωνία ορατότητας $110^{\circ} \times 70^{\circ}$. Εκπέμπει σήμα υπό τη μορφή ψηφιακού παλμού με τιμή 0 όταν δεν ανιχνεύεται κίνηση και τιμή 1 όταν ανιχνεύεται κίνηση.

Οι αισθητήρες κίνησης PIR είναι μικροί σε μέγεθος, φθηνοί και δεν καταναλώνουν αρκετή ενέργεια. Έχουν τρία πόδια, ένα για ηλεκτροδότηση, ένα για γείωση και ένα για την έξοδο ψηφιακού σήματος. Χρησιμοποιούνται κυρίως για ανίχνευση κίνησης ανθρώπου σε ένα χώρο και μπορεί να τους συναντήσει κανείς σε διάφορες συσκευές σε ένα σπίτι ή οργανισμούς και επιχειρήσεις. Επίσης, σημειώνεται ότι είναι κατάλληλοι τόσο για εσωτερικούς όσο και εξωτερικούς χώρους.

Ο συνδυασμός τους με άλλες συσκευές ή αισθητήρες δημιουργεί συστήματα τα οποία μπορούν να αυτοματοποιήσουν διαδικασίες ή να κάνουν συγκεκριμένες προγραμματισμένες ενέργειες, όπως για παράδειγμα ένα σύστημα συναγερμού το οποίο ενεργοποιεί μια συσκευή εκπομπής ήχου ότι ανιχνευθεί κίνηση σε ένα χώρο.

3.2.2 Αισθητήρας φωτεινότητας LDR



Εικόνα 3.9: Αισθητήρας φωτεινότητας LDR

Ο αισθητήρας φωτεινότητας LDR (Light-Depended Resistor) [20] (εικόνα 3.9) ή φωτοαντιστάτης είναι αναλογικός αισθητήρας ο οποίος μπορεί να ανιχνεύσει την

παρουσία φωτός σε ένα χώρο. Στην πραγματικότητα πρόκειται για ένα αντιστάτη μεταβλητής αντίστασης, του οποίου η αντίσταση μειώνεται όταν το φως κτυπά την επιφάνεια του και αυξάνεται στο σκοτάδι, παράγοντας αναλογικό σήμα

Χρησιμοποιείται σε κυκλώματα όπου υπάρχει ανάγκη ανίχνευσης φωτός ή σκότους ανάλογα με την περίπτωση. Είναι πολύ μικρός σε μέγεθος, φτηνός και υποστηρίζεται από όλες τις πλατφόρμες μικροελεγκτή που έχουν αναλυθεί.

3.2.3 Αισθητήρας θερμοκρασίας DS18B20

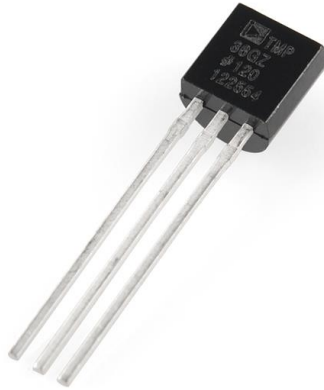


Εικόνα 3.10: Αισθητήρας θερμοκρασίας DS18B20

Ο αισθητήρας DS18B20 [22] είναι ψηφιακός αισθητήρας ο οποίος μετράει τη θερμοκρασία σε βαθμούς Κελσίου από -55°C μέχρι 125°C ($\pm 0.5^{\circ}\text{C}$). Υπάρχουν δύο είδη αισθητήρων DS18B20 εκ των οποίων ο ένας είναι αδιάβροχος και ο άλλος όχι. Στα πλαίσια της παρούσας διπλωματικής εργασίας χρησιμοποιήθηκε ο αδιάβροχος αισθητήρας, ο οποίος φαίνεται στην εικόνα 3.10.

Ο αισθητήρας αποτελείται από μια μεταλλική επιφάνεια που μετράει την θερμοκρασία του περιβάλλοντος και από τρία καλώδια, ένα για την ηλεκτροδότηση του αισθητήρα, ένα για την γείωση του και ένα για έξοδο ψηφιακού σήματος. Δίνει αρκετά ακριβείς μετρήσεις με μέγιστη τιμή λάθους 0.5°C . Χρησιμοποιείται σε

3.2.4 Αισθητήρας θερμοκρασίας TMP36



Ο αισθητήρας TMP36 [23] (εικόνα 3.11) είναι ο δεύτερος αισθητήρας θερμοκρασίας ο οποίος χρησιμοποιείται στα πλαίσια της παρούσας διπλωματικής εργασίας. Πρόκειται για ένα αναλογικό αισθητήρα θερμοκρασίας, χαμηλού κόστους και δίνει ως έξοδο τάση ρεύματος (voltage), η οποία μπορεί στη συνέχεια να μεταφραστεί σε βαθμούς κελσίου.

33

3.2.5 Αισθητήρας ήχου

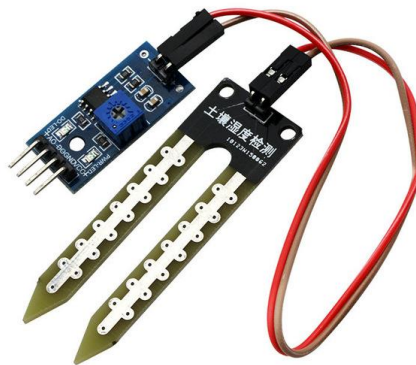
Ο αισθητήρας ήχου [24] (εικόνα 3.12) μπορεί να αντιληφθεί τα επίπεδα ήχου που υπάρχουν σε ένα χώρο. Είναι μικρός σε μέγεθος, φθηνός, εύκολος στη χρήση του και δεν καταναλώνει αρκετή ενέργεια.



Εικόνα 3.12: Αισθητήρας ήχου

Ο συγκεκριμένος αισθητήρας περιλαμβάνει ένα μικρόφωνο το οποίο ανιχνεύει τον ήχο που υπάρχει στο περιβάλλον, ένα ποτενσιόμετρο, μέσω του οποίου μπορεί να ρυθμιστεί η ευαισθησία του και τέσσερα πόδια, ένα για την ηλεκτροδότηση του, ένα για την γείωση του, ένα για έξοδο αναλογικού σήματος και ένα για έξοδο ψηφιακού σήματος.

3.2.6 Αισθητήρας υγρασίας FC-28



Εικόνα 3.13: Αισθητήρας υγρασίας FC-28

Ο αισθητήρας FC-28 (εικόνα 3.13) [25] είναι αισθητήρας ανίχνευσης υγρασίας εδάφους. Αυτό σημαίνει ότι μπορεί να μετρήσει το ογκομετρικό περιεχόμενο του νερού που βρίσκεται σε έδαφος και όχι την υγρασία που υπάρχει στην ατμόσφαιρα. Μπορεί να χρησιμοποιηθεί σε γεωργικές καλλιέργειες για αυτοματοποίηση της διαδικασίας ύδρευσης όταν για παράδειγμα τα επίπεδα υγρασίας του εδάφους φτάσουν κάτω από ένα συγκεκριμένο όριο.

Ο τρόπος με τον οποίο δουλεύει είναι απλός. Οι ανιχνευτές του αισθητήρα όταν έρθουν σε επαφή με το έδαφος μπορούν να μετρήσουν την υγρασία μέσω της αντίστασης του ρεύματος. Όταν υπάρχει νερό στο έδαφος μεταφέρεται περισσότερος ηλεκτρισμός άρα μικρότερη αντίσταση, ενώ αν το έδαφος είναι ξηρό μεταφέρεται λιγότερος ηλεκτρισμός άρα μεγαλύτερη αντίσταση.

Ο συγκεκριμένος αισθητήρας είναι μικρός, φθηνός και απλός στη χρήση του. Έχει δύο ανιχνευτές, ένα ποτενσιόμετρο μέσω του οποίου μπορεί να αλλάξει η ευαισθησία του και τέσσερα πόδια, ένα για την ηλεκτροδότηση του, ένα για την γείωση του, ένα για έξοδο αναλογικού σήματος και ένα για έξοδο ψηφιακού σήματος.

3.3 Επιπλέον εξαρτήματα

Σε αυτό το υποκεφάλαιο θα γίνει αναφορά σε επιπλέον υλικό το οποίο χρειάστηκε για την δημιουργία του μηχανισμού παραγωγής κώδικα. Συγκεκριμένα θα παρουσιαστούν δύο ασπίδες (shields), οι οποίες συνδέονται στην πλατφόρμα μικροελεγκτή Arduino Uno, καθώς και κάποιες κεραίες μαζί με μια κάρτα διαδικτύου για σύνδεση του Intel Galileo στο Wi-Fi.

Γενικά υπάρχουν πολλά shields τα οποία μπορούν να εγκατασταθούν πάνω στο Arduino προσφέροντας του επιπλέον λειτουργικότητα. Είναι εύκολη η χρήση τους, αφού λόγω της κατασκευής τους, δεν χρειάζονται επιπλέον καλώδια ή αντιστάσεις για να δουλέψουν και τροφοδοτούνται μέσω της πλατφόρμας μικροελεγκτή.

3.3.1 Wi-Fi Shield CC3000

Για τη σύνδεση της πλατφόρμας μικροελεγκτή Arduino Uno με το Wi-Fi χρησιμοποιήθηκε η ασπίδα (shield) Adafruit CC3000 [26]. Κατασκευάστηκε από την εταιρία Texas Instruments (εταιρία κατασκευής της πλατφόρμας μικροελεγκτή Beaglebone Black) και ενώνεται στο Arduino Uno προσφέροντας του την δυνατότητα να συνδεθεί στο διαδίκτυο μέσω Wi-Fi.



Εικόνα 3.14: Adafruit CC3000 Wi-Fi shield

Στην εικόνα 3.14 φαίνεται η ασπίδα τοποθετημένη στο Arduino Uno. Σημειώνεται ότι κατά την εφαρμογή της ασπίδας στην πλατφόρμα όλες οι εισδοχές της πλατφόρμας είναι διαθέσιμες με ακριβώς την ίδια αναδιάταξη.

3.3.2 Ethernet Shield

Άλλη μια ασπίδα για την σύνδεση της πλατφόρμας μικροελεγκτή Arduino με το διαδίκτυο είναι η ethernet shield [31] (εικόνα 3.15). Η συγκεκριμένη ασπίδα προσθέτει στην πλατφόρμα μικροελεγκτή μια θύρα ethernet και επιτρέπει την σύνδεση της σε router μέσω ethernet καλωδίου.



Εικόνα 3.15: Ethernet shield

Ο τρόπος τοποθέτησης της στην πλατφόρμα μικροελεγκτή είναι πολύ απλός και δεν απαιτεί την σύνδεση οποιουδήποτε επιπλέον εξαρτήματος. Για να γίνει εφικτή η χρήση της χρειάζεται η εγκατάσταση και χρησιμοποίηση της βιβλιοθήκης Ethernet.

3.3.3 Wireless card και αντένες Wi-Fi

Για τη σύνδεση της πλατφόρμας μικροελεγκτή Intel Galileo στο Wi-Fi, χρησιμοποιείται η wireless κάρτα Intel Centrino Wireless-N 135 σε συνδυασμό με αντένες Wi-Fi για ενίσχυση του σήματος.



Εικόνα 3.16: Wireless card Intel Centrino Wireless-N 135 με Wi-Fi αντένες

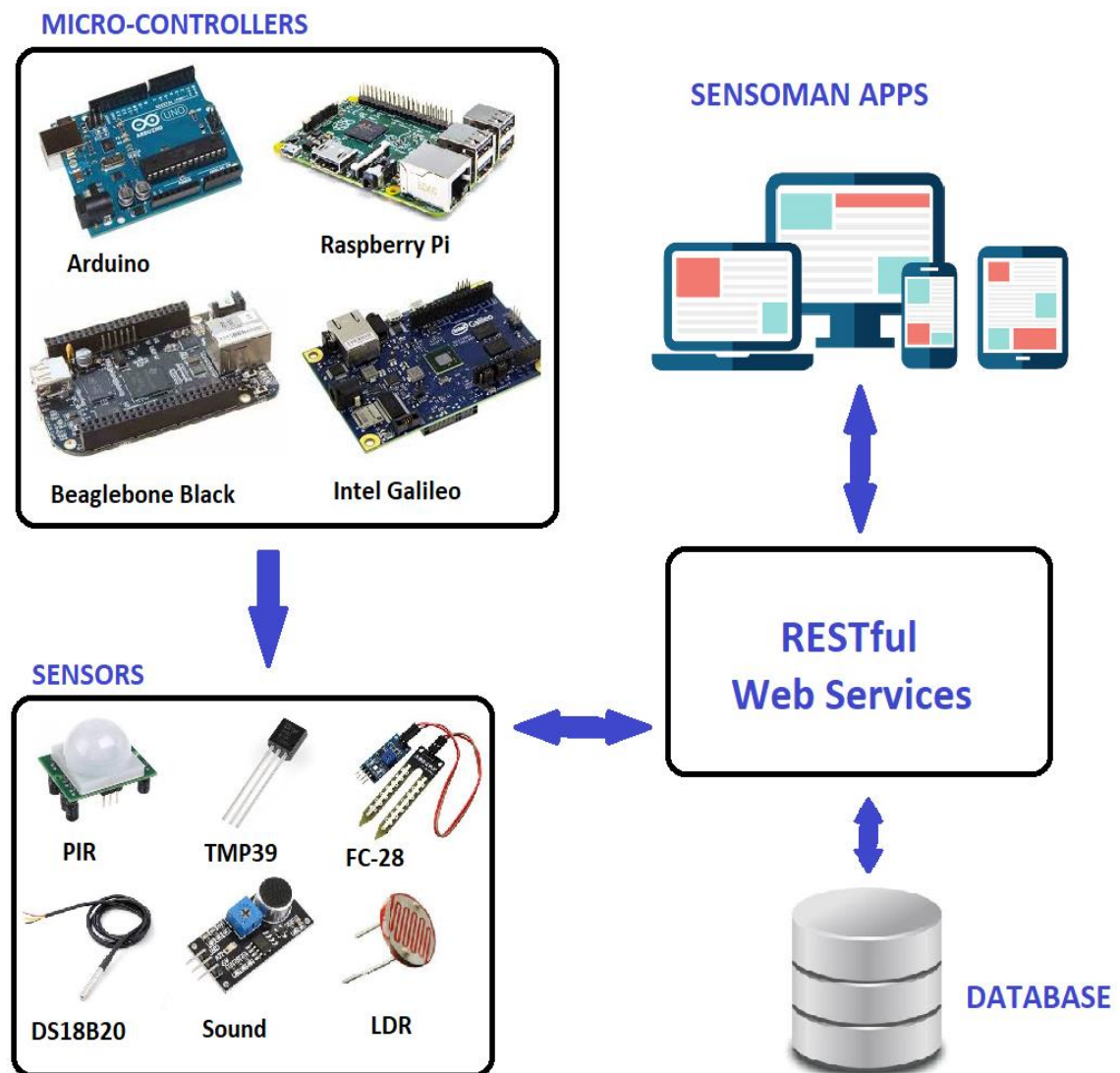
Κεφάλαιο 4

Σύστημα SensoMan

Σύστημα SensoMan	38
4.1 Αρχιτεκτονική Συστήματος	39
4.2 Βάση Δεδομένων	42
4.3 SensoMan RESTful υπηρεσία	46
4.4 Υποσυστήματα του SensoMan	50
4.4.1 Διαδικτυακή Εφαρμογή SensoMan	51
4.4.2 Κινητή Εφαρμογή SensoMan	56
4.4.3 Μηχανή Κανόνων	56
4.4.4 Κοινωνικό Δίκτυο	57

4.1 Αρχιτεκτονική Συστήματος

Το σύστημα SensoMan, όπως αναφέρθηκε και σε προηγούμενο κεφάλαιο δημιουργήθηκε στα πλαίσια παλιών διπλωματικών εργασιών και συνεχίζει να επεκτείνεται κάθε χρόνο. Στόχος του συστήματος είναι να προσφέρει ένα ολοκληρωμένο τρόπο παρακολούθησης και διαχείρισης μικροελεγκτών και αισθητήρων, οι οποίοι είναι τοποθετημένοι σε διάφορες τοποθεσίες.



Εικόνα 4.1: Αρχιτεκτονική Συστήματος SensoMan

Στην εικόνα 4.1 απεικονίζεται η αρχιτεκτονική του συστήματος SensoMan, τα κομμάτια από τα οποία αποτελείται και πως συνδέονται μεταξύ.

Η υποδομή του συστήματος SensoMan απαρτίζεται από τα εξής πέντε βασικά λειτουργικά στοιχεία:

- Πλατφόρμες μικροελεγκτών (Micro-controllers)
- Αισθητήρες (Sensors)
- Εφαρμογές SensoMan (SensoMan Apps)
- SensoMan RESTful υπηρεσίες διαδικτύου (RESTful Web Services)
- Βάση Δεδομένων (Database)

Η αρχιτεκτονική αυτή, χαρακτηρίζεται ως επεκτάσιμη λόγω του ότι περιέχει αρκετά στοιχεία τα οποία μπορούν να τροποποιηθούν και να αναπτυχθούν ανεξάρτητα, με καθόλου ή ελάχιστη αλλαγή στα άλλα μέρη του συστήματος. Για παράδειγμα, μπορεί να γίνεται επέκταση του συστήματος προσθέτοντας νέες πλατφόρμες μικροελεγκτών, νέους αισθητήρες, νέες λειτουργικότητες στην διαδικτυακή εφαρμογή, χωρίς να αλλάξουν στοιχεία που αφορούν τη RESTful υπηρεσία διαδικτύου ή τη βάση δεδομένων.

Τα πέντε ξεχωριστά κομμάτια (components) του συστήματος συνδέονται, επικοινωνούν και ανταλλάζουν πληροφορίες, δημιουργώντας μια σχέση μεταξύ τους. Η σχέση αυτή μπορεί να χαρακτηριστεί ως άμεση (direct relationship) ή έμμεση (indirect relationship).

Μέσω της άμεσης σχέσης, τα δύο στοιχεία επικοινωνούν μεταξύ τους απευθείας, ανταλλάζοντας δεδομένα και πληροφορίες, χωρίς να χρειάζεται κάποιος μεσολαβητής στο να μεταφέρει τα μηνύματα. Παράδειγμα αυτής της σχέσης αποτελεί η πλατφόρμα μικροελεγκτή με ένα αισθητήρα. Σε αυτή τη περίπτωση οι μετρήσεις που λαμβάνονται από τους αισθητήρες αποστέλλονται απευθείας στην πλατφόρμα μικροελεγκτή για επεξεργασία.

Μέσω της έμμεσης σχέσης, τα δύο στοιχεία επικοινωνούν μεταξύ τους δια μέσου ενός μεσολαβητή. Για την επικοινωνία αυτή πραγματοποιείται ένας κύκλος δράσεων και αποστολής μηνυμάτων. Αρχικά το πρώτο στοιχείο της σχέσης αποστέλλει ένα αίτημα στο μεσολαβητή για πραγματοποίηση κάποιας ενέργειας. Ο μεσολαβητής λαμβάνει το αίτημα αυτό, το επεξεργάζεται και το αποστέλλει στο δεύτερο στοιχείο της σχέσης. Το δεύτερο στοιχείο της σχέσης πραγματοποιεί την ενέργεια και αποστέλλει, με τη σειρά του, μήνυμα απάντησης στο πρώτο στοιχείο δια μέσω του μεσολαβητή. Παράδειγμα αυτής της σχέσης αποτελεί η πλατφόρμα μικροελεγκτή με τη βάση δεδομένων, όπου ο μεσολαβητής είναι η RESTful υπηρεσία διαδικτύου (θα γίνει εκτενέστερη περιγραφή της επικοινωνίας αυτής στο υποκεφάλαιο 4.3).

Για την πραγματοποίηση των λειτουργιών και αναγκών του συστήματος ακολουθείται μια ροή δραστηριοτήτων, δηλαδή μια σειρά ενεργειών που εκτελείται η μια μετά την άλλη, για να φέρουν το σύστημα σε μια νέα κατάσταση. Αυτή η ροή δραστηριοτήτων περιλαμβάνει την επίδραση και των πέντε στοιχείων του συστήματος για να πραγματοποιηθεί. Ο τρόπος λειτουργίας του συστήματος περιγράφεται συνοπτικά πιο κάτω.

Αρχικά οι διάφορες πλατφόρμες μικροελεγκτών τοποθετούνται σε διάφορες τοποθεσίες, όπου ο χρήστης θέλει να κάνει ανάλυση μετρήσεων στο χώρο και να αυτοματοποιήσει διάφορες ενέργειες. Κάθε πλατφόρμα μικροελεγκτή περιέχει ένα αριθμό από αισθητήρες συνδεδεμένους στις ανάλογες εισδοχές. Η πλατφόρμα μικροελεγκτή ενεργοποιείται και εκτελεί το πρόγραμμα που έχει εγκατασταθεί σε αυτήν.

Το πρόγραμμα αυτό είναι υπεύθυνο στο να θέσει σε λειτουργία τους αισθητήρες, να ορίσει το είδος εισόδου του καθ' ενός από αυτούς, να συλλάβει και να αναλύσει την μέτρηση του. Επίσης μέσω του προγράμματος καθορίζεται ο χρόνος κατά τον οποίο ο μικροελεγκτής θα παραμείνει αδρανής και έπειτα θα ξεκινήσει από την αρχή, κάνοντας ακριβώς την ίδια διαδικασία.

Ακολουθώς, οι μετρήσεις του κάθε αισθητήρα αποστέλλονται και αποθηκεύονται στη βάση δεδομένων διαμέσου της RESTful υπηρεσίας διαδικτύου. Ο χρήστης έπειτα μπορεί χρησιμοποιώντας την διαδικτυακή και την κινητή εφαρμογή του SensoMan να δει και να διαχειριστεί πληροφορίες που αφορούν τις πλατφόρμες μικροελεγκτών του, τους αισθητήρες του και τις μετρήσεις.

Τα δύο από τα πέντε στοιχεία του συστήματος (πλατφόρμες μικροελεγκτών και αισθητήρες) έχουν περιγραφεί στο κεφάλαιο 3. Ακολουθεί ανάλυση της βάσης δεδομένων, της RESTful υπηρεσίας διαδικτύου και των υποσυστημάτων του SensoMan.

4.2 Βάση Δεδομένων

Η βάση δεδομένων του συστήματος SensoMan χρησιμοποιείται για την αποθήκευση των δεδομένων και πληροφοριών που χρειάζεται το σύστημα να λειτουργήσει. Συγκεκριμένα πρόκειται για βάση δεδομένων MySQL [4], η οποία είναι μια από τις δημοφιλέστερες βάσεις δεδομένων ανοικτού κώδικα στο κόσμο. Βρίσκεται εγκατεστημένη σε εξυπηρετητή του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου.

Οι πληροφορίες που αποθηκεύονται σε αυτή αφορούν τους χρήστες του συστήματος, τους αισθητήρες, τις πλατφόρμες μικροελεγκτών, τις μετρήσεις των αισθητήρων, τις τοποθεσίες των μικροελεγκτών, τους κανόνες, το κοινωνικό δίκτυο και τα αρχεία του κάθε χρήστη. Πιο κάτω περιγράφονται μερικοί από τους πίνακες της βάσης δεδομένων.

Ο πίνακας ο οποίος περιέχει πληροφορίες που αφορούν το χρήστη ονομάζεται “user”. Σε αυτόν αποθηκεύονται στοιχεία όπως το username του, το email του, ο κωδικός πρόσβασης του στο σύστημα και ο τύπος του. Σημειώνεται ότι ο τύπος του κάθε χρήστη μπορεί να είναι Απλός (Simple), Ιδιοκτήτης (Owner), Διαχειριστής (Admin) και Ιδιοκτήτης-Διαχειριστής (Owner-Admin). Μέσω των τύπων, η διαδικτυακή εφαρμογή

ελέγχει την πρόσβαση του κάθε χρήστη και δίνει διαφορετικές πληροφορίες και δυνατότητες σε κάθε ένα που συνδέεται στο σύστημα.

Ο χρήστης για να εισέλθει στο σύστημα πρέπει να εισάγει το username του και το κωδικό πρόσβασης του, τα οποία μπορεί να αλλάξει αν επιθυμεί στη συνέχεια μέσω των ρυθμίσεων της εφαρμογής. Όταν τα στοιχεία του χρήστη ελεγχθούν και είναι ορθά, τότε παράγεται από το σύστημα ένα “token” το οποίο είναι μια τυχαία συμβολοσειρά η οποία αποθηκεύεται στη βάση δεδομένων (πίνακας “token”) και αντιστοιχεί στον συνδεδεμένο χρήστη. Το token αυτό χρησιμοποιείται για την επικύρωση του χρήστη, για την είσοδο του στις όψεις (views) του συστήματος και έχει διάρκεια ζωής μισής ώρας. Όταν λήξει το token, ο χρήστης πρέπει να ξαναδώσει τα στοιχεία του για να συνδεθεί. Σημειώνεται ότι όταν ο χρήστης αποσυνδεθεί από το σύστημα διαγράφεται το προηγούμενο token του από τη βάση δεδομένων.

Ο πίνακας “micro-controller” περιέχει πληροφορίες που αφορούν τις πλατφόρμες μικροελεγκτών. Σε αυτόν αποθηκεύονται στοιχεία όπως το όνομα του μικροελεγκτή, ο τύπος του (Arduino, Raspberry Pi, Beaglebone Black, Intel Galileo), η IP διεύθυνση του και το port στο οποίο ακούει.

Ο πίνακας “user-controller” είναι βοηθητικός πίνακας ο οποίος αντιστοιχεί την κάθε πλατφόρμα μικροελεγκτή με το χρήστη στον οποίο ανήκει. Η αντιστοίχιση αυτή είναι σχέση ένα προς πολλά (one to many) και χρησιμοποιεί τα ids (μοναδικά κλειδιά) των δύο οντοτήτων. Κατά την εισαγωγή μιας νέας πλατφόρμας μικροελεγκτή, εντοπίζεται το id του χρήστη μέσω του token του και δημιουργείται μια νέα εγγραφή σε αυτόν τον πίνακα.

Στο πίνακα “type-sensor” αποθηκεύονται όλα τα είδη αισθητήρων που υποστηρίζονται από το σύστημα μαζί με ένα μοναδικό αριθμό. Συγκεκριμένα υπάρχουν πέντε είδη και αριθμούνται ως εξής: 1 – humidity (υγρασία), 2 – light (φωτεινότητα), 3 -sound (ήχος), 4 – temperature (θερμοκρασία), 5 – motion (κίνηση).

Ο πίνακας “sensor” περιέχει πληροφορίες που αφορούν τους αισθητήρες. Αποθηκεύεται το όνομα του αισθητήρα, το είδος του (αριθμός 1 μέχρι 5), και ο τύπος του (Humidity FC28, Temperature TMP36, Temperature DS18B20, Motion PIR, Light LDR, Sound analog, Sound digital).

Ο πίνακας “sensor-controller” είναι βοηθητικός πίνακας ο οποίος αντιστοιχεί τον κάθε αισθητήρα με την πλατφόρμα μικροελεγκτή στην οποία είναι τοποθετημένος. Η αντιστοίχιση αυτή είναι σχέση ένα προς πολλά (one to many) και χρησιμοποιεί τα ids (μοναδικά κλειδιά) των δύο οντοτήτων. Κατά την εισαγωγή ενός νέου αισθητήρα, επιλέγεται ο μικροελεγκτής στο οποίο θα ενσωματωθεί και έτσι δημιουργείται μια νέα εγγραφή σε αυτόν το πίνακα.

Στο πίνακα “sensor-measurement” περιέχονται όλες οι μετρήσεις οι οποίες λαμβάνονται από τους αισθητήρες. Συγκεκριμένα, κάθε εγγραφή του πίνακα αποτελείται από το id του αισθητήρα, τη μέτρηση του και τη ημερομηνία στην οποία λήφθηκε (datetime).

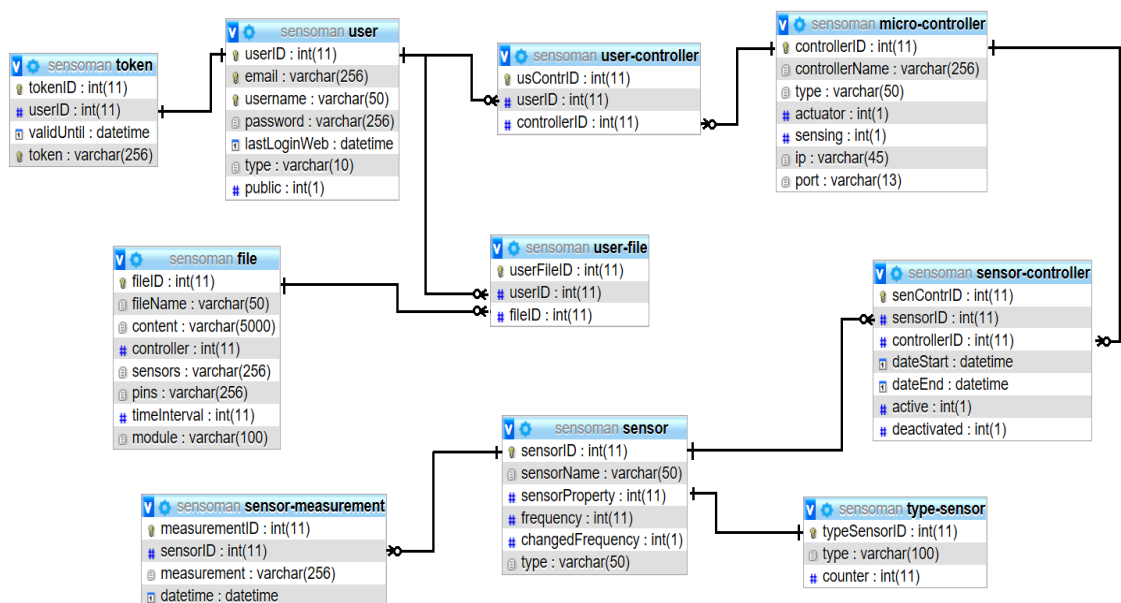
Στο πίνακα “rule” περιέχονται πληροφορίες που αφορούν τη μηχανή κανόνων που δημιουργήθηκε στα πλαίσια της διπλωματικής εργασίας του Φίλιππου Αζίζ [30]. Αποθηκεύονται πληροφορίες όπως ο κανόνας σε XML μορφή, η περιγραφή του και η ημερομηνία έκδοσης του.

Οι πίνακες “sn_association”, “sn_group”, “sn_notification”, “sn_request”, “sn_requested_sensors”, “sn_shared_boards”, “sn_shared_sensors” “sn_user-group” και “sn_user_visible_controller” δημιουργήθηκαν στα πλαίσια την διπλωματικής εργασίας του φοιτητή Άγγελου Ευσταθίου [29] και αφορούν το κοινωνικό δίκτυο του SensoMan.

Ο πίνακας “file” περιέχει πληροφορίες που αφορούν τα αρχεία κώδικα που παράγονται από την μηχανή παραγωγής κώδικα. Αποθηκεύονται στοιχεία όπως το όνομα του αρχείου, η πλατφόρμα μικροελεγκτή για την οποία δημιουργήθηκε το

συγκεκριμένο αρχείο, το είδος της πλατφόρμας, οι αισθητήρες που είναι τοποθετημένοι στην πλατφόρμα, οι εισδοχές (pins) του κάθε αισθητήρα, ο χρόνος επανάληψης της διαδικασίας συλλογής μετρήσεων (time interval) και αν η πλατφόρμα είναι συνδεδεμένη στο διαδίκτυο ή όχι.

Ο πίνακας “user-file” είναι βοηθητικός πίνακας ο οποίος αντιστοιχεί το κάθε αρχείο με το χρήστη στον οποίο ανήκει. Η αντιστοίχιση αυτή είναι σχέση ένα προς πολλά (one to many) και χρησιμοποιεί τα ids (μοναδικά κλειδιά) των δύο οντοτήτων. Κατά την δημιουργία νέου αρχείου, εντοπίζεται το id το χρήστη μέσω του token του και δημιουργείται μια νέα εγγραφή σε αυτό τον πίνακα. Περισσότερες πληροφορίες για τους δύο αυτούς πίνακες θα δοθούν στο κεφάλαιο 5 όπου θα αναλυθεί η μηχανή παραγωγής κώδικα.



Εικόνα 4.2: Σχεσιακό Διάγραμμα Βάσης Δεδομένων

Η εικόνα 4.2 παρουσιάζει το σχεσιακό διάγραμμα για μερικούς από τους πίνακες της βάσης δεδομένων που αναλύθηκαν πιο πάνω και διακρίνονται τα διάφορα πεδία του

κάθε ενός από αυτούς, το πεδίο ορισμού τους και οι σχέσεις που έχουν μεταξύ τους (ένα προς ένα, ένα προς πολλά).

Επίσης από το σχεσιακό διάγραμμα εξάγονται τα εξής συμπεράσματα:

- Ένα token αντιστοιχεί σε ένα μοναδικό χρήστη
- Ένας χρήστης μπορεί να έχει πολλές (ή και 0) πλατφόρμες μικροελεγκτή
- Μια πλατφόρμα μικροελεγκτή μπορεί να έχει πολλούς (ή και 0) αισθητήρες
- Ένας χρήστης μπορεί να έχει πολλά (ή και 0) αρχεία
- Ένας αισθητήρας έχει ένα μοναδικό είδος
- Ένας αισθητήρας μπορεί να έχει πολλές (ή και 0) μετρήσεις.

4.3 SensoMan RESTful υπηρεσία

Στο κεφάλαιο 2 υποκεφάλαιο 4 παρουσιάστηκε τι είναι οι RESTful υπηρεσίες διαδικτύου και σε αυτό το κεφάλαιο θα γίνει περιγραφή των λειτουργιών που προσφέρει η SensoMan RESTful υπηρεσία.

Η SensoMan RESTful υπηρεσία αποτελεί την γέφυρα επικοινωνίας μεταξύ των εφαρμογών του συστήματος SensoMan (διαδικτυακή και κινητή) και των πλατφόρμων μικροελεγκτών με τη βάση δεδομένων. Όλα τα μηνύματα που μεταφέρονται εντός του συστήματος, από και προς τη βάση δεδομένων, ελέγχονται εκτελούνται και κατευθύνονται από αυτή.

Γενικά τα μηνύματα αυτά χωρίζονται στις εξής κατηγορίες:

- **Upload:** Αφορούν την αποθήκευση νέου στοιχείου στη βάση δεδομένων.
- **Edit:** Αφορούν την επεξεργασία υφιστάμενου στοιχείου στη βάση δεδομένων
- **Delete:** Αφορούν τη διαγραφή κάποιου στοιχείου από τη βάση δεδομένων
- **Retrieve:** Αφορούν τη λήψη στοιχείων από τη βάση δεδομένων

Το κάθε μήνυμα που αποστέλλεται είναι στην ουσία ένα αίτημα HTTP (HTTP request) στην RESTful υπηρεσία διαδικτύου της μορφής GET ή POST. Συνήθως τα GET αιτήματα χρησιμοποιούνται για την λήψη ενός στοιχείου από τη βάση δεδομένων, μέσω του URL (Uniform Resource Locator) της RESTful υπηρεσίας διαδικτύου που προσφέρει αυτή τη δυνατότητα. Για την ανάκτηση ενός εξειδικευμένου στοιχείου χρησιμοποιούνται παράμετροι στο κομμάτι που αφορά το “query” του URL. Τα POST αιτήματα αφορούν συνήθως την αποστολή και επεξεργασία στοιχείων στη βάση δεδομένων. Αποστέλλουν δεδομένα μέσω μιας HTML φόρμας (HTML form) ή μέσω JSON στο σώμα (body) του αιτήματος. Αξίζει να αναφερθεί ότι σχεδόν όλα τα μηνύματα που αποστέλλονται στο σύστημα SensoMan είναι τύπου POST για το λόγο ότι είναι πιο ασφαλή (δεν φαίνονται τα στοιχεία που αποστέλλονται στο URL) και επειδή χρειάζεται κάθε φορά το token του χρήστη για να γίνει κάποια ενέργεια.

Όταν η SensoMan RESTful υπηρεσία λάβει το αίτημα, ελέγχει το token του χρήστη αν είναι έγκυρο και προχωράει σε επικοινωνία με τη βάση δεδομένων. Ανοίγει σύνδεση με τη βάση δεδομένων και εκτελεί το κατάλληλο SQL query το οποίο θα κάνει τις απαραίτητες ενέργειες που χρειάζονται. Αφού γίνει αυτό, δημιουργείται ένα μήνυμα απάντησης (response) το οποίο αποστέλλεται πίσω στον αιτητή της υπηρεσίας.

Η απάντηση αυτή είναι ένα JSON μήνυμα το οποίο αποτελείται από τα εξής στοιχεία:

- **status:** “success” αν η ενέργεια εκτελέσθηκε επιτυχώς, “fail” αν όχι
- **request time:** Η ημερομηνία και ώρα αποστολής της απάντησης
- **IP Address:** Η IP διεύθυνση του αιτητή
- **message:** Επιπρόσθετη πληροφορία για το λόγο αποτυχίας ή επιτυχίας της ενέργειας

Το JSON αυτό μπορεί να περιέχει και επιπρόσθετα στοιχεία ανάλογα με την ενέργεια που εκτελείται κάθε φορά. Για παράδειγμα, σε ένα μήνυμα απάντησης για λήψη των αρχείων κάποιου χρήστη υπάρχει το επιπρόσθετο στοιχείο “files”, που αφορά ένα πίνακα αρχείων για το συγκεκριμένο χρήστη.

Παραδείγματα μηνυμάτων απάντησης φαίνονται στις εικόνες 4.3, 4.4 και 4.5.

```
{
  "status": "success",
  "request time": "2018-05-08 09:05:59",
  "IP Address": "::1",
  "message": "Successfully retrieved",
  "Sensor Info": {
    "sensorID": "158",
    "sensorName": "light1",
    "sensorType": "2",
    "controllerID": "225"
  }
}
```

Εικόνα 4.3: Παράδειγμα απάντησης 1

Η πιο πάνω απάντηση αφορά την επιτυχής λήψη των στοιχείων ενός αισθητήρα.

```
{
  "status": "success",
  "request time": "2018-05-08 09:05:21",
  "IP Address": "::1",
  "message": "Sensor deleted successfully."
}
```

Εικόνα 4.4: Παράδειγμα απάντησης 2

Η πιο πάνω απάντηση αφορά την επιτυχής διαγραφή ενός αισθητήρα.

```
{
  "status": "fail",
  "request time": "2018-05-08 09:05:19",
  "IP Address": "::1",
  "message": "There are some sensors or actuators attached on this controller."
}
```

Εικόνα 4.5: Παράδειγμα απάντησης 3

Η πιο πάνω απάντηση αφορά την αποτυχημένη διαγραφή μιας πλατφόρμας μικροελεγκτή.

Ακολούθως θα γίνει παρουσίαση των RESTful υπηρεσιών που δημιουργήθηκαν και χρησιμοποιήθηκαν στα πλαίσια της παρούσας διπλωματικής εργασίας υπό μορφή πίνακα για κάθε κατηγορία (upload, edit, delete, retrieve). Στη πρώτη στήλη αναφέρεται το όνομα του αρχείου που προσφέρει την υπηρεσία, στη δεύτερη στήλη το είδος της αίτησης (GET ή POST) και στην τρίτη στήλη περιγράφεται η λειτουργία την οποία προσφέρει η συγκεκριμένη υπηρεσία.

RETRIEVE		
Αρχείο	Είδος αίτησης	Λειτουργία
controllerInfo.php	POST	Επιστρέφονται τα στοιχεία ενός μικροελεγκτή.
sensorInfo.php	POST	Επιστρέφονται τα στοιχεία ενός αισθητήρα.
sensorsOfController.php	POST	Επιστρέφονται όλοι οι αισθητήρες ενός μικροελεγκτή.
controllersOfUser.php	POST	Επιστρέφονται όλοι οι μικροελεκτές ενός χρήστη.
sensorsOfUser.php	POST	Επιστρέφονται όλοι οι αισθητήρες ενός χρήστη.
measurementsOfSensor.php	POST	Επιστρέφονται όλες οι μετρήσεις ενός αισθητήρα.
filesOfUser.php	POST	Επιστρέφονται όλα τα αρχεία ενός χρήστη.
checkFilename.php	POST	Γίνεται έλεγχος αν υπάρχει κάποιο όνομα αρχείου για ένα χρήστη.
filesOfController.php	POST	Επιστρέφονται όλα τα αρχεία ενός μικροελεγκτή.

Πίνακας 4.1: Υπηρεσίες κατηγορίας retrieve

UPLOAD		
Αρχείο	Είδος αίτησης	Λειτουργία
upload_controller.php	POST	Αποθήκευση στοιχείων ενός νέου μικροελεγκτή.
upload_sensor.php	POST	Αποθήκευση στοιχείων ενός νέου αισθητήρα.
registration.php	POST	Αποθήκευση στοιχείων ενός νέου χρήστη.

upload_file.php	POST	Αποθήκευση στοιχείων ενός νέου αρχείου.
-----------------	------	---

Πίνακας 4.2: Υπηρεσίες κατηγορίας upload

DELETE		
Αρχείο	Είδος αίτησης	Λειτουργία
delete_controller.php	POST	Διαγραφή μιας πλατφόρμας μικροελεγκτή.
delete_sensor.php	POST	Διαγραφή ενός αισθητήρα και των μετρήσεων του.
delete_token.php	POST	Διαγραφή ενός token που έχει λήξει.
delete_file.php	POST	Διαγραφή ενός αρχείου.

Πίνακας 4.3: Υπηρεσίες κατηγορίας delete

EDIT		
Αρχείο	Είδος αίτησης	Λειτουργία
edit_controller.php	POST	Αλλαγή ονόματος ενός μικροελεγκτή.
edit_sensor.php	POST	Αλλαγή ονόματος ενός αισθητήρα.
edit_password.php	POST	Αλλαγή κωδικού πρόσβασης ενός χρήστη.
edit_username.php	POST	Αλλαγή username ενός χρήστη.
edit_file.php	POST	Αλλαγή χρόνου αποθήκευσης μετρήσεων (time interval) και εισδοχών αισθητήρων (pins) ενός αρχείου.

Πίνακας 4.4: Υπηρεσίες κατηγορίας edit

4.4 Υποσυστήματα του SensoMan

Σε αυτό το υποκεφάλαιο θα γίνει σύντομη επεξήγηση των εφαρμογών και υποσυστημάτων τους SensoMan. Συγκεκριμένα θα επεξηγηθεί η διαδικτυακή εφαρμογή, η κινητή εφαρμογή, η μηχανή κανόνων και το κοινωνικό δίκτυο.

4.4.1 Διαδικτυακή Εφαρμογή SensoMan

Η διαδικτυακή εφαρμογή του συστήματος SensoMan όπως αναφέρθηκε και σε προηγούμενα κεφάλαια αποτελεί το κύριο κομμάτι ολόκληρου του συστήματος. Περιλαμβάνει την υποδομή για τη λειτουργία και διαχείριση των πλατφορμών μικροελεγκτών, των αισθητήρων καθώς και όλων των υποσυστημάτων του SensoMan.

Ο κάθε χρήστης μέσω προσωπικού του λογαριασμού μπορεί να εισέρχεται στην διαδικτυακή εφαρμογή παρουσιάζοντας το username και τον κωδικό πρόσβασης του. Έπειτα μπορεί να βλέπει πληροφορίες που αφορούν τις πλατφόρμες μικροελεγκτών του, τους αισθητήρες του και αρχεία κώδικα που έχει δημιουργήσει (περισσότερες πληροφορίες στο κεφάλαιο 5). Επίσης μπορεί να προσθέσει νέες πλατφόρμες μικροελεγκτών βάσει τοποθεσίας τους στο χάρτη (αν είναι τύπου διαχειριστής ή ιδιοκτήτης) ή και ανεξαρτήτως τοποθεσίας αν το επιθυμεί. Μπορεί να προσθέσει νέους αισθητήρες πάνω σε κάποια πλατφόρμα μικροελεγκτή και να διαγράψει κάποιον αν επιθυμεί.

Ακόμα, μπορεί να δει μετρήσεις οι οποίες λήφθηκαν από τους αισθητήρες του μέσω γραφικών παραστάσεων που δημιουργούνται από τη διαδικτυακή εφαρμογή. Κάθε γραφική παράσταση παρουσιάζει τις μετρήσεις για ένα συγκεκριμένο αισθητήρα, γεγονός το οποίο κάνει εύκολη την παρακολούθηση ενός χώρου για εύρεση ύποπτης κίνησης ή ανεπιθύμητης κατάστασης.

Στα πλαίσια της παρούσας εργασίας εκτός από την προσθήκη της μηχανής παραγωγής κώδικα η οποία θα αναλυθεί στη συνέχεια, προστέθηκαν και κάποιες επιπλέον λειτουργίες. Συγκεκριμένα, στην αρχική σελίδα της εφαρμογής (dashboard), ο χρήστης μπορεί να δει σε μορφή πίνακα τον αριθμό των μικροελεγκτών και των αισθητήρων, τους οποίους έχει στο λογαριασμό του, όπως επίσης και τον αριθμό των αρχείων κώδικα που έχει δημιουργήσει χρησιμοποιώντας τη μηχανή παραγωγής κώδικα.

Controllers:	3	Info	+ Add
Sensors:	6	Info	+ Add
Files:	2	Info	+ Add

Εικόνα 4.6: Πίνακας στο dashboard του χρήστη

Στην εικόνα 4.6 παρουσιάζεται ένα παράδειγμα αυτού του πίνακα. Πατώντας το κουμπί “Info” το οποίο βρίσκεται δίπλα από τους μικροελεγκτές ανοίγει ένα νέο παράθυρο το οποίο περιέχει ένα πίνακα με όλες τις πλατφόρμες μικροελεγκτή που έχει ο χρήστης (εικόνα 4.7).

My Controllers

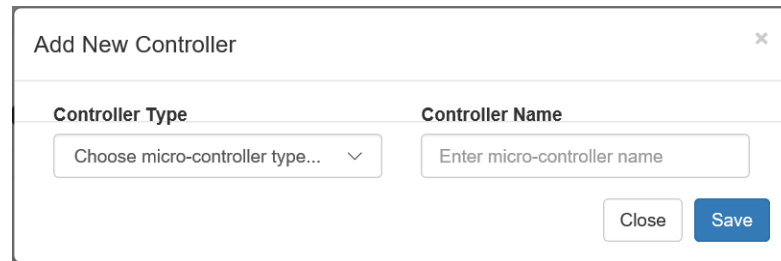
Name	Type	Number of Sensors	
myArduinoUno	Arduino	3	
newArduino	Arduino	0	
myRaspberryPi	Raspberry Pi	2	

Close

Εικόνα 4.7: Πίνακας πλατφορμών μικροελεγκτών

Σε κάθε γραμμή του δίνονται πληροφορίες που αφορούν ένα συγκεκριμένο μικροελεγκτή όπως το όνομα του, ο τύπος του και ο αριθμός των αισθητήρων που έχει τοποθετημένους πάνω του. Επίσης, ο χρήστης έχει τη δυνατότητα να διαγράψει μια πλατφόρμα μικροελεγκτή πατώντας το ανάλογο κουμπί διαγραφής. Αν η πλατφόρμα δεν έχει τοποθετημένους αισθητήρες τότε το κουμπί διαγραφής εμφανίζεται με κόκκινο χρώμα και ο χρήστης μπορεί να την διαγράψει, αλλιώς το κουμπί διαγραφής εμφανίζεται με γκριζο χρώμα και το σύστημα δεν επιτρέπει την διαγραφή της εμφανίζοντας το κατάλληλο μήνυμα.

Πατώντας το κουμπί “Add” το οποίο βρίσκεται δίπλα από τους μικροελεγκτές, ανοίγει ένα νέο παράθυρο επιτρέποντας στο χρήστη να προσθέσει μια καινούργια πλατφόρμα μικροελεγκτή (εικόνα 4.8).

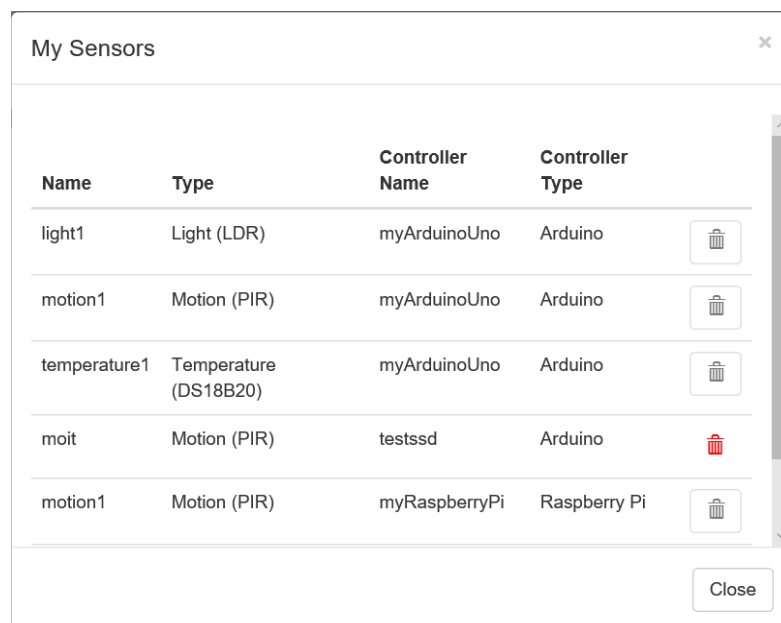


The dialog box titled "Add New Controller" contains two input fields: "Controller Type" with a dropdown menu labeled "Choose micro-controller type..." and "Controller Name" with a text input labeled "Enter micro-controller name". At the bottom right, there are "Close" and "Save" buttons.

Εικόνα 4.8: Προσθήκη νέας πλατφόρμας μικροελεγκτή

Για να πραγματοποιηθεί η συγκεκριμένη ενέργεια, ο χρήστης χρειάζεται να δώσει το τύπο της πλατφόρμας μικροελεγκτή (Arduino, Raspberry Pi, Beaglebone Black, Intel Galileo) και ένα όνομα για αυτή.

Πατώντας το κουμπί “Info” το οποίο βρίσκεται δίπλα από τους αισθητήρες ανοίγει ένα νέο παράθυρο το οποίο περιέχει ένα πίνακα με όλους τους αισθητήρες που έχει ο χρήστης (εικόνα 4.9).



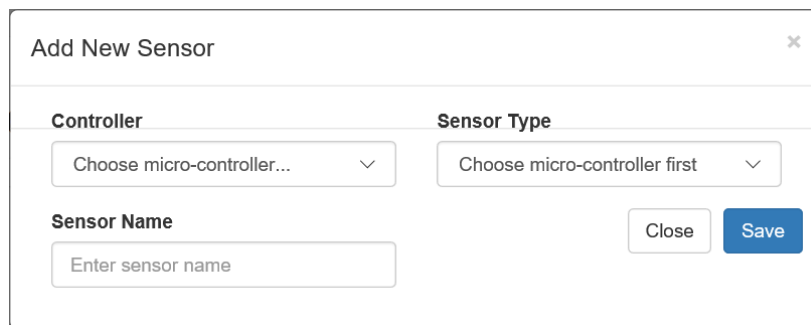
The dialog box titled "My Sensors" displays a table of sensors. The table has four columns: Name, Type, Controller Name, and Controller Type. Each row also includes a delete icon. The sensors listed are light1, motion1, temperature1, moit, and motion1. The last row is highlighted in red.

Name	Type	Controller Name	Controller Type
light1	Light (LDR)	myArduinoUno	Arduino
motion1	Motion (PIR)	myArduinoUno	Arduino
temperature1	Temperature (DS18B20)	myArduinoUno	Arduino
moit	Motion (PIR)	testssd	Arduino
motion1	Motion (PIR)	myRaspberryPi	Raspberry Pi

Εικόνα 4.9: Πίνακας αισθητήρων

Σε κάθε γραμμή του δίνονται πληροφορίες που αφορούν ένα συγκεκριμένο αισθητήρα όπως το όνομα του, ο τύπος του, το όνομα της πλατφόρμας μικροελεγκτή στην οποία είναι τοποθετημένος και ο τύπος της πλατφόρμας αυτής. Επίσης, ο χρήστης έχει τη δυνατότητα να διαγράψει ένα αισθητήρα πατώντας το ανάλογο κουμπί διαγραφής. Αν ο αισθητήρας δεν χρησιμοποιείται σε κάποιο αρχείο τότε το κουμπί διαγραφής εμφανίζεται με κόκκινο χρώμα και ο χρήστης μπορεί να τον διαγράψει, αλλιώς το κουμπί διαγραφής εμφανίζεται με γκριζο χρώμα και το σύστημα δεν επιτρέπει την διαγραφή του εμφανίζοντας το κατάλληλο μήνυμα.

Πατώντας το κουμπί “Add” το οποίο βρίσκεται δίπλα από τους αισθητήρες, ανοίγει ένα νέο παράθυρο επιτρέποντας στο χρήστη να προσθέσει ένα καινούριο αισθητήρα (εικόνα 4.10).



Εικόνα 4.10: Προσθήκη νέου αισθητήρα

Για να πραγματοποιηθεί η συγκεκριμένη ενέργεια, ο χρήστης χρειάζεται να επιλέξει την πλατφόρμα μικροελεγκτή στην οποία θα τοποθετηθεί ο αισθητήρας, να δώσει το τύπο του και ένα όνομα για αυτόν.

Πατώντας το κουμπί “Info” δίπλα από τα αρχεία κώδικα, ο χρήστης μεταβαίνει στη σελίδα “My Files” όπου φαίνονται πληροφορίες για τα αρχεία και πατώντας το κουμπί “Add”, μεταβαίνει στη μηχανή παραγωγής κώδικα.

Τέλος, έχει προστεθεί η σελίδα “Settings” (εικόνα 4.11) όπου ο χρήστης μπορεί να δει πληροφορίες για τα στοιχεία λογαριασμού του, όπως username, email και το είδος του.

Settings

Username: earist07

Email: evangelos.ar@gmail.com

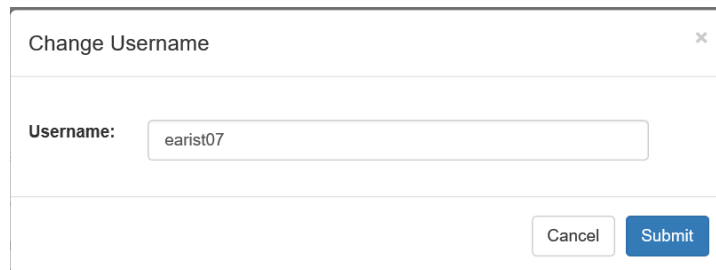
User Type: simple

[Change my username](#)

[Change my password](#)

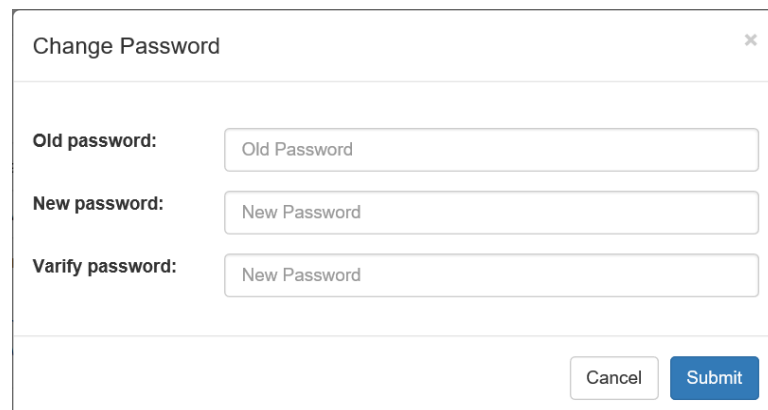
Εικόνα 4.11: Σελίδα “Settings”

Πατώντας στο σύνδεσμο “Change my username”, ανοίγει ένα νέο παράθυρο (εικόνα 4.12) όπου ο χρήστης μπορεί να αλλάξει το username του.

A dialog box titled "Change Username" with a close button (X) in the top right corner. It contains a label "Username:" followed by a text input field containing the text "earist07". At the bottom right, there are two buttons: "Cancel" and "Submit".

Εικόνα 4.12: Αλλαγή username

Πατώντας στο σύνδεσμο “Change my password”, ανοίγει ένα νέο παράθυρο (εικόνα 4.13) όπου ο χρήστης μπορεί να αλλάξει το κωδικό πρόσβασης του εισάγοντας το παλιό κωδικό του και δύο φορές τον καινούριο που θέλει να εφαρμόσει.

A dialog box titled "Change Password" with a close button (X) in the top right corner. It contains three labels with corresponding text input fields: "Old password:" with "Old Password", "New password:" with "New Password", and "Verify password:" with "New Password". At the bottom right, there are two buttons: "Cancel" and "Submit".

Εικόνα 4.13: Αλλαγή κωδικού πρόσβασης

4.4.2 Κινητή Εφαρμογή SensoMan

Η κινητή εφαρμογή του συστήματος φτιάχτηκε στα πλαίσια διπλωματική εργασίας άλλου φοιτητή. Πρόκειται για μια Android εφαρμογή που στόχο έχει να δώσει στο χρήστη τη δυνατότητα απομακρυσμένης διαχείρισης ενός αριθμού από αισθητήρες και πλατφορμών μικροελεγκτών. Για την επικοινωνία της με τη βάση δεδομένων του συστήματος χρησιμοποιεί την RESTful υπηρεσία, όπως έχει περιγραφεί πιο πάνω.

Εν συντομία, ο χρήστης της Android εφαρμογής έχει τη δυνατότητα προβολής των πλατφορμών μικροελεγκτών σε χάρτη, προβολής αισθητήρων και εκτέλεσης διαφόρων δράσεων σε αισθητήρες (ενεργοποίηση, απενεργοποίηση, αλλαγή συχνότητας). Η προβολή των αισθητήρων γίνεται με ταξινόμηση τους είτε βάσει των πλατφορμών μικροελεγκτών στις οποίες ανήκουν είτε βάσει των τύπων τους.

Τέλος, η κινητή εφαρμογή προσφέρει στο χρήστη τη δυνατότητα να δει τις τελευταίες μετρήσεις των αισθητήρων του ή ακόμα ολόκληρο το ιστορικό τους για περεταίρω επίγνωση του χώρου στον οποίο βρίσκονται τοποθετημένοι.

4.4.3 Μηχανή Κανόνων

Η μηχανή κανόνων προστέθηκε στο σύστημα SensoMan με στόχο την αυτοματοποίηση διαδικασιών, δηλαδή την εκτέλεση ενεργειών με ελάχιστη ή μηδενική παρέμβαση από το χρήστη. Η αυτοματοποίηση αυτή γίνεται με τη χρήση αισθητήρων για λήψη μετρήσεων και συσκευών δράσης (actuators) για εκτέλεση ενεργειών.

Παρέχεται η δυνατότητα στο χρήστη, μέσω γραφικής διεπαφής στην διαδικτυακή εφαρμογή, να ορίσει κανόνες τύπου IF-THEN μεταξύ αισθητήρων και συσκευών δράσης. Το πρώτο σκέλος του κανόνα (IF) προσδιορίζει μια συνθήκη και το δεύτερο σκέλος (THEN) την ενέργεια που πραγματοποιείται αν ικανοποιείται η συνθήκη.

Ένα παράδειγμα αυτού του κανόνα είναι «Αν η θερμοκρασία ξεπεράσει τους 30 βαθμούς κελσίου, τότε άναψε το κλιματιστικό». Το πρώτο σκέλος («η θερμοκρασία ξεπεράσει τους 30 βαθμούς κελσίου») αποτελεί την συνθήκη (IF) του κανόνα και το δεύτερο σκέλος («άναψε το κλιματιστικό») αποτελεί την ενέργεια που θα γίνει (THEN).

Ο κάθε κανόνας που εισάγεται από το χρήστη μεταφράζεται σε XML μορφή και αποθηκεύεται στη βάση δεδομένων του SensoMan για μελλοντική χρήση.

4.4.4 Κοινωνικό Δίκτυο

Το κοινωνικό δίκτυο υλοποιήθηκε με στόχο την παροχή κοινωνικής δικτύωσης των χρηστών με άλλους χρήστες εντός του συστήματος. Η δικτύωση αυτή επιτρέπει στους χρήστες να αλληλοεπιδρούν μεταξύ τους, προσφέροντας τους τη δυνατότητα να παρακολουθούν δεδομένα που αφορούν αισθητήρες χρηστών εντός του κοινωνικού τους δικτύου.

Ο χρήστης μέσω της διαδικτυακής εφαρμογής μπορεί να δει με ποιους άλλους χρήστες είναι συνδεδεμένος την παρούσα χρονική στιγμή, να στείλει αίτημα σύνδεσης σε νέο χρήστη και να έχει πρόσβαση στους αισθητήρες άλλων χρηστών. Τέλος, ο χρήστης μπορεί να δημιουργήσει ομάδες (groups), οι οποίες δίνουν δικαιώματα πρόσβασης σε όλους τους χρήστες εντός της ομάδας.

Κεφάλαιο 5

Μηχανή Παραγωγής Κώδικα

Μηχανή Παραγωγής Κώδικα	58
5.1 Περιγραφή	59
5.2 Διαδραστική Φόρμα Επιλογών	61
5.3 Συναρμολόγηση Αρχείου Κώδικα	65
5.3.1 Κώδικας για Arduino και Intel Galileo	69
5.3.2 Κώδικας για Raspberry Pi	74
5.3.3 Κώδικας για Beaglebone Black	77
5.4 Επιπλέον Λειτουργίες	79
5.4.1 Λίστα Υλικού	80
5.4.2 Οδηγίες συναρμολόγησης κυκλώματος – εγκατάστασης κώδικα.....	81
5.4.3 Βοηθητικά Διαγράμματα	82
5.4.4 Αποθήκευση αρχείων	84

5.1 Περιγραφή

Η δημιουργία εφαρμογών γύρο από το πλαίσιο του Δικτύου των Πραγμάτων (Internet of Things) έχει γίνει πολύ δημοφιλής δραστηριότητα στις μέρες μας. Συνεχώς, κατασκευάζονται νέες πλατφόρμες μικροελεγκτών και αισθητήρες με νέα χαρακτηριστικά και δυνατότητες που στόχο έχουν να χρησιμοποιηθούν σε διάφορους τομείς της καθημερινότητας για να κάνουν τη ζωή του ανθρώπου πιο εύκολη και πιο ευχάριστη.

Για την δημιουργία ενός κυκλώματος αισθητήρων για λήψη και ανάλυση μετρήσεων από το περιβάλλον απαιτείται μια προκαθορισμένη διαδικασία την οποία πρέπει να ακολουθήσει κάποιος. Αρχικά, απαραίτητη προϋπόθεση είναι η γνώση της πλατφόρμας μικροελεγκτή που χρησιμοποιείται και των χαρακτηριστικών της. Έπειτα ο χρήστης πρέπει να επιλέξει τους κατάλληλους αισθητήρες, οι οποίοι λύνουν το πρόβλημα του και να συναρμολογήσει το κύκλωμα αισθητήρων βάσει της ανάλογης πλατφόρμας μικροελεγκτή. Τέλος, ο χρήστης πρέπει να γράψει το πρόγραμμα το οποίο θα εγκατασταθεί στην πλατφόρμα μικροελεγκτή για να θέσει σε λειτουργία τους αισθητήρες.

Λόγω τις πληθώρας επιλογών από μικροελεγκτές και αισθητήρες που υπάρχουν σήμερα, η πιο πάνω διαδικασία είναι δύσκολη, περίπλοκη και χρονοβόρα. Έτσι στα πλαίσια της παρούσας διπλωματικής εργασίας δημιουργήθηκε ένας μηχανισμός παραγωγής κώδικα, ο οποίος έχει ως στόχο να αυτοματοποιήσει την πιο πάνω διαδικασία, προσφέροντας στο χρήστη το αρχείο κώδικα και οδηγίες που χρειάζεται για φέρει σε λειτουργία το κύκλωμα αισθητήρων που επιθυμεί.

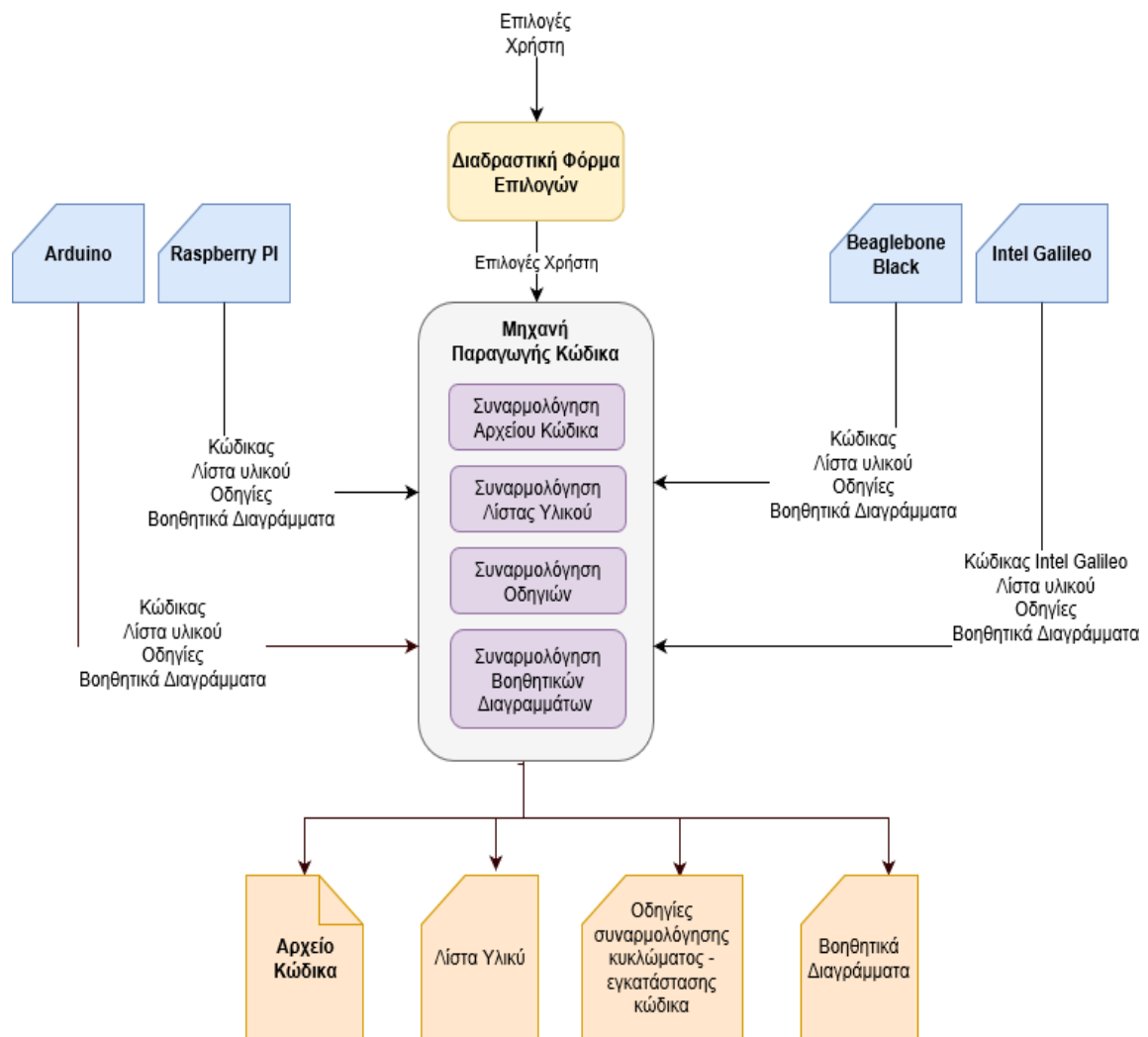
Συγκεκριμένα, η μηχανή παραγωγής κώδικα αποτελεί ξεχωριστό κομμάτι του συστήματος SensoMan, το οποίο μπορεί να τροποποιηθεί και να αναπτυχθεί ανεξάρτητα από τα υπόλοιπα. Μέσω του μηχανισμού αυτού ο χρήστης συμπληρώνει μια διαδραστική φόρμα επιλογών, εισάγοντας το είδος της πλατφόρμας μικροελεγκτή

και τους αισθητήρες που θέλει να χρησιμοποιήσει και παράγεται αυτόματα από το σύστημα το αρχείο κώδικα για το κύκλωμα αυτό.

Γενικά ο κώδικας ο οποίος παράγεται σε κάθε περίπτωση για να θέσει σε λειτουργία τους αισθητήρες χωρίζεται σε τρεις φάσεις. Στην πρώτη φάση περιλαμβάνεται η εισαγωγή οποιονδήποτε βιβλιοθηκών χρειάζονται για την λειτουργία της πλατφόρμας, ο καθορισμός των ολικών μεταβλητών και ο ορισμός βοηθητικών συναρτήσεων. Η δεύτερη φάση (setup) περιλαμβάνει οποιεσδήποτε αρχικοποιήσεις χρειάζεται να γίνουν σε ολικές μεταβλητές ή και αισθητήρες. Ορίζεται το είδος (εισόδου ή εξόδου) των συνδεδεμένων συσκευών που ενώνονται στην πλατφόρμα μικροελεγκτή, όπου στην περίπτωση των αισθητήρων είναι όλοι συσκευές εισόδου. Τέλος, η τρίτη φάση αποτελεί ένα βρόγχο (loop) ο οποίος επαναλαμβάνεται συνεχώς, όσο η πλατφόρμα μικροελεγκτή είναι συνδεδεμένη σε πηγή ενέργειας. Σε κάθε επανάληψη του βρόγχου λαμβάνονται οι μετρήσεις από τους αισθητήρες, επεξεργάζονται (αν χρειάζεται) και παρουσιάζονται στην οθόνη ή αποθηκεύονται στη βάση δεδομένων του συστήματος.

Μετά τη δημιουργία του αρχείου κώδικα από το σύστημα, αποθηκεύεται αυτόματα στα αρχεία του χρήστη ("My Files") για μελλοντική χρήση και έχει τη δυνατότητα να το κατεβάσει στο προσωπικό του υπολογιστή. Έπειτα μαζί με το αρχείο κώδικα, του δίνονται οδηγίες και βοηθητικά διαγράμματα για το πώς να συναρμολογήσει το κύκλωμα, οδηγίες για το πώς να εγκαταστήσει τον κώδικα στην πλατφόρμα μικροελεγκτή και λίστα με το υλικό που χρειάζεται για τη δημιουργία του κυκλώματος.

Η εικόνα 5.1 παρουσιάζει το διάγραμμα βημάτων που ακολουθούνται από τη μηχανή παραγωγής κώδικα.



Εικόνα 5.1: Διάγραμμα βημάτων μηχανής παραγωγής κώδικα

Στα επόμενα υποκεφάλαια θα αναλυθούν όλα τα κομμάτια της μηχανής παραγωγής κώδικα, μαζί με συγκεκριμένα παραδείγματα.

5.2 Διαδραστική Φόρμα Επιλογών

Η συμπλήρωση της διαδραστικής φόρμας επιλογών αποτελεί το πρώτο βήμα για την δημιουργία του αρχείου κώδικα. Η φόρμα αυτή συγκεντρώνει όλες τις απαραίτητες πληροφορίες και προτιμήσεις του χρήστη που χρειάζονται για την δημιουργία του

κώδικα. Χαρακτηρίζεται ως διαδραστική για το λόγο ότι αλληλοεπιδρά με το χρήστη και προσαρμόζεται ανάλογα με τις επιλογές του.

Αρχικά ο χρήστης δίνει το όνομα που θέλει να έχει το αρχείο όταν δημιουργηθεί. Έπειτα, θα πρέπει να επιλεγεί η πλατφόρμα μικροελεγκτή που θα χρησιμοποιηθεί. Στη φάση αυτή ο χρήστης έχει δύο επιλογές: είτε θα επιλέξει μια από τις ήδη υπάρχουσες πλατφόρμες μικροελεγκτών που υπάρχουν στο λογαριασμό του (εικόνα 5.2), είτε θα προσθέσει μια καινούρια (εικόνα 5.3).

Στη περίπτωση που επιλέξει μια ήδη υπάρχουσα, εμφανίζεται ο τύπος της (Arduino, Raspberry Pi, Beaglebone Black, Intel Galileo), το όνομα της και όλοι οι αισθητήρες που είναι τοποθετημένοι σε αυτή (αν υπάρχουν). Στην περίπτωση που επιλέξει να προσθέσει μια καινούργια, τότε πρέπει να δώσει το όνομα και τον τύπο της. Και στις δύο περιπτώσεις πρέπει να επιλέξει αν η συγκεκριμένη πλατφόρμα μικροελεγκτή θα είναι συνδεδεμένη στο διαδίκτυο ή όχι και κάθε πόσο θα λαμβάνονται μετρήσεις από τους αισθητήρες της (time interval). Επίσης θα πρέπει να αποθηκεύσει την νέα πλατφόρμα πατώντας το κουμπί “Save” που υπάρχει στο κάτω μέρος.

Micro-controller

Select a micro-controller that already exists.

Micro-controller arduino2 ▾	Type Arduino
Time Interval 60	Internet Connection No internet connection ▾

Time to upload measurements in database (in seconds).

Εικόνα 5.2: Επιλογή υπάρχοντος μικροελεγκτή

Micro-controller

Type	Name
Beaglebone Black	newbeaglebone
Time Interval	Internet Connection
25	Wi-Fi adapter

Time to upload measurements in database (in seconds).

Save

Εικόνα 5.3: Εισαγωγή νέου μικροελεγκτή

Οι επιλογές του χρήστη για τη σύνδεση της πλατφόρμας στο διαδίκτυο αλλάζουν ανάλογα με τον τύπο της πλατφόρμας που επιλέγεται. Για παράδειγμα, αν η πλατφόρμα μικροελεγκτή είναι τύπου Arduino υπάρχουν οι επιλογές “Adafruit CC3000 Wi-Fi shield” και “Ethernet Shield”, αν είναι τύπου Raspberry Pi υπάρχουν οι επιλογές “Wi-Fi adapter” και “Ethernet cable” και αν είναι τύπου Beaglebone Black υπάρχει η επιλογή “Ethernet cable”. Αν ο χρήστης επιλέξει μια από τις μεθόδους σύνδεσης στο διαδίκτυο τότε ο κώδικας που θα παραχθεί θα επικοινωνεί με την RESTful υπηρεσία του SensoMan για αποθήκευση των μετρήσεων στη βάση δεδομένων. Εναλλακτικά ο χρήστης μπορεί να επιλέξει “No internet connection”, όπου οι μετρήσεις θα εμφανίζονται μόνο στην οθόνη του χωρίς να αποθηκεύονται στη βάση δεδομένων. Σημειώνεται ότι κατά την επιλογή “Adafruit CC3000 Wi-Fi shield” η φόρμα επεκτείνεται ζητώντας από το χρήστη να εισάγει το όνομα και το κωδικό (SSID, password) του τοπικού δικτύου Wi-Fi, λόγω του ότι το συγκεκριμένο shield τα χρειάζεται για να κάνει τη σύνδεση στο διαδίκτυο.

Ο χρόνος λήψης μετρήσεων (time interval) εισάγεται στη φόρμα σε μορφή δευτερολέπτων (seconds). Όπως αναφέρθηκε και προηγουμένως, η πλατφόρμα μικροελεγκτή ακολουθεί ένα ατέρμονα βρόγχο, όπου σε κάθε επανάληψη του, οι αισθητήρες λαμβάνουν μέτρηση από το περιβάλλον τους. Ο χρόνος αυτός προσθέτει μια καθυστέρηση σε κάθε κύκλο του βρόχου ούτως ώστε να λαμβάνονται μετρήσεις με ένα συγκεκριμένο ρυθμό. Για παράδειγμα αν ο χρήστης συμπληρώσει τον αριθμό 60 (time interval), αυτό σημαίνει ότι κάθε λεπτό θα λαμβάνεται μια μέτρηση από κάθε αισθητήρα.

Ακολουθώντας, ο χρήστης μπορεί να προσθέσει αισθητήρες πάνω στην πλατφόρμα μικροελεγκτών που έχει επιλέξει (εικόνα 5.4). Οι αισθητήρες που υποστηρίζονται είναι έξι και είναι αυτοί που αναλύθηκαν στο κεφάλαιο 3. Κατά την επιλογή ενός καινούριου αισθητήρα ο χρήστης πρέπει να δώσει ένα όνομα για αυτόν και να επιλέξει από μια λίστα την εισδοχή που θέλει να τον τοποθετήσει στην πλατφόρμα μικροελεγκτή. Έπειτα ο χρήστης έχει την επιλογή να αποθηκεύσει τον αισθητήρα πατώντας το κουμπί “Save”, αλλιώς μπορεί να τον αφαιρέσει πατώντας το κουμπί “Cancel”. Για προσθήκη επιπλέον αισθητήρων στη φόρμα, υπάρχει το κουμπί “Add Sensor” στο κάτω μέρος.

Εικόνα 5.4: Εισαγωγή νέου αισθητήρα

Η λίστα εισδοχών αλλάζει ανάλογα με το είδος του μικροελεγκτή και το είδος του αισθητήρα (αναλογικός ή ψηφιακός) που επιλέγεται. Στον πίνακα 5.1 παρουσιάζεται η λίστα εισδοχών που εμφανίζεται σε κάθε περίπτωση, ανάλογα με τον τύπο του μικροελεγκτή και του αισθητήρα.

Τύπος πλατφόρμας μικροελεγκτή	Τύπος αισθητήρα	Λίστα εισδοχών
Arduino Uno	Ψηφιακός	0-13
	Αναλογικός	A0-A5
Raspberry Pi 2 Model B	Ψηφιακός	GPIO2 – GPIO27
	Αναλογικός	GPIO2 – GPIO27
Beaglebone Black	Ψηφιακός	P8_7 – P8_12, P8_14 – P8_18, P8_26, P9_11 – P9_13, P9_15, P9_17, P9_18, P9_23 – P9_27, P9_30, P9_41

	Αναλογικός	P9_33, P9_35 – P9_40
Intel Galileo	Ψηφιακός	0-13
	Αναλογικός	A0 - A5

Πίνακας 5.1: Λίστα εισδοχών

Τέλος, ο χρήσης όταν συμπληρώσει όλη τη φόρμα επιλογών μπορεί να πατήσει το κουμπί “GENERATE” το οποίο βρίσκεται στο κάτω δεξιά μέρος της σελίδας. Πατώντας το κουμπί ελέγχεται αν η φόρμα έχει συμπληρωθεί ορθά.

Για να είναι η φόρμα συμπληρωμένη σωστά πρέπει να ικανοποιούνται οι πιο κάτω συνθήκες:

- Έχει δοθεί έγκυρο όνομα αρχείου (γράμματα και αριθμοί)
- Το όνομα του αρχείου δεν χρησιμοποιείται ξανά
- Έχει δοθεί έγκυρος χρόνος καθυστέρησης (ακέραιος αριθμός)
- Έχουν επιλεγθεί εισδοχές για όλους τους αισθητήρες
- Δεν χρησιμοποιείται η ίδια εισδοχή για δύο ή περισσότερους αισθητήρες
- Όλοι οι αισθητήρες έχουν αποθηκευτεί στη βάση δεδομένων

Σε περίπτωση που δεν ικανοποιείται μια ή περισσότερες συνθήκες, δημιουργείται και εμφανίζεται μια αναφορά, στο πάνω μέρος της σελίδας, με όλα τα λάθη που έχουν γίνει. Σε περίπτωση που όλα είναι εντάξει, συγκεντρώνονται όλες οι πληροφορίες της φόρμας επιλογών και δίνονται ως είσοδος στην μηχανή παραγωγής κώδικα για επεξεργασία.

5.3 Συναρμολόγηση Αρχείου Κώδικα

Το δεύτερο βήμα αποτελεί η συναρμολόγηση κώδικα. Η μηχανή παραγωγής κώδικα επεξεργάζεται όλες τις επιλογές του χρήστη που δόθηκαν στη διαδραστική φόρμα επιλογών και τις χρησιμοποιεί για την παραγωγή του αρχείου. Ανάλογα με το τύπο του μικροελεγκτή που επιλέχθηκε, αποφασίζεται ο τύπος του αρχείου. Για τύπο

μικροελεγκτή Arduino και Intel Galileo, ο τύπος του αρχείου είναι Arduino Sketch File με κατάληξη .ino, το οποίο διαβάζεται από το Arduino IDE και για τύπο μικροελεγκτή Raspberry Pi και Beaglebone Black ο τύπος του αρχείου είναι Python με κατάληξη .py.

Για τη παραγωγή του κώδικα, τη συναρμολόγηση οδηγιών, λίστας υλικού, οδηγιών εγκατάστασης αρχείου-συναρμολόγησης κυκλώματος και βοηθητικών διαγραμμάτων χρησιμοποιούνται τέσσερα διαφορετικά αρχεία php, όπου το κάθε ένα από αυτά αντιπροσωπεύει ένα τύπο πλατφόρμας μικροελεγκτή (arduino.php, raspberry.php, beaglebone_black.php, intel_galileo).

Σε κάθε αρχείο υπάρχουν τέσσερις διαφορετικές συναρτήσεις όπου η κάθε μια επιστρέφει ένα από τα επιθυμητά αποτελέσματα και έχει αντιπροσωπευτικό όνομα. Οι συναρτήσεις που επιστρέφουν το κώδικα ονομάζονται getArduinoCode, getRaspberryCode, getBeagleboneBlackCode και getIntelGalileoCode. Οι συναρτήσεις λαμβάνουν ως παράμετρο ένα πίνακα με τους τύπους των αισθητήρων που επιλέχθηκαν. Για τον κάθε ένα από τους αισθητήρες, παράγεται ο κώδικας του, ο οποίος περιλαμβάνει την εισαγωγή τυχών βιβλιοθηκών που χρειάζονται, την αρχικοποίηση του (καθορισμός της εισδοχής στην οποία είναι τοποθετημένος και ορισμός του ως συσκευή εισόδου) και την λήψη μετρήσεων σε κάθε κύκλο επανάληψης (loop).

Στον κώδικα που παράγεται, οι εισδοχές των αισθητήρων δεν είναι ορισμένες με το πραγματικό τους αριθμό, αλλά με τη συμβολοσειρά <pinX>, όπου X ο αύξων αριθμός του αισθητήρα (ξεκινώντας από το 0). Για παράδειγμα, αν υπάρχουν τρεις αισθητήρες, τότε ο πρώτος θα αρχικοποιηθεί με <pin0>, ο δεύτερος <pin1> και ο τρίτος με <pin2>. Το ίδιο συμβαίνει και με το χρόνο καθυστέρησης για λήψη μετρήσεων (time interval), όπου αντί να οριστεί με τον πραγματικό αριθμό του, ορίζεται με τη συμβολοσειρά <time>. Ο λόγος που γίνεται αυτό είναι για να μπορούν να αλλαχθούν από το χρήστη μετά την δημιουργία του κώδικα.

Όταν δημιουργηθεί ο κώδικας όλων των αισθητήρων, ενώνονται όλα τα κομμάτια, δημιουργώντας τον ολοκληρωμένο κώδικα, ο οποίος επιστρέφεται από την αντίστοιχη συνάρτηση. Έπειτα, ο κώδικας ενσωματώνεται σε αρχείο και ο χρήστης μεταβαίνει σε νέα σελίδα.

Η εικόνα 5.5 παρουσιάζει την νέα σελίδα και οι αριθμοί (1-4) αντιπροσωπεύουν τις ενέργειες και πληροφορίες που παρέχονται στο χρήστη.

[File](#) [Hardware](#) [Instructions](#) [Diagrams](#)

File Created!

File Name: test01.ino

 1

[Download](#)

[Edit File](#) 2

Micro-controller

Name	Type	Internet Connection	Time Interval
arduino2	Arduino	No internet connection	60 3

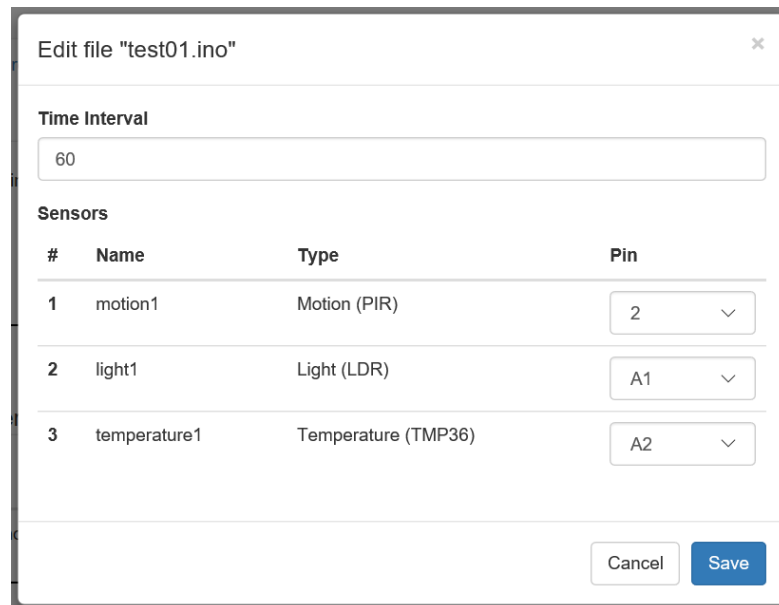
Sensors

#	Name	Type	Pin
1	motion1	Motion (PIR)	2 4
2	light1	Light (LDR)	A1
3	temperature1	Temperature (TMP36)	A2

Εικόνα 5.5: Σελίδα εμφάνισης του αρχείου

Στον αριθμό 1 παρουσιάζεται το αρχείο κώδικα που δημιουργήθηκε. Πατώντας το κουμπί αυτό οι συμβολοσειρές <pin0>, <pin1>, ... , <pinX> και <time> αντικαθίστανται με τους πραγματικούς αριθμούς των εισδοχών των αισθητήρων και χρόνου καθυστέρησης. Έπειτα το αρχείο κατεβαίνει στο προσωπικό υπολογιστή του χρήστη.

Στον αριθμό 2 φαίνεται το κουμπί “Edit”. Πατώντας το κουμπί αυτό ανοίγει ένα παράθυρο στο οποίο ο χρήστης μπορεί να αλλάξει τους αριθμούς των εισδοχών όλων των αισθητήρων, καθώς και το χρόνο καθυστέρησης για λήψη μετρήσεων (εικόνα 5.6).



#	Name	Type	Pin
1	motion1	Motion (PIR)	2
2	light1	Light (LDR)	A1
3	temperature1	Temperature (TMP36)	A2

Εικόνα 5.6: Επεξεργασία αρχείου

Στον αριθμό 3 φαίνονται όλες οι πληροφορίες που αφορούν την πλατφόρμα μικροελεγκτή. Συγκεκριμένα παρουσιάζεται το όνομα της, ο τύπος της, αν είναι ή όχι συνδεδεμένη στο διαδίκτυο και ο χρόνος λήψης μετρήσεων (time interval).

Τέλος, στον αριθμό 4 φαίνονται όλες οι πληροφορίες που αφορούν τους αισθητήρες υπό μορφή πίνακα. Κάθε γραμμή του πίνακα αντιπροσωπεύει ένα αισθητήρα και δείχνει το όνομα του, τον τύπο του και τον αριθμό εισδοχής όπου θα τοποθετηθεί στην πλατφόρμα μικροελεγκτή.

Αξίζει να σημειωθεί ότι ο κώδικας για τις πλατφόρμες μικροελεγκτή Arduino και Intel Galileo είναι ο ίδιος επειδή και οι δύο πλατφόρμες λειτουργούν με τον ίδιο τρόπο μέσω του Arduino IDE. Επίσης, παρόλο που στις δύο άλλες πλατφόρμες (Raspberry Pi και Intel Galileo) χρησιμοποιείται κώδικας pythοn, λόγω των διαφορετικών εισδοχών

που έχουν, ο κώδικας τους διαφέρει. Στα επόμενα υποκεφάλαια παρουσιάζονται παραδείγματα κώδικα για την κάθε μια από τις πλατφόρμες μικροελεγκτών.

5.3.1 Κώδικας για Arduino και Intel Galileo

Οι πλατφόρμες μικροελεγκτή Arduino και Intel Galileo, όπως αναφέρθηκε και σε προηγούμενο κεφάλαιο, χρησιμοποιούν το Arduino IDE για την δημιουργία και εγκατάσταση του κώδικα.

Σε κάθε πρόγραμμα γραμμένο στο Arduino IDE ορίζονται δύο απαραίτητες συναρτήσεις. Η πρώτη ονομάζεται `setup`, η οποία καλείται μόνο μια φορά και χρησιμοποιείται για τον ορισμό οποιονδήποτε αρχικοποιήσεων όπως μεταβλητών και αισθητήρων. Η δεύτερη συνάρτηση ονομάζεται `loop` και αποτελεί ένα ατέρμονα βρόχο ο οποίος σε κάθε επανάληψη του θέτει σε λειτουργία τους αισθητήρες για λήψη μετρήσεων.

Επίσης, για τη λειτουργία των αισθητήρων χρησιμοποιούνται τρεις βασικές συναρτήσεις:

- **`pinMode(<pinNumber>, <pinMode>)`:** Η συνάρτηση αυτή καλείται μέσα στο σώμα την συνάρτησης `setup` και για κάθε συσκευή που είναι τοποθετημένη στην πλατφόρμα μικροελεγκτή. Ορίζει αν η συγκεκριμένη συσκευή είναι συσκευή εισόδου (INPUT) ή συσκευή εξόδου (OUTPUT). Παίρνει δύο παραμέτρους, η πρώτη (`pinNumber`) είναι ο αριθμός της εισδοχής όπου είναι συνδεδεμένη η συσκευή και η δεύτερη (`pinMode`) λαμβάνει την τιμή INPUT αν η συγκεκριμένη συσκευή είναι εισόδου (π.χ. αισθητήρας) ή την τιμή OUTPUT αν είναι συσκευή εξόδου (π.χ. συσκευή δράσης – actuator).
- **`digitalRead(<pinNumber>)`:** Η συνάρτηση αυτή καλείται μέσα στο σώμα της συνάρτησης `loop` για κάθε ψηφιακή συσκευή εισόδου και λαμβάνει ως παράμετρο την εισδοχή της συγκεκριμένης συσκευής. Σκοπός της είναι να διαβάσει την ψηφιακή τιμή της συσκευής.

- **analogRead(<pinNumber>):** Η συνάρτηση αυτή καλείται μέσα στο σώμα της συνάρτησης loop για κάθε αναλογική συσκευή εισόδου και λαμβάνει ως παράμετρο την εισδοχή της συγκεκριμένης συσκευής. Σκοπός της είναι να διαβάσει την αναλογική τιμή της συσκευής.

Ακολουθούν παραδείγματα κώδικα αισθητήρων στα οποία φαίνεται η χρήση των πιο πάνω συναρτήσεων. Σημειώνεται ότι στα παραδείγματα αυτά δεν επιλέγεται κάποιος τρόπος σύνδεσης της πλατφόρμας μικροελεγκτή με το διαδίκτυο.

```
int motion_pin1 = 3;
int motion_value1;

void setup() {
  Serial.begin(9600);
  pinMode(motion_pin1, INPUT);
}

void loop() {
  motion_value1 = digitalRead(motion_pin1);
  Serial.print('Motion = ');
  Serial.println(motion_value1);

  delay(50000);
}
```

Εικόνα 5.7: Κώδικας αισθητήρα Motion PIR σε Arduino IDE

Στην εικόνα 5.7 παρουσιάζεται ο κώδικας για ένα αισθητήρα κίνησης Motion PIR. Αρχικά ορίζεται ότι ο αισθητήρας είναι τοποθετημένος στην εισδοχή με αριθμό 3 (int motion_pin1 = 3) και έπειτα στη συνάρτηση setup καθορίζεται ότι είναι τύπου εισόδου (pinMode(motion_pin1, INPUT)). Η κλήση της συνάρτησης Serial.begin(9600) αρχικοποιεί την σειριακή οθόνη εκτύπωσης (serial monitor) του Arduino IDE στα 9600 bits per seconds. Η οθόνη αυτή χρησιμοποιείται για την εκτύπωση μηνυμάτων από την πλατφόρμα μικροελεγκτή στον ηλεκτρονικό υπολογιστή. Τέλος, στη συνάρτηση loop, μέσω της συνάρτησης digitalRead διαβάζεται η μέτρηση του αισθητήρα κίνησης (0 (low) ή 1 (high)) και εκτυπώνεται στην οθόνη. Η κλήση της συνάρτησης delay λαμβάνει ως παράμετρο το χρόνο καθυστέρησης υπό τη μορφή milliseconds (ms).

Ο κώδικας για τον αισθητήρα Sound (digital) ακολουθεί την ίδια λογική.

```

int light_pin1 = A0;
int light_value1;

void setup() {
  Serial.begin(9600);
  pinMode(light_pin1, INPUT);
}

void loop() {
  light_value1 = analogRead(light_pin1);
  Serial.print('Brightness = ');
  Serial.println(light_value1);

  delay(50000);
}

```

Εικόνα 5.8: Κώδικας αισθητήρα LDR σε Arduino IDE

Στην εικόνα 5.8 παρουσιάζεται ο κώδικας για ένα αισθητήρα φωτεινότητας LDR σε πλατφόρμα μικροελεγκτή Arduino ή Intel Galileo. Ο αισθητήρας αυτός είναι αναλογικού τύπου και ο κώδικας του είναι παρόμοιος με το κώδικα ψηφιακού αισθητήρα όπου αναφέρθηκε πιο πάνω. Η μοναδική διαφορά που έχει είναι ότι χρησιμοποιείται η συνάρτηση `analogRead` (αντί η `digitalRead`), για να διαβάσει το αναλογικό σήμα του αισθητήρα. Σημειώνεται ότι το αναλογικό σήμα παίρνει τιμές από 0 μέχρι 1023.

Ο κώδικας για τους υπόλοιπους αναλογικούς αισθητήρες (Humidity FC-28, Temperature TMP36, Sound analog) είναι παρόμοιος.

```

#include <OneWire.h>
#include <DallasTemperature.h>

#define ONE_WIRE_BUS 2
int temperature_value1;
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);

void setup() {
  Serial.begin(9600);
  sensors.begin();
}

void loop() {
  Serial.print('Measuring temperature...');
  sensors.requestTemperatures();
  temperature_value1 = sensors.getTempCByIndex(0);
  Serial.print('Temperature = ');
  Serial.print(temperature_value1);
  Serial.println('degrees celsius');

  delay(50000);
}

```

Εικόνα 5.9: Κώδικας αισθητήρα Temperature DS18B20 σε Arduino IDE

Στην εικόνα 5.9 παρουσιάζεται ο κώδικας για τον αισθητήρα θερμοκρασίας DS18B20. Ο αισθητήρας αυτός, αν και είναι ψηφιακού σήματος, διαφέρει ο κώδικας του από τους άλλους ψηφιακούς αισθητήρες που αναφέρθηκαν πιο πάνω και χρειάζεται τις βιβλιοθήκες OneWire.h και DallasTemperature.h. Αρχικά, ορίζεται ένα αντικείμενο τύπου OneWire, το οποίο λαμβάνει ως παράμετρο τον αριθμό εισδοχής του αισθητήρα και χρησιμοποιείται για την επικοινωνία της πλατφόρμας μικροελεγκτή με το συγκεκριμένο αισθητήρα. Ακολούθως, δημιουργείται ένα αντικείμενο DallasTemperature (με παράμετρο το αντικείμενο OneWire) που προσφέρει συναρτήσεις για ενεργοποίηση του αισθητήρα και λήψη μετρήσεων. Ο αισθητήρας ενεργοποιείται μέσω της μεθόδου begin μέσα στο σώμα της συνάρτησης setup και λαμβάνει μέτρηση θερμοκρασίας σε βαθμούς κελσίου μέσω της getTempCByIndex(0) σε κάθε κύκλο επανάληψης της συνάρτησης loop.

Σε περίπτωση σύνδεσης της πλατφόρμας μικροελεγκτή Arduino με το διαδίκτυο, ο κώδικας αλλάζει ανάλογα με τη μέθοδο που επιλέχθηκε για το σκοπό αυτό. Όπως αναφέρθηκε και στο προηγούμενο υποκεφάλαιο, η μηχανή παραγωγής κώδικα υποστηρίζει δύο τρόπους σύνδεσης της πλατφόρμας μικροελεγκτή Arduino στο διαδίκτυο. Ο πρώτος είναι μέσω της ασπίδας “Adafruit CC3000 Wi-Fi shield” και ο δεύτερος είναι μέσω τις ασπίδας “Ethernet shield”. Ο κώδικας ο οποίος παράγεται για κάθε μια από τις ασπίδες αυτές διαφέρει στις βιβλιοθήκες που χρησιμοποιούνται και στο τρόπο σύνδεσης της πλατφόρμας μικροελεγκτή στο διαδίκτυο.

```
#include <Adafruit_CC3000.h>
#include <ccspi.h>
#include <SPI.h>
#include <string.h>
#include "utility/debug.h"

// These are the interrupt and control pins
#define ADAFRUIT_CC3000_IRQ 3 // MUST be an interrupt pin!
// These can be any two pins
#define ADAFRUIT_CC3000_VBAT 5
#define ADAFRUIT_CC3000_CS 10
// Use hardware SPI for the remaining pins
// On an UNO, SCK = 13, MISO = 12, and MOSI = 11
Adafruit_CC3000 cc3000 = Adafruit_CC3000(ADAFRUIT_CC3000_CS, ADAFRUIT_CC3000_IRQ, ADAFRUIT_CC3000_VBAT, SPI_CLOCK_DIVIDER);

#define WLAN_SSID "myWiFiName" // cannot be longer than 32 characters!
#define WLAN_PASS "myWiFiPass"
// Security can be WLAN_SEC_UNSEC, WLAN_SEC_WEP, WLAN_SEC_WPA or WLAN_SEC_WPA2
#define WLAN_SECURITY WLAN_SEC_WPA2

#define WEBSITE "http://thesis.in.cs.ucy.ac.cy/sensoman2"
uint32_t ip;
Adafruit_CC3000_Client www;
int motion_pin1 = 3;
int motion_value1;
```

Εικόνα 5.10: Κώδικας Arduino με χρήση ασπίδας Adafruit CC3000 Wi-Fi shield

Στην εικόνα 5.10 φαίνεται το πρώτο κομμάτι κώδικα για την ασπίδα “Adafruit CC3000 Wi-Fi shield”. Αρχικά στον αριθμό 1 φαίνονται όλες οι βιβλιοθήκες που χρειάζονται για τη λειτουργία της συγκεκριμένης ασπίδας. Στον αριθμό 2 ορίζονται τρεις διαφορετικές σταθερές, όπου η κάθε μια αντιπροσωπεύει μια εισδοχή της πλατφόρμας όπου τοποθετείται η ασπίδα αυτή. Έπειτα ορίζεται ένα αντικείμενο τύπου Adafruit_CC3000 το οποίο αντιπροσωπεύει ένα κανάλι επικοινωνίας μεταξύ της πλατφόρμας μικροελεγκτή και της ασπίδας. Στον αριθμό 3 ορίζονται τα στοιχεία του δικτύου Wi-Fi του χρήστη, όπως το όνομα του δικτύου (WLAN_SSID) και ο κωδικός σύνδεσης (WLAN_PASS). Σημειώνεται ότι τα στοιχεία αυτά λήφθηκαν μέσω της διαδραστικής φόρμας επιλογών από το χρήστη. Τέλος, στον αριθμό 4 ορίζεται το URL της RESTful υπηρεσίας διαδικτύου του συστήματος SensoMan με σκοπό την αποστολή των μετρήσεων στη βάση δεδομένων.

```
void sendMeasurement(String id, String value){
    String data= "sensorID=" + id + "&measurement=" + value;
    www = cc3000.connectTCP(ip, 80);
    if (www.connected()) {
        www.println("POST /upload/post_measurement.php HTTP/1.1");
        www.print("Host: ");
        www.println(WEBSITE);
        www.println("Content-Type: application/x-www-form-urlencoded");
        www.println("Connection: close");
        www.println("User-Agent: Arduino/1.0");
        www.print("Content-Length: ");
        www.println(value.length());
        www.println();
        www.print(data);
        www.println();
    }
    else {
        Serial.println(F("Connection failed"));
        return;
    }
    www.close();
}
```

Εικόνα 5.11: Συνάρτηση αποστολής μηνύματος HTTP POST

Στην εικόνα 5.11 φαίνεται η συνάρτηση sendMeasurement. Η συνάρτηση αυτή δημιουργεί TCP σύνδεση μεταξύ της πλατφόρμας μικροελεγκτή και της RESTful υπηρεσίας διαδικτύου του συστήματος SensoMan και αποστέλλει μήνυμα HTTP post με το id του αισθητήρα και τη μέτρηση του, τα οποία λαμβάνει ως παράμετρο. Κάθε φορά που ένας αισθητήρας λαμβάνει μέτρηση από το περιβάλλον του, καλείται η συνάρτηση αυτή για να αποθηκευτεί η μέτρηση του στη βάση δεδομένων.

```

#include <SPI.h>
#include <Ethernet.h>

// Enter a MAC address for your controller below.
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
char server[] = "http://thesis.in.cs.ucy.ac.cy/sensoman2";

// Enter the static IP address to use if the DHCP fails to assign
IPAddress ip(192, 168, 0, 177);

EthernetClient client;
int light_pin1 = A1;
int light_value1;

void sendMeasurement(String id, String value){
  String data= "sensorID=" + id + "&measurement=" + value;
  if (client.connect(server, 80)){
    client.println("POST /upload/post_measurement.php HTTP/1.1");
    client.print("Host: ");
    client.println(server);
    client.println("Content-Type: application/x-www-form-urlencoded");
    client.println("Connection: close");
    client.println("User-Agent: Arduino/1.0");
    client.print("Content-Length: ");
    client.println(value.length());
    client.println();
    client.print(data);
    client.println();
  }
  else{
    Serial.println("Cannot connect to Server");
  }
}

```

Εικόνα 5.12: Κώδικας Arduino με χρήση ασπίδας Ethernet shield

Στην εικόνα 5.12 φαίνεται ο κώδικας που παράγεται με χρήστη της ασπίδας “Ethernet shield” σε πλατφόρμα μικροελεγκτή Arduino. Αρχικά, εισάγονται οι βιβλιοθήκες SPI.h και Ethernet.h και ορίζονται δύο γενικές μεταβλητές, η mac όπου ορίζει την mac διεύθυνση του μικροελεγκτή (κάθε μικροελεγκτής που χρησιμοποιείται πρέπει να έχει διαφορετική διεύθυνση mac) και η sever όπου ορίζει την URL διεύθυνση της RESTful υπηρεσίας διαδικτύου του συστήματος SensoMan. Ακολούθως, δημιουργείται αντικείμενο τύπου EthernetClient για την επικοινωνία της ασπίδας με την πλατφόρμα μικροελεγκτή. Τέλος, ορίζεται η συνάρτηση sendMeasurement που είναι παρόμοια με τη συνάρτηση που ορίστηκε στην ασπίδα “Adafruit CC3000 Wi-Fi shield”.

5.3.2 Κώδικας για Raspberry Pi

Η πλατφόρμα μικροελεγκτή Raspberry Pi τρέχει κώδικα γραμμένο σε Python. Λόγω του μειονεκτήματος της πλατφόρμας αυτής, δηλαδή ότι δεν έχει εισδοχές εισόδου αναλογικού σήματος, οι αναλογικοί αισθητήρες δεν υποστηρίζονται. Παρ’ όλα αυτά

για τον αισθητήρα LDR χρησιμοποιείται ένας διαφορετικός τρόπος για διάβασμα της φωτεινότητας, ο οποίος επεξηγείται στη συνέχεια.

Σε όλα τα προγράμματα γραμμένα σε python για πλατφόρμα μικροελεγκτή Raspberry Pi, υπάρχουν κάποια βασικά στοιχεία τα οποία είναι κοινά. Γενικά, χωρίζονται σε τρία μέρη. Το πρώτο μέρος αποτελεί την εισαγωγή βιβλιοθηκών και ορισμό γενικών μεταβλητών, το δεύτερο περιέχει οποιεσδήποτε αρχικοποιήσεις και ορισμούς συναρτήσεων και το τρίτο περιέχει ένα ατέρμονα βρόγχο, ο οποίος επαναλαμβάνεται συνεχώς για λήψη μετρήσεων από τους αισθητήρες.

Αρχικά, εισάγονται δύο βασικές βιβλιοθήκες, η time και η RPi. Η βιβλιοθήκη time προσφέρει τη μέθοδο time.sleep η οποία χρησιμοποιείται με τον ίδιο τρόπο με την delay στο Arduino IDE, για καθυστέρηση του χρόνου. Από τη βιβλιοθήκη RPi χρησιμοποιείται η κλάση GPIO, η οποία προσφέρει μεθόδους για λειτουργίες που αφορούν τις εισδοχές της πλατφόρμας Raspberry Pi. Οι πιο χρησιμοποιημένες μέθοδοι είναι οι εξής:

- **GPIO.setup(<pinNumber>, <pinMode>):** Η συνάρτηση αυτή ορίζει αν η συγκεκριμένη συσκευή, η οποία είναι τοποθετημένη στην εισδοχή που δίνεται ως παράμετρος (pinNumber), είναι εισόδου GPIO.IN ή εξόδου GPIO.OUT.
- **GPIO.setMode(<mode>):** Η συνάρτηση αυτή ορίζει το είδος (mode) των εισδοχών που θα χρησιμοποιηθούν στο πρόγραμμα. Αν δοθεί σαν παράμετρος GPIO.BOARD τότε χρησιμοποιούνται οι φυσικοί αριθμοί των εισδοχών που είναι τυπωμένοι στην πλατφόρμα μικροελεγκτή. Αντίθετα, αν δοθεί ως παράμετρος GPIO.BCM τότε χρησιμοποιούνται οι αριθμοί που αναγράφονται στο διάγραμμα εισδοχών της πλατφόρμας, μετά τη λέξη GPIO.
- **GPIO.input(<pinNumber>):** Η συνάρτηση αυτή διαβάζει το ψηφιακό σήμα της συσκευής εισόδου που είναι τοποθετημένη στην εισδοχή που δίνεται ως παράμετρος.

```

import time
import RPi.GPIO as GPIO

motion_pin1 = 4;

GPIO.setup(motion_pin1, GPIO.IN)
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)

while True:
    motion_value = GPIO.input(motion_pin1)
    print ('Motion = ',motion_value,'\n')

    time.sleep(60)

```

Εικόνα 5.13: Κώδικας αισθητήρα Motion PIR σε Raspberry Pi

Στην εικόνα 5.13 παρουσιάζεται ο κώδικας για τον αισθητήρα Motion PIR γραμμένος στη γλώσσα προγραμματισμού python για την πλατφόρμα μικροελεγκτή Raspberry Pi. Η εισδοχή του αισθητήρα ορίζεται στον αριθμό 4 (άρα GPIO4 αφού η μέθοδος setMode λαμβάνει ως παράμετρο GPIO.BCM) και ως συσκευή εισόδου. Επίσης, σε κάθε κύκλο επανάληψης λαμβάνεται η μέτρηση από τον αισθητήρα και εκτυπώνεται στην οθόνη. Παρόμοιος κώδικας παράγεται και στην περίπτωση των αισθητήρων Humidity FC-28 και Sound digital.

Για τη χρήση του αισθητήρα LDR σε Raspberry Pi χρησιμοποιείται μια βοηθητική συνάρτηση η οποία προσομοιώνει την μέτρηση της φωτεινότητας ενός χώρου. Για το σκοπό αυτό χρειάζεται ένας επιπλέον πυκνωτής 1uF, οποίος συνδέεται στο κύκλωμα μαζί με τον αισθητήρα LDR. Η συνάρτηση αυτή μετράει στην ουσία το χρόνο που χρειάζεται ο πυκνωτής να γεμίσει μέσω της τάσης του LDR αισθητήρα και έτσι προσημειώνεται η φωτεινότητα του χώρου. Στην εικόνα 5.14 φαίνεται ο κώδικας αυτής της μεθόδου.

```

import time
import RPi.GPIO as GPIO

light_pin1 = 5;

GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)

def getLightValue (pin):
    count = 0
    GPIO.setup(pin, GPIO.OUT)
    GPIO.output(pin, GPIO.LOW)
    time.sleep(0.1)
    GPIO.setup(pin_to_circuit, GPIO.IN)
    while (GPIO.input(pin_to_circuit) == GPIO.LOW):
        count += 1
    return count;

while True:
    light_value = getLightValue(light_pin1)
    print ('Light = ',light_value,'\n')

    time.sleep(60)

```

Εικόνα 5.14: Κώδικας αισθητήρα LDR σε Raspberry Pi

Για λήψη μέτρησης από τον αισθητήρα θερμοκρασίας DS18B20 χρησιμοποιείται η βιβλιοθήκη W1ThermSensor [27]. Στην εικόνα 5.15 παρουσιάζεται ο κώδικας για το συγκεκριμένο αισθητήρα.

```

import time
from w1thermsensor import W1ThermSensor
#Connect the temperature sensor to pin GPIO_4;

sensor = W1ThermSensor()

while True:
    temperature_value = sensor.get_temperature()
    print ('Temperature = ',temperature_value,' degrees celsius','\\n')

    time.sleep(60)

```

Εικόνα 5.15: Κώδικας αισθητήρα DS18B20 σε Raspberry Pi

5.3.3 Κώδικας για Beaglebone Black

Η πλατφόρμα μικροελεγκτή Beaglebone Black τρέχει κώδικα γραμμένο σε python με παρόμοιο τρόπο όπως το Raspberry Pi. Η διαφορά της πλατφόρμας Beaglebone Black

σε σύγκριση με την Raspberry Pi είναι ότι έχει εισδοχές εισόδου αναλογικού σήματος και έτσι υποστηρίζονται όλοι οι αισθητήρες που αναφέρθηκαν στο κεφάλαιο 3.

Για τις ενέργειες που αφορούν τις εισδοχές της πλατφόρμας χρησιμοποιείται η βιβλιοθήκη Adafruit_BBIO και συγκεκριμένα οι κλάσεις GPIO για τις εισδοχές ψηφιακού σήματος και ADC για τις εισδοχές αναλογικού σήματος. Για ανάγνωση των μετρήσεων από τους αισθητήρες χρησιμοποιούνται οι συναρτήσεις GPIO.input και ADC.read αντίστοιχα.

Στην εικόνα 5.16 παρουσιάζεται ο κώδικας γραμμένος σε python για τον αισθητήρα θερμοκρασίας TMP36. Σημειώνεται ότι ο κώδικας είναι παρόμοιος για όλους τους αναλογικούς αισθητήρες (Humidity FC-28, Sound analog, Light LDR).

```
import time
import Adafruit_BBIO.ADC as ADC

temperature_pin1 = P9_33;

ADC.setup()

while True:
    temperature_value = ADC.read(temperature_pin1)
    temperature_value = ((temperature_value * 1800)-500)/10;
    print ('Temperature = ',temperature_value, ' degrees celsius', '\n')

    time.sleep(60)
    GPIO.cleanup()
```

Εικόνα 5.16: Κώδικας αισθητήρα θερμοκρασίας TMP36 σε Beaglebone Black

Στην εικόνα 5.17 παρουσιάζεται ο κώδικας για τον αισθητήρα κίνησης PIR. Ο κώδικας για τον αισθητήρα ήχου με ψηφιακή έξοδο είναι παρόμοιος.

```

import time
import Adafruit_BBIO.GPIO as GPIO

motion_pin1 = P8_12;

GPIO.setup(motion_pin1, GPIO.IN)

while True:
    motion_value = GPIO.input(motion_pin1)
    print ('Motion = ',motion_value,'\n')

    time.sleep(60)
    GPIO.cleanup()

```

Εικόνα 5.17: Κώδικας αισθητήρα κίνησης PIR σε Beaglebone Black

Όταν ο χρήστης επιλέξει την σύνδεση της πλατφόρμας μικροελεγκτή Beaglebone Black ή Raspberry Pi με το διαδίκτυο, στον κώδικα περιέχεται μια επιπλέον βοηθητική συνάρτηση (postMeasurement) η οποία χρησιμοποιείται για την αποστολή των μετρήσεων στη βάση δεδομένων του συστήματος SensoMan. Η συνάρτηση αυτή καλεί τη RESTful υπηρεσία διαδικτύου και συγκεκριμένα τη λειτουργία postMeasurement στέλνοντας τις μετρήσεις σε μορφή JSON. Στην εικόνα 5.18 παρουσιάζεται ο κώδικας της συγκεκριμένης συνάρτησης.

```

def postMeasurement(sensorID, value):
    url = 'http://thesis.in.cs.ucy.ac.cy/sensoman2/upload/post_measurement.php'
    data = {'sensorID': str(sensorID), 'measurement': str(value)}
    req = requests.post(url, json = data)
    if (req.status_code==200):
        print('Measurement uploaded successfully!\n')
    else:
        print('Unable to upload the measurement.\n')

```

Εικόνα 5.18: Συνάρτηση postMeasurement

5.4 Επιπλέον Λειτουργίες

Η μηχανή παραγωγής κώδικα εκτός από την κύρια λειτουργία της, παραγωγή αρχείου κώδικα, κάνει κάποιες επιπλέον λειτουργίες οι οποίες προσφέρουν περισσότερες πληροφορίες και δυνατότητες στο χρήστη για το κύκλωμα των αισθητήρων που έχει επιλέξει. Οι επιπλέον λειτουργίες που γίνονται είναι η συναρμολόγηση λίστας υλικού που χρειάζεται, συναρμολόγηση οδηγιών συναρμολόγησης του κυκλώματος των

αισθητήρων και εγκατάστασης κώδικα στην πλατφόρμα μικροελεγκτή, εμφάνιση βοηθητικών διαγραμμάτων και αποθήκευση του αρχείου στη βάση δεδομένων για μελλοντική χρήση.

Στόχος αυτών των επιπλέον πληροφοριών είναι να βοηθήσουν το χρήστη του συστήματος να συναρμολογήσει και να θέσει σε λειτουργία το κύκλωμα των αισθητήρων του γρήγορα και εύκολα. Στα υποκεφάλαια που ακολουθούν παρουσιάζονται οι πιο πάνω λειτουργίες.

5.4.1 Λίστα Υλικού

Μετά την συναρμολόγηση του αρχείου κώδικα, το σύστημα συναρμολογεί μια λίστα με όλο το υλικό (hardware) το οποίο χρειάζεται για το κύκλωμα των αισθητήρων που επιλέχθηκε. Η λίστα αυτή δημιουργείται δυναμικά, υποστηρίζοντας όλους τους συνδυασμούς πλατφορμών μικροελεγκτών με αισθητήρες.

Η λίστα εμφανίζεται πατώντας την καρτέλα περιήγησης (tab) “Hardware” που υπάρχει στο πάνω μέρος της σελίδας όπου μεταβαίνει ο χρήστης μετά τη συμπλήρωση της φόρμας επιλογών. Η λίστα εμφανίζεται στην οθόνη του χρήστη μέσω εικόνων και περιγραφών των απαραίτητων εξαρτημάτων.

Στην εικόνα 5.19 παρουσιάζεται ένα παράδειγμα λίστας υλικού. Το κύκλωμα στο οποίο αναφέρεται περιέχει την πλατφόρμα μικροελεγκτή Beaglebone Black και δύο αισθητήρες φωτεινότητας LDR. Επίσης, έχει επιλεγθεί σύνδεση της πλατφόρμας στο διαδίκτυο μέσω καλωδίου ethernet. Όπως φαίνεται και στην εικόνα, παρουσιάζεται όλο το απαραίτητο υλικό για αυτό το κύκλωμα αισθητήρων συμπεριλαμβανομένου και δύο επιπλέον αντιστάσεων που χρειάζονται για την λειτουργία των αισθητήρων LDR.

Hardware

-  1 BeagleBone Black board + Power supply
-  1 Ethernet Cable
-  1 Breadboard
-  Jumper wires
-  2 LDR sensor(s)
-  2 Resistor(s) 10k ohm.

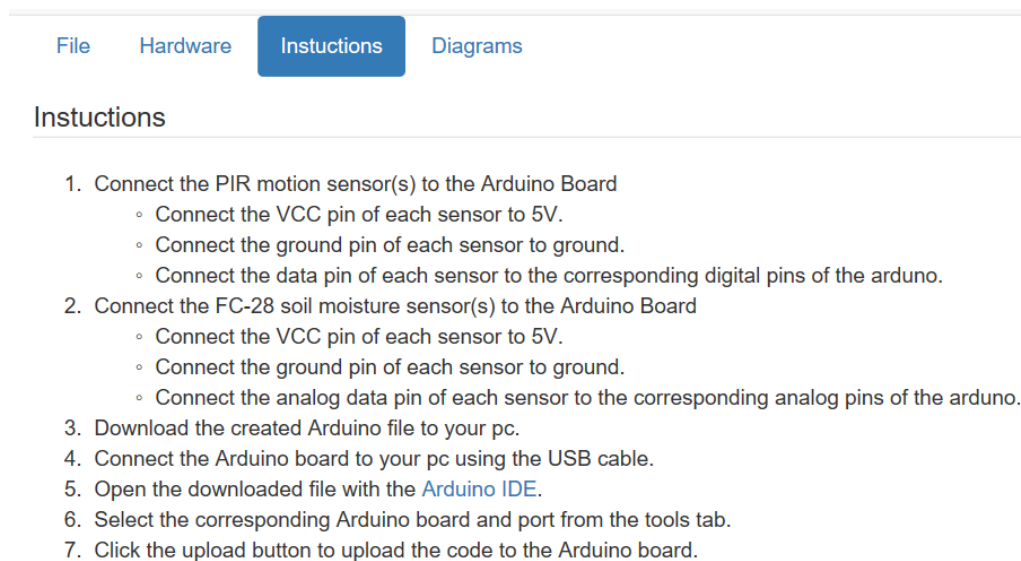
Εικόνα 5.19: Παράδειγμα Λίστας Υλικού

5.4.2 Οδηγίες συναρμολόγησης κυκλώματος – εγκατάστασης κώδικα

Μια ακόμα σημαντική και απαραίτητη λειτουργία της μηχανής παραγωγής κώδικα είναι η παροχή οδηγιών συναρμολόγησης του κυκλώματος των αισθητήρων και εγκατάστασης του κώδικα στην πλατφόρμα μικροελεγκτή.

Οι οδηγίες αυτές, όπως και η λίστα υλικού, δημιουργούνται δυναμικά ανάλογα με το κύκλωμα αισθητήρων που επιλέγεται. Για την εμφάνιση των οδηγιών, ο χρήστης πατάει την καρτέλα περιήγησης “Instructions”, στο πάνω μέρος της σελίδας.

Επεξηγούν την τοποθέτηση του καθ’ ενός αισθητήρα πάνω στην αντίστοιχη πλατφόρμα μικροελεγκτή καθώς και επιπλέον υλικού αν χρειάζεται. Η επεξήγηση είναι απλή και περιγραφική για να γίνει εφικτή η συναρμολόγηση του κυκλώματος ακόμα και από αρχάριους χρήστες. Ακόμα, περιγράφεται ο τρόπος με τον οποίο ο χρήστης θα εγκαταστήσει τον κώδικα στην πλατφόρμα μικροελεγκτή και να την θέσει σε λειτουργία. Στην εικόνα 5.20 φαίνεται ένα παράδειγμα οδηγιών για ένα κύκλωμα το οποίο αφορά την πλατφόρμα μικροελεγκτή Arduino, ενός αισθητήρα κίνησης και ενός αισθητήρα υγρασίας εδάφους.

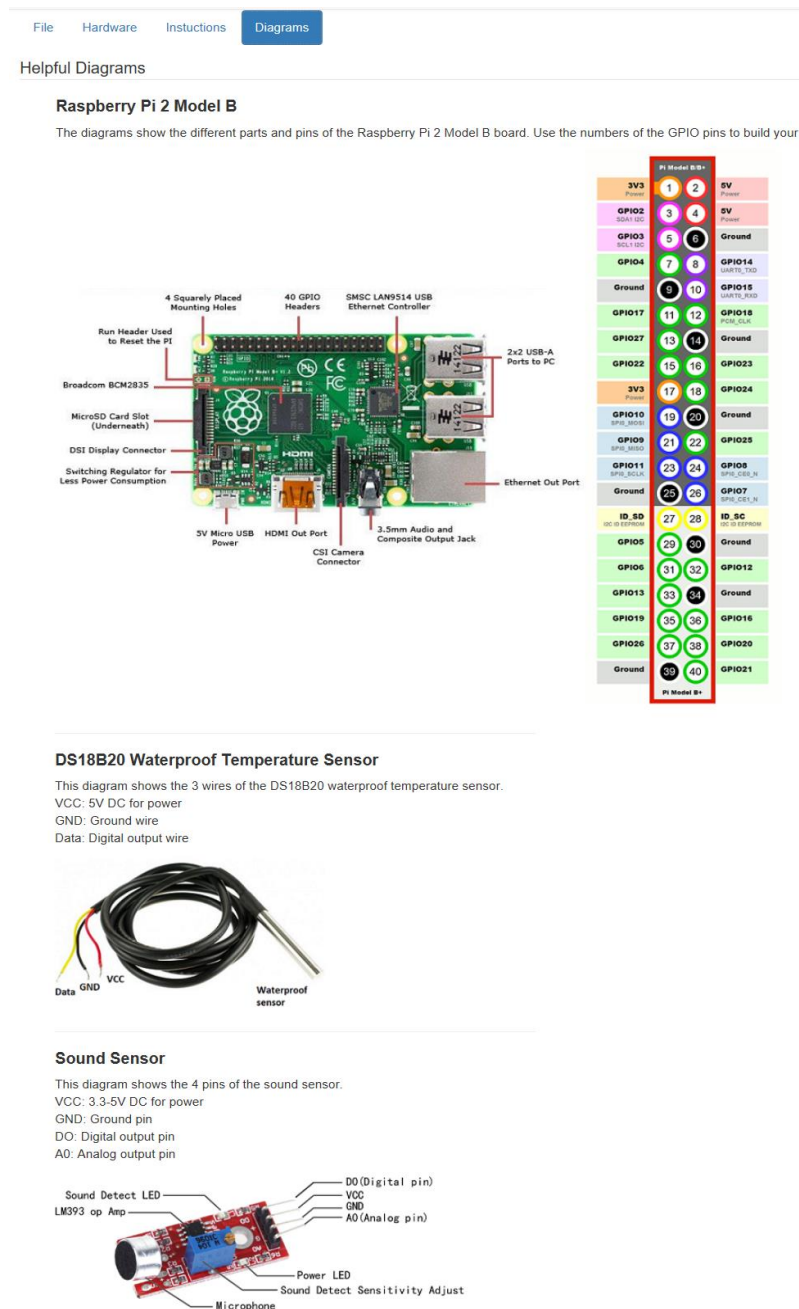


Εικόνα 5.20: Παράδειγμα οδηγιών κυκλώματος αισθητήρων σε Arduino

5.4.3 Βοηθητικά Διαγράμματα

Σε συνδυασμό με τις οδηγίες που αναφέρθηκαν στο προηγούμενο υποκεφάλαιο, προσφέρονται από το σύστημα διαγράμματα τα οποία δίνουν περισσότερες πληροφορίες για την πλατφόρμα μικροελεγκτή και τους αισθητήρες που επιλέχθηκαν.

Συγκεκριμένα, παρουσιάζουν με γραφικό τρόπο διάφορα μέρη της πλατφόρμας μικροελεγκτή, όπως θύρες, εξαρτήματα και εισδοχές (pins). Επίσης δίνουν πληροφορίες για τους αισθητήρες, οι οποίες βοηθούν άμεσα την τοποθέτηση τους στην πλατφόρμα. Στην εικόνα 5.21 παρουσιάζονται τα βοηθητικά διαγράμματα για κύκλωμα, το οποίο περιλαμβάνει τη πλατφόρμα μικροελεγκτή Raspberry Pi, ένα αισθητήρα θερμοκρασίας DS18B20 και ένα αισθητήρα ήχου.



Εικόνα 5.21: Παράδειγμα βοηθητικών διαγραμμάτων

5.4.4 Αποθήκευση αρχείων

Το αρχείο κώδικα αφού δημιουργηθεί από το σύστημα, αποθηκεύεται αυτόματα στη βάση δεδομένων για μελλοντική χρήση. Ο χρήστης μπορεί να δει όλα τα αρχεία που έχει δημιουργήσει μεταβαίνοντας στη σελίδα “My Files”. Τα αρχεία εμφανίζονται υπό τη μορφή πίνακα, όπου η κάθε γραμμή αντιπροσωπεύει ένα συγκεκριμένο αρχείο, δίνοντας βασικές πληροφορίες για αυτό. Φαίνεται ο τύπος του αρχείου, το όνομα και ο τύπος της πλατφόρμας μικροελεγκτή που επιλέχθηκε, καθώς και πόσοι αισθητήρες αφορούν το συγκεκριμένο κύκλωμα.

My Files					
	File Name	Controller Name	Controller Type	Sensors	
	file1.ino	newarduno1111	Arduino	1	
	file2.py	beagle1	Beaglebone Black	1	
	file3.py	raspberry_con	Raspberry Pi	2	
	file4.ino	intel11	Intel Galileo	4	

Εικόνα 5.22: Πίνακας Αρχείων Κώδικα

Στην εικόνα 5.22 παρουσιάζεται η σελίδα “My Files” η οποία περιέχει το πίνακα με τα αρχεία που έχει δημιουργήσει ο χρήστης.

Πατώντας οπουδήποτε πάνω σε μια γραμμή του πίνακα, η γραμμή επεκτείνεται, φανερώνοντας περισσότερες πληροφορίες για το αρχείο, όπως επίσης επιπλέον ενέργειες τις οποίες μπορεί να κάνει ο χρήστης για αυτό (εικόνα 5.23).

file4.ino	intel11	Intel Galileo	4
Time interval:		25 seconds	
Internet Connection:		No internet connection	
Sensors:	Name	Type	Pin
	motion1	Motion (PIR)	5
	soil1	Humidity (FC28)	A0
	light1	Light (LDR)	A2
	sound1	Sound (analog)	A1
Download Edit More Info Delete			

Εικόνα 5.23: Επεκτεινόμενη γραμμή πίνακα

Συγκεκριμένα, όπως φαίνεται και στην εικόνα, παρουσιάζονται πληροφορίες που αφορούν το χρόνο καθυστέρησης (time interval), αν η πλατφόρμα μικροελεγκτή είναι συνδεδεμένη στο διαδίκτυο και με ποιο τρόπο και ένας πίνακας με τους αισθητήρες που χρησιμοποιήθηκαν (όνομα αισθητήρα, τύπος και εισδοχή). Επιπλέον ο χρήστης έχει τη δυνατότητα να κατεβάσει το συγκεκριμένο αρχείο, να το επεξεργαστεί (χρόνος καθυστέρησης και εισδοχές αισθητήρων), να πάρει περισσότερες πληροφορίες για αυτό (λίστα υλικού, οδηγίες συναρμολόγησης κυκλώματος – εγκατάστασης κώδικα) και να το διαγράψει.

Κεφάλαιο 6

Σενάρια Χρήσης και Αξιολόγηση Μηχανής Παραγωγής Κώδικα

Σενάρια Χρήσης και Αξιολόγηση Μηχανής Παραγωγής Κώδικα.....	86
6.1 Σενάρια Χρήσης.....	87
6.1.1 Πρώτο Σενάριο Χρήσης.....	87
6.1.2 Δεύτερο Σενάριο Χρήσης.....	95
6.2 Σύνοψη Σεναρίων.....	101
6.3 Αξιολόγηση.....	103

6.1 Σενάρια Χρήσης

Σε αυτό το κεφάλαιο θα παρουσιαστούν δύο σενάρια χρήσης της μηχανής παραγωγής κώδικα με σκοπό να γίνει έλεγχος ότι όλα λειτουργούν ορθά. Το κάθε σενάριο αποτελεί όλα τα βήματα που ακολουθούνται από το σύστημα και από το χρήστη μέχρι τη φάση λειτουργίας του επιθυμητού κυκλώματος αισθητήρων. Τα βήματα αυτά είναι η συμπλήρωση της διαδραστικής φόρμας επιλογών, συναρμολόγηση κώδικα, συναρμολόγηση λίστας υλικού, συναρμολόγηση οδηγιών, συναρμολόγηση βοηθητικών διαγραμμάτων, λήψη αρχείου κώδικα, συναρμολόγηση κυκλώματος αισθητήρων και λειτουργία κυκλώματος.

6.1.1 Πρώτο Σενάριο Χρήσης

Το πρώτο σενάριο αποτελεί ένα κύκλωμα τριών αισθητήρων πάνω στην πλατφόρμα μικροελεγκτή Arduino Uno. Συγκεκριμένα, τοποθετούνται στην πλατφόρμα μικροελεγκτή ένας αισθητήρας κίνησης PIR, ένας αισθητήρας θερμοκρασίας DS18B20 και ένας αισθητήρας φωτεινότητας LDR, οι οποίοι λαμβάνουν μετρήσεις από το περιβάλλον τους κάθε 5 δευτερόλεπτα. Η πλατφόρμα μικροελεγκτή δεν είναι συνδεδεμένη στο διαδίκτυο και οι μετρήσεις των αισθητήρων εμφανίζονται στην οθόνη του χρήστη.

Το πρώτο βήμα αποτελεί η συμπλήρωση της διαδραστικής φόρμας επιλογών. Ο χρήστης επιλέγει το όνομα του αρχείου κώδικα (myfile1), το είδος της πλατφόρμας μικροελεγκτή (Arduino), το όνομα της (myArduinoUno), το χρόνο καθυστέρησης λήψης μετρήσεων (5 δευτερόλεπτα) και τη σύνδεση στο διαδίκτυο (No internet connection). Έπειτα ο χρήστης τοποθετεί τους προαναφερθέντες αισθητήρες, επιλέγοντας το είδος τους (Motion PIR, Temperature DS18B20, Light LDR), τα ονόματά τους (motion1, temperature1, light1) και τις εισδοχές στις οποίες θα τοποθετηθούν στην πλατφόρμα μικροελεγκτή (2,7,A0). Στην εικόνα 6.1 φαίνεται η συμπληρωμένη φόρμα επιλογών.

File information

File name
myfile1

Micro-controller

Select a micro-controller that already exists.

Micro-controller
myArduinoUno

Type
Arduino

Time Interval
5
Time to upload measurements in database (in seconds).

Internet Connection
No internet connection

Change

New Sensor

Sensor Type
Motion (PIR)

Sensor Name
motion1

Pin
2

Saved

Delete

New Sensor

Sensor Type
Temperature (DS18B20)

Sensor Name
temperature1

Pin
7

Saved

Delete

New Sensor

Sensor Type
Light (LDR)

Sensor Name
light1

Pin
A0

Saved

Delete

Add Sensor

GENERATE

Εικόνα 6.1: Διαδραστική Φόρμα Επιλογών (σενάριο 1)

Έπειτα πατώντας το κουμπί “GENERATE”, η φόρμα επιλογών ελέγχεται για τυχών λάθη και ο χρήστης μεταβαίνει σε μια νέα σελίδα. Στο πάνω μέρος της σελίδας αυτής υπάρχουν τέσσερις καρτέλες περιήγησης όπου η κάθε μια δίνει διαφορετικές πληροφορίες για το κύκλωμα των αισθητήρων που επιλέχθηκε.

Η πρώτη καρτέλα (File), η οποία είναι επιλεγμένη στην αρχή δίνει πληροφορίες που αφορούν το αρχείο κώδικα το οποίο έχει παραχθεί και των επιλογών που έχει κάνει ο χρήστης, η δεύτερη (Hardware) παρουσιάζει τη λίστα υλικού που χρειάζεται για την συναρμολόγηση του κυκλώματος, η τρίτη (Instructions) παρουσιάζει τις οδηγίες

εγκατάστασης και συναρμολόγησης του κυκλώματος και η τέταρτη (Diagrams) δίνει κάποια βοηθητικά διαγράμματα.

Στην εικόνα 6.2 φαίνεται οι πληροφορίες που δίνει η πρώτη καρτέλα περιήγησης.

[File](#) [Hardware](#) [Instructions](#) [Diagrams](#)

File Created!

File Name: myfile1.ino



[Download](#)

[Edit File](#)

Micro-controller

Name	Type	Internet Connection	Time Interval
myArduinoUno	Arduino	No internet connection	5

Sensors

#	Name	Type	Pin
1	light1	Light (LDR)	A0
2	motion1	Motion (PIR)	2
3	temperature1	Temperature (DS18B20)	7

Εικόνα 6.2: Πληροφορίες Αρχείου Κώδικα (σενάριο 1)

Όπως παρατηρείται, στο πάνω μέρος της σελίδας υπάρχει το κουμπί Download, όπου ο χρήστης μπορεί να κατεβάσει το αρχείο κώδικα για το Arduino, το κουμπί Edit, όπου μπορεί να αλλάξει το χρόνο καθυστέρησης και τις εισδοχές των αισθητήρων, ένας πίνακας με πληροφορίες για την πλατφόρμα μικροελεγκτή και τέλος ένας πίνακας με στοιχεία για τους τρεις αισθητήρες που αναφέρθηκαν πιο πάνω.

Στην εικόνα 6.3 φαίνεται η λίστα υλικού η οποία έχει δημιουργηθεί για το συγκεκριμένο σενάριο (δεύτερη καρτέλα περιήγησης). Η λίστα περιέχει 1 πλατφόρμα μικροελεγκτή Arduino μαζί με το USB καλώδιο για σύνδεση στον υπολογιστή, 1 Breadboard το οποίο βοηθάει στην συναρμολόγηση του κυκλώματος μακριά από την

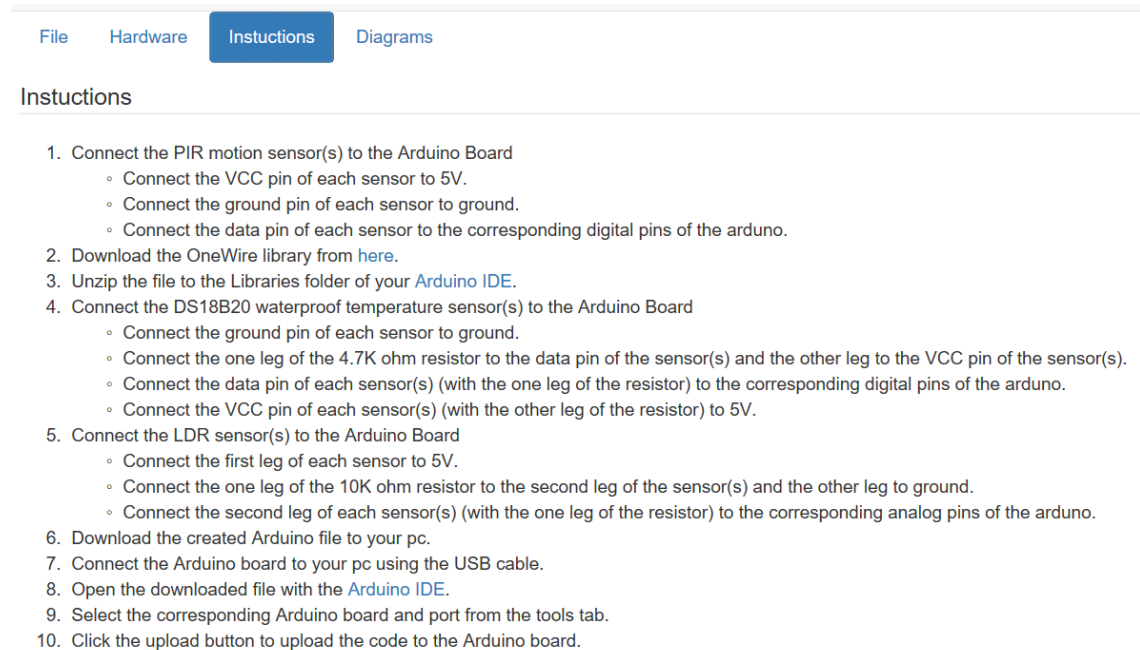
πλατφόρμα μικροελεγκτή, jumper καλώδια, 1 αισθητήρα κίνησης PIR, 1 αισθητήρα DS18B20, 1 αντιστάτη 4.7K Ohm (για το DS18B20), 1 αισθητήρα LDR και 1 αντιστάτη 10K Ohm (για τον LDR).



Εικόνα 6.3: Λίστα Υλικού (σενάριο 1)

Στην εικόνα 6.4 φαίνονται οι οδηγίες εγκατάστασης του κώδικα στην πλατφόρμα μικροελεγκτή Arduino Uno, καθώς και οδηγίες συναρμολόγησης του κυκλώματος των

αισθητήρων (τρίτη καρτέλα περιήγησης). Σημειώνεται ότι οι οδηγίες αυτές περιέχουν συνδέσμους όπου παραπέμπουν το χρήστη σε άλλες σελίδες για να δώσουν περισσότερες πληροφορίες ή για λήψη συγκεκριμένου εργαλείου ή βιβλιοθήκης (π.χ. βιβλιοθήκη OneWire).



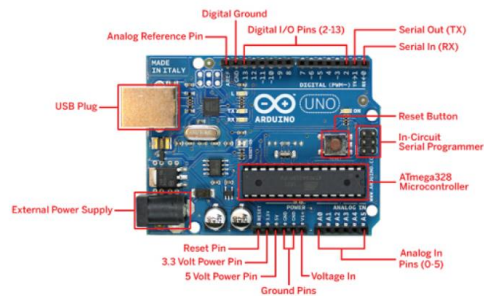
Εικόνα 6.4: Οδηγίες (σενάριο 1)

Στην εικόνα 6.5 φαίνονται τα διαγράμματα (τέταρτη καρτέλα περιήγησης), τα οποία δίνουν περισσότερες πληροφορίες για της πλατφόρμα μικροελεγκτή Arduino Uno και τους αισθητήρες Motion PIR, Temperature DS18B20 και Light LDR. Τα διαγράμματα αυτά μπορούν να συνδυαστούν με τις οδηγίες της τρίτης καρτέλας με σκοπό να βοηθήσουν το χρήστη να συναρμολογήσει το κύκλωμα των αισθητήρων.

Συγκεκριμένα, το πρώτο διάγραμμα περιγράφει τα βασικά στοιχεία της πλατφόρμας μικροελεγκτή Arduino Uno, το δεύτερο δίνει πληροφορίες για τα pins του αισθητήρα κίνησης PIR, το τρίτο δίνει πληροφορίες για τον αισθητήρα θερμοκρασίας DS18B20 και το τέταρτο δείχνει τον τρόπο σύνδεσης (κύκλωμα) του αισθητήρα φωτεινότητας LDR μαζί με τον αντιστάτη.

Arduino Uno

This diagram shows the different parts of the Arduino Uno board, like the digital I/O pins, analog in pins, ground pins, volt power pins and the USB plug.



PIR Motion Sensor

This diagram shows the 3 pins of the PIR motion sensor.

VCC: 3.3-5V DC for power

GND: Ground pin

Data: Digital output pin



DS18B20 Waterproof Temperature Sensor

This diagram shows the 3 wires of the DS18B20 waterproof temperature sensor.

VCC: 5V DC for power

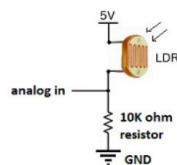
GND: Ground wire

Data: Digital output wire



Light Dependent Resistor (LDR)

This picture shows the circuit of the LDR with the 10k ohm resistor. The analog input goes to an analog in pin.



Εικόνα 6.5: Βοηθητικά Διαγράμματα (σενάριο 1)

Πατώντας το κουμπί Download (καρτέλα περιήγησης “File”), το αρχείο κώδικα κατεβαίνει στο προσωπικό υπολογιστή του χρήστη. Ακολούθως, ο χρήστης ανοίγει το αρχείο μέσω του Arduino IDE. Ο κώδικας που έχει παραχθεί από το σύστημα φαίνεται στην εικόνα 6.6.

```

#include <OneWire.h>
#include <DallasTemperature.h>

int light_pin1 = A0;
int light_value1;
int motion_pin1 = 2;
int motion_value1;
#define ONE_WIRE_BUS 7
float temperature_value1;
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);

void setup() {
  Serial.begin(9600);
  pinMode(light_pin1, INPUT);
  pinMode(motion_pin1, INPUT);
  sensors.begin();
}

void loop() {
  light_value1 = analogRead(light_pin1);
  Serial.print("Brightness = ");
  Serial.println(light_value1);

  motion_value1 = digitalRead(motion_pin1);
  Serial.print("Motion = ");
  Serial.println(motion_value1);

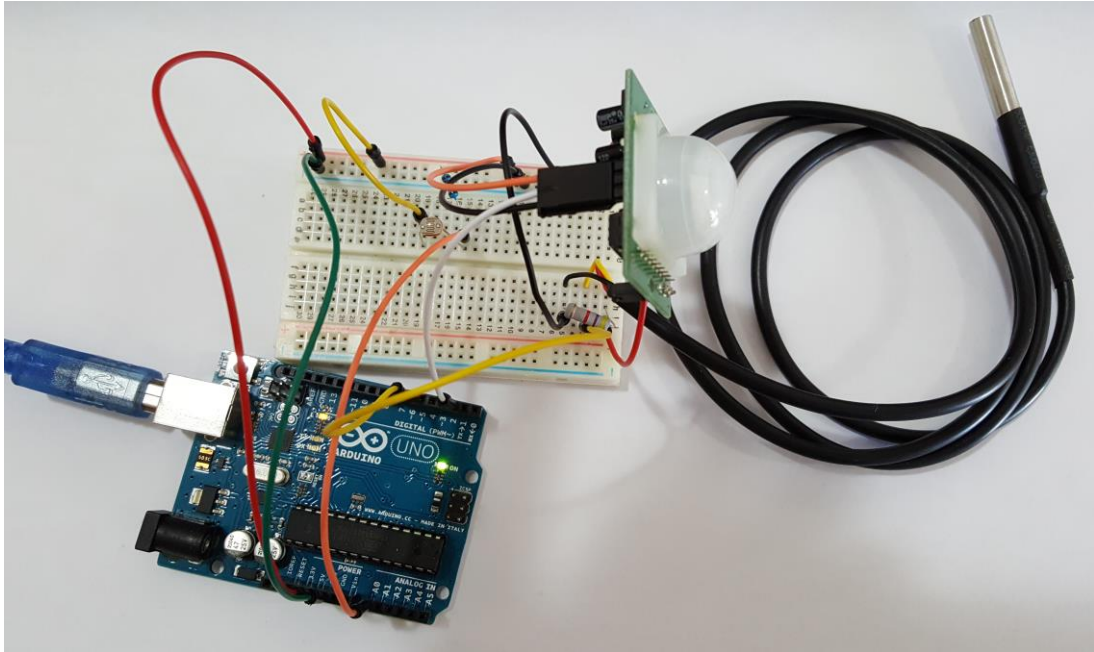
  Serial.println("Measuring temperature...");
  sensors.requestTemperatures();
  temperature_value1 = sensors.getTempCByIndex(0);
  Serial.print("Temperature = ");
  Serial.print(temperature_value1);
  Serial.println(" degrees celsius");

  Serial.println();
  delay(5000);
}

```

Εικόνα 6.6: Κώδικας (σενάριο 1)

Ακολουθώντας τις οδηγίες και τα βοηθητικά διαγράμματα, ο χρήστης μπορεί να συναρμολογήσει το κύκλωμα των αισθητήρων για το σενάριο 1. Στην εικόνα 6.7 φαίνεται το ολοκληρωμένο κύκλωμα της πλατφόρμας μικροελεγκτή Arduino Uno και των τριών αισθητήρων. Όπως φαίνεται και στην εικόνα ο αισθητήρας θερμοκρασίας DS18B20 (δεξιά) τοποθετήθηκε στην ψηφιακή εισδοχή 7, ο αισθητήρας κίνησης PIR (μέση) τοποθετήθηκε στην ψηφιακή εισδοχή 2 και ο αισθητήρας φωτεινότητας τοποθετήθηκε στην αναλογική εισδοχή A0. Τέλος, η πλατφόρμα μικροελεγκτή Arduino Uno συνδέθηκε μέσω του USB καλωδίου σε ηλεκτρονικό υπολογιστή.



Εικόνα 6.7: Κύκλωμα αισθητήρων (σενάριο 1)

Τέλος, ο κώδικας εγκαθίσταται στην πλατφόρμα μικροελεγκτή μέσω του Arduino IDE και εκτελείται. Στην εικόνα 6.8 παρουσιάζονται οι μετρήσεις που λαμβάνονται από τους αισθητήρες κάθε 5 δευτερόλεπτα.

```
Brightness = 1004
Motion = 1
Measuring temperature...
Temperature = 22.31 degrees celsius

Brightness = 1004
Motion = 0
Measuring temperature...
Temperature = 22.44 degrees celsius

Brightness = 979
Motion = 0
Measuring temperature...
Temperature = 22.62 degrees celsius
```

Εικόνα 6.8: Μετρήσεις αισθητήρων (σενάριο 1)

6.1.2 Δεύτερο Σενάριο Χρήσης

Το δεύτερο σενάριο αποτελεί ένα κύκλωμα δύο αισθητήρων πάνω στην πλατφόρμα μικροελεγκτή Raspberry Pi 2, Model B. Συγκεκριμένα, τοποθετούνται στην πλατφόρμα μικροελεγκτή ένας αισθητήρας κίνησης PIR και ένας αισθητήρας ήχου ψηφιακού σήματος, οι οποίοι λαμβάνουν μετρήσεις από το περιβάλλον τους κάθε 10 δευτερόλεπτα. Η πλατφόρμα μικροελεγκτή συνδέεται στο διαδίκτυο μέσω Wi-Fi adapter και οι μετρήσεις των αισθητήρων αποθηκεύονται στην βάση δεδομένων του συστήματος SensoMan.

Το πρώτο βήμα αποτελεί η συμπλήρωση της διαδραστικής φόρμας επιλογών. Ο χρήστης επιλέγει το όνομα του αρχείου κώδικα (myfile2), το είδος της πλατφόρμας μικροελεγκτή (Raspberry Pi), το όνομα της (myRaspberryPi), το χρόνο καθυστέρησης λήψης μετρήσεων (10 δευτερόλεπτα) και τη σύνδεση στο διαδίκτυο (Wi-Fi adapter). Έπειτα ο χρήστης τοποθετεί τους προαναφερθέντες αισθητήρες, επιλέγοντας το είδος τους (Motion PIR, Sound digital), τα ονόματά τους (motion1, sound1) και τις εισδοχές στις οποίες θα τοποθετηθούν στην πλατφόρμα μικροελεγκτή (GPIO_4, GPIO_5). Στην εικόνα 6.9 φαίνεται η συμπληρωμένη φόρμα επιλογών.

The image shows a web-based configuration form for the SensoMan system. It is divided into several sections:

- File information:** A text input field for 'File name' containing 'myfile2'.
- Micro-controller:** A section with the instruction 'Select a micro-controller that already exists.' It contains two dropdown menus: 'Micro-controller' (set to 'myRaspberryPi') and 'Type' (set to 'Raspberry Pi'). Below these are a 'Time Interval' input field (set to '10') with a note 'Time to upload measurements in database (in seconds)' and an 'Internet Connection' dropdown (set to 'Wi-Fi adapter'). A 'Change' link is at the bottom right of this section.
- New Sensor:** This section is repeated twice. Each instance has three dropdown menus: 'Sensor Type' (set to 'Motion (PIR)' and 'Sound (digital)' respectively), 'Sensor Name' (set to 'motion1' and 'sound1' respectively), and 'Pin' (set to 'GPIO_4' and 'GPIO_5' respectively). Each instance also has a blue 'Saved' button and a red 'Delete' link.
- Bottom:** An 'Add Sensor' button and a 'GENERATE' button.

Εικόνα 6.9: Διαδραστική Φόρμα Επιλογών (σενάριο 2)

Έπειτα πατώντας το κουμπί “GENARATE”, ελέγχεται η φόρμα επιλογών για τυχών λάθη και ο χρήστης μεταβαίνει σε νέα σελίδα, όπου του παρουσιάζονται οι πληροφορίες για το συγκεκριμένο πρόβλημα, με την ίδια δομή όπως στο σενάριο 1.

Στην εικόνα 6.10 φαίνονται οι επιλογές που έχει κάνει ο χρήστης για το συγκεκριμένο σενάριο (πρώτη καρτέλα περιήγησης), όπως το αρχείο κώδικα που παράχθηκε, η πλατφόρμα μικροελεγκτή και οι αισθητήρες.

[File](#) [Hardware](#) [Instuctions](#) [Diagrams](#)

File Created!

File Name: myfile2.py



[Download](#)

[Edit File](#)

Micro-controller

Name	Type	Internet Connection	Time Interval
myRaspberryPi	Raspberry Pi	Wi-Fi adapter	10

Sensors

#	Name	Type	Pin
1	motion1	Motion (PIR)	GPIO_4
2	sound1	Sound (digital)	GPIO_5

Εικόνα 6.10: Πληροφορίες Αρχείου Κώδικα (σενάριο 2)

Ακολούθως, στη δεύτερη καρτέλα περιήγησης (εικόνα 6.11) φαίνεται η λίστα με το υλικό που χρειάζεται για τη συναρμολόγηση του κυκλώματος αυτού του σεναρίου. Συγκεκριμένα, η λίστα αυτή αποτελείται από 1 πλατφόρμα μικροελεγκτή Raspberry Pi, 1 Wi-Fi adapter, 1 κάρτα SD χωρητικότητας 8 ή περισσότερων GBs, 1 breadboard, jumper καλώδια, 1 καλώδιο για επέκταση των εισδοχών GPIO της πλατφόρμας, 1 αισθητήρα κίνησης PIR και ένα αισθητήρα ήχου.

[File](#)[Hardware](#)[Instructions](#)[Diagrams](#)

Hardware

-  1 Raspberry Pi board + Power supply
-  1 Wi-Fi adapter
-  1 SD card (8+ GB)
-  1 Breadboard
-  Jumper wires
-  1 GPIO Extension cable (optional)
-  1 PIR motion sensor(s)
-  1 Sound sensor(s)

Εικόνα 6.11: Λίστα Υλικού (σενάριο 2)

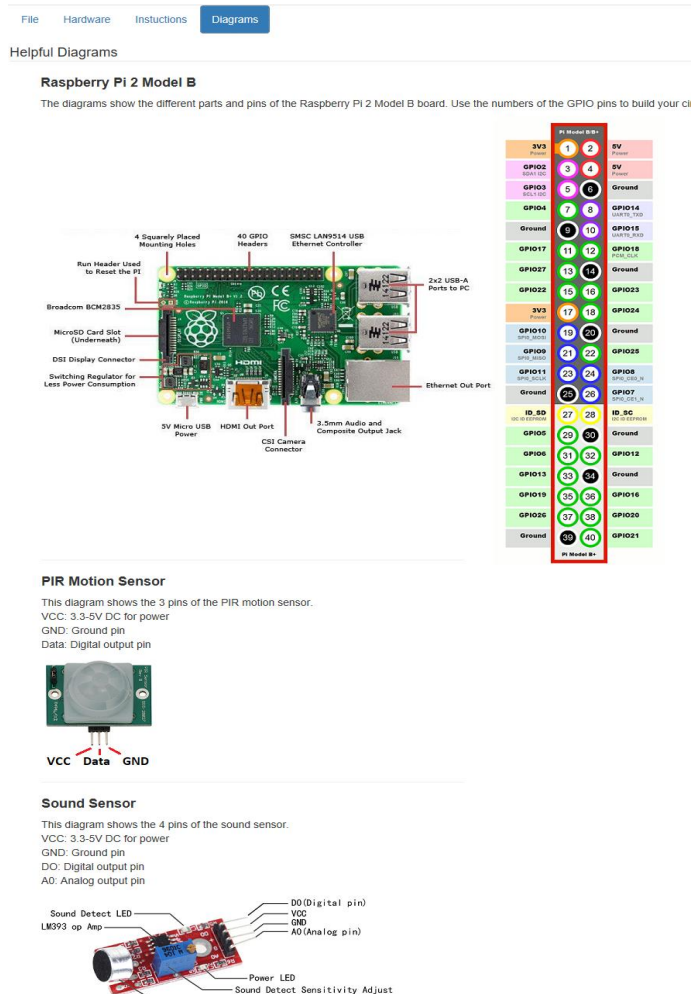
Στην εικόνα 6.12 φαίνεται η τρίτη καρτέλα περιήγησης, η οποία περιλαμβάνει τις οδηγίες συναρμολόγησης του κυκλώματος των αισθητήρων και εγκατάστασης του κώδικα στην πλατφόρμα μικροελεγκτή Raspberry Pi. Οι οδηγίες περιλαμβάνουν συνδέσμους προς επιπλέον πληροφορίες ή για εγκατάσταση εργαλείων – βιβλιοθηκών που χρειάζονται.

Instructions

1. Connect the PIR motion sensor(s) to the Raspberry Pi Board
 - Connect the VCC pin of each sensor to 5V (on board pins: 2,4).
 - Connect the ground pin of each sensor to ground (on board pins: 6,9,14,20,25,30,34,39).
 - Connect the data pin of each sensor to the corresponding GPIO pins of the raspberry pi.
2. Connect the sound sensor(s) to the Raspberry Pi Board
 - Connect the VCC pin of each sensor to 5V (on board pins: 2,4).
 - Connect the ground pin of each sensor to ground (on board pins: 6,9,14,20,25,30,34,39).
 - Connect the digital data pin of each sensor to the corresponding GPIO pins of the raspberry pi.
3. Download the created Raspberry Pi file to your pc.
4. Install an image of an operating system to your SD card. (More info [here](#))
5. Insert the SD card to Raspberry Pi.
6. Power up the board using the power supply.
7. Connect the board to the internet using the wifi adapter.
8. Connect to your Raspberry Pi board using [SSH](#). (default username: pi , password: raspberry)
9. Install the [Python packages](#) to the board.
10. Create a new python file (.py) to your board and copy the contents of the downloaded file.
11. Save and run the script.

Εικόνα 6.12: Οδηγίες (σενάριο 2)

Στην εικόνα 6.13 φαίνονται τα βοηθητικά διαγράμματα για αυτό το σενάριο.



Εικόνα 6.13: Βοηθητικά Διαγράμματα (σενάριο 2)

Στην εικόνα 6.14 παρουσιάζεται ο κώδικας, ο οποίος παράχθηκε από τη μηχανή παραγωγής κώδικα.

```
import time
import requests
import RPi.GPIO as GPIO

motion_pin1 = 4;
sound_pin1 = 5;

GPIO.setup(motion_pin1, GPIO.IN)
GPIO.setup(sound_pin1, GPIO.IN)
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)

def postMeasurement(sensorID, value):
    url = 'http://thesis.in.cs.ucy.ac.cy/sensoman2/upload/post_measurement.php'
    data = {'sensorID': str(sensorID), 'measurement': str(value)}
    req = requests.post(url, json = data)
    if (req.status_code==200):
        print('Measurement uploaded successfully!\n')
    else:
        print('Unable to upload the measurement.\n')

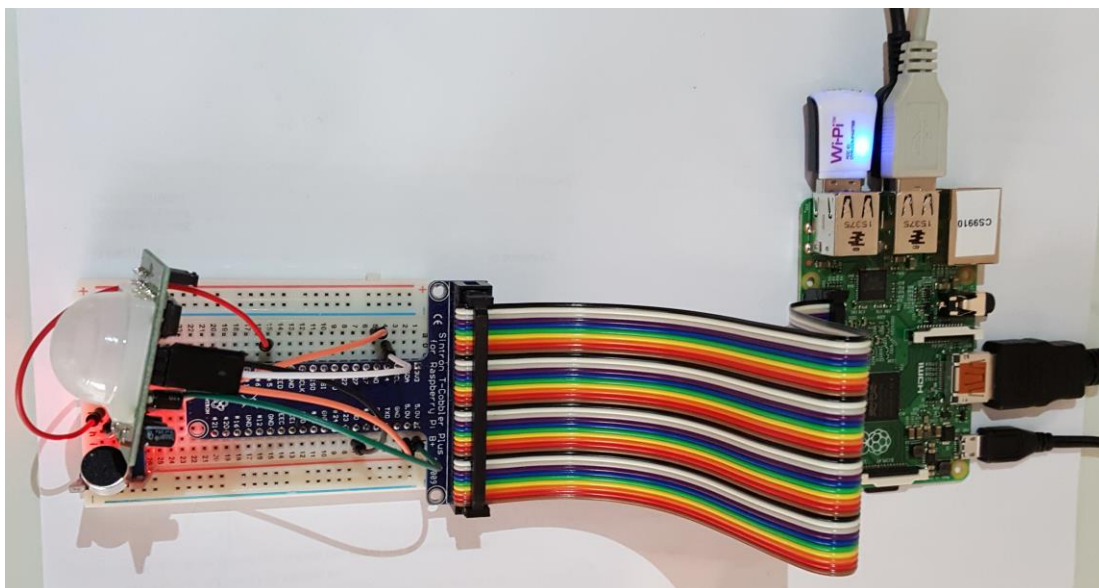
while True:
    motion_value = GPIO.input(motion_pin1)
    print ('Motion = ',motion_value,'\n')
    postMeasurement(225, motion_value)

    sound_value = GPIO.input(sound_pin1)
    print ('Sound = ',sound_value,'\n')
    postMeasurement(228, sound_value)

    time.sleep(10)
```

Εικόνα 6.14: Κώδικας (σενάριο 2)

Ακολουθώντας τις οδηγίες και τα βοηθητικά διαγράμματα, ο χρήστης μπορεί να συναρμολογήσει το κύκλωμα των αισθητήρων για το σενάριο 2. Στην εικόνα 6.15 φαίνεται το ολοκληρωμένο κύκλωμα της πλατφόρμας μικροελεγκτή Raspberry Pi και των δύο αισθητήρων. Η πλατφόρμα μικροελεγκτή συνδέεται με πληκτρολόγιο και ποντίκι μέσω των USB θυρών που περιέχει και με εξωτερική οθόνη μέσω καλωδίου HDMI. Επίσης, επεκτείνονται οι GPIO εισδοχές της πλατφόρμας μέσω καλωδίου πάνω σε breadboard και η σύνδεση της στο διαδίκτυο επιτυγχάνεται μέσω Wi-Fi adapter. Όπως φαίνεται και στην εικόνα, ο αισθητήρας ήχου (κάτω αριστερά) τοποθετήθηκε στην ψηφιακή εισδοχή 5 και ο αισθητήρας κίνησης PIR (πάνω αριστερά) τοποθετήθηκε στην ψηφιακή εισδοχή 4.



Εικόνα 6.15: Κύκλωμα αισθητήρων (σενάριο 2)

Τέλος, ο κώδικας που παράγεται από τη μηχανή παραγωγής κώδικα γράφεται και εκτελείται μέσω του λογισμικού Thonny [32], το οποίο είναι ένα IDE για τη γλώσσα προγραμματισμού Python. Σημειώνεται ότι το αρχείο κώδικα μπορεί να εκτελεστεί και μέσω τερματικού (terminal), εκτελώντας την εντολή “python myfile2.py”. Στην εικόνα 6.16 παρουσιάζονται οι μετρήσεις που λαμβάνονται από τους δύο αισθητήρες κάθε 10 δευτερόλεπτα.

```
Shell
>>> %Run myfile2.py
Motion = 1
Sound = 1
Motion = 0
Sound = 1
Motion = 0
Sound = 1
```

Εικόνα 6.16: Μετρήσεις αισθητήρων (σενάριο 2)

Οι μετρήσεις των αισθητήρων αποθηκεύονται στη βάση δεδομένων του συστήματος, μέσω κλήσης της RESTful υπηρεσίας του SensoMan. Η εικόνα 6.17 αντιπροσωπεύει ένα στιγμιότυπο οθόνης της βάσης δεδομένων, όπου φαίνονται αποθηκευμένες οι μετρήσεις των δύο αισθητήρων. Ο αισθητήρας κίνησης έχει id με αριθμό 225 και ο αισθητήρας ήχου έχει id με αριθμό 228. Παρατηρείται ότι οι μετρήσεις των αισθητήρων αποθηκεύονται κάθε 10 δευτερόλεπτα, όπως έχει οριστεί από την αρχή.

measurementID	sensorID	measurement	datetime
9297	228	1	2018-05-22 22:58:33
9296	225	0	2018-05-22 22:58:33
9295	228	0	2018-05-22 22:58:23
9294	225	0	2018-05-22 22:58:22
9293	228	0	2018-05-22 22:58:12
9292	225	1	2018-05-22 22:58:12
9291	228	1	2018-05-22 22:58:02
9290	225	1	2018-05-22 22:58:02

Εικόνα 6.17: Αποθήκευση μετρήσεων στη βάση δεδομένων (σενάριο 2)

6.2 Σύνοψη Σεναρίων

Σε αυτό το κεφάλαιο παρουσιάστηκαν δύο σενάρια χρήσης της μηχανής παραγωγής κώδικα, η οποία δημιουργήθηκε στα πλαίσια της παρούσας διπλωματικής εργασίας. Όπως παρατηρήθηκε στα σενάρια αυτά, η μηχανή βάσει των επιλογών που κάνει ο χρήστης μέσω της διαδραστικής φόρμας επιλογών, παράγει τα επιθυμητά αποτελέσματα (αρχείο κώδικα, οδηγίες χρήσης, λίστα υλικού, βοηθητικά διαγράμματα) προσαρμοσμένα στο συγκεκριμένο σενάριο κάθε φορά.

Τα αποτελέσματα αυτά “κτίζονται” δυναμικά ανάλογα με το είδος της πλατφόρμας μικροελεγκτή και τους αισθητήρες που επιλέγονται. Στο πρώτο σενάριο παρουσιάστηκε ένα κύκλωμα τριών αισθητήρων (κίνησης, θερμοκρασίας, φωτεινότητας) βασισμένο στην πλατφόρμα μικροελεγκτή Arduino. Η μηχανή παραγωγής κώδικα παρήγαγε αρχείο τύπου Arduino Sketch File με κατάληξη .ino, το οποίο μπορεί να ανοίξει στο γραφικό περιβάλλον του Arduino IDE. Λόγω του ότι

αποφασίστηκε η πλατφόρμα μικροελεγκτή να μην συνδεθεί στο διαδίκτυο, ο κώδικας ο οποίος δημιουργήθηκε, έβαζε σε λειτουργία τους αισθητήρες και εμφάνιζε τις μετρήσεις τους στην οθόνη του χρήστη.

Το δεύτερο σενάριο αποτελούσε ένα κύκλωμα δύο αισθητήρων (κίνησης και ήχου) βασισμένο στην πλατφόρμα μικροελεγκτή Raspberry Pi. Η μηχανή παραγωγής κώδικα παρήγαγε αρχείο κώδικα στην γλώσσα προγραμματισμού Python, με κατάληξη .py. Λόγω του ότι η πλατφόρμα συνδέθηκε στο διαδίκτυο με Wi-Fi adapter, ο κώδικας συμπεριλάμβανε μια συνάρτηση, η οποία επικοινωνούσε με τη RESTful υπηρεσία διαδικτύου του συστήματος SensoMan με σκοπό να αποθηκευτούν οι μετρήσεις των αισθητήρων στη βάση δεδομένων. Για να γίνει αυτό εφικτό, ορίστηκε μέσα στο κώδικα του προγράμματος μια μεταβλητή η οποία κρατούσε το URL της υπηρεσίας διαδικτύου που εξυπηρετούσε το σκοπό αυτό. Επίσης, η κάθε μέτρηση συνοδευόταν με το μοναδικό αριθμό (id) του συγκεκριμένου αισθητήρα, έτσι οι μετρήσεις αποθηκεύονταν στη βάση δεδομένων υπό τη μορφή ζευγών τιμής μέτρησης και id αισθητήρα.

Τέλος, τα δύο αρχεία έχουν αποθηκευτεί στο προσωπικό λογαριασμό του χρήστη, στο σύστημα SensoMan. Ο χρήστης μπορεί να δει όλα τα αρχεία που έχει δημιουργήσει μεταβαίνοντας στη σελίδα “My Files” και από εκεί μπορεί να τα κατεβάσει ξανά, να τα επεξεργαστεί, να δει πληροφορίες για αυτά και να τα διαγράψει. Στην εικόνα 6.18 φαίνονται τα δύο αρχεία των πιο πάνω σεναρίων.

My Files

	File Name	Controller Name	Controller Type	Sensors	
	myfile1.ino	myArduinoUno	Arduino	3	
	myfile2.py	myRaspberryPi	Raspberry Pi	2	

Εικόνα 6.18: Αρχεία σεναρίων στην σελίδα “My Files”

6.3 Αξιολόγηση

Η αξιολόγηση της μηχανής παραγωγής κώδικα έγινε από τρεις διαφορετικούς χρήστες. Συγκεκριμένα, δόθηκε όλο το υλικό (τέσσερις πλατφόρμες μικροελεγκτή, έξι αισθητήρες, καλώδια) στους χρήστες και ο κάθε ένας δημιούργησε ένα διαφορετικό κύκλωμα της επιλογής του, χρησιμοποιώντας μια πλατφόρμα μικροελεγκτή και μερικούς αισθητήρες.

Αρχικά, οι χρήστες κληθήκαν να συναρμολογήσουν το κύκλωμα αισθητήρων και να γράψουν το κώδικα για λήψη μετρήσεων, μέσω πληροφοριών που υπάρχουν ελεύθερες στο διαδίκτυο και έπειτα μέσω χρήσης της μηχανής παραγωγής κώδικα. Και στις δύο περιπτώσεις μετρήθηκε ο χρόνος που χρειάστηκε να ολοκληρώσουν τη διαδικασία, με σκοπό να εξαχθούν συμπεράσματα για τη χρησιμότητα της μηχανής.

Όπως παρατηρήθηκε από το πείραμα αυτό, ο χρόνος συναρμολόγησης του καθ' ενός από τα κυκλώματα που επιλέχθηκαν βασιζόταν στο γεγονός αν ο χρήστης είχε προηγούμενη γνώση από πλατφόρμες μικροελεγκτή και αν είχε προγραμματιστικό υπόβαθρο. Ο ένας από τους τρεις χρήστες, ο οποίος είχε μερικές γνώσεις προγραμματισμού χρειάστηκε περισσότερο χρόνο από τους άλλους δύο οι οποίοι είχαν υψηλό υπόβαθρο. Επίσης, ο χρόνος επηρεαζόταν και ανάλογα με το υλικό που είχε χρησιμοποιηθεί. Για παράδειγμα, οι πληροφορίες που υπάρχουν στο διαδίκτυο για κάποιους από τους αισθητήρες είναι πιο δύσκολο να βρεθούν και να γίνουν κατανοητές, σε σύγκριση με πληροφορίες για κάποιους άλλους αισθητήρες.

Κατά μέσο όρο, παρατηρήθηκε ότι ο χρόνος που χρειάζεται για τη συναρμολόγηση του κυκλώματος και εγγραφής του κώδικα με χρήση της μηχανής παραγωγής κώδικα είναι περίπου τέσσερις φορές μικρότερος έναντι του χρόνου της ίδιας διαδικασίας με χρήση πληροφοριών από το διαδίκτυο. Το γεγονός αυτό δείχνει τη σημαντικότητα της μηχανής παραγωγής κώδικα και την μεγάλη επίδραση της στη μείωση του χρόνου της διαδικασίας λειτουργίας ενός κυκλώματος αισθητήρων.

Κεφάλαιο 7

Συμπεράσματα και Μελλοντικές Επεκτάσεις

Συμπεράσματα και Μελλοντικές Επεκτάσεις.....	104
7.1 Συμπεράσματα.....	105
7.2 Μελλοντικές Επεκτάσεις.....	106

7.1 Συμπεράσματα

Στις μέρες μας, οι εφαρμογές γύρο από το πεδίο του Δικτύου των Πραγμάτων (Internet of Things) αποκοτούν ολοένα και μεγαλύτερη δημοσιότητα. Ο αριθμός των πλατφορμών μικροελεγκτών και αισθητήρων έχει αυξηθεί δραματικά τα τελευταία χρόνια με σκοπό την εξυπηρέτηση των αναγκών του ανθρώπου. Το σύστημα SensoMan αποτελεί ένα βασικό σύστημα στην διαχείριση και απομακρυσμένο έλεγχο αισθητήρων. Αποτελεί το θεμέλιο λίθο για δημιουργία μικρών αλλά και πολύπλοκων εφαρμογών βασισμένες σε αισθητήρες, οι οποίοι συλλέγουν πληροφορίες από το περιβάλλον στο οποίο βρίσκονται και αυτοματοποιούν διαδικασίες με ελάχιστη ή μηδενική ανθρώπινη παρέμβαση.

Η δημιουργία και προσθήκη της μηχανής παραγωγής κώδικα στο SensoMan έχει αυξήσει σημαντικά την χρησιμότητα του συστήματος και ανοίγει τις πόρτες για τη δημιουργία αμέτρητων εφαρμογών κάτω από το πλαίσιο του Δικτύου των Πραγμάτων. Όπως παρατηρήθηκε και στο προηγούμενο κεφάλαιο, επιταχύνει τη διαδικασία δημιουργίας και λειτουργίας κυκλωμάτων αισθητήρων για λήψη μετρήσεων (περίπου κατά τέσσερις φορές έναντι της διαδικασίας χωρίς τη χρήση της μηχανής), αφού το μόνο που έχει να κάνει ο χρήστης είναι να δώσει τις επιλογές του και το σύστημα θα τον κατευθύνει στην επίτευξη του σκοπού αυτού. Η χρήση της δεν απαιτεί εξειδικευμένες γνώσεις προγραμματισμού, αφού το αρχείο κώδικα δίνεται έτοιμο από το σύστημα. Το γεγονός αυτό, επιτρέπει τη συναρμολόγηση κυκλωμάτων αισθητήρες και από χρήστη με χαμηλό προγραμματιστικό υπόβαθρο.

Η μηχανή αυτή μαζί με τα υπόλοιπα κομμάτια του συστήματος SensoMan προσφέρουν ένα πιο ολοκληρωμένο σύστημα προς το χρήστη και συμβάλλουν στην προώθηση καινοτομικών λύσεων σε προβλήματα που απασχολούν την καθημερινότητα του ανθρώπου.

Για τη δημιουργία της μηχανής παραγωγής κώδικα χρειάστηκε να γίνουν αλλαγές τόσο στην διαδικτυακή εφαρμογή του συστήματος, όπως για παράδειγμα στις όψεις

(front end), όσο και στην βάση δεδομένων και RESTful υπηρεσία διαδικτύου (back end). Προστέθηκαν δύο επιπλέον πίνακες στη βάση δεδομένων με σκοπό την αποθήκευση των αρχείων κώδικα που παράγονται, καθώς και αρκετές επιπλέον υπηρεσίες για την εξυπηρέτηση των λειτουργιών της μηχανής.

Τα σενάρια χρήσης του προηγούμενου κεφαλαίου δείχνουν βήμα προς βήμα όλες τις λειτουργίες που προστέθηκαν στο σύστημα μέσω της μηχανής παραγωγής κώδικα και αποδεικνύουν την ορθή λειτουργία της. Παρ' όλα αυτά η παρούσα κατάσταση της μηχανής είναι μόνο η αρχή για κάτι μεγαλύτερο. Οι δυνατότητες επέκτασης της αναλύονται στο επόμενο υποκεφάλαιο.

7.2 Μελλοντικές Επεκτάσεις

Μέχρι στιγμής η μηχανή παραγωγής κώδικα υποστηρίζει τέσσερις πλατφόρμες μικροελεγκτών (Arduino, Raspberry Pi, Beaglebone Black, Intel Galileo) και έξι αισθητήρες (κίνησης PIR, θερμοκρασίας DS18B20, θερμοκρασίας TMP36, υγρασίας FC-28, ήχου, φωτεινότητας LDR). Εντούτοις υπάρχουν πάρα πολλοί αισθητήρες και πλατφόρμες μικροελεγκτών που μπορούν να προστεθούν και να υποστηρίζονται από τη μηχανή.

Υπάρχουν πλατφόρμες μικροελεγκτών πέραν των τεσσάρων προαναφερθέντων, οι οποίες κατασκευάζονται από άλλες εταιρίες. Επίσης, η μηχανή μπορεί να επεκταθεί και με πλατφόρμες που ανήκουν στην ίδια κατηγορία με αυτές τις τέσσερις. Για παράδειγμα, η πλατφόρμα μικροελεγκτή Arduino αποτελείται από διαφορετικά μοντέλα πέραν της Arduino Uno, όπως την Arduino Mega, Arduino Nano και Arduino Micro κ.α. Οι πλατφόρμες αυτές διαφέρουν στα χαρακτηριστικά και στις εισδοχές τις οποίες έχουν, όμως ο κώδικας που χρειάζονται για να λειτουργήσουν είναι ο ίδιος. Έτσι, μπορούν να ενταχθούν στη μηχανή παραγωγής κώδικα πολύ εύκολα, χωρίς να αλλάξει ο κώδικας ο οποίος παράγεται, παρά μόνο οι εισδοχές τις οποίες μπορεί να επιλέξει ο χρήστης στην διαδραστική φόρμα επιλογών.

Το ίδιο συμβαίνει και με την πλατφόρμα μικροελεγκτή Raspberry Pi. Στην παρούσα διπλωματική εργασία χρησιμοποιήθηκε η πλατφόρμα Raspberry Pi 2 Model B αλλά υπάρχουν περισσότερα μοντέλα της ίδιας πλατφόρμας, όπως για παράδειγμα η Raspberry Pi 1 Model A, η Raspberry Pi 1 Model B και η Raspberry Pi 3 Model B. Οι πλατφόρμες αυτές, όπως εξηγήθηκε και στο κεφάλαιο 3 διαφέρουν στα χαρακτηριστικά και στον αριθμό των εισδοχών τους αλλά ο κώδικας που χρησιμοποιείται για λήψη μετρήσεων από αισθητήρες είναι ο ίδιος (Python). Άρα η προσθήκη τους στην μηχανή παραγωγής κώδικα είναι επίσης απλή.

Μια ακόμα επέκταση που μπορεί να γίνει είναι η προσθήκη συσκευών ή εξαρτημάτων πέραν από αισθητήρες. Τα εξαρτήματα αυτά μπορεί να είναι οποιεσδήποτε έξυπνες συσκευές μέσα σε ένα έξυπνο σπίτι που μπορούν να ελεγχθούν από πλατφόρμες μικροελεγκτών ή ακόμα απλά εξαρτήματα όπως λάμπες LED και buzzers τα οποία μπορούν να χρησιμοποιηθούν σε συνδυασμό με τους αισθητήρες. Για παράδειγμα, μπορεί να δίνεται η δυνατότητα στο χρήστη, την ώρα προσθήκης ενός νέου αισθητήρα στη φόρμα επιλογών, να προσθέσει μια λάμπα LED ή ένα buzzer στο κύκλωμα του, το οποίο θα ενεργοποιείται αν η μέτρηση του συγκεκριμένου αισθητήρα ξεπεράσει κάποιο όριο που θα θέτει ο χρήστης. Έτσι, με αυτό τον τρόπο το φάσμα εφαρμογών που μπορούν να δημιουργηθούν μέσω της μηχανής αυξάνεται δραματικά.

Τέλος, σε αυτό το σημείο αξίζει να σημειωθεί ότι η αρχιτεκτονική της μηχανής παραγωγής κώδικα χαρακτηρίζεται ως αρθρωτή και επεκτάσιμη. Αυτό σημαίνει ότι υπάρχει ξεκάθαρη δόμηση των στοιχείων που την απαρτίζουν και έτσι η προσθήκη επιπλέον λειτουργιών και εξαρτημάτων μπορεί να γίνει χωρίς πολλές αλλαγές σε διάφορα σημεία του προγράμματος.

Βιβλιογραφία

- [1] Wikipedia. "HTML" [Online] Available at: <https://en.wikipedia.org/wiki/HTML>

- [2] Wikipedia. "PHP" [Online] Available at: <https://el.wikipedia.org/wiki/PHP>

- [3] Wikipedia. "JavaScript" [Online] Available at: <https://el.wikipedia.org/wiki/JavaScript>

- [4] Oracle. 2018. "MySQL" [Online] Available at: <http://www.oracle.com/technetwork/database/mysql/index.html>

- [5] Oracle 2013. "The Java EE 6 Tutorial - What Are RESTful Web Services?" [Online] Available at: <https://docs.oracle.com/javaee/6/tutorial/doc/gijqy.html>

- [6] Jacob Morgan. 2018. Forbes "A Simple Explanation Of 'The Internet Of Things'" [Online] Available at: <https://www.forbes.com/sites/jacobmorgan/2014/05/13/simple-explanation-internet-things-that-anyone-can-understand/#7178e1321d09>

- [7] "Μικροελεκτές" [Online] Available at: <http://www.eln.teilam.gr/sites/default/files/Lesson03.pdf>

- [8] Ιωάννης Καλομοίρος. TEI Κεντρικής Μακεδονίας. 2015 "Εισαγωγή στους μικροελεγκτές PIC" [Online] Available at: http://teachers.teicm.gr/kalomiros/Mtptx/e-books/Embedded_PIC_new.pdf

- [9] openHAB Official Site. [Online] Available at: <https://docs.openhab.org/index.html>

- [10] Wikipedia. “Αισθητήρας” [Online] Available at:
<https://el.wikipedia.org/wiki/Αισθητήρας>
- [11] Arduino. “What is Arduino” [Online] Available at:
<https://www.arduino.cc/en/Guide/Introduction>
- [12] B_E_N. SparkFun Electronics. “What is an Arduino?” [Online] Available at:
<https://learn.sparkfun.com/tutorials/what-is-an-arduino>
- [13] Microsoft 2018. “Uploading Board Code and Arduino IDE” [Online] Available at:
<https://support.office.com/en-us/article/uploading-board-code-and-arduino-ide-a9723765-1314-49e0-a69b-bb5c3e1f628d>
- [14] Wikipedia. “Raspberry Pi” [Online] Available at:
https://en.wikipedia.org/wiki/Raspberry_Pi
- [15] Raspberry Pi Foundation. “Raspberry Pi 2 Model B” [Online] Available at:
<https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>
- [16] Wikipedia. “BeagleBoard” [Online] Available at:
<https://en.wikipedia.org/wiki/BeagleBoard>
- [17] toptechboy.com. June 2015. “Pinout for Beaglebone Black” [Online] Available at:
<http://www.toptechboy.com/beaglebone-black-rev-c/beaglebone-black-lesson-1-understanding-beaglebone-black-pinout/attachment/beaglebone-black-pinout/>

- [18] Arduino. "Intel Galileo" [Online] Available at:
<https://www.arduino.cc/en/ArduinoCertified/IntelGalileo>

- [19] lady ada. Adafruit. January 2014. "How PIRs Work" [Online] Available at:
<https://learn.adafruit.com/pir-passive-infrared-proximity-motion-sensor/how-pirs-work> [last updated September, 2016]

- [20] Wikipedia. "Photoresistor" [Online] Available at:
<https://en.wikipedia.org/wiki/Photoresistor>

- [21] PotentialLabs. "LDR - Light Dependent Resistor - Large" [Online] Available at:
<https://potentiallabs.com/cart/ldr-big>

- [22] Konstantin Dimitrov. November 2016. "DS18B20 (digital temperature sensor) and Arduino" [Online] Available at:
<https://create.arduino.cc/projecthub/TheGadgetBoy/ds18b20-digital-temperature-sensor-and-arduino-9cc806>

- [23] lady ada. Adafruit. July, 2012. "TMP36 Temperature Sensor" [Online] Available at: <https://learn.adafruit.com/tmp36-temperature-sensor/overview> [last updated December, 2017]

- [24] instructables. "Arduino - Sound Sensor (with LED)" [Online] Available at:
<http://www.instructables.com/id/Arduino-Sound-Sensor-with-LED/>

- [25] Ali Hamza. "How to Test Soil With Arduino and an FC-28 Moisture Sensor" [Online] Available at: <https://maker.pro/arduino/projects/arduino-soil-moisture-sensor>

- [26] Rick Lesniak. Adafruit. September, 2013. "Adafruit CC3000 WiFi" [Online]
Available at: <https://learn.adafruit.com/adafruit-cc3000-wifi/cc3000-shield>
[last updated May, 2015]

- [27] Github. "timofurrer/w1thermsensor" Available at:
<https://github.com/timofurrer/w1thermsensor>

- [28] Final Report on System Specification - FP7 i-Treasures Deliverable 2.4 -
Scientific Figure on ResearchGate. Available from:
https://www.researchgate.net/RESTful-Web-Service_fig21_289479402
[accessed 20 May, 2018]

- [29] Άγγελος Ευσταθίου. 2017. "Δημιουργία κοινωνικού δικτύου για χειρισμό
αισθητήρων μέσω Arduino". BSc Dissertation, University of Cyprus

- [30] Φίλιππος Αζίζ. 2016. "Δημιουργία μηχανής κανόνων για αυτοματοποιημένο
έλεγχο συσκευών βάσει δεδομένων από αισθητήρες." BSc Dissertation,
University of Cyprus

- [31] Arduino. "Getting Started with the Arduino Ethernet Shield and Ethernet Shield
2" [Online] Available at:
<https://www.arduino.cc/en/Guide/ArduinoEthernetShield>

- [32] Thonny. "Python IDE for beginners" [Online] Available at: <http://thonny.org/>